

# Problem Statement and Goals

## Software Engineering

Team 6, Pitch Perfect

Damien Cheung

Jad Haytaoglu

Derek Li

Temituoyo Ugborogho

Emma Wigglesworth

Table 1: Revision History

Date	Developer(s)	Change
Sept. 24, 2024	Jad Haytaoglu	Initial creation

## 1 Problem Statement

The current McMaster GSA softball league platform is outdated and requires extensive programming knowledge for administrators to manage league operations. This limitation makes it difficult for non-technical users, such as league representatives, to effectively maintain the system, resulting in inefficient processes for scheduling, score tracking, rescheduling, and waiver management. These inefficiencies often cause delays in communication between administrators, team captains, and participants, negatively affecting the overall league experience. Additionally, the platform lacks an intuitive, user-friendly interface, limiting the ability for non-technical users to perform essential administrative tasks autonomously, further hindering smooth league management and participant engagement.

## 1.1 Problem

The current McMaster GSA softball league platform suffers from significant limitations that prevent effective and efficient management.

The website is solely maintained by a an individual who relies on outdated methods and technologies which makes it difficult to modernize or integrate new features. Additionally, the current maintainer lacks strong communication skills, further complicating collaboration with league administrators and participants when issues arise.

One prominent issue is the platform's broken system for standings which fails to update or reflect accurate information. This undermines the competitive integrity of the league and frustrates both players and team captains, who rely on accurate standings for season progression.

Furthermore, the website contains a number of irrelevant and empty pages that serve no purpose. The disorganized structure of the website makes navigation difficult for both administrators and participants, contributing to confusion and inefficiency. This cluttered and unorganized structure wastes time and reduces the overall usability of the platform.

The user interface (UI) is also outdated, presenting a visually unappealing and unintuitive experience. It lacks the modern design and accessibility features that users expect, making it difficult for administrators to manage the league and for participants to easily find information. The poor UI further exacerbates the inefficiencies in scheduling, score tracking, and waiver management.

Without addressing these issues, the league will continue to face operational challenges, technical roadblocks, and user dissatisfaction. The system's inefficiencies lead to errors, delays in communication, and a lack of clarity for participants, which threaten the long-term success of the McMaster GSA softball league. A modernized, user-friendly platform is critical to resolving these problems and providing a better experience for both administrators and players.

## 1.2 Inputs and Outputs

Inputs include team and player data, scheduling preferences, game results, and rescheduling requests, provided by users such as commissioners, captains, and players. These inputs facilitate the generation of outputs like league schedules, standings, team rosters, and notifications. Additionally, the platform manages system-driven inputs like available game timeslots and handles outputs such as rescheduling confirmations, payment and waiver status updates, and league-wide announcements.

## 1.3 Stakeholders

- **Commissioners**

Commissioners oversee the entire league and are responsible for high-level tasks such as managing league-wide schedules, approving rescheduling requests, and handling player waivers.

- **Team Captains**

Captains manage their respective teams, handle team registrations, report game scores, and request game reschedules. They interact with the system to manage their rosters, view team standings, and coordinate game logistics.

- **Players**

Players join teams, view schedules, check standings, and receive important notifications. They need a simple way to sign waivers and interact with the platform to stay informed about team activities and game schedules.

- **Umpires**

Umpires require access to weekly game schedules to know which games they are assigned to. They benefit from clear, accessible scheduling outputs that help them prepare for upcoming matches.

- **General League Participants/Volunteers**

Beyond specific roles, all league participants, including volunteers or those indirectly involved, benefit from smooth operations, up-to-date information, and reliable game scheduling and standings.

- **Technical Support Staff/Developers**

The technical support staff/developers are responsible for maintaining the functionality and performance of the McMaster GSA softball league platform. They address technical issues, implement system updates, and ensure smooth operation for all users. Their expertise is crucial for troubleshooting problems and supporting the platform as it evolves, ultimately enhancing the user experience for other stakeholders.

## 1.4 Environment

The McMaster GSA softball league platform will operate within a web-based environment accessible through standard web browsers. The software environment will utilize the MERN stack (MongoDB, Express.js, React, Node.js) to enable efficient development and a responsive user interface. JSON Web Tokens (JWT) will be implemented for role-based access control (RBAC), ensuring secure user authentication and authorization. The platform will be hosted on a reliable cloud service provider to guarantee scalability and uptime during peak usage periods. User devices will include desktops, laptops, tablets, and smartphones, allowing flexible access to league information and functionality from various locations. The platform will prioritize responsive design to provide an optimal user experience across different screen sizes and devices.

## 2 Goals

### 1. User-Friendly League Management Platform

Develop a web-based platform that allows non-technical users, such as administrators and team representatives, to easily manage all aspects of the league without requiring programming knowledge. The platform should have intuitive navigation and role-specific access to facilitate operations like scheduling, score tracking, waiver management, and communication.

**Measurable Goal:** Achieve a user satisfaction rating of at least 85% based on post-implementation surveys with league administrators and captains.

### 2. Automated Scheduling and Standings System

Implement an automated scheduling feature that dynamically updates schedules, handles rescheduling requests, and automatically calculates and displays league standings based on match outcomes.

**Measurable Goal:** Reduce the time required to create and update schedules by 75%, verified by comparing the new platform with the previous system.

### 3. Streamlined Communication Tools

Integrate a communication system within the platform to allow direct and efficient communication between administrators and team captains, including notifications for important updates, match reschedules, or waiver completions.

**Measurable Goal:** Ensure that 100% of all notifications related to scheduling or administrative updates are successfully delivered within 1 minute.

### 4. Secure Waiver Management

Provide a secure, digital waiver management system where participants can electronically sign and submit waivers, and administrators can easily track and verify completion. The system should store waivers in compliance with relevant privacy and security standards.

**Measurable Goal:** Ensure 100% compliance with privacy policies and securely manage all participant waiver data.

### 5. Role-Based Access Control

Implement role-based access control to ensure that different users, such as league administrators and team captains, can access only the information and functionalities relevant to their responsibilities. This will minimize administrative errors and safeguard sensitive league data.

**Measurable Goal:** Verify that role-based permissions prevent 100% of unauthorized access attempts.

### 3 Stretch Goals

#### 1. Payment Status Tracking for Players

Add a feature that displays the payment status of players, indicating whether they have completed their payment via e-transfer. This visibility will help captains and commissioners manage team rosters and ensure compliance with league participation requirements. **Measurable Goal:** Achieve 100% accuracy in reflecting players' payment statuses, verified through periodic audits against external payment records.

#### 2. League-Wide Announcement System

Implement a feature that allows commissioners to post league-wide announcements directly on the platform. This functionality would enhance communication by ensuring that all participants receive important updates, such as schedule changes, event reminders, or policy updates, in a centralized location. **Measurable Goal:** Ensure that at least 90% of announcements are acknowledged by users, as measured through read receipts or user feedback surveys.

#### 3. Mobile-Friendly Interface

Design a mobile-responsive version of the platform so that users can access and manage league functionalities from their smartphones. This feature would add convenience for participants and administrators who need to interact with the system on the go.

**Measurable Goal:** Ensure that the mobile interface has at least 80% feature parity with the desktop version, verified through feature testing and user feedback.

#### 4. Predictive Scheduling with Machine Learning

Explore the use of machine learning algorithms to optimize scheduling by predicting potential conflicts, such as weather conditions or team availability, and automatically proposing alternative schedules to minimize disruptions.

**Measurable Goal:** Achieve a conflict prediction accuracy of 85%, verified by comparing predicted and actual conflicts.

## 4 Challenge Level and Extras

[State your expected challenge level (advanced, general or basic). The challenge can come through the required domain knowledge, the implementation or something else. Usually the greater the novelty of a project the greater its challenge level. You should include your rationale for the selected level. Approval of the level will be part of the discussion with the instructor for approving the project. The challenge level, with the approval (or request) of the instructor, can be modified over the course of the term. —SS]

[Teams may wish to include extras as either potential bonus grades, or to make up for a less advanced challenge level. Potential extras include usability testing, code walkthroughs, user documentation, formal proof, GenderMag personas, Design Thinking, etc. Normally the maximum number of extras will be two. Approval of the extras will be part of the discussion with the instructor for approving the project. The extras, with the approval (or request) of the instructor, can be modified over the course of the term. —SS]

Our expected challenge level is general.

The reason for this choice is this project has many potentially challenging components such as creating a scheduling algorithm that we may have difficulty with amongst many other functional requirements of the web application. That being said, we all have some relevant experience in full stack development making it more manageable. Additionally, many of us have experience playing intramurals and thus using McMaster's intramural sports management website. This gives us similar first-hand perspective into the stakeholders of our project allowing us to empathize with their needs and develop accordingly with minimal confusion. Lastly, we have access to the existing website which can be leveraged to further understand our requirements and how to implement them.

The extras we have described for our project include usability testing, user documentation, and design thinking. We believe these are strongly relevant to our project since it is a heavily user-centric web application that must incorporate many principles of human-computer interfacing. We will track our changes consistently with written documentation as well as version control (via Git and GitHub). We will incorporate design thinking when developing and designing our system and UI, and we will conduct comprehensive usability tests with existing/potential stakeholders to compile improvements and verify design decisions.

## Appendix — Reflection

[Not required for CAS 741 —SS]

The purpose of reflection questions is to give you a chance to assess your own learning and that of your group as a whole, and to find ways to improve in the future. Reflection is an important part of the learning process. Reflection is also an essential component of a successful software development process.

Reflections are most interesting and useful when they're honest, even if the stories they tell are imperfect. You will be marked based on your depth of thought and analysis, and not based on the content of the reflections themselves. Thus, for full marks we encourage you to answer openly and honestly and to avoid simply writing "what you think the evaluator wants to hear."

Please answer the following questions. Some questions can be answered on the team level, but where appropriate, each team member should write their own response:

1. What went well while writing this deliverable?
2. What pain points did you experience during this deliverable, and how did you resolve them?
3. How did you and your team adjust the scope of your goals to ensure they are suitable for a Capstone project (not overly ambitious but also of appropriate complexity for a senior design project)?