Electrical Engineering Department
California Polytechnic State University

Senior Project Report

Wildfire Early Detection System (WEDS)

June 7th, 2021

Mason McIver
Vincent Liang
JeanReno Racines

Advisor: Dr. Andrew Danowitz

Table of Contents

List of Tables and Figures

Acknowledgement

We would like to acknowledge Professor Andrew Danowitz and Professor Rich Murray for their assistance and support throughout the duration of this project. WEDS is dedicated to those who lost their lives and homes in the Camp Fire in Paradise California during November of 2018.

Abstract

With climate change causing an increase in temperature over the past several decades, wildfires have been burning hotter and moving quicker leaving a trail of destruction in their path. Detecting a wildfire early allows firefighters to respond efficiently and effectively to ensure containment. With the rise of advanced computer vision and algorithms, autonomous systems can be used to monitor and report any fire activity. Having multiple devices spread out across a large area will allow first responders to map out the fire location and track the fire. By utilizing smart technologies, property damage can be minimized and residents living in fire prone areas can be evacuated earlier.

The wildfire early detection system (WEDS) is a low-powered, low-cost (in both manufacturing and maintenance), easily deployable unit that can be mass-produced. The goal is to produce large volumes of this product to cover as much acreage in forests as possible. In a given area, multiple devices would report to one control center. Ideally, users would deploy these units in dense forests as this is where fires are harder to control and detect. Onboard sensors and cameras will detect heat signatures, and smoke particles to determine if a wildfire is present. To keep long-lasting, LiPo batteries supported by solar cells are required to power it. Keeping these devices small and portable will allow for wider distribution, increasing the area covered.

General Introduction and Background

Figure 1: Bidwell Bar Bridge surrounded by the Bear fire's flames in Lake Oroville, California [1].

Wildfires – also referred to as forest fires, bushfires, or unplanned fires – are uncontrolled fires that originate from wildland areas that can spread over landscapes [1]. *Figure 1* shows an example of such an event in Lake Oroville, California. Over the past several decades, there has been an increase in not only the number of fires the intensity of wildfires across the globe. Organizations and private companies are developing ways to detect wildfires in order to limit the damage that is done regardless of the source of the fire.

Uncontrolled fires are devastating to human life as they are capable of damaging whole communities, leaving thousands of residents evacuated with nowhere to go. For example, in 2018 the Camp Fire left the town of Paradise, California in ruins after the community was burned down in just four hours. The fire killed 85 people and damaged and destroyed over 18,000 structures and homes, making it the deadliest fire in California's history up to date [2]. Besides the trail of destruction wildfires leave, short and long-term human health is also jeopardized. As the fire burns, smoke and other fine airborne particles are released into the air and depending on wind patterns at the time, winds could carry the smoke particles hundreds of miles away in all directions. This can lead to a higher risk for disease such as ischemic heart disease or asthma in both adults and children [3].

There are numerous natural and human causes of wildfires. Natural causes include lightning, heatwaves, and dry weather conditions while human causes include faulty power lines, human negligence, and arson. However, over the past several decades, the frequency of wildfires has increased. With climate change occurring, there have been higher temperatures which increase the potential for a wildfire to spark. According to Running, the weather has been contributing to the wildfire intensity [4]. Due to warmer temperatures, snowpacks have been melting sooner, causing longer dryer seasons which leave more tinder and brush on the forest floor, which helps fires to burn longer [4]. Since the industrial revolution, global average temperatures have increased about 1.8°F (1℃) and since the 1980s, the rate at which temperature increase is now about 0.3°F (0.2℃) per decade [5]. Even though the increase seems small, it has a huge effect on humidity which can determine how dry brush and leaves are. Humidity is not the only factor that affects the strength of fire. Factors such as wind speed or direction, vegetation type or density, and topology can make fighting wildfires more difficult.

With the increase in their intensity over the past several decades, fires have been burning more uncontrollably and for longer periods, burning more acres of forestland.

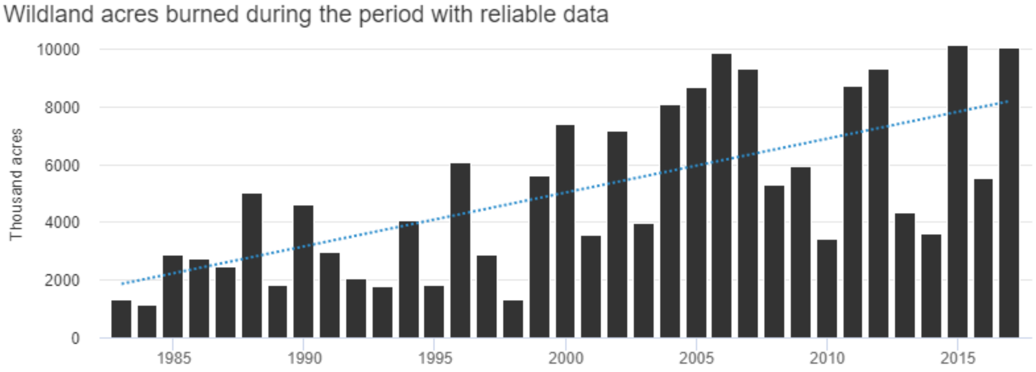


Figure 2: Wildland acres burned in the United States [6]

The US Forest Service and other firefighting agencies have been spreading wildfire awareness for decades to reduce human-caused fires. One way to limit their destruction is to detect them earlier and faster. Currently, the main methods to detect wildfires include fire lookouts, aerial patrols, ground patrols, infrared scanning, and advanced weather prediction systems [7]. Companies including Insight Robotics, Vigilys, and even a Cal Poly startup called Perch have developed technologies that aid in early wildfire detection [8,9,10]. Most of these early detection systems use a combination of infrared and smoke sensors that then alert the proper authorities when a fire is detected.

Product Description

The Wildfire Early Detection System (WEDS) is a device that monitors conditions in areas prone to wildfires. Each unit contains a smoke sensor and a camera to detect a fire. Based on the parts per million of smoke particles in the air and the images received by the camera, the Raspberry Pi computer can determine if a fire is nearby. Once a fire has been detected, a signal is sent to the main central hub where it notifies first responders of the fire's location. Because of the limited communication range, multiple devices would communicate with each other and would daisy chain back to a central hub as shown in *Figure 3*. Each unit is equipped with 10 solar panels, which totals to about thirty square inches of solar panels and an 8000 mAh LiPo battery. The unit would be mounted on a pole to provide wide viewpoints to cover as much area as possible. Ideal locations for these devices include densely forested areas that experience dry weather periods and/or the border of human communities.

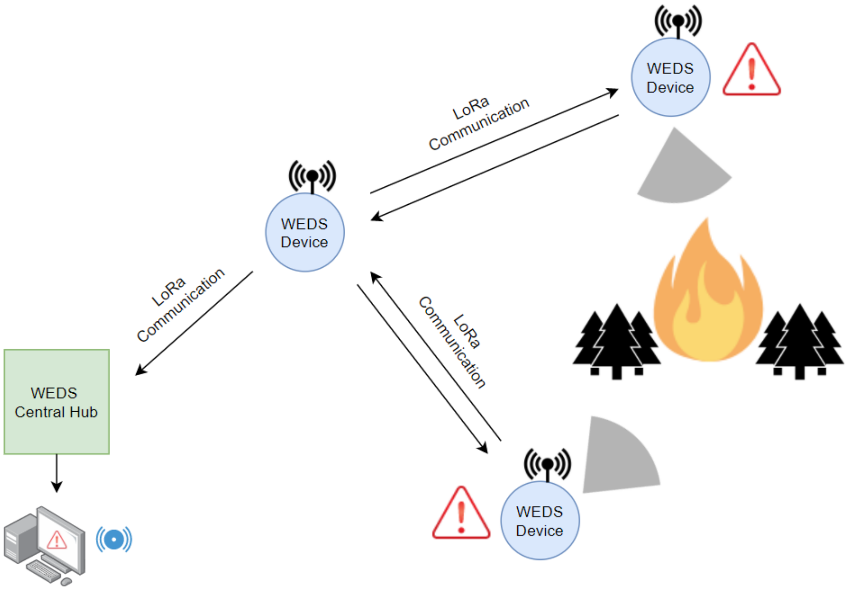


Figure 3: Illustration of WEDS system

Market and Product Research

Autonomous fire detection technology is still relatively new, and the market is quite small. There are a few companies that have developed similar systems. These other companies mainly use AI technology and computer vision to detect fires using thermal imaging and smoke patterns. Some other companies use the temperature and sag of transmission lines to determine if there is a fire. Most are very expensive and have not been widely adopted. With the increase in the number of fires in the western region of the United States, the need for better and more reliable fire detection devices are at an all-time high.

WEDS aims for mass production and distribution to cover as much forest area as possible while competitors place their devices in targeted areas. To achieve mass adoption, WEDS needs to be cheaper and easier to deploy than our competitors. Like competitors, wildfire systems need to be low maintenance and be able to function in almost all-weather conditions.

Customer Archetype

Currently, a few companies have designed and developed some wildfire detection systems. By selling a simple, versatile, and lower-powered device, users can purchase and deploy a large number of units throughout dense, fire-prone areas. To ensure this product's effectiveness, it is important to work with local forest agencies to find ideal locations. Other customers may include utility companies, residents, and fire protection agencies. Government agencies such as the U.S Forest Service, National Park Service, CAL Fire, and PG&E all stand to gain from early detection of wildfires. These agencies could work together to invest in WEDS to eventually provide statewide wildfire detection in California. While fire protection agencies such as Cal Fire are focused on fighting fires that watchtowers, aerial patrols, and public reporting have already detected, the adoption of WEDS will ensure earlier wildfires detection to help fire protection agencies save money and lives.

The agencies that are targeted by WEDS are the most involved in wildfire prevention throughout the United States and getting these agencies to invest in WEDS could allow for private residents to become potential customers. The U.S Forest Service as well as the National Park Service would benefit by investing in WEDS because the current methods for detecting wildfires do not always prevent mass destruction. WEDS will provide a means for these agencies to detect wildfires while they are still in the containable surface fire stage and can reduce the costs of fire suppression [11].

Residents and landowner who live in fire-prone wildlands could also use WEDS to provide an early warning for potentially devastating wildfires. This is especially useful for when these people spend time away from their homes.

Type of Customer	**Description**	**Reason**	**Product Use**
US Forest Service: Pacific Northwest and Pacific Southwest Regions	Those who are involved with maintaining heavily forested areas [12].	WEDS would aid in minimizing their response time to fires that threaten areas maintained by the US Forest Service.	They will use this to minimize damages to government-owned land and property.
Private Landowners in Pacific Northwest and Southwest Forest regions	These customers have a preference for living near forested areas. They may either own properties or land in fire prone areas.	They may want to use WEDS to monitor their surroundings for fire danger.	They will use this device for property protection and early detection which will give homeowners time to evacuate.

Table 1: Customer Archetype

Along with weather conditions, one of the biggest reasons that wildfires cause so much destruction is the delayed response time to these incidents. Firefighters are not constantly monitoring fire-prone areas and wildfires are often discovered via public reporting, a very slow method of detecting wildfires [11]. When fires are finally detected, it may begin spreading from treetop to treetop, making it nearly uncontrollable and able do extreme damage to human-made structures [11]. Alerting firefighters about wildfires as fast as possible is what WEDS overcomes. WEDS monitors critical areas 24/7 so that humans do not have to. It can eliminate much of the costs involved in fighting fires since it detects them quicker. Additionally, the early wildfire detection technology can save lives as residents are alerted earlier on allowing for adequate evacuation time.

Market Description

It would take considerable effort to penetrate the market and stand out against our competitors. A wildfire detection device is not a new idea, so WEDS would have to present something that none of its competitors can do. The device could initially enter the market as an economical alternative to expensive sensors that still gets the job done: preventing fires from their onset or at least before they become difficult to contain.

Key partners include plastics, electronics, and PCB manufacturing companies. Working closely with these partners would guarantee the on-time production of prototypes as well as the steady production of the final design that goes out to the consumer. Other important key partners include government organizations like the National Park Service whose implementation of this product would be key in proving its effectiveness. This is important because our customer archetype could grow to include not only government agencies but also landowners that need solutions for protecting their land from fire.

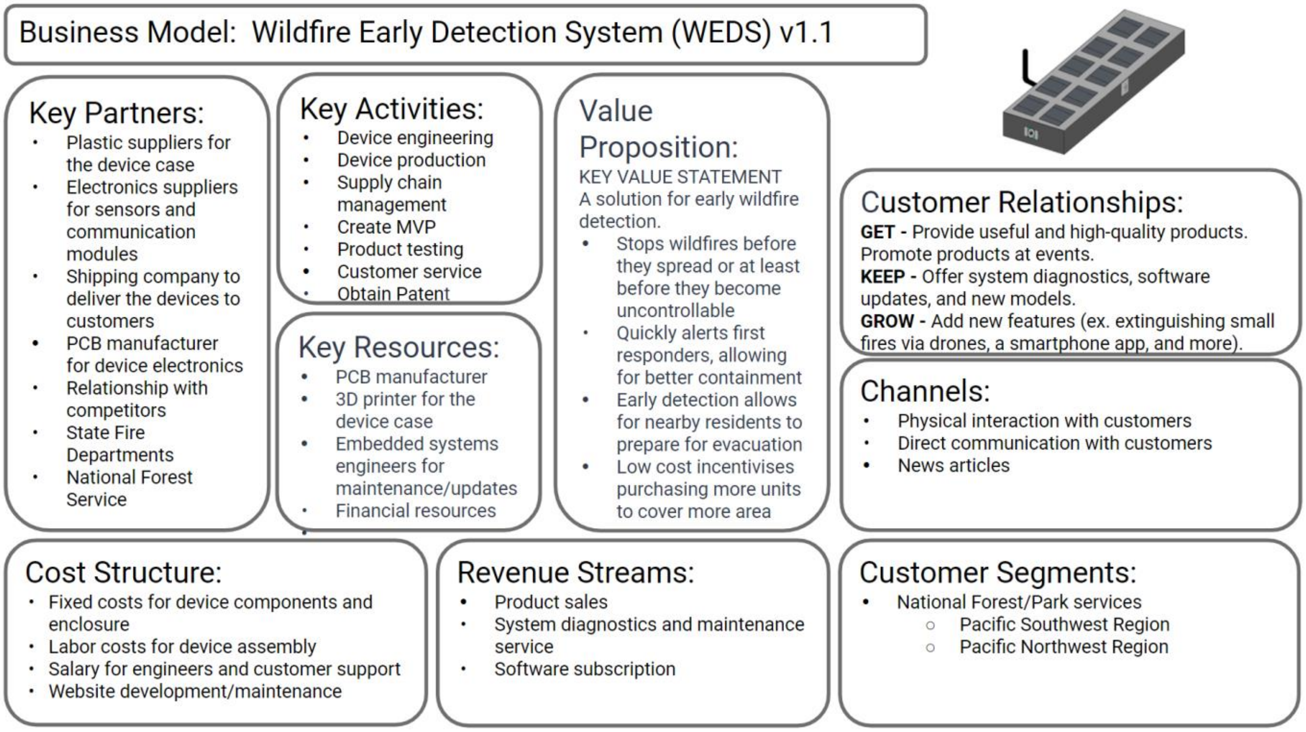
According to Alkhatib, the current methods of wildfire detection include watchtowers, weather forecasts, spotter planes, and reports by the public [11]. Even though these methods work, a fire in a very desolate area could burn rapidly and spread before someone detects it. Automated systems would help decrease response times and allow firefighters to respond quicker. WEDS solves this problem by alerting the proper authorities early on and relaying vital information that would help first responders decide what the best option is to contain the fire.

Currently, the limitations of the WEDS device is the range at which it could communicate with the WEDS central hub and the power it can store. Because the device runs off of solar power, if some clouds or obstructions block the solar cells, the device will eventually run out of power. The forest area is extremely vast and with limited communication range, there will need to be multiple WEDS central hubs to cover fire-prone areas.

Advantages	**Disadvantages**
Low cost to produce and maintain	Does not have global market penetration like other competitors
Versatile mounting hardware	Requires multiple WEDS central hubs to cover all fire-prone areas.
Relays vital weather information	Limited battery capabilities
Provides the customer with wildfire detection and monitoring to minimize damage caused by wildfires.	Limited data transfer and communications capabilities
Small form minimizes wildlife disruption	Will only be available on the west coast at product launch

Table 2: Advantages and Disadvantages of WEDS

Business Model Canvas Graphic



Business Model: Wildfire Early Detection System (WEDS) v1.1

Key Partners:
- Plastic suppliers for the device case
- Electronics suppliers for sensors and communication modules
- Shipping company to deliver the devices to customers
- PCB manufacturer for device electronics
- Relationship with competitors
- State Fire Departments
- National Forest Service

Key Activities:
- Device engineering
- Device production
- Supply chain management
- Create MVP
- Product testing
- Customer service
- Obtain Patent

Key Resources:
- PCB manufacturer
- 3D printer for the device case
- Embedded systems engineers for maintenance/updates
- Financial resources

Value Proposition:

KEY VALUE STATEMENT

A solution for early wildfire detection.
- Stops wildfires before they spread or at least before they become uncontrollable
- Quickly alerts first responders, allowing for better containment
- Early detection allows for nearby residents to prepare for evacuation
- Low cost incentivises purchasing more units to cover more area

Customer Relationships:

GET - Provide useful and high-quality products. Promote products at events.

KEEP - Offer system diagnostics, software updates, and new models.

GROW - Add new features (ex. extinguishing small fires via drones, a smartphone app, and more).

Channels:
- Physical interaction with customers
- Direct communication with customers
- News articles

Cost Structure:
- Fixed costs for device components and enclosure
- Labor costs for device assembly
- Salary for engineers and customer support
- Website development/maintenance

Revenue Streams:
- Product sales
- System diagnostics and maintenance service
- Software subscription

Customer Segments:
- National Forest/Park services
    - Pacific Southwest Region
    - Pacific Northwest Region

Figure 4: Business Model Canvas

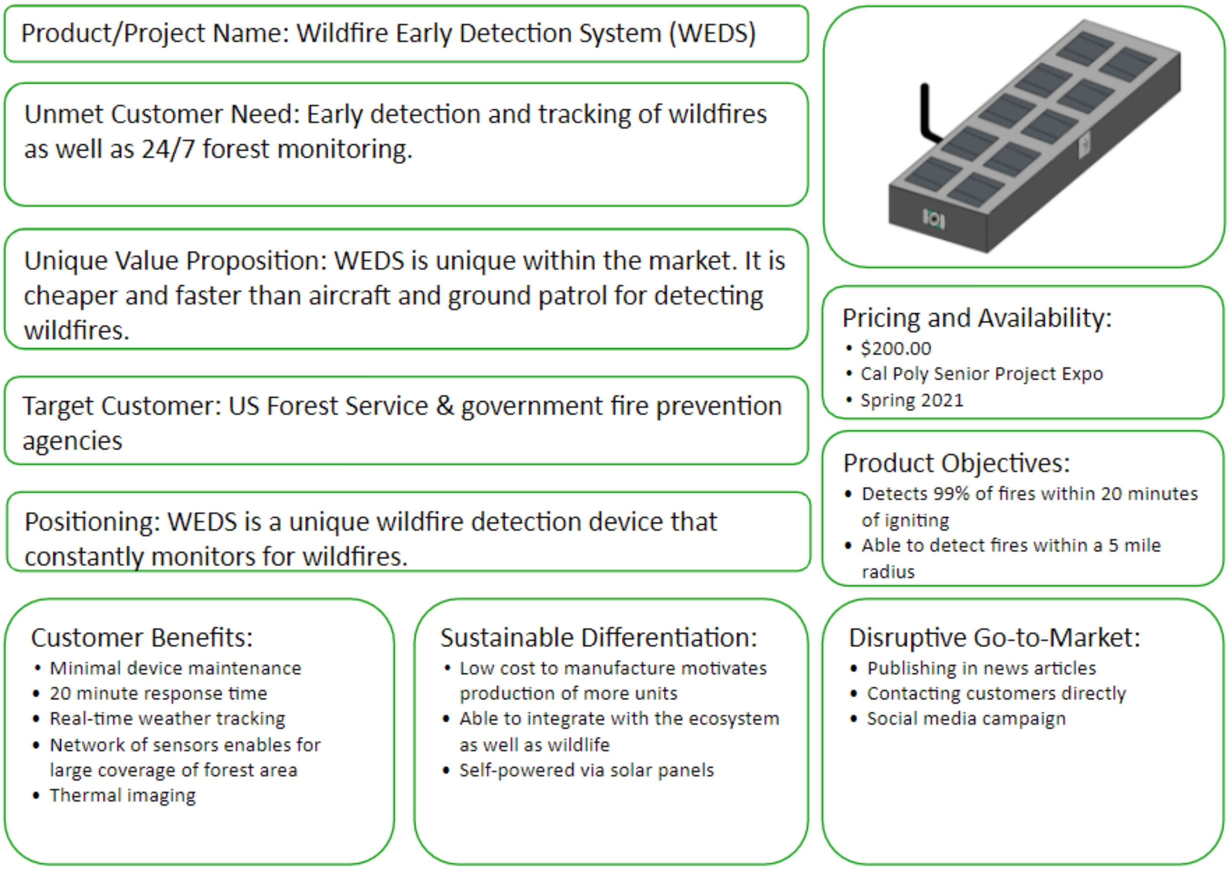
Marketing Requirements

Automatic wildfire detection systems must be able to alert the proper authorities when a wildfire emerges. This means the device must have multiple reliable sensors that can detect heat, smoke, and other weather conditions to allow for the system to accurately pinpoint the wildfire location. An automated system can monitor forests without humans, so the market demands that these devices are low maintenance and self-powered. Along with that, devices must have a reliable communication network to send data and alert first responders.

Marketing Requirements	**Reason**
Wireless Technology	Data can be passed device-to-device to cover larger areas than a single unit can.
Camera	Used with computer vision to detect and confirm the location of the fire.
Solar Charging	The device can recharge the batteries through solar cells to limit the maintenance of these devices.
Long Lasting Battery	The device is self-powered for an extended period of time. If there is no sunlight for the solar cells to recharge the battery, the battery must be able to power all components for an extended period of time until the sunlight returns.
Environmentally Friendly	If one of the WEDS units breaks down or falls from its post, it must not cause harm to the surrounding ecosystems. This means that the device is designed such that animals cannot easily get inside and potentially consume any of the electronics.
Weather Resistant	For this device to sustain long periods of use, it must be adequately weather-sealed and robust The enclosure protecting the electronics is resistant to high temperatures in case the fire gets close to the unit.

Table 3: Marketing Requirements

Marketing Data Sheet



Product/Project Name: Wildfire Early Detection System (WEDS)

Unmet Customer Need: Early detection and tracking of wildfires as well as 24/7 forest monitoring.

Unique Value Proposition: WEDS is unique within the market. It is cheaper and faster than aircraft and ground patrol for detecting wildfires.

Target Customer: US Forest Service & government fire prevention agencies

Positioning: WEDS is a unique wildfire detection device that constantly monitors for wildfires.

Pricing and Availability:
- $200.00
- Cal Poly Senior Project Expo
- Spring 2021

Product Objectives:
- Detects 99% of fires within 20 minutes of igniting
- Able to detect fires within a 5 mile radius

Customer Benefits:
- Minimal device maintenance
- 20 minute response time
- Real-time weather tracking
- Network of sensors enables for large coverage of forest area
- Thermal imaging

Sustainable Differentiation:
- Low cost to manufacture motivates production of more units
- Able to integrate with the ecosystem as well as wildlife
- Self-powered via solar panels

Disruptive Go-to-Market:
- Publishing in news articles
- Contacting customers directly
- Social media campaign

Figure 5: Marketing Datasheet

Engineering Requirements

Engineering Requirements	**Implementation**
Smoke sensor	Use of gas sensor to detect parts per million (PPM) of smoke particles
Camera that can capture photos of the surrounding area	Use of a 1080p camera to provide general image and detection
Wireless Communication	Used to communicate with other WEDS devices. Implement with software to transmit data at fixed intervals and allow for devices to be daisy chained.
Operates from 5V source	Uses 8000mAh LiPo battery with PiJuice uninterruptible power supply. A 5 V source is required to power the Raspberry Pi.
Charging batteries using solar panel	Use of 10, 5V 2.5W solar panels with a clear line of sight with the sky.
Computer that can run off of a 5V supply and run computer vision algorithms	Use of a Raspberry Pi Zero that can control multiple modules at a time.
Low profile enclosure to minimize wildlife disruption	Use of an aluminum lid with lock to prevent tampering. Robust wooden enclosure to house all components

Table 4: Engineering Requirements

Design

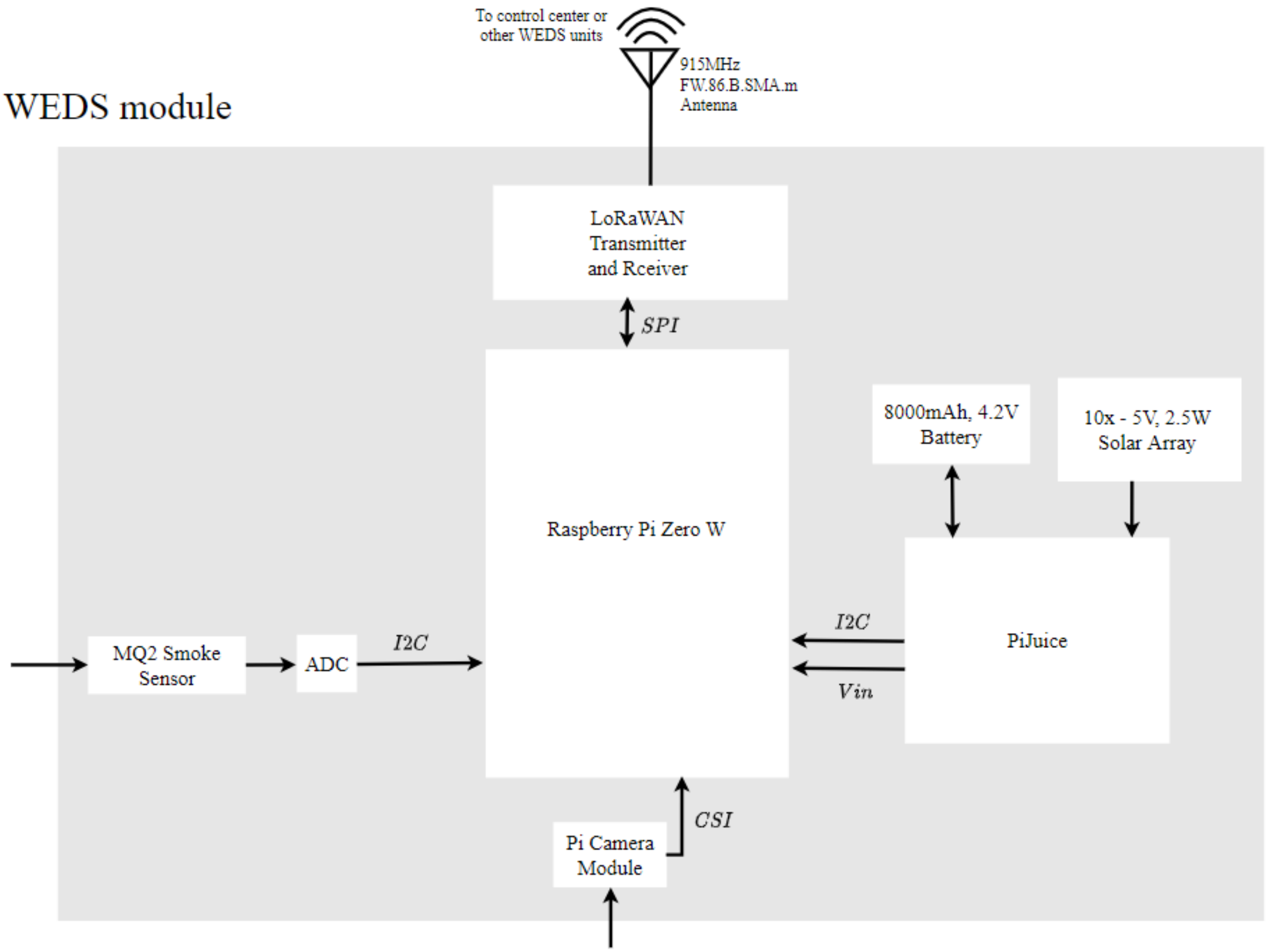


Figure 6: Level 1 Block Diagram

Raspberry Pi:

The main method of detecting wildfires is using a camera paired with computer vision algorithms to recognize fires. To achieve this, a powerful enough computer is needed to compute and run the computer vision code. The system also needs to be low powered since this device will be placed in remote areas for long periods of time. The cheapest, most effective computer on the market is the Raspberry Pi Zero W.

The Raspberry Pi Zero is a single board computer with a 1GHz single-core ARMv6 CPU with 512MB of RAM. The Raspberry Pi Zero consumes 150mA, however can consume as low as 120mA if the HDMI and LEDs are turned off [13]. The Raspberry Pi that is used to control the WEDS module runs on the Raspberry Pi OS which can be downloaded onto an SD card from raspberrypi.org/software/.

We wrote our computer vision algorithms in Python for its simplicity. We used the OpenCV library and trained our models on an external machine with a much faster processor.

Accessing Raspberry Pi via ssh over Bluetooth

The WEDS central Hub will be powered by a Raspberry Pi that is connected to a LoRaWAN radio, mouse, keyboard, and a monitor. This will allow a technician to view and interact with wildfire alerts and any software issues. When the WEDS module is placed in a desolate location that is prone to wildfires, it is not possible to control the Raspberry Pi with a monitor, keyboard and mouse so it will be controlled with a laptop over SSH (secure shell protocol). There is often no internet connection in the locations where the WEDS modules are placed so Bluetooth will be used to SSH into the raspberry pi. To SSH into the Raspberry Pi through Bluetooth the blueman program needs to be downloaded onto the Raspberry Pi which can be done by typing the following command:

```
sudo apt install network-manager-gnome blueman
sudo reboot
```

Battery and Solar:

The WEDS module is to be self-sustained, meaning that when the WEDS module is mounted in a wildfire prone location, the solar panels will charge the battery faster than the battery is discharged.

Preliminary Design: Using 18650 Batteries

The original plan for keeping the WEDS module powered as a self-sustaining device was to use 3.7V nominal, 2600mAh 18650 batteries. This plan ultimately resulted in a dead end and an alternative power solution was used in the final design. The plan to use 18650 batteries involved connecting a solar panel and a single 18650 battery to a TC4056 charge controller. The output of the TC4056 charge controller would then be connected to a boost converter in order to create the 5V needed for the Raspberry Pi. The TC4056 charge controller which is meant for charging a single 18650 battery would work to power the Raspberry Pi for only a few hours. This means that a single battery is not sufficient enough to power the WEDS module if it was to be self-sustained (via solar). Two solutions were proposed to add more 18650 batteries: one was to cycle through an array of single 18650 batteries and another was to create a battery bank made up of multiple 18650 batteries placed in parallel.

An attempt to cycle through an array of single 18650 batteries was explored however there were several problems that were encountered. The first problem with cycling through batteries was that a single 18650 battery would not consistently boot up the Raspberry Pi. One battery with a charge capacity of 2600mAh was not able to provide enough current to the Raspberry Pi upon bootup. Another problem was that the switching system needed to have a power supply. The plan was for the switching algorithm to be controlled by Raspberry Pi which means that the switching system would be switching its own power supply which could cause problems that would be very hard to detect via simulation results.

The next problem was finding switches that work for this application. If a MOSFET was used as the switch, a high side driver would be needed because the switch will be switching between two different batteries which means that there will be a “high” voltage connected to both the source and the drain bringing the MOSFET to the saturation region, preventing the MOSFET from acting like a switch. This is because the positive terminal of one battery will be connected to the source and the positive terminal of another battery will be connected to the drain. This will add extra cost and unnecessary complexity to the design. Using a relay was also proposed but a significant current draw would be necessary to keep the relay coil energized to keep the switch on. The final reason that the switching system was not viable was that there was little to no documentation on a similar design. Nearly all solar powered systems used a single battery bank instead of switching between single batteries.

The method of using a battery bank made up of 18650 batteries was not explored as in depth as the first method because of the increased cost of this method. This would require purchasing an expensive charge controller, battery management system, and a buck converter without a good way to test if this is even a viable method for powering the WEDS module. The next step was to look for a power solution that was built for powering the Raspberry Pi.

Final Design: PiJuice HAT

After the plan to use 18650 batteries resulted in a dead end, the decision was made to use a PiJuice HAT module to monitor and regulate the battery charging and discharging. The PiJuice HAT is an uninterruptible supply module attached on top of the Raspberry Pi [14]. The PiJuice only uses five pins on the Raspberry Pi (5V, 3v3, GND, SCL, SDA) but provides access to all 40 pins through the HAT [14].

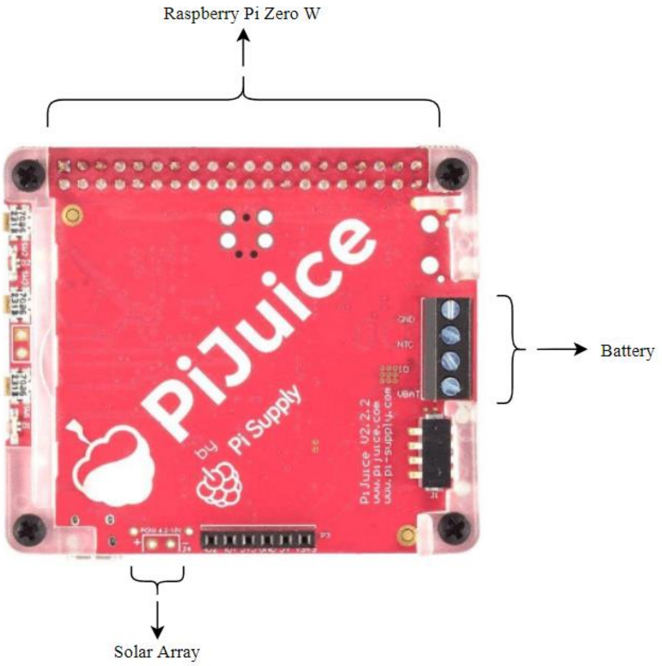


Figure 7: PiJuice Uninterruptible Supply Module

The PiJuice HAT also contains a power management board that is meant specifically to be used as a power supply for the Raspberry Pi. The PiJuice can be used with lithium-ion batteries however the PiJuice can only support the connection of a single cell battery. This limits the amount of energy that can be stored for powering the WEDS module, which will limit the amount of time that the WEDS module can be operational before needing to be recharged. Although it is possible that the battery that is powering the WEDS module will run out of charge during a cloudy season, the PiJuice power supply is more reliable and safer than using a battery pack made up of 18650 cells.

The next step is to size the battery that will be used to power the WEDS module and calculate how long the WEDS module will last. This is a difficult task because of the uncertainty in how much power the Raspberry Pi will require in order to operate the WEDS module. Raspberry Pi provides the typical bare board current consumption, as well as the typical current draw for some peripherals. The bare board current draw for the Raspberry Pi Zero W is 150 mA, the camera module will draw 250 mA, and the combination of GPIO pins used will draw a max of 50 mA [15]. The documentation on the current draw of each component is inconsistent and there is a large variability in current draw depending on what the device is doing. The current draw of the LoRaWAN radio, MQ-2 smoke sensor, and the ADS 1115 Analog to Digital converter can be found in *Table 5* but 50 mA will be used as an estimate for the average overall current draw of these components. This means that the WEDS module will draw approximately 450 mA total. The power the WEDS module requires can be calculated as follows:

$$P = VI = (5V)(450mA) = 2.25W \quad \text{(a)}$$

Device	LoRaWAN Radio		MQ-2 Smoke Sensor		ADS 1115 ADC
Current Draw	Transmission	Listening	Maximum	Minimum	Continuous Mode
	100 mA [16]	30 mA [16]	160 mA [17]	0.1 mA [17]	150µA [18]

Table 5: Component Current Draw Approximations

The PiJuice comes with an 1820mAh LiPo (Lithium Polymer) battery but because the WEDS system is required to be constantly operational a larger capacity battery is needed. PiSupply, the supplier of the PiJuice module, makes a 12,000 mAh battery that is specifically made to be used with the PiJuice [19]. As stated previously, the WEDS module will draw about 450mA [15] and the energy that is stored in a $4.2V_{max,}$ 12,000 mAh battery can be calculated as follows:

$$E = QV = (12{,}000\ mAh)(4.2\ V) = 50.4\ Wh \quad \text{(b)}$$

Knowing that the amount of power dissipated by the WEDS module is 2.25 was calculated in equation (a). The amount of time that each battery will last after it is fully charged can be calculated as follows:

$$t_{discharge} = \frac{Energy}{Power} = \frac{50.4\ Wh}{2.25\ W} = 22.4\ hrs \quad \text{(c)}$$

The solar panel array was chosen to provide enough current to charge the battery as well as power the WEDS module even at half of the maximum current. The solar array is made up of 10, 5V, 2.5W solar panels. This means that at maximum power production, each solar panel will provide 500 mA of current which is shown in equation (d). When 10 of these solar panels are connected in parallel, the maximum amount of current produced will be 10 times 500mA resulting in a maximum current production of 5A.

$$P = IV \rightarrow I = \frac{P}{V} = \frac{2.5\,W}{5\,V} = 500\,mA \tag{d}$$

With a maximum current production of 5A, this means that even when the solar array is producing half of this current, there will be enough current to power the WEDS module (450mA) and to charge the battery at 1.5A. While the PiJuice can charge the battery at a maximum of 2.5A, the charging current is assumed to be 1.5A which should take about 8 hours to charge. This is shown in equation (e) below.

$$Charging\ time = \frac{Battery\ Capacity}{Charging\ Current} = \frac{12000mAh}{1.5A} = 8\ hrs \tag{e}$$

The important figures regarding charging the LiPo battery and powering the WEDS module via the solar array are tabulated in *Table 6* below. All values are calculated. Test data is shown in *Table 12*.

*Note that while the final WEDS product would ideally be powered by the PIS-1129, a 12,000 mAh battery that is available from PiSupply, this battery is only available from an overseas supplier. LiPo batteries cannot be shipped by air and therefore would not arrive in time for testing of this prototype. An 8000 mAh LiPo battery is used in place of the PIS-1129, therefore the power calculations are done using 12000 mAh and 8000 mAh in order to be easily compared to the test data. Both batteries have a nominal voltage of 4.2 V.

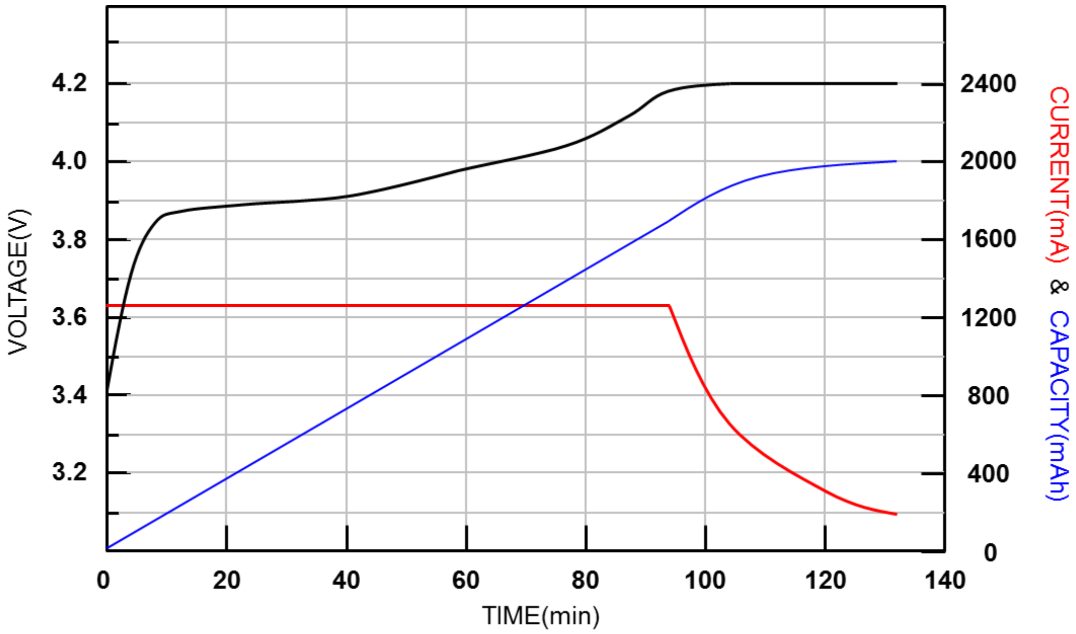


Figure 8: shows the charging curve for a 4.2 V, 2000 mAh LiPo battery [20]

The three curves in *Figure 8* show the current, voltage, and charge capacity of a LiPo battery that is being charged. The charge capacity of the battery increases somewhat linearly with time as the battery is charged. The voltage steadily increases non-linearly throughout the charging process and the current is constant until the very end of the process where it decreases exponentially. For estimation purposes, it can be assumed that each hour that the battery is charged at 1.5 A will correspond to a constant amount of discharge time.

$$t_{runtime\ per\ charge} = \frac{22.4\ hrs}{8\ hrs} = 2.8\ hrs/hr \qquad \text{(f)}$$

This means that for every hour that the battery is charged at 1.5 A, the battery can then power the WEDS module for 2.8 hours. In order for continuous operation the WEDS module must be able to operate for 24 hours off of the energy received from the sun in a single day. Each hour of sunlight results in 1 hour of operation as well as 2.8 hour of “future operation” that is stored as energy in the battery. The amount of necessary sun per day is calculated in equation (g) below.

$$t_{sun} = \frac{24hrs}{1\ hr + 2.75\ hrs} = 6.4\ hrs\ of\ sunlight \qquad \text{(g)}$$

	8000 mAh	12000 mAh
Power Consumption	2.295 W	2.295 W
Energy in Battery	33.6 Wh	50.4 Wh
Battery Discharge Time	14.6 hrs	22.0 hrs
Battery Charge Time (0.5 A)	16.0 hrs	24.0 hrs
Battery Charge Time (1.5 A)	5.3 hrs	8.0 hrs
Battery Charge Time (2.5 A)	3.2 hrs	4.8 hrs

	8000 mAh	12000 mAh
Maximum Solar Array Current	5 A	5 A
Hours of Sun Needed (0.5 A)	12.5 hrs	12.5 hrs
Hours of Sun Needed (1.5 A)	6.4 hrs	6.4 hrs
Hours of Sun Needed (2.5 A)	4.3 hrs	4.3 hrs

Table 6: Theoretical Values: 8000mAh vs 12000mAh Battery

(Compare to Measured values in Table 12)

The hours of sun needed for the 12000mAh battery and the 8000mAh battery are the same because the limiting factor in keeping the WEDS module self-sustaining is how fast the solar panels can charge the battery. The capacity of the battery is not as important as long as anytime there is sun, the battery is charging. So, although observing the theoretical calculations may cause one to think that there is no difference between using the two batteries it is important to take into account that the amount of sunlight that will reach the solar panels in a day is very inconsistent. There will be some days where the sun shines very brightly on the solar panels for an extended period of time and a larger capacity battery will allow the system to take advantage of this extra energy to be used on days that may be cloudier. Note from *Table* 6 that if the battery is charging at 0.5 A, this is not fast enough for the WEDS module to be self-sustaining because there is not 12.5 hrs of sunlight available per day.

Integrating PiJuice and Raspberry Pi

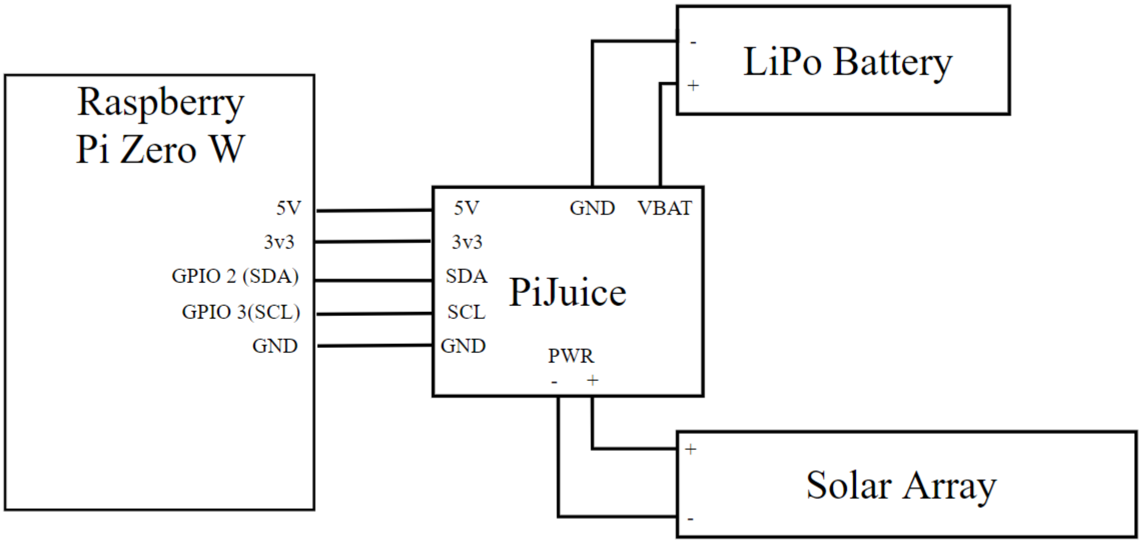


Figure 9: PiJuice Setup Block Diagram

The PiJuice has the ability to go into a deep-sleep state that will allow the Raspberry Pi to save power. This can be set to a calendar so the Raspberry Pi will wake up. The software used to control the Pijuice can easily be installed on the Raspberry Pi by entering the following command:

```
sudo apt-get install pijuice-gui
```

The WEDS module will have 4 different modes of operation. These four modes will be distinguished by the wait time between checking for evidence of a wildfire. Mode 1 will constantly be checking for wildfires, Mode 2 will check every 5 minutes, Mode 3 will check every 10 minutes, and Mode 4 will check every 15 minutes. For modes 2 through 4 the Raspberry Pi will enter a deep-sleep state between checking for evidence of a wildfire. The goal of WEDS is to detect wildfires as early as possible so intermittent checking may help improve the battery life of the module but even five minutes can make a huge difference in how easily the fire will be extinguished. For example, the Camp Fire that destroyed Paradise CA ignited around 6:20 am [21] and by 6:30am Cal Fire was already being flooded with calls reporting a wildfire [22]. At its peak the Camp Fire was spreading as fast as 80 football fields per minute [22].

It is recommended that each WEDS module is operated in Mode 1 in order to constantly check for evidence of a wildfire in order to be most effective. The addition of modes 2-4 are implemented in order to save battery life during long periods of cloud cover. It has been determined that 15 minutes is the longest period of time between checking for wildfire evidence where the WEDS module would still be effective.

In the rare case that the WEDS module battery is completely drained it will be necessary for a technician to recharge the module via micro-USB. When the PiJuice HAT senses that the battery is completely discharged it will signal the system to send a low power message to the central hub via LoRaWAN and then safely power off.

Communications:

Since the WEDS device will be placed in the middle of the forest, the communication protocol needs to be reliable, low power, and have a long range. Because of these requirements, the best communications protocol would be to use LoRaWAN (Low Power, Wide Area Network).

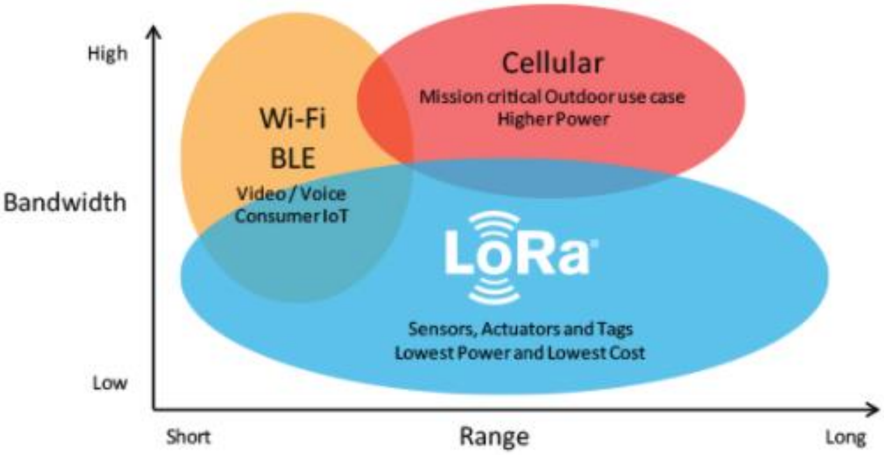


Figure 10: Communication Range Chart

LoRaWAN modules use approximately 100mA peak when transmitting and their range varies from 1.2 to 12 miles depending on the different obstructions and type of antenna used [16]. The module is capable of transmitting and receiving at 868 MHz and 915MHz frequency to communicate with other modules; however, the standard frequency band in the United States is 915 MHz which is used in this project. The LoRaWAN module will be used to send air quality data if the parts per million of smoke particles in the air crosses over the safe threshold, or if there is a wildfire detected on the camera via the computer vision software. There is not a large quantity of data that needs to be sent, but it needs to be sent reliably over long distances.

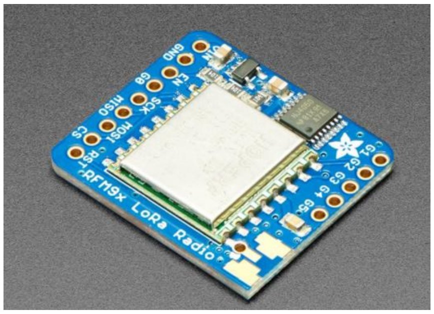


Figure 11: LoRaWAN Radio Module

All of the WEDS modules will use LoRaWAN to communicate with the central hub via other WEDS modules. Each LoRaWAN device can send and receive data and therefore if a WEDS module is out of the central hub's range and it detects signs of a wildfire, the warning will be sent through WEDS devices that are within range of the device that has detected the wildfire. This means that each WEDS device must be within range of another WEDS device, or within the range of the central hub to communicate in a daisy-chain manner. The daisy-chain communication of WEDS devices to the central hub is shown in *Figure 12*. The range of a LoRaWAN module depends heavily on the obstructions that lay between two devices that are attempting to communicate but 1.2 miles is used as the base range for the illustration.

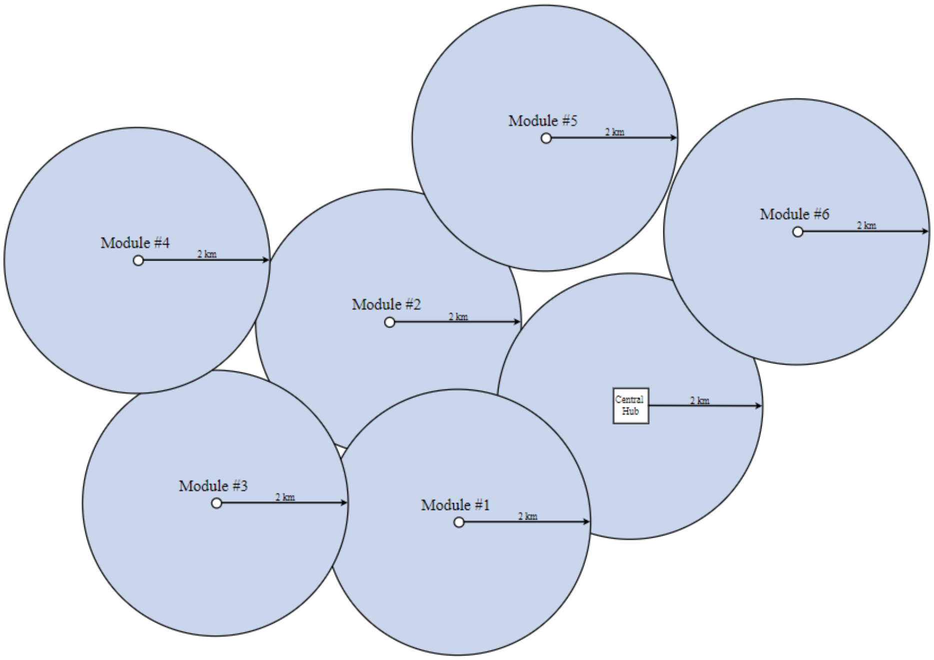


Figure 12: Daisy Chain communication of WEDS devices via LoRaWAN

The LoRaWAN module communicates to the Raspberry Pi via serial peripheral interface (SPI). This means that SPI must be enabled on the Raspberry Pi. Before interfacing the LoRaWAN with the Raspberry Pi, the following updates were installed:

```
sudo apt-get update
sudo apt-get upgrade
sudo apt-get install python3-pip
sudo pip3 install --upgrade setup tools
```

SPI was enables by using the following command:

```
sudo raspi-config
```

After the configuration utility is opened, interfacing options must be selected and then SPI can be enabled. The library for the RFM9x LoRaWAN module must be installed by entering the following command:

```
sudo pip3 install adafruit-circuitpython-rfm9x
```

The file called `font5x8.bin` must also be downloaded. This file can be found in the Github repository (See Appendix D).

The LoRaWAN module must then be connected to the Raspberry Pi according to the connections that are shown in *Table 7*. The python file for sending and receiving data with the LoRaWAN radio can be found in the Github repository (See Appendix D).

Raspberry Pi Pinout	LoRaWAN Module Pinout
3v3	Vin
GND	GND
No Connection	EN
GPIO 5	G0
GPIO 25	RST
GPIO 23 (SCK)	CLK
GPIO 9 (MISO)	MISO
GPIO 10 (MOSI)	MOSI
GPIO (CE1)	CS

Table 7: LoRaWAN module pin connections to Raspberry Pi

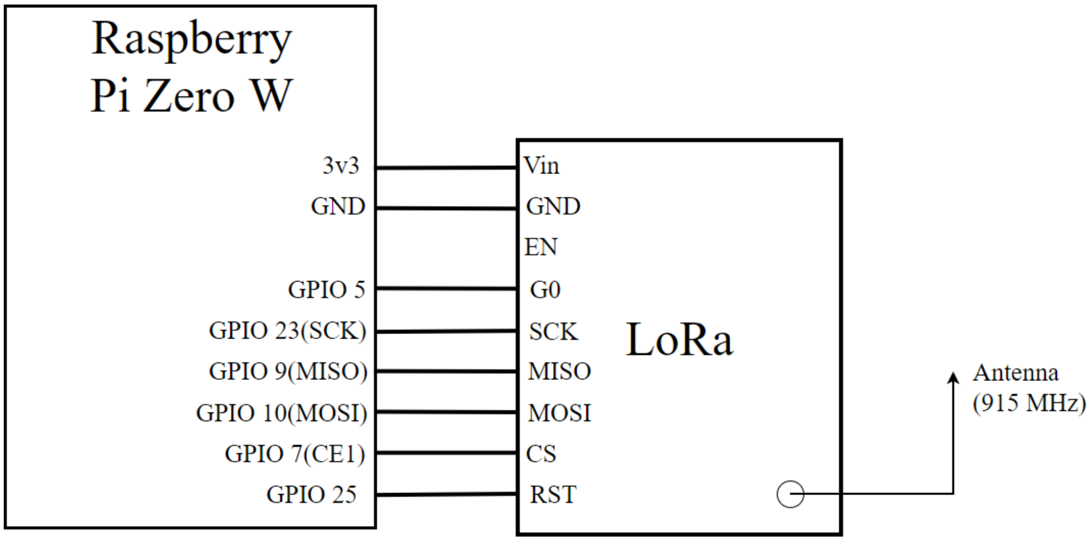


Figure 13: LoRaWAN setup with Raspberry Pi

Smoke Sensor:

A MQ-2 gas sensor will be used to help determine if there is a fire in the nearby area by measuring the parts per million (PPM) of smoke particles in the air. The MQ-2 is an analog sensor that can measure different gases in the air depending on the parts per million of that particular gas as shown in the graph in *Figure 14*. The analog output of the smoke sensor is a voltage between 0 V and 5V. This value must first be converted to the resistance ratio R_S/R_0 for parts per million to be read from the smoke sensor. An equation for the line labeled smoke in *Figure 14* can be found that will enable the WEDS module to calculate the parts per million of smoke particles based on the R_s/R_0 ratio. The graph is approximately linear for ppm values between 200 and 3000, the graph is also approximately linear for ppm values between 3000 and 10000 therefore two different equations were found. The equation used depends on the region that the R_s/R_0 ratio is in.

<u>200 ppm <x< 3000 ppm</u>

$$log(y) = mlog(x) + log(k) \; where \; m = \frac{\Delta log(x)}{\Delta log(y)}$$

Use the following points from *Figure 14*: $(x_1, y_1) = (800, 2) \; and \; (x_2, y_2) = (5000,1)$

$$m = \frac{\Delta log(x)}{\Delta log(y)} = \frac{log(5000) - log(800)}{log(1) - log(2)} = -0.3$$

Plug in (x_2, y_2)into $log(y) = mlog(x) + log(k)$ to find k

$$log(k) = log(1) + (0.3)log(5000) \rightarrow k = 1.11$$

$$y = 1.11x^{-0.3} \tag{h}$$

<u>3000 ppm <x<10000 ppm</u>

$$log(y) = mlog(x) + log(k)\ where\ m = \frac{\Delta log(x)}{\Delta log(y)}$$

Use the following points from *Figure 14*:

$$(x_1, y_1) = (5000, 0.9)\ and\ (x_2, y_2) = (100000, 0.6)$$

$$m = \frac{\Delta log(x)}{\Delta log(y)} = \frac{log(10000) - log(5000)}{log(0.6) - log(0.9)} = -1.71$$

Plug in (x_2, y_2)into $log(y) = mlog(x) + log(k)$ to find *k*

$$log(k) = log(1) + (1.71)log(10000) \rightarrow k = 6.84$$

$$y = 6.84x^{-1.71} \tag{i}$$

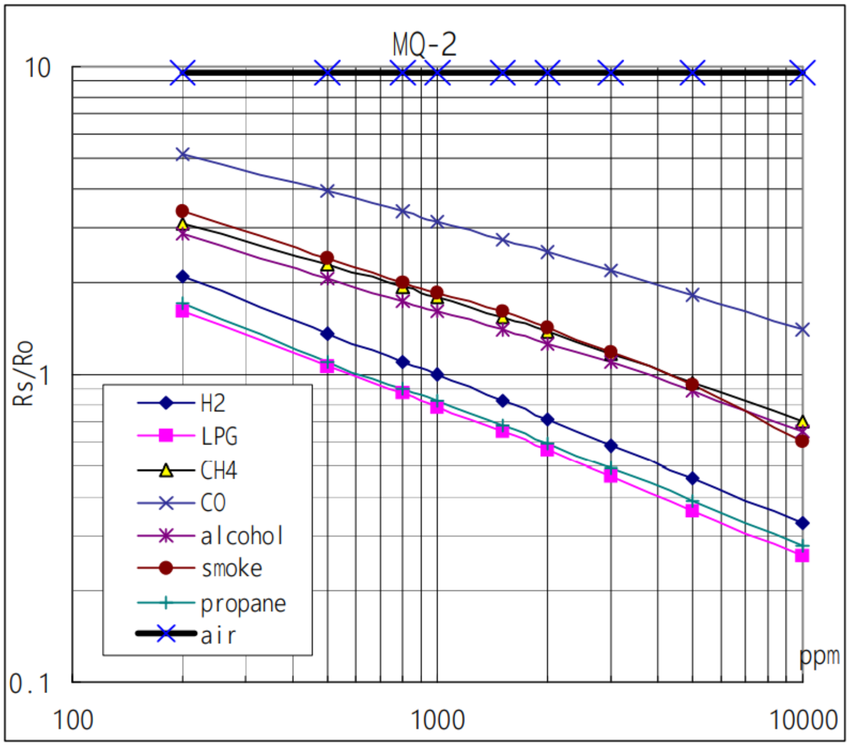


Figure 14: The ppm of various particles based on the ratio of Rs/Ro [23]

The MQ-2 smoke sensor relies on a tin-dioxide sensing element to detect the presence of smoke and flammable gases in the air. The sensing element also consists of an aluminum oxide ceramic and a heating coil that will heat the sensing element to a working temperature. The structure of the sensing element is shown in *Figure 15* below.

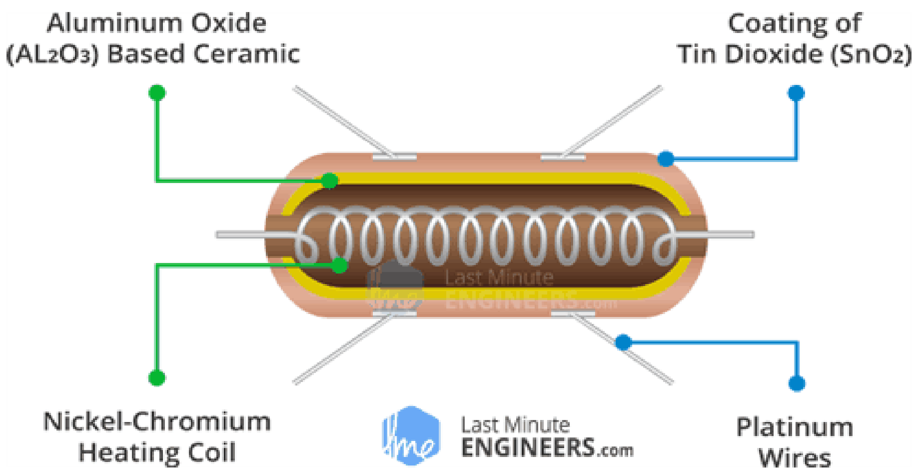


Figure 15: Heating and Sensing System [24]

The resistance of the sensing element will change depending on the concentration of smoke particles in the air. The value of R_0 is determined by the resistance of the sensing element in 1000ppm hydrogen. Note that 1000 ppm hydrogen is not clean air, from the graph in *Figure 14* it can be observed that the ratio R_s/R_0 will be 10 in clean air. The value of R_0 must be measured experimentally for each MQ-2 sensor.

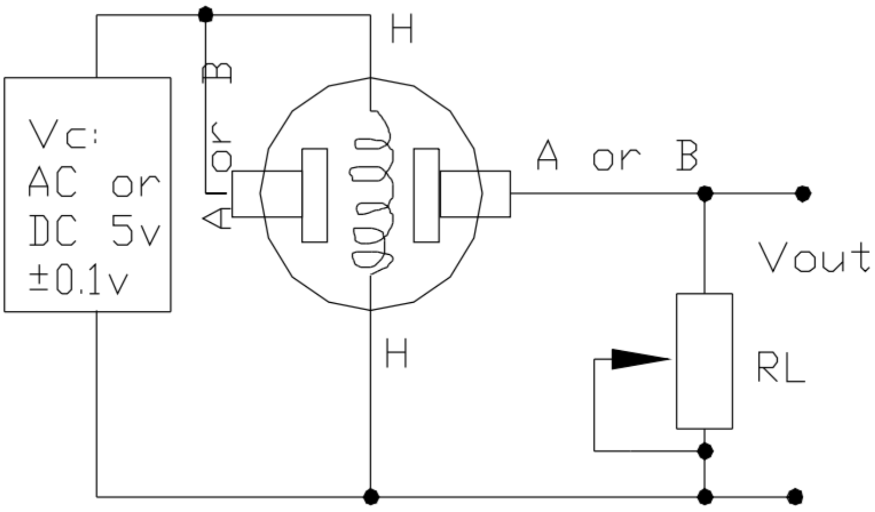


Figure 16: Schematic for using the MQ-2 Sensor [24]

The resistance of the sensing element can be calculated by using a voltage divider as long as both the supply voltage and the output voltage are known. The value of R_L on the breakout board is measured to be R_L = 18.1 kΩ. This means that the resistance can be calculated by using the following equation:

$$R_s = \frac{R_L(5\,V - V_{out}\ \ )}{V_{out}} \tag{j}$$

In order to start measuring the concentration of smoke particles in the air, the value of R_0 must be measured experimentally. Note that R_0 is the sensing resistance, R_s in 1000 ppm of hydrogen (which cannot be easily measured) but it is known that the ratio of R_s/R_0 =10 in clean air. The analog output of the smoke detector was measured outside in clean air for 5 consecutive days in order to find an accurate average value of R_0. Each day, 60 measurements were taken, and the average of those measurements were recorded in *Table 8.* Note that the MQ-2 smoke sensor must warm up for approximately 10 minutes to start reading accurate values.

$$R_{air} = \frac{R_L(5\,V - V_{out}\ \ )}{V_{out}} \tag{k}$$

$$\frac{R_{air}}{R_0} = 10 \ \rightarrow R_0 = \frac{R_{air}}{10} \tag{l}$$

	Day 1	Day 2	Day 3	Day 4	Day 5
V_{out}	0.660 V	0.682 V	0.645 V	0.681 V	0.66 3V
R_{air}	119.0 kΩ	114.6 kΩ	122.2 kΩ	114.8 kΩ	118.4 kΩ
R_0	11.9 kΩ	11.46 kΩ	12.22 kΩ	11.48 kΩ	11.84 kΩ

Table 8: MQ-2 Gas Sensor Test Data

The value that is used to R_0 is found by taking the average value of R_0 over the five days. This results in a value of R_0 = 11.78 kΩ. When the value of R_0 is known, and the value of R_s is measured via the smoke sensor, the ratio can then be calculated using the following code (where volts is the voltage measured at the analog output of the smoke sensor):

```
rl = 18100
r0 = 11780
```

```
rs = (r0*(5- volts))/volts
ratio = rs/r0
```

The parts per million of smoke particles can then be found using the equation (h) and equation (i)

The WEDS Raspberry Pi requires a digital input so the MQ-2 gas sensor output will go to an analog to digital converter (ADC) then to the Raspberry Pi where it can determine if the PPM value is high enough to consider a fire in the nearby area.

Figure 17: MQ-2 Gas Sensor

Figure 18: ADS 1115 16-bit Analog to Digital Converter

Interfacing with the Analog to Digital Converter

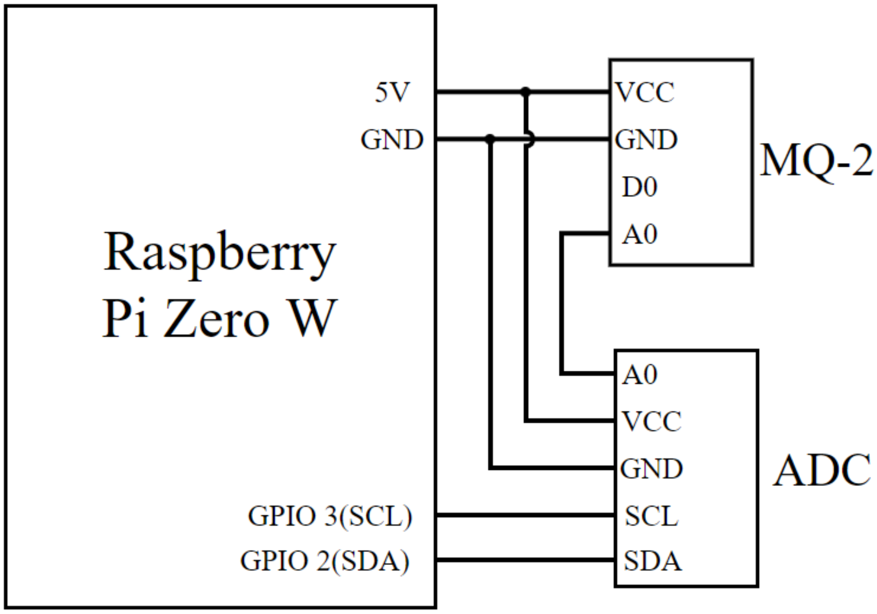


Figure 19: MQ-2 and ADC with Raspberry Pi Diagram

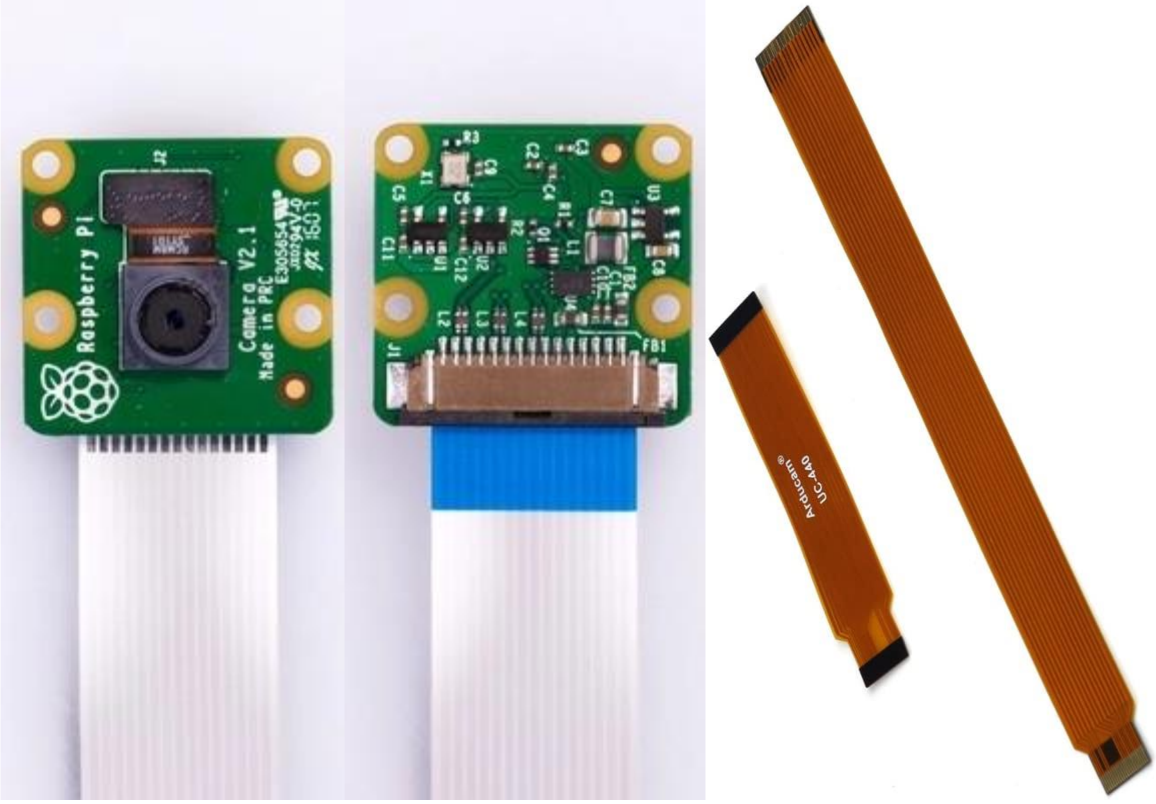
For the Raspberry Pi to interface with the ADS1115 analog to digital converter, the Adafruit Blinka library must be installed [25]. This can be done by entering the following into the command line:

```
sudo pip3 install adafruit-circuitpython-ads1x15
```

The voltage can then be read using the raspberry pi because the ADS1115 will convert the analog voltage and transmit that value to the Raspberry Pi using I2C.

Camera:

The WEDS system uses the Raspberry Pi Camera Module V2 to maintain a video feed of the surrounding area. The frames of the video serve as input to the system's computer vision algorithms. The module mounts to the body of the WED's enclosure and connects to the Raspberry Pi Zero W's CSI port via a 15cm ribbon cable. The camera features the Sony IMX219 sensor with a 3.68 x 2.76 mm sensor area [26]. It captures 1080p30 video but is also capable of 720p60 and 640 x 480p60/90.



(a) *(b)*

Figure 20: (a) Raspberry Pi Camera Module V2 [27] (b) Arducam 15 Pin 1.0mm Pitch to 22 Pin 0.5mm Camera Cable for Raspberry Pi Zero [28]

Since the Raspberry Pi Zero W uses a smaller CSI camera connected than the one shipped with the camera module originally, a 15 pin 1.0mm pitch to 22 pin 0.5mm pitch adapted must be used with this module [28].

Enclosure:

The enclosure design needs to be robust, durable, and reliable. Because the WEDS device will be placed outside, size is not a top priority. The box enclosure will contain all of the electronic components and will contain a lid that users can open to perform maintenance when needed. To secure the enclosure, there will be a latch with a lock.

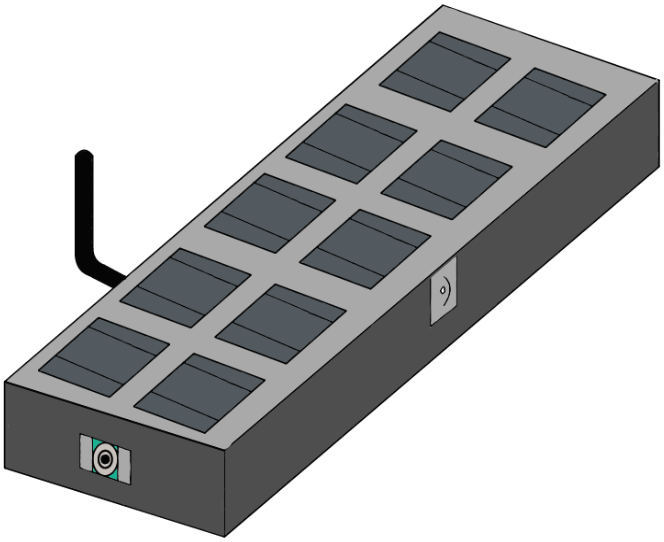


Figure 21: Preliminary Sketch of the Enclosure

The WEDS device uses 10 solar panels, each being 5.9" x 5.12". The solar panels will be arranged in a 2x5 pattern meaning the enclosure needs to be at least 25.6" long and 12" wide. Each solar panel will be mounted on a solar panel plate which would then be mounted to the lid through nuts and bolts. This is so that the solar panels can be easily replaced if damaged. There will be a larger hole drilled out in each solar panel plate for the solar panel wires which will run and connect on the underside of the lid. This is shown in *Figure 23*. There will also be two LED strips mounted on the underside of the lid for technicians when the WEDS device is being serviced. Both the solar panel plate and enclosure lid will be made of aluminum as the material is cheap, durable, and easy to manufacture.

The box enclosure will be constructed out of wood with 3/4" thick walls. Inside the enclosure, all the electronic components, battery, switches, wires, and lid stands will be mounted using nuts and bolts. *Figure 22* shows a preliminary outline of how all the components will be laid out. The two lid stands will be attached to a hinge connected on the inside of the box and will have Velcro strips attached at the top to hold the hid up while the device is being serviced. The two switches will be located towards the front of the box. One switch to turn on and off the LED strips on the underside of the lid and another switch to change between solar panel charging and micro-USB charging. The Solar/USB charging switch is primarily used for testing and should be switched to solar when the device is out in the field. The LoRaWAN module and ADC will be added and connected to a protoboard (See *Appendix D: Protoboard Wiring Schematic* for details) with a Raspberry Pi GPIO extension board which will connect to the Raspberry Pi through a 40-wire ribbon cable. The PiJuice will be mounted on top of the Raspberry Pi with the LiPo battery connected to the power terminal blocks of the PiJuice. There will be cutouts for the MQ-2 gas sensor, camera module and coaxial cable connection which will be located on the backside of the enclosure for the antenna. Ideally, the entire enclosure would be mounted on a tall pole/post to allow for the camera to view wide areas. The module would also be tilted slightly south to maximize solar panel efficiency.

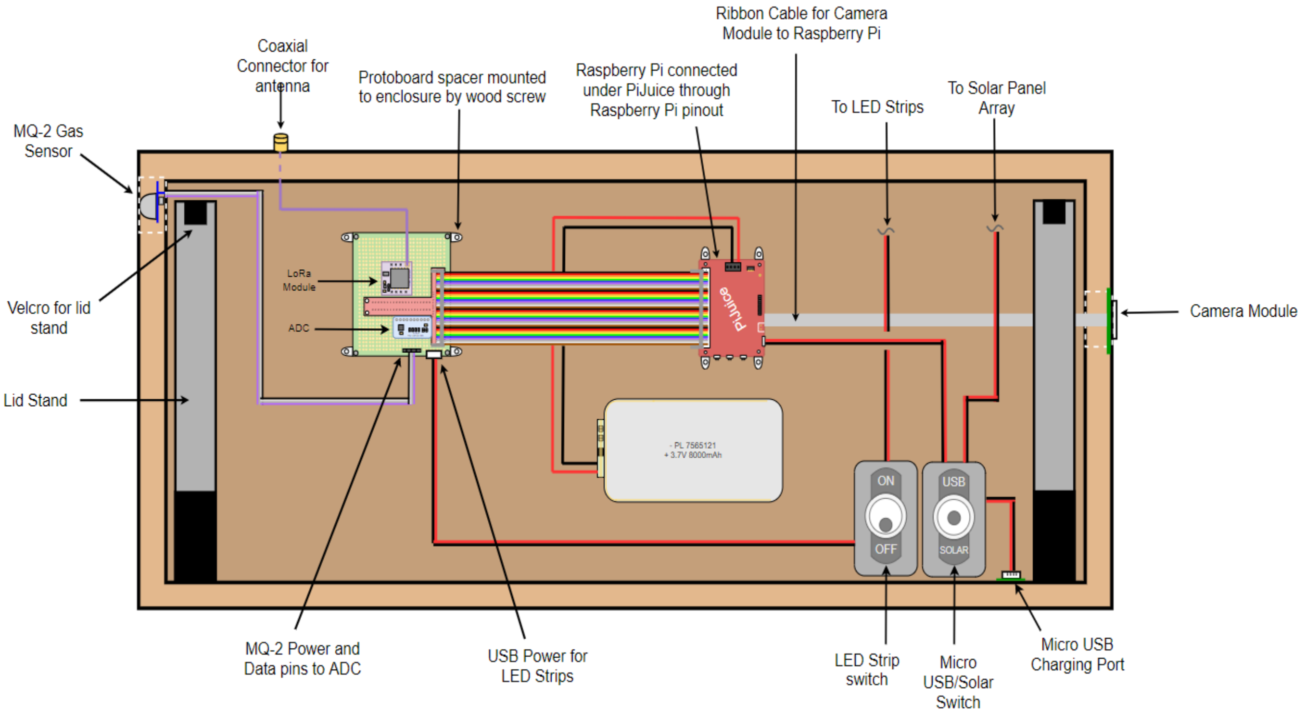


Figure 22: Inside Layout of Components

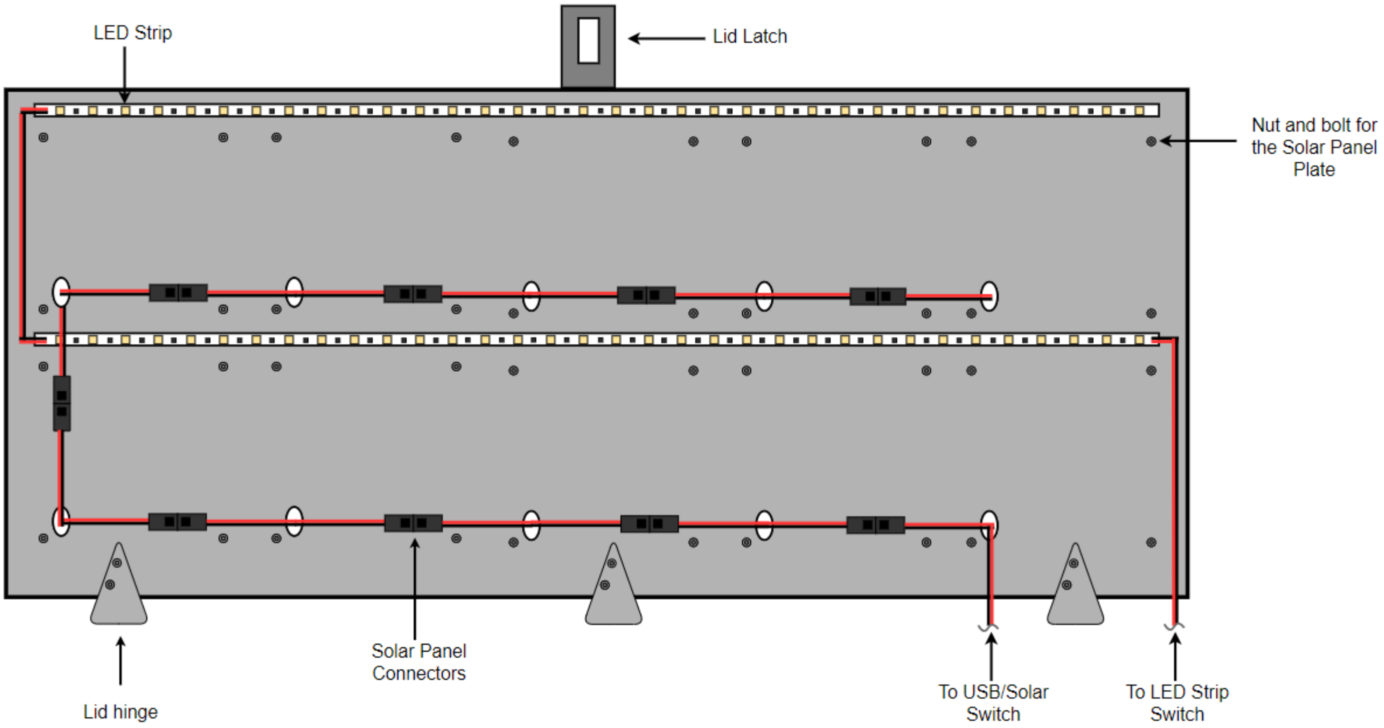
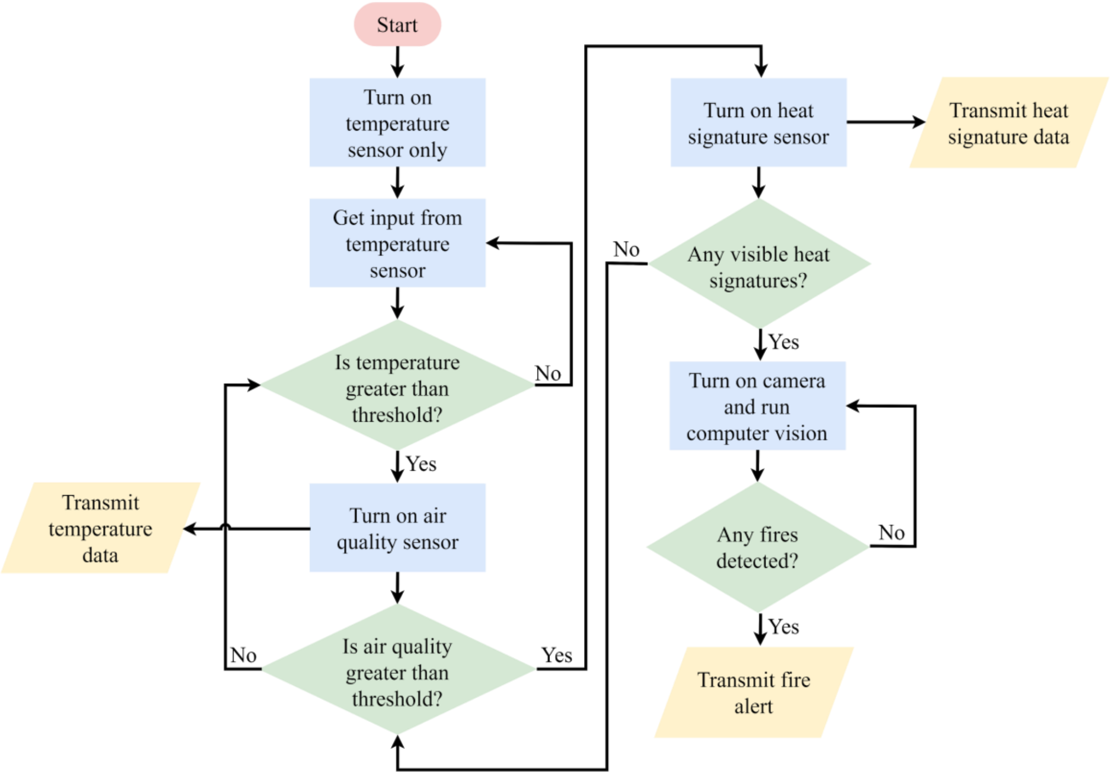


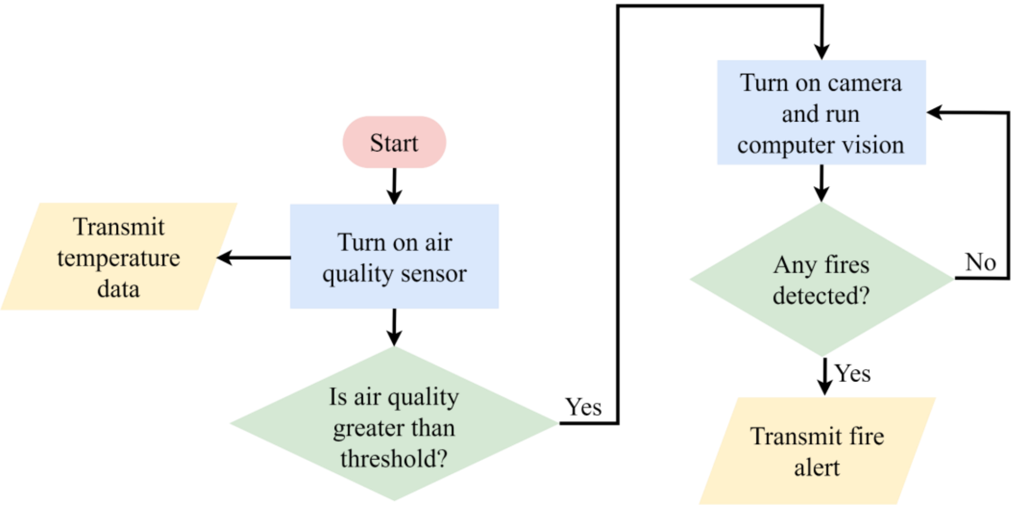
Figure 23: Underside of Enclosure Lid

Software Definition

Figure 24a below is a flowchart which originally defined the software operation of the WEDS device. This was back when the planned design included a temperature sensor and heat signature sensor. The addition of these two modules would make the system more robust and reduce false positives when it comes to detecting a fire. However, for the final design, the modules were omitted due to budget restrictions. Future engineers working on this device may intend to include those modules to further improve the WEDS device.



(a)



(b)

Figure 24: (a) Original Software flowchart for the Raspberry Pi Zero W. (b) Current Software flowchart for the Raspberry Pi Zero W.

The following Figure 24b above depicts the current flowchart for the current design of the WEDS device. The current design features an air quality sensor which is the smoke detector and the computer vision code which utilizes the Raspberry Pi Camera.

Computer Vision Code

Computer vision, on the surface, is focused on getting computers to interpret digital images and videos in the same way that humans do. Something as simple as detecting and classifying an object is a computer vision problem which is not as simple as it seems when implemented in software. For this project, the problem is detecting a wildfire within an image. Several factors contribute to the difficulty of this task: variations in environment, lighting, color, fire size, and distance from the fire. As humans, we can easily discern a fire when we see one since we have an understanding of what it is through our experiences. Teaching a computer to do the same requires a different approach that modern computer vision and machine learning is capable of.

OpenCV

Open-Source Computer Vision Library (OpenCV) is an open-source library used for machine learning and other projects [29]. We chose this library for its popularity and simplicity in terms of software development. It provides thousands of algorithms to streamline computer vision development especially for beginners. These algorithms are capable of recognizing objects, tracking movement, augmented reality, and much more. OpenCV also provides an abundance of documentation and tutorials on how to use and navigate their extensive library. Not having to write these algorithms from scratch saves us a lot of time and effort during our project timeline. While our project uses OpenCV 2 in Python, OpenCV also supports development in C++ (in which it is natively written), Java, and MATLAB.

Haar Cascade Classifier

Our first method of object detection is the Haar Cascade classifier which Paul Viola and Michael Jones developed [29]. This is a machine learning based approach that trains from two separate pools of positive and negative images. Positive images contain the target that the system needs to detect and negative images do not contain that target. In our case, that target is a wildfire. This approach uses various features, like edges and lines, to detect objects. Mathematically, these kernels, or filters, describe these features in terms of a matrix. These matrices then slide over the whole image, multiplying itself with 3x3 areas of the image to bring out some features and smooth others out. In a given model, there can be several of these kernels in a cascade, hence the cascade classifier. The images below depict what these kernels look like visually and numerically.

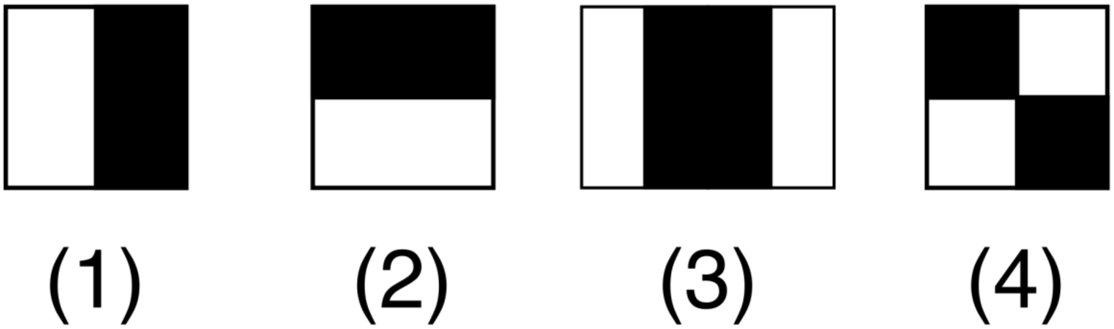
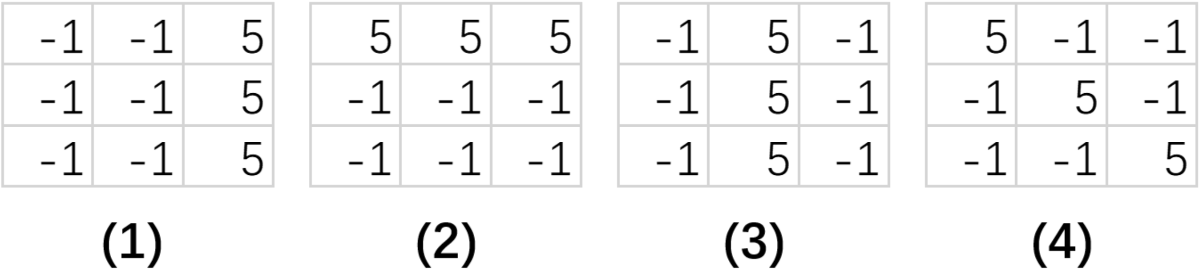


Figure 25: Example kernels taken from towards data science [29].



-1	-1	5
-1	-1	5
-1	-1	5

(1)

5	5	5
-1	-1	-1
-1	-1	-1

(2)

-1	5	-1
-1	5	-1
-1	5	-1

(3)

5	-1	-1
-1	5	-1
-1	-1	5

(4)

Figure 26: Numerical representations of the above kernels taken from towards data science [29].

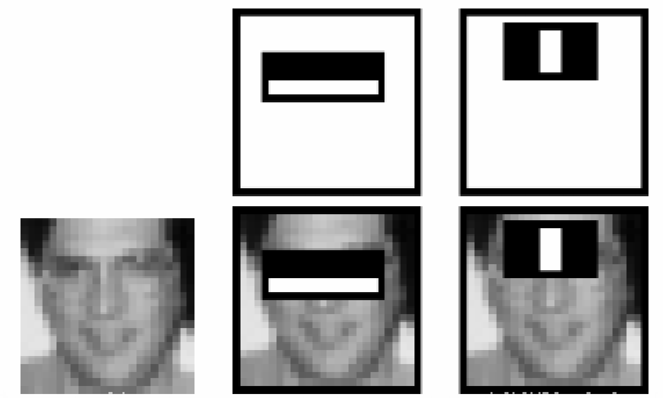


Figure 27: Kernel sliding over an image to detect face features taken from OpenCV documentation [29].

The difficulty with the Haar Cascade classifier is that wildfires vary greatly in size and shape. This means the edges are not consistent between different training images and thus it requires an extensive dataset to achieve consistency with this algorithm. We are hoping that with an adequate pool of positive images, we can get this classifier to achieve above average results.

We tested a cascade classifier and trained it to detect the flame of a lighter. We trained it with 50 positive images and 50 negative images. While these images could be different sizes, each image must be equal or larger than the desired training window size. We ensured that the positive images had as much variation of the flame as possible. We used the OpenCV tool *opencv_annotation.exe* to annotate each positive image, indicating where the flame is by drawing a green box around it. After doing that, we generated a positive description file which contained a list of paths to all the positive images. Next to each positive image were pixel coordinates describing the rectangle that we drew for that image. From this file, we generated a positive vector file using the OpenCV tool *opencv_createsamples.exe*. We also created a negative description file which is similar except that there are no corresponding pixel coordinates for each image. We didn’t need a special tool for this, so we used a simple Python script.

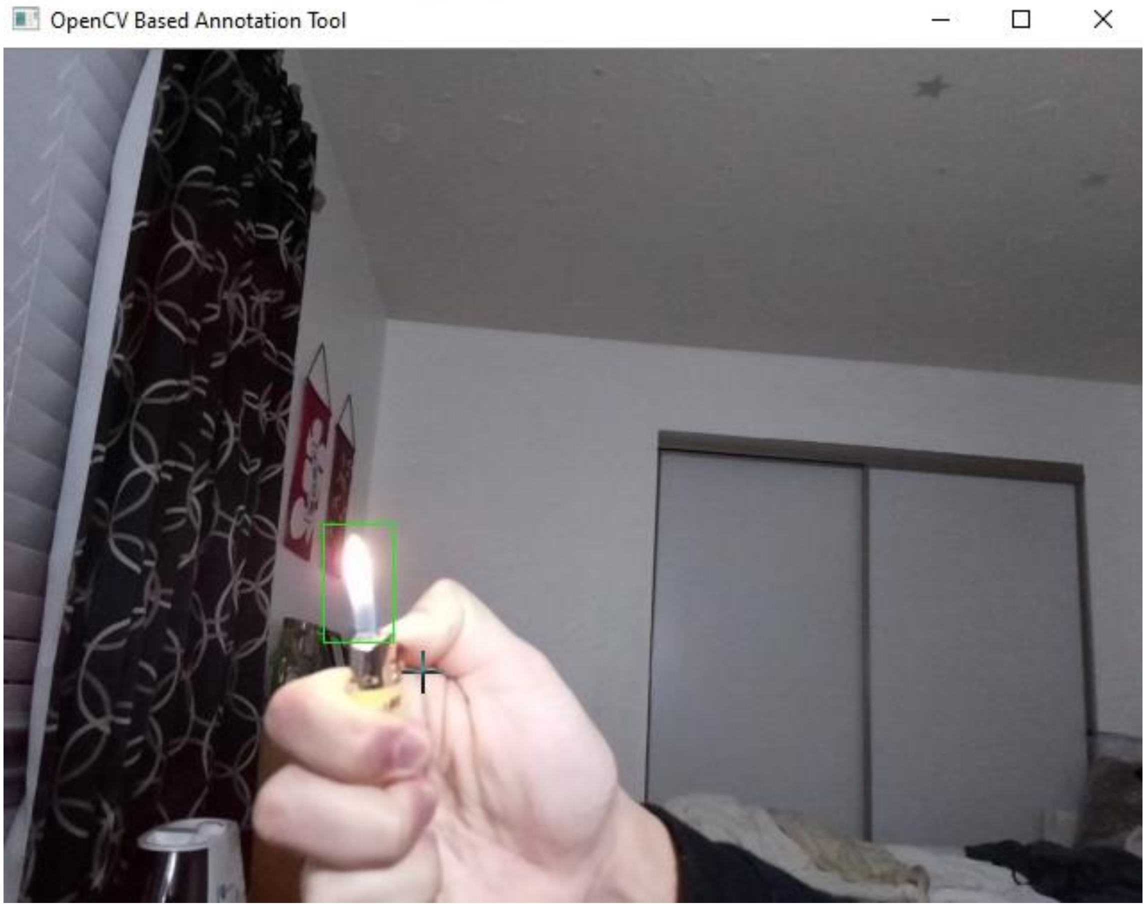
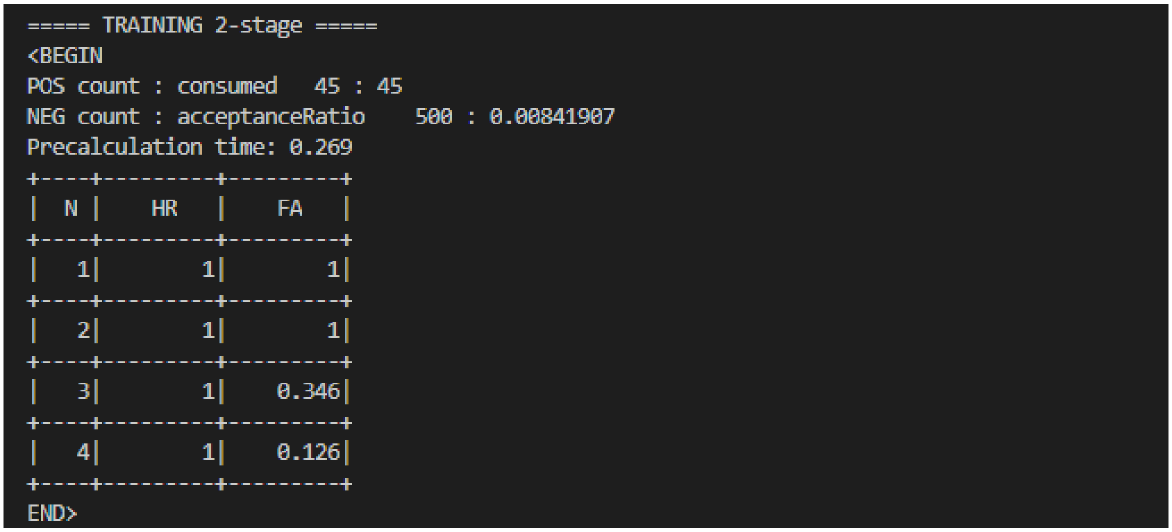


Figure 28: Manual annotation of positive images to indicate where the flame is prior to compiling the cascade classifier model.

After generating the positive and negative description files and the positive vector file, we were ready to train our model using another OpenCV tool *opencv_traincascade.exe*. We had control over how many stages we wanted to train, and the number of stages was proportional to the overall training time. These stages, or kernels, are filters that the model uses to extract features from images. While more stages might lead to a better model, we were aware that too many stages might also lead to overtraining. An indication of overtraining is if the system detects more objects than just the flame while it is running. Another sign of overtraining is a smaller negative acceptance ratio specified in *NEG count: acceptanceRatio*. In the table output, N is the weak layer number, HR is the hit rate, and FA is the false alarm rate. We want the layers near the end to have the smallest false alarm rate since it has the finest amount of detail.



```
===== TRAINING 2-stage =====
<BEGIN
POS count : consumed   45 : 45
NEG count : acceptanceRatio    500 : 0.00841907
Precalculation time: 0.269
+----+---------+---------+
|  N |    HR   |    FA   |
+----+---------+---------+
|   1|        1|        1|
+----+---------+---------+
|   2|        1|        1|
+----+---------+---------+
|   3|        1|    0.346|
+----+---------+---------+
|   4|        1|    0.126|
+----+---------+---------+
END>
```

Figure 29: Example terminal output of one of the training stages.

There are multiple parameters to consider when building the cascade models [30]. Below is an example terminal command to build the model using *opencv_traincascade.exe*.

```
C:/Users/Jayare/Downloads/opencv/build/x64/vc15/bin/opencv_traincascade.exe -data
old_cascades/cascade_019/ -vec pos.vec -bg neg.txt -w 24 -h 24 -precalcValBufSize 10000 -precalcIdxBufSize
10000 -numPos 50 -numNeg 60 -numStages 15 -maxFalseAlarmRate 0.1 -minHitRate 0.999999 >
old_cascades/cascade_019/command.txt
```

Required information for this tool includes the path to the positive vector file and the negative description file specified by the flags -vec and -bg, respectively. The flags -w and -h are the width and height in pixels defining the smallest image that this model can detect which must be the same as the parameters used when creating the positive vector file. The flags -precalcValBufSize and -precalcIdxBufSize determine how much system memory this tool should use to speed up the training process. The flags -numPos and -numNeg specify the number of positive and negative samples, respectively, for each cascade stage. The flag -numSatages refers to the number of stages to train. The flag -maxFalseAlarmRate is the max false alarm rate for each stage. The tool will continue to add layers for each stage until it achieves that rate. The same goes for the flag -minHitRate, except that the tool will only add layers as long as the hit rate stays above the specified value.

For the test, we arbitrarily trained it for seven stages, which turned out to be adequately consistent. The training generated *.xml files that we used in a Python script. In this script, we used OpenCV's *CascadeClassifier* class to load the .xml file which is our cascade classifier. This class has a method called *detectMultiScale* which takes an input image and returns a list of rectangles where the model attempted to detect our target object. We then used that list and drew rectangles on the frame and displayed it to the screen. Here is a video of the system detecting a flame. In this video, we have the Raspberry Pi connected directly to a monitor via HDMI. We fed the image input, coming from the aforementioned Raspberry Pi Camera, directly into our script and displayed each incoming image.

After running that initial test, we attempted to build a model using wildfire images found on Google images. This proved to be unsuccessful at first because not only did the images we found online vary in size, but they also varied in file type. So, we only included images with *.jpg file endings and also wrote a Python script to resize all the images to an arbitrary fixed resolution of 1200 x 800. This led to a successful build of a model. We expected that once we have a model successfully compiled, it would not be nearly as consistent as the simple test model we generated for the lighter. Thus, it was a matter of tweaking the training parameters until we got a desired result. The process of refining this model resides under the Testing and Verification section.

Figure 30: Annotating an image containing a wildfire.

Convolutional Neural Network

Our second means of object detection is a convolutional neural network (CNN) through Keras and Tensorflow. However, we ran into difficulties importing the TensorFlow libraries onto the Pi Zero so we must leave it to future improvements to implement a CNN. This most likely requires a stronger computing machine like the NVIDIA Jetson AGX Xavier which is geared towards AI development and machine learning [31]. The difference here is that unlike in the Haar cascade where we manually determine the kernel values by annotating each image, a CNN learns the values of these kernels on its own. Theoretically, it should detect more than just a subset of features like edges and lines and overall be more reliable than the Haar cascade.

Construction and Build

Enclosure:

The enclosure is 35” long by 12.25” wide with 3/4” thick walls. The weight of the entire enclosure with the lid and components is about 15lbs.

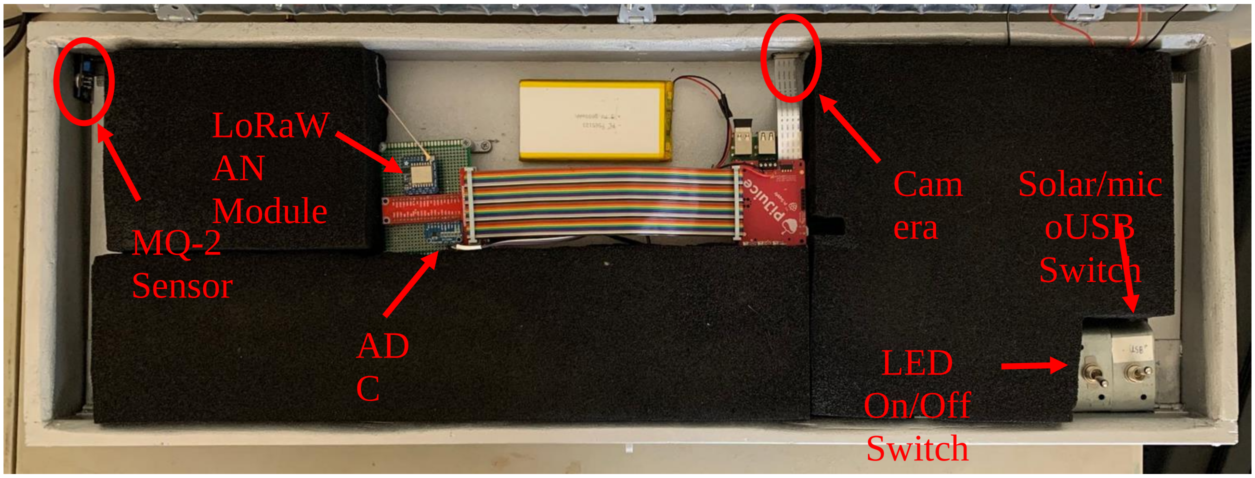


Figure 31: Component Layout

On the inside of the enclosure, all the components are laid out similarly to the design proposed in the *Enclosure Design* position of the report as shown in *Figure 31*. The protoboard and Raspberry Pi were mounted to the bottom side of the enclosure with nylon stands and screws while the camera and smoke sensor were mounted in a cutout on the walls of the enclosure. See *Smoke Sensor, Camera, and Coaxial Connector Cutouts* section for more information. The LoRaWAN and ADC were soldered onto a protoboard. See the *Protoboard Layout* section in the *Construction and Build* section for additional information. The two toggle switches are mounted on aluminum brackets that are screwed in on the bottom side of the enclosure. On the ends of the enclosure, there are two lid stands which are used to prop up the lid for maintenance as shown in *Figure 32*. Foam pads were also added to the inside of the enclosure for durability and aesthetic reasons.

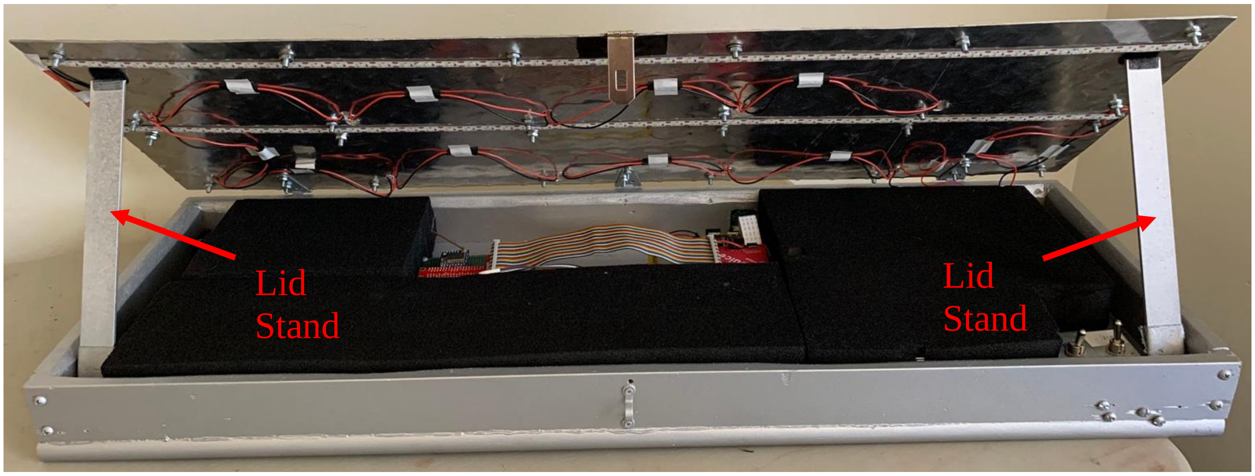


Figure 32: Lid Stand

In order to make testing easier, a toggle switch was used to switch between solar and USB battery charging. The power from the solar panel and the USB goes to the input of the PiJuice which would then charge and power the Raspberry Pi. A micro-USB charging port was added to the interior front side of the enclosure as shown. *Figure 34* shows the wiring diagram between the toggle switch, solar panels, and microUSB charging port.

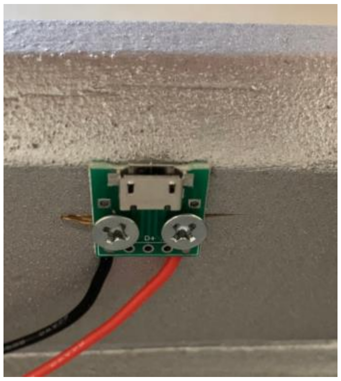


Figure 33: microUSB Charging Port

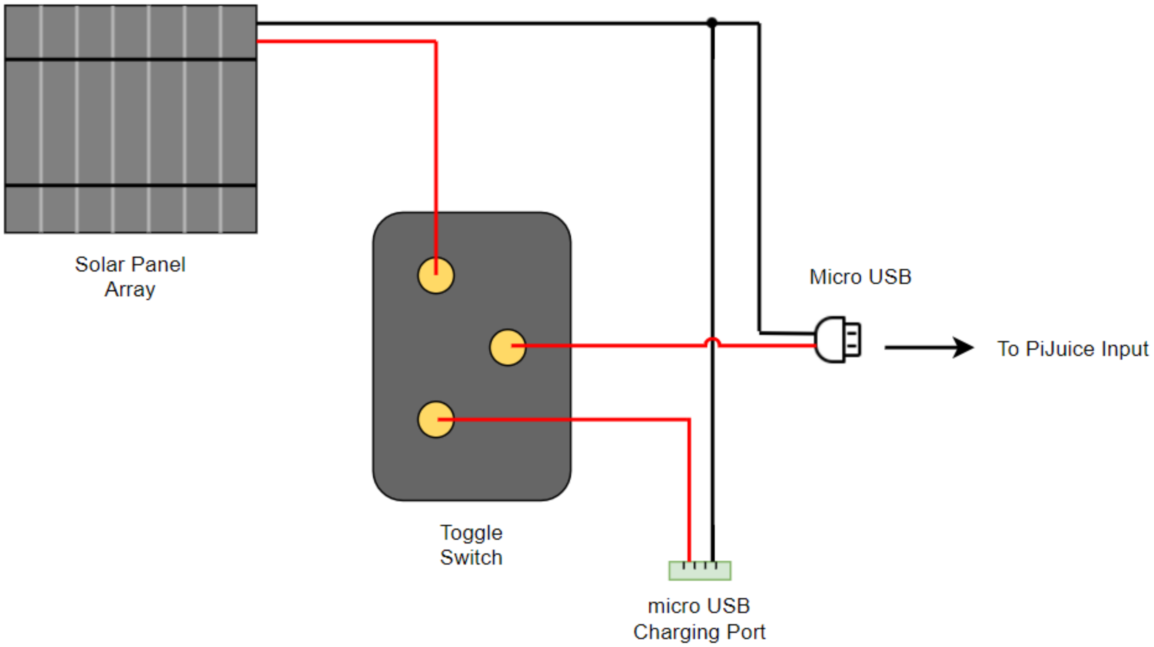


Figure 34: Diagram of Charging Toggle Switch

Solar/Enclosure Lid:

The final enclosure lid was constructed using 1/16" aluminum with the solar panel plates made of 1/50" aluminum. The solar panels were laid in a 2x5 pattern as shown in *Figure 35* with the wires feeding to the underside of the enclosure lid through a hole in the solar panel plate**.** There were two sets of jumper wires soldered to the solar panels so that they could be connected in parallel.

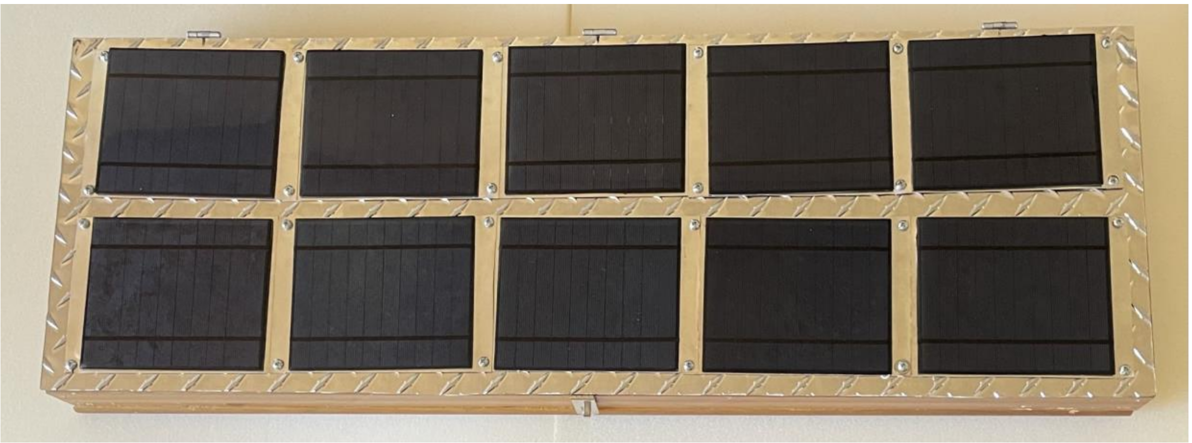


Figure 35: Solar Panel Array Layout

Figure 36: Solar Panel Plate

There are two LED strips that run on the underside of the enclosure led as shown in *Figure 37* and connect to a switch inside the enclosure. The LED receives its power from the Raspberry Pi through a micro-USB port connected on the protoboard (See *Appendix D: Protoboard Wiring Schematic*) which is connected to the 5V port on the Raspberry Pi.

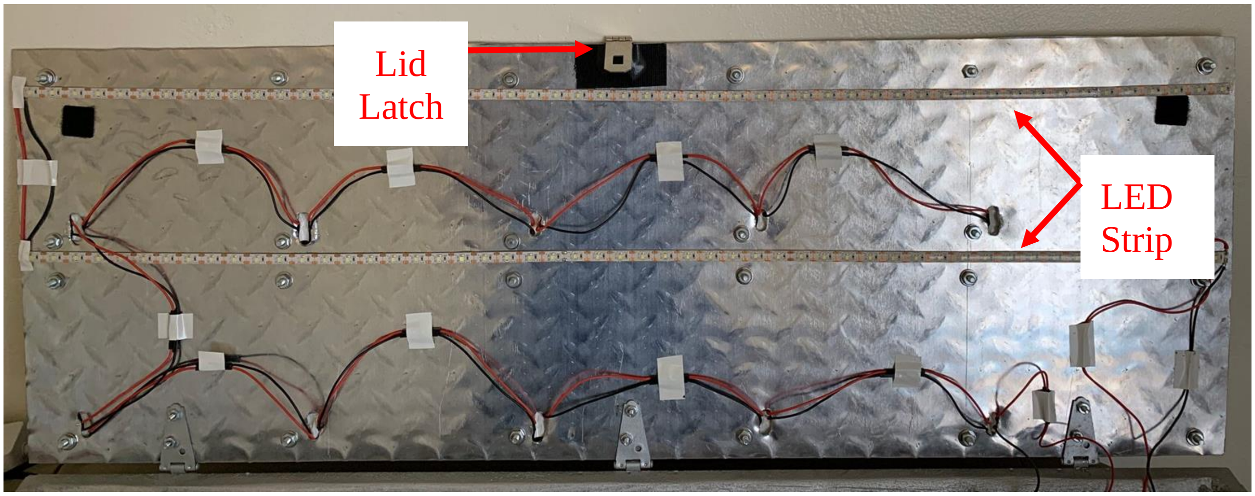


Figure 37: Underside of Enclosure Lid

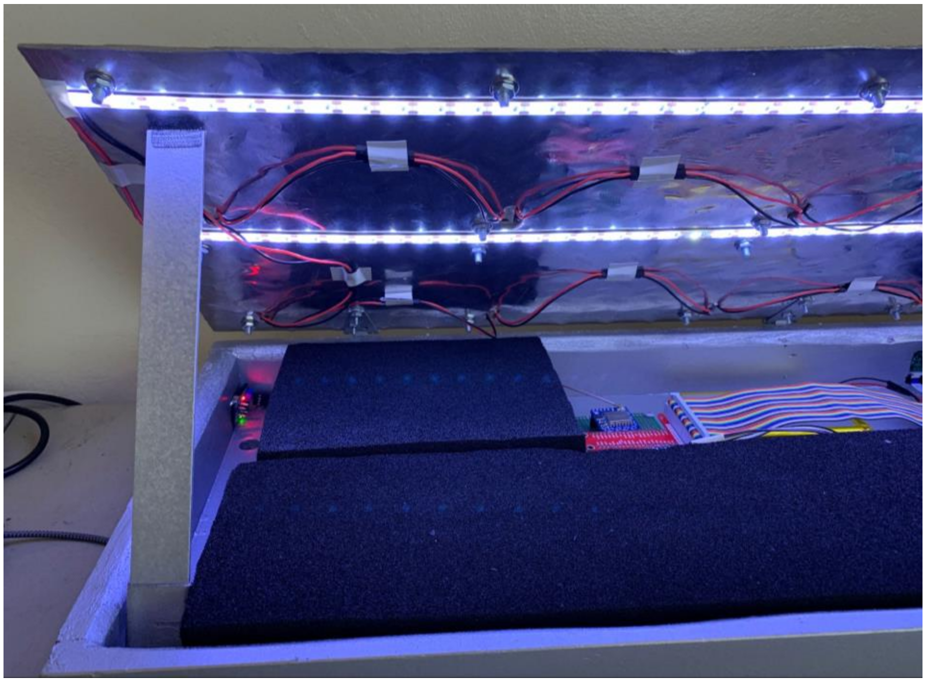


Figure 38: Propped up lid with LEDs turned on

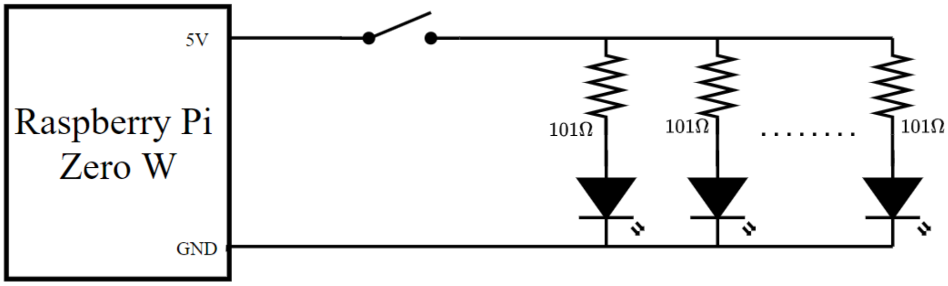


Figure 39: LED Strip Wiring Diagram

Protoboard Layout

A protoboard was used to connect the LoRaWAN and ADC to the Raspberry Pi via a GPIO breakout board. The GPIO breakout board is connected to the Raspberry Pi through a 40-pin ribbon cable. Because the MQ-2 gas sensor is mounted on the side of the enclosure, a 4-pin header was used to connect the ADC pins to the MQ-2 gas sensor through a longer cable. The microUSB power port for the LEDs is also located on the protoboard. *Figure 40* shows a top view of the protoboard. For the protoboard wiring details see Appendix C: Protoboard Wiring Schematic. The protoboard is mounted on 4 nylon stands to isolate the bottom of the protoboard from the enclosure.

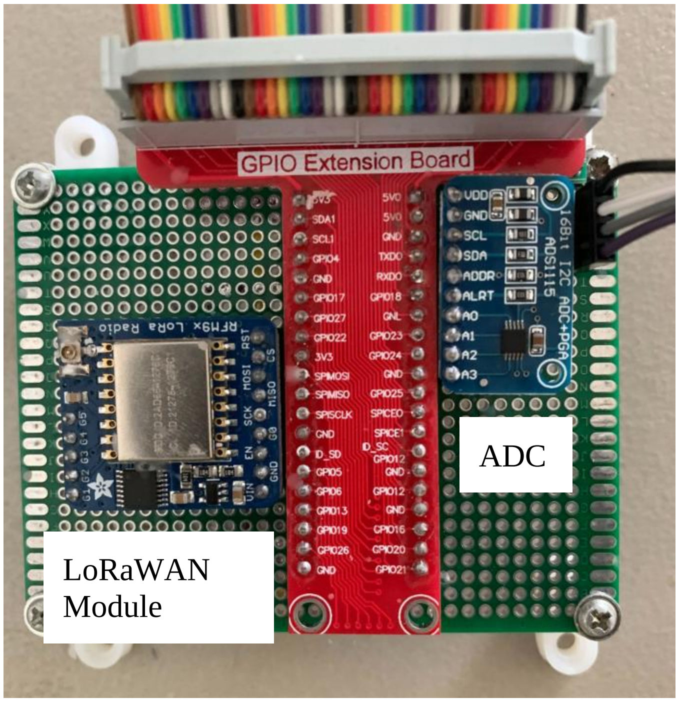


Figure 40: Protoboard Layout

Smoke Sensor, Camera, and Coaxial Connector Cutouts

The MQ-2 gas sensor, camera and coaxial connector each have their own designated cutouts. The MQ-2 gas sensor cut out is approximately 0.8” in diameter which is enough for the sensor to stick out of the enclosure and read the ambient air. The camera cutout contains a rectangular portion that houses the camera PCB while the hole allows the camera to see outside of the enclosure. The coaxial connect hole is approximately 0.2” in diameter and is just big enough to fit the coaxial connector cable through the hole. The antenna is then screwed into the coaxial connector.

Figure 41: MQ-2 Sensor Cutout Exterior

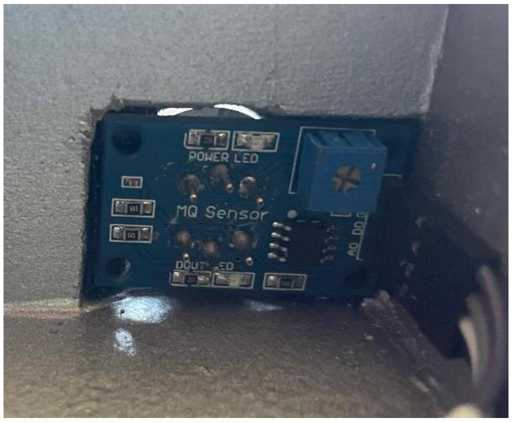


Figure 42: MQ-2 Sensor Cutout Interior

Figure 43: Camera Cutout Exterior

Figure 44: Camera Cutout Interior

Figure 45: Coaxial Cable

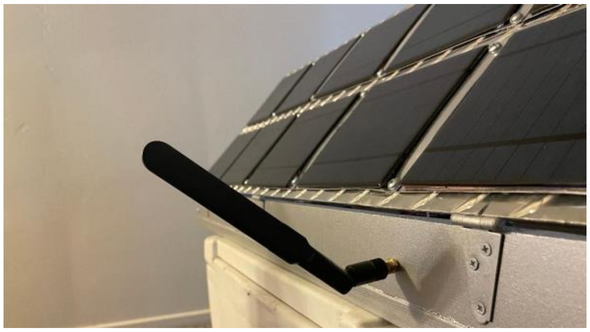


Figure 46: Antenna Connected

WEDS Central Hub:

The WEDS central hub is a Raspberry Pi connected to a monitor with a LoRaWAN module that receives data coming from the WEDS device. A larger antenna is used to pick up the transmitting signal from the WEDS device. The monitor is used to view the communication between the WEDS module and the WEDS central hub.

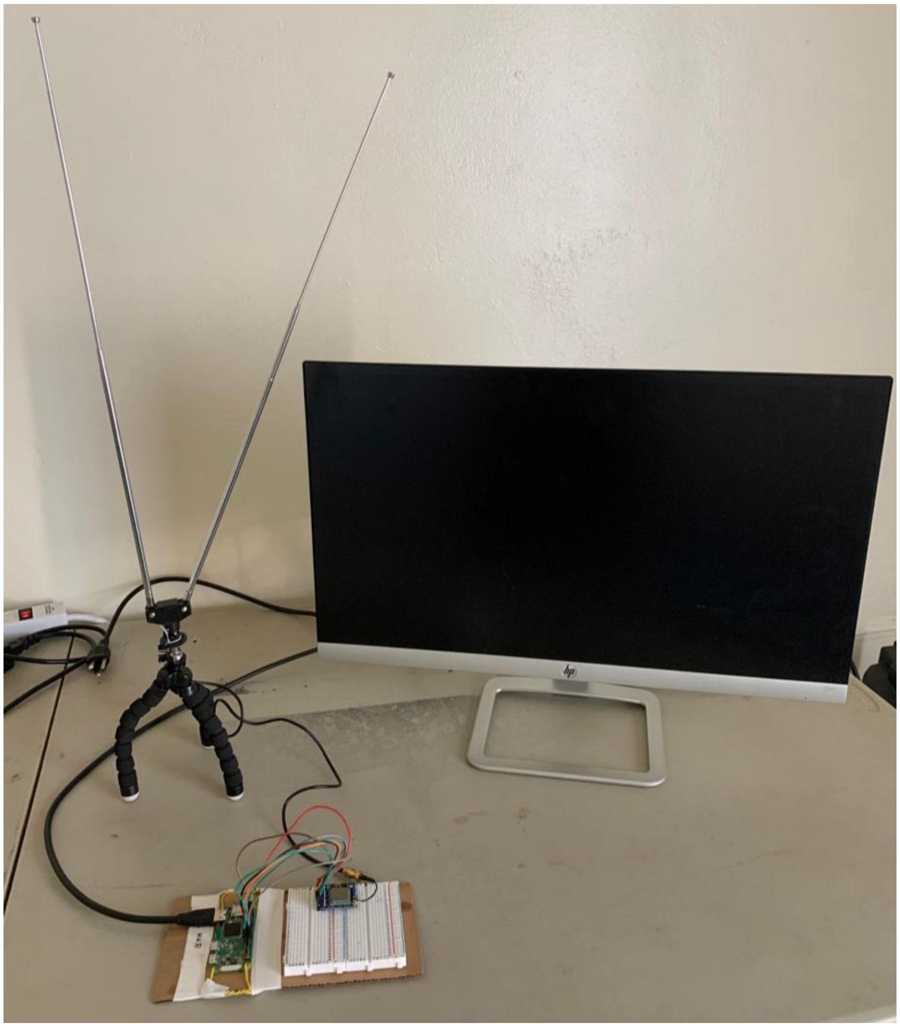


Figure 47: WEDS Central Hub

Testing and Verification

Smoke Sensor:

The calibration of the MQ-2 smoke sensor is explained in the smoke sensor section on page 28. This gives a baseline for the R_0/R_s resistance ratio in air that is virtually smoke free. This ratio is directly dependent on the voltage on the analog pin of the smoke sensor.

During the testing of the smoke detector, readings were very inconsistent outdoors. While the WEDS device is near a fire, there is only a very slight increase in the voltage that is read on the analog pin of the smoke sensor. The only significant increase in the smoke sensor voltage occurred when the smoke sensor was held directly above the fire. A plot of a single test of the smoke sensor held directly above the fire is compared to a plot of a single test of the smoke sensor held adjacent to the fire is shown in *Figure 48* below. The smoke sensor voltage was read 60 times for each case where readings were done every 3 seconds.

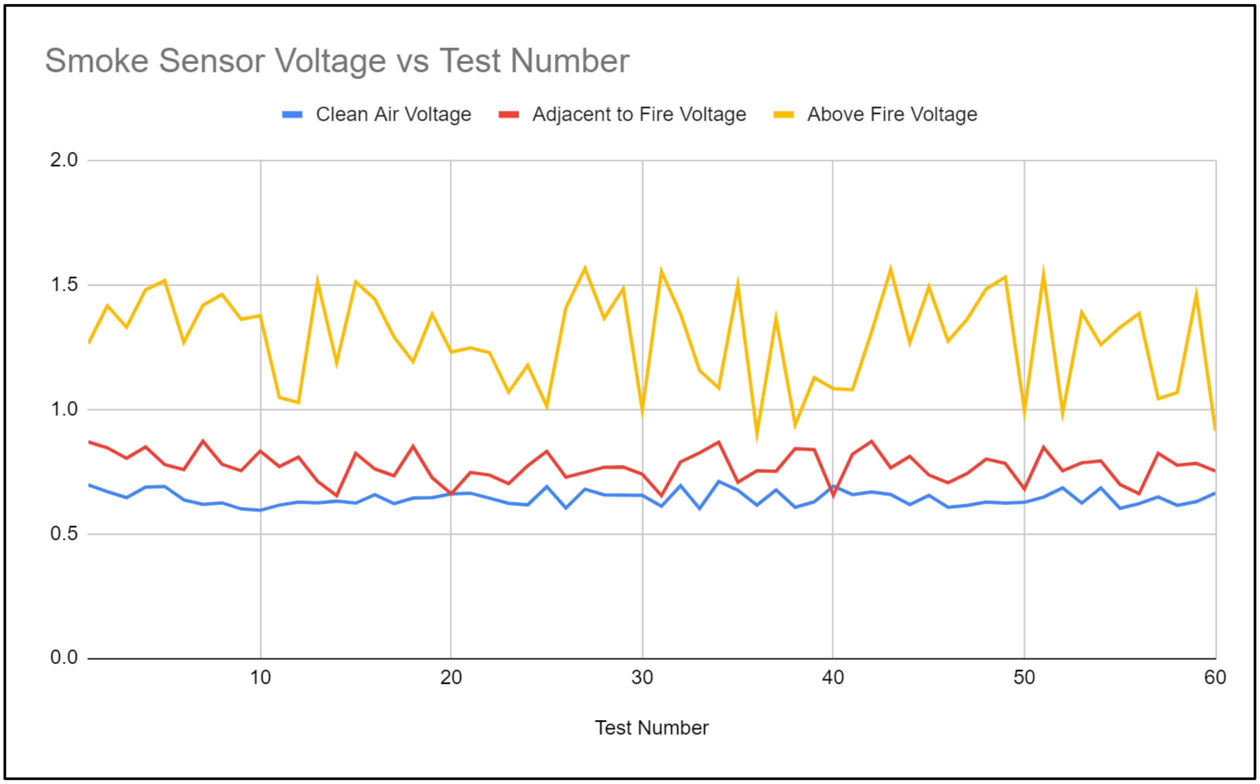


Figure 48: Smoke Sensor Voltage vs Test Number Graph

Detecting fires with the smoke detector is inferior to the computer vision method because the camera can detect fires at a distance while the smoke detector must be directly above the fire. In many cases by the time the smoke detector voltage is high enough for evidence of a fire to be apparent, the wildfire may have already increased to uncontainable levels. This is why the computer vision algorithm is the main fire detection method and when a fire is detected, smoke levels are transmitted to the central hub. Smoke levels may help determine the distance that the fire is from the central hub.

Camera with Computer Vision:

We gathered a dataset of 146 images of wildfires and 64 images of non-wildfires to train our Haar cascade model with. Since consistent testing with real fire would be detrimental to our lungs and the environment, we did a majority of the initial testing for the Haar cascade model using online footage of wildfires. We used 32 different 30-second video clips of footage found on YouTube. To make testing easier on the Raspberry Pi's limited computing power, these videos were limited to a 640 x 360 resolution at 30 frames per second. Figure 49 below shows example frames from the pool of test footage.

Figure 49: Example frames from footage used to test the cascade models.

To train the model, we used an external computer with a significantly better CPU and more RAM to make this process more efficient. Under normal operation, the WEDS device will not need to train any models. The initial model contained the parameters shown below in Table 9. These values were arbitrary as we just needed a base model to work off of. The *Software Definition* section includes an explanation of these parameters. This model could not detect anything with the footage provided.

-precalcValBufSize	6000
-precalcIdxBufSize	6000
-numPos	45
-numNeg	500
-numStages	12
-maxFalseAlarmRate	0.3
-minHitRate	0.999

Table 9: Parameters of the first attempt of a Haar cascade model.

To attempt to correct the model, we adjusted -numNeg, -numPos. In the beginning, we generally kept -numPos above -numNeg. Lowering -numNeg too far under -numPos led to an overtrained model that detected random objects in the surroundings apart from the fire. Increasing -numPos too far above -numNeg led to a model that was too computationally intense to run for the Raspberry Pi. We also tried adjusting -numStages but quickly realized that too many stages were not only too intense for the Raspberry Pi to run but also too intense to train on the external PC. To provide perspective, a model with 75 stages took nearly four hours to build. While the model did work decently, waiting that long to tweak the parameters of the model proved to be too inefficient.

Testing created many different iterations of cascade models. Desirable aspects of each model built were based on how smoothly the Raspberry Pi could run the model, how consistently the model identified fires as opposed to other background elements, and the resulting negative acceptance ratio of the last stage. Isolating the adjustments between each model to -numPos, -numNeg, and -maxFalseAlarmRate provided adequate control of the model behavior. The value -minHitRate was kept constant at 0.999999 and -numStages at 15. Table 10 below shows data for training iterations for different values of -numPos, -numNeg, and -maxFalseAlarmRate. The column for NEG count: acceptanceRatio is a ratio which is part of the plethora of data the *opencv_traincascade.exe* tool provides at the end of training each stage of a model. We used the ratio of the last stage, the most detailed stage, to expect how consistent a model would be when attempting to detect a fire.

-maxFalseAlarmRate	-numPos	-numNeg	NEG count: acceptanceRatio (14th stage)	Hit Rate (14th stage, last layer)	False Alarm Rate (14th stage, last layer)	Number of layers in 14th stage
0.1	50	60	893330.3953	1	0.0833333	9
0.15	50	60	795101.117	1	0.05	7
0.2	50	60	257718.6743	1	0.0666667	7
0.3	50	60	83455.0386	1	0.0833333	6
0.15	50	100	14390332	1	0.11	7
0.15	80	100	532844.5373	1	0.12	12
0.15	110	100	100079.9639	1	0.12	12
0.15	110	150	727812.6319	1	0.14	17
0.15	110	180	873375.1583	1	0.15	19
0.15	110	200	5681221.008	1	0.145	19
0.15	130	200	1736819.709	1	0.15	23
0.15	130	220	1908827.459	1	0.127273	22
0.15	150	250	2720031.509	1	0.144	25
0.15	150	280	3571893.282	1	0.132143	30
0.15	170	280	1692384.872	1	0.15	28
0.15	170	300	2398369.109	1	0.0966667	30
0.15	170	330	2774344.876	1	0.130303	35
0.25	170	330	562925.7126	1	0.245455	30
0.1	170	330	15487504.4	1	0.0848485	40
0.125	170	330	2661333.247	1	0.112121	36

Table 10: Training data for different iterations of Haar cascade models.

In the first four rows of the table above, -maxFalseAlarmRate is the only changing variable. The next thirteen rows show changes in -numPos and -numNeg. The last three rows show final changes to -maxFalseAlarmRate in an attempt to fine tune the model. Keeping the NEG count: acceptanceRatio between 2000000 and 3000000 seemed to be the sweet spot between an overtrained and undertrained model. Additionally, the number of layers for each stage, which determines how detailed it is, increases with -numPos and -numNeg. The final iteration, cascade_019 behaved more consistently than its predecessors when attempting to detect fires using the test footage. The parameters for this model are shown below. The model can further be improved using a larger dataset and more time tweaking the model parameters. The test footage, cascade models, and image dataset all reside in the GitHub repository in appendix E.

-precalcValBufSize	10000
-precalcIdxBufSize	10000
-numPos	170
-numNeg	330
-numStages	15
-maxFalseAlarmRate	0.125
-minHitRate	0.999999

Table 11: Parameters of the final attempt of a Haar cascade model.

During normal operation the WEDS device is completely self-sustaining and the only information sent to the WEDS central hub is if a fire is detected on the camera, and the smoke sensor data when that fire is detected. This means that videos/images from the camera are NOT transmitted to the WEDS central hub.

During the testing phase we did spend a few days working with a firepit to simulate a wildfire and see how the WEDS device would detect that fire. The WEDS device was placed on a table with the camera facing the fire pit. The WEDS device was connected to a monitor via HDMI in order to observe the fire detection. This setup can be viewed on the demo video shown in appendix E. Before a fire was started inside of the fire pit, the haar cascade classifier did not detect any fires, and after a fire was started in the fire pit, the haar cascade classifier was able to detect the fire as shown in the demo video in appendix F.

LoRaWAN Wireless communication protocol:

The LoRaWAN modules were taken to an open space and the communication distance was tested. The LoRaWAN module claims to have the ability to communicate at 2km (1.24 mi) line of sight, but this was not what was observed during our testing. The farthest distance that the receiver could get a constant signal was approximately 354 ft which is only 0.07 mi. The line-of-sight test is shown in *Figure 50* and the measurement made on google maps is shown in *Figure 51*.

Figure 50: LoRaWAN line of sight test.

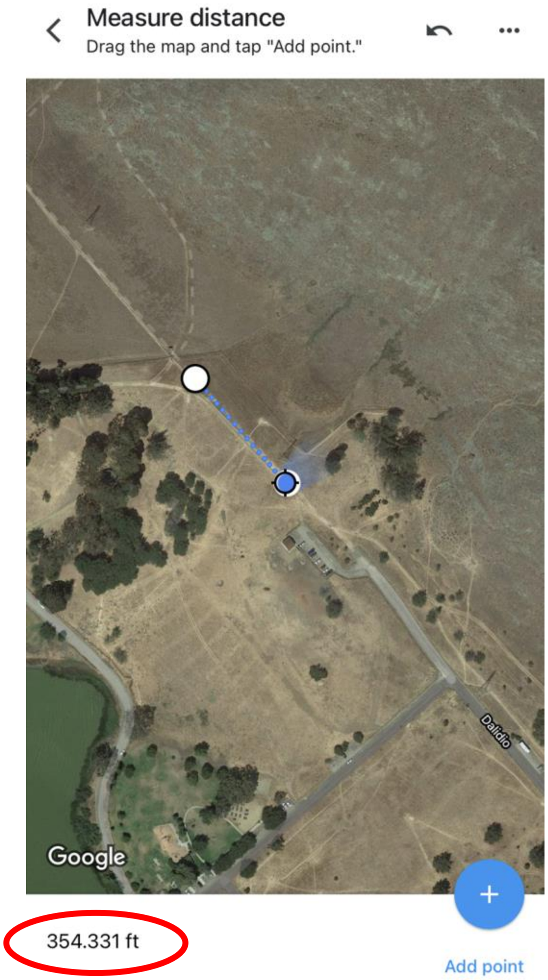


Figure 51: LoRaWAN distance measurement made on Google Maps.

Battery and Solar Panel Charging:

The data that was calculated and tabulated in *Table 6* was experimentally measured and tabulated. It was not possible to keep the charging currents constant throughout the charging process so as while *Table 6* gives an idea of how long it will take to charge the battery at 3 different constant currents, *Table 12* records the time it actually took for the battery to charge via the solar array using the PiJuice as the charge controller. Note that in the field the charging current will change depending on the amount of solar energy that is being harvested as well as the charge level of the battery. This is why there is only one experimentally measured charging time as opposed to *Table 6* where the charging time is calculated at three different charging currents.

Battery Charge Capacity	8000 mAh
Current Draw	0.7 A
Power Consumption	3.5 W
Energy in Battery	33.6 Wh
Battery Discharge Time	13.5 hrs
Battery Charge Time	5.5 hrs

Table 12: Experimental Data using 8000 mAh Battery

(Compare with theoretical calculations shown in Table 6)

References

[1] J. Smith, “These Surreal Photos of the West Coast Wildfires Look Like the End of the World”, Prevention, Hearst Magazine Media, Inc., September 10, 2020. https://www.prevention.com/life/a33980276/wildfires-photos-west-coast-2020/

[2] P. Boghani, “Camp Fire: By the Numbers”, FRONTLINE, October 29, 2019 [Online]. Available: https://www.pbs.org/wgbh/frontline/article/camp-fire-by-the-numbers/

[3] EPA, “How Smoke from Fires Can Affect Your Health”, United State Environmental Protection. [Online] Available: https://www.epa.gov/pm-pollution/how-smoke-fires-can-affect-your-health

[4] S. W. Running, “Is Global Warming Causing More, Larger Wildfires?”, American Association for the Advancement of Science. [Online] Available: https://www.jstor.org/stable/3846966?seq=1&cid=pdf-reference#references_tab_contents

[5] A. BORUNDA, “The science connecting wildfires to climate change”, National Geographic, September 17, 2020. [Online] Available: https://www.nationalgeographic.com/science/2020/09/climate-change-increases-risk-fires-western-us/

[6] Z. Hausfather, “Factcheck: How global warming has increased US wildfires”, CarbonBrief, September 18, 2018. [Online] Available: https://www.carbonbrief.org/factcheck-how-global-warming-has-increased-us-wildfires

[7] Government of Northwest, “Detecting wildfire”, Government of Northwest. [Online] Available: https://www.enr.gov.nt.ca/en/services/wildfire-operations/detecting-wildfire#:~:text=One%20of%20the%20ways%20fires,to%20locate%20and%20report%20wildfires.

[8] Insight Robotics, “Wildfire Detection System InsightFD + Insight Globe”, Insight Robotics, 2018. [Online] Available: https://www.insightrobotics.com/en/services/wildfire-detection-system/

[9] Vigilys, “Wildfire Detection”, Vigilys, 2015. [Online] Available: https://vigilys.com/technology/firealert/#:~:text=FireALERT%20MK%20I%20

[10] Perch, “Wildfire Systems Detect. Predict. Protect.”, Perch San Luis Obispo, 2020. [Online] Available: https://jcallene.github.io/perchsense/

.

[11] A. A. A. Alkhatib, “A Review on Forest Fire Detection Techniques”, SAGE Journals, March 4, 2014. [Online] Available: https://journals.sagepub.com/doi/full/10.1155/2014/597368#:~:text=operational%20detection%20systems%3A%20fire%20towers,water%20tankers%20like%20in%20Canada.

[12] National Park Service, “Experience Your America”, U.S. Department of the Interior, 2020. [Online] Available: https://www.nps.gov/index.htm

[13] Raspberry Pi Dramble, “Power Consumption Benchmarks”, Raspberry Pi Dramble, 2020. [Online] Available: https://www.pidramble.com/wiki/benchmarks/power-consumption

[14] Pi Supply, “PiJuice HAT - A Portable Power Platform for Every Raspberry Pi”, Pi Supply, 2021. [Online] Available: https://uk.pi-supply.com/products/pijuice-standard

[15] Raspberry Pi Foundation, “Power Supply”, Raspberry Pi Foundation, 2021. [Online] Available:https://www.raspberrypi.org/documentation/hardware/raspberrypi/power/README.md

[16] Adafruit, “Adafruit RFM95W LoRaWAN Radio Transceiver Breakout - 868 or 915 MHz - RadioFruit”, Adafruit, 2021. [Online] Available: https://www.adafruit.com/product/3072

[17] Jiankai.li , “Grove - Gas Sensor (MQ2) User Manual”, Seeed, 2015. [Online] Available: https://docs.particle.io/assets/datasheets/electronsensorkit/MQ-2.pdf

[18] Adafruit, “ADS1115 16-Bit ADC - 4 Channel with Programmable Gain Amplifier”, Adafruit, 2021. [Online] Available: https://www.adafruit.com/product/1085#technical-details

[19] Distrelec Schweiz AG, “PIS-1129 - PiJuice LiPo Battery, 12000mAh, Pi Supply”, Distrelec Schweiz AG, 2021. [Online] Available: https://www.distrelec.biz/en/pijuice-lipo-battery-12000mah-pi-supply-pis-1129/p/30163381

[20] DNK Power Company Limited, “Lithium Polymer Battery Complete Guide”, DNK Power Company Limited, 2021. [Online] Available: https://www.dnkpower.com/lithium-polymer-battery-guide/

[21] D. Kasler, R. Sabalow, “Paradise was ‘well-prepared’ for deadly Camp Fire. But it wasn’t enough, new study finds”, The Sacramento Bee, 2021. [Online] Available: https://www.sacbee.com/news/california/fires/article249017825.html

[22] P. Boghani, “Camp Fire: By the Numbers”, Frontline, PBS, 2019. [Online] Available: https://www.pbs.org/wgbh/frontline/article/camp-fire-by-the-numbers/

[23] Hanwei Electronics Co., Ltd, “Technical Data MQ-2 Gas Sensor”, Mouser Electronics, 2019. [Online] Available: https://www.mouser.com/datasheet/2/321/605-00008-MQ-2-Datasheet-370464.pdf

[24] Last Minute Engineers, “How MQ2 Gas/Smoke Sensor Works? & Interface it with Arduino”, Last Minute Engineers, 2021. [Online] Available: https://lastminuteengineers.com/mq2-gas-senser-arduino-tutorial/

[25] B. Earl, “Python & CircuitPython”, Adafruit, 2021. [Online] Available: https://learn.adafruit.com/adafruit-4-channel-adc-breakouts/python-circuitpython

[26] Raspberry Pi Foundation, “Camera Module”, Raspberry Pi Foundation, 2021. [Online] Available: https://www.raspberrypi.org/documentation/hardware/camera/

[27] Raspberry Pi Foundation, “Camera Module V2”, Raspberry Pi Foundation, 2021. [Online] Available: https://www.raspberrypi.org/products/camera-module-v2/

[28] Arducam Store, “Arducam 15 Pin 1.0mm Pitch to 22 Pin 0.5mm Camera Cable for Raspberry Pi Zero Version 1.3 with 150mm and 73mm (Pack of 2)”, Amazon, 2021. [Online] Available: https://www.amazon.com/dp/B076Q595HJ?psc=1&ref=ppx_yo2_dt_b_product_details

[29] A. Huamán, “Cascade Classifier”, OpenCV, 2021. [Online] Available: https://docs.opencv.org/master/db/d28/tutorial_cascade_classifier.html

[30] S. Puttemans, “Cascade Classifier Training”, OpenCV, 2021. [Online] Available: https://docs.opencv.org/master/dc/d88/tutorial_traincascade.html

[31] NVIDIA Developer, “Jetson AGX Xavier Developer Kit”, NVIDIA Corporation, 2021. [Online] Available: https://developer.nvidia.com/embedded/jetson-agx-xavier-developer-kit

[32] Down To Earth, “Lithium extraction should be sustainable”, downtoearth.org. [Online] Available: https://www.downtoearth.org.in/blog/pollution/lithium-extraction-should-be-sustainable-68988#:~:text=Deposits%20of%20lithium%20are%20located,sensitive%20ecosystems%20in%20the%20world.&text=In%202016%2C%20a%20toxic%20chemical,banks%20of%20the%20Liqi%20River.

[33] S. Blank, K. Mullaney, “How to Build a Startup”, Udacity, Inc., 2021. [Online] Available: https://www.udacity.com/course/how-to-build-a-startup--ep245

Appendices:

Appendix A: Analysis of Senior Project Report

Project Title: Wildfire Early Detection System (WEDS)
Students' Names: Mason McIver, Vincent Liang, JeanReno Racines
Advisor's Name: Professor Andrew Danowitz
Advisors's Initials: _____________ Date:_________________

Summary of Functional Requirements

The Wildfire Early Detection System (WEDS) is a device that monitors forest areas and detects wildfires at their onset. Powered solely on a LiPo battery and supported by solar cells, the device should last for an indefinite period. If a fire occurs within its proximity, the device will use the camera and smoke sensor to confirm the presence of that fire. The camera will take photos and the Raspberry Pi will perform computer vision techniques to determine the presence of a wildfire. With the LoRaWAN communication protocol, multiple WEDS devices can cover vast forest areas, transferring data between each other and to the main control hub.

Primary Constraints

One main constraint of this project is the battery life of the device. Given that users will place these devices in forests, it would be ideal for them to require minimal maintenance. This requires careful consideration of the combined current specifications for the sensors, camera, Raspberry Pi, and transceiver. Another constraint is the physical robustness of these devices. To prevent them from disrupting wildlife, it should be difficult for animals to break into them and should not stand out. Also, the housing needs to withstand drops from tens or hundreds of feet and be weather resistant.

Economic

What economic impacts result?

Human Capital

The WEDS device is a new product in a relatively new market that can stimulate jobs and new methods for preventing and controlling wildfire. As the new industry grows, more engineering, marketing, and sales representatives are needed as requirements improve and develop.

Financial Capital

This product targets the US Forest Service. This new product can save the USFS hundreds of thousands of dollars by alerting first responders earlier allowing them to save money by tackling a smaller wildfire. By detecting forest fires early on, there will be less damage to federal, state, and private property.

Natural Capital

Given that a WEDS device consists mainly of electrical components, most of the materials within it are non-renewable and likely non-recyclable.

When and where do the costs and benefits accrue throughout the project's lifecycle?

Nearly all of the costs of a WEDS device lifecycle come from the manufacturing. This comes from the components that were purchased in order to manufacture the WEDS device. The design process did not accrue any additional costs. During the development of the WEDS device, additional costs were accrued for components that were used in the test design. These components would not be necessary to purchase for the manufacturing of every WEDS device. The only additional costs for operation of a WEDS device are maintenance and operation. Maintenance includes purchasing components that need replacing, paying someone to fix the WEDS device and to transport the device to remote locations. Operation requires an employee to monitor the WEDS Central Hub for any evidence of wildfires.

What inputs does the experiments require? How much does the project cost? Who pays?

The total project cost is $681.32. The Electrical Engineering department will reimburse $598.33, and the rest is paid by the team members. There are some items used in the construction of the WEDS device that were obtained for free and therefore not included in the bill of materials. There are also items purchased that were not used in the final prototype design and therefore are not included in the estimated component cost for manufacturing but are included in the budget in Appendix B.

Hidden costs include the cost of the wood used to build the fires that were used to test, the wood used to build the enclosure which came from an old cabinet obtained for free, and spray paint to paint the enclosure was obtained for free, as well as all of the antennas were obtained for free. In this case hidden costs are defined as the cost of parts used in construction of the WEDS device that were not paid by the WEDS team.

Original estimated cost of component parts

Item	QTY	MFG	Part Number	Cost
Protoboard	1	---	---	
Female Header: 2x20-Pin, straight	1	Polulu	---	$1.25
Voltage regulator	3	ON Semiconductor	LM317MTG	$0.57
Thermocouple digitizer	1	Maxim Integrated	MAX31855KASA+T	$5.17
Comparator	1	Texas Instruments	LM393P	$0.35
Case, Housing, PETG (3D print filament)	1	---	---	$2.00
Computer	1	Raspberry Pi Foundation	---	$10.00
Antenna, 915 Mhz	1	Taoglas	FW.86.B.SMA.m	$9.96
MIPI CSI Camera Module	1	Raspberry Pi Foundation	---	$29.95
LoRaWAN Transceiver - 868 MHz	1	Adafruit Industries	RFM95W	$39.90
K-Type thermocouple (0-1000c)	1	T-PRO	VF1000	$16.99
Smoke detector (ionization)	1	Kidde Safety	21026056	$9.00
IR sensor	1	Adafruit Industries	MLX90640	$59.95
Solar panel 10W 2.6A	5			
3400mAh 4.9A battery	3	Panasonic	NCR18650B	$4.99
				$190.08

Actual final cost of component parts

Item	QTY	MFG	Unit Cost	Total Cost
Raspberry Pi Zero W	1	Raspberry Pi Foundation	$10.00	$10.00
Header Pins	1	OCR	$7.99	$7.99
LoRaWAN Module	1	AdaFruit	$19.95	$19.95
MQ2 Gas Sensor	1	Youling	$6.79	$6.79
16-Bit ADC	1	MakerFocus	$4.50	$4.50
Camera Module V2		PiShop	$29.95	$29.95
PiJuice Module	1	PiSupply	$79.95	$79.95
5V, 2.5W Solar Panel	10	AllPowers	$6.50	$65.00
8000 mAh LiPo Battery	1	YDL	$21.99	$21.99
GPIO Breakout Board	1	Pastall	$7.99	$7.99
Protoboard 7x9cm	1	Deyue	$0.30	$0.30
2 Pole Toggle Switch	1	Gardner Blender	$4.49	$4.49
On/Off Toggle Switch	1	Gardner Blender	$4.49	$4.49
Coaxial SMT connector	1	AdaFruit	$0.75	$0.75
micro-USB female connector	1	HiLetGo	$0.45	$0.45
USB LED Strip Lights	1	Renohef	$14.99	$14.99
2” Bulk Foam	1	---	$5.97	$5.97
Flat Bar Aluminum 36x1x1/8	1	Home Depot	$7.98	$7.98
Hinge Narrow 1" Zinc	1	Home Depot	$2.89	$2.89
Phil Zince Screws	2x12	HomeDepot	$1.18	$2.36
Velcro Squares BLK	1	3M	$2.93	$2.92
				$301.70

*Components are only included if purchased and included in the final prototype design
** See Appendix C for a total of all expenses throughout the duration of the project

Additional equipment costs

The only additional equipment costs are the cost of the wood used to build the fires as well as the cost of the Analog Discovery 2 oscilloscope. The wood was obtained at no cost and the oscilloscope used for testing was borrowed therefore are not presently included.

How much does the project earn? Who profits?

The earnings of the WEDS project is heavily dependent on the adoption of the system by the U.S Forest Service. WEDS currently does not earn any profit. WEDS will operate as a government contractor that produces WEDS devices that are sold to government agencies to earn a profit.

Timing

When do products emerge?

New WEDS devices will be released every year before the fire season begins.

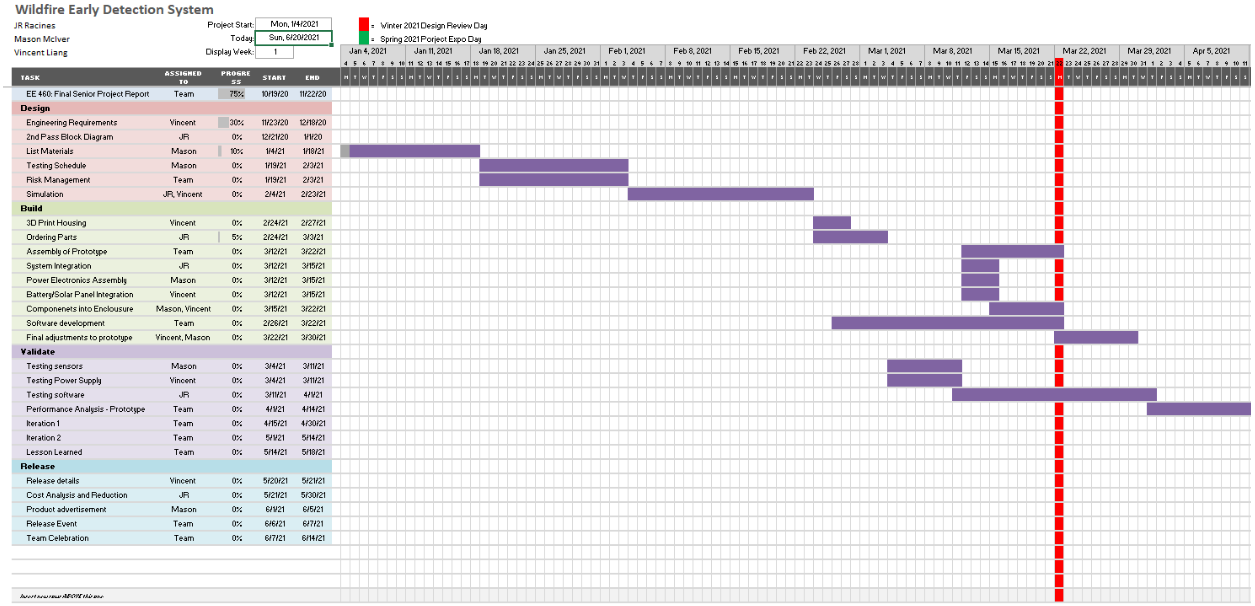
How long do products exist?

WEDS devices will have the ability to be serviced indefinitely. Software updates will be provided frequently. Previous versions of WEDS devices will never be phased out but customers will have the ability to upgrade the hardware at any point.

What maintenance or operation costs exist?

Normal operation of WEDS devices is free! WEDS plans to train technicians that will be able to service WEDS devices. Service charges will depend on the nature of the problem that is being serviced. In the future a WEDS protection plan will be available to be purchased for $10 per month per device (price subject to change).

Original estimated development time



Wildfire Early Detection System

JR Racines
Mason McIver
Vincent Liang

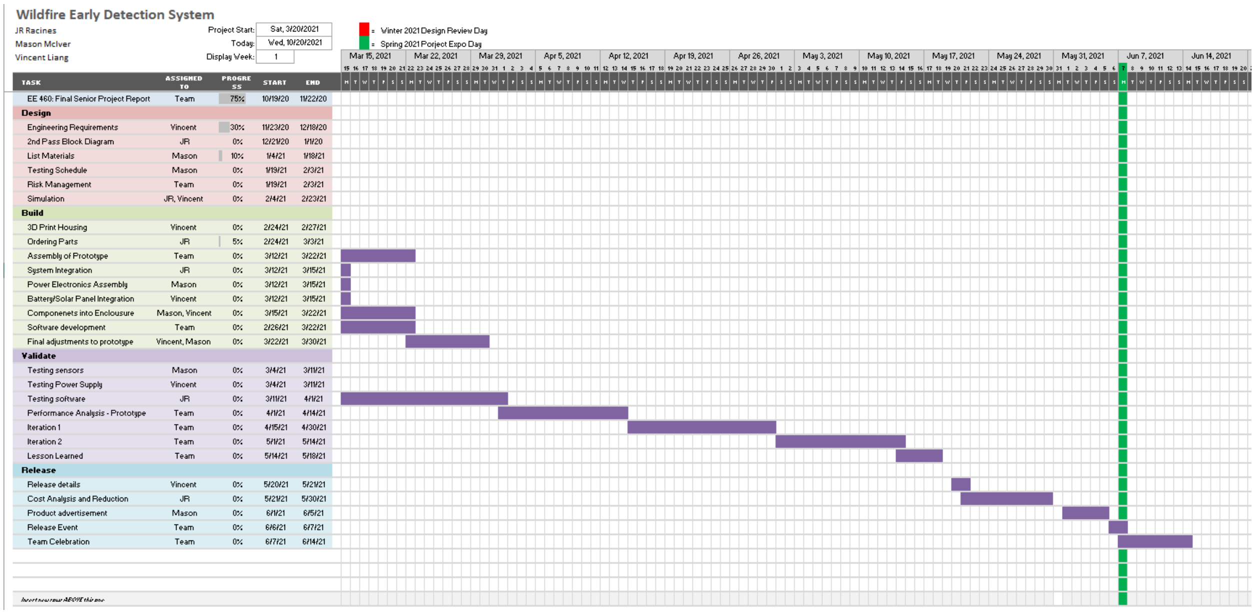
Project Start: Mon, 1/4/2021
Today: Sun, 6/20/2021
Display Week: 1

🟥 = Winter 2021 Design Review Day
🟩 = Spring 2021 Porject Expo Day

TASK	ASSIGNED TO	PROGRESS	START	END
EE 460: Final Senior Project Report	Team	75%	10/19/20	11/22/20
Design				
Engineering Requirements	Vincent	30%	11/23/20	12/18/20
2nd Pass Block Diagram	JR	0%	12/21/20	1/1/20
List Materials	Mason	10%	1/4/21	1/18/21
Testing Schedule	Mason	0%	1/19/21	2/3/21
Risk Management	Team	0%	1/19/21	2/3/21
Simulation	JR, Vincent	0%	2/4/21	2/23/21
Build				
3D Print Housing	Vincent	0%	2/24/21	2/27/21
Ordering Parts	JR	5%	2/24/21	3/3/21
Assembly of Prototype	Team	0%	3/12/21	3/22/21
System Integration	JR	0%	3/12/21	3/15/21
Power Electronics Assembly	Mason	0%	3/12/21	3/15/21
Battery/Solar Panel Integration	Vincent	0%	3/12/21	3/15/21
Componenets into Enclousure	Mason, Vincent	0%	3/15/21	3/22/21
Software development	Team	0%	2/26/21	3/22/21
Final adjustments to prototype	Vincent, Mason	0%	3/22/21	3/30/21
Validate				
Testing sensors	Mason	0%	3/4/21	3/11/21
Testing Power Supply	Vincent	0%	3/4/21	3/11/21
Testing software	JR	0%	3/11/21	4/1/21
Performance Analysis - Prototype	Team	0%	4/1/21	4/14/21
Iteration 1	Team	0%	4/15/21	4/30/21
Iteration 2	Team	0%	5/1/21	5/14/21
Lesson Learned	Team	0%	5/14/21	5/18/21
Release				
Release details	Vincent	0%	5/20/21	5/21/21
Cost Analysis and Reduction	JR	0%	5/21/21	5/30/21
Product advertisement	Mason	0%	6/1/21	6/5/21
Release Event	Team	0%	6/6/21	6/7/21
Team Celebration	Team	0%	6/7/21	6/14/21

Insert new rows ABOVE this one

Figure 52: Previous Gantt Chart part 1



Wildfire Early Detection System

JR Racines
Mason McIver
Vincent Liang

Project Start: Sat, 3/20/2021
Today: Wed, 10/20/2021
Display Week: 1

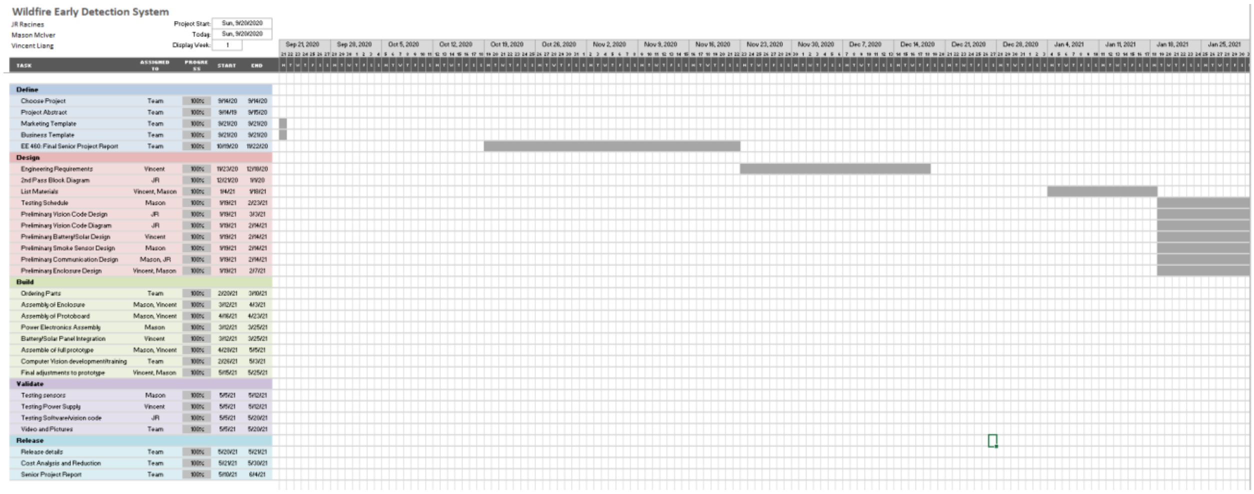
🟥 = Winter 2021 Design Review Day
🟩 = Spring 2021 Porject Expo Day

TASK	ASSIGNED TO	PROGRESS	START	END
EE 460: Final Senior Project Report	Team	75%	10/19/20	11/22/20
Design				
Engineering Requirements	Vincent	30%	11/23/20	12/18/20
2nd Pass Block Diagram	JR	0%	12/21/20	1/1/20
List Materials	Mason	10%	1/4/21	1/18/21
Testing Schedule	Mason	0%	1/19/21	2/3/21
Risk Management	Team	0%	1/19/21	2/3/21
Simulation	JR, Vincent	0%	2/4/21	2/23/21
Build				
3D Print Housing	Vincent	0%	2/24/21	2/27/21
Ordering Parts	JR	5%	2/24/21	3/3/21
Assembly of Prototype	Team	0%	3/12/21	3/22/21
System Integration	JR	0%	3/12/21	3/15/21
Power Electronics Assembly	Mason	0%	3/12/21	3/15/21
Battery/Solar Panel Integration	Vincent	0%	3/12/21	3/15/21
Componenets into Enclousure	Mason, Vincent	0%	3/15/21	3/22/21
Software development	Team	0%	2/26/21	3/22/21
Final adjustments to prototype	Vincent, Mason	0%	3/22/21	3/30/21
Validate				
Testing sensors	Mason	0%	3/4/21	3/11/21
Testing Power Supply	Vincent	0%	3/4/21	3/11/21
Testing software	JR	0%	3/11/21	4/1/21
Performance Analysis - Prototype	Team	0%	4/1/21	4/14/21
Iteration 1	Team	0%	4/15/21	4/30/21
Iteration 2	Team	0%	5/1/21	5/14/21
Lesson Learned	Team	0%	5/14/21	5/18/21
Release				
Release details	Vincent	0%	5/20/21	5/21/21
Cost Analysis and Reduction	JR	0%	5/21/21	5/30/21
Product advertisement	Mason	0%	6/1/21	6/5/21
Release Event	Team	0%	6/6/21	6/7/21
Team Celebration	Team	0%	6/7/21	6/14/21

Insert new rows ABOVE this one

Figure 53: Previous Gantt Chart part 2

<u>Actual development time</u>



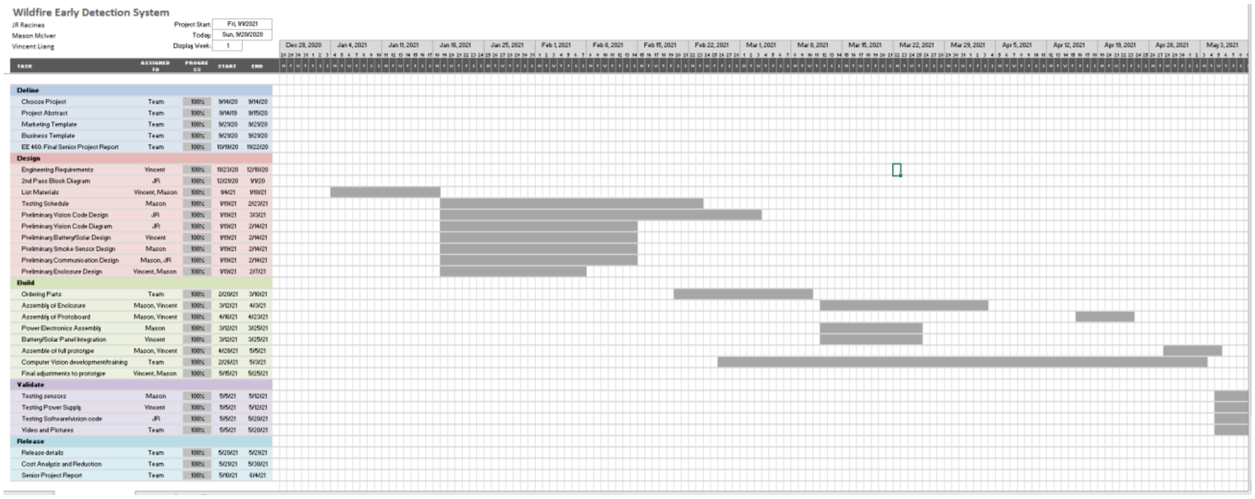
Wildfire Early Detection System

JR Racines
Mason McIver
Vincent Liang

Project Start: Sun, 9/20/2020
Today: Sun, 9/20/2020
Display Week: 1

TASK	ASSIGNED TO	PROGRESS	START	END
Define				
Choose Project	Team	100%	9/14/20	9/14/20
Project Abstract	Team	100%	9/14/19	9/15/20
Marketing Template	Team	100%	9/21/20	9/21/20
Business Template	Team	100%	9/21/20	9/21/20
EE 460: Final Senior Project Report	Team	100%	10/19/20	11/22/20
Design				
Engineering Requirements	Vincent	100%	11/23/20	12/18/20
2nd Pass Block Diagram	JR	100%	12/21/20	1/1/20
List Materials	Vincent, Mason	100%	1/4/21	1/18/21
Testing Schedule	Mason	100%	1/19/21	2/23/21
Preliminary Vision Code Design	JR	100%	1/19/21	3/3/21
Preliminary Vision Code Diagram	JR	100%	1/19/21	2/14/21
Preliminary Battery/Solar Design	Vincent	100%	1/19/21	2/14/21
Preliminary Smoke Sensor Design	Mason	100%	1/19/21	2/14/21
Preliminary Communication Design	Mason, JR	100%	1/19/21	2/14/21
Preliminary Enclosure Design	Vincent, Mason	100%	1/19/21	2/7/21
Build				
Ordering Parts	Team	100%	2/20/21	3/10/21
Assembly of Enclosure	Mason, Vincent	100%	3/12/21	4/3/21
Assembly of Protoboard	Mason, Vincent	100%	4/16/21	4/23/21
Power Electronics Assembly	Mason	100%	3/12/21	3/25/21
Battery/Solar Panel Integration	Vincent	100%	3/12/21	3/25/21
Assemble of full prototype	Mason, Vincent	100%	4/28/21	5/5/21
Computer Vision development/training	Team	100%	2/26/21	5/3/21
Final adjustments to prototype	Vincent, Mason	100%	5/15/21	5/25/21
Validate				
Testing sensors	Mason	100%	5/5/21	5/12/21
Testing Power Supply	Vincent	100%	5/5/21	5/12/21
Testing Software/vision code	JR	100%	5/5/21	5/20/21
Video and Pictures	Team	100%	5/5/21	5/20/21
Release				
Release details	Team	100%	5/20/21	5/21/21
Cost Analysis and Reduction	Team	100%	5/21/21	5/30/21
Senior Project Report	Team	100%	5/10/21	6/4/21

Figure 54: Actual Gantt Chart part 1



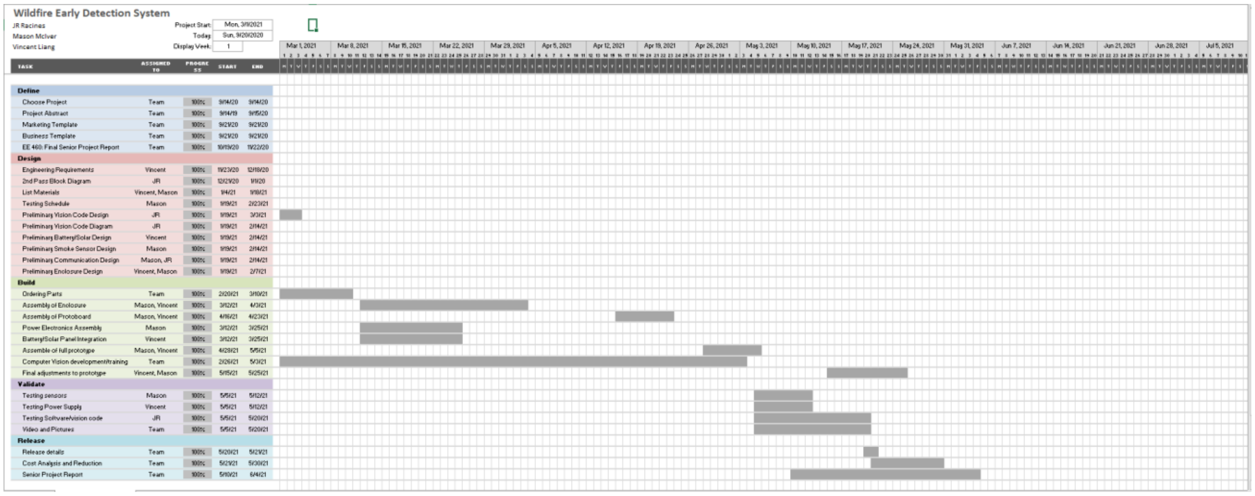
Wildfire Early Detection System

JR Racines
Mason McIver
Vincent Liang

Project Start: Fri, 1/1/2021
Today: Sun, 9/20/2020
Display Week: 1

TASK	ASSIGNED TO	PROGRESS	START	END
Define				
Choose Project	Team	100%	9/14/20	9/14/20
Project Abstract	Team	100%	9/14/19	9/15/20
Marketing Template	Team	100%	9/21/20	9/21/20
Business Template	Team	100%	9/21/20	9/21/20
EE 460: Final Senior Project Report	Team	100%	10/19/20	11/22/20
Design				
Engineering Requirements	Vincent	100%	11/23/20	12/18/20
2nd Pass Block Diagram	JR	100%	12/21/20	1/1/20
List Materials	Vincent, Mason	100%	1/4/21	1/18/21
Testing Schedule	Mason	100%	1/19/21	2/23/21
Preliminary Vision Code Design	JR	100%	1/19/21	3/3/21
Preliminary Vision Code Diagram	JR	100%	1/19/21	2/14/21
Preliminary Battery/Solar Design	Vincent	100%	1/19/21	2/14/21
Preliminary Smoke Sensor Design	Mason	100%	1/19/21	2/14/21
Preliminary Communication Design	Mason, JR	100%	1/19/21	2/14/21
Preliminary Enclosure Design	Vincent, Mason	100%	1/19/21	2/7/21
Build				
Ordering Parts	Team	100%	2/20/21	3/10/21
Assembly of Enclosure	Mason, Vincent	100%	3/12/21	4/3/21
Assembly of Protoboard	Mason, Vincent	100%	4/16/21	4/23/21
Power Electronics Assembly	Mason	100%	3/12/21	3/25/21
Battery/Solar Panel Integration	Vincent	100%	3/12/21	3/25/21
Assemble of full prototype	Mason, Vincent	100%	4/28/21	5/5/21
Computer Vision development/training	Team	100%	2/26/21	5/3/21
Final adjustments to prototype	Vincent, Mason	100%	5/15/21	5/25/21
Validate				
Testing sensors	Mason	100%	5/5/21	5/12/21
Testing Power Supply	Vincent	100%	5/5/21	5/12/21
Testing Software/vision code	JR	100%	5/5/21	5/20/21
Video and Pictures	Team	100%	5/5/21	5/20/21
Release				
Release details	Team	100%	5/20/21	5/21/21
Cost Analysis and Reduction	Team	100%	5/21/21	5/30/21
Senior Project Report	Team	100%	5/10/21	6/4/21

Figure 55: Actual Gantt Chart part 2



Wildfire Early Detection System

JR Racines
Mason McIver
Vincent Liang

Project Start: Mon, 3/1/2021
Today: Sun, 9/20/2020
Display Week: 1

TASK	ASSIGNED TO	PROGRESS	START	END
Define				
Choose Project	Team	100%	9/14/20	9/14/20
Project Abstract	Team	100%	9/14/19	9/15/20
Marketing Template	Team	100%	9/21/20	9/21/20
Business Template	Team	100%	9/21/20	9/21/20
EE 460: Final Senior Project Report	Team	100%	10/19/20	11/22/20
Design				
Engineering Requirements	Vincent	100%	11/23/20	12/18/20
2nd Pass Block Diagram	JR	100%	12/21/20	1/1/20
List Materials	Vincent, Mason	100%	1/4/21	1/18/21
Testing Schedule	Mason	100%	1/19/21	2/23/21
Preliminary Vision Code Design	JR	100%	1/19/21	3/3/21
Preliminary Vision Code Diagram	JR	100%	1/19/21	2/14/21
Preliminary Battery/Solar Design	Vincent	100%	1/19/21	2/14/21
Preliminary Smoke Sensor Design	Mason	100%	1/19/21	2/14/21
Preliminary Communication Design	Mason, JR	100%	1/19/21	2/14/21
Preliminary Enclosure Design	Vincent, Mason	100%	1/19/21	2/7/21
Build				
Ordering Parts	Team	100%	2/20/21	3/10/21
Assembly of Enclosure	Mason, Vincent	100%	3/12/21	4/3/21
Assembly of Protoboard	Mason, Vincent	100%	4/16/21	4/23/21
Power Electronics Assembly	Mason	100%	3/12/21	3/25/21
Battery/Solar Panel Integration	Vincent	100%	3/12/21	3/25/21
Assemble of full prototype	Mason, Vincent	100%	4/28/21	5/5/21
Computer Vision development/training	Team	100%	2/26/21	5/3/21
Final adjustments to prototype	Vincent, Mason	100%	5/15/21	5/25/21
Validate				
Testing sensors	Mason	100%	5/5/21	5/12/21
Testing Power Supply	Vincent	100%	5/5/21	5/12/21
Testing Software/vision code	JR	100%	5/5/21	5/20/21
Video and Pictures	Team	100%	5/5/21	5/20/21
Release				
Release details	Team	100%	5/20/21	5/21/21
Cost Analysis and Reduction	Team	100%	5/21/21	5/30/21
Senior Project Report	Team	100%	5/10/21	6/4/21

Figure 56: Actual Gantt Chart part 3

What happens after the project ends?

After each cycle of the WEDS device is complete the device will be available for purchase for government agencies as well as private consumers.

If Manufactured on a Large-Scale Basis

a.	Estimated number of devices per year:	25k - 50k
b.	Estimated manufacturing cost for each device:	$ 301.70
c.	Estimated purchase price for each unit:	$ 500.00
d.	Estimated profit per year:	$ 12,500,000
e.	Estimated cost for user to operate devices:	$ 50 per month per device

Environmental Impact

Environmental impacts associated with manufacturing or use of the product and where they occur.

Since users will place WEDS devices throughout forests, it will have a presence in the surrounding wildlife. The housing prevents animals from easily breaking into it and it does not easily break upon falling from a great height. It must also withstand weather conditions such as rain, snow, high winds, and more. These devices should also not emit any harmful electromagnetic radiation as the communication protocol follows FCC guidelines. All these design choices are made with the hopes that WEDS devices will have as little impact on the ecosystem as possible.

Natural resources and ecosystem services that the product uses directly and indirectly.

The components within WEDS devices are made of silicon along with metals including copper, aluminum, silver, and more. The PCBs in these devices are made of FR4. These materials are sourced via mining, which may affect ecosystems. To maintain operation, WEDS devices do not need to stay wired to an electrical source since they are powered via batteries and solar cells.

Natural resources and ecosystem services that the product improves or harms.

The fundamental materials that compose the electronics within WEDS devices have impacts on the environment where they are mined. Lithium Polymer (LiPo) batteries are made from lithium, a reactive alkali metal that powers many phones, laptops, and other electronics today. However, obtaining this material can be harmful to ecosystems. An example of this is the toxic chemical leak from the Ganzizhou Rongda Lithium mine in Upper Tibet that killed many fish and other wildlife [32].

How the project impacts other species.

WEDS devices are planned to be mounted in remote fire prone locations. These locations often happen to be home to various types of wildlife. The WEDS device is completely passive and will not negatively interfere with wildlife. Perch guards can be purchased to keep birds from using the WEDS device mounting system as a perch and to prevent nest building. The lid and enclosure will be sealed tight so little animals cannot build nests and use the device as a home.

Manufacturability

Describe any issues or challenges associated with manufacturing.

Manufacturing of a WEDS device consists of two portions: the housing and the PCB. The housing is a simple box design that can be manufactured by humans. The housing also needs weather sealing, which will be added manually. The current prototype design for the WEDS device uses a protoboard for the component connections but ideally, the commercially available WEDS devices will be built with a PCB. The bare PCBs require a third-party for manufacturing. Space is not an issue for the WEDS device as the inside of the enclosure is mostly empty space filled with foam to provide protection and durability. The electrical components will be through hole mounted and will be manually soldered onto the PCB. The sensors can also be manually positioned inside the housing.

Sustainability

Issues or challenges associated with maintaining the completed device, or system.

The completed WEDS device’s lifecycle must be as long as possible. Frequent physical maintenance of these devices would be tedious given that they are placed throughout remote forests. Thus, the LiPo battery that is supported by the solar cells must power the device for an indefinite period. It must also be physically robust to withstand various weather conditions and prevent wildlife from tampering with the electronics. These design requirements prevent the device from breaking easily and ending up in landfills. A longer lasting WEDS device is more sustainable and effective.

How the product impacts the sustainable use of resources.

Since the device operates from batteries and does not require power from the grid, it must be energy efficient. We will also source as many components as possible from recyclable materials, which include electronics, PCBs, and housing.

Describe any upgrades that would improve the design of the project.

Upgrades that would improve the design of the project include, an enclosure with less of a footprint, addition of a temperature sensor and IR sensor, improvement of the smoke detection software, improvement of the computer vision fire detection software, and better battery and solar panel efficiency.

Describe any issues of challenges associated with upgrading the design.

Software updates are easier to distribute to customers than hardware upgrades. Software updates will require an internet connection so the WEDS device that is being upgraded must be brought to a location with a secure internet connection for the update to be installed. Adding the temperature sensor will require a redesign of the protoboard layout as well as an additional mount in the enclosure. It will also require updates to the software to transmit temperature data. Improvement of the enclosure footprint will consist of a 3D-printed design for the solar panel mounts and a significantly smaller enclosure for the electrical components. Improvement in energy management would consist of adding a higher energy density battery with a more efficient solar panel. This is challenging because a battery management system and current and voltage regulation is needed to ensure the battery is charged properly and energy is used efficiently. This means an advanced design in power electronics is needed. All hardware upgrades will be in the form of future versions of the WEDS device.

Ethical

Describe ethical implications relating to the design, manufacture, use, or misuse of the project.

WEDS devices aim to reduce the chance that wildfires become uncontrollable in the hopes of protecting any wildlife that the fire may harm and protecting private, state and federal property. This task concerns the safety, health, and welfare of the public, which aligns with the first IEEE code of ethics. The detection of a fire via the computer vision cascade model does not guarantee that there is a fire within the frame of the camera. Furthermore, if no fire is being detected via the computer vision cascade model, this does not guarantee that there is not a fire within the frame of the camera.

The main ethical issue with WEDS devices is how they affect ecosystems. Ideally, the device will be extremely robust so animals cannot easily break into the device, and it can withstand large forces such as falling from a great height. It cannot have anything in its exterior that animals can break off and possibly digest. Users must also place these devices in elevated positions, out of the way of wildlife. Even with these design considerations, it may not protect against intentional vandalism.

Another ethical issue is what happens to WED devices after the end of their lifetime or if they are replaced with an improved version. If they are not recyclable, some of the electrical components may end up as electronic waste in landfills.

Health and Safety

Describe any health and safety concerns associated with the design, manufacture or use of the project.

WEDS devices do not pose any direct health or safety risks to humans as they require very minimal human interaction with the device. They can put humans in danger if they fail to identify wildfires quickly or fail to detect them at all. This depends on not only the quality of its sensors but also the quality of the software code on the Raspberry Pi. Thus, WEDS devices pose indirect health and safety risks which mainly stem not from user error but design error.

Social and Political

Describe social and political issues associated with design, manufacture, and use. Who does the project impact?

The social and political issues arise with the device's reliability and adoption. If these devices fail to detect a wildfire, the fire may spread rapidly, cause property damage, and endanger human lives. If this occurs, people’s trust in these devices will be questioned. Another problem arises when users need to replace these devices or have maintenance done on them. Since the US Forest Service is a government agency, political challenges can arise causing them to not have enough funding or the manpower to replace and maintain them, thus decreasing the effectiveness and reliability of the WEDS device. For now, our target is the Pacific Northwest and Southwest; however, this product can expand to other parts of the United States and even other countries that have forest fires. The goal is to not only start a new market but to provide a solution that could potentially save lives, money, and property.

Who are the direct and indirect stakeholders? How does the project benefit or harm various stakeholders?

The stakeholders would be the US Forest Service, private residents in fire prone areas, investors, community members, and manufacturing and component partners. The project would benefit the various stakeholders because this new device would be expanding this market, bringing business to the manufacturing and electronic component partners. The US Forest Service and private residents in the area would benefit since their assets would be protected and they would be notified early on if a fire was detected, giving them time to either evacuate or respond to the emergency.

To what extent do stakeholders benefit equally? Pay equally? Does the project create any inequities?

The stakeholders wouldn’t necessarily benefit equally since the device doesn’t stop wildfires but detects them instead. This means that private residents in fire prone areas and the US Forest Service could still lose homes, land, and assets from the wildfire. They would lose less since they were notified early, but it is not guaranteed that the fire will not cause any destruction.

Development

Describe any new tools or techniques, used for either development or analysis that you learned independently during the course of your project. Include a literature search.

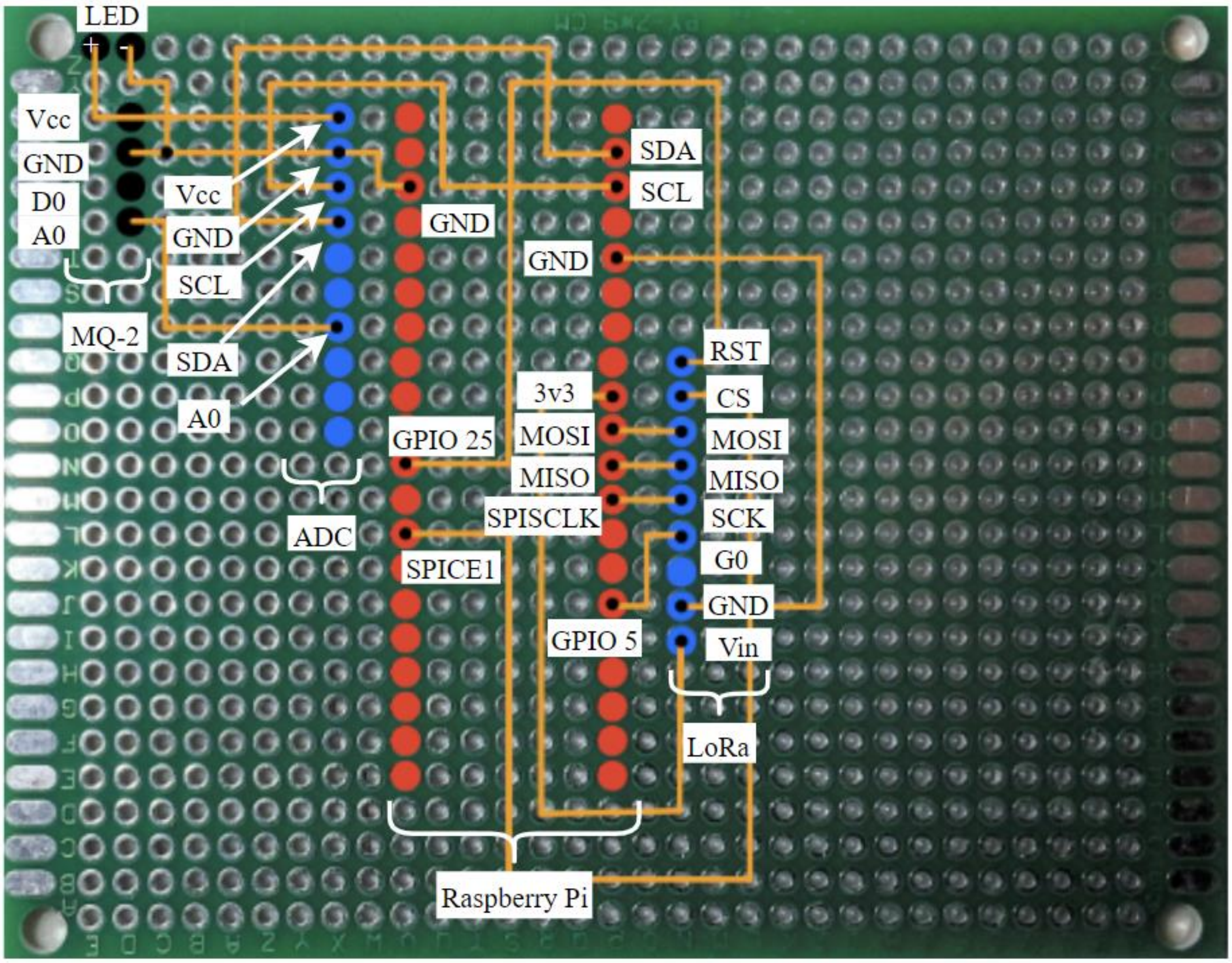
During the development of this project the design process discussed in the Udacity course *How to Build a Startup* by Steven Blank was utilized [33]. By understanding our customer and their needs, a product was able to be designed such that a problem was addressed, and a solution was presented. The planning process was extensive to ensure a design was constructed carefully. During our planning process a gantt chart was used to ensure our goals were reached at a certain time which was part of the Senior Project Preparation class with Professor Murray. When developing the product, research was very extensive. After research, the design was created and tested. When problems were encountered, we went back to the research phase to understand what was happening which then allowed us to make another iteration of the design. By doing this, we were able to figure out what worked and what didn't work.

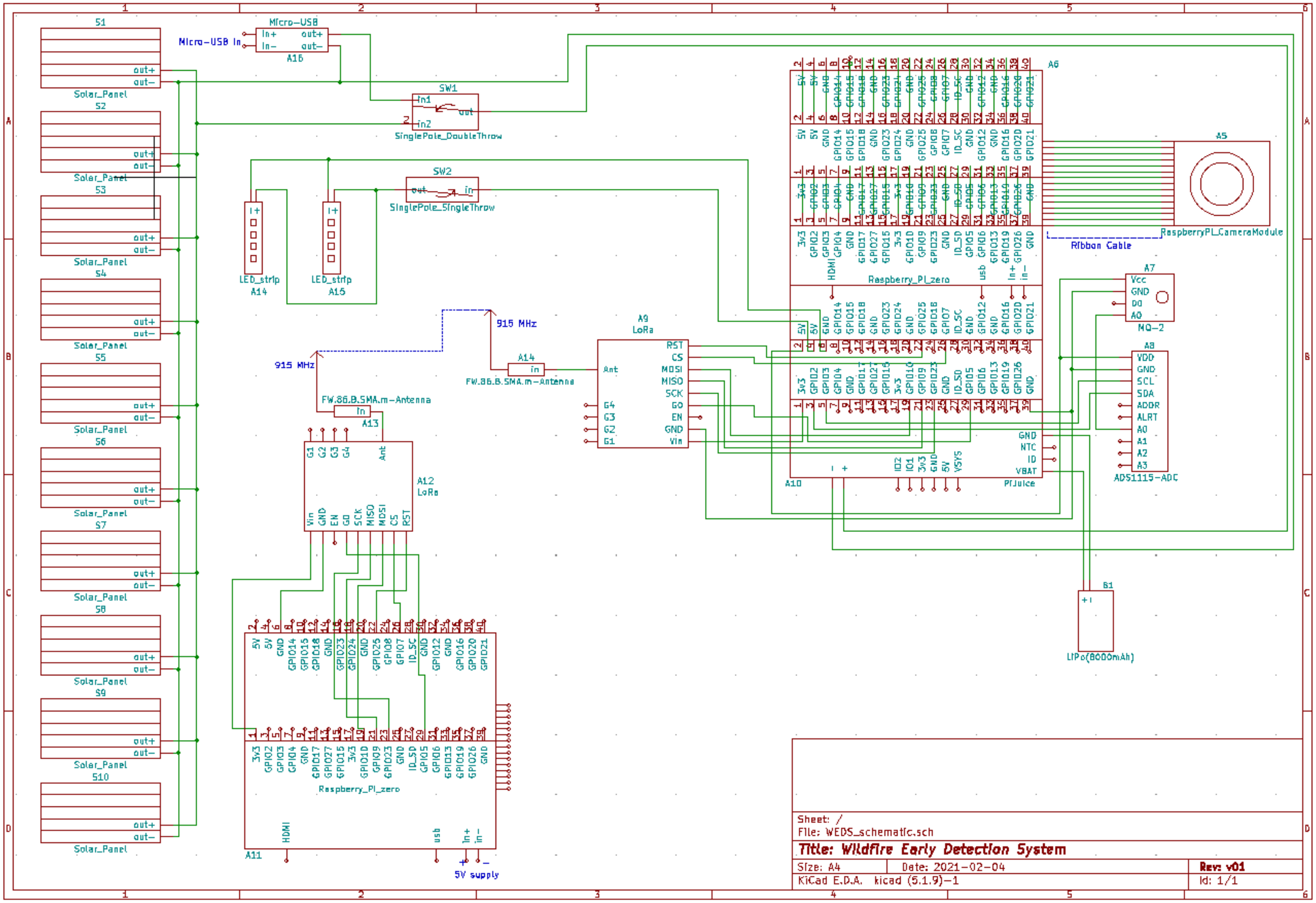
One of the challenging aspects in making the final product was the testing phase. Even though the design should theoretically work, in practice it can be very different since there are other variables that may have been overlooked. Another important aspect in this phase was testing the design in multiple conditions. This was important since the device would normally be in a very dynamic environment where conditions such as temperature, sunlight, and weather could affect the operation.

Appendix B: Bill of Materials (BOM)

https://docs.google.com/spreadsheets/d/1cB02kqmTzt25Ek7mQp59SgnUpvNuWkgK2id-0ahcTu0/edit?usp=sharing

Appendix C: Protoboard Wiring Schematic





Micro-USB In
Micro-USB
A16
SW1
SinglePole_DoubleThrow
SW2
SinglePole_SingleThrow
LED_strip
A14
LED_strip
A15
S1
S2
S3
S4
S5
S6
S7
S8
S9
S10
Solar_Panel
915 MHz
FW.86.B.SMA.m-Antenna
A13
A14
A9
LoRa
A12
LoRa
A11
Raspberry_Pi_zero
5V supply
A6
Raspberry_Pi_zero
A5
RaspberryPi_CameraModule
Ribbon Cable
A7
MQ-2
A8
ADS1115-ADC
A10
PiJuice
B1
LiPo(8000mAh)
Sheet: /
File: WEDS_schematic.sch
Title: Wildfire Early Detection System
Size: A4
Date: 2021-02-04
Rev: v01
KiCad E.D.A. kicad (5.1.9)-1
Id: 1/1

Appendix D: Github Repository

Below is a link to the project’s GitHub repository. Our main computer vision code is in the directory fireDetection > haar2.

https://github.com/jracines/CalPolyWEDS.git

Appendix E: Video Link

Below is a link to a demonstration of the WEDS device. This demo includes the detection of fires via the camera module and smoke sensor.

https://youtu.be/jx-Vs6iw3O8