

SW Engineering CSC648/848 Summer 2019

Name of the application: The Better City

Team: 04

Maithri Chullikana House: Team Leader

Email: mchullikanahouse@mail.sfsu.edu

Soham Harshadbhai Prajapati: Front-End Leader

Patreck Benjamin Raymore: Back-End Leader

Bahar Moattar: GitHub Master, Document Editor

Ziping Huang: Back-End Developer

Aashutosh Bajgain: Back-End Developer

Fengze You: Front-End Developer

Zeyuan Cai: Front-End Developer

Milestone: 1

Date: 07/01/2019

Revision: Second Draft (1.1)

## Personas

Emily: College student and a member **in** **of** an environmental club

### About Emily:

- Emily is 20 years old college student and has a part time job.
- She would volunteer for environmental related organizations.
- She likes to go to national parks and hike.
- She likes taking pictures of the **environmental places** (nature? Maybe a better word) and post them on the internet.
- **She is environmentally conscious and passionate about the environment (too many “environment”**
- She would report environmental issues if she had seen any.
- Knowledgeable with technology and the internet.
- Impatient and hates slow loading websites.

### Goals and Scenario:

- Emily went hiking in a national park and suddenly saw the roads were not safe to travel which needed to be fixed.
- **She wants to find a way to report this to the specialist and get this problem fixed and make this place safer for people to hike.**
- **She thinks maybe she can take a picture of this issue and finds a way on the internet to report it to the workers and specialists in this area.**

- She wants to make a report by taking a picture and pinpointing the location of this issue so that the workers can fix the roads to make it safer for people to hike. (too straight to the point maybe?)

David: Environmental worker for a big company (Maybe a name for an imaginary company?)  
the government

About David:

- David is 51 Years old married and father of 3 children.
- Works for the government (companies instead of government?) involved with monitoring and fixing environmental issues.
- Very busy with his job. (he loves his job)
- He browse the internet and wants to find reports where people have reported these issues. (He browses the internet and looking for what people have been reported about these issues.)
- He would organizes a team of workers and try to fix the issues in the location based on the reports.
- Has proficient skills with technology and the internet.
- Not too so patient with web applications since there is so much reports for him to handle.

Goals and Scenario:

- David is currently at work and have to monitor environmental related issues that are reported by people.

- He has to prioritize which environmental issues **he** has to **be fixed** first and set the status **to let people know to show that** they are currently working on it.
- He would then organize a team of workers to fix these **environmental** (too many “environmental” again, we already know these issues that we are talking about are. “environmental” issues) issues.

## Main Use Cases:

Emily wants to go **hiking today to a hike**. She checks **about for** the nearest park in our website for any serious environmental issues. **She** goes to the park **and** she sees that **the** trash is overflowing. She goes to our **website and** searches the park by **the** name, and then **she will try to post the issue by posting a photo from the situation and add a small description. tries to post the issue.** Emily chooses the appropriate function from our website and writes about the issue. She is prompted to sign up for **our this website** before she can submit her write up.

David logs in to our website to monitor **the** environmental issues. He sees **the Emily's report issue reported by Emily** and assigns it to his team of workers **to get it fixed. David's team fix the issue** **After the issue has been fixed by David's team, he and David** changes the status of **issue the report** to “resolved”. **in our site.**

## Users:

- Emily goes to the park near her school. She sees that **the** trash is overflowing, **and She** goes to our **website**, searches the park by name and then reports this issue.
- **While reporting the issue**, she can also write some descriptions **about it the issue**. Also, she **can** wants to take some pictures of **the** overflown trash and **she can** uploads **them in there. some of those pictures as well**. When she is finished **with the report, and tried to submit it by clicking on “report/add post”, it will prompt her to sign in or register on the website. writing and uploading the picture, she clicked the report/add post button which will prompt her to sign in or register.**
- Emily wants to be a part of this organization. So, she **goes went on to** our website and **she** has registered herself **signed up** as a User **of this website. After registering, Now she is will be** able to login by her credentials.
- Once she logged in, she will be able to see **her** all **the** reported issues **and** **. Also, she can** check the status of **them** all reported issues.

## Administrator/Staff:

- David is a **environment issue** manager/worker **in an environmental company**. He wants to check for any pending or new reported issues. He goes to the website **and login as a system administrator with his credentials.** and he has administrator credentials of this **website. So, he will login as a system administrator.**
- Now, once David logged in **as an administrator**, he will be able to see all **the** reported issues. He can also filter **these reports reported issues** based on their status. **E.g,for**

example, currently he wants to check only on newly reported issues. So, he filters the list by the “Open” status. status as “Open”.

- Once he sees any an open issue, he will be able to assign it these open issues to some staff members which who are suitable to solve them. these issues.
- When the problem is resolved by the team members, he will change the status and mark it as “Solved”.
- If David notices that some issues are solved already by some staff members, then he will be able to mark that issue as “Solved”.

### Main Data Items and Entities:

- Users
  - Name
  - email
  - Password
  - Id
  - Photo
  - Issues

The user’s entity will be used to store the information for all individuals who login into the site and wish to post an Issue. Upon registration each user must input a Name, valid email, and password. However, the user is not required to upload a photograph. The issues data item is used to store all the past issues that the user had reported.

- Admins (i.e. City Manager/worker)
  - Name
  - Email
  - city
  - Password
  - Id
  - Photo
  - Issues

The Admin entity is used to store all users of the application who have the role of Admin. The admin has a similar registration process as a regular user. However, after registering they must be approved by another admin. Although the issues data field is named similar to those of regular users, it serves a different function. This data item is used to store the ID of all the issues that the admin is tracking. And it’s only these issues that the admin can change the status of.

- Issues
  - Photo(s)
  - Location
  - Status (e.g. open/in progress/solved)
  - Description

- Reporter (i.e. user name)
- Timestamp

The Issues entity will be used to store each issue reported by a user. Each issue must have at least one photo but can also have multiple (i.e. up to five). The location for each issue can be inputted manually or gathered via geo-location. The status of the issue can be toggled between three states: open, In-progress, and solved. However, the state of the status can only alter by a user with the role “admin”. Furthermore, each issue must have a description in order to be submitted. The time stamp will be used to keep track of how long each issue has been open.

### Initial list of functional requirements:

1. Users shall be able to report environmental issues. Only the registered users can report the issues. Guest Users will be asked to register before submitting the report issue.
2. Users shall be able to edit the reported issues.
3. Users shall be able to post comments on the reports issues. Only the registered users can comment on the reports issues. Guest users will be asked to register before posting a comment.
4. Users shall be able to see the reported Issues.
5. Users shall be able to sign up and log in to the website. Registered users shall use the registered email address and password to log in to the website. Guest user shall use a First name, Last name, email address and a password to sign up to the website.
6. Users shall be able to search reported issues. Guest users shall sign up for searching the reports. reported issues.
7. Users shall be able to browse through the issues.
8. Users shall be able to filter issues by categories, ratings, and status. Guest users shall sign up for filtering the issues.
9. Users shall be able to rate issues based on their priority. Only registered users shall rate the reports issues. Guest users shall be asked to sign up for rating the issue.
10. Users shall be able to geotag.
11. Users shall be able to choose the park or environment they want to make the report for.
12. Users shall be able to see the ratings of a report. an issue
13. User shall be able to see the current status of any reported the issue
14. User shall be able to upload a photo to their report to show more details picture related to the issue
15. User shall be able to upload a video related to their reports.
16. Admin shall be able to publish a report the issue.
17. Admin shall be able to change the status of a report. an issue.
18. Admin shall be able to log in, by . Admin shall using the organization email and password to login to the website.
19. Admin shall be able to assign tasks to their staff to get the issues resolved. the issues to staff.
20. Admin shall be able to manage the user database. e.g Admin shall be able to remove duplicate or suspicious accounts.
21. Admin shall be able to edit/delete reported issues. e.g Admin shall be able to delete duplicate or irrelevant issues reports.

### List of non-functional requirements:

1. The 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).
2. The application shall be optimized for standard desktop/laptop browsers e.g. must render correctly on the two latest versions of two major browsers
3. Selected application functions must render well on mobile devices
4. Data shall be stored in the team's chosen database technology on the team's deployment server.
- 5.No more than 50 concurrent users shall be accessing the application at any time
6. Privacy of users shall be protected, and all privacy policies will be appropriately communicated to the users.
7. The language used shall be English.
8. The application shall be very easy to use and intuitive.
9. Google analytics shall be added
- 10.No e-mail clients shall be allowed
11. Pay functionality, if any (e.g. paying for goods and services) shall not be implemented nor simulated.
- 12.Site security: basic best practices shall be applied (as covered in the class)
13. Modern SE processes and practices shall be used as specified in the class, including collaborative and continuous SW development
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).

### Competitive analysis:

Feature:	Competitor A (The Wilderness Society):	Competitor B(IVAN Online):	Competitor C(EPA US Environmental Protection Agency):	Our Product:
----------	--	----------------------------------	---	-----------------

User Report Issue(text, pictures, video)	+	+	++	++
User Filter Issue	-	-	-	+
Rate Issue based on Priority	-	-	-	+
Geotag where the issue happened	+	-	-	++
<ul style="list-style-type: none"> <li>+ feature exists , ++ superior , - did not exist</li> </ul>				

There are 3 main competitors on the market for our product (A: The Wilderness Society, B: IVAN Online and C: EPA US Environmental Protection Agency). Based on the key features that we have listed on the table, none of them are better compared to our product. For this product , the most important feature would be the Issue report function. A, B competitors do have the feature, but they cannot upload any images or video. C competitors have the feature to upload image and video, but before they do so, there will be a long login, verification process. Our product on the other hand give users easy access to report an issue with text, image and video. The next features which are filter issue and rate issue based on priority, for these two features, none of these competitors have these features. Our product does take these two features serious, our team felt being able to rate issue based on priority is very important feature and it allows the admin or user to know the priority of each issue and be able to take action as soon as possible. Geotag was any key feature for our product, Competitor B and C did not have that feature. Compare the geotag feature between A and our product, the advantage that we have would be the convenience of putting Geotag. Competitor A allows users to insert the location by adding text. Our product would allow user to get their reported issue location by using their GPS in their phone. Each competitor has their own specialty but based on the features that each of them get, our team believes that our product has advantages on the market.

Team:

Aashutosh Bajgain: Back-end developer team member back end

Bahar Moattar:GitHub master, document editor master

Fengze You: team member front end Front-end developer

Maithri Chullikana House: team leader

Patreck Benjamin Raymore: back-end leader

Soham Harshadbhai Prajapati: front-end leader

Ziping Huang: team member back end Back-end developer

Zeyuan Cai: team member front end Front-end developer

Checklist:



for each item below you must answer with only one of the following: DONE; or ON TRACK (meaning it will be done on time, and no issues perceived); or ISSUE (you have some problems, and then define what is the problem with 1-3 lines)

- Team found a time slot to meet outside of the class
- Github master chosen
- Team decided and agreed together on using the listed SW tools and deployment server
- Team ready and able to use the chosen back and front end frameworks and those who need to learn are working on learning and practicing
- Team lead ensured that all team members read the final M1 and agree/understand it before submission
- Github organized as discussed in class (e.g. master branch, development branch, folder for milestone documents, etc.)