

# PiFeed: Feed your pets with a Raspberry Pi

Danny Duangphachanh, Daniel Friedman, Igor Janjic

Bradley Department of Electrical and Computer Engineering, Virginia Tech [ECE 4564]

## Objectives

The purpose of PiFeed is to be able to remotely monitor and control a fish tank and cat feeder from the internet. The ability to maintain a healthy eating schedule for pets is a concern for members of our group, as well as many others who travel or are away from home for an extended amount of time. By successfully implementing a pet feeding and monitoring system using Raspberry Pis, the stresses of animal care while away will be a thing of the past!

- The system is composed of two Raspberry Pi's and one computer.
  - Rasp1 collects information about the fish tank and controls the fish feeder.
  - Rasp2 collects information about the cat and control the cat feeder.
- A user will interface with the two Raspberry Pi's with a client application
  - It is able to monitor both the aquarium and the cat feeder using Pi cameras.
  - It also allows the user to customize the feeders and control them manually.
  - We will consider the system successful if our pets can be fed remotely.

## Introduction

There are two Rasperry Pi's in this setup. RASPF1 is the Rasperry Pi that controls the hardware for feeding for the fish tank. RASPC1 is the Rasperry Pi that controls the hardware for feeding for the cat. PiFeed is separated into the following modules: PiFeedControl, PiFeedFish, and PiFeedCat. The programming language that was used to implement the feeding system is written in Python. PiFeedControl is a program that is run on a client computer that has access to the internet. It is used to remotely control the feeders using a custom socket protocol. PiFeedFish is a program that is run on RASPF1 while PiFeedCat is a program run on RASPC1. Each of these programs is responsible for controlling the camera, sensors, and feeder servos.

## Hardware Description

The following hardware was used to construct the feeding system:

- 2 Raspberry Pis
- 2 SD cards
- 2 WiFli + Bluetooth 3.0 USB adapters
- 2 MicroUSB cables
- 2 USB chargers
- 2 5MP camera board modules
- 1 Cheap LCD
- 1 Servo
- 1 Servo board
- 1 H bridge
- 1 DHT22 Temperature + Humidity sensor
- Jumper wires

## Methods

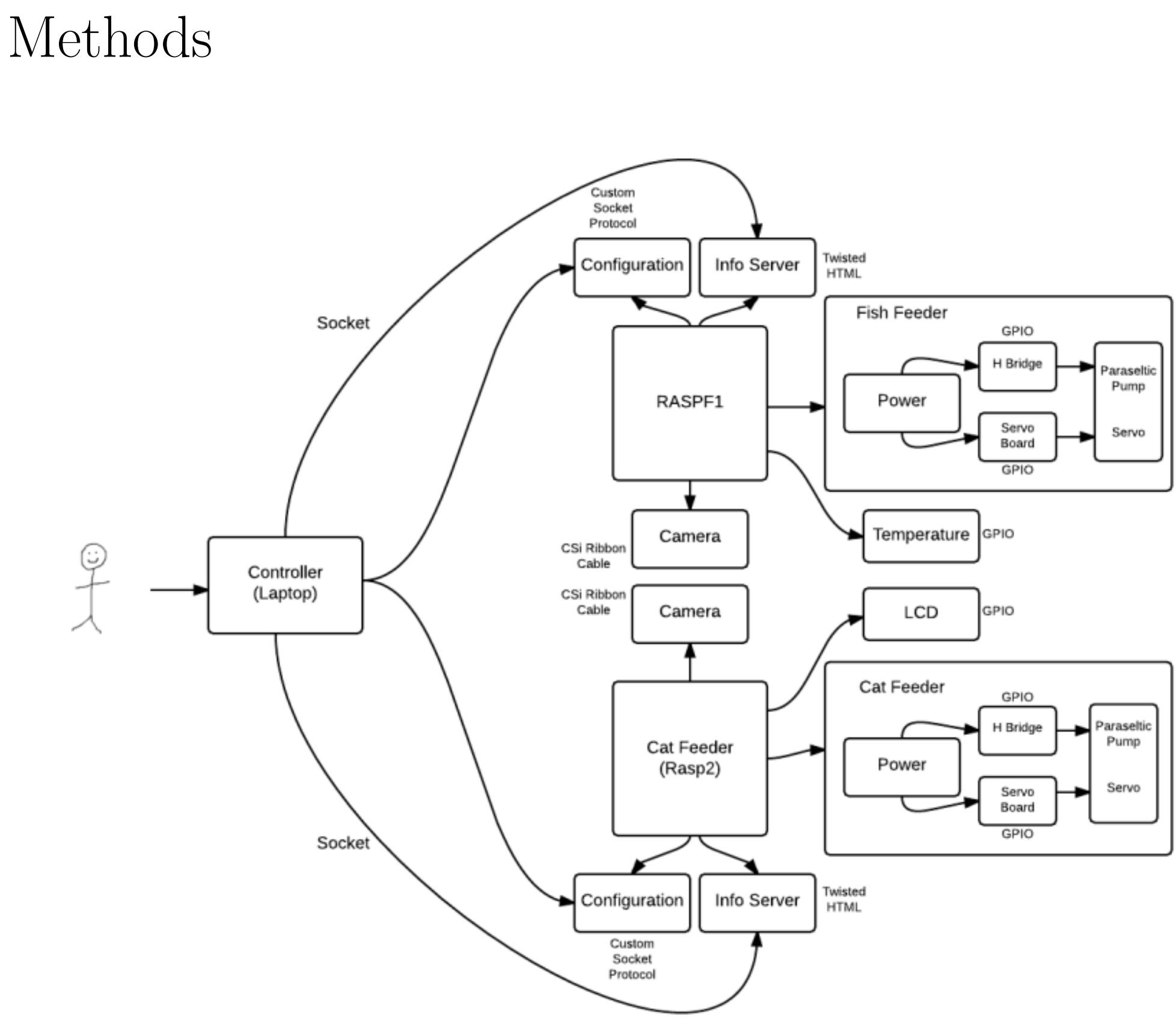


Figure 1: Use case diagram of the system.

## GitHub Page

— <https://github.com/zergler/PiFeed> —

## Testable Requirements

A completely successful project should demonstrate that the following requirements are satisfied.

- Able to feed the fish locally, remotely, manually and at specific times of the day.
- Able to feed the cat both locally, remotely, manually and at specific times of the day.
- Able to monitor the fish locally and remotely using the Pi camera.
- Able to monitor the cat locally and remotely using the Pi camera.
- Able to view the fish tank sensors locally and remotely.
- Able to view the cat sensors locally and remotely.

## Results

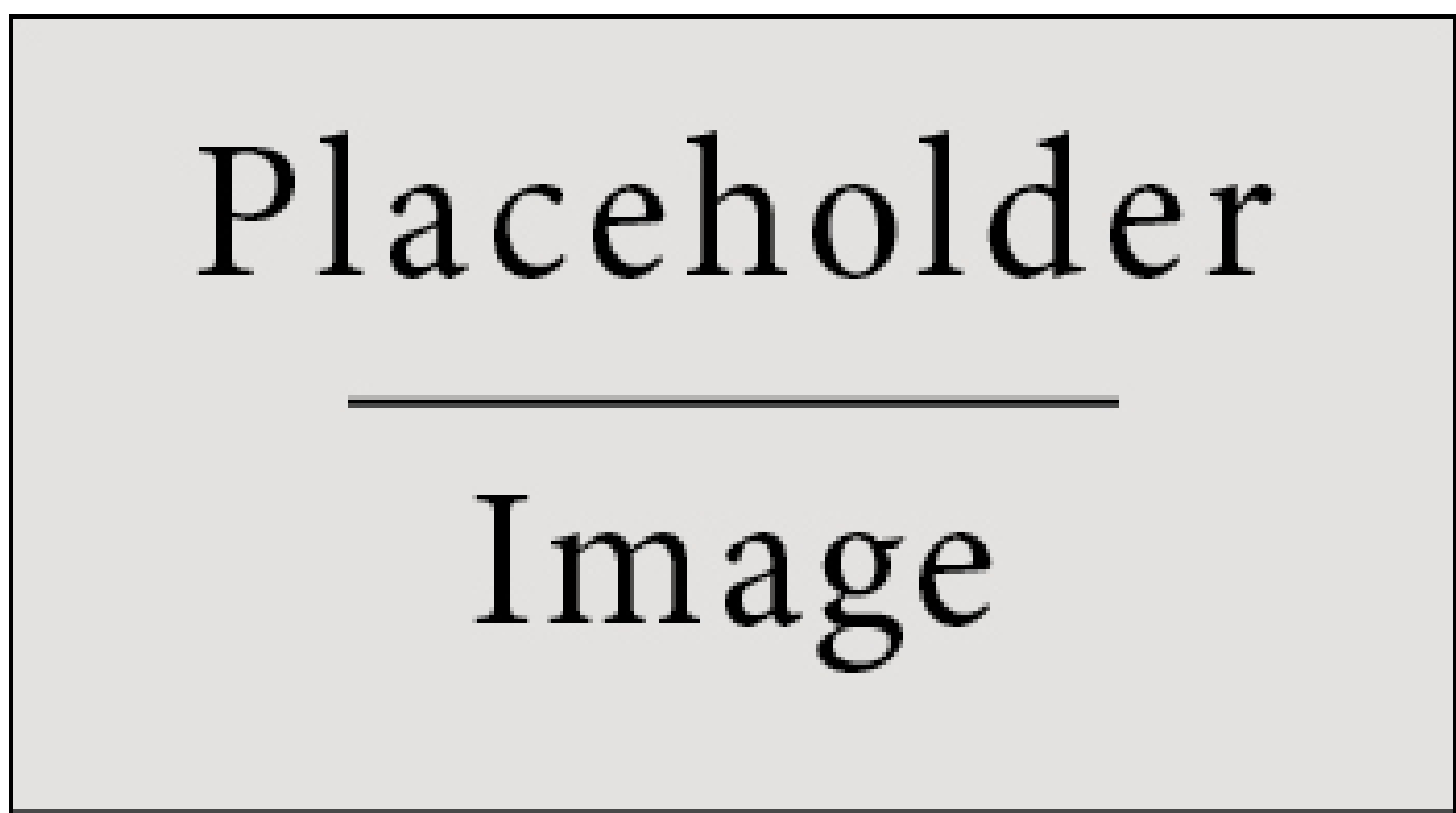


Figure 2: Figure caption

Nunc tempus venenatis facilisis. Curabitur suscipit consequat eros non porttitor. Sed a massa dolor, id ornare enim:

| Treatments  | Response 1 | Response 2 |
|-------------|------------|------------|
| Treatment 1 | 0.0003262  | 0.562      |
| Treatment 2 | 0.0015681  | 0.910      |
| Treatment 3 | 0.0009271  | 0.296      |

Table 1: Table caption

## Conclusion

PiFeed is a system of connected Raspberry Pi units that allow the user to care for their cat and/or fish locally and/or remotely. The PiFeed system is capable of maintaining a feeding schedule and setting food for your pet cats and/or fish. The PiFeed system also allows the user to manually control when a feeding occurs by accessing the feeder via web service. Both the cat feeder and fish feeder may be monitored through the use of a Raspberry Pi camera and may be seen via web service. For the cat feeder, a sensor is used to detect the presence of a cat and allows the distribution of food once detected.

## Additional Information

Maecenas ultricies feugiat velit non mattis. Fusce tempus arcu id ligula varius dictum.

- Curabitur pellentesque dignissim
- Eu facilisis est tempus quis
- Duis porta consequat lorem

## References

## Acknowledgements

Nam mollis tristique neque eu luctus. Suspendisse rutrum congue nisi sed convallis. Aenean id neque dolor. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas.

## Contact Information

- Danny Duangphachanh: bboydd@vt.edu
- Daniel Friedman: adfriedman@vt.edu
- Igor Janjic: ijanjic@vt.edu

