

Sprint Overview

Overview

This is the last sprint and we will be fine tuning the app. We hope to find most bugs in the code and increase its robustness. The UI is also very rudimentary and needs improving/updating. Due to this there will be fewer user stories implemented this sprint.

Scrum Master: Josh Foeh

Meeting Location/Time: Wednesday 12:30pm, CL50

Risks/Challenges: This sprint is different than the others since it encapsulates Thanksgiving, meaning that some of us will not be around on campus and might not be able to do any work during that time. This means that we have to be more productive with the shorter amount of time. We may have less time on top of this since we will want to do extensive robustness testing before the final demo as well as make aesthetic changes.

Current Sprint Detail

User Story: As a user, I would like to be able to create a group.

Tasks:

- Create UI list of users in a group. - Tushar - 4hrs
- Create UI list of all groups a user is in -Tushar - 4hrs
- Database to store groups - Josh - 4 hrs
- Database to store users in groups - Josh - 4 hrs
- Backend method to create a group - Scott - 4hrs
- Database method to store group - Trevor - 4hrs
- Backend method to get all groups associated to a profile - Scott - 4hrs
- Database method to find groups given a profile - Trevor - 4hrs
- Backend method to get group info and members - Scott - 4hrs
- Database method to fetch group info - Trevor - 4hrs

Acceptance Criteria:

- Any user can create a group
- Any user can add any user to the newly created group
- Any user can join any group
- Viewing a group will display group info and members

User Story: As a user, I would like to search based on tags

Tasks:

- Add an option on the search page to search by tag - Josh - 2 hrs
- Send tag to backend - Tushar - 2 hrs
- Database method to query database for all profiles with requested tag - Josh, Trevor - 8 hrs

- Backend method to send requested profiles to app - Scott - 1hr
- Populate listview with fetched profiles - Josh 2 hrs

Acceptance Criteria:

- Be able to specify a tag and get all users that have that tag associated with their profile
- Manually check database for all users with requested tag
- Make sure no users returned that do not have the requested tag

User Story: As a user I would like to use an app that is visually pleasing and functions smoothly

Tasks:

- Polish UI - Tushar , Josh - 10 hrs
- Bug test - All Members - 4hrs
- Robustness test - All Members - 4hrs

Acceptance Criteria:

- App never crashes
- Looks polished
- Handles errors elegantly

Remaining Backlog*Functional Requirements*

1. ~~As a user, I would like to search based on tags~~
 - a. I would like to view popular tags
2. As a user, I would like to be able to favorite other users.
3. As a user, I would like to be able to block other users.
4. As a user, I would like to be able to report users
 - a. I would like to report users that may be spam accounts
 - b. I would like to report other for disorderly conduct
5. As a user, I would like to be able to chat with other users.
 - a. I would like to receive a message when a message is sent to me
6. As a developer, I would like to be able to give others administrative powers so that the developers do not have to be the only ones monitoring reported users, etc.
7. As a developer, I would like to be able to send warnings to reported users.
8. As a developer, I would like to collect spam/reported user information.
9. As a developer, I would like to collect bug reports.
10. As a developer, I would like to have space for ads (time permitting).

11. As a developer, I would like to match users based on previous search patterns (time permitting).
12. As a developer, I would like to be able to release new updates to all users (time permitting).
13. As a developer, I would like to be able to track and store user information to utilize from a business aspect (time permitting).
14. As a developer, I would like this app to be able to handle a large number of users simultaneously (time permitting).
15. This app must integrate with SoundCloud (time permitting)
16. This app must integrate with YouTube (time permitting)

Non-Functional Requirements

1. The app must be available for Android phones.
2. This app must be available for iOS (time permitting)
3. As a developer, I would like to be able to load profiles quickly.
4. As a developer, I would like to keep sensitive user information secure.
5. The system must not be down for more than one hour a month (time permitting)