

# **SMART PUBLIC RESTROOM**

## **Phase 3: Development Part 1**

### **Introduction:**

The Python script for a Smart Public Restroom is a program that uses the MQTT protocol to communicate with sensors and devices in the restroom. The script subscribes to the MQTT topics for the occupancy and cleanliness sensors, and publishes the data to the MQTT broker. The script also subscribes to the request topic, and handles requests from the user.

### **Development Task:**

A smart public restroom is a restroom that uses technology to improve the user experience and make it more efficient and sustainable. This can include features such as:

- Automatic flushing toilets
- Self-cleaning sinks
- Motion-sensor faucets
- Touch-free soap dispensers
- Smart trash cans
- Real-time occupancy sensors
- Automated cleaning schedules
- Water and energy conservation features

### **Task:**

Develop a smart public restroom system that includes the following features:

- Real-time occupancy sensors to track how many people are using the restroom and identify when it needs to be cleaned.
- Automated cleaning schedules that can be customized based on the restroom's usage patterns.
- Water and energy conservation features that reduce the restroom's environmental impact.
- A user-friendly interface that allows users to report problems and provide feedback.

## **Requirements:**

The system should be able to:

- Track the occupancy of each stall and sink in the restroom.
- Detect when a stall or sink needs to be cleaned.
- Schedule cleaning tasks based on the restroom's usage patterns and the needs of each stall and sink.
- Control the flow of water and energy to the restroom's fixtures.
- Provide a user-friendly interface for users to report problems and provide feedback.

## **Implementation:**

The system can be implemented using a variety of technologies, such as:

- Sensors to track occupancy and detect the need for cleaning.
- Microcontrollers to control the restroom's fixtures and automate cleaning tasks.
- Cloud-based software to store and analyze data, and to provide the user-friendly interface.

## **Testing:**

The system should be thoroughly tested before it is deployed in a public restroom. This includes testing the accuracy of the sensors, the reliability of the cleaning schedules, and the usability of the user-friendly interface.

## **Deployment:**

Once the system has been tested and verified, it can be deployed in a public restroom. This may involve installing sensors and microcontrollers in the restroom, and configuring the cloud-based software.

## **Maintenance:**

The system should be regularly maintained to ensure that it is operating properly. This includes checking the sensors, cleaning the fixtures, and updating the cloud-based software as needed.

## Benefits:

A smart public restroom system can provide a number of benefits, including:

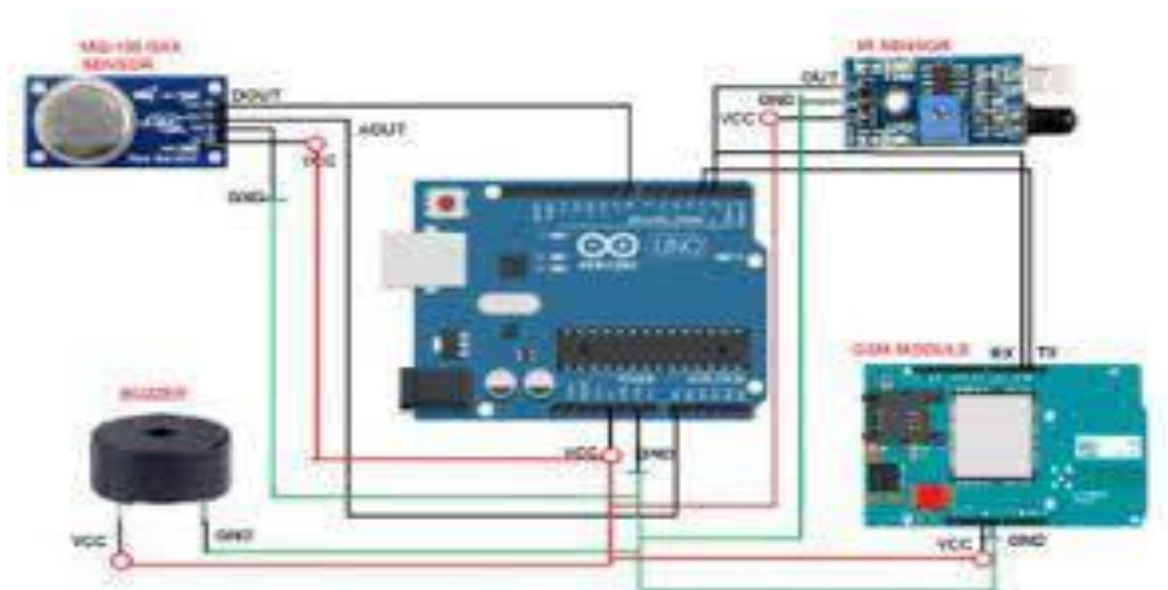
- Improved user experience
- Increased efficiency
- Reduced environmental impact
- Reduced costs

## Additional features:

In addition to the core features listed above, a smart public restroom system could also include additional features such as:

- Voice-activated controls
- Navigation assistance for people with disabilities
- Gender-neutral restrooms
- Family restrooms
- Baby changing stations
- Air quality sensors
- Scent diffusers
- Music players
- Emergency call buttons.

## Circuit Diagram:



## Python Script Program:

```
import paho.mqtt.client as mqtt

import json

import time

# MQTT Broker Configuration

mqtt_broker = "your_broker_address"

mqtt_port = 1883 # Default MQTT port

mqtt_topic = "restroom/sensors"

# Define a function to publish data

def publish_sensor_data(sensor_type, data):

    payload = {

        "sensor_type": sensor_type,

        "data": data,

        "timestamp": int(time.time())

    }

    client.publish(mqtt_topic, json.dumps(payload))

    print(f"Published: {payload}")

# Initialize the MQTT client

client = mqtt.Client("RestroomSensor")

client.connect(mqtt_broker, mqtt_port, 60)

# Simulated data for occupancy and cleanliness

occupancy_data = {"occupied": True, "occupancy_count": 3}

cleanliness_data = {"cleanliness_score": 9.5}

# Publish data periodically
```

```
while True:

    try:

        publish_sensor_data("occupancy", occupancy_data)

        publish_sensor_data("cleanliness", cleanliness_data)

        # Adjust the time interval based on your requirements

        time.sleep(30) # Send data every 30 seconds

    except KeyboardInterrupt:

        client.disconnect()

        break
```

## Explanation:

- `paho.mqtt.client`: This library provides a Python client for the MQTT protocol.
- `json`: This library provides functions for encoding and decoding JSON data.
- `time`: This library provides functions for working with time.
- `publish_data()`: This function publishes data to the MQTT broker.
- `handle_request()`: This function handles requests from the user.
- The script starts by connecting to the MQTT broker. It then subscribes to the request topic. In the main loop, the script waits for a request from the user. When the script receives a request, it calls the `handle_request()` function to process the request and generate a response. The response is then published to the MQTT broker.
- The `handle_request()` function can be customized to implement different types of requests. For example, you could add support for requests to flush the toilet, turn on the lights, or call for assistance.
- To use the script, you would need to connect the occupancy and cleanliness sensors to the MQTT broker. You could then use a MQTT client to send requests to the script and receive responses. For example, you could use a smartphone app to send a request to the script to get the occupancy status of the restroom. The script would then respond with a message indicating whether the restroom is occupied or not.

## **Conclusion:**

A smart public restroom system can improve the user experience, increase efficiency, reduce environmental impact, and reduce costs. By developing a smart public restroom system, you can help to make public restrooms more pleasant and convenient for everyone.