Team 18 - Sprint Overview

Study Buddy

SCRUM Leader: Andy Chen
Team Members: Abhijay Gupta, Joseph Miller, Nathan Scott, Zeb Wezensky
Meeting Schedules:

- During (former) CS307 class time (12:30p-1:20p)
- Saturday Afternoon Coding Session (2p-4p)

Overview:
In this sprint, we plan on establishing the connection between the client and the server/database. We also plan to set up the chat system and begin dealing with multi-person connections, i.e. group chat and matches with multiple people.

1. As a developer, I want to become familiar with the environment I will be working in and the technologies I will be working with.
   (**15 hours**) **(Andy, Nate, Zeb)**
   a. Download development environment (Android Studio), get server with database up and configured, learn Android programming, implement small app with database connectivity to become familiar with coding, debugging, and deploying processes.
   b. Acceptance: Given an Android Studio session, when I try and implement an application that connects to a database, then I have the skills to do so.
2. As a user, I want to be able to chat with and send messages to other students.
   a. Frontend: Design a suitable UI that accepts messages and allows the user to send them.
   (Andy, Nate, Zeb)(5 hours)
   b. Backend: Design and implement database tables and functions to store and retrieve user messages.
   (Abhijay, Joey)(20 hours)
   c. Acceptance: Given two users and a message, when I send the message from one user to the other, the sender is able to easily send the message and the receiver is able to view it in a timely fashion.

3. As a user, I want to view my chat history.
    a. Frontend: Design a history list that will be displayed on the messages page when communicating with other users.
        (Andy, Nate, Zeb)(10 hours)
    b. Backend: Add API functions to retrieve chat messages sent and received by a particular user in chronological order.
        (Abhijay, Joey)(10 hours)
    c. Acceptance: Given a user, when I try to view the chat history, I am able to and the history is accurate.
4. As a developer, I want the API I use to be robust and well-documented
        (Abhijay, Joey)(10 hours)
    a. Improve existing API call logic
    b. Increase usability of returned data
    c. Document existing API functions and create framework for new documentation
    d. Acceptance: Given an API function, when our team wants to use it, they can find out how to use it quickly and easily and the data is returned efficiently and in a usable format.
5. As a user, I want to be able to login using my Purdue email and password.
    a. Frontend: Add user to database
        (Andy, Nate, Zeb)(5 hours)
    b. Acceptance: Given a purdue student, when they open the app, then they are presented with the Purdue login page and can login successfully.
6. As a user, I want to enter any classes.
    a. Frontend: Use search to display classes, and add class to class list.
        (Andy, Nate, Zeb)(5 hours)
    b. Acceptance: Given a list of classes, when I want to enter them on the app, then I am able to see the change and it propagates to the database.
7. As a user, I want to enter other interests so that I can be matched with a more compatible individual.
    a. Frontend: Add to interest list
        (Andy, Nate, Zeb)(10 hours)
    b. Acceptance: Given a list of interests, when I want to enter them on the app, then I am able to see the change and it propagates to the database.
8. As a user, I want to view other people’s profile.
    a. Frontend: Populate profile information from database
        (Andy, Nate, Zeb)(5 hours)
    b. Acceptance: Given a user, when I want to enter them on the app, then I am able to see their profile picture, bio, course, and interest.

9. As a user, I want to see contact information for the students I am matched with.
    a. Frontend: Add contact to profile and populate username from email
        (Andy, Nate, Zeb)(5 hours)
    b. Acceptance: Given a user, when I want to enter them on the app, then I am able to see their username on the profile.
10. As a user, I want to be able to meet with students as a group.
    a. Frontend: Implement group chat feature and page. Link users in the chat and allow multi-directional communication.
        (Andy, Nate, Zeb)(5 hours)
    b. Backend: Implement calls to retrieve messages sent to a certain group
        (Abhijay, Joey)(5 hours)
    c. Acceptance: Given a group of users, when I want to create a group chat with them, I am able to, and all included users receive all messages sent.
11. As a user, I want to get a notification when contacted.
    a. Frontend: Add ability to display notifications in the taskbar.
        (Andy, Nate, Zeb)(10 hours)
    b. Backend: Add functionality to retrieve only new messages
        (Abhijay, Joey)(10 hours)
    c. Acceptance: When we receive a message, the notifications icon changes from its default view.
12. As a user, I want to be able to block other users so that I do not see their messages.
    a. Frontend: Add block functionality to UI by sending this information to the database and implementing the ability to not receive messages from the blocked user.
        (Andy, Nate, Zeb)(5 hours)
    b. Backend: Do not send data related to blocked users
        (Abhijay, Joey)(5 hours)
    c. Acceptance: When a user blocks another user, they wouldn’t be able to view their profile or chat with them.
13. As a developer, I want users to be able to provide feedback.
    a. Frontend: Add functionality to the feedback page and send the feedback to the developers via the database.
        (Andy, Nate, Zeb)(5 hours)
    b. Acceptance: When a user sends any feedback, it should appear on the feedback database for the developers to view.
14. As a user, I want to view other people’s profile.
    a. Frontend: Implement linkage to other people’s profile pages from the courses they and the user have in common.
        (Andy, Nate, Zeb)(5 hours)

b. Acceptance: Given some profile information, when a user enters that information on the manage profile screen, then the changes propagate to the database.

Remaining Backlog

Functional:

1. As a user, I want to be able to specify whether I am looking for a group of students or just one.
2. As a user, I want a “feed” of announcements about classes I’ve enrolled in.
3. As a user, I want to be able to post to the announcements feed for each of my classes.
4. As a user, I want to see students who are in the same classes as me.
5. As a user, I want to be able to customize my notification settings.
6. As a user, I want to be able to report other users for inappropriate conduct.
7. As a user, I want to be able to customize the look and feel of the application.
8. As a developer, I want to match users based on what classes they are taking and interests they may have.

Non-functional:

1. As developers, we want our application to be simple, straightforward, and easy to use with little to no introduction.
2. As developers, we want our application to utilize a clean and professional user interface.
3. As developers, we want our application to have no bugs if possible.
4. As users, we want our application to be as fast as possible without sacrificing features of the app.
5. As developers, we must be able to accommodate as many users as possible.
6. As developers, we want to store as little data on the client device as possible.
7. As a user, I want to be able to see all relevant information with minimal effort.
8. As a user, I want to always be able to access the app.
9. As a developer, I want to reduce server downtime as much as possible.