Project Documentation

July 24, 2025

HING JOC Wildfire Incident Project

TABLE OF CONTENTS

[Introduction 4](#_Toc204156384)

[Mission Objective 5](#_Toc204156385)

[Problem Statement 5](#_Toc204156386)

[Solution Compatibility 5](#_Toc204156387)

[Functional Requirements 6](#_Toc204156388)

[Interface Requirements 7](#_Toc204156389)

[User Interface 7](#_Toc204156390)

[Proposed Layout 7](#_Toc204156391)

[Finalized Dashboard 11](#_Toc204156392)

[Alert Form 12](#_Toc204156393)

[Incident and News Page Section 13](#_Toc204156394)

[Active Alerts 13](#_Toc204156395)

[Login Page 14](#_Toc204156396)

[Planned Features 15](#_Toc204156397)

[Website Framework 16](#_Toc204156398)

[API Framework 17](#_Toc204156399)

[Data Sources 19](#_Toc204156400)

[Weather, Air Quality, Drought, and Fires 19](#_Toc204156401)

[Other 20](#_Toc204156402)

[Aerial Tracking 20](#_Toc204156403)

[Scraped Content 21](#_Toc204156404)

[Security, Maintenance, Reliability Overview 22](#_Toc204156405)

[Security 22](#_Toc204156406)

[Maintenance 23](#_Toc204156407)

[Reliability 23](#_Toc204156408)

[Lines of Effort 26](#_Toc204156409)

[Installation Guide 28](#_Toc204156410)

[Host Server Requirements 29](#_Toc204156411)

[Schedule For JOC HING Firewatch Project 30](#_Toc204156412)

[Proposed Budget 33](#_Toc204156413)

[Tracking 33](#_Toc204156414)

[Hosting 33](#_Toc204156415)

# Introduction

This Joint Operations Center (JOC) Firewatch Project, under the Hawaii National Guard is a data-driven web application solution designed to support the mitigation efforts across the State of Hawaii, accounting for all counties. This platform consolidates key information from both state and federal sources, delivering real-time insights to emergency responders, policy makers, and strategic planners.

This project includes the federal and scientific communities to provide modern mapping with extensive insights into Hawaii's environmental threats.

Through this documentation, stakeholders and planners alike will find the use of this project for the deployment of resources to fighting fires, and coordinating with the County and State to gather and pinpoint areas of need.

The project includes tools for pulling data, creating manual alerts, and tracking vital resources like emergency aerial support groups. Combined with other tools like NGB COP and WebEOC will ease the difficulties in responding to wildfire threats due to the number of systems currently being used.

Being a custom solution, there could be further developments on this platform for further expansions and extensions.

As a note this document will mix and match the terms: web solution, web application, and software solution.

This document was created under the Pacific Intelligence Innovation Imitative (P3I): VICEROY MAVEN HING group.



# Mission Objective

The mission objective of this project is to aggregate multiple open-source datasets into a single web-based software solution. By aggregating state and federal data, the platform will assist emergency response services, including the National Guard and the State to pull resources from available aerial units to combat wildfires across the counties in the State of Hawaii. Our team is developing multi-layer mapping capabilities to visualize current and potential wildfires. State entities provide information for reports, locations, first-response teams, and report times. The federal information flows consist of UV indexes, air quality, brightness mapping, drought, fires, and geographic scans.

# Problem Statement

Due to the limitations of the software solutions that will be the final result, we need information that is not locked away by requests-only access and/or is not locked by API access keys such as a higher-tier paid service requirement or services locked to enterprise environments. In addition, APIs need to have interoperability with one another to ensure that they can be used in the end product. This will require documentation of the software solution and APIs, access to those API keys and programming interface, and authorization as well. For the predictability of future events, this project will need access to a server, or hardware solutions that will make use of certain compute units that can process data fast enough for machine learning and possibly LLM usage. Similarly, the data pulled must follow a certain format, which can help mitigate further processing and delays in output processes.

# Solution Compatibility

As a web solution, this platform was intended to target a multitude of devices. Such as desktops, laptops, tablets, and mobile computing devices like smartphones. For the hosting aspect of this project, it will depend on the hardware and software package and/or libraries that will support the technologies and requirements later specified in this document. A multitude of APIs could be applied but due to time constraints and current implementation of technologies, there have been some limitations that could be addressed and more technologies added in the future.

# Functional Requirements

1. Data Safeguarding Requirements
2. Data fed through the solution may require safeguarding, such as weather data collected by federal systems.
3. Confidentiality: Medium
4. Some data collected is restricted to project-only usage and may be subject to fetch limitations.
5. API Key Security
6. These systems may involve API keys that must be hidden or obfuscated.
7. NOTE: API keys or tokens act as identifiers used to authenticate requests to a service or database.
8. Hosting Requirements and Access Control

The web solution will require internet access:

1. If hosted via a third-party cloud provider
2. Or hosted internally on a system requiring proper security and access controls

Integrity: High, Availability: High

For third-party hosting, contractual agreements will govern data sensitivity and handling procedures.

1. Self-Hosting Considerations
2. Self-hosted deployments will require:

* Network and firewall configuration
* System security and updates
* Physical access control
* Ongoing technical support

1. Interagency Communication & Notifications
2. Communication will involve federal, state, and third-party actors who use and manage API key access.
3. Additional alerts and notifications can be triggered through an interactive web form.
4. This alerting system can be restricted to registered users within the platform.

Interface Requirements

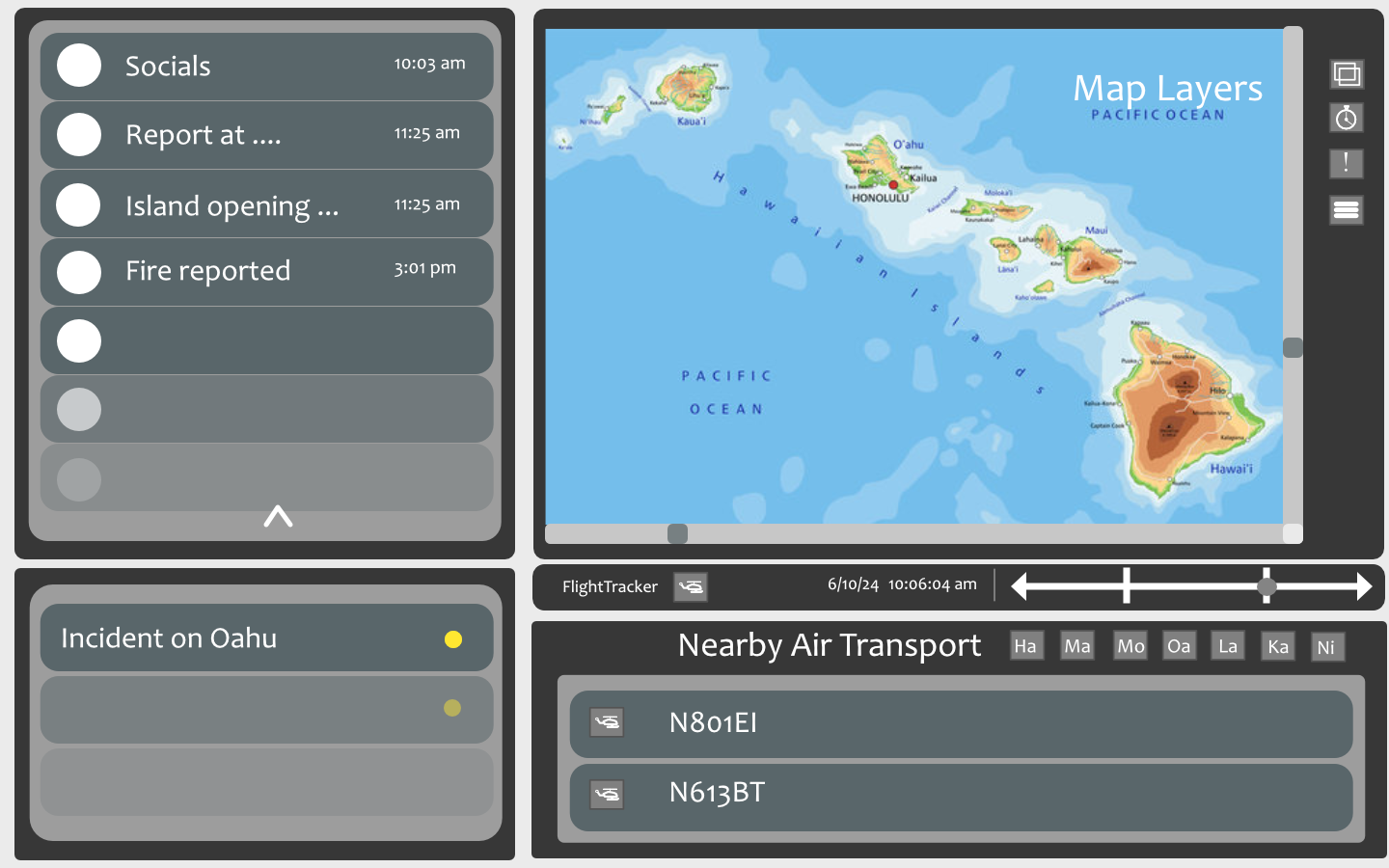
## User Interface

The dashboard will be implemented to follow a three or four-quadrant section appearance. The biggest and main component will be the interactive map that will be at the top. Followed by a notification and alert section that will be on the side. The last section which could be considered is a notification section dedicated to the nearest aerial vehicles, filtered to each island.

The map will contain different layers. The two main views will be a flat and a satellite representation. There will also be multiple layers to track temperature/UV, points of interest (fires), wind direction, and air pressure.

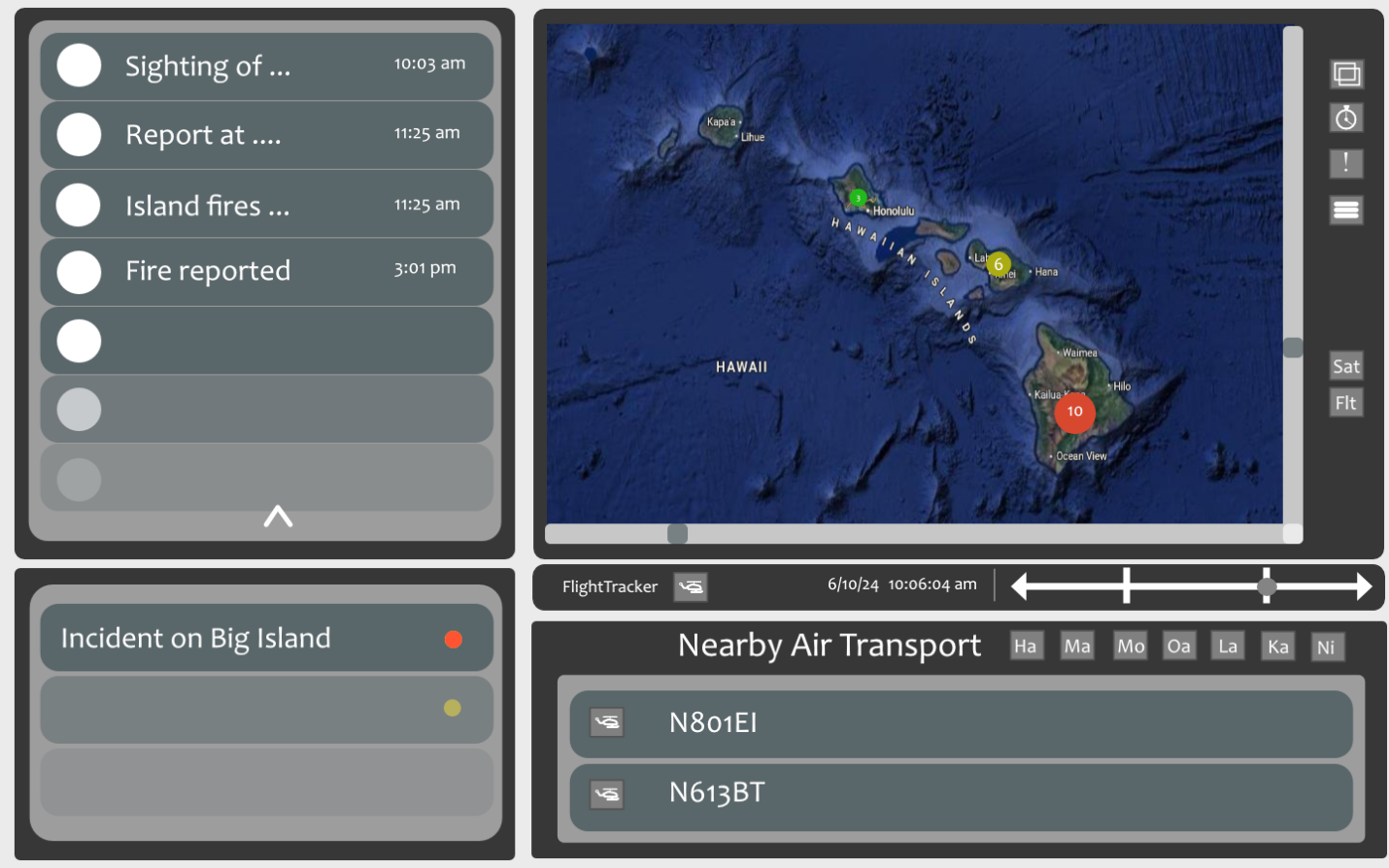
These elements can viewed under the proposed layout section below.

## Proposed Layout

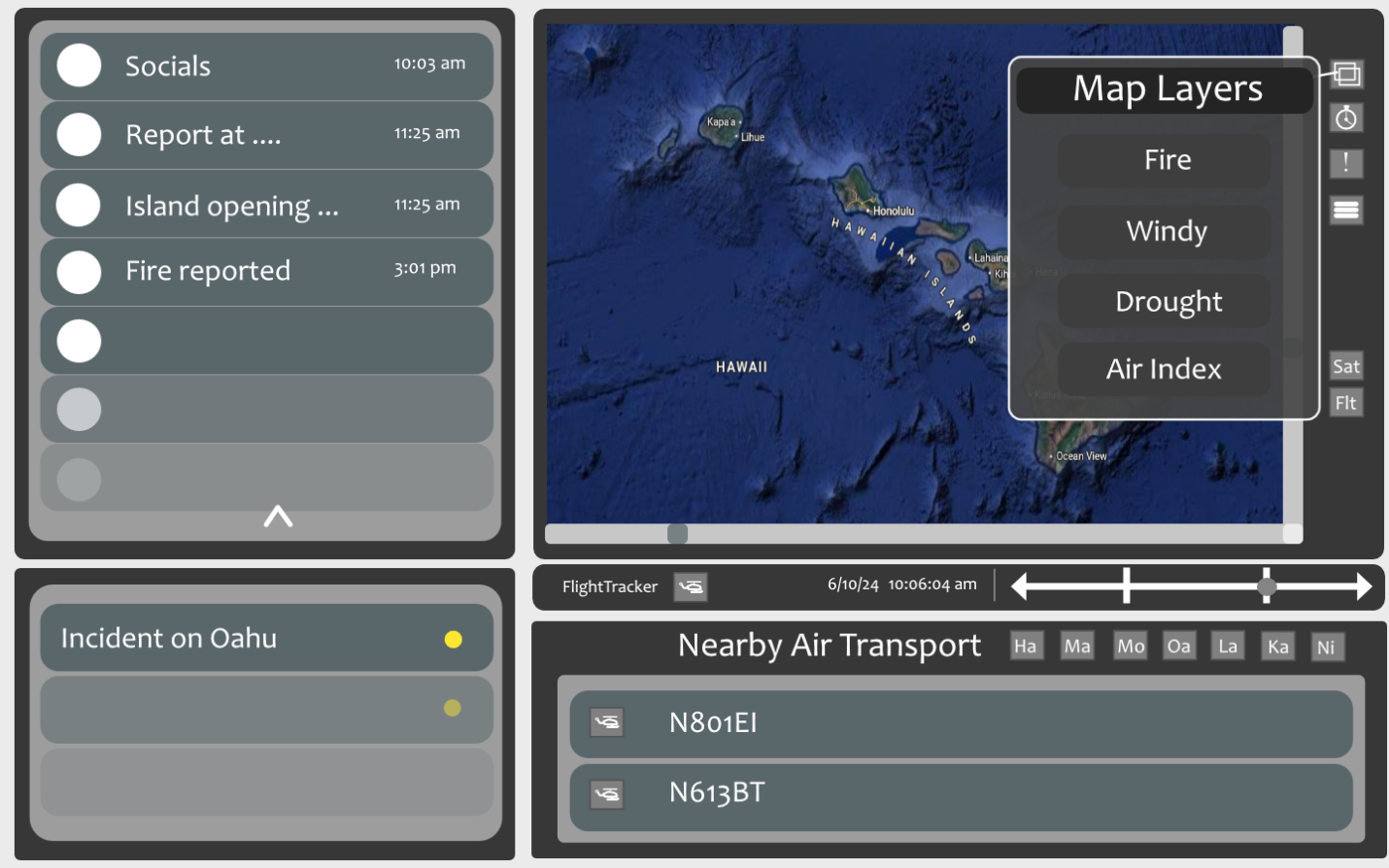


**Figure 1**  Layout for custom-made website

The proposed layout will have a news report message slider on the top left. It will contain social media posts about fires limited to the islands in addition with posts made by state departments like the HPD (police) and HFD (fire) and surrounding counties. The lower left notification section will include major fires within the islands. The top left section contains the map and map buttons for layers, filters, and additional information. The navigation bar right below the map contains the toggle overlay for the FlightTracker24 viewer and a time slot bar for past events. On the bottom right we finally have the emergency transport vehicle notification section, additionally there will be a filter per island.



**Figure 2** Layout from default map, Points are clustered until zoomed in as indicated by Figure 4.



**Figure 3**  Backup layout overlay with Satellite layer mode.

This layout now shows a overlay for map layers using the layers button on the map page. In the image it shows Fire, Windy, Drought, and Air Index layer filters. If you click on the hamburger icon for additional map options it should show additional options to select between different map points such as fires with an estimated zone, different data sources, showing fire station locations, and other displayed map points/data.

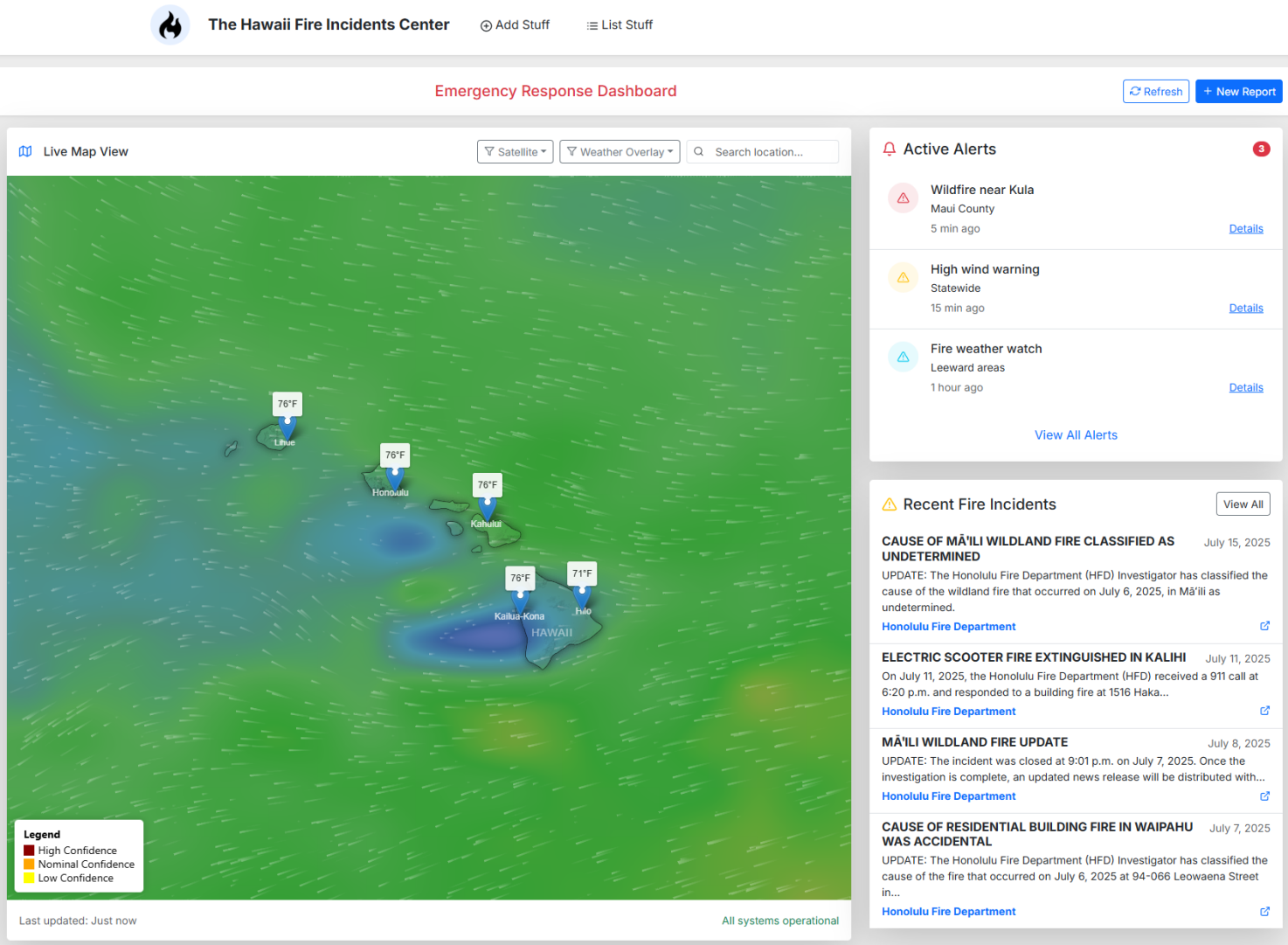


**Figure 4**  Layout overlay with flat layer mode.



**Figure 5** Layout from selecting a point and the overlay for data

## Finalized Dashboard



**Figure 6** Actual design layout for dashboard

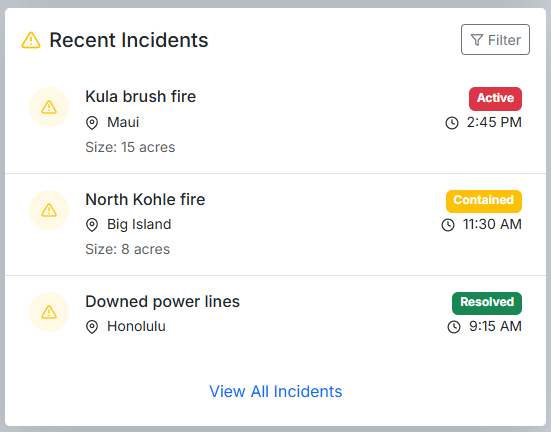
## Alert Form

A screenshot of a computer

AI-generated content may be incorrect.

**Figure 7** Alert Creation Dashboard

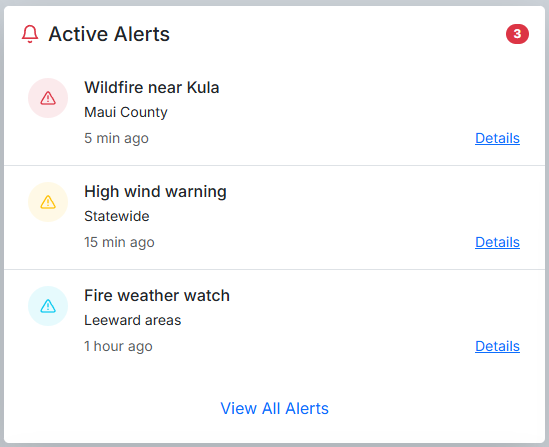
## Incident and News Page Section



**Figure 8** The bottom left section showcasing recent incidents from pulled reports.

## Active Alerts

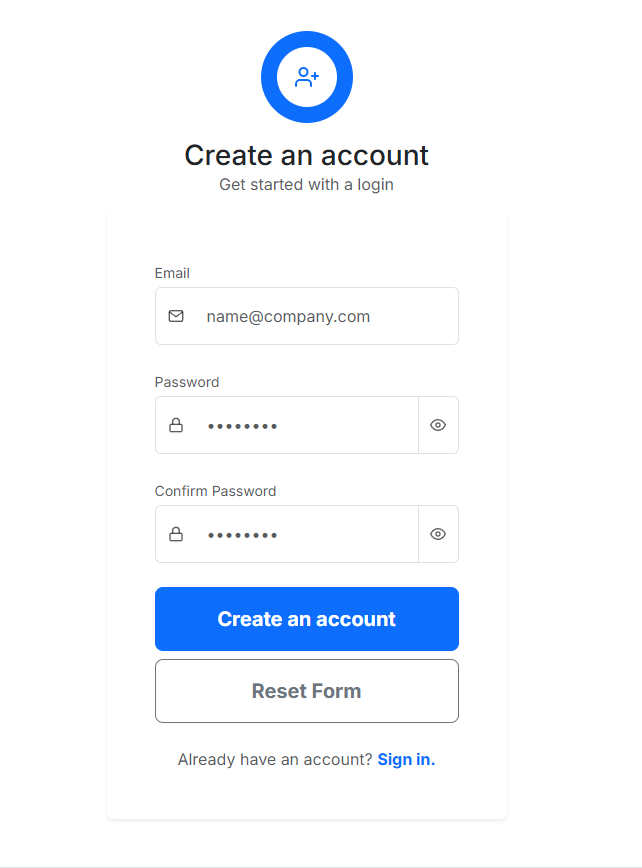
Users can create active alerts manually by using the alert form.



**Figure 9** Right-side alert page listing recently made notifications.

## Login Page

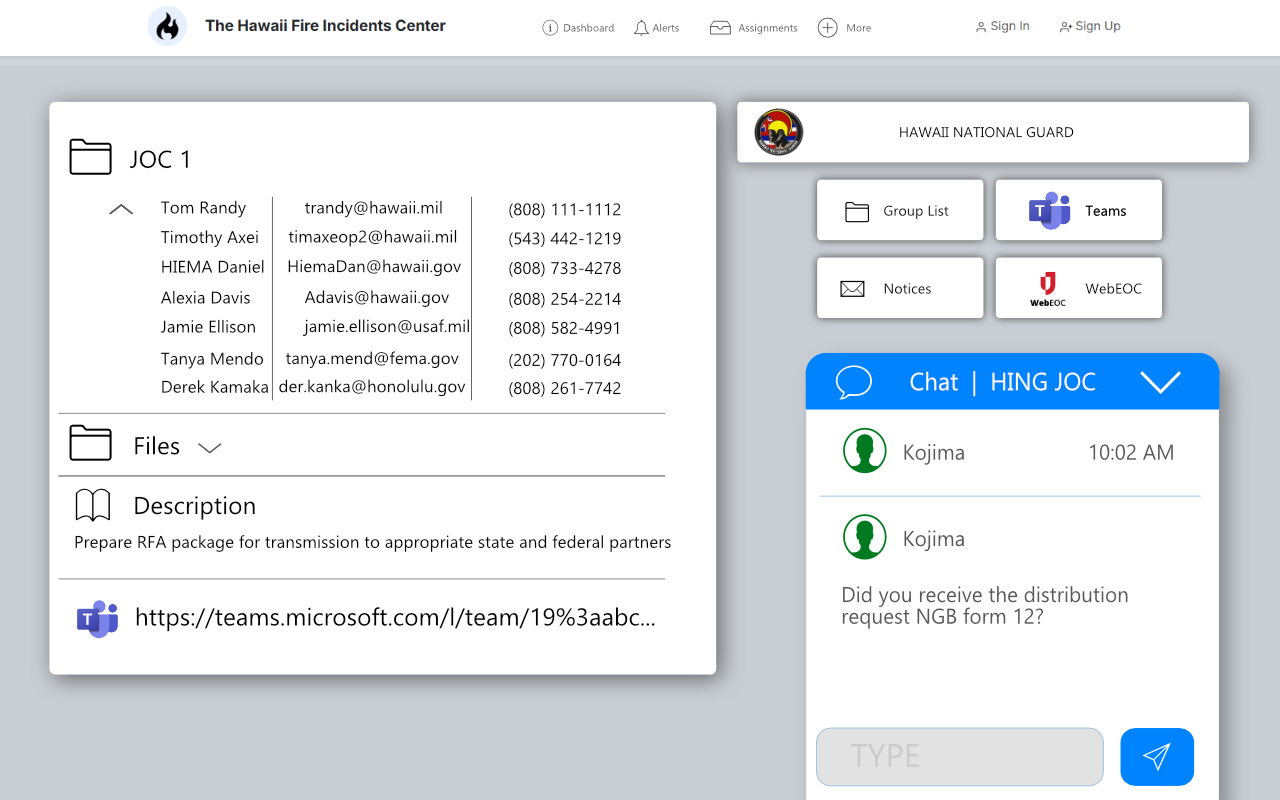
Users will require to make an account to access mission essential features like alert notifications.



**Figure 10** Login Portal

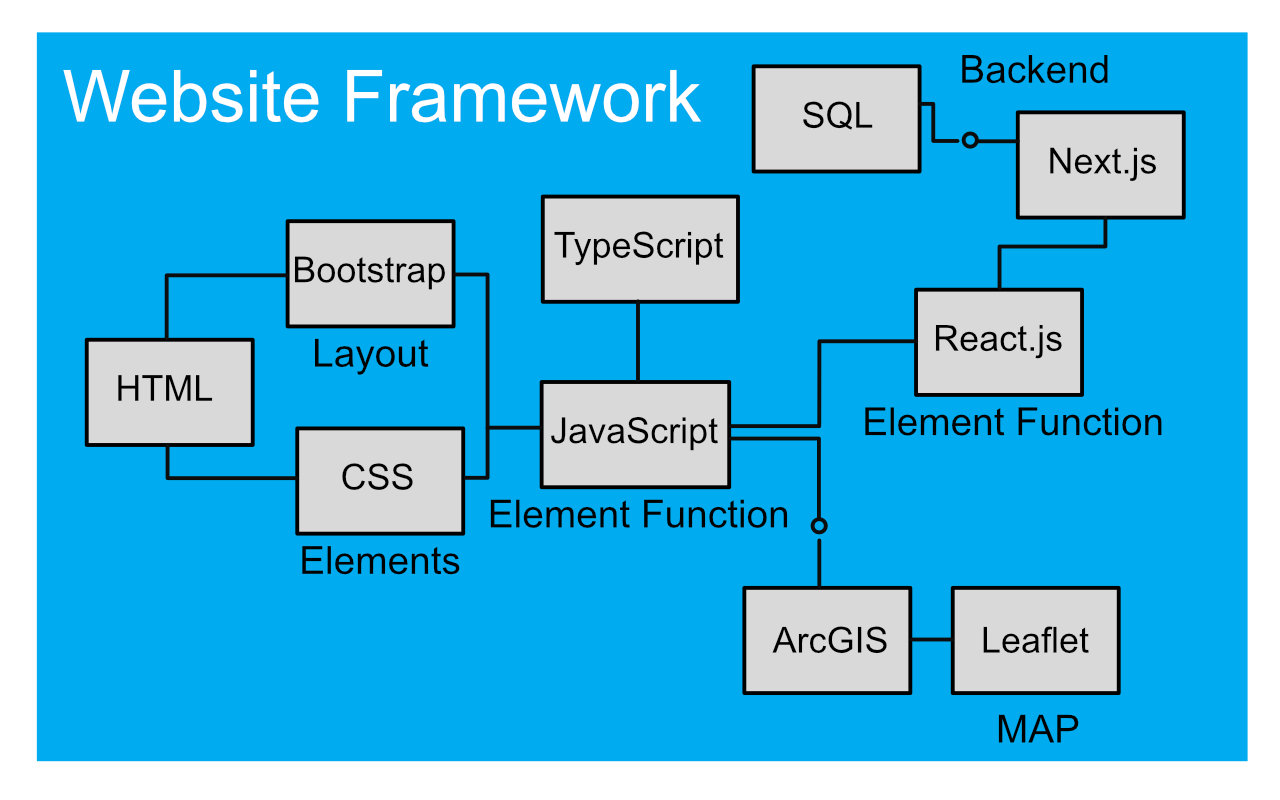
# Planned Features

A popup bar for chatting, page for team/group assignment, shared resources, and assigned Microsoft Teams groups.



**Figure 11** Assignment page for each assigned group following WebEOC format for group folder assignment, includes staff and contact details, a shared resource folder for each assigned group, a description for group mission objective or task, and an assigned Microsoft Teams link. Additionally, a chat pop-up for communications between members and buttons for web pages and organizational resources.

Website Framework

****

**Figure 12** The diagram shows the respective programming languages, libraries, and packages that will be required in the front-end and back-end of the project.

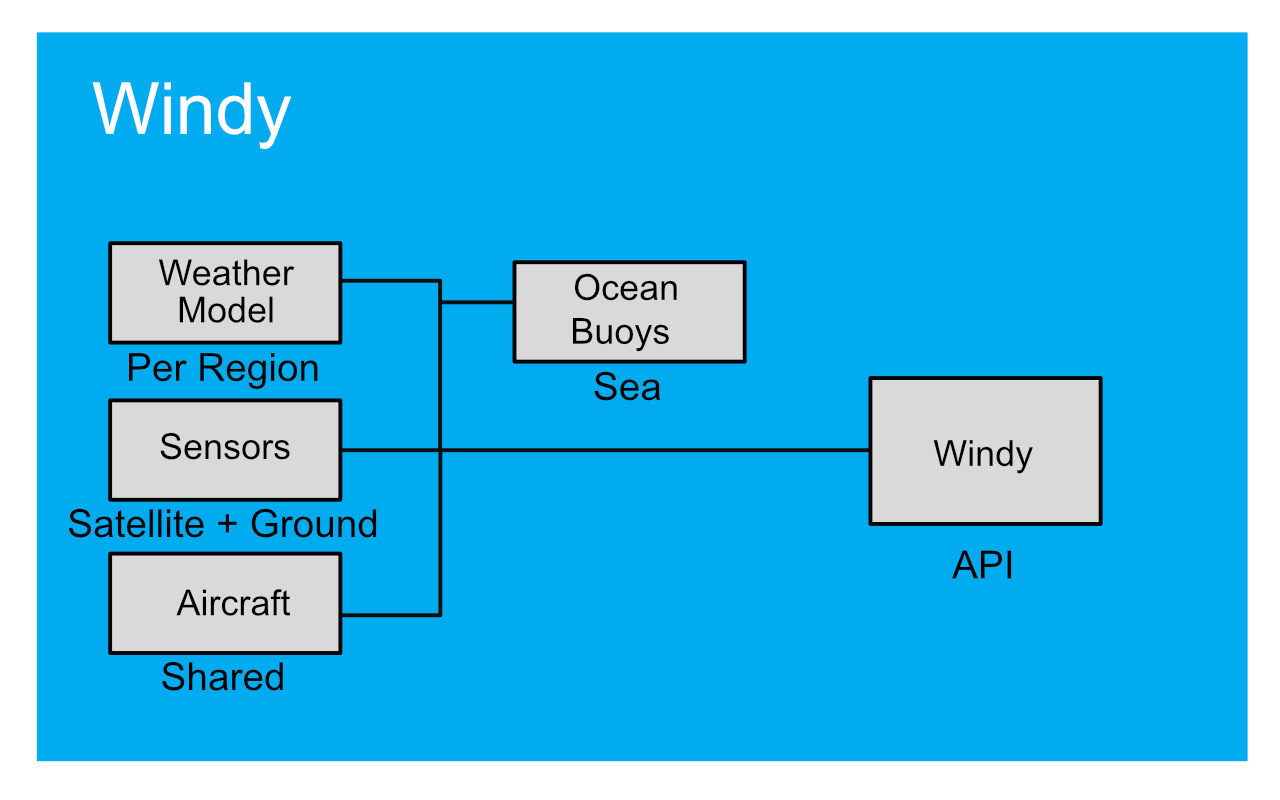
# API Framework

**FIRMS API**



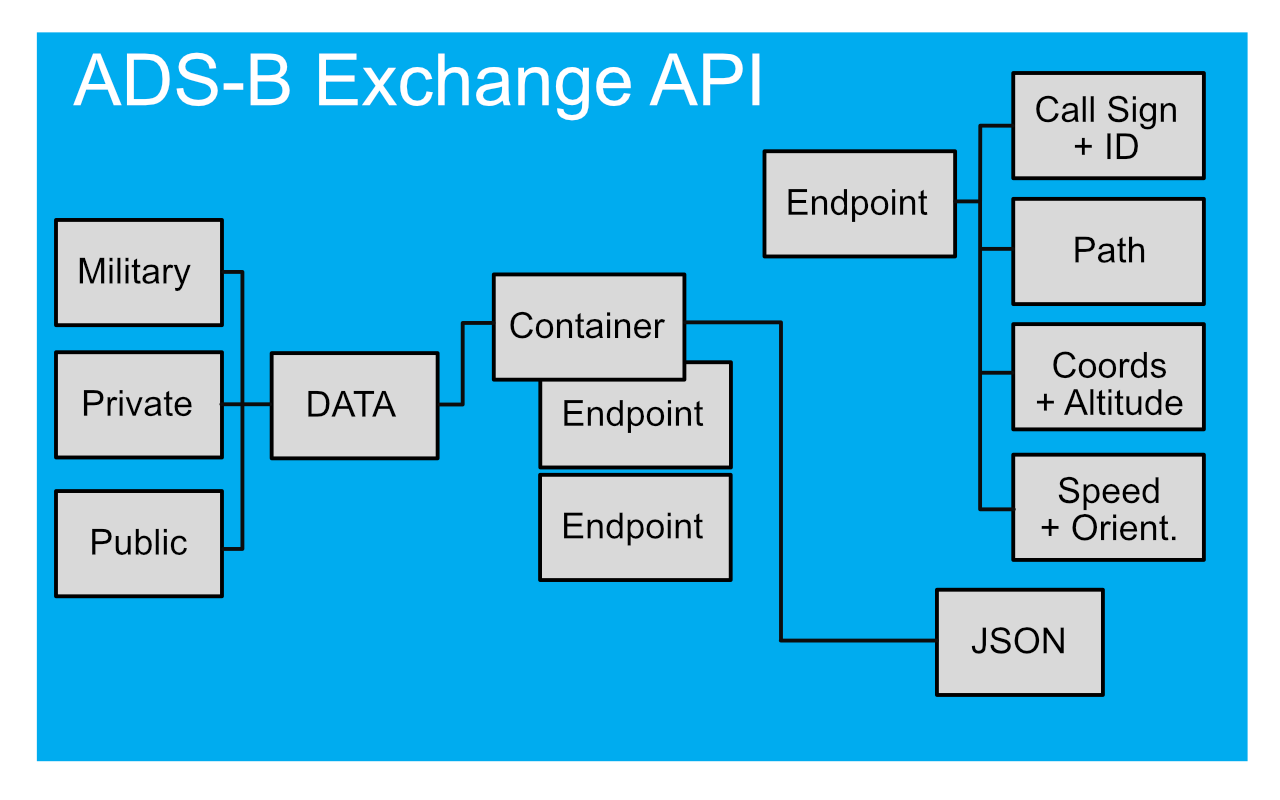
**Figure 13** NASA FIRMS API diagram

**Windy API**



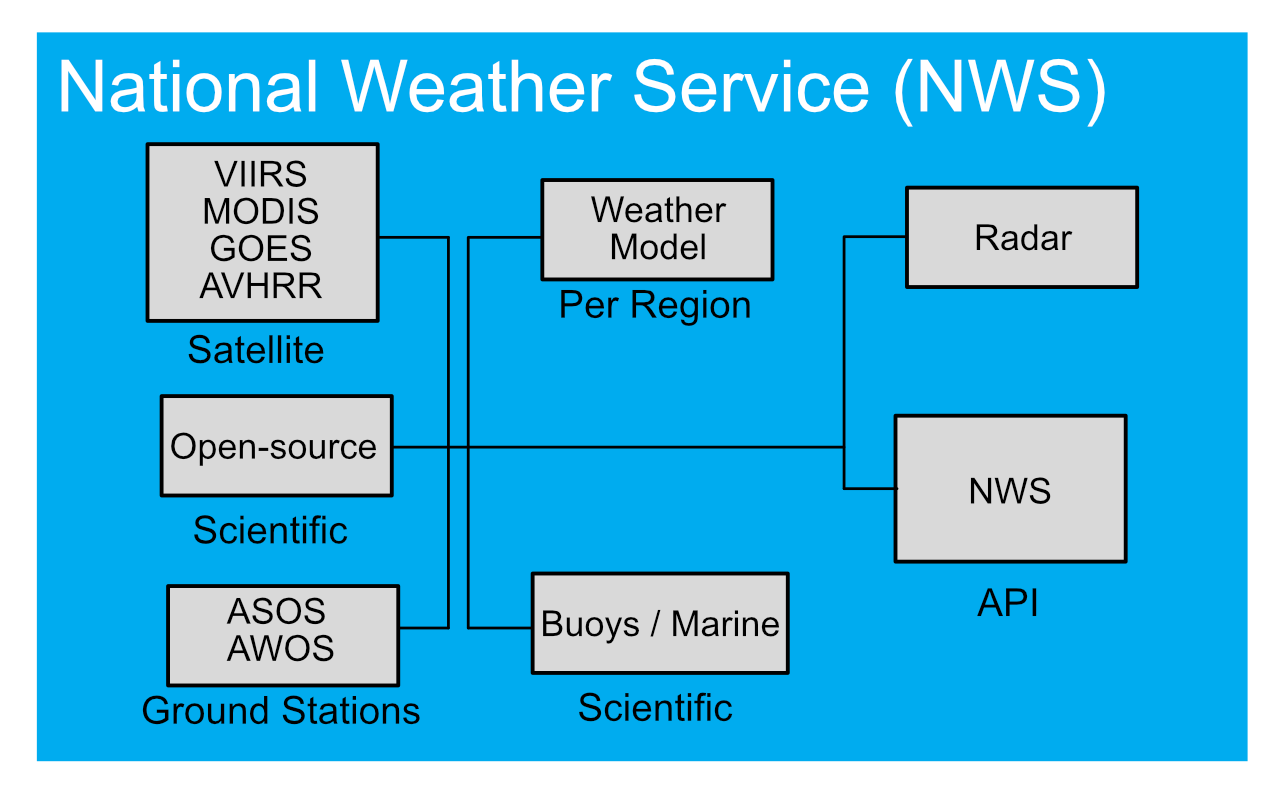
**Figure 14** Windy API diagram

**ADS-B Exchange API**



**Figure 15** ADS-B Exchange API diagram

**NWS API**

****

**Figure 16** National Weather Service API diagram

# Data Sources

* Fire Guard
* WebEOC
* Fire Department (all counties)
* Police Data (all counties)
* County activity and indicators
* Flight Trackers
* Social Media

## Weather, Air Quality, Drought, and Fires

AIRNOW

* <https://fire.airnow.gov/#8.89/21.3814/-157.6607>

NOAA FIRMS

* <https://firms.modaps.eosdis.nasa.gov/map/#d:24hrs;@-157.89,21.41,10.53z>

NOAA Worldview

* <https://worldview.earthdata.nasa.gov/>

NOAA NIFC MODAS, EOSDIS

* <https://firms.modaps.eosdis.nasa.gov/usfs/map/#d:24hrs;@-100.0,40.0,4.0z>

Climate Trace

* <https://climatetrace.org/>

FEMA - national risk indicator

* <https://hazards.fema.gov/nri/map>

Windy

* <https://www.windy.com/-Waves-waves?waves,19.124,-154.089,7,m:eF1auc>

ALERTWest

* <https://alertwest.live/>

Drought

Drought monitor

* <https://droughtmonitor.unl.edu/>

Drought.gov

* <https://www.drought.gov/>

## Other

NOAA

* <https://www.weather.gov/>

Resourcewatch

* <https://resourcewatch.org/dashboards/climate>

OpenData Hawaii Gov

* <https://opendata.hawaii.gov/>

Hawaii GIS

* <https://geoportal.hawaii.gov/>

NASA worldview

* <https://worldview.earthdata.nasa.gov/?v=-161.58038707413584,16.861705037718462,-153.00129968484953,23.26341001358357&t=2025-06-21-T16%3A40%3A35Z>

## Aerial Tracking

ADS-B

* <https://www.adsbexchange.com/data/rest-api-samples/>

Open-Sky

* <https://openskynetwork.github.io/opensky-api/>

## Scraped Content

**Oahu**

Fire Department:

* <https://fire.honolulu.gov/>

Police Department:

* <https://www.honolulupd.org/>

**Maui, Lanai, and Kahoolawe**

Fire & Police Department

* <https://www.mauicounty.gov/>

**Hawai'i / Big Island**

Fire Department:

* <https://www.hawaiicounty.gov/departments/fire>

Police Department:

* <https://www.hawaiipolice.gov/>

**Kauai**

Fire Department:

* <https://www.kauai.gov/Government/Departments-Agencies/Fire-Department>

Police Department:

* <https://www.kauai.gov/Government/Departments-Agencies/Police-Department>

# Security, Maintenance, Reliability Overview

# **Security**

Depending on data storage decisions, this project will contain sensitive data or will be used privately for government use, there will be some items that will be considered for this project.

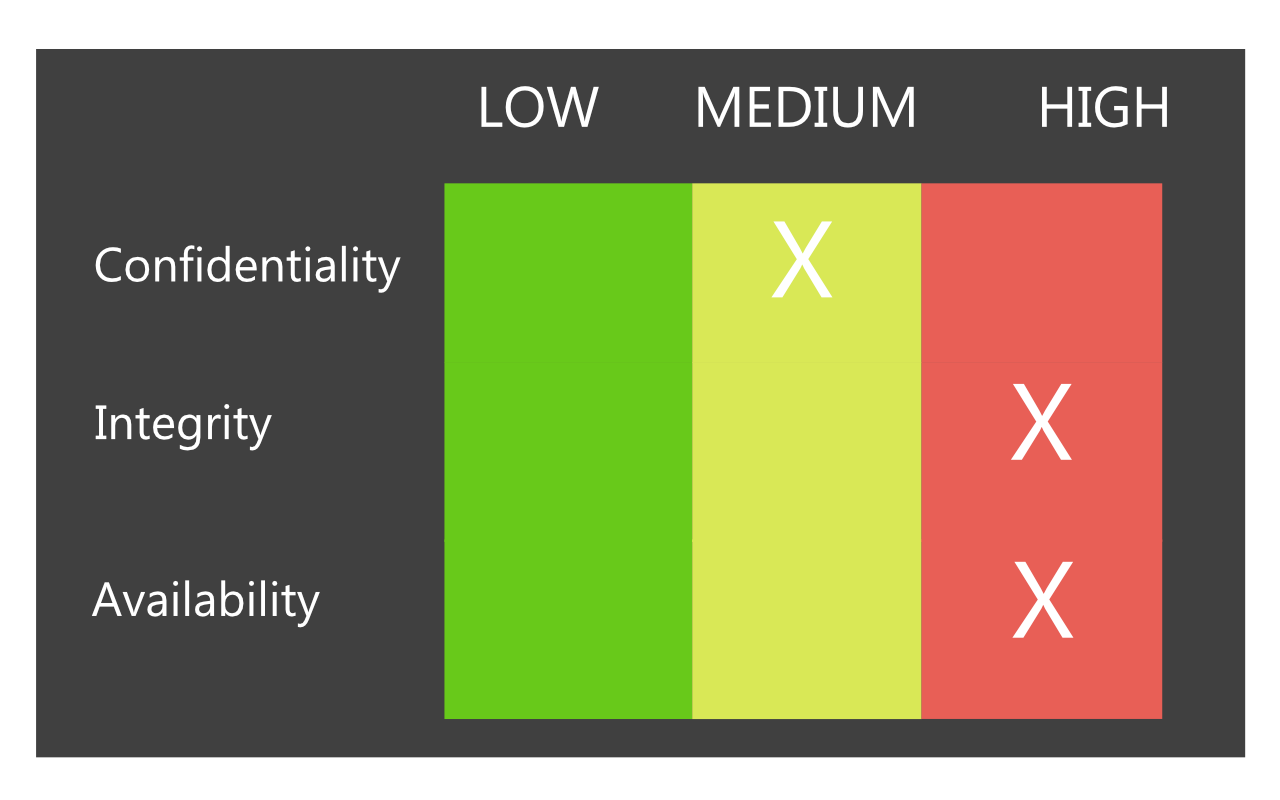
Sensitive data such as PII will contain personal information from reports, camera correspondence, and contact for emergency, state, federal, and military purposes. Additional information flow will also contain weather and scientific data locked to third parties locked to certain use cases.

Based on the CIA triad, Confidentiality, Integrity, and Availability will explain information

security principles for business operations and risk.

Due to the mission objective of this project, it is important that the reports and data are strictly untouched or unaltered.

As with the availability of this project. There could be threats due to DDoS, malware, etc. so it is imperative to keep this project backed up, have alternative means of hosting, and constantly manage for any changes to API structures or security concerns.



**Figure 17** CIA Triad - securiy heat map.

## Maintenance

This project will require maintenance (at minimal) for API key access upon expiration and due to technical errors such as revoked tokens. In addition to adding new API keys for new API and services added in the future.

There might be additional efforts in filtering for alert notifications and clearing or correcting past manually added alerts.

## Reliability

The map for this project will pull from three to four sources of data from both federal and community and scientific-driven sources. If one of the sources is down the project will include other resources to related services. The majority of these sources of data are real-time and are frequently updated.Recommended Hardware Specificaitions

The web application may require the lists hardware requirements to ensure that operation of the software is run smoothly.

**Minimum Requirements:**

|  |  |
| --- | --- |
| Component | Specifications |
| CPU | Intel Core i3 (8th gen) OR  AMD Ryzen 3 (3000 series) |
| RAM | 4 GB |
| Storage | 60 GB HDD or SSD |
| GPU | Integrated Intel or AMD |
| OS | Windows 10, macOS 10.13+, or Ubuntu 20.04+ |
| Browser | Latest version of Chrome, Firefox, or Edge |
| Network | 10 Mpbs broadband |

**Recommended Requirements:**

|  |  |
| --- | --- |
| Component | Specifications |
| CPU | Intel Core i5 (10th gen) OR  AMD Ryzen 5 (3000 series) |
| RAM | 16 GB |
| Storage | 256 GB SSD |
| GPU | Integrated Intel or AMD OR Dedicated AMD Radeon or NVIDIA GeForce |
| OS | Windows 10, macOS 10.13+, or Ubuntu 20.04+ |
| Browser | Latest version of Chrome, Firefox, or Edge |
| Network | 25+ Mpbs broadband |

# Lines of Effort

* Determine API compatibility
* Determine Data Sources
  + Access to Data Sources
    - Authentication and Authorization
      * API Keys
      * Database

**Line 1: Access + Data**

* Way points (RAPIDS, FIRMS, 911/State, WebEOC, ARCGIS, NGB COP)

**Line 2: Clarity of data points**

* Waypoint placement (amount)
* Dashboard information

**Line 3: API Inter-compatibility**

* Leaflet, Windy, FlightRadar24, BreezoMeter
* Layers, Filters, Toggles

**Line 4: Security**

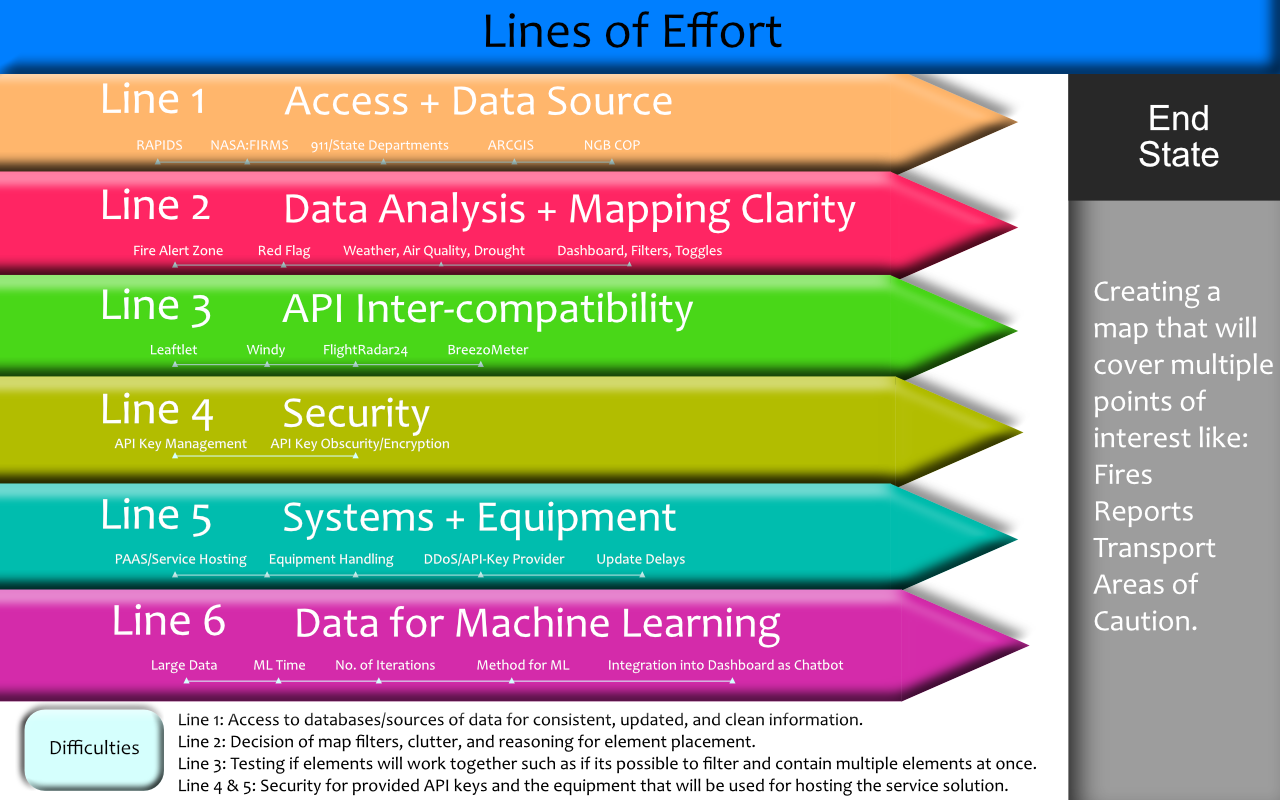
* API Keys (encryption)
* Data security
* SQL database for login

**Line 5: Systems + Equipment**

* PAAS/Hosting, Domain
* Equipment handling (Specifications)
* Security: DDoS, API-key provider, encryption
* Regulations with sensitive data storage
* Update Delays

**Line 6: Data for ML & Waypoints based on high/possible risk**

* Importing multiple datasets
* Learning time for multiple cycles
* Multiple iterations/models for data output
* LLM and ML methods for consideration
* Integrate into a API for dashboard as chatbot



**Figure 15** Lines of Effort - reference to military approach to milestones.

# Installation Guide

Recommended Tools:

Integrated Development Environment (IDE) like Microsoft VS Code

**Website**

***Prerequisites***

Python 3.8+

Node.js + npm

PostgreSQL

***Bootstrap***

npm install bootstrap

***Next.js***

npx create-next-app@latest my-app

cd my-app

***React.js***

npm install react react-dom next

**Map**

***Flask***

python -m venv venv

source venv/bin/activate

pip install Flask Flask-Cors Flask-SQLAlchemy psycopg2-binary

***Leaflet***

npm install leaflet react-leaflet

**Web Scraper Notification**

***Python: BeautifulSoup + requests***

pip install beautifulsoup4

pip install requests

# Host Server Requirements

|  |  |
| --- | --- |
| Component | Specifications |
| CPU | 8-16 Core Intel Xeon Silver OR AMD EPYC |
| RAM | 64-128 GB |
| Storage | 1-2 TB NVMe SSD |
| Bandwidth | 1 Gbps dedicated uplink |
| OS | Windows Server 2022 or Ubuntu 22.04 LTS |
| Database | PostgreSQL with SSD-backend |
| Security | Hardware firewall, DDoS protection, SSL, intrusion detection |
| Backup | Daily automated backups with offsite redundancy |
| Virtualization | Docker or Kernetes support |
| Compliance | FedRAMP or NIST 800-53 |

# Schedule For JOC HING Firewatch Project

Schedule For Weeks 1 - 7

*Week 1: June 23 - 26*

**Framework Planning**

* Learn the programs NGB COP , WebEOC & ARCGIS
* API access and learning
* API compatibility
* Access to State Databases
  + Police Reports
  + Fire Department Reports
  + Access to data from all counties in Hawaii
* Base map planning
  + Filters
    - Air Traffic (Helicopters) – identifiers, response type
  + Air Quality, drought, and current fires
  + Possible ETA for emergency services

*Week 2: June 30 – July 4*

**Sources Input Review**

* Review of sources
  + From NASA, State, and government entities
  + Review delayed update time frames, and currently updates sources
  + Access to State data (mostly that are by request only)
* Compatibility of information
  + Overlays
  + Filters
  + Website layout and design

*Week 3: July 7 - 11*

**Implementation and Design**

* Inter piecing the APIs to the desired selected software solution
* Determine which are compatible and tweak performance from pulled data
* Cycle and prototyping
  + Graphic Design Layout for expected outcome

*Week 4: July 14 - 17*

**Security, Delays, Problems, and Backup Plan**

* Determine Security gaps
  + APIs, DDoS, Service Delays, Server load-balancing
* Delays
  + Timed delays due to data pulls
* Problems
  + Create a problem statement
    - Delays in coding
    - Problems in API keys
    - Problems in API compatibility
    - Complications from the proposed software solutions for end-product
* Backup plan
  + Find development need to work on the second proposed software solution if the first does not meet expectations
  + Proposed time frame for development on this system

*Week 5 & 6: July 21- 30*

**Finalize & Tweak**

* Make tweaks to coding structure
* Tweaks for graphical representation
* Prepare problem statement

*Week 6: July 28 - 31*

**Prepare Presentation**

* Make slideshows
* Snapshot final-end solution
* Prepare showcase for prototype
* Relate to other projects and current solutions in the market or currently being used
* Showcase how project improves

Address:

* Mission Objective
* Input Sources, Output Sources
* Scope of Project
* Organization
* Supporting Members

# Proposed Budget

## Tracking

**FlightRadar24** - $900 / month | $10,800 / year

Cancel anytime

Need: Due to limitations of other flight tracking APIs, this API would give results of not only public but private and military based aerial vehicles that are easy to implement with this API.  
In addition FlightRadar24 provides instant API Key access instead of requesting which may takes days or weeks to arrive.

**Windy** - $2.08 / month | $24.99 / year

Cancel anytime

Need: This API has multiple elements that can feed through data such as satellite, radar, wind, temperature, clouds, air and UV, and drought.

**AeroAPI** - $ 1000 / month | $12,000 / year

## Hosting

**DigitalOcean** - $24 / month | $288 / year

Need: A place to host the web application. Bypass the process and procedures from hosting on-premise.

|  |  |  |
| --- | --- | --- |
| Solution Name | Cost Per Month | Cost Per Year |
| FlightRadar24 | $900.00 | $10,800.00 |
| Windy | $2.08 | $24.99 |
| AeroAPI | $1000.00 | $12,000 |
| DigitalOcean | $24.00 | $288.00 |