FLOOD MONITORING AND EARLING WARNING

TEAM MEMBER

911721104080 : ROSLIN SYLVIA.A

Phase 5 Submission Document

PROJECT **:** Flood Monitoring Analysis

FLOOD MONITORING SYSTEM

INTRODUCTION

Floods are among the most devastating natural disasters, causing significant loss of life and property. Timely and accurate monitoring of flood-prone areas is essential for effective disaster management. This paper presents a Flood Monitoring System utilizing the Internet of Things (IoT) technology to provide real-time data collection, analysis, and alerts. The system integrates various IoT sensors, including water level sensors, rainfall sensors, and weather stations, to continuously collect data from flood-prone regions. Data is transmitted to a central server via wireless communication, where it is processed and analyzed in real-time. The system employs data analytics and machine learning algorithms to predict potential flood events and issue alerts to relevant authorities and the public through various communication channels such as mobile apps, SMS, and web interfaces. This flood monitoring system enhances disaster preparedness and response by providing early warning and critical data for decision-makers. The practical implementation of this system demonstrates its effectiveness in reducing flood-related risks and improving community safety.

INNOVATION IN FLOOD MONITORING SYSTEM

Remote Sensing Technology:

The use of satellite imagery and remote sensing technologies can provide real-time data on rainfall, river levels, and flood extent. This data can help in monitoring and predicting flood events.

Internet of Things (IOT):

IOT devices can be deployed in flood-prone areas to monitor water levels, rainfall intensity, and soil moisture. These devices can transmit data in real-time to a central system for analysis and decision making.

Artificial Intelligence (AI) and Machine Learning:

By analyzing historical flood data and patterns, AI and machine learning algorithms can be trained to predict flood events more accurately. These algorithms can automatically analyze large volumes of data and provide early warning alerts.

Unmanned Aerial Vehicles (UAVs):

UAVs or drones equipped with cameras and sensors can monitor flood-prone areas and gather valuable data such as water levels, flood extent, and damage assessment. This data can help emergency responders make informed decisions during rescue and relief operations.

Mobile Applications:

Mobile applications can be used to crowdsource flood reports from the public. Users can submit real time information such as pictures, videos, and locations of flood events. This data can help authorities in monitoring and responding to floods more effectively.

Data Integration and Visualization:

Advanced data integration techniques can combine data from various sources such as weather stations, river gauges, and social media feeds. This integrated data can be visualized on a digital dashboard for better decision-making and situational awareness.

Community Engagement:

Innovative flood monitoring systems also involve the participation of local communities. Community based flood monitoring programs can use low-cost sensors and citizen science initiatives to collect data and raise awareness about flood risks.

Overall, these innovations in flood monitoring systems aim to improve early warning systems, enhance flood preparedness, and mitigate the impacts of floods on communities and infrastructure.

Incorporating Internet of Things (IOT) innovations can significantly enhance flood monitoring systems. Here are some IIT-driven innovations to solve problems in flood monitoring:

Low-Cost Sensor Networks:

Develop cost-effective IOT sensor networks that can be deployed in flood-prone areas at scale. These sensors can measure water levels, rainfall, temperature, and other relevant data. Lower costs make it feasible to deploy more sensors, improving monitoring coverage.

Energy-Efficient Sensors:

Design IOT sensors with low power consumption to extend their operational life, especially in remote areas with limited access to power sources. Solar-powered and energy-efficient sensors can function continuously without frequent battery replacements.

Wireless Connectivity:

Utilize wireless IOT communication protocols like LORAWAN or NB-IOT to transmit data from sensors to a central monitoring system. These protocols offer long-range and low-power capabilities, making them suitable for remote and rural areas.

Data Fusion:

Employ IoT-enabled data fusion techniques to integrate information from various sensors, weather forecasts, and remote sensing technologies. This allows for a comprehensive understanding of flood conditions and better-informed decision-making.

Edge Computing :

Implement edge computing capabilities in IoT devices to process and analyze data locally, reducing the need for continuous data transmission and enabling faster response times in critical situations.

Real-Time Data Analytics:

Utilize IoT data analytics platforms to process and analyze incoming data in real time. This enables the detection of anomalies and the generation of early warnings, ensuring timely responses to changing flood conditions.

Predictive Maintenance:

Apply IoT for predictive maintenance of flood monitoring equipment. Sensors can monitor the health of monitoring devices, detecting issues before they lead to failures and ensuring system reliability.

Scalability:

Design IoT-based flood monitoring systems to be scalable and adaptable to changing monitoring requirements. As the need for monitoring expands or shifts to different areas, the system can easily accommodate new sensors and data sources.

Mobile Applications:

Develop user-friendly mobile applications that allow both authorities and the public to access real-time flood data, receive alerts, and report local conditions. These apps can facilitate community engagement and data collection.

Machine Learning for Predictive Modeling:

Integrate machine learning algorithms into IoT systems to create predictive flood models. These models can forecast flood events based on historical data, current conditions, and climate trends, providing valuable lead time for evacuation and response planning.

LITERATURE REVIEW

By leveraging IoT technologies and innovations, flood monitoring systems can become more robust, efficient, and responsive. These advancements enable better flood prediction, early warning, and disaster management, ultimately reducing the impact of flooding on communities and infrastructure.

Natural hazards such as floods, storms, tsunamis and others pose a significant threat to lives and property around the world . Without proper monitoring and effective mitigation measures, these natural perils often culminate in disasters that have severe implications in terms of economic loss, social disruptions, and damage to the urban environment . Historical records have shown that flood is the most frequent natural hazard (see Figure 1), accounting for 41% of all natural perils that occurred globally in the last decade . In this period alone (2009 to 2019), there were over 1566 flood occurrences affecting 0.754 billion people around the world with 51,002 deaths recorded and damage estimated at $371.8 billion . Put in context, these statistics only account for "reported" cases of large-scale floods, typically considered flood disasters. A flood disaster is defined as a flood that significantly disrupts or interferes with human and societal activity, whereas a flood is the presence of water in areas that are usually dry .

The global impact of a flood would be more alarming if these statistics incorporated other numerous small-scale floods where less than 10 people may have died, Another method of flood monitoring and prediction is the use of wireless sensors powered by the Internet of Things (IoT) technology. IoT and computational models such as artificial neural network (ANNs) have opened up new doorways, allowing the design of new hardware and software to provide real-time water-level data as required for flood monitoring and forecasting . Today, many flood-prone countries, including the tropical nation of Indonesia that suffers from annual monsoonal rainfall, are exploring IoT sensors to gather intelligence for issuing early warnings and evacuate orders to people at risk of major floods . The IoT has gained increased popularity in the last decade, particularly within the context of smart city applications such as real-time monitoring of urban drainage networks using wireless sensors . A review of the relevant literature is needed to provide an in-depth understanding of the research scope and progress achieved in the last decade of using IoT sensors for flood monitoring in both occupied lands and other coastal sites such as lakes and lagoons.

This study provides an opportunity to update readers on recent advancements in flood monitoring, and how technology is used in the literature to map the flood events. The motivation behind this study is to highlight existing solutions and adapt them to better manage coastal lagoons, which impose flood threat to the local communities. This study presents a systematic review of the literature focusing on the use of computer vision and IoT-based sensors in flood monitoring, mapping and prediction for both occupied lands and coastal sites such as lagoons.

The main contributions of this article are as follows:100 or more people may have been affected or where there is no declaration of a state of emergency or a call for international assistance. Nevertheless, the current situation calls for improved ways of monitoring and responding to floods. The importance of improved flood monitoring cannot be overemphasized given the growing uncertainty associated with climate change and the increasing numbers of people living in flood-prone areas .

Significant efforts have been made globally to develop cost-effective and robust flood monitoring solutions. A common approach is based on computer vision, wherein relevant images from existing urban surveillance cameras are captured and processed to improve decision making about floods [8]. These types of camera-based applications involve low equipment cost and wide aerial coverage thereby enabling the detection of flood levels at multiple points. The wider coverage gives the computer vision approach an advantage over the traditional flood monitoring method that relies on fixed-point sensors [9]. Computer vision is based on image processing techniques that have been widely applied in many fields, including aerospace, medicine, traffic monitoring, and environmental object analysis [10]. In the last decade, research efforts have intensified in exploring computer vision to improve flood monitoring, flood inundation mapping, debris flow estimation, and post-flood damage estimation. To effectively harness this knowledge and foster rapid research progress, it is important to review the relevant literature and provide a constructively critical appraisal of scientific production, including recommended directions for future research.

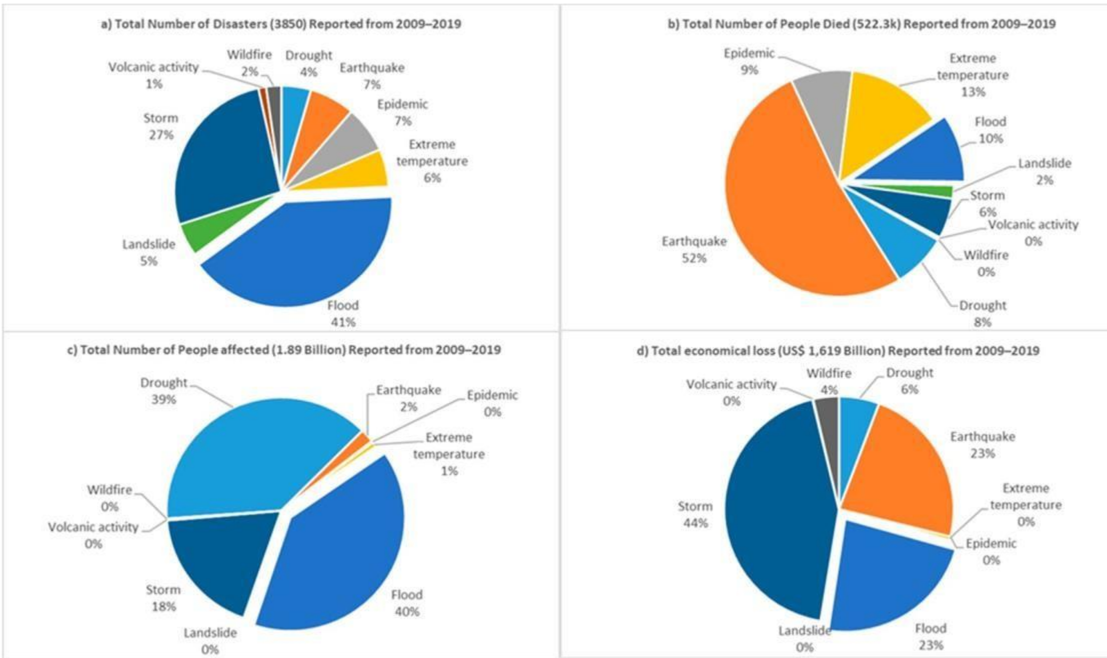


Figure 1. Comparison of different disaster types reported from 2009 to 2019: (**a**) total number of reported disasters; (b) total number of deaths; (**c**) total number of people affected; and (**d**) total economic loss [4].

APPLICATIONS

- Early Warning and Alerting:

Flood monitoring systems are used to detect rising water levels in rivers, lakes, and urban areas. When water levels reach critical levels, these systems send out alerts to relevant authorities and the public, allowing for timely evacuations and flood preparedness.

- Weather Forecast Integration:

By integrating real-time weather data, these systems can predict potential flooding events based on rainfall forecasts, enabling proactive measures to be taken to reduce flood risks.

- Infrastructure Protection:

Flood monitoring systems help protect critical infrastructure such as bridges, roads, and buildings. Sensors can trigger alarms when water levels threaten the integrity of these structures, allowing for rapid response and preventive actions.

- Reservoir Management:

In regions with dams and reservoirs, flood monitoring systems help manage water release to prevent downstream flooding during heavy rainfall, ensuring the safety of downstream communities.

- Environmental Monitoring:

Flood monitoring systems also track the environmental impact of floods, including water quality, sediment transport, and damage to ecosystems. This data aids in post-flood recovery and restoration efforts.

- Insurance and Risk Assessment:

Insurance companies and risk assessment agencies use data from flood monitoring systems to determine flood risk and set insurance premiums accordingly. This helps individuals and businesses in flood-prone areas make informed decisions about coverage.

- Urban Planning:

City planners use flood monitoring data to inform zoning decisions, building codes, and infrastructure development, reducing flood vulnerability in growing urban areas.

- Research and Analysis:

Scientists and researchers rely on flood monitoring data to study flood patterns, climate change impacts, and the effectiveness of flood mitigation measures.

- Disaster Response:

During and after a flood event, real-time data from monitoring systems assists emergency response teams in deploying resources, assessing the extent of damage, and coordinating rescue and relief efforts efficiently.

- Public Awareness:

Flood monitoring systems can disseminate information to the public through various channels, including websites, mobile apps, and social media, helping residents stay informed and make informed decisions during flood events.

- Community Resilience :

By providing communities with accurate and up-to-date information on flood risks and conditions, these systems empower individuals and organizations to take proactive measures to enhance their resilience to flooding.

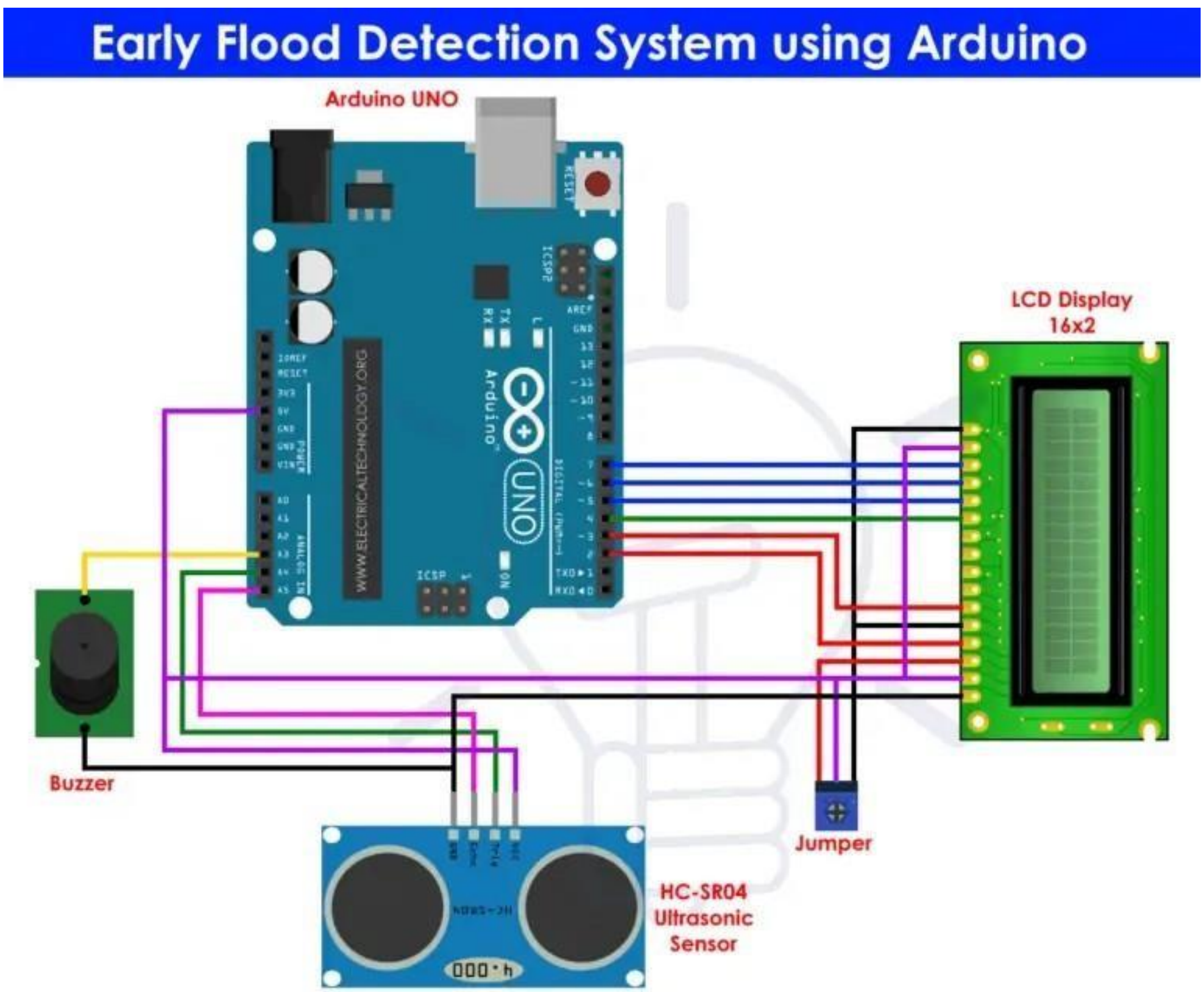
In summary, flood monitoring systems serve a wide range of applications, from protecting lives and property to supporting research and urban planning. They play a crucial role in mitigating the impact of floods and improving overall disaster preparedness and response.

STEPS TO DESIGN FLOOD MONITORING SYSTEM

- Gather the necessary materials: Arduino Uno board, ESP8266 Wi-Fi module, breadboard, jumper wires, water level sensor (e.g., ultrasonic sensor), and a computer with Arduino IDE installed.
- Connect the Arduino Uno and ESP8266 using jumper wires. Connect the VCC pin of the ESP8266 to the 3.3V pin of the Arduino, connect the GND pin of the ESP8266 to the GND pin of the Arduino, connect the TX pin of the ESP8266 to the RX pin of the Arduino, and connect the RX pin of the ESP8266 to the TX pin of the Arduino.
- Connect the water level sensor to the Arduino. Connect the VCC pin of the water level sensor to the 5V pin of the Arduino, connect the GND pin of the water level sensor to the GND pin of the Arduino, and connect the output pin of the water level sensor to any digital pin of the Arduino (e.g., pin 2).
- Connect the Arduino to your computer using a USB cable. Open the Arduino IDE on your computer and create a new sketch.
- Write the code to read the water level sensor data. Use the Arduino’s digital input/output functions to read the sensor output pin and store the data in a variable.
- Write the code to establish a Wi-Fi connection with the ESP8266. Use the ESP8266 library for Arduino to connect to your Wi-Fi network and set up a TCP/IP connection
- Write the code to send the water level data to a server. Use the ESP8266 library for Arduino to send an HTTP request to a server, including the water level data as a parameter.
- Upload the code to the Arduino Uno board by clicking on the “Upload” button in the Arduino IDE.
- Disconnect the USB cable from the Arduino and power it using an external power source (e.g., battery or power adapter).
- Mount the water level sensor at a suitable location for flood monitoring. Ensure that the sensor is positioned correctly and securely.
- Power on the Arduino and ESP8266. The Arduino will read the water level data from the sensor and send it to the server via the Wi-Fi connection.
- On the server side, set up a program or database to receive and store the incoming water level data. This can be done using programming languages like Python or PHP, or using a database management system like MySQL or MongoDB.

- Analyze the stored water level data on the server to monitor and track flood conditions. You can create visualizations, alerts, or notifications based on the water level readings.
- Test the flood monitoring system by simulating different water levels and verifying that the data is being correctly transmitted and stored on the server.
- Make any necessary adjustments or improvements to the system based on the test results.
- Deploy the flood monitoring system in the desired location for continuous monitoring of the water levels and early detection of potential flooding events.

CIRCUIT DIAGRAM



APPARATUS REQUIRED

- ARDUINO UNO
- ESP866 WIFI MODULE
- HC-SR04 ULTRASONIC SENSOR
- BUZZER
- LCD DISPLAY
- JUMPER

ARDUINO UNO

Arduino is an open source platform which is used to develop electronics project. It can be easily programmed, erased and reprogrammed at any instant of the time. There are many Arduino boards available in the market like Arduino UNO, Arduino Nano, Arduino Mega, Arduino lilypad etc. with having different specification according to their use.

In this project we are going to use Arduino UNO to control home appliances automatically. It has ATmega328 microcontroller IC on it which runs on 16MHz clock speed. It is a powerful which can work on USART, I2C and SPI communication protocols.

This board is usually programmed using software Arduino IDE using micro USB cable. ATmega328 comes with pre programmed onboard boot loader which makes it easier to upload the code without the help of the external hardware. It has vast application in making electronics projects or products. The C and C++ language is used to program the board which is very easy to learn and use.

Arduino IDE makes it much easier to program. It separates the code in two parts i.e. void setup() and void loop(). The function void setup() runs only one time and used for mainly initiating some process whereas void loop() consists the part of the code which should be executed continuously.

This model consists of 6 analog input pins and 14 digital GPIO pins which can be used as input output 6 of which provides PWM output and analog using pinMode(), digitalWrite(), digitalRead() and analogRead() functions. 6 analog input channels are from pins A0 to A5 and provide 10 bit resolution.

The board can be powered either from using USB cable which operates at 5 volts or by DC jack which operates between 7 to 20 volts. There is on board voltage regulator to generate 3.3 volts for operating low powered devices.

Since the ATmega328 work on USART, SPI and I2C communication protocol, has 0 (Rx) and 1(Tx) pins for USART communication, SDA (A4) and SCL (A5) pin for I2C and SS (10), MOSI (11), MISO (12) and SCK (13) pins for SPI communication protocol.

ESP8266 WIFI MODULE

In this advanced system the initial stage indicates the level of water and the other parameters like flow rate temperature and humidity. Then these information is transferred to the web server or the IOT via a Wi-Fi module, here the ESP8266 is used as Wi-Fi module.

Its Wi-Fi connectivity enables real-time communication, facilitating immediate response to changing flood conditions and the ability to issue alerts in a timely manner. Cloud Storage: Data collected by ESP8266 devices can be stored in the cloud, making it easily accessible for historical analysis and future planning. Scalability: The ESP8266’s cost-effectiveness and ease of use make it a scalable solution for deploying multiple flood monitoring sensors across a large area.

HC-SR04 ULTRASONIC SENSOR

HC-SR04 is an ultrasonic sensor which helps to measure distances in many places with no human contact. It works on the principle same as of RADAR and SONAR and provides an efficient way of measuring distances in a very precise way.

Theoretically it can measure distances up to 450 cm but practically it can measure distances from 2 cm to 80 cm with accuracy of 3 mm. It is operated at 5 volts, current less than 15mA and 40 Hertz frequency.

The transmitter of the HC-SR04 sensor transmits an ultrasonic wave in the air. If this wave is reflected by some object in the range of sensor then the reflected wave in the air is received by the receiver of the sensor. So to calculate the distance using above formula we should know the speed and time.

BUZZER

:

Flood Detection: Flood sensors, such as water level sensors or rain sensors, can be integrated into the IoT system. These sensors detect changes in water levels or heavy rainfall.

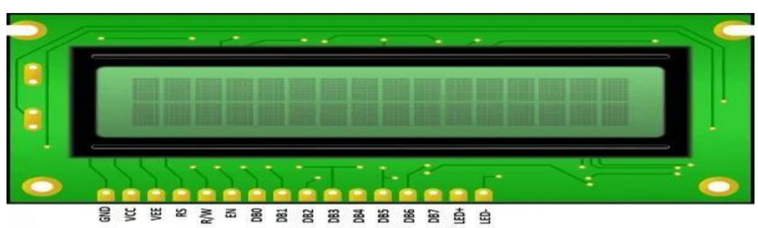
Thresholds: Set predefined thresholds for the sensors. When water levels rise above a certain point or when there's heavy rainfall, the IoT system triggers an alert. Buzzer Alert: When the system detects the predefined conditions, it activates a buzzer. The buzzer generates a loud and distinctive sound that can alert nearby individuals to the potential flood situation.

Integration with Other Alerts: The buzzer can be part of a comprehensive alert system that also includes notifications sent via SMS, email, or mobile apps to ensure a wider range of people are informed about the flood situation.

Power Source: Ensure that the buzzer has a reliable power source, such as a backup battery, in case of power outages during the flood. Remote Monitoring: With the help of IoT, the flood monitoring system can be remotely monitored, allowing authorities to take immediate action and provide real-time updates to the community.

This setup helps improve flood preparedness and safety by providing timely warnings to residents and authorities when flood conditions are detected.

16×2 LCD DISPLAY



LCD (Liquid Crystal Display) technology is commonly employed in flood monitoring systems to visually convey real-time information critical for assessing and responding to flood events. These displays offer excellent readability, even in various lighting conditions, ensuring that vital data, such as water level measurements, weather conditions, and flood alerts, is readily accessible to emergency responders and the public.

Designed for durability, LCD displays can withstand outdoor elements when properly encased, making them dependable for long-term use in such systems. Additionally, LCDs serve as user interfaces for configuration and calibration, enable the presentation of warning messages when flood levels reach critical thresholds, and can even facilitate remote monitoring and customization of displayed data, making them versatile and indispensable components of flood monitoring solutions.

Interfacing 16 X 2 LCD with Arduino UNO is pretty easy. There are various type of LCDs available in the market but the one we are using in this project is 16×2 which means it has two rows and in each row we can display 16 characters.

This module has HD44780 driver from Hitachi on it which helps to interface and communicate with the microcontrollers. This LCD can work in 4 bit mode and 8 bit mode. In 4 bit mode only 4 data pins are required to establish connection between LCD and microcontroller whereas in 8 bit mode 8 data pins are required.

Pin Description of 16×2 LCD module:

Pin on LCD	Description
VSS	Ground Pin
VCC	+5V power supply
VEE	Pin to change the contrast of LCD
RS	Register Select: Data Mode or Command Mode
RW	Read or Write Mode
E	Enable LCD
DB0-DB7	Data and command is fed using these pins
LED+	Anode of the backlight LED
LED-	Cathode of the backlight LED

Here we are going to use it in 4 bit mode as it requires less number of wires and makes the circuit simplified. Let's look at the pin description of 16×2 LCD.

This LCD does not has its own light so there is a LED behind the screen which acts as the backlight for the display. Interfacing this LCD with Arduino UNO is pretty easy as Arduino IDE provides a LiquidCrystal library which has many inbuilt function to make initialize and print anything on the display easier.

ESP8266 Python Code (for forwarding data to Thinkspeak):

To forward data to ThingSpeak using an ESP8266 in Python, you'll need to use the ThingSpeak API. Here's an example code snippet that demonstrates how to send data to ThingSpeak using the ESP8266 and the `urequests` library:

```python
Import urequests as requests

Import machine

Import dht


# Define your ThingSpeak API key and channel ID

Api_key = 'YOUR_API_KEY'

Channel_id = 'YOUR_CHANNEL_ID'


# Initialize the DHT sensor (you can replace this with your sensor)

D = dht.DHT22(machine.Pin(2))


# Function to read sensor data

Def read_sensor_data():

   d.measure()

   temperature = d.temperature()

   humidity = d.humidity()

   return temperature, humidity


# Function to send data to ThingSpeak

Def send_to_thingspeak(temp, hum):

   url = https://api.thingspeak.com/update?api_key={0}&field1={1}&field2={2}.format(api_key, temp, hum)

   response = requests.get(url)

   print("Data sent to ThingSpeak. Status code:", response.status_code)

   response.close()
```

```
# Main loop

While True:

   Temperature, humidity = read_sensor_data()

   Send_to_thingspeak(temperature, humidity)

   Machine.deepsleep(300000)  # Sleep for 5 minutes (300,000 ms)
```

Make sure to replace 'YOUR_API_KEY' and 'YOUR_CHANNEL_ID' with your actual ThingSpeak API key and channel ID. You'll also need to adjust the sensor setup according to your hardware.

This code collects temperature and humidity data from a DHT22 sensor, sends it to your ThingSpeak channel, and then puts the ESP8266 into deep sleep for 5 minutes before repeating the process. This is a basic example, and you can modify it according to your specific requirements.

Arduino Code (for collecting sensor data):

Arduino code for collecting sensor data in a flood monitoring project using an ultrasonic sensor (HC-SR04) to measure water level. This code sends the data to a ThingSpeak channel:

```cpp
#include <ThingSpeak.h>

#include <ESP8266WiFi.h>


// Define your WiFi credentials

Const char* ssid = "YOUR_SSID";

Const char* password = "YOUR_PASSWORD";
```

```
// Define your ThingSpeak API key
Const char* apiKey = "YOUR_API_KEY";

// Define the pins for the ultrasonic sensor
Const int triggerPin = D1;  // GPIO5
Const int echoPin = D2;    // GPIO4

// Initialize the WiFi client
WiFiClient client;

Void setup() {
  Serial.begin(115200);
  pinMode(triggerPin, OUTPUT);
  pinMode(echoPin, INPUT);
  ThingSpeak.begin(client);
  WiFi.begin(ssid, password);
  While (WiFi.status() != WL_CONNECTED) {
    Delay(1000);
    Serial.println("Connecting to WiFi…");
  }
  Serial.println("Connected to WiFi");
}

Void loop() {
  Long duration, distance;

  // Trigger the ultrasonic sensor
  digitalWrite(triggerPin, LOW);
```

```
  delayMicroseconds(2);
  digitalWrite(triggerPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(triggerPin, LOW);

  // Read the echo pulse duration
  Duration = pulseIn(echoPin, HIGH);

  // Calculate distance in centimeters
  Distance = duration / 29 / 2;

  // Send the data to ThingSpeak
  Serial.print("Distance: ");
  Serial.print(distance);
  Serial.println(" cm");

  ThingSpeak.setField(1, distance);
  Int httpCode = ThingSpeak.writeFields(1, apiKey);

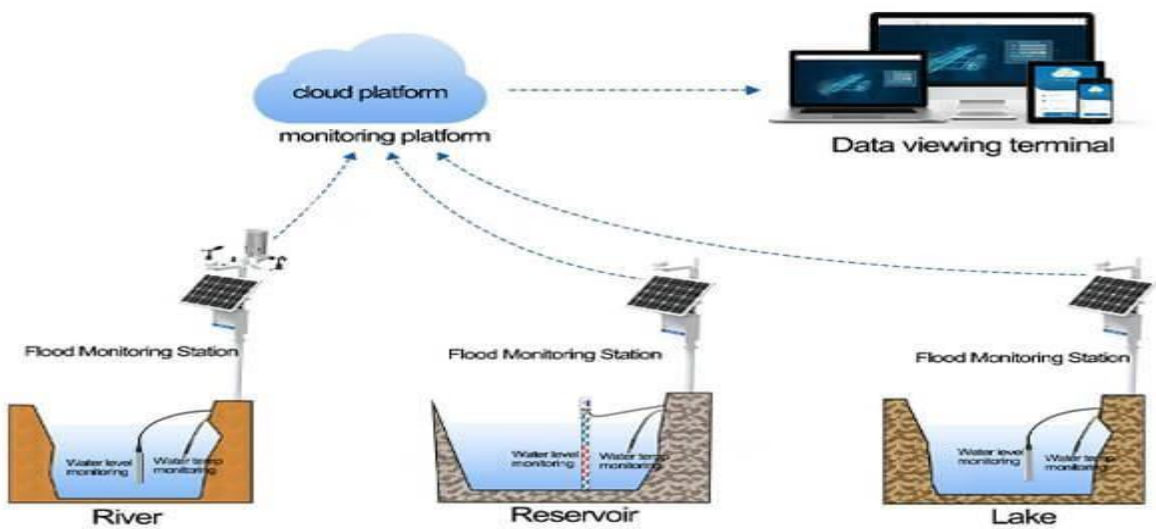
  If (httpCode == 200) {
    Serial.println("Data sent to ThingSpeak successfully.");
  } else {
    Serial.print("Error in sending data. HTTP code: ");
    Serial.println(httpCode);
  }

  // Wait for a while before taking the next reading
  Delay(600000);  // Send data every 10 minutes (adjust as needed)
}
```

```

Before using this code, make sure you've installed the ThingSpeak library in your Arduino IDE, and replace 'YOUR_SSID', 'YOUR_PASSWORD', and 'YOUR_API_KEY' with your own WiFi credentials and ThingSpeak API key.

This code sets up the ultrasonic sensor, connects to WiFi, measures water level, and sends the data to a ThingSpeak channel every 10 minutes. You may need to adjust the delay to match your specific project requirements.




## *BASIC CONCEPTS/TECHNOLOGIES USED*

Hardware module

In this project, some hardware is used that are microcontroller, sensors, components required for power supply. The hardware collects the water level, pressure of water, rainfall measure to detect the levels of the flood. The hardware consists of Wi-Fi enabled controller which connects to the server and allows to share the data to through internet.
```

1. Microcontroller- This does the controlling with processing .Microcontroller will take the information from the sensor .This information will sent to the admin through the database

2. Sensors-This will collect the information from the particular nodes which are located at certain site. There are four sensors we are going to use in this project. They Are as follows:

Water level measurement:

This sensor is used to measure the water level height. For that we are going to use ultrasonic sensor which emits short, high frequency sound pulses at regular intervals. If they strike an object, then they are reflected back as echo signals to the sensors.

Rainfall measurement:

This sensor is used to measure the average rainfall. For that we are going to use same ultrasonic sensor. Ultrasonic sensor is 4 pin sensor. Those are ground connection (GND), trigger, echo and last current (VCC).

Temperature and Humidity:

This sensor is used to measure change in atmospheric temperature and Humidity. For this we are using DHT11 sensor which works on one wire protocol and gives digital output.

Pressure measurement:

This sensor is used to determine the atmospheric pressure. For this we are going to use BMP 180Barometric sensor.

3. Power Supply- In real time we get 230v AC, in actual project we do not need this amount of power supply so we convert this AC power supply to DC power supply.
4. Software Module

In this module, we have done an android application as well as the website application for this project. admin web page will contain and display the information like login, registration, number of users registered to the app, status of the sensor, safe places near flood affected area where people can migrate and that places are shown on the map.

The android application will be used by the users who are register. After registration the user can login with a unique username and password. And then user can access all facilities provided by application.

Application is provided the information like current status of water level and temperature etc. This app contain map which are show the safe places near the user and also the current place where the user is.

Database Module

Microcontroller will send the values measured by the sensors to the server. This will contain the number of users registered to App; this will also show the safe places through the map. The data uploaded on server is stored on the database. The stored data is then routed to the front end web applications and mobile application.

PROPOSED SYSTEM

1. There will be a node as shown in above diagram.

2. This node is the independent flood monitoring node Equipped with necessary sensors and connectivity Modules.

3. It has three major stages, Including Sensors, Controller, Wi-Fi interface to upload the information on server.

4. Data from various sensors are collected by the ESP and Is then computed and uploaded on the server.

5. The data uploaded on server is stored on the database.

6. The stored data is then routed to the front end web Applications and mobile applications.

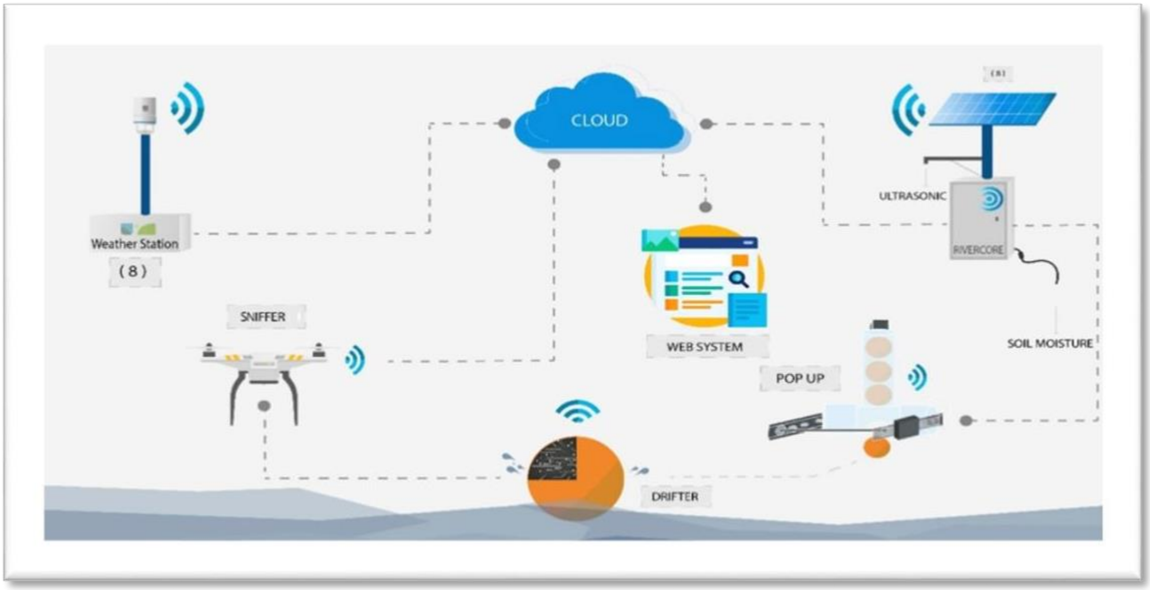
Module Description

The overall system consists of 3 main stages –

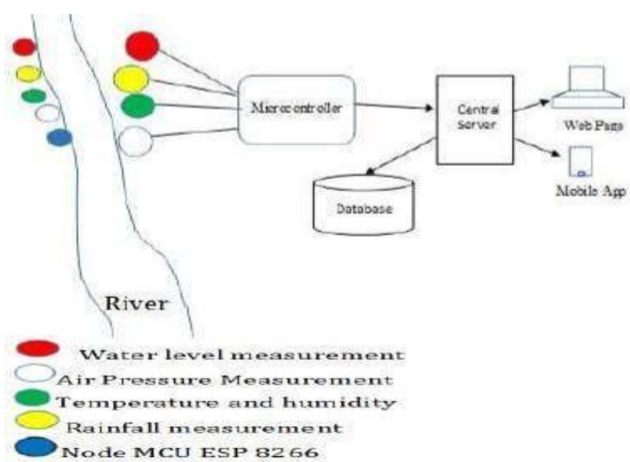
1] Hardware nodes

2] Cloud Architecture

3] Front end clients (mobile app)



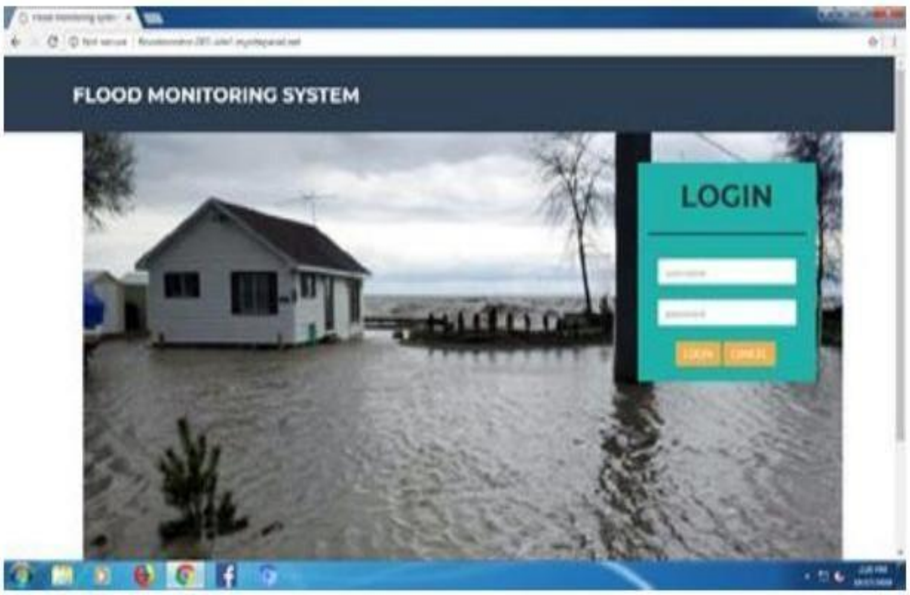
The Hardware collects the water level, Pressure of water, Rainfall measure to detect the levels of the flood. The hardware consists of Wi-Fi enabled controller which connects to the server and allows to share the data to through internet .The architecture contains server and database which handles the data coming from the devices and saves it in the database. The Front end apps will have http client to establish connection to device and backend. The app will collect the data from backend and represent it on the map. All these communication will be done over the internet though http protocol.



OUTPUT OF IMPLEMENTATION

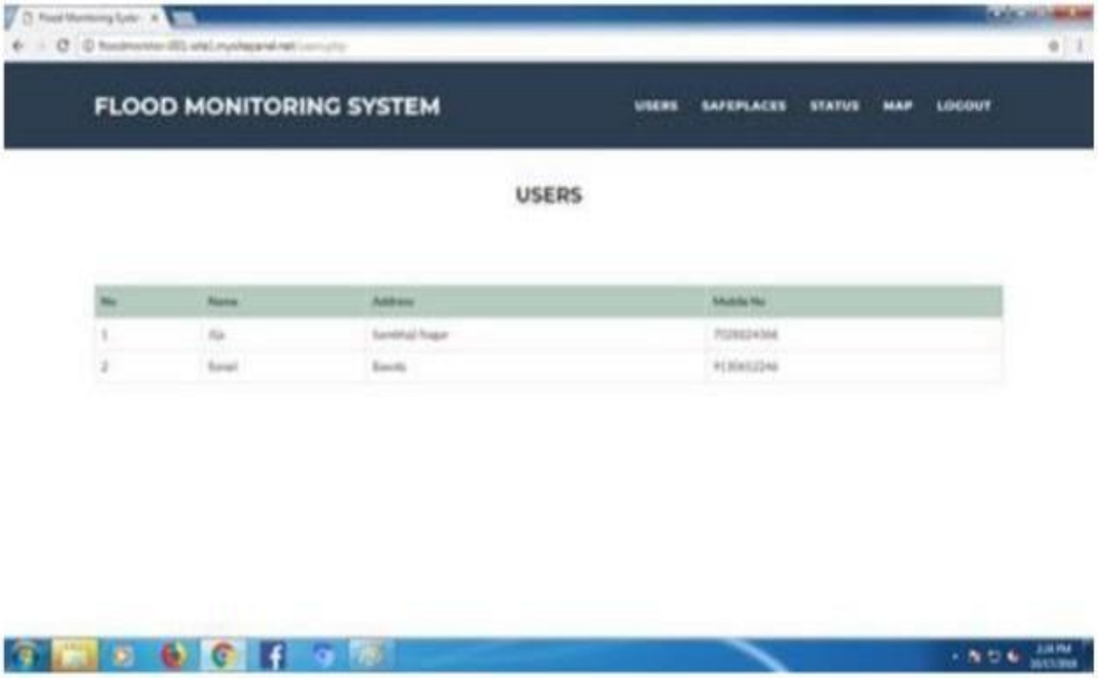
SNAPSHOT OF WEBSITE

WEBSITE: LOGIN PAGE



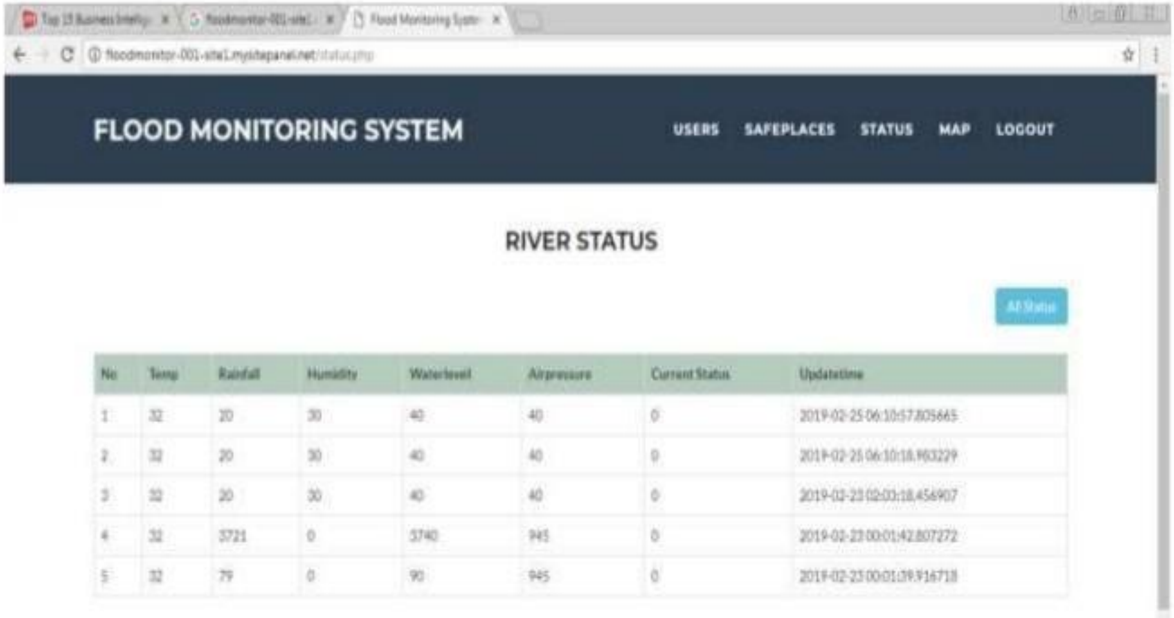
Above snapshot is of the website login page (the first page of website) through which the admin can enter his username and password and have access to the facilities Provided by the website.

WEBSITE : USERS



Above page shows the list of registered users to admin on website who have registered using the android application.

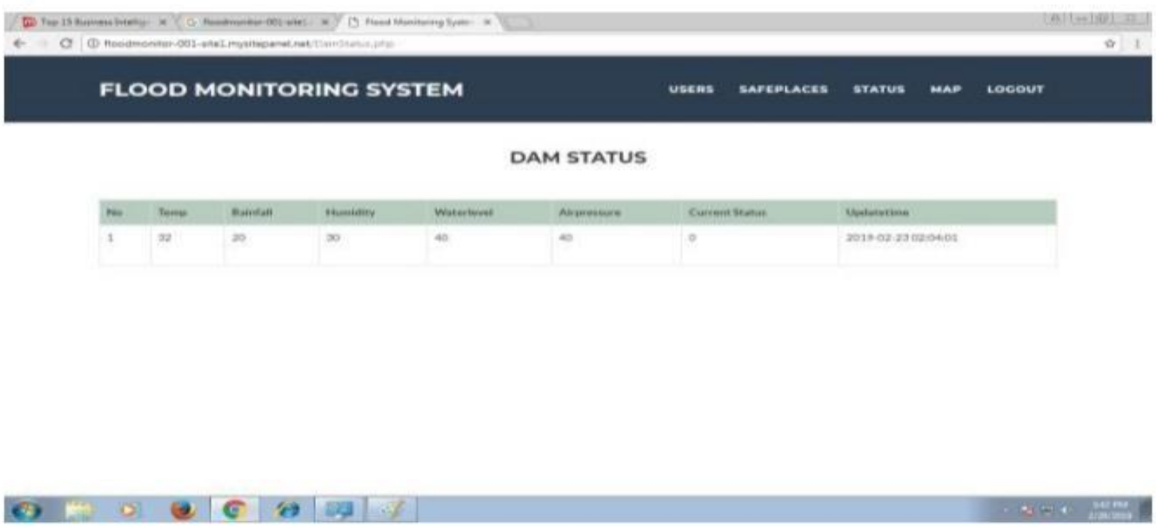
WEBSITE: RIVER STATUS



No	Temp	Rainfall	Humidity	Waterlevel	Airpressure	Current Status	Updatetime
1	32	20	30	40	40	0	2019-02-25 06:10:57.805665
2	32	20	30	40	40	0	2019-02-25 06:10:18.963229
3	32	20	30	40	40	0	2019-02-23 02:03:18.456907
4	32	3721	0	3740	945	0	2019-02-23 00:01:42.807272
5	32	79	0	90	945	0	2019-02-23 00:01:39.916718

Above page shows the current values detected by the sensors from river side, recent top 10 status updates are shown.

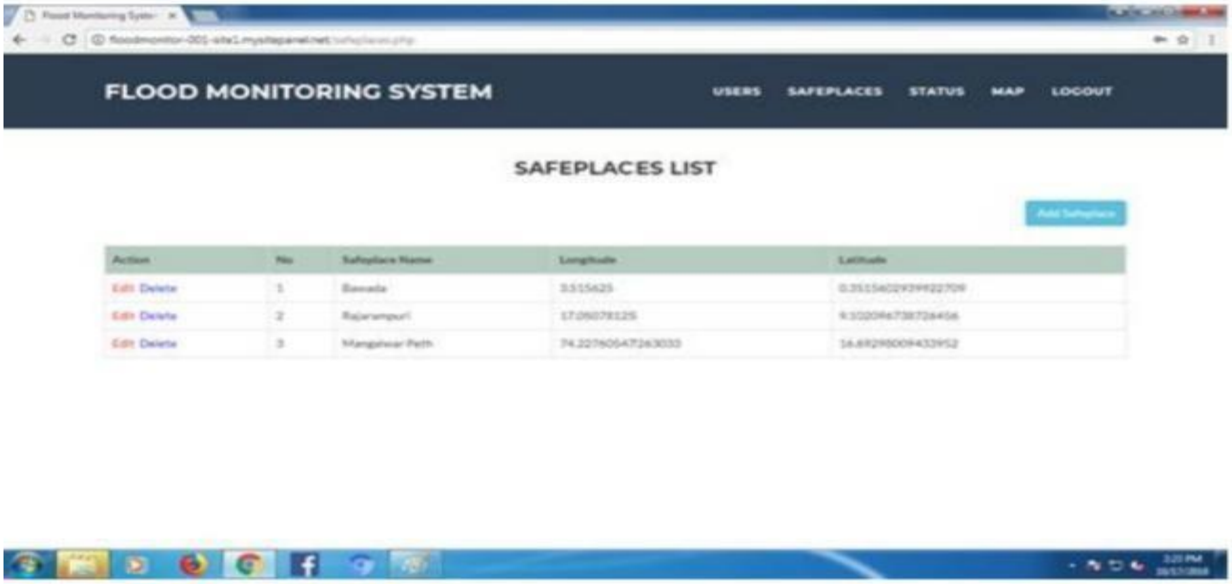
WEBSITE: DAM STATUS



No	Temp	Rainfall	Humidity	Waterlevel	Airpressure	Current Status	Updatetime
1	32	20	30	40	40	0	2019-02-23 02:04:01

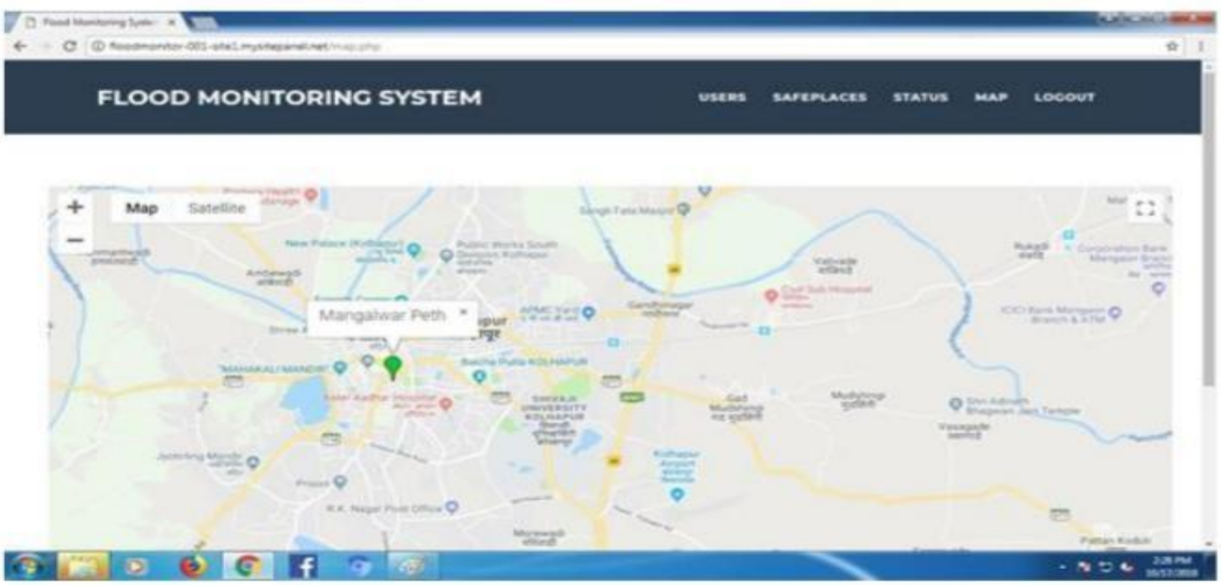
Above page shows the current values at admin site detected by the sensors from dam side, recent top 10 status updates are shown.

WEBSITE : SAFE PLACES LIST



Above page shows the safe places list where the people can migrate after the alert is received for migrating to a safe place. (Alert will be given after a threshold value is reached).

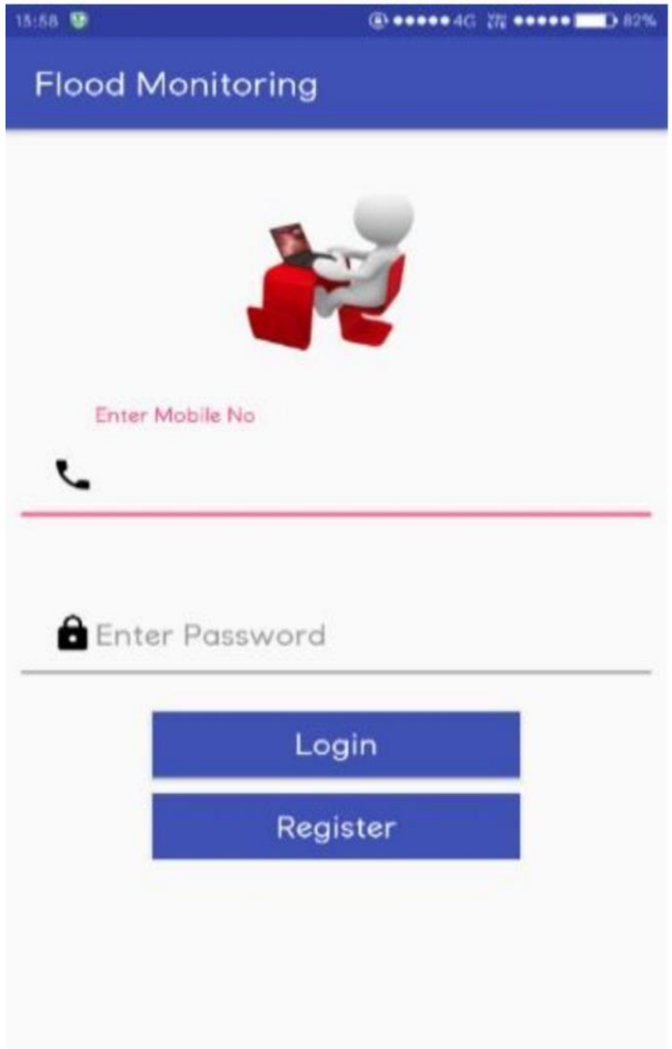
WEBSITE:MAP



Above page shows the map indicating the safe places.

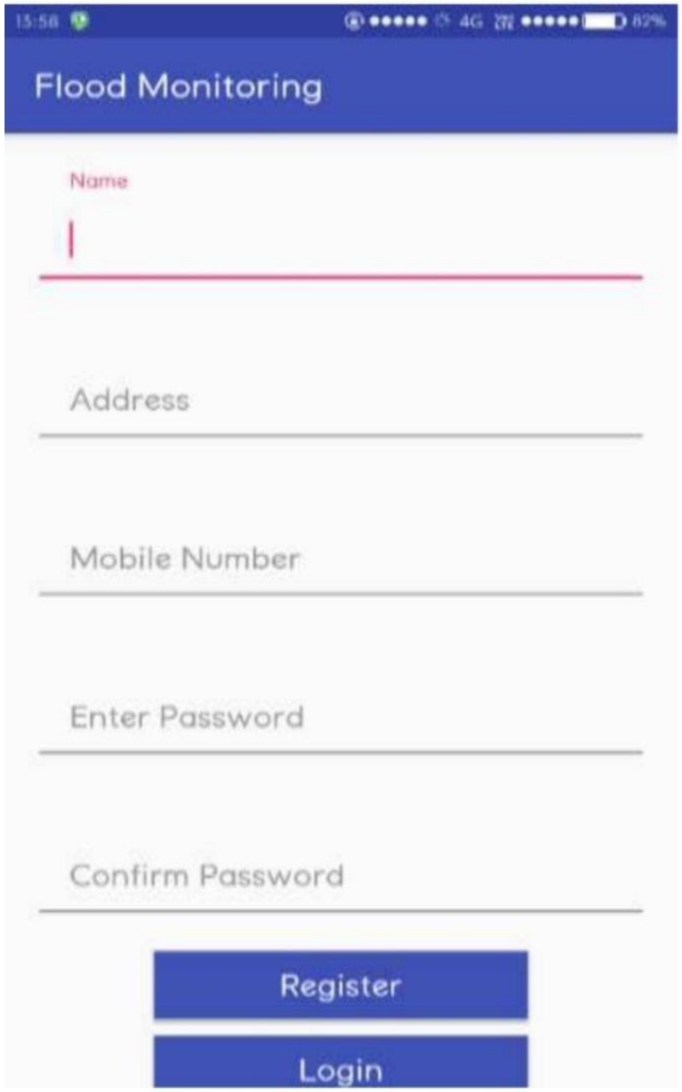
SNAPSHOTS OF ANDROID

ANDROID:USER LOGIN



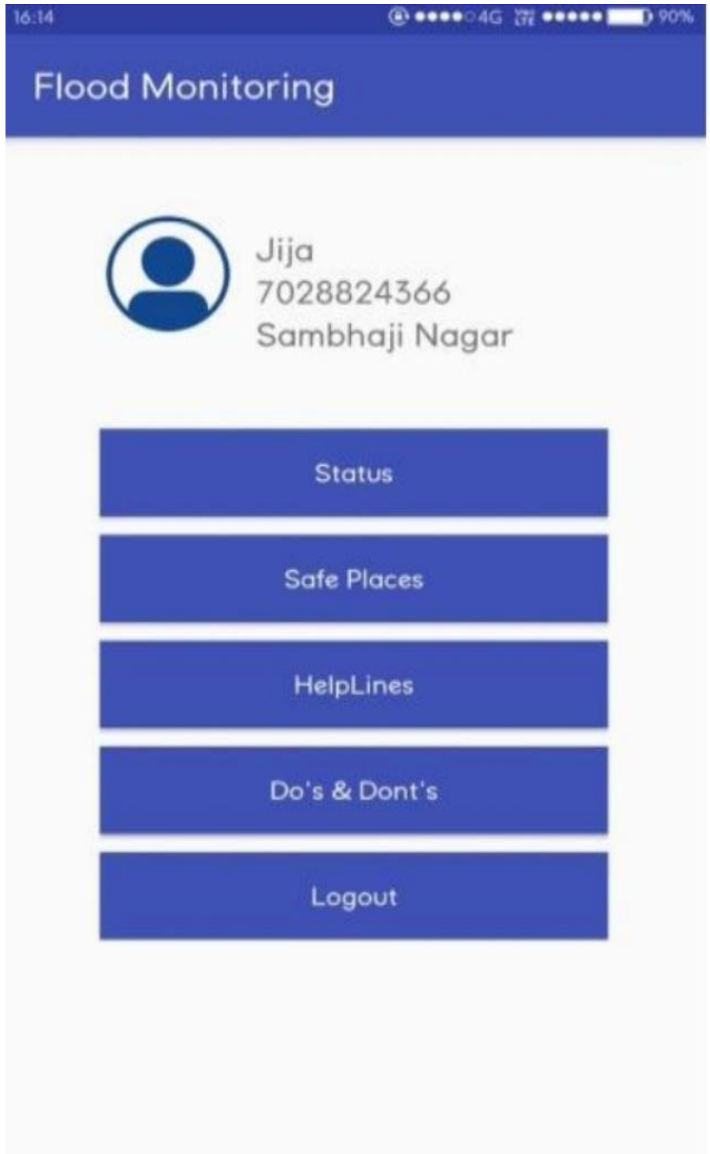
Above page is the login page of android application users who can login using their registered mobile numbers and Password.

ANDROID: REGISTRATION



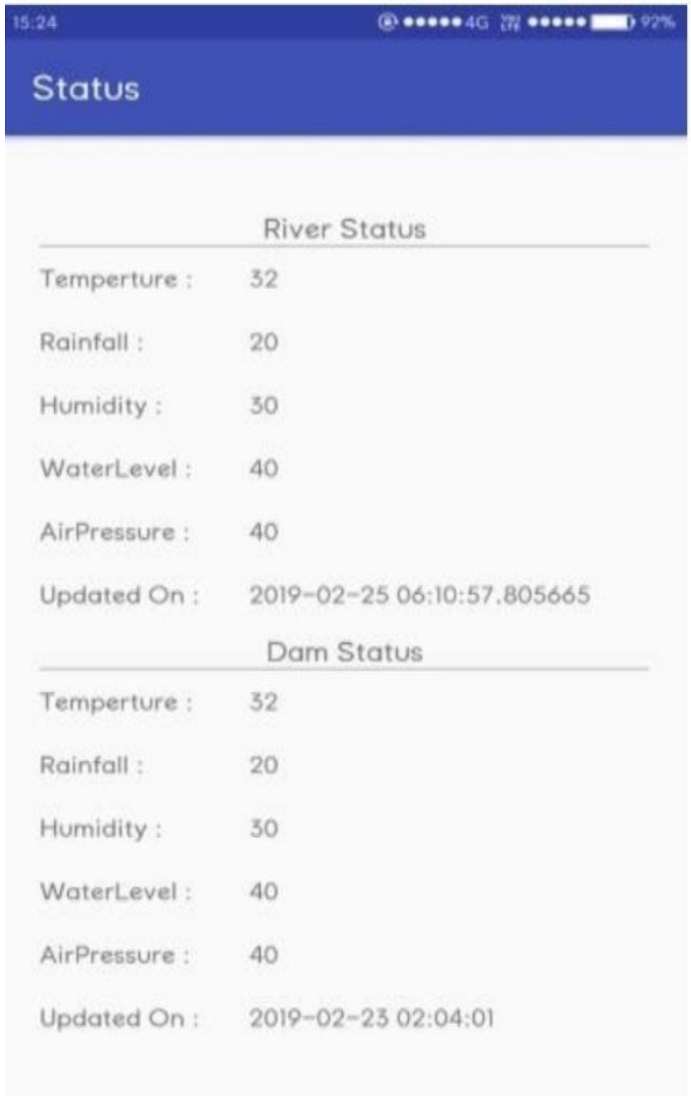
Above page shows the registration page of android application where the users can register by entering their name, address, mobile number and password.

ANDROID: LOGGED IN PAGE



Above page shows the application view after a registered user has logged in. We have provided five buttons which after pressing will direct the user to the status, safe places page helpline page,instructions page respectively. A third button is provided which is the logout button which the user will press when he/she wishes to leave the application.

ANDROID: STATUS PAGE



Above page shows the Android application view of the data that is detected by the sensors from both river and dam side. Most recent update will be shown to the user.

ANDROID: INSTRUCTIONS

Above page shows the instruction for taking care during the floods.

ANDROID: HELPLINES

Above page contains emergency helpline numbers. Through these numbers users can make call on those helpline number in emergency by simply clicking on the number.

ANDROID:MAP

Above page shows the android application map view where the current location and safe location of the user is shown.

In a flood monitoring IoT project, web development technologies can be used in various aspects to enhance data visualization, user interaction, and data analysis. Here are some areas where web development technologies can be applied:

1. User Interface (UI):

HTML/CSS: Use HTML and CSS to create a user-friendly web interface for users to access flood data and monitor conditions.

JavaScript: Implement interactive elements on the web interface for real-time updates and user-friendly features.

2. Data Visualization:

JavaScript Libraries: Utilize JavaScript libraries such as D3.js or Chart is to create dynamic and visually appealing charts, graphs, and maps to display flood-related data.

3. Real-time Data Streaming:

WebSockets: Implement WebSockets for real-time data streaming from IoT sensors to the web interface, ensuring users receive up-to-the-minute information.

4. Data Storage and Retrieval:

Backend Technologies: Use server-side scripting languages like Node is, Python, or Ruby to manage data storage, retrieval, and processing.

Databases: Utilize databases like MySQL, PostgreSQL, or NoSQL databases for storing historical flood data.

5. Mobile Compatibility:

Make the web interface responsive to ensure it works well on both desktop and mobile devices for easy access in the field.

6. User Authentication and Security:

Implement user authentication and authorization systems to control access to sensitive flood data.

7. Geospatial Integration:

Use mapping libraries like Leaflet or Google Maps API to display flood data on interactive maps, allowing users to see the affected areas.

8. Alerting and Notification:

Integrate email, SMS, or push notification systems using web technologies to alert authorities and residents in real-time when flood thresholds are reached.

9. Data Analysis:

Implement data analysis algorithms on the server-side to process and analyze flood data for trends, predictions, and alerts.

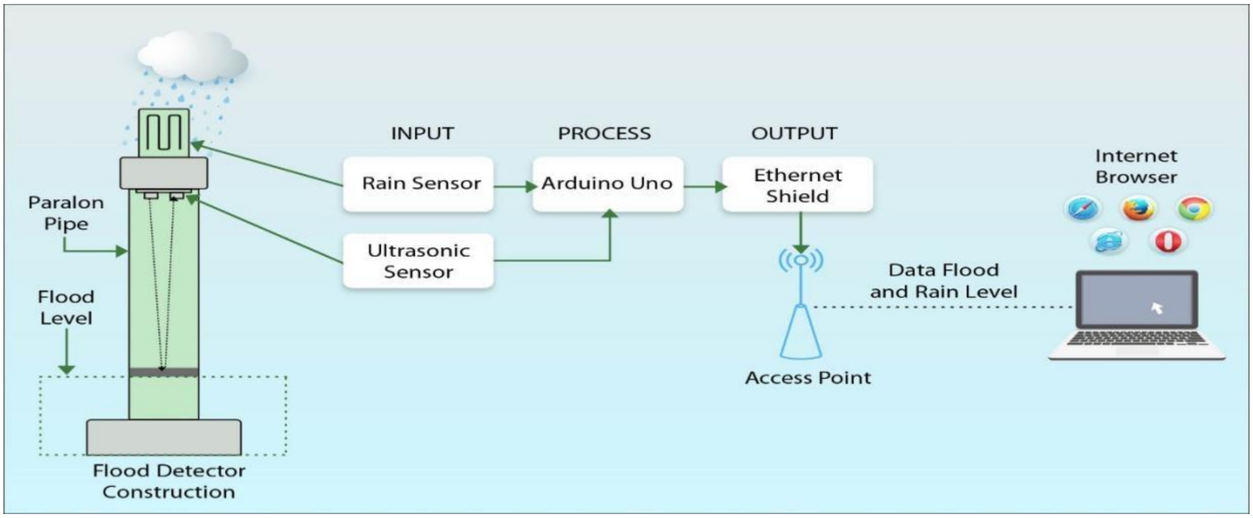
10. API Development:

Create APIs using RESTful or GraphQL to allow other applications or services to access flood data programmatically.

11. Cloud Services:

Utilize cloud platforms like AWS, Azure, or Google Cloud for scalability, storage, and processing of flood data.

By incorporating web development technologies in these aspects of your flood monitoring IoT project, you can provide a comprehensive and user-friendly solution for monitoring and responding to flood conditions.



CONCLUSION

As India faced recent devastating flood in Kerala, there arise a need of efficient flood monitoring systems. Flood forecasting and the issuing of flood warnings are effective ways to reduce damage. The proposed system will be efficient because it has better coordination of monitoring, communication and transmission technologies which are adaptable to background condition. The proposed system also ensures increased accessibility for assessment of emergency situations and enhances effectiveness and efficiency in responding to catastrophic incidents. In summary, the proposed system would be beneficial to the community for decision making and evacuation planning purpose.