**Project Sprint #1**

The SOS game is described in CS449HomeworkOverview.docx. You should read the description very carefully.

In this assignment, you aim to specify the requirements (i.e., user stories and acceptance criteria) of the target software that allows a human player to play a simple or general SOS game against a human opponent. These requirements will be fully implemented by the end of sprint 3. The minimum features include **choosing the board size,** **choosing the game mode (simple or general)**, **starting a new game**, **making a move (in a simple or general game)**, **determining if a simple or general game is over**. The following is a sample GUI layout.

|  |  |  |
| --- | --- | --- |
| 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 | Chart, line chart  Description automatically generated | Red player  Icon  Description automatically generated S  Icon  Description automatically generated O |
|  | Current turn: blue (or red) | New Game |

Figure 1. Sample GUI layout of the first working program by the end of Sprint 3

Use the following tables to document your user stories and acceptance criteria.

You are required to use the free ChatGPT or another LLM to complete 2 user stories and 2 of their respective acceptance criteria. You must ensure that the generated user stories are correct and fix them if not. At the end of the submission, provide screenshots of your LLM prompts and answers, along with the description of errors the LLM made that you had to correct. Points will be deducted if no screenshots/description are provided.

1. **User Stories (3 points)**

* **User Story Template**: As a <role>, I want <goal> [so that <benefit>]

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID** | **User Story Name** | **User Story Description** | **Priority** | **Estimated effort (hours)** |
| 1 | Choose a board size | As an SOS Player, I want to be able to be able to choose the size of the game board so that I can play the game at a size of my liking. | High | 2 |
| 2 | Choose the game mode of a chosen board | As an SOS Player, I want to be able to choose the game mode that the game board uses so that I can play simple or general SOS games. | High | 2 |
| 3 | Start a new game of the chosen board size and game mode | As an SOS Player, I want to be able to start the game with a configured board size and game mode of my choice so that I can play the game how I want. | High | 3 |
| 4 | Make a move in a simple game | As an SOS Player, I want to be able to place either an "S" or an "O" in an empty cell so that I can make a move when playing with the simple game mode. | High | 4 |
| 5 | A simple game is over | As an SOS Player, I want to be able to tell when the game is over to know when I have won, lost or drew the game | High | 3 |
| 6 | Make a move in a general game | As an SOS Player, I want to be able to place either an "S" or an "O" in an empty cell during general game mode so that I can try to create as many SOS sequences as possible. | High | 4 |
| 7 | A general game is over | As an SOS Player, I want to be able to tell when a general game is over so that I know the final score, who won, or if the game ended in a draw. | High | 3 |

1. **Acceptance Criteria (AC) (12 points): Add/delete rows as needed.**

|  |  |  |  |
| --- | --- | --- | --- |
| **User Story ID and Name** | **AC**  **ID** | **Description of Acceptance Criterion** | **Status (completed, toDo, inPprogress)** |
| 1. Choose a board size | 1.1 | AC 1.1 Successful configuration of a game board size.  Given A user entered valid game board size.  When The user starts a game with this board size  AND a valid game mode  Then The program should start a new game  AND the board will be the correct size  AND all cells will be initialized to EMPTY. | In Progress |
| 1.2 | AC 1.2 Unsuccessful configuration of a game board size.  Given A user entered invalid board size.  When The user starts a game with this board size  AND a valid game mode  Then The game shouldn't start,  AND an error message should be displayed on screen | todo |
| 2. Choose the game mode of a chosen board | 2.1 | AC 2.1 Successful configuration of game mode  Given The user selects one of the two game modes available  When The user starts the game with the selected game mode  AND a valid board size  Then The game will start  AND be initialized correctly | todo |
| 3. Start a new game of the chosen board size and game mode | 3.1 | AC 3.1 Successfully start a new game configured with proper board size and game mode  Given A valid board size and game mode has been entered  When The user clicks the new game button  Then The game will start  AND the board will be of specified size  AND the game mode will be of specified type  AND it will be the red players turn | todo |
| 4. Make a move in a simple game | 4.1 | AC 4.1 Successfully make a move in a simple game  Given An ongoing valid simple game  When The user clicks/selects a valid board slot  AND has selected a “S” or “O” move type  Then The user’s selected move type will be placed on the board  AND the game board state will reflect the move  AND the UI will reflect the new game state  AND the game will end if an SOS occurred  AND the game will end if all board spots are full, and no SOS was made  AND the game will switch player turns if no game over occurred | todo |
|  | 4.2 | AC 4.2 Reject invalid move due to spot already taken  Given A valid ongoing simple game  When The user clicks/selects a board slot that is not empty  AND has a valid row index  AND has a valid column index  Then The board will reject the invalid move  AND a message will be displayed to the user  AND the turn is not changed | todo |
|  | 4.3 | AC 4.3 Reject invalid move due to invalid row index  Given A valid ongoing simple game  When The user clicks/selects a board slot that has a row index not in bounds of the board row size  AND the spot is empty  AND has a valid column index  Then The board will reject the invalid move  AND a message will be displayed to the user  AND the turn is not changed | todo |
|  | 4.4 | AC 4.4 Reject invalid move due to invalid column index  Given A valid ongoing simple game  When The user clicks/selects a board slot that has a column index not in bounds of the board column size  AND the spot is empty  AND has a valid row index  Then The board will reject the invalid move  AND a message will be displayed to the user  AND the turn is not changed | todo |
| 5. A simple game is over | 5.1 | AC 5.1 Successfully end a simple game on first SOS made  Given An ongoing valid simple game instance  When A player makes a valid move that makes an SOS  Then The game will end in a win state  AND display who won the game  AND the game will wait until the new game is clicked to start again. | todo |
|  | 5.2 | AC 5.2 Successfully end a simple game on board full  Given An ongoing valid simple game instance  When A player makes a valid move that fills that last empty slot  AND no SOS has been made  Then The game will end in a draw state  AND display the game ended in a draw  AND the game will wait until the new game is clicked to start again. | todo |
| 6. Make a move in a general game | 6.1 | AC 6.1 Successfully make a move in a general game  Given A valid ongoing general game  When The user clicks/selects an valid board slot  AND has selected a valid “S” or “O” move type  Then The selected letter is placed in that cell  AND the game board state reflects the move  AND the UI reflects the updated board state.  AND the game checks for SOSs from that move  AND the game ends if all board spots are full  AND the game switches turn iff an SOS was not made | todo |
|  | 6.2 | AC 6.2 Reject invalid move due to spot already taken  Given A valid ongoing general game  When The user clicks/selects a board slot that is not empty  AND has a valid row index  AND has a valid column index  Then The board will reject the invalid move  AND a message will be displayed to the user  AND the turn is not changed | todo |
|  | 6.3 | AC 6.3 Reject invalid move due to invalid row index  Given A valid ongoing general game  When The user clicks/selects a board slot that has a row index not in bounds of the board row size  AND the spot is empty  AND has a valid column index  Then The board will reject the invalid move  AND a message will be displayed to the user  AND the turn is not changed | todo |
|  | 6.4 | AC 6.4 Reject invalid move due to invalid column index  Given A valid ongoing general game  When The user clicks/selects a board slot that has a column index not in bounds of the board column size  AND the spot is empty  AND has a valid row index  Then The board will reject the invalid move  AND a message will be displayed to the user  AND the turn is not changed | todo |
| 7. A general game is over | 7.1 | AC 7.1 Winner is player with more SOSs  Given An ongoing valid general game instance  When A player makes a valid move that fills the last spot available on the board  AND one player has more SOSs than the other  Then The game will end  AND go into a win state  AND display who won the game based on the who made the most SOSs  AND the game will wait until the new game is clicked to start again. | todo |
|  | 7.2 | AC 7.2 End in draw on board full  Given An ongoing valid general game instance  When A player makes a valid move that fills the last spot available on the board  AND both players have equal amount of SOSs  Then The game will end  AND go into a draw state  AND display that the game ended in a draw  AND the game will wait until the new game is clicked to start again. | todo |

GPT Prompts and descriptions

LLM Used: ChatGPT 5

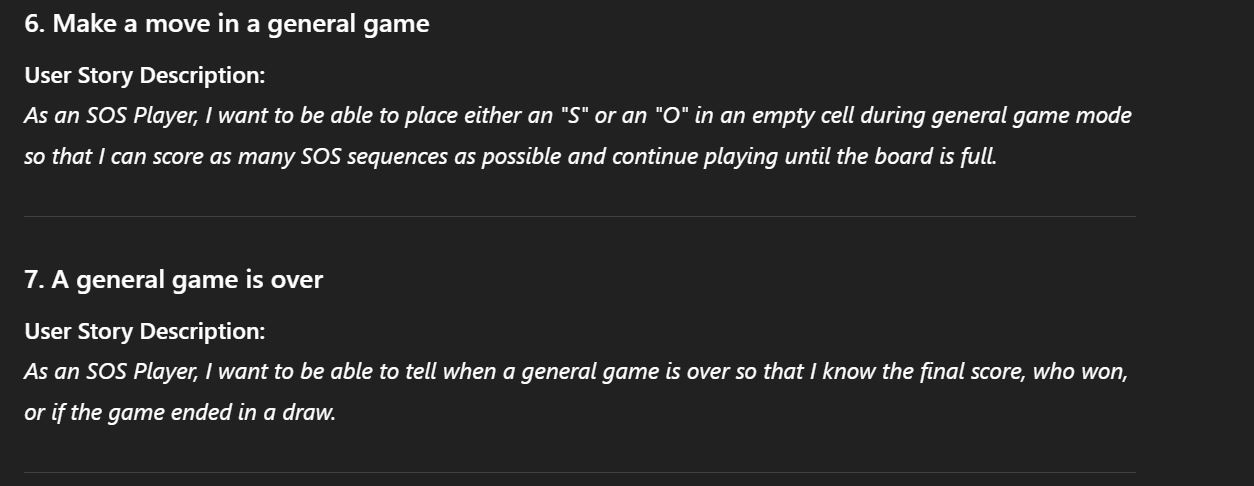
User Stories + Acceptance Criteria (User story 6 and 7 + Acceptance Criteria 6.1, 6.2, 7.1, and 7.2):

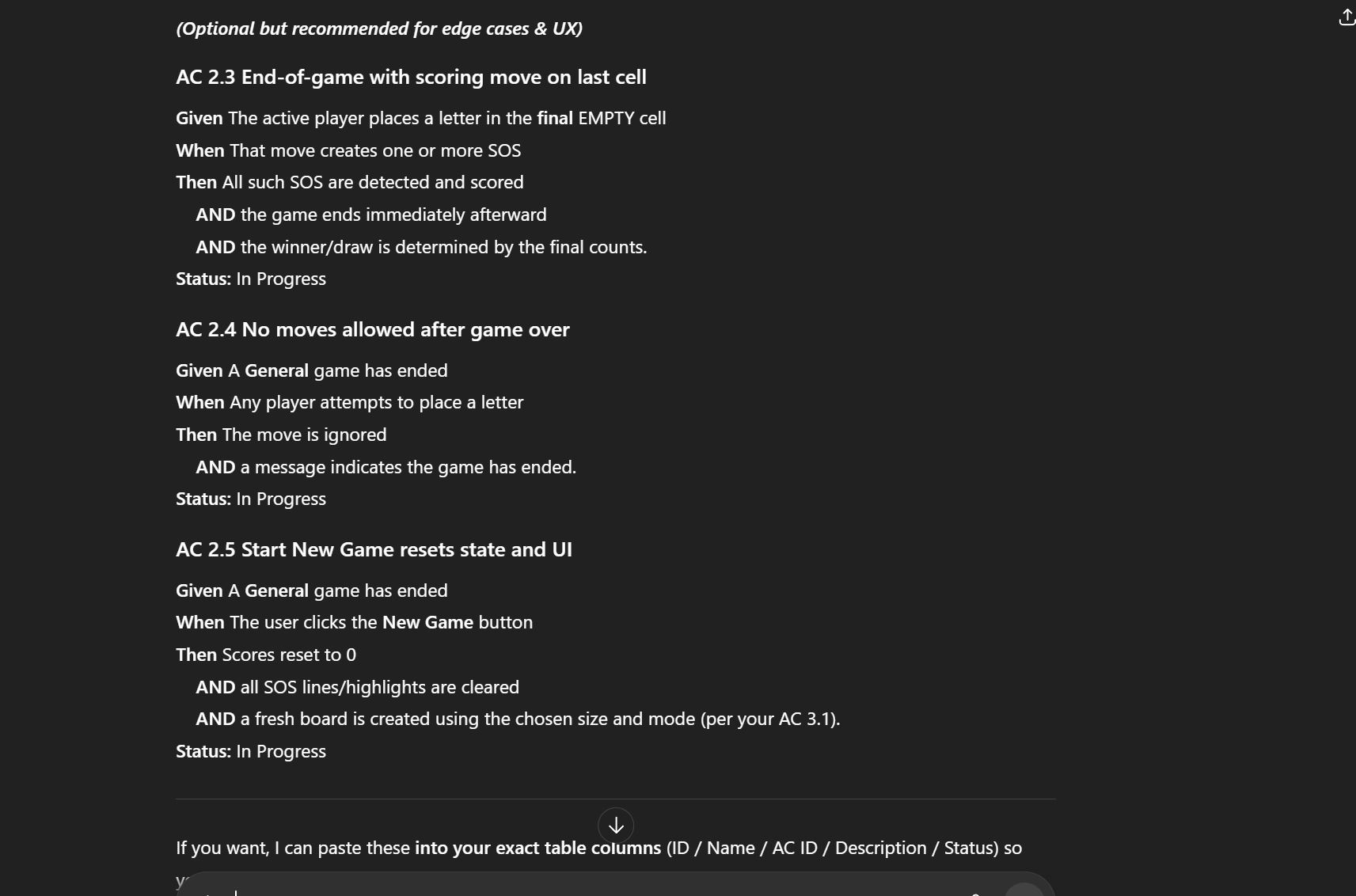
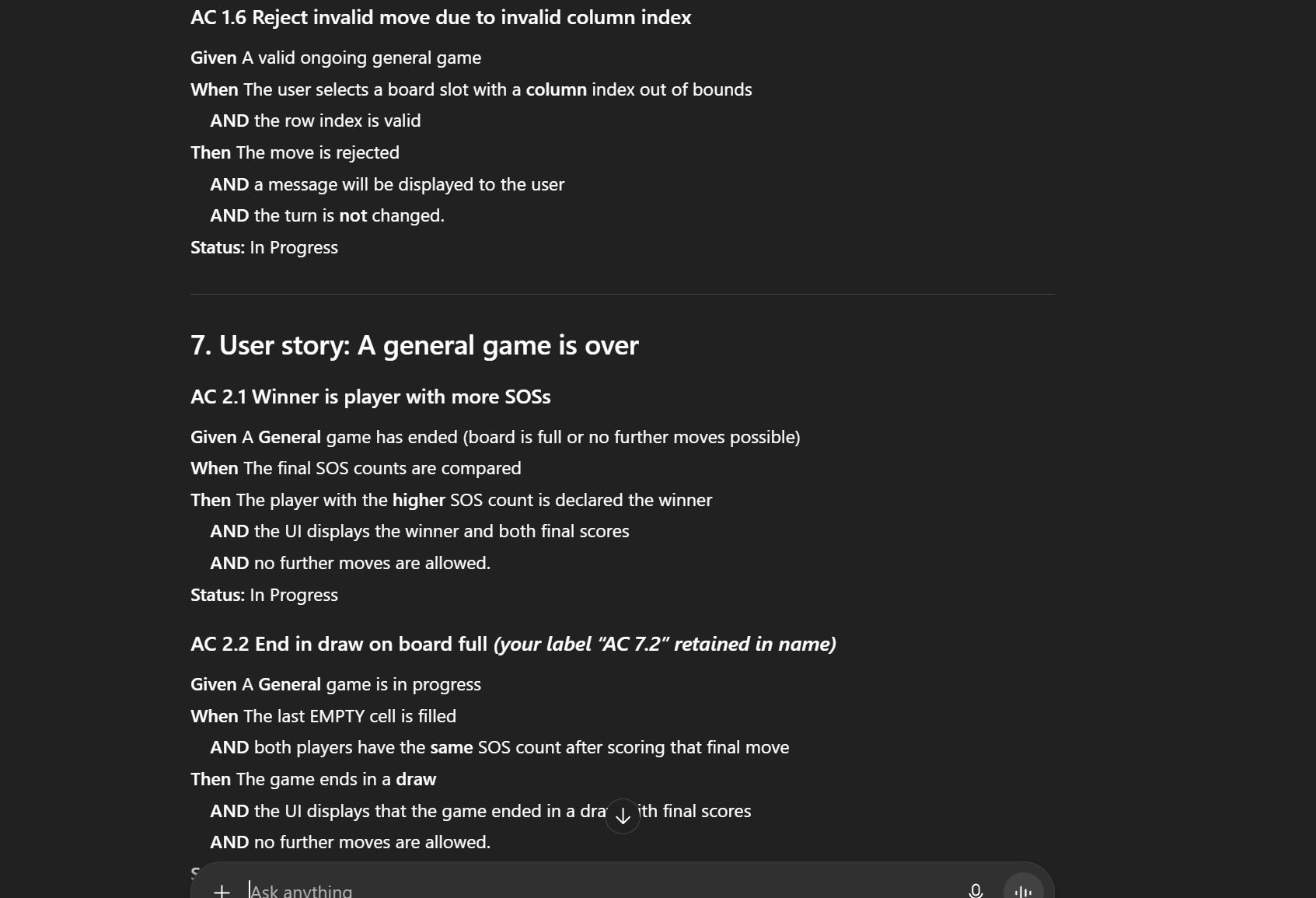
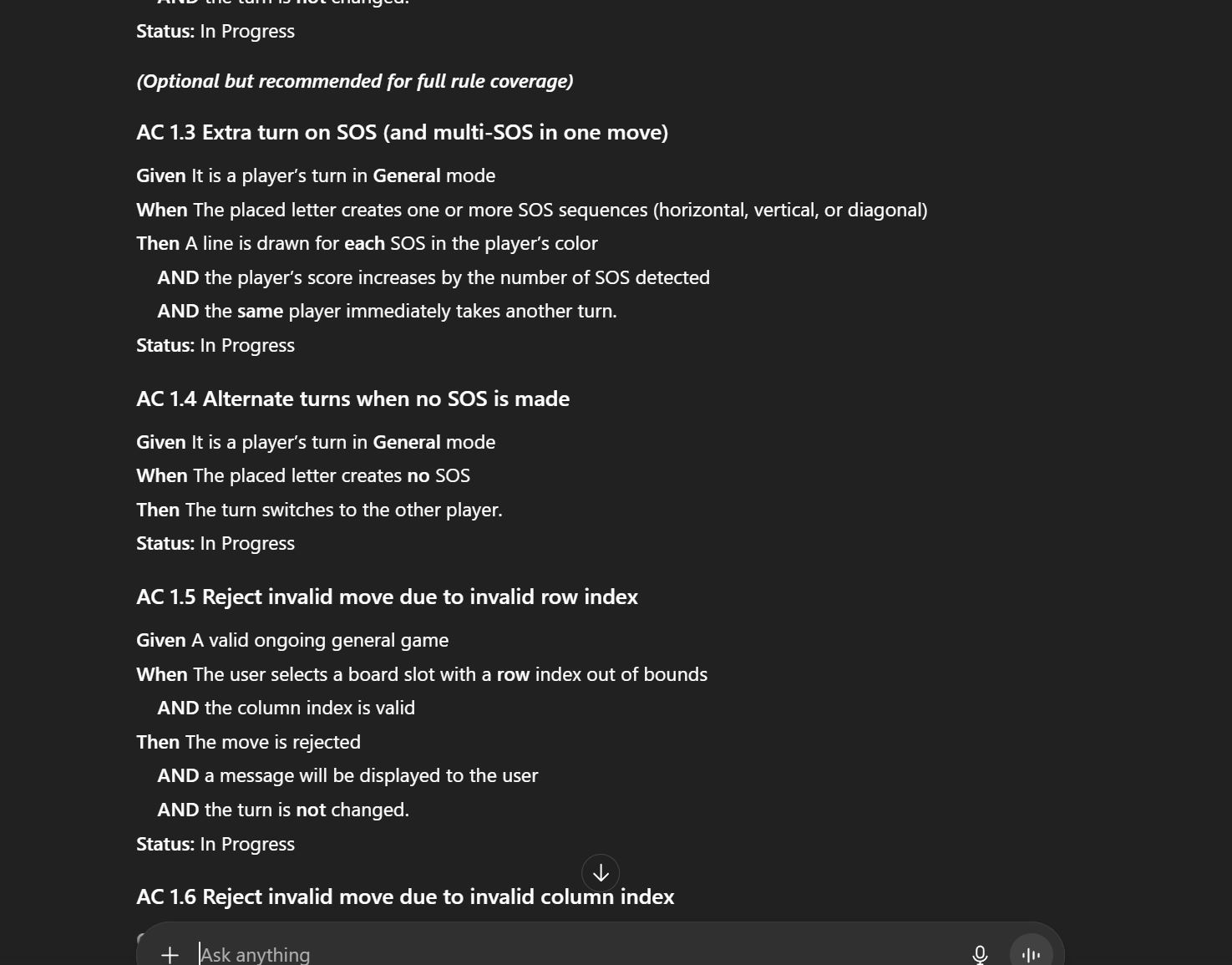
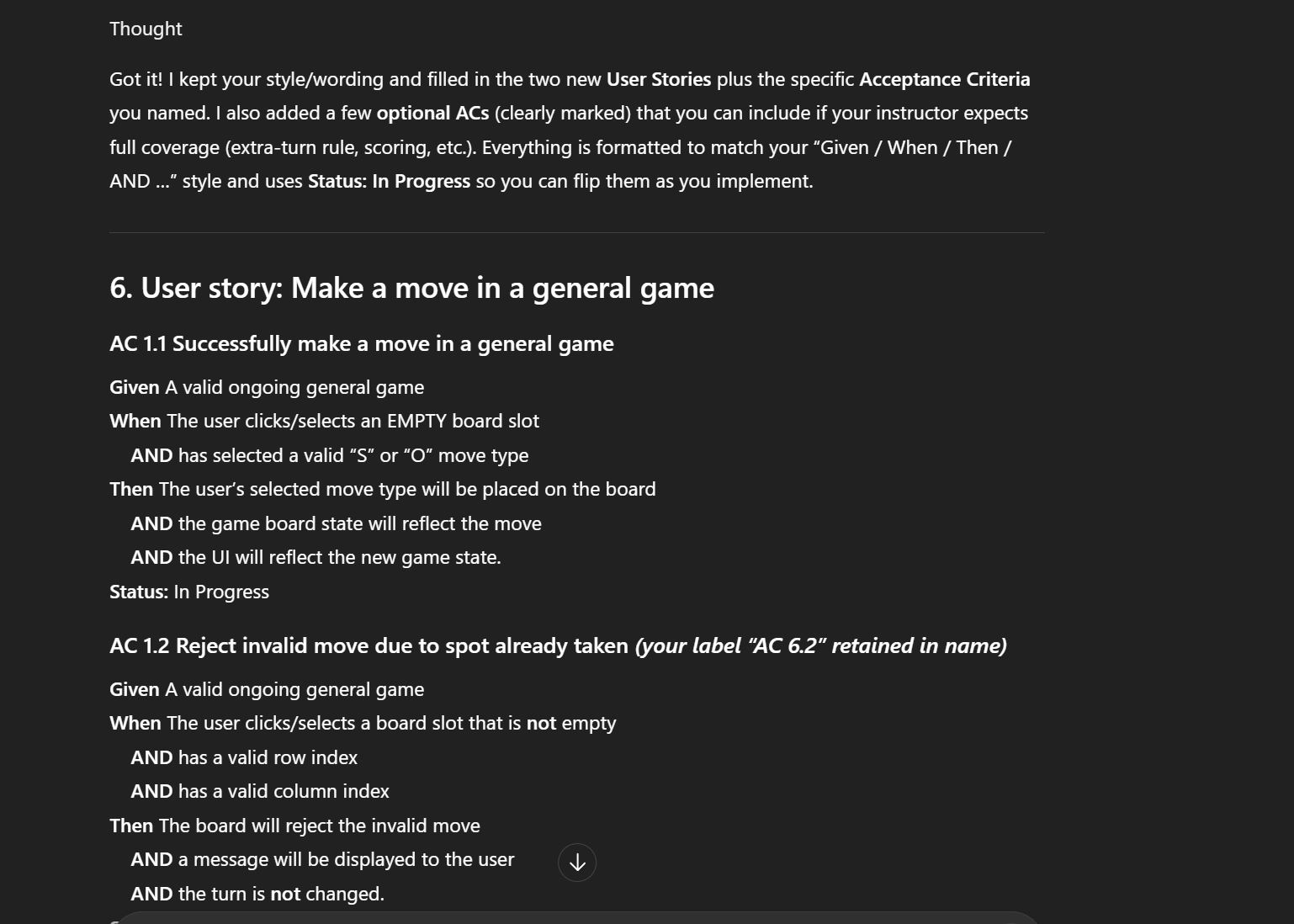
Description/Corrections made:

The LLM answer was very good due to providing it max context in the form of the assignment/game problem description and my previously made user stories and acceptance criteria. No errors were made in the user story descriptions, but the LLM did not provide any Priority or Time Estimations for the stories. Acceptance Criteria 6.1 Had an error where it said the board cell only needed to be empty instead of just saying valid to account for all restrictions. It also did not go into enough detail for the “Then” part of the AC, so extra statements had to be made. AC 6.2 was good and needed no changes because it directly related/followed AC 4.2 which was used in the prompt as an example. AC 7.1 had an error where it said that the game was already over in the “Given”, when it should still be ongoing until the “When” takes place. Also, in AC 7.2 the “Then” assumes that the game should automatically reset and start again when it should wait for new game to be clicked.

Prompt Question and Answer Screenshots

Prompt Answer Screenshots:





Prompt Question Screenshots:

