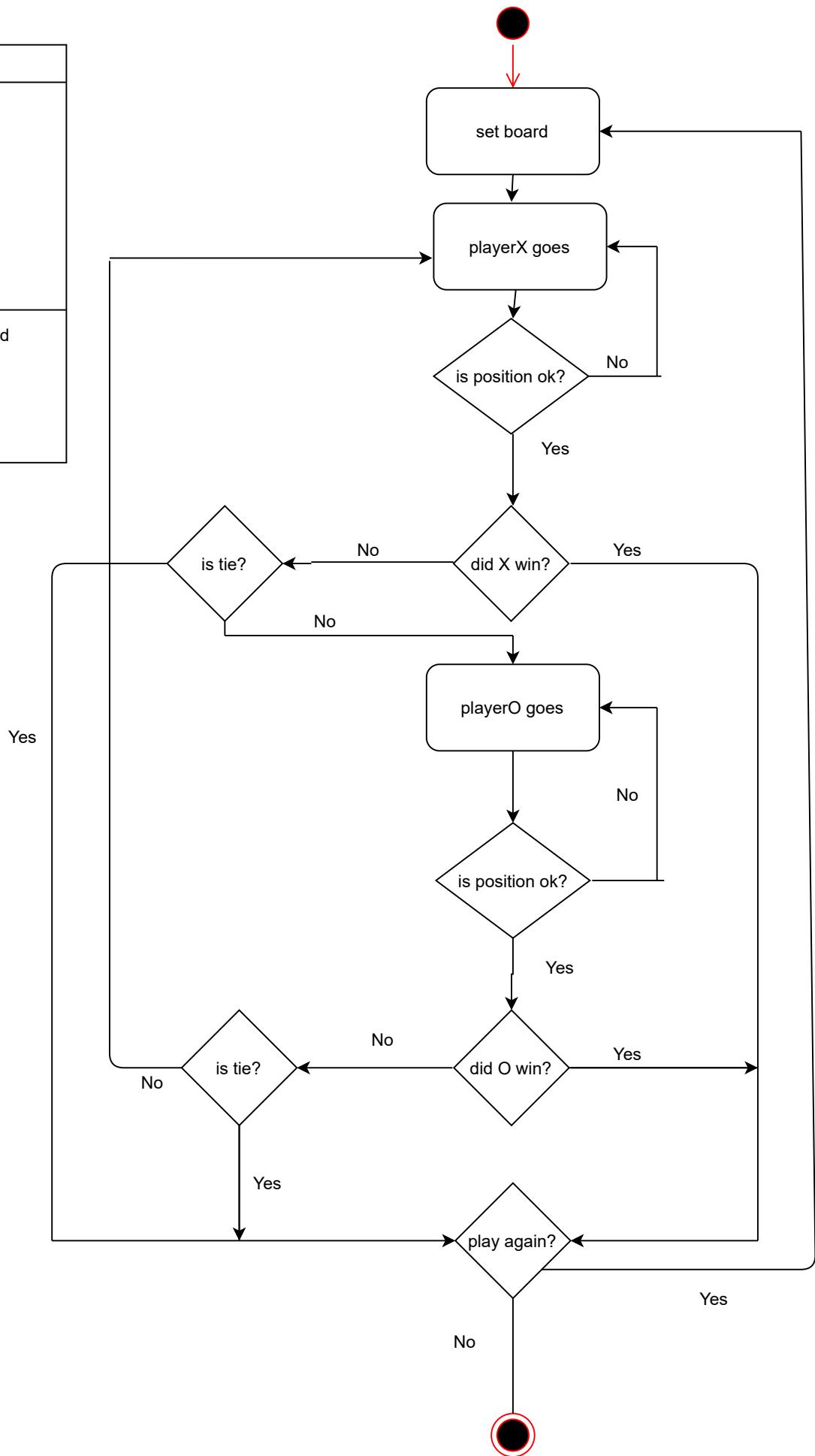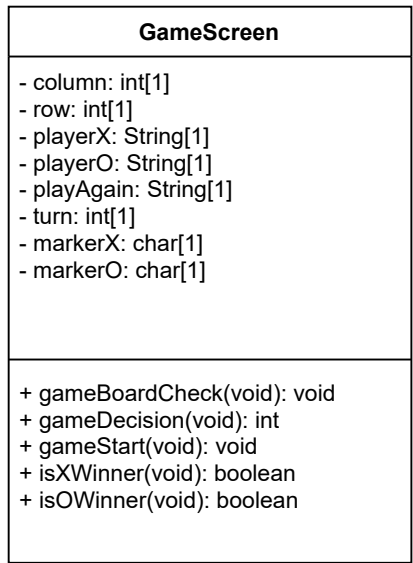User Stories
1. The users can decide which player goes first for the start of the game
2. The users can input their column selection to input their token onto the game board
3. The users can re-enter their column decision if their current selection is not available
4. The user can re-enter their token position if their input is not on a position that is already taken so that the game will run smoothly
5. The user can re-enter their token position fi their input is not a valid position in the game board so that there will be no confusion.
6. The users gets a congratulatory message if they win so that they will know the game is over
7. The users can play again if one wins so that the game can continue
8. The users can play on a blank board after a new game is started so that the game will be new
9. The users will be notified of a tie so that they can play again or end the game
10. The users can see the current game board after each turn that does not result in a win/loss or tie so that they know where to put their next token
11. The users can choose their token to know who will go first
12. The users must wait after a turn to let the other user play their token so that the game will be fair
13. The user can re-enter an input to the programs questions if the user enters an incorrect response so that the game will run smoothly

Non-functional Requirements
1. The game must be written in java
2. The game must be played on the computer

GameScreen

- column: int[1]
- row: int[1]
- playerX: String[1]
- playerO: String[1]
- playAgain: String[1]
- turn: int[1]
- markerX: char[1]
- markerO: char[1]

+ gameBoardCheck(void): void
+ gameDecision(void): int
+ gameStart(void): void
+ isXWinner(void): boolean
+ isOWinner(void): boolean

set board

playerX goes

is position ok? — No

Yes

did X win? — Yes

is tie? — No

No

playerO goes

is position ok? — No

Yes

did O win? — Yes

is tie? — No

Yes

No

play again? — Yes

No

BoardPosition
- row: int[1] - column: int[1]
+ BoardPosition(int, int): void + getRow(void): int + getColumn(void): int + equals(void): boolean + toString(int,int): string

GameBoard

-

+ GameBoard(void): void
+ checkIfFree(int): boolean
+ checkForWin(int): boolean
+ checkTie(void): boolean
+ placeToken(char, int): void
+ checkHorizWin(BoardPosition, char): boolean
+ checkVertWin(BoardPosition, char): boolean
+ checkDiagWin(BoardPosition, char): boolean
+ isPlayerAtPos(BoardPosition, char): boolean
+ whatsAtPos(BoardPosition): char
+ toString(void): String

game start

player X goes

place token

is space free?

No

Yes

check for Horizontal win

check for Vertical win

check for Diagonal win

did player X win?

No

is tie?

Yes

No

prints string of current game board

player O goes

play again?

Yes

No

Yes