SW Engineering CSC648/848 Summer 2019
Park Parole
Team 07

Erick Velez - evelez1@mail.sfsu.edu
Vincent Santos
Jimmy Kwan
Jackie Shan
Hector Aguirre
Kevin Paredes

Milestone 4
August 6, 2019

Date	Version	Comments
8/6/19	1.0	Initial Draft

1.) Product Summary

Ever watch a movie were actors are strolling down on a beautiful local park where all the flowers are perfectly bloomed. A trail of fallen leaves on a trail to make it more mesmerizing.to the point that you make a trip to that park and there is trash all over the place. Construction next door where you can not even hear the birds sing? And you wish you had known this before traveling all the way to the park? Fear no more! With our website anyone can inform you about any park in the area you wish! One click and BOOM! You find your park and it will tell you any current issues going on and you are also free to post and warn anyone! Completely free ! no registration necessary! Who doesn't love free stuff?

“Park Parole”

Things that were promised:

- User login
- User signup
- Users able to search all issues
- Search issues specific to category
- Search issues according to description
- Able to see issues related to Park on a visual map
- Search for issues specific to Park

What is unique in your product

- Our product is visually appealing
- Easy to maneuver

URL to your product accessible to instructors, on deployment server

At the moment: http://34.68.209.201:3000

2.) Usability test plan

The function that we would test is the search usability.

Test objectives:

Our test objective is to test our search function. We want to make sure that users are able to use the drop down menu then click in search for any type of issues. Whether clients are either trying to post any park issue or search for other environmental issues within the park.

Test background and setup

System setup: Setup is on public-facing server with tested latest changes.

Starting point: Starting point will be the application's homepage.

Intended users: Our intended users should be anyone that has access to our website. So we could get some feedback and criticism as well so everyone has the availability to post any issues.

URL: http://34.68.209.201:3000/

Usability Task description:

Effectiveness

- Search results appear in a clean manner
- Search is easy for user to understand
- Search categories are relevant to user interest

Efficiency

- Search results come up quickly
- Search page seems familiar to user (easy to identify results)
- Search is always easily accessible to user

	Strongly	**Agree**	**neutral**	**disagree**	**Strongly**

	agree				**disagree**
1. Search brings relevant issues					
2. Categories are relevant to interest					
3.					
4. Overall, the site is not complex and easy to follow					

3.) QA test plan

- Test objectives: Search is being tasted
- HW and SW setup(including URL):
- Feature to be tested: Search
- QA Test plan: 3 test cases and results of testing them on your system: appr. 1 page. Use tabular formats as in the class, see below

Test #	Test Title	Test Description	Test Input	Expected Correct Output	Test Results (Chrome)	Test Results (Firefox)
1	Blank Query	Test search functionality with a blank, empty query	Nothing	All issues currently in table	PASS	PASS

2	General Query	A query containing text but no chosen category to narrow results	Garbage Fire Golden Gate Park	Only issues containing words within the query in the description	FAIL	FAIL
3	Specified Query	A query containing text and a selected category	Golden Gate Park (‘Fire’ category selected)	Only issues within the ‘Fire’ category that contain the words from the query within the description	FAIL	FAIL

4.) Code Review:

The coding style chosen will be based on Google’s JavaScript style guide.

Here are the main points:

- Camel-case variable, function, and file names (numOfCounters) with descriptive names.
    - They should always begin with a lowercase character, no numbers
- Semicolons must end all statements and function declarations, for sake of clarity
- Indentations must be consistent relative to the height of the code block

All other points should adhere to the style guide, but these points take precedence.

code review - Hector Aguirre

Pibe Aguirre <pibeaguirre9@gmail.com>
Tue 8/6/2019 10:46 PM
To: Erick Velez <evelez1@mail.sfsu.edu>

```
const
db = require('../models/database.js');

// REVIEW: have a header that is very descriptive

module.exports = {

issues:
function (req,
res,
next) {

//

//REVIEW this is the users search


var
searchTerm = req.query.search;

//user's selected category

var
category = req.query.category;




//Review looks clean but I feel we should separate to have the code easy to see

let
query = 'SELECT * FROM Issue';

if (searchTerm !=
null &&
category != null) {

query =
`SELECT * FROM Issue WHERE Category = '` +
category + `' AND Location LIKE '%` +
searchTerm +
`%'`;
```

5.) **Self-check on best practices for security**

- We are encrypting passwords in the backend upon user submission of registration
- We are validating fields for login, registration, posting and search.
    - Login, registration, and posting fields are required to be filled before being submitted.
    - The search field will be checked for up to 50 alphanumeric characters.

6.) Self-check: Adherence to original Non-functional specs

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). **ON TRACK**
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 **ON TRACK**
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. **DONE**
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 **DONE**
10. No e-mail clients shall be allowed **DONE**
11. Pay functionality, if any (e.g. paying for goods and services) shall not be implemented nor simulated. **N/A**
12. Site security: basic best practices shall be applied (as covered in the class) **ON TRACK**
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, Summer 2019. For Demonstration Only” at the top of the WWW page. (Important so as to not confuse this with a real application). **DONE**