SW Engineering CSC648/848
Section 01
Spring 2018
Team 02

ecoHero

Milestone 2

March 18, 2018

Andrew Wong
Connor Aguilera
Marcelo Coronado
Preston Segura
Emilio Quiambao
Patrick Shi
Sahas Arora

Table of Contents

1. Data Definitions

[Guest]: An unregistered user. Can only browse the website, cannot post, reply and edit.

[Registered Member]: A registered user. Can browse, submit, and reply to hazard posts. Submitting, editing or deleting their own posts requires approval from an administrator.

[Administrator]: Appointed by the website owner. Can post announcements and regular posts. Admins check registered members' posts before being shown on the website. Able to permanently or temporarily ban users.

[Environmental Agents]: A special user, environmental company user. Can observe incoming posts by registered users. Agents are able to update a post's status as either "IN PROGRESS" or "FINISHED". By default, a post will be seen as "PENDING".

- IN PROGRESS: a hazard status to indicate a post is in progress
- FINISHED: a hazard status to indicate a post is complete
- PENDING: the default hazard status to indicate an agent has not checked the post

[Post]: Contains description, images, date, time, location (Google Map), and comments about the hazard. The post itself has a 'status' and 'approval'.

- Title: the title of the hazard the user inputs
- Description: a description of the hazard the user inputs
- Image: images of the hazard the user uploads (formats: JPEG, JPG, PNG, max size: 1mb)
- Time: the time the post has been submitted

- Location: the location of the hazard (links to Google maps)
- Type: the type of hazard the user inputs
- Comments: text replies by registered members
- Approval: Approval by the admin if post is approved or denied for public posting
- Status: the status of the hazard updated by an environmental agent
  (see Environmental Agent definition for types of statuses)

[User Record]: Data on a user which contains their username, password, and date from when they registered. Also contains a status on if their banned.

- Username: Name to log in and identity when posting or replying
- Password: Used to log in. Minimum 8 characters.
- Register date: The date the user has registered an account
- Ban: Status if a user is temporarily or permanently banned

2. Functional Requirements

Priority 1

Unregistered users shall be able to...

1. Navigate and view approved environmental hazard posts
2. Sort or filter the posts by date, title, or area
3. Search for posts by title or description
4. Log in or create a registered account

Registered users shall be able to...

5. Do all what an unregistered member can do
6. Post environmental hazards with a title, description, location, date, and image
    a. And categorize the hazard

    * Must be approved by an administrator
7. Edit or delete their previous posts

    * Must be approved by an administrator

Administrators shall be able to..

8. Do all what a registered member can do
9. Approve or reject a pending registered user’s post
10. Approve or reject a pending edit or deletion of a registered user’s post
11. Delete a post with reason
12. Ban a user temporarily or permanently with reason

Environmental Agents shall be able to..

13. Do all what a registered member can do
14. Confirm an environmental post
15. Update a post’s status as “in progress” or “solved”

Priority 2

Registered users shall be able to...

16. ‘Bump’ a post
17. Comment on a post
18. Report a post

Administrators shall be able to..

19. Post an announcement

Priority 3

Registered users shall be able to...

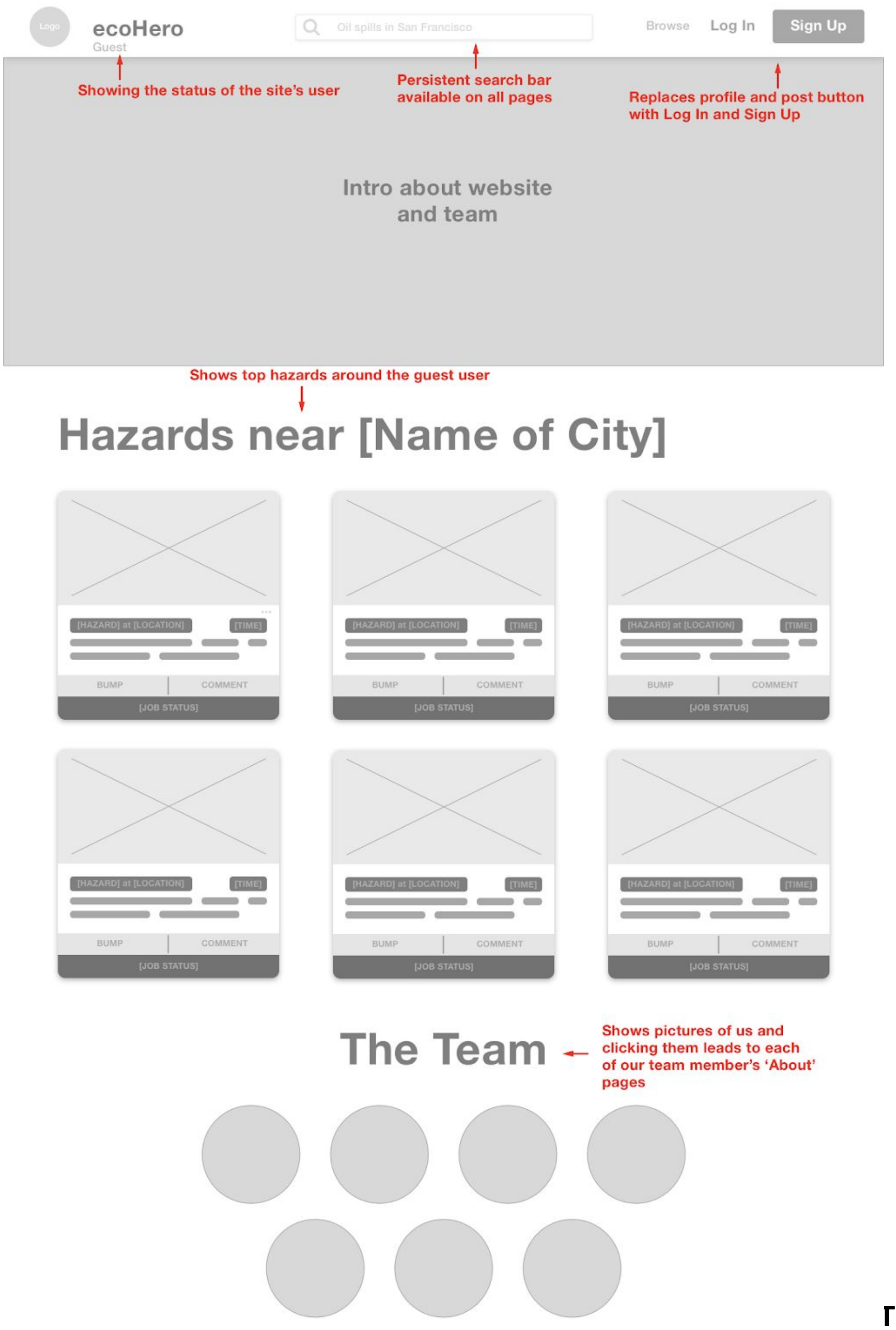
20. Switch between a ‘compact’ or ‘expanded’ view

Anyone shall be able to...

21. Utilize an ‘advanced search’
22. Share a post

3. UI Mockups

Home (Guest View): Show information about our site and hazards near the guest user.



Showing the status of the site's user

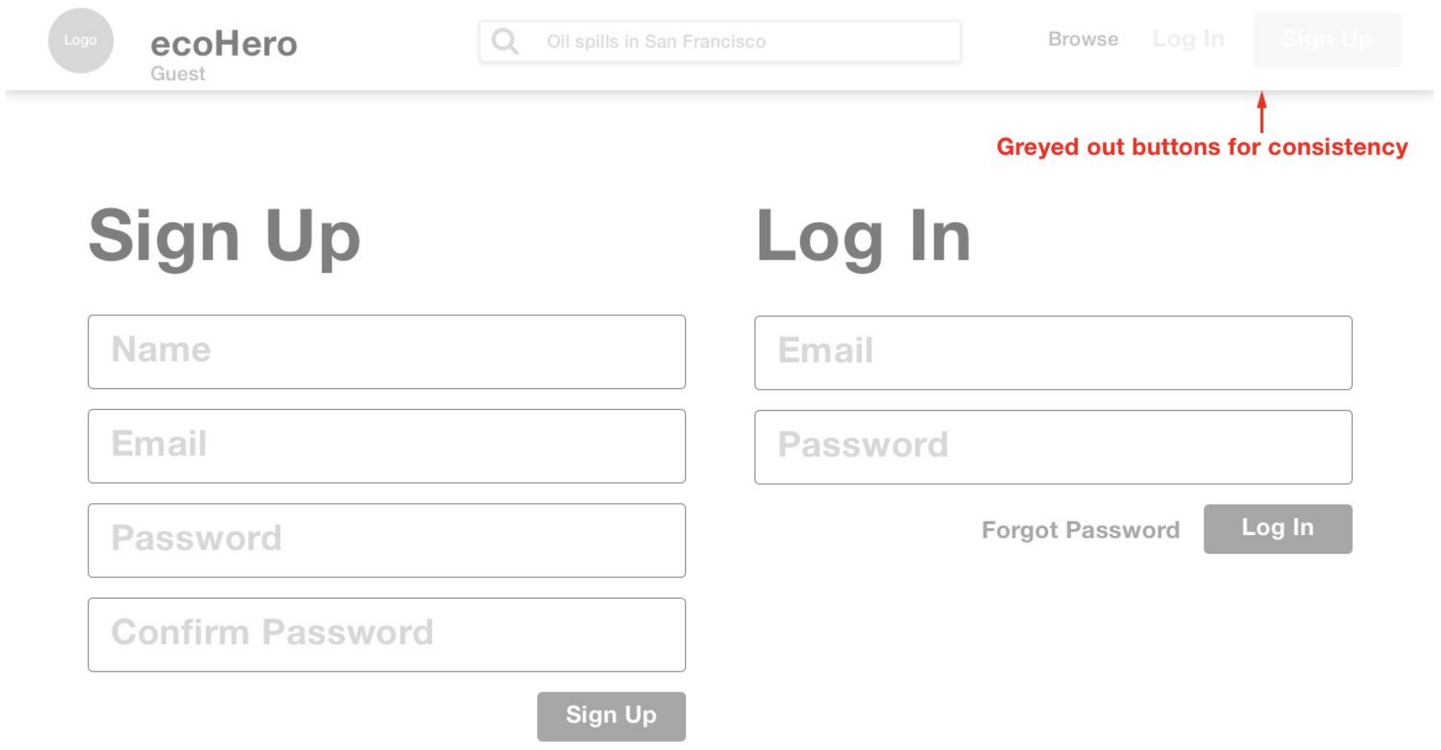
Persistent search bar available on all pages

Replaces profile and post button with Log In and Sign Up

Shows top hazards around the guest user

Shows pictures of us and clicking them leads to each of our team member's 'About' pages

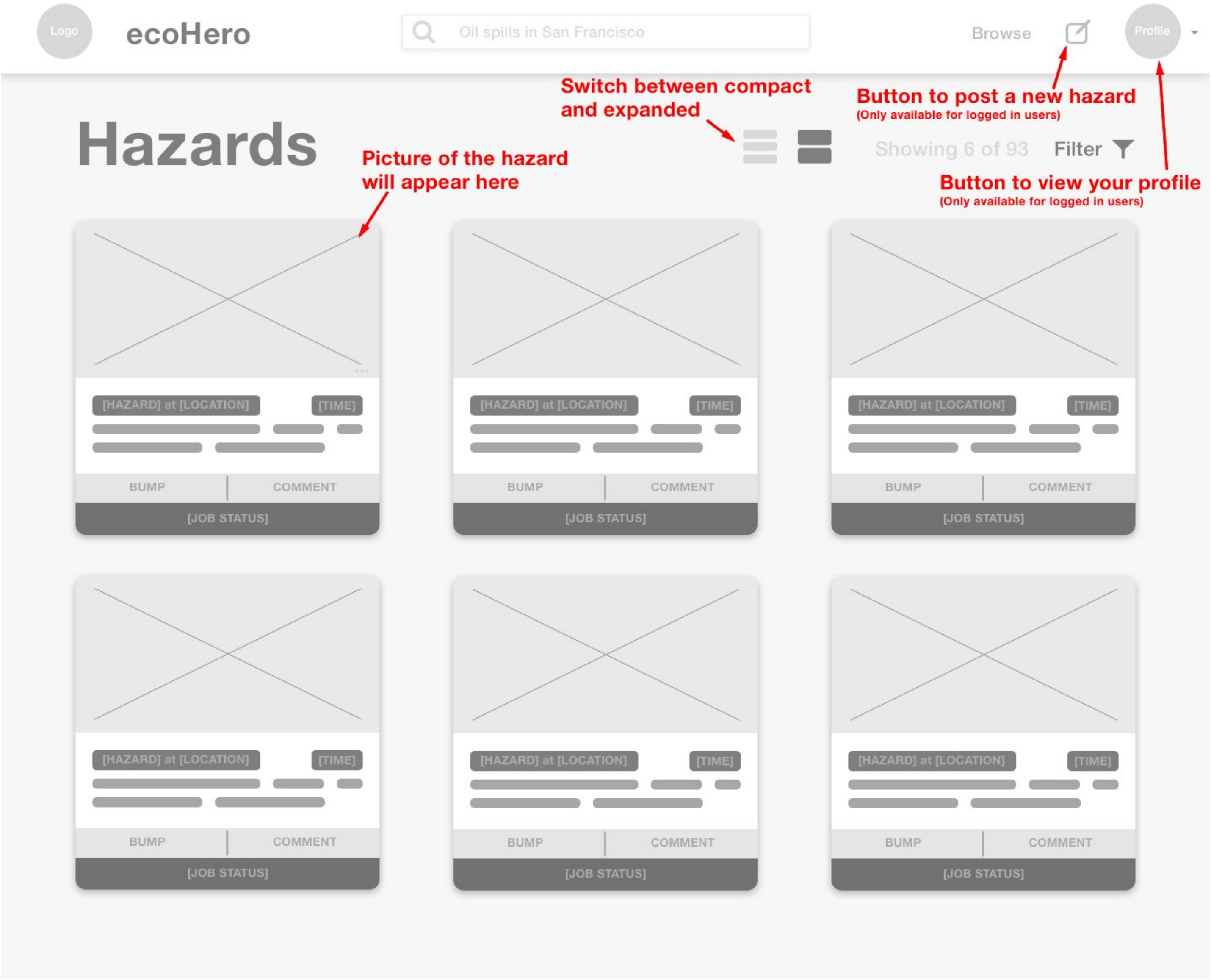
Sign Up/ Log In (Guest View)



Greyed out buttons for consistency

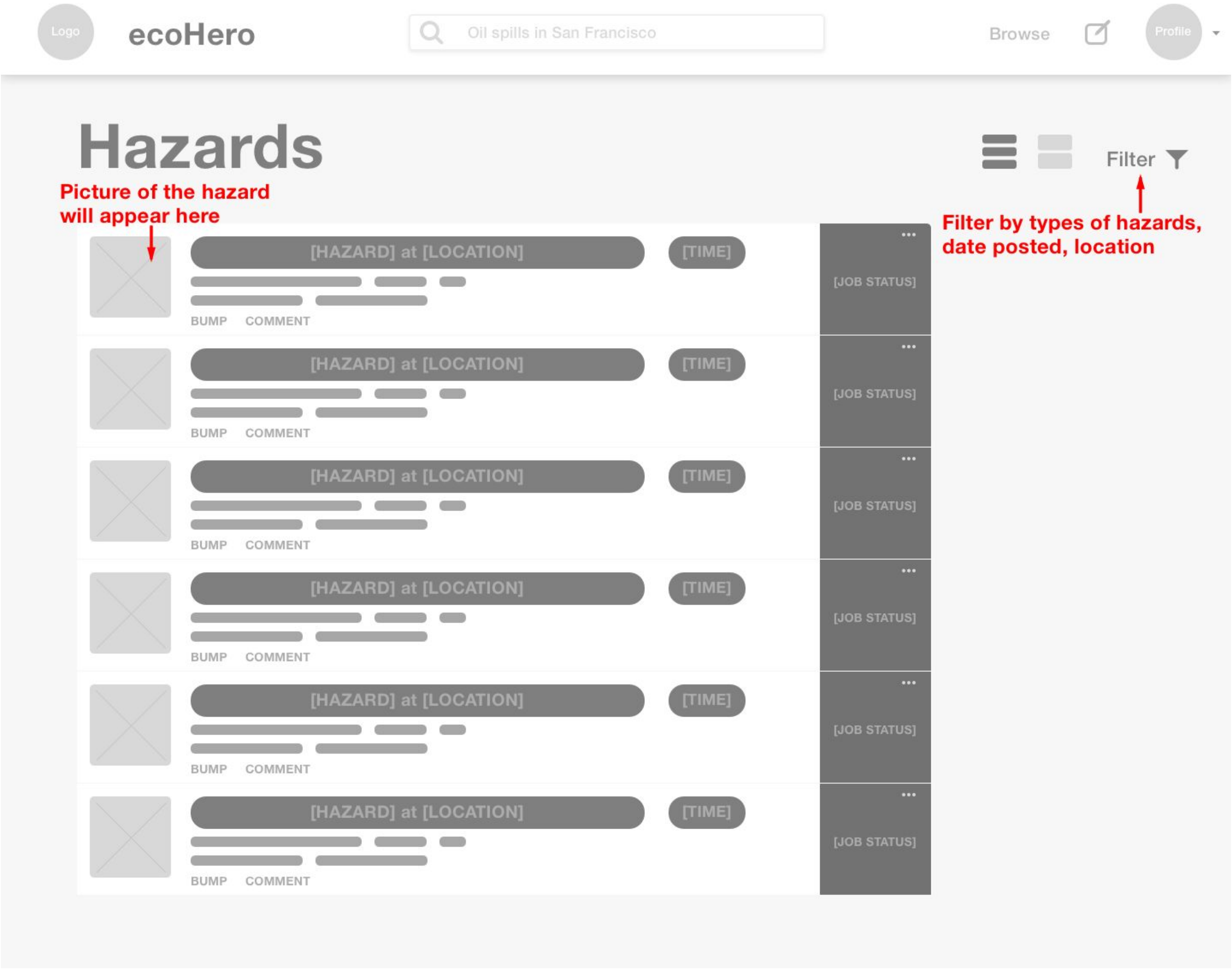
Browse Hazards Expanded (Signed In View)

- GUI view of our [POST] data showing: title, description, image, time posted, location described, type of hazard, any comments associated with the post, approval by the admin, status of the hazard


Logo
ecoHero
Oil spills in San Francisco
Browse
Profile
Hazards
Switch between compact and expanded
Button to post a new hazard
(Only available for logged in users)
Picture of the hazard will appear here
Showing 6 of 93
Filter
Button to view your profile
(Only available for logged in users)
[HAZARD] at [LOCATION]
[TIME]
BUMP
COMMENT
[JOB STATUS]

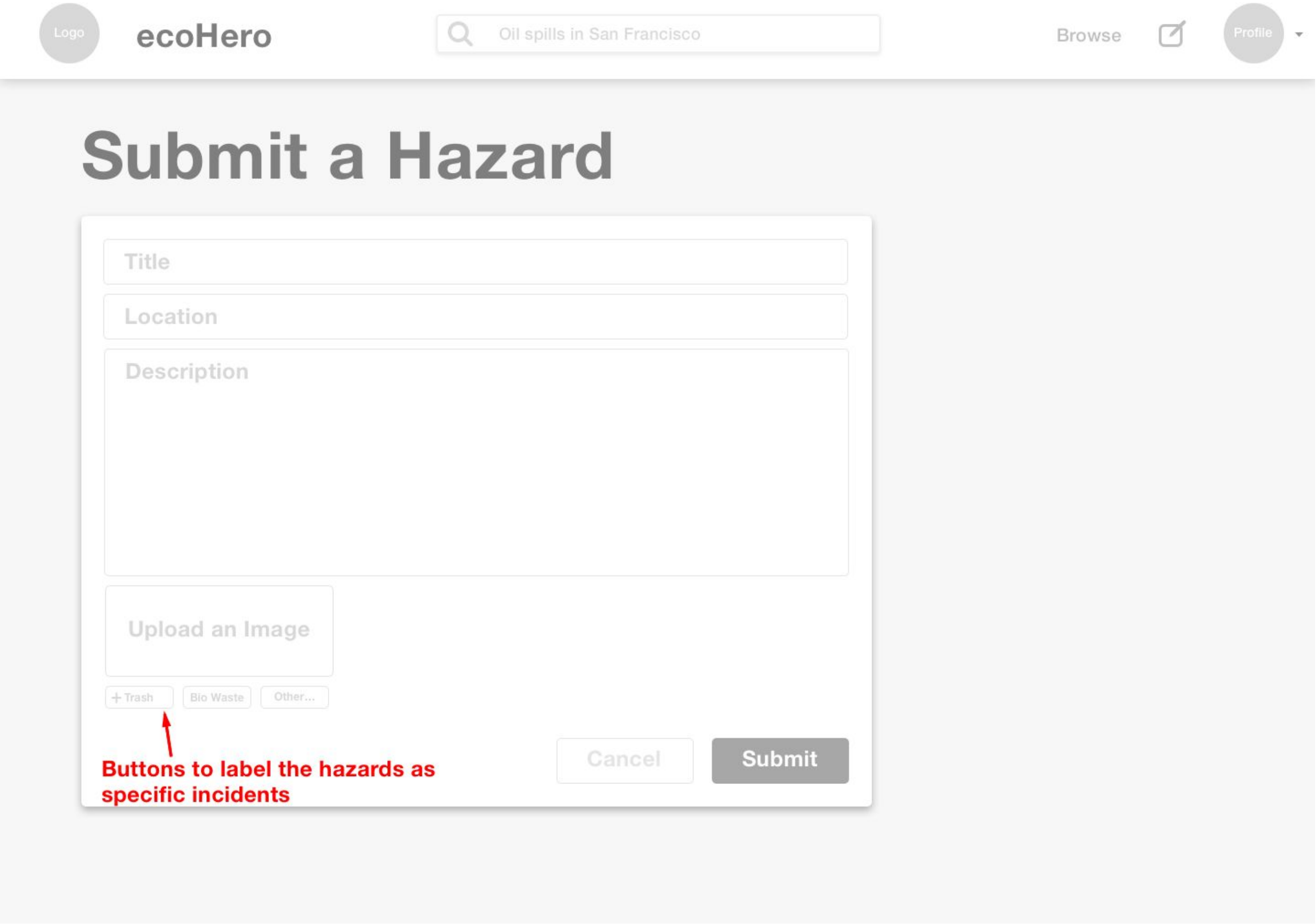
Browse Hazards Compact (Signed In View)

- GUI view of our [POST] data showing: title, description, image, time posted, location described, type of hazard, any comments associated with the post, approval by the admin, status of the hazard



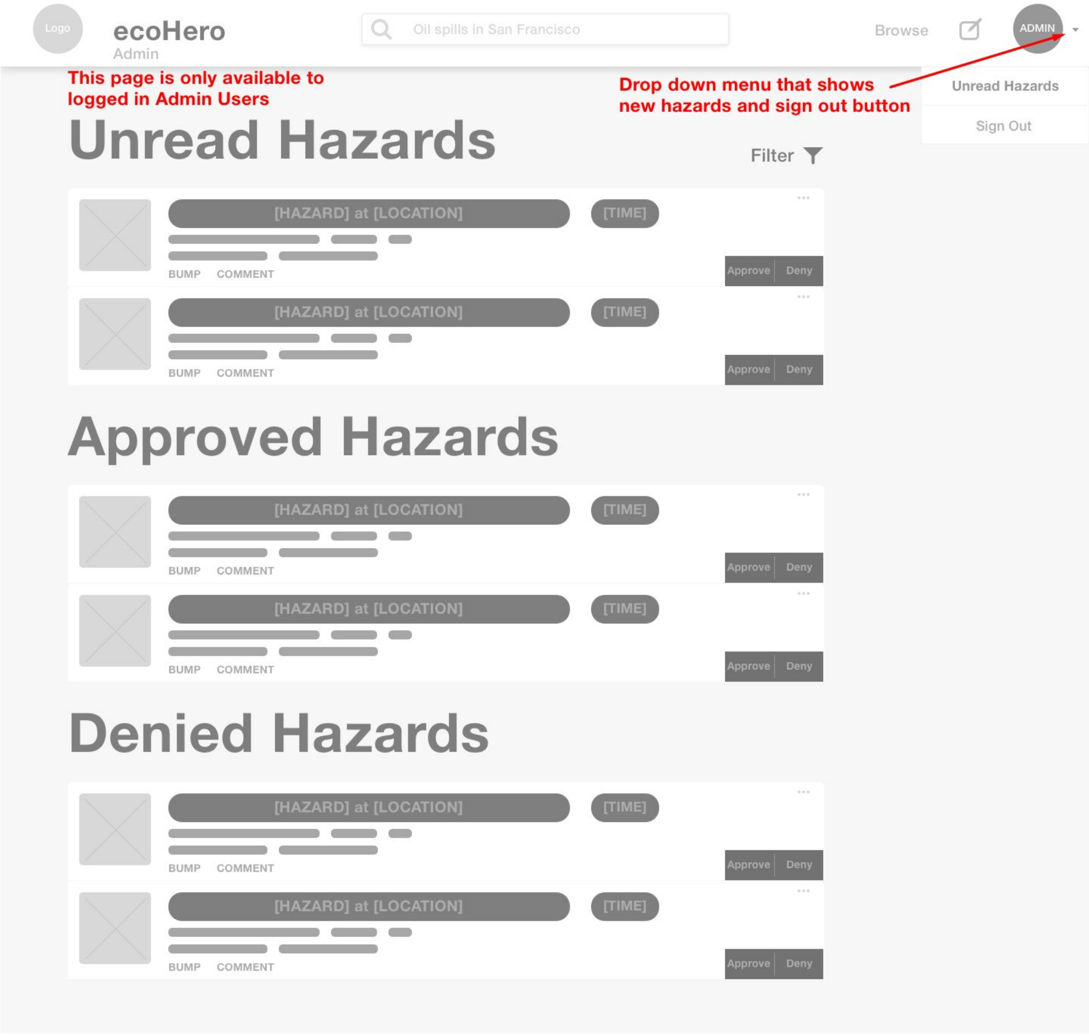
Submitting a Hazard (Signed In View)

- Produces a [Post] output
- Only available to signed in users



Approving Hazards (Admin View)

- Page showing the admin what [Post]'s are unread and need approval
- Shows the drop down menu that appears when clicking your profile in the upper right
- Gives the admin an Approve or Deny button for pending hazard [Post]'s

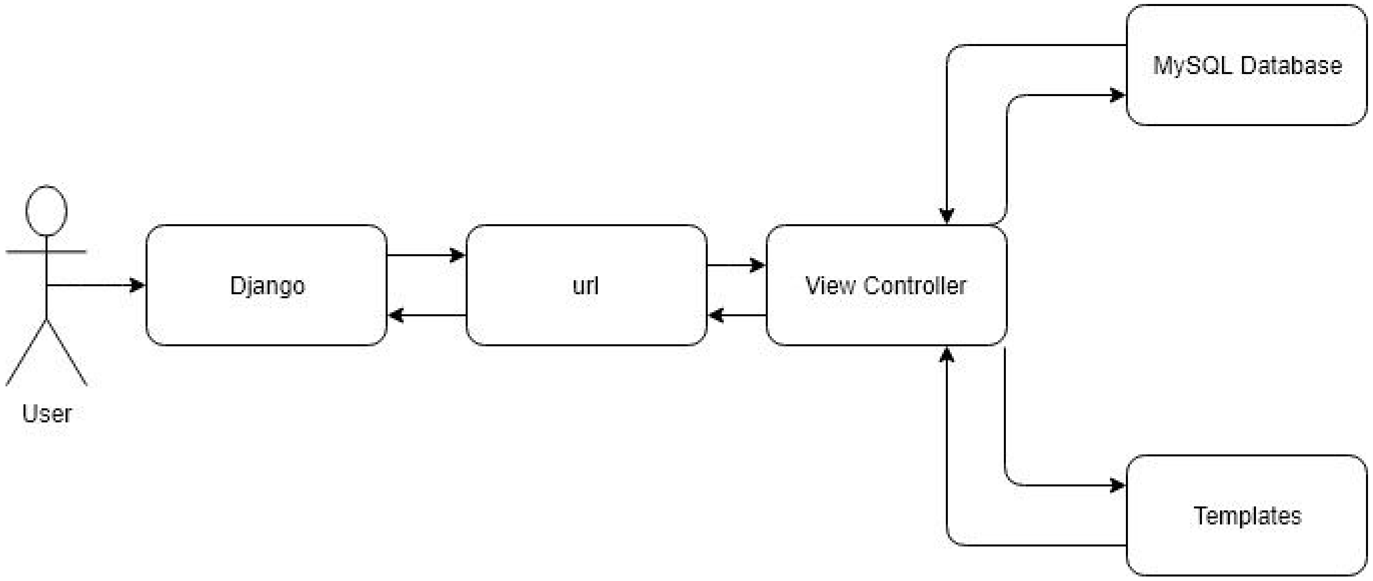


This page is only available to logged in Admin Users

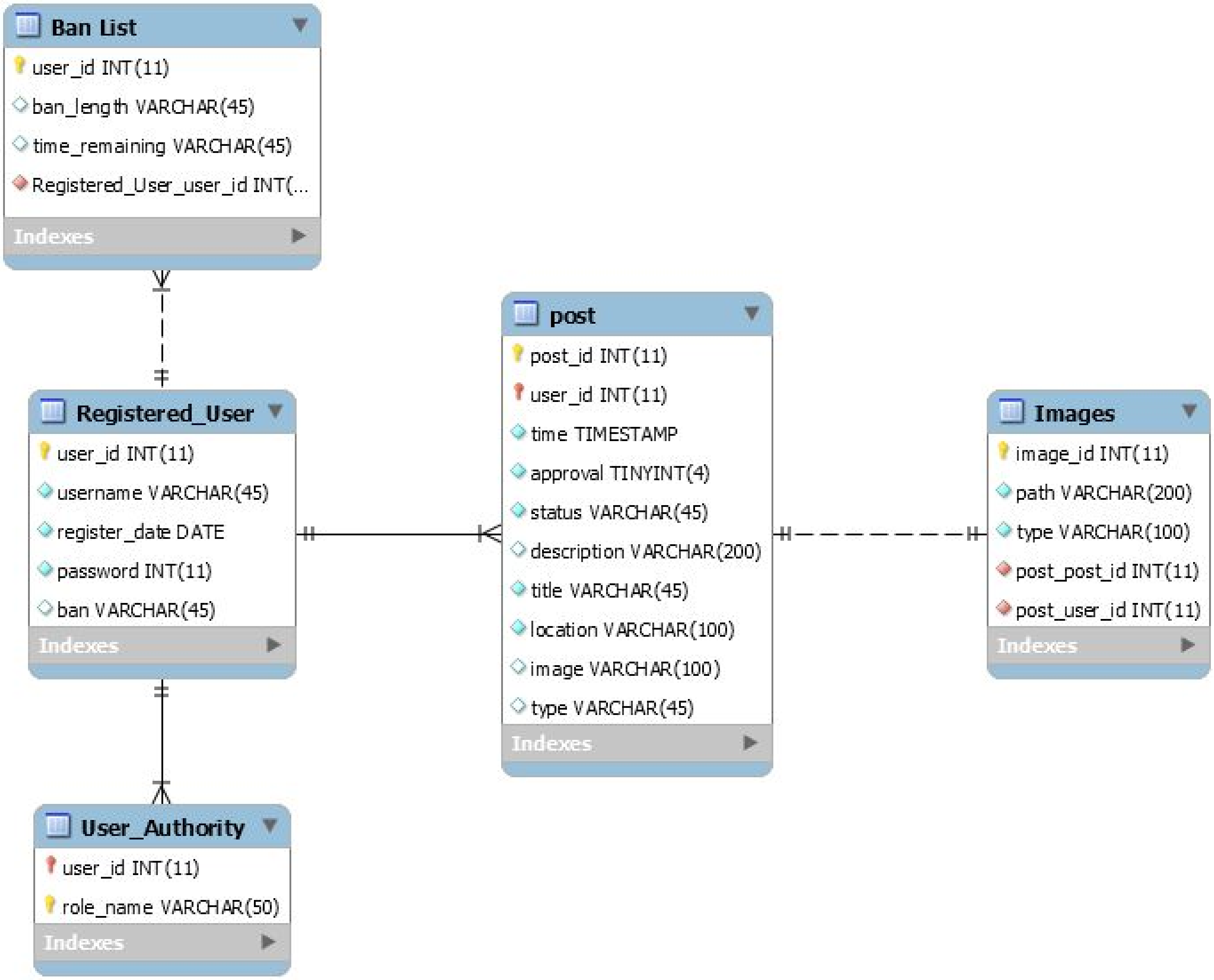
Drop down menu that shows new hazards and sign out button

4. High level Architecture / Database Organization

High Level Architecture:



DB organization:



Registered_User is for instances of user.

Different levels of privileges are listed by the table *User_Authority*, such as ‘Administrator’ and ‘Environmental Agent’. Posts made by user are defined by posts table.

Media Storage: File systems will be used to store images.

Accepted formats: JPEG, JPG, PNG Max size: 1mb

Search/filter architecture and implementation: In order to search, the website utilizes PHP’s mySQLi functions to communicate and maintain a connection to the database server. When a user inputs key words to the search bar and submits it, the key words would be passed as string named “search” and be brought to the ‘searchresults’ page.

```
<input name="search" type="text" size="40" maxlength="50"/>
```

In the ‘searchresults’ page, PHP will first try to successfully connect to the database. Once connected, the algorithm will first check if the string exists. If it does not, the user will be brought back to the same page (in this case, the home page). A case like this will usually happen if a user enters the search result page URL directly without submitting a search.

```
if(!isset($_POST['search'])) {header("Location:index_test.php"); }
```

With the user input, we can use PHP’s mySQLi functions to communicate SQL commands to the database and match the user’s input with any data stored in the database. In this case, we matched the user’s input with any of the stored hazard’s title or location using SQL’s “LIKE” function.

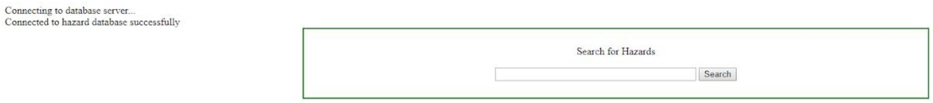
```
$search_sql= "SELECT * FROM hazards WHERE name LIKE '%" .$_POST['search']. "%' OR
location LIKE '%" .$_POST['search']. "%'";
```

With these functions, we will also be able to filter or sort posts by any of a post's attribute such as time, date, location, and so on.

When retrieving the data, the number of rows that matched will be read. If it is zero, we echo to the user that no hazards were found. If it's not zero, a do-while loop will be used. Every loop, a row's attributes will be echoed such as the title, description, and location. Images can be displayed by reading the posts unique ID (its primary key in the database). When the ID is read, an image file shares the same name which can be retrieved in the webserver's filesystem where it's stored. If it does not exist, a default picture will be used. The loop will end when there are no more rows to be read.

5. Content for Vertical Prototype

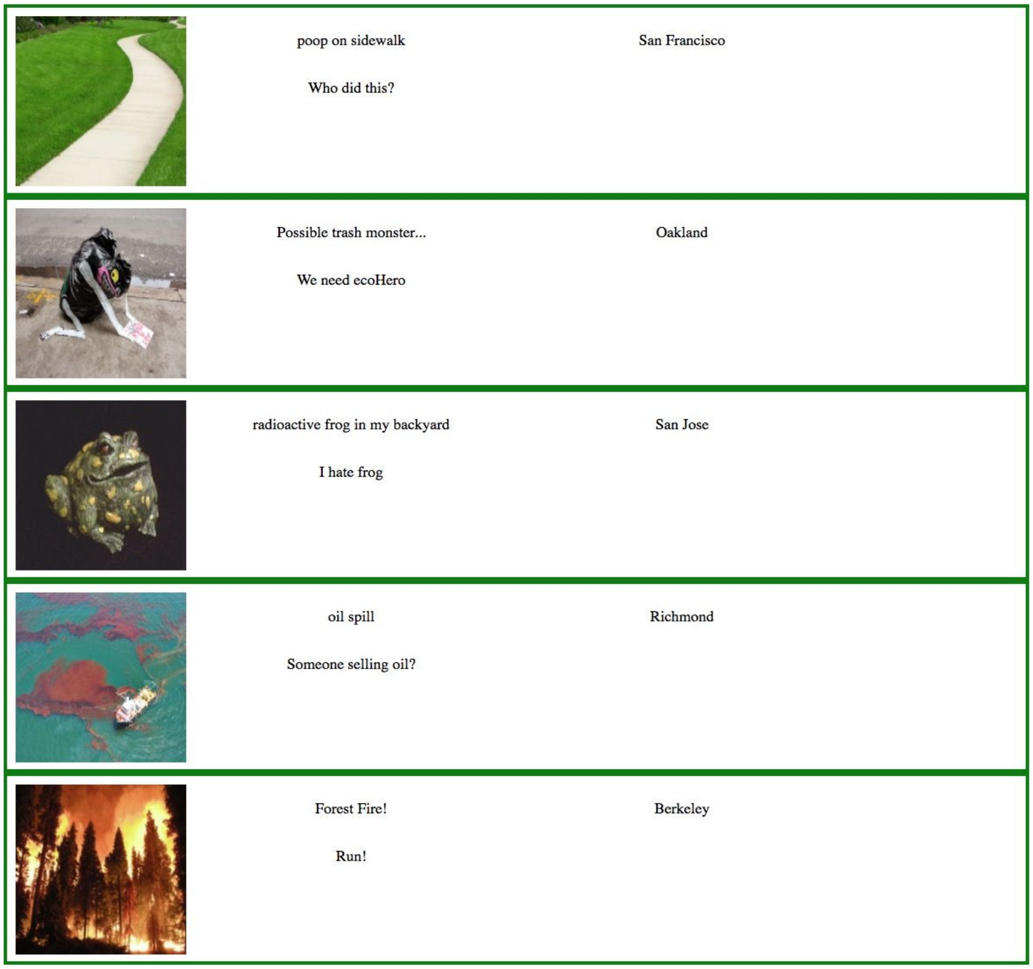
Search Bar



Connecting to database server...
Connected to hazard database successfully

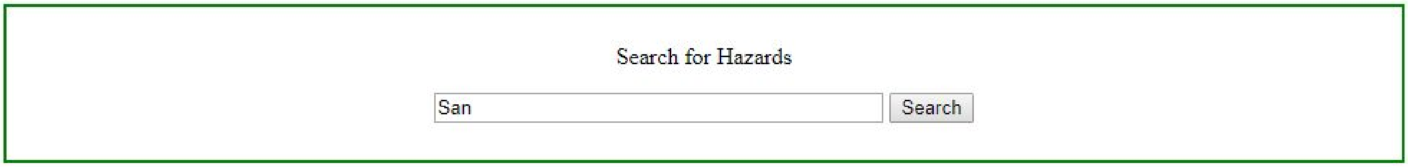
Search for Hazards

Search results



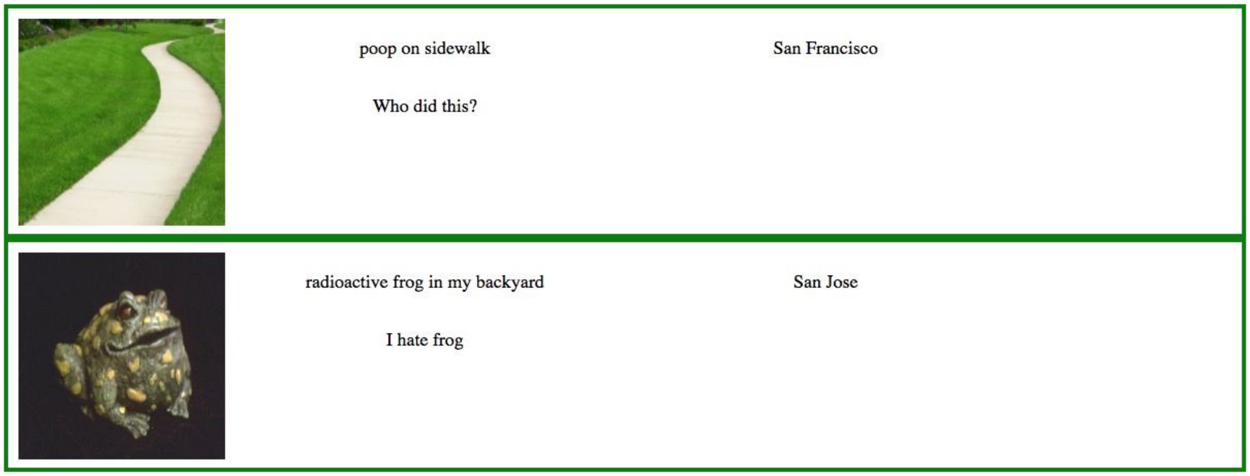
	poop on sidewalk Who did this?	San Francisco
	Possible trash monster... We need ecoHero	Oakland
	radioactive frog in my backyard I hate frog	San Jose
	oil spill Someone selling oil?	Richmond
	Forest Fire! Run!	Berkeley

Searching ‘San’ in search bar



Search for Hazards

Results for searching ‘San’



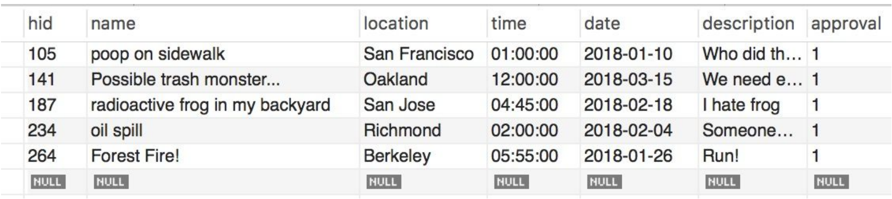
poop on sidewalk San Francisco

Who did this?

radioactive frog in my backyard San Jose

I hate frog

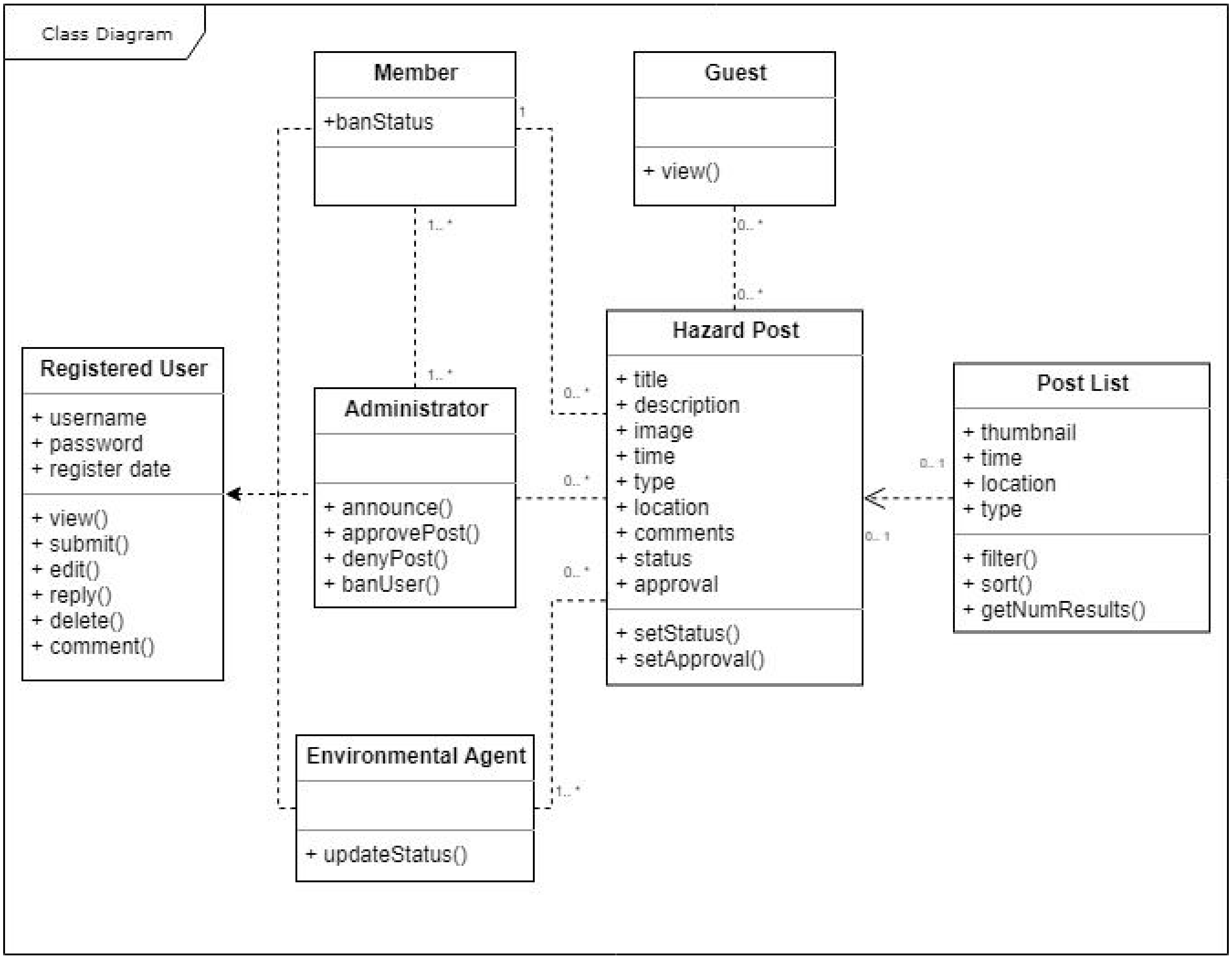
Database



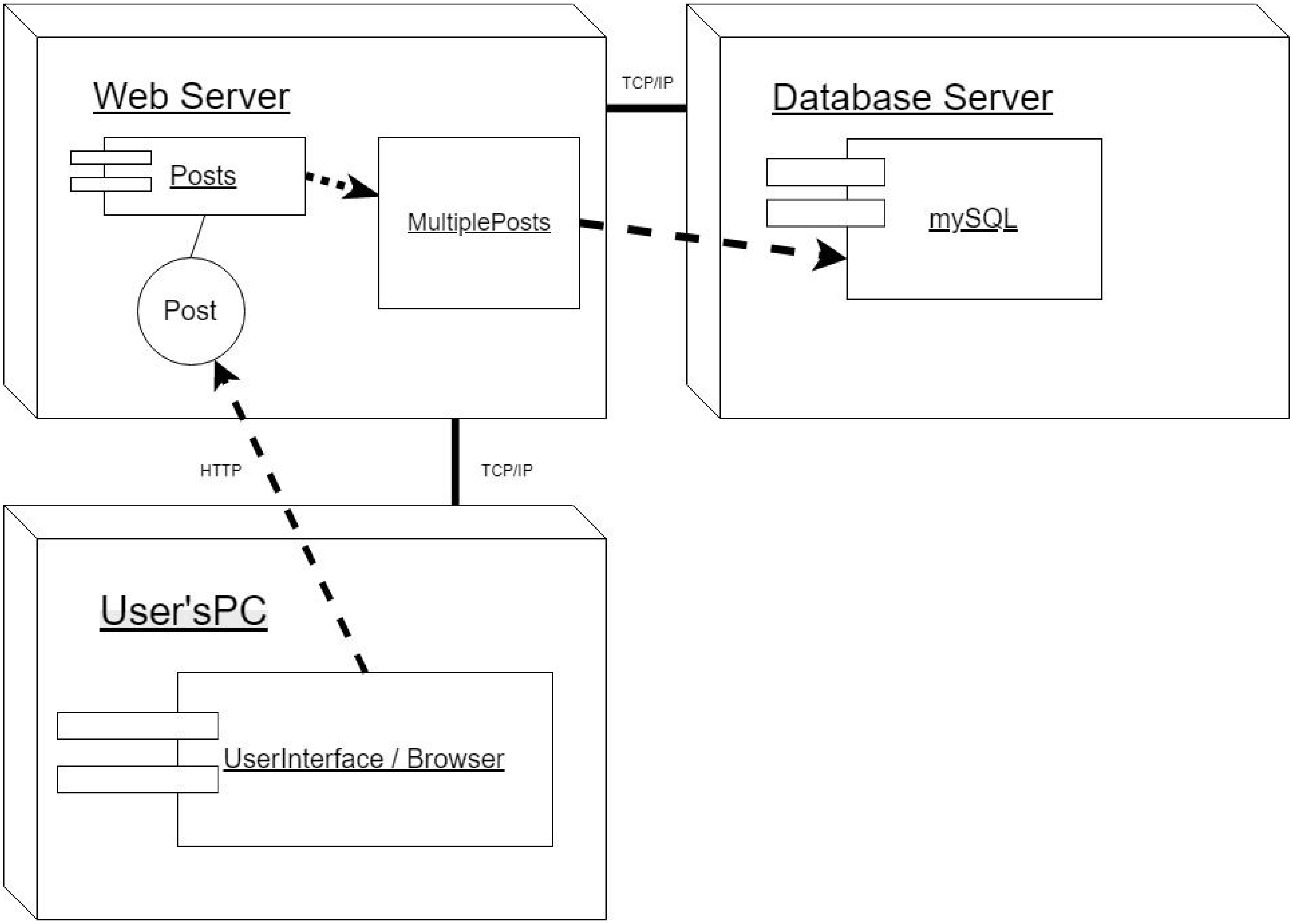
hid	name	location	time	date	description	approval
105	poop on sidewalk	San Francisco	01:00:00	2018-01-10	Who did th…	1
141	Possible trash monster...	Oakland	12:00:00	2018-03-15	We need e…	1
187	radioactive frog in my backyard	San Jose	04:45:00	2018-02-18	I hate frog	1
234	oil spill	Richmond	02:00:00	2018-02-04	Someone…	1
264	Forest Fire!	Berkeley	05:55:00	2018-01-26	Run!	1
NULL	NULL	NULL	NULL	NULL	NULL	NULL

6. High Level UML Diagrams

Class Diagram



Component and Deployment Diagram



7. Key Risks

Skill Risk

Problem: Not every member in the team is not familiar or at a decent enough level in the languages we plan to use.

Solution: Every member gathers and shares resources to the team of the language to learn. Code review is also a significant activity for further learning experiences and for everyone member to understand the workings of the website.

Technical Risk

Problem: Most of the team are using Windows OS, but there are about two members who are using Mac OS which may possibly run into problems in the future.

Solution: While we have not ran into any problems yet besides minor UI differences, if any problem does occur, the first step is to research a solution online. If all else fails, we can let a Windows user in our team to handle the problem.

Teamwork Risk

Problem: There is an imbalancement with the amount of members between front-end and back-end. Majority of the members are currently in front-end and only about two members are in back-end

Solution: In the beginning, the team lead has agreed to switch between front-end and back-end to act as an intermediary role to oversee both sides and help whichever is lacking. Recently, one of the Front-end members has also agreed to a similar role.

Schedule Risk

Problem: As a team of seven members, admittedly, there has not been a single meeting where every single member has shown up. This is a possible communication problem that can affect reaching deadlines, project organization, or making sure everyone is on the same page.

Solution: After every meeting, the team lead or whoever held the meeting will create a write-up that summarizes what happened in the meeting and what decisions have been made in it. As a team, we also plan to use “Trello” which will better organize our tasks and give clearer instructions and assignments to individual team members.