Use Case 1: View Leaderboard

* Iteration: 1
* Primary actor: User
* Goal in context: User wants to view the leaderboard of a game
* Preconditions:
  + The user is logged in to the application
  + A game has been selected
* Trigger: user open the leaderboard section and selects a game
* Scenario:

1. User selects a game
2. System gets the leaderboard data from the game that user has selected
3. System sorts the players by points/wins in descending order.
4. System displays the sorted list of players

* Postconditions:
  + The leaderboard is displayed with the ranking of players.
* Exceptions:
  + If there is no data to display
  + If user is not logged in
  + If app is not running
* Priority: High
* When available: Iteration 2
* Frequency of Use: High (user checks ranking often)
* Channel to actor: Application GUI
* Secondary Actors: None
* Channels to Secondary Actors: N/A
* Open Issues:
  + How often should leaderboard update?
  + Should leaderboard constantly update while app is running?

Use Case 2: Interacting with the Leaderboard System Basic (ILSB)

* Iteration: 1, started March 4 by Puneet Dhawan
* Primary Actor: Player, Alex
* Goal in context: To interact with the leaderboard system, including viewing rankings, updating score, and testing system limits.
* Preconditions:
  + The leaderboard system is operational and accessible.
  + Players have valid accounts with scores stored in the system.
  + The system can process updates and display ranking correctly.
* Trigger:
  + Alex completes a game and attempts to interact with the leaderboard,
* Scenario:

1. Alex finishes a game session and earns a new score.
2. The system automatically updates Alex’s score in the database.
3. Alex navigates to the leaderboard screen.
4. The system retrieves and displays the latest rankings.
5. Alex sees their updated ranking, verifies their score is correct, and is happy that he is higher than his rival now.
6. Alex logs out

* Post conditions:
  + The leaderboard accurate reflects the updated score.
* Exceptions:
  + Delay in output/new updates to leaderboard ranking/score
  + Network issue
* Priority:
  + Medium, as leaderboard enhance player engagement but do not affect core gameplay.
* When available:
  + Iteration 4
* Frequency of Use:
  + Regular, depending on player activity.
* Channel to actor:
  + App Interface (in-game leaderboard menu)
* Secondary actors:
  + Game server (processes score updates and retrieves rankings)
* Channel to secondary actors:
  + Game database (scores and retrieves player scores)
  + LeaderBoardUpdater
* Open issues:
  + How long until the system will update the scores of the player?
  + How would the ranking system handle players with the same score/ranking?

Use Case 3: Automatically Update Leaderboard when opening app

* Iteration: 1
* Primary actor: System
* Goal in Context: System automatically updates the leaderboard with the new results
* Preconditions:
  + User must be logged in
  + User navigates to leaderboard page
* Trigger: The application is opened or when the leaderboard page is opened (Fix later)
* Scenario:

1. Text
2. Text
3. Text

* Postconditions: High
* Exceptions: Iteration 2
* Priority:
* When available:
* Frequency of use:
* Channel to actor:
* Secondary actors:
* Channels to secondary actors:
* Open issues:

Use Case 4: Update leaderboard when update button is pressed in leaderboard page

* Iteration:
* Primary actor:
* Goal in Context: User can manually update the leaderboard page to show new results
* Preconditions:
  + Text
* Trigger: User presses the update button in leaderboard page
* Scenario:

1. Text
2. Text
3. Text

* Postconditions:
* Exceptions:
* Priority:
* When available:
* Frequency of use:
* Channel to actor:
* Secondary actors:
* Channels to secondary actors:
* Open issues:

Use Case 5: Search Leaderboard for player names

* Iteration:
* Primary actor:
* Goal in Context: User can find a specific user’s wins/ranking
* Preconditions:
  + Text
* Trigger:
* Scenario:

1. Text
2. Text
3. Text

* Postconditions:
* Exceptions:
* Priority:
* When available:
* Frequency of use:
* Channel to actor:
* Secondary actors:
* Channels to secondary actors:
* Open issues:

Use Case 6: Filter Leatherboard

* Iteration:
* Primary actor:
* Goal in Context: User wants to filter the leaderboard using a specific criteria
* Preconditions:
  + Text
* Trigger:
* Scenario:

1. Text
2. Text
3. Text

* Postconditions:
* Exceptions:
* Priority:
* When available:
* Frequency of use:
* Channel to actor:
* Secondary actors:
* Channels to secondary actors:
* Open issues:

Use Case 7: Change Leaderboard filter

* Iteration:
* Primary actor:
* Goal in Context: User can change the filter on the leaderboard.
* Preconditions:
  + Text
* Trigger:
* Scenario:

1. Text
2. Text
3. Text

* Postconditions:
* Exceptions:
* Priority:
* When available:
* Frequency of use:
* Channel to actor:
* Secondary actors:
* Channels to secondary actors:
* Open issues: