♻️ Smart Waste Management System (IoT-Based)

This project is a Smart Waste Management System that uses sensors and communication modules to detect waste levels, classify waste as dry or wet, and send alerts via **SMS** and **email**. It also uploads real-time data to **ThingSpeak** for monitoring.

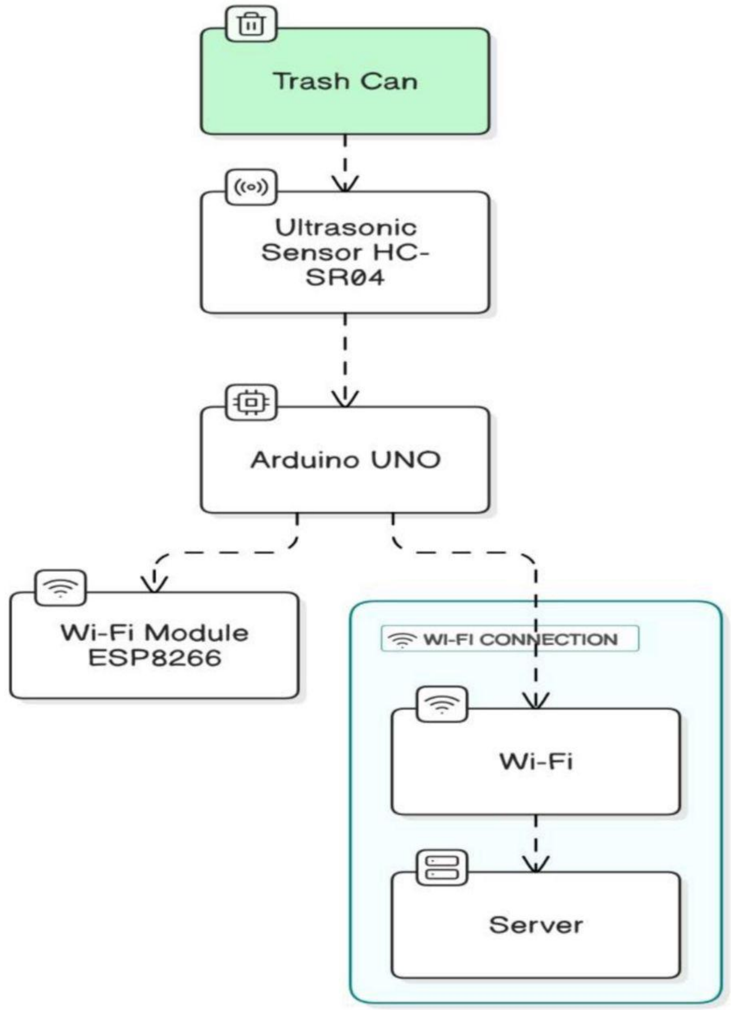
📦 Features

- Detects waste using an **IR sensor**
- Classifies waste using a **soil moisture sensor**
- Measures bin levels using **ultrasonic sensors**
- Controls a **servo motor** to direct waste to dry or wet sections
- Sends **SMS alerts** when bins are full using **SIM800L GSM**
- Sends **email notifications** via **ESP8266 and Gmail SMTP**
- Uploads data (moisture, dry/wet bin level) to **ThingSpeak**

🛠️ Hardware Components

Components	Quantity
Arduino Uno	1
ESP8266 NodeMCU	1
Servo Motor	1
IR Sensor	1
Soil Moisture Sensor	1
Ultrasonic Sensor (HC-SR04)	2
SIM800L GSM Module	1
Jumper Wires	
Breadboard/PCB	1
Power Supply (5V)	1

🧠System Architecture



📁 Code Structure

1. `arduino_dustbin.ino`

This file runs on **Arduino UNO** and does the following:

- Reads sensor values (IR, soil moisture, ultrasonic)

- Controls the servo motor to classify waste

- Sends waste level & moisture data to ESP8266 via Serial

- Sends SMS using SIM800L when a bin is full

2. `esp8266_monitoring.ino`

This file runs on **ESP8266 NodeMCU** and:

- Connects to WiFi

- Reads serial data from Arduino

- Uploads data to ThingSpeak

- Sends email alert via Gmail SMTP when bin is full

🌐 ThingSpeak Setup

1. Create a ThingSpeak account: https://thingspeak.com

2. Create a new channel and enable 3 fields:

   - Field 1: Soil Moisture

   - Field 2: Dry Bin Level

   - Field 3: Wet Bin Level

3. Note your **Channel ID** and **Write API Key** and update them in the ESP8266 code.

📧 Email Setup (Gmail SMTP)

> Gmail requires an **App Password** (not your Gmail password)

1. Enable 2FA on your Gmail account

2. Go to [App Passwords](https://myaccount.google.com/apppasswords)

3. Generate a new password and replace it in:

```cpp
```

```
#define AUTHOR_PASSWORD "your_app_password_here"
```

SMS Configuration (SIM800L)

```
sim800l.println("AT+CMGS=\"+91XXXXXXXXXX\"");
```

Dependencies

Install these Arduino libraries:

- Servo
- SoftwareSerial
- ESP8266WiFi
- ThingSpeak
- ESP_Mail_Client

🔧 Arduino IDE Setup

1. **Download Arduino IDE**:

   https://www.arduino.cc/en/software

2. **Install Required Libraries**:

   Go to `Sketch > Include Library > Manage Libraries` and install:

   - `Servo`

   - `SoftwareSerial`

   - `ThingSpeak`

   - `ESP Mail Client` (by Mobizt)

3. **Add ESP8266 Board Support**:

   - Go to `File > Preferences`

   - In **Additional Board Manager URLs**, paste:

     ```
     http://arduino.esp8266.com/stable/package_esp8266com_index.json
     ```

- Then go to `Tools > Board > Boards Manager`, search **ESP8266**, and install.

4. **Select Boards**:

- For Arduino UNO: `Tools > Board > Arduino Uno`

- For NodeMCU/ESP8266: `Tools > Board > NodeMCU 1.0 (ESP-12E Module)` *(or whichever matches your ESP)*
