SOFTWARE ARCHITECTURE

1.1 Software Architecture Overview

The software follows a **layered IoT architecture**:

```
Application Layer
(Dashboard, Alerts, Routing)

Cloud / Gateway Layer
(ESP32 + Processing)

Communication Layer
(LoRa SX1278)

Edge Layer
(Arduino + Sensors)
```

1.2 Software Responsibilities

Edge Layer (Arduino UNO)

- Sensor reading
- Distance → Fill % conversion
- Packet formation
- Low-frequency transmission

Gateway Layer (ESP32)

- Packet reception
- Threshold evaluation
- Alert generation
- Cloud forwarding (conceptual)

1.3 Data Flow (Software Perspective)

```
Sensor Read → Local Processing → LoRa Tx
→ LoRa Rx → Alert Logic → Dashboard Update
```

ALGORITHMS USED

2.1 Fill Level Detection Algorithm

Logic

- Empty bin → larger distance
- Filled bin → smaller distance

```
fill_percentage =
(1 - (measured_distance / bin_height)) × 100
```

Mapping used for simplicity.

2.2 Alert Algorithm

```
IF fill_level ≥ 80%
    Trigger ALERT
ELSE
    Store data normally
```

2.3 Transmission Control Algorithm

```
Read Sensor
Process Data
Send Packet
Sleep for fixed interval
```

FLOW CHART

```
Start
 |
Initialize Hardware
 |
Read Ultrasonic Sensor
 |
Calculate Fill Level
 |
Is Fill ≥ 80%?
 |       |
Yes      No
 |       |
Send Alert  Normal Data
 |
Transmit via LoRa
 |
Gateway Receives
 |
Display / Upload
 |
End
```