

**KGiSL INSTITUTE OF TECHNOLOGY**

(Approved By AICTE, New Delhi, Affiliate to Anna University

Recognized by UGC, Accredited by NBA(IT)

265, KGISL Campus, Thudiyalur Road, Saravanampatti, Coimbatore-641035**.)**

**DEPARTMENT OF**

**ARTIFICIAL INTELLIGENCE AND DATA SCIENCE**

**NAAN MUDHALVAN - INTERNET OF THINGS**

**SMART PUBLIC RESTROOMS**

**NAME:** GOWTHAM A

**REG NO:** 711721243304

**NM ID:** au21aia68

**TEAM MENTOR:** Mr**.** Mohankumar M

**TEAM EVALUATOR:** Ms. Akilandeeshwari M

**Phase 3: Development Part 1**

**Building a Smart Public Restrooms With IoT sensors**

**Materials and Components Needed:**

1. IoT Sensors
2. IoT Communication Infrastructure
3. Central Server/Cloud
4. User Interface
5. Control System
6. Hygiene Features
7. Cleaning and Maintenance Systems
8. Energy Management
9. Power Supply
10. Security Cameras

**Procedure:**

**1. Define Project Objectives:**

Identify the specific goals and objectives of the smart public restroom project. Determine what features and capabilities you want to incorporate, such as occupancy monitoring, hygiene control, energy efficiency, and maintenance optimization**.**

**2. Site Assessment:**

Conduct a thorough assessment of the site where you plan to build the smart public restroom. Consider factors like location, available utilities (water and power sources), accessibility, and the number of expected users**.**

**3. Design the Restroom Layout:**

Work with architects and designers to create a restroom layout that incorporates the IoT sensor placement, plumbing, electrical systems, and user-friendly features. Ensure that it complies with local building codes and regulations.

**4. Select IoT Sensors:**

Choose the appropriate IoT sensors based on your project objectives. This may include occupancy sensors, toilet flush sensors, water leakage sensors, and hygiene sensors. Ensure they are compatible with the IoT communication infrastructure you plan to use**.**

**5. IoT Communication Infrastructure:**

Select the communication infrastructure, which can include Wi-Fi, cellular, or other wireless technologies. Ensure it provides reliable connectivity within the restroom area and can transmit data to a central server or cloud platform**.**

**6. Choose Control Systems:**

Determine the microcontrollers and actuators needed to control various devices in the restroom, such as automatic flush valves, automatic faucets, and hand sanitizer dispensers.

**7. Central Server/Cloud Platform:**

Set up a central server or cloud platform to collect, store, and process data from IoT sensors. Choose a reliable and secure cloud service provider if you opt for a cloud-based solution.

**8. Develop User Interface:**

Create a user interface, such as a mobile app or a web interface, that allows users to access information about restroom availability, cleanliness, and other features. Ensure that it is user-friendly and provides real-time data.

**9. Install Hardware:**

Install IoT sensors, control systems, actuators, and other hardware components in the restroom according to the layout and design**.**

**10. Implement Security Measures:**

Implement robust security protocols to protect the data being transmitted from sensors to the central server. Security is crucial, especially in public spaces with privacy concerns**.**

**11. Test the System:**

Conduct comprehensive testing to ensure that all sensors, control systems, and communication infrastructure are functioning correctly. Test user interfaces for ease of use.

**12. Maintenance and Support:**

Establish a maintenance plan to regularly inspect and service the sensors and equipment to keep the restroom in optimal condition. This may involve remote monitoring and alert systems for maintenance needs.

**13. User Training:**

Train maintenance staff and users on how to interact with the smart restroom and its features.

**14. Compliance and Regulations:**

Ensure that your smart public restroom complies with local regulations, accessibility standards, and privacy laws**.**

**15. Monitor and Optimize:**

Continuously monitor the system's performance, collect data, and use analytics to optimize resource usage, maintenance schedules, and overall efficiency.

**Python Scripts on IoT**

import time

import random

import requests

# Simulated occupancy sensor function

def simulate\_occupancy\_sensor():

while True:

# Generate a random occupancy status (0 for vacant, 1 for occupied)

occupancy\_status = random.choice([0, 1])

# Send data to the central server (replace with your server endpoint)

server\_url = "http://localhost:5000/api/occupancy"

payload = {"occupancy\_status": occupancy\_status}

try:

response = requests.post(server\_url, json=payload)

if response.status\_code == 200:

print(f"Occupancy status sent: {occupancy\_status}")

else:

print(f"Failed to send occupancy status: {response.status\_code}")

except Exception as e:

print(f"Error: {str(e)}")

# Wait for a defined interval (e.g., 5 seconds) before sending the next status

time.sleep(5)

if \_\_name\_\_ == "\_\_main\_\_":

simulate\_occupancy\_sensor()

from flask import Flask, request, jsonify

app = Flask(\_\_name)

# Store the latest occupancy status

occupancy\_status = 0

# API route to receive occupancy data

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

def update\_occupancy():

global occupancy\_status

data = request.get\_json()

occupancy\_status = data.get('occupancy\_status')

return jsonify({"message": "Occupancy status updated."})

# API route to get the latest occupancy status

@app.route('/api/get\_occupancy', methods=['GET'])

def get\_occupancy():

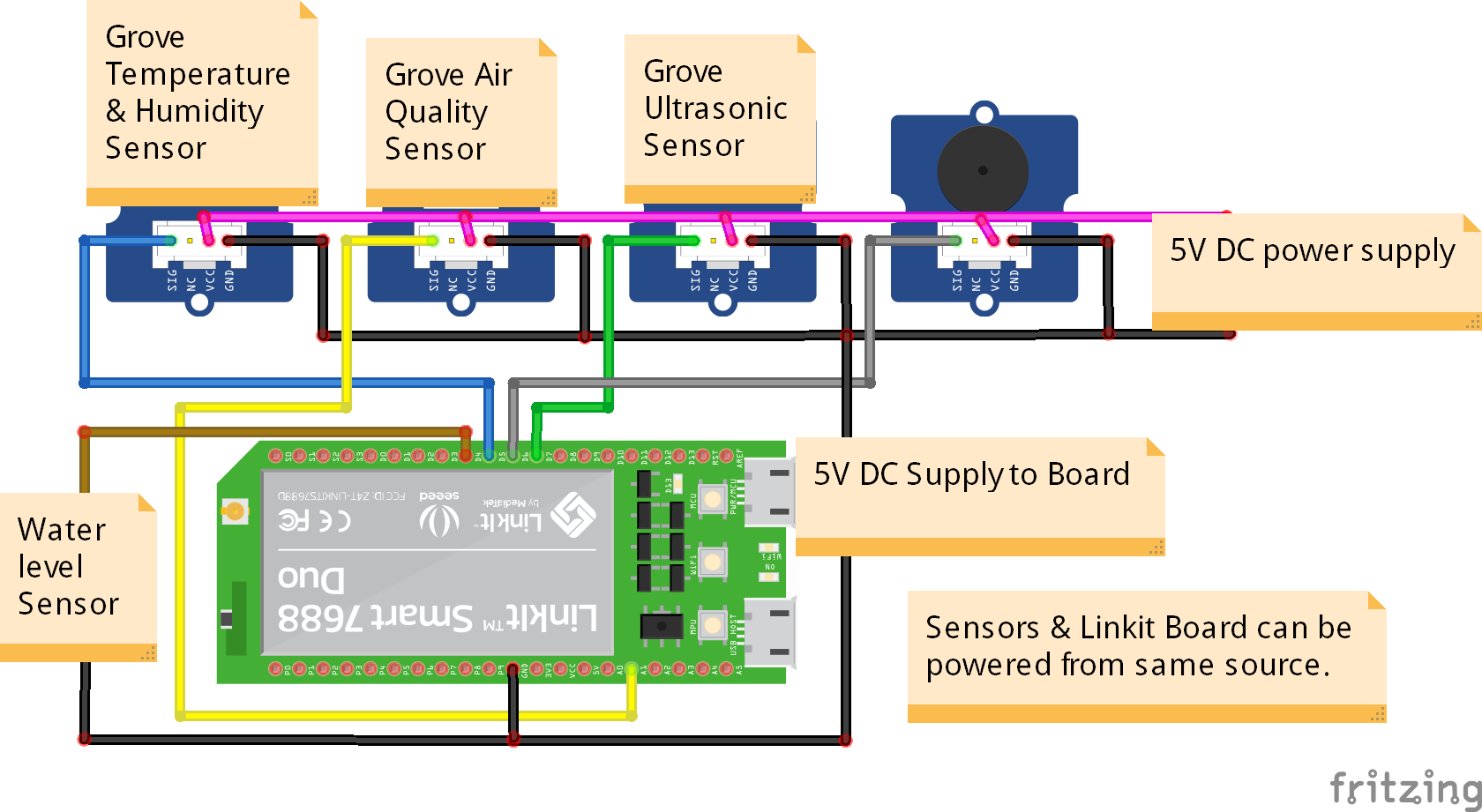
return jsonify({"occupancy\_status": occupancy\_status})

if \_\_name\_\_ == "\_\_main\_\_":

app.run(host='0.0.0.0', port=5000)

python server.py

python sensor.py



**CONCLUSION**:

Building a smart public restroom is a substantial project that involves the integration of various technologies, sensors, and systems. It should be approached with a multidisciplinary team, including experts in IoT, architecture, plumbing, and software development. The goal is to create a restroom that enhances user experience, improves resource management, and ensures cleanliness and accessibility.