

COLLEGE NAME: Priyadarshini Engineering college

COLLEGE CODE :5119

COURSE NAME: Internet Of Things

GROUP NUMBER: Group 2

PROJECT TITLE:Flood Monitoring And Early Warning

PROJECT SUBMITTED TO: Skill Up online

YEAR: IIIrd Year

DEPARTMENT:Electronic And Communication Engineering

SEMESTER: 5th  
GROUP MEMBERS: Arjun V [511921106002]

Mohan Raj R[511921106017]

Sathish Kumar R[511921106032]

Gowtham Kumar B[5119211006301]

GUIDED BY:Dr.A.Banupriya,HOD/ECE

SPOC NAME:Dr.R.Thenmozhi,HOD/EEE

DESIGN :

1. \*\*Data Sources\*\*: Gathering data from various sources is crucial. This includes weather data, river and water level gauges, satellite imagery, and even social media reports.
2. \*\*Sensors and Gauges\*\*: These are placed in flood-prone areas to measure factors like water level, rainfall, and river flow. They transmit real-time data to monitoring centers.
3. \*\*Monitoring Centers\*\*: These are where the collected data is processed and analyzed. They use specialized software and algorithms to assess the potential for flooding.
4. \*\*Early Warning Systems\*\*: When the data indicates a flood risk, early warning systems are activated. These systems can send alerts to the public, emergency services, and local authorities.
5. \*\*Modeling and Prediction\*\*: Flood monitoring often involves computer models that predict how floods may evolve based on current conditions and historical data.
6. \*\*Communication Channels\*\*: Alerts and information must be communicated to the public through various channels, including mobile apps, TV, radio, and emergency sirens.
7. \*\*Response Plans\*\*: Authorities should have established flood response plans in place, including evacuation procedures, shelter management, and resource allocation.
8. \*\*Community Engagement\*\*: Public awareness and education are vital. Communities need to know what to do when a flood is imminent.

CONNECTING MOBILE APP TO THE DEVICE :

To connect a mobile app with a flood monitoring device, you would typically follow these steps:

1. \*\*Select App Platform\*\*: Determine whether your mobile app will be developed for Android, iOS, or both. This choice will influence the development tools and programming languages you’ll use.
2. \*\*Choose Communication Protocol\*\*: Identify the communication protocol or technology your flood monitoring device uses to transmit data. Common options include Wi-Fi, Bluetooth, Zigbee, LoRa, or cellular networks (e.g., 4G/5G).
3. \*\*Develop Mobile App\*\*: Create the mobile app that will receive, process, and display data from the flood monitoring device. This may involve designing a user-friendly interface and integrating the necessary features.
4. \*\*Implement Connectivity\*\*: Write the code to establish a connection between the mobile app and the monitoring device. This may require using the appropriate APIs or SDKs provided by the device manufacturer or custom programming to handle data transmission.
5. \*\*Data Parsing and Display\*\*: Implement data parsing within the app to extract and interpret information from the monitoring device. Display this data in a user-friendly format, such as graphs, maps, or notifications.
6. \*\*Security\*\*: Ensure that the communication between the mobile app and the monitoring device is secure. This may involve encryption, authentication, and other security measures to protect the data.
7. \*\*User Permissions\*\*: Set up user permissions and access controls within the app to determine who can access the flood monitoring data and any related functionalities.
8. \*\*Testing\*\*: Thoroughly test the app-device connection to ensure data accuracy and reliability. Test various scenarios, such as poor network conditions, to make sure the app handles them gracefully.
9. \*\*Deployment\*\*: Publish the mobile app on app stores (Google Play Store, Apple App Store) or distribute it through other channels, depending on your target audience.
10. \*\*User Training and Support\*\*: Provide user training or documentation to help users understand how to connect their mobile app with the monitoring device and how to interpret the data.
11. \*\*Maintenance and Updates\*\*: Continue to maintain and update the app to ensure compatibility with new devices, operating system updates, and to address any security vulnerabilities.

Connecting a mobile app with a flood monitoring device requires a combination of software development, hardware compatibility, and data transmission expertise. It’s important to work closely with device manufacturers and consider user experience throughout the process.

PYTHON PROGRAM :

Import bluetooth # Import the Bluetooth module

# Specify the device’s Bluetooth address (MAC address)

Device\_address = ’00:11:22:33:44:55’

# Establish a Bluetooth connection to the device

Try:

Sock = bluetooth.BluetoothSocket(bluetooth.RFCOMM)

Sock.connect((device\_address, 1)) # Use the appropriate channel

# Connection successful, now you can send and receive data

While True:

Data = sock.recv(1024) # Receive data from the device

If data:

# Process and display the received data in your app

Print(“Received data: “ + data.decode(“utf-8”))

Except bluetooth.btcommon.BluetoothError as e:

Print(“Connection error:”, e)

Finally:

Sock.close() # Close the Bluetooth connection when done

DIAGRAM :

