Tic Tac Toe Use Case Description

Use Case 1: Load Tic Tac Toe Game

Iteration: 1

Primary Actor: Player

Goal in Context: To load the tic tac toe game.

Preconditions:

- Player has accessed an online multiplayer game platform
- Tic tac toe is available to play

Trigger: Player selects the option to load the game.

Scenario:

1. The player selects “tic tac toe” to play.
2. The game system will load the game.

Post Conditions: The game mode selection screen appears allowing the player to choose the desired game mode they want to play.

Exceptions: The system fails to load the game.

Priority: High priority, this is a game that one of the online multiplayer games that must be available on the board game platform

When Available: Always available

Frequency of Use: Used once every game session, if a player decides they want to play tic tac toe

Channel to Actor: The user interface, using a mouse and/or a keyboard

Secondary Actors: None

Channel to Secondary Actors: N/A

Open Issues: How should the game deal with a player abandoning the game midway through? Should the game set a default mode if a game mode isn’t selected? How should the system allow multiple games to be played at the same time? How should the system handle network issues when attempting to load into the game?

Use Case 2: Choose game mode

Iteration: 1

Primary Actor: Player

Goal in Context: To allow the player to choose between single player and multiplayer game mode.

Preconditions: The game mode selection screen is displayed

Trigger: The player loads into the game

Scenario:

1. The player selects single player mode or multiplayer mode.
2. The system prepares the tic tac toe game board.
3. If the single player mode is selected:
    a. The player and the bot are both assigned either an X or O.
4. If multiplayer mode is selected:
    a. The system allows the player to choose who they want to play against
    b. The player's are randomly assigned either an X or an O
5. The game display's who starts the game.

Post Conditions: The player takes turns making a move against the AI bot or another player

Exceptions:

- The system fails to load the game with the AI bot
- Fails to connect the AI bot
- Encounters some connection issues trying to find an online match.
- System fails to connect to the selected game mode

Priority: High priority

When Available: Always available during a game

Frequency of Use: Used once during a game session

Channel to Actor: The user interface, using a mouse and/or keyboard

Secondary Actors: None

Channel to Secondary Actors: N/A

Open Issues: How does the system determine the AI difficulty? What happens when the AI fails to make a move when it is their turn? How should the system handle network issues when the AI won't connect? If a player disconnects, can the player switch to an AI opponent? Should the game provide a communication feature? How should the system handle network issues when players won't connect or disconnect?

Use Case 3: Make a move

Iteration: 1

Primary Actor: Player

Goal in Context: To place the player's mark (X or O) in the selected cell in the game board.

Preconditions:

- The Game has started.
- It is the player's turn to a move

Trigger: There are still empty slots and the player clicks on an empty slot in the game board.

Scenario:

1. The player selects an empty slot on the game board.
2. The system places the player's mark on the selected slot.
3. The system updates the game board and switches turns to the other player.

Post Conditions:

- The selected slot is updated with the player's mark.
- The system is prepared for the next player's move.

Exceptions: The player selects a non-empty slot, and the system denies the move.

Priority: High priority

When Available: Always available during a game

Frequency of Use: Used multiple times a game

Channel to Actor: The user interface, using a mouse and/or keyboard

Secondary Actors: None

Channel to Secondary Actors: N/A

Open Issues: Should the game implement a time limit for making a move? Will the player be able to undo a move? What happens if a player chooses an invalid slot in the game board?

Use Case 4: Check if a player won the game

Iteration: 1

Primary Actor: Game system

Goal in Context: To check if a player has won the game by achieving three of the same mark in a row, column or diagonal.

Preconditions:

- The player has completed their turn
- There are at least 5 moves made on the game board.

Trigger: A player has completed their turn.

Scenario:

1. Player makes moves on the game board leaving their marks on the played slots (X or O)
2. A player has at three of the same mark in a row, column or diagonal
3. The game system verifies the win

Post Conditions: The game verifies a player has won and prepares to end the game

Exceptions: The system fails to evaluate a win (anything other then 3 of the same mark in a row, column or diagonal)

Priority: High priority as this determines the outcome of the game session

When Available: Always available during a game

Frequency of Use: Used after a player move, if valid

Channel to Actor: Automatic System Procedure

Secondary Actors: None

Channel to Secondary Actors: N/A

Open Issues: How should the game handle network issues or other issues if a win is not detected? How often should the game check for a win condition?

Use Case 5: Check if the game is a draw

Iteration: 1

Primary Actor: Game System

Goal in Context: To check if the game has ended in a draw, where all slots are filled with players' marks (X or O)

Preconditions:

- The player has completed their turn
- All slots are filled with players' marks (X or O)

Trigger: The player completes their turn, and no win condition is met

Scenario:

1. Player makes moves on the game board leaving their marks on the played slots (X or O)
2. The game board evaluates to see if all slots are filled
3. If all game slots are filled with no signs of a player's mark that is three in a row, column, or diagonal
4. If the board is not full, the next player makes a move

Post Conditions: The game verifies it ends with a draw and prepares to end the game

Exceptions: The system fails to evaluate a draw

Priority: medium as this determines the outcome of the game session if a player does not win

When Available: Always available during a game session

Frequency of Use: Used after a player move, if valid

Channel to Actor: Automatic System Procedure

Secondary Actors: None

Channel to Secondary Actors: N/A

Open Issues: Should the game suggest a new match or rematch option? Should the game reset automatically after the draw?

Use Case 6: End game

Iteration: 1

Primary Actor: System

Goal in Context: To conclude the game session and provide the player with end-game options

Preconditions: Players have completed all of their turns ending the game with a winner or a draw

Trigger: The game detects a win or a draw

Scenario:

1. The system declared the results of the game
2. The system offers the player with an option to play again(the system resets the game board) or exit the game(the systems navigates the player out of the game)

Post Conditions: The game board has reset or the player has exited the game

Exceptions: The game system fails to reset the game board or fails to exit

Priority: High priority as this determines the outcome of what to do next at the end of the game

When Available: Always available at the end of the game session

Frequency of Use: Used once at the end of every game session

Channel to Actor: Automatic System Procedure

Secondary Actors: None

Channel to Secondary Actors: N/A

Open Issues: Is it possible for a player to save results or statistics of the game? What should happen if a player disconnects during the ending of the game?

Use Case 7: Exit Game

Iteration: 1

Primary Actor: Player

Goal in Context: To exit the game.

Preconditions: Game is in progress or completed.

Trigger: The player selects the exit option.

Scenario:

1. The player chooses the exit game option .
2. The system closes the game interface.

Post Conditions: The player is returned to the main menu or exits the game interface.

Exceptions: The game system fails to close properly.

Priority: High priority as this allows the player to exit the interface whenever they desire.

When Available: Always available during the game

Frequency of Use: When the player wants to leave the game

Channel to Actor: Automatic System Procedure

Secondary Actors: None

Channel to Secondary Actors: N/A

Open Issues: How should the game handle unsaved progress if the player exits suddenly? Should the game prompt the player with a confirmation message to exit the game? Should the game provide an option to save statistics?

Use Case 8: Restart Game

Iteration: 1

Primary Actor: Player

Goal in Context: To restart the game after a win, draw.

Preconditions: When the current game session has ended.

Trigger: The player selects the restart option.

Scenario:

1. The player clicks on the restart option.
2. The system resets the game board and restarts the game session.

Post Conditions: The game will reset the gameboard to prepare for a new game.

Exceptions: The game system fails to reset the game board.

Priority: Medium priority as it is not necessary to complete the game, but gives the player options after the game ends.

When Available: Always available after the game ends

Frequency of Use: When a new game session is requested

Channel to Actor: Automatic System Procedure

Secondary Actors: None

Channel to Secondary Actors: N/A

Open Issues: Should the game offer multiple options when restarting, for instance, changing the game mode? Should the game provide an option to save statistics? Should the game prompt the player with a confirmation message to restart the game?