

# FLOOD WATER MANAGEMENT AND EARLY WARNING SYSTEM

*INTERNET OF THINGS - phase 4 - group1 project*

*clg name: Madha Institute of Engineering and Technology*

*name: Elden Nicholas. N*

*reg no: 211221205004*

*naam mudhalvan id : au211221205004*

## **OBJECTIVE:**

- To continuously monitor the temperature and humidity of the environment.
- Utilizing web API, you can use the SMS system to warn individuals.
- To instantly determine the water level.

## **APPLICATION:**

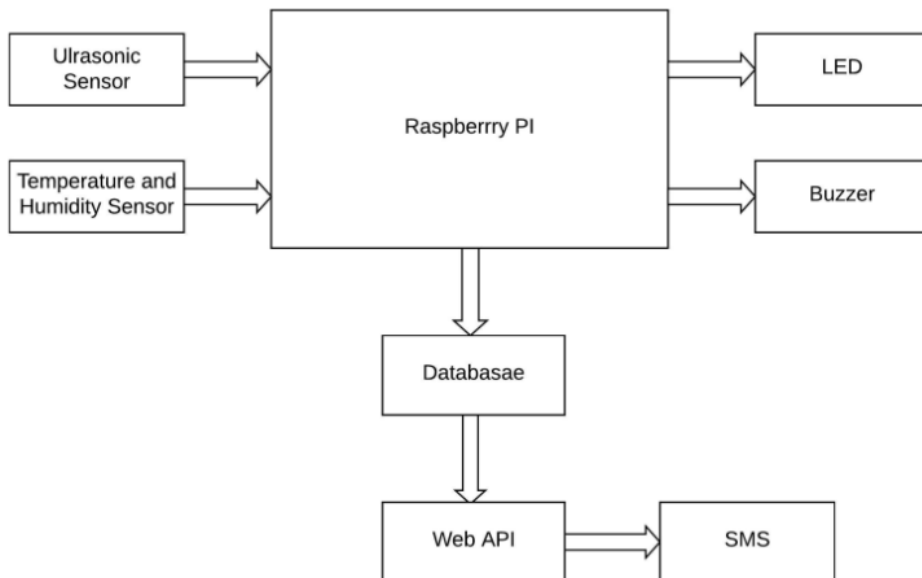
- early warnings of a flood
- provides current information on the temperature, humidity, and water level.

## **LITERARY REVIEW:**

**"IOT-based-Centralized-Remote-Sensing-for-Early-Flood-Detection"** .The goal of this telemetry-based project is to monitor the flood status as soon as possible and transmit an alert on the website in case of danger. Through IOT, the notice delivered may be read globally. The microcontroller is attached to an **ultrasonic sensor** that detects the amount of water in dams or rivers and transmits that data to the microcontroller. **The GPRS** uses the IOT network to transmit the notice over the internet to the website.

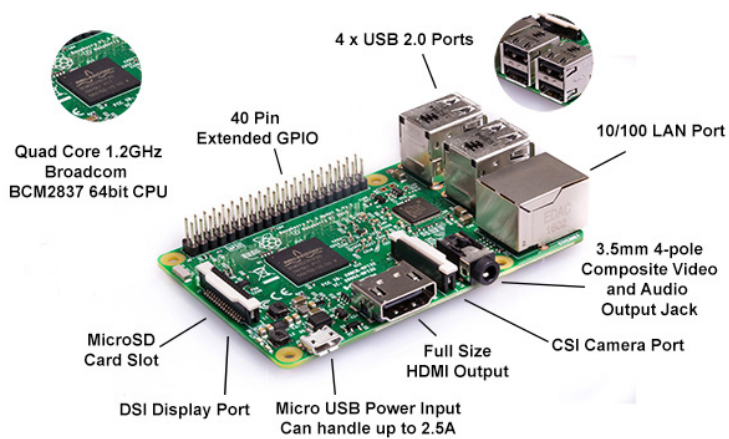
A safe, well-prepared, and less-casualty society is envisioned in **"Flood Monitoring and Early Warning System Using Ultrasonic Sensor"** before, during, and after typhoon disaster. Additionally, the model encourages the use of real-time monitoring systems via created web-based applications and SMS notification systems as an efficient means of information dissemination, particularly in remote locations. Giving the system **two-way communication** allows the community more options in receiving crucial information.

## **BLOCK DIAGRAM:**



as we discussed and provided the hardware components needed in previous phase document:

### ***RASBERRY Pi:***



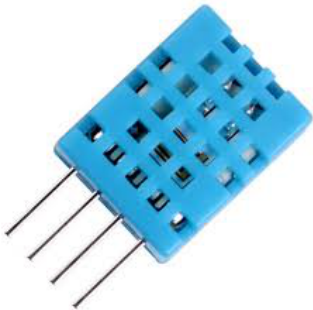
### ***ULTRASONIC SENSOR:***



### ***BUZZER:***

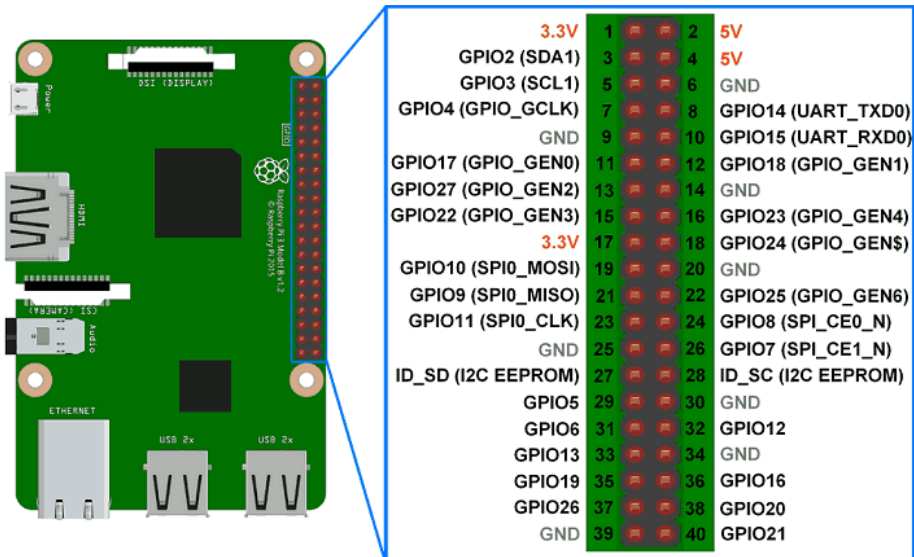


TEMPERATURE AND HUMIDITY SENSOR(DHT11):

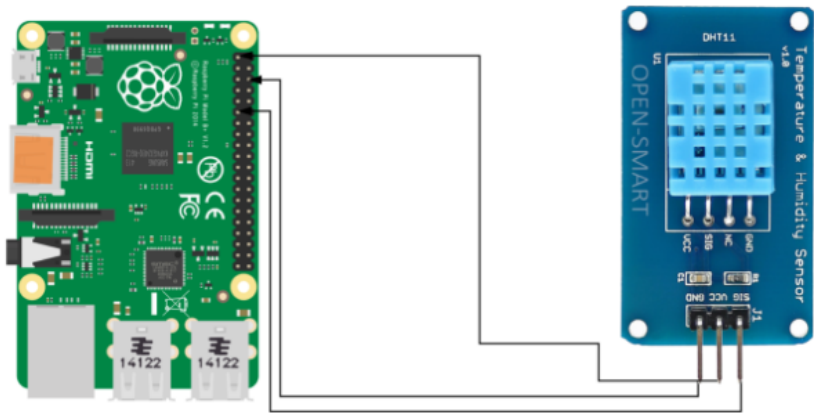


DESIGN AND IMPLEMENTATION:

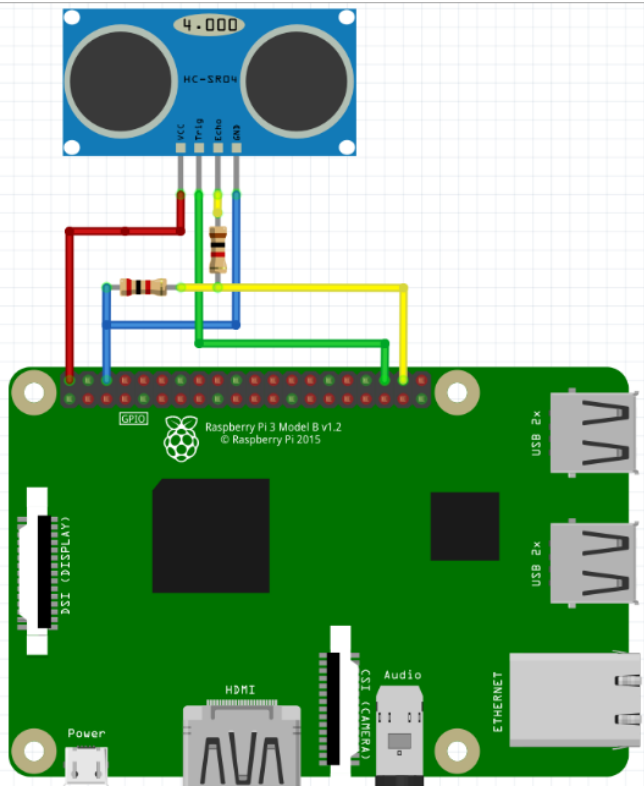
RASPBERRY PI:



DTH11 SENSOR:



ULTRASONIC SENSOR WITH PI:



in the next page attached source code of a simplified application in Arduino and Python

```

import conf
from boltiot import Sms, Email, Bolt
import json, time
intermediate_value = 55
max_value = 80
mybolt = Bolt(conf.API_KEY, conf.DEVICE_ID)
sms = Sms(conf.SID, conf.AUTH_TOKEN, conf.TO_NUMBER, conf.FROM_NUMBER)
mailer = Email(conf.MAILGUN_API_KEY, conf.SANDBOX_URL, conf.SENDER_EMAIL,
conf.RECIPIENT_EMAIL)
def twillo_message(message):
    try:
        print("Making request to Twilio to send a SMS")
        response = sms.send_sms(message)
        print("Response received from Twilio is: " + str(response))
        print("Status of SMS at Twilio is : " + str(response.status))
    except Exception as e:
        print("Below are the details")
        print(e)
def mailgun_message(head,message_1):
    try:
        print("Making request to Mailgun to send an email")
        response = mailer.send_email(head,message_1)
        print("Response received from Mailgun is: " + response.text)
    except Exception as e:
        print("Below are the details")
        print(e)
while True:
    print ("Reading Water-Level Value")
    response_1 = mybolt.serialRead('10')
    response = mybolt.analogRead('A0')
    data_1 = json.loads(response_1)
    data = json.loads(response)
    Water_level = data_1['value'].rstrip()
    print("Water Level value is: " + str(Water_level) + "%")
    sensor_value = int(data['value'])
    temp = (100*sensor_value)/1024
    temp_value = round(temp,2)
    print("Temperature is: " + str(temp_value) + "°C")
    try:
        if int(Water_level) >= intermediate_value:
            message ="Orange Alert!. Water level is increased by " +str(Water_level) +
"% at your place. Please be Safe. The current Temperature is " + str(temp_value) + "°C."
            head="Orange Alert"
            message_1="Water level is increased by " + str(Water_level) + "% at your place.
Please be Safe. The current Temperature is " + str(temp_value) + "°C."
            twillo_message(message)
            mailgun_message(head,message_1)
        if int(Water_level) >= max_value:
            message ="Red Alert!. Water level is increased by " + str(Water_level) + "% at your place
Please Don't move out of the house. The Current Temperature is " + str(temp_value) +
"°C"
            head="Red Alert!"
            message_1="Water level is increased by " + str(Water_level) + "% at your place.
Please Don't move out of the house. The Current Temperature is " + str(temp_value)
+ "°C."
            twillo_message(message)
            mailgun_message(head,message_1)

    except Exception as e:
        print ("Error ocured: Below are the details")

```

```
print (e)  
time.sleep(15)
```

## TEAM MEMBERS

211221205001  
211221205002  
211221205003  
211221205004  
211221205005  
211221205006