Assignment-1 req Analysis

Specification:

When the software starts, it displays the empty 15 x 15 grids. At the each grid, either Black or White stone (circle) can be placed.
Each player takes 32 disks and chooses one color to use throughout the game.
Black place a disk first followed by white placing a disk.
players take turns in putting their disks on the board.
The game ends when one of player archived an unbroken row of five stones in the same color.

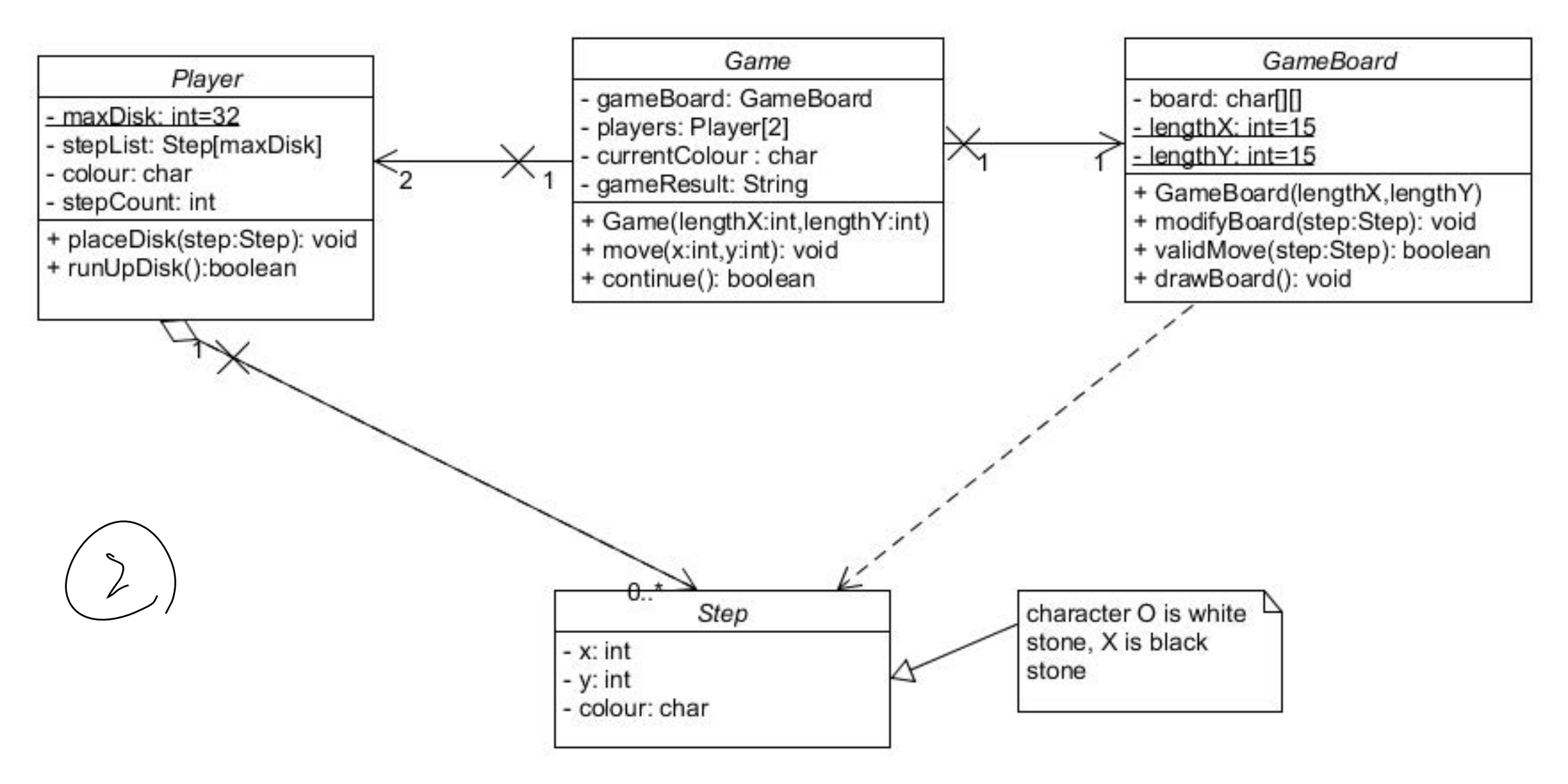
From those specifications, this game involves two players and a chess board. At first, players could
conceptualise to a Player model class, which store the necessary informations, in this case, it should have a counter for the number
of steps a player plays, maximum disks the player can have and the colour of disk the player owns.
And also, a list of step is recored in order to keep track of steps the player plays. A player has a action for place a disk,
which acutally stores the data for a step(placeDisk), unless the player uses up the disks which is determined by runUpDisk menthod.

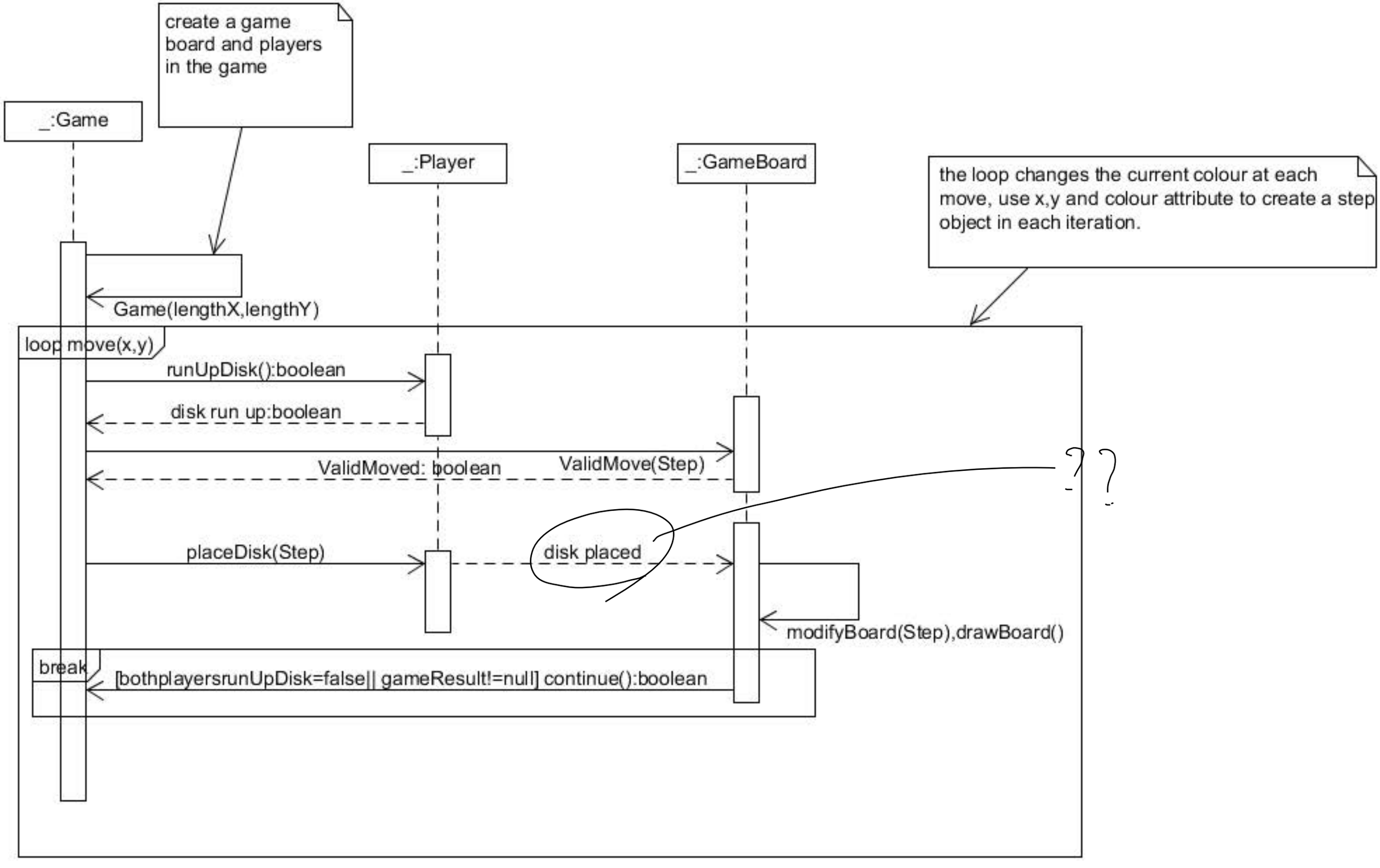
Secondly, a chess board conceptualizes to a GameBoard model class, that has a 2-dimension array simulates the disk board,
indicating the horizontal and vertical position of each grid in the board.
GameBoard has a action for storing the disks in the board(modifyBoard),
a check action(validMove) for each steps is valid in the game before storing the disks data in the array,
and a visual action(drawBoard) for displaying the board on the screen.

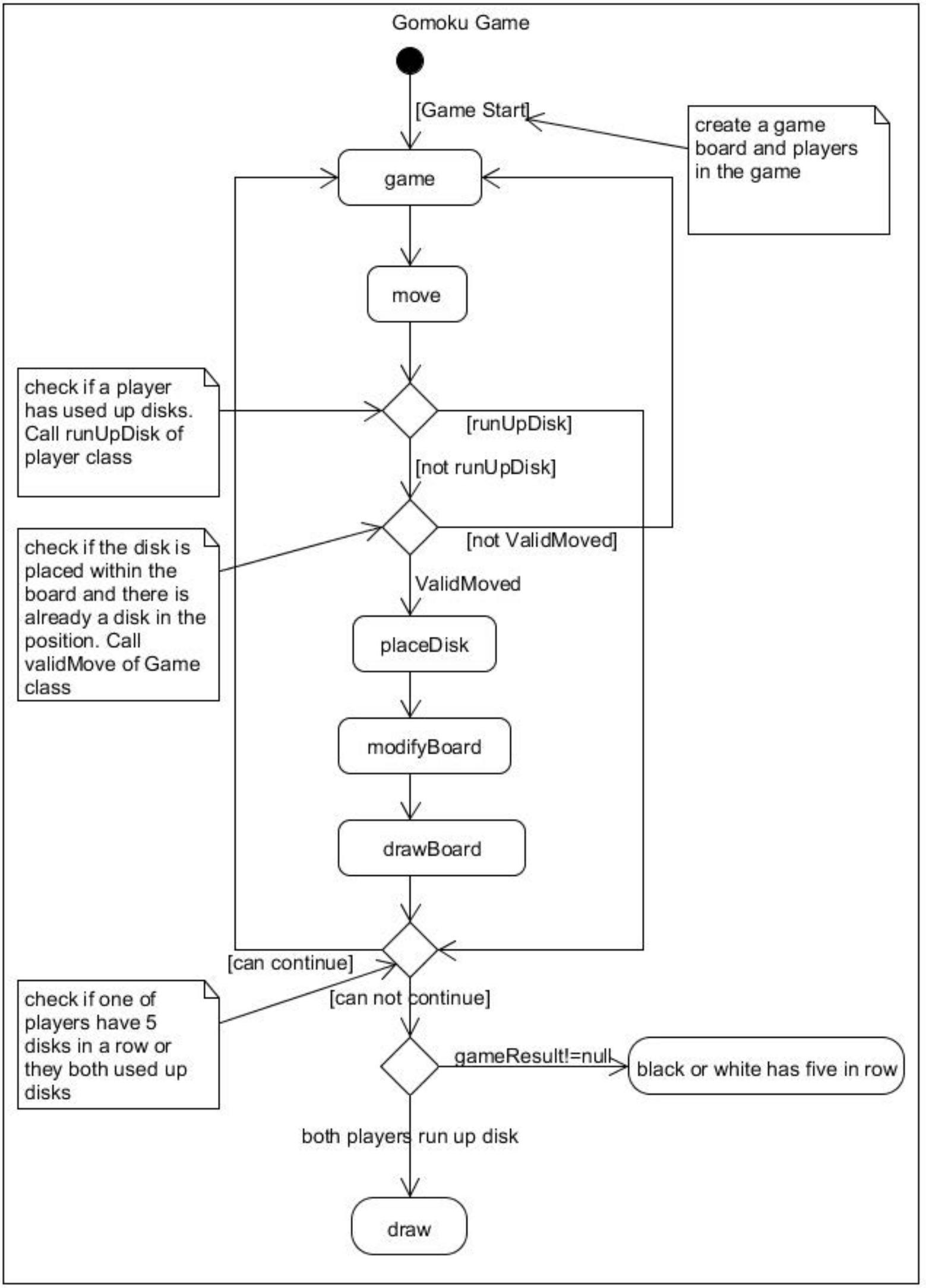
What's more, A Game class take the responsibility for controlling the round of a game. Initialise a game by creating a
board and 2 players. Then, take a turn for each move which including the actions for players and the gameBoard. Check the state
of game and report the result of a game if it comes to end, such as draw or win, if not, then contine. The state is done by
contine method.

For each step a player plays, it models to a Step class, that stores the horizontal and vertical locaiton,
and colour of disk is placed in the board.

This is how you model this game.


Player
- maxDisk: int=32
- stepList: Step[maxDisk]
- colour: char
- stepCount: int
+ placeDisk(step:Step): void
+ runUpDisk():boolean
Game
- gameBoard: GameBoard
- players: Player[2]
- currentColour : char
- gameResult: String
+ Game(lengthX:int,lengthY:int)
+ move(x:int,y:int): void
+ continue(): boolean
GameBoard
- board: char[][]
- lengthX: int=15
- lengthY: int=15
+ GameBoard(lengthX,lengthY)
+ modifyBoard(step:Step): void
+ validMove(step:Step): boolean
+ drawBoard(): void
Step
- x: int
- y: int
- colour: char
character O is white stone, X is black stone
2
1
1
1
2
1
1
0..*


create a game board and players in the game
_:Game
_:Player
_:GameBoard
the loop changes the current colour at each move, use x,y and colour attribute to create a step object in each iteration.
Game(lengthX,lengthY)
loop move(x,y)
runUpDisk():boolean
disk run up:boolean
ValidMoved: boolean
ValidMove(Step)
placeDisk(Step)
disk placed
modifyBoard(Step),drawBoard()
break
[bothplayersrunUpDisk=false|| gameResult!=null] continue():boolean
??


Gomoku Game
[Game Start]
create a game board and players in the game
game
move
check if a player has used up disks. Call runUpDisk of player class
[runUpDisk]
[not runUpDisk]
[not ValidMoved]
check if the disk is placed within the board and there is already a disk in the position. Call validMove of Game class
ValidMoved
placeDisk
modifyBoard
drawBoard
[can continue]
[can not continue]
check if one of players have 5 disks in a row or they both used up disks
gameResult!=null
black or white has five in row
both players run up disk
draw