



Sprint 1 Planning Document

Abhijeet CHAKRABARTI
Brian QUINN
Sanjit SAMA
Shengqi WANG

TEAM #16

January 31, 2019

1 | Sprint Overview

This sprint, being our first sprint aims to set up the basic framework we need to develop SeekMyTeam. A lot of this will involve having our more experienced team members teach and guide the others in the basics of web development and getting a good development work-flow set up. This will also include setting up the hosting architecture for the SeekMyTeam website on AWS, making a basic Front-End and setting up the boilerplate server code for the web app. A significant time for the sprint will also be dedicated to setting up the architecture of the app and ensuring all dependencies are rendered correctly. We will also be setting some time aside to set up a CI/ CD build environment where our commits to the master branch passes through a set of regression and integration tests before getting deployed.

Scrum Master Abhijeet Chakrabarti

Team Leader Brian Quinn

Meeting Plan Meeting on Fridays from 4 PM - 5 PM in FUSE as it is the closest to most of the team members.

Risks And Challenges Some of the members of our team are not yet familiar with web development practices or skilled in the languages required to create our web-app. For our team, a large part of Sprint 1 will be attempting to bring all of the members of our team up to speed with the technical skills required to complete this project.

2 | Current Sprint Details

User Story #1

As a user, I want to be able to register for an account.

#	Description	Time	Team	Owner
1	Design registration page	5 hrs	Front-End	Sanjit
2	Set up Angular framework and connect it to AWS	6 hrs	Front-End	Sanjit
3	Implement registration UI	8 hrs	Front-End	Shengqi
4	Implement Front-End logic for registration process	5 hrs	Front-End	Abhijeet
5	Configure database to store users	5 hrs	Back-End	Abhijeet
6	Configure database to communicate with server	8 hrs	Back-End	Brian
7	Configure server logic to interact with database	5 hrs	Back-End	Abhijeet
8	Create a record for the user in the database when they register	4 hrs	Back-End	Abhijeet
9	Handle two step email verification	6 hrs	Back-End	Brian
10	Host and Deploy webpage on AWS server with elastic IP	12 hrs	Back-End	Abhijeet
11	Unit testing for user sign up to test successful registration.	8 hrs	Testing	Shengqi

Acceptance Criteria

- *Given* a user goes to the webapp, *when* they hit the sign-up button, *then* they should be brought to a page where they can enter their registration details.
- *Given* a user has entered their registration information, *when* they submit their user registration, *then* they should be emailed a verification code which they will input back into the application.
- *Given* a user choosing to register, *when* the user enters the incorrect verification, *then* an error message should be displayed to the user.
- *Given* a user choosing to register, *when* they enter the correct verification code, *then* a record representing them should be added to the database.

Testing

- Test sign-up button functionality by hitting sign-up button and verifying if it redirects user to the registration page where the user can enter his/her details.
- Test if user can sign-up with the same email twice and verifying an error message.
- Test if our server can generate a unique code for the user to finish the registration process.
- Test if the user information is added to the database by running database query.
- Test if the user-interface shows error to the user if the user has entered an incorrect verification code.

As a user, I want to be able to delete my account.

#	Description	Time	Team	Owner
1	Design account deletion UI	2 hrs	Front-End	Sanjit
2	Implement account deletion UI	3 hrs	Front-End	Sanjit
3	Implement Front-End logic for deleting an account	3 hrs	Front-End	Sanjit
4	Implement server endpoints for deleting an account	4 hrs	Back-End	Brian
5	Handle deletion of all user created entities upon account deletion	8 hrs	Back-End	Brian
6	Add unit testing for removing an user to test successful deletion	4 hrs	Testing	Shengqi

Acceptance Criteria

- *Given* that a user has an account, *when* the user chooses to delete their account, *then* the user's record should be deleted from the database.
- *Given* that a user has an account, *when* the user chooses to delete their account, *then* other postings and data associated with the user should also be deleted from the database.
- *Given* that a user has an account, *when* the user chooses to delete their account, *then* other users will be redirected to a 404 page if they try to access the profile page.

Testing

- Test delete functionality by hitting delete button and verifying if it opens a confirm dialog box where the user can choose yes or no.
- Test if the user information is removed from the database by running database query.
- Test if the app redirects users to a 404 page if they try to access the profile page of a deleted page

As a user, I want to be able to log in to my account.

#	Description	Time	Team	Owner
1	Design the log-in page UI	4 hrs	Front-End	Sanjit
2	Implement log-in page UI	3 hrs	Front-End	Shengqi
3	Implement Front-End logic for Login	3 hrs	Front-End	Shengqi
4	Implement server endpoints for log-in authentication	5 hrs	Back-End	Abhijeet
5	Handle authentication and session creation on the back-end	5 hrs	Back-End	Brian
6	Make session handling code to handle timeouts	3 hrs	Back-End	Brian
7	Unit testing for log-in procedure	2 hrs	Testing	Brian

Acceptance Criteria

- *Given* that a user has already signed up, *when* they go to the Login page, *then* they should be able to enter log-in credentials.
- *Given* that a user has entered their log-in credentials, *when* they click the log-in button, *then* they should either be redirected to the home page or given an error for incorrect credentials.
- *Given* that a user has logged in successfully, *when* the home page loads, *then* they should be able to see their profile icon on the nav-bar.
- *Given* a logged in user, *when* they have been inactive for a certain period of time, *then* their session should expire and they should be required to log back in to use the application.

Testing

- Test log-in functionality by hitting log-in button and verifying if it redirects to the home page upon a successful log-in.
- Test if the user-interface hides the characters entered under the password field.
- Test if the user-interface shows error to the user if the user has entered incorrect credentials.
- Test if the right user credentials are being verified per the database query.
- Test if the user is getting a unique session ID once they logged in and verify that it is stored in the database.

As a user, I want to have a profile page.

#	Description	Time	Team	Owner
1	Design profile page	3 hrs	Front-End	Sanjit
2	Implement profile page UI	8 hrs	Front-End	Sanjit
3	Implement Front-End logic for the profile page	4 hrs	Front-End	Shengqi
4	Implement server endpoints for fetching profile data	2 hrs	Back-End	Brian
5	Populate webpage with information from the database	4 hrs	Front-End	Abhijeet

Acceptance Criteria

- *Given* a user is logged in, *when* they click the link to display their profile, *then* the user will be able to view the contents of their profile.
- *Given* a user is on their profile page, *when* the user presses edit on any editable fields, *then* the user will be able to change the content of that field.
- *Given* a user is registered, *when* the user finishes filling all the information under profile page, *then* the user's records can be accessed from the database.

Testing

- Test if the user is able to view the contents of their profile page when the user clicks the link to display his/her profile.
- Test if the all the information added by an user under their profile page is added to the database and linked to the user's unique id.
- Test if the user is able to edit the *editable fields* under the user's profile page.
- Test if the fields edited by the user are also updated in the database.

User Story #5

As a user, I want to be able to have people view my profile.

User Story #6

As a user, I want to be able to view another user's profile.

#	Description	Time	Team	Owner
1	Implement Front-End logic to populate the selected profile page	3 hrs	Front-End	Sanjit
2	Configure server routes to map profile pages based on user name	6 hrs	Back-End	Shengqi
3	Add unit testing for getting the right queries for the current profile	4 hrs	Testing	Shengqi

Acceptance Criteria

- *Given* a user who wants to view someone's profile, *when* the user navigates to the url of that profile page, *then* the user will be directed to the profile page of the other user.
- *Given* a user who is logged in, *when* the user tries to view another user's profile, *then* he/she will be able to read only and not edit any information of that user.
- *Given* a user who wants to view someone's profile, *when* they are redirected to the url of that profile page, *then* the user will be able to see that user's description and skills.

Testing

- Test if the user is able to view the contents of the selected profile page when the user clicks the link to display another user's profile.
- Test if the user selected profile page will load the right user's information per the database query.
- Test if the user viewing another user's profile is unable to edit any fields under that user's profile page.

User Story #9

As a user, I want to be able to write a description of my experience.

User Story #10

As a user, I want to be able to change my description of my experience.

#	Description	Time	Team	Owner
1	Populate profile with updated description upon saving edits	1 hr	Front-End	Sanjit
2	Implement server endpoints to save description to the database	2 hrs	Back-End	Brian
3	Unit testing for adding/editing the description	2 hrs	Testing	Sanjit

Acceptance Criteria

- *Given* a user who is logged in, *when* the user opens their profile page *then* the user is able to add a description of their experience.
- *Given* a user who is logged in, *when* the user opens their profile page *then* the user is able to change the description of their experience that he/she has previously.
- *Given* a user is registered, *when* the user finishes filling his/her description, *then* the user's description of their experience can be accessed from the database.

Testing

- Test if the user can write a description for their profile page and have the user-interface display the information.
- Test if the user can edit their description for their profile page and have the user-interface update the information accordingly.
- Test if the edit description will save the updates to the database.

Personal Hours

- Abhijeet Chakrabarti: 40 hours
- Brian Quinn: 40 hours
- Sanjit Sama: 40 hours
- Shengqi Wang: 40 hours

Team Hours: 160 hours

3 | Remaining Backlog

1. ~~As a user, I want to be able to register for an account.~~
2. ~~As a user, I want to be able to delete my account.~~
3. ~~As a user, I want to be able to log in to my account.~~
4. ~~As a user, I want to have a profile page.~~
5. ~~As a user, I want to be able to have people view my profile.~~
6. ~~As a user, I want to be able to view another user's profile.~~
7. As a user, I want to be able to upload a profile picture.
8. As a user, I want to be able to change my profile picture.
9. ~~As a user, I want to be able to write a description of my experience.~~
10. ~~As a user, I want to be able to change my description of my experience.~~
11. As a user, I want to be able to create a list of skills.
12. As a user, I want to be able to change my list of skills.
13. As a user, I want to be able to display my social media links for other users to view.
14. As a user, I want to be able to change my social media links for other users to view.
15. As a user, I want to be able to post a project that I am looking for collaboration with.
16. As a user, I want to be able to edit my project posting.
17. As a user, I want to be able to delete my project posting.
18. As a user, I want to be able to view projects posted by other users.
19. As a user, I want to be able to apply to projects posted by other users.
20. As a user, I want to be able to find contact information for the user posting the project.
21. As a user, I want to be able to view a list of projects that I am a part of.
22. As a user, I want to be able to search for projects by posting name.
23. As a user, I want to be able to view all of the users available on the platform.
24. As a user, I want to be able to filter user results by certain tags.
25. As a user, I want to be able to search for other users by name.
26. (If time permits) As a user, I want to be able to view a list of projects I have applied to.
27. (If time permits) As a user, I want to be able to apply to a project by clicking a button that will send an automatic email notification to the poster.

28. (If time permits) As a user, I want to be able to invite a user to apply to a project by clicking a button that will send an automatic email notification to the user.
29. (If time permits) As a user, I would like to have multiple owners associated with a project.
30. (If time permits) As a user, I would like to have multiple owners associated with a project.