COLLEGE CODE:9512

COLLEGE NAME:JP COLLEGE OF ENGINEERING

DEPARTMENT:ECE

PROJECT CODE:Proj_211933_Team_1

FLOOD MONITROING AND EARLY WARNING SYSTEM

PHASE 4:

TEAM MEMBERS:

1.JAYA MAHA VARSHINI.S(au951221106014)

2.RAMYA.G(au951221106031)

3.KALPANA DEVI(au951221106017)

4.RISWANA FATHIMA(au951221106034)

5.THANGA DURACHI (au951221106050)

DEVELOPMENT PART-2:

THE FLOOD MONITORING AND EARLY WARNING SYSTEM PROJECT HAS BEEN DEVELOPED USING WEB TECHNOLOGY CODE TO SHOW THE REAL TIME OUTPUT OF DETECTING THE WATER LEVEL AND DISPLAYING THE

LEVEL OF WATER BY RECEIVING THE LEVEL OF WATER THROUGH THE SENSORS AND DISPLAYING IT WITH A WARNING USING ARDUINO.THE DEVELOPED MODEL OF THE PROJECT AND THE WAY OF DISPLAYING THE OUTPUT WITH WEB TECHNOLOGY CODE EVERYTHING WAS SHOWN BELOW.

WEB DEVELOPMENT CODE :

```
//slave
#include <Wire.h>
#define PIR 7
#define LDR A0
#define bulb 8
#define buzzer 9
#define SLAVE_ADDR 9
#define ANSWERSIZE 5

#define led1 4
// led1 red
#define led2 3
// led2 blue
#define led3 5
```

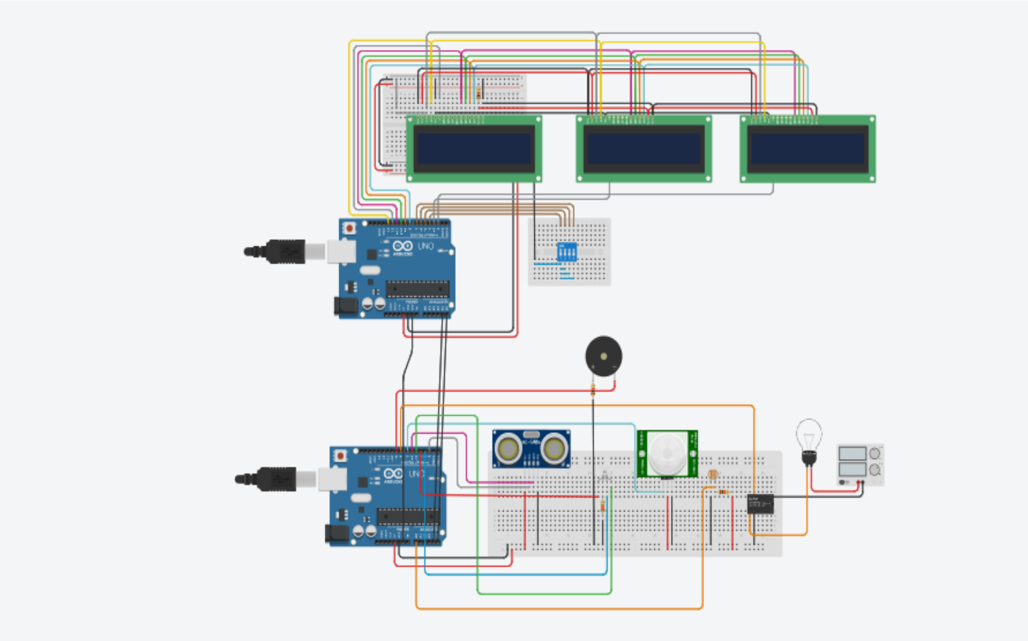
```
// led3 green

int trigPin = 2;
int echoPin = 6;
int val = 0;
int duration;
float distance;
float meter;

union
{
  float x;
  byte myData[4];

}data;

void setup() {
  Wire.begin(SLAVE_ADDR);                // join i2c bus with
address #8
  Wire.onRequest(sendEvent); // register event
  Serial.begin (9600);
  pinMode(trigPin, OUTPUT);
  digitalWrite(trigPin, LOW);
  delayMicroseconds(2);
  pinMode(echoPin, INPUT);
  pinMode(PIR,INPUT);
  pinMode(LDR,  INPUT);
  pinMode(bulb, OUTPUT);
  pinMode(led1, OUTPUT);
  pinMode(led2, OUTPUT);
  pinMode(led3, OUTPUT);
  pinMode(buzzer, OUTPUT);

  delay(6000);
  Serial.println("Distance:");
  Serial.println("Water detect:");
}

void loop() {
  delay(50);

  int i = 0; // This can be set to any value to change the frequency.
  digitalWrite(trigPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin, LOW);
  duration = pulseIn(echoPin, HIGH);
  distance = (duration/2) / 29.1;
  data.x = distance;
  Serial.print(distance);
  Serial.print("cm");
  Serial.print("\n"); // This is required to calculate the distance in centimeters.

  val = digitalRead(PIR);
  if (val == HIGH && distance <= 30) {
    RGB_color(255, 0, 0); // Red
    digitalWrite(buzzer, HIGH);
}
```

```
  else {
    digitalWrite(buzzer, LOW);

  }
if (distance <= 30) {
    RGB_color(255, 0, 0); // Red
}
  else {

  }
  if (distance <= 100 && distance > 30) {
      RGB_color(255, 255, 0); // Yellow

}
  else {

  }
  if (distance > 100) {
      RGB_color(0, 255, 0); // Green
}
```

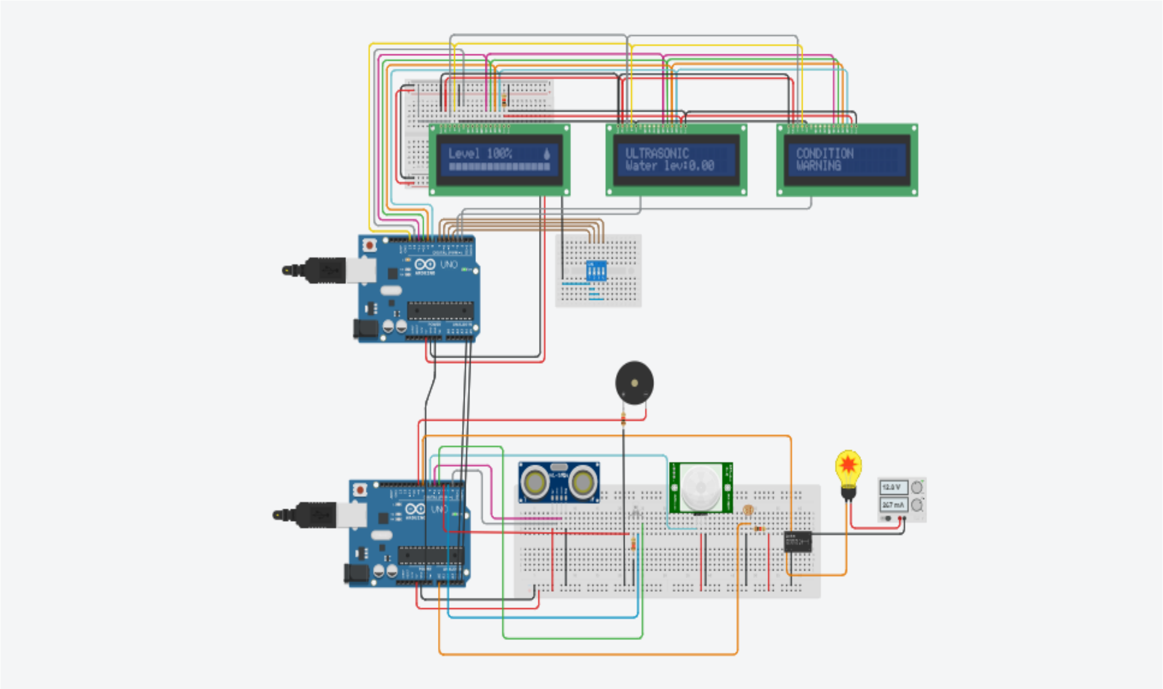
```
  else {

  }

  if (analogRead(A0) > 500 && val == HIGH && distance
<= 30) {
      digitalWrite(8, LOW);
}
  else {
    digitalWrite(8, HIGH);
  }


}

void RGB_color(int red_light_value, int
green_light_value, int blue_light_value)
{
  analogWrite(led1, red_light_value);
  analogWrite(led3, green_light_value);
  analogWrite(led2, blue_light_value);
```

```
}

// function that executes whenever data is requested by master
// this function is registered as an event, see setup()
void sendEvent()
  {
   Wire.write(data.myData, sizeof(data.myData));//float distance goes to Master as 32-bit binary32 data


  /*
  while (1 < Wire.available()) // Loop through all but the last
  {
    float distance = Wire.read(); // Receive byte as a character
    Serial.print(distance); // Print the distance values
  }
  int x = Wire.read(); // Receive byte as an integer
  Serial.println(x);
  delay(10); */

}
```

DEVELOPED MODEL:



OUTPUT AFTER RECEIVING THE WATER LEVEL:



CONCLUSION:

THUS THE PROJECT WAS DEVELOPED TO GET THE WATER LEVEL AND DISPLAY THE WARNING BASED ON THE LEVEL OF WATER USING SENSORS.