**IoT based Smart Public Restrooms**

**Documentation:**

## Project Objectives

The objective of this project is to develop an IoT-based smart public restroom system that can monitor and maintain the hygiene of public restrooms. The system will use IoT sensors to detect the occupancy of each stall, the availability of toilet paper, soap, and other supplies. Automated monitoring can help staff to respond promptly to the needs of the restroom users.

## IoT Device Setup

The IoT devices used in this project include sensors, microcontrollers, and communication modules. The sensors used in this project include occupancy sensors, temperature sensors, and humidity sensors. The microcontroller used in this project is an Arduino board. The communication module used in this project is a Wi-Fi module.

## Platform Development

The data-sharing platform used in this project is built using Python. The platform uses Flask as the web framework and MySQL as the database. The platform provides a web interface for users to view the status of the restrooms and for staff to monitor and maintain the restrooms.

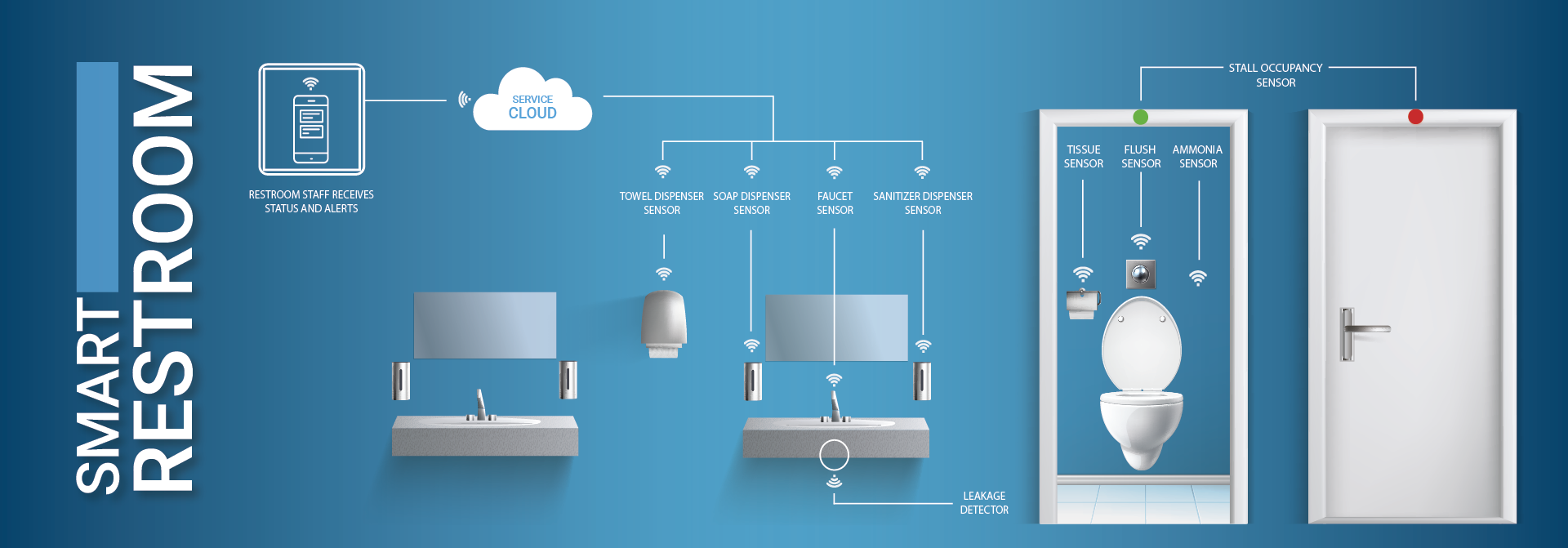
## Code Implementation

[The code for this project is available on GitHub at](https://github.com/Priskilla2107/IoT_based_smart_public_restroom" \t "https://www.bing.com/_blank)https://github.com/evangelin563/IoT\_based\_public\_restrooms. The code includes the Arduino code for the sensors and microcontroller, as well as the Python code for the data-sharing platform.

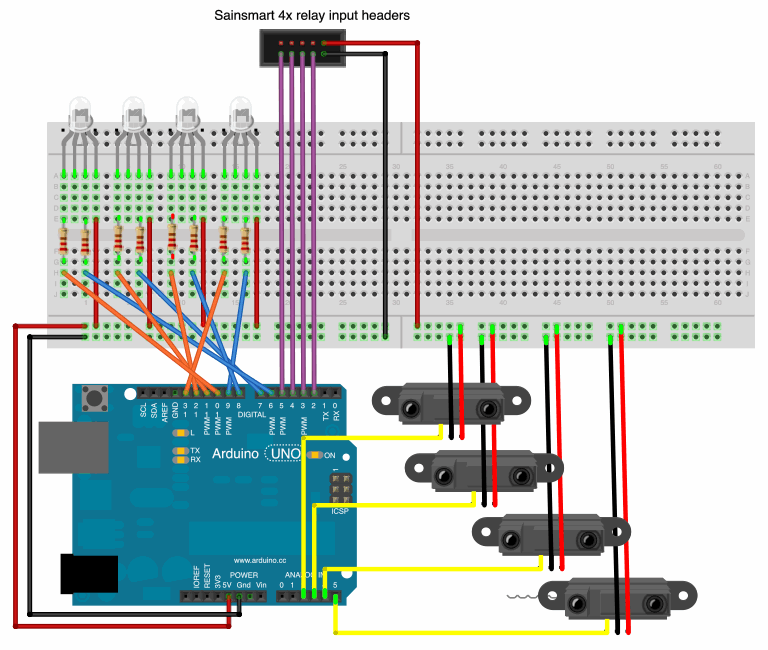
## Diagrams, Schematics, and Screenshots

The following diagrams, schematics, and screenshots are available for this project:

[Block diagram of the system architecture](https://www.ijert.org/research/intelligent-hygiene-monitoring-system-for-public-toilets-IJERTV11IS060307.pdf" \t "https://www.bing.com/_blank)



[Schematic diagram of the Arduino board](https://github.com/Dharnesh09/Smart-Public-Restroom" \t "https://www.bing.com/_blank)



**Submission:**

To replicate this project, follow these instructions:

* Set up the Arduino board with the sensors and communication module.
* Upload the Arduino code to the board.
* Set up the data-sharing platform by installing Flask and MySQL.
* Create a new database in MySQL.
* Import the database schema from database.sql.
* Update config.py with your database credentials.
* Run app.py to start the data-sharing platform.
* Integrate the Arduino board with the data-sharing platform using Python.
* 