**Project Sprint #2**

Implement the following features of the SOS game: (1) the basic components for the game options (board size and game mode) and initial game, and (2) S/O placement for human players ***without*** checking for the formation of SOS or determining the winner. The following is a sample interface. The implementation of a GUI is strongly encouraged. You should practice object-oriented programming, making your code easy to extend. It is important to separate the user interface code and the game logic code into different classes (refer to the TicTacToe example). xUnit tests are required.

|  |  |  |
| --- | --- | --- |
| SOS Icon  Description automatically generated Simple game Icon  Description automatically generated General game Board size  8 | | |
| Blue player  Icon  Description automatically generated S  Icon  Description automatically generated O | |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | |  |  |  |  |  |  |  |  | | O |  |  |  |  |  |  |  | |  |  | S | O | S |  |  |  | |  |  |  |  | S |  |  |  | |  |  |  |  |  |  |  |  | |  |  |  |  |  |  |  |  | |  |  |  |  |  |  |  |  | |  |  |  |  |  |  |  | S | | Red player  Icon  Description automatically generated S  Icon  Description automatically generated O |
|  | Current turn: blue (or red) |  |

Figure 1. Sample GUI layout of the Sprint 2 program

**Deliverables:**

1. **Demonstration (8 points)**

Submit a video of no more than three minutes, clearly demonstrating that you have implemented the required features and written some automated unit tests. In the video, you must explain what is being demonstrated.

|  |  |  |
| --- | --- | --- |
|  | **Feature** |  |
| 1 | Choose board size | Checkmark with solid fill |
| 2 | Choose game mode | Checkmark with solid fill |
| 3 | Initial game of the chosen board size and game mode | Checkmark with solid fill |
| 4 | “S” moves | Checkmark with solid fill |
| 5 | “O” moves | Checkmark with solid fill |
| 6 | Automated unit tests | Checkmark with solid fill |
| … |  |  |

1. **Summary of Source Code (1 points)**

|  |  |  |
| --- | --- | --- |
| Source code file name | Production code or test code? | # lines of code |
| Sprint2GameLogic | production | 25 |
| Sprint2UI | production | 218 |
| Sprint2UITest | test | 131 |
| Total | | 374 |

**You must submit all source code to get any credit for this assignment.**

1. **Production Code vs User stories/Acceptance Criteria (3 points)**

Update your user stories and acceptance criteria from the previous assignment and ensure they adequately capture the requirements. Summarize how each of the following user story/acceptance criteria is implemented in your production code (class name and method name etc.)

|  |  |
| --- | --- |
| **User Story ID** | **User Story Name** |
| 1 | Choose a board size |
| 2 | Choose the game mode of a chosen board |
| 3 | Start a new game of the chosen board size and game mode |
| 4 | Make a move in a simple game |
| 6 | Make a move in a general game |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **User Story ID and Name** | **AC ID** | **Class Name(s)** | **Method Name(s)** | **Status (complete or not)** | **Notes (optional)** |
| 1. Choose a board size | 1.1 | Sprint2UI | startGame(): this method will retrieve the selected board size.  displayBoard(int): this method creates the game board with the chosen size. | complete | It’s worth mentioning at the moment the Sprint2GameLogic file isn’t in much use in most things on this sheet. |
|  | 1.2 |  |  | complete | I went with predetermined board sizes that the user chooses from, so there is no possibility of an invalid choice anymore. |
|  | … |  |  |  |  |
| 2. Choose the game mode of a chosen board | 2.1 | Sprint2UI | startGame(): this method will retrieve the selected game mode.  displayBoard(int): this method creates the game board with the chosen game mode. | Complete |  |
|  | 2.2 | Sprint2UI | startGame(): this method will retrieve the selected game mode.  displayBoard(int): this method creates the game board with the chosen game mode. | Complete |  |
|  | 2.3 |  |  | complete | I removed the possibility of no game mode being selected by having an option pre-selected so there will always be a game mode selected. The info regarding 2.3 is now null. |
| 3. Start a new game of the chosen board size and game mode | 3.1 | Sprint2UI | startGame(): this method will start the game with the chosen board size and game mode. | complete |  |
|  | 3.2 |  |  |  | Similarly to previous items on this sheet, I removed the possibility of this happening by having a preset board size and preset game mode selection so no matter what, the game can’t be created without a board size and game mode. Therefore, the info regarding 3.2 is now null. |
| 4. Make a move in a simple game | 4.1 | Sprint2UI | takeTurn(JButton) allows a user to make a move. | complete |  |
|  | 4.2 | Sprint2UI | takeTurn(JButton): This technically comes into play here, but also nothing will happen if no move is made. | complete |  |
|  | 4.3 | Sprint2UI | TakeTurn(JButton): There is some logic to show a pop up if a move is invalid and lets the user make a new move. | complete |  |
| 5. A simple game is over | 5.1, 5.2 |  |  | incomplete | This has not been implemented into the game logic yet. |
| 6. Make a move in a general game | 6.1 | Sprint2UI | takeTurn(JButton) allows a user to make a move. | complete |  |
|  | 6.2 | Sprint2UI | takeTurn(JButton): This technically comes into play here, but also nothing will happen if no move is made. | complete |  |
|  | 6.3 | Sprint2UI | TakeTurn(JButton): There is some logic to show a pop up if a move is invalid and lets the user make a new move. | complete |  |
| 7. A general game is over |  |  |  | incomplete | This has not been implemented into the game logic yet. |
| … |  |  |  |  |  |

1. **Tests vs User stories/Acceptance Criteria (3 points)**

Summarize how each of the user story/acceptance criteria is tested by your test code (class name and method name) or manually performed tests.

|  |  |
| --- | --- |
| **User Story ID** | **User Story Name** |
| 1 | Choose a board size |
| 2 | Choose the game mode of a chosen board |
| 3 | Start a new game of the chosen board size and game mode |
| 4 | Make a move in a simple game |
| 6 | Make a move in a general game |

4.1 Automated tests directly corresponding to the acceptance criteria of the above user stories

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **User Story ID and Name** | **Acceptance Criterion ID** | **Class Name (s) of the Test Code** | **Method Name(s) of the Test Code** | **Description of the Test Case (input & expected output)** |
| 1. Choose a board size | 1.1 | Sprint2UITest | testBoardSizeSelection() | Tested board sizes: 3,4,5,6  expected output: board size would match given inputs |
|  | 1.2 | Sprint2UITest | testBoardSizeSelection() | There are not options for incorrect board sizes so there is nothing to test. |
|  | … |  |  |  |
| 2. Choose the game mode of a chosen board | 2.1 | Sprint2UITest | testChoosingGameMode() | Test inputs: test simple radio button  expected outputs: the corresponding game mode selection matches up |
|  | 2.2 | Sprint2UITest | testChoosingGameMode() | Test inputs: Test general radio button.  expected outputs: the corresponding game mode selection matches up |
|  | 2.3 | Sprint2UITest | testChoosingGameMode() | There is no possibility of a game mode not being chosen due to a preset option. |
| 3. Start a new game of the chosen board size and game mode | 3.1 | Sprint2UITest | testStartGame() | Test inputs: check for empty board.  expected output: the board will be set.  If the board is full that means the game mode and board size was chosen. |
|  | 3.2 | Sprint2UITest | testStartGame() | There is no possibility of a game not starting without a board size and game mode since both have preset options. |
| 4. Make a move in a simple game | 4.1 | Sprint2UITest | testMakingMoveSimpleGame() | Test inputs: tries s and o inputs with simple game selected  expected outputs: the outputs will be an S and O on the board |
|  | 4.2 | Sprint2UITest | testMakingMoveSimpleGame() | Test input: nothing  test output: nothing  There won’t be any inputs or outputs if a move isn’t made |
|  | 4.3 | N/A | N/A | This has to be a manual test. My auto test won’t cover this |
| 5. A simple game is over | 5.1, 5.2 | N/A | N/A | N/A |
| 6. Make a move in a general game | 6.1 | Sprint2UITest | testMakingMoveGeneralGame() | Test inputs: tries s and o inputs with general game selected  expected outputs: the outputs will be an S and O on the board |
|  | 6.2 | Sprint2UITest | testMakingMoveGeneralGame() | Test input: nothing  test output: nothing  There won’t be any inputs or outputs if a move isn’t made |
|  | 6.3 | Sprint2UITest | N/A | This has to be a manual test. My auto test won’t cover this |
| 7. A general game is over | N/A | N/A | N/A | N/A |
| … |  |  |  |  |

4.2 Manual tests directly corresponding to the acceptance criteria of the above user stories

**Everything that isn’t covered by auto tests:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **User Story ID and Name** | **Acceptance Criterion ID** | **Test Case Input** | **Test Oracle (Expected Output)** | **Notes** |
| 1. Choose a board size | 1.1 |  |  |  |
|  | 1.2 |  |  |  |
|  | … |  |  |  |
| 2. Choose the game mode of a chosen board | 2.1 |  |  |  |
|  | 2.2 |  |  |  |
|  | 2.3 |  |  |  |
| 3. Start a new game of the chosen board size and game mode | 3.1 |  |  |  |
|  | 3.2 |  |  |  |
| 4. Make a move in a simple game | 4.1 |  |  |  |
|  | 4.2 |  |  |  |
|  | 4.3 | I tried making a move on a space that is taken. | A popup showed up letting me know I can’t. I get the chance to go again. |  |
| 5. A simple game is over | 5.1, 5.2 |  |  |  |
| 6. Make a move in a general game | 6.1 |  |  |  |
|  | 6.2 |  |  |  |
|  | 6.3 |  |  |  |
| 7. A general game is over |  |  |  |  |
| … |  |  |  |  |

4.3 Other automated or manual tests not corresponding to the acceptance criteria of the above user stories

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Number** | **Test Input** | **Expected Result** | **Class Name of the Test Code** | **Method Name of the Test Code** |
|  |  |  |  |  |
|  |  |  |  |  |