**PROJECT DEFINITION:**

The project entails the deployment of IoT sensors in proximity to water bodies and flood-prone regions, with the primary aim of meticulously monitoring water levels and furnishing early flood warnings through a publicly accessible platform. The overarching objective of this initiative is to bolster the overall preparedness and responsiveness in the face of potential floods, effectively disseminating timely warnings to both the general populace and emergency response teams. This multifaceted endeavor encompasses a comprehensive approach, commencing with the articulation of precise objectives, progressing to the intricacies of designing an IoT sensor network, followed by the development of a robust warning platform. This platform will act as the focal point for aggregating and disseminating critical information. The integration of these components, facilitated by IoT technology and Python programming, harmoniously synergizes the disparate elements into a cohesive and efficient system. By executing this project, you're not only advancing technological innovation but also directly contributing to public safety and disaster mitigation efforts.

Components and sensors:

**Node MCU ESP8266:**

* Arduino Uno is the logical next step if you’re looking for a more compact module that encompasses Wi-Fi. NodeMCU is predicated on the Esperessif ESP8266-12E Wi-Fi System-On-Chip. It is based on Lua-based firmware and is open-source.
* It’s perfect for IoT projects, especially other Wireless connectivity projects as Arduino does not work wirelessly. We either need to connect it to a Bluetooth or nRF module This chip has a great deal in common with the Arduino – they’re both microcontroller-equipped prototyping boards that can be programmed using the Arduino IDE. The ESP8266 is more updated and younger than Arduino, and therefore the ESP has stronger  specifications than Arduino.

**Ultra Sonic HC-SR04:**

**The HC-SR04 ultra Sonic** distance sensor is a sensor used for detecting the distance to an object using sensor. It is commonly used in robotics project.which requires you to avoid object,by detecting how close they are away from us. Here it used to sense the water level of the river or dam.

**LED:**

**LED** is commonly known as light emitting diode. Here we have used two diode one in green colour and another one in red colour.

**GREEN LED**:

The green LED is used to detect the normal condition in the system.

**RED LED** :

Th**e red LED** is used to denote warning or danger sign to the system and alert with buzzer .

**BUZZER**:

 The Buzzer or beeper is an audio signaling device, which may be mechanical, electromechanical, or piezoelectric (piezo for short). Typical uses of buzzers and beepers include alarm devices, timers, train and confirmation of user input such as a mouse click or keystroke. There are two types of buzzer they are active buzzer and passive buzzer.

**POWER SUPPLY:**

A power supply adapter that provides from 7 to 12V (Volts) of  Direct Current is required. The adapter is plugged onto the wall socket and the other end goes directly onto the board’s AC socket.

**BREAD BOARD:**

A bread board is used for building temporary circuits. It is useful to designer because it allows components to be removed and replaced easily. And the components can be reused again.

**JUMPER WIRE:**

A jumper wire is an electric wire that connects remote electric circuit used for printing circuit board.

**IOT WEB PLATFORM:**

We have used thing speak iot cloud platform to process the code to node MCU.

**DESIGN THINKING:**

1.PROJECT OBJECTIVE:

Objective 1: Real-time Flood Monitoring

Goal: To monitor areas at risk of flooding and provide up-to-date information to the public and decision makers.

Steps:

- Establish a network of sensors or other monitoring systems to detect flooding in vulnerable areas.

- Collect and analyze data from the sensors to determine if and where flooding is occurring.

- Provide real-time alerts to the public and decision makers when flooding is detected.

Objective 2: Early Warning Issuance

Goal: To issue early warnings to the public and decision makers when flooding is likely to occur.

Steps:

- Establish a network of sensors or other monitoring systems to detect changes in water levels in vulnerable areas.

- Analyze data from the sensors to determine when flooding is likely to occur.

- Provide early warnings to the public and decision makers when flooding is likely.

Objective 3: Public Safety

Goal: To ensure public safety by providing up-to-date information about flooding and taking proactive measures to mitigate or prevent flooding.

Steps:

- Establish a network of sensors or other monitoring systems to detect flooding in vulnerable areas.

- Collect and analyze data from the sensors to determine if flooding is occurring or likely to occur.

- Provide real-time alerts to the public when flooding is detected.

- Take proactive measures such as issuing evacuation orders or closing roads to mitigate or prevent flooding.

Objective 4: Emergency Response Coordination

Goal: To coordinate emergency response operations in the event of flooding.

Steps:

- Establish an emergency response team and plan.

- Develop protocols for communication between the emergency response team and other stakeholders.

- Provide real-time alerts to the emergency response team when flooding is detected.

- Monitor the progress of the emergency response operations and provide updates to stakeholders as needed.

2.IoT SENSOR NETWORK DESIGN:

1. Identify the areas at risk of flooding: Research and map out which areas are most prone to flooding. Determine the size and scope of the area to be monitored.

2. Define the monitoring requirements: Establish the specific requirements for the monitoring system. Consider factors such as the frequency of data collection and analysis, type of sensors, power requirements, etc.

3. Select the appropriate sensors: Choose the type of sensor that will best meet the requirements for the monitoring system. Consider factors such as the accuracy of the sensors, the range of detection, and the power requirements.

4. Deploy the sensors: Determine the optimal locations for the sensors. Ensure that the sensors are placed in areas where they will be able to accurately measure water levels.

5. Maintain the sensors: Establish a plan for regularly checking and maintaining the sensors to ensure that they are working properly.

6. Monitor the data: Establish a system for regularly collecting and analyzing the data from the sensors. Use this data to assess the risk of flooding in the area and to take appropriate action.

3.EARLY WARNING PLATFORM:

1. Requirements gathering: Clarify the scope of the project and determine the requirements needed to design the early warning platform.

2. Technical Design: Develop a technical design for the platform, including an architecture diagram and a list of components and technologies.

3. Database Design: Design the database structure necessary to store water level data and relevant information.

4. User Interface Design: Design an intuitive user interface that will enable users to view real-time water level data and receive flood warnings.

5. Prototyping: Create a prototype of the early warning platform to test usability and ensure that the system meets the initial requirements.

6. Development: Develop the platform, integrating the database, user interface and other components according to the technical design.

7. Testing: Test the platform to ensure that it meets the requirements and is free of bugs and other issues.

8. Deployment: Deploy the platform to a production environment and make it available to users.

9. Maintenance: Monitor and maintain the platform to ensure that it is functioning properly.

4.INTEGRATION APPROACH:

1. Identify the type of data that needs to be sent from the IoT sensors to the Early Warning platform.

2. Assess the bandwidth, latency, and other network requirements for transmitting the data from the IoT sensors to the Early Warning platform.

3. Identify the appropriate protocol for sending data from the IoT sensors to the Early Warning platform.

4. Configure the IoT sensors to send the data to the Early Warning platform using the identified protocol.

5. Develop a system to process and store the data from the IoT sensors on the Early Warning platform.

6. Test the data transmission to ensure that the data is correctly received and stored on the Early warning platform.

7. Monitor the performance of the data transmission to ensure the data is correctly transmitted.

8. Optimize the data transmission system for improved performance.