

**Class:** ECE/CS585

**Team Name:** Cold Fries

**Members:**

Kenny Do

Michael Nguyen

Kevin Perez

Davin Sonexarth

Khai Vu

Date: 10/24/21

**ARF FEEDER DEFINITION**

The ARF feeder will be an automatic pet feeder that the user will be able to control from an online client. The purpose for the remote feeder is to feed the pet whenever the owner is unable to physically do so. In addition, there will be other remote access to it allowing for scheduled feeding and more autonomy. The reasoning behind the autonomous remote feeder (ARF) is to target the demographic of those who either: 1) tend to forget to feed their pets at certain times/unable to consistently feed their pets at certain times, 2) those who travel for small periods of time and have to feed their pet, 3) when having to feed a pet during times such as school and/or work but are unable to do so in person.

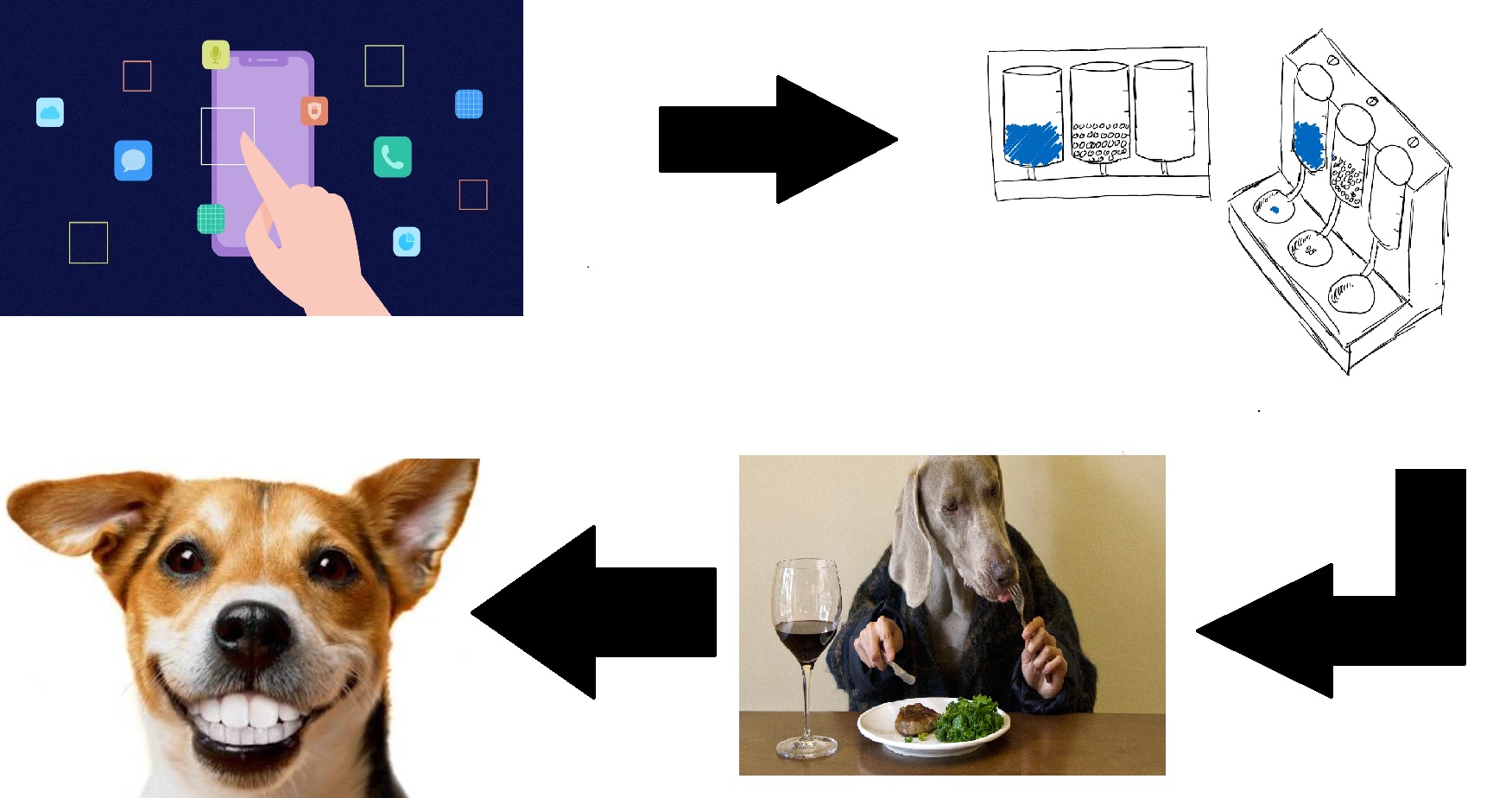


Diagram as to how the ARF pet feeder works

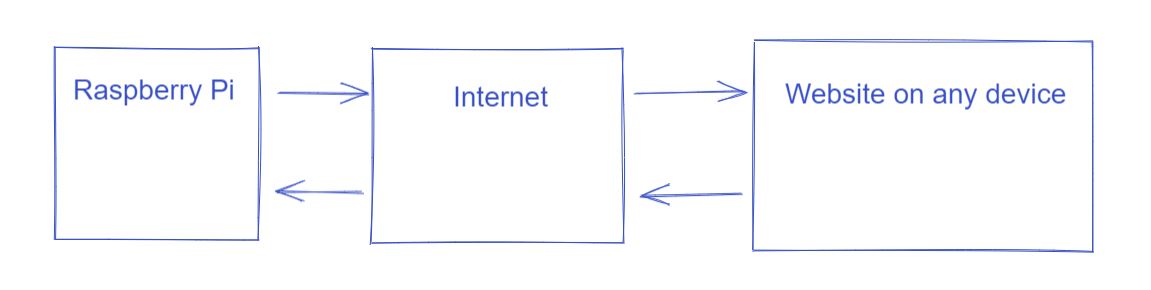
REALISTIC CONSTRAINTS OF THE ARF

Realistically, time and budget are limited which factor into the outcome of the project. There were ideas that we wanted to incorporate but were limited by the time and our own efficiency constraints. Some ideas that we wanted to have (which may be included in the final project but will be cited here anyway in case of not being able to complete them):

* LCD screen that would be programmable by the user in person instead of through the app [9]
* A camera that would show feeding times and either take pictures for the user to see through their IOT connected app or video feed of their pet [3]
* A speaker/microphone system in which the user could communicate with and hear their pet if wanted
* Variable sized feeding mechanism (in which we could distribute by certain weights instead of the set unit of cups like we intend)
* Better system to incorporate the wifi adaptability and functions into as custom smart device instead of using the raspberry pi that is programmable)
* Electrical systems would be better routed and fit towards the components and their power draw/protection instead of generalized and rated products we will use (such as PCB boards, resistors, etc.)
* Modular system in which we could connect multiple feeders at once and seeing them as one unit rather than multiple
* Creation of accounts so that people can access their devices from any device, using one master account
* Sensors alerting the user when food/water is low, if there is a possible jam, leakages, etc
* Detachable feeding bowls that would use a screw sort of mechanism so that it can be easily cleaned in a sink/dishwasher
* The frame of the food dispenser would use a different type of silicon/plastic that is more structurally sound but still light to pick up instead of PLA filament used for 3D printing and wood
* Would make the pet feeder deconstructable so that it can be taken apart and assembled anywhere without having to haul the large pet feeder around
* We could make a version that work operate well outside that would also have solar cells to charge the battery as it shines (as our pet feeder is indoor only)
* Use a better battery that is meant for long term use and exposure that would also be rechargeable
* Implement a better surge protector so that a surge in current will not destroy the technology
* All components would be custom made to fit the pet feeder
* If we had more time we could design a PCB specifically for our project that only has inputs and outputs desired and we could keep the cost down for mass production.
* A pump for the water bowl portion to keep the animals water fresh and happy for the pets/consumer

REQUIRED SPECIFICATIONS

Description:



ARF is a physical product/device that needs a mandatory internet connection along with the usage of a web browser to access the ARF website. We based our product requirements off of these conditions.

We will be using GitHub as our main website hosting service. We use GitHub since it is an extremely popular repository with many features including free web-hosting that is available 24/7 and is able to run our website without the need for constant maintenance.

Github can provide a way to host the web interface since it can take in simple HTML files and other related file types to base the website on. The HTML files allow for the use of web interface elements such as hyperlinks, redirects, buttons, as well as any other needed function. These features would allow for functionality between the raspberry pi and the web client. which we can use to send signals through the internet to our raspberry pi endpoint.

| Requirements | Specifications | Details/Standards |
| --- | --- | --- |
| **Must be indoors only** | Internal temperature of 30-90F | Our current design only allows for indoor use |
| **Product Dimensions** | L30” x W30” x H25” (Water will be separate for presentation)  Lip height 3”  L30” x W30” x H4”(Where electrical will go in) | These are the dimensions that will incorporate all of the components such as the two sections for pet feeding and also the water reservoir  This section is part of the product dimensions, but where the components will lay |
| **Electrical plug w/ passthrough** |  | This is to supply the constant power needed for the ARF feeder, the purpose of the passthrough is so that it opens up portion for the user to use the outlet |
| **Power bank Model BWA10WI199** | * 20.1Ah * 72.36 Wh * Input: 5V @ 2A * Output: 5V @ 2.4A/ 4.8A for both ports | Keeps power from our plug-in stored, and can regulate different power flows into our raspberry pi and other functions such as the motor or LED lights.  [14] |
| **Wall plug-Switching Adaptor w/ male connecting micro-usb** | * Input: 100-240V @ 50/60 Hz w/ 0.4 A * Output: 5V w/ 2.5A | [1](https://www.microcenter.com/product/627789/vilros-raspberry-pi-zero-w-complete-starter-kit) |
| **Raspberry Pi Zero W** | * Dimensions: 65mm × 30mm × 5mm * SoC: Broadcom BCM2835 * CPU: ARM11 running at 1GHz * RAM: 512MB * Wireless: 2.4GHz 802.11n wireless LAN * Bluetooth: Bluetooth Classic 4.1 and Bluetooth LE * Power: 5V, supplied via micro USB connector * Video & Audio: 1080P HD video & stereo audio via mini-HDMI connector * Storage: MicroSD card * Output: Micro USB * GPIO: 40-pin GPIO, unpopulated * Pins: Run mode, unpopulated; RCA composite, unpopulated * Camera Serial Interface (CSI) | [1](https://www.microcenter.com/product/627789/vilros-raspberry-pi-zero-w-complete-starter-kit)  Used to connect the mechanical/electrical aspects to the IoT, where the program will tell the electrical what to do. |
| * **ROHS STEP MOTOR 28BYJ-48 5VDC** * **1208 Stepper** | * 5VDC * 4 phases * 0.5A at 10 VDC | [4](https://components101.com/motors/28byj-48-stepper-motor) used for prototyping  [5](https://www.digikey.com/en/products/detail/pololu-corporation/1208/10449951?utm_adgroup=Motors%2C%20Solenoids%2C%20Driver%20Boards%2FModules&utm_source=google&utm_medium=cpc&utm_campaign=Shopping_DK%2BSupplier_Pololu%20Corporation&utm_term=&utm_content=Motors%2C%20Solenoids%2C%20Driver%20Boards%2FModules&gclid=CjwKCAjw-sqKBhBjEiwAVaQ9a5ydxkZlHlWfqFQGc_q4UutAwQBrupnOzT-7mEiJKhpNJ1s2jnJi-xoCVDEQAvD_BwE) something larger for final project |
| **Inland Relay Module X 4** | * Type: Digital * Rated current: 10A normally open and 5A normally closed * Max switching voltage 150VAC/ 24 VDC * Max switching power: Digital * Contact action time: 10 ms | [2](https://www.microcenter.com/product/617949/inland-single-5v-relay-module-for-arduino) used to power components that need more power than the raspberry pi can put out. We have plenty but are unsure how many we will use yet. Depending on how many components we can successfully add to the project. |
| **Wire sizing** | * AWG 40 through AWG 24. * No specifics here but we won’t have any loads exceeding 2 amps at once on any wire so this range should be fine. | [10](https://www.youtube.com/watch?v=KSYAqQ4xyJI),[11](https://en.wikipedia.org/wiki/American_wire_gauge) |
| **Diode and NPN transistors to be decided upon** |  |  |
| **5 inch LCD** | * 800X400 * HDMI * Touch compatible if using GPIO | [9](https://www.aliexpress.com/item/1005002547669690.html?spm=a2g0o.cart.0.0.3d0a3c00mkyyP4&mp=1) |
| **Raspberry Pi Camera** | * 5 megapixel * 16cm * 3 grams | [3](https://www.aliexpress.com/item/32901067278.html?src=google&memo1=freelisting&src=google&albch=shopping&acnt=708-803-3821&slnk=&plac=&mtctp=&albbt=Google_7_shopping&albagn=888888&isSmbAutoCall=false&needSmbHouyi=false&albcp=1582410664&albag=59754279756&trgt=539263010115&crea=en32901067278&netw=u&device=c&albpg=539263010115&albpd=en32901067278&aff_fcid=d87b284daad54c148305a71495b0721b-1635123333104-01894-UneMJZVf&aff_fsk=UneMJZVf&aff_platform=aaf&sk=UneMJZVf&aff_trace_key=d87b284daad54c148305a71495b0721b-1635123333104-01894-UneMJZVf&terminal_id=34f97073c9ae460fa1a73a0d921c0553) |
| Looking for solenoid to control a door to stop food from getting out and animals getting in from the bottom up. |  | [8](https://core-electronics.com.au/tutorials/solenoid-control-with-raspberry-pi-relay.html) |
| The current/voltage output of a wall outlet must be taken into consideration (this is where we will plug it into) | [16] Most house hold outlets have a breaker of 15A and 120V in the US. Plenty of power for our project. | This will supply a constant flow in power that is regulated from the wall itself, pumping the amount of power stated |
| Surge protection of some sort (most likely from a surge protector) |  | Prevents electrical shortage from things such as storms to keep the device safe |
| **Wood/ PLA filament/ Epoxy/ Resin/ screws/water jug** |  | These are very cheap materials to work and build with compared to metals. |

ENGINEERING STANDARDS

When looking at the CS aspect, something that we have to take into consideration are hackers and glitches. When it comes to glitches (or bugs) we find that these can effectively cause our product to no longer work. The only way to fix a glitch is through software bug fixes and patches. In order to fix that, we can send out updates if needed, but also, we can make the code simple enough so that there is no room for error in terms of glitches, in just terms of I/O. When it comes to hackers trying to get into the website, we will need a sort of encryption/decryption to keep account information safe while also making it harder for third parties to gain access to this. We can also implement a hardware based security component that prevents tampering and also has tampering evident finders so that if someone tries to access the raspberry pi, we will know if it happens. By implementing a tamper preventative device, this allows for randomness of encryption/decryption while also preventing any access to information before bootup as the device creates the key and other preventatives before the OS boots up (so that malicious programs and third parties cannot access it).

When looking at the electrical aspect, something that needs to be taken into consideration is surge protection. It is a common thing where a burst of current goes through a device, frying them from the inside. By putting a surge protector in place, this will effectively block the excess amount of current going through, keeping the devices that are plugged in to be safe (in most cases). Another thing that we need to look into are temperatures. Since we plan on keeping the devices plugged in 24/7, we want the temperatures of the devices to stay relatively cool so that they do not overheat. Since the constant flow of electricity causes heat, too much heat may shorten/kill a device. By creating heat dissipation or keeping the heat at a constant temperature all round, this will prevent the overheating of devices. Luckily, our module will be inside and this should help the temperature range and weathering greatly.

PROTOTYPE MODEL SPECIFICATIONS

We want to be able to test our concept to see if it is achievable. The main components to make ARF a working product are the mechanisms and internet connection. This will require testing and application of motors and wiring, alongside testing to see if signals can move from the raspberry pi to the internet and vice versa reliably.

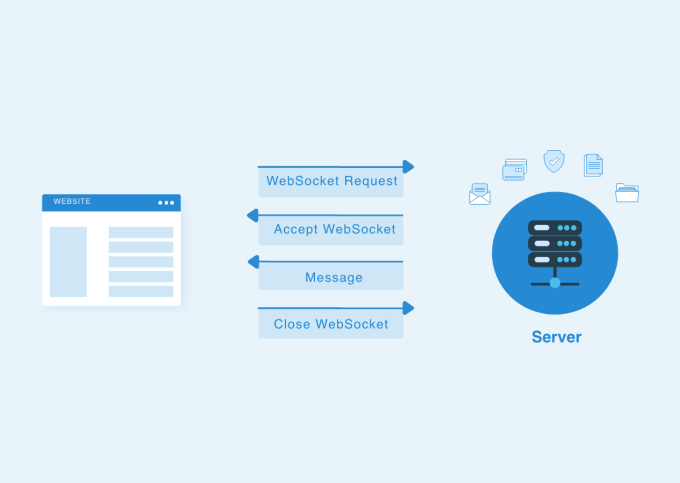
**Electrical and mechanical wiring:**

The main component will be centered around the Raspberry Pi, which will connect the mechanical and electrical side to the programming side. With the solenoids, camera, screen unable to do anything without an input/output signal, the raspberry pi will be able to translate the connectivity and signals into input/output dependent on what the user inputs on the program side. Since this will all require a lot of power, we decided to connect it up to a battery so that even if the electrical connection is cut in some way (such as blackout), there will still be a connection until the battery drains. To keep constant power going, it will be plugged directly into the wall, supplying the electrical power needed for the device to function.

**Wifi and Internet connectivity:**

A raspberry pi has a built-in wifi card which allows it to connect to the internet (7). We will be utilizing said feature alongside GitHub Pages (6). GitHub Pages allows us to take html, CSS, and javascript files and lets us host them as a website by creating a repository and links it directly through the github domain. Given that we have the internet. There are many repositories available for us to reference. Example being github buttons that are on Github and can be accessed at any point. Being able to do all the work within the ARF github allows us to track the progress and efficiently add to the code and work on the connectivity.

We accomplish this by using websockets and Node.js, a javascript runtime environment (12). This allows us to communicate in both directions on a HTTP server to the raspberry pi.



By putting our names here, we acknowledge that we have all read and certify this document to be the work done to the best of our abilities by each individual and group standards.

Kenny Do, P969V743  
Kevin Perez, A453B383

Khai Vu, U326Q228

Davin Sonexarth, S555D296

Michael Nguyen, P955Z557

**Sources**

1. [**https://www.microcenter.com/product/627789/vilros-raspberry-pi-zero-w-complete-starter-kit**](https://www.microcenter.com/product/627789/vilros-raspberry-pi-zero-w-complete-starter-kit) **[1]**
2. <https://www.microcenter.com/product/617949/inland-single-5v-relay-module-for-arduino> **[2]**
3. <https://www.aliexpress.com/item/32901067278.html?src=google&memo1=freelisting&src=google&albch=shopping&acnt=708-803-3821&slnk=&plac=&mtctp=&albbt=Google_7_shopping&albagn=888888&isSmbAutoCall=false&needSmbHouyi=false&albcp=1582410664&albag=59754279756&trgt=539263010115&crea=en32901067278&netw=u&device=c&albpg=539263010115&albpd=en32901067278&aff_fcid=d87b284daad54c148305a71495b0721b-1635123333104-01894-UneMJZVf&aff_fsk=UneMJZVf&aff_platform=aaf&sk=UneMJZVf&aff_trace_key=d87b284daad54c148305a71495b0721b-1635123333104-01894-UneMJZVf&terminal_id=34f97073c9ae460fa1a73a0d921c0553> [3]
4. <https://components101.com/motors/28byj-48-stepper-motor> [4]
5. [DIGI KEY STEPPER](https://www.digikey.com/en/products/detail/pololu-corporation/1208/10449951?utm_adgroup=Motors%2C%20Solenoids%2C%20Driver%20Boards%2FModules&utm_source=google&utm_medium=cpc&utm_campaign=Shopping_DK%2BSupplier_Pololu%20Corporation&utm_term=&utm_content=Motors%2C%20Solenoids%2C%20Driver%20Boards%2FModules&gclid=CjwKCAjw-sqKBhBjEiwAVaQ9a5ydxkZlHlWfqFQGc_q4UutAwQBrupnOzT-7mEiJKhpNJ1s2jnJi-xoCVDEQAvD_BwE) [5]
6. <https://docs.github.com/en/pages/getting-started-with-github-pages/about-github-pages> \*
7. <https://socialcompare.com/en/comparison/raspberrypi-models-comparison> \*
8. <https://core-electronics.com.au/tutorials/solenoid-control-with-raspberry-pi-relay.html> [8]
9. <https://www.aliexpress.com/item/1005002547669690.html?src=google&aff_fcid=6600a001a21a46ad801fde042fa6ba21-1635120161648-05596-UneMJZVf&aff_fsk=UneMJZVf&aff_platform=aaf&sk=UneMJZVf&aff_trace_key=6600a001a21a46ad801fde042fa6ba21-1635120161648-05596-UneMJZVf&terminal_id=34f97073c9ae460fa1a73a0d921c0553> [9]
10. <https://www.youtube.com/watch?v=KSYAqQ4xyJI> [10]
11. <https://en.wikipedia.org/wiki/American_wire_gauge> [11]
12. <https://www.w3schools.com/nodejs/nodejs_raspberrypi_webserver_websocket.asp> \*
13. <https://towardsdatascience.com/how-to-create-a-free-github-pages-website-53743d7524e1>
14. <https://www.walmart.com/ip/Blackweb-20000mAh-Power-Bank-Black/247544454> [14]
15. <https://github.com/tutorials-website>
16. <https://survivalfreedom.com/what-is-the-standard-voltage-and-amps-for-an-outlet/#:~:text=In%20the%20United%20States%2C%20a,chargers%2C%20music%20systems%2C%20etc>. [16]