



Sprint One Planning Document - CS407 Team 4

Aaron Cox, Yu-Lun Gao, Michael Maganini, Raziq Ramli, Matthew Story, Bryan Wang

Sprint Overview

The goal of sprint one will be to get the application up and running with its core functionality. Mainly, the team would like to have the technology set up and running on all of our devices and have basic database functionality set up initially. With the basic technology set up, the team will be able to start development on core functionality on certain features for sprint 1. Specifically, users will be able to create and log into an account in order to access the site. They will be able to edit their profile in addition to resetting their password if forgotten. Once a user is logged-in, they will be taken to their homepage that contains links to our main features, including clubs, the class forums, events, the marketplace, and more. Additionally, each of those features has its own basic functionality that needs to be implemented as part of our “core” deliverables. For this sprint, we will mainly be focusing on user stories revolving around creation, deletion, and a basic UI for each of those features to showcase database updates/posts.

Scrum Master: Matthew Story

Meeting Schedule: Monday's at 4:00 PM + As needed on a flexible basis

Risks and Challenges

While some members of our team have thorough experience working with React, Redux, and Firebase, we have other individuals who have had little exposure to those technologies. As a result, some extra overhead is incurred because we will have to bring those members up to speed to allow for equal contribution. Another challenge surrounding our team is collaboration and communication. Due to our size, finding appropriate times to work together has proven to be difficult, something that will certainly continue moving forward. We will need to ensure that every member is responsible in their work and flexible when it comes to collaboration.

Current Sprint Detail

User Story #1

As a user, I would like to create a new account with my own email address.

#	Description	Estimated Time	Owner
1	Initialize a new Firebase Project	2 Hr	Matt
2	Configure Firestore Database	2 Hr	Matt
3	Configure Firebase Authentication	2 Hr	Matt
4	Create UI for account creation	4 Hr	Michael
5	Integrate Firebase Authentication with account creation	2 Hr	Matt
6	Implement database access with Redux	4 Hr	Matt
7	Inject additional user information into database	2 Hr	Matt
8	Test input validation	1 Hr	Matt
9	Test to ensure database is properly connected	1 Hr	Matt
10	Test to ensure user authentication is properly configured	1 Hr	Matt
User Story #1 Total Hours		21 Hours	

Acceptance Criteria

- Given the database is configured correctly, when new data is injected into the backend, the data should appear in the Firestore Database console.
- Given that user authentication is configured correctly, when a user creates a new account, that user should receive a unique entry and ID in the Firebase Authentication console.
- Given that additional user information is correctly injected into the database, when a user creates a new account, there should be an entry in the database under their unique UID that contains their bio and email address.
- Given that input validation is correctly implemented in the front-end, when a user inputs invalid information, they should receive an error message indicating what needs to be corrected.
- Given the UI for account creation is properly implemented, when a user navigates to the account creation page, they should have a username, password, biography, and image field to populate with relevant information.

User Story #2

As a user, I would like to be able to edit my account.

#	Description	Estimated Time	Owner
1	Develop UI to allow a user to edit their account	3 Hr	Yu-Lun
2	Update user information in Firebase Authentication	2 Hr	Yu-Lun
3	Update additional user information in Firestore Database	2 Hr	Yu-Lun
4	Test to ensure the user's login credentials are updated in Firebase Authentication	2 Hr	Yu-Lun
5	Test to ensure the user's additional information is updated in Firestore Database	2 Hr	Yu-Lun
6	Verify integrity of non-updated information in both Firebase Authentication and Realtime Database	3 Hr	Yu-Lun
User Story #2 Total Hours		14 Hours	

Acceptance Criteria

- Given that the UI is built correctly, when a user navigates to the edit account page, they should be met with a form that includes all relevant fields of the user's account information.
- Given that the UI is built correctly, when a user attempts to update their account information with invalid information, an error message should appear indicating which field had bad data.
- Given that updating user information in Firebase Authentication is correctly implemented, when a user edits their basic information, those changes should be reflected in the Firebase Authentication module.
- Given that updating additional user information in Firestore Database is correctly implemented, when a user edits their additional fields, those changes should be reflected in the Firestore Database module.
- Given that the code for updating user information is well-built, when a user edits any of their account information, the fields that were not updated should not be changed in Firebase.

User Story #3

As a user, I would like to delete my account.

#	Description	Estimated Time	Owner
1	Develop UI for account deletion button in edit profile page	4 Hr	Michael
2	Remove additional account information from Firestore Database	3 Hr	Michael
3	Update Firebase Authentication with account removal	3 Hr	Michael
4	Test UI works as intended	1 Hr	Michael
5	Ensure that the user data is no longer in database	1 Hr	Michael
User Story #3 Total Hours		12 Hours	

Acceptance Criteria

- Given that updating Firebase Authentication is done correctly, when a user deletes their account, they should no longer be able to login with that account.
- Given that updating the Firestore Database is done correctly, when a user deletes their account, their additional account information (including bio, image, etc.) should no longer be included in the Firestore Database.
- Given that the UI for the account deletion button works correctly, when a user clicks it, they should be prompted with an additional “Are you sure?” message box.
- Given that the UI for the account deletion button works correctly, when a user deletes their account, they should be redirected to the login page.
- Given that updating the Firestore Database and Firebase Authentication module is done correctly, when a user deletes their account, other users’ personal database information should not be affected.

User Story #4

As a user, I would like to have a unique username.

#	Description	Estimated Time	Owner
1	Create UI for user to enter their desired username	4hr	Aaron
2	Implement Firebase Database function to check if username is unique	3hr	Aaron
3	Test to ensure that unique username is stored in Firebase	1hr	Aaron
User Story #4 Total Hours		8 Hours	

Acceptance Criteria

- Given that a basic UI for creating a user is built correctly, when a user navigates to the create user page, there should be a space to enter a unique username.
- Given that Firebase Database is integrated correctly, when a user enters a username that is not unique, an error message should occur saying that the username is not unique.
- Given that Firebase Database is integrated correctly, when a user enters a username that is unique, those fields should be reflected in the Firebase Authentication module.

User Story #6

As a user, I would like to be able to change my password for my account.

#	Description	Estimated Time	Owner
1	Create UI for user to change their password for their account	4hr	Aaron
2	Implement Firebase Authentication function to ensure the user can change their password	4hr	Aaron
3	Test to ensure that the user can log in with new password	1hr	Aaron
4	Test to ensure that the user cannot log in with the old password	1hr	Aaron
5	Test to ensure that the password has been updated in Firebase	1hr	Aaron
User Story #6 Total Hours		11 Hours	

Acceptance Criteria

- Given that a UI for change password is built correctly, when a user navigates to the change password page, there should be a field to enter a new password.
- Given that a UI for change password is built correctly, when a user attempts to update their password with poor security (lacking required characters), they should be notified that their new password is not strong enough.
- Given that Firebase Authentication is integrated correctly, when a user successfully updates their password, they should be able to login with that new password.
- Given that Firebase Authentication is integrated correctly, when a user successfully updates their password, they should be logged out.
- Given that the UI is built correctly, when a user successfully updates their password, they should be redirected to the login page.

User Story #7

As a user, I would like to be able to login.

#	Description	Estimated Time	Owner
1	Develop UI for the login form	4 Hr	Michael
2	Create check against Firebase Authentication	4 Hr	Michael
3	Test to ensure the form rejects invalid users and prompts with an error message when necessary	2 Hr	Michael
User Story #7 Total Hours		10 Hours	

Acceptance Criteria

- Given that the UI for the login form is built correctly, when a user navigates to the log-in page, they should be met with a form that includes a username field.
- Given that the UI for the login form is built correctly, when a user navigates to the log-in page, they should be met with a form that includes a password field.
- Given that the basic UI is implemented correctly, when a user provides valid log-in information, that user should be redirected to their home page.
- Given that Firebase Authentication is integrated correctly, when a user provides invalid log-in information, that user should be met with an error message indicating their credentials could not be found.

User Story #8

As a user, I would like my password to be obscured when I type it to log in.

#	Description	Estimated Time	Owner
1	Develop UI to obscure password during login	2hr	Aaron
2	Test to ensure that the password is obscured during login	1hr	Aaron
User Story #8 Total Hours		3 Hours	

Acceptance Criteria

- Given that a basic UI for login is built correctly, when a user navigates to the login page, there should be a field to enter a password.
- Given that there is a space to enter a password, when a user enters a password, the characters entered should be obscured.
- Given that a basic UI for login is built correctly, when a user enters their obscured password, they should not be able to copy/paste the obscured text.

User Story #9

As a user, I would like to be able to log out.

#	Description	Estimated Time	Owner
1	Adjust each UI page to include a log-out button	4 Hr	Michael
2	Adjust UI to redirect user to login page upon logging out	4 Hr	Michael
3	Communicate with Firebase Authentication to de-Authenticate the session	2 Hr	Michael
4	Test that logout button works as intended	1 Hr	Michael
5	Test that logging out redirects to login page	1 Hr	Michael
User Story #9 Total Hours		12 Hr	

Acceptance Criteria

- Given that the UI for the logout button works correctly, when a user clicks it, they should be redirected to the login page.
- Given that the UI for the logout button works correctly, when a user clicks it, they should be locked out of their account until they log back in.
- Given that Firebase Authentication is configured correctly, when a user logs out, they should no longer be able to access the rest of the site.

User Story #10

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

#	Description	Estimated Time	Owner
1	Create UI for user to change their password for their account through login screen	4hr	Aaron
2	Incorporate Firebase Authentication functions to ensure user can change password and to change the password	3hr	Aaron
3	Implement Firebase function to send email to user for changing password	4hr	Aaron
4	Test to ensure that an email was sent to user to change password	1hr	Aaron
5	Test to ensure that the user can log in with new password	1hr	Aaron
6	Test to ensure that the user cannot log in with the old password	1hr	Aaron
7	Test to ensure that the password has been updated in Firebase	1hr	Aaron
User Story #10 Total Hours		15 Hours	

Acceptance Criteria

- Given that a UI for reset password is built correctly, when a user navigates to the reset password page, there should be a field to enter a password.
- Given that Firebase Authentication is integrated correctly, when a user enters valid information, an email should be sent to the email address of the user.
- Given that the user clicks on the link provided in the reset password email, when the user navigates to the change password page, there should be a field to enter a new password.
- Given that Firebase Authentication is integrated correctly, when a user enters valid information, those changes should be reflected in the Firebase Authentication module.
- Given that Firebase Authentication is integrated correctly, when a user enters invalid information to login, an error message should occur saying that the information entered is invalid.
- Given that the user has entered the valid information and the changes were made in the Firebase Authentication module, when the user navigates to the login page, that user should be able to login with the new password.

User Story #16

As a user, I would like to be able to create public personal events.

#	Description	Estimated Time	Owner
1	Develop UI for create personal events page	4 Hr	Bryan
2	Communicate with Firestore Database to store event information	3 Hr	Bryan
3	Test to ensure the UI responds to changes in the Firestore Database	1 Hr	Bryan
4	Test input validation	1 Hr	Bryan
5	Test to ensure the UI redirects to the appropriate page	1 Hr	Bryan
User Story #16 Total Hours		10 Hours	

Acceptance Criteria

- Given that a UI for events page is built correctly, when a user navigates to the events page, there should be a link to create an event.
- Given that a UI for the create event page is built correctly, when a user attempts to generate an event, they should be redirected to a form to enter information for their event.
- Given that the communication with Firestore Database is configured correctly, when a user creates a new event, the newly-created event object should now be visible in the database.
- Given that the UI for creating personal events is implemented correctly, when a user successfully creates a new event, the user should be notified that the event was successfully created.
- Given that the UI for the events page is built correctly, when a user inputs invalid information into the events form, they should be notified that bad data was supplied.

User Story #17

As a user, I would like to be able to edit an event I created.

#	Description	Estimated Time	Owner
1	Develop UI for edit events page	4 Hr	Yu-Lun
2	Communicate with Firebase Database to update the events information	3 Hr	Yu-Lun
3	Test to ensure the UI responds to changes in the Firestore Database	1 Hr	Yu-Lun
4	Test input validation	1 Hr	Yu-Lun
5	Test to ensure the UI redirects to the appropriate page	1 Hr	Yu-Lun
User Story #16 Total Hours		10 Hours	

Acceptance Criteria:

- Given that a UI for events page is built correctly, when a user navigates to the events page, there should be a component that allows a user to edit an event.
- Given that a UI for the edit events page is built correctly, when a user attempts to edit an event, they should be prompted with a form that allows them to update the fields of their event.
- Given that the communication with Firestore Database is configured correctly, when a user updates an existing event, the relevant fields should be changed in the database.
- Given that the communication with Firestore Database is configured correctly, when a user updates an existing event, fields that weren't updated should not be changed in the database.
- Given that a basic UI for the edit events page is built correctly, when a user finishes editing an event, the user should be notified that the event was successfully (or unsuccessfully) edited.

User Story #19

As a user, I would like to be able to view a list of Purdue events.

#	Description	Estimated Time	Owner
1	Develop UI for events landing page	4 Hr	Raziq
2	Pull events from Firebase by pages	3 Hr	Raziq
3	Implement dynamic events rendering when there are too many of them	2 Hr	Raziq
4	Integrate the UI with actual data	2 Hr	Raziq
5	Test to ensure that the UI displays mock data correctly	1 Hr	Raziq
User Story #19 Total Hours		12 Hours	

Acceptance Criteria

- Given the events landing page is built correctly, when a user navigates to the page, they will see a list of events along with some short information about them.
- Given the UI integration with Firebase is done correctly, when a user navigates to the events landing page, they will be able to see actual events that have been added to the database.
- Given the dynamic rendering functionality is done correctly, when a user visits the events landing page, only a handful of events will be displayed the first time the page loads.
- Given the dynamic rendering functionality is done correctly, when a user visits the events landing page, the user will have the option to load/view more events.
- Given the UI integration with Firebase is done correctly, when a user navigates to the events landing page, they will not be able to see events that have already happened.

User Story #21

As a user, I would like each event to have an information page.

#	Description	Estimated Time	Owner
1	Develop UI for event information page	4 Hr	Raziq
2	Pull event information from event object in Firestore Database and display to UI	2 Hr	Raziq
3	Test to ensure that the UI displayed mock data correctly	1 Hr	Aaron
User Story #21 Total Hours		7 Hours	

Acceptance Criteria

- Given the event information page is implemented correctly, when a user navigates to the event page, the user will be able to see the event title.
- Given the event information page is implemented correctly, when a user navigates to the event page, the user will be able to see the event description.
- Given the event information page is implemented correctly, when a user navigates to the event page, the user will be able to see the event duration.
- Given the event information page is implemented correctly, when a user navigates to the event page, the user will be able to see the name of the entity posting the event.
- Given the event information page is implemented correctly, when a user navigates to the event page, the user will be able to see the event date.

User Story #22

As a user, I would like to be able to filter some events based on some criterias.

#	Description	Estimated Time	Owner
1	Develop the UI components for event filtering	1 Hr	Raziq
2	Integrate the UI components with the events listing page and make the page responsive	3 Hr	Raziq
3	Implement the filter functionality	2 Hr	Raziq
4	Test the filter functionality works	2 Hr	Aaron
User Story #22 Total Hours		8 Hours	

Acceptance Criteria

- Given the event filtering components are integrated to the page correctly, when the user navigates to the event listing page, there will be some event filtering options.
- Given the event filtering feature is done correctly, when the user filters the events for some criteria, only events that meet the criteria will appear.
- Given the event filtering UIs are integrated to the page correctly, when the user opens the page from any device, the rendering of the page should still be responsive.

User Story #27

As a user, I would like to be able to create a new club.

#	Description	Estimated Time	Owner
1	Develop UI for a new club page	4 Hr	Bryan
2	Communicate with Firestore Database functions to add club pages	4 Hr	Bryan
3	Test to ensure the club is updated in Firestore Database	1 Hr	Bryan
4	Test to ensure the UI redirects to an appropriate page	1 Hr	Bryan
5	Test input validation	1 Hr	Bryan
6	Test to ensure the UI responds to changes in the Firestore Database	1 Hr	Bryan
User Story #27 Total Hours		12 Hours	

Acceptance Criteria

- Given that a UI for clubs page is built correctly, when a user navigates to the clubs page, there should be a link to create a club.
- Given that a UI for the create club page is built correctly, when a user clicks on the edit club link, they should be redirected to a form that contains the relevant information input fields.
- Given that the communication with Firestore Database is configured correctly, when a user adds a new club, the newly-created club object should now be visible in the database.
- Given that a UI for the create club page is built correctly, when a user successfully creates a club, they should be notified that their new club page has been successfully generated.
- Given that a UI for the create club page is built correctly, when a user inputs invalid information, they should be informed that bad data was supplied.

User Story #48

As a user, I would like to be able to create sell listings on the marketplace.

#	Description	Estimated Time	Owner
1	Develop UI for create sell-listing page	4 Hr	Bryan
2	Develop UI for individual sell-listings	2 Hr	Matt
2	Communicate with Firestore Database functions to add club pages	3 Hr	Bryan
3	Test to ensure data is stored correctly in the Firestore Database	1 Hr	Bryan
4	Test to ensure the UI responds to changes in the Firestore Database	1 Hr	Bryan
5	Test input validation	1 Hr	Bryan
6	Test to ensure the UI redirects to the appropriate page	1 Hr	Matt
User Story #48 Total Hours		13 Hours	

Acceptance Criteria

- Given that a UI for the marketplace page is built correctly, when a user navigates to the marketplace page, there should be a link to create a new sell-listing.
- Given that a UI for the create sell-listing page is built correctly, when a user clicks the link to create a sell-listing, that user should be redirected to a page that contains the relevant information input fields.
- Given that the communication with the Firestore Database is configured correctly, when a user creates a new sell listing, the newly-created sell-listing object should now be visible in the database.
- Given that the UI is implemented correctly, when a user creates a new sell-listing, the user should be notified that the page was created correctly.
- Given that the UI is implemented correctly, when a user inputs invalid information, they should be informed that bad data was supplied.

User Story #49

As a user, I would like to be able to see a list of sell listings on the marketplace.

#	Description	Estimated Time	Owner
1	Develop UI for the marketplace page	5 Hr	Matt
2	Connect UI to sell-listing collection in Firestore Database	3 Hr	Matt
3	Connect individual sell-listing tiles to their expanded individual sell-listing page	3 Hr	Matt
4	Test to ensure the UI responds to changes in the Firestore Database	1 Hr	Matt
5	Test to make sure UI works as expected	1 Hr	Matt
User Story #49 Total Hours		13 Hours	

Acceptance Criteria

- Given that a UI for the marketplace page is built correctly, when a user navigates to the marketplace page and there are no sell listings in the database, the marketplace page should indicate that there are no available listings.
- Given that a UI for the marketplace page is built correctly and the database is correctly configured, when there are sell listings in the database, the marketplace page should show a list of sell-listings.
- Given that the connection between the marketplace and sell-listing is correct, when a user creates a new sell listing, the newly-created sell-listing should now be visible on the marketplace.
- Given that the connection between the marketplace and sell-listing is correct, when a user clicks on an individual sell-listing, the page should redirect to that specific sell-listing's page.

User Story #56

As a user, I would like to be able to view class forums.

#	Description	Estimated Time	Owner
1	Develop UI for the class forum landing page	5 Hr	Yu-Lun
2	Develop UI for specific class pages	5 Hr	Yu-Lun
3	Implement Firestore Database functions to add Class pages	3 Hr	Yu-Lun
4	Test to ensure that modifications to class pages (including addition and deletion of pages) are reflected in Firestore Database	1 Hr	Yu-Lun
5	Test to ensure the UI responds to changes in the Firestore Database	1 Hr	Matt
6	Test to ensure the UI redirects to the appropriate pages	2 Hr	Yu-Lun
User Story #56 Total Hours		17 Hours	

Acceptance Criteria

- Given that the UI for the class forum landing page is built correctly, when a user navigates to the class forum landing page, they should be met with a dynamically-rendered page.
- Given that the UI for the class forum landing page is implemented correctly, when a user clicks on a specific class, they should be redirected to that specific class' forum.
- Given that the UI for the specific class pages is implemented correctly, when a user navigates to a specific class page, they should be met with posts relevant to that class and a sidebar containing additional information.
- Given that the functions to modify the Firestore Database are correctly implemented, when changes are made to class pages, those changes should be reflected within the Firestore Database console.
- Given that both UI's are built correctly, when changes are made to the Firestore Database, the new information should replace the old information upon a page reload.

User Story #57

As a user, I would like to post on class forums.

#	Description	Estimated Time	Owner
1	Develop UI for create post on the class forum page	4 Hr	Matt
2	Implement Firestore Database Functions to add posts	2 Hr	Bryan
3	Update post information in Firebase Authentication	2 Hr	Bryan
4	Test to ensure the post is updated in Firebase Authentication	1 Hr	Bryan
5	Test input validation	1 Hr	Bryan
6	Test to ensure the UI shows the newly created post	1 Hr	Bryan
User Story #57 Total Hours		11 Hours	

Acceptance Criteria

- Given that a UI for a class forum page is built correctly, when a user navigates to the class forum page, there should be a link to create a new post.
- Given that a UI for the create new post page is built correctly, when a user clicks on the link to make a new post, they should be redirected to a page to enter information for their new post.
- Given that a Firestore Database functions to add posts are implemented correctly, when a user adds a new post, the newly-created post object should now be visible in the database.
- Given that the post creation feature was implemented correctly, when the user created a new post, the user should be notified that their post creation was successful.
- Given that input validation works as intended, when the user enters invalid input for a post, the user should be notified that their post creation was unsuccessful.

User Story #68

As a user, I would like to be able to navigate to various pages on this website from a landing page.

#	Description	Estimated Time	Owner
1	Research and develop UI for responsive navigation bar	4 Hr	Raziq
2	Research React router	2 Hr	Raziq
3	Develop UI for landing page content	5 Hr	Raziq
4	Test that page navigation works	2 Hr	Raziq
5	Integrate summary data from other pages into the landing page.	3 Hr	Raziq
User Story #68 Total Hours		16 Hours	

Acceptance Criteria

- Given the landing page is implemented correctly, when the user navigates to the page, there will be options to navigate to other pages in this website.
- Given the navigation bar is implemented correctly, when the user opens the website from any device, the rendering of the navigation bar should be responsive.
- Given the UI for the landing page content is implemented correctly, when the user navigates to the page, there will be some cards that show information from a specific page.
- Given the landing page content is implemented correctly, when the user opens the page from any device, the page rendering should be responsive.

Current Total: 235 Hours

Name	Estimated Time
Aaron Cox	40 Hrs
Yu-Lun Gao	40 Hrs
Michael Maganini	38 Hrs
Raziq Ramli	40 Hrs
Matthew Story	38 Hrs
Bryan Wang	39 Hrs

Remaining Backlog

Profile

- ~~1. As a user, I would like to create a new account with my own email address.~~
- ~~2. As a user, I would like to edit my profile.~~
- ~~3. As a user, I would like to delete my account.~~
- ~~4. As a user, I would like to have a unique username.~~
5. As a user, I would like to have a profile with a brief bio and image.
- ~~6. As a user, I would like to be able to change my password for my account.~~
- ~~7. As a user, I would like to be able to log in.~~
- ~~8. As a user, I would like my password to be obscured when I type it to log in.~~
- ~~9. As a user, I would like to be able to log out.~~
- ~~10. As a user, I would like to reset my password if I forget it.~~
11. As a user, I would like to check other user profiles.
12. As a user, I would like to choose between light and dark mode (if time allows).
13. As a user, I would like to hide my profile from users (if time allows).
14. As a user, I would like to choose specific features from the site as part of my homepage (if time allows).

Events Board

15. As a user, I would like to be able to add events for my Purdue club if I have club permission.
- ~~16. As a user, I would like to be able to create public personal events.~~
- ~~17. As a user, I would like to be able to edit an event I created.~~
18. As a user, I would like to be able to delete an event I created.
- ~~19. As a user, I would like to be able to view a list of Purdue events.~~
20. As a user, I would like to be able to view public personal events (if time allows).
- ~~21. As a user, I would like each event to have an information page.~~
- ~~22. As a user, I would like to be able to filter some events based on some criterias.~~
23. As a user, I would like to be able to RSVP to club events (if time allows).
24. As a user, I would like to see past events be hidden from my feed (if time allows).
25. As a user, I would like to see club events I create added to that club page if I have permission.
26. As a user, I would like to see the events posted on the boiler links website to be listed on this application as well (if time allows).

Club Pages

- ~~27. As a user, I would like to be able to create a new club.~~
28. As a user, I would like to be able to manage user permissions of the club if I am a creator.
29. As a user, I would like to be able to add a description for my club.
30. As a user, I would like to be able to add contact information for the club administrators.
31. As a user, I would like to be able to view a list of clubs available at Purdue.
32. As a user, I would like to be able to search the list of Purdue clubs.
33. As a user, I would like to view relevant information about clubs I'm interested in.

- 34. As a user, I would like to be able to edit a club's information if I have necessary permission.
- 35. As a user, I would like events that I make to be automatically added into the events board.
- 36. As a user, I would like to have access to edit the events that automatically added into the events board.
- 37. As a user, I would like to have access to delete the events that automatically added into the events board.
- 38. As a user, I would like to see a list of events that a club is holding on their page.
- 39. As a user, I would like to see the clubs created in the boiler links official website to be listed on this application as well (if time allows)

Living

- 40. As a user, I would like to be able to view a labeled map with bus stops and routes that run through the Purdue campus.
- 41. As a user, I would like to be able to view a bus route's next stop.
- 42. As a user, I would like to be able to view a list of all Purdue dining halls.
- 43. As a user, I would like to be able to view daily menus for each dining hall.
- 44. As a user, I would like to be able to check how crowded each dining hall is.
- 45. As a user, I would like to be able to check how crowded different sections of the Purdue CoRec are.
- 46. As a user, I would like to be able to see which washing machines are currently in use in each dormitory.
- 47. As a user, I would like to be able to see which dryers are currently in use in each dormitory.

Marketplace

- ~~48. As a user, I would like to be able to create sell listings on the marketplace.~~
- ~~49. As a user, I would like to be able to see a list of sell listings on the marketplace.~~
- 50. As a user, I would like to edit my sell listing on the marketplace.
- 51. As a user, I would like to delete my sell listing on the market place.
- 52. As a user, I would like to be able to set a sell price and item description for my sell listings.
- 53. As a user, I would like to be able to list my contact information to potential buyers.
- 54. As a user, I would like to be able to include an image of the item I am selling in my sell listings (if time allows).
- 55. As a user, I would like to be able to message other students on the marketplace about their listings (if time allows).

Class Forum

- ~~56. As a user, I would like to be able to view class forums.~~
- ~~57. As a user, I would like to post on class forums.~~
- 58. As a user, I would like to edit my post.
- 59. As a user, I would like to delete my post.

- 60. As a user, I would like to comment on posts.
- 61. As a user, I would like to comment on comments.
- 62. As a user, I would like to edit my comment.
- 63. As a user, I would like to delete my comment.
- 64. As a user, I would like to filter posts by popularity or recency (if time allows).
- 65. As a user, I would like to join class forums (if time allows).
- 66. As a user, I would like to leave class forums (if time allows).

Other

- 67. As a user, I would like to install this web application on my desktop and mobile devices as a progressive web app (if time allows).
- ~~68. As a user, I would like to be able to navigate to various pages on this website from a landing page.~~