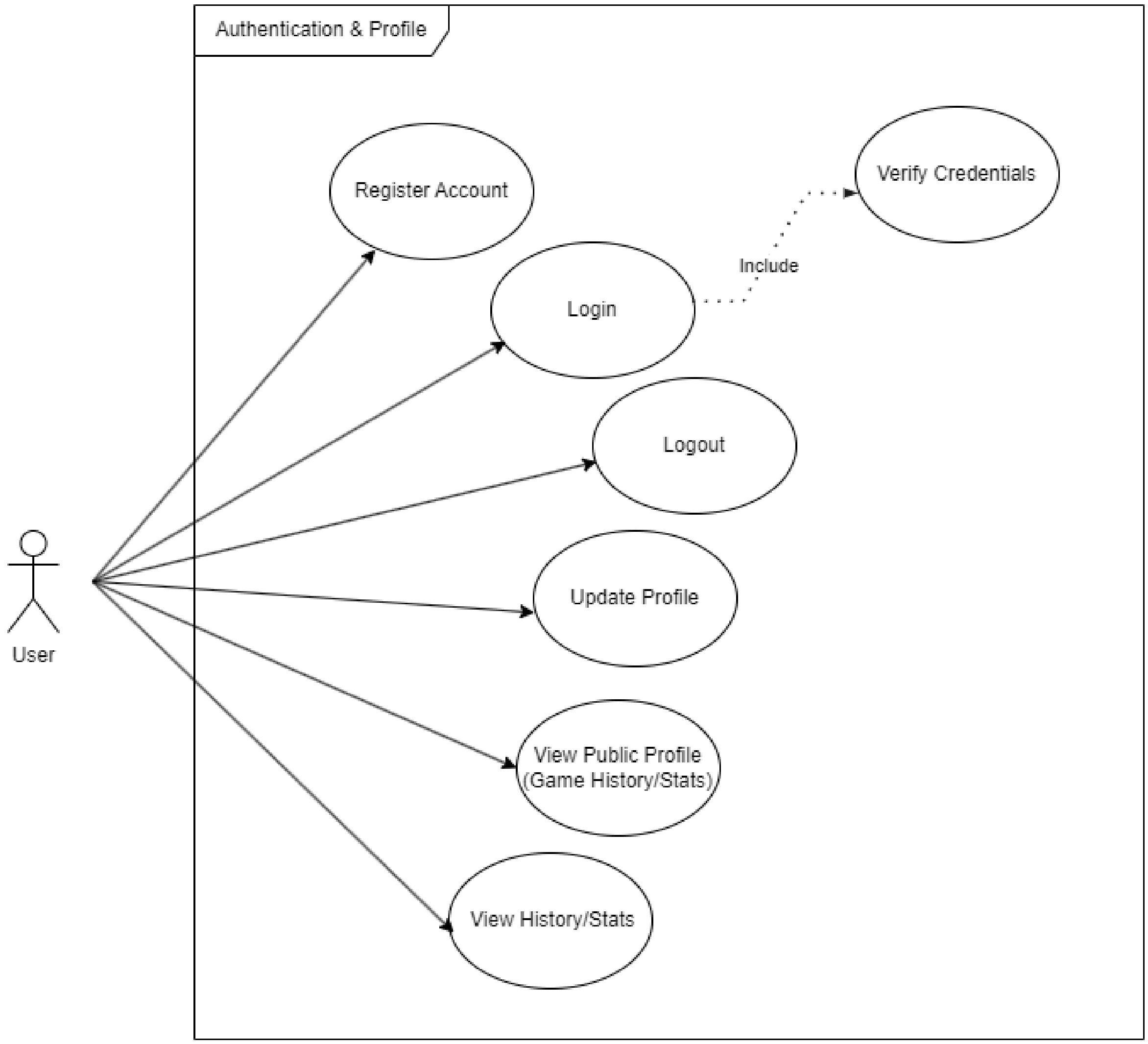

Use Case: User Registration

Iteration: 1

Primary Actor: User

Goal in Context: Register a new account.

Preconditions: User is on the User Registration GUI

Trigger: User clicks on the “Register” button

Scenario:

1. The user provides username and password, email
2. System makes sure username is unique
3. System stores this new username and password in the database.

Postcondition: User Registration is complete, an account has been created.

Exceptions:

- Username/email (optional) already exists

Priority: High. As it’s crucial to have an account to play games.

When available: Increment 1

Frequency of use: High

Channel to actor: GUI

Secondary actors: Database

Channel to secondary actors: Database/API?

Open issues:

- Should emails be mandatory in the process of user registration?

Use Case: User Login

Iteration: 1

Primary Actor: User.

Goal in Context: Login and access the application.

Preconditions: User account exists.

Trigger: User clicks the Login button.

Scenario:

1. User enters username and password
2. System Verifies Credentials (See Verify Credentials Usecase)
3. User has access to the platform.

- Incorrect password and/or username doesn’t exist. (See Verify Credentials Use Case)

Priority: High

When available: Increment 1

Frequency of use: Frequent.

Channel to actor: GUI

Secondary actors: Database

Channel to secondary actors: Database

Open issues: (none)

Use Case: Verify Credentials

Iteration: 1

Primary Actor: Database

Goal in Context: Verify account exists and login information is correct

Preconditions: User has entered username and password

Trigger: User clicks the Login button.

Scenario:

1. Username is located in database
4. System confirms password entered matches password associated with the username
5. User is granted access

Postcondition: User’s credentials have been verified

Exceptions:

- Username doesn’t exist, login denied
- Password doesn’t match, login denied

Priority: High

When available: Increment 1

Frequency of use: Frequent.

Channel to actor: GUI

Secondary actors: (none)

Channel to secondary actors: (none)

Open issues: (none)

Use Case: View Public Profile

Iteration: 1

Primary Actor: Registered User

Goal in Context: Be able to View profile details and game history (Wins and losses).

Preconditions: User is logged in.

Trigger: User clicks on “View Profile”

Scenario:

1. Users are able to view their profile.
2. System shows the game history (Wins, losses, games played, player statistics) and other relevant information

- GUI issues
- Data issues.

Priority: Medium

When available: Iteration 1

Frequency of use: Frequent

Channel to actor: GUI

Secondary actors: Database

Channel to secondary actors: Database

Open issues:

- How much can the user see?

Use Case: Logout

Iteration: 1

Primary Actor: Registered User

Goal in Context: Log out

Preconditions: User is logged in.

Trigger: User clicks Logout button

Scenario:

1. User clicks the Logout button
2. System logs the user out
3. User is redirected to login page

Postcondition: User is logged out

Exceptions:

- Data issues

Priority: High

When available: Iteration 1

Frequency of use: Frequent

Channel to actor: GUI

Secondary actors: Data base?

Channel to secondary actors: Data base?

Open issues:

- Should there be a confirmation popup before logging the user out?

‘

Use Case: Update Profile

Iteration: 2

Primary Actor: Registered User

Goal in Context: Update profile information

Preconditions: User is logged in

Trigger: User clicks on update profile option

Scenario:

1. Profile details are visible to the user
2. User can select which details they would like to update
3. User can update their profile information details
4. User clicks save

- User clicks cancel/exit and profile updates are not

saved **Priority**: Medium

When available: Iteration 2

Frequency of use: Occasionally

Channel to actor: GUI

Secondary actors: Data base?

Channel to secondary actors: Data
base? **Open issues**:

- What profile details are visible?

Use Case: View Public Game History/Statistics

Iteration: 1

Primary Actor: Registered User

Goal in Context: View public game history/statistics

Preconditions: User is logged in

Trigger: User clicks Player Hub option

Scenario:

1. Current leaderboard is displayed on screen
2. List of players is displayed on screen
3. User can select player profile to view
4. Public player information, status, game history and ranking is displayed on screen

Postcondition: Public player information is displayed

Exceptions: (none)

Priority: Medium

When available: Iteration 2

Frequency of use: Frequent

Channel to actor: GUI

Secondary actors: Data base

Channel to secondary actors: Data base

Open issues:

- How much profile information is available to other players?