Use Case: Search Profile

Iteration: 1

Primary Actor: User

Goal in context: Allows user to search for a profile within the system they want to either view or challenge to a match.

Preconditions: The user has an active session in the main interface where they can access the search function.

Trigger: User initiates a search to find another player profile they would like to view or challenge to a match.

Scenario:

1. User is in the main interface ready to search for another player
2. User enters the name or identifier of the player
3. Search gets processed and retrieves player profile options
4. User selects a player profile from their search results where they can choose to view profile or challenge player

Post-conditions: User lands on selected player profile successfully where they can view or challenge player

Exceptions:

1. Unable to find player profile based on search results
2. Desired player profile is unavailable
3. Unable to search for players if search field is unresponsive

Priority: High. Enhances user experience through being able to interact with other players to challenge and/or view player information.

When available: Within 1st iteration.

Frequency of use: When user wants to search for other players on the platform

Channel to actor: Interaction through search query interface.

Secondary actors: N/A.

Channel to secondary actors: N/A.

Open issues: N/A.

Use Case: Challenge Player

Iteration: 1

Primary Actor: User

Goal in context: Allows user to challenge another player to a match.

Preconditions: The user has an active session in the main interface where they can access the search function, has identified a player, and is ready to initiate challenge to a game.

Trigger: User initiates challenge after selecting a player profile from their search results.

Scenario:

1. User has selected player to challenge based on search results
2. User initiates challenge to selected player

Postconditions: Challenge request has been sent to selected player

Exceptions:

1. Option to challenge player is unresponsive
2. System fails to send challenge request to desired player

Priority: High. Allows multiplayer gameplay to function with competitive play.

When available: Within 1st iteration.

Frequency of use: Whenever user wants to challenge another player to a match per game session.

Channel to actor: Interaction through challenge player option after landing on desired opponent.

Secondary actors: N/A.

Channel to secondary actors: N/A.

Open issues: N/A.

Use Case: View Profile

Iteration: 1

Primary Actor: User

Goal in context: Allows user to view selected player profile.

Preconditions: The user has an active session in the main interface where they can access the search function, and has identified a player they want to view.

Trigger: User selects a player profile from search results they want to view details of.

Scenario:

1. User has selected a player from search results
2. User initiates profile overview for desired player profile

Postconditions: User is able to view selected player’s profile successfully where they can access match history, wins/losses, etc…

Exceptions:

1. Unable to find player profile based on search results
2. System fails to retrieve profile details regarding desired player profile

Priority: High. Enhances user experience through being able to view other user statistics.

When available: Within 1st iteration.

Frequency of use: Whenever user wants to view other player profiles.

Channel to actor: Interaction through view player profile option upon selecting player they want to view.

Secondary actors: N/A.

Channel to secondary actors: N/A.

Open issues: N/A.

Use Case: Exit Application

Iteration: 1

Primary Actor: User

Goal in context: Allows user to exit out of the main application.

Preconditions: The user is in the main interface and wishes to exit out of the application.

Trigger: User presses the “Exit” button to exit session.

Scenario:

1. User is in the main menu interface
2. User selects the “Exit” button from the application menu to quit session

Postconditions: Application is closed and the user is out of the application.

Exceptions:

1. “Exit” button fails to exit user out and their application remains open

Priority: High. Users should be able to exit out of the application whenever they desire.

When available: Within 1st iteration.

Frequency of use: Once per active user session.

Channel to actor: “Exit” button in the main menu interface.

Secondary actors: N/A.

Channel to secondary actors: N/A.

Open issues: N/A.

Use Case: Leaderboard Button

Iteration: 1

Primary Actor: User

Goal in context: Allows user to view the leadership board for various games.

Preconditions: The user is in the main interface and wants to view the leadership board for various games.

Trigger: User presses the “Leaderboard” button

Scenario:

1. User is in the main menu interface
2. User selects the “Leaderboard” button from the menu

Postconditions: Display of leaderboards for Tic-Tac-Toe, Checkers, and Connect4 are available.

Exceptions:

1. “Leadership” button is unresponsive
2. Leaderboard button fails to retrieve data for leadership statistics of various games

Priority: High. Access to leaderboard is crucial for user engagement and competitive play.

When available: Within 1st iteration.

Frequency of use: Whenever user wants to check out leaderboard options.

Channel to actor: “Leaderboard” button in main menu interface.

Secondary actors: N/A.

Channel to secondary actors: N/A.

Open issues: N/A.

Use Case: Connect4 Leaderboard

Iteration: 1

Primary Actor: User

Goal in context: Allows user to view the leadership board for Connect 4.

Preconditions: The user is in the main interface and wants to view the leadership board for Connect 4.

Trigger: User presses the “Connect4 Leaderboard” button

Scenario:

1. User is in the main menu interface
2. User selects the “Leaderboard” button from the menu
3. User selects the “Connect4 Leaderboard” button

Postconditions: Display of leaderboard for Connect 4 made available for user.

Exceptions:

1. “Connect4 Leadership” button is unresponsive
2. System fails to retrieve information regarding Connect4 leaderboard data

Priority: High. Access to leaderboard is crucial for user engagement and competitive play.

When available: Within 1st iteration.

Frequency of use: Whenever user wants to view leadership information for Connect4

Channel to actor: “Connect4 Leaderboard” button after selecting “Leaderboard” option

Secondary actors: N/A.

Channel to secondary actors: N/A.

Open issues: N/A.

Use Case: TicTacToe Leaderboard

Iteration: 1

Primary Actor: User

Goal in context: Allows user to view the leadership board for Tic-Tac-Toe game.

Preconditions: The user is in the main interface and wants to view the leadership board for Tic-Tac-Toe.

Trigger: User presses the “Tic-Tac-Toe Leaderboard” button

Scenario:

1. User is in the main menu interface
2. User selects the “Leaderboard” button from the menu
3. User selects the “Tic-Tac-Toe Leaderboard” button

Postconditions: Display of leaderboard for Tic-Tac-Toe made available for user.

Exceptions:

1. “Tic-Tac-Toe Leadership” button is unresponsive
2. System fails to retrieve information regarding Tic-Tac-Toe leaderboard data

Priority: High. Access to leaderboard is crucial for user engagement and competitive play.

When available: Within 1st iteration.

Frequency of use: Whenever user wants to view leadership information for Tic-Tac-Toe

Channel to actor: “Tic-Tac-Toe Leaderboard” button after selecting “Leaderboard” option

Secondary actors: N/A.

Channel to secondary actors: N/A.

Open issues: N/A.

Use Case: Checkers Leaderboard

Iteration: 1

Primary Actor: User

Goal in context: Allows user to view the leadership board for Checkers game.

Preconditions: The user is in the main interface and wants to view the leadership board for Checkers.

Trigger: User presses the “Checkers Leaderboard” button

Scenario:

1. User is in the main menu interface
2. User selects the “Leaderboard” button from the menu
3. User selects the “Checkers Leaderboard” button

Postconditions: Display of leaderboard for Checkers made available for user.

Exceptions:

1. “Checkers Leaderboard” button is unresponsive
2. System fails to retrieve information regarding Checkers leaderboard data

Priority: High. Access to leaderboard is crucial for user engagement and competitive play.

When available: Within 1st iteration.

Frequency of use: Whenever user wants to view leadership information for Checkers.

Channel to actor: “Checkers Leaderboard” button after selecting “Leaderboard” option

Secondary actors: N/A.

Channel to secondary actors: N/A.

Open issues: N/A.

Use Case: Choose Checkers

Iteration: 1

Primary Actor: User

Goal in context: Allows user to play Checkers game if that is the game they want to play.

Preconditions: The user is in the main interface, they are on the interface to choose a game, Checkers is available to play for the user.

Trigger: User presses the “Choose Checkers” button

Scenario:

1. User is in the game menu
2. User selects the “Choose Checkers” option from the list of available games to play

Postconditions: Display of Checkers grid will be available for user to play.

Exceptions:

1. Checkers game fails to load due to technical issues
2. Interface is unresponsive when Checkers is selected

Priority: High. Users should be able to choose Checkers as an available game option.

When available: Within 1st iteration.

Frequency of use: Whenever user wants to play Checkers.

Channel to actor: “Choose Checkers” option as one of the games available for the user to select in the game menu.

Secondary actors: N/A.

Channel to secondary actors: N/A.

Open issues: N/A.

Use Case: Setup Game

Iteration: 1

Primary Actor: User

Goal in context: Sets up game necessary for user to engage in.

Preconditions: The user has selected the game they want to play from the options available and has started a match

Trigger: User has an active game match session.

Scenario:

1. User has started a game session after matchmaking finished
2. The interface displays and loads the game environment
3. User is presented with the game board they chose and resources to play

Postconditions: Game is fully set-up for user to engage in.

Exceptions:

1. Game fails to set-up properly through configuration errors

2. Interface for game mode is dysfunctional

Priority: High. Game needs to be set-up in order for gameplay to begin.

When available: Within 1st iteration.

Frequency of use: Once per active game session

Channel to actor: Through the selective game environment of their preferred game.

Secondary actors: N/A.

Channel to secondary actors: N/A.

Open issues: N/A.

Use Case: Send Message in Chat

Iteration: 1

Primary Actor: User

Goal in context: Allows communication between users in-game.

Preconditions: User has entered the game environment they have chosen after starting a match and game has been set-up successfully with the chat feature.

Trigger: User enters a message into the chat field and sends it to the entire game room.

Scenario:

1. User has entered the game session after matchmaking is complete
2. The game environment is set-up including the chat feature
3. User types a message into the chat field
4. User presses “Send” button
5. The message is displayed for all players to see inside the game session

Exceptions:

1. Chat feature fails to display from configuration issues
2. Chat feature is unresponsive preventing messages to be sent
3. Messages are not visible to everyone

Postconditions: Message is sent and visible to everyone engaged in the space

Priority: Medium. Non-essential but allows in-game communication between players.

When available: Within 1st iteration.

Frequency of use: Multiple times within one game session.

Channel to actor: Chat field within chosen game session.

Secondary actors: N/A.

Channel to secondary actors: N/A.

Open issues: N/A.

Use Case: Player Move

Iteration: 1

Primary Actor: User

Goal in context: Allows player to take their turn

Preconditions: Game environment has been properly set-up with players in session

Trigger: User initiates a move when it is their turn

Scenario:

1. User has entered the game session after matchmaking is complete
2. The game environment is set-up properly
3. It has indicated that it is the user’s turn
4. User makes their move (ex. moving a piece, selecting a grid space)

Exceptions:

1. Move failed to process
2. User turn-taking is dysfunctional
3. Interface is unresponsive

Postconditions: Player’s move is processed and the game state is updated accordingly

Priority: High. Allows active gameplay to happen.

When available: Within 1st iteration.

Frequency of use: Once per turn iteration.

Channel to actor: Game environment interface where moves can be made

Secondary actors: N/A.

Channel to secondary actors: N/A.

Open issues: N/A.

Use Case: Show Win Alert

Iteration: 1

Primary Actor: User

Goal in context: Alerts user if a player has won the game (either user or opponent)

Preconditions: User is in game session and a win condition is met by either user or opponent

Trigger: Game detects that a player has met the win condition after a move

Scenario:

1. User is in-game session
2. Game is set-up
3. User makes move when it is their turn
4. Game logic processes the move to check for win condition
5. If win condition met, then a win alert is displayed and the match is ended

Exceptions:

1. Win condition fails to process
2. Win alert fails to display on screen

Postconditions: User is made aware of win condition via alert and the match is concluded

Priority: High. Should indicate to user if a win condition is met to end the match

When available: Within 1st iteration.

Frequency of use: Once per game session when win condition is detected

Channel to actor: A win alert is displayed on the screen for user to view

Secondary actors: N/A.

Channel to secondary actors: N/A.

Open issues: N/A.

Use Case: Show Loss Alert

Iteration: 1

Primary Actor: User

Goal in context: Alerts user if a player has lost the game (either user or opponent)

Preconditions: User is in game session and a loss condition is met by either user or opponent (the other player has won)

Trigger: Game detects that a player has met the win condition after a move

Scenario:

1. User is in-game session
2. Game is set-up
3. User makes move when it is their turn
4. Game logic processes the move to check for win condition
5. If win condition met for either player, then the other player receives a loss

Exceptions:

1. Win condition fails to process
2. Loss alert fails to display on screen

Postconditions: User is made aware of loss via alert and the match is concluded

Priority: High. Should indicate to user if a loss is met to end the match

When available: Within 1st iteration.

Frequency of use: Once per game session when win condition is detected

Channel to actor: A loss alert is displayed on the screen for user to view

Secondary actors: N/A.

Channel to secondary actors: N/A.

Open issues: N/A.

Use Case: Match Ends

Iteration: 1

Primary Actor: User

Goal in context: Shows to user that a match is concluded after an outcome is met

Preconditions: User is in game session and a win condition has been met by either the user or the opponent

Trigger: Game detects that a game has ended via player win

Scenario:

1. User is in a game session
2. Game is set-up
3. Game logic has processed that a win condition has been met by either player after a turn
4. The system concludes the match, showing appropriate alert to the user

Exceptions:

1. Win condition fails to process
2. Loss alert fails to display on screen

Postconditions: Match has ended and user can play again or return to the game menu

Priority: High. Match needs to conclude after outcome reached and user can select next action

When available: Within 1st iteration.

Frequency of use: Once per game session when win condition is detected

Channel to actor: Match conclusion is displayed to the user on the screen

Secondary actors: N/A.

Channel to secondary actors: N/A.

Open issues: N/A.

Use Case: Play Again Button

Iteration: 1

Primary Actor: User

Goal in context: Allows user to play again if they so choose to after a match has ended

Preconditions: A match has concluded based on either player reaching the win condition

Trigger: User presses the “Play Again” button after a match has finished

Scenario:

1. The system detects a win condition has been met
2. The match concludes for the user
3. The user is provided with the option to play again

Exceptions:

1. “Play Again” button is unresponsive
2. Failure to reset the game environment

Postconditions: User is brought back to game set-up screen to open a new match

Priority: High. Provides user with intuitive option to continue playing after a match has ended without going back to the main menu

When available: Within 1st iteration.

Frequency of use: Once per game session after match has concluded

Channel to actor: Through the “Play Again” button in the interface

Secondary actors: N/A.

Channel to secondary actors: N/A.

Open issues: N/A.

Use Case: Return to Game Menu

Iteration: 1

Primary Actor: User

Goal in context: Allows user to return back to the game menu after a match has concluded to access other options

Preconditions: A match has concluded based on either player reaching the win condition

Trigger: User indicates they want to go back to the game menu by selecting “Return to Game Menu”

Scenario:

1. The system processes that a win condition has been met
2. The match is concluded with displays of win/loss to user
3. User is presented with the option to return to the game menu

Exceptions:

1. Button is unresponsive
2. Failure to load the main menu upon input

Postconditions: User is taken back to the game menu upon choosing to return

Priority: High. Provides user with a way to exit a concluded match and to navigate back to the main menu.

When available: Within 1st iteration.

Frequency of use: Once per game session after match has concluded

Channel to actor: “Return to Game Menu” button available for user to select after a match has concluded

Secondary actors: N/A.

Channel to secondary actors: N/A.

Open issues: N/A.