

# **Pengembangan Permainan Poker Multipemain dengan Basis Data**



**DISUSUN OLEH:**

**Kelompok 2**

Rakesh Bramantyo (105221009)

Ichiro Albert(105221022)

Reva Ananda (105221025)

Baiquni Krisna(105221037)

**Program Studi:**

Ilmu Komputer

**Mata Kuliah:**

Pemrograman Berbasis Objek

## Daftar Isi

<b>Daftar Isi.....</b>	<b>2</b>
<b>Bab I.....</b>	<b>3</b>
<b>Pendahuluan.....</b>	<b>3</b>
1.1 Latar Belakang.....	3
1.2 Rumusan Masalah.....	3
1.3 Tujuan.....	3
1.4 Manfaat.....	3
1.5 Permainan Poker.....	4
<b>Bab II.....</b>	<b>5</b>
<b>Metodologi.....</b>	<b>5</b>
2.1 Desain Sistem.....	5
2.1.1 Flowchart.....	5
i. Flowchart Gameplay.....	5
ii. Flowchart Penentuan Pemenang.....	6
2.1.2 Class Diagram.....	6
2.1.3 Konsep PBO.....	8
i. Enkapsulasi.....	8
i.a Variabel private.....	8
i.b Metode Getter dan Setter.....	8
ii. Inheritance.....	9
ii.a Kelas Parent (Induk).....	9
ii.b Kelas child (Anak).....	9
iii. Interface.....	9
iii.a Method interface.....	9
iii.b Implementasi Interface.....	9
iv. Polymorphism.....	10
2.2 Implementasi.....	10
2.2.1 Graphical User Interface.....	10
2.2.2 Integrasi Database.....	10
i. MySql.....	10
i.a. Tabel serverdb.....	11
i.b. Tabel userdb.....	11
<b>Bab III.....</b>	<b>12</b>
<b>Hasil dan Pembahasan.....</b>	<b>12</b>
3.1 Program Java.....	12
3.2 Output Program.....	12
3.2.1 Hasil dan Pembahasan Output Program.....	12
i. Output Frame Loading dan Menu Utama.....	12
ii. Output Frame Tutorial/ Frame Info.....	13
iii. Output Fase Autentikasi Akun.....	13
iv. Output Frame Gameplay Default dan Frame Profile.....	14
v. Output Membuat Room (Host) dan mencari Room (Client).....	14
vi. Output Fase Gameplay.....	15
vii. Output Fase Penentuan Pemenang.....	16

vii.a. Menang.....	16
vii.b. Kalah.....	17
<b>Bab IV.....</b>	<b>17</b>
<b>Penutup.....</b>	<b>17</b>
4.1 Kesimpulan.....	17

## **Bab I**

### **Pendahuluan**

#### **1.1 Latar Belakang**

Permainan poker memiliki sejarah yang panjang sejak tahun 1892 dan berasal dari Prancis. Seiring berjalananya waktu, popularitas permainan ini terus meningkat di seluruh dunia. Sebagai mahasiswa Ilmu Komputer di Universitas Pertamina, kami tertarik untuk menjadikan permainan poker sebagai topik tugas besar dalam mata kuliah Pemrograman Berbasis Objek. Dalam pengembangan permainan ini, kami memilih Bahasa Pemrograman Java untuk mengorganisir dan mengelola komponen permainan secara efisien. Selain itu, kami menggunakan database MySQL untuk memfasilitasi koneksi antar pemain dan menyimpan data pemain secara terstruktur.

#### **1.2 Rumusan Masalah**

Berikut adalah rumusan masalah yang ingin kami selesaikan sebagai pengembang dalam mengembangkan permainan poker multiplayer ini, masalah-masalah tersebut antara lain:

1. Bagaimana cara merancang program permainan yang sesuai dengan kaidah permainan poker ?
2. Bagaimana mengintegrasikan permainan poker dengan database MySQL untuk menyimpan data dan menghubungkan para pemain?
3. Apa saja fitur-fitur yang harus ada dalam permainan poker multipemain yang menggunakan pemrograman berbasis objek dan database?

#### **1.3 Tujuan**

Tujuan kami untuk membuat aplikasi ini, adalah untuk menerapkan prinsip 4 pemrograman berorientasi objek (PBO), beserta MySQL dan menganalisis berbagai aspek yang terkait dengan desain, pengembangan, dan pengalaman pengguna. kami akan menunjukkan bagaimana PBO meningkatkan proses pengembangan, pengorganisasian kode, modularitas, dan pengalaman pengguna dalam konteks aplikasi permainan poker.

#### **1.4 Manfaat**

Manfaat yang didapat dari pengembangan permainan poker tersebut antara lain:

Manfaat yang didapat dari pengembangan permainan poker tersebut antara lain:

1. Hiburan dan Rekreasi: Menyediakan pengalaman bermain yang seru dan menyenangkan bagi pemain dalam interaksi dengan teman-teman atau keluarga.
2. Pengembangan Keterampilan: Mengasah keterampilan analitis, strategis, pengambilan keputusan, dan pembacaan situasi, serta memperoleh pengalaman praktis dalam mengelola risiko, strategi, dan peluang.
3. Pengelolaan Emosi: Membantu pemain mengendalikan emosi, mengambil keputusan yang lebih baik, dan menghadapi tantangan dengan kepala dingin, berdampak positif pada pengelolaan emosi sehari-hari.
4. Peningkatan Keberanian: Membantu pemain mengembangkan keberanian dan kemampuan mengambil risiko yang terkontrol, melalui pertimbangan peluang, analisis situasi, dan pengambilan keputusan berdasarkan keyakinan.

### 1.5 Permainan Poker



**Gambar 1.1**

Poker adalah permainan kartu yang dimainkan antara beberapa pemain di mana setiap pemain bertaruh dengan nilai tertentu berdasarkan kombinasi kartu yang mereka pegang. Tujuan utama dalam poker adalah memenangkan taruhan dengan mengumpulkan kombinasi kartu terbaik di antara semua pemain atau membuat pemain lain menyerah. Dalam permainan poker, terdapat beberapa tindakan atau aksi yang dapat dilakukan oleh pemain selama tahap taruhan. Berikut adalah penjelasan singkat mengenai tiga aksi utama dalam poker: Check, Raise, dan Fold.

Poker melibatkan unsur keberuntungan, keterampilan, dan strategi. Selain itu, terdapat berbagai variasi poker yang berbeda, seperti Texas Hold'em, Omaha, Seven-Card Stud, dan lainnya, yang memiliki aturan dan format permainan yang sedikit berbeda. Pada tugas besar kali ini, kelompok kami sebagai pengembang akan membuat permainan Poker dengan variasi Texas Hold'em dan melakukan beberapa penyesuaian.

## Bab II

# Metodologi

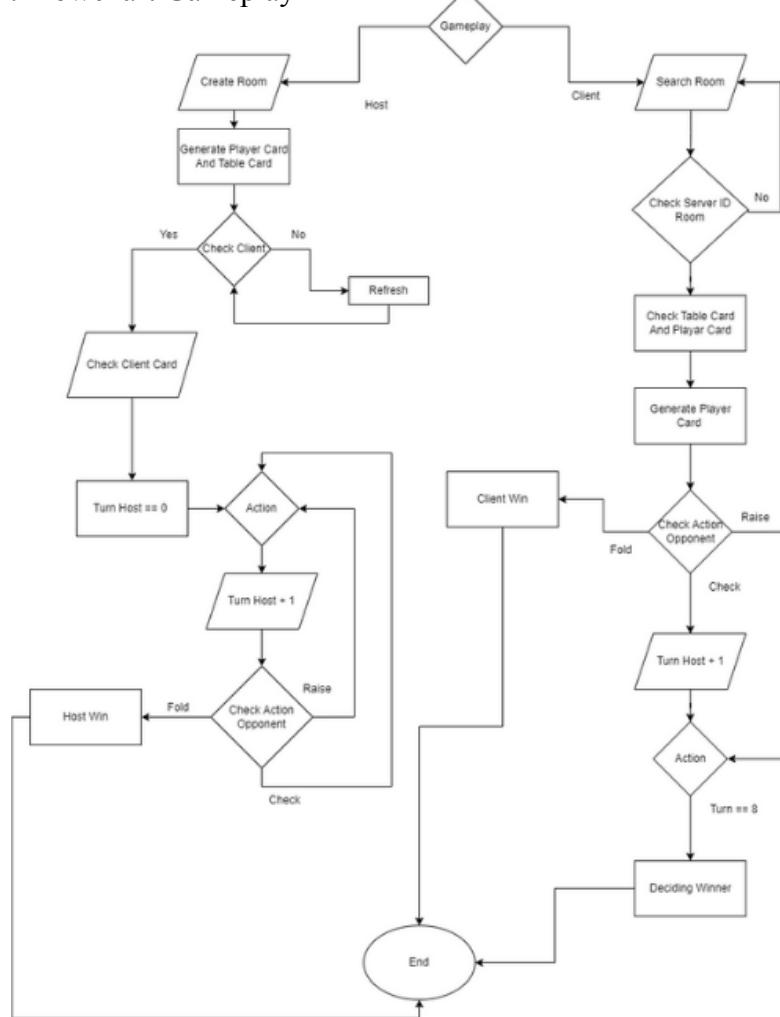
### 2.1 Desain Sistem

Dalam membuat desain pada sistem permainan poker kami terdapat beberapa alur sistem permainan poker dan hubungan antara kelas-kelas yang ada pada sistem kami:

#### 2.1.1 Flowchart

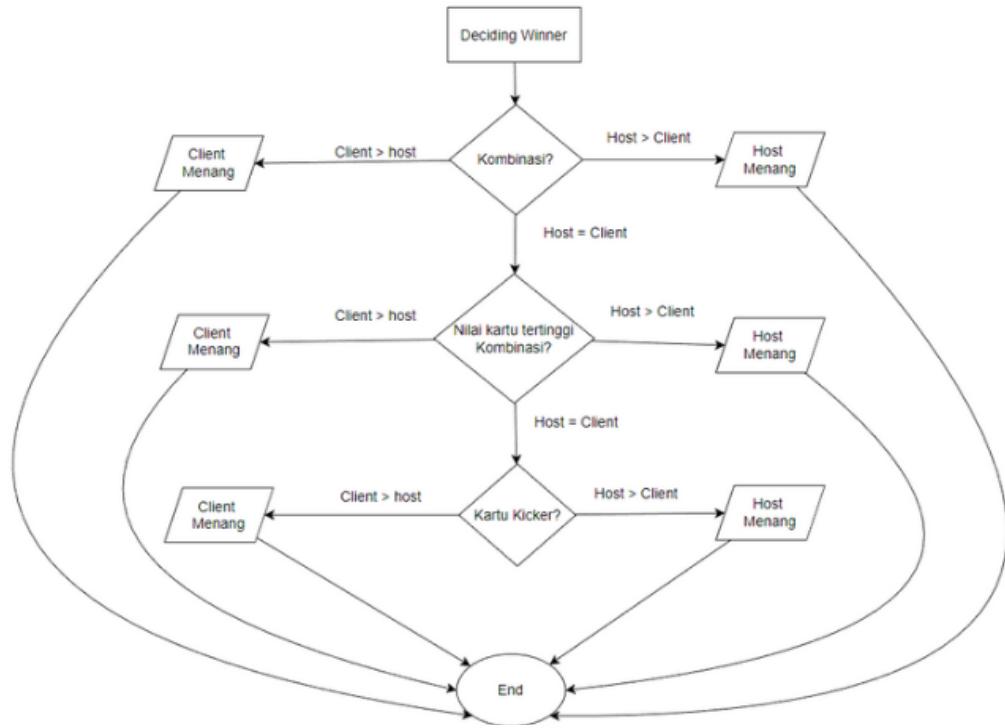
Kami melampirkan *Flowchart* dari program yang sedang kami kembangkan. *Flowchart* ini dibagi menjadi dua bagian, yaitu *flowchart* untuk proses permainan dan *flowchart* untuk penentuan pemenang. Pembagian ini dilakukan untuk meningkatkan keterbacaan dan memudahkan pemahaman.

##### i. Flowchart Gameplay



Gambar 2.1

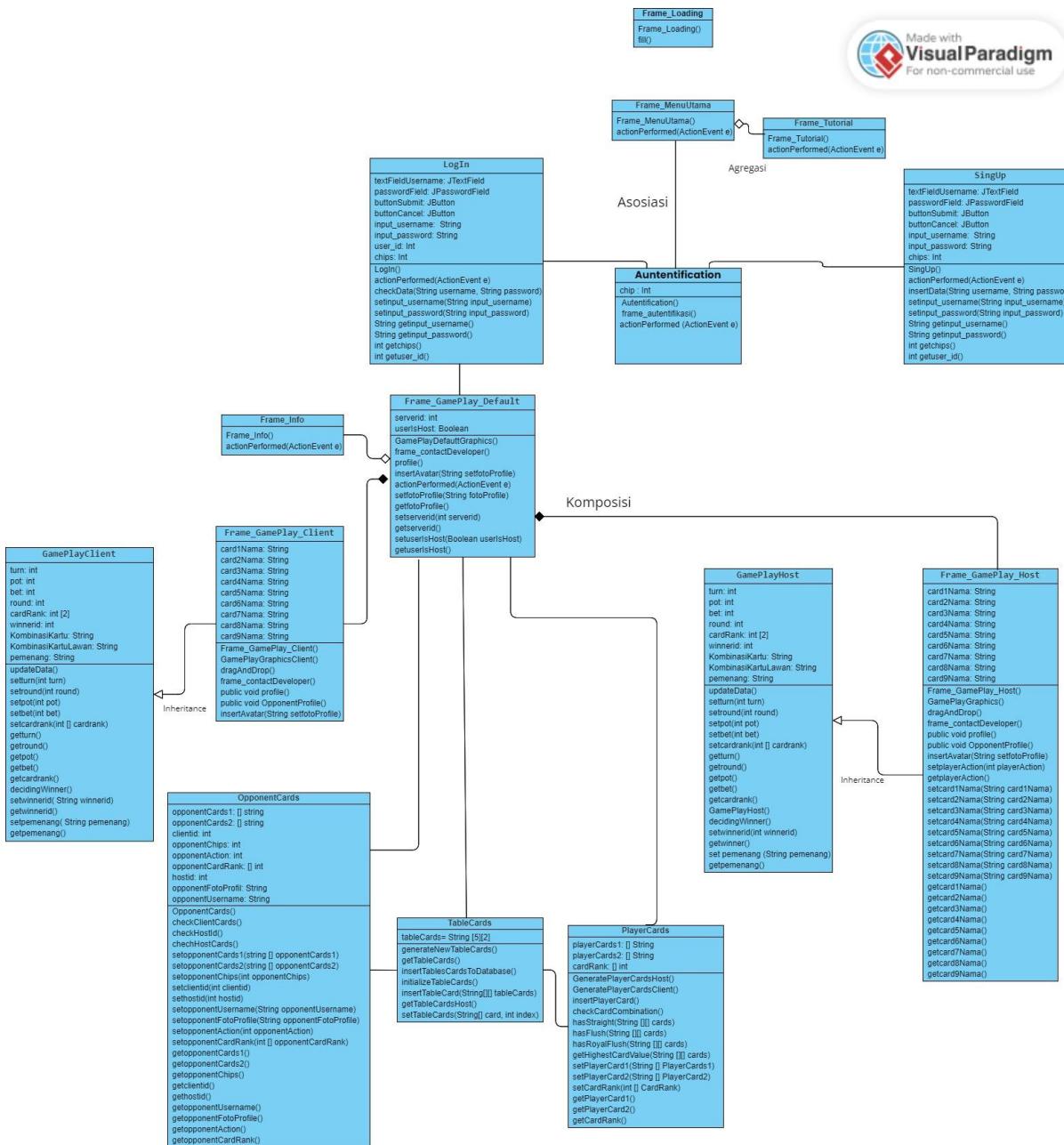
## ii. Flowchart Penentuan Pemenang



**Gambar 2.2**

### 2.1.2 Class Diagram

Kami sangat menghargai pendekatan yang diterapkan dalam pembuatan kelas diagram untuk program permainan poker ini. Dalam mengatur kelas diagram, kami sebagai pengembang memperhatikan kaidah dan konsep yang telah dipelajari dalam mata kuliah ini, serta menjaga hubungan yang konsisten antara kelas dan objek-objek dalam program permainan poker. Hal ini akan memastikan struktur yang baik dan kemudahan pemahaman bagi pengembang dan pembaca diagram.



Gambar 2.3

Pada *class diagram* di atas nampak beberapa hubungan antara kelas, seperti hubungan asosiasi, agregasi, komposisi, dan pewarisan (*Inheritance*). Hubungan Asosiasi merupakan hubungan antara dua objek yang bersifat statis. Hubungan Agregasi merupakan hubungan satu objek yang memiliki keterkaitan atau bagian dari objek lain yang lebih besar, namun masing-masing kelas tersebut berdiri secara mandiri. Hubungan Komposisi merupakan hubungan satu objek yang memiliki keterkaitan atau bagian dari objek yang lebih besar yang menyebabkan kedua hubungan tersebut menjadi suatu kesatuan yang tak terpisahkan. Hubungan Pewarisan (*Inheritance*) merupakan hubungan antara kelas tua dengan kelas turunannya yang dimana kelas turunan akan menggunakan seluruh sifat atau *method* dari kelas orang tua.

### 2.1.3 Konsep PBO

Pada program permainan poker yang kami kembangkan, kami sebagai pengembang menerapkan beberapa konsep dasar pemrograman berbasis objek yang digunakan baik untuk mempermudah proses pengkodean, meningkatkan efisiensi program maupun menambah *readability* dan *reusability* program. Terdapat 4 konsep pemrograman berorientasi objek, yaitu:

#### i. Enkapsulasi

Enkapsulasi merupakan cara mengisolasi sebuah data agar tidak dapat diakses secara langsung oleh suatu kelas atau objek lain. Enkapsulasi pada program java bertujuan untuk keamanan data, kontrol akses yang baik, kode yang lebih mudah dipahami, dan kemampuan pemeliharaan yang lebih baik.

##### i.a Variabel private

```
private static String input_username;
private static String input_password;
private static int user_id;
private static int chips;
```

Gambar 2.4

##### i.b Metode Getter dan Setter

```
public void setinput_username(String input_username) {
    LogIn.input_username = input_username;
}
public void setinput_password(String input_password) {
    LogIn.input_password = input_password;
}

public static void setchips(int chips) {
    LogIn.chips = chips;
}
public void setuser_id(int user_id) {
    LogIn.user_id = user_id;
}

public static String getinput_username() {
    return input_username;
}

public static String getinput_password() {
    return input_password;
}

public static int getchips() {
    return chips;
}
public static int getuser_id() {
    return user_id;
}
```

**Gambar 2.5**

ii. Inheritance

*Inheritance* (Pewarisan) adalah konsep dalam pemrograman berorientasi objek (PBO) yang memungkinkan suatu kelas baru untuk mewarisi properti dan metode dari kelas yang sudah ada. Kelas yang menerima warisan properti dari kelas yang sudah ada disebut sebagai kelas turunan, sedangkan kelas yang mewarisi properti mewarisi disebut sebagai kelas orang tua (*Parent Class*).

ii.a Kelas Parent (Induk)

```
public class GamePlayHost {
    private static int turn; // pembukaan kartu
    private static int pot; // jumlah taruhan t
    private static int bet; // jumlah taruhan y
    private static int round; // giliran pemain
    private static int[] cardRank = new int[3];
    |
    |
    |
    |
    private static int winnerid;
    public static String KombinasiKartu;
    public static String KombinasiKartuLawan;
    public static String pemenang;

    public static void checkOpponentAction() {
```

**Gambar 2.6**

ii.b Kelas child (Anak)

```
public class Frame_GamePlay_Host extends GamePlayHost implements ActionListener {
```

**Gambar 2.7**

iii. Interface

*Interface* adalah kerangka yang menggambarkan tindakan atau perilaku yang diharapkan dari suatu objek. *Interface* berfungsi sebagai kerangka yang menyediakan metode yang harus diimplementasikan oleh kelas yang menggunakan kerangka *Interface* tersebut.

iii.a Method interface

```
@Override
public void actionPerformed(ActionEvent e) {

    if (e.getSource() == buttonStartGame) {

        int optionSelected = JOptionPane.showOptionDialog(parentComponent:null, message:"SE
        JOptionPane.INFORMATION_MESSAGE, icon:null, new Object[] { "Create Room", "Join
        if (optionSelected == 0) {
```

**Gambar 2.8**

### iii.b Implementasi Interface

```
public class Frame_GamePlay_Default implements ActionListener {
```

Gambar 2.9

### iv. Polymorphism

*Polymorphism* adalah konsep dalam pemrograman berorientasi objek (PBO) yang memungkinkan suatu *class* memiliki banyak “bentuk” *method* yang berbeda, meskipun namanya sama. Maksud dari “bentuk” adalah isinya yang berbeda, namun tipe data dan parameternya berbeda.

```
GamePlayHost gamePlay_Host = new Frame_GamePlay_Host(); // upcasting

if (gamePlay_Host instanceof Frame_GamePlay_Host) { // downcasting
    System.out.println("GamePlayHost");
    Frame_GamePlay_Host frameGamePlayHost = (Frame_GamePlay_Host) gamePlay_Host;

    frameGamePlayHost.GamePlayGraphics();
} else {
    System.out.println("Bukan GamePlayHost");
}

System.out.println("Giliran Jalan");
}
```

Gambar 2.10

## 2.2 Implementasi

### 2.2.1 Graphical User Interface

GUI atau ‘*Graphical User Interface*’ atau Diimplementasikan pada program dengan menggunakan beberapa *library* kelas yang tersedia pada bahasa pemrograman Java seperti ‘*swing*’ dan ‘*awt*’ untuk membuat objek dan komponen-komponen yang diperlukan seperti *JFrame*, *JPanel* *JLabel*, *JButton* dll. Komponen-komponen tersebut kemudian diatur ukuran, posisi, tampilan, warna, background, sifat dan lain sebagainya menggunakan sintaksis dan fungsi yang tersedia.

Pada program ini juga digunakan asset-asset lain seperti gambar-gambar *png* dan *jpg* untuk tampilan latar belakang, kartu-kartu, tampilan dll juga musik latar belakang bertipe ‘*wav*’ yang didapatkan secara gratis melalui internet.

### 2.2.2 Integrasi Database

#### i. MySql

Pada program permainan poker multipemain ini diperlukan *database* bernama ‘*pokerdb*’ yang digunakan untuk menampung tabel-tabel yang dibutuhkah, untuk menampung data para pemain dan data permainan. Berikut adalah gambar database dan tabel-tabel nya:

```
MariaDB [pokerdb]> SHOW TABLES;
+-----+
| Tables_in_pokerdb |
+-----+
| serverdb          |
| userdb            |
+-----+
```

### Gambar 2.11

Pada gambar tersebut terlihat ada 2 tabel, tabel-tabel tersebut adalah ‘serverdb’ yang digunakan untuk menampung data permainan atau ‘server’ yang digunakan para pemain untuk membuat, mengundang (*invite*), dan bergabung (*join*) sebuah ‘room’ yang berfungsi sebagai wadah atau tempat para pemain dalam proses permainan poker. Berikut adalah informasi lengkap mengenai komponen apa saja yang ada dalam tabel-tabel tersebut beserta tipe data dan kunci yang dimiliki:

i.a. Tabel serverdb

Field	Type	Null	Key	Default	Extra
tbCard1_Value	int(11)	YES		NULL	
tbCard1_Symbol	varchar(255)	YES		NULL	
tbCard2_Value	int(11)	YES		NULL	
tbCard2_Symbol	varchar(225)	YES		NULL	
tbCard3_Value	int(11)	YES		NULL	
tbCard3_Symbol	varchar(255)	YES		NULL	
tbCard4_Value	int(11)	YES		NULL	
tbCard4_Symbol	varchar(255)	YES		NULL	
tbCard5_Value	int(11)	YES		NULL	
tbCard5_Symbol	varchar(255)	YES		NULL	
hostid	int(11)	YES	MUL	NULL	
clientid	int(11)	YES		NULL	
serverid	int(11)	YES		NULL	
pot	int(11)	YES		NULL	
turn	int(11)	YES		NULL	
bet	int(11)	YES		NULL	
round	int(11)	YES		NULL	
winnerid	int(11)	YES		NULL	

Gambar 2.12

Bisa dilihat dari gambar diatas terdapat ‘Field’ atau komponen data pada tabel, field tableCard\_Value1 sampai 5 digunakan untuk menyimpan nilai dari kelima kartu meja, sedangkan field tbCard\_Symbol1 sampai 5 digunakan untuk menyimpan simbol atau ‘suit’ dari kelima kartu meja. lalu ada ‘hostid’, ‘clientid’, ‘serverid’ untuk menyimpan informasi ‘ID’ dari pemilik room (host), tamu ‘client’ dan server itu sendiri.

Selanjutnya ada field ‘pot’ dan ‘bet’ untuk menyimpan data jumlah total taruhan chips (pot) dan jumlah chips yang sedang dipertaruhkan (bet) dalam permainan. Field ‘turn’ dan ‘round’ digunakan sebagai iterator dan penentu babak permainan, lalu ada field ‘winnerid’ untuk menyimpan data ID dari pemenang permainan tersebut.

i.b. Tabel userdb

MariaDB [pokerdb]> describe userdb;					
Field	Type	Null	Key	Default	Extra
userid	int(11)	NO	PRI	NULL	auto_increment
username	varchar(50)	YES		NULL	
password	varchar(50)	YES		NULL	
chips	int(11)	YES		NULL	
card1_Value	int(11)	YES		NULL	
card1_Symbol	varchar(225)	YES		NULL	
card2_Value	int(11)	YES		NULL	
card2_Symbol	varchar(225)	YES		NULL	
fotoProfile	varchar(225)	YES		NULL	
action	int(11)	YES		NULL	
cardComb	int(11)	YES		NULL	
highestCombCard	int(11)	YES		NULL	
highestPlayerCard	int(11)	YES		NULL	

Gambar 2.13

Gambar diatas menunjukkan keterangan field atau komponen yang ada pada tabel userdb, field ‘userid’, ‘username’, ‘password’ dan ‘chips’ digunakan untuk menyimpan data dari pemain, field ‘card1\_Value’ dan ‘card2\_Value’ digunakan untuk menyimpan nilai dari kedua kartu yang didapat pemain, field ‘card1\_Symbol’ dan ‘card2\_Symbol’ digunakan untuk menyimpan simbol atau ‘suit’ dari kedua kartu pemain tersebut.

Selanjutnya field ‘action’ digunakan untuk menyimpan aksi yang dilakukan pemain dalam permainan, aksi-aksi tersebut seperti melakukan *check*, *raise*, *fold* call dll. Field ‘cardComb’ digunakan untuk menyimpan kombinasi yang didapat dari kartu pemain, lalu ‘highestCombCard’ digunakan untuk menyimpan nilai kartu tertinggi dari kartu yang membentuk kombinasi, dan ‘highestPlayerCard’ digunakan untuk menyimpan nilai tertinggi diantara kedua kartu pemain.

## Bab III

### Hasil dan Pembahasan

#### 3.1 Program Java

Hasil Program Java beserta pembahasannya telah kami lampirkan dalam bentuk tautan *Google Document* berikut:

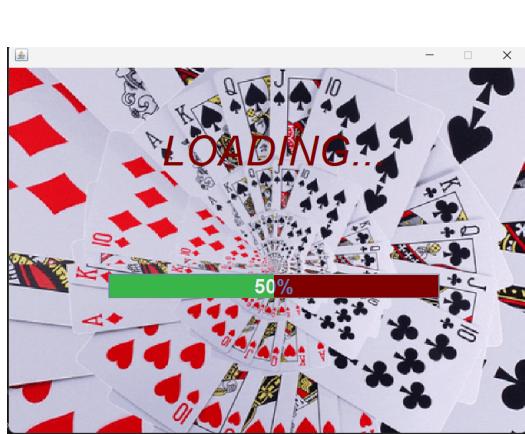
[https://docs.google.com/document/d/1u19bKwNJDM8U\\_NP\\_8ZjDXgCPRH1KNzG7sKzoGIdef4o/edit?usp=sharing](https://docs.google.com/document/d/1u19bKwNJDM8U_NP_8ZjDXgCPRH1KNzG7sKzoGIdef4o/edit?usp=sharing)

#### 3.2 Output Program

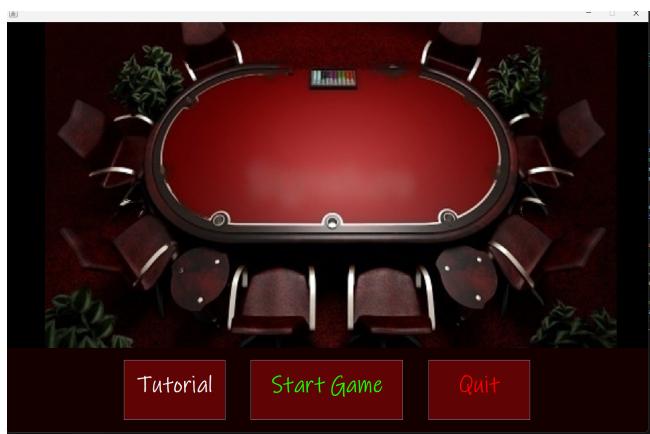
##### 3.2.1 Hasil dan Pembahasan Output Program

###### i. Output Frame Loading dan Menu Utama

Disaat program dimulai maka akan muncul frame loading dengan bar yang akan berjalan dari 0 sampai 100%, lalu setelah itu akan muncul frame menu utama



Gambar 3.1



Gambar 3.2

###### ii. Output Frame Tutorial/ Frame Info

*Frame* tutorial atau info dalam permainan poker memiliki fungsi penting untuk memberikan pemain informasi tentang tata cara bermain dan peraturan permainan. *Frame* ini akan muncul ketika pemain memilih tombol tutorial pada *frame* menu utama, serta dapat diakses kembali saat pemain memilih menu bar *setting* bagian info pada *frame* *gameplay default* dan *gameplay host/client*.



Gambar 3.3

### iii. Output Fase Autentikasi Akun

Pada permainan, terdapat frame autentikasi yang muncul saat pemain memulai permainan dari menu utama. Frame ini menyediakan opsi untuk melakukan log in, sign up, dan cancel. Jika pemain memilih menu log in, maka akan muncul frame log in yang meminta pemain untuk mengisi field username dan password untuk mencocokkan akun pemain dalam database. Sementara itu, jika pemain memilih menu sign up, data yang diisi pada field tersebut akan digunakan untuk mendaftarkan akun baru dalam database. Menu cancel digunakan untuk kembali ke frame menu utama.



Gambar 3.4

The image shows two side-by-side windows. The left window is titled 'Log In' and has fields for 'Username:' and 'Password:', with 'Log In' and 'Cancel' buttons. The right window is titled 'Sign Up' and also has fields for 'Username:' and 'Password:', with 'Submit' and 'Cancel' buttons.

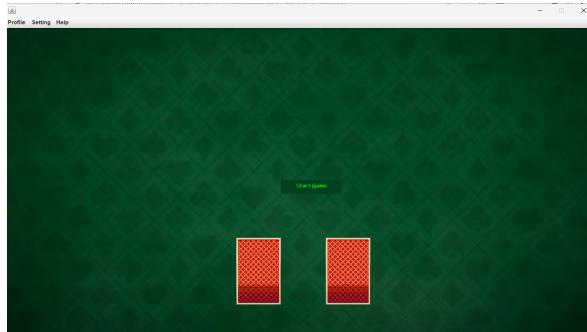
Gambar 3.5

Gambar 3.6

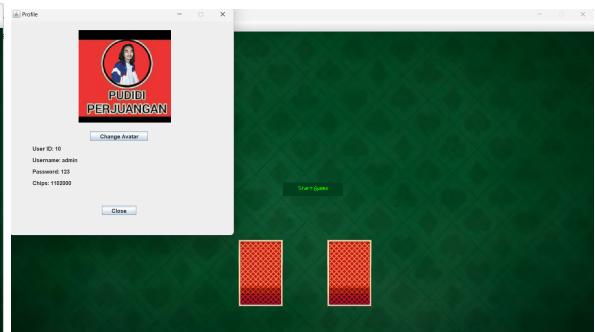
### iv. Output Frame Gameplay Default dan Frame Profile

Setelah melakukan proses *log in*, tampilan *frame gameplay default* akan muncul dengan menu bar di bagian atas kiri. Menu bar ini memiliki opsi *profile*, *settings*, *info*, dan *help*. Pilihan *profile* akan menampilkan *frame profil* pemain dengan *avatar* dan data bisa dilihat pada Gambar 3.8. Pilihan *settings* memiliki submenu *info* dan *logout* untuk keluar dari akun.

Terakhir, menu *help* akan membuka *frame contact developer* yang mencantumkan informasi kontak pengembang, seperti *email*. Pada bagian tengah frame Terdapat tombol ‘*Start Game*’ yang digunakan untuk memulai permainan.



Gambar 3.7

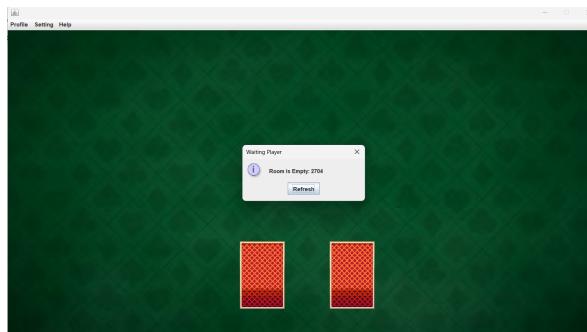


Gambar 3.8

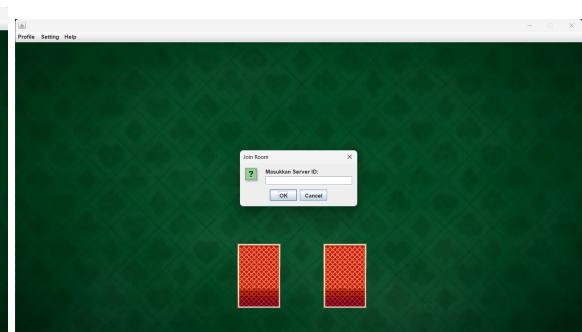
#### v. Output Membuat Room (Host) dan mencari Room (Client)

Setelah pemain memilih tombol ‘*Start Game*’, pemain akan diberikan pilihan untuk membuat (*create*) room baru atau bergabung (*join*) dengan room yang ada. Jika pemain memilih membuat room, program akan menghasilkan sebuah ID room baru dan menambahkan entri baru pada tabel ‘serverdb’ dalam database. Pemain yang membuat room akan menjadi host atau tuan rumah dari room tersebut.

Untuk opsi *join room* pemain akan melihat *field* yang memungkinkan pemain untuk memasukkan ID *room*, dapat dilihat pada gambar 3.10. Jika pemain memilih untuk bergabung, pemain akan berstatus sebagai tamu atau ‘*guest*’ dan bertindak sebagai *client* dari *server*.



Gambar 3.9



Gambar 3.10

#### vi. Output Fase Gameplay

Setelah *host* mendeteksi ada pemain lain yang masuk kedalam *room*, maka akan memasuki fase *gameplay*, pada fase *gameplay* pemain dapat melakukan aksi dengan memilih tombol yang terdapat pada layar. pertama adalah sesi ‘*preflop*’ dimana para pemain melakukan taruhan awal yang telah ditentukan sebanyak 1000 *chips*. Pemain juga dapat melakukan aksi ‘*call*’,

'raise' dan 'fold' terhadap taruhan awal tersebut, dapat dilihat pada gambar 3.11.

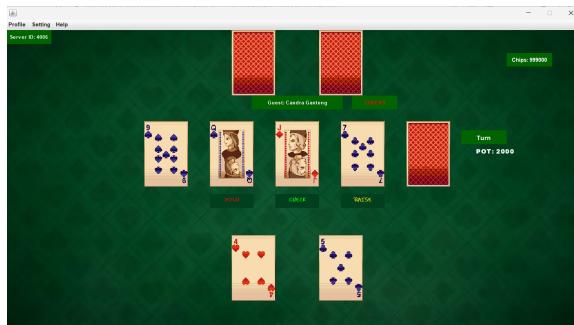
Berikutnya memasuki sesi 'flop' dimana akan dibuka 3 kartu meja yang dapat dilihat pada gambar 3.12, selanjutnya para pemain akan memasuki sesi 'turn' dimana kartu meja yang dibuka akan bertambah menjadi 4 dapat lihat pada gambar 3.13. Terakhir adalah sesi 'river' dimana semua kartu meja akan dibuka yang dapat dilihat pada gambar 3.14, sesi ini merupakan kesempatan terakhir para pemain dalam menentukan jumlah taruhan, jika semua pemain tidak ada yang melakukan 'fold', maka akan masuk tahap penentuan pemenang atau 'Showdown' .



Gambar 3.11



Gambar 3.12



Gambar 3.13



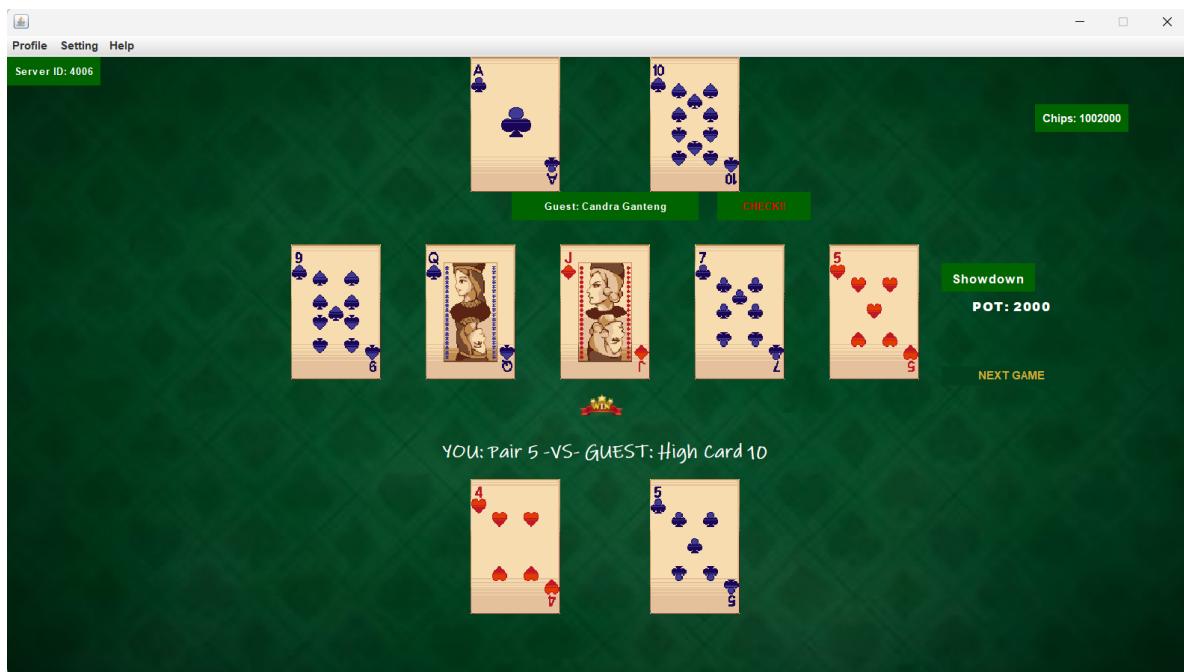
Gambar 3.14

#### vii. Output Fase Penentuan Pemenang

Fase penentuan pemenang tau 'showdown' terjadi setelah fase 'river', pada fase ini program akan melakukan penentuan pemenang didasarkan oleh nilai kombinasi kartu, kartu tertinggi dari kombinasi, dan nilai tertinggi kartu pemain. Pemain dengan nilai lebih besar akan menjadi pemenang dan akan terlihat pada gambar 3.15, untuk pemain yang kalah akan terlihat pada gambar 3.16. Setelah pemenang ditentukan maka *chips* pemain akan bertambah atau berkurang dengan besar *chips* pada 'pot' berdasarkan status pemain sebagai pemenang atau kalah.

Pada bagian tengah kanan frame terdapat tombol 'NEXT GAME' untuk melanjutkan ke permainan selanjutnya, program akan men *generate* kartu pemain dan kartu meja yang baru dan fase *gameplay* akan terulang lagi sampai salah satu pemain keluar meninggalkan *room*.

vii.a. Menang



Gambar 3.15

vii.b. Kalah



Gambar 3.16

## **Bab IV**

### **Penutup**

#### 4.1 Kesimpulan

Dalam penyelesaian tugas besar ini, kami berhasil mengembangkan permainan poker multiplayer yang dilengkapi dengan antarmuka pengguna grafis (GUI) dan integrasi dengan database MySQL. Berdasarkan pengembangan dan pengujian yang dilakukan, kami dapat menyimpulkan hal-hal berikut:

1. Fungsionalitas permainan: Permainan poker multiplayer yang kami kembangkan dapat berjalan dengan baik. Fungsionalitas permainan termasuk pembagian kartu, penghitungan chips, pengaturan taruhan dan total taruhan, serta peringkat kombinasi dan kartu poker telah diimplementasikan dengan benar.
2. Grafis Antarmuka Pengguna(GUI): GUI yang kami desain memungkinkan pemain untuk berinteraksi dengan permainan secara intuitif. Pemain dapat melihat kartu mereka, memilih aksi yang ingin dilakukan, serta melihat informasi penting seperti chips, dan profil yang dimiliki oleh pemain lain.
3. Integrasi dengan Database MySQL: Kami berhasil mengintegrasikan permainan dengan database MySQL untuk menyimpan data menghubungkan pemain dalam proses permainan. Hal ini memungkinkan pemain untuk mengajak pemain lainnya untuk bermain bersama dengan perangkat yang berbeda, sehingga menjamin keamanan dan keadilan dalam permainan

Secara keseluruhan, proyek ini telah mencapai tujuan yang ditetapkan dalam tugas besar ini. Pengembangan permainan poker multiplayer dengan GUI dan integrasi database MySQL memberikan pengalaman bermain yang menyenangkan dan interaktif bagi para pemain. Kami berharap bahwa proyek ini dapat memberikan kontribusi yang positif dalam memperkuat pemahaman dan penerapan konsep pemrograman berorientasi objek yang telah kami pelajari selama mata kuliah ini.

## 4.2 Referensi

Bro Code, "Java GUI: Full Course ☕ (FREE)," Bro Code, Dipublikasikan pada 14 Sep 2020, diakses pada Juni 19, 2023, dari  
<https://www.youtube.com/watch?v=Kmgo00avvEw&pp=ygUHYnJvY29kZQ%3D%3D>.

W3Schools. (2022). Java Variables. Diakses pada Juni 19, 2023, dari  
<https://www.w3schools.com/java/>

W3Schools. (2022). SQL Tutorial. Diakses pada Juni 19, 2023, dari  
<https://www.w3schools.com/sql/>

Stack Overflow. (2010, Mei 15). Connect Java to a MySQL Database. Diakses pada Juni 19, 2023, dari <https://stackoverflow.com/questions/2839321/connect-java-to-a-mysql-database>