1. What game are you making?
   1. It’ll be a connect 4 game where any player can drop in/drop out whenever they need to.
      1. If player 1 places a piece then leaves, someone can join and take their spot.
   2. There’s also going to be an option to send messages to a chat log that’s on every Client.
2. What RPC’s will you need?
   1. Client -> Server (Logic here rests on the Server)
      1. CallStartTurnOnClients(bool passTurn = true)
         1. Changes who’s turn it is and calls the StartTurn RPC on all clients
         2. Allows for you to force whose next turn it is
      2. TransmitMessage(string, name, string message);
         1. Sends a chat message to the server which in turn sends it to all the clients.
         2. Allow sends the name of the person that sent the message.
      3. PlayerIsReady();
         1. Tells the server when a player is ready.
         2. Gets called as soon as the player connects.
         3. Game starts when both players are connected and ready.
      4. SetBoardState(int row, int column, int type);
         1. Allows the server to remember and keep track of what’s on the board.
         2. Sends a message to all clients
      5. SendServerMessageToClients(string message)
         1. Get called in order to tell each client who’s turn it is and
   2. Server -> Client (Logic here rests on the Client)
      1. SetPieceID(int type);
         1. Sets what piece each player will play as.
         2. Will be represented as either ‘2’, ‘1’, or ‘0’ internally but externally will only be a represented as a color.
         3. Server will assign a random piece to a player and the server should remember what has already been assigned.
      2. UpdateName(string name);
         1. Sets a name on a client.
      3. UpdateChatLog(string name, string message);
         1. Displays a message along with the name of the person who sent it.
      4. SetBoardStateData (int y, int x, int type);
         1. Sets a piece on the board to the given type
         2. Get called from the server.
      5. StartTurn(int pieceType);
         1. Gets called to another player when the previous player passes a turn.
         2. Clients determines if it’s their turn from the pieceType that’s given.
      6. DeclareWinner(int pieceID);
         1. Get called from the server when it determines that someone has won the game.
         2. Enables particles to show who won using the pieceID that’s passed in.
      7. UpdateServerStatus(string message);
         1. Used to update text on every client’s screen.
         2. Would include message like:
            1. “Waiting for Red’s turn…”
            2. “Waiting for a Second Player to Join…”
            3. “Yellow Wins!”
3. What role will your server have for the game?
   1. The server should keep track of which pieces are on the board.
   2. Should remember which team already has a player assigned to it. (just bools)
   3. If 1 out of 2 players leave the game should continue up to a point.
   4. When a player joins it should tell the client what piece it is and update the board to show the pieces that have been placed already.