1. **Register New user:** A user arrives at a registration page and is prompted to enter an email (which the system verifies is unique), a password, and a nickname (which the system verifies is unique). If successful, user is registered, otherwise user is prompted for information again.
   Primary Actor: Unregistered User
   Stakeholders and Interests: Unregistered User - wants to be able to register with an email, nickname & password.
   Preconditions: Unregistered user is not registered.
   Postconditions: Account is created with a unique email/nickname.
   Success Scenario:
       1. Unregistered user arrives on signup/login page
       2. Unregistered user enters password and unique email and nickname
       3. System verifies uniqueness of email and nickname and stores user information
       4. Unregistered user becomes registered user
   Extensions:
      a.
          i.   System fails to load signup/login page
          ii.   User retries until successful or contacts system admins
      b.
          i.   Unregistered user enters non-unique email and nickname
          ii.   System returns error message to user and prompts them to enter a different email/nickname
          iii. Process repeats until success
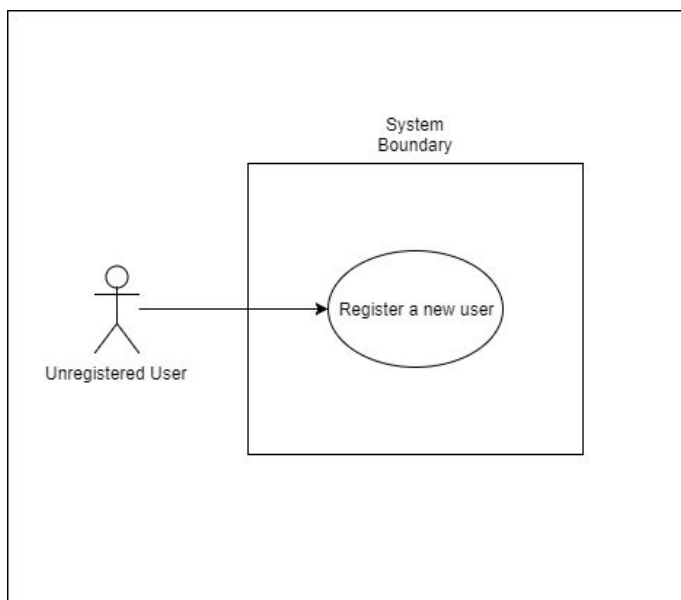
2. **Create new game:** A registered user arrives on a landing page where they are given the option to create a new game. If they create a new game they then become a participant in the newly created game.
   Primary Actor: Registered User

Stakeholders and Interests: Registered User - wants to be able to create a new game and be included in that game immediately

Preconditions: User has created an account and is logged into the system

Postconditions: A new game is started and the game is in its initial stage with the user as a player

Success Scenarios:
1. Registered user arrives at a main page.
2. Registered user clicks a button to start a new game.
3. A new game is created and that game is in its initial starting stage

Extensions:
a. User is not directed to the landing page after he/she logs into the system
b. When the user clicks on the button to create a new game, a new game is not started
c. System fails at any point
   i. System restarts
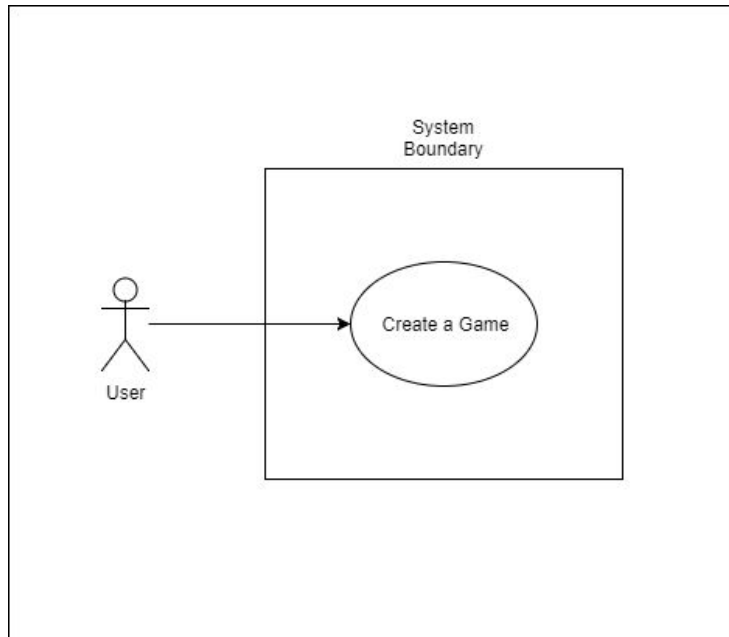
3. **Invite player:** A registered user who has created a game is given the option to invite another registered users to their created game by using an email or nickname.

Primary Actor: Registered User

Stakeholders and Interests: Registered User - wants to be able to invite one or more players to the game that he/she just created.

Preconditions: User has logged into the system and successfully created a new game

Postconditions: User requests other users to join the game and those users are added to that game

Success Scenarios:

1. After starting a game, the user is prompted to enter one or more users to join the game.

Extensions:
    a. System crashes at any point in this process
    b. System is restarted
    c. User is not presented with a window to request other users once the game is created
    d. User requests users that are not currently logged in.
    e. User is presented with an error message saying the users he/she requested are not logged into the system
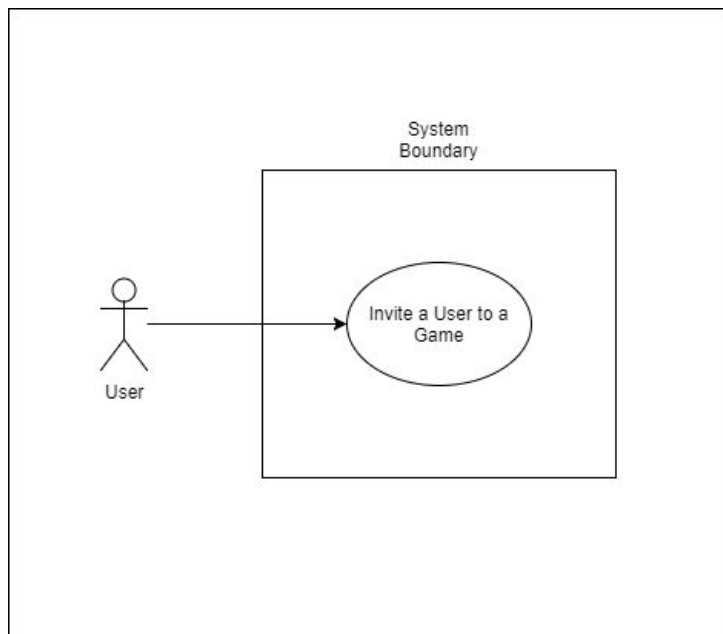
4. **Manage Invitations:** A registered user will receive a notification when they have been invited to a created game. The registered user has the option of accepting or rejecting the invitation to play. If the user accepts the invitation they are added as a player to the game, otherwise the notification disappears.

Primary Actor: Registered User

Stakeholders and Interests: Registered User - wants to be notified when he/she has been added to a new game by another user and respond to that invitation by either accepting or rejecting. If he/she accepts the invitation, they are added to the game.

Preconditions: User has registered for the system and is currently logged in

Postconditions: A message is shown to the user and they can either click an accept button or a reject button. If they accept, they are added to the game, but if they reject, they are returned to the landing page.

Success Scenarios:
1. A message is shown to a user if another registered user invites them to a game
2. That user has the option of either accepting or rejecting the invitation
3. If he/she accepts, they are added to the game

4. If he/she rejects, they are not added to the game and are returned to the landing page

Extensions:
   a. The system crashes at any point during this process
   b. A user is not notified with a message when another user invites them to a game
   c. A user in not added to a game when they accept an invitation from another user
   d. A user is not returned to the landing page after rejecting an invitation
   e. A user is added to a game even though they rejected an invitation
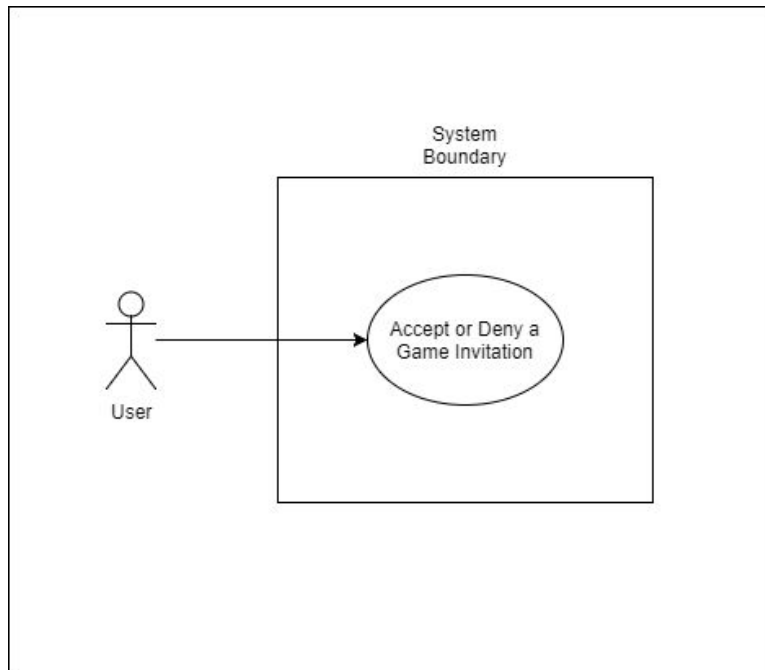
5. **Choose Game:** A registered user has a personal dashboard where they may choose to play between any of their already created games.
   Primary Actor: Registered User
   Stakeholders and Interests: Registered User - wants to be able to see all of the games that he/she is currently a part of.
   Preconditions: User is logged into the system and a part of one or more games
   Postconditions: A window is present that shows each of the games a user is involved in. That user can click on any of these games to switch from one game to another.
   Success Scenarios:
      1. A window is located at the side of user's screen that shows which games the user is a player in.
      2. The user is allowed to click on any of these games to change to that game.
   Extensions:
      a. The system crashes at any point during this process
      b. System is restarted
      c. While in one or more games, the window is not present to the user
      d. The user is not taken to another game once he/she has clicked on that game in the window.
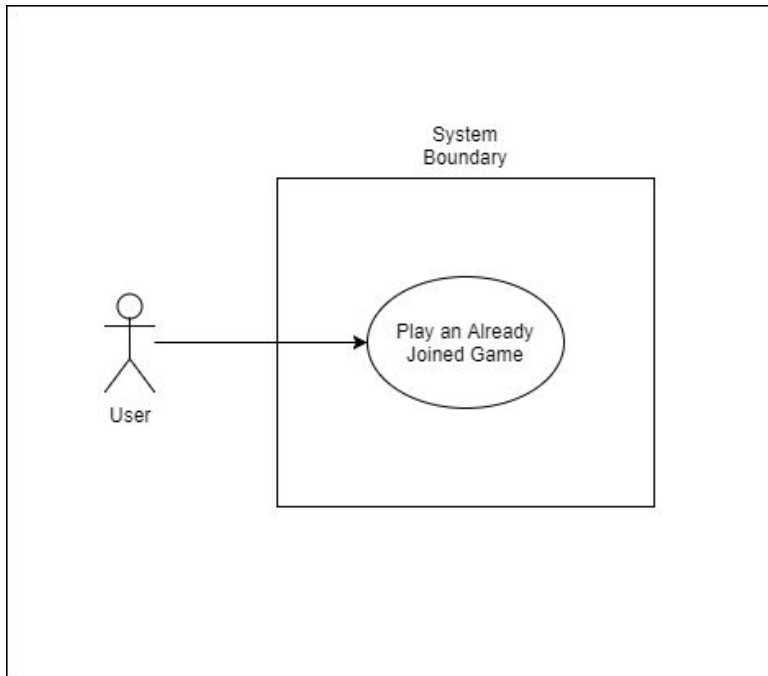
6. **Game Isolation:** Any user not participating in some given game may not interact with that game.

Primary Actor: Registered User

Stakeholders and Interests: Registered User - wants users outside a game to not be able to interact with that game.

Preconditions: A game has been created and users have joined

Postconditions: Only users that have been included in the game can participate in the game

Success Scenarios:
  1. Users that are not involved in a game can not participate or interact with that game

Extensions:
  a. System crashes at any point during this process
  b. System restarts
  c. Users that are outside of a game can participate in that game

7. **Quit Game**: A player of some game may quit that game via their dashboard.

Primary Actor: Player

Stakeholders and Interests: Player - wants to be able to quit a game that they are currently involved in.

Preconditions: A user is currently involved in a game

Postconditions: That user is not involved in the game that they just quit out of

Success Scenarios:
  1. A button is shown on a user's screen so that he/she can quit out of a game at any point.

2. Once a player clicks that button, a popup is presented asking the user if he/she is sure they want to quit the game.
3. If user clicks yes, they are returned to the landing page
4. If user clicks no, they stay in the current game

Extensions:
    a. System crashes at any point during this process
    b. System restarts
    c. Player is not taken out of a game when they decide to quit
    d. Player is taken out of a game that they did not want to quit
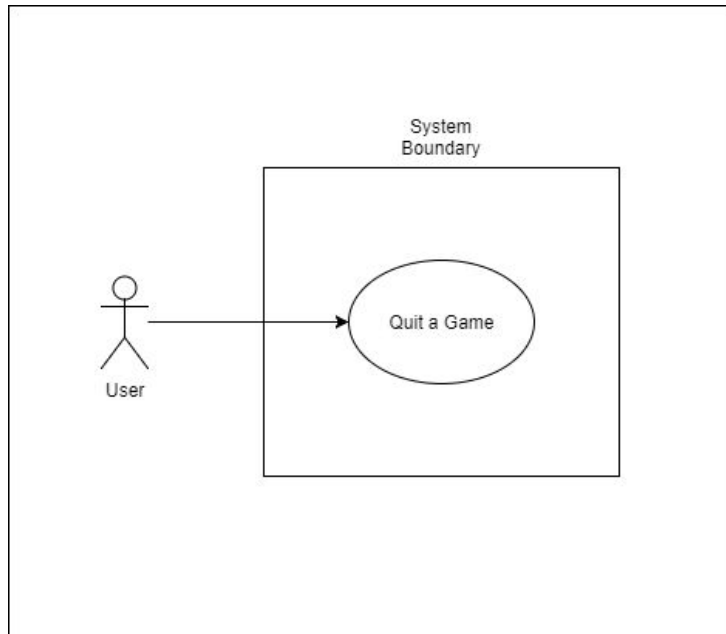
8. **Deregister User:** A registered user can deregister from the system at any point. In which case, their games are deleted and any opponents they had at the time will receive a notification of this. In addition, their email and nickname are freed up for use by other people wanting to register.

Primary Actor: Registered User

Stakeholders and Interests:
- Registered User - wants to deregister from the system
- Other registered users in games with the primary actor - want to only have playable games on their account
- Current unregistered users who will register in the future - want to be able to use any email/nickname previously taken by the deregistered user

Preconditions:
- Registered User is registered with the system

Postconditions:
- Registered User becomes deregistered from system
- All games associated with registered user deleted from system
- All previous opponents notified of deleted games

- Previously registered user's nickname and password are freed up for future use

Success Scenario:
1. Registered user selects option to deregister their account
2. User receives email that they have been deregistered from the system
3. Previously registered user's opponents receive in-game notification that their games with said user have been deleted and will no longer show up in their account

Extensions:
a. System fails at any point
    i. User either reloads the page until success or contacts system admins for help
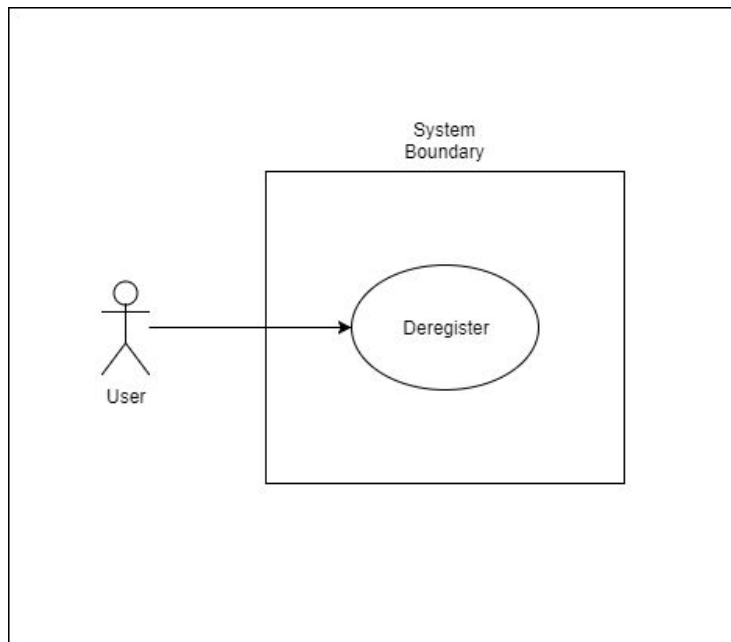
9. **View History**: A registered user may look over the history of all their games played via their dashboard. Their history includes information such as the opponent against whom they played, the games start/end date/times, and the end result of the game.
Primary Actor: Registered User
Stakeholders and Interests:
- Registered User - wants to be able to view game history

Preconditions: User has registered with the system
Postconditions: User can see game-history
Success Scenarios:
1. Registered User clicks on the game-history link from their dashboard
2. Screen displays game-history for the user which includes game opponent, game start and end times, and game winner for each game played

Extensions:
a. Registered user has no games played

i.   Screen displays that the user has not started any games and encourages them to begin one

b.   System fails at any point

i.   User reloads page until successful or contacts system administrator for help
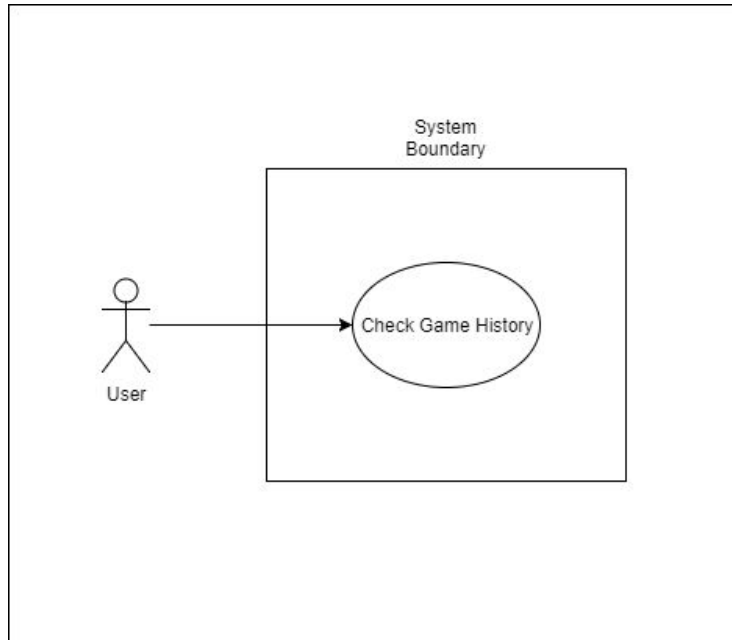
10. **View Profile**: A registered user may access their or another's profile from the dashboard, seeing their nickname and a history of their played games.

Primary Actor: Registered User

Stakeholders and Interests:
- Registered User - wants to be able to see another player's game history
- Other Registered User - wants to show off their game history

Preconditions:
- User is registered with the system
- Other users are registered with the system

Postconditions:
- Registered users are able to see each other's game history

Success Scenario:
1. Registered user searches for another user by nickname or email, selects their profile and can view their game history

Extensions:
a. Registered user searches for an email or nickname that is not registered with the system

i. Screen displays text that informs user no account exists that is associated with the email/nickname
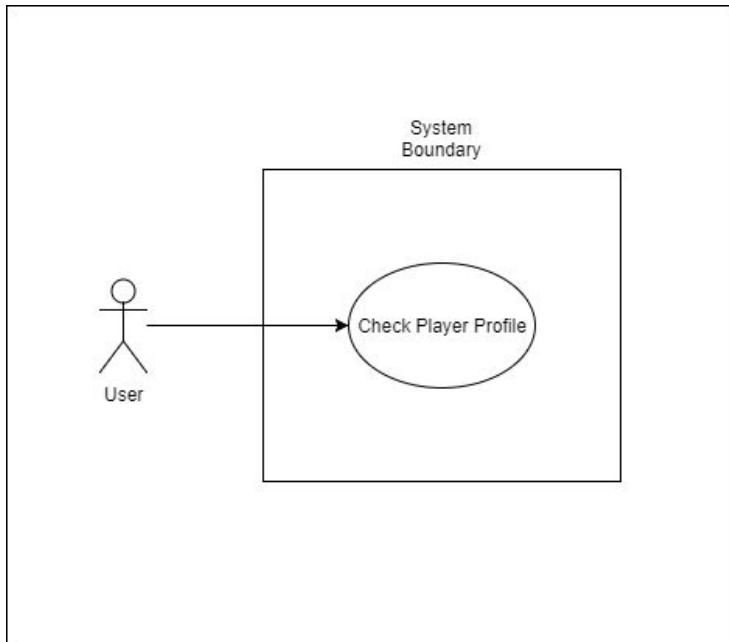
11. **Start Game:** The system does not start the game until a second player has joined or the game is against a bot player.
Primary Actor: System
Stakeholders and Interests:
- Player - wants to play a game only if another player is present

Preconditions:
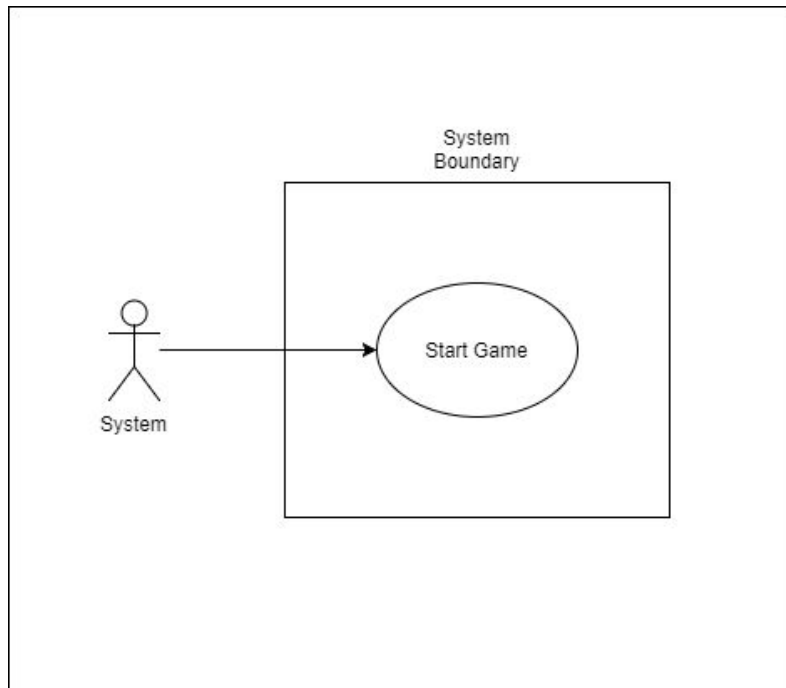- A game has been created

Postconditions:
- Game is started between two players

Success Scenarios:
1. Player creates a game
2. Player invites another player to join game
3. Other player joins game
4. Game begins

Extensions:
a. Player invites another player to join the game and the other player never accepts the invitation
   i. Game is never started

12. **Game full:** A registered user may invite multiple players to the same game, however, only one invitation may be accepted.

Primary Actor: Registered User

Stakeholders and Interests:
- Registered user - wants to play against only one player
- Invited Players - want to play against only one player

Preconditions:
- Game has been created

Postconditions:
- One player has joined created game and game then starts

Success Scenarios:
1. Player invites many players to a created game
2. One of the invited players accepts the invitation to play
3. All other users' invitations are rescinded
4. The game begins
- OR -
1. Player invites one player to a created game
2. The invited player accepts the invitation to play
3. The game begins

Extensions:
a. Player doesn't invite players to a created game
   i. Game does not begin
b. Multiple players accept invitation
   i. First accepted invitation received by the system is the one used

13. **First Move:** A player in a game plays first in a game that they have created.
    Primary Actor: System
    Stakeholders and Interests:
    ● Players - expect to have a consistent set of rules where the game creator always plays first
    Preconditions:
    ● Game has been created and started
    Postconditions:
    ● Game creator is given the first move
    Success Scenarios:
    1. Registered user creates a game
    2. Registered user invites user(s) to game
    3. User accepts invitation
    4. Game begins
    5. Game creator receives first move
    Extensions:
    a. System fails at any point
       i. User deletes game and tries again or contacts system admins if failure persists
14. **Next Move:** The system permits the game creator to make the first move then the opponent is permitted to move.
    Primary Actor: Game Creator
    Stakeholders and Interests: Game Creator - makes the first move. Opponent - makes the next move.
    Preconditions: The game creator is the registered user who has created the game; the opponent is a registered user who didn't make the game; no move has occurred in the game yet.
    Postconditions: The game creator is allowed to make the first move.
    Success Scenario:
    1. Game creator sees on their dashboard that they are able to make the first move in the game they have created.
    Extensions:
    a.
       i. System does not allow game creator to play first.
       ii. Game creator contacts system admin.
    b.
       i. System does not allow game creator to play at all.
       ii. Game creator contacts system admin.
    c.
       i. System lets opponent make the first move.
       ii. Game creator or opponent contacts the system admin to have the game reset with the game creator making the first move.

15. **Player Makes a Move** | The system allows users to only make moves in games that they are currently included in.
    Primary Actor: Player
    Stakeholders and Interests: Player - wants to make a move in a game in which he is a player.
    Preconditions: Player is a participant in some game that has not yet ended.
    Postconditions: Player is allowed to make a move in such a game.
    Success Scenario:
      1. Player sees on their dashboard that they can make a move in a game.
    Extensions:
      a.
        i. Player cannot make a move in a game in which they are included.
        ii. Player contacts system admin to resolve this issue.
16. **Players Make Moves** | The system only allows player to make a move if it their turn to play.
    Primary Actor: Player
    Stakeholders and Interests: Player - wants to make a move in a game.
    Preconditions: The player is included in some game in which it is their turn to move.
    Postconditions: The player successfully makes a move.
    Success Scenario:
      1. Player loads a game that they wish to play into their system.
      2. Player is allowed to make a move in the game.
    Extensions:
      a.
        i. Player is unable to make a move when it is their turn to play.
        ii. Player contacts system admin to resolve this issue.
      b.
        i. Player is able to make a move when it is not their turn to play.
        ii. Player contacts system admin to resolve this issue.
17. **Player Makes Valid Move** | The system allows users that are enrolled in a game to only make moves that are legal for the game. An illegal move is disregarded and an error message is shown by the system.
    Primary Actor: User
    Stakeholders and Interests: User - wants to make a move.
    Preconditions: User is enrolled in a game.
   Postconditions: User makes a move in such a game.
    Success Scenario:
      1. User sees a game in which they are enrolled.
      2. User attempts to make a valid move in a game in which they enrolled.
      3. Uer makes a move.
    Extensions:
      a.
        i. User is not enrolled in a game.

        ii.   User enrolls in a game.

    b.

        i.   User is enrolled in a game, but attempts to make an illegal move.

        ii.   Move is disregarded and an error message is shown.
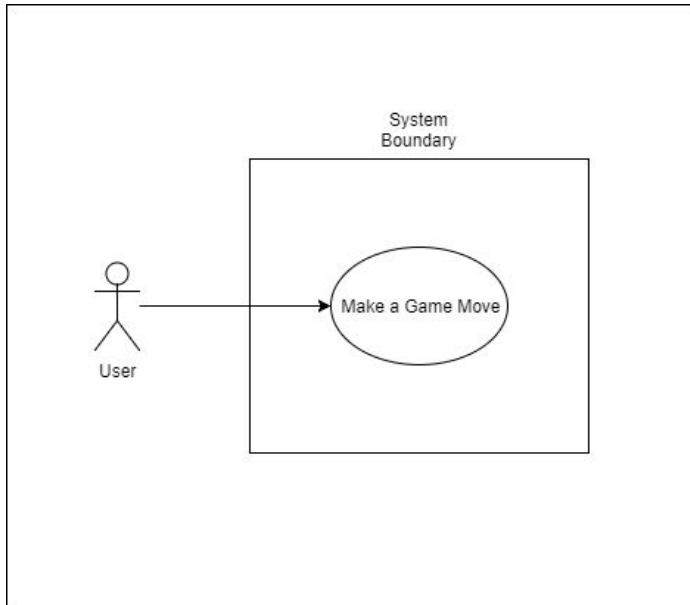
18. **Game Saved** | The system saves the state of the game, allowing player to play asynchronously.

Primary Actor: Player

Stakeholders and Interests: Player - wants to play asynchronously.

Preconditions: System has a game in which the player is enrolled that has begun.

Postconditions: System saves the state of the game, allowing the player to play asynchronously.

Success Scenarios:

    1. System saves the state of a game.

    2. Player loads the state of the game

Extensions:

    a.

        i.   System fails to save the state of the game, or the player is not allowed to load the state of a saved game and perform a move on it.

        ii.   System admin resets the system and fixes the problem.
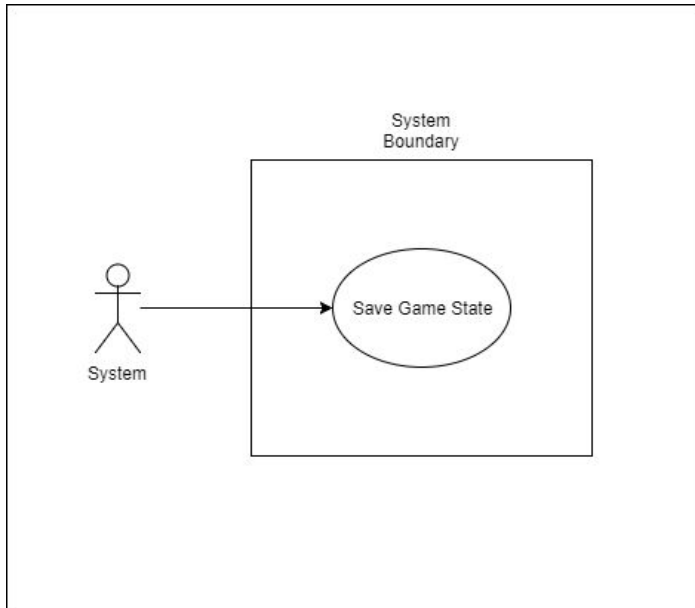
19. **Game Ends** | The system determines when a game has ended and also determine who is the winner and loser.

Primary Actor: System

Stakeholders and Interests: System - wants to determine when a game is ended and who has won and who has lost.

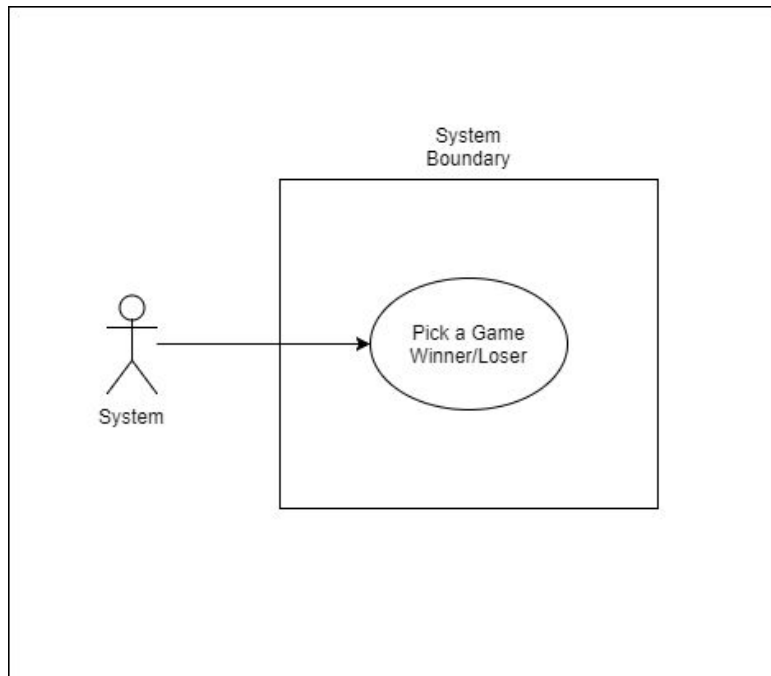Preconditions: A game has been played and has reached an end state.

Postconditions: The winner and loser are chosen.
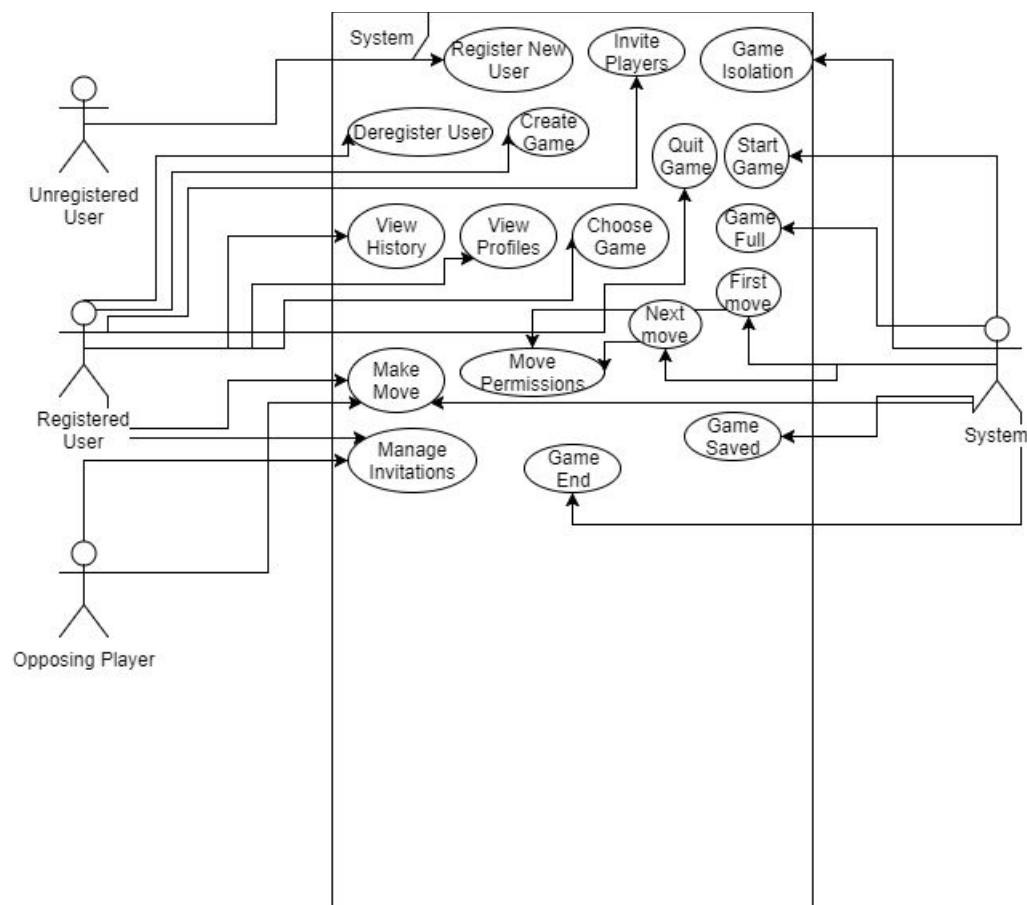
Success Scenarios:

1. System determines the winner and loser of an ended game.

Extensions:

a.

    i. System fails to determine the winner and loser of an ended game.

    ii. System is fixed by a system administrator.

Complete use case diagram...