Invasive Alien Plants Programme | Narrowleaf Firethorn

Project Execution Plan

IAP-NLFT-S001-001

May 2024

Team Members:

Ali Yalcin	PXL(Belgium)	12203001
Massimo Sarli	PXL(Belgium)	12200638
Zegert Nieuwoudt	BC(Pretoria)	577711
Zoë Treutens	BC (Pretoria)	577989

TABLE OF CONTENT

1. Project Details

1.1 Project Scope

Develop a web application for identifying invasive species with the goal to decrease the ecological damage Pyracantha angustifolia causes in South Africa (SA), particularly in Gauteng, focusing on the specific coordinates within Gauteng: -28.544878491189962, 28.418751663332362.

Key Components for Success:

- **Research**: Conduct thorough research on the ecological and environmental consequences of narrowleaf firethorn to help direct successful mitigation actions.
- **Integration of Google Street View**: Use the app to connect to Google Street View satellite data to get images that may be used to identify various the Firethorn invasive species.
- **Data Collection**: Gather a variety of natural flora and Pyracantha Angustifolia images from Google Street View to use as training and testing data.
- **Selecting Algorithms**: Use appropriate machine learning and image processing algorithms to accurately identify the invasive species.

The following metrics can be used to assess the project's success:

- **Precision:** To successfully distinguish Pyracantha angustifolia from native vegetation, a high degree of precision in species identification needs to be achieved.
- **Usability:** To ensure that stakeholders can utilize the web application for monitoring and mitigation purposes, make sure it is user-friendly and accessible.
- **Scalability:** Make the software as scalable as possible so that it may be extended to more locations where invasive species pose a threat.
- **Engagement of stakeholders:** Encourage participation from relevant stakeholders, including locals, researchers, and environmental authorities, to gain support and ensure the project's long-term sustainability.

1.2 Roles and Responsibilities

Massimo Sarli: As a student studying Artificial Intelligence at PXL, Massimo Sarli contributes to the project by leveraging his expertise in AI technologies.

Ali Yalcin: Specializing in Application Development at PXL, Ali Yalcin plays a crucial role in the implementation of the web application.

Zegert Nieuwoudt: Pursuing studies in Software Engineering at Belgium Campus, brings his skills in software development to the project.

Zoë Treutens: With a focus on Business Intelligence at Belgium Campus, she contributes to the project by providing insights into data analysis and visualization.

1.3 Stakeholders

Stakeholders are individuals or organizations, who influence the results or actions of a project and have a personal stake in its success. They include everyone who is either directly or indirectly involved in the project.

List of Stakeholders for the Project:

- **Project supervisor:** Mr. Dino Giovannoni, from the Technology Aided Biocontrol Group (TAB) at Belgium Campus ITversity, oversees the project's progress and offers guidance.
- **Technical support**: Mr. Francois Venter, affiliated with the Technology Aided Biocontrol Group (TAB) at Belgium Campus ITversity, provides technical assistance and expertise in project implementation.
- **Project sponsor:** Prof. Grant Martin, associated with the Centre for Biological Control (CBC) at Rhodes University, provides institutional backing for the project's activities.

1.4 Project Schedule

The project is scheduled from 22 February 2024 until 30 May 2024. Over this period, several significant milestones will mark its progress toward completion:

Project Proposal Presentation (14 March 2024): The presentation of the project proposal, which details the objectives, scope, and methodology for developing a web application for identifying invasive species, with a focus on Pyracantha angustifolia in South Africa.

Field Implementation (15-20 April 2024): This phase is dedicated to the project's practical implementation, which involves fieldwork for data collection, testing, and validation of the web application's functionality. Fieldwork will focus on a particular location in Gauteng (-28.544878491189962, 28.418751663332362) to determine the presence and density of Pyracantha angustifolia.

Final Project Presentation (30 May 2024): The final project presentation will focus on the built web application for identifying invasive species. This presentation will discuss the project's findings, methodologies, and potential impact on minimizing the environmental damage caused by Pyracantha angustifolia in South Africa.

2. Introduction

2.1 History

Invasive plant species represent a significant worldwide issue given that they endanger ecosystems to such an extreme degree. Whether purposely or inadvertently introduced for a variety of purposes, such as ornamental or agricultural, these non-native plants have the capacity to rapidly spread and compete with native flora. Their lack of natural controls and predators, causes devastation on nearby ecosystems, changing habitats, reducing biodiversity, and costing money. Narrowleaf firethorn, or Pyracantha angustifolia, is one such invasive plant that originates in Asia and is indigenous to Japan and China (Wikipedia, 2024). Originally introduced to South Africa as an ornamental shrub, it expanded swiftly and developed aggressive behaviours, forcing several regions to classify it as an invasive species (Rebelo & Gilmore, 2024).

Two distinctive features of Pyracantha angustifolia are its glossy, evergreen leaves and clusters of small white flowers that change into vibrant orange or red berries in the fall. While they may be aesthetically appealing, these berries are toxic to humans and animals alike if eaten. The dense, thorny branches of the shrub provide shelter for birds and help disperse seeds (Gardenia, 2024). In South Africa, Pyracantha angustifolia has successfully naturalized in a range of habitats, including grasslands, forests, and disturbed areas. Numerous native plant species are threatened by its unrestrained growth, which changes ecological structures and disrupts ecosystem processes.

Despite being an invasive species, Pyracantha Angustifolia has several benefits for the environment. Its berries serve as a vital source of nutrition for birds, particularly in the winter when there is a serious shortage of food. Its dense growth patterns are sometimes used as security hedges in metropolitan areas because they actively discourage trespassers (Gardenia, 2024). However, these advantages pale in comparison to the destruction that its unrestrained development has caused to the environment.

Pyracantha Angustifolia serves as an example of caution concerning the unforeseen consequences of introducing non-native plants into strange environments. It is necessary to reduce invasive species by diligent monitoring and practical management strategies to preserve the integrity of native ecosystems.

2.2 Research Problem

South Africa's delicate ecosystems are at risk by the invasive Pyracantha Angustifolia, commonly referred to as narrowleaf firethorn, because of its unregulated growth that takes over native plant species. To effectively combat this issue, it is imperative to create a web application that can identify Pyracantha Angustifolia from images taken through Google Street View and generate a density map that displays the widespread presence of this invasive species along South Africa's roadsides, particularly in Gauteng.

2.3 Research Questions

What is the efficacy of using Google Street View data combined with image recognition algorithms to reliably detect and map the spread of Pyracantha Angustifolia shrubs across different terrain in South Africa?

- Why is it important to detect and map the spread of Firethorn?
- Who and what does the spread of Firethorn threaten?
- What are the technical requirements for developing a web application capable of processing images and generating density maps?
- What are the best frameworks for implementing mapping functionalities in the web application?
- What are the unique visual features of Pyracantha Angustifolia that can be consistently detected by image processing techniques?
- How can one reliably extract data from Google Street View to be used for image recognition?

- How can image processing algorithms be trained to reliably distinguish Pyracantha Angustifolia from other species in Google Street View images?

2.4 Impact and Significance

2.4.1 Ecological Impact

- **Loss of Biodiversity:** The Firethorn species fiercely rivals native plant species for resources, which frequently results in the eradication and dwindling of native flora. Thus, the entire ecosystem is impacted, including the wildlife that depends on native plants for habitat and food.
- **Habitat Alterations:** The thick growth of the species has the potential to change habitats' structural characteristics, rendering them unsuitable for native wildlife. This can result in a decrease in species diversity by upsetting ecological processes.
- **Ecological Imbalance:** The growth of firethorn can disrupt natural equilibriums by offering certain species food and shelter while excluding others. This could lead to an overabundance of certain kinds of animals and further ecological deterioration (Canavan, et al., 2022).

2.4.2 Economic and Social Impact

- **Management Expenses:** Firethorn management and removal cost a substantial amount of funding. This covers the price of chemical treatments, mechanical removal, and continuous monitoring.
- **Impact on Agriculture:** When invasive species spread into agricultural land, the productivity of agricultural land and crop yield decreases. The agricultural economy and nearby farmers may suffer as a result.
- **Health Risks:** Although consuming excessive amounts of the berries of the firethorn plant may induce moderate stomach irritation, they are not thought to be hazardous to people or animals.
- **Damage to Infrastructure:** The thick, prickly growth might impede roads, walkways, and other infrastructure, increasing the expense of upkeep and repairs (Chari, et al., 2020).

2.4.3 Significance the Project

The health and balance of South Africa's ecosystems are dependent on controlling the Pyracantha angustifolia invasion. The project's objective is to create a web application that will precisely identify and map the spread of this invasive species using photos taken from Google Street View, therefore reducing the ecological damage this plant poses. An effective tool for researchers, environmental authorities, citizens scientists, educational institutions, botanists, people of areas affected by invasive plants, and researchers will be made available via the application, which will precisely identify and map the spread of this species.

By enabling users to follow the spread and density of firethorn throughout Gauteng, the tool will improve monitoring. Additionally, it will provide insightful data to help prioritize mitigation activities and make sure resources are directed toward the regions most impacted by the

Firethorn invasion. The application will assist in biodiversity conservation and expedite urgent intervention efforts by facilitating the rapid and accurate detection of invasive species.

The technological specifications needed to develop a web application that can analyse photos and generate density maps to identify invasive species will be understood by users. Students will gain knowledge about the best frameworks for integrating mapping features into web applications as well as the various image processing methods used to recognize the distinctive visual traits of Pyracantha Angustifolia and distinguish it from other invasive or native species in South Africa.

The initiative also intends to raise stakeholder participation and knowledge the significance of ecosystem management and the effects of invasive species. Additionally, this study lays the foundation for the development of a comprehensive website in the future that will be able to identify additional invasive species worldwide, therefore extending its impact outside of South Africa.

3. Research Design

3.1 Methodology

The best methodology for this project will be a mixed-method approach. To give a thorough grasp of the issue and to create a reliable online application for locating the effects of the invasive Pyracantha angustifolia in South Africa, this technique combines qualitative and quantitative research methodologies.

3.1.1 Qualitative Approach

Research and Literature Review

This is compiling a body of knowledge on Pyracantha Angustifolia's effects on the environment and ecology. To comprehend the origins, distribution, and effects of the invasive species, the qualitative component will involve reading scientific publications, environmental reports, and case studies.

Stakeholder Engagement

Interviewing and holding focus groups with pertinent stakeholders, such as researchers, environmental authorities, and agricultural specialists. This will guarantee that the web application satisfies user demands by highlighting the requirements and practical challenges.

3.1.2 Quantitative Approach

Data Collection

Images from street views and Google photos are gathered on the different types, shapes, and colours of the Pyracantha angustifolia presence in Gauteng.

Algorithm Development

The gathered photos are analysed using image processing and machine learning techniques. These algorithms will be trained, tested, and validated quantitatively to make sure they correctly detect the invasive species even when a different species of plant is shown.

Mapping and Analysis

Creating a map to visualise the location of the Pyracantha Angustifolia species.

Through the integration of quantitative data and qualitative concepts from stakeholders, the project guarantees accurate species identification and user-friendly design. Using cutting-edge statistical and computational methods, this strategy guarantees solid data analysis while also enhancing the validity and dependability of the study findings. This strategy places a strong emphasis on engaging stakeholders to meet their goals and concerns, build community support, and guarantee the project's long-term viability. The mixed-methods approach's adaptability and flexibility enable it to evolve as the project continues, with newly formed qualitative insights influencing the gathering and analysis of quantitative data and vice versa.

Thus, by employing a mixed-methods approach, the project team may effectively address Pyracantha Angustifolia's ecological challenge in South Africa. This methodology guarantees that the project is based on sound research, formed by feedback from stakeholders, and able to produce accurate and useful outcomes, all of which contribute to a thorough and efficient solution.

3.2 Procedure

3.2.1 Tools & Resources Required

These resources and tools are essential to the project at each phase, from planning and ideation to coding and documentation. Each tool has been selected based on unique characteristics that facilitate the project procedures. They guarantee that the project's goals are achieved successfully and economically in addition to increasing productivity. The readers are given details in this section about the tools' intended usage in the project and how they contribute to a productive and effective workflow.

Visual Studio Code

Microsoft created Visual Studio Code (VS Code), a lightweight yet capable code editor that is free and open source. It has syntax highlighting, IntelliSense code completion, and a wide range of extensions for debugging, version management, and language support. VS Code offers developers a flexible environment for creating, editing, and debugging code across several programming languages and platforms, thanks to its built-in terminal, Git integration, and debugging features, as well as customisation possibilities and cross-platform compatibility.

PhpStorm

PhpStorm is the development environment that will be used in this project. It has bult-in support for Vue.js (the framework that will be used) and integrates with the Vue CLI. It provides debugging tools which help the development team identify and fix the issues more efficiently. PhpStorm’s rich ecosystem of plugins extends its functionality and allows the development team to make better use of specific extensions.

Teams

Microsoft Teams serves as the primary communication and coordination platform for our team's conceptualization, planning, and creation of the project's outline and prototype. Teams dynamic interface enables members to effortlessly participate in real-time discussions, exchange ideas, and share important documents. Through Teams' interface with Microsoft Office applications, the team will work efficiently on the project proposal paper and presentation, streamlining the planning process. Furthermore, scheduled meetings will allow for in-depth discussions, progress reports, and feedback sessions, fostering a sense of connection and accountability among team members.

GitHub

GitHub will be the environment for all team members to work on the coding of the project's prototype. Using its robust version control system, the team will easily share and maintain their code, assuring seamless collaboration. Each member will have real-time access to the most recent updates and changes, allowing the team to stay on track and work more effectively toward the project objectives. By leveraging GitHub's features, the team will be able to successfully optimize their workflow, increasing productivity and speeding up the progress of their project prototype.

Confluence

Confluence is collaboration software developed by Atlassian, aiding teams in efficiently sharing knowledge and working together. It offers features such as page creation for documentation and projects, organizing content into spaces, commenting and feedback, and version control and access management. With powerful search functionality, users can quickly locate relevant content. It's a platform facilitating teamwork by providing a central hub for knowledge sharing and project management.

3.2.2 API and ML Model Usage Procedure

This detailed guide describes how to configure, test, train, and deploy a machine learning model for image categorization by way of an API to make it simple for other researchers to replicate.

Step 1: Set up the Environment

Objective: Set up an appropriate environment and make sure all tools and libraries are available.
Estimated Result: An environment ready to execute the code with no external components missing.

1. Install Python and VS Code:
    - If Python or Visual Studio Code is not installed already on to the machine, follow the instructions on the Python and VS Code home page to install the different environments.

2. Install Extensions in VS Code:

- Installing the following extensions makes it possible to take full advantage of VS Code's Python environment.
    - ***Python:*** For Python development, this extension offers functionality including code formatting, debugging, and linting.
    - ***Pylance:*** This language server offers extensive features for Python, including code navigation, type checking, and IntelliSense.
    - ***Vue:*** This addon offers snippets, syntax highlighting, Emmet support, linting/debugging, and more if you work with Vue.js.

3. Install and Load Python Libraries:
    - Python programmers may create online applications with **Flask**, a lightweight web framework that manages HTTP requests. To install it, use the command prompt *pip install flask* .
    - Neural network model designing and training are made possible by **Keras**, a high-level neural network API that can operate on top of TensorFlow. . To install it, use the command prompt *pip install keras*.
    - **TensorFlow** is an open-source machine learning platform that offers resources and tools for developing, building, and implementing ML-powered apps. To install it, use the command prompt *pip install tensorflow*.
    - The **NumPy** library is commonly used to handle mathematical operations like as statistical operations and linear algebra operations, as well as numerical operations and array manipulation. To install it, use the command prompt *pip install numpy*.

Step 2: Train the Model

Objective: Train a Convolutional Neural Network (CNN) to identify different types of pictures and save the trained model for future deployment.

Estimated Result: A trained model saved to a file.

1. Prepare the Dataset:
    - Images are arranged in a folder with subfolders, each of which should represent a different category of image.
2. Augment and Preprocess Images:
    - To enhance the training images, use image data generators like rescaling, shearing, zooming, and flipping to increase the generalization capability of the model.
3. Define and Compile the Model:
    - Set up a Sequential CNN model with layer for feature extraction, pooling for down sampling, a flattening layer, and dense layer for classification.
    - Making use of an optimizer and a suitable loss function to compile the model.
4. Train the Model:
    - Utilize the pre-processed dataset and train the model spanning many epochs so that the validation accuracy and loss can be monitored during the training phase.

5. Save the Model:
    - After training is finished, save the model as a classifier.keras file.

Step 3: Set Up the API

Objective: Use Flask to build an API that can process uploaded images and return a predicted class.

Expected Outcome: A functioning API service that can accept image submissions and returns a class prediction.

1. Initialize Flask Application:
    - Create an app.py file that contains a Flask application.
    - Turn on CORS (Cross-Origin Resource Sharing) to accept requests originating from multiple IP addresses.
2. Define API Endpoint:
    - Create an endpoint that accepts POST requests.
3. Handle Image Upload:
    - Verify that the endpoint can accept an image file, saves it in the upload's directory, and check if the request included an image file.
4. Integrate Prediction Function:
    - To determine the class of the uploaded picture, use a method that loads the stored model and maps the prediction to a class label that is readable.
5. Return Prediction:
    - Return the anticipated class label as a JSON message.

Step 4: Image Processing Functions

Objective: Implement preprocessing algorithms for images and make predictions using the model.

Expected Outcome: Functional image processing tools to assist with the smooth operation of the API.

1. Load the Trained Model:
    - The model saved in the classifier.keras is loaded in an image processing python file.
2. Preprocess the Images:
    - Define a function that loads an image, transforms its pixel values, and resizes it to the model's required input size.
3. Predict Image Class:
    - Make a function that will return the expected class probabilities and use the pre-processed image as input to the model and convert it to a class label.
4. Extract Metadata (optional):

- This step is optional and can be used to create a function that uses EXIF data to retrieve GPS metadata from the uploaded images.

Step 5: Test the API

Objective: Verify if the API interprets the images accurately and provides accurate predictions.

Expected Outcome: Successful image uploads and accurate class predictions returned by the API.

1. Run the Flask Server:
    - Run the app.py file to open the Flask application.
2. Test the API with a Sample Image:
    - Send POST requests with an image file using programs like Postman or curl.
    - Verify that the JSON's response from the API has the accurate anticipated class label.

Researchers may develop an API, train a model, set up an environment, and make predictions using images by meticulously following these steps. This guarantees that the approach may be repeated and verified by other scholars.

3.2.3 Displaying Model Predictions on a Web Application Procedure

This detailed guide outlines how a web application is set up using Vue to display a machine learning models predictions. The steps are made to make it simple for other researchers to replicate them.

Step 1: Set Up the Environment

Objective: Create an appropriate setting and make sure that all the tools and libraries required for developing with Vue.js are available.

Expected Outcome: A prepared environment with all the requirements needed to launch the Vue.js application.

1. Install Node.js, npm and Vue.js:
    - Make sure the system has Node.js, npm (Node Package Manager) and Vue CLI installed. If the system does not have it installed, follow the links to the NodeJS and Vue CLI homepages for installation instructions.
2. Create Vue.js Project and Install Required Libraries:
    - Create a project and use npm to install any required libraries for that project, such leaflet and axios.

Step 2: Set Up Vue Router

Objective: Set up Vue Router to control how users navigate the web application.

Expected Outcome: A router set up to provide access to different application views.

1. Create Router File:
    - Create a file called index.js for the project.
2. Configure Routes:
    - Specify the paths the program will take by creating a route for the MapView component.
3. Set Up History Mode:
    - To ensure cleaner URLs, use createWebHistory to activate history mode in the router.

Step 3: Create the MapView Component

Objective: Develop a MapView.vue component to manage the uploading of images, show locations on a map, and show projections.

Expected Outcome: A functional component with the ability to display map locations, upload images, and create and display predictions.

1. Set Up Template:
    - Outline the HTML structure, considering the parts for coordinate input, map display, image upload, and image display.
2. Handle Image Uploads:
    - Provide a file input for users to upload images.
    - Read the uploaded image and display it in a div component.
3. Upload Image and Display Predictions:
    - Create an interface for transferring the submitted image to the backend API so that Axios may be used for prediction.
    - Show the predicted plant name on the website or in an alert.
4. Coordinate Input and Map Display:
    - Create a Leaflet function that uses the coordinates provided by the user to display the location of the plant sighting on google maps as a pin.

Step 4: Configure the Backend API Integration

Objective: Verify that the Flask backend API and the Vue.js frontend are communicating for image processing and prediction.

Expected Outcome: Enabling image upload and receiving predictions through the communication of the frontend and backend.

1. Set Up Axios:
    - Submit HTTP queries to the backend API, use Axios.
2. Define API Endpoint:
    - To transfer the image file to the Flask server for prediction, provide the backend API endpoint in the upload function.
3. Handle API Responses:

- Handle errors correctly or extract and display the predicted plant name based on the response from the API.

Step 5: Implement Map Functionality

Objective: To display a map and identify the user-inputted coordinates, using Leaflet.

Expected Outcome: A map with the user-inputted coordinates indicated with a marker.

1. Install and Import Leaflet:
   - Import Leaflet into the MapView.vue component after installing it with npm (npm install leaflet).
2. Initialize the Map:
   - Create a method that sets up and centers the Leaflet map at the given coordinates.
3. Add Marker to Map:
   - Position a marker on the map at the given coordinates, then bind a popup window to show the name of the location or a pre-defined message.

Step 6: Test the Web Application

Objective: Verify that the web application displays locations on a map, processes image uploads accurately, and makes predictions.

Expected Outcome: Correct map displays, successful image uploads, and accurate predictions.

1. Run the Development Server:
   - In the project directory, execute npm run serve to launch the Vue.js development server.
2. Test Image Upload and Prediction:
   - Launch the web application using a web browser, submit an image, and confirm that the prediction appears as intended.
3. Test Map Functionality:
   - Enter the coordinates and confirm that the map marker is centred on the given location.

4. Expected Outcome

4.1 Literature Review

Characteristics of Firethorn

Pyracantha Angustifolia, frequently referred to as firethorn, is a species of shrub belonging to the rose family. The Greek terms "pyr," which translates to fire, and "akanthos," which means thorn, are the inspiration for the name "pyracantha," which refers to the flaming red berries and jagged thorns that are indicative of plants in this genus. Despite it being mistaken for P. fortuneana or P. koidzumii, firethorn can be easily identified by its distinctive maturity on the adaxial parts of the leaves, receptacle, and calyx (Chari, et al., 2020).

This shrub can reach heights of 4 to 6 feet and widths of 6 to 8 feet. It is a densely branched, evergreen shrub with lance-shaped, glossy, dark green leaves that are thorny (Roelofsen, n.d.). The little, spherical pomes that are produced from the white blossoms can range in hue from orange to red. Birds eat these fruits because they are bitter and acidic, rendering them unappealing to humans. Hydrogen cyanide can be found in the leaves, fruits, and seeds which gives them their bitter flavour. There are sharp spines on the stems and branches (Adams, 2020). Although it originated in China, this species has been brought to Australia, North America, and South Africa as an attractive plant for garden hedges to improve the security of houses.

Distribution and Impact in South Africa

After being brought to South Africa, the firethorn plant has proliferated over all nine provinces. The provinces of Gauteng, the Free State, and the Eastern Cape are particularly abundant in it. Early in the 1980s, the plant started to spread over South Africa's grassland habitats. It is currently common in the neighbouring nations of Lesotho, Eswatini, and Zimbabwe as well as the tropical grasslands of the Eastern Cape, Free State, KwaZulu-Natal, and Mpumalanga Provinces (Roelofsen, n.d.).

Various other plants suffer exclusion because of the plant's dense thicket growth. The firethorn canopy attractions various plants that do not create seedbanks while affecting the microenvironmental conditions needed to promote the growth of specific native plants (Chari, et al., 2020). This density also provides smaller trees, shrubs, and seedlings with a physical defence mechanism against omnivores. However, in southern Africa, it is categorized as an invasive plant because it replaces the natural flora and pastures, contaminates seedbanks, and inhibits access by livestock and humans (Roelofsen, n.d.).

Invasive Nature

Firethorn has been assigned the classification of Category 1B invasive species. According to this categorization, these species need to be managed and, if possible, eradicated. Planting or selling of any kind is forbidden (Chari, et al., 2020). These plants have taken over rivers and lakes, cliff faces, and grasslands in southern Africa. These plants are larger than native grasses and frequently create complex Rosaceae thickets that alter the structure of the vegetation by casting a large shadow upon them. The ecosystems of high elevation grasslands are seriously threatened by this (Roelofsen, n.d.).

South Africa's high-altitude grasslands are important areas for biodiversity with a high level of plant diversity; they must be preserved. Since the grassland biome contributes half of the nation's water runoff, it is also incredibly valuable to the economy (Chari, et al., 2020).

Reproductive Traits and Spread of Firethorn

A thorough overview of Pyracantha Angustifolia's biology can be found in the research study "Biology of Invasive Plants 1. Pyracantha angustifolia (Franch.) C.K. Schneid" that was published in the Invasive Plant Science and Management journal. The properties, distribution, and environmental effects of the plant are covered in this study. Additionally, it emphasizes that firethorn is a relatively cold-tolerant plant, with a limited survival period at temperatures of -18.9 C and below. Nonetheless, the plant flourishes in a variety of environmental situations outside of its natural region (Chari, et al., 2020).

The reproductive characteristics of Firethorn are examined in a different research paper titled "Reproductive Ecology of the Invasive Alien Shrub Pyracantha angustifolia in the Grassland Biome, South Africa." According to the study, all opportunist insects, that were observed visiting flowers, had high purity (>70%) pollen loads in them. Firethorn can germinate seeds (66%) without pollen vectors, according to floral visitor exclusion trials however, the production of fruit was greater (91%), when pollination occurred naturally. Furthermore, the study found that under bushes, there were a high seed density of 46,400 ± (SE) 8934 m^{-2}, which decreased with distance from the shrub (Adams, et al., 2023).
Lehlohonolo investigates the impact of local condition variation in a dry intermontane valley on the spread of firethorn in the report "Reproductive Ecology of Pyracantha Angustifolia in Afromontane Grasslands of the Eastern Free State". According to the paper, the rate of spread would be significantly greater in shrublands (by up to 33%) and emphasizes how pressure from grazing would significantly slow the pace of the spread by up to 53% through cattle directly consuming the seeds. These findings imply that the valley's shrubland areas are the most susceptible to invasion and that grazing might lessen the firethorn's effects (2020).

As it stands, Pyracantha Angustifolia is a significant invasive plant that is widely distributed throughout South Africa, especially in Gauteng. The surrounding environment is greatly impacted by its dense thickets and canopy shadowing, which replace native flora and prevent access for both individuals and cattle.

4.2 Systems Analysis and Design

4.2.1 System Analysis

With the objective to develop a web application that allows image uploads from users, identification of the plant species, and locate the species position on a map, our team utilized cutting edge machine learning and web technologies in a comprehensive and integrated approach. This section provides a detailed description of the procedure and the reasoning behind each approach used along with suggesting possible improvements for future prototypes. A GitHub repository was created to contain the code for the project.

Backend Development

The Flask Python web framework is a lightweight and adaptable tool for developing backend APIs. Flask was chosen because of its ease of use, versatility, and simplicity, all of which promote iterative improvements and quick development. Since Flask is modular, it enables a systematic separation of tasks, which makes codebase management and upkeep easier. The convolutional neural network (CNN) model that the API endpoints use to determine plant species is trained before handling image uploads and processing.

CORS (Cross-Origin Resource Sharing) was implemented to enable effortless interaction between the frontend and backend, which are located on different domains. Any security issues that could arise with the cross-origin requests by activating CORS was resolved allowing for requests from the frontend to be executed by the backend.

Image Processing and Model Prediction

Our team utilized TensorFlow-based Keras, a high-level neural networks API, for image processing and model prediction. To ensure flexibility and simplicity of maintenance, the functions to load the model, preprocess images, and predicted the class were created in a separate module. Keras was selected because of its comprehensive support for rapid prototyping and experimentation in machine learning challenges, its robust interface with TensorFlow, and its user-friendly API.

To ensure consistent input to the model, the image preparation phases included reducing images to a uniform size and normalizing the pixel values. Furthermore, we developed an EXIF data extraction procedure to extract GPS coordinates from the images metadata. This enabled the application to embed location metadata into the images whenever it is available.

Frontend Development

Vue.js is a progressive JavaScript framework aimed at creating user interfaces with ease. It was used in the development of the frontend. Because of its minimal learning curve, component-based design, and reactive data binding, Vue.js was chosen as the framework of choice to develop a dynamic and interactive user interface.

A Leaflet (open-source JavaScript tool for interactive maps) was utilised to display maps. This was used for its affordability, flexibility, and user-friendliness instead of other commercial mapping programs like Google Maps. Axios was used to manage asynchronous processes in a straightforward and effective manner, facilitating smooth communication between the frontend and backend.

Considerations and Alternatives

Although Flask was picked, alternative backend frameworks such as Django or Node.js were discussed but eventually excluded. Django is another Python-based framework with built-in support for admin interfaces, ORM, and user authentication. It also packed with features and has a strong environment. But Django's complexity and size made it less appropriate for our prototype, where iteration and quick development were of utmost importance. Node.js non-blocking I/O operations excels in managing real-time, event-driven applications. Despite this, Flask offered a smoother interface with our current Python-based machine learning architecture because our application needed to integrate extensively with machine learning models that were written in Python.

For image processing, an alternative could have been OpenCV. With its broad image modification and processing capabilities, it is a potent toolkit for machine learning tasks. And yet because of its high-level abstraction, user-friendliness, and direct support for deep learning model building, Keras was chosen instead as it more closely matches the objectives of employing pre-trained neural network models for image categorization.

React and Angular were taken into consideration as frontend frameworks, however they were not chosen. React has a broad environment and effective renderings using a virtual DOM. However, React's learning curve and the additional setup requires for managing states with tools like Redux made it not ideal for the project. ANother possibility was Angular which is a feature-rich framework with a strong CLI and dependency injection. However, the complexity and steep learning curve of Angular make Vue.js the superior choice for rapidly developing an interactive and responsive frontend with no setup.

Future Improvements

In future versions, the application can potentially be improved in several areas. By utilizing transfer learning from pre-trained models or experimenting with deeper architectures, the existing model can be further customized for speed and accuracy. The user experience would be greatly enhanced by improving user feedback and error management in both the frontend and backend.

Deploying the backend on a scalable cloud service and incorporating a database to store user uploads and predictions would be advantageous to increase performance and data management. the collection of GPS coordinates from image metadata can be automated for autonomous geotagging. This will result in a decrease in the amount of human input which eliminates any errors that can be made. Furthermore, Google maps API can be integrated into the image processing model to extract the location and the image of the selected plant species directly from the google maps which can then be uploaded, and a prediction can be made.

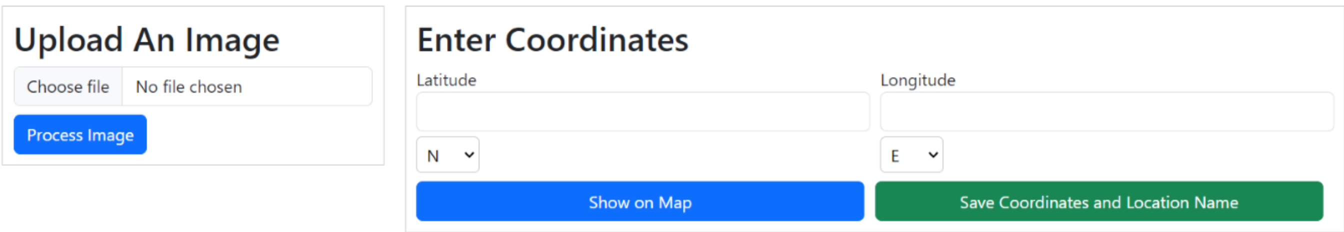
Ultimately, the user experience would be greatly enhanced by improving the frontend design to increase user engagement and functionality, such as offering a preview of the map location prior to submission.

The technologies and techniques utilized made it possible to develop a functional prototype that highlights the main functions of the application. For future research and development, there is a great deal of potential for improvement and additional refining to provide a more reliable and user-friendly web application program.

4.2.2 Web Application Prototype User Manual

Presented in this section is a visual guide to the frontend web application prototype. This user manual provides a walkthrough of the application's interface, demonstrates the interaction with its features, and offers a preview of the expected output displays.

Home Page Display
When the application is run, a home page containing an Upload Image and Coordinates section will be displayed for the user. This should look as follows:

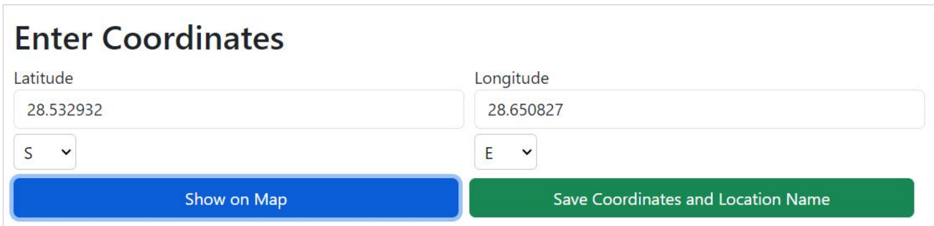


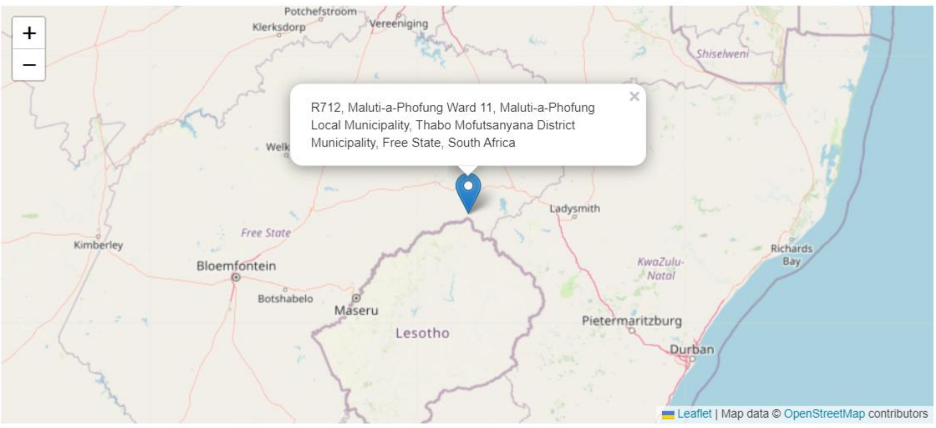
Upload Image Area

- The first task the user will be asked to do is upload an image they have taken by selecting the "Choose file" section.
- The image uploaded will be displayed below to show the user which image is going to be uploaded for processing.
- The "Process Image" button will process the image and load it into the integrated machine learning model so that a prediction can be made on what type of species it is.
- This prediction will be displayed in a box below the image.
  - A green box and the text "Firethorn" indicate the image uploaded was a Firethorn.
  - A Red box indicated the image is a different species.

Map Area

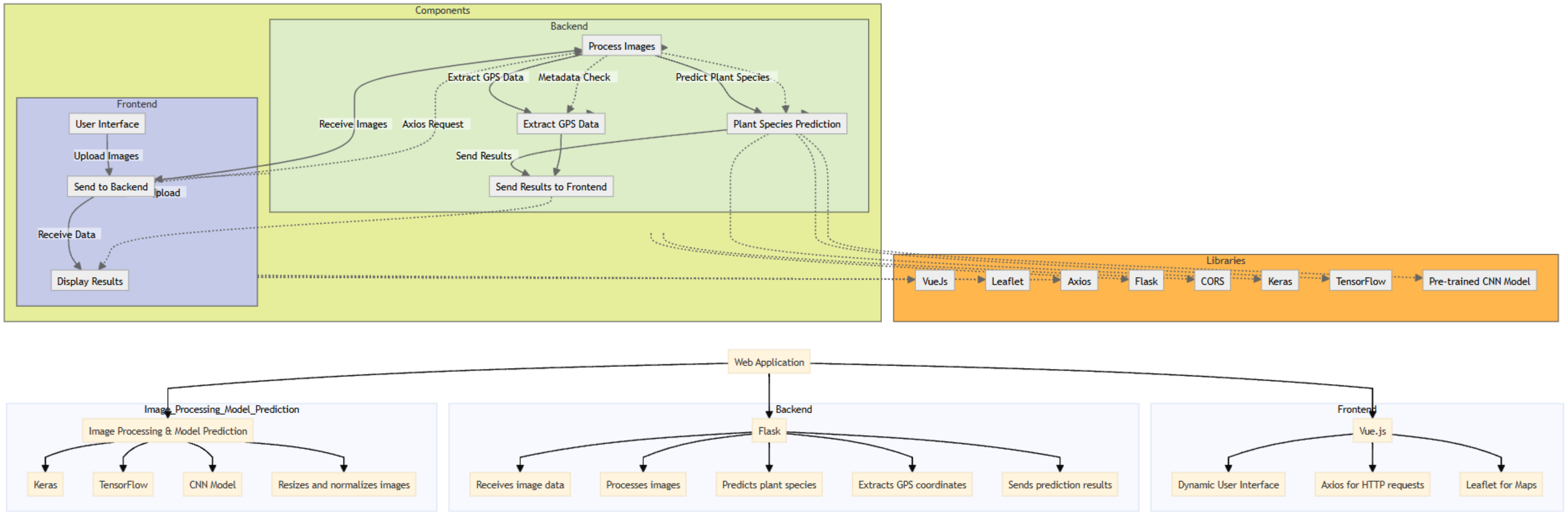
- Once the model has made a prediction, the user can enter the coordinates of where the image was taken into the provided fields.
- Additionally, they can specify the location using the dropdown boxes providing the option of:
  - **N** – North
  - **S** – South
  - **E** – East
  - **W** – West
- When the user selects the "Show on Map" button, a marker will display on a made showing the exact location of the firethorn.
- When the user selects the "Save Coordinates and Location Name" button, the image, coordinates, and prediction are saved to a file.





APPENDIX

Web application prototype component structure and data flow



GitHub Repository link: https://github.com/AliYalcinPXL/International-Research-Project.git

BIBLIOGRAPHY

Adams, L., 2020. *REPRODUCTIVE ECOLOGY OF PYRACANTHA ANGUSTIFOLIA IN AFROMONTANE GRASSLANDS OF THE EASTERN FREE STATE*. [Online]
Available at: https://scholar.ufs.ac.za/server/api/core/bitstreams/797ea0bf-54e3-414e-892d-08101252844e/content
[Accessed 27 May 2024].

Adams, L. et al., 2023. *Reproductive Ecology of the Invasive Alien Shrub Pyracantha angustifolia in the Grassland Biome, South Africa*. [Online]
Available at: https://www.mdpi.com/2223-7747/12/6/1308
[Accessed 22 May 2024].

Canavan, K. et al., 2022. *Invasive Alien Plants in the Montane Areas of South Africa: Impacts and Management Options*. [Online]
Available at: https://link.springer.com/chapter/10.1007/978-3-031-01980-7_14
[Accessed 25 May 2024].

Chari, L. et al., 2020. *Biology of Invasive Plants 1. Pyracantha angustifolia (Franch.) C.K. Schneid*. [Online]
Available at: https://bioone.org/journals/invasive-plant-science-and-management/volume-13/issue-3/inp.2020.24/Biology-of-Invasive-Plants-1-Pyracantha-angustifolia-Franch-CK-Schneid/10.1017/inp.2020.24.full
[Accessed 25 May 2024].

Gardenia, 2024. *https://www.gardenia.net/plant/pyracantha-angustifolia-yukon-belle*. [Online]
Available at: https://www.gardenia.net/plant/pyracantha-angustifolia-yukon-belle
[Accessed 08 March 2024].

Rebelo, T. & Gilmore, J., 2024. *Narrowleaf Firethorn Pyracantha angustifolia*. [Online]
Available at: https://www.inaturalist.org/taxa/54053-Pyracantha-angustifolia
[Accessed 08 March 2024].

Roelofsen, D., n.d. *Firethorns and Cotoneasters are Category 1B invasive species*. [Online]
Available at: https://www.gardeninginsouthafrica.co.za/firethorns-and-cotoneasters-are-category-1b-invasive-species
[Accessed 27 May 2024].

Wikipedia, 2024. *Pyracantha angustifolia*. [Online]
Available at: https://en.wikipedia.org/wiki/Pyracantha_angustifolia
[Accessed 08 March 2024].