

Executive Summary

PetByte, our automated pet feeder, is a smart device designed to help pet owners feed their pets remotely through an internet-connected system. It combines hardware automation with cloud-based communication and a mobile interface to allow scheduled feeding, manual feed activation, and monitoring. The system aims to ensure consistent and safe feeding even when the owner is not physically present.

PetByte integrates four major components: the Raspberry Pi Hardware controls sensors, motor, camera and executes feeding commands; the Network manages communication between PetByte and the mobile app through APIs; the Mobile App provides an interface for users to schedule feedings, view live video, and receive notifications; Computer Vision is responsible for the detection and classification of the pet captured by the camera.

Background

Pet ownership in the United States continues to rise, with millions of households relying on consistent feeding routines to maintain their pets' health. However, busy professionals, elderly individuals, and families with children often struggle to maintain those routines due to time constraints, forgetfulness, or physical limitations; putting pets at risk of irregular nutrition.

A survey done in 2025 by the American Pet Products Association recorded United States households having 53 million people own cats and 71 million people own dogs [1]. Each of these households must feed their pet every day at different times according to the pet owner's schedule. Proper nutrition is a basic necessity for all pets, and inconsistent feeding leads to "nutrient deficiencies and imbalances" [2]. It's important for pet owners to have a reliable way to feed their pet to maintain the best health possible.

An automated pet feeder addresses this gap by combining hardware automation, cloud communication, and a mobile interface to give pet owners complete control over their pet's diet from anywhere, at any time.

Experimental Setup

SPECIFICATIONS – Remote scheduling, manual feeding, and portion control via mobile app. Push notifications for pet activity, meal completion, and low food levels. Live video feed and pet detection through camera and PIR sensor.

CONSTRAINTS – Requires stable Wi-Fi for full functionality. Fixed camera angle limits pet tracking range. Rpi compute limitations. Pet Feeder needs to be connected to power.

ASSUMPTIONS – Users own an iOS or Android device with notifications enabled. Stable home internet connection is available.

CONCEPTUAL DESIGN – Four-layer modular architecture: Mobile App, Network/Storage, Pet Feeder, Computer Vision. Camera and PIR sensor mounted low at the front of the device for optimal pet detection.

JUSTIFICATION – Raspberry Pi over Arduino: needed Python, camera support, and compute for CV. LiteRT over full TensorFlow: lower memory usage, sufficient accuracy. SQLite over PostgreSQL: lightweight, no server overhead, fits project scope

Experimental Test Plan

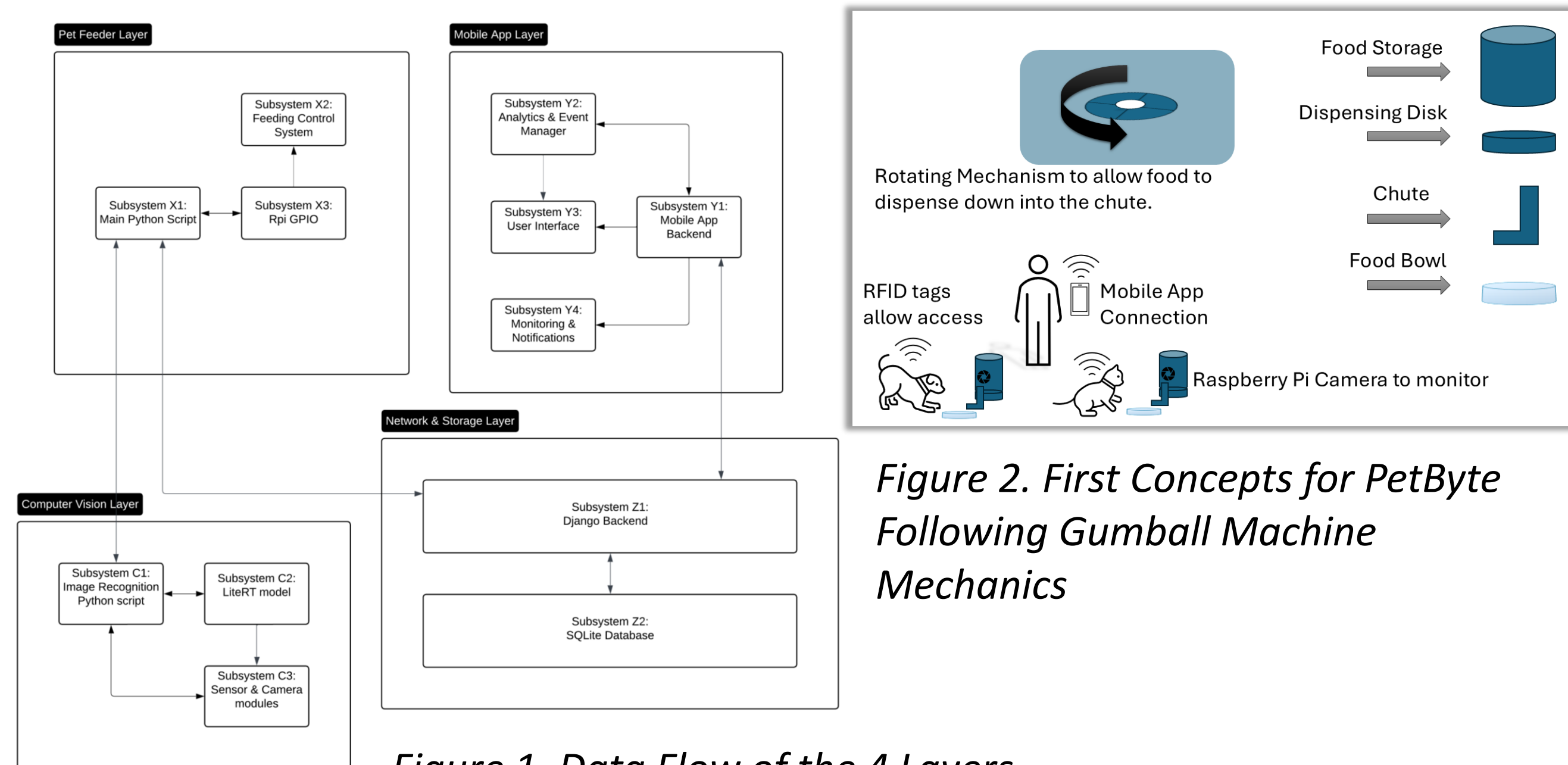


Figure 1. Data Flow of the 4 Layers.

PET FEEDER LAYER – Runs on the Raspberry Pi. The main Python script controls food dispensing via a servo motor; interfaces with GPIO pins to manage sensors and motors, and relays pet detection signals from the CV layer to the mobile app.

MOBILE APP LAYER – Provides the U.I. including dashboard, feeding scheduling, portion control, live camera feed, and settings. Logs pet eating patterns, syncs data with the backend in real time, and delivers push notifications for feeding events, low food alerts, and feeder errors.

COMPUTER VISION LAYER –Represents the utilization of a connected Raspberry Pi Camera to detect any potential cats or dogs in frame via TensorFlow model.

NETWORK AND STORAGE LAYER – Django backend manages bidirectional data between the mobile app and PetByte, handling API requests, authentication, and push notifications. SQLite database stores feeding schedules, user data, and event logs, allowing the app and feeder to retrieve data without requiring a shared network.

Experimental Results

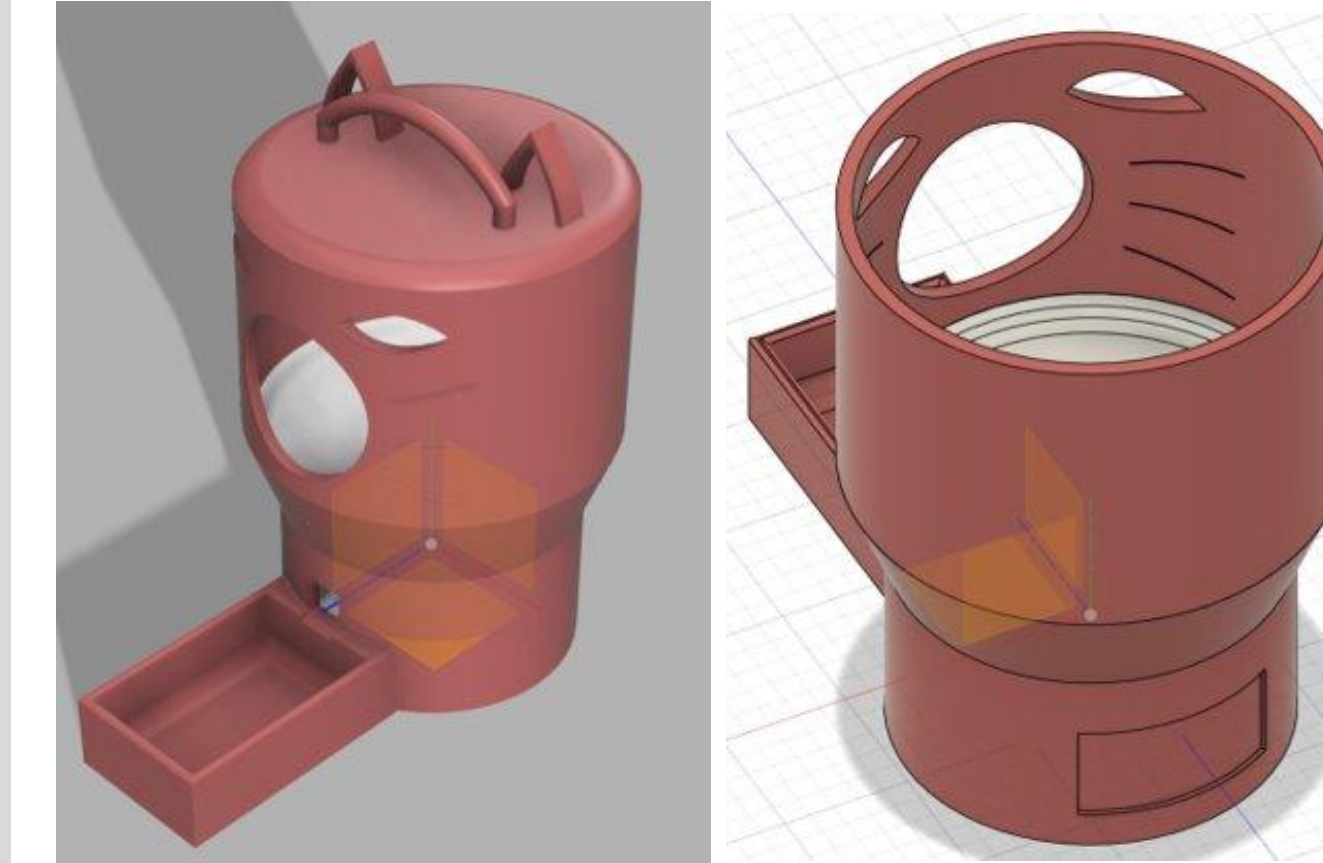


Figure 3. Initial Idea for PetByte.

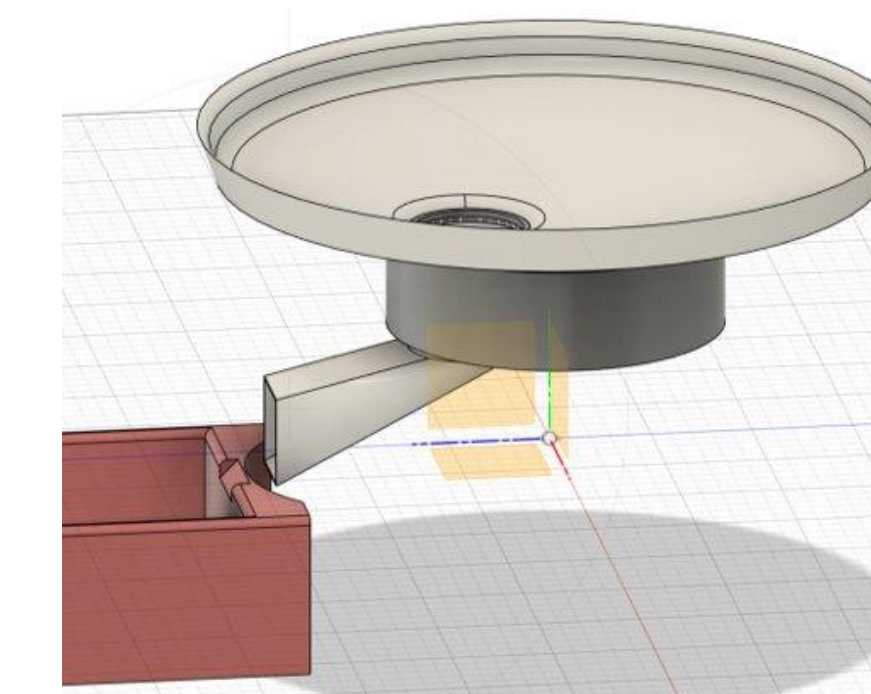


Figure 4. Initial Idea for dispensing food with inner funnel and chute.

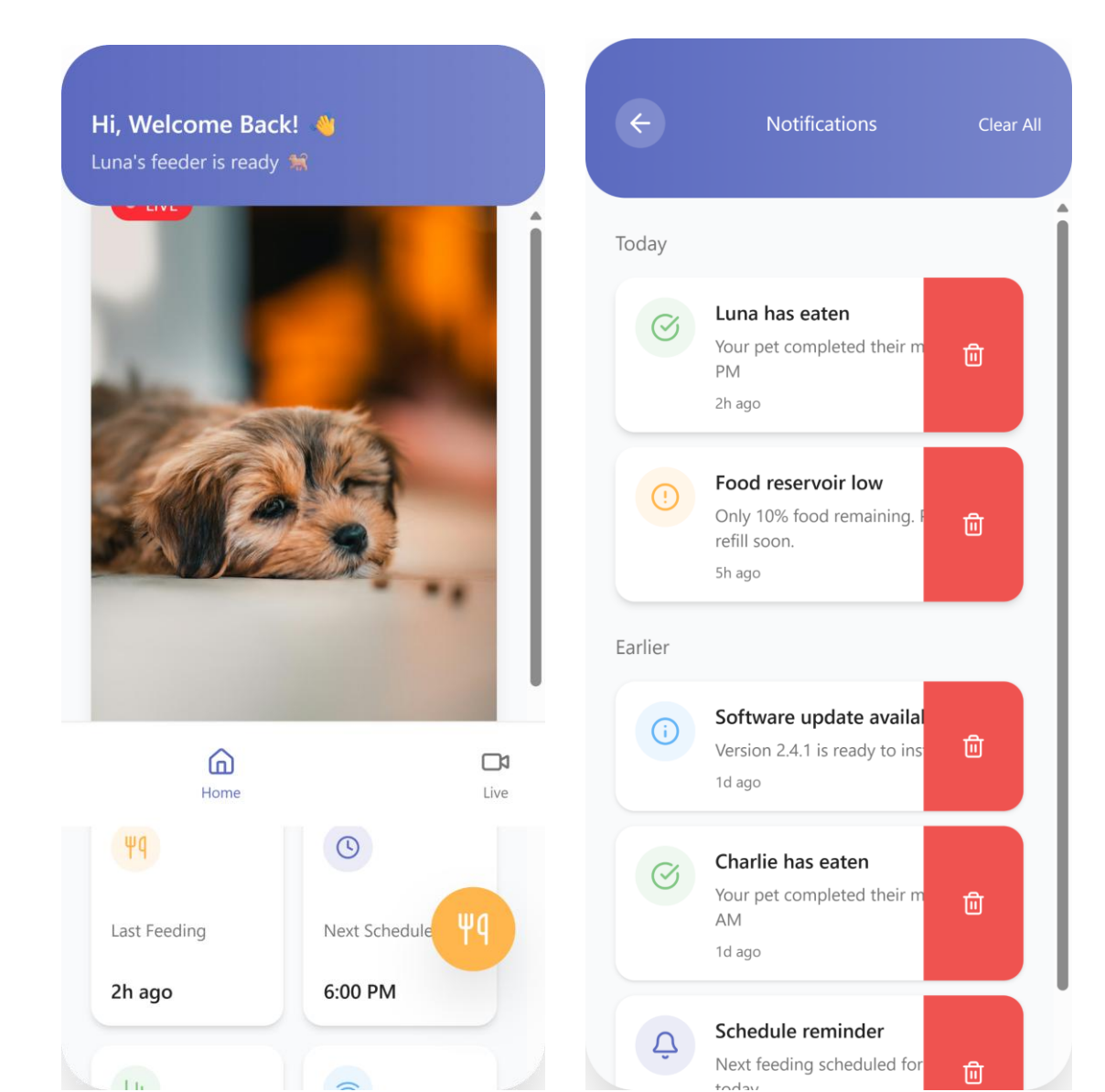


Figure 5. Mockup of the Mobile App: Home Dashboard and Notifications.

Conclusions

PetByte went through many different changes as the process of 3D printing and testing software components went on. During testing, the base needed to be improved by increasing the height to fit the inner components (Rpi, breadboard) and for the wires to go through. And the bowl design was changed to fit the new base. And as we were 3D printing the model, there were edits that needed to be made for overall structure and stability. Although PetByte has seen its many variations, it is a unique pet feeder that busy pet owners can use to care for their pet's nutrition.



Figure 6. Prototypes and some Final Versions of PetByte over time.

References

1. American Pet Products Association. "Industry Trends and Stats." APPA, 2025, americanpetproducts.org/industry-trends-and-stats.
2. Dodd, S., Cave, N., Abood, S., Shoveller, A.-K., Adolphe, J. and Verbrugghe, A. (2020), An observational study of pet feeding practices and how these have changed between 2008 and 2018. Veterinary Record, 186: 643-643. <https://doi.org/10.1136/vr.105828>