**Phase 5 project:**

**Project Title :** SMART WATER FOUNTAIN

**Project ID :** proj\_223731\_Team\_5

**College Code :** 6208

**College :** Gnanamani College of Technology

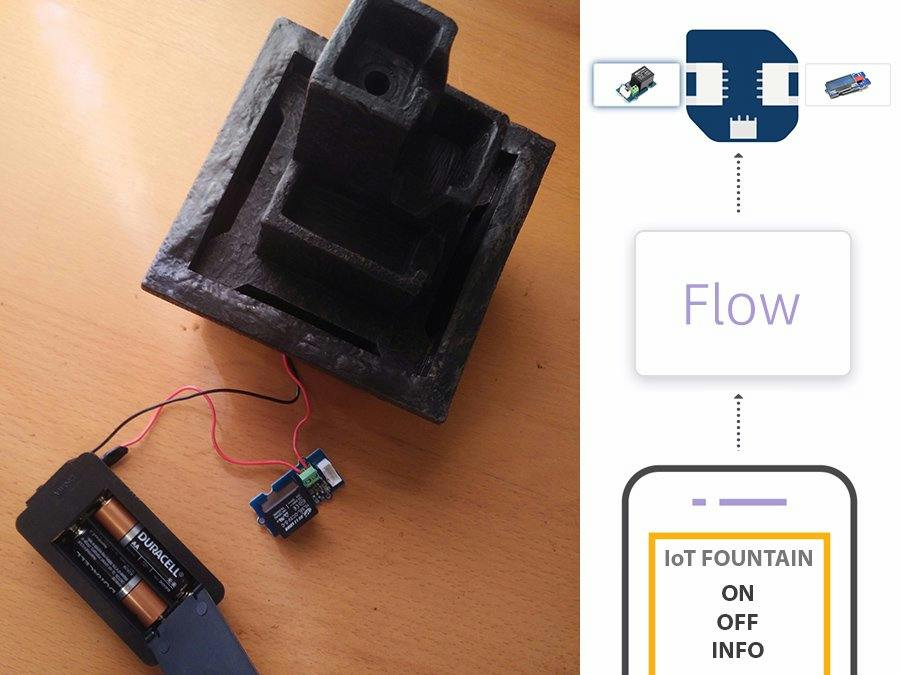
**Branch :** B.Tech-Information Techology

**Year :** IIIrd year

**Team members:**

* MATHANRAJ NS (620821205035)
* SIVAPRIYAN UV(620821205306)
* MOHANRAJ R(620821205036)
* ARULSURIYAN R (620821205004)
* SIVANESAN S(620821205305)

Smart Water Fountain



# Introduction

**Definition:**

A Smart Water Foundation project using IoT refers to an initiative that employs Internet of Things (IoT) technology to enhance the management, conservation, and sustainability of water resources. It involves the deployment of IoT devices equipped with sensors and communication capabilities to collect real-time data on various aspects of water systems, such as quality, quantity, and infrastructure conditions. This data is then processed and analyzed to optimize water usage, improve efficiency, detect issues like leaks or contamination, and promote responsible water management practices. These projects aim to leverage IoT technology to address water-related challenges and contribute to more intelligent, efficient, and sustainable water management.

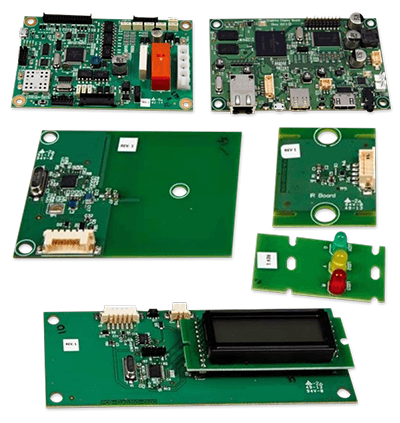
## Objective:

Today, more people around the world have pets than ever before. According to American Pet Products Association’s survey in 2020, 67% of U.S. households own a pet which is about 84.9 million homes. This proportion has been increased by 20% in thirty years . Breakdown of the pet types, cats and dogs are the most popular animals, they contribute to about 80% of all pets. Same trend happens all over the world. On average, one in three households own a dog globally and about a quarter of households worldwide own a cat . Both cats and dogs prefer flowing water. A source of fresh clean running water can encourage pets to drink. Drinking a certain amount of water daily plays an important role in long-term health for pets, especially cats. As a result, a water fountain is essential to most households having cats or dogs as pets. However, we can not ensure the water quality when we are away from home for several days. It can happen when pets have finished all remaining water in the water fountain, or water has been polluted somehow by the pet.

These can cause the pet to be unwilling to drink water from the fountain.Our goal is to design a smart water fountain that can monitor the water quality and automatically replace water when polluted(not healthy) or running out. We will use sensors to measure the water quality. Common water quality measurement factors include temperature, Ph-value, conductance, turbidity and hardness . Considering the pollution at home can only affect limited factors, we choose temperature, Ph-value and conductance to be the three properties used for calculating water quality in our water fountain. These data will be collected, calculated, and reflected to the user in terms of “Good”, “Average” and “Bad”. The water fountain is also designed to self-filter the water every time when water is pumped through the submersible water pump.

**2.REQUIREMENTS:**

* IoT Module
* ESP32
* Sensors
* Ultrasonic Sensors
* pH Sensors
* Turbidity Sensors
* Flow Sensors
* Actuators
* Power supply
* Communication
* MQTT
* HTTP
* IoT platform
* AWS IoT
* Azure IoT
* Google Cloud IoT
* User Interface
* Data storage

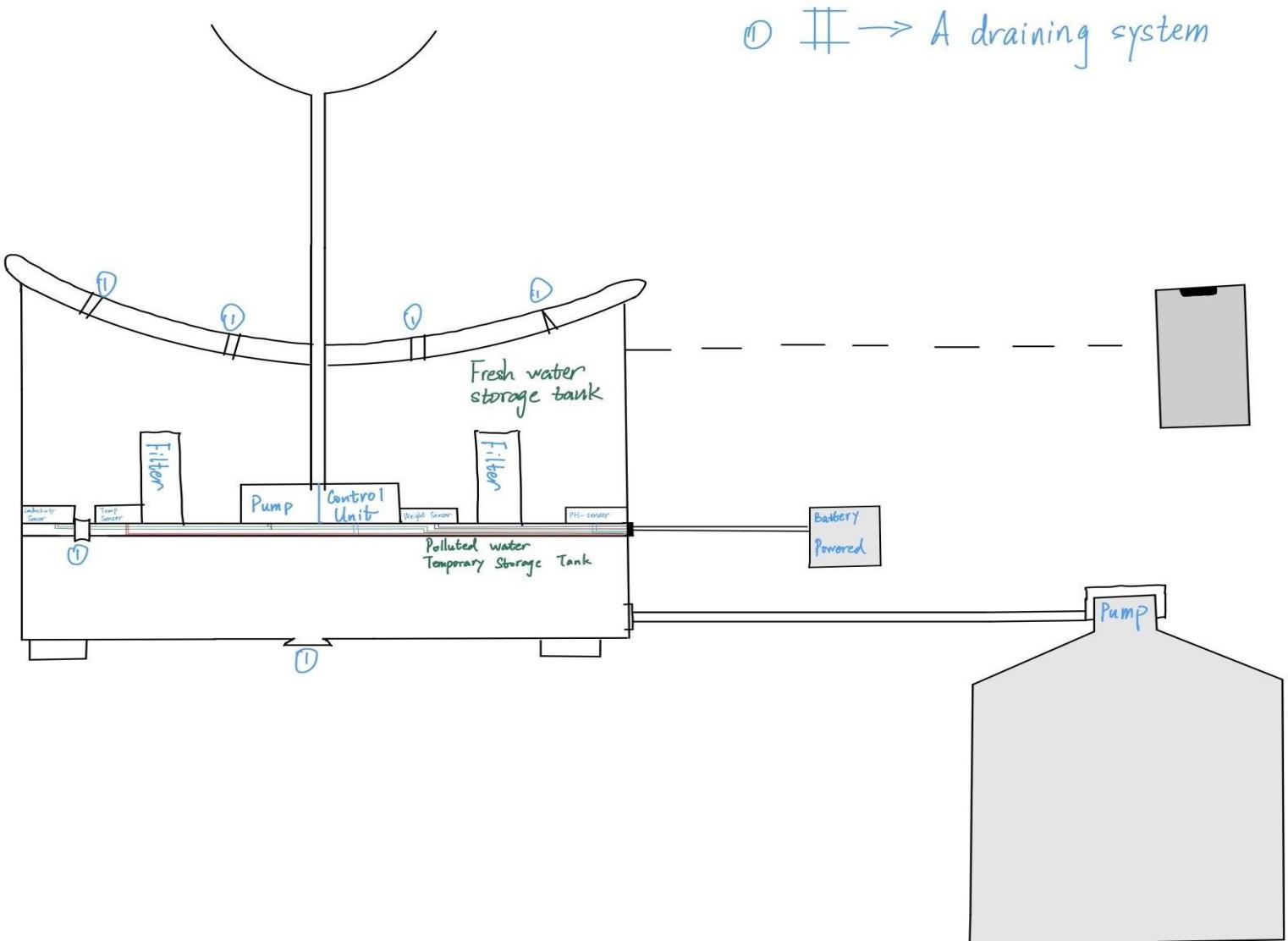


**DESIGN FEATURES:**

* Fountains wirelessly communicate with base stations.
* Base stations collect and transmit usage, filter, and system health information to the cloud via Ethernet.
* Wireless communications use a low-power unlicensed band for improved security and power savings.

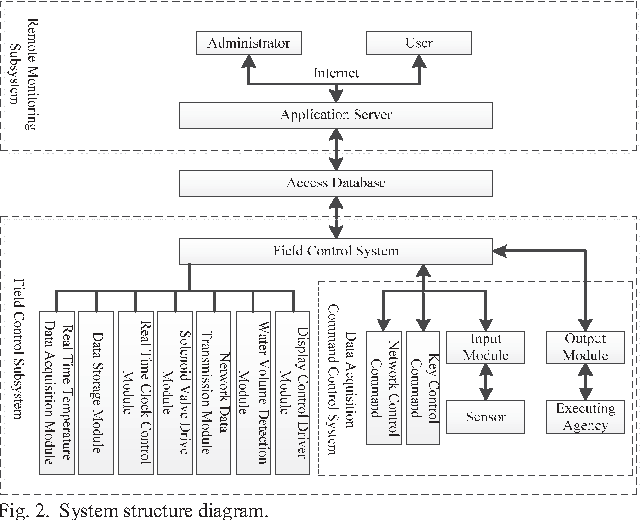
## 3.Physical Design:

A pictorial representation of your project that puts your solution in context. Not necessarily restricted to your design. Include other external systems relevant to your project (e.g. if your solution connects to a phone via Bluetooth, draw a dotted line between your device and the phone). Note that this is not a block diagram and should explain how the solution is used, not a breakdown of inner components.

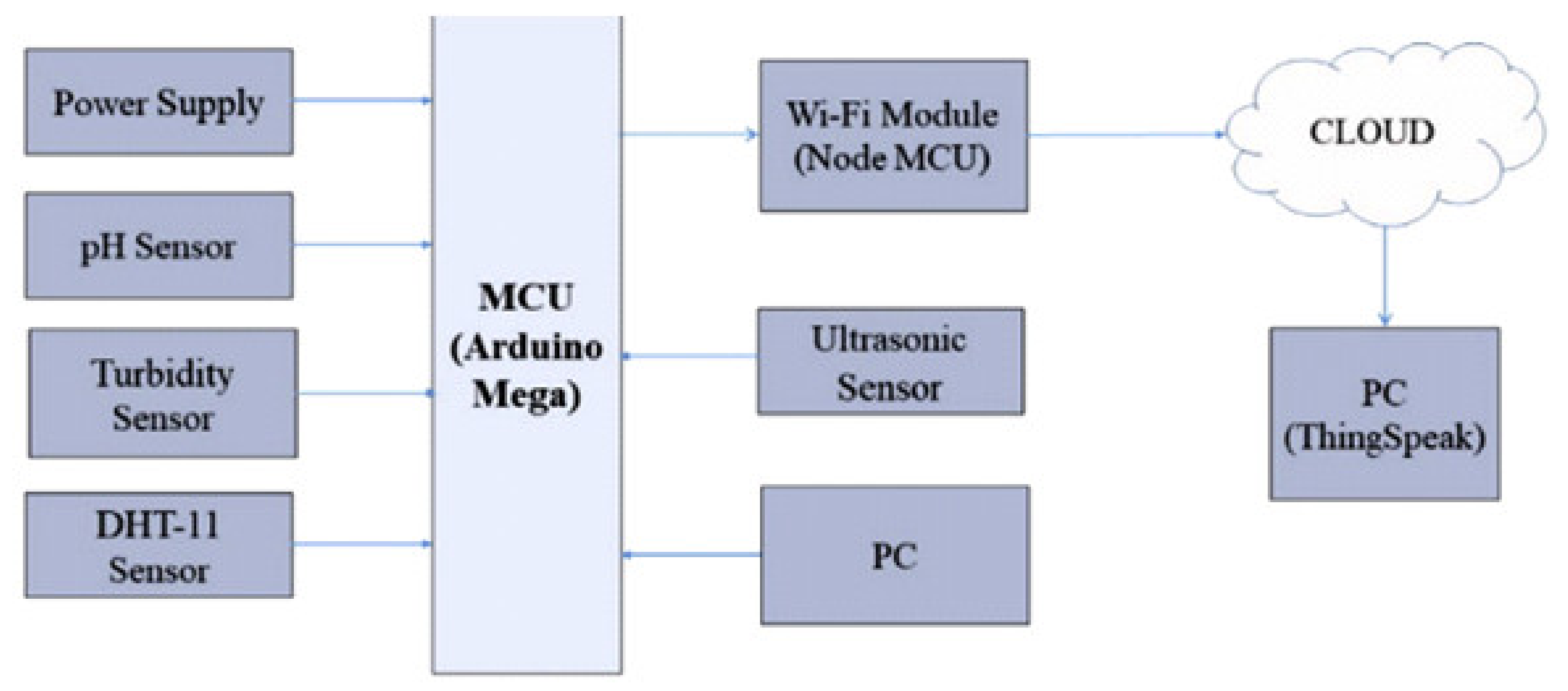


***Smart Fountain Physical Diagram***

**Flowchart:**



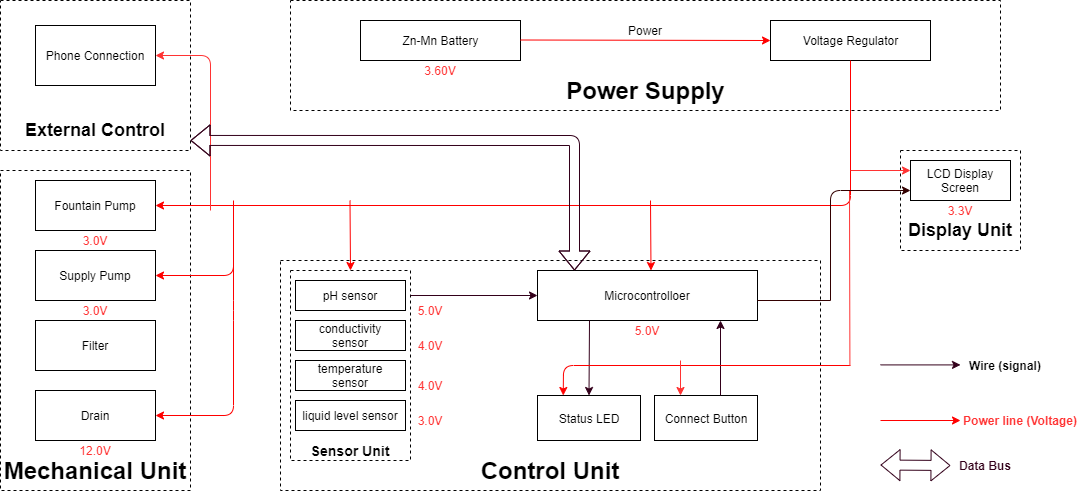
**Design Thinking:**



# 4.Design :

The block diagram below is a general design of our solution. We divide our design into four modules, including Power Supply, Control Unit, External Control, and Mechanical Unit.

Details of each unit is presented in the diagram and described in the next section.



***Block Diagram of Smart Water Fountain***

## 5.Sensor Unit :

This block contains the four sensors. The data acquired from the sensors will be transmitted to the control unit. Control unit will then have some logic designed to send corresponding signals to control other blocks of the water fountain. At the same time, the display screen on the water fountain will display the readings along with the determined water quality level and remaining water quantity.

For the PH-value sensor, temperature sensor and conductivity sensor, values will be retrieved and calculated to determine the overall water quality level. When poor water quality is determined, the water replacement procedures will take place. The weight sensor readings will be used to determine the amount of fresh water left in the water tank.

**Ultrasonic Sensors:**

Ultrasonic sensors are electronic devices that use sound waves with frequencies higher than the human audible range (typically above 20 kHz) to measure distance or detect the presence of objects. They work on the principle of emitting ultrasonic pulses and measuring the time it takes for the sound waves to bounce back after hitting an object. Ultrasonic sensors are commonly used for various applications, including:

**1. Distance Measurement:** They can accurately measure the distance to an object by calculating the time it takes for the sound waves to return. This makes them suitable for applications like parking assistance, level monitoring, and robotics.

**2. Object Detection:** Ultrasonic sensors can be used to detect the presence or absence of objects within a certain range. They are often employed in security systems, industrial automation, and obstacle detection in autonomous vehicles.

**3. Proximity Sensing:** These sensors are used to determine the proximity of an object or obstacle, making them valuable for applications like automatic doors, hand sanitizers, and touchless switches.

**pH Sensors:**

pH sensors, also known as pH meters, are analytical instruments used to measure the acidity or alkalinity of a solution. pH is a measure of the concentration of hydrogen ions in a solution, and it is measured on a scale from 0 to 14, with 7 being neutral, values below 7 indicating acidity, and values above 7 indicating alkalinity. pH sensors are commonly used in various fields, including:

**1.** **Environmental Monitoring:** pH sensors are used to assess the pH levels of natural waters, such as lakes, rivers, and oceans, to monitor water quality and environmental conditions.

**2. Industrial Processes:** They are crucial in chemical processes, wastewater treatment, and the pharmaceutical industry to control and maintain the pH of solutions.

**3. Agriculture:** pH sensors can help farmers monitor the pH of soil and irrigation water, ensuring optimal conditions for plant growth.

**Turbidity Sensors:**

Turbidity sensors are used to measure the cloudiness or haziness of a fluid caused by the presence of suspended particles, such as sediment, silt, or organic matter. Turbidity is typically measured in Nephelometric Turbidity Units (NTU). These sensors find applications in:

**1. Water Quality Monitoring:** Turbidity sensors are used in environmental monitoring and water treatment facilities to assess water quality, detect pollution, and ensure the removal of particulate matter from drinking water.

**2. Wastewater Treatment:** They are essential for monitoring the efficiency of wastewater treatment processes by measuring the level of suspended solids in the effluent.

**Flow Sensors:**

Flow sensors, also known as flow meters, are devices used to measure the rate of fluid flow (liquids or gases) through a pipe or conduit. They are widely used in various industries and applications, including:

**1. Industrial Processes:** Flow sensors are used to monitor and control the flow of liquids and gases in industrial processes, ensuring accurate dosing and process control.

**2. HVAC Systems:** They help maintain the efficiency of heating, ventilation, and air conditioning systems by measuring fluid flow in pipes.

**3. Environmental Monitoring:** Flow sensors are used to measure the flow rates of water in rivers, streams, and sewers, aiding in flood prediction and water resource management.

1. **Automotive:** In the automotive industry, flow sensors are used in fuel injection systems and air intake measurements for engine efficiency.

**Mechanical Unit:**

Fountain Pump The fountain pump must maintain a continuous water supply through the fountain mechanism. The pump must work 24 hours a day, 7 days a week unless the user manually turns off the power supply.

**Requirement 1:** The fountain pump must lift a cylindrical water stream of diameter 6mm for a height of 400mm.

**Requirement 2:** The fountain pump must serve for a duration of 2 years without maintenance or replacement under heavy workload.

**Requirement 3:** The fountain pump should have an operational condition around 3V, 200mA.

**Supply Pump :**

The supply pump must function when a low water level alert is raised. While no water supply is requested, the pump must prevent water flow between the main supply and the fountain. Requirement: The supply pump should have an operational condition around 3V, 200mA.

**Filter :**

The filter must maintain the water quality through controlling the pH value and conductivity of the water.

**Requirement 1:** The filter must have a cost less than $5 each for frequent replacement. Each new filter must serve a duration no less than 3 month.

**Requirement 2:** The filter must be designed for easy removal and installation, while the connection mechanism must have a low degenerate rate when submerged in water.

**Drain:**

The drain must be able to hold and release water in the fountain. When water in the fountain should be replaced, the faucet should automatically drain the fountain once instruction is received from the integrated circuit.

**Control Unit :**

This unit contains the control unit which does the following things:

• When the weight sensor reports a weight less than the minimum weight setting, the control unit will send an alert signal to the user and then control the water supply unit to refill the water fountain with a certain amount of water.

• Computes the water quality with data transferred from the three sensors in the water quality module and sends the result in terms of “Good”, “Average” or “Bad” to the user.

• If the water quality is “Bad”, the control unit will control the drain module to drain the water in the fountain and then control the water supply to refill.

• Water quality result is sent to the user with wireless connection and screen display as described above in the display unit.(unsure about keeping this function)

**Risk Analysis:**

**Control Unit Block:**

One of the most challenging points in this project is the precise control of the control unit between different blocks. To react accurately and promptly based on the results from the sensors is the key. The control unit needs to accommodate the mechanical and the electrical part so that the pumps, draining system can work collaboratively smoothly. From acquiring the data from sensors, analyzing the data, communicating and displaying the data to users, and then sending signals to activate the corresponding actions(drain or add fresh water), these are all to be performed by the control unit. Thus, it is the block that brings the greatest risk.

We will divide all the overall control unit functions into three parts: data retrieving, data manipulation, data delivering. Data retrieving is the logic used to read data from all sensors. Necessary algorithm is to be written to ensure successful and accurate data acquisition. Data manipulation is the process of calculating the water quality levels, and the formula to integrate all the data to produce a credible result. The data delivering is used to connect the control unit to the screen, displaying the necessary information as described above. This part will also be responsible for building the connection between the water fountain and the users’ phones through WIFI.

**Mechanical Unit Block:**

This is very challenging and extremely important. As most of the components will be exposed to water. Sensors, pumps, filters, draining system motors are all to be placed in the water tank. This means that we need to ensure no water can leak into the electrical-related mechanical parts. This puts pressure on the design and also the implementation. In addition, the motor-controlled valves used to drain the polluted water need to be firm when closed. Otherwise the fresh water will be leaking to the polluted water storage and the water consumption will be uncontrollable. To achieve those points, we will make sure the designs are carefully implemented. The actual building process for the container should be proved before placing the electronic parts in.

**Ethics and Safety :**

**Mechanical Unit Block :**

**IEEE Code of Ethics:**

Quoted from IEEE Code of Ethics “To hold paramount the safety, health, and welfare of the public, to strive to comply with ethical design and sustainable development practices, to protect the privacy of others, and to disclose promptly factors that might endanger the public or the environment.” We will carefully choose the materials used to build the container. Non-toxic are sure to be used. We will prefer using reusable materials. In addition to that, the users can choose to buy reusable bottles of water for the freshwater supply for the water fountain. Those universal water bottles are safe and reusable. A special connector will be designed and the universal connection is to be used. After the water in the bottle is used up, this reusable bottle can be recycled and reused. This is the most environmentally-friendly solution and complies with the IEEE Code of Ethics . It not only improves the practicality, convenience, and reduces the future cost when using the water fountain.

To treat all persons fairly and with respect, to not engage in harassment or discrimination, and to avoid injuring others.” As mentioned in the 3.2, the mechanical unit involves electronic components that are physically placed in the water tank. The consequence can be serious if the leakproofness is not performed properly. To maintain a safe, convenient using experience, we will be responsible for testing and ensuring all containers meet the demand. These actions must be taken to ensure the safety of using the water fountain and protect the others.

“to maintain and improve our technical competence and to undertake technological tasks for others only if qualified by training or experience, or after full disclosure of pertinent limitations.” All team members involved in the development of the water fountain have completed “Laboratory Safety training” and have gained required and necessary knowledge in dealing with emergency situations. In case of accidents, proper reaction will be made to ensure the safety of people and property to the largest extent.

**6.PROPOSAL SYSTEM:**

**1.Leak Detection:**

Use IoT sensors to detect leaks in water supply pipelines. This can help in early leak detection, reducing water wastage, and preventing damage to infrastructure.

**Algorithm:** Machine Learning Algorithms

**2.Water Usage Analytics:**

Collect data on water consumption patterns to identify trends and anomalies. This can aid in optimizing water distribution and billing processes.

**Algorithm:** Ramdom Forest Algorithm

**3.Mobile Apps for Water Management:**

Develop user-friendly mobile applications that allow consumers to monitor their water usage, receive alerts, and report issues such as leaks or water quality concerns.

**Algorithm:** Machine Learning Algorithms

**4.Remote Valve Control:**

Enable remote control of water valves to shut off or redirect water flow during emergencies or maintenance, improving system resilience.

**Algorithm:** Machine Learning Algorithm

**5.Real-time Alerts:**

Set up real-time alerts and notifications for critical events such as equipment failures, water quality breaches, or excessive water usage.

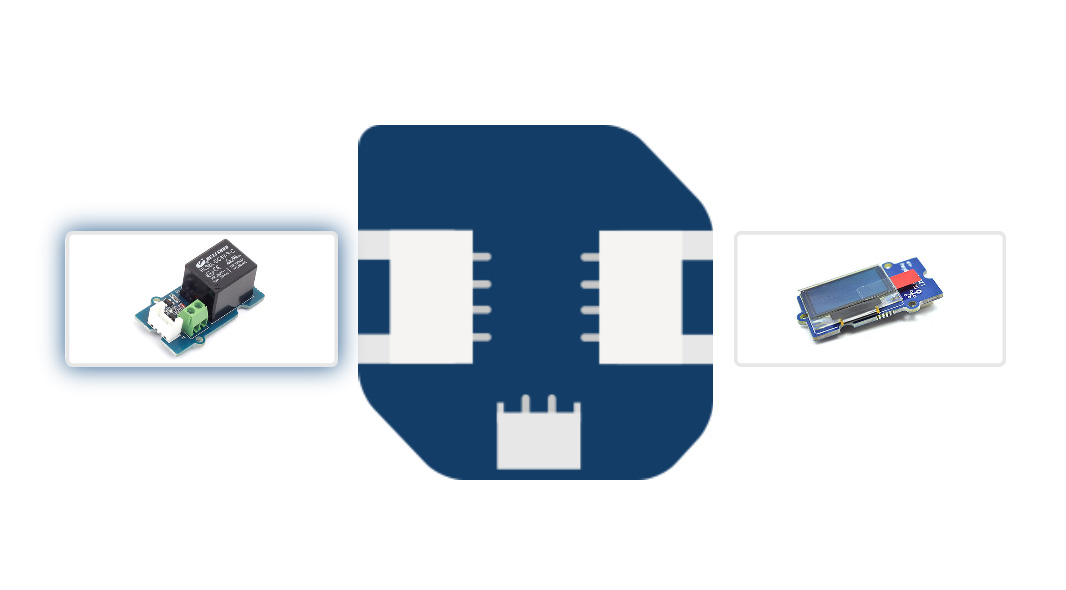
**Algorithm:** AI Algorithm

**6.Solar-Powered IoT:**

Consider using solar power to operate IoT sensors and devices, reducing energy costs and making the system more sustainable.

**Algorithm:** Maximum Power Point Tracking Algorithm

## Schematics:



**CODE IMPLEMENTATION :**

Developing a mobile application for a smart water fountain using a Raspberry Pi as part of an IoT project involves multiple components, including hardware setup and software development. Below, I'll provide an outline of the steps required to create such an application using Python, Flask for the server, and a mobile app for control.

**Hardware Components:**

1. Raspberry Pi (with Wi-Fi connectivity)

2. Water pump and associated hardware

3. Sensors (e.g., water level sensor)

4. Relay module (for controlling the pump)

5. Power source for the water pump

**Software Components:**

1. Python for Raspberry Pi

2. Flask for creating a web server on the Raspberry Pi

3. Mobile app development tools (e.g., React Native for cross-platform apps)

Steps to Create the Mobile App for Smart Water Fountain:

**1. Hardware Setup:**

Set up the Raspberry Pi, water pump, sensors, and relay module as per your project requirements. Ensure that the Raspberry Pi is connected to the same Wi-Fi network as your mobile device.

**2. Raspberry Pi Server:**

Develop a Python script using the Flask framework to create a REST ful API that will control the water pump and monitor sensors. Your script might look like this:

python

from flask import Flask, request

app = Flask(\_\_name)

@app.route('/control', methods=['POST'])

def control\_fountain():

# Use GPIO libraries to control the water pump via the relay module.

if request.json['action'] == 'on':

# Turn the water pump on.

else:

# Turn the water pump off.

return "OK"

@app.route('/status', methods=['GET'])

def get\_status():

# Read the status of water level sensor, fountain status, etc.

return "Status information"

if \_\_name\_\_ == '\_\_main\_\_':

app.run(debug=True, host='0.0.0.0')

**3. Mobile App Development:**

Create a mobile app using a framework like React Native or a language like Kotlin (for Android) or Swift (for iOS). The app should have user interfaces to control the fountain and display its status. Use HTTP requests to interact with the Raspberry Pi's Flask server.

Here's a simplified example of how to send requests to control the fountain in React Native:

<!DOCTYPE html>

<html lang="en-us">

<head>

<meta charset="utf-8">

<title>IOT Fountain</title>

<style type="text/css">

#mainContent {

font-family: Arial, Helvetica, sans-serif;

font-size: xx-large;

font-weight: bold;

background-color: #E3F0FB;

border-radius: 4px;

padding: 10px;

text-align: center;

}

.buttonStyle {

border-radius: 4px;

border: thin solid #F0E020;

padding: 5px;

background-color: #F8F094;

font-family: "Segoe UI", Tahoma, Geneva, Verdana, sans-serif;

font-weight: bold;

color: #663300;

width: 75px;

}

.buttonStyle:hover {

border: thin solid #FFCC00;

background-color: #FCF9D6;

color: #996633;

cursor: pointer;

}

.buttonStyle:active {

border: thin solid #99CC00;

background-color: #F5FFD2;

color: #669900;

cursor: pointer;

}

</style>

<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.1.1/jquery.min.js"></script>

<script>

$(document).ready(function(){

$("#foff").click(function(){

$.get("https://run-east.att.io/xyz/xyz/xyz/in/flow/wateroff", function(data, status){

// alert("Data: " + data + "\nStatus: " + status);

logSomething(data);

});

});

$("#fon").click(function(){

$.get("https://run-east.att.io/xyz/xyz/xyz/in/flow/wateron", function(data, status){

// alert("Data: " + data + "\nStatus: " + status);

logSomething(data);

});

});

$("#finfo").click(function(){

$.get("https://run-east.att.io/xyz/xyz/xyz/in/flow/contact", function(data, status){

// alert("Data: " + data + "\nStatus: " + status);

logSomething(data);

});

});

$("#fclear").click(function(){

$.get("https://run-east.att.io/xyz/xyz/xyz/in/flow/clear", function(data, status){

//alert("Data: " + data + "\nStatus: " + status);

logSomething(data);

});

});

});

</script>

</head>

<body>

<div id="mainContent">

<p id="statusText">IOT Fountain</p>

<button id="foff" class="buttonStyle">Fountain Off</button>

<button id="fon" class="buttonStyle">Fountain On</button>

<button id="finfo" class="buttonStyle">Contact Info</button>

<button id="fclear" class="buttonStyle">Clear</button>

</div>

<script>

var myText = document.getElementById("statusText");

function logSomething(str) {

myText.textContent = str;

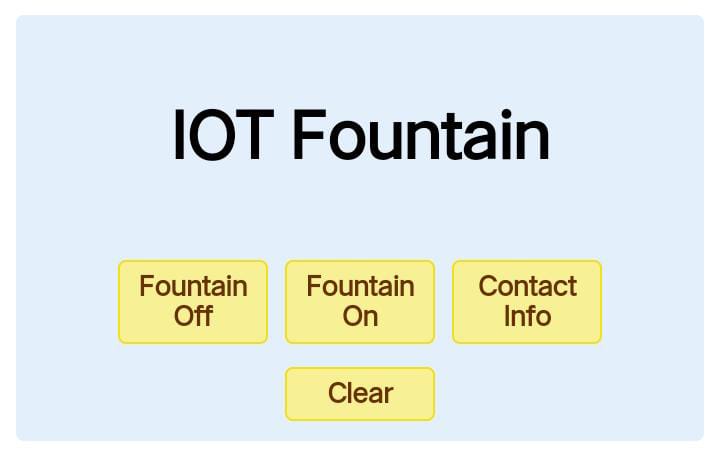
}

</script>

</body>

</html>

**OUTPUT:**

****

1. **Test and Deploy:**

Test your mobile app and Raspberry Pi server locally. Once everything works as expected, deploy the Flask server on the Raspberry Pi, and publish your mobile app on app stores if needed.

Please note that this is a simplified outline, and in practice, you will need to consider security, error handling, and additional features like real-time monitoring and notifications. Additionally, the actual implementation may vary depending on the specific hardware and software components you are using for your smart water fountain IoT project.

**Set up the hardware:**

Connect the water level sensor to the Arduino to measure the water level.

Connect the water pump to the Arduino through a relay module if necessary.

Connect LEDs for status indicators.

**install the required libraries:**

Depending on your hardware components, you may need to install libraries for the water level sensor, relay module, and IoT connectivity (Arduino IoT Cloud ).

**Arduino sketch:**

cpp

Copy code

#include <ArduinoIoTCloud.h> // If using Arduino IoT Cloud

//#include <BlynkSimpleEsp8266.h> // If using Blynk

#include <WiFiNINA.h> // Include the WiFi library for connectivity

// Define your network credentials

const char\* ssid = "YourWiFiSSID";

const char\* pass = "YourWiFiPassword";

// Define your IoT Cloud or Blynk authentication token

const char\* iotCloudToken = "YourIoTCloudToken";

//const char\* blynkToken = "YourBlynkToken";

// Define pin connections

const int waterLevelPin = A0;

const int pumpPin = 7;

const int statusLEDPin = 13;

void setup() {

// Initialize serial communication for debugging

Serial.begin(9600);

// Connect to Wi-Fi

WiFi.begin(ssid, pass);

while (WiFi.status() != WL\_CONNECTED) {

delay(1000);

Serial.println("Connecting to WiFi...");

}

Serial.println("Connected to WiFi");

// Initialize the IoT platform

ArduinoCloud.begin(iotCloudToken);

// Set up pin modes

pinMode(waterLevelPin, INPUT);

pinMode(pumpPin, OUTPUT);

pinMode(statusLEDPin, OUTPUT);

}

void loop() {

// Read the water level from the sensor

int waterLevel = analogRead(waterLevelPin);

// Check water level and control the pump

if (waterLevel < 500) {

// Low water level detected, turn on the pump

digitalWrite(pumpPin, HIGH);

digitalWrite(statusLEDPin, HIGH);

} else {

// Sufficient water, turn off the pump

digitalWrite(pumpPin, LOW);

digitalWrite(statusLEDPin, LOW);

}

// Update IoT platform with the water level data

ArduinoCloud.update();

// Add delay to avoid continuous readings

delay(1000);

}

**Benefits:**

* Provides clean and safe drinking water.
* Offers convenience with remote control and monitoring.
* Helps in water conservation through usage data analysis.

**Challenges:**

* Ensuring water quality and filter maintenance.
* Securing the IoT system against cyber threats.
* Power management to ensure the system operates continuously.

This project involves various technologies and requires a good understanding of IoT, sensors, and water filtration systems. It's important to prioritize safety and data security, as well as regularly maintain the components for optimal performance.

**AT&T Flow Designer:**

We can use Flow Designer to bring the API's together.

Here we create the following new API calls:

* wateroff - turns relay off & sends 'off' text to OLED.
* wateron - turns relay on & sends 'on' text to OLED.
* contact - sends Contact Info text to OLED.
* clear - clears the OLED.

**Future Plans/Ideas:**

* Control via Alexa or other Natural Language Processing.
* Show random inspirational quotes.
* Show data stats; stocks, emails, notifications, meetings, etc.
* Add timers.

**Conclusion:**

Now you can control your Meditation Fountain from any web browser! Turn it on from your phone, and put your own messages on the screen. Never again have to worry if you left it on after you leave the office.