

Team 39 Sprint 3 Retrospective



Team Members:

Prahas Pattem, Ryan Chang, Nick Norton, Karim Mammadli, Nabi Nabiyeu

What went well?

During this sprint, we successfully completed all of our listed user stories. Users can now view their accumulated points, vote to remove users and tasks, view information that reflects data stored in the backend on their profile and in the room headers, write an “about me” section specific to each room, change their personal information and password, create posts and comment on said posts in each room, demo the application without an account, and attach files to their submitted tasks and download those files later. The UI now updates dynamically without manually refreshing the page. Performing actions on the website creates custom notifications, and the input fields are protected from invalid inputs. All the major features we planned to incorporate were implemented into our project.

User Story #1

As an individual user, I would like to be able to easily view all of my completed tasks in a list along with my total points accumulated and number of tasks completed.

#	Description	Estimated Time	Owner
1	Create UI to display a user's completed tasks in a list	2 Hrs	Nick, Ryan, & Prahas
2	Create an algorithm to retrieve all the tasks completed by the user	2 Hrs	Nick, Nabi, & Karim
3	Debug and test algorithm for getting all the tasks completed by the user	2 Hrs	Nick, Nabi, & Karim
4	Create UI to display total points accumulated and number of tasks completed	2 Hrs	Nick, Ryan, & Prahas
5	Create an algorithm to retrieve total points accumulated and number of tasks completed	2 Hrs	Nick, Nabi, & Karim
6	Debug and test algorithm for getting the total points accumulated and number of tasks completed	2 Hrs	Nick, Nabi, & Karim

Completed:

Users can now view each task that they themselves completed in a list. The list also shows the combined point total of all of their completed tasks as well as the number of tasks that they have completed.

User Story #2

As a team member, I would like to be able to start and participate in a voting system to kick a team member from a room, remove a task from the Completed tasks list and add it back to the Team tasks list, and remove a User from a Pending task.

#	Description	Estimated Time	Owner
1	Add a button to homepage UI to access the voting section of a Room	1 Hrs	Nick, Ryan, & Prahas
2	Create a UI for creating a vote	3 Hrs	Nick, Ryan, & Prahas
3	Create a UI for participating in a vote	3 Hrs	Nick, Ryan, & Prahas
4	Create necessary backend routes and schema for voting on removing a team member from a room	2 Hrs	Nick, Nabi, & Karim
5	Create necessary backend routes and schema for voting on undoing marking a task as complete	2 Hrs	Nick, Nabi, & Karim
6	Create necessary backend routes and schema for voting on removing a team member from a Pending task	2 Hrs	Nick, Nabi, & Karim
7	Testing & Debugging	2 Hrs	Everyone

Completed:

Users can now vote to forcibly remove a member from the room at any moment. They can also use the voting system to force a user off of a pending task or remove a completed task from the list if it has been deemed incomplete.

User Story #3

As an individual user, I would like to be able to view accurate room information on my homepage, profile, header, footer, and the leaderboard.

#	Description	Estimated Time	Owner
1	Create an algorithm to retrieve all the accurate information on the user's profile page	2 Hrs	Nick, Nabi, & Karim
2	Debug the algorithm for retrieving all the accurate information on the user's profile page	2 Hrs	Nick, Nabi, & Karim
3	Create an algorithm to retrieve all the accurate information for the footer	2 Hrs	Nick, Nabi, & Karim
4	Debug the algorithm for retrieving all the accurate information for the footer	2 Hrs	Nick, Nabi, & Karim
5	Create pages and hyperlink them to clicks on the footer	3 Hrs	Nick, Ryan, & Prahas
6	Update leaderboard frontend so that each element is a button that leads to a corresponding profile page	3 Hrs	Nick, Ryan, & Prahas
7	Make it so that Users cannot modify a profile that belongs to another User (since we will likely reuse our existing profile page)	3 Hrs	Nick, Ryan, & Prahas

Completed:

The homepage, profile, header, footer, and leaderboard now change to reflect information that is accurate to each room. When the user changes rooms, their profile and the room's header and leaderboard will change to reflect the new room.

User Story #4

As a team member, I would like to be able to have an About Me section that is specific to each room.

#	Description	Estimated Time	Owner
1	Create an algorithm to retrieve the About Me information the user inputs	2 Hrs	Nick, Nabi, & Karim
2	Debug the algorithm for retrieving the About Me information the user inputs	2 Hrs	Nick, Nabi, & Karim
3	Create/update necessary backend routes and schema to include About Me section	4 Hrs	Nick, Nabi, & Karim
4	Testing & Debugging About Me backend	2 Hrs	Nick, Nabi, & Karim

Completed:

Each User can go to my profile section and view their “About Me” content. When the User switches to a different room, they can see that their “About Me” content is changed.

User Story #5

As an individual user, I would like to be able to edit my personal information and change my password.

#	Description	Estimated Time	Owner
1	Create an algorithm to retrieve all the personal information to be displayed on the profile	2 Hrs	Nick, Nabi, & Karim
2	Debug the algorithm for retrieving all the personal information to be displayed on the profile	2 Hrs	Nick, Nabi, & Karim
3	Create an algorithm to update a user's personal information	2 Hrs	Nick, Nabi, & Karim
4	Debug the algorithm for retrieving all the personal information to be displayed on the profile	2 Hrs	Nick, Nabi, & Karim
5	Create an algorithm to update a user's password	2 Hrs	Nick, Nabi, & Karim
6	Debug the algorithm for updating a user's password	2 Hrs	Nick, Nabi, & Karim
7	Create necessary backend routes and schema to store the profile picture, banner photo, and profile color	2 Hrs	Nick, Nabi, & Karim
8	Debug the backend routes to store the profile picture, banner photo, and profile color	2 Hrs	Nick, Nabi, & Karim
9	Create necessary backend routes to retrieve the profile picture, banner photo, and profile color	2 Hrs	Nick, Nabi, & Karim
10	Debug the backend routes to retrieve the profile picture, banner photo, and profile color	2 Hrs	Nick, Nabi, & Karim

Completed:

To edit personal information and change password a User has to press profile information button and press gear icon besides his username icon. After doing that, User can change their password, change about Me content, email and their username.

User Story #6

As a team member, I would like to be able to create posts and comment on other users' posts.

#	Description	Estimated Time	Owner
1	Create UI to display posts	2 Hrs	Nick, Ryan, & Prahas
2	Create algorithm to retrieve posts	3 Hrs	Nick, Ryan, & Prahas
3	Debug the algorithm for retrieving posts	2 Hrs	Nick, Ryan, & Prahas
4	Create UI to create a new post	2 Hrs	Nick, Ryan, & Prahas
5	Create algorithm to store new posts	3 Hrs	Nick, Nabi, & Karim
6	Debug the algorithm for creating new posts	2 Hrs	Nick, Nabi, & Karim
7	Create UI to display comments for a particular post	2 Hrs	Nick, Ryan, & Prahas
8	Create algorithm to retrieve comments for a particular post	3 Hrs	Nick, Ryan, & Prahas
9	Debug the algorithm for retrieving the comments of a particular post	2 Hrs	Nick, Ryan, & Prahas
10	Create UI to write comments for a particular post	2 Hrs	Nick, Ryan, & Prahas
11	Create algorithm to create a new comment for a particular post	3 Hrs	Nick, Ryan, & Prahas
12	Debug the algorithm for creating a new comment for a particular post	2 Hrs	Nick, Ryan, & Prahas

Completed:

Users can now create posts that are visible to all other members of a room. The users and other members can comment on each post.

User Story #7

As an individual user, I would like to be able to view a landing page where I can choose to login if I have an account, register for a new account, or demo the application if I do not have an account.

#	Description	Estimated Time	Owner
1	Create UI for the landing page	2 Hrs	Nick, Ryan, & Prahas
2	Create algorithm for the demo account	2 Hrs	Nick, Ryan, & Prahas
3	Debug algorithm for the demo account	2 Hrs	Nick, Ryan, & Prahas
4	Set up redirects on the landing page	1 Hrs	Nick, Ryan, & Prahas
5	Testing and debugging redirects on the landing page	1 Hrs	Nick, Ryan, & Prahas

Completed:

On the landing page users can log-in to an existing account, sign-up for a new account or simply demo the application.

User Story #8

As an individual user, I would like to be able to retrieve the file uploaded with tasks that are marked as Completed.

#	Description	Estimated Time	Owner
1	Add a button to the UI of Completed tasks that sends a request for each task's associated file	1 Hrs	Nick, Ryan, & Prahas
2	Update frontend check to instead prompt a User when they try to submit a Task as Complete without a file attached	2 Hrs	Nick, Ryan, & Prahas
3	Testing & Debugging the frontend check	1 Hrs	Nick, Ryan, & Prahas
4	Only show button to get file on Completed task information UI if there is a corresponding file in the database	3 Hrs	Nick, Ryan, & Prahas
5	Testing & Debugging button visibility	2 Hrs	Nick, Ryan, & Prahas

Completed:

Users can now choose whether or not to upload files with their tasks. Tasks that have associated files will display clickable text that allows any user in the room to download said file. Tasks without files will simply display text that says "No file".

User Story #9

As an individual user, I would like the application to be secure from invalid field inputs and unauthorized requests.

#	Description	Estimated Time	Owner
1	Add validity checks on all field inputs	2 Hrs	Nick, Ryan, & Prahas
2	Testing & Debugging validity checks	2 Hrs	Nick, Ryan, & Prahas
3	Add warnings that inform the User if an input is invalid	2 Hrs	Nick, Ryan, & Prahas
4	Implement session cookies in the backend when a user signs in	3 Hrs	Nick, Nabi, & Karim
5	Testing & Debugging session cookie backend	2 Hrs	Nick, Nabi, & Karim
6	Update frontend to use session cookies instead of User_ids	1 Hrs	Nick, Ryan, & Prahas
7	Testing & Debugging frontend requests that are updated to use session cookies	2 Hrs	Nick, Ryan, & Prahas

Completed:

The application is safe from invalid field inputs and unauthorized requests. The app uses session cookies and checks inputs to see if they are valid.

User Story #10

As an individual user, I would like to be able to see the UI dynamically change instead of having to reload the page.

#	Description	Estimated Time	Owner
1	Change all UI elements that are subject to updates to use React hooks	3 Hrs	Nick, Ryan, & Prahas
2	Ensure requests to backend are set up and result in automatic changes on the UI	2 Hrs	Nick, Nabi, & Karim
3	Testing & Debugging	2 Hrs	Nick, Nabi, & Karim

Completed:

The UI now changes dynamically to reflect differences in the backend. The user no longer needs to manually refresh the page every time a task is submitted for the progress bar or leaderboard to update.

User Story #11

As an individual user, I would like to be shown information through custom notifications rather than browser alerts.

#	Description	Estimated Time	Owner
1	Create a Notification component	2 Hrs	Nick, Ryan, & Prahas
2	Update application to create Notifications rather than browser alerts	1 Hrs	Nick, Ryan, & Prahas
3	Make general Notifications automatically disappear after a set amount of time	1 Hrs	Nick, Ryan, & Prahas
4	Testing & Debugging of the notification system	2 Hrs	Nick, Ryan, & Prahas
5	Enable displaying multiple Notifications, shown in chronological order	3 Hrs	Nick, Ryan, & Prahas
6	Testing & Debugging of the notification system in chronological order	2 Hrs	Nick, Ryan, & Prahas

Completed:

When the user does something, they are now shown information through custom notifications instead of browser alerts. These notifications can let the user know that a change has occurred, that the website worked correctly, or that something went wrong. Notifications that are unrelated to errors will disappear after 10 seconds, and multiple notifications can be shown at one time.

What did not go well?

There were no user stories that we could not fully implement, but in general, time management was once again a minor issue in sprint 3. We wound up spending a disproportionate amount of time working during the final week of the project again.

How should you improve?

Overall, while our third sprint went much smoother than the first and second, there were still some points for improvement.

First, we could still be better at managing our time. During the week leading up to the third sprint review, many of us worked on the project for up to twice as long as normal. This would not have been a problem as much if we had better timed our pace in the first several weeks. To reduce the effort in the later stages of development, we can attempt to handle more challenging user stories earlier in the sprint the next time.

Our Sprint Planning document's structure is the second area where we should make improvements. We can save time and effort by better organizing things instead of constantly rearranging what needs to be done and in what order.