SW Engineering CSC648/848 Section 01 Spring 2018

PARSE

Pollution and Accident Reporting for a Safer Environment

Milestone 5

May 21st, 2018

Team 04

Rodrigo Bell (rbell594@gmail.com)

Charlie Tuttle

Rohan Patel

Dion Matthew Lagos

Kamran Khadivi-Dimbali

Michael Schwiebs

Zhenru Huang

Table of Contents

Product Summary

PARSE is a website tool made by our team of 7 students at San Francisco State University. Our web-based service specializes in raising awareness about local environmental issues in San Francisco. A few of our functional specs:

- Ability to browse site as a guest user
- Guest users can register for an account using username and encrypted password
- Registered users can upload environmental issues for others to see
- All users can search for issues using zip code and category
- Admins are the only ones able to manage the database

Our website can be accessed at the following URL: https://csc648-team04.herokuapp.com/

SW Engineering CSC648/848 Section 01 Spring 2018

PARSE

Pollution and Accident Reporting for a Safer Environment

Milestone 1

February 28, 2018

Team 04

Rodrigo Bell (rbell594@gmail.com)

Charlie Tuttle

Rohan Patel

Dion Matthew Lagos

Kamran Khadivi-Dimbali

Michael Schwiebs

Zhenru Huang

Revisions

February 28, 2018	Initial Draft
March 5, 2018	Revision after instructor feedback, Frozen

Table of Contents

Executive Summary

Some of our most valued experiences take place in the natural landmarks our cities have to offer. San Francisco alone has over 220 parks that people have been enjoying for years. Maintaining these parks and our general environment has always been important step towards keeping our children and the local wildlife safe. Unfortunately, in this modern age, these natural environments can be tainted by garbage and hazardous materials.

PARSE started with the belief that technology should help clean our planet, not pollute it. With the number of environmental problems that are seen in many cities and parks today, it can be difficult for the average person to bring these issues to the attention of the public and proper authorities. The goal of PARSE is to close this communication gap by providing a web application with a quick and detailed reporting system.

PARSE is a web application that allows everyday people to report an environmental problem that they may have witnessed, such as misplaced garbage or hazardous spills. These reports can be detailed with a location, physical description, and any supplementary photos. All of this done in a quick and straightforward manner without any extensive setup on the user's end, and PARSE's makes it easy for the user to submit a report from their smartphone within minutes.

These reports can be easily viewed by anyone by simply visiting our website. The user can search for an environmental issue by location or keyword, and the results that follow will display any unresolved issues. The content itself can be managed by Content Moderators, who can use PARSE to open and close issues as they see fit.

The people at PARSE are proud to call Earth our home, and like many people, we care that our home is clean. We believe that PARSE can bring the people of any city or town together by providing them with an easy way to bring environmental issues to light, and together they can address them in a professional manner.

PARSE was started by a team of seven student developers, including five undergraduate students and two graduate students. The project's CEO is Dragutin Petkovic, Associate Chair and Professor of the Computer Science department at San Francisco State University, and the CTO is Anthony Souza, former graduate student and current lecturer at SFSU. This team has a wealth of experience in web development and software engineering. With all of the members living in the San Francisco Bay Area, we have an strong appreciation of the natural beauties this area has to offer. Our goal is to keep our oceans, parks, and cities clean of all hazardous waste and litter, and with PARSE, we believe we can make that a reality.

Use cases

1. Unregistered User
Rebecca is an activist who is passionate about caring for the environment. Looking for ways she can help around her local community in San Francisco, she begins searching online for issues currently occurring in the area. Upon examining the results returned by a search engine, she finds and visits ***CSC-648 Team 04's Website***. Here she can see all of the current listings of environmental problems that have been posted by the local residents of San Francisco who are registered on the site. From here she will be able to browse by a location, keyword, or category, and find any issues she may be interested in becoming involved with. After selecting an issue, she shall be able to view more details about it, including pictures, specific location, and any other information regarding the subject.

2. Registered User
Any registered user on ***CSC-648 Team 04's Website*** will be able to post an environmental issue than can be viewed by the public after approval. This allows residents to report issues in the area that normally would not have been brought to public attention otherwise. An example of includes John, a father of two who enjoys biking around the city with his children. One day as he and his kids are riding around town, they decide to take a lesser known bike trail that runs by a creek that leads into a water supply for local farmers to grow crops. While on the trail, John and his children notice that someone had dumped a large amount of motor oil into the creek, threatening the water supply connection further down the stream. John then decides to take action by taking a picture of the scene and logging onto his computer at home to report the issue. After using the UI tools to add details about the issue, he is then prompted to create an account in order to post it.

3. Environmental Service Company / Content Moderator
Any company or organization that works to combat environmental issues can use *CSC-648 Team 04's Website* as a tool to help further their work. PARSE is a local organization that aims to educate people on environmental hazards and ways they can decrease their amount of consumption and waste. Using the website as a tool, they can survey neighborhoods and regulate the issues, being able to confirm/approve content and close open issues. They can also use this website to assign a specific issue to a unit or team to work on fixing it. Furthermore, they will be able to evaluate priority of issues and assess which issues need to be addressed first.

Data Definitions

Unregistered user: Able to view environmental issue posts. No login required.

Registered User: Same abilities as unregistered user AND is able to post about environmental problems in a specific neighborhood. Account must be created with email, username and password. Must login to access personal posts, saved posts.

Environmental Service Company/ Content Moderator: Same ability as registered user AND can update status on a post. Also has access Priority Issue Tool. Account created with email, username, password, Service company verification.

Administrator: Same ability as registered user AND can manage user accounts and delete inappropriate content. Ensures the proper user and content policies are followed.

Environmental Issues Posts (Visible to all):

- Images of issue/exact location
- Section to add keywords for other users searching
- Maps/location/address
- Description of problem
- Search tool
- Option for user to save a post

Priority Issue Tool (Environmental Service Company use only):

- Analytics for service company to evaluate priority of issues

Bookmarks: Registered users may save viewed listings for later viewing. Registration will require email, username, and password.

High-Level Functional Requirements

Unregistered Users (guests):

1. Application shall display location of posted environmental issues on Google Maps
2. Application shall provide a search functionality filtering by city, state, or zip code
3. Application shall allow guests to register for a User or Service Company account.
4. Application shall display new and pending environmental incident listings.

Registered Users (in addition to unregistered user functions):

5. Users shall be required to register with full name, email, and password.
6. Users shall be able to provide a photo and a description of the environmental issue.
7. Users shall have permission to create postings of environmental issues to the database.
8. Users shall have a log of reported, pending, and resolved issues saved to their account.

Administrator:

9. Administrators shall have permission and power to remove postings
10. Administrators shall have the permission and power to remove any type of account or listing.

Service Company:

11. Service company accounts shall have access to a dashboard to choose which issues to resolve.
12. Listings can only be resolved by Service Company accounts or Administrators.

High-Level Non-Functional requirements

1) Application shall be developed and deployed using SW stack approved by class CTO Anthony Souza.
2) Application shall be hosted and deployed on Heroku as approved by Anthony.
3) Application shall be optimized for standard desktop/laptop browsers, i.e., must render correctly on the latest versions of all major browsers: Chrome, Safari, Mozilla
4) Application shall have responsive UI code so it can be adequately rendered on mobile devices but no mobile app is to be developed.
5) Data shall be stored in the MySQL database on the server in the team's Heroku account.
6) Maps showing location of the environmental issue shall be provided.
7) No more than 50 concurrent users shall be accessing the application at any time.
8) Privacy of users shall be protected and all privacy policies will be appropriately communicated to the users.
9) The language used shall be English.
10) Application shall be very easy to use and intuitive. No prior training shall be required to use the website.
11) Google analytics shall be added.
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) Paid site features shall not be implemented.
15) The website shall prominently display the following text on all pages “SFSU Software Engineering Project, Spring 2018. For Demonstration Only.”

Competitive Analysis

Feature	Epa	Lema	Broward	PARSE
Text Search	+	+	+	+
Search Filtering	++	-	-	++
Mobile End	-	+	++	++
User Login	-	-	-	+
Report Violation	++	-	++	++
Browse	+	+	+	++

+ feature exits
++ superior feature
- feature does not exist

As the table was showed above, our product provides a great user experience by comparing to other competitive products exit on the market right now. Our product has a better search design and user interface design which can provide an easier access for our user. And it may bring more potential users. And our users can report environment issues easily by using our product and protect our environment better.

System Architecture

Development Stack

- Server Host: Heroku 1x CPU 512 MB RAM
- Operating System: Ubuntu 16.04 Server (We're using Heroku which is a PaaS but it uses ubuntu internally)
- Database: MySQL 5.7
- Web Server: Node JS Server with Express
- Server-Side Language: Javascript

Frameworks & APIs

- Express - Minimal and flexible Node.js framework for serving web content
- Bootstrap - Front-end web development framework for responsive interfaces
- Google Analytics - Google API offering accurate website statistics

Version Control: Git (v 2.16.2)

Development Team

Rodrigo Bell, *Team Lead*
Charlie Tuttle, *Front-End Lead*
Rohan Patel, *Back-End Lead*
Michael Schwiebs, *Front-End Dev*
Kamran Khadivi-Dimbali, *Front-End Dev*
Dion Matthew Lagos, *Back-End Dev*
Zhenru Huang, *Back-End Dev*

Checklist

- Team decided on basic means of communications - DONE
- Team found a time slot to meet outside of class - DONE
- Front and back end team leads chosen - DONE
- Github master chosen - DONE
- Team ready and able to use the chosen back and front end frameworks - DONE
- Skills of each team member defined and known to all - DONE
- Team lead ensured that all team members read and understand the final M1 - DONE

SW Engineering CSC648/848 Section 01 Spring 2018

PARSE

Pollution and Accident Reporting for a Safer Environment

Milestone 2

March 18, 2018

Team 04

Rodrigo Bell (rbell594@gmail.com)

Charlie Tuttle

Rohan Patel

Dion Matthew Lagos

Kamran Khadivi-Dimbali

Michael Schwiebs

Zhenru Huang

Revisions

March 18, 2018	Initial Draft
March 26, 2018	Revised after instructor feedback - Frozen

Table of Contents

Data Definitions

Unregistered user: Able to view environmental issue posts. No login required.

Registered User: Same abilities as unregistered user AND is able to post about environmental problems in a specific neighborhood. User registration data requires email, username and password. Must login to access personal posts, saved posts.

Administrator: Same ability as registered user AND accept and delete environmental issues. Ensures the proper user and content policies are followed.

City Environmental Manager: Same ability as registered user AND can update status on a post. Also has access Priority Issue Tool. The development team will assign and manage site administrators.

Environmental Issues Posts (Visible to all):

- Images of issue
- Maps/location/address
- Description of problem
- Option for user to save a post
- Category of issue

Priority Issue Tool (City Env. Manager use only):

- Dashboard for service company to evaluate priority of issues

Functional Requirements

Priority 1 – MUST HAVE

Unregistered Users (guests):

5. Users shall display listings of posted environmental issues on Google Maps.
6. Users shall provide a search functionality filtering by keywords.
7. Users shall allow guests to register for a User account.
8. Users shall display new and pending environmental incident listings.

Registered Users (in addition to unregistered user functions):

5. Users shall be required to register with full name, email, and password.
6. Users shall be able to provide a photo and a description of the environmental issue.
7. Users shall have permission to create postings of environmental issues to the database.

Administrator:

9. Administrators shall have permission and power to remove postings.
10. Administrators shall have the permission and power to remove any type of account or listing.

City Environmental Manager:

11. Development team shall have access to a dashboard to list users, issues, and be able to set the issue status.
12. Listings can only be resolved by Service Company accounts or Administrators once the environmental issue is no longer a concern.

Priority 2 – MEDIUM IMPORTANCE

Unregistered Users (guests):

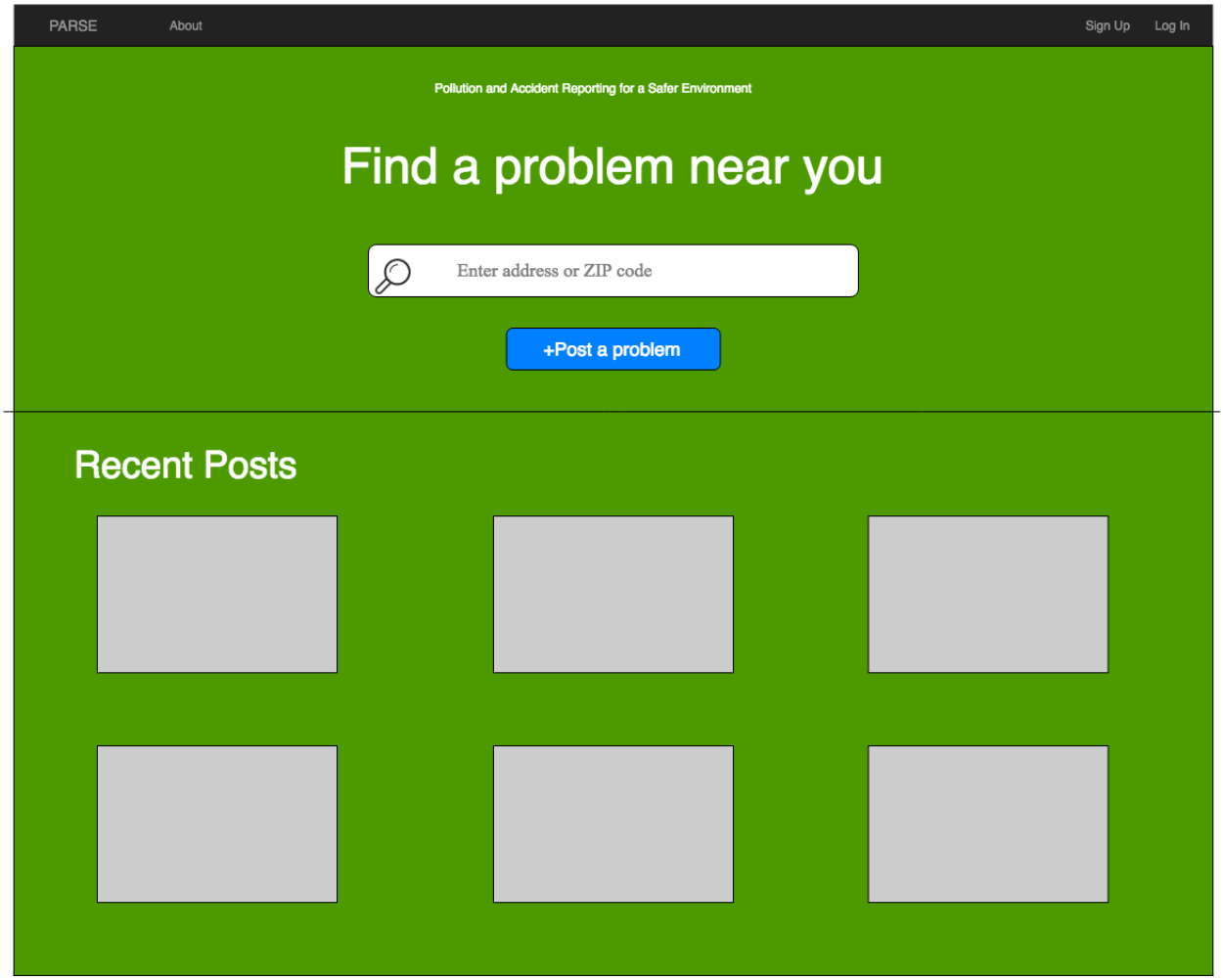
1. Application shall provide a search functionality filtering by address, zipcode, or category.

Registered Users (in addition to unregistered user functions):

2. Users shall have a log of reported, pending, and resolved issues saved to their account.
3. Users shall be able to make bookmarks of environmental issues.

UI Mockups and Storyboards

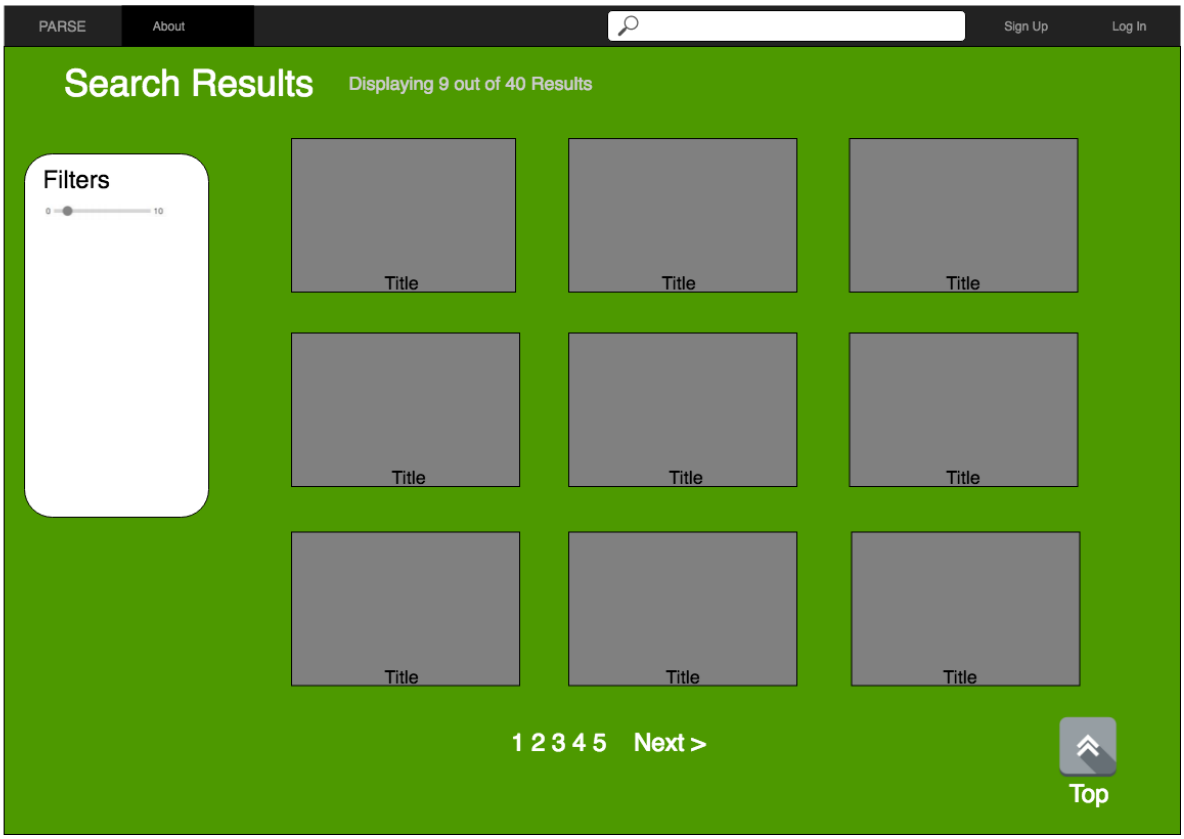
Landing Page



Use Cases

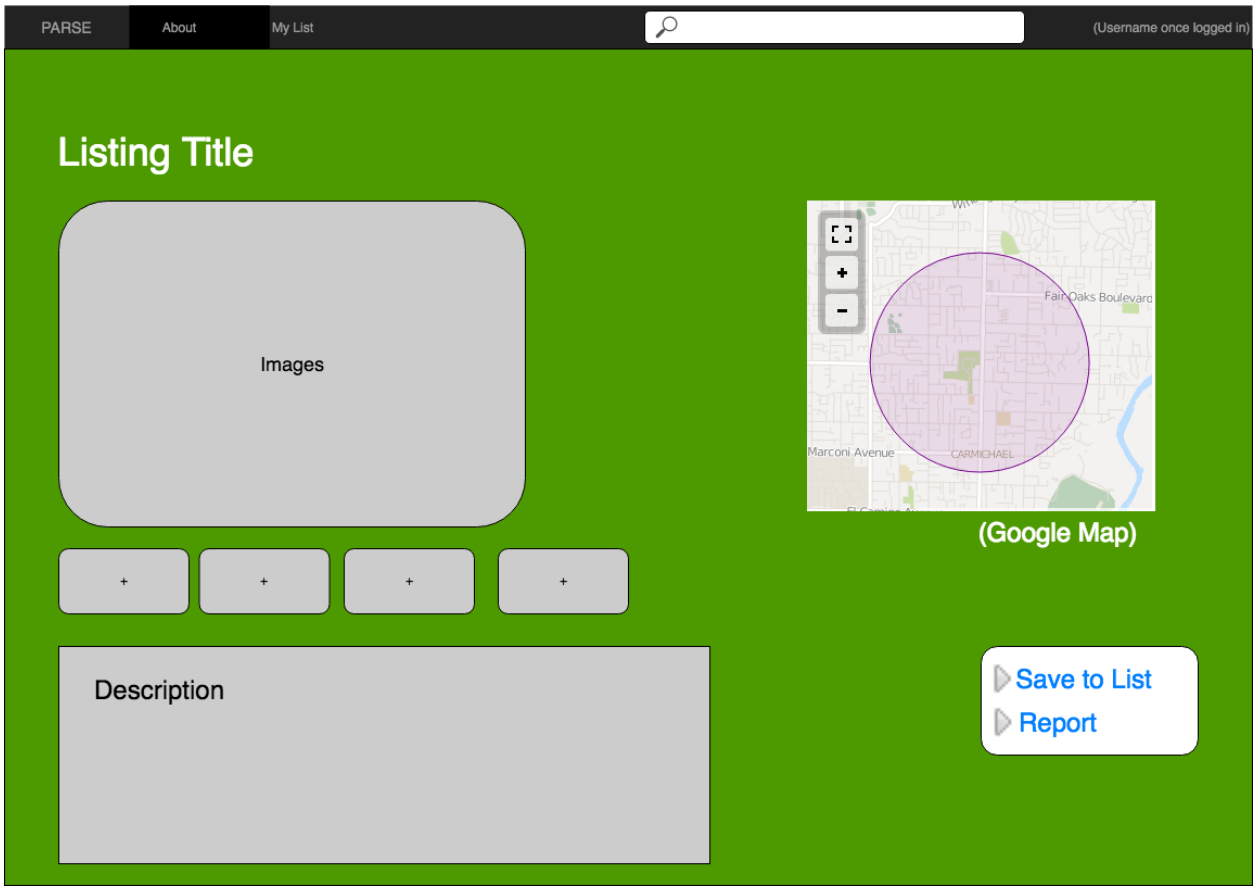
- **Unregistered User:** Able to search/view posts, sign up in the top right corner.
- **Registered User:** Able to sign in at top right corner, post/search/view a post.
- **Service Company:** Same abilities as registered user.

Search Results Page



All users: This page shows the listings according to the search. You can also refine the search by using the filter tool, with specifications coming in the future. The user can go to the next page by clicking the “Next>” button or clicking the page number. Search bar is included if the user wants to do a different search.

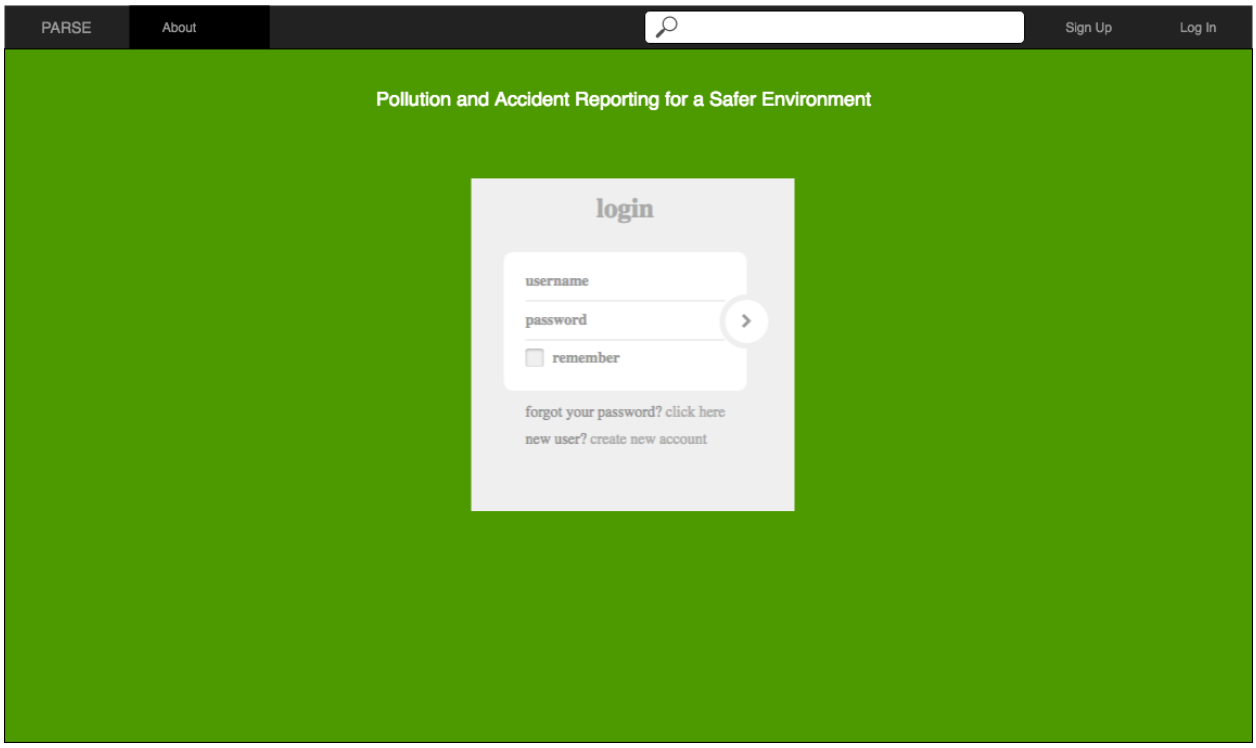
Individual Post Page



Unregistered User: Can only view the post.

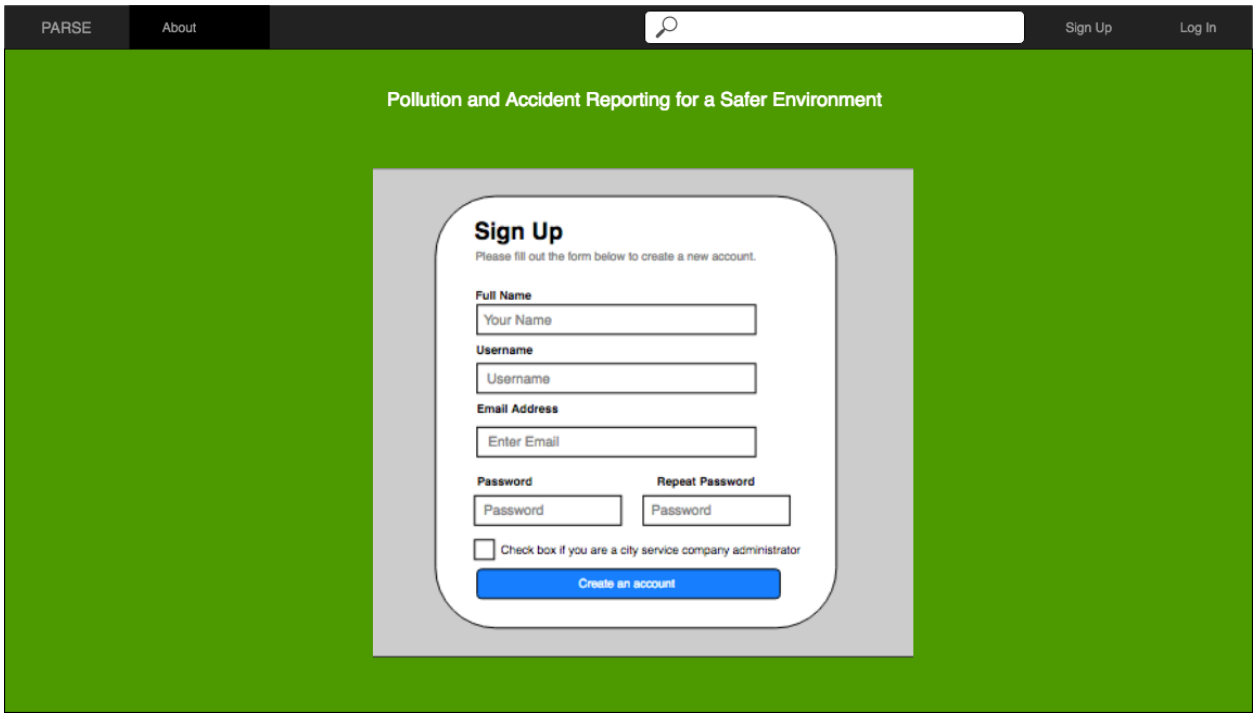
Registered User and Service Company: Able to save to their personal list and report the post if it does not abide to the terms of agreement.

Login Page



All Users: Registered users enter their username and password.

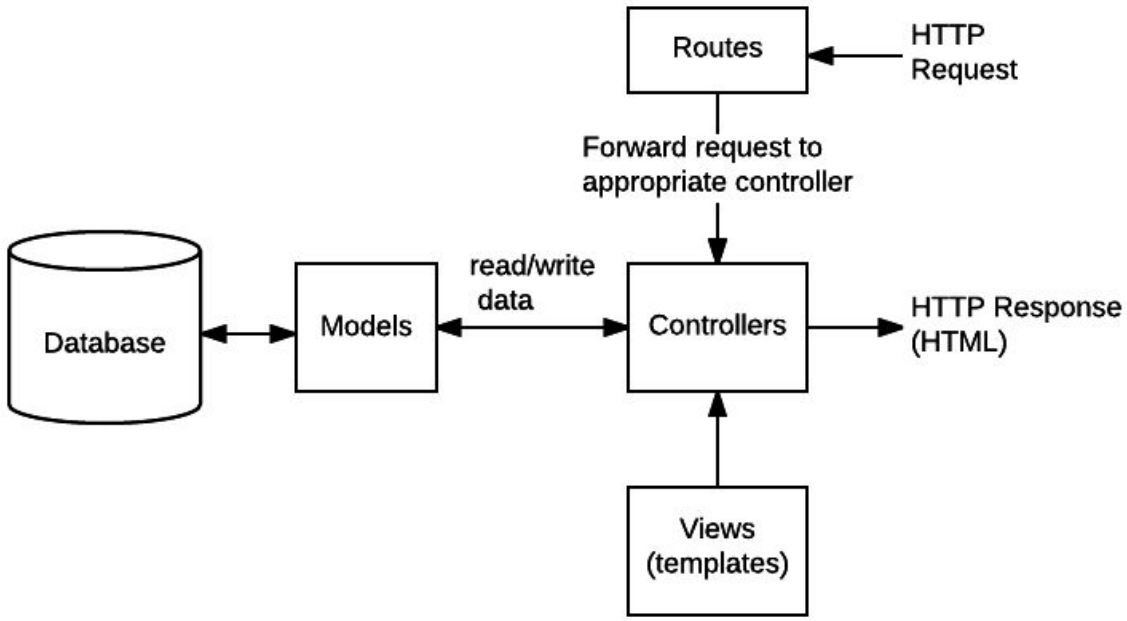
Sign Up Page



All users: Sign up by filling the following form. If you are from the service company, you check the box as you finish creating your account.

High-Level Architecture

For this project, we will use the general MVC pattern which is under Express framework in Node JS.



Source: https://developer.mozilla.org/en-US/docs/Learn/Server-side/Express_Nodejs/routes

The diagram shows the MVC pattern we are going to use for our project. First user will send a HTTP request to our Controller. And with this request, the Controller will request the Model to call the Database to get the data that the user request. And then the Model will send the data back to our Controller. Finally, our Controller will use these response data and render them into our View templates and send it back as HTTP or HTML response.

Database Organization

Schema

category	**issue**	**user**
-id(PK)	-id(PK)	-user_id(PK)
-name	-title	-name
	-category(FK)	-email
	-description	-phone
	-zipcode	-password
	-image	-isAdmin
	-isApproved	
	-isResolved	
	-user_id(FK)	
	-latitude	
	-longitude	

Image System

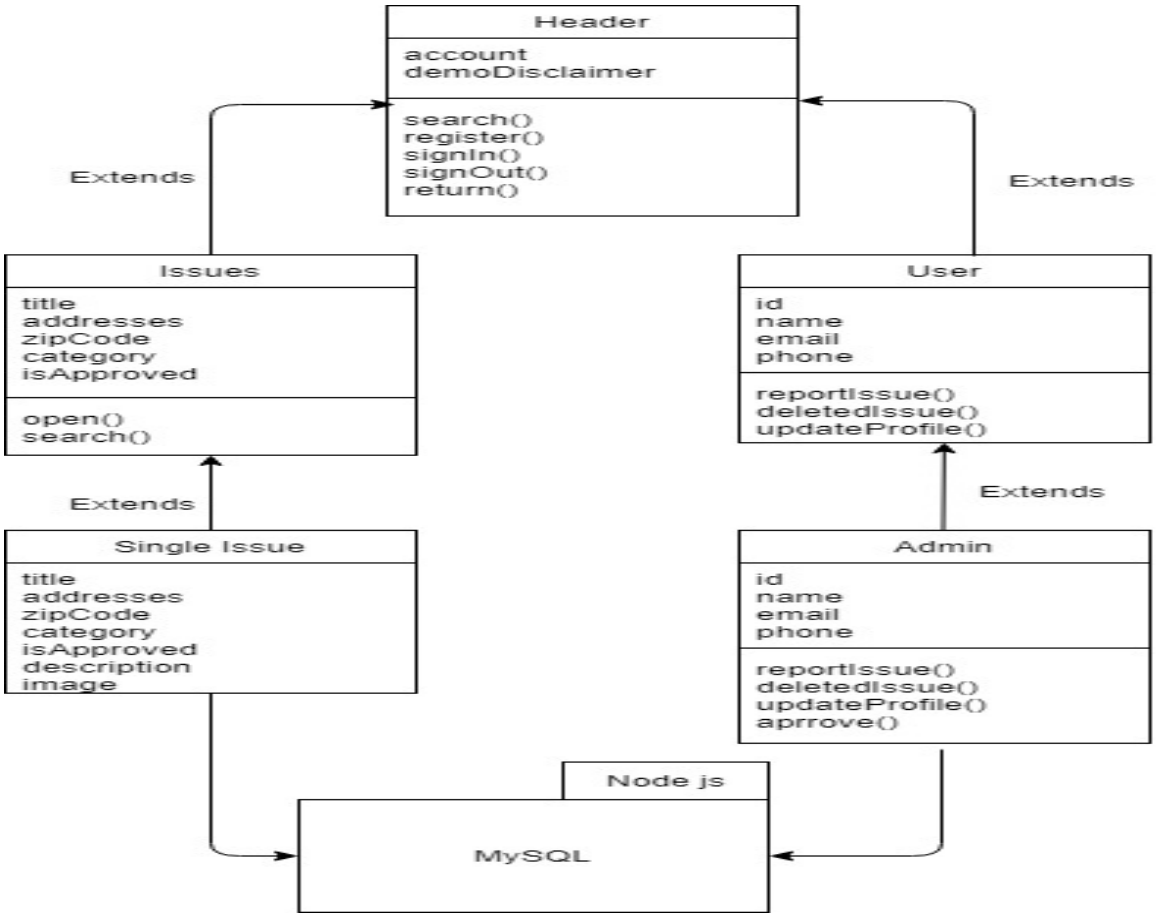
For images, we will use the file system to store it. We will have a path for images source. Once the users try to add images, the images will all be uploaded to our cloud under this path. And our database system will store the path and name of these image files. And once we try to use these images, we can easily pull up their path from out database and get it from our server.

Search Architecture

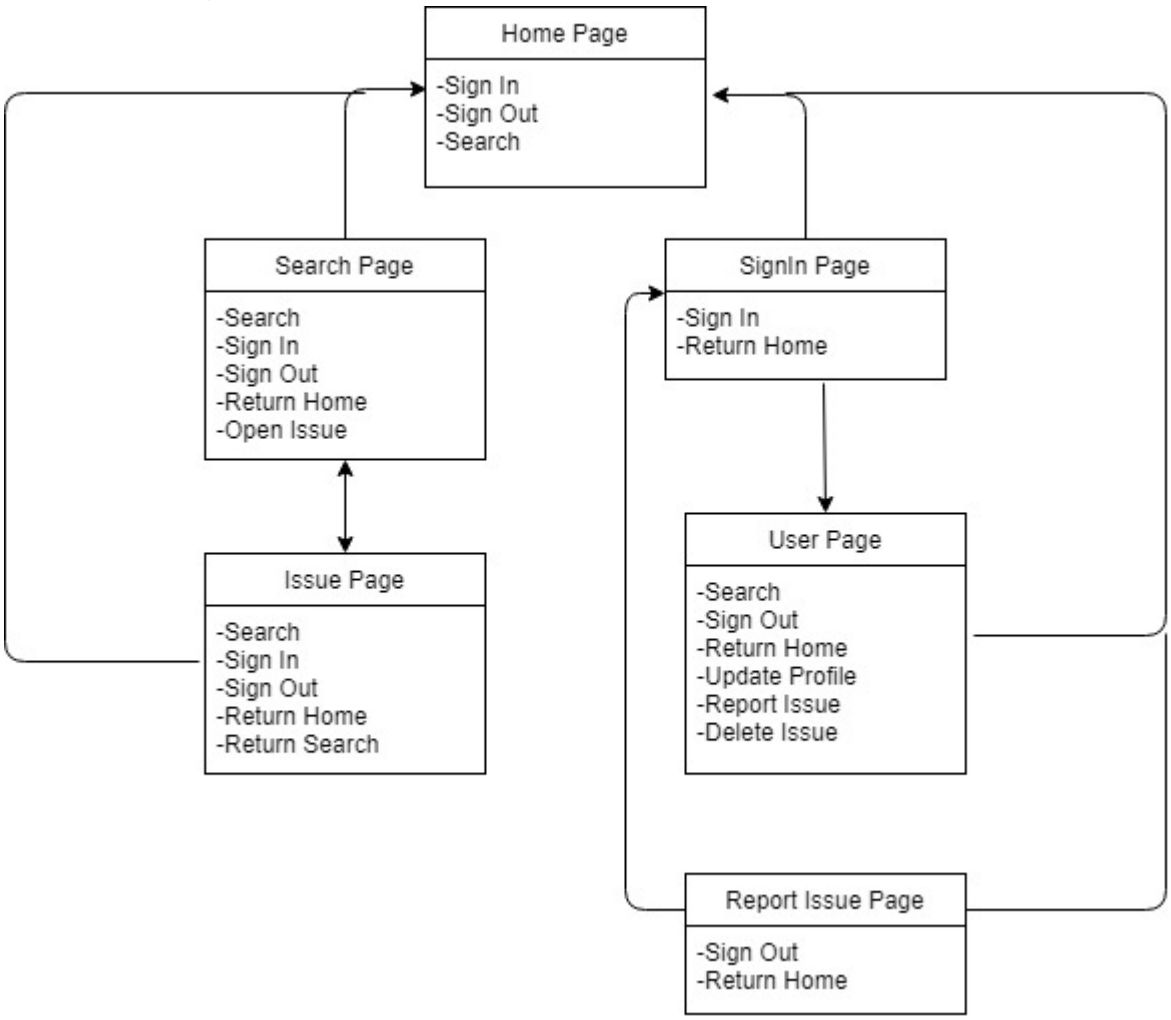
For search system, we will have a drop down menu for categories, so users can filter their results with different category. And also the users can type in the zip code and get the results of nearby.

UML Diagrams

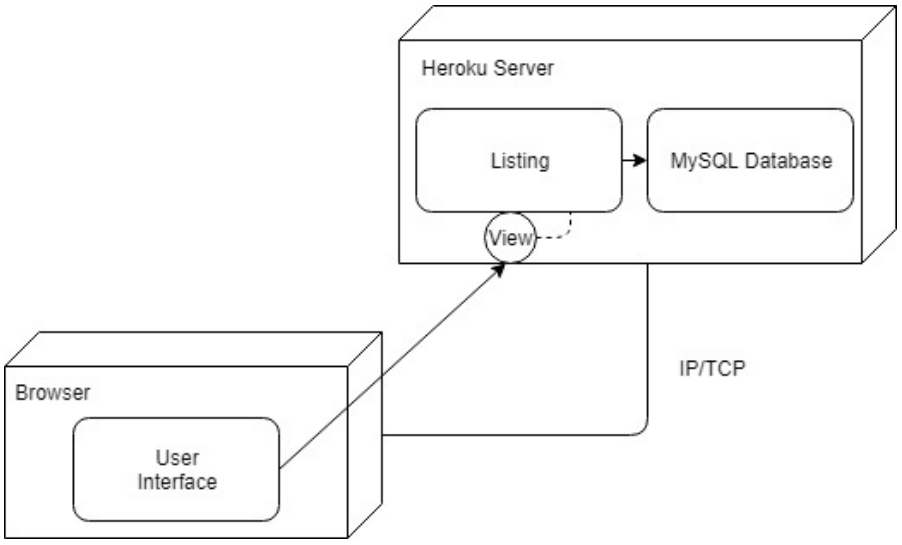
Class Diagram



View Flow Diagram



Component and Development Diagram



Project Risk Assessment

Upon review of the organization of the PARSE web application and its development team, the following risks have been identified:

- *Skills risks*: the majority of the team is unfamiliar with the given development stack (node.js/express) for the project, and may need accommodate time to gather and learn from various resources.
- *Schedule risks*: as the development team is comprised of senior level undergraduate students, scheduling conflicts may emerge due to individual projects or jobs.
- *Legal/content risks*: this project requires the use of media (uploaded images) which may potentially infringe upon the rights of its owners.

In order to resolve the identified risks, SCRUM-style meetings will be held between the team lead and the rest of the team weekly either online or in person to discuss any issues with the application development. At scheduled extended meetings, we will periodically review the coding procedures that need to be done, and research the methods of accomplishing them within the chosen framework and development stack. Code reviews will be conducted within the smaller teams to resolve any coding issues. The back-end team will work together to document a vertical prototype and educate the rest of the team.

During the weekly SCRUM-style meetings, we discuss availabilities for the week and plan for any scheduling conflicts that may arise with any of the members. To ensure deadlines are met, tasks are assigned with soft deadlines depending on the availabilities discussed. An extended weekly meeting is typically scheduled to discuss issues in depth at these meetings. We will keep the scope of the project to a minimal P1 set as development continues on.

For demo purposes, our project will use images of properties. In order to avoid any legal issues, all images must be reviewed by team leads before pushed to the live project site. The sources will be checked for the distributive rights of the images - only commercial-free (and no attribution required) images will be approved for demo use.

Milestone 3

Instructor to Check and comment below:

- Git/GitHub organization (e.g. organization of branches) GH organization file wise looks good. Branches are better used for features instead of per-member branches but this is OK.

- Git/GitHub usage: Comments on positing; Number of posting to GitHub; Apr. even distribution of submissions among team members (check GitHub post stats for all members) Commit messages look good, little to few duplicate messages. Distribution is OK some members have a low number of commits.

- Code documented (header, in code) with good coding style Little to no comments, no file headers.

- MVC/OO patterns followed up Followed.

- Frameworks (back end front end) deployed correctly Seems to be deployed correctly.

- Database organization (tables, naming...) Table names are OK, usually they are plural, user should be users, but this is a small issue.

- Blobs being used? If so, is it working? Image paths are used.

- Adherence to best practices of security (PW encrypted, search inputs verified etc.) Passwords seem to be encrypted now. Inputs looks not to be verified,

- Efficiency (proper use of image thumbnails, efficient search eytc.) Looks good.

- Other

SW Engineering CSC648/848 Section 01 Spring 2018

PARSE

Pollution and Accident Reporting for a Safer Environment

Milestone 4

May 17th, 2018

Team 04

Rodrigo Bell, Team Lead (rbell594@gmail.com)

Charlie Tuttle

Rohan Patel

Dion Matthew Lagos

Kamran Khadivi-Dimbali

Michael Schwiebs

Zhenru Huang

Table of Contents

Product Summary

PARSE is a web application that strives to be a simple to use listing platform for local environmental issues. With PARSE, people can easily check for environmental issue wherever they go in the city.
PARSE provides a simple and easy to use set of features:

- All guests can view environmental issues right from the PARSE home page
- Anyone can easily search for issues by typing in a zipcode or category in the search bar.
- Issues provide plentiful information and images with Google Maps integration.
- City managers have access to a simple to use dashboard making it simple to set the issue status.
- Explore the PARSE web application at: csc648-team04.herokuapp.com

Usability Test Plan

Test Objective

- The objective of this test is to gauge the usability of the site's main function: Search.
- Expose flaws with the Search function by asking our testers to perform the search task with no prior instructions. The testers will need to intuit how to use the Search function in order to search for the targeted queries.

Test Plan

System setup

- Tester will log into a computer with access to the internet and access to the following accepted internet browsers: Google chrome or Firefox.

Starting point

- Tester will begin at the site homepage wherein a search button will be present.

Task to be accomplished

- Tester will attempt a search for listings in a city of their choosing.
- Tester will attempt a search for listings with a zip code of their choosing.

Intended Users

- The targeted user is broad, as any person who is searching for a property is a potential customer.
- Users will be assumed to have no technical expertise and may be a complete novice to web browsing.

Completion criteria

- Tester is satisfied that they have accomplished the search query they attempted. This is evaluated from a post test questionnaire.

Usability Questionnaire

I found the Search function intuitive and easy to use (Check one)

__ Strongly Disagree __ Disagree __ Neither Agree or disagree __ Agree __ Strongly Agree

It was easy and intuitive for me to find the listings for the Categary I was searching for (Check one)

__ Strongly Disagree __ Disagree __ Neither Agree or disagree __ Agree __ Strongly Agree

It was easy and intuitive for me to find the listings for the zip code I was searching for (Check one)

__ Strongly Disagree __ Disagree __ Neither Agree or disagree __ Agree __ Strongly Agree

Comments:

QA Test Plan

Test Objective

1. The objective of this test is to validate the functioning of one of the primary features of the website, i.e. Search.
2. The tester shall be given specific instructions to follow and the results shall be recorded as a PASS/FAIL in each of the individual cases.

Hardware and Software setup

1. The tester should have access to a device that supports recent versions of two web browsers – Google Chrome and Mozilla Firefox.
2. The devices used should be connected to internet via Wi-Fi, Ethernet, Cellular, etc.
3. The internet connection speed should be adequate (at least 2.5MBps) for flawless browsing experience.
4. And finally, the user should open the following link on either of the prescribed browsers to begin the test.

Features to be tested

This test will solely focus on the Search feature of the website. The Search feature shall be operated by inputting a query in the search field on the very top of the homepage of the website. The permitted inputs are names of cities or the zip-codes.

Test #	Test title	Description	Test input	Expected output	Result (Chrome)	Result (Firefox)
1.	Zip-code search	Tester shall input the prescribed zip-code number into the search bar and the results shall be validated through comparison with the expected output.	94112	1 result with retrieved item having 94112 as zip code	PASS	PASS
2.	Category search	Tester shall input the prescribed name of city into the search bar and the results shall be validated through comparison with the expected output.	“Accident” category selected	3 results with retrieved item having “Accident” category selected	PASS	PASS
3.	Zip-code and category search	Tester shall input the prescribed zip-code and issue category in the search bar and the results shall be validated through comparison with the expected output	94112 with “Pollution” category selected	1 result with retrieved item having “Pollution” category and 94112 zip code	PASS	PASS

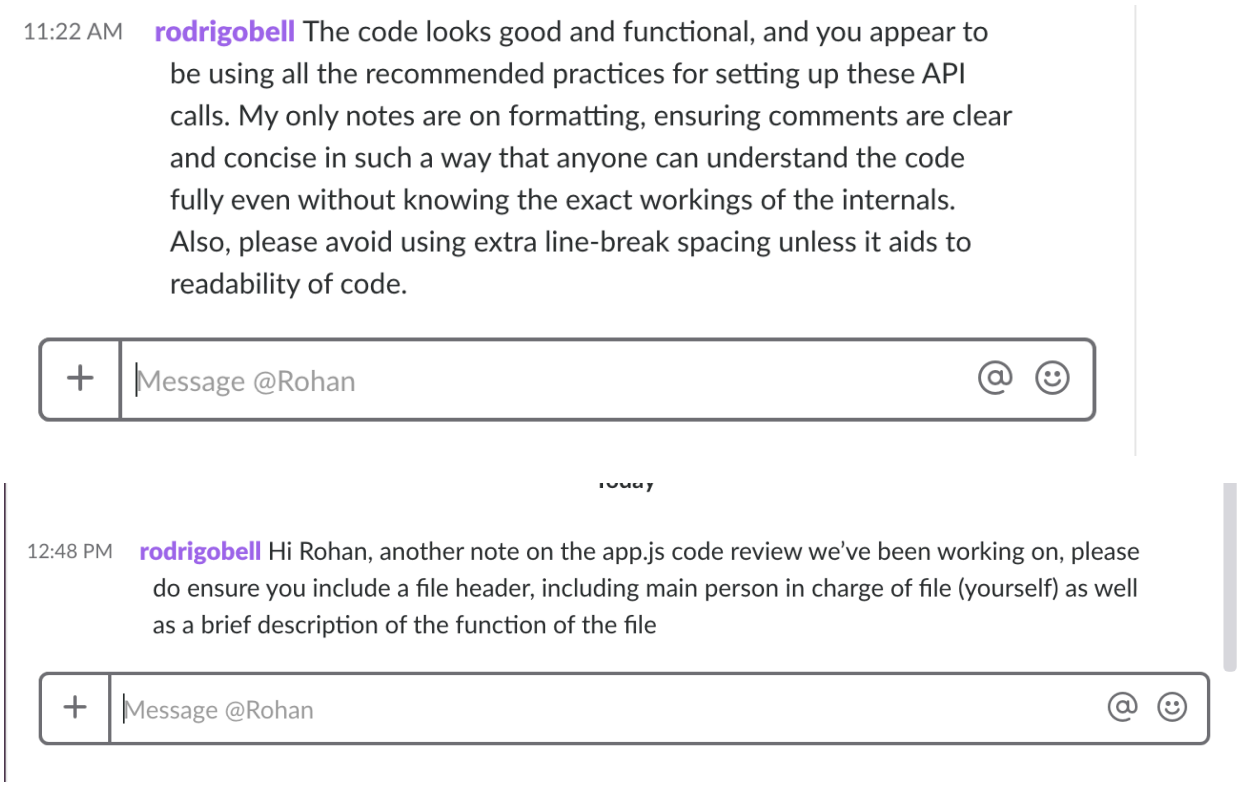
Code Review

A code review was conducted on the basic implementation the Node.js app entry point. Specifically, this code review shows the code to setup our MySql connection to the database.

In order to ensure clear code, indentation styles is used with 4 spaces for indentation.

Feature: Database setup

Submitted by: Rodrigo Bell



/app.js

```
var app = express();

/* Database Connection Info */
var connection  = require('express-myconnection');
var mysql = require('mysql');

// required for passport
app.use(session({
    secret: 'vidyapathaisalwaysrunning',
    resave: true,
    saveUninitialized: true
} )); // session secret
app.use(passport.initialize());
app.use(passport.session()); // persistent login sessions
app.use(flash());


app.use(

    connection(mysql,{

        host: "us-cdbr-iron-east-05.cleardb.net",
        user: "b3220b75dccc0a",
        password: "ddd8323b",
        database: "heroku_d6fcf8fd2312a32",
        multipleStatements: true

    },'pool') //or single

);

// view engine setup
app.set('views', path.join(__dirname, 'views'));
app.set('view engine', 'ejs');
```

Self-check: Security

Major Assets Being Protected

- Name
- Username
- Password

Passwords

Passwords are hashed, and no plain-text passwords are stored in the database. Name and username are protected and only accessible by database admin.

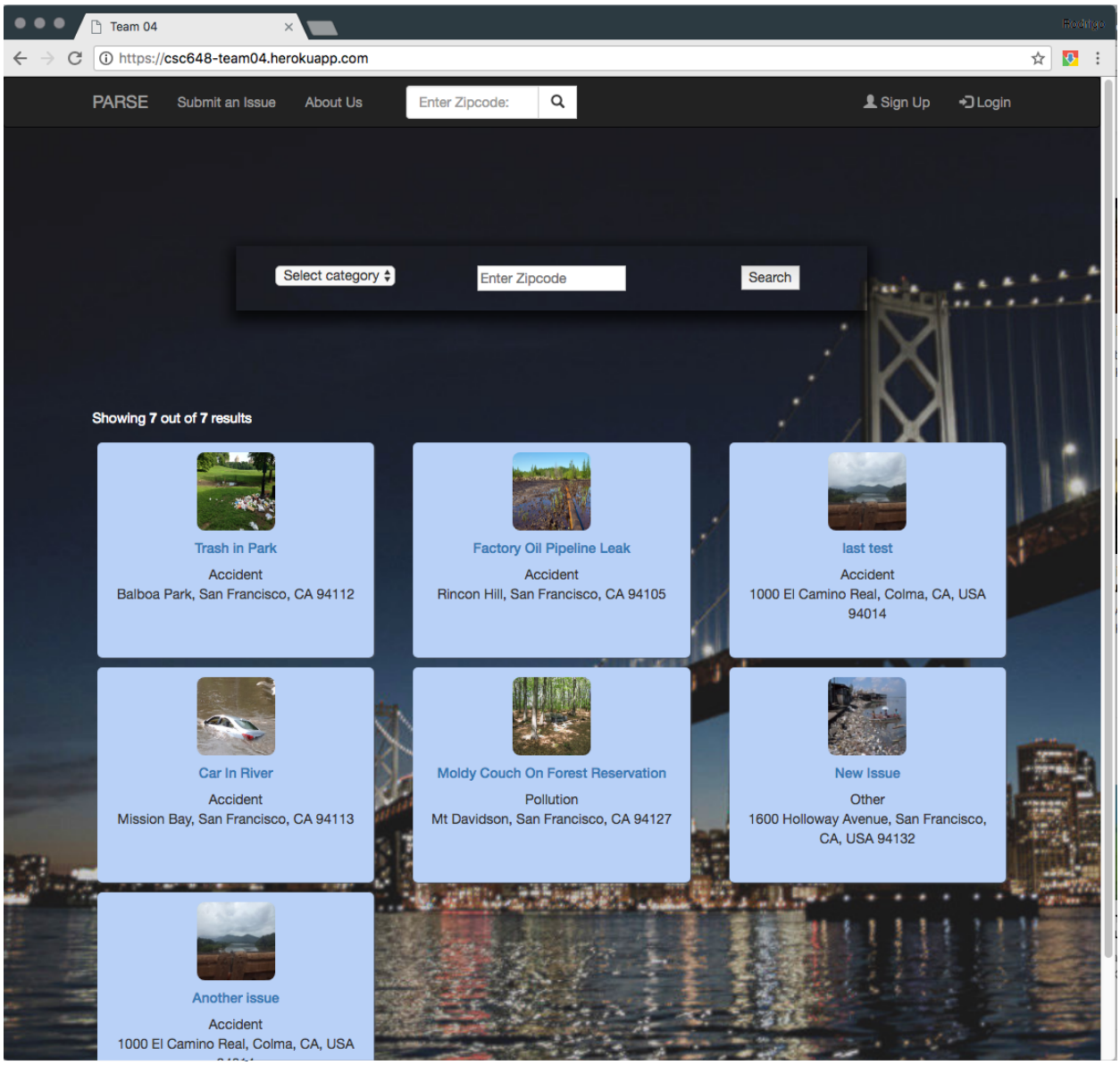
```
// Field validation
req.checkBody('Name', 'Name field cannot be empty').notEmpty();
req.checkBody('username', 'Username cannot be empty').notEmpty();
req.checkBody('password', 'Password cannot be empty').notEmpty();
req.checkBody('password', 'Password must be at least 8 characters long').len(8, 100);
req.checkBody('password2', 'Password must be at least 8 characters long').len(8, 100);
req.checkBody('password2', 'Passwords do not match').equals(req.body.password);
```

Self-check: Non-functional Requirements

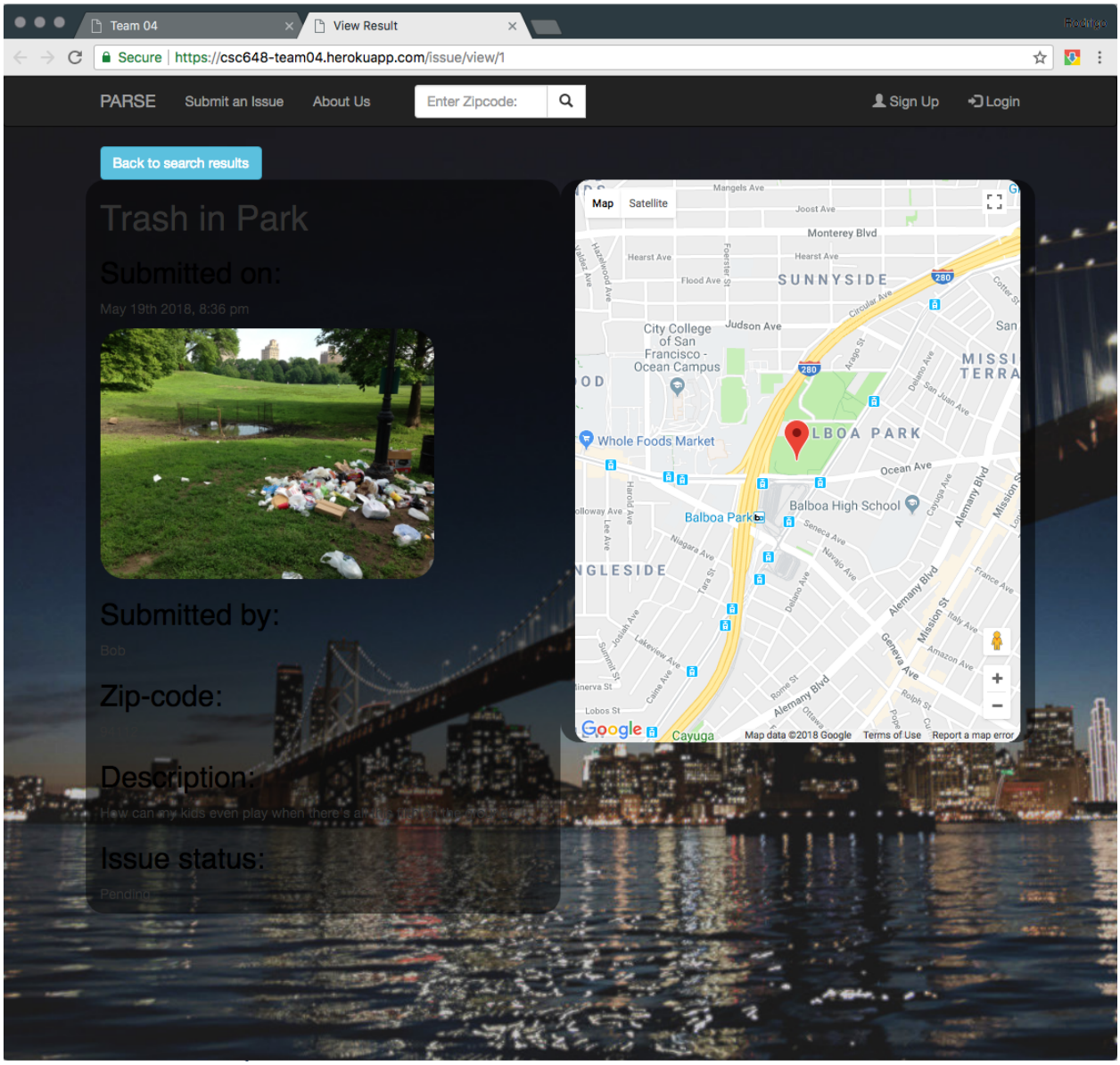
1	Application shall be developed and deployed using SW stack approved by class CTO Anthony Souza.	DONE
2	Application shall be hosted and deployed on Heroku as approved by Anthony.	DONE
3	Application shall be optimized for standard desktop/laptop browsers, i.e., must render correctly on the latest versions of all major browsers: Chrome, Safari, Mozilla	DONE
4	Application shall have responsive UI code so it can be adequately rendered on mobile devices but no mobile app is to be developed.	DONE
5	Data shall be stored in the MySQL database on the server in the team's Heroku account.	DONE
6	Maps showing location of the environmental issue shall be provided.	DONE
7	No more than 50 concurrent users shall be accessing the application at any time.	DONE
8	Privacy of users shall be protected and all privacy policies will be appropriately communicated to the users.	ON-TRACK
9	The language used shall be English.	DONE
10	Application shall be very easy to use and intuitive. No prior training shall be required to use the website.	DONE
11	Google analytics shall be added.	DONE
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.	DONE

Product Screenshots

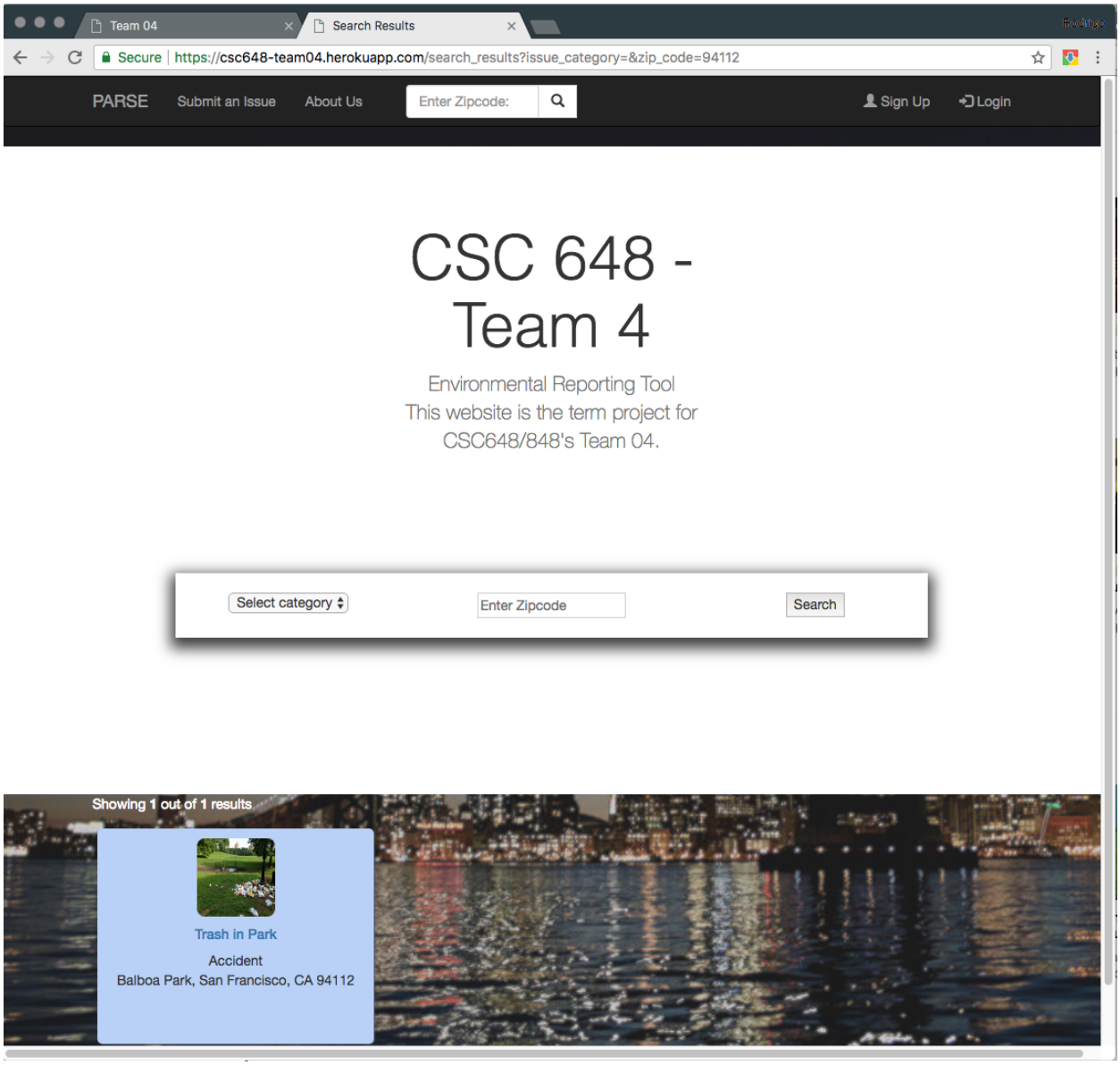
Home Page



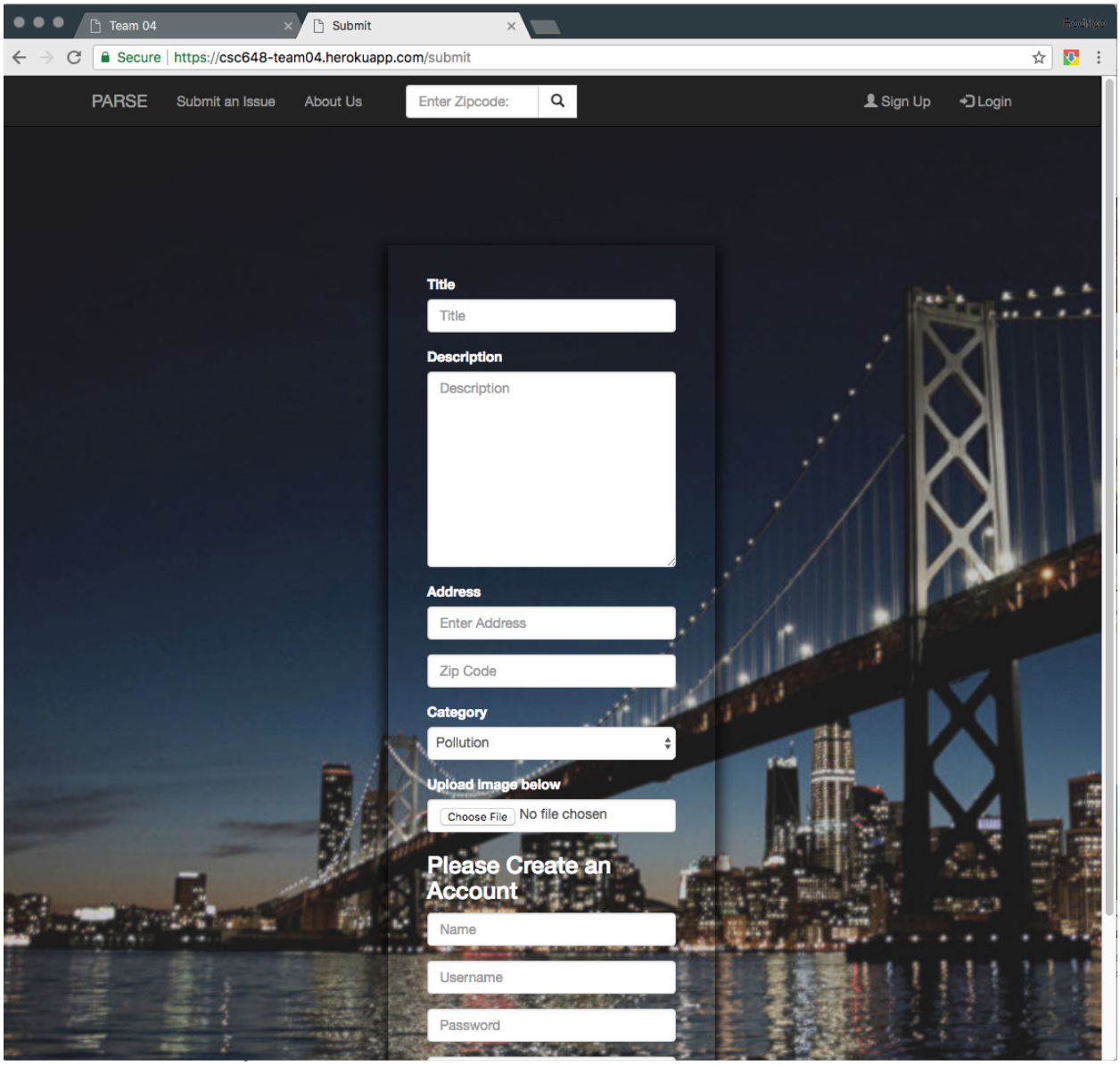
Issue Details Page



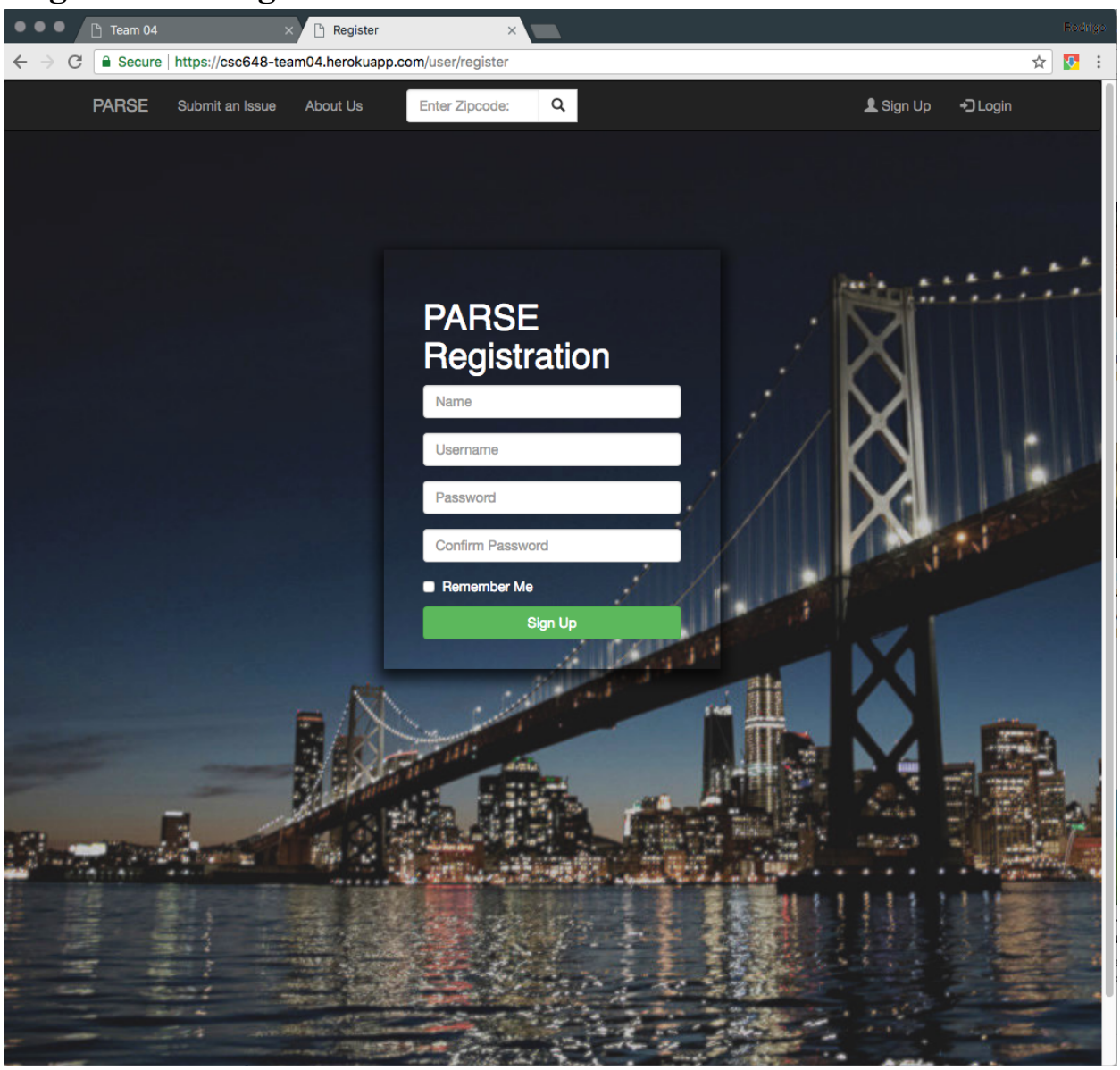
Search Results Page



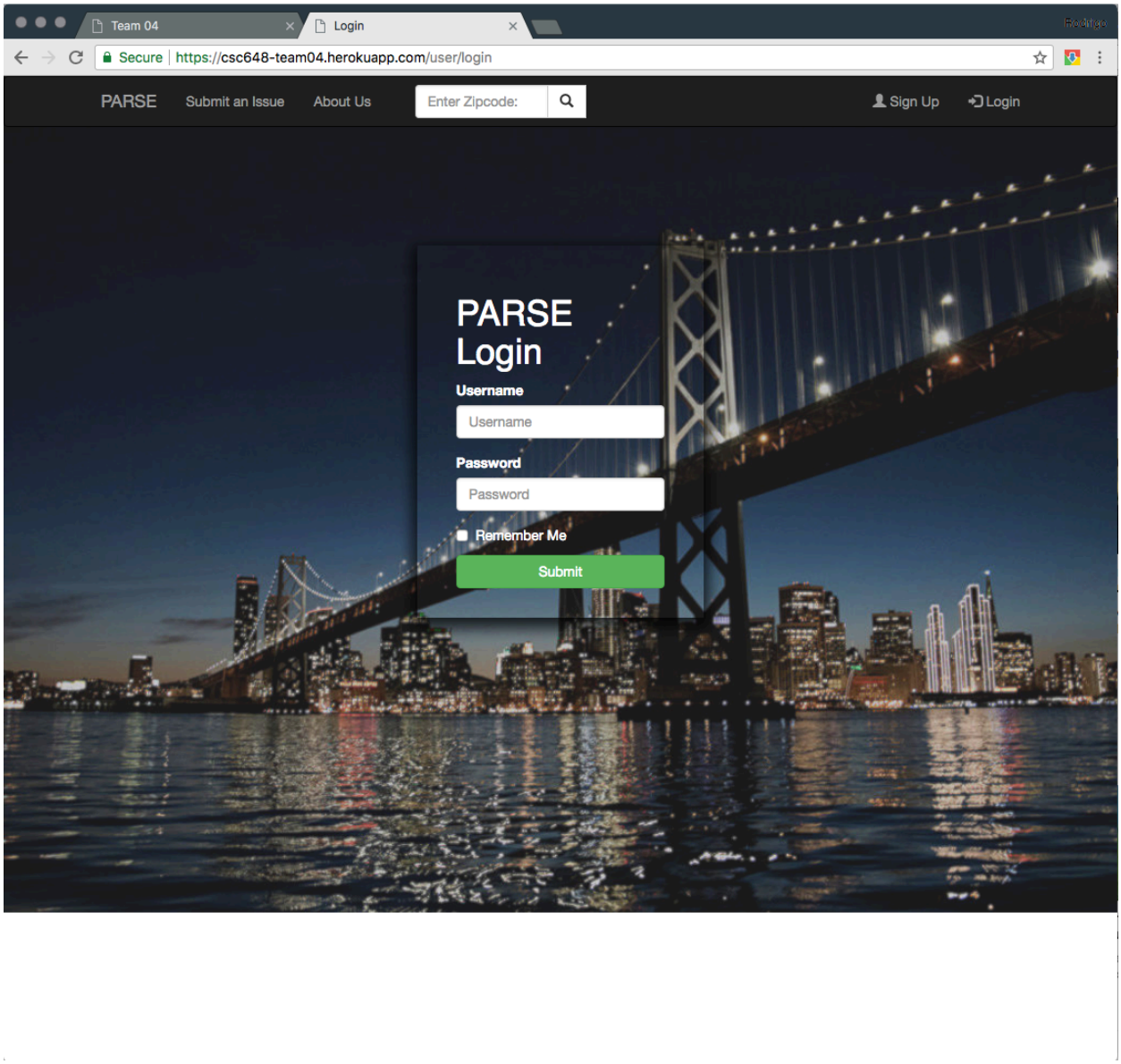
Issue Submission Page



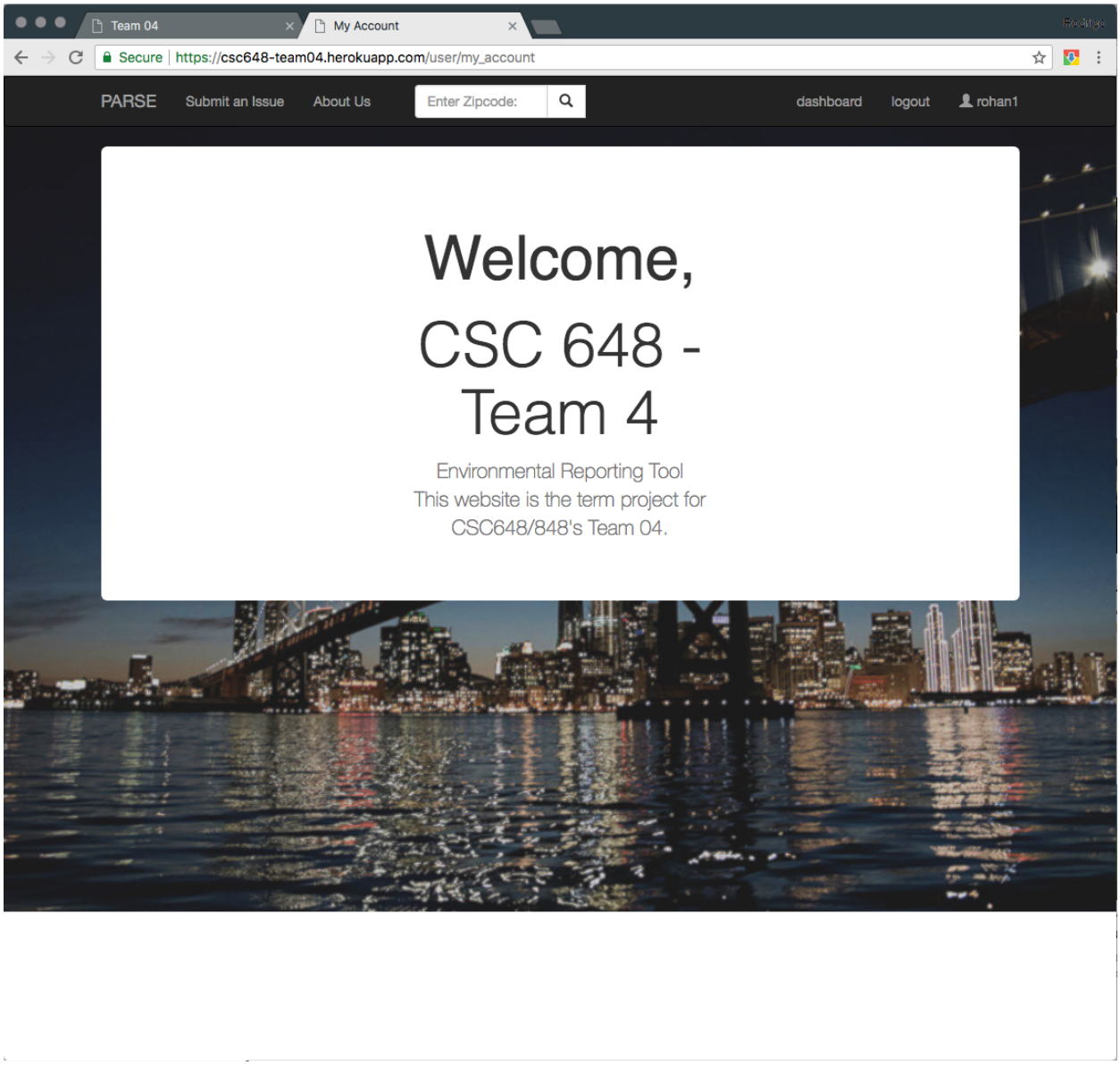
Registration Page



Login Page


Team 04
Login
Secure | https://csc648-team04.herokuapp.com/user/login
PARSE
Submit an Issue
About Us
Enter Zipcode:
Sign Up
Login
PARSE
Login
Username
Username
Password
Password
Remember Me
Submit

User Logged In Welcome page



Database Screenshots

User

user_id	name	email	phone	password	isAdmin
		NULL	NULL		0
101	Bob	NULL	NULL	$2a$10$G7if600DORlzWkBBh.Rzz.n/pW.ipesDS...	0
201	Sarah	NULL	NULL	$2a$10$G7if600DORlzWkBBh.Rzz.n/pW.ipesDS...	1
301	John	NULL	NULL	$2a$10$G7if600DORlzWkBBh.Rzz.n/pW.ipesDS...	0
AnotherTest1	Dion	NULL	NULL	$2a$10$G7if600DORlzWkBBh.Rzz.n/pW.ipesDS...	1
dtest	dion	NULL	NULL	$2a$10$rlohli6/0oxr/AWtOeqZ.un2/k1E45Cf7A...	0
ionnv	ion	NULL	NULL	$2a$10$ICJAaOAAdCOv.O3BPpkPouFAPEizWis...	0
rohan1	rohan	NULL	NULL	$2a$10$lf18BxonAOvkhMr8U6DKK./6CXqVY0M...	0
NULL	NULL	NULL	NULL	NULL	NULL

Status

id	index	status
1	0	NotAproved
2	1	Pending
3	2	Completed
NULL	NULL	NULL

Issues

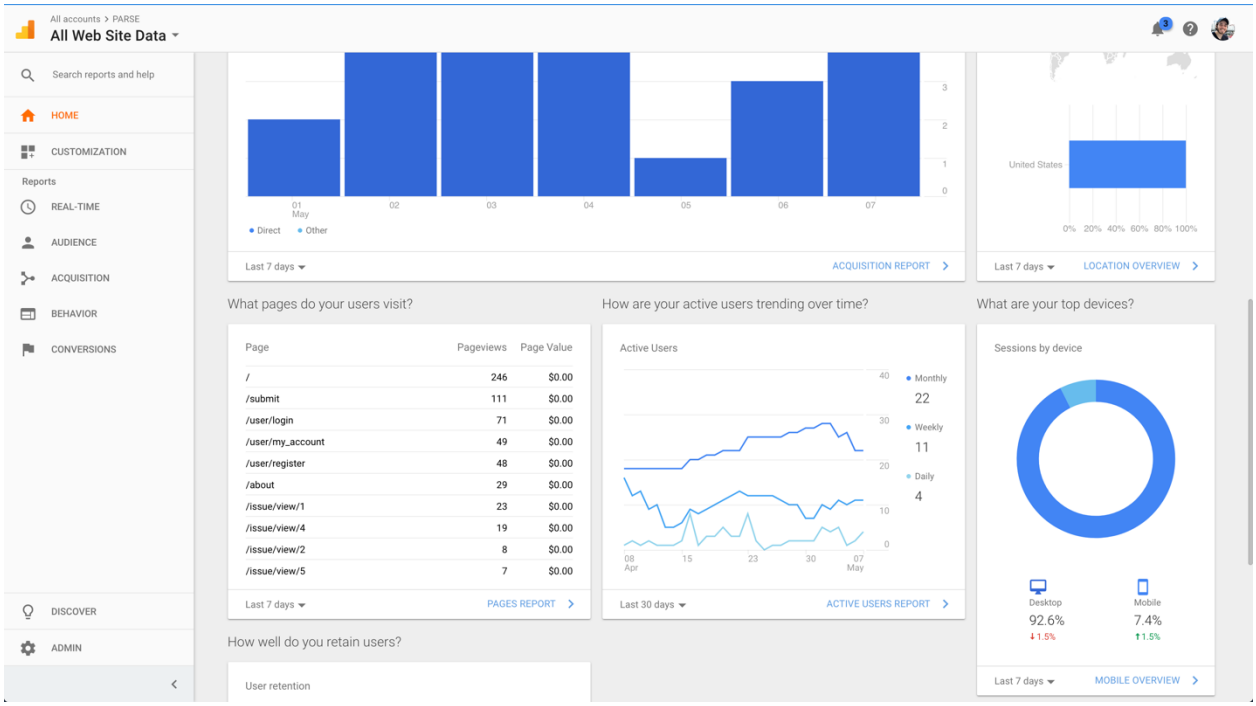
id	title	description	zipcode	image
1	Trash in Park	How can my kids even play when there's all this ...	94112	/images/issue_images/Park-Trash.jpg
2	Factory Oil Pipeline Leak	All this oil is killing the fish in the ocean. My favo...	94105	/images/issue_images/pipeline_leak.jpg
3	Car In River	Some dude literally parked his car in the river an...	94113	/images/issue_images/Car-in-river.JPG
4	Moldy Couch On Forest Reservation	A person thought this'd be a good place to start...	94127	/images/issue_images/couch.jpg
5	Radioactive Waste on Playground	My kids are growing more limbs!!!! Someone cle...	94137	/images/issue_images/radioactive.jpg
10	new issue	test	94132	https://res.cloudinary.com/csc648-team04/

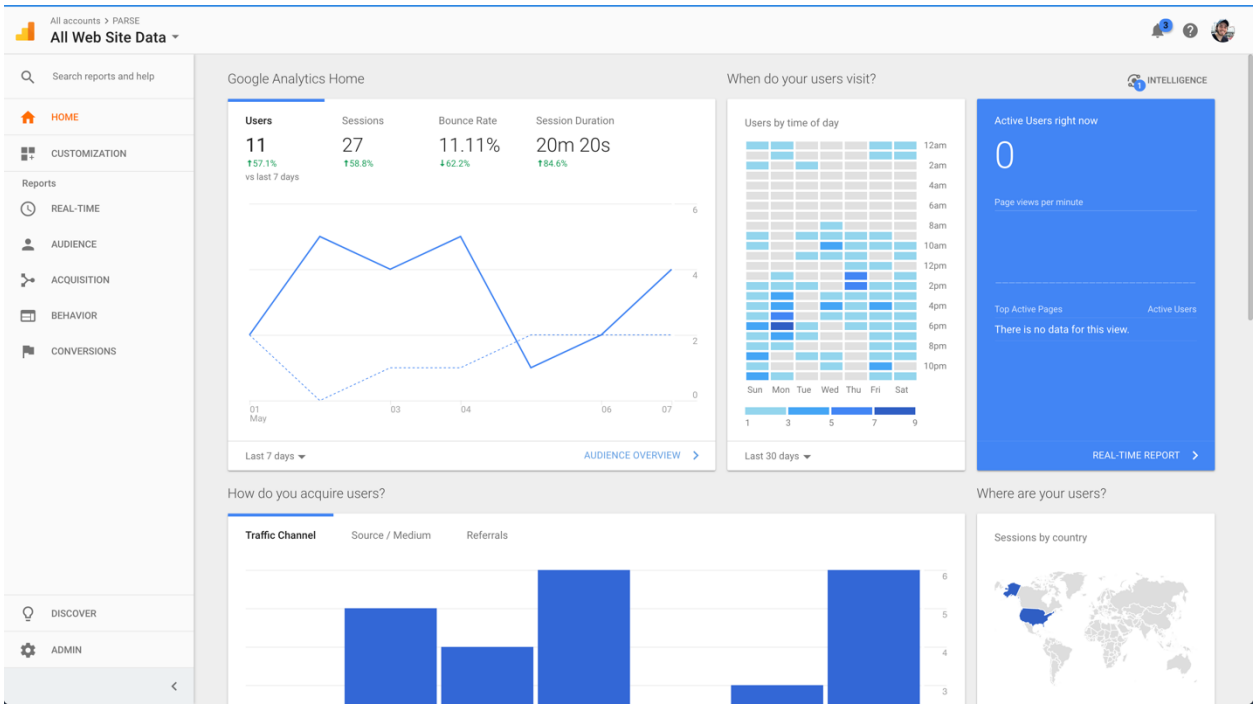
user_id	latitude	longitude	category	status	thumbnail	address
101	37.723584	-122.445814	1	3	/images/thumbnails/Park-Trash_thumb.jpg	Balboa Park, San Francisco
201	37.787890	-122.391489	2	2	/images/thumbnails/pipeline_leak_thumb.jpg	Rincon Hill, San Francisco,
101	37.774866	-122.393384	2	2	/images/thumbnails/Car-in-river_thumb.JPG	Mission Bay, San Francisco
301	37.738471	-122.456299	1	2	/images/thumbnails/couch_thumb.jpg	Mt Davidson, San Francisc
301	37.791365	-122.399206	3	1	/images/thumbnails/radioactive_thumb.jpg	Mechanics Monument Plaz
	37.7215776	-122.47561539999998	1	3	http://res.cloudinary.com/csc648-team04/imag...	SFSU Station Cafe, Hollow
NULL	NULL	NULL	NULL	NULL	NULL	NULL

Category

id	name
1	Pollution
2	Accident
3	Other
NULL	NULL

Google Analytics Screenshots





Team Member Contributions

Rodrigo Bell <rbell594@gmail.com>
to Kamran, Michael, dlagos, Zhenru, Charlie, Rohan

Rodrigo:

- served as team lead
- assigned work to front/back end leads to be further delegated among each sub-team
- managed team communication
- edited and finalized milestone documents
- aided in initial back-end set up
- aided in setting up google analytics
- 159 lines of code contributed to Github

Charlie Tuttle
to dlagos, Michael, Zhenru, Rohan, me, Kamran

Charlie Tuttle:

- As front end lead, I helped my team when using github and bootstrap.
- Set up the UI for the city manager's dashboard page.
- Set up the display of the home page's search results in inline block format.
- Helped organize the website's repeated code into modular templates.
- Contributed to search bar display and form validation.
- Contributed to the general styling of the website.
- Added 1,993 lines of code to the Github repo.

Rohan Patel
to me, dlagos, tuttle.charlie, kamkhadivi, mschwiebs, zhuang4

Rohan Patel:

- Served as back-end lead
- Worked on initial node js server set up
- Assisted with initial database setup
- Implemented search functionality
- Integrated google maps api
- Implemented issue posting functionality, including uploading image to filesystem
- Setup filesystem on cloudinary
- 1,727 lines of code on github

Dion Matthew Morales Lagos
to Charlie, Kamran, Michael, Zhenru, Rohan, me

Dion Matthew Lagos (Back-End Developer):

- Developed user registration and login system with password encryption
- Contributed to initial database setup, including connecting on the back-end
- Contributed to issue posting functionality
- Contributed to city manager account dashboard, including the ability to update an issue's status in the database
- Assisted Front-End team with proper form validation
- 1,885 lines of code

Kamran Khadivi-Dimbali
to me

---------- Forwarded message ----------
From: **Kamran Khadivi-Dimbali** <kamkhadivi@gmail.com>
Date: Sat, May 19, 2018 at 9:07 PM
Subject: Re: Charlie Tuttle contributions
To: Charlie Tuttle <tuttle.charlie@gmail.com>
Cc: Michael Schwiebert <mschwiebs@gmail.com>, dlagos@mail.sfsu.edu, Zhenru Huang <zhuang4@mail.sfsu.edu>, Rohan Patel <Rohan.8594@gmail.com>

Kamran Khadivi-Dimbali:

-Set up UI for Submit an issue page
-Set up UI Login and Registration pages
-UI mockups
-Contributed to Login/Registration form validation
-Contributed to Milestone documentation
-Contributed to the general styling of the website.
-302 lines of code

Michael Schwiebert
to Charlie, me, dlagos, Kamran, Zhenru, Rohan

Michael Schwiebert:

- Helped design database schema
- Implemented navigation bar
- Contributed to high level designs of major web pages
- Implemented singe issue details page
- Extensive Q/A testing
- Contributed to general styling of website

Zhenru Huang
to Kamran, Michael, Dion, Charlie, Rohan, me

Zhenru Huang:

Help to initial heroku server

Set up the database

Add foreign keys and manage the database tables

Implement the thumbnail function

Documentations
923 lines of code.

Post Analysis

This project was a great exercise in learning how to come together as a team on a large scale software engineering project. Many of us had multiple scheduling conflicts outside of class times which prevented us from finding a recurring time to meet as a whole team. We overcame this by scheduling work meetings on the fly at different times, and in smaller groups (e.g. front-end specific, back-end specific meetings). We also used Slack as our primary means of communication which we found to be very effective.

Going back, we could have improved our communication even more by setting a scheduled standup earlier in the project. We also could have chosen a different framework that is better documented and easier to work with.

Overall, the project went very smoothly, and every team member was able to pull their own weight, which meant that we were very productive as a team.