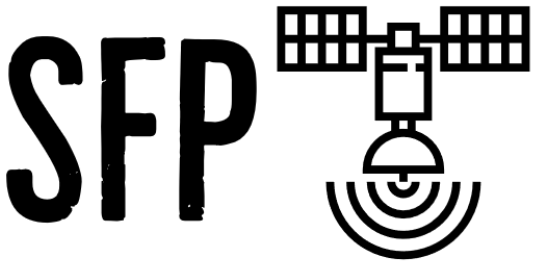
Team Satellite Fire Patrol

Requirements Specification v1.0
October 21, 2023



Team Members:

Aaron Santiago, Ricardo Chairez and Zachary Hallemeyer

Clients:

Benjamin Wiebe, Dr. Camille Gaillard, and Dr. Christopher Doughty

Mentor:

Saisri Muttineni

Accepted as baseline requirements for the project:

For the Clients:

Benjamin Wiebe

Camille Gaillard

Christopher Doughty

For the Team:

Aaron Santiago

Introduction

As climate change's threat becomes ever present in the modern world, its effects are becoming more prominent and difficult to dismiss. For instance, the Hawaiin wildfire, in August 2023, resulted in monumental loss in human life as well as billions of dollars in infrastructure destroyed as well. As climate change worsens, natural disasters such as the before mentioned will become harder to predict and increase in both intensity as well as frequency.

Furthermore, the effects of climate change also create vulnerabilities in various ecosystems such as tropical forest canopies and coral reef systems. This is because both forest canopies and coral reefs have a certain temperature threshold, and if the threshold is passed the ecosystem may start dying. Because of this, solutions must be developed that are able to more effectively predict and alert people of warning signs in their environment before it is too late.

In order to warn people about possible warning signs, we must leverage temperature data collected with satellites. If the temperature data is able to be processed and displayed, users may be able to identify if an area is vulnerable to the damaging effects of climate change, as well as perform necessary precautionary measures to help remedy the vulnerable area.

Our clients, Benjamin Wiebe, Dr. Camille Gaillard, and Dr. Christopher Doughty who are professors and researchers at NAU, have identified this problem and have sponsored a solution to create an open source warning system that can be used by anyone including individuals as well as companies and local governments. The goal is not to solve the problem of climate change's effects, but rather provide a resource that interested individuals can use to be alerted when a given custom area's temperatures exceed a custom threshold.

Our clients work with temperature data to identify vulnerable areas in forest canopies, and the solution developed will be used by the clients to help them with their research. However, the goal is for the solution to be used by anyone that can be helped by said software. For this reason, the solution must be flexible, and the resulting solution will be invaluable to organizations attempting to predict natural disasters brought on by increasing world temperatures.

Problem Statement

The current problem with identifying vulnerable areas is that there is no readily available software to both display surface temperatures of the world with the latest data. While specialized software can be developed for a certain project, the issue still remains that other projects do not have access to the said software and will have to develop the entire software on their own. This may be out of scope for projects, which results in a lack of effectiveness.

To be more specific, a project focusing on ocean bleaching will have to create their own software to identify ocean temperatures that exceed coral's temperature limit. If the project does not have the resources to create this software, the project will have to physically retrieve the temperature data of the ocean.

Another instance of this issue rears itself with the capturing of canopy heat in forests. The data is collected, it is just not understandable or easily accessible to the public. Climate change is causing extreme temperatures and extreme weather conditions. With these extreme temperatures, wildfires are more likely to occur and other catastrophic events. Wildfires like the one that occurred in Maui, Hawaii, earlier this year were caused by extreme temperatures and the lack of an early warning system to detect those temperatures.

Sensitive Ecosystems are especially vulnerable to climate change, and wildfires can be mitigated by preventative measures for specific areas. Wildfires often start at the same time in multiple locations because of similar temperature conditions, so to save these environments warning people about these red flag temperatures in these sensitive areas can help prevent fires.

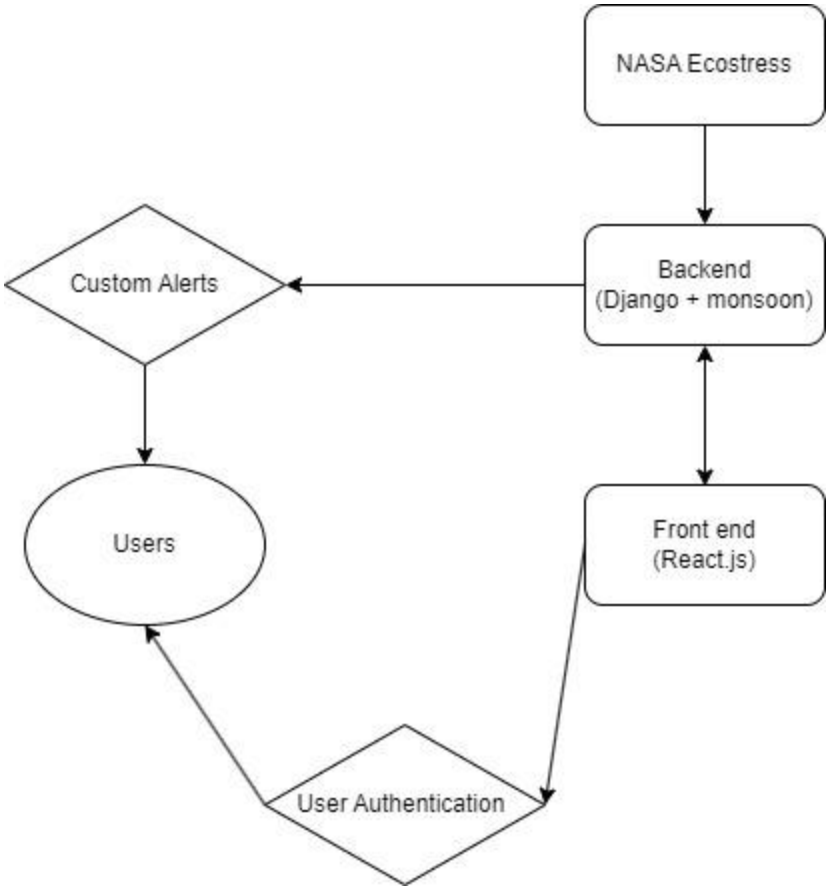
Solution Vision

As the problem has been explored above, a solution may now be developed. Team Satellite Fire Patrol will create a web application that will take near real-time thermal data and display it for users on top of an interactive map of Hawaii. This web application will be available to the public with the project being open source for further development and specialization. With the thermal data being displayed, users will also be able to create custom alerts which alerts the user when their custom region exceeds a custom thermal threshold. Thermal data will be retrieved from NASA Ecostress and stored on a custom database.

Key features:

- Overlayed Thermal Data
    - Retrieve thermal data from satellites, process, and store said data
    - Display thermal data overlaid on top of an interactive map
    - Display historical thermal data from the beginning of data collection from NASA Ecostress
- Custom Alerts
    - Users will be able to create custom regions by drawing polygons on an interactive map. The users will then be able to assign custom temperature average thresholds for that area. If the region exceeds said threshold, the user will be alerted with sms messaging and emails.
- General
    - User accounts

The system will extract and process thermal data, then transmit and store it in a dedicated location. Both the front-end and back-end data will be stored together in the same location. The diagram below shows the general system architecture:


NASA Ecostress
Backend
(Django + monsoon)
Custom Alerts
Users
Front end
(React.js)
User Authentication

Project Requirements [1 - 2 PAGES]

EXPLORE THE TECHNICAL REQUIREMENTS AND SOFTWARE/APIS USED

1. A web application that provides visualization of near real time land surface temperature data anomaly compared to historical averages.
2. A system for users to create an account, set up alerts for when a specific area reaches a certain temperature threshold, and receive alerts.
3. A system that processes historical and real time land surface temperature data and stores it.

Functional Requirements

1. **Users will be able to view near real time land surface temperatures of the Hawaiian islands.**
    a. Users can view a map of the Hawaiian islands. This map will be displayed through the MapBox framework. The map will be limited to the Hawaiian islands only.
    b. Colored overlays on the map of Hawaii will represent temperatures in certain areas.
    c. A legend will be displayed next to the map to signify what temperature the color represents.
    d. Numerical Fahrenheit and Celsius temperatures of a given area will be displayed.
    e. Map will update in real time as more recent temperature data comes in.
2. **Users will be able to view historical land surface temperatures of the Hawaiian islands.**
    a. A slider will be implemented for users to view the change in temperature over time. The slider will go as far back as our data goes.
    b. Users will be able to enter a specific date and view the land surface temperature of that time.
    c. Selecting a date will update the map in real time.
3. **Users will be able to create an account through the website.**
    a. If the user is not signed in, a create an account button will appear on the website.
    b. Accounts will be created through the Google Accounts API.
4. **Users will be able to log into an existing account with the following information:**
    a. Google Email
    b. Password
5. **Users will be able to set up custom alerts with the following parameters:**
    a. Region of the island
    b. Temperature threshold

6. **Users will receive alerts via SMS/email when the system detects the custom alert conditions have been met.**
    a. Users will receive SMS alerts when temperatures are excessive
    b. Users will receive SMS alerts with a link to access and view the map
    c. User notifications will default to SMS but have email notifications for backup

Non-Functional Requirements

Ease of Use
Our web application must be user-friendly and easy to use for users of any kind. In the future, our team will create a small walkthrough that walks new users through the website once they have logged in. The application will need to be easy to use to allow as many people as possible to take advantage of its features and alert system users.

Speed
The web application will be relied upon by users to determine if environments are in danger and because of this urgency our system must be fast. Our system must function and complete its task of displaying data within seconds. The alert system must work quickly because if there is the possibility of a fire occurring, time is of the essence. To accomplish this goal, we will attempt to make our scripts on NAU's monsoon work as quickly as possible.

Data Verification
To avoid the occurrence of needless false positives for temperature warnings, we will need to ensure that the data we receive from NASA's ECOSTRESS is correct. Incoming data will be optimized to ensure that the data we receive can be properly compared against historical averages. By cleansing the data we will have much more accurate comparisons, and we will reduce the chance of false positives occurring and alerting users when an alert was not necessary.

Scalability
Our clients have expressed interest in expanding this project to work outside of Hawaii. The goal of our project is to create code that is easily scalable to potentially work in a worldwide setting. To accomplish this we will be using scalable frameworks and systems for our project to allow for the expansion of this alert system to be used in a worldwide setting. Our data management must be incredibly flexible and able to handle and display data from multiple locations.

Environmental Requirements
While there are not many environmental requirements desired by our clients, our web application must use NAU's Monsoon for data storing and processing as the clients want the data to be stored at NAU.

User accounts including personal information must be protected in order to comply with local and international laws. Alongside this, use of cookies must be brought to the users attention if cookies are used in the final version of the web application.

Finally, NASA ecostress data is publicly available, so the data that the web application will be using is legal.

With the base requirements of our web application having been explored. We may now explore the risks associated with our approach and requirements.

Potential Risks

Our project is acting as a warning system that will eventually be released for use in the public. Our risks in this project are as follows.

Temperature Calculation

If we fail to implement a reliable calculation of average temperatures for the alert system, then the system will never alert users who are interested in monitoring sensitive environments. The system must calculate and compare the historical average temperature of an area reliably, so it can determine whether to alert the users. If this mistake occurs and the product is already released to the public, the results could be devastating. If users rely on the web application for fire preventative measures, then a potential fire could go undetected for longer than it needs to be, which could cause infrastructure damage or permanent damage to sensitive ecosystems. Worst case scenarios include loss of life and infrastructure damage if our product is going to be used as an early warning system for firefighters.

In addition, if users rely on the information from the web application to perceive if there is risk for a fire, then they could be given a false sense of security if our temperature calculations are off. This false sense of security is dangerous because it can cause people to act more dangerously. In our case, we could inadvertently cause a forest fire by giving a user false information, letting them believe they can do something like fireworks when it's a really dry season.

Alert Errors

When our product is up and running, it will potentially be used by those interested in wildlife conservation. If our alert system is not reliable or able to alert users within at least an hour of the out of average temperature, then possible environmental effects/data can be missed. If users are

using our web application as an alert system to get data from these environments when excessive heat is occurring, then important scientific information can be missed out on if our system is not reliable.

Competitor Product

Since our product is open source, we are expecting competitors to arise after the initial release from our product. The hope is that potential developers would rather invest in the use of our product than create their own. If a competing product is created, then our project will hopefully stand out with its simplicity, price (free), and openness since everything we create will be readily available.

Project Plan

Through the rest of 2023 into early 2024 the team plans on working on this project with multiple milestones along the way. Our current plan is to have a working prototype by the end of the semester for Fall 2023. The prototype will include a working web page and hopefully having it connected to NAU's monsoon. The hardest part of this project will be connecting the monsoon database to the web application using ITS' RD port. Using the prototype, we will build upon it in the upcoming semester to make the features as efficient as possible.

Milestones

By the end of the Fall 2023 semester, we will have a working prototype and complete most of the design aspects of the project. Every week, the team will be working on the design requirements and overall architecture of the project. We will also be making adjustments to the requirements as the project develops to create a working product for our clients that fulfills their needs.

Fall 2023 Capstone

By the end of this semester, we will have completed the design aspects of this project, which include the assignments listed below.

- **Requirements Specifications Document:** The team will evaluate the requirements of the project which include the functional, non-functional, and environmental requirements for the project. The document will also include the possible risks and the team's schedule for development to ensure the team stays on track.
  Completion Date: November 2nd, 2023

- **Design Review:** The design review will be a short presentation to the capstone class. It will be covering a summary of this current document. Presenting to the class allows the

team to receive feedback from the whole class. The team as a whole will understand the project, and so will the capstone class.

Completion Date: November 10th, 2023

Spring 2024 Capstone

This semester will include the bulk of the software development process. We will be splitting our development process into two parts. The first part will focus on the backend design of the project. For our backend design, we will be setting up NAU's monsoon to manage and organize the data from NASA's ECOSTRESS in a database in a way that is easy to access from. The second part of our project will focus on the Frontend of our project by creating an intuitive and beautiful design.

Data Organization ~ January 2024

We will be using Python and the Django framework to successfully manage and organize the data in the Monsoon server. We will be creating scripts that automatically request the data from the NASA satellite and process it to determine if the temperatures we are currently viewing are outliers.

Alert System ~ February 2024

We will be relying on an API to function for our alert system. The script for this will be included in the backend of development, so it can notify people as soon as possible.

User Authentication ~ February 2024

Instead of managing usernames and passwords, we will be incorporating google account's API to ensure users can easily log in and access our web application.

Web Application ~ February 2024 – May 2024

The web application is a crucial part of the project because it will be the part that users interact with. Users will be able to request data and control which regions they want to monitor through the web application. Our website will be using Mapbox AI because of how easy it is to implement our data with it. We will be showing historical temperature data and current data on the website on a 2D/3D map that has an overlay for the temperatures.

Conclusion

As climate change's effects are continually increasing in intensity, it is more important than ever that researchers and organizations have tools that can track the effects. Along this logic, software

available to the public that is capable of displaying and working with near real-time thermal data is necessary and with no open source software capable of this, the software must be developed.

The web application that team Satellite Fire Patrol will develop will be able to display thermal data, but there will also be tools to help researchers and organizations to mitigate the losses brought on by climate change by enabling users to set alerts on when a region exceeds a temperature threshold.

This web application will be developed with a variety of software. The system architecture will use NAU's Monsoon for retrieving, processing, and storing thermal data from NASA's Ecostress API as well as storing user accounts and user details such as near real-time alerts. The web application will use Django and Vue as frameworks for the web application's back end and front end respectively. the application will be deployed on NAU servers. Finally, the thermal data will be overlaid on a map API called mapbox.

There are potential risks associated with the development of this project. As it stands, the only consistently available thermal data comes from NASA Ecostress. This means that we are unable to perform quality control by cross-referencing thermal data with other readings. While quality control will still be performed by using historical averages, there is risk for false positives or false negatives in regard to alerts, which need to be addressed as more development continues.

The web application will be developed over time by designing the system architecture and deciding on frameworks and APIs. Then prototyping will begin by attempting the necessary functionality of the system with the decided frameworks and APIs. Then when the functionality has been proven, the system will come together by working on each component in parallel and connecting them with the base functionality. Finally, stretch goals and quality of life features will be implemented if time allows.

In the culmination of the capstone course at the end of May 2024, a web application that is capable of displaying thermal data and sending custom temperature alerts will be completed and be available to the public.