

TIC TAC TOE GAMES

by Lkc Malang

Submission date: 20-Jun-2022 12:31PM (UTC+0700)

Submission ID: 1859929303

File name: N_FINAL_PROJECT_OBJECT_ORIENTED_PROGRAM_BISMO_AGUNG_TRI_ACHM.pdf (2.08M)

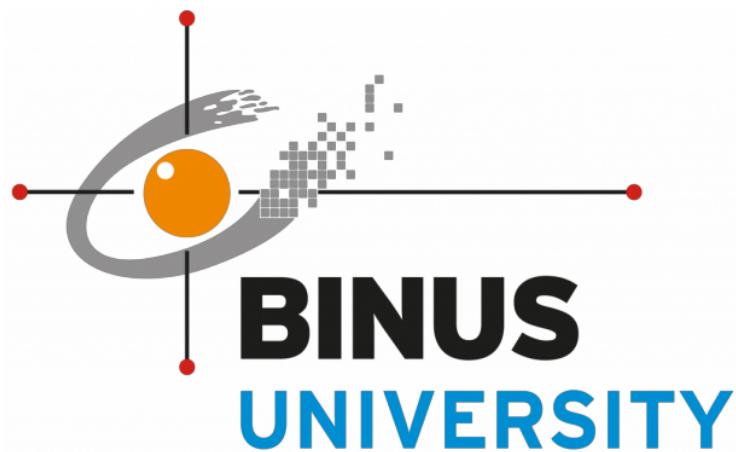
Word count: 6377

Character count: 42389

TIC TAC TOE GAMES

LAPORAN STUDI KASUS

MATA KULIAH COMP6364004 - OBJECT-ORIENTED PROGRAMMING



Oleh:

Afif Ramadhan Yafarayhan - 2440100172

Bismo Agung Tri Achmad Bramantyo - 2440059495

Evan Raditya - 2440085442

Muhammad Rifqi Lutfi - 2440040412

DAFTAR ISI

HALAMAN JUDUL	i
DAFTAR ISI	ii
DAFTAR GAMBAR.....	iii
DAFTAR TABEL	iv
BAB 1 PENDAHULUAN	1
1.1. Penjelasan Program	1
1.2. Teori Object-Oriented	2
BAB II PEMBUATAN PROGRAM	3
2.1. Class Diagram	3
2.2. Use Case Diagram.....	4
2.3. Penjelasan Class dan Modul.....	5
2.4. Tampilan Layar	9
BAB III PEMBAGIAN TUGAS	16
3.1. Rincian Pembagian Tugas per-Individu	16
LAMPIRAN	17

Daftar Gambar

Gambar 2.4. 1 Class Diagram	3
Gambar 2.4. 2 Use Case	4
Gambar 2.4. 3 Choose Board and Start Game	10
Gambar 2.4. 4 Choose Board and Start Game	10
Gambar 2.4. 5 Choose Board and Start Game	11
Gambar 2.4. 6 - Player Turn	11
Gambar 2.4. 7 Game Logic	12
Gambar 2.4. 8 Game Logic	12
Gambar 2.4. 9 Win Logic	12
Gambar 2.4. 10 Win Logic	13
Gambar 2.4. 11 Win Logic	13
Gambar 2.4. 12 Check Score	13
Gambar 2.4. 13 Check Who Play Before (history)	14
Gambar 2.4. 14 Exit	14
Gambar 2.4. 15 History	15

Daftar Tabel

Tabel 3. 1 Tabel Rincian Tugas per Individu	16
Tabel 3. 2 DAFTAR REFERENSI FINAL PROJECT OOP KELOMPOK 3	80

BAB 1

PENDAHULUAN

1.1. Penjelasan Program

Program *Tic Tac Toe Game* merupakan sebuah program permainan yang terinspirasi dari permainan legendaris *Tic Tac Toe*. Permainan *Tic Tac Toe* adalah sebuah permainan yang memiliki tujuan untuk menang dengan cara membuat sebuah garis baik secara vertikal, horizontal, maupun diagonal dalam petak berukuran 3 x 3. Biasanya *Tic Tac Toe* dimainkan oleh dua orang, dengan menggunakan 2 simbol yang berbeda untuk menandai langkah yang diambil oleh pemain. Pemain akan saling bergantian untuk mengisikan simbol dalam petak, pemain yang dapat membuat garis terlebih dahulu dinyatakan sebagai pemenang dari permainan tersebut.

Program yang kami buat memiliki cara kerja yang sama dengan permainan *Tic Tac Toe* klasik, yaitu pemain akan berlomba untuk membuat garis dalam petak agar dapat memenangkan permainan. Pada program kami terdapat pengembangan luas petak yang digunakan untuk bermain, dimana program akan memberikan opsi ukuran petak yang dapat dipilih yaitu ukuran 3x3 atau 5x5. Untuk jumlah minimum simbol yang dapat digunakan sebagai garis pada petak 3x3 adalah 3 simbol yang lurus secara vertikal, horizontal atau diagonal, sedangkan pada petak ukuran 5x5 jumlah simbol yang dibutuhkan sebanyak 4 simbol yang lurus secara vertikal, horizontal atau diagonal. Dalam program ini apabila user memasukan simbol pada petak yang sudah diisi simbol maka kesempatan untuk ronde tersebut akan dilewati. Untuk akhir ronde user dapat memilih untuk melanjutkan permainan melihat skor ataupun mengakhiri permainan.

1.2. Teori Object-Oriented

Object Oriented merupakan suatu strategi pembangunan *software* dengan memperhatikan objek, ciri objek, dan perilakunya atau dengan mengorganisasikan *software* sebagai kumpulan objek yang berisi data dan operasi yang diberlakukan terhadapnya. metode *Object Oriented* banyak dipilih karena aplikasi yang dikembangkan saat ini lebih kompleks, *Object Oriented* juga mempermudah untuk pengembangan secara kolaboratif, karena di dalam *Object Oriented* project dibagi menjadi beberapa *class* (kelompok) selain itu *Object Oriented* juga memiliki banyak keuntungan, antara lain:

- Meningkatkan Produktivitas
- Kecepatan Pengembangan
- Konsistensi
- Meningkatkan Kualitas Software
- Reusability
- Scalability
- Efficiency

Dengan adanya keuntungan tersebut maka pengembangan suatu project dapat dilakukan secara maksimal.

Structure pada *Object Oriented* terbagi menjadi 4 bagian, antara lain:

- Class

Class merupakan tipe data yang ditentukan oleh programmer yang berfungsi sebagai pointer untuk object, method dan attribute.

- Object

Object merupakan instance dari *class* yang dibuat dengan data yang ditentukan secara khusus.

- Method

Method merupakan fungsi yang didefinisikan di dalam *class* yang menggambarkan perilaku suatu objek.

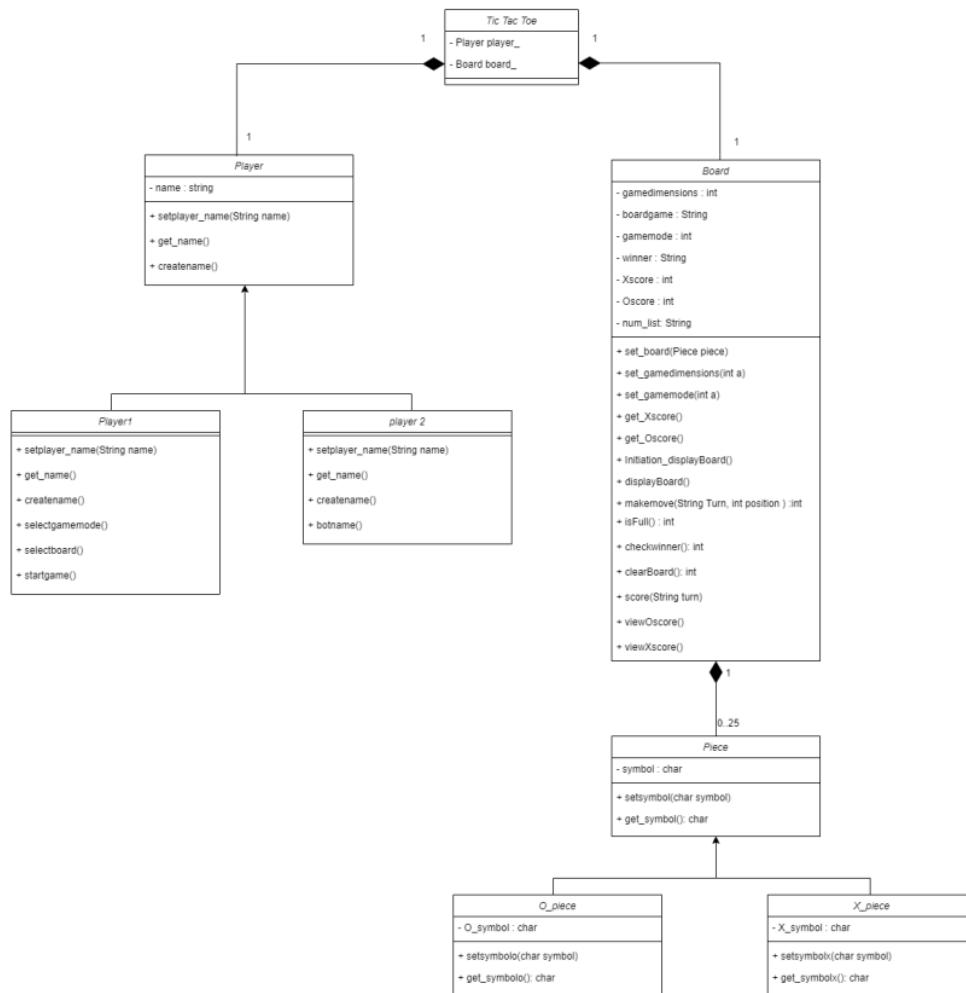
- Attributes

Attribute merupakan data yang membedakan antara objek satu dengan yang lainnya.

BAB II

PEMBUATAN PROGRAM

2.1. Class Diagram



Gambar 2.4. 1 Class Diagram

Class diagram

2.2. Use Case Diagram



2.3. Penjelasan Class dan Modul

Untuk Class Diagram *TicTacToe* kami, kami mempunyai 3 class, yang terdiri dari

1. Class TicTacToe

Class Tic Tac Toe merupakan sebuah *Class* penyusun yang memiliki hubungan *aggregation* dengan *Class Player & Board*, dimana *Class Player & Board* dan tidak dapat berdiri tanpa adanya *Class Tic Tac Toe*.

2. Class Player

pada class ini terdapat 3 method dengan 1 attribute, yaitu method *setplayer_name*, *get_name* & *createname*. untuk method *setplayer_name* merupakan method dimana untuk merubah nilai atribut *private name* pada *Class Player*, untuk method *get_name* merupakan method untuk mengembalikan nilai dari atribut *name*, kemudian untuk method *createname* merupakan method dimana nama yang di masukan oleh *user* kemudian di lempar ke method *setplayer_name*.

3. Class Board

class ini berfungsi sebagai penyokong utama program, dimana akan menampilkan papan petak permainan *tic tac toe*, dimana class ini memiliki 12 method

- *set_board(Piece piece)*

Method ini merupakan konstruktor dari class board yang mempunyai hubungan aggregation dengan class Piece.

- *set_gamedimensions(int a)*

Method ini merupakan setter untuk atribut gamedimensions pada class board, memiliki parameter int a sebagai penentu nilai gamedimensions yang diambil dari inputan user.

- set_gamemode(int a)

Method ini merupakan sebuah method yang mentukan game mode yang dipilih oleh user yaitu VS Computer atau 2 Player.

- get_Xscore()

Merupakan sebuah method getter nilai atribut Xscore pada class board.

Method ini digunakan untuk mengambil score X (player 1).

- get_Oscore()

Merupakan sebuah method getter nilai atribut Oscore pada class board.

Method ini digunakan untuk mengambil score O (player 2).

- initiation_displayBoard()

Method ini berfungsi untuk tampilan inisiasi board pada awal permainan dimana memiliki indeks pada petak sesuai dengan game dimensions yang dipilih.

- displayBoard()

Method ini berfungsi untuk menampilkan board yang sudah berubah nilainya pada indeks-indeks yang dipengaruhi oleh method makemove().

- makemove(String Turn, int position)

Method ini berfungsi untuk meletakan simbol untuk tiap giliran player , baik O maupun X. Memiliki parameter Turn untuk menentukan siapa yang bermain saat ini apabila player 1 maka akan di replace oleh symbol X begitu juga dengan player 2, kemudian position untuk menentukan di petak mana simbol akan diletakan. Method ini mereturn int dimana ketika salah maka akan mereturn nilai 1, apabila benar maka akan mereturn nilai 0.

- **isFull()**

Method ini berfungsi untuk menghitung apakah petak dalam board sudah penuh atau tidak dengan mendeteksi apakah petak tersebut sudah di replace dengan symbol atau belum, apabila petak sudah di replace dengan symbol maka variabel lokal pada method akan bertambah nilainya. Apabila nilai variabel sama dengan petak maka petak dinayatakan sebagai full dan kondisi menjadi draw apabila keadaan win belum ada.

- **checkwinner()**

Method ini berfungsi untuk mendeteksi apakah ada garis yang sudah dibentuk dari kedua player bai secara diagonal, horizontal atau vertikal. Apabila ditemukan keadaan garis sudah terbentuk, maka akan di return nilai 1, namun apabila tidak maka akan di return nilai 0;

- **clearBoard()**

Method ini berfungsi untuk mereset nilai board yang sudah di replace dengan symbol, akan di reset dengan indeks integer dari 1 hingga gamedimensions.

- **score(String Turn)**

Method ini berfungsi untuk menambah score untuk setiap player dengan penentuan melalui parameter Turn, apabila turn X maka variabel Xscore akan ditambah 1, selain itu maka variabel Oscore akan ditambah 1.

- **viewXscore()**

Method ini akan menampilkan variabel Xscore.

- **viewOscore()**

Method ini akan menampilkan variabel Xscore.

Pada class diagram *TicTacToe* kami juga terdapat 4 subclass, yaitu

1. Player 1

untuk subclass player 1 merupakan bagian dari class Player, pada subclass ini 5 method, yaitu *setplayer_name*, *get_name*, *createname*, *selectgamemode*, *selectboard* dan *startgame*. untuk method *selectgamemode* merupakan method dimana player memilih ingin bermain dengan siapa ? bot atau player, untuk method *selectboard* merupakan method dimana player memilih papan permainan yang akan dimainkan apakah 3x3,4x4, atau 5x5, untuk *startgame* merupakan method dimana player memulai permainannya atau mengakhiri permainannya, untuk method *setplayer_name* untuk membuat nama player, *get_name* untuk mereturn nama.

2. Player 2

subclass ini juga bagian dari class player tetapi, pada subclass ini hanya berisikan nama untuk menyimpan nama pemain kedua. method yang digunakan diantaranya *setplayer_name*, *get_name*, *createname*, dan *bot_name*. untuk *setplayer_name* nantinya jika player 1 memilih *selectgamemode* 1 maka *setplayer_name* akan berisikan *bot_name* yang nantinya akan bernama Bot, akan tetapi jika player 1 memilih *selectgamemode* 2 maka *set_playername* akan berisikan inputan nama sesuai dengan inputan user player 2.

3. X piece

subclass ini merupakan bagian dari class piece, subclass ini berisikan simbol X yang merupakan tanda penting pada permainan.

4. O piece

subclass ini merupakan bagian dari class piece, subclass ini berisikan simbol O yang merupakan tanda penting pada permainan.

2.4. Tampilan Layar

1.) Main menu

```
===== Welcome to ANGER ISSUES TIC TAC TOE Game =====  
Rules:  
1. Each Player only have 1 chance each turn, so if you are input the wrong/choose position, your turn will be skipped and you have nothing to put :P  
2. Player who make line first will be the winner  
  
Wanna Play ? 1. YASS    2. CHEECK SOME SCORE    3. CHEECK SLAYY BEFORE    4. UMH NAURR  
  
Enter your choices:
```

Gambar 2.4.1 Main menu

2.) select game mode

```
===== Welcome to ANGER ISSUES TIC TAC TOE Game =====  
Rules:  
1. Each Player only have 1 chance each turn, so if you are input the wrong/choose position, your turn will be skipped and you have nothing to put :P  
2. Player who make line first will be the winner  
  
Wanna Play ? 1. YASS    2. CHEECK SOME SCORE    3. CHEECK SLAYY BEFORE    4. UMH NAURR  
  
Enter your choices:  
1  
  
Please choose your game mode:  
1. Solo (vs Computer)    2. 2 Player  
Enter Your Choice:  
2  
ENTER YOUR PLAYER 1 NAME:  
Udin  
ENTER YOUR PLAYER 2 NAME:  
Bambang
```

Gambar 2.4.2 Select Game Mode

3.) Chooseboard and start game

```
||||| ANGER ISSUES TIC TAC TOE LIST PLAYER |||||
PLAYER 1: Udin      |    PLAYER 2: Bambang
Please Select Board:
1. 3x3  2. 4x4  3. 5x5:
Enter your Choice:
1
| - | - | - |
| 1 | 2 | 3 |
| - | - | - |
| 4 | 5 | 6 |
| - | - | - |
| 7 | 8 | 9 |
| - | - | - |
```

Gambar 2.4. 3 Choose Board and Start Game

```
||||| ANGER ISSUES TIC TAC TOE LIST PLAYER |||||
PLAYER 1: Udin      |    PLAYER 2: Bambang
Please Select Board:
1. 3x3  2. 4x4  3. 5x5:
Enter your Choice:
2
| - | - | - | - |
| 1 | 2 | 3 | 4 |
| - | - | - | - |
| 5 | 6 | 7 | 8 |
| - | - | - | - |
| 9 | 1 | 2 | 3 |
| - | - | - | - |
| 4 | 5 | 6 | 7 |
| - | - | - | - |
```

Gambar 2.4. 4 Choose Board and Start Game

```

||||| ANGER ISSUES TIC TAC TOE LIST PLAYER |||||
PLAYER 1: Udin      |    PLAYER 2: Bambang
Please Select Board:

1. 3x3  2. 4x4  3. 5x5:
Enter your Choice:
3
| - | - | - |
| 1 | 2 | 3 | | |
|---|---|---|---|---|
| 6 | 7 | 8 | 9 | 10 |
| - | - | - | - |
| 11 | 12 | 13 | 14 | 15 |
| - | - | - | - |
| 16 | 8 | 9 | 10 | 11 |
| - | - | - | - |
| 12 | 13 | 14 | 15 | 16 |

```

Gambar 2.4. 5 Choose Board and Start Game

4.) player turn

```

Udin(X) Turn:
1
| - | - | - |
| X | 2 | 3 |
| - | - | - |
| 4 | 5 | 6 |
| - | - | - |
| 7 | 8 | 9 |
| - | - | - |

Position selected

Bambang(O) Turn:
5
| - | - | - |
| X | 2 | 3 |
| - | - | - |
| 4 | 0 | 6 |
| - | - | - |
| 7 | 8 | 9 |
| - | - | - |

Position selected

```

Gambar 2.4. 6 - Player Turn

5.) game logic

```
Udin(X) Turn:  
2  
| - | - | - |  
| X | X | 3 |  
| - | - | - |  
| 4 | 0 | 6 |  
| - | - | - |  
| 7 | 8 | 9 |  
| - | - | - |  
  
Position selected  
  
Bambang(O) Turn:  
5  
  
Position selected  
  
Position selected  
  
Udin(X) Turn:  
3  
| - | - | - |  
| X | X | X |  
| - | - | - |  
| 4 | 0 | 6 |  
| - | - | - |  
| 7 | 8 | 9 |  
| - | - | - |  
  
Position selected
```

Gambar 2.4. 7 Game Logic

```
Udin(X) Turn:  
11  
Input invalid, enter another position  
  
Udin(X) Turn:  
3  
| - | - | - |  
| X | X | X |  
| - | - | - |  
| 4 | 0 | 6 |  
| - | - | - |  
| 7 | 8 | 9 |  
| - | - | - |  
  
Position selected
```

Gambar 2.4. 8 Game Logic

6.) Win logic

```
Udin(X) Turn:  
3  
| - | - | - |  
| X | X | X |  
| - | - | - |  
| 4 | 0 | 6 |  
| - | - | - |  
| 7 | 8 | 9 |  
| - | - | - |  
  
Position selected  
  
OUR YASS QUEEN IS:  
Udin
```

Gambar 2.4. 9 Win Logic

```

Bambang(O) Turn:
11
| - | - | - | - |
| 0 | 2 | 3 | X |
| - | - | - | - |
| X | 0 | 7 | X |
| - | - | - | - |
| 9 | 10 | 0 | 12 |
| - | - | - | - |
| 13 | 14 | 15 | 16 |
| - | - | - | - |

Position selected

OUR YASS QUEEN IS:
Bambang

```

Gambar 2.4. 10 Win Logic

```

Udin(X) Turn:
16
| - | - | - | - | - |
| X | 0 | 0 | 4 | 5 |
| - | - | - | - | - |
| X | 7 | 8 | 9 | 10 |
| - | - | - | - | - |
| X | 12 | 13 | 14 | 15 |
| - | - | - | - | - |
| X | 17 | 18 | 19 | 20 |
| - | - | - | - | - |
| 0 | 22 | 23 | 24 | 25 |

Position selected

OUR YASS QUEEN IS:
Udin

```

Gambar 2.4. 11 Win Logic

7. check score

```

----- Welcome to ANGER ISSUES TIC TAC TOE Game -----
Rules:
1. Each Player only have 1 chance each turn, so if you are input the wrong/chooseen position, your turn will be skipped and you have nothing to put :P
2. Player who make line first will be the winner

Wanna Play ? 1. YASS    2. CHEECK SOME SCORE    3. CHEECK SLAYY BEFOR    4. UMH NAURR

Enter your choices:
2

Score:
Udin:
2

Bambang:
1

```

Gambar 2.4. 12 Check Score

8. Check who play before (history)

```
===== Welcome to ANGER ISSUES TIC TAC TOE Game =====  
Rules:  
1. Each Player only have 1 chance each turn, so if you are input the wrong/choose position, your turn will be skipped and you have nothing to put :P  
2. Player who make line first will be the winner  
  
Wanna Play ? 1. YASS    2. CHEECK SOME SCORE    3. CHEECK SLAYY BEFOR    4. UMH NAURR  
  
Enter your choices:  
3  
Hystory Match 19/06/2022 20:39/n  
Udin:0/n  
Slay:1/n  
/n
```

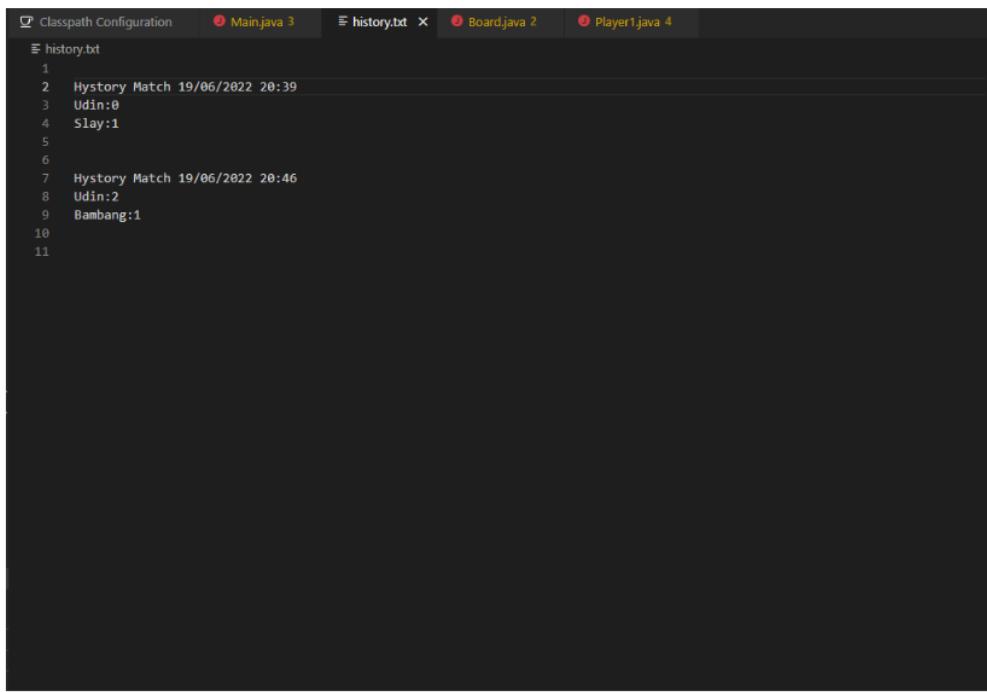
Gambar 2.4. 13 Check Who Play Before (history)

9. Exit

```
===== Welcome to ANGER ISSUES TIC TAC TOE Game =====  
Rules:  
1. Each Player only have 1 chance each turn, so if you are input the wrong/choose position, your turn will be skipped and you have nothing to put :P  
2. Player who make line first will be the winner  
  
Wanna Play ? 1. YASS    2. CHEECK SOME SCORE    3. CHEECK SLAYY BEFOR    4. UMH NAURR  
  
Enter your choices:  
4  
  
===== OK THANKS FOR USING THIS APP WITH ANGER ISSUES =====  
PS D:\Tugas Binus\SMT 4\OOP\TUGAS\LAB> █
```

Gambar 2.4. 14 Exit

10.) Check update gameplay history



The screenshot shows a terminal window with several tabs at the top: Classpath Configuration, Main.java 3, history.txt (the active tab), Board.java 2, and Player1.java 4. The history.txt tab displays the following text:

```
1 Hystory Match 19/06/2022 20:39
2 Udin:0
3 Slay:1
4
5
6
7 Hystory Match 19/06/2022 20:46
8 Udin:2
9 Bambang:1
10
11
```

Gambar 2.4. 15 History

BAB III

PEMBAGIAN TUGAS

3.1. Rincian Pembagian Tugas per-Individu

Tabel 3. 1 Tabel Rincian Tugas per Individu

Nama	NIM	Rincian Tugas
Bismo Agung Tri Achmad Bramantyo	2440059495	<ul style="list-style-type: none">- Koding awal semua Class- Memperbaiki Class Diagram- Menyelesaikan dan mendefinisikan method pada setiap class- Finalisasi Koding Class Board & Player- Membuat Template Laporan- Membuat PPT
Evan Raditya	2440085442	<ul style="list-style-type: none">- Membantu koding append text in exist file.- Membantu mengoding dan merapikan beberapa class.- Membantu memperbaiki class diagram.
Afif Ramadhani Yafarayhan	2440100172	<ul style="list-style-type: none">- Membantu membuat class diagram- Membantu memperbaiki class diagram.

		<ul style="list-style-type: none"> - Membuat laporan bab 1 dan bab 2
Muhammad Rifqi Lutfi	2440040412	<ul style="list-style-type: none"> - Membantu membuat dan memperbaiki class diagram, dan PPT

LAMPIRAN

i. Source Code

```

class TicTacToe{

    Board board_;
    Player player_;

    TicTacToe(Board board_, Player player_){
        this.board_ = board_;
        this.player_= player_;
    }
}

```

```
import java.util.*;  
  
class Player{  
  
    protected String name;  
  
    public void setplayer_name(String name){  
        this.name = name;  
    }  
  
    public String get_name(){  
        return name;  
    }  
  
    public void createname(){  
        Scanner inp = new Scanner(System.in);  
        String name;  
  
        System.out.println("ENTER YOUR NAME: ");  
        name = inp.nextLine();  
  
        setplayer_name(name);  
    }  
}
```

```
}
```

```
import java.util.*;
```

```
class Player1 extends Player{
```

```
    //private String name;
```

```
    public void setplayer_name (String name){2
```

```
        this.name = name;
```

```
    }
```

```
    public String get_name(){
```

```
        return name;
```

```
    }
```

```
    public void createname(){
```

```
        Scanner inp = new Scanner(System.in);
```

```
        String name;
```

```
        System.out.println("ENTER YOUR PLAYER 1 NAME: ");
```

```
        name = inp.nextLine();
```

```
        setplayer_name(name);

    }

public int selectgamemode(){

    Scanner inp = new Scanner(System.in);

    int input;

    System.out.println("\n\nPlease choose your game mode: ");
    System.out.println("1. Solo (vs Computer)  2. 2 Player\n");

    do{
        System.out.println("Enter Your Choice: ");
        input = inp.nextInt();
    }while(input<1 && input>2);

    return input;

}

public int selectboard(){
```

```
Scanner inp = new Scanner(System.in);
int input;

System.out.println("\nPlease Select Board: ");
System.out.println("\n\n1. 3x3 2. 4x4 3. 5x5: \n");

do{
    System.out.println("Enter your Choice:");
    input = inp.nextInt();

}while(input < 1 && input > 3);

return input;

}

public int startgame(){ //ini start sama end jadi satu

    int play;
    Scanner inp = new Scanner(System.in);

    do{
        System.out.println("\nWanna Play ? 1. YASS      2. CHEECK
SOME SCORE      3. CHEECK SLAYY BEfore 4. UMH
NAURR\n");
        System.out.println("\nEnter your choices: ");
    }
}
```

```
play = inp.nextInt();

}while(play < 1 && play > 4);

return play;

}

//cleanboard udah ada di board nya, habis permainan langsung
otomatis hapus.

}

import java.util.*;

class Player2 extends Player{

//private String name;

    public void setplayer_name(String name){

        this.name = name;

    }

    public String get_name(){

        return name;

    }

    public void createname(){


```

```
Scanner inp = new Scanner(System.in);

String name;

System.out.println("ENTER YOUR PLAYER 2 NAME: ");
name = inp.nextLine();

setplayer_name(name);

}

public void botname(){

String name = "bot";

setplayer_name(name);

}

}

import java.util.*;

class Board{

private int gamedimensions;

private ArrayList<String> boardgame;
```

```
private int gamemode;  
private String winner;  
private int Xscore = 0;  
private int Oscore = 0;  
private ArrayList<String> num_list = new  
ArrayList<String>(gamedimensions*gamedimensions);  
//private String winstatus;  
Piece piece;  
  
Board(int a){  
    boardgame = new ArrayList<String>(a);  
}  
  
public void set_board(Piece piece){  
  
    this.piece = piece;  
  
}  
  
① public void set_gamedimensions(int a){  
    this.gamedimensions = a;  
}  
  
public void set_gamemode(int a){  
    this.gamemode = a;  
}  
public int get_Xscore(){
```

```
        return Xscore;
    }

    public int get_Oscore(){
        return Oscore;
    }

    public void clear_list(){
        this.num_list.clear();
    }

    public void Initiation_displayBoard(){

        int b = gamedimensions*gamedimensions;

        for (int i = 0 ; i < b ; i++){
            boardgame.add(String.valueOf(i+1));
        }

        if(gamedimensions == 3){

            System.out.println("| - | - | - |");

            System.out.println("| " + boardgame.get(0) + " | " +
                boardgame.get(1) + " | " + boardgame.get(2) + " |");

            System.out.println("| - | - | - |");

            System.out.println("| " + boardgame.get(3) + " | " +
```

```

1
boardgame.get(4) + " | " + boardgame.get(5) + " | ");

System.out.println("| - | - | - |");

System.out.println(" | " + boardgame.get(6) + " | " +
boardgame.get(7) + " | " + boardgame.get(8) + " | ");

System.out.println(" | - | - | - |");

}

else if(gamedimensions == 4){

    System.out.println(" | - | - | - | - | - |");

    System.out.println(" | " + boardgame.get(0) + " | " +
boardgame.get(1) + " | " + boardgame.get(2) + " | " +
boardgame.get(3) + " | ");

1
    System.out.println(" | - | - | - | - | - |");

    System.out.println(" | " + boardgame.get(4) + " | " +
boardgame.get(5) + " | " + boardgame.get(6) + " | " +
boardgame.get(7) + " | ");

1
    System.out.println(" | - | - | - | - | - |");

    System.out.println(" | " + boardgame.get(8) + " | " +
boardgame.get(9) + " | " + boardgame.get(10) + " | " +
boardgame.get(11) + " | ");

1
    System.out.println(" | - | - | - | - | - |");

    System.out.println(" | " + boardgame.get(12) + " | " +
boardgame.get(13) + " | " + boardgame.get(14) + " | " +
boardgame.get(15) + " | ");

    System.out.println(" | - | - | - | - |");

}

else if(gamedimensions == 5){

```

```

        System.out.println("| - | - | - | - | - |");
        System.out.println("| " + boardgame.get(0) + " | " +
boardgame.get(1) + " | " + boardgame.get(2) + " | " +
boardgame.get(3) + " | " + boardgame.get(4) + " |");
        System.out.println("| - | - | - | - | - |");
        System.out.println("| " + boardgame.get(5) + " | " +
boardgame.get(6) + " | " + boardgame.get(7) + " | " +
boardgame.get(8) + " | " + boardgame.get(9) + " |");
①      System.out.println("| - | - | - | - | - |");
        System.out.println("| " + boardgame.get(10) + " | " +
boardgame.get(11) + " | " + boardgame.get(12) + " | " +
boardgame.get(13) + " | " + boardgame.get(14) + " |");
①      System.out.println("| - | - | - | - | - |");
        System.out.println("| " + boardgame.get(15) + " | " +
boardgame.get(16) + " | " + boardgame.get(17) + " | " +
boardgame.get(18) + " | " + boardgame.get(19) + " |");
①      System.out.println("| - | - | - | - | - |");
        System.out.println("| " + boardgame.get(20) + " | " +
boardgame.get(21) + " | " + boardgame.get(22) + " | " +
boardgame.get(23) + " | " + boardgame.get(24) + " |");

    }

}

public void displayBoard(){

    if(gamedimensions == 3){

```

```

        System.out.println("| - | - | - |");

        System.out.println("| " + boardgame.get(0) + " | " +
boardgame.get(1) + " | " + boardgame.get(2) + " |");

        System.out.println("| - | - | - |");

        System.out.println("| " + boardgame.get(3) + " | " +
boardgame.get(4) + " | " + boardgame.get(5) + " |");

        System.out.println("| - | - | - |");

        System.out.println("| " + boardgame.get(6) + " | " +
boardgame.get(7) + " | " + boardgame.get(8) + " |");

        System.out.println("| - | - | - |");

    }

else if(gamedimensions == 4){

    System.out.println("| - | - | - | - | - |");

    System.out.println("| " + boardgame.get(0) + " | " +
boardgame.get(1) + " | " + boardgame.get(2) + " | " +
boardgame.get(3) + " |");

    System.out.println("| - | - | - | - | - |");

    System.out.println("| " + boardgame.get(4) + " | " +
boardgame.get(5) + " | " + boardgame.get(6) + " | " +
boardgame.get(7) + " |");

    System.out.println("| - | - | - | - | - |");

    System.out.println("| " + boardgame.get(8) + " | " +
boardgame.get(9) + " | " + boardgame.get(10) + " | " +
boardgame.get(11) + " |");

    System.out.println("| - | - | - | - | - |");

    System.out.println("| " + boardgame.get(12) + " | " +
boardgame.get(13) + " | " + boardgame.get(14) + " | " +
boardgame.get(15) + " |");
}

```

```

boardgame.get(15) + " |");
System.out.println("| - | - | - | - | - |");

}

else if(gamedimensions == 5){

System.out.println("| - | - | - | - | - | - |");
System.out.println("| " + boardgame.get(0) + " | " +
boardgame.get(1) + " | " + boardgame.get(2) + " | " +
boardgame.get(3) + " | " + boardgame.get(4) + " |");
System.out.println("| - | - | - | - | - | - |");
System.out.println("| " + boardgame.get(5) + " | " +
boardgame.get(6) + " | " + boardgame.get(7) + " | " +
boardgame.get(8) + " | " + boardgame.get(9) + " |");
① System.out.println("| - | - | - | - | - | - |");
System.out.println("| " + boardgame.get(10) + " | " +
boardgame.get(11) + " | " + boardgame.get(12) + " | " +
boardgame.get(13) + " | " + boardgame.get(14) + " |");
① System.out.println("| - | - | - | - | - | - |");
System.out.println("| " + boardgame.get(15) + " | " +
boardgame.get(16) + " | " + boardgame.get(17) + " | " +
boardgame.get(18) + " | " + boardgame.get(19) + " |");
① System.out.println("| - | - | - | - | - |");
System.out.println("| " + boardgame.get(20) + " | " +
boardgame.get(21) + " | " + boardgame.get(22) + " | " +
boardgame.get(23) + " | " + boardgame.get(24) + " |");

}

```

```
}

public int makemove(String Turn, int position){

    int parameter = 0;

    if(boardgame.get(position - 1).equals(String.valueOf(position))){


        boardgame.set(position - 1, Turn);

        displayBoard();

    }

    else{



        System.out.println("\nPosition selected");



        parameter = 1;

    }

    return parameter;

}
```

```
public int isFull(){

    int parameter = 0;

    int b = gamedimensions * gamedimensions;

    int count = 0;

    if(num_list.isEmpty() == true){

        parameter = 0;

    }

    else{

        for( int i = 0 ; i < b ; i++ ){

            if(num_list.get(i) == "X" || num_list.get(i) == "O"){

                count = count + 1;

            }

            else{

                count = count + 0;

            }

        }

        if(count == b){

            parameter = 1;

        }

        else{
```

```
        parameter = 0;  
    }  
}  
  
return parameter;  
}  
  
public int checkwinner(){  
  
    int parameter = 0;  
  
    switch (gamedimensions) {  
  
        case 3:  
            //player 1  
  
            //mendarat  
            if((boardgame.get(0) == "X") && (boardgame.get(1) == "X")  
&& (boardgame.get(2) == "X")){//012  
  
                parameter = 1;  
                winner = "X";  
  
            }  
            else if((boardgame.get(3) == "X") && (boardgame.get(4) ==
```

```
"X") && (boardgame.get(5) == "X")){//345

    parameter = 1;
    winner = "X";

}

else if((boardgame.get(6) == "X") && (boardgame.get(7) ==
"X") && (boardgame.get(8) == "X")){//678

    parameter = 1;
    winner = "X";

}

//menurun

else if((boardgame.get(0) == "X") && (boardgame.get(3) ==
"X") && (boardgame.get(6) == "X")){//036

    parameter = 1;
    winner = "X";

}

else if((boardgame.get(1) == "X") && (boardgame.get(4) ==
"X") && (boardgame.get(7) == "X")){//147

    parameter = 1;
```

```
    winner = "X";  
  
}  
  
else if((boardgame.get(2) == "X") && (boardgame.get(5) ==  
"X") && (boardgame.get(8) == "X")){//258  
  
    parameter = 1;  
  
    winner = "X";  
  
}  
  
//diagonal  
  
else if((boardgame.get(0) == "X") && (boardgame.get(4) ==  
"X") && (boardgame.get(8) == "X")){//048  
  
    parameter = 1;  
  
    winner = "X";  
  
}  
  
else if((boardgame.get(2) == "X") && (boardgame.get(4) ==  
"X") && (boardgame.get(6) == "X")){//246  
  
    parameter = 1;  
  
    winner = "X";  
  
}
```

```
//player 2

//mendatar

else if((boardgame.get(0) == "O") && (boardgame.get(1) ==
 "O") && (boardgame.get(2) == "O")){//012

    parameter = 1;
    winner = "O";

}

else if((boardgame.get(3) == "O") && (boardgame.get(4) ==
 "O") && (boardgame.get(5) == "O")){//345

    parameter = 1;
    winner = "O";

}

else if((boardgame.get(6) == "O") && (boardgame.get(7) ==
 "O") && (boardgame.get(8) == "O")){//678

    parameter = 1;
    winner = "O";

}

//menurun
```

```
        else if((boardgame.get(0) == "O") && (boardgame.get(3) ==  
"O") && (boardgame.get(6) == "O")){//036  
  
        parameter = 1;  
        winner = "O";  
  
    }  
  
    else if((boardgame.get(1) == "O") && (boardgame.get(4) ==  
"O") && (boardgame.get(7) == "O")){//147  
  
        parameter = 1;  
        winner = "O";  
  
    }  
  
    else if((boardgame.get(2) == "O") && (boardgame.get(5) ==  
"O") && (boardgame.get(8) == "O")){//258  
  
        parameter = 1;  
        winner = "O";  
  
    }  
  
    //diagonal  
  
    else if((boardgame.get(0) == "O") && (boardgame.get(4) ==  
"O") && (boardgame.get(8) == "O")){//048
```

```
parameter = 1;  
winner = "O";  
  
}  
else if((boardgame.get(2) == "O") && (boardgame.get(4) ==  
"O") && (boardgame.get(6) == "O")){//246  
  
parameter = 1;  
winner = "O";  
  
}  
else{  
parameter = 0;  
}  
  
break;  
  
case 4:  
  
//player 1  
  
//mendatar  
if((boardgame.get(0) == "X") && (boardgame.get(1) == "X")  
&& (boardgame.get(2) == "X")){//012  
  
parameter = 1;  
winner = "X";
```

```
    }

    else if((boardgame.get(1) == "X") && (boardgame.get(2) ==
"X") && (boardgame.get(3) == "X")){//123

        parameter = 1;
        winner = "X";


    }

    else if((boardgame.get(4) == "X") && (boardgame.get(5) ==
"X") && (boardgame.get(6) == "X")){//456

        parameter = 1;
        winner = "X";


    }

    else if((boardgame.get(5) == "X") && (boardgame.get(6) ==
"X") && (boardgame.get(7) == "X")){//567

        parameter = 1;
        winner = "X";


    }

    else if((boardgame.get(8) == "X") && (boardgame.get(9) ==
"X") && (boardgame.get(10) == "X")){//8910

        parameter = 1;
        winner = "X";


    }
}
```

```
    }

    else if((boardgame.get(9) == "X") && (boardgame.get(10) ==
"X") && (boardgame.get(11) == "X")){//91011

        parameter = 1;
        winner = "X";


    }

    else if((boardgame.get(12) == "X") && (boardgame.get(13)
== "X") && (boardgame.get(14) == "X")){//121314

        parameter = 1;
        winner = "X";


    }

    else if((boardgame.get(13) == "X") && (boardgame.get(14)
== "X") && (boardgame.get(15) == "X")){//131415

        parameter = 1;
        winner = "X";


    }

//menurun

    else if((boardgame.get(0) == "X") && (boardgame.get(4) ==
"X") && (boardgame.get(8) == "X")){//048
```

```
parameter = 1;  
winner = "X";  
  
}  
  
else if((boardgame.get(4) == "X") && (boardgame.get(8) ==  
"X") && (boardgame.get(12) == "X")){//4812  
  
parameter = 1;  
winner = "X";  
  
}  
  
else if((boardgame.get(1) == "X") && (boardgame.get(5) ==  
"X") && (boardgame.get(9) == "X")){//159  
  
parameter = 1;  
winner = "X";  
  
}  
  
else if((boardgame.get(5) == "X") && (boardgame.get(9) ==  
"X") && (boardgame.get(13) == "X")){//5913  
  
parameter = 1;  
winner = "X";  
  
}  
  
else if((boardgame.get(2) == "X") && (boardgame.get(6) ==  
"X") && (boardgame.get(10) == "X")){//2610
```

```
parameter = 1;  
winner = "X";  
  
}  
  
else if((boardgame.get(6) == "X") && (boardgame.get(10) ==  
"X") && (boardgame.get(14) == "X")){//61014  
  
parameter = 1;  
winner = "X";  
  
}  
  
else if((boardgame.get(3) == "X") && (boardgame.get(7) ==  
"X") && (boardgame.get(11) == "X")){//3711  
  
parameter = 1;  
winner = "X";  
  
}  
  
else if((boardgame.get(7) == "X") && (boardgame.get(11) ==  
"X") && (boardgame.get(15) == "X")){//71115  
  
parameter = 1;  
winner = "X";  
  
}
```

```
//diagonal

else if((boardgame.get(0) == "X") && (boardgame.get(5) ==
"X") && (boardgame.get(10) == "X")){//0510

    parameter = 1;
    winner = "X";

}

else if((boardgame.get(5) == "X") && (boardgame.get(10) ==
"X") && (boardgame.get(15) == "X")){//51015

    parameter = 1;
    winner = "X";

}

else if((boardgame.get(3) == "X") && (boardgame.get(6) ==
"X") && (boardgame.get(9) == "X")){//369

    parameter = 1;
    winner = "X";

}

else if((boardgame.get(6) == "X") && (boardgame.get(9) ==
"X") && (boardgame.get(12) == "X")){//6912

    parameter = 1;
    winner = "X";
```

```
}

//player 2

//mendarat

else if((boardgame.get(0) == "O") && (boardgame.get(1) ==
"O") && (boardgame.get(2) == "O")){//012

    parameter = 1;

    winner = "O";


}

else if((boardgame.get(1) == "O") && (boardgame.get(2) ==
"O") && (boardgame.get(3) == "O")){//123

    parameter = 1;

    winner = "O";


}

else if((boardgame.get(4) == "O") && (boardgame.get(5) ==
"O") && (boardgame.get(6) == "O")){//456

    parameter = 1;

    winner = "O";
```

```
    }

    else if((boardgame.get(5) == "O") && (boardgame.get(6) ==
"O") && (boardgame.get(7) == "O")){//567

        parameter = 1;
        winner = "O";

    }

    else if((boardgame.get(8) == "O") && (boardgame.get(9) ==
"O") && (boardgame.get(10) == "O")){//8910

        parameter = 1;
        winner = "O";

    }

    else if((boardgame.get(9) == "O") && (boardgame.get(10) ==
"O") && (boardgame.get(11) == "O")){//91011

        parameter = 1;
        winner = "O";

    }

    else if((boardgame.get(12) == "O") && (boardgame.get(13) ==
"O") && (boardgame.get(14) == "O")){//121314

        parameter = 1;
        winner = "O";

    }

}
```

```
    }

    else if((boardgame.get(13) == "O") && (boardgame.get(14)
== "O") && (boardgame.get(15) == "O")){//131415

        parameter = 1;
        winner = "O";

    }

    //menurun

    else if((boardgame.get(0) == "O") && (boardgame.get(4) ==
"O") && (boardgame.get(8) == "O")){//048

        parameter = 1;
        winner = "O";

    }

    else if((boardgame.get(4) == "O") && (boardgame.get(8) ==
"O") && (boardgame.get(12) == "O")){//4812

        parameter = 1;
        winner = "O";

    }

    else if((boardgame.get(1) == "O") && (boardgame.get(5) ==
"O") && (boardgame.get(9) == "O")){//159
```

```
parameter = 1;  
winner = "O";  
  
}  
  
else if((boardgame.get(5) == "O") && (boardgame.get(9) ==  
"O") && (boardgame.get(13) == "O")){//5913  
  
parameter = 1;  
winner = "O";  
  
}  
  
else if((boardgame.get(2) == "O") && (boardgame.get(6) ==  
"O") && (boardgame.get(10) == "O")){//2610  
  
parameter = 1;  
winner = "O";  
  
}  
  
else if((boardgame.get(6) == "O") && (boardgame.get(10) ==  
"O") && (boardgame.get(14) == "O")){//61014  
  
parameter = 1;  
winner = "O";  
  
}  
  
else if((boardgame.get(3) == "O") && (boardgame.get(7) ==  
"O") && (boardgame.get(11) == "O")){//3711
```

```
parameter = 1;  
winner = "O";  
  
}  
  
else if((boardgame.get(7) == "O") && (boardgame.get(11) ==  
"O") && (boardgame.get(15) == "O")){//71115  
  
parameter = 1;  
winner = "O";  
  
}  
  
//diagonal  
  
else if((boardgame.get(0) == "O") && (boardgame.get(5) ==  
"O") && (boardgame.get(10) == "O")){//0510  
  
parameter = 1;  
winner = "O";  
  
}  
else if((boardgame.get(5) == "O") && (boardgame.get(10) ==  
"O") && (boardgame.get(15) == "O")){//51015  
  
parameter = 1;  
winner = "O";
```

```
    }

    else if((boardgame.get(3) == "O") && (boardgame.get(6) ==
"O") && (boardgame.get(9) == "O")){//369

        parameter = 1;
        winner = "O";

    }

    else if((boardgame.get(6) == "O") && (boardgame.get(9) ==
"O") && (boardgame.get(12) == "O")){//6912

        parameter = 1;
        winner = "O";

    }

    else{

        parameter = 0;
    }

    break;

case 5:

//player 1

//mendarat

if((boardgame.get(0) == "X") && (boardgame.get(1) == "X") && (boardgame.get(2) == "X") && (boardgame.get(3) ==
```

```
"X")){//[0],[1],[2],[3]

    parameter = 1;
    winner = "X";

}

else if((boardgame.get(1) == "X") && (boardgame.get(2) ==
"X") && (boardgame.get(3) == "X") && (boardgame.get(4) ==
"X")){//[1],[2],[3],[4]

    parameter = 1;
    winner = "X";

}

else if((boardgame.get(5) == "X") && (boardgame.get(6) ==
"X") && (boardgame.get(7) == "X") && (boardgame.get(8) ==
"X")){//[5],[6],[7],[8]

    parameter = 1;
    winner = "X";

}

else if((boardgame.get(6) == "X") && (boardgame.get(7) ==
"X") && (boardgame.get(8) == "X") && (boardgame.get(9) ==
"X")){//[6],[7],[8],[9]

    parameter = 1;
    winner = "X";
```

```

    }

    else if((boardgame.get(10) == "X") && (boardgame.get(11)
    == "X") && (boardgame.get(12) == "X") && (boardgame.get(13) ==
    "X")){/[10],[11],[12],[13]

        parameter = 1;
        winner = "X";

    }

    else if((boardgame.get(11) == "X") && (boardgame.get(12)
    == "X") && (boardgame.get(13) == "X") && (boardgame.get(14) ==
    "X")){/[11],[12],[13],[14]

        parameter = 1;
        winner = "X";

    }

    else if((boardgame.get(15) == "X") && (boardgame.get(16)
    == "X") && (boardgame.get(17) == "X") && (boardgame.get(18) ==
    "X")){/[15],[16],[17],[18]

        parameter = 1;
        winner = "X";

    }

    else if((boardgame.get(16) == "X") && (boardgame.get(17)
    == "X") && (boardgame.get(18) == "X") && (boardgame.get(19) ==
    "X")){/[16],[17],[18],[19]

        parameter = 1;
    }
}

```

```
    winner = "X";  
  
}  
  
else if((boardgame.get(20) == "X") && (boardgame.get(21)  
== "X") && (boardgame.get(22) == "X") && (boardgame.get(23) ==  
"X")){//[20],[21],[22],[23]  
  
    parameter = 1;  
    winner = "X";  
  
}  
  
else if((boardgame.get(21) == "X") && (boardgame.get(22)  
== "X") && (boardgame.get(23) == "X") && (boardgame.get(24) ==  
"X")){//[21],[22],[23],[24]  
  
    parameter = 1;  
    winner = "X";  
  
}  
  
//menurun  
  
else if((boardgame.get(0) == "X") && (boardgame.get(5) ==  
"X") && (boardgame.get(10) == "X") && (boardgame.get(15) ==  
"X")){//[0],[5],[10],[15]  
  
    parameter = 1;  
    winner = "X";
```

```

    }

    else if((boardgame.get(5) == "X") && (boardgame.get(10) ==
    "X") && (boardgame.get(15) == "X") && (boardgame.get(20) ==
    "X")){/[5],[10],[15],[20]

        parameter = 1;

        winner = "X";


    }

    else if((boardgame.get(1) == "X") && (boardgame.get(6) ==
    "X") && (boardgame.get(11) == "X") && (boardgame.get(16) ==
    "X")){/[1],[6],[11],[16]

        parameter = 1;

        winner = "X";


    }

    else if((boardgame.get(6) == "X") && (boardgame.get(11) ==
    "X") && (boardgame.get(16) == "X") && (boardgame.get(21) ==
    "X")){/[6],[11],[16],[21]

        parameter = 1;

        winner = "X";


    }

    else if((boardgame.get(2) == "X") && (boardgame.get(7) ==
    "X") && (boardgame.get(12) == "X") && (boardgame.get(17) ==
    "X")){/[2],[7],[12],[17]

        parameter = 1;

```

```

        winner = "X";

    }

    else if((boardgame.get(7) == "X") && (boardgame.get(12) ==
"X") && (boardgame.get(17) == "X") && (boardgame.get(22) ==
"X")){//[7],[12],[17],[22]

        parameter = 1;

        winner = "X";


    }

    else if((boardgame.get(3) == "X") && (boardgame.get(8) ==
"X") && (boardgame.get(13) == "X") && (boardgame.get(18) ==
"X")){//[3],[8],[13],[18]

        parameter = 1;

        winner = "X";


    }

    else if((boardgame.get(8) == "X") && (boardgame.get(13) ==
"X") && (boardgame.get(18) == "X") && (boardgame.get(23) ==
"X")){//[8],[13],[18],[23]

        parameter = 1;

        winner = "X";


    }

    else if((boardgame.get(4) == "X") && (boardgame.get(9) ==
"X") && (boardgame.get(14) == "X") && (boardgame.get(19) ==
"X")){//[4],[9],[14],[19]

```

```
        parameter = 1;  
        winner = "X";  
  
    }  
  
    else if((boardgame.get(9) == "X") && (boardgame.get(14) ==  
"X") && (boardgame.get(19) == "X") && (boardgame.get(24) ==  
"X")){//[9],[14],[19],[24]  
  
        parameter = 1;  
        winner = "X";  
  
    }  
  
    //diagonal  
  
    else if((boardgame.get(0) == "X") && (boardgame.get(6) ==  
"X") && (boardgame.get(12) == "X") && (boardgame.get(18) ==  
"X")){//[0],[6],[12],[18]  
  
        parameter = 1;  
        winner = "X";  
  
    }  
  
    else if((boardgame.get(6) == "X") && (boardgame.get(12) ==  
"X") && (boardgame.get(18) == "X") && (boardgame.get(24) ==  
"X")){//[6],[12],[18],[24]  
  
        parameter = 1;
```

```
    winner = "X";  
  
}  
  
else if((boardgame.get(4) == "X") && (boardgame.get(8) ==  
"X") && (boardgame.get(12) == "X") && (boardgame.get(16) ==  
"X")){//[4],[8],[12],[16]  
  
    parameter = 1;  
    winner = "X";  
  
}  
  
else if((boardgame.get(8) == "X") && (boardgame.get(12) ==  
"X") && (boardgame.get(16) == "X") && (boardgame.get(20) ==  
"X")){//[8],[12],[16],[20]  
  
    parameter = 1;  
    winner = "X";  
  
}  
  
//player 2  
  
//mendatar  
  
else if((boardgame.get(0) == "O") && (boardgame.get(1) ==  
"O") && (boardgame.get(2) == "O") && (boardgame.get(3) ==  
"O")){//[0],[1],[2],[3]  
  
    parameter = 1;  
    winner = "O";
```

```
    }

    else if((boardgame.get(1) == "O") && (boardgame.get(2) ==
    "O") && (boardgame.get(3) == "O") && (boardgame.get(4) ==
    "O")){//[1],[2],[3],[4]

        parameter = 1;

        winner = "O";


    }

    else if((boardgame.get(5) == "O") && (boardgame.get(6) ==
    "O") && (boardgame.get(7) == "O") && (boardgame.get(8) ==
    "O")){//[5],[6],[7],[8]

        parameter = 1;

        winner = "O";


    }

    else if((boardgame.get(6) == "O") && (boardgame.get(7) ==
    "O") && (boardgame.get(8) == "O") && (boardgame.get(9) ==
    "O")){//[6],[7],[8],[9]

        parameter = 1;

        winner = "O";


    }

    else if((boardgame.get(10) == "O") && (boardgame.get(11) ==
    "O") && (boardgame.get(12) == "O") && (boardgame.get(13) ==
    "O")){//[10],[11],[12],[13]
```

```

        parameter = 1;
        winner = "O";

    }

    else if((boardgame.get(11) == "O") && (boardgame.get(12)
== "O") && (boardgame.get(13) == "O") && (boardgame.get(14) ==
"O")){//[11],[12],[13],[14]

        parameter = 1;
        winner = "O";

    }

    else if((boardgame.get(15) == "O") && (boardgame.get(16)
== "O") && (boardgame.get(17) == "O") && (boardgame.get(18) ==
"O")){//[15],[16,[17],[18]

        parameter = 1;
        winner = "O";

    }

    else if((boardgame.get(16) == "O") && (boardgame.get(17)
== "O") && (boardgame.get(18) == "O") && (boardgame.get(19) ==
"O")){//[16],[17],[18],[19]

        parameter = 1;
        winner = "O";

    }

    else if((boardgame.get(20) == "O") && (boardgame.get(21)
== "O") && (boardgame.get(22) == "O") && (boardgame.get(23) ==

```

```

    "O")){//[20],[21],[22],[23]

        parameter = 1;
        winner = "O";

    }

    else if((boardgame.get(21) == "O") && (boardgame.get(22)
== "O") && (boardgame.get(23) == "O") && (boardgame.get(24) ==
"O")){//[21],[22],[23],[24]

        parameter = 1;
        winner = "O";

    }

    //menurun

    else if((boardgame.get(0) == "O") && (boardgame.get(5) ==
"O") && (boardgame.get(10) == "O") && (boardgame.get(15) ==
"O")){//[0],[5],[10],[15]

        parameter = 1;
        winner = "O";

    }

    else if((boardgame.get(5) == "O") && (boardgame.get(10) ==
"O") && (boardgame.get(15) == "O") && (boardgame.get(20) ==
"O")){//[5],[10],[15],[20]

```

```
        parameter = 1;  
  
        winner = "O";  
  
    }  
  
    else if((boardgame.get(1) == "O") && (boardgame.get(6) ==  
"O") && (boardgame.get(11) == "O") && (boardgame.get(16) ==  
"O")){//[1],[6],[11],[16]  
  
        parameter = 1;  
  
        winner = "O";  
  
    }  
  
    else if((boardgame.get(6) == "O") && (boardgame.get(11) ==  
"O") && (boardgame.get(16) == "O") && (boardgame.get(21) ==  
"O")){//[6],[11],[16],[21]  
  
        parameter = 1;  
  
        winner = "O";  
  
    }  
  
    else if((boardgame.get(2) == "O") && (boardgame.get(7) ==  
"O") && (boardgame.get(12) == "O") && (boardgame.get(17) ==  
"O")){//[2],[7],[12],[17]  
  
        parameter = 1;  
  
        winner = "O";  
  
    }  
  
    else if((boardgame.get(7) == "O") && (boardgame.get(12) ==  
"O") && (boardgame.get(17) == "O") && (boardgame.get(22) ==
```

```
"O")){//[7],[12],[17],[22]

    parameter = 1;
    winner = "O";

}

else if((boardgame.get(3) == "O") && (boardgame.get(8) ==
"O") && (boardgame.get(13) == "O") && (boardgame.get(18) ==
"O")){//[3],[8],[13],[18]

    parameter = 1;
    winner = "O";

}

else if((boardgame.get(8) == "O") && (boardgame.get(13) ==
"O") && (boardgame.get(18) == "O") && (boardgame.get(23) ==
"O")){//[8],[13],[18],[23]

    parameter = 1;
    winner = "O";

}

else if((boardgame.get(4) == "O") && (boardgame.get(9) ==
"O") && (boardgame.get(14) == "O") && (boardgame.get(19) ==
"O")){//[4],[9],[14],[19]

    parameter = 1;
    winner = "O";
```

```

    }

    else if((boardgame.get(9) == "O") && (boardgame.get(14) ==
    "O") && (boardgame.get(19) == "O") && (boardgame.get(24) ==
    "O")){//[9],[14],[19],[24]

        parameter = 1;

        winner = "O";

    }

    //diagonal

    else if((boardgame.get(0) == "O") && (boardgame.get(6) ==
    "O") && (boardgame.get(12) == "O") && (boardgame.get(18) ==
    "O")){//[0],[6],[12],[18]

        parameter = 1;

        winner = "O";

    }

    else if((boardgame.get(6) == "O") && (boardgame.get(12) ==
    "O") && (boardgame.get(18) == "O") && (boardgame.get(24) ==
    "O")){//[6],[12],[18],[24]

        parameter = 1;

        winner = "O";

    }

    else if((boardgame.get(4) == "O") && (boardgame.get(8) ==
    "O") && (boardgame.get(12) == "O") && (boardgame.get(16) ==
    "O"))
}

```

```
"O")){/[4],[8],[12],[16]

    parameter = 1;

    winner = "O";

}

else if((boardgame.get(8) == "O") && (boardgame.get(12) ==
"O") && (boardgame.get(16) == "O") && (boardgame.get(20) ==
"O")){/[8],[12],[16],[20]

    parameter = 1;

    winner = "O";

}

else{

    parameter = 0;

}

break;

default:

break;

}

return parameter;
```

```
1 public void clearBoard(int a){  
    int b = a*a;  
  
    for (int i = 0 ; i < b ; i++){  
        boardgame.set(i, String.valueOf(i+1));  
    }  
  
}  
  
public void score(String turn){  
  
    if(turn == "X"){  
        Xscore++;  
    }  
    else if(turn == "O"){  
        Oscore++;  
    }  
  
}  
4 public void viewXscore() {  
    System.out.println(Xscore);  
  
}  
  
public void viewOscore() {  
    System.out.println(Oscore);
```

```
}

}

}

class Piece{

    private char symbol;

    public void setsymbol (char symbol){

        this.symbol = symbol;

    }

    public char get_symbol(){

        return symbol;

    }

}
```

```
class O_piece extends Piece{  
  
    private char O_symbol;  
  
    public void setsymbolo (char O_symbol){  
        this.O_symbol = 'O';  
    }  
  
    public char get_symbolo(){  
        return O_symbol;  
    }  
}
```

```
class X_piece extends Piece{  
  
    private char X_symbol;  
  
    public void setsymbolx (char X_symbol){  
        this.X_symbol = 'X';  
    }  
  
    public char get_symbolx(){  
        return X_symbol;  
    }  
}
```

```
import java.io.BufferedReader;
import java.io.BufferedWriter;
import java.io.FileReader;
import java.io.FileWriter;
import java.io.IOException;
import java.time.LocalDateTime;
import java.time.format.DateTimeFormatter;
import java.util.*;

public class Main {

    public static void main(String[] args) {

        int parameter = 0;
        int gamedimension=0; int choose_board = 0;
        int min = 1;
        int num_of_player;
        String Turn = "X";
        String winner = null;
        int correct = 1;
```

```
int pos = 0;  
int play;  
Player1 player1 = new Player1();  
Player2 player2 = new Player2();  
  
Random randnum = new Random();  
  
Board board = new Board(gamedimension);  
  
Scanner inputs = new Scanner(System.in);  
  
while(parameter == 0){  
  
    winner = null;  
  
    System.out.println("\n\n===== Welcome to ANGER  
ISSUES TIC TAC TOE Game =====");  
    System.out.println("\nRules: ");  
    System.out.println("\n1. Each Player only have 1 chance each  
turn, so if you are input the wrong/choosen position, your turn will be  
skipped and you have nothing to put :P\n");  
    System.out.println("2. Player who make line first will be the  
winner\n");  
  
    play = player1.startgame(); //pake method di class player 1 untuk  
start game dan end game
```

```

if(play == 1){

    parameter = 0;
    //System.out.println("\n\nPlease choose your game mode: ");
    //System.out.println("\n1. Solo (vs Computer) 2. 2
Player\n");

    num_of_player = player1.selectgamemode(); //pake method di
class player 1 untuk milih jumlah pemain

    if(num_of_player == 1){//kalau cuman 1 maka nama player 2
bakal jadi bot

        player1.createname();
        player2.botname();
    }

    else{//kalau 2 orang maka akan masukin 1 1 namanya

        player1.createname();
        player2.createname();

    }

}

3
System.out.println("\n");

System.out.println("\n\n||| ANGER ISSUES TIC TAC TOE
LIST PLAYER |||"); //print list player

System.out.println("\nPLAYER 1: " + player1.get_name() +
| PLAYER 2: " + player2.get_name());

```

```
choose_board = player1.selectboard()//pake method di class
player 1 untuk milih ukuran papan

if(choose_board == 1){//kalo pilih 1 maka ukutan boardnya 3x3
    gamedimension = 3;
}
else if(choose_board == 2){//kalo pilih 2 maka ukutan boardnya
4x4
    gamedimension = 4;
}
else{//kalo pilih 3 maka ukutan boardnya 5x5
    gamedimension = 5;
}

board.set_gamedimensions(gamedimension);//set ukuran board

int max = gamedimension * gamedimension;//jumlah max petak
int bot_input = randnum.nextInt(max) + 1;
board.Initiation_displayBoard();//tampilan board awal

while (winner == null){

    switch (num_of_player) { //switch dari jumlah player
```

```

case 1://MODE VS BOT

while(correct == 1){

    if(Turn.equals("X")){
        try{
            System.out.println(player1.get_name() + "(" + Turn +
")" + " Turn: ");
            pos = inputs.nextInt();
            if(!(pos <= max && pos >= min)){
                System.out.println("Input invalid, enter another
position\n");
                continue;
            }
        }

        catch (InputMismatchException e){
            System.out.println("Input invalid, enter another
position\n");
            continue;
        }
    }

    else{

        try{
            System.out.println(player2.get_name() + "(" + Turn +
")" + " Turn: ");
            bot_input = randnum.nextInt(max) + 1;
        }
    }
}

```

```
pos = bot_input;

    if !(pos <= max && pos >= min)){
        System.out.println("");
        continue;
    }

}

catch (InputMismatchException e){
    System.out.println("");
    continue;
}

System.out.println(bot_input);

}

board.makemove(Turn, pos);

if(board.makemove(Turn, pos) == 1){
    correct = 1;
    Turn = Turn;
}

else{
    correct = 0;
}
```

```
        break;  
    }  
  
    if(Turn.equals("X")){  
  
        Turn = "O";  
  
    }  
    else{  
        Turn = "X";  
    }  
  
    board.checkwinner();  
  
    if(board.checkwinner() == 1){  
        winner = Turn;  
  
        if(Turn.equals("X")){  
  
            Turn = "O";  
            4  
            System.out.println("\n OUR YASS QUEEN IS:  
\\n" + player2.get_name());  
            board.score(Turn);  
  
        }  
    }
```

```
        else{

            Turn = "X";
4
            System.out.println("\n OUR YASS QUEEN IS: \n"
+ player1.get_name());

            board.score(Turn);

        }

        board.clearBoard(gamedimension);

    }

    else{

        winner = null;

    }

4
    System.out.println("\n");

break;

case 2: //MODE PVP

while(correct == 1){

    if(Turn.equals("X")){
```

```
try{
    System.out.println(player1.get_name() + "(" +
Turn + ")" + " Turn: ");

    pos = inputs.nextInt();

    if !(pos <= max && pos >= min)){
        System.out.println("Input invalid, enter another
position\n");
        continue;
    }

}

catch (InputMismatchException e){
    System.out.println("Input invalid, enter another
position\n");
    continue;
}

else{

try{
    System.out.println(player2.get_name() + "(" + Turn
+ ")" + " Turn: ");

    pos = inputs.nextInt();

    if !(pos <= max && pos >= min)){
        System.out.println("Input invalid, enter another
position\n");
        continue;
    }

}
```

```
        }

    catch (InputMismatchException e){

        System.out.println("Input invalid, enter another
position\n");

        continue;

    }

}

board.makemove(Turn, pos);

if(board.makemove(Turn, pos) == 1){

    correct = 1;

    Turn = Turn;

}

else{

    correct = 0;

}

break;

}

if(Turn.equals("X")){
```

```
Turn = "O";  
  
}  
else{  
    Turn = "X";  
}  
  
  
  
board.checkwinner();  
  
  
if(board.checkwinner() == 1){  
    winner = Turn;  
  
  
    if(Turn.equals("X")){  
  
        Turn = "O";  
        System.out.println("\nOUR YASS QUEEN IS: \n" +  
player2.get_name());  
        board.score(Turn);  
  
    }  
    else{  
        Turn = "X";  
        System.out.println("\nOUR YASS QUEEN IS: \n" +  
player1.get_name());  
        board.score(Turn);  
    }  
}
```

```
        board.clearBoard(gamedimension);

    }

    else{
        winner = null;
    }

    System.out.println("\n");

    break;

default:
    break;
}

}

}
```

```
//check score

else if(play == 2){

    System.out.println("\nScore:");
    System.out.println("\n" + player1.get_name() +":");
    board.viewXscore();
    System.out.println("\n" + player2.get_name() +":");
    board.viewOscore();

}

//check history

else if(play == 3){
    3
    try{

        FileReader fr = new FileReader("history.txt");
        BufferedReader br = new BufferedReader(fr);

        String history;
        if ((history = br.readLine()) == null){

            System.out.println("no matches found");
        }
        3
        while ((history = br.readLine()) != null){

            System.out.println(history + "/n");
        }
        br.close();
    }
}
```

```

} catch (IOException e){

    System.out.println("ERROR!");

}

//ketika player tidak mau main lagi

else{

    DateTimeFormatter dtf =
DateTimeFormatter.ofPattern("dd/MM/yyyy HH:mm");

    LocalDateTime now = LocalDateTime.now();

    try{

        FileWriter FW = new FileWriter("history.txt",true);

        BufferedWriter BW = new BufferedWriter(FW);

        BW.write("\nHystory Match " + dtf.format(now) + "\n");
        BW.write(player1.get_name() +":");
        BW.write(board.get_Xscore() + "\n");
        BW.write(player2.get_name() +":");
        BW.write(board.get_Oscore() + "\n");
        BW.newLine();
        BW.close();
    } catch (IOException e){

        System.out.println("ERROR!");

    }

System.out.println("\n===== OK THANKS FOR USING"

```

```

THIS APP WITH ANGER ISSUES ======");

    parameter = 1;

}

}

}

}

}

```

ii. Referensi

Tabel 3. 2 DAFTAR REFERENSI FINAL PROJECT OOP KELOMPOK 3

CITATION	LINK	ISI
<i>How to write a tic tac toe game in Java using classes.</i> Hello Code. (2021, October 1). Retrieved June 16, 2022, from https://www.hellocodeclub.com/tic-tac-toe-java-how-to-write-game-using-classes/	https://www.hellocodeclub.com/tic-tac-toe-java-how-to-write-game-using-classes/	5 How to Write a Tic Tac Toe Game in Java using classes

Tic-tac-toe game in Java. GeeksforGeeks. (2020, October 13). Retrieved June 16, 2022, from https://www.geeksforgeeks.org/tic-tac-toe-game-in-java/	http://www.geeksforgeeks.org/tic-tac-toe-game-in-java/	Tic-Tac-Toe Game in Java
Java array - javatpoint. www.javatpoint.com. (n.d.). Retrieved June 16, 2022, from https://www.javatpoint.com/array-in-java	https://www.javatpoint.com/array-in-java	Java Arrays
w3schools. Java constructors. (n.d.). Retrieved June 16, 2022, from https://www.w3schools.com/java/java_constructors.asp	https://www.w3schools.com/java/java_constructors.asp	Java Constructors
Randomly select items from a list in Java. GeeksforGeeks. (2021, June 17). Retrieved June 16, 2022, from https://www.geeksforgeeks.org/randomly-select-items-from-a-list-in-java/	https://www.geeksforgeeks.org/randomly-select-items-from-a-list-in-java/	Randomly select items from a List in Java
How to generate random number in Java - Javatpoint. www.javatpoint.com. (n.d.). Retrieved June 16, 2022, from https://www.javatpoint.com/how-to-generate-random-number-in-java	https://www.javatpoint.com/how-to-generate-random-number-in-java	How to Generate Random Number in Java

w3schools. Java create and write to files. (n.d.). Retrieved June 16, 2022, from https://www.w3schools.com/java/java_files_create.asp	https://www.w3schools.com/java/java_files_create.asp	Java Create and Write To Files
<i>How to generate random numbers in Java.</i> Educative. (n.d.). Retrieved June 16, 2022, from https://www.educative.io/answers/how-to-generate-random-numbers-in-java	https://www.educative.io/answers/how-to-generate-random-numbers-in-java	How to generate random numbers in Java
Mikahikaai. (2021, January 4). <i>Tictactoe/tictactoe.java at master · Mikahikaai/Tictactoe</i> . GitHub. Retrieved June 16, 2022, from https://github.com/mikahikaai/tictactoe/blob/master/src/main/java/tictactoe/TicTacToe.java	https://github.com/mikahikaai/tictactoe/blob/master/src/main/java/tictactoe/TicTacToe.java	mikahikaai tictactoe v0.1

iii. Daftar Hadir Diskusi Kelompok

ABSENSI DISKUSI KELOMPOK

No	Tanggal	Pembahasan Diskusi	Hadir
1.	30/05/2022	Use Case Diagram	<ul style="list-style-type: none"> ● 2440100172 - Afif Ramadhani Yafarayhan ● 2440059495 – Bismo Agung T.A.B ● 2440085442 – Evan raditya ● 2440040412 – Muhammad Rifqi Lutfi
2.	06/06/2022	Class Diagram	<ul style="list-style-type: none"> ● 2440100172 - Afif Ramadhani Yafarayhan ● 2440059495 – Bismo Agung T.A.B ● 2440085442 – Evan raditya ● 2440040412 – Muhammad Rifqi Lutfi
3.	12/06/2022	Membuat Class dalam Kodingan	<ul style="list-style-type: none"> ● 2440100172 - Afif Ramadhani Yafarayhan ● 2440059495 – Bismo Agung T.A.B ● 2440085442 – Evan raditya ● 2440040412 – Muhammad Rifqi Lutfi
4.	13/06/2022	Perbaikan Class dalam Kodingan	<ul style="list-style-type: none"> ● 2440100172 - Afif Ramadhani Yafarayhan ● 2440059495 – Bismo Agung T.A.B ● 2440085442 – Evan raditya

iv. PPT Presentasi

01



Final Project Tic Tac Toe Game

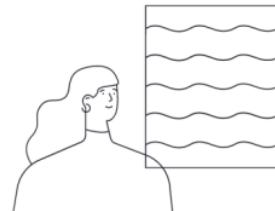
-ANGER ISSUES EDITION-

2440100172 - Afif Ramadhani Yafarayhan
2440059495 - Bismo Agung T.A.B
2440085442 - Evan raditya
2440040412 - Muhammad Rifqi Lutfi

02

OUTLINE

- Penjelasan Program
- Diagram
- Tampilan Program



03



Penjelasan Program

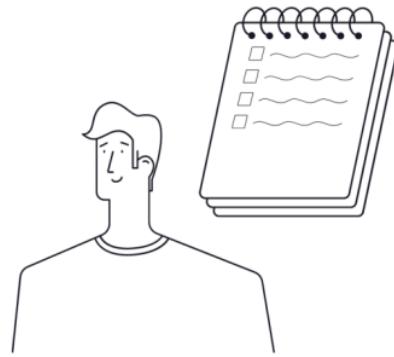
TIC TAC TOE

Program yang kami buat adalah permainan tic tac toe dimana pemain akan berlomba untuk meletakan simbol pada petak guna memenangkan permainan, pemain yang membuat garis baik secara vertikal, horizontal maupun diagonal akan memangkan permainan

Program permainan tic tac toe:

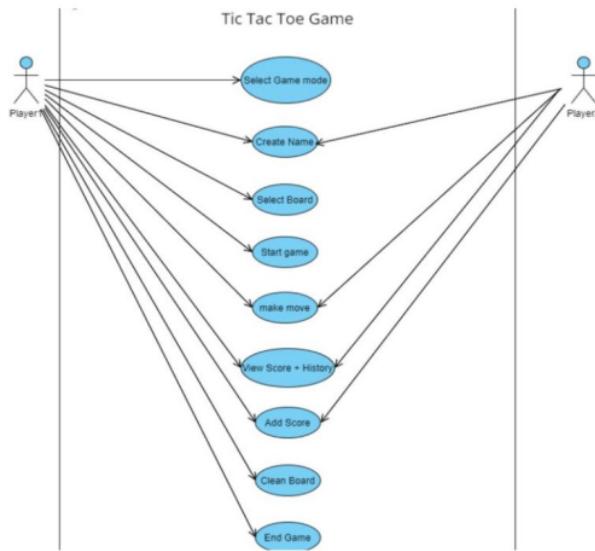
- 2 Game mode (Solo vs Computer & PVP)
- 3 Pilihan ukuran petak (3x3, 4x4, 5x5)
- 1 turn only

DIAGRAM



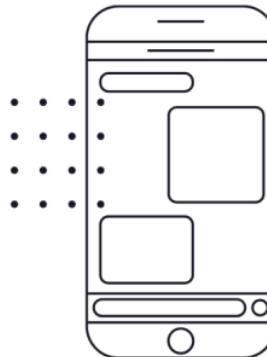
CLASS DIAGRAM

USE CASE



08

PROGRAM OVERVIEW



87

Main Menu

```
===== Welcome to ANGER ISSUES TIC TAC TOE Game =====  
Rules:  
1. Each Player only have 1 chance each turn, so if you are input the wrong/chooseen position, your turn will be skipped and you have nothing to put :P  
2. Player who make line first will be the winner  
  
Wanna Play ? 1. YASS    2. CHEECK SOME SCORE    3. CHEECK SLAYY BEfore    4. UMH NAURR  
  
Enter your choices:
```

Select Game Mode

```
===== Welcome to ANGER ISSUES TIC TAC TOE Game =====  
Rules:  
1. Each Player only have 1 chance each turn, so if you are input the wrong/chooseen position, your turn will be skipped and you have nothing to put :P  
2. Player who make line first will be the winner  
  
Wanna Play ? 1. YASS    2. CHEECK SOME SCORE    3. CHEECK SLAYY BEfore    4. UMH NAURR  
  
Enter your choices:  
1  
  
Please choose your game mode:  
1. Solo (vs Computer)    2. 2 Player  
Enter Your Choice:  
2  
ENTER YOUR PLAYER 1 NAME:  
Udin  
ENTER YOUR PLAYER 2 NAME:  
Banbang
```

Chooseboard and start game 3x3

```
||||| ANGER ISSUES TIC TAC TOE LIST PLAYER |||||
```

```
PLAYER 1: Udin | PLAYER 2: Bambang
```

```
Please Select Board:
```

```
1. 3x3 2. 4x4 3. 5x5:
```

```
Enter your Choice:
```

```
1
```

-	-	-
1	2	3
-	-	-
4	5	6
-	-	-
7	8	9
-	-	-

Chooseboard and start game 4x4

```
||||| ANGER ISSUES TIC TAC TOE LIST PLAYER |||||
```

```
PLAYER 1: Udin | PLAYER 2: Bambang
```

```
Please Select Board:
```

```
1. 3x3 2. 4x4 3. 5x5:
```

```
Enter your Choice:
```

```
2
```

-	-	-	-
1	2	3	4
-	-	-	-
5	6	7	8
-	-	-	-
9	1	2	3
-	-	-	-
4	5	6	7
-	-	-	-

Chooseboard and start game 5x5

```
||||| ANGER ISSUES TIC TAC TOE LIST PLAYER |||||
PLAYER 1: Udin    |    PLAYER 2: Bambang
Please Select Board:
1. 3x3  2. 4x4  3. 5x5:
Enter your Choice:
3
| - | - | - | - | - |
| 1 | 2 | 3 | 4 | 5 |
| - | - | - | - | - |
| 6 | 7 | 8 | 9 | 10 |
| - | - | - | - | - |
| 11| 12| 13| 14| 15 |
| - | - | - | - | - |
| 16| 8 | 9 | 10 | 11 |
| - | - | - | - | - |
| 12| 13| 14| 15 | 16 |
```

Player turn

```
Udin(X) Turn:
1
| - | - | - |
| X | 2 | 3 |
| - | - | - |
| 4 | 5 | 6 |
| - | - | - |
| 7 | 8 | 9 |
| - | - | - |

Position selected

Bambang(O) Turn:
5
| - | - | - |
| X | 2 | 3 |
| - | - | - |
| 4 | 0 | 6 |
| - | - | - |
| 7 | 8 | 9 |
| - | - | - |

Position selected
```

Game Logic ketika memilih posisi yang sama

```
Udin(X) Turn:  
2  
| - | - | - |  
| X | X | 3 |  
| - | - | - |  
| 4 | 0 | 6 |  
| - | - | - |  
| 7 | 8 | 9 |  
| - | - | - |  
  
Position selected  
  
Bambang(O) Turn:  
5  
  
Position selected  
  
Position selected  
  
Udin(X) Turn:  
3  
| - | - | - |  
| X | X | X |  
| - | - | - |  
| 4 | 0 | 6 |  
| - | - | - |  
| 7 | 8 | 9 |  
| - | - | - |  
  
Position selected
```

Win Logic Horizontal

```
Udin(X) Turn:  
11  
Input invalid, enter another position  
  
Udin(X) Turn:  
3  
| - | - | - |  
| X | X | X |  
| - | - | - |  
| 4 | 0 | 6 |  
| - | - | - |  
| 7 | 8 | 9 |  
| - | - | - |  
  
Position selected
```

Game Logic ketika input angka melebihi ukuran board

```
Udin(X) Turn:  
11  
Input invalid, enter another position  
  
Udin(X) Turn:  
3  
| - | - | - |  
| X | X | X |  
| - | - | - |  
| 4 | 0 | 6 |  
| - | - | - |  
| 7 | 8 | 9 |  
| - | - | - |  
  
Position selected
```

Game Logic ketika input angka melebihi ukuran board

```
Udin(X) Turn:  
11  
Input invalid, enter another position  
  
Udin(X) Turn:  
3  
| - | - | - |  
| X | X | X |  
| - | - | - |  
| 4 | 0 | 6 |  
| - | - | - |  
| 7 | 8 | 9 |  
| - | - | - |  
  
Position selected
```

Game Logic ketika input angka melebihi ukuran board

```
Udin(X) Turn:  
11  
Input invalid, enter another position  
  
Udin(X) Turn:  
3  
| - | - | - |  
| X | X | X |  
| - | - | - |  
| 4 | 0 | 6 |  
| - | - | - |  
| 7 | 8 | 9 |  
| - | - | - |  
  
Position selected
```

Win logic

```
Udin(X) Turn:  
3  
| - | - | - |  
| X | X | X |  
| - | - | - |  
| 4 | 0 | 6 |  
| - | - | - |  
| 7 | 8 | 9 |  
| - | - | - |  
  
Position selected  
  
OUR YASS QUEEN IS:  
Udin
```

3x3 Win logic

```
Ramlong(O) Turn:  
11  
| - | - | - | - |  
| 0 | 2 | 3 | X | |
|---|---|---|---|---|
| X | 0 | 7 | X |  
| - | - | 0 | - |  
| 9 | 10 | - | 12 |  
| - | 13 | 14 | 15 | 16 |  
| - | - | - | - |  
  
Position selected  
  
OUR YASS QUEEN IS:  
Ramlong
```

4x4 Win logic

```

Udin(X) Turn:
16
- | - | - | - | -
X | 0 | 0 | 4 | 5 |
- | - | - | - | -
X | 7 | 8 | 9 | 10 |
- | - | - | - | -
X | 12 | 13 | 14 | 15 |
- | - | - | - | -
X | 17 | 18 | 19 | 20 |
- | - | - | - | -
0 | 22 | 23 | 24 | 25 |

Position selected

OUR YASS QUEEN IS:
Udin

```

5x5 Win logic

Check Score

```

===== Welcome to ANGER ISSUES TIC TAC TOE Game =====

Rules:
1. Each Player only have 1 chance each turn, so if you are input the wrong/chooseen position, your turn will be skipped and you have nothing to
put :P
2. Player who make line first will be the winner

Wanna Play ? 1. YASS    2. CHEEK SOME SCORE    3. CHEEK SLAYY BEFOR    4. UMH NAURR

Enter your choices:
2

Score:
Udin:
2

Bambang:
1

```

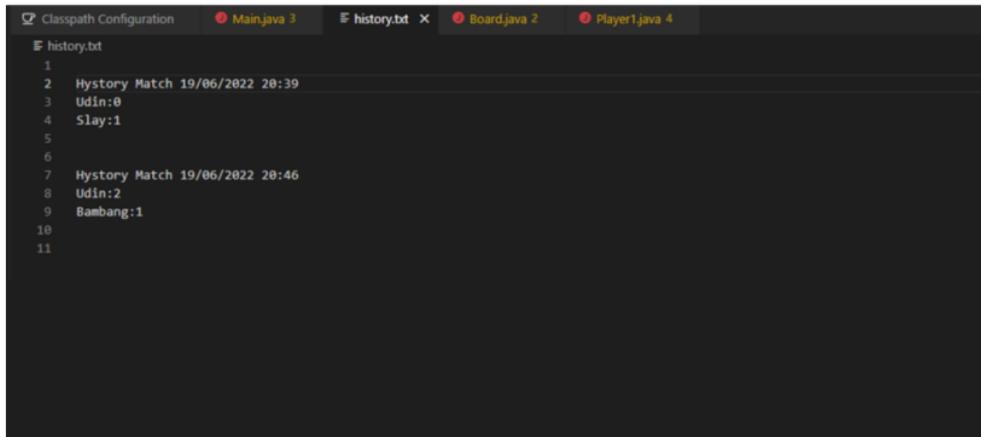
Check History

```
===== Welcome to ANGER ISSUES TIC TAC TOE Game =====  
Rules:  
1. Each Player only have 1 chance each turn, so if you are input the wrong/chooseen position, your turn will be skipped and you have nothing to put :P  
2. Player who make line first will be the winner  
  
Wanna Play ? 1. YASS    2. CHEECK SOME SCORE    3. CHEECK SLAYY BEfore    4. UMH NAURR  
  
Enter your choices:  
3  
Hystory Match 19/06/2022 20:39/n  
Udin:0/n  
Slay:1/n  
/n
```

Exit game

```
===== Welcome to ANGER ISSUES TIC TAC TOE Game =====  
Rules:  
1. Each Player only have 1 chance each turn, so if you are input the wrong/chooseen position, your turn will be skipped and you have nothing to put :P  
2. Player who make line first will be the winner  
  
Wanna Play ? 1. YASS    2. CHEECK SOME SCORE    3. CHEECK SLAYY BEFORE    4. UMH NAURR  
  
Enter your choices:  
4  
===== OK THANKS FOR USING THIS APP WITH ANGER ISSUES =====  
PS D:\Tugas Binus\SMT 4\OOP\TUGAS\LAB> █
```

Check Update Gameplay History



A screenshot of a terminal window titled "history.txt". The window shows two entries from a file named "history.txt". The first entry is dated 19/06/2022 at 20:39, showing a match between "Udin" (0) and "Slay" (1). The second entry is dated 19/06/2022 at 20:46, showing a match between "Udin" (2) and "Bambang" (1). The terminal window also lists other files: "Classpath Configuration", "Main.java 3", "Board.java 2", and "Player1.java 4".

```
1
2 Hystory Match 19/06/2022 20:39
3 Udin:0
4 Slay:1
5
6
7 Hystory Match 19/06/2022 20:46
8 Udin:2
9 Bambang:1
10
11
```

v. Hasil Pemeriksaan Turn it in

TIC TAC TOE GAMES

ORIGINALITY REPORT



PRIMARY SOURCES

1	"Beginning Java® Programming", Wiley, 2012 Publication	2%
2	Submitted to The Robert Gordon University Student Paper	1 %
3	www.howtobuildsoftware.com Internet Source	1 %
4	archive.org Internet Source	1 %
5	javaproficiency.blogspot.com Internet Source	1 %

Exclude quotes Off
Exclude bibliography Off

Exclude matches < 1%