

CSCI 3308

Team Number: 105-4

Team Name: Team Rad

Members:

Colton O'Connor

Ryan Jones

Haifeng Jiang

Reid Pritchard

Cade Gorman

Milestone 5

All requests will be sent using Mocha and Chai libraries. The tests will be separated into a tests.js file to be run with the npm test command.

1. Test Plan 1:

a. User account creation including selection of interests

- i. A test POST request will be sent to the server. The data to be sent will include name, username, email, password. Multiple test cases with repeated emails will be sent to verify user creation only if the user is new. The test users will then be cleaned from the database.

ii. Steps:

1. User will navigate to the landing page
2. They will click on the "Sign In" button on the Nav bar
3. The user will click the link in the pop up menu stating "Create Account"
4. They will be taken to a sign up page that asks for the following inputs:
 - a. Name
 - b. Username
 - c. Email
 - d. Password
 - e. Password Confirmation
5. If username and email is unique to the user, the account will create.
6. Once the account creation is confirmed, the user will then sign into account at the landing page
7. On a successful login the user will be redirected to their profile page
8. The user will choose the link "Select Interests"
9. This will take the user to the interests select page where they will be able to check the interests that suit them.
10. On the profile page, the user will be able to edit their interests by selecting the "Edit Interests" link.

iii. Final Thoughts:

1. This element is important in our project because it shows the ability for the user to create an account specific to them, as well as select their interests and verify that the software stores this information properly.

2. Test Plan 2:

- a. Event comment section entry
 - i. A test POST request will be sent to the server. There will be multiple ‘agents’ sending requests to verify only logged in users can post. The data that will be sent will include a message and username as well as an event id that will be in the url of the request. A GET query to the event will then be sent to verify the comment was submitted successfully.
 - ii. Steps:
 - 1. The user will navigate to a specific event page
 - 2. Once on this page the user will scroll to the bottom where the comment section is present
 - 3. The user will then type out a message in the window labeled “Comment”
 - 4. The user will verify the the comment section has updated with the message they typed out in the window
 - iii. Final Thoughts:
 - 1. This test is important because it works as the main communication between users. With this we can ensure that discussion can be made about specific events happening in the area, and the statements made can directly be labeled with the user's name.
- 3. Test Plan 3:
 - a. Event suggestion and indication of attending
 - i. A test GET request will be sent to the “/” route. This will check if the user is logged in. If not the server will respond with a default home page listing all the events in the database. If the user is logged in, it will respond with recommended events based on the user’s preferences. The events the user is already “attending” will not be shown. They will be shown in the table listed under “Events I’m Attending”. Once a suggested event is marked as “Attending” it will be added to this list.
 - ii. Steps:
 - 1. The user will navigate to their unique profile page
 - 2. At the bottom of the page the user can observe the carousel of events scroll through, suggesting events that pertain to their interests
 - 3. The user will select the intended event that they wish to attend
 - 4. This will navigate the user to the event page where they can select “Attending” at the top of the event.
 - 5. This event page should hold date, time, location, and a short description of the event
 - 6. Once the user selects “Attending” on the event, they will navigate back to their profile page
 - 7. On the “Events I’m attending” tabel, they should see that the new event has been added.
 - iii. Final Thoughts:
 - 1. This, again, is an important test case because it ensures that the user will have quick access to events they intend to attend. This is the core

purpose of the project, to allow users easy access to events they might miss out on. To have events suggested, as well as an easy way to identify them once marked, is a key part to achieve our finished product.