CPSC 2150 HW2 Writeup

Rex Oliver

February 26 2019

1 Requirements Analysis

Functional Requirements:

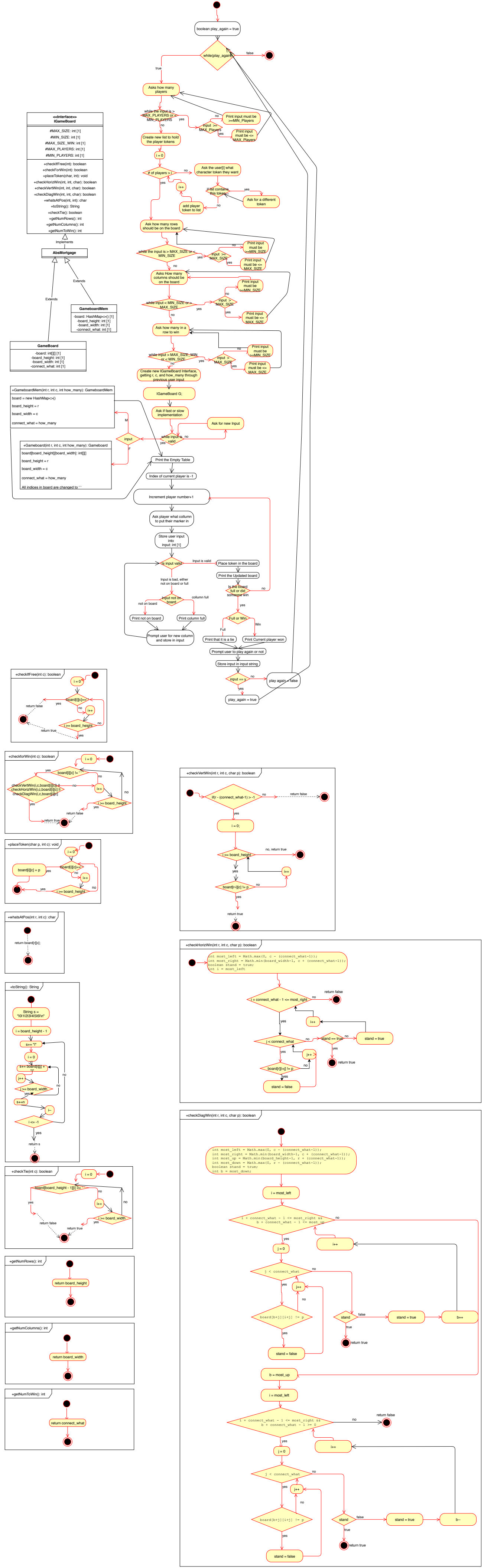
- As a user, I can choose how many rows should be on the board.
- As a user, I can choose how many columns should be on the board.
- As a user, I can choose how many in a row to win.
- As a user, I can choose what column to place my marker in.
- As a user, I can choose to play again or not.
- As a user, I can view the prompt that asks the user to make a move.
- As a user, I can view the prompt that asks the user to play again or not.
- As a user, I can view the prompt that asks the user how many rows should be on the board.
- As a user, I can view the prompt that asks the user how many column should be on the board.
- As a user, I can view the prompt that asks the user how many in a row to win?
- As a user, I can view the printed board with updated moves each turn.
- As a user, I can be notified of a win.
- As a user, I will not lose a turn for bad input.
- As a user, I can view the prompt that asks the user how many players to play the game.
- As a user, I can choose how many players to play in the game.
- As a user, I can view the prompt that asks each individual player what character their token will be.
- As a user, I can choose what character token each player will have.
- As a user, I can view the prompt that asks the user to choose a fast or memory efficient implementation of the game.
- As a user, I can choose the game to be a fast of memory efficient implementation.

Non-functional Requirements

- The system must run on the School of Computing Linux Systems.
- The system must be written in Java.
- The system uses a 2D array implementation or a List implementation

2 Design

UML Diagram on next page.



«interface»
IGameBoard
+checkIfFree(int): boolean
+checkForWin(int): boolean
+placeToken(char, int): void
+checkHorizWin(int, int, char): boolean
+checkVertWin(int, int, char): boolean
+checkDiagWin(int, int, char): boolean
+whatsAtPos(int, int): char
+toString(): String
+checkTie(): boolean
+getNumRows(): int
+getNumColumns(): int
+getNumToWin(): int
Implements
AbsMortgage
Extends
GameboardMem
GameBoard
boolean play_again = true
while(play_again)
Asks how many players
Print the Empty Table
Increment player number+1
Prompt user to play again or not
«checkIfFree(int c): boolean
«checkForWin(int c): boolean
«placeToken(char p, int c): void
«whatsAtPos(int r, int c): char
«toString(): String
«checkTie(int c): boolean
«getNumRows(): int
return board_height
«getNumColumns(): int
return board_width
«getNumToWin(): int
return connect_what
«checkVertWin(int r, int c, char p): boolean
«checkHorizWin(int r, int c, char p): boolean
«checkDiagWin(int r, int c, char p): boolean

3 Deployment

Navigate inside the directory cpsc2150
Enter the command `make`
Enter the command `make run`