

Use Case Model

Use case: Place a mark on the tic-tac-toe board.

Actors: players

Goal: to allow one of the basic actions of the game of tic-tac-toe to be played

Preconditions: a match of tic-tac-toe must have started and it must be the actor's turn.

Summary: When a user wants to place mark on the board, a match of tic-tac-toe must be in progress and it must be the user's turn. They then select one of the empty cells on the board, after which their symbol (either X or O) is placed on the cell and the system moves the turn order forward.

Steps:

<i>Actor Actions</i>	<i>System Responses</i>
1. Select a cell on the board during their turn	2. Place the user's symbol on the cell, and then moves to the next player's turn.

Postconditions: the board remembers the user's selection by placing a mark in the indicated cell.

Use case: Time out an inactive player.

Actors: players

Goal: Prevent a game from lasting forever by limiting the amount of time allotted to each player's turn.

Preconditions: a match of tic-tac-toe must have started and the player's current turn must have longer than the amount of time allotted to their turn timer.

Summary: If a turn lasts more than the allotted time on the timer, the tic-tac-toe system ends the game and declares the winner to be the player whose turn did not time out.

Steps:

<i>Actor Actions</i>	<i>System Responses</i>
1. During their turn, does not select a valid move within the allotted time limit	2. Declares the player whose turn did not time out the winner and ends the match.

Postconditions: the match of tic-tac-toe ends and one player is declared the winner.

Use case: start a match

Actors: player

Goal: to start a match of tic-tac-toe

Preconditions: system must not be in the middle of a tic-tac-toe match.

Summary: when the user wants to start a match of tic-tac-toe, they select the kind of opponent that they would like to play against (human or basic computer). The system then queries the user for the size of the board (m, n) and the number of pieces in a row to determine a victory.

Steps:

<i>Actor Actions</i>	<i>System Responses</i>
1. Choose opponent	2. Display 'board options' dialog
3. Specify board options (m, n, k)	
4. Confirm selection	5. Remove 'board options' dialog
	6. Initialize and display custom board

Postconditions: the game board is initialized according to the user's preferences and displayed to the user.

Use case: start a match with invalid values

Related use cases:

Extension of:

start a match (extension point: step 3: Specify board options)

Steps:

<i>Actor Actions</i>	<i>System Responses</i>
1. Choose opponent	2. Display 'board options' dialog
3. Specify board options (m, n, k)	
4. Confirm selection	5a. Indicate that values are invalid
5b. Specify valid board options	
5c. Confirm selection	5. Remove 'board options' dialog
	6. Initialize and display custom board

Use case: restart the board

Actors: player

Goal: allow another match of tic-tac-toe to begin

Preconditions: either a match must be in progress or have ended.

Summary: this option allows the user to reset the board at any point after a game has begun or ended.

Steps:

<i>Actor Actions</i>	<i>System Responses</i>
1. Selects the reset option	2. Begins another match of tic-tac-toe by initializing an empty board and then displaying it to the user