PROJECT DEFINITION:

In this project we are going to measure the water consumption and going to measure the contamination level in the water by implementing IOT sensors. And these measured values will be used to know whether we can use the water and how much we can use to reduce the consumption in terms of to save the wastage of water. If the water is not contaminated, we can use it for general purpose but with the amount we needed and if the water was contaminated then we can use it for any other purposes. These are all done to promote the water conservation. These can be implemented anywhere like home, public places, Urban areas, Rural areas, Industry, Agriculture, etc. These data will be visible to anyone who implemented the sensor. If it is in public place, then all the public can see the data of water consumption and how clean it is and if it is in home then it can be seen by who implemented the sensor. This project includes defining objectives, designing the IoT sensor system, developing the data-sharing platform, and integrating them using IoT technology and Python.

Project Objectives:

*In this Project we can measure the level of water and pH Value for analysing whether the water is contaminated or not.

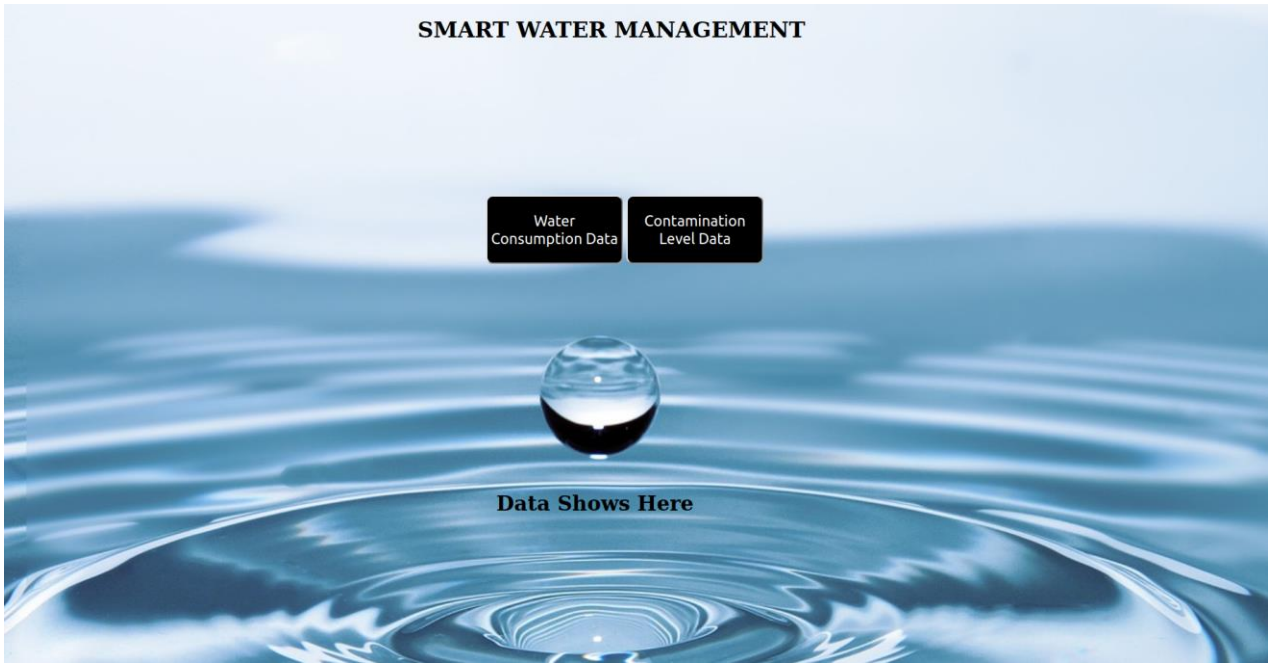
*By doing this we can save water for our future.

*If we use the water by seeing the water level we can use it for useful purposes and save water.

*When we check the water is contaminated or not , we can use the water based on the contamination of water.

Real Time data Showing Platform:

*We use a Web based Real time data visualization in which when we click a button we can check the water level and the contamination of water.




SMART WATER MANAGEMENT
Water Consumption Data
Contamination Level Data
Water Level: 241.09470 cm

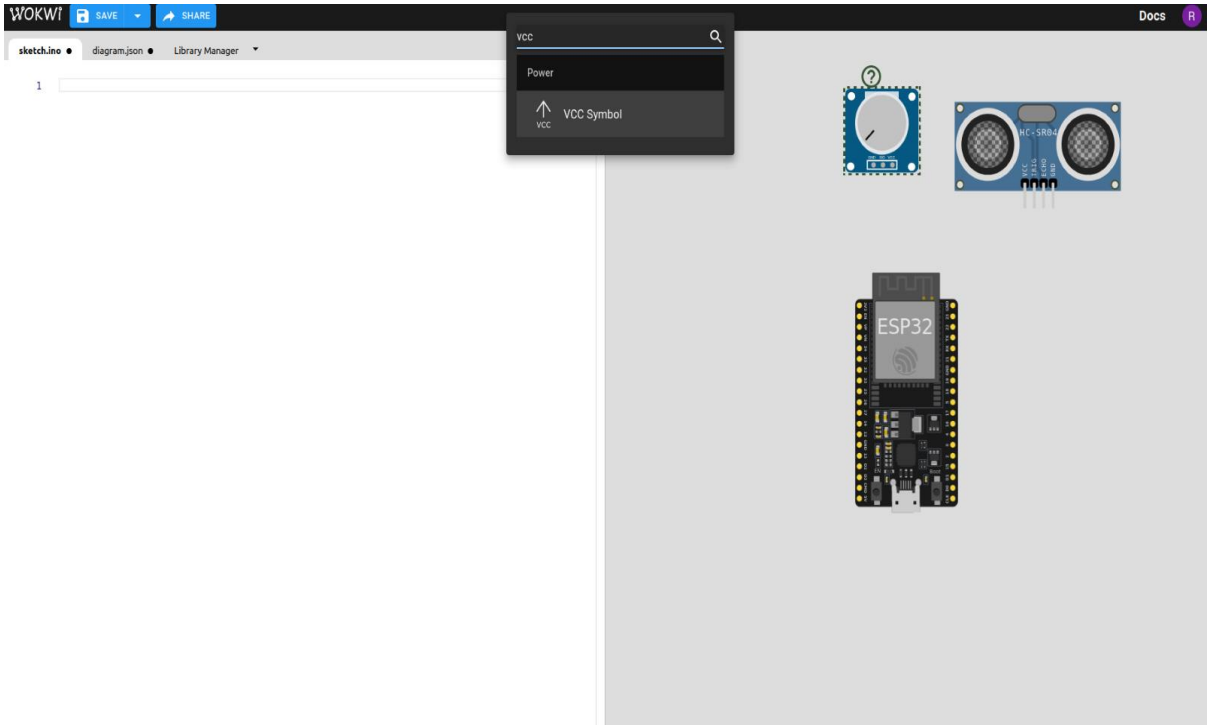

SMART WATER MANAGEMENT
Water Consumption Data
Contamination Level Data
Contamination Level: 19.82000, Status: Contaminated

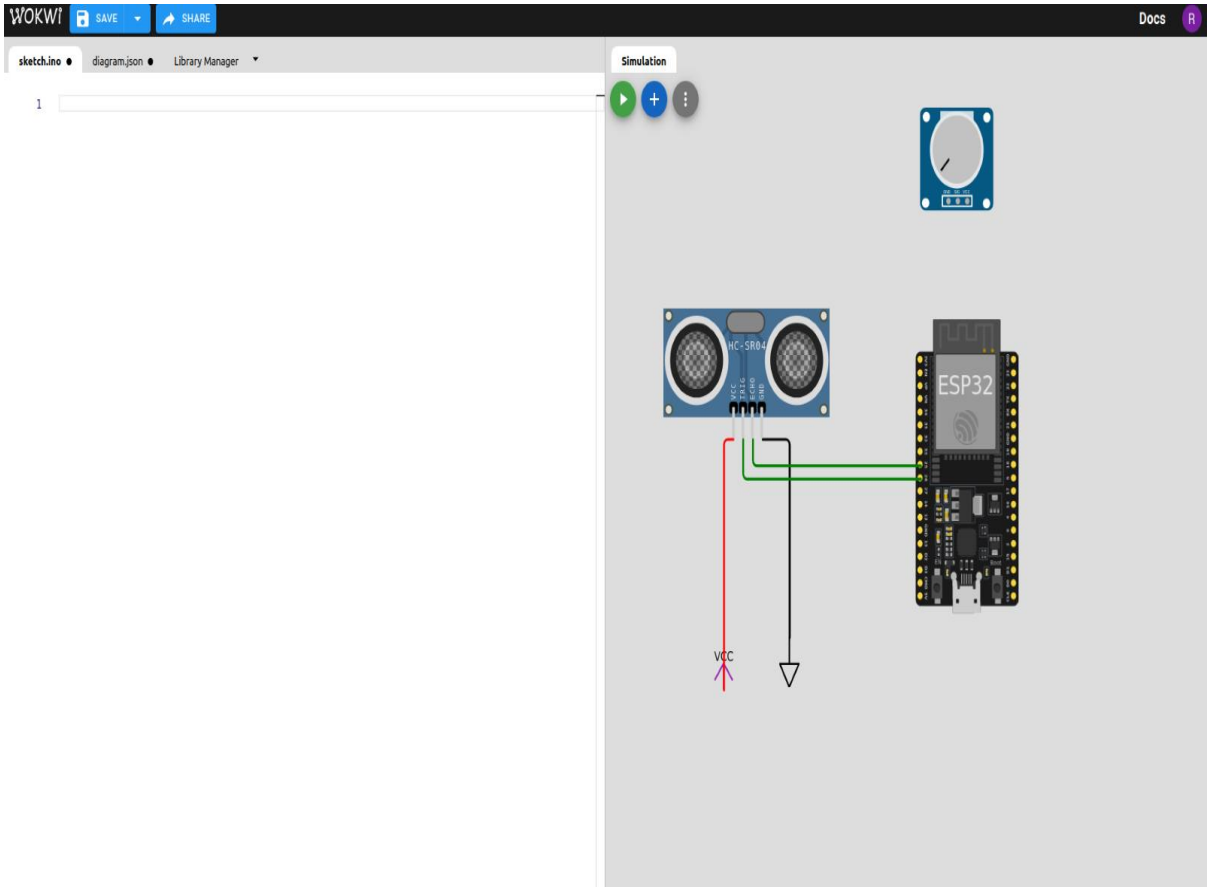
Program:

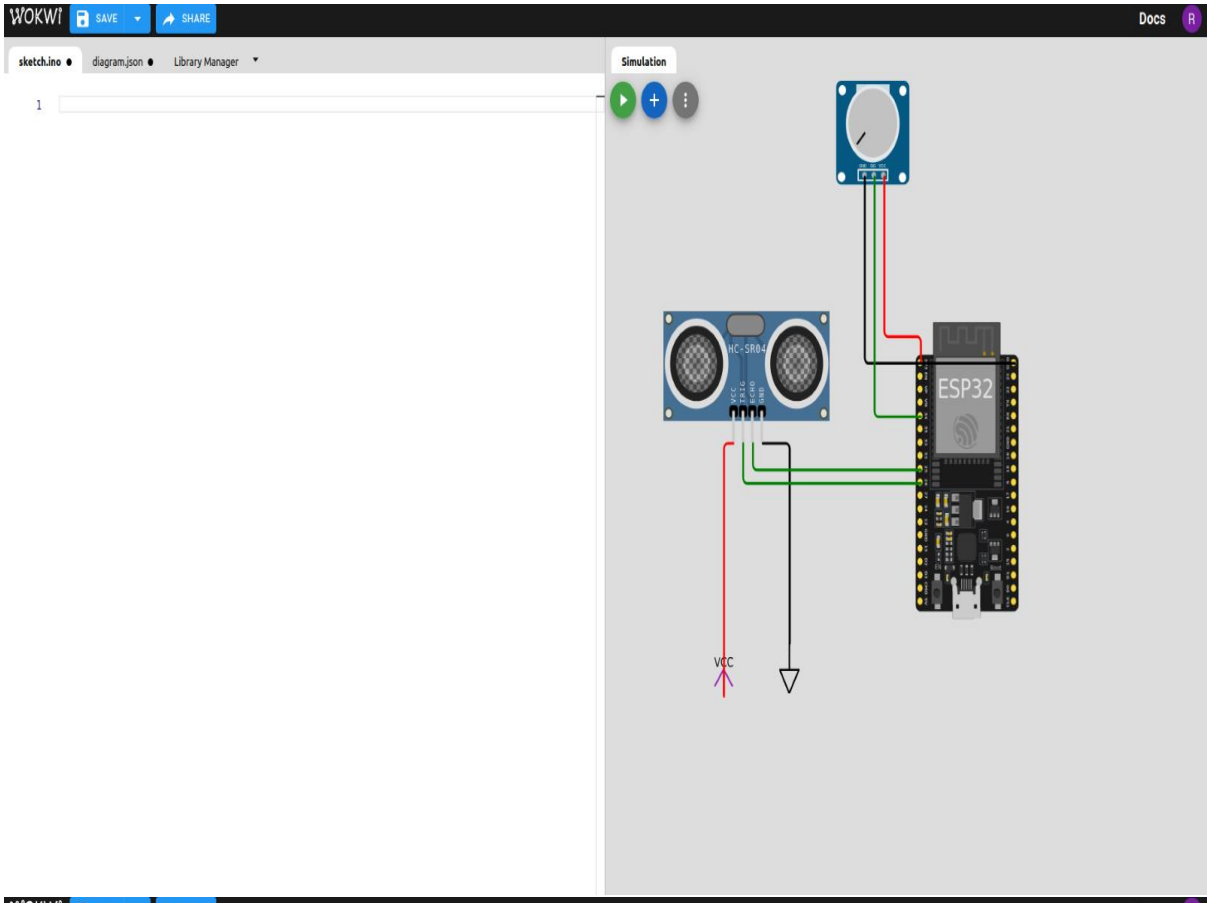
```
#include <Wire.h>
#include <WiFi.h>
#include <ThingSpeak.h>


const char *ssid = "Wokwi-GUEST";
const char *password = "";


#define TRIG_PIN 26
#define ECHO_PIN 25
#define POTENTIOMETER_PIN A0
const float contaminationThreshold = 7.0;


unsigned long channelID = 2327527; // Use Your ThingSpeak Channel ID
const char *writeAPIKey = "3K1OPPG28L2BIA3E"; // Use Your ThingSpeak Write API Key


WiFiClient client;


void setup() {
Serial.begin(115200);
connectToWiFi();
ThingSpeak.begin(client);
pinMode(TRIG_PIN, OUTPUT);
pinMode(ECHO_PIN, INPUT);
}


void loop() {
float distance = readUltrasonicDistance();
float pHValue = analogRead(POTENTIOMETER_PIN) / 100.0;
Serial.print("Ultrasonic Distance: ");
Serial.print(distance);
Serial.println(" cm");


Serial.print("pH Value: ");
Serial.println(pHValue);


if (pHValue < contaminationThreshold) {
Serial.println("Water is contaminated!");
} else {
```

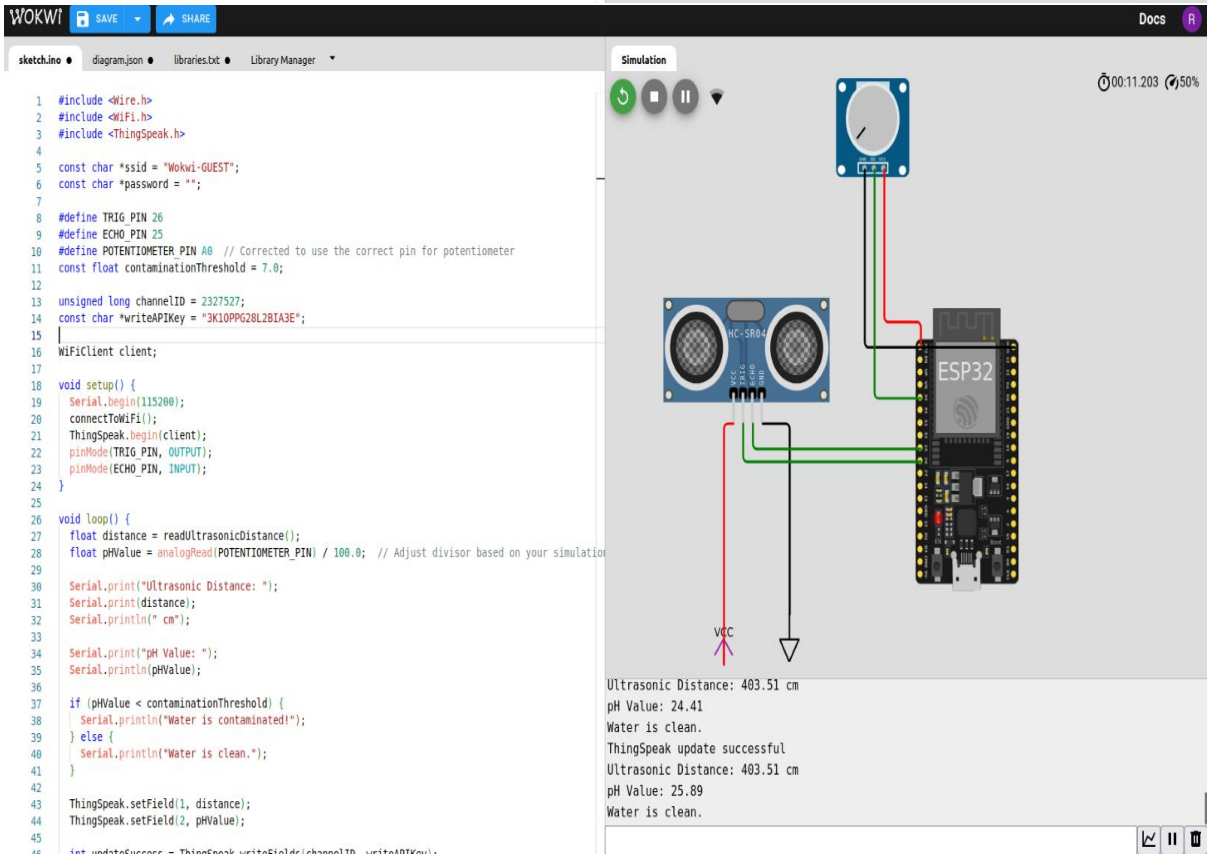
```
Serial.println("Water is clean.");
}


ThingSpeak.setField(1, distance);
ThingSpeak.setField(2, pHValue);


int updateSuccess = ThingSpeak.writeFields(channelID, writeAPIKey);


if (updateSuccess) {
Serial.println("ThingSpeak update successful");
} else {
Serial.println("Error updating ThingSpeak");
}


delay(2000);
}


void connectToWiFi() {
Serial.print("Connecting to WiFi");
WiFi.begin(ssid, password);


while (WiFi.status() != WL_CONNECTED) {
delay(1000);
Serial.print(".");
}


Serial.println("\nConnected to WiFi");
}


float readUltrasonicDistance() {
digitalWrite(TRIG_PIN, LOW);
delayMicroseconds(2);
digitalWrite(TRIG_PIN, HIGH);
delayMicroseconds(10);
digitalWrite(TRIG_PIN, LOW);


return pulseIn(ECHO_PIN, HIGH) * 0.0343 / 2;
}
```

Diagrams:

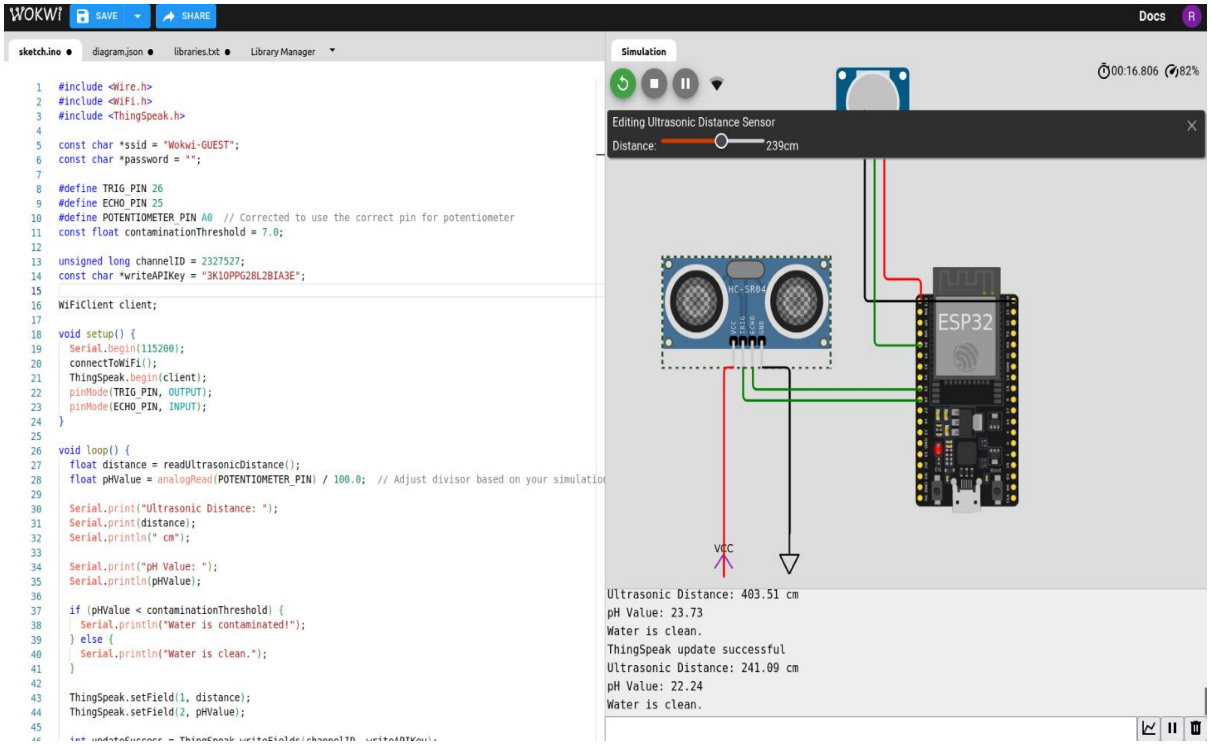

WOKWI
SAVE
SHARE
Docs
sketch.ino
diagram.json
Library Manager
1
vcc
Power
VCC Symbol
HC-SR04
ESP32


WOKWI
SAVE
SHARE
Docs
sketch.ino
diagram.json
Library Manager
1
Simulation
HC-SR04
ESP32
VCC






```
#include <Wire.h>
#include <WiFi.h>
#include <ThingSpeak.h>

const char *ssid = "Wokwi-GUEST";
const char *password = "";

#define TRIG_PIN 26
#define ECHO_PIN 25
#define POTENTIOMETER_PIN A0  // Corrected to use the correct pin for potentiometer
const float contaminationThreshold = 7.0;

unsigned long channelID = 2327527;
const char *writeAPIKey = "3K1OPPG28L2BIA3E";

WiFiClient client;

void setup() {
  Serial.begin(115200);
  connectToWiFi();
  ThingSpeak.begin(client);
  pinMode(TRIG_PIN, OUTPUT);
  pinMode(ECHO_PIN, INPUT);
}

void loop() {
  float distance = readUltrasonicDistance();
  float pHValue = analogRead(POTENTIOMETER_PIN) / 100.0;  // Adjust divisor based on your simulatio

  Serial.print("Ultrasonic Distance: ");
  Serial.print(distance);
  Serial.println(" cm");

  Serial.print("pH Value: ");
  Serial.println(pHValue);

  if (pHValue < contaminationThreshold) {
    Serial.println("Water is contaminated!");
  } else {
    Serial.println("Water is clean.");
  }

  ThingSpeak.setField(1, distance);
  ThingSpeak.setField(2, pHValue);
```



Uses of this real time water consumption monitoring system:

- **Real-Time Feedback:** Users receive immediate feedback on their water consumption patterns, encouraging awareness of daily usage.
- **Educational Insights:** Visualization of consumption trends helps users understand their impact on water resources, fostering a sense of responsibility.
- **Comparative Analysis:** Users can compare their water consumption with neighbors or community averages,

fostering healthy competition and community-wide conservation efforts.

• **Community Challenges:** Implementing challenges or initiatives within a community can further encourage water-saving practices.

*These are all the uses and advantages when we monitor the real time data of water consumption system.

*Because water is the driving force.

*We can save water by seeing the data and it can be useful to the environment.

*By doing this we can save water for future generations.

*So we want to save water to save life.

"Water is life, and clean water means health."

- Audrey Hepburn