SMART DUSTBIN
A MINI-PROJECT REPORT

Submitted by

G HARINI 210701068

HARINI S 210701070

in partial fulfilment of the award of the degree

of

BACHELOR OF ENGINEERING

IN

COMPUTER SCIENCE AND ENGINEERING

RAJALAKSHMI ENGINEERING COLLEGE, CHENNAI

RAJALAKSHMI ENGINEERING COLLEGE, CHENNAI

An Autonomous Institute

CHENNAI

MAY 2024

BONAFIDE CERTIFICATE

Certified that this Thesis titled **“SMART DUSTBIN**” is the bonafide work of “**HARINI G (2116210701068), HARINI S (2116210701070)”** who carried out the work under my supervision. Certified further that to the best of my knowledge the work reported herein does not form part of any other thesis or dissertation on the basis of which a degree or award was conferred on an earlier occasion on this or any other candidate.

SIGNATURE

Dr . N.DURAIMURUGAN M.E.,Ph.D.,

PROJECT COORDINATOR

Professor

Department of Computer Science and Engineering Rajalakshmi Engineering College

Chennai - 602 105

Submitted to Project Viva-Voce Examination held on ________________

INTERNAL EXAMINER **EXTERNAL EXAMINER**

ACKNOWLEDGEMENT

First, we thank the almighty god for the successful completion of the project. Our sincere thanks to our chairman **Mr. S. Meganathan B.E., F.I.E.,** for his sincere endeavor in educating us in his premier institution. We would like to express our deep gratitude to our beloved Chairperson **Dr. Thangam Meganathan Ph.D.,** for her enthusiastic motivation which inspired us a lot in completing this project and Vice Chairman **Mr. Abhay Shankar Meganathan B.E., M.S.,** for providing us with the requisite infrastructure.

We also express our sincere gratitude to our college Principal, **Dr. S. N. Murugesan M.E., PhD.,** and **Dr. P. KUMAR M.E., PhD, Director computing and information science , and Head Of Department of Computer Science and Engineering** and our project coordinator **Dr. N.Duraimurugan M.E.,Ph.D.,** for her encouragement and guiding us throughout the project towards successful completion of this project and to our parents, friends, all faculty members and supporting staffs for their direct and indirect involvement in successful completion of the project for their encouragement and support.

TABLE OF CONTENTS

ABSTRACT

The goal of the planned IoT-enabled smart dustbin project is to automate garbage segregation and transform waste management through the use of cutting-edge technologies. Effective sorting at the moment of disposal is made possible by the system's ability to distinguish between wet and dry waste by integrating sensors and actuators within the dustbin. By encouraging appropriate waste disposal techniques, this innovation promotes environmental sustainability in addition to streamlining the waste collection process. The smart dustbin optimizes trash handling, lowers landfill usage, and opens the door for smarter, greener cities through real-time monitoring and data analysis. In this project, a new paradigm in urban sanitation is introduced through the confluence of IoT technology and waste management concepts. Using intelligent sensors to identify moisture content and other relevant parameters, the trashcan automatically divides incoming waste into sections for moist and dry waste. This improves the overall quality of recycled products by reducing contamination of recyclable materials and increasing waste collection efficiency. Stakeholders may remotely monitor fill levels, improve collection routes, and even incentivise sustainable waste practices with the incorporation of cloud connectivity, so promoting a more sustainable and cleaner environment for future generations. The system continuously improves its sorting abilities through sophisticated machine learning algorithms, adjusting over time to changing waste compositions and user behaviours. Municipalities can use data analytics to make well-informed decisions and resource allocation by gaining important insights into garbage generation patterns. Additionally, the incorporation of mobile applications and user-friendly interfaces encourages community engagement and environmental stewardship by enabling residents to actively participate in waste reduction initiatives.

LIST OF FIGURES

LIST OF ABBREVIATION

ABBREVIATION	ACCRONYM
NodeMCU	Open source IoT platform/micro-controller with Wi-Fi module integrated.
RAIN DROP SENSOR	Nickel coated in the form of lines to measure moisture.
SERVOMOTOR	Rotary actuator for precise control of angular position
IR SENSOR	Pyroelectric infrared detectors is for motion detection

CHAPTER 1
INTRODUCTION

1.1 INTRODUCTION

The Smart Dustbin IoT project introduces a novel solution to the omnipresent problem of waste management, marking a significant breakthrough in urban sustainability. Since more and more people live in metropolitan areas across the world, efficient waste management and segregation are essential for maintaining resource conservation and environmental health. The inadequate separation of recyclable materials from organic trash by conventional waste disposal procedures frequently results in inefficiency and environmental deterioration. Nevertheless, by utilizing Internet of Things (IoT) technology to automate the segregation process, the Smart Dustbin IoT project offers a revolutionary solution. The integration of sophisticated algorithms and smart sensors is expected to improve recycling and environmental responsibility in addition to streamlining waste collection.

1.2 SCOPE OF THE WORK

The Smart Dustbin IoT project's scope includes a wide range of activities meant to create, execute, and enhance a state-of-the-art waste management system. To find the best sensors, actuators, and IoT devices to integrate into the smart dustbin prototype, the project will first need extensive research and analysis. To evaluate the smart dustbin's dependability and performance in a range of usage situations and environmental circumstances, field testing will be carried out. In order to facilitate real-time monitoring and decision-making, the project will also involve the design of protocols for data transfer, storage, and analysis.

1.3 PROBLEM STATEMENT

The essential problem of ineffective waste management techniques in urban environment where ineffective segregation increases landfill consumption and degrades the environment is addressed by the Smart Dustbin IoT project. Recyclable materials are frequently not sufficiently separated from organic trash by the current waste disposal systems, which leads to contamination and lower recycling rates. Additionally, manual waste sorting procedures require a lot of labour and are prone to mistakes, which results in inefficient garbage disposal and collection. Through the development of a solution, this project seeks to optimize garbage collection procedures, minimize the amount of waste that ends up in landfills.

1.4 AIM AND OBJECTIVES OF THE PROJECT

By creating an intelligent system that can independently separate wet and dry garbage at the point of disposal, the Smart Dustbin IoT project seeks to transform waste management techniques. By incorporating cutting-edge sensors, actuators, and Internet of Things technologies, the project aims to improve the efficacy and efficiency of waste collection procedures in urban settings. The method seeks to decrease contamination, increase recycling rates, and lessen the environmental effect of inappropriate garbage disposal by automating trash sorting. The initiative also intends to provide real-time data insights to waste management businesses and municipalities, enabling them to better allocate resources, expedite processes, and support environmentally friendly trash management techniques. The Smart Dustbin IoT project's main goal is to solve the urgent problems brought on by the growing amounts of rubbish produced in cities across the globe. The project intends to give waste management agencies and municipalities the instruments they need to efficiently manage waste streams and advance environmental sustainability by creating a scalable and flexible solution. The project aims to limit landfill utilization, lower greenhouse gas emissions related to garbage disposal, and optimize waste collection routes by implementing cutting-edge technologies, such as machine learning algorithms and cloud connectivity. Additionally, the project seeks to instill in the public a sense of environmental responsibility and the value of proper waste segregation and recycling in order to promote a cleaner, healthier Earth.

CHAPTER 2

LITERATURE SURVEY

[1] Smart Garbage Segregation and Management Using Iot. This literature says about population growth has resulted in a massive increase in pollution. It has the potential to cause a slew of chronic illnesses. To eliminate or reduce rubbish and maintain cleanliness, a smart garbage management architecture is required. However, there is another major issue: segregating the collected garbage. The microcontroller serves as an interface between the sensors and the IoT module.

[2] Iot Based Automatic Waste Segregator. This study describes an IoT-based automatic garbage segregator system that aims to expedite home waste sorting while reducing human intervention. The system uses sensors to detect and categorize trash into dry, moist, and metallic varieties, allowing for efficient segregation at the source. Furthermore, the system shows dustbin rubbish levels on an LCD screen and sends alarms via GSM and Arduino when the bins are full. The experimental results confirm successful waste isolation and demonstrate the system's usefulness.

[3] Energy saving smart waste segregation and notification system.The research emphasizes the critical need for waste segregation in Bangladesh due to rapid urbanization and industry, which worsen environmental concerns. The old manual segregation approach is viewed as inefficient and dangerous to workers' health. To solve these issues, a smart segregation system is presented that aims to improve waste collection while minimizing environmental effect. This technology alerts workers and city officials when bins reach capacity, increasing efficiency and lowering greenhouse gas emissions. Integrating solar energy also enhances energy efficiency, providing a long-term solution to Bangladesh's waste management concerns.

[4] Iot Based intelligent route selection of wastage segregation for smart cities using solar energy. The article introduces the Smart Bin, an innovative automated waste management system designed to address the major difficulties of global garbage management, particularly in the context of smart cities. This cutting-edge solution automates trash segregation processes using innovative technology such as HC-SR04 ultrasonic sensors, TowerPro SG90 servo motors, and an Arduino Uno microcontroller. The Smart Bin efficiently separates dry and moist rubbish using IoT sensors built into the bin, hence optimizing waste collection and disposal. Furthermore, when the bin reaches 80% capacity, it automatically sends warning signals to garbage collectors, assuring prompt waste pickup. Notably, the Smart Bin incorporates ecological practices by capturing solar energy via fixed solar panels, allowing for self-sufficient functioning while reducing environmental effect. Its versatility extends across numerous urban areas.

CHAPTER 3
SYSTEM SPECIFICATIONS

3.1 HARDWARE SPECIFICATIONS

NodeMCU	:	ESP32 (38 Pins)
IR Sensor	:	Proximity Sensor
Rain Drop Sensor	:	LM393 (Volt-comp)
Servo Motor	:	2.2KW
Power Supply	:	5V DC

3.2 SOFTWARE SPECIFICATIONS

The software requirements document is the specifications of the system. It should include both a definition and a specification of requirements. It is a set of what the system should rather be doing than focus on how it should be done. The software requirements provide a basis for creating the software requirements specification. It is useful in estimating the cost, planning team activities, performing tasks, tracking the team, and tracking the team's progress throughout the development activity.

ARDUINO IDE, and **Chrome** would all be required

CHAPTER 4

MODULE DESCRIPTION

IR SENSOR

An infrared (IR) sensor is a device that detects infrared radiation produced by nearby objects and converts it into a measurable signal. These devices, which operate on the basis of detecting changes in IR radiation intensity, are widely utilized in a variety of applications, including motion detection, proximity sensing, and temperature monitoring. Passive infrared sensors detect natural heat released by objects, whereas active infrared sensors emit infrared radiation and measure reflections or changes in the radiated signal. This adaptability makes infrared sensors important in security systems, consumer electronics, automotive applications, and industrial automation, where they enable tasks such as motion detection, object tracking, and distance measuring, hence improving safety, convenience, and efficiency.

NODE-MCU

The NodeMCU is a flexible open-source firmware and development kit built on the ESP8266 Wi-Fi module. It combines a microcontroller unit (MCU) with Wi-Fi capabilities to allow for smooth prototype and development of Internet of Things (IoT) projects. Its design includes GPIO ports, analog-to-digital converters (ADCs), and Lua programming language support, providing an intuitive environment for connecting sensors, actuators, and other devices to the internet. With its low cost, small form factor, and extensive community support, the NodeMCU has grown in popularity among hobbyists, makers, and professionals for efficiently building Wi-Fi-enabled applications and IoT solutions, establishing it as a go-to choice in the realm of connected devices.

RAIN DROP SENSOR

A raindrop sensor is a small gadget designed to detect precipitation or moisture. The one in question uses a set of conductive traces and relies on changes in conductivity generated by water droplets spanning the traces. When rain falls on the sensor's surface, the conductivity between these traces rises, causing the sensor to emit a signal signaling wetness. These sensors are widely used in weather monitoring systems, automatic irrigation settings, and car rain detection systems, providing a simple yet effective method of detecting and responding to changes in environmental moisture levels. Their simplicity, combined with their dependability, makes them indispensable tools for a wide range of applications that require real-time precipitation monitoring and control.

SERVO MOTOR

A servo motor is a type of motor that is widely used in applications requiring accurate control of angular position, speed, and acceleration. Unlike ordinary motors, servo motors have a closed-loop feedback system that constantly checks and adjusts the motor's position in response to input signals. This feedback system often includes a position sensor, such as an encoder, that sends feedback to the motor controller, allowing it to modify the motor's output accordingly. Servo motors are noted for their great precision, accuracy, and responsiveness, making them suitable for use in robotics, CNC machines, 3D printers, and automated systems. They come in a variety of sizes and designs to meet varied torque and speed needs. Furthermore, servo motors may function at a wide variety of speeds.

VIBRATION SENSOR

Vibration sensors incorporated into smart dustbins provide an innovative way to improve waste management systems. These sensors detect motion or vibration within the bin and provide current data about the fill level and usage trends. Monitoring these signals allows governments to efficiently organize waste collection routes, avoiding wasted trips and maximizing resource allocation. Furthermore, vibration sensors provide preventative maintenance by detecting issues such as blockages or mechanical breakdowns, ensuring that the waste disposal infrastructure remains operational. With the addition of vibration sensors, smart dustbins become not only more efficient but also more responsive, helping to cleaner, smarter, and greener cities.

ARDUINO SOFTWARE

The Arduino software, often known as the Arduino IDE (Integrated Development Environment), is a user-friendly environment for programming Arduino microcontrollers. It offers a simple interface for creating, compiling, and uploading code to Arduino-compatible devices. The Arduino IDE is a C/C++-based programming language designed for embedded systems, making it simple for novices while yet providing complex functionality for expert developers. With a large user base and copious documentation, Arduino software allows for rapid development and testing in a variety of projects, including robotics and home automation, as well as wearable technology and IoT applications.

CHAPTER 5

SYSTEM DESIGN

5.1 ARCHITECTURE DIAGRAM

An architectural diagram depicts the components, linkages, and interactions of an arrangement or structure. It functions as a blueprint for comprehending the architecture's design, including its different levels, components, and dependencies. These diagrams are useful tools for architects, developers, and stakeholders, helping them communicate, make decisions, and identify potential design defects or optimization opportunities. Teams can work more efficiently and maintain alignment with project objectives throughout the development lifecycle when the architecture is clearly visualized.

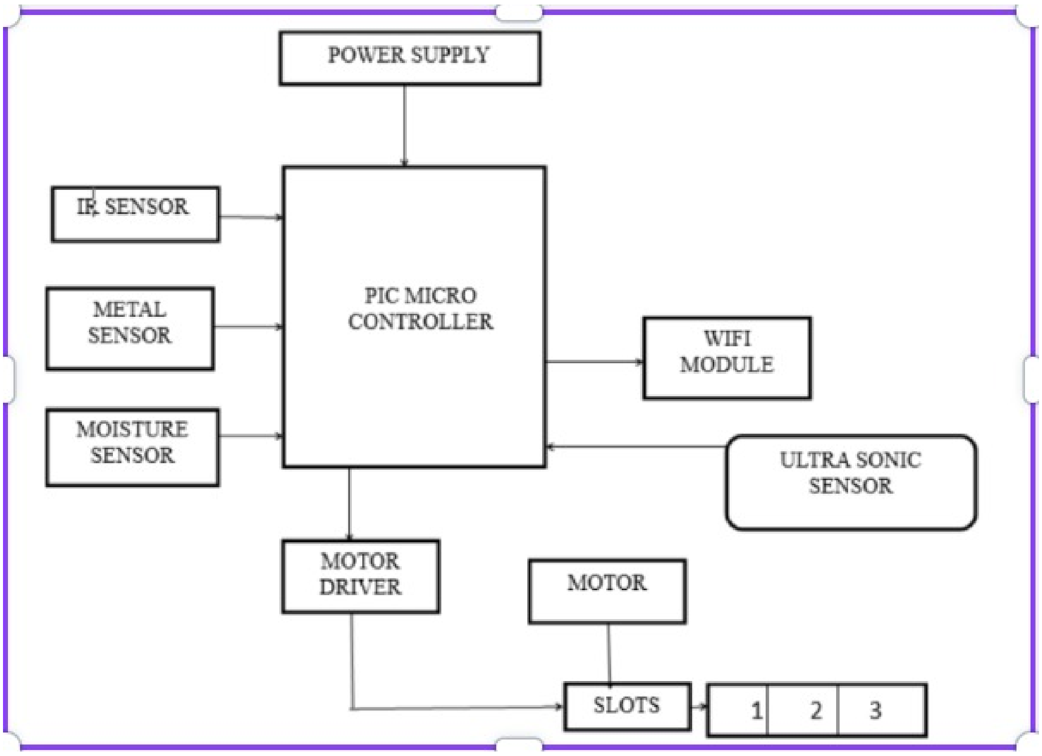


Figure 5.1 Architecture Diagram

5.2 FLOW CHART

A flowchart is a type of diagram that represents an algorithm, workflow or process. The flowchart shows the steps as boxes of various kinds, and their order by connecting the boxes with arrows. This diagrammatic representation illustrates a solution model to a given problem.

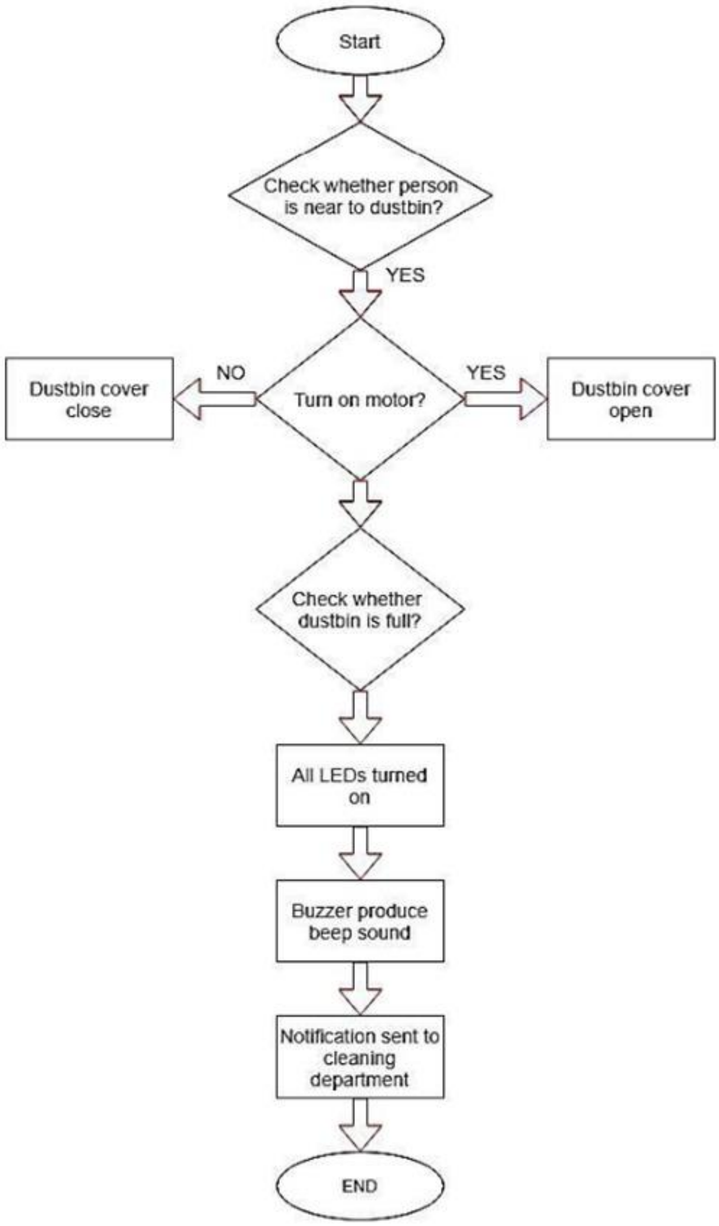


Figure 5.2 Flow Chart

5.3 CIRCUIT DIAGRAM

In an Internet of Things (IoT) circuit diagram, you'll find a visual representation of the electronic components and their connections that make up a specific IoT device or system. At the core of the diagram is usually a microcontroller or microprocessor, acting as the device's brain, managing data processing and controlling other elements. Surrounding this central component are sensors, which detect various physical properties or environmental conditions, and actuators, which carry out actions based on the sensor data from the microcontroller.

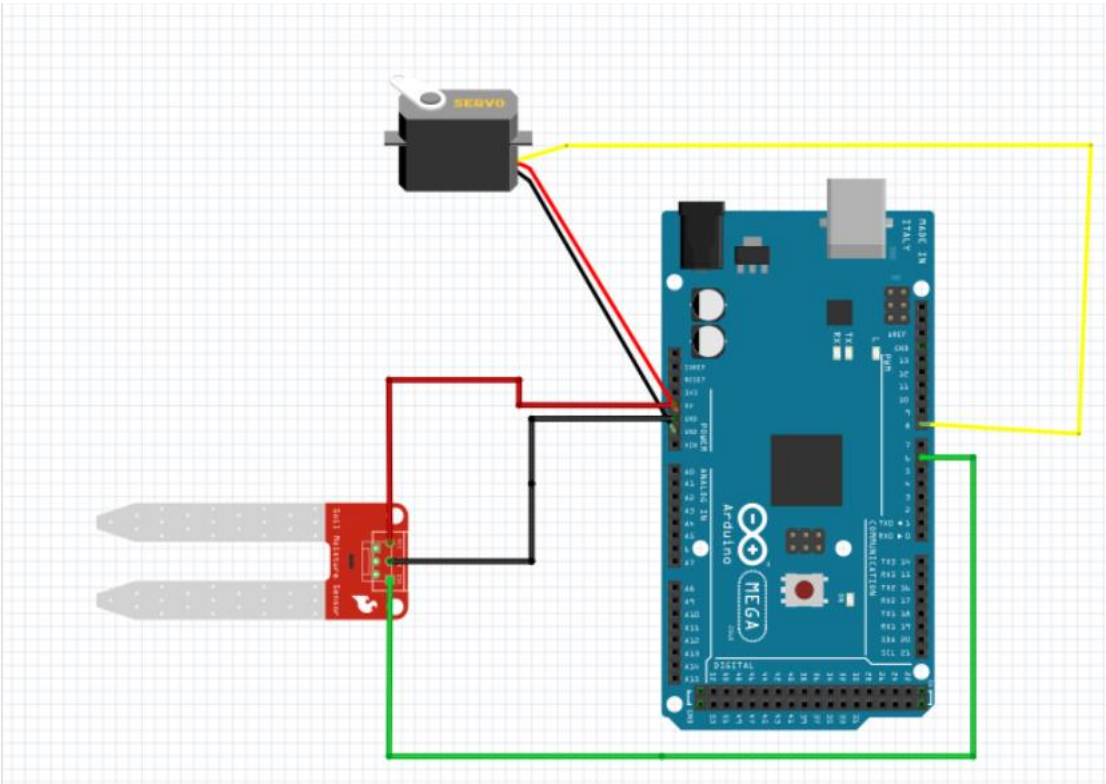


Figure 5.3 Circuit Diagram

CHAPTER 6

SAMPLE CODING :

```
#include <ESP8266WiFi.h>
#include "Adafruit_MQTT.h"
#include "Adafruit_MQTT_Client.h"

#include <Servo.h>

Servo myservo;
/********* WiFi Access Point ***********/

#define WLAN_SSID       "Unknown"
#define WLAN_PASS       "43214321"

/********* Adafruit.io Setup ***********/

#define AIO_SERVER      "io.adafruit.com"
#define AIO_SERVERPORT  1883                   // use 8883 for SSL
#define AIO_USERNAME    "jeff_ffej"
#define AIO_KEY         "aio_FHiZ24aKH9yGztbDTg8PSKlfwrt0"

/**** Global State (you don't need to change this!) ******/

// Create an ESP8266 WiFiClient class to connect to the MQTT server.
WiFiClient client;
// or... use WiFiClientSecure for SSL
//WiFiClientSecure client;
```

```
// Setup the MQTT client class by passing in the WiFi client and MQTT server and login details.
Adafruit_MQTT_Client mqtt(&client, AIO_SERVER, AIO_SERVERPORT, AIO_USERNAME, AIO_KEY);

/********** Feeds *************/

// Setup a feed called 'photocell' for publishing.
// Notice MQTT paths for AIO follow the form: <username>/feeds/<feedname>
Adafruit_MQTT_Publish photocell = Adafruit_MQTT_Publish(&mqtt, AIO_USERNAME "/feeds/photocell");


/********* Sketch Code ************/

// Bug workaround for Arduino 1.6.6, it seems to need a function declaration
// for some reason (only affects ESP8266, likely an arduino-builder bug).
void MQTT_connect();
int pos=90;
void setup() {
  Serial.begin(115200);
  delay(10);

  Serial.println(F("Adafruit MQTT demo"));

  // Connect to WiFi access point.
  Serial.println(); Serial.println();
  Serial.print("Connecting to ");
  Serial.println(WLAN_SSID);
```

```
  WiFi.begin(WLAN_SSID, WLAN_PASS);
  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
  }
  Serial.println();

  Serial.println("WiFi connected");
  Serial.println("IP address: "); Serial.println(WiFi.localIP());
  myservo.attach(D6);  // attaches the servo on GIO2 to the servo object
pinMode(D5,INPUT_PULLUP);
pinMode(D2,INPUT_PULLUP);
myservo.write(pos);
}

uint32_t x=0;

void loop() {
  // Ensure the connection to the MQTT server is alive (this will make the first
  // connection and automatically reconnect when disconnected).  See the
MQTT_connect
  // function definition further below.
  MQTT_connect();

  // this is our 'wait for incoming subscription packets' busy subloop
  // try to spend your time here

  if(!digitalRead(D5)){
```

```
  if(!digitalRead(D2)){
   for (pos = 90; pos <= 180; pos += 1) {  // goes from 0 degrees to 180 degrees
  // in steps of 1 degree
  myservo.write(pos);  // tell servo to go to position in variable 'pos'
  delay(15);          // waits 15ms for the servo to reach the position
 }
photocell.publish("wet");
 delay(3000);
 for (pos = 180; pos >= 90; pos -= 1) {  // goes from 180 degrees to 0 degrees
  myservo.write(pos);                 // tell servo to go to position in variable 'pos'
  delay(15);                          // waits 15ms for the servo to reach the position
 }
  }
  else{
 for (pos = 90; pos >= 0; pos -= 1) {  // goes from 180 degrees to 0 degrees
  myservo.write(pos);                 // tell servo to go to position in variable 'pos'
  delay(15);                          // waits 15ms for the servo to reach the position
 }
photocell.publish("dry");
 delay(3000);
   for (pos = 0; pos <= 90; pos += 1) {  // goes from 0 degrees to 180 degrees
  // in steps of 1 degree
  myservo.write(pos);  // tell servo to go to position in variable 'pos'
  delay(15);          // waits 15ms for the servo to reach the position
 }
  }
 }
```

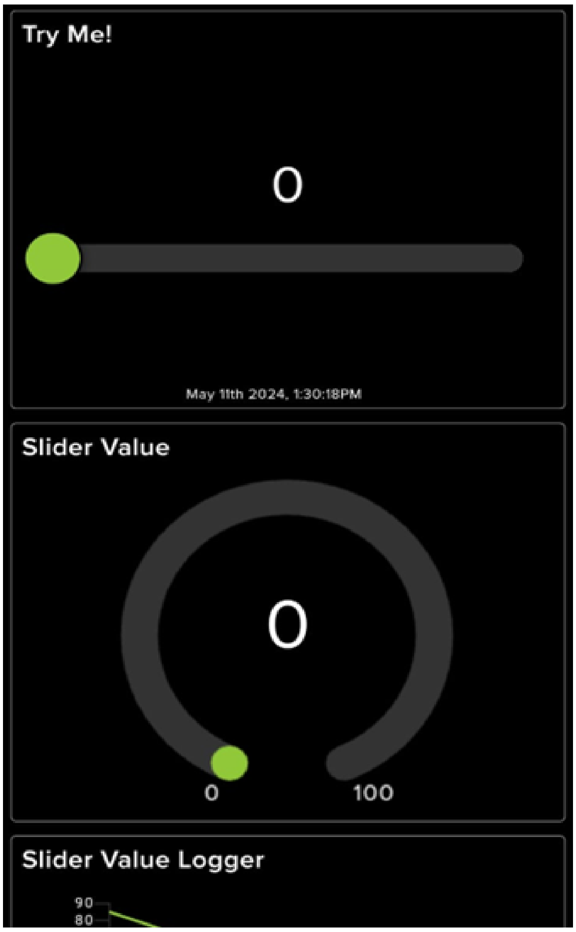
```
// Now we can publish stuff!
  // ping the server to keep the mqtt connection alive
  // NOT required if you are publishing once every KEEPALIVE seconds
  /*
  if(! mqtt.ping()) {
    mqtt.disconnect();
  }
  */
}

// Function to connect and reconnect as necessary to the MQTT server.
// Should be called in the loop function and it will take care if connecting.
void MQTT_connect() {
  int8_t ret;

  // Stop if already connected.
  if (mqtt.connected()) {
    return;
  }

  Serial.print("Connecting to MQTT... ");

  uint8_t retries = 3;
  while ((ret = mqtt.connect()) != 0) { // connect will return 0 for connected
       Serial.println(mqtt.connectErrorString(ret));
       Serial.println("Retrying MQTT connection in 5 seconds...");
       mqtt.disconnect();
       delay(5000);  // wait 5 seconds
       retries--;
```

```
      if (retries == 0) {
        // basically die and wait for WDT to reset me
        while (1);
      }
  }
  Serial.println("MQTT Connected!");
}
```

CHAPTER 7

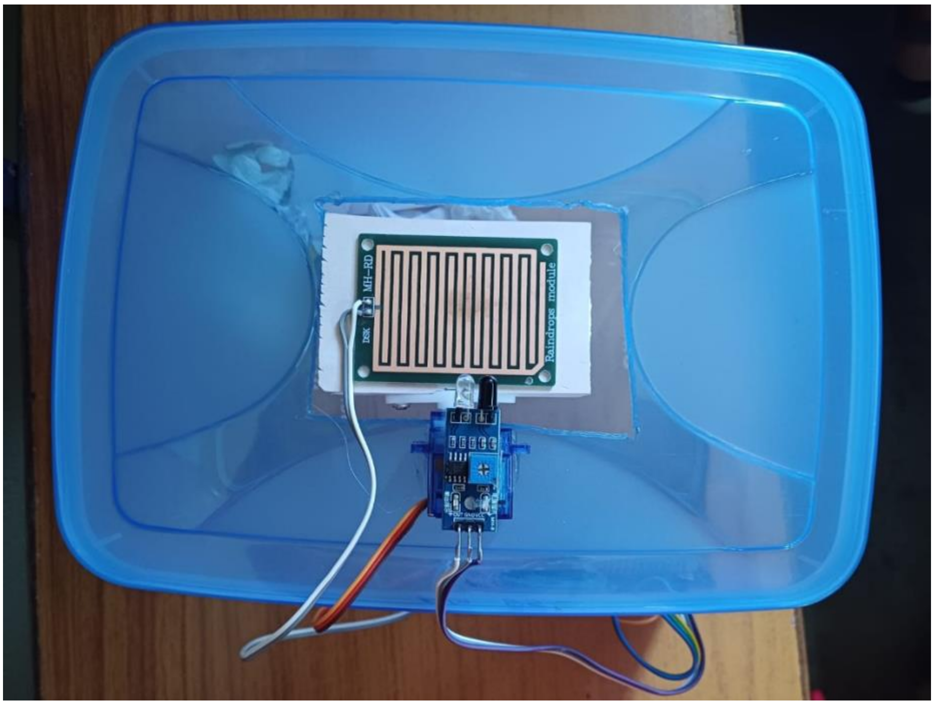
SCREEN SHOTS

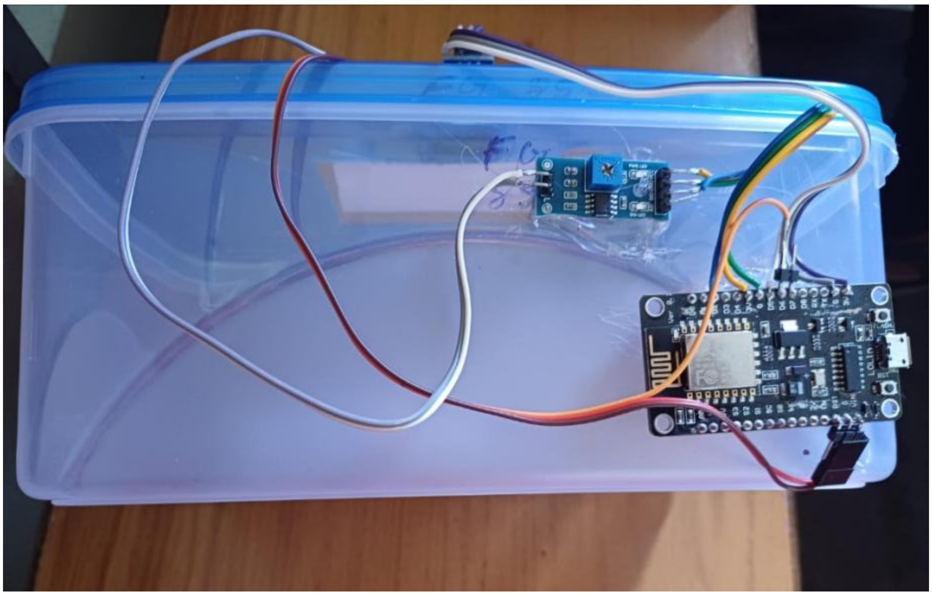
OPEN-SOURCE WEBSITE DISPLAY OF RESULT:



The open source webiste dashboard is used to display whether the garbage content detected is Wet waste or dry waste after the motion sensor detects the object and the rain drop sensor detects the presence of moisture to classify the waste as dry or wet.

PROTOTYPE MODEL





CHAPTER 8

CONCLUSION AND FUTURE ENHANCEMENT

In summary, the Smart Dustbin IoT project offers a workable answer to the problems caused by growing urbanization and unsustainable trash disposal techniques, marking a substantial advancement in contemporary waste management techniques. The project shows the potential to transform the way we handle and dispose of waste in urban contexts by using IoT technology to automate waste segregation and optimize collection operations. The project's novel strategy not only advances environmental sustainability but also improves resource utilization and operational efficiency for waste management bodies and municipalities. In order to increase trash segregation efficiency and accuracy, future improvements to the Smart Dustbin IoT project may involve the addition of more sensors and sophisticated machine learning algorithms. In order to lessen the smart dustbins' carbon impact and operational expenses, the project might also look into the idea of using renewable energy sources, including solar panels. Furthermore, broadening the project's scope to incorporate elements like intelligent invoicing systems for waste collection services and public awareness initiatives may aid in advancing environmentally friendly waste management techniques and stimulating community involvement. All things considered, more study and development in this area could result in trash management strategies for future cities that are more intelligent, effective, and sustainable.

REFERENCES

[1] S. V. Kumar, T. S. Kumaran, A. K. Kumar and M. Mathapati, "Smart garbage monitoring and clearance system using internet of things", 2017 IEEE International Conference on Smart Technologies and Management for Computing Communication Controls Energy and Materials (ICSTM), pp. 184-189, 2017.

[2] M. Arebey, M. A. Hannan, H. Basri and H. Abdullah, "Solid waste monitoring and management using RFID GIS and GSM", 2009 IEEE Student Conference on Research and Development (SCOReD) UPM Serdang, pp. 37-40, 2009.

[3] S. Thakker and R. Narayanamoorthi, "Smart and wireless waste management", 2015 International Conference on Innovations in Information Embedded and Communication Systems (ICIIECS), pp. 1-4, 2015.

[4] M. Hannan, M. Arebey, R. A. Begum and H. Basri, "Radio Frequency Identification (RFID) and communication technologies for solid waste bin and truck monitoringsystem", Waste Management, vol. 31, pp. 2406-2413, 2011.

[5] J. Joshi et al., "Cloud computing based smart garbage monitoring system", 2016 3rd International Conference on Electronic Design (ICED), pp. 70-75, 2016.

[6] A. S. Bharadwaj, R. Rego and A. Chowdhury, "IoT based solid waste management system: A conceptual approach with an architectural solution as a smart city application", 2016 IEEE Annual India Conference (INDICON), pp. 1-6, 2016.