Draft plan

How the game works: We have an interface class that contains play function, and we implements the function differently based on the player class type (computer, player, network).
In main game class, the constructor takes two objects of type interface and starts to handle the game and the turns. In player vs computer: it let you choose between 3 levels: (1,2,3).
level 1 the computer implements play function which sends 3 arguments (board, available moves, current stone) and return a random move from 1:9.
At level 2: computer return the move after calculating the best case scenario from alpha-beta with max depth 2.
At level 3: we make the max depth 8 to make it smarter.

At player vs player on same machine: one machine plays and waits a request stone from the second one. And both use play function to return the move.

At player vs player on different network: one player initiate the game and writes the ip of the second player. Or waits for a connection for 3 minutes (if some one writes his ip the game starts, if not failed to connect will appear to him). Both players implements the play function as well and return their move.

When a game start it randomly selects who will go first. And check if the returned move from this player is valid if so it places it on the board else would request the player to make a valid move. Game win: when there's is three stone of same type on board it finishes the game and return the board and player. If board has 9 stones and no one made a 3 winning moves it returns the board with tie.