FrogCrew

Vision

Version 1.3

Revision History

| **Date** | **Version** | **Description** | **Author** |
| --- | --- | --- | --- |
| 23/09/24 | 1.0 | Initial draft | James Edmonson, Aliya Suri, Kate Bednarz, Dave Park, Manuel Burciaga, James Clarke |
| 25/10/24 | 1.1 | Fixed comments and improved descriptions | James Edmonson, Aliya Suri, Kate Bednarz, Dave Park, Manuel Burciaga |
| 27/10/24 | 1.2 | Revised and completed all sections | Aliya Suri |
| 14/1/25 | 1.3 | Made corrections from feedback | Aliya Suri |

Table of Contents

[1. Introduction](#_30j0zll) **4**

[1.1 Background](#_1fob9te) 4

[1.2 References](#_3znysh7) 4

[2. Business Requirements](#_66h73mupbtch) **5**

[2.1 Business Opportunity/Problem Statement](#_tyjcwt) 5

[2.2 Business Objectives](#_1t3h5sf) 5

[2.3 Success Metrics](#_4d34og8) 5

[2.4 Vision Statement](#_2s8eyo1) 6

[2.5 Business Risks](#_17dp8vu) 6

[2.6 Business Assumptions and Dependencies](#_3rdcrjn) 6

[3. Stakeholder Profiles and User Descriptions](#_lnxbz9) **8**

[3.1 Stakeholder Profiles](#_35nkun2) 8

[3.2 User Environment](#_1ksv4uv) 9

[3.3 Alternatives and Competition](#_2jxsxqh) 9

[4. Scope and Limitations](#_3j2qqm3) **10**

[4.1 Product Perspective](#_1y810tw) 10

[4.2 Major Features / Scope](#_4i7ojhp) 10

[4.3 Deployment Considerations](#_2xcytpi) 11

[5. Other Product Requirements](#_1ci93xb) **12**

Vision

# Introduction

*[The purpose of this document is to collect, analyze, and define the business requirements, i.e., high-level needs, desired ultimate business outcomes and features of the* <<System Name>>*. It focuses on the capabilities needed by the stakeholders and the target users, and* ***why*** *these needs exist in the first place. The details of how the* <<System Name>> *fulfills these needs are detailed in the use-case and supplementary specifications.]*

*[The introduction of the* ***Vision*** *document provides an overview of the entire document. It includes the purpose and references of this* ***Vision*** *document.]*

The purpose of this document is to collect, analyze, and define the business requirements, i.e., high-level needs, desired ultimate business outcomes and features of the TCU Frog Crew System. It focuses on the capabilities needed by the stakeholders and the target users, and why these needs exist in the first place. The details of how the TCU Frog Crew System fulfills these needs are detailed in the use case and software requirements specification.

## Background

*[Summarize the rationale and context for the new product or for changes to be made to an existing one. Describe the history or situation that led to the decision to build this product.]*

*Example: Employees at the company Process Impact presently spend an average of 65 minutes per day going to the cafeteria to select, purchase, and eat lunch. About 20 minutes of this time is spent walking to and from the cafeteria, selecting their meals, and paying by cash or with credit card. When employees go out for lunch, they spend an average of 90 minutes off-site. Some employees phone the cafeteria in advance to order a meal to be ready for them to pick up. Employees don’t always get the selections they want because the cafeteria runs out of certain items. The cafeteria wastes a significant quantity of food that is not purchased and must be thrown away. These same issues apply to breakfast and supper, although far fewer employees use the cafeteria for those meals than for lunch.*

Mike Martin is an Associate Professor of Professional Practice at Texas Christian University, responsible for managing crew lists for athletic events, some of which are partnered with ESPN+, depending on the sport. His responsibilities include coordinating crew schedules for Men’s Football, Women’s Basketball, Men’s Basketball, Women’s Soccer, Women’s Volleyball, and Men’s Baseball.

The current workflow begins with crew members submitting their availability via email or text (see **Figure 1** for a visual representation of the existing process). Mike manually compiles this information into a Comprehensive Crew List, which serves as a master document containing crew member details, such as contact information and the roles they are qualified to perform (e.g., Producer, Director, Camera Operator, Technical Director, EVS Operator, and Audio Engineer). From this, he creates additional documents to manage and communicate the necessary information for each event.

One key document is the Crew Availability Spreadsheet, which lists crew members' availability for a specific event. This spreadsheet provides Mike with the basis for assigning roles, ensuring the crew is staffed appropriately for the event’s needs. Another document Mike generates is the Game Day Schedule, a detailed schedule shared with crew members to inform them of upcoming game times, locations, and general logistical information.

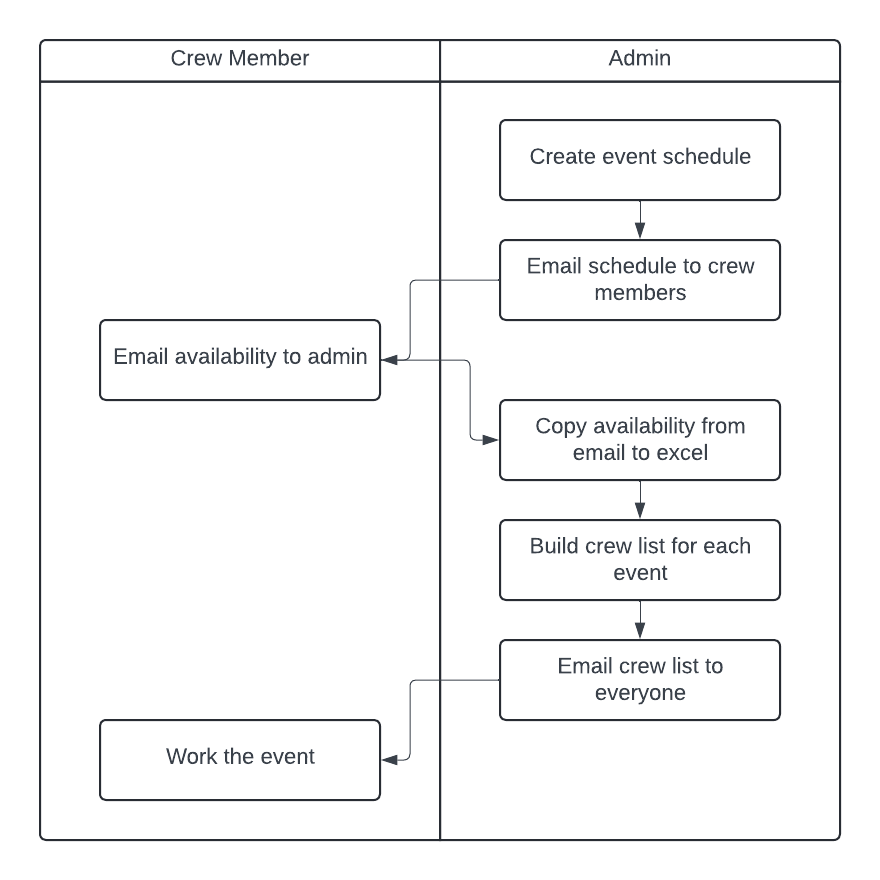
Using these resources, Mike creates a Game Day Crew List for each event, specifying the assignments for each crew member based on availability and the event’s specific requirements. The Game Day Crew List includes:

* Assigned roles for each crew member (e.g., Camera 1, GFX, EVS, or Technical Director).
* Contact details for immediate communication.
* Notes for role-specific tasks or responsibilities, such as handling specialized equipment like EVS or managing graphics overlays.

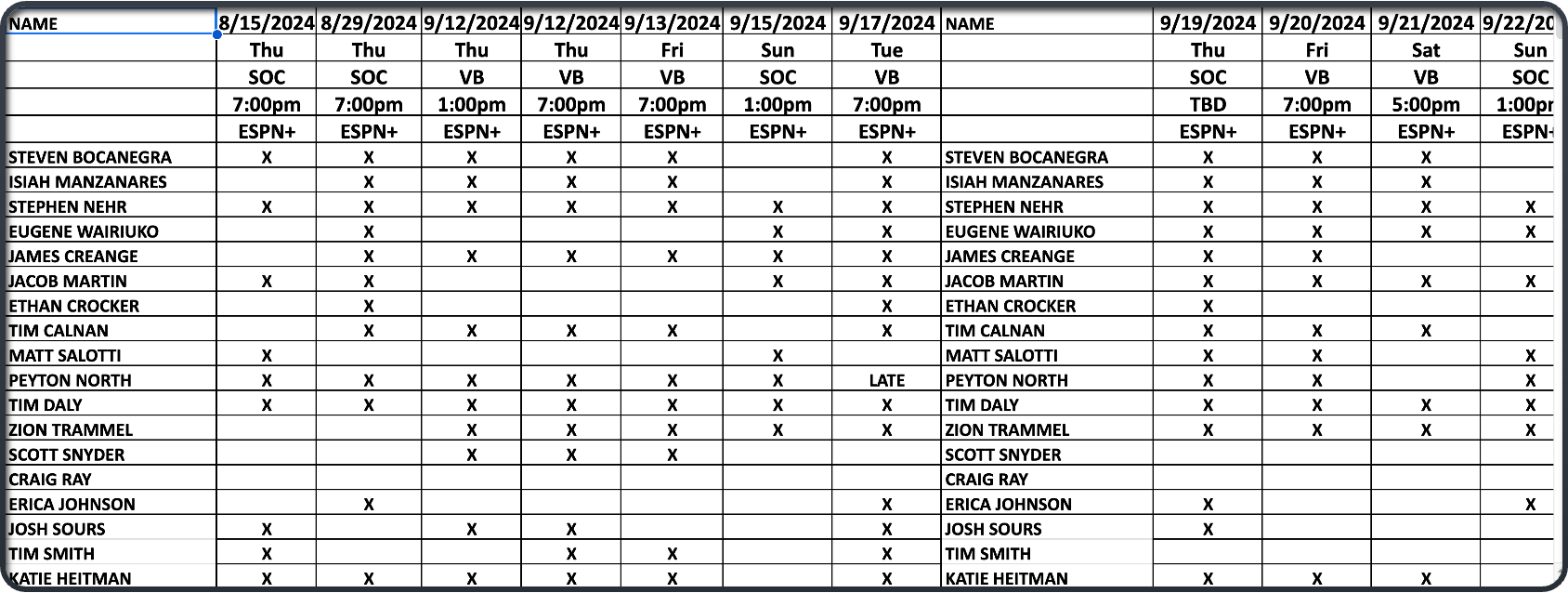
Once the schedule is finalized, Mike sends individual confirmation emails or texts to crew members with their assignments. This manual process requires significant time and effort, as it involves consolidating availability, scheduling roles, and communicating assignments across multiple channels (see **Figure 2** for an illustration of the inefficiencies in the current system). Additionally:

* Errors in data entry can result in scheduling conflicts or missed assignments.
* Important communications are at risk of being lost in personal email inboxes or text threads.
* Cost calculations, which depend on crew hours and roles, are managed separately and add further complexity to the workflow.

The new FrogCrew system is designed to streamline these tasks. It will allow crew members to input their availability directly into the system, automate the generation of schedules and assignments, and send notifications through a centralized platform. The system will also feature a built-in cost calculator, simplifying financial management for each event. By automating data entry, centralizing communication, and reducing the risk of errors, FrogCrew will save significant time and effort while ensuring a more reliable scheduling process for TCU’s athletic events.



*Figure 1: Flow of the current FrogCrew process.*



*Figure 2: Current form of crew member availability.*

## References

*[This subsection provides a complete list of all documents referenced elsewhere in the* ***Vision*** *document. Identify each document by title, report number if applicable, date, and publishing organization. Specify the sources from which the references can be obtained. This information may be provided by reference to an appendix or to another document.]*

[1 Project Glossary](https://docs.google.com/document/d/1T0NsAEpn77NMQb7ukbIDJjpBE6mgKREWUbEzZuvA2Nw/edit?usp=sharing)

# Business Requirements

*[Projects are launched in the belief that creating or changing a product will provide worthwhile benefits for someone and a suitable return on investment. The business requirements describe the primary benefits that the new system will provide to its sponsors, buyers, and users. Input to the business requirements should come from people who have a clear sense of* ***why they are undertaking the project****. These individuals might include the customer or development organization’s senior management, a product visionary, a product manager, a subject matter expert, or members of the marketing department. Business requirements directly influence which user requirements to implement and in what sequence. So, take it seriously!]*

## Business Opportunity/Problem Statement

*[Provide a statement summarizing the opportunity being exploited or problem being solved by this project. The following format may be used:]*

*[Or you can provide a paragraph like this:]*

*Example: Many employees have requested a system that would permit a cafeteria user to order meals (defined as a set of one or more food items selected from the cafeteria menu) online, to be picked up at the cafeteria or delivered to a company location at a specified time and date. Such a system would save employees time, and it would increase the chance of them getting the items they prefer. Knowing what food items customers want in advance would reduce wastage in the cafeteria and would improve the efficiency of cafeteria staff. The future ability for employees to order meals for delivery from local restaurants would make a wide range of choices available to employees and provide the possibility of cost savings through volume discount agreements with the restaurants.*

The problem of manual scheduling for TCU athletic events often leads to mistakehttp://localhost:5173/s, such as assigning crew members to the wrong roles or shifts or losing track of availability emails. These errors can result in confusion, miscommunication, and scheduling conflicts. Currently, Mike Martin, the sole person responsible for managing the schedules, spends hours collecting crew availability from emails and texts, manually inputting the information into a comprehensive crew list and generating additional documents, such as crew availability spreadsheets and game day schedules. These spreadsheets are then used to create game day crew lists, which assign specific roles (e.g., camera operator, EVS operator, technical director) to each crew member for upcoming events. The manual nature of this workflow is not only time-consuming but also prone to errors, such as incorrect assignments or missing details.

By implementing an automated scheduler that allows crew members to input their availability directly into the system, the entire process could be streamlined. The system would generate schedules and assignments automatically, eliminating the need for manual data entry, reducing errors, saving time, and improving communication. This would ensure that every crew member receives the correct assignment and remains informed, without the risks and inefficiencies of lost emails or manual data handling.

## Business Objectives

*[Summarize the important business benefits the product will provide in a quantitative and measurable way. Platitudes (“become recognized as a world-class <whatever>”) and vaguely stated improvements (“provide a more rewarding customer experience”) are neither helpful nor verifiable.]*

BO-1: Reduce the amount of time spent on creating schedules for TCU athletic events by 90%, automating the currently manual process.

BO-2**:** Reduce the number of emails in the administrator’s inbox by 50%, minimizing the risk of important communications being lost and improving overall communication efficiency.

BO-3: Help the administrator see data corresponding to financials, positions, and specific crew members to help

TCU Sports Broadcasting stay on budget and for crew members to be crewed accurately and fairly.

## Success Metrics

*[Specify the indicators that stakeholders will use to define and measure success on this project.]*

SM-1: 80% of crew members involved in TCU athletic events submit their availability through the FrogCrew system within the first 3 months following initial release, reducing the need for manual input via email or text.

SM-2: The time spent by Mike Martin on scheduling and communication for each athletic event is reduced by 90% within 6 months of deploying the FrogCrew system, compared to the time spent using the previous manual method.

SM-3: 95% of the crew schedules generated by the FrogCrew system are free from scheduling conflicts or errors, as reported by crew members, within 3 months of system deployment.

SM-4: The number of emails or text messages sent by Mike Martin to confirm crew member assignments decreases by 75% within 3 months of implementing the FrogCrew system, with notifications now being centralized within the platform.

SM-5: The cost calculation process for each event is reduced by 70% in time spent by Mike Martin, with the system’s built-in cost calculator streamlining financial management, compared to the manual process.

SM-6: 90% of crew members rate the FrogCrew system’s ease of use at 4 or higher on a 5-point scale in a user satisfaction survey conducted 6 months after the initial release.

## Vision Statement

*[Provide an overall statement summarizing, at the highest level, the unique position the product intends to fill in the marketplace. The following format may be used:]*

| For | TCU Sports Broadcasting administrators and crew members |
| --- | --- |
| Who | Submit work availability, schedule broadcasting events, generate reports |
| The SuperFrog Scheduler | Is an online web-application |
| That | Simplifies operations for administrators and crew members when scheduling sports production events |
| Unlike | The current manual process of submitting availability and scheduling work |
| Our product | Will automate and streamline the entire scheduling process and allow for easy scheduling, crew list generation, and reporting, ensuring smooth operation for all athletic events covered by TCU Sports Broadcasting. |

*[A product vision/position statement communicates the intent of the application and the importance of the project to all concerned personnel.]*

*Example: For employees who want to order meals from the company cafeteria or from local restaurants on-line, the Cafeteria Ordering System is an Internet-based and smart phone–enabled application that will accept individual or group meal orders, process payments, and trigger delivery of the prepared meals to a designated location on the Process Impact campus. Unlike the current telephone and manual ordering processes, employees who use the Cafeteria Ordering System will not have to go to the cafeteria to get their meals, which will save them time and will increase the food choices available to them.*

The Frog Crew Scheduling System aims to streamline and automate the scheduling process for TCU Sports Broadcasting. By eliminating the manual handling of crew availability and reducing errors caused by scattered communication, this system will simplify operations for administrators and crew members alike. The system will allow for easy scheduling, crew list generation, and reporting, ensuring smooth operation for all athletic events covered by TCU Sports Broadcasting.

## Business Risks

*[Summarize the major business risks associated with developing—or not developing—this product. Risk categories include marketplace competition, timing issues, user acceptance, implementation issues, and possible negative impacts on the business. Business risks are not the same as project risks, which often include resource availability concerns and technology factors. Estimate the potential loss from each risk, the likelihood of it occurring, and any potential mitigation actions.]*

*Example:*

*RI-1: The Cafeteria Employees Union might require that their contract be renegotiated to reflect the new employee roles and cafeteria hours of operation. (Probability = 0.6; Impact = 3)*

*RI-2: Too few employees might use the system, reducing the return on investment from the system development and the changes in cafeteria operating procedures. (Probability = 0.3; Impact = 9)*

*RI-3: Local restaurants might not agree to offer delivery, which would reduce employee satisfaction with the system and possibly their usage of it. (Probability = 0.3; Impact = 3)*

*RI-4: Insufficient delivery capacity might not be available, which means that employees would not always receive their meals on time and could not always request delivery for the desired times. (Probability = 0.5; Impact = 6).*

RI-1: If users input incorrect availability or scheduling data, the system could still produce scheduling errors, undermining its reliability and effectiveness.

RI-2: Employees might be resistant to using the new system, reducing the return on investment and negating the time-saving benefits of automation.

RI-3: The system may struggle to accommodate shift patterns or last-minute changes, limiting its usefulness and requiring manual intervention.

RI-4: The system may not seamlessly integrate with TCU IT’s existing network and infrastructure, leading to issues with deployment and longevity.

## Business Assumptions and Dependencies

*[An assumption is a statement that is believed to be true in the absence of proof or definitive knowledge. For example, an assumption may state that a specific operating system will be available for the hardware designated for the software product. If the operating system is not available, the* ***Vision*** *document will need to change. Record any assumptions that the stakeholders made. Record any major dependencies the project has on external factors. Examples are pending industry standards or government regulations, deliverables from other projects, third-party suppliers, or development partners. Note the potential impact of an assumption not being true, or the impact of a broken dependency, to help stakeholders understand why it is critical.]*

*Example:*

*AS-1: Systems with appropriate user interfaces will be available for cafeteria employees to process the expected volume of meals ordered.*

*AS-2: Cafeteria staff and vehicles will be available to deliver all meals for specified delivery time slots within 15 minutes of the requested delivery time.*

*DE-1: If a restaurant has its own on-line ordering system, the Cafeteria Ordering System must be able to communicate with it bi-directionally.*

AS-1: Event schedule data is provided in a consistent and usable form

AS-2: Usability of scheduler will be easier for both user and admin.

AS-3: A profile system (Name, Email, Phone Number, etc) will be available for users to update personal information

# Stakeholder Profiles and User Descriptions

*[To effectively provide products and services that meet your stakeholders’ and users’ real needs it is necessary to identify and involve all of the stakeholders as part of the Requirements Modeling process. You must also identify the users of the system and ensure that the stakeholder community adequately represents them. This section provides a profile of the stakeholders and users involved in the project, and the key problems that they perceive to be addressed by the proposed solution. It does not describe their specific requests or requirements as these are captured in a separate stakeholder requests artifact. Instead, it provides the background and justification for why the requirements are needed.]*

## Stakeholder Profiles

*[There are a number of stakeholders with an interest in the development and some of them are end users. Present a summary list of these stakeholders.]*

*[The major value or benefit that the stakeholder will receive from the product. Stakeholder value could be defined in terms of*

* *Improved productivity*
* *Reduced rework and waste*
* *Cost savings*
* *Streamlined business processes*
* *Automation of previous manual tasks*
* *Ability to perform entirely new tasks*
* *Compliance with pertinent standards or regulations*
* *Improved usability compared to current products*

| **Stakeholder** | **Major value or benefit from this product** | **Attitudes** | **Major features of interest** | **Constraints** | **End user or not?** |
| --- | --- | --- | --- | --- | --- |
| Admin  (Mike Martin) | Improved productivity; time saving; reduce scheduling errors | Strong commitment through the whole process | Time saving, ease of communication with crew members | None identified | Yes |
| Crew Members | More efficient use of viewing schedules, providing availability, and viewing crew lists | End users | Ease of communication with admin, ease of viewing schedules | Training for staff in using the software | Yes |
| Athletic Department | More efficient crew for broadcasting events | Won’t be a user but will appreciate the use of a better scheduling software | Scheduling crew feature | Getting logo approval from TCU athletics | No |
| TCU IT Department | Hosting the software | Periodic meetings | Following TCU IT’s software rules | Backend: C#  Database: MS SQL  Front end: Wordpress | No |

## User Environment

*[Detail the working environment of the target user. Here are some suggestions:*

*Number of people involved in completing the task? Is this changing?*

*How long is a task cycle? Amount of time spent in each activity? Is this changing?*

*Any unique environmental constraints: mobile, outdoors, in-flight, and so on?*

*Which system platforms are in use today? Future platforms?*

*What other applications are in use? Does your application need to integrate with them?*

*This is where extracts from the Business Model could be included to outline the task and roles involved, and so on.]*

The working environment will continue to involve the same number of people, primarily the scheduler and the team being scheduled. Currently, task cycles are short, with minimal time spent on each activity, but our goal is to further reduce this time through automation. There are no unique environmental constraints, such as mobility or outdoor requirements, to consider. Presently, there are no dedicated system platforms in use, and no plans to introduce new ones. The only application being used is Excel, and we aim to ensure that our solution integrates seamlessly with Excel for easy data import and export.

## Alternatives and Competition

*[Identify alternatives the stakeholder perceives as available. These can include buying a competitor’s product, building a homegrown solution, or simply maintaining the status quo. List any known competitive choices that exist or may become available. Include the major strengths and weaknesses of each competitor as perceived by the stakeholder or end user.]*

Some alternative scheduling tools include options like Calendly and When2Work. Both of these tools offer a variety of features tailored to different scheduling needs but come with notable costs. Calendly’s pricing is structured as follows: Standard at $10 per seat/month, Teams at $16 per seat/month, and Enterprise starting at $15,000 per year. Calendly is widely known for its intuitive and user-friendly interface, allowing users to set up personalized meeting types and availability windows. It integrates seamlessly with multiple calendars like Google Calendar, Outlook, and Office 365, enabling automatic syncing and reducing the chances of double-booking. Furthermore, Calendly offers advanced features like automated reminders, custom branding for premium plans, and team scheduling for collaborative bookings. However, users of the free version face limitations, such as restricted meeting types and fewer integrations, with more robust features only available at higher pricing tiers.

On the other hand, When2Work caters more specifically to industries that rely on shift-based scheduling. Pricing for When2Work varies based on the number of employees, with tiers such as $38 for 1 month, $100 for 3 months, $164 for 6 months, and $273 for a full year for up to 10 employees. When2Work’s primary strength lies in its ability to manage complex work shifts, track employee availability, facilitate shift swaps, and process time-off requests. It is particularly useful for businesses in healthcare, retail, and hospitality, where managing varying schedules and ensuring adequate staffing is critical. When2Work also includes mobile access, allowing employees to view their schedules and make adjustments on the go. However, it may be more complex to navigate compared to simpler scheduling tools like Calendly, and its focus on shift scheduling may not make it the ideal choice for businesses or individuals seeking appointment booking or general meeting coordination.

While both tools offer essential scheduling features, the right choice depends largely on the user’s specific needs, with Calendly excelling in appointment scheduling and When2Work standing out in managing shift-based work environments. Additionally, the cost structure of each may affect the overall decision, as Calendly’s fees can escalate with larger teams, while When2Work's pricing is more geared toward smaller workgroups but may still be costly for small businesses or startups.

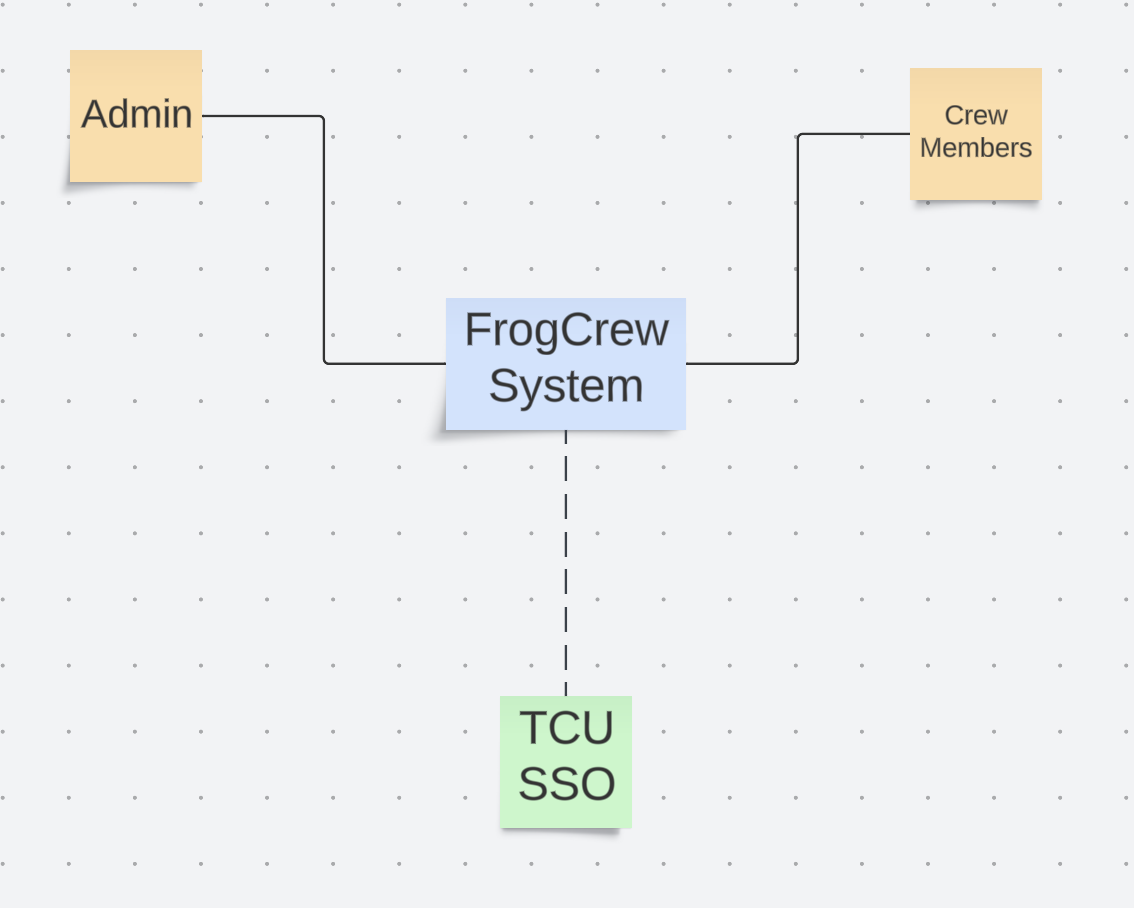
# Scope and Limitations

*[This section provides a high-level view of the product capabilities, interfaces to other applications, and system configurations.]*

This section provides a high-level view of the TCU Frog Crew scheduling system and system configuration.

## Product Perspective

*[This subsection of the* ***Vision*** *document puts the product in perspective to other related systems and the user’s environment. If the product is independent and totally self-contained, state it here. If the product is a component of a larger system, then this subsection needs to relate how these systems interact and needs to identify the relevant interfaces between the systems. One easy way to display the major components of the larger system, interconnections, and external interfaces is with a context diagram.]*

**

The context diagram above provides an overview of the FrogCrew System and its primary user roles: Admin and Crew Members. The system is designed to manage and facilitate crew-related activities, with Admin responsible for overseeing crew management, scheduling, and communication, while Crew Members can create and update profiles, view schedules, and manage their availability. The diagram also includes the TCU Single Sign-On (SSO) as a potential component, represented with a dashed line to indicate that integration with TCU's SSO system is not yet confirmed. If implemented, TCU SSO would provide a streamlined and secure authentication method for users accessing the FrogCrew System. This layout focuses on core functionalities and highlights the main interactions between user roles and the system.

## Major Features / Scope

*[List and briefly describe the product features. Features are the high-level* ***capabilities*** *of the system that are necessary to deliver benefits to the users. Each feature is an externally desired service that typically requires a series of inputs to achieve the desired result. For example, a feature of a problem tracking system might be the ability to provide trending reports. As the use-case model takes shape, update the description to refer to the use cases.*

*Because the* ***Vision*** *document is reviewed by a wide variety of involved personnel, the level of detail needs to be general enough for everyone to understand. However, enough detail must be available to provide the team with the information they need to create a use-case model (In other words, use cases in SRS* ***are derived from those features****).*

*To effectively manage application complexity, we recommend for any new system, or an increment to an existing system, capabilities be abstracted to a high enough level so 25-99 features result. These features provide the fundamental basis for product definition, scope management, and project management. Each feature will be expanded in greater detail in the use-case model.*

*Throughout this section, each feature will be externally perceivable by users, operators, or other interacting external systems. These features should include a description of functionality and any relevant usability issues that must be addressed. The following guidelines apply:*

*• Avoid design. Keep feature descriptions at a general level. Focus on capabilities needed and why (not how) they should be implemented.*

*[Define the priority of the different system features. Include, if useful, attributes such as stability, benefit, effort, and risk.]*

*How do we partition the system?*

*Tips: based on use cases: one subsystem can accomplish multiple logically related use cases (e.g. manager set promotion, manager delete promotion, manager publish a promotion, manager search a promotion can be grouped into a subsystem call Promotion Management) Use cases should not be written here in the vision document. Each use case needs many functions (the system shall save…, the system shall verify fields, the system shall delete, search, reset etc). One function can be used in many use cases (functional requirements will be written in SRS, not here).*

*Here is an example of Customer Relation Management system and its major subsystems:*

**FE-1**: Creates schedules based on game and season requirements.

**FE-2**: Allows communication with crew members through notifications within the system.

**FE-3**: Enables Admin to review, edit, and finalize schedules.

**FE-4**: Provides options for Admin to assign crew positions, addressing any open positions for each game.

**FE-5**: Exports crew lists and schedules for reporting or external use.

**FE-6**: Tracks crew pay rates and allows Admin to generate financial and position reports for budgeting and allocation purposes.

**FE-7**: Enables crew members to submit and edit their availability for games.

**FE-8**: Facilitates user registration and authentication, including profile creation and login.

## Deployment Considerations

*[Summarize the information and activities that are needed to ensure an effective deployment of the*

*solution into its operating environment. Describe the access that users will require to use the system,*

*such as whether the users are distributed over multiple time zones or located close to each other.*

*State when the users in various locations need to access the system. If infrastructure changes are*

*needed to support the software’s need for capacity, network access, data storage, or data migration,*

*describe those changes. Record any information that will be needed by people who will be preparing*

*training or modifying business processes in conjunction with deployment of the new solution.]*

To ensure the effective deployment of the solution into its operating environment, we will need to work closely with TCU IT, as they control the server that will host the project. Since this is a project for TCU, the IT department will oversee and manage the infrastructure required to support the deployment.

The primary requirement from TCU IT involves adhering to specific frameworks for both the front-end and back-end of the system. In this case, we have selected Vue.js for the front-end framework, which will provide a responsive and dynamic user interface, and WordPress to manage content easily. For the back-end, we are utilizing C# along with MS SQL for database management, ensuring compatibility with the university's existing infrastructure.

As for user access, only those who are directly invited by Mike will have access to the site. This will be controlled through an email-based sign-in system, where users will create a password after receiving a link to the site. This ensures that access is limited and secure. There are no other specific access control requirements beyond the invitation process, so the system will be restricted to those with the link.

Given the nature of the project, there are no immediate needs for changes in infrastructure such as additional capacity, network access, or data storage. The existing server capacity at TCU should be sufficient for the expected load. However, if the user base grows significantly in the future, additional infrastructure considerations may be required, such as increased storage or capacity to handle additional traffic.

As part of the deployment process, training will be required for users on how to access and navigate the system. Since the users will be located in specific, predetermined locations and are only granted access by Mike, the deployment will not require widespread user training. However, it is crucial for those preparing the training materials to ensure clear instructions on how users can sign up and log in to the system. Additionally, any necessary modifications to business processes (such as the invitation process or content management) will be incorporated during the deployment phase.

Overall, deployment will focus on collaboration with TCU IT to ensure that the system is securely hosted and accessible only to authorized users. By working with TCU’s infrastructure, we aim to streamline the deployment process and ensure that the system operates smoothly once live.

# Other Product Requirements

*[At a high level, list applicable standards, legal, hardware, or platform requirements; performance requirements; and environmental requirements.*

*Define the quality ranges for performance, robustness, fault tolerance, usability, and similar characteristics that are not captured in the Feature Set.*

*Note any design constraints, external constraints, or other dependencies.*

*Define any specific documentation requirements, including user manuals, online help, installation, labeling, and packaging requirements.*

*Define the priority of these other product requirements. Include, if useful, attributes such as stability, benefit, effort, and risk.]*

## Performance Requirements

The system should have a response time of less than 5 seconds for user requests under normal load conditions.

* 1. **Quality**

The system should maintain functionality under varying loads, with error handling to manage unexpected inputs or failures. The system should be able to recover gracefully from errors, ensuring minimal disruption to user experience

* 1. **Design/External Constraints**

The FrogCrew system must integrate seamlessly with existing TCU IT systems, including email and calendar applications. The system should allow for easy import and export of data from Excel and other common formats.

* 1. **Documentation Requirements**

An online help resource must be available to assist users with common issues and questions.

* 1. **Priority**

High priority should be assigned to compliance and performance requirements to ensure the system meets legal obligations and user expectations.