THE COMMUNITY WILDFIRE PROTECTION PLAN REPOSITORY:

USING VGI TO CREATE A NATIONAL COLLECTION OF WILDFIRE MANAGEMENT
INFORMATION

by

Matthew Bauer

A Thesis Presented to the
FACULTY OF THE USC DORNSIFE COLLEGE OF LETTERS, ARTS AND SCIENCES
UNIVERSITY OF SOUTHERN CALIFORNIA
In Partial Fulfillment of the
Requirements for the Degree
MASTER OF SCIENCE
(GEOGRAPHIC INFORMATION SCIENCE AND TECHNOLOGY)

December 2023

To my father

Acknowledgements

Table of Contents

List of Tables

List of Figures

Abbreviations

AGOL	ArcGIS Online
AJAX	Asynchronous JavaScript and XML
API	Application programming interface
AWS	Amazon Web Services
CI/CD	Continuous integration/continuous deployment
CSS	Cascading style sheets
CWPP	Community wildfire protection plan
FACNET	Fire Adapted Communities Learning Network
GIS	Geographic information system
HFRA	Healthy Forests Restoration Act
HTML	HyperText markup language
I/O	Input/output
NF	National forest
NP	National park
OSU	Ohio State University
PaaS	Platform-as-a-service
PDF	Portable document format
PSU	Portland State University
QA	Quality assurance
RDBMS	Relational database management system
RDS	Relational database service
SME	Subject matter expert

SQL	Structured query language
SSL	Secure sockets layer
UI	User interface
URL	Uniform resource locator
US	United States
USC	University of Southern California
USDA	United States Department of Agriculture
USFS	United States Forest Service
UX	User experience
VGI	Volunteered geographic information
WUI	Wildland urban interface

Abstract

Wildland fires ravage the United States every year, and Community Wildfire Protection Plans (CWPP) are one of the main tools used to plan for and mitigate them. CWPPs are a community planning document that outlines the wildfire risk in the relevant jurisdiction and lays out plans on how to mitigate these risks. Currently there is not a standardized, national dataset for documenting them, which is the gap this thesis seeks to fill. The focus of this work is the development of a web geographic information system application and corresponding database, which collects and stores CWPP documentation and maps the extent of CWPP jurisdictions. The project is supported by volunteered geographic information (VGI) input by any users that take the initiative to do so. An automated data pipeline has been developed to expedite the confirmation of user-inputted data accuracy, and a PostgreSQL database has been developed to house all the CWPP data, which came from both an existing Excel Workbook. A prior version of this application was an Esri Storymap hosted on the Fire Adapted Communities Learning Network (FACNET) website. This project developed a new web application, from the ground up using hypertext markup language, cascading style sheets, and JavaScript. Overall, this work expands the availability of national CWPP information for planners and stakeholders, so civilians that wish to view their jurisdiction's plans can do so, and those responsible for writing new CWPPs can easily view historical and existing plans and other resources. The project currently contains CWPPs for 11 western states but its coverage will grow as VGI is inputted via the CWPP Repository. It will be managed by the writer of this thesis and hosted by FACNET. The completed web application can be found at https://cwpp-repository.com/, and the GitHub repository for it can be found at https://github.com/SpatialSolutions93/CWPP-Database-Website/tree/main/WebApp.

Chapter 1 Introduction

In the United States (US), Community Wildfire Protection Plans (CWPPs) are the main tool used to plan for and mitigate the risk from wildfires primarily at a county, community, and fire protection district level. CWPPs are collaborative plans created by communities and agencies, and they typically identify and prioritize areas for fuel reduction, as well as recommend measures that homeowners and communities can take to reduce the ignitability of structures (ODF 2023). The enactment of CWPPs began in 2003 as a result of the Healthy Forests Restoration Act (HFRA)(2003), which incentives the creation of them by making communities that establish CWPPs eligible for federal funding to go towards wildfire mitigation.

Currently, no national database exists of historical and current CWPPs, making it hard for stakeholders to track those relevant to their jurisdiction. Prior research at Portland State University (PSU), to which this author contributed, collected and analyzed CWPPs and shared the results in a StoryMap (Palsa et al. 2022), but it did not set up a standardized, efficient database to hold these plans, nor a simple straightforward process to update them. This thesis project created a database and a web application, called the CWPP Repository, driven by volunteered geographic information (VGI), to house the largest collection of CWPPs in the country. This is intended to serve as a resource for civilians to inform them of wildfire mitigation efforts in their area, and for those in the fields of natural resource and emergency management to view prior plans in their area and resources to assist with the creation of more in the future. Understanding the history of CWPPs in any given area is important because past mitigation measures can help to inform what future mitigation measures should be taken. The development of this web application and database is accomplished using all open-source technologies, making it completely transparent and free of cost other than cloud hosting services.

The research goal is to fill a gap that exists in wildfire emergency planning where more cohesion among entities is desperately needed to improve wildfire resilience in the US. The CWPP Repository provides access to all CWPP PDFs and boundary data, gives planners a greater ability to document the history of CWPPs and access planning resources, and gives concerned citizens a central location to learn about planning and mitigation efforts in their jurisdiction.

1.1 Spatial and Temporal Extent

The study area for this effort is the 11 western states of the contiguous US: Washington, Oregon, California, Arizona, Nevada, Utah, Idaho, Montana, Wyoming, Colorado, and New Mexico (Figure 1). These states were chosen because the western US sustains more damage from wildfires than the east and is expected to see an increase in large wildfires (CISA 2023). The temporal extent is 2003 through 2022, and the temporal resolution varies by plan: some are updated annually, and some are updated every five or more years. The spatial resolution is high, as most plans contain a clear map of the CWPP boundary, and those that do not almost always are defined by an existing jurisdiction, such as a county boundary. The hope is that more users will continue to update the database with new plans, increasing the temporal and spatial extent of the database over time.

Figure 1. Study Area

1.2 Research Benefit

Wildfires caused $81.6 billion in damage in the US between 2017 and 2021, which is ten times the cost of damage that occurred between 2012 and 2016 (CISA 2023). This shows that communities need support with mitigating these wildfire impacts, to prevent damage to homes and businesses, and particularly loss of life.

1.2.1 Community Benefit

Countless communities across the US are under seasonal threat from wildfires, and this work should assist them by preparing stakeholders and subject matter experts (SME) for potential impacts from wildfires. It will spread awareness of mitigation and planning techniques, and it gives users access to a comprehensive history of over 1000 CWPPs that have already been written and approved by their respective governing bodies. In addition to that it compiles

existing literature on wildfire planning and the usefulness of CWPPs. This should enable citizens and SMEs to access state and national level documentation of the history and use of these plans and provide templates for disaster planners. Experts that work in this field will also be able to generate plans more quickly and efficiently for the communities they are responsible for, ensuring their best chance at mitigating the impacts of potential wildfire impacts.

The 11 western states included in the project comprise the area of the contiguous US that will likely see the highest increase of wildfire occurrence due to climate change (Figure 2).
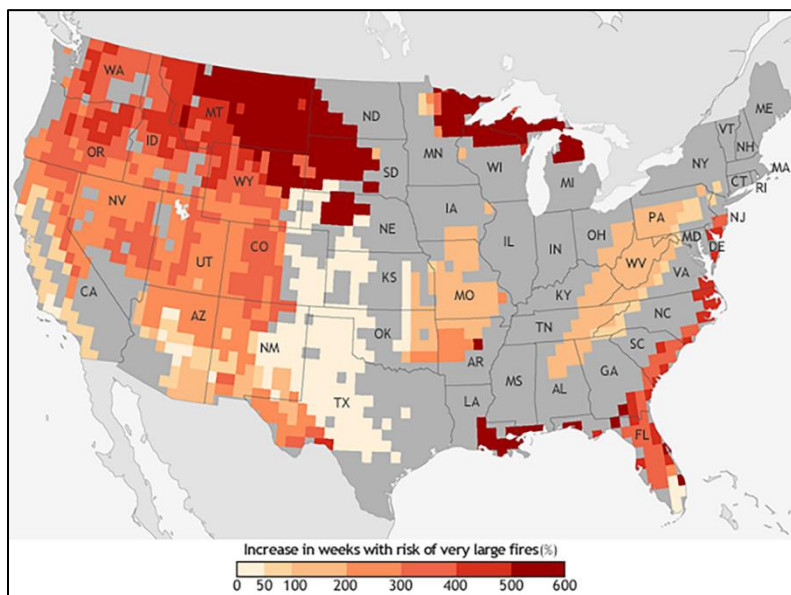
Figure 2. Projected increase in very large fire weeks by 2021-2070 compared to 1971-2000 (CISA 2023)

1.2.2 Scientific Benefit

Scientific knowledge benefits from this effort through the expansion of knowledge of CWPPs, their benefits, history, and intricacies of their planning. The development of a database of CWPPs can help to inform future scientists and natural resource managers of the history of wildfire mitigation measures for any given area, which improves the overall understanding of the effectiveness of wildfire mitigation measures and planning efforts. It should also benefit the

greater body of knowledge with a novel approach to web GIS application development, and

implementation, including an automated process for updates over a long period of time. Open-

source JavaScript libraries are a common backbone of web GIS applications, and one of the most

influential and flexible is Leaflet (Farkas 2017). This research works to leverage Leaflet in the

most efficient way possible to expand the documentation of CWPPs.

This is particularly important in the field of disaster management, where the utilization of

technology, and in particular GIS, can help us prepare for and manage in real-time the impacts of

disasters such as wildfires (Yang et al. 2023). Oftentimes these events are catastrophic for

communities, and the expansion of scientific knowledge when it comes to the utilization of web

GIS technology for disaster planning is a great benefit to communities and scientific

advancement alike.

1.3 Automating CWPP Database Updates

Manually recording data is a time-consuming process, and automation, where possible, is

almost always superior. With funding secured for the continual updating of this database, it is

important to make that process as efficient as possible, so only the minimal amount of human

intervention is needed. The user interface (UI) of the web application allows users to submit new

plans, and a robust data pipeline architecture allows for the review and confirmation or denial of

all plans into the authoritative database.

1.3.1 Automation Process

The initial compilation of CWPPs for the PSU project was done manually, all by hand.

This thesis project improves upon the prior workflow with automation. Recording plans

manually was necessary for the PSU project, because a large volume of detailed information

about each plan was collected. The CWPP Repository, on the other hand, only collects the basic

details for each plan. These include the boundary, name, state, and creation process (private or public), as well as a portable document format (PDF) file of the CWPP itself. Automation of the collection of this data using JavaScript and a PostgreSQL relational database (hosted in AWS RDS) is therefore possible. The only exception is those instances of CWPPs with spatial extents that differ from existing jurisdictional boundaries.

PostgreSQL is one of the most widely used and efficient relational databases, with the great benefit of being open-source (Obe and Hsu 2017). This allows for the storage of a large amount of data, which can be updated almost 100% automatically using JavaScript hosted as an AWS Lambda function, Express as a backend web application framework, and PostgreSQL as the backend database (Huy 2021). This research expands on prior work done on the publishing of data through JavaScript web applications, into PostgreSQL, enabling dynamic datasets constantly evolving based on community input (Brown 2019).

1.3.2 Human Verification

Another key aspect of this automated data pipeline is a human verification step. This safeguard addresses the concern that VGI is error prone, due to a lack of oversight. For this project, an automated emails to the website administrator ensures that all submitted data is reviewed and that all spatial data is accurate before it is published. It provides an opportunity for the website administrator to verify new data as well as contact a user who inputs new VGI data with any questions before sending the information through the remainder of the data pipeline.

1.4 Value of CWPPs

Prior research has concluded that planning for wildfires using CWPPs can have a significant beneficial impact to wildfire preparedness and overall mitigation (Jakes and Sturtevant 2013). While each plan varies greatly – some are a single page that allocate funding,

and some are 50 pages of detailed planning steps – they all have some tangible benefit in preparing communities for wildfire occurrence. For that reason, having clear documentation of these plans, both historical and current, is invaluable to disaster planners and everyone else that has a role in their creation, as well as citizens that wish to become better informed.

1.4.1 CWPP Analysis Goals

Previous research has found that CWPPs improve relationships between bodies responsible for wildfire mitigation, planning, and fighting, and that they improve social learning as the result of documentation of wildfire experience (Jakes and Sturtevant 2013). For that reason, this seems like a worthwhile addition to the academic body of research which includes prior documentation and analysis of these plans (Palsa et al. 2022).

In emergency planning efforts, communication between stakeholders is often lacking, which can lead to differing goals and expectations (see Thompson et al. 2023). Wildfire planning suffers from this problem largely because of how management responsibilities of landscapes is broken up in the US. For example, the United States Department of Agriculture (USDA) is responsible for managing National Forests (NF), National Parks (NP), and other areas, state-level organizations are typically responsible for managing state forests, and counties and fire protection districts generally are responsible for managing more urban or developed areas. This delineation of responsibility leads to a lack of communication between stakeholders and land managers, which can cause inconsistent planning procedures. Having a comprehensive database of plans and planning documentation should help to bring these stakeholders together to address their goals.

1.4.2 Use Cases

The CWPP Repository is intended for several main types of users. The first, and likely most important, is natural resource managers. These are the individuals with the responsibility of mitigating wildfire risk for a specific jurisdiction (sometimes multiple). They are typically tasked with planning and executing new CWPPs, most often for federal, state, local, or private entities. Given the heretofore lack of a comprehensive database of existing CWPPs, there is not a great way for them to locate either prior work in or near their jurisdiction or resources to assist them in plan development. The CWPP Repository provides an easily accessible plan history so they can review past plans and wildfires that have occurred in their jurisdiction since the creation of those plans, allowing resource managers to understand the complex history and dynamics of wildfire planning and wildfire events in their area.

The CWPP Repository also provides nationally applicable CWPP creation resources (templates, guiding documents, etc.), enabling them to quickly identify the relevant literature to assist them in their efforts. This should streamline and improve the CWPP creation process and bring stakeholders from all organizations and entities into a common platform with a clear set of plan creation standards.

Another common type of user will be more specialized natural resource professionals/SMEs, such as silviculturists, hydrologists, botanists, and biologists. These individuals work for a variety of organizations and are commonly tasked with managing natural resources in their jurisdiction. While they are not necessarily in charge of creating CWPPs, they at times assist with their creation and more generally need to have a comprehensive understanding of wildfire occurrence in their area and current and historical mitigation efforts. The CWPP Repository is an educational resources which provides CWPPs, the resources used to create them, and a comprehensive wildfire history dataset. This will make them better equipped

to take management actions with the clear purpose of mitigating wildfires as climate change continues to increase their occurrence over time (Piñol, Terradas, and Lloret 1998).

The last intended type of user is civilians that live in areas that are at risk from wildfire occurrence such as the wildland urban interface (WUI). The CWPP Repository will give them information on what their jurisdiction and community leaders are doing to mitigate the wildfire risk. This will give them a better understanding of wildfire risks and mitigation measures in their area. Community input is a required step in CWPP creation, so engaged citizens is a crucial aspect in the plan creation process.

Chapter 2 Related Work

This chapter provides of overview of related work that was important for the work conducted for this thesis. First, it covers some of the literature on CWPPs. The HFRA laid the groundwork for CWPPs and their use as a community disaster planning tool by providing wildfire mitigation funding to communities that adopted these plans (Radmall 2004). This motivated communities to build out wildfire planning documentation, which has now continued for two decades. Next, this chapter discusses aspects of GIS web application development. GIS web applications can visually communicate a large amount of information, like CWPPs and their spatial aspects, helping users to become more informed. Finally, this chapter discusses literature on open-source coding techniques for database creation and automation. Open-source coding and database utilization is critical for low cost development efforts and a practical solution to building out the front and back end of a web application. Open source databases and code allow you to build a GIS web application from scratch, for very little cost, only having to pay to cloud services.

2.1 Community Wildfire Protection Plans

The most important legislation in the history of CWPPs was the HFRA (Radmall 2004). This was put forward in 2003 and was intended to increase the health and resiliency of US forestland (Radmall 2004). It allowed communities to spend more time on mitigating wildfire impacts by allotting federal funds to communities for mitigation efforts when they worked together to conduct community wildfire planning, by creating CWPPs through collaborative processes (Figure 3) (ODF 2023).

Figure 3. Invitation for a planning workshop (Permann 2022)

In the immediate aftermath of the passing of the HFRA, a number of articles were written that documented it, highlighted its importance, and described how it provided funding for communities that conducted wildfire planning (Radmall 2004). Other work has documented the effectiveness of CWPPs, finding that they are an effective tool for wildfire planning (Jakes and Sturtevant 2013). This research highlighted the fact that these plans help to improve cohesion between a variety of entities responsible for wildfire mitigation (Jakes and Sturtevant 2013). This research makes it clear that CWPPS are an effective tool for getting wildfire mitigation accomplished and provides rational behind this project. With wildfire occurrence on the rise largely due to climate change it is important to use all effective tools to mitigate their occurrence to save lives and land, especially in the advent of megafires that lack the benefits of small-scale lower intensity fires that provide natural systems with key nutrients (Jones et al. 2022).

Further literature has shown that organizational diversity is crucial in CWPP creation, which is an important thing to understand because the HFRA dictated that a variety of

organizations and individuals were involved in the creation of each CWPP (Palsa et al. 2022). Community input is one of the most important aspects of CWPP creation, as communities are home to a variety of people and organizations impacted by and responsible for wildfires (Palsa et al. 2022). Many times wildfires are started by humans and, even when they are not, often continue due to poor brush management and increased effects from climate change, all which can be impacted, even in a small way, by local citizens (Balch et al. 2017). Meeting to discuss the creation of CWPPs spreads awareness and helps to highlight the things that are most important to address in CWPPs. The literature that focused on this organizational diversity factor also built the dataset being used for the database and web application that this thesis is building, which is important to understand (Palsa et al. 2022).

2.2 GIS Web Application Development

GIS web applications are used by a variety of industries to easily spread spatial information in a visual way. There has been a wealth of research into GIS web application development, and this thesis draws from that research for web application development work that is required.

The Leaflet JavaScript library is an open-source web GIS JavaScript library that provides functionality for creating web applications with geospatial elements (Farkas 2017). Leaflet is a lightweight solution that can be expanded upon greatly with custom JavaScript and can utilize ArcGIS Online data through the integration of the ArcGIS API for JavaScript (Farkas 2017). While this approach involves more custom code than extensive web GIS JavaScript libraries like Mapbox GL (built on Leaflet) and Cesium, it provides some basic functionality, things like popups, legends, and basemaps, that can be combined with custom code to build out a dynamic web application (e.g. Figure 4).
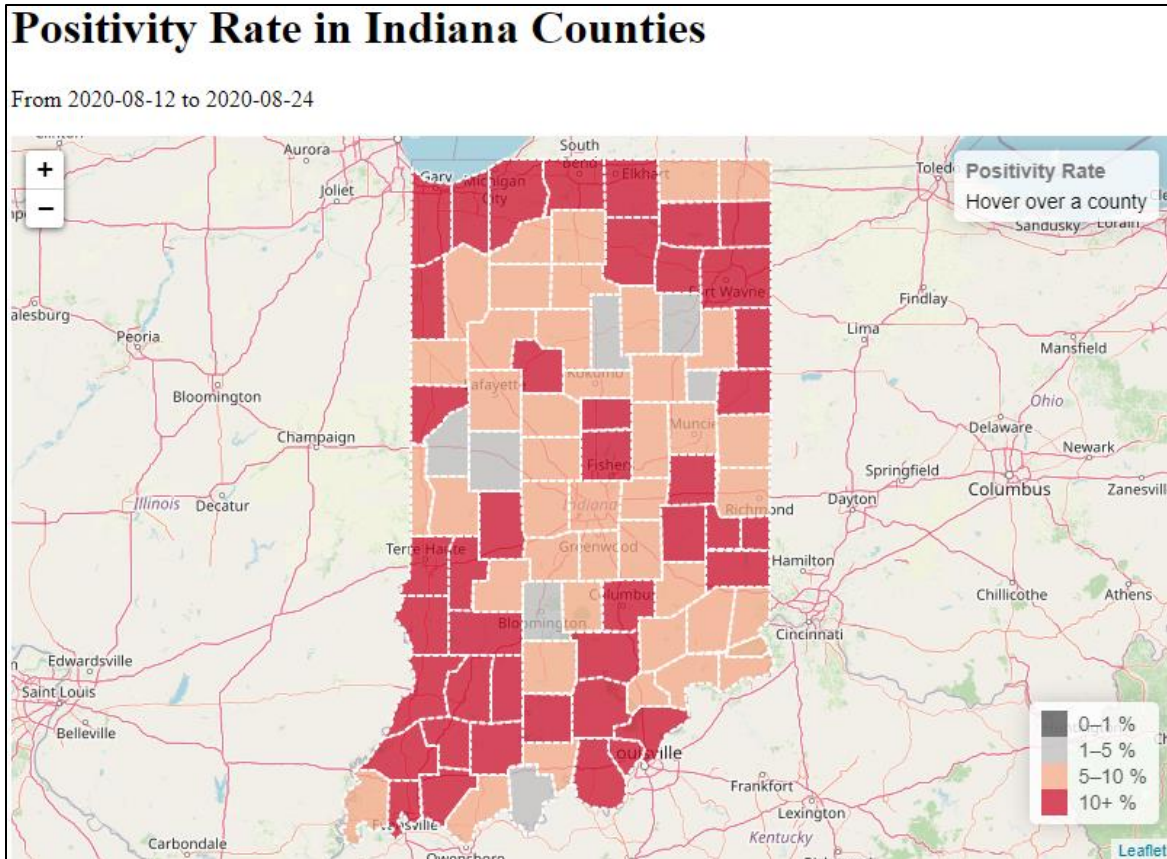
Figure 4. Example of a Leaflet GIS web application (Christie 2020)

One other important aspect of GIS web applications is the incorporation of VGI. Incorporating the input of VGI into a GIS web application allows the dataset of interest to be built out in a much more collaborative and efficient way by crowd sourcing data (Basiouka, Potsiou, and Bakogiannis 2015). However, there is the concern of accuracy with VGI because without vetting users, anyone can enter whatever data they want to, making it possible for inaccurate data to be entered (Basiouka, Potsiou, and Bakogiannis 2015). Human verification is a way to ensure that real-time data is provided quickly, by directing the website administrator to review it as soon as it is inputted by users, and then it can assist the website administrator by communicating through automated emails to users that choose to input data. This is supported by literature that highlights how you can use PostgreSQL databases hosted in cloud environments

like AWS to incorporate PostgreSQL data with a Leaflet web application (Lata et al. 2022) and literature that outlines how to create automated email notifications with JavaScript/Node.js and the Nodemailer module (Satyal 2020).

Existing literature highlights the usefulness of GIS web applications in providing up-to-date data for disaster management (Yang et al. 2023). Real-time data can be provided to users concerning disasters like wildfires, which allows the dissemination of information to happen much more efficiently (Yang et al. 2023). With that information decision-makers (SMEs) and citizens can be more informed during future wildfire seasons to analyze the existing mitigation efforts as emergencies progress and conduct after action reports to see what went wrong and what went right.

2.3 Open-Source Coding Techniques for Database Creation and Automation

Open-source coding is a highly effective way to leverage free software to make a variety of manual tasks automated. Using a combination of JavaScript as the coding language and PostgreSQL as the relational database, you have a low cost and effective way to reduce the workload of manually collecting and entering data. There is a wealth of research on using open-source coding software and database software, which provided a background for this thesis.

PostgreSQL is a widely used relational database management system (RDBMS) that is supported by a wealth of research showing how it can be utilized for efficient and user-friendly database management (Obe and Hsu 2017). This existing research and documentation describe processes for initial database creation, Excel Workbook to PostgreSQL conversion, and the functionality needed for a data pipeline written in JavaScript (Brown 2019).

JavaScript integration with PostgreSQL is supported by a number of open-source modules, which have been thoroughly researched by academics, who have shown how

JavaScript can be integrated with PostgreSQL to create data pipelines that allow JavaScript to

publish data directly to the RDBMS (Brown 2019). This is a computationally efficient process

due to both PostgreSQL and JavaScript architecture, and it allows users to input data, for it to

then be verified, all parties to be notified, and then finally for it to be published or denied all

within a relatively small amount of code.

Chapter 3 Methods

This chapter documents all steps that were followed to complete this project. It starts with the foundational step, the development of use cases and choice of key functions. It was critical to analyze all potential use cases of the CWPP Repository prior to development, to ensure that all required aspects were considered and that all key functionality is included. Without a good base of planning likely the user experience (UX) and UI would be lacking some functionality that would be needed to properly address each use case. Next, this chapter reviews all data needed and its purpose in the CWPP Repository. Then the chapter provides an overview of the development process. This includes everything from gathering the initial data, to coding the web application, to deploying it with AWS. The following two sections break down this overview into detail by delving into the development of the PostgreSQL database and then the web application development process.

3.1 Use Cases, UX/UI, and Key Functionality

One crucial piece of application development is use case analysis and UX/UI assessment. To design an application with a clear purpose you first need to understand who it is intended for, the experience it will provide them, and the motivation for them to visit your web application.

There are two main use cases for the CWPP Repository. The first is SMEs, those that work in natural resource and emergency management. These people are currently lacking an authoritative database to review what plans were previously created and any way to add in their newly created plans to an authoritative database. They also lack a central location to view resources for CWPP creation. This development process imagines that these SMEs will use the CWPP Repository to access resources relevant to their work, such as a historical collection of

CWPPs for their jurisdiction, guidelines, templates. SMEs also need a way to add their plans to the authoritative CWPP database. The UI of the CWPP Repository should allow them to easily find and view plans that cover their jurisdiction (or any other jurisdiction they may be studying). Because the data is organized by jurisdiction – a spatial entity – the heart of the UX/UI must be the map itself.

The next main use case is interested civilians. The development process imagines that concerned citizens will use the CWPP Repository to access resources that clearly show how their community is preparing for wildfires and working to mitigate the threats from them. As with the SMEs, the UI of the CWPP Repository should focus on the map and allow laypeople to easily find and view plans that cover their residence, and the UX should provide them with a variety of data related to CWPP development in their area that is easily accessible and understandable. This is also very important because community feedback is a required part of CWPP creation. The CWPP Repository should provide community members with background information that they can research and then go to community feedback meetings prepared with useful questions about their communities CWPP development, to ensure that shortcuts are not taken when it comes to wildfire mitigation planning efforts.

With the UI and UX of the CWPP Repository set, specific functionality could be chosen. The CWPP Repository should assist SMEs and concerned citizens, providing a central location for a historical record (database) of CWPPs, a variety of data related to each record, some visuals showing dynamics of plan creation, and resources for those that need to work on developing CWPPs for their own community. Table 1 itemizes the specific technological elements and functions required to be developed to fulfill these use cases and the following paragraphs describes them.

Table 1. Required functions

	Required Functions
1.	Enter CWPP data
2.	Review CWPP entries (by administrator)
3.	Read a tutorial
4.	Submit a contact form
5.	View and interact with a list of map layers
6.	View legend
7.	Filter for data by state
8.	Choose from multiple basemap options
9.	View map pop ups
10.	View and download resources for CWPP creation
11.	See tips by hovering over various website elements

Function 1, the ability to add new CWPPs to the CWPP Repository, is the heart of this VGI Project. Function 2, the ability of administrators to review VGI entries prior to their addition to the database and appearance of the website, is key to maintain the accuracy of the information in the CWPP Repository. Working through a website tutorial, function 3, is necessary to ensure that users are able to interact with the full functionality of the website. Function 4, the submission of a contact form, allows users to contact the website administrator directly, allowing for bug identification and assistance where needed. Function 5, the ability to view and interact with a list of layer, is key for effective engagement with the map. In addition to the key data layer of CWPPs, the CWPP Repository must provide additional wildfire data so users can explore real-time fire impacts. The user must be able to turn data layers on and off, otherwise the map would be overloaded and difficult to understand. Function 6, the ability to

view a legend, provides symbology so the user understands what the data layers on the map represent. The ability to filter the data shown on the map by state, function 7, lets users focus on their state of interest, which is critical as many natural resource managers work at the state level. Function 8, the ability to change basemaps, allows the user to view satellite imagery, topo maps, and others, adding context to all data layers. Function 9, pop ups, provide for the identification of any specific feature on the map. The ability to view and download resources for CWPP creation, function 10, is key for SMEs to assist them with developing new CWPPs and laypeople to learn about mitigation efforts. Function 11, hover tips, allow the user to gain insight to specific elements without having to delve into the full tutorial.

3.2 Data Requirements/Sources

The web application and database needed to contain a variety of data, including the CWPP PDFs, tabular and spatial CWPP data, jurisdictional boundaries, and wildfire boundaries and points (Table 2).

Table 2. Data required for web application

Title	Data Location/Format	Source
CWPP PDFs	Google Drive/PDF	Previous PSU research
CWPP Data	PostgreSQL/RDBMS Table	Previous PSU research
Boundary Datasets (County and Community)	AGOL (ArcGIS Online)/Hosted Feature Layer	Census data published by Esri Data & Maps hosted on AGOL
Wildfire Data	AGOL/Hosted Feature Layer	USDA data published by Esri Live Feeds hosted on AGOL

All data layers in this project have a high level of precision and accuracy, as all data is from reliable sources, including the USDA AGOL portal, the Census Bureau, and previous work conducted by the author on behalf of PSU and Ohio State University (OSU), which was meticulously conducted to ensure quality data was maintained.

The previous research conducted that built some of the base data set out to research whether one of the requirements in the HFRA, that CWPPs need to be created through a collaborative process with a diversity of stakeholders and organizations involved, was being met. (Palsa et al. 2022). To analyze that the team of researchers compiled all CWPPs they could find for the 11 states focused on in this thesis, to document every individual that contributed to every plan, and their role in the plan development as well as who they work for or represent (Palsa et al. 2022). Then they conducted a linear regression to see what factors correlated to high diversity of organizations were involved in CWPP creation for each plan (Palsa et al. 2022). That is where the plan PDF collection used in this thesis came from, and where the tabular and spatial data for every CWPP in the CWPP Repository came from. The original CWPP data contained a variety of details about every individual and every plan, but only the most basic, important fields were used for the CWPP Repository, as the rest wasn't needed and would overcomplicate future CWPP collection.

The fire data from AGOL includes a layer of every active wildfire boundary in the US, and a layer of every active wildfire point in the US, although there isn't a perimeter provided for every point. Each of these layers contains information like the incident name, mapping method used to create the data, comments, calculated area, creation date, and current date.

3.3 Development Process Overview

To build the web application and database there were a number of steps involved (Figure 5). The first step was to create a GitHub repository to house the scripts for the web application. This ensures that there is a comprehensive version history of this code repository maintained and also allows it all to be open-source and available to anyone, so that the techniques within it can be utilized in other individual's coding efforts. The best storage solution identified for this project was AWS RDS, for hosting an instance of PostgreSQL to store the needed data in, as a scalable and inexpensive solution.
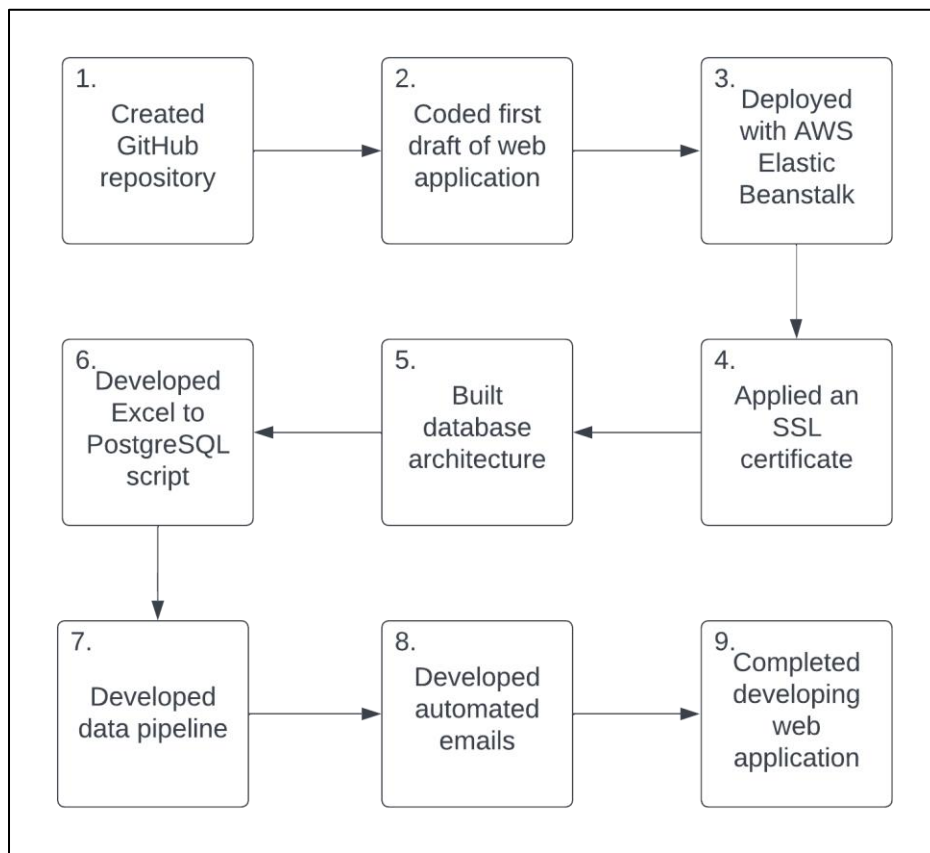

Figure 5. Development process

The next step was to code the first draft of the basic web application using the Leaflet JavaScript library, HTML, CSS, and JavaScript, and then to deploy it and apply a free secure

sockets layer (SSL) certificate with Elastic Beanstalk/AWS. AWS was chosen for cloud services because of compatibility between its variety of services and the scalability its provides.

With that accomplished a PostgreSQL database and required tables were created for data storage. Then the Python script to convert Excel Workbooks containing CWPP data to tables in the PostgreSQL database was written and pushed to the same GitHub repository. JavaScript was written to setup the data pipeline, which uses JavaScript and the Express backend web framework to import user inputted data into the PostgreSQL database, and then uses the Node.js Nodemailer module to alert the website administrator that there is new data which needs to be verified, which is run based on triggers in the server code. Then the data is verified, and an application programming interface (API) endpoint setup in the server code handles what happens to it when it is accepted or denied and sends confirmation emails to the user and website administrator confirming actions taken. This automated process ensures that the user and administrator are kept informed throughout the process. With all of that complete the data pipeline and automated email process was incorporated into the CWPP Repository and it was developed completely.

3.4 Database Development

PostgreSQL was chosen as the RDBMS to house CWPP PDFs and associated data. It is a widely used RDBMS, and it provides a robust and open-source environment to build out a database to store CWPP data in (Obe and Hsu 2017). The spatial dimension of these plans required a geospatial friendly RDBMS, which is why PostGIS/PostgreSQL was chosen. PostGIS is an extension to PostgreSQL that allows the user to store geospatial data in the RDBMS as a new data type it defines called geometry (Obe and Hsu 2021). The geometry data type allows for the storage and manipulation of geospatial data (Obe and Hsu 2021). This capability ensures that

each CWPP is not just a text document but a spatial entity that is able to be analyzed, visualized, and utilized in GIS software.

Hosting this database in AWS RDS was another important step, aimed at ensuring easy access to data in the frontend of the application (Figure 6). As users ranging from professionals to community members use the web application to submit new plans or access existing ones, the underlying database infrastructure needs to be resilient and scalable. AWS RDS ensures this, allowing the application to remain a reliable resource that provides real-time access to CWPPs and enables collaborative efforts for wildfire management and planning.
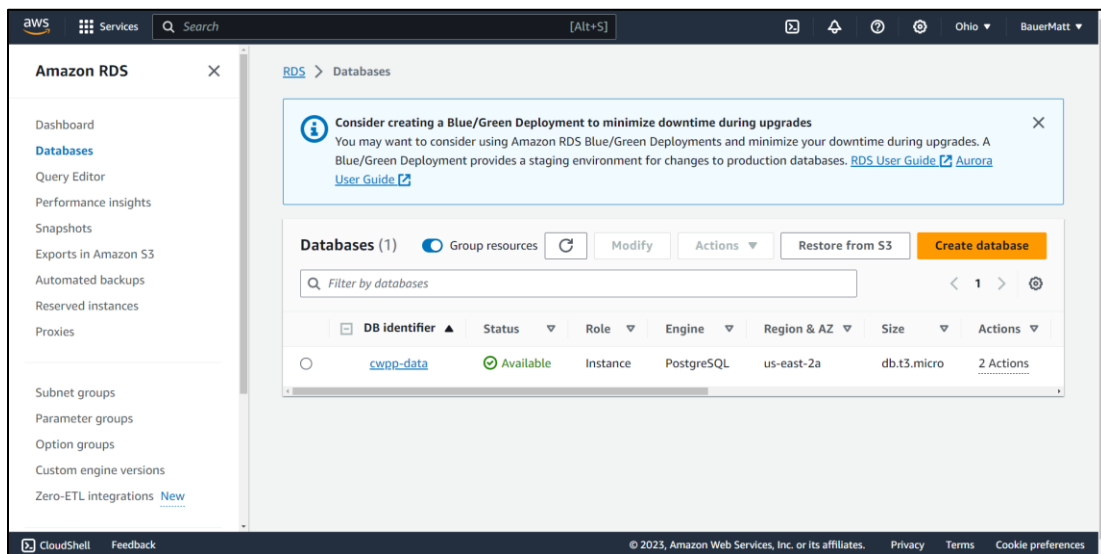

Figure 6. AWS RDS setup

3.4.1 Database Structure

The backbone of the CWPP Repository is its database, using the default title postgres, housed within a server called cwpp-data (Figure 7).
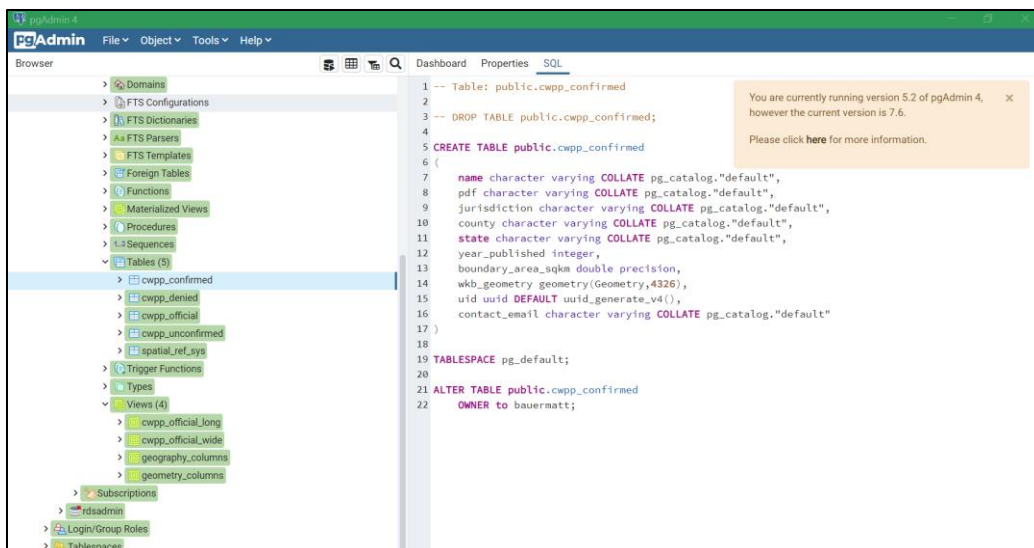
Figure 7. PostgreSQL database tables and views used for backend

This database has the PostGIS extension installed, allowing for greater geospatial capabilities than the base PostgreSQL geospatial data types. The combination of PostgreSQL and PostGIS ensures that the data, including text, numeric, and spatial attributes of the CWPPs is efficiently stored and queried for use in this application.

Within this database, several tables are used to manage the data pipeline from the website to the database. The cwpp_confirmed table contains entries that have been reviewed and approved by the website administrator. Each record captures important aspects of the plans, from basic attributes like name and jurisdiction to more intricate details like the associated PDF link and the boundary area. Potentially most important is the wkb_geometry column which contains the spatial data for each plan. By capturing the spatial dimension of each plan it allows the user to clearly identify the plan boundary that each individual CWPP covers.

The cwpp_denied table is an archive for plans that were submitted, but denied by the website administrator because they were either duplicate plans to those already in the database, or not CWPPs. While they do not contribute directly to the application's official dataset, their retention is important just to track who has been submitting plans and maintain a record of those

24

transactions. The cwpp_official table on the other hand serves as the official CWPP database, containing all validated plans, both historical and newly approved.

Another important table in the postgres database is cwpp_unconfirmed. This acts as a staging area for new CWPP submissions, ensuring that every user's contribution is stored for later review, which is planned to happen within 48 hours from the plan's submission. Over time these entries all end up in the confirmed dataset or the denied one, depending on their review.

To have clean, processed data for popups and general use and viewing two views were created. The cwpp_official_long view, designed for the looking at each plan individually (as an administrator), presents data in a more detailed format, with each CWPP having its own row. The cwpp_official_wide view is needed for popups where each boundary on the map represents every plan ever made for that jurisdiction, in order to prevent many stacked matching boundaries from appearing on the map and confusing the user.

In sum, the structure of the postgres database is a blend of uniformity, functionality, and foresight. It ensures that every CWPP, irrespective of its source, characteristics, or spatial footprint, is accurately captured, readily accessible, and useful for both present and future analyses, as well as processed according to decisions taken, like denial or confirmation of a plan (Figure 8).
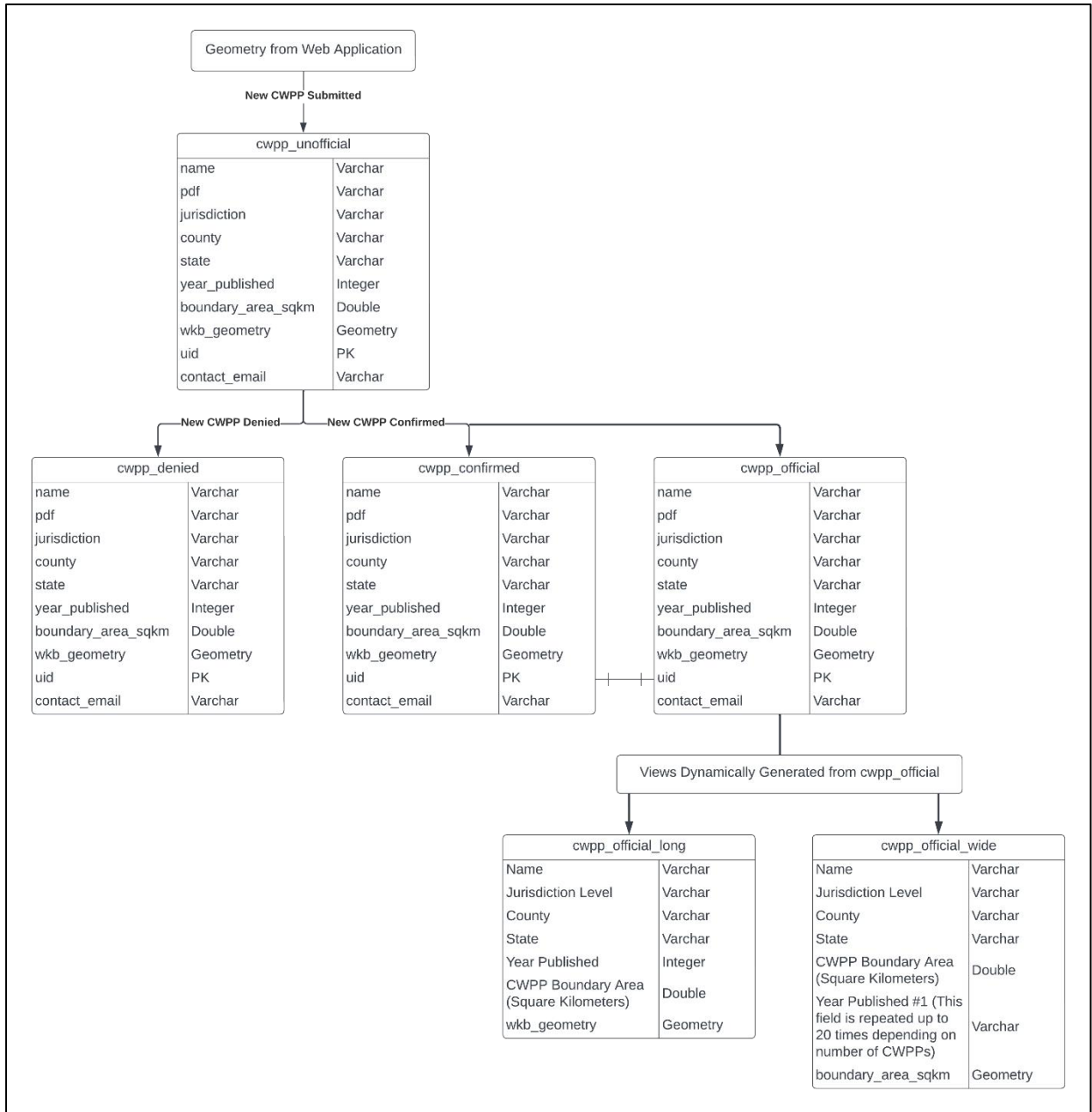
Figure 8. Diagram of database configuration and sequence of events

3.4.2 Connection and Queries

One of the most important aspects of developing this website was building out the

interaction between the database and frontend of the web application, as well as API endpoints

used for CWPP confirmation to the official dataset. This serves as the bridge between raw data

and user interactions, and it sets up the application's responsiveness, reliability, and security.

The web application, built on top of the Express.js framework, uses a Node.js

environment to manage server-side operations. Express.js is a minimalist and modular approach,

giving access to a method for routing HTTP requests, handling responses, and integrating

middleware (Krause 2017). It is within this framework that the connection to the PostgreSQL

database is established, and where queries, crucial for data retrieval and manipulation, are

executed (Figure 9).


```
app.get('/api/data', async (req, res) => {
    try {
        const { rows } = await pool.query('SELECT *, ST_AsGeoJSON(wkb_geometry) as geom FROM cwpp_official');
        res.send(rows.map(row => {
            let properties = { ...row };
            delete properties.wkb_geometry;
            delete properties.geom;
            return {
                type: 'Feature',
                geometry: JSON.parse(row.geom),
                properties: properties
            };
        }));
    } catch (err) {
        console.log(err.stack);
        res.status(500).send('Error fetching data');
    }
});
```

Figure 9. API setup in the server code to pull data from the AWS RDS instance of the
PostgreSQL database

The pg package, a non-blocking PostgreSQL client for Node.js, plays a crucial role in

this piece of the project. It provides the tools and methods for connecting to the database,

querying it, and fetching results of those queries. One of its most useful features is connection

pooling (Carlson 2023). Given the potentially high number of concurrent users accessing the web

application, connection pooling ensures that multiple connections are maintained and reused,

which helps to improve the application's efficiency and reduces the load on the database.

When a user interacts with the web application, whether they are submitting a new

CWPP or accessing an existing plan's data, their request is routed by Express.js to the

appropriate handler. This handler, utilizing the capabilities of the pg package, connects to the database, constructs the necessary structured query language (SQL) query based on the request, and retrieves the desired data. When a user accesses the web application a query leveraging PostGIS functionalities is executed, extracting the relevant geospatial data from the wkb_geometry column for display on the map view.

Security and data integrity are another crucial part of this process. By utilizing Express.js middleware the application validates and sanitizes user inputs, guarding against potential threats. This protective layer is further reinforced using prepared statements and parameterized queries within the pg package, minimizing risks like SQL injection.

Once the query is executed and the data is retrieved, it is processed and sent back to the client side of the application, ensuring the user receives real-time, accurate, and relevant information. This frontend to backend communication, facilitated by Express.js and the pg package, ensures that the application remains responsive and always synchronized with the underlying database.

The process of establishing connections and executing queries requires the use of several technologies, strategies, and best practices. It ensures that the wealth of information housed within the postgres database is not just stored but is also readily accessible for every user of the web application.

3.5 Web Application Development

The CWPP Repository aims to provide a VGI driven repository of CWPP data where users can access existing plans, visualize their boundaries, and explore their data. It should provide CWPP data for users just interested in looking into local CWPPs and seeing what their community is doing to mitigate the risks of wildfire, and it should serve as a resource for SMEs

to generate their own CWPPs and conduct research on those done in the past. Most importantly it is meant to provide a place for people to collaborate when it comes to wildfire planning. Currently there is not a way for wildfire and emergency management professionals to submit CWPPs for storage in a centralized location, and the CWPP Repository seeks to remedy that.

While the underlying database provides the foundational data layer, the web application is where users are able to visually assess the data and existing CWPPs. By using simple Leaflet widgets, an interactive map, and tabular data in pop ups, the application allows users to gain insight from spatial and non-spatial data (Figure 10). Another key piece of this website was user accessibility and ease of use. Understanding that the CWPP Repository will be used by people with less experience using geospatial technology, and those with a lot of experience, guided the development, requiring everything to be straightforward and easy to understand.

Figure 10. Web GIS architecture diagram
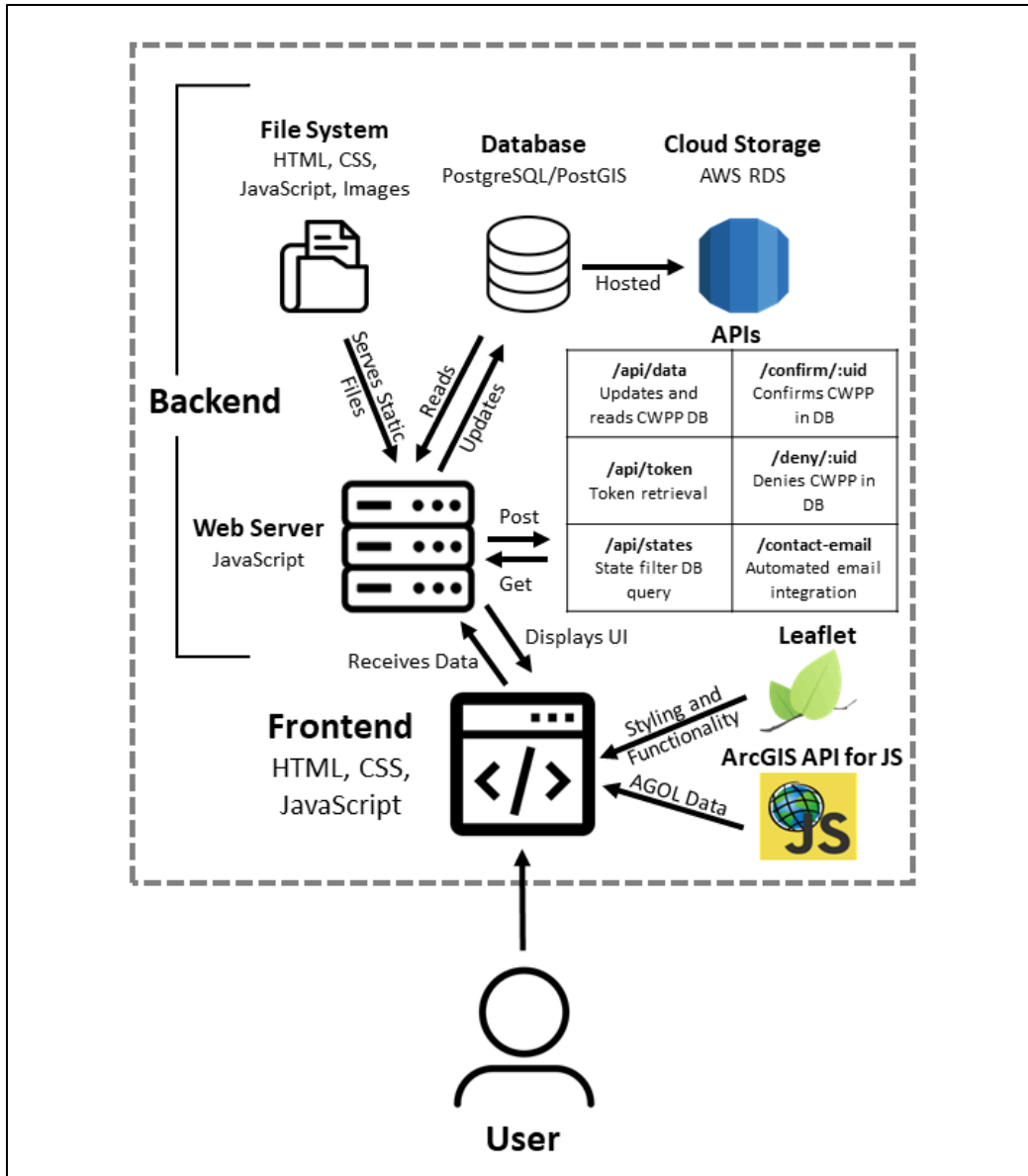
3.5.1 Frontend Development

The frontend of the CWPP Repository is where users can visualize all of the spatial data,

inspect the tabular data, interact with the map, and submit new plans. Creating this interface

required a combination of design principles, technological integrations and considerations

focused on user experience to ensure both functionality and appeal.

The foundation of the frontend relies on three components: HTML, CSS and JavaScript. HTML provides the structure by organizing the web application and making content accessible. Each element, such, as headers and form inputs, is thoughtfully placed to guide users through the application. CSS is responsible for styling this structure. It is what sets up all of the colors and fonts and animations that bring the web application to life and ensure that it remains visually appealing for the user.

JavaScript plays a role, in making the frontend dynamic by enabling interactivity. Every button click, form submission or map interaction is controlled by JavaScript. However, its significance extends beyond interactions. Through the integration of JavaScript real time data retrieval and rendering becomes possible. This means that as users navigate through the application they will experience transitions, live updates, and quick feedback without experiencing disruptive page reloads.

At the core of the CWPP Repository lies visualization. Recognizing the aspects of wildfires and the importance of protection plans, Leaflet.js was chosen as the module to integrate geospatial capabilities. This lightweight mapping library transforms data into interactive maps allowing users to easily pan, zoom click on CWPP boundaries and access detailed plan attributes. By combining data with user interactivity Leaflet.js ensures that users not only view maps but actively engage with them. This interaction enables them to gain insights, identify patterns, and visualize the coverage of CWPPs effectively. In addition to that, a loading screen was developed so users do not see the data loading, they are just temporarily in the loading screen showing a wildland fire with a spinning circle.

The frontend development process for the CWPP Repository was developed with the idea of creating a platform for exploring CWPPs and gaining VGI data for the comprehensive CWPP

database. It evolved into an interface that strikes a balance between design/visual appeal, functionality, and user experience. Each decision made during development, from selecting the color palette to designing map interactions, was carefully considered with the user in mind.

3.5.2 Backend Development

The backend of the CWPP Repository is responsible for loading basemaps and CWPP data from the AWS RDS instance and for setting up APIs that allow for database updates through automated emails. While users do not see this part of the application, it is a critical component for ensuring that data integrity is upheld, user requests are processed and the subsequent data is returned, and interactions between the frontend and database are maintained as needed for the proper functionality of the CWPP Repository.

Node.js plays a role in the backend operations by providing a runtime environment that has transformed server side development. Its event driven architecture and non-blocking input/output (I/O) allow the server to effectively manage requests (Krause 2017). Whether it is a user submitting a CWPP or querying an existing plan, each request is handled quickly to minimize wait times and optimize the user experience, ensuring that the server manages requests without causing lag in the frontend, and maintains the data quality needed.

Express.js, built on top of Node.js, serves as the foundation for server side logic. As a web application framework Express.js simplifies route creation, middleware integration and management of HTTP requests and responses (Krause 2017). When a user submits a new CWPP through the embedded form in the web application Express.js directs this request to the handler. The handler then communicates with the database, stores the data, and sends back a confirmation to the frontend. This entire process, although intricate, becomes efficient. Express.js streamlines it and makes development of frontend to backend interactions much simpler and ensures that

they are executed more efficiently and securely. Middleware functions in Express.js work to

sanitize user inputs, and safeguard against SQL injection threats through queries, providing this

enhanced security, making attacks that could compromise the security of the underlying database

less likely.

The interaction with the PostgreSQL database is crucial in development especially

considering its extensions through PostGIS. To facilitate these interactions, pg, a Node.js

module, was utilized. This module helps to establish connections efficiently by pooling them.

Additionally, it enables the execution of queries so the data in the AWS RDS instance can

quickly be accessed, modified, and analyzed as needed.

A few innovative pieces of code were involved, like server side code that allows the web

application to communicate with the PostgreSQL database hosted in an AWS RDS (Figure 6),

enabling users to make real-time edits to a database that is managed by the website administrator,

and when those edits are confirmed the respective plan is added to the authoritative dataset.

```
20   // Connect to database
21   client.connect()
22       .then(() => console.log('Connected to database'))
23       .catch((err) => console.error('Connection error', err.stack));
24
25   app.use(cors());
26
27   // UPDATE UPDATE LIMIT STATEMENT
28   app.get('/api/data', (req, res) => {
29       client.query('SELECT *, ST_AsGeoJSON(wkb_geometry) as geom FROM cwpp_wide LIMIT 100', (err, dbRes) =>
30           if (err) {
31               console.log(err.stack);
32           } else {
33               res.send(dbRes.rows.map(row => {
34                   // copy the row object
35                   let properties = { ...row };
36
37                   // remove the geometry from the properties
38                   delete properties.wkb_geometry;
39                   delete properties.geom;
40
41                   // return the GeoJSON Feature
42                   return {
43                       type: 'Feature',
44                       geometry: JSON.parse(row.geom),
45                       properties: properties
46                   };
47               }));
48           }
49       });
50   });
51
52   app.post('/api/data', (req, res) => {
53       const uid = req.body.properties.uid;
54       const cwpp_name = req.body.properties.cwpp_name;
55       const jurisdicti = req.body.properties.jurisdicti;
56       const county = req.body.properties.county;
57       const geometry = req.body.geometry;
58
59       const query = `
60           INSERT INTO cwpp_wide (uid, cwpp_name, jurisdicti, county, wkb_geometry)
61           VALUES ($1, $2, $3, $4, ST_GeomFromGeoJSON($5))
62       `;
63
64       client.query(query, [uid, cwpp_name, jurisdicti, county, JSON.stringify(geometry)], (err, dbRes) => {
65           if (err) {
66               console.log(err.stack);
67               res.status(500).json({
68                   status: 'error',
69                   error: {
70                       message: err.message,
71                       code: err.code,
72                       detail: err.detail
```

Figure 11. Custom JavaScript server functionality

3.5.3 Testing and Quality Assurance

Testing and ensuring the quality of the web application went beyond creating features and functionalities. It also involved making sure that every component worked correctly and consistently. By following a testing and quality assurance (QA) process throughout the development phase it was ensured that the CWPP Repository will always work as expected for users, that every single component will function even when situations differ based on all related interactions that a user could make.

The testing process started with unit testing that was performed on every feature in both the front and backend. This type of testing focused on validating components in isolation to ensure that each function, module, and interface behaved as expected under a variety of conditions. Since the CWPP Repository deals with data, user submissions, and multiple APIs for reviewing and confirming or denying data, these unit tests were invaluable in identifying edge cases, errors, and potential inefficiencies.

In addition to unit testing integration testing was conducted. Unlike unit tests that examined components, integration tests evaluated how these components interacted with each other. One piece of this testing was analyzing how the frontend communicated with the backend during a CWPP submission, and how the server interacted with the database during a query. These tests ensured that all individual components seamlessly collaborated to deliver a worthwhile user experience.

Testing and QA of the CWPP Repository was not just done at the end of development, but consistently and frequently throughout development. For each change made to the code behind the CWPP Repository further testing had to be accomplished to ensure that it is not just feature rich but is also reliable and secure. Constantly conducting tests ensured that this application was built over time in a reliable way that ensured users will always have a good experience when they go to it, and that any errors will be minimal and logged for review by the administrator.

3.5.4 Deployment and Scalability

For deployment the AWS cloud service, Elastic Beanstalk, was chosen. Elastic Beanstalk offers a platform as a service (PaaS) solution that simplifies the deployment and management of web applications and services (Vliet 2011). With its automated infrastructure setup, deploying

the application became as simple making some small edits to configuration files, and uploading the code, although some extra care had to be taken to ensure database access in the frontend that allows users to explore and add to the data without compromising its security. Elastic Beanstalk also makes it easy to monitor the health of the website, and review logs related to its utilization by users on the open internet.

Beyond just deploying the website initially, Elastic Beanstalk also makes scaling it as the user base increases much simpler. Through Elastic Beanstalk's inherent auto scaling capabilities the application is able to adjust its server automatically according to the varying user demand, enabling seamless and constant scaling of the website, a feature that few other services offer in such a user-friendly way (Vliet 2011). Whether there is a surge in traffic or increased data processing, Elastic Beanstalk will automatically allocate resources to maintain performance for users. Similarly, during periods of inactivity resources can be reduced to optimize AWS costs.

Additionally, continuous integration/continuous deployment (CI/CD) practices have been implemented with Elastic Beanstalk to ensure that the application is always up to date with the latest features and fixes. Any code modifications that pass testing in AWS are seamlessly deployed to the production environment, which greatly helps to minimize any website downtime, ensuring users are always provided with a great experience even when the website is undergoing an update.

Using Elastic Beanstalk ensures that the CWPP Repository can fulfill the current needs of this website and that it will scale easily in the future as the user base increases, or the web application requirements change. This will allow it to remain agile and adaptable in the future as CWPP documentation needs and demand vary over time, reducing future work required for this effort.

Chapter 4 Results

This chapter walks the reader through the completed CWPP Repository while evaluating how intuitive and user friendly this application is. The chapter assesses how AWS cloud services impacted the CWPP Repository. The chapter then walks through the semi-automated process for review by an site administrator of new VGI entries to the CWPP Repository. The final deployed web application can be found at https://cwpp-repository.com/, and the GitHub repository for it can be found at https://github.com/SpatialSolutions93/CWPP-Database-Website/tree/main/WebApp.

4.1 Web Application Performance

The success and usefulness of a web application heavily rely on its performance. This is especially true for the CWPP Repository, which serves as a bridge between the CWPP database and individuals interesting in inspecting and analyzing its data. The performance metrics of this application go beyond measurements; they reflect how efficient, reliable, and effective the platform is in achieving its goal.

Developing such a platform poses a challenge of finding the balance between functionality and simplicity. While it provides insights into wildfire management and community planning through its data environment it also prioritizes accessibility for users with varying technical skills. The CWPP Repository was designed not as a data storage center but as a platform that fosters communication between users and the data and furthers collaborative efforts in the realm of wildfire management.

This section aims to explore aspects of the applications performance in detail. It will dig into aspects of the web application such as UI design details and decisions, data retrieval

mechanisms, and data visualization, to understand how each component contributes to enhancing the user experience.

In addition to that, the impact of this application in real-world scenarios will be assessed by considering the interactions that users will have with it. It will aim to understand how its features can be utilized effectively and what value it brings to the potential community of users, including interested citizens and SMEs that work in the field of wildfire and emergency management.

4.1.1 User Experience and Interface

The CWPP Repository is situated at the crossroads of data and a wide range of users. It serves as a platform where professionals in wildfire management, community planners, and curious residents come together to gain insights into CWPPs. Given this user base it was crucial to design an interface that is both user friendly and informative, for the variety of users that may visit it with different goals and needs in mind. It aims to be easy to use and robust in functionality.

The core principle driving the approach in designing this applications interface was recognizing that an effective interface goes beyond appeal; it should also foster user engagement. Every element on the platform including layout and color choices has been carefully selected to ensure clarity and make website navigation intuitive. Interactive components like custom Leaflet widgets, data overlaid on the map view, and all elements within each widget have been seamlessly integrated to enable users to explore data in depth while maintaining an uncluttered view (Figure 12).

Figure 12. Landing page of the web application

A critical aspect that the development of the CWPP Repository focused on was incorporating call to action buttons and straightforward navigation menus. Recognizing that users may have a variety of goals when using the platform; some may want to submit CWPP data, while others may be looking for specific information, the design of the CWPP Repository aimed to help users easily navigate through it to whatever their intended destination and use case may be. To further enhance the UX a handful of feedback mechanisms were implemented, in particular a contact us form and a tutorial were included in the UI to help provide users with some technical assistance in how to navigate and utilize the web application, as well as providing them with a way to contact the website administrator with any issues that they may have whilst utilizing the web application.

The UX and UI of the CWPP Repository demonstrate a design approach that prioritizes meeting user needs. By combining functionality with simplicity, the platform ensures that users can easily access and interact with CWPP data making it an invaluable tool in wildfire management and documentation management and storage.

4.1.2 Data Retrieval and Display

With the CWPP Repository being focused on the querying and updating of data in the AWS RDS database, the speed, accuracy and efficiency of data retrieval are some of the most critical factors in its use and design. The architecture and design choices for this application were mainly driven by ensuring prompt and efficient data retrieval and visualization, ensuring that data is used in a way that maximizes its accessibility to users that may be visiting it for a variety of reasons.

The backbone of this data retrieval system is a PostgreSQL/PostGIS database. Recognized for its ability to seamlessly handle queries and its open-source nature, this RDBMS was a fitting choice for an application like this. Whenever users interact with the platform, whether its navigating maps, submitting a new CWPP, or applying data filters, the application is optimized to access this database and fetch data with as little delay as possible. The state filters are a key component because they allow for the data to be filtered by state, which should be helpful to individuals only interested in CWPPs in a particular state, as it zooms to the chosen state and clears all CWPP data outside of it from the map (Figure 13).
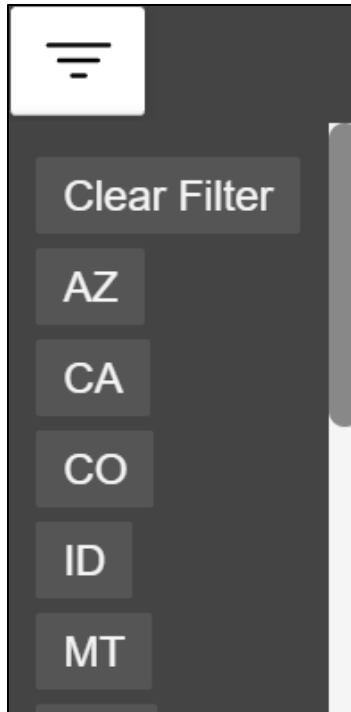
Figure 13. State filters

When a user clicks a CWPP boundary it is highlighted in a semi-transparent red, and a pop-up appears showing its related data, and a hyperlink which routes to a PDF of the CWPP hosted in Google Drive (Figure 14Figure 13). If there are overlapping boundaries there is an option to switch between each CWPP that exists at the point clicked.
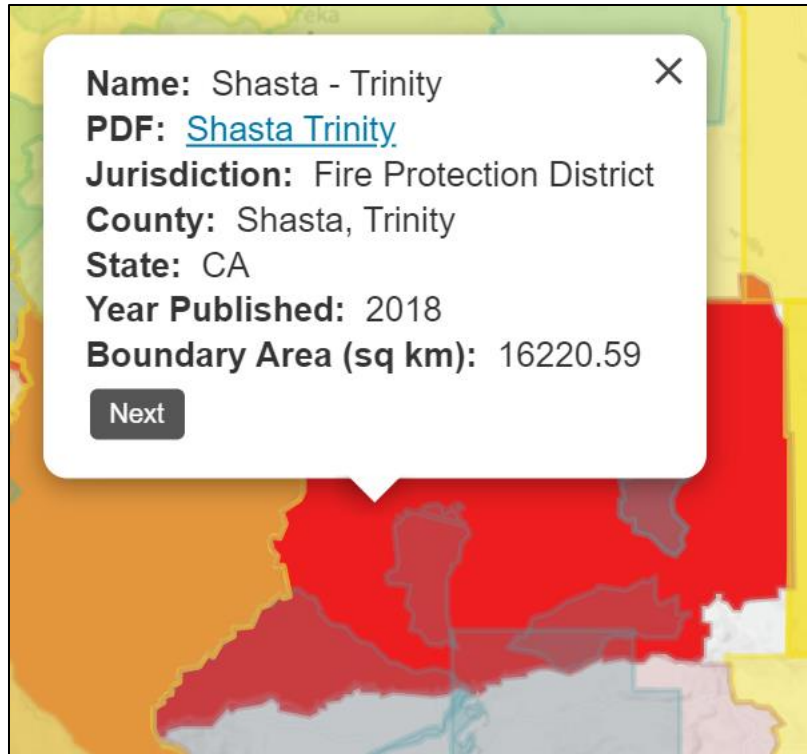
Figure 14. CWPP pop-up

There are also two real-time datasets from AGOL: wildfire points and boundaries. In testing, these both seem to be extremely stable, and they load quickly without noticeable lag. In the future, it will be important to monitor these, to ensure they remain accurate, existent, and stable. They can be accessed and turned on and off in the layer list (Figure 15).
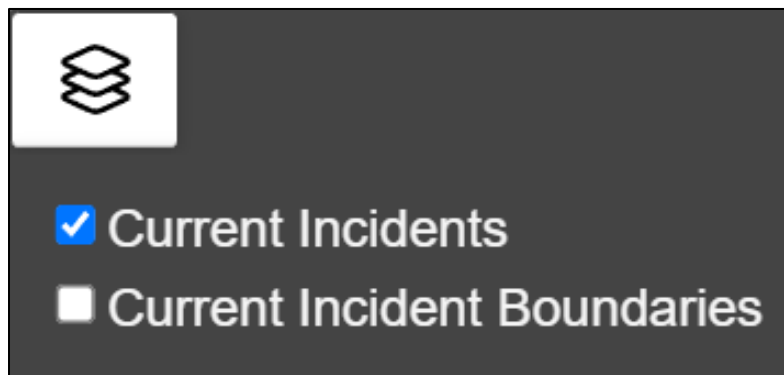

Figure 15. Layer list

When a layer is turned on and visible on the map users can view its symbology in the legend (Figure 16).
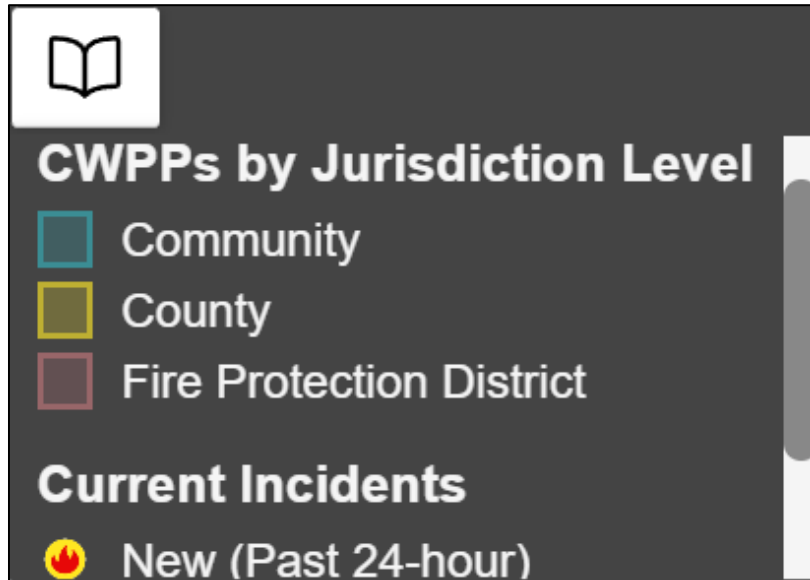

Figure 16. Legend

In addition to the CWPP and wildfire data that is available to the user, they also can change the basemap to one of three choices pulled in from the Mapbox API, including a light gray basemap (the default when the CWPP Repository loads), a streets basemap, and a satellite imagery basemap (Figure 17).
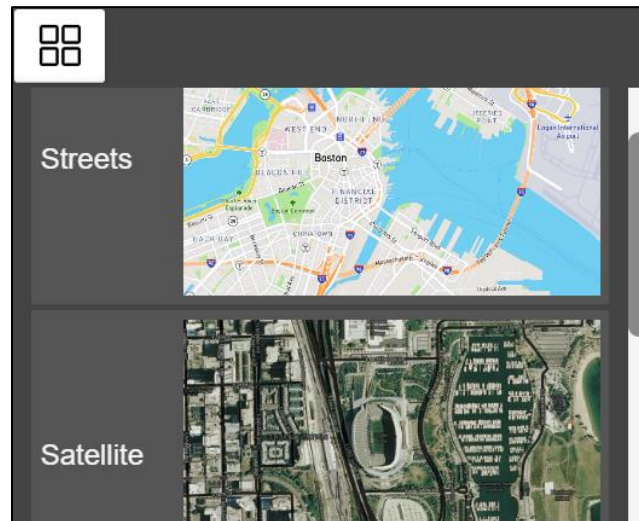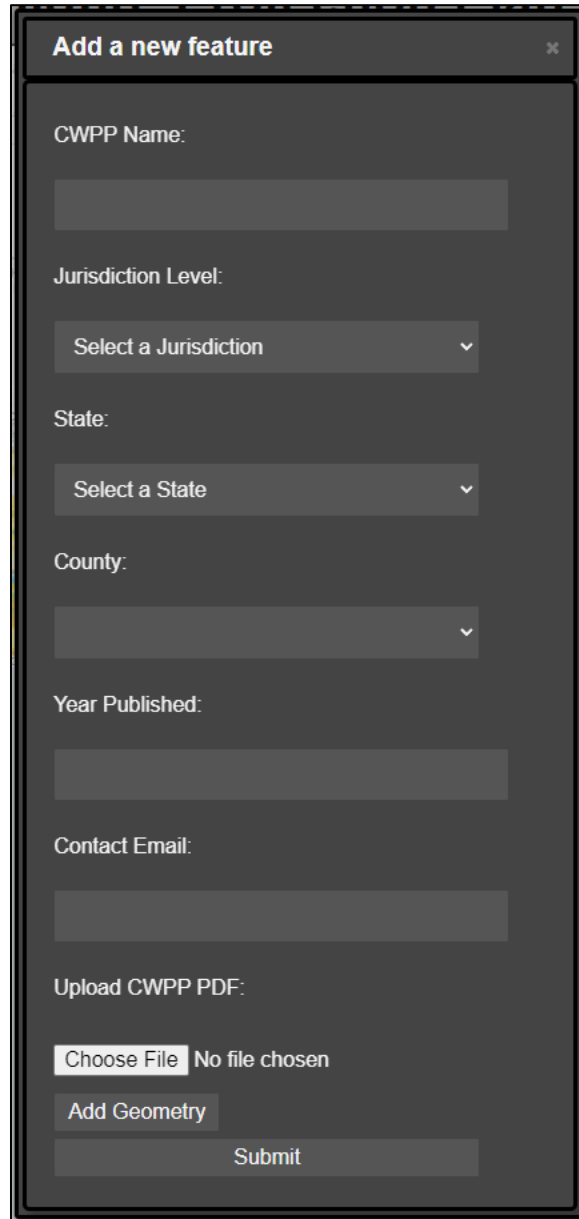

Figure 17. Basemaps

Efficient data retrieval is only part of the challenge. The true test lies in presenting this information to users in a way that is both understandable and actionable. Acknowledging this, the web application employs a rendering mechanism, so that as data is retrieved it is processed and visualized on the go adapting to user inputs and interactions, such as when a state filter is applied to the data. This ensures as little lag time and user distractions as possible, smoothly loading data as query parameters change based on user interaction. Another important aspect of that was simply changing zoom levels based on data filtered, giving quick transition times where the relevant data could be loaded into the map without the user noticing any delay.

4.1.3 User Interactions

User interactions are the most critical aspect of any web application, as their efficiency and intuitiveness are what allows users to easily navigate and utilize it as they need. In the case of the CWPP Repository user interactions are not merely navigating different aspects of the application, but easily adding to the database of CWPPs through the input form used to gain more VGI data.

The intention of the user interactions in the CWPP Repository are to foster a collaborative approach to wildfire planning through a community of users that are actively engaged with the current set of CWPP documentation. It does not just serve as a passive web application, instead it is meant to evolve over time as more and more VGI data is added to it, only increasing its usefulness over time. It intends to drive users to explore the current repository of CWPPs, understanding what the wildfire management landscape looks like in their local jurisdiction, which is critical given the intentionally collaborative nature of CWPPs that require a level of community input.

Potentially the most significant user interaction in the CWPP Repository is the CWPP

submission process (Figure 18).

Figure 18. CWPP submission form

Due to the fact that wildfire mitigation planning is always evolving and that jurisdictions

are frequently updating their CWPPs and wildfire mitigation guidance over time, the CWPP

Repository allows users to contribute new CWPPs to the official dataset, allowing for less

manual tracking of plans, opting instead for VGI submitted data and administrator review of all entered plans. This process ensures that the database remains as current as possible and encourages a sense of responsibility to those involved in wildfire mitigation planning. Each user has the opportunity to play a key role in the underlying database, embracing the ethos originally laid out in the HFRA.

Another set of important user interactions are the aspects of the CWPP Repository that help the user to navigate the website, and access resources to assist them with developing CWPPs. The most basic piece of the UI that helps users to navigate the web application is hover tips (Figure 19). These allow the user to view the description of a widget when they hover on it, which allows them to make a quick evaluation of its functionality.
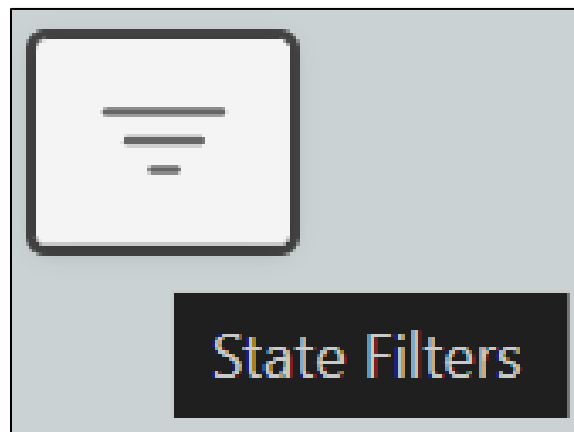
Figure 19. State filter hover tip

The other main tool available for helping to navigate the CWPP repository is a tutorial (Figure 20). This shows each widget symbol, along with an expanded description of it.
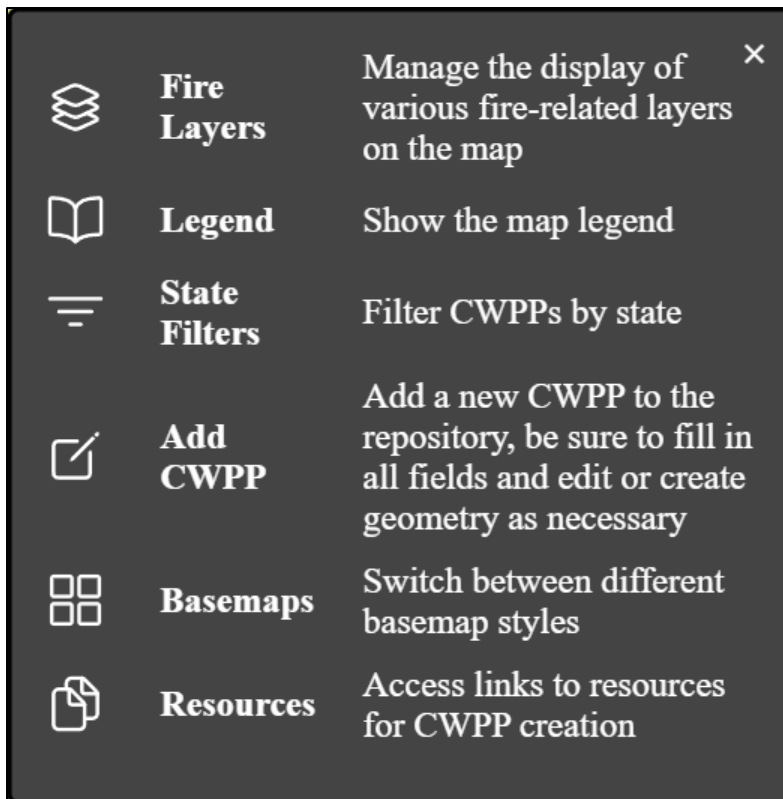
Figure 20. CWPP Repository tutorial

In the case that a user is unable to figure out any functionality of the CWPP Repository, or has a concern or a question, they can use a contact form to reach out to the website administrator (Figure 21). When a user submits a contact form the website administrator receives an automated email notifying them and providing them with the user's enquiry and contact information, and the user receives an email letting them know that they'll be contacted by the website administrator addressing their enquiry within 48 hours.

Figure 21. Contact us form

Another resource available to users in the CWPP Repository is a list of resources to assist

with CWPP development (Figure 22). This provides a set of PDFs authored by authoritative

government sources which can help natural resource managers and others develop CWPPs

specific to their jurisdiction of responsibility.

Figure 22. Resources for CWPP development

The user interactions within the CWPP Repository exemplify an approach that considers users as participants rather than just consumers. By creating an environment that gives importance to the needs, feedback, and contributions of users this platform guarantees its status as an evolving and valuable tool in the field of wildfire protection planning.

4.2 Database Efficacy

The effectiveness of any web application that handles large amounts of data, like this one, relies on the structure and eff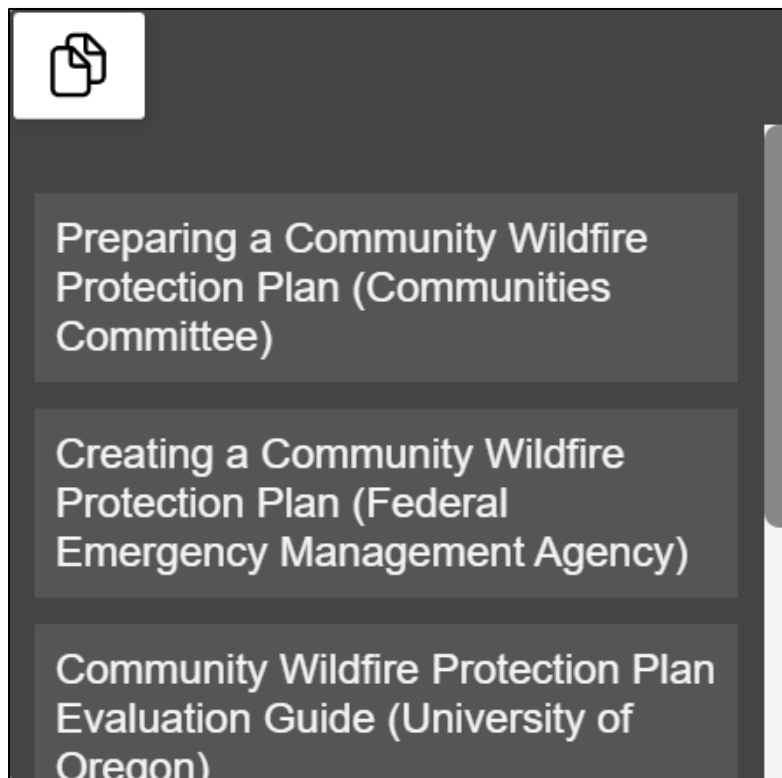iciency of the database utilized in its backend. While the visible parts of the application are the user interface and interaction mechanisms, the database is essentially its core. It works in the background to ensure that every query is handled correctly so the needed data is returned, each CWPP submitted is recorded accurately, and all interactions the user has with the web application during this process are smooth and go as expected. This section will delve into the aspects of the database that powers the CWPP Repository. It will explore the different design choices made and the challenges and successes encountered throughout the development process. It is also important to review how the database architecture ensures data integrity while also handling each piece of the data pipeline, including the pre-existing data, unconfirmed user submitted data, and all other data through the review and archiving process. This section also shows how the database responds to queries, along with how this database will scale in the future.

4.2.1 Data Integrity and Consistency

The CWPP Repository has a designed database that utilizes data validation checks to protect its integrity. These validation mechanisms are especially important when new VGI is being entered into the database. When users submit data strict input constraints are in place to identify any errors or unintended discrepancies. Tabular data like jurisdiction names and plan

details must adhere to predefined formats. This validation process ensures that all data, whether it comes from existing datasets or user submissions, is uniform and structured in the same way. This ensures consistency, which is important for any authoritative database, and ensures that users do not come across varying types and formats of data, which could cause some uncertainty of the data's quality and accuracy.

Automated emails to the administrator and user upon submission of a new CWPP move the process along. When a user initially submits a CWPP, the administrator gets an email letting them know. This email also contains "confirm" and "deny" buttons that the administrator will return to after they have reviewed the submission in the database. When either button is clicked, a trigger to an API is sent that updates the database. The user gets an email upon initial submission, confirmation the submission was received. Another email is sent to the user when the administrator selects the confirm or deny button, letting them know what action was taken. Of course, the administrator is expected to communicate with the user in the case that there are any issues with their submission, or clarification required. This ensures that all VGI data is reviewed for consistency and accuracy, and that the user is kept informed throughout the process, from submitting a new CWPP to having it confirmed or denied.

By emphasizing data integrity and consistency, as well as a review process for the VGI data, the web application can be relied upon as an accurate source of CWPPs. It is built on top of a consistent database and has the checks in place to ensure that all data is accurate, making it far easier to understand and analyze as needed.

4.2.2 Query Performance

One of the biggest challenges when developing a GIS web application that is utilizing a database with a large amount of complex structured data in it is ensuring that the user receives all

data they should within a timely manner. Whomever is accessing the web application, whether it be an interested citizen, an SME looking to build out their own CWPP, or a user wanting to submit one missing from the authoritative database, they all need to be able to access existing CWPPs, both their tabular and spatial data, quickly enough that they are not put off by an unreasonable load time. This is a huge part of the UX, as long load times can really impact the useability of a web application.

Query performance is not only about speed, but also about delivering comprehensive results that show the user whatever is needed, like every CWPP in history for any given spatial boundary, per row, for popups (so every boundary with greater than one CWPP is not polygons layered over each other). For that reason, all queries were carefully engineered to pull in the data needed for the respective circumstance, in all potential situations within the CWPP Repository.

Server side caching mechanisms were also implemented to increase performance. By storing the outcomes of executed or recent queries the system can quickly retrieve results for repeated requests without needing to access the database again. This improves response time for users in the front end and, by reducing strain on the AWS RDS instance, ensures that the web application functions smoothly for all users even during times that the user load on the application increases significantly.

4.2.3 Data Storage and Scalability

There are two main challenges when it comes to data storage. First, the database needs to be able to store a variety of data efficiently so it can be easily queried and return accurate results consistently. Secon, the database needs to be able to scale easily, so if there is an influx of CWPPs for any reason it can handle it without the website needing down time that could

interrupt the communities progress when it comes to submitting more VGI data, continuing to build out the CWPP database.

To meet these requirements, AWS RDS, running an instance of PostgreSQL with the PostGIS extension installed, was chosen as the storage solution. This cloud based platform was chosen because it is able to provide high data availability, backend functionality for the web application (AWS provides Node.js modules), and automated backups, as well as robust security measures.

AWS RDS also offers scalability. As the volume of CWPP data potentially grows over time the storage capacity can be increased without causing disruptions to the applications functionality. This ensures that the application remains agile and responsive regardless of how large the dataset becomes, making it useable deep into the future regardless of how much the community providing VGI data expands.

Another critical aspect of scalability relates to query performance. It is crucial that even as the stored data expands the time required to retrieve datasets or execute queries remains optimal. AWS RDS is able to work in tandem with other AWS services such as Elastic Beanstalk, allowing for the dynamic allocation of resources to different EC2 servers driving these cloud services. This ensures that that website performance and data access remains consistent, providing a worthwhile experience to all users that visit the web application and contribute VGI data to the underlying database stored in AWS RDS.

4.3 Data Submission

The biggest reason for the creation of the CWPP Repository is to build out an authoritative dataset of all CWPPs within the US. Previously similar work was done, but it required the administrator to manually enter all data, This section will explore in detail how VGI

plays a role in expanding the CWPP Repository's dataset while maintaining its integrity through oversight.

4.3.1 User Data Submission

In an effort to remain the most comprehensive CWPP database in the US, the CWPP Repository relies upon user-submitted data that comes through a simple interface in the frontend to tables in the backend. While traditional methods like contributions or curated datasets have their advantages, they can sometimes be limited in terms of updates and ground level insights and generally limit the future feasibility of this database, given the fact that there is not anyone tasked or responsible for tracking down new CWPP data as it is created. To address this limitation, this project harnesses the power of VGI.

Anyone that uses the CWPP Repository, whether they are wildfire management professionals or just interested citizens, has the capability to submit new. The CWPP submission process was engineered to be as user friendly as possible, offering pre-defined boundaries to reduce the manual digitization of spatial boundaries required by the user. These boundaries are listed in the standardized form where users are prompted to enter all relevant data about the CWPP, including things like the jurisdiction level, year the CWPP was published, and its PDF file.

This approach driven by VGI brings several advantages. It allows the dataset to expand, even in the absence of a dedicated team tasked with researching and identifying CWPPs not in the existing database. It also ensures that every plan is reviewed by an administrator to ensure that the data entered by a user is accurate and formatted correctly, matching all previously recorded data. This mix of user-submitted data and administrative review ensures community

collaboration that fosters a sense of ownership, and administrator review ensuring that the data in the authoritative database is accurate and formatted correctly.

4.3.2 Administrative Review

As mentioned in the previous section, while the CWPP Repository allows community inputted VGI data, it also addresses some of the inherent challenges of VGI data, namely data quality. Allowing for user submitted data acts as a strength of the web application by enabling collaboration among all interested users, however it also presents the issue of data quality and structure issues. In order to prevent these issues from degrading the quality of the underlying database a rigorous administrative review progress has been established (Figure 23, Figure 24).
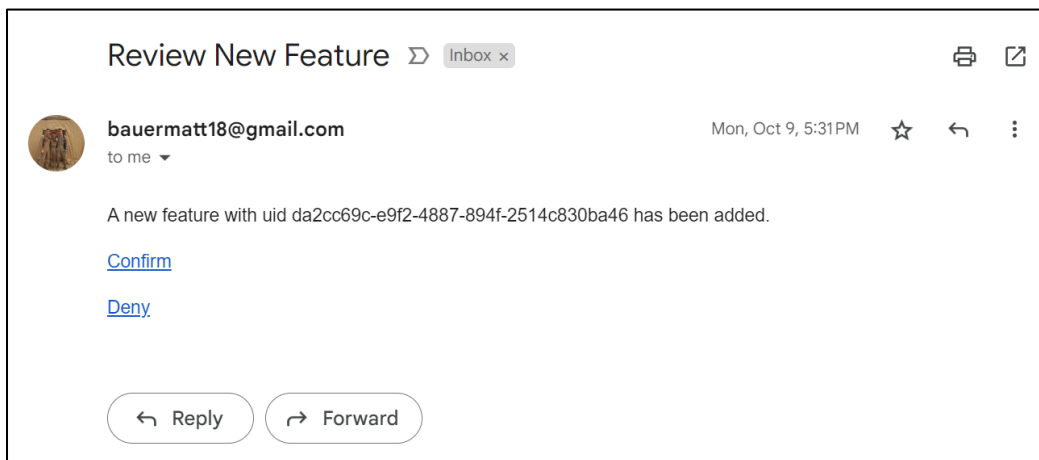
Figure 23. Email the administrator receives when a new CWPP is submitted
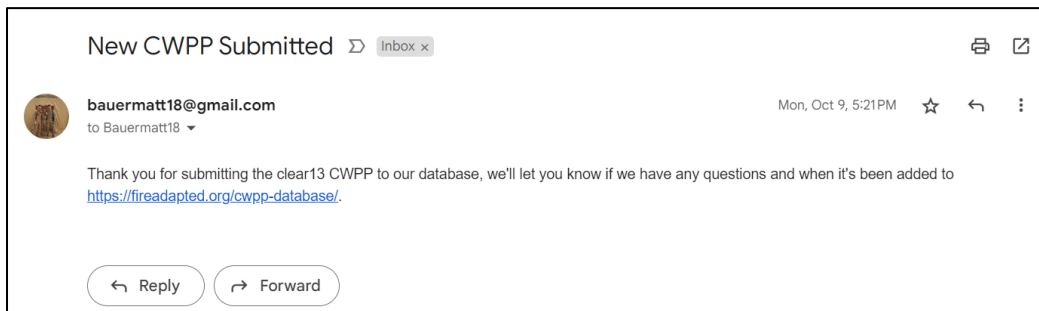
Figure 24. Email the user receives when they submit a new CWPP to the database

All data submitted by users first goes to a holding table in the database while it waits for administrative review. The node.js Nodemailer module enables automated emails to the administrator and user, ensuring communication and clear expectations between parties, namely a 48-hour review process, and easy to use buttons in the email to the administrator run queries in confirmation and denial APIs set up in the server script for the CWPP Repository. This ensures that all data remains accurate and is reviewed and confirmed or denied in a timely manner, reducing the possibility of unverified and incorrect data making its way into the authoritative database.

Once a user submitted CWPP and its respective data has been confirmed by the administrator it is automatically moved from the unverified table in PostgreSQL to the authoritative dataset. In the case it is denied, it is moved automatically from the unverified table to the archive table for denied CWPPs. This however should be rare since communication can be established with the user form the administrator when further questions are needed to clarify aspects of the VGI data.

Chapter 5 Discussion and Conclusions

Throughout the process of developing the CWPP Repository, there were a number of challenges encountered and goals attained. This chapter analyzes these experiences and results, reflecting on what this application achieves and how it can be a tool for wildfire management efforts and the centralized location for CWPP documentation. While the initial goal was to just develop a way to build out the most comprehensive database of CWPPs in the US, the CWPP Repository has turned into a thorough and complete resource for SMEs and interested citizens alike that look to gain more knowledge into wildfire planning and mitigation all the way from the local city level to the state level.

When comparing the CWPP Repository to works discussed in Chapter 2 it becomes clear that this project did not just fill gaps where they exist, but developed some new standards for VGI data collection, emergency management resources, and administrative review of data linked up to an automated data pipeline. The user friendly interface, backend infrastructure, and seamless user interactions all serve as evidence of the planning and execution put into this project. However, like any evolving platform there are areas that can be further enhanced. Looking towards the future there is plans to further develop the error logging and handling, to ensure website useability remains consistent, and to develop a custom analytic widget that lets users conduct their own analysis of CWPPs and wildfire data, which could be useful for digging into shortcomings and successes of wildfire mitigation in the United States.

5.1 Successes and Impact

The process of developing the CWPP Repository, from the beginning of just planning it out, to its current state, has been marked by a number of accomplishments and milestones, where

custom scripts had to be developed to solve problems, and each time they successfully were. This section will look into these achievements, discussing the advantages brought about by the application in terms of wildfire protection.

One of the most important successes of this application is its user-centric design. It prioritizes accessibility, interactivity, and collaboration through user submitted VGI. By doing so, it should engage a diverse user base ranging from wildfire and emergency management SMEs to citizens just interested in learning about wildfire mitigation and what is taking place at their local scale. Each unique feature implemented has added a degree of value to this application, and it can be used as a template for future endeavors that involve developing Leaflet web applications using PostgreSQL/PostGIS as the backend of them.

Beyond just technical achievements, the application's impact on the broader world of wildfire management and community planning is notable. It's streamlined access to CWPP documentation and works to foster a sense of community and collective responsibility, which is especially important given the fact that wildfire management often falls on the shoulders of small communities. By providing a central hub for information and collaboration the idea that wildfire management protection and mitigation is a shared effort that is greatly improved by community engagement.

5.1.1 Unique Features

The CWPP Repository is an example of robust interaction availability and practical usefulness. It allows the user to view and download the PDF document for every CWPP, while simultaneously analyzing their spatial boundary to see where they may overlap with other plans. This spatial aspect adds depth and understanding to the data allowing for a exploration of each

plan within its geographic region. It offers an experience where data points can be easily navigated to, zoomed in on, and understood in relation to neighboring areas.

Expanding on this foundation the application encourages community engagement through its submission form and automated notifications as well as data pipeline processes. Acknowledging that communities, professionals, and stakeholders possess current and historical CWPPs, this mechanism allows users to contribute CWPPs to the authoritative database. It transforms the platform from being static, into a repository driven by community participation. Each time a user submits a new CWPP it also helps to keep the database up to date, as well as imbuing a sense of shared responsibility and ownership. Regardless of wildfire or GIS experience, this application has something useful for anyone interested in wildfire planning and preparedness.

The CWPP Repository was designed with an emphasis on simple and engaging UI. The interface mainly relies upon custom widgets created with Leaflet that accomplish a variety of things. They allow users to add data layers to the map, change the basemap, filter CWPPs by state, and access some help in the form of a short tutorial and contact form that can be used to contact the website administrator. The filter widget allows users to instantly zoom to their state of interest, which also filters all data to only that state to speed up website interactions. When users are done with a state they can clear the filter which then quickly adds all CWPPs back to the map and zooms out to the full extent. These interactions all depend on cached data, allowing them to always run smoothly and quickly even when a user has a poor internet connection.

5.1.2 Broader Implications and Impact

The creation and subsequent development of the CWPP Repository goes beyond its technical achievements. It could have impacts on community engagement with wildfire

management, and community wildfire preparedness more generally through better informed SMEs.

First and foremost, this platform has given access to CWPPs to a much larger audience than when they were dispersed within many smaller collections. Spreading access to these plans ensures that communities, local governments, and individuals all can understand and take actions based on these plans. It provides resources in the form of historical CWPPs, and other resources for CWPP creation. Increasing accessibility of data is an especially important thing in disaster preparedness, where a lack of collaboration and access to data can have dire consequences.

The CWPP application also serves as a catalyst for community driven data collection and sharing. Its CWPP submission feature emphasizes the importance of decentralized sources of knowledge. This not only keeps the platform up to date and comprehensive but fosters a sense of ownership and empowerment among users. This is particularly important in the field of wildfire management where shared responsibilities can be complex and unclear, with fires burning across jurisdiction lines often causes issues between responsible agencies.

As cities expand and human settlements go deeper into the current wildland WUI wildland fires will likely only continue to become a more pressing and dangerous issue. Tools like the CWPP Repository play a role in bridging knowledge gaps and providing stakeholders with insights and resources to navigate these challenges. By fostering an understanding of vulnerabilities, resource allocation, and strategic planning, this platform can have a positive impact on institutional cooperation and collaboration when it comes to managing and mitigating wildfires.

The success of this application highlights the importance of incorporating technology into disaster preparedness and community engagement efforts. In today's era platforms like these act as connections between people and information/resources.

5.2 Limitations

While the CWPP Repository is undoubtedly an achievement in the field of wildfire protection planning and community engagement, it is crucial to approach its evaluation with a balanced perspective. Every technological project has some limitations. Acknowledging and addressing these limitations ensures transparency and provides a roadmap for future improvements and enhancements to the CWPP Repository.

This section gets into some of the challenges encountered during the development and implementation of the application and areas where the platform may not fully realize its potential. Challenges related to user participation and data accuracy offer valuable insights into the complexities involved in building and maintaining a digital platform within an ever evolving domain, particularly one that allows VGI inputted data to all users. By examining these areas for improvement, there is space for feedback informed iterations and an ongoing commitment to making the platform even more robust and impactful in years to come.

5.2.1 Recognized Shortcomings

One notable challenge relates to scaling limitations. Despite utilizing AWS cloud services that allow for auto-scaling the applications infrastructure and servers, there is still the potential that the web application could experience performance issues if the user base expands rapidly for any reason. As the application scales over time, it becomes critical to ensure that the application remains responsive and efficient. The initial designs and hosting solutions may need reevaluation to accommodate the increase in traffic and data. Additionally, while the geospatial features

enhance user experience, they also introduce some complexities. Ensuring data accuracy and uniformity, maintaining rendering speeds for visualizations, and ensuring cross platform compatibility can be challenging as the platform evolves and integrates more advanced functionalities.

Another aspect worth considering is data accuracy and completeness. The platforms strength lies in its submission mechanism that encourages users to contribute CWPPs. However, this decentralized approach to data collection brings completion and validation challenges. The CWPP Repository will only be complete if users add all existing CWPPs. Thus, it is crucial that those who know about new CWPPs also know about the CWPP Repository. Reaching out to the wildfire management community to share news of this project is important. Further, every effort has been made to ensure that the plans submitted are authentic and accurate before they are added to the database. However, since the content is generated by users and verified by a human, there is always a possibility of discrepancies or incomplete data.

Engaging with a user base brings its own set of challenges. Creating an interface that caters to both professionals who are familiar with GIS data and community members seeking basic information, requires a strategically created UI that enables both groups of users to have a worthwhile experience in this web application. Although the current design intends to be as intuitive and user friendly as possible, there is always room for improvement based on user feedback and evolving practices.

5.2.2 Logging Mechanisms

Logging systems play an important role in web applications, especially those that handle large amounts of data and user interactions like the CWPP application. Effective logging

provides information about user behavior, system performance and areas that may need attention

allowing for interventions and continuous improvements.

Currently the application uses some basic logging in the JavaScript to capture system

events and user interactions. These logs are vital for diagnosing issues and understanding how

users navigate through the platform, helping to ensure its continued wellbeing. However, as the

platform has grown and evolved over time it has become evident that a more comprehensive

logging mechanism is necessary to meet the expanding needs of the application.

One area of concern relates to log granularity. While current logs cover system events

there is a need for more detailed logs specifically related to user interactions with geospatial data,

submission processes and feedback mechanisms. Detailed logs would not enable troubleshooting

but also provide deeper insights into user behavior to enhance the overall UX.

Versioning is also a concern. Right now, there is no versioning system setup for the

PostgreSQL data. This is something that could be improved significantly in the future and would

ensure that backups exist in case of any issues that may arise with the database tables.

Additionally, real time monitoring and alerting mechanisms are crucial. As the number of

users increases and the platform handles interactions being able to detect anomalies or system

bottlenecks in real time becomes essential. To do this at some point AWS CloudWatch will be

integrated with the Elastic Beanstalk deployment to ensure real time monitoring and alerts are

available.

5.3 Future Enhancements

Like any platform there are always opportunities for growth, refinement, and expansion.

The CWPP Repository is intended to be a continually evolving project, with new features being

added as they are needed and continuous maintenance and refinement to ensure that it keeps up with all changes in web development techniques that take place into the future.

These enhancements demonstrate a commitment to keeping this platform relevant and user focused. Whether it is improving the UX, integrating data analytics, expanding outreach efforts, or entirely revising what a CWPP is in the first place, each enhancement aims to empower communities, professionals, and individuals with insights on wildfire protection that are accurate and actionable.

5.3.1 Welcome Pane

As digital platforms continue to progress the importance of an intuitive user onboarding process becomes more evident. A structured initial interaction can have an impact on how users perceive, engage with, and experience a platform. Introducing a welcome pane to the CWPP Repository is seen as a future enhancement to address this need.

The envisioned welcome pane will be designed to be interactive and dynamic. Its purpose will be to guide users into the platform's ecosystem, providing them with an understanding of its capabilities and offerings. For those who are new or unfamiliar with GIS web applications, a fully geospatial application with a wealth of data can be complex and intimidating. The welcome pane should reduce these issues by highlighting the CWPP Repository's features and demonstrating how to navigate through it effectively. It also should describe some of the data available and give a brief overview of its purpose and what it shows.

In addition to covering the basics the welcome pane will also provide users with a sense of purpose and context. By offering a concise narrative about the platform's role in wildfire protection and its importance for community collaboration, users can immediately grasp its

value. This context is crucial not just for engagement, but also for fostering a sense of community and shared goals.

As the platform keeps progressing and incorporating new functionalities the welcome pane can also act as a log of updates, giving returning users an overview of any recent additions or modifications. This way users that continue to utilize this web application overtime can easily stay up to date as it evolves and changes in the future.

5.3.2 Analytics

The true value of data is not just in gathering it and centralizing it, but also in interpreting it. In recent years, society has begun to realize the true value of large data stores, and analysis of data has become the backbone of many fields and aspects of human life. In the realm of CWPPs, wherein data plays a role in guiding important community actions, analytical tools that help a user draw conclusions from data can have a large impact. For this reason, the highest priority improvement to the CWPP Repository is an analytics widget that is modular and easy to use.

The analytics widget should go beyond just standardized graphs and charts. Instead, it should transform raw data into worthwhile insights. Its goal should be to let users explore the data with a variety of methods, including the visualization of trends, the comparison of variables, and the identification of patterns that can help to inform decision making and understanding of CWPPs and wildland fire characteristics in the US. Understanding the temporal nature of CWPPs, made possible by a robust archive of plans, would be another good way to identify where jurisdictions are falling short, or where they are doing a great job with wildfire planning, which could be considered when it comes to allocating funding for wildfire planning and mitigation.

This widget should also be interactive and highly customizable, it should include a variety of methods to customize and analytic insights exactly to the user's needs and desires, for all data housed within the CWPP Repository's database. This could allow for more precise filtering of data and end-insights delivered, to exactly support the user's needs. This flexibility ensures that users can tailor their analytics according to their needs and inquiries.

Another benefit of developing an analytics widget is that it could be designed to blend in with all the other features of this web application. Users should be able to explore data on the map and extract insights simultaneously, with each process having its own pane seated on top of the map view. Integrating it like this helps to ensure that trying to utilize one tool does not completely throw a user off path from what they were working on with another tool.

The introduction of the analytics widget will help to show the platform's effort to not just be a data repository, but to also serve as a resource that helps to empower users with a variety of tools that facilitate greater understanding of wildfire management. It signifies a step towards transforming the platform from being a data repository, to becoming a comprehensive analytical tool that caters to the diverse needs of its user community as they grow and evolve over time and into the future.

5.3.3 CWPP Format

Another way this project could be used to improve future wildfire mitigation efforts is by redefining what a CWPP is entirely. Right now, CWPPs have their basic standard requirements discussed earlier, and they are typically a PDF document. In the future CWPPs could be transformed from a PDF into a spatialized, purely digital format. These plans could still maintain the integral aspects of a classic CWPP, just all in a spatial format. It is not hard to imagine how this could be done: areas targeted for mitigation efforts and the funding allocated for each could

easily be stored in a geodatabase as polygons, and all analysis and historical review and such could also be spatialized, since CWPPs consist of information that has clear spatial characteristics.

This transformation could make collaborating on CWPPs much simpler, with living spatial datasets being developed and evolving over time as a CWPP planning committee refine them. They also could be done in a uniform, standardized way, making aggregation of them in a central location, like the web application developed for this project, much simpler. In addition to that, the spatial data could be easily viewed in GIS software and geospatial web applications, making it much easier to identify historical wildfire mitigation measures taken for any given boundary, giving SMEs a much clearer picture of all mitigation measures done in an area for the recorded history of it. Analysis at any scale would be much more accessible with nationally aggregated data, making it easier to conduct analysis and identify any shortcomings in CWPP development, or any clear successes. This would be an excellent step forward, moving CWPPs into a more easily analyzed and clearly delineated plan format.

References

Balch, J.K., B.A. Bradley, J.T. Abatzoglou, C.R. Nagy, E.J. Fusco, and A.L. Mahood. 2017. "Human-Started Wildfires Expand the Fire Niche Across the United States." *Proceedings of the National Academy of Sciences* 114, no. 11: 2946-2951.

Basiouka, S., C. Potsiou, and E. Bakogiannis. 2015. "OpenStreetMap for Cadastral Purposes: An Application Using VGI for Official Processes in Urban Areas." *Survey Review* 47, no. 344: 333–341.

Brown, E. (2019). *Web Development with Node and Express: Leveraging the JavaScript Stack*. Sebastopol: O'Reilly Media.

Carlson, B. 2023. *Node-Postgres*. Accessed November 5, 2023. https://node-postgres.com/.

Christie, M. 2020. "Positivity Rate in Indiana Counties." *COVID-19: Indiana Counties*. Accessed December 3, 2023. https://machristie.github.io/covid-19-in-counties-positivity-rate/.

Cybersecurity and Infrastructure Security Agency. 2023. "Wildfires." *Cybersecurity and Infrastructure Security Agency*. Accessed July 16, 2023. https://www.cisa.gov/topics/critical-infrastructure-security-and-resilience/extreme-weather-and-climate-change/wildfires.

Farkas, G. 2017. "Applicability of Open-Source Web Mapping Libraries for Building Massive Web GIS Clients." *Journal of Geographical Systems* 19, no. 3: 273-295.

Healthy Forests Restoration Act of 2003. Pub. L. No. 108-148, 117 Stat. 1887, 1887-1915. 2003. https://www.govinfo.gov/app/details/STATUTE-117/STATUTE-117-Pg1887/summary.

Huy, T. 2021. "Build and Deploy a High-performance Full Stack JavaScript Web Application."

Jakes, P.J., and V. Sturtevant. 2013. "Trial by fire: Community Wildfire Protection Plans put to the Test." *International Journal of Wildland Fire* 22, no. 8: 1134-1143.

Jones, G.M., A.R. Keyser, L.A. Westerling, J.W. Baldwin, J.J. Keane, S. C. Sawyer, J.D. Clare, R.J. Gutiérrez, and Z.M. Peery. 2022. "Forest Restoration Limits Megafires and Supports Species Conservation under Climate Change." *Frontiers in Ecology and the Environment* 20, no. 4: 210-216.

Krause, J. (2017). *Programming Web Applications with Node, Express and Pug*. Berkeley: Apress.

Lata, K., A. Sood, K. Kaur, A. Benipal, and B. Pateriya. 2022. "Web-GIS based Dashboard for Real-Time Data Visualization & Analysis using Open Source Technologies." *Journal of Geomatics* 16, no. 2: 134-146.

Obe, R.O., and L.S. Hsu. (2017). *PostgreSQL: Up and Running: A Practical Guide to the Advanced Open Source Database*. Beijing: O'Reilly Media.

— (2021). *Postgis in Action*. Shelter Island: Manning.

Oregon Department of Forestry. 2023. "Community Wildfire Protection Plan." *Oregon Department of Forestry*. Accessed July 16, 2023. https://www.oregon.gov/ODF/Fire/Pages/CWPP.aspx.

Palsa, E., M. Bauer, C. Evers, M. Hamilton, and M. Nielsen-Pincus. 2022. "Engagement in Local and Collaborative Wildfire Risk Mitigation Planning across the Western US—Evaluating Participation and Diversity in Community Wildfire Protection Plans." *PLoS One* 17, no. 2: e0263757.

Permann, A. 2022. "Community Wildfire Protection Planning Workshops: Sept 14th and 15th." *Yolo County Resource Conservation District*. Accessed December 3, 2023. https://yolorcd.org/community-wildfire-protection-planning-workshops-sept-14th-and-15th/.

Piñol, J., J. Terradas, and F. Lloret. 1998. "Climate Warming, Wildfire Hazard, and Wildfire Occurrence in Coastal Eastern Spain." *Climatic Change* 38, no. 3: 345-357.

Radmall, L. 2004. "President George W. Bush's Forest Policy: Healthy Forest Restoration Act of 2003." *Journal of Land, Resources & Environmental Law.* 24: 511.

Satyal, A. 2020. "Designing and Developing a Website with ReactJS: Progressive Web Application."

Thompson, M.P., E.J. Belval, J. Bayham, D.E. Calkin, C.S. Stonesifer, and D. Flores. 2023. "Wildfire Response: A System on the Brink?" *Journal of Forestry* 121, no. 2: 121–124, https://doi.org/10.1093/jofore/fvac042

Vliet, J. (2011). *Elastic Beanstalk*. Beijing: O'Reilly Media.

Yang, Z., J. Li, J. Hyyppä, J. Gong, J. Liu, and B. Yang. 2023. "A Comprehensive and up-to-date Web-Based Interactive 3D Emergency Response and Visualization System using Cesium Digital Earth: Taking Landslide Disaster as an Example." *Big Earth Data*.