**INTRODUCTION TO INFORMATION TECHNOLOGY 8936**  
assessment 1: The Problem solving Process (pSP)  
DUE DATE: 17 August 2025

# **i BACKGROUND** Animals in shelters must be fed appropriate food types and portion sizes at regular intervals, typically twice daily, to maintain their health and wellbeing. Currently, shelter staff, who are often volunteers, manage all feeding activities manually. This creates the following issues:

**Animal Welfare Concerns:**

* Missed meals or incorrect portions can harm animal health
* Different breeds require specific diets that are difficult to track without systems
* Inconsistent feeding times cause stress and health problems

**Operational Problems:**

* Staff have no reliable way to track which animals have been fed/underfed or overfed
* Limited systems exist to record feeding history or dietary requirements
* Food inventory planning is difficult without feeding data
* Manual processes consume staff time

**Record-Keeping Issues:**

* Poor documentation of feeding schedules and portions
* Inability to quickly access animal feeding histories
* Inadequate records for shelter operations and compliance

# **ii PSP STEP 1. Understand and Define the Problem**

## **Animal Shelter Pet Feeding Problem Statement**

The current manual approach is time-intensive, error-prone, and prevents staff from focusing on other essential animal care activities. Staff need a low-cost, automated system that ensures the following core requirements:

1. The correct animal receives the right food type and portion size
2. Feeding occurs at the appropriate times
3. Comprehensive records are maintained
4. Food inventory can be planned effectively
5. Staff time is used efficiently

## **Automated Pet Feeder Functional Requirements**

### Feeding Schedule: Define how often and how much food will be dispensed.

### Portion Control: Determine how to measure and dispense specific amounts of food.

### User Notifications: Implement alerts for low food levels or feeding status.

### Pet Safety: the feeder system is safe for pets.

The following data inputs, outputs and considerations were identified:

**1. Profile of pet and requirements**

* **Type of pet: (e.g., dog, cat etc.)**
* **Size, age of pet: (small, medium, large)**
* **Feeding schedule: times per day, portion size**
* **Food type: dry, wet food, pellets.**

**2. Feeder Mechanism & Operation**

* **Dispensing method: gravity, motorised unit, rotary dispenser**
* **Portion control: fixed or adjustable servings.**
* **Food storage capacity: volume needed between refills**
* **Power source: battery, USB, electricity, solar power.**

**3. Control System**

* **Automation level: timer-based, sensor-based, remote-controlled**
* **Sensors needed: scale/weight sensor, RFID sensor, food level detector**
* **User interface: buttons, LED display.**

**4. Manufacturing Assumptions and Cost Constraints**

* **Food-safe container: ensure materials are safe for pet food**
* **Manufacturing process: injection molding, 3D printing, assembly**
* **Estimated materials cost and availability of components**
* **Estimated maintenance ease and cost**

**6. Safety & Reliability Assumptions**

* **Pet safety: avoid sharp edges, toxic materials, contamination**
* **Food freshness: airtight storage, moisture control**
* **Prevent jamming: mechanism to prevent food blockage**

**7. Maintenance & Usability**

* **Ease of cleaning: removable parts, dishwasher-safe**
* **Refill frequency: how often staff need to add food**
* **Setup simplicity: easy use, minimal assembly**

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

**A. Limitations and assumptions:**

* **Different feeder required for each animal type**
* **Physical space for feeders required**
* **Tech costs or skills to maintain system**
* **Ease of use for staff to programme**
* **Dry food dispensing only \*Adult and baby options but excludes special diets**

**B. Sketch of system****

### **C. Automated Pet Feeder System Input and Output Variables Table**

|  |  |  |  |
| --- | --- | --- | --- |
| **Variables** | **Type** | **Value** | **Unit** |
| **INPUTS** |  |  |  |
| Animal type | Discrete Boolean | dog = 0cat = 1 | - |
| Pet ID | Discrete Boolean | True = 1False = 0 | - |
| Pet age | Discrete Boolean | Pup= 0 Adult = 1 | - |
| Time: current | continuous | 0-24hrs | Date time |
| Feeding schedule | discrete Boolean | 08:00AM 18:00PM | - |
| Portion quantity | discrete Boolean | 200 or 500 | grams |
| food level of container | discrete Boolean | True=1False= 0 | - |
| bowl ready (empty) | discrete Boolean | True=1 False= 0 | - |
| Manual feeding | discrete Boolean | ON=1 NOT=0 | - |
| Motor controller | discrete Boolean | ON=1 OFF=0 | - |
| **OUTPUTS** |  |  |  |
| Dispensed (servo) | discrete Boolean | Dispensing=1 idle=0 | Alert LED |
| Low food Alert | discrete Boolean | Alert= 1 true/false=0 | Buzzer |
| Empty Bowl Alert | discrete Boolean | Empty/true=1 Filled/False=0 | LED |
| Feeding complete Alert | discrete Boolean | Complete=1 Dispensing=0 | LED |
| System status Alert | discrete Boolean | ON/OFF | Buzzer |
| **CONSTANTS** |  |  |  |
| Energy supply | - | ON/OFF |  |
| Pet presence and detection | - | True/False |  |
| food level requirement | - | >80% |  |
| Check level frequency | - | prior to/ after feeding |  |

## **Data Validation Features:**

### Input checking for all sensor LED

### Timeout protections for all operations

### RFID validation against registered pets

### **Step 3: Plan the Solution (Design the Algorithm)**

### **A. Logical sequence of steps:** 1. Ensure system is on 2. Identify feeder container is full 3. Confirm time

### 4. Confirm pet ID 7. Dispense food 8. Confirm if food eaten (additional condition after dispensing)

## **Each step would typically be implemented in a program: with each step having true/false binary pathways. Power Check → Container Check → Time Check → Pet Check → Bowl Check →Feed Check → Dispense → Verify.**

## **B. Steps by Decision Gates:** **1. Is system On? 2. Is Feeder full? (F)** Gate: Decision No → Alert/refill Yes → Proceed to the next step **3. Is pet identified? (P)** Gate: Decision No → Wait Yes →Proceed to the next step **4. Is it the scheduled time? (F)** Gate: Decision No→ wait Yes→ proceed to the next step **5. Dispense food (D)** Action: Dispense **6. Verify food consumed? (C)** GATE: decision Yes → log pet as fed No → Wait/alert

## **C. Table: Boolean Conditions and Actions**

|  |  |  |  |
| --- | --- | --- | --- |
| **Step** | **Boolean Value** | **Action if True** | **Action if False** |
| System ON | **True/false** | **Continue** | **Turn On** |
| Food check | Container full  true/false | Continue | Alert to refill |
| Pet ID check | Matched  true/false | Continue | Stop  Alert manual match |
| Time check | Scheduled time confirmedtrue/false | Dispense | Stop  Wait PM feed |
| Dispensing | **Dispensing true/false** | **Motor on dispense portion** | **Alert manual dispense** |
| Food consumed | **Plate empty  True/false** | **Log and end** | **Alert** |

### S= System on F= Food level P = Pet ID T = Time Checked D= Dispensing C= Food consumed

## **D. Truth Table 1= True 0=False**

|  |  |  |  |
| --- | --- | --- | --- |
| **F** | **P** | **T** | **D** |
| 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 |
| 0 | 1 | 0 | 0 |
| 0 | 1 | 1 | 0 |
| 1 | 0 | 0 | 0 |
| 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 0 |
| 1 | 1 | 1 | 1 |

### Expression D= F AND P AND T

### **Truth table Step 6. Confirm fed (C)**

|  |  |  |
| --- | --- | --- |
| **D** | **B** | **C** |
| 0 | 1 | 0 |
| 1 | 0 | 1 |
|  |  |  |
|  |  | 1 |

### Expression C= D AND B

### **FLOWCHART**

### A group of rectangular shapes with text AI-generated content may be incorrect.

### **Step 4: Implement the Solution (Word Coding)**

### **Decision Logic conditions and MODULAR sequence**

1. **Feeder is Full**
   * feederFull = true/false
   * If *feederFull* == false, **block dispensing** and trigger refill alert.
2. **Correct Time is Scheduled** 
   * if scheduledFeedingTime == true: proceed
   * else: dispensing = false
3. **Pet is Identified Electronically Pet Portion Matches Size, Age,**
   * petIdentified = true/false
   * If *petIdentified* == false, **do not proceed.**
4. **Feeder Releases Gate and Dispenses Food**
   * Action occurs if all above are true.
5. **Confirm If Pet Has Eaten Food** 
   * foodEaten = true/false (detected by bowl weight change)
   * If foodEaten == true, update *petHasEaten* to true for the next cycle.

6. Post confirmation  
After feed confirmed, check if food storage is empty (if *foodStorageEmpty* == true: *foodAvailable* = false, then signal refill). May also add option to **Repeat Cycle** Loop or timer resets the “*scheduled Feeding Time”* for the next interval.

### **Step 5: Test and Refine the Solution (Debug and Verify)**

## Test planning for software is designed to manage risk prevent failures, improve quality and assist documentation – it outlines what will be verified and validated and the scope of what and when will be tested. The Test Plan will review logic with the following scenarios and refine solution where necessary: Test objectives:

-Pet is identified to portion  
-Pet is not identified to portion  
-Pet eats as expected  
-Pet does not eat  
-Food bin is empty  
  
-Test with container full and empty  
-Test with sensors disconnected   
-Test LED lamps  
-ID scan and weight sensor

## **Table: Test Plan**

| **Scenario** | **Steps to Reproduce** | **Expected Output** | **Actual Output** | **Pass/Fail** |
| --- | --- | --- | --- | --- |
| **Normal Operation** | Ensure food bin is >10%. Wait for a scheduled time.  Scan pet's RFID tag. Remove food from the bowl after dispensing. | Feeder dispenses correct portion.  System logs a successful feeding.  Status LED is green. |  |  |
| **Unidentified Pet Feed** | Ensure food bin is >10%.  At feeding time, scan an incomplete RFID tag. | Feeder does **not** dispense. System sends an "Incomplete Pet ID" alert.  Status LED is yellow/orange. |  |  |
| **Pet Does Not Eat** | Perform steps for 1. Do **not** remove food from the bowl. Wait for the consumption timeout to expire. | System sends a "Pet Has Not Eaten alert. System logs the feeding as "Not Eaten". |  |  |
| **Food container empty** | Ensure food bin is <10% full.  Wait for the scheduled feeding time. | Feeder does **not** dispense. System sends a "Refill Food" alert.  Refill LED is red. |  |  |
| **Timeout - No Pet** | Ensure food bin is >10%. At feeding time, do **not** scan any RFID tag.  Wait for the pet detection timeout. | Feeder does **not** dispense. System returns to idle/wait state.  No alerts are sent. |  |  |
| **Failure - Jam** | Manually block the dispenser mechanism. Trigger a feeding event | Motor attempts to run. Motor stops after a short timeout System sends a "Feeder Jammed" alert. |  |  |
| **Failure - RFID** | Disconnect the RFID reader. Attempt to trigger a feeding event. | System cannot detect any pet.  *\**System sends a "Hardware Error: RFID Reader" alert. |  |  |

## Test Cases\*

## For each requirement, business process, or system feature to be tested, the tester will execute a

## set of pre-defined test cases. Each test case will have a series of actions and expected results. As

## each action is performed, the results are evaluated. If the observed results are equal to the

## expected results, a checkmark is placed in the “pass” column. If the observed results are not equal

## to the expected results, a checkmark is placed in the “fail” column. There are three main methods of testing to confirm the primary function and establish the baseline for successful outcomes are:

## Positive Path Testing (“the happy path”): Test 1: Normal Feeding Cycle Scenario: All systems working, scheduled time, pet identified, sufficient food

#### Negative Testing Scenarios: Feeder is empty Pet is not identified Wrong time for feeding Pet does not eat portion

#### Failure Testing Scenarios to identify whether, if the path fails, it can do so safely. Power failure ID sensor fails feeder weight sensor fails feeding schedule timer fails LED and Buzzer Alert testing feeder jam Motor failure

#### User Testing Scenario: employee can’t operate system

**iii AI AGENT INTEGRATION: Co-pilot GPT as co-editor**

Using Microsoft Copilot GenAI to assist in solving this problem, refining the logic and enhancing documentation produced the following:   
  
RESPONSE 1: identifying the Automated Pet Feeder Hardware options

### **Hardware**

### Automated Feeding Mechanism: Decide on the type of mechanism (gravity-fed, motorized, etc.) for dispensing food.

### Food Storage: Determine the size and material of the food container to ensure durability and safety.

### Sensors: Consider using sensors for detecting food levels, pet presence, or even weight.

### Power Source: Decide whether the feeder will be battery-operated, plug-in, or solar-powered.

### **Software/Circuits** Control System: Develop a microcontroller or microprocessor-based system (like Arduino or Raspberry Pi) to manage feeding schedules and amounts. User Interface: Create a mobile app or web interface for users to set feeding times and portions. Connectivity: Decide on Wi-Fi, Bluetooth, or other communication methods for remote control and monitoring- method to identify animal breed and instructions RFID example

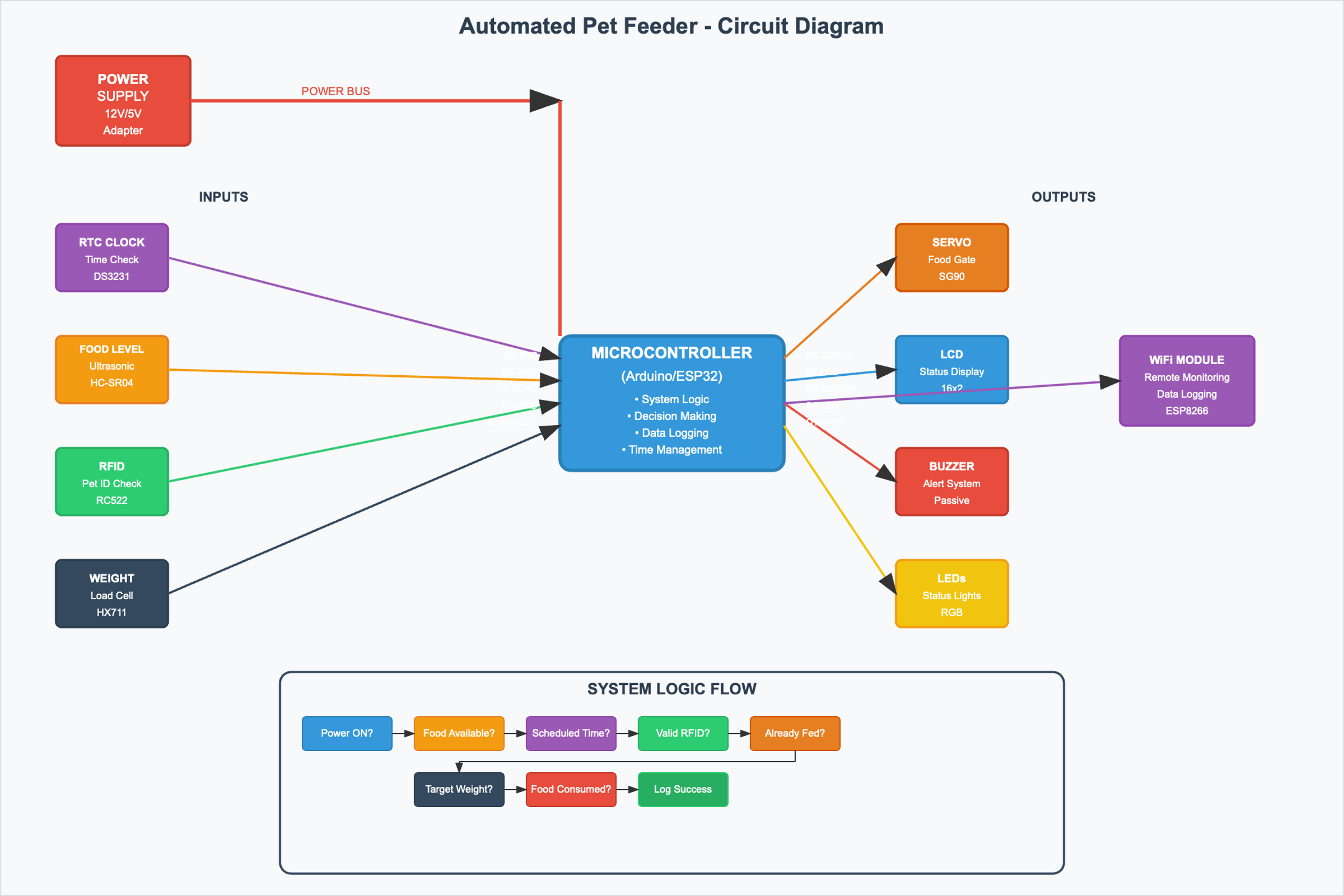
**RESPONSE 2: Review Test Plan Copilot Recommended Refinements:**

* **Robust Timeout Protections: Your initial list was spot on. The pseudocode explicitly includes a timeout for the dispensing motor. This is critical to prevent the motor from burning out if food gets jammed. Timeouts for waiting for a pet and for consumption are also essential for the system to reset itself and not get stuck in one state.**
* **Specific Error Alerts: Instead of a generic "Alert," the system should provide specific messages: "Food is low," "Feeder is jammed," "Unregistered pet detected," or "[Pet Name] has not eaten." This gives the user actionable information.**
* **Hardware Self-Checks: A sophisticated system could run a self-check on startup. It could briefly test the motor, check for a signal from the RFID reader, and get a reading from the weight sensors. If any component fails, it can alert the user immediately instead of failing silently during a feeding cycle.**
* **Manual Override: There must be a physical button on the device to trigger a manual dispense. This is a crucial backup for when the automated logic fails or a pet needs an unscheduled meal.**
* **Power Outage Logic: What happens if the power goes out and comes back on? The system needs to reload the time, feeding schedule, and pet database from non-volatile memory so it can resume its function without needing to be completely reprogrammed.**

RESPONSE 3: CoPilot Review of Step 4 Implementation recommendation to refine logic:

* **Step numbering:** you skip 6 in one place and repeat 7—made consistent below.
* **Truth table:** several rows have missing outputs, and “T=1 F=O” likely means “F=0”.
* **Variables:** you use Dispensing (D) as both a condition and an action. Treat D as the **result** of the decision to dispense.
* **Consumption (C):** happens **after** dispensing; don’t include it in the dispense truth table. Give it its own post-dispense decision.

RESPONSE 4: different ways to enhance the flowchart logic

**  
  
PROMPT:** “make a sketch of the automated pet feeder with circuits showing the inputs and outputs System Check: Power ON? → Yes: Continue / No: Wait ·  Container Check: Food Available? → Yes: Continue / No: Alert & Stop ·  Time Check: Scheduled Time? → Yes: Continue / No: Wait ·  Pet Check: Valid RFID? → Yes: Continue / No: Deny Access ·  Feed Check: Already Fed? → Yes: Block / No: Continue ·  Dispense: Target Weight Reached? → Yes: Stop / No: Continue ·  Verify: Food Consumed? → Yes: Log Success / No: Alert :”  
  
**RESPONSE:**   
I'll create a technical sketch of your automated pet feeder system showing the circuit components, inputs, outputs, and the logical flow you've outlined.

## **RESPONSE 5: Reflection on Ethics of automation of care systems.**

All new processes involve change to people as users and in this case the animals as subjects of the changed process. Automating the feeding of the animals must have an impact on the relationship between them. The bond of care is altered by removing the human from the process and it is not clear this was a factor in making the technological impact change. To identify some of the implications of using AI in particular would be the homogenisation of the pets, automated decisions about their feeding habits reduced to very few observations and variations; the risk of getting this wrong and measuring the impact are very important.  
Animal behaviour and wellbeing need to be considered and measured when introducing machine led decisions for humans and animals.  
AI governance and ethics frameworks provoke us to consider questions on decision making with technology and human flourishing, this might also apply to the best interests test for animals. This was the reflection of the Copilot GPT on ethics, accountability and benefit of automating pet feeding.   
  
PROMPT:

You said:

Analyse, synthesise and summarise this perspective on introducing automated pet feeding systems into animal shelters where the subjects are vulnerable:  
Reflection on Consultation and ethics… All new processes involve change to people as users and in this case the animals as subjects of the changed process. Automating the feeding of the animals must have an impact on the relationship between them. The bond of care is altered by removing the human from the process and it is not clear this was a factor in making the technological impact change. To identify some of the implications of using AI in particular would be the homogenisation of the pets, automated decisions about their feeding habits reduced to very few observations and variations; the risk of getting this wrong and measuring the impact are very important. Animal behaviour and wellbeing need to be considered and measured when introducing machine led decisions for humans and animals. Interview stakeholders such as staff, vets, pet owners, engineers, food manufacturers eg to determine what the current process is, what concerns there are with it and what can be improved. Identify the level of support for the change to the current method of feeding and any introduction of automated system at the shelter. Key Ethical Questions Does efficiency justify reduced human contact? While automation improves consistency, animals in shelters often need human interaction for emotional healing. Who bears responsibility for failures? When technology malfunctions, the ethical burden falls on the institution to have adequate backup systems. Does this technology serve animals or institutions? The primary beneficiary should be animal welfare, not just operational convenience. How do we balance innovation with compassion? Proper management provides environment, housing, and care that permit animals to maintain good health and well-being Ethical Implementation Framework To implement automated feeding ethically: Maintain human oversight and regular interaction Ensure robust backup systems and fail-safes Train staff to interpret technology data meaningfully Use automation to enhance, not replace, compassionate care Regular evaluation of animal welfare outcomes versus efficiency gains The technology itself isn't inherently ethical or unethical—its moral value depends entirely on how thoughtfully it's implemented and whether it truly serves the animals' best interests.

**Copilot said:**

**Copilot**

Here’s a structured **analysis, synthesis, and summary** of the perspective you shared on introducing automated pet feeding systems into animal shelters, particularly where the animals are vulnerable:

**🔍 Analysis**

**1. Ethical Concerns**

* **Human-Animal Bond**: Automation risks weakening the emotional and psychological bond formed through human-led feeding, which is often therapeutic for shelter animals.
* **Responsibility & Accountability**: When technology fails, institutions must bear the ethical responsibility and have contingency plans.
* **Purpose of Technology**: There's a tension between serving operational efficiency and genuinely improving animal welfare.

**2. Risks of Automation**

* **Homogenisation of Care**: AI systems may generalize feeding routines, ignoring individual animal needs and behaviors.
* **Data Limitations**: Automated decisions may rely on limited observations, increasing the risk of inappropriate feeding or neglect.
* **Impact Measurement**: There’s a need for robust metrics to assess how automation affects animal wellbeing.

**3. Stakeholder Engagement**

* The reflection emphasizes the importance of consulting:
  + **Shelter staff** (daily caregivers)
  + **Veterinarians** (health experts)
  + **Pet owners** (empathy and expectations)
  + **Engineers and food manufacturers** (technical feasibility and safety)

**🧠 Synthesis**

The perspective integrates **technological innovation** with **ethical reflection**, advocating for a **balanced approach**. It recognizes that while automation can enhance consistency and efficiency, it must not come at the cost of compassion and individualized care. The ethical framework proposed is not anti-technology but **pro-responsible implementation**, emphasizing that technology should **support**, not **replace**, human empathy and oversight.

**📌 Summary**

Introducing automated pet feeding systems in shelters requires careful ethical consideration, especially for vulnerable animals. Key concerns include the potential loss of human-animal connection, risks of generalized care, and the need for accountability in case of system failures. Stakeholder consultation is essential to understand current practices and assess support for change. An ethical implementation framework should ensure human oversight, robust fail-safes, staff training, and regular evaluation of animal welfare outcomes. Ultimately, technology must serve the animals’ best interests, not just institutional efficiency.

https://www.geeksforgeeks.org/dsa/backtracking-algorithms/  
https://www.geeksforgeeks.org/software-testing/test-plan-software-testing/  
https://www.practitest.com/assets/pdf/system-integration-testing-template.pdf  
Tech Policy Press Podcast   
Animal and tech Ethics