In this workshop, you are responsible to create a Tic- Tac- Toe game. you should developed a program for a tic-tac-toe game that enables two players to play the game on the same machine. You should develop a distributed tic-tac-toe game using multithreads and networking with socket streams. A distributed tic-tac-toe game enables users to play on different machines from anywhere on the Internet.

Player 1	Server	Player 2
1. Initialize user interface.	Create a server socket.	1. Initialize user interface.
2. Request connection to the server and learn which token to use from the server.	Accept connection from the first player and notify the player who is Player 1 with token X.	
	Accept connection from the second player and notify the player who is Player 2 with token O. Start a thread for the session.	2. Request connection to the server and learn which token to use from the server.
	Handle a session:	
3. Get the start signal from the server.	1. Tell Player 1 to start.	
4. Wait for the player to mark a cell, send the cell's row and column index to the server.	2. Receive row and column of the selected cell from Player 1.	3. Receive status from the server.
5. Receive status from the server.	3. Determine the game status (WIN, DRAW, CONTINUE). If Player 1 wins, or draws, send the status (PLAYER1_WON, DRAW) to both players and send Player 1's move to Player 2. Exit.	4. If WIN, display the winner. If Player 1 wins, receive Player 1's last move, and break the loop.
6. If WIN, display the winner; if Player 2 wins, receive the last move from Player 2. Break the loop.	4. If CONTINUE, notify Player 2 to take the turn, and send Player 1's newly selected row and column index to Player 2.	5. If DRAW, display *game is over*, and receive Player 1's last move, and break the loop.
7. If DRAW, display game is over; break the loop.	5. Receive row and column of the selected cell from Player 2.	6. If CONTINUE, receive Player 1's selected row and index and mark the cell for Player 1.
	6. If Player 2 wins, send the status (PLAYER2_WON) to both players, and send Player 2's move to Player 1. Exit.	
8. If CONTINUE, receive Player 2's selected row and column index and mark the cell for Player 2.		7. Wait for the player to move, and send the selected row and column to the server.
	7. If CONTINUE, send the status, and send Player 2's newly selected row and column index to Player 1.	