KLE Technological University
Hubballi

A Project Report on

"Leaf Of Faith"

A Project Report Submitted in Partial Fulfillment of the Requirement for the Course of

Minor Project-1 (23ECSW303)

in

6th Semester of Computer Science and Engineering

by

Chandan Satwani	02FE21BCS021
Vrashabh Patil	02FE21BCS110
Jay Sancheti	02FE21BCS035

Under the guidance of

Dr. Santosh Pattar

Assistant Professor,

Department of Computer Science and Engineering,

KLE Technological University's Dr. MSSCET, Belagavi.

KLE Technological University's

Dr. M. S. Sheshgiri College of Engineering and Technology,

Belagavi – 590 008.

June 2024

DECLARATION

We hereby declare that the matter embodied in this report entitled "**Leaf Of Faith**" submitted to KLE Technological University for the course completion of Minor Project-1 (23ECSW303) in the 6th Semester of Computer Science and Engineering is the result of the work done by us in the Department of Computer Science and Engineering, KLE Technological University's Dr. M. S. Sheshgiri College of Engineering, Belagavi under the guidance of Dr. Santosh Pattar, Assistant Professor, Department of Computer Science and Engineering. We further declare that to the best of our knowledge and belief, the work reported here in doesn't form part of any other project on the basis of which a course or award was conferred on an earlier occasion on this by any other student(s), also the results of the work are not submitted for the award of any course, degree or diploma within this or in any other University or Institute. We hereby also confirm that all of the experimental work in this report has been done by us.

Belagavi – 590 008
Date :

Chandan Satwani
(02FE21BCS021)

Vrashabh Patil
(02FE21BCS110)

Jay Sancheti
(02FE21BCS035)

CERTIFICATE

This is to certify that the project entitled "Leaf Of Faith" submitted to KLE Technological University's Dr. MSSCET, Belagavi for the partial fulfillment of the requirement for the course – Minor Project-1 (23ECSW303) by Chandan Satwani (02FE21BCS021), Vrashabh Patil (02FE21BCS110), Jay Sancheti (02FE21BCS035), students in the Department of Computer Science and Engineering, KLE Technological University's Dr. MSSCET, Belagavi, is a bonafide record of the work carried out by them under my supervision. The contents of this report, in full or in parts, have not been submitted to any other Institute or University for the award of any other course completion.

Belagavi – 590 008
Date :

Dr. Santosh Pattar
(Project Guide)

Prof. Anirudha Potdar
(Course Coordinator)

Dr. Rajashri Khanai
(Head of the Department)

Abstract

Our planet's incredible plant life is under attack. Widespread deforestation disrupts delicate ecosystems, putting countless species at risk. Traditional conservation methods struggle to monitor these vast areas efficiently.

But there's hope! We're developing a powerful new weapon: a centralized platform designed specifically for forest monitoring. This platform transcends limitations on specific data sources. Instead, it leverages cutting-edge technologies like Semantic Web concepts and SPARQL queries. Imagine a central hub that empowers conservation groups with these tools. They can identify areas most vulnerable to destruction and track changes within ecosystems over time.

This data-driven approach, facilitated by a REST API for seamless communication, has the potential to revolutionize how we protect our planet's precious plant biodiversity. By equipping conservationists with the right tools, we can significantly improve their efforts and ensure a thriving future for plant life on Earth.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The unrelenting loss of forests and the degradation of animal habitats pose significant threats to the world's rich biodiversity. To combat this crisis, innovative solutions are needed to empower wildlife conservation efforts. This report explores the potential of developing a Java backend API specifically designed for forest monitoring. This API would leverage the freely available data from Sentinel-2 satellites for land cover analysis and integrate animal movement data from the Animal Movement Bank.

By creating a centralized platform for wildlife organizations and NGOs to access critical data and collaborate seamlessly, this project aspires to contribute significantly to achieving the United Nations Sustainable Development Goal (SDG) 15: Life on Land. This report will delve into the technical, functional, economic, and scheduling feasibility of the project, identifying potential risks and outlining recommendations for successful development.

FIGURE 1.1: The devastating impact of deforestation on animal habitats.

The unrelenting loss of forests and the degradation of animal habitats pose significant threats to the world's rich biodiversity. Habitat destruction fragments ecosystems, disrupts food chains, and displaces wildlife populations. This alarming decline in biodiversity has a domino effect, impacting everything from climate regulation to human health. Scientists estimate that we are currently experiencing a mass extinction event, with species disappearing at an unprecedented rate.

To combat this crisis, innovative solutions are needed to empower wildlife conservation efforts. Traditional conservation methods, while crucial, often face limitations in terms of data collection, analysis, and collaboration. The proposed Java backend API for forest monitoring has the potential to revolutionize how conservationists approach habitat protection. By leveraging freely available satellite data and animal movement information, the API can provide real-time insights into deforestation patterns and animal migration routes. This centralized platform would equip conservation groups with the tools they need to make informed decisions, collaborate effectively, and ultimately, safeguard the future of our planet's incredible biodiversity.

1.1 Background

For millennia, humans have interacted with and impacted natural habitats. Early civilizations often relied on slash-and-burn agriculture, leading to localized deforestation. However, the Industrial Revolution in the 18th century marked a turning point. The surge in demand for resources like timber and land for large-scale agriculture fueled a rapid acceleration of habitat destruction. Historical records, for instance, reveal how the once vast Mesopotamian forests, crucial for the rise of early civilizations, were significantly depleted by 2000 BCE due to unsustainable agricultural practices.

Today, the rate of habitat loss is unprecedented. The World Wildlife Fund (WWF) reports that between 1970 and 2018, vertebrate populations have declined by an alarming 68 percent on average, primarily due to habitat loss [Source: WWF Living Planet Report 2020]. Forests, once covering over half of the Earth's land surface, have dwindled to just 30 percent according to the UN Food and Agriculture Organization (FAO) [Source: FAO Global Forest Resources Assessment 2020]. These statistics paint a grim picture, highlighting the urgency of addressing this escalating crisis.

The consequences of habitat loss are far-reaching and profoundly impact the health of our planet. Forests play a crucial role in regulating the climate by absorbing carbon dioxide, a major greenhouse gas. With deforestation, this vital function is compromised, accelerating climate change. Habitat loss also disrupts natural water cycles, leading to increased soil erosion and flooding. Ultimately, the loss of biodiversity weakens the entire web of life, jeopardizing food security, clean air and water, and the essential services that natural ecosystems provide.

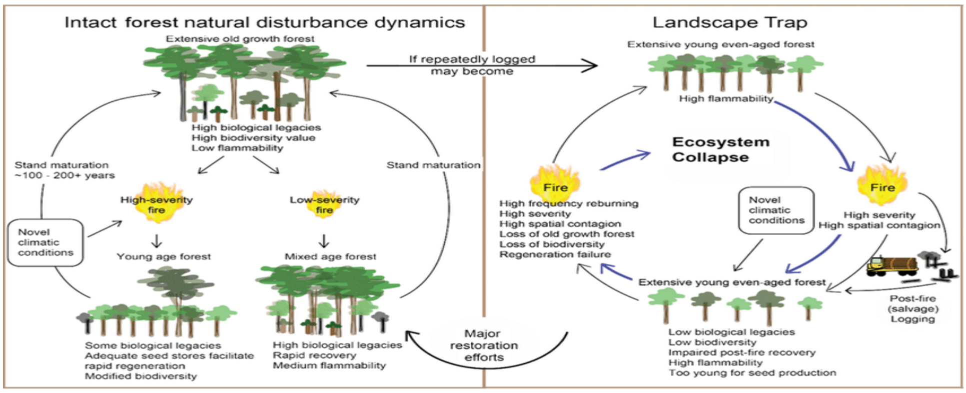

FIGURE 1.2: From vast forests to fragmented remnants: A sobering look at humanity's impact on habitats.

1.2 Problem Statement

Developing a user-friendly platform is crucial to bridge the gap between available environmental data and the needs of wildlife conservation groups for effective decision-making.

1.2.1 Objectives

The objectives of the problem statement is as follows:

- Develop a Java backend API for forest monitoring:
  - A well-functioning Java backend API will be delivered. This API will efficiently handle forest monitoring data and ensure scala- bility.
- Integrate functionalities to access and incorporate animal movement data:
  - By successfully integrating animal movement data, the platform provides a more holistic view of forest health. This allows wildlife groups and NGOs to not only monitor deforestation but also understand how changes in forest cover impact animal behavior patterns. This comprehensive data contributes to achiev- ing targets within SDG 15.
- Design a platform to connect wildlife groups and NGOs:
  - A successful platform will provide wildlife organizations with a central hub for accessing forest monitoring data, collaborating with other groups, and planning conservation actions. This fosters a more coordinated ap- proach towards achieving SDG 15 targets.

Chapter 2

Literature Survey

The literature survey is one of the important step carried out in the process of project development as it provides the understanding of the project domain and also provide us insights of the existing solutions for the problem statements. The following paragraphs

A literature survey on immediate reporting for rare flora conservation reveals its potential for early threat detection, enabling faster intervention and resource mobilization. However, challenges like public awareness gaps and inefficient reporting systems persist. Stakeholder interviews (e.g., conservation biologist) can offer valuable insights, potentially confirming existing research on these challenges while also suggesting targeted solutions like improved public awareness campaigns and user-friendly reporting apps. By combining research findings with stakeholder perspectives, we can create a more comprehensive understanding of immediate reporting's role in protecting rare plant life

Chapter 3

System Modeling

System design is process of defining the elements of system such as components, architecture, interface that matches with the requirements. The system design involves a process called system modeling, that is a process of deriving abstract model of a system, where each model presenting different perspective or view.

Basically, there are 5 types of system thinking views: reinforced loops, Balanced loops,Behaviour Overtime,Collect and Cluster and Cazual mapping . Reinforcing Loops depict how positive feedback cycles within a system can cause changes to accelerate over time. Balancing Loops capture negative feedback cycles that counteract change and bring a system back to equilibrium.Behavior Over Time focuses on observing how a system's behavior changes and evolves in response to internal or external influences.Collect and Cluster involves gathering data on various factors affecting the system and then grouping them based on their similarities or how they influence each other.Causal Mapping visually depicts cause-and-effect relationships between different factors within a system.

Therefore, in system modeling we have 5 types of system thinking views:

- Reinforced loops
- Balancing loops
- Behaviour Overtime
- Collect and Cluster
- Casual Mapping

Let us describe our system with respect to these system tinking views taking scenarios to understand, in the following sections.

3.1 Reinforcing loops

Reinforcing loops, like a fire spreading, depict positive feedback cycles in a system. As one factor increases, it triggers changes that further amplify that initial increase, creating a self-perpetuating chain reaction.Here in our project the reinforced can be drawn as follows.

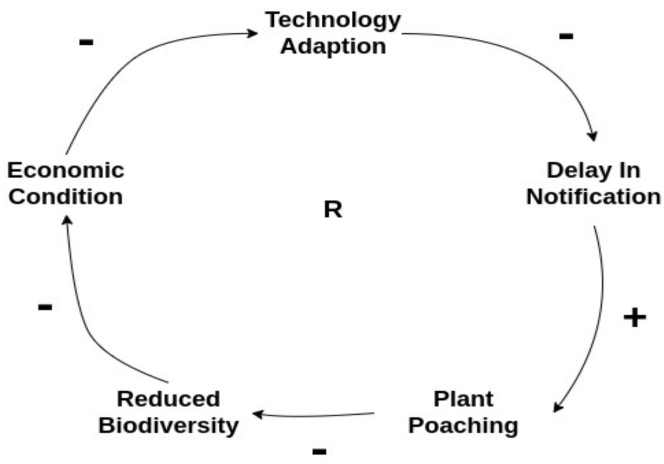

FIGURE 3.1: reinforced graph

The above fig Figure 3.1 is a reinforced loops shows the issues and how they are causing a chain of events. We can see that the economic condition is weaker it will create opstacle for technology adaption which in term will delay in notifing the concerned authority which will give rise to the plant poaching swhich will cause loss in bio diversity and this loop continues.

3.2 Balancing Loops

Balancing loops, like a thermostat, act as the system's self-correcting mechanisms. When a factor strays from its ideal state, these loops trigger counterbalancing effects that push the system back towards equilibrium, maintaining stability.The following fig 3.2 shows the balancing loop.

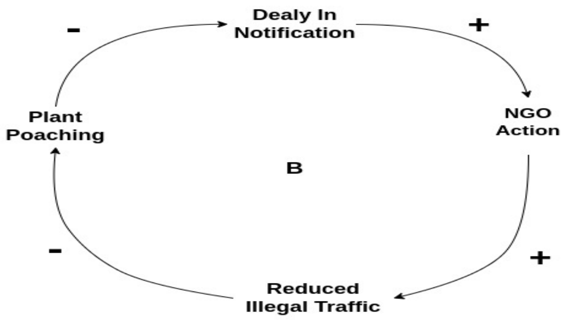

Figure 3.2: Balancing Loop

In the Balancing loop given we see that when the delay in notification is minimised the NGO's will take a faster action which will reduce the illegal trafficing the rare flora which will reduce the plant poaching and the loop continous.

3.3 Behaviour Overtime

Behavior Over Time tracks a system's evolving responses. Like watching a seed sprout, it observes how the system's actions and outputs change in response to internal and external influences over extended periods, helping predict future behavior and identify potential tipping points.

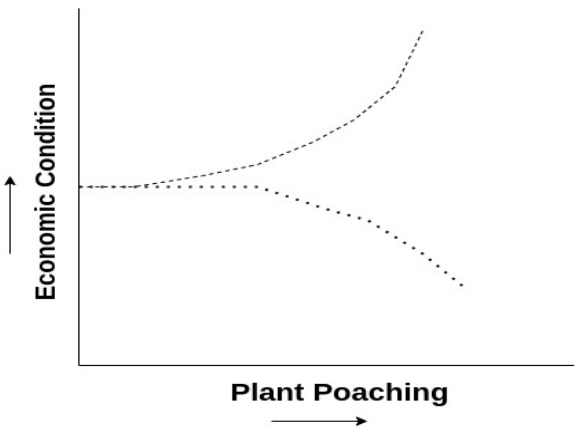

Figure 3.3: Sequence diagram of the system

In the given behaviour overtime graph we see how an factor affects the another factor in the given graph we see that as the economic condition increase the plant poaching reduce and the opposite case is also true when the economic condition reduce the plant poaching increase.

3.4 Collect and Cluster

Collect and Cluster, in system thinking, involves gathering data on various factors impacting a system. These factors are then grouped based on their similarities or how they influence each other. It's like sorting Legos – red pieces go together because they share a trait, helping us understand the system's interconnectedness.

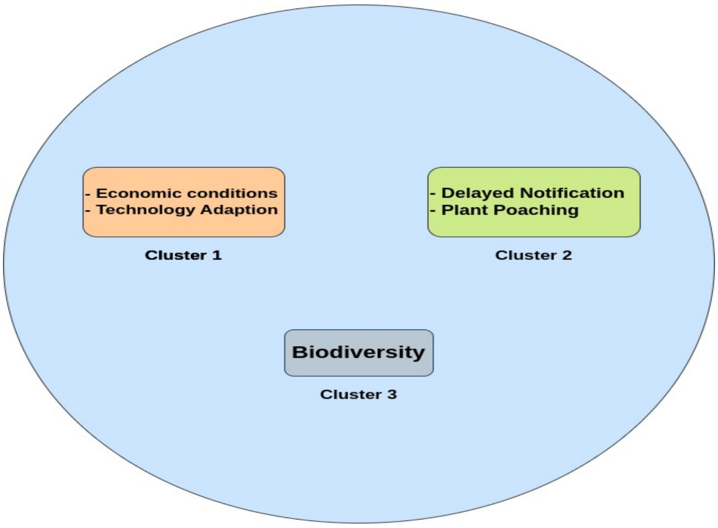

FIGURE 3.4: Collect and Cluster graph

In the graph we see that we try club the issue in certain cluster depending on how they are related to each other we see in the given diagram there are three cluster first cluster comprises of economic condition and technology adaption in the second cluster we see Dealyed Notification and Plant Poaching and in the third cluster we there is biodiversity.

3.5 Casual Mapping

Causal mapping visually untangles a system's complexity. It acts like a mind map, where factors affecting the system are branches. Arrows connect them, showing cause-and-effect relationships. By analyzing these connections, we gain a clear picture of how different elements influence each other, helping us predict system behavior and potential outcomes.

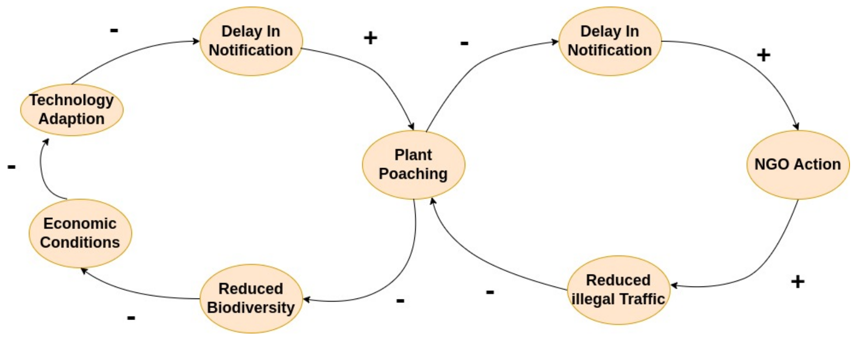

FIGURE 3.5: Casual Mapping

The casual mapp in the given graph we see that the balanced and the reinforced loops are combined together where we take a common factor between the two loop where it was the plant poaching and the graph was created which show the system with high complexity combined together.

Chapter 4

Implementation Details

The implementationpart consists upon building up an website which leverages a two-pronged approach one is front-end and a semantic web back-end. React.js provides a robust framework for building the user interface. We've structured the application using a component-based approach, ensuring modularity and reusability. State management is handled effectively, enabling dynamic updates based on user interactions. The UI itself is designed with responsiveness in mind, ensuring a seamless experience across various devices. User input, such as text fields and button clicks, triggers updates within the React components and seamlessly communicates with the back-end.

For the back-end, we delve into the realm of semantic web concepts. Our Data base is meticulously structured using the N3 format, providing a rich representation of entities and their relationships. This allows for flexible data modeling and efficient querying using SPARQL, a powerful query language, empowers us to retrieve relevant information from the knowledge base in response to user queries. We've carefully crafted SPARQL queries to match specific user needs, ensuring the retrieval of accurate data.

Finally,A well-defined REST API acts as the bridge between React's front-end and the semantic web back-end. Designed for clarity and efficiency, it uses appropriate data formats. Rigorous testing ensures its functionality and performance. The front-end seamlessly sends queries, receives insightful responses from the back-end, and displays them to the user, creating a smooth experience.

Chapter 5

Testing

The below is the snpashot of test cases table.

Test Case ID	Test Level	Test Type	Test Case Description	Input	Expected Output	API Endpoint
TC_001	Unit	Functional	Test `readRdfFile` method with valid file path	Valid file path	Successful file read and model creation	N/A
TC_002	Unit	Functional	Test `readRdfFile` method with invalid file path	Invalid file path	Exception thrown	N/A
TC_003	Unit	Functional	Test `executeRdfQuery` method	RDF file with valid data	Successful query execution and result retrieval	N/A
TC_004	Component	Functional	Test `MapComponent` rendering with valid coordinates	Valid coordinates	Map rendered with markers at correct coordinates	/getCoordinates
TC_005	Component	Functional	Test `MapComponent` rendering with invalid coordinates	Invalid coordinates	Error handling and no markers rendered	/getCoordinates
TC_006	System	Functional	Test complete flow from RDF file	RDF file with valid data	Successful map rendering	/getCoordinates

Chapter 6

Results and Outcomes

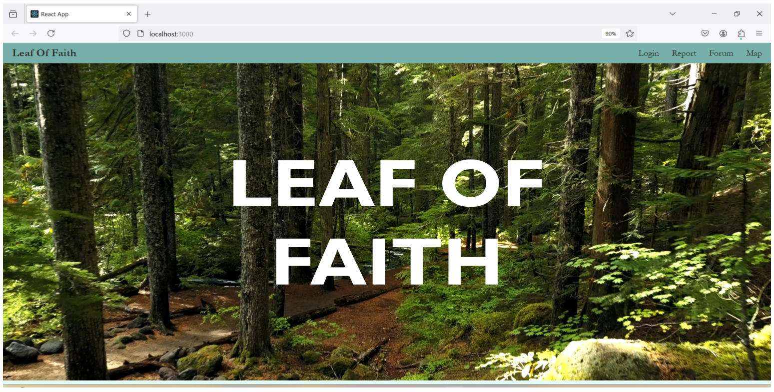

FIGURE 6.1: Home page.

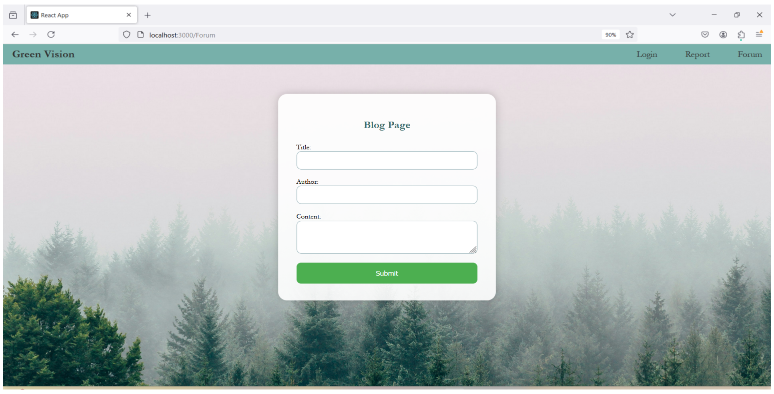

FIGURE 6.2: Blog adding page.

 Chapter 6 Results and Outcomes

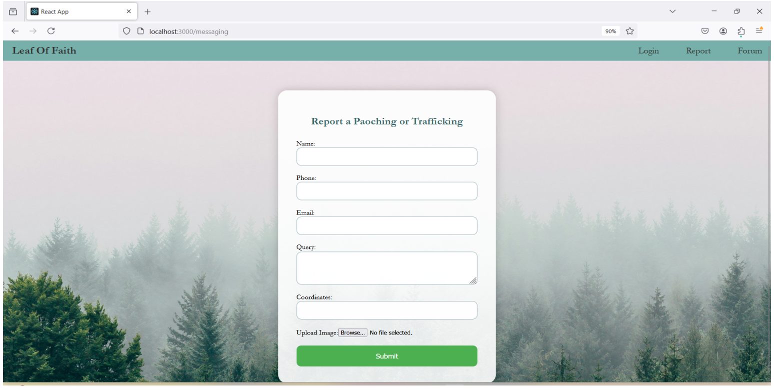

Figure 6.3: Reporting page.

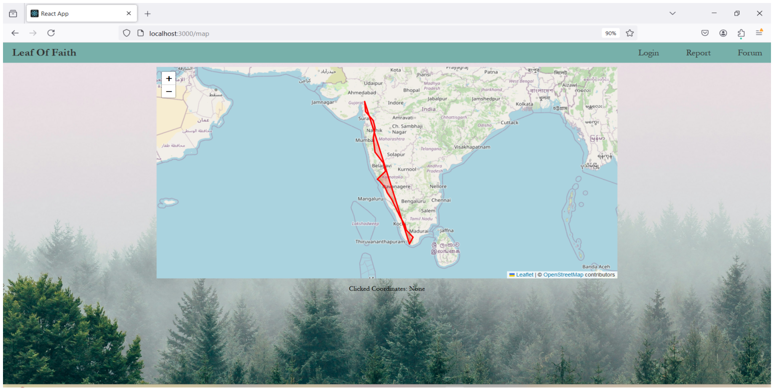

Figure 6.4: Forest tracking page.

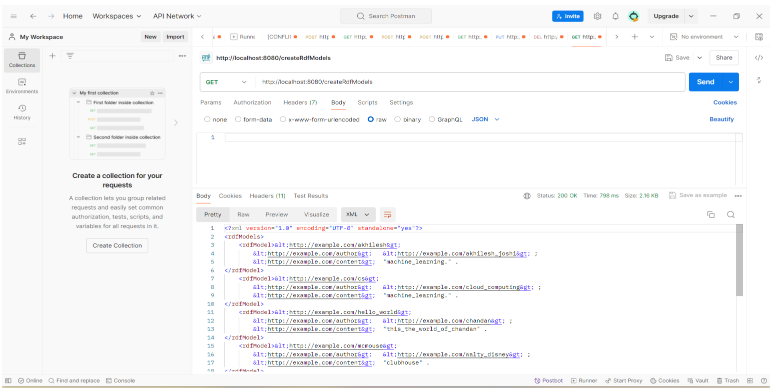

Figure 6.5: Sample outcome for SPARQL query in REST api.

Conclusions

The problem statement for our project was developing a user-friendly platform is crucial to bridge the gap between avail- able environmental data and the needs of wildlife conservation groups for effective decision-making this project addresses this issue by creating a user-friendly website. The website itself is built using React.js, a powerful framework that allows for a clear and responsive interface. This means the website looks great and works seamlessly on any device, from desktops to phones. Users can easily interact with the website, and any changes they make update in real-time, providing a smooth user experience.

Behind the scenes, the website utilizes a powerful technology called the semantic web. This allows us to organize the environmental data in a flexible format, making it easier to search and retrieve specific information. We then use a special language called SPARQL to craft precise queries that match the user's needs. This ensures that users get the most relevant and accurate data possible.

The final piece of the puzzle is a well-designed API, which acts as a bridge between the user-friendly front-end and the powerful back-end. This API allows for smooth communication and efficient data delivery. Users can submit queries through the website, and the API seamlessly retrieves the relevant information from the back-end. The results are then displayed clearly on the website, empowering conservation groups to make informed decisions based on the data they need.