Apollo

🚀

Sprint 3 Retrospective

Team 4

Jacob Beene

Ali Darwish

Brandon Jiang

Edward Potapov

Daniel Sowers

Saimonish Tunguturu

What Went Well

Overall, Sprint 3 was a success for Team 4; however, that is not to say that everything went perfectly. This was an ambitious sprint with several important goals to achieve, and we did a great job of making progress on all fronts.

Last retrospective, we put forth a few areas that we wanted to improve on. Those areas were testing, schedule conflicts, and dedication to the project. This sprint, these areas had huge improvement. We implemented tests during development which was a great improvement over our previous approach, we planned meetings well ahead of time and communicated schedules more effectively, and put more focus into enjoying the process and being more present as a team.

One of the most important goals for Sprint 3 was implementing notifications and direct messages in real time. The immediate nature of these alerts was non-negotiable, and we are pleased to share that we were able to successfully implement this feature.

But that's not all - we also learned a ton of new skills throughout the sprint. We familiarized ourselves with Socket.IO, learned how to use new React components, and even dabbled around with Google Maps API. It's amazing to see how much progress we've made in such a short amount of time.

In addition, we also polished some of our previously released features. We brought edit and delete functionality to threads, reviews, and comments making the user experience more seamless. We also added more descriptive fields to reviews for our users.

Overall, Team 4 is proud of what we have been able to accomplish over the past ten weeks. We all started out with little to no experience building a project like this. It is bittersweet (mostly sweet) to have reached the end of the semester, but we are excited to continue our undergraduate careers with the skills and knowledge that we have gained throughout this course.

User Story #29

As a user, I want ratings / reviews to contain more specific information such as difficulty, enjoyability, whether or not attendance is required, etc.

Tasks	Status	Estimated Time	Owner
Add UI components	Completed ▾	3 hour	Ali
Update POST payload	Completed ▾	0.5 hour	Ali
Update API (backend)	Completed ▾	2 hour	Daniel
Automatic tests for correctness + debugging	Completed ▾	1 hour	Daniel
Automatic tests for error handling + debugging	Completed ▾	1 hour	Daniel

Completed

We added more fields to reviews that gives users a better understanding of the quality of the class. This included adding more elements to the review form and displaying those values in the review section of Apollo.

User Story #30

As a professor, I want to be able to remove files from the course homepage.

Tasks	Status	Estimated Time	Owner
Add UI component	Completed ▾	1 hour	Edward
Send "remove" request	Completed ▾	0.5 hour	Edward
Add remove file API request	Completed ▾	2 hours	Jacob
Manual Tests for correctness + debugging	Completed ▾	1 hour	Jacob
Manual Tests for error handling + debugging	Completed ▾	1 hour	Jacob

Completed

We added a trash can icon for only the professor to see. Once it was clicked we had the backend remove it from our database and also delete the file in the file system. This change would remove the link for the pdf on the front end for all other users.

User Story #34

As a user, I would like to view an interactive map of campus on the map page

Tasks	Status	Estimated Time	Owner
Integrate Map API	Completed ▾	2 hours	Edward
Create Map Overlay	Completed ▾	3 hours	Edward
Collect data	Completed ▾	3 hours	Sai
Backend API	Completed ▾	2 hours	Sai
Manual Tests for correctness + debugging	Completed ▾	1 hour	Sai
Manual Tests for error handling + debugging	Completed ▾	1 hour	Sai

Completed

We added a new page to Apollo that can be searched for in the search bar. When you search and click Map you are given a map fixed on Purdue that shows pins for all the locations across campus and the surrounding area.

User Story #49

As a user, I want to be able to send direct messages to my friends.

Tasks	Status	Estimated Time	Owner
Create "messages" page	Completed ▾	3 hours	Ali
Create "conversation" UI	Completed ▾	3 hours	Edward
Add search for friends when creating dm	Completed ▾	1 hour	Ali
Send API requests	Completed ▾	1 hour	Jacob
Write backend objects	Completed ▾	3 hours	Jacob
Write API requests	Completed ▾	3 hours	Jacob
Automatic tests for correctness + debugging	Completed ▾	2 hour	Jacob

Tasks	Status	Estimated Time	Owner
Automatic tests for error handling + debugging	Completed ▾	1 hour	Jacob

Completed

We implemented a real-time chat room for registered users. This is found by clicking the "Chats" button in the top right of the landing page. Users can directly message and create group chats with their friends.

User Story #53

As a user, I want to be able to add / remove other users from my friends list

Tasks	Status	Estimated Time	Owner
Create button to send friend request	Completed ▾	1 hour	Edward
Create UI to a view a list of your friends	Completed ▾	3 hours	Edward
Create notification for friend requests	Completed ▾	1 hour	Daniel
Create UI for accept / reject friend request	Completed ▾	1 hour	Sai
Handle pending friend requests	Completed ▾	2 hours	Brandon
Add backend to store, manage and retrieve friend list.	Completed ▾	2 hour	Daniel
Automatic tests for correctness + debugging	Completed ▾	1 hour	Daniel
Automatic tests for error handling + debugging	Completed ▾	1 hour	Daniel

Completed

We added the ability for users to either add or remove other users from their friends list. When the user visits his/her profile and clicks on their friends list, they will now have the option to either add or remove the user from this list.

User Story #54

As a user, I want to be able to add and remove other users to my block list

Tasks	Status	Estimated Time	Owner
Create UI to add user to your block list	Completed ▾	1 hour	Brandon
Add "blocked" restrictions (frontend)	Completed ▾	2 hours	Brandon
Add "blocked" restrictions (backend)	Completed ▾	2 hours	Jacob
API work (backend)	Completed ▾	1 hour	Jacob
Manual tests for correctness + debugging	Completed ▾	1 hour	Sai
Manual tests for error handling + debugging	Completed ▾	1 hour	Sai

Completed

We added the ability for users to either add or remove other users from their block list. When the user visits his/her profile and clicks on their block list, they will now have the option to either add or remove the user from this list. You can also block or unblock a user by visiting the other user's profile. Once blocked, the user will automatically be removed from the friends list.

User Story #57

As a user, I want to be able to see a user's recent activity (ex: their threads and comments) on their profile page.

Tasks	Status	Estimated Time	Owner
Create UI on profile to display recent activity	Completed ▾	4 hours	Ali
Modify all back end functions to append to a list of recent activity when a user does	Completed ▾	3 hours	Jacob

Tasks	Status	Estimated Time	Owner
something (Comment / create thread / upvote / downvote, etc).			
Make a button to clear recent activity	Completed ▾	1 hour	Ali
Clear recent activity backend	Completed ▾	1 hour	Brandon
Manual tests for correctness + debugging	Completed ▾	1 hour	Sai
Manual tests for error handling + debugging	Completed ▾	1 hour	Sai

Completed

We added a new feature to keep track of the user's most recent activities. This will be displayed in the user's avatar bar on the top right of the landing page. The recent activities mostly are activities done when interacting with a thread.

User Story #59

As a user, I want to see all of my notifications in one place.

Tasks	Status	Estimated Time	Owner
Create UI pop notification	Completed ▾	2 hours	Ali
Create a switch to toggle between receive and don't receive in app notifications in the edit profile section	Completed ▾	1 hour	Ali
Add general backend function to notify (poke) frontend to send in app notifications	Completed ▾	4 hours	Daniel
Manual tests for correctness + debugging	Completed ▾	1 hour	Brandon
Manual tests for error	Completed ▾	1 hour	Brandon

Tasks	Status	Estimated Time	Owner
handling + debugging			

Completed

We added a notification center inside our avatar bar on the right side of the screen. If users have notifications enabled, they will see the notification center. Otherwise, it will be automatically hidden. When a user receives a notification, they will also be prompted with a pop notification in the top left that disappears after a few seconds. The avatar bar also has a badge that indicates how many unread messages the user has.

User Story #61

As a user, I want to be able to delete things I post such as threads, reviews, and comments.

Tasks	Status	Estimated Time	Owner
Create backend functions to delete threads/reviews/comments from database	Completed ▾	2 hours	Brandon
Add UI option on thread, review, comment to delete	Completed ▾	2 hours	Edward
Automatic tests for correctness + debugging	Completed ▾	1.5 hours	Edward
Automatic tests for error handling + debugging	Completed ▾	1 hour	Brandon

Completed

For all of the elements mentioned above we had a button that allowed only the creator of the thread, comment, or review to delete that element. This would provide them a confirmation message and persist across all users.

User Story #62

As a user, I want to be able to edit things I post such as threads, reviews, comments.

Tasks	Status	Estimated Time	Owner
Add edit UI for threads	Completed ▾	1.5 hours	Edward

Tasks	Status	Estimated Time	Owner
Add edit UI for reviews	Completed ▾	1.5 hours	Edward
Add edit UI for comments	Completed ▾	2 hours	Edward
Write PATCH requests frontend	Completed ▾	2 hours	Sai
Create backend functions for edit	Completed ▾	3 hours	Brandon
Automatic tests for correctness + debugging	Completed ▾	1.5 hour	Sai
Automatic tests for error handling + debugging	Completed ▾	1 hour	Sai

Completed

We added further functionality to our threads, comments, and reviews by adding edit capability to these three objects. Thread / comment / review creators will now see an edit icon next to their post. They will be able to click it and be prompted with a window where they can edit and submit their updated post.

User Story #63

As a user, I want to be able to mark my notifications as read or unread.

Tasks	Status	Estimated Time	Owner
User Story 59	Completed ▾	See User Story 59	See User Story 59
Add read / unread UI	Completed ▾	2 hours	Brandon
API for read / unread	Completed ▾	2 hours	Daniel
Manual Tests for correctness + debugging	Completed ▾	1 hour	Sai
Manual tests for error handling + debugging	Completed ▾	1 hour	Sai

Completed

We added read / unread functionality in our notification center. To toggle read / unread for a notification, simply click on it. We also added a button at the top of the notification center for "read all notifications".

User Story #64

As a user, I want to receive notifications when a professor posts an announcement to a course that I have added to my "favorite courses".

Tasks	Status	Estimated Time	Owner
Merge "favorite courses" and "my courses"	Completed ▾	3 hours	Jacob
Add notification support to create thread API	Completed ▾	2 hours	Brandon
Automatic tests for correctness + debugging	Completed ▾	1 hour	Sai
Automatic tests for error handling + debugging	Completed ▾	1 hour	Sai

Completed

We added notifications for favorite courses. Users will receive notifications from their favorite courses when professors post an announcement. The notification they receive contains a link to the page.

User Story #65

As a user, I want to be able to add a "tag / topic" to the thread I create.

Tasks	Status	Estimated Time	Owner
Add UI for tags on threads	Completed ▾	3 hours	Edward
Add UI for filtering by tag	Completed ▾	3 hours	Edward
Update create thread API	Completed ▾	2 hours	Ali
Automatic tests for correctness + debugging	Completed ▾	1 hour	Sai
Automatic tests for error handling + debugging	Completed ▾	1 hour	Sai

Completed

We added extra fields in threads to allow users to be more specific about the type of thread they create. These tags can be topics such as "Lecture", "Projects", "Homework", etc. They also have the ability to filter threads by tag.

User Story #66

As a user, I want to be able to see a user's profile picture beside their username in threads, reviews, comments.

Tasks	Status	Estimated Time	Owner
Add UI to display the profile picture	Completed ▾	2 hours	Brandon
Profile picture links to user's profile	Completed ▾	1 hour	Ali
Manual tests for correctness + debugging	Completed ▾	1 hour	Sai
Manual tests for error handling + debugging	Completed ▾	1 hour	Sai

Completed

We added functionality that allows profile pictures to display beside a user's username wherever they post whether that be in threads, reviews, or comments.

User Story #67

As a group chat creator, I want to be able to edit my group chat (chat name and who is in it).

Tasks	Status	Estimated Time	Owner
User Story 49	Completed ▾	See User Story 49	See User Story 49
Add edit name UI	Completed ▾	2 hours	Ali
Add UI for adding / removing users	Completed ▾	2 hours	Ali
Write PATCH request for chat name	Completed ▾	0.5 hours	Edward
Handle PATCH chat name	Completed ▾	1.5 hours	Jacob
Write PATCH request for adding / removing	Completed ▾	2 hours	Edward

Tasks	Status	Estimated Time	Owner
members			
Handle PATCH add / remove members	Completed ▾	1.5 hours	Jacob
Automatic tests for correctness + debugging	Completed ▾	1 hour	Brandon
Automatic tests for error handling + debugging	Completed ▾	1 hour	Brandon

Completed

Along with the chat room we implemented, we added extra features for group chat admin so that they had the ability to add / remove users and change the name of the group chat. We also ensured that only the admin has the capability to do this.

User Story #68

As a user, I want to receive an email for every 5 unread notifications.

Tasks	Status	Estimated Time	Owner
Create email notification api in the backend	Completed ▾	2 hours	Brandon
Integrate with external email API	Completed ▾	2 hours	Ali
Automatic tests for correctness + debugging	Completed ▾	1.5 hours	Sai
Automatic tests for error handling + debugging	Completed ▾	1 hour	Sai

Completed

We added email notifications for the notification center. Users will be notified by email for every five unread in-app notifications that they receive.

User Story #69

As a user, I want to see a more exciting login page about things at Purdue (slideshow)

Tasks	Status	Estimated Time	Owner
Collect some images, Purdue related	Completed ▾	1 hour	Sai
Store images in backend	Completed ▾	1 hour	Daniel
Add Images to login page in a slideshow	Completed ▾	2 hours	Brandon
Manual Tests for correctness + debugging	Completed ▾	1 hour	Brandon
Manual Tests for error handling + debugging	Completed ▾	1 hour	Jacob

Completed

We added a slideshow on the login page that cycles through various pictures related to Purdue.

User Story #70

As a user, I want to be able to see the timeline of my friends' activity.

Tasks	Status	Estimated Time	Owner
Store users' recent activity	Completed ▾	3 hours	Daniel
Write necessary backend for users to view their friends' recent activities	Completed ▾	3 hours	Jacob
Write POST request to store recent activity	Completed ▾	0.5 hours	Daniel
Write GET request to get friends' recent activity	Completed ▾	0.5 hours	Sai
Display recent activity of friends on the UI	Completed ▾	3 hours	Edward
Manual tests for correctness + debugging	Completed ▾	2 hours	Sai
Manual tests for error handling + debugging	Completed ▾	1 hour	Sai

Completed

We added functionality in our avatar bar to view friends' timeline. It can be accessed by clicking the dropdown on Friends Timeline and then clicking the + next to a friend's name. Each friends' timeline includes their recent activity such as where they have been creating threads / comments along.

User Story #71

As a user, I want to be able to create a "club" on the clubs page.

Tasks	Status	Estimated Time	Owner
GET requests	Completed ▾	0.5 hour	Daniel
Make UI display to show all pages that are created	Completed ▾	4 hour	Ali
POST requests	Completed ▾	0.5 hour	Brandon
PATCH requests	Completed ▾	0.5 hours	Brandon
Make UI to create a new page with all information	Completed ▾	3 hours	Edward
Manual tests for correctness + debugging	Completed ▾	1 hour	Sai
Manual tests for error + debugging	Completed ▾	1 hour	Sai

Completed

We implemented a feature where users can create at most one group based on their interests. This is similar to "Facebook groups". Users can join / leave the group and create threads and leave comments on the group page. We also added edit capability to the group's name and description for the group admin.

What Did Not Go Well

Overall, Sprint 3 was a success for Team 4; however, that is not to say that everything went perfectly. Three specific areas that did not go well were delegation of work, expanding previous features, and plain old procrastination.

First, let's talk about delegation of work. This sprint, rather than having specific frontend and backend developers, we resorted to more of a full stack development system. We did this to eliminate unnecessary debugging and frustration that occurred in previous sprints due to lack of communication and error handling on the backend. This worked well, but in turn, it unbalanced the distribution of work as one developer would take on more work while another would have their amount of work reduced..

Second, we had a fair amount of issues when adding new functionality to old features. Being as new as we were to the frameworks that we were using, shortcuts and poor coding practices were definitely taken during Sprint 2 in order to finish work by the deadline. This introduced a lot of unnecessary debugging and refactoring of code during Sprint 3.

Finally, we just have plain old procrastination. We are college students and have other work for other classes. This sprint, due to that fact, we did not start work until the second week which forced us to put in overtime for week two and three of this sprint.

Overall, despite the challenges that we faced, these were all good lessons to learn which will be covered in the next section.

How We Plan to Improve

Overall, while Sprint 3 was a success, there were still areas that could be improved. As mentioned previously, we faced three main issues. These were delegation of work, expanding previous features, and plain old procrastination.

With regards to delegation of work, we moved from distinct frontend and backend developers to a full stack development system. While we think that the full stack methodology works well, it was probably not the right move for us to make in the middle of the semester. In the future, if we desire to develop in a full stack manner, it would be best to start that way. Additionally, we could put more than one full stack developer on a story. Having two people work on a bigger story together would be perfectly fine.

The challenges we faced when expanding on previous features were directly caused by the highly coupled nature of our project. This was simply due to our lack of experience when beginning development at the beginning of the semester. We found it difficult to simultaneously learn a new framework and code it in a clean, organized fashion all within a given window of time. In the future, it would be much more effective to spend more time planning and understanding the proper practices and coding styles of the languages and frameworks that we would be using.

Finally, the issue of procrastination. While unavoidable at times for students, there were definitely things we could have done differently. During the lecture section of the class, we learned about GANTT, PERT, etc. charts. This would have definitely been helpful to use throughout the sprint to ensure that we were on track to make our goals. In the future, we would definitely resort to using a more robust, organized system of time management.

Overall, this semester taught Team 4 a lot about the field of software engineering. We learned how to effectively implement the Scrum methodology, how to work as a team, and most importantly how to take action. Our time in the education system teaches us how the real world *theoretically* works. It requires a whole different approach to actually pull something off and take action. This project gave us the confidence and experience needed to go build projects on our own, and we will definitely take this newfound confidence into our future endeavors.