

Sprint 2 Planning Document

Eric Aguilera, Justin Bonner, Carson Harmon, Dan Zheng

1 Sprint Overview

For our second sprint, we will focus on the main features of our application: messaging, chat, notifications and privacy settings.

By the end of our sprint, users will be able to view public profiles of other users. Users viewing a persons public profile will be able to see their public information, such as their enrolled courses, as well as start a chat with the person.

Users will be able to create a meeting, specify a name and location, suggest possible dates and times, and invite other users to respond. Meeting invitees can RSVP with their ideal times and the user interface will allow attendees to clearly visualize which times are the best. Meeting creators can finalize a date and time and notify all attendees.

Users will also be to join chat groups and send and receive messages. Chat administrators (the first joining member) will be able to modify the privacy settings of the chat, in particular whether or not the chat is publicly visible. Notifications will help to update users about activities that have occurred in their chats and meetings.

2 Meeting Plans

Our SCRUM master will be Dan Zheng. We plan on meeting at least twice per week. We plan to meet on MWF at 11:30 AM for one to two hours.

3 Risks and Challenges

The biggest risk of this sprint is the large amount of work that needs to be done. Though it should be feasible to finish all of our tasks, there is the possibility that our features will be more difficult to implement than we think. We believe the biggest challenge in the upcoming sprint will be implementing the messaging. Messaging will require live updates to the chat page using Socket.io, which is a technology that our team is not familiar with and will need to learn.

4 Current Sprint Detail

User story #1

As a user, I would like to view my notifications.

#	Task description	Estimated Time	Owner
1	Create notification model and associate with user model	2 hrs	Dan
2	Add back-end controller for adding user notifications	2 hrs	Dan
3	Add front-end view for notifications	2 hrs	Dan
4	Add logic to existing actions that require notifications	2 hrs	Dan

Acceptance Criteria

- Given that the back-end controller for notifications is completed, when an activity involving a user, such as a meeting creation, takes place, a notification will be saved to the account of the user.
- Given that the front-end view for notifications is completed, when a user navigates to their home page, they will see a list of recent activity notifications.

User Story #2

As a user, I would like to view public profiles of other users.

#	Task description	Estimated Time	Owner
1	Add front-end view for viewing public profiles.	5 hrs	Justin
2	Add back-end controller for generating a public profile.	5 hrs	Justin
3	Incorporate privacy settings into public profiles.	5 hrs	Justin

Acceptance Criteria

- Given that the back-end controller for user profiles is completed, when a `GET` request to a user profile page is sent, the relevant user information will be retrieved from the database and sent via `express` to the front-end view.
- Given that the front-end view for user profiles is completed, when a user navigates to a profile page, they will be able to see the public information of that user..

User story #3

As a user, I would like to view and change my profile privacy settings.

#	Task description	Estimated Time	Owner
1	Add a front-end interface for privacy settings in the user profile view.	5 hrs	Eric
2	Add back-end controller for updating a users privacy settings.	5 hrs	Eric

Acceptance Criteria

- Given that the back-end controller for privacy settings is completed, when an authenticated request is sent to update a user's privacy settings, then the user's settings will be saved in the database and the information shown on the user's profile page will update accordingly.
- Given that the front-end view for privacy settings is completed, when a user navigates to their profile page, they will see buttons allowing them to change their privacy preferences.

User story #4

As a user, I would like to send and receive messages from my chats.

#	Task description	Estimated Time	Owner
1	Add messages model and associate with group model	5 hrs	Eric
2	Create prototype front-end view for chats	5 hrs	Carson
3	Add back-end logic using Socket.io	5 hrs	Eric
4	Add front-end logic using Socket.io	5 hrs	Eric

Acceptance Criteria

- Given that the back-end logic for messaging is completed, when a "send message" request is sent to the server, the new message will be saved in the database and a socket event will be sent to the client. Also, when a "get messages" request is sent to the server, then the relevant messages will be found in the database and sent to the client.
- Given that the front-end view for messaging is completed, when a user visits a chat page, they will be able to type and send messages using an input box and view messages in the chat.

User story #5

As a user, I would like to be able to create a meeting, specify a name and location, and invite other users.

#	Task description	Estimated Time	Owner
1	Create meetings model and associate with user model	3 hrs	Dan
2	Create back-end controller for creating meetings and inviting users	3 hrs	Dan
3	Create front-end view for meetings page and inviting users	3 hrs	Dan

Acceptance Criteria

- Given that the front-end view for the meetings page is completed, when a user navigates to the page for creating an account, they will see a form containing fields for email and password. When they submit the form, the fields will be validated and the data will be sent to the back-end controller.
- Given that the meetings model and back-end controller for creating meetings is completed, when a request is sent to create a meeting with name, location, and date-time fields, then the fields will be validated and the meeting will be saved in the database. When the back-end controller for inviting users is completed, when a request is sent to invite a user to a meeting, then the user-meeting relation will be saved in the database and a notification will be sent to the user.

User story #6

As a meeting attendee, I would like to be able to RSVP to a meeting by selecting the dates and times that are best for me.

#	Task description	Estimated Time	Owner
1	Create back-end controller for meetings RSVP	5 hrs	Dan
2	Create front-end view for meetings RSVP	4 hrs	Dan
3	Add data visualization of available times for meetings	5 hrs	Dan

Acceptance Criteria

- Given that the front-end view for meetings RSVP is completed, when a meeting attendee navigates to a meetings page, they will see a calendar view allowing them to select the times and dates that they are available.
- Given that the data visualization of available times for meetings is completed, when a meeting attendee navigates to a meetings page, they will see a colored calendar view that will clearly display the best times for meeting for all attendees.
- Given that the back-end controller for meetings RSVP is completed, when a meetings RSVP request is sent by a meeting attendee, then the attendee's available times will be saved in the database.

User story #7

As a meeting creator, I would like to finalize a time and location for a meeting.

#	Task description	Estimated Time	Owner
1	Create back-end controller for finalizing time and location	3 hrs	Carson
2	Create front-end view for finalizing time and location	3 hrs	Carson
3	Add logic for notifying meeting attendees	3 hrs	Carson

Acceptance Criteria

- Given that the front-end view for finalizing a meeting's time and location is completed, when a meeting creator navigates to a meetings page, they will see a calendar view that allows them to select a final time and location and a submit button.
- Given that the back-end controller for finalizing a meeting's time and location is completed, when a request is finalize a meeting, then the meeting's fields will be updated in the database and notifications will be sent to all attendees.

User story #8

As a meeting creator, I would like to be able to cancel a meeting and notify participants.

#	Task description	Estimated Time	Owner
1	Add back-end controller for handling cancellation of meetings.	3 hrs	Carson
2	Add back-end controller for sending notifications upon meeting cancellation.	3 hrs	Carson

Acceptance Criteria

- Given that the front-end view for cancelling a meeting is completed, when a meeting creator navigates to a meetings page, they will see a button allowing them to delete the meeting.
- Given that the back-end controller for cancelling a meeting is completed, when a request is sent to cancel a meeting, then meeting attendees will be notified and the meeting will be deleted in the database.

5 Remaining Backlog

Note: the user stories that will be completed during this sprint are crossed out and bolded. Previously completed user stories are crossed out but not bolded.

As a user, I would like to:

1. ~~create a Meetchu account using a valid email and password.~~
2. ~~recover my password using my email.~~
3. ~~add profile information (name, photo, major, phone number, short bio) to my account.~~
4. ~~update my account information at any time.~~
5. ~~search for courses by title and number.~~
6. ~~add a course to my account.~~
7. ~~view people in my courses.~~
8. ~~view a list of my courses.~~
9. ~~create and join study groups with people in my classes.~~
10. ~~**send personal messages to any other user.**~~
11. ~~**view message history with other users.**~~
12. see users who are online (if time allows)
13. ~~link my account with and authenticate using Google and Facebook (if time allows).~~
14. view my class schedule (if time allows).
15. search for people in my classes by name or email (if time allows).
16. view the schedules of members in my study groups (if time allows).

As a meeting creator, I would like to:

1. ~~**create a one-time meeting for a study group.**~~
2. ~~**create a personal one-time meeting with selected members.**~~
3. ~~**suggest possible dates and times for a meeting.**~~
4. ~~**add a title, purpose, and location to a meeting.**~~
5. ~~**add or remove people from a meeting and notify them.**~~
6. ~~**cancel a meeting and notify participants.**~~
7. ~~**finalize the date and time for a meeting and notify participants.**~~
8. ~~**view participants who have RSVP'd to a meeting.**~~
9. add an address to a meeting and view on Google Maps (if time allows).
10. view meeting attendance (if time allows).
11. create a recurring meeting for a study group (if time allows).

As a meeting participant, I would like to:

1. ~~**view my meetings.**~~
2. ~~**indicate my preferred times for a meeting.**~~
3. ~~**RSVP to a meeting after its date and time have been finalized.**~~
4. ~~**set time preferences for group meetings.**~~
5. ~~**receive email notifications about upcoming meetings.**~~

As a group creator, I would like to:

1. ~~be able to delete the group and notify members.~~

As a group member, I would like to:

1. ~~be able to leave the group.~~
2. ~~invite people to the group.~~
3. ~~send messages to the group.~~
4. ~~view group messages.~~