

SW Engineering CSC 648/848
Section 01, Team 01
Spring 2018

R-Earth

Umang Mathur
(umathur@mail.sfsu.edu)

Ryan Liszewski

Rosalba Rodriguez

Chloe Zirbel

Lorenzo Moises

Taylor Marquez

Oleksandr Nibyt

Milestone 1

Document Version	Notes	Submission Date
Version 1.0	First Draft	02/28/2018
Version 2.0	Fixed issues with the summary, data definitions and use cases.	03/07/2018

1. Executive Summary

Globalization, industrialization, and technological evolution have brought environmental pollution to our front doors. Forests are cut down, toxic waste is spilled into rivers, reservoirs are used as dumpsters. Here at R-Earth we lamented the way in which humanity has harmed and degraded our most important possession: the natural world surrounding us. R-Earth aims to combat these effects at a local, community level by building a tool that makes addressing and resolving local environmental issues as easy and accessible as possible. Using R-Earth, citizens of San Francisco (SF) can pool information about their immediate surroundings that can in turn be used by local agencies to respond quickly and efficiently to environmental hazards.

R-Earth, a web application, bridges the gap between the people of SF and SF environmental Agencies by providing people a platform to post, and view community environmental hazards. Users can register through the application in order to report various types of environmental issues that might be taking place in their parks, and neighborhoods all over SF. These reports are made directly accessible to the local SF environmental agencies who will have designated admins can resolve the issues directly and update the posts with detailed information on the problem and its resolution. The application allows any user to search for environmental hazards that may be close to them or a selected location. Ultimately, the basic objective of this project is to give individual citizens the power to affect environmental change around them and spread awareness within the local community of San Francisco.

This project is brought to you by a team of Computer Science students from San Francisco State University. Umang is our Team Lead, Ryan is our Front End Lead, and Chloe is our Back End Lead. Lorenzo, Taylor, and Alex are back end developers, and Rosalba is a front end developer.

2. Use Cases

1. **Non-Registered User:** Calei is going on a camping trip at a national park and wants to check if there are any hazards within the park that she should avoid. Without the need to register, the website allows her to enter a location (name of park, street, city, zip code) and the website shall display a list of all environmental hazards within a given radius of the location she enters. The website will also display a map with each hazard with the entered location as the center. If she clicks on any of the hazards, information about the hazard (image, description, location, date of posting, etc.) will be displayed.
2. **Registered User:** Carmelo finds loose electrical wiring on a sidewalk in his neighborhood and wants to report it. The website allows him to submit a new hazard listing. If the user is unregistered at the time of submission, he will be prompted to register. The website will then display several different data fields to be filled out (description, location, etc.) He will also have the ability to submit a image of the hazard. Lastly, he will have all the functionalities of a non-registered user.
3. **Environmental Specialist:** Cairo, an environmental specialist, can review all hazard submissions. The website allows him to update the status of all the hazards (resolved, unresolved, in progress). The environmental specialist can also report if any of the hazards are illegitimate.
4. **Administrator:** John Cena, the administrator, oversees the website. If a hazard is reported and confirmed as illegitimate, he can delete the post from the website. If a registered user breaks the code of conduct, he can issue a warning to that user. If that user continues to break the code of conduct, he can ban the user. Lastly, he could answer any questions any user has about the website or hazards.

3. Data Definition

1. **Unregistered User:** They shall only browse the website without being able to post or edit the websites posts. They shall be able to also view already posted posts, details and updates to those posts.
2. **Registered User:** Users who have an account, and are logged into it, that are using the website. Registered Users shall be able to create new posts with a photos of an environmental problem, its description, and exact location with “google maps”.
3. **Administrator:** Users that have special privileges, and have the ability to remove posts from the site, update the status of post, issue warnings and bans from the site, and generally enforce the Code of Conduct for the site. Administrators also are responsible for helping users when needed.
4. **Environmental Specialist:** Users that have special privileges, and have the ability to update the status of post(1: process, 2: resolved, 3: not resolved) and also shall provide a description of solution.
5. **Environmental Listing:** Shall be created by registered users. Shall contain two parts, one for registered users and another for environmental specialists where the second shall be able to provide updates on a problem. Registered user shall be able to view the name, description, pictures and location of a problem. Also registered user shall be able to see status updates which shall be posted in a special field by the environmental specialists.
6. **Registration entry:** User is able to register on the website by providing user-name and password.

4. Initial List of Functional Requirements

Unregistered Users:

1. Unregistered users shall be able to browse listed environmental problems
2. Unregistered users shall be able to search listed environmental problems based on zip code for environmental issues
3. Unregistered users shall be able to register with the website using a username and captcha

Registered Users:

4. Registered users shall have all privileges of unregistered users
5. Registered users shall be able to create environmental problem listings using a designated form
6. Registered users shall be able to edit environmental problem listings if the status of the problem is changed
7. Registered users shall be able to edit their profile

Environmental Specialist:

8. Environmental Specialists shall have all privileges of registered users
9. Environmental Specialists shall be able to monitor environmental listings in order to assign workers to address the environmental problems
10. Environmental Specialists shall be able to update the environmental listings with tentative date of resolution or actual date of resolution
11. Environmental Specialists shall be able to update the environmental listings with any of the following: potential/actual cause of problem, plan of action, actual action taken to resolve the problem
12. Environmental Specialists shall be able to update the environmental listings to indicate they were unable to detect the problem

Administrator:

13. The administrator shall have all privileges of environmental specialists
14. The administrator shall be able to suspend a registered user if necessary
15. The administrator shall be able to remove an environmental listing if necessary
16. The administrator shall not be authorized to edit any environmental listing

5. List of Non-Functional Requirements

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).
2. Application shall be optimized for standard desktop/laptop browsers e.g. must render correctly on the two latest versions of all major browsers: Mozilla, Safari, Chrome.
3. Application shall have responsive UI code so it can be adequately rendered on mobile devices but no mobile native app is to be developed
4. Data shall be stored in the team's chosen database technology on the team's deployment server.
5. Application shall be media rich (at minimum contain images and maps)
6. No more than 50 concurrent users shall be accessing the application at any time
7. Privacy of users shall be protected and all privacy policies will be appropriately communicated to the users.
8. The language used shall be English.
9. Application shall be very easy to use and intuitive.
10. Google analytics shall be added
11. No e-mail clients shall be allowed
12. Pay functionality, if any (e.g. paying for goods and services) shall not be implemented nor simulated.
13. Site security: basic best practices shall be applied (as covered in the class)
14. Modern SE processes and practices shall be used as specified in the class, including collaborative and continuous SW development
15. The website shall prominently display the following exact text on all pages "SFSU Software Engineering Project, Spring 2018. For Demonstration Only" at the top of the WWW page. (Important so as to not confuse this with a real application).

6. Competitive Analysis

Competitors:

EPA: <https://www.epa.gov/enforcement/report-environmental-violations>

Broward: <http://www.broward.org/Environment/Resources/Pages/EnviroComplaint.aspx>

<u>Features</u>	EPA	Broward	R-Earth
Report an environmental complaint on the website via a form.	+	+	++
Report an environmental complaint via email.	+	+	-
Display Complaints to the User's.	-	-	+
Location of Complaint	+	+	++

+ Feature Exists

++ Superior Feature

- Feature does not exist

As compared to other similar product offerings, our product shall provide a better, intuitive interface for reporting an environmental issue. The interface strives to make reporting each issue a seamless task. One primary drawback in our competitors websites is the absence of a comprehensive list of all issues that have been crowdsourced till now. R-Earth shall make the data easily accessible and understandable for the common user. Our product shall allow users to quickly access posts with pictures and description of an environmental problem along with response to the problem provided by environmental specialists. As of now, one feature that is missing from the 'R-Earth' product proposal is the ability to report complaints via email. 'R-Earth' also allows people to see the exact location of the issue/hazard on an interactive map. This is a feature that shall make 'R-Earth' significantly superior to its competitors.

7. High Level System Architecture

1. Server Host, Instance size: Heroku, RAM: 512MB, CPU Share: 1x, Compute: 1x-4x
2. Operating System and Version Number: Heroku-16 (default stack) - Internally uses Ubuntu 16.04
3. Database and Version Number: PostgreSQL 10.1
4. Web Server and Version Number: Node.js 8.9.4
5. Server-Side language and Version Number: Javascript i.e. ECMAScript 2015 (ES6)
6. Other technologies/major libraries:
 - a. Express Web Application Framework , Version 4.15.5
 - b. Sequelize ORM, Version: 4.33.2 (Compatible with PostgreSQL database)
 - c. EJS view engine for client side templating, Version 2.5.7
 - d. Bootstrap Front End Framework, Version: 3.3.6(To build responsive UI)
7. Browsers supported: - Firefox 57, 58 - Chrome 63,64 - Safari 10,11

8. Team

<u>Team Member</u>	<u>Role</u>
Umang Mathur	Team Lead
Ryan Liszewski	Front End Lead
Rosalba Rodriguez	Front End
Chloe Zirbel	Back End Lead
Lorenzo Moises	Back End
Taylor Marquez	Back End
Oleksandr Nibyt	Back End

9. Checklist

1	Team found a time slot to meet outside of the class	Done
2	Github master chosen	Done
3	Team decided and agreed together on using the listed SW tools and deployment server	Done
4	Team ready and able to use the chosen back and front end frameworks and those who need to learn are working on it	Done
5	Team lead ensured that all team members read the final M1 and agree/understand it before submission	Done