# Pokémon League Manager Use Cases

| | |
|---|---|
| **Name** | Register for League. |
| **ID** | UC_001 |
| **Description** | To participate in the league and be able to save their player information and statistics, the user must register their account in the system. |
| **Actors** | League Participant. |
| **Organizational Benefits** | Helps the league moderators keep track of players who are participating in the current season, increasing efficiency in managing the league. |
| **Frequency of Use** | Every player must register first, so it will be used 100% at the start of a season. |
| **Triggers** | User types "=register" in the Discord text server. |
| **Preconditions** | User is a member of the league Discord server. |
| **Postconditions** | The player and their competitors are able to view their player information.<br>The player is able to be selected for matches and subsequently check-in to them once the match is ready to begin.<br>Administrators are able to view every player currently participating in the season. |
| **Main Course** | 1. Player attempts to use the Discord commands.<br>2. Bot checks to see if the user is registered as a player in the league.<br>3. Bot determines the player is not registered and sends an error message directing the user to register first.<br>4. User registers.<br>5. Bot saves the player's data in the database.<br>6. Bot allows the user to use its commands. |
| **Alternate Courses** | AC1. Bot determines the player is already registered.<br>1. Return to Main Course step 6 |
| **Exceptions** | EX1. User is already registered.<br>1. Throw an error message informing the user they have already registered |

| | |
|---|---|
| **Name** | Update Player Info and Statistics. |
| **ID** | UC_002 |
| **Description** | The player and moderators will be able to update player statistics as the season goes on. The statistics include the player's name, wins, losses, win/loss average, kills, deaths, and kill/death average. |
| **Actors** | League Participant and Moderators. |
| **Organizational Benefits** | Helps the league moderators and players have a single place to access and update all of their info as the season goes on, making it so no one has to use an external website to manually update statistics. |
| **Frequency of Use** | As the season goes on, player statistics will regularly be updated, being used 100% of the time over the course of a season. |
| **Triggers** | User types '=': "won", "lost", "killed", or "died". |
| **Preconditions** | User is a member of the league Discord server and has already registered as a participant. |
| **Postconditions** | The player and their competitors are able to update their player information.<br>The player is able to keep track of their progress over the course of a season.<br>Players are able to be sorted into a leaderboard that ranks players based on their statistics. |
| **Main Course** | 1. Player attempts to use the associated commands starting with the prefix "=": name, won [number], lost [number], killed [number], died [number].<br>2. Bot checks to see if the user is registered as a participant.<br>3. If the bot determines the user is not registered as a participant in the league and throws and error informing them and directing them to register.<br>4. User registers. |

| | |
|---|---|
| | 5. Bot updates the player's statistics based on the commands listed above.<br>6. The user is able to successfully update their player statistics. |
| **Alternate Courses** | AC1. Bot determines the player is already registered.<br>1. Return to Main Course step 5 |
| **Exceptions** | EX1. User enters an incorrect command.<br>1. Bot doesn't respond.<br><br>EX2. The user enters an invalid data type, such as entering a word after a command requiring a number after it.<br>1. Bot doesn't update the statistic and instead throws an error message informing the user of the invalid entry. |

| | |
|---|---|
| **Name** | Display and Update Leaderboards |
| **ID** | UC_003 |
| **Description** | The bot will sort players by their win count from greatest to least, and display a leaderboard for all of the players along with their complete stats from left to right. |
| **Actors** | League Participants or Moderators |
| **Organizational Benefits** | Helps the league moderators and players view the current leaders and be able to boast about their stats. This will help automate the leaderboard process instead of the moderators having to write it out themselves. |
| **Frequency of Use** | It will likely be used 20-30% of the time, only when players are interested in others' stats. |
| **Triggers** | User types "=lb" in the Discord text channel. |
| **Preconditions** | User is a member of the league Discord server and there is enough data to display (which is only more than one current player). |
| **Postconditions** | The player and their competitors are able to view the player rankings. |
| **Main Course** | 1. Players type "=lb" into the Discord text channel.<br>2. Bot checks to see if the user is registered.<br>3. Bot sorts the players by wins, from greatest to least.<br>4. Bot sends a message displaying the leaderboard. |
| **Alternate Courses** | AC1. Bot determines the player isn't registered.<br>1. Bot asks the player to register |
| **Exceptions** | EX1. Player either an invalid or incorrect command.<br>1. Bot either doesn't respond, or doesn't perform the lb command. |

| | |
|---|---|
| **Name** | Set Up and Check-In for Matches |
| **ID** | UC_004 |
| **Description** | To set up weekly matches and allow players to check-in for their matches using bot commands. |
| **Actors** | League Participant and Moderators. |
| **Organizational Benefits** | Helps streamline the process of setting up weekly player matches and having players check-in for them. |
| **Frequency of Use** | It will be a weekly process, so it will be used once a week. |
| **Triggers** | Moderator types "=setmatch" into the Discord text channel. |
| **Preconditions** | User is a moderator and there are enough players to set up matches. Players can only check-in at the designated time for their match. |
| **Postconditions** | The moderators successfully schedule all the players for their matches.<br>The player is able to be selected for matches and subsequently check-in to them once the match is ready to begin. |

| | |
|---|---|
| **Main Course** | 1. Moderator enters the "=setmatch" command.<br>2. Bot checks to see if they're registered.<br>3. Bot checks to see if they have a moderator role attached to their User ID.<br>4. Bot randomly selects players from the player data and sets up the matches until there are no more players to pick from. If an extra player is left, they are given a "bye week."<br>5. Bot displays the match schedule. |
| **Alternate Courses** | 1. Scheduled players enter the "=checkin" command.<br>2. Bot checks to see if player is registered.<br>3. Bot checks to see if the player is scheduled.<br>4. Bot checks to see if the player hasn't already checked in.<br>5. Bot flags the player as ready for the match. |
| **Exceptions** | EX1. The player enters an invalid or incorrect command.<br>1. Bot either doesn't execute the right command or doesn't respond at all.<br><br>EX2. Bot determines the player isn't registered.<br>1. Player needs to register.<br><br>EX3. Bot determines the player isn't a moderator.<br>1. If the user is a moderator in the Discord but doesn't have that role, they must ask an existing moderator to give them that role.<br><br>EX4. Bot determines the player isn't scheduled.<br>1. Bot sends an error message informing them. |