# 

DEBUG AND VERIFY

# Testing

**Scenario 1: Pet Eats as Expected**

**The Assumptions**

* dispenser working
* scheduled feeding time is 8:00 AM
* current time is 8:00 AM
* food level in container is not low
* Pet eats food in 20 minutes
* current bowl weight 78g

**Test output**

* Start device & check dispenser , Dispenser working ,Proceed to next step
* Display current time ,scheduled time & food level - 8:00 AM,8:00am, food: 500 g
* Scheduled time = current time , Proceed to next step
* Check food level , sufficient , Proceed
* Dispense 10 g portions , bowl weight 88 g , dispense more , 98 g
* Start 20-min timer , bowl weight < 75 g , log event & update display
* Loop back to wait for next feeding

**Logic Discussion**

* Check dispenser status to ensure it is working.
* Verify current time matches the scheduled feeding time.
* Check food level to prevent underfeeding.
* Dispense food in portions until the target bowl weight is reached to ensure bowl is not overfilled.
* Start a 20-minute timer to monitor consumption and log the event if food is eaten.
* Loop back to wait for the next feeding.

**Scenario 2: Pet does not eat as Expected**

**The Assumptions**

* dispenser working
* scheduled feeding time is 8:00 AM
* current time is 8:00 AM
* food level in container is not low
* Pet does not eat food in 20 minutes
* current bowl weight 88g

**Test output**

* Start device & check dispenser , Dispenser working ,Proceed to next step
* Display current time ,scheduled time & food level - 8:00 AM,8:00am, food: 500 g
* Scheduled time = current time , Proceed to next step
* Check food level , sufficient , Proceed
* Dispense in portions of 10 g until target bowl weight reached , 98 g
* Start 20-min timer , bowl weight - >75 g , Send an alert “food hasn’t been consumed”.
* End the process

**Logic Discussion**

* Check dispenser status to ensure it is working.
* Verify current time matches the scheduled feeding time.
* Check food level to prevent underfeeding.
* Dispense food in portions until the target bowl weight is reached to ensure bowl is not overfilled.
* Start a 20-minute timer to monitor consumption; if food is not eaten, send an alert.

**Scenario 3: Food bin empty**

**The Assumptions**

* dispenser working
* scheduled feeding time is 8:00 AM
* current time is 8:00 AM

**Test output**

* Start device & check dispenser , Dispenser working ,Proceed to next step
* Display current time ,scheduled time & food level - 8:00 AM,8:00am, food: 0 g
* Scheduled time = current time , Proceed to next step
* Check food level , low , send an alert “food level is low”
* End the process

**Logic Discussion**

* Check dispenser status to ensure it is working.
* Verify current time matches the scheduled feeding time.
* Check food level to ensure sufficient supply and send an alert if the container is empty.

**Comparing expected and actual outcome**

|  |  |  |
| --- | --- | --- |
| **Scenario** | **Expected Outcome** | **Actual Outcome** |
| Pet eats as expected | Pet consumes food within 20 min, event logged, next feeding scheduled | Food eaten, event logged, next feeding scheduled |
| Pet does not eat | Alert sent if food not consumed | Alert sent after 20 min |
| Food bin empty | Alert sent if food level low or empty | Alert sent immediately |

provide an estimated number of meals based on current food level

create another form of alert like app notification SMS or e-mail

use cloud system for saving the logged events

implement a sleep mode to save battery life