I.C.E.D.
Interface for Coordinating Emergencies and Disasters

September 17, 2018

1 Team

Daniel Campos (dcampos2015@my.fit.edu)
George Nelson (gnelson2015@my.fit.edu)
Thomas van Haastrecht (tvanhaastrec2015@my.fit.edu)
Kevin Crowley (kcrowley2015@my.fit.edu)

Advisor: Dr. Crawford (hcrawford@fit.edu)

2 Goals & Motivation

As the occurrence of natural disasters continues to cause catastrophic damage to metropolitan and rural areas alike [1] [2], the need for new innovative disaster relief solutions is readily apparent. After researching currently available solutions for disaster relief management, we have found that there is not a unified platform for the public to use during a disaster situation to centralize related information. Likewise, the public does not have access to a platform or service that allows users to check on the status of local resources, like open shelters or available gas. Possessing such a platform would not only enable more effective relief efforts from aid organizations like FEMA, but also would allow for the implementation of other improvements that have been critically identified [3]. The gap between the systems that currently exist and what new needs have been identified can be bridged by building an interface that would coordinate civilians, first responders, and relief organizations throughout the entire disaster relief process.

The Interface for Coordinating Emergencies and Disasters (I.C.E.D.) aims to be an innovative solution that addresses three major areas of concern that have not been previously answered by another platform or service. These critical areas, based on client interviews with relief organizations are as follows:

- Creating an interface for managing and filtering incoming data for relief organizations. (traffic, downed utilities, road hazards, etc)
- Providing a user-friendly platform for distributing information to civilians in an affected area
- Assisting local organizations in communicating their logistical operations to parent organizations, such as FEMA

3 Client Perspectives

We interviewed two clients late July 2018, Russ V. who worked at Brevard EOC (Emergency Operations Center) and James Kriegh from PNS (Probability and Statistics). Early August we pulled a third client perspective from a FEMA post-disaster report following the 2017 hurricane season.

[1] Hurriance Harvey
[2] State of California Wildfires
[3] FEMA Key Findings

The Brevard EOC employee gave us an internal perspective on the things volunteers struggle with inside the organization. Probability and Statistics gave us a business and analytics overview as well as the initial blockchain proposal. They also helped narrow down which organizations and individuals would be interested in the collected data. FEMA's post-disaster report shed light on what they perceived as weaknesses in their plans to deal with the aftermath of hurricanes Irma and Maria. This report ultimately gave us the inspiration to include crowd sourced data and assist in gathering data for organizations' logistics.

3.1 Brevard Emergency Operations Center (EOC)

Brevard EOC is a subsidiary organization that provides Homeland security services to Brevard County, Florida. EOC's disaster relief operation facilitates the coordination of local fire and police departments to aid with servicing the community in response to flooding, sinkholes, hurricanes, etc. In response to asking Russ V. about the key features of our project his main concerns are quoted below:

- “A single platform for affiliates would be useful only if the governing organization is demarcated from the user base.”
  - Based on Russ' comment regarding having a separation between civilian and organizational use, the need for civilians to use a readily available application that interfaces with organizational dashboard came to fruition.
- “Tracking admittance to shelters would benefit the logistics of organizations like FEMA, but power loss is a prevalent issue with electronic systems.”
  - The feature that divulged from this perspective <is> the active information distribution/coordination between subsidary groups to a primary disaster relief organization as well as tracking logistical information for these clients.
- "The real time data monitoring would allow organizations such as Brevard EOC to respond more quickly and effectively to critical areas/victims."
  - The real time data monitoring suggestion allows for faster response times, and could end up saving lives.

3.2 Probability and Statistics

Late July the project was proposed to the software development manager, James Kriegh, currently employed at Probability and Statistics (PNS). PNS is a company specializing in computation and blockchain solutions, which helped provide a business perspective and potential difficulties for the project goal. Most notably, Mr. Kriegh's concerns were as follows:

- The amount of incoming data will be enormous, and some method of pre-processing or filtering will be needed to make the incoming data easier to use for regular individuals.
- Providing some form of analytics processing on public data will be valuable to organizations like FEMA. This data often includes expenses, resources used, affected individuals, and damages. Organizations like FEMA use this data extensively when compiling post-disaster reports..
- Providing an interface for volunteer/assisting organizations like hotels, hospitals, etc. to record and view information in a standardized way would simplify a lot of operations.

3.3 FEMA Post-Disaster Report

FEMA provided a post-disaster report[4] indicating their perceived strengths and weaknesses dealing with logistics, recovery efforts, and response times for the 2017 hurricane season. From this report, they summarized and explained some key findings:

[4] FEMA Disaster Report

3.3.1 Key Finding #3

"FEMA could have better leveraged open source information and preparedness data, such as capability assessments and exercise findings, for Puerto Rico and the U.S. Virgin Islands."
Through these findings we came to the realization that larger organizations may also not have easy access to open source or crowd sourced resources, let alone trustworthy sources.

3.3.2 Key Finding #9

"FEMA...experienced challenges in comprehensively tracking resources moving across multiple modes of transportation to Puerto Rico and the U.S. Virgin Islands due to staffing shortages and business process shortfalls."
In areas where operations were not conducted with the same zeal as those done in the continental United States, performance suffered. There is potential in providing a logistics platform where users with limited experience can easily find and record information to expedite the recovery process.

3.3.3 Key Finding #16

"...FEMA helped survivors quickly transition from congregate shelters to other options such as hotels...FEMA faced challenges implementing non-congregate sheltering programs."
Adding onto the realization through Key Finding #9, we further supported the argument for a logistics platform that could help standardize and assist in coordinating between relief organizations and volunteer organizations.

4 Key Features

4.1 Information Collection from Anonymous Sources

Disaster relief organizations have historically struggled with determining how to effectively administer aid to all parties affected by a disaster. With attempting to process large influxes of aid requests, relief organizations are consistently constrained to a reactionary mindset rather than a proactive one. A lack of accurate data makes it difficult to provide aid, while prioritizing more severe situations. The I.C.E.D. framework will allow users to submit their aid request to a centralized system that will enable relief organizations to analyze and determine the best course of action based on analyzed real-time data. Proposed functionality:

- Status Update
  - Allows users to check in and update their current health and safety information such as: Need medical attention immediately, need water soon, low on food, etc.
- Event Reporting
  - Allows users to report events such as power loss, water loss, wind damage, storm surge, etc.
- Relief Locator
  - Allows users to search within a radius for nearby food, water, and gas distribution centers as well as nearby evacuation shelters

4.2 Information Distribution and Analytics

A large concern with emergency relief is the distribution of verifiable information for civilians. This information is often used and not limited to:

- Provide evacuation route / shelter locations
- Track storms / affected areas
- List hazardous areas in the case of sinkholes, lava flows, downed power lines, etc.
- List hospital locations and/or availability of resources

Unfortunately, disaster operations have to operate independently from common utilities like Internet access or mobile data since all levels of society will be affected. In these situations, information is easily lost, faked, or not distributed in a common way. Most users are forced to listen to radio broadcasts, twitter feeds, and facebook posts, sometimes all at once in order to grasp a clear picture of the situation and how to proceed.
To rectify this, we acknowledge three primary objectives:

4.2.1 Information must be accessible in both a centralized and decentralized way

Individuals who are prepared for an impending disaster will most likely prepare materials like maps beforehand. Information like open hospitals, camp locations and capacity, and shelter locations may be marked initially, but can be difficult to update without a reliable source of information. If this information is digital it is even harder to distribute and verify. Thus, we must ensure a method exists that allows users to initially download data from a common source, then allow them to share and retrieve newer information from alternative sources that include but are not limited to:

- Other user's mobile devices
- Shelter networks
- Camp networks

By allowing users to distribute data, we also reduce the load on site servers and ensure densely populated areas have a local archive of information, regardless of whether or not that area has an active internet connection. This objective comes with security concerns, which are addressed in our next objective.

4.2.2 Information must be verifiable, both in the order it is received and its contents

If information can be verified we prevent the manipulation and spreading of false information. By doing so, we can ensure clients always have an offline, provably authentic log of information that can assist in their decision making processes. If the order that information is received is also verified, we can ensure that no information is withheld by malicious clients. An example is when a malicious user transmits old information giving the location of a shelter that was later updated listing it at maximum capacity. Without a method to verify the sequence of received events, a malicious client can pick and choose what records are transmitted leading to scattered and most likely incorrect information. If both of these concerns are addressed, then users can easily and safely distribute updated emergency information.

4.2.3 Information must be permanent

To address one of our clients' concerns for analytics, we need some way of ensuring that information that is distributed, whether privately or publicly is permanent. By doing so, we can keep a log of everything that was distributed between organization and user, organization and organization, and a collection of post-disaster reports. Including this information will make it easier for both parties planning to deploy to anticipated areas or evacuate as they will have years of reports to build off of.

4.3 Logistics

To efficiently allocate resources, an organization must be able to actively track the resources available to it. Keeping track of supplies, such as water, food, and gas, and which areas have already been allocated resources, allows for a clear view of the situation, which allows for proper planning.

5 Novel Features

Our biggest novel feature will lie in how data is distributed between servers and clients, most notably in our blockchain feature. Most notification services rely on a centralized server or collection of servers to distribute information, making them dependent on an active Internet connection.

Utilizing a technology like blockchain will allow us to freely distribute information without a direct connection to the servers. This allows people to intermittently share and distribute updated copies of information with no added complexity.

6 Technical Challenges

The creation of a project results in the emergence of technical challenges that may negatively affect the success of the project if incorrectly address and mitigated. The challenges we have identified are as follows:

6.1 Big Data Volume and Processing

Currently, we are expecting enormous amounts of data between organizations tracking resources and crowd source data. This includes things like:

- Offline utilities
- Downed power lines
- Broken water mains
- Inoperable Roads
- Hospital inventory

With this much data, including potential custom inputs, we face issues with storing and searching through the data efficiently. For long-term use, a method to archive the data and move it out of the database into a blockchain would assist in keeping the main database smaller. Such a method can be implemented by exporting data into a blockchain once 6 . However, it does not change the fact that a large amount of processing will need to be done, or a more efficient method will need to be found.

6.2 Limited Front-End Development Experience

Currently only two members have front-end experience from previous positions spanning a month or more. The remaining two members have little to no experience and will have to spend time learning and experimenting with the various front-end languages needed to create a functional interface. Since the majority of this project focuses on interfaces between users and data, this will post a significant challenge.

6.3 Limited Team Based Blockchain Experience

While not as significant as front-end experience, only one member has significant blockchain experience. This will severely limit the team's ability to work out the communication and data distribution features, until all members can be trained on the technology.

6.4 API Security

The front-end features will run all functions through a JSON API located on another server. This allows us to keep back-end code and front-end code in different places so when it comes time to scale and deploy, it will be simple to spin up / down the number of servers to meet different demands (static content versus data processing). Unfortunately, this means everything will be run through a common API, so a significant amount of work must go into securing all endpoints. This includes:

- Ensuring applications can recover from garbled data input
- As many errors are caught and handled appropriately
- All database queries are sanitized appropriately
- Any expensive calls, like those that use extensive compute time are rate-limited or paginated

- Some form of API hit rates are recorded to detect repeated function calls with the sole intent of wasting server resources.

Currently one member has experience in testing and securing endpoints effectively, which requires more training for the other team members.

6.5 Implementing cross-platform compatibility

While not as large of an issue as the previously mentioned challenges, implementing cross platform compatibility will still be a technical challenge. Several equally functional methods are named differently across several browsers which can cause pages to appear incorrectly depending on what browser / OS combination is being used. Some features, especially with the rollout of HTML5, may not work correctly on older browsers. This will require significant amounts of research to determine what technologies we can use to support the majority of the population, as well as to help decide where we draw the line for minimum requirements.

7 Milestone 1 (Oct 1)

- Create Requirement Document
- Create Design Document
- Create Test Plan
- Evaluate Technical Tools and Collaboration Tools
- Resolve Technical Challenges

Tools		
Category	Preferred Tools	Alternative Tools
Blockchain	Geth	Parity
Database	MariaDB	Microsoft SQL
Back-end Language	PHP	ASP.NET
Front-End Scripting	Javascript	Coffescript/Typescript
Front-End Markup	HTML	XHTML
Front-End Styling	CSS	-
Mobile Language	Java/C#	Objective C
OS	Linux	Microsoft Windows
Smart Contract Language	Solidity	Julia

Collaboration Tools		
Category	Preferred Tools	Alternative Tools
Source Control	Gitlab	Subverison
Instant Messaging	Discord	Telegram
LaTeX	Overleaf	Gummi
Shared Storage	Google Drive	SFTP

7.1 Task Matrix for Milestone 1

Task	Thomas	Daniel	George	Kevin
Create Requirement Document	20%	40%	20%	20%
Create Design Document	40%	20%	20%	20%
Create Test Plan	20%	20%	40%	20%
Evaluate Technical & Collaboration Tools	20%	20%	20%	40%
Resolve Technical Challenges	Front-end Development	Database optimize, security	Blockchain, Front-end	Front-end Development

8 Milestone 2 (Oct 29)

- Begin Implementation of back-end functions for crowd sourced data collection
- Begin Implementation of back-end functions for information distribution
- Begin Implementation of the front-end
- Begin Implementation of the initial database
- Begin the Implement small maintenance functions for analytics
- Architect archiving chain functionality

9 Milestone 3 (Nov 26)

- Implement basic back-end to front-end communication
- Implement functional prototype of back-end functions for crowd sourced data collection
- Implement functional prototype of the information distribution feature
- Implement functional prototype of the front-end
- Implement functional database

10 Faculty Sponsor Approval

- "I have discussed with the team and approve this project plan. I will evaluate the progress and assign a grade for each of the three milestones."
- Signature: ______________________________ Date: __________