**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 | Complete |
| 2 | Choose game mode | Complete |
| 3 | Initial game of the chosen board size and game mode | Complete |
| 4 | “S” moves | Complete |
| 5 | “O” moves | Complete |
| 6 | Automated unit tests | Complete |
| … |  |  |

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

|  |  |  |
| --- | --- | --- |
| Source code file name | Production code or test code? | # lines of code |
| Main.py | Production | ~210 |
| Tests.py | Test | ~263 |
| Total | | ~473 |

**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 Board Size | 1.1 | App | update\_board\_size | Complete |  |
| 1 Choose invalid board size | 1.2 | App | update\_board\_size | Complete | Makes sure board size input is greater than 3 and less than 10. |
|  | … |  |  |  |  |
| 2 Choose Simple Game Mode | 2.1 | App | Create\_widgets, Reset | Complete | When selecting simple, after general had been chose, completely reset the board to start a new game. |
| 2 Choose General game mode | 2.2 | App | Create\_widgets, Reset | Complete | When selecting general, after simple had already been selected |
|  | … |  |  |  |  |
| 3 Start a new game of the chosen board size and game mode | 3.1 | App | Update\_board\_size, Reset | Complete | When updating the board size at the top of the game, it calls the update board size function that resets the board to the new size, and when selecting a new game mode, it calls the reset function to start a new game of the given game mode. |
| 3 Start a new game with invalid board size or invalid game mode | 3.2 | App | Update\_board\_size, Reset | Complete | When inputing an invalid board size option in, the update\_board\_size method will still create a new game of chosen size, but will update the new chosen size to be either 3 or 10 depending on the closest option. |
|  | … |  |  |  |  |
| 4 Make a valid move in a simple game | 4.1 | App | Clicked, Create\_widgets | Complete | When the game mode is simple and an empty space is clicked on by a player, the current player token will be set to fill that empty space. |
| 4 Make an invalid move in a simple game | 4.2 | App | Clicked, Create\_widgets | Complete | When a non empty space is clicked in a simple game mode, inside of the board, an error message will display asking the player to select another option. |
|  | … |  |  |  |  |
| Make a valid move in a general game. | 6.1 | App | Clicked, Create\_widgets | Complete | When the game mode is set to general, and there are still empty slots available, and when one is selected, update the clicked box to be the symbol of the current player. |
| Make and invalid move in a general game. | 6.2 | App | Clicked, Create\_widgets | Complete | When the game mode is set to general, and an invalid move is played, display an error message and let the current player choose a new empty square until they have made a vlaid move. |

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 Board Size | 1.1 | NewTestApp | Test\_Resizeboard() | Test resize board is initialized with a board size of 5, then gets updated to be a board size of three, which the assert statement confirms. |
| 1 Choose invalid board size | 1.2 | NewTestApp | testInvalidResizeBoard() | Test attempts to resize board to be 0, however if statements make sure the boardsize is not outside of playable limits, and resets the board to be the minimum size of 3. Then places player1’s token at board row 3 col 3, to make sure the board is resized. |
| 2 Choose the game mode of the board | 2.1 | NewTestApp | Test\_chooseGameMode() | By default the SOS game mode is set to simple, but clicking the radio button will update it and reset the board to start a new game of the intended game mode. The assortment statement confirms this. |
|  | … |  |  |  |
| 3 Start a new game of the chosen board size and game mode. | 3.1 | NewTestApp | Tests\_StartNewGame | Start new game, selects the simple game mode and keeps the default board size, and tests the game has started by “placing” player 1’s marker at a spot on the board. The assertion statement confirms the game mode, and that player1 and player2 can place their tokens. |
|  | … |  |  |  |
| 4 Make a valid move in a simple game | 4.1 | NewTestApp | Tests\_MakeMoveSimpleGameMode() | Sets the game mode to simple, and places player 1’s token down in board spot 0,0. Then the assertion statement verifys, board at 0,0 has an S in that spot. |
| 4 Make an invalid move in a simple game. | 4.2 | NewTestApp | test\_MakeMoveGeneralGameMode() | Sets the game mode of the board to simple, and places player 1’s token on the button. Player2 attemps to place the opposite token in the same spot, but the error is caught. And the button is not updated. |
| 6 Make a valid move in a general game. | 6.1 | NewTestApp | Tests\_MakeMoveGeneralGameMode() | Sets the game mode to general and starts a new game, places player 1’s token in spot 0,0 to make sure the game has officially started. The assertion statement verifies that in spot 0,0 player 1’s token is there. |
| 6 Make an invalid move in a general game. | 6.2 | NewTestApp | Test\_MakeInvalidGeneralGameMode | Sets the game mdoe to general and starts a new game. Places player 1’s token in spot 0,0 and then trys to place player2’s token in the same place. However this is caught and player 2 is asked to choose a new spot. The assert statement confirms that player2 did not overwrite player 1’s spot. |

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

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **User Story ID and Name** | **Acceptance Criterion ID** | **Test Case Input** | **Test Oracle (Expected Output)** | **Notes** |
| 1 Choose a valid board size | 1.1 | Input board size of 5. | Board size GUI should be updated to have a row and col of 5x5. |  |
| Choose an invalid board size | 1.2 | Input board size of 11. | Board size GUI should only be updated to be a 10x10 grid. |  |
| 2 Choose the game mode of the chosen board | 2.1 | Any player can select to change the game mode. | The GUI should reset the board and run with the new rule set active. |  |
| 4 Make a valid move in a simple game | 4.1 | Player 1 selects their token and choose an empty spot on the board. | The board should update so the pressed button should have the updated player token on it. |  |
| Make an invalid move in a simple game | 4.2 | Player 2 Selects their token, and tries to place it down on an already marked spot. | The GUI should display an error message saying to choose a new spot, and player 2 can choose a new spot to place their token. |  |
| 6 Make a valid move in a general game | 6.1 | Player 1 selects their token and choose an empty spot on the board. | The board should update so the pressed button should have the updated player token on it. |  |
| Make an invalid move in a general game. | 6.2 | Player 2 Selects their token, and tries to place it down on an already marked spot. | The board should update so the pressed button should have the updated player token on it. |  |

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

* N/A at this stage.