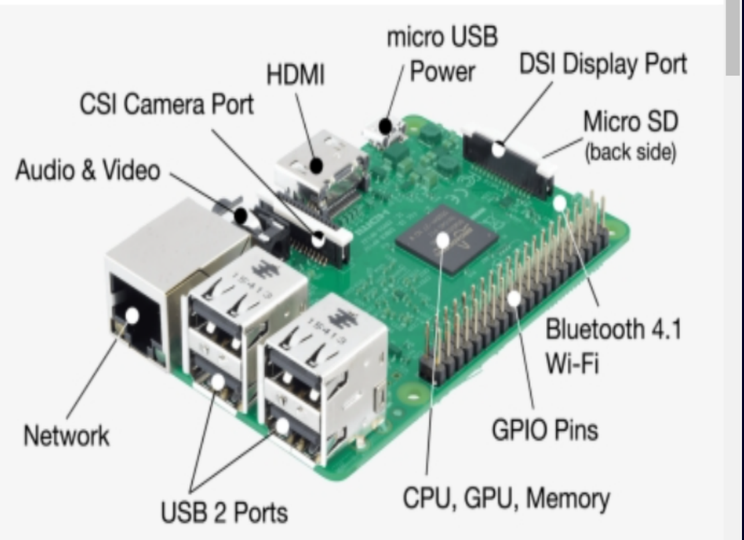
Smart Water Fountain

Introduction to Respberry pi 4

Raspberry Pi is a series of affordable,credit card-sized single-board computers . Equipped with GeneralPurpose Input/Output (GPIO) pins andsupporting various operating systems.utilized in diverse projects, ranging from basic computing tasks to complex DIY electronics, robotics, anInternet of Things (IoT) applications.Monitor water usage patterns and promote efficient waterconsumption.Control water flow based on real-time usage and demand.Provide users with real-time data and access to control thefountain remotely.Raise awareness about water conservation and environmentalsustainability.Demonstrate the capabilities of IoT and smart technology in promoting responsible water usage.



Project Definition and Design Thinking

Project Objective’s :

1. Problem Explain & Design Thinking.
2. Improve water conservation by implementing intelligent sensors and controls to regulate water flow and minimize wastage.
3. Integration the verious required components with Respberry pi 4 .
4. Enable real-time data monitoring and analytics by establishing a robust data collection system, allowing for continuous monitoring of water usage patterns and identifying areas for further optimization.
5. Conclusion for Smart Water Fountain .

Required Components :

1. Flow rate Sensor.
2. Water Level Sensor.
3. Water pump.
4. Relay module
5. Power supply.
6. Raspberry Pi 4 Model B.
7. Raspberry Pi OS (formerly Raspbian OS).
8. Libraries for GPIO control .
9. USB CDC (Serial) supported .
10. Python code for Program.
11. Cloud AWS IoT .
12. Real-time Updation UI Design (HTML , CSS , JS).

Hardware Components Step :

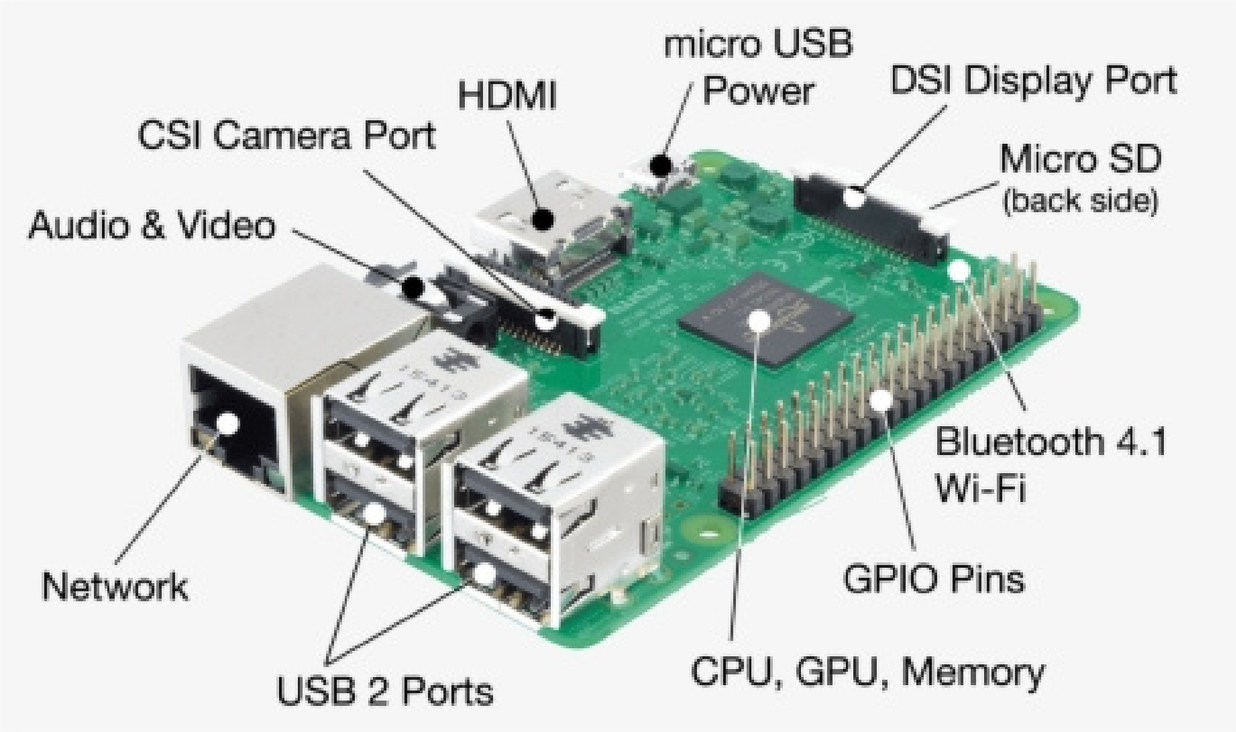
1. Flow Rate Sensor Integration.
2. Water Level Sensor Implementation .
3. Smart Water Pump Control .
4. Raspberry Pi 4 as the Central Control Unit .
5. Efficient Power Supply System .
6. Cloud Connectivity and Remote Access .

Install Required Software Libraries :

1. Raspberry Pi OS Integration .
2. GPIO Control Libraries Implementation .
3. USB CDC (Serial) Support Integration .
4. Python Programming for System Control .
5. AWS IoT Cloud Integration .
6. Real-time Updation UI Design (HTML , CSS , JS).
7. Data Visualization and Analytics .
8. Scalability and Flexibility .

Raspberry Pi Integration :

Utilize the Raspberry Pi 4 as the central control unit to manage the data from the sensors, control the water pump, and facilitate communication with the cloud server for remote monitoring and management.



Raspberry Pi 4 Data Transfer :

- Install the latest version of Raspberry Pi OS (formerly Raspbian OS) on your Raspberry Pi 4.
- Connect necessary hardware components, including the water level sensor, flow rate sensor, water pump, and any other relevant components, ensuring proper wiring and connections.
- Install required libraries for GPIO control to enable communication with the connected sensors and actuators .
- Define MQTT Topics and Messages .
- Implement Data Transfer Logic .
- Monitor and Manage Data on AWS IoT Console .

Web Server UI Design

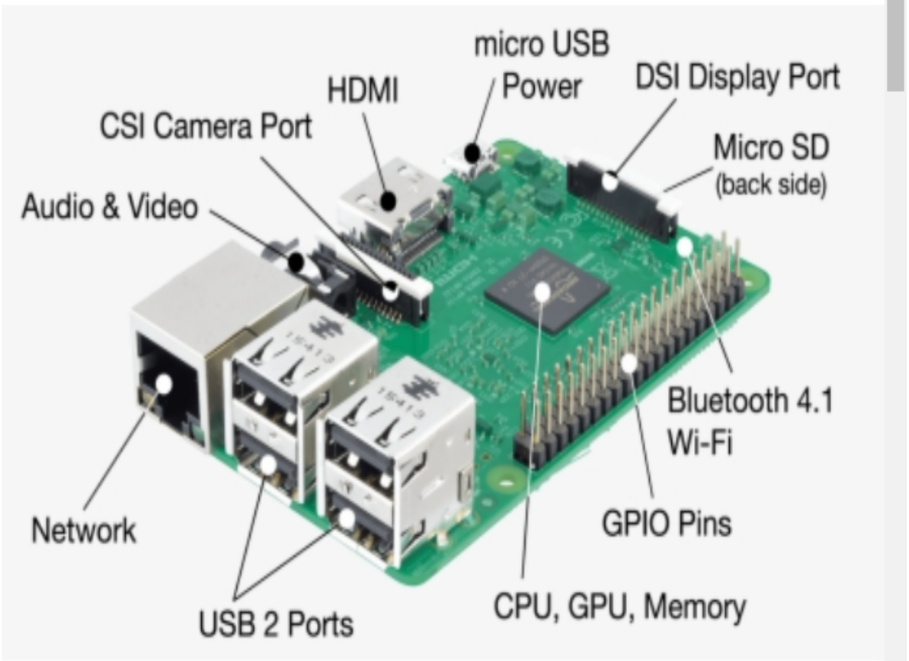
```html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Smart Water Fountain</title>
</head>
<body>
    <header class="header">
        <div class="logo">
            <h2>Smart Water Fountain</h2>
        </div>
        <div class="navbar">
            <nav class="nav">
                <a href="#">Home</a>
                <a href="#aboutus">About us</a>
            </nav>
        </div>
    </header>

    <!-- About Page -->

    <div class="info_about">
        <div class="items">
            <h2>Introduction to Respberry
            <p>Raspberry Pi is a series of
        </div>
        <div class="items">
            <img src="raspberry2.png" alt=
        </div>
    </div>
```

Smart Water Fountain

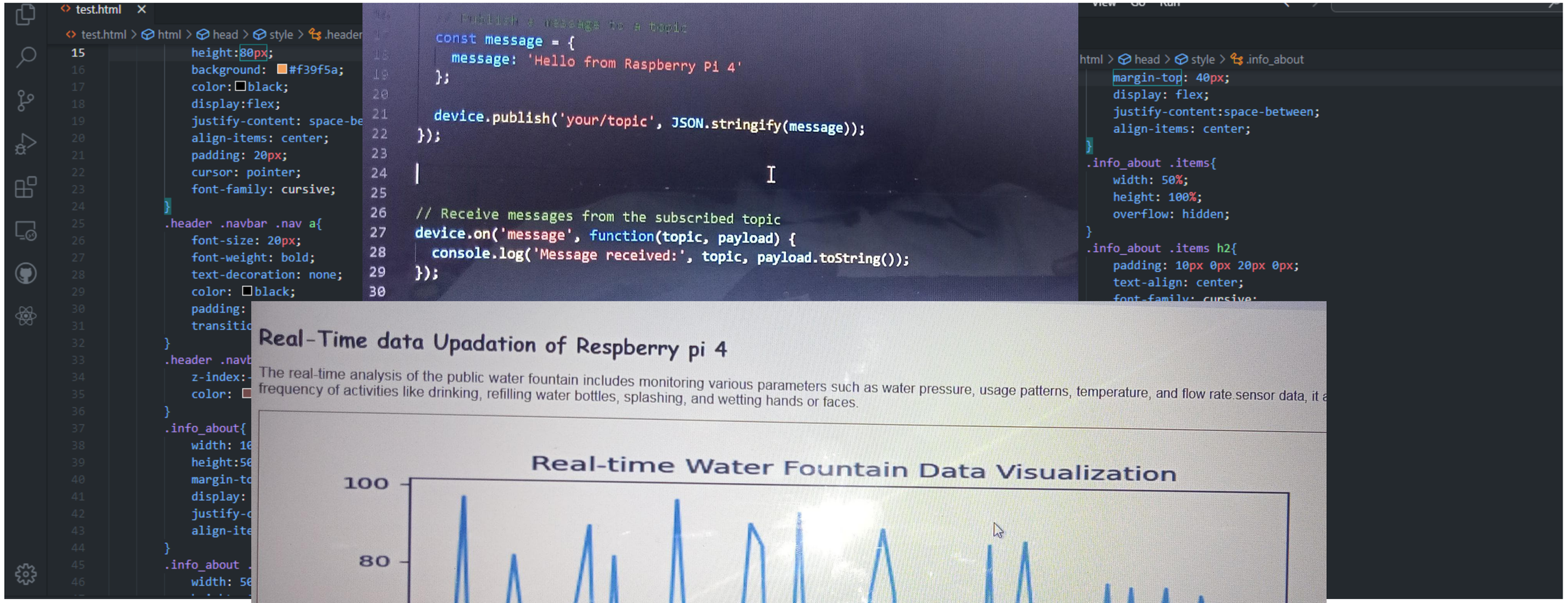
Home About us

Introduction to Respberry pi 4

Raspberry Pi is a series of affordable,credit card-sized single-board computers . Equipped with GeneralPurpose Input/Output (GPIO) pins andsupporting various operating systems.utilized in diverse projects, ranging from basic computing tasks to complex DIY electronics, robotics, anInternet of Things (IoT) applications.Monitor water usage patterns and promote efficient waterconsumption.Control water flow based on real-time usage and demand.Provide users with real-time data and access to control thefountain remotely.Raise awareness about water conservation and environmentalsustainability.Demonstrate the capabilities of IoT and smart technology in promoting responsible water usage.

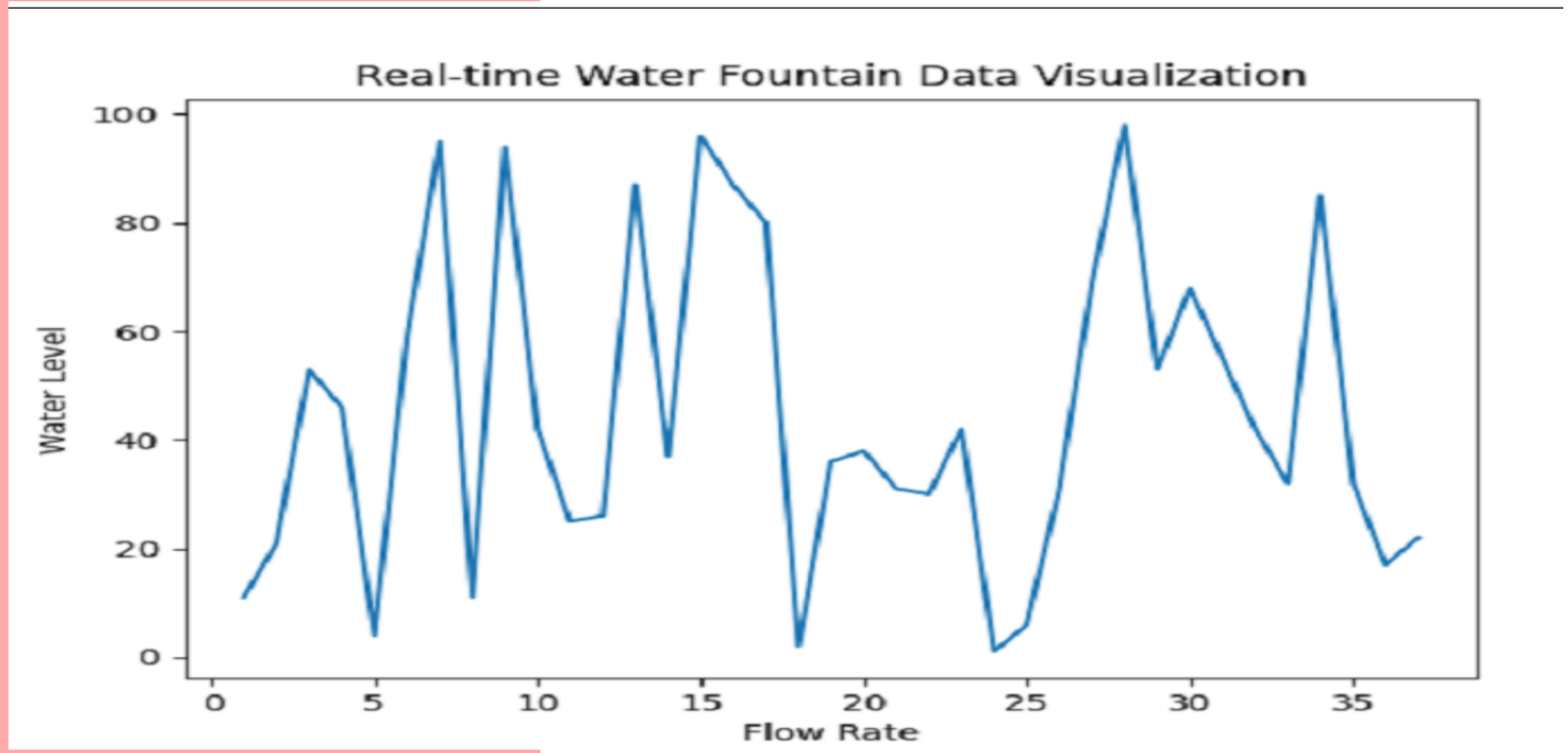


UI Design Output Screenshot's:



Python Script to Route the Server and Display Real-time Upadate

```
from AWSIoTPythonSDK.MQTTLib import
AWSIoTMQTTClient

# AWS IoT configuration
host = "your-aws-iot-endpoint"
root_ca_path = "path/to/root/ca"
certificate_path = "path/to/certificate"
private_key_path = "path/to/private/key"
port = 8883  # standard MQTT port for AWS
IoT

myMQTTClient =
AWSIoTMQTTClient("RaspberryPi")
myMQTTClient.configureEndpoint(host, port
myMQTTClient.configureCredentials(root_ca
path, private_key_path, certificate_path)

# Connect to AWS IoT
myMQTTClient.connect()

# Publish a message
myMQTTClient.publish("myTopic", "Hello fron
Raspberry Pi", 1)

# Disconnect from AWS IoT
myMQTTClient.disconnect()
```



Mobile App UI Design :

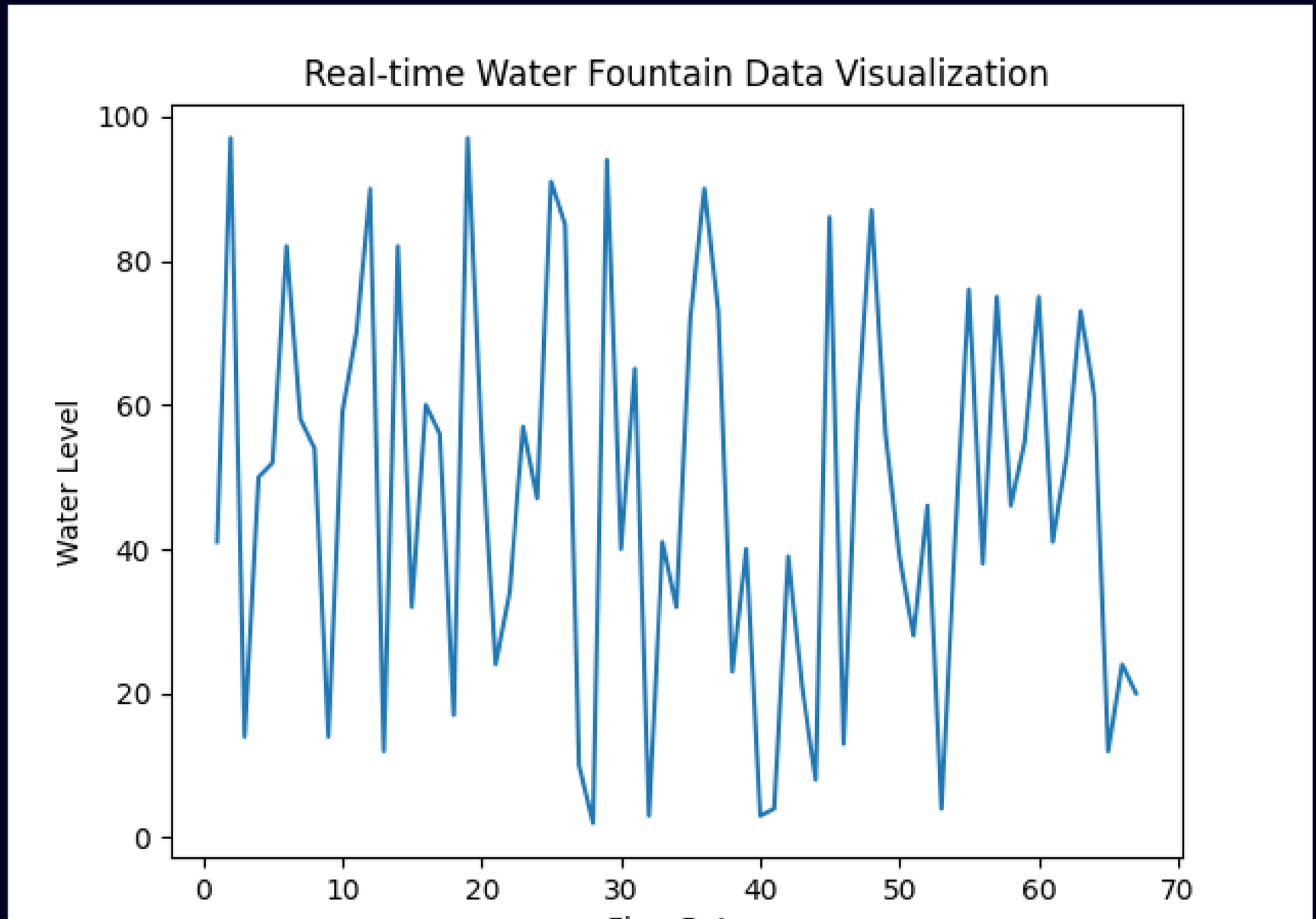
```
File  Edit  Format  Run  Options  Window  Help
from kivy.app import App
from kivy.uix.boxlayout import BoxLayout
from kivy.uix.label import Label

class SmartWaterFountainApp(App):
    def build(self):
        layout = BoxLayout(orientation='vertical')

        # Raspberry Pi data - replace with actual data
        temperature = "25°C"
        water_level = "80%"

        # UI Design for the mobile app
        layout.add_widget(Label(text='Smart Water Fountain Monitoring', fo
        layout.add_widget(Label(text=f'Temperature: {temperature}', font_s
        layout.add_widget(Label(text=f'Water Level: {water_level}', font_s

        return layout

if __name__ == '__main__':
    SmartWaterFountainApp().run()
```

Python Code to Integrate the Respberry Pi 4 and AWS IoT :

This code is a basic template and assumes that you have already set up AWS IoT on your Raspberry Pi and have a Flask application configured for the web page. Please make sure to replace the placeholder values with your actual AWS IoT host, rootCA, certificate, and private key paths.

```
host = "your-aws-iot-hostname"
rootCAPath = "path/to/rootCA.pem"
certificatePath = "path/to/certificate.pem.crt"
privateKeyPath = "path/to/private.pem.key"
my_rpi = AWSIoTMQTTClient("basicPubSub")
my_rpi.configureEndpoint(host, 8883)
my_rpi.configureCredentials(rootCAPath,
privateKeyPath, certificatePath)

# Flask web page configuration
app = Flask(__name__)

# Function to publish data to AWS IoT
def publish_to_aws(data):
    my_rpi.connect()
    my_rpi.publish("topic", data, 0)
    my_rpi.disconnect()

# Flask route to display data on a web page
```

Real-Time Analysis :



Public Awareness :

- The importance of water conservation and sustainable usage through educational campaigns.
- Promote the benefits of using smart water fountains in reducing plastic waste from single-use water bottles.
- Showcase how smart water fountains contribute to the reduction of water waste and overall conservation efforts.
- Educate the public about the technological features and innovations incorporated in smart water fountain designs.
- information on the cost-effectiveness of smart water fountains compared to traditional water dispensing methods.

Conclusion for Smart Water Fountain :

- Smart water fountains represent a critical step towards sustainable and eco-friendly water consumption practices.
- The integration of advanced technology in water fountains has significantly improved accessibility, convenience, and efficiency.
- These fountains have a positive impact on reducing the carbon footprint by minimizing the need for bottled water production and transportation.
- The incorporation of IoT and sensor technology enables real-time monitoring, promoting effective maintenance and conservation of water resources.
- The data collected from these fountains can be used for further research and analysis to improve water management and infrastructure planning.

THANK YOU