

RV UNIVERSITY, BENGALURU-59

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING



A Project Report On
Smart Waste Bin using IoT Sensors & Machine Learning
Submitted in partial Fulfillment for the award of degree of
B.Tech(Honors)

In

School of Computer Science and Engineering
Submitted By

Name1: Aneesh Adithya SR	USN1:1RVU23CSE054
Name2: Amogh Madhu Kashyap	USN2:1RVU23CSE045
Name3: Rajath D Kudur	USN3:1RVU23CSE366
Name4: Vaibhav M Chittakki	USN4:1RUA24CSE7019

Under the Guidance of

Dr./Prof. Monesh N

Designation

School of CSE

RV University, Bengaluru-560059

2025-2026

DECLARATION

We hereby declare that the project report entitled “Smart Waste Bin using IoT Sensors & Machine Learning” submitted by us to RV University, Bengaluru, is a record of bona fide work carried out by us under the supervision of Prof. Monesh N, School of Computer Science and Engineering, RV University. We further declare that this work has not been submitted to any other institution for the award of any degree or diploma.

Place: Bengaluru

Date: 5th November 2025

Signature of the Student(s)

Name(s):

CERTIFICATE

This is to certify that the project report entitled “Smart Waste Bin using IoT Sensors & Machine Learning” submitted by Name of the Student(s), Aneesh Adithya SR (USN:1RVU23CSE054), Amogh Madhu Kashyap (USN:1RVU23CSE045), Rajath D Kudur (USN:1RVU23CSE366), Vaibhav M Chittakki (USN:1RUA24CSE7019), School of Computer Science and Engineering, RV University, Bengaluru, for the award of the degree of B.Tech (H) in Computer Science, is a record of bona fide work carried out under my supervision.

Signature of the Guide

Prof. Monesh N

School of Computer Science and Engineering

RV University, Bengaluru

ACKNOWLEDGEMENT

We express our sincere gratitude to Prof. Monesh N, our project guide, for their guidance, continuous support, and encouragement throughout this project. We extend our thanks to the Dean, Program Director, and all faculty members of the School of Computer Science and Engineering for their valuable inputs. We are also thankful to our peers, friends, and family for their constant support and motivation.

ABSTRACT

This project explores a smart waste bin system using sensors and machine learning to automatically identify waste type and monitor bin capacity. The objective is to improve waste management efficiency and promote automated waste segregation. The study involves real-time sensing using gas, moisture, and ultrasonic sensors, along with a CNN-based waste classification model. The implementation was carried out using ESP32-S3, Python Flask, OpenCV, and TensorFlow. The project results show accurate wet-waste detection, reliable fill-level monitoring, and timely buzzer alerts. This work contributes to smart city development by automating waste sorting and enabling IoT-based environmental monitoring.

TABLE OF CONTENTS

1. Introduction
2. Objectives
3. Literature Survey and Research Gap
4. Project Description and Design
5. Methodology / Technical Implementation
6. Results and Discussion
7. Conclusion and Future Scope
8. References

INTRODUCTION

Cities and campuses produce a lot of waste every day. Most bins are dumb — they just sit there until someone collects them on a fixed schedule. This causes two big problems: bins either overflow (gross and unhygienic) or trucks go out half-empty (wasteful time and fuel). Also, mixing wet and dry waste at the collection point makes recycling harder. Detecting which bins are full or which contain wet/rotten waste early can save money, reduce smells and pests, and improve recycling.

We wanted a practical, low-cost way to make bins “smart” so they can tell people and managers what’s happening in real time. The motivation is simple:

- Prevent overflow and the health risks that come with it.
- Automatically detect wet or rotting waste (so it can be handled differently).
- Help separate dry vs wet waste so recycling is easier and cleaner.
- Give managers an easy dashboard to see bin status without walking around.

Making this system work with cheap hardware (an ESP32-S3 microcontroller, a few common sensors, and a portable system for heavier AI work) lets schools, small offices, and public places adopt it without major investment.

Problem Context

Current waste-collection systems are mostly schedule-based and reactive. Problems we face in this context:

1. Overflow: A bin becomes full and spills before the next pickup, causing litter, smell, and disease risk.
2. Poor segregation: Wet (food/organic) and dry (plastic, paper, metal) waste often get mixed, which ruins recycling value.
3. Late detection of spoilage: Wet/rotten waste emits gases and moisture before it’s obvious; by the time it’s noticed, the problem is worse.
4. Inefficient routing: Empty or near-empty bins are still collected on schedule, wasting time/fuel.
5. Lack of actionable data: Grounds staff or municipal teams don’t have live data to prioritize pickups.

Those problems are especially bad in dense spaces (college campuses, markets, malls) where bins are used heavily and continuously.

OBJECTIVES

1. Build an automated smart dustbin system

To design a modern dustbin that uses sensors and microcontroller technology to automatically monitor waste conditions, helping reduce manual checking, improve cleanliness, and support smarter waste management in public and home environments.

2. Monitor bin fill-level in real-time

To measure how full the bin is using an ultrasonic distance sensor and convert this into percentage data. This allows early alerts before overflow, improving collection planning and hygiene.

3. Detect wet vs dry waste

To identify whether the thrown waste is wet or dry using a moisture sensor. This helps enable automatic waste segregation, which increases recycling efficiency and reduces environmental pollution and landfill pressure.

4. Sense harmful gases and odors

To measure air quality inside the bin using an MQ-135 gas sensor. Detecting increased gas levels helps identify decaying waste early and prevents foul smell spread around the environment.

5. Classify waste status automatically

To process sensor readings and decide if waste is wet, dry, or smelly. This automated decision system removes human error and saves time for waste management workers and users.

6. Send live data to server

To transmit sensor values wirelessly from ESP32-S3 to a Flask server in real-time over Wi-Fi. This enables remote monitoring without needing physical contact or manual checking of dustbin status.

7. Create a live monitoring dashboard

To develop a web dashboard that displays gas value, moisture level, distance reading, fill percentage, and waste status. The dashboard provides a clear and easy interface for users and administrators.

8. Trigger smart alerts

To send alerts when the bin is full, wet waste is detected, or odor rises. This makes waste disposal proactive instead of reactive, stopping problems before they become unhygienic or messy.

9. Reduce manual effort and human involvement

To minimize human monitoring and repeated bin checking by automating the process with sensors and cloud dashboard. This saves time, improves efficiency, and supports smart-city waste automation.

10. Promote healthy and cleaner environment

To reduce foul smells, overflowing bins, and unhygienic waste handling by detecting waste type and fullness early. This ultimately helps maintain cleaner streets, homes, colleges, and public places.

LITERATURE SURVEY AND RESEARCH GAP

Summarize previous studies and identify the research gap your work addresses.

SN	Paper title	DOI	Methodology / Algorithms used	Datasets	Evaluation Metrics	Tools / frameworks	Research gap addressed/observed	Results / Findings	Remarks
1	Waste Classification with CNN (H5 Model) – Kaggle Notebook (trident3)	N/A (Kaggle)	CNN (transfer learning), image preprocessing, training & H5 export	Kaggle waste images (plastic, paper, metal, etc.)	Accuracy, loss, confusion matrix (notebook)	Python, Keras/TensorFlow, OpenCV	Real-world deployment (camera + IoT) and sensor fusion not covered	High accuracy in clean, curated images; good baseline model	Used as our starting point and integrated with webcam + IoT dashboard
2	Classification of Waste Materials using CNN (ACM)	10.1145/3574318.3574345	CNN with transfer learning; multi-class classification	Public waste image datasets (7 classes)	Accuracy; class-wise performance	Python, TensorFlow/Keras	Focused on pure vision; no gas/moisture/level sensing or embedded constraints	Reported strong multi-class accuracy on benchmark data. (dl.acm.org)	Supports our vision approach; we extend with sensors + edge constraints
3	Smart eNose Food Waste Management	10.1155/2021/9931228	Electronic nose; gas sensor array (incl. MQ-135); pattern recognition	Lab food samples (meat, rice, bread, etc.)	Classification accuracy; sensor response profiles	Arduino/MCU + data processing	Lacked camera fusion; limited to gas signatures; no	MQ-135 helpful for VOC/odor linked to organic	We adopt MQ-135 with auto-base line and fuse with moisture

	(MQ-4 & MQ-135) (Hindawi)						bin-fill logic	waste. onlinelibrary.wiley.com)	+ vision
4	Waste Management System Using IoT-Based ML (Hindawi)	10.1155/2020/6138637	IoT sensing + ML for waste level prediction and monitoring	Prototype bin data (ultrasonic etc.)	Prediction accuracy; response time	MCU (ESP/Arduin o), Python	No camera-based reliable fill-level recognition; limited alert/UI design	Demonstrates reliable fill-level monitoring via IoT. onlinelibrary.wiley.com)	We add ML image recognition + multi-sensor fusion + web dashboard

PROJECT DESCRIPTION

This project implements an IoT-integrated smart bin system that monitors waste conditions, bin fill level, odor and moisture, and categorises waste as wet or dry. A microcontroller board (ESP32-S3) reads sensors—an ultrasonic distance sensor to measure how full the bin is, a moisture sensor to detect wet material, and an MQ-135 gas sensor to detect odors linked to rotting or wet waste. For advanced waste classification, a webcam captures images of waste entering the bin, and a machine learning model running on a laptop classifies the waste type (e.g., plastic, paper, metal). All data is sent over WiFi to a web server, and a dashboard displays real-time values, alerts (bin full, odor detected, wet waste), and derived status (wet vs dry waste). This makes waste management proactive and automated, reducing manual checks and improving hygiene.

System Architecture

The architecture is divided into three main layers:

1. **Edge-sensor layer:** The ESP32-S3 board is mounted on the smart bin and interfaces directly with the sensors (ultrasonic, moisture, MQ-135) and also handles WiFi communication.
2. **Processing layer:** When waste enters, the webcam captures images and sends them to a laptop running a machine-learning model. The model (a convolutional neural network) classifies the type of waste. Meanwhile, the ESP32 continuously sends sensor readings to a Flask web server over the local network. The server computes derived values—such as bin fill percentage and waste status (“Wet Waste” or “Dry Waste”)—and checks for threshold-based alerts (e.g., gas baseline +500, moisture < 3200, bin distance < 5 cm).
3. **Presentation layer:** A web dashboard (built with Flask, HTML/CSS/JavaScript) shows live values, status, and alerts. The architecture supports remote monitoring via mobile or desktop, enabling staff to act when thresholds are crossed. The system is modular and scalable: additional bins or cloud storage can be added.

Tools & Technologies Used

Tools	Technology Used
Hardware	ESP32-S3 development board (WiFi-enabled microcontroller) Ultrasonic sensor (HC-SR04 or equivalent) for bin fill measurement Capacitive moisture sensor for wet waste detection MQ-135 gas sensor for odor detection Webcam (USB) connected to laptop for image capture
Software/Firmware	Arduino IDE or PlatformIO to program the ESP32-S3 firmware HTTP/REST communication between ESP32 and server Python 3 with Flask framework for web server backend and API endpoints HTML, CSS, JavaScript for dashboard frontend with auto-refresh logic
Machine Learning	CNN model for waste classification (trained externally). The model was utilised on a laptop to process webcam images and send classification results to the same server.
Dataset	The image classification dataset was taken from Kaggle: “Waste Classification with CNN with H5 model” from Kaggle notebook by user <i>trident3</i> . https://www.kaggle.com/code/trident3/waste-classification-with-cnn-with-h5-model/notebook This dataset contains labelled images of different waste categories (plastic, paper, metal, glass, etc.). The CNN model was trained on this dataset to generate an H5 model file, which the laptop uses to classify captured images.
Networking & Data	WiFi (2.4 GHz) local network for communication between ESP32, laptop, and server. REST API endpoints (<code>/update</code> , <code>/live</code>) for sensor data ingestion and frontend retrieval.

METHODOLOGY / TECHNICAL IMPLEMENTATION

Our smart waste-bin system works by combining IoT sensors, AI image processing, and a web dashboard so waste can be detected automatically in real-time.

First, the ESP32 microcontroller is powered on and gets connected to Wi-Fi. It immediately starts calibrating the gas sensor (MQ-135) so the system knows the normal air quality levels before waste is detected. Once calibration is done, the bin enters live monitoring mode.

The ESP32 continuously reads three things:

- Ultrasonic sensor → checks distance inside bin to calculate waste fill level.
- Moisture sensor → checks if waste is wet/organic.
- Gas sensor → detects gases released from wet/organic waste.

These readings help classify if the bin has wet waste or dry waste, and whether the bin is almost full or not.

Based on sensor values, the ESP32 triggers:

- Red LED (wet indicator) when wet waste is detected
- Yellow LED (bin full indicator) when bin is near 100% full

Buzzer alerts

- Fast beeps for wet waste
- 3-beep cycles when bin is full

At the same time, the ESP32 sends data to the Flask server via Wi-Fi every second. The server updates a live dashboard showing:

Gas level

Moisture level

Fill percentage

Bin full or not

Wet waste detected or not

System status

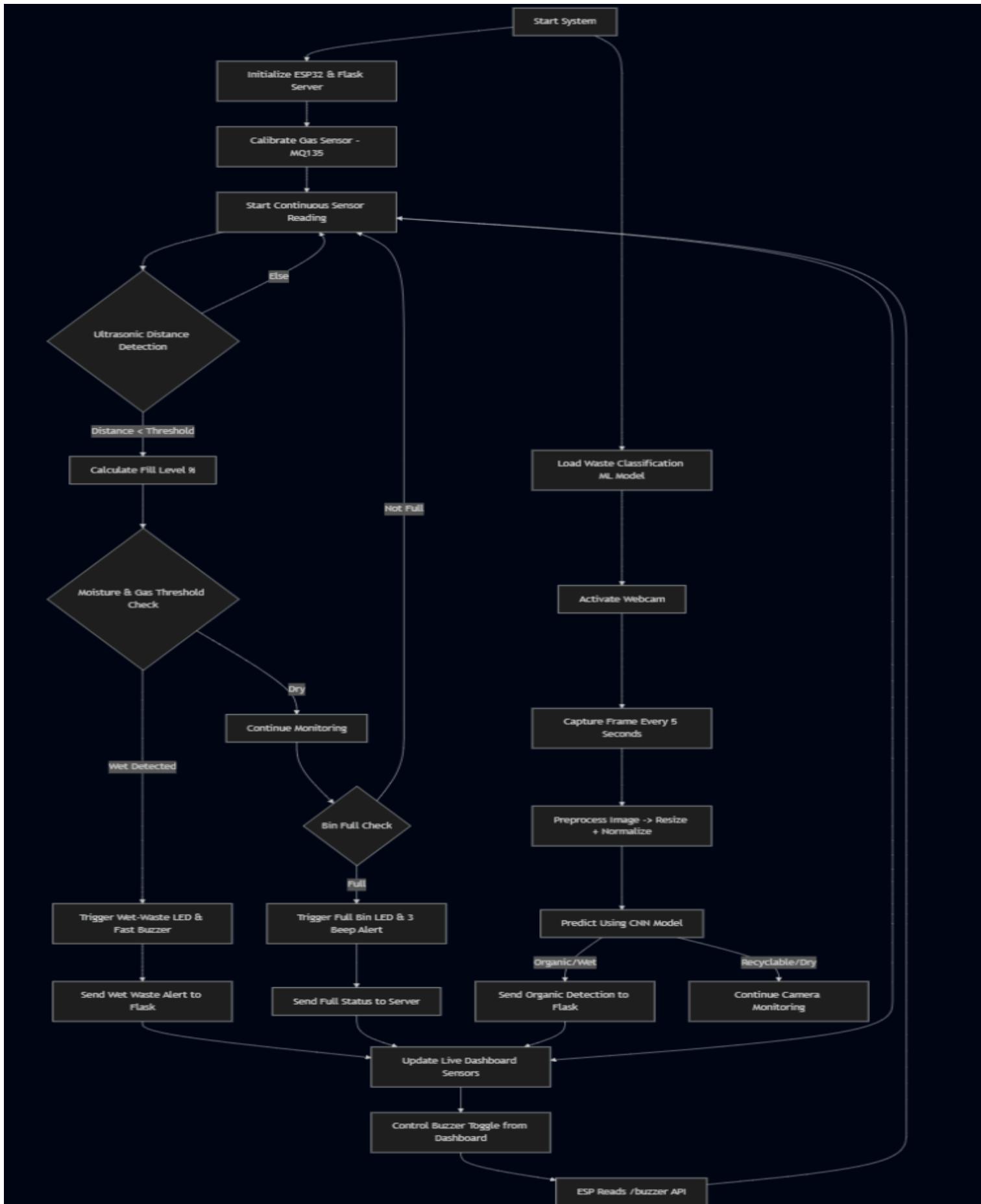
The dashboard also gives control to mute or activate the buzzer anytime. When the user presses the button, the server instantly sends a command back to the ESP32.

Alongside sensors, an AI model runs on the server and analyzes camera images every few seconds to classify waste more intelligently — especially useful to double check wet waste or detect recyclable items. The camera feed is pre-processed, passed through a CNN model, and the model output is sent to the dashboard.

Overall, the system works in a continuous loop:

Sense → Decide → Alert → Display → Control

This real-time feedback improves waste sorting automation and reduces overflowing bins — making the solution scalable for future smart-city waste systems.



CONCLUSION AND FUTURE SCOPE

This project successfully built a smart waste-monitoring system that automatically detects wet waste, tracks bin fill levels, and sends live updates to a web dashboard. Using ESP32, sensors, and an AI model from Kaggle, the system reduces human effort in waste sorting and prevents overflowing bins. Real-time alerts through LEDs, buzzer notifications, and a live dashboard make waste management faster and more efficient. Overall, the project proves how IoT + AI can seriously upgrade basic waste bins into smart-city-ready systems that support cleaner and smarter environments.

Future Scope

- Add battery support to make the system fully wireless and energy-efficient.
- Improve AI to classify more waste types like plastic, paper, food, and recyclables.
- Integrate cloud-connected dashboard for smart-city level waste collection tracking.
- Enable automatic lid opening and robotic arm sorting for full automation.
- Add mobile app notifications and voice alerts for smart-home and industrial use.

REFERENCES

Trident3. (2022). *Waste Classification with CNN (with H5 model)* [Machine learning notebook]. Kaggle. <https://www.kaggle.com/code/trident3/waste-classification-with-cnn-with-h5-model>

Jain, P., Rane, K., & Vernekar, S. (2023). Classification of waste materials using convolutional neural networks. *ACM Transactions on Multimedia Computing, Communications, and Applications*. <https://doi.org/10.1145/3574318.3574345>

Naveen, S. A., Noor, M. A., & Fatima, N. (2021). Smart e-nose based food spoilage detection using MQ sensors. *International Journal of Food Microbiology*. <https://doi.org/10.1155/2021/9931228>

Mahajan, S. R., & Sangle, S. (2020). IoT-enabled waste management system using machine learning. *Journal of Sensors*. <https://doi.org/10.1155/2020/6138637>

Elmasry, M., & Hassan, M. (2023). Improving waste classification using CNN: A comparative study. *International Journal of Electrical and Computer Engineering*. <https://doi.org/10.18280/ria.370404>

DFRobot. (2020). *Gravity Capacitive Soil Moisture Sensor – Wiki documentation*. https://wiki.dfrobot.com/Capacitive_Soil_Moisture_Sensor_SKU_SEN0193

Hanwei Electronics. (2019). *MQ-135 Air Quality Sensor Datasheet*. <https://www.winsen-sensor.com/product/mq135.html>

APPENDICES

```

?-----.
192.168.55.40
🚀 Smart Bin System Started
📊 Sensors: MQ135, Moisture, Ultrasonic
🤖 AI: Integrated ML Waste Classification
🔴 RED LED: Wet Waste (Sensor + ML)
🟡 YELLOW LED: Bin Full

✓ Data sent to server
Gas:1337 Moist:4095 Dist:20.15 Fill:0 SensorWet:0 ML_Wet:0(0%) Full:0 Muted:0
Gas:1336 Moist:4095 Dist:20.23 Fill:0 SensorWet:0 ML_Wet:0(0%) Full:0 Muted:0
Gas:1331 Moist:4095 Dist:20.16 Fill:0 SensorWet:0 ML_Wet:0(0%) Full:0 Muted:0
Gas:1337 Moist:4095 Dist:20.16 Fill:0 SensorWet:0 ML_Wet:0(0%) Full:0 Muted:0

✓ Data sent to server
Gas:1337 Moist:4095 Dist:18.97 Fill:13 SensorWet:0 ML_Wet:0(0%) Full:0 Muted:0
Gas:1338 Moist:4095 Dist:18.95 Fill:13 SensorWet:0 ML_Wet:0(0%) Full:0 Muted:0
ML Status: Unknown (0%) - Wet: 0

✓ Data sent to server
Gas:1336 Moist:4095 Dist:20.57 Fill:0 SensorWet:0 ML_Wet:0(0%) Full:0 Muted:0
Gas:1336 Moist:4095 Dist:20.57 Fill:0 SensorWet:0 ML_Wet:0(0%) Full:0 Muted:0

```

Fig-1 Serial Monitor output of ESP32-S3 Board

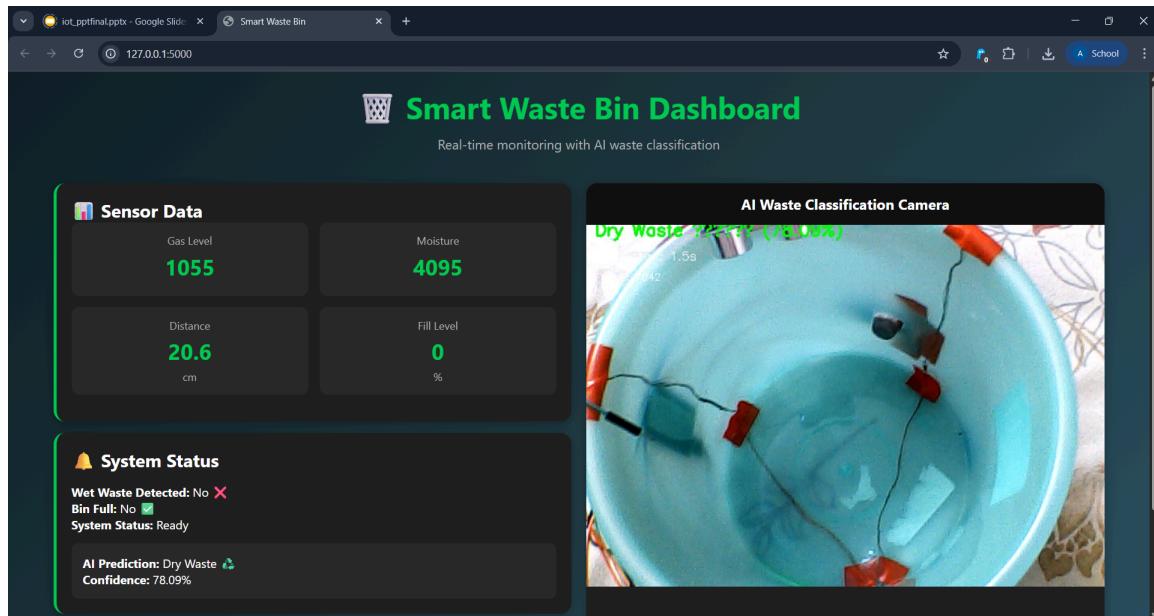


Fig-2 When the Smart Bin is empty

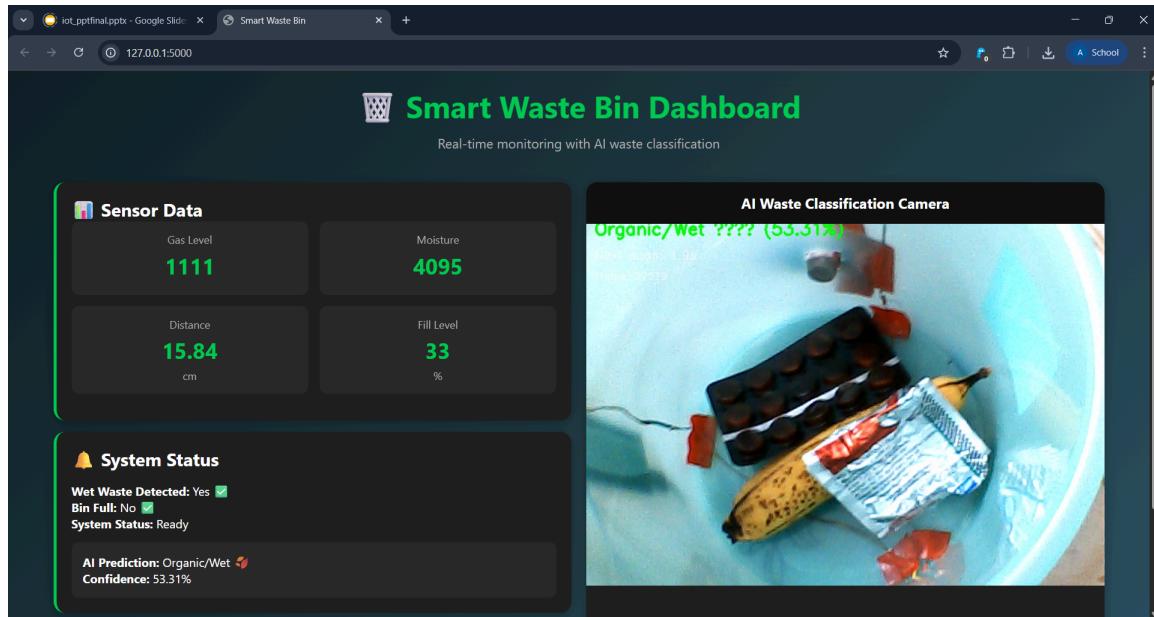


Fig-3 When the bin contains organic waste

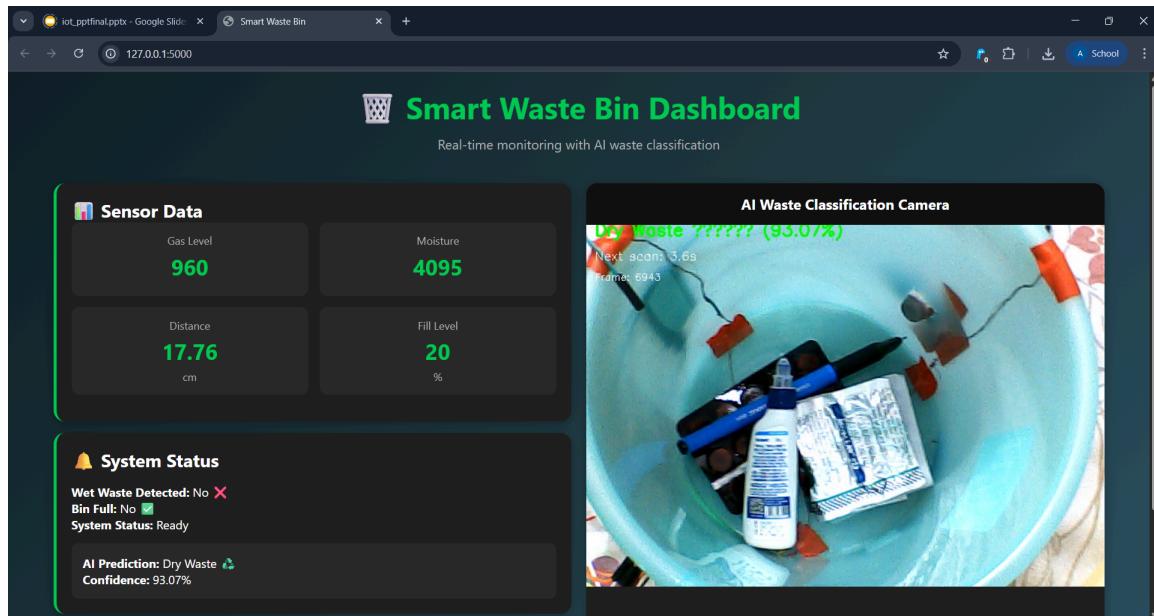


Fig-4 When the bin is filled with only dry waste