SMARTPUBLICRESTROOM-BUILDINGMYPROJECT

INTERNETOFTHINGS-PHASE3-GROUP1–PROJECT MADHA

INSTITUTE OF ENGINEERING AND TECHNOLOGY COLLEGE CODE:

2112

Regno:211221104003

Building a smart public restroom using IoT involves integrating various sensors, devices, and technologies to enhancetheuserexperience,improveefficiency,andensurecleanliness.Here'sacomprehensiveguideforbuilding a project on a smart public restroom using IoT

- **AutomatedEntryandExitSystem**

Implement RFID or biometric-based access control for secure entry and

exit.Usesensorstodetectentryandexit,triggeringappropriateactionsandservices.

- **OccupancyMonitoring**

Deploy occupancy sensors to monitor the usage of different sections of the restroom.

Displayreal-timeoccupancystatusoutsidetherestroomtoguideuserstoavailablefacilities.

- **HygieneMaintenance**

Installautomatedsoapdispensersandwaterfaucetstominimizetouchandimprovehygiene. Integrate

sensors to monitor hand-sanitizer levels and automate refills.

UseUVdisinfectionsystemsorself-cleaningsurfacestomaintainacleanenvironment.

- **SecurityandPrivacyMeasures**

Implementrobustsecurityprotocolstoprotectuserdataandensurethesafety ofIoTdevices.

Adheretoprivacyregulationsandindustrystandardstoprotectuserprivacyanddatasecurity.

CreatingasmartpublicrestroomusingIoTinvolvesintegratingvarioussensorsanddevicestomonitor and control different aspects of the restroom. Here is a basic Python script to demonstrate how to control and monitor some features of a smart public restroom using the MQTT protocol, which is commonly used in IoT applications.

Torunthisscript,you'llneedtoinstallthepaho-mqttlibraryusingpip(pipinstallpaho-mqtt).

Python

Copycode

```
importpaho.mqtt.clientasmqtt
import time

#MQTTsettings
mqtt_broker="broker.example.com"#Updatewithyourbroker'sIPorhostname
mqtt_port = 1883# Default MQTT port
mqtt_keepalive=60#Keep-alivetimeforMQTTconnection

#SensorsandActuators sensor_data
= {
   "temperature":25,#Placeholdertemperaturevalue
   "humidity": 60,# Placeholder humidity value "occupancy":
   False# Placeholder occupancy status
}

#DefineMQTTcallbacks
defon_connect(client,userdata,flags,rc):
```

```
    ifrc==0:
        print("ConnectedtoMQTTBroker!") else:
        print(f"Failedtoconnect,returncode{rc}")

defon_message(client,userdata,message):
    print(f"Receivedmessage:{message.payload.decode()}ontopic{message.topic}")

#InitializeMQTTclient
client=mqtt.Client("SmartRestroom")
client.on_connect = on_connect
client.on_message = on_message
client.connect(mqtt_broker,mqtt_port,mqtt_keepalive)

# Subscribe to relevant topics
client.subscribe("restroom/occupancy")

#SimulatesensordataupdatesandpublishtoMQTTbroker
while True:
    # Simulate sensor data changes
    sensor_data["temperature"]+=1
    sensor_data["humidity"]-= 1
    sensor_data["occupancy"]=notsensor_data["occupancy"]

    #Publishsensordatatorespectivetopics
```

```
client.publish("restroom/temperature",sensor_data["temperature"])

client.publish("restroom/humidity", sensor_data["humidity"])

client.publish("restroom/occupancy",str(sensor_data["occupancy"]))


time.sleep(5)#Updatedataevery5seconds
```

ThisisabasicsimulationofanMQTTclientthatsimulatesdatacomingfromvarioussensorswithinthe restroom. Inareal-worldscenario,you wouldneed tointegratephysicalsensorsto obtain actual data. Additionally, you would need to create a corresponding MQTT broker and handle the incoming messages in a meaningful way. Make sure to replace the placeholder MQTT broker address with the actual address of your broker.

TeamMembers:

1)211221104001

2)211221104002

3)211221104003

4)211221104004

5)211221104005

6)211221104006