FOREST FIRE DETECTION SYSTEM

Quad Core Crew

Version 1.0

1/19/2023

Matthew Wilson, Edwin Hernandez, Luis Guevara, Lluviana Vasquez

teamquadcorecrew@groups.com

Implementation Plan Document

Version 1.0

Contents

A. Introduction

1. Purpose

The “Wildfire, Forest Fire, and Detection System” or “Wildfire” is a project that will use a combination of existing “on the shelf” technology that will detect fires, forest fires, and wildfires. It will sample the air for both smoke with both ionization and/or photoelectric smoke detector technologies. Also, it will use a temperature sensor in an outdoor enclosure that will be distributed across a grid of forest, woodlands, prairie, and fire prone areas to detect the presence of combusting ions and/or smoke that will signify that a fire is in progress. This project will use cellemetry – to a satellite or cellular/wireless network - to notify Command, Control, and Dispatch of a potential fire that could become something even bigger. Fires that are small can be extinguished by a small crew that is dispatched or by water drops by an aircraft – this will prevent the forest from burning. This project will be able to detect and locate the direction and speed of a burn. The idea of this project is to mount it onto a steel pole, tree, or anywhere with altitude so we can get a better reading of where a fire could potentially start. It will be a rechargeable station where it relies on solar energy to generate enough power to run all its components such as: mini camera, microcontroller and alarm panel, battery charger, heat sensor, telemetry/cellemetry transmitter, and the ionization and photoelectric circuit boards. The ideal plan is to build three stations – triangulation – for a better reading on location, so it could alarm the Command Control.

2. Functional Requirements

ID	Functional Requirements	Team Member Responsible	Effort (in %)	Verification
FR1	The “Wildfire” system should be able to detect combustion ions using ionization technology.	Lluviana Vasquez	25%	Demonstration
FR2	The “Wildfire” system should be able to detect combustion ions using photoelectric technology.	Luis Guevara	25%	Demonstration
FR3	The “Wildfire” system should be able to send an alert to the cloud when the outside temperature increases to a maximum of 58 degrees Celsius.	Matthew Wilson	25%	Demonstration
FR4	The “Wildfire” system should allow user to observe and query the sensor readings of	Edwin Hernandez	30%	Demonstration

	each unit stored in the cloud database using a website.			
FR5	The “Wildfire” system should be able to wirelessly transmit the location of the wildfire, which is found using triangulation and GPS.	Luis Guevara	25%	Demonstration
FR6	The “Wildfire” system should be able to detect the direction and speed of the wind.	Matthew Wilson	10%	Demonstration
FR7	The "Wildfire" system should be able to query and display the values of all the sensors' readings to the user's computer for in the field debugging.	Luis Guevara	20%	Demonstration
FR8	The “Wildfire” system should be able to display on the website if there is a reading outside a certain threshold indicating an issue with sensors.	Edwin Hernandez	25%	Demonstration
FR9	The “Wildfire” system should have day and night vision mini camera that sends a video feed to the cloud database when a fire is detected.	Lluviana Vasquez	25%	Demonstration
FR10	The “Wildfire” system should send the sensor readings of each unit to a cloud database using cellemetry.	Matthew Wilson	30%	Test
FR11	The “Wildfire” system should be able to send push notifications to cell phones over a cellular network that has their number registered with the website.	Edwin Hernandez	25%	Demonstration
FR12	The enclosure should be able to keep the temperature of	Matthew Wilson	15%	Demonstration

	the internal components less than 90 degrees Celsius using active cooling.			

3. Non-Functional Requirements

ID	Non-Functional Requirements	Team Member Responsible	Effort (in %)	Verification
NFR1	The "Wildfire" system should be in a plastic weather resistant enclosure that meets the IP67 standard.	Lluviana Vasquez	25%	Test
NFR2	The "Wildfire" system should be in an insect proof enclosure.	Lluviana Vasquez	15%	Test
NFR3	The "Wildfire" system should be charged by solar panels and should work for 48 hours when there is no sunlight.	Luis Guevara	15%	Test
NFR4	Enclosure should be 20 cm x 15 cm x 15 cm.	Lluviana Vasquez	10%	Inspection
NFR5	Device should weigh no more than 1 Kg.	Matthew Wilson	10%	Inspection
NFR6	Device enclosure should be green.	Edwin Hernandez	5%	Inspection
NFR7	Should be able to indicate if the device is on/off.	Edwin Hernandez	5%	Demonstration
NFR8	The components for the "Wildfire" system should not cost more than $300.	Matthew Wilson	10%	Analysis
NFR9	A PCB for the "Wildfire" system will be designed so all its sensors and components are on a compact and organized circuit board.	Luis Guevara	15%	Test
NFR10	The website designed for the "Wildfire" system should be able to open in any portable device.	Edwin Hernandez	10%	Demonstration

4. Specifications

a. Functional Requirements Specifications

ID	Functional Requirement Specification	Team Member Responsible
FRS1	An ionization type sensor interfaced to the microcontroller will be used to detect combustion ions. The sensor will be placed in an area inside the enclosure where the sensor can detect combustion ions. When there is a combustion, the microcontroller will use the signal from the ionization sensor to then alert Command Control of the presence of combustion ions.	Lluviana Vasquez
FRS2	A photoelectric fire detection sensor interfaced to the microcontroller will be used to detect combustion particles. The sensor will be placed in an area inside the enclosure where the sensor will be exposed to combustion particles. When there is a combustion, the microcontroller will use the signal from the photoelectric sensor to then alert Command Control of the presence of combustion particles.	Luis Guevara
FRS3	The alert threshold is set at 58 degrees Celsius because that temperature is barely higher than most animals can survive. This threshold will allow the system to be sensitive to irregular temperatures while limiting false positives. This can be accomplished in the software function controlling the temperature sensor by making an if/else statement: if the temperature is above 58 degrees Celsius, alert command control; else, monitor the surroundings.	Matthew Wilson
FRS4	Command control will need a mobile application to view and receive data analytics that can showcase potential fire cases for each "Wildfire" system unit. Command control will not have access to edit the data, but only view and receive. Command control will be notified if a specific "Wildfire" system unit notices a potential fire.	Edwin Hernandez
FRS5	A GSM module for the microcontroller will allow connectivity to cellular networks, allowing for communication with Command Control. Triangulation using cell towers will allow for the location of the "Wildfire" system unit to be returned. Triangulation of the fire's location will be done by triangulating the coordinates of the "Wildfire" units when they detect combustion. The microcontroller will do the triangulation to then find where the source of the fire was. The GPS module will also triangulate the location of the unit as redundancy.	Luis Guevara

FRS6	Wind vanes measure wind direction and anemometers measure wind speed. These sensors can be bought separately or combined into one sensor. They will need to be placed somewhere on the outside of the enclosure to be affected enough by the wind to give accurate and precise readings. The wind direction and speed will indicate what direction a fire is coming from and where it will spread if one is detected.	Matthew Wilson
FRS7	For the system to monitor the values of all the sensors' readings using serial communication for testing, it will have to loop to transmit and receive information. A loopback test is usually what's used to use both sides of communication for constant checking. Ideally, it will send information to the Arduino software on a computer. The user will receive the readings on the Arduino serial monitor after entering a command to test all sensors.	Luis Guevara
FRS8	This can be accomplished by using error checking programming. If the individual sensors give any reading that doesn't make sense, then there is an issue with that sensor. It will send a notification pop-up on the website alerting that there is an issue with a sensor. This error checking will take place before the sensor readings are sent to the website. If/else statements will be used for this. For example, if the wind vane sensor indicates the wind speed is less than 0 or greater than 500 mph, then there is an issue with the sensor because readings outside those bounds are impossible in our environment. The sensor data will still be sent to the website for users to view. The sensor status will be viewed only by the troubleshooter.	Edwin Hernandez
FRS9	A weatherproof camera will be interfaced to the microcontroller. The microcontroller will then allow for video feed to be sent over to the cloud. The video feed will only be active when the "Wildfire" system detects a fire. The camera module will include IR LEDs that will illuminate the area for nighttime monitoring. The camera will monitor at 30 frames per second.	Lluviana Vasquez
FRS10	Each unit will have a network interface card (NIC) that uses a SIM card to communicate using cellemetry. The NIC will communicate to cell towers using TCP protocol over one or more radio frequencies, the towers will then communicate to satellites, then finally the data is then sent to the server acting as the cloud database. The service we will use for the cloud	Matthew Wilson

	database will be Amazon Web Services (AWS) and for cell service will be AT&T or T-Mobile.	
FRS11	While users have the application installed on their mobile phone, they can receive push notifications. The notifications context will include warnings and potential fires. These notifications will be sent only if a box recognizes a potential fire starting.	Edwin Hernandez
FRS12	90 degrees Celsius is the maximum operating temperature for most processors. The components that generate the most heat are the PSU and processor. The system will use active cooling that will use 1 fan to ventilate the hot air out of the enclosure and colder air in. The fans will activate if the external temperature exceeds 80 degrees Celsius.	Matthew Wilson

b. Non-Functional Requirements Specifications

ID	Non-Functional Requirement Specification	Team Member Responsible
NFRS1	The enclosure that the "Wildfire" system would ideally use is a polycarbonate plastic material. The reason is because it's a tough plastic that has outstanding strength, stiffness, and impact resistance. Since this box will be outside, the material needs to be resistant to any weather and needs to be able to not retain too much heat. If polycarbonate plastic is hard to get, then we need to use just a plastic material that can withstand the weather. The plastic weather resistant enclosure should meet the standard IP67.	Lluviana Vasquez
NFRS2	The open parts of the enclosure will be covered in mesh to prevent insects from going in. The mesh will allow smoke and air to pass through. The mesh will be fine enough, blocking anything bigger than 400 microns. The mesh will be UV proof and durable. To protect the box from water, we would need to add a slanted edge on top (four sides) to make sure the water slides off without touching the sides of the box.	Lluviana Vasquez
NFRS3	A solar panel will be interfaced to the charge regulator, which will charge a pack of lithium-ion batteries. The batteries will hold enough power to operate the "Wildfire" system at night. The capacity of the battery is 2000 mAH, 3.7 V, and the time to charge is 2.4 hours with a 20% efficiency loss. The solar panel will be a 6 V, 9 W, outputting 6 V at around 1.5 A. The solar energy will be prioritized over battery power to conserve battery. The solar panel will be mounted on the enclosure, in an area that will be the most exposed to the sun.	Luis Guevara

NFRS4	The "Wildfire" system needs to be in a small enclosure that won't attract too much attention to it, but big enough to handle all its components. It being 20 cm x 15 cm x 15 cm is enough for its components without it being a clutter inside the enclosure. If there is enough space, then that means that all the components will be able to run smoothly in their own section. If we are using any type of plastic, we will need to measure out the size of it and then weld it together for it to be that exact size.	Lluviana Vasquez
NFRS5	Before ordering the components for the "Wildfire" system, I will add together the all the masses given in each component's datasheet to ensure that their combined mass is less than 700 grams. This will leave about 300 grams for the enclosure and PCB. A triple beam balance or digital scale will be used to measure the mass of the entire unit. Material from the enclosure will need to be removed if the unit exceeds 1 kg.	Matthew Wilson
NFRS6	To ensure the device is visible and blends into the forest environment, the device enclosure should be green. This will prevent animals from messing with the device while not disturbing the natural nature look	Edwin Hernandez
NFRS7	The "wildfire" system needs a visible indicator on the system enclosure. An emitting LED will be the indicator for the device. The purpose is to showcase if the device is operating, not working, or turned off.	Edwin Hernandez
NFRS8	The most expensive component of each unit will be the processor. Each member of the Quad Core Crew will select components that will best fulfill their requirements. I will review each component and make certain that the combined price of all the components and the manpower needed to integrate them does not exceed $300.	Matthew Wilson
NFRS9	The components and sensors of the "Wildfire" system will be soldered onto a PCB, so we do not have wires everywhere within the unit. PCB design software will be used to optimize the design of the PCB. This will help reduce the size of the "Wildfire" system unit.	Luis Guevara
NFRS10	There are 3rd party add-ons and software that will allow the website to be properly formatted for any display.	Edwin Hernandez

5. Constraints

a. *Completed on 4/30/2023.*

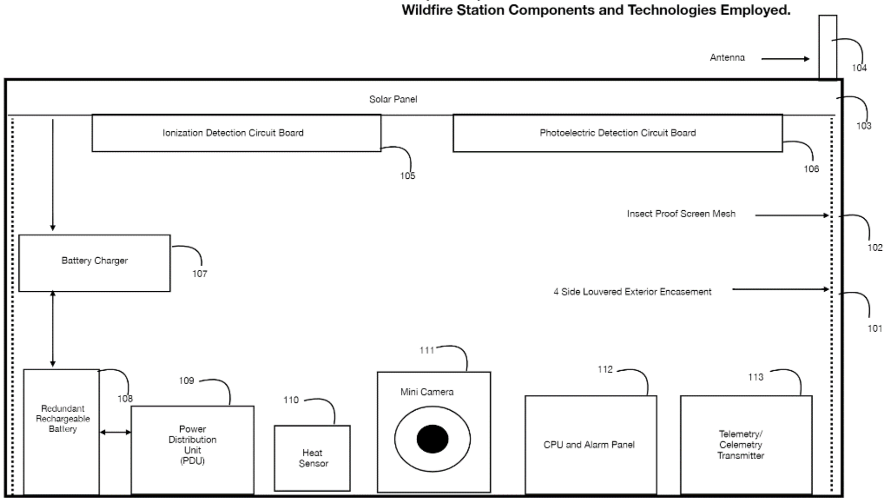
b. *Money constraint on components (should not be more than $300 per box)*
c. *Sensors need to be compatible with SPI or I2C buses.*
d. *Power limited by size and cost of the solar panels, usually 0.5 volts per solar cell.*
e. Requirements needing a PCB to be fulfilled one month before the due date, to give more time for PCB processing.

6. System Overview

a. System Overview

The "Wildfire, Forest Fire, and Detection System" or "Wildfire" will use the combination of existing "on the self" technology that will detect fires, forest fires, and wildfires. It will include sensors, such as: ionization and photoelectric smoke detectors, temperature and humidity sensors, a GSM/GPS module, a mini camera, a potentiometer, and an anemometer sensor. These sensors will send their data to a user that requests it through a website. This system will use cellemetry to notify Command, Control, and Dispatch of a potential fire. It'll have a rechargeable station where it relies on solar energy to generate enough power to run all its components, such as: mini camera, cpu and alarm panel, battery charger, heat sensor, cellemetry transmitter, and its ionization/photoelectric sensors. The ideal plan is to build three stations – triangulation – for a better reading on its location, so it could alarm the Command Control. Some constraints that we encountered were time, money, sensors not being compatible with SPI or I2C buses, and power being limited to the size and cost of solar panels.

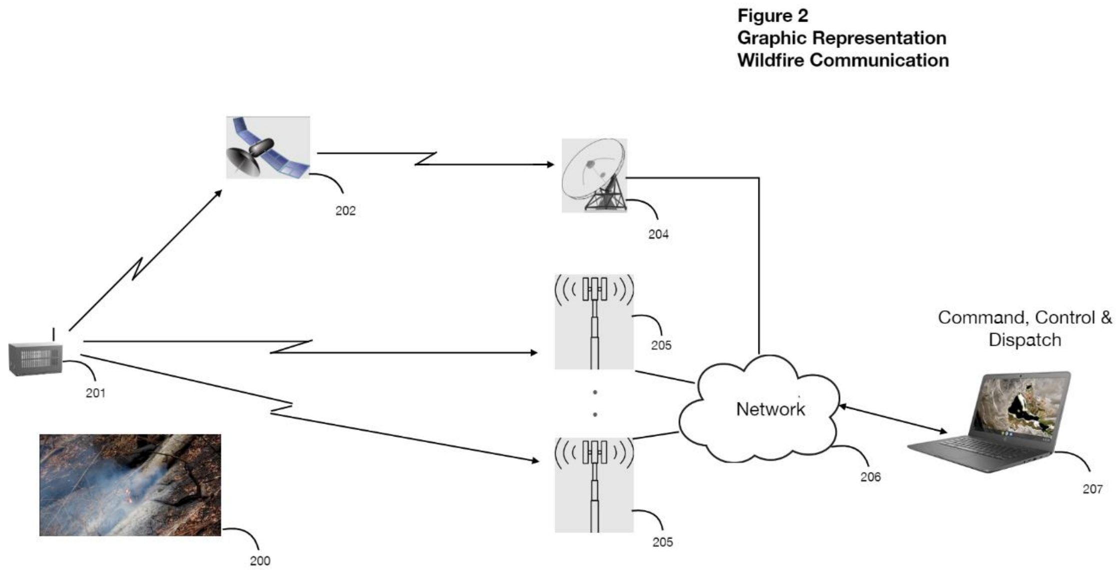
Figure 1
Graphic Representation
Wildfire Station Components and Technologies Employed.



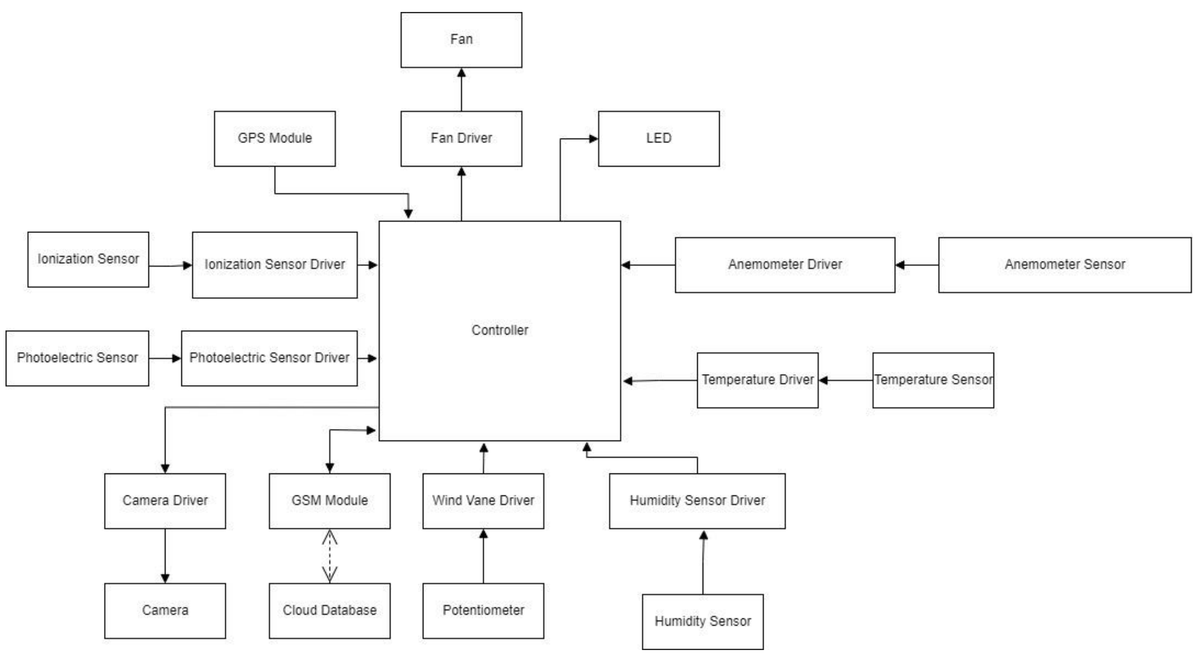
The image shows the basic diagram of what the box should include.

b. System Organization

Diagram of system sending notifications:



Block Diagram:



The “Wildfire” system is modeled after IOT devices, consisting of a microcontroller and interfaced modules, sensors and actuators that process input and give the according output, which are connected to a cell network. The major components of the “Wildfire” system are as follows.

Major Hardware Components:

GSM Module – allows for communication with the cellular network. Allows for sending/receiving of data and for triangulation purposes.

Camera – engages on when in the presence of a possible wildfire. Records video which is sent to command control to verify that there is a fire.

Photoelectric sensor – fire alarm optical sensor that detects smoldering fires and combustible particulate.

Ionization sensor – fire alarm sensor that utilizes smoke to detect a fire.

GPS Module – allows finding the location of the "Wildfire" device with precision.

Fan – used to cool down the "Wildfire" device if it overheats, as well as routing smoke to a chamber where sensors are housed.

LED – indicates status of the "Wildfire" device.

Anemometer sensor – allows to detect speed of wind, possibly of a wildfire.

Temperature sensor – measures temperature of "wildfire" device, temperature is used to control when the fan engages.

Humidity Sensor – measures relative humidity where the "wildfire" box is located.

Potentiometer – allows to determine the direction of the wind.

Major Software Components:

DynamoDB – Database used store necessary information such as field data, user login, subscribers, and video footage from hardware devices.

AWS Amplify– JavaScript SDK Package to allow access to AWS services in development environment.

Next.js - JavaScript Framework for building server-rendered React applications.

Arduino IDE – IDE used for developing software for the Arduino processor.

7. Project References

- ***Hybrid Positioning and CellLocate, https://content.ublox.com/sites/default/files/documents/CellLocate_WhitePaper_%28GSM-X-11001%29_0.pdf, 2011***
- ***Ionization vs Photoelectric, https://www.nfpa.org/Public-Education/Staying-safe/Safety-equipment/Smoke-alarms/Ionization-vs-photoelectric, 2013***

- ***MKRGSM1400 Library Examples, [https://docs.arduino.cc/tutorials/mkr-gsm-1400/mkr-gsm-library-examples,](https://docs.arduino.cc/tutorials/mkr-gsm-1400/mkr-gsm-library-examples) 2021***
- ***Arduino ArduCam – Motion Activated Video Spycam, https://www.elithecomputerguy.com/2020/06/arduino-arducam-motion-activated-video-spycam/, 2020***
- ***GitHub ArduCam/Arduino, https://github.com/ArduCAM/Arduino, 2015***
- ***Ionization Smoke Detected ASIC, https://www.microchip.com/en-us/product/RE46C120, 1998-2023***
- ***AWS Amplify Documentation, https://docs.amplify.aws/cli/, 2022***
- ***AWS IoT MQTT Documentation, https://docs.aws.amazon.com/iot/latest/developerguide/mqtt.html, 2022***
- ***Soracom Beam Documentation, https://developers.soracom.io/en/docs/beam/, 2022***
- ***AWS IoT to DynamoDB Tutorial, https://docs.aws.amazon.com/iot/latest/developerguide/iot-ddb-rule.html, 2022***
- ***U-Blox AT Commands Examples, https://content.u-blox.com/sites/default/files/AT-CommandsExamples_AppNote_%28UBX-13001820%29.pdf, 2019***

8. Glossary

Field Data	Temperature, humidity, wind speed, wind direction, ionization smoke detection, photoelectric smoke detection, fire alarm flag.
Cellemetry	Consists of software programs that provide telemetry features to facilitate the use by the cellular user.
Anemometer	Instrument that measures wind speed.
S3 Bucket	Used to for storing and retrieving files in the cloud.
Lambda Function	Used to run code in response to specific events or triggers.
Ionization Smoke Alarms	Contains a small amount of radioactive material between two electrically charged plates that ionizes the air and causes current to flow between the plates. When smoke enters the chamber, it disrupts the flow of ions which reduces the flow of current and activating the alarm.
Photoelectric Smoke Alarms	Has a light source pointing into a sensing chamber at an angle away from the sensor. Smoke enters the chamber, reflecting light onto the light sensor, triggering the alarm.
SSL	Secure Socket Layer – encryption security protocol that uses certificates and key-pairs to encrypt and authenticate the data.

Soracom Beam	Soracom is a brand of SIM card used for cellemetry. Soracom Beam is a service that authenticates data using SIM tokens, routes the data from an IoT device to an external service, and encrypts the data using security credentials. We are using Beam to route data from the device to AWS IoT and encrypting using SSL credentials.

B. Management Overview

1. Description of Implementation

The "Wildfire" device consists of the Arduino MKRGSM 1400, a microcontroller capable of sending data through cellemetry. The microcontroller will be programmed using Arduino IDE. Sensors and actuators will be interfaced to the MKRGSM 1400, to fulfill the functionalities of the device. The GSM module is already on the MKRGSM 1400 board, so no need to interface. The only sensors/actuators needing to be interfaced to the microcontroller are the camera, the photoelectric sensor, the ionization sensor, the GPS Module, a fan, an LED, the anemometer sensor, the temperature sensor, the Humidity Sensor and finally the potentiometer. With the camera, GPS module, anemometer sensor, temperature, and humidity sensor there are libraries that ease development and interfacing of those components. The photoelectric sensor, ionization sensor, fan and LED do not have libraries associated with them but will still need to be interfaced.

Once all sensors and actuators are interfaced, the power system for the device will be designed. The "wildfire" device will run on a battery that provides power to the microcontroller, sensors, and actuators. The battery will be charged by solar power, as the "wildfire" device will be in a remote location with no power source.

2. Points of Contact

Project Manager: Robin Pottathuparambil, **Email**: Robin.Pottathuparambil@unt.edu
Team Leader: Matthew Wilson, **Email**: MatthewWilson10@my.unt.edu
Hardware Leader: Luis Guevara, **Email**: LuisGuevara@my.unt.edu
Software Leader: Edwin Hernandez, **Email**: edwinhernandez2@my.unt.edu
Reporter: Lluviana Vasquez, **Email**: lluvianavasquez@my.unt.edu

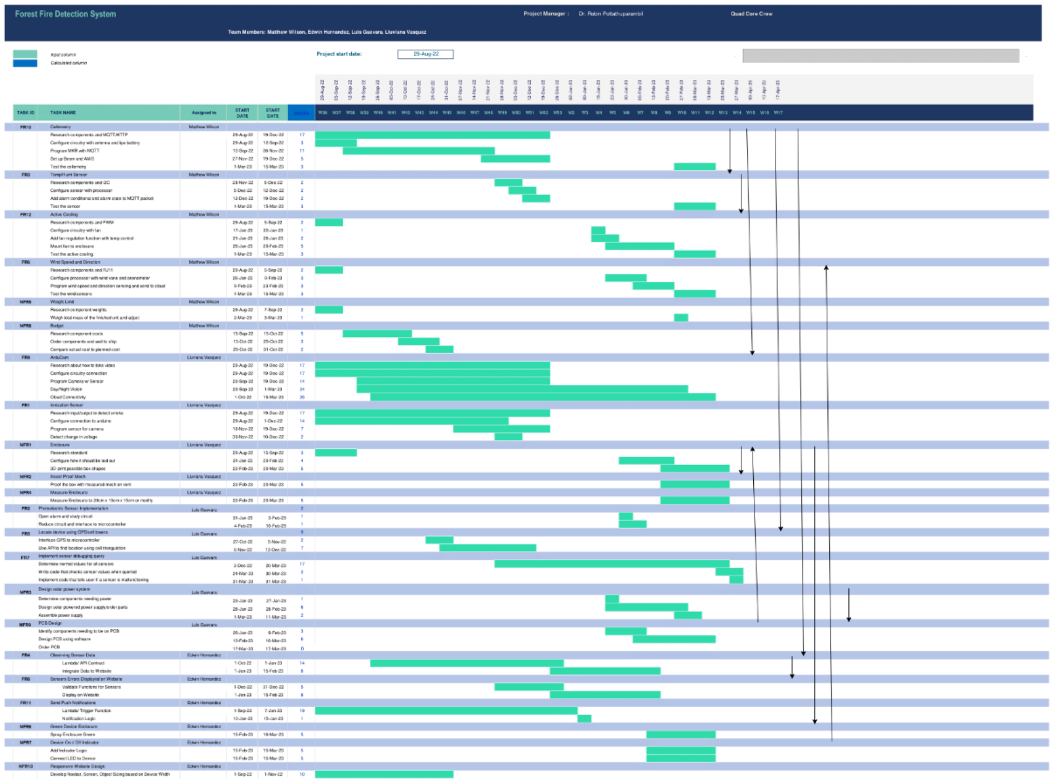
3. Major Tasks

ID	Major Task	Assigned to	Start Date	End Date
FR1	Research input/output to detect smoke. Configure connection to Arduino. Program sensor for camera. Detect change in voltage.	Lluviana Vasquez	29-Aug-22 29-Aug-22 10-Nov-22 30-Nov-22	19-Dec-22 1-Dec-22 19-Dec-22 10-Dec-22
FR2	Open alarm and study circuit. Reduce circuit and interface to microcontroller.	Luis Guevara	31-Jan-23 4-Feb-23	3-Feb-23 10-Feb-23
FR3	Research components and I2C.	Matthew Wilson	28-Nov-22	5-Dec-22

	Configure sensor with processor. Add alarm conditional and alarm state to MQTT packet. Test the sensor.		5-Dec-22 12-Dec-22 1-Mar-23	12-Dec-22 19-Dec-22 15-Mar-23
FR4	Create Lambda Function with Fetch Logic from DynamoDB Table. Connect Map to Fetched Data.	Edwin Hernandez	1-Oct-22 1-Jan-23	1-Jan-23 15-Feb-23
FR5	Interface GPS to microcontroller. Use API to find location using cell triangulation.	Luis Guevara	27-Oct-22 6-Nov-22	5-Nov-22 13-Dec-22
FR6	Research components and RJ11 Configure processor with wind vane and anemometer. Program wind speed and direction sensing and send it to cloud. Test the wind sensors.	Matthew Wilson	29-Aug-22 26-Jan-23 9-Feb-23 1-Mar-23	5-Sep-22 9-Feb-23 23-Feb-23 15-Mar-23
FR7	Determine normal values for all sensors. Write code that checks sensor values when queried. Implement code that tells user if a sensor is malfunctioning.	Luis Guevara	3-Dec-22 24-Mar-23 31-Mar-23	20-Mar-23 30-Mar-23 31-Mar-23
FR8	Create Validate Functions for each sensor. Display output in a dropdown component on the trouble shooter page.	Edwin Hernandez	1-Dec-22 1-Jan-23	31-Dec-22 15-Feb-23
FR9	Research about how to take videos. Configure circuitry connection. Program Camera w/ Sensor. Day/Night Vision. Cloud Connectivity.	Lluviana Vasquez	29-Aug-22 29-Aug-22 23-Sep-22 23-Sep-22 1-Oct-22	19-Dec-22 19-Dec-22 19-Dec-22 1-Mar-23 15-Mar-23
FR10	Research components and MQTT/HTTP. Configure circuitry with antenna and lipo battery. Program MKR with MQTT. Set up Beam and AWS. Test the cellemetry.	Matthew Wilson	29-Aug-22 29-Aug-22 12-Sep-22 27-Nov-22 1-Mar-23	19-Dec-22 12-Sep-22 26-Nov-22 19-Dec-22 15-Mar-23
FR11	Create Lambda Function with Fetch Subscribed Phone Numbers and connect Trigger method to DynamoDB Table. Create Notification Logic.	Edwin Hernandez	1-Sep-22 10-Jan-23	7-Jan-23 15-Jan-23
FR12	Research components and PWM. Configure circuitry with fan. Add fan regulation function with temp control. Mount fan to enclosure. Test the active cooling.	Matthew Wilson	29-Aug-22 17-Jan-23 21-Jan-23 26-Jan-23 1-Mar-23	5-Sep-22 20-Jan-23 26-Jan-23 20-Feb-23 15-Mar-23
NFR1	Research standard. Configure how it should be laid out.	Lluviana Vasquez	29-Aug-22 31-Jan-23	15-Sep-22 20-Feb-23

	3D-print possible box shapes.		22-Feb-23	20-Mar-23
NFR2	Proof the box with measured mesh on vent.	Lluviana Vasquez	22-Feb-23	20-Mar-23
NFR3	Determine components needing power.	Luis Guevara	23-Jan-23	27-Jan-23
	Design solar powered power supply/order parts.		28-Jan-23	28-Feb-23
	Assemble power supply		1-Mar-23	11-Mar-23
NFR4	Measure Enclosure to 20cm x 15cm x 15cm or modify.	Lluviana Vasquez	22-Feb-23	20-Mar-23
NFR5	Research component weights.	Matthew Wilson	29-Aug-22	7-Sep-22
	Weigh total mass of the finished unit and adjust.		2-Mar-23	3-Mar-23
NFR6	Spray Enclosure Green	Edwin Hernandez	15-Feb-23	16-Mar-23
NFR7	Add Indicator Logic	Edwin Hernandez	15-Feb-23	15-Mar-23
	Connect LED to Device		15-Feb-23	15-Mar-23
NFR8	Research component costs.	Matthew Wilson	15-Sep-22	15-Oct-22
	Order components and wait for them to ship.		15-Oct-22	29-Oct-22
	Compare actual cost to planned cost.		29-Oct-22	31-Oct-22
NFR9	Identify components needing to be on PCB.	Luis Guevara	26-Jan-23	9-Feb-23
	Design PCB using software.		10-Feb-23	16-Mar-23
	Order PCB		17-Mar-23	17-Mar-23
NFR10	Develop Navbar, Screen, Object Sizing based on Device Width	Edwin Hernandez	1-Sep-22	1-Nov-22

4. Implementation Schedule



5. Security

a. System Security Features

- Website Authorization: We are using Amazon Cognito User Pools which handles user registration, authentication, and account recovery. Cognito uses a one-way hashing algorithm, specifically SHA-256 (Secure Hash Algorithm 256-bit) to encrypt the data when being sent to process Authorization.
- Storing User Data: Uses AWS Cognito to send Subscribed user's data to DynamoDB table.
- Network Encryption: The packets sent from the device are authenticated by the SIM card. Soracom Beam sends the data to AWS IoT using SSL authentication and encryption. SIM card credentials are stored and referenced in header file as global variables. This way they are hidden from the source code while still being usable.

b. Security During Implementation

- Lock the computers when not in use.
- Sensitive user data is never sent to or from the devices.

- In case of theft, the location of the device can be tracked as long as it has a GPS or cellemetry connection.
- Devices are secured to trees/poles in the air to prevent theft.

C. Implementation Support

1. Hardware, Software, Facilities, and Materials

a. Hardware

- Arduino MKRGSM 1400
- 2 in 1 Photoelectric and Ionization sensor smoke detector
- NEO-6M GPS Module
- Solar Panel
- Power supply
- Li-Po battery
- ArduCam Arduino camera
- SD Card shield for Arduino

b. Software

- AWS Services (Commercial off-the-shelf)
    - AWS DynamoDB
    - AWS Location Service (Maps)
    - AWS SNS (Simple Notification Service)
    - AWS Amplify
    - AWS Lambda Functions
    - AWS CloudFront
    - AWS IoT (Internet of Things)
- Next.js (React/TypeScript)
- Arduino IDE
- Soracom Beam

c. Facilities

- Workspace to assemble components.
- Soldering station to solder components.
- Hot air station to disassemble ready-made components.
- Remote outdoor area to test smoke detector and test GPS module functionality.

d. Material

- Breadboards
- DuPont cables and connectors
- Wire cutters
- SD Card

2. Personnel

a. Personnel Requirements and Staffing

- Number of personnel: 4

- Length of time: 4 months

Types of Skills	Skill Level
Soldering	Medium (training needed)
3D Printing	Varies (training needed)
Coding	Medium to High
PCB design	High
Networking	High
Circuitry	Medium

b. Training of Implementation Staff

Implementation staff will need to have the mechanical skill to secure the device to a sturdy tree about 10 – 15 feet tall, or to install a secure pole of similar sturdiness and height to secure the device. Staff will need to read the user manual for calibrating the sensors. In case of low cellemetry signal, staff will need to be trained in installing hardware to extend cellemetry signal.

Training Curriculum:

Courses Provided	Dates
Wildfire device training	March 15,2023 – April 15, 2023
Wildfire website training	March 15,2023 – April 15, 2023
Wildfire mounting training	March 15,2023 – April 15, 2023

D. Implementation Requirements by Site

1. Forests, woodlands

Protecting forests and woodlands was the main motivation behind implementing the "Wildfire" device. The "Wildfire" devices are to be deployed in forests, or heavily wooded areas that are prone to fires.

a. Site Requirements

- *Hardware Requirements* - For the "Wildfire" device to communicate with command control, as well as to find the location of the "Wildfire" device, the device needs to be in proximity of GPS satellites and cell towers. The "Wildfire" device provides all the hardware needed to detect forest fires and alert command control.
- *Software Requirements* – Command control will utilize the OS of their choice to access the web page that monitors the forest areas. No specialized software is needed to be provided by the site.
- *Data Requirements* - Manually assign the hardware device a name so when sending data to the cloud, it will be a unique dataset.
- *Facilities Requirements - Describe the site-specific physical facilities and accommodation required during the system implementation period. Some examples of this type of information are provided in Section 3.]*
- The "Wildfire" devices will be mounted high so that the sun will help power them. An area will be necessary to mount, test and verify that the "Wildfire" device is in

working order. The area will need abundant sunlight. On-site computers or cell phones with internet access will be needed to access the webpage that alerts command control of the presence of a wildfire.

b. Site implementation Details

Team:

- A team of 4 will be required to deploy the “Wildfire” devices in a forest. One team member will determine at what locations the “Wildfire” will be at. Two team members will be responsible for installing the “Wildfire” devices at those locations. One team member will be responsible for opening the webpage to see the status of the “Wildfire” devices. The team of 2 will test the “Wildfire” device, and the team member checking the status of the devices will verify with the team of 2 that the device is working.

Schedule:

1st: Build the devices with all their components – Date: Take at least 2 months to do this since you must build the PCBs (Ex. Start date: 9/1 to End date: 11/1)

2nd: Start connecting them using cellemetry to make sure that they are online – Date: This should take at least 2 months to make sure everything is running smoothly on all programs (Ex. Start date: 11/1 to End date: 1/1)

3rd: Once everything is set up, start the website up and see if the devices show up – Date: Should take at least half a month (Ex. Start date: 1/1 to End date: 2/1)

4th: Start testing the devices by creating a fake fire to get data and video – Date: At least a month (Ex. Start date: 2/1 to End date: 3/1)

5th: If everything is positive, start making plans to deploy and test in a real scenario. Locate a forest where small fires are allowed – Date: Take a week at least (Ex. Start date: 3/1 to End date: 3/10)

6th: Once location is located, travel to designated area and start looking for a place to set up devices (3) so triangulation can work. – Date: Start in the morning of that designated day (Ex. Start date: 3/15 to End date: TBD)

7th: Devices once mounted to poles or higher ground, will need to be ran through the website to see if they display on map and if they are all working correctly – Date: Throughout, the morning of that designated date (Ex. Start date: 3/15 to End date: TBD)

8th: Start a fake fire and see if sensors start detecting and video is recorded for that period – Date: Still the same designated date (Ex. Start date: 3/15 to End date: TBD)

9th: If successful, try to triangulate the location of the fire by using the (3) devices – Date: Still the same designated date (Ex. Start date: 3/15 to End date: TBD)

10th: Run multiple tests on these devices and make sure no errors appear on the website – Date: Still the same designated date (Ex. Start date: 3/15 to End date: TBD)

- Tests may differ as well as timeline. In *Section B - 3. Major Tasks*, there is a table with dates on them. This is a visual representation of how long it might take a team to build.

Procedures:

The deployment of the “Wildfire” device is to happen immediately after verifying the functionality of the device. The size of the implementation team, time used to implements are all dependent on the number of “Wildfire” devices deployed to the area, which is dependent on the size of the area. Time to implement should take a day for a small area of 1-5 acres, and multiple for larger areas.

- Locate areas where device should be mounted.

The device will be mounted on a tall place where there is abundant sunlight. A building, steel pole will be appropriate for mounting the device.

- Log the device number and location of device.

Each device will have an ID associated with them. A log will be kept which includes the device ID and the location of the device.

- Install devices with necessary tools at specified locations.

Each device will be installed according to the terrain. Most devices will be mounted on a pole on the terrain. Some devices can be mounted on a building, or whatever is available.

- Test devices with controlled fire

A team of 3 people will test each device. 2 people will give the device test inputs, and one other person will verify that the device is in working order by referencing the webpage that gives the alerts.

- Verify devices are operational.

Sufficient testing must be done to verify the device is in working order. Another team can test a device that was already tested to double check the functionality of the device.

Database:

There will be an SD card to temporarily hold captured video footage and then utilize Amazon Services, specifically AWS DynamoDB and S3 Bucket. There is only a production level environment that will store necessary data.

When the Device database retrieves data, a lambda function passes in the device data as a parameter. The logic checks the alert attribute value and if conditions are met, the operation will continue to retrieve subscribed users and send a message about which device is being alarmed. While this is going on, another trigger event is executed to when the video file is being stored in the S3 bucket, it is converting the video format from avi to mp4. To establish the database, we would need to provide the necessary credentials to access the databases.

Data Update:

If data update procedures are described in another document, such as the operations manual or conversion plan, that document may be referenced here. The following are examples of information to be included:

c. Back-Off Plan

It is imperative that the “Wildfire” devices are deployed as soon as possible, to prevent future forest fires. The “Wildfire” devices will require to be mounted in an area where there is abundant sunlight. Only then can the “Wildfire” device be installed.

To restore the site to its original condition, the “Wildfire” devices, along with their mount must be removed. The “Wildfire” device is non-invasive and does not alter the terrain or landscape. The devices leave no trace, with no evidence of the devices being mounted.

Forest Fire Detection System

Project Manager : Dr. Robin Pottathuparambil

Quad Core Crew

Team Members: Matthew Wilson, Edwin Hernandez, Luis Guevara, Lluviana Vasquez

Input column
Calculated column

Project start date: 29-Aug-22

29-Aug-22 | 05-Sep-22 | 12-Sep-22 | 19-Sep-22 | 26-Sep-22 | 03-Oct-22 | 10-Oct-22 | 17-Oct-22 | 24-Oct-22 | 31-Oct-22 | 07-Nov-22 | 14-Nov-22 | 21-Nov-22 | 28-Nov-22 | 05-Dec-22 | 12-Dec-22 | 19-Dec-22 | 26-Dec-22 | 02-Jan-23 | 09-Jan-23 | 16-Jan-23 | 23-Jan-23 | 30-Jan-23 | 06-Feb-23 | 13-Feb-23 | 20-Feb-23 | 27-Feb-23 | 06-Mar-23 | 13-Mar-23 | 20-Mar-23 | 27-Mar-23 | 03-Apr-23 | 10-Apr-23 | 17-Apr-23

W36 W37 W38 W39 W40 W41 W42 W43 W44 W45 W46 W47 W48 W49 W50 W51 W52 W53 W2 W3 W4 W5 W6 W7 W8 W9 W10 W11 W12 W13 W14 W15 W16 W17

TASK ID	TASK NAME	Assigned to	START DATE	START DATE	WEEKS
FR10	Cellemetry	Matthew Wilson			
	Research components and MQTT/HTTP		29-Aug-22	19-Dec-22	17
	Configure circuitry with antenna and lipo battery		29-Aug-22	12-Sep-22	3
	Program MKR with MQTT		12-Sep-22	26-Nov-22	11
	Set up Beam and AWS		27-Nov-22	19-Dec-22	5
	Test the cellemetry		1-Mar-23	15-Mar-23	3
FR3	Temp/Humi Sensor	Matthew Wilson			
	Research components and I2C		28-Nov-22	5-Dec-22	2
	Configure sensor with processor		5-Dec-22	12-Dec-22	2
	Add alarm conditional and alarm state to MQTT packet		12-Dec-22	19-Dec-22	2
	Test the sensor		1-Mar-23	15-Mar-23	3
FR12	Active Cooling	Matthew Wilson			
	Research components and PWM		29-Aug-22	5-Sep-22	2
	Configure circuitry with fan		17-Jan-23	20-Jan-23	1
	Add fan regulation function with temp control		21-Jan-23	26-Jan-23	2
	Mount fan to enclosure		26-Jan-23	20-Feb-23	5
	Test the active cooling.		1-Mar-23	15-Mar-23	3
FR6	Wind Speed and Direction	Matthew Wilson			
	Research components and RJ11		29-Aug-22	5-Sep-22	2
	Configure processor with wind vane and anemometer		26-Jan-23	9-Feb-23	3
	Program wind speed and direction sensing and send to cloud		9-Feb-23	23-Feb-23	3
	Test the wind sensors		1-Mar-23	15-Mar-23	3
NFR5	Weight Limit	Matthew Wilson			
	Research component weights		29-Aug-22	7-Sep-22	2
	Weigh total mass of the finished unit and adjust		2-Mar-23	3-Mar-23	1
NFR8	Budget	Matthew Wilson			
	Research component costs		15-Sep-22	15-Oct-22	5
	Order components and wait to ship		15-Oct-22	29-Oct-22	3
	Compare actual cost to planned cost		29-Oct-22	31-Oct-22	2
FR9	ArduCam	Lluviana Vasquez			
	Research about how to take video		29-Aug-22	19-Dec-22	17
	Configure circuitry connection		29-Aug-22	19-Dec-22	17
	Program Camera w/ Sensor		23-Sep-22	19-Dec-22	14
	Day/Night Vision		23-Sep-22	1-Mar-23	24
	Cloud Connectivity		1-Oct-22	15-Mar-23	25
FR1	Ionization Sensor	Lluviana Vasquez			
	Research input/output to detect smoke		29-Aug-22	19-Dec-22	17
	Configure connection to arduino		29-Aug-22	1-Dec-22	14
	Program sensor for camera		10-Nov-22	19-Dec-22	7
	Detect change in voltage		30-Nov-22	10-Dec-22	2
NFR1	Enclosure	Lluviana Vasquez			
	Research standard		29-Aug-22	15-Sep-22	3
	Configure how it should be laid out		31-Jan-23	20-Feb-23	4
	3D-print possible box shapes		22-Feb-23	20-Mar-23	5
NFR2	Insect Proof Mesh	Lluviana Vasquez			
	Proof the box with measured mesh on vent		22-Feb-23	20-Mar-23	5
NFR4	Measure Enclosure	Lluviana Vasquez			
	Measure Enclosure to 20cm x 15cm x 15cm or modify		22-Feb-23	20-Mar-23	5
FR2	Photoelectric Sensor Implementation	Luis Guevara			2
	Open alarm and study circuit		31-Jan-23	3-Feb-23	1
	Reduce circuit and interface to microcontroller		4-Feb-23	10-Feb-23	1
FR5	Locate device using GPS/cell towers	Luis Guevara			9
	Interface GPS to microcontroller		27-Oct-22	5-Nov-22	2
	Use API to find location using cell triangulation		6-Nov-22	13-Dec-22	7
FR7	Implement sensor debugging query	Luis Guevara			
	Determine normal values for all sensors		3-Dec-22	20-Mar-23	17
	Write code that checks sensor values when queried		24-Mar-23	30-Mar-23	2
	Implement code that tells user if a sensor is malfunctioning		31-Mar-23	31-Mar-23	1
NFR3	Design solar power system	Luis Guevara			
	Determine components needing power		23-Jan-23	27-Jan-23	1
	Design solar powered power supply/order parts		28-Jan-23	28-Feb-23	6
	Assemble power supply		1-Mar-23	11-Mar-23	2
NFR9	PCB Design	Luis Guevara			
	Identify components needing to be on PCB		26-Jan-23	9-Feb-23	3
	Design PCB using software		10-Feb-23	16-Mar-23	6
	Order PCB		17-Mar-23	17-Mar-23	0
FR4	Observing Sensor Data	Edwin Hernandez			
	Lambda/ API Contract		1-Oct-22	1-Jan-23	14
	Integrate Data to Website		1-Jan-23	15-Feb-23	8
FR8	Sensors Errors Displayed on Website	Edwin Hernandez			
	Validate Functions for Sensors		1-Dec-22	31-Dec-22	5
	Display on Website		1-Jan-23	15-Feb-23	8
FR11	Send Push Notifications	Edwin Hernandez			
	Lambda/ Trigger Function		1-Sep-22	7-Jan-23	19
	Notification Logic		10-Jan-23	15-Jan-23	1
NFR6	Green Device Enclosure	Edwin Hernandez			
	Spray Enclosure Green		15-Feb-23	16-Mar-23	5
NFR7	Device On // Off Indicator	Edwin Hernandez			
	Add Indicator Logic		15-Feb-23	15-Mar-23	5
	Connect LED to Device		15-Feb-23	15-Mar-23	5
NFR10	Responsive Website Design	Edwin Hernandez			
	Develop Navbar, Screen, Object Sizing based on Device Width		1-Sep-22	1-Nov-22	10