Design and Implementation of a Smart Pet Feeder Using Blynk and Load Sensor Technology

Nguyen Huu Dat[1], Le Tien Dat[1], Cung Nguyen Bich Tram[1], Tran Manh Tuan[1] and Le Vu Truong[1]

[1]Department of Software Engineering, FPT University, Thu Duc City, Ho Chi Minh City, Vietnam
E-mail: datnhqe180140@fpt.edu.vn .

Keywords: Pet Feeder, IoT, Embedded Systems

Abstract
The Pet Feeder Project aims to provide an automated and user-friendly solution for pet owners to manage their pet's feeding schedule remotely. Leveraging the Internet of Things (IoT) and embedded systems, the project ensures accurate and consistent feeding times and portions, even when the owner is not present. The system consists of an Arduino-based control unit, sensors for monitoring food levels, and a motorized mechanism for precise food dispensing. Additionally, the feeder is integrated with the Blynk mobile application, allowing users to schedule feedings, adjust portions, and receive notifications on feeding status. The prototype was tested under various scenarios to validate its reliability and functionality. The results demonstrate that the system effectively automates the pet feeding process, reducing manual intervention while ensuring the pet's health and well-being. This project contributes to the growing adoption of smart technology in pet care and offers a practical solution to common challenges faced by pet owners.

1. Introduction

In today's fast-paced world, balancing the responsibilities of pet care with busy personal and professional schedules can be a challenge for many pet owners. One of the most fundamental aspects of pet care is ensuring that pets are fed on time and in the correct portions. However, maintaining a consistent feeding routine can become difficult for those with unpredictable schedules, leading to issues such as overfeeding, underfeeding, or missed feeding times. These irregularities can contribute to various health problems in pets, including obesity, malnutrition, and digestive issues.

To address these challenges, there is a growing demand for automated solutions that can assist pet owners in managing feeding schedules more effectively. The integration of smart technology, particularly the Internet of Things (IoT), presents a promising opportunity to automate pet care tasks. By using IoT and embedded systems, it is possible to create intelligent feeding systems that ensure pets receive their meals consistently, even when their owners are not at home.

The Pet Feeder Project is designed to meet this need by providing an automated feeding system that not only dispenses food at scheduled times but also allows for remote control and monitoring through a mobile application. The system is equipped with sensors to monitor food levels, a motorized dispensing mechanism to ensure precise portioning, and wireless connectivity to enable real-time communication between the feeder and the owner's smartphone. This combination of hardware and software offers a practical solution to common pet care challenges, allowing owners to maintain a healthy feeding routine for their pets with minimal manual intervention.

This project also highlights the broader trend of integrating smart home technology into everyday life. As more households adopt connected devices, smart pet care solutions like the Pet Feeder have the potential to improve the quality of life for both pets and their owners. By automating routine tasks such as feeding, pet owners can focus on other aspects of pet care, confident that their pets' dietary needs are being met consistently and accurately.

2. Methods and Materials

2.1. System Model and Block Diagram

The developed Pet Feeder system integrates both hardware components and software to automate the feeding process and provide real-time monitoring. On the hardware side, key devices ensure seamless functionality. A load sensor is used to accurately measure the weight of the food in the dispenser, ensuring precise portion control. A manual button allows users to manually activate the food dispensing process. The system's core controller is the Arduino Uno, which processes inputs from the sensors and sends commands to various output components such as the servo motor, which opens and closes the feeder door to dispense food. An LED indicator provides status updates, signaling events like low food levels or active feeding processes.

Additionally, the system integrates the ESP8266 NodeMCU module for wireless communication, utilizing its built-in Wi-Fi capabilities to connect the Pet Feeder to the Blynk platform. This allows users to remotely control the feeder via a smartphone, manage feeding schedules, and receive real-time notifications. The LCD display presents important information, such as current food levels, feeding times, and system status.

On the software side, the programming code is responsible for coordinating the sensors, actuators, and communication modules, ensuring smooth operation. The block diagram in Figure 1 illustrates the system's components and their interactions. As shown in the diagram, the Arduino Uno takes inputs from the load sensor and manual button and controls the servo motor, LED, and LCD display as outputs. The ESP8266 NodeMCU manages communication with the Blynk platform, enabling remote monitoring and control of the system from a mobile device.

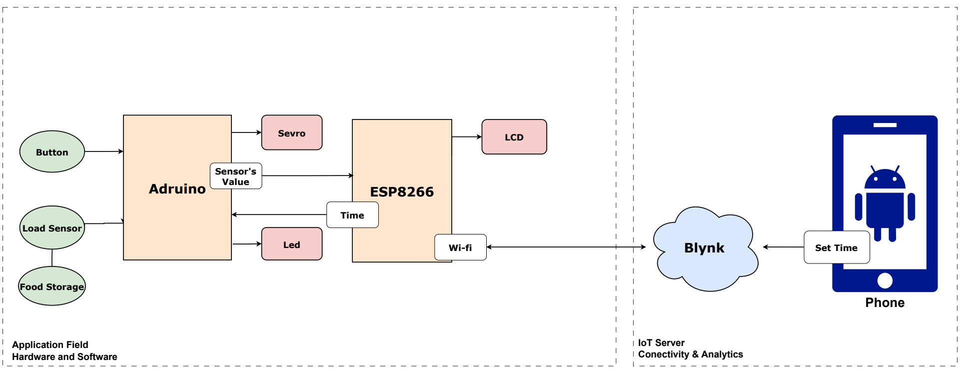

Figure 1. *Block diagram of the Pet Feeder system.*

2.2. Components and Peripheral Devices

1. **Arduino Uno**
   The Arduino Uno serves as the central processing unit of the Pet Feeder system. This microcontroller board is responsible for coordinating all the inputs and outputs, processing data from the sensors, and controlling the actuators. With its flexibility and ease of programming, the Arduino Uno is an ideal choice for managing the various components of the feeder, ensuring that each part operates in harmony to deliver a seamless user experience.
2. **Servo Motor**
   The Servo Motor is an essential component responsible for the precise control of the food dispensing mechanism. Connected to the Arduino Uno, the servo motor operates the dispensing door, opening and closing it as required to release the correct portion of food. Its ability to rotate to

Components/devices	ID/remarks
Arduino Uno R3	ATmega328P based
Liquid–crystal display (LCD)	16 × 2 LCD
Servo Motor	SG90
NodeMCU	ESP8266
Load Cell Sensor	YZC131
Led	5mm
Push Button	

Table 1. *Components and Peripheral Devices.*

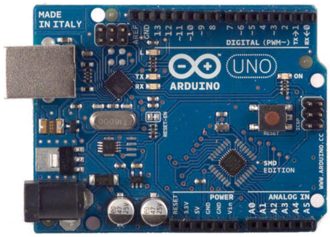

Figure 2. *Arduino Uno.*

a specified angle with high accuracy makes it perfect for controlling the amount of food dispensed, ensuring consistency with each feeding session.

Figure 3. *Servo Motor.*

3. **Load Cell Sensor**
   The Load Cell Sensor is responsible for measuring the weight of the food in the feeder, providing real-time data to the Arduino Uno. This sensor plays a key role in tracking the amount of food available and ensuring the correct portion is dispensed. By converting the force from the food's weight into an electrical signal, the load cell enables the system to accurately assess whether more food should be dispensed or if the feeder requires refilling. An **HX711** amplifier module is used in conjunction with the load cell to convert the small analog signals from the sensor into a form that can be read accurately by the Arduino.
4. **ESP8266 NodeMCU V1.0**
   The ESP8266 NodeMCU V1.0 is a Wi-Fi-enabled microcontroller that facilitates wireless communication between the Pet Feeder and the user's smartphone or other remote devices via the Blynk platform. This module enables users to monitor the feeding process, adjust settings, and receive notifications in real-time, regardless of their location. The integration of the ESP8266 ensures that the Pet Feeder is not only automated but also highly accessible and easy to control from anywhere.

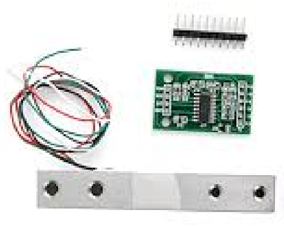

Figure 4. *Load Cell Sensor.*

Figure 5. *ESP8266 NodeMCU V1.0.*

5. **LCD 1602**
The LCD 1602 with I2C Interface is a versatile display module that provides real-time visual feedback for the Pet Feeder system, offering essential information directly on the device. With two rows of 16 characters, this display enables users to monitor feeding schedules, food levels, and the overall system status with ease. It presents key details such as the next feeding time, the amount of food to be dispensed, and important alerts like "Low Food Supply" or "Feeding in Progress." By connecting the LCD 1602 to the Arduino Uno via the I2C interface, the wiring is significantly simplified, requiring only two wires for communication, which also reduces the number of input/output pins needed on the microcontroller. This integration ensures seamless communication between the LCD display and other peripheral devices, such as the load cell, making the system more efficient and easier to expand if needed. The I2C interface enhances the overall functionality of the Pet Feeder, providing a user-friendly and accessible way for pet owners to stay informed about the feeder's operation without constantly checking a mobile application.

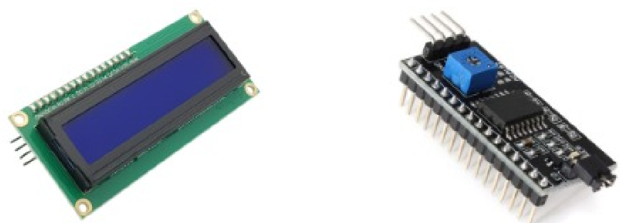

Figure 6. *LCD 1602 module (left) and I2C adapter module (right).*

6. **LED and Button**

The LED and Button play a key role in the Pet Feeder system, providing simple yet effective interaction and feedback. The button is used as an input device, allowing the user to initiate certain actions, such as starting or stopping the feeding process. The LED serves as a visual indicator, signaling the system's status, such as whether the feeder is active or in standby mode. By connecting the button to a digital input pin and the LED to a digital output pin on the Arduino, the system can easily detect button presses and control the LED accordingly. This interface ensures an intuitive and user-friendly experience while keeping the circuit design straightforward.

Figure 7. *Led (left) and Button (right).*

2.3. Electronic Circuit/Hardware Interfacing

The Pet Feeder system integrates various electronic components to ensure seamless operation, including an Arduino microcontroller, a servo motor, load cells, and an ESP8266 for Wi-Fi connectivity. The components are connected using the I2C communication protocol for efficiency, requiring only two wires for multiple connections, which simplifies the overall design.

Figure 8 shows the circuit schematic of the developed system, illustrating the connections between the Arduino, the various sensors, actuators, and the communication modules.

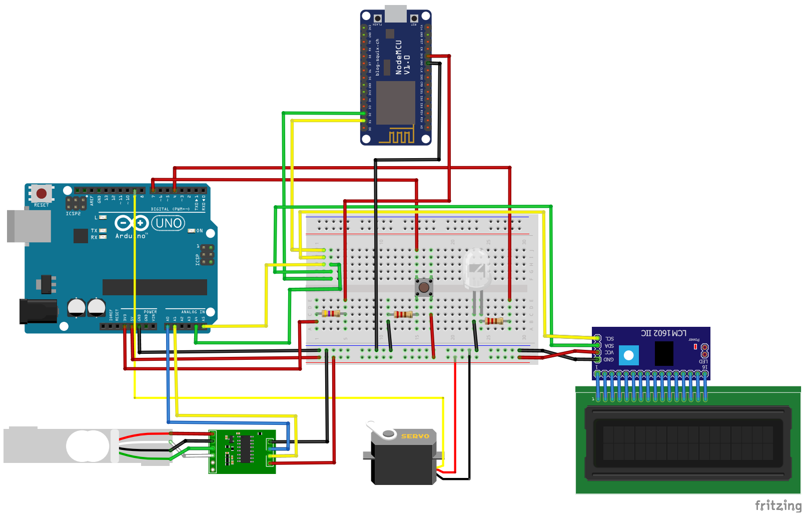

Figure 8. *Circuit schematic/hardware interfacing.*

Table 2. *Interfacing between Arduino Uno and its components (pin-to-pin) .*

Arduino	Servo Motor	HX711 Module	I2C Interface	Button	ESP8266 CH340	LED
GND	GND	GND	GND	PIN 1	GND	CATHODE
VCC (5V)	VCC	VCC	VCC	PIN 3		
3.3V					3.3V	
Pin 9	PWM					
Pin 7				PIN 2		
Pin 4						ANODE
Pin A0		DT				
Pin A1		SCK				
Pin A4			SDA		D2	
Pin A5			SCL		D1	

Table 3. *Interfacing between HX71 and CH340 (pin-to-pin).*

HX711 Module	Load Cell Sesnsor
E+	Red
E-	Black
A-	White
A+	Green

2.4. Programming Flowchart

The Pet Feeder system operates through a detailed programming flow designed to automate and manage the feeding process efficiently. As depicted in Figure 9 , the system follows a structured sequence of operations that integrate various sensors, user inputs, and predefined schedules to ensure that pets are fed accurately and on time.

The process begins with the system's initialization phase, where essential setup tasks are performed, such as calibrating the load sensor and establishing communication with the Blynk platform. Once initialized, the system enters its main operational loop, continuously monitoring three key inputs: the load sensor's value, the manual button state, and the current time against the pre-set feeding schedule.

Upon receiving data from the Blynk application, the system checks for any updates or commands from the user, ensuring that feeding times and portions align with the latest user preferences. The core decision-making process hinges on whether the manual button has been pressed or if the scheduled feeding time has been reached. If either condition is met, the system activates the servo motor to dispense food, with the amount monitored by the load sensor.

If the sensor detects that the food level is below a specified threshold after dispensing, the system will continue to dispense food until the appropriate level is reached, ensuring the pet receives the correct portion. Once the feeding process is complete, the servo motor closes, and the system resets to its monitoring state.

Throughout the operation, the system communicates with the Blynk platform, updating the user on the feeding status via the connected LCD display and the smartphone app. This real-time feedback allows the user to monitor and control the feeding process remotely, making adjustments as needed.

The flowchart illustrates how the Pet Feeder integrates hardware components and software logic to provide a reliable, user-friendly automated feeding solution. The system's robust design ensures that pets are fed consistently, whether managed manually by the user or automatically through pre-set schedules, enhancing the convenience and reliability of pet care.

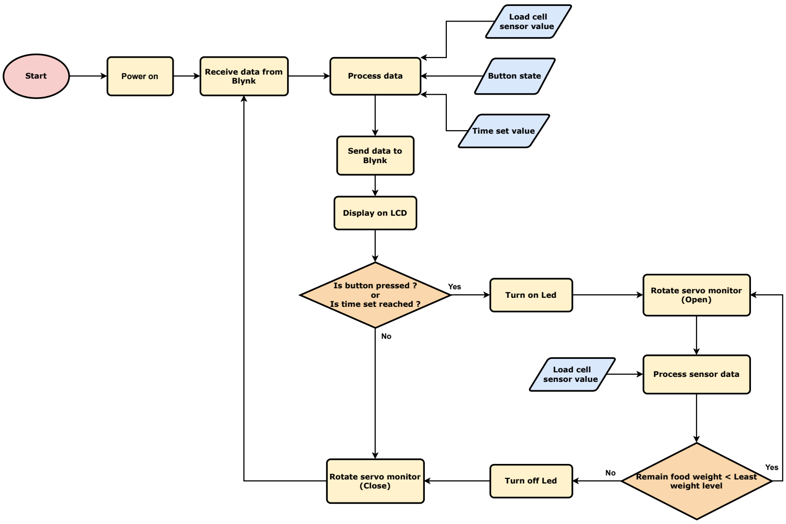

Figure 9. Flow diagram of the developed system.

3. Result and Discussion

3.1. Prototype Implementation

The **Pet Feeder System** prototype was developed and implemented based on the block diagram (Figure 9) and circuit design discussed in previous sections of the report. The key components used in this prototype include the Arduino Uno as the central microcontroller, responsible for managing and controlling the various inputs and outputs of the system. Other essential components, such as the servo motor, load sensor, ESP8266 NodeMCU, and LED indicators, were integrated into the system using a breadboard for prototyping purposes.

The programming code for the system was written in the Arduino IDE and flashed onto the Arduino Uno and ESP8266 NodeMCU to handle the system's operations. The code follows the logic outlined in the flowchart provided earlier, ensuring that feeding times and portion sizes are controlled automatically or manually, depending on the user's input through the Blynk mobile application.

The components were connected and tested on a full-size MB-102 breadboard, which allowed for easy interfacing of all peripherals. Most components were powered by a 5V DC power supply, except for the OLED screen, which required a 3.3V DC power source.

The prototype was tested to ensure the system's functionality, including checking the operation of the load sensor for accurate food level measurement, the servo motor for proper food dispensing, and the communication between the ESP8266 module and the mobile app for remote control. The integration of these components created a fully functional prototype of the Pet Feeder System, which was evaluated in real-world conditions to assess its effectiveness and reliability in automating pet feeding.

3.2. Results from Prototype Testing

The prototype pet feeder system was tested to evaluate the functionality and accuracy of its components, particularly focusing on weight measurements and user interface integration. During the testing process, weight readings and interface responses were observed under two separate configurations as displayed in the Serial Monitor outputs and the graphical interface.

The load cell, connected to both the NodeMCU and Arduino Uno, consistently measured the weight of the pet food within a range of 139.2 to 139.5 grams across multiple trials. In the Serial Monitor output connected to the NodeMCU, weight readings were recorded at one-second intervals, displaying minimal

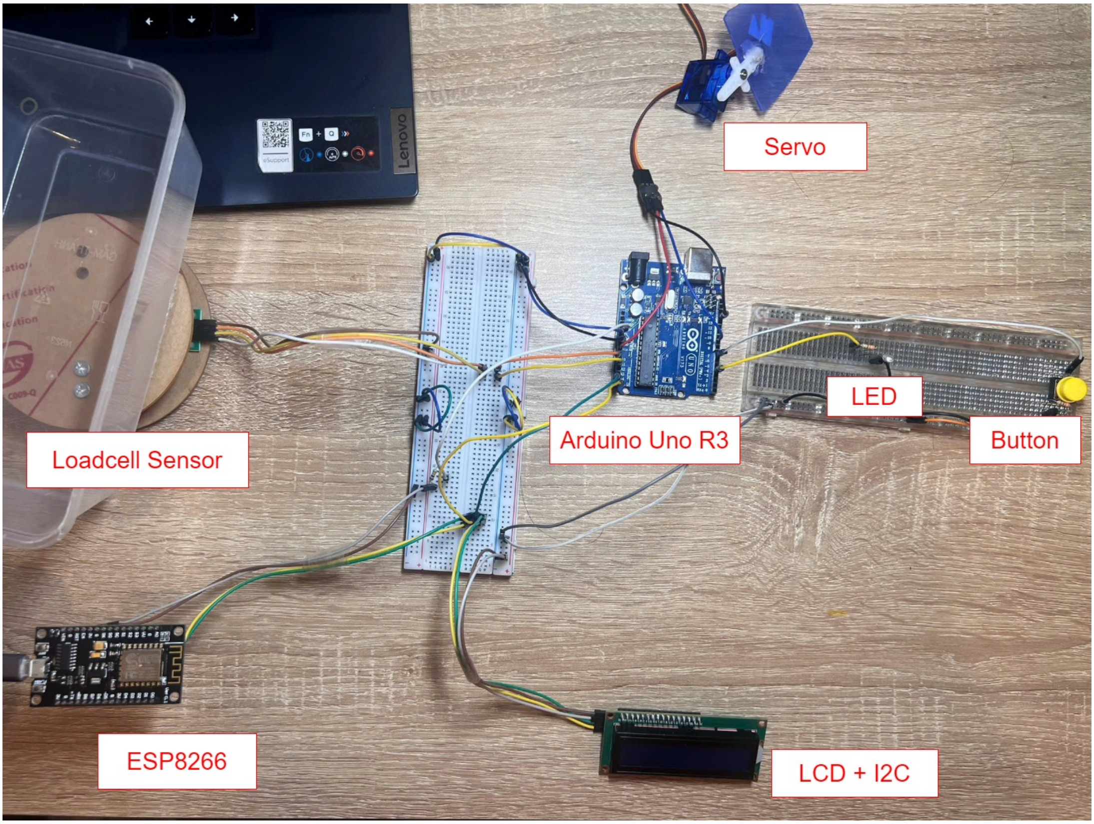

Figure 10. *The prototype implementation of pet feeder.*

fluctuations, which indicates stable sensor performance. Similarly, the Arduino Uno setup produced nearly identical results, confirming the load cell's accuracy after calibration. The minor variations in weight readings (±0.2 grams) suggest that the system is capable of maintaining precise food weight measurement, which is critical for the feeder's purpose.

The user interface, designed using the Blynk app, enabled real-time monitoring and control of the feeding schedule. Users could set specific feeding times and monitor the current food weight accurately, as shown in the graphical display. The interface consistently reflected the actual weight measurements recorded by the load cell, demonstrating effective communication between the hardware and software components. However, occasional delays were noted in data synchronization between the NodeMCU and the Blynk app, possibly due to network latency, which requires further optimization.

The physical hardware setup, consisting of components like the load cell, Arduino Uno, ESP8266, servo motor, LCD with I2C interface, LED, and button, was successful in ensuring stable operation. The servo motor responded promptly to commands from the user interface, facilitating controlled food dispensing. Meanwhile, the LED and button provided clear indications of operational status and manual feeding mode.

Overall, the prototype pet feeder demonstrated effective performance during testing, fulfilling its primary objectives in terms of weight measurement accuracy, user interaction, and hardware integration. Despite minor synchronization delays, the system proved to be reliable, with further improvements anticipated in data transmission speed and interface responsiveness in future iterations.

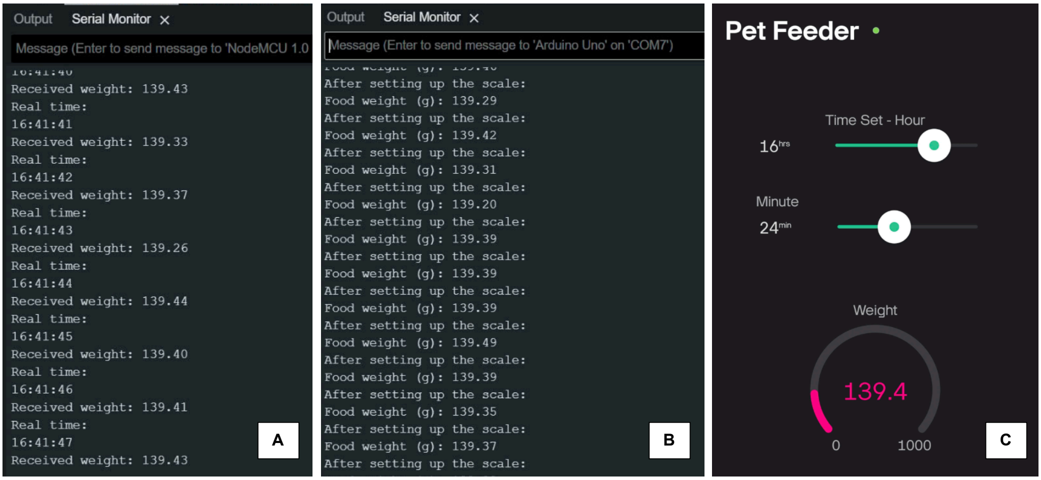

Figure 11. *Loadcell sensor data from the serial monitor of NodeMCU (A) and Arduino Uno (B). The graphical user interface (C) displays the time settings and current weight of the food in grams..*

3.3. Discussion

The test results indicate that the Pet Feeder system achieves high accuracy in food measurement and effectively automates the feeding process, especially with the support of the Blynk interface, which allows for easy remote control. However, there are certain limitations to improve, such as data synchronization delays between the NodeMCU and the Blynk app, which may affect the user experience when quick adjustments are needed.

Currently, the system lacks a function to measure the remaining food in the container to warn users when the supply is low. This can lead to potential food shortages that users may not be aware of, especially when they cannot check manually. Future enhancements will include this alert function, along with voice integration, allowing for improved interaction and ensuring that pets are consistently well-fed.

Overall, the Pet Feeder has proven effective in automating pet care, with strong potential for further development to provide a more convenient solution for busy pet owners.

4. Conclusion

The Pet Feeder system developed in this project provides an efficient and automated solution for managing pet feeding schedules. By integrating components such as the load cell sensor, HX711 module, ESP8266 Wi-Fi module, and a simple interface with buttons and LEDs, the system ensures precise food dispensing and real-time monitoring through a mobile application. The use of the I2C communication protocol and NodeMCU enhances the flexibility and scalability of the system, allowing for easy integration of additional features in the future.

This automated pet feeder improves the convenience and reliability of pet care, especially for busy pet owners who may not always be home during feeding times. Additionally, it ensures that pets are fed the correct amount of food, preventing overfeeding or underfeeding. With the system's user-friendly interface and its ability to remotely monitor and control the feeding process, this project demonstrates the practical application of IoT and embedded systems in everyday life. Future developments could include additional sensors or smart features, such as integration with voice assistants or more advanced data analytics for tracking feeding habits over time.

References

[1] **Adriansyah, Andi, Muchd Arief Wibowo, and Eko Ihsanto,** E. (2016). "Design of pet feeder using web server as internet of things application.", In *International conference on electrical engineering and informatics.* (pp. 151-156).

[2] **A. Manoj, T.G. Prasannakumar, V. S. Sathish Kumar, S. B. Surichandh and B. A. Saravanan,** (2020) Automatic Pet Feeder via IOT, *International Journal of Innovative Research in Science, Engineering and Technology,* vol. 9, no. 2, pp. 13865–13872, 2020, doi: 10.15680/IJIRSET.2020.0902073.

[3] **F. Mattern and C. Floerkemeier** (2010) “From the Internet of Computers to the Internet of Things.”, *Institute for Pervasive Computing.*

[4] **Adriansyah, A., Wibowo, M. A., and Ihsanto, E.** (2016). Design of pet feeder using web server as internet of things application. In *International conference on electrical engineering and informatics* (pp. 151-156).