

**GitHub Repository: [github.com/zran28320-lgtm/CS449-Solitaire](https://github.com/zran28320-lgtm/CS449-Solitaire)**  
**Sprint #1 Report**

**I. User Stories**

ID	User Story Name	User Story Description	Priority	Estimated effort (hours)
1	Choose a board size	As a player, I want to input the board size before starting a game so that the game is generated with the correct dimensions.	Low	1
2	Choose the board type	As a player, I want to select the board type (English, Hexagon, or Diamond) so that I can play different variations of Solitaire.	Low	0.5
3	Start a new game of the chosen board size and type	As a player, I want to start a new game with the selected board size and type so that I can begin playing from the correct initial board state.	Middle	2
4	Make a move in a game	As a player, I want to jump a peg orthogonally or diagonally over an adjacent peg into an empty hole so that I can remove the jumped peg and progress in the game.	High	6
5	A game is over	As a player, I want the system to automatically detect when no more valid moves are possible so that I can see my final rating based on the number of marbles left.	High	5

**II. Acceptance Criteria (AC)**

User Story ID and Name	AC ID	Description of Acceptance Criterion	Status (completed, toDo, inProgress)
1. Choose a board size	1.1	AC 1.1 <Valid board size input> <b>Given</b> the game setup screen <b>When</b> the player inputs a valid integer into the board size field <b>Then</b> the size parameter is updated to the integer for the next game.	inProgress
	1.2	AC 1.2 Invalid board size input <b>Given</b> the game setup screen <b>When</b> the player inputs an invalid character like letters or negative numbers <b>Then</b> the input is considered invalid AND the game maintains its previous valid size.	inProgress

2. Choose the board typ	2.1	AC 2.1 <Default board type> <b>Given</b> a newly launched application <b>When</b> the setup screen loads <b>Then</b> the "English" board type is selected by default.	inProgress
	2.2	AC 2.2 <Change board type to “Hexagon”> <b>Given</b> the current board type is "English" <b>When</b> the player clicks the "Hexagon" radio button <b>Then</b> the active board type parameter is changed to “Hexagon” for the next game.	inProgress
	2.3	AC 2.3 <Change board type to “Diamond”> <b>Given</b> the current board type is "English" <b>When</b> the player clicks the "Diamond" radio button.	inProgress
3. Start a new game of the chosen board size and type	3.1	AC 3.1 <Initial board state generation> <b>Given</b> the selected board type and size <b>When</b> the player clicks the "New Game" button <b>Then</b> the corresponding board is generated on the screen <b>AND</b> the entire board is filled with pegs except for the central hole which remains empty.	toDo
	3.2	AC 3.2 <Resetting an ongoing game> <b>Given</b> an ongoing game where some moves have already been made and pegs removed <b>When</b> the player clicks the "New Game" button <b>Then</b> the current game is discarded <b>AND</b> the board resets to the initial fully-filled state with only the central hole vacant.	toDo
4. Make a move in a game	4.1	AC 4.1 <A valid jump move> <b>Given</b> an ongoing game with at least one valid move available <b>When</b> the player moves a peg orthogonally or diagonally over an adjacent peg into an empty hole two positions away <b>Then</b> the moved peg is placed in the new empty hole <b>AND</b> the jumped peg is removed from the board.	toDo
	4.2	AC 4.2 <An invalid move on an occupied target hole> <b>Given</b> an ongoing game <b>When</b> the player attempts to jump a peg into a hole that already contains another peg	toDo

		<b>Then</b> the move is rejected <b>AND</b> the board state remains unchanged.	
	4.3	AC 4.3 <An invalid move over an empty adjacent hole> <b>Given</b> an ongoing game <b>When</b> the player attempts to move a peg into an empty hole two positions away, but the intermediate adjacent position is empty <b>Then</b> the move is rejected <b>AND</b> the board state remains unchanged.	toDo
	4.4	AC 4.4 <An invalid move outside the board> <b>Given</b> an ongoing game <b>When</b> the player attempts to jump a peg to a position that is outside the boundaries or invalid areas of the chosen board shape <b>Then</b> the move is rejected <b>AND</b> the board state remains unchanged.	toDo
5. A game is over	5.1	AC 5.1 <Game over with Outstanding rating> <b>Given</b> an ongoing game <b>When</b> the player makes a valid move that leaves exactly 1 solitary peg on the board and no more moves are possible <b>Then</b> the game ends <b>AND</b> the system displays a message with an "Outstanding" rating.	toDo
	5.2	AC 5.2 <Game over with Very Good rating> <b>Given</b> an ongoing game <b>When</b> the player makes a move, after which no pegs on the board can legally jump over any adjacent pegs into an empty hole <b>AND</b> leaves 2 solitary peg on the board <b>Then</b> the game ends <b>AND</b> the system automatically detects this and displays an "Very Good" rating.	toDo
	5.3	AC 5.3 <Game over with Good rating> <b>Given</b> an ongoing game <b>When</b> the player makes a move, after which no pegs on the board can legally jump over any adjacent pegs into an empty hole <b>AND</b> leaves 3 solitary peg on the board <b>Then</b> the game ends <b>AND</b> the system automatically detects this and displays an "Good" rating.	toDo
	5.4	AC 5.4 <Game over with Average rating>	toDo

		<p><b>Given</b> an ongoing game</p> <p><b>When</b> the player makes a move, after which no pegs on the board can legally jump over any adjacent pegs into an empty hole</p> <p><b>AND</b> leaves 4 or more solitary peg on the board</p> <p><b>Then</b> the game ends</p> <p><b>AND</b> the system automatically detects this and displays an "Average" rating.</p>	
	5.5	<p>AC 5.5 &lt;A continuing game after a move&gt;</p> <p><b>Given</b> an ongoing game</p> <p><b>When</b> the player makes a valid move</p> <p><b>AND</b> there is still at least one legal jump available anywhere on the board</p> <p><b>Then</b> the game continues</p> <p><b>AND</b> the player is allowed to make another move.</p>	toDo

## Screenshots of LLM prompts and answers:



Complete first two user stories and two of their respective acceptance criteria in this file.

Here's drafted content you can paste directly into your document for **User Stories 1 & 2** and **two acceptance criteria for each**. I've written them cleanly in the required format so you can adjust wording if your instructor expects something slightly different.

### I. User Stories

ID: 1

User Story Name: Choose a board size

User Story Description:

As a player, I want to choose the board size before starting a game so that I can play with my preferred level of difficulty.

Priority: High

Estimated effort (hours): 3

ID: 2

User Story Name: Choose the board type

User Story Description:

As a player, I want to choose the board type (English, Hexagon, or Diamond) before starting a game so that I can play different variations of Solitaire.

Priority: High

Estimated effort (hours): 3



### II. Acceptance Criteria

User Story 1: Choose a board size

AC ID: 1.1

Description of Acceptance Criterion:

AC 1.1 – Valid board size selection

Given the game is on the setup screen

When the player selects a valid board size option

Then the selected board size is stored and displayed as the current selection

Status: toDo

AC ID: 1.2

Description of Acceptance Criterion:

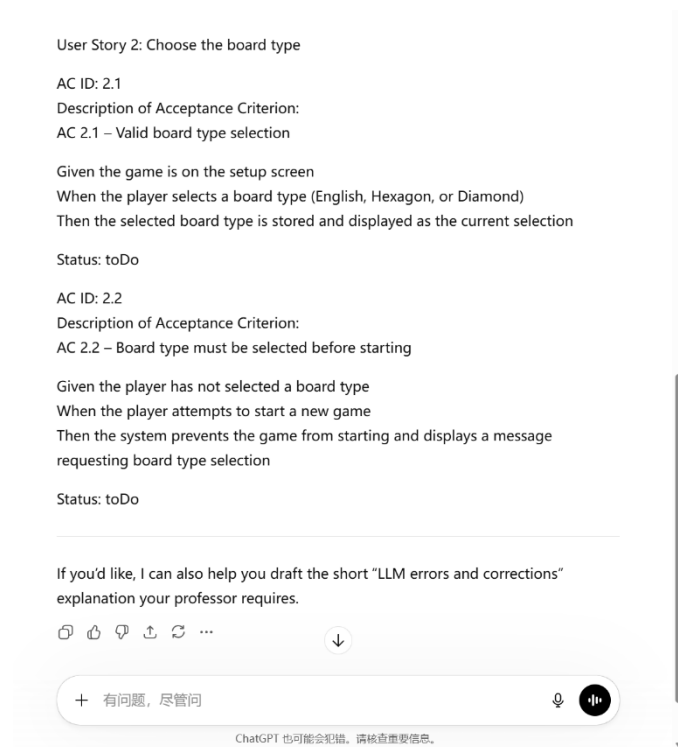
AC 1.2 – Board size must be selected before starting

Given the player has not selected a board size

When the player attempts to start a new game

Then the system prevents the game from starting and displays a message requesting board size selection

Status: toDo



### LLM Errors and My Corrections:

- (1) The LLM generated Acceptance Criteria assuming the game could have an "unselected" or blank state for board size and type, suggesting the system should prevent the game from starting and show an error. However, the "English" and a size of "7" are selected by default. I corrected this by replacing the "prevent start" criteria with an AC that tests the default state (AC 2.1) and an AC that tests actual invalid inputs for the size field (AC 1.2).
- (2) The LLM used vague terms like "valid board size option" without defining what happens if an invalid input is provided. I aligned my corrections by specifically testing for valid inputs and defining the expected behavior for invalid inputs like letters or negative numbers.