

COMP 4350 • Winter 2020 • John Braico

Project Proposal

Bonnie Tang	tangb34@myumanitoba.ca
Brenna Epp	eppb@myumanitoba.ca
Daryl Fung	lerhxind@myumanitoba.ca
Earl Placido	placidej@myumanitoba.ca
Emmanuella Osinaike	osinaike@myumanitoba.ca
Jennifer Seo	seoj@myumanitoba.ca
Kevin Hoang	hoangk@myumanitoba.ca
Ovietobore (Toby) Oghre	oghreo@myumanitoba.ca
Renz Cabusas	cabusasr@myumanitoba.ca

schedule-me-up

Vision Statement

For students and work groups, make scheduling meetings less time consuming for all party members.

Motivation

Our motivation came out of our own struggles to find a time to all meet up. We are building this software in hopes that scheduling meetings becomes easier for others.

Although there are similar products in the market, these do not ultimately select or even list good meeting times. In these applications, it is up to the group to review all possible meeting times and select a final meeting time. Our application will provide the most optimal time for all group members to meet up. Our application also alleviates the time burden from the organizer - it distributes the responsibility among all team members by allowing each group member to fill in their own availability and having the application calculate the optimal meeting time(s).

Success Criteria

The following are our two success criteria:

- Decrease the overall time spent on determining a meeting time by at least 50%
- Decrease an individual's time spent on inputting their availability by at least 50%

We consider the project successful if it meets at least one of the two success criteria.

For example, it took our group (10 members) 20 minutes to find a time that works for the majority of our members. We would like to decrease that time to 10 minutes or less.

Technical Details

Server and Server-side Frameworks

We will use AWS EC2 to deploy a Node.js server with Express.js.

We chose Node.js as it has a large and active community which is a huge advantage for support. Additionally, we will be using React for our web application and therefore, it makes sense for us to build our backend with JavaScript as well.

We chose Express because it has a lot of open-source libraries for common tasks (routing, authentication, and so on) which would be at our disposal. Express is also easy to learn and has a large community around it. Some of our group members also have experience with Express, which will help us get started.

Because of the popularity of Node.js and Express there is an abundance of learning material, from documentation to tutorials, that we can take advantage of.

Data Storage

We will use a relational database management system, MySQL.

A relational database fits our app well as there are relatively clear relationships in our app - a rigid design will not impair our development. Additionally, relational databases are fast performing, which is important for the time-saving vision of our app.

We chose MySQL because it is open source, it is easy to use, and it has a large community, so finding solutions to problems will be fairly easy. Integrating MySQL with our Node server is also straightforward due to the many modules available.

Platforms

We will deploy to web and to Android.

A web app is essential as it is versatile and can be used by anyone regardless of whether they are on a PC, a laptop, an Android phone or an iPhone. We decided to build an Android app as well since most of us use or own an Android phone.

Client-side Web Frameworks

We will use React to build the front-end for the web application and Node.js/Express for our backend.

We decided on React due to its popularity and marketability as a skill. A couple of points in favour of using React:

- On Stack Overflow's developer survey, React was the most loved framework for 2019.
- Gaining knowledge in React would also mean we have a basis for writing applications in React Native, another marketable skill.
- React has lots of documentation and a few people on the team have some experience using React - they can get us started and help the rest of us if need be.
- React encourages easy to create and reusable components, a pattern that those of us who have used React or similar frameworks find valuable.

Programming Languages

We will use JavaScript, with the library ReactJS to create our web application, as the library is tied to React.

We will use Java to program the native Android application since we are familiar with both Java and Android Studio, and most Android Studio resources reference Java.

Mobile Application Technologies

We will use Android Studio.

We chose Android Studio as it is officially supported by Google - this means that it is both a desirable skill for future native development and that it has a relatively large community and good documentation. Additionally, most of us know how to use it from previous experiences, such as from COMP 3350, which makes ramp-up take less time so we can focus more on programming the actual application.

Repository

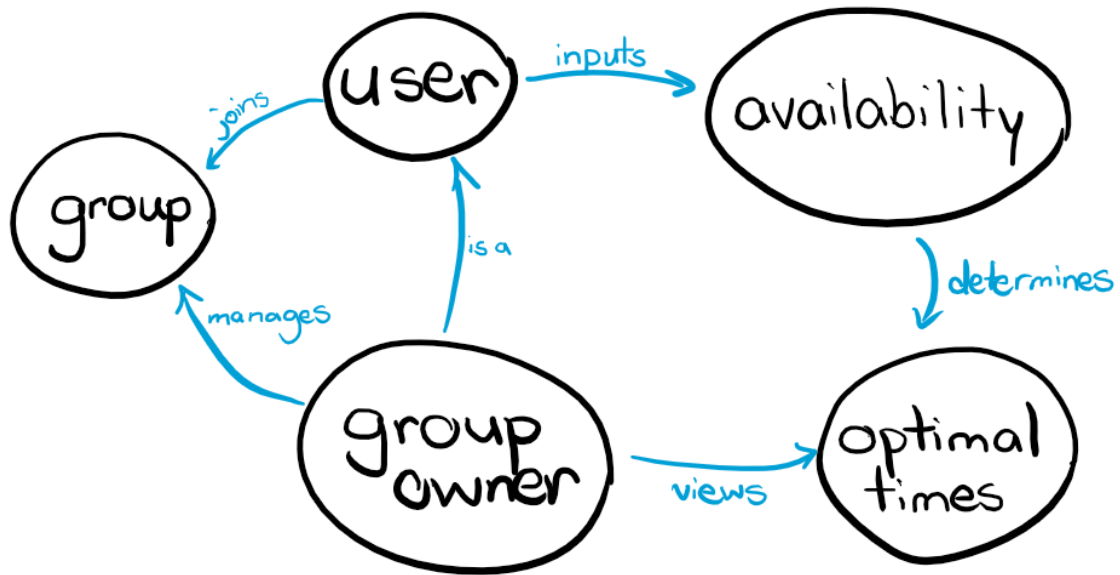
[Initial repository on GitHub](#)

Tracking System

We are using Trello for our project board as it seems more usable than GitHub's.

[Link to the project board](#)

Data Model Diagram



User Stories

As a user, I want to be able to create groups. **(high)**

As a user, I want to be able to send invitations to other users to join my group. **(high)**

As a user, I want to be able to input my availability so that the application can determine the optimal meeting time. **(high)**

As a user, I want to be able to "log in" as a guest when inputting my availability for a group. **(high)**

As a group owner*, I want to be able to see the best time in descending order so I can select a time. **(high)**

As a user, I want to be able to create an account to keep track of my different groups and change my availability when I open the app again. **(high)**

As a group owner, I want to be able to specify the length of the meeting so an appropriate time can be chosen. **(high)**

As a user, I want to be able to set public or private events to my availability so that only public events are visible to others. **(medium)**

As a user, I want to be able to edit my availability if I make a mistake or my availability changes. **(medium-high)**

As a group creator, I want to be able to restrict others in the group from access to group owner functionality (ex: deleting the group, selecting a meeting time...). **(medium)**

As a group owner, I want to be able to assign the group owner role to others in the group as well. **(medium)**

As a group owner, I want to notify other users in my group once the meeting time has been set. **(medium)**

As a user with an account, I want to be able to view the groups I am currently in. **(medium)**

As a group owner, I want to be able to select options for the meeting frequency I'm scheduling (recurring meetings) so I don't have to re-do a group or meeting each time. **(medium)**

As a user, I want to be able to sync the application with my calendar. **(medium-low)**

As a group owner, I want to be able to limit the number of users who can join my group. **(low)**

As a group owner, I want to be able to duplicate a created group. **(low)**

As a group owner, I want to be able to start a poll with times I select asking group members which they prefer, so I can select the best time if multiple optimal times exist. **(low)**

As a group owner, I want to be able to remove a specific user's availability from the group. **(low)**

As a user with an account, I want to be able to leave a group I am currently in. **(low)**

As a user with an account, I want a scheduled meeting to auto-populate my future availability. **(low)**

As a user, I want to receive a message when I haven't inputted my availability within a reasonable time frame (after a group has been created and is waiting for my response). **(low)**

As a user, I want to be able to choose whether I input my unavailability or my availability, depending on which one is better or easier for me to input. (**low**)

As a user, I want to be able to change my visible name. (**low**)

Note: Group owners and group creators may be different people. The owner/admin may be modified to include more people or give the ownership to another group member.