Milestone Submission 5
Group 101-6
Mason Dobbins
Andrew Hahn
Jared Laroco
John Salame
Ben Wright

1. Search and Display

Test Case 1:

- User presses submit with no other input. The search function should not attempt to generate results in this case. We assume that this was accidental input from the user and with no information to work from do not make an attempt to search.

Test Case 2:

- User provides skills, interests, languages or some combination thereof but no text input. The search function finds matches to this combination of tags and presents matching results in order of most exact fit. The user has indicated general areas of interest in this case but no specifics. We present any results that exactly match their selections, then results that match some but not all selections.

Test Case 3:

- User provides text search input but no tag selections. We search the database for exact matches of the project title, then exact matches within the description, then fuzzy matches on the title and finally fuzzy matches within the description. We display results to the user in that order.

Test Case 4:

- User provides both text and tag information to the search function. We follow the same protocol as with text only search but weight ties based on matching to tag input.

2. User Authentication

Test Case 1:

- Unregistered user attempts to login. The project should not allow users who are not registered to login without registering, because the database would have no user information on these individuals. Users who want to use the application will have to register with their email and password so that they have an account to login with.

Test Case 2:

- User attempts to register account for email that is already registered to a user. If a user attempts to do this, an error message should be presented to a user if this is attempted, and it should alert them that an account is already registered with the email they are attempting to use.

Test Case 3:

- Invalid email or password. If a user attempts to login with a invalid email or password, that does not match with the database, then the program should let the user know that this is the case. The user should only be able to login with valid login credentials, and should not be able to access accounts without the valid password for that account.

3. Project Submission

Test Case 1:

- Unauthenticated user attempts to submit a project. The application should not allow the project to be submitted, and repudiate the user with a message stating that they must use a valid login to submit a project. Failure conditions for this case would be if the project can be submitted without authentication, or if the project is not able to be submitted but the user is not repudiated.

Test Case 2:

- Authenticated user submits a valid project. Validate that the project is correctly submitted to the SQL database via the integration layer. To test this, submit a valid project under an authenticated user account. Correct submission to the database can be tested either within the postgres database by viewing the projects table, or, under the assumption that search is working as intended, search for the project with matching criteria and see if that project is displayed properly with all fields containing the full information submitted.

Test Case 3:

- User attempts to submit a project with empty or invalid fields. Program should not allow the project to be submitted to the database and should repudiate the user with the reason their submission is invalid.