

MID-TERM PROJECT

# AUTOMATIC DUSTBIN SYSTEM

BY:SAKHI SHARMA

Github Repository Link:

[https://github.com/Sakhi1011/Automatic-Dustbin/tree/8b64afe4692d44806225d62a518891ec2e578fb3/VIT\\_23BEE1174](https://github.com/Sakhi1011/Automatic-Dustbin/tree/8b64afe4692d44806225d62a518891ec2e578fb3/VIT_23BEE1174)

## Problem Statement

In many households and public spaces, traditional dustbins require manual operation, which poses hygiene issues, especially in shared environments. Users must touch the lid or foot pedal to dispose of waste, increasing the risk of germ transmission. There is a need for a **hands-free, hygienic waste disposal solution** that is low-cost, reliable, and easy to implement using basic electronics and automation.

---

## Scope of the Solution

The proposed solution is a **smart automatic dustbin** that uses an **ultrasonic sensor** to detect the presence of a user's hand or object near the lid. Upon detection, a **servo motor** opens the lid automatically and keeps it open for a few seconds before closing it, ensuring a **touchless and convenient disposal process**.

This project is ideal for use in homes, schools, hospitals, and offices, helping to **promote better hygiene and reduce cross-contamination risks**. The system is compact, cost-effective, and built using readily available hardware and open-source tools, making it scalable and adaptable to various bin sizes and environments.

---

## **Required Components to Develop the Solution**

### **Hardware Components**

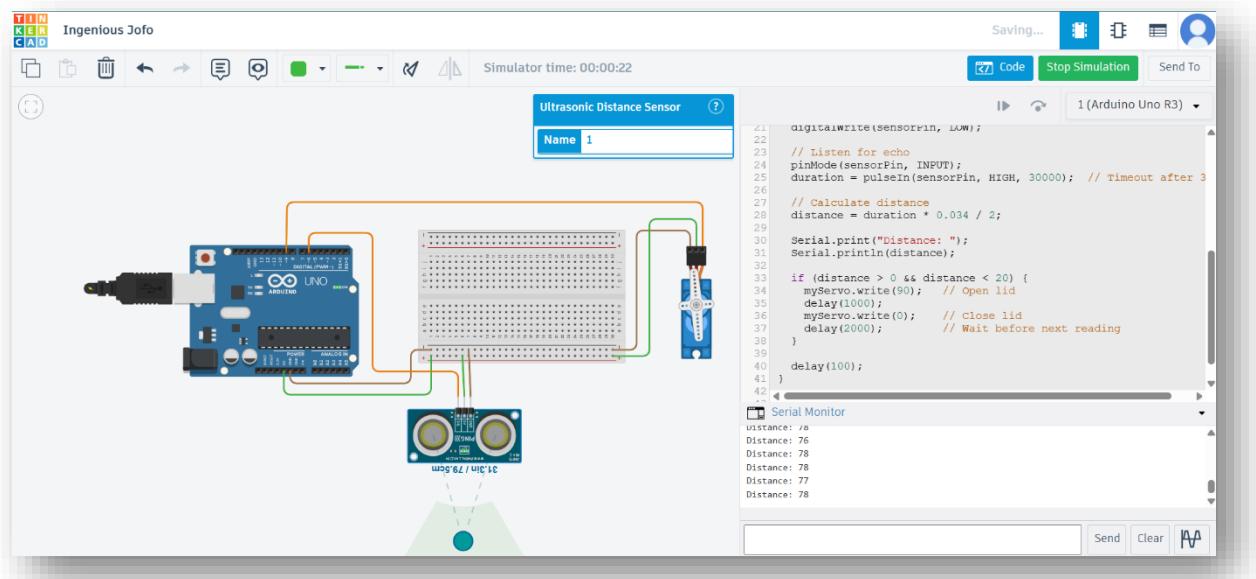
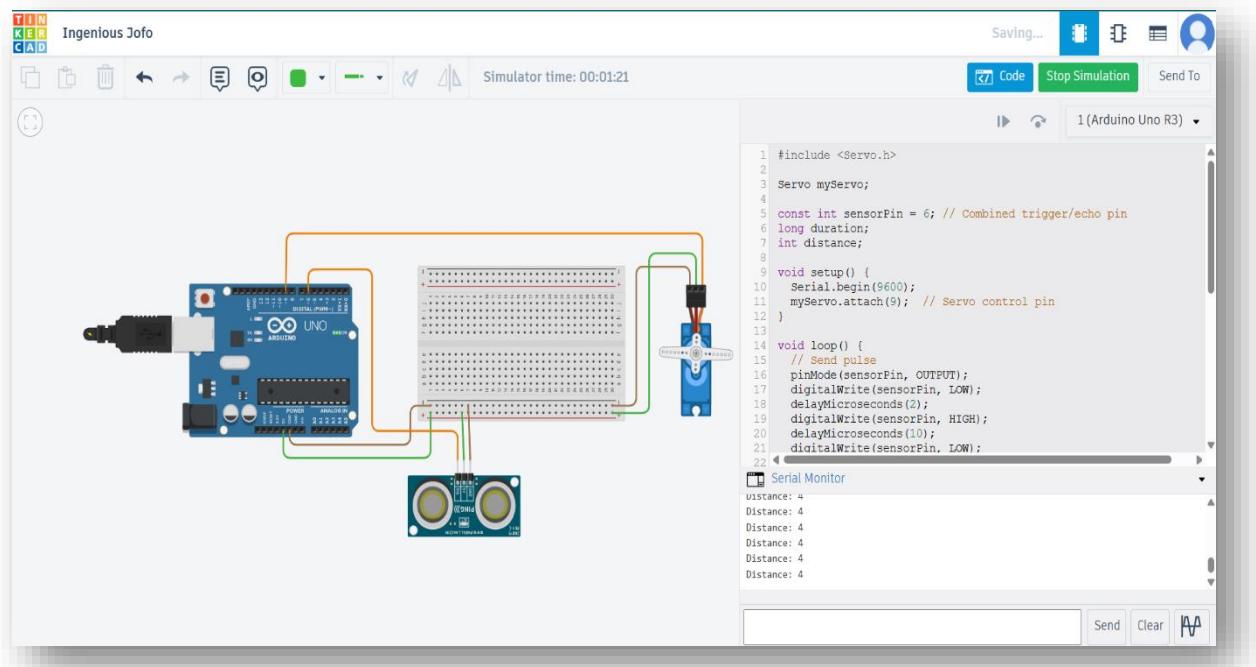
| <b>Component</b>                     | <b>Quantity</b> | <b>Description</b>                          |
|--------------------------------------|-----------------|---|
| Arduino Uno R3                       | 1               | Microcontroller board to control the system |
| HC-SR04 Ultrasonic Sensor            | 1               | For object distance detection               |
| MG90S Micro Servo Motor              | 1               | To open/close the dustbin lid               |
| Jumper Wires                         | As needed       | For electrical connections                  |
| Breadboard                           | 1               | For testing circuit layout                  |
| USB Cable (Type-B)                   | 1               | To upload code and power Arduino from PC    |
| Electrical tape                      | As needed       | For mounting the servo and mechanical setup |
| Plastic Dustbin with Pedal Mechanism | 1               | To convert into an automatic dustbin        |

---

## **Software Requirements**

| <b>Software / Tool</b>                 | <b>Purpose</b>                              |
|--|---|
| <b>Arduino IDE</b>                     | Writing and uploading code to Arduino board |
| <b>Tinkercad</b>                       | Simulate and test circuit virtually         |
| <b>Serial Monitor (in Arduino IDE)</b> | View angle/status messages during testing   |

# Simulated Circuit(in Tinkercad)



## Code for the solution

```
#include <Servo.h>

const int trigPin = 9;
const int echoPin = 10;
const int servoPin = 6;

Servo lidServo;

const int openDistance = 20;    // Distance threshold in cm
const int openAngle = 90;
const int closeAngle = 0;

bool isLidOpen = false;
int currentAngle = closeAngle;
unsigned long lastDetectionTime = 0;
const unsigned long lidOpenDuration = 3000; // Stay open for 5 seconds

void setup() {
  Serial.begin(9600);
  lidServo.attach(servoPin);
  lidServo.write(closeAngle);
  currentAngle = closeAngle;

  pinMode(trigPin, OUTPUT);
  pinMode(echoPin, INPUT);

  Serial.print("Initial Lid Angle: ");
  Serial.println(currentAngle);
}

void loop() {
  // Trigger ultrasonic sensor
  digitalWrite(trigPin, LOW);
  delayMicroseconds(2);
  digitalWrite(trigPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin, LOW);

  // Measure distance
  long duration = pulseIn(echoPin, HIGH, 30000); // 30 ms timeout
  int distance = duration * 0.034 / 2;

  unsigned long currentTime = millis();

  // If object is within range
  if (distance > 0 && distance < openDistance) {
    lastDetectionTime = currentTime;
```

```
if (!isLidOpen) {
    lidServo.write(openAngle);
    currentAngle = openAngle;
    isLidOpen = true;
    Serial.print("Lid opened. Angle: ");
    Serial.println(currentAngle);
}
}

// If enough time has passed since last detection, close the lid
if (isLidOpen && (currentTime - lastDetectionTime > lidOpenDuration)) {
    lidServo.write(closeAngle);
    currentAngle = closeAngle;
    isLidOpen = false;
    Serial.print("Lid closed. Angle: ");
    Serial.println(currentAngle);
}

delay(100);
}
```

## Demo Video

