# STUFF THAT IS WRONG

## objPlayer Object

This object will be responsible for holding information about the player and his/her associated pieces on the board. Including things such as their colour, Username, pieces (an array of 4 individual pieces), and the current state of the players Home and Start locations.

## objTHEgame Object

This object is responsible for all the game functionality that should happen on any given players turn, This includes things such as rolling the die, Selecting a piece to move, moving said piece, and checking to see if a player has won the game.

## Trouble Object

This object is responsible for instantiating the objTHEGame object, assigning a player a colour, as well as figuring out how many players are in the game. This object is also responsible for switching the active player. This is the object that will be sent as an envelope to the other players.

## Major Requirements

**Moving:**

Getting a 6 from the roll should let you take a piece out of the start and place it at the beginning place on the board. When you roll the die, you should be able to select a piece and it will move that many spaces.

**Landing on an opponent/your own:**

During the move stage of your turn, if you selected a piece that would land on another piece at the end of its move, you must check if it’s an opponent or your own, if it is your own piece then you can’t move there, if it is an opponent’s, then you take their place on the board and their piece is put back into start.

**Board GUI:**

The game will include an interactive board that shows where all the pieces are. The board will look similar but not exactly like the game trouble. OBVIOUSLY.

**2-4 Players:**

The game should be able to support 2-4 players at the same time with only 1 current player ever being able to move his pieces. When a player finishes there turn the program should send a copy of the updated board to each of the other players and then the program will assign a new current player.

## Minor Requirements

**Colorful:**

The game is colorful and epic to look for.

## Troublesome UML

Trouble

GUIClientConsole

objPlayer

ChatClient

**Trouble**

numOfPlayers:int

player1:objPlayer

player2:objPlayer

player3:objPlayer

player4:objPlayer

strCurrentPlayer:String

board[]:objPlayer

Trouble(){}

Trouble(P1:String,P2:String){}

Trouble(P1:String,P2:String,P3:String){}

Trouble(P1:String,P2:String,P3:String,P4:String){}

switchActivePlayer()

getCurrentPlayer():String

setCurrentPlayer(strCurrentPlayer:String)

getBoard():objPlayer[]

setBoard(board:objPlayer[])

objBoard:objPlayer[]

op:objPlayer

currentRoll:int

moveFromStart:Boolean

objTheGame()

StartGame()

rollDie():int

piecePosition(piece:Object[]):int

onTurnStart()

checkStart():boolean

move(pos:int,n:int,currentPlayer:char,piece:Object[])

ckechWin()

**objPlayer**

objInStart:Object[]

objPiece:Object[]

inHome:int

strUserName:String

pColour:char

objPlayer()

objPlayer(colour:char,Username:String)

setStart()

getHomeCount():int

gotHome()

getNumInStart():int

setColour(colour:char)

getColour():char

getHomeIndex(col:char)

getStrUserName():String

setStrUserName(strUserName:String)

getObjInStart():Object[]

setObjInStart(intInStart:Object[])