SOFTWARE CODE

1.1 Arduino UNO Code (Smart Bin Node)

```
#include <SPI.h>
#include <LoRa.h>

#define TRIG 5
#define ECHO 6

long duration;
int distance;
int fillLevel;

void setup() {
  Serial.begin(9600);

  pinMode(TRIG, OUTPUT);
  pinMode(ECHO, INPUT);

  LoRa.setPins(10, 9, 2); // NSS, RESET, DIO0

  if (!LoRa.begin(433E6)) {
    Serial.println("LoRa Init Failed");
    while (1);
  }
}

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

  duration = pulseIn(ECHO, HIGH);
  distance = duration * 0.034 / 2;

  // Map distance to fill level
```

```
  fillLevel = map(distance, 30, 5, 0, 100);
  fillLevel = constrain(fillLevel, 0, 100);

  // Send via LoRa
  LoRa.beginPacket();
  LoRa.print(fillLevel);
  LoRa.endPacket();

  Serial.print("Bin Fill Level: ");
  Serial.print(fillLevel);
  Serial.println("%");

  delay(30000); // Send every 30 sec
}
```

1.2 ESP32 Code (Gateway Node)

```
#include <SPI.h>
#include <LoRa.h>

void setup() {
  Serial.begin(9600);

  LoRa.setPins(18, 14, 26); // NSS, RESET, DIO0

  if (!LoRa.begin(433E6)) {
    Serial.println("LoRa Init Failed");
    while (1);
  }
}

void loop() {
  int packetSize = LoRa.parsePacket();

  if (packetSize) {
    int fillLevel = 0;

    while (LoRa.available()) {
      fillLevel = LoRa.parseInt();
    }
```

```
    Serial.print("Received Fill Level: ");
    Serial.print(fillLevel);
    Serial.println("%");

    if (fillLevel >= 80) {
      Serial.println("⚠ ALERT: Bin Almost Full!");
    }
  }
}
```