Use Case: Sending a Friend Request

Iteration: 1, Initial version.

Primary Actor: User (Player)

Goal in Context: To allow users to send friend requests to other players in order to connect, play together, and interact on the OMG platform.

Preconditions:

- The user is logged into the OMG platform.
- The user has a valid account and is active on the platform.
- The user is able to search for or browse other players' profiles.

Trigger:

- The user wants to add another player to their friends list to connect and play together.

Scenario:

1. The user navigates to the profile or search section to find the player they wish to send a friend request to.
2. The system displays a list of available players or a search function for the user to enter a username or email.
3. The user selects the player they want to send a friend request to.
4. The system displays the selected player's profile, showing their current status (online/offline), games played, and other relevant information.
5. The user clicks on the "Send Friend Request" button on the profile.
6. The system sends the friend request to the selected player and notifies the user that the request has been sent successfully.
7. The system notifies the selected player of the incoming friend request, which they can accept or decline.
8. If the request is accepted, the system adds the users to each other's friends list, and they are able to see each other's online status and send game invites.

Exceptions:

- **Friend Request Already Sent**: If the user has already sent a friend request to the player, the system will display a message stating that the request has already been sent.
- **Player Already in Friends List**: If the user and the selected player are already friends, the system will display a message indicating that they are already connected.
- **User Blocked**: If the selected player has blocked the user, the system will inform the user that they cannot send a friend request.
- **Request Declined**: If the player declines the friend request, the system will notify the user of the declined status.

Priority: Medium priority, as it enhances social interaction but is not essential for the core gameplay.

When Available: First increment.

Frequency of Use: Moderate, as players typically send friend requests to people they want to play with or communicate with more regularly.

Channel to Actor: Via web browser or application.

Secondary Actors:

- **System Administrator**: For monitoring and resolving any issues related to user accounts and the friend request system.
- **Customer Support**: For assisting users with issues related to sending or receiving friend requests (e.g., blocked users, incorrect functionality).

Channels to Secondary Actors:

- **System Administrator**: Admin dashboard for managing system errors and resolving issues with the friend request feature.
- **Customer Support**: Customer support ticket system for troubleshooting and resolving any user complaints or issues related to friend requests.

Open Issues:

- Should there be a limit on how many friend requests a user can send in a day to prevent spam?
- Should friend requests expire if not accepted within a certain period?
- Should the system provide an option to customize friend request messages, or should it be a simple request notification?