**Problem Statement : REAL- TIME RIVER WATER QUALITY MONITORING AND CONTROL SYSTEM**

**Domain :**

Internet of Things

**ASSIGNMENT 1**

**By,**

**M.AYYANAR - 19EC011**

**M.ESAKKIMUTHU - 19EC022**

**M.KAVIARASAN – 19EC046**

**A.AROKIYAJEYAPRAKASH – 19EC008**

**CODE:**

const int pingPin = 10; const int ledUS

= 2; const int light =

7; const int pir = 4;

#define photoSensor A0 #define buzzer 3 int const PINO\_SGAS = A5; int const ledGas =

8; int const button = 5; int const motor =

13; void setup()

{

pinMode(ledUS, OUTPUT); pinMode(light, OUTPUT); pinMode(buzzer, OUTPUT); pinMode(ledGas, OUTPUT); pinMode(motor, OUTPUT); pinMode(pir, INPUT); pinMode(button, INPUT); pinMode(photoSensor,

INPUT); Serial.begin(9600);

}

void loop()

{

long duration, cm; int valLight = analogRead(photoSensor); int valPIR= digitalRead(pir); int valGAS = analogRead(PINO\_SGAS); valGAS = map(valGAS, 300, 750, 0, 100); int valBt = digitalRead(button); pinMode(pingPin, OUTPUT); digitalWrite(pingPin, LOW); delayMicroseconds(2); digitalWrite(pingPin, HIGH); delayMicroseconds(5); digitalWrite(pingPin, LOW); pinMode(pingPin, INPUT); duration = pulseIn(pingPin, HIGH); cm = microsecondsToCentimeters(duration); if(cm < 336){

digitalWrite(ledUS, HIGH);

}else{ digitalWrite(ledUS, LOW);

}

if(valLight < 890){

digitalWrite(light,

HIGH);

}else{ digitalWrite(light, LOW);

}

if(valPIR == 1){

digitalWrite(buzzer,

HIGH);

}else{ digitalWrite(buzzer, LOW);

}

if(valBt == 1){

digitalWrite(motor,

HIGH);

}else{

digitalWrite(motor, LOW);

}

if(valGAS > 20){

digitalWrite(ledGas,

HIGH);

}else{ digitalWrite(ledGas, LOW);

}

Serial.print(valPI

R);

Serial.println();

}

long microsecondsToCentimeters(long

microseconds) { return microseconds / 29 / 2;

}

**Simulation :**

