**Step 2: Organise and Describe the Data**

**Inputs (sensors and configuration)**

| **Name** | **Type** | **Source** | **Unit / Format** | **Sample Values** | **Valid Range / Constraints** | **Update Freq.** | **Notes** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| current\_time | time | RTC module | 24-hr HH:MM | 08:00, 18:00 | Must match any scheduled time to trigger feed | 1s–60s | System clock assumed accurate |
| feeding\_schedule | list | User config | Array of HH:MM | [08:00, 18:00] | 1–6 entries/day; unique; 24-hr format | On change / boot | Editable by staff |
| portion\_size\_g | integer | User config | grams | 120 | 20–400 g per feed | On change / per feed | Depends on pet/diet |
| food\_level\_state | enum | Hopper level sensor | `FULL | OK | LOW | EMPTY` | OK |
| hopper\_weight\_g (optional) | float | Load cell | grams | 2500 | 0–5000 g; ±5 g acc. | 5–30s | Alternative to discrete level |
| bowl\_weight\_g\_before | float | Bowl load cell | grams | 50 | 0–1000 g; ±5 g acc. | At dispense | Used to confirm drop |
| bowl\_weight\_g\_after | float | Bowl load cell | grams | 170 | 0–1000 g; ±5 g acc. | After wait window | Used to detect eating |
| door\_open (optional) | boolean | Lid/door switch | true/false | false | Must be false to dispense | 1–5s | Safety interlock |
| manual\_feed\_button | boolean | User input | press event | false→true | Debounced; lockout 2–5 min | On event | Optional manual override |
| network\_ok (optional) | boolean | System | true/false | true | Needed for cloud/SMS alerts | 10–60s | Fallback to buzzer if false |
| power\_ok (optional) | boolean | System | true/false | true | Battery >20% if mains off | 10–60s | For reliability logging |

**Outputs (actuators and notifications)**

| **Name** | **Type** | **Target** | **Unit / Format** | **Sample Commands** | **Constraints** | **Trigger Conditions** |
| --- | --- | --- | --- | --- | --- | --- |
| servo\_dispense | actuator | Dispenser | rotation/time | rotate(120°) x N or run(3s) | Max 10s per cycle; jam detect | At scheduled time & food available |
| status\_led | indicator | LED | pattern | GREEN steady, RED blink | Non-blocking | State feedback (OK/ERROR) |
| buzzer | indicator | Piezo | pattern | beep(500ms) x3 | ≤60 dB (shelter rule) | Error or empty hopper |
| alert\_message | notification | App/SMS/email | text | "Food not eaten in 10m" | Network required | Any error/exception |
| log\_event | record | Local/Cloud | JSON | see schema below | Persistent storage | Every significant state change |

Recommended alert\_message reasons: HOPPER\_EMPTY, DISPENSE\_FAILURE, NOT\_EATEN, SENSOR\_FAULT, DOOR\_OPEN, LOW\_FOOD.

**Operational parameters and thresholds**

| **Parameter** | **Purpose** | **Default** | **Allowed Range** | **Impact / Rationale** |
| --- | --- | --- | --- | --- |
| dispense\_cycle\_g | Grams per servo cycle | 30 g | 10–80 g | Calibrate per hardware; used to compute cycles = ceil(portion\_size\_g / dispense\_cycle\_g) |
| max\_dispense\_cycles | Fail-safe limit | 6 | 1–10 | Prevents endless running/jams |
| consumption\_wait\_min | Wait time before “eaten?” check | 10 min | 5–20 min | Enough time for pet to eat |
| consumed\_delta\_g | Drop in bowl weight to count as “eaten” | 20% of portion (min 15 g) | 10–50% | Avoids false positives due to evaporation/spillage |
| low\_food\_threshold\_% | Hopper level considered LOW | 20% | 10–30% | Early warning before EMPTY |
| jam\_detect\_timeout\_s | Motor run max before jam alert | 8 s | 3–12 s | Protects motor & gears |
| lockout\_after\_manual\_min | Cool-down after manual feed | 3 min | 1–10 min | Prevents overfeeding |
| time\_sync\_drift\_max\_s | Max RTC drift before resync | 30 s | 10–120 s | Keeps schedule accurate |

**Data validation rules (quick checklist)**

* Time format: HH:MM 24-hr; no duplicates; within 00:00-23:59.
* Portion: integer gramsl within 20-400 g.
* Sensor sanity: bowl\_weight\_g\_after >= bowl\_weight\_g\_before immediately after dispense; if not, flag DISPENSE\_FAILURE.
* Level/weight coherence: if hopper\_weight\_g < portion\_size\_g or food\_level\_state == EMPTY ⇒ block dispense + alert.
* Interlocks: door\_open == false to permit motor.
* Retry: At most one auto-retry for dispense before alerting jam/failure.

**Example values in a typical feed**

* Schedule: [08:00, 18:00]
* Portion: 120 g
* Calibrated cycle: 30 g/cycle ⇒ 4 cycles
* Before dispense: bowl\_weight\_g\_before = 50 g
* After dispense (immediate): ~170 g(≈ +120 g)
* After wait 10 min: ~90 g (drop ≈ 80 g ≥ threshold 24 g) ⇒ count as eaten, no alert.

**Log/event schema (recommended)**

{

"ts": "2025-08-17T08:00:02+05:30",

"event": "FEED\_ATTEMPT",

"schedule\_time": "08:00",

"portion\_g": 120,

"cycles": 4,

"pre\_bowl\_g": 50,

"post\_bowl\_g": 170,

"hopper\_g": 2500,

"level": "OK",

"result": "SUCCESS",

"alerts": []

}