

A Project Report On

DISASTER MANAGEMENT SYSTEM



PAPER CODE: - ESC 591

Prepared By: -

- SUBHADEEP MUKHERJEE [12000121104]
- SAYANTANI CHATTERJEE [12000121122]
- MUKULIKA CHATTERJEE [12000122134]
- DEBRIK CHAKRABORTY [12000122136]
- SOUVIK CHOWDHURY [12000122140]

Dr B.C.ROY ENGINEERING COLLEGE, DURGAPUR

**DEPARTMENT OF COMPUTER SCIENCE AND
ENGINEERING**

Guided By:

PROF. BISWAJIT SAHA

PROF. MONALISA CHAKRABORTY

ACKNOWLEDGEMENT

We want to convey our heartfelt appreciation and gratitude to our professors, Prof. Biswajit Saha and Prof. Monalisa Chakraborty for providing us with the chance to work on the Software Engineering Project “DISASTER MANAGEMENT SYSTEM”, which also aided us in conducting extensive study and learning about a bunch of new topics.

Secondly, we want to thank our friends and parents for their assistance in completing this presentation in such a short period of time. It was quite useful in terms of expanding our knowledge and abilities.

DECLARATION

We as a team pledge to prioritize safety, empower through knowledge, leverage technology, promote collaboration, continuously improve, ensure accessibility, and forge partnerships in our commitment to enhancing disaster preparedness and resilience through the Disaster Management System project.

Signed,

1. Subhadeep Mukherjee
2. Sayantani Chatterjee
3. Mukulika Chatterjee
4. Debrik Chakraborty
5. Souvik Chowdhury

30th November 2023

CONTENT

- Introduction
- Project Objectives
- Key Features
- Requirements Analysis
- Hardware and Software Requirements
- Data Flow Diagram
- Project Modules
- Future Scope
- Conclusion
- Reference

INTRODUCTION

CalamityCare is a website dedicated to disaster management and preparedness. Our core objectives revolve around the provision of timely alerts pertaining to an array of potential calamities, the facilitation of indispensable resources and expert guidance, the cultivation of community involvement and collaboration, the dissemination of knowledge derived from data-driven analyses, and the advocacy for enhanced education and training in disaster-related fields. Through the execution of these multifarious initiatives, CalamityCare aspires to bestow individuals and communities with the tools, knowledge, and resources necessary to fortify their preparedness and fortitude when confronted with unforeseen emergencies and crises.

PROJECT OBJECTIVES

- **Resource and Information Hub:** Establish an extensive online resource centre that offers comprehensive disaster preparedness plans, emergency kits, and practical guidance for individuals, communities, and organizations
- **Community Engagement:** Foster community collaboration and engagement by creating forums, local disaster response groups, and volunteer opportunities.
- **Global Reach:** Expand the reach of CalamityCare to serve communities and regions worldwide, irrespective of their level of technological development.
- **Alert-based System:** Developing an alert-based system for the disaster management website to spot unusual data patterns on the local host. When potential disaster signs arise, instantly send alerts within the platform. The main aim is to promptly notify users and stakeholders, enabling quick responses to emerging risks during disasters.

KEY FEATURES

- **Warning Alerts:** Providing real-time warnings and alerts for various disasters, enabling timely response.
- **Resource Library:** Offering a comprehensive repository of disaster preparedness plans, emergency kits, and guidelines.
- **Community Forums:** Fostering collaboration and information sharing among users in local and global communities.
- **Educational Resources:** Offering online courses, tutorials, and materials on disaster management and response.
- **Global Accessibility:** Ensuring inclusivity and accessibility for users worldwide, regardless of technological resources.
- **Continuous Updates:** Regularly updating the platform with the latest technology and disaster management best practices.
- **Collaborative Partnerships and Funding:** Working with government agencies, NGOs, and academic institutions to maximize impact.

REQUIREMENT ANALYSIS

Functional Requirements:

- User registration and authentication.
- Incident reporting.
- Resource allocation and management.
- Communication and notification tools.
- Reporting and analytics.
- Geospatial capabilities.
- Integration with external data sources (e.g., weather data, government alerts).

Non-Functional Requirements:

- Performance, scalability, security, and usability.
- Response times for critical operations.
- Outline security measures for data protection and user privacy.
- Address accessibility and compliance with relevant standards.

HARDWARE AND SOFTWARE REQUIREMENTS

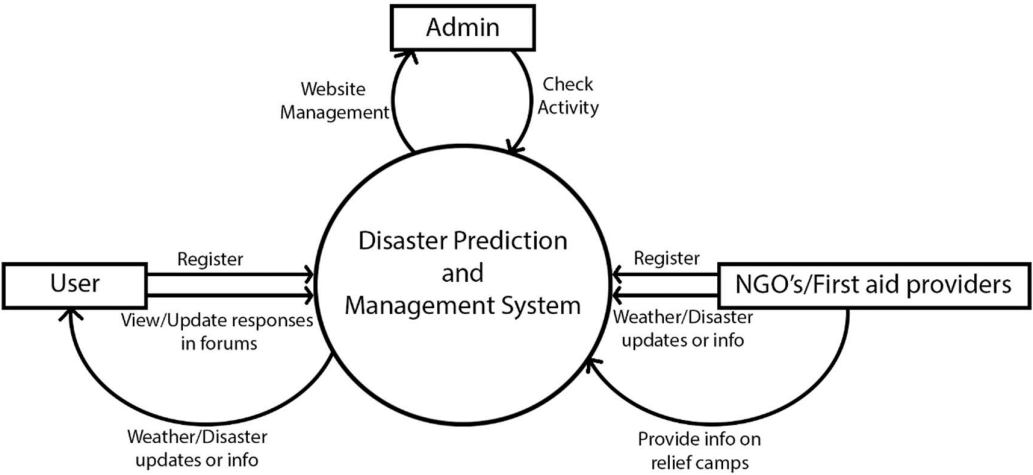
HARDWARE: -

- Processor: Intel Core i3 or equivalent
- RAM: 4 GB (for basic functionality)
- Storage: Minimum 128 GB SSD/HDD
- Network Infrastructure: Reliable internet connectivity
- End-User Devices: Compatible devices like laptops, desktops, or mobile devices based on system requirements

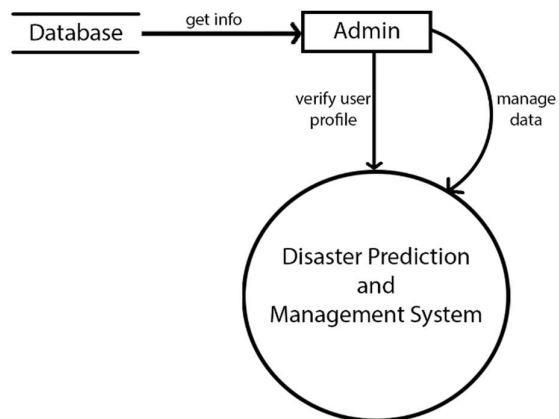
SOFTWARE: -

- Operating System: Windows 10 or later, macOS, or a Linux distribution
- Database Management System (DBMS): MySQL, PostgreSQL, SQLite, or similar (depends on project needs)
- Web and Mobile Applications: Compatible web browsers (Chrome, Firefox, Safari, etc.), and mobile OS compatibility (iOS, Android)
- API (Open Weather)
- Communication and Collaboration Tools: Email
- Python

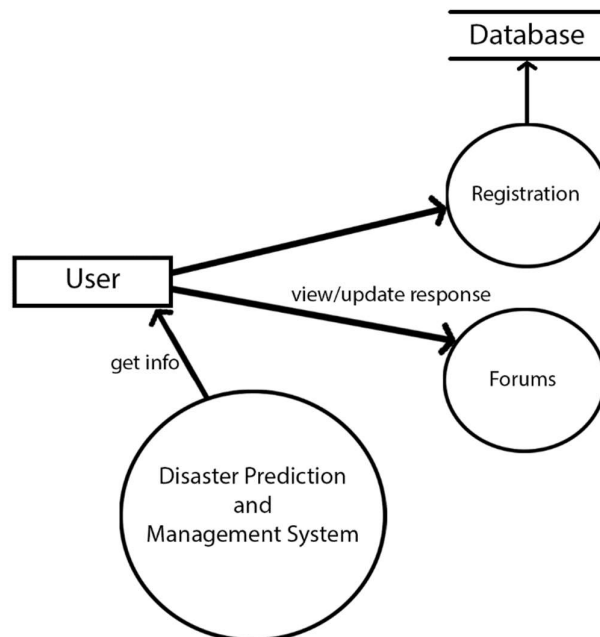
DATA FLOW DIAGRAM



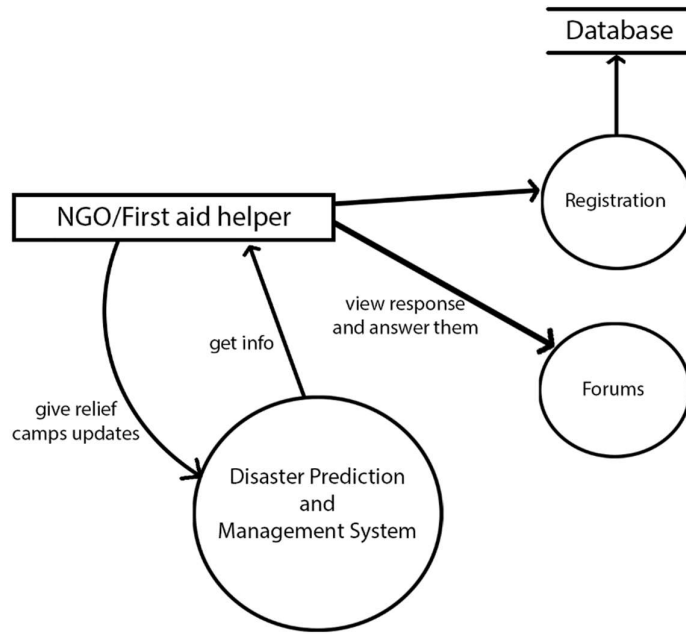
0-Level DFD for Disaster Management System



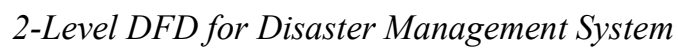
1-Level DFD for Disaster Management System for Admin



1-Level DFD for Disaster Management System for User



I-Level DFD for Disaster Management System for NGO's



2-Level DFD for Disaster Management System

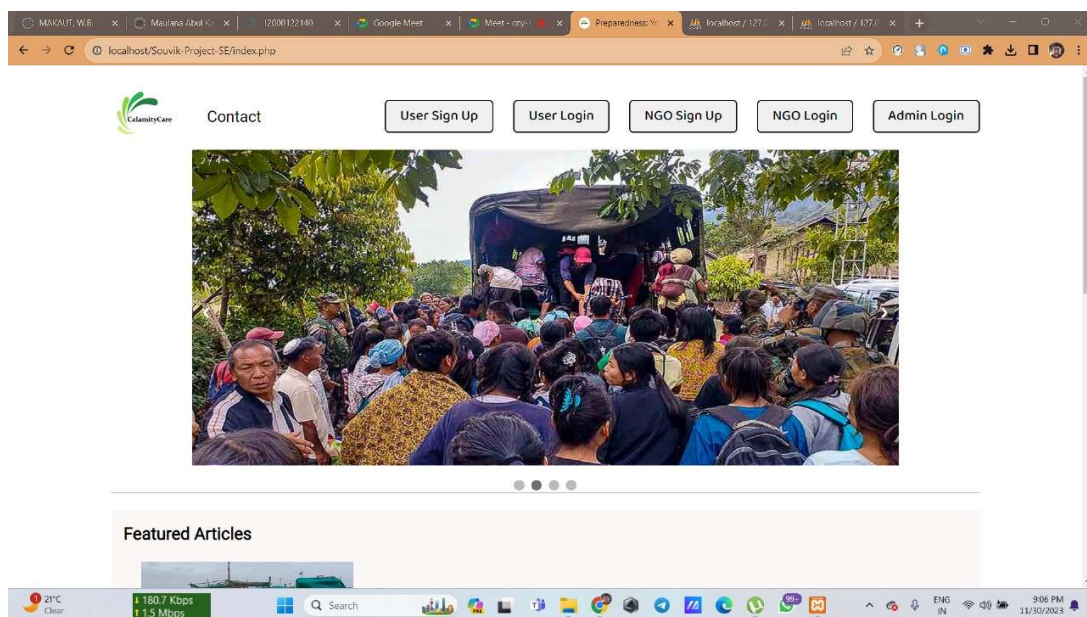
PROJECT MODULES

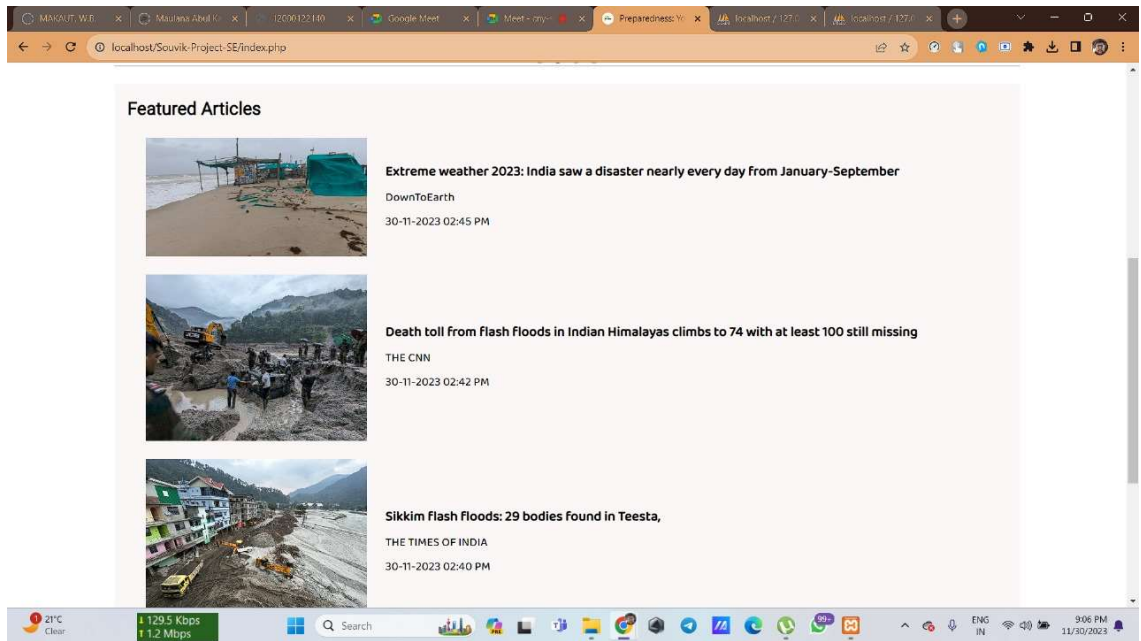
1. HOME PAGE: -

Code: -

```
File Edit Selection View Go Run ... Search
index.html x
E:\SE (2) > SE (1) > SE > ResponsiveBlogTemplate > index3.html > html > head > style > .slideshow-container
1 <!DOCTYPE html>
2 <html lang="en">
3
4 <head>
5
6 <link rel="icon" href="img/icon.png">
7 <meta charset="UTF-8">
8 <meta name="viewport" content="width=device-width, initial-scale=1.0">
9 <link rel="stylesheet" href="css/utlis.css">
10 <link rel="stylesheet" href="css/style.css">
11 <link rel="stylesheet" href="css/mobile.css">
12 <title>Preparedness: Your shield against disasters!</title>
13
14 <style>
15
16 * {box-sizing: border-box}
17 body {font-family: Verdana, sans-serif; margin:0}
18 .mySlides {display: none}
19 img {vertical-align: middle;}
20
21 /* Slideshow container */
22 .slideshow-container {
23   max-width: 1000px;
24   position: relative;
25   margin: auto;
26 }
27
28 /* Next & previous buttons */
29 .prev, .next {
30   cursor: pointer;
31   position: absolute;
32   top: 50%;
```

Output: -





2. FORUMS PAGE

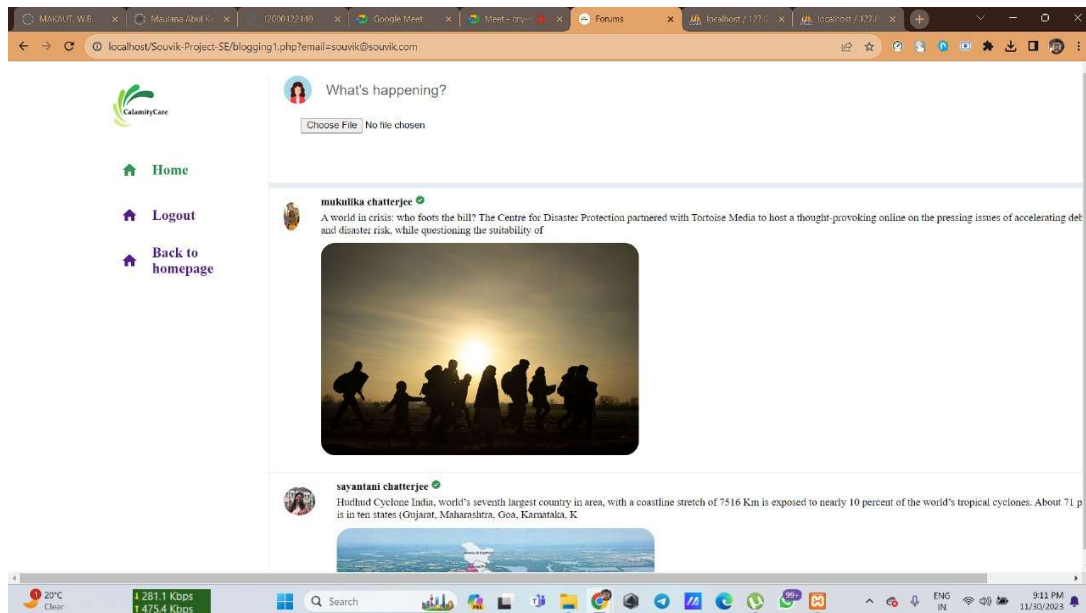
Code: -

```

File Edit Selection View Go Run ...
index3.html
E:\> SE (2) > SE (1) > SE > ResponsiveBlogTemplate > <> index3.html > ...
1  <!DOCTYPE html>
2  <html lang="en">
3
4  <head>
5
6  <link rel="icon" href="img/icon.png">
7  <meta charset="UTF-8">
8  <meta name="viewport" content="width=device-width, initial-scale=1.0">
9  <link rel="stylesheet" href="css/utills.css">
10 <link rel="stylesheet" href="css/style.css">
11 <link rel="stylesheet" href="css/mobile.css">
12 <title>Preparedness: Your shield against disasters!</title>
13
14 <meta name="viewport" content="width=device-width, initial-scale=1">
15 <style>
16 * {box-sizing: border-box}
17 body {font-family: Verdana, sans-serif; margin:0}
18 .mySlides {display: none}
19 img {vertical-align: middle;}
20
21 /* Slideshow container */
22 .slideshow-container {
23   max-width: 1000px;
24   position: relative;
25   margin: auto;
26 }
27
28 /* Next & previous buttons */
29 .prev, .next {
30   cursor: pointer;
31   position: absolute;
32   top: 50%;

```

Output: -

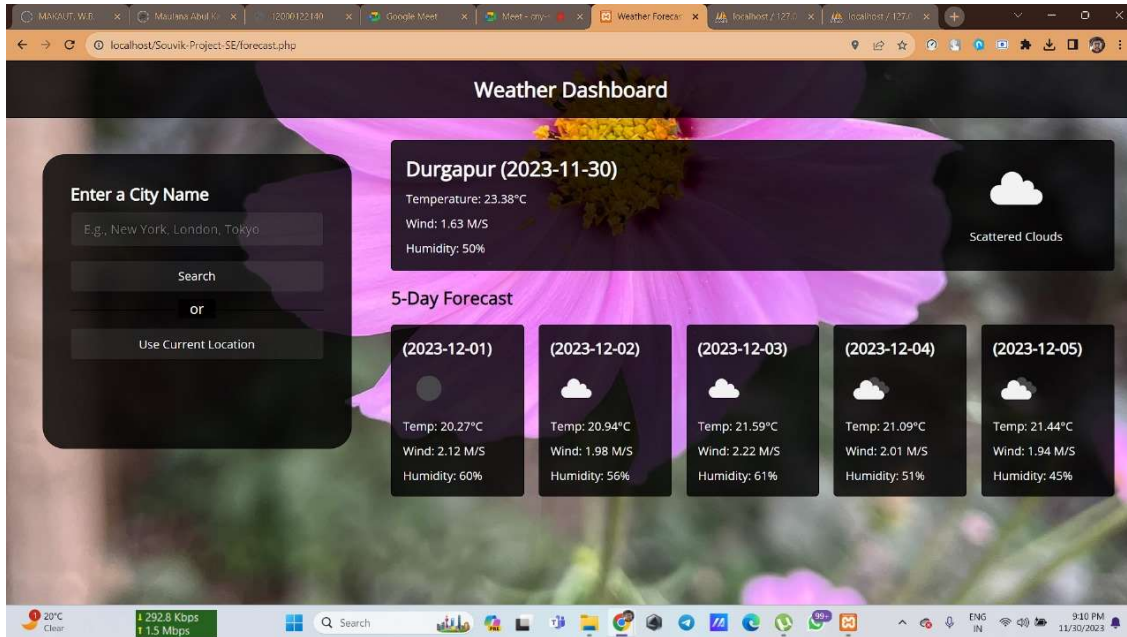


3. FORECAST PAGE

Code: -

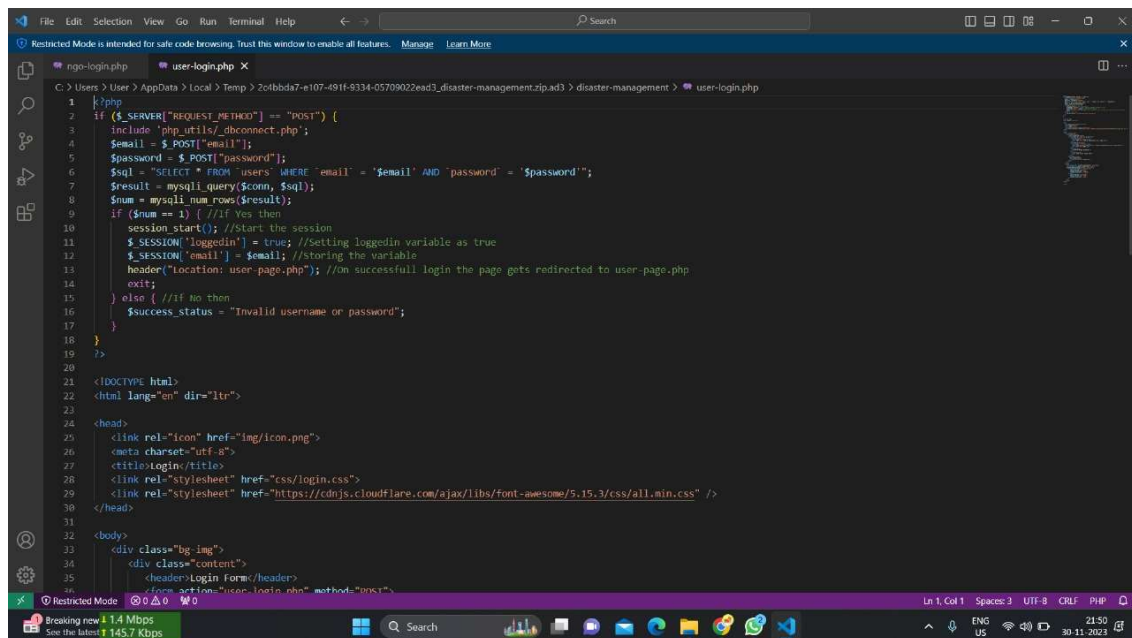
A screenshot of a Visual Studio Code editor window. The top bar shows standard Windows taskbar icons on the left, followed by VS Code menu items (File, Edit, Selection, View, Go, Run, Terminal, Help) and search icons. Below the menu is a search bar containing "Search". The main editor area has a tab titled "script.js" and contains JavaScript code for a web application. The code includes DOM selectors for input fields, API key configuration, HTML generation functions for weather cards and forecasts, and an AJAX call to fetch weather details from OpenWeatherMap. The bottom status bar displays system information like temperature (22°C), network speed (413 Mbps), storage usage (193.9 Kbps), language (English US), and date/time (30-11-2023, 21:36).

Output: -

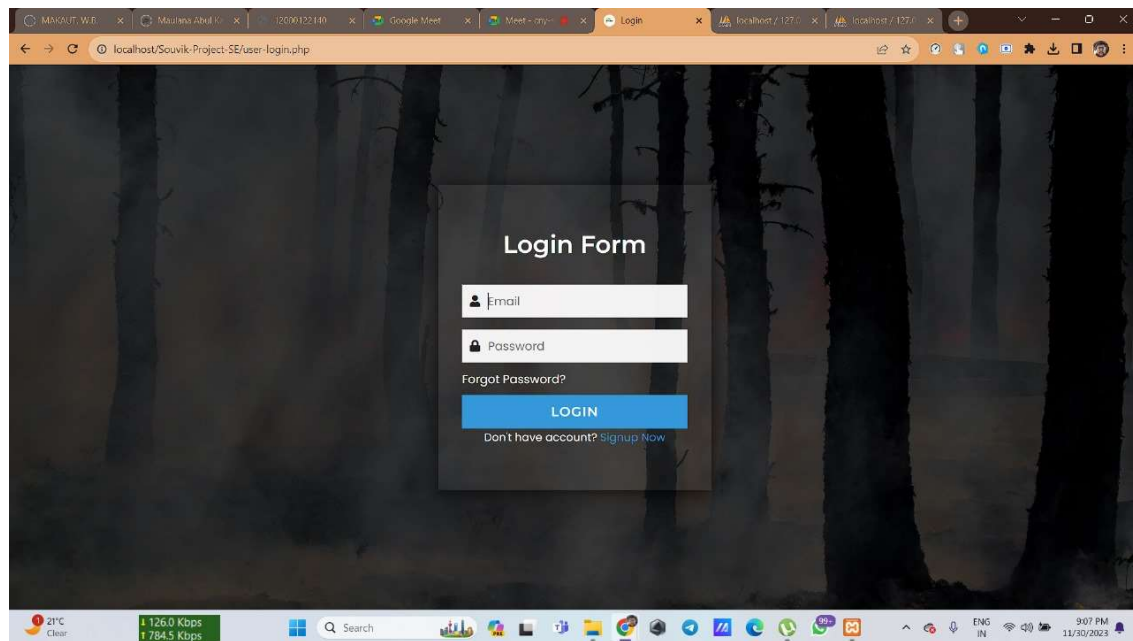


4. LOGIN PAGE

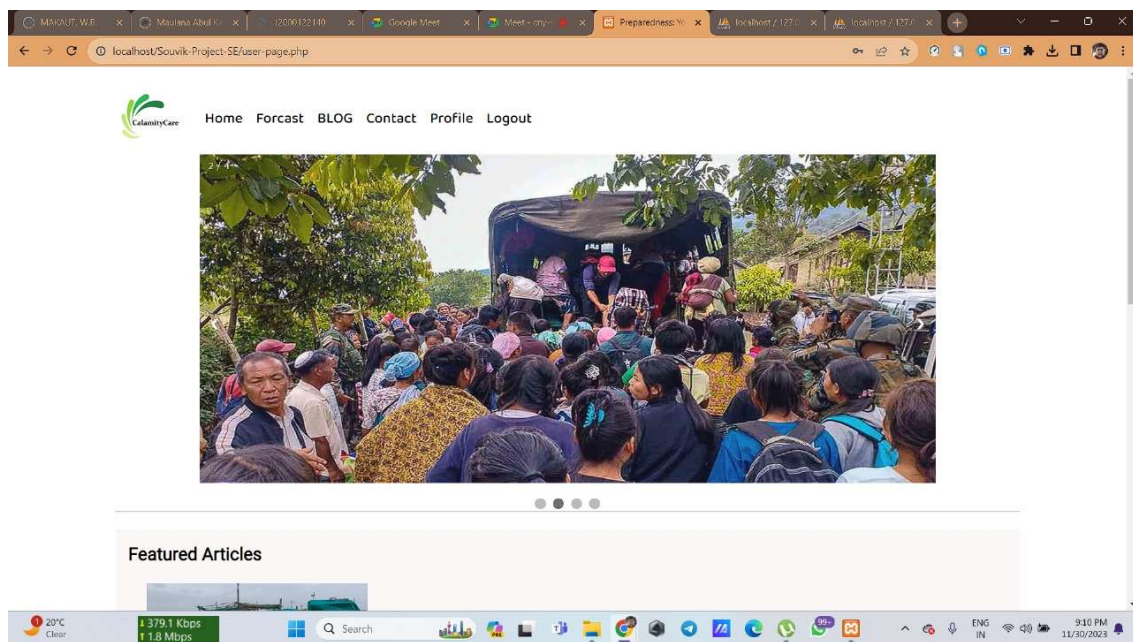
Code: -



Output: -



Home page after user login-



5. SIGNUP PAGE

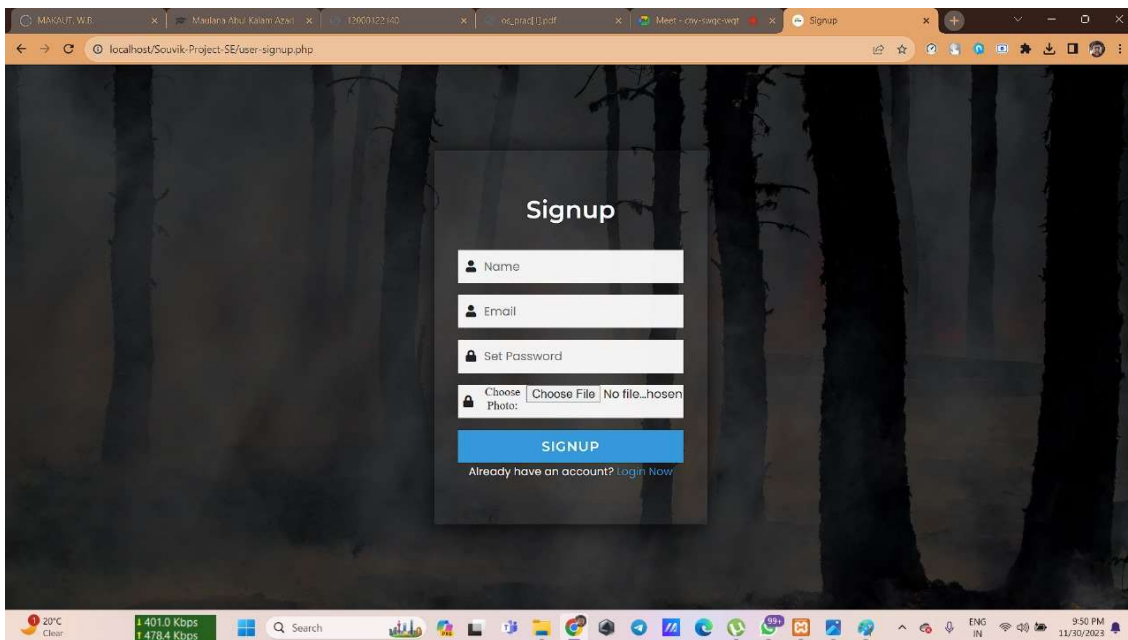
Code: -

```
File Edit Selection View Go Run Terminal Help
Restricted Mode is intended for safe code browsing. Trust this window to enable all features. Manage Learn More

C:\Users\User > User > AppData > Local > Temp > b6c7e014-e255-4e21-af66-63196d01eb4c_disaster-management.zip.b4c > disaster-management > user-signup.php

1 <?php
2 if ($_SERVER["REQUEST_METHOD"] == "POST") {
3     include 'php_utils/_dbconnect.php';
4
5     if (!$conn) {
6         die("Connection failed: " . mysqli_connect_error());
7     }
8
9     $name = $_POST['name'];
10    $email = $_POST['email'];
11    $password = $_POST['password'];
12
13    $photo_path = 'users/' . basename($_FILES['photo']['name']);
14
15    // check whether the email exists in the database table
16    $sql_checkusername = "SELECT * FROM users WHERE email = '$email'";
17    $result = mysqli_query($conn, $sql_checkusername);
18    $numExistsRows = mysqli_num_rows($result);
19
20    if ($numExistsRows > 0) {
21        $duplicate_username_error = "The Email already exists!";
22    } else {
23        $sql_insertdetails = "INSERT INTO users ('name', 'email', 'password', 'photo') VALUES ('$name', '$email', '$password', '$photo_path')";
24        $result_insertdetails = mysqli_query($conn, $sql_insertdetails);
25
26        if ($result_insertdetails && move_uploaded_file($_FILES['photo']['tmp_name'], $photo_path)) {
27            $insertdetails_result = "Details entered successfully";
28            header('location: login.php');
29            exit();
30        } else {
31            $insertDetails_error = "Errors: " . mysqli_connect_errno();
32        }
33    }
34 }
35 ?>
```

Output: -



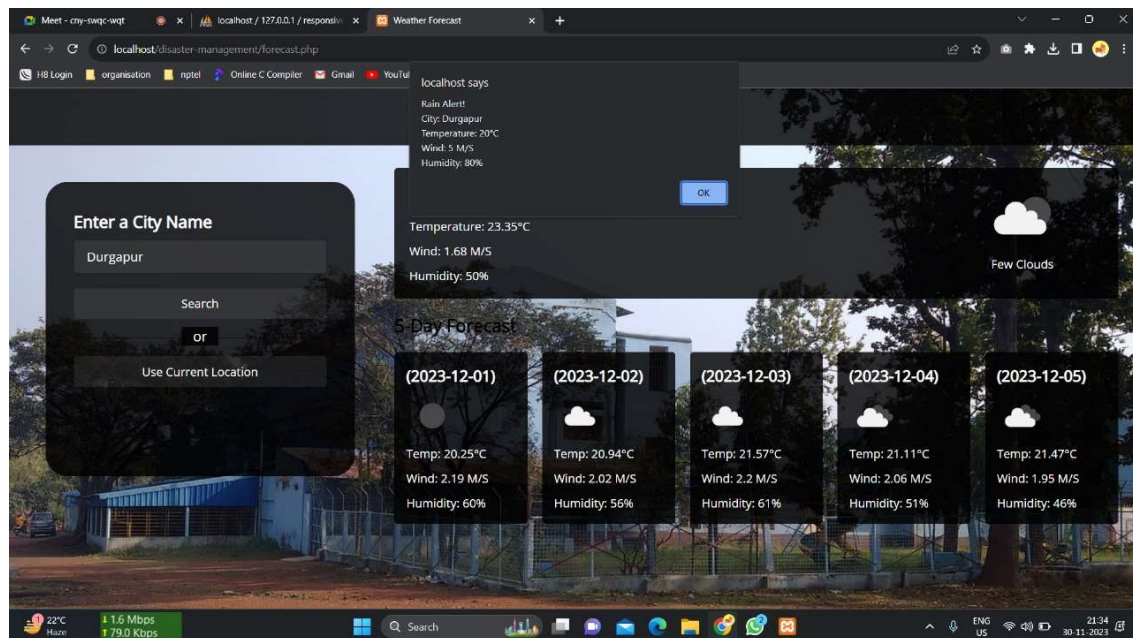
6. ALERT SYSTEM

Code: -

```
File Edit Selection View Go Run Terminal Help
Restricted Mode is intended for safe code browsing. Trust this window to enable all features. Manage Learn More

forecast.php
C:\Users> User > Desktop > disaster-management > forecast.php
14 <html lang="en">
15 <head>
16 <meta charset="utf-8">
17 <title>Weather Forecast</title>
18 <link rel="preconnect" href="https://fonts.gstatic.com">
19 <link href="https://fonts.googleapis.com/css2?family=Open+Sans&display=swap" rel="stylesheet">
20 <link rel="stylesheet" href="style3.css">
21 <meta name="viewport" content="width=device-width, initial-scale=1.0">
22 <script src="script3.js" defer></script>
23 <script>
24 function showRainAlert(city, temperature, wind, humidity) {
25     if (humidity > 70) {
26         alert("Rain Alert!\nCity: ${city}\nTemperature: ${temperature}°C\nWind: ${wind} M/S\nHumidity: ${humidity}%");
27     }
28 }
29
30 // This function will be called when the user clicks the "Search" button
31 function searchWeather() {
32     const cityInput = document.querySelector("#city-input");
33     const cityName = cityInput.value.trim();
34
35     // Assume these values for demonstration, replace with actual weather data
36     const temperature = 20;
37     const wind = 5;
38     const humidity = 80;
39
40     // Call the alert function with weather data
41     showRainAlert(cityName, temperature, wind, humidity);
42 }
43 </script>
44 </head>
45 <body>
46 <h1>Weather Dashboard</h1>
47 <div class="container">
48 <div class="weather-input">
49 <input type="text" value="Enter a City Name"/>
50 </div>
51 </div>
52 </body>
53 </html>
Ln 1, Col 1 Spaces: 2 UTF-8 CRLF PHP
```

Output: -



FUTURE SCOPE

The future scope of Calamity Care is to further expand its disaster management and relief efforts by leveraging advanced technology, data analytics, and global partnerships. This includes improving disaster preparedness, and scaling up rapid response initiatives to address an ever-increasing number of natural and man-made disasters worldwide.

CONCLUSION

In conclusion, Calamity Care stands as a beacon of hope and support in the face of adversity. Through its unwavering commitment to disaster management and relief, Calamity Care has demonstrated the power of resilience, compassion, and effective response.

REFERENCE

1. https://www.youtube.com/watch?v=9FD2ugeS4OU&t=997s&ab_channel=CodeWithHarry
2. <https://www.udemy.com/course/build-responsive-website-using-html5-css3-js-and-bootstrap-p/learn/lecture/24008850?start=15#overview>
3. <https://chat.openai.com/>