SW Engineering CSC648/848 Spring 2023

CrisisConnect

Team Details:

Class Section: 2
Team: 4

Contributors:

Team Lead: Mo Moses (mmoses3@mail.sfsu.edu)
Frontend Lead: Marc Castro
Backend Lead: Axel Biehler
FrontEnd Dev: Nicholas Chan
FrontEnd Dev: Virel Patel
BackEnd Dev: Raquel Lutges
BackEnd Dev: Lokesh Telaprolu

Milestone 4: April 25th, 2023

Revisions	
4/25/2023	Initial Submission
5/16/2023	Revision - Product Summary

Product Summary

Product Name: CrisisConnect
Functionality:

- Our product provides a dashboard giving users access to real-time reported data about critical events in their area
- We give our users access to information and resources for use in an emergency.
- We provide a way for people to search for critical events pertaining to Fire, Weather, Security, and Health in the state of California and more specifically in their counties by selecting or telling us which county is relevant to them.
- Not all events are relative to all people so we make it possible for users to filter to events based on the kind they are interested in (the results of the filtration are displayed on the Map). Now you can hone in on what matters to you.
- To make it easier to gauge the impact of the event on our users, we provide a display of events/incidents on a live Map showing an icon which represents the event. So users can see clearly where the event is and what it is at a glance. No more guessing if that county is near you or not.
- To keep track of what is and has happened, we provides our users with access to a live feed showing current and past incidents.
- For users that want greater engagement, we've added the ability to register with the application service either as a citizen or even a county official.
- If our users register as citizens, one of the added features they will be able to participate in is an alerting system where in they will receive alerts submitted by county administrators about emergent events.
- If county officials register they receive a privileged status and are able to input and update data about critical events happening in their county to keep their citizens informed.
- Registered county officials are also recognized as admins and can trigger alerts to users that are registered to receive them.

Website address: https://www.abiehler.com

Usability Test Plan

-**Test objectives**:

For our Usability Test, we will be testing our search function. This is one of the most important functions of our web applications. Using the search function, our users will be able to see any events in their selected area that might be relevant to the population's safety. Additionally, our search function allows our users to filter the results to more accurately provide them the information that they are looking for. It is important that we prioritize accuracy above all in our search function. If the user is provided with results that do not accurately show them the results they want, then the application will lose its value in the eyes of the users. Additionally, it is also important that our search function is convenient to the user. The search function should be very intuitive and require very little if any prior knowledge in order to use it. Our search

function should also be very fast and provide results immediately after a very short and simple series of steps by the user.

Test Background:

- System Setup: The system setup for our web application is extremely simple. All the user has to do is connect to the Internet on their computer and open up a web browser to go to our URL. Our web application is also supported on most browsers. The search function is already on our homepage, so there is no additional routing required.
- Starting Point: The starting point is at our homepage which will show up automatically once the user has visited our website through the URL.
- Intended Users: The intended user for our Search function is essentially anybody who has access to the Internet and our website. There is no registration or login required to use our search function, making it accessible to Internet users of all types.
- URL of the system to be tested: www.abiehler.com
- What is to be measured: For this usability test, we are measuring user satisfaction with using our search function to find results.

Usability Task Description:

1. Search for events in the San Francisco County
2. Filter results to only Weather events
3. Filter Results to both Wildfire and Security Events
4. Keeping the filter, search for events in San Mateo County

-To measure the effectiveness of our search function, we will ask users to perform specific search tasks, such as finding events related to a specific category or within a certain county. We will measure the success rate and the time it takes to complete each task to determine the effectiveness of our search function.

-To measure efficiency, we will track the number of clicks it takes users to find the desired results and the time it takes to complete each task. This will allow us to evaluate how quickly users can find what they are looking for using our search function.

-Likert Subjective Test:

Statement	Strongly Disagree	Disagree	Neutral	Agree	Strongly Agree
Overall, I found the search function easy to use.					
The search results were accurate and relevant to my needs.					
I was able to quickly find what I was looking for using the search function.					
Comments:					

QA Test Plan

Test Objectives:

1. Verify that the search function is able to display **all** events in the user's selected area that are relevant to the population's safety.

2. Verify that the search function is able to filter the events/alerts accurately based on user-selected filters such as location, or category.
3. Verify that the search function is able to filter **only the selected events** in the user's selected area.
4. Verify that the search function provides appropriate feedback to the user if there are no alerts or if there is an error in the search query.
5. Verify that the search function prioritizes accuracy above all else and does not return any irrelevant results. (i.e. like displaying another event when one event is triggered)
6. Verify that the search function is reliable and consistent in its performance across different devices and browsers.
7. Verify that the search function is fast and provides results immediately after the user inputs their search criteria.

Hardware Setup:

- The hardware requirements for the testing will be a computer or mobile device that meets the minimum system requirements to run the application.

Software Setup:

- The testing will begin once the software code is pushed to the testing environment. The search function should be fully functional and integrated into the below web application.
    - URL of the system to be tested: www.abiehler.com

Feature to be tested: Search function

Test Plan:

No:	Title	Description	Test Input	Expected output	Results

1	Testing the application whether it could filter all events.	Verify that the search function is able to display all events in the user's selected area that are relevant to the population's safety. Verify that the search function is able to filter the events/alerts accurately based on user-selected filters such as location, or category.	1. Trigger two events (Health and Weather for a specific county, say it: Alameda) 2. Navigate to the web application from this URL. 3. Choose a county that was selected in Step 1 and filter for **all** events.	The application should filter out and display the events on the map against the selected county.	Pass/ Fail
2	Testing the application whether it could filter only selected events.	Verify that the search function is able to filter **only the selected events** in the user's selected area.	1. Trigger three events (Health, Security, and Weather for a specific county, say it: Mono) 2. Navigate to the web application from this URL. 3. Choose a county that was selected in Step 1 and filter for **only specific events.** (Health and Security)	1. The application should display only the specific events (which were filtered out) on the map against the selected county. 2. It should not display the events for counties that are not filtered.	Pass/ Fail
3	Testing how the application works if no events are triggered.	Verify that the search function provides appropriate feedback to the user if there are no alerts or if there is an error in the search query.	1. Do not trigger any events/alerts for one county. 2. Navigate to the web application from this URL. 3. Filter out the events for the county selected in Step 1	The application shouldn't display any events since we didn't schedule any events.	Pass/ Fail
4	Performance across different devices and browsers.	Verify that the search function is reliable and consistent in its performance across different devices and browsers.	1. Test the above cases in Chrome, Firefox, or Edge browsers. 2. Test the same cases on mobile	The Application should be reliable and consistent across all browsers and devices.	Pass/ Fail

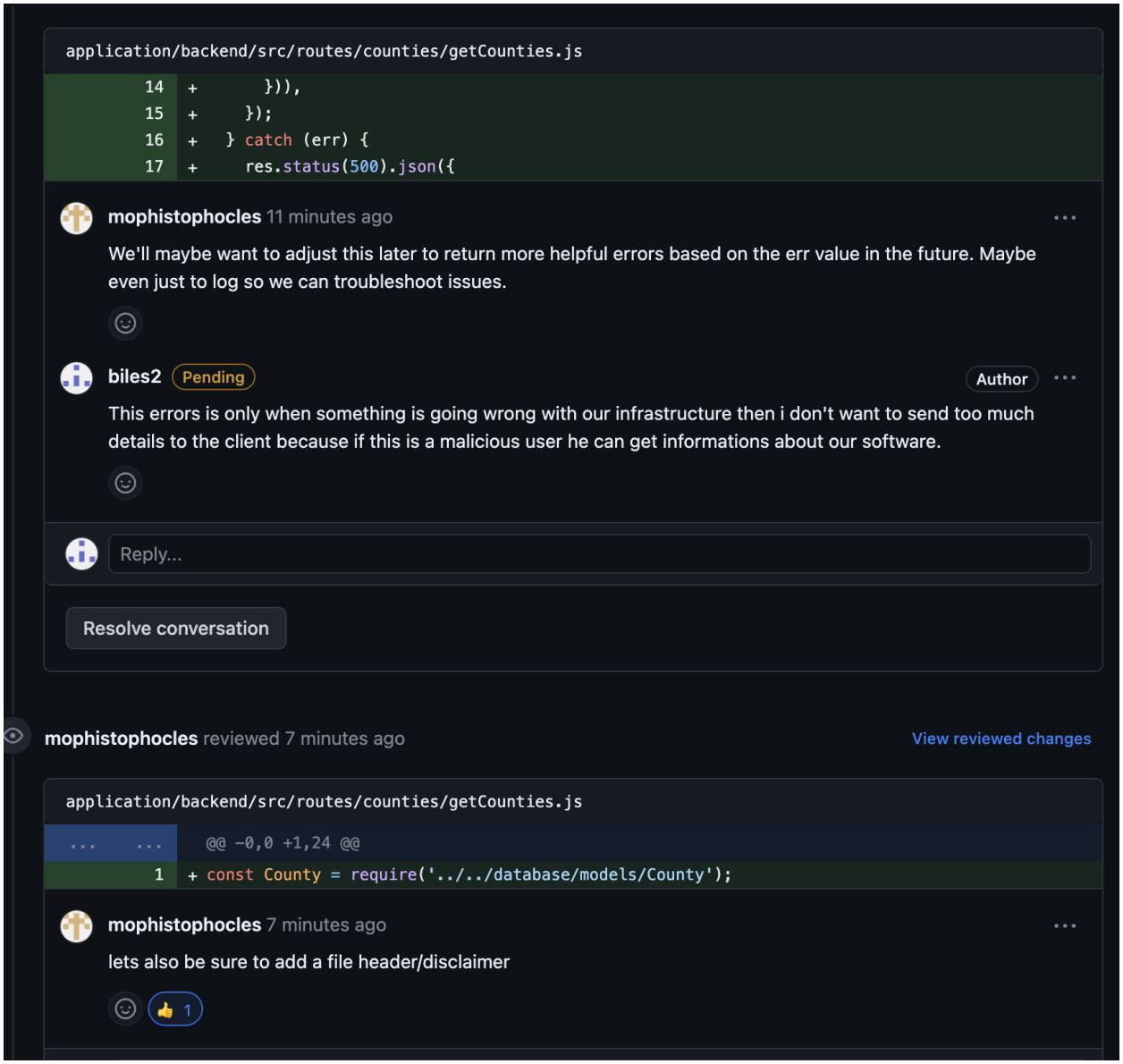
			phones(use chrome for Android and Safari for iOS)		

Code Review

Code Style: Coding in javascript allows a lot of flexibility in style choices as the language is not very strict. To make this more controlled we implemented ESLint in our repository and followed the basic guidelines it provided.

- Comments were not heavily enforced and descriptive functions and variables preferred
- Camel case for methods and variables.
- Single line comments as opposed to multiline comments preferred (where used)
- Spacing instead of tab for cross IDE compatibility

Code Review Example:



Self Check: Security

Protected Assets:

➔ Email

➔ Passwords using hash

➔ Input Sanitation to check for invalid characters

- User authentication via email address and password on sign up.
- User roles: Authorization is implemented using role-based access that controls privileges giving only authorized accounts access to certain functionalities like inputting county data.
- We confirm that using bcrypt makes sure passwords are encrypted and stored passwords securely in the database.
- Input validation is predefined for counties in the drop down menu.
- Search input data validation for a string containing 1 to 40 alphanumeric characters

```
if  !searchInput match /^[a-zA-Z0-9 ]{1,40}$/
```

Self check: Non-Functional specs

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	Done

chosen by the student team but all tools and servers have to be approved by class CTO).	
Application shall be optimized for standard desktop/laptop browsers e.g., must render correctly on the two latest versions of two major browsers	Done
Selected application functions must render well on mobile devices (this is a plus)	OnTrack
Data shall be stored in the team's chosen database technology on the team's deployment server.	Done
Privacy of users shall be protected, and all privacy policies will be appropriately communicated to the users.	OnTrack
The language used shall be English.	Done
Application shall be very easy to use and intuitive.	Done
Google maps and analytics shall be added	Done
No e-mail clients shall be allowed. You shall use webmail.	OnTrack
Pay functionality, if any (e.g. paying for goods and services) shall not be implemented nor simulated in UI.	Done
Site security: basic best practices shall be applied (as covered in the class)	Done
Modern SE processes and practices shall be used as specified in the class, including collaborative and continuous SW development	OnTrack
The website shall prominently display the following exact text on all pages "SFSU Software Engineering Project CSC 648-848, Spring 2023. For Demonstration Only" at the top of the WWW page. (Important so not to confuse this with a real application).	OnTrack