Part 3 – AI Agent Integration Reflection

To refine my Step 4 implementation, I consulted Microsoft Copilot for targeted feedback. My first prompt was:
 "Review this word code for an automated pet feeder and suggest improvements for error handling and efficiency."
 Copilot recommended adding a short time buffer to prevent double-feeding when the schedule check runs multiple times within the same minute. It also suggested implementing a retry mechanism for the dispensing motor in case of jams or mechanical resistance.

Next, I asked:
 "Suggest an alternative way to monitor if the pet has eaten without using weight sensors."
It proposed using an infrared beam sensor to detect the presence of food in the bowl. This could be more reliable in shelters where multiple pets might disturb the feeding area.

Finally, I explored real-world deployment by prompting:
 "How could this design be implemented using Arduino or Raspberry Pi with affordable components?"
 Copilot outlined using an Arduino Uno with a servo motor, ultrasonic sensor for bin levels, a load cell for bowl weight, and a real-time clock module for accurate scheduling.

These suggestions made my design more practical, cost-effective, and adaptable. By integrating hardware considerations early, I now have a clearer path from concept to functional prototype while ensuring the system is robust and reliable in real-world conditions.

Review this word code for an automated pet feeder and suggest improvements for error handling and efficiency

```
Plaintext                                                          Copy

BEGIN AutomatedPetFeeder
    SET feedingTimes = [08:00, 18:00]
    LOOP forever
        currentTime = GET_CURRENT_TIME()
        IF currentTime IN feedingTimes THEN
            IF foodBinLevel == LOW THEN
                SEND_ALERT("Food bin empty. Please refill.")
            ELSE
                ROTATE_MOTOR(90)  // Dispense 50g of food
                WAIT(10 minutes)
                IF bowlWeightChange >= 50 THEN
                    LOG_EVENT("Feeding successful at " + currentTime)
                ELSE
                    SEND_ALERT("Food not eaten at " + currentTime)
                ENDIF
            ENDIF
        ENDIF
    ENDLOOP
END
```

Answer:
Adding a time buffer to prevent double-feeding if the schedule check loops multiple times in the same minute, and introducing a retry mechanism for the motor in case of mechanical blockage.

Suggest an alternative way to monitor if the pet has eaten without using weight sensors.

Using an infrared beam sensor to detect the presence of food in the bowl. This could be more reliable in shelters where multiple pets might disturb the feeding area.

How could this design be implemented using Arduino or Raspberry Pi with affordable components?

Using an Arduino Uno with a servo motor, ultrasonic sensor for bin levels, a load cell for bowl weight, and a real-time clock module for accurate scheduling.

Figure 1: Microsoft Copilot Response