Bangladesh University of Professionals

FACULTY OF SCIENCE & TECHNOLOGY

DEPARTMENT OF ICT

Web Technology Lab Project

Disaster Relief Website

Submitted To

Dr. Shamim Al Mamun
Professor
IIT, JU

Md. Shahiduzzaman
Assistant Professor
CSE, BUBT

Submitted By

Anindita Roy (ID: 2154901080)

Fayzul Noor Tabeeb (ID: 2154901084)

Dipto Das (ID: 2154901092)

Date of Submission: June 6, 2024

Abstract

Disaster Relief is a humanitarian website designed to bridge the gap between aid providers and disaster-affected individuals. The primary goal of this website is twofold: to enable victims of natural disasters to request relief aid, and to help volunteers find and assist these victims based on their specific needs. This platform aims to eliminate discrepancies between victims and volunteers by allowing victims to precisely state their needs, and volunteers to accurately understand and fulfill them. Victims can use the "Request Relief" section to specify the items they need and their quantities, ensuring their requests are clear and precise. This helps volunteers meet the victims' needs more effectively. On the "Volunteer Task" page, volunteers can browse through posts made by victims, which have been approved by the admin. These posts provide detailed information about each victim's needs and condition. When a volunteer accepts a victim's request, the website generates a real-time route to guide the volunteer to the victim, facilitating prompt delivery of the relief aid. Additionally, the "Disaster Forecast" page offers a real-time interactive weather map, helping both victims and volunteers stay informed about the weather conditions in various areas. This feature enhances their ability to prepare and respond accordingly.

Table of Contents

Introduction

Overview of the project:

Disaster Relief website actually connects the natural disaster victims with the volunteers who will provide the aid. The key feature “Request Relief” section is for the victim who specify his/her needed supply. At the same time “Volunteer Task” section is dedicated for the volunteers who can view all the aid request post and can respond to these requests. Another key feature “Disaster Forecast” presents a real- time weather map to inform the users of current conditions, also the volunteer can verify the request location by this map seeing the condition of the weather.

Purpose and Significance of the Project:

Our main focus to develop this site is to create a platform between the victim and the volunteer where victims can clearly state their needs and volunteer can accurately understand and fulfill those needs. In disaster relief scenarios, miscommunication and incomplete information can result in unfulfilled requirements and unnecessary resource consumption. This lessens the indiscipline that are frequently observed in these situations. By offering real-time routing for volunteers and up-to-date weather forecasts, the project also enhances the coordination of relief efforts.

Scope of projects:

User side:

1. Request Relief: allows victims to request specific supplies, quantities also can provide a precise and clear information about their needs.
2. Volunteer Task: enable volunteers to accept and decline the aid requests, ensuring they can provide the best assistance based on victim’s need.
3. Real-time routing: accurately locate victim and volunteer dynamically and route volunteers to the location of the victim.
4. Disaster Forecast: provides an interactive weather map that helps user to get not only past and current weather situation but also gives the predicted future state of weather.

Admin side:

1. Admin Accounts: create and delete accounts
2. User Accounts: delete accounts
3. Pending Post: the relief requests which are posted by the victims, admin can approve and decile after verifying. Then the post will visible in “Volunteer Task” section.

4. Disaster Alert: Admin also can post about the updates like- the weather situation alert, volunteer updates, relief updates.

Contributions:

Here are the contributions for this project

- Project management: Ensured effective communication and collaboration among team members.
- Website Development: Designed and developed the user interface for a seamless and intuitive user experience. Implemented front-end and back-end functionalities using PHP, html.
- Database Management: Set up and maintained the database for storing user information, requests, and volunteer responses.
- API: Integrated APIs to provide accurate and up-to-date weather information.

By contributing to these areas, the team ensured the successful development and deployment of the Disaster Relief website, providing a valuable tool for disaster response and aid coordination.

Methodology

The development process of this website had three phases. It started with the planning phase, where a rough idea of the website was drawn. Then we moved on to designing phase where a visual representation was given to the initial idea. Layouts for all the webpages were created in this phase. The development process ended with implementation phase. In this phase we the idea of the website was brought to reality in a fully functional way. Many technologies and tools were used throughout these phases. These are mentioned as follows

Tools:

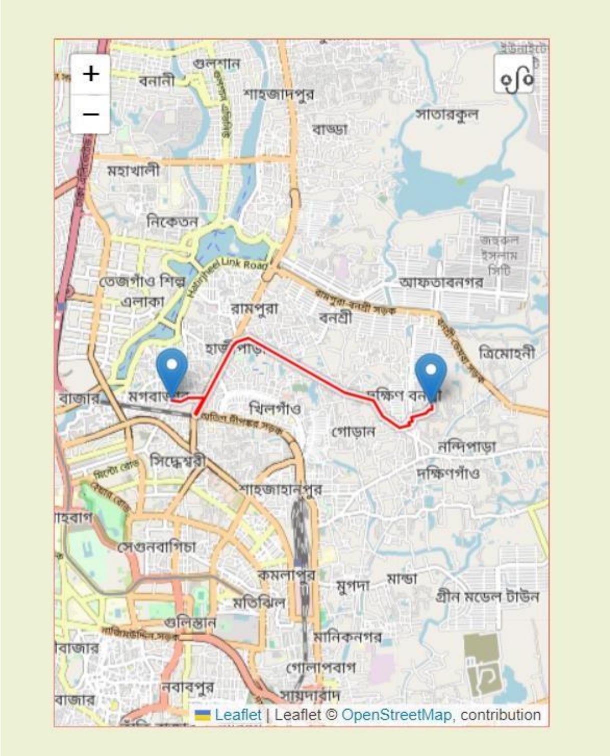
1. Colorhunt website: for finding the color pallet of the website [1]
2. Figma: for sketching out layouts of all the webpages [2]
3. Leaflet JS: for developing dynamic routing system [3][4]
4. Ventusky Website: for developing a weather map page [5]

Technologies:

1. XAMPP: for setting up Local Host server
2. Visual Studio: for editing and coding
3. phpMyAdmin: for handling the administration of MySQL over the Web

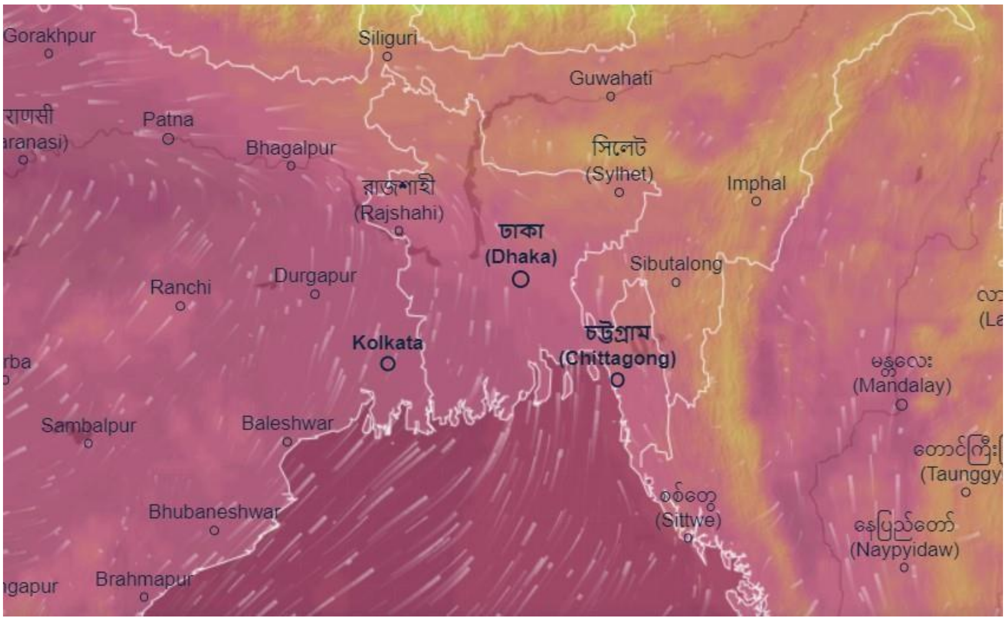
The major work of development process started in the designing phase. In this phase Colorhunt and Figma played a very big role. Colorhunt website helped us to choose color pallet for our website. After finalizing color pallet, we went on to design the webpages of our website using Figma. The main challenge that was faced in this phase was to make the website's layout as user friendly and decent looking as possible. Even though all the webpages were designed in this phase, but not all webpage's design made it to the final product's appearance. Some changes were brought later based on the working mechanism of those webpages. Changes were also brought to remove any sort of misunderstanding that could have been created at user's end because of the design.

After the completion of design face, software development phase started. User of XAMPP, phpMyAdmin, Microsoft Visual Studio along with HTML, CSS, JavaScript, MySQL Database were heavily used in this process. XAMPP software helped us to dispatch our device as a local host server and deploy our website on it. It also provided us with phpMyAdmin service which helped us to manage and share data using MySQL database through the locally hosted website. The whole coding process was done on VS IDE where we used HTML, CSS and JS accordingly to bring the vision of this website to reality. Leaflet JS library was used in the development phase in order to implement the dynamic routing mechanism feature of the website. Not only that it was also used for collecting latitude and longitude value of the victim during his/her request posting procedure.

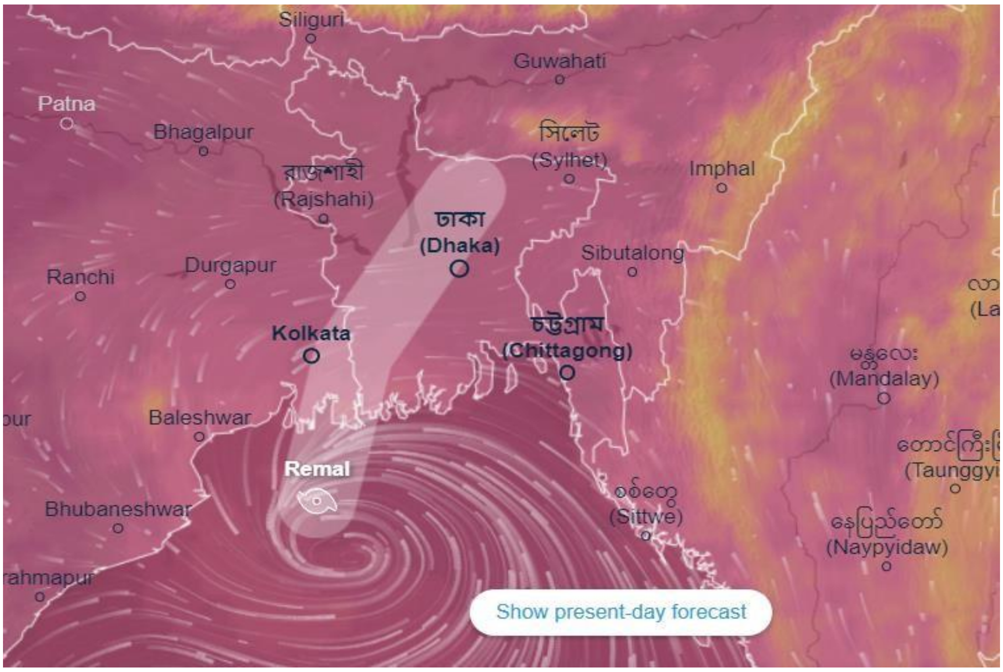


Dynamic Routing System

Help of the website “ventusky” was also taken for developing a weather map webpage (Webpage name: Disaster Forecast) for the website. Through this webpage users can get weather info of present and past days along with predicted weather of future in an easy to understand, animated and interactive way. This webpage can also aware users about incoming disasters like cyclone, tornado, hurricane along with their travel path. It can even show weather abnormalities like extremely heavy rainfall and drought.



Normal Weather Scenario



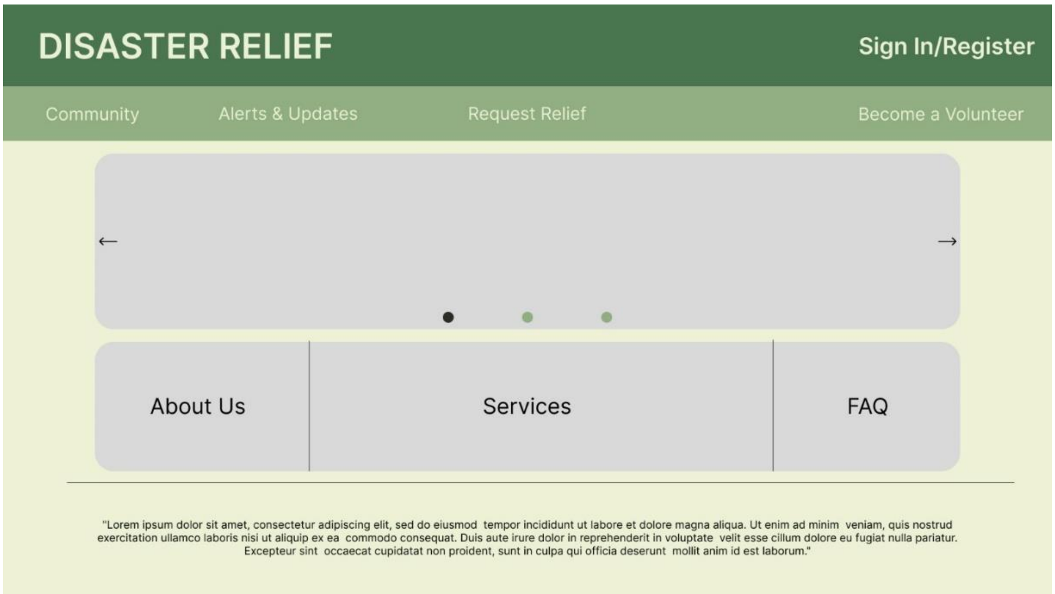
Abnormal Weather Scenario

Design

The website frontend is created using HTML, JS and CSS. PHP was used for the backend, along with MySQL database. Additionally, the leaflet and leaflet-routing-machine libraries were used to provide mapping and routing.

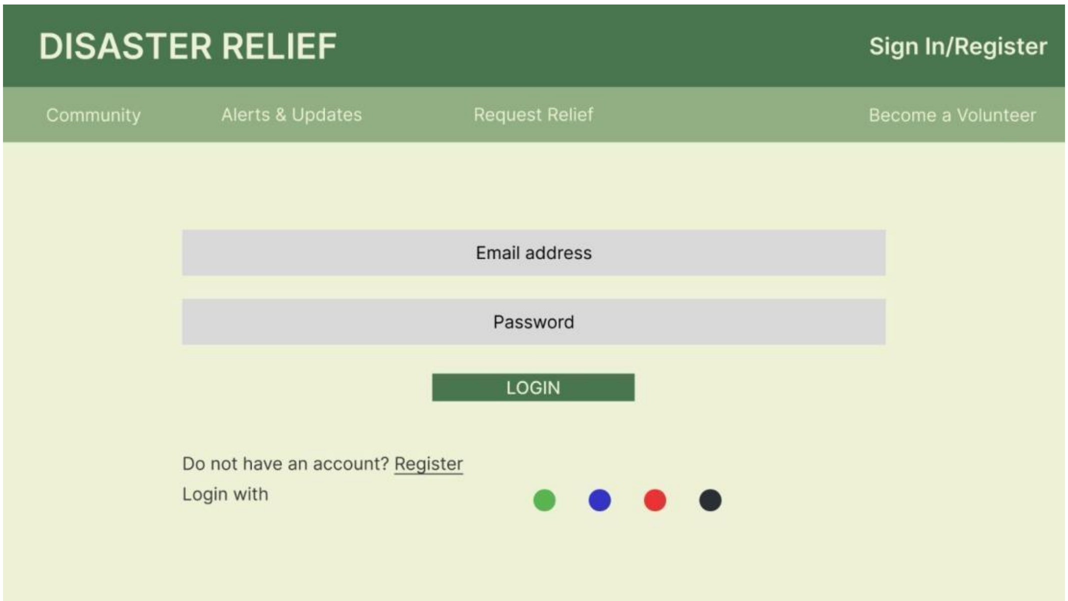
The website has two sides, one user-facing and one admin side. The user facing side consists of a few pages, where the user can request for help and track the progress of any requests they have made. The admin side allows for requests to be either accepted or rejected, and to see information about any user of the website. The MySQL database is used to store request information, along with other metadata, such as the user ID, time of request, status etc. Below are the wireframes of the website design.

User Side:



"Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum."

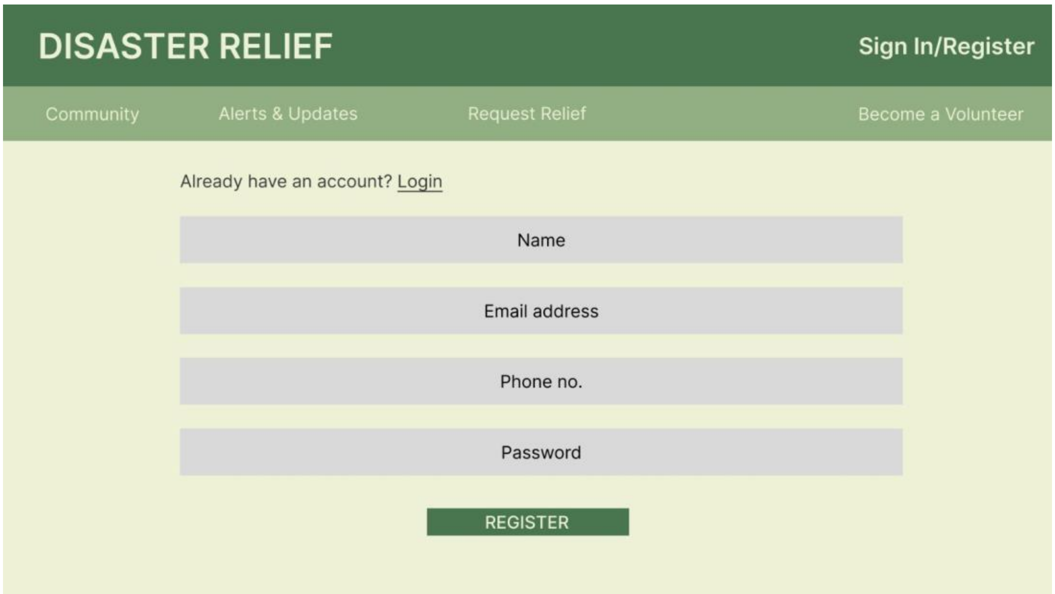
User-facing home page



Do not have an account? Register

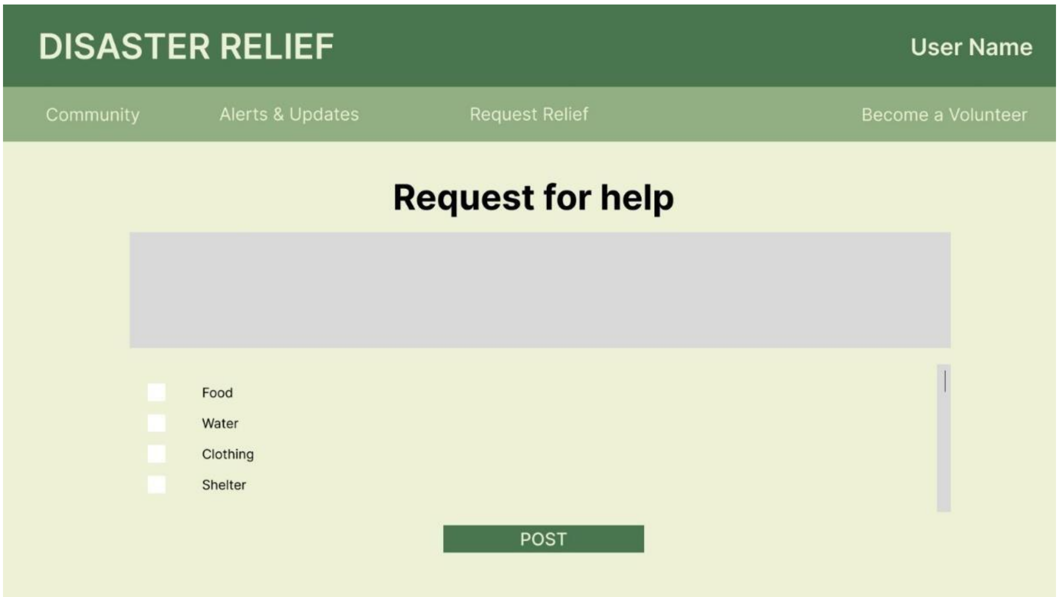
Login with

User Login Page

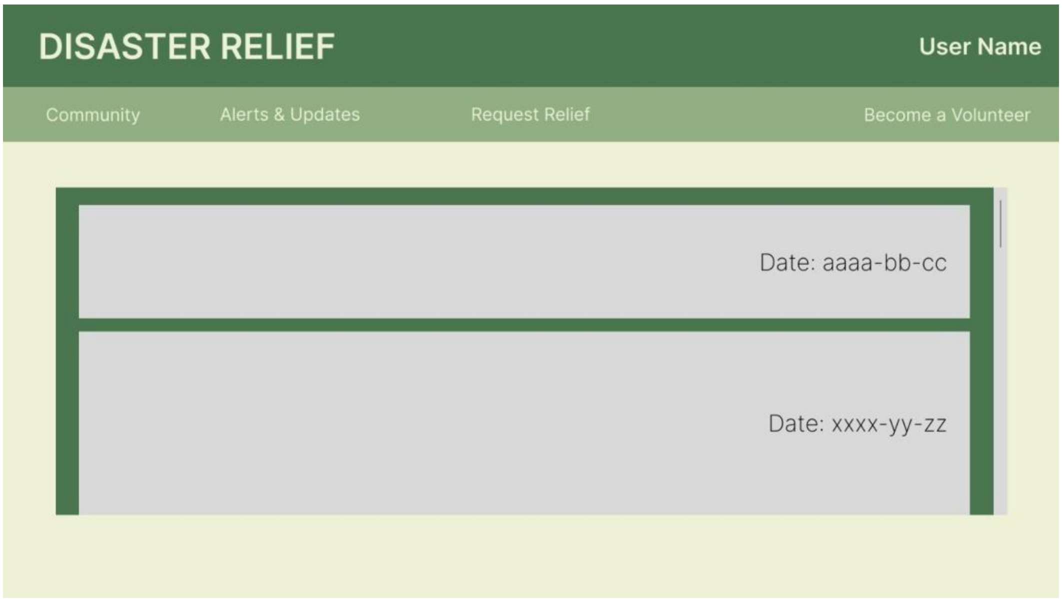


Already have an account? Login

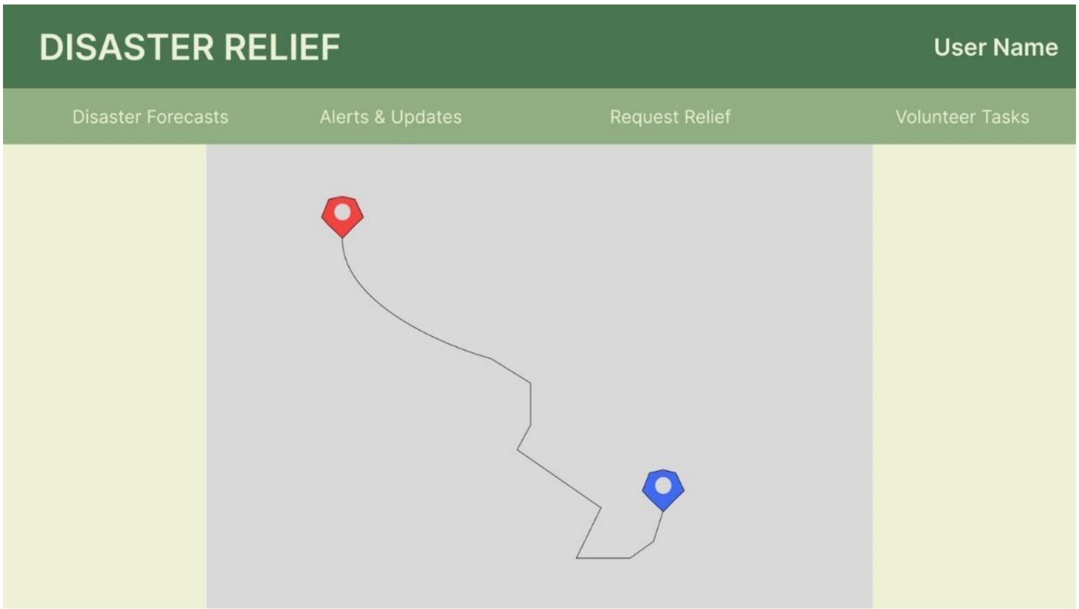
User Registration Page



Help Request Page

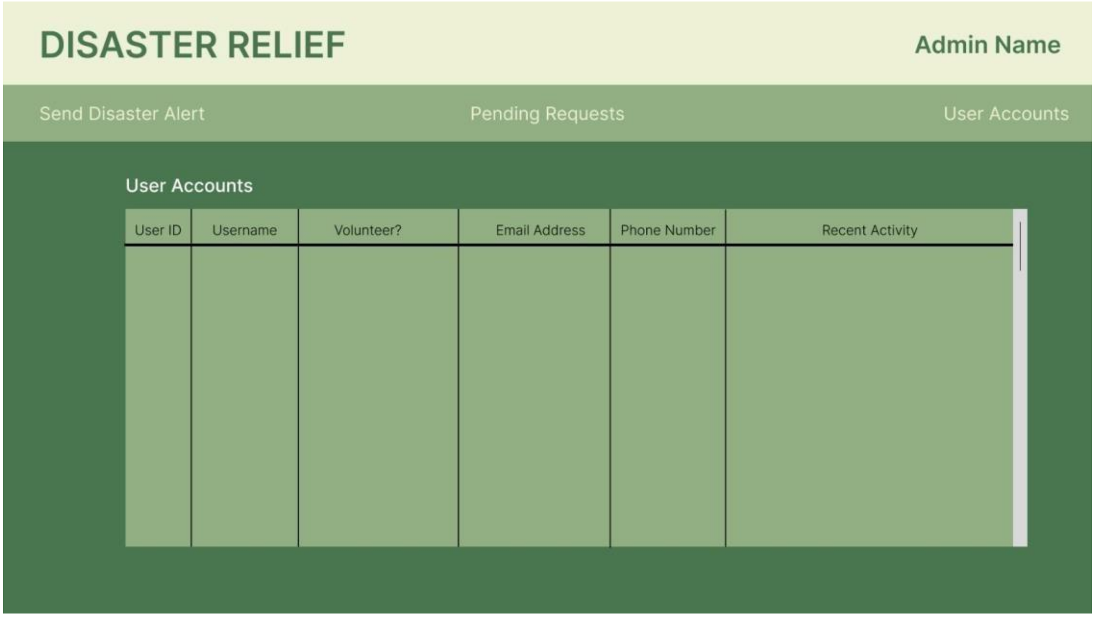


User Alerts & Updates



User Map and Routing Page

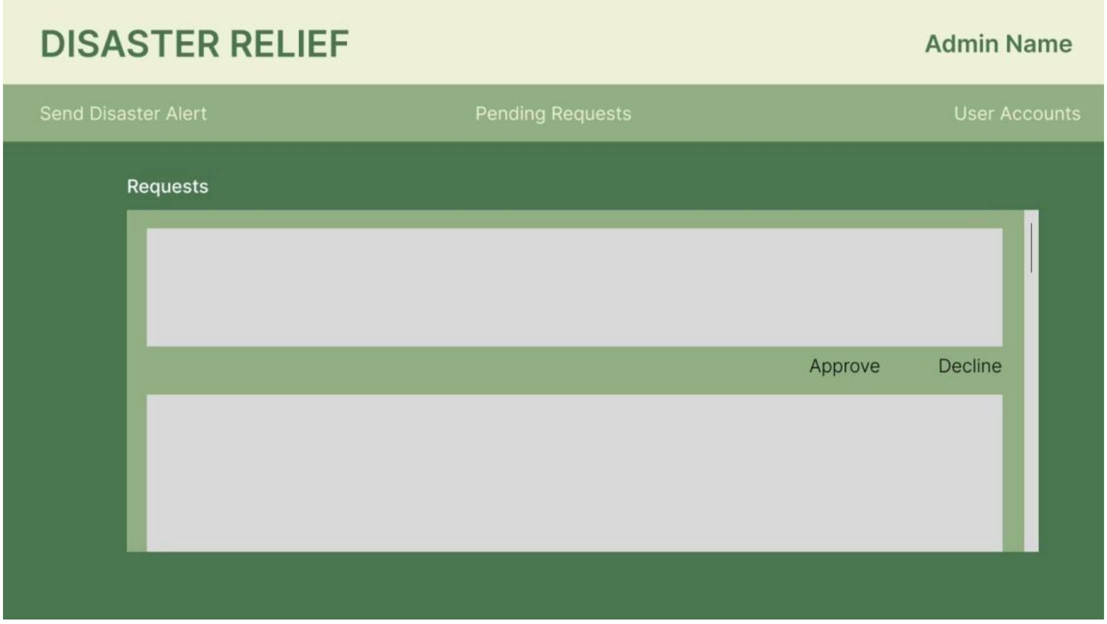
Admin Side



User Accounts

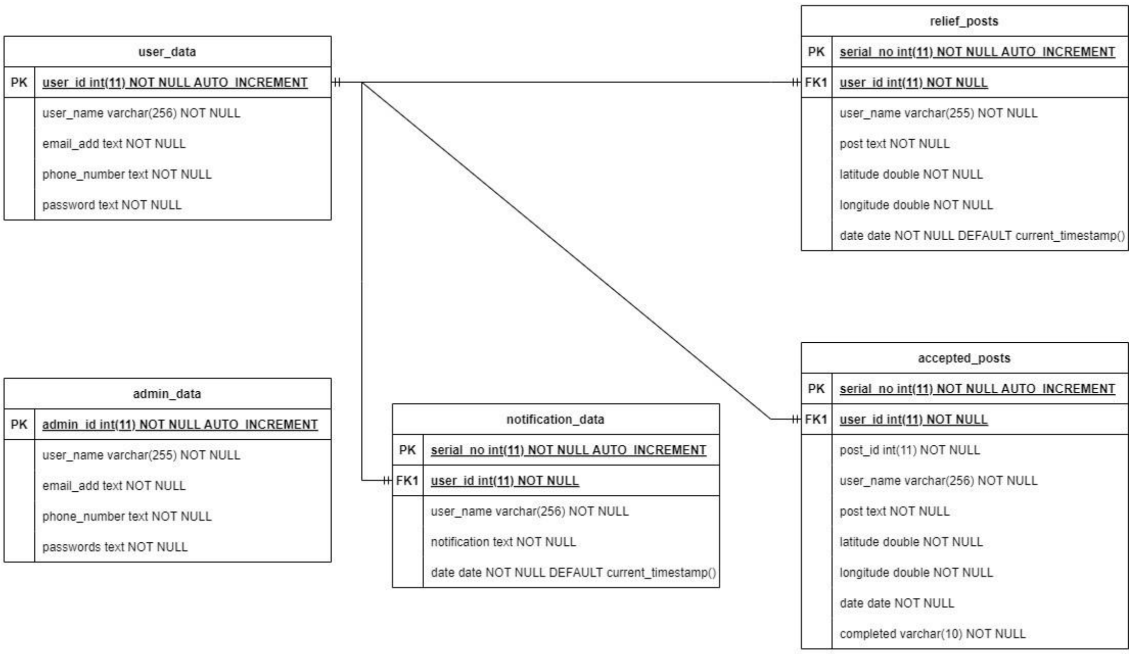
User ID	Username	Volunteer?	Email Address	Phone Number	Recent Activity

Admin-side User Account Page



Admin-side Request Approval Page

E-R Diagram of MySQL Database:



	user_data
PK	user_id int(11) NOT NULL AUTO_INCREMENT
	user_name varchar(256) NOT NULL
	email_add text NOT NULL
	phone_number text NOT NULL
	password text NOT NULL

	relief_posts
PK	serial_no int(11) NOT NULL AUTO_INCREMENT
FK1	user_id int(11) NOT NULL
	user_name varchar(255) NOT NULL
	post text NOT NULL
	latitude double NOT NULL
	longitude double NOT NULL
	date date NOT NULL DEFAULT current_timestamp()

	admin_data
PK	admin_id int(11) NOT NULL AUTO_INCREMENT
	user_name varchar(255) NOT NULL
	email_add text NOT NULL
	phone_number text NOT NULL
	passwords text NOT NULL

	notification_data
PK	serial_no int(11) NOT NULL AUTO_INCREMENT
FK1	user_id int(11) NOT NULL
	user_name varchar(256) NOT NULL
	notification text NOT NULL
	date date NOT NULL DEFAULT current_timestamp()

	accepted_posts
PK	serial_no int(11) NOT NULL AUTO_INCREMENT
FK1	user_id int(11) NOT NULL
	post_id int(11) NOT NULL
	user_name varchar(256) NOT NULL
	post text NOT NULL
	latitude double NOT NULL
	longitude double NOT NULL
	date date NOT NULL
	completed varchar(10) NOT NULL

Implementation

The key functionalities of the website are carried in the “Request Relief”, “Volunteer Task” page of client side and “Pending Posts” page, “Disaster Alert” page of admin side.

Request Relief Page (user side):

In this page victims can ask for relief aid from other user’s of the platform by posting their need in the text input area. When a user uploads a post, the website using the gps feature of the device collects the latitude and longitude value of the victim’s location. It then sends the input data along with latitude and longitude coordinates to the database.

```
<form action="requestAction.php" method="post">
        <div class="container textbox-container">
            <div>
                <textarea id="message" name="message" rows="9"
cols="90" placeholder="mention the amount of chosen goods or
volunteer will bring amount according to his
capability"></textarea>
            </div>
        </div>
        <input type="hidden" name="js_lat_var" id="js_lat"
value="">
        <input type="hidden" name="js_lon_var" id="js_lon"
value="">
        <button id="postButton" onclick="submit">Post</button>
    </form>
```

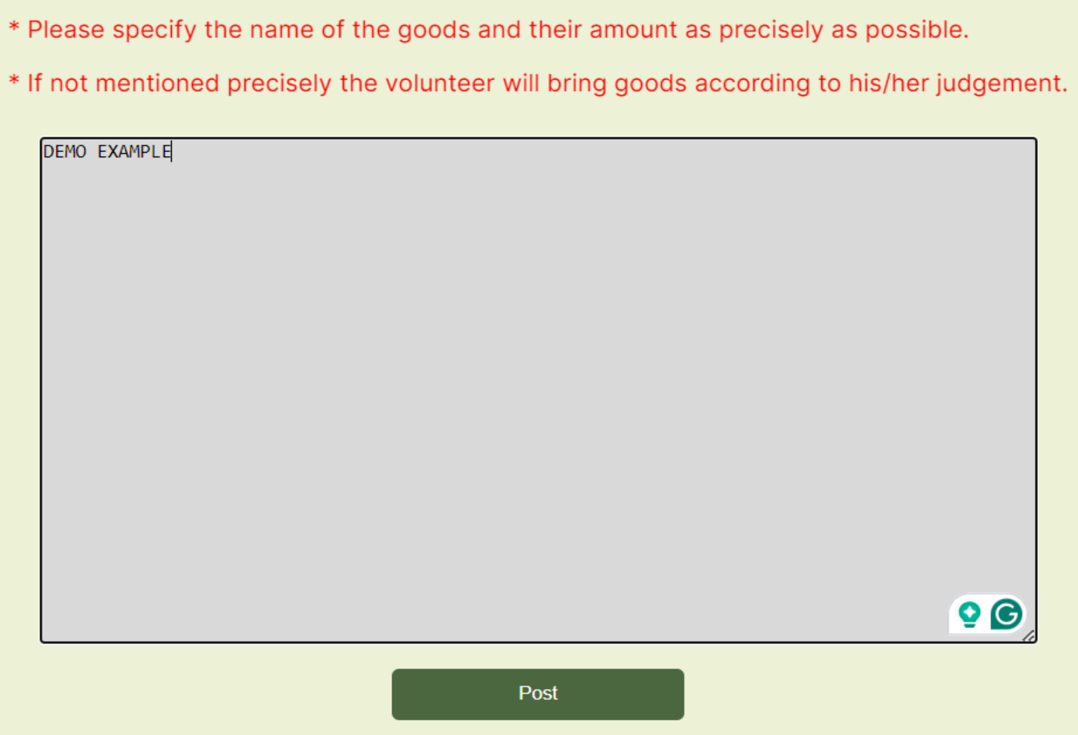
This is the form through which the webpage is taking victims input in form of text. Three input fields were taken for implementing this task. Two of which are kept hidden. These two inputs fields are sending the latitude and longitude co-ordinates. This is being done so with the help of a JS code.

```
var lat;
var lon;

if(!navigator.geolocation){
    console.log("Your browser doesnt support geolocaton
feature!")
} else {
    navigator.geolocation.getCurrentPosition(getposition)
}

function getposition(position){
    console.log(position)
    lat = position.coords.latitude;
    lon = position.coords.longitude;
}
```

The above JS script by using the leaflet library finds out the latitude and longitude value of the victim’s location. When the “getposition()” function of the code is called, the JS script uses the device’s gps service and generate an in-detail location data of the user and stores it in the “positon” variable. From that variable the script file extracts the latitude and longitude co-ordinates then manipulates the hidden text input field’s of the form with those values simultaneously and then uploads the form data to the database.

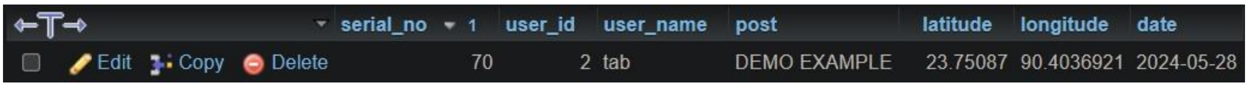


* Please specify the name of the goods and their amount as precisely as possible.

* If not mentioned precisely the volunteer will bring goods according to his/her judgement.

DEMO EXAMPLE

Fig: Request Relief Page of user end

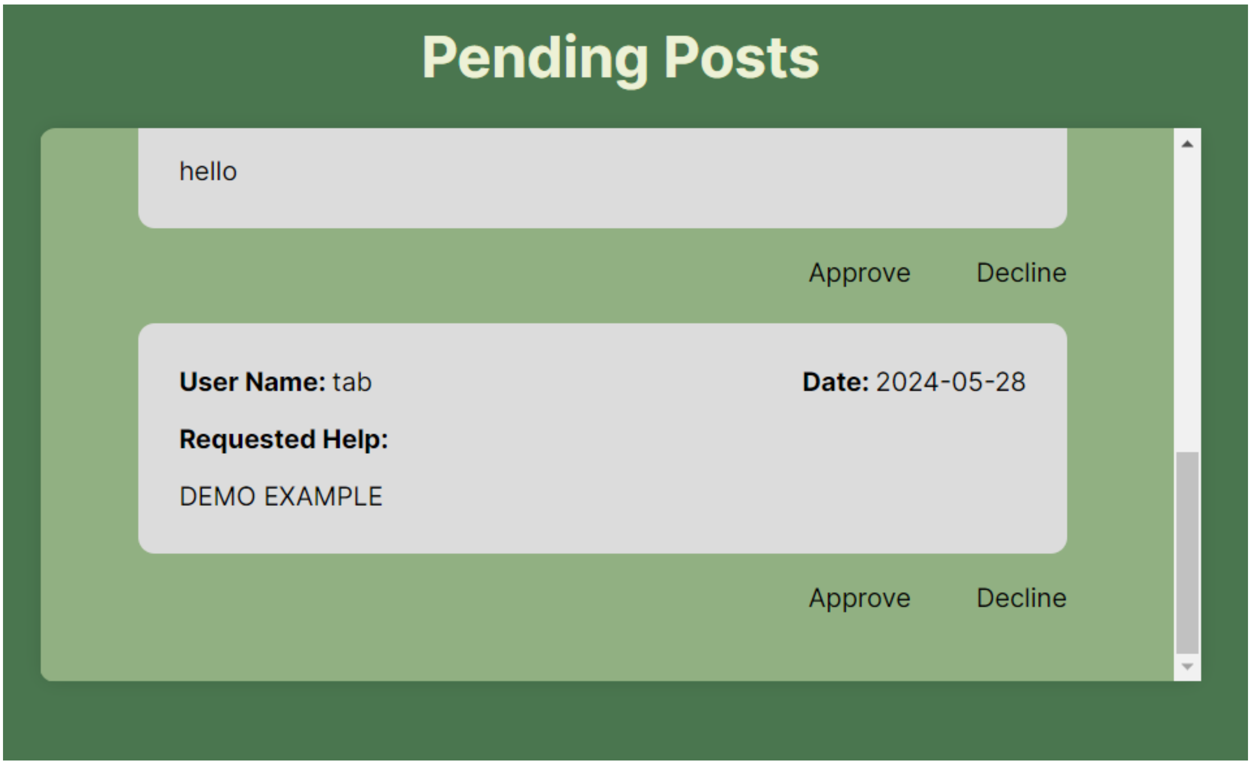


	serial_no ▾ 1	user_id	user_name	post	latitude	longitude	date
□ Edit Copy Delete	70	2	tab	DEMO EXAMPLE	23.75087	90.4036921	2024-05-28

Fig: Uploaded post of user in the “relief_post” database

Pending Post Page (admin side):

After the victim has made his/her post the data gets send to the database. Using that database all the submitted post of various users are shown at admin’s end for moderation. Through this page the admin can moderate the post by approving and declining them. If the post is declined it will simply get deleted from the database and the user who posted it will get a notification in the alerts and updates page. Where as if the admin approves the post, the post data will be moved to a separate database through which all the other users can see that approved post in the “Volunteer Task” page and an approval notification will be sent to the user who made that post.


Pending Posts

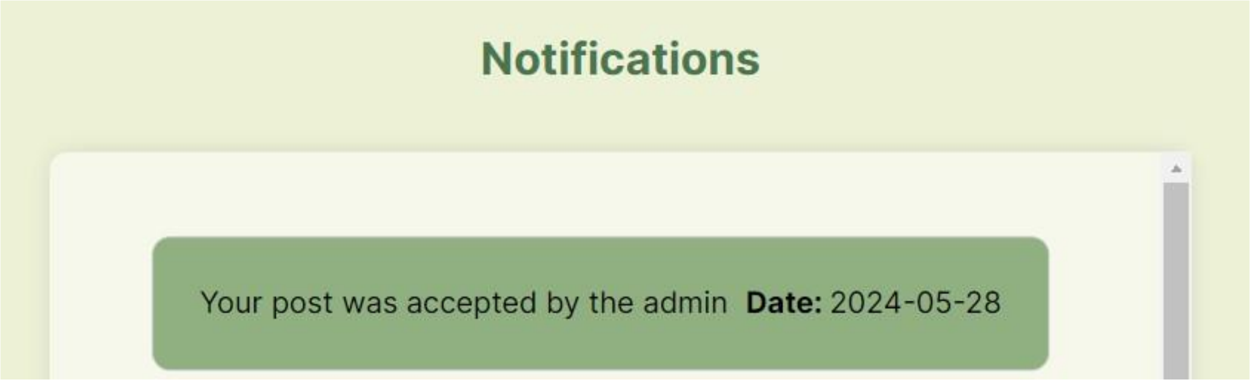
hello

User Name: tab **Date:** 2024-05-28

Requested Help:

DEMO EXAMPLE

Fig: Posts waiting for moderation at the admin end

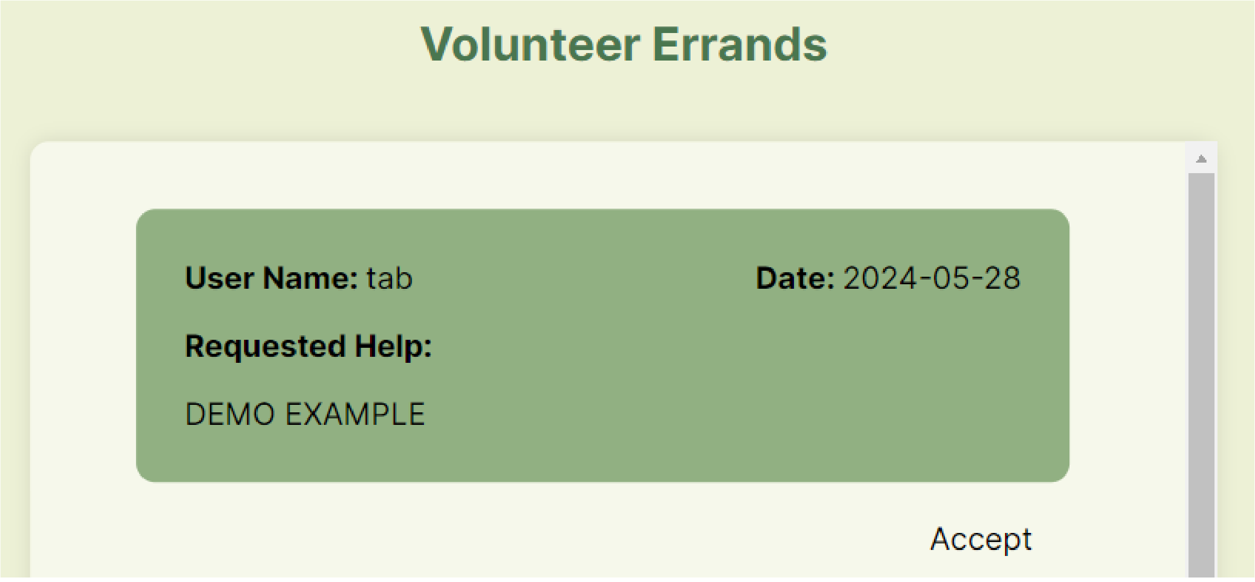

Notifications

Your post was accepted by the admin **Date:** 2024-05-28

Fig: Notification for the post being approved by the admin

Volunteer Task Page (user side):

The user who made the post won’t see his/her post in the “Volunteer Task” page. Only other users will be able to see it and accept it. Clicking on the accept button will make the user accept the victims call for help and start a dynamic routing system which can help him navigate his way to the victim.



Volunteer Errands

User Name: tab

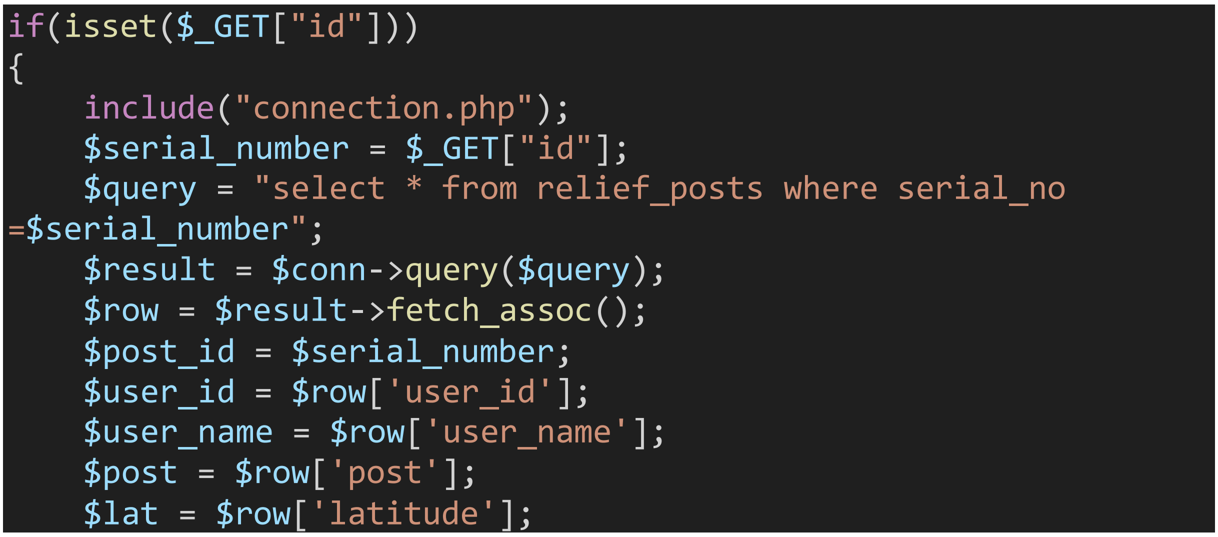
Date: 2024-05-28

Requested Help:

DEMO EXAMPLE

Fig: approved post displaying at other user’s end

The dynamic routing mechanism has been done by using leaflet library, PHP, JS and MySQL database. At first PHP collects the latitude and longitude co-ordinates of the volunteer from the database and sends it to the JS code written within script tag. The JS script then finds the value of the longitude and latitude value of the victim using leaflet library and GPS functionality of the device.



```
if(isset($_GET["id"]))
{
    include("connection.php");
    $serial_number = $_GET["id"];
    $query = "select * from relief_posts where serial_no
=$serial_number";
    $result = $conn->query($query);
    $row = $result->fetch_assoc();
    $post_id = $serial_number;
    $user_id = $row['user_id'];
    $user_name = $row['user_name'];
    $post = $row['post'];
    $lat = $row['latitude'];
    $lon = $row['longitude'];
    $date = $row['date'];
    $query = "insert into accepted_posts (serial_no, post_id, 
user_id, user_name, post, latitude, longitude, date, completed) 
values 
('','$post_id','$user_id','$user_name','$post','$lat','$lon','$
date','NO')";
    $result = $conn->query($query);
    $post = "Your post was accepted by the admin";
    $query = "insert into notification_data (serial_no, 
user_id, user_name, notification) values 
('','$user_id','$user_name','$post')";
    $result = $conn->query($query);
    $query = "delete from relief_posts where serial_no 
=$serial_number";
    $conn->query($query);
    header("Location:admin_pending_Posts.php");
}
```

```
<script>
        var lat2=<?php echo $lat; ?>;
        var lon2=<?php echo $lon; ?>;
```

It then plots two points using the two sets of latitude and longitude values (one set is of the volunteer’s and the other one is of the victim’s) on the openStreetMap. After that using a bult-in routing algorithm connects the two points on the map and highlights it to mark the shortest and most favorable path.

```
<script>
        var lat2=<?php echo $lat; ?>;
        var lon2=<?php echo $lon; ?>;
        var lat;
        var lon;

        if(!navigator.geolocation){
            console.log("Your browser doesnt support geolocaton 
feature!")
        } else {
            navigator.geolocation.getCurrentPosition(getpositio
n)
        }

        function getposition(position){
        console.log(position);
        lat = position.coords.latitude;
        lon = position.coords.longitude
        L.Routing.control({
            waypoints: [
                L.latLng(lat2, lon2),
                L.latLng(lat, lon)
                ],
                addWaypoints: false,
                draggableWaypoints: false
            }).addTo(map);
        }

        var map = L.map('map').setView([23.840058810555206,
90.3575901199828], 12);
        mapLink = "<a
href='http://openstreetmap.org'>OpenStreetMap</a>";
        L.tileLayer('http://{s}.tile.osm.org/{z}/{x}/{y}.png',
{ attribution: 'Leaflet &copy; ' + mapLink + ', contribution',
maxZoom: 25 }).addTo(map);
    </script>
```

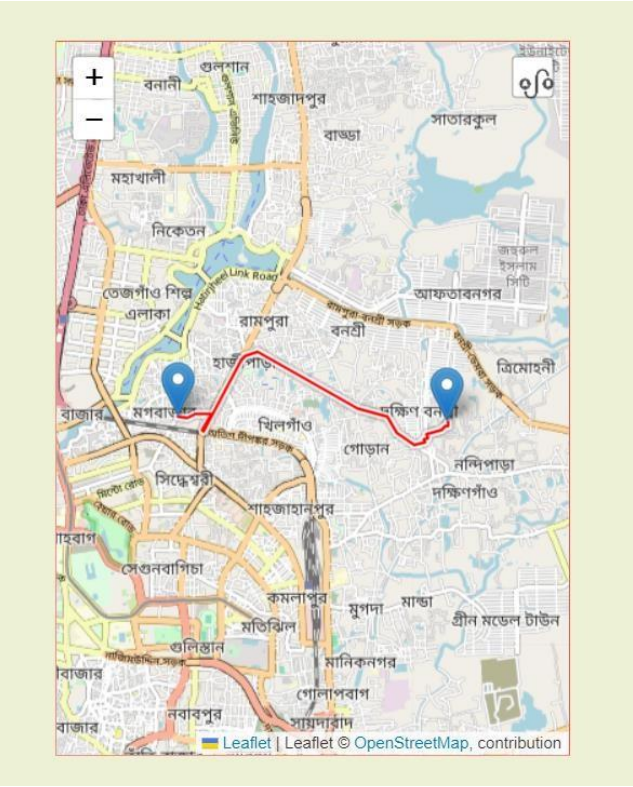


Fig: Dynamic Routing System

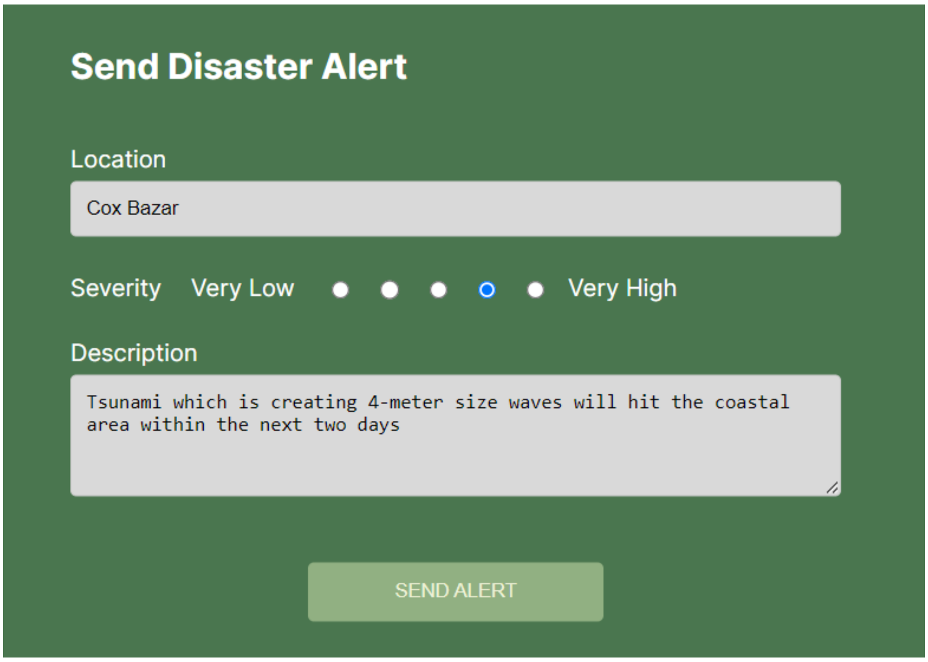
Another important functionality was carried out in the “Disaster Alert” page. The main aim of this page was to send alert notification to users on upcoming natural disaster by the admin. The page has three input sections. Of which two are text input fields, using which the admin can enter the location data and the description of the natural disaster. Other input area is the radio input area, which takes input of the severity status. This is happening fully in a php file, which takes the input coming from the form page as “post” method and sending them to the database. From the database the data is then displayed on every users “Alerts & Updates” page in form of a notification.

```
<div class="alert-container">
<h2>Send Disaster Alert</h2><br>
<form action="posting_alerts.php" method="post">
<label for="location">Location</label>
<input type="text" id="locations" name="location">
<div class="severity">
<label for="severity">Severity</label>
<label>Very Low</label>
<input type="radio" name="severity" value="Less Severe">
<input type="radio" name="severity" value="Intense">
<input type="radio" name="severity" value="Alarming">
<input type="radio" name="severity" value="Severe">
<input type="radio" name="severity" value="Very Severe">
<label>Very High</label>
</div>
<label for="description">Description</label>
<textarea id="descriptions" name="description" rows="4"
cols="70"></textarea>
<button type="submit">SEND ALERT</button>
</form>
</div>
</div>
```

The input data posted through this form then gets processed in the “posting_alert.php” page. After that it gets sent to the database from there.

```
<?php
include('connection.php');
    $admin_id = -1;
    $admin_name = 'admin';
    $location = $_POST['location'];
    $severity = $_POST['severity'];
    $description = $_POST['description'];
    $post = "$severity disaster will occur at $location:
$description";
    echo $post;
    $querry = "insert into notification_data (serial_no, user_id, user_name, notification) values ('','$admin_id','$admin_name', '$post')";
    $result = $conn->query($querry);
    header("Location:admin_send_alert.php")
?>
```

In the above it can be seen how the separate data taken from the form page gets processed and combined into a single data. This data is then sent to “notification_data” named database.



Send Disaster Alert

Location

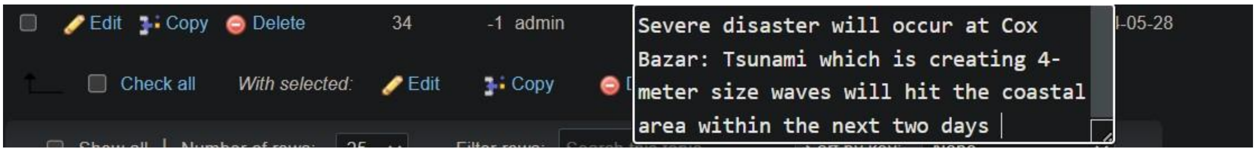
Cox Bazar

Severity Very Low ● ● ● ◉ ● Very High

Description

Tsunami which is creating 4-meter size waves will hit the coastal area within the next two days

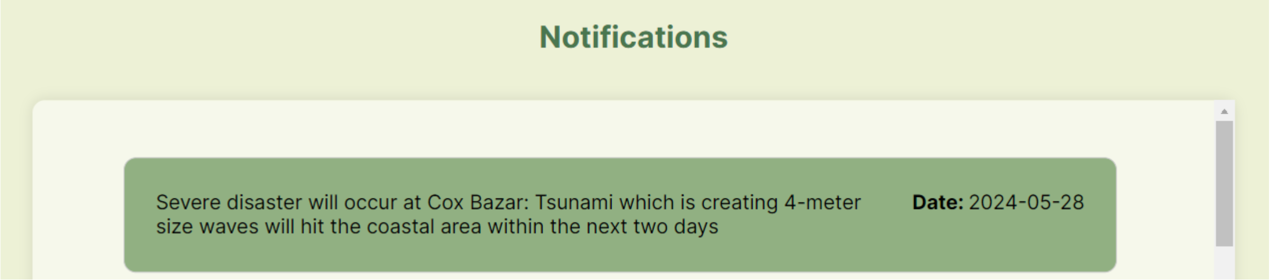
Fig: Alert message is being posted by the admin



□ Edit Copy Delete 34 -1 admin Severe disaster will occur at Cox Bazar: Tsunami which is creating 4-meter size waves will hit the coastal area within the next two days -05-28

□ Check all *With selected:* Edit Copy

Fig: Entry of the alert post in the database



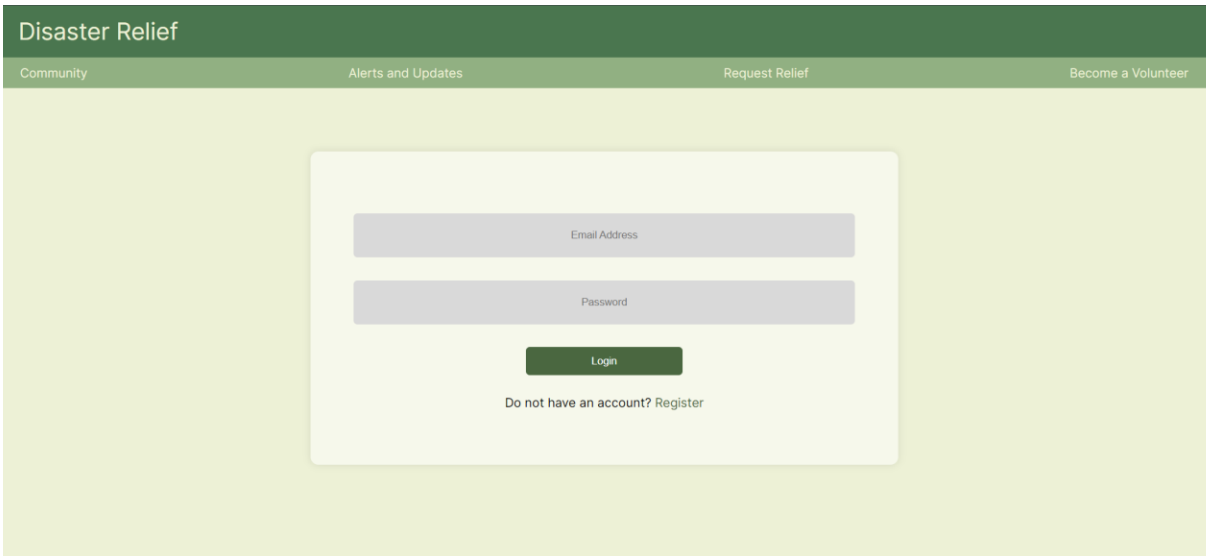
Notifications

Severe disaster will occur at Cox Bazar: Tsunami which is creating 4-meter size waves will hit the coastal area within the next two days **Date:** 2024-05-28

Fig: Notification being shown at the user end

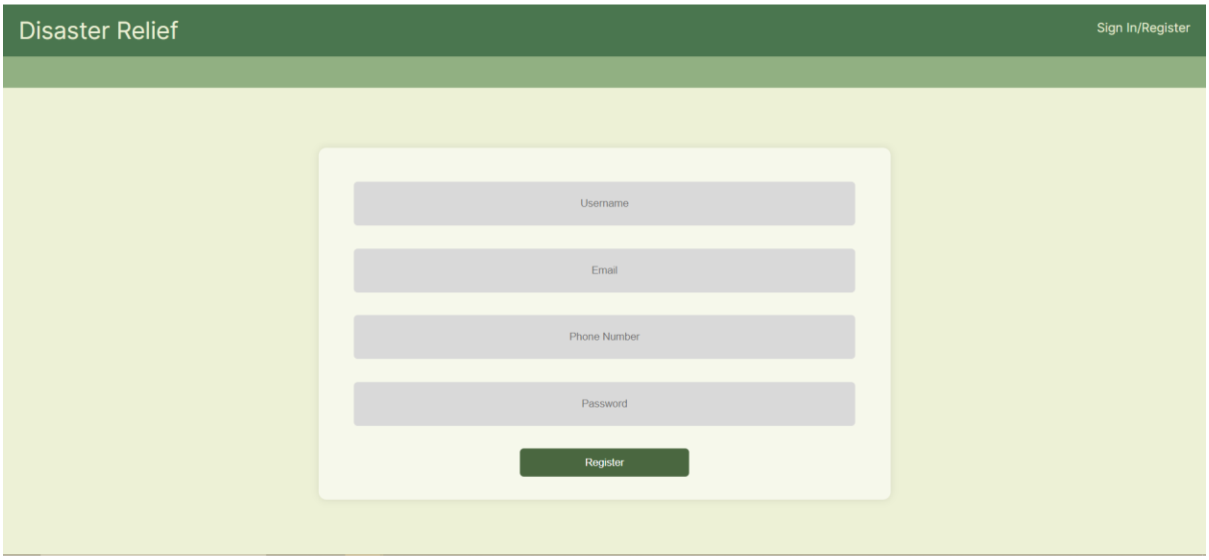
Result

User Side:

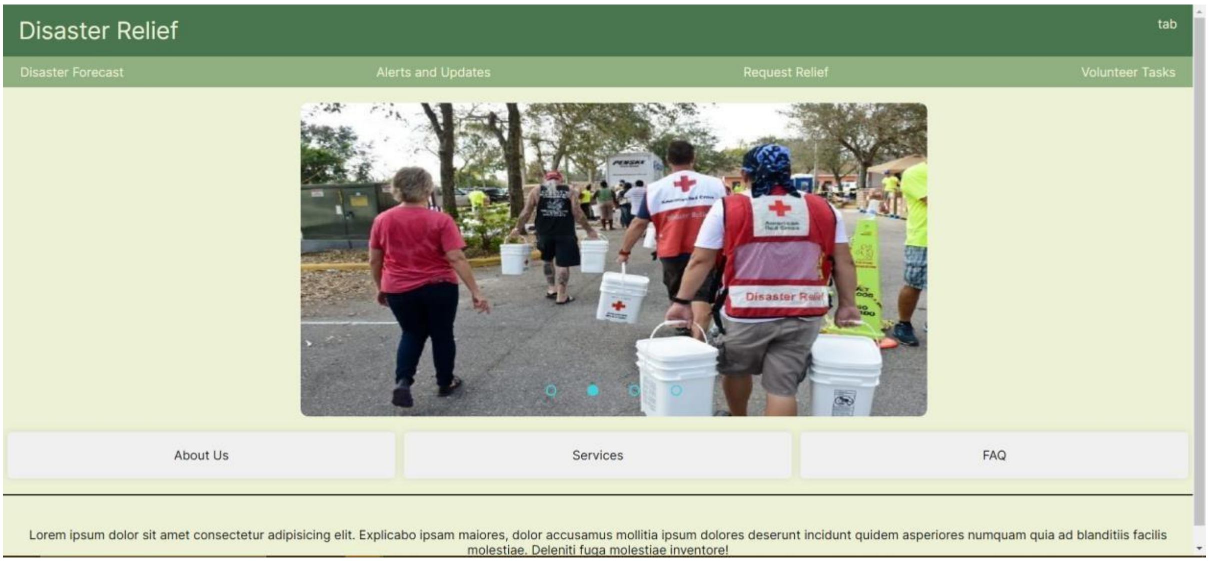


Do not have an account? Register

User Sign-In Page

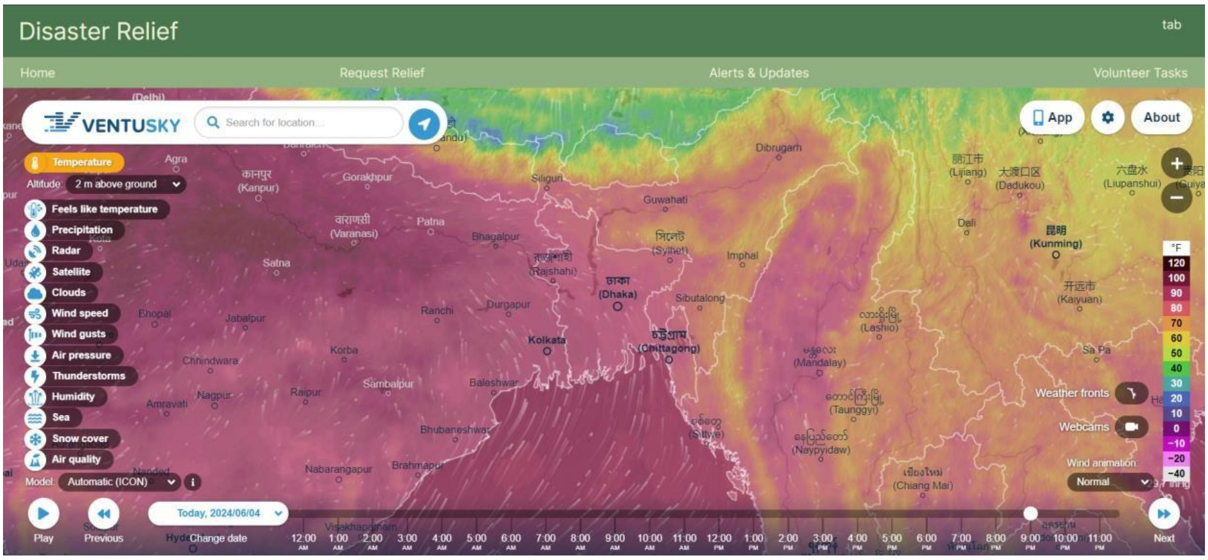


User Register Page

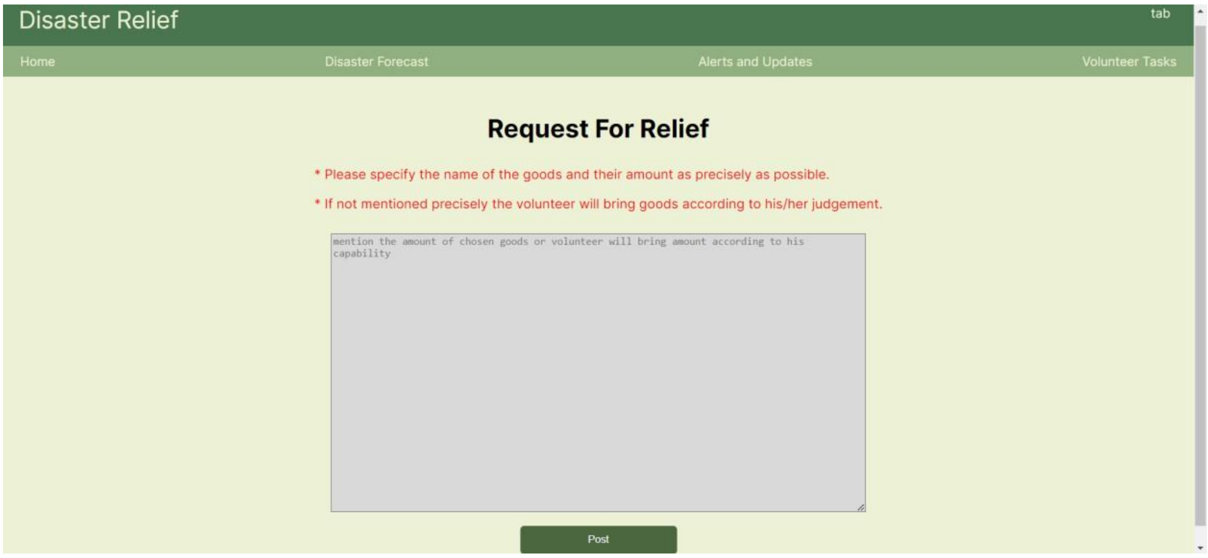


Lorem ipsum dolor sit amet consectetur adipisicing elit. Explicabo ipsam maiores, dolor accusamus mollitia ipsum dolores deserunt incidunt quidem asperiores numquam quia ad blanditiis facilis molestiae. Deleniti fuga molestiae inventore!

User Home Page



User Disaster Forecast Page



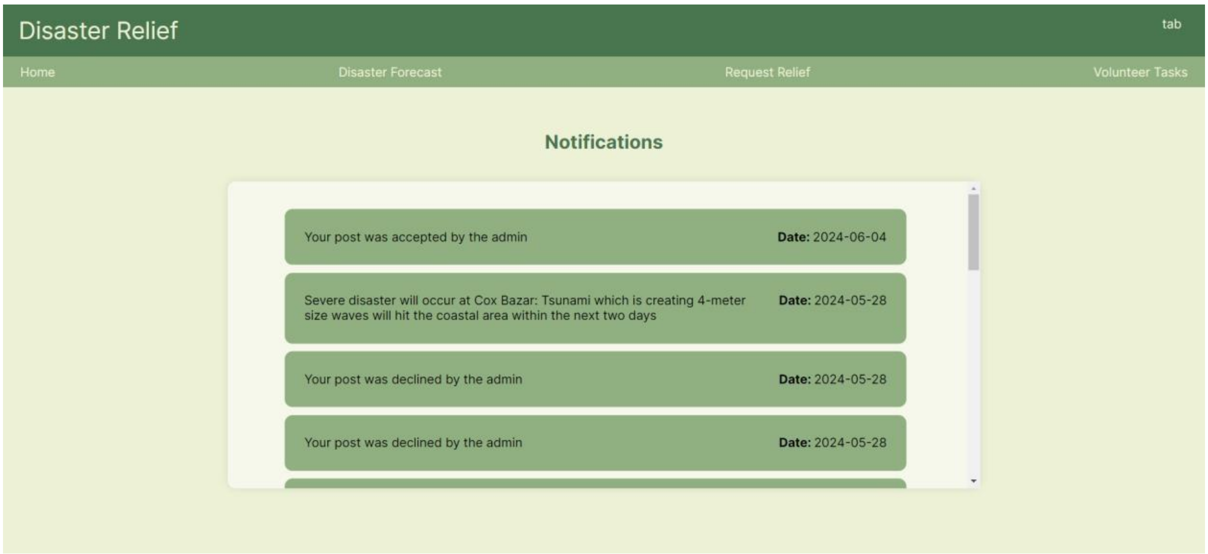
Request For Relief

* Please specify the name of the goods and their amount as precisely as possible.

* If not mentioned precisely the volunteer will bring goods according to his/her judgement.

mention the amount of chosen goods or volunteer will bring amount according to his capability

User Request Relief Page



Notifications

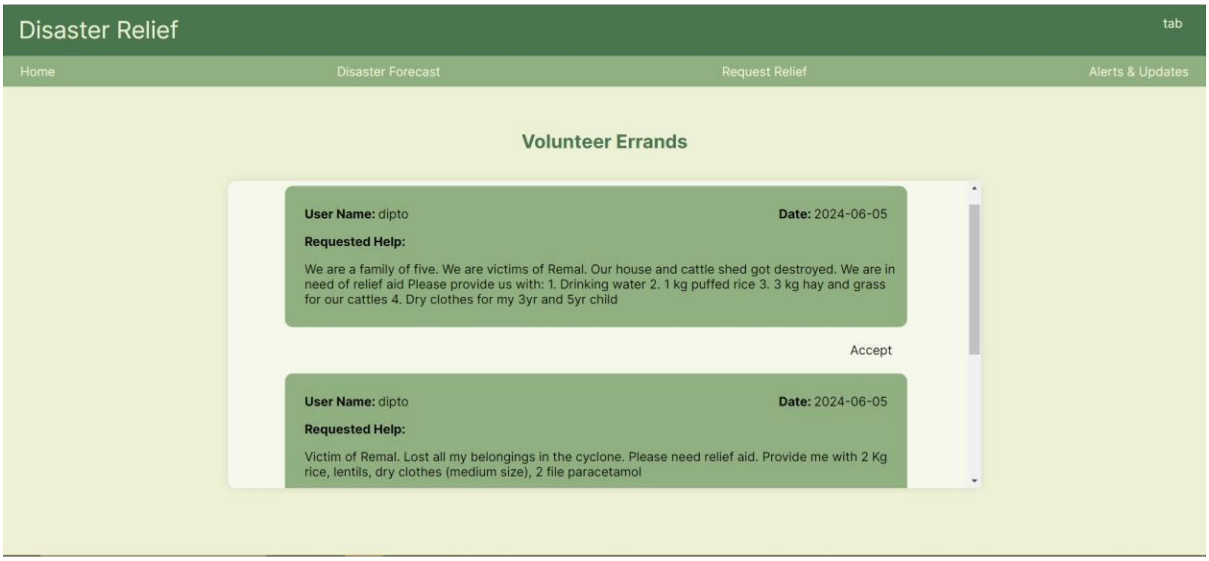
Your post was accepted by the admin **Date:** 2024-06-04

Severe disaster will occur at Cox Bazar: Tsunami which is creating 4-meter size waves will hit the coastal area within the next two days **Date:** 2024-05-28

Your post was declined by the admin **Date:** 2024-05-28

Your post was declined by the admin **Date:** 2024-05-28

User Alert & Updates Page



Volunteer Errands

User Name: dipto **Date:** 2024-06-05

Requested Help:

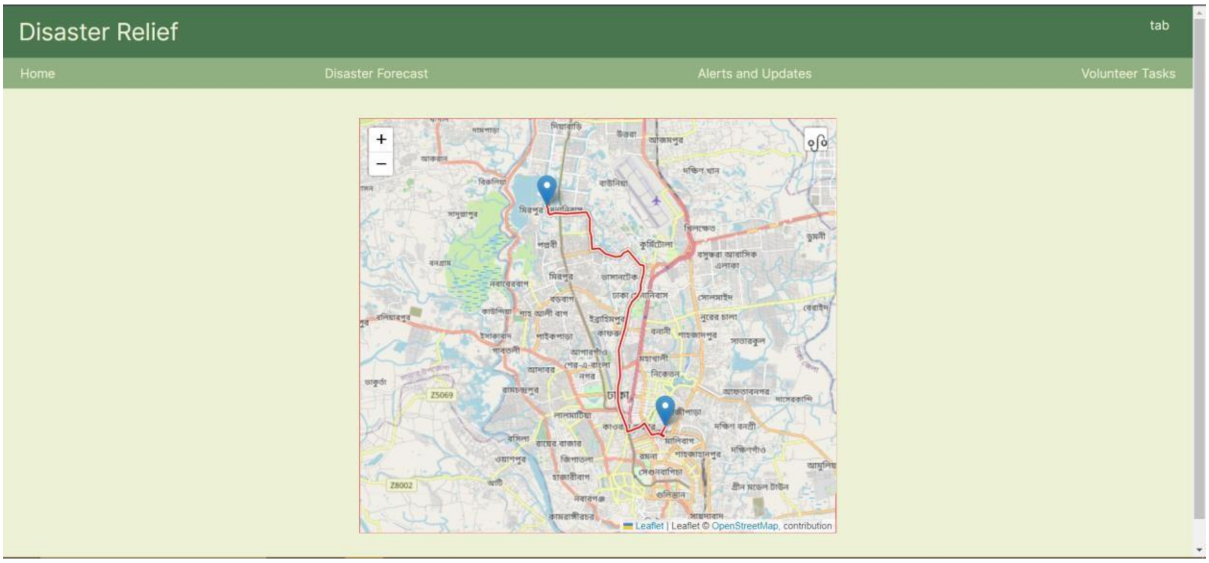
We are a family of five. We are victims of Remal. Our house and cattle shed got destroyed. We are in need of relief aid Please provide us with: 1. Drinking water 2. 1 kg puffed rice 3. 3 kg hay and grass for our cattles 4. Dry clothes for my 3yr and 5yr child

User Name: dipto **Date:** 2024-06-05

Requested Help:

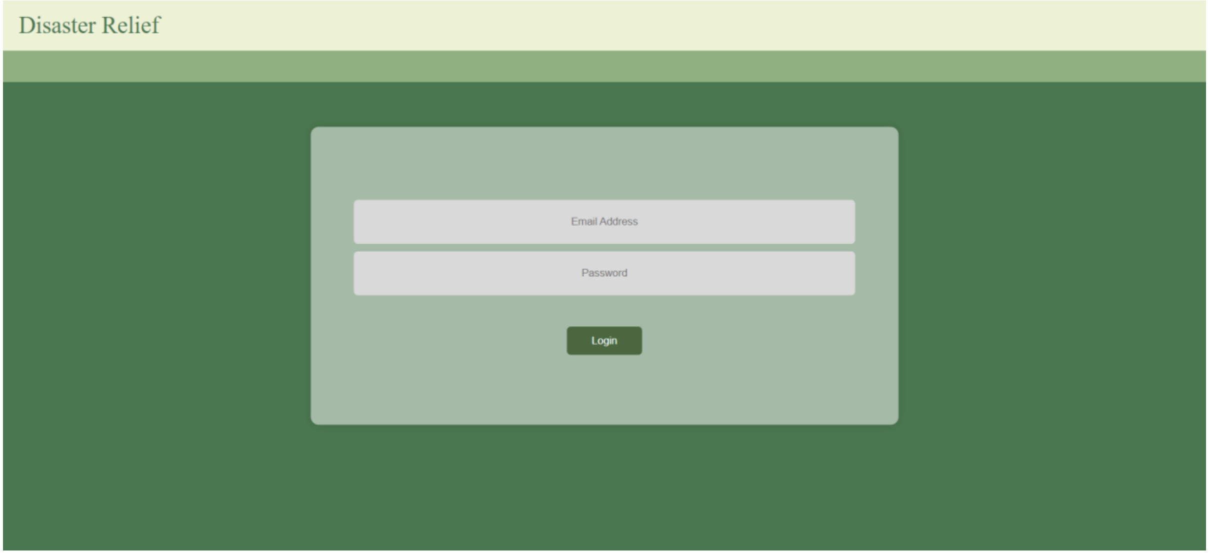
Victim of Remal. Lost all my belongings in the cyclone. Please need relief aid. Provide me with 2 Kg rice, lentils, dry clothes (medium size), 2 file paracetamol

User Volunteer Task Page

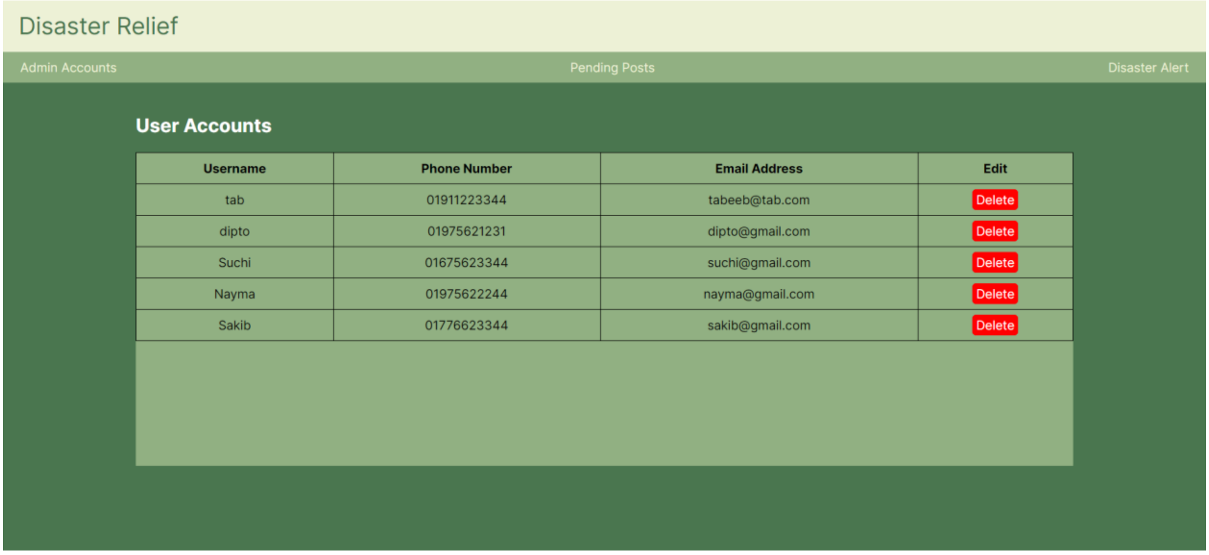


User Routing System Page

Admin Side:



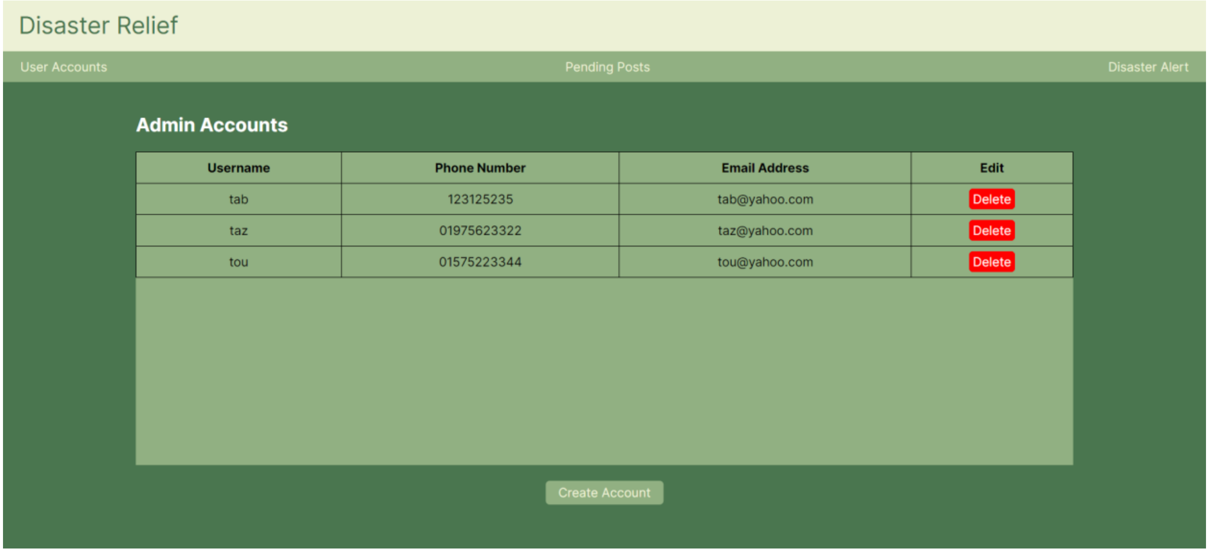
Admin Sign-in Page



User Accounts

Username	Phone Number	Email Address	Edit
tab	01911223344	tabeeb@tab.com	Delete
dipto	01975621231	dipto@gmail.com	Delete
Suchi	01675623344	suchi@gmail.com	Delete
Nayma	01975622244	nayma@gmail.com	Delete
Sakib	01776623344	sakib@gmail.com	Delete

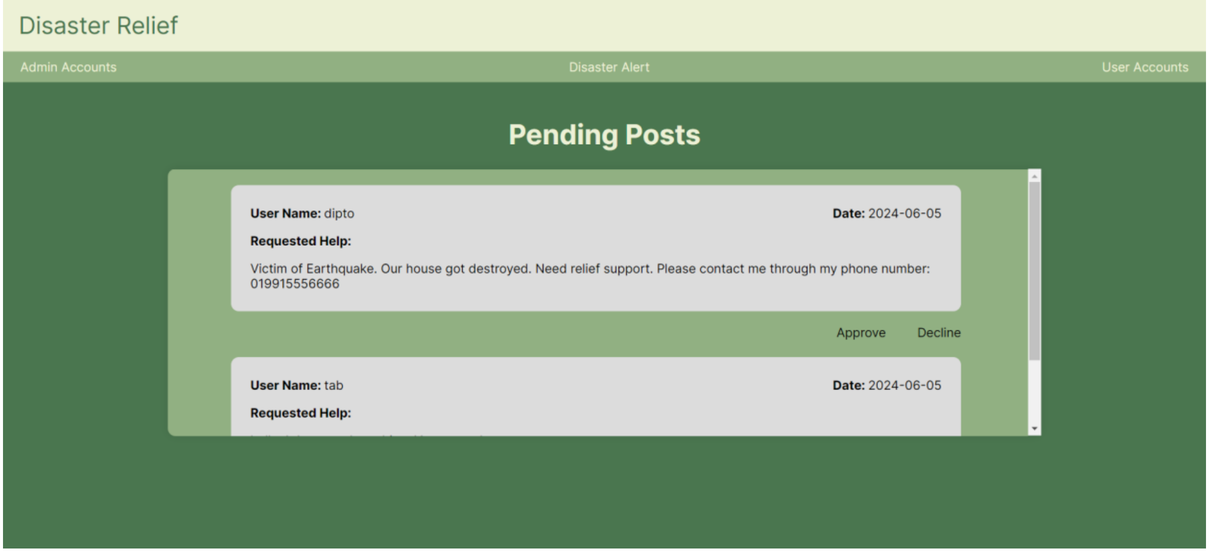
Admin User Account



Admin Accounts

Username	Phone Number	Email Address	Edit
tab	123125235	tab@yahoo.com	Delete
taz	01975623322	taz@yahoo.com	Delete
tou	01575223344	tou@yahoo.com	Delete

Admin Account Page



Pending Posts

User Name: dipto **Date:** 2024-06-05

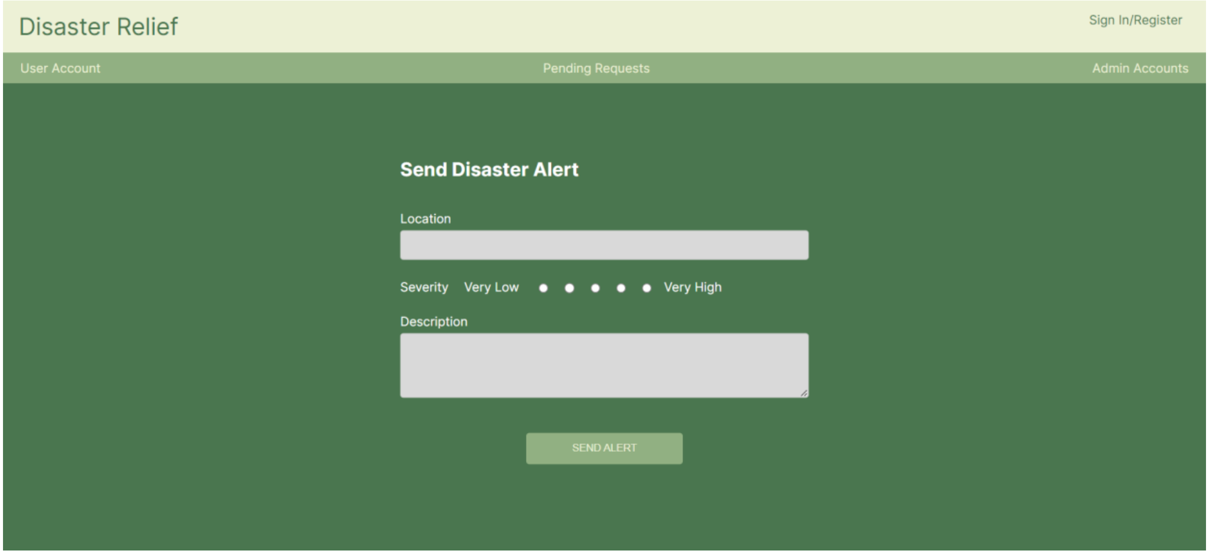
Requested Help:

Victim of Earthquake. Our house got destroyed. Need relief support. Please contact me through my phone number: 019915556666

User Name: tab **Date:** 2024-06-05

Requested Help:

Admin Pending Posts Page



Admin Disaster Alert Page

Conclusion

The Disaster Relief website is an initiative designed to bridge the gap between aid providers and disaster-affected individuals. By facilitating direct connections between natural disaster victims and volunteers, the project ensures that aid is delivered efficiently and effectively. Key features include the "Request Relief" section for victims to specify their needs, the "Volunteer Task" section for volunteers to view and respond to aid requests, and the "Disaster Forecast" page with a real-time interactive weather map. This project streamlines the aid delivery process, providing victims with the precise help they need in a timely manner and improving overall disaster response coordination.

Through this project, we learned the importance of clear communication and precise information in disaster relief efforts. We gained valuable insights into web development, including designing user-friendly interfaces and integrating real-time data updates. Overall, the Disaster Relief website has the potential to significantly impact disaster response efforts by enhancing the efficiency and accuracy of aid distribution, ultimately helping those in need more effectively.

Reference

[1] “Color Hunt - Color Palettes for Designers and Artists,” [Online] Available: https://colorhunt.co/

[2] “Figma: the Collaborative Interface Design tool.,” [Online] Available: https://www.figma.com/

[3] “Leaflet — an open-source JavaScript library for interactive maps,” [Online] Available: https://leafletjs.com/

[4] “Leaflet Routing Machine,” [Online] Available: https://www.liedman.net/leaflet-routing-machine/

[5] “Ventusky - Weather Forecast Maps,” [Online] Available: https://www.ventusky.com/