SW Engineering CSC 648 Fall 2019

Section 2

Team 8

December 12, 2019

Milestone 4

Team:

- Oleksii Butakov ---------- Team Lead / Front-end (obutakov@mail.sfsu.edu)
- Andrew Copas -------- GitHub Master / Front-end Lead
- Alex Lee ---------- Front-end
- Emanuel Saunders ---------- Back-end Lead
- Kevin Truong ---------- Back-end
- Tim Lei ---------- Back-end

Content

1) Product Summary ---------- 1
2) Usability Test Plan ---------- 4
3) QA Test Plan ---------- 5
4) Code Review ---------- 6
5) Self-check on Best Practices for Security ---------- 6
6) Self-check: Adherence to Original Non-Functional Specs ---------- 7

1) **Product summary (e.g. how would you market and sell your product – about ½ page)**

- Name of the product - **SFSUQuad**
- Explicit **itemized list of ALL major committed functions, 1 line per function (your FINAL P1 functions for which you will be graded) your team shall actually deliver (and test for). This is your FINAL functional commitment e.g. failure to deliver on some of P1 functions results in reduced grade. Please write it in the list format (each item 1 line of text) so it is easy to check. List of functions to be written in regular English and not like specs. You can stay with the list recommended in M3**

Unregistered Users:

1. Unregistered users shall be able to browse the posts.

    1.1 Unregistered users shall be able to search all approved posts

    1.2 Unregistered users shall be able to sort the search results by price

2. Unregistered users shall be able to access posts details.

3. Unregistered uses shall be able to register.

Registered Users:

7. +Functions of unregistered Users.

8. Registered users shall be able to login.

    8.1. Registered users shall require an Email and Password consistent with register information to login.

9. Registered Users shall be able to make posts.

    9.1. Posts shall require a name, photo, price, description, pickup locations, contact information, and category.

10.Registered Users shall be able to contact sellers about posts.

16. Registered users shall be able to choose their pickup location.

Admin:

18. Admin shall be able to accept selling requests.

19.Admin shall be able to take down selling posts.

20. Admin shall be able to delete accounts.

- Say what is unique in your product (if anything)
- URL to your product accessible to instructors, on deployment server

***** The above list of final functions is used to <u>grade your final project</u> and will be checked on your final delivery for functionality and correct operation. Failure to deliver complete list of these committed functions will result in reduced grade for SE Product rubric. ******

2) Usability test plan

- **Test objectives:** To test the usability of the “post” function of our website. This is to discover the actual effectiveness and efficiency of our website. We can then take this data to further our UCD approach and make our web application and user friendly and usable as possible.

- **Test background and setup:**

  For ease of the user, the user needs little prior knowledge of web applications other than the requisite knowledge of how to access and navigate the Internet using URLs. The intended users of the post function are registered SF State Students, who would like to sell an item or service on our site to other SF State Students via their smartphone or computer connected to the internet. The starting point will be the homepage of the website, to see if a user is properly able to locate the post creation feature in the navigation bar/header quickly as it is displayed on every page on the site. The test starting URL is: *sfsuquad.com* nonetheless, users will have to navigate through the site to the /register, and /create_post pages on the website to complete the task.

- **Usability Task description**:
  - Go to the website.
  - Create an account.
  - Create a post that includes a description, select a category, and upload a photo.
- We would measure effectiveness by the number of respondents who were able to complete the task. Also, we would measure it by the average score of users giving “creating a post was easy” a 4 (Agree) or higher on the Lickert scale.
- We would measure efficiency by looking at our Google Analytics and analyzing the number of clicks it takes a user to complete this task. It takes only 5 clicks to complete this task. Under 10 clicks and we would feel that the site is efficient.

- **Lickert subjective test:**
  - Creating a post was easy.
  - The website was simple to navigate.
  - I had a positive experience with the website.

3) QA test plan

- ***Test objectives: - what is being tested***

  Posting functionality

- ***HW and SW setup (including URL):***

Server Host: Google Compute Engine 1vCPU 1.7 GB RAM
Operating System: Ubuntu 18.04 LTS
Database: MySQL Community Server 8.0.2
Web Server: Apache HTTP Server 2.4.41
Server-Side Language: Python
Additional Technologies:
Web Framework: Flask, Bootstrap v4.3.1
IDE: PyCharm
Web Analytics: Google Analytics
URL: sfsuquad.com

- ***Feature to be tested***

  Ability to post by a user of SFSUQuad

- ***QA Test plan:***

Test #	Test Title	Description	Input	Expected Output	Actual Result
1	Search	On Google Chrome v78. Go to the top of the screen, type in the search bar “couch”.	couch	8 Results with “couch”	PASS
2	Post item that is for sale	Using Mozilla Firefox v68. Press the “Sell” button, then fill out the form (select category, price, and description)	Posting of a couch for sale	Couch posting on the website in under 24h/pending admin approval	PASS
3	Sign Up	On Google Chrome v78. Press “Sign up” button, fill out the form, submit	Sign-up request to the server	Credentials saved to database and valid on login	PASS

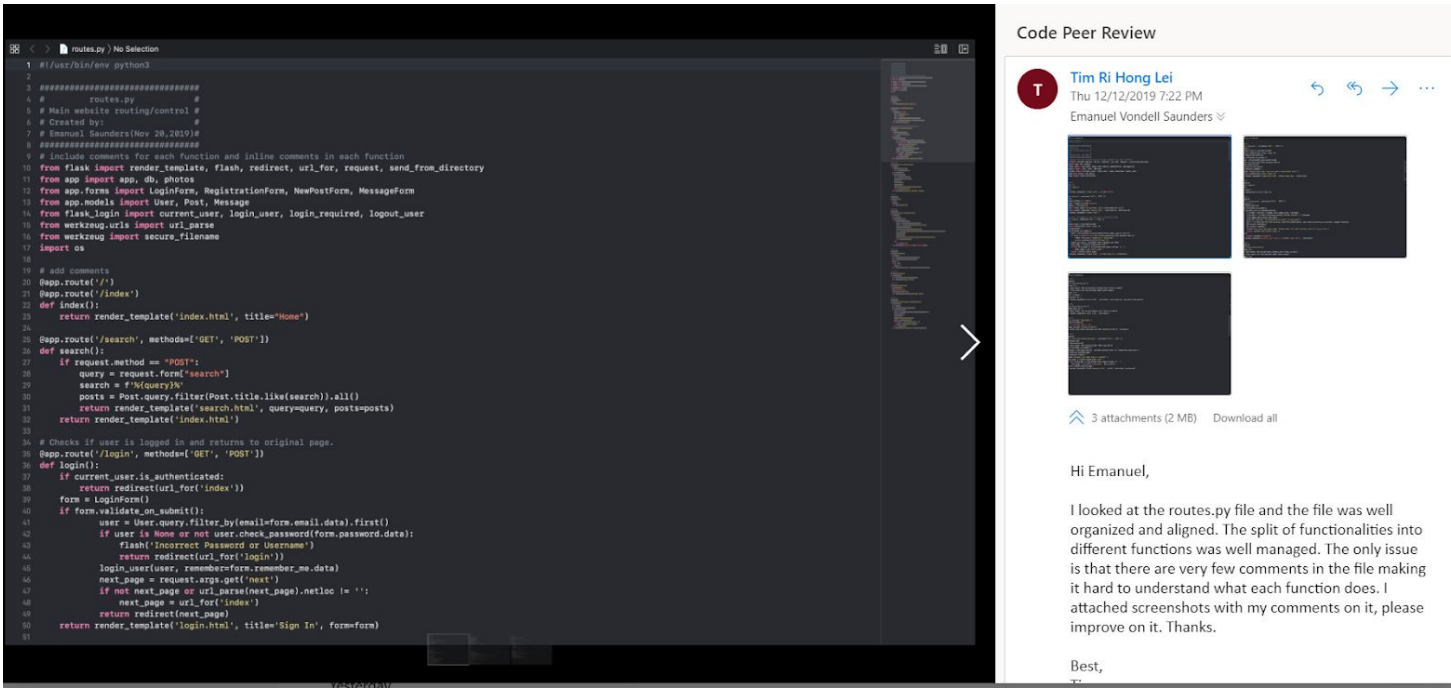
4) Code Review:

a) By this time you shall have chosen a coding style. In the report say what coding style you chose.

Google Coding Style

b) Choose the code (substantial portion of it) related to the feature you used for QA and usability test. You need to submit an example of the code under review (or part of it – 2 pages or so MAX) for this function to be peer reviewed, and document this as follows:

Peer Code Review (Done by Tim Lei on Routes.py by Emanuel Saunders):



5) Self-check on best practices for security – ½ page

1. List major assets you are protecting
   - **1.1. User Information**
     - **1.1.1. Passwords**
     - **1.1.2. Email**
     - **1.1.3. User Images**
   - 1.2. Database
2. Say how you are protecting each asset (1-2 lines of text per each)
   - **2.1. Email and Passwords - Hashing (Werkzeug and MD5)**

2.2. Images - Hash filenames and change file permission on the server to restrict global access/modification.

3. Confirm that you encrypt PW in the DB
   3.1. Yes
4. Confirm Input data validation (list what is being validated and what code you used) – we request you validate search bar input for up to 40 alphanumeric characters;
   4.1. Yes

6) Self-check: Adherence to original Non-functional specs – performed by team leads

1. Application shall be developed, tested and deployed using tools and servers approved by Class CTO and as agreed in M0 (some may be provided in the class, some may be chosen by the student team but all tools and servers have to be approved by class CTO). - **DONE**
2. Application shall be optimized for standard desktop/laptop browsers e.g. must render correctly on the two latest versions of two major browsers - **DONE**
3. Selected application functions must render well on mobile devices - **DONE**
4. Data shall be stored in the team's chosen database technology on the team's deployment server. - **DONE**
5. No more than 50 concurrent users shall be accessing the application at any time - **ON TRACK**
6. Privacy of users shall be protected and all privacy policies will be appropriately communicated to the users. - **ON TRACK**
7. The language used shall be English. - **DONE**
8. Application shall be very easy to use and intuitive. - **DONE**
9. Google analytics shall be added - **ON TRACK**
10. No email clients shall be allowed - **DONE**
11. Pay functionality, if any (e.g. paying for goods and services) shall not be implemented nor simulated in UI. - **DONE**
12. Site security: basic best practices shall be applied (as covered in the class) - **DONE**
13. Modern SE processes and practices shall be used as specified in the class, including collaborative and continuous SW development - **ON TRACK**
14. The website shall prominently display the following exact text on all pages *"SFSU Software Engineering Project CSC 648-848, Fall 2019. For Demonstration Only"* at the top of the WWW page. (Important so as to not confuse this with a real application). - **DONE**