
TCU Sports Broadcasting

FrogCrew Use Cases

Version 3.0

FrogCrew	Version: 3.1
Use Cases	Date: 4/10/2025
FrogCrew Use Cases	

Revision History

Date	Version	Description	Author
10/02/2024	1.0	Initial Draft of Use Cases	Aliya Suri, Kate Bednarz, Manuel Burciaga, James Edmonson, James Clarke, Dave Park, Michala Rogers
2/25/2025	2.0	Post Christmas Break Updates	Kate Bednarz, James Edmonson, James Clarke, Dave Park, Michala Rogers
2/25/2025	3.0	Major Overhaul and Changes + Reformatting	Michala Rogers
4/10/2025	3.1	Rewriting certain UCs for more clarity and more edits to the overall document	Michala Rogers

FrogCrew	Version: 3.1
Use Cases	Date: 4/10/2025
FrogCrew Use Cases	

Table of Contents

Use Case 1: Crew Member Creates Crew Member Profile	7
Use Case 2: Crew Member Edits Profile	9
Use Case 3: Crew Member Views A Crew Member’s Profile	11
Use Case 4: Crew Member Views Their Scheduled Games	12
Use Case 5: Crew Member Views General Game Schedule	13
Use Case 6: Crew Member Views Crew List	15
Use Case 7: Crew Member Submits Availability	16
Use Case 8: Crew Member Edits Availability	17
Use Case 9: Crew Member ‘A’ Requests Coverage	18
Use Case 10: Crew Member ‘B’ Accepts/Denies the Exchange	20
Use Case 11: Admin Approves Coverage	22
Use Case 12: Admin/Crew Member Views Notification	23
Use Case 13: Admin/Crew Member Deletes Notification	24
Use Case 14: Admin Invites Crew Member	26
Use Case 15: Admin Deletes A Crew Member	27
Use Case 16: Admin views crew members	29
Use Case 17: Admin Views Crew Member’s Availability	31
Use Case 18: Admin Creates Game Schedule	32
Use Case 19: Admin Edits Crew Member’s Profile	33
Use Case 20: Admin Adds Games To Game Schedule	34
Use Case 21: Admin Edits The Schedule	35
Use Case 22: Admin Edits Games In Game Schedule	36
Use Case 23: Admin Schedules Crew	38
Use Case 24: Admin Publishes Game Schedule	40
Use Case 25: Admin Generates Crew List To Send	41
Use Case 26: Admin Generates Financial Report	43
Use Case 27: Admin Generates Position Report	45
Use Case 28: Admin Generates Individual Crew Member Report	47
Use Case 29: Admin Creates a Position	49
Use Case 30: Admin Edits a Position	50
Use Case 31: Admin Finds Positions	51
Use Case 32: Admin Creates a Crew List Template	52
Use Case 33: Admin Finds Crew List Templates	53
Use Case 34: Admin Views a Crew List Template	54
Use Case 35: Admin Edits a Crew List Template	55
Use Case 36: Admin Deletes a Crew List Template	56

FrogCrew	Version: 3.1
Use Cases	Date: 4/10/2025
FrogCrew Use Cases	

Use Cases

Use Case List

Primary Actor	Use Cases
Crew Member	UC 1: Creates Profile UC 2: Edits Profile UC 3: Views Other Crew Member's Profile UC 4: Views Their Scheduled Games UC 5: Views General Game Schedule UC 6: Views Crew List UC 7: Submits Availability UC 8: Edits Availability UC 9: Requests Shift Exchange UC 10: Accepts/Denies Shift Exchange
Admin	UC 11: Approves Shift Exchange UC 12: Views Notifications UC 13: Deletes Notification UC 14: Invites Crew Member UC 15: Deletes Crew Member UC 16: Views Crew Members UC 17: Views Crew Member's Availability UC 18: Creates Game Schedule UC 19: Edits Crew Member's Profile UC 20: Adds Games to Game Schedule UC 21: Edits the Schedule UC 22: Edits Game Schedule UC 23: Schedules Crew UC 24: Publishes Game Schedule UC 25: Generates Crew List to Send UC 26: Generates Financial Report UC 27: Generates Position Report UC 28: Generates Individual Crew Member Report UC 29: Creates a Position UC 30: Edits a Position UC 31: Finds Positions UC 32: Creates a Crew List Template UC 33: Finds Crew List Templates UC 34: View a Crew List Template UC 35: Edits a Crew List Template UC 36: Deletes a Crew List Template

FrogCrew	Version: 3.1
Use Cases	Date: 4/10/2025
FrogCrew Use Cases	

FrogCrew	Version: 3.1
Use Cases	Date: 4/10/2025
FrogCrew Use Cases	

Use Case 1: Crew Member Creates Crew Member Profile

UC ID and Name:	UC-1: Crew Member Creates Crew Member Profile																						
Created By:	Aliya Suri	Date Created:	10/01/24																				
Primary Actor:	Crew Member	Secondary Actors:																					
Trigger:	The Crew Member clicks the registration link in the invitation email.																						
Description:	The Crew Member wants to set up an account, so that they can be crewed for games.																						
Preconditions:	PRE-1. An invitation email is sent to the Crew Member.																						
Postconditions:	POST-1. The Crew Member account is set up.																						
Main Success Scenario:	<ol style="list-style-type: none"> 1. The Crew Member clicks the registration link in the invitation email. 2. The System opens a new page and asks the Crew Member to enter the details of this new account according to the “Details” defined in the Associated Information of this use case. 3. The Crew Member enters the details of this new account and confirms that they have finished. 4. The System validates the Crew Member’s inputs according to the “Details” defined in the Associated Information of this use case. 5. The System displays the details of the new account and asks the Crew Member to confirm the registration. 6. The Crew Member either confirms the registration (continues the normal flow) or chooses to modify the details (return to step 3). 7. The System saves the information about the new account and informs the Crew Member that this account has been created. 8. The System redirects the Crew Member to the login page. 9. Use case ends. 																						
Extensions:	<p>2a. The Crew Member has already set up the account:</p> <p>2a1. The System alerts the Crew Member that they have already set up their account and shall log in.</p> <p>2a2. The System redirects the Crew Member to the login page.</p> <p>4a. Input validation rule violation:</p> <p>4a1. The System alerts the Crew Member that an input validation rule is violated and displays the nature and location of the error.</p> <p>4a2. The Crew Member corrects the mistake and returns to step 4 of the normal flow.</p>																						
Priority:	High																						
Frequency of Use:	Approximately 35-40 users, 1 usage per year.																						
Business Rules:	<p>BR-1: A Crew Member must have a unique email address in the system.</p> <p>BR-2: Crew Members can only edit their own profile.</p> <p>BR-3: Phone numbers must follow the format 999-999-9999.</p> <p>BR-4: All required fields (First Name, Last Name, Email, Phone Number, Role, Qualified Position) must be filled before saving changes.</p>																						
Associated Information:	<table border="1"> <thead> <tr> <th>Property name</th> <th>Data type</th> <th>Editability</th> <th>Validation rule</th> </tr> </thead> <tbody> <tr> <td>First Name</td> <td>String</td> <td>Yes</td> <td>Required</td> </tr> <tr> <td>Last Name</td> <td>String</td> <td>Yes</td> <td>Required</td> </tr> <tr> <td>Email</td> <td>String</td> <td>Yes</td> <td>Required, Valid Email</td> </tr> <tr> <td>Phone Number</td> <td>String</td> <td>Yes</td> <td>Required, 999-999-999 Format</td> </tr> </tbody> </table>			Property name	Data type	Editability	Validation rule	First Name	String	Yes	Required	Last Name	String	Yes	Required	Email	String	Yes	Required, Valid Email	Phone Number	String	Yes	Required, 999-999-999 Format
Property name	Data type	Editability	Validation rule																				
First Name	String	Yes	Required																				
Last Name	String	Yes	Required																				
Email	String	Yes	Required, Valid Email																				
Phone Number	String	Yes	Required, 999-999-999 Format																				

FrogCrew	Version: 3.1
Use Cases	Date: 4/10/2025
FrogCrew Use Cases	

	<table border="1"> <tr> <td>Password</td> <td>String</td> <td>Yes</td> <td>Required</td> </tr> <tr> <td>Role</td> <td>String</td> <td>Yes</td> <td>Required</td> </tr> <tr> <td>Qualified Position</td> <td>String</td> <td>Yes</td> <td>Required</td> </tr> </table>	Password	String	Yes	Required	Role	String	Yes	Required	Qualified Position	String	Yes	Required
Password	String	Yes	Required										
Role	String	Yes	Required										
Qualified Position	String	Yes	Required										
Related Use Cases	UC 2: Crew Member Edits Profile UC 14: Admin Invites Crew Member UC 16: Admin Views Crew Members												
Assumptions:													
Open Issues:													

FrogCrew	Version: 3.1
Use Cases	Date: 4/10/2025
FrogCrew Use Cases	

Use Case 2: Crew Member Edits Profile

UC ID and Name:	UC-2: Crew Member Edits Profile																										
Created By:	Aliya Suri	Date Created:	10/01/24																								
Primary Actor:	Crew Member	Secondary Actors:																									
Trigger:	The Crew Member indicates to change the details of their account.																										
Description:	The Crew Member wants to change the details of their account, so that they can update their profile or change the password.																										
Preconditions:	PRE-1. The Crew Member is logged into the System																										
Postconditions:	POST-1. Changes made to the account are stored in the system.																										
Main Success Scenario:	<ol style="list-style-type: none"> 1. The Crew Member indicates to change the details of their account. 2. The System displays the details of their account. 3. The Crew Member chooses to change the details of this account. 4. The System asks the Crew Member to make changes to this account where allowed according to the “Details” defined in the Associated Information and the “Security/access concerns” defined in the Business Rules of this use case. 5. The Crew Member makes changes to this account until they confirm that they have finished changing. 6. The System validates the Crew Member’s changes and alerts warning messages according to the “Details” defined in the Associated Information of this use case. 7. The Crew Member acknowledges the warnings and chooses to continue. 8. The System displays the updated details of this account and alerts the Crew Member to confirm the change. 9. The Crew Member either confirms the change (continues the normal flow) or chooses to continue to change the details (return to step 5). 10. The System saves the changes, carries out the effect of change according to the “Details” defined in the Associated Information of this use case, and informs the Student that this account has been changed. 11. Use case ends. 																										
Extensions:	6a. Input validation rule violation: <ol style="list-style-type: none"> 6a1. The System alerts the Crew Member that an input validation rule is violated and displays the nature and location of the error. 6a2. The Crew Member corrects the mistake and returns to step 6 of the normal flow. 																										
Priority:	High																										
Frequency of Use:	Rare. Approximately 35-40 users, 1 usage per year.																										
Business Rules:	BR-2: Crew Members can only edit their own profile. BR-3: Phone numbers must follow the format 999-999-9999. BR-4: All required fields (First Name, Last Name, Email, Phone Number, Role, Qualified Position) must be filled before saving changes.																										
Associated Information:	Details: <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th>Property name</th> <th>Data type</th> <th>Editability</th> <th>Validation rule</th> </tr> </thead> <tbody> <tr> <td>First name</td> <td>String</td> <td>Yes</td> <td>Required</td> </tr> <tr> <td>Last name</td> <td>String</td> <td>Yes</td> <td>Required</td> </tr> <tr> <td>Phone number</td> <td>String</td> <td>Yes</td> <td>Required, (999999-9999 format)</td> </tr> <tr> <td>Email</td> <td>String</td> <td>Yes</td> <td>Required, valid email</td> </tr> <tr> <td>Crew member Positions</td> <td>String</td> <td>Yes</td> <td>Required</td> </tr> </tbody> </table>			Property name	Data type	Editability	Validation rule	First name	String	Yes	Required	Last name	String	Yes	Required	Phone number	String	Yes	Required, (999999-9999 format)	Email	String	Yes	Required, valid email	Crew member Positions	String	Yes	Required
Property name	Data type	Editability	Validation rule																								
First name	String	Yes	Required																								
Last name	String	Yes	Required																								
Phone number	String	Yes	Required, (999999-9999 format)																								
Email	String	Yes	Required, valid email																								
Crew member Positions	String	Yes	Required																								

FrogCrew	Version: 3.1
Use Cases	Date: 4/10/2025
FrogCrew Use Cases	

	<table border="1"> <tr> <td>Password</td> <td>String</td> <td>Required</td> </tr> </table>	Password	String	Required
Password	String	Required		
	The Crew Member shall be able to cancel the use case at any time prior to submitting it.			
Related Use Cases:	UC 1: Crew Member Creates Profile UC 16: Admin Views Crew Members UC 17: Admin Views Crew Member's Availability UC 19: Admin Edits Crew Member's Profile			
Assumptions:				
Open Issues:				

FrogCrew	Version: 3.1
Use Cases	Date: 4/10/2025
FrogCrew Use Cases	

Use Case 3: Crew Member Views A Crew Member's Profile

UC ID and Name:	UC-3: Crew Member Views A Crew Member's Profile		
Created By:	Aliya Suri	Date Created:	10/01/24
Primary Actor:	Crew Member	Secondary Actors:	
Trigger:	The User indicates to view the details of a Crew Member's profile.		
Description:	The User wants to view the details of a Crew Member's profile, so that they can contact that Crew Member.		
Preconditions:	PRE-1. The User is logged into the System.		
Postconditions:	POST-1. The details of the specified Crew Member profile are displayed to the User.		
Main Success Scenario:	<ol style="list-style-type: none"> 1. The User indicates to view the details of a Crew Member profile. 2. The User finds a list of Crew Members through <u>UC-12: Views crew members</u>. 3. The User views the list and chooses to view the details of one specific Crew Member. 4. The System retrieves and displays details of this Crew Member profile according to the "Details" defined in the Associated Information and the "Security/access concerns" defined in the Business Rules of this use case. 5. The User views the details of this Crew Member profile. 6. Use case ends. 		
Extensions:			
Priority:	Medium		
Frequency of Use:	Approximately *** user, average of *** usages per week.		
Business Rules:	BR-13: Crew Members can only access directory information of other Crew Members. BR-37: Only Admins with the correct permissions can access sensitive data.		
Associated Information:	Details: <ul style="list-style-type: none"> • Same as the "Details" defined in the Associated Information of <u>UC-1: Set up a crew member account</u> 		
Related Use Cases:	UC 6: Crew Member Views Crew List UC 16: Admin Views Crew Members		
Assumptions:			
Open Issues:			

FrogCrew	Version: 3.1
Use Cases	Date: 4/10/2025
FrogCrew Use Cases	

Use Case 4: Crew Member Views Their Scheduled Games

UC ID and Name:	UC-4: Crew Member Views Their Scheduled Games		
Created By:	Michala Rogers	Date Created:	1/29/2025
Primary Actor:	Crew Member	Secondary Actors:	
Trigger:	The user indicates to view their scheduled games.		
Description:	The user wants to view their scheduled games.		
Preconditions:	PRE-1: The user is logged into the system. PRE-2: The game schedule has been created. PRE-3: The game schedule has been posted.		
Postconditions:	POST-1: The details of the game schedule are displayed to the user. POST-2: The user has access to detailed information about each game, including date, time, location, assigned role, and reporting instructions.		
Main Success Scenario:	<ol style="list-style-type: none"> 1. The user logs into the system. 2. The user navigates to the "View Schedule" section. 3. The system retrieves the users' assigned game schedule. 4. The system displays a list of their scheduled games, including the following details: <ol style="list-style-type: none"> a. Game Date b. Game Time c. Opponent d. Venue e. Assigned Role f. Reporting Time & Location 5. The user reviews their scheduled games. 6. The user can filter, sort, or search for specific games. 7. The user exits the schedule view. 8. Use case ends. 		
Extensions:	3a. No games scheduled for the Crew Member: <ul style="list-style-type: none"> • 3a1. The system displays a message: "You are not scheduled for any upcoming games." • 3a2. The Crew Member can choose to return to the main dashboard. 		
Priority:	Medium		
Frequency of Use:	Approximately 50 users, average of 2 usage per week.		
Business Rules:	BR-1: Crew Members can only view their own schedule. BR-2: Schedule data must be accurate and updated in real time. BR-3: Only Admins can modify the schedule.		
Associated Information:	<ul style="list-style-type: none"> • Sorting options: Date, Opponent, Venue • Notifications: If a schedule update occurs, the Crew Member is notified 		
Related Use Cases:	UC 5: Crew Member Views General Game Schedule UC 22: Admin Schedules Crew UC 23: Admin Publishes Game Schedule		
Assumptions:	The schedule has been finalized and posted. Crew Members have access to their assigned schedules via the system.		
Open Issues:	Determine if crew members should be able to view the entire game schedule or just their assigned games. Consider adding a feature where crew members can request schedule changes or swaps		

FrogCrew	Version: 3.1
Use Cases	Date: 4/10/2025
FrogCrew Use Cases	

	directly from this view.
--	--------------------------

Use Case 5: Crew Member Views General Game Schedule

UC ID and Name:	UC-5: Crew Member Views General Game Schedule		
Created By:	Michala Rogers	Date Created:	2/25/25
Primary Actor:	Crew Member	Secondary Actors:	
Trigger:	The User indicates to view the game schedule.		
Description:	The Crew Member wants to view the game schedule to check upcoming games, event details, and available positions. This ensures they are aware of upcoming events and can plan their availability accordingly.		
Preconditions:	PRE-1: A game schedule has been created and published. PRE-2: The Crew Member is logged into the system.		
Postconditions:	POST-1: The details of the general game schedule are displayed to the Crew Member. POST-2: The Crew Member can view key game details such as date, time, location, opponent, and required crew positions.		
Main Success Scenario:	<ol style="list-style-type: none"> 1. The Crew Member navigates to the "View Game Schedule" section. 2. The System retrieves and displays the general game schedule 3. The Crew Member reviews the schedule. 4. The Crew Member can filter, sort, or search for specific games. 5. The Crew Member exits the schedule view. 6. Use case ends. 		
Extensions:	<p>3a. No games are currently scheduled:</p> <ul style="list-style-type: none"> • 3a1. The system displays a message: "No upcoming games available at this time." • 3a2. The Crew Member can return to the main dashboard. <p>5a. The Crew Member wants to view more details about a specific game:</p> <ul style="list-style-type: none"> • 5a1. The Crew Member selects a game from the list. • 5a2. The system retrieves and displays additional information about that game (e.g., assigned crew, reporting time, and contact details). 		
Priority:	Medium		
Frequency of Use:	Approximately 50 users, averaging 5-10 views per week.		
Business Rules:	BR-5: Crew Members can only view their own schedule. BR-9: Only Admins can modify the schedule.		
Associated Information:	<p>Game Schedule Information:</p> <ul style="list-style-type: none"> • Game Date • Game Time • Opponent • Venue • Required Crew Positions <p>Sorting Options:</p> <ul style="list-style-type: none"> • Sort by Date (Newest to Oldest) • Sort by Opponent • Sort by Venue <p>Notifications:</p> <ul style="list-style-type: none"> • If a schedule update occurs, the Crew Member receives a notification. 		

FrogCrew	Version: 3.1
Use Cases	Date: 4/10/2025
FrogCrew Use Cases	

Related Use Cases:	waiting until use case numbers are solidified
Assumptions:	UC 4: Crew Member Views Their Scheduled Games UC 22: Admin Schedules Crew UC 23: Admin Publishes Game Schedule
Open Issues:	The game schedule is finalized before being displayed to Crew Members. The Crew Member has the correct permissions to access the schedule.

FrogCrew	Version: 3.1
Use Cases	Date: 4/10/2025
FrogCrew Use Cases	

Use Case 6: Crew Member Views Crew List

UC ID and Name:	UC-6: Crew Member Views Crew List																																						
Created By:	Kate Bednarz	Date Created:	01/25/25																																				
Primary Actor:	Crew member	Secondary Actors:																																					
Trigger:	The user indicates to view a crew list.																																						
Description:	The user wants to view a crew list.																																						
Preconditions:	PRE-1: A crew list has been created.																																						
Postconditions:	POST-1: The details of the specified crew list are displayed to the user.																																						
Main Success Scenario:	<ol style="list-style-type: none"> 1. The user indicates to view a crew list for a specific game. 2. The system retrieves and displays details of the crew list according to the “Details” defined in the Associated information. 3. The user views the details of the crew list. 4. Use case ends. 																																						
Extensions:																																							
Priority:	Medium																																						
Frequency of Use:	Approximately 30 users, average of 2 usages per week.																																						
Business Rules:	BR-13: Crew Members can only access directory information of other Crew Members.																																						
Associated Information:	<p>Details:</p> <table border="1"> <thead> <tr> <th>Property name</th> <th>Data type</th> <th>Editability</th> <th>Validation rule</th> </tr> </thead> <tbody> <tr> <td>Sport</td> <td>String</td> <td>Yes</td> <td>Required</td> </tr> <tr> <td>Opponent</td> <td>String</td> <td>Yes</td> <td>Required</td> </tr> <tr> <td>Game date</td> <td>Datetime</td> <td>Yes</td> <td>Required</td> </tr> <tr> <td>Game time</td> <td>Datetime</td> <td>Yes</td> <td>Required</td> </tr> <tr> <td>Position</td> <td>String <ul style="list-style-type: none"> • Producer • Asst Prod • Director • Asst Director • Technical Dir • Graphics • Bug Op • Replay EVS • EIC • Video • Audio • Camera • Utility • Tech manager • TOC • Observer </td> <td>Yes</td> <td>Required</td> </tr> <tr> <td>Name</td> <td>String</td> <td>Yes</td> <td>Required</td> </tr> <tr> <td>Report time</td> <td>Datetime</td> <td>Yes</td> <td>Required</td> </tr> <tr> <td>Report location</td> <td>String</td> <td>Yes</td> <td>Required</td> </tr> </tbody> </table>			Property name	Data type	Editability	Validation rule	Sport	String	Yes	Required	Opponent	String	Yes	Required	Game date	Datetime	Yes	Required	Game time	Datetime	Yes	Required	Position	String <ul style="list-style-type: none"> • Producer • Asst Prod • Director • Asst Director • Technical Dir • Graphics • Bug Op • Replay EVS • EIC • Video • Audio • Camera • Utility • Tech manager • TOC • Observer 	Yes	Required	Name	String	Yes	Required	Report time	Datetime	Yes	Required	Report location	String	Yes	Required
Property name	Data type	Editability	Validation rule																																				
Sport	String	Yes	Required																																				
Opponent	String	Yes	Required																																				
Game date	Datetime	Yes	Required																																				
Game time	Datetime	Yes	Required																																				
Position	String <ul style="list-style-type: none"> • Producer • Asst Prod • Director • Asst Director • Technical Dir • Graphics • Bug Op • Replay EVS • EIC • Video • Audio • Camera • Utility • Tech manager • TOC • Observer 	Yes	Required																																				
Name	String	Yes	Required																																				
Report time	Datetime	Yes	Required																																				
Report location	String	Yes	Required																																				
Related Use Cases:	UC 3: Crew Member Views Other Crew Member’s Profile UC 16: Admin Views Crew Members UC 23: Admin Publishes Game Schedule																																						
Assumptions:																																							
Open Issues:																																							

FrogCrew	Version: 3.1
Use Cases	Date: 4/10/2025
FrogCrew Use Cases	

Use Case 7: Crew Member Submits Availability

UC ID and Name:	UC-7: Crew Member Submits Availability														
Created By:	Kate Bednarz	Date Created:	09/29/2024												
Primary Actor:	Crew Member	Secondary Actors:	Admin												
Trigger:	The Crew Member indicates to submit availability for a game.														
Description:	The Crew Member wants to submit their availability for a game, so that they can be scheduled to work a game.														
Preconditions:	PRE-1. The Crew Member is logged in to the System. PRE-2. The Admin has published a game schedule.														
Postconditions:	POST-1. The Crew Member's availability is stored in the System.														
Main Success Scenario:	<ol style="list-style-type: none"> 1. The Crew Member indicates to submit their availability for a game. 2. The System displays games that they haven't submitted availability for. 3. The Crew Member selects the games they are available for. 4. The System validates the Crew Member's inputs according to the "Details" defined in the Associated Information of this use case. 5. The System saves the new availability, and informs the Crew Member that their availability has been saved. 6. The System notifies relevant actors about the creation of the availability according to the "Notification" defined in the Associated Information of this use case. 7. Use case ends. 														
Extensions:	4a. Input validation rule violation: <ol style="list-style-type: none"> 6a1. The System alerts the Crew Member that an input validation rule is violated and displays the nature and location of the error. 6a2. The Crew Member corrects the mistake and returns to step 6 of the normal flow. 														
Priority:	High														
Frequency of Use:	All crew members are users, average of 1 usage per user per new schedule release.														
Business Rules:	BR-6: Crew Members must submit availability before being scheduled. BR-8: Availability updates notify the Admin.														
Associated Information:	Details: <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th>Property name</th> <th>Data type</th> <th>Editability</th> <th>Validation rule</th> </tr> </thead> <tbody> <tr> <td>Available</td> <td>Boolean</td> <td>Yes</td> <td>Required</td> </tr> <tr> <td>Comment</td> <td>String</td> <td>Yes</td> <td></td> </tr> </tbody> </table> Notification: <ul style="list-style-type: none"> • The System sends a notification to the Admin 			Property name	Data type	Editability	Validation rule	Available	Boolean	Yes	Required	Comment	String	Yes	
Property name	Data type	Editability	Validation rule												
Available	Boolean	Yes	Required												
Comment	String	Yes													
Related Use Cases	UC 8: Crew Member Edits Availability UC 17: Admin Views Crew Member's Availability UC 22: Admin Schedules Crew														
Assumptions:															
Open Issues:															

FrogCrew	Version: 3.1
Use Cases	Date: 4/10/2025
FrogCrew Use Cases	

FrogCrew	Version: 3.1
Use Cases	Date: 4/10/2025
FrogCrew Use Cases	

Use Case 8: Crew Member Edits Availability

UC ID and Name:	UC-8: Crew Member Edits Availability		
Created By:	Kate Bednarz	Date Created:	09/29/2024
Primary Actor:	Crew member	Secondary Actors:	Admin
Trigger:	The crew member indicates that they wish to edit their availability.		
Description:	The crew member desires to edit their availability after it has already been submitted.		
Preconditions:	PRE-1. The crew member is logged into the system. PRE-2. The crew member has submitted their availability.		
Postconditions:	POST-1. The crew member's availability is updated in the system.		
Main Success Scenario:	<ol style="list-style-type: none"> 1. The crew member indicates they want to edit their availability. 2. The system displays the crew member's previously submitted availability. 3. The crew member edits their availability. 4. The system validates the changes and asks the crew member to confirm the change. 5. The Crew Member either confirms the changes (continues the normal flow) or chooses to modify the details (return to step 3). 6. The system notifies the admin about the availability update. 7. Use case ends. 		
Extensions:	3a. The crew member does not change their availability: 3a1. The crew member's availability stays the same. 4a. The crew member does not submit their new availability: 4a1. The crew member's availability is not updated.		
Priority:	Medium		
Frequency of Use:	Approximately 1 usage per game season of each crew member.		
Business Rules:	BR-6: Crew Members must submit availability before being scheduled. BR-7: Crew Members cannot edit availability for games that have already been scheduled. BR-8: Availability updates notify the Admin.		
Associated Information:	Notifications: The system should notify the admin if significant changes are made to the availability that may impact scheduling.		
Related Use Cases:	UC 7: Crew Member Submits Availability UC 17: Admin Views Crew Member's Availability UC 22: Admin Schedules Crew		
Assumptions:			
Open Issues:			

FrogCrew	Version: 3.1
Use Cases	Date: 4/10/2025
FrogCrew Use Cases	

Use Case 9: Crew Member ‘A’ Requests Coverage

UCID and Name:	UC-9: Crew Member ‘A’ Requests to Exchange Shift													
Created By:	Manuel Burciaga	Date Created:	10/1/2024											
Primary Actor:	Crew Member A (request coverage)	Secondary Actors:	Admin											
Trigger:	Crew Member A is unable to work their assigned shift and initiates the process to drop or exchange the shift.													
Description:	Crew Member A initiates the request to exchange or drop a shift, making it available for other qualified crew members to pick up.													
Preconditions:	PRE-1: Crew Member A is logged into the system. PRE-2: Crew Member A has an assigned shift they want to drop or exchange. PRE-3: The shift has not started yet.													
Postconditions:	POST-1: The shift is made available for other qualified crew members to pick up. POST-2: Crew Member A is notified if no one picks up the shift before the deadline.													
Main Success Scenario:	<ol style="list-style-type: none"> 1. Crew Member A logs into the system and navigates to their schedule. 2. Crew Member A selects the "Request Coverage" option for their assigned shift. 3. The system checks that the shift meets the criteria for being dropped (e.g., it is not within 24 hours of the start time). 4. The system notifies all eligible crew members (with the same role) about the available shift. 5. Crew Member A receives confirmation that their shift has been successfully covered and is picked up by another crew member. 6. Use case ends. 													
Extensions:	<p>3a. The shift is not eligible for exchange:</p> <ul style="list-style-type: none"> • 3a1: The system notifies Crew Member A that cannot be exchanged due to business rules (e.g., within 24 hours of start). <p>5a. No crew member volunteers for the shift before the deadline:</p> <ul style="list-style-type: none"> • 5a1: The system notifies Crew Member A that they are still responsible for the shift. • 5a2: Admin intervention may be required if Crew Member A is unable to work the shift • 													
Priority:	Medium													
Frequency of Use:	Dependent on crew shift schedules and availability													
Business Rules:	BR-10: A Crew Member may request shift coverage only if the shift is more than 24 hours away.													
Associated Information:	<table border="1"> <thead> <tr> <th>Details</th> <th>Property Name</th> <th>Data Type</th> <th>Editability</th> <th>Validation Rule</th> </tr> </thead> <tbody> <tr> <td>Crew Member A</td> <td>String</td> <td>Yes</td> <td>Required</td> <td>Must be an active user</td> </tr> </tbody> </table>				Details	Property Name	Data Type	Editability	Validation Rule	Crew Member A	String	Yes	Required	Must be an active user
	Details	Property Name	Data Type	Editability	Validation Rule									
Crew Member A	String	Yes	Required	Must be an active user										

FrogCrew	Version: 3.1
Use Cases	Date: 4/10/2025
FrogCrew Use Cases	

	Shift ID	String	No	Required	Must be a valid shift
	<p>Notification:</p> <ul style="list-style-type: none"> • The system notifies all eligible crew members about the available shift. • Crew Member A is notified if the shift is not picked up before the deadline. 				
Related Use Cases:	UC 10: Crew Member Accepts/Denies Shift Exchange UC 11: Admin Approves Shift Exchange UC 22: Admin Schedules Crew				
Assumptions:					
Open Issues:					

FrogCrew	Version: 3.1
Use Cases	Date: 4/10/2025
FrogCrew Use Cases	

Use Case 10: Crew Member 'B' Accepts/Denies the Exchange

UC ID and Name:	UC-10: Crew Member 'B' Accepts/Denies the Exchange		
Created By:	Manuel Burciaga	Date Created:	1/14/2025
Primary Actor:	Crew Member B (picking up or volunteering for the shift)	Secondary Actors:	Crew Member A (original shift holder), Admin (notified if required)
Trigger:	Crew Member B receives a notification of an available shift and decides whether to accept or decline it.		
Description:	Crew Member B reviews the available shift and decides to either accept or deny the request to take on the shift. If accepted, the system updates the schedule to assign the shift to Crew Member B.		
Preconditions:	PRE-1: Crew Member B is logged into the system. PRE-2: The shift is available for Crew Member B to pick up. PRE-3: Crew Member B has the same role required for the shift		
Postconditions:	POST-1: If accepted, the shift is assigned to Crew Member B, and Crew Member A is relieved of responsibility. POST-2: If denied, the shift remains available until picked up by another crew member or the deadline passes		
Main Success Scenario:	<ol style="list-style-type: none"> 1. Crew Member B logs into the system and navigates to the "Available Shifts" section. 2. Crew Member B selects the available shift they wish to review. 3. The system verifies Crew Member B's eligibility for the shift (e.g., matching role). 4. Crew Member B accepts the shift. 5. The system updates the schedule, assigning the shift to Crew Member B and removing it from the "Available Shifts" list. 6. The system notifies Crew Member A that the shift has been picked up. 7. Both Crew Member A and Crew Member B receive confirmation of the successful shift exchange. 8. If required, the system notifies the Admin of the completed exchange. 9. Use case ends. 		
Extensions:	<p>3a. Crew Member B is not eligible for the shift:</p> <ul style="list-style-type: none"> ● 3a1: The system notifies Crew Member B that they do not meet the criteria for the shift. <p>4a. Crew Member B declines the shift:</p> <ul style="list-style-type: none"> ● 4a1: The system keeps the shift available for other qualified crew members to pick up. 		
Priority:	Medium		

FrogCrew	Version: 3.1
Use Cases	Date: 4/10/2025
FrogCrew Use Cases	

Frequency of Use:	Dependent on crew member schedules and availability				
Business Rules:	BR-11: Only eligible Crew Members with the same role can accept a shift exchange.				
Associated Information:	Details	Property Name	Data Type	Editability	Validation Rule
	Crew Member B	String	Yes	Required	Must be an active user with the same role
	Shift ID	String	No	Required	Must be a valid shift
	Notification: <ul style="list-style-type: none"> • The system notifies Crew Member B about the available shift. • The system sends confirmation to Crew Member A once the shift is picked up. • The system notifies the Admin if approval is required. 				
Related Use Cases:	UC 9: Crew Member Requests Shift Exchange UC 11: Admin Approves Shift Exchange UC 12: Admin Views Notifications				
Assumptions:					
Open Issues:					

FrogCrew	Version: 3.1
Use Cases	Date: 4/10/2025
FrogCrew Use Cases	

Use Case 11: Admin Approves Coverage

UC ID and Name:	UC-11: Admin Approves Coverage		
Created By:	Michala Rogers	Date Created:	1/29/2025
Primary Actor:	Admin	Secondary Actors:	Crew Member
Trigger:	A crew member submits a shift exchange request that requires admin approval.		
Description:	The Admin reviews and approves or denies a shift exchange request submitted by a crew member. This ensures shift coverage is maintained and meets operational requirements.		
Preconditions:	PRE-1: A shift exchange request has been submitted. PRE-2: The shift exchange meets eligibility criteria (e.g., both crew members are eligible, the shift is within the exchangeable timeframe).		
Postconditions:	POST-1: If approved, the system updates the schedule to reflect the new shift assignments. POST-2: If denied, the system notifies the involved crew members and provides a reason for the decision.		
Main Success Scenario:	<ol style="list-style-type: none"> 1. The Admin accesses the list of pending shift exchange requests. 2. The Admin selects a request to review. 3. The system displays details of the shift exchange request, including crew members involved and shift details. 4. The Admin reviews the request and either approves or denies it. 5. If approved, the system updates the schedule accordingly. 6. If denied, the Admin provides a reason, and the system notifies the requesting crew members. 7. The Admin completes the review process. 8. Use case ends. 		
Extensions:	<p>4a. Shift exchange request violates scheduling rules:</p> <ul style="list-style-type: none"> • 4a1. The system alerts the Admin about the violation. • 4a2. The Admin either denies the request or overrides the rule (if permitted). <p>4b. One or both crew members are no longer eligible for the shift exchange:</p> <ul style="list-style-type: none"> • 4b1. The system notifies the Admin. • 4b2. The Admin either denies the request or contacts the crew members for further resolution. 		
Priority:	High		
Frequency of Use:	Approximately 10 users, average of 5 usages per week.		
Business Rules:	BR-12: Admin approval is required for shift exchanges in all roles.		
Associated Information:	<ul style="list-style-type: none"> • Shift exchange request details include: <ul style="list-style-type: none"> ○ Original shift owner ○ Requested shift replacement ○ Date and time of exchange ○ Reason for exchange 		
Related Use Cases:	UC 9: Crew Member Requests Shift Exchange UC 10: Crew Member Accepts/Denies Shift Exchange UC 12: Admin Views Notifications UC 22: Admin Schedules Crew		
Assumptions:	Admin has sufficient permissions to approve or deny shift exchanges. The scheduling system enforces business rules and eligibility checks.		

FrogCrew	Version: 3.1
Use Cases	Date: 4/10/2025
FrogCrew Use Cases	

Open Issues:	Should the system auto-approve certain exchanges based on predefined rules? How should emergency shift exchanges be handled?
--------------	---

Use Case 12: Admin/Crew Member Views Notification

UC ID and Name:	UC-12: Admin/Crew Member Views Notification																		
Created By:	Manuel Burciaga	Date Created:	10/1/2024																
Primary Actor:	Admin																		
Trigger:	The User indicates to view notifications.																		
Description:	The user will be able to view notifications relevant to them.																		
Preconditions:	PRE-1. The user must be logged in																		
Postconditions:	POST-1. The notification is optionally either marked as read.																		
Main Success Scenario:	<ol style="list-style-type: none"> 1. The admin logs into the system. 2. The system displays a notification icon, indicating unread notifications. 3. The admin clicks on the notification icon to view the list of pending notifications. 4. The system shows a detailed list of notifications, such as shift exchange requests, availability updates, or system alerts. 5. The admin reviews the notification and takes the required action, if any. 6. The notification is marked as read or archived after action is taken. 7. Use case ends 																		
Extensions:	<p>4a. No pending notifications:</p> <ul style="list-style-type: none"> • 4a1: The system displays a message stating "No new notifications." <p>5a. Admin action required for notification:</p> <ul style="list-style-type: none"> • 5a1: If the notification involves an action (e.g., approving a shift exchange), the system prompts the admin to take action before marking the notification as resolved 																		
Priority:	High																		
Frequency of Use:	Multiple times per day, depending on system and crew activities.																		
Business Rules:	BR-14: Crew Members receive notifications for schedule updates. BR-28: Admins must respond to actionable notifications within 24 hours.																		
Associated Information:	<table border="1"> <thead> <tr> <th>Detail</th> <th>Property Name</th> <th>Data Type</th> <th>Editability</th> <th>Validation Rule</th> </tr> </thead> <tbody> <tr> <td>Notification Type</td> <td>String</td> <td>No</td> <td>Required</td> <td>System-generated, related to crew</td> </tr> <tr> <td>Action Required</td> <td>Boolean</td> <td>No</td> <td>Optional</td> <td>If true, Admin must take</td> </tr> </tbody> </table>				Detail	Property Name	Data Type	Editability	Validation Rule	Notification Type	String	No	Required	System-generated, related to crew	Action Required	Boolean	No	Optional	If true, Admin must take
Detail	Property Name	Data Type	Editability	Validation Rule															
Notification Type	String	No	Required	System-generated, related to crew															
Action Required	Boolean	No	Optional	If true, Admin must take															

FrogCrew	Version: 3.1
Use Cases	Date: 4/10/2025
FrogCrew Use Cases	

	<table border="1"> <tr> <td></td> <td></td> <td></td> <td></td> <td>action</td> </tr> </table> <p>Notification:</p> <ul style="list-style-type: none"> The system sends an alert to the admin about new notifications. The system tracks whether the notification has been reviewed or requires action. 					action
				action		
Related Use Cases:	UC 10: Crew Member Accepts/Denies Shift Exchange UC 11: Admin Approves Shift Exchange UC 13: Admin Deletes Notification UC 22: Admin Schedules Crew					
Assumptions:						
Open Issues:						

Use Case 13: Admin/Crew Member Deletes Notification

UC ID and Name:	UC-13: Admin/Crew Member Deletes Notification		
Created By:	Michala Rogers	Date Created:	2/3/2025
Primary Actor:	Admin Crew Member	Secondary Actors:	
Trigger:	The Admin or Crew Member indicates they want to delete a notification from their system inbox.		
Description:	The Admin or Crew Member wants to remove a notification from their notification list after reviewing or addressing it. This ensures that only relevant, unresolved notifications remain visible.		
Preconditions:	PRE-1: The Admin or Crew Member is logged into the system. PRE-2: There is at least one notification present in their notification inbox.		
Postconditions:	POST-1: The selected notification is deleted from the user's notification list. POST-2: The system updates the notification status accordingly.		
Main Success Scenario:	<ol style="list-style-type: none"> The Admin or Crew Member logs into the system. The user navigates to the "Notifications" section. The system displays a list of unread and read notifications. The user selects one or multiple notifications to delete. The system prompts the user to confirm the deletion. The user confirms the deletion. The system removes the notification(s) from the user's notification list. The system updates the user's notification inbox. Use case ends. 		
Extensions:	<p>4a. No notifications available to delete:</p> <ul style="list-style-type: none"> 4a1. The system displays a message stating "No notifications available to delete." <p>5a. User cancels deletion confirmation:</p> <ul style="list-style-type: none"> 5a1. The system retains the notification(s) in the inbox. 5a2. The user is returned to the notification list. 		
Priority:	Low		
Frequency of Use:	Approximately 50 users, average of 10 usages per week.		

FrogCrew	Version: 3.1
Use Cases	Date: 4/10/2025
FrogCrew Use Cases	

Business Rules:	BR-15: Deleted notifications cannot be recovered. BR-29: Notifications should not be deleted until the Admin has reviewed them. BR-30: The system automatically removes notifications after 30 days if no action is required.
Associated Information:	The system should provide filtering options for read/unread notifications before deletion. The system may automatically delete old notifications after a predefined time period (e.g., 30 days).
Related Use Cases:	UC 12: Admin Views Notifications
Assumptions:	Users should have control over managing their notification inbox. System-critical notifications may require an archive option instead of direct deletion.
Open Issues:	Should certain high-priority notifications (e.g., final schedule changes) be restricted from deletion? Should the system allow bulk deletion of notifications?

FrogCrew	Version: 3.1
Use Cases	Date: 4/10/2025
FrogCrew Use Cases	

Use Case 14: Admin Invites Crew Member

UC ID and Name:	UC-14: Admin Invites Crew Member																							
Created By:	Manuel Burciaga	Date Created:																						
Primary Actor:	Admin	Secondary Actors:	Crew Member (receiving the invite)																					
Trigger:	The Admin indicates to invite new crew members.																							
Description:	The admin invites a new crew member by sending a link. The invited crew member receives an email with a link to create their profile.																							
Preconditions:	PRE-1. The Admin is logged into the System.																							
Postconditions:	POST-1: The crew member receives an email invitation with a registration link.																							
Main Success Scenario:	<ol style="list-style-type: none"> 1. The admin navigates to the "Invite Crew Member" section. 2. The admin provides email(s). 3. The system sends an email invitation to the crew member with a registration link. 4. Use case ends. 																							
Extensions:	4a. Invalid email address: <ul style="list-style-type: none"> • 4a1: The system detects the email is invalid and prompts the Admin to correct it. 																							
Priority:	High																							
Frequency of Use:	As needed when new crew members are added.																							
Business Rules:	BR-1: A Crew Member must have a unique email address in the system.																							
Associated Information:	<table border="1"> <thead> <tr> <th>Details</th> <th>Property Name</th> <th>Data Type</th> <th>Editability</th> <th>Validation Rule</th> </tr> </thead> <tbody> <tr> <td>Crew Member Name</td> <td>String</td> <td>Yes</td> <td>Required</td> <td>Must not Exceed 50 characters</td> </tr> <tr> <td>Crew Member Email</td> <td>String</td> <td>Yes</td> <td>Required</td> <td>Must be a valid, and unique email</td> </tr> <tr> <td>Crew Member Role</td> <td>String</td> <td>Yes</td> <td>Required</td> <td>Crew member must select one or more roles</td> </tr> </tbody> </table>				Details	Property Name	Data Type	Editability	Validation Rule	Crew Member Name	String	Yes	Required	Must not Exceed 50 characters	Crew Member Email	String	Yes	Required	Must be a valid, and unique email	Crew Member Role	String	Yes	Required	Crew member must select one or more roles
	Details	Property Name	Data Type	Editability	Validation Rule																			
	Crew Member Name	String	Yes	Required	Must not Exceed 50 characters																			
	Crew Member Email	String	Yes	Required	Must be a valid, and unique email																			
	Crew Member Role	String	Yes	Required	Crew member must select one or more roles																			
Notification:																								
<ul style="list-style-type: none"> • The system sends an email invitation to the new crew member. • The system sends a confirmation to the crew member once the profile is approved. 																								

FrogCrew	Version: 3.1
Use Cases	Date: 4/10/2025
FrogCrew Use Cases	

	<ul style="list-style-type: none"> The system notifies the admin if the crew member fails to complete the profile within a certain time frame (e.g., 7 days).
Related Use Cases:	UC 1: Crew Member Creates Profile UC 16: Admin Views Crew Members
Assumptions:	
Open Issues:	

Use Case 15: Admin Deletes A Crew Member

UC ID and Name:	UC-15: Admin Deletes A Crew Member		
Created By:	Michala Rogers	Date Created:	2/5/2025
Primary Actor:	Admin	Secondary Actors:	Crew Member
Trigger:	The Admin initiates the removal of a Crew Member from the system.		
Description:	The Admin removes a Crew Member who is no longer part of the team. This ensures that only active members remain in the system and prevents scheduling or communication issues with inactive users.		
Preconditions:	PRE-1: At least one Crew Member is registered in the system. PRE-2: The Admin is logged into the system. PRE-3: The Crew Member is not assigned to any future scheduled games.		
Postconditions:	POST-1: The Crew Member's account and associated data are removed from the system. POST-2: System notifications are sent to relevant users informing them of the deletion. POST-3: If required, historical data related to the Crew Member (e.g., past game assignments, reports) remains accessible for record-keeping.		
Main Success Scenario:	<ol style="list-style-type: none"> The Admin navigates to the "Manage Crew Members" section. The Admin selects the Crew Member to be deleted. The System prompts the Admin with a confirmation message, including any warnings about active assignments. The Admin confirms the deletion. The System checks if the Crew Member has any pending or assigned shifts. <ol style="list-style-type: none"> If no pending assignments exist, proceed to Step 6. If assignments exist, prompt the Admin to reassign or remove the Crew Member from schedules. The System deletes the Crew Member's profile from the active crew list. The System notifies the Admin that the deletion was successful. The System sends a notification to the Crew Member informing them that their account has been removed (if applicable). Use case ends. 		
Extensions:	<ol style="list-style-type: none"> The Crew Member has pending or scheduled shifts: <ul style="list-style-type: none"> 5a1. The System alerts the Admin that the Crew Member is assigned to upcoming shifts. 5a2. The Admin is prompted to reassign or remove the Crew Member from these shifts before proceeding. 5a3. Once reassigned or removed, the Admin can confirm the deletion. The Crew Member has historical work data that must be retained: 		

FrogCrew	Version: 3.1
Use Cases	Date: 4/10/2025
FrogCrew Use Cases	

	<ul style="list-style-type: none"> • 6a1. Instead of full deletion, the System archives the Crew Member’s account and marks it as inactive. • 6a2. The System ensures past assignments remain accessible for reporting purposes. <p>7a. Admin cancels the deletion:</p> <ul style="list-style-type: none"> • 7a1. The System retains the Crew Member’s account with no changes.
Priority:	Low
Frequency of Use:	Approximately 1 user, average of 1 usage per week.
Business Rules:	BR-16: Admins can only delete Crew Members who do not have upcoming scheduled games. BR-17: Admins cannot delete Crew Members if they are assigned to financial reports.
Associated Information:	Notification: <ul style="list-style-type: none"> • The System notifies the Admin of the successful deletion. • The System notifies the Crew Member of the deletion. <p>The Admin shall be able to cancel the process at any time prior to confirming it.</p>
Related Use Cases:	UC 16: Admin Views Crew Members UC 17: Admin Views Crew Member’s Availability UC 22: Admin Schedules Crew
Assumptions:	The Admin has the necessary permissions to delete a Crew Member. The System provides a warning before deletion if the Crew Member has upcoming shifts. Crew Member deletions do not impact past payroll or financial reports.
Open Issues:	Should the Crew Member receive an email notification when their account is deleted? Should Admins be allowed to restore deleted Crew Members within a certain timeframe? Should there be an automated check preventing deletion if the Crew Member has outstanding payroll or other obligations?

FrogCrew	Version: 3.1
Use Cases	Date: 4/10/2025
FrogCrew Use Cases	

Use Case 16: Admin views crew members

UC ID and Name:	UC-16: Admin views crew members		
Created By:	Michala Rogers	Date Created:	2/5/2025
Primary Actor:	Admin	Secondary Actors:	Crew Member
Trigger:	The Admin initiates the action to view the list of all active Crew Members.		
Description:	The Admin needs to access a list of all registered Crew Members in the system. This allows for quick reference, filtering, and management of crew assignments, ensuring efficient scheduling and coordination.		
Preconditions:	PRE-1. There exists at least one Crew Member registered in the System. PRE-2. The Admin is logged into the System.		
Postconditions:	POST-1: The system retrieves and displays the list of current Crew Members. POST-2: The Admin can apply filters or sorting options to organize the displayed list.		
Main Success Scenario:	<ol style="list-style-type: none"> 1. The Admin navigates to the "Manage Crew Members" section. 2. The Admin selects the option to view all active Crew Members. 3. The system retrieves the names and details of all Crew Members from the database. 4. The system displays the Crew Member list based on predefined sorting mechanisms. 5. The Admin can apply filters (e.g., by role, availability, experience, or name) to refine the results. 6. The Admin can click on a Crew Member's name to view detailed profile information. 7. The Admin completes the review process. 8. Use case ends. 		
Extensions:	<p>6a. No Crew Members found in the system:</p> <ul style="list-style-type: none"> • 6a1. The system displays a message: "No Crew Members registered in the system." • 6a2. The Admin is prompted to invite new Crew Members. <p>7a. The Admin wants to export the Crew Member list:</p> <ul style="list-style-type: none"> • 7a1. The system provides an option to download the list in CSV or Excel format. • 7b. The Admin wants to perform bulk actions (e.g., send messages, update roles, deactivate accounts): <ul style="list-style-type: none"> ○ 7b1. The system provides checkboxes next to each Crew Member's name for bulk selection. ○ 7b2. The Admin selects the desired action and confirms. 		
Priority:	Medium		
Frequency of Use:	Approximately 1 user, average of 1-2 usages per day during active scheduling periods.		
Business Rules:	BR-18: Admins can view, edit, or delete Crew Members' profiles.		
Associated Information:	<p>Sorting & Filtering Mechanisms:</p> <ul style="list-style-type: none"> • Sorting Criteria: <ul style="list-style-type: none"> ○ Crew Member Role ○ First Name ○ Last Name ○ Availability Status • Filtering Options: <ul style="list-style-type: none"> ○ Active vs. Inactive Crew Members ○ Availability for upcoming games ○ Experience Level (e.g., New, Experienced, Senior) ○ Assigned vs. Unassigned Crew 		

FrogCrew	Version: 3.1
Use Cases	Date: 4/10/2025
FrogCrew Use Cases	

Related Use Cases:	UC 1: Crew Member Creates Profile UC 2: Crew Member Edits Profile UC 3: Crew Member Views Other Crew Member's Profile UC 6: Crew Member Views Crew List UC 14: Admin Invites Crew Member UC 15: Admin Deletes Crew Member
Assumptions:	The Admin has the necessary permissions to view the full Crew Member list. Crew Members have already been registered in the system. The system supports real-time updates to the Crew Member list.
Open Issues:	Should the Admin have the ability to archive Crew Members instead of deleting them? Should the system allow multiple Admins to view and manage the list simultaneously? Should Admins be able to assign or remove Crew Members from schedules directly from this view?

FrogCrew	Version: 3.1
Use Cases	Date: 4/10/2025
FrogCrew Use Cases	

Use Case 17: Admin Views Crew Member’s Availability

UC ID and Name:	UC-17: Admin Views Crew Member’s Availability														
Created By:	James Clarke	Date Created:	10/01/24												
Primary Actor:	Administrator	Secondary Actors:	Crew members												
Trigger:	Administrator indicates they would like to access crew member availability.														
Description:	The admin wants to view the details of a crew member's availability, so that they can get a better idea of who is able to work specific games.														
Preconditions:	PRE-1. The admin is logged into the system with “admin privileges”. PRE-2: Crew members have successfully submitted their availability.														
Postconditions:	POST-1: The admin is displayed a list of available times for an individual crew member that they are available to work.														
Main Success Scenario:	<ol style="list-style-type: none"> 1. The admin indicates to view the details of a crew member’s availability. 2. The admin finds a list of crew members through <u>UC-13: Admin views crew members</u>. 3. The admin views the list and chooses to view the availability details of one specific crew member. 4. The System retrieves and displays details of this crew member’s availability according to the “Details” defined in the Associated Information and the “Security/access concerns” defined in the Business Rules of this use case. 5. The admin views the details of this crew member’s availability. 6. The admin can sort through the crew member’s availability information based on the “Sorting and Organization” defined in associated information of this use case. 7. Use case ends. 														
Extensions:	3a. No associated availability for a given crew member: 3a1. The System alerts the Administrator that the specified crew member availability is not available. 3a2. The system prompts the admin to return to the crew member viewing page [UC-12].														
Priority:	Medium														
Frequency of Use:	Approximately 1 user, average of 15 usage per semester.														
Business Rules:	BR-6: Crew Members must submit availability before being scheduled.														
Associated Information:	Details: <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th>Property name</th> <th>Data type</th> <th>Security/access concerns</th> <th>Reference to glossary</th> </tr> </thead> <tbody> <tr> <td>First name</td> <td>String</td> <td></td> <td></td> </tr> <tr> <td>Last name</td> <td>String</td> <td></td> <td></td> </tr> </tbody> </table> Sorting and Organization: <ul style="list-style-type: none"> • View is initially displayed in chronological order • Can be sorted based on the following: <ul style="list-style-type: none"> ○ Before/after [date] ○ Before/after [time] ○ Between [date, date] ○ Based on event [sports game] Notifications: <ul style="list-style-type: none"> • Admin is notified of invalid availability data. 			Property name	Data type	Security/access concerns	Reference to glossary	First name	String			Last name	String		
Property name	Data type	Security/access concerns	Reference to glossary												
First name	String														
Last name	String														
Related Use Cases:	UC 7: Crew Member Submits Availability UC 8: Crew Member Edits Availability UC 16: Admin Views Crew Members UC 22: Admin Schedules Crew														
Assumptions:	The administrator has successfully viewed current crew members, following the main success scenario from UC-12														

FrogCrew	Version: 3.1
Use Cases	Date: 4/10/2025
FrogCrew Use Cases	

Open Issues:	
--------------	--

Use Case 18: Admin Creates Game Schedule

UC ID and Name:	UC-18: Admin Creates Game Schedule		
Created By:	Michala Rogers	Date Created:	2/3/2025
Primary Actor:	Admin	Secondary Actors:	
Trigger:	The Admin initiates the process to create a new game schedule.		
Description:	The Admin wants to create a game schedule, assigning specific games, dates, venues, and other details necessary for organizing crew assignments. This schedule will be used for crew assignment, shift planning, and event coordination.		
Preconditions:	PRE-1: The Admin is logged into the system.		
Postconditions:	POST-1: A new game schedule is created in the system. POST-2: The schedule is saved as a draft and can be edited before publishing. POST-3: The schedule is accessible for future assignments and updates.		
Main Success Scenario:	<ol style="list-style-type: none"> 1. The Admin navigates to the "Create Game Schedule" section. 2. The system prompts the Admin to enter key schedule details 3. The Admin inputs the schedule details and submits them for validation. 4. The system verifies the input data for completeness and validity. 5. The system saves the schedule 6. Use case ends. 		
Extensions:	4a. Missing or invalid game details: <ul style="list-style-type: none"> ● 4a1. The system alerts the Admin of missing or incorrect fields. ● 4a2. The Admin corrects the issue and resubmits the schedule. 		
Priority:	High		
Frequency of Use:	Approximately 1 user, average of 5 usages per week (during scheduling periods).		
Business Rules:	BR-19: Only Admins with scheduling permissions can create a game schedule. BR-20: A game schedule must have at least one valid game entry before saving. BR-22: A schedule cannot be published until it is reviewed and finalized.		
Associated Information:	Schedule details must follow predefined formats for dates, times, and venues. Crew position requirements should align with event needs. Schedule Details <ul style="list-style-type: none"> ● Sport type ● Game date and time ● Venue ● Opponent (if applicable) ● Required crew positions 		
Related Use Cases:	UC 19: Admin Edits Crew Member's Profile UC 20: Admin Adds Games to Game Schedule UC 21: Admin Edits Game Schedule UC 22: Admin Schedules Crew		
Assumptions:	The system contains accurate game details and venue information. Admins have the necessary permissions to create schedules.		
Open Issues:	Should the system auto-suggest crew assignments based on previous schedules? Should the schedule support recurring events for ongoing sports seasons?		

FrogCrew	Version: 3.1
Use Cases	Date: 4/10/2025
FrogCrew Use Cases	

Use Case 19: Admin Edits Crew Member's Profile

UC ID and Name:	UC-19: Admin Edits Crew Member's Profile																										
Created By:	Dave Park	Date Created:	1/14/2025																								
Primary Actor:	Admin	Secondary Actors:	Crew Member																								
Trigger:	The Admin indicates to edit the details of a crew member's profile.																										
Description:	The Admin wants to edit a crew member's profile so that they can correct mistakes found in a profile.																										
Preconditions:	PRE-1. There exists at least one crew member's profile. PRE-2. The Admin is logged into the System. PRE-3. The User has the "admin" privilege.																										
Postconditions:	POST-1. The profile is updated in the System.																										
Main Success Scenario:	<ol style="list-style-type: none"> 1. The Admin indicates to edit the details of a crew member's profile. 2. The Spirit Director views the details of this profile through <u>UC-12: Admin views crew members</u>. 3. The Admin chooses to change the details of this profile. 4. The System asks the Admin to make changes to this profile where allowed according to the "Details" defined in the Associated Information. 5. The Admin makes changes to this profile until they confirm that they have finished changing. 6. The System validates the Admin's changes and alerts warning messages. 7. The Admin acknowledges the warnings and chooses to continue. 8. The System displays the updated details of this profile and alerts the Admin to confirm the change. 9. The Admin either confirms the change (continues the normal flow) or chooses to continue to change the details (returns to step 5). 10. The System saves the changes and informs the Admin that this request has been changed. 11. The System notifies relevant actors about the change of this request according to the "Notification" defined in the Associated Information of this use case. 12. Use case ends. 																										
Extensions:	6a. Input validation rule violation: 6a1. The System alerts the Admin that an input validation rule is violated and displays the nature and location of the error. 6a2. The Admin corrects the mistake and returns to step 6 of the normal flow.																										
Priority:	High																										
Frequency of Use:	Approximately 1 user, average of 3 usages per month.																										
Business Rules:	BR-18: Admins can view, edit, or delete Crew Members' profiles.																										
Associated Information:	Details: <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th>Property name</th> <th>Data type</th> <th>Editability</th> <th>Validation rule</th> </tr> </thead> <tbody> <tr> <td>Role</td> <td>String</td> <td>Yes</td> <td>Required</td> </tr> <tr> <td>First Name</td> <td>String</td> <td>Yes</td> <td>Required</td> </tr> <tr> <td>Last Name</td> <td>String</td> <td>Yes</td> <td>Required</td> </tr> <tr> <td>Email</td> <td>String</td> <td>Yes</td> <td>Required, valid email</td> </tr> <tr> <td>Phone number</td> <td>int</td> <td>Yes</td> <td>Required, (999) 999-9999 format</td> </tr> </tbody> </table>			Property name	Data type	Editability	Validation rule	Role	String	Yes	Required	First Name	String	Yes	Required	Last Name	String	Yes	Required	Email	String	Yes	Required, valid email	Phone number	int	Yes	Required, (999) 999-9999 format
Property name	Data type	Editability	Validation rule																								
Role	String	Yes	Required																								
First Name	String	Yes	Required																								
Last Name	String	Yes	Required																								
Email	String	Yes	Required, valid email																								
Phone number	int	Yes	Required, (999) 999-9999 format																								

FrogCrew	Version: 3.1
Use Cases	Date: 4/10/2025
FrogCrew Use Cases	

	<table border="1"> <tr> <td>Password</td> <td>String</td> <td>Yes</td> <td>Required</td> </tr> <tr> <td>Position</td> <td>List<String></td> <td>Yes</td> <td>Required</td> </tr> <tr> <td>Pay Rate</td> <td>int</td> <td>Yes</td> <td>Required</td> </tr> </table> <p>Notification:</p> <ul style="list-style-type: none"> The System notifies the Crew Member about the modification. The System notifies the Admin about the modification. <p>The Admin shall be able to cancel the use case at any time prior to submitting it.</p>	Password	String	Yes	Required	Position	List<String>	Yes	Required	Pay Rate	int	Yes	Required
Password	String	Yes	Required										
Position	List<String>	Yes	Required										
Pay Rate	int	Yes	Required										
Related Use Cases:	UC 1: Crew Member Creates Profile UC 2: Crew Member Edits Profile UC 16: Admin Views Crew Members												
Assumptions:													
Open Issues:													

Use Case 20: Admin Adds Games To Game Schedule

UC ID and Name:	UC-20: Admin Adds Games To Game Schedule		
Created By:	Michala Rogers	Date Created:	2/3/2025
Primary Actor:	Admin	Secondary Actors:	
Trigger:	The Admin initiates the action to add one or more games to an existing game schedule.		
Description:	The Admin wants to add new games to an existing game schedule, ensuring the schedule remains updated with the latest game events. This allows for ongoing adjustments, including scheduling new games, rescheduling postponed games, or filling gaps in the current season's schedule.		
Preconditions:	PRE-1: The Admin is logged into the system. PRE-2: A game schedule already exists in the system.		
Postconditions:	POST-1: The new game(s) are successfully added to the existing game schedule. POST-2: The schedule reflects the updated list of games. POST-3: The system saves changes and updates relevant crew assignment workflows.		
Main Success Scenario:	<ol style="list-style-type: none"> The Admin navigates to the "Manage Game Schedule" section. The Admin selects an existing game schedule to update. The system displays the current list of scheduled games. The Admin clicks "Add New Game" and enters the required details The system validates the new game details. The Admin confirms the addition of the game(s). The system saves the changes and updates the schedule. The system notifies relevant actors (e.g., crew members, admins) of the updated schedule. Use case ends. 		
Extensions:	<p>6a. Missing or invalid game details:</p> <ul style="list-style-type: none"> 6a1. The system alerts the Admin of missing or incorrect fields. 6a2. The Admin corrects the issue and resubmits. <p>8a. The Admin wants to save changes as a draft before finalizing:</p> <ul style="list-style-type: none"> 8a1. The system allows the Admin to save the updated schedule as a draft. 		
Priority:	High		
Frequency of Use:	Approximately 1 user, average of 5-10 usages per week (during scheduling periods).		

FrogCrew	Version: 3.1
Use Cases	Date: 4/10/2025
FrogCrew Use Cases	

Business Rules:	BR-21: New games cannot overlap with existing scheduled games unless explicitly allowed. BR-22: A schedule cannot be published until it is reviewed and finalized.
Associated Information:	Added games should follow system-defined formats for dates, times, and locations. Crew position requirements must align with event needs. Required Details: <ul style="list-style-type: none"> • Sport type • Game date and time • Venue • Opponent (if applicable) • Required crew positions
Related Use Cases:	UC 18: Admin Creates Game Schedule UC 21: Admin Edits Game Schedule UC 22: Admin Schedules Crew
Assumptions:	The system contains accurate game details and venue information. Admin has the game schedule to add. Admins have the necessary permissions to modify schedules.
Open Issues:	Should the system allow bulk additions of multiple games at once? Should the system auto-suggest potential game slots based on venue availability?

Use Case 21: Admin Edits The Schedule

UC ID and Name:	UC-21: Admin Can Edit The Schedule		
Created By:	Dave Park	Date Created:	1/14/2025
Primary Actor:	Admin	Secondary Actors:	
Trigger:	The Admin indicates to edit the details of a schedule.		
Description:	The Admin wants to edit a schedule so that they can correct mistakes found in a request.		
Preconditions:	PRE-1. There exists at least one completed schedule in the System. PRE-2. The Admin is logged into the System. PRE-3. The User has the “admin” privilege.		
Postconditions:	POST-1. The schedule is updated in the System.		
Main Success Scenario:	<ol style="list-style-type: none"> 1. The Admin indicates to edit the details of a schedule. 2. The Admin views the details of this schedule. 3. The Admin chooses to change the details of this schedule. 4. The System asks the Admin to make changes to this request where allowed according to the “Details” defined in the Associated Information. 5. The Admin makes changes to this request until she confirms that they have finished changing. 6. The System validates the Admin’s changes and alerts a warning message. 7. The Admin acknowledges the warning and chooses to continue. 8. The System displays the updated details of this request and alerts the Admin to confirm the change. 9. The Admin either confirms the change (continues the normal flow) or chooses to continue to change the details (returns to step 5). 10. The System saves the changes and informs the Admin that this request has been changed. 11. The System notifies relevant actors about the change of this request according to the “Notification” defined in the Associated Information of this use case. 		

FrogCrew	Version: 3.1
Use Cases	Date: 4/10/2025
FrogCrew Use Cases	

	12. Use case ends.																																
Extensions:	<p>6a. Input validation rule violation:</p> <p>6a1. The System alerts the Admin that an input validation rule is violated and displays the nature and location of the error.</p> <p>6a2. The Admin corrects the mistake and returns to step 6 of the normal flow.</p>																																
Priority:	High																																
Frequency of Use:	1 user, average of 3 usages per month.																																
Business Rules:	<p>BR-19: Only Admins with scheduling permissions can create a game schedule.</p> <p>BR-21: New games cannot overlap with existing scheduled games unless explicitly allowed.</p> <p>BR-22: A schedule cannot be published until it is reviewed and finalized.</p>																																
Associated Information:	<p>Details:</p> <table border="1"> <thead> <tr> <th>Property name</th> <th>Data type</th> <th>Editability</th> <th>Validation rule</th> </tr> </thead> <tbody> <tr> <td>Season</td> <td>DateTime</td> <td>Yes</td> <td>Required</td> </tr> <tr> <td>Sport</td> <td>String</td> <td>Yes</td> <td>Required</td> </tr> <tr> <td>Games</td> <td>List<Game></td> <td>Yes</td> <td>Required</td> </tr> <tr> <td>Game Start</td> <td>datetime</td> <td>Yes</td> <td>Required</td> </tr> <tr> <td>Venue</td> <td>String</td> <td>Yes</td> <td>Required</td> </tr> <tr> <td>Open Positions</td> <td>List<String></td> <td>Yes</td> <td>Required</td> </tr> <tr> <td>Crew Members</td> <td>List<Crewmember></td> <td>Yes</td> <td>Required</td> </tr> </tbody> </table> <p>Notification:</p> <ul style="list-style-type: none"> The System notifies the Admin about the modification. <p>The Admin shall be able to cancel the use case at any time prior to submitting it.</p>	Property name	Data type	Editability	Validation rule	Season	DateTime	Yes	Required	Sport	String	Yes	Required	Games	List<Game>	Yes	Required	Game Start	datetime	Yes	Required	Venue	String	Yes	Required	Open Positions	List<String>	Yes	Required	Crew Members	List<Crewmember>	Yes	Required
Property name	Data type	Editability	Validation rule																														
Season	DateTime	Yes	Required																														
Sport	String	Yes	Required																														
Games	List<Game>	Yes	Required																														
Game Start	datetime	Yes	Required																														
Venue	String	Yes	Required																														
Open Positions	List<String>	Yes	Required																														
Crew Members	List<Crewmember>	Yes	Required																														
Related Use Cases:	<p>UC 18: Admin Creates Game Schedule</p> <p>UC 20: Admin Adds Games to Game Schedule</p> <p>UC 22: Admin Schedules Crew</p>																																
Assumptions:																																	
Open Issues:																																	

FrogCrew	Version: 3.1
Use Cases	Date: 4/10/2025
FrogCrew Use Cases	

Use Case 22: Admin Edits Games In Game Schedule

UC ID and Name:	UC-22: Admin Edits Games In Game Schedule		
Created By:	Michala Rogers	Date Created:	2/3/2025
Primary Actor:	Admin	Secondary Actors:	
Trigger:	The Admin initiates an action to modify an existing game entry in the game schedule.		
Description:	The Admin needs to edit details of a scheduled game, such as changing the date, time, venue, opponent, or required crew positions. This ensures that the game schedule remains accurate and up to date for proper event planning and staffing.		
Preconditions:	PRE-1: The Admin is logged into the system. PRE-2: At least one game schedule exists in the system. PRE-3: The Admin has permission to edit game schedules.		
Postconditions:	POST-1: The edited game details are saved in the system. POST-2: The updated game schedule reflects the changes. POST-3: Relevant stakeholders (crew members, admins) are notified if the changes impact assignments.		
Main Success Scenario:	<ol style="list-style-type: none"> 1. The Admin logs into the system. 2. The Admin navigates to the "Manage Game Schedule" section. 3. The Admin selects an existing game from the schedule to edit. 4. The system displays the current game details. 5. The Admin updates the necessary details, such as: <ol style="list-style-type: none"> a. Sport type b. Game date and time c. Venue d. Opponent (if applicable) e. Required crew positions 6. The system validates the changes to ensure they comply with scheduling rules. 7. The Admin confirms the edits. 8. The system saves the updated game details. 9. The system notifies affected users (crew members, admins) if applicable. 10. Use case ends. 		
Extensions:	6a. Invalid or conflicting game details: <ul style="list-style-type: none"> • 6a1. The system alerts the Admin of any errors (e.g., double booking, invalid venue). • 6a2. The Admin corrects the issue and resubmits the changes. 8a. The Admin wants to save changes as a draft before finalizing: <ul style="list-style-type: none"> • 8a1. The system allows the Admin to save the changes as a draft. 		
Priority:	High		
Frequency of Use:	Approximately 1 user, average of 5-7 usages per week (during scheduling periods).		
Business Rules:	BR-23: Admins can only assign Crew Members to roles they are qualified for. BR-24: Admins cannot assign Crew Members to overlapping shifts.		
Associated Information:	Game details must follow system-defined formats for dates, times, and venues. Crew assignments should be updated accordingly if a game's date/time is changed.		
Related Use Cases:	UC 4: Crew Member Views Their Scheduled Games UC 5: Crew Member Views General Game Schedule UC 7: Crew Member Submits Availability UC 8: Crew Member Edits Availability UC 9: Crew Member Requests Shift Exchange UC 10: Crew Member Accepts/Denies Shift Exchange UC 11: Admin Approves Shift Exchange		

FrogCrew	Version: 3.1
Use Cases	Date: 4/10/2025
FrogCrew Use Cases	

	UC 17: Admin Views Crew Member's Availability UC 18: Admin Creates Game Schedule UC 19: Admin Edits Crew Member's Profile UC 20: Admin Adds Games to Game Schedule UC 21: Admin Edits Game Schedule UC 23: Admin Publishes Game Schedule
Assumptions:	The system contains accurate data on games, venues, and assigned crew members. Admins have the authority to modify schedules without additional approval.
Open Issues:	Should certain changes (e.g., date/time adjustments) require re-confirmation from assigned crew members? Should the system log a history of all schedule edits for tracking purposes?

Use Case 23: Admin Schedules Crew

UC ID and Name:	UC-23: Admin Schedules Crew		
Created By:	Michala Rogers	Date Created:	2/3/2025
Primary Actor:	Admin	Secondary Actors:	Crew Member
Trigger:	The Admin initiates the process to assign crew members to scheduled games.		
Description:	The Admin assigns crew members to available roles in a game schedule. This ensures that each game has the required staff based on availability, qualifications, and predefined roles. Once scheduled, crew members receive notifications confirming their assignments.		
Preconditions:	PRE-1: The Admin is logged into the system. PRE-2: A game schedule exists with games requiring crew assignments. PRE-3: Crew members have submitted availability for the scheduled games. PRE-4: The system contains up-to-date records of crew members and their qualified roles.		
Postconditions:	POST-1: Crew members are successfully assigned to scheduled games. POST-2: The system saves the updated crew schedule. POST-3: Crew members receive notifications regarding their assigned games.		
Main Success Scenario:	<ol style="list-style-type: none"> 1. The Admin navigates to the "Schedule Crew" section. 2. The Admin selects a game from the list of scheduled events. 3. The system displays a list of available crew positions for the selected game. 4. The Admin assigns crew members to each required position based on availability and qualifications. 5. The system validates the assignments to ensure: <ol style="list-style-type: none"> a. Crew members meet role qualifications. b. There are no scheduling conflicts. 6. The Admin confirms the crew assignments. 7. The system updates the game schedule with assigned crew members. 8. The system sends notifications to scheduled crew members. 9. Use case ends. 		
Extensions:	6a. Crew member is unavailable or has scheduling conflict: <ul style="list-style-type: none"> • 6a1. The system alerts the Admin about conflicts. • 6a2. The Admin selects an alternative crew member. 7a. The Admin wants to save progress but not finalize assignments: <ul style="list-style-type: none"> • 7a1. The system allows the Admin to save the schedule as a draft. 		
Priority:	High		
Frequency of Use:	Approximately 1 user, average of 10-15 usages per week (during scheduling periods).		
Business Rules:	BR-22: A schedule cannot be published until it is reviewed and finalized.		
Associated	Crew roles must align with predefined event requirements.		

FrogCrew	Version: 3.1
Use Cases	Date: 4/10/2025
FrogCrew Use Cases	

Information:	The system should allow batch scheduling for multiple games at once.
Related Use Cases:	UC 4: Crew Member Views Their Scheduled Games UC 5: Crew Member Views General Game Schedule UC 6: Crew Member Views Crew List UC 22: Admin Schedules Crew
Assumptions:	The system maintains an updated list of available crew members. Admins have full visibility into crew availability before scheduling.
Open Issues:	Should the system suggest optimal crew assignments based on previous schedules? Should there be an automatic backup system in case a crew member cancels last-minute?

Use Case 24: Admin Publishes Game Schedule

UC ID and Name:	UC-24: Admin Publishes Game Schedule		
Created By:	Dave Park	Date Created:	09/27/24
Primary Actor:	Administrator	Secondary Actors:	Crew members
Trigger:	The admin successfully starts the creation of a sports schedule.		
Description:	After a new sports schedule has been created, the administrator wants to finalize the sports schedule and then publish information regarding what events crew members have been scheduled for, what position they hold at these events, and the event type.		
Preconditions:	PRE-1. The admin is logged into the system with “admin privileges”. PRE-2. The admin builds the schedule for a given sport/event.		
Postconditions:	POST-1. Each crew member included in the schedule receives information regarding their booked dates and events.		
Main Success Scenario:	<ol style="list-style-type: none"> 1. The admin navigates to the schedule management section. 2. The admin views a list of draft game schedules. 3. The admin selects the draft game schedule that needs to be finalized. 4. The system displays the details of the selected draft game schedule. 5. The admin reviews the schedule to ensure all details are correct. 6. The admin validates the information on the schedule. 7. The system finalizes the schedule and notifies relevant users. 8. The admin selects “Publish” to release the schedule by sport. 9. The system prompts the admin to confirm publication. 10. The admin confirms to publish the schedule. 11. The system grabs the contact information from the user accounts attached to the schedule. 12. The system sends relevant information pertaining to their scheduled events/games to relevant actors via email. 13. The system notifies relevant actors about the publication of the new schedule via text. 14. Use case ends. 		

FrogCrew	Version: 3.1
Use Cases	Date: 4/10/2025
FrogCrew Use Cases	

Extensions:	<p>5a. Invalid Inputs</p> <p>5a1. The system displays an error message</p> <p>5a2. The admin is prompted to correct the input before proceeding.</p> <p>5a3. The admin inputs valid values and proceeds.</p> <p>5a4. The system proceeds to step 6.</p> <p>11a. Input validation rule violation:</p> <p>11a1. The System alerts the admin that the schedule has not been completed correctly, due to input fields with no value.</p> <p>11a2. The admin updates empty fields and continues normal flow to step 3.</p> <p>14a. The System cannot find relevant contact information from user accounts:</p> <p>14a1. The System alerts the admin that there is no contact information connected to accounts in the crew list.</p> <p>14a2. The System displays a list of users with missing contact information.</p> <p>14a3. The admin updates the missing contact information fields and clicks “Update”, then continues normal flow to step 16.</p>
Priority:	High
Frequency of Use:	Approximately 1 user, average of 5 usage per semester.
Business Rules:	BR-31: Crew list template names must be unique.
Associated Information:	<p>Notifications:</p> <ul style="list-style-type: none"> ● Admin notified to confirm sending of schedule ● Admin is notified of successful generation and send of schedule ● Admin is notified is there was an issue locating user contact data <p>Event types - This entails the sport that the game schedule is associated with</p> <ul style="list-style-type: none"> ● Men’s Football ● Women’s Basketball ● Men’s Basketball ● Women's Soccer ● Men’s Baseball ● Women’s Volleyball <p>Schedule Details</p> <ul style="list-style-type: none"> ● Sport type ● Game date and time ● Venue ● Opponent (if applicable) ● Required crew positions
Related Use Cases	<p>UC 16: Admin Views Crew Members</p> <p>UC 22: Admin Schedules Crew</p> <p>UC 23: Admin Publishes Game Schedule</p>
Assumptions:	
Open Issues:	

FrogCrew	Version: 3.1
Use Cases	Date: 4/10/2025
FrogCrew Use Cases	

Use Case 25: Admin Generates Crew List To Send

UC ID and Name:	UC-25: Admin Generates Crew List To Send		
Created By:	James Clarke	Date Created:	09/27/24
Primary Actor:	Administrator	Secondary Actors:	Crew members
Trigger:	The admin successfully finalizes a new crew list and indicates to send the generated crew list to relevant actors.		
Description:	After a new crew list has been published, the administrator wants to send information regarding included crew members, meeting times and location, so that the production crew is informed of their new booking and have the necessary information for the upcoming sports production.		
Preconditions:	PRE-1. The admin is logged into the system with “admin privileges”. PRE-2. The admin schedules a crew to work a specific game. PRE-3. The admin finalizes and publishes the crew list for the game. PRE-4. Finalized crew list is displayed.		
Postconditions:	POST-1. Each crew member included in the crew list receives the new crew list, along with all necessary information regarding the event.		
Main Success Scenario:	<ol style="list-style-type: none"> 1. The admin verifies and finalizes a new crew list for a sports event. 2. The admin selects “Publish” on the new crew list to indicate it is ready to be sent to the relevant actors. 3. The system generates a new crew list, updating meeting times based on crew position and location based on sports events. 4. The system displays a confirmation message to verify the admin wishes to send the crew list information to included crew members. 5. The admin confirms the system to send the information to the relevant actors (continues the normal flow) or chooses to “Publish later” to modify the crew list further (return to step 2). 6. The system grabs the contact information from the user accounts attached to the crew list. 7. The system sends a copy of the crew list information to each of the crew members included in the crew list via email. 8. The system notifies relevant actors about the publication of the new crew list via text. 9. Use case ends. 		
Extensions:	<ol style="list-style-type: none"> 2a. Input validation rule violation: <ol style="list-style-type: none"> 2a1. The System alerts the admin that the crew list has not been completed correctly, due to input fields with no value. 2a2. The admin updates empty fields and continues normal flow to step 3. 6a. The System cannot find relevant contact information from user accounts: <ol style="list-style-type: none"> 6a1. The System alerts the admin that there is no contact information connected to accounts in the crew list. 6a2. The System displays a list of users with missing contact information. 6a3. The admin updates the missing contact information fields and clicks “Update”, then continues normal flow to step 7. 		
Priority:	High		
Frequency of Use:	Approximately 1 user, average of 45 usage per semester.		
Business Rules:	BR-25: Pay rate data must be stored in Crew Member profiles for financial reporting. BR-26: Financial reports must be available for both individual games and entire seasons.		
Associated Information:	Notifications: <ul style="list-style-type: none"> • Admin notified to confirm sending of crew list 		

FrogCrew	Version: 3.1
Use Cases	Date: 4/10/2025
FrogCrew Use Cases	

	<ul style="list-style-type: none"> • Admin is notified of successful generation and send of crew list • Admin is notified is there was an issue locating user contact data
Related Use Cases	UC 26: Admin Generates Position Report UC 27: Admin Generates Individual Crew Member Report
Assumptions:	Crew list has already been built, and is ready to be generated in the correct format.
Open Issues:	This use case may need to be broken down/merged with other use cases. It seems the sending mechanism isn't the focus of this use case, or the use case should be re-worked to provide a clearer explanation of its intended action. [discuss with team]

FrogCrew	Version: 3.1
Use Cases	Date: 4/10/2025
FrogCrew Use Cases	

Use Case 26: Admin Generates Financial Report

UC ID and Name:	UC-26: Admin Generates Financial Report		
Created By:	Michala Rogers	Date Created:	9/29/24
Primary Actor:	Admin	Secondary Actors:	None
Trigger:	The Admin decides to generate a financial report for one or more games to review crew expenditures.		
Description:	The system allows the Admin to generate a report showing how much money has been spent on crew members for specific games or a season. The report includes details about each crew member's name, the number of games they worked, the positions they filled, and their corresponding pay rates. The report can be organized by game or by the entire season, showing the total cost of staffing for each event and for the entire season. The system pulls data from individual crew profiles where their pay rate is stored.		
Preconditions:	PRE-1: The Admin must be logged into the system. PRE-2: Crew members must have worked a game and their profiles must contain the pay rates associated with their roles.		
Postconditions:	POST-1: The financial report is generated and available for download as an Excel or CSV file. POST-2: The Admin can see detailed financial data for each crew member and game.		
Main Success Scenario:	<ol style="list-style-type: none"> 1. The Admin logs into the system. 2. The Admin navigates to the "Generate Financial Report" option. 3. The system prompts the Admin to select games or a date range. 4. The Admin selects the desired games or range and confirms. 5. The system compiles the financial data for all crew members involved in those games, using pay rates stored in their profiles. 6. The system generates a detailed financial report showing total spending per game, crew member, and position. 7. The Admin downloads the report. 8. Use case ends. 		
Extensions:	4a. No pay rate data for a crew member: 4a1. The system alerts the Admin that a crew member is missing pay rate data. 4a2. The Admin can choose to manually enter the pay rate or exclude the crew member from the report.		
Priority:	Medium		
Frequency of Use:	Weekly during game seasons		
Business Rules:	BR-27: Historical Crew Member assignments must remain available for financial records even after a Crew Member is deleted.		
Associated Information:	<p>Report generating parameters:</p> <ul style="list-style-type: none"> ● Report title: Financial Report ● Data source: The System ● Data selection criteria: The Admin selects the game(s) or date range. ● Report visualization (table, charts, graph): Summary table displayed in the system. ● Sorting criteria: By game, crew member, or position. ● Pagination criteria: 10 items per page (if applicable). ● Disposition of the report after it is generated: Available for download as an Excel or CSV file. ● Format of the generated report: Excel or CSV 		

FrogCrew	Version: 3.1
Use Cases	Date: 4/10/2025
FrogCrew Use Cases	

	<p>Report generating algorithm:</p> <ul style="list-style-type: none"> The system retrieves financial data for all crew members based on the selected games or date range. It calculates total pay per crew member and organizes the report by game, position, and crew member details. <p>Report content:</p> <ul style="list-style-type: none"> Crew Member Name Position(s) worked Games worked (with dates) Pay rate Total pay
Related Use Cases:	<p>UC 16: Admin Views Crew Members</p> <p>UC 22: Admin Schedules Crew</p> <p>UC 25: Admin Generates Financial Report</p>
Assumptions:	<ol style="list-style-type: none"> The system has access to accurate pay rate information for all crew members. Games worked and positions filled with data are already recorded in the system. The Admin is aware of how to select the correct date range and games to generate the report.
Open Issues:	

FrogCrew	Version: 3.1
Use Cases	Date: 4/10/2025
FrogCrew Use Cases	

Use Case 27: Admin Generates Position Report

UC ID and Name:	UC-27: Admin Generates Position Report		
Created By:	Michala Rogers	Date Created:	9/30/24
Primary Actor:	Admin	Secondary Actors:	
Trigger:	The Admin needs to generate a report detailing the crew members who worked specific positions during games.		
Description:	The system allows the Admin to generate a report that shows all crew members who have worked a specific position (e.g., Director, Producer, Camera Operator) during a specified game or range of games. This report helps the Admin track who filled key roles for each game and their performance in those positions.		
Preconditions:	PRE-1: The Admin must be logged into the system. PRE-2: Crew members must have been scheduled for and worked the specified games.		
Postconditions:	POST-1: The position report is generated and available for download as an Excel or CSV file. POST-2: The Admin can see which crew members worked in specific positions across different games.		
Main Success Scenario:	<ol style="list-style-type: none"> 1) The Admin logs into the system. 2) The Admin navigates to the "Generate Position Report" option. 3) The system prompts the Admin to select the position and games or date range. 4) The Admin selects the position and date range and confirms. 5) The system compiles the data for all crew members who worked in that position. 6) The system generates a report showing crew members, positions, and games worked. 7) The Admin downloads the report. 8) Use case ends. 		
Extensions:	4a. No crew member data for the position: The system alerts the Admin that no crew members were scheduled for the selected position and suggests choosing a different position or time range.		
Priority:	Medium		
Frequency of Use:	As needed, particularly after large events or end of season		
Business Rules:	BR-27: Historical Crew Member assignments must remain available for financial records even after a Crew Member is deleted.		
Associated Information:	<p>Report generating parameters:</p> <ul style="list-style-type: none"> ● Report title: Position Report ● Data source: The System ● Data selection criteria: The Admin selects the position and game(s) or date range. ● Report visualization (table, charts, graph): Summary table displayed in the system. ● Sorting criteria: Chronological order of games worked. ● Pagination criteria: 10 items per page (if applicable). ● Disposition of the report after it is generated: Available for download as an Excel or CSV file. ● Format of the generated report: Excel or CSV <p>Report generating algorithm:</p> <ul style="list-style-type: none"> ● The system retrieves data for all crew members who worked in the specified position(s) during the selected game(s) or date range. The report is generated as an Excel or CSV file summarizing this data. <p>Report content:</p>		

FrogCrew	Version: 3.1
Use Cases	Date: 4/10/2025
FrogCrew Use Cases	

	<ul style="list-style-type: none"> • Crew Member Name • Position worked • Games worked (with dates)
Related Use Cases:	UC 16: Admin Views Crew Members UC 22: Admin Schedules Crew UC 25: Admin Generates Financial Report
Assumptions:	<ol style="list-style-type: none"> 1. The system has access to all position data for crew members. 2. The Admin can filter reports by both date ranges and specific positions. 3. The report will be generated based on up-to-date data of crew members' roles in specific games.
Open Issues:	

FrogCrew	Version: 3.1
Use Cases	Date: 4/10/2025
FrogCrew Use Cases	

Use Case 28: Admin Generates Individual Crew Member Report

UC ID and Name:	UC-28: Admin Generates Individual Crew Member Report		
Created By:	Michala Rogers	Date Created:	10/1/24
Primary Actor:	Admin	Secondary Actors:	Crew Member (for review)
Trigger:	The Admin wants to generate a report on an individual crew member to review their work history, positions filled, and total pay.		
Description:	The system allows the Admin to generate a report for a specific crew member, detailing their work history (games worked), positions filled, and total pay earned. The Admin can use this report to review crew performance and compensation for auditing or review purposes.		
Preconditions:	PRE-1: The Admin must be logged into the system. PRE-2: The crew member must have worked at least one game.		
Postconditions:	POST-1: The individual report is generated and available for download. POST-2: The Admin has a clear view of the crew member's work history and pay.		
Main Success Scenario:	<ol style="list-style-type: none"> 1. The Admin logs into the system. 2. The Admin navigates to the "Generate Individual Crew Member Report" option. 3. The system prompts the Admin to select a crew member. 4. The Admin selects the crew member and confirms. 5. The system compiles data on games worked, positions filled, and total pay. 6. The system generates a report summarizing the crew member's performance and earnings. 7. The Admin downloads the report. 8. Use case ends. 		
Extensions:	4a. No work history for crew member: The system alerts the Admin that the crew member has not worked any games and suggests selecting a different crew member.		
Priority:	Low		
Frequency of Use:	As needed for performance reviews or audits		
Business Rules:	BR-31: Crew list template names must be unique.		
Associated Information:	<p>Report generating parameters:</p> <ul style="list-style-type: none"> ● Report title: Individual Crew Member Report ● Data source: The System ● Data selection criteria: The Admin selects the crew member from the list of registered crew members. ● Report visualization (table, charts, graph): Summary table displayed in the system. ● Sorting criteria: Chronological order of games worked. ● Pagination criteria: 10 items per page (if applicable). ● Disposition of the report after it is generated: Available for download as a PDF. ● Format of the generated report: PDF <p>Report generating algorithm:</p> <ul style="list-style-type: none"> ● The system retrieves data from the selected crew member's profile, including games worked, positions filled, and total pay. The report is generated as a PDF summarizing this data. <p>Report content (for one crew member):</p> <ul style="list-style-type: none"> ● Name ● Positions filled ● Games worked (with dates) ● Total pay earned 		

FrogCrew	Version: 3.1
Use Cases	Date: 4/10/2025
FrogCrew Use Cases	

Related Use Cases:	UC 22: Admin Schedules Crew UC 23: Admin Publishes Game Schedule
Assumptions:	<ol style="list-style-type: none"> 1) The system contains accurate records of the crew member's work history, including games and positions. 2) The Admin has the ability to filter by crew member and generate a comprehensive report. 3) The crew member's profile and work data are complete and up to date in the system.
Open Issues:	

FrogCrew	Version: 3.1
Use Cases	Date: 4/10/2025
FrogCrew Use Cases	

Use Case 29: Admin Creates a Position

UC ID and Name:	UC-29: Create a position		
Created By:	Michala Rogers	Date Created:	4/10/2025
Primary Actor:	Admin	Secondary Actors:	
Trigger:	The Admin indicates to create a new position.		
Description:	The Admin creates a new position within the system. Positions are used in game scheduling and crew list templates. Each position must be clearly named and categorized to maintain consistency and enable proper assignment.		
Preconditions:	PRE-1: The Admin is logged into the system. PRE-2: The Admin has access to Position Management features.		
Postconditions:	POST-1: The new position is added to the system and is available to be used in crew list templates and game scheduling.		
Main Success Scenario:	<ol style="list-style-type: none"> 1. The Admin navigates to the “Position Management” section. 2. The Admin clicks “Add New Position.” 3. The System prompts the Admin to enter position details. 4. The Admin inputs the position name, category, and optional description. 5. The System validates that the position name is unique and all required fields are filled. 6. The Admin confirms creation of the position. 7. The System saves the new position and displays a success message. 8. The position is now available in the system for use. 9. Use case ends. 		
Extensions:	<p>5a. Input validation fails:</p> <ol style="list-style-type: none"> 5a1. The System identifies missing or invalid data and displays error messages. 5a2. The Admin corrects the inputs and proceeds to step 6. <p>5b. Duplicate position name detected</p> <ol style="list-style-type: none"> 5b1. The System alerts the Admin that the position name already exists. 5b2. The Admin updates the name and retries submission. 		
Priority:	Medium		
Frequency of Use:	1 user, 1–2 times per semester		
Business Rules:	BR-31: Crew list template names must be unique. BR-32: A Crew list template must include at least one position. BR-34: All input fields must pass validation before submission. BR-35: Unique identifiers must be validated against existing records.		
Associated Information:	Position Name	String	Yes Required, must be unique
	Category	String	Yes Required
	Description	String	Yes Optional
Related Use Cases:	UC-30: Admin Edits a Position UC-35: Admin Edits a Crew List Template		
Assumptions:	Admin understands the organizational structure of crew roles and naming conventions.		
Open Issues:	Should positions be grouped by event type or production team function?		

FrogCrew	Version: 3.1
Use Cases	Date: 4/10/2025
FrogCrew Use Cases	

Use Case 30: Admin Edits a Position

UC ID and Name:	UC-30: Edit a position		
Created By:	Michala Rogers	Date Created:	4/10/2025
Primary Actor:	Admin	Secondary Actors:	
Trigger:	The Admin indicates to change the details of an existing position.		
Description:	The Admin edits a position's name, category, or description to correct errors, update naming conventions, or improve clarity in crew assignments.		
Preconditions:	PRE-1: The Admin is logged into the System. PRE-2: A position exists in the System.		
Postconditions:	POST-1: The selected position is updated with the new information and saved in the system.		
Main Success Scenario:	<ol style="list-style-type: none"> 1. The Admin indicates to change the details of an existing position. 2. The System asks the Admin to make changes to this position where allowed according to the "Details" defined in the Associated Information. 3. The Admin makes changes to this position until she confirms that they have finished changing. 4. The System validates the Admin's changes and alerts warning messages according to the "Details" defined in the Associated Information of this use case. 5. The Admin acknowledges the warnings and chooses to continue. 6. The System displays the updated details of this position and alerts the Admin to confirm the change. 7. The Admin either confirms the change (continues the normal flow) or chooses to continue to change the details (return to step 5). 8. The System saves the changes, carries out the effect of change according to the "Details" defined in the Associated Information of this use case, and informs the Admin that this senior design section has been changed. 9. Use case ends. 		
Extensions:	4a. Input validation rule violation: <ol style="list-style-type: none"> 4a1. The System alerts the Admin that an input validation rule is violated and displays the nature and location of the error. 4a2. The Admin corrects the mistake and returns to step 6 of the normal flow. 		
Priority:	Medium		
Frequency of Use:	1 user, 1-3 times per semester		
Business Rules:	BR-34: All input fields must pass validation before submission. BR-35: Unique identifiers must be validated against existing records.		
Associated Information:	Property Name	Data Type	Editability Validation Rule
	Position Name	String	Yes Required, must be unique
	Category	String	Yes Required
	Description	String	Yes Optional
Related Use Cases:	UC-29: Admin Creates a Position		

FrogCrew	Version: 3.1
Use Cases	Date: 4/10/2025
FrogCrew Use Cases	

	UC-36: Admin Deletes a Crew List Template
Assumptions:	Admin understands how edits to a position may affect crew list templates and scheduling.
Open Issues:	

Use Case 31: Admin Finds Positions

UC ID and Name:	UC-31: View Positions		
Created By:	Michala Rogers	Date Created:	4/10/2025
Primary Actor:	Admin	Secondary Actors:	
Trigger:	The Admin indicates to find positions		
Description:	The Admin searches for a crew position using filters such as name or category. This helps verify existing entries or prepare for edits and assignments.		
Preconditions:	PRE-1: The Admin is logged into the system. PRE-2: At least one position exists in the system.		
Postconditions:	POST-1: The Admin is presented with a filtered or full list of positions. POST-2: The Admin may choose to view, edit, or delete positions from the results		
Main Success Scenario:	<ol style="list-style-type: none"> 1. The Admin navigates to the "Position Management" section. 2. The Admin selects the "Search" or "View All" function. 3. The System prompts the Admin to enter filters (e.g., name, category). 4. The Admin enters one or more search criteria and submits. 5. The System retrieves matching results. 6. The System displays the results in a sortable table. 7. The Admin views or selects a position from the results. 8. Use case ends. 		
Extensions:	5a. No results found: 5a1. The System displays a message: "No positions match your criteria." 5a2. The Admin may clear filters or enter different search terms.		
Priority:	Medium		
Frequency of Use:	1 user, as needed		
Business Rules:	BR-35: Unique identifiers must be validated against existing records.		
Associated Information:			
Related Use Cases:	UC-30: Admin Edits a Position UC-36: Admin Deletes a Crew List Template		
Assumptions:	Admin knows which positions or categories they are looking for.		
Open Issues:	Should deleted positions remain searchable as inactive/archive entries		

FrogCrew	Version: 3.1
Use Cases	Date: 4/10/2025
FrogCrew Use Cases	

Use Case 32: Admin Creates a Crew List Template

UC ID and Name:	UC-32: Create a crew list template		
Created By:	Michala Rogers	Date Created:	4/10/2025
Primary Actor:	Admin	Secondary Actors:	
Trigger:	The Admin indicates to create a new crew list template.		
Description:	The Admin builds a reusable template containing predefined positions for a specific sport or event type. These templates streamline the process of scheduling crews for future games.		
Preconditions:	PRE-1: The Admin is logged into the system. PRE-2: At least one crew position exists in the system.		
Postconditions:	POST-1: A new crew list template is saved in the system. POST-2: The template is available for use in future games.		
Main Success Scenario:	<ol style="list-style-type: none"> 1. The Admin navigates to the “Crew List Template Management” section. 2. The Admin selects “Create New Template.” 3. The System prompts the Admin to enter a template name and select positions. 4. The Admin enters a unique template name. 5. The Admin selects one or more crew positions from the list. 6. The System validates that the name is unique and at least one position is selected. 7. The Admin confirms the creation of the template. 8. The System saves the template and displays a confirmation message. 9. Use case ends. 		
Extensions:	6a. Template name already exists: 6a1. The System displays an error message: “Template name must be unique.” 6a2. The Admin edits the name and retries. 6b. No positions selected: 6b1. The System displays an error message: “At least one position must be selected.” 6b2. The Admin adds a position and continues.		
Priority:	Medium		
Frequency of Use:	1 user, 5 usage per year		
Business Rules:	BR-31: Crew list template names must be unique. BR-32: A Crew list template must include at least one position. BR-34: All input fields must pass validation before submission.		
Associated Information:	Property Name	Data Type	Editability Validation Rule
	Template Name	String	Yes Required, must be unique
	Positions	List	Yes Required, must include at least one
Related Use Cases:	UC-33: Admin Finds Crew List Templates UC-35: Admin Edits a Crew List Template		
Assumptions:	Admin understands which positions are standard for the template’s intended use.		
Open Issues:			

FrogCrew	Version: 3.1
Use Cases	Date: 4/10/2025
FrogCrew Use Cases	

Use Case 33: Admin Finds Crew List Templates

UC ID and Name:	UC-33: Find crew list templates		
Created By:	Michala Rogers	Date Created:	4/10/2025
Primary Actor:	Admin	Secondary Actors:	
Trigger:	The Admin indicates to find crew list templates.		
Description:	The Admin searches for existing templates to view, edit, or apply them to game schedules. This enables reuse of standard crew configurations and ensures consistency across events.		
Preconditions:	PRE-1: The Admin is logged into the system. PRE-2: At least one crew list template exists.		
Postconditions:	POST-1: The Admin is presented with a list of templates based on the applied filters. POST-2: The Admin may take further actions such as view, edit, or delete.		
Main Success Scenario:	<ol style="list-style-type: none"> 1. The Admin navigates to the “Crew List Template Management” section. 2. The Admin selects the option to “Find Templates.” 3. The System prompts the Admin to enter search criteria (e.g., template name, sport). 4. The Admin enters filters and submits the query. 5. The System retrieves and displays matching crew list templates. 6. The Admin selects a template to view details or perform other actions. 7. Use case ends. 		
Extensions:	5a. No templates match the search criteria: 5a1. The System displays: “No templates found.” 5a2. The Admin may clear filters or adjust search terms.		
Priority:	Medium		
Frequency of Use:	1 user, 5 usage per year		
Business Rules:	BR-35: Unique identifiers must be validated against existing records.		
Associated Information:	Template Name	String	Yes Partial or full name match
	Created Date	Date	Yes Used to filter by creation date
	Associated Sport	String	Yes Used to filter by sport category
Related Use Cases:	UC-32: Admin Creates a Crew List Template UC-34: Admin Views a Crew List Template UC-36: Admin Deletes a Crew List Template		
Assumptions:	Admin understands the naming conventions or tags used in templates.		
Open Issues:			

FrogCrew	Version: 3.1
Use Cases	Date: 4/10/2025
FrogCrew Use Cases	

Use Case 34: Admin Views a Crew List Template

UC ID and Name:	UC-34: View a crew list template		
Created By:	Michala Rogers	Date Created:	4/10/2025
Primary Actor:	Admin	Secondary Actors:	
Trigger:	The Admin indicates to view the details of a crew list template		
Description:	The Admin views the structure of an existing crew list template, including the positions it contains. This allows verification before assigning the template to a game or editing it.		
Preconditions:	PRE-1: The Admin is logged into the system. PRE-2: At least one crew list template exists.		
Postconditions:	POST-1: The selected template's details are displayed to the Admin. POST-2: The Admin may choose to edit or delete the template after viewing.		
Main Success Scenario:	<ol style="list-style-type: none"> 1. The Admin navigates to the "Crew List Template Management" section. 2. The Admin searches for and selects a crew list template. 3. The System retrieves the template details 4. The System displays the template name, creation date, and list of included positions 5. The Admin reviews the template details 6. Use case ends. 		
Extensions:	2a. No templates found: 2a1. The System displays: "No crew list templates available." 2a2. The Admin may choose to create a new template.		
Priority:	Medium		
Frequency of Use:	1 user, many times a week		
Business Rules:	BR-35: Unique identifiers must be validated against existing records.		
Associated Information:	Field Name	Data Type	Description
	Template Name	String	Unique name of the template
	Created Date	Date	Date the template was created
	Included Positions	List	All positions assigned to this template
Related Use Cases:	UC-33: Admin Finds Crew List Templates UC-35: Admin Edits a Crew List Template UC-36: Admin Deletes a Crew List Template		
Assumptions:	Admin has permission to view templates and understands position structu		
Open Issues:			

FrogCrew	Version: 3.1
Use Cases	Date: 4/10/2025
FrogCrew Use Cases	

Use Case 35: Admin Edits a Crew List Template

UC ID and Name:	UC-35: Edit a crew list template																		
Created By:	James Edmonson	Date Created:	2/7/2025																
Primary Actor:	Admin	Secondary Actors:																	
Trigger:	The Admin indicates to change the details of an existing crew list template.																		
Description:	The Admin wants to change the crew list template, to edit the required positions for a template.																		
Preconditions:	PRE-1: The Admin is logged into the System. PRE-2: A crew list template exists.																		
Postconditions:	POST-1: Changes made to the crew list template are stored in the System.																		
Main Success Scenario:	<ol style="list-style-type: none"> 1. The Admin indicates to change the details of an existing crew list template. 2. The Admin views the details of this crew list template through UC-34: View a crew list template. 3. The Admin chooses to change the details of this crew list template. 4. The System asks the Admin to make changes to this crew list template where allowed according to the “Details” defined in the Associated Information. 5. The Admin makes changes to this crew list template until she confirms that she has finished changing. 6. The System validates the Admin’s changes and alerts warning messages according to the “Details” defined in the Associated Information of this use case. 7. The Admin acknowledges the warnings and chooses to continue. 8. The System displays the updated details of this crew list template and alerts the Admin to confirm the change. 9. The Admin either confirms the change (continues the normal flow) or chooses to continue to change the details (return to step 5). 10. The System saves the changes, carries out the effect of change according to the “Details” defined in the Associated Information of this use case. 11. Use case ends. 																		
Extensions:	6a. Input validation rule violation: 6a1. The System alerts the Admin that an input validation rule is violated and displays the nature and location of the error. 6a2. The Admin corrects the mistake and returns to step 8 of the normal flow.																		
Priority:	Medium																		
Frequency of Use:	1 user, 1 usage per year.																		
Business Rules:																			
Associated Information:	<table border="1"> <thead> <tr> <th>Property name</th> <th>Data type</th> <th>Editability</th> <th>Security/access concerns</th> <th>Reference to glossary</th> </tr> </thead> <tbody> <tr> <td>Template name</td> <td></td> <td></td> <td></td> <td></td> </tr> <tr> <td>Template positions</td> <td></td> <td></td> <td></td> <td></td> </tr> </tbody> </table> <p>No two templates can have the same name. The template name must be unique.</p> <p>The Admin shall be able to cancel the use case at any time prior to submitting it.</p>				Property name	Data type	Editability	Security/access concerns	Reference to glossary	Template name					Template positions				
Property name	Data type	Editability	Security/access concerns	Reference to glossary															
Template name																			
Template positions																			
Related Use Cases:																			
Assumptions:																			
Open Issues:																			

FrogCrew	Version: 3.1
Use Cases	Date: 4/10/2025
FrogCrew Use Cases	

Use Case 36: Admin Deletes a Crew List Template

UC ID and Name:	UC-36: Delete a crew list template		
Created By:	Michala Rogers	Date Created:	4/10/2025
Primary Actor:	Admin	Secondary Actors:	
Trigger:	The Admin indicates to delete a crew list template		
Description:	The Admin removes a crew list template from the system. This is used to eliminate outdated or unused templates, keeping the system organized and relevant.		
Preconditions:	PRE-1: The Admin is logged into the system. PRE-2: At least one crew list template exists.		
Postconditions:	POST-1: The selected template is permanently deleted and no longer appears in search results. POST-2: The deletion is confirmed to the Admin.		
Main Success Scenario:	<ol style="list-style-type: none"> 1. The Admin navigates to the “Crew List Template Management” section. 2. The Admin searches for and selects a crew list template. 3. The System displays template details. 4. The Admin clicks the “Delete” option. 5. The System prompts the Admin to confirm deletion. 6. The Admin confirms. 7. The System deletes the template and displays a success message. 8. Use case ends. 		
Extensions:	<p>5a. Admin cancels deletion:</p> <ol style="list-style-type: none"> 5a1. The System closes the confirmation prompt. 5a2. The Admin is returned to the template view page. <p>6a. Template is already deleted by another Admin:</p> <ol style="list-style-type: none"> 6a1. The System displays an error: “Template no longer exists.” 6a2. The Admin is redirected to the template list. 		
Priority:	Low		
Frequency of Use:	Rare. 1 user, 1 usage per year.		
Business Rules:	BR-33: Deleted Crew list templates cannot be recovered. BR-34: All input fields must pass validation before submission.		
Associated Information:	<p>Deletion strategy:</p> <ul style="list-style-type: none"> • Crew list template deletion is a physical delete. In other words, this will permanently remove the crew list template from the database (cannot be recovered). <p>The Admin shall be able to cancel the use case at any time prior to submitting it.</p>		
Related Use Cases:	UC-33: Admin Finds Crew List Templates UC-34: Admin Views a Crew List Template		
Assumptions:	The Admin is authorized to delete templates and understands that the action is permanent.		
Open Issues:			

FrogCrew	Version: 3.1
Use Cases	Date: 4/10/2025
FrogCrew Use Cases	