PALESTINE POLYTECHNIC UNIVERSITY

College of IT And Computer Engineering

Computer Engineering Department

Graduation Project

Smart Pet Care

Group

Nidal Baradee / 211001

Mohammad Bahar / 211053

Supervisor

Dr. Radwan Tahboub

2025 - 2026

Table of Contents

Abstract

Cats require regular care and attention, including timely provision of food and water, which can be challenging for owners especially when they are away for long periods — often leading to concern about the pet's well-being. In this project, meals are scheduled based on data entered by the owner through a mobile application and are automatically dispensed using a Real-Time Clock. The device fills the food bowl according to the cat's predefined feeding schedule, making the feeding process more convenient and reducing the owner's burden.

The system also includes an automated water supply to ensure proper hydration. In addition, a water drainage system is implemented and used when needed to clean the water bowl from impurities or accumulated debris, helping to maintain hygiene and water quality. To promote physical activity, a laser play feature is integrated into the system.

Live video monitoring is enabled through an onboard camera, allowing the owner to remotely observe the pet at any time. After providing global access via an render server, a security layer was added to block unauthorized viewing of the camera stream.. The system uses One-Time Password (OTP) verification along with short-lived JSON Web Token (JWT) sessions and session revocation support, ensuring that only the authorized owner can access the live feed and device functions, even if the stream link is obtained directly.

Furthermore, the system integrates a lightweight rule-based AI assistant implemented with Firebase Cloud Functions. Results and validation: the hardware prototype and the mobile application were fully implemented and integrated, and the complete system was tested in real usage scenarios; the scheduled feeding, automated water management (refill/drain), laser activity feature, and secured camera access operated successfully as intended, confirming the feasibility and reliability of the proposed solution.

Overall, the project introduces complete solution integrating feeding, hydration, water management, entertainment, and secure camera monitoring to ensure pet comfort and owner confidence.

Keywords**:** Smart Pet-Care System, Automated Feeding, Remote Monitoring, Secure Access, IoT

الملخص

القطط تحتاج إلى رعاية واهتمام منتظمين، بما في ذلك توفير الطعام والماء في الوقت المناسب، وهذا قد يكون صعبًا على المالكين خصوصًا عند الغياب لفترات طويلة، مما يسبب قلقًا على سلامة الحيوان الأليف. في هذا المشروع، يتم جدولة الوجبات بناءً على البيانات التي يُدخلها المالك عبر تطبيق للهاتف المحمول، ثم يتم توزيع الطعام تلقائيًا باستخدام ساعة وقت حقيقي .يقوم الجهاز بتعبئة وعاء الطعام وفق جدول التغذية المحدد مسبقًا للقط، مما يجعل عملية الإطعام أكثر سهولة ويقلل العبء على المالك.

يتضمن النظام أيضًا تزويدًا آليًا بالماء لضمان الترطيب المناسب. بالإضافة إلى ذلك، تم تنفيذ نظام لتصريف الماء يُستخدم عند الحاجة لتنظيف وعاء الماء من الشوائب أو الأوساخ المتراكمة، مما يساعد على الحفاظ على النظافة وجودة الماء. ولتعزيز النشاط البدني، تم دمج ميزة اللعب بالليزر ضمن النظام.

كما يوفّر النظام مراقبة فيديو مباشرة عبر كاميرا مدمجة، مما يتيح للمالك متابعة الحيوان الأليف عن بُعد في أي وقت. وبعد تفعيل الوصول العالمي عبر خادم وسيط خارجي تم تطبيق آلية أمان لمنع الوصول غير المصرح به إلى بث الكاميرا. يستخدم النظام تحقق كلمة مرور لمرة واحدة مع جلسات قصيرة العمر ودعم إلغاء الجلسات لضمان أن المالك المخوّل فقط يستطيع الوصول للبث المباشر ووظائف الجهاز، حتى لو تم الحصول على رابط البث بشكل مباشر.

علاوةً على ذلك، يدمج النظام مساعدًا ذكيًا خفيفًا قائمًا على قواعد البيانات النتائج والتحقق **:**تم تنفيذ النموذج والتطبيق الهاتفي بشكل كامل ودمجهما معًا، وتم اختبار النظام في سيناريوهات استخدام فعلية؛ وقد عملت وظائف جدولة الإطعام، وإدارة الماء تلقائيًا (التعبئة/التصريف)، وميزة الليزر، وآلية الوصول الآمن للكاميرا بنجاح كما هو مخطط، مما يؤكد قابلية تطبيق الحل المقترح وموثوقيته.

بشكل عام، يهدف هذا المشروع إلى تقديم حل متكامل للعناية الذكية بالحيوانات الأليفة يجمع بين الإطعام، والترطيب، وتنظيف وإدارة الماء، ودعم النشاط، والمراقبة الآمنة عن بُعد لضمان راحة الحيوان الأليف وطمأنينة المالك.

.

CHAPTER ONE: INTRODUCTION

1.1 Preface

Pet ownership brings joy, but also responsibility especially in ensuring regular feeding and hydration.

This project introduces Smart Pet Care, an intelligent, automated system designed to help cat /dog owners provide optimal care for their pets, even when away from home. Through a combination of IoT devices, sensors, automation, and AI, the system simplifies essential pet care tasks.

1.2 Problem Statement

In an ideal scenario, every pet would be fed on time and provided with fresh water continuously. Such an environment would eliminate the risks associated with irregular feeding and dehydration.

In reality, many pet owners struggle with maintaining these routines. Busy schedules, unexpected delays, and inconsistent monitoring result in irregular feeding times and insufficient hydration. Traditional pet care devices usually work in a limited way, relying mostly on fixed timers without responding to real-time changes. In addition, most of these systems do not provide a proper method to keep water fresh. Over time, water bowls may become dirty, which can create hygiene concerns for pets. To address this issue, a water bowl draining feature was added to help ensure the water stays is refreshed and clean regularly.

These limitations may negatively affect the pet's wellbeing and can also increase the owner's worry. Smart Pet Care was designed to overcome these challenges by offering an integrated and automated solution for essential pet-care tasks. Through real-time sensor monitoring, automatic control, scheduled functions, and remote access using a mobile application, the system helps provide reliable and consistent care, closer to the ideal routine for both pets and owners.

1.3 Project Aims and Objectives

Main Aim:

To design and build an automated and intelligent pet care system that delivers scheduled feeding , hydration, , and enables real-time monitoring and remote control through an AI-supported mobile application.

Key Objectives

- Develop an integrated smart pet care solution that helps pet breeder keeps their pets under monitoring
- Provide automated feeding by dispensing meals at predefined times with accurate portion by using mobile application
- Ensure continuous hydration by smart water system that water drop in bowl if it is empty
- Design and integrate a drainage system to empty the water bowl when required
- Add laser pointer module to encourage play and physical activity.
- Implement a secure live camera feature so owners can view real-time video of their pets remotely, at any time.
- Implement an alert and notification system to inform the owner when food or water supplies are low or empty.

1.4 Project Requirements (functional and non-functional)

- Functional Requirements:

- **Automated Food Dispensing:**
  - Allow user to create and edit meals depends on amount and time
  - Monitoring weight and food level using ultrasonic and load cell.
  - Allow manual feeding by mobile application

- **Automated Water Supply:**
  - monitoring water levels using ultrasonic sensors.
  - Water drops automatically when bowl is empty

- **Live Camera Monitoring:**
  - Providing real-time video streaming through the ESP32-CAM module, allowing the user to view real-time video inside the mobile application

- **Pet Profile Management:**
  - The system shall store and manage pet profiles in Firebase, including (at minimum): name, age, weight, pet type, and special notes.

- **Mobile Application Interface:**
  - Enable remote control for whole the system
  - Display sensors readings
  - Offer rule based ai

- **Entertainment System (Laser Play):**
  - Enable an automated laser play mode using a laser module with servo

- **Water Bowl Draining System**
  - Allow the user to manually activate the water draining process through the mobile application.

- **Rule-Based Artificial Intelligence System:**
  - Allow users to ask questions about whole system
  - The system shall compute daily energy needs by calculating **RER** then converting to **MER** using simplified factors (life stage / activity), and then generate recommended meals (times + portions).

  - Provide clear recommendations + alerts [3]

- ## Non-Functional Requirements:

  - **Performance:**
    - The system shall process sensor updates and execute control commands in near real time to ensure timely feeding, refilling, and notifications.
  - **Reliability:**
    - The system shall operate reliably with a target operational effectiveness of at least 90%, and include basic backup considerations for power interruptions (e.g., safe state, recovery behavior).
  - **Scalability:**
    - The design shall support adding new sensors/modules later with minimal changes to the core architecture.
  - **Usability:**
    - The mobile application shall be easy to navigate for non-technical users, with clear statuses, simple controls, and readable alerts.
  - **Security:**
    - Remote access shall be protected using short-lived JWT sessions and OTP-based authentication to reduce unauthorized access risk to the camera stream and device functions.

1.5 System Description

The Smart Pet Care system is an intelligent pet-care solution designed around a microcontroller, which acts as the main control unit of the project. The system combines several components that work together to provide a complete care experience. Figure 1.1 illustrates the overall project structure and how the different modules are connected:

- **Automated Pet Feeder:** Dispenses measured food portions either through an AI-based schedule, or through a manual feeding .Food availability is monitored using a load cell and ultrasonic sensors to ensure refilling when needed.
- **Automated Water Supply:** Ensures that the pet always has enough water by continuously checking the water level and if the water low the water drops automatically
- **Live Camera Monitoring:** Provides real-time video streaming so owners can see their pets remotely at any time.
- **Mobile Application with AI Features:** Offers a user-friendly interface that allows owners to schedule meals, monitor system status, and control the device functions. AI-supported tools also provide useful pet-care
- **Water Bowl Draining System:** Enables the owner to empty and refresh the water bowl manually through the mobile application.
- **Pet Profile Management (Firebase Cloud):** Stores detailed pet information in the Firebase Realtime Database. Each pet has a personal profile including name, age, breed, weight, type, and notes. This information allows the system to customize feeding portions, schedules, and care suggestions. Owners can update the profile directly from the mobile app.
- **Entertainment Laser Play Module:** Supports pet activity and engagement through an interactive laser pointer that moves in programmed patterns using a servo motor.
- **Rule-Based Artificial Intelligence System:** Includes an in-app assistant that responds to user questions about the system and general pet care. It also generates personalized meal plans using rule-based energy calculations (RER and MER) based on pet data stored in Firebase for execution and monitoring.

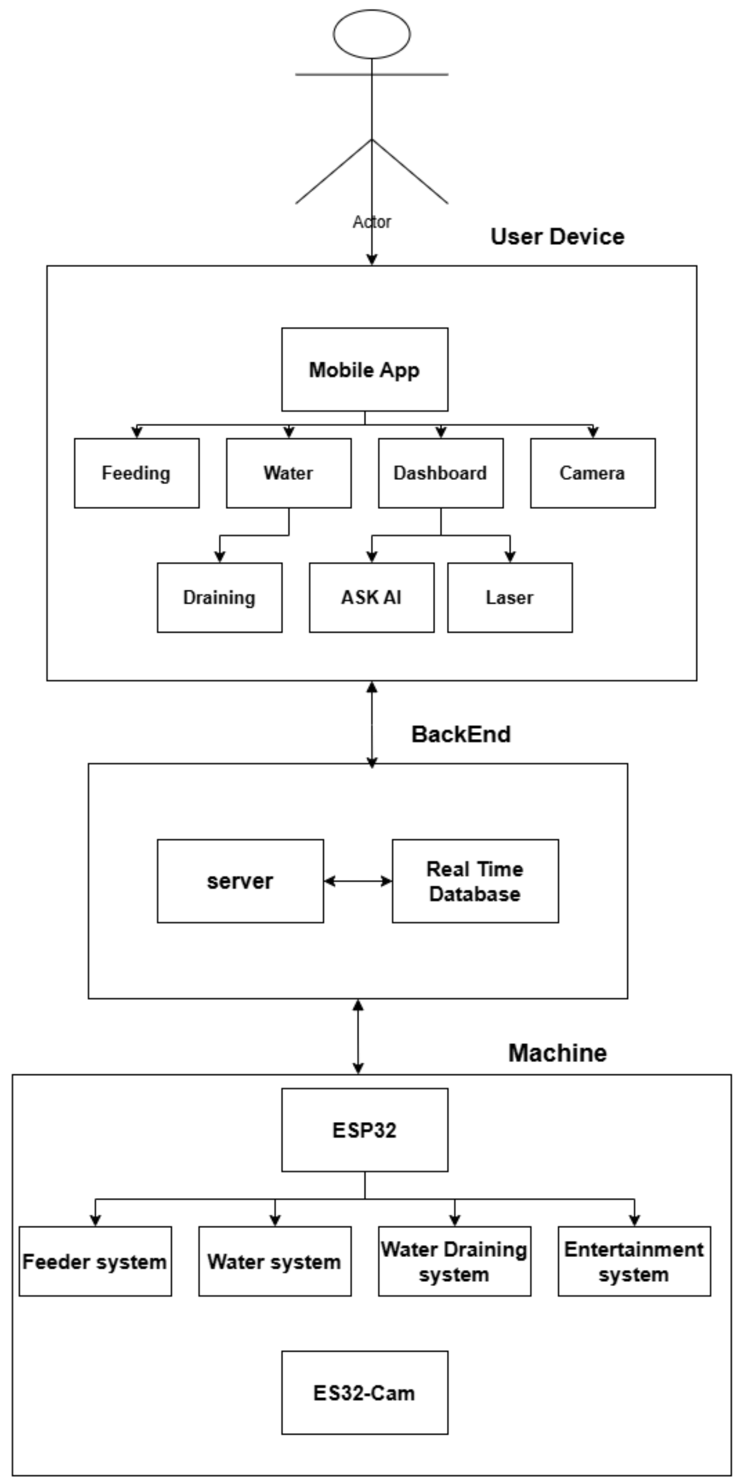

Figure 1.1 : Project Structure

1.6 Project Limitations/Constraints

- **Platform Specificity:**
  The system is mainly designed for smart pet care purposes. Extending it to support different types of animals may require further modifications in both hardware and software, which could add extra time and complexity to the development process.
- **Dependence on Network Connectivity:**
  Remote monitoring and control features depend strongly on having a stable Wi-Fi connection. If the internet connection is weak or interrupted, the system may not perform as expected, and real-time updates could be affected.
- **Environmental Considerations:**
  The project is intended primarily for indoor environments. Using it outdoors or in harsh conditions would require additional protection and may reduce the overall reliability of the system under extreme weather or environmental factors.

1.7 Project Schedule

Table 1.1 presents the project timeline, showing how time was distributed across each stage, including idea development, research, design, and documentation.

Table 1.1: Time Frame

stage	Weeks
Selection of project Idea	5 Weeks
Describe idea	1 Week
Idea Analyses	1 Week
Literature review	1 Week
Design	1 Week
Documentation	1 Wees
Machine / Hardware Fabrication	6 Weeks
Programming / Coding	3 Weeks
Integration of Hardware & Software	1 Week
Testing and Debugging	2 Week
Final Documentation	2 Week

1.8 Report Outline

- **Chapter One:** is an introduction and overview of the project. We discussed the project's limitations, objectives, and when it would work in its ideal form.
- **Chapter Two:** provides reviews of the theories used, as it uses air quality monitoring sensors and microcontrollers.
- **Chapter Three:** will discuss the system components, the various design options proposed, and the software needed to design the system also discuss about sensors
- **Chapter Four**: Focuses on the implementation and testing phase of the proposed system. It highlights development challenges, integration steps, validation methods, testing procedures, and includes the obtained results with discussion.
- **Chapter Five**: Concludes the project and outlines possible future work. It summarizes the overall outcomes, reviews the achieved goals, addresses system limitations, and suggests potential improvements and enhancements.

CHAPTER TWO: THEORETICAL BACKGROUND

2.1 Preface

This chapter outlines the background knowledge and related studies that support the Smart Pet Care project. It introduces the main technologies used in the system—IoT connectivity, mobile application development, and rule-based decision logic—and then reviews similar solutions from the literature. The purpose is to position Smart Pet Care within existing work and clarify the specific additions and improvements introduced by this project.

2.2 Theoretical Background Concepts

A core element in Smart Pet Care is a **rule-based decision system** designed to assist the owner through the mobile application. the system applies predefined rules to interpret real-time sensor readings together with stored pet information, then returns practical guidance and system-aware recommendations.

Internet of Things (IoT) Architecture

a. **Perception Layer:** includes the physical sensing and actuation components. In this project, sensing is performed using load cells, ultrasonic measurement, and liquid-level sensing, while actions are carried out using actuators such as servo motors, pumps, and valves. [7].

b. **Network Layer:** Responsible for communication between the device and cloud services. The embedded controller connects over Wi-Fi and pushes/receives updates to support real-time control. [7].

c. **Cloud Layer:** Data storage and synchronization are handled through Firebase Realtime Database, which supports real-time JSON data syncing across clients and device states without requiring a traditional custom backend server [7].

d. **Application Layer:** The user interacts through a mobile app built with Flutter, which provides the interface for scheduling, monitoring, notifications, and control features [7].

e. **Business Layer:** Captures how the system is used in real scenarios, such as scheduled feeding, continuous hydration, hygiene support (drainage), real-time alerts, and rule-based guidance that reduces owner workload and improves convenience[7].

2.2.1 AI-Powered Decision Making (Rule-Based Artificial Intelligence System)

Within Smart Pet Care, the AI element is implemented using a rule-based decision approach to support interactive assistance and system-aware recommendations inside the mobile application. The rule engine evaluates the current system context (sensor readings and device states) together with saved pet profile information to guide user decisions and automate selected behaviors.
It analyzes real-time IoT sensor data together with pet profile information (such as pet type and weight) to generate guidance and suggest appropriate actions.
In addition, a dedicated Cloud Function is responsible for generating a daily feeding plan.
Cloud Functions provide a serverless way to run backend logic in response to HTTPS requests or database events, which fits the needs of mobile-to-cloud AI requests and automated calculations in this project.
The daily feeding plan logic is based on standard veterinary energy estimation using **RER** and **MER**, where RER is commonly estimated using the $70 \times BW_{kg}^{0.75}$ form and MER is obtained by applying a multiplier that reflects the pet's condition and activity. The resulting daily calories are then converted into food grams using configured kcal-per-gram values for the selected diet.
[3].

Function 1: AskAi (Interactive Q&A)
This function enables direct interaction between the mobile application and the rule-based assistant. When the user submits a question, the function returns a structured JSON response that includes:

- A short answer text (guidance)
- Practical tips
- Detected intent category (e.g., feeding / hydration / status)
- Severity level (if relevant)
- Suggested actions
- A snapshot of key system readings (for example):
    - Food status/level
    - Water availability
    - Entertainment/activity state

Function 2: GenerateMealsAi (Meal Plan Generator)
This function is triggered manually from the mobile application (button-based). It follows a clear sequence:

1. Read pet profile data (type and weight) from the database.
2. Retrieve kcal-per-gram configuration for the selected food.
3. Compute daily energy needs using RER, then convert to MER using a simple multiplier approach.
4. Convert the resulting calories/day into food grams/day using the configured kcal-per-gram value.
5. Generate a simple daily Generates a simple, fixed daily feeding schedule and stores it in the feeding/meals

2.3 Literature Review

In this section we do not find a project in PPU like this or similar to it, so we research in another Universities and found two projects in Al-Najah University with same idea (Catcare) [1], (Purrfect House) [2] in this table 2.1: Comparing Projects shows the comparison between the projects.

Table 2.1 : Comparing Projects

Feature / Project	Smart Pet Care (Proposed)	CatCare	Purrfect House
Automated Feeding	✓ Scheduled + manual via mobile app	✓ Scheduled + manual feeding	✓ Scheduled feeding with appetite monitoring
Automated Water Supply	✓ Water level monitoring + auto refill	✓ Basic hydration monitoring with sensors	✓ Water level alerts and auto supply
Mobile Application	✓ Full control, AI support, and real time alerts	✓ Manual control and scheduling	✓ App-based control + WhatsApp alerts
AI Integration	✓ Rule-Based Artificial Intelligence System	✕ Not available	✕ Not available
Litter Box Automation	✕ Not included (future scope)	✓ Low-cost automatic litter box cleaning	✓ Self-cleaning litter box with notifications
Health Monitoring	✓ Age-based analysis and smart suggestions	✕ Not mentioned	✕ Not mentioned
Entertainment System	✓ Laser Play	✕ Not mentioned	✕ Not mentioned
Camera Monitoring	✓ LIVE Camera	✕ Not mentioned	✕ Not mentioned

Water Bowl Draining System	✓ Manual draining control via mobile app + automatic refresh cycle	✕ Not mentioned	✕ Not available
Pet Profiles (Cloud-Based)	✓ Firebase-based pet profile with age, weight, type, and health notes	✕ Not mentioned	✕ Not available

2.4 Summary

This chapter presents the theoretical foundation and prior research behind the Smart pet Care project, highlighting its innovative use of IoT and AI technologies for intelligent pet monitoring and care.

- **Internet of Things (IoT):** Utilizes sensors (load cells, ultrasonic, liquid-level), actuators (servo motors, water pumps), and microcontrollers (ESP32/Arduino) to automate feeding and hydration.
- **AI Integration:** Leverages the Rule-Based Artificial Intelligence System to answer user queries, provide personalized pet care advice, and suggest system adjustments
- **Literature Review:**

  Compares Smart Pet Care with two similar projects from Al-Naja University:

  - Cat Care [1].
  - Purrfect House [2].

  Smart Pet Care stands out by:

  - Providing personalized AI assistance.

CHAPTER THREE: SYSTEM DESIGN

3.1 Preface:

This chapter provides the system design of the Smart Pet Care project, covering the module design , hardware/software architecture, component communication, and the algorithms used for automated operation and remote management. It explains how the feeding, hydration, drainage, monitoring, and entertainment functions are implemented and integrated with the central controller, and why we chose these sensors

3.2 System Components and Design Alternatives :

The heart of the system is the microcontroller that coordinates all modules, including the feeder, water dispenser, water drainage system, entertainment system, Camera, rule-base intelligence system and mobile app. Our design required a microcontroller that supports:

- Wi-Fi connectivity for cloud and mobile app integration
- Sufficient GPIO pins for connecting various sensors and motors
- Real-time processing for scheduling tasks accurately
- Support for power-efficient operation
- Rule-Based Artificial Intelligence System
- Compatibility with Camera

We evaluated several options based on cost, performance, and compatibility, as shown in **Table 3.1: Controller Compare** below [8] [9]:

Table 3.1: Controller Compare

Feature	Selected: ESP32 Dev Board	Rejected: Arduino Uno
Feature		
Type	**Microcontroller**	**Microcontroller**
Processor	**Dual-core 240 MHz**	**Single-core 16 MHz**
Memory	**520 KB SRAM**	**2 KB SRAM**
Connectivity	**Built-in Wi-Fi and Bluetooth**	**No built-in Wi-Fi or Bluetooth**
GPIO Pins	**34**	**14**
Power Efficiency	**Supports deep sleep**	**No deep sleep**
Interface	**Digital, Analog, SPI, I2C, UART**	**Digital, Analog, SPI, I2C, UART**
Key Features	**Wireless communication, high processing speed**	**Basic microcontroller for simple tasks**
Final Decision	**Ideal for Smart pet system requirements**	**Not sufficient for multi-module control**

❖ **Why ESP32?**

- Has Wi -Fi and Bluetooth.
- Dual-core 240 MHz processor is suitable for multi-tasking across modules.
- Deep sleep mode for energy saving.
- Sufficient GPIO for all sensor and actuator connections.

Sensors and Actuators

We evaluated several options based on cost, performance, and compatibility to choice Sensors and Actuators as shown in Table 3.2 : Automated Feeder , Table 3.3 : Automated Water , Table 3.4 : Camera And Table 3.5 : Laser Module, Table 3.6 :Water Bowl Draining System

A. Automated Pet Feeder

Table 3.2: Automated Feeder

Component	Selected	Rejected	Reason
Ultrasonic Sensor [12],[13]	**HC-SR04**  • Trigger & Echo pins for easy interfacing • Low cost and easy to use • Compact and lightweight	**JSN-SR04T**  • Higher cost • Slower response compared to HC-SR04	**HC-SR04 is affordable, precise and faster in response.**
Load Cell [14]	**5KG HX711** 	**60KG Load Cell** 	**5KG is ideal for daily pet food portions. 60KG model is unnecessary and more expensive.**
Servo Motor [16] ,[17]	**MG996R**  • Torque: 9–11 kg·cm • Metal gears • Wide rotation • High performance and reliable	**SG90 Micro Servo**  • Torque: 1.8–2.5 kg·cm • Plastic gears • Limited rotation angle	**MG996R have larger Torque and better for tasks, SG90 is only for small light stuff**

RTC Module [10],[11]	**DS3231**  • Very high accuracy • Keeps time during power loss • Temperature compensated oscillator	**DS1307**  • Lower accuracy • No temperature compensation • Cheaper but less reliable	**DS3231 offers better accuracy and temperature compensation and The difference in price is not huge only 0.5 $.**
Vibration Motor [18],[19]	**Vibration Motor (2200–7200 RPM)**  • High vibration intensity • Strong torque suitable for notifications or physical feedback • Higher RPM (up to 7200 RPM)	**Typical Small Vibrator Motors**  • Lower RPM (typically 1000–3000 RPM) • Weak vibration strength • Not suitable for strong alerts or mechanical applications	**The FP-130 / Micro 030 Vibration Motor provides significantly stronger vibration, higher RPM, and better reliability, making it more suitable for pet alerts or mechanical feedback.**

B. Automated Water Supply

Table 3.3 : Automated Water

Component	Selected	Rejected	Reason
Ultrasonic Sensor [12],[13]	**HC-SR04**  • Trigger & Echo pins for easy interfacing • Low cost and easy to use Compact and lightweight	**JSN-SR04T**  • Higher cost Slower response compared to HC-SR04	**HC-SR04 is affordable, precise and faster in response.**
Solenoid Valve [4]	**12V Valve**  • reducing leakage risk • Delivers water in controlled way	**Gravity Valve**  • No sensors for control • Bad flow rate	**A solenoid valve enables precise, automated water control, while a gravity-based valve offers limited accuracy and no automation.**
Water Pump	**12V DC Pump**  • reliable and strong water flow • Long life	**5V DC Pump**  • Lower power and weaker flow • Short life	**The 12V pump provides stronger, more consistent performance, and long service life.**
Rain Detection Sensor [5]	**Simple Analog Rain Sensor**  • Detects via analog signal • Detects via digital signal		

C. Camera

Table 3.4 : Camera

Component	Selected	Rejected	Reason
Camera [6]	**Original ESP32 CAM Camera**  • External PSRAM for better image buffering and processing • Supports real-time video streaming • Built-in ESP32 Wi-Fi + Bluetooth	**Generic ESP32-CAM without PSRAM**  • Lower performance in image processing • Restricted resolution and frame rate • ESP32 microcontroller but limited memory (no PSRAM)	**The original ESP32-CAM with PSRAM provides higher memory capacity, flexible camera options, and supports low-light monitoring. The generic version is cheaper but less capable for real-time video.**

D. Entertainment System

Table 3.5 : Laser Module

Component	Selected	Rejected	Reason
Laser Module [20]	**1mW Red Laser Diode**  • Very low power output (≤1 mW) • Falls under **Class 2 laser → considered safe**	**10mW Green Laser**  • Power output (10 mW), falls under **Class 3B** • Potentially harmful to eyes with direct exposure	he **1 mW red laser** is **safer (Class 2)** and suitable for pet play without significant eye hazard, while the **10 mW green laser** is too powerful (**Class 3B**) and poses safety risks.
Servo Motor [16],[17]	**MG996R**  • Torque: 9–11 kg·cm • Metal gears • Wide rotation • High performance and reliable	**SG90 Micro Servo**  • Torque: 1.8–2.5 kg·cm • Plastic gears • Limited rotation angle	**MG996R have larger Torque and better for tasks, SG90 is only for small light stuff**

E. Water Bowl Draining System

Table 3.6 : Water Bowl Draining System

Component	Selected	Rejected	Reason
Solenoid Valve [4]	**Normally Closed 12V Valve**  • Prevents water leakage when powered off • Precise and controlled water release	**Gravity Valve**  • No electronic control supports • Inconsistent flow rate	**Solenoid valve ensures accurate, automated control, whereas gravity valves lack automation and precision.**
Water Pump	**Mini12V DC Pump**  • Strong and reliable water flow • Long service life	**Mini 5V DC Pump**  • Lower power and weaker flow • Short service life	**The 12V pump provides stronger, more consistent performance, and long service life.**
Rain Detection Sensor [5]	**Simple Analog Rain Sensor**  • Detects moisture/rain via analog signal • Easy integration with ESP32 • Low cost and low power		

Software Tools and Platforms:

In this table we will show Software Tools and Platforms and the reason to selected it Table 3.7 : Tools and Platforms

Table 3.7 : Tools and Platforms

Component	Purpose	Tool/Platform	Reason for Selection
Mobile App	Provides user interface for system control, scheduling, monitoring, and receiving alerts	Flutter	Cross-platform (iOS & Android), fast development, beautiful and customizable UI.
Cloud Backend	Central storage for data, device states, profiles, feeding and watering commands	Firebase Realtime Database	Real-time synchronization, JSON-based structure, easy integration with ESP32 and Flutter, no custom backend needed.
AI Assistant	Smart responses and user interaction	Rule-Based Artificial Intelligence System	Provides natural language interaction and intelligent suggestions.
Notifications System	Sends alerts and reminders (via mobile app notifications)	Firebase Cloud Messaging (FCM)	Free, reliable, real-time push notifications on all mobile devices.
Live Camera Streaming	Allows real-time viewing of the pet from any where	ESP32-CAM + Render server	Low-cost camera module; Render provides stable web streaming accessible from the mobile app.

3.3 Conceptual system description

1. Block Diagram

in figure 3.1: Block Diagram Show all main hardware modules and software modules(ESP32, feeder, water system, camera, entertainment system, mobile app, backend, server , Draining).

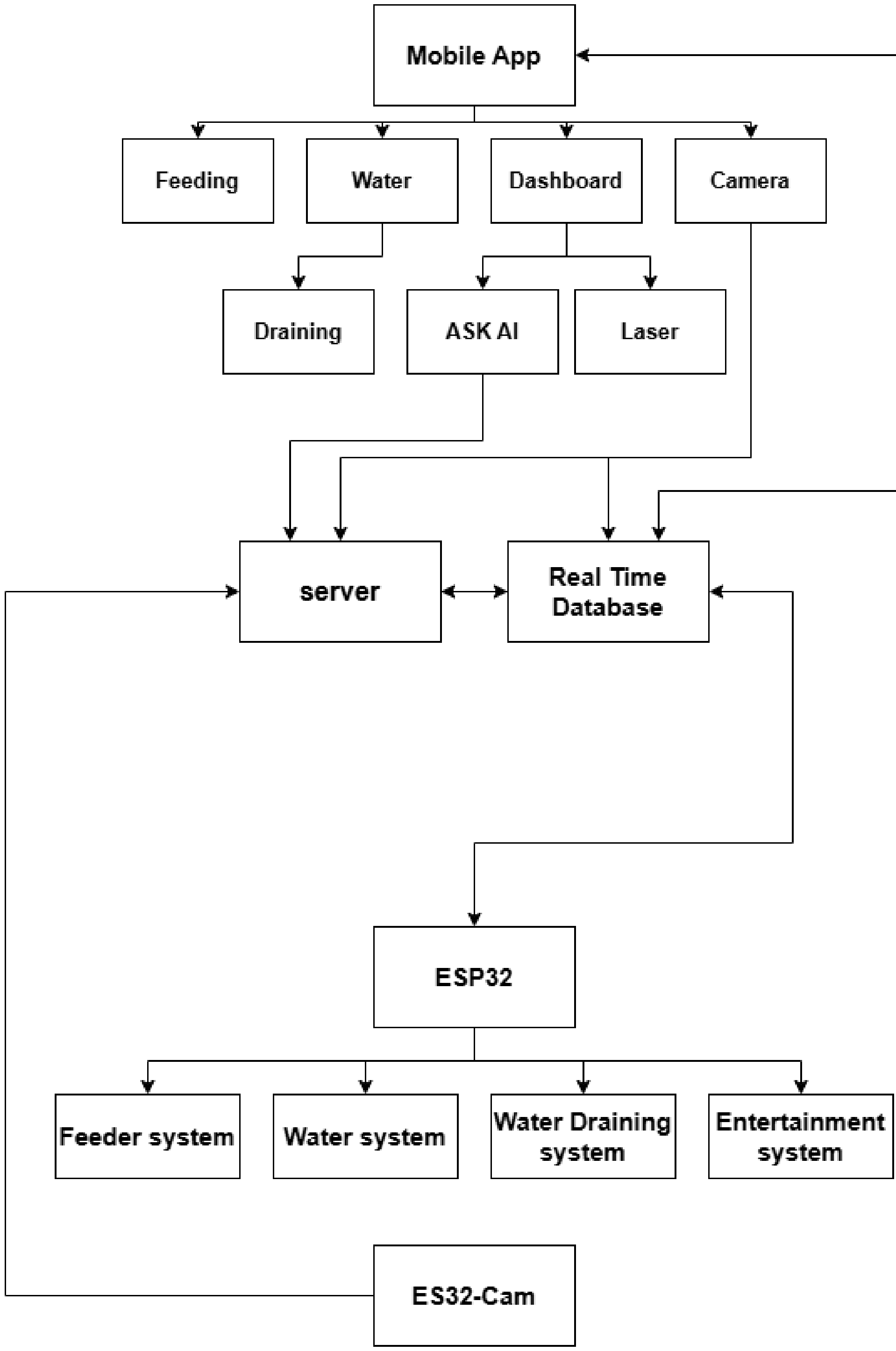

Figure 3.1 : Block Diagram

2. Conceptual Diagram

in figure 3.2 : Conceptual Diagram view of the system: pet care process, interaction between user, app, and hardware

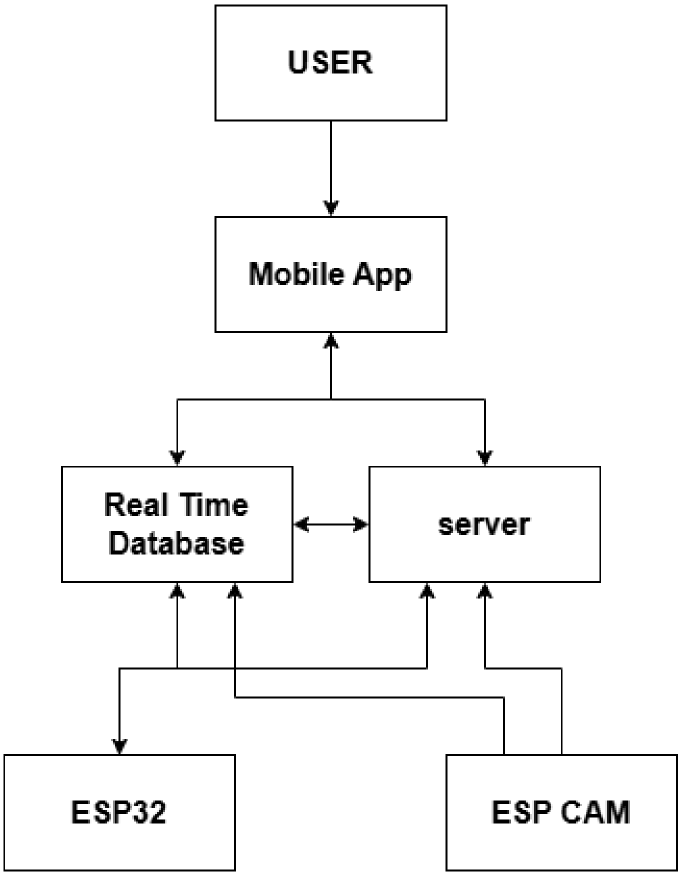

Figure 3.2 : Conceptual Diagram

3.4 Algorithms and Methodologies

- **Flow Charts**

a. Flow Chart for System:

the figure 3.3 : Chart System will show the logical process for system by check the food and water containers and the way to send alert .

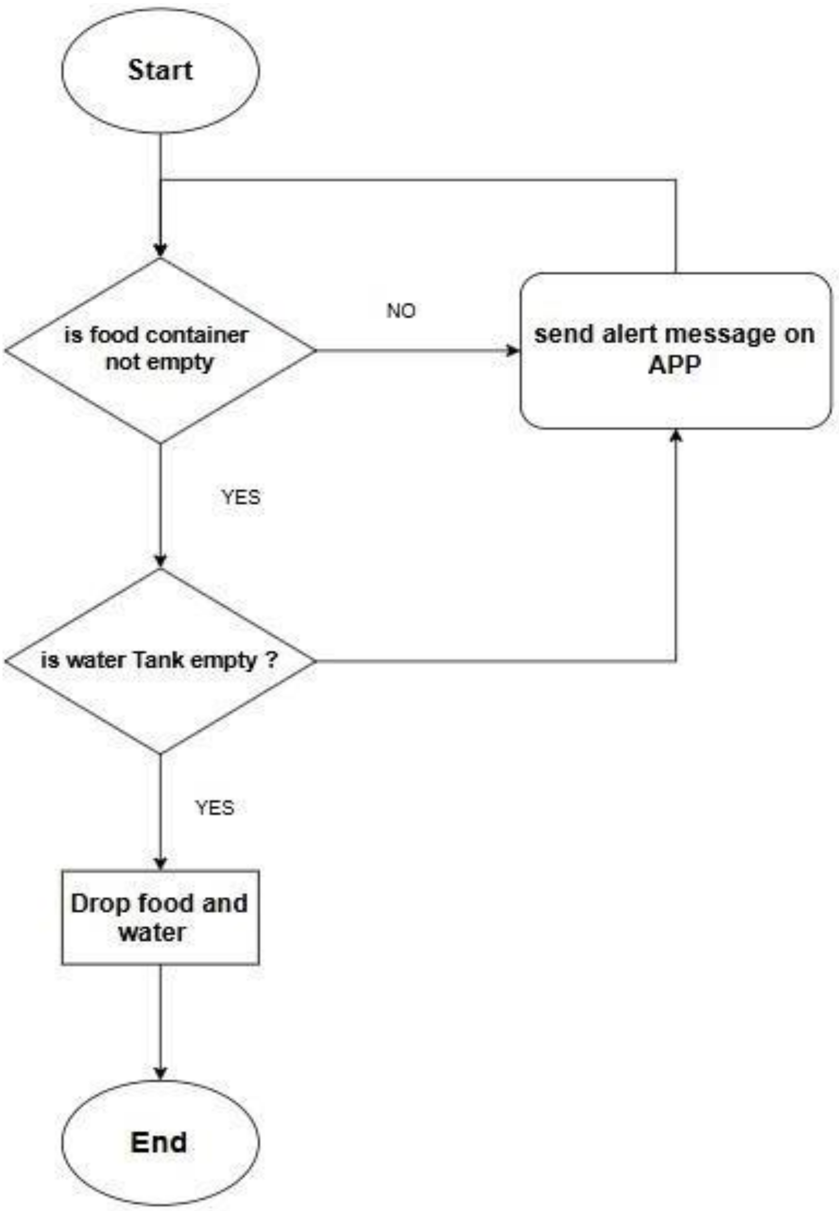

Figure 3.3 : Chart System

b. Flowchart for Feeding System :

the figure 3.4 : Chart Feeding will show the logical process for feeder system by check the food container and the way to send alert

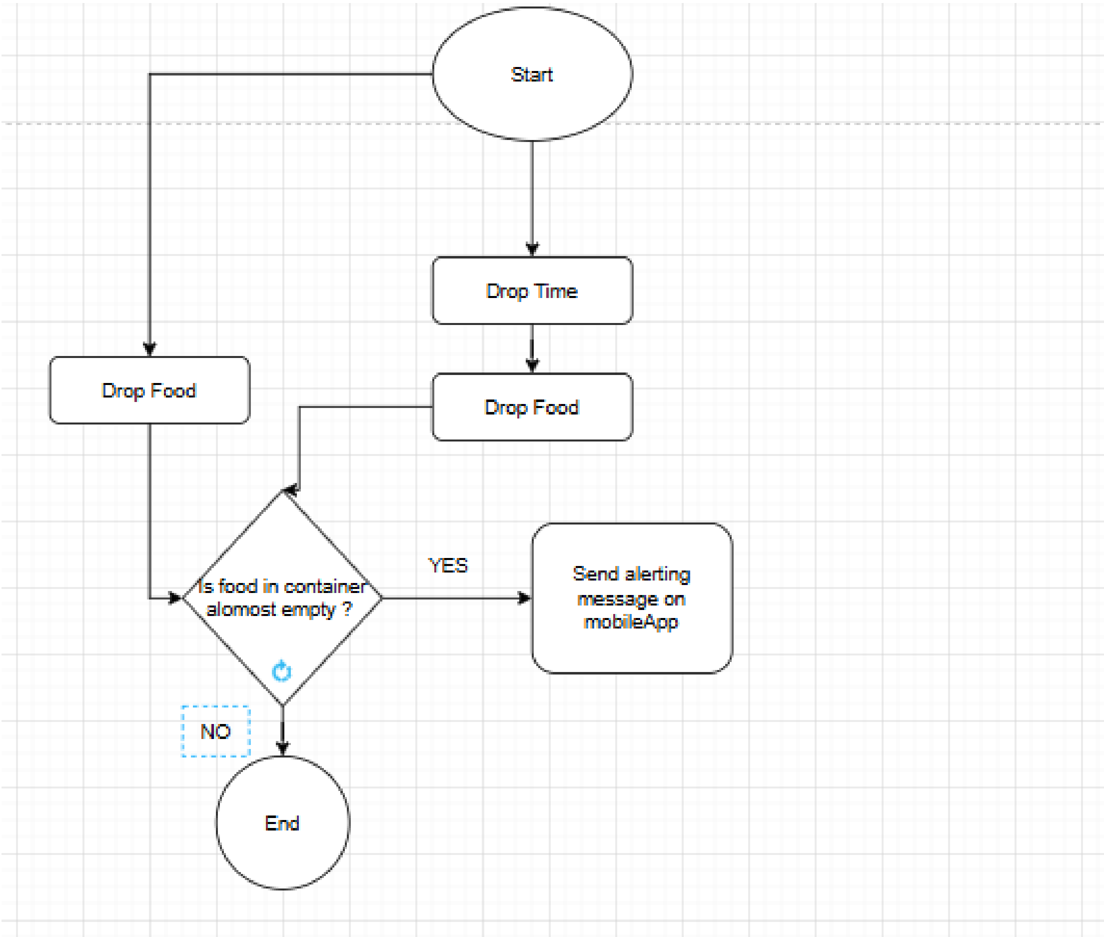

Figure 3.4 Chart Feeding

c. **Flowchart for water system :**

the figure 3.5 :Chart Water will show the logical process for feeder system by check the food container and the way to send alert .

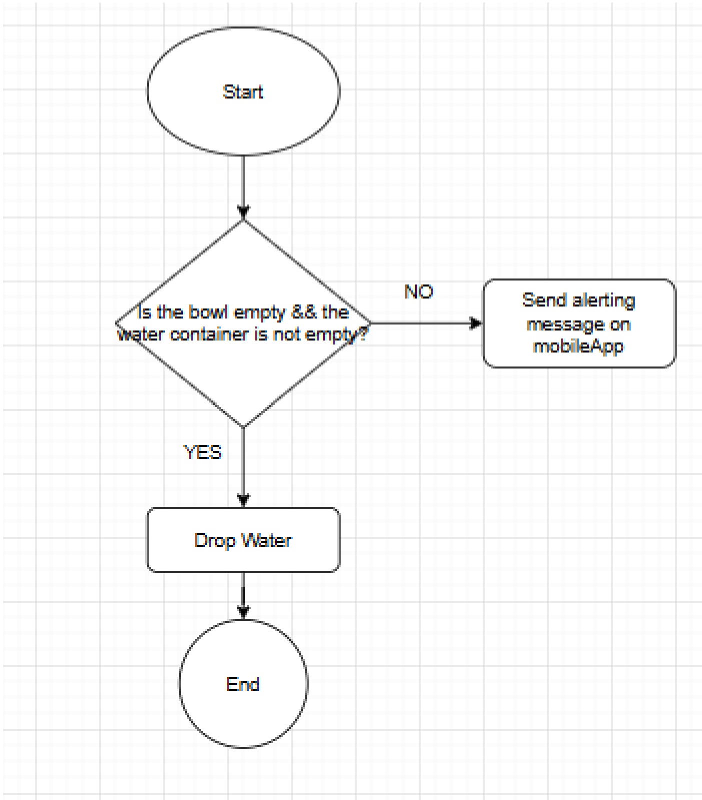

Figure 3.5 :Chart Water

- ## Sequence Diagrams

a. Sequence Diagram (Feed Now) :

in figure 3.6 : Sequence Diagram (Feed Now) Shows how the mobile app request passes through the backend to the ESP32

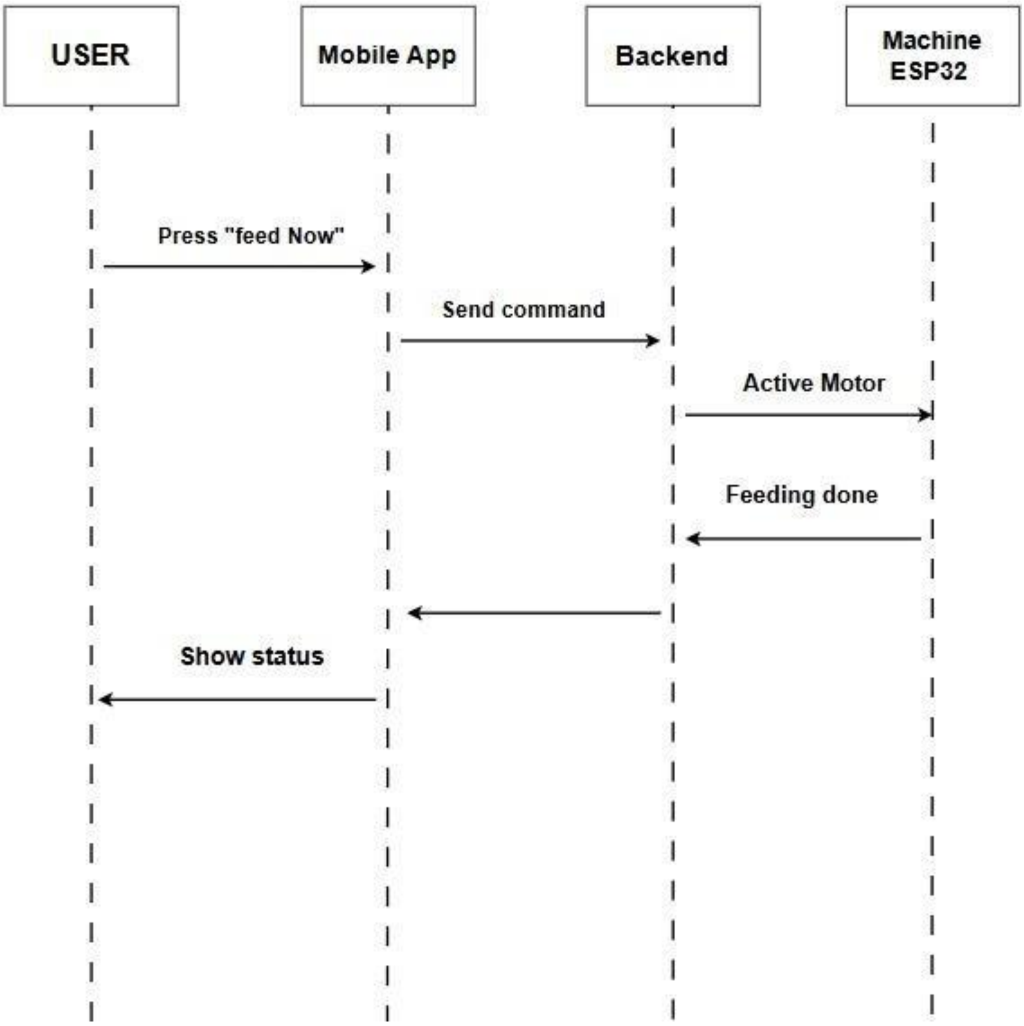

Figure 3.6 : Sequence Diagram (Feed Now)

b. Sequence Diagram – AI (Meal Schedule):

in figure 3.7 : Sequence Diagram (AI) shows how AI schedule meals depend on pet weight from data base

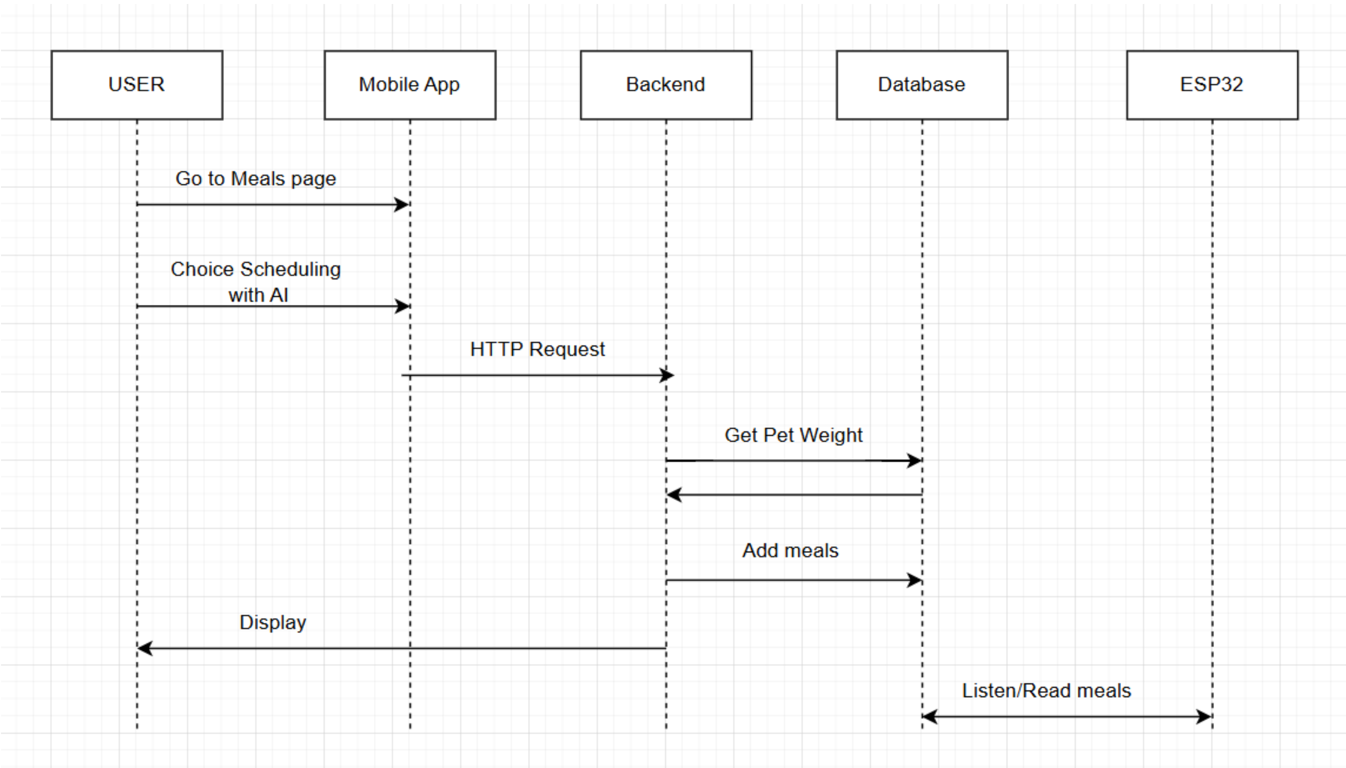

Figure 3.7 : Sequence Diagram (AI)

- **Use Case:**
  we have one actor called User and the pages in our app can the actor see it shows in the figure 3.8 : Use Case Digram

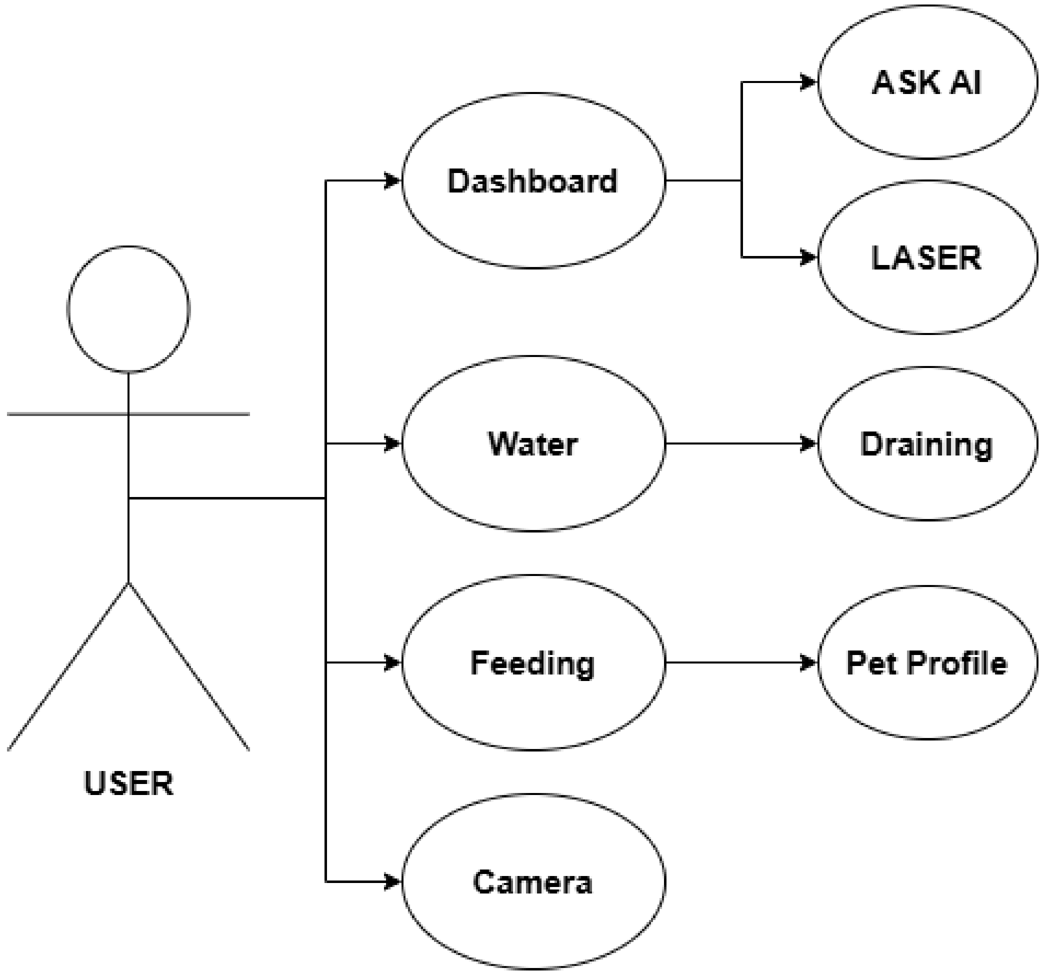

Figure 3.8 :Use Case Digram

3.5 Schematic diagrams

The schematic diagram shows how the ESP32 connects to the feeder, water pump, sensors, and camera, and how it communicates with the mobile app through the backend to control the system Figure 3.9: Schematic Diagram shows the details and how all component connect to esp32 .

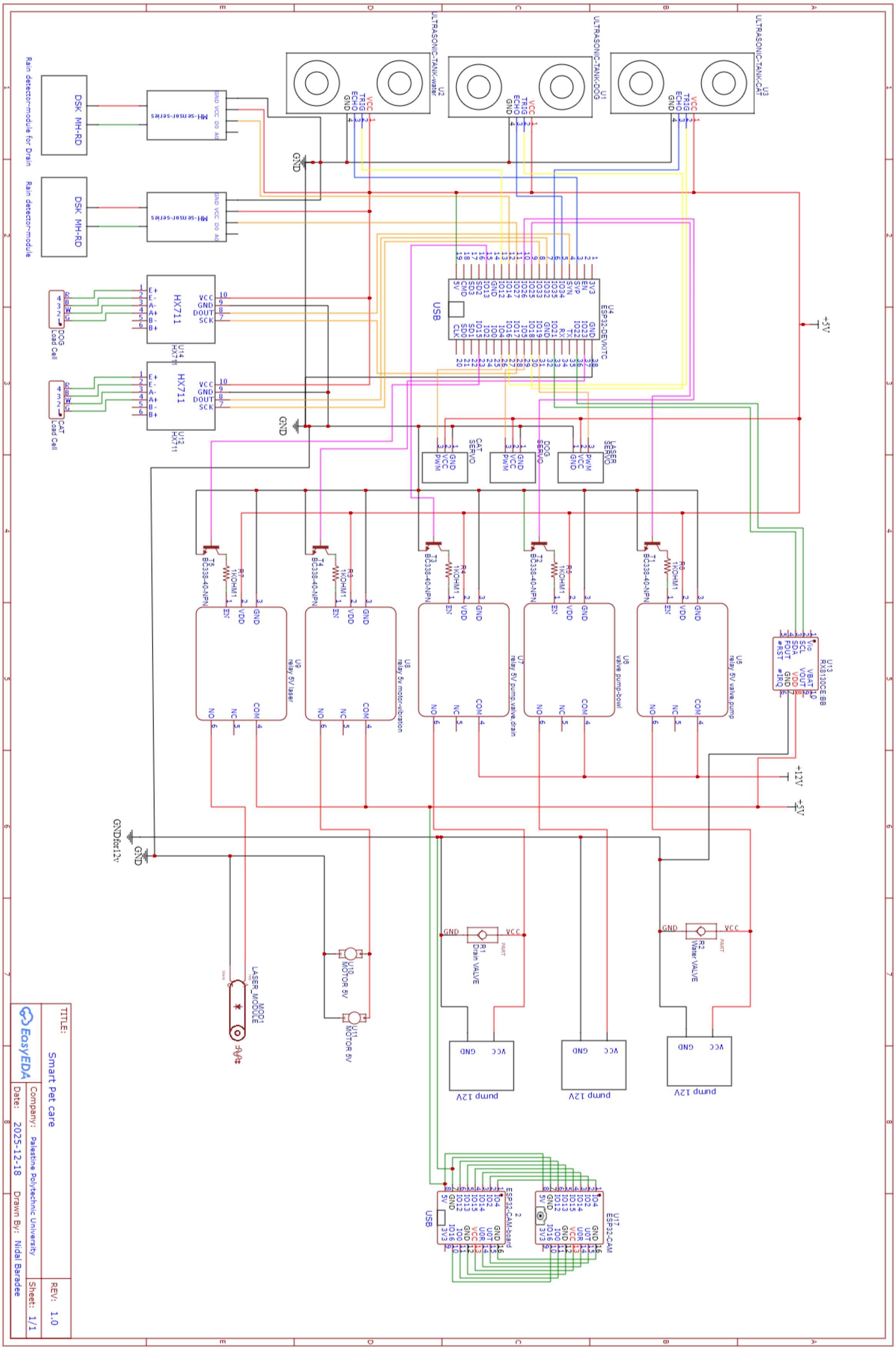

Figure 3.9: Schematic Diagram

- **DS3231 RTC Module Real Time Clock:**

The DS3231 is a low-cost, extremely accurate I2C Real Time Clock (RTC) with an integrated temperature compensated crystal oscillator (TCXO) and crystal. The device incorporates a battery input, and maintains accurate timekeeping when main power to the device is interrupted. The integration of the crystal resonator enhances the long-term accuracy of the device as well as reduces the piece-part count in a manufacturing line. The RTC maintains seconds, minutes, hours, day, date, month, and year information. The date at the end of the month is automatically adjusted for months with fewer than 31 days, including corrections for leap year. The clock operates in either the 24-hour or 12-hour format with an AM/PM indicator[10].

- **Ultrasonic sensor:**

An ultrasonic sensor is an instrument that measures the distance to an object using ultrasonic sound waves. An ultrasonic sensor uses a transducer to send and receive ultrasonic pulses that relay back information about an object’s proximity. Ultrasonic sensors work by sending out a sound wave at a frequency above the range of human hearing. The transducer of the sensor acts as a microphone to receive and send the ultrasonic sound. Our ultrasonic sensors, like many others, use a single transducer to send a pulse and to receive the echo. The sensor determines the distance to a target by measuring time lapses between the sending and receiving of the ultrasonic pules[13].

- **Servo Motor:**

The Tower pro MG996R Digital Servo is 180° rotation servo. It is a Digital Servo Motor which receives and processes PWM signal faster and better. It equips sophisticated internal circuitry that provides good torque, holding power, and faster updates in response to external forces[16

- **Load Cell Sensor 5 KG:**

A load cell is a transducer that converts force into a measurable electrical output. The straight bar load cell can convert pressure (force) into an electrical signal. Each load cell is able to measure the electrical resistance that changes in response to and proportional to the strain applied to the bar. Load cells generally consist of a spring element on which strain gauges have been placed. Then finally, the deformation of the strain gauge is interpreted by analysis electronics to determine the weight [15].

- **Solenoid Valve for Water :**

A solenoid valve is a control unit that uses electricity to automate the opening and closing of an orifice in the valve body, allowing, preventing, or redirecting the flow of water, gas, or other medium. It is made up of three main parts: the solenoid body, the electromagnetically inductive coil, and internal components (stem, plunger, and spring). It's works by controlling the flow of liquids in a positive, fully-closed or fully-open mode. They are often used to replace manual valves or for remote control. Solenoid valve function involves either opening or closing an orifice in a valve body, which either allows or prevents flow through the valve. A plunger opens or closes the orifice by raising or lowering within a sleeve tube by energizing the coil .

- **Rain Detection Sensor:**

The Detection sensor module is used for rain detection. It is also for measuring rainfall intensity. Rain sensor can be used for all kinds of weather monitoring and translated into output signals and AO. It includes a printed circuit board (control board) that "collects" the raindrops. As raindrops are collected on the circuit board, they create paths of parallel resistance that are measured via the op-amp. The lower the resistance (or the more water), the lower the voltage output. Conversely, the less water, the greater the output voltage on the analog pin. A completely dry board, for example, will cause the module to output 5V. So for that when water exists the output is 0 and if not the output is 1 [5],

- **12V Pump for Water:**

The water pump can be defined as a pump which uses the principles like mechanical as well as hydraulic throughout a piping system and to make sufficient force for its future use. At present these pumps are utilized within a wide range of housing, farming, municipal, and manufacturing applications. The working principle of a water pump mainly depends upon the positive displacement principle as well as kinetic energy to push the water. These pumps use AC power otherwise DC power for energizing the motor of the water pump. Pumps like these are used to transport huge amounts of water

- **ESP32 CAM:**

The Original ESP32-CAM Camera Module Kit is a compact and powerful IoT camera solution featuring an ESP32 dual-core processor with integrated 2.4 GHz Wi-Fi and Bluetooth. It comes with 8 MB PSRAM, 4 MB flash, and an OV2640 camera sensor supporting resolutions up to 1600×1200 (UXGA). Lens options include 66°, 120°, and 160° fields of view, with 850 nm IR night vision capability. The module supports MicroSD storage, multiple I/O interfaces (UART, SPI, I2C, PWM, ADC, DAC), and operates from a 5 V supply, making it ideal for wireless video streaming, surveillance, and embedded AI vision applications [6]

- **Laser Module**:

Small device that emits a focused light beam, often red (650 nm). Runs on 3–5 V, low power, used for pointing, alignment [20], or sensors

3.6 Summary

- **Main Controller:** ESP32 chosen for built-in Wi-Fi/Bluetooth, dual-core processing, power efficiency, and enough GPIO pins.
- **Subsystems:**
  - **Automated Feeder**
  - **Automated Water**
  - **Camera Module**
  - **Entertainment System**.
  - Water Bowl Draining System
  - Pet Profile Management (Cloud-based)
- **Software Tools:**
  - **Mobile App:** Flutter (cross-platform).
  - **Cloud Backend: Firebase Realtime Database**
  - **Notifications:** Firebase Cloud Messaging.

- **Camera Streaming:** ESP32-CAM with Render server (cost-effective and stable live video streaming).
- **AI-assisted guidance**

- **Design Documentation:**
  - Block diagrams and conceptual diagrams.
  - Flowcharts for system, feeder, and water modules.
  - Sequence diagrams for AI meal scheduling and Feed Now
  - A schematic diagram showing all ESP32 hardware connections and the communication link with the backend services.

3.7 Future Work

The following steps are planned: Hardware Implementation with ESP32 with sensors (load cell, ultrasonic, liquid level, rain detector) and actuators (servo motors, solenoid valve, water pump, laser module). Integrate the ESP32-CAM module for live video streaming.

Ensure power supply and cable management

Software Development : Develop (Flutter) application with user-friendly interfaces for feeding, hydration, AI, camera, entertainment

Improve Firebase database structure and expand mobile application features

Connect the system to the Rule-Based Artificial Intelligence System for intelligent pet care guidance and decision-making support. Integrate Firebase Cloud Messaging for real-time alerts and notifications.

CHAPTER 4: IMPLEMENTATION AND TESTING

4.1 Implementation Issues:

1. **Load Cell Calibration:** Initial readings from the load cell and HX711 amplifier were unstable due to vibration, mechanical mounting issues, and electrical noise. The problem was mitigated by firmly securing the load cell onto a rigid base, improving wire connections, and applying a software-based moving average filter to smooth out fluctuations. Proper taring and calibration were also repeated to ensure accurate food-weight measurement the figure 3.18:Load Cell shows how load cell stabilizing to base.

Figure 3.18 : Load Cell

2. **Firebase Synchronization Delays**: Occasional delays occurred when reading or writing data to Firebase, especially during poor Wi-Fi conditions. A reconnection strategy and retry mechanism were added in the firmware to ensure continuous cloud communication. Real-time listeners were also enabled in the app to allow instant updates without manual refresh.
3. **Ultrasonic Sensor inside food container :** The ultrasonic sensor was installed inside the food container to monitor the food level. It was fixed on the inner side of the container lid, facing downward toward the food. This position allows the sensor to measure the distance to the food surface and determine the remaining food level accurately.

The sensor was securely mounted to ensure stability during normal use and to prevent movement when opening or closing the container. Placing the sensor at the top also helps protect it from direct. contact with food and helps maintain stable and reliable readings during operation.

Figure 3.19 shows the ultrasonic sensor fixed inside the food container for food level monitoring.

Figure 3.19: ultrasonic in food container

4. **Integration Between Subsystems:** When all subsystems (feeding, water supply, draining, camera, and laser play) were tested together, certain operations interfered with others due to shared power or processing load. Separating power sources, optimizing code timing, and reorganizing task priorities solved most conflicts and ensured stable operation during full-system tests.
5. **Entertainment System (Laser Module with Servo Motor):** The entertainment system consists of a laser module mounted on top of a servo motor. The servo was installed at the upper part of the system, and the laser was fixed at an angle of approximately 45 degrees. This angle allows the laser beam to project onto the ground in front of the device.The servo motor controls the movement of the laser, enabling it to rotate and create motion on the ground for pet interaction. The laser module was securely attached to the servo using adhesive material to ensure stable operation during movement.Figure 3.20 shows the laser module mounted on the servo motor for the entertainment system.

Figure 3.20: Laser module

6. **Ultrasonic Sensor for Water Level Measurement:** An ultrasonic sensor was installed on the top of the water tank to measure the water level. The sensor was fixed facing downward toward the water surface, allowing it to measure the distance between the sensor and the water level.Based on the measured distance, the system calculates the remaining water percentage inside the tank. The sensor was securely mounted to ensure stable readings and to prevent movement during normal operation.Figure 3.21 shows the ultrasonic sensor installed on the water tank for water level monitoring.

Figure 3.21 : Ultrasonic Sensor for Water Level

7. **Water Bowl Level Control (Rain Sensor and Pump):** A rain sensor was installed inside the water bowl to detect the water level during filling. The sensor was positioned so that when the water reaches a specific level and touches the sensor, the filling process automatically stops.A small water pump was placed inside the bowl to create a fountain effect. The pump operates only after the bowl is filled to the required level, ensuring proper water circulation while preventing overflow.Figure 3.22 shows the rain sensor and water pump installed inside the water bowl.

Figure 3.22 : rain detector

8. C **Camera Installation and Field of View:** The camera module was installed on the front side of the system, with only the camera lens visible from the outside. The camera was positioned to cover the food and water bowls as well as the surrounding area.The second image shows the camera field of view, demonstrating the area monitored by the camera during operation. This setup allows continuous observation of the pet's activity around the feeding area while keeping the camera safely enclosed inside the system.Figure 3.22 shows the camera installation,

Figure 3.23 : camera

9. **Water Tank Drainage Safety System:** This figure shows the water tank drainage system. A rain sensor was installed inside the water tank to detect the water level and prevent the drainage system from operating while the tank is being filled.The drainage function is disabled when water is detected by the sensor, ensuring that water is not drained during the filling process. This setup helps avoid water loss and ensures safe and controlled tank operation.Figure X shows the water tank drainage safety system with the installed rain sensor.

Figure 3.24: draining system

4.2 Implementation Challenges:

1. **Relay Activation Issue Due to ESP32 3.3V Logic Level:**

   During testing, the ESP32 could not directly drive the 5V relay modules because its GPIO pins output only 3.3V logic signals. To solve this, a transistor-based interface using an NPN transistor as a low-side switch was added between the ESP32 and the relay input.

   When the ESP32 output is 0V, the transistor is OFF and the relay input stays at 5V, so the relay remains OFF. When the ESP32 output is 3.3V, the transistor turns ON and pulls the relay input to 0V, activating the relay because it operates in an Active-LOW mode.

   This solution converts the ESP32's 3.3V logic into a compatible 5V control signal while keeping the control behavior effectively Active-HIGH from the ESP32 side.

2. **ESP32-CAM Global Streaming and security :**

During the development of the monitoring system using the ESP32-CAM module, the camera stream initially worked only within the local network (LAN), which meant that remote access from outside the home network was not possible.

To enable global access, a dedicated Render server was deployed to act as a relay server, allowing the ESP32-CAM video stream to be accessed over the internet. This successfully enabled remote viewing from anywhere outside the local network.

However, once global access was enabled, a new security issue appeared — anyone who obtained the stream link could open it without any authentication.

To address this, a comprehensive security layer was implemented, including:

- OTP-based email verification
- Short-lived session authentication using JWT
- Session revocation and re-verification when needed

This ensured that only authorized users could access the live stream, significantly improving the system’s security.

Despite this improvement, some limitations related to video resolution and frame rate still exist due to the hardware constraints of the ESP32-CAM, such as limited processing power, bandwidth, and streaming stability.

4.3 Validation and Testing

Validation and Testing: System testing and validation were conducted to ensure that all hardware and software components of the Smart Pet Care system operate correctly and meet the specified functional requirements. The testing process was divided into three main stages: unit testing, integration testing, and full system validation.

1- **Unit Testing:** Each hardware and software component was tested individually to verify correct functionality before system integration. The load cell and ultrasonic sensors were tested separately to ensure accurate food weight and level measurements. The liquid-level sensor was tested to confirm proper water detection. Servo motors were tested independently to verify accurate angular movement for food dispensing and laser motion. The water pump and solenoid valve were tested using direct relay activation. The ESP32-CAM module was tested for local video streaming functionality. Firebase database read/write operations were tested independently, and each Flutter application interface module was tested using simulated data.

2- **Integration Testing:** After successful unit testing, system components were gradually integrated. The ESP32 was connected to all sensors and actuators, and real-time data communication with Firebase was verified. Control commands from the Flutter application (Feed Now, Drain Now, Laser ON/OFF) were tested to ensure correct execution by the ESP32. The camera stream was integrated with the external server and tested for global accessibility. Data synchronization between hardware, Firebase, and the mobile application was verified under real operating conditions.

3- **System Testing and Validation:** Full system testing was performed by operating the Smart Pet Care system as a complete integrated solution. Manual and scheduled feeding operations were tested and validated through real food dispensing and real-time weight updates. The automated water refill system was validated by lowering the water level until the pump activated automatically. The water bowl draining system was tested through manual activation from the mobile application. Live camera monitoring was validated through successful global video streaming. Notifications and alerts were validated by generating feeding and water-level events and confirming their arrival on the mobile application.

Table 4.1 summarizes the feeding system test cases, showing the Firebase paths, inputs, and expected versus actual outputs to verify that all feeding and sensing functions operate correctly.

Table 4.1: Feeding System Test Cases

Test ID	Test Description	Firebase Path	Input	Expected Output	Actual Output	Result
01	Manual cat feeding command	/feeding/commands/feed_cat	Change value from 0 → 1	Cat feeding servo activates and food is dispensed	As expected	Pass
02	Manual dog feeding	/feeding/commands/feed_dog	Change value from 0	Dog feeding	As expected	Pass

	command		→ 1	servo activates and food is dispensed		
03	Cat food level update	/feeding/sensors/cat_food_level	Food level changes	New food level value stored in Firebase	As expected	Pass
04	Dog food level update	/feeding/sensors/dog_food_level	Food level changes	New food level value stored in Firebase	As expected	Pass
05	Cat weight update via load cell	/feeding/sensors/cat_weight	Food dispensed	Updated cat food weight displayed in Firebase	As expected	Pass
06	Dog weight update via load cell	/feeding/sensors/dog_weight	Food dispensed	Updated dog food weight displayed in Firebase	As expected	
07	Cat distance detection	/feeding/sensors/cat_distance	Pet approaches feeder	Updated dog food weight displayed in Firebase	As expected	Pass
08	Dog distance detection	/feeding/sensors/dog_distance	Pet approaches feeder	Distance value updated in Firebase	As expected	Pass
09	Cat food low alert	/feeding/alerts/cat_food_low	Food level below threshold	Value changes to true	As expected	Pass
10	Dog food low alert	/feeding/alerts/dog_food_low	Food level below threshold	Value changes to true	As expected	Pass
11	Load scheduled meals	/feeding/meals/	App reads schedule	All meals displayed in app	As expected	Pass

Table 4.4 summarizes the water supply and draining system test cases, showing the Firebase paths, inputs, and expected versus actual outputs to verify that the draining functions and water-level alerts operate correctly

Table 4.4: Water Supply & Draining System Test Cases

Test ID	Test Description	Firebase Path	Input	Expected Output	Actual Output	Result
01	Manual drain activation	/water/controls/drain_button	Change value from false → true	Draining process starts and water is released from the bowl	As expected	Pass
02	Draining status update	/water/status/is_draining	Draining starts	Value changes to true	As expected	Pass
03	Stop draining status	/water/status/is_draining	Draining completes	Value changes to false	As expected	Pass
04	Low water alert	/water/alerts/water_low	Water level drops below threshold	Alert value changes to true	As expected	Pass
05	Dish empty detection	/water/sensors/dish_empty	Bowl becomes empty	Value changes to true	As expected	Pass
06	Tank full detection	/water/sensors/tank_full	Water tank reaches maximum	Value changes to true	As expected	Pass
07	Tank percentage update	/water/sensors/tank_percentage	Water level changes	Updated tank percentage stored in Firebase	As expected	Pass
08	Automatic refill	/water/sensors/dish_empty	Value = true	Water pump turns ON	As expected	Pass

	activation			automatically		
09	Automatic refill stop	/water/sensors/tank_full	Value = true	Water pump turns OFF	As expected	Pass
10	Refill status consistency	/water/status/is_draining	Refill in progress	Draining remains disabled during refill	As expected	Pass

Table 4.4 summarizes the camera monitoring system test cases, outlining the inputs and expected versus actual outputs to verify that both local and global video streaming functionalities operate as intended.

Table 4.4: Camera Monitoring System Test Cases

Test ID	Test Description	Input	Expected Output	Actual Output	Result
01	Local camera streaming	Access local IP	Live video stream displayed	As expected	Pass
02	Global camera streaming	Access via Render server	Live video stream available	Low FPS	Pass with limitation

Table 4.5 summarizes the camera access security (OTP) test cases, detailing the inputs and expected versus actual outputs to verify that OTP generation, validation, expiration handling, and reuse prevention all function correctly.

Table 4.5: Camera Access Security (OTP) Test Cases

Test ID	Test Description	Firebase Path	Input	Expected Output	Actual Output	Result
01	Generate camera OTP	/otps/cam001/code	User requests camera access	OTP code is generated and stored in Firebase	As expected	Pass

02	Store user email with OTP	/otps/cam001/email	User enters email address	Email is saved correctly with OTP record	As expected	Pass
03	OTP expiration time	/otps/cam001/expiresAt	OTP generated	Expiration timestamp stored correctly	As expected	Pass
04	OTP usage validation	/otps/cam001/used	Valid OTP entered	Value changes from false → true	As expected	Pass
05	Block reused OTP	/otps/cam001/used	Reuse old OTP	Access denied	As expected	Pass
06	Expired OTP rejection	/otps/cam001/expiresAt	Current time > expiry	Access denied	As expected	Pass

Table 4.6 summarizes the AI assistant test cases, outlining the inputs and expected versus actual outputs to confirm that the system provides accurate responses and personalized recommendations.

Table 4.6: AI Assistant Test Cases

Test ID	**Test Description**	**Input**	**Expected Output**	**Actual Output**	**Result**
01	AI response	Ask pet care question	Automated response based on Firebase readings	As expected	Pass
02	AI recommendation	Pet weight	Personalized	As expected	Pass

		entered	feeding advice given		

Table 4.7 summarizes the entertainment (laser play) system test cases, presenting the inputs and expected versus actual outputs to verify that the laser system activates and functions as intended.

Table 4.7: Entertainment (Laser Play) System Test Cases

Test ID	Test Description	Firebase Path	Input	Expected Output	Actual Output	Result
01	Manual laser system activation	/entertainment/commands/ system_on	Change value from false → true	Laser system turns ON and servo starts movement	As expected	pass
02	Manual laser system deactivation	/entertainment/commands/ system_on	Change value from true→ false	Laser system turns OFF and servo stops	As expected	pass
03	Continuous entertainment operation	/entertainment/commands/ system_on	Keep value = true for a period	Laser continues operating without interruption	As expected	pass

Table 4.8 summarizes the pet profile management test cases, showing the inputs and expected versus actual outputs to verify that pet data is correctly stored and retrieved from the system.

Table 4.8: Pet Profile Management Test Cases

Test ID	Test Description	Firebase Path	Input	Expected Output	Actual Output	Result
01	Add new pet profile	/profiles	Enter name, age, breed, type,	Data stored correctly in Firebase	As expected	Pass
02	Load pet	/profiles	App request	Pet data	As expected	Pass

	profile in app			displayed correctly		

4.4 Testing Results and Discussion

- Validation followed unit testing → integration testing → full system testing, to confirm each module works alone and then works end-to-end (Flutter app → Firebase → ESP32 → sensors/actuators).
- Test cases confirmed correct operation of the main subsystems: feeding, water supply + draining, camera monitoring, OTP camera security, AI assistant, entertainment (laser), and pet profile management.
- Camera results: global streaming was achieved via a Render server, but low video quality and FPS remained due to inherent ESP32-CAM hardware constraints.
- Security: OTP access control was tested successfully (generation, expiration, reuse blocking, and rejection of expired OTPs).
- You can additionally state that protected API requests are secured with JWT authentication
- Overall, the system's major functionalities were validated under real conditions, with the camera FPS noted as the main limitation.

One of the significant results achieved in this project is the development of the Smart Pet Care mobile application, which provides a user-friendly interface that allows pet owners to remotely monitor and control the system. Through the application, users can check food and water levels, activate manual or scheduled feeding, control the water bowl draining process, operate the entertainment (laser play) system, and access the live camera stream using secure authentication. The application also enables pet profile management, where users can enter pet information such as age, weight, and type. This data is utilized by the integrated AI system to generate personalized feeding schedules and care recommendations.

The following figures, Figure 4.1: Dashboard Screen, Figure 4.2: Ask AI Screen, Figure 4.3 : Feeding Screen , Figure 4.4 : Water System Screen , Figure 4.5 : Live Camera Screen .

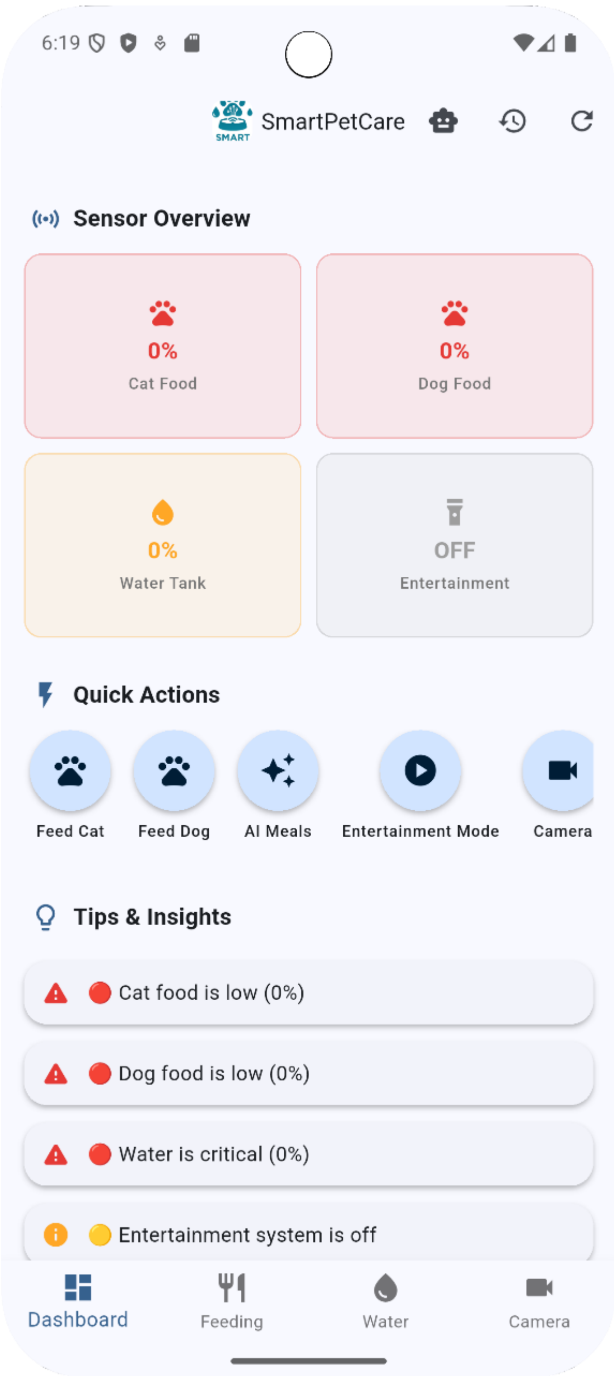

Figure 4.1 : Dashboard Screen

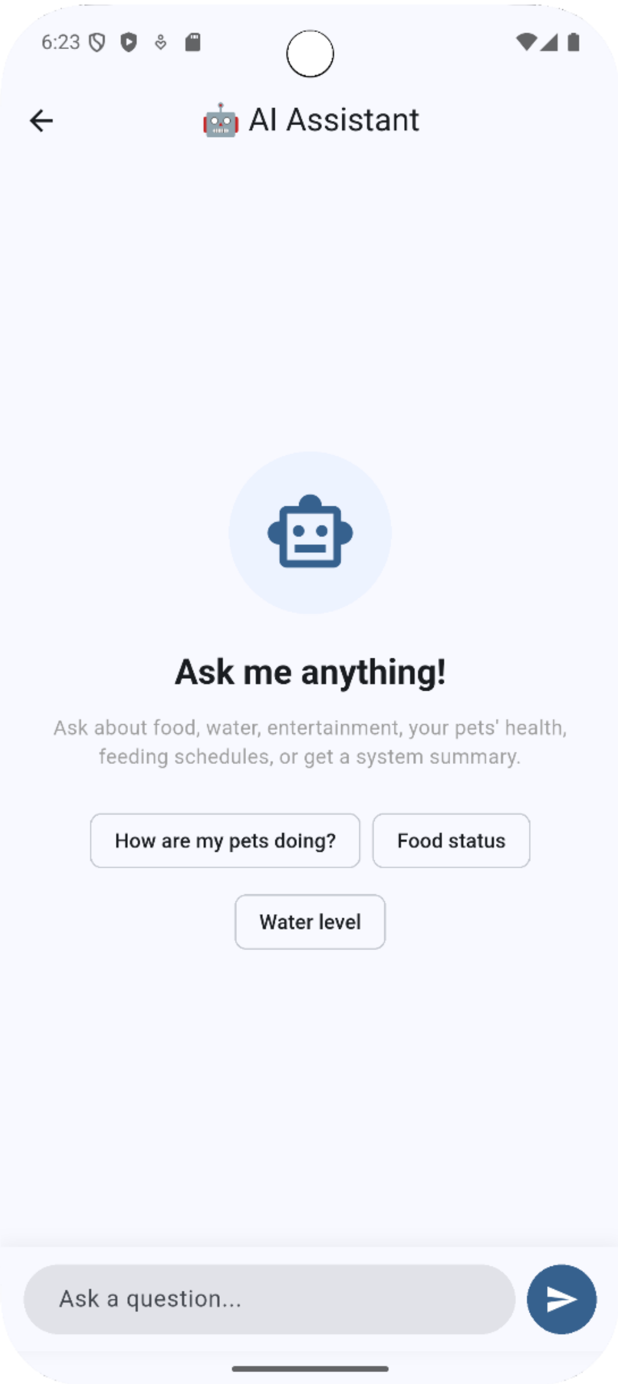

Figure 4.2 : Ask AI screen

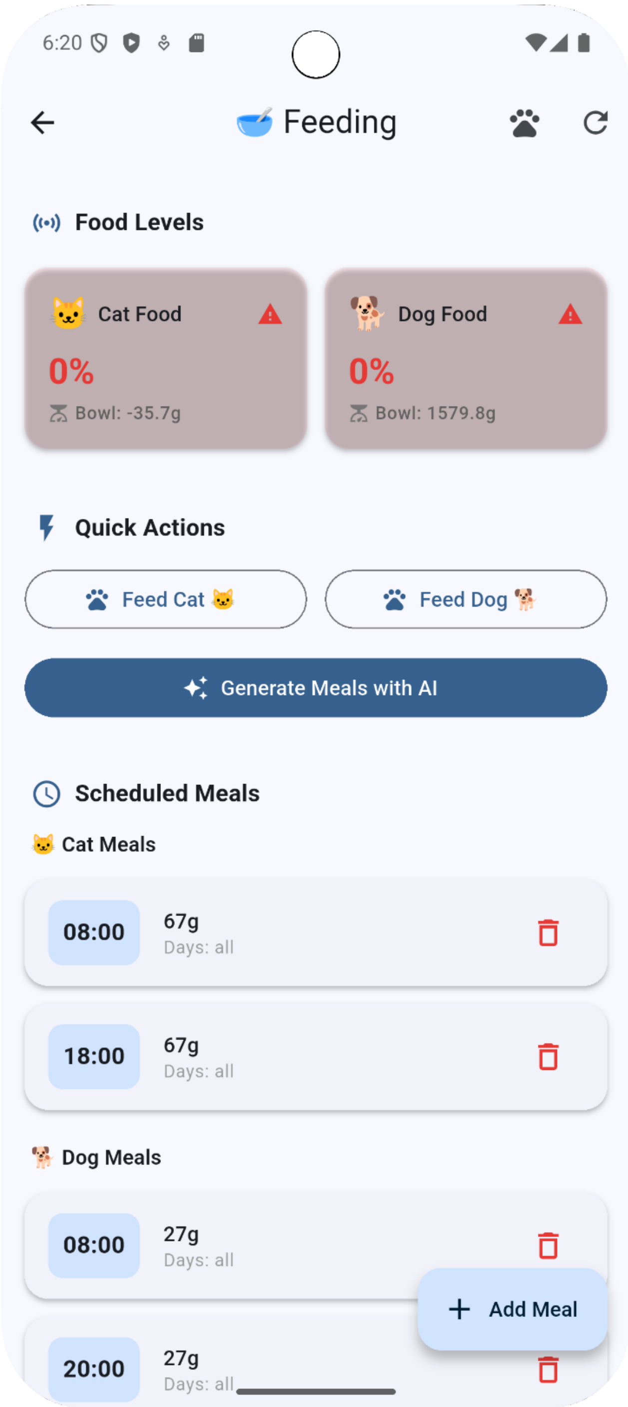

Figure 4.3 : Feeding Screen

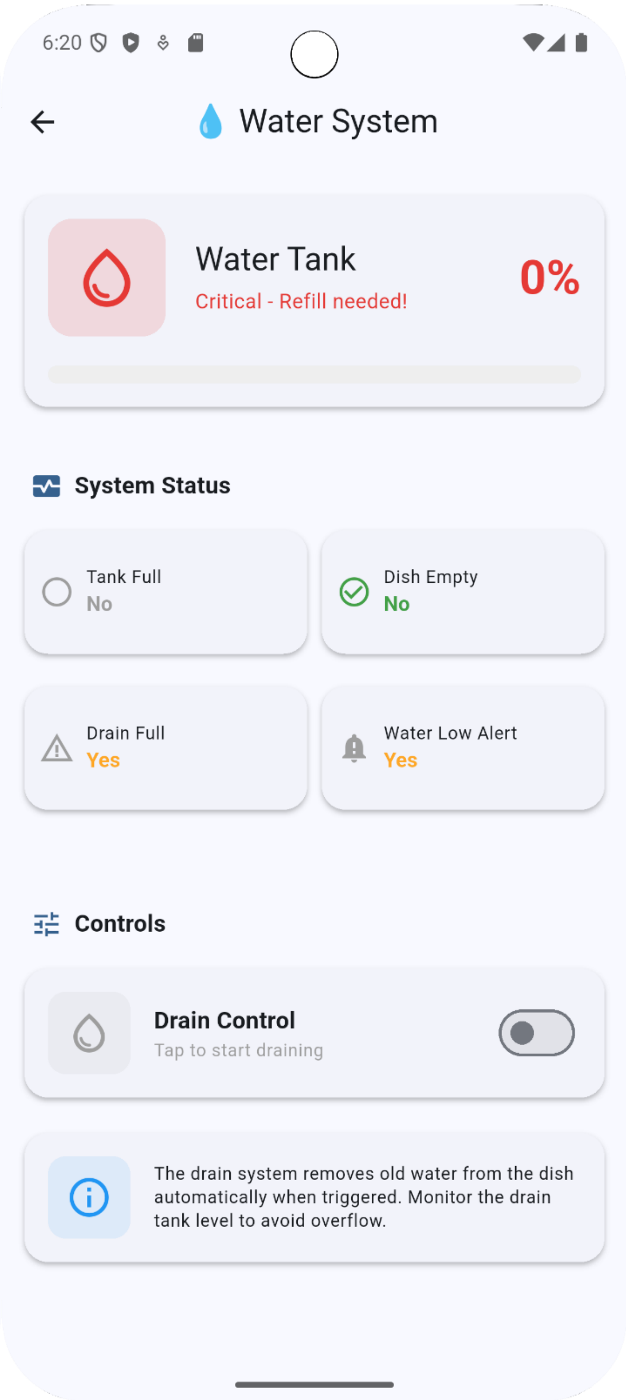

Figure 4.4 : Water System Screen

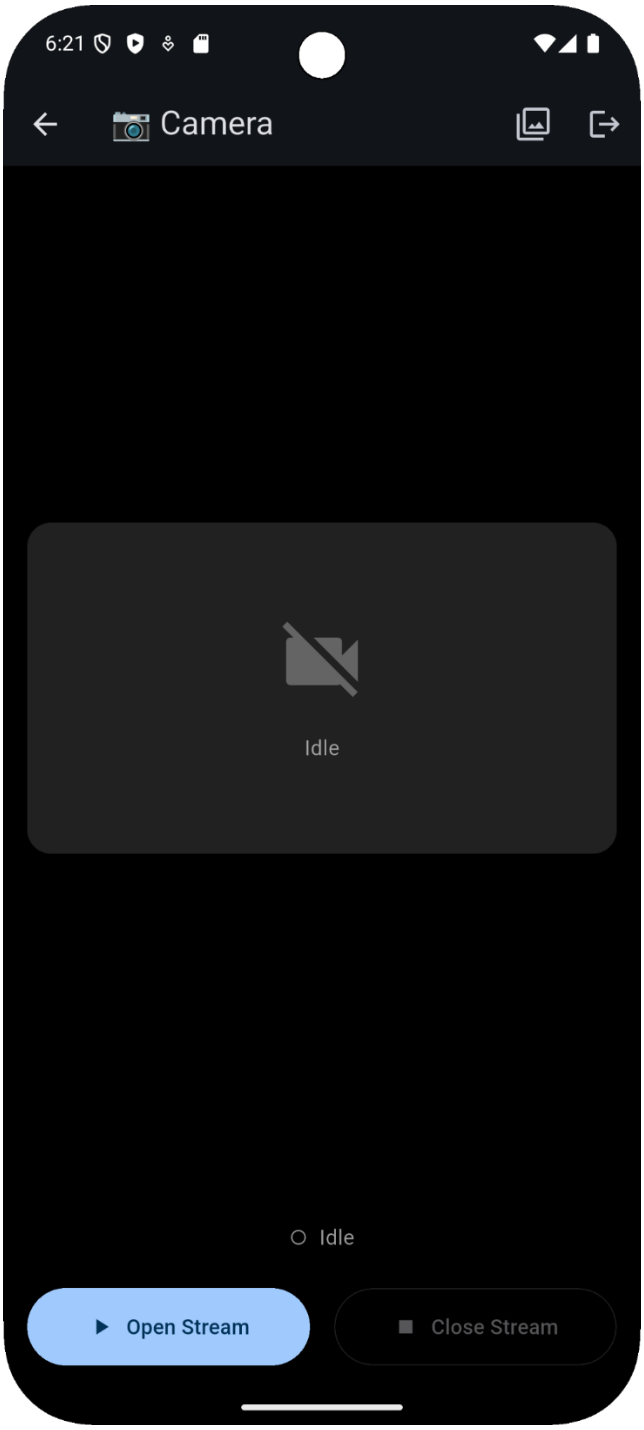

Figure 4.5 : Live Camera Screen

4.4 summary

This chapter presented the complete implementation and testing process of the Smart Pet Care system. It detailed the practical issues encountered during development, such as load cell instability, ultrasonic measurement fluctuations, Firebase synchronization delays, and subsystem integration challenges, along with the solutions applied to resolve them. Major implementation challenges were also discussed, including the relay interfacing limitation caused by the ESP32's 3.3V logic level and the global streaming limitations of the ESP32-CAM module.

Comprehensive validation and testing were conducted through unit testing, integration testing, and full system testing to ensure reliable performance of all subsystems. Test cases were designed and executed for the feeding system, water supply and draining system, camera monitoring, OTP-based camera security, AI assistant, entertainment (laser play) system, and pet profile management. All major functionalities were successfully validated under real operating conditions, with only minor limitations observed in camera frame rate due to hardware constraints.

The results confirm that the Smart Pet Care system operates as intended, providing reliable automation, real-time monitoring, secure camera access, and intelligent interaction through the mobile application. This ensures that the system meets its functional requirements and is ready for practical deployment.

CHAPTER 5: CONCLUSION AND FUTURE WORK

5.1 Concluding Remarks

This project successfully designed and implemented an automated and intelligent pet care system that supports scheduled feeding, continuous hydration, and a dedicated water drainage mechanism to maintain bowl cleanliness. The system also provides real-time monitoring and remote control through a mobile application enhanced with AI features.

Testing was performed using a structured process (unit, integration, and full system validation) to confirm correct operation of all subsystems.

The outcomes confirm reliable automation and real-time monitoring, with secure camera access using OTP and intelligent user interaction through the AI assistant.

The main observed constraint is related to the **ESP32-CAM streaming performance**, which is an inherent limitation of the chosen platform. However, the system compensates by providing a robust security mechanism: **OTP** is used as a time-bound verification step to protect camera access, while **JWT** is used to maintain authenticated sessions and authorize API requests, reducing the risk of unauthorized access to streaming endpoints and user data.

5.2 Future Work

1. **Upgrade the camera streaming subsystem**
   Replace ESP32-CAM with a higher-performance camera platform, or optimize streaming configuration and server pipeline to improve FPS and stability, since the limitation is tied to ESP32-CAM hardware constraints under constant load.
2. **Improve network resilience (offline tolerance)**
   Since Firebase synchronization delays were observed under poor Wi-Fi conditions, implementing stronger offline caching, queued commands, and robust reconnection strategies in firmware and app can improve reliability during unstable connectivity
3. **Strengthen safety and power design**
   Increase electrical isolation and improve power distribution (separate supplies, better grounding, protective components) to reduce interference between subsystems during full-load operation.
4. **Expand AI capabilities using historical data**
   The current AI assistant already provides responses and personalized recommendations. Future work can incorporate time-series trends (food/water consumption patterns) to provide predictive advice, smarter scheduling, and early warnings
5. **Upgrade the Draining system:**
   Add turbidity sensor When the sensor indicates high water turbidity beyond a predefined threshold, the system automatically activates the draining mechanism.

5.3 References

[1] An-Najah National University Repository, Research Item c1f3a663-9f2d-4de8-ba46-e9f31a446fb5. [Online]. Available: https://repository.najah.edu/items/c1f3a663-9f2d-4de8-ba46-e9f31a446fb5 [Accessed: Mar. 16, 2025].

[2] An-Najah National University Repository, Research Item dc30a190-f6c2-4ab3-a969-8c6fec7fc4f9. [Online]. Available: https://repository.najah.edu/items/dc30a190-f6c2-4ab3-a969-8c6fec7fc4f9 [Accessed: Mar. 14, 2025].

[3] Google Firebase Documentation, Cloud Functions for Firebase. [Online]. Available: https://firebase.google.com/docs/functions. [Accessed: Dec. 14, 2025].

[4] ESP32io, ESP32 Water Liquid Valve Tutorial. [Online]. Available: https://esp32io.com/tutorials/esp32-water-liquid-valve [Accessed: Aug. 11, 2025].

[5] ESP32io, ESP32 Rain Sensor Tutorial. [Online]. Available: https://esp32io.com/tutorials/esp32-rain-sensor [Accessed: Oct. 20, 2025].

[6] Random Nerd Tutorials, ESP32-CAM Video Streaming and Face Recognition with Arduino IDE. [Online]. Available: https://randomnerdtutorials.com/esp32-cam-video-streaming-face recognition-arduino-ide/ [Accessed: Aug. 26, 2025].

[7] MongoDB, What Is IoT Architecture
[Online]. Available:
https://www.mongodb.com/resources/basics/cloud-explained/iot-architecture. [Accessed: Nov. 15, 2025].

[8] Espressif Systems, ESP32 Series Datasheet (v5.2). [Online]. Available: https://www.espressif.com/sites/default/files/documentation/esp32_datasheet_en.pdf. [Accessed: Dec. 15, 2025].

[9] Arduino, Arduino UNO Rev3 Datasheet (A000066). [Online]. Available: https://docs.arduino.cc/resources/datasheets/A000066-datasheet.pdf. [Accessed: Aug. 15, 2025].

[10] Analog Devices (Maxim Integrated), DS3231: Extremely Accurate I²C-Integrated RTC/TCXO/Crystal. [Online]. Available: https://www.analog.com/media/en/technical-documentation/data-sheets/ds3231.pdf. [Accessed: Aug. 25, 2025].

[11] Analog Devices (Maxim Integrated), DS1307: 64 × 8, Serial, I²C Real-Time Clock. [Online]. Available: https://www.analog.com/media/en/technical-documentation/data-sheets/ds1307.pdf. [Accessed: Aug. 15, 202].

[12] SparkFun Electronics, Ultrasonic Ranging Module HC-SR04 (Datasheet). [Online]. Available: https://cdn.sparkfun.com/datasheets/Sensors/Proximity/HCSR04.pdf. [Accessed: Aug. 15, 2025].

[13] Manorshi (Ultrasonic Sensor Module), JSN-SR04T-3.0 Ultrasonic Sensor Module (Datasheet). [Online]. Available: https://jlrorwxhjiilll5q.ldycdn.com/JSN-SR04T-3.0-aidnqBpoKliRljSlqnqkilqj.pdf. [Accessed: Nov. 11, 2025].

[14] AVIA Semiconductor (via SparkFun), HX711: 24-Bit ADC for Weigh Scales (Datasheet). [Online]. Available: https://cdn.sparkfun.com/datasheets/Sensors/ForceFlex/hx711_english.pdf. [Accessed: Aug. 3, 2025].

[15] DwyerOmega, Load Cell Specifications (Measuring Range/Capacity Guidance). [Online]. Available: https://www.dwyeromega.com/en-us/resources/load-cell-specifications. [Accessed: Aug. 15, 2025].

[16] Tower Pro, MG996R High Torque Metal Gear Servo (Specifications). [Online]. Available: https://towerpro.com.tw/product/mg996r/. [Accessed: Aug. 15, 2025].

[17] Handson Technology, SG90 TowerPro Micro Servo (Datasheet). [Online]. Available: https://handsontec.com/dataspecs/motor_fan/SG90-Servo.pdf. [Accessed: Aug. 7, 2025].

[18] Need-For-Power Motor Co., Ltd., DATASHEET: NFP-FF-030SB-1863 Vibration Motor. [Online]. Available: https://nfpshop.com/wp-content/uploads/2023/05/DATASHEET-NFP-FF-030SB-1863-20230529_Watermark.pdf. [Accessed: Aug. 15, 2025].

[19] Precision Mini Drives, 130 ERM Vibration Motor Data Sheet (NFP-130SHZ-B). [Online]. Available: https://precisionminidrives.com/product/20mm-vibration-motor-25mm-type-model-nfp-130sh-10105. [Accessed: Oct. 19, 2025].

[20] Circuits-DIY, Laser Diode Driver Using LM317 Voltage Regulator IC. [Online]. Available: https://www.circuits-diy.com/laser-diode-driver-using-lm317-voltage-regulator-ic/. [Accessed: Aug. 15, 2025].

Appendix A:

The following Table A.1: Keywords lists the main keywords used in this report. These terms represent the core concepts, technologies, and system components of the Smart Pet Care project, helping readers understand the key topics discussed throughout the study.

Table A.1: Keywords

Keyword	Description
Smart Pet Care	Title and main system: an intelligent automated pet-care solution integrating feeding, hydration, monitoring, and AI
Internet of Things (IoT)	Architecture connecting sensors, actuators, ESP32, and cloud services
ESP32	Main microcontroller controlling all hardware modules with Wi-Fi/Bluetooth connectivity
ESP32-CAM	Camera module enabling live video streaming for pet monitoring
Firebase Realtime Database	Cloud backend for data storage, synchronization, authentication, and device state management
Rule-Based Artificial Intelligence System	AI assistant providing pet-care guidance and automatic meal scheduling
Load Cell Sensor (HX711)	Measures food weight to monitor feeding portions
Ultrasonic Sensor (HC-SR04)	Detects food and water levels inside containers
Solenoid Valve	Controls water flow electronically in refill and draining systems
Water Pump (12V DC)	Provides water supply and drainage operations
OTP Authentication	One-Time Password used to secure camera access
JWT Sessions	Short-lived authentication tokens securing system APIs and camera stream
RER / MER Formulas	Veterinary equations used by AI to calculate pet daily energy needs
RTC (Real-Time Clock)	Used to schedule feeding times and maintain accurate timekeeping, even during power loss.

Appendix B : ESP32 Code

Appendix c:ESP32-CAM Code

نموذج إقرار وتوثيق استخدام أدوات الذكاء الاصطناعي في مشروع التخرج

اسم الطالب / أسماء الطلبة: نضال برادعي / محمد بحر	عنوان المشروع: Smart Pet Care
الرقم / الأرقام الجامعية: 211001 / 211053	اسم المشرف الأكاديمي: د. رضوان طهبوب
الكلية / الدائرة: كلية تكنلوجيا المعلومات و هندسة الحاسوب /هندسة الحاسوب	السنة/الفصل الدراسي: 2025 الفصل الثاني

الغرض من الاستخدام	اسم الأداة	أماكن الاستخدام في التقرير (فصول / صفحات)	الأمر الأساسي المستخدم (Prompt)	هل تم تعديل الناتج؟	نسبة الاستخدام التقديرية	ملاحظات
توليد نصوص	☐ ChatGPT	**1.2 Problem Statement 3.5 camera**	This is our problem statement make it better	نعم	8 %	تم توليد النصوص ولكن تم تعديلها جميعا في قسم الكاميرا تم اخذ النص بالكامل من الذكائ الاصطناعي ولكن بالتاكيد تم طلب المرجع والتاكد من كل الخصائص ولم نقوم بتعديل النص لانه عبارة عن خصائص و مميزات الكامير
	☐ Gemini	Non-Functional Requirements	This is our project idea what is the Non-Functional Requirements	نعم	10%	
	☐ Claude	**1.6 Project Limitations/Constraints**	This is our project idea what is the limitations	نعم		
	☐ Microsoft Copilot					
	☐ other: ____					
تدقيق لغوي	☐ Grammarly	كامل التقرير	—	نعم	5%	
	☐ Quillbot					
	☐ Word Editor AI					

					☐ other: ____	
	5%	نعم	Give me summary for this project	**2.4 Summary** **3.6 Summary**	☐ ChatGPT	تلخيص محتوى
					☐ SMMRY	
					☐ Notion AI	
					☐ Scholarcy	
					☐ other: ____	
					☐ Excel Copilot	تحليل بيانات
					☐ Python AI Assistants	
					☐ Tableau AI	
					☐ other: ____	
					☐ DALL·E	رسم مخططات / أشكال
					☐ Canva AI	
					☐ Visio AI	
					☐ Lucidchart AI	

	☐ other: ChatGPT	Figure 2.1	ChatGPT/API road and how it works I give him the details	نعم		This is the only figure we generated by Chat
كتابة أكواد برمجية	☐ GitHub Copilot					
	☐ Replit AI					
	☐ Codeium					
	☐ other: ____					
توثيق مراجع	☐ EndNote					
	☐ Mendeley					
	☐ Zotero					
	☐ ChatGPT					
	☐ other: ____					
أخرى (حدد: ____)						

ملاحظات عامة:

- يجب ألا تتجاوز النسبة الإجمالية للمحتوى غير المعدل 20% لكل مكون من مكونات المشروع ما لم تحدد كلية الطالب أو المشرف نسبة مختلفة.
- يُشترط أن يقوم الطالب بالتأكد من صحة أي جزء تم توليده أو دعمه بالذكاء الاصطناعي و أن يكون الطالب قادرا على شرحه والاجابة على أي أسئلة حوله.

إقرار فريق مشروع التخرج:

نقر بأن استخدام أدوات الذكاء الاصطناعي تم بشكل مسؤول وبما يتوافق مع السياسات الجامعية، وقد راجعنا وحررنا كافة المخرجات بما يعكس فهمنا الشخصي.

توقيع الطالب / الطلبة: .... التاريخ:.................................

Certification and Anti-Plagiarism Declaration

This is to declare that the graduation project produced under the supervision of (name of supervisor) having the title “ ” was prepared by student/s(name of students) below in partial fulfilment of the requirements for the degree of Bachelor in Engineering of (name of engineering major) and no part hereof has been reproduced illegally (in particular: cut and paste) which can be considered as Plagiarism.

All referenced parts have been used to support and argue the idea and have been cited properly. I/We certify that I/we will not commit any plagiarism, cheating, or any other academic integrity

violation. I/We will be responsible and liable for any consequence if violation of this declaration is proven.

Any use of Artificial Intelligence (AI) tools in the preparation of this project has been clearly declared and documented in the related appendix at the end of the report, in accordance with the PPU guidelines.

Date: 18/8/2025

Graduation project group's student(s): Nidal Baradee , Mohammad Bahar

Name: Signature:

Name: Signature: