

Distributed Battleships - Manual

Annika Laumets, Andre Tättar, Viktoria Plemakova, Sander Tars

December 29, 2016

Setup process and dependencies

Firstly, the (middleware) dependencies are pika python library and RabbitMQ. They have to be installed. Pika can be installed using *pip*. If using Linux, RabbitMQ can be installed using *apt-get*.

To run the application, first server needs to be run. To do this, from the command line run the file "server.py" from the *server* directory. That should set up the server.

Once the server is running, client user interfaces can be opened by running "gui.py" from the command line. The file "gui.py" is located in the *client* directory.

User interface manual

The user interface is explained by the following screenshots and their descriptions.

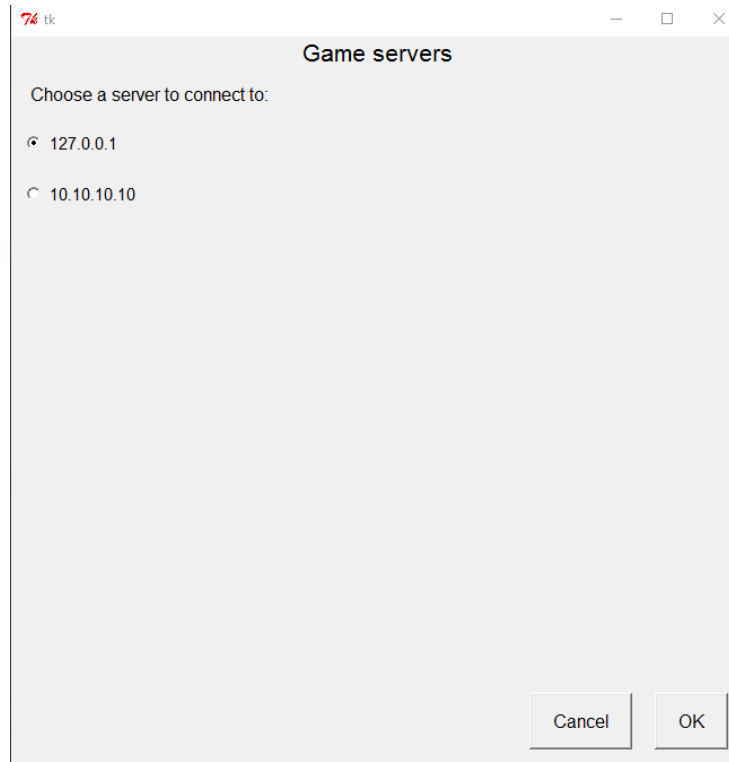


Figure 1: Opening page of the user interface. A list of available servers is displayed for connecting. OK button proceeds to the name choosing. Cancel button closes the application.

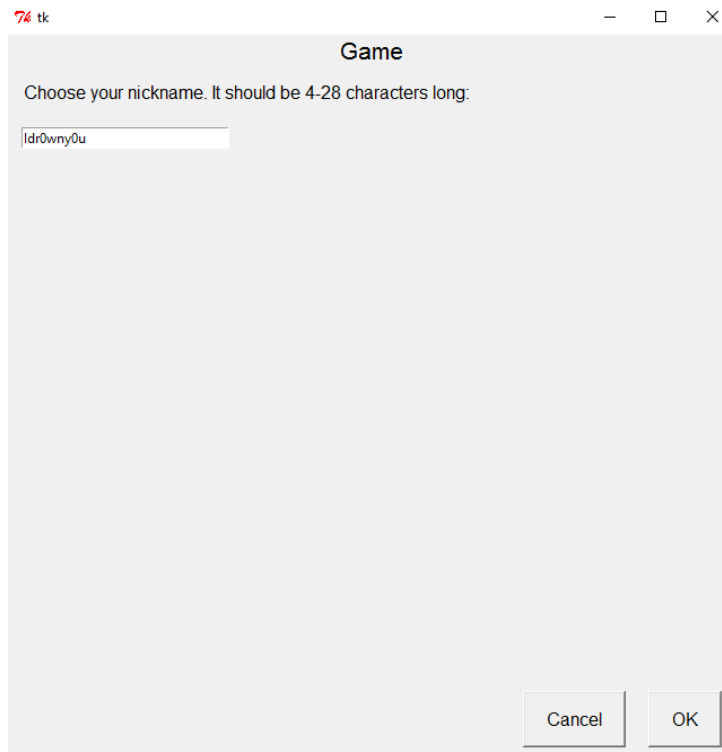


Figure 2: Nickname choosing page for user. A nickname can be entered in the text box. OK button sends the nickname and proceeds to the game choosing if nickname has been approved by the server. Cancel button goes back to server choosing view. If the nickname is unsuitable, a pop-up is presented.

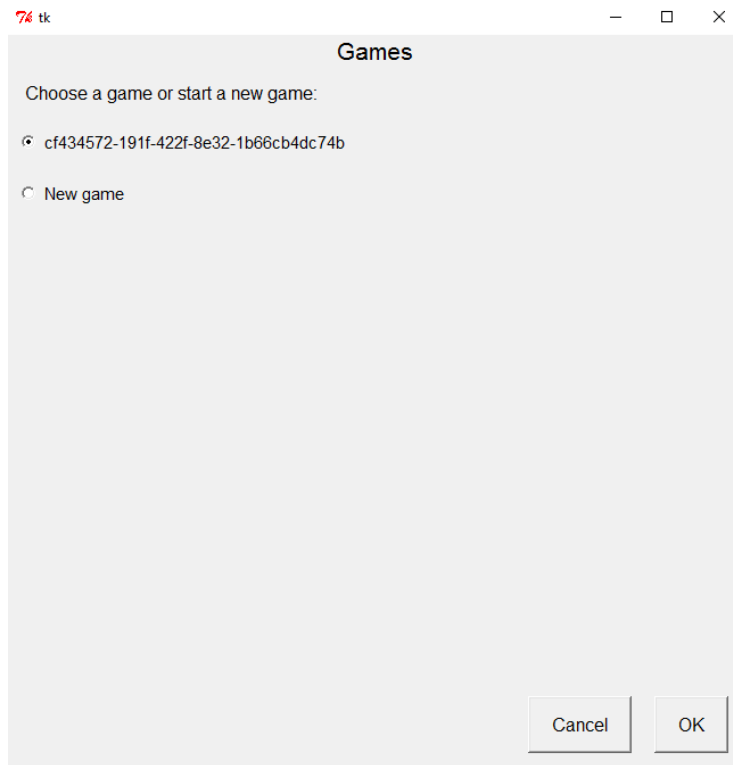


Figure 3: Game can be chosen from a displayed list or a new game can be created. OK proceeds to the next view, Cancel returns to the server choosing view.

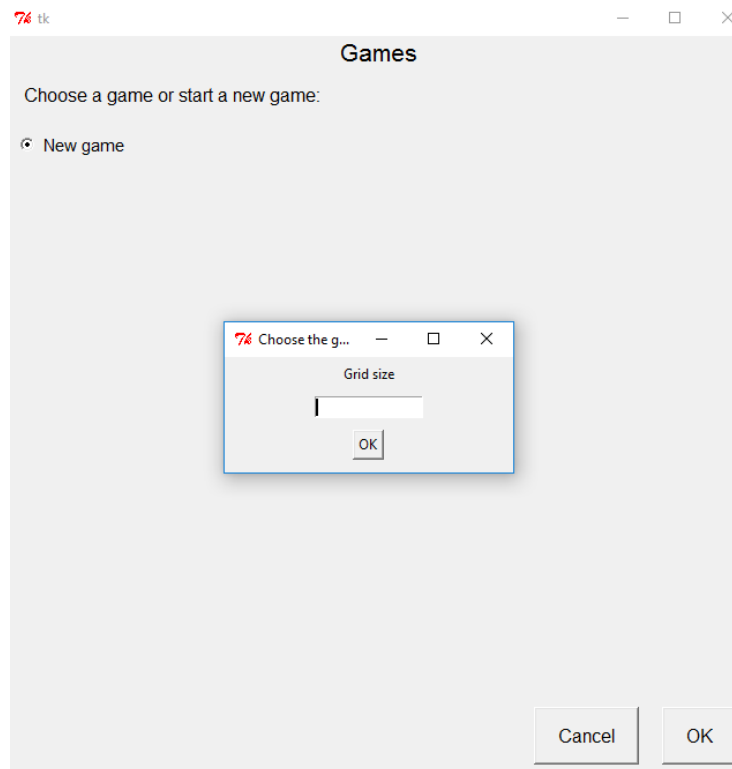


Figure 4: In case a new game is created, user is prompted with a pop-up to choose the grid size. OK button proceeds to the ship placement view.

tk

Place your ships

| | 0 | 1 | 2 | 3 | 4 |
|---|---|---|---|---|---|
| A | | | | | |
| B | | | | | |
| C | | | | | |
| D | | | | | |
| E | | | | | |

Starting point (A,0): Direction:

Carrier - size 5 a,0 Horizontal

Battleship - size 4 b,0 Horizontal

Submarine - size 3 c,0 Vertical

Cruiser - size 3 c,4 Vertical

Destroyer - size 2 e,0 Horizontal

Cancel OK

Figure 5: Ship placement view of the UI. User can choose the "root" place of the ship by typing it in the textbox and the direction of the ships by choosing from the dropdown menu. OK button proceeds to the game "lobby", Cancel button returns to the game choosing view.

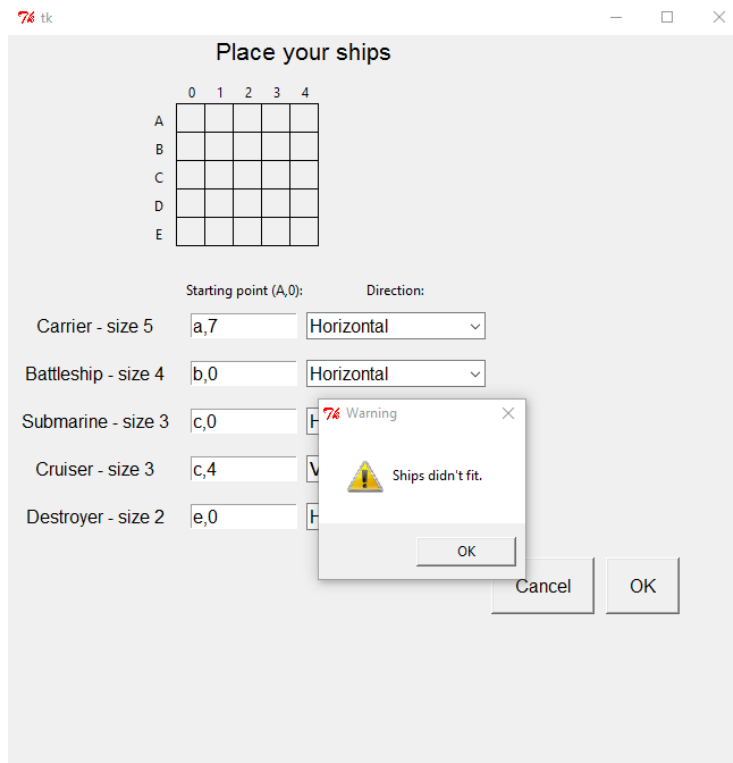


Figure 6: In case ship placement is unfit, user is notified with a pop-up.

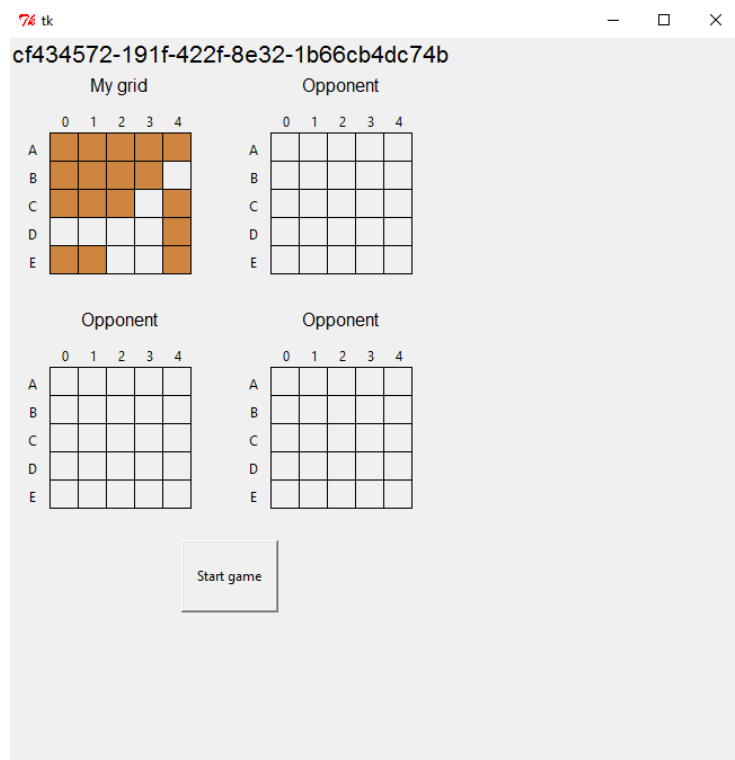


Figure 7: Once ships are accepted, the user is directed to a wait-page. Master user has Start game button.

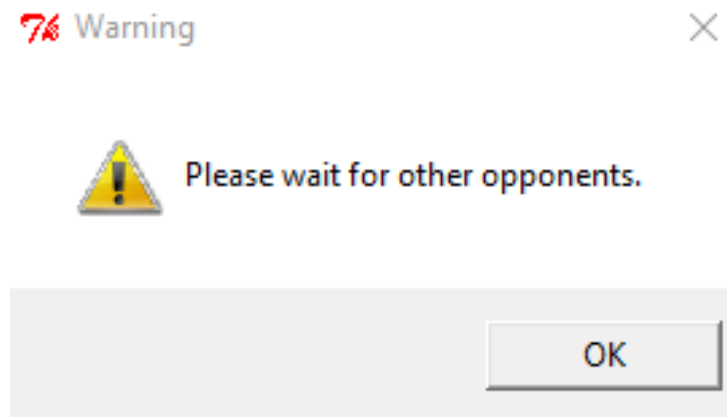


Figure 8: If master starts game without other players, he is notified with a pop-up.

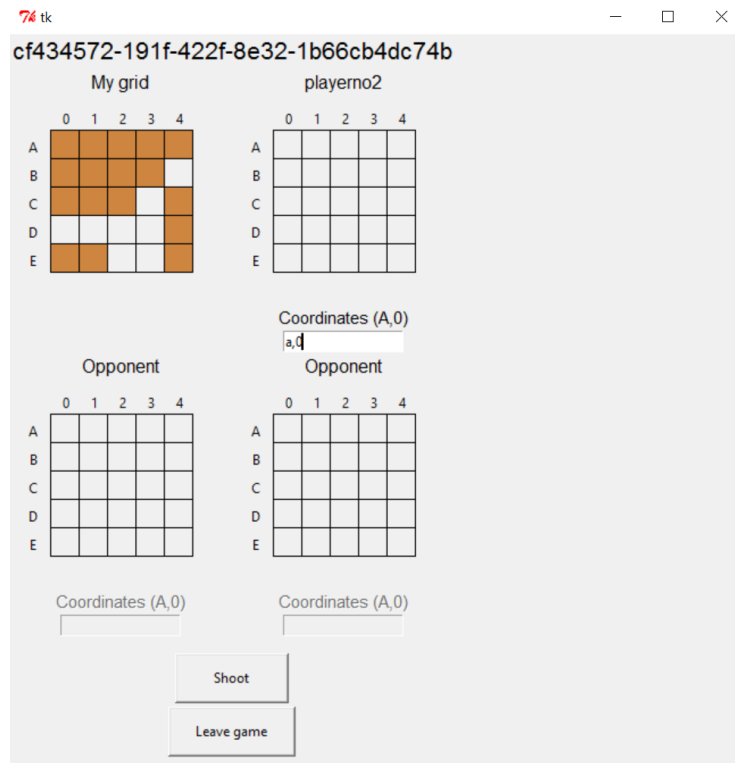


Figure 9: If the game has been started, the player whose turn it is can place their shots against the rest of the players whose names are displayed above the grids. If there are less than four player, excessive fields are disabled. Shoot button delivers the shots. Leave game button causes pop-up where user is asked whether they are sure they want to quit. If they are, then user leaves the game.

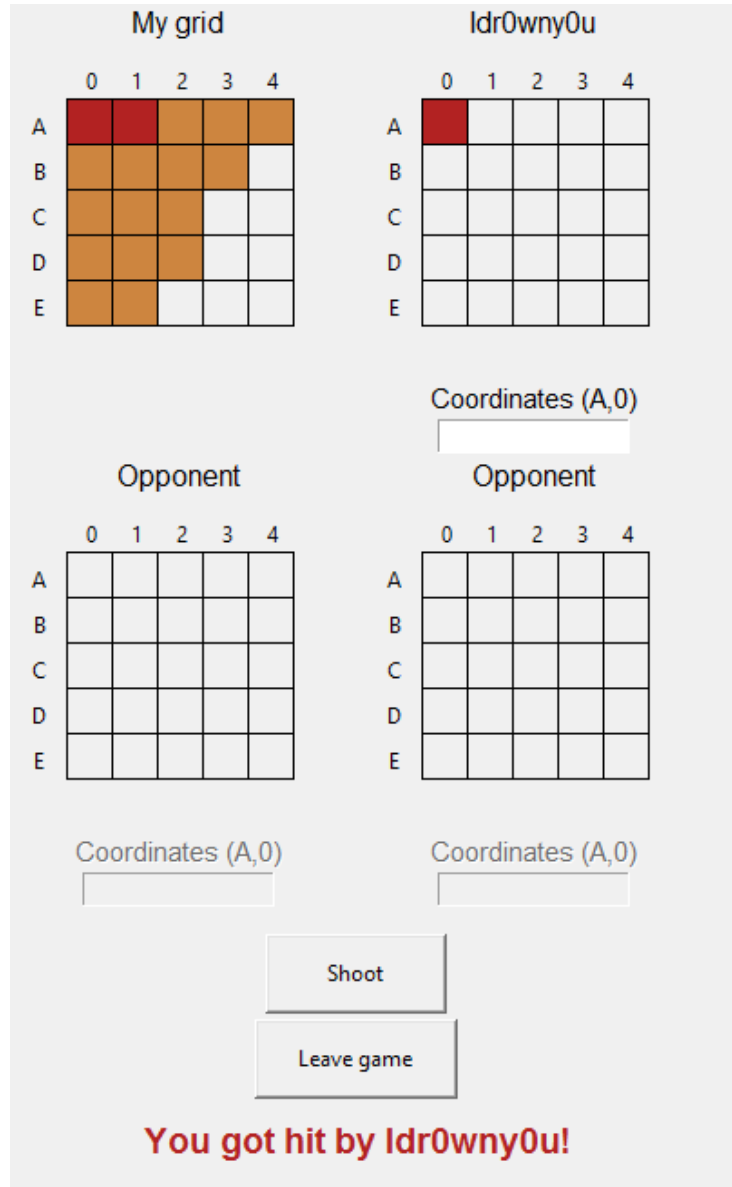


Figure 10: If a player is hit, the corresponding square is coloured red, missed shots are coloured as blue. Also a text is displayed for 3.5 seconds explaining who placed the scoring hit. Multiple hits to one square are allowed, but useless. Falsely placed hit is not accepted as a shot.

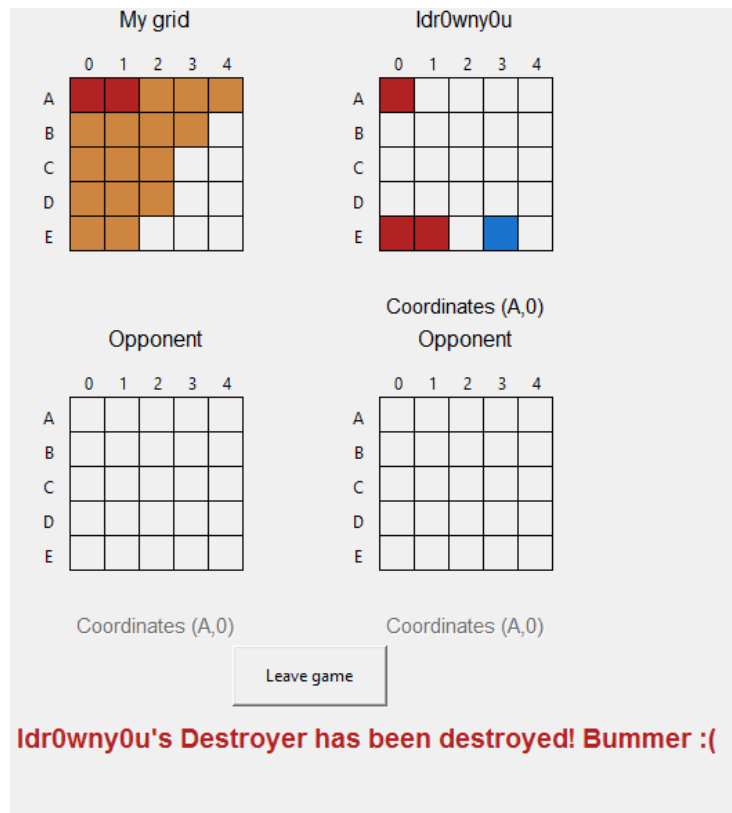


Figure 11: If a player's ship is drowned, the message and placement of the ship is broadcast to everyone.

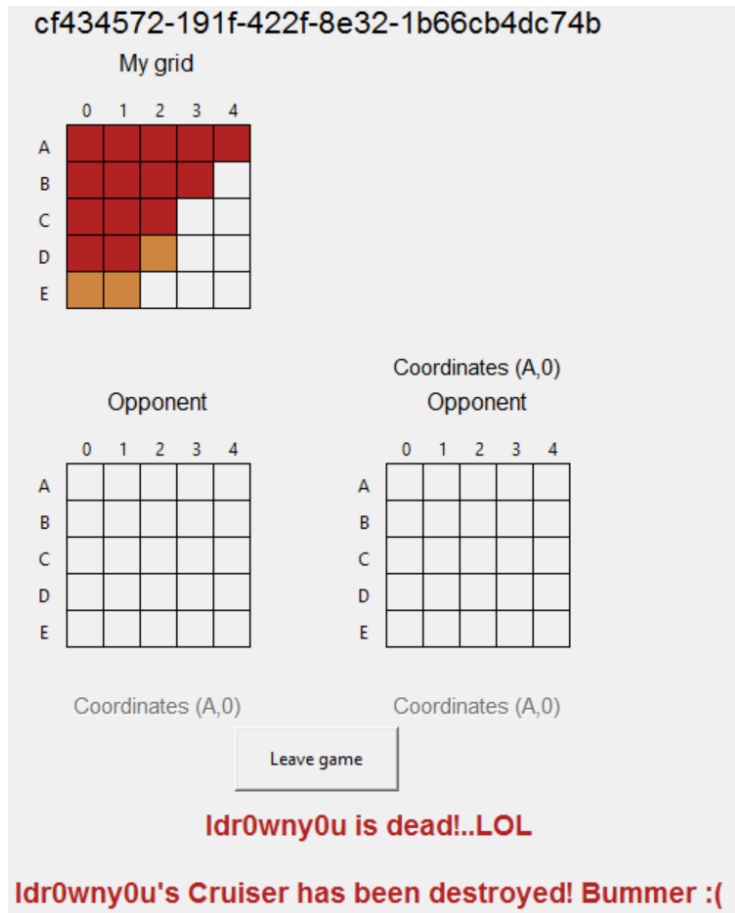


Figure 12: If a player loses, it is broadcast to everyone in a polite manner, and their grid is deleted to avoid confusion. If no more players remain, only Leave game is an option.

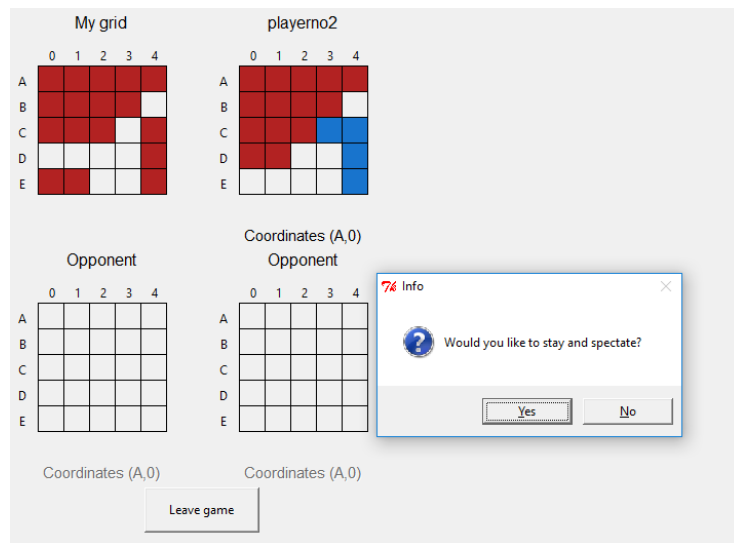


Figure 13: If a player loses, they are presented with a pop-up where they can either choose to remain spectating or leave the game to be directed to game choosing view.