# Extremely Casual Game Organizer

## By

Toney Hale

Jacob Lindauer

Name

Name

Table of Contents

[Extremely Casual Game Organizer 1](#_Toc121165500)

[By 1](#_Toc121165501)

[Executive Summary 9](#_Toc121165502)

[Goals 9](#_Toc121165503)

[Stakeholders 9](#_Toc121165504)

[Tools 9](#_Toc121165505)

[Timeline 9](#_Toc121165506)

[User Stories 10](#_Toc121165507)

[Participants 10](#_Toc121165508)

[Player 10](#_Toc121165509)

[Potential Player 10](#_Toc121165510)

[Spectator 11](#_Toc121165511)

[Commentator 11](#_Toc121165512)

[Videographer 11](#_Toc121165513)

[Organizers 12](#_Toc121165514)

[League Admin 12](#_Toc121165515)

[Tournament Admin 13](#_Toc121165516)

[Coach 14](#_Toc121165517)

[Treasurer 14](#_Toc121165518)

[Logistics 15](#_Toc121165519)

[Equipment Manager 15](#_Toc121165520)

[Donor 15](#_Toc121165521)

[Vendor 15](#_Toc121165522)

[Venue Representative 16](#_Toc121165523)

[Site Administrators 17](#_Toc121165524)

[User 17](#_Toc121165525)

[Administrator 18](#_Toc121165526)

[Moderator 18](#_Toc121165527)

[Others 19](#_Toc121165528)

[Score Reporter 19](#_Toc121165529)

[Volunteers 19](#_Toc121165530)

[Use Cases 20](#_Toc121165531)

[Participants 20](#_Toc121165532)

[Player 20](#_Toc121165533)

[Potential Player 27](#_Toc121165534)

[Spectator 28](#_Toc121165535)

[Commentator 30](#_Toc121165536)

[Videographer 33](#_Toc121165537)

[Organizers 35](#_Toc121165538)

[League Admin 35](#_Toc121165539)

[Tournament Admin 50](#_Toc121165540)

[Coach 58](#_Toc121165541)

[Treasurer 65](#_Toc121165542)

[Logistics 66](#_Toc121165543)

[Equipment Manager 66](#_Toc121165544)

[Donor 66](#_Toc121165545)

[Vendor 66](#_Toc121165546)

[Venue Representative 66](#_Toc121165547)

[Site Administrators 66](#_Toc121165548)

[User 66](#_Toc121165549)

[Administrator 66](#_Toc121165550)

[Moderator 66](#_Toc121165551)

[Others 66](#_Toc121165552)

[Score Reporter 66](#_Toc121165553)

[Volunteers 66](#_Toc121165554)

[User Interfaces 67](#_Toc121165555)

[Participants 67](#_Toc121165556)

[Player 67](#_Toc121165557)

[Potential Player 67](#_Toc121165558)

[Spectator 67](#_Toc121165559)

[Commentator 67](#_Toc121165560)

[Videographer 67](#_Toc121165561)

[Organizers 67](#_Toc121165562)

[League Admin 67](#_Toc121165563)

[Tournament Admin 67](#_Toc121165564)

[Coach 67](#_Toc121165565)

[Treasurer 67](#_Toc121165566)

[Logistics 68](#_Toc121165567)

[Equipment Manager 68](#_Toc121165568)

[Donor 68](#_Toc121165569)

[Vendor 68](#_Toc121165570)

[Venue Representative 68](#_Toc121165571)

[Site Administrators 68](#_Toc121165572)

[User 68](#_Toc121165573)

[Administrator 68](#_Toc121165574)

[Moderator 68](#_Toc121165575)

[Others 68](#_Toc121165576)

[Score Reporter 68](#_Toc121165577)

[Volunteers 68](#_Toc121165578)

[Use Case Diagrams 69](#_Toc121165579)

[Participants 69](#_Toc121165580)

[Player 69](#_Toc121165581)

[Potential Player 69](#_Toc121165582)

[Spectator 69](#_Toc121165583)

[Commentator 69](#_Toc121165584)

[Videographer 69](#_Toc121165585)

[Organizers 69](#_Toc121165586)

[League Admin 69](#_Toc121165587)

[Tournament Admin 69](#_Toc121165588)

[Coach 69](#_Toc121165589)

[Treasurer 69](#_Toc121165590)

[Logistics 70](#_Toc121165591)

[Equipment Manager 70](#_Toc121165592)

[Donor 70](#_Toc121165593)

[Vendor 70](#_Toc121165594)

[Venue Representative 70](#_Toc121165595)

[Site Administrators 70](#_Toc121165596)

[User 70](#_Toc121165597)

[Administrator 70](#_Toc121165598)

[Moderator 70](#_Toc121165599)

[Others 70](#_Toc121165600)

[Score Reporter 70](#_Toc121165601)

[Volunteers 70](#_Toc121165602)

[Data Flow Diagrams 71](#_Toc121165603)

[Participants 71](#_Toc121165604)

[Player 71](#_Toc121165605)

[Potential Player 71](#_Toc121165606)

[Spectator 71](#_Toc121165607)

[Commentator 71](#_Toc121165608)

[Videographer 71](#_Toc121165609)

[Organizers 71](#_Toc121165610)

[League Admin 71](#_Toc121165611)

[Tournament Admin 71](#_Toc121165612)

[Coach 71](#_Toc121165613)

[Treasurer 71](#_Toc121165614)

[Logistics 72](#_Toc121165615)

[Equipment Manager 72](#_Toc121165616)

[Donor 72](#_Toc121165617)

[Vendor 72](#_Toc121165618)

[Venue Representative 72](#_Toc121165619)

[Site Administrators 72](#_Toc121165620)

[User 72](#_Toc121165621)

[Administrator 72](#_Toc121165622)

[Moderator 72](#_Toc121165623)

[Others 72](#_Toc121165624)

[Score Reporter 72](#_Toc121165625)

[Volunteers 72](#_Toc121165626)

[Activity Diagrams 73](#_Toc121165627)

[Participants 73](#_Toc121165628)

[Player 73](#_Toc121165629)

[Potential Player 73](#_Toc121165630)

[Spectator 73](#_Toc121165631)

[Commentator 73](#_Toc121165632)

[Videographer 73](#_Toc121165633)

[Organizers 73](#_Toc121165634)

[League Admin 73](#_Toc121165635)

[Tournament Admin 73](#_Toc121165636)

[Coach 73](#_Toc121165637)

[Treasurer 73](#_Toc121165638)

[Logistics 74](#_Toc121165639)

[Equipment Manager 74](#_Toc121165640)

[Donor 74](#_Toc121165641)

[Vendor 74](#_Toc121165642)

[Venue Representative 74](#_Toc121165643)

[Site Administrators 74](#_Toc121165644)

[User 74](#_Toc121165645)

[Administrator 74](#_Toc121165646)

[Moderator 74](#_Toc121165647)

[Others 74](#_Toc121165648)

[Score Reporter 74](#_Toc121165649)

[Volunteers 74](#_Toc121165650)

[State Chart Diagrams 75](#_Toc121165651)

[Participants 75](#_Toc121165652)

[Player 75](#_Toc121165653)

[Potential Player 75](#_Toc121165654)

[Spectator 75](#_Toc121165655)

[Commentator 75](#_Toc121165656)

[Videographer 75](#_Toc121165657)

[Organizers 75](#_Toc121165658)

[League Admin 75](#_Toc121165659)

[Tournament Admin 75](#_Toc121165660)

[Coach 75](#_Toc121165661)

[Treasurer 75](#_Toc121165662)

[Logistics 76](#_Toc121165663)

[Equipment Manager 76](#_Toc121165664)

[Donor 76](#_Toc121165665)

[Vendor 76](#_Toc121165666)

[Venue Representative 76](#_Toc121165667)

[Site Administrators 76](#_Toc121165668)

[User 76](#_Toc121165669)

[Administrator 76](#_Toc121165670)

[Moderator 76](#_Toc121165671)

[Others 76](#_Toc121165672)

[Score Reporter 76](#_Toc121165673)

[Volunteers 76](#_Toc121165674)

[Sequence Diagrams 77](#_Toc121165675)

[Participants 77](#_Toc121165676)

[Player 77](#_Toc121165677)

[Potential Player 77](#_Toc121165678)

[Spectator 77](#_Toc121165679)

[Commentator 77](#_Toc121165680)

[Videographer 77](#_Toc121165681)

[Organizers 77](#_Toc121165682)

[League Admin 77](#_Toc121165683)

[Tournament Admin 77](#_Toc121165684)

[Coach 77](#_Toc121165685)

[Treasurer 77](#_Toc121165686)

[Logistics 78](#_Toc121165687)

[Equipment Manager 78](#_Toc121165688)

[Donor 78](#_Toc121165689)

[Vendor 78](#_Toc121165690)

[Venue Representative 78](#_Toc121165691)

[Site Administrators 78](#_Toc121165692)

[User 78](#_Toc121165693)

[Administrator 78](#_Toc121165694)

[Moderator 78](#_Toc121165695)

[Others 78](#_Toc121165696)

[Score Reporter 78](#_Toc121165697)

[Volunteers 78](#_Toc121165698)

[Data Dictionary 79](#_Toc121165699)

[Data Domain 80](#_Toc121165700)

[Storage Models 80](#_Toc121165701)

[View Models 80](#_Toc121165702)

[Database Design 81](#_Toc121165703)

[Entity Relational Diagram 82](#_Toc121165704)

[Stored Procedure Definitions 83](#_Toc121165705)

[UML Models – Data Management Class Diagrams 84](#_Toc121165706)

[UML Models – Data Access Class Diagrams 85](#_Toc121165707)

[Implementation Plan 86](#_Toc121165708)

[Project Structure 86](#_Toc121165709)

[Timeline 86](#_Toc121165710)

[Technology and Tools 86](#_Toc121165711)

[Testing 86](#_Toc121165712)

[QA Process 86](#_Toc121165713)

# Executive Summary

## Goals

## Stakeholders

## Tools

## Timeline

# User Stories

## Participants

### Player

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| User Story Name | Feature | As a(n) <actor> | I would like to <description> | So that <outcome> |
| Player\_001 | Show unavailability | Player, PoPlayer | Show my unavailability | I am able to show others when I am not available by posting a busy event on my schedule. |
| Player\_002 | View Team role | Player | See my team role | When I am viewing my team, I can see what role I am selected to me by my coach. |
| Player\_003 | Viewing Schedule | Player, Referee | View my schedule | I can view the upcoming events I am signed up for. |
| Player\_004 | Editing Schedule | Player, PoPlayer | Be able to edit my schedule | I can change my availability to participate in events. |
| Player\_005 | Edit User Bio | Player, PoPlayer, Referee | Specify which sports I enjoy | Other users can see which sports events I am interested in participating in. |
| Player\_007 | Sort enrolled events | Player | sort the events I am enrolled in | I can see a sorted list of upcoming events, sorted by date, sport, or location. |
| Player\_009 | View designated parking | Player, Referee, Spectator | View designated parking | I can see if there is designated parking and if so where it is located. |
| Player\_010 | View Messages | Player | View Messages I received | See what messages teammates or coaches have sent me. |
| Player\_011 | Respond to Messages | Player, PoPlayer | Respond to Messages | Respond to messages I have received |
| Player\_014 | View Tournament Progress | Player | View Tournament Progress | See the progress of any tournament I am participating in. |
| Player\_015 | View Event Media | Player | View Event Media | I can view the game posts provided by the  commentators and videographers. |
| Player\_016 | Comment on Media | Player, User | Comment On Media | I can let others know how I feel about games that were played |

### Potential Player

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| User Story Name | Feature | As a(n) <actor> | I would like to <description> | So that <outcome> |
| PoPlayer\_005 | Respond to join messages | Potential Player | Respond to join messages | I can respond to the coach and let them know I would like to join the team. |
| PoPlayer\_008 | Message team coach | Potential Player | Message Coach | I can message a coach about any questions I have on their team. |

### Spectator

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| User Story Name | Feature | As a(n) <actor> | I would like to <description> | So that <outcome> |
| Spectator\_001 | View donation options | Spectator | View donation options | I can leave a donation if I wish to support an event |
| Spectator\_002 | See seating locations | Spectator | See seating locations | I know where I am supposed to sit at an event |

### Commentator

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| User Story Name | Feature | As a(n) <actor> | I would like to <description> | So that <outcome> |
| Commentator\_001 | Make Game Post | Commentator | Upload Audio Recordings | I can share my audio recordings with other users. |
| Commentator\_002 | See provided equipment | Commentator, Player, Referee | See provided equipment | I know what equipment is provided and what equipment I need to bring |
| Commentator\_003 | Remove Game Post | Commentator | Remove Recordings | I have the ability to remove any audio recordings I have previously uploaded. |
| Commentator\_004 | View players in event | Commentator, Player, PoPlayer, Referee | View players | I know the names of the players participating in the event I am commentating. |

### Videographer

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| User Story Name | Feature | As a(n) <actor> | I would like to <description> | So that <outcome> |
| Videographer\_002 | Post a video | Videographer | Post a video | I can share video links for Recording I have made during the event |
| Videographer\_003 | Remove a video | Videographer | Remove a video | I can remote video links I have posted for event if I shared the incorrect link |

## Organizers

### League Admin

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| User Story Name | Feature | As a(n) <actor> | I would like to <description> | So that <outcome> |
| League\_001 | set up league | League Admin, User | Set up a league/Assign a sport | so that I can post and get people to sign up. |
| League\_002 | modify a league | League Admin | Modify a league | so that I can edit a league already made |
| League\_003 | remove a league | League Admin | Remove a league | it no longer appears as a league |
| League\_004 | create a season | League Admin | Create a season/generate games | I can host a season of games |
| League\_005 | modify a season | League Admin | Modify a season | I can change the season details or dates |
| League\_006 | copy a season | League Admin | Copy a season | I can save time and make seasons annually. |
| League\_007 | remove a season | League Admin | Remove a season | it no longer appears as a season. |
| League\_008 | open league registration | League Admin | Open League registration | teams can register for league |
| League\_009 | close league registration | League Admin | Close League registration | teams cannot register for league |
| League\_010 | add team to league | League Admin | Add a team to a league | to add a team in case there are last minute changes |
| League\_011 | remove team from league | League Admin | Remove a team from a league | I can remove a team in case of last-minute changes |
| League\_012 | message participant | League Admin | Message a participant | I can update them on important details |
| League\_013 | message participants | League Admin | Message a list of participants | I can update everyone on changes or details |
| League\_014 | copy a league | League Admin | Copy a League | I can save time and make annual leagues. |
| League\_015 | generate games | League Admin | Generate a list of Games | I can determine which team is playing which team |

### Tournament Admin

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| User Story Name | Feature | As a(n) <actor> | I would like to <description> | So that <outcome> |
| Tournament\_001 | create a tournament | Tournament Admin, User | Create a tournament | announce to others that there is a tournament |
| Tournament\_002 | modify a tournament | Tournament Admin | Modify a tournament | change a tournament that I have created. |
| Tournament\_003 | copy a tournament | Tournament Admin | Copy a tournament | copy a tournament that was already created. |
| Tournament\_004 | Remove a tournament | Tournament Admin | Remove a tournament | so that it no longer appears as a tournament |
| Tournament\_005 | Open tournament Registration | Tournament Admin | Open tournament registration | so that teams can register for my event |
| Tournament\_006 | Close Tournament Registration | Tournament Admin | Close tournament registration | so that no more teams can register for my event |
| Tournament\_007 | Add team to tournament | Tournament Admin | Add a team to a tournament | teams can be manually added in case there are last minute changes. |
| Tournament\_008 | remove team from tournament | Tournament Admin | Remove team from tournament | teams can be manually removed in case there are last minute changes. |
| Tournament\_011 | generate matches | Tournament Admin | Generate a list of matches | I know which teams will compete against each other |
| Tournament\_012 | generate tournament bracket | Tournament Admin | Generate a tournament bracket | I know which teams are set to play against each other in the future. |

### Coach

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| User Story Name | Feature | As a(n) <actor> | I would like to <description> | So that <outcome> |
| Coach\_001 | view teams | Coach | View list of teams | I can make sure there is enough teams for an event. |
| Coach\_002 | create team | Coach | Create team profile | I can register for leagues or tournaments |
| Coach\_003 | change team | Coach | Change Team information | I can update players and information about the team |
| Coach\_004 | remove own team | Coach | Remove Own Team | I can withdraw from events. |
| Coach\_005 | add player to team | Coach | Add player to team | I can form the team |
| Coach\_006 | change player role on team | Coach | Change player role on team | I can update changes |
| Coach\_007 | remove player from team | Coach | Remove player from team | changes can be made if needed |
| Coach\_008 (MSG) | message whole team | Coach | Message all players on a team | I can notify all the players about events |
| Coach\_009 | create practices | Coach | Create recurring practices | I can easily schedule practices during a season |
| Coach\_010 | remove practices | Coach | Remove a practice | I can remove practices that will no longer be happening |

### Treasurer

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| User Story Name | Feature | As a(n) <actor> | I would like to <description> | So that <outcome> |
| Treasurer\_001 | request fund | Treasurer | Request funds | I can receive a request for funds |

## Logistics

### Equipment Manager

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| User Story Name | Feature | As a(n) <actor> | I would like to <description> | So that <outcome> |
| EquipmentManager\_001 | Equipment List | Equipment Manager | keep inventory of equipment and include quality | stay organized and show what equipment can be replaced or added |
| EquipmentManager\_001 | Equipment List | Equipment Manager | keep inventory of equipment and include quality | stay organized and show what equipment can be replaced or added |
| EquipmentManager\_001 | Equipment List | Equipment Manager | keep inventory of equipment and include quality | stay organized and show what equipment can be replaced or added |
| EquipmentManager\_002 | Equipment List | Equipment Manager | share this list with my team and other equipment managers | make sure my team has what they need to play |

### Donor

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| User Story Name | Feature | As a(n) <actor> | I would like to <description> | So that <outcome> |
| Donor\_004 | Request List | Donor | view the equipment list of a team | I can help out my chosen team |

### Vendor

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| User Story Name | Feature | As a(n) <actor> | I would like to <description> | So that <outcome> |
| Vendor\_001 | Venue, Venue Event | Vendor | be able to send a connection request to be approved by a venue | be associated with a venue |
| Vendor\_002 | Requests | Vendor | view, approve, and deny requests for tournaments | I can sign up for them accordingly |
| Vendor\_003 | Profile | Vendor | Display type of vendor, images of what I sell, information about what I sell, prices, address, email, social media link, link to personal website, phone number, location, if they are located in a venue, business hours, contact personal, if they do fundraisers, for profit or volunteer, rating) | I can show what kind of merchandise I sell on my profile |
| Vendor\_004 | Profile | Vendor | Have a way to indicate I am a vendor | so people know I am a vendor |
| Vendor\_005 | Profile | Vendor | a way to view upcoming events and see the time, date, location, and type of event being hosted | that way if the vendor wants a way to see potential customers they can get that basic info |
| Vendor\_006 | Profile | Vendor | Creating a list of previous jobs | coordinators can see what experience a vendor has |
| Vendor\_007 | Form | Vendor | a way to edit a limited list of products with a price for each one | so they can send it to people and receive it |

### Venue Representative

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| User Story Name | Feature | As a(n) <actor> | I would like to <description> | So that <outcome> |
| VenueRep\_002 | Equipment list | Venue Rep | have an equipment list | I can keep track of what I have available at my venue |
| VenueRep\_003 | Connection Request | Venue Rep | receive or send association requests | I can confirm or create associations with Vendors, or my home team |
| VenueRep\_004 | Vendor Profile | Venue Rep | display information about my venue (description, venue capacity, location, home team, donors, photos, costs, website, vendors, rating, rules, availability, equipment, venue type) | I can inform potential teams/tournaments who will use it |

## Site Administrators

### User

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| User Story Name | Feature | As a(n) <actor> | I would like to <description> | So that <outcome> |
| User\_001 | Create Account | User | create an account | I can use the system |
| User\_002 | Add Profile Picture | User | add a profile picture | I can have a picture that identifies me |
| User\_003 | Login | User | log in | I can access my account |
| User\_004 | Logout | User | log out | I can prevent unauthorized access to my account |
| User\_005 | Browse Events | User | browse events | I can find events |
| User\_006 | Create Events | User | create events | I can create events |
| User\_007 | Join Events | User | join events | I can participate in events |
| User\_008 | Leave Events | User | leave events | I can leave events that I am no longer interested in |
| User\_012 | Report Issue | User | report an issue | I can report issues with the site, my account, or other users |
| User\_015 | Suggest Feature | User | suggest a feature | I can provide a suggestion on what I want to be added to the site or removed |
| User\_016 | Reset Password | User | reset password | I can reset my password if they are lost or forgotten |
| User\_017 | Deactivate Account | User | deactivate account | I can temporarily or indefinitely hide my account. |
| User\_020 | Search for Events | User | search for events | I can see what is happening |
| User\_021 | Search for Teams | User | search for teams | I can locate teams |
| User\_022 | Search by Category | User | narrow teams/events down by categories (locality , type of sport,team name,tournaments, promoter, coach, player name) | I can help find what I am looking for |
| User\_023 | See Team/Tournament Details | User | see team and tournament details (players profile and locality) | I can learn more about a team |
| User\_024 | Edit Personal Bio | User | edit my personal bio | I can show others what activities or tasks I am apart of |

### Administrator

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| User Story Name | Feature | As a(n) <actor> | I would like to <description> | So that <outcome> |
| Admin\_003 | Ban User | Admin | ban users | I can manage and moderate site against harmful or malicious users for a set amount of time |
| Admin\_005 | Grant Privileges | Admin | grant moderator privileges | I can assign trusted individuals to monitor the site |
| Admin\_006 | Revoke Privileges | Admin | revoke moderator privileges | I can remove non trusted individuals from the site |
| Admin\_007 | Delete Events | Admin | delete events | I can remove events/activities from the site |
| Admin\_008 |  | Admin | delete users | I can remove users from the site |
| Admin\_009 | Unlock Account | Admin | unlock account | I can reinstate accounts that have been removed |
| Admin\_010 | Reset Password | Admin | reset password | I can reset users passwords if they are lost or forgotten |

### Moderator

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| User Story Name | Feature | As a(n) <actor> | I would like to <description> | So that <outcome> |
| Moderator\_001 | Update Roles | Moderators | Update Roles | I can update or change the user roles |
| Moderator\_002 | Add User | Moderators | Add users to team | I can add users to teams |
| Moderator\_003 | Remove User | Moderators | Remove users from team | I can remove users from teams |
| Moderator\_004 | Ban User | Moderators | ban users | I can manage and moderate site |

## Others

### Score Reporter

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| User Story Name | Feature | As a(n) <actor> | I would like to <description> | So that <outcome> |
| Score\_001 | Add Score to Event | Score Reporter | add score to the event | I can create match reports |
| Score\_002 | Update Score to Event | Score Reporter | update score to the event | I can update match reports |
| Score\_003 | Accept Reporter Role | Score Reporter | Accept role to be a reporter | I can add scores to match reports |

### Volunteers

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| User Story Name | Feature | As a(n) <actor> | I would like to <description> | So that <outcome> |
| Volunteer\_001 | See Available Events | Volunteers | see events that are available to volunteer | I can decide what interests me or see what I would like to participate in |

# Use Cases

## Participants

### Player

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID:** | Player\_001 | | | |
| **Use Case Name:** | Show Unavailability | | | |
| **Created By:** | Elijah Morgan | | **Last Updated By:** | Michael Haring |
| **Date Created:** | 9/17/2022 | | **Last Revision Date:** | 11/07/2022 |
| **Actors:** | | 1. Player 2. Potential Player | | |
| **Description:** | | A player/potential player can change their availability to show when they are able to attend games/events so that coaches and team captains know when a player is available. | | |
| **Trigger:** | | A player goes into their profile and clicks on the availability button | | |
| **Preconditions:** | | 1. User must be logged in to the website | | |
| **Postconditions:** | | 1. The player will have their availability set for whatever days they have selected | | |
| **Normal Flow:** | | 1. User logs in to the website 2. User enters their profile page 3. User selects their player schedule 4. User selects button that says “Change your availability” 5. User selects days on calendar for availability 6. User selects button, confirming their selected days 7. User is taken back to their player schedule | | |
| **Alternative Flows:**  **[Alternative Flow 1 – Player already scheduled]** | | 6a. In step 6 of the normal flow, if user selects day with an already scheduled event:   1. User is prompted with events scheduled for that day. 2. User clicks event. 3. Popup appears for user to remove event. 4. User is prompted to confirm change. 5. User confirms change. 6. User is taken back to their player schedule with updated event information. | | |
| **Exceptions:** | | None | | |
| **Includes:** | | TBD | | |
| **Frequency of Use:** | | The player will be able to use this function as many times as they would like. | | |
| **Special Requirements:** | | Someone who is not a player should not be able to change their own current status (I.E., spectators) | | |
| **Assumptions:** | | The player can read English | | |
| **Notes and Issues:** | |  | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID:** | Player\_002 | | | |
| **Use Case Name:** | Team Role | | | |
| **Created By:** | Brendan Klostermann | | **Last Updated By:** | Toney Hale |
| **Date Created:** | 09/18/2022 | | **Last Revision Date:** | 11/14/2022 |
| **Actors:** | | Player  Coach | | |
| **Description:** | | A Player needs to be able to see what position they are assigned on their team. | | |
| **Trigger:** | | Player selects view team option | | |
| **Preconditions:** | | 1. User must be signed in 2. User must have the player role 3. Player must be currently on a team 4. Team Coach must have set team roles | | |
| **Postconditions:** | | 1. Player receives a list of the team players roles 2. Every Player has a team role assigned to their name | | |
| **Normal Flow:** | | 1. User signs in the website 2. User navigates to their team page 3. Player selects Team Roles button 4. System display list of players with their roles on the team 5. System displays option to close Team Role View | | |
| **Alternative Flows:**  **[Alternative Flow 1 – Not in Network]** | | 1. Player is not on the selected team 2. Player selects Team Roles Button 3. System displays player is not on team | | |
| **Exceptions:** | | 1. Team that user selected is not found 2. Error message appears that the team cannot be found | | |
| **Includes:** | | (None) | | |
| **Frequency of Use:** | | On Demand | | |
| **Special Requirements:** | | User must be on a team | | |
| **Assumptions:** | | Assumes User is on a team | | |
| **Notes and Issues:** | | None | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID:** | Player\_003 | | | |
| **Use Case Name:** | Viewing Schedule | | | |
| **Created By:** | Jacob Lindauer | | **Last Updated By:** | Jacob Lindauer |
| **Date Created:** | 10/2/2022 | | **Last Revision Date:** | 11/05/2022 |
| **Actors:** | | Player  Referee | | |
| **Description:** | | User able to view their schedule | | |
| **Trigger:** | | User goes to their schedule tab and can see their schedule in a calendar and list  format | | |
| **Preconditions:** | | 1. User must be on a team. | | |
| **Postconditions:** | | 1. User is able to view their events and practices in a calendar format. 2. User is able to click on dates and set up availability for those days. | | |
| **Normal Flow:** | | 1. User signs in the website 2. User navigates to their event screen 3. User is able to view their events and practices in a calendar format. | | |
| **Alternative Flows:**  **[Alternative Flow 1 – Not in Network]** | | No Alternative flows for this Use Case | | |
| **Exceptions:** | | User is not on a team:   * Message is prompted telling the User events and practices will be listed once they join a team. | | |
| **Includes:** | | (None) | | |
| **Frequency of Use:** | | Very frequently as this would be the primary function the User can use to view events and practices. | | |
| **Special Requirements:** | | User must be on a team, team must have scheduled games and practices | | |
| **Assumptions:** | | Assumes User is on a team | | |
| **Notes and Issues:** | | None | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID:** | Player\_004 | | | |
| **Use Case Name:** | Playing Editing Schedule | | | |
| **Created By:** | Jacob Lindauer | | **Last Updated By:** | Jacob Lindauer |
| **Date Created:** | 10/2/2022 | | **Last Revision Date:** | 11/05/2022 |
| **Actors:** | | * Player * Potential Player | | |
| **Description:** | | User able to edit their own schedule | | |
| **Trigger:** | | 1. User navigates to their schedule and would like to add or edit an event | | |
| **Preconditions:** | | 1. Peron clicks on their Schedule page 2. User clicks on an existing event to edit or clicks the add event button. | | |
| **Postconditions:** | | 1. User can see the added / edited event on their events list and calendar. | | |
| **Normal Flow:** | | 1. User navigates to their schedule. 2. User clicks on a date on their calendar or clicks the add event button. 3. A screen is displayed to create an event for the selected date. 4. User fills out the form and clicks save to add the created event. 5. User is returned to their calendar and can see their updated calendar events. | | |
| **Alternative Flows:**  **[Alternative Flow 1 – Removing an Event]**    **[Alternative Flow 2 - Editing an existing event]** | | 3a. In step 3 of the normal flow, the player clicks on an existing event and would like to remove the event.   1. User clicks on entered event on their calendar 2. A screen identical to the creation screen shows fields that can be updated. 3. The User clicks the remove button and is prompted if they are sure they would like to remove the event. 4. Regardless of their selection they are sent back to the schedule screen and the updated schedule is displayed.   2a. User clicks on a date with an existing event.   1. User clicks on an existing event on their events list. 2. A screen identical to the creation screen shows what fields can be updated. 3. User updates fields and then clicks the Save button. 4. User is returned to their schedule page and their updated schedule is listed. | | |
| **Exceptions:** | | (None) | | |
| **Includes:** | | Includes Player\_003 (Viewing Schedule) | | |
| **Frequency of Use:** | | This would be used very frequently as the player would need to be able to add / edit events they create. | | |
| **Special Requirements:** | | No special requirements for this Use Case. | | |
| **Assumptions:** | | None | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID:** | Player\_005 | | | |
| **Use Case Name:** | Show Sport Interests | | | |
| **Created By:** | Joseph Philipp | | **Last Updated By:** |  |
| **Date Created:** | 09/17/2022 | | **Last Revision Date:** |  |
| **Actors:** | | * User * Potential Player | | |
| **Description:** | | User can view messages they have about a certain event. | | |
| **Trigger:** | | User has received a message. | | |
| **Preconditions:** | | 1. User account is active 2. User is logged in 3. User has messages | | |
| **Postconditions:** | | 1. User views messages  2. User messages are marked as read | | |
| **Normal Flow:** | | 1. System displays user profile 2. User navigates to messages button 3. User clicks messages button 4. System displays user messages | | |
| **Alternative Flows:**  **[Alternative Flow 1 – Removing an Event]** | | 5a. In step 5 of the normal flow, if the user doesn’t have any messages   1. System will display no messages message 2. Use Case returns to step 4 of normal flow | | |
| **Exceptions:** | | 2a. In step 2 of the normal flow, if the user doesn’t have an account  1. System will display credentials don’t match message  2. Use Case returns to step 1 | | |
| **Includes:** | |  | | |
| **Frequency of Use:** | | Once a day | | |
| **Special Requirements:** | | No special requirements for this Use Case. | | |
| **Assumptions:** | | None | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Use Case ID:** | Player\_009 | | |
| **Use Case Name:** | View Designated Parking | | |
| **Created By:** | Joseph Philipp | **Last Updated By:** | Toney Hale |
| **Date Created:** | 09/17/2022 | **Last Revision Date:** | 11/14/2022 |
| **Actors:** | * Player * Referee | | |
| **Description:** | User can see where they are supposed to park at an event | | |
| **Trigger:** | User is signed up for an event that started or is starting soon | | |
| **Preconditions:** | 1. Player account is active 2. Player is logged in 3. Player is enrolled in an event 4. Player’s event has started or is starting soon | | |
| **Postconditions:** | 1. Player receives a map on where to park for event | | |
| **Normal Flow:** | 1. System displays player profile 2. Player clicks View Enrolled Events button 3. System displays enrolled events 4. Player clicks View Details button on event 5. System displays that events details 6. Player clicks event map 7. System displays event map 8. Player finds designated player parking on map | | |
| **Alternative Flows:**  **[Alternative Flow 1 – No Enrolled Events] [Alternative Flow 2 – No Map for Event]** | 2a. In step 2 of the normal flow, if the User is not enrolled in any events   1. System will show message stating player isn’t enrolled in any events 2. Use Case returns to step 1 of normal flow   9a. In step 9 of the normal flow, if the event doesn’t have a map   1. System will display message stating there is no map 2. Use Case returns to step 8 of normal flow | | |
| **Exceptions:** |  | | |
| **Includes:** | View Event Details | | |
| **Frequency of Use:** | On Demand | | |
| **Special Requirements:** |  | | |
| **Assumptions:** |  | | |
| **Notes and Issues:** |  | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID:** | Player\_010 | | | |
| **Use Case Name:** | View Messages | | | |
| **Created By:** | Joseph Philipp | | **Last Updated By:** | Jacob Lindauer |
| **Date Created:** | 09/17/2022 | | **Last Revision Date:** | 11/14/2022 |
| **Actors:** | | Player  PoPlayer  Coach  System | | |
| **Description:** | | User is able to view messages they have received from other teammates or coaches. | | |
| **Trigger:** | | User is sent a message from a teammate or coach. | | |
| **Preconditions:** | | 1. Coach or teammate messages another player.  2. User receives a message and notification for a new message.  3. User would like to view the message sent by another player or their coach. | | |
| **Postconditions:** | | 1. User is able to view the message sent to them. | | |
| **Normal Flow:** | | 1. User receives a message from another user. 2. User receives a notification from the system that they have received a new message. 3. User clicks on their notifications screen. 4. Notification screen indicates a new message was sent to them. 5. User clicks on notification and is brought to the message. 6. User is able to view the message and respond to it if they would like to. | | |
| **Alternative Flows:**  **[Alternative Flow 1 - User navigates to their message screen]**  **[Alternative Flow 2 - User has not received any messages]** | | 3a. In step 3 of the normal flow, the user navigates to their message screen rather than going through notification page.   1. User navigates to their message screen. 2. Their message list indicates which message is the new message. 3. User clicks on the message and normal flow continues at step 5.   5a. In step 5 of the normal flow. User navigates to their message screen but has not received any messages.   1. User is prompted with a screen saying “New messages will be listed here”. | | |
| **Exceptions:** | | * N/A | | |
| **Includes:** | | * Player\_018 for player viewing notifications | | |
| **Frequency of Use:** | | * On Demand | | |
| **Special Requirements:** | | * N/A | | |
| **Assumptions:** | | * N/A | | |
| **Notes and Issues:** | |  | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID:** | Player\_011 | | | |
| **Use Case Name:** | Responding to messages | | | |
| **Created By:** | Jacob Lindauer | | **Last Updated By:** | Jacob Lindauer |
| **Date Created:** | 9/17/2022 | | **Last Revision Date:** | 11/06/2022 |
| **Actors:** | | Player  PoPlayer | | |
| **Description:** | | User responding to messages sent to them | | |
| **Trigger:** | | User clicks on the messaging icon and is provided a list of people who have messaged them. User then clicks on the message they would like to respond to. | | |
| **Preconditions:** | | 1. User receives a message from another user. | | |
| **Postconditions:** | | 1. Message is sent to the other user and is displayed in the chat window. | | |
| **Normal Flow:** | | 1. User clicks on the messaging icon. 2. Messaging menu displays a list of all users who have messaged them. 3. User clicks on a message and a chat window appears. 4. User closes the chat tab with the User when finished. | | |
| **Alternative Flows:**  **[Alternative Flow 1 – None]** | | N/A | | |
| **Exceptions:** | | N/A | | |
| **Includes:** | | Includes Player\_010 | | |
| **Frequency of Use:** | | High Frequency. On Demand. | | |
| **Special Requirements:** | | N/A | | |
| **Assumptions:** | | N/A | | |
| **Notes and Issues:** | |  | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID:** | Player\_014 | | | |
| **Use Case Name:** | View Game Post | | | |
| **Created By:** | Elijah Morgan | | **Last Updated By:** | Elijah Morgan |
| **Date Created:** | 09/17/2022 | | **Last Revision Date:** | 11/07/2022 |
| **Actors:** | | * Player | | |
| **Description:** | | Player can view details of an event | | |
| **Trigger:** | | Player opens selects an event on the calendar | | |
| **Preconditions:** | | 1. Player account is active 2. Player is logged in | | |
| **Postconditions:** | | 1. Player is shown the details of the event (or events) on the selected date | | |
| **Normal Flow:** | | 1. Player selects the events calendar 2. Player selects an event 3. Event details are shown | | |
| **Alternative Flows:**  **[Alternative Flow 1 – Event already Happened]** | | 2a. In step 2 of the normal flow if player selects an event that has already happened   1. System displays message stating “This event has already taken place, you may still view the details” 2. Continue to step 3 of the normal flow | | |
| **Exceptions:** | |  | | |
| **Includes:** | |  | | |
| **Frequency of Use:** | | Whenever the player wants | | |
| **Special Requirements:** | |  | | |
| **Assumptions:** | |  | | |
| **Notes and Issues:** | |  | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID:** | Player\_015 | | | |
| **Use Case Name:** | View Game Post | | | |
| **Created By:** | Joseph Philipp | | **Last Updated By:** | Joseph Philipp |
| **Date Created:** | 09/17/2022 | | **Last Revision Date:** | 11/07/2022 |
| **Actors:** | | * Player * Spectator | | |
| **Description:** | | Player can view messages they have about a certain event. | | |
| **Trigger:** | | Player has received a message. | | |
| **Preconditions:** | | 1. Player account is active 2. Player is logged in 3. Player has participated in an event 4. Player’s event has ended 5. Video is posted to event details | | |
| **Postconditions:** | | 1. Player watches event video | | |
| **Normal Flow:** | | 1. System displays player profile 2. Player navigates to past events 3. System displays past events 4. Player views past event details 5. System displays past event details 6. Player clicks video link 7. System navigates to video 8. System plays video | | |
| **Alternative Flows:**  **[Alternative Flow 1 – No Events Attended] [Alternative Flow2 – No Video for Event]** | | 2a. In step 2 of the normal flow if player hasn’t attended any events   1. System displays message stating player hasn’t attended any events. 2. Use Case returns to step 1   5a. In step 5 of the normal flow, if the past event doesn’t have a video   1. System displays message stating there is no video   Use Case returns to step 4 | | |
| **Exceptions:** | |  | | |
| **Includes:** | | View Event Details | | |
| **Frequency of Use:** | | After an event (weekly) | | |
| **Special Requirements:** | |  | | |
| **Assumptions:** | |  | | |
| **Notes and Issues:** | |  | | |

### Potential Player

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID:** | PoPlayer\_005 | | | |
| **Use Case Name:** | Accept Join Invites | | | |
| **Created By:** | Jacob Lindauer | | **Last Updated By:** | Jacob Lindauer |
| **Date Created:** | 9/17/2022 | | **Last Revision Date:** | 11/14/2022 |
| **Actors:** | | Potential Player  Players | | |
| **Description:** | | User would like to accept an invite from a captain to join their team. | | |
| **Trigger:** | | 1. User has received a message from a coach asking if they would like to join their team  2. User would like to respond and acknowledge they would like to join. | | |
| **Preconditions:** | | 1. User has received a message from a coach asking if they would like to join their team. | | |
| **Postconditions:** | | 1. Coach adds the user to their team roster.  2. User is listed as a player and their schedule is updated accordingly. | | |
| **Normal Flow:** | | 1. Coach reached out to a user asking if they would like to join their team.  2. User responds saying they would like to join their team.  3. Coach goes to team profile and adds the player to their roster.  4. Users schedule is updated to display team events and practices. | | |
| **Alternative Flows:**  **[Alternative Flow 1 – User would not like to join the team]** | | 2a. In step 2 of the normal flow. User responds to Coach indicating they do not want to join the team.   1. User responds to Coach saying they are not interested in joining. 2. Coach does not add player to player roster. | | |
| **Exceptions:** | | N/A | | |
| **Includes:** | | Includes Player\_018 where a User can view notifications. | | |
| **Frequency of Use:** | | This is a high priority feature as the captains need to be able to invite players and those players should be able to accept invites. | | |
| **Special Requirements:** | | N/A | | |
| **Assumptions:** | | Assumes User is wanting to join the team. | | |
| **Notes and Issues:** | |  | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID:** | PoPlayer\_008 | | | |
| **Use Case Name:** | Message Team Captain | | | |
| **Created By:** | Brendan Klostermann | | **Last Updated By:** | Toney Hale |
| **Date Created:** | 09/18/2022 | | **Last Revision Date:** | 11/14/2022 |
| **Actors:** | | Potential Player (Primary) | | |
| **Description:** | | As a potential player I want the ability to message team captains so that I can respond to messages from captains looking for players. | | |
| **Trigger:** | | Potential Player selects messages from other users | | |
| **Preconditions:** | | 1. User must be signed in | | |
| **Postconditions:** | | 1. System will send notification to team captain/coach | | |
| **Normal Flow:** | | 1. Potential Player selects team they wish to apply for 2. Player click apply for team 3. Notification is sent to team about the request | | |
| **Alternative Flows:**  **[Alternative Flow 1 – Not in Network]** | |  | | |
| **Exceptions:** | | 1. Team is full | | |
| **Includes:** | |  | | |
| **Frequency of Use:** | | On Demand | | |
| **Special Requirements:** | |  | | |
| **Assumptions:** | | Assumes user is from that area | | |
| **Notes and Issues:** | | None | | |

### Spectator

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID:** | Spectator\_001 | | | |
| **Use Case Name:** | View donation options | | | |
| **Created By:** | Elijah Morgan | | **Last Updated By:** |  |
| **Date Created:** | 9/18/2022 | | **Last Revision Date:** |  |
| **Actors:** | | 1. Spectator | | |
| **Description:** | | A spectator can donate to a team or to a tournament to help fund future events | | |
| **Trigger:** | | A player goes onto the site and views an option where they can donate to teams through the application | | |
| **Preconditions:** | | 1. The Spectator must be logged into the website | | |
| **Postconditions:** | | 1. The spectator gets a message thanking them for donating to a team or tournament | | |
| **Normal Flow:** | | 1. Spectator logs into the website 2. Spectator clicks a button that leads them to the donation process 3. The spectator’s payment info is filled in 4. Spectator is shown a prompt, thanking them for donating | | |
| **Alternative Flows:**  **[Alternative Flow 1 – Payment method is accepted]**  **[Alternative Flow 2 – Payment method is not accepted]** | | 3a. In step 3 of the normal flow, if the Spectator has their payment method is authorized   1. The donation will be completed 2. Use Case resumes on step 4 of normal flow   3b. In step 3 of the normal flow, if the Spectator has their payment method is not authorized   1. A message will appear, telling the user that their payment has not been accepted 2. They will be reverted to step 2 of the normal flow | | |
| **Exceptions:** | | 3a. In step 3 in normal flow an internal error prevents a spectator from processing the donation   1. Spectator is shown a prompt with a message saying “An internal error has occurred, please try again later. If issues continue, please contact support.” | | |
| **Includes:** | | **TBD** | | |
| **Frequency of Use:** | | The Spectator will be able to use this function as many times as they would like. | | |
| **Special Requirements:** | | A spectator should be able to see their past donations they have made | | |
| **Assumptions:** | | The spectator can read English | | |
| **Notes and Issues:** | | 1. Are the donations meant for the teams, tournament owners or both | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID:** | Spectator\_002 | | | |
| **Use Case Name:** | View Seating Locations | | | |
| **Created By:** | Jacob Lindauer | | **Last Updated By:** | Jacob Lindauer |
| **Date Created:** | 9/17/2022 | | **Last Revision Date:** | 11/06/2022 |
| **Actors:** | | * Spectators | | |
| **Description:** | | User viewing seating locations | | |
| **Trigger:** | | User is going to attend an event and would like to see if there are designated seating areas. | | |
| **Preconditions:** | | 1. User navigates to the Event Details page. 2. User clicks on the button to view seating. | | |
| **Postconditions:** | | 1. User is displayed seating locations 2. User can close the window when finished. | | |
| **Normal Flow:** | | 1. User is attending an event and would like to view seating locations. 2. User navigates to the event details page. 3. User clicks on the button to view seating. 4. A screen is displayed should the seating location information. | | |
| **Alternative Flows:**  **[Alternative Flow 1 – Event does not have seating locations]**  **[Alternative Flow 2 – Event does not have any in-person attendance]** | | 4a. In step 4 the event does not have a designated seating area for the event.   1. The screen where the seating area is displayed will say “There is no seating information provided.”   4a. In step 4 the event is virtual and does not have in-person attendance.   1. The screen for the seating area information will display a message saying, “Attendance for this event will be virtual.” | | |
| **Exceptions:** | | (None) | | |
| **Includes:** | | Includes User\_020 for viewing event details. | | |
| **Frequency of Use:** | | * This would potentially be used regularly if this information is provided for the event. | | |
| **Special Requirements:** | | * User must be viewing an event with designated seating areas. | | |
| **Assumptions:** | | * This assumes the user is viewing an event that has designated seating areas. * This assumes the user is viewing an event that has in person seating. | | |
| **Notes and Issues:** | | * Need to keep in mind that not all events have designated seating areas. * Some events can be attended remotely via live stream. | | |

### Commentator

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID:** | Commentator\_001 | | | |
| **Use Case Name:** | Upload audio recordings | | | |
| **Created By:** | Elijah Morgan | | **Last Updated By:** | Elijah Morgan |
| **Date Created:** | 9/18/2022 | | **Last Revision Date:** | 11/6/2022 |
| **Actors:** | | 1. Commentator | | |
| **Description:** | | A Commentator is able to make a game post of audio recordings that are linked to YouTube | | |
| **Trigger:** | | A game has ended and the commentator recorded audio and uploaded it to YouTube. | | |
| **Preconditions:** | | 1. The commentator must be logged into the website 2. The commentator must have an audio recording of the event 3. The audio is uploaded to YouTube | | |
| **Postconditions:** | | 1. The game post with the link to the audio commentary is posted for other users to view | | |
| **Normal Flow:** | | 1. Commentator is logged into the website 2. Commentator selects a button to create a game post 3. System displays game post text box 4. Commentator navigates to YouTube and finds their audio for certain game 5. Commentator copies the YouTube link to the audio to post 6. Commentator pastes the YouTube link into the game post text box 7. Commentator clicks post button 8. The game post is posted for people to view | | |
| **Alternative Flows:**  **[Alternative Flow 1 – The file type is wrong]** | | 4a. In step 4 of the normal flow, if there is no audio for certain game   1. The commentator goes to YouTube 2. Commentator uploads the audio for the game if they have it 3. Resume on step 4 of the normal flow | | |
| **Exceptions:** | | 7a. In step 7 of the normal flow, if the link exceeds the allotted characters for a game post.   1. Game post is disapproved 2. System displays message saying to many characters 3. Use case returns to step 3 in normal flow | | |
| **Includes:** | | **TBD** | | |
| **Frequency of Use:** | | after games | | |
| **Special Requirements:** | | A person who is not a verified commentator should not be able to post videos | | |
| **Assumptions:** | | The commentator can read English  The commentator has knowledge about the game they are commentating | | |
| **Notes and Issues:** | | 1. Will the recordings be shown on the persons profile or will they be public | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID:** | Commentator 002 | | | |
| **Use Case Name:** | See provided Equipment | | | |
| **Created By:** | Elijah Morgan | | **Last Updated By:** | Elijah Morgan |
| **Date Created:** | 9/18/2022 | | **Last Revision Date:** | 11/6/2022 |
| **Actors:** | | Commentator | | |
| **Description:** | | A commentator is able to see what equipment is going to be provided for the commentator | | |
| **Trigger:** | | A commentator goes onto the site to view the tournament, the commentator is able to see an option to view the event’s provided equipment | | |
| **Preconditions:** | | 1. The commentator must be logged into the site 2. The commentator must have the role as commentator | | |
| **Postconditions:** | | 1. The commentator is able to view an event and see an option to view an events equipment | | |
| **Normal Flow:** | | 1. Commentator logs into the website 2. Commentator views all events 3. Commentator selects an event from the list 4. Commentator selects an option to view the provided equipment 5. Commentator is shown a list of given equipment | | |
| **Alternative Flows:**  **[Alternative Flow 1 – The event is over]** | | 4a. In step 4 of the normal flow, if the commentator tries to view an event that has already ended   1. The commentator is not given the option to see provided equipment 2. The commentator is returned to step 3 from the normal list flow | | |
| **Exceptions:** | | 3a. In step 3 in normal flow an internal error prevents a commentator from viewing an event’s details  Commentator is shown a prompt with a message saying “An internal error has occurred, please try again later. If issues continue, please contact support.”  5a. In step 5 in normal flow an internal error prevents a commentator from equipment details   1. Commentator is shown a prompt with a message saying “An internal error has occurred, please try again later. If issues continue, please contact support.” | | |
| **Includes:** | | **TBD** | | |
| **Frequency of Use:** | | The Spectator will be able to use this function as many times as they would like.as long as the event is still available | | |
| **Special Requirements:** | | None | | |
| **Assumptions:** | | The spectator can read English | | |
| **Notes and Issues:** | | 1. Will the role who sets the provided equipment be the tournament organizer or the venue owner 2. Will the provided equipment be divided into who uses it (E.I, bats for players) | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID:** | Commentator\_003 | | | |
| **Use Case Name:** | Remove Game Post | | | |
| **Created By:** | Elijah Morgan | | **Last Updated By:** | Elijah Morgan |
| **Date Created:** | 9/18/2022 | | **Last Revision Date:** | 11/6/2022 |
| **Actors:** | | Commentator | | |
| **Description:** | | A Commentator is able to delete game posts that that link the YouTube audio recordings of a game | | |
| **Trigger:** | | A commentator has posted a game post linking the YouTube audio recording of the game | | |
| **Preconditions:** | | 1. The commentator must be logged into the website 2. There must be a game post of the audio for certain game 3. The commentator must be the same person who uploaded the audio file | | |
| **Postconditions:** | | 1. The game post with the link to the audio commentary is deleted | | |
| **Normal Flow:** | | 1. Commentator is logged into the website 2. Commentator navigates to the game posts 3. Commentator selects which game post with audio link they would like to delete 4. The commentator clicks delete post button 5. System displays message asking if they are sure about deleting the game post 6. Commentator clicks yes/confirm 7. System displays message saying post is deleted | | |
| **Alternative Flows:**  **[Alternative Flow 1 – The confirmation info is wrong]** | | 4a. In step 4 of the normal flow, the commentator decides they don't want to delete the game post.   1. Return to step 2 of the normal flow. | | |
| **Exceptions:** | | 3a. in step 3 of the normal flow, the commentator has no audio linked game posts   1. System displays message saying there are no audio linked game posts 2. Use case returns to step 2 of the normal flow | | |
| **Includes:** | | **TBD** | | |
| **Frequency of Use:** | | Anytime | | |
| **Special Requirements:** | | The videos can only be removed by the person who posted them or a site admin | | |
| **Assumptions:** | | The commentator can read English | | |
| **Notes and Issues:** | |  | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID:** | Commentator\_004 | | | |
| **Use Case Name:** | Remove Game Post | | | |
| **Created By:** | Elijah Morgan | | **Last Updated By:** | Elijah Morgan |
| **Date Created:** | 9/18/2022 | | **Last Revision Date:** | 11/6/2022 |
| **Actors:** | | Commentator  Player  Potential Player  Referee | | |
| **Description:** | | A commentator is able to see what players are performing in an event | | |
| **Trigger:** | | A commentator has logged into the website and clicks a button to view all events | | |
| **Preconditions:** | | 1. The commentator must be logged into the website | | |
| **Postconditions:** | | 1. The commentator | | |
| **Normal Flow:** | | 1. Commentator is logged into the website 2. Commentator navigates to the list of games 3. Commentator selects an event from the list 4. Commentator is given a detail of the selected event 5. A list of participating players is shown | | |
| **Alternative Flows:**  **[Alternative Flow 1 – The event has already happened]** | | 4a. In step 3 of the normal flow, the commentator selects an event that has already happened   1. The commentator is shown a message saying “This event has already passed, but you can still view details of the event.”” 2. Continue to step 4 of the normal flow | | |
| **Exceptions:** | |  | | |
| **Includes:** | | **TBD** | | |
| **Frequency of Use:** | | Anytime | | |
| **Special Requirements:** | | None | | |
| **Assumptions:** | | The commentator can read English | | |
| **Notes and Issues:** | |  | | |

### Videographer

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID:** | Videographer\_002 | | | |
| **Use Case Name:** | Upload audio recordings | | | |
| **Created By:** | Elijah Morgan | | **Last Updated By:** | Elijah Morgan |
| **Date Created:** | 9/18/2022 | | **Last Revision Date:** | 11/6/2022 |
| **Actors:** | | 1. Videographer | | |
| **Description:** | | A Videographer is able to make a game post of audio recordings that are linked to YouTube | | |
| **Trigger:** | | A game has ended and the Videographer recorded audio and uploaded it to YouTube. | | |
| **Preconditions:** | | 1. The Videographer must be logged into the website 2. The Videographer must have an audio recording of the event 3. The audio is uploaded to YouTube | | |
| **Postconditions:** | | 1. The game post with the link to the audio commentary is posted for other users to view | | |
| **Normal Flow:** | | 1. Videographer is logged into the website 2. Videographer selects a button to create a game post 3. System displays game post text box 4. Videographer navigates to YouTube and finds their audio for certain game 5. Videographer copies the YouTube link to the audio to post 6. Videographer pastes the YouTube link into the game post text box 7. Videographer clicks post button 8. The game post is posted for people to view | | |
| **Alternative Flows:**  **[Alternative Flow 1 – The file type is wrong]** | | 4a. In step 4 of the normal flow, if there is no audio for certain game   1. The Videographer goes to YouTube 2. Videographer uploads the audio for the game if they have it 3. Resume on step 4 of the normal flow | | |
| **Exceptions:** | | 7a. In step 7 of the normal flow, if the link exceeds the allotted characters for a game post.   1. Game post is disapproved 2. System displays message saying to many characters 3. Use case returns to step 3 in normal flow | | |
| **Includes:** | | **TBD** | | |
| **Frequency of Use:** | | after games | | |
| **Special Requirements:** | | A person who is not a verified Videographer should not be able to post videos | | |
| **Assumptions:** | | The Videographer can read English  The Videographer has knowledge about the game they are commentating | | |
| **Notes and Issues:** | | 1. Will the recordings be shown on the persons profile or will they be public | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID:** | Videographer\_003 | | | |
| **Use Case Name:** | Videographer being able to remove a video | | | |
| **Created By:** | Jacob Lindauer | | **Last Updated By:** | Elijah Morgan |
| **Date Created:** | 9/17/2022 | | **Last Revision Date:** | 10/10/2022 |
| **Actors:** | | * Videographer | | |
| **Description:** | | Videographer can remove a video attachment from an event | | |
| **Trigger:** | | Videographer clicks the remove video button next to a video they attached to an event. | | |
| **Preconditions:** | | 1. Videographer has attached a video to an event 2. Videographer returns to the event details page and wants to remove the event they had added. 3. There is a remove button next to the video link. 4. Videographer clicks remove button to remove the link. | | |
| **Postconditions:** | | 1. Videographer clicks remove on the video link 2. Page refreshes and the link is removed from the event details. | | |
| **Normal Flow:** | | 1. Videographer is signed in with an active account 2. Videographer goes to the an event details page. 3. A video has been attached to an event and the videographer would like to remove this attached link. 4. Videographer is given a remove button next to the link. 5. When the remove button is clicked the user is prompted with a screen asking if they are sure. 6. When the user selects yes at the prompt the page refreshes and the link has bene removed. | | |
| **Alternative Flows:**  **[Alternative Flow 1 – User clicks no on the prompt]** | | 5a. In step 5 the videographer clicks NO on this prompt.   1. Videographer clicks no on the removal prompt. 2. The page refreshes and the video link is not removed. | | |
| **Exceptions:** | | 5a. In step 5 the videographer clicks on the removal link but an error prevents it from being removed from the details page.   1. Videographer clicks YES on the removal prompt. 2. An error is displayed saying an error has occurred. 3. Page refreshes and video is not removed from the event details page. | | |
| **Includes:** | | * This should only include the videographer role | | |
| **Frequency of Use:** | | * This would be rarely used but it important if the videographer attached the incorrect video to an event | | |
| **Special Requirements:** | | * A video must be attached to an event. | | |
| **Assumptions:** | | * This assumes the event has a posted video that can be removed. | | |
| **Notes and Issues:** | | * Need to consider that the videographer can only remove videos they have posted to prevent them from removing link to other events. * May want to include a way coaches or site admins can remove videos that may not be appropriate or irrelevant to the site. | | |

## Organizers

### League Admin

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID:** | UC\_Leag\_001 | | | |
| **Use Case Name:** | Create a League | | | |
| **Created By:** | Toney Hale | | **Last Updated By:** | Toney Hale |
| **Date Created:** | 9/12/2022 | | **Last Revision Date:** | 9/20/2022 |
| **Actors:** | | League Admin (primary)  System (secondary) | | |
| **Description:** | | Grant league admin ability to add a league to system, allowing users to search and register for it. | | |
| **Trigger:** | | League Admin clicks Add League link | | |
| **Preconditions:** | | 1. User is authenticated as League Admin  2. User is at “List Leagues” view | | |
| **Postconditions:** | | 1. League information is added to system 2. User is notified of information addition success\failure 3. User is returned to “List Leagues” view | | |
| **Normal Flow:** | | 1. User clicks “Add League” link 2. System displays “Add League” view 3. User enters league information 4. User clicks ok 5. System validates form data 6. System adds league information to database 7. System displays success message 8. System displays “List Leagues” view | | |
| **Alternative Flows:**  **[Alternative Flow 1 – User Clicks Cancel]** | | 4a. In step 4 of the normal flow, if the user clicks cancel  1. System will prompt user to confirm cancel  2. User accepts  3. User is returned to “List Leagues” view  4b. In step 4 of the normal flow, if the user clicks cancel   1. System will prompt user to confirm cancel 2. User declines 3. Use Case resumes on step 4 of normal flow | | |
| **Exceptions:** | | 5a. In step 5 of the normal flow, if the form data is not valid   1. System notifies user form data invalid 2. Use Case resumes on step 3 of normal flow   6a. In step 6 of the normal flow, if there is an error persisting the data   1. System notifies user there was an error 2. Use Case resumes on step 3 of normal flow | | |
| **Includes:** | |  | | |
| **Frequency of Use:** | | 1 per season | | |
| **Special Requirements:** | |  | | |
| **Assumptions:** | |  | | |
| **Notes and Issues:** | |  | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID:** | UC\_Leag\_002 | | | |
| **Use Case Name:** | Modify a League | | | |
| **Created By:** | Toney Hale | | **Last Updated By:** | Toney Hale |
| **Date Created:** | 9/12/2022 | | **Last Revision Date:** | 9/20/2022 |
| **Actors:** | | League Admin (primary)  System (secondary) | | |
| **Description:** | | Grant league admin ability to modify a league in the system. | | |
| **Trigger:** | | League Admin clicks modify League link | | |
| **Preconditions:** | | 1. User is authenticated as League Admin 2. User is at “List Leagues” view | | |
| **Postconditions:** | | 1. League information is updated in the system 2. User is notified of information addition success\failure 3. User is returned to “List Leagues” view | | |
| **Normal Flow:** | | 1. User clicks “Modify League” link 2. System displays “Modify League” view 3. User enters league information 4. User clicks ok 5. System validates form data 6. System adds league information to database 7. System displays success message 8. System displays “List Leagues” view | | |
| **Alternative Flows:**  **[Alternative Flow 1 – User Clicks Cancel]** | | 4a. In step 4 of the normal flow, if the user clicks cancel  1. System will prompt user to confirm cancel  2. User accepts  3. User is returned to “List Leagues” view  4b. In step 4 of the normal flow, if the user clicks cancel   1. System will prompt user to confirm cancel 2. User declines 3. Use Case resumes on step 4 of normal flow | | |
| **Exceptions:** | | 5a. In step 5 of the normal flow, if the form data is not valid   1. System notifies user form data invalid 2. Use Case resumes on step 3 of normal flow   6a. In step 6 of the normal flow, if there is an error persisting the data   1. System notifies user there was an error 2. Use Case resumes on step 3 of normal flow | | |
| **Includes:** | |  | | |
| **Frequency of Use:** | |  | | |
| **Special Requirements:** | |  | | |
| **Assumptions:** | |  | | |
| **Notes and Issues:** | |  | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID:** | UC\_Leag\_003 | | | |
| **Use Case Name:** | Remove a League | | | |
| **Created By:** | Toney Hale | | **Last Updated By:** | Toney Hale |
| **Date Created:** | 9/12/2022 | | **Last Revision Date:** | 9/20/2022 |
| **Actors:** | | League Admin (primary)  System (secondary) | | |
| **Description:** | | Grant league admin ability to remove a league and its attributes. | | |
| **Trigger:** | | League Admin clicks Remove League Link | | |
| **Preconditions:** | | 1. User is authenticated as League Admin 2. User is at “List Leagues” view 3. User clicks league wanted for removal | | |
| **Postconditions:** | | 1. League information is removed to system 2. User is notified of information addition success\failure 3. User is returned to “List Leagues” view | | |
| **Normal Flow:** | | 1. User clicks “Remove League” link 2. System displays “Remove League” view 3. User confirms they want to remove the league 4. User clicks ok 5. System validates form data 6. System removes league information to database 7. System displays success message 8. System displays “List Leagues” view | | |
| **Alternative Flows:**  **[Alternative Flow 1 – User Clicks Cancel]** | | 4a. In step 4 of the normal flow, if the user clicks cancel  1. System will prompt user to confirm cancel  2. User accepts  3. User is returned to “List Leagues” view  4b. In step 4 of the normal flow, if the user clicks cancel   1. System will prompt user to confirm cancel 2. User declines 3. Use Case resumes on step 4 of normal flow | | |
| **Exceptions:** | | 5a. In step 5 of the normal flow, if the form data is not valid   1. System notifies user form data invalid 2. Use Case resumes on step 3 of normal flow   6a. In step 6 of the normal flow, if there is an error persisting the data   1. System notifies user there was an error 2. Use Case resumes on step 3 of normal flow | | |
| **Includes:** | |  | | |
| **Frequency of Use:** | |  | | |
| **Special Requirements:** | |  | | |
| **Assumptions:** | |  | | |
| **Notes and Issues:** | |  | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID:** | UC\_Leag\_004 | | | |
| **Use Case Name:** | Generate List of Games | | | |
| **Created By:** | Toney Hale | | **Last Updated By:** | Toney Hale |
| **Date Created:** | 9/12/2022 | | **Last Revision Date:** | 9/20/2022 |
| **Actors:** | | League Admin (primary)  System (secondary) | | |
| **Description:** | | Grant league admin ability to generate a list of games, allowing users to see game times | | |
| **Trigger:** | | League Admin clicks “Generate Schedule” Link | | |
| **Preconditions:** | | 1. User is authenticated as League Admin 2. User is at “List Leagues” view 3. User clicks what league they want to generate games for | | |
| **Postconditions:** | | 1. Team information is removed from the league 2. User is notified of information addition success\failure 3. User is returned to Current League view | | |
| **Normal Flow:** | | 1    User clicks “Generate Schedule” link   1. System displays “Generate Schedule” view 2. User clicks what teams they want to include and start times 3. User clicks ok 4. System validates form data 5. System generates the game schedule 6. System displays success message 7. System displays Current League view | | |
| **Alternative Flows:**  **[Alternative Flow 1 – User Clicks Cancel]** | | 4a. In step 4 of the normal flow, if the user clicks cancel  1. System will prompt user to confirm cancel  2. User accepts  3. User is returned to Current League view  4b. In step 4 of the normal flow, if the user clicks cancel   1. System will prompt user to confirm cancel 2. User declines 3. Use Case resumes on step 4 of normal flow | | |
| **Exceptions:** | | 5a. In step 5 of the normal flow, if the form data is not valid   1. System notifies user form data invalid 2. Use Case resumes on step 3 of normal flow   6a. In step 6 of the normal flow, if there is an error persisting the data   1. System notifies user there was an error 2. Use Case resumes on step 3 of normal flow | | |
| **Includes:** | |  | | |
| **Frequency of Use:** | | Start of Season | | |
| **Special Requirements:** | |  | | |
| **Assumptions:** | |  | | |
| **Notes and Issues:** | |  | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID:** | UC\_Leag\_005 | | | |
| **Use Case Name:** | Modify a Season | | | |
| **Created By:** | Toney Hale | | **Last Updated By:** | Toney Hale |
| **Date Created:** | 9/28/2022 | | **Last Revision Date:** | 9/28/2022 |
| **Actors:** | | League Admin (primary)  System (secondary) | | |
| **Description:** | | Grant league admin ability to modify a season in the system. | | |
| **Trigger:** | | League Admin clicks “Edit League” link | | |
| **Preconditions:** | | 1. User is authenticated as League Admin 2. User is at “List Leagues” view 3. User click the league they are wanting to use | | |
| **Postconditions:** | | 1. League information is updated in system 2. User is notified of information update success\failure 3. User is returned to “List Leagues” view | | |
| **Normal Flow:** | | 1. User clicks “Edit Season” link 2. System displays “Edit Season” view 3. User enters league information 4. User clicks ok 5. System validates form data 6. System updates season information in database 7. System displays success message 8. System displays “List Leagues” view | | |
| **Alternative Flows:**  **[Alternative Flow 1 – User Clicks Cancel]** | | 4a. In step 4 of the normal flow, if the user clicks cancel  1. System will prompt user to confirm cancel  2. User accepts  3. User is returned to “List Leagues” view  4b. In step 4 of the normal flow, if the user clicks cancel   1. System will prompt user to confirm cancel 2. User declines 3. Use Case resumes on step 4 of normal flow | | |
| **Exceptions:** | | 5a. In step 5 of the normal flow, if the form data is not valid   1. System notifies user form data invalid 2. Use Case resumes on step 3 of normal flow   6a. In step 6 of the normal flow, if there is an error persisting the data   1. System notifies user there was an error 2. Use Case resumes on step 3 of normal flow | | |
| **Includes:** | |  | | |
| **Frequency of Use:** | | 1 per month | | |
| **Special Requirements:** | |  | | |
| **Assumptions:** | |  | | |
| **Notes and Issues:** | | 1. What is definition of league information (Name, genre, league start date, league end date, registration start date, registration end date, etc.)? | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID:** | UC\_Leag\_006 | | | |
| **Use Case Name:** | Copy a Season | | | |
| **Created By:** | Toney Hale | | **Last Updated By:** | Toney Hale |
| **Date Created:** | 9/28/2022 | | **Last Revision Date:** | 9/28/2022 |
| **Actors:** | | League Admin (primary)  System (secondary) | | |
| **Description:** | | Grant league admin ability to duplicate an existing season in the system. | | |
| **Trigger:** | | League Admin clicks Copy Season link | | |
| **Preconditions:** | | 1. User is authenticated as League Admin 2. User is at “List Leagues” view 3. User Clicks League they are wanting to copy | | |
| **Postconditions:** | | 1. New Season information is added to system 2. User is notified of information addition success\failure 3. User is returned to “List Leagues” view | | |
| **Normal Flow:** | | 1. User clicks “Copy Season” link 2. System displays “Add Season” view with details preloaded 3. User verifies\updates data 4. User clicks ok 5. System validates form data 6. System adds season information to database 7. System displays success message 8. System displays “List Leagues” view | | |
| **Alternative Flows:**  **[Alternative Flow 1 – User Clicks Cancel]** | | 4a. In step 4 of the normal flow, if the user clicks cancel  1. System will prompt user to confirm cancel  2. User accepts  3. User is returned to “List Leagues” view  4b. In step 4 of the normal flow, if the user clicks cancel   1. System will prompt user to confirm cancel 2. User declines   Use Case resumes on step 4 of normal flow | | |
| **Exceptions:** | | 5a. In step 5 of the normal flow, if the form data is not valid   1. System notifies user form data invalid 2. Use Case resumes on step 3 of normal flow   6a. In step 6 of the normal flow, if there is an error persisting the data   1. System notifies user there was an error 2. Use Case resumes on step 3 of normal flow | | |
| **Includes:** | |  | | |
| **Frequency of Use:** | | 1 per month | | |
| **Special Requirements:** | |  | | |
| **Assumptions:** | |  | | |
| **Notes and Issues:** | |  | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID:** | UC\_Leag\_006 | | | |
| **Use Case Name:** | Copy a Season | | | |
| **Created By:** | Toney Hale | | **Last Updated By:** | Toney Hale |
| **Date Created:** | 9/28/2022 | | **Last Revision Date:** | 9/28/2022 |
| **Actors:** | | League Admin (primary)  System (secondary) | | |
| **Description:** | | Grant league admin ability to duplicate an existing season in the system. | | |
| **Trigger:** | | League Admin clicks Copy Season link | | |
| **Preconditions:** | | 1. User is authenticated as League Admin 2. User is at “List Leagues” view 3. User Clicks League they are wanting to copy | | |
| **Postconditions:** | | 1. New Season information is added to system 2. User is notified of information addition success\failure 3. User is returned to “List Leagues” view | | |
| **Normal Flow:** | | 1. User clicks “Copy Season” link 2. System displays “Add Season” view with details preloaded 3. User verifies\updates data 4. User clicks ok 5. System validates form data 6. System adds season information to database 7. System displays success message 8. System displays “List Leagues” view | | |
| **Alternative Flows:**  **[Alternative Flow 1 – User Clicks Cancel]** | | 4a. In step 4 of the normal flow, if the user clicks cancel  1. System will prompt user to confirm cancel  2. User accepts  3. User is returned to “List Leagues” view  4b. In step 4 of the normal flow, if the user clicks cancel   1. System will prompt user to confirm cancel 2. User declines   Use Case resumes on step 4 of normal flow | | |
| **Exceptions:** | | 5a. In step 5 of the normal flow, if the form data is not valid   1. System notifies user form data invalid 2. Use Case resumes on step 3 of normal flow   6a. In step 6 of the normal flow, if there is an error persisting the data   1. System notifies user there was an error 2. Use Case resumes on step 3 of normal flow | | |
| **Includes:** | |  | | |
| **Frequency of Use:** | | 1 per month | | |
| **Special Requirements:** | |  | | |
| **Assumptions:** | |  | | |
| **Notes and Issues:** | |  | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID:** | UC\_Leag-008 | | | |
| **Use Case Name:** | Set League to Open | | | |
| **Created By:** | Toney Hale | | **Last Updated By:** | Toney Hale |
| **Date Created:** | 9/12/2022 | | **Last Revision Date:** | 9/20/2022 |
| **Actors:** | | League Admin (primary)  System (secondary) | | |
| **Description:** | | Grant league admin ability to set league start time, allowing users to search and register for it. | | |
| **Trigger:** | | League Admin clicks Add League link | | |
| **Preconditions:** | | 1. User is authenticated as League Admin 2. User is at “List Leagues” view | | |
| **Postconditions:** | | 1. League information is added to system 2. User is notified of information addition success\failure 3. User is returned to “List Leagues” view | | |
| **Normal Flow:** | | 1. User clicks “Add League” link 2. System displays “Add League” view 3. User enters league information of dates/start times 4. User clicks ok 5. System validates form data 6. System adds league information to database 7. System displays success message 8. System displays “List Leagues” view | | |
| **Alternative Flows:**  **[Alternative Flow 1 – User Clicks Cancel]** | | 4a. In step 4 of the normal flow, if the user clicks cancel  1. System will prompt user to confirm cancel  2. User accepts  3. User is returned to “List Leagues” view  4b. In step 4 of the normal flow, if the user clicks cancel   1. System will prompt user to confirm cancel 2. User declines 3. Use Case resumes on step 4 of normal flow | | |
| **Exceptions:** | | 5a. In step 5 of the normal flow, if the form data is not valid   1. System notifies user form data invalid 2. Use Case resumes on step 3 of normal flow   6a. In step 6 of the normal flow, if there is an error persisting the data   1. System notifies user there was an error 2. Use Case resumes on step 3 of normal flow | | |
| **Includes:** | |  | | |
| **Frequency of Use:** | | Start of new season | | |
| **Special Requirements:** | |  | | |
| **Assumptions:** | | Assumes league has already been created | | |
| **Notes and Issues:** | |  | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID:** | UC\_Leag\_009 | | | |
| **Use Case Name:** | Set League to Close | | | |
| **Created By:** | Toney Hale | | **Last Updated By:** | Toney Hale |
| **Date Created:** | 9/12/2022 | | **Last Revision Date:** | 9/20/2022 |
| **Actors:** | | League Admin (primary)  System (secondary) | | |
| **Description:** | | Grant league admin ability to close a league and disable attributes, removing it remove “List Leagues” no longer allowing users to search or register for it. | | |
| **Trigger:** | | League Admin clicks Add League link | | |
| **Preconditions:** | | 1. User is authenticated as League Admin 2. User is at “List Leagues” view | | |
| **Postconditions:** | | 1. League information is added to system 2. User is notified of information addition success\failure 3. User is returned to “List Leagues” view | | |
| **Normal Flow:** | | 1. User clicks “Add League” link 2. System displays “Add League” view 3. User enters league information of dates/end times 4. User clicks ok 5. System validates form data 6. System adds league information to database 7. System displays success message 8. System displays “List Leagues” view | | |
| **Alternative Flows:**  **[Alternative Flow 1 – User Clicks Cancel]** | | 4a. In step 4 of the normal flow, if the user clicks cancel  1. System will prompt user to confirm cancel  2. User accepts  3. User is returned to “List Leagues” view  4b. In step 4 of the normal flow, if the user clicks cancel   1. System will prompt user to confirm cancel 2. User declines 3. Use Case resumes on step 4 of normal flow | | |
| **Exceptions:** | | 5a. In step 5 of the normal flow, if the form data is not valid   1. System notifies user form data invalid 2. Use Case resumes on step 3 of normal flow   6a. In step 6 of the normal flow, if there is an error persisting the data   1. System notifies user there was an error 2. Use Case resumes on step 3 of normal flow | | |
| **Includes:** | |  | | |
| **Frequency of Use:** | | Start of new season | | |
| **Special Requirements:** | |  | | |
| **Assumptions:** | | Assumes league has already been created | | |
| **Notes and Issues:** | | 1. What is my Use Case ID 2. What is definition of league information (Name, genre, league start date, league end date, registration start date, registration end date, etc.)? | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID:** | UC\_Leag-010 | | | |
| **Use Case Name:** | Add a Team | | | |
| **Created By:** | Toney Hale | | **Last Updated By:** | Toney Hale |
| **Date Created:** | 9/12/2022 | | **Last Revision Date:** | 9/20/2022 |
| **Actors:** | | League Admin (primary)  System (secondary) | | |
| **Description:** | | Grant league admin ability to copy a league and set new attributes, allowing users to search and register for it. | | |
| **Trigger:** | | League Admin clicks League link | | |
| **Preconditions:** | | 1. User is authenticated as League Admin 2. User is at “List Leagues” view 3. User clicks what league they want to add too | | |
| **Postconditions:** | | 1. Team information is added to system 2. User is notified of information addition success\failure 3. User is returned to Current League view | | |
| **Normal Flow:** | | 1. User clicks “Add Team” link 2. System displays “Add Team” view 3. User enters team information 4. User clicks ok 5. System validates form data 6. System adds team information to the league 7. System displays success message 8. System displays Current League view | | |
| **Alternative Flows:**  **[Alternative Flow 1 – User Clicks Cancel]** | | 4a. In step 4 of the normal flow, if the user clicks cancel  1. System will prompt user to confirm cancel  2. User accepts  3. User is returned to Current League view  4b. In step 4 of the normal flow, if the user clicks cancel   1. System will prompt user to confirm cancel 2. User declines 3. Use Case resumes on step 4 of normal flow | | |
| **Exceptions:** | | 5a. In step 5 of the normal flow, if the form data is not valid   1. System notifies user form data invalid 2. Use Case resumes on step 3 of normal flow   6a. In step 6 of the normal flow, if there is an error persisting the data   1. System notifies user there was an error 2. Use Case resumes on step 3 of normal flow | | |
| **Includes:** | |  | | |
| **Frequency of Use:** | |  | | |
| **Special Requirements:** | |  | | |
| **Assumptions:** | |  | | |
| **Notes and Issues:** | | 1. What is my Use Case ID 2. What is definition of league information (Name, genre, league start date, league end date, registration start date, registration end date, etc.)? | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID:** | UC\_Leag-011 | | | |
| **Use Case Name:** | Remove a Team | | | |
| **Created By:** | Toney Hale | | **Last Updated By:** | Toney Hale |
| **Date Created:** | 9/12/2022 | | **Last Revision Date:** | 9/20/2022 |
| **Actors:** | | League Admin (primary)  System (secondary) | | |
| **Description:** | | Grant league admin ability to copy a league and set new attributes, allowing users to search and register for it. | | |
| **Trigger:** | | League Admin clicks League link | | |
| **Preconditions:** | | 1. User is authenticated as League Admin 2. User is at “List Leagues” view 3. User clicks what league they want to remove from | | |
| **Postconditions:** | | 1. Team information is removed from the league 2. User is notified of information addition success\failure 3. User is returned to Current League view | | |
| **Normal Flow:** | | 1    User clicks “Remove Team” link   1. System displays “Remove Team” view 2. User clicks what team they want to remove 3. User clicks ok 4. System validates form data 5. System removes team information from league 6. System displays success message 7. System displays Current League view | | |
| **Alternative Flows:**  **[Alternative Flow 1 – User Clicks Cancel]** | | 4a. In step 4 of the normal flow, if the user clicks cancel  1. System will prompt user to confirm cancel  2. User accepts  3. User is returned to Current League view  4b. In step 4 of the normal flow, if the user clicks cancel   1. System will prompt user to confirm cancel 2. User declines 3. Use Case resumes on step 4 of normal flow | | |
| **Exceptions:** | | 5a. In step 5 of the normal flow, if the form data is not valid   1. System notifies user form data invalid 2. Use Case resumes on step 3 of normal flow   6a. In step 6 of the normal flow, if there is an error persisting the data   1. System notifies user there was an error 2. Use Case resumes on step 3 of normal flow | | |
| **Includes:** | |  | | |
| **Frequency of Use:** | |  | | |
| **Special Requirements:** | |  | | |
| **Assumptions:** | |  | | |
| **Notes and Issues:** | | 1. What is my Use Case ID 2. What is definition of league information (Name, genre, league start date, league end date, registration start date, registration end date, etc.)? | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID:** | Leag\_012 | | | |
| **Use Case Name:** | Message a Participant | | | |
| **Created By:** | Toney Hale | | **Last Updated By:** | Jacob Lindauer |
| **Date Created:** | 9/12/2022 | | **Last Revision Date:** | 11/27/2022 |
| **Actors:** | | League Admin (primary)  System (secondary) | | |
| **Description:** | | League admin messaging a participant in their league. | | |
| **Trigger:** | | League Admin has a message they would like to send to a user in a league. | | |
| **Preconditions:** | | 1. User is at “List Leagues” view 2. User clicks a league to view its list of participants. 3. User selects the user they want to message. | | |
| **Postconditions:** | | 1. Selected User receives a notification from the system for their new message. 2. User is returned to Current League view | | |
| **Normal Flow:** | | 1    User clicks “Message Participant” link   1. System displays “Message” view 2. User enters the message 3. User clicks Send Message button 4. Selected user receives a notification from the system about their new message. 5. Selected user is able to view the sent message from the Message screen. | | |
| **Alternative Flows:** | | (None) | | |
| **Exceptions:** | | (None) | | |
| **Includes:** | |  | | |
| **Frequency of Use:** | |  | | |
| **Special Requirements:** | |  | | |
| **Assumptions:** | | Assumes the selected user is set as active. | | |
| **Notes and Issues:** | |  | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID:** | Leag\_013 | | | |
| **Use Case Name:** | Message All Participants | | | |
| **Created By:** | Toney Hale | | **Last Updated By:** | Jacob Lindauer |
| **Date Created:** | 9/12/2022 | | **Last Revision Date:** | 11/27/2022 |
| **Actors:** | | League Admin (primary)  System (secondary) | | |
| **Description:** | | Grant league admin ability to message all participants in a league | | |
| **Trigger:** | | League Admin clicks “Message All Participants” Link | | |
| **Preconditions:** | | 1. User is authenticated as League Admin 2. User is at “List Leagues” view 3. User clicks what league they want to message | | |
| **Postconditions:** | | 1. User is notified of information addition success\failure 2. User is returned to Current League view | | |
| **Normal Flow:** | | 1    User clicks “Message All Participants” link   1. System displays “Message” view 2. User enters the message 3. User clicks ok 4. System validates form data 5. System displays success message 6. All Participants (Users) receive a notification from the system for the sent message. 7. System displays Current League message board | | |
| **Alternative Flows:**  **[Alternative Flow 1 – User Clicks Cancel]** | | 4a. In step 4 of the normal flow, if the user clicks cancel  1. System will prompt user to confirm cancel  2. User accepts  3. User is returned to Current League view  4b. In step 4 of the normal flow, if the user clicks cancel   1. System will prompt user to confirm cancel 2. User declines 3. Use Case resumes on step 4 of normal flow | | |
| **Exceptions:** | | 5a. In step 5 of the normal flow, if the form data is not valid   1. System notifies user form data invalid 2. Use Case resumes on step 3 of normal flow   6a. In step 6 of the normal flow, if there is an error persisting the data   1. System notifies user there was an error 2. Use Case resumes on step 3 of normal flow | | |
| **Includes:** | |  | | |
| **Frequency of Use:** | |  | | |
| **Special Requirements:** | |  | | |
| **Assumptions:** | |  | | |
| **Notes and Issues:** | |  | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID:** | UC\_Leag-014 | | | |
| **Use Case Name:** | Copy a League | | | |
| **Created By:** | Toney Hale | | **Last Updated By:** | Toney Hale |
| **Date Created:** | 9/12/2022 | | **Last Revision Date:** | 9/20/2022 |
| **Actors:** | | League Admin (primary)  System (secondary) | | |
| **Description:** | | Grant league admin ability to copy a league and set new attributes, allowing users to search and register for it. | | |
| **Trigger:** | | League Admin clicks Copy League link | | |
| **Preconditions:** | | 1. User is authenticated as League Admin 2. User is at “List Leagues” view | | |
| **Postconditions:** | | 1. League information is added to system 2. User is notified of information addition success\failure 3. User is returned to “List Leagues” view | | |
| **Normal Flow:** | | 1. User clicks “Copy League” link 2. System displays “Copy League” view 3. User enters league information 4. User clicks ok 5. System validates form data 6. System adds league information to database 7. System displays success message 8. System displays “List Leagues” view | | |
| **Alternative Flows:**  **[Alternative Flow 1 – User Clicks Cancel]** | | 4a. In step 4 of the normal flow, if the user clicks cancel  1. System will prompt user to confirm cancel  2. User accepts  3. User is returned to “List Leagues” view  4b. In step 4 of the normal flow, if the user clicks cancel   1. System will prompt user to confirm cancel 2. User declines 3. Use Case resumes on step 4 of normal flow | | |
| **Exceptions:** | | 5a. In step 5 of the normal flow, if the form data is not valid   1. System notifies user form data invalid 2. Use Case resumes on step 3 of normal flow   6a. In step 6 of the normal flow, if there is an error persisting the data   1. System notifies user there was an error 2. Use Case resumes on step 3 of normal flow | | |
| **Includes:** | |  | | |
| **Frequency of Use:** | | Start of new season | | |
| **Special Requirements:** | |  | | |
| **Assumptions:** | |  | | |
| **Notes and Issues:** | |  | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID:** | Leag\_015 | | | |
| **Use Case Name:** | Generate List of Games | | | |
| **Created By:** | Toney Hale | | **Last Updated By:** | Jacob Lindauer |
| **Date Created:** | 9/12/2022 | | **Last Revision Date:** | 11/27/2022 |
| **Actors:** | | League Admin (primary)  System (secondary) | | |
| **Description:** | | Grant league admin ability to generate a list of games, allowing users to see game times | | |
| **Trigger:** | | League Admin clicks “Generate Schedule” Link | | |
| **Preconditions:** | | 1. User is authenticated as League Admin 2. User is at “List Leagues” view 3. User clicks what league they want to generate games for | | |
| **Postconditions:** | | 1. Team information is removed from the league 2. User is notified of information addition success\failure 3. User is returned to Current League view | | |
| **Normal Flow:** | | 1    User clicks “Generate Schedule” link   1. System displays “Generate Schedule” view 2. User clicks what teams they want to include and start times 3. User clicks ok 4. System validates form data 5. System generates the game schedule 6. System displays success message 7. System displays Current League view | | |
| **Alternative Flows:**  **[Alternative Flow 1 – User Clicks Cancel]** | | 4a. In step 4 of the normal flow, if the user clicks cancel  1. System will prompt user to confirm cancel  2. User accepts  3. User is returned to Current League view  4b. In step 4 of the normal flow, if the user clicks cancel   1. System will prompt user to confirm cancel 2. User declines 3. Use Case resumes on step 4 of normal flow | | |
| **Exceptions:** | | 5a. In step 5 of the normal flow, if the form data is not valid   1. System notifies user form data invalid 2. Use Case resumes on step 3 of normal flow   6a. In step 6 of the normal flow, if there is an error persisting the data   1. System notifies user there was an error 2. Use Case resumes on step 3 of normal flow | | |
| **Includes:** | |  | | |
| **Frequency of Use:** | | Start of Season | | |
| **Special Requirements:** | |  | | |
| **Assumptions:** | |  | | |
| **Notes and Issues:** | |  | | |

### Tournament Admin

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID:** | UC\_Tour\_001 | | | |
| **Use Case Name:** | Create a Tournament | | | |
| **Created By:** | Calvin Ryan | | **Last Updated By:** | Alex Korte |
| **Date Created:** | 9/12/2022 | | **Last Revision Date:** | 11/07/2022 |
| **Actors:** | | Tournament Admin (primary)  System (secondary) | | |
| **Description:** | | Grant tournament admin ability to add a tournament to system, allowing users to search and register for it. | | |
| **Trigger:** | | Tournament Admin clicks Add Tournament link | | |
| **Preconditions:** | | 1. User is authenticated as Tournament Admin (might be any user, given title of tournament admin once created) 2. User is at “List Tournaments” view | | |
| **Postconditions:** | | 1. Tournament information is added to system 2. User is notified of information addition success\failure 3. User is returned to “List Tournaments” view | | |
| **Normal Flow:** | | 1. User clicks “Add Tournament” link 2. System displays “Add Tournament” view 3. User enters tournament information 4. User clicks ok 5. System validates form data 6. System adds tournament information to database 7. System displays success message 8. System displays “List Tournaments” view | | |
| **Alternative Flows:**  **[Alternative Flow 1 – User Clicks Cancel]** | | 4a. In step 4 of the normal flow, if the user clicks cancel  1. System will prompt user to confirm cancel  2. User accepts  3. User is returned to “List Tournaments” view  4b. In step 4 of the normal flow, if the user clicks cancel   1. System will prompt user to confirm cancel 2. User declines 3. Use Case resumes on step 4 of normal flow | | |
| **Exceptions:** | | 5a. In step 5 of the normal flow, if the form data is not valid   1. System notifies user form data invalid 2. Use Case resumes on step 3 of normal flow   6a. In step 6 of the normal flow, if there is an error persisting the data   1. System notifies user there was an error 2. Use Case resumes on step 3 of normal flow | | |
| **Includes:** | | 1. being a registered user 2. being logged in | | |
| **Frequency of Use:** | | 1 per month | | |
| **Special Requirements:** | |  | | |
| **Assumptions:** | | user knows english  user registered  user knows information required to start | | |
| **Notes and Issues:** | | 1. What is my Use Case ID 2. What is definition of tournament information (Name, genre, tournament start date, tournament end date, registration start date, registration end date, etc.)? | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID:** | UC\_Tour\_002 | | | |
| **Use Case Name:** | Modify a Tournament | | | |
| **Created By:** | Calvin Ryan | | **Last Updated By:** | Alex Korte |
| **Date Created:** | 9/14/2022 | | **Last Revision Date:** | 11/07/2022 |
| **Actors:** | | Tournament Admin (primary)  System (secondary) | | |
| **Description:** | | Grant tournament admin ability to modify a tournament in the system. | | |
| **Trigger:** | | Tournament Admin clicks “Edit Tournament” link | | |
| **Preconditions:** | | 1. User is authenticated as Tournament Admin 2. User is at “List Tournaments” view | | |
| **Postconditions:** | | 1. Tournament information is updated in system 2. User is notified of information update success\failure 3. User is returned to “List Tournaments” view | | |
| **Normal Flow:** | | 1. User clicks “Edit Tournament” link 2. System displays “Edit Tournament” view 3. User enters tournament information 4. User clicks ok 5. System validates form data 6. System updates tournament information in database 7. System displays success message 8. System displays “List Tournaments” view | | |
| **Alternative Flows:**  **[Alternative Flow 1 – User Clicks Cancel]** | | 4a. In step 4 of the normal flow, if the user clicks cancel  1. System will prompt user to confirm cancel  2. User accepts  3. User is returned to “List Tournaments” view  4b. In step 4 of the normal flow, if the user clicks cancel   1. System will prompt user to confirm cancel 2. User declines 3. Use Case resumes on step 4 of normal flow | | |
| **Exceptions:** | | 5a. In step 5 of the normal flow, if the form data is not valid   1. System notifies user form data invalid 2. Use Case resumes on step 3 of normal flow   6a. In step 6 of the normal flow, if there is an error persisting the data   1. System notifies user there was an error 2. Use Case resumes on step 3 of normal flow | | |
| **Includes:** | | sp that update data | | |
| **Frequency of Use:** | | 1 per month | | |
| **Special Requirements:** | | being the tournament admin  being logged in | | |
| **Assumptions:** | | user knows english | | |
| **Notes and Issues:** | | 1. What is my Use Case ID 2. What is definition of tournament information (Name, genre, tournament start date, tournament end date, registration start date, registration end date, etc.)? | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID:** | UC\_Tour\_003 | | | |
| **Use Case Name:** | Copy a Tournament | | | |
| **Created By:** | Calvin Ryan | | **Last Updated By:** | Alex Korte |
| **Date Created:** | 9/14/2022 | | **Last Revision Date:** | 11/07/2022 |
| **Actors:** | | Tournament Admin (primary)  System (secondary) | | |
| **Description:** | | Grant tournament admin ability to duplicate an existing tournament in the system. | | |
| **Trigger:** | | Tournament Admin clicks Copy Tournament link | | |
| **Preconditions:** | | 1. User is authenticated as Tournament Admin 2. User is at “List Tournaments” view | | |
| **Postconditions:** | | 1. New Tournament information is added to system 2. User is notified of information addition success\failure 3. User is returned to “List Tournaments” view | | |
| **Normal Flow:** | | 1. User clicks “Copy Tournament” link 2. System displays “Add Tournament” view with details preloaded 3. User verifies\updates data 4. User clicks ok 5. System validates form data 6. System adds tournament information to database 7. System displays success message 8. System displays “List Tournaments” view | | |
| **Alternative Flows:**  **[Alternative Flow 1 – User Clicks Cancel]** | | 4a. In step 4 of the normal flow, if the user clicks cancel  1. System will prompt user to confirm cancel  2. User accepts  3. User is returned to “List Tournaments” view  4b. In step 4 of the normal flow, if the user clicks cancel   1. System will prompt user to confirm cancel 2. User declines   Use Case resumes on step 4 of normal flow | | |
| **Exceptions:** | | 5a. In step 5 of the normal flow, if the form data is not valid   1. System notifies user form data invalid 2. Use Case resumes on step 3 of normal flow   6a. In step 6 of the normal flow, if there is an error persisting the data   1. System notifies user there was an error 2. Use Case resumes on step 3 of normal flow | | |
| **Includes:** | |  | | |
| **Frequency of Use:** | | 1 per month | | |
| **Special Requirements:** | |  | | |
| **Assumptions:** | |  | | |
| **Notes and Issues:** | | 1. What is definition of tournament information (Name, genre, tournament start date, tournament end date, registration start date, registration end date, etc.)? | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID:** | UC\_Tour\_004 | | | |
| **Use Case Name:** | Modify a Tournament | | | |
| **Created By:** | Calvin Ryan | | **Last Updated By:** | Calvin Ryan |
| **Date Created:** | 9/14/2022 | | **Last Revision Date:** | 9/17/2022 |
| **Actors:** | | Tournament Admin (primary)  System (secondary) | | |
| **Description:** | | Grant tournament admin ability to remove a tournament from the system. | | |
| **Trigger:** | | Tournament Admin clicks Remove Tournament link | | |
| **Preconditions:** | | 1. User is authenticated as Tournament Admin 2. User is at “List Tournaments” view | | |
| **Postconditions:** | | 1. Tournament information is deactivated in system 2. User is notified of information update success\failure 3. User is returned to “List Tournaments” view | | |
| **Normal Flow:** | | 1. User clicks “Remove Tournament” link 2. System will prompt user to confirm desire to remove 3. User clicks ok 4. System updates tournament information in database 5. System displays success message 6. System displays “List Tournaments” view | | |
| **Alternative Flows:**  **[Alternative Flow 1 – User Clicks Cancel]** | | 3a. In step 3 of the normal flow, if the user clicks cancel  1. User is returned to “List Tournaments” view | | |
| **Exceptions:** | | 4a. In step 4 of the normal flow, if there is an error persisting the data   1. System notifies user there was an error 2. Use Case resumes on step 3 of normal flow | | |
| **Includes:** | |  | | |
| **Frequency of Use:** | | 1 per month | | |
| **Special Requirements:** | |  | | |
| **Assumptions:** | |  | | |
| **Notes and Issues:** | | 1. What is definition of tournament information (Name, genre, tournament start date, tournament end date, registration start date, registration end date, etc.)? | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID:** | UC\_Tour\_005 | | | |
| **Use Case Name:** | Open Registration | | | |
| **Created By:** | Calvin Ryan | | **Last Updated By:** | Calvin Ryan |
| **Date Created:** | 9/17/2022 | | **Last Revision Date:** | 9/17/2022 |
| **Actors:** | | Tournament Admin (primary)  System (secondary) | | |
| **Description:** | | Grant tournament admin ability to open registration, allowing entities to register for the tournament. | | |
| **Trigger:** | | Tournament Admin clicks Open Registration link | | |
| **Preconditions:** | | 1. User is authenticated as Tournament Admin 2. User is at “List Registrations” view | | |
| **Postconditions:** | | 1. Tournament registrations status is updated to open in system 2. User is notified of information update success\failure 3. User is returned to “List Registrations” view | | |
| **Normal Flow:** | | 1. User clicks “Open Registration” link 2. System asks user to confirm desire to open registration 3. User clicks ok 4. System updates registration information in database 5. System displays success message 6. System displays “List Registrations” view | | |
| **Alternative Flows:**  **[Alternative Flow 1 – User Clicks Cancel]** | | 3a. In step 3 of the normal flow, if the user clicks cancel   1. User is returned to “List Registrations” view | | |
| **Exceptions:** | | 4a. In step 4 of the normal flow, if there is an error persisting the data   1. System notifies user there was an error 2. User is returned to “List Registrations” view | | |
| **Includes:** | |  | | |
| **Frequency of Use:** | | 1 per month | | |
| **Special Requirements:** | |  | | |
| **Assumptions:** | |  | | |
| **Notes and Issues:** | | 1. Is this something the system can do automatically based on a parameter set when creating the tournament? | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID:** | UC\_Tour\_006 | | | |
| **Use Case Name:** | Close Registration | | | |
| **Created By:** | Calvin Ryan | | **Last Updated By:** | Calvin Ryan |
| **Date Created:** | 9/17/2022 | | **Last Revision Date:** | 9/17/2022 |
| **Actors:** | | Tournament Admin (primary)  System (secondary) | | |
| **Description:** | | Grant tournament admin ability to close registration, preventing further entities from registering. | | |
| **Trigger:** | | Tournament Admin clicks Close Registration link | | |
| **Preconditions:** | | 1. User is authenticated as Tournament Admin 2. User is at “List Registrations” view | | |
| **Postconditions:** | | 1. Tournament registration status is updated to closed in system 2. User is notified of information update success\failure 3. User is returned to “List Registrations” view | | |
| **Normal Flow:** | | 1. User clicks “Close Registration” link 2. System asks user to confirm desire to close registration 3. User clicks ok 4. System updates registration information in database 5. System displays success message 6. System displays “List Registrations” view | | |
| **Alternative Flows:**  **[Alternative Flow 1 – User Clicks Cancel]** | | 3a. In step 3 of the normal flow, if the user clicks cancel   1. User is returned to “List Registrations” view | | |
| **Exceptions:** | | 4a. In step 4 of the normal flow, if there is an error persisting the data   1. System notifies user there was an error 2. User is returned to “List Registrations” view | | |
| **Includes:** | |  | | |
| **Frequency of Use:** | | 1 per month | | |
| **Special Requirements:** | |  | | |
| **Assumptions:** | |  | | |
| **Notes and Issues:** | | 1. Is this something the system can do automatically based on a parameter set when creating the tournament? | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID:** | UC\_Tour\_007 | | | |
| **Use Case Name:** | Add Team to Tournament | | | |
| **Created By:** | Calvin Ryan | | **Last Updated By:** | Calvin Ryan |
| **Date Created:** | 9/16/2022 | | **Last Revision Date:** | 9/17/2022 |
| **Actors:** | | Tournament Admin (primary)  System (secondary) | | |
| **Description:** | | Grant tournament admin ability to add a team to a tournament in the system. | | |
| **Trigger:** | | Tournament Admin clicks Add Team(s) link | | |
| **Preconditions:** | | 1. User is authenticated as Tournament Admin 2. User is at “List Teams” view | | |
| **Postconditions:** | | 1. Team information is associated with Tournament in system 2. User is notified of information addition success\failure 3. User is returned to “List Teams” view | | |
| **Normal Flow:** | | 1. User clicks “Add Team(s)” link 2. System displays “Add Team(s)” view 3. User selects team(s) to add 4. User clicks ok 5. System adds association(s) to database 6. System displays success message 7. System displays “List Teams” view | | |
| **Alternative Flows:**  **[Alternative Flow 1 – User Clicks Cancel]** | | 4a. In step 4 of the normal flow, if the user clicks cancel  1. System will prompt user to confirm cancel  2. User accepts  3. User is returned to “List Teams” view  4b. In step 4 of the normal flow, if the user clicks cancel   1. System will prompt user to confirm cancel 2. User declines 3. Use Case resumes on step 4 of normal flow | | |
| **Exceptions:** | | 5a. In step 5 of the normal flow, if there is an error persisting the data   1. System notifies user there was an error 2. Use Case resumes on step 3 of normal flow | | |
| **Includes:** | |  | | |
| **Frequency of Use:** | | 10 per month | | |
| **Special Requirements:** | |  | | |
| **Assumptions:** | |  | | |
| **Notes and Issues:** | |  | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID:** | UC\_Tour\_008 | | | |
| **Use Case Name:** | Remove Team from Tournament | | | |
| **Created By:** | Calvin Ryan | | **Last Updated By:** | Calvin Ryan |
| **Date Created:** | 9/16/2022 | | **Last Revision Date:** | 9/17/2022 |
| **Actors:** | | Tournament Admin (primary)  System (secondary) | | |
| **Description:** | | Grant tournament admin ability to remove a team from a tournament in the system. | | |
| **Trigger:** | | Tournament Admin clicks Remove Team link | | |
| **Preconditions:** | | 1. User is authenticated as Tournament Admin 2. User is at “List Teams” view | | |
| **Postconditions:** | | 1. Team\Tournament association is removed from system 2. User is notified of information removal success\failure 3. User is returned to “List Teams” view | | |
| **Normal Flow:** | | 1. User clicks “Remove Team” link 2. System prompts user confirming desire to remove 3. User clicks ok 4. System removes association from database 5. System displays success message 6. System displays “List Teams” view | | |
| **Alternative Flows:**  **[Alternative Flow 1 – User Clicks Cancel]** | | 3a. In step 3 of the normal flow, if the user clicks cancel  1. User is returned to “List Teams” view | | |
| **Exceptions:** | | 4a. In step 4 of the normal flow, if there is an error persisting the data   1. System notifies user there was an error 2. User is returned to “List Teams” view | | |
| **Includes:** | |  | | |
| **Frequency of Use:** | | 2 per month | | |
| **Special Requirements:** | |  | | |
| **Assumptions:** | |  | | |
| **Notes and Issues:** | |  | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID:** | UC\_Tour\_011 | | | |
| **Use Case Name:** | Generate Matches | | | |
| **Created By:** | Calvin Ryan | | **Last Updated By:** | Calvin Ryan |
| **Date Created:** | 9/17/2022 | | **Last Revision Date:** | 9/17/2022 |
| **Actors:** | | Tournament Admin (primary)  System (secondary) | | |
| **Description:** | | Grant tournament admin ability to generate a list of matches between entities. | | |
| **Trigger:** | | Tournament Admin clicks Generate Matches link | | |
| **Preconditions:** | | 1. User is authenticated as Tournament Admin 2. User is at “List Matches” view | | |
| **Postconditions:** | | 1. Match information is added to system 2. User is notified of information addition success\failure 3. User is returned to “List Matches” view | | |
| **Normal Flow:** | | 1. User clicks “Generate Matches” link 2. System displays “Generate Matches” view 3. User enters match requirement\threshold criteria 4. User clicks ok 5. System validates form data 6. System generates list of matches meeting criteria in step 3 7. System asks user to approve generated matches 8. User clicks ok 9. System adds match(es) information to database 10. System displays success message 11. System displays “List Matches” view | | |
| **Alternative Flows:**  **[Alternative Flow 1 – User Clicks Cancel]**    **[Alternative Flow 2 – User Clicks Cancel]** | | 4a. In step 4 of the normal flow, if the user clicks cancel  1. System will prompt user to confirm cancel  2. User accepts  3. User is returned to “List Matches” view  4b. In step 4 of the normal flow, if the user clicks cancel   1. System will prompt user to confirm cancel 2. User declines 3. Use Case resumes on step 4 of normal flow   8a. In step 8 of the normal flow, if the user clicks cancel   1. System will prompt user to confirm cancel 2. User confirms 3. Use Case resumes on step 4 of the normal flow   8b. In step 8 of the normal flow, if the user clicks cancel   1. System will prompt user to confirm cancel 2. User declines 3. Use Case resumes on step 8 of the normal flow | | |
| **Exceptions:** | | 5a. In step 5 of the normal flow, if the form data is not valid   1. System notifies user form data invalid 2. Use Case resumes on step 3 of normal flow   9a. In step 6 of the normal flow, if there is an error persisting the data   1. System notifies user there was an error 2. Use Case resumes on step 4 of normal flow | | |
| **Includes:** | |  | | |
| **Frequency of Use:** | | 1 per month | | |
| **Special Requirements:** | |  | | |
| **Assumptions:** | |  | | |
| **Notes and Issues:** | | 1. What is match requirement\threshold criteria (entities to schedule, venues available, length of matches, buffer time between matches, each entity must have at least x, no more than y matches against each other entity, etc)? | | |

### Coach

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID:** | Coach\_001 | | | |
| **Use Case Name:** | View a list of teams | | | |
| **Created By:** | Alex | | **Last Updated By:** | Jacob Lindauer |
| **Date Created:** | 10/2/2022 | | **Last Revision Date:** | 11/27/2022 |
| **Actors:** | | Coach  Admin  Organizer  User  System | | |
| **Description:** | | Users will be able to see a list of teams scheduled to attend an event. | | |
| **Trigger:** | | User will trigger event by going to view event details. | | |
| **Preconditions:** | | User must be registered. | | |
| **Postconditions:** | | 1. System will display teams registered for an event | | |
| **Normal Flow:** | | 1. User will navigate schedule 2. User will click on an event 3. System will display list of teams and details about event | | |
| **Alternative Flows:** | | 1a. User is not registered.  2a. System will prompt user to log in  1b. Schedule does not display event  2b.  System will prompt user that there are no scheduled events | | |
| **Exceptions:** | |  | | |
| **Includes:** | | Commentator\_004 where a user can view players in an event | | |
| **Frequency of Use:** | | on demand | | |
| **Special Requirements:** | | User being registered | | |
| **Assumptions:** | |  | | |
| **Notes and Issues:** | |  | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID:** | Coach\_002 | | | |
| **Use Case Name:** | Create a Team | | | |
| **Created By:** | Calvin Ryan | | **Last Updated By:** | Jacob Lindauer |
| **Date Created:** | 9/28/2022 | | **Last Revision Date:** | 11/27/2022 |
| **Actors:** | | Coach (primary)  System (secondary) | | |
| **Description:** | | Grant coach ability to add a team to the system, allowing users to search and register for it. | | |
| **Trigger:** | | Coach clicks the “Create Team” option under the Teams menu. | | |
| **Preconditions:** | | 1. User is authenticated as Coach 2. User hovers over the Teams drop-down list 3. User selected the “Create Team” option. | | |
| **Postconditions:** | | 1. Team information is added to system 2. User is notified of information addition success\failure 3. User is returned to “List Teams” view | | |
| **Normal Flow:** | | 1. User hovers over the Team’s drop-down list. 2. User selected the “Create Team” option. 3. User fills out the new team form. 4. User clicks Save. 5. System validates form data 6. System adds team information to database 7. System displays success message 8. System displays “List Teams” view | | |
| **Alternative Flows:**  **[Alternative Flow 1 – User Clicks Cancel]** | | 4a. In step 4 of the normal flow, if the user clicks cancel  1. System will prompt user to confirm cancel  2. User accepts  3. User is returned to “List Teams” view  4b. In step 4 of the normal flow, if the user clicks cancel   1. System will prompt user to confirm cancel 2. User declines 3. Use Case resumes on step 4 of normal flow | | |
| **Exceptions:** | | 5a. In step 5 of the normal flow, if the form data is not valid   1. System notifies user form data invalid 2. Use Case resumes on step 3 of normal flow   6a. In step 6 of the normal flow, if there is an error persisting the data   1. System notifies user there was an error 2. Use Case resumes on step 3 of normal flow | | |
| **Includes:** | |  | | |
| **Frequency of Use:** | | On Demand | | |
| **Special Requirements:** | |  | | |
| **Assumptions:** | |  | | |
| **Notes and Issues:** | |  | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Use Case ID:** | Coach\_003 | | |
| **Use Case Name:** | Modify a Team Profile | | |
| **Created By:** | Calvin Ryan | **Last Updated By:** | Jacob Lindauer |
| **Date Created:** | 9/28/2022 | **Last Revision Date:** | 11/27/2022 |
| **Actors:** | | /Team Admin (primary)  System (secondary) | |
| **Description:** | | Grant team admin ability to modify a team in the system. | |
| **Trigger:** | | Team Admin clicks “Edit Team” link | |
| **Preconditions:** | | 1. User is at “List Teams” view 2. User selects a team they would like to make changes to. | |
| **Postconditions:** | | 1. Team information is updated in system 2. User is notified of information update success\failure 3. User is sent to updated team page | |
| **Normal Flow:** | | 1. User navigates to the team page for the team they would like to edit. 2. User clicks “Edit Team” link 3. System displays “Edit Team” view 4. User enters team information 5. User clicks ok 6. System validates form data 7. System updates team information in database 8. System displays success message 9. System displays “List Teams” view | |
| **Alternative Flows:**  **[Alternative Flow 1 – User Clicks Cancel]** | | 4a. In step 4 of the normal flow, if the user clicks cancel  1. System will prompt user to confirm cancel  2. User accepts  3. User is returned to “List Teams” view  4b. In step 4 of the normal flow, if the user clicks cancel   1. System will prompt user to confirm cancel 2. User declines 3. Use Case resumes on step 4 of normal flow | |
| **Exceptions:** | | 5a. In step 5 of the normal flow, if the form data is not valid   1. System notifies user form data invalid 2. Use Case resumes on step 3 of normal flow   6a. In step 6 of the normal flow, if there is an error persisting the data   1. System notifies user there was an error 2. Use Case resumes on step 3 of normal flow | |
| **Includes:** | |  | |
| **Frequency of Use:** | | 1 per month | |
| **Special Requirements:** | |  | |
| **Assumptions:** | |  | |
| **Notes and Issues:** | |  | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID:** | Coach\_004 | | | |
| **Use Case Name:** | Remove a team from an event | | | |
| **Created By:** | Alex | | **Last Updated By:** | Jacob Lindauer |
| **Date Created:** | 10/2/2022 | | **Last Revision Date:** | 11/27/2022 |
| **Actors:** | | Coach  Admin  Organizer  User  System | | |
| **Description:** | | User will be able to remove a team that they registered for an event from the event. | | |
| **Trigger:** | | User will trigger the event by navigating to the scheduled event details, where they will have the option to withdraw from an event. | | |
| **Preconditions:** | | User must be registered, as well as be the captain/coach/have the correct roles | | |
| **Postconditions:** | | 1. System will prompt admin/organizer/user that the team has withdrawn 2. system will prompt user that there was an error removing them | | |
| **Normal Flow:** | | 1. User will navigate schedule 2. User will click on the details of the event 3. user will click option to remove their team from the event 4. system will prompt user to confirm 5. user will confirm 6. system will prompt admin/organizer 7. system will prompt user that they were removed 8. Team will no longer display as a team | | |
| **Alternative Flows:** | | 1a. User is not registered.  2a. System will prompt user to log in | | |
| **Exceptions:** | | System will throw an error if the information does not update correctly | | |
| **Includes:** | | Login  register  schedule  schedule filter | | |
| **Frequency of Use:** | | on demand | | |
| **Special Requirements:** | | User being registered to selected event. | | |
| **Assumptions:** | |  | | |
| **Notes and Issues:** | |  | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID:** | Coach\_005 | | | |
| **Use Case Name:** | Add Player | | | |
| **Created By:** | Nick Vroom | | **Last Updated By:** | Jacob Lindauer |
| **Date Created:** | 9/18/2022 | | **Last Revision Date:** | 11/27/2022 |
| **Actors:** | | Coach (primary)  System (secondary) | | |
| **Description:** | | Grants the coach of a team the ability to add a new player to their team. | | |
| **Trigger:** | | Coach would like to invite player to their team. | | |
| **Preconditions:** | | 1. User is authenticated as Coach 2. User is on “Team” view | | |
| **Postconditions:** | | 1. Tournament information is added to system 2. User is notified of information addition success\failure 3. User is returned to the team page showing the updated roster | | |
| **Normal Flow:** | | 1. User navigates to their team page. 2. User selects the “Add Player” button. 3. User is prompted with a search window to search for user to add. 4. User clicks the “Add User” button 5. System add user to the team roster 6. System sends notification to the added player that they have been added to the team. 7. User is taken back to the team page showing the new player has been added. | | |
| **Alternative Flows:**  **[Alternative Flow 1 – User Clicks Cancel]** | | 4a. In step the user selects cancel instead of “Add User”.   1. User clicks cancel on the add user prompt. 2. User is returned to the team view page and user is not added to the team. | | |
| **Exceptions:** | | 5a. In step 5 of the normal flow, if the form data is not valid   1. System notifies user form data invalid (username not found) 2. Use Case resumes on step 3 of normal flow | | |
| **Includes:** | |  | | |
| **Frequency of Use:** | | As often as needed | | |
| **Special Requirements:** | |  | | |
| **Assumptions:** | | Assumes the searched user is active in the system. | | |
| **Notes and Issues:** | |  | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID:** | Coach\_006 | | | |
| **Use Case Name:** | Edit Player | | | |
| **Created By:** | Nick Vroom | | **Last Updated By:** | Jacob Lindauer |
| **Date Created:** | 9/18/2022 | | **Last Revision Date:** | 11/27/2022 |
| **Actors:** | | Coach (primary)  System (secondary) | | |
| **Description:** | | Grants the coach (owner) of a team the ability to edit a player’s status on a team. | | |
| **Trigger:** | | Coach clicks ‘Edit Roster’ | | |
| **Preconditions:** | | User is authenticated as Coach  User is on “Team” view | | |
| **Postconditions:** | | 1. Team information is added to system 2. User is notified of information addition success\failure 3. User is returned to “Team” view | | |
| **Normal Flow:** | | 1. User clicks “Edit Roster” link 2. User is displayed the team roster. 3. User edits dropdown boxes of Bench/Starting 4. User clicks Save 5. System validates form data 6. System adds team information to database 7. System displays success message 8. System displays “Team” view | | |
| **Alternative Flows:**  **[Alternative Flow 1 – User Clicks Cancel]** | | 4a. User clicks cancel instead of save.   1. User is prompted if they would are sure they would like to cancel. 2. No will return them to roster editing page. 3. Yes will return user to roster page without applying any changes. | | |
| **Exceptions:** | | 5a. In step 5 of the normal flow, if the form data is not valid   1. System notifies user form data invalid 2. Use Case resumes on step 3 of normal flow | | |
| **Includes:** | |  | | |
| **Frequency of Use:** | | As often as needed | | |
| **Special Requirements:** | |  | | |
| **Assumptions:** | |  | | |
| **Notes and Issues:** | |  | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID:** | Coach\_007 | | | |
| **Use Case Name:** | Remove Player | | | |
| **Created By:** | Nick Vroom | | **Last Updated By:** | Jacob Lindauer |
| **Date Created:** | 9/18/2022 | | **Last Revision Date:** | 11/27/2022 |
| **Actors:** | | Coach (primary)  System (secondary) | | |
| **Description:** | | Grants the coach (owner) of a team the ability to remove a player from the team. | | |
| **Trigger:** | | Coach clicks ‘Edit Roster’ | | |
| **Preconditions:** | | 1. User is on “Team” view 2. User has a player they would like to remove from the team. | | |
| **Postconditions:** | | 1. Team information is added to system 2. User is notified of information addition success\failure 3. User is returned to “Team” view | | |
| **Normal Flow:** | | 1. User clicks “Edit Roster” link 2. System displays all players and a remove option by them 3. User clicks Remove by a player’s name 4. System confirms the removal of a player 5. User clicks Confirm 6. System validates form data 7. System adds team information to database 8. System displays success message 9. Systems send a notification to the removed user that they have been removed from the team. 10. System displays “Team” view page | | |
| **Alternative Flows:**  **[Alternative Flow 1 – User Clicks Cancel]** | | 4a. In step 4 of the normal flow, if the user clicks no  1. User stays on “Edit Roster” page | | |
| **Exceptions:** | | 5a. In step 5 of the normal flow, if the form data is not valid   1. System notifies user form data invalid 2. Use Case resumes on step 3 of normal flow | | |
| **Includes:** | |  | | |
| **Frequency of Use:** | | As often as needed | | |
| **Special Requirements:** | |  | | |
| **Assumptions:** | |  | | |
| **Notes and Issues:** | |  | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID:** | Coach-008 | | | |
| **Use Case Name:** | Message All Players | | | |
| **Created By:** | Nick Vroom | | **Last Updated By:** | Jacob Lindauer |
| **Date Created:** | 9/18/2022 | | **Last Revision Date:** | 11/27/2022 |
| **Actors:** | | Coach (primary)  System (secondary) | | |
| **Description:** | | Grants the coach of a team the ability to message all players on his/her team | | |
| **Trigger:** | | Coach clicks 'Message’ | | |
| **Preconditions:** | | User is authenticated as Coach  User is on player profile | | |
| **Postconditions:** | | 1. Message is logged in inbox 2. User is notified of information addition success\failure 3. User is returned to “Team” view | | |
| **Normal Flow:** | | 1. User clicks “Shout” link 2. System displays subject/body text boxes to message players 3. User enters text in textboxes 4. User clicks Send 5. Systems sends the message to all players on the team 6. System sends notification to users for a new message 7. System displays success message 8. User is taken back to the team view page | | |
| **Alternative Flows:** | |  | | |
| **Exceptions:** | | (None) | | |
| **Includes:** | |  | | |
| **Frequency of Use:** | | As often as needed | | |
| **Special Requirements:** | |  | | |
| **Assumptions:** | |  | | |
| **Notes and Issues:** | |  | | |

### Treasurer

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID:** | Trea001 | | | |
| **Use Case Name:** | Request Funds | | | |
| **Created By:** | Alex Korte | | **Last Updated By:** | Jacob Lindauer |
| **Date Created:** | 10/3/2022 | | **Last Revision Date:** | 10/27/2022 |
| **Actors:** | | Treasurer role  system  Admin/Organizer  Venue | | |
| **Description:** | | User with appropriate role will be able to request money from the league dues, in order to help pay for equipment/venue | | |
| **Trigger:** | |  | | |
| **Preconditions:** | | 1. User must be logged in 2. User must have appropriate role | | |
| **Postconditions:** | | 1. System tells user that their request was sent 2. system tells treasurer that there is a request for funds 3. system will tell user that there was an error requesting funds | | |
| **Normal Flow:** | | 1. User will navigate to a team they would like to request funds from. 2. User clicks on the “Request Funds” button on the team view page. 3. User will fill out the request form. 4. System will check that the fields are correct 5. System will confirm with user that they want to submit 6. User will submit their request. 7. Coaches for the team are sent a message from the treasurer with fund request information. 8. System will tell user that request was sent 9. System will redirect user to previous page | | |
| **Alternative Flows:**  **[Alternative Flow 1 – Not in Network]** | | 6a. User cancels request   1. System redirects user to initial request page | | |
| **Exceptions:** | | (None) | | |
| **Includes:** | |  | | |
| **Frequency of Use:** | | On demand | | |
| **Special Requirements:** | |  | | |
| **Assumptions:** | | Assumes treasurer is sending a message to an active team. | | |
| **Notes and Issues:** | |  | | |

## Logistics

### Equipment Manager

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID:** | EquipmentManager\_001 | | | |
|  |  | | | |
| **Use Case Name:** | Keep inventory of equipment and include quality | | | |
| **Created By:** | Izy Doughty | | **Last Updated By:** | Izy Doughty |
| **Date Created:** | 9-25-2022 | | **Last Revision Date:** | 10-3-2022 |
| **Actors:** | | Equipment Manager | | |
| **Description:** | | A premade checklist is created for the Sport the Equipment Manager is playing. From this checklist, the Equipment manager can check off what pieces of equipment they have.  Along with the equipment they have, they can also post the quality of it so that anyone looking at the list may be able to help the team with replacements or donations.  The following are what the quality can be set to: GOOD, NEEDS REPLACEMENT | | |
| **Trigger:** | | When a team signs up for a game or tournament, they must have a completed list to enter. | | |
| **Preconditions:** | | 1. A team needs to tell to their Equipment manager what they have 2. A team needs to have an Equipment Manager set | | |
| **Postconditions:** | | Minimal Guarantee   1. A list is available to use 2. List will show when it was last updated   Maximum Guarantee   1. A custom list may be able to be created 2. Users can add custom equipment to lists | | |
| **Normal Flow:** | | 1. Existing User with the Equipment Manager title logs in 2. Click on My Team section 3. Click on the team they are manager for 4. Use checklist to show what they have. Check off what they have, how many, and what quality each piece is 5. Save updates | | |
| **Alternative Flows:**  **[Alternative Flow 1 – Not in Network]** | | 4a.Click "Create Custom Checklist"   1. Set what Equipment is needed 2. Check them off 3. Save updates | | |
| **Exceptions:** | | 1. If a "Team" does not need Equipment, they do not have a need to set an Equipment Manager | | |
| **Includes:** | |  | | |
| **Frequency of Use:** | | On Demand | | |
| **Special Requirements:** | | A checklist needs to be created for popular sports with equipment.  Users need to be able to create listings per piece of equipment to be able to set the equipment. | | |
| **Assumptions:** | | Teams will set up a list and use it | | |
| **Notes and Issues:** | | 1. Teams will need to have this setup in order to receive help from other users and move forward with their games | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID:** | EquipmentManager\_002 | | | |
|  |  | | | |
| **Use Case Name:** | Share Equipment List with other users | | | |
| **Created By:** | Izy Doughty | | **Last Updated By:** | Izy Doughty |
| **Date Created:** | 9-25-2022 | | **Last Revision Date:** | 10-3-2022 |
| **Actors:** | | Equipment Managers | | |
| **Description:** | | Equipment Managers need an easy way to share their equipment list with other equipment managers so they can know what needs to be brought to their game.  The team can benefit from this functionality, as would sending it on to other kinds of users. | | |
| **Trigger:** | | When an Equipment Manager has saved their updated Equipment List,  a share button should become available to them. | | |
| **Preconditions:** | | 1. The Equipment list must be updated 2. A Equipment manager needs to be appointed 3. Equipment Managers need to be able to communicate with each other and be able to communicate what they can do for the other team | | |
| **Postconditions:** | | Minimal Guarantee   1. The list can be sent via direct message for viewing 2. Equipment list will be public on the Team's profile   Maximum Guarantee   1. The Equipment List can be sent automatically once the teams are set for the next game 2. The Equipment List can be shared from the profile to other users 3. The Equipment List can have a notification sent from it to what is being offered for the game or donation | | |
| **Normal Flow:** | | 1. Existing User with the Equipment Manager title logs in 2. Click on My Team section 3. Click on the team they are manager for and go to their profile 4. They click "Share List" from the list section 5. A dialog shows a list of connections along with a field to search for them. Along with the connections, the next game's equipment manager is made apparent with its own section. 6. The user makes their selection and sends an automatic message template to start the conversation | | |
| **Alternative Flows:**  **[Alternative Flow 1 – Not in Network]** | |  | | |
| **Exceptions:** | | 1. If a user or team does not require an equipment list | | |
| **Includes:** | |  | | |
| **Frequency of Use:** | | On Demand | | |
| **Special Requirements:** | |  | | |
| **Assumptions:** | | Teams will communicate with each other | | |
| **Notes and Issues:** | | 1. A combined list of the venue, and both teams can show what the game may be missing before they start their games. They'll need to check off what is being taken care of if something's missing and send out notifications | | |

### Donor

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID:** | Donor\_004 | | | |
| **Use Case Name:** | UC\_VenueRep\_LST\_007 | | | |
| **Created By:** | Garion Opiola | |  |  |
| **Date Created:** | 9/16/22 | |  |  |
| **Actors:** | | Donor, Venue User | | |
| **Description:** | | View tournament equipment | | |
| **Trigger:** | | hosting a local event | | |
| **Preconditions:** | | 1. User decides to host tournament | | |
| **Postconditions:** | | 1. Set schedule | | |
| **Normal Flow:** | | 1. User goes to site 2. User looks at tournament their hosting 3. User views available equipment | | |
| **Alternative Flows:**  **[Alternative Flow 1 – Not in Network]** | | 2a. In step 2 of the normal flow, if the User doesn’t have an account or not logged in   1. System will prompt User to create account or login 2. User creates account or logs in 3. Use Case resumes on step 3 | | |
| **Exceptions:** | | 2a.   In step 2 of the Alternative flow, if the User enters an invalid username or password   1. Login is denied 2. Message to User to retry login 3. User enters correct login information 4. Use Case resumes on step 3 of normal flow | | |
| **Includes:** | | Search engine | | |
| **Frequency of Use:** | | On demand | | |
| **Special Requirements:** | | Internet connection | | |
| **Assumptions:** | | The User understands English  . | | |
| **Notes and Issues:** | | N/A | | |

### Vendor

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID:** | Vendor\_001 | | | |
| **Use Case Name:** | Venue, Venue Event | | | |
| **Created By:** | Gideon Trevor | | **Last Updated By:** |  |
| **Date Created:** |  | | **Last Revision Date:** |  |
| **Actors:** | | Vendor | | |
| **Description:** | | be able to send a connection request to be approved by a venue be associated with a venue | | |
| **Trigger:** | | Select ‘account options’ | | |
| **Preconditions:** | | **1 have an account**  2 is logged in | | |
| **Postconditions:** | | 1 when viewing my account I should be able to see that I am listed as a vendor and have vendor sections on my profile | | |
| **Normal Flow:** | | 1 The vendor will navigate to the Messages tab  2 Then they will select a “more options” button that sits on the chat that they have started with the venue representative  3 in the drop down provide, since they are a vendor and are targeting a venue Representative they will be able to select “send a associate request”, which if accepted through their notifications tab will make it so they are associated | | |
| **Alternative Flows:**  **Alternant flow 1, no messages already started** | | 2a if the venue representative has not messaged someone that they wish to be associated with  1 go to the search bar and enter any relevant filter or simply browse until they find a venue representative account  2 go to the venue's profile  3 Press "request"  4 do step three of the normal flow | | |
| **Exceptions:** | | 1 you can not edit your account when your account is banned  2 can’t perform this on a group message | | |
| **Includes:** | | May include Arrangement, direct messages, may include notifications | | |
| **Frequency of Use:** | | Not frequently used | | |
| **Special Requirements:** | |  | | |
| **Assumptions:** | |  | | |
| **Notes and Issues:** | |  | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID:** | Vendor\_002 | | | |
| **Use Case Name:** | Vendor: Accept/Deny Requests | | | |
| **Created By:** | Rith Sivaprakash | | **Last Updated By:** | - |
| **Date Created:** | 9-29-2022 | | **Last Revision Date:** | - |
| **Actors:** | | Vendor | | |
| **Description:** | | Vendors can accept and deny requests for their presence at a particular tournament or event. | | |
| **Trigger:** | | Vendor clicks on “requests” tab | | |
| **Preconditions:** | | 1. Has an account as a vendor | | |
| **Postconditions:** | | Minimal Guarantee:   1. Vendor can accept and deny requests   Maximum Guarantee:   1. Vendor can successfully maintain their business through accepting/denying requests | | |
| **Normal Flow:** | | 1. Vendor opens their requests tab 2. The system displays current vendor requests 3. Vendor clicks on request 4. Vendor chooses to accept or decline. | | |
| **Alternative Flows:** | | 3a. In step 3 of the normal flow, vendor cannot decide on whether to accept or decline  1. Vendor clicks “save for later” | | |
| **Exceptions:** | | Vendor should have existing requests | | |
| **Includes:** | | None | | |
| **Frequency of Use:** | | On demand | | |
| **Special Requirements:** | | None | | |
| **Assumptions:** | | They can read English.  They can easily access their request tab | | |
| **Notes and Issues:** | | Do we really need this feature? | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID:** | Vendor\_003 | | | |
| **Use Case Name:** | UC\_Vendor\_PRO\_004 | | | |
| **Created By:** | Garion Opiola | |  |  |
| **Date Created:** | 9/16/22 | |  |  |
| **Actors:** | | Vendor | | |
| **Description:** | | so event hirers can have an idea of what vendors are available (?) | | |
| **Trigger:** | | Setting up available products | | |
| **Preconditions:** | | 1. Vendor has account | | |
| **Postconditions:** | | 1. Vendor sets up available products | | |
| **Normal Flow:** | | 1. Vendor goes to profile 2. Vendor types in available products in their profile description 3. Vendor saves changes | | |
| **Alternative Flows:**  **[Alternative Flow 1 – Not in Network]** | | 2a. In step 2 of the normal flow, if the Vendor doesn’t have an account or not logged in   1. System will prompt vendor to create account or login 2. Vendor creates account or logs in 3. Use Case resumes on step 3 | | |
| **Exceptions:** | | 2a.   In step 2 of the Alternative flow, if the Vendor enters an invalid username or password   1. Login is denied 2. Message to vendor to retry login 3. Vendor enters correct login information 4. Use Case resumes on step 3 of normal flow | | |
| **Includes:** | | Search engine | | |
| **Frequency of Use:** | | On demand | | |
| **Special Requirements:** | | Internet connection | | |
| **Assumptions:** | | The vendor understands English  . | | |
| **Notes and Issues:** | | N/A | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID:** | Vendor\_004 | | | |
| **Use Case Name:** | Profile | | | |
| **Created By:** | Gideon Trevor | | **Last Updated By:** |  |
| **Date Created:** |  | | **Last Revision Date:** |  |
| **Actors:** | | Vendor | | |
| **Description:** | | I want to be able to add the fact that I am a venue who may be able to operate for the event so I can access vendor (options/features) for my account | | |
| **Trigger:** | | Select ‘account options’ | | |
| **Preconditions:** | | **1 have an account**  2 is logged in | | |
| **Postconditions:** | | 1 when viewing my account I should be able to see that I am listed as a vendor and have vendor sections on my profile | | |
| **Normal Flow:** | | 2 select ‘Account options’ (maybe in a dropdown with ‘My profile’)  3 there will be a check box to enable that you are a vendor (it will say ‘I am a vendor’ next to a check box  4 at the bottom of this screen when you make a change there should be a place to commit the change(s) | | |
| **Alternative Flows:** | |  | | |
| **Exceptions:** | | 1 you can not edit your account when your account is banned | | |
| **Includes:** | | profile | | |
| **Frequency of Use:** | | People would not be flipping their account often, or at least should not have a reason to | | |
| **Special Requirements:** | | 1 an account setting page that allows for more technical changes to your account | | |
| **Assumptions:** | | 1 that we will have an account setting page that allows for more technical changes to your account | | |
| **Notes and Issues:** | | 1 we will need either a drop down to hold this so we can allow more navigation from the home screen or to have this be accessible when you are viewing your own account | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID:** | Vendor\_006 | | | |
| **Use Case Name:** | UC\_Vendor\_PRO\_006 | | | |
| **Created By:** | Garion Opiola | |  |  |
| **Date Created:** | 9/16/22 | |  |  |
| **Actors:** | | Doner, social secretary/promoter | | |
| **Description:** | | A way to send info that they set up to a possible event to work at | | |
| **Trigger:** | | Doner/Promoter has created account | | |
| **Preconditions:** | | 1. Work history must be empty | | |
| **Postconditions:** | | 1. Doner/promoter work history is made | | |
| **Normal Flow:** | | 1. User goes to site 2. User creates account 3. User fills out work history 4. System creates work history | | |
| **Alternative Flows:**  **[Alternative Flow 1 – Not in Network]** | | 2a. In step 3 of the normal flow, if the user doesn’t have an account   1. System will prompt User to create account 2. User creates account or logs in 3. Use Case resumes on step 3 | | |
| **Exceptions:** | | 2a.   In step 2 of the Alternative flow, if the user enters an invalid username or password   1. Login is denied 2. Message to user to retry login 3. User enters valid login information 4. Use Case resumes on step 4 of normal flow | | |
| **Includes:** | | Search engine | | |
| **Frequency of Use:** | | On demand | | |
| **Special Requirements:** | | Internet connection | | |
| **Assumptions:** | | The user understands English  . | | |
| **Notes and Issues:** | | N/A | | |

### Venue Representative

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID:** | VendorRep\_003 | | | |
| **Use Case Name:** | Arrangement | | | |
| **Created By:** | Gideon Trevor | | **Last Updated By:** |  |
| **Date Created:** |  | | **Last Revision Date:** |  |
| **Actors:** | | Venue representative | | |
| **Description:** | | receive or send association request so I can confirm or create associations with Vendors, or my home team | | |
| **Trigger:** | | 1 select the more “send a associate request” which will | | |
| **Preconditions:** | | **1 has an account**  2 is logged in  3 has indicated that they are a venue representative | | |
| **Postconditions:** | | 1 can find and contact a venue representative | | |
| **Normal Flow:** | | 1 The venue representative will navigate to the Messages tab  2 Then they will select a “more options” button that sits on the chat that they have started with the venue representative or vendor target  3 in the drop down provide, since they are a venue owner and are targeting another venue representative or a vendor they will be able to select “send a associate request”, which if accepted through their notifications tab will make it so they are associated | | |
| **Alternative Flows:**  **Alternant flow 1, no messages already started** | | 2a if the venue representative has not messaged someone that they wish to be associated with  1 go to the search bar and enter any relevant filter or simply browse until they find a vendor or venue representative account  2 select the “more options” button on the vendor or venue representative  3 do step three of the normal flow | | |
| **Exceptions:** | | 1 can not do this while banned  2 can’t perform this on a group message | | |
| **Includes:** | | Arrangement, Direct messaging | | |
| **Frequency of Use:** | | Not frequently used | | |
| **Special Requirements:** | | We’ll need a messages page that way it can all be kept orderly and clean | | |
| **Assumptions:** | | 1 They have an account  2 searching is done with filters  3 that the conversation is saved somewhere | | |
| **Notes and Issues:** | |  | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID:** | VendorRep\_004 | | | |
| **Use Case Name:** | Vendor: Edit Information on Profile | | | |
| **Created By:** | Rith Sivaprakash | | **Last Updated By:** | - |
| **Date Created:** | 9-03-2022 | | **Last Revision Date:** | - |
| **Actors:** | | Vendor | | |
| **Description:** | | Vendors can customize their profile | | |
| **Trigger:** | | Vendor clicks on their profile button | | |
| **Preconditions:** | | 1. Has an account as a vendor | | |
| **Postconditions:** | | Minimal Guarantee:   1. Vendor can customize their profile   Maximum Guarantee:   1. Vendor can successfully advertise their business to teams/tournaments | | |
| **Normal Flow:** | | 1. Vendor opens their profile 2. The vendor selects the “Edit” button 3. Vendor edits their description, venue capacity, location, home team, donors, photos, costs, website, vendors, rating, rules, availability, equipment, venue type 4. Vendor saves the information. | | |
| **Alternative Flows:** | | 3a. In step 3 of the normal flow, vendor enters in invalid information  1. System will display a popup saying information is invalid  2. Flow resumes at step 3 | | |
| **Exceptions:** | | None | | |
| **Includes:** | | None | | |
| **Frequency of Use:** | | On demand | | |
| **Special Requirements:** | | None | | |
| **Assumptions:** | | They can read English.  They can easily access their profile | | |
| **Notes and Issues:** | |  | | |

## Site Administrators

### User

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Use Case ID: | User\_001 | | | |
| Use Case Name: | Create an Account | | | |
| Created By: | Heritier Otiom | | Last Updated By: | Michael Haring |
| Date Created: | 09/18/2022 | | Last Revision Date: | 11/13/2022 |
| Actors: | | User | | |
| Description: | | User creates account in order to access the system | | |
| Trigger: | | User chooses to create an account | | |
| Preconditions: | | 1. All fields must be completed 2. Phone number must be valid 3. Must be a new user with a new email/phone number | | |
| Postconditions: | | 1. Account is created 2. Email confirmation or text sent 3. User has access to app | | |
| Normal Flow: | | 1. Guest clicks on signup button 2. User enters personal information (First name, last name, middle initial, sex, age, favorite sport, phone number etc.) 3. User enters email 4. User enters password 5. User confirms password 6. User agrees to the terms 7. User clicks sign up | | |
| Alternative Flows:  [Alternative Flow 1 – Not in Network] | | 2a. User enters age that is under the age of 18.  User is prompted to enter date with age > 18.  7a. If email is already in the DB, then the system will prompt an error. | | |
| Exceptions: | | 1. Email in use  2. Password doesn’t meet requirements  3. Phone number is invalid  4. Password and Password confirmation don’t match  5. Terms not agreed | | |
| Includes: | | N/A | | |
| Frequency of Use: | | High Frequency. On Demand. | | |
| Special Requirements: | | All fields must be valid, meaning must contain alphanumeric characters.  Password:   * Must be longer than 8 characters * Must not begin with a number * Must contain at least one special character   Numbers must not follow each other in a sequential way (eg. 1234) | | |
| Assumptions: | | User is above 18 years of age. | | |
| Notes and Issues: | | N/A | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Use Case ID: | User\_002 | | | |
| Use Case Name: | Add Profile Picture | | | |
| Created By: | Heritier Otiom | | Last Updated By: | Michael Haring |
| Date Created: | 09/18/2022 | | Last Revision Date: | 11/27/22 |
| Actors: | | User | | |
| Description: | | Picture that is displayed on User profile | | |
| Trigger: | | User clicks on “Update Profile Picture” | | |
| Preconditions: | | 1. Only accepts .JPG, .JPEG, .PNG, .BMP filetypes | | |
| Postconditions: | | User profile picture is set/updated | | |
| Normal Flow: | | 1. User clicks on their profile 2. User clicks on their profile picture 3. User clicks on “Update Profile Picture” button 4. User chooses the image they wish to upload 5. User clicks “Save” 6. User is prompted for confirmation 7. User selects “Yes” to confirm | | |
| Alternative Flows:  [Alternative Flow 1 – Not in Network] | | 4a. When selecting file to upload, the user has two options:  - Select the image from the device  - Use the camera to take a selfie | | |
| Exceptions: | | * The system will prompt an error if the file size is above 5Mb | | |
| Includes: | | * Create Account (User) | | |
| Frequency of Use: | | * High Frequency. On Demand. | | |
| Special Requirements: | | * The size must be under 5Mb | | |
| Assumptions: | | * N/A | | |
| Notes and Issues: | | * N/A | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Use Case ID: | User\_003 | | | |
| Use Case Name: | User Login | | | |
| Created By: | Heritier Otiom | | Last Updated By: | Michael Haring |
| Date Created: | 09/18/2022 | | Last Revision Date: | 11/14/2022 |
| Actors: | | User | | |
| Description: | | Users must login to participate in the events as players, coaches, etc. | | |
| Trigger: | | User interacts with login button | | |
| Preconditions: | | 1. Must have an account 2. Account must not be deactivated | | |
| Postconditions: | | 1. User is logged into their account with full account functionality and any roles they have been given enabled | | |
| Normal Flow: | | 1. User clicks profile image  2. User is taken to login page  3. User enters username and password  4. User is logged into account | | |
| Alternative Flows:  [Alternative Flow 1 – Not in Network] | | 3a. User enters username or password incorrectly.  3b. User is prompted to re-enter credentials until they are correct.  3c. If user does not get credentials right after 5 tries, user is temporarily locked out of account and will need to contact a Moderator for assistance. | | |
| Exceptions: | | The system will prompt an error if the user doesn’t exist or deactivated their account. | | |
| Includes: | | N/A | | |
| Frequency of Use: | | High frequency. On Demand. | | |
| Special Requirements: | | * The user won’t have to login every time to use the App. They will only login if they logged out last time they used the app (e.g. TikTok, Instagram, YouTube, etc.) * Credentials must be saved on the user’s device. | | |
| Assumptions: | | User has a working account | | |
| Notes and Issues: | | N/A | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Use Case ID: | User\_004 | | | |
| Use Case Name: | User Logout | | | |
| Created By: | Heritier Otiom | | Last Updated By: | Michael Haring |
| Date Created: | 09/18/2022 | | Last Revision Date: | 11/14/2022 |
| Actors: | | User | | |
| Description: | | User logs out of their account | | |
| Trigger: | | User clicks logout button under profile picture dropdown options | | |
| Preconditions: | | 1. Must be logged into account | | |
| Postconditions: | | 1. User is safely logged out of their account | | |
| Normal Flow: | | 1. User clicks on profile image 2. Dropdown appears 3. User clicks logout under dropdown options 4. System clear credentials cached in browser and logs user out of account | | |
| Alternative Flows:  [Alternative Flow 1 – Not in Network] | | N/A | | |
| Exceptions: | | N/A | | |
| Includes: | | N/A | | |
| Frequency of Use: | | High Frequency. On Demand. | | |
| Special Requirements: | | Credentials are cleared from browser cache | | |
| Assumptions: | | User has a working account | | |
| Notes and Issues: | | N/A | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Use Case ID: | User\_005 | | | |
| Use Case Name: | Browse Events | | | |
| Created By: | Heritier Otiom | | Last Updated By: | Michael Haring |
| Date Created: | 09/18/2022 | | Last Revision Date: | 10/31/22 |
| Actors: | | User | | |
| Description: | | User is able to browse for active or future events | | |
| Trigger: | | User interacts with events tab | | |
| Preconditions: | | 1. Event must exist 2. Event must be an active or future event | | |
| Postconditions: | | 1. Events user is looking for are shown | | |
| Normal Flow: | | 1. User select Events tab  2. Events tab drops down displaying option to select active or future events  3. User makes selection  4. User is taken to events page of selection | | |
| Alternative Flows:  [Alternative Flow 1 – Not in Network] | | User searches for event using event search.  Actively filters out events while user types. | | |
| Exceptions: | | * Event does not exist | | |
| Includes: | | * N/A | | |
| Frequency of Use: | | * High Frequency. On Demand. | | |
| Special Requirements: | | * Past events can be searched via site search or search engine but will not be tabbed. | | |
| Assumptions: | | * Event exists | | |
| Notes and Issues: | | * N/A | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Use Case ID: | User\_006 | | | |
| Use Case Name: | Create Events | | | |
| Created By: | Heritier Otiom | | Last Updated By: | Michael Haring |
| Date Created: | 09/18/2022 | | Last Revision Date: | 11/13/2022 |
| Actors: | | User | | |
| Description: | | User creates event | | |
| Trigger: | | User selects “Create Event” | | |
| Preconditions: | | 1. Must have a proper name 2. Must have a particular sport field 3. Must have a description 4. Must set the number of participants 5. Must specify the price of a ticket if there is one 6. Must have a location 7. Must have a time | | |
| Postconditions: | | 1. Event is created | | |
| Normal Flow: | | 1. User goes to events  2. User clicks “Create Event”  3. User sets required event information (name, location, time, price, etc.)  4. User clicks create event.  5. User is prompted whether they are sure they wish to create event.  6. User selects “Yes” to create event.  7. Event is created. | | |
| Alternative Flows:  [Alternative Flow 1 – Not in Network] | | 6a. User selects “No” to create event.  6b. User is taken back to previous screen. | | |
| Exceptions: | | * The event doesn’t agree with terms of the site | | |
| Includes: | | * N/A | | |
| Frequency of Use: | | * High Frequency. On Demand. | | |
| Special Requirements: | | * Events must be in accordance with the terms and conditions of site | | |
| Assumptions: | | * Event is in accordance with the terms of the site | | |
| Notes and Issues: | | * The system will prompt an error if the event doesn’t match the requirements | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Use Case ID: | User\_007 | | | |
| Use Case Name: | Join Events | | | |
| Created By: | Heritier Otiom | | Last Updated By: | Michael Haring |
| Date Created: | 09/18/2022 | | Last Revision Date: | 11/13/2022 |
| Actors: | | User | | |
| Description: | | User joins an event | | |
| Trigger: | | User wants to join an event | | |
| Preconditions: | | 1. Event must exist or active | | |
| Postconditions: | | 1. User has joined event | | |
| Normal Flow: | | 1. User browses to active or upcoming events  2. User clicks into event page  3. User clicks “join event” button  4. User is prompted whether they are sure they wish to join event  5. User selects “Yes” to confirm | | |
| Alternative Flows:  [Alternative Flow 1 – Not in Network] | | 1a. User uses search to discover event.  1b. Events are actively filtered as user searches.  1c. Normal Flow continues as normal.  2.  If event requires payment to join, the user must provide payment before joining the event. | | |
| Exceptions: | | * The event name/id is incorrectly typed | | |
| Includes: | | * Browse Events (User) | | |
| Frequency of Use: | | * High Frequency. On Demand. | | |
| Special Requirements: | | * N/A | | |
| Assumptions: | | * Event exists | | |
| Notes and Issues: | | * N/A | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Use Case ID: | User\_008 | | | |
| Use Case Name: | Leave event | | | |
| Created By: | Heritier Otiom | | Last Updated By: | Michael Haring |
| Date Created: | 09/18/2022 | | Last Revision Date: | 11/14/2022 |
| Actors: | | User | | |
| Description: | | User can leave event they are no longer interested in. | | |
| Trigger: | | User interacts with “leave” button on the event page. | | |
| Preconditions: | | 1. User is signed up for the event. | | |
| Postconditions: | | 1. User is returned to their events screen and the event they have dropped is no longer listed. | | |
| Normal Flow: | | 1. User navigates to their events page and selects the event they would like to drop. 2. User clicks the “leave” button which brings up a prompt if they are sure they would like to drop out of the event. 3. User clicks “yes” and is returned to their events page with the event no longer listed. | | |
| Alternative Flows:  [Alternative Flow 1 – Not in Network] | | 3a. On step 3 of the normal flow, User clicks cancel on the leave event prompt.   1. User clicks leave on the event. 2. User clicks cancel instead of yes on the prompt. 3. User is returned to the event details page and the event is still listed. | | |
| Exceptions: | | * N/A | | |
| Includes: | | * This includes the use case where the reports are able to view their events page. | | |
| Frequency of Use: | | * Used Frequently. On Demand. | | |
| Special Requirements: | | * Required to be enrolled in an event already. | | |
| Assumptions: | | * Assumes they are already signed up for an event they would like to drop. | | |
| Notes and Issues: | | * N/A | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Use Case ID: | User\_012 | | | |
| Use Case Name: | Report an Issue | | | |
| Created By: | Heritier Otiom | | Last Updated By: | Michael Haring |
| Date Created: | 09/18/2022 | | Last Revision Date: | 11/27/2022 |
| Actors: | | User | | |
| Description: | | User would like to report an issue | | |
| Trigger: | | User clicks on “Report Issue” | | |
| Preconditions: | | 1. The report fields must not be blank | | |
| Postconditions: | | 1. Report is sent | | |
| Normal Flow: | | 1. User goes to the contact page 2. User clicks on “Report Issue” 3. User writes their report 4. User clicks on “Submit” | | |
| Alternative Flows:  [Alternative Flow 1 – Not in Network] | | * N/A | | |
| Exceptions: | | * N/A | | |
| Includes: | | * N/A | | |
| Frequency of Use: | | * Used Frequently. On Demand. | | |
| Special Requirements: | | * The user can report everything they would like (Other users, events, post, etc.) | | |
| Assumptions: | | * User has an issue they would like to report | | |
| Notes and Issues: | | * All reports will go to the site administrator to review | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Use Case ID: | User\_015 | | | |
| Use Case Name: | Suggest a Feature | | | |
| Created By: | Heritier Otiom | | Last Updated By: | Michael Haring |
| Date Created: | 09/18/2022 | | Last Revision Date: | 11/27/2022 |
| Actors: | | User | | |
| Description: | | User would like to suggest a feature for use on site | | |
| Trigger: | | User clicks on “Suggest a Feature” | | |
| Preconditions: | | 1. The suggest fields must not be blank | | |
| Postconditions: | | 1. Feature suggested by user is submitted | | |
| Normal Flow: | | 1. User goes to the contact page 2. User clicks on “Suggest a Feature” 3. User writes their comments 4. User clicks on “Submit” | | |
| Alternative Flows:  [Alternative Flow 1 – Not in Network] | | * N/A | | |
| Exceptions: | | * N/A | | |
| Includes: | | * N/A | | |
| Frequency of Use: | | * Infrequent Use. On Demand. | | |
| Special Requirements: | | * N/A | | |
| Assumptions: | | * User has a feature that they would like to suggest | | |
| Notes and Issues: | | * All suggestions are sent to the site administrator to review | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Use Case ID: | User\_016 | | | |
| Use Case Name: | Reset Password | | | |
| Created By: | Heritier Otiom | | Last Updated By: | Michael Haring |
| Date Created: | 09/18/2022 | | Last Revision Date: | 11/27/2022 |
| Actors: | | User | | |
| Description: | | User would like to reset their password | | |
| Trigger: | | User clicks “Forgot Password” | | |
| Preconditions: | | 1. User must have an active account | | |
| Postconditions: | | 1. Password reset email is sent | | |
| Normal Flow: | | 1. User goes to login page 2. User clicks “Forgot Password” 3. User enters their email when prompted 4. User receives email from app to reset password 5. User clicks the link on the email 6. User creates new password | | |
| Alternative Flows:  [Alternative Flow 1 – Not in Network] | | * N/A | | |
| Exceptions: | | * N/A | | |
| Includes: | | * N/A | | |
| Frequency of Use: | | * High Frequency. On Demand. | | |
| Special Requirements: | | * The email address must already be registered with site in order for reset to work. | | |
| Assumptions: | | * User forgot or would like to change their password. | | |
| Notes and Issues: | | * User can reset their password anytime they would like, or if their account is locked. * Application will log User off from all devices they are currently logged into. | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Use Case ID: | User\_017 | | | |
| Use Case Name: | Deactivate Account | | | |
| Created By: | Heritier Otiom | | Last Updated By: | Michael Haring |
| Date Created: | 09/18/2022 | | Last Revision Date: | 11/27/2022 |
| Actors: | | User | | |
| Description: | | User would like to deactivate their account | | |
| Trigger: | | User clicks “Deactivate My Account” | | |
| Preconditions: | | 1. User must have an active account | | |
| Postconditions: | | 1. User account is deactivated | | |
| Normal Flow: | | 1. User goes to their profile 2. User clicks “Deactivate My Account” | | |
| Alternative Flows:  [Alternative Flow 1 – Not in Network] | | * N/A | | |
| Exceptions: | | * N/A | | |
| Includes: | | * N/A | | |
| Frequency of Use: | | * High Frequency. On Demand. | | |
| Special Requirements: | | * System will prompt a verification question for confirmation that user would like to deactivate their account. * Confirmation email will also be sent once account is deactivated. | | |
| Assumptions: | | * User has an active account. * User would like to deactivate their account. | | |
| Notes and Issues: | | * User account will not be deleted from system. It will just be suspended and hidden until the user reactivates it. | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID:** | User\_018 | | | |
| **Use Case Name:** | UC\_User\_SRCH\_020 | | | |
| **Created By:** | Rith | | **Last Updated By:** | Rith |
| **Date Created:** | 9/16/22 | | **Last Revision Date:** | 11/7/2022 |
| **Actors:** | | User | | |
| **Description:** | | Search for events | | |
| **Trigger:** | | User enters the search tab | | |
| **Preconditions:** | | 1. User has an account | | |
| **Postconditions:** | | 1. User views search results | | |
| **Normal Flow:** | | 1. User logs in 2. User clicks on “search” tab 3. User enters information into the search bar 4. User clicks enter 5. Search results are displayed | | |
| **Alternative Flows:**  **[Alternative Flow 1 – Not in Network]** | | None | | |
| **Exceptions:** | | 1. 5a. In step 5 of the normal flow, if no events are related to the searched terms 2. System display the message “No events found.” | | |
| **Includes:** | | Search function | | |
| **Frequency of Use:** | | On demand | | |
| **Special Requirements:** | | None | | |
| **Assumptions:** | | The user understands English.  The user has something they want to search by. | | |
| **Notes and Issues:** | | N/A | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID:** | User\_021 | | | |
|  |  | | | |
| **Use Case Name:** | Search for potential teams | | | |
| **Created By:** | Izy Doughty | | **Last Updated By:** | Rith |
| **Date Created:** | 9-25-2022 | | **Last Revision Date:** | 11-7-2022 |
| **Actors:** | | User | | |
| **Description:** | | Users search for teams based on specific criteria to donate to. Criteria includes: Team name, player name, location, sport | | |
| **Trigger:** | | User goes to the search section on the home page and uses the search to do a basic, or advanced search to find a team.  A basic search would include one filter, an advanced search would include more than 1 filter. | | |
| **Preconditions:** | | 1. Search needs to use filters 2. The user needs to consider who they are looking for | | |
| **Postconditions:** | | Minimal Guarantee   1. User will be able to see a list of possible teams 2. User will be able to see a message informing them that the criteria they have chosen does not match anything list and that they should try again   Max Guarantee   1. User will find their team based on criteria 2. User will be able to donate to a team based on what they see on their profile | | |
| **Normal Flow:** | | 1. Existing User logs in 2. Search for a team based on chosen single criteria (basic search) using a dropdown. 3. Press "Submit" 4. See a list of teams 5. Select a team to go to their profile | | |
| **Alternative Flows:**  **[Alternative Flow 1 – Not in Network]** | | 2b. Search for a team based on chosen criteria (advanced search) using dropdowns. If the dropdown is left blank, it is not used in the search.   1. Press "Submit" 2. See a list of teams 3. Select a team to go to their profile | | |
| **Exceptions:** | | 1. If the criteria chosen does not match to a team, a message needs to show on the results section. 2. If the user does not use all the drop downs in the drop-down section (leaves them blank), they are ignored for the search. | | |
| **Includes:** | |  | | |
| **Frequency of Use:** | | On demand | | |
| **Special Requirements:** | | User must be logged in | | |
| **Assumptions:** | | * User will know more about a team aside from their name * There will be more than one team with the same name | | |
| **Notes and Issues:** | |  | | |

### Administrator

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Use Case ID: | Admin\_003 | | | |
| Use Case Name: | Ban User | | | |
| Created By: | Heritier Otiom | | Last Updated By: | Jacob Lindauer |
| Date Created: | 09/18/2022 | | Last Revision Date: | 11/07/2022 |
| Actors: | | Admin  User | | |
| Description: | | Ban users from the site | | |
| Trigger: | | Admin would like to ban a user from the site. | | |
| Preconditions: | | 1. User is reported by another user on the site. | | |
| Postconditions: | | 1. Selected userID is banned from website | | |
| Normal Flow: | | 1. Admin views submitted user report. 2. Admin determines if a suspected user should be banned from the site. 3. Admin uses options to select how long the user will be banned from the site and fills in the optional box to explain why they are banned. 4. Once completed, the admin clicks the “Ban User” option and the ban is submitted. | | |
| Alternative Flows:  [Alternative Flow 1 – Ban user from search] | | 1a. In step 1 of the normal flow. The admin searches for a user and bans them manually.   1. Admin searches for specific user they would like to ban. 2. Admin selects a user and is taken to their profile page. 3. Admin clicks the edit button on their profile and selects the ban user option. 4. Admin is prompted with a window to select amount | | |
| Exceptions: | | (None) | | |
| Includes: | | (None) | | |
| Frequency of Use: | | On demand | | |
| Special Requirements: | | (None) | | |
| Assumptions: | | * Assumes user is someone that is not banned already. | | |
| Notes and Issues: | | (None) | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID:** | Admin\_005 | | | |
| **Use Case Name:** | Grant Moderator Privileges | | | |
| **Created By:** | Michael Haring | | **Last Updated By:** | Jacob Lindauer |
| **Date Created:** | 9/19/2022 | | **Last Revision Date:** | 11/07/2022 |
| **Actors:** | | Administrator  Moderator  User | | |
| **Description:** | | Allows an administrator to assign users moderator privileges. | | |
| **Trigger:** | | Administrator interacts with user details. | | |
| **Preconditions:** | | 1. Admin would like to elevate someone's access to the moderator role. | | |
| **Postconditions:** | | 1. Intended user is granted moderator role access features. | | |
| **Normal Flow:** | | 1. Administrator clicks on user details. 2. Administrator clicks edit button. 3. Administrator changes user roles and adds moderator role. 4. Popup asks the administrator to confirm. 5. Administrator chooses “yes”. 6. User gains moderator privileges. | | |
| **Alternative Flows:**  **[Alternative Flow 1 – Not in Network]** | | 5b. Administrators choosing the “No” options when confirming change.   1. Administrator chooses “no”. 2. Administrator is returned to user details page. | | |
| **Exceptions:** | | 1. User already has moderator role. | | |
| **Includes:** | | N/A | | |
| **Frequency of Use:** | | On demand. Expected to be used with light/moderate use. | | |
| **Special Requirements:** | | N/A | | |
| **Assumptions:** | | User must have an active account.  User is not “deleted” or hidden. | | |
| **Notes and Issues:** | | N/A | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID:** | Admin\_006 | | | |
| **Use Case Name:** | Revoke Moderator Privileges | | | |
| **Created By:** | Michael Haring | | **Last Updated By:** | Jacob Lindauer |
| **Date Created:** | 10/03/2022 | | **Last Revision Date:** | 11/07/2022 |
| **Actors:** | | Administrator  Moderator  User | | |
| **Description:** | | Allows an administrator to revoke user’s moderator privileges. | | |
| **Trigger:** | | Administrator interacts with user details. | | |
| **Preconditions:** | | 1. Admin would like to remove the moderator access for another user. | | |
| **Postconditions:** | | 1. Intended user’s moderator privileges are revoked. | | |
| **Normal Flow:** | | 1. Administrator clicks on user details. 2. Administrator clicks edit button. 3. Administrator changes user roles and revokes moderator role. 4. Popup asks the administrator to confirm. 5. Administrator chooses “yes”. 6. Users moderator privileges are removed. | | |
| **Alternative Flows:**  **[Alternative Flow 1 – Not in Network]** | | 5b. Admin select the “No” option at the confirmation prompt.   1. Administrator chooses “no”. 2. Administrator is returned to user details page. | | |
| **Exceptions:** | | 1. User is not given the moderator role. | | |
| **Includes:** | | N/A | | |
| **Frequency of Use:** | | On demand. Expected to be used with light/moderate use. | | |
| **Special Requirements:** | | N/A | | |
| **Assumptions:** | | User must have an active account.  User is not “deleted” or hidden. | | |
| **Notes and Issues:** | | N/A | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID:** | Admin\_007 | | | |
| **Use Case Name:** | Delete Events | | | |
| **Created By:** | Michael Haring | | **Last Updated By:** | Michael Haring |
| **Date Created:** | 10/03/2022 | | **Last Revision Date:** | 11/27/2022 |
| **Actors:** | | Administrator | | |
| **Description:** | | Allows administrators to delete events. | | |
| **Trigger:** | | Administrator interacts with event details. | | |
| **Preconditions:** | | 1. Administrator would like to remove an event from the system. | | |
| **Postconditions:** | | 1. Event is “deleted” from the site and no longer visible to users. | | |
| **Normal Flow:** | | 1. Administrator clicks on event details. 2. Administrator clicks edit button. 3. Administrator clicks the “Delete Event” option. 4. Popup asks the administrator to confirm. 5. Administrator chooses “Yes”. 6. Event is removed from the site and is no longer visible. | | |
| **Alternative Flows:**  **[Alternative Flow 1 – Not in Network]** | | 5b. Administrator selects “No” on the confirmation prompt.   1. Administrator chooses “No”. 2. Administrator is returned to the event details page. | | |
| **Exceptions:** | | N/A | | |
| **Includes:** | | N/A | | |
| **Frequency of Use:** | | Moderate Frequency. On Demand. | | |
| **Special Requirements:** | | N/A | | |
| **Assumptions:** | | Event exists. | | |
| **Notes and Issues:** | | N/A | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID:** | Admin\_008 | | | |
| **Use Case Name:** | Delete Users | | | |
| **Created By:** | Michael Haring | | **Last Updated By:** | Michael Haring |
| **Date Created:** | 10/03/2022 | | **Last Revision Date:** | 11/27/2022 |
| **Actors:** | | Administrator | | |
| **Description:** | | Allows an administrator to “Delete” users. | | |
| **Trigger:** | | Administrator interacts with user details. | | |
| **Preconditions:** | | 1. Administrator has a user they would like to remove from the system. | | |
| **Postconditions:** | | 1. User is “removed” from the site. 2. User can no longer sign into the site. | | |
| **Normal Flow:** | | 1. Administrator clicks on user details. 2. Administrator clicks edit button. 3. Administrator clicks “Delete” 4. Popup asks the administrator to confirm. 5. Administrator selects “Yes”. 6. User is “Deleted” from the site. | | |
| **Alternative Flows:**  **[Alternative Flow 1 – Not in Network]** | | 5b. Admin selects “No” on the confirmation screen.   1. Administrator chooses “no”. 2. Administrator is returned to the user details page. | | |
| **Exceptions:** | | (None) | | |
| **Includes:** | | (None) | | |
| **Frequency of Use:** | | On demand. Expected to be used with moderate use. | | |
| **Special Requirements:** | | (None) | | |
| **Assumptions:** | | User is not removed or set inactive already. | | |
| **Notes and Issues:** | | User would not be removed from the database and would instead be set as inactive. | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID:** | Admin\_010 | | | |
| **Use Case Name:** | Unlock Account | | | |
| **Created By:** | Michael Haring | | **Last Updated By:** | Michael Haring |
| **Date Created:** | 10/03/2022 | | **Last Revision Date:** | 11/27/2022 |
| **Actors:** | | Administrator | | |
| **Description:** | | Allows an administrator to send password reset to user email that is locked out of their account. | | |
| **Trigger:** | | Administrator clicks “Reset” | | |
| **Preconditions:** | | 1. User has forgotten their password for the system. | | |
| **Postconditions:** | | 1. User password is set to temp password and is set to change on next login. | | |
| **Normal Flow:** | | 1. Administrator clicks on user details. 2. Administrator clicks edit button. 3. Administrator clicks “Reset”. 4. Email is sent to email address registered to user to choose a new password in order to gain access to their account. | | |
| **Alternative Flows:**  **[Alternative Flow 1 – Not in Network]** | | 3a. Admin closes out of the password reset screen.   1. Admin is prompted with a window asking if they are sure they would like to cancel password change. 2. Clicking no returns admin to the user profile page. | | |
| **Exceptions:** | | * N/A | | |
| **Includes:** | | * N/A | | |
| **Frequency of Use:** | | * Moderate Use. On Demand. | | |
| **Special Requirements:** | | * N/A | | |
| **Assumptions:** | | * User has an active account. | | |
| **Notes and Issues:** | | * N/A | | |

### Moderator

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID:** | Moderator\_001 | | | |
| **Use Case Name:** | Update Roles | | | |
| **Created By:** | Michael Haring | | **Last Updated By:** | Michael Haring |
| **Date Created:** | 10/03/2022 | | **Last Revision Date:** | 11/27/2022 |
| **Actors:** | | Moderator  User | | |
| **Description:** | | Allows a moderator to update and change user roles. | | |
| **Trigger:** | | Moderator would like to add roles to a specific user | | |
| **Preconditions:** | | 1. User must be an active user in the system. | | |
| **Postconditions:** | | 1. User account roles are added and any privileges with the role are unlocked. | | |
| **Normal Flow:** | | 1. Moderator clicks on user details. 2. Moderator clicks the edit button. 3. Moderator selects the team or league on player profile and role dynamically changes with selection. 4. Moderator clicks on “Role” button. 5. Roles dropdown and moderator is able to select correct role from list. | | |
| **Alternative Flows:**  **[Alternative Flow 1 – Not in Network]** | | 5b. Moderator selects “No” on the confirmation screen.   1. Moderator chooses “no”. 2. Moderator is returned to the user details page. | | |
| **Exceptions:** | | 7a. User already has the selected role.   1. Moderator is prompted with a screen saying what selected roles the user already has. 2. Normal flow continues and changes are applied. | | |
| **Includes:** | | * N/A | | |
| **Frequency of Use:** | | * Moderate Use. On Demand. | | |
| **Special Requirements:** | | * N/A | | |
| **Assumptions:** | | * User has an active account. | | |
| **Notes and Issues:** | | * N/A | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID:** | Moderator\_002 | | | |
| **Use Case Name:** | Add Users to Team | | | |
| **Created By:** | Michael Haring | | **Last Updated By:** | Jacob Lindauer |
| **Date Created:** | 10/03/2022 | | **Last Revision Date:** | 11/07/2022 |
| **Actors:** | | Moderator  User | | |
| **Description:** | | Allows moderator to update User’s team information. | | |
| **Trigger:** | | Moderator interacts with team details. | | |
| **Preconditions:** | | 1. User must be active in the system.  2. Team must be active in the system.  3. Moderator would like to add user to selected team. | | |
| **Postconditions:** | | 1. User account team information is updated and user is added to the team. Team privileges are unlocked for user. | | |
| **Normal Flow:** | | 1. Moderator searches for a team and selects the team details. 2. Moderator selects the add user button. 3. Moderator selects the desired user they would like to add. 4. Moderator selects the “Add” button. 5. Moderator is prompted with a confirmation screen, moderator select yes. 6. User is added to the team. | | |
| **Alternative Flows:**  **[Alternative Flow 1 – Moderator Select No]** | | 5a. Moderator select No on the confirmation prompt.   1. Moderator chooses “no”. 2. Moderator is returned to user details > teams page. | | |
| **Exceptions:** | | User is already a part of the team. | | |
| **Includes:** | | Coach\_005 where a coach can add a user but a moderator can do it manually, bypassing the invite system. | | |
| **Frequency of Use:** | | On demand. Expected to be used with moderate to heavy use. | | |
| **Special Requirements:** | | N/A | | |
| **Assumptions:** | | User does not exist on the team roster already and User can be added to selected team. | | |
| **Notes and Issues:** | | N/A | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID:** | Moderator\_003 | | | |
| **Use Case Name:** | Remove Users from Team | | | |
| **Created By:** | Michael Haring | | **Last Updated By:** | Jacob Lindauer |
| **Date Created:** | 10/03/2022 | | **Last Revision Date:** | 11/13/2022 |
| **Actors:** | | Moderator | | |
| **Description:** | | Moderator removing a user from a team. | | |
| **Trigger:** | | Moderator would like to remove a user from a team. | | |
| **Preconditions:** | | 1. User must be on the selected team. | | |
| **Postconditions:** | | 1. User account team information is updated and the user is removed from team(s). Team privileges are removed for the user. | | |
| **Normal Flow:** | | 1. Moderator searched for the team they want to remove a user from. 2. Moderator clicks on team details. 3. Moderator clicks the edit team button. 4. Moderator clicks the option to remove a user from the team. 5. Moderator is prompted if they are sure they would like to remove the selected user. 6. Moderator clicks yes to remove the requested user. 7. Users are removed from team roster and team events. | | |
| **Alternative Flows:**  **[Alternative Flow 1 – Moderator selects no at the removal prompt]** | | 6a. Moderator selects No at the prompt.   1. Moderator chooses “no”. 2. Moderator is returned to user details > teams page and user is not removed. | | |
| **Exceptions:** | | (None) | | |
| **Includes:** | | Includes Coach\_007 where a coach removes a player from the team. | | |
| **Frequency of Use:** | | On demand. Expected to be used with moderate to heavy use. | | |
| **Special Requirements:** | | N/A | | |
| **Assumptions:** | | - User is a member of the selected team | | |
| **Notes and Issues:** | | N/A | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID:** | Moderator\_004 | | | |
| **Use Case Name:** | Ban Users | | | |
| **Created By:** | Michael Haring | | **Last Updated By:** | Jacob Lindauer |
| **Date Created:** | 10/03/2022 | | **Last Revision Date:** | 11/13/2022 |
| **Actors:** | | Moderator | | |
| **Description:** | | Allows moderator to ban or “remove” users from site pages. | | |
| **Trigger:** | | Moderator would like to ban a user from the system | | |
| **Preconditions:** | | User must exist in system.  User must not already be removed, banned or “deleted”. | | |
| **Postconditions:** | | 1. User account is “removed” from site access. | | |
| **Normal Flow:** | | 1. Moderator clicks on user details. 2. Moderator clicks the edit button. 3. Moderator clicks option to ban the user. 4. Moderator is prompted with a confirmation screen. 5. Moderator selected Yes to ban the user from the system. 6. Moderator confirms action and user account is “removed” from site. | | |
| **Alternative Flows:**  **[Alternative Flow 1 – Not in Network]** | | 5a. Moderator selects no on the confirmation prompt.   1. Moderator chooses “no”. 2. Moderator is returned to user details screen and user is not removed. | | |
| **Exceptions:** | | (None) | | |
| **Includes:** | | Same as Admin\_003 where an admin is able to ban users | | |
| **Frequency of Use:** | | On demand. Expected to be used with moderate to heavy use. | | |
| **Special Requirements:** | | N/A | | |
| **Assumptions:** | | User must be marked active already. | | |
| **Notes and Issues:** | | N/A | | |

## Others

### Score Reporter

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID:** | Score\_001 | | | |
| **Use Case Name:** | Add Score to Event | | | |
| **Created By:** | Alex Korte | | **Last Updated By:** | Michael Haring |
| **Date Created:** | 9/26/2022 | | **Last Revision Date:** | 11/27/2022 |
| **Actors:** | | Score Reporter  League Admin  Tournament Admin | | |
| **Description:** | | User is able to add scores or standings. | | |
| **Trigger:** | | User clicks “add score” button under events score options. | | |
| **Preconditions:** | | 1. User must have the appropriate role (Score Reporter/Tournament Admin/League Admin). | | |
| **Postconditions:** | | 1. Score will reflect on the tournament/match/league | | |
| **Normal Flow:** | | 1. User navigates to the scores they would like to add under event page 2. User clicks “add score” button 3. User enters score 4. User clicks submit 5. User is shown score preview and prompted to confirm update 6. User selects “yes” to confirm 7. User is taken back to event page with score now added for event | | |
| **Alternative Flows:**  **[Alternative Flow 1 – Not in Network]** | | 4a. If the information is not correct (data types) System will throw an error  4b. User is taken back to update the information to be correct  4c. User can cancel update  6a. User selects “no” on confirmation  6b. User is taken back to update the information to be correct  6c. User can cancel update | | |
| **Exceptions:** | | 1. If the user does not enter correct data types, the system will prompt the user for correct data. | | |
| **Includes:** | | * N/A | | |
| **Frequency of Use:** | | * High Frequency. On Demand. | | |
| **Special Requirements:** | | * N/A | | |
| **Assumptions:** | | * User will be given role through Tournament Admin, League Admin. | | |
| **Notes and Issues:** | | * N/A | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID:** | Score\_002 | | | |
| **Use Case Name:** | Update Score | | | |
| **Created By:** | Alex Korte | | **Last Updated By:** | Michael Haring |
| **Date Created:** | 9/26/2022 | | **Last Revision Date:** | 11/13/2022 |
| **Actors:** | | Score Reporter  League Admin  Tournament Admin | | |
| **Description:** | | User is able to modify (update) scores or standings. | | |
| **Trigger:** | | User clicks “update” button under events score options. | | |
| **Preconditions:** | | 1. User must have the appropriate role (Score Reporter/Tournament Admin/League Admin). 2. Event must have score already submitted in the system. | | |
| **Postconditions:** | | 1. Event will display updated score. | | |
| **Normal Flow:** | | 1. User navigates to the scores they would like to modify 2. User clicks “update” button 3. User modifies score with correct information 4. User clicks submit 5. User is shown updated score preview and prompted to confirm update 6. User selects “yes” to confirm 7. User is taken back to event page with updated event score | | |
| **Alternative Flows:**  **[Alternative Flow 1 – Not in Network]** | | 4a. If the information is not correct (data types) System will throw an error  4b. User is taken back to update the information to be correct  4c. User can cancel update  6a. User selects “no” on confirmation  6b. User is taken back to update the information to be correct  6c. User can cancel update | | |
| **Exceptions:** | | 1. If the user does not enter correct data types, the system will prompt the user for correct data. | | |
| **Includes:** | | * N/A | | |
| **Frequency of Use:** | | * Moderate use. On Demand. | | |
| **Special Requirements:** | | * N/A | | |
| **Assumptions:** | | * User will be given role through Tournament Admin, League Admin. | | |
| **Notes and Issues:** | | * N/A | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID:** | Score\_003 | | | |
| **Use Case Name:** | Request Score Reporter | | | |
| **Created By:** | Alex Korte | | **Last Updated By:** | Michael Haring |
| **Date Created:** | 10/3/2022 | | **Last Revision Date:** | 11/28/2022 |
| **Actors:** | | Score Reporter  Tournament Admin  League Admin  User | | |
| **Description:** | | User can accept the role of score reporter as assigned by tournament or league admins. | | |
| **Trigger:** | | User clicks “Accept” on notification that they have been invited to be score reporter for a tournament or league. | | |
| **Preconditions:** | | 1. User does not already have Score Reporter role for event. 2. User is a part of event they are being assigned role of Score Reporter. | | |
| **Postconditions:** | | 1. User is assigned role of Score Reporter for event. | | |
| **Normal Flow:** | | 1. User receives invite from tournament or league admin in notifications panel 2. User clicks on notification 3. Notification pops out for user to read and further interact with 4. User clicks “Accept” on invite 5. Notification is sent back to User who sent invite that request was accepted 6. User is assigned role of Score Reporter | | |
| **Alternative Flows:**  **[Alternative Flow 1 – Not in Network]** | | 4a. User clicks “Deny” on invite.  Invite is denied and user is not assigned role of Score Reporter for event.  Notification is sent back to user who sent invite that request was denied. | | |
| **Exceptions:** | | System will notify the user if their request could not be sent due to user already being assigned role of Score Reporter for event. | | |
| **Includes:** | | N/A | | |
| **Frequency of Use:** | | Infrequent Use. On Demand. | | |
| **Special Requirements:** | | N/A | | |
| **Assumptions:** | | User is not already assigned role of score reporter for event. | | |
| **Notes and Issues:** | |  | | |

### Volunteers

# User Interfaces

## Participants

### Player

### Potential Player

### Spectator

### Commentator

### Videographer

## Organizers

### League Admin

### Tournament Admin

### Coach

### Treasurer

## Logistics

### Equipment Manager

### Donor

### Vendor

### Venue Representative

## Site Administrators

### User

### Administrator

### Moderator

## Others

### Score Reporter

### Volunteers

# Use Case Diagrams

## Participants

### Player

### Potential Player

### Spectator

### Commentator

### Videographer

## Organizers

### League Admin

### Tournament Admin

### Coach

### Treasurer

## Logistics

### Equipment Manager

### Donor

### Vendor

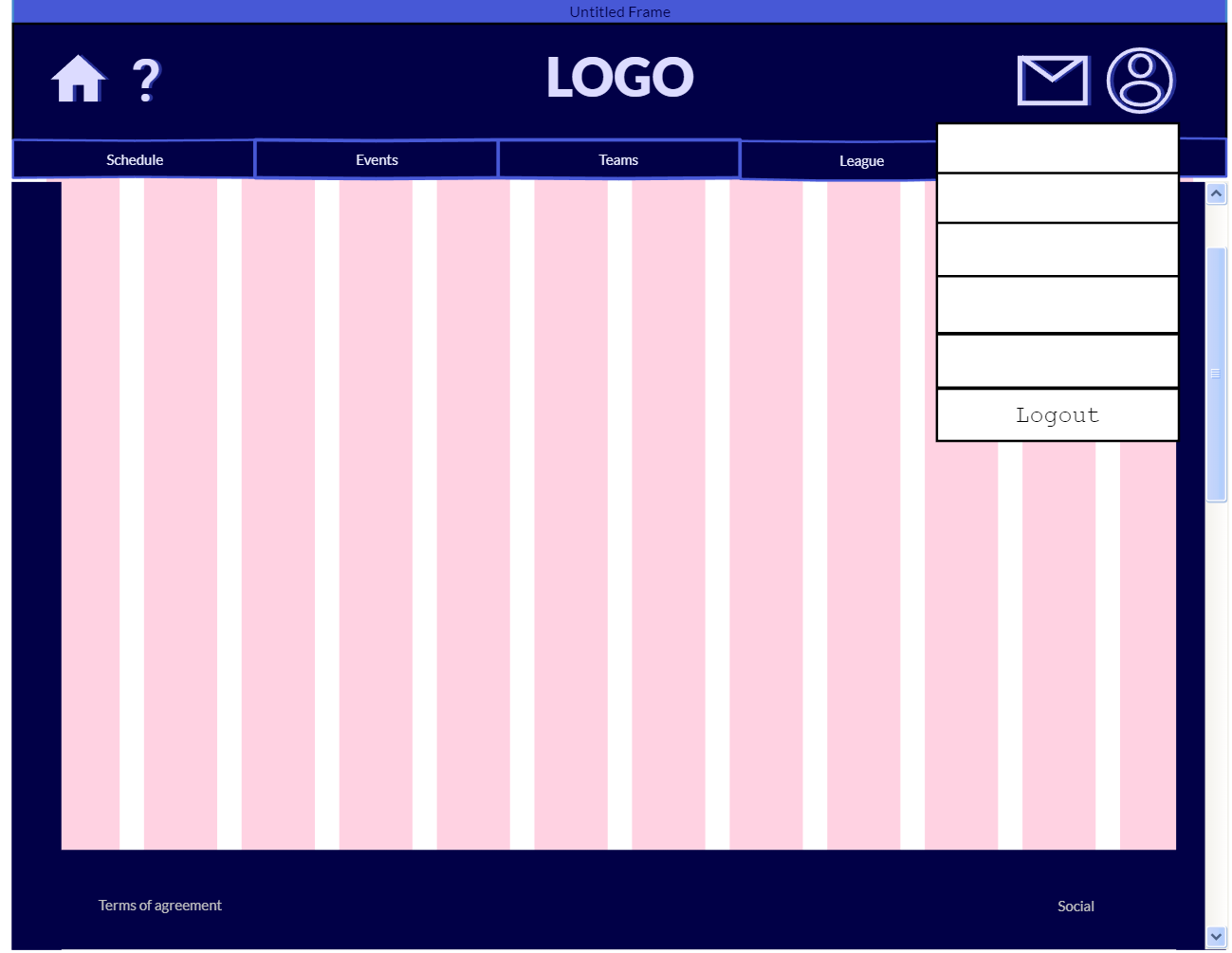
### Venue Representative

## Site Administrators

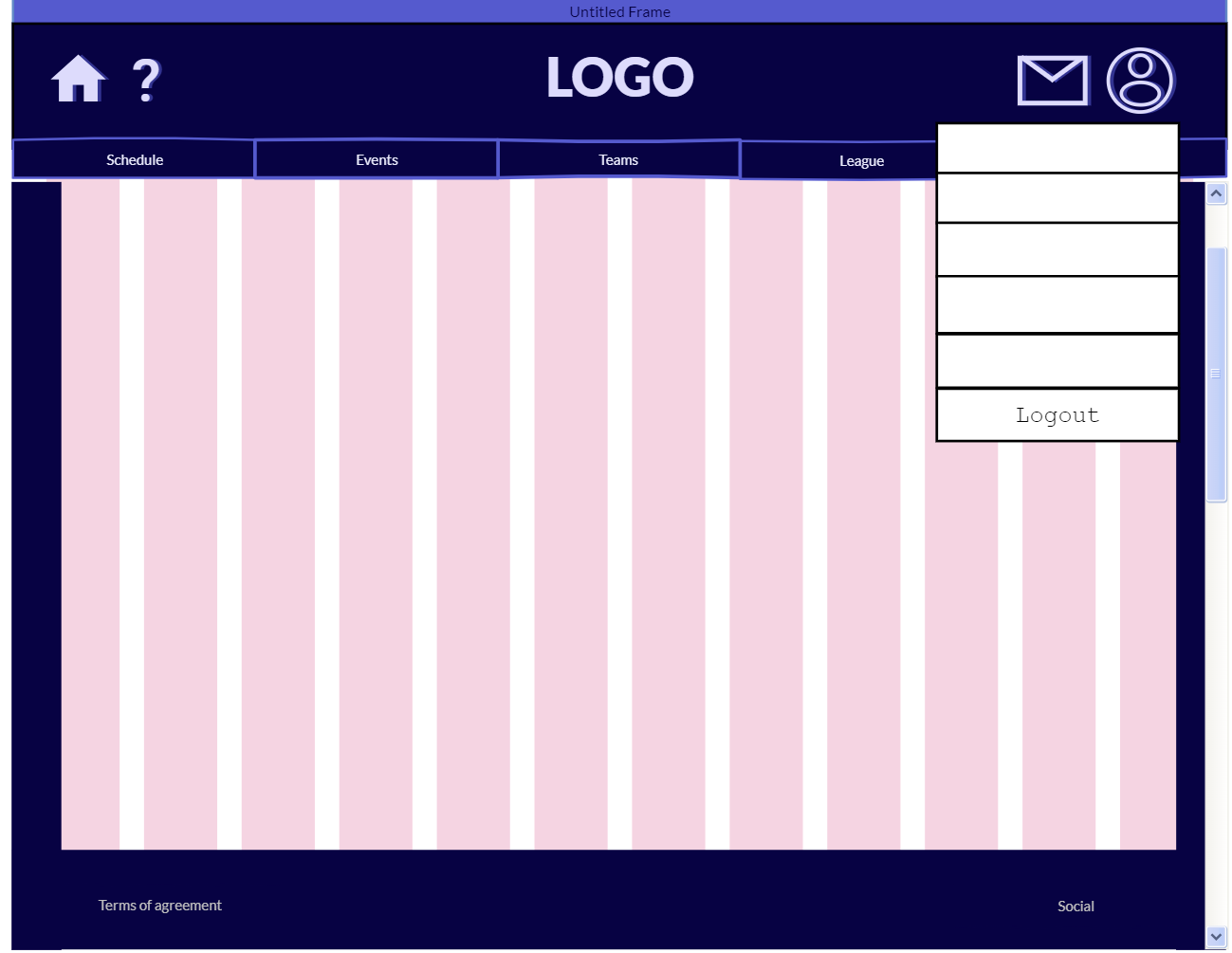
### User

### Administrator

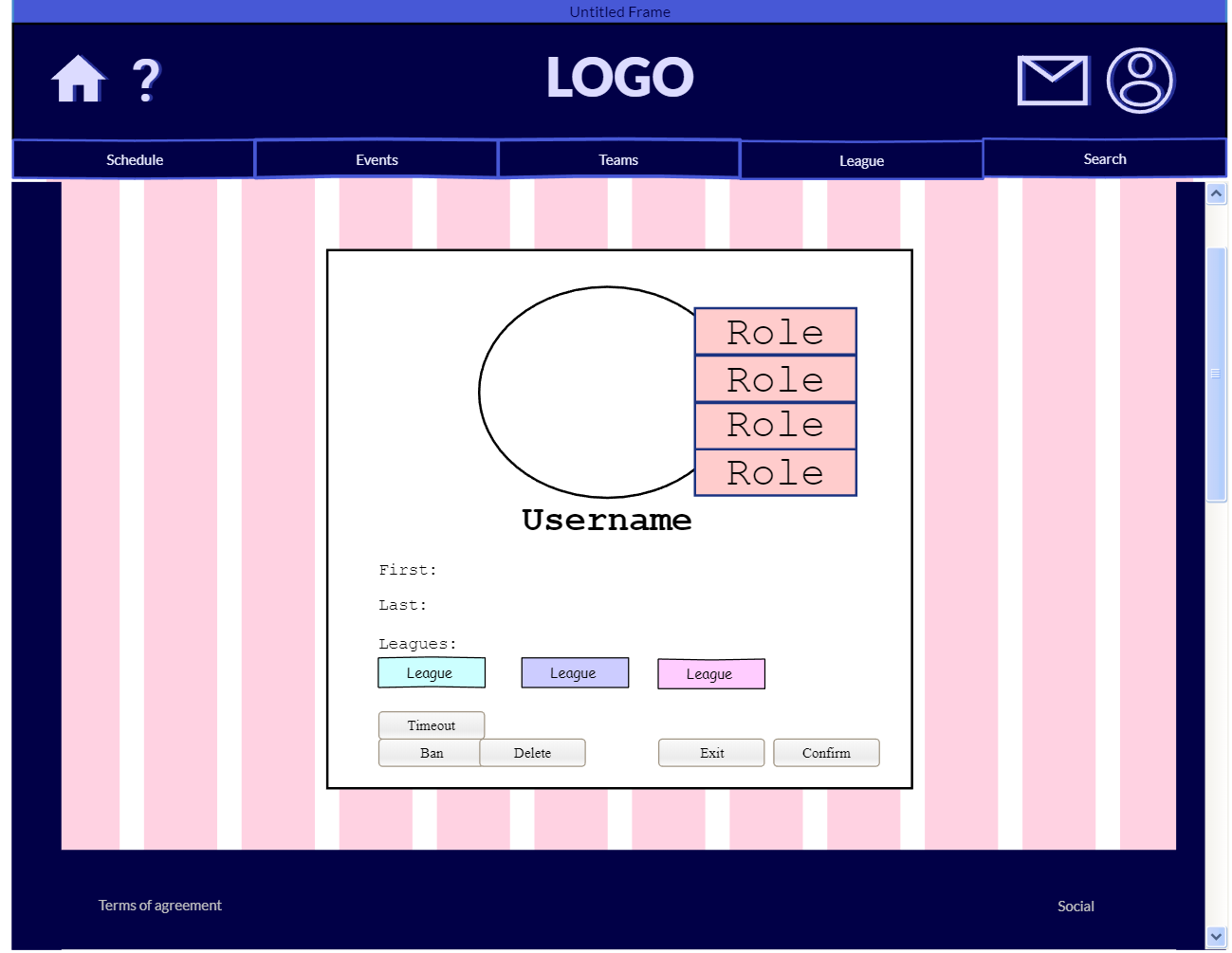
Admin\_002\_Desktop



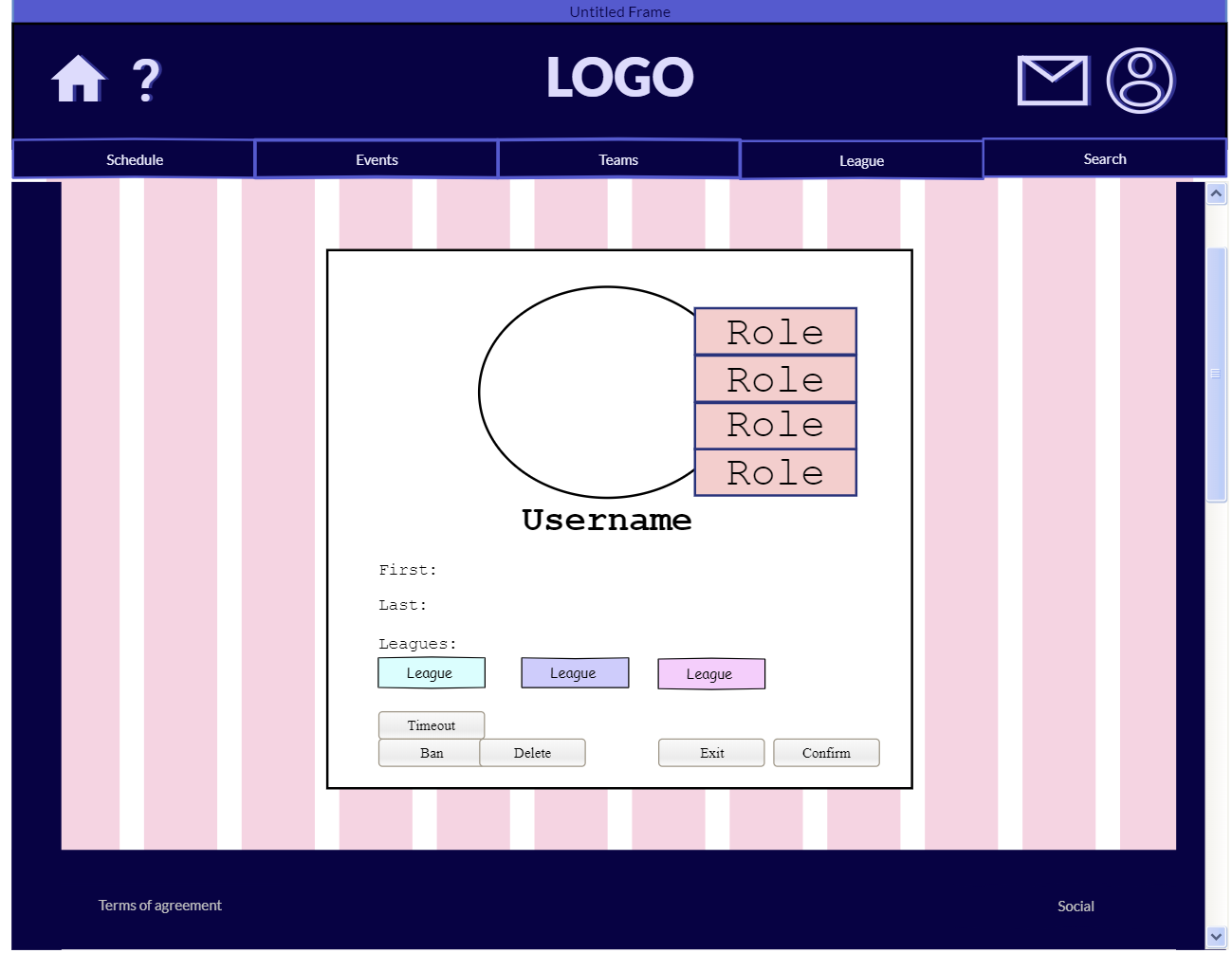
Admin\_002\_Web



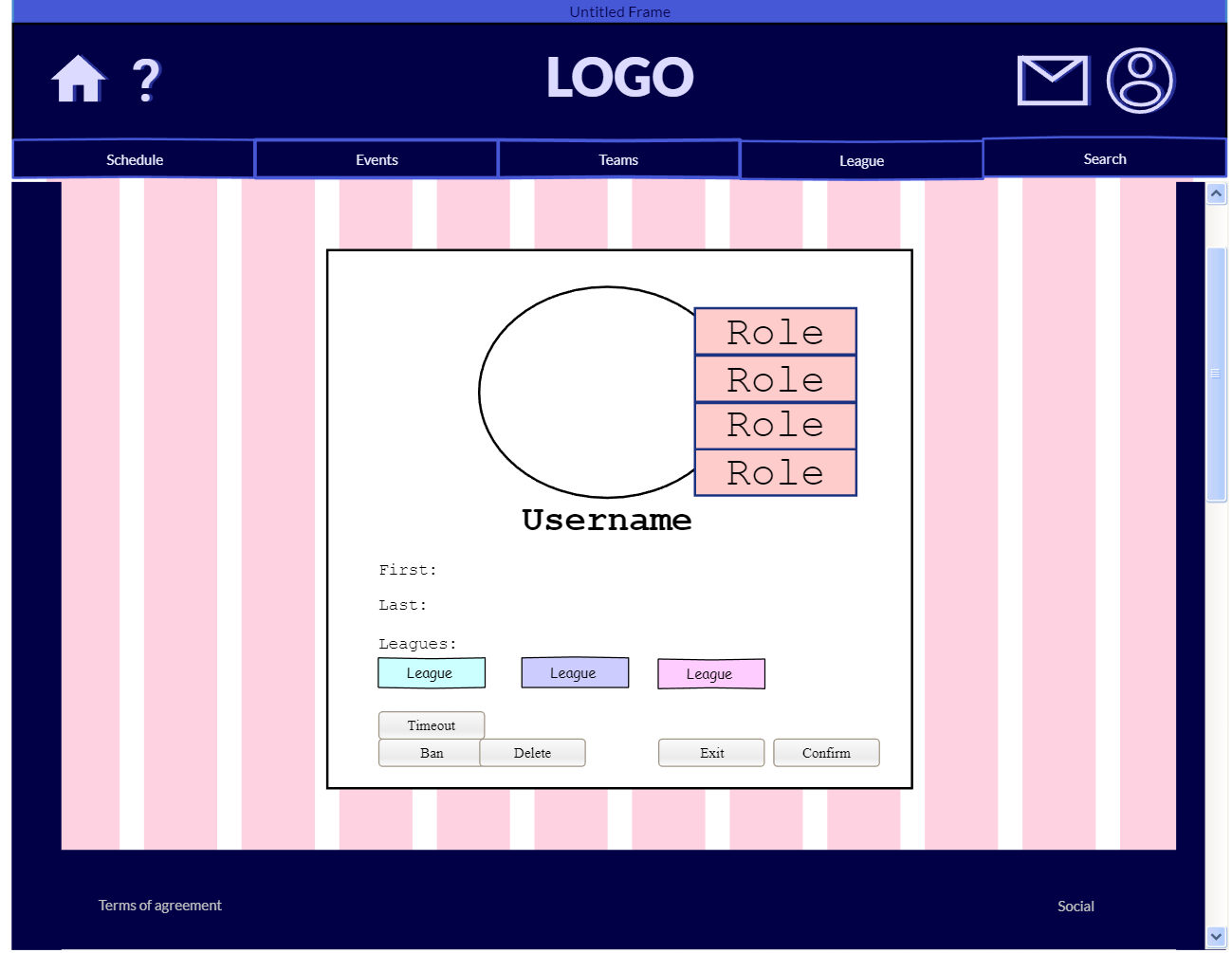
Admin\_003\_Desktop



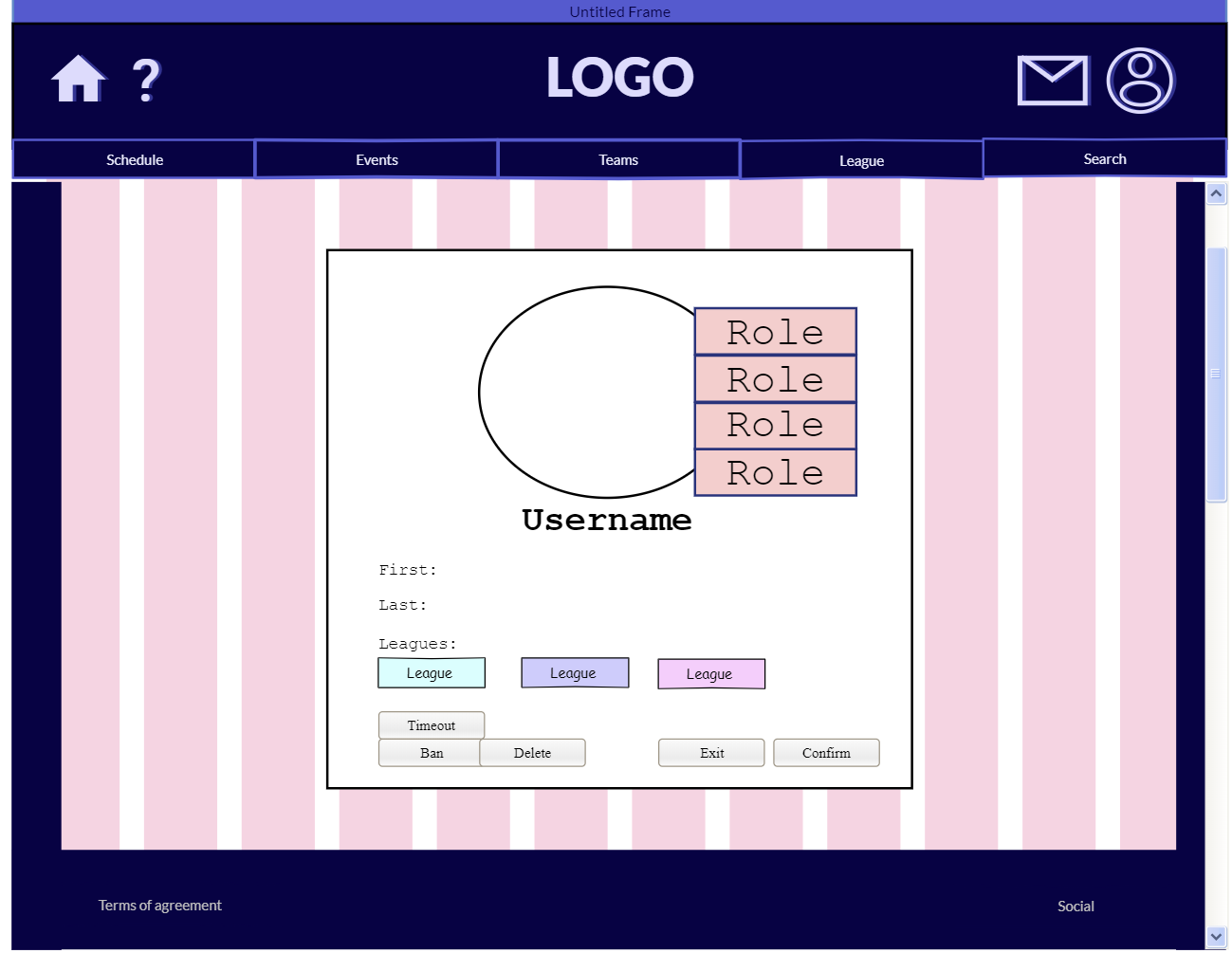
Admin\_003\_Web



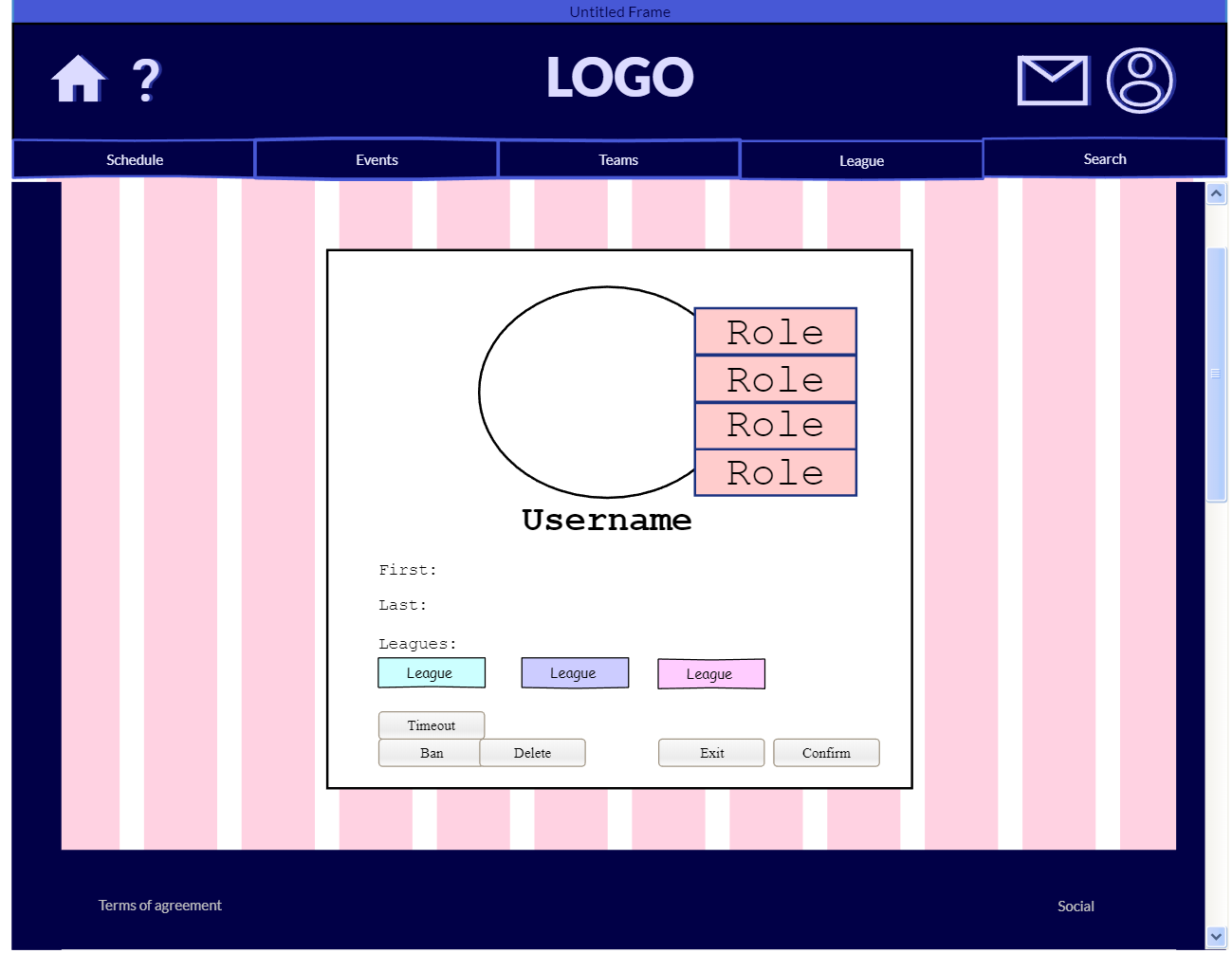
Admin\_005\_Desktop



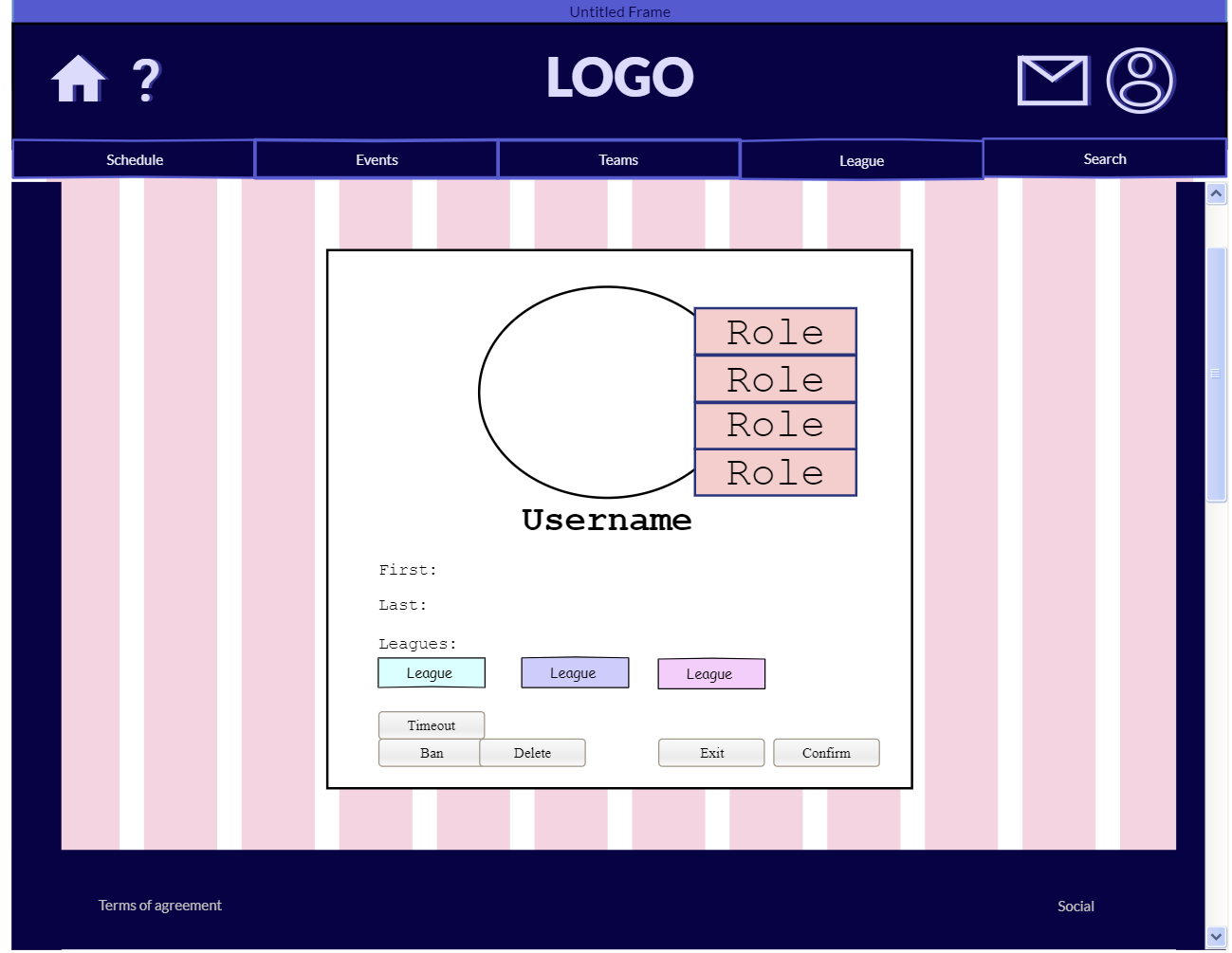
Admin\_005\_Web



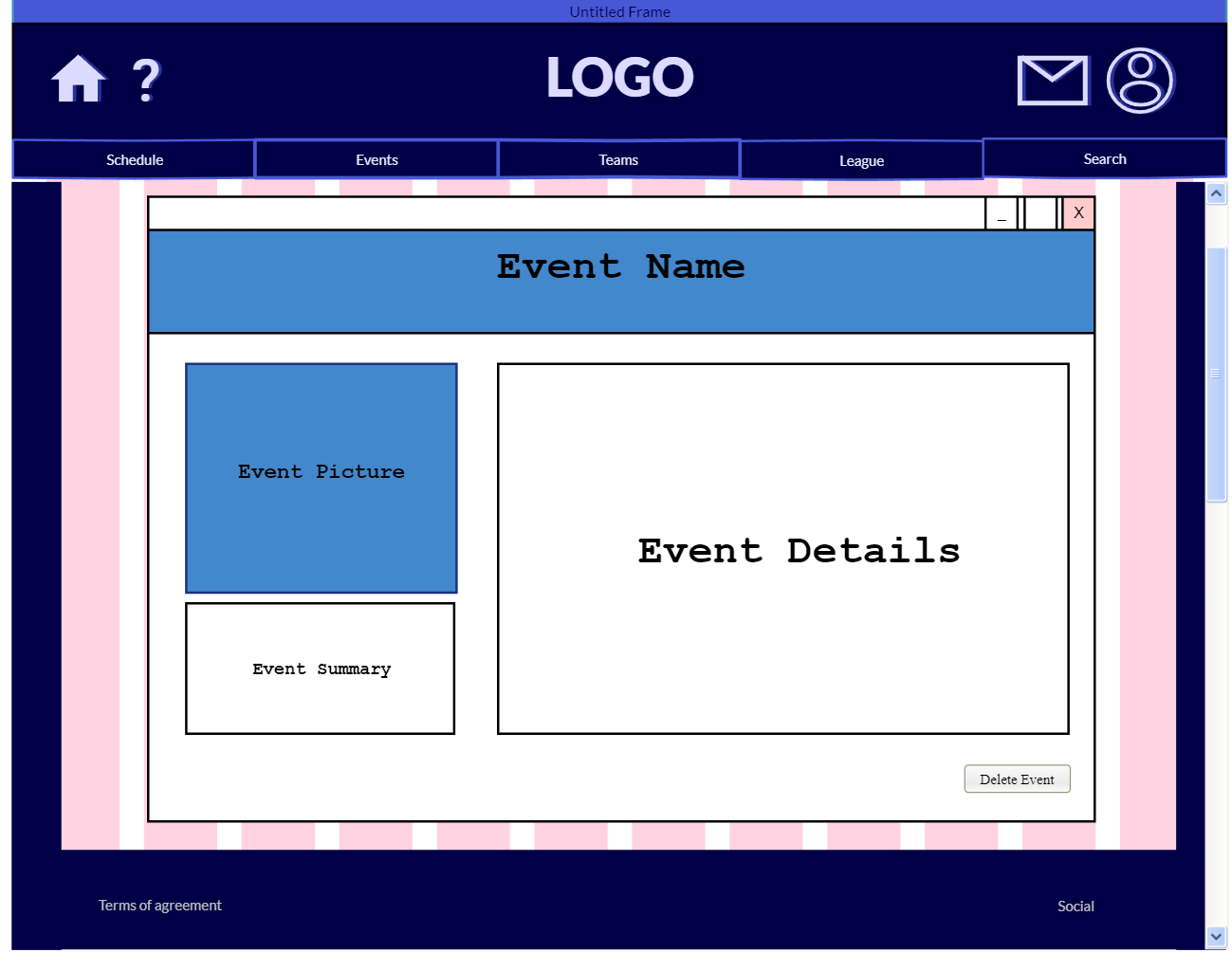
Admin\_006\_Desktop



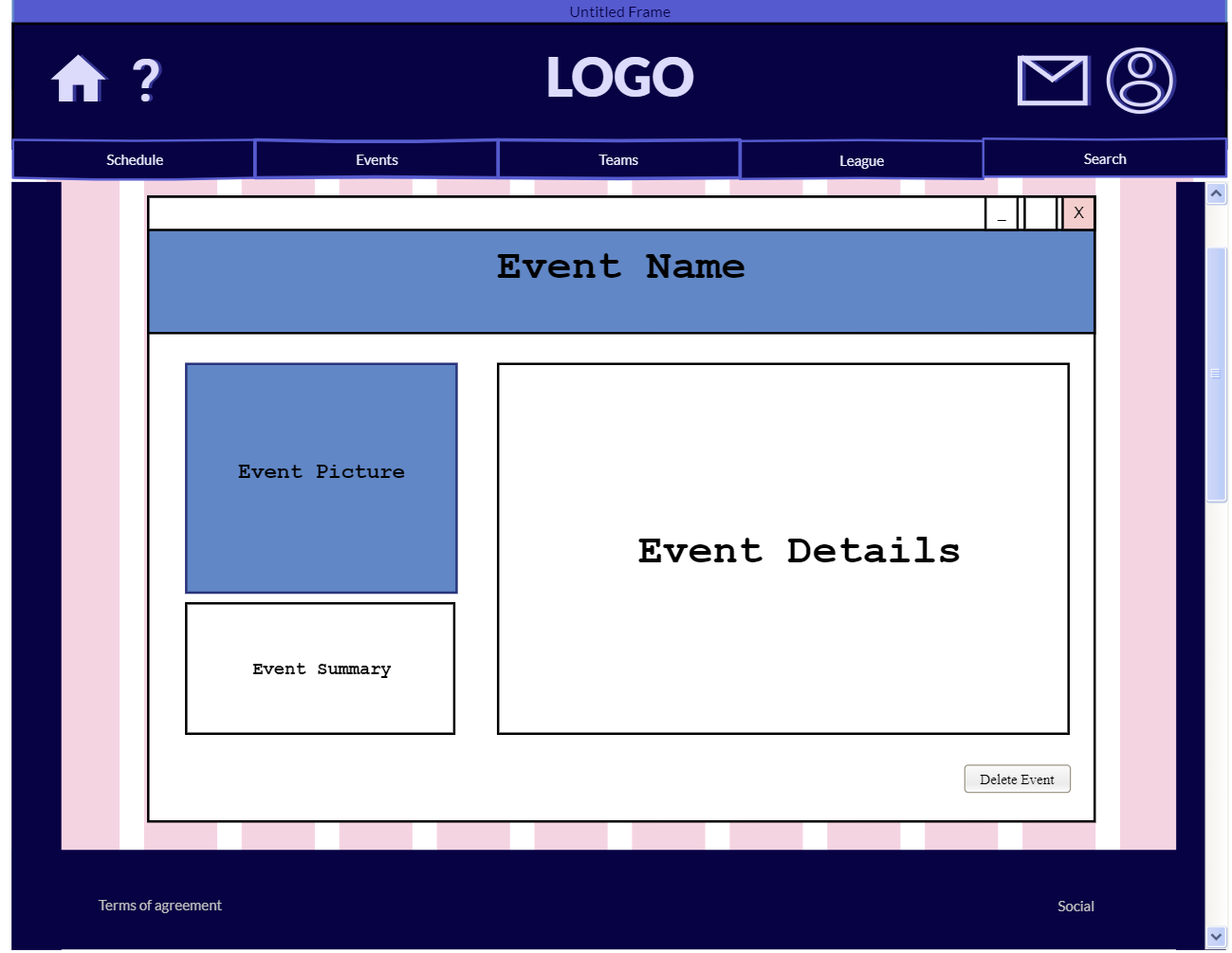
Admin\_006\_Web



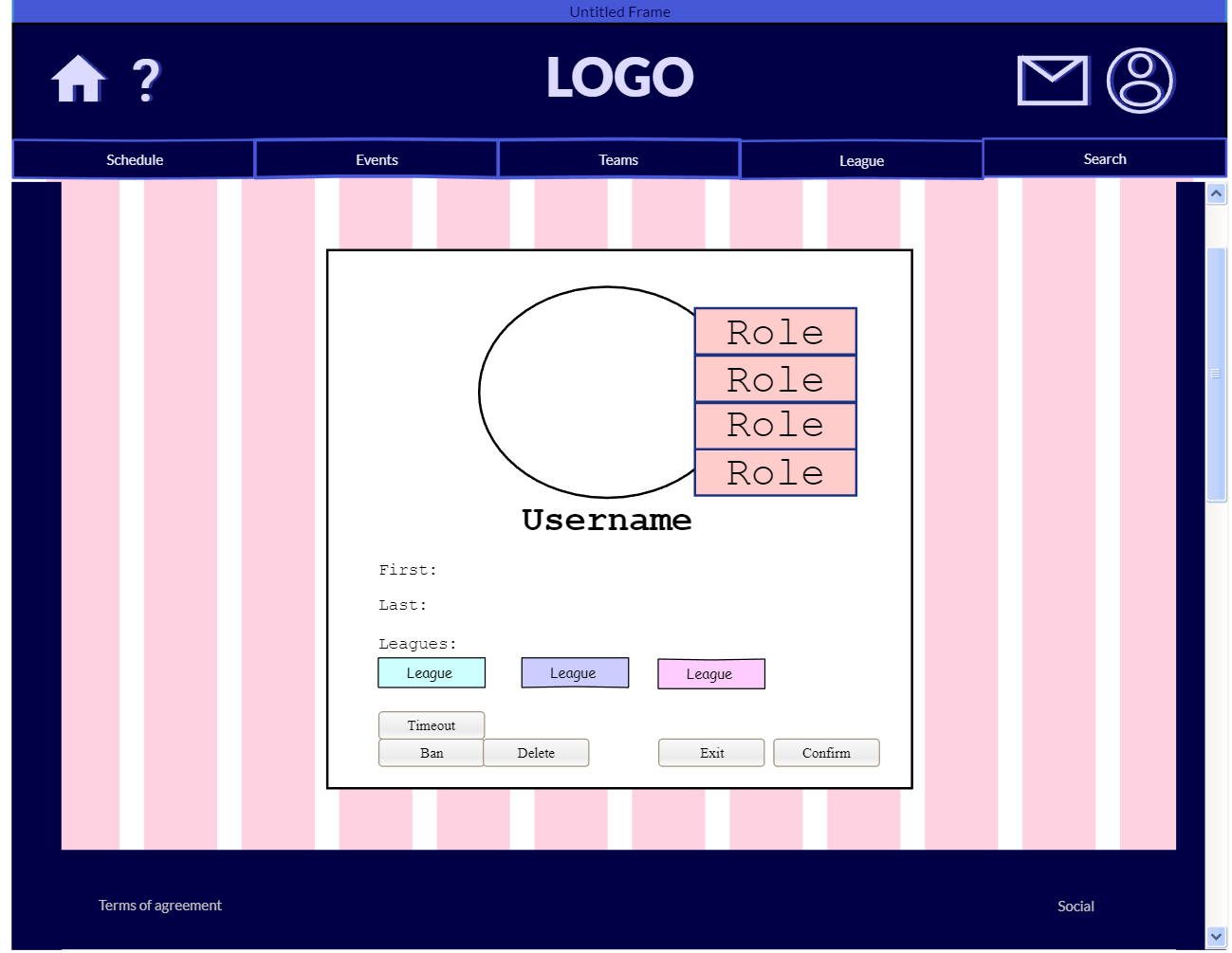
Admin\_007\_Desktop



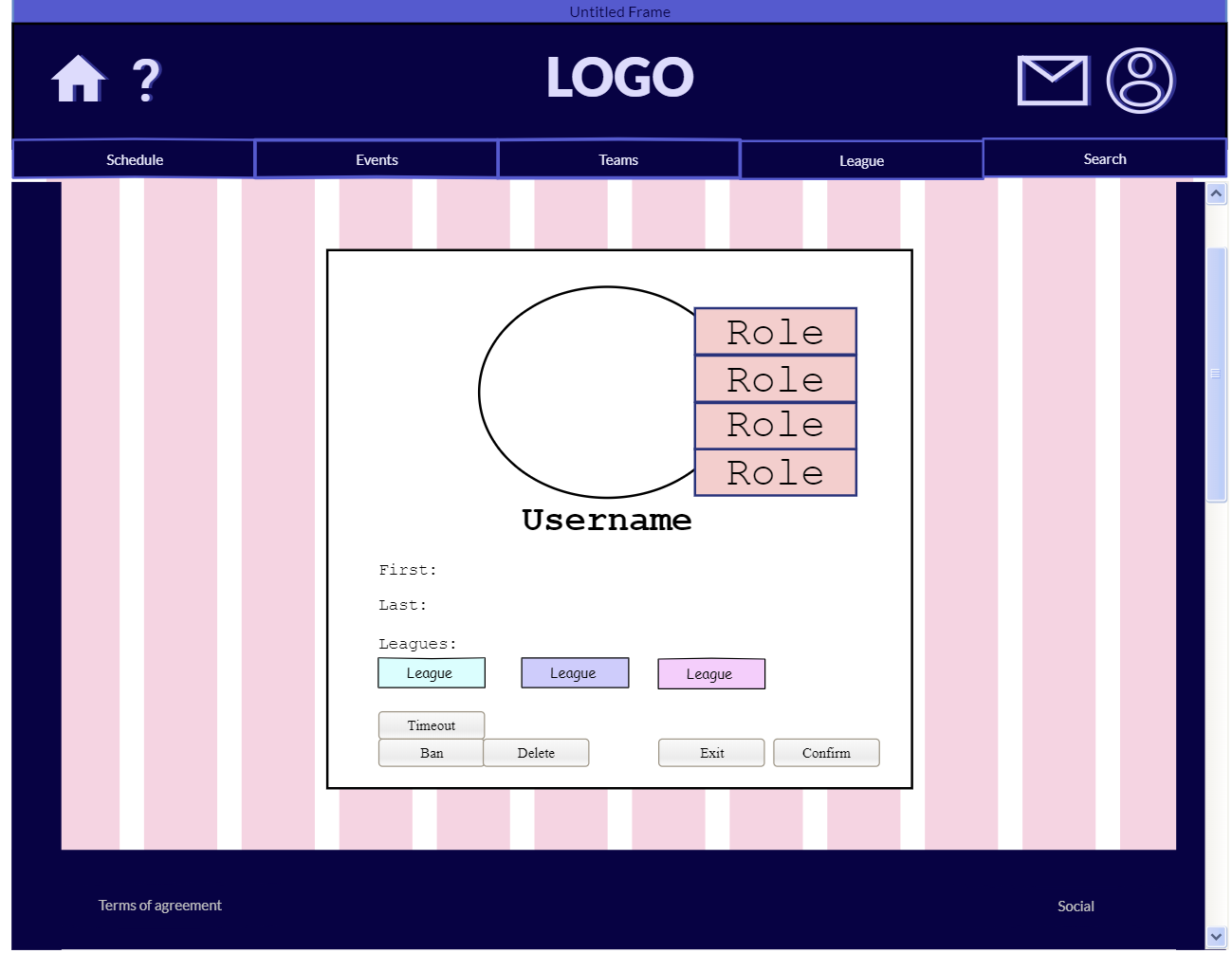
Admin\_007\_Web



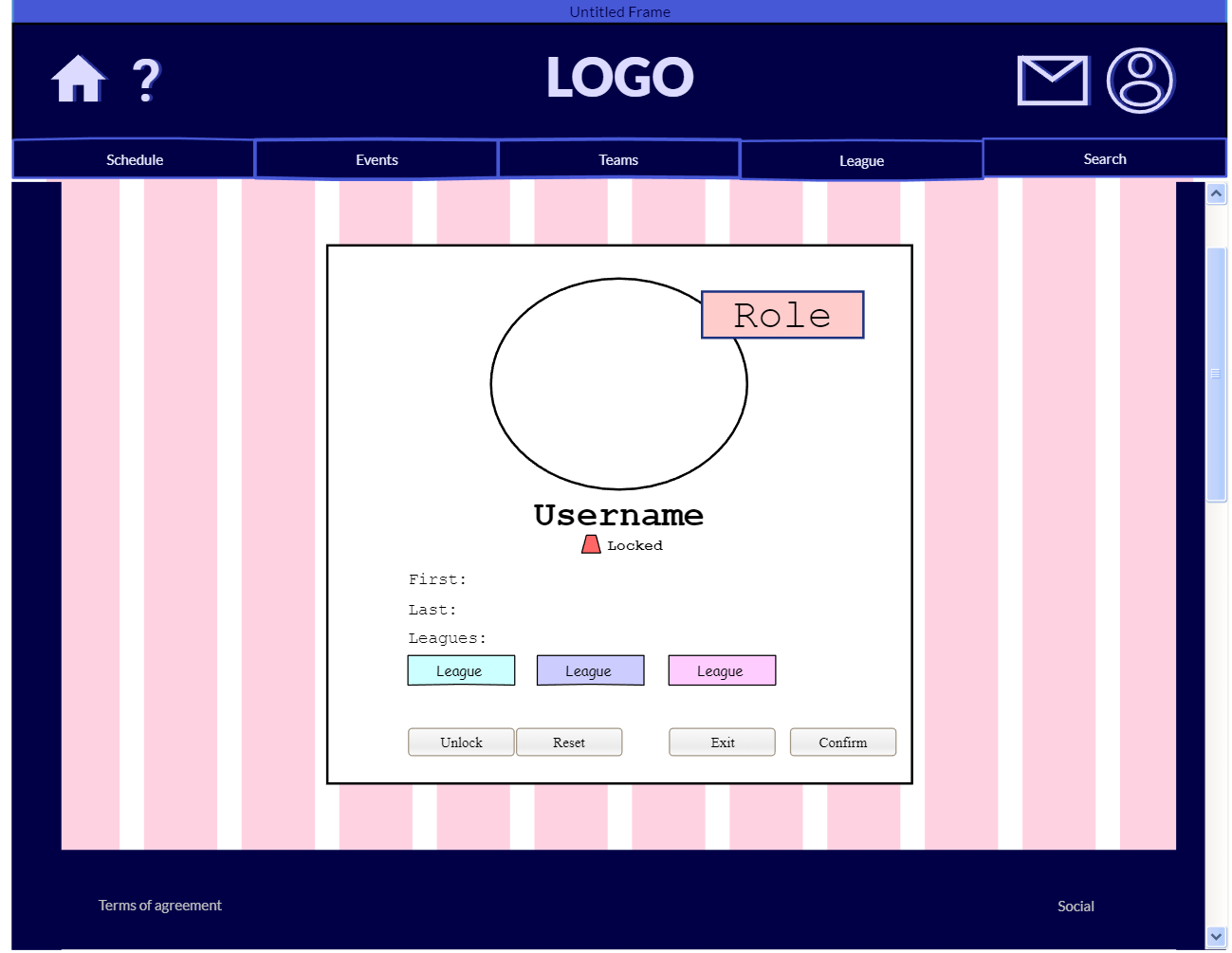
Admin\_008\_Desktop



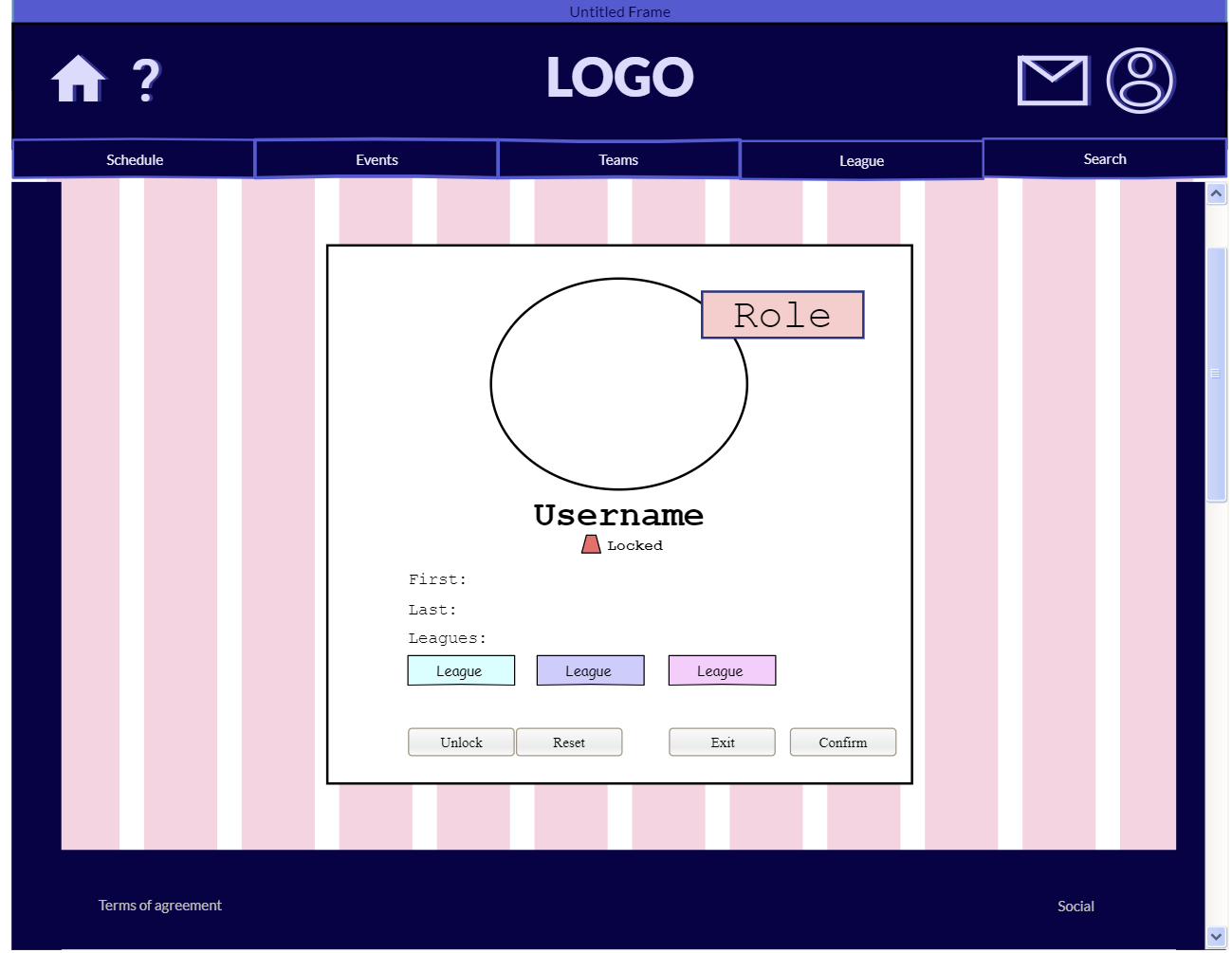
Admin\_008\_Web



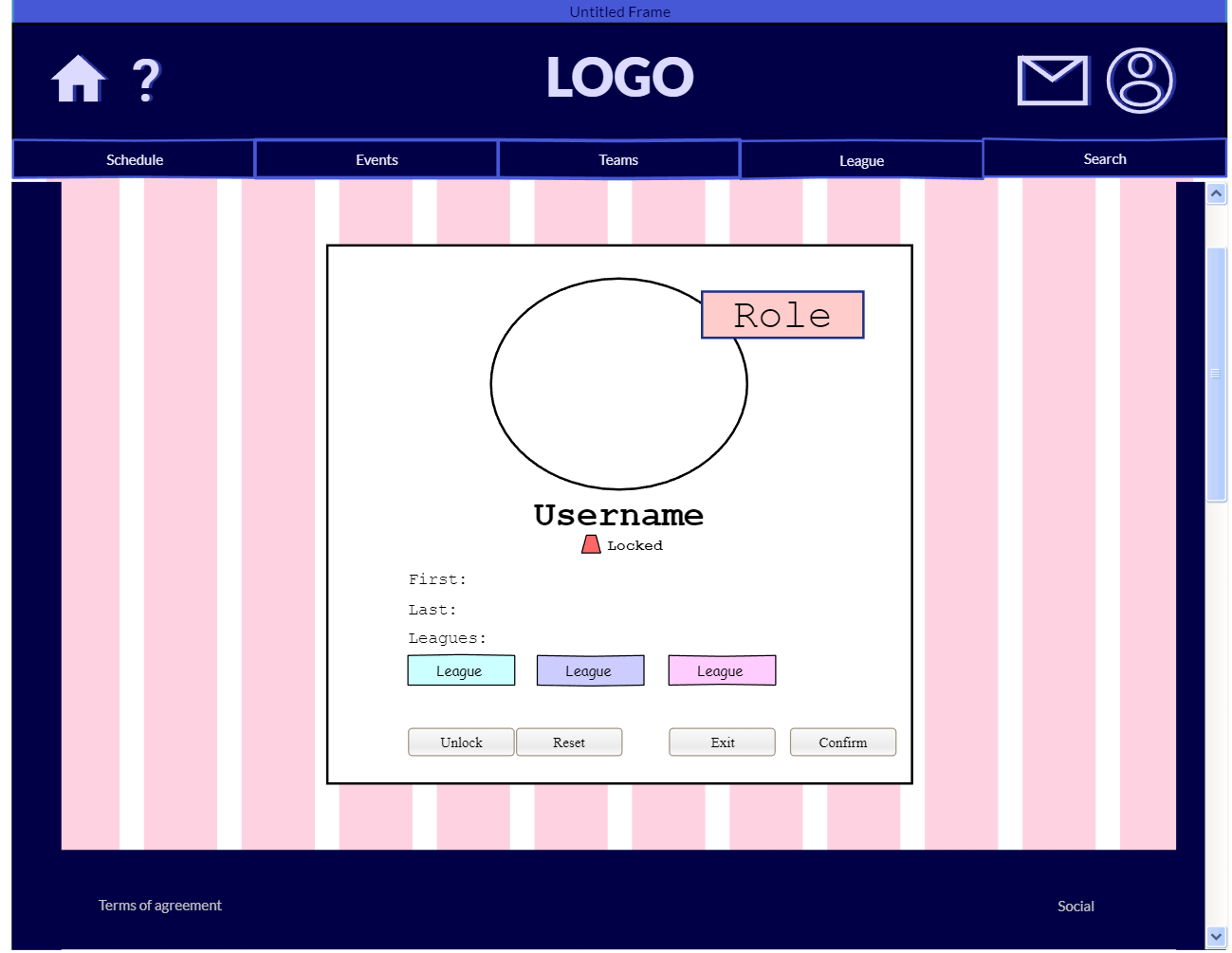
Admin\_009\_Desktop



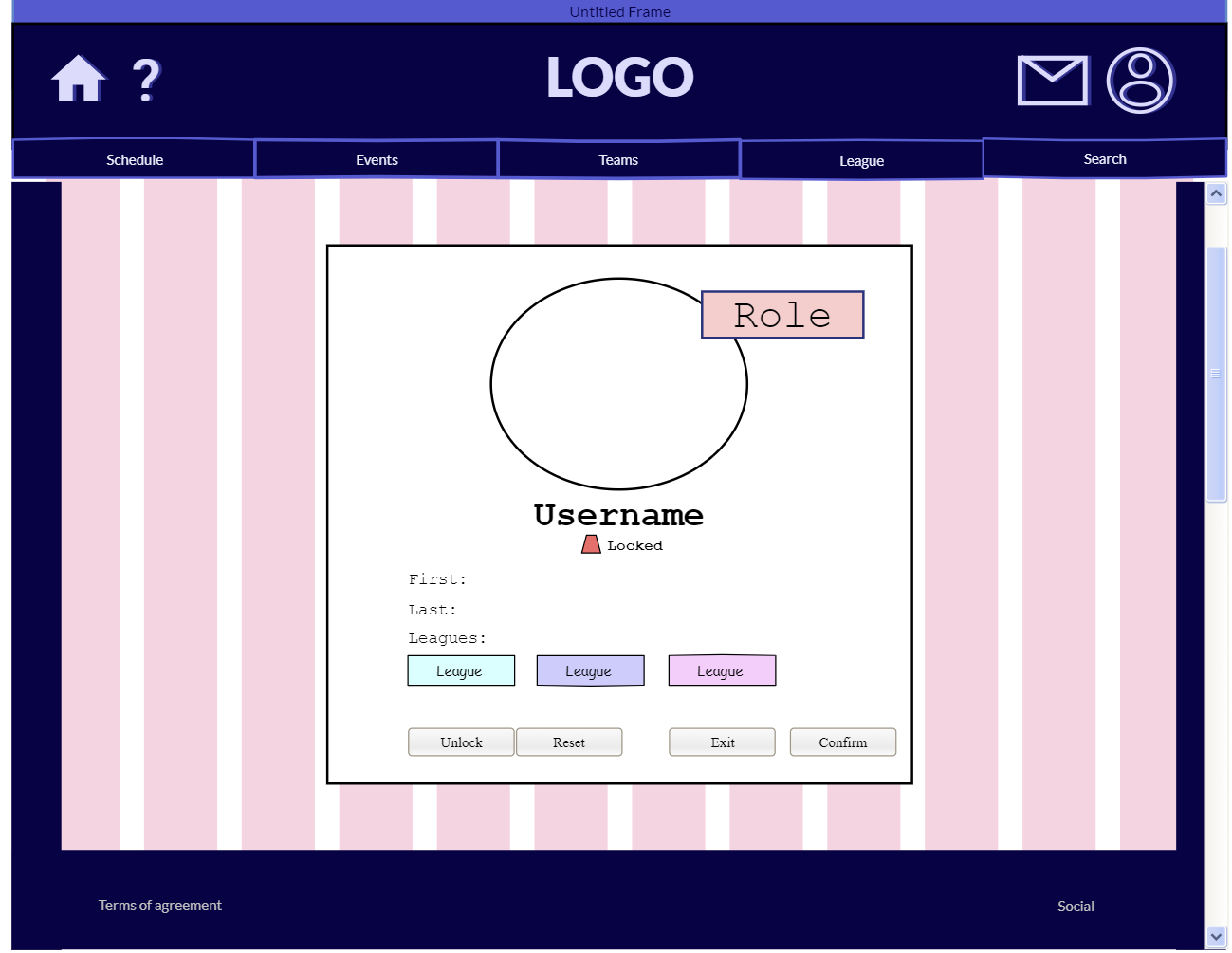
Admin\_009\_Web



Admin\_010\_Desktop



Admin\_010\_Web



### Moderator

## Others

### Score Reporter

### Volunteers

# Data Flow Diagrams

## Participants

### Player

### Potential Player

### Spectator

### Commentator

### Videographer

## Organizers

### League Admin

### Tournament Admin

### Coach

### Treasurer

## Logistics

### Equipment Manager

### Donor

### Vendor

### Venue Representative

## Site Administrators

### User

### Administrator

### Moderator

## Others

### Score Reporter

### Volunteers

# Activity Diagrams

## Participants

### Player

### Potential Player

### Spectator

### Commentator

### Videographer

## Organizers

### League Admin

### Tournament Admin

### Coach

### Treasurer

## Logistics

### Equipment Manager

### Donor

### Vendor

### Venue Representative

## Site Administrators

### User

### Administrator

### Moderator

## Others

### Score Reporter

### Volunteers

# State Chart Diagrams

## Participants

### Player

### Potential Player

### Spectator

### Commentator

### Videographer

## Organizers

### League Admin

### Tournament Admin

### Coach

### Treasurer

## Logistics

### Equipment Manager

### Donor

### Vendor

### Venue Representative

## Site Administrators

### User

### Administrator

### Moderator

## Others

### Score Reporter

### Volunteers

# Sequence Diagrams

## Participants

### Player

### Potential Player

### Spectator

### Commentator

### Videographer

## Organizers

### League Admin

### Tournament Admin

### Coach

### Treasurer

## Logistics

### Equipment Manager

### Donor

### Vendor

### Venue Representative

## Site Administrators

### User

### Administrator

### Moderator

## Others

### Score Reporter

### Volunteers

# Data Dictionary

# Data Domain

## Storage Models

## View Models

# Database Design

# Entity Relational Diagram

# Stored Procedure Definitions

# UML Models – Data Management Class Diagrams

# UML Models – Data Access Class Diagrams

# Implementation Plan

## Project Structure

## Timeline

## Technology and Tools

## Testing

## QA Process