Enhancing Public Restroom Management with IoT Sensors

Phase 5 : Documentation

Introduction:

This pioneering project seeks to transform public restroom management using IoT sensors to provide instantaneous updates on occupancy and cleanliness. The IoT sensors, including occupancy and cleanliness sensors, play a pivotal role in collecting real-time data crucial for decision-making. Utilizing a user-centric platform through a mobile app and a web interface, the project aims to grant easy access to vital restroom information. With Python integration and IoT technology, the project strives to streamline data processing and communication, thereby enhancing user experience and facilitating data-driven decision-making for efficient facility management. The goal is to set new standards in restroom management, emphasizing convenience and cleanliness through innovative technological solutions.

Design Thinking and Objectives:

Certainly, the primary objectives within the design thinking phase are intricately crafted to revolutionize public restroom management using IoT sensors, mobile app integration, and user-centric platforms.
Here's a detailed breakdown of these objectives:

1. Real-Time Restroom Availability Information

Objective: Provide instantaneous and accurate data on restroom availability.
Explanation: By deploying occupancy sensors, the project aims to enable users to access up-to-the-minute information on restroom occupancy. This real-time data empowers users to make informed decisions, ensuring they locate available restrooms promptly.

2. Ongoing Cleanliness Monitoring

***Objective**:* Continuously monitor and assess restroom cleanliness.
Explanation: Employing cleanliness sensors, the project commits to consistent and realtime monitoring of restroom conditions. This sustained assessment aims to ensure a high standard of cleanliness, fostering a pleasant and hygienic restroom environment for users.

3. Enhancing Overall Efficiency

Objective: Streamline processes to elevate efficiency in restroom management.
Explanation: Integrating IoT sensor data and Python technology, the project seeks to optimize resource allocation and decision-making processes. By creating a user-centric platform through mobile app development and a web interface, the initiative aims to improve the overall efficiency of managing public restrooms.

Significance

This approach aims not only to enhance user convenience by providing instant access to vital information but also to streamline facility management processes. The project's multifaceted objectives cater to the diverse needs of users, aiming to ensure both convenience and cleanliness in public restroom facilities. Through innovative technological solutions, the initiative strives to set new benchmarks in public restroom management, ultimately aiming for an elevated standard of user experience and facility efficiency.

IoT Sensor Setup:

The IoT sensor setup is a pivotal phase designed to establish a comprehensive network of sensors within public restroom facilities. This deployment encompasses various sensor types, each serving a specific purpose in capturing and relaying essential data for efficient management.

1. Occupancy Sensors

Purpose: Detect human presence within restroom stalls.
Explanation: These sensors are strategically placed within each stall to detect human presence, instantly providing data on restroom occupancy. By utilizing passive infrared (PIR) sensors, they effectively detect individuals' presence, facilitating the real-time assessment of restroom usage.

2. Cleanliness Sensors

Purpose: Monitor restroom conditions and cleanliness.
Explanation: Cleanliness sensors, which encompass particulate matter sensors, volatile matter sensors, temperature, and humidity sensors, play a crucial role in the continuous assessment of restroom conditions. This real-time monitoring ensures the maintenance of optimal hygiene standards, contributing to a pleasant user experience.

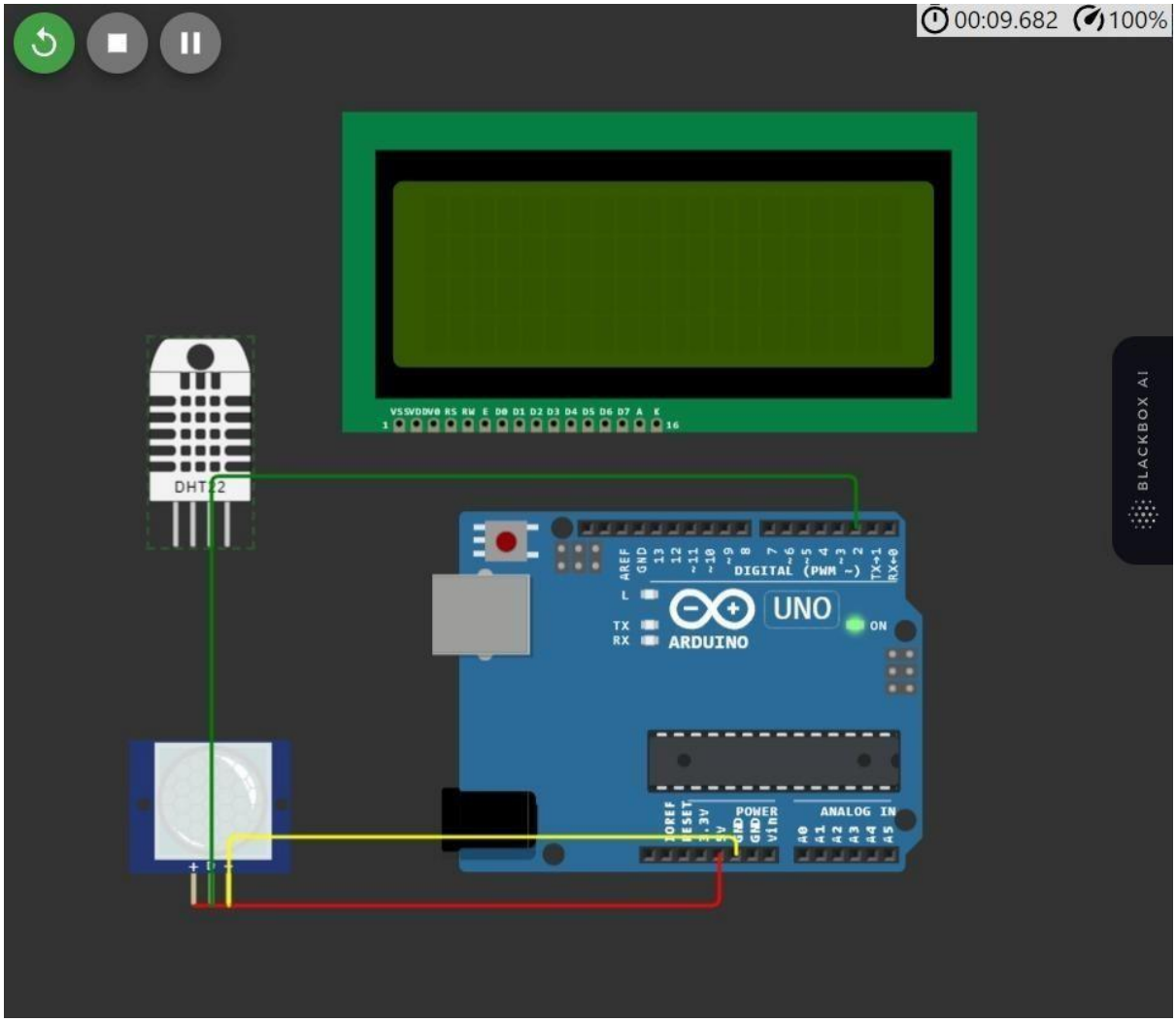
3. Environmental Sensors

Purpose: Monitor environmental factors for holistic insights.
Explanation: These sensors, if integrated, contribute to a comprehensive assessment of restroom environments. By monitoring factors like temperature and humidity, they provide supplementary insights into the overall conditions, ensuring a holistic understanding of the restroom environment.

4. Integration with Raspberry Pi

Purpose: Process and relay sensor data for informed decision-making.
Explanation: The integration of IoT sensors with Raspberry Pi technology is crucial for processing and transmitting data collected by the sensors. Python serves as the backbone for data processing, enabling seamless communication and relaying of crucial information to the user-centric platforms.

This setup is aimed not only at offering real-time insights into restroom occupancy and cleanliness but also at establishing a robust infrastructure for efficient data processing and dissemination, pivotal for informed decision-making in public restroom management.



Real-Time Transit Information Platform:

The development of a real-time transit information platform is a key component of the project, aiming to provide seamless access to crucial data about restroom availability and cleanliness. This user-centric platform involves both a mobile application and a web-based interface, each designed to offer intuitive and instant access to vital restroom information.

1. Mobile App Development

Objective: Provide real-time data access on restroom status for mobile users.
Explanation: The mobile app, crafted for both iOS and Android platforms, is designed to grant users immediate access to real-time restroom information. Leveraging a crossplatform framework like Flutter, it will provide a user-friendly interface for users to check restroom availability and cleanliness status, enabling informed decision-making.

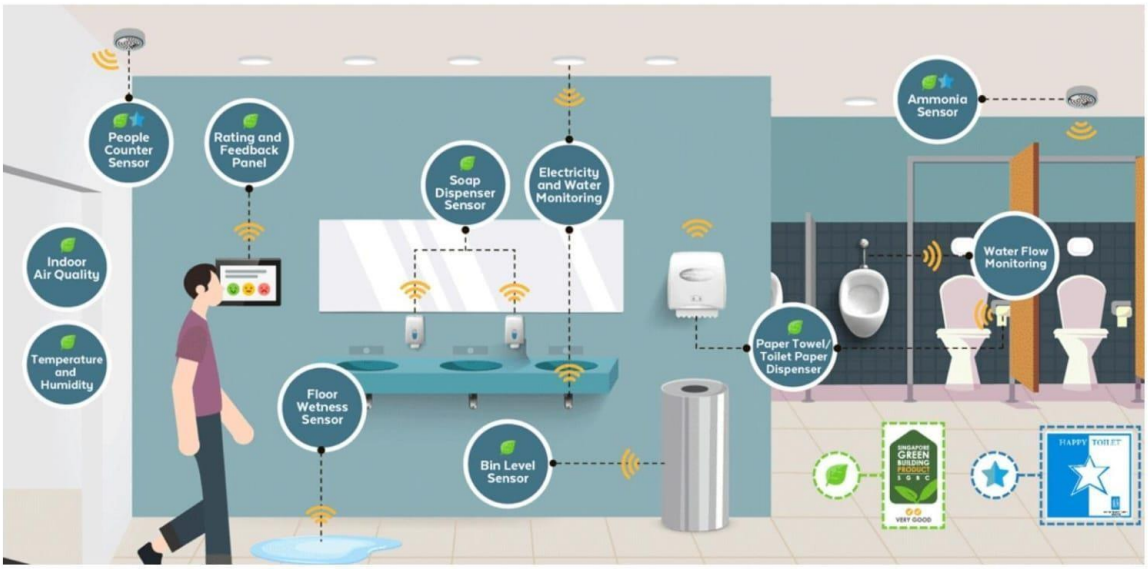
2. Restroom Information Platform

Objective: Offer a user-friendly web-based platform for real-time updates. ***Explanation***: The web-based platform is designed to provide real-time updates on restroom availability and cleanliness. Its user interface will be straightforward, granting easy access to vital information. Integrated mapping and navigation features will guide users to the nearest available restrooms, ensuring convenience and facilitating user decision-making.

3. Integration with Sensor Data

Purpose: Integrate sensor data for real-time updates.
Explanation: The integration of sensor data collected by occupancy and cleanliness sensors, processed through Python and Raspberry Pi integration, is crucial. This integration enables the provision of real-time data updates to both the mobile app and the web platform, ensuring users have immediate and accurate information for their restroom needs.

This real-time transit information platform plays a vital role in the project's objective of enhancing user experience and restroom management efficiency. By offering immediate access to pertinent data and user-friendly interfaces, the platform strives to set new standards in public restroom management, emphasizing both convenience and cleanliness for users and facility managers alike.



Code Implementation:

The code implementation phase encompasses the development and execution of scripts and applications vital for the project's functionality. It involves the creation of various scripts and codes that simulate sensor data, power the web-based platform, and provide real-time updates via the mobile application. These codes are meticulously designed to ensure seamless integration and data transmission across the entire system.

1. Python Script Simulation

This Python script is a critical component simulating sensor data to mimic realtime conditions within the restroom facilities. By using Raspberry Pi and Python, it generates data that mirrors occupancy levels, cleanliness status, and environmental factors. This simulation enables testing and validation of the system's functionality.

```
from wokwi.arduino import WokwiBoard, DHT22, LiquidCrystal
from wokwi.components import PIRMotionSensor board =
WokwiBoard()
pir_sensor = PIRMotionSensor(board, "D2") dht22 = DHT22(board, "D3")  lcd =
LiquidCrystal(board, rs="D7", en="D6", d4="D5", d5="D4", d6="D8", d7="D9",
cols=16, rows=2)  occupancy = 0 cleanliness = 10 def handle_motion_detected(pin):
global occupancy  occupancy = 1  update_display()
pir_sensor.set_callback(handle_motion_detected)  def update_display():
lcd.clear()
lcd.print("Restroom Status:")
if occupancy == 1:
lcd.setCursor(0, 1)
lcd.print("Occupied")  else:
lcd.setCursor(0, 1)
lcd.print("Vacant")

def check_cleanliness():
global cleanliness
cleanliness -= 1  if
cleanliness < 0:
cleanliness = 0
update_display()
board.simulation.setup()
while True:
board.simulation.loop()
dht22.set_temperature(25)
dht22.set_humidity(40)
check_cleanliness()
```

2. HTML, CSS, and JavaScript Integration

The HTML, CSS, and JavaScript code is responsible for powering the web-based platform's interface. These languages collectively ensure a user-friendly design, enabling the real-time display of information on restroom status. JavaScript facilitates dynamic updates, while CSS stylizes the platform for an enhanced user experience.

HTML

```
CTYPEhtml>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=l
<title>Smart Restroom Status</title>
<link rel="stylesheet" href="styles.css">
</head>
<bodp
<header>
<hl >Smart Restroom Status</hl >
</header>
<main>
<section id="restroom-status">
<h2>Restroom Availability and Cleanliness</h2>
<div id="availability"> <h3>AvailabilitY</h3>
id="occupancy">Loading...</p>
</div>
<div id="cleanliness">
<h3>Cleanliness</h3> id="cleanlinessstatus">Loading...</p>
</div>
</section>
</main>
<script src="script.js"></script>
</bodp </html>
```

CSS

```
body { font-family: Arial,  sans-serif;
margin: O;
```

```
padding: O; backgroundcolor: #fOfOfO; header
{ backgroundcolor: #333; color:  white; text-
align:
center; padding: Irem;
main { max-width:
800px; margin: O
auto; padding:  2rem;
font-size: 24px; #restroom-status {
background-color:  white; border: 1 PX
solid #CCC; borderradius: 5px; padding:
Irem; box-shadow: 0
2px 4px rgba(O, O, O,
0.1);
#availability, #cleanliness {  margin:
1 rem O;
#occupancy, #cleanliness-status  {
font-size: 18px; font-weight:
bold;
```

JavaScript

```
document.addEventListener("DOMContentLoaded", () => { const  occupancyElement
= document.getElementByld("occupancy"); const  cleanlinessElement =
document.getElementByld("cleanliness-status");
// Function to update the UI with sensor data const updateUl =  (occupancy,
cleanliness) => { occupancyElement.textContent =
Occupancy: ${occupancyF , cleanlinessElement.textContent =
Cleanliness: ${cleanliness}' ,
// Simulate fetching sensor data (replace with real data)
// In a real scenario, you would make an AJAX or fetch request to your server to get
sensor data. setTimeout(() { const  simulatedData = {  occupancy: "Occupied",
cleanliness: "Clean",
updateUl(simulatedData.occupancy, simulatedData.cleanliness);
}, 2000); // Simulate a 2-second delay for data retrieval
```

3. Mobile App Development

The mobile app, created using Flutter, allows users access to real-time data. Through Dart programming, the app provides a seamless interface for users to check restroom availability and cleanliness status. This development ensures consistent and intuitive functionality across both iOS and Android platforms.

The comprehensive code implementation serves as the backbone of the project, ensuring that users have immediate access to critical restroom information through both the webbased platform and the mobile application. The integration of Python, HTML, CSS, JavaScript, and Flutter technologies ensures a seamless, user-centric experience while striving to elevate the standards of public restroom management.

```
import 'package:flutter/material.dart'; void
main() {
runApp(RestroomlnfoApp());
class RestroomlnfoApp extends StatelessWidget {
@override
Widget build(BuildContext  context)
{ return MaterialApp(  home:
RestroomlnfoScreen(),
class RestroomlnfoScreen extends StatefulWidget {
@override
_RestroomlnfoScreenState createState() => _RestroomlnfoScreenState(); class
_ RestroomlnfoScreenState extends
State<RestroomlnfoScreen> { String occupancyStatus =
"Loading..."; String cleanlinessStatus = "Loading. //
Function to fetch and update restroom information  void
refreshRestroomlnfo() {
// Simulate fetching data from your server or IOT
sensors setState(() { occupancyStatus = "Occupied"; //
Replace with real data cleanlinessStatus = "Clean"; //
Replace with real data
@override
Widget build(BuildContext
context) { return Scaffold(
appBar: AppBar( title:
Text('Restroom Info App'), body:
Center( child: Column(
mainAxisAlignment:
MainAxisAlignment.center, children:
<Widget>[
Text('Restroom Availability: $occupancyStatus'),
Text('Cleanliness Status:
$cleanlinessStatus'), ElevatedButton(  onPressed:
refreshRestroomlnfo, child:
Text('Refresh'),
),
],
),
```

),
); }
}

Real-Time Restroom Information System:

The Real-Time Restroom Information System is a comprehensive framework that operates through a sequence of interconnected elements, forming a cohesive and dynamic system aimed at enhancing user experience and elevating the efficiency of restroom management.

System Components:

1. IoT Sensor Network

Occupancy Sensors: Strategically placed sensors detect human presence within restroom stalls, providing instant data on occupancy levels.
Cleanliness Sensors: Monitor restroom conditions in real time, ensuring and relaying information on cleanliness standards.
Environmental Sensors: Monitor additional factors like temperature and humidity, contributing to a comprehensive understanding of the restroom environment.

2. User-Centric Platform

Mobile App: Developed for both iOS and Android, granting users immediate access to real-time restroom data for informed decision-making.
Web-Based Interface: Provides continuous updates on restroom status, with features such as mapping and navigation for user convenience.

3. Python & Raspberry Pi Integration

Python Scripts: Simulate and process sensor data, providing real-time updates on occupancy and cleanliness statuses.
Raspberry Pi Integration: Functions as a hub for data processing and transmission to the user-centric platform.

System Functionality:

Immediate Data Access: Users can instantly access restroom availability and cleanliness information through the user-centric platform.

Continuous Monitoring: The IoT sensor network ensures ongoing assessment and relaying of critical data to the platform.
Seamless Integration: Python and Raspberry Pi enable smooth communication and realtime updates on restroom conditions.
Enhanced User Experience: The system aims to set new standards by offering convenience and cleanliness for both users and facility managers.

The Real-Time Restroom Information System is a fusion of technology, data processing, and user-centric design, revolutionizing public restroom management. Through its multifaceted approach, it aims to redefine standards, ensuring convenient, clean, and efficient restroom experiences for all stakeholders involved.

Conclusion:

The Real-Time Restroom Information System symbolizes a revolutionary stride in public restroom management, amalgamating IoT sensor networks, a user-centric platform, and refined code implementation. This innovative system excels in providing instantaneous and precise data on restroom occupancy and cleanliness, fostering a paradigm shift in user experience and facility management. By integrating an array of sensors—occupancy, cleanliness, and environmental—the system ensures ongoing surveillance and rapid data dissemination. The user-centric platform, comprising a mobile app and web interface, grants users prompt access to critical restroom information, while Python and Raspberry Pi integration seamlessly process and relay data. Not just emphasizing user convenience, this system streamlines facility management by aiding in efficient resource allocation. Its continuous monitoring and intuitive design set new benchmarks in cleanliness and accessibility, pioneering a techdriven transformation in public restroom management, ensuring a superior and usercentric experience for all stakeholders involved.