**NAAN MUDHALVAN IOT TEAM PROJECT**

**TITLE :SMART WATER FOUNTAINS**

**TEAM NAME : Proj\_228481\_Team\_1**

**COLLEGE NAME: JEPPIAAR INSTITUTE OF TECHNOLOGY**

**TEAM MEMBERS**:

**1.BHUVANASRI.S**

**2.DEEPIKA.M**

**3.KIRUTHIKA.K**

**4.SELVAMEENA.N**

**5.SHOFIYA.M**

**PHASE 03:**

In this part you will begin building your project.

Start building the IoT-enabled Smart Water Fountains system.

Deploy IoT sensors (e.g., flow rate sensors, pressure sensors) in public water fountains to monitor water flow and detect malfunctions.

Develop a Python script on the IoT sensors to send real-time water fountain status data to the platform.

**OBJECTIVE:**

Design and implement an IoT-enabled Smart Water Fountains system that utilizes flow rate sensors, pressure sensors, and IoT technology to monitor the water flow and detect malfunctions in public water fountains. Develop a Python-based IoT sensor script to collect real-time data and transmit it securely to a central platform. Create a robust, user-friendly dashboard for real-time monitoring and implement automated alerts for anomalies. The objective is to enhance public water fountain efficiency, reduce water wastage, and ensure timely maintenance, thereby contributing to sustainable water management.

**PYTHON CODE FOR THE SMART WATER FOUNTAIN:**

To build an IoT-enabled Smart Water Fountain system, need to deploy sensors, collect data, and send it to a platform for analysis.

**Hardware Setup:**

1. Flow Rate Sensor: Measure the flow of water in the fountain.

2. Pressure Sensor: Monitor water pressure to detect irregularities.

3. Microcontroller (e.g., Raspberry Pi, Arduino): Interface with sensors and send data to the platform.

4. Internet Connectivity (Wi-Fi, GSM, LoRa, etc.): Enable communication between the IoT devices and the platform.

**Software Setup**:

1. Python Script on IoT Sensors:

```python

Import requests

Import time

From gpiozero import InputDevice # For reading sensors

# Define sensor pins

FLOW\_SENSOR\_PIN = 14

PRESSURE\_SENSOR\_PIN = 15

# Initialize sensors

Flow\_sensor = InputDevice(FLOW\_SENSOR\_PIN)

Pressure\_sensor = InputDevice(PRESSURE\_SENSOR\_PIN)

Def read\_sensors():

Flow\_rate = 0 # Read flow rate from flow sensor (implement logic based on your sensor)

Pressure = 0 # Read pressure from pressure sensor (implement logic based on your sensor)

Return flow\_rate, pressure

Def send\_data\_to\_platform(flow\_rate, pressure):

Data = {

“flow\_rate”: flow\_rate,

“pressure”: pressure,

“timestamp”: time.time()

}

Try:

Response = requests.post(API\_ENDPOINT, json=data)

If response.status\_code == 200:

Print(“Data sent successfully!”)

Else:

Print(“Failed to send data. Status code:”, response.status\_code)

Except Exception as e:

Print(“Error occurred while sending data:”, str€)

Def main():

While True:

Flow\_rate, pressure = read\_sensors()

Send\_data\_to\_platform(flow\_rate, pressure)

Time.sleep(1) # Send data every 1 second

If \_\_name\_\_ == “\_\_main\_\_”:

Main()

```

2. Platform:

Set up a server or cloud-based platform to receive data from IoT sensors. You’ll need to implement an API endpoint (`/data` in the example above) to receive POST requests containing sensor data.Ma

3. Data Storage and Analysis:

Store received data in a database for historical analysis. You can use databases like MySQL, MongoDB, or cloud-based solutions like AWS DynamoDB or Firebase. Perform data analysis to detect patterns, malfunctions, or irregularities in water flow and pressure.

4. Alert System:

Implement an alert system to notify relevant authorities or maintenance personnel if irregularities or malfunctions are detected. This could be done via email, SMS, or push notifications depending on your requirements.

**CONCLUSION:**

In conclusion, implementing an IoT-enabled Smart Water Fountains system offers numerous advantages, including real-time monitoring of water flow and quick detection of malfunctions. By deploying sensors and developing Python scripts, we can ensure efficient data collection and transmission to the monitoring platform. This technology not only improves the maintenance of public water fountains but also contributes to water conservation efforts. Embracing such innovative solutions enhances the overall reliability and sustainability of public infrastructure, making our communities smarter and more resource-efficient.