

The Dice Game



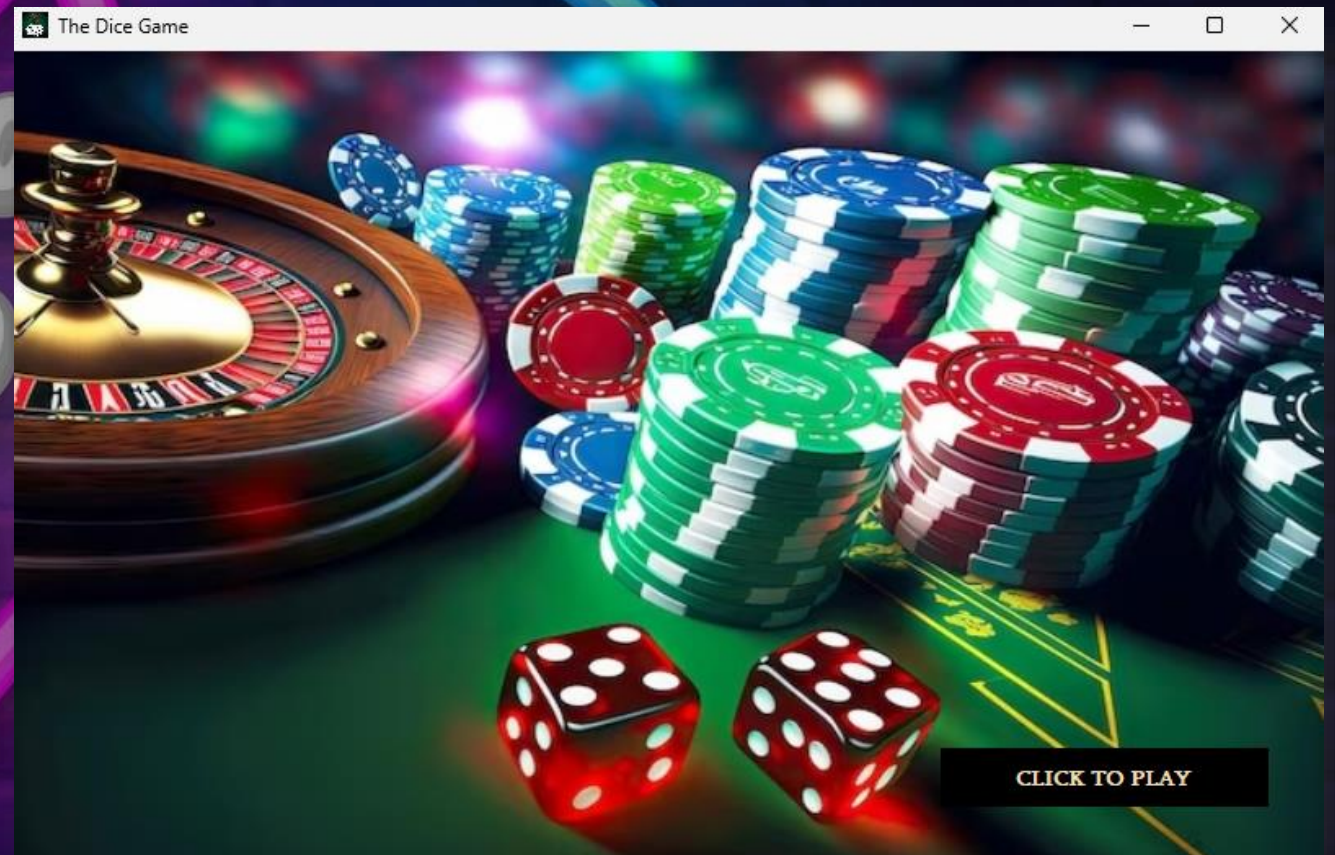
GABANELLA LEONARDO

MIZZON ALESSANDRO

Home screen

To proceed press "CLICK TO PLAY".

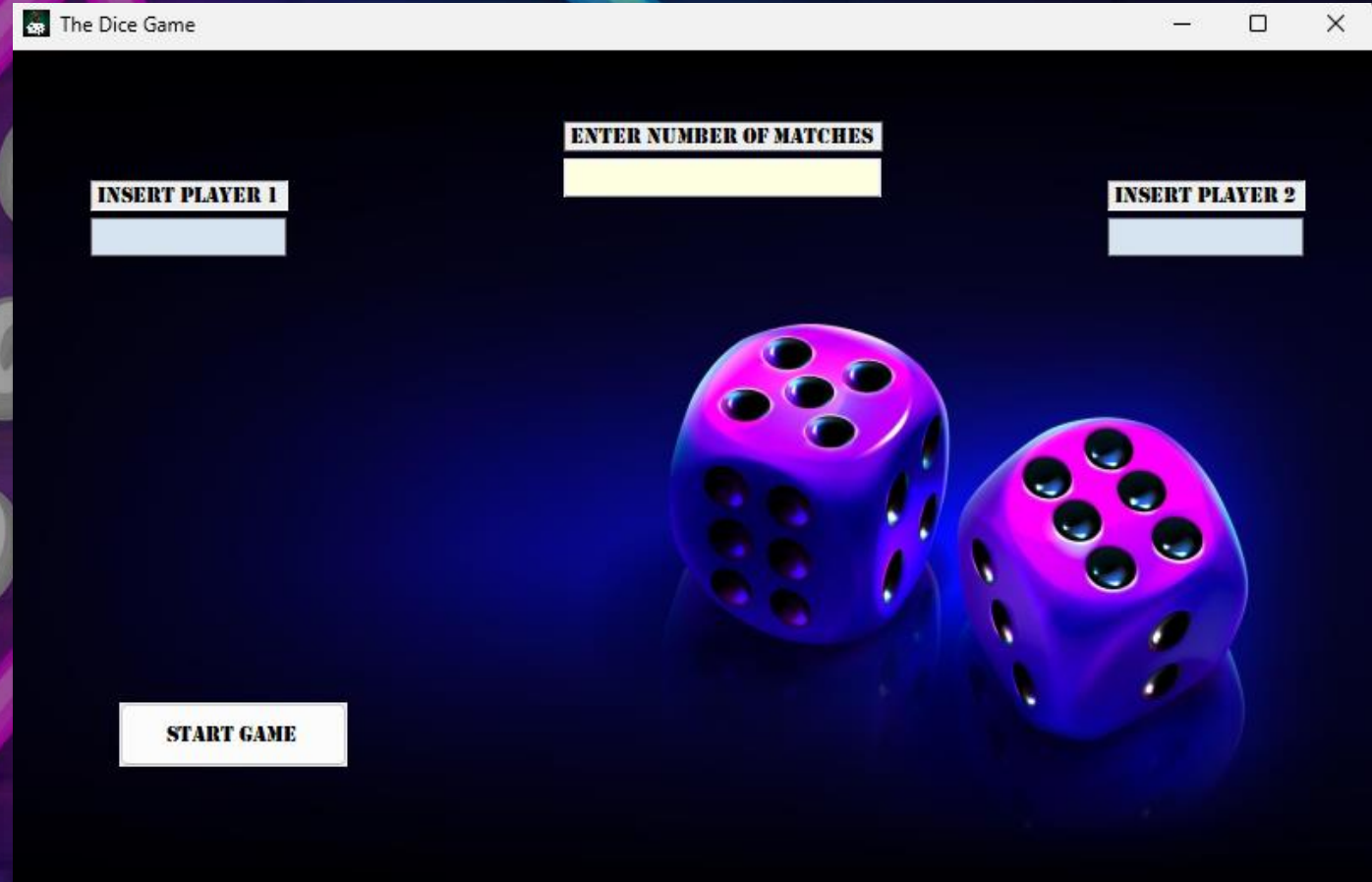
The transition from one screen to another during the game is based on changing the visibility of the components.



Login screen

Once you have filled in the required fields, click the "START GAME" button to start the game.

In case you don't fill all the required fields or insert a wrong value, an error message will appear on the screen.

The screenshot shows a web browser window titled "The Dice Game". The login interface has a dark background with a glowing blue dice illustration. It contains three input fields: "INSERT PLAYER 1" (light blue), "ENTER NUMBER OF MATCHES" (yellow), and "INSERT PLAYER 2" (light blue). A red "START GAME" button is at the bottom left. The browser window includes standard OS controls (minimize, maximize, close) in the top right corner.

The Dice Game

INSERT PLAYER 1

ENTER NUMBER OF MATCHES

INSERT PLAYER 2

START GAME

Roll dice

This screen represents the actual game.

Press "ROLL DICE" to throw the players' dice; the player with the highest dice face obtains a point and in case of the same face one point will be assigned to each player.



How to reset the game

This button allows to reset the current game, unless it is already finished.



Who is the winner?

When the number of round reaches zero, the player who has won the most number of rounds wins the game; if the players have won the same number of rounds, the message "TIE" will appear on the screen.



UML diagrams

Gara

```
- g1;
- g2;
- numeroPartite;
- buffer;
- winner;
+ Gara(string _g1, string _g2, int _numeroPartite)
+ FineGara()
+ Round()
+ GameWin()
+ ResetGame()
+ G1GetName()
+ G2GetName()
+ G1GetPoints()
+ G2GetPoints()
+ G1GetNum()
+ G2GetNum()
<<Proprietà>> + GetPartiteRimanenti: string
<<Proprietà>> + GetWinner: int
```

Giocatore

```
- nome;
- punteggioCorrente;
- dado;
+ Giocatore(string _nome)
+ IncreasePoints()
+ LanciaDado()
+ ResettaPunteggio()
+ GetNumero()
+ operator >(Giocatore g1, Giocatore g2)
<<Proprietà>> + GetName: string
<<Proprietà>> + GetDado: Dado
<<Proprietà>> + GetPoints: int
```

Dado

```
- randomNumber;
- num;
- facce;
+ Dado(int _facce)
+ LancioDado()
+ operator > (Dado d1, Dado d2)
+ operator < (Dado d1, Dado d2)
<<Proprietà>> + GetNum : int
```


How we've devided the work

Gabanella	Mizzon
Class "Dado"	Class "Gara"
Class "Giocatore"	Form
Power Point	UML diagrams