

ECE 445

Senior Design Laboratory

Vision-Assisted Pet Feeder

with BLE Proximity and Consumption Monitoring

Design Document

Team Number:	18
Team Members:	Junwei Pan (junweip2) Jiaao Zhang (jiaao2) Ziteng Qian (zitengq2)
Teaching Assistant:	Pusong Li
Date:	October 2, 2026

Contents

1	Introduction	1
1.1	Problem and Solution	1
1.2	Visual Aid	1
1.3	High-Level Requirements	2
2	Design	2
2.1	System Block Diagram	2
2.2	Physical Design	3
2.3	Operating Sequence and Quantitative Interfaces	4
2.4	Subsystem 1: Orchestration and Clocking	5
2.5	Subsystem 2: BLE Proximity Sensing	6
2.6	Subsystem 3: Vision and Feeding-Zone Monitoring	7
2.7	Subsystem 4: Mechanics and Closed-Loop Dispensing	8
2.8	Subsystem 5: Remote Communications	9
2.9	Subsystem 6: General Power and Main-Control PCB	10
2.10	Event Classification Logic	11
2.11	Tolerance Analysis	12
3	Cost and Schedule	13
3.1	Cost Analysis	13
3.1.1	Labor	13
3.1.2	Parts and Fabrication	13
3.1.3	Total Cost and Budget	15
3.2	Schedule	16
4	Discussion of Societal Impact, Engineering Standards, Ethics, and Safety Considerations	17
4.1	Societal Impact	17
4.2	Engineering Standards	17
4.3	Ethics	17
4.4	Safety and Verification	18
4.5	Operating Procedure	18

List of Figures

1	Use-context visual aid. At a stored meal time, the feeder measures and dispenses food, uses the collar BLE beacon to control camera workload, checks whether the pet enters the camera’s feeding zone, and sends meal status to the owner through Telegram or an equivalent service.	2
2	System block diagram. The six modular subsystems preserve the supplied sketch’s power, clocking, BLE sensing, vision, mechanical control, and remote communication functions while making the ESP32-S3 control authority and quantitative interfaces explicit.	3
3	Preliminary physical placement, not to scale. Critical relationships are the isolated bowl load path, enclosed food drive, fixed camera view, protected electronics, and separation of the low-level load-cell wiring from the motor path.	4
4	ESP32 meal-event state machine. Pet proximity changes the camera workload only inside OBSERVE; it never starts another dispense transaction.	4

List of Tables

1	Subsystem interface definitions.	5
2	Orchestration and clocking requirements and verification.	6
3	BLE proximity-sensing requirements and verification.	7
4	Vision-subsystem requirements and verification.	8
5	Mechanical and closed-loop control requirements and verification.	9
6	Remote-communications requirements and verification.	10
7	Power and main-control PCB requirements and verification.	11
8	Closed-loop dispensing tolerance budget.	12
9	Estimated engineering labor.	13
10	Estimated parts cost and procurement notes.	13
11	Conservative estimate before school-stock confirmation.	15
12	Implementation schedule and course milestones.	16

1 Introduction

1.1 Problem and Solution

Pet owners may be away from home when a scheduled meal is due. Conventional timer-based feeders can release food at a programmed time, but motor motion alone does not prove that the correct mass reached the bowl, that the pet approached the feeder, or that food was removed afterward. These outcomes require different responses: a jammed auger is a hardware fault, an untouched meal may mean that the pet missed or refused the meal, and an approach without a measurable decrease in bowl mass may justify an owner check. Nutritional management also requires an individualized feeding amount and frequency rather than an unverified open-loop dispense [1]. The project therefore addresses the entire scheduled feeding event while reporting objective observations only; it is not a medical diagnostic system.

The proposed solution is a single-pet, single-bowl feeder that combines RTC scheduling, closed-loop weight-controlled dispensing, BLE proximity sensing, local computer vision, and remote event reporting. At each stored meal time, a DS3231M RTC wakes the ESP32-S3 main controller. The feeder weighs any residual food and dispenses only $m_{\text{add}} = \max(0, m_0 - m_1)$, where m_0 is the configured target and m_1 is the stable pre-meal bowl mass. During a nominal 30 min observation window, a collar BLE beacon provides a low-computation near/far cue; BLE-near requests full-rate camera inference, while low-rate checks remain available during the window as a fallback. A Raspberry Pi reports pet presence in a fixed bowl region, and the ESP32 combines presence with bowl-mass history to classify the event. Outside a scheduled meal window, BLE scanning and active camera inference are disabled. This event-driven design improves on a conventional timer feeder by measuring the delivered mass, avoiding repeated top-ups, confirming feeding-zone presence, estimating food removal, and sending a timestamped Telegram notification or equivalent TLS-protected record to the owner.

1.2 Visual Aid

Figure 1 is a use-context drawing rather than an internal block diagram: it shows the pet, the feeder, the camera field of view, the collar beacon, and the remote owner notification.

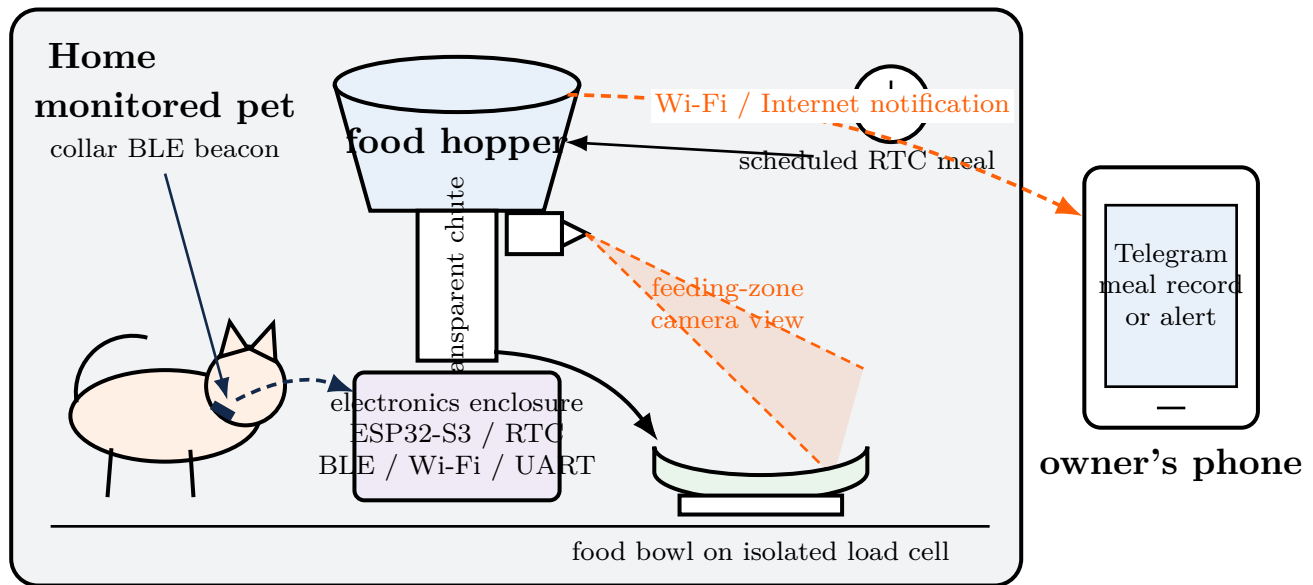


Figure 1: Use-context visual aid. At a stored meal time, the feeder measures and dispenses food, uses the collar BLE beacon to control camera workload, checks whether the pet enters the camera’s feeding zone, and sends meal status to the owner through Telegram or an equivalent service.

1.3 High-Level Requirements

The project is successful only if all of the following objective requirements are met.

- HLR 1.** For a configured 20.0 g target and an initial residual-food mass from 0 g through 10 g, the feeder shall finish with $20.0\text{ g} \pm 2.0\text{ g}$ in at least 9 of 10 trials and shall perform no more than one dispense transaction for that scheduled meal.
- HLR 2.** During scheduled observation windows, the combined BLE, vision, and weight pipeline shall correctly classify at least 27 of 30 controlled trials: 10 consumption trials with at least 5 g removed, 10 no-approach trials, and 10 approach-without- consumption trials with less than 5 g removed. A result shall be produced within 2 s after the applicable threshold is met or the observation window ends.
- HLR 3.** Under normal Wi-Fi and Internet connectivity, a timestamped meal record or abnormal-event alert shall appear on the owner’s remote device within 10 s of classification in at least 18 of 20 end-to-end trials. Loss of Internet service shall not prevent the scheduled local dispense operation.
- HLR 4.** Outside a stored meal’s nominal 30 min observation window, the controller shall execute no BLE scan windows and the vision coprocessor shall process no active-inference frames during a 10 min idle test. The controller shall begin the corresponding meal state sequence within 2 s of the RTC alarm and shall return both functions to idle after reporting.

2 Design

2.1 System Block Diagram

Figure 2 formalizes every subsystem and information path from the supplied hand-drawn block diagram. The ESP32-S3 remains the authoritative orchestrator. The BLE sensing subsystem requests a vision

workload change, but it never directly commands food dispensing. Likewise, the Raspberry Pi reports observations only; all motor decisions and safety limits stay on the main-control PCB.

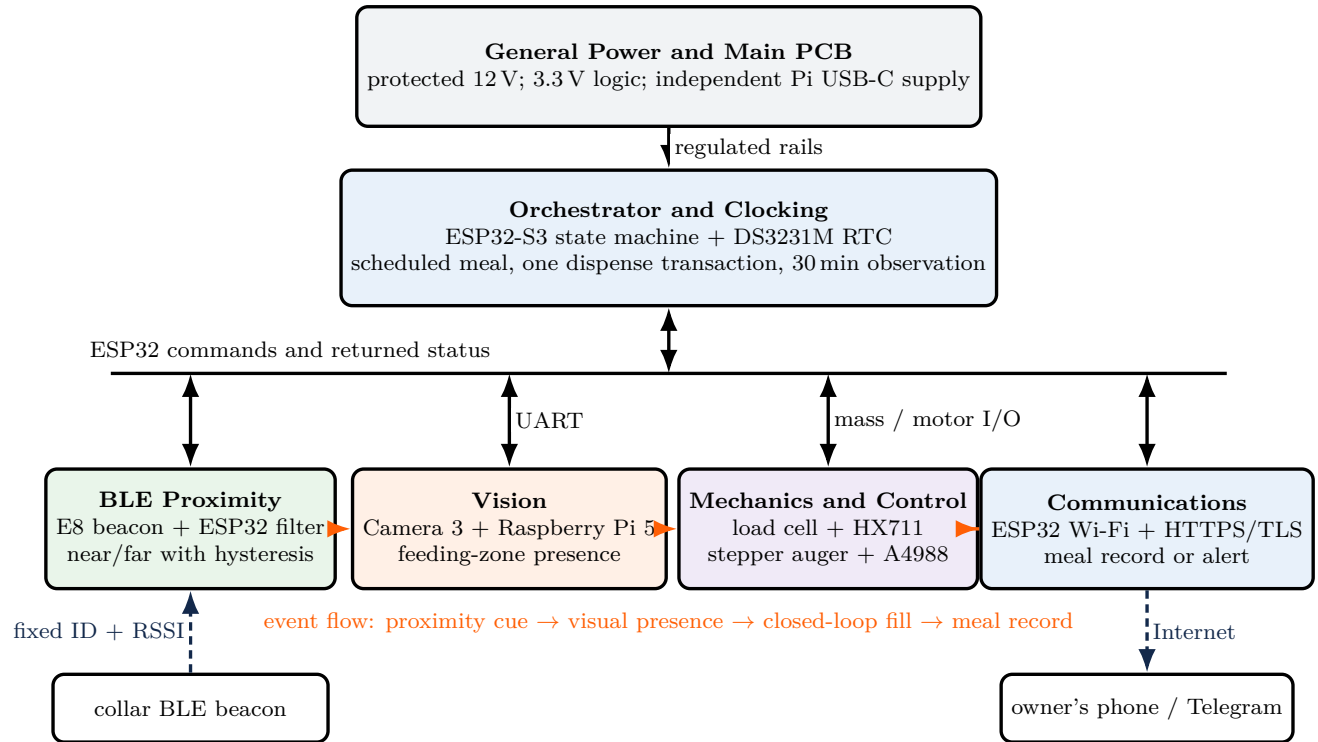


Figure 2: System block diagram. The six modular subsystems preserve the supplied sketch’s power, clocking, BLE sensing, vision, mechanical control, and remote communication functions while making the ESP32-S3 control authority and quantitative interfaces explicit.

2.2 Physical Design

The feeder uses a vertical hopper above an enclosed auger and transparent delivery chute. The bowl is supported only by a mechanically isolated load-cell platform so hopper weight and frame forces do not enter the measurement. The camera is fixed to the feeder body and aimed downward so its field of view contains the entire bowl plus a user-defined surrounding feeding polygon. The main PCB and Raspberry Pi are enclosed away from the food path, with locking connectors for the motor, load cell, camera/Pi UART, and power. The sketch in Figure 3 defines the relative placement needed for assembly; final external dimensions follow the hopper and auger CAD and do not change the electrical interfaces specified in this document.

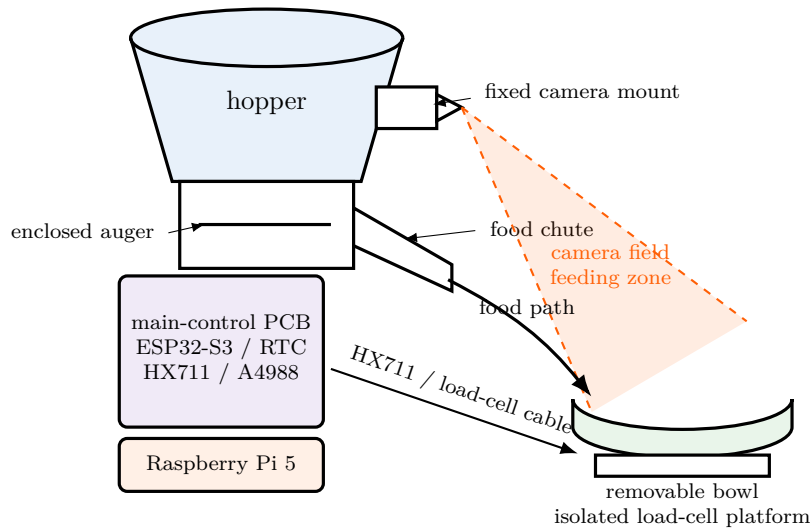


Figure 3: Preliminary physical placement, not to scale. Critical relationships are the isolated bowl load path, enclosed food drive, fixed camera view, protected electronics, and separation of the low-level load-cell wiring from the motor path.

2.3 Operating Sequence and Quantitative Interfaces

The RTC alarm is the only normal entry from IDLE into a meal event. In PRE-WEIGH, the motor is disabled and the controller accepts the median of five samples only when their range is at most 0.8 g. If $m_1 \geq m_0$, dispensing is skipped; otherwise the controller runs coarse auger motion while the deficit exceeds 5 g, then uses fine bursts separated by at least 300 ms and a new stable-mass decision. The meal identifier is committed to nonvolatile memory before observation so a reset or a repeated pet approach cannot cause a second fill. Figure 4 shows the full sequence, including the non-meal low-activity behavior requested in the supplied diagrams.

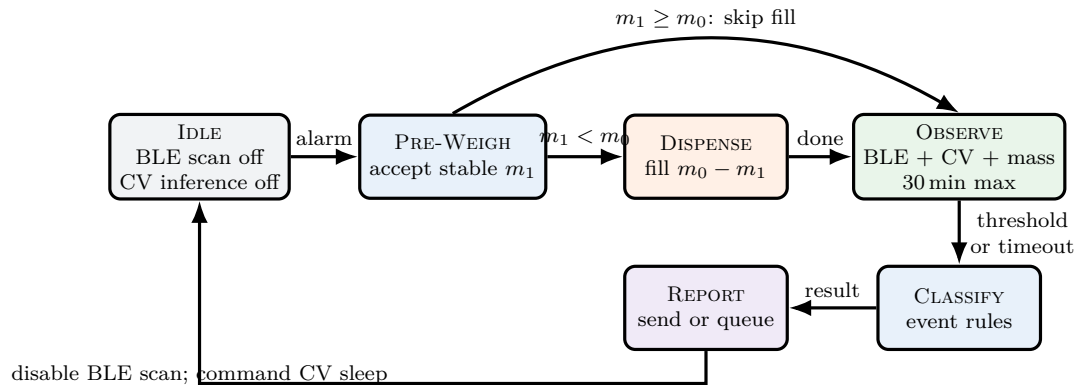


Figure 4: ESP32 meal-event state machine. Pet proximity changes the camera workload only inside OBSERVE; it never starts another dispense transaction.

Table 1: Subsystem interface definitions.

Source	Destination	Physical or logical interface	Contract
DS3231M	ESP32-S3	3.3 V I ² C up to 400 kHz; active-low alarm interrupt	Stored alarm begins one meal within 2 s; time survives at least 60 s without main power.
Minew E8	ESP32-S3	BLE advertisement, nominal 250 ms interval	Accept only the configured ID; use at least five valid RSSI samples and at least 6 dB hysteresis.
ESP32-S3	Raspberry Pi 5	3.3 V UART, 115200 baud, 8-N-1	ESP32 sends sleep/low-duty/active command; Pi returns a 12 byte sequence-numbered, CRC-protected observation.
Camera Module 3	Raspberry Pi 5	CSI-2 ribbon	Process at least 640×480 pixels and at least 5 frame/s in active mode.
TAL220B/HX711	ESP32-S3	Bridge input to HX711; DOUT/PD_SCK at 10 sample s^{-1}	Stable calibrated mass has at most 0.5 g RMS error from 0 g to 100 g.
ESP32-S3	A4988	3.3 V STEP, DIR, ENABLE; pulses at least $2 \mu\text{s}$	Current limit is set before connecting the motor; accumulated motor-on time is at most 15 s per meal.
ESP32-S3	Owner service	2.4-GHz Wi-Fi; HTTPS over TLS	Acknowledge a timestamped metadata record or retain it locally; raw camera images are not sent by default.

2.4 Subsystem 1: Orchestration and Clocking

The ESP32-S3 on the custom PCB owns the complete state machine and is the only device permitted to enable the A4988. A DS3231M provides timekeeping, an alarm interrupt, and backup operation when the main 3.3 V rail is absent [2]. Each configured meal has a unique identifier derived from its schedule entry and date. The controller writes that identifier and a “dispense complete” flag to nonvolatile storage before entering OBSERVE; after a brownout or watchdog reset, the controller restores the record and cannot re-run a completed meal. This subsystem contributes directly to HLR 1 by preventing duplicate fills and to HLR 4 by enforcing the meal window and idle sensing policy.

In IDLE, the controller disables BLE scan windows and sends a `CV_SLEEP` UART command. The Raspberry Pi may remain electrically powered, but its camera capture and inference pipeline must be stopped; this is the implementable meaning of “camera off” in the concept sketch. At the RTC alarm, the controller enters PRE-WEIGH within 2 s. Once the meal has been classified and its record has been acknowledged or queued, the controller returns to IDLE. A pet’s BLE signal cannot initiate a meal outside this sequence.

Table 2: Orchestration and clocking requirements and verification.

Requirement	Verification procedure and pass condition
O1. A stored meal shall enter PRE-WEIGH within 2 s of the RTC alarm in 20 of 20 trials.	Connect a two-channel logic analyzer to the DS3231M alarm pin and a debug GPIO asserted on entry to PRE-WEIGH. Program alarms at least 1 min apart and record the falling-edge-to-debug-edge delay for 20 trials. Pass if every delay is 2 s or less.
O2. Time and the next alarm shall remain valid through at least 60 s of main-power loss.	Synchronize the RTC to a reference clock, remove the 12 V input for 60 s while leaving the RTC backup source installed, restore power, and read the time and next alarm over I ² C. Repeat five times. Pass if time error is at most 2 s and the alarm entry from O1 still occurs exactly once in every trial.
O3. One scheduled meal shall cause no more than one dispense transaction, including after a reset during and after dispensing.	Instrument the motor-enable signal with a logic analyzer. For ten meal IDs, reset the ESP32 at a randomized point after the nonvolatile completion record is written, then replay BLE-near transitions five times. Pass if each meal ID contains at most one contiguous dispense transaction and no replay enables the motor.
O4. During a 10 min off-meal IDLE test, the ESP32 shall execute zero BLE scan windows and the Pi shall report zero captured inference frames; both functions shall return to idle after a completed meal.	Enable firmware counters for BLE scan-window starts and camera frames. Clear the counters, observe IDLE for 10 min, then run one complete meal and wait for REPORT to finish. Pass if the idle interval adds zero to both counters, the counters increase only during OBSERVE, and both stop increasing within 2 s after return to IDLE.

2.5 Subsystem 2: BLE Proximity Sensing

The sensing subsystem consists of the Minew E8 collar beacon and the ESP32-S3 integrated BLE receiver. The beacon broadcasts a fixed identifier at a nominal 250 ms interval [3]. The ESP32 rejects other identifiers, then filters at least five recent valid RSSI values using a median statistic. A calibrated entry threshold T_{near} and exit threshold T_{far} satisfy

$$T_{\text{near}} - T_{\text{far}} \geq 6 \text{ dB}, \quad (1)$$

which prevents rapid state changes near the boundary. RSSI is affected by multipath, orientation, and body shadowing, so this subsystem reports only **NEAR** or **FAR**; it does not report centimeters and is not used as sole evidence that the pet is eating.

Within **OBSERVE**, **NEAR** requests **CV_ACTIVE**. While the tag is far, the Pi may perform a fallback low-duty check no more frequently than one frame every 5 s; this prevents a missing or depleted tag from making the system blind without running full-rate inference continuously. If BLE disagrees with camera and weight evidence, the latter two determine classification. This subsystem reduces the vision workload in support of HLR 4 and provides the approach cue used by HLR 2.

Table 3: BLE proximity-sensing requirements and verification.

Requirement	Verification procedure and pass condition
B1. At 0.5 m line of sight, the algorithm shall assert NEAR within 2 s in at least 18 of 20 trials.	Mark a 0.5 m separation using a tape measure in the intended room. Begin each trial from FAR , enable scanning, and log the first valid advertisement and state-transition timestamps. Randomize tag orientation between trials. Pass if at least 18 delays are no more than 2 s.
B2. At 3.0 m in the same room, the algorithm shall remain FAR for a 10 s observation in at least 18 of 20 trials.	Move the same tag to a measured 3.0 m location, randomize orientation, and record filtered state for 10 s. Repeat 20 times. Pass if at least 18 trials contain no NEAR assertion.
B3. Every state decision shall use at least five valid advertisements, and the calibrated entry/exit thresholds shall differ by at least 6 dB.	Feed a recorded RSSI sequence through the firmware test interface while logging raw samples, the median, and state. Include a ramp crossing both thresholds and 100 samples inside the hysteresis band. Pass if no state decision precedes the fifth sample, the configured threshold difference is at least 6 dB, and no toggles occur solely within the band.
B4. BLE shall never directly enable the motor.	Hold the RTC outside any meal window, replay 100 valid NEAR decisions, and observe A4988 ENABLE with a logic analyzer. Pass if the motor remains disabled for the entire test and no dispense state is entered.

2.6 Subsystem 3: Vision and Feeding-Zone Monitoring

A Raspberry Pi 5 receives images from a Raspberry Pi Camera Module 3 through CSI-2. The camera uses the Sony IMX708 sensor and supports autofocus and 1080p video [4]. The design processes at least 640×480 pixels with a lightweight, quantized detector. A MobileNetV2- based SSD is a candidate because its inverted-residual architecture is intended for efficient mobile inference [5]; the final model is selected from measured accuracy and latency rather than model name alone. A cat or dog detection counts only when its bounding box overlaps the fixed polygon surrounding the bowl. Presence changes state only after at least three positive detections in five consecutive processed frames.

The ESP32 and Raspberry Pi communicate through a 3.3 V UART link at 115200 baud. The implementation uses the ESP32’s standard UART driver and the Python `pySerial` library on the Raspberry Pi, so low-level serial transmission, framing, and buffering are handled by existing libraries. The ESP32 sends newline-terminated text commands such as `CV_SLEEP`, `CV_LOW_DUTY`, and `CV_ACTIVE`. The Raspberry Pi reads these commands using the serial library and returns a short text message containing the detected state and confidence value, for example `PRESENT,85`. The Raspberry Pi provides vision observations only and cannot command the motor. Raw camera frames remain local and are neither stored nor uploaded by default. This subsystem supplies the primary feeding-zone evidence for HLR 2 while its commanded modes support HLR 4.

Table 4: Vision-subsystem requirements and verification.

Requirement	Verification procedure and pass condition
V1. The installed camera shall contain the complete bowl and feeding polygon in every processed frame at a resolution of at least 640×480 .	Place fiducial markers at the polygon vertices and capture 100 frames after final camera mounting. Use the Pi log to record resolution and inspect the frames. Pass if all four markers and the full bowl boundary appear in all frames and every processed frame is at least 640×480 .
V2. In <code>CV_ACTIVE</code> , the Pi shall process at least 5 frame/s and provide a valid observation no later than 500 ms after the fifth frame used for a decision.	Run 20 one-minute active sessions while logging frame-completion and UART-transmit timestamps. Use a logic analyzer on UART TX as an independent timing reference. Pass if each session averages at least 5 frame/s and every fifth-frame-to-response delay is at most 500 ms.
V3. Feeding-zone presence classification shall achieve at least 90% accuracy on at least 200 labeled frames, including at least 100 positive and 100 negative frames, two indoor lighting conditions, and partial occlusions.	Freeze the labeled evaluation set before final tuning. Run inference once on every frame, form a confusion matrix, and compute $(TP + TN)/N$. Pass if the set composition is met and accuracy is at least 90%. Report false-positive and false-negative rates separately.
V4. A presence transition shall require at least three agreeing detections among five consecutive processed frames.	Inject all 32 possible five-bit detection sequences into the temporal-filter unit test and compare against an independently calculated majority result. Pass if all 32 outputs agree and no one- or two-frame positive sequence asserts presence.
V5. The UART interface shall correctly exchange the defined newline-terminated text commands and responses at 115200 baud. Valid commands shall be parsed correctly, and malformed or unknown messages shall not change the vision state or motor state.	Send each defined command (<code>CV_SLEEP</code> , <code>CV_LOW_DUTY</code> , and <code>CV_ACTIVE</code>) 20 times from the ESP32 to the Raspberry Pi and verify that the Pi enters the corresponding vision mode. Then send valid response messages such as <code>PRESENT,85</code> and verify that the ESP32 parses the state and confidence correctly. Inject malformed and unknown messages and confirm that they are ignored without changing the current vision state or enabling the motor. Pass if all valid messages are parsed correctly and no malformed message causes an unintended state change.

2.7 Subsystem 4: Mechanics and Closed-Loop Dispensing

The bowl is mounted on a TAL220B 5 kg load cell. An HX711 supplies bridge excitation, gain, and 24-bit analog-to-digital conversion [6, 7]. Only measurements taken with the motor disabled are eligible for control. Five consecutive 10 samples^{-1} readings must span no more than 0.8 g; their median is the stable mass. The configured target is m_0 , the stable pre-meal mass is m_1 , and the commanded addition is

$$m_{\text{add}} = \max(0, m_0 - m_1). \quad (2)$$

If $m_1 \geq m_0$, the motor remains disabled but observation still occurs. The controller does not replace food removed later in the same meal window.

A 17HS4401 NEMA-17 bipolar stepper rotates the enclosed auger. The motor has a 1.8° full-step angle and approximately 0.40 N m holding torque [8]. The A4988 driver accepts 3.3 V STEP, DIR, and ENABLE signals; the initial phase-current limit is 1.2 A [9]. Continuous coarse motion is used while the remaining deficit exceeds 5 g; fine bursts are followed by at least 300 ms and a new stable measurement. Two consecutive bursts producing less than 0.5 g total increase indicate a jam or empty hopper. Total motor-on time is capped at 15 s per meal. This subsystem directly satisfies HLR 1 and provides the mass-change evidence required by HLR 2.

Table 5: Mechanical and closed-loop control requirements and verification.

Requirement	Verification procedure and pass condition
M1. After two-point calibration, stable mass from 0 g to 100 g shall have a maximum absolute error of at most 0.5 g.	Use certified masses at 0 g, 25 g, 50 g, 75 g, and 100 g. Record ten stable readings at each mass with the motor disabled. Compute RMS and maximum absolute error against the certified values. Pass if the maximum absolute error is at most 0.5 g.
M2. Five consecutive samples shall be accepted only when their range is at most 0.8 g, and the reported mass shall be their median.	Inject ten synthetic five-sample groups below and above the 0.8 g limit, including boundary cases, and compare the firmware result with a spreadsheet calculation. Pass if all stable groups are accepted with the correct median and all unstable groups are rejected.
M3. For target 20.0 g and initial residual masses of 0 g, 5 g, and 10 g, final bowl mass shall be $20.0 \text{ g} \pm 2.0 \text{ g}$ in at least 9 of 10 randomized trials, with one transaction per meal.	Use the same pellet geometry and hopper fill range intended for demonstration. Randomize ten trials across the three residual masses, weigh the final bowl using an independent calibrated scale, and capture motor enable. Pass if at least nine final masses lie in 18.0 g to 22.0 g and every trial contains at most one transaction.
M4. If two consecutive commanded bursts produce less than 0.5 g total increase, the A4988 shall be disabled within 2 s; accumulated motor-on time shall never exceed 15 s per meal.	Block the outlet or empty the hopper while monitoring ENABLE with a logic analyzer and phase current with a current probe. Perform ten fault trials. Pass if every trial disables the driver within 2 s of the second ineffective burst, records a fault, and has no more than 15 s accumulated motor-on time.
M5. Removal of at least 5 g after dispensing shall be distinguished from no removal in at least 18 of 20 controlled trials.	Create ten no-removal trials and ten trials using a certified 5 g or larger pellet packet. Wait for stable mass after each action and compare the event output to the known condition. Pass if at least 18 classifications are correct.

2.8 Subsystem 5: Remote Communications

The ESP32-S3 integrated 2.4-GHz Wi-Fi radio sends event metadata through HTTPS over TLS. The initial service is the Telegram Bot API, which avoids a custom phone application; the interface is kept service-independent so another TLS-protected messaging endpoint can be substituted. A record contains meal ID, scheduled time, pre-dispense mass, target mass, post-dispense mass, estimated removal, BLE result, vision result, final classification, fault code, and sequence number. Raw camera images are

excluded by default.

The local meal sequence does not wait for Internet access. If transmission fails, the controller stores at least 50 records in nonvolatile memory and retries them in sequence when connectivity returns. An application-layer acknowledgement, rather than successful Wi-Fi association alone, marks a record delivered. This subsystem satisfies HLR 3 and provides the owner-facing outcome shown in the original visual aid.

Table 6: Remote-communications requirements and verification.

Requirement	Verification procedure and pass condition
C1. Every classified meal record shall contain all defined fields with a unique sequence number and shall contain no raw image payload.	Generate one record for each event class and capture the decrypted test-endpoint payload in a controlled development environment. Compare keys and types with the record schema. Pass if every field is present, sequence numbers are unique and monotonic, and no image or frame bytes appear.
C2. Under normal connectivity, the owner device shall display the record within 10s of classification in at least 18 of 20 trials.	Synchronize ESP32 and phone timestamps to the same reference, then run 20 classification events on the intended Wi-Fi network. Record classification and Telegram display times. Pass if at least 18 delays are no more than 10s.
C3. With Internet access removed, the subsystem shall retain at least 50 records, continue scheduled local feeding, and deliver the records in order after access returns.	Disconnect the access point from the Internet without removing local power. Generate 50 uniquely numbered events plus one scheduled feeding event, reboot once, and restore Internet access. Pass if feeding occurs on schedule, all 50 records survive the reboot, and all arrive once and in sequence after reconnection.
C4. A remote message shall never directly energize the motor.	Send valid and malformed service responses, duplicate acknowledgements, and 100 unsolicited inbound messages while monitoring A4988 ENABLE. Pass if no input causes a transition to DISPENSE and ENABLE remains inactive unless an independent RTC meal is active.

2.9 Subsystem 6: General Power and Main-Control PCB

The custom PCB is the main embedded controller, not a passive breakout. It contains the ESP32-S3-WROOM-1-N8R8, DS3231M, HX711, A4988, input protection, fuse, bulk and local decoupling, status indicators, programming header, and locking subsystem connectors. A listed external adapter provides 12V for the motor rail. A buck stage and 3.3V regulator supply logic. At least 100 μF of local bulk capacitance is placed beside the A4988 motor supply because stepper wiring can generate supply transients [10]. The Raspberry Pi uses its independent recommended 27W USB-C supply [11]; only signal ground is shared at the UART interface.

Motor return current is routed directly to the power-entry ground. The load-cell analog return is kept separate until one controlled connection point, and its cable is routed away from motor wiring. On reset, invalid mass, jam, or motor timeout, the safe actuator state is A4988 disabled. Loss of valid Pi packets for 2s marks vision unavailable but does not prevent scheduled local feeding when weighing and controller operation remain valid. This subsystem supports every HLR by providing bounded rails and a

fail-safe controller, with particular importance to the accuracy and no-duplicate behavior of HLR 1.

Table 7: Power and main-control PCB requirements and verification.

Requirement	Verification procedure and pass condition
P1. With 12 V input varied by $\pm 5\%$, the logic rail shall remain $3.3\text{ V} \pm 5\%$ during BLE scanning, Wi-Fi transmission, HX711 sampling, and worst-case motor operation.	Use a bench supply at 11.4 V, 12.0 V, and 12.6 V. Run all listed loads concurrently. Measure the logic rail with a calibrated DMM and oscilloscope for 60 s at each input. Pass if DC voltage remains in 3.135 V to 3.465 V and no transient leaves that range.
P2. The motor input shall include reverse-polarity protection, a replaceable or resettable fuse, and at least 100 μF local A4988 bulk capacitance.	Inspect the schematic, PCB, and assembled board against the bill of materials. With the motor disconnected and a current-limited bench supply set to 0.10 A, apply reverse polarity for 5 s. Pass if all three protection elements are present, reverse current remains below the set limit, and normal polarity afterward produces no damage or abnormal heating.
P3. Initial A4988 phase-current limit shall be no more than 1.2 A, and the driver shall complete a 15 s worst-case run without thermal shutdown.	Set the current limit with the selected sense resistors and reference voltage, then measure one phase using a current probe. Run the loaded auger for 15 s while observing ENABLE, current, and driver temperature with a thermal camera. Pass if phase current is at most 1.2 A, motion does not stop from thermal shutdown, and the test is terminated at 15 s.
P4. A jam, timeout, or controller reset shall drive A4988 ENABLE inactive within 100 ms; loss of valid Pi packets for 2 s shall mark vision unavailable without cancelling local feeding.	Use a logic analyzer on the fault/reset stimulus, Pi UART, and ENABLE. Inject each condition ten times. Pass if ENABLE becomes inactive within 100 ms every time, the vision-unavailable flag appears within 2 s of UART loss, and a separately scheduled local pre-weigh still begins.
P5. UART levels shall stay between 0 V and 3.3 V, and the Pi and PCB shall share only the intended signal reference.	With both supplies on, measure UART TX/RX high and low levels using an oscilloscope and verify continuity of the common ground path with a DMM while power is removed. Pass if no UART pin exceeds $3.3\text{ V} + 0.1\text{ V}$ or falls below -0.1 V , and no unintended motor-rail connection to the Pi exists.

2.10 Event Classification Logic

The ESP32 combines the three evidence channels conservatively. BLE provides context, vision provides feeding-zone presence, and stable mass change provides evidence that material left the bowl. The rules are intentionally explicit so the HLR 2 trial labels can be generated before testing:

- **Consumed:** vision confirms presence and the stable bowl mass decreases by at least 5 g from the

post-dispense baseline.

- **No approach:** neither BLE-near nor vision-present occurs during the 30 min window.
- **Approach without consumption:** vision is present for at least 15 s, or the pet enters at least three times within 15 min, while cumulative stable mass decrease remains below 5 g.
- **Hardware fault:** two consecutive bursts add less than 0.5 g, the 15 s motor limit is reached, or a stable pre-weigh cannot be obtained.

For the end-to-end verification, the team will prepare 30 controlled trials with the class counts specified in HLR 2. Certified masses and manual pellet removal provide mass ground truth; labeled or prerecorded pet imagery and a mock pet target provide repeatable vision conditions. Live-pet testing is not required to prove the engineering requirements.

2.11 Tolerance Analysis

The most critical interface is the closed-loop auger and load-cell combination because it alone determines whether HLR 1 can be met. Pellets are discrete, material may remain in the chute after the motor stops, and vibration corrupts weight measurements. Unlike a BLE or vision mistake, an overshoot cannot be corrected in software because the feeder cannot remove food from the bowl.

Let T be the target mass, $\hat{m}_k$ the accepted stable measurement after fine burst k , e_L the worst-case load-cell measurement magnitude, q_f the largest mass released by one fine burst, and m_f the maximum material still in flight when a stable measurement is accepted. Fine dispensing stops when $\hat{m}_k \geq T$. The worst positive error is

$$m_{\text{final}} - T \leq e_L + q_f + m_f. \quad (3)$$

The design allocations are

$$e_L \leq 0.5 \text{ g}, \quad q_f \leq 0.8 \text{ g}, \quad m_f \leq 0.6 \text{ g}. \quad (4)$$

Therefore,

$$m_{\text{final}} - T \leq 0.5 \text{ g} + 0.8 \text{ g} + 0.6 \text{ g} = 1.9 \text{ g} < 2.0 \text{ g}. \quad (5)$$

The worst-case positive margin is 0.1 g. Under normal operation, because the controller does not stop before $\hat{m}_k \geq T$, the underfill magnitude is bounded by the load-cell error:

$$T - m_{\text{final}} \leq e_L \leq 0.5 \text{ g}. \quad (6)$$

Table 8: Closed-loop dispensing tolerance budget.

Term	Allocated maximum	Share of 1.9 g	How established
Load-cell error e_L	0.5 g	26.3%	Certified masses, motor disabled
Fine-burst mass q_f	0.8 g	42.1%	Maximum of 30 single-burst trials
Food in flight m_f	0.6 g	31.6%	Catch tray after commanded motor stop
Worst-case total	1.9 g	100%	Arithmetic worst-case sum
Acceptance limit	2.0 g	–	HLR 1
Remaining margin	0.1 g	–	Limit minus worst-case sum

The 0.8 g fine-burst allocation is the largest contributor and is therefore the first parameter to reduce if testing exceeds the budget. Each term will be measured separately before the end-to-end trials. Coarse feed stops at least 5 g below target; fine bursts begin only after a 300 ms minimum delay and satisfaction of the five-sample stability criterion. If $q_f > 0.8$ g, the team will reduce the burst step count or auger speed. If $m_f > 0.6$ g, the outlet geometry or chute angle will be revised. If either term still exceeds its allocation, the mechanical design will be changed rather than widening the HLR 1 acceptance range.

3 Cost and Schedule

3.1 Cost Analysis

3.1.1 Labor

The reported 2024–2025 average starting salary for electrical engineering graduates is \$87,429 [12]. Dividing by 2,080 hours/year gives approximately \$42/hour. With 140 hours per member and the required 2.5 multiplier, $C_i = 42 \times 140 \times 2.5 = \$14,700$.

Table 9: Estimated engineering labor.

Member	Rate (\$/h)	Hours	Cost (\$)
Junwei Pan	42	140	14,700
Jiaao Zhang	42	140	14,700
Ziteng Qian	42	140	14,700
Total		420	44,100

3.1.2 Parts and Fabrication

Table 10 lists the current hardware and development accessories. Q denotes vendor prices checked October 2, 2026, before tax/shipping; A denotes an unconfirmed allowance. R = required, C = conditional/replaceable, D = development-only, and S = spare. Values are line totals. No loans are available; unavailable school parts must be purchased. Only confirmed free supplies are priced at \$0. E-Shop stocks general motors, MCUs, regulators and passives, but exact models and quantities require verification [13]. The free E-Shop (1041) is distinct from the paid Supply Center (1031).

Table 10: Estimated parts cost and procurement notes.

Item	Part / price basis	Qty.	Cost (\$)	Use	Source / purchase decision
Vision computer	Pi 5 2 GB, SC1110; Q [14]	1	77.50	R	Purchase required; no loan available.
Camera	Camera Module 3; Q [15]	1	29.25	R/C	Camera required; alternative needs validation.
Pi supply	SC1153, 27 W; Q [16]	1	12.95	R/C	Required; verify any substitute ratings.
Pi cooler	SC1148; Q [17]	1	10.95	C	Recommended; thermal test before omission.

Table 10 – continued

Item	Part / price basis	Qty.	Cost (\$)	Use	Source / purchase decision
Camera cable	22-to-15-pin FPC; base price Q [18]	1	3.95	C	Required for CSI; verify length.
Storage	32 GB microSD, SKU TBD; A	1	10.00	R/C	Required boot storage.
Vision subtotal			144.60		
Load cell	TAL220B, SEN-14729; Q [6]	1	15.50	R	Verify school stock; otherwise purchase.
HX711 debug board	SEN-13879; Q [19]	1	4.95	D	Early testing only; verify stock.
A4988 debug board	Adafruit 6109; Q [20]	1	6.95	D	Early testing only; verify stock.
RTC debug board	DS3231, Adafruit 3013; Q [21]	1	17.50	D	Early testing only; verify stock.
MCU debug board	ESP32-S3 DevKit, SKU TBD; A	1	15.00	D	Early testing only; verify stock.
Stepper motor	17HS4401, vendor TBD; A	1	20.00	R	Verify E-Shop model; otherwise purchase.
BLE beacon	Minew E8, quote pending; A	1	25.00	R/C	Required by current BLE design.
12 V supply	Isolated adapter, rating TBD; A	1	20.00	R/C	Verify free supply/rating; otherwise purchase.
RTC cells/holder	Exact type TBD; A	1 lot	4.00	C	For time retention after power loss.
Collar/tag mount	Breakaway collar, SKU TBD; A	1 set	8.00	R/C	Safe mount required; collar for live-pet use.
Bowl/hopper	Food-contact materials, SKU TBD; A	1 set	20.00	R	Food-safe containers required.
Auger/chute	Team CAD, vendor TBD; A	1 set	20.00	R	Verify fabrication/material quote.
Frame/enclosure	Team CAD, materials TBD; A	1 lot	25.00	R/C	Support/guards required; simplify exterior.
Coupler/fasteners	Dimensions TBD; A	1 lot	10.00	R	Verify E-Shop stock; otherwise purchase.
Calibration and test food	Reference masses and test food; A	1 lot	10.00	D	Test supplies; verify availability.
Non-PCB subtotal			366.50		
Final MCU modules	ESP32-S3-WROOM-1-N8R8; A	2	12.00	R+S	One required + one spare; verify exact E-Shop stock.
Final RTC ICs	DS3231MZ+; A	2	20.00	R+S	One required + one spare; verify stock.
Final weighing ADCs	HX711; A	2	4.00	R+S	One required + one spare; verify stock.
Final driver ICs	A4988SETTR-T; A	2	8.00	R+S	One required + one spare; verify stock.

Table 10 – continued

Item	Part / price basis	Qty.	Cost (\$)	Use	Source / purchase decision
Regulators	Buck and 3.3 V stages, BOM TBD; A	1 lot	12.00	R	Verify E-Shop ratings/footprints; otherwise purchase.
Protection and decoupling	Fuse, protection, resistors, capacitors; A	1 lot	12.00	R	Verify lab/E-Shop stock; otherwise purchase.
Connectors and indicators	Connectors, LEDs, passives; A	1 lot	15.00	R/C	Verify stock; optional LEDs may be omitted.
PCB fabrication	Course PCB order, quote pending; A	1 lot	25.00	R	Separate course PCB allocation; verify quote.
Parts and fabrication total			474.50		
Breadboard / hookup wires	Lab supply, per TA	1 lot	0.00	D	Confirmed supplied; not a purchase.

The four debug boards total \$44.40 and support early testing; a verified custom PCB may remove some of this need. The second MCU/RTC/HX711/A4988 set is a \$22 spare allowance. The final PCB uses DS3231M, so the DS3231 debug module requires firmware compatibility checks. The Pi 5 2 GB baseline needs an inference/memory test; 4 GB adds \$32.50 [14, 22]. No spare or debug purchase is automatically required for the final product.

3.1.3 Total Cost and Budget

Table 11: Conservative estimate before school-stock confirmation.

Category	Cost (\$)
Engineering labor	44,100.00
Parts and fabrication	474.50
Tax/shipping allowance	40.00
Contingency (15% of parts/fabrication)	71.18
Non-labor total	$585.68 + C_{\text{shop}}$
Grand total	$44,685.68 + C_{\text{shop}}$

Machine-shop charges C_{shop} require a quote; materials are already in the BOM. The course budget is \$150 for three students [13]; labor is an academic estimate and course PCB fabrication is separate per the TA. Without confirmed free parts, the non-PCB subtotal is \$366.50 before tax/shipping. With no borrowing available, the current all-purchase plan exceeds the course budget. Verify school stock and final quotes before ordering; paid PCB components still count unless supplied free. If necessary, reduce debug/spare purchases or revise the design. Personal purchases are not reimbursed [13]. A course-funded order must satisfy

$$C_{\text{course}} = \sum_{j \in \text{paid items}} n_j p_j + C_{\text{charged tax/shipping}} \leq \$150.$$

3.2 Schedule

The plan follows the Fall 2026 calendar [23]. Proposed leads are JP (Junwei Pan): Communications, network and PCB design; JZ (Jiaao Zhang): vision/interface and PCB design; ZQ (Ziteng Qian): mechanics/weighing and PCB design. All members handle integration and submissions; the team contract takes precedence.

Table 12: Implementation schedule and course milestones.

Week	Milestone	Task	Lead
Sep. 28	PCB review Sep. 29; PCBWay Sep. 30; document/sign-up Oct. 2	Finalize interfaces, R&V, cost and review materials.	JP: PCB; JZ: Sections 3–4; ZQ: mechanics
Oct. 5	Design review Oct. 5–7; PCBWay Oct. 8	Resolve feedback; verify E-Shop stock and purchase hardware; begin module tests.	JP: control; JZ: vision; ZQ: sourcing
Oct. 12	Breadboard demo Oct. 12–14; teamwork Oct. 14; PCBWay Oct. 15; shop revisions Oct. 16	Demonstrate sensing/control; verify rails, motor disable and scale; freeze mechanics.	JP: power/RTC; JZ: UART; ZQ: dosing
Oct. 19	Internal assembly gate	Bring up PCB; assemble guards, camera and bowl; measure fine-burst/in-flight mass.	JP: PCB; JZ: interface; ZQ: assembly
Oct. 26	Final JLCPCB order Oct. 29	Run full meal cycle; test reset, jam and offline operation; submit revisions if needed.	All; subsystem leads
Nov. 2	Progress demo Nov. 2–4; individual reports Nov. 4	Demonstrate integration and report measured results/failures.	All
Nov. 9	Contract assessment Nov. 13	Run HLR 1–4 and fault trials; correct failures.	JP: control; JZ: classification; ZQ: dose
Nov. 16	Mock demo at TA meeting; mock presentations Nov. 19–20	Rehearse, retest and freeze verified prototype.	All
Nov. 23	Fall break Nov. 23–27	Draft paper/video from saved results; no critical lab work.	All
Nov. 30	Final demos Nov. 30–Dec. 2; presentations Dec. 3–4; video Dec. 4	Complete demonstration, presentation and video.	All
Dec. 7	Paper Dec. 9; checkout Dec. 10, 3–4:30 p.m.; notebooks Dec. 10	Submit documentation, archive files and clean station.	All

PCB audits/orders close at 4:45 p.m.; listed document submissions close at 11:59 p.m. Verification retains 9/10 dosing, 27/30 classification, 18/20 reporting and HLR 4 idle/RTC criteria. Reconcile the 1.0 g peak weighing limit with the 0.5 g tolerance assumption.

4 Discussion of Societal Impact, Engineering Standards, Ethics, and Safety Considerations

4.1 Societal Impact

The feeder supports scheduled portions and remote meal monitoring for owners with irregular schedules or limited mobility. Local control maintains feeding during Internet loss when power and sensors remain functional. Cost, maintenance and Internet access may limit adoption. The owner remains responsible for diet and daily care; a 20 g test portion is not a dietary recommendation, and food removal does not prove ingestion or diagnose illness.

Measured deficit filling may reduce food waste. Washable, replaceable parts support cleaning and repair; electronics, plastics, batteries and Pi idle power still have environmental costs. Performance must be tested across food types, lighting and household conditions. The single-pet system cannot prevent other animals from eating leftovers; deployment requires local electrical, radio, privacy and food-contact review.

4.2 Engineering Standards

These references guide design; the assembled prototype is not product-certified.

Reference	Application
IEEE 802.11 [24, 25]	ESP32-S3 b/g/n Wi-Fi; follow vendor antenna guidance and test notification/network loss.
Bluetooth LE [26, 25]	Beacon/scan interoperability; RSSI is a coarse proximity cue, not authenticated identity.
TLS; RFC 8446 [27]	Validate server certificates; RFC 8446 applies if TLS 1.3 is negotiated.
UL/IEC 62368-1 [28]	Reference for external supplies and computing electronics; select rated isolated adapters and apply energy-hazard protection principles.
21 CFR 177.1520 [29]	Applies to selected olefin food-contact materials; obtain actual supplier/use documentation.
IEEE/ACM and course guidelines [30, 31, 32, 33]	Safety, privacy, honest results, attribution and laboratory procedures.

4.3 Ethics

IEEE commitments 1 and 5 and ACM principles 1.2, 1.3, 1.6 and 2.9 guide safety, honest reporting, privacy and security [30, 31]. Report dose errors, missed trials and sensor failures; do not label unavailable monitoring as a normal meal. Cite external code/models and comply with licenses. Validate negative examples, occlusion and lighting variation.

Images remain local and are not saved/uploaded by default. Development recording requires consent, restricted access and deletion after evaluation. Keep credentials out of code/logs; remote records contain only event labels, timestamps and masses.

Use mock targets and test pellets for routine verification. Live-pet testing requires owner consent and consultation with course staff about animal-care review. Never deprive food, force positioning or expose an animal to machinery; stop if distressed. The development prototype must not be an unattended animal's sole food source.

4.4 Safety and Verification

Complete course safety training and work with at least two people [33]. Only isolated low-voltage DC enters the feeder. Planned controls must be verified before use.

Hazard	Control	Verification
Auger/pinch	Secure guards; power off for jams; 15 s motor-on limit per meal; disable on faults.	Check reach paths with mocks and inject jam/timeout.
Electrical/heat	Fuse, reverse-polarity protection, insulation and strain relief; initial phase limit ≤ 1.2 A; local bulk capacitance ≥ 100 μ F [9].	Check rails/current setting and temperature during repeated 15 s operation.
Wrong dose	Fill measured deficit; reject invalid/unstable mass; record meal transactions.	Test residual food, no-flow, resets and 20 g $\pm$ 2 g dosing.
Food/tag/cells	Washable documented food-contact parts, smooth edges, secured beacon and inaccessible cells.	Inspect materials, cleaning access and retention; never charge primary cells.
Commands/link	MCU alone controls motor; validate packets, reject stale data and protect configuration/TLS.	Inject malformed frames/link loss; mark vision unavailable after 2 s.

Unresolved design issues. Persist an in-progress meal before motor enable so a reset cannot repeat a partial dispense; recovery should disable feeding pending inspection. Verify hardware default-disable behavior. Guards are specified, but no lid interlock is committed; assess one if opening the hopper exposes the auger. The 1.9 g overshoot budget has only 0.1 g margin; resolve the 1.0 g peak versus 0.5 g assumed weighing error using measurements.

4.5 Operating Procedure

1. Inspect guards, wiring, supplies, bowl and cells; keep animals away during bench tests.
2. With motor disconnected, use current-limited power to verify rails/default disable; power off before connecting the motor and setting its current limit.
3. Calibrate and dry-test with discarded pellets. Verify jam/timeout disable within 100 ms of detection; measure detection latency separately.
4. Supervise powered tests. Stop for abnormal heat, smell, smoke, exposed motion or faults.
5. Disconnect both supplies before clearing jams or cleaning; wait for motion to stop, keep liquid away from electronics and retest after repairs.

References

- [1] American Animal Hospital Association, “2021 AAHA nutrition and weight management guidelines for dogs and cats,” *Journal of the American Animal Hospital Association*, vol. 57, no. 4, pp. 153–178, 2021, doi: 10.5326/JAAHA-MS-7232. [Online]. Available: <https://www.aaha.org/resources/2021-aaha-nutrition-and-weight-management-guidelines/home/>
- [2] Analog Devices, “DS3231M: ± 5 ppm, I^2C real-time clock,” rev. 7, Mar. 2015. [Online]. Available: <https://www.analog.com/media/en/technical-documentation/data-sheets/DS3231M.pdf>

- [3] Minew, “E8 asset tracking tag,” product specification. [Online]. Available: <https://www.minew.com/product/e8-asset-tracking-tag/>. Accessed: Sep. 18, 2026.
- [4] Raspberry Pi Ltd., “Raspberry Pi Camera Module 3 product brief,” June 2024. [Online]. Available: <https://datasheets.raspberrypi.com/camera/camera-module-3-product-brief.pdf>
- [5] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, “MobileNetV2: Inverted residuals and linear bottlenecks,” in *Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition*, 2018, pp. 4510–4520, doi: 10.1109/CVPR.2018.00474.
- [6] SparkFun Electronics, “Load cell - 5 kg, straight bar (TAL220B),” SEN-14729. [Online]. Available: <https://www.sparkfun.com/load-cell-5kg-straight-bar-tal220b.html>. Accessed: Oct. 2, 2026.
- [7] Avia Semiconductor, “HX711: 24-bit analog-to-digital converter (ADC) for weigh scales,” datasheet. [Online]. Available: https://cdn.sparkfun.com/datasheets/Sensors/ForceFlex/hx711_english.pdf. Accessed: Sep. 18, 2026.
- [8] OMC Corporation, “17HS4401 bipolar stepper motor specification.” [Online]. Available: <https://www.omc-stepperonline.com/download/17HS4401.pdf>. Accessed: Sep. 18, 2026.
- [9] Allegro MicroSystems, “A4988 DMOS microstepping driver with translator and overcurrent protection,” 4988-DS, rev. 8, Apr. 2022. [Online]. Available: <https://www.allegromicro.com/-/media/files/datasheets/a4988-datasheet.pdf>
- [10] Pololu Corporation, “A4988 stepper motor driver carrier.” [Online]. Available: <https://www.pololu.com/product/1182>. Accessed: Sep. 18, 2026.
- [11] Raspberry Pi Ltd., “Getting started: Power supply.” [Online]. Available: <https://www.raspberrypi.com/documentation/computers/getting-started.html#power-supply>. Accessed: Sep. 18, 2026.
- [12] University of Illinois Urbana-Champaign, Grainger College of Engineering, “Electrical engineering: Post-graduation success,” 2024–2025 outcomes. [Online]. Available: <https://grainger.illinois.edu/academics/undergraduate/majors-and-minors/electrical-engineering>. Accessed: Oct. 2, 2026.
- [13] University of Illinois Urbana-Champaign, “ECE 445: Getting parts for your project.” [Online]. Available: <https://courses.grainger.illinois.edu/ece445/lab/getting-parts.asp>. Accessed: Oct. 2, 2026.
- [14] PiShop US, “Raspberry Pi 5 2GB, SC1110.” [Online]. Available: <https://www.pishop.us/product/raspberry-pi-5-2gb/>. Accessed: Oct. 2, 2026.
- [15] PiShop US, “Raspberry Pi Camera Module 3.” [Online]. Available: <https://www.pishop.us/product/raspberry-pi-camera-module-3/>. Accessed: Oct. 2, 2026.
- [16] PiShop US, “Raspberry Pi 27W USB-C power supply, SC1153.” [Online]. Available: <https://www.pishop.us/product/raspberry-pi-27w-usb-c-power-supply-white-us/>. Accessed: Oct. 2, 2026.
- [17] PiShop US, “Raspberry Pi Active Cooler, SC1148.” [Online]. Available: <https://www.pishop.us/product/raspberry-pi-active-cooler/>. Accessed: Oct. 2, 2026.

- [18] PiShop US, “Camera cable for Raspberry Pi 5.” [Online]. Available: <https://www.pishop.us/product/camera-cable-for-raspberry-pi-5/>. Accessed: Oct. 2, 2026.
- [19] SparkFun Electronics, “Load cell amplifier, HX711, SEN-13879.” [Online]. Available: <https://www.sparkfun.com/sparkfun-load-cell-amplifier-hx711.html>. Accessed: Oct. 2, 2026.
- [20] Adafruit Industries, “A4988 stepper motor driver breakout, product 6109.” [Online]. Available: <https://www.adafruit.com/product/6109>. Accessed: Oct. 2, 2026.
- [21] Adafruit Industries, “DS3231 Precision RTC Breakout, product 3013.” [Online]. Available: <https://www.adafruit.com/product/3013>. Accessed: Oct. 2, 2026.
- [22] PiShop US, “Raspberry Pi 5 4GB, SC1111.” [Online]. Available: <https://www.pishop.us/product/raspberry-pi-5-4gb/>. Accessed: Oct. 2, 2026.
- [23] University of Illinois Urbana-Champaign, “ECE 445 course calendar,” Fall 2026. [Online]. Available: <https://courses.grainger.illinois.edu/ece445/calendar.asp>. Accessed: Oct. 2, 2026.
- [24] IEEE, *IEEE Standard for Information Technology—Telecommunications and Information Exchange between Systems—Local and Metropolitan Area Networks— Specific Requirements—Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*, IEEE Std 802.11-2024. [Online]. Available: <https://standards.ieee.org/ieee/802.11/10548/>. Accessed: Oct. 2, 2026.
- [25] Espressif Systems, “ESP32-S3-WROOM-1 and ESP32-S3-WROOM-1U datasheet,” ver. 1.8. [Online]. Available: https://documentation.espressif.com/esp32-s3-wroom-1_wroom-1u_datasheet_en.pdf. Accessed: Oct. 2, 2026.
- [26] Bluetooth SIG, “Core Specification 5.0 (amended),” amendment effective July 1, 2024. [Online]. Available: <https://www.bluetooth.com/specifications/specs/core-specification-amended-5-0/>. Accessed: Oct. 2, 2026.
- [27] E. Rescorla, “The Transport Layer Security (TLS) protocol version 1.3,” IETF RFC 8446, Aug. 2018, doi: 10.17487/RFC8446. [Online]. Available: <https://datatracker.ietf.org/doc/html/rfc8446>.
- [28] UL Solutions, “Power supply test and certification.” [Online]. Available: <https://www.ul.com/services/power-supply-test-and-certification>. Accessed: Oct. 2, 2026.
- [29] U.S. Food and Drug Administration, “21 CFR 177.1520—Olefin polymers,” *Electronic Code of Federal Regulations*. [Online]. Available: <https://www.ecfr.gov/current/title-21/chapter-I/subchapter-B/part-177/subpart-B/section-177.1520>. Accessed: Oct. 2, 2026.
- [30] IEEE, “IEEE Code of Ethics,” sec. 7.8. [Online]. Available: <https://www.ieee.org/about/corporate/governance/p7-8.html>. Accessed: Oct. 2, 2026.
- [31] Association for Computing Machinery, “ACM Code of Ethics and Professional Conduct,” 2018. [Online]. Available: <https://www.acm.org/code-of-ethics>. Accessed: Oct. 2, 2026.
- [32] University of Illinois Urbana-Champaign, “ECE 445 ethical guidelines.” [Online]. Available: <https://courses.grainger.illinois.edu/ece445/guidelines/ethical-guidelines.asp>. Accessed: Oct. 2, 2026.
- [33] University of Illinois Urbana-Champaign, “ECE 445 safety guidelines.” [Online]. Available: <https://courses.grainger.illinois.edu/ece445/guidelines/safety.asp>. Accessed: Oct. 2, 2026.