Quality Control Team Report
8-1-2019

Group Members:

Aleksey Titov
Steven Ginsberg
Jerfyn Rosario
Kenny Shi
Kibria Mohammed
Vladislav Gladyshev

What has your team accomplished in the past week?

Over the past week, we have been testing the Issue system from GitHub and verified that an email is sent to the users, at least for most teams. We will use this as a sort of ticket system for all the teams once testing is underway. In case certain teams are unable to receive a notification from this, we'll send a message through either Slack or directly through the Team Members Google Docs.

During the week, some teams have been uploading working code for various features of the website project. The Database team, for instance, has uploaded functions involving adding, removing, and finding users. We have begun reviewing the code and checking for any errors while planning on how to test this code, whether through scripts or through our own program. The Front End Team have uploaded HTML templates to the GitHub repo, and the Back end team has been updating their code regularly.

Discuss with your team as a group what should be tackled for the upcoming week.

Now that many of the groups have been uploading their progress so far, we are able to start doing some testing on our end. Each team can let us know if a code they have uploaded is non-functional in case it is still under development.

For the Front End Team, we will download the HTML and style sheets they have uploaded to GitHub and deploy it to a page. We will be checking whether or some free standing UI works and if a user would have trouble with navigation.

For the Database Team, we are currently discussing how to test out the code they have uploaded. What we are leaning towards is creating our own program to test out whether or not their database works. We will ask the Database team for additional input on their code and how we may test it.

We need to get details db connection, tables and attributes and check to see whether these fields matches with the UI functionality. Also, we might check if the user input is synchronized in the DB successful or not. We need the user stories from the database,

front-end and backend people and also the functionality that they are implementing against services and database.

For the Back End Team, we have downloaded Postman to each of our machines as suggested by them in order to test out their code. We will begin testing their code using Postman over the weekend.

The backend team should provide the domain and end point and we can perform crud operation on the service end point. Moreover, the backend service should be running on some server or in our local machine with the code inorder for us to test the service end point. From user story we can derive the acceptance criteria for the stories and what are the expected results (ex- hard assertion or soft-assertion) when we consume the services as a client. For example, developer should implement what should the get,post,delete,put should display after sending request to service either via ui or endpoint in postman. We can derive the test case fro each end point and get json response and validate the data directly hitting the db. We may also write the test case to check the end point json data is match with the backend db and frontend. Also, we can check whether the end point is stateless or stateful.

We will double check on what each team have uploaded and see if we can combine them into a usable alpha site. If this is possible, we can perform a more in depth review of the site and provide better information about any issues to each team. We can also use this as the alpha version of the site and show the rest of the teams how the alpha site looks like.
