

**PROJECT UAS**

**PERANCANGAN DAN IMPLEMENTASI SISTEM BASIS DATA INVOICE TOP UP  
GAME**

*Laporan ini dibuat guna memenuhi Ujian Akhir Semester kuliah Pemograman Basis Data yang  
diampu oleh:*

*Ridwan Dwi Irawan, S. Kom., M. Kom.*



**UNIVERSITAS  
DUTA BANGSA  
SURAKARTA**

Disusun Oleh:

1. Ivan Yosa Andriyanto (240103165)
2. Kholiq Nur Cahyo (240103165)
3. Radja Zain Nuur Rofiif (240130111)

**PROGRAM STUDI S1 TEKNIK INFORMATIKA**

**FAKULTAS ILMU KOMPUTER**

**UNIVERSITAS DUTA BANGSA**

**2026**

## DAFTAR ISI

DAFTAR ISI.....	ii
BAB I.....	4
PENDAHULUAN .....	4
1.1. Latar Belakang.....	4
1.2. Tujuan Pembuatan Sistem.....	4
1.3. Ruang Lingkup dan Batasan Masalah.....	4
1.4. Gambaran Umum Sistem Invoice Top Up Game .....	5
BAB II.....	6
LANDASAN TEORI.....	6
2.1 Konsep Basis Data dan DBMS .....	6
2.2 Relasi Tabel.....	6
2.3 Entity Relationship Diagram (ERD).....	6
2.4 Normalisasi Data.....	7
2.5 Primary Key dan Foreign Key .....	7
2.6 MySQL Workbench .....	8
2.7 DDL, DML, dan TCL .....	8
2.8 Query JOIN, GROUP BY, HAVING, dan Agregasi .....	8
BAB III .....	9
PERRANCANGAN DAN IMPLEMENTASI .....	9
3.1. Analisis Kebutuhan Sistem Invoice .....	9
3.2. Perancangan Basis Data .....	9
3.2.1. ERD Sistem Top Up Game .....	10
3.2.2. Desain Tabel.....	11
3.2.3. Relasi Primary Key dan Foreign Key .....	12
3.3. Implementasi Database .....	12
3.3.1. DDL .....	12
3.3.2. DML.....	13
3.3.3. TCL .....	13
3.4. Implementasi Query.....	14

3.4.1.	Query Agregasi dan HAVING .....	14
3.4.2.	Query GROUP BY .....	15
3.4.3.	Query JOIN .....	16
BAB IV	.....	17
PENUTUP	.....	17
4.1.	Hasil Pengujian Sistem .....	17
4.2.	Kendala dan Solusi .....	17
4.3.	Kesimpulan .....	18
4.4.	Saran Pengembangan .....	18
LAMPIRAN	.....	19

## **BAB I**

### **PENDAHULUAN**

#### **1.1.Latar Belakang**

Dalam perkembangan industri game online yang semakin pesat, layanan top up menjadi kebutuhan utama bagi para pemain untuk memperoleh item virtual. Proses transaksi yang tidak dikelola dengan sistem basis data yang terstruktur dapat menimbulkan berbagai permasalahan, seperti kesalahan pencatatan, duplikasi data, serta kesulitan dalam pembuatan invoice dan laporan transaksi. Oleh karena itu, diperlukan suatu sistem yang mampu mengelola data pelanggan, produk top up, metode pembayaran, dan transaksi secara terintegrasi sehingga informasi yang dihasilkan lebih akurat dan mudah diakses.

Pemanfaatan sistem basis data pada pembuatan invoice top up game memungkinkan proses pencatatan transaksi dilakukan secara otomatis dan konsisten. Dengan perancangan database yang baik melalui penggunaan ERD, normalisasi, serta penerapan query SQL seperti JOIN, GROUP BY, dan agregasi, sistem dapat menghasilkan invoice dan laporan transaksi secara cepat dan tepat. Proyek ini bertujuan untuk merancang dan mengimplementasikan sistem invoice top up game sebagai bentuk penerapan konsep pemrograman basis data dalam menyelesaikan permasalahan nyata pada layanan digital.

#### **1.2.Tujuan Pembuatan Sistem**

Tujuan dari pembuatan proyek sistem invoice top up game ini adalah untuk merancang dan mengimplementasikan basis data yang mampu mengelola data pengguna, produk top up, transaksi, dan pembayaran secara terstruktur serta menghasilkan invoice secara otomatis, sehingga dapat meminimalkan kesalahan pencatatan, meningkatkan keakuratan informasi, dan memudahkan proses pelaporan sebagai penerapan konsep pemrograman basis data seperti perancangan ERD, normalisasi, serta penggunaan query SQL (DDL, DML, JOIN, GROUP BY, dan agregasi).

#### **1.3.Ruang Lingkup dan Batasan Masalah**

Tugas ini merupakan tugas kelompok yang menghasilkan satu proyek pemrograman basis data berbasis studi kasus. Studi kasus boleh dipilih bebas selama relevan dengan materi perkuliahan, misalnya sistem penjualan, inventori, perpustakaan, klinik, akademik, reservasi, atau layanan publik. Proyek harus menunjukkan proses perancangan basis data, implementasi basis data, pembuatan query, serta penggunaan basis data melalui program atau antarmuka yang digunakan pada praktik perkuliahan. Setiap kelompok wajib menunjukkan pembagian peran dan kontribusi anggota secara jelas di dalam laporan.

#### 1.4. Gambaran Umum Sistem Invoice Top Up Game

Sistem invoice top up game yang dikembangkan dalam proyek ini merupakan sebuah sistem berbasis basis data yang berfungsi untuk mengelola seluruh informasi terkait proses transaksi top up secara terstruktur dan terintegrasi. Sistem ini mencakup pengelolaan data pengguna, data game, produk top up, metode pembayaran, transaksi, serta pembuatan invoice sebagai bukti pembayaran. Melalui perancangan Entity Relationship Diagram (ERD) dan penerapan relasi antar tabel, setiap proses transaksi dapat direkam secara konsisten mulai dari pemilihan produk hingga terbentuknya invoice. Sistem ini dirancang menggunakan DBMS MySQL dan diimplementasikan dengan query SQL untuk mendukung operasi pembuatan, pengolahan, dan penarikan data, sehingga mampu menghasilkan informasi transaksi dan laporan secara akurat serta sesuai dengan kebutuhan studi kasus top up game.

## **BAB II**

### **LANDASAN TEORI**

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

Basis data merupakan sekumpulan data yang saling berkaitan dan disimpan secara terstruktur agar mudah dikelola dan digunakan kembali. Dalam sebuah sistem informasi, basis data berperan penting karena menjadi tempat penyimpanan utama seluruh data yang digunakan dalam proses pengolahan informasi.

Pada sistem invoice top up game, basis data digunakan untuk menyimpan data customer, data admin, data produk top up, serta data transaksi yang terjadi. Pengelolaan basis data dilakukan menggunakan Database Management System (DBMS) MySQL. DBMS ini digunakan karena mampu mengelola data dalam jumlah besar, mendukung relasi antar tabel, serta menyediakan perintah SQL yang memudahkan proses pembuatan dan pengolahan data.

#### **2.2 Relasi Tabel**

Relasi tabel merupakan hubungan yang terjadi antara satu tabel dengan tabel lainnya dalam sebuah basis data. Relasi digunakan agar data yang disimpan tidak berdiri sendiri, melainkan saling terhubung sehingga memudahkan proses pengolahan, pencarian, dan pengambilan informasi. Dengan adanya relasi, data dapat disusun secara lebih rapi dan terhindar dari pengulangan data yang tidak diperlukan.

Dalam basis data, terdapat beberapa jenis relasi yang umum digunakan. Relasi one to one merupakan hubungan di mana satu data pada sebuah tabel hanya berhubungan dengan satu data pada tabel lain. Relasi one to many merupakan hubungan di mana satu data pada tabel pertama dapat berhubungan dengan banyak data pada tabel kedua, sedangkan data pada tabel kedua hanya berhubungan dengan satu data pada tabel pertama, dan terakhir relasi many to many yaitu hubungan di mana banyak data pada tabel pertama dapat berhubungan dengan banyak data pada tabel kedua.

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

Entity Relationship Diagram (ERD) merupakan sebuah diagram yang digunakan untuk menggambarkan struktur basis data secara konseptual. ERD menunjukkan hubungan antara entitas, atribut, serta relasi yang terjadi dalam suatu sistem basis data. Dengan adanya ERD, perancang sistem dapat memahami alur data dan keterkaitan antar tabel sebelum database diimplementasikan ke dalam DBMS.

ERD terdiri dari beberapa komponen utama, yaitu entitas yang merepresentasikan objek atau data yang akan disimpan, atribut yang menggambarkan karakteristik dari entitas, serta relasi yang menunjukkan hubungan antar entitas. Penggunaan ERD bertujuan untuk

membantu proses perancangan database agar lebih terstruktur, mudah dipahami, dan meminimalkan kesalahan dalam pembuatan tabel serta relasi pada tahap implementasi.

## 2.4 Normalisasi Data

Normalisasi merupakan proses pengelompokan data ke dalam tabel-tabel tertentu dengan tujuan untuk mengurangi duplikasi data dan menjaga konsistensi informasi dalam basis data. Proses ini dilakukan agar data tersimpan lebih terstruktur dan mudah dikelola. Normalisasi dilakukan secara bertahap mulai dari bentuk normal pertama hingga bentuk normal ketiga.

Pada bentuk normal pertama (1NF), setiap atribut harus memiliki nilai tunggal dan tidak boleh terdapat data yang berulang dalam satu kolom. Selanjutnya pada bentuk normal kedua (2NF), setiap atribut non-key harus bergantung sepenuhnya pada primary key. Pada bentuk normal ketiga (3NF), struktur tabel disusun agar tidak terdapat ketergantungan antar atribut non-key, sehingga database menjadi lebih efisien, rapi, dan mudah dikembangkan.

## 2.5 Primary Key dan Foreign Key

Primary key merupakan atribut yang digunakan sebagai identitas unik pada setiap data dalam suatu tabel sehingga tidak boleh bernilai sama dan tidak boleh kosong, sedangkan foreign key merupakan atribut yang digunakan untuk menghubungkan satu tabel dengan tabel lainnya dengan mengacu pada primary key di tabel terkait. Penggunaan primary key dan foreign key bertujuan untuk menjaga keterkaitan antar data, menghindari inkonsistensi, serta membantu membentuk relasi yang terstruktur dalam sebuah basis data.

Pada tabel customer, primary key yang digunakan adalah id\_game. Tabel produk\_topup menggunakan id\_product sebagai primary key. Tabel admin memiliki id\_admin sebagai primary key. Pada tabel transaksi, id\_transaksi digunakan sebagai primary key, sedangkan id\_game dan id\_admin digunakan sebagai foreign key untuk menghubungkan transaksi dengan customer dan admin. Tabel detail\_transaksi menggunakan primary key gabungan yang terdiri dari id\_transaksi dan id\_product, yang sekaligus berfungsi sebagai foreign key ke tabel transaksi dan produk\_topup.

## 2.6 MySQL Workbench

MySQL Workbench merupakan aplikasi yang digunakan untuk membantu proses perancangan dan pengelolaan basis data MySQL. Pada proyek ini, MySQL Workbench digunakan untuk membuat desain ERD, mengatur relasi antar tabel, serta melakukan implementasi database melalui perintah SQL. Selain itu, aplikasi ini juga digunakan untuk menjalankan query dan melihat hasil eksekusi data sehingga memudahkan proses pengujian sistem invoice top up game.

## 2.7 DDL, DML, dan TCL

Dalam pengelolaan basis data digunakan beberapa jenis perintah SQL. Data Definition Language (DDL) digunakan untuk membuat dan mengatur struktur database seperti pembuatan tabel. Data Manipulation Language (DML) digunakan untuk memasukkan dan mengelola data ke dalam tabel. Sedangkan Transaction Control Language (TCL) digunakan untuk mengatur proses transaksi agar data yang disimpan tetap konsisten dan aman.

## 2.8 Query JOIN, GROUP BY, HAVING, dan Agregasi

Query JOIN digunakan untuk menggabungkan data dari beberapa tabel sehingga menghasilkan informasi invoice yang lengkap. GROUP BY digunakan untuk mengelompokkan data transaksi berdasarkan kriteria tertentu, sedangkan HAVING digunakan sebagai kondisi pada hasil pengelompokan. Fungsi agregasi seperti COUNT digunakan untuk menghitung jumlah transaksi yang terjadi pada sistem invoice top up game.



## **BAB III**

### **PERRANCANGAN DAN IMPLEMENTASI**

#### **3.1. Analisis Kebutuhan Sistem Invoice**

Sistem invoice top up game dirancang untuk membantu proses pencatatan transaksi pembelian item atau diamond game secara terstruktur. Sistem ini dibutuhkan untuk menyimpan data customer, data produk top up, data transaksi, serta detail pembelian yang dilakukan oleh customer. Selain itu, sistem juga menyediakan pencatatan admin yang berperan dalam mengelola transaksi.

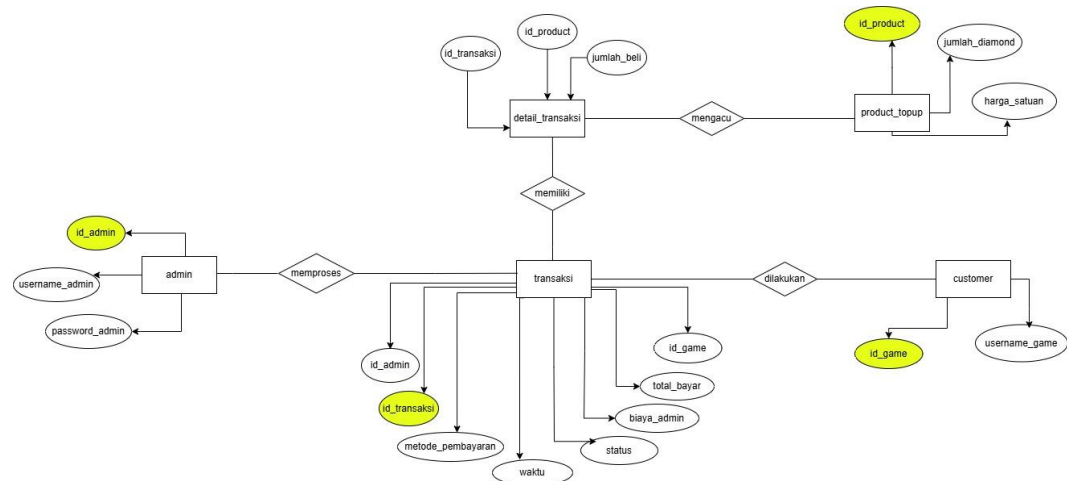
Kebutuhan utama dari sistem ini adalah mampu mencatat setiap transaksi top up, menampilkan rincian pembelian produk, serta menghasilkan informasi invoice yang jelas dan mudah dipahami. Dengan adanya sistem berbasis basis data, proses pencatatan transaksi dapat dilakukan dengan lebih rapi, mengurangi kesalahan pencatatan, dan memudahkan proses pembuatan laporan.

#### **3.2. Perancangan Basis Data**

Perancangan basis data dilakukan untuk menentukan struktur tabel dan hubungan antar data yang digunakan dalam sistem. Basis data dirancang agar mampu menyimpan data transaksi secara rapi, menghindari duplikasi data, serta memudahkan proses pengolahan informasi.

Basis data sistem invoice top up game dirancang menggunakan beberapa tabel utama, yaitu tabel user (customer), produk, transaksi, detail transaksi, admin, dan metode pembayaran. Setiap tabel memiliki fungsi yang berbeda dan saling terhubung melalui relasi primary key dan foreign key.

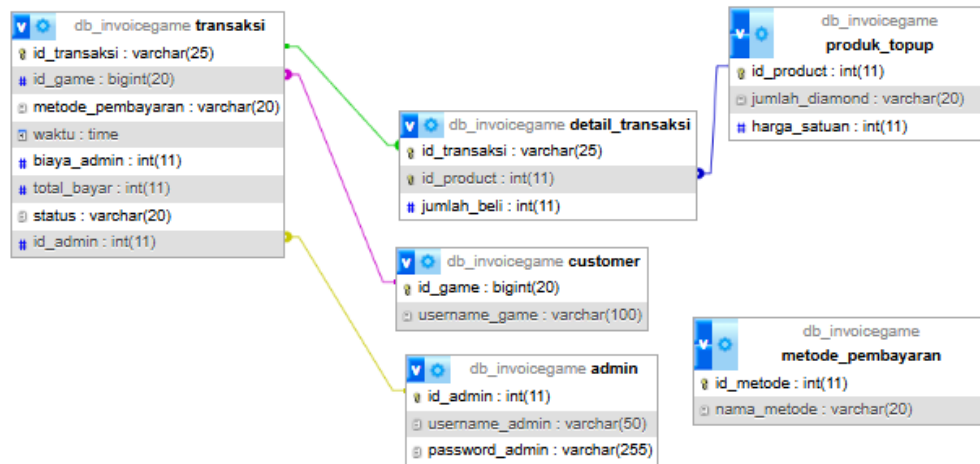
### 3.2.1. ERD Sistem Top Up Game



Gambar 1. ERD

Entity Relationship Diagram (ERD) pada sistem invoice top up game menggambarkan hubungan antara entitas Customer, Product, Transaksi, Laporan\_Transaksi, dan Admin, di mana Customer dengan atribut utama id\_game dan username melakukan transaksi pembelian Product yang memiliki atribut id\_product, jumlah\_diamond, harga\_satuan, dan jumlah\_beli, kemudian proses tersebut dicatat dalam entitas Transaksi dengan id\_transaksi sebagai primary key serta atribut total\_harga, biaya\_admin, metode\_bayar, waktu, dan status; setiap transaksi menghasilkan data pada Laporan\_Transaksi yang memiliki id\_laporan dan dikelola oleh Admin dengan atribut id\_admin, username\_admin, dan password\_admin, dengan relasi yang menunjukkan bahwa satu customer dapat melakukan banyak transaksi, satu produk dapat terlibat dalam banyak transaksi, satu transaksi menghasilkan satu laporan, dan satu admin dapat mengelola banyak laporan transaksi.

### 3.2.2. Desain Tabel



Gambar 2. Relasi antar tabel

Desain tabel pada sistem invoice top up game disusun berdasarkan hasil perancangan ERD dan kebutuhan sistem. Tabel customer digunakan untuk menyimpan data pemain game yang melakukan transaksi top up, dengan atribut utama berupa id\_game dan username\_game. Tabel admin berfungsi untuk menyimpan data pengelola sistem yang terdiri dari id\_admin, username\_admin, dan password\_admin. Tabel produk\_topup digunakan untuk menyimpan informasi paket top up yang tersedia, seperti jumlah diamond dan harga satuan.

Tabel transaksi digunakan untuk mencatat seluruh proses pembelian top up yang dilakukan oleh customer. Pada tabel ini disimpan informasi penting seperti id\_transaksi, metode\_pembayaran, waktu transaksi, biaya admin, total bayar, dan status transaksi. Selain itu, tabel transaksi juga memiliki keterkaitan dengan tabel customer dan admin. Untuk menyimpan rincian produk yang dibeli, digunakan tabel detail\_transaksi yang berfungsi sebagai penghubung antara transaksi dan produk\_topup. Selain itu, sistem juga menggunakan tabel metode\_pembayaran sebagai tabel pendukung yang menyimpan daftar metode pembayaran yang dapat digunakan dalam proses transaksi.

### 3.2.3. Relasi Primary Key dan Foreign Key

Relasi antar tabel pada sistem invoice top up game dibuat agar setiap data saling terhubung dan tidak berdiri sendiri. Setiap tabel memiliki primary key yang digunakan sebagai penanda data utama. Pada tabel customer digunakan id\_game sebagai identitas pemain, sedangkan pada tabel admin digunakan id\_admin untuk membedakan setiap akun admin. Tabel produk\_topup menggunakan id\_product sebagai identitas setiap paket top up yang tersedia.

Pada tabel transaksi digunakan id\_transaksi sebagai primary key. Selain itu, tabel ini juga memiliki id\_game dan id\_admin sebagai foreign key yang berfungsi untuk menunjukkan bahwa suatu transaksi dilakukan oleh customer tertentu dan diproses oleh admin tertentu. Untuk menyimpan rincian produk yang dibeli dalam satu transaksi, digunakan tabel detail\_transaksi yang memiliki primary key gabungan antara id\_transaksi dan id\_product. Kedua atribut tersebut juga berfungsi sebagai penghubung antara tabel transaksi dan tabel produk\_topup. Dengan adanya relasi ini, data transaksi dapat tersimpan dengan lebih rapi dan mudah ditelusuri kembali.

## 3.3. Implementasi Database

### 3.3.1. DDL

```
-- DDL
CREATE TABLE metode_pembayaran (
    id_metode INT PRIMARY KEY,
    nama_metode VARCHAR(20) NOT NULL
);
```

*Gambar 3. Code DDL*

DDL digunakan untuk mendefinisikan struktur basis data. Pada sistem invoice top up game, DDL digunakan untuk membuat tabel metode\_pembayaran yang berfungsi menyimpan daftar metode pembayaran yang tersedia. Tabel ini memiliki atribut id\_metode sebagai primary key dan nama\_metode sebagai nama metode pembayaran. Dengan penggunaan DDL, struktur tabel dapat dibentuk secara jelas sebelum dilakukan pengisian data.

### 3.3.2. DML

```
-- DML

INSERT INTO metode_pembayaran VALUES
(1, 'GOPAY'),
(2, 'DANA'),
(3, 'QRIS');
```

*Gambar 4. Code DML*

DML digunakan untuk mengelola dan memasukkan data ke dalam tabel yang telah dibuat. Pada sistem ini, DML digunakan untuk mengisi tabel metode\_pembayaran dengan beberapa metode pembayaran, seperti GOPAY, DANA, dan QRIS. Data ini nantinya digunakan sebagai referensi dalam proses transaksi top up game.

### 3.3.3. TCL

```
-- TCL

START TRANSACTION;
INSERT INTO transaksi
VALUES ('TRX999', 217978152, 'GOPAY', '20:30', 2000, 50000, 'SELESAI');
COMMIT;
```

*Gambar 5. Code TCL*

TCL digunakan untuk mengatur proses transaksi agar data yang disimpan tetap konsisten. Pada sistem ini, perintah START TRANSACTION digunakan untuk memulai proses penyimpanan transaksi, kemudian dilakukan perintah INSERT untuk menambahkan data transaksi baru, dan diakhiri dengan perintah COMMIT untuk menyimpan perubahan secara permanen. Penggunaan TCL memastikan bahwa data transaksi hanya akan tersimpan apabila seluruh proses berjalan dengan baik.

### 3.4.Implementasi Query

Pada tahap implementasi query, dilakukan pengujian terhadap basis data untuk memastikan bahwa data yang tersimpan dapat diolah sesuai dengan kebutuhan sistem invoice top up game. Query yang digunakan meliputi JOIN, GROUP BY, HAVING, serta fungsi agregasi.

#### 3.4.1. Query Agregasi dan HAVING

Query agregasi digunakan untuk melakukan perhitungan data transaksi berdasarkan metode pembayaran. Pada sistem invoice top up game, fungsi agregasi COUNT digunakan untuk mengetahui jumlah transaksi yang terjadi pada setiap metode pembayaran. Selain itu, klausa HAVING digunakan untuk memberikan kondisi pada hasil pengelompokan data.

```
-- AGRESI DAN HAVING

SELECT metode_pembayaran, COUNT(*) AS total_transaksi
FROM transaksi
GROUP BY metode_pembayaran
HAVING COUNT(*) >= 2;
```

Gambar 6. Code Query Agregasi dan Having

Hasil dari query di atas menampilkan metode pembayaran yang digunakan minimal dua kali transaksi. Dari hasil tersebut dapat diketahui metode pembayaran yang paling sering digunakan oleh customer dalam melakukan top up game.



	metode_pembayaran	total_transaksi
▶	GOPAY	3

Gambar 7. Output Code Query Agregasi dan Having

### 3.4.2. Query GROUP BY

Query GROUP BY digunakan untuk mengelompokkan data transaksi berdasarkan metode pembayaran. Dengan adanya pengelompokan ini, data transaksi dapat dianalisis berdasarkan kategori tertentu sehingga memudahkan proses pembuatan laporan.

```
-- GROUP BY
SELECT metode_pembayaran, COUNT(*) AS total_transaksi
FROM transaksi
GROUP BY metode_pembayaran
HAVING COUNT(*) >= 2;
```

Gambar 8. Code Query Group By

Query menghasilkan jumlah transaksi pada masing-masing metode pembayaran yang memenuhi kondisi. Informasi ini dapat digunakan untuk mengetahui metode pembayaran yang paling sering digunakan dalam sistem invoice top up game.



	metode_pembayaran	total_transaksi
▶	GOPAY	3

Gambar 9. Output Query Group By

### 3.4.3. Query JOIN

Query JOIN digunakan untuk menggabungkan beberapa tabel agar menghasilkan informasi invoice secara lengkap. Pada sistem ini, JOIN dilakukan terhadap tabel transaksi, customer, detail\_transaksi, dan produk\_topup.

```
-- JOIN

SELECT
    c.username_game,
    t.id_transaksi,
    p.jumlah_diamond,
    d.jumlah_beli,
    t.total_bayar
FROM transaksi t
JOIN customer c ON t.id_game = c.id_game
JOIN detail_transaksi d ON t.id_transaksi = d.id_transaksi
JOIN produk_topup p ON d.id_product = p.id_product;
```

Gambar 10. Code Query Join

Hasil query menampilkan data invoice berupa nama pemain, kode transaksi, jenis paket diamond, jumlah pembelian, serta total pembayaran. Dengan penggunaan JOIN, data dari beberapa tabel dapat ditampilkan menjadi satu informasi yang utuh sehingga memudahkan proses pengecekan dan pelaporan transaksi.

username_game	id_transaksi	jumlah_diamond	jumlah_beli	total_bayar
PlayerML03	AGSTF288IGSSYTFZ	500+5 Diamond	1	85000
PlayerML01	ASJDBNJAVHK128WF	250 Diamond	1	29500
PlayerML06	ASWY73YSGGOSU11	50+6 Diamond	4	62000
PlayerML04	DVASUHFWDVU9Z1E	300 Diamond	2	72000
PlayerML02	LSACBQLHFU3Q74B	2000 Diamond	3	252000
PlayerML05	WHDBSJADBAVJDSA	400+3 Diamond	3	197000

Gambar 11. Output Code Query Join



## **BAB IV**

### **PENUTUP**

#### **4.1. Hasil Pengujian Sistem**

Pengujian sistem dilakukan untuk memastikan bahwa basis data invoice top up game dapat berjalan sesuai dengan perancangan yang telah dibuat. Pengujian dilakukan menggunakan MySQL Workbench dengan menjalankan perintah SQL pada setiap tabel yang telah dibuat. Hasil pengujian menunjukkan bahwa proses pembuatan tabel, pengisian data, serta relasi antar tabel dapat berjalan dengan baik tanpa terjadi error.

Selain itu, pengujian query seperti JOIN, GROUP BY, HAVING, dan fungsi agregasi berhasil menampilkan data sesuai dengan yang diharapkan. Data transaksi dapat ditampilkan secara lengkap, mulai dari informasi customer, produk yang dibeli, hingga total pembayaran. Berdasarkan hasil tersebut, dapat disimpulkan bahwa sistem basis data invoice top up game telah berfungsi dengan baik dan siap digunakan sebagai dasar pengelolaan data transaksi.

#### **4.2. Kendala dan Solusi**

Dalam proses perancangan dan implementasi sistem basis data invoice top up game, terdapat beberapa kendala yang dihadapi. Salah satu kendala yang muncul adalah penentuan relasi antar tabel, terutama pada saat menghubungkan tabel transaksi dengan produk top up sehingga diperlukan tabel penghubung agar data dapat tersimpan dengan benar. Selain itu, pada tahap awal juga ditemukan kesalahan dalam penulisan query yang menyebabkan data tidak dapat ditampilkan sesuai dengan yang diharapkan.

Untuk mengatasi kendala tersebut, dilakukan penyesuaian pada desain database dengan memperbaiki relasi primary key dan foreign key serta melakukan pengujian ulang menggunakan MySQL Workbench. Kesalahan query diperbaiki dengan menyesuaikan struktur tabel dan penggunaan perintah SQL yang tepat. Setelah dilakukan perbaikan, sistem dapat berjalan dengan baik dan data dapat ditampilkan secara lengkap.

#### 4.3.Kesimpulan

Berdasarkan hasil perancangan dan implementasi yang telah dilakukan, sistem basis data invoice top up game berhasil dibangun dengan baik menggunakan MySQL. Sistem ini mampu mengelola data customer, produk top up, transaksi, serta detail transaksi secara terstruktur dan saling terhubung melalui relasi antar tabel.

Penerapan konsep basis data seperti ERD, normalisasi, primary key, foreign key, serta penggunaan query JOIN, GROUP BY, HAVING, dan agregasi dapat membantu menghasilkan informasi transaksi secara lengkap dan mudah dipahami. Dengan demikian, sistem basis data yang dibuat dapat digunakan sebagai dasar pengelolaan transaksi top up game secara lebih rapi dan efisien.

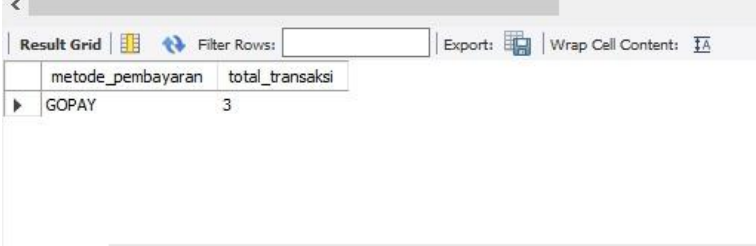
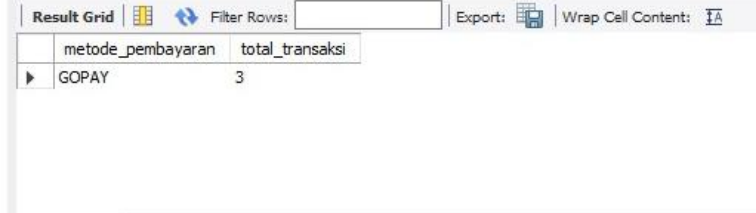

#### 4.4.Saran Pengembangan

Sistem basis data invoice top up game yang telah dibuat masih dapat dikembangkan lebih lanjut agar menjadi lebih baik. Pengembangan yang dapat dilakukan antara lain dengan menambahkan antarmuka berbasis web atau aplikasi sehingga sistem dapat digunakan secara langsung oleh pengguna. Selain itu, sistem juga dapat dikembangkan dengan menambahkan fitur keamanan data, pengelolaan hak akses pengguna, serta integrasi dengan payment gateway agar proses transaksi menjadi lebih otomatis dan aman.

Dengan adanya pengembangan tersebut, diharapkan sistem invoice top up game dapat digunakan secara lebih optimal dan sesuai dengan kebutuhan layanan top up game yang sebenarnya.

## LAMPIRAN

No	Foto	Deskripsi
1.		Gambar tabel ERD sisitem Invoice TopUp Game.
2.		Gambar relasi antar tabel dan relasi PK dan FK.
3.	<pre>-- DDL CREATE TABLE metode_pembayaran (     id_metode INT PRIMARY KEY,     nama_metode VARCHAR(20) NOT NULL );</pre>	Gambar Code DDL
4.	<pre>-- DML INSERT INTO metode_pembayaran VALUES (1, 'GOPAY'), (2, 'DANA'), (3, 'QRIS');</pre>	Gambar Code DML
5.	<pre>-- TCL START TRANSACTION; INSERT INTO transaksi VALUES ('TRX999', 217978152, 'GOPAY', '20:30', 2000, 50000, 'SELESAT'); COMMIT;</pre>	Gambar Code TCL

6.	<pre>-- AGRESI DAN HAVING  SELECT metode_pembayaran, COUNT(*) AS total_transaksi FROM transaksi GROUP BY metode_pembayaran HAVING COUNT(*) &gt;= 2;</pre>	Gambar Code Query AGREGASI DAN HAVING
7.		Gambar Output Code Query Agregasi dan Having
8.	<pre>-- GROUP BY  SELECT metode_pembayaran, COUNT(*) AS total_transaksi FROM transaksi GROUP BY metode_pembayaran HAVING COUNT(*) &gt;= 2;</pre>	Gambar code Query GROUP BY.
9.		Gambar output code Query GROUP BY.
10.	<pre>-- JOIN  SELECT     c.username_game,     t.id_transaksi,     p.jumlah_diamond,     d.jumlah_beli,     t.total_bayar FROM transaksi t JOIN customer c ON t.id_game = c.id_game JOIN detail_transaksi d ON t.id_transaksi = d.id_transaksi JOIN produk_topup p ON d.id_product = p.id_product;</pre>	Gambar code Query JOIN
11.		Output code Query JOIN

