Smart Public Restroom

phase-05

# Smart Restroom Project – Problem Understanding and Design Approach

Problem Statement:

user experience and optimizes resource utilization. Key objectives include:

• Efficient Resource Management: Ensure water, energy, and supplies are used judiciously.

• User Convenience: Provide a seamless, user-friendly experience within the restroom.

• Maintenance and Monitoring: Implement systems for real-time monitoring and predictive maintenance.

User Experience Enhancement

To improve user experience, we’ll focus on the following areas:

• Automated Entry: Implement a touchless entry system using sensors or RFID technology.

• Occupancy Monitoring: Utilize occupancy sensors to track restroom usage for maintenance and cleaning scheduling.

• Smart Fixture Controls: Incorporate motion-sensing technology to activate faucets, soap dispensers, and hand dryers, minimizing touchpoints.

Monitoring and Maintenance:

Real-time monitoring will be established through:

• IoT Sensors: Deploy a network of sensors to collect data on restroom conditions, including foot traffic, water and energy usage, and supply levels.

• Predictive Maintenance: Utilize machine learning algorithms to analyze sensor data for predictive maintenance, flagging potential issues before they escalate.

Hardware Components:

• Occupancy Sensors: Deploy throughout the restroom to monitor user presence.

• Smart Fixtures: Incorporate motion-activated faucets, soap dispensers, and hand dryers.

• RFID/Touchless Entry System: Enable touchless entry for enhanced hygiene and security.

Software Integration:

• IoT Platform: Utilize a cloud-based IoT platform for data collection, storage, and analysis.

• Machine Learning Algorithms: Develop algorithms for predictive maintenance based on sensor data.

• User Interface: Design a user-friendly interface for monitoring and control, accessible via mobile or desktop.

Security and Privacy:

• Data Encryption: Implement strong encryption protocols to protect user data.

• Access Controls: Ensure only authorized personnel can access sensitive system functions.

Conclusion:

This document outlines the understanding of the problem and provides a structured approach to designing and implementing the smart restroom project.

### Innovation:

Once you have all of the necessary components, you can start implementing the smart restroom system. The following steps are involved:

1.Install the required components in the restroom and monitor the cleanliness of the restroom.

2.To ensure that the components are fitted/connected in the restroom all the sensors working properly.

3.To monitor and the maintain through embedded system and Iot devices and to maintain the quality of the restroom

**Circuit Components:**

1. Occupancy Sensor for Lighting Control

**Components:**

Passive Infrared (PIR) Sensor: 

Relay Module:



LED Lighting :



**Circuit Description:**

The PIR sensor detects motion in the restroom.

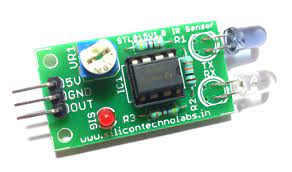
When motion is detected, the sensor sends a signal to the microcontroller.

The microcontroller activates the relay module, which controls the LED lighting.

**2. Automated Faucet Control**

**Components:**

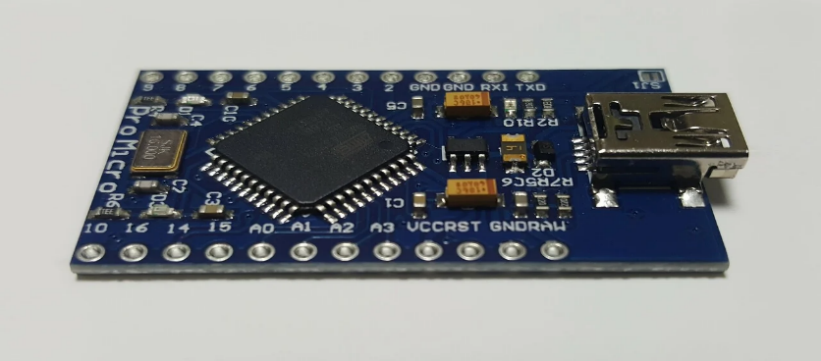
IR Sensor (for proximity detection):



Solenoid Valve:



Microcontroller:



**Circuit Description:**

The IR sensor detects the presence of a user's hand near the faucet.

The microcontroller processes the signal and activates the solenoid valve to allow water flow.

The microcontroller sets a timer for water flow duration to conserve water.

**3. Smart Toilet Flushing System**

**Components:**

Ultrasonic Sensor:

****

Solenoid Valve

Microcontroller

**Circuit Description:**

The ultrasonic sensor measures the distance to detect if the toilet is occupied.

If occupied, the microcontroller prevents flushing until the toilet is vacant.

When vacant, the microcontroller activates the solenoid valve to flush the toilet.

**4. Hand Sanitizer Dispenser**

**Components:**

IR Sensor (for hand detection)

Motorized Dispenser

Microcontroller

**Circuit Description:**

The IR sensor detects the presence of a user's hand.

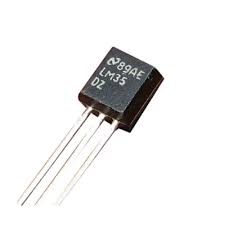
The microcontroller triggers the motorized dispenser to dispense sanitizer.

**5. Air Quality and Temperature Sensor**

**Components:**

Air Quality Sensor (e.g., CO2, VOC)



Temperature Sensor

Microcontroller

**Circuit Description:**

The sensors monitor air quality and temperature.

The microcontroller processes the data for analysis or displays it on a screen for users.

**6. Display and User Interface**

**Components:**

LCD Display



Microcontroller

Push Buttons (for manual overrides or settings)

**Circuit Description:**

The microcontroller communicates with the LCD display to provide feedback to users.

Push buttons allow manual control or adjustment of settings.

**7. Communication Module (Optional)**

**Components:**

Wi-Fi or Bluetooth Module

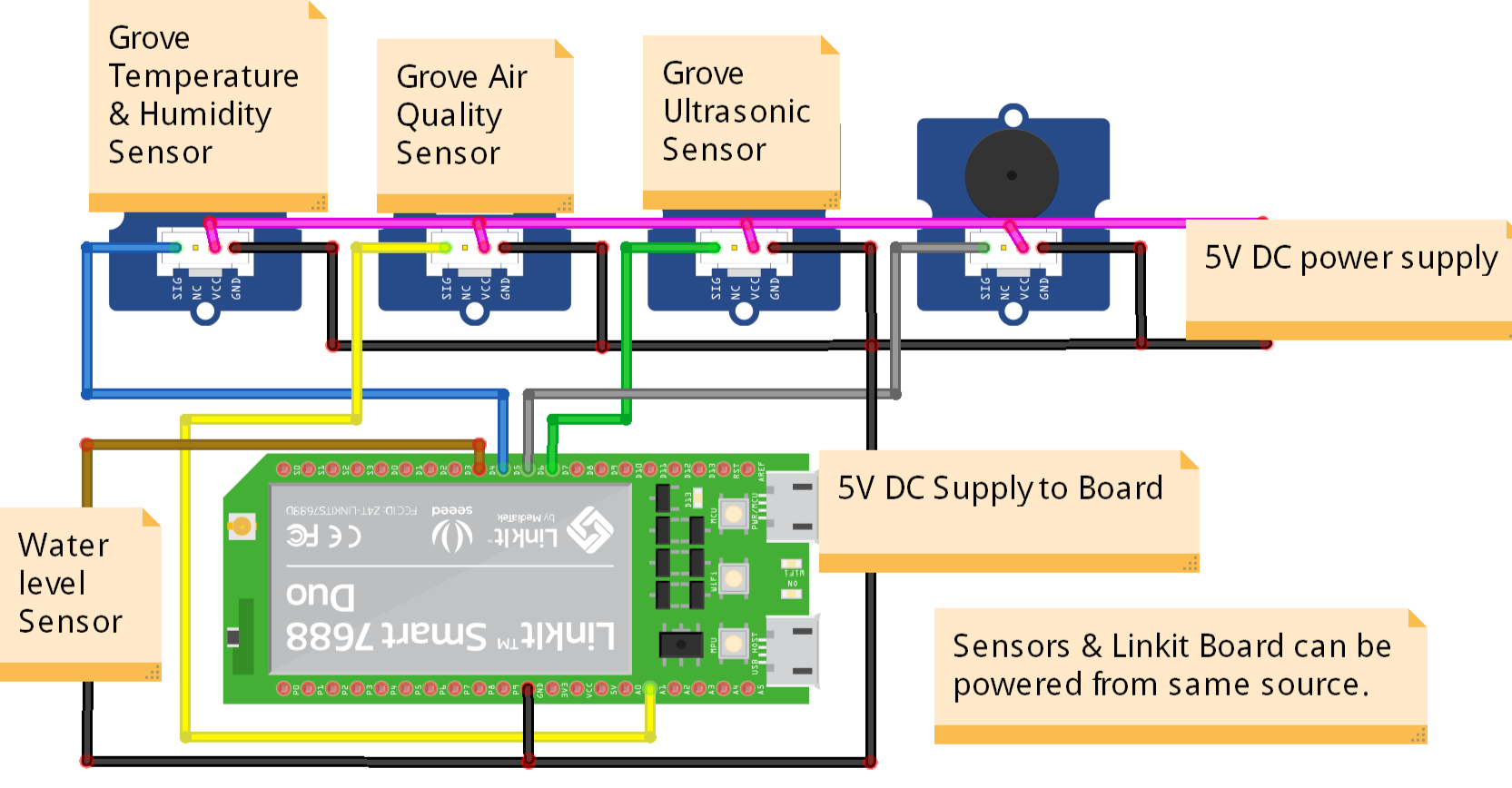
Microcontroller

**Circuit Description:**

The communication module allows the restroom system to connect to a central control or be accessed remotely for monitoring and control.

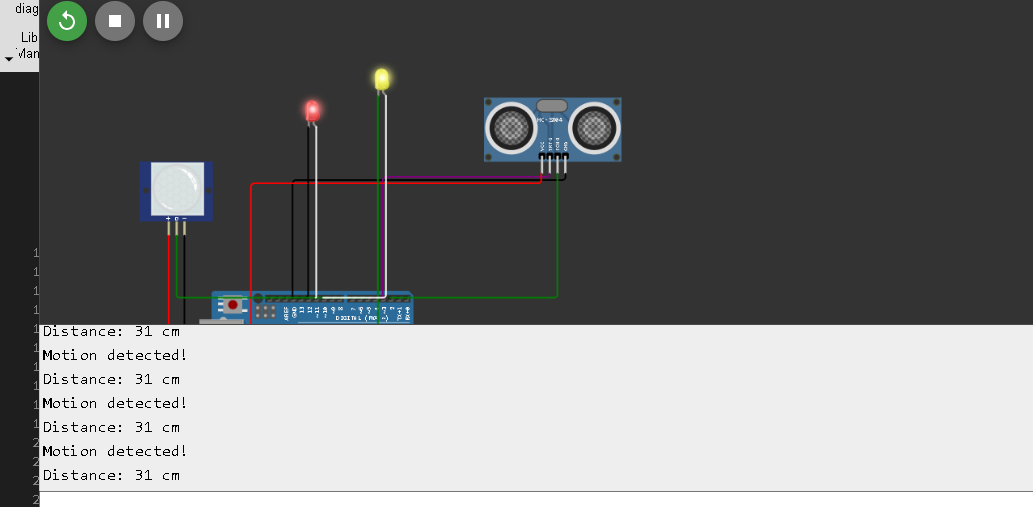
## THE SIMPLE CIRCUIT DIAGRAM FOR SMART RESTROOM SYSTEM

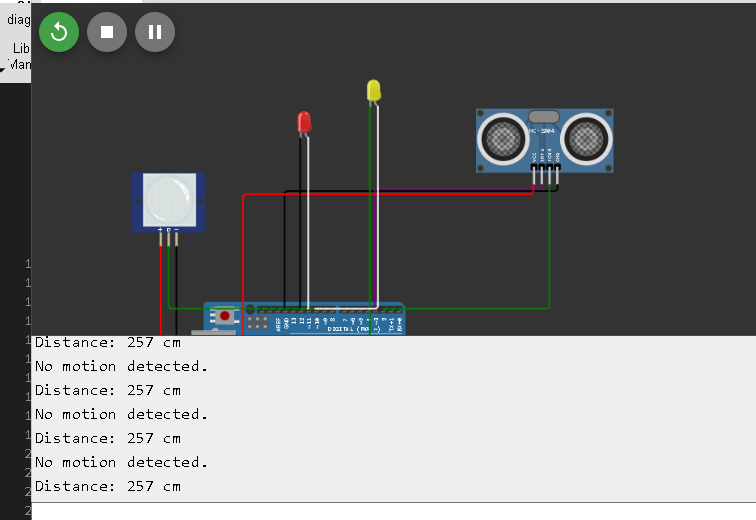
The following is a simple circuit diagram for a smart restroom system using a microcontroller:

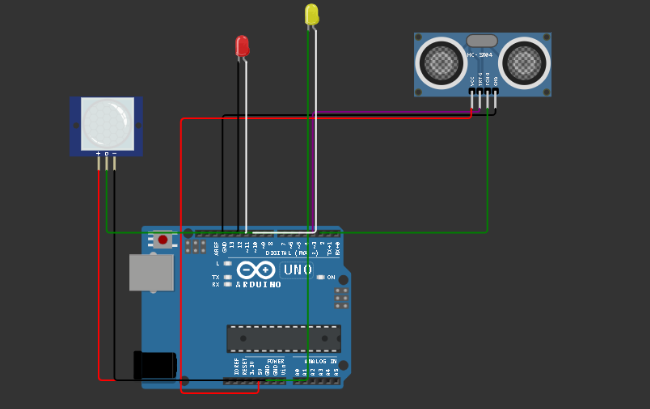


This a simple and basic circuit diagram of the smart restroom using iot.

Here the simulation and the output simple basic circuit with the pir and ultra sonic sensors in the simulation.

This picture denotes the simulation that occurs in the pir and ultra sonic sensor the pir sensor detects the motion and the ultra sonic sensor detects the distance .the pir sensor detects the motion the light gets turn on. if the motion does not detected it gets turned off. The minimum distance that set is detected and the distance is below or above the value the light gets turned on . if it doesn’t attain the required distance it gets turned off. The below picture show the no motion detected and the distance is set to above level.





This is the basic circuit of the sensor and Arduino board pir senor ground positive and negative terminals are connected out center and vcc are connected to board, then the ultrasonic sensor is connected vcc, trigger, echo, ground terminals are connected to the board. The Arduino board got power supply and the led are connected to ultra-sonic and pir to detect the motion and distance.

Coding for this simulation in python

import RPi.GPIO as GPIO

import time

# Pin Definitions

pir\_pin = 5 # PIR sensor pin

trig\_pin = 3 # Ultrasonic sensor Trig pin

echo\_pin = 2 # Ultrasonic sensor Echo pin

pir\_led = 11 # LED for PIR sensor

ultrasonic\_led = 10 # LED for Ultrasonic sensor

# Setup GPIO

GPIO.setmode(GPIO.BOARD)

GPIO.setup(pir\_pin, GPIO.IN)

GPIO.setup(trig\_pin, GPIO.OUT)

GPIO.setup(echo\_pin, GPIO.IN)

GPIO.setup(pir\_led, GPIO.OUT)

GPIO.setup(ultrasonic\_led, GPIO.OUT)

def read\_pir():

pir\_value = GPIO.input(pir\_pin)

GPIO.output(pir\_led, pir\_value) # Turn PIR LED on if motion is detected

if pir\_value == GPIO.HIGH:

print("Motion detected!")

else:

print("No motion detected.")

def read\_ultrasonic():

GPIO.output(trig\_pin, GPIO.LOW)

time.sleep(0.2)

GPIO.output(trig\_pin, GPIO.HIGH)

time.sleep(0.00001)

GPIO.output(trig\_pin, GPIO.LOW)

pulse\_start = time.time()

while GPIO.input(echo\_pin) == GPIO.LOW:

pulse\_start = time.time()

pulse\_end = time.time()

while GPIO.input(echo\_pin) == GPIO.HIGH:

pulse\_end = time.time()

pulse\_duration = pulse\_end - pulse\_start

distance = pulse\_duration \* 17150 # Speed of sound is 34300 cm/s

distance = round(distance, 2)

GPIO.output(ultrasonic\_led, distance < 50) # Turn Ultrasonic LED on if distance < 50cm

print(f"Distance: {distance} cm")

# Main loop

try:

while True:

read\_pir()

read\_ultrasonic()

time.sleep(1) # Delay for readability, adjust as needed

except KeyboardInterrupt:

GPIO.cleanup()

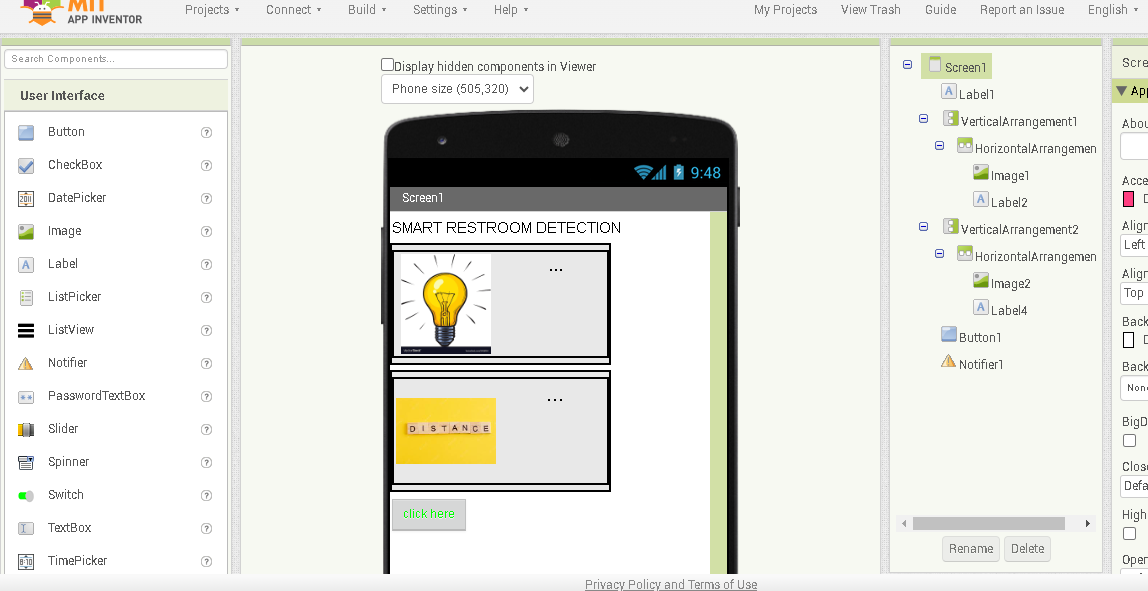
According to pin numbers in the program the sensors are connected and executed and simulated.

The smart public restroom is gives the real time data in the smart app. Which can be accessed by the android apk.

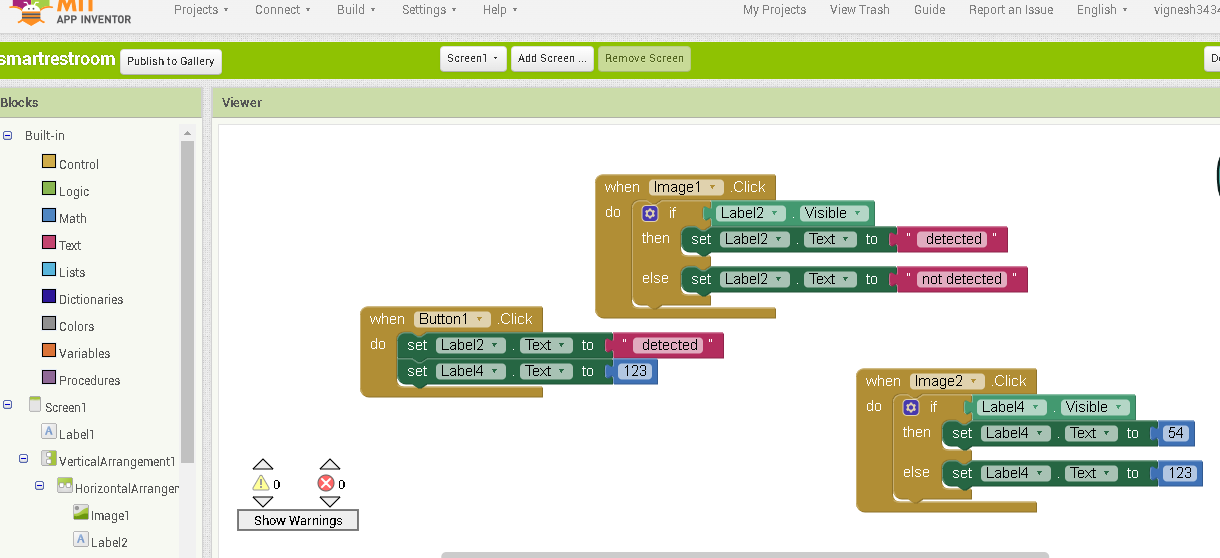
First the app is build in the MIT app inventor so make sure that the required things like (image, layout, button,mqqt file).

First set the screen size and make the horizontal and vertical layout of the insertion button and images.

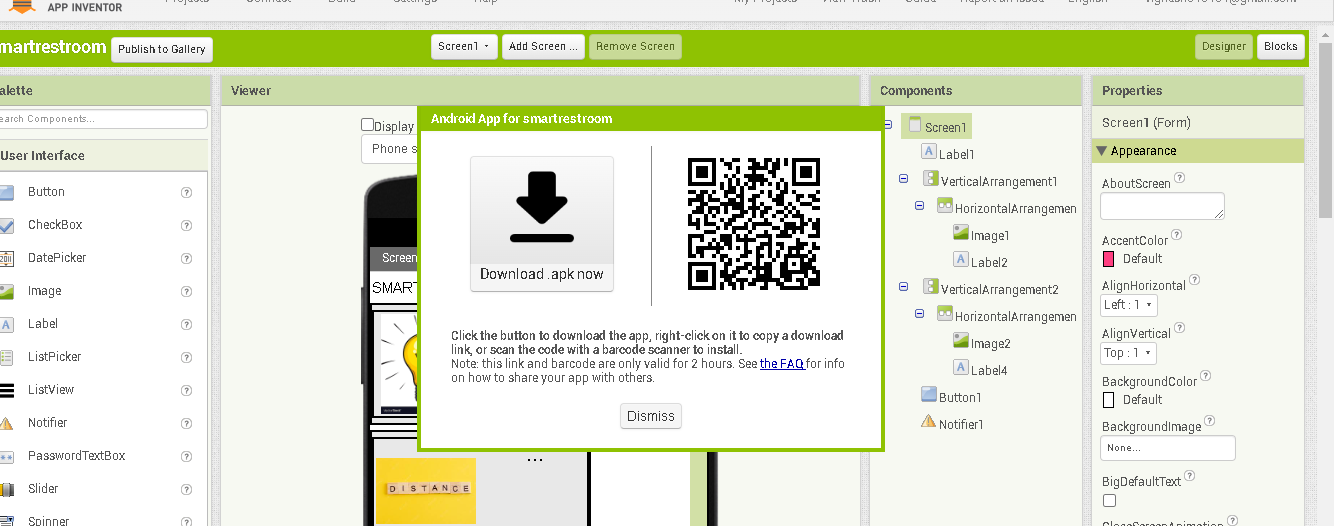
Then use the required image and layout text with required text used to click and show the data and represent the component using image.

The usage in my app is based on my simulation so the basic of two components pir sensor to detect motion and ultra sonic sensor to detect the distance.

So if we click the button we can know the data of the motion detection and distance in usage of the smart restroom. The designer app is then the layot and the processing input are given correctly in design.



The blocks are like coding we have to give the proper function in the blocks to identify the layout image insertion and buttons .we have to give the data for the first image and second image and click button what to do? And the give the information about the data in the app.



After completing the basic information details and getting the work done and download the mqqt extension download the zip file and extract the file and upload in the base apk.

Then build the app and the QR and the download dialog box will appear after the compiling the apk.

Download and the check the apk.it shows the real time usage in the smart restroom.

The QRis given in the above for apk.

The github link for the app is given below

<https://github.com/karthi290/IOT.git>

the final submission of the project is done here.