IOT BASED SMART WATER MANAGEMENT
Phase 5: DOCUMENTATION PART

ABSTRACT:

Water is one of the fundamental resources that aid life and there are speculations that estimate at 2025 almost half of the urban population will live under short supply and water stress. With the usage of new technological advancements in IoT (Internet of Things) powered smart devices for water management, it can become a worthy implementation towards avoiding the predicted water depletion. In the past years up until recently, water monitoring and management were manually carried out with intensive power requirements and high capital expense with low efficiency recorded. Overflow of water overhead tanks in residential, commercial, cooperate and educational settings, as well as broken pipes resulting in spillage, contribute to wastage at large. Regular reservoirs for water cannot monitor nor give analytics and automated water level detection in the tank.
Vandalization or transmission blockages on distributions pipes may take so long to discover. The proposed model addresses problems mentioned above by the application of portable smart systems with interoperability and easily configurable to handle automated management of water supply with energy efficiency and a reduction in power cost in both homes and enterprise environment within smart cities as well as reduction of the rate of building degradation as a result of overflow from overhead tanks. Our model also integrates the application of Natural Language Processing for speech recognition as an alternate medium useful in operating the system.

OBJECTIVES:

The primary objective of smart water management is reasonable and sustainable usage and recycling of water resources. Growing population, increasing environmental issues and pressure on the food and agriculture sector make water even a more precious asset.

In this respect, water management technologies and activities pursue the following objectives:

1. **Reduce wasting water used in high volumes for agriculture, manufacturing, power production. It implies the introduction of high-tech practices like precision farming, smart irrigation, crop water management, real-time water metering and other applications of Internet of Things in agriculture. Learn about our agriculture software development services.**
2. **Improve water quality and prevent contamination by chemical waste and natural pollution such as acidification. In order to improve and maintain the quality of water, companies use sensors and IoT technology for real-time monitoring and control.**
3. **Enhance the efficiency of water systems such as water collectors, treatment plants, distribution mains and wastewater recycling centers. Using IoT and data solutions for asset management, companies can keep important measurements such as water pressure, temperature, flow, etc. at hand, integrate predictive maintenance and avoid breakage and downtime of equipment.**
4. **Implement leakage control by using smart water management devices equipped with leak and moisture sensors. Given that almost $3 billion are spent on fixing the damage caused by leakage yearly, leakage control is essential to keep water resources and budgets safe.**

5. **Practice consumption monitoring via IoT-based water management systems. It helps to optimize and keep under control the usage of water resources at different levels — households, communities, countries and the whole planet.**

INTRODUCTION:

Smart water management is the activity of planning, developing, distributing and managing the use of water resources using an array of IOT technology. Which are designed to increase transparency, and make more reasonable and sustainable usage of these water resource. Water Level Monitoring System in Water Tanks can be used in Houses to avoid overflow and wastage of water. In this project, one of the important parts is the High Sensitivity Water Sensor.

It is easy to use, light in weight, compact in size, high identification, and detection of water, droplets.

The water sensor has many different applications like sensing of rainfall, leakage of water and water level detection. This sensor works by having a series of exposed mark lines(yellow lines that can be seen in the image above) connected to GND. The sensor has a low resistance resistor.

The resistor keeps the sensor value low till the water shorts the sensor. This sensor then changes the water detected to analog signal/ digital signal with the help of Bolt IOT, and those analog values are used in the program, to achieve the function of water level monitoring and other similar applications. The good point of this sensor is that it uses very less power and has higher sensitivity.

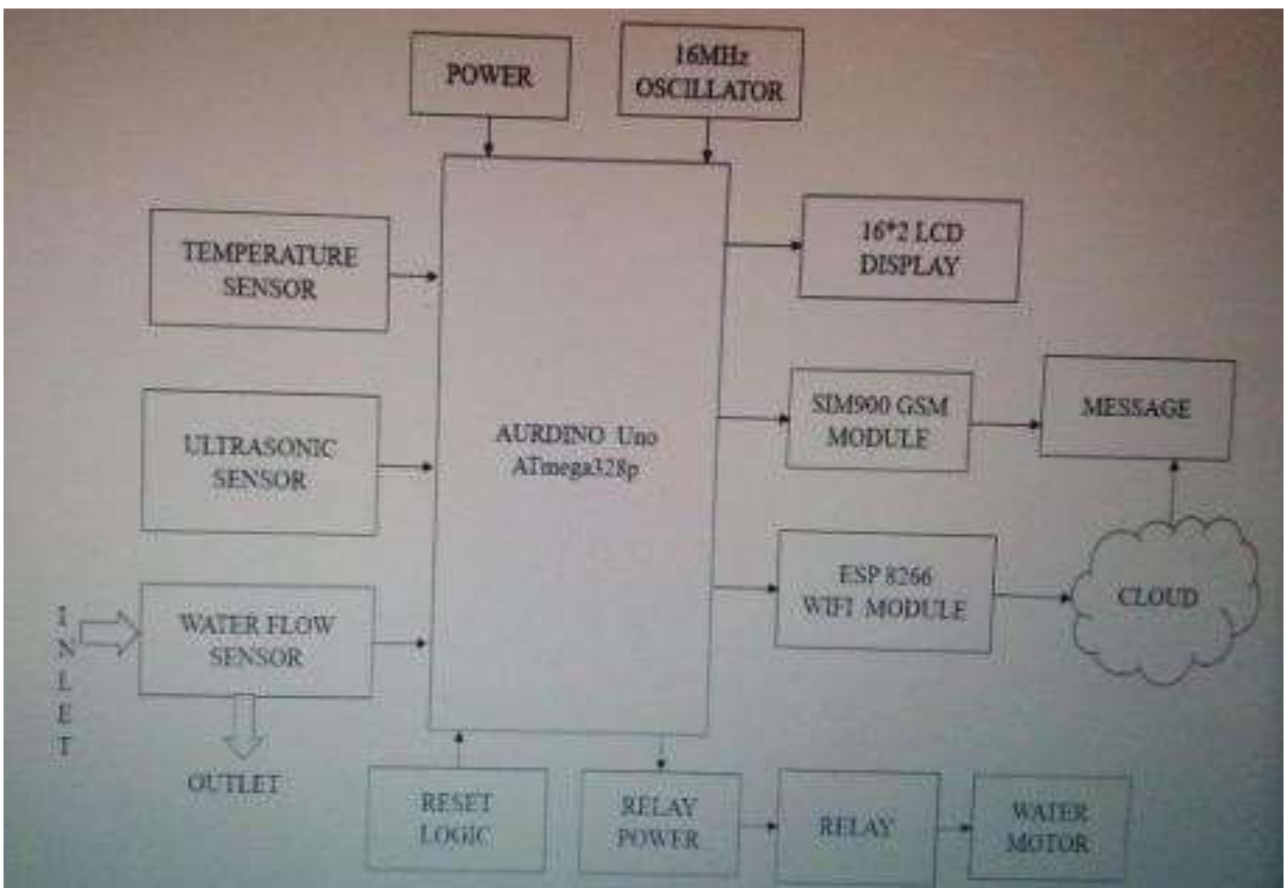
In the 21st century, there were lots of inventions, but at the same time were pollutions, global warming and so on are being formed, because of this there is no safe drinking water for the worlds pollution. Nowadays, water quality monitoring in real time faces challenges because of global warming limited water resources, growing population, etc. Hence there is need of developing better methodologies to monitor the water quality parameters in real time. The monitoring of water quality is extremely important for maintaining the safety of water resources used for various purposes.

IoT is progressing with millions of things connecting each day to generate large amount of information resulting in useful future actions.To ensure the safe supply of drinking water the quality should be monitored in real time for that purpose new approach IOT (Internet of Things) based water quality monitoring has been proposed. In this project, we will implement the design of IOT for monitoring system that monitors the quality of water in real time.

This system consists some sensors which measure the water quality parameter.The real-time monitoring of water resources information will benefit the water resources management department and the public. The

primary concept of real-time IOT based water resources information system is to provide comprehensive and accurate information. The system
isdeveloped through defining some explicit water resource parameters then, Water level are defined for water measure & management,
followed
by a sensor network for water resources information monitoring is constructed based on IOT.

BLOCK DIAGRAM:



BASIC ELEMENTS

A. Micro Controller

TheAtmega328isa one of thevery popular microcontroller chip produced by Atmel It is an 8 - bit microcontroller that has32Kof flash memory, 1K of EEPROM, and2K ofSRAM.The Atmega328 is one of the microcontroller chips
that are used with the popular Arduino boards.This microcontroller hasanalogpin and digital pin for easy interface of the
=MicrocontrollerOperating Voltage: – 1.8 - 5.5V23 Programmable I/OLinesTwo 8-bit Timer/CountersReal Time
Counter with Separate OscillatorSix PWM Channels6- channel 10- bit ADC .

B. .IOT Module

Wi-Fi Direct (P2P), soft-APIntegrated TCP/IP protocol stack+19.5dBm output power in802.11b modeSupports antennadiversityPower down leakage current of < 2msStandby power consumption Operating Voltage : 3.3V

This is exceed and it isburn the esp module.GND is connected to the ground terminal.Rx pin is the receiver pin UART serial communication The Tx pin is a transmitter.GPIO general purpose input and output.Reset pin reset the module apply in 3.3v. the CH-PD pin configure channel.

C. .nRf transreceiver

The nrfwireless transreceiver is 8 pin of the operation. GND pin it isalso used to for the ground terminal.Vccis a power supply operated bythe voltagerange is 1.9v to the 3.6v and it is mostly apply the 3v . The CE pin is a
select the mode of operation either is operated by transmit data or receive a data.CSN it is used to for the enable the SPI chip.SPI provise is high the clock
is enableand low the clock is disable.MOSI transmit a data from user module to the external circuit.MISO receive a data from the external circuit
or modulethen finally IRQ ias interrupt request pin it isdoes not need to connect Fig 2.3 nrf 2401L Connection in Wireless Communication.

This communication is called as Serial Peripheralinterface (SPI) . It has following pin name Serial Peripheral Interface, or SPI is a very common communication protocol used for two-way communication between two devices. A standard SPI bus consists of 4 signals, Master Out Slave In (MOSI) Master In Slave Out (MISO), the clock (SCK), and Slave Select (SS)An SPI bus has one master and one or more slavesThe master can talk to any slave on the bus, but each slave can only talk to the master.Each slave on the bus must haveit's own unique slave select signal.The master uses the slave select signals to select which h slave it will be talking to.

D. Salt Sensor

It is used to monitoring the salt content of the sewage water and communicate with microcontroller for posting this information to internet. It has consists of two rods one is reference rod and measuring rod. The voltage is given to the reference rod and the conducting current passes to measuring rod. The voltage present in the measuring rod is proportional to the salt content of the water.

E. pH Sensor

pH sensor used to determine the pH value content in the water. The pH value range from the acidity – Neutral – Alkaline. It has two rod to measure the value of the pH value in the water. The pH meter is used for thequality check if water is safe for drinking. A balanced pH level is very important for human health; it should be approximately equal to 7. It gives Full range pH reading from 1 to 5 voltage scale range and gives a Single reading.

PROJECT RESOURCES:

Hardware components

1. **Sensors Used With Boltduino/Arduino**
2. **5V Relay**
3. **I2C LCD**
4. **Boltduino**

4. 9V Battery

5. Bolt Wifi Module

6. IRF540 MOSFET

7. Water Flow Sensor

8. Ultrasonic Sensor X 2

9. 1N4007 Rectifier Diode

10. 12V DC Solenoid Valve

11. Water Lifting Submersible Pump

12. 4-way Capacitive Touch Switch Module

13. 3-6 V Mini Micro Submersible Water Pump

14. LM35 IC (Temperature sensor)

2. Sensors Used With Boltduino/Arduino

1. Nodemcu

2. Piezo Buzzer

3. IR Sensor X 2

4. DC Motors X 2

5. 12V DC Adapter

6. TCS3200 Color Sensor

7. Capacitive Touch Sensor

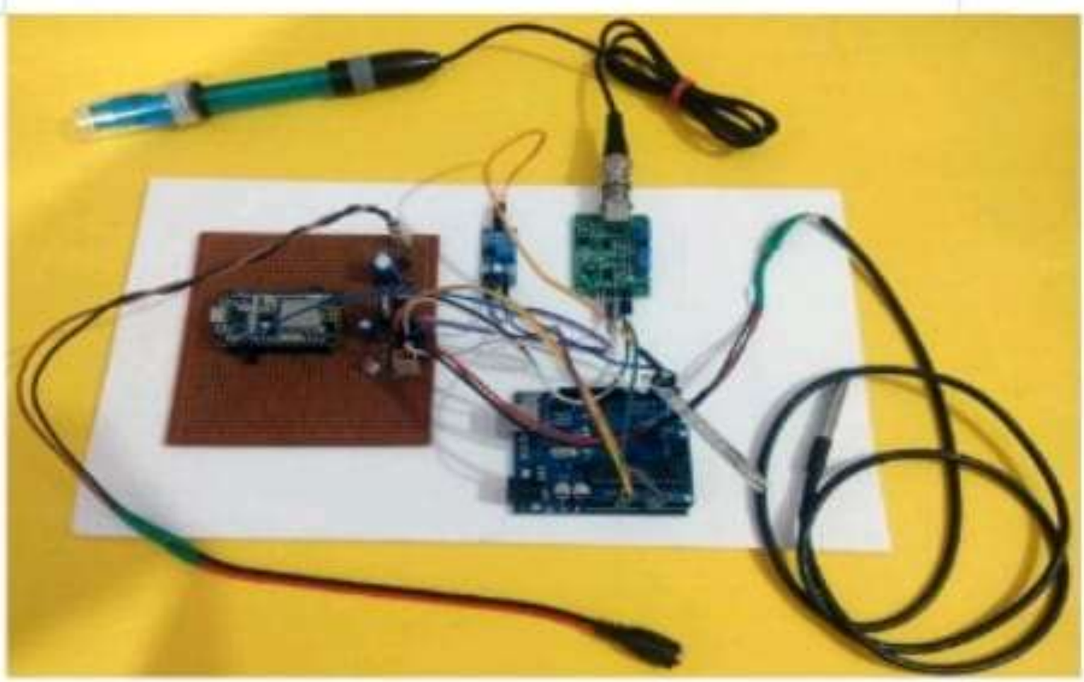
8. ESP8266 Motor Driver Shield

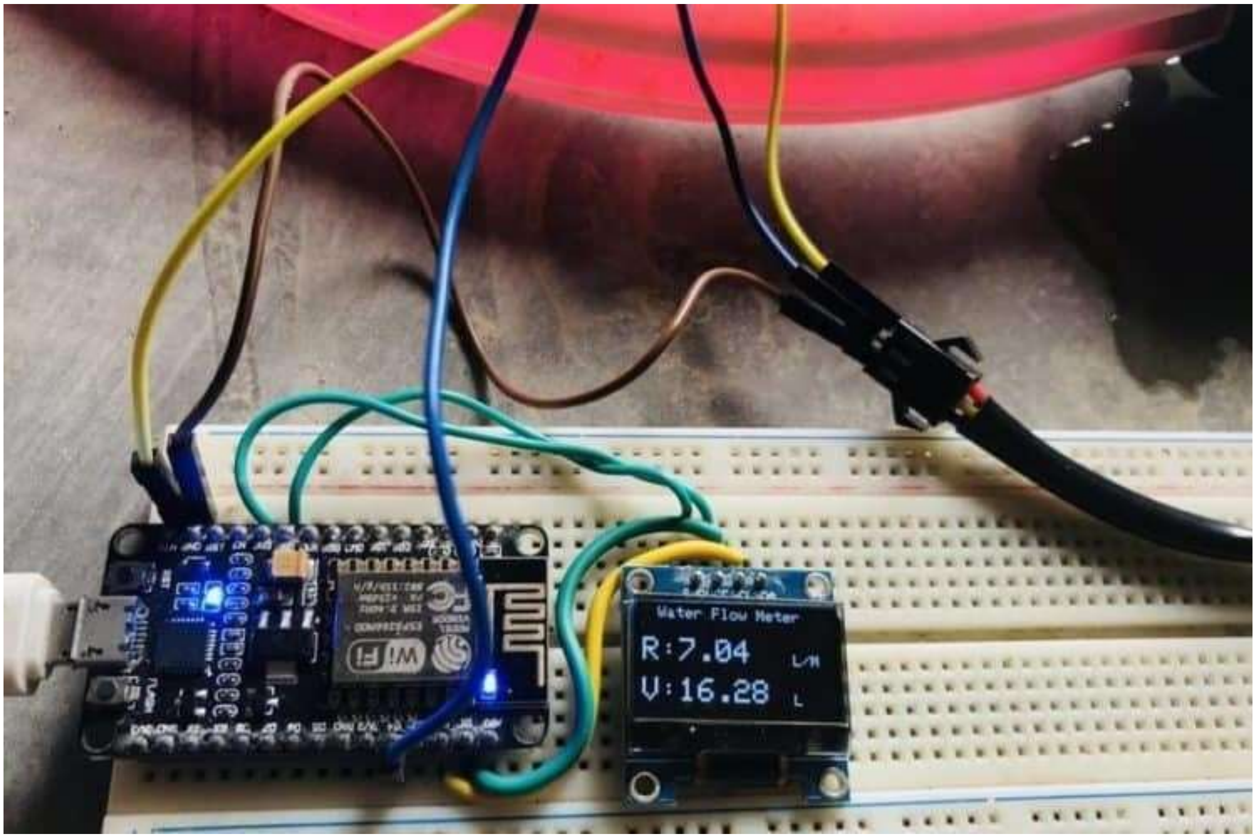
9. Analog Multiplexer IC – CD4051

b) Software apps and online services

1.

1. **Arduino IDE**
2. **Bootstrap Studio**
3. **Spyder (Anaconda)**
4. **Twilio**
5. **Canva**
6. **Hostinger**
7. **Integromat**
8. **Mega Creator**
9. **Pichon (Icons8)**





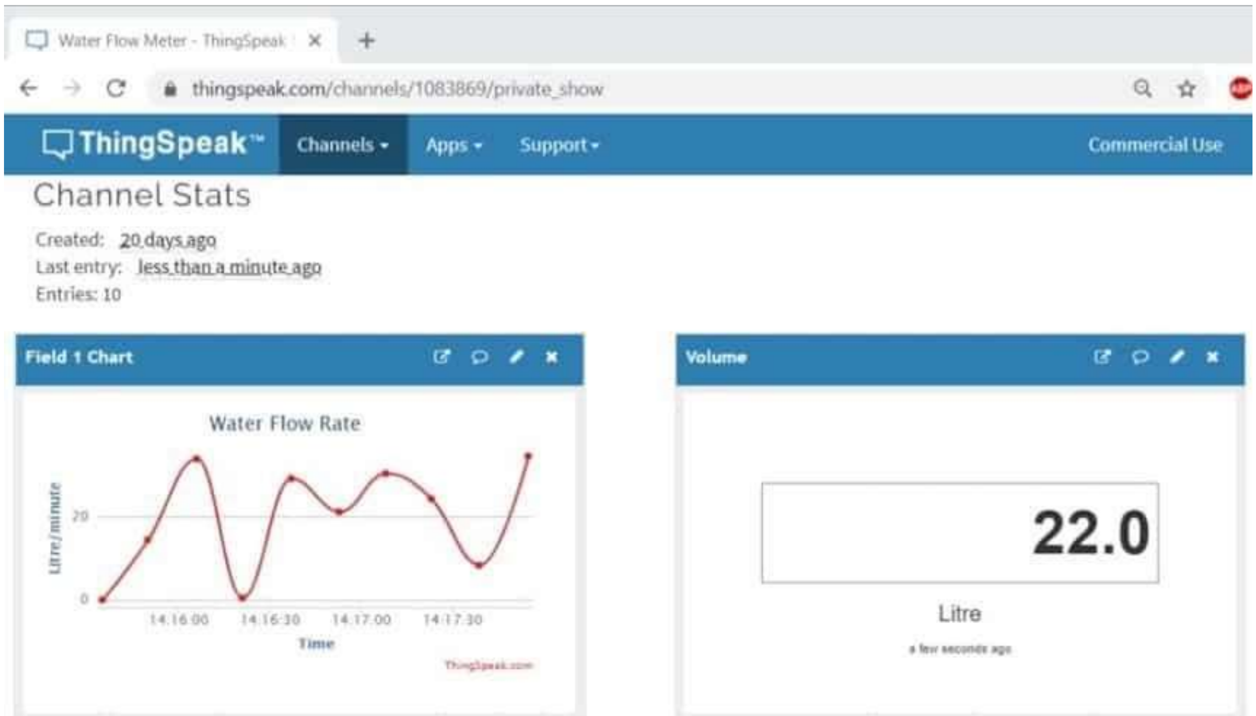
PROGRAM:

```
#include <ESP8266WiFi.h>
#include <SPI.h>
#include <Wire.h>
#include <Adafruit_GFX.h>
#include <Adafruit_SSD1306.h>

#define SCREEN_WIDTH 128 // OLED display width, in pixels
#define SCREEN_HEIGHT 64 // OLED display height, in pixels
#define OLED_RESET -1       // Reset pin # (or -1 if sharing Arduino reset pin)

Adafruit_SSD1306 display(SCREEN_WIDTH, SCREEN_HEIGHT, &Wire, OLED_RESET);

String apiKey = "KBD1JSZTUKCXJ15V"; // Enter your Write API key from ThingSpeak
const char *ssid = "Alexahome";      // replace with your wifi ssid and wpa2 key
const char *pass = "loranthus";
const char* server = "api.thingspeak.com";

#define LED_BUILTIN 16
#define SENSOR  2

long currentMillis = 0;
long previousMillis = 0;
int interval = 1000;
boolean ledState = LOW;
float calibrationFactor = 4.5;
volatile byte pulseCount;
byte pulse1Sec = 0;
float flowRate;
```

```
unsigned long flowMilliLitres;
unsigned int totalMilliLitres;
float flowLitres;
float totalLitres;

void IRAM_ATTR pulseCounter()
{
  pulseCount++;
}

WiFiClient client;

void setup()
{
  Serial.begin(115200);
  display.begin(SSD1306_SWITCHCAPVCC, 0x3C); //initialize with the I2C addr 0x3C (128x64)
  display.clearDisplay();
  delay(10);

  pinMode(LED_BUILTIN, OUTPUT);
  pinMode(SENSOR, INPUT_PULLUP);

  pulseCount = 0;
  flowRate = 0.0;
  flowMilliLitres = 0;
  totalMilliLitres = 0;
  previousMillis = 0;

  attachInterrupt(digitalPinToInterrupt(SENSOR), pulseCounter, FALLING);
}

void loop()
{
  currentMillis = millis();
  if (currentMillis - previousMillis > interval)
  {

    pulse1Sec = pulseCount;
    pulseCount = 0;

    // Because this loop may not complete in exactly 1 second intervals we calculate
    // the number of milliseconds that have passed since the last execution and use
    // that to scale the output. We also apply the calibrationFactor to scale the output
    // based on the number of pulses per second per units of measure (litres/minute in
    // this case) coming from the sensor.
    flowRate = ((1000.0 / (millis() - previousMillis)) * pulse1Sec) / calibrationFactor;
    previousMillis = millis();

    // Divide the flow rate in litres/minute by 60 to determine how many litres have
    // passed through the sensor in this 1 second interval, then multiply by 1000 to
    // convert to millilitres.
    flowMilliLitres = (flowRate / 60) * 1000;
    flowLitres = (flowRate / 60);

    // Add the millilitres passed in this second to the cumulative total
    totalMilliLitres += flowMilliLitres;
    totalLitres += flowLitres;

    // Print the flow rate for this second in litres / minute
    Serial.print("Flow rate: ");
    Serial.print(float(flowRate));  // Print the integer part of the variable
    Serial.print("L/min");
    Serial.print("\t");       // Print tab space

    display.clearDisplay();

```

```
   display.setCursor(10,0); //oled display
   display.setTextSize(1);
   display.setTextColor(WHITE);
   display.print("Water Flow Meter");

   display.setCursor(0,20); //oled display
   display.setTextSize(2);
   display.setTextColor(WHITE);
   display.print("R:");
   display.print(float(flowRate));
   display.setCursor(100,28); //oled display
   display.setTextSize(1);
   display.print("L/M");

   // Print the cumulative total of litres flowed since starting
   Serial.print("Output Liquid Quantity: ");
   Serial.print(totalMilliLitres);
   Serial.print("mL / ");
   Serial.print(totalLitres);
   Serial.println("L");

   display.setCursor(0,45); //oled display
   display.setTextSize(2);
   display.setTextColor(WHITE);
   display.print("V:");
   display.print(totalLitres);
   display.setCursor(100,53); //oled display
   display.setTextSize(1);
   display.print("L");
   display.display();
  }

  if (client.connect(server, 80)) // "184.106.153.149" or api.thingspeak.com
  {
   String postStr = apiKey;
    postStr += "&field1=";
    postStr += String(float(flowRate));
    postStr += "&field2=";
    postStr += String(totalLitres);
    postStr += "\r\n\r\n";

   client.print("POST /update HTTP/1.1\n");
   client.print("Host: api.thingspeak.com\n");
   client.print("Connection: close\n");
   client.print("X-THINGSPEAKAPIKEY: " + apiKey + "\n");
   client.print("Content-Type: application/x-www-form-urlencoded\n");
   client.print("Content-Length: ");
   client.print(postStr.length());
   client.print("\n\n");
   client.print(postStr);

  }
   client.stop();
}
```

OUTPUT:



CONCLUSION:

An internet-based approach to measuring water quality and delivery systems on a real-time basis. The results of the various parameters of water quality are verified that the system achieved the reliability and feasibility of using it for the actual monitoring purposes. The WSN network will be developed in the future comprising of more number of nodes to extend the coverage range. In our proposed system, water level can be monitored continuously from anywhere using web browser. Motor can be controlled automatically full smart automation is achieved. It is a robust system & small in size. This Project useultrasonic sensors which provide more accurate and calibrated information for water level in tank. An electromagnetic box is used to drop the chlorine power in the tank by automated system and show the various parameter of water in a web browser that can be viewed any whereby user.