ASIAN COLLEGE OF ENGINEERING AND TECHNOLOGY

SMART PUBLIC RESTROOM

TEAM MEMBER

ABINAYA.T

ARTHI.M

AJAY.V

BAKKIYA.G

AATHAM

AMSAVIRUTHAN

BUVANESHVARAN

LIST OF FIGURES

LIST OF GRAPH

ABSTRACT

In the cutting edge world, the advances are definitely grown, yet at the same time the cleanliness in our nation is under risk. The abstract of this paper is to deliver clean and hygiene toilets. All the public toilets should be clean and hygiene. In our country, our government has introduced the scheme called "Swachh Bharat" (Clean India). Keeping the toilets uncontaminated is the one of the objective of Clean India scheme. This paper can be helpful to encourage the clean India project. In future, it can show the major part in clean India scheme. In an Existing system, they are focused only on identifying the dirt in the toilets. In our proposed system, we have determined on keeping clean toilets, observing the sweeper's working activities. It can dodge many syndromes. It may create the consciousness amongst people about the toilet management. Therefore, our development is to use safe and hygienic toilets. This paper is based on IOT and image-processing concepts using different sensors like smell sensor, IR sensor, sonic sensor, RFID reader. By using these sensors, we can create the smart toilets.

CHAPTER 1

INTRODUCTION

IoT (Internet of Things) technology has enabled a wide range of applications in various domains, and one such domain is the management of public restrooms. The use of IoT technology in smart public restrooms can provide better service to users, optimize maintenance and cleaning schedules, and save water and energy. Here are some of the IoT applications in smart public restrooms: Automated monitoring: IoT sensors can be installed in public restrooms 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. Predictive maintenance: IoT sensors can IoT Applications in Smart Public Restrooms IoT (Internet of Things) technology has enabled a wide range of applications in various domains, and one such domain is the management of public restrooms. The use of IoT technology in smart public restrooms can provide better service to users, optimize maintenance and cleaning schedules, and save water and energy. Here are some of the IoT applications in smart public restrooms: Automated monitoring: IoT sensors can be installed in public restrooms 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. Predictive maintenance: IoT sensors can A smart washroom is a customizable solution that integrates operational and environmental smart washroom sensors and IoT technology, in commercial and retail washrooms, to provide a clear view of data in real-time.

Smart restroom sensors and mobile device integration make smart restroom technology enablement seamless and easy. The opportunity to improve restroom management is here. And the time to get smart about restrooms is now.Building leaders must be strategic about deploying their cleaning resources with a heightened awareness of building hygiene in today's post-pandemic era. Their efforts must be visible and reassure building occupants that their health and safety are at the forefront of the facility management team's efforts. Adding smart restroom technology that makes it easy for building managers to meet the public's expectations is not only suitable for occupants, but it's also good for business.

CHAPTER 2

SYSTEM REQUIREMENTS

2.1 HARDWARE REQUIREMENTS

Processor :IntelI3 6TH Genprocessor

RAM :4GB

Hard Disk :500GB

Input Devices :Keyboard and Mouse Data (DAU),central system unit(CSU),

Output Devices :Monitor.

2.2 SOFTWARE REQUIREMENTS

Operating system :Windows 11

Language :PYTHON,JSON

2.3 SOFTWARE DESCRIPTION

A. PYTHON:

Python is a general-purpose language which means it is versatile and can be used to program many different types of functions. Because it is an interpreted language, it precludes the need for compiling code before execution and because it is a high-level programming language, Python is able to abstract

details from code. In fact, Python focuses so much attention on abstraction that its code can be understood by most novice programmers.

Python code tends to be short and when compared to compiled languages like C and C++, it executes programs slower. Its user-friendliness makes it a popular language for citizen developers working with machine learning algorithms in low-code no-code (LCNC) software applications.

Python has a simply syntax and is known for having a large community that actively contributes to a growing selection of software modules and libraries. Python's initial development was spearheaded by Guido van Rossum in the late 1980s. Today, Python is managed by the Python Software Foundation.

Techopedia Explains Python

Python offers several frameworks for web development. A Python Web framework is a group of modules and libraries that enable programmers to re-use another developer's code. This collaborative approach can developers avoid dealing with low-level issues such as protocols, sockets and process/thread management.

Python Frameworks

Here are 10 frameworks that web developers, machine learning teams and data analytics teams should consider when using Python:

Open-source Django is a popular Python web framework that facilitates quick web design and development. Django is a free-to-use framework that enables developers to reuse code to build high-quality web apps and APIs. Django is known for:

- Helping programmers avoid security blunders.
- Supporting a data-driven architecture.
- Moving software from concept to launch quickly.

Pyramid is a compact open-source web framework that works in all supported versions of Python. It offers the essential elements required for online applications including delivering static content and converting URLs to code. Some of Pyramid's attributes include:

- Security APIs that support authentication and authorization.
- A cookiecutter that generates sample Pyramid projects from project templates.
- Supporting the SQLAlchemy project and using its object-relational mapper (ORM) to interface with databases.

Bottle is a Web Server Gateway Interface *(*WSGI) micro-web framework for Python that is known for being lightweight and easy to use. Bottle is distributed as a single file module and the default Python library is the only dependency of the framework. It is is a popular framework for building mobile applications and supports:

- Python versions 2.7 and above.
- Mako, Jinja2, and Cheetah templates.
- WSGI-capable HTTP servers, including Bjoern, Google App Engine, fapws3 and CherryPy.
- URL mapping using condensed syntax.

CherryPy is an object-oriented HTTP framework that supports Apache and Microsoft IIS. Some of CherryPy's attributes include:

- A robust configuration system suitable for both developers and deployers.
- Built-in support for testing, coverage and profiling.
- Tools for authentication and caching.
- Flexible plugins.
- Robust configuration management.

Flask offers more control than its closest competitor, Django, and features support for unit testing. Along with RESTful request-dispatching and WSGI compatibility, Flask is known for:

- Providing an integrated development server with a debugger.
- Jinja2 templating (tags, filters, macros, and more).
- 100% compliance with WSGI 1.0.

Web2py allows developers to create, distribute, debug, test, manage a database and maintain applications. It has no setup files and can operate from a USB disk. Web2py can:

- Serve as a manual for web developers using the Model View Controller (MVC) paradigm.
- Automatically fix problems that may result in security risks.
- Support a database abstraction layer (DAL) that dynamically writes SQL is part of the framework.

Tornado is an open-source asynchronous framework for I/O operations. Tornado is known for supporting applications that require long-lived connections, real-time location services and allowing the integration of authentication and authorization methods from third parties.

BlueBream is a web application framework, server and library for Python programmers that was initially known as Zope 3. BlueBream is known for being durable, reliable and adaptive. It supports reusable software components as well as:

- WSGI (Web Server Gateway Interface) compatibility for Python.
- A template-development language that complies with XHTML.
- A program for creating forms automatically.

Grok

Grok is a robust framework for creating dependable and adaptable web applications. It supports DRY (Don't Repeat Yourself) software development and has a quick learning curve. Like other full-stack Python web frameworks, Grok features an intuitive UI (user interface).

Quixote

Quixote allows Python programmers to quickly create Web-based apps. This framework's objective is to offer web developers exceptional performance and flexibility for producing HTML with Python code

b.JSON

JSON, or JavaScript Object Notation, is a format used to represent data. It was introduced in the early 2000s as part of JavaScript and gradually expanded to become the most common medium for describing and exchanging text-based data. Today, JSON is the universal standard of data exchange. It is found in every area of programming, including front-end and server-side development, systems, middleware, and databases.

This article introduces you to JSON. You'll get an overview of the technology, find out how it compares to similar standards like XML, YAML, and CSV, and see examples of JSON in a variety of programs and use cases.

TABLE OF CONTENTS

SHOW MORE

A little bit of history

JSON was initially developed as a format for communicating between JavaScript clients and back-end servers. It quickly gained popularity as a human-readable format that front-end programmers could use to communicate with the back end using a terse, standardized format. Developers also discovered that JSON was very flexible: you could add, remove, and update fields ad hoc. (That flexibility came at the cost of safety, which was later addressed with the JSON schema.)

[Why Wasm is the future of cloud computing | The rise of WebAssembly]

In a curious turn, JSON was popularized by the AJAX revolution. Strange, given the emphasis on XML, but it was JSON that made AJAX really shine. Using REST as the convention for APIs and JSON as the medium for exchange proved a potent combination for balancing simplicity, flexibility, and consistence.

Next, JSON spread from front-end JavaScript to client-server communication, and from there to system config files, back-end languages, and all the way to databases. JSON even helped spur the NoSQL movement that revolutionized data storage. It turned out that database administrators also enjoyed JSON's flexibility and ease of programming.

Today, document-oriented data stores like MongoDB provide an API that works with JSON-like data structures. In an interview in early 2022, MongoDB CTO Mark Porter noted that, from his perspective, JSON is still pushing the frontier of data. Not bad for a data format that started with a humble curly brace and a colon.

Why developers use JSON

No matter what type of program or use case they're working on, software developers need a way to describe and exchange data. This need is found in databases, business logic, user interfaces, and in all systems communication. There are many approaches to structuring data for exchange. The two broad camps are binary and text-based data. JSON is a text-based format, so it is readable by both people and machines.

JSON is a wildly successful way of formatting data for several reasons. First, it's native to JavaScript, and it's used inside of JavaScript programs as JSON literals. You can also use JSON with other programming languages, so it's useful for data exchange between heterogeneous systems. Finally, it is human readable. For a language data structure, JSON is an incredibly versatile tool. It is also fairly painless to use, especially when compared to other formats.

How JSON works

When you enter your username and password into a form on a web page, you are interacting with an object with two fields: username and password.

CHAPTER 3

SYSTEM DESIGN

3.1 ALGORITHM

MFB-FA: Modified Flashing Behavior-based Firefly Algorithm

Input : Size of the population

Output : Best optimal solution (bestsubset)

Begin

set params for Modified_FA(); initpopulation;

Det of light intensity_I();

Object(fun) = f(q), q = (q1, . . . , qd)T

Generate an initial chaotic population of fireflies

qi, i = 1, 2, . . . n

compute the ′I′ so that it is associated with f(q) while(T Max Iteration)

Define absorption coefficient β with chaos Gaussian Map

for m = 1 : i (i ff) for n = 1 : i (i ff) if Im > In

move ff towards n; end if

Vary attractiveness withdistance Dist via exp−βDist ^{2}

Evaluation of new solutions and updating of light intensity (LI)

end for end for

Firefly ranking and determination of best solution;

T = T + 1

end

MANN-AM (Modified Artificial Neural Network with Attention Mechanism)

Input : Input : dataset df, xtrain, xtest − independent variable, ytrain, ytesttarget variable Output : target ypreds with tract infection of UTX_DIAGONSIS(0, 1)

scale the datasets using // input layer

divide xtrainxtest and ytrain, ytest //input layer

W is calculated based on dist using //hidden layer

set value for parameter k

Estimating the distance between train and test Sorting dist in an ascending pattern

select the best kneigh

Repeating steps 2–4 until the algorithm is over W matrix is saved as result

Simulation involves using //hidden layer Saving results

Retrieving ANN with attention model

Setting values for number of input, output and hidden layers

Primary weighing of existing neurons in input, output and hidden layers Calculating the output (y) for each neuron in output layer

Updating ANN parameters

3.2 SYSTEM ARCHITECTURE

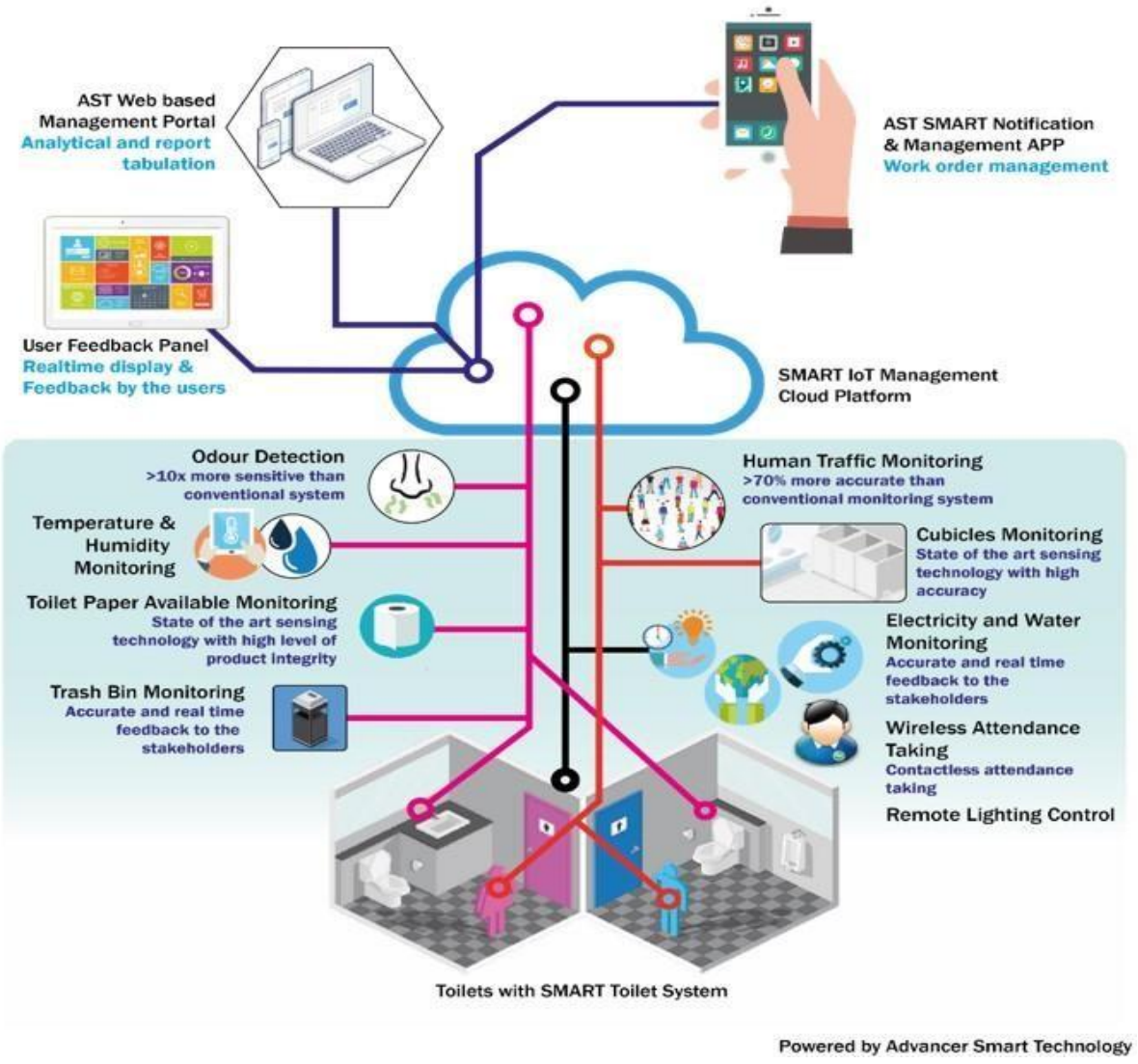


Fig 3.2.1 system architecture

3.3 UML DIAGRAM

3.3.1 USE CASE DIAGRAM

Fig 3.3.1 use case diagram

3.3.2 ACTIVITY DIAGRAM

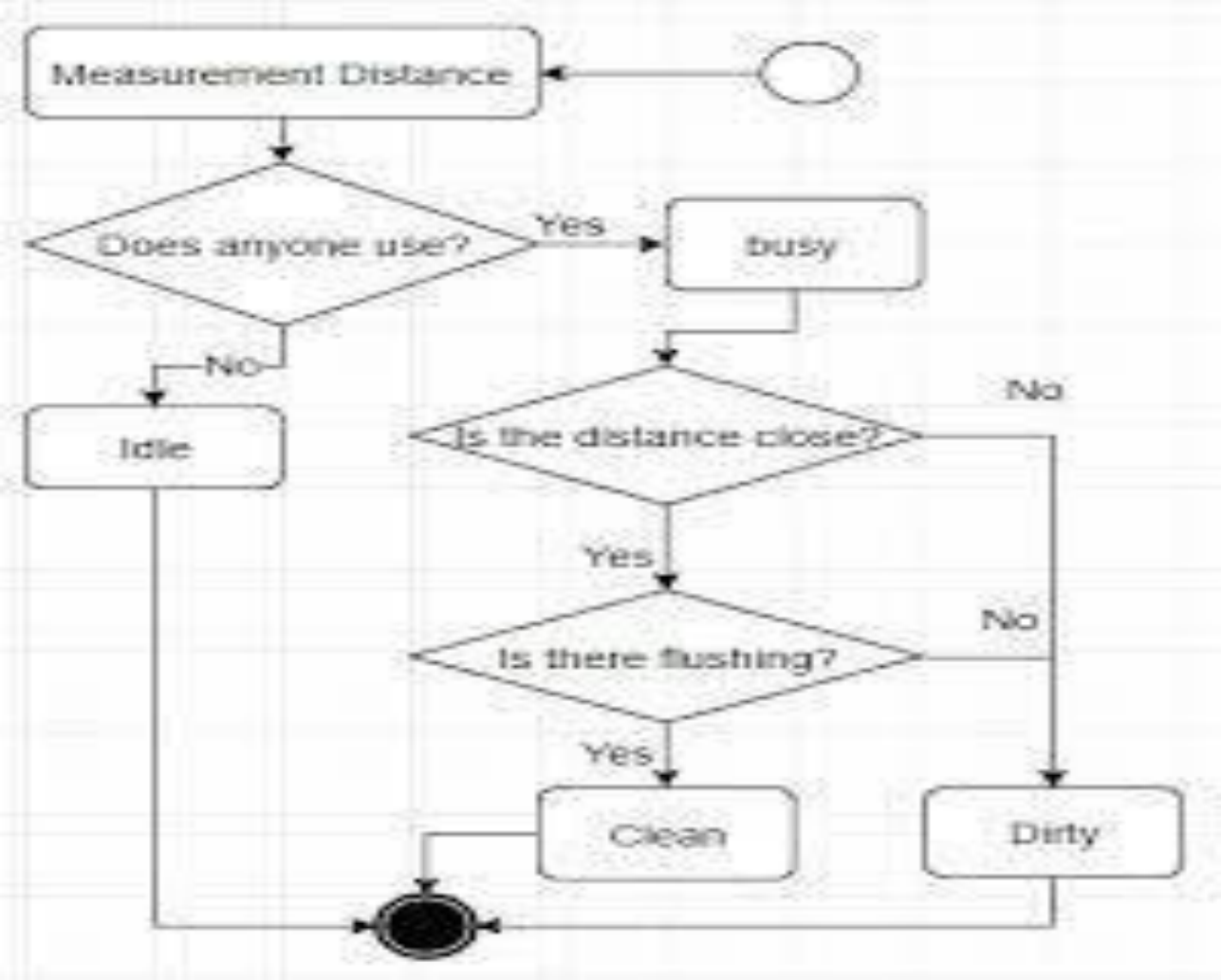
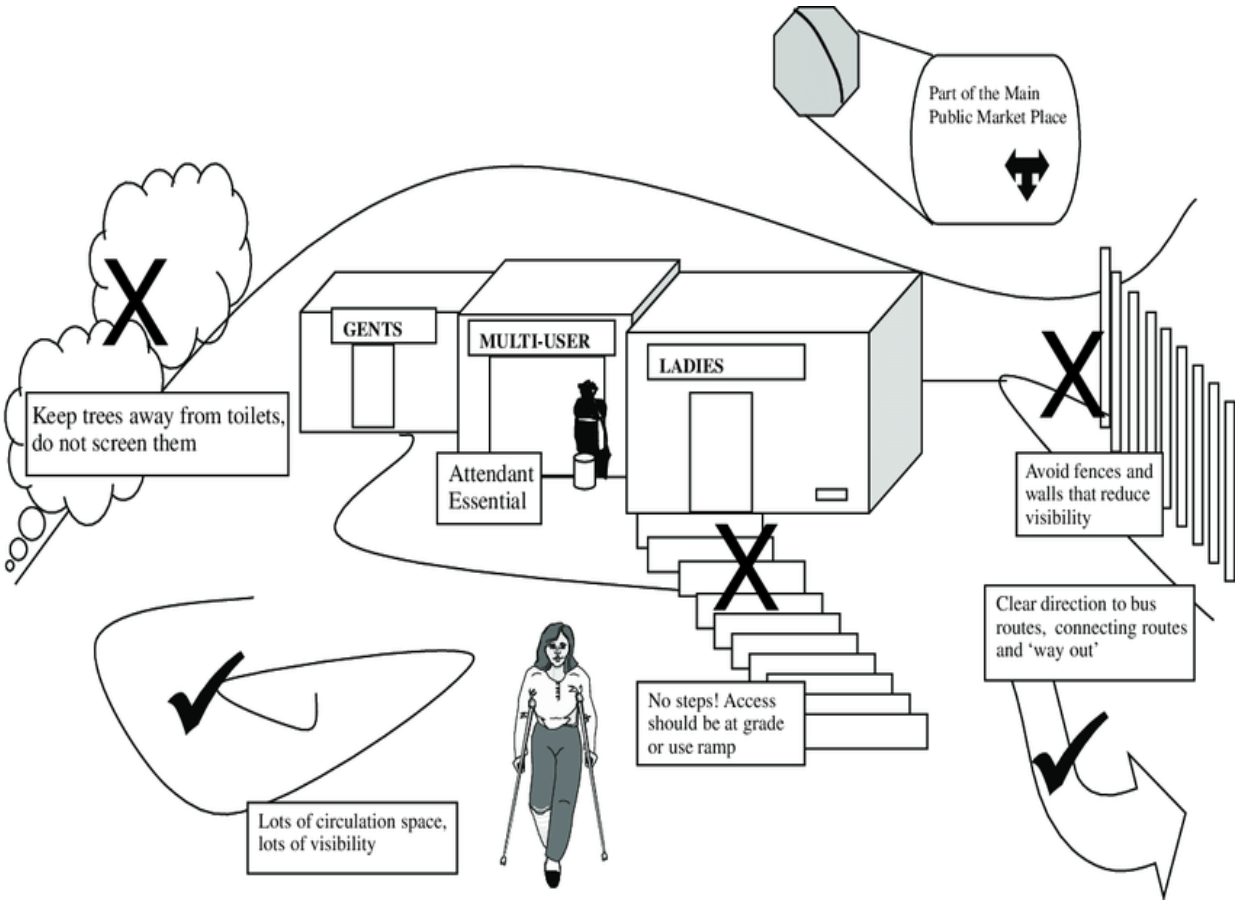


Fig3.3.2 activity diagram

3.3.3 CLASS DIAGRAM



Fif 3.3.3. class diagram

CHAPTER 4

SYSTEM ANALYSIS

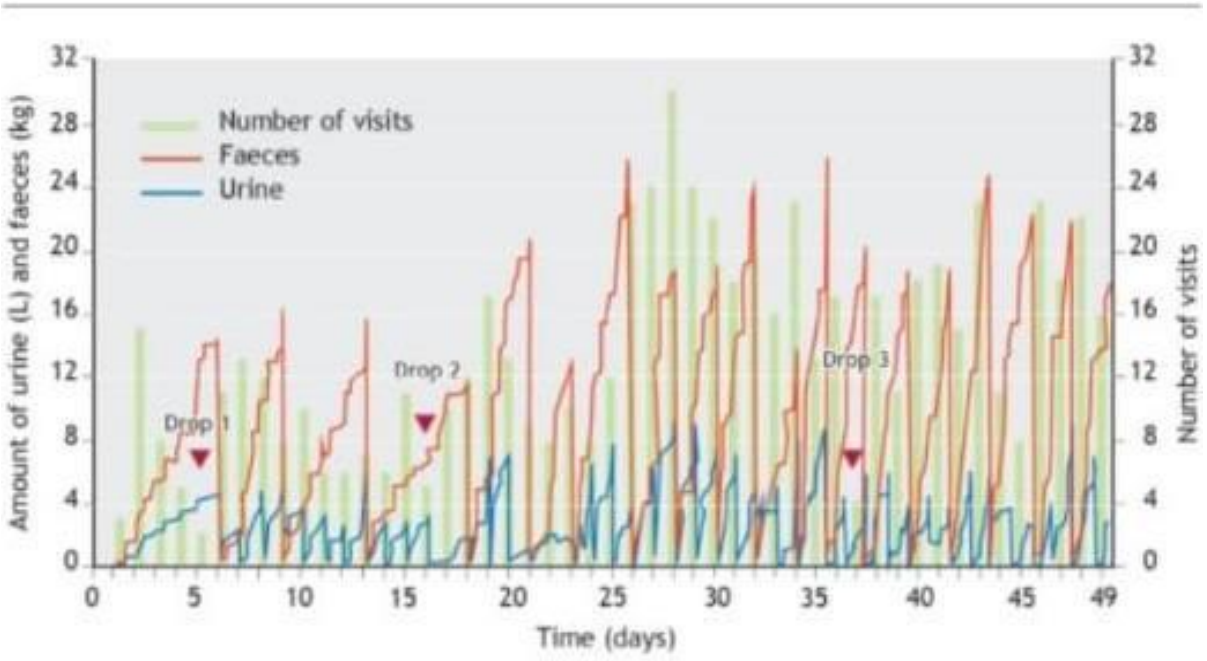


Fig 4.1 system analysis graph

The Advantages of Using Smart Toilets:

There is no secret that a smart toilet uses advanced technology in the sector of sanitation.But is that the only reason to use this piece of smart technology? The answer to it is that thereare more than just one benefit of smart toilets, and we shall look into those perks in this section.

A Smart Toilet Reduces Wastage of Water-

The planet is facing a crunch in its resources, water being one of them. Several parts of the world have no access to clean water, and people in these parts of the world have been in

this crisis for a very long time now. It is, therefore, necessary for people who have access to aplentitude of this resource to pull strains on their water usage and save as much as possible. We tend to use a lot of water in toilets and bathrooms. But smart toilets reduce the wastage ofwater to a great extent. Smart toilets have the ability to calculate the amount of water requiredin every flush and uses as less as 0.6 gallons of water for the task. That, compared to a regulartoilet, is quite a less quantity. Therefore, installing smart toilets in public places and living spaces makes for a brilliant idea.

Smart Toilets Have Several Environmental Benefits-

Smart toilets gave smart 'cleaning up' options that reduce the toxins put out into the environment to a large extent. Smart toilets make use of bidets or what we know as jet sprays (or hand faucets) for cleaning up, and this reduces the usage of toilet paper significantly. The amount of toxic waste, toilet papers and plastics dumped into the environment after every useof the restroom by just one individual, is horrifying. And that when multiplied by the total number of people inhabiting the planet, is a scary figure. A smart toilet makes sure that you donot have to use toilet paper to clean up, and even if you do, the quantity in which you do so will be quite less. This, thus, reduces wastage of resources and felling of trees to make the toiletpapers. Therefore, a smart toilet helps in maintaining the balance of nature by preserving the resources.

A Smart Toilet Has Hygienic Benefits-

The most important benefits that make smart toilets smart are the hygienic benefits that these toilers are loaded with. Many might think that using water to clean up is gross. But whenyou really think about it, using water is the best way to maintain basic sanitary hygiene. It reduces your risk of contracting an infection and therefore, keeps you healthy. Using toilet papers is neither hygienic nor is beneficial for the environment. Having smart toilets installed in public restrooms can help people who do not have access to primary and clean sanitation, live a healthy life and also help many aged and differently-abled individuals to be comfortablewhile using the restrooms. Plus, these toilets flush out the excreta with great swiftness and perfection, and are, therefore, the smartest way to practice healthy sanitation.

The Disadvantages of Smart Toilets

Cost: The higher cost of smart toilets can be attributed to various factors. The incorporation of advanced technologies, such as touchscreens, motion sensors, and water-saving mechanisms, requires additional engineering, research, and development costs. Moreover, the production of specialized components and the integration of complex systems contribute to theoverall price. Additionally, the limited supply and demand for smart toilets compared to traditional toilets can also impact the cost. While the initial investment may be higher, it is important to consider long-term savings that may result from water efficiency and energy- saving features.

1. Complexity: The complexity of smart toilets can manifest in multiple ways. Installation often involves more than just basic plumbing knowledge. It may require additional electrical wiring, water supply adjustments, and compatibility checks with existing infrastructure. Professional installation can ensure proper functioning and prevent potential issues. In terms of everyday use, the advanced features and controls of smart toilets can include a variety of settings, such as water temperature, pressure, nozzle position, and air-drying options. Users need to familiarize themselves with these settings, which may require referring to user manuals or spending time experimenting with different configurations to find their preferred settings.
2. Maintenance and Repairs: Maintenance and repairs for smart toilets can be more complex than those for traditional toilets. If any of the electronic components malfunction, such as the control panel, sensors, or motorized parts, it may require the expertise of a specialized technician or contacting the manufacturer for support. These components may not be readily available in local hardware stores, potentially leading to longer repair times and higher costs. Routine maintenance, such as cleaning the nozzles or checking for leaks, may also involve specific instructions and cleaning agents to ensure the longevity and performance of the smart toilet.
3. Reliance on Power and Connectivity: Smart toilets require a power source to operate their advanced features. They typically need to be connected to an electrical outlet to power the integrated systems, such as seat heating, bidet functions, or automatic flushing. In the event of a power outage, these features may become unavailable until power is restored. Some smart toilets may offer backup battery systems to mitigate this issue, but they may have limited capacity. Similarly, connectivity interruptions, such as Wi-Fi or Bluetooth signal loss, can disrupt certain features that rely on wireless communication, such as remote

control via smartphone apps or integration with smart home systems. Users should considerbackup power solutions or evaluate the impact of temporary unavailability of certain features when choosing a smart toilet.

4. Privacy and Security Concerns: The inclusion of sensors and connectivity capabilities in smart toilets can raise privacy concerns for users. Sensors are used to detect a user's presence, adjust settings, or activate certain functions automatically. While this can enhance convenience, some individuals may find it intrusive to have their toilet usage monitored or tracked. Additionally, if the smart toilet connects to the internet or a home network, there is the potential for data breaches or unauthorized access to the system. To address these concerns, users should carefully review the privacy policies of smart toilet manufacturers, ensure secure network connections, and consider models that prioritize privacy features, such as data encryption or user anonymity.
5. Compatibility and Interoperability: The smart toilet market is still evolving, resulting in a lack of standardized protocols and technologies across different manufacturers. This lack of compatibility can lead to challenges when attempting to integrate a smart toilet with other devices or systems in a smart home environment. For example, if a user wants to connect their smart toilet to a centralized home automation system, they may encounter difficulties if the toilet uses a different communication protocol or does not have compatibleinterfaces. Compatibility issues may limit the extent to which users can control or monitor their smart toilet alongside other smart devices in their home.

CHAPTER 5

SYSTEM IMPLEMENTATION

5.1 PROJECT DESCRIPTION

- Say goodbye to unsanitary restrooms! Smart toilets utilize **self-cleaning technology** and **touchless operation** tominimize contact with germs. With **integrated UV sterilization** and **antibacterial surfaces**, these toilets ensure a safer and healthier restroom experience for everyone. Smart toilets are designed with **universal accessibility** in mind. Featuring **adjustable heights**, **handrail support**, and**braille instructions**, these facilities cater to individuals with diverse needs. By promoting inclusivity, smart toilets are transforming public restrooms into welcoming spaces for everyone

- Smart toilets are designed with **universal accessibility** in mind. Featuring **adjustable heights**, **handrail support**, and**braille instructions**, these facilities cater toindividuals with diverse needs. By promoting inclusivity, smart toilets are transforming public restrooms into welcoming spaces for everyone.

5.2 PROJECT IMPLEMENTATION

Fig:5.2.1 Transparent public restroom.

Modern public restrooms are often cramped, dirty, and smelly. They are also usually located in the middle of a busy street or in a crowded mall. This makes it difficult for people to use them without feeling uncomfortable. Smart toilet restroom technology aims to improve the design and overall user experience of public restrooms. Making them more comfortable and convenient for everyone. One way to improve the design of smart public restrooms is by making them more user-friendly. For example, some restrooms now have touch screen-controlled toilets that allow users to select various settings such as water pressure and temperature. Other features that can make public restrooms more user-friendly include automatic flushing toilets, hands-free soap dispensers, and paper towel dispensers.

Another way to improve the design of public restrooms is by making them clean and sanitary. This can be done by installing automatic flush toilets, hands-free soap dispensers, and paper towel dispensers. Automatic flush toilets help to keep the Clean Toilet Bowl and sanitary by flushing it after each use. Hands-free soap dispensers help to prevent the spread of germs by allowing users to wash their hands without having to touch a dirty soap dispenser. Paper

towel dispensers help to keep the restroom clean by allowing users to dry their hands without

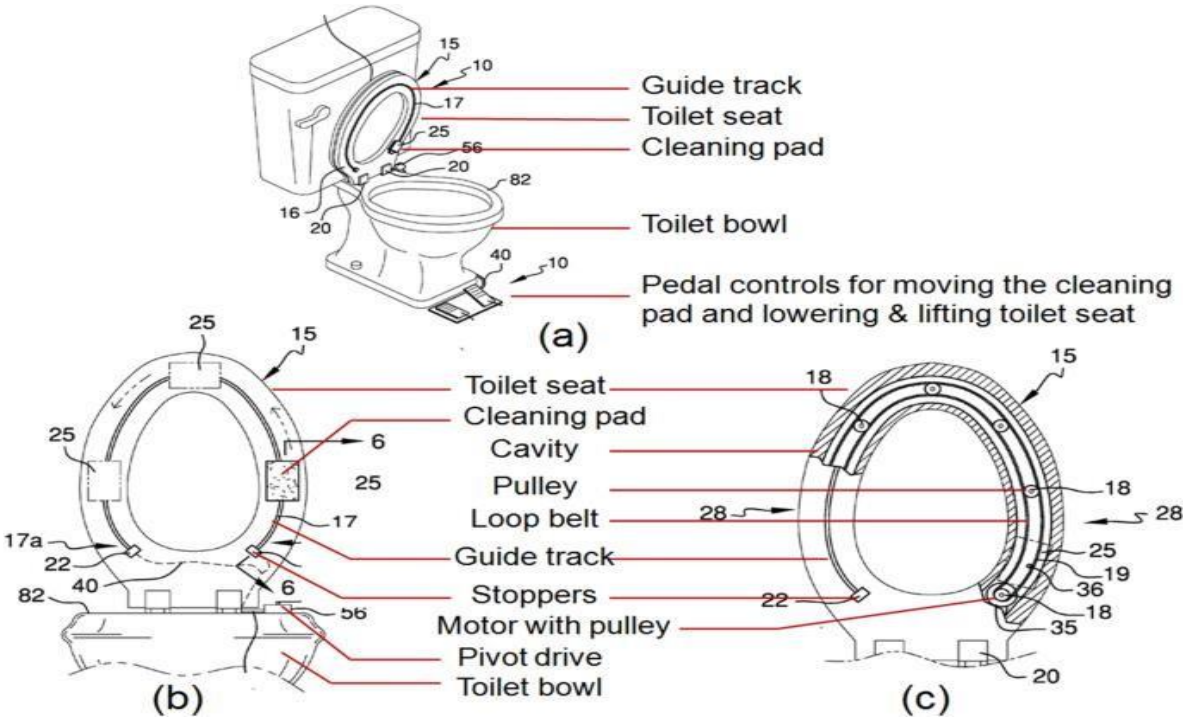


Fig:5.2.2 Smart punlic restroom

CHAPTER 6

SOURCE CODE

6.1 FRONT END CODE

```
#include<ESP32Servo.h>

#define TRIGGERPIN 32
#define ECHOPIN    35
#define RED_LED    33
#define GREEN_LED  25

Servo servo_1;

long duration;
int pos, distance, i=0;

void setup()
{
  servo_1.attach(18);
  Serial.begin(115200);
  pinMode(TRIGGERPIN, OUTPUT);
  pinMode(ECHOPIN, INPUT);
  pinMode(RED_LED, OUTPUT);
  pinMode(GREEN_LED, OUTPUT);

  Serial.println(" ");
  Serial.println("Sensing the Height");
  digitalWrite(RED_LED, HIGH);
  digitalWrite(GREEN_LED, LOW);
```

```
  pos = 0;
  servo_1.write(pos);
}

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 >= 100 && distance <= 150)
    {
      i = 1;
      if (pos != 180)
      {
        servo_1.write(180);
        pos = 180;
        i = 1;
      }
    }

  // for Child
    else if (distance >= 200 && distance <= 250)
      {
```

```
        i = 1;
        if (pos != 0)
        {
         servo_1.write(0);
         pos = 0;
         i = 1;
        }
       }

    else if (distance > 300 && i == 1)
      {
        digitalWrite(RED_LED, LOW);
        digitalWrite(GREEN_LED, HIGH);
        delay(5000);
        digitalWrite(RED_LED, HIGH);
        digitalWrite(GREEN_LED, LOW);
        i = 0;
       }

       delay (500);

  Serial.println(" ");
  Serial.print("Free Level : ");
  Serial.print(distance);
  Serial.print(" ");
  Serial.print("Position : ");
  Serial.print(pos);
  delay (500);
}
```

6.2 BACK END CODE

```
            {
  "version": 1,
```

```
"author": "Budhaditya Biswas",
"editor": "wokwi",
"parts": [
  { "type": "wokwi-esp32-devkit-v1", "id": "esp", "top": 57.34, "left": 242, "attrs": {} },
  {
    "type": "wokwi-hc-sr04",
    "id": "ultrasonic1",
    "top": -69.03,
    "left": 9.5,
    "attrs": { "distance": "139" }
  },
  { "type": "wokwi-servo", "id": "servo1", "top": 84.81, "left": -96.4, "attrs": {} },
  {
    "type": "wokwi-led",
    "id": "led1",
    "top": -44.13,
    "left": -86.2,
    "attrs": { "color": "red" }
  },
  {
    "type": "wokwi-led",
    "id": "led2",
    "top": -48.8,
    "left": 227.13,
    "attrs": { "color": "limegreen", "flip": "1" }
  }
],
"connections": [
  [ "esp:TX0", "$serialMonitor:RX", "", [] ],
  [ "esp:RX0", "$serialMonitor:TX", "", [] ],
  [ "servo1:GND", "esp:GND.2", "black", [ "h-44.1", "v112.72", "h280.67", "v-35.33" ] ],
  [ "esp:VIN", "servo1:V+", "red", [ "h-100.83", "v41.03", "h-297.33", "v-0.67" ] ],
  [ "servo1:PWM", "esp:D18", "green", [ "h-31.43", "v133.06", "h520.67", "v-2.67" ] ],
  [ "ultrasonic1:GND", "esp:GND.2", "black", [ "v183", "h8.03" ] ],
```

```
    [ "ultrasonic1:VCC", "esp:VIN", "red", [ "v191", "h166.2" ] ],
    [ "ultrasonic1:TRIG", "esp:D32", "green", [ "v0" ] ],
    [ "ultrasonic1:ECHO", "esp:D35", "green", [ "v0" ] ],
    [ "led1:A", "ultrasonic1:VCC", "red", [ "v40.33", "h142" ] ],
    [ "led2:A", "ultrasonic1:VCC", "red", [ "v46.33", "h-161.33" ] ],
    [ "led1:C", "esp:D33", "green", [ "v56.33", "h225.37", "v94" ] ],
    [ "led2:C", "esp:D25", "green", [ "v56.33", "h-87.3", "v102" ] ]
  ],
  "dependencies": {}
}
```

6.3 OUT PUT

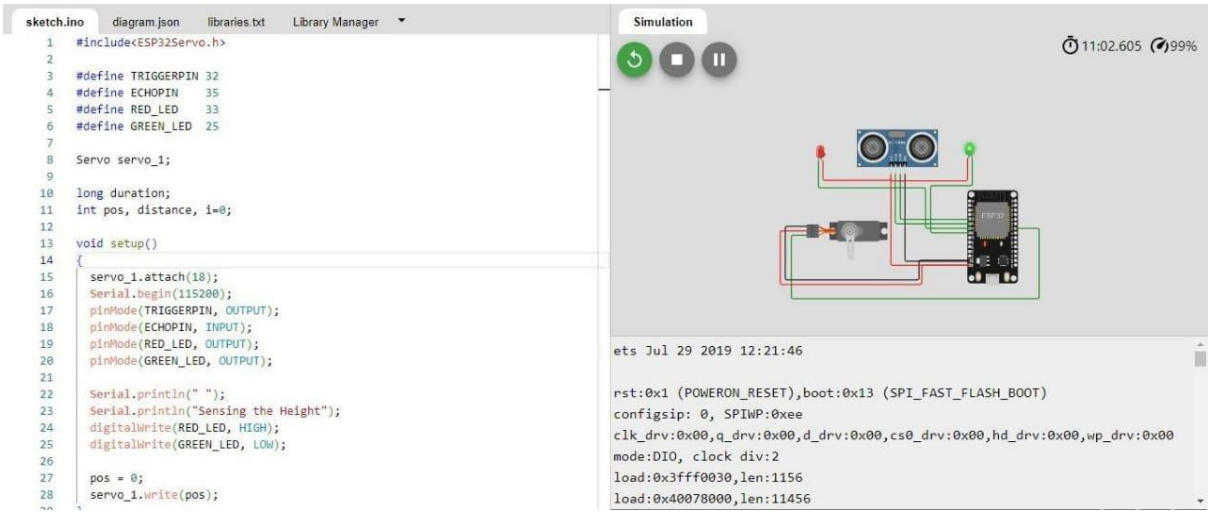


Fig:6.1

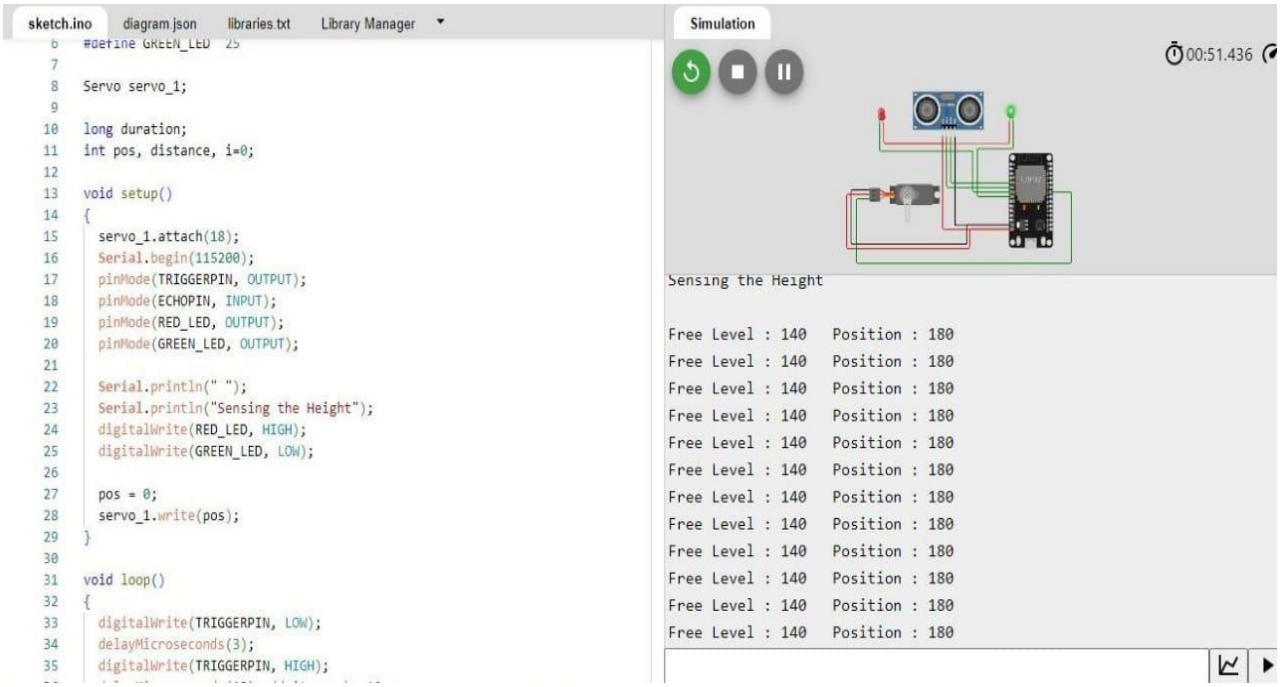


Fig:6.2

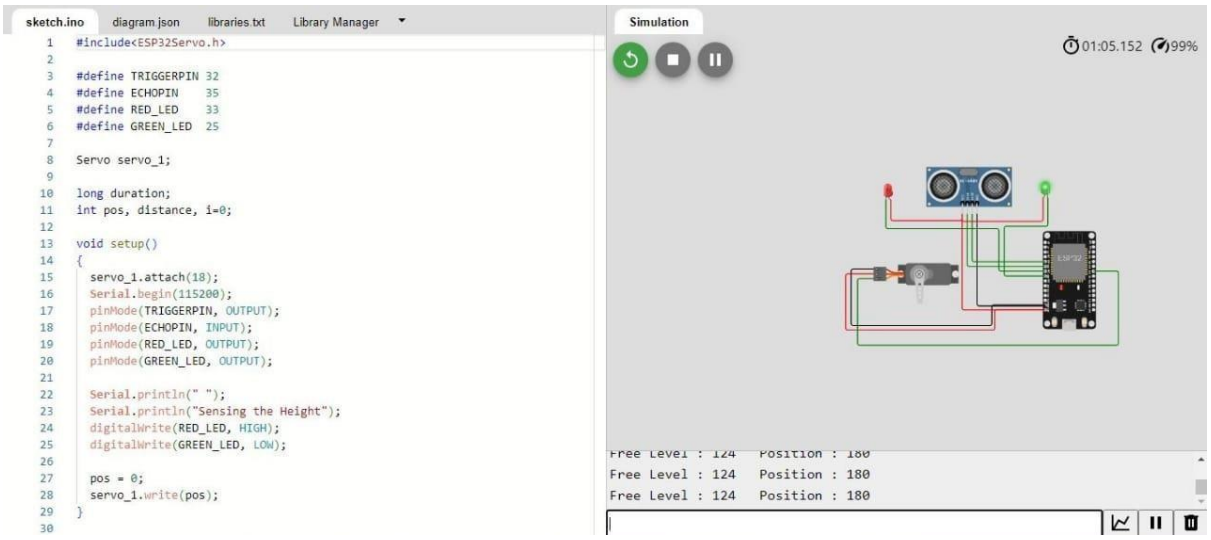

sketch.ino
diagram.json
libraries.txt
Library Manager
Simulation
01:05.152
99%
Free Level : 124 Position : 180
Free Level : 124 Position : 180
Free Level : 124 Position : 180

Fig:6.3

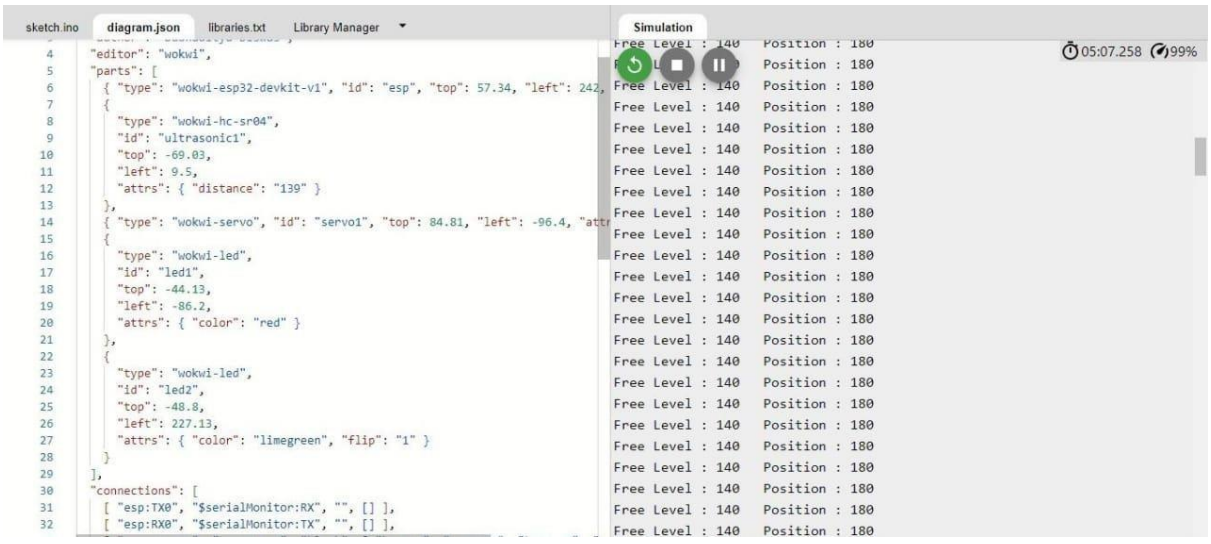

sketch.ino
diagram.json
libraries.txt
Library Manager
Simulation
05:07.258
99%
"editor": "wokwi",
"parts": [
{ "type": "wokwi-esp32-devkit-v1", "id": "esp", "top": 57.34, "left": 242,
{
"type": "wokwi-hc-sr04",
"id": "ultrasonic1",
"top": -69.03,
"left": 9.5,
"attrs": { "distance": "139" }
},
{ "type": "wokwi-servo", "id": "servo1", "top": 84.81, "left": -96.4,
{
"type": "wokwi-led",
"id": "led1",
"top": -44.13,
"left": -86.2,
"attrs": { "color": "red" }
},
{
"type": "wokwi-led",
"id": "led2",
"top": -48.8,
"left": 227.13,
"attrs": { "color": "limegreen", "flip": "1" }
}
],
"connections": [
["esp:TX0", "$serialMonitor:RX", "", []],
["esp:RX0", "$serialMonitor:TX", "", []],
Free Level : 140 Position : 180

Fig:6.4

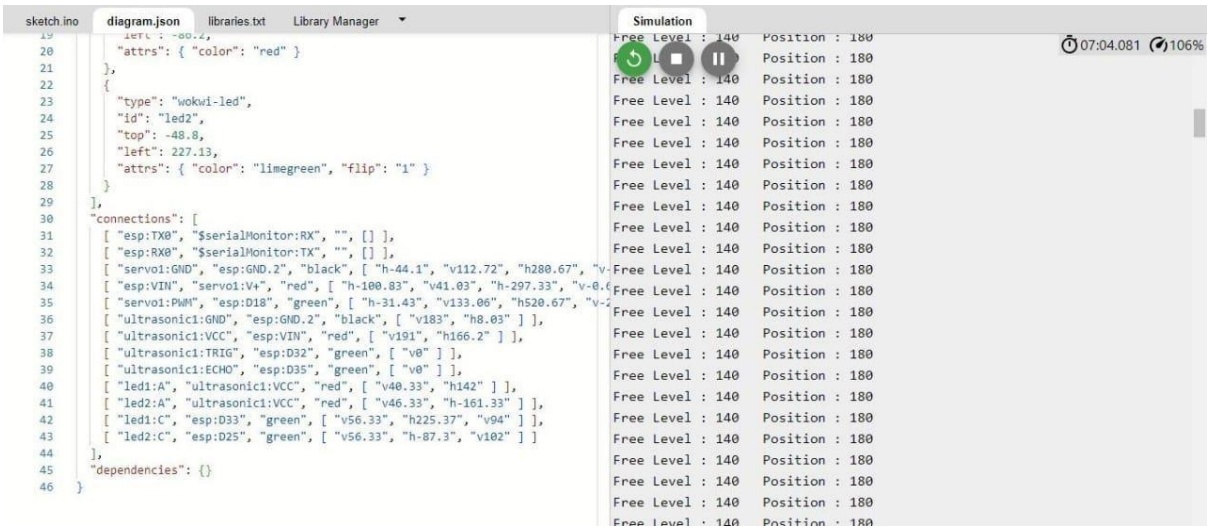

sketch.ino
diagram.json
libraries.txt
Library Manager
Simulation
07:04.081
106%
"attrs": { "color": "red" }
"type": "wokwi-led",
"id": "led2",
"top": -48.8,
"left": 227.13,
"attrs": { "color": "limegreen", "flip": "1" }
"connections": [
["esp:TX0", "$serialMonitor:RX", "", []],
["esp:RX0", "$serialMonitor:TX", "", []],
["servo1:GND", "esp:GND.2", "black", ["h-44.1", "v112.72", "h280.67",
["esp:VIN", "servo1:V+", "red", ["h-100.83", "v41.03", "h-297.33",
["servo1:PWM", "esp:D18", "green", ["h-31.43", "v133.06", "h520.67",
["ultrasonic1:GND", "esp:GND.2", "black", ["v183", "h8.03"]],
["ultrasonic1:VCC", "esp:VIN", "red", ["v191", "h166.2"]],
["ultrasonic1:TRIG", "esp:D32", "green", ["v0"]],
["ultrasonic1:ECHO", "esp:D35", "green", ["v0"]],
["led1:A", "ultrasonic1:VCC", "red", ["v40.33", "h142"]],
["led2:A", "ultrasonic1:VCC", "red", ["v46.33", "h-161.33"]],
["led1:C", "esp:D33", "green", ["v56.33", "h225.37", "v94"]],
["led2:C", "esp:D25", "green", ["v56.33", "h-87.3", "v102"]]
"dependencies": {}
Free Level : 140 Position : 180

Fig:6.5

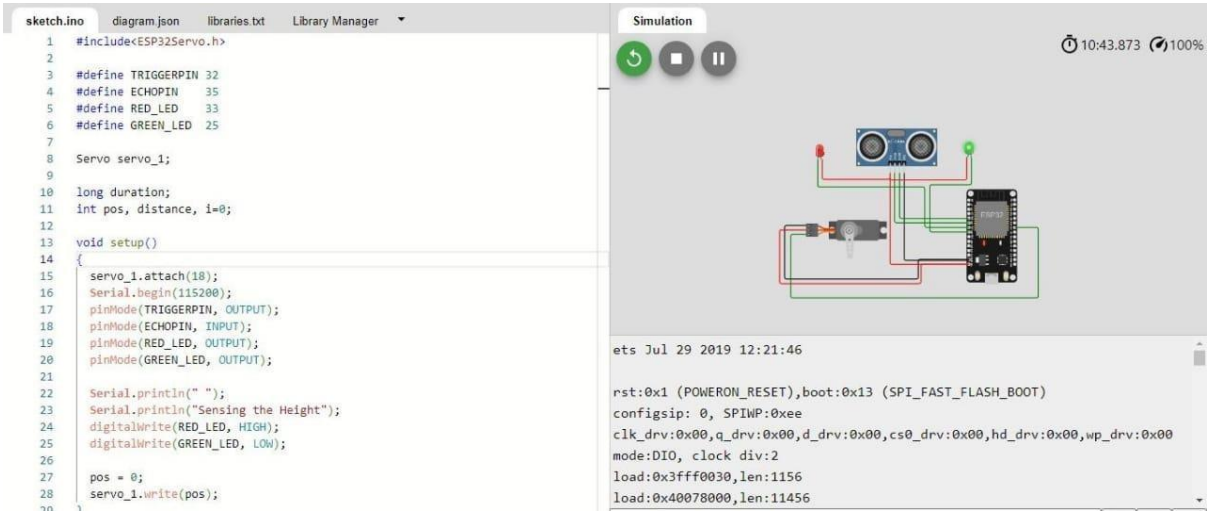

sketch.ino
diagram.json
libraries.txt
Library Manager
Simulation
10:43.873
100%
#include<ESP32Servo.h>
#define TRIGGERPIN 32
#define ECHOPIN 35
#define RED_LED 33
#define GREEN_LED 25
Servo servo_1;
long duration;
int pos, distance, i=0;
void setup()
{
servo_1.attach(18);
Serial.begin(115200);
pinMode(TRIGGERPIN, OUTPUT);
pinMode(ECHOPIN, INPUT);
pinMode(RED_LED, OUTPUT);
pinMode(GREEN_LED, OUTPUT);
Serial.println(" ");
Serial.println("Sensing the Height");
digitalWrite(RED_LED, HIGH);
digitalWrite(GREEN_LED, LOW);
pos = 0;
servo_1.write(pos);
ets Jul 29 2019 12:21:46
rst:0x1 (POWERON_RESET),boot:0x13 (SPI_FAST_FLASH_BOOT)
configsip: 0, SPIWP:0xee
clk_drv:0x00,q_drv:0x00,d_drv:0x00,cs0_drv:0x00,hd_drv:0x00,wp_drv:0x00
mode:DIO, clock div:2
load:0x3fff0030,len:1156
load:0x40078000,len:11456

Fig:6.6


sketch.ino
diagram.json
libraries.txt
Library Manager
#include<ESP32Servo.h>
#define TRIGGERPIN 32
#define ECHOPIN 35
#define RED_LED 33
#define GREEN_LED 25
Servo servo_1;
long duration;
int pos, distance, i=0;
void setup()
{
servo_1.attach(18);
Serial.begin(115200);
pinMode(TRIGGERPIN, OUTPUT);
pinMode(ECHOPIN, INPUT);
pinMode(RED_LED, OUTPUT);
pinMode(GREEN_LED, OUTPUT);
Serial.println(" ");
Serial.println("Sensing the Height");
digitalWrite(RED_LED, HIGH);
digitalWrite(GREEN_LED, LOW);
pos = 0;
servo_1.write(pos);
}
Simulation
11:02.605
99%
ets Jul 29 2019 12:21:46
rst:0x1 (POWERON_RESET),boot:0x13 (SPI_FAST_FLASH_BOOT)
configsip: 0, SPIWP:0xee
clk_drv:0x00,q_drv:0x00,d_drv:0x00,cs0_drv:0x00,hd_drv:0x00,wp_drv:0x00
mode:DIO, clock div:2
load:0x3fff0030,len:1156
load:0x40078000,len:11456

Fig:6.7

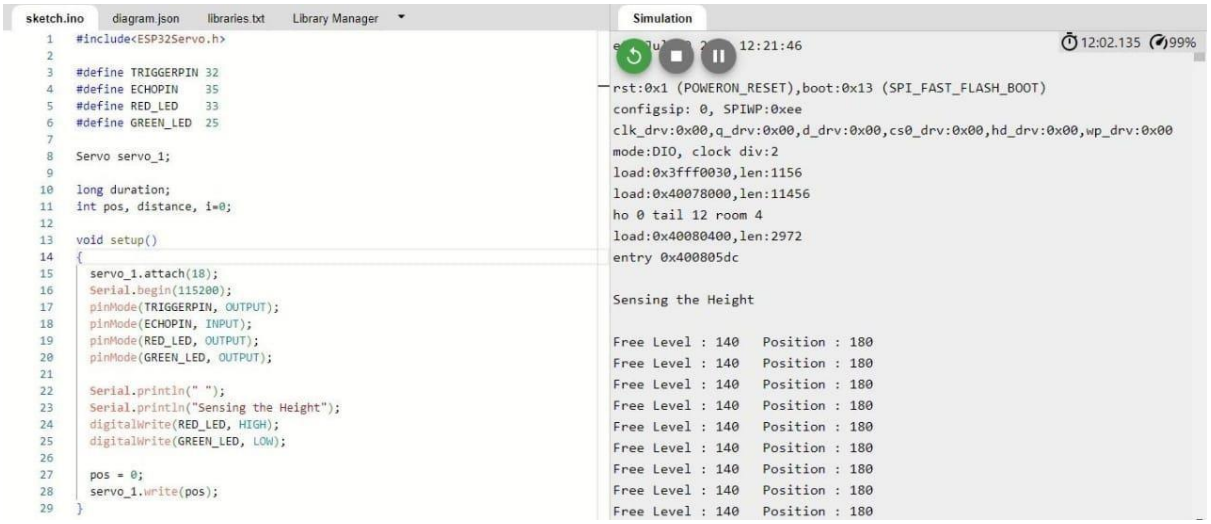

sketch.ino
diagram.json
libraries.txt
Library Manager
#include<ESP32Servo.h>
#define TRIGGERPIN 32
#define ECHOPIN 35
#define RED_LED 33
#define GREEN_LED 25
Servo servo_1;
long duration;
int pos, distance, i=0;
void setup()
{
servo_1.attach(18);
Serial.begin(115200);
pinMode(TRIGGERPIN, OUTPUT);
pinMode(ECHOPIN, INPUT);
pinMode(RED_LED, OUTPUT);
pinMode(GREEN_LED, OUTPUT);
Serial.println(" ");
Serial.println("Sensing the Height");
digitalWrite(RED_LED, HIGH);
digitalWrite(GREEN_LED, LOW);
pos = 0;
servo_1.write(pos);
}
Simulation
12:02.135
99%
12:21:46
rst:0x1 (POWERON_RESET),boot:0x13 (SPI_FAST_FLASH_BOOT)
configsip: 0, SPIWP:0xee
clk_drv:0x00,q_drv:0x00,d_drv:0x00,cs0_drv:0x00,hd_drv:0x00,wp_drv:0x00
mode:DIO, clock div:2
load:0x3fff0030,len:1156
load:0x40078000,len:11456
ho 0 tail 12 room 4
load:0x40080400,len:2972
entry 0x400805dc
Sensing the Height
Free Level : 140 Position : 180
Free Level : 140 Position : 180
Free Level : 140 Position : 180
Free Level : 140 Position : 180
Free Level : 140 Position : 180
Free Level : 140 Position : 180
Free Level : 140 Position : 180
Free Level : 140 Position : 180
Free Level : 140 Position : 180

Fig:6.8

CHAPTER 7

CONCLUSION AND FUTURE ENHANCEMENT

Smart restroom technology is transforming the way we experience public and private restrooms. With features like occupancy sensors, odor detection, and supply level monitoring, smart restrooms enhance hygiene, efficiency, and user satisfaction. The implementation of smart restroom management systems and monitoring systems streamlines operations, optimizes resource allocation, and ensures a safe and pleasant restroom experience.

As technology continues to evolve, the future of smart restrooms looks promising, with AI-powered systems, voice-activated controls, and blockchain-based solutions on the horizon. By embracing these advancements, facilities can create smart restrooms that meet the ever-changing needs and expectations of users.

Investing in smart restroom solutions is not only a step towards operational excellence but also a testament to a commitment to hygiene, sustainability, and user satisfaction. With the continuous development of innovative technologies, smart restrooms are set to revolutionize the way we think about and experience public and private restrooms.

FUTURE ENHANCEMENT

In recent years, smart toilets have become a buzzword in the technology industry. The adoption of smart toilets is not only for the sake of luxury but also stems from the necessity to promote hygiene and health. Smart toilets have advanced features that transform the way we use the bathroom and our daily routine.

One of the most significant advantages of smart toilets is their ability to detect and diagnose medical conditions. These toilets have sensors that measure key diagnostic parameters like blood sugar levels, kidney function, and heart rate. This data can be sent to a doctor who can analyze and make a suitable diagnosis. With the increasing frequency of chronic diseases, smart toilets have become a crucial tool for early detection and preventive healthcare.

Smart toilets also improve personal hygiene in the bathroom. They include elements like bidet-style cleaning systems that are more efficient in cleaning compared to regular toilet paper. They also have automatic flushing mechanisms, and some models feature odor control systems. Moreover, smart toilets have heated seats, lighting, and music players to make the bathroom experience even more soothing and enjoyable.

Another feature of smart toilets is reducing unnecessary water consumption. Traditional toilets use around 1.6 gallons of water per flush, while smart toilets only use one gallon of water. This saves water and also promotes wastewater management, making them a more eco-friendly option.

The future direction of smart toilets includes integrating more functions, such as artificial intelligence, which can analyze a user's data to customize their toilet experience, and virtual reality systems, which can provide an immersive experience while using the bathroom.

In conclusion, smart toilets have revolutionized our daily habits by promoting hygiene, health, and a sustainable ecosystem. Their advanced features have made using the bathroom experience more enjoyable and functional. As technology evolves, we can expect to see smart toilets become a staple in all homes.

Xiamen Wing Technology Co.,ltd has been specialized in R & D, manufacturing and marketing the hygienic toilet seat since 2007. Its patented product- NAVISANI automatic toilet seat has been exported to more than 80 countries which covers all the continents in the world. Its trademark" NAVISANI" has been registered and protected in 54 countries and regions. Xiamen Wing Technology Co.,Ltd has been active in promoting its brand by attending world famous industry-related exhibitions,such as KBC, ISSA Interclean, ISH Frankfurt,KBIS,Interclean Amsterdam... Its hygienic toilet seat is widely installed in the public toilets in commercial facilities like office building, mall, hotel, airport, bank, health care center,clinic,club,bar,salon,gym,cinema,casino,school... This product has been generally accepted to be a good solution to offering public washrooms with more hygienic toilet condition. If you have any business contacts with sanitary wares, office euipment, cleaning product, hotel product and airport equipment, can recommend them this hygienic toilet seat, and you need more information, please contact +86 15359860976 or send email to wtc@china-wing.com.

CHAPTER 8

REFERENCES

1. Xavier Gibert, Vishal M Patel, and Rama Chellappa, in their IEEE paper titled as “Deep Multi-Task Learning for Railway Track Inspection” Volume 18, Issue 1, Jan 2017, pp 153 – 167.

2. S Mohamed Ashiq, K Karthikeyan, S Karthikeyan. “Fabrication of SemiAutomated Pressurized Flushing System in Indian Railway Toilet”, International Journal of Engineering and Advanced Technology (IJEAT), Volume-2, Issue-3, February2013.

3. Dr. ManojHedaoo, Dr. SuchitaHirde ,Ms. Arshi Khan “Sanitation In Indian Railway Premises: A Great Cause Of Concern”, International Journal of Advanced Engineering Technology, Mar 2012, Volume 3, Issue 1, pp 50 -55.

4. Dhanajay G Dange, Dattaprakash G Vernekar, Sagar D Kurhade, Prashant D Agwane, “Methodology for Design and Fabrication of Human Waste Disposal System for Indian Railway”, International Journal of Science Technology & Engineering, Volume 2, Issue 07, January2016, pp 14 – 19.

5. Mesch, F., Puente Le´on, F. &Engelberg, T., Train-based location by detecting rail switches. Computers in Railways VII, eds. J. Allen, R.J. Hill, C.A. Brebbia, G. Sciutto & S. Sone, WIT Press, Southampton, pp. 1251– 1260

6. K. Osathanunkul, K. Hantarkul, P. Pramokchon, P. Khoenkaw and N. Tantitharanukul, “Design and Implementation of an Automatic Smart Urinal Flusher”,International Computer Science and Engineering Conference (ICSEC2016), Chiang Mai, Thailand, Dec,2016, pp 14-17.

7. J. Shah and B. Mishra, “IoT enabled Environmental Monitoring System for Smart Cities”, International Conference on Internet of Things and Applications (IOTA), Maharashtra Institue of Technology, Pune, India, Volume 3, Issue 2, Jan 2016, pp383- 388.

8. K. Hantrakul, P. Pramokchon, P. Khoenkaw, N. Tantitharanukul, and K. Osathanunkul, “Automatic Faucet with Changeable Flow based on MQTT protocol”,International Computer Science and Engineering Conference (ICSEC2016), Chiang Mai, Thailand, 14-17 Dec,2016.

9. “Smart toilets using turbidity sensor” in international journal of innovative technology and exploring engineering (IJITEE) ISSN:2278-3075, volume-8 Issue5S March,2019, ES345501839/19©BEIESP. By Mithya V, Divya Prabha.N, Sisma Samlein S, Madhumitha M .

10. “Smart toilet using BLE beacon technology” in proceeding of the International Conference and Electronics systems(ICCES 2018) IEEE Xplore part number CFP18AO-ART; ISBN:978-1-5386—4765-3, By Ms. Nidhi R Mishra, Mr. Paras M Suri, Dr.(Mrs.) Shalu Chopra.

11. 2015 International conference on control, Instrumentation, communication and Computational Technologies (ICCICCT)-“GPS enabled Employee Registration and Attendance Tracing System”, IEEE paper.

12. International Journal of computer applications (0975- 8887)- “vehicle Tracking, Monitoring and Alerting System: A Review” Volume 119 – No.10, June 2015

13. Sneha Jangid, Sandeep Sharam, “An embedded system model for air quality monitoring,” 2016 International conference on computing for sustainable global development (India.Com), school of ICT, Gautam Buddha University Greater Noida, India.

14. Xavier Gibert, Vishal M Patel, Rama Chellappa, in their IEEE paper titled as “Deep Muiti-task Learning for Railway Track Inspection” Volume 18, Issue1, jan 2017, pp 153-167

15. . A.D Kadge, A. K. Varute, P. G. Patil, P. R. Belukhi “Automatic sewage disposal system for train”, International journal of Emerging Research in management and technology (Volume-5, Issue-5), May