



Team 28 Sprint 3 Planning Document

Curtis Maves, Geoffrey Myers, Jonathan Poholarz, Zachary Rich

Sprint Overview

The main goal of our third sprint will be to implement the message board and enhancing the look and feel of the application. Unlike the previous sprint, this sprint will require a lot less learning because the main features that need to be implemented, such as the message board, are implemented fairly similarly to the messaging system and existing features. Because of this, most of our time will be spent fine tuning the application and fixing bugs.

Scrum Master: Geoffrey Myers

Meeting Schedule

Mondays and Wednesdays at 11:30am

Risks/Challenges

The largest challenge of this sprint will be time management. Since this sprint is shorter due to presentations, we have to manage our time well to implement all the features we wanted to get to and create a finished product that our whole team is happy with.

Current Sprint Detail

User Story #1

As a user, I want a notification when someone answers my ad.

#	Description	Time	Owner
1.	Implement system to connect with native android notifications	2 hours	Geoffrey Myers
2.	Connect message system to notifications	4 hours	Curtis Maves
3.	Implement the ability to tap on the notification and have it take a user into the app.	2 hours	Geoffrey Myers

Acceptance Criteria:

- Given that a user is messaged, they should receive a notification on their phone that a message has been sent to them.
- Given that the user notification system is implemented properly, when a user is notified, they should be able to see what class the student who responded to their ad is in.
- Given that the user notification system is created properly, when a user receives a notification and taps on it, they should be taken to the messaging screen.

User Story #2

As a user, I want to message other users.

#	Description	Time	Owner
1.	Create messaging page UI	2 hours	Geoffrey Myers
2.	Setup messaging functionality on client to connect with server	2 hours	Jonathan Poholarz
3.	Write and test client-side functionality for message sending and retrieval	2 hours	Jonathan Poholarz
4.	Write function for only sending a certain number of messages back when requested	5 hours	Curtis Maves

Acceptance Criteria:

- Given that a user clicks the “Message” button on a user’s profile, they will be taken to a page where they may chat with selected user.
- Given that the messaging system is implemented properly, when the user goes to view their messages, they should be able to scroll through their previous messages with the other user.
- Given that the messaging system is created correctly, when the user goes to view their messages, they should be able to view all their conversations with other users.

User Story #3

As a user, I want to be able to delete my own post or ad.

#	Description	Time	Owner
1.	Implement client-side functionality	3 hours	Jonathan Poholarz
2.	Make sure posts get deleted as they should on the database	3 hours	Curtis Maves

Acceptance Criteria:

- Given that a user has a post and wants it deleted, the user will be able to select their post and send the request to the server.
- Given that the user is trying to delete their own post, the server will receive the request and remove it from the database.
- Given that the post was properly deleted, the status 200 will be received by the client and the post will be deleted from users' views.

User Story #4

As a user, I want to block other users.

#	Description	Time	Owner
1.	Implement block button	2 hours	Geoffrey Myers
2.	Implement server function for blocking messages/anything from blocked user	5 hours	Zachary Rich
3.	Ensure client does not display what should be blocked	3 hours	Jonathan Poholarz

Acceptance Criteria:

- Given that a user does not wish to hear from another user, they should be able to block the other user and not be messaged by them again.

User Story #5

As a user, I want to reset my password if I forget it.

#	Description	Time	Owner
1.	Write functions to send the emails to the users' stored email address.	3 hours	Jonathan Poholarz
2.	Create functions to ensure the password is reset correctly	2 hours	Jonathan Poholarz
3.	Ensure server gets updated as it should	3 hours	Curtis Maves

Acceptance Criteria:

- Given that the email system is setup correctly, when the users go to reset their password, they will be sent an email in which they will use to reset their password.
- Given that the password reset system is implemented properly, when a user is done creating a new password, they will be taken back to the initial login page.
- Given the user submits a request for a password request and changes their password, the server should update the database accordingly and allow the user to log in with their new password.

User Story #6

As a user, I want to upload an avatar

#	Description	Time	Owner
1.	Add UI and client-side functionality for user to select an avatar image	3 hours	Geoffrey Myers
2.	Provide a way to transfer avatar images to the server	5 hours	Jonathan Poholarz
3.	Implement functionality for storing the images in the database	4 hours	Zachary Rich
4.	Implement functionality for updating the user's photo in the database	6 hours	Curtis Maves

Acceptance Criteria:

- Given that the upload function is completed, when a user goes to upload a photo, they should be able to upload straight from their photo gallery on their phone.
- Given that avatar updating functionality is completed correctly, when a user goes to update their avatar, their photo should then be updated in the database and be updated in every instance where their avatar appears.
- Given that the avatar updating functionality is implemented correctly, when a user is done updating their avatar, they should then be taken back to their new updated profile page.

User Story #7

As a user, I want to post to a message board for a class.

#	Description	Time	Owner
1.	Create UI page for message board	4 hours	Geoffrey Myers
2.	Create route for the message board	5 hours	Zachary Rich
3.	Create document for all messages to be stored in the database	2 hours	Curtis Maves
4.	Add ability to retrieve only a certain number of messages	3 hours	Curtis Maves
5.	Implement message board client-server communication functions	4 hours	Jonathan Poholarz

Acceptance Criteria:

- Given that a user clicks on a class, they will be able to navigate to the message board
- Given that a user reaches the message board, they will be able to view a list of messages from various users related to that class

User Story #8

As a user, I want to be able to report ads and posts.

#	Description	Time	Owner
1.	Create UI report button/Admin report page	3 hours	Geoffrey Myers
2.	Implement server function to save reports	5 hours	Curtis Maves
3.	Allow only admins to access reports client side	3 hours	Jonathan Poholarz
4.	Allow only admins to access reports server side	5 hours	Zachary Rich

Acceptance Criteria:

- Given that someone finds a post inappropriate or offensive, they should be able to report the post
- Given a post is reported, only admins should be able to see the reports.

User Story #9

As a user, I want the app to be visually appealing.

#	Description	Time	Owner
1.	Make UI pretty	5 hours	Geoffrey Myers
2.	Make server code pretty	3 hours	Zachary Rich

Acceptance Criteria:

- Given that a user installs the app, they should be greeted with an aesthetically pleasing login page and a easy to navigate UI to complement it.
- Given that multiple pages connect to each other, the UI should be laid out similarly across all pages under a general theme.
- Given that someone is reading through the server code, it should be clean and easily readable.

User Story #10

As a student I want to filter results.

#	Description	Time	Owner
1.	Create filter UI	4 hours	Geoffrey Myers
2.	Implement filter functionality	6 hours	Jonathan Poholarz

Acceptance Criteria:

- Given that the filters are implemented effectively, the users will be able to narrow their class and user search results based on various criteria

User Story #11

As an admin, I want to ban users.

#	Description	Time	Owner
1.	Create buttons for admin accounts	6 hours	Geoffrey Myers
2.	Create server function for banning and blocking users	5 hours	Zachary Rich
3.	Create server functions for moderating posts	5 hours	Zachary Rich

Acceptance Criteria:

- Given that a user is offensive and/or posting offensive content, an admin or moderator should be able to ban the user if necessary
- Given that a post is reported, an admin or moderator should be able to take action as needed to fix the issue.
- Given that an ad or post is offensive, an admin or moderator should be able to change the post to be more appropriate.

User Story #12

As a developer, I want well-documented code.

#	Description	Time	Owner
1.	/*Commenting*/	2 hours each	Everyone

Acceptance Criteria:

- Given that someone is trying to read through the code and understand what the functions, routes, and other code is doing, they should be able to read well structured comments before every function and route.
- Given that logic is not straightforward how it is done, someone should be able to find a comment explaining what the specific snippet of code is doing and possibly why.

Remaining Backlog

Functional

1. As a user, I want to sign in.
2. As a user, I want to select a class.
3. As a user, I want to create a profile.
4. **As a user, I want to upload an avatar.**
5. **As a user, I want a notification when someone answers my ad.**
6. ~~As a user, I want to message other users.~~
7. As a user, I want to post to a message board for a class.
8. As a user, I want to create an account.
9. **As a user, I want to reset my password if I forget it.**
10. As a user, I want to choose if I am a tutor or student, or both.
11. As a user, I want to display what hours I am available.
12. As a user, I want to access this all from a web browser (if time permits).
13. As a user, I want to block other users.
14. As a user, I want to be able to report ads and posts.
15. As a user, I want to be able to pay or tip a tutor (if time permits).
16. As a user, I want to create a group conversation (if time permits).
17. As a user, I want to be able to delete my own post or ad.
18. As a user, I want my messages to be private (if time permits).
19. **As a user, I want the app to be visually appealing.**
20. As a student, I want to post an ad.
21. As a student, I want to be able to rate tutors.
22. **As a student I want to filter results.**
23. As a tutor, I want to list my classes.
24. As a tutor, I want to post an ad.
25. As a tutor, I want to list my prices.
26. **As an admin, I want to ban users.**
27. **As an admin, I want moderating capabilities of ads and posts.**
28. **As a developer, I want well-documented code.**
29. As a developer, I want to take a cut of payment through the app (if time permits).

Non-Functional

1. ~~The app will run on android devices~~
2. The app will run on other platforms, such as iOS (if time permits).
3. The app will be accessible from the web (if time permits).
4. ~~The app will connect to a back-end webservice and database, such as mongoDB.~~
5. **The app will be simple to use and easy to navigate.**
6. **The app will be quick and responsive.**
7. ~~The system will be able to handle multiple users.~~