

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

PROJECT DESCRIPTION :

A flood monitoring project typically involves the use of various components and technologies to assess, predict, and respond to flooding events.

COMPONENTS OF WORKING MODEL :

1. Data Collection:

- Sensors: Deploy various types of sensors, such as water level sensors, rain gauges, and weather stations, to collect real-time data on precipitation, water levels, and weather conditions.

- Remote Sensing: Utilize satellite imagery, radar, and drones to gather data over larger geographical areas.

2. Data Transmission:

- Communication Infrastructure: Establish a network for transmitting data from sensors to a central server or data center. This may involve cellular, satellite, or IoT technologies.

- Telemetry: Implement telemetry systems to send data in real time, allowing for immediate analysis and response.

3. Data Analysis:

- Data Processing: Use software and algorithms to process and analyze the collected data, identifying patterns, trends, and potential flood risks.

- Geographic Information Systems (GIS): Integrate geographical data to create flood risk maps and predict flood-prone areas.

4. Warning Systems:

- Early Warning System: Develop a system that can issue alerts to authorities, emergency services, and the public when flood risks are detected.

- Mobile Apps and Public Alerts: Utilize mobile apps and public alert systems to disseminate information and safety instructions to the public.

5. Decision Support:

- Decision Support Systems (DSS): Implement DSS to assist authorities in making informed decisions during flood events, such as evacuations and resource allocation.

- Forecast Models: Use predictive models to forecast flood events and their potential impact.

6. Community Engagement:

- Public Awareness: Educate communities about flood risks, safety measures, and evacuation plans.

- Community Feedback: Establish feedback mechanisms for residents to report flooding incidents and issues.

7. Emergency Response:

- Evacuation Plans: Develop and communicate evacuation plans to ensure the safety of residents in flood-prone areas.

- Resource Allocation: Allocate resources like emergency personnel, equipment, and shelters in response to flood events.

8. Data Storage and Archiving:

- Data Management: Store historical and real-time data for analysis and future planning.

- Disaster Recovery: Implement redundancy and backup systems to ensure data availability in case of system failures.

9. Maintenance and Calibration:

- Regular Maintenance: Ensure sensors and equipment are well-maintained and calibrated to provide accurate data.

- System Upkeep: Continuously update and improve the flood monitoring system to enhance its effectiveness.

10. Collaboration:

- Collaboration with Agencies: Coordinate efforts with meteorological, environmental, and disaster management agencies to enhance the project’s efficiency.

A comprehensive flood monitoring project integrates these components to provide timely warnings, reduce flood-related risks, and enhance disaster preparedness and response in flood-prone areas.

PYTHON PROGRAM :

import RPi.GPIO as GPIO

import time

# Define the GPIO pin connected to the water level sensor

water\_level\_pin = 17

# Set up GPIO

GPIO.setmode(GPIO.BCM)

GPIO.setup(water\_level\_pin, GPIO.IN)

def check\_water\_level():

try:

while True:

water\_level = GPIO.input(water\_level\_pin)

if water\_level == 0:

print("Water level is low. No flooding.")

else:

print("High water level detected. Possible flooding!")

time.sleep(5) # Check the water level every 5 seconds

except KeyboardInterrupt:

print("Monitoring stopped.")

finally:

GPIO.cleanup()

if \_\_name\_\_ == "\_\_main\_\_":

check\_water\_level()