SMART PUBLIC RESTROOM

USING IOT

V.R.S.COLLEGE OF ENGINEERING AND TECHNOLGY

ARASUR

SUBMITTED BY

1.M.NANDHITHADOSS

2.V.SARANYA

3.M.SINDHU

4.E.KALKI

Abstract:

In the world, the advances are definitely grown, yet at the same time the cleanliness in our nation is under risk.The abstract of this project is to deliver clean and hygiene toilets. All the public toilets should be clean and hygiene. The project is based on IoT concepts using different sensors like smell sensor, dirt sensor, RFID reader, sonic sensor, Database. Using these materials we are trying to provide the clean toilets and create the awareness among the people.

Project's objectives:

India is a country with a rich linguistic and cultural diversity population. Many people in our country still do not have access to a healthy sanitation system. To maintain health and hygiene, developing a smart latrine system is a great challenge. This has motivated us to provide a proper solution for our people by designing an eco-friendly and hygienic sanitation system. The project aims to enhance public restroom management by installing IoT sensors to monitor occupancy and maintenance needs. The goal is to provide real-time data on restroom availability and cleanliness to the public through a platform or mobile app. This project includes defining objectives, designing the IoT sensor system, developing the restroom information platform, and integrating them using IoT technology and Python. Our project objectives such as real-time restroom availability information, cleanliness monitoring, improved user experience, and efficient restroom. We are plan the deployment of IoT sensors such as occupancy sensors, cleanliness sensors in public restrooms and design a web-based platform and mobile app to display real-time restroom availability and cleanliness data. After the integration of all the subsystems , the entire system will be tested and put into practice physically. It is necessary to mobilize and engage money, resources, people, equipment, and information to implement.

IoT sensor setup:

Setting up IoT sensors for a smart public restroom involves deploying various sensors and devices to monitor different aspects of the restroom's condition and usage. Here's a typical IoT sensor setup for a smart public restroom:

1.Occupancy Sensors:

Install motion or infrared sensors to detect when someone enters or exits the restroom. This data can be used to determine restroom occupancy and trigger cleaning routines.

2.Toilet Usage Sensors:

Equip toilet bowls with sensors that detect usage and flush automatically or send alerts for maintenance if issues are detected.

3.Water Quality Sensors:

Use water quality sensors to monitor the cleanliness and hygiene of the water supply in sinks, showers, and toilets. This can help ensure that the water is safe for use.

4.Toilet Paper and Soap Level Sensors:

Implement sensors that monitor the levels of toilet paper and soap in dispensers. Low levels trigger alerts for maintenance staff to refill supplies.

5.Air Quality Sensors:

Deploy air quality sensors to measure factors like temperature, humidity, and air freshness. Poor air quality can be an indicator of cleanliness issues.

6.Sound and Noise Sensors:

Sound sensors can monitor noise levels within the restroom, and high noise levels might indicate disturbances or issues that need attention.

7.Smart Locks:

Use smart locks with RFID or mobile access for doors and cubicles. These locks can be remotely controlled and monitored for security and usage data.

8.Hand Sanitizer Dispenser Sensors:

Monitor the usage of hand sanitizer dispensers and receive alerts when they need refilling.

9.Water and Energy Usage Sensors:

Track water and energy consumption to promote sustainability and identify anomalies that may indicate leaks or inefficiencies.

10.Connected Waste Bins:

Use IoT sensors in waste bins to monitor fill levels, optimizing waste collection and ensuring cleanliness.

11.Centralized Control and Monitoring:

Develop a central control system that collects, analyzes, and presents data from these sensors, providing real-time insights and control for restroom management.

12.Alerts and Notifications:

Set up alerts and notifications to inform maintenance staff and users of important events or issues.

A well-planned and integrated IoT sensor setup in a smart public restroom can help maintain cleanliness, improve user experience, and optimize resource management. It’s important to ensure data security, privacy, and compliance with relevant regulations when deploying such sensors.

Mobile app development:

Creating a Python program for a complete mobile app with all the specified features for an IoT-based smart public restroom is a complex and extensive task. Below is a simplified Python program outline using the Kivy framework to demonstrate a prototype of such an app. Please note that this is a high-level representation, and actual development requires in-depth knowledge of mobile app development, IoT integration, and various libraries. Additionally, connecting to real IoT devices and GSM services would require additional programming and potentially other languages and libraries.

Program:

```
Import kivy
From kivy.app import App
From kivy.uix.boxlayout import BoxLayout
From kivy.uix.label import Label
From kivy.uix.button import Button
From kivy.uix.textinput import TextInput
Class SmartRestroomApp(App):
    Def build(self):
    # Create the main user interface
    Layout = BoxLayout(orientation=’vertical’)
     # Add UI components
    Occupancy_label = Label(text=”Occupancy: Vacant”)
    Odor_label = Label(text=”Odor Level: Low”)
    Soap_label = Label(text=”Soap Level: Sufficient”)
    Tissue_label = Label(text=”Tissue Paper Level: High”)
    Location_label = Label(text=”Location: Restroom A”)
    Cleanliness_label = Label(text=”Cleanliness: Good”)
    Feedback_label = Label(text=”Provide Feedback:”)
    Feedback_input = TextInput()
```

```
        Reserve_button = Button(text=”Reserve Restroom”)
         # Define event handlers (to be implemented)
        Def on_reserve(instance):
            # Handle reservation logic here
            Pass
            Def on_feedback(instance):
            # Handle feedback submission logic here
            Pass
         # Bind events to UI components
        Reserve_button.bind(on_release=on_reserve)
        Feedback_input.bind(on_text_validate=on_feedback)
        # Add components to the layout
        Layout.add_widget(occupancy_label)
        Layout.add_widget(odor_label)
        Layout.add_widget(soap_label)
        Layout.add_widget(tissue_label)
        Layout.add_widget(location_label)
        Layout.add_widget(cleanliness_label)
        Layout.add_widget(feedback_label)
        Layout.add_widget(feedback_input)
        Layout.add_widget(reserve_button)
        Return layout
If __name__ == ‘__main__’:
    SmartRestroomApp().run()
```

This code provides a basic prototype of a mobile app interface. To create a complete mobile app for an IoT-based smart public restroom with occupancy, odor, soap, and tissue paper level monitoring, location tracking, cleanliness ratings, and GSM integration, you would need to implement real-time data updates, connect to IoT devices, handle user feedback, and connect to GSM services for reservations and notifications. The above code is a starting point for the UI part of the app and requires substantial additional development and integration for full functionality.

Raspberry Pi integration:

Creating a program for Raspberry Pi integration with occupancy, odor, soap and tissue paper level monitoring, location tracking, and cleanliness for an IoT-based smart public restroom is a complex task. Below is a simplified Python program that demonstrates how to monitor occupancy using a PIR sensor and simulate monitoring other parameters. You can expand upon this code to include additional sensors and functionalities:

Program:

```
import RPi.GPIO as GPIO

import time

# GPIO setup

PIR_PIN = 17  # GPIO pin connected to the PIR sensor

GPIO.setmode(GPIO.BCM)

GPIO.setup(PIR_PIN, GPIO.IN)

# Simulated sensor values (replace with actual sensor data)

odor_level = "Low"

soap_level = "Sufficient"

tissue_paper_level = "High"

location = "Restroom A"

cleanliness = "Good"

try:

    print("PIR Module Test (CTRL+C to exit)")

    time.sleep(2)

    print("Ready")

    while True:

        if GPIO.input(PIR_PIN):

            print("Occupied")

            occupancy = "Occupied"

        else:
            print("Vacant")
            occupancy = "Vacant"
       # Simulated sensor data (replace with actual sensor data)
        # You can add code to read actual sensor values here
        print(f"Odor Level: {odor_level}")
        print(f"Soap Level: {soap_level}")
        print(f"Tissue Paper Level: {tissue_paper_level}")
        print(f"Location: {location}")
        print(f"Cleanliness: {cleanliness}")
        # You can send this data to a cloud service or store it locally for further processing
        time.sleep(5)  # Adjust the interval as needed
except KeyboardInterrupt:
    print("Exiting...")
    GPIO.cleanup()
```

This program uses a PIR sensor to detect restroom occupancy and simulates monitoring other parameters such as odor, soap, tissue paper level, location, and cleanliness. To create a complete IoT-based smart public restroom system, you would need to expand this program to include additional sensors, integrate communication with IoT devices, and manage data processing and storage on the Raspberry Pi or in the cloud.

Additionally, you can create a web-based or mobile app to access and display this data in real-time and allow users to provide feedback and make reservations. The actual implementation will depend on your specific requirements and hardware components.

Coding implementation:

Creating a full implementation of an IoT-based smart public restroom with actual sensors and IoT devices is a complex task. Below, I’ll provide a simplified example using Python and a Raspberry Pi to monitor occupancy and send an alert when occupancy is detected. Please note that this is a basic example, and a complete system would require more advanced sensors and components.

In this example, we’ll use a PIR (Passive Infrared) sensor to detect occupancy and notify via email when occupancy is detected. You can expand this system with additional sensors for odor, soap, tissue paper levels, and more.

Program :

```
Import RPi.GPIO as GPIO
Import time
Import smtplib
From email.mime.text import MIMEText
From email.mime.multipart import MIMEMultipart
# GPIO setup
PIR_PIN = 17  # GPIO pin connected to the PIR sensor
GPIO.setmode(GPIO.BCM)
GPIO.setup(PIR_PIN, GPIO.IN)
# Email configuration
Smtp_server = ‘your_smtp_server’
Smtp_port = 587  # Update with your SMTP server’s port
Smtp_user = ‘your_email@example.com’
Smtp_pass = ‘your_password’
Sender_email = ‘your_email@example.com’
Receiver_email = ‘receiver@example.com’
Def send_email(subject, message):
    Msg = MIMEMultipart()
    Msg[‘From’] = sender_email
    Msg[‘To’] = receiver_email
    Msg[‘Subject’] = subject
    Msg.attach(MIMEText(message, ‘plain’))
Try:
        Server = smtplib.SMTP(smtp_server, smtp_port)
        Server.starttls()
        Server.login(smtp_user, smtp_pass)
        Server.sendmail(sender_email, receiver_email, msg.as_string())
        Server.quit()
    Print('Notification email sent successfully.')
    Except Exception as e:
        Print(f'Error sending email: {str(e)}')
Try:
    Print("IoT-Based Smart Restroom Monitoring (Ctrl+C to exit)")
    Time.sleep(2)
    Print("Ready")
    While True:
        If GPIO.input(PIR_PIN):
            Print("Occupied")
            Send_email('Restroom Occupied', 'Occupancy detected in the
restroom.')
        Else:
            Print("Vacant")
        Time.sleep(5)  # Adjust the interval as needed
Except KeyboardInterrupt:
    Print("Exiting...")
    GPIO.cleanup()
```

In this code, we use a PIR sensor to detect occupancy. When occupancy is detected, an email notification is sent. You can add more sensors and functionalities to monitor other aspects of the restroom. Additionally, consider integrating a cloud service for data storage and a mobile app or web interface for user interaction and data visualization.

Web development:

Creating a complete platform for real-time restroom monitoring in Python involves multiple components, including a back end to collect and provide data, and a front end to display it. Here's a simplified example of how to create a basic platform using Python, Flask for the back end, and HTML/CSS/JavaScript for the front end. Please note that this is a simplified version for demonstration purposes and doesn’t include real-time updates:

Back End (Using Flask):

1.Set Up a Flask Application:

Install Flask and create a new Flask web application.

2.Database or Data Source:

You can use a database (e.g., SQLite, PostgreSQL) or an in-memory data structure to store restroom data.

3.API Endpoints:

Create API endpoints in your Flask application to provide real-time restroom data. For example, you might have an endpoint like /api/restroom to get the data.

Front End (HTML/CSS/JavaScript):

1.User Interface:

Design the user interface using HTML and CSS. Create a web page that displays real-time restroom data.

2.JavaScript:

Use JavaScript to fetch data from the back end using API calls (e.g., Fetch API) and update the user interface with real-time data.

Example Back End (Using Flask):

Here’s a simplified example of a Flask back end to get you started. This doesn’t include real-time updates and assumes you’re storing data in-memory:

Program:

```
from flask import Flask, jsonify

app = Flask(__name)

# Simulated restroom data (replace with your data source)

restroom_data = {

   "availability": True,

   "cleanliness": 5,

   "odor_level": "Low",

   "location": "Restroom 1",
    "soap_level": 75,
    "tissue_level": 50,
}
@app.route('/api/restroom', methods=['GET'])
def get_restroom_data():
    return jsonify(restroom_data)
if __name__ == '__main__':
    app.run(debug=True)
```

Front End (HTML/CSS/JavaScript):

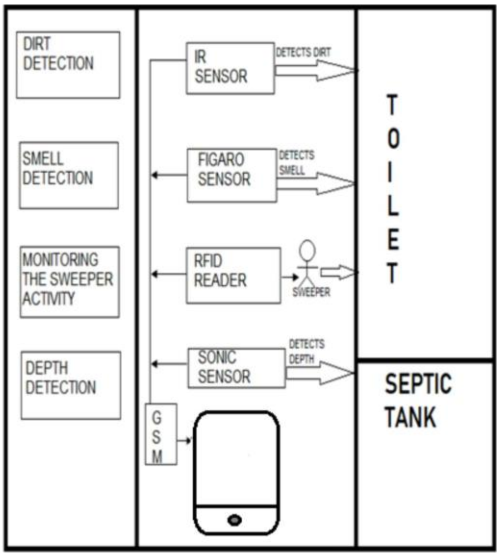
Here’s a basic HTML and JavaScript example to display restroom data. Create an HTML file (e.g., index.html):

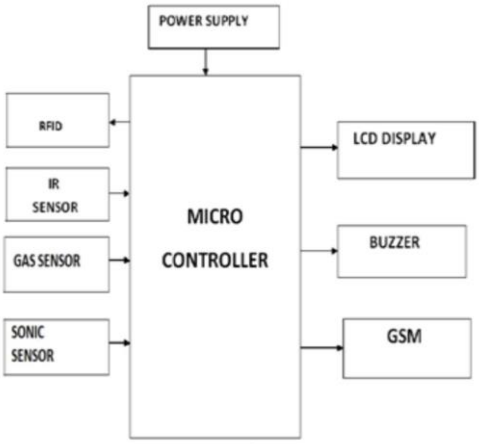
Program:

```
<!DOCTYPE html>
<html>
<head>
    <title>Restroom Monitor</title>
</head>
<body>
    <h1>Restroom Status</h1>
    <div id=”restroom-data”></div>
    <script>
        Function updateRestroomData() {
            Fetch(‘/api/restroom’)
                .then(response => response.json())
                .then(data => {
                    Const restroomDataDiv = document.getElementById(‘restroom-data’);
                    restroomDataDiv.innerHTML = `
                        <p>Availability: ${data.availability ? ‘Available’ : ‘Occupied’}</p>
            <p>Cleanliness: ${data.cleanliness}</p>
            <p>Odor Level: ${data.odor_level}</p>
            <p>Location: ${data.location}</p>
            <p>Soap Level: ${data.soap_level}%</p>
            <p>Tissue Level: ${data.tissue_level}%</p>
          `;
        });
    }
    // Update data every 10 seconds (adjust as needed)
    setInterval(updateRestroomData, 10000);
    // Initial data load
    updateRestroomData();
  </script>
</body>
</html>
```

This example sets up a simple Flask back end and a basic front end for displaying restroom data. Remember, in a real-world scenario, you’d use a database to store and retrieve data, and you’d implement WebSocket support for real-time updates.

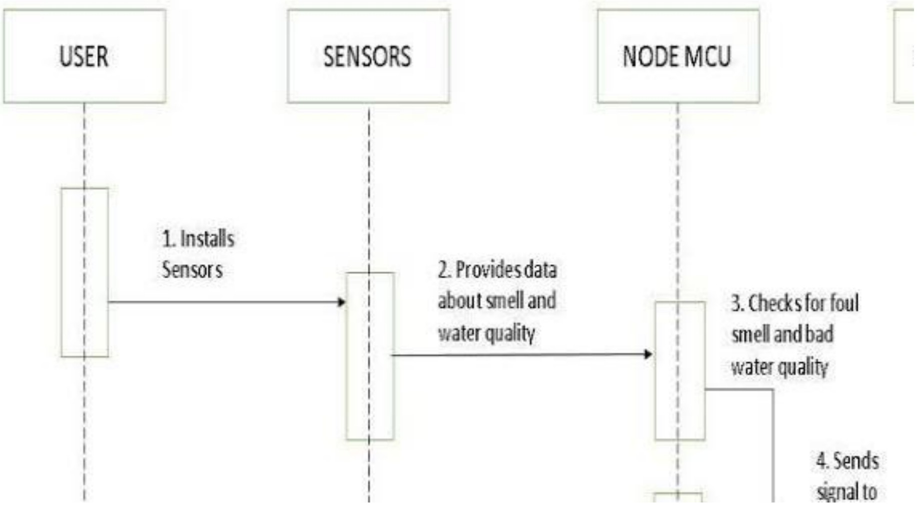
Block diagram:





Architecture diagram:

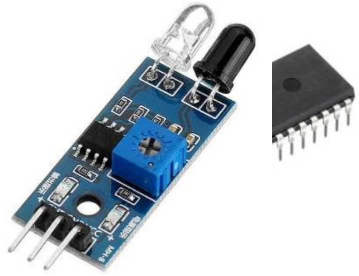
Schematic diagram:


USER
SENSORS
NODE MCU
1. Installs Sensors
2. Provides data about smell and water quality
3. Checks for foul smell and bad water quality
4. Sends signal to

Components:

1.Microcontroller:

A microcontroller is a small computer on a single combined circuit holdingprocessor core, memory and programmable input/output peripherals. Program memory in the form of Ferroelectric RAM, NOR flash or OTP ROM isalso often included on chip, as well as a typically small amount of RAM. Microcontrollers are designed for embedded applications, in contrast to the microprocessors used in personal computers or other general-purpose applications.



PIC 16F877 is one of the most advanced microcontroller from Microchip. This controller is commonly used for experimental and modern applications because of its low price, wide range of requests, high quality, and ease of obtainability. It is ideal for applications such as machine control applications, measurement devices, study purpose, and so on. The PIC 16F877 features all the mechanismswhich present microcontrollers usually have.

2.IR Sensor:

The IR sensor is used to detect the dirt present in the toilet. Here we nourish the image models into the sensor. It can perceive the dirt by comparing the images we feed into it, after using the toilet. If it can detect the dirt, it raises the alarm, and the users may get embraced and they clean it. This system can create theresponsiveness among the people.

3.Smell sensor:

The Smell Sensor is used to detect the unwanted smell and gases in the toilet. For this purpose, we are going to use the sensor called Figaro sensor.
It cansintellect the dry gases present in the toilets such as NH3, CO2, CH4, H2S, etc. By taking those gases leads to Nausea, Drowsiness, instant loss of awareness, etc. After sensing the unwanted gases, it can blink the red light. Then the sweeper can clean it by using particular Cleaning Agents.

4.RFID Reader:

The RFID stands for Radio Frequency Identification. It can be used for monitoring the Sweeper. The Organization wishes to provide the identity tag for the Sweeper. The Sweeper desires toshow the tag before the cleaning process is going to start and after it is finished.

Then the CR4 sensor can spot the presence of dirt. If it is present, it can blink the red light. If it is clean, it can blink the blue light. It assistances to understand the responsibilities of sweeper by his/her own. If Sweeper is not clean the toilets for period of time, his/her absence in cleaning the toilet also reported to the dependable organization. These all the details are stored in the database

5.Sonic sensor:

The Sonic Sensor is used for computingthe depth. Here it is used to measure the depth of the septic tank. The Sonic Sensor is fixed into the Septic tank. Then the Septic tank get filled means, it can sends the communications to particular organization. Then they will allot persons to clean the septic tank. Then septic tank cleaners will clean the tank. After cleaning it, the sensor can detect the level, and send messages to consistent organization.

This ultrasonic sensor can be used for measuring distance, object sensor, motion sensors etc. High sensitive module can be used with microcontroller to integrate with motion circuits to measure the distance, position & motion sensitive products.
In a nutshell, water depth sensing is using a sensor to measure the depth of

water in a tank or container. Although various sensors can be used for this application, we will talk about ultrasonic sensor application.
With ultrasonic sensors, we can find the water depth calculation by finding the distance between the transceiver and the surface of the water. The sensor will transmit a short ultrasonic pulse, and we can measure the travel time of that pulse to the liquid and back. We can then subtract that distance fromthe total depth of the tank to determine the water depth.

6.GSM Module:

GSM stands for Global System for Mobile communication. It establishes the mobile communication from one place to another place.
It transfers the information from main circuit to operator. It uses Time Division Multiple Access (TDMA).
GSM is mainly used for communicating and transferring message from one person to concerned organisation. GSM module is used to establish communication between a computer and a GSM and GPRS system.

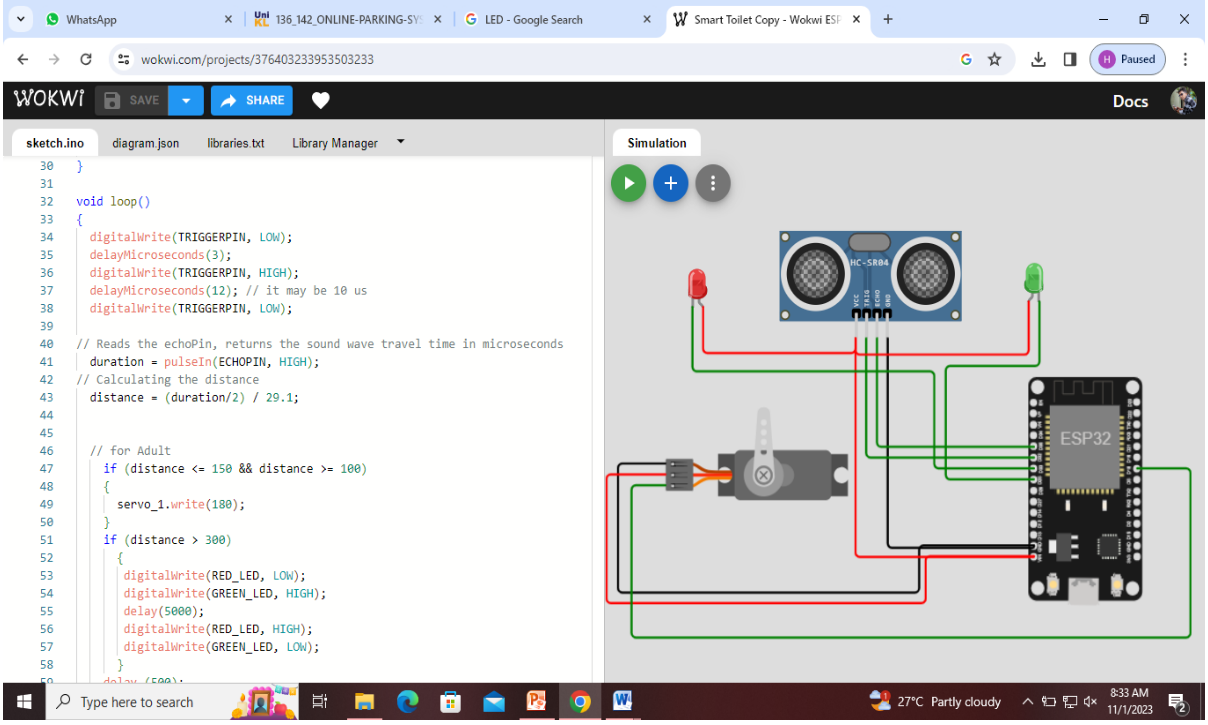
Here we are using GSM LT-2 communication module makes it possible to use GSM paths to provide monitoring and messaging functions in alarm systems. It facilitates cooperation with SATEL and third party control panel diallers or correctly configured outputs.

He GSM LT-2 module makes it possible to implement monitoring as well as text and voice messaging functions. The caller ID retransmission function creates it likely to present the incoming callers number on telecommunication stationsarmed with this functionality.

GSM alarm system built-in GSM communication module inside, work as a mobile handset. After purchased the GSM alarm system, people need to acquisition the SIM card, and select the mobile service package. GSM alarm system can program several phone numbers for alarm receiving. When any abnormal event happens, the system will response, then inform the owner via voice call and short message (SMS).

GSM will check the messaging activities for sweepers and also need to check with their cleanliness duty for their work. The sweepers need to check with particular activity of its work by their sensors

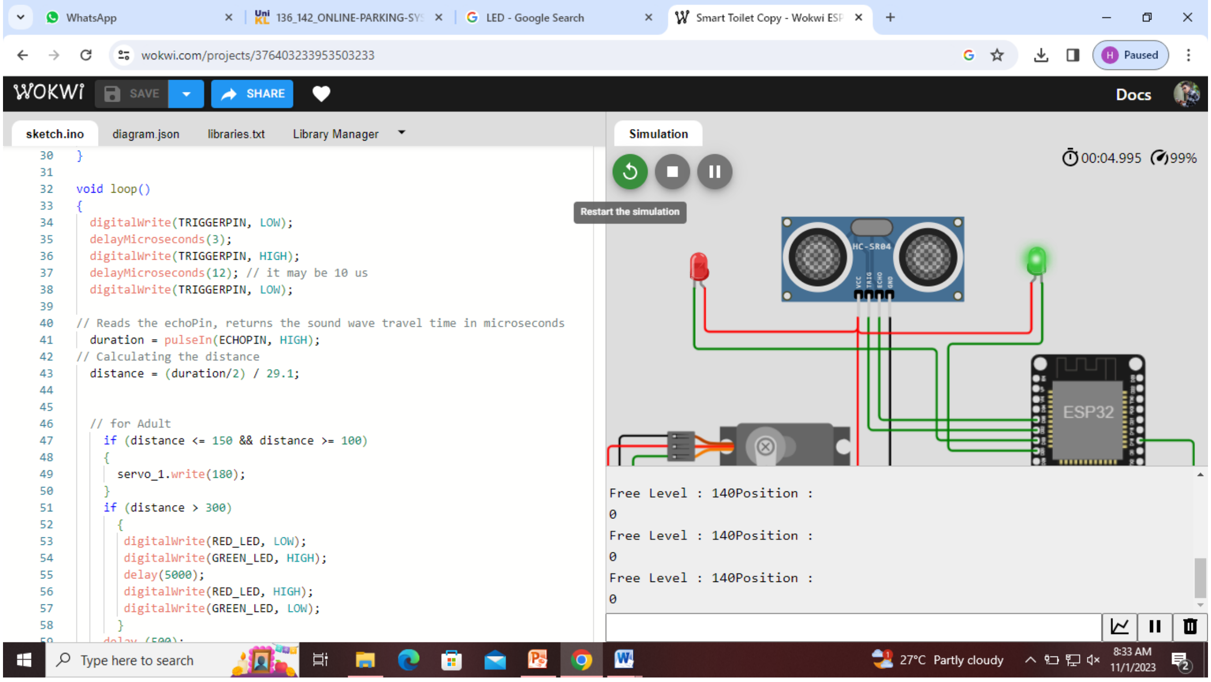
.



```
}

void loop()
{
  digitalWrite(TRIGGERPIN, LOW);
  delayMicroseconds(3);
  digitalWrite(TRIGGERPIN, HIGH);
  delayMicroseconds(12); // it may be 10 us
  digitalWrite(TRIGGERPIN, LOW);

// Reads the echoPin, returns the sound wave travel time in microseconds
  duration = pulseIn(ECHOPIN, HIGH);
// Calculating the distance
  distance = (duration/2) / 29.1;


  // for Adult
    if (distance <= 150 && distance >= 100)
    {
      servo_1.write(180);
    }
    if (distance > 300)
     {
      digitalWrite(RED_LED, LOW);
      digitalWrite(GREEN_LED, HIGH);
      delay(5000);
      digitalWrite(RED_LED, HIGH);
      digitalWrite(GREEN_LED, LOW);
     }
    delay (500);
```

Restroom simulation in wokwi

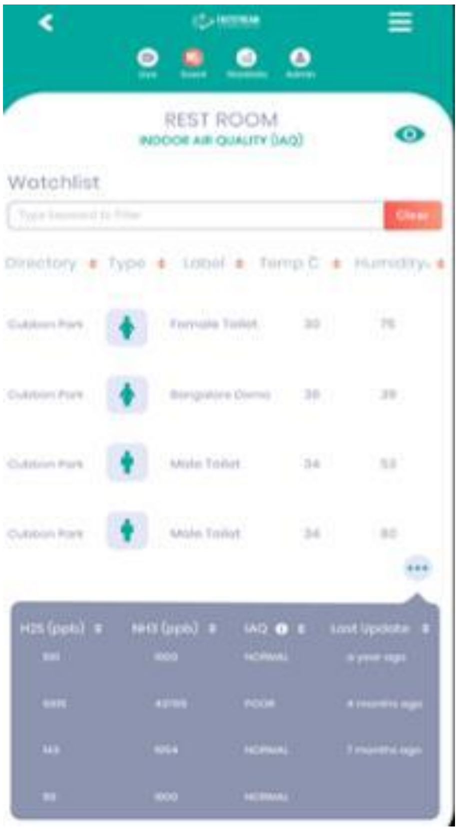
Simulation output



```
}

void loop()
{
  digitalWrite(TRIGGERPIN, LOW);
  delayMicroseconds(3);
  digitalWrite(TRIGGERPIN, HIGH);
  delayMicroseconds(12); // it may be 10 us
  digitalWrite(TRIGGERPIN, LOW);

// Reads the echoPin, returns the sound wave travel time in microseconds
  duration = pulseIn(ECHOPIN, HIGH);
// Calculating the distance
  distance = (duration/2) / 29.1;


  // for Adult
    if (distance <= 150 && distance >= 100)
    {
      servo_1.write(180);
    }
    if (distance > 300)
     {
      digitalWrite(RED_LED, LOW);
      digitalWrite(GREEN_LED, HIGH);
      delay(5000);
      digitalWrite(RED_LED, HIGH);
      digitalWrite(GREEN_LED, LOW);
     }
    delay (500);
```

```
Free Level : 140Position :
0
Free Level : 140Position :
0
Free Level : 140Position :
0
```

Cloud using



Directory	Type	Label	Temp C	Humidity
Cubbon Park		Female Toilet	30	75
Cubbon Park		Bangalore Dome	38	39
Cubbon Park		Male Toilet	34	53
Cubbon Park		Male Toilet	34	80

H2S (ppb)	NH3 (ppb)	IAQ	Last Update
510	1000	NORMAL	a year ago
6915	42705	POOR	4 months ago
143	1054	NORMAL	7 months ago
80	1000	NORMAL	

Real Time Restroom Information System:

A real-time restroom information system, powered by IoT (Internet of Things) technology, can significantly enhance user experience and restroom management for an IoT-based smart public restroom. Here's how:

1.Improved User Experience:

- **Real-Time Occupancy Monitoring:**
  Users can quickly check whether a restroom is occupied or vacant, reducing wait times and enhancing convenience.

- **Reservations:**
  Users can reserve a restroom stall in advance, ensuring they don't have to wait in line, particularly in busy locations.

- **Cleanliness Ratings:**
  Users can provide feedback on restroom cleanliness and hygiene, helping others make informed decisions about whether to use the facility.

- **Odor Monitoring:**
  Real-time odor level monitoring can ensure that users have a better experience by alerting management to address unpleasant odors promptly.

- **Supplies Availability:**
  Users can check the availability of soap, tissue paper, or other supplies before entering the restroom, reducing frustration due to empty dispensers.

- **Location-Based Services:**
  Users can easily locate the nearest available smart public restroom using a mobile app or signage.

- **User Feedback and Alerts:**
  Users can report issues such as cleanliness concerns, damaged fixtures, or malfunctioning equipment, providing a means for timely maintenance and improvements.

2. Enhanced Restroom Management:

- **Optimized Cleaning Schedules:**

Real-time data on restroom cleanliness can help restroom management teams schedule cleaning and maintenance when it’s needed, rather than on fixed schedules.

- **Resource Efficiency:**
  Monitoring soap and tissue paper levels helps ensure that resources are used efficiently, with refills happening precisely when needed.

- **Predictive Maintenance:**
  IoT sensors can predict when equipment (e.g., hand dryers, faucets) requires maintenance, reducing downtime and ensuring a more reliable user experience.

- **Traffic Patterns and Usage Trends:**
  Data collected from IoT sensors can provide insights into traffic patterns, peak usage times, and usage trends, enabling more efficient resource allocation and staffing.

- **Cost Reduction:**
  Efficiently managing supplies and resources based on real-time data can lead to cost savings in terms of maintenance and waste reduction.

- **Emergency Response:**
  In case of emergency situations (e.g., malfunctioning equipment, water leaks), real-time alerts can be sent to maintenance personnel for rapid response.

- **Data-Driven Decision-Making:**
  Management can use historical data to make informed decisions regarding restroom facilities, user satisfaction, and necessary improvements.

3.Sustainability and Resource Efficiency:

- **Reduced Resource Wastage:**
  By tracking soap and tissue paper levels and monitoring water usage, the system can contribute to resource conservation and waste reduction.

- **Energy Efficiency:**
  Smart restroom systems can integrate energy-saving features like motion-activated lighting and water flow controls.

Conclusion:

Our proposed project will create awareness among the people about the proper sanitation. It makes use of Internet of things, which is a rapidly growing technology. Our proposed system will make everyone to strictly follow the cleanliness and proper sanitation in the toilets. It prevents the many new contagious diseases that spread due to improper sanitation of the toilets.Thus by using technologies in the smarter way, we can maintain the cleanliness which is next to the godliness. Keep Clean, Be Safe.