

**University of Science and Technology Chittagong (USTC)**

Faculty of Science, Engineering & Technology

Department of Computer Science & Engineering

**Lab Report - 6, 7 & 8**

**Course code: CSE 324**

**Course Title: Artificial Intelligence and Expert Systems Lab**

**Team Name - The Elite**

**Submitted by:**

Pritom Gupta

Unique ID: 0022210005101034

Roll No: 22010134

Reg No: 914

Semester: 6th

Batch: 38th

**Submitted to:**

Debabarata Mallick

[Lecturer]

FSET, USTC.

**Analysis of Arduino Code**

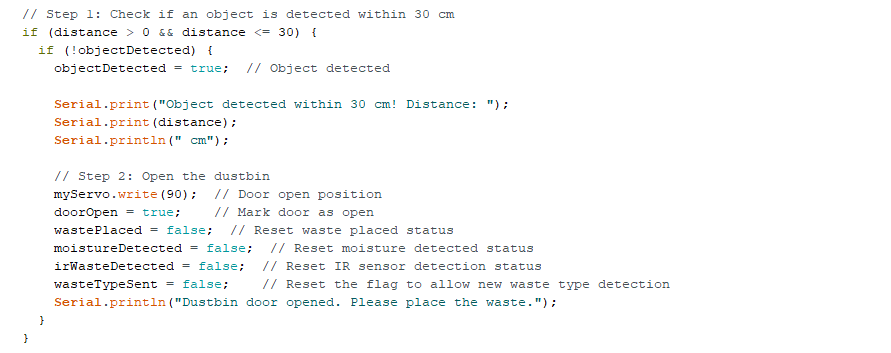
**Why Did We Use Arduino Code in our Project:**

The Arduino code is essential for automating and controlling the smart bin ai by integrating sensors (ultrasonic, IR, and soil moisture) and actuators like the servo motor and buzzer. It enables the dustbin to detect objects, identify waste types (organic or inorganic), and operate the lid autonomously. The code processes real-time sensor inputs, making decisions like opening the lid when waste is detected and closing it once the object is removed. It also communicates with a PC to send waste data for further analysis. Overall, the Arduino code makes the dustbin efficient, user-friendly, and capable of functioning without manual intervention.

**How It Works:**

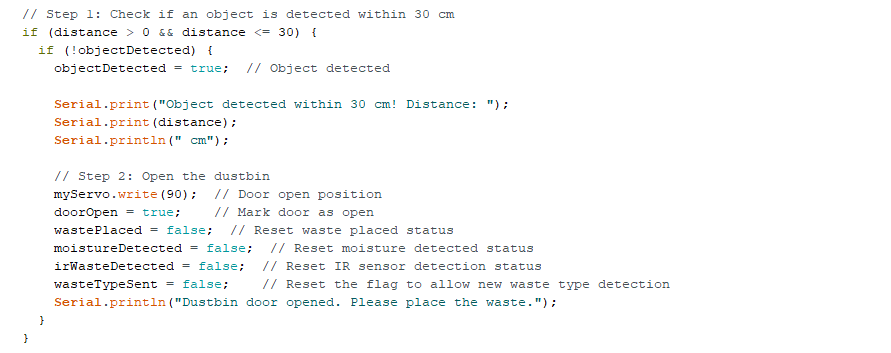
**STEP 1: Object Detection (Ultrasonic Sensor)**

The ultrasonic sensor measures the distance to objects in front of the dustbin. If an object is detected within 30 cm, the system considers that something is approaching the bin.



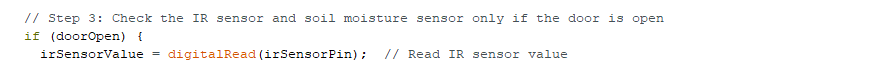
**Step 2: Opening the Dustbin (Servo Motor)**

Once the ultrasonic sensor detects an object, the servo motor moves the dustbin door to the open position (90°). This allows the user to place waste into the bin.



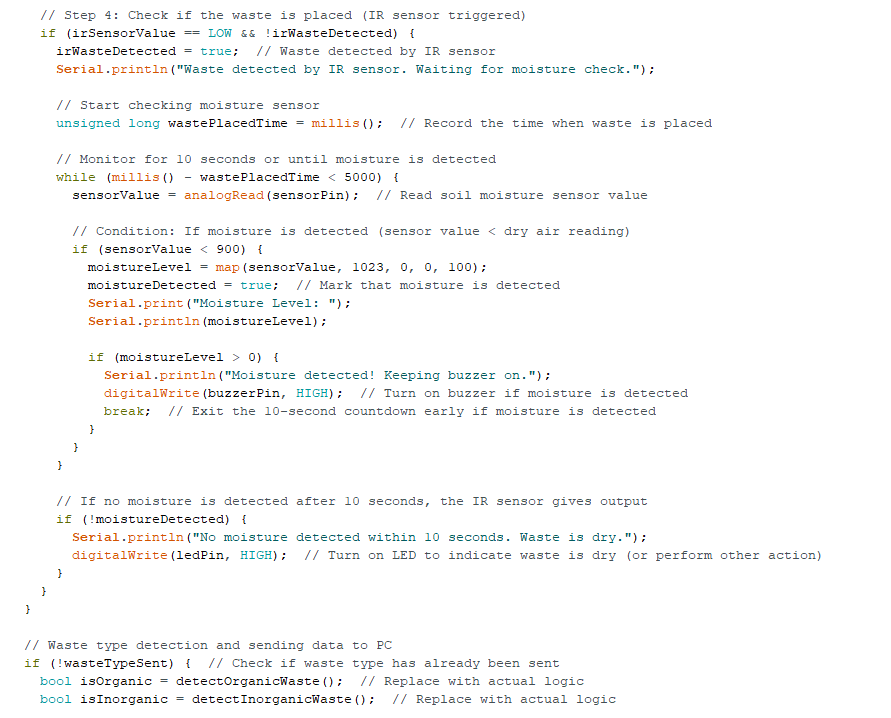
**Step 3: Waste Detection (IR Sensor and Soil Moisture Sensor)**

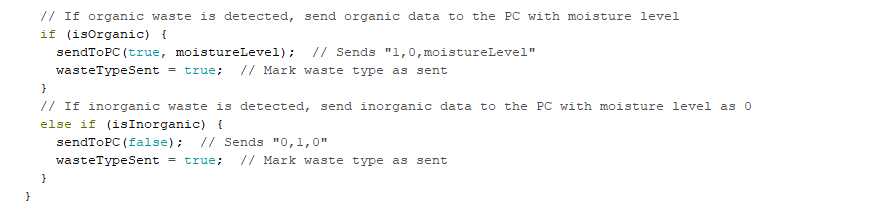
After the door is open, the IR sensor monitors for waste. Once waste is placed, it checks for the presence of moisture using the soil moisture sensor. If moisture is detected, the system assumes the waste is organic.



**Step 4: Moisture Detection and Buzzer/LED Alerts**

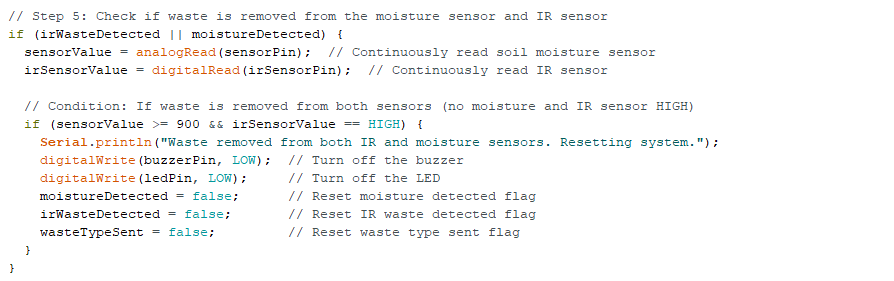
If moisture is detected, the system classifies the waste as organic, turns on the buzzer, and logs the moisture level. If no moisture is found, the waste is considered dry (inorganic), and the LED is turned on to indicate dry waste.





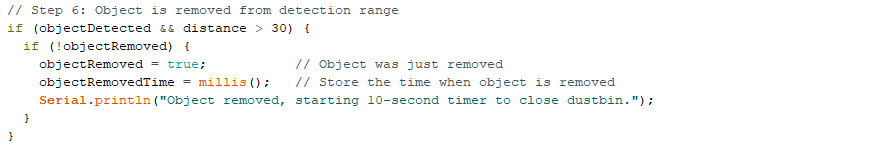
**Step 5: Waste Removal Detection (IR and Soil Moisture Sensor)**

After the waste is placed, the system continuously monitors both the IR and soil moisture sensors. If the waste is removed from the bin (IR sensor is high, and moisture sensor detects no moisture), the system resets the states.



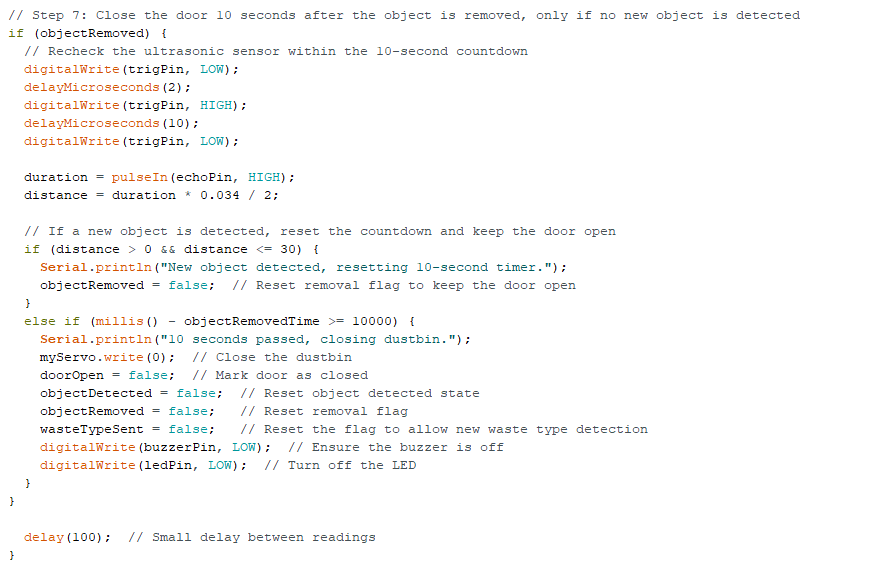
**Step 6: Object Removal from Detection Range**

After placing the waste, if the object (e.g., user's hand) is removed from the detection range of the ultrasonic sensor, a 10-second countdown begins to close the dustbin.



**Step 7: Closing the Dustbin (Servo Motor)**

After 10 seconds without any new objects being detected, the servo motor moves the door to the closed position (0°), sealing the bin. If a new object is detected within this time, the countdown resets to keep the door open.



**Implementation of Random Forest**

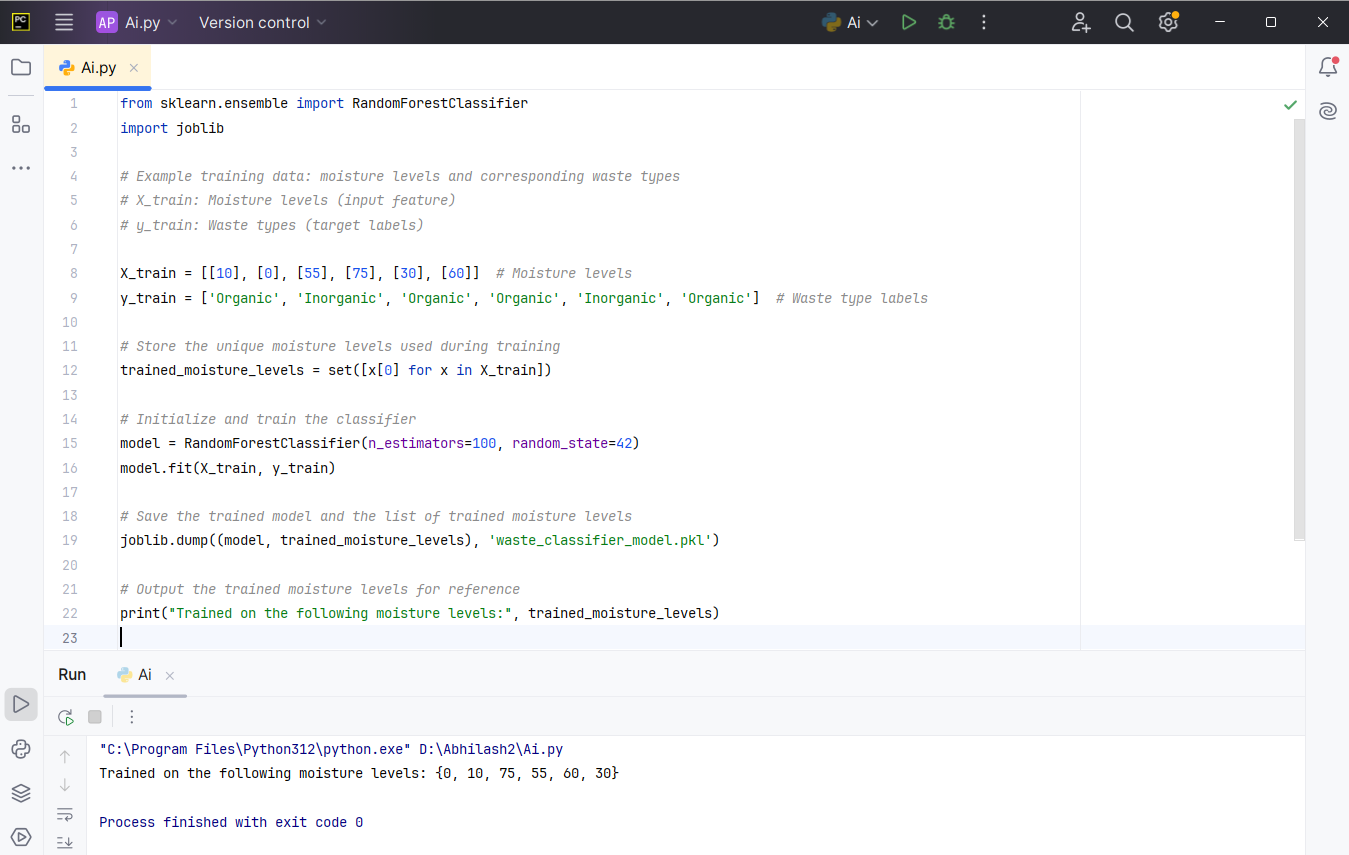
**Why Did We Use Random Forest in our Project**

* Random Forest is an ensemble machine learning algorithm that builds multiple decision trees during training.
* Each decision tree makes a prediction, and the majority vote (or average in regression tasks) across all trees is taken as the final output.
* For our project, the input is the moisture level of the waste, and the output is whether the waste is classified as organic or inorganic.

**How It Works**

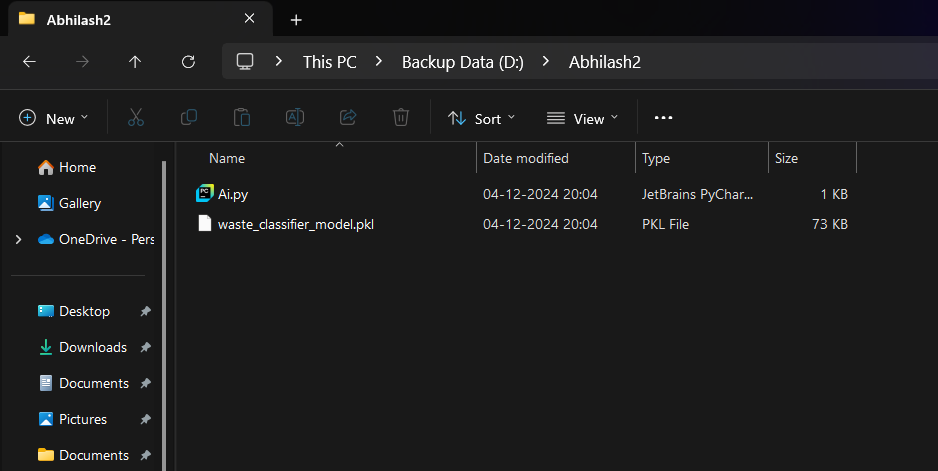
**Training the Model:**

* We have training data consisting of moisture levels and corresponding waste types. These values are used to train the model.
* The moisture levels (X\_train) are the input features, and the waste types (y\_train) are the target labels.
* A Random Forest Classifier is trained on this data to predict the waste type based on new moisture levels.



**Saving the Model:**

* After training, the model is saved using joblib so that it can be loaded and used later to classify new moisture levels in real-time as the Arduino provides them.



**Implementation of Machine Learning**

**Why Did We Use ML in our Project:**

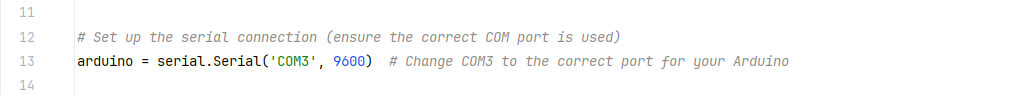
We use Machine Learning (ML) in our smart dustbin project to enhance the efficiency and accuracy of waste classification. By leveraging ML, the system can analyze data from various sensors, such as the soil moisture sensor and IR sensor, to identify patterns and classify waste more intelligently. An ML model can learn to distinguish between organic and inorganic waste based on sensor readings and other inputs, improving detection accuracy.

The implementation of ML allows the dustbin to adapt and make more informed decisions, enabling continuous improvement in waste identification. This helps in better sorting of materials, leading to more effective waste management, reduced environmental impact, and optimized recycling processes. By utilizing ML, our project moves toward a more advanced and responsive waste management system.

**How It Works:**

**Data Preprocessing:**

* The Arduino sends data in the format moisture\_level.
* This data is read via the serial connection and split into individual components.



* For our ML model, the primary input is likely the moisture\_level, which the model will use to predict whether the waste is organic or inorganic.

**Using the Pre-trained ML Model:**

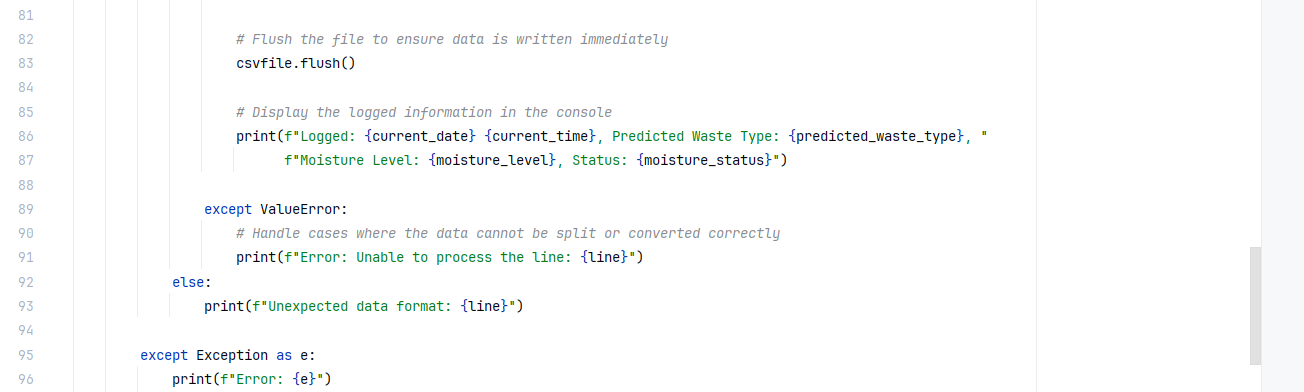
* You have loaded a pre-trained model (waste\_classifier\_model.pkl) using joblib.load. This model takes moisture levels as input and predicts the waste type (organic or inorganic).



* We will pass the moisture level to the model and let it predict the waste type.

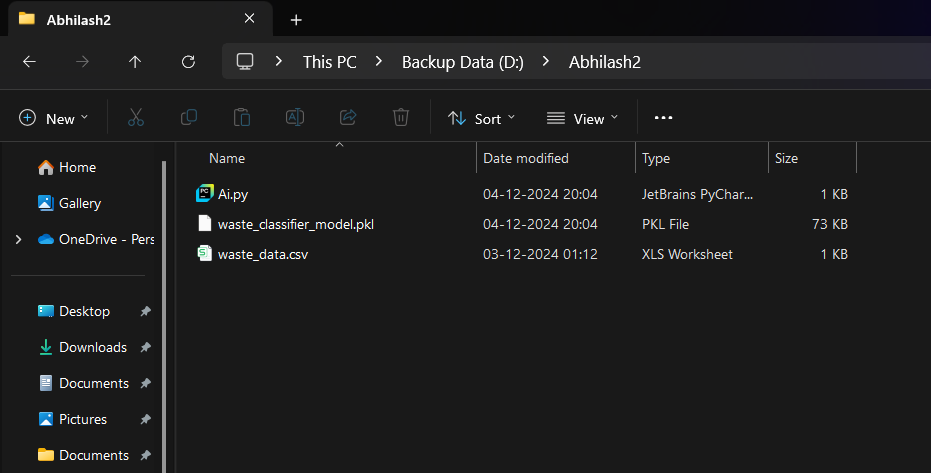




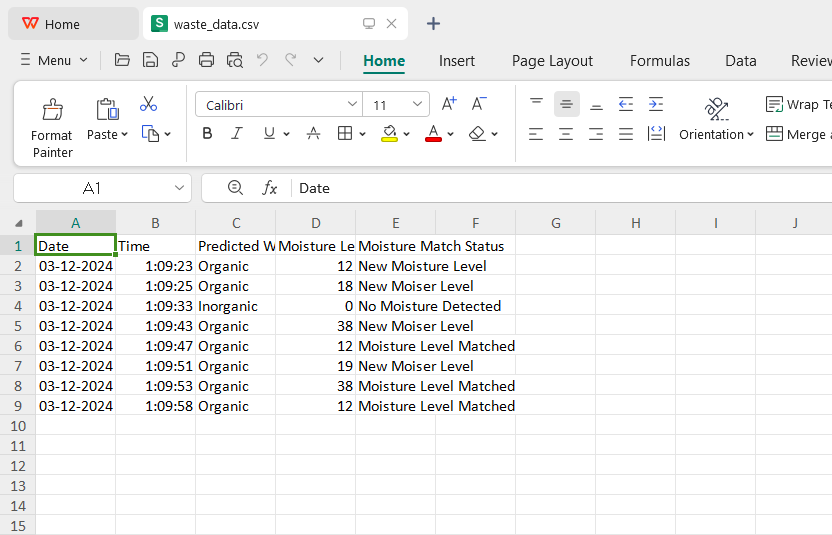


**Logging Data:**

* After predicting the waste type using the model, the predicted type, along with the moisture level, will be logged into a CSV file.



* The system also checks if the current moisture level matches any previously recorded levels (to track if the same waste type has been classified before).



**Github Link:** <https://github.com/P-r-i-t-o-m/Smart_Bin_AI>