Problem Statement and Goals
Student Evaluation App

Team 29
Nicholas Fabugais-Inaba
Casra Ghazanfari
Alex Verity
Jung Woo Lee

Table 1: Revision History

Date	Developer(s)	Change
September 23, 2024	NFI, JL, CG, AV	Initial Draft
Date2	Name(s)	Description of changes
...	...	...

1 Problem Statement

[You should check your problem statement with the problem statement checklist. —SS]

[You can change the section headings, as long as you include the required information. —SS]

1.1 Problem

1.2 Inputs and Outputs

1.2.1 Inputs

- Student course feedback
- item 2
- item 3

1.2.2 Outputs

- item 1

- item 2
- item 3

[Characterize the problem in terms of "high level" inputs and outputs. Use abstraction so that you can avoid details. —SS]

1.3 Stakeholders

- Course instructors
- Data scientists
- Educational research experts
- Administrators

1.4 Environment

Software Course instructors

Hardware Data scientists

[Hardware and software environment —SS]

2 Goals

Accomplish everything the existing league website does The current website allows captains to log in and record their matches and scores. It allows scheduling and rescheduling, and provides a place to see the league rules, parking information and other information. The current website often breaks, requiring the current website admin to fix issues as they arise. First and foremost we need to recreate the original league website functionality.

User interface should be intuitive to all users. The current interface is unintuiative and awkward to use. Users should understand how to log in and how to view their schedule just by looking at their homepage. No external information should be required.

Allow players to make accounts Currently, only captains have accounts in the system. Player accounts should only be able to view the contact information of their team captain, captains should only be able to view the contact of their players and other captains, and commisioners should be able to see everything.

Matches should be able to be scheduled and rescheduled. Team captains should be able to give their team's availablity and the software will algorithmically schedule the season's matches. If a team isn't available for a match after it has been scheduled, captains can send a reschedule request with a selection of possible alternative times that the opposition team's captain can agree to.

Commisioners should be able to notify captains with information Commisioner level accounts should be able to easily send out a notification to specific users or entire groups of users, such as all captains or all players. The information in the notification should be customizable by the commisioner.

3 Stretch Goals

Commisioners should be able to "rain out" matches After a match has been scheduled, commisioner level accounts should be able to force a reschedule if the weather makes the game unreasonable to play. This will send a notification to the two team captains so they can choose a date that works.

League template saving A season's teams and players should be able to be saved as a template that can be loaded the next season. This is useful as many teams remain the same or similar between seasons, and it would be convienient for all returning teams to avoid reinviting all returning players.

A mobile application companion Users would be able to perform some actions they can on the website, like viewing schedules and standings.

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]

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)?