Emergency Resource Allocation (E.R.A.) Website

By

Jose Martinez

Final Report for CSCI 4390

Computer Science Department

The University of Texas Rio Grande Valley

2018

Abstract

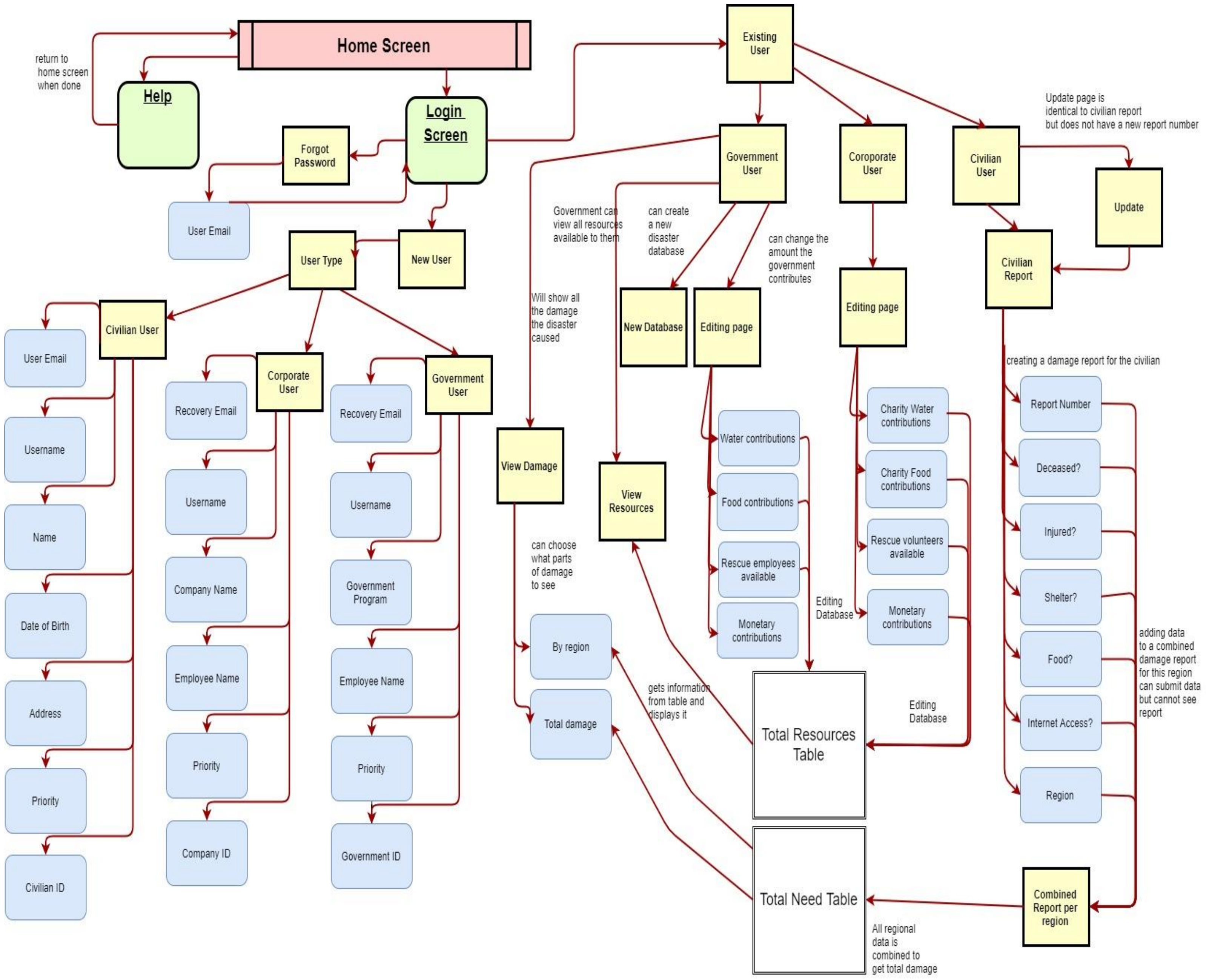
The overall goal of this project was to create a functioning webpage that would help gather and organize resource data in the event of a natural disaster. The website consists of a database of users, natural disasters, and the resources required by the civilian victims of the disaster. The purpose of gathering all this data is to speed reaction and recovery time of the affected area. By helping provide organized damage information, government organizations like the Federal Emergency Management Agency (FEMA) can become users and send help exactly where it is most needed and prevent having wasteful surpluses in. The government users of this webpage will have different privileges from those of the civilian user. The government user will have to specify which organization they belong to and provide an organization code when signing up with the website. These users will be able to see different "disaster pages" and create new ones. They will also be able to see the total amount of resources needed per disaster and see which areas need which resources. Civilian users will be able to log in and report damage such as fires or flooding in their area. They will be shown available resources and report what they need and how much they need (limits will be set on how much an individual can request). They will also have the option to create a new resource and request that. Their resource will be added to the database and should be available for viewing by other users. Future possibilities for this project may involve the implementation of google maps to improve the location of resource need accuracy. Further implementations can also involve the prediction of which resources will be needed once a disaster has been declared, predictions would use the maps data to determine possible quantities of resources while data from previous disasters can help determine which items will be needed.

Table of Contents

Initial Design

When a natural disaster strikes there is a lot of confusion and chaos getting help to where it is needed can be problematic since no one knows which regions need what resources. Misallocation of resources can lead to loss of life, people may not get the help they need in time. Misallocation of resources can lead to inefficient resource dispersal, one region can have a surplus of a particular resource while another region is desperately lacking it. Initial design of the website was intended to include a corporate user as a third kind of user. The corporate user was supposed to represent charitable corporations that wanted to donate resources to help during a disaster but did not know what to donate. Due to time constraints and unforeseen circumstance this user-type was cut out of the main design plan, being deemed as an extra feature rather than being inherently necessary to the main idea of the project.

The following is a description of the original design to be used for comparison to the end product:



Home Screen
return to home screen when done
Help
Login Screen
Forgot Password
User Email
New User
User Type
Civilian User
User Email
Username
Name
Date of Birth
Address
Priority
Civilian ID
Corporate User
Recovery Email
Username
Company Name
Employee Name
Priority
Company ID
Government User
Recovery Email
Username
Government Program
Employee Name
Priority
Government ID
Existing User
Government User
Government can view all resources available to them
can create a new disaster database
can change the amount the government contributes
Will show all the damage the disaster caused
View Damage
can choose what parts of damage to see
By region
Total damage
View Resources
New Database
Editing page
Water contributions
Food contributions
Rescue employees available
Monetary contributions
Editing Database
gets information from table and displays it
Total Resources Table
Total Need Table
All regional data is combined to get total damage
Coroporate User
Editing page
Charity Water contributions
Charity Food contributions
Rescue volunteers available
Monetary contributions
Editing Database
Civilian User
Update page is identical to civilian report but does not have a new report number
Update
Civilian Report
creating a damage report for the civilian
Report Number
Deceased?
Injured?
Shelter?
Food?
Internet Access?
Region
adding data to a combined damage report for this region can submit data but cannot see report
Combined Report per region

User Authentication

The database will access the users table to and run a query searching for the username and password, if found the screen will move on to the corresponding screen. The next screen will be determined by the value in the “priv” column of the returned row from the query, here priv means privilege, what the specific user has access to. The three possible integer values for this attribute will be 2 for government, 1 for corporation, or 0 for civilian. These three values identify what type of user the person logging in is and what they will be able to do and what databases they will be able to alter.

New User

Creating a new user will be determined by the user type the “priv” is an attribute in the users table that will identify the permissions of the user logging in, like which databases they can alter and what information they can see. Filling out the “sign up” fields in the login screen will allow the addition of new users. All users will have to sign up through the same sign up fields but depending on the user profile being created some fields will not be required.

Existing users

Existing users will fill out the login fields and be sent to a webpage corresponding to their user privileges.

1. Government users

   Government users are able to create new resource fields. The GU’s will be able to edit the resources table, having the ability to edit the total amount of help the government program will be able to provide for various resources. Government users can also choose to view the total resources listed so far in the resources table. In viewing the resources table, GU’s will be able to determine how much help has been given and determine how much more, if any, can be offered. Also they can see which resources are the lowest in supply. GU’s can also view data in the government table where they will essentially see who else is in their government program and they will be able to view information from the corporation table to see which organizations are helping and how many volunteers there are per organization. The government users will also be the users that are able to dispense resources to specific regions.

2. Corporate users

    Corporate users have fewer permissions than GU’s. They cannot view the data in the government or corporate tables and cannot dispense resources. CU’s can only edit what they are choosing to contribute to the total resources table, they can add a new resource to donate and view the current state of the resources, such as how much there is in total and how much is needed. They cannot view how much of a given resource each region has. Government officials can always let charities know what is needed. They are also able to donate resources.

3. Civilian user

    Civilian users have different permissions than GU’s and CU’s. When a civilian logs in they are essentially signing in to fill out a damage report specifying what it is they need. This information will essentially be a record in the regions table where the civilian reports can be organized by region. The damage report is a raw gathering of data from each individual who makes a claim. The regions table takes the raw information from the report and shows the needs of each individual region. In using this design, further implementations are possible where once a region has a new shelter available for those without a home, the users whose addresses are nearest the shelter could be notified via the provided email or phone number. This is an implementation I do not have enough experience to add but by maintaining the individuality of the data there is more versatility for development and use.

The ability for civilians to update their need will be determined by allowing the input of negative values. This will in turn affect the resources and regions tables, essentially notifying the government users that progress has been made in supplying the needs of the people this also helps the government not waste resources.

Layout explanation

Government and corporate pages will have a Donate Resource button that will take them to the donate resources page. Government and corporate pages will also have a View Resources button that will take them to the View Resources page. Government and corporate pages will also have an Add Resource button that will take them to the Add Resource page.

Government pages will also have a view region button, they will be taken to the View Regions page. And a view volunteers page where they can view all the volunteers they have (corporation people), name and username username will become company name and password will act as an id. Second part of the code will become the company name.

View Regions – will use a page like the show recipes page with a search box so that they can enter the region id that they want to see. It will list the regions from the regions table. View Regions will display the region id from region_Id, number of dead from numdead, number of houses destroyed from destrhome

View resources- uses a page like the view recipes page to view all the resources available. This page will also have a Donate Resources button and an Add Resources button. This page will be getting its information from the total_resources table.

Add Resource - Use a page like the add ingredients page to add a new resource with the unit and amount

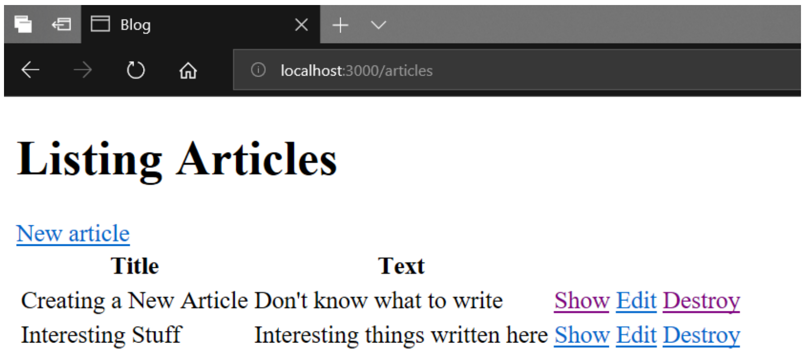
Donate Resource – will have a login form kind of appearance that will list all the resources available with check boxes next to the resource. To the right of the resource will be empty input text boxes where the user can enter the amount they want to donate this value will be added to the amount column of the total_resources table. Unit from the total_resources table will not be able to be modified from here

Framework Justification

Originally I was going to use React as the framework for my ERA website project, what I had failed to notice was that it was front-end only and would require a back-end framework as well. I felt that learning two frameworks would be over complicating things so I decided to switch over to Rails, a full stack framework. One of the things that I liked about rails is that it does many things automatically, it is quite intuitive. I also liked that rails has many guides and video tutorials, these were crucial to helping me learn to use the framework much faster. I am following tutorials to get used to Rails.

Blog Tutorial

The blog tutorial was my first introduction to ruby on rails, I followed a written online guide to create a blog application. I went through the entire tutorial once and implemented it as described in the instructions to make sure that I had a working product that I could study and analyze. This is what the blog tutorial looked like:



Blog

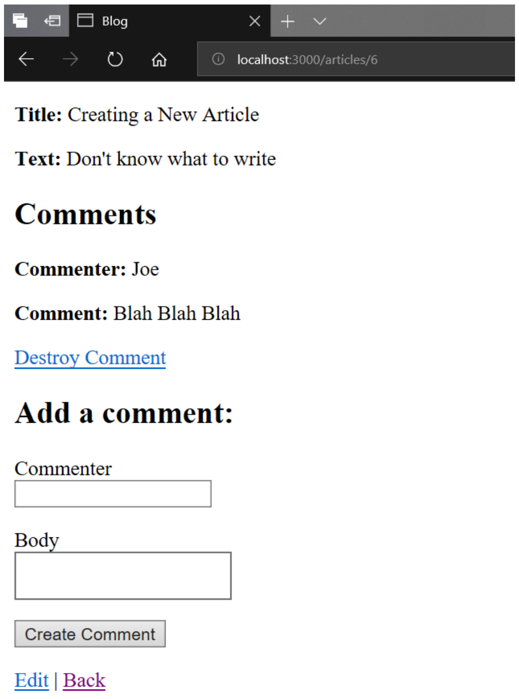
localhost:3000/articles

Listing Articles

New article

Title	Text			
Creating a New Article	Don't know what to write	Show	Edit	Destroy
Interesting Stuff	Interesting things written here	Show	Edit	Destroy

the blog part of the application



Blog

localhost:3000/articles/6

Title: Creating a New Article

Text: Don't know what to write

Comments

Commenter: Joe

Comment: Blah Blah Blah

Destroy Comment

Add a comment:

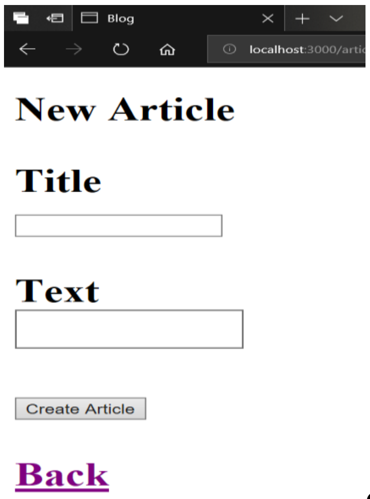
Commenter

Body

Create Comment

Edit | Back

viewing an article in the blog



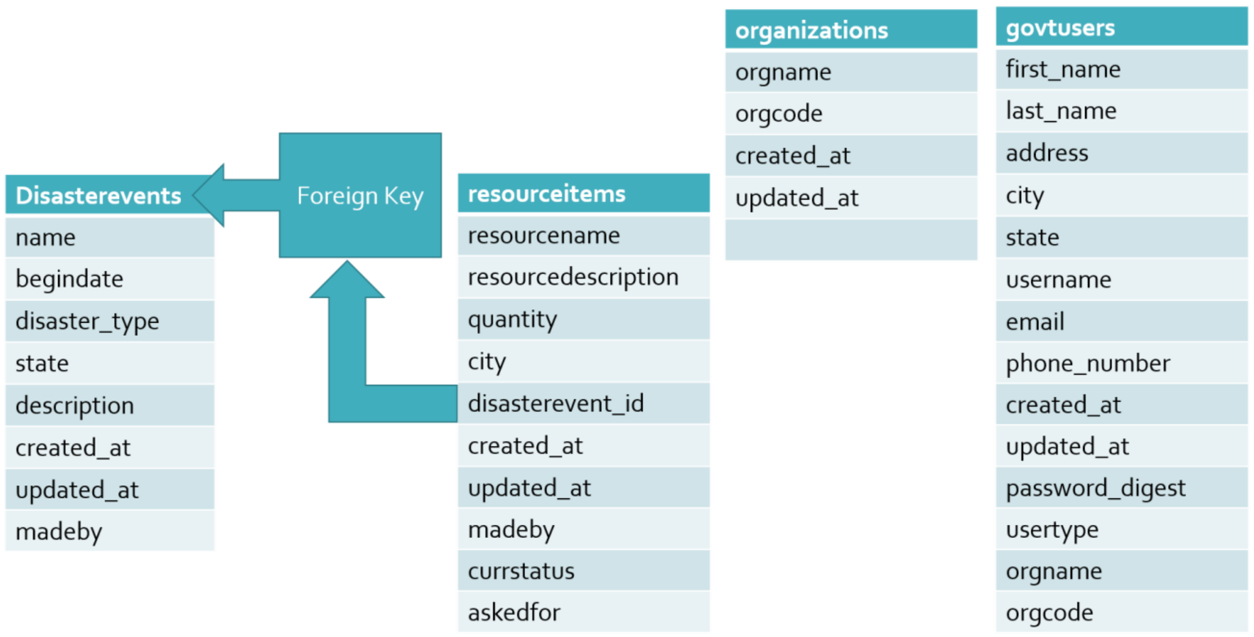
Creating a new article in the blog

While this tutorial did provide a good explanation of all the basic parts for building a web application it lacked a level of interaction between its parts that would have brought it to the next level.

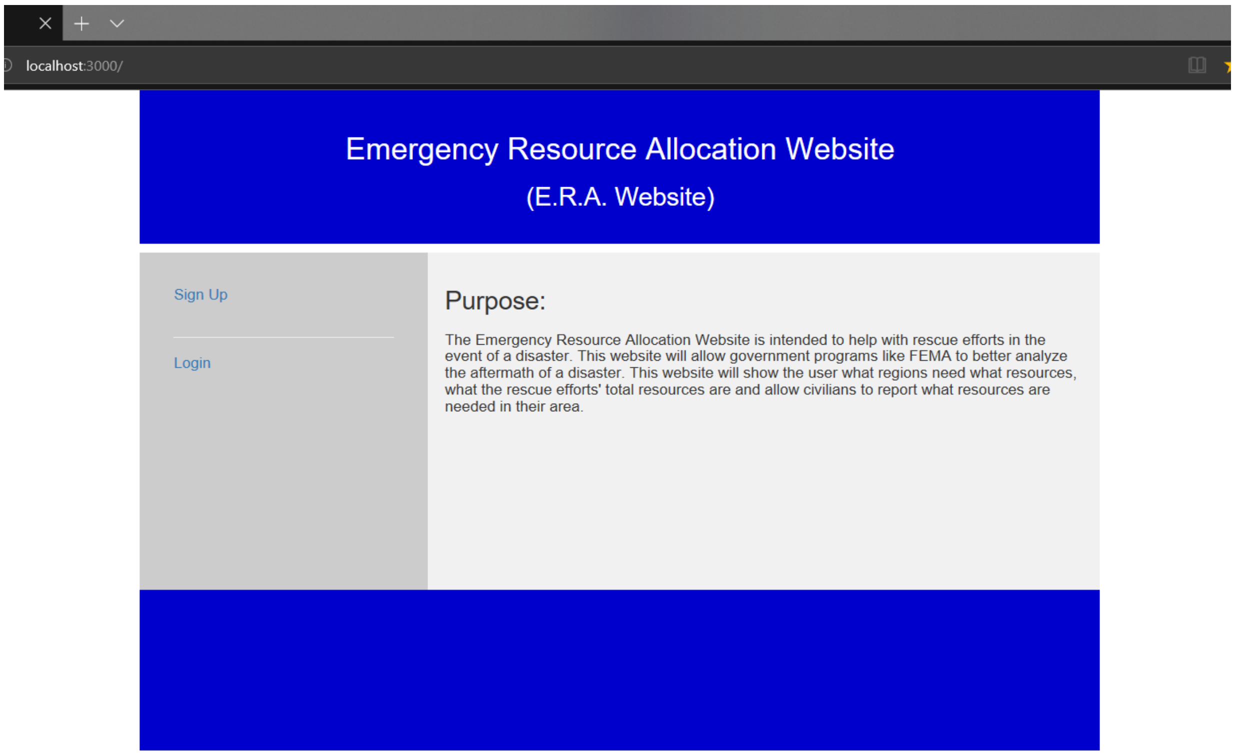
Plan and Design changes

One of the changes made from the original plans was the use of the "privilege" field in the user table. This column was going to be used to limit the abilities of the user according to the field's value. Instead of doing this I decided to generate two databases, one for the government users and one for the civilian users and using two different links for the sign up page. Looking to the future, with many more users, both government and civilian, searching will be reduced if they are separated anyway. The following is a walkthrough of the website with a description of each webpage and how I came about creating it and what my reasons were for designing it that way.

Database layout



Though there are not many databases in this project yet, they do accomplish all the primary tasks that I had set as the goals for my project. There is a relational connection between the resourceitems database and the disasterevents database, in that a disasterevent can have many resources associated with it and each resource report belongs to a specific disaster event. Originally I had considered some resources that would be more universal, in that they would be connected to many disaseterevents. The decision to not follow this path will be further explained in the Adding Resources section of this report.


localhost:3000/

Emergency Resource Allocation Website

(E.R.A. Website)

Sign Up

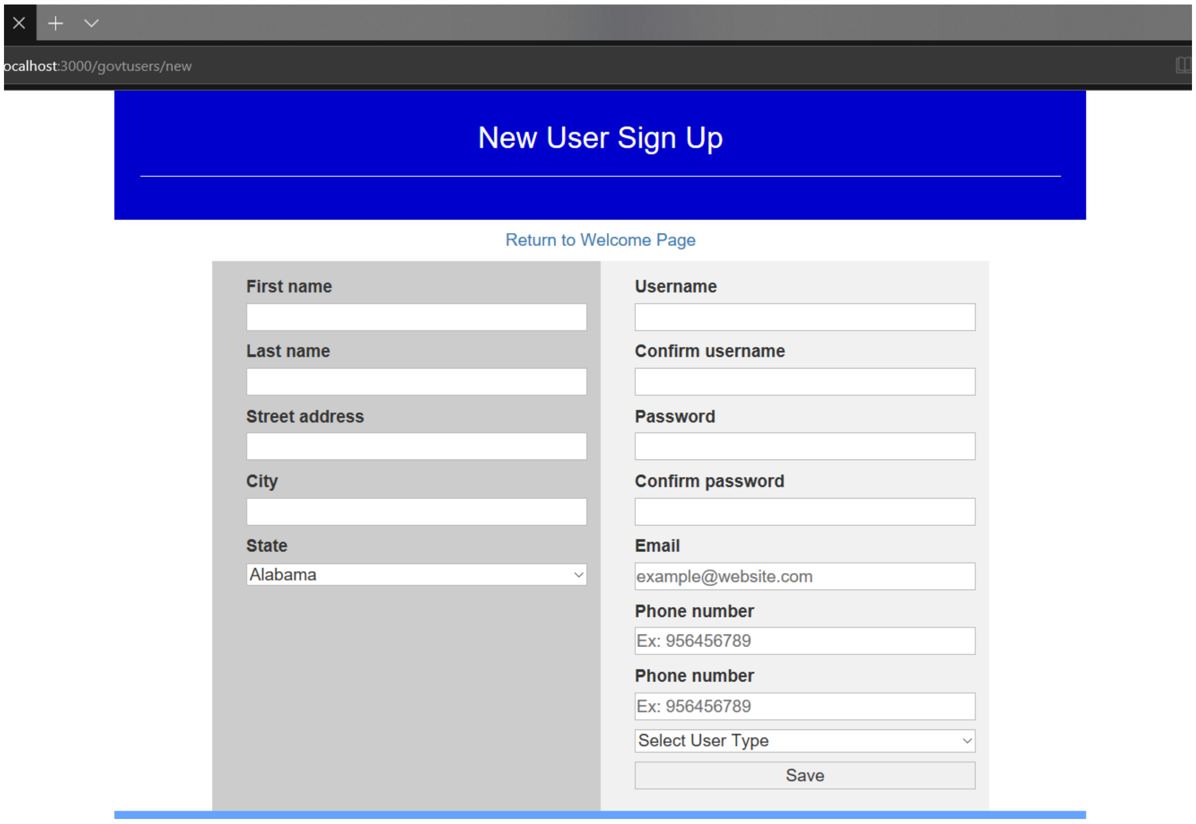
Login

Purpose:

The Emergency Resource Allocation Website is intended to help with rescue efforts in the event of a disaster. This website will allow government programs like FEMA to better analyze the aftermath of a disaster. This website will show the user what regions need what resources, what the rescue efforts' total resources are and allow civilians to report what resources are needed in their area.

The Welcome Page

The “Welcome page” is the simplest page of the website and intentionally so. I didn’t want to add too much to this page because if a user was really going thru a natural disaster I think they would find more comfort in a simple design rather than something that would appear much more complicated.



ocalhost:3000/govtusers/new

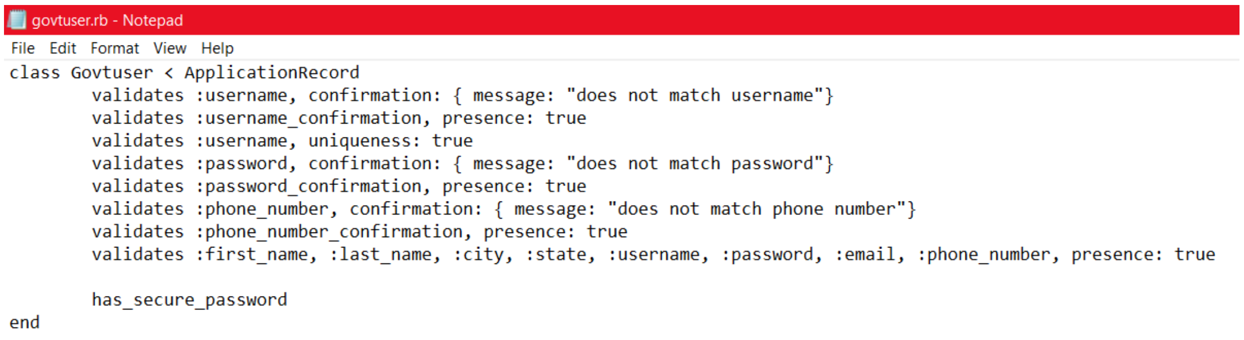
New User Sign Up

Return to Welcome Page

First name	Username
Last name	Confirm username
Street address	Password
City	Confirm password
State: Alabama	Email: example@website.com
	Phone number: Ex: 956456789
	Phone number: Ex: 956456789
	Select User Type
	Save

Sign-Up page

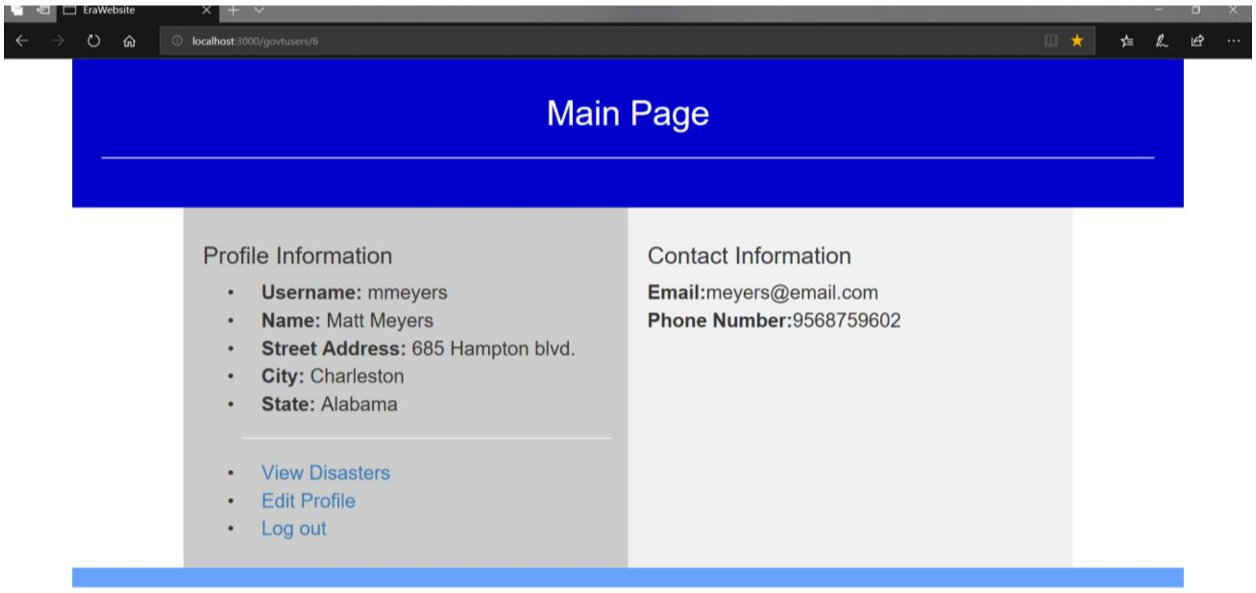
The original idea behind the “Sign Up” pages was to have one for each kind of user: Admin, Government, and Civilian. As I began to integrate the session’s part of the login process it seemed that it would be more troublesome to have the sessions controller set up to search multiple databases to verify whether or not a user was going to be allowed to login. Furthermore, at that point the SessionsHelper.rb file began to give me some trouble as well since the government users database table was explicitly the one it searched when I would use the Current_user command. Rewriting these commands seemed like it would take longer to implement than allowing all the users to log in thru the same page. Also it would make things messier and more confusing to anyone reading the code. I decided to stop using the Civiluser database and chose to add some more fields to the govt user database. I added the orgname and orgcode fields to the government user database to represent a sort of relation to the Organizations database. The entire purpose of the organizations database is merely to work as a kind of username/password storage area. Also since I was going to start having civilian users and the Admin I didn’t want to have to worry about them having foreign keys to nonexistent government organizations and their codes. New users will enter their first name, last name, phone number, email, and address: street/city/state/country. Government users are required to input all these fields as well as two more. When the user selects the Government option from the selection menu located just above the submit button they activate two hidden fields that they can now fill out. They will be inputting the name of their government organization and the corresponding government code. Government codes can be added and changed by Admin level users only. It would seem counterintuitive to deny the government users the ability to change the database that holds the name and code for their own organization but the reason I chose to do this is so that the Admin user would have to establish communications with any organization wanting to join website. This way they could act as a screener for mal-intent. After the submission of the forms the users are taken to their respective profile pages or shown flash messages that correspond to each user. There are validators in the govtuser.rb model:



```
class Govtuser < ApplicationRecord
        validates :username, confirmation: { message: "does not match username"}
        validates :username_confirmation, presence: true
        validates :username, uniqueness: true
        validates :password, confirmation: { message: "does not match password"}
        validates :password_confirmation, presence: true
        validates :phone_number, confirmation: { message: "does not match phone number"}
        validates :phone_number_confirmation, presence: true
        validates :first_name, :last_name, :city, :state, :username, :password, :email, :phone_number, presence: true

        has_secure_password
end
```

Here I used confirmations to help ensure the user entered their password correctly and that their phone number is correct. Another thing probably worth mentioning is that I used a username instead of an email for user identification. When I made this decision I had been reading about user capabilities and design that takes into account possible user limitations. It is more likely that someone will have a cell phone than it is likely they'll have an email address. So I made the confirmation of the telephone number a higher priority than making sure they got their email spelled out correctly.



Main Page

Profile Information

- **Username:** mmeyers
- **Name:** Matt Meyers
- **Street Address:** 685 Hampton blvd.
- **City:** Charleston
- **State:** Alabama

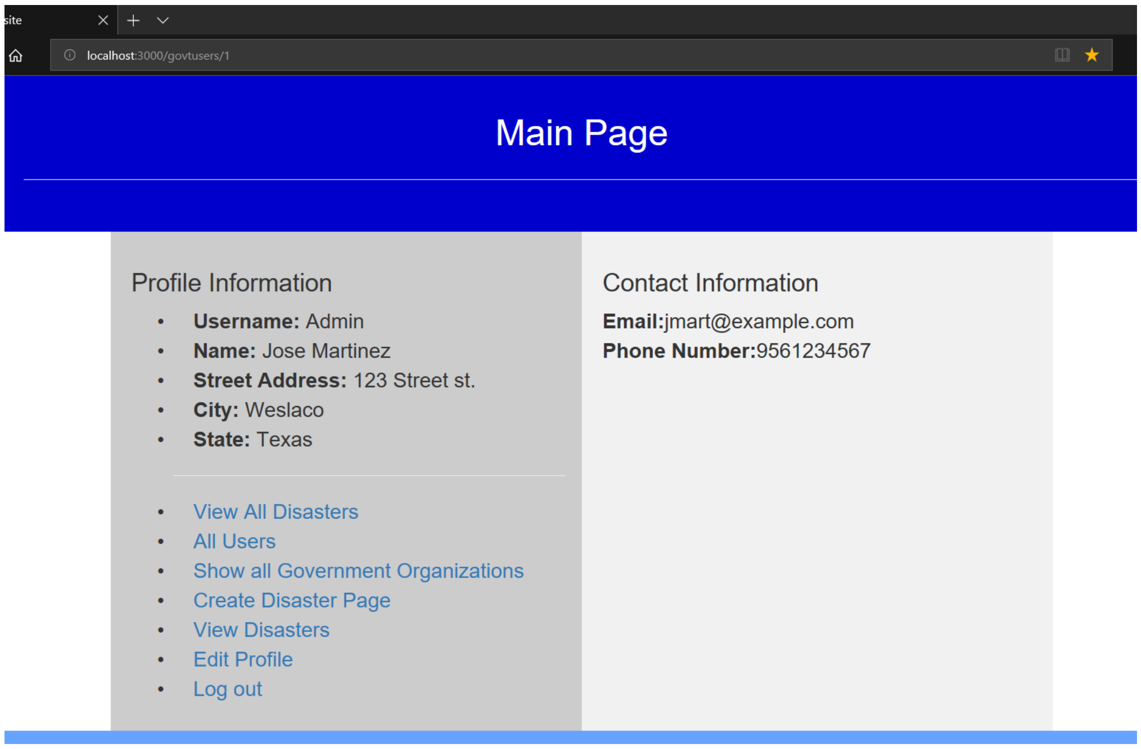
- View Disasters
- Edit Profile
- Log out

Contact Information

Email:meyers@email.com
Phone Number:9568759602

Profile Pages

Profile pages act as the central hub of the user experience with this website. On these pages the users see their own information and have links that lead them to corresponding pages for the activities each can make. Civilian users have a link that will allows them to see a list of currently active disaster pages, which will be filtered by the state they live in. Civilian users are also able to edit their own profile and log out of the website. The main reason I restricted the civilian users so much is that although I am trying to help them, I still don't think they should be trusted with too many options. The civilian main page is displayed above with no links beyond the main ability to edit their own profiles or view the disasters that have been created by them.



Main Page

Profile Information

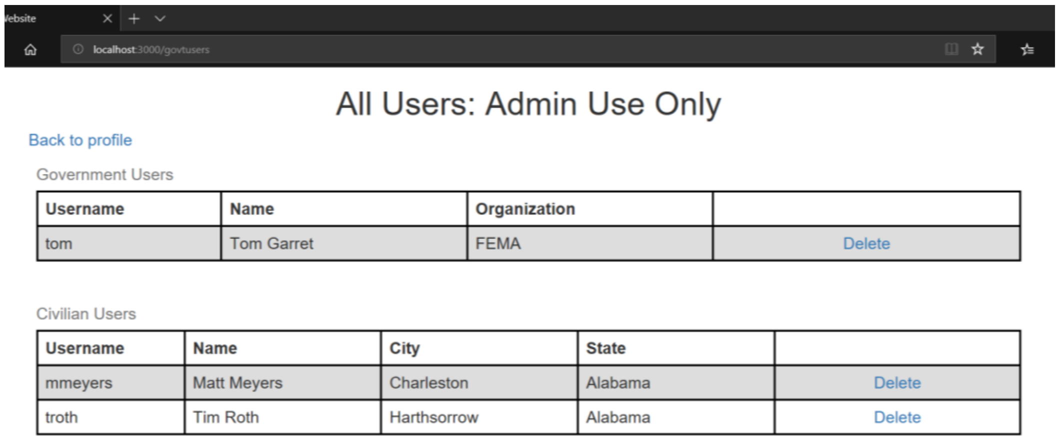
- **Username:** Admin
- **Name:** Jose Martinez
- **Street Address:** 123 Street st.
- **City:** Weslaco
- **State:** Texas

- View All Disasters
- All Users
- Show all Government Organizations
- Create Disaster Page
- View Disasters
- Edit Profile
- Log out

Contact Information

Email:jmart@example.com
Phone Number:9561234567

The admin user in the picture above clearly has more abilities than the civilian user, more so than it initially seems. The view all disasters link gives the admin user an unfiltered view of all the disasters currently existing on the websites databases. This is so that the admin user can monitor disaster page activity and make sure there aren't any malicious actions taking place. For example, a single government user creating dozens of disaster pages and the same group of certain people keep requesting odd things on that disaster would probably warrant a look into the things going on there. The "all users" link allows the user to see all the other users enlisted on the website, including government and civilian users.



All Users: Admin Use Only

Back to profile

Government Users

Username	Name	Organization	
tom	Tom Garret	FEMA	Delete

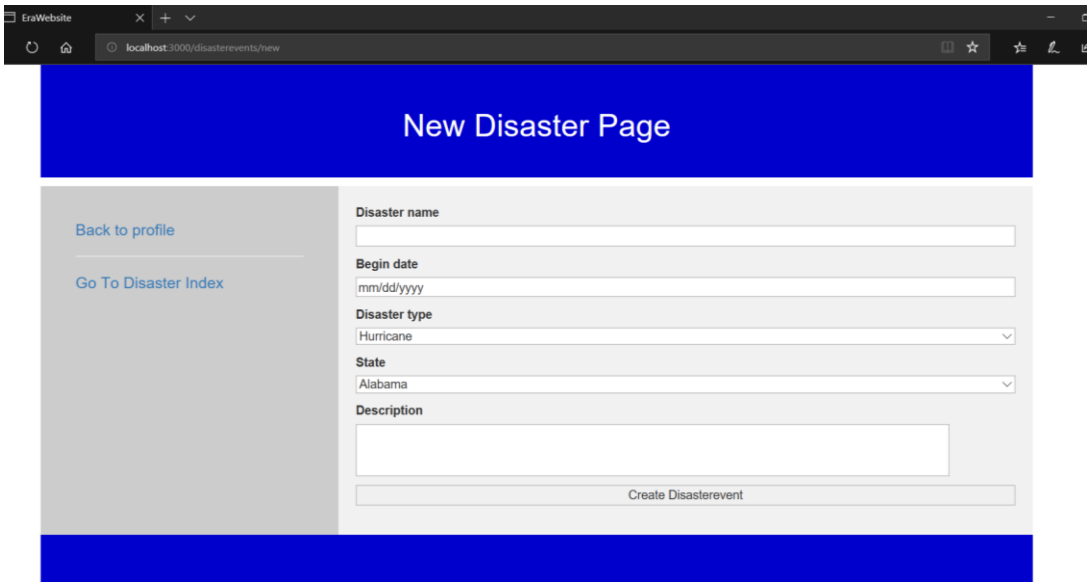
Civilian Users

Username	Name	City	State	
mmeyers	Matt Meyers	Charleston	Alabama	Delete
troth	Tim Roth	Harthsorrow	Alabama	Delete

An important thing I did here is I chose to no include a show link next to the delete link. That would give the admin user a bit too much information and it might make the users uncomfortable to know that someone had the ability to see their address and other contact information.

Moving on to the next link in the Admin user's profile is the "Show all Government Organizations". This ability is unique to the admin user because as mentioned earlier, it allows them to become a screener for new organizations that want to join the website.

The Create Disaster Page



Disaster name

Begin date
mm/dd/yyyy

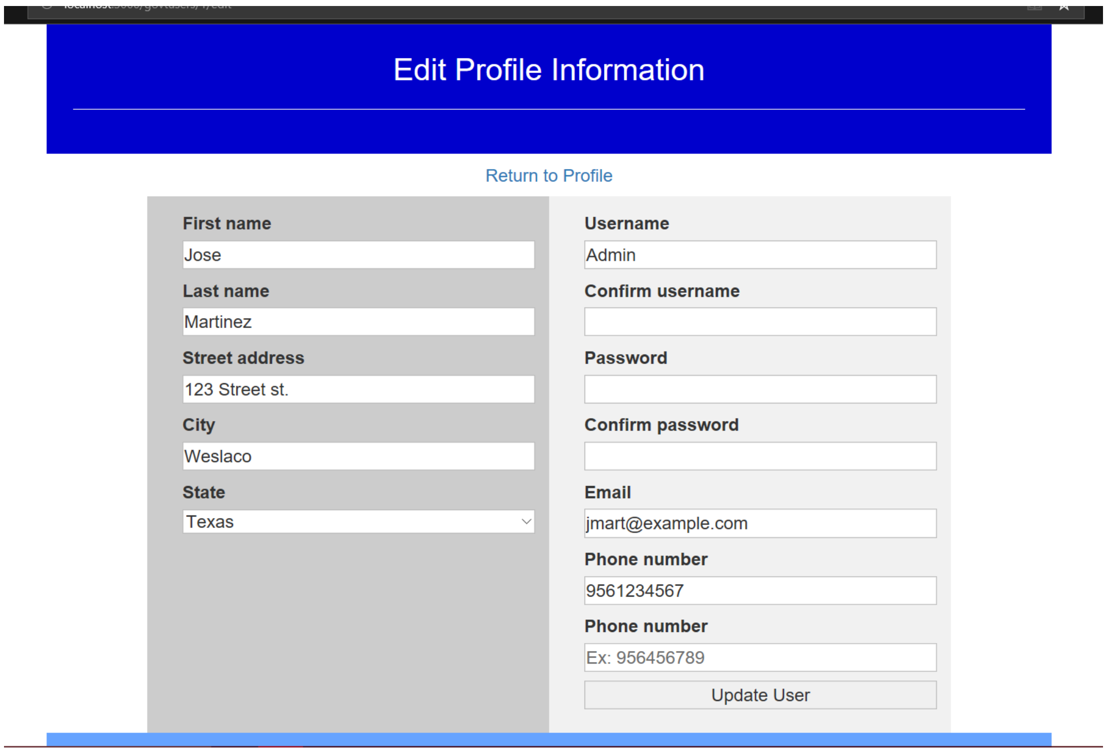
Disaster type
Hurricane

State
Alabama

Description

The create disaster page is not an ability limited to the admin user, the government user also has the option to create new disaster pages. Though it is not a visible field in the picture above, I included a hidden “madeby” field that automatically saves the username of the disaster page creator. In further enhancements to the website I would consider adding more fields to this page. I would even add fields that were disaster specific. Like mentioning flooding if the hurricane option is chosen or mentioning crumbled or destroyed buildings for the earthquake disaster.

Edit Profile



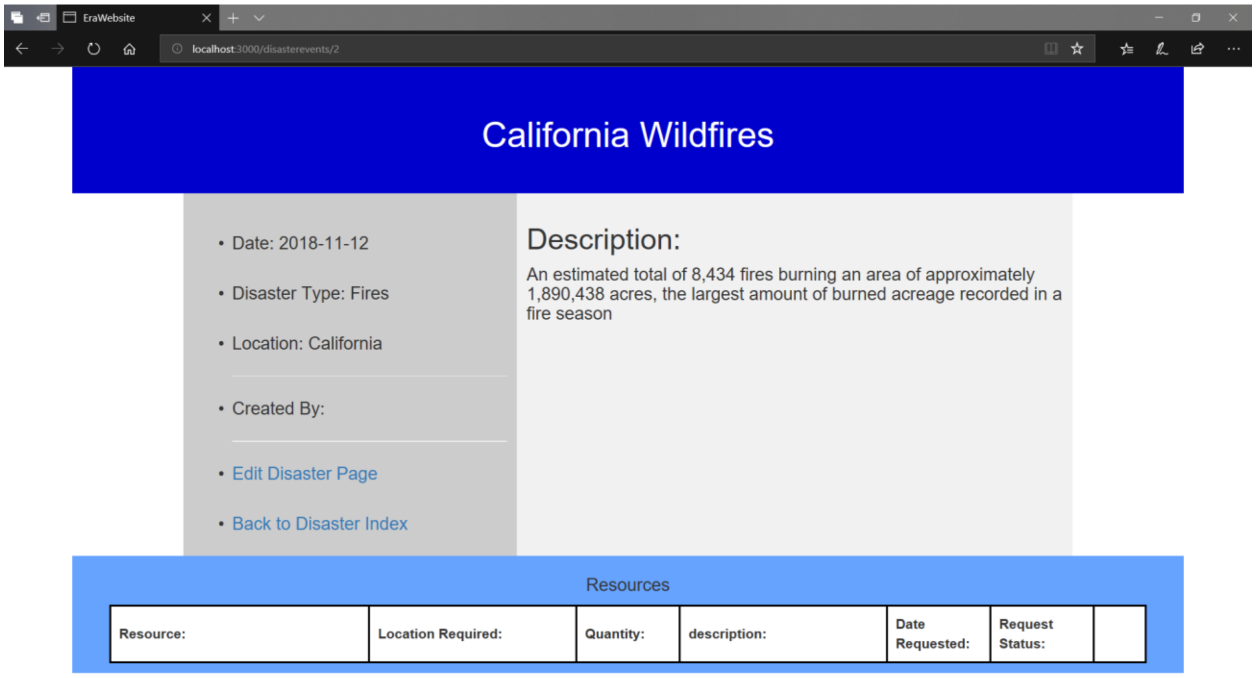
Edit Profile Information

First name: Jose	Username: Admin
Last name: Martinez	Confirm username:
Street address: 123 Street st.	Password:
City: Weslaco	Confirm password:
State: Texas	Email: jmart@example.com
	Phone number: 9561234567
	Phone number: Ex: 956456789

Clicking edit profile takes the user to a page similar to the sign-up screen, indeed I used partial renders for the new and edit views. The difference between the two views is that this edit view has the users information pre-loaded into fields and it runs an update instead of a create command. The rest of this page is pretty self-explanatory, the user is able to change whatever fields they want. Initially I wasn’t going to include the name fields in this view, but after some thought I decided that it was probable that someone could change their name and would want it reflected in their user profile, though having those fields in this view worries me that they could somehow miss-use their privileges and then change their profile name to hide their involvement.

View Disaster Page and Add Resource

Any user can view a disaster page, or at least those disaster pages related to them. Either made by the user or sharing the users same state location. Only the civilian users can add resources to a disaster page. Thus the following two versions of the disaster page are possible:



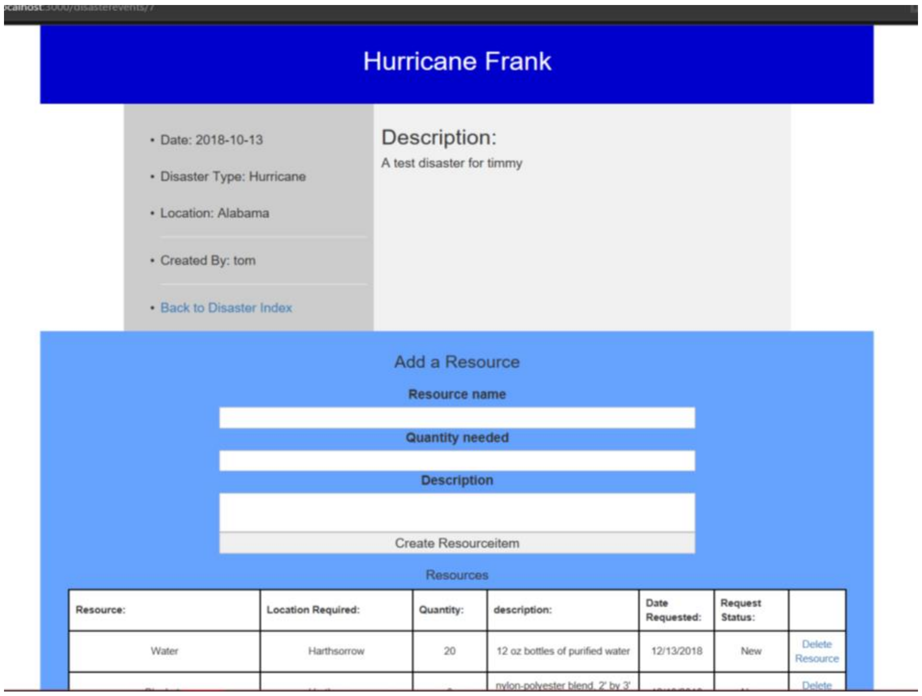
An estimated total of 8,434 fires burning an area of approximately 1,890,438 acres, the largest amount of burned acreage recorded in a fire season

Resources

Resource:	Location Required:	Quantity:	description:	Date Requested:	Request Status:	

The admin user's view of the disaster page before anyone has had a chance to request a resource

The admin user does not have the form that appears in the next screenshot:



Hurricane Frank

- Date: 2018-10-13
- Disaster Type: Hurricane
- Location: Alabama
- Created By: tom
- Back to Disaster Index

Description:
A test disaster for timmy

Add a Resource

Resource name

Quantity needed

Description

Create Resourceitem

Resources

Resource:	Location Required:	Quantity:	description:	Date Requested:	Request Status:	
Water	Harthsorrow	20	12 oz bottles of purified water	12/13/2018	New	Delete Resource
[illegible]	[illegible]	[illegible]	nylon-polyester blend, 2' by 3'	[illegible]	[illegible]	Delete

This screenshot shows how the civilian user can add a resource to this disaster page via the form that appears above the resources table. The “add a resource” form also has some more hidden fields. You may notice that the Resources table has a “date requested” column and a “request status” column that don’t appear in the form. This is to keep the user from “cheating” by trying to make it seem like their request has been there the longest and they deserve to have their request fulfilled first. I set these columns up by making them hidden then setting their default values as such:

```
<%= form.text_field :askedfor, value: DateTime.now.strftime("%m/%d/%Y") %>
```

The request status column is set up but not fully operational, to get this column working correctly, allowing it to be edited, meant setting an edit view and setting up some routes. I did these things and tried for several hours to get it to work and looked online at tutorials that were doing similar things but I think there must have been something that was set up differently between my project and those on the tutorials that changed some route or variable to the point where their solutions wouldn’t work for me. The first error I got while trying to set this up was a missing id's error when setting up the link to edit the column. I remedied this by passing the id in the route, but that led to another problem where the form loop in the partial rend was saying that the first argument could not be nil or uninitialized. So I set up some functions in the private section of the resourceitems controller that would find the relevant disasterevent and resourceitem beforehand. This was doing its part but for some reason I kept getting the same error and it was interfering with the display of the disaster pages. With time running short I decided that it would have been a good feature to have but one that I could not complete at the moment.

Conclusion

Learning how to program in ruby was overall a good experience. There are plenty of online videos, guides, and reference books to help a new programmer learn the ropes. Learning a new programming language is always advantageous but learning rails is a bit better because of all the companies that are running on rails these days. Companies like GitHub, Twitch, Shopify, and Airbnb are some easily recognizable and incredibly successful companies that were all built using ruby on rails. My website, though not perfect or extremely polished, achieved many of the goals from the initial design.

Future Works

A fully developed version of this project would attempt to take this data from the some other government document, a census program or city planning document. Some of the attributes the database would need to include would be number of people injured or needing medical assistance, whether or not an area has water, food, and electricity. Other attributes would be the status of an area such as whether or not it is flooded, has large amounts of fires, and the number of buildings or homes destroyed. The number of homes destroyed could be calculated by people logging into a website, typing in their address, and indicating whether their home was destroyed. A large amount of data can be gathered by allowing people to individually report what is needed via website. Phone companies could submit information regarding which of their cell towers were affected, since this would indicate a lack of communication available in a given area, a high level of down cell towers would indicate that a survey team would need to be sent to that area.