

ecoHero

Milestone 4

May 17, 2018

Andrew Wong

Connor Aguilera

Marcelo Coronado

Preston Segura

Emilio Quiambao

Patrick Shi

Sahas Arora

History Table

2/26/18 Checked by Petkovic	2/28/18 Submitted final draft	5/17/18 Submitted Milestone 4 final draft		
--	--	---	--	--

Table of Contents

1. Product Summary.....	2
2. Usability Test Plan.....	4
3. QA Test Plan.....	6
4. Code Review.....	9
5. Self-Check: Best Security Practices.....	12
6. Self-Check: Adherence to Original Non-functional Specs....	14

1. Product Summary

Product Name:

ecoHero

All Major Committed Functions:

Unregistered users can...

1. Navigate and view approved environmental hazard posts
2. Sort or filter the posts by date
3. Search for posts by title, description, or location
4. Begin creating a hazard post before being asked to sign up before submission
5. Open hazard location on Google maps
6. Log in or create a registered account

Registered users can...

7. Do all the above
8. Post environmental hazards with a title, description, location, date, and image

Must be approved by an administrator

Administrators can...

9. Do all of the above
10. Approve or reject a pending registered user's post

What is Unique About our Project:

N/A

URL of Project:

<http://54.176.158.142/>

2. Usability Test Plan

Overview:

The usability test will be focused on the **search** functionality; we will direct test candidates to perform simple tasks (listed below in Test Plan) on our software.

Test Objectives:

We intend to determine whether or not our UI layout and core mechanics are intuitive to use for an average user.

Test Plan:

Intended User - Average case user (comfortable using computers for basic operation, experience with navigating popular websites).

Test Procedure:

- 1) Test starts with Users being presented the ecoHero main page at
<http://54.176.158.142/>
- 2) Users are directed to search for Hazard listings in “San Francisco” and view them on the search results page.
- 3) Users will then be directed to view any specific Hazard in more detail by viewing its specific hazard post page.

4) Users will conclude with a short questionnaire (shown below)

Questionnaire:

Please answer each question below with a 1-5 rating (1 being “I Strongly Disagree” and 5 being “I Strongly Agree”)

1. Finding the Search bar from the main page was easy and intuitive. _____
2. Viewing a hazard’s specific post information was easy and intuitive. _____
3. Using the software, overall, was an intuitive process. _____

3. QA Test Plan

Objectives:

To determine the efficiency and accuracy of core hazard search functionalities and general usability.

Hardware setup:

- Test administrators laptop/desktop computer, peripherals included (mouse/keyboard)

Software setup:

- Two internet browser windows opened to eCoHero main page (in both Chrome and Firefox)

Tested Feature:

- Search Functionality

Test Cases:

- 1) Correct search results with given input; correct/accurate usage of % like.

Input: "San Francisco"

Expected Output: get 6 results, all having "San Francisco" in location field

- 2) Test category filtering in search field

Input: Select BIOLOGICAL option for filtering

Expected Output:

- 3) Proper page navigation from search page to individual hazard information page

Input: Click on the first Hazard on the search results page

Expected Output: Display that specific Hazard's information page

QA Test Case Results:

Case 1 -

Test #	Test Description	Test Input	Expected Output	Chrome Browser Pass/Fail	FireFox Browser Pass/Fail
1	Test location input in search field	San Francisco	Get 6 results, all have "San Francisco" in location field	Pass	Pass
2	Test category filtering in search field	Select BIOLOGICAL option for filtering	Get 5 results, all have "BIOLOGICAL" in detailed hazard category field	Pass	Pass
3	Proper page navigation from search page to individual hazard information page	Click on the first Hazard on the search results page	Display that specific Hazard's information page	Pass	Pass

Case 2 -

Test #	Test Description	Test Input	Expected Output	Chrome Browser Pass/Fail	FireFox Browser Pass/Fail
1	Test location input in search field	San Francisco	Get 6 results, all have "San Francisco" in location field	Pass	Pass
2	Test category filtering for in search field	Select BIOLOGICAL option for filtering	Get 5 results, all have "BIOLOGICAL" in detailed hazard category field	Pass	Pass
3	Proper page navigation from search page to individual hazard information page	Click on the first Hazard on the search results page	Display that specific Hazard's information page	Pass	Pass

Case 3 -

Test #	Test Description	Test Input	Expected Output	Chrome Browser Pass/Fail	FireFox Browser Pass/Fail
1	Test location input in search field	San Francisco	Get 6 results, all have "San Francisco" in location field	Pass	Pass
2	Test category filtering in search field	Select BIOLOGICAL option for filtering	Get 5 results, all have "BIOLOGICAL" in detailed hazard category field	Pass	Pass
3	Proper page navigation from search page to individual hazard information page	Click on the first Hazard on the search results page	Display that specific Hazard's information page	Pass	Pass

4. Code Review

Coding Style:

- Nothing should ever start with a capital letter besides Model classes
- Keep everything lowercase
- For database: use underscores when needed
 - (ex. `this_is_an_example`)
- For website development: use lower camel case when needed
 - (ex. `thisIsAnExample`)
- If you created and coded a new file, add your name to the header comment.
- If you edited a majority of a file, add your name on to the header comment.
- If you added a view, write a brief comment block above it describing what it does

Peer Review

[ecoHero] Review on Hazard Search

2 messages

Emilio Quiambao <emilioquiambao@gmail.com>

Thu, May 16, 2018 at 10:27 PM

To: Andrew Wong <ayounghowong@gmail.com>

Hey Andrew,

I updated the search results view.

It now checks if the post ID is approved, so now only approved posts get shown in the search results.

Take a look and let me know if it looks good.

Thanks,

Emilio

Andrew Wong <ayounghowong@gmail.com>

Thu, May 16, 2018 at 11:35 PM

To: Emilio Quiambao <emilioquiambao@gmail.com>

Code looks good, but remember to add what you changed into the block comment (i.e. checking if the post id is approved).

You also forgot to add the 'sort' part in the comment block.

Other than that, nice job!

#Hazard Search Results

#Gets user search input and matches with Posts' title, location, and description in database

#Also matches category based on dropdown option

def searchresults2(request):

if request.method == 'POST':

search_input = request.POST.get('search')

category = request.POST.get('category')

sort = request.POST.get('sort')

else:

search_input = ""

category = 'none'

sort = "newest"

approvedIDs = PostApproval.objects.filter(approval=1).values_list('post', flat=True)

if category == 'Chemical':

posts = Post.objects.filter(Q(post_id__in = approvedIDs)

& (Q(type__icontains = 'CHEMICAL')

& (Q(title__icontains = search_input)|Q(location__icontains =

search_input)|Q(description__icontains = search_input))))

elif category == 'Biological':

posts = Post.objects.filter(Q(post_id__in = approvedIDs)

& (Q(type__icontains = 'BIOLOGICAL')

& (Q(title__icontains = search_input)|Q(location__icontains =

search_input)|Q(description__icontains = search_input))))

elif category == 'Physical':

posts = Post.objects.filter(Q(post_id__in = approvedIDs)

& (Q(type__icontains = 'PHYSICAL')

& (Q(title__icontains = search_input)|Q(location__icontains =

search_input)|Q(description__icontains = search_input))))

else:

posts = Post.objects.filter(Q(post_id__in = approvedIDs)

& (Q(title__icontains = search_input)|Q(location__icontains =

search_input)|Q(description__icontains = search_input))))

if sort == "oldest":

posts = posts.order_by('post_id')

else:

posts = posts.order_by('-post_id')

return render(request, 'searchresults2.html', context={'posts': posts, 'input': search_input,
'category': category, 'sort': sort})

5. Self-Check: Best Security Practices

Major assets protected:

- Passwords of users
- Access to admin only pages
- Access to unapproved posts
- Access to guest only pages (logging in, registering)
- Access to submitting a hazard

Passwords encrypted in the Database:

Below, you can see the fourth column where passwords are stored are all hashed

28	pretesttest3	2018-04-23	pbkdf2 sha256\$100000\$uuGcOt6mBVJ\$ft1vo409PO35C990cb4a8RX0btuJAGVDOKS4rLetSws=	NULL	2018-04-23 18:27:07
23	pwtest	2018-04-22	pbkdf2 sha256\$100000\$RIMCKChm1ooo\$A8LBGwEOuTtG05WABca6ftHvonwRt3/vcoX3eWxnIuE=	NULL	2018-04-23 04:51:23
30	asdfahkl	2018-05-10	pbkdf2 sha256\$100000\$nohECB9nIiO6\$0CIG+OGnu7KTNMHkWRO7+JA9zXIDCO+1hU1fMiaVEY=	NULL	2018-05-11 00:36:31
24	isthispasswor...	2018-04-22	pbkdf2 sha256\$100000\$43Z0sc18bmE\$Yv9tPo4Xab7ZY5McuoLA/kam2DBaaFGistFzab/9srE=	NULL	2018-04-23 04:45:20
29	patrickshi	2018-04-23	pbkdf2 sha256\$100000\$iiT8P9hO381M\$O+MLahJ+dGKOFX68iNxxXE+bENk0suMo0rvzL/B3mic=	NULL	2018-04-24 00:15:17
31	sahas.arora	2018-05-11	pbkdf2 sha256\$100000\$08ai01l35vMY\$KM7hsIR0hfrxO1xzPXoMnU6azzomObUaJM28UF5voo=	NULL	NULL
27	pretesttest	2018-04-23	pbkdf2 sha256\$100000\$6e9UOBVndhsc\$9o3cCOI38XRKdan91kHPxZCldX7UVRJD9OafO41lGO=	NULL	2018-04-23 08:09:23
25	emilooo	2018-04-22	pbkdf2 sha256\$100000\$3odh85EwhGa0\$8HMAvRmr1K+uCV3wdacufO3sXootM33x5FCZ29LSOC=	NULL	2018-05-11 23:42:56

Input Data Validation:

All Forms

- Cross Site Request Forgery protection
`<form method="post">{% csrf_token %}`

Registering / Logging In

- Username must be at least 4 characters
`if len(reg_username) <= 3`
- Username must be unique
`if RegisteredUser.objects.filter(username = reg_username).exists():`
- Password must be at least 5 characters
`if len(reg_password) <= 4`

Submitting a Hazard

- Title must be less than 45 characters
`title = models.CharField(max_length=45)`
- Description must be less than 200 characters
`description = models.CharField(max_length=200, blank=True, null=True)`
- Location must be less than 100 characters
`location = models.CharField(max_length=100)`
- Title is required
HTML/CSS
`<input required type="text" name="title" class="form-control" id="title" value="{{title}}"/>`
Python
`if(title == "")`
- Location is required
HTML/CSS
`<input required type="text" name="location" class="form-control" id="location" value="{{location}}"/>`
Python
`if(location == "")`
- Upload file must be an image file or will be converted to '.jpg'
`if (len(request.FILES) != 0): image = request.FILES['image']`

Searching a Hazard

- Search input must be 50 characters or less
`<input class=" form-control col-lg-8" type="text" name="search" size="40" maxlength="50" value="{{input}}" style="max-width: 20%"/>`

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).

DONE

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.

DONE

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

DONE

4. Data shall be stored in the team's chosen database technology on the team's deployment server.

DONE

5. Application shall be media rich (at minimum contain images and maps)

DONE

6. No more than 50 concurrent users shall be accessing the application at any time

DONE

7. Privacy of users shall be protected and all privacy policies will be appropriately communicated to the users.

DONE

8. The language used shall be English.

DONE

9. Application shall be very easy to use and intuitive.

DONE

10. Google analytics shall be added

DONE

11. No e-mail clients shall be allowed

DONE

12. Pay functionality, if any (e.g. paying for goods and services) shall not be implemented nor simulated.

DONE

13. Site security: basic best practices shall be applied (as covered in the class)

DONE

14. Modern SE processes and practices shall be used as specified in the class, including collaborative and continuous SW development

DONE

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).

ON TRACK