

Project Report

Date	17 November 2022
Team ID	PNT2022TMID18534
Project Name	Project - Digital Naturalist - AI Enabled tool forBiodiversity Researchers
Content	The Project Report has a detailed description of the documentation done for the intended project.

1. **INTRODUCTION**
 1. Project Overview
 2. Purpose
2. **LITERATURE SURVEY**
 1. Existing problem
 2. References
 3. Problem Statement Definition
3. **IDEATION & PROPOSED SOLUTION**
 1. Empathy Map Canvas
 2. Ideation & Brainstorming
 3. Proposed Solution
 4. Problem Solution fit
4. **REQUIREMENT ANALYSIS**
 1. Functional requirement
 2. Non-Functional requirements
5. **PROJECT DESIGN**
 1. Data Flow Diagrams
 2. Solution & Technical Architecture
 3. User Stories
6. **PROJECT PLANNING & SCHEDULING**
 1. Sprint Planning & Estimation
 2. Sprint Delivery Schedule
7. **CODING & SOLUTIONING (Explain the features added in the project along with code)**
 1. Feature 1
 2. Feature 2
 3. Database Schema (if Applicable)
8. **TESTING**
 1. Test Cases
 2. User Acceptance Testing
9. **RESULTS**
 1. Performance Metrics
10. **ADVANTAGES & DISADVANTAGES**
11. **CONCLUSION**
12. **FUTURE SCOPE**
13. **APPENDIX**

Source Code

GitHub & Project Demo Link

1.INTRODUCTION

1.1 Project Overview

- ▶ Biodiversity is the variation among living organisms from different sources including terrestrial, marine and desert ecosystems, and the ecological complexes of which they are a part.
- ▶ Biodiversity includes the number of different organisms and their relative frequencies in an ecosystem. It also reflects the organization of organisms at different levels.
- ▶ The term biodiversity was coined in 1985. It is important in natural as well as artificial ecosystems. It deals with nature's variety, the biosphere. It refers to variabilities among plants, animals and microorganism species.
- ▶ Biodiversity holds ecological and economic significance. It provides us with nourishment, housing, fuel, clothing and several other resources. It also extracts monetary benefits through tourism. Therefore, it is very important to have a good knowledge of biodiversity for a sustainable livelihood.

Importance of Biodiversity

- ▶ Biodiversity and its maintenance are very important for sustaining life on earth. A few of the reasons explaining the importance of biodiversity are:

Ecological Stability

- ▶ Every species has a specific role in an ecosystem. They capture and store energy and also produce and decompose organic matter. The ecosystem supports the services without which humans cannot survive. A diverse ecosystem is more productive and can withstand environmental stress.

Economic Importance

- ▶ Biodiversity is a reservoir of resources for the manufacture of food, cosmetic products and pharmaceuticals.
- ▶ Crops livestock, fishery, and forests are a rich sources of food.
- ▶ Wild plants such as Cinchona and Foxglove plant are used for medicinal purposes.
- ▶ The national parks and sanctuaries are a source of tourism.

- ▶ Humans should not cause their voluntary extinction. Biodiversity preserves different cultures and spiritual heritage. Therefore, it is very important to conserve biodiversity.
- ▶ A naturalist is someone who studies the patterns of nature, identifies a different kind of flora and fauna in nature. Being able to identify the flora and fauna around us often leads to an interest in protecting wild spaces, and collecting and sharing information about the species we see on our travels is very useful for conservation groups like NCC.
- ▶ When venturing into the woods, field naturalists usually rely on common approaches like always carrying a guidebook around everywhere or seeking help from experienced ornithologists. There should be a handy tool for them to capture, identify and share the beauty to the outside world.
- ▶ Field naturalists can only use this web app from anywhere to identify the birds, flowers, mammals and other species they see on their hikes, canoe trips and other excursions.
- ▶ In this project, we are creating a web application which uses a deep learning model(Convolutional Neural Network Model), trained on a dataset which contains different species of birds, flowers and mammals (2 subclasses in each for a quick understanding) and get the prediction of the name of the bird/flower/mammal when an image is been given.

1.2 Purpose

The project aims to create an application for the hikers to find the rare species of birds, flowers , and mammals by giving a picture taken by them.

Not only finding the name of the species , but also displaying all the possible description or the information about in every possible aspect.

Species are classified into one of nine Red List Categories: Extinct, Extinct in the Wild, Critically Endangered, Endangered, Vulnerable, Near Threatened, Least Concern, Data Deficient and Not Evaluated. Vulnerable, Endangered and Critically Endangered species are considered to be threatened with extinction.

These categories can be displayed to the user so that the user can learn the importance of the preserving the species in the realm of biodiversity chain.

Whether the species is dangerous or not, can also be predicted or displayed to the user.

If they are dangerous , the naturalist can probably stay away from them and if there are any poisonous flowers (i.e) plants , they can also take down a note of them and be alert.

If there are any medical significances of the plants , and if they contribute to the cure of any disease in the world, the information provided to the naturalists can be helpful.

The unique characteristics of any species can also be noted down by the naturalist while displaying the information about them.

2. LITERATURE SURVEY

2.1 Existing Problem

For any given research question, ecologists and data scientists should carefully consider the steps that might be required to ensure the relevance and accuracy of AI-generated data for any given research question. To aid this we have summarized our experience into an eight-point list of questions which we recommend researchers ask themselves when using AI classifier naturalists:

1. Does the spatial distribution of images fit your needs? Images from social media are often aggregated in areas of high population density or tourist hotspots. If the distribution is biased in some way, could this be accounted for in subsequent analyses?
2. Can you filter images before classification? For example, filtering can be done by carefully selecting your source of images, using GPS location, focusing on keywords in image metadata, or using high-level AI classifiers to remove non-target images.
3. What is the appropriate taxonomic resolution for your study? This will be driven by your research question, as well as an assessment of the AI naturalist's accuracy. Classifiers will tend to be more accurate at higher taxonomic levels, but this may vary between taxonomic groups.
4. What reporting biases exist in your dataset? For example, to what degree are charismatic species over-represented, or nocturnal species under-represented? Can you filter the data, or model the results to account for these biases if they are relevant?
5. Do reporting biases change over space or time? We observed significant differences in reporting bias between urban and rural settings, and we anticipate that temporal biases are likely to exist where public interest in elements of the natural environment change over time.
6. How will you propagate uncertainty in classifications? AI classifications are associated with a classification score which is indicative of the uncertainty in the identification. This can be used both as a threshold for removing erroneous results, and/or could be included in models to account for variation in uncertainty between observations.

7. Is the dataset used to train your AI naturalist a good match to the images being classified? A poor match between training and prediction datasets will result in higher error rates, which may not always be associated with low classification scores.
8. Have you adequately documented your dataset? To ensure reproducibility and interoperability ensure that you document the model used for classification, filtering steps used to collate images, and other metadata useful to future researchers, and which may be specified in data standards for AI-generated biodiversity which do not exist at the time of writing.

2.2 References

1. Bonnet P., Goëau H., Hang S.T., Lasseck M., Šulc M., Malécot V., Jauzein P., Melet J.-C., You C., Joly A. Plant identification: experts vs. machines in the era of deep learning. In: Joly A., Vrochidis S., Karatzas K., Karppinen A., Bonnet P., editors. *Multimedia Tools and Applications for Environmental & Biodiversity Informatics Multimedia Systems and Applications*. Springer; 2018. pp. 131–149. [[Google Scholar](#)]
2. Norouzzadeh M.S., Nguyen A., Kosmala M., Swanson A., Palmer M.S., Packer C., Clune J. Automatically identifying, counting, and describing wild animals in camera-trap images with deep learning. *Proc. Natl. Acad. Sci. U S A*. 2018;115:E5716–E5725. [[PMC free article](#)] [[PubMed](#)] [[Google Scholar](#)]
3. Di Minin E., Fink C., Tenkanen H., Hiippala T. Machine learning for tracking illegal wildlife trade on social media. *Nat. Ecol. Evol.* 2018;2:406–407. [[PubMed](#)] [[Google Scholar](#)]
4. Jarić I., Correia R.A., Brook B.W., Buettel J.C., Courchamp F., Di Minin E., Firth J.A., Gaston K.J., Jepson P., Kalinkat G. iEcology: harnessing large online resources to generate ecological insights. *Trends Ecol. Evol.* 2020;35:630–639.. [[PubMed](#)] [[Google Scholar](#)]
5. Carranza-Rojas J., Mata-Montero E., Goeau H. 2018 *IEEE International Work Conference on Bioinspired Intelligence (IWOB)* IEEE; 2018. Hidden biases in automated image-based plant identification; pp. 1–9. [[Google Scholar](#)]
6. Lintott C.J., Schawinski K., Slosar A., Land K., Bamford S., Thomas D., Raddick M.J., Nichol R.C., Szalay A., Andreescu D. Galaxy zoo: morphologies derived from visual inspection of galaxies from the Sloan digital sky survey. *Mon. Not. R. Astron. Soc.* 2008;389:1179–1189. [[Google Scholar](#)]
7. Nguyen K., Fookes C., Jillela R., Sridharan S., Ross A. Long range iris recognition: a survey. *Pattern Recognit.* 2017;72:123–143. [[Google Scholar](#)]
8. Zhang Y., Li X., Gao L., Li P. A new subset based deep feature learning method for intelligent fault diagnosis of bearing. *Expert Syst. Appl.* 2018;110:125–142. [[Google Scholar](#)]

9. Rattani A., Derakhshani R. A survey of mobile face biometrics. *Comput. Electr. Eng.* 2018;72:39–52. [[Google Scholar](#)]

10. Wang A. The Shazam music recognition service. *Commun. ACM.* 2006;49:44–48. [[Google Scholar](#)]

2.3 Problem Statement Definition

Problem Statement :

- Biodiversity is the variation among living organisms from different sources including terrestrial, marine and desert ecosystems, and the ecological complexes of which they are a part.
- Biodiversity is the life support system.
- Biodiversity is the variety of different forms of life on earth, including the different plants , animals , micro–organism, the genes they contain & the ecosystem they form.
- Humans should not cause their voluntary extinction. Biodiversity preserves different cultures and spiritual heritage. Therefore, it is very important to conserve biodiversity
- There are millions of species of plant & animals ; every single variety could be valuable oneday or the other.
- Human beings cannot exist without the abundance and diversity of these biologicalresources.
- In this project, we are creating a web application which uses a deep learning model(Convolutional Neural Network Model), trained on a dataset which contains different species of birds, flowers and mammals (2 subclasses in each for a quick understanding) and get the prediction of the name of the bird/flower/mammal when an image is been given.
- 3 Feature Modules in web application: welcome page, upload page and description page
- A dataset is given . Firstly the dataset is downloaded
- Then the dataset is augmented and then a folder for augmented dataset is created through Data Augmentation
- The Data is split into train and test images and then the CNN model is built by data pre-processing and model creation by adding the three layers : Sequential, Dense and Hidden layers.
- Finally the model now created is compiled, fitted and saved in the form of h5 file.
- Now the h5 file is very prominent and can be used for prediction of the images while we upload those images on our web application.

Customer Problem Statement Template:

I am Marine researcher	I'm trying to study the different underwater species to understand their behaviour	But unable to identify species	Because of poor quality images	Which makes me feel frustrated <small>micro</small>
I am Wildlife Photographer	I'm trying to capture exclusive images of endangered species	But cannot find their gathering places anywhere near my place of stay	Because the animals have migrated to the middle of the forest due to lack of resources	Which makes me feel Exhausted <small>micro</small>
I am Zoologist	I'm trying to Preserve a particular sepecies	But The species is found very rarely on Earth	Because It is extremely endangered in wildlife	Which makes me feel outraged
I am ornithologist	I'm trying to trying to identify different species of birds	But cannot find suitable place of work in the national park	Because of poor management in the national park	Which makes me feel outraged <small>micro</small>

3.IDEATION & PROPOSED SOLUTION

3.1 Empathy Map Canvas



3.2 Ideation & Brainstorming

Template

Brainstorm & idea prioritization

Use this template in your own brainstorming sessions so your team can unleash their imagination and start shaping concepts even if you're not sitting in the same room.

10 minutes to prepare
 1 hour to collaborate
 2-8 people recommended

Date	19 September 2022
Team ID	PNT2022TMID18534
Project Name	Digital Naturalist - AI Enabled Tool For Biodiversity Researchers
Maximum Marks	4 Marks

[Share template feedback](#)

Before you collaborate

A little bit of preparation goes a long way with this session. Here's what you need to do to get going.

10 minutes

- Team gathering**
Define who should participate in the session and send an invite. Share relevant information or pre-work ahead.
- Set the goal**
Think about the problem you'll be focusing on solving in the brainstorming session.
- Learn how to use the facilitation tool**
Use the Facilitation Superpowers to run a happy and productive session.
[Open article](#) →

1 Define your problem statement

What problem are you trying to solve? Frame your problem as a How Might We statement. This will be the focus of your brainstorm.

5 minutes

PROBLEM

How might we solve [your problem statement]?

Key rules of brainstorming

To run an smooth and productive session

Stay in topic.

Encourage wild ideas.

Defer judgment.

Listen to others.

Go for volume.

If possible, be visual.

Need some inspiration?

See a finished version of this template to kickstart your work.

[Open example](#) →

2

Brainstorm

Write down any ideas that come to mind that address your problem statement.

🕒 10 minutes

Shankar

To Create a web Application

And Identify those Species

To Predict The Animals Species

To Collect the Species Images

To Calssify that videos in categories

To store and retrieve from the database

To Predict The Flower and Plant species

That will looks the web attractive

To build web

Shri Kamal

To identify using object identification

To test the CNN model using the another images

And identify those Species

To Detect the images in that videos

Create the mobile app

To upload the information in form of label

To collect The Flower, plants and Animal Species images

Build the Computer Vision mode

To make category

Soundhar

To collect the videos of Flora and fauna

Upload the videos of those Species in Web

To collect the all the images

Use Computer Vision

To Detect Flora and fauna Species

To Build Web app using Flask

Save and implement the model in Web App

Upload the image in Web app

To Build CNN model

Shobanram

To implement that Model

This could be a Handy tool for Bio diversity

To Calssify the different species

To upload the images of any species

The analytics are display about the species

The database retrieve that image information

The database Are stored with all images

Use Data Visualization using python

It could detect only the learned objects



3

Group ideas

Take turns sharing your ideas while clustering similar or related notes as you go. Once all sticky notes have been grouped, give each cluster a sentence-like label. If a cluster is bigger than six sticky notes, try and see if you can break it up into smaller sub-groups.

⌚ 20 minutes



4

Prioritize

Your team should all be on the same page about what's important moving forward. Place your ideas on this grid to determine which ideas are important and which are feasible.

⌚ 20 minutes



3.3 Proposed Solution

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	<ul style="list-style-type: none"> ✓ Helps Naturalists and Non-Naturalists users or the common people who go for hikes , canoe trips , excursions to explore the different species of flora and fauna found in that terrain ✓ Provides the name of the different plants and animals along with their names in seven taxonomic ranks namely kingdom, phylum or division, class, order, family, genus and species ✓ Naturalists and Non-Naturalists are shown the characteristics and other fascinating facts about the flora and fauna ✓ Naturalists and Non-Naturalists need to know about the poisonous and dangerous plants and animals in order to stay away from them, medicinal values of plant to use in case of emergencies and rarity of various species of plants, animals and birds so that they can take necessary steps to conserve them.
2.	Idea / Solution description	<ul style="list-style-type: none"> ✓ Display biological names of flora and fauna ✓ What type of disease does the plant cure ✓ Show alert messages for plants/ animals using different colours and in which way they are highlighted ✓ Display rarities of the species ✓ Description about the species
3.	Novelty / Uniqueness	<ul style="list-style-type: none"> ✓ Alerting the user based on rarity of the species found ✓ Complete description about the characteristics of the species

		<ul style="list-style-type: none"> ✓ Alert the users if the species is dangerous or not ✓ Giving the medicinal values of plants and its description ✓ Displaying the names in 7 taxonomical levels of each flora and fauna
4.	Social Impact / Customer Satisfaction	<ul style="list-style-type: none"> ✓ Identifying the flora and fauna in our locality / environment helps in improving and understanding biodiversity and the importance of conserving and preserving them for our future generations ✓ Establishment of more national parks and wildlife sanctuaries
5.	Business Model (Revenue Model)	<ul style="list-style-type: none"> ✓ Introducing Subscription can be a major source of revenue for the application ✓ Partnership with many naturalists, universities and scientists around the world
6.	Scalability of the Solution	<ul style="list-style-type: none"> ✓ As the application grows more popular, new and innovative features can be added now and then ✓ As the number of users for the application grows steadily, we can introduce the application's version in different languages spoken in different geographical locations ✓ Subscriptions can be classified according to the type of users including the Normal plan, Educational plan, and Business plan

3.4 Problem Solution fit

Define CS, fit into CC	<p>1. CUSTOMER SEGMENT(S) CS</p> <p>Persons those who have interest in bio-diversity researches, identifying and classify objects in a frame that contains a large number of objects. On account of increasing computer vision strategies the rate of accuracy has been increasing significantly. It aims to implement the unique set of rules for</p>	<p>6. CUSTOMER CONSTRAINTS L</p> <p>The network issues are the biggest constraints. The increasing availability of the digital images, Edited images, coupled with AI techniques for image classification, presents an exciting opportunities for the biodiversity researchers to create a new datasets for species observations.</p>	<p>5. AVAILABLE SOLUTIONS L</p> <p>Developing a solution that can be able to identify the correct species , location and the environment for the given input image would be beneficial for lots character as well as ornithologist. Merits : interplay between the individual &</p>	Explore AS, differentiate
Focus on J&P, tap into BE, understand RC	<p>2. JOBS-TO-BE-DONE / PROBLEMS J&P</p> <p>One of the maximum struggle faced by using the character are biodiversity, or the form of all the matters on our planet, has been declining at an alarming charge in recent years, in particular because of human sports together with land use adjustments, pollution , weather and all other modifications happened.</p>	<p>9. PROBLEM ROOT CAUSE RC</p> <ol style="list-style-type: none"> 1. Field naturalist frequently use tried-and-true strategies when exploring the woods, such as always having a guidebook with them 2. Asking an expert ornithologist for assistance. 3. Inadequate documentation. 4. Not having a training set with them. 	<p>7. BEHAVIOUR BE</p> <p>The customers whoever going to spend their time in this application can able to searching the species and find the right species which is an inbuilt technology that performs the right action. They always look for finding the new species.</p>	Focus on J&P, tap into BE, understand RC

3. TRIGGERS TR <ol style="list-style-type: none"> Welcome message Onboarding Early activities 	10. YOUR SOLUTION SL <p>This application is an image sharing and retrieval application for the identification of various species in the Earth. This app is available on android. It is something different from other applications that can work with several species. Bio diversity is the life support system. The project's goal is to develop a web application that would allow travellers to recognize uncommon species of animals, flowers, and other plants. The artificial intelligence computer model is used in the suggested system for biodiversity study aids in the detection of environments, specific species, and locations.</p>	8.CHANNELS of BEHAVIOUR CH <p>8.1 ONLINE</p> <ol style="list-style-type: none"> Visit a landing Page Download content Provide feedback Submit an Email Refer a friend
4. EMOTIONS: BEFORE / AFTER EM <ol style="list-style-type: none"> Reduce customer chum Acquire more customers Cultivate customer loyalty 		<p>8.2 OFFLINE</p> <p>Decide the smallest amount of data for storing locally.</p>

4. REQUIREMENT ANALYSIS

4.1 Functional Requirements

Following are the functional requirements of the proposed solution

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration (Optional)	Registration through Gmail Registration through Form
FR-2	User Confirmation (Optional)	Confirmation via Email & OTP
FR-3	Dataset	2 Birds, 2 Flowers and 2 Mammals
FR-4	API	IBM Based API
FR-5	User details (Optional)	Users are required to register their personal details. . Like name, age, phone no, email
FR-6	Updating and bug fixing	Updated Based on user feedback
FR-7	Final Output	Final description of the image captured.
FR-8	Description	Facts about different plants and animals (system)

4.2 Non-Functional Requirements

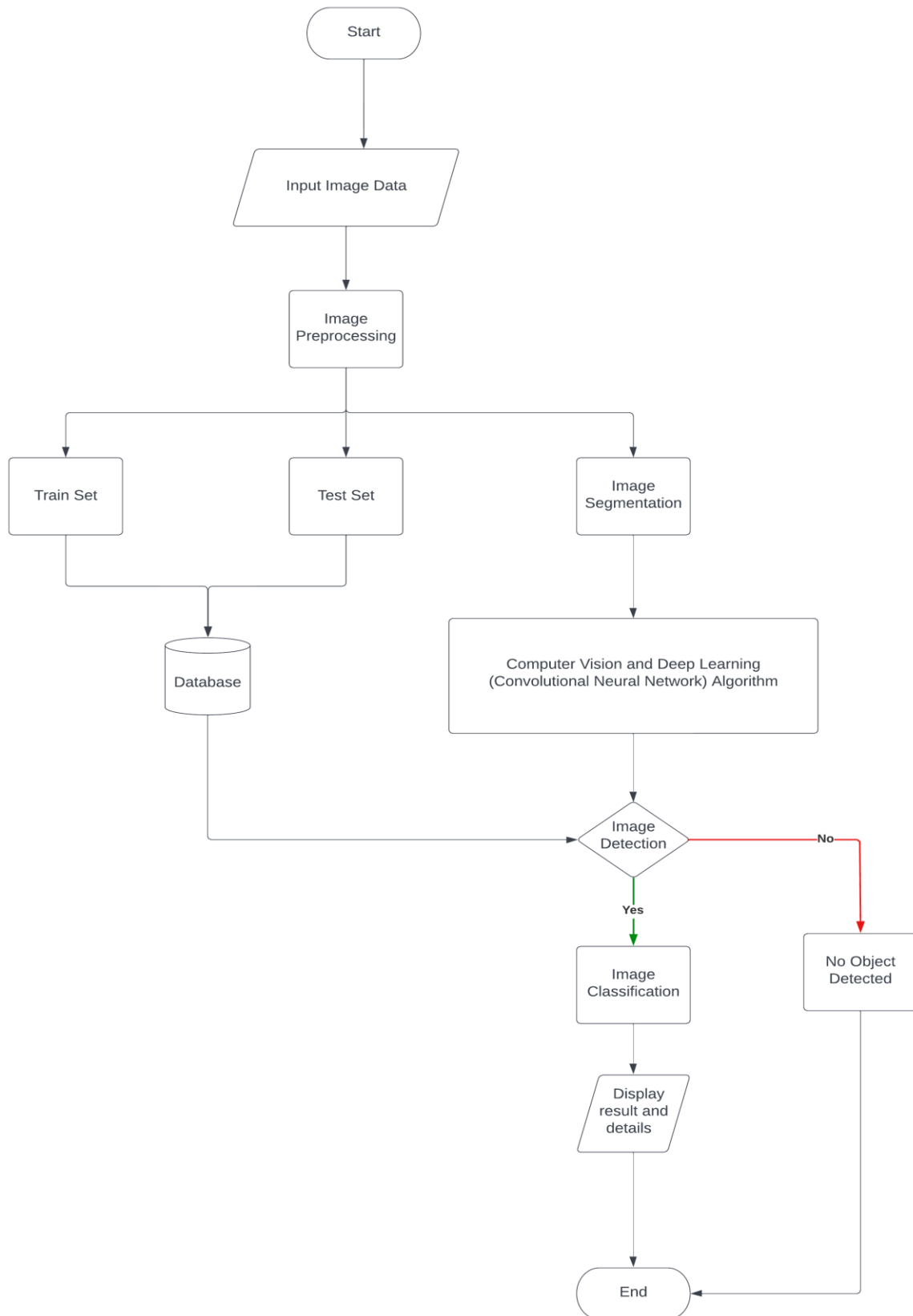
Following are the non-functional requirements of the proposed solution.

FR No.	Non-Functional Requirement	Description
NFR-1	Usability	Efficient for the frequent users. Users can easily understand what the application does and feel satisfied with the system.
NFR-2	IBM Cloud Security	Workload protection, X-Force Red offensive security services and Data security services
NFR-3	Reliability	The system must perform without failure in most of the time.
NFR-4	Performance	This application supporting maximum users per hour must provide less response time including the rendering of text and images.
NFR-5	Availability	The application will be available most of the time in a month.
NFR-6	Scalability	The application must be scalable enough to support maximum visits at the same time while maintaining optimal performance

5. PROJECT DESIGN

5.1 Data Flow Diagrams

1. User captures photo using camera
2. Dataset contains photos that are used for training and testing
3. Photo is sent to CNN for classification and processing based on dataset
4. Processed image is checked with database for additional info
5. Photo from CNN algorithm is sent to the user
6. Matching data from database is sent to the use

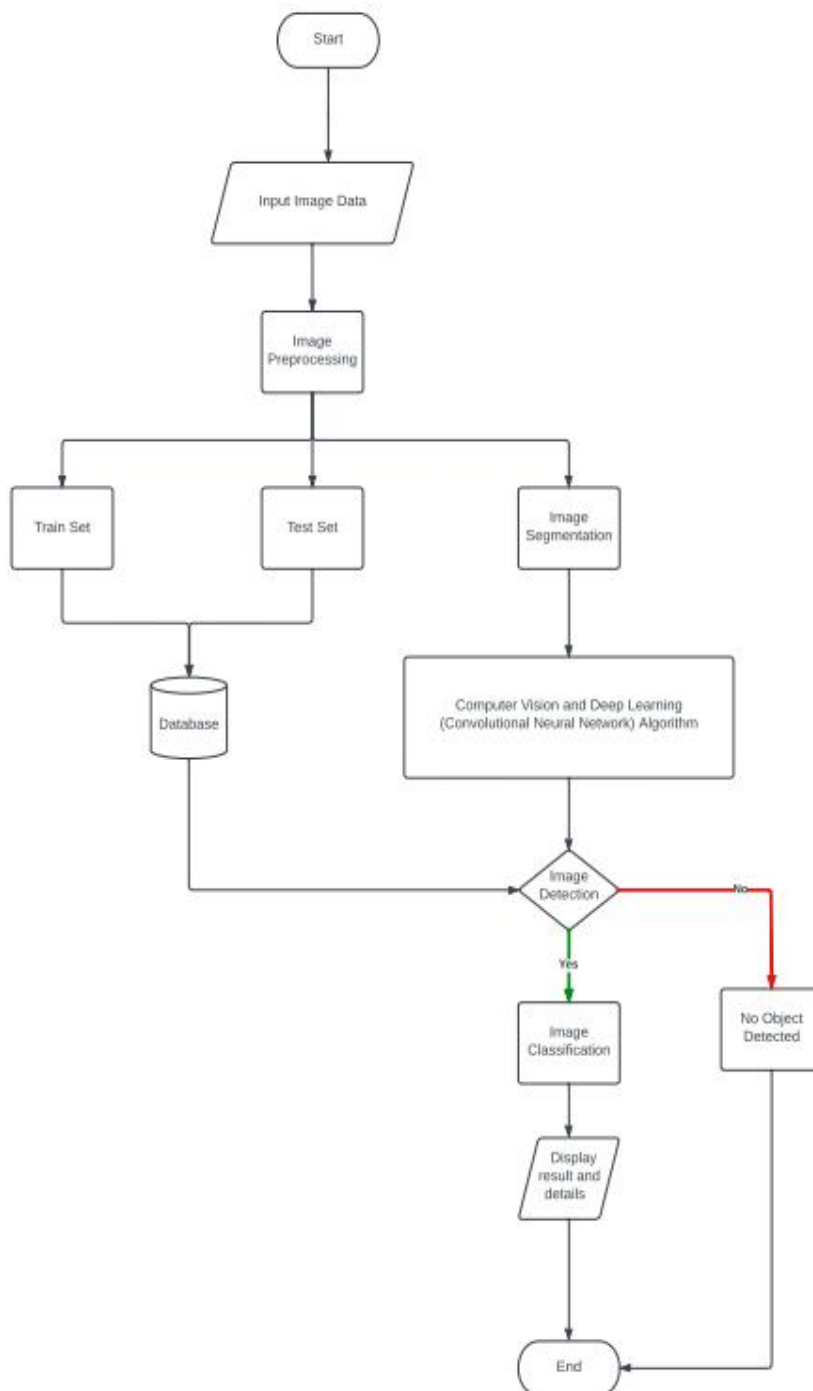


5.2 Solution & Technical Architecture

SOLUTION ARCHITECTURE

Project Name : Digital Naturalist- AI Enabled tool for
Biodiversity Researchers

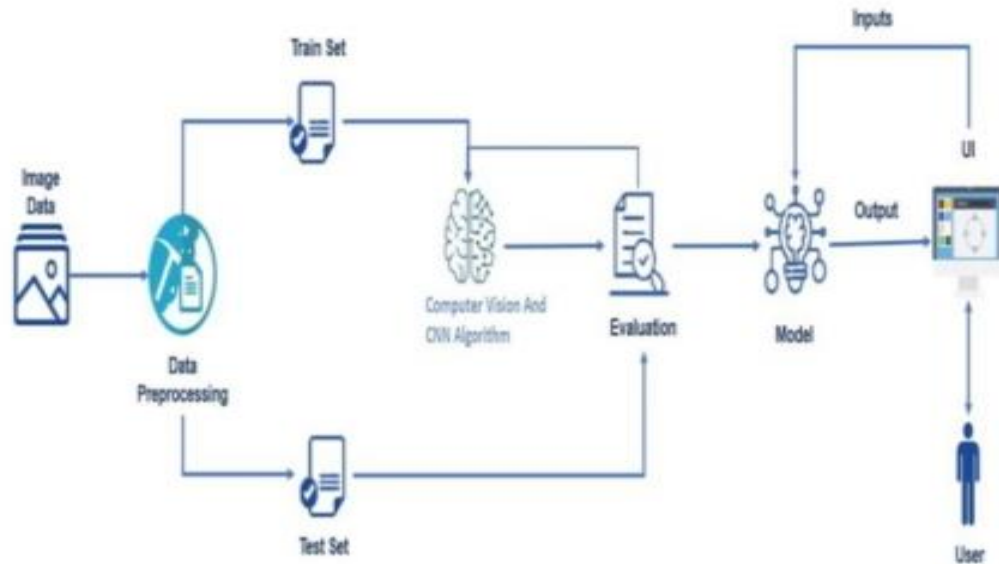
Team ID: PNT2022TMID18534



Technical Architecture:

The Deliverable shall include the architectural diagram as below and the information as per the table1 & table 2

Reference: <https://developer.ibm.com/patterns/ai-powered-backend-system-for-order-processing-during-pandemics/>



Guidelines:

1. Include all the processes (As an application logic / Technology Block)
2. Provide infrastructural demarcation (Local / Cloud)
3. Indicate external interfaces (third party API's etc.)
4. Indicate Data Storage components / services
5. Indicate interface to machine learning models (if applicable)

Table-1 : Components & Technologies:

S.No	Component	Description	Technology
1.	User Interface	How user interacts with application e.g. Web UI, Mobile App, Chatbot etc.	HTML, CSS, JavaScript / Angular JS / React JS etc.
2.	Camera	Capture photos for processing	Normal phone camera
3.	CNN Algorithm	For processing photos	—
4.	Database	Data Type, Configurations etc.	MySQL, etc.
5.	Cloud Database	Database Service on Cloud	IBM DB2, IBM Cloudant etc.
6.	File Storage	File storage requirements	IBM Block Storage or Other Storage Service or Local Filesystem
7.	External API-1	Location service	Location or google api

Table-2: Application Characteristics:

S.No	Characteristics	Description	Technology
1	Open-Source Frameworks	Datasets, API , etc	SQL or CSV
2	Security Implementations	List all the security / access controls implemented, use of firewalls etc.	e.g. SHA-256, Encryptions, IAM Controls, OWASP etc.
3	Scalable Architecture	Justify the scalability of architecture (3 – tier, Micro-services)	Python , MySQL
4	Availability	Justify the availability of application (e.g. use of load balancers, distributed servers etc.)	Python , MySQL
5	Performance	Design consideration for the performance of the application (number of requests per sec, use of Cache, use of CDN's) etc.	—

References:

<https://c4model.com/>
<https://developer.ibm.com/patterns/online-order-processing-system-during-pandemic/>
<https://www.ibm.com/cloud/architecture>
<https://aws.amazon.com/architecture>
<https://medium.com/the-internal-startup/how-to-draw-useful-technical-architecture-diagrams-2d20c9fda90d>

5.3 User Stories

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority
Customer (Mobile user)	Image capture	USN-1	As a user, I can take photos of the plant life, animals and birds	I can take photos when required	High
		USN-2	As a user, I will receive processed information about the type of species	I can see the type of plant or animal or plant	High
		USN-3	As a user, I can share it with others	I can share using share option	Low
	Data process	USN-4	Data must be trained and tested and CNN algorithm must work properly.	I must see the correct processed information	High
	Output	USN-5	As a user, I can see the scientific name of the species	I must see the correct data	High
		USN-6	As a user, I can see the characteristics and alert messages	I must see the correct data	High
Administrator	Manage	USN-1	As a admin I must add various data and edit information	I must edit the data present	High

6. PROJECT PLANNING & SCHEDULING

6.1 Sprint Planning & Estimation

S.No	Milestone	Activities	Team Members
1.	Data Collection	Create Train and Test Folders	Shri Kamal S Soundhararajan S Shobanram R
2.	Image Preprocessing	Import ImageDataGenerator Library and Configure	Shankar Subramaniyam G Shri Kamal S
3.	Image Preprocessing	Apply ImageDataGenerator functionality to Train and Test set	Shri Kamal S Shankar Subramaniyam G Soundhararajan S
4.	Model Building	Import the required model building libraries	Shankar Subramaniyam G Shobanram R
5.	Model Building	Initialize the model	Shankar Subramaniyam G Shobanram R Soundhararajan S
6.	Model Building	Add the convolution layer	Shankar Subramaniyam G Soundhararajan S
7.	Model Building	Add the pooling layer	Shri Kamal S Shankar Subramaniyam G Soundhararajan S
8.	Model Building	Add the flatten layer	Shobanram R Soundhararajan S

9.	Model Building	Adding the dense layers	Shankar SubramaniyamG Shobanram R Soundhararajan S
10.	Model Building	Compile the model	Shobanram R Soundhararajan S
11.	Model Building	Fit and save the model	Shri Kamal S Shankar SubramaniyamG Soundhararajan S
12.	Test the model	Import the packages andload the saved model	Soundhararajan SShri Kamal S

6.2 Sprint Delivery Schedule

Product Backlog, Sprint Schedule, and Estimation

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Dataset Modelling Phase	USN-1	Data Collecting and digitalizing for analysing	3	Medium	SHANKAR SUBRAMANIYAM G
Sprint-1		USN-2	Adding more data to avoid overfitting	2	Medium	SOUNDHARARAJAN S
Sprint-1		USN-3	Building a CNN model using the collected data	5	High	SHRI KAMAL S
Sprint-1		USN-4	Evaluating the model to check the accuracy and precision	3	High	SHOBANRAM R
Sprint-2	Development Phase	USN-5	Home page Creation – Shows the features of our application	1	Low	SHOBANRAM R
Sprint-2		USN-6	Setting up facilities for user to feed the image	2	Medium	SOUNDHARARAJAN S
Sprint-2		USN-7	Prediction page creation – shows prediction for the user given image	4	Medium	SHANKAR SUBRAMANIYAM G
Sprint-2		USN-8	Model loading – API creation using flask	5	High	SHRI KAMAL S
Sprint-3	Deployment Phase	USN-9	Integrating UI & backend – Connecting the front end and backend using API calls	3	Medium	SHOBANRAM R
Sprint-3		USN-10	Cloud deployment – Deployment of application using IBM Cloud	5	High	SHANKAR SUBRAMANIYAM G
Sprint-4	Testing Phase	USN-11	Functional testing – Checking the scalability and robustness of the application	5	High	SHANKAR SUBRAMANIYAM G
Sprint-4		USN-12	Non-Functional testing – Checking for user acceptance and integration	5	High	SOUNDHARARAJAN S, SHRI KAMAL S

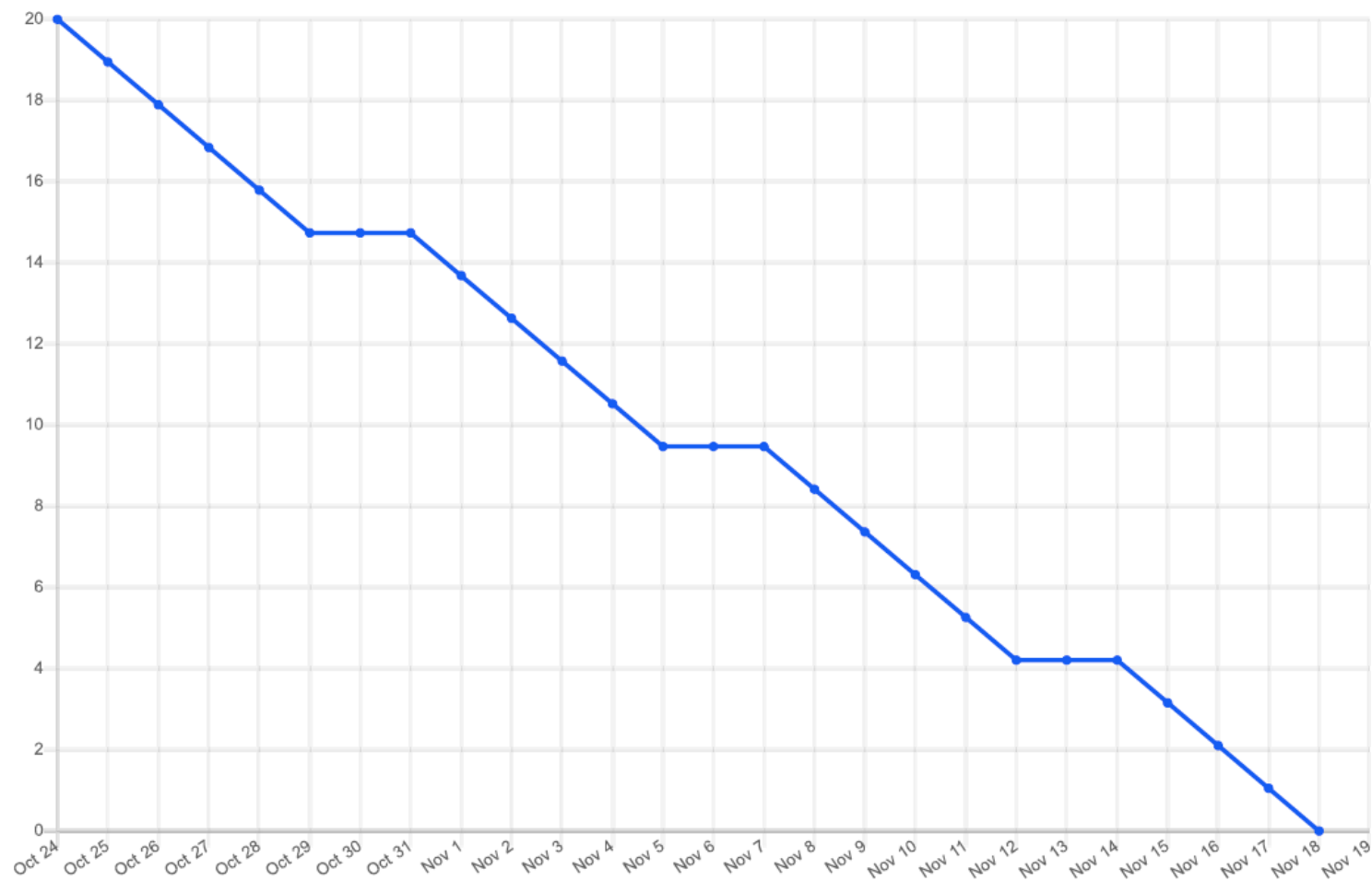
Project Tracker, Velocity & Burndown Chart: (4 Marks)

Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
Sprint-1	13	6 Days	24 Oct 2022	29 Oct 2022	13	29 Oct 2022
Sprint-2	12	6 Days	31 Oct 2022	05 Nov 2022	12	05 Nov 2022
Sprint-3	8	6 Days	07 Nov 2022	12 Nov 2022	8	12 Nov 2022
Sprint-4	10	6 Days	14 Nov 2022	19 Nov 2022	10	19 Nov 2022

Velocity:

Average Velocity = $61/24 = 2.51$

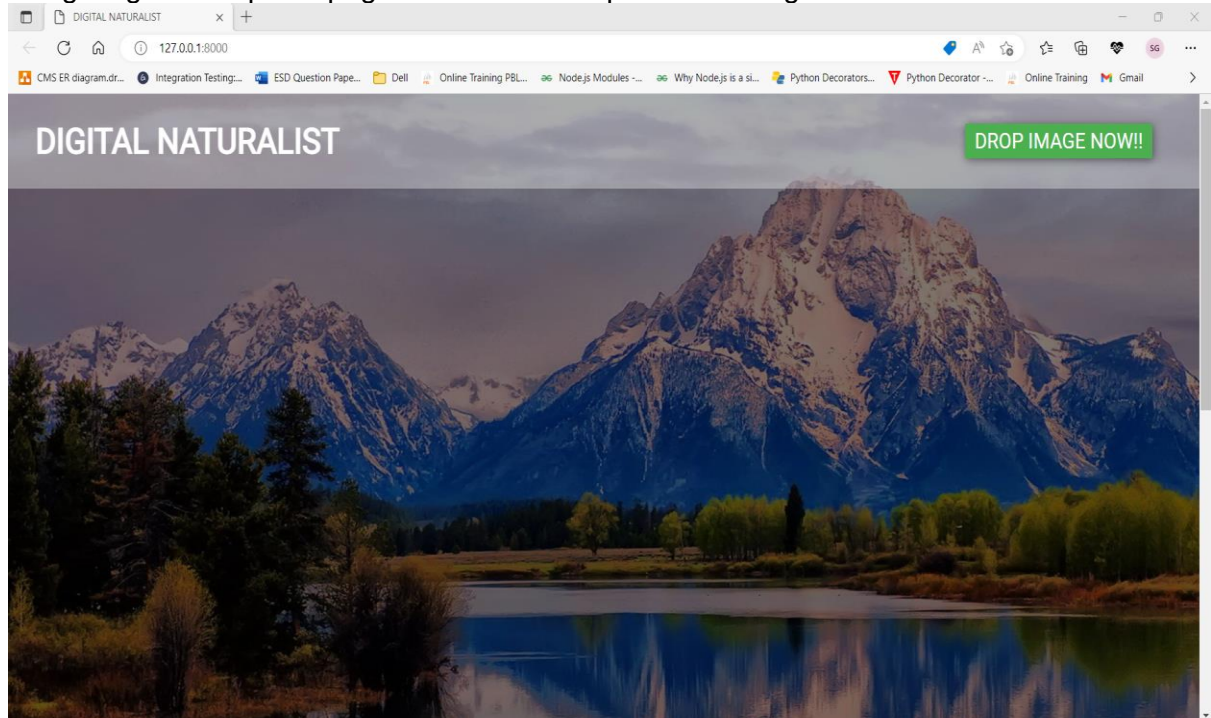
Burndown Chart :



7.CODING & SOLUTIONING

7.1 Feature 1

- A welcome page is designed and a button “DROP THE IMAGE NOW” will be navigating to the upload page where we can upload the images.



welcome.html

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">

  <script
src="https://ajax.googleapis.com/ajax/libs/jquery/3.4.1/jquery.min.js"></sc
ript>
    <link rel="stylesheet"
href="https://ajax.googleapis.com/ajax/libs/jqueryui/1.12.1/themes/smoothne
ss/jquery-ui.css">
<script src="https://ajax.googleapis.com/ajax/libs/jqueryui/1.12.1/jquery-
ui.min.js"></script>
<script
src="https://cdnjs.cloudflare.com/ajax/libs/chosen/1.8.7/chosen.jquery.min.
js"></script>
<link rel="stylesheet"
href="https://cdnjs.cloudflare.com/ajax/libs/chosen/1.8.7/chosen.min.css">
<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/4.7.0/css/font-awesome.min.css">
<link rel="stylesheet" type="text/css" href="swiper.min.css">

  <link
href="https://fonts.googleapis.com/css2?family=Roboto&display=swap"
rel="stylesheet">
```

```

    <link
href="https://fonts.googleapis.com/css2?family=Playfair+Display&display=swa
p" rel="stylesheet">
    <link
href="https://fonts.googleapis.com/css2?family=Dancing+Script:wght@700&disp
lay=swap" rel="stylesheet">
    <link rel="stylesheet" href="/static/welcome.css">
    <title>DIGITAL NATURALIST </title>
</head>
<body>

    <section class = "hero">
        <header>
            <!-------Logo----->

            <h1><a href="welcome.html"> DIGITAL NATURALIST </a></h1>

            <!-------Login Section----->

            <section id="login">
                <a href="index" id="login">DROP IMAGE NOW!!</a>
<!--        <a href="http://localhost/agriportal/login.php"
id="login">Login</a>-->
<!--        <a href="signup.html" id="signup">Signup</a>-->
            </section>
        </header>

<!--        <div class = "hero-text">-->
<!--            <h2>Connecting With Nature</h2>-->
<!--            <p>Lorem ipsum dolor sit amet consectetur adipisicing elit.
Earum eaque dolores numquam tenetur laborum quis, facilis dolorem,
voluptatum? Soluta illum, similique dolorum obcaecati exercitationem culpa,
vitae voluptatibus repellendus earum voluptates saepe hic sed perferendis
inventore perspiciatis rem asperiores quibusdam enim odio tempore sequi et
facilis. Magni est, ipsam pariatur perspiciatis.</p>-->
<!--            </div>-->

    </section>

    <!-------Counters----->

    <section class="counters">
        <div class="container2">
            <div>
                <i class="fa fa-users fa-4x"></i>
                <div class="counter" data-target="60000">0</div>
                <h3>Visitors</h3>
            </div>
            <div>
                <i class="fa fa-youtube fa-4x"></i>
                <div class="counter" data-target="40000">0</div>
                <h3>Subscribers</h3>
            </div>
            <div>
                <i class="fa fa-twitter fa-4x"></i>
                <div class="counter" data-target="15000">0</div>
                <h3>Followers</h3>
            </div>
        </div>
    </section>

```

```

        </div>
        <div>
            <i class="fa fa-facebook fa-4x"></i>
            <div class="counter" data-target="9000">0</div>
            <h3>Likes</h3>
        </div>
        <div>
            <i class="fa fa-linkedin fa-4x"></i>
            <div class="counter" data-target="5000">0</div>
            <h3>Connections</h3>
        </div>
    </div>
</section>

<!-------
-----
-----Footer-----
-----
----->

<div class="footer">
    <div class="footer-content">
        <div class="footer-section about">
            <h1 class="logo-text"><span>DIGITAL</span>NATURALIST</h1>
            <div class="contact">
                <p> DESIGNED BY SHRI KAMAL S AND SHANKAR G </p>-->
                <span><i class="fa fa-phone">&nbsp; 123-456-789</i></span>
                <span><i class="fa fa-envelope">&nbsp;
info@digitalnature4u.com</i></span>
            </div>
            <div class="socials">
                <a href="#"><i class="fa fa-facebook"></i></a>
                <a href="#"><i class="fa fa-instagram"></i></a>
                <a href="#"><i class="fa fa-twitter"></i></a>
                <a href="#"><i class="fa fa-youtube"></i></a>
            </div>
        </div>
        <div class="footer-section links">
            <h2>Get to know us</h2><br>
            <ul>
                <a href="#"><li>About us</li></a>
                <a href="#"><li>Culture</li></a>
                <a href="#"><li>Blog</li></a>
                <a href="#"><li>Careers</li></a>
                <a href="#"><li>Report fraud</li></a>
                <a href="#"><li>Contact us</li></a>
            </ul>
        </div>
        <div class="footer-section links">
            <h2>More</h2><br>
            <ul>
                <a href="#"><li>Privacy</li></a>
                <a href="#"><li>Terms</li></a>
                <a href="#"><li>API Policy</li></a>
                <a href="#"><li>CSR</li></a>
                <a href="#"><li>Security</li></a>
                <a href="#"><li>Sitemap</li></a>
            </ul>
        </div>
    </div>
</div>

```

```

    <div class="footer-bottom">
      Copyrights &copy; 2020-2040 DIGITAL NATURALIST, All Rights
Reserved.
    </div>
</div>
<script type="text/javascript" src="swiper.min.js"></script>
    <script>
      var swiper = new Swiper('.swiper-container', {
        effect: 'coverflow',
        grabCursor: true,
        centeredSlides: true,
        slidesPerView: 'auto',
        coverflowEffect: {
          rotate: 50,
          stretch: 0,
          depth: 100,
          modifier: 1,
          slideShadows : true,
        },
        pagination: {
          el: '.swiper-pagination',
        },
      });
    </script>
<script src="main.js"></script>
<script src="js/main.js"></script>
</body>
    <script type="text/javascript">
      $(".chosen").chosen();
    </script>

</body>
</html>

```

welcome.css

```

:root{
  --transWhite: #fdfdfd5c;
  --themeBlue: #013781;
  --themeGreen: #4caf50;
  --themeOrangeSelect: #c00202;
  --offWhite: #F4F4F4;
  --offBlack: #313131;
  --shapeShadow: 1px 2px 10px #00000078;
}

body{
  width: 100%;
  margin: 0;
  padding: 0;
  box-sizing: border-box;
  font-family: 'Roboto', sans-serif;
}

body div, section, article, header, button{
  box-sizing: border-box;
}

a{
  font-family: 'Roboto', sans-serif;
  text-decoration: none;
}

```

```

}

button{
  border: none;
  box-shadow: var(--shapeShadow);
}

h2.secHeading{
  color: var(--themeBlue);
  font-size: 2em;
}

button.viewAll{
  padding: 0.5em;
  background-color: var(--themeBlue);
  font-size: 1.5em;
  color: white;
}

button.viewAll:hover{
  background-color: var(--themeGreen);
}

/*-----
-----
-----Hero Image-----
-----
-----*/

section.hero{
  width: 100%;
  height: 100vh;
  background-image: linear-gradient( rgba(0, 0, 0, 0.5), rgba(0, 0, 0, 0.5) ),url("/static/img/background.jpg");
  position: relative;
}

section.hero div.hero-text{
  color: var(--transWhite);
  position: absolute;
  top: 30%;
  text-align: center;
}

section.hero div.hero-text h2{
  font-size: 4em;
  font-family: 'Dancing Script', cursive;
}

section.hero div.hero-text p{
  font-family: 'Roboto', sans-serif;
  font-size: 1.5em;
}

/*-----
-----
-----Header-----
-----
-----*/

```

```

-----*/
header{
  background-color: var(--transWhite);
  padding: 1em;
  margin: 0;
  width: 100%;
  height: 15vh;
  display: flex;
  align-items: center;
}

/*-----H1-----*/
-----*/

header h1{
  margin: 0;
  font-size: 2.5em;
  padding-left: 0.5em;
  flex: 10%;
}

header h1 a{
  color: white;
}

/*-----Login Section-----*/
-----*/

header section#login{
  padding-right: 2em;
  flex: 15%;
  margin: 0;
  display: flex;
  justify-content: flex-end;
}

header section#login a{
  color: white;
  font-size: 1.5em;
  padding: 0.2em 0.5em;
  margin-right: 0.5em;
}

header section#login a#login{
  background-color: var(--themeGreen);
  box-shadow: var(--shapeShadow);
}

header section#login a#login:hover{
  background-color: white;
  border: solid 2px var(--themeGreen);
  color: var(--themeGreen);
}

header section#login a#signup{
  color: white;
}

header section#login a#signup:hover{
  color: var(--themeGreen);
}

```



```

    text-decoration: underline;
}

/*-----
-----Counters-----
-----*/

.counters{
    display: flex;
    flex-direction: column;
    justify-content: flex-end;
    height: 36vh;
    padding-bottom: 50px;
    color: teal;
}

.container2{
    max-width: 1100px;
    margin: 0 auto;
    overflow: auto;
}

.counters .container2{
    display: grid;
    grid-gap: 90px;
    grid-template-columns: repeat(5, 1fr) ;
    text-align: center;
}

.counters i{
    margin-bottom: 5px;
}

.counters .counter{
    font-size: 45px;
    margin: 5px 0;
}

@media(max-width: 700px){
    .counters .container2{
        grid-template-columns: repeat(2, 1fr);
    }
    .counters .container2 > div:nth-of-type(1),.counters .container2 >
div:nth-of-type(2){
        border-bottom: 1px lightskyblue solid;
        padding-bottom: 20px;
    }
}

section.counters{
    margin-top: 50px;
    color: var(--themeGreen);
}

/*-----Footer-----
-----*/

.footer{
    background: #303036;
    color: #d3d3d3;
    height: 400px;
    position: relative;
}

.footer .footer-content{

```

```
    height: 350px;
    display: flex;
}
.footer .footer-content .footer-section{
    flex: 1;
    padding: 25px;
}
.footer .footer-content h1,
.footer .footer-content h2{
    color: white;
    font-family: 'Oswald', sans-serif;
}
.footer .footer-content .about h1 span{
    color: var(--themeGreen);
}

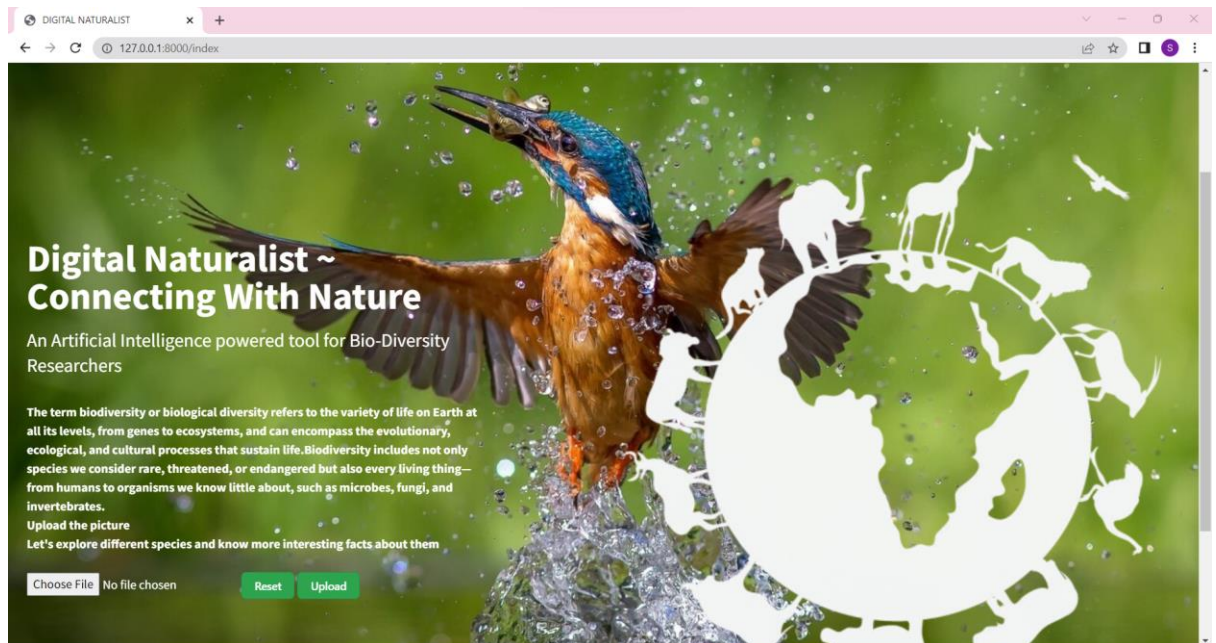
.footer .footer-content .about .contact p
{
    color: white;
    font-family: 'Oswald', sans-serif;
}

.footer .footer-content .about .contact span{
    display: block;
    font-size: 1.2rem;
    margin-bottom: 8px;
}
.footer .footer-content .about .socials a{
    border: 1px solid gray;
    width: 45px;
    height: 41px;
    padding-top: 5px;
    margin-right: 5px;
    text-align: center;
    display: inline-block;
    font-size: 1.3rem;
    border-radius: 5px;
    color: gray;
    transition: all .3s;
}
.footer .footer-content .about .socials a:hover{
    border: 1px solid white;
    color: #fff;
    transition: all .3s;
}
.footer .footer-content .links ul a{
    color: gray;
    display: block;
    margin-bottom: 10px;
    font-size: 1.2rem;
}
.footer .footer-content .links ul a:hover{
    color: white;
    margin-left: 15px;
    transition: all .3s;
}
.footer .footer-bottom{
    background: #343a40;
    color: #686868;
    height: 50px;
    width: 100%;
}
```

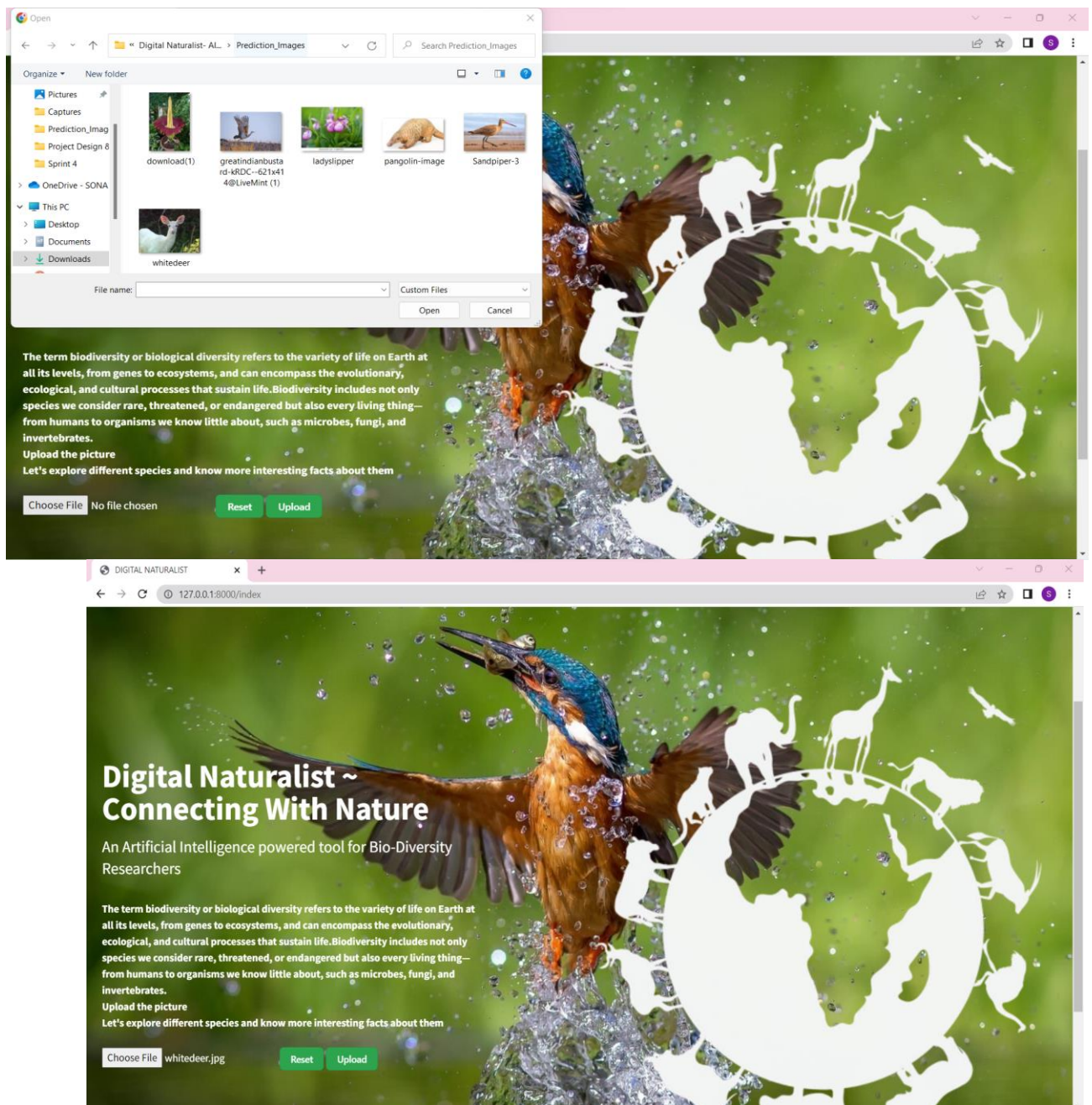
```
text-align: center;
position: absolute;
bottom: 0px;
left: 0px;
padding-top: 10px;
}
```

7.2 Feature 2

- The upload page is redirected and an image can be uploaded by clicking Choose File button available and then click Upload button or we can even re-upload or cancel the image already uploaded by clicking Reset button.



- Now the image is chosen and then the filename of that image is displayed to indicate that this image has been chosen by the user.



index.html

```
<!DOCTYPE html>
<html lang="en">

<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <title>DIGITAL NATURALIST</title>
  <meta name="description" content="">
  <meta name="keywords" content="">
  <link
href="https://fonts.googleapis.com/css?family=Source+Sans+Pro:400,700"
rel="stylesheet">
  <link rel="stylesheet" href="/static/style.css">
```

```

</head>

<body class="leading-normal tracking-normal text-gray-900" style="font-
family: 'Source Sans Pro', sans-serif;">

    <div class="h-screen pb-14 bg-right bg-cover">
        <!--Nav-->
        <div class="w-full container mx-auto p-6">

            </div>

            <!--Main-->
            <div class="container pt-24 md:pt-48 px-6 mx-auto flex flex-wrap
flex-col md:flex-row items-center">

                <!--Left Col-->
                <div class="flex flex-col w-full xl:w-2/5 justify-center lg:items-
start overflow-y-hidden" style="color : white">
                    <h1
                        class="my-4 text-3xl md:text-5xl text-white-800 font-bold
leading-tight text-center md:text-left slide-in-bottom-h1">
                        Digital Naturalist ~ Connecting With Nature</h1>
                    <p class="leading-normal text-base md:text-2xl mb-8 text-center
md:text-left slide-in-bottom-subtitle">
                        An Artificial Intelligence powered tool for Bio-Diversity
Researchers</p>

                    <p class="text-white-400 font-bold pb-8 lg:pb-6 text-center
md:text-left fade-in">The term biodiversity or biological diversity refers
to the variety of life on Earth at all its levels, from genes to
ecosystems, and can encompass the evolutionary, ecological, and cultural
processes that sustain life.Biodiversity includes not only species we
consider rare, threatened, or endangered but also every living thing—from
humans to organisms we know little about, such as microbes, fungi, and
invertebrates.<br>Upload the picture<br>Let's explore different species and
know more interesting facts about them<br>
                    <!--DOCTYPE html>
                    </p>
                    <div class="flex w-full justify-center md:justify-start pb-24
lg:pb-0 fade-in">
                        <form action="/predict" id="upload-file" method="post"
enctype="multipart/form-data">
                            <input type="file" name="uploadedimg" id="uploadedimg"
required accept=".jpg, .png, .jpeg, .gif, .bmp, .tif, .tiff|image/*" >
                            <input type="reset" value="Reset" class="upload">
                            <input type="submit" value="Upload" class="upload"
onsubmit="check_file">
                        </form>
                    </div>

                </div>

                <!--Right Col-->
                <div class="w-full xl:w-3/5 py-6 overflow-y-hidden">
                    
                </div>

```

```

        <!--Footer-->
        <div class="w-full pt-16 pb-6 text-sm text-center md:text-left
fade-in">
            <a class="text-gray-500 no-underline hover:no-underline"
                href="https://github.com/IBM-EPBL/IBM-Project-26473-
1660027397">&copy; Digital Naturalist</a>
            </div>

        </div>

    </div>

</div>

<script>
document.getElementById("uploadedimg").addEventListener("change",
validateFile)

function validateFile(){
    const allowedExtensions = ['jpg','png'],
        sizeLimit = 1_000_000;
    const { name:fileName, size:fileSize } = this.files[0];
    const fileExtension = fileName.split(".").pop();
    if(!allowedExtensions.includes(fileExtension)){
        alert("Only image files - .jpg, .jpeg, .png, .tiff ");
        this.value = null;
    }else if(fileSize > sizeLimit){
        alert("file size too large")
        this.value = null;
    }
}
</script>
</body>

</html>

```

style.css

```

.slide-in-bottom {
    -webkit-animation: slide-in-bottom .5s cubic-bezier(.25, .46, .45, .94)
both;
    animation: slide-in-bottom .5s cubic-bezier(.25, .46, .45, .94) both
}

.slide-in-bottom-h1 {
    -webkit-animation: slide-in-bottom .5s cubic-bezier(.25, .46, .45, .94)
.5s both;
    animation: slide-in-bottom .5s cubic-bezier(.25, .46, .45, .94) .5s
both
}

.slide-in-bottom-subtitle {
    -webkit-animation: slide-in-bottom .5s cubic-bezier(.25, .46, .45, .94)
.75s both;
    animation: slide-in-bottom .5s cubic-bezier(.25, .46, .45, .94) .75s
both
}

.fade-in {
    -webkit-animation: fade-in 1.2s cubic-bezier(.39, .575, .565, 1.000) 1s
both;
}

```

```

    animation: fade-in 1.2s cubic-bezier(.39, .575, .565, 1.000) 1s both
}

.bounce-top-icons {
  -webkit-animation: bounce-top .9s 1s both;
  animation: bounce-top .9s 1s both
}

@-webkit-keyframes slide-in-bottom {
  0% {
    -webkit-transform: translateY(1000px);
    transform: translateY(1000px);
    opacity: 0
  }

  100% {
    -webkit-transform: translateY(0);
    transform: translateY(0);
    opacity: 1
  }
}

@keyframes slide-in-bottom {
  0% {
    -webkit-transform: translateY(1000px);
    transform: translateY(1000px);
    opacity: 0
  }

  100% {
    -webkit-transform: translateY(0);
    transform: translateY(0);
    opacity: 1
  }
}

@-webkit-keyframes bounce-top {
  0% {
    -webkit-transform: translateY(-45px);
    transform: translateY(-45px);
    -webkit-animation-timing-function: ease-in;
    animation-timing-function: ease-in;
    opacity: 1
  }

  24% {
    opacity: 1
  }

  40% {
    -webkit-transform: translateY(-24px);
    transform: translateY(-24px);
    -webkit-animation-timing-function: ease-in;
    animation-timing-function: ease-in
  }

  65% {
    -webkit-transform: translateY(-12px);
    transform: translateY(-12px);
    -webkit-animation-timing-function: ease-in;
    animation-timing-function: ease-in
  }
}

```



```

    }

    82% {
      -webkit-transform: translateY(-6px);
      transform: translateY(-6px);
      -webkit-animation-timing-function: ease-in;
      animation-timing-function: ease-in
    }

    93% {
      -webkit-transform: translateY(-4px);
      transform: translateY(-4px);
      -webkit-animation-timing-function: ease-in;
      animation-timing-function: ease-in
    }

    25%,
    55%,
    75%,
    87% {
      -webkit-transform: translateY(0);
      transform: translateY(0);
      -webkit-animation-timing-function: ease-out;
      animation-timing-function: ease-out
    }

    100% {
      -webkit-transform: translateY(0);
      transform: translateY(0);
      -webkit-animation-timing-function: ease-out;
      animation-timing-function: ease-out;
      opacity: 1
    }
  }
}

@keyframes bounce-top {
  0% {
    -webkit-transform: translateY(-45px);
    transform: translateY(-45px);
    -webkit-animation-timing-function: ease-in;
    animation-timing-function: ease-in;
    opacity: 1
  }

  24% {
    opacity: 1
  }

  40% {
    -webkit-transform: translateY(-24px);
    transform: translateY(-24px);
    -webkit-animation-timing-function: ease-in;
    animation-timing-function: ease-in
  }

  65% {
    -webkit-transform: translateY(-12px);
    transform: translateY(-12px);
    -webkit-animation-timing-function: ease-in;
    animation-timing-function: ease-in
  }
}

```



```

82% {
  -webkit-transform: translateY(-6px);
  transform: translateY(-6px);
  -webkit-animation-timing-function: ease-in;
  animation-timing-function: ease-in
}

93% {
  -webkit-transform: translateY(-4px);
  transform: translateY(-4px);
  -webkit-animation-timing-function: ease-in;
  animation-timing-function: ease-in
}

25%,
55%,
75%,
87% {
  -webkit-transform: translateY(0);
  transform: translateY(0);
  -webkit-animation-timing-function: ease-out;
  animation-timing-function: ease-out
}

100% {
  -webkit-transform: translateY(0);
  transform: translateY(0);
  -webkit-animation-timing-function: ease-out;
  animation-timing-function: ease-out;
  opacity: 1
}
}

@-webkit-keyframes fade-in {
  0% {
    opacity: 0
  }

  100% {
    opacity: 1
  }
}

@keyframes fade-in {
  0% {
    opacity: 0
  }

  100% {
    opacity: 1
  }
}

/*! tailwindcss v2.2.19 | MIT License | https://tailwindcss.com */
/*! modern-normalize v1.1.0 | MIT License |
https://github.com/sindresorhus/modern-normalize */
*,
::after,
::before {
  box-sizing: border-box
}

```

```

}

html {
  -moz-tab-size: 4;
  tab-size: 4
}

html {
  line-height: 1.15;
  -webkit-text-size-adjust: 100%
}

body {
  margin: 0
}

body {
  font-family: system-ui, -apple-system, 'Segoe UI', Roboto, Helvetica,
Arial, sans-serif, 'Apple Color Emoji', 'Segoe UI Emoji';
  background-color: #fffefe;
  background-image: url("background.jpg");
}

.upload {
  appearance: none;
  background-color: #2ea44f;
  border: 1px solid rgba(27, 31, 35, .15);
  border-radius: 6px;
  box-shadow: rgba(27, 31, 35, .1) 0 1px 0;
  box-sizing: border-box;
  color: #fff;
  cursor: pointer;
  display: inline-block;
  font-family: -apple-system,system-ui,"Segoe UI",Helvetica,Arial,sans-
serif,"Apple Color Emoji","Segoe UI Emoji";
  font-size: 14px;
  font-weight: 600;
  line-height: 20px;
  padding: 6px 16px;
  position: relative;
  text-align: center;
  text-decoration: none;
  user-select: none;
  -webkit-user-select: none;
  touch-action: manipulation;
  vertical-align: middle;
  white-space: nowrap;
}

.upload:focus:not(:focus-visible):not(.focus-visible) {
  box-shadow: none;
  outline: none;
}

.upload:hover {
  background-color: #2c974b;
}

.upload:focus {
  box-shadow: rgba(46, 164, 79, .4) 0 0 0 3px;
  outline: none;
}

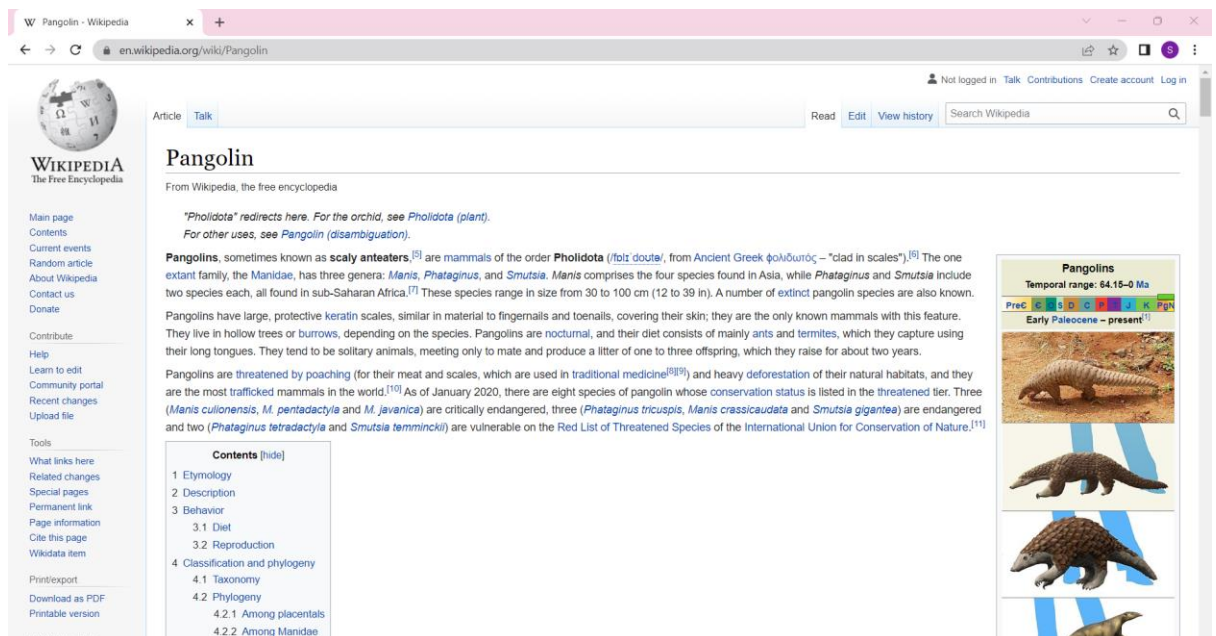
```

```
.upload:disabled {
  background-color: #94d3a2;
  border-color: rgba(27, 31, 35, .1);
  color: rgba(255, 255, 255, .8);
  cursor: default;
}

.upload:active {
  background-color: #298e46;
  box-shadow: rgba(20, 70, 32, .2) 0 1px 0 inset;
}
```

7.3 Feature 3

- Click “Upload” button and wait till the Wikipedia webpage is loaded and then finally the webpage is shown.



app.py

```
from __future__ import division, print_function

import os

import numpy as np
import tensorflow as tf

from flask import Flask, redirect, render_template, request
from keras.applications.inception_v3 import preprocess_input
from keras.models import model_from_json
from werkzeug.utils import secure_filename

global graph
graph=tf.compat.v1.get_default_graph()
#this list is used to log the predictions in the server console
```

```

predictions = ["Corpse Flower",
               "Great Indian Bustard",
               "Lady's slipper orchid",
               "Pangolin",
               "Spoon Billed Sandpiper",
               "Seneca White Deer"
              ]

#this list contains the link to the predicted species
found = [
    "https://en.wikipedia.org/wiki/Amorphophallus_titanum",
    "https://en.wikipedia.org/wiki/Great_Indian_bustard",
    "https://en.wikipedia.org/wiki/Cypripedioideae",
    "https://en.wikipedia.org/wiki/Pangolin",
    "https://en.wikipedia.org/wiki/Spoon-billed_sandpiper",
    "https://en.wikipedia.org/wiki/Seneca_white_deer",
]

app = Flask(__name__, template_folder="Templates")

@app.route('/index')
def pop():
    return render_template('index.html')

@app.route('/', methods=['GET', 'POST'])
def index():
    # Home Page
    return render_template("welcome.html")

@app.route('/predict', methods=['GET', 'POST'])
def upload():
    if request.method == 'GET':
        return ("<h6 style=\"font-face:\"Courier New\";\">No GET request  
herd.....</h6 >")
    if request.method == 'POST':
        # Fetching the uploaded image from the post request using the id
        'uploadedimg'
        f = request.files['uploadedimg']
        basepath = os.path.dirname(__file__)
        #Securing the file by creating a path in local storage
        file_path = os.path.join(basepath, 'uploads',
secure_filename(f.filename))
        #Saving the uploaded image locally
        f.save(file_path)
        #loading the locally saved image
        img = tf.keras.utils.load_img(file_path, target_size=(224, 224))
        #converting the loaded image to image array
        x = tf.keras.utils.img_to_array(img)
        x = preprocess_input(x)
        # Converting the preprocessed image to numpy array
        inp = np.array([x])
        with graph.as_default():
            #loading the saved model from training
            json_file = open('DigitalNaturalist.json', 'r')
            loaded_model_json = json_file.read()
            json_file.close()
            loaded_model = model_from_json(loaded_model_json)
            #adding weights to the trained model
            loaded_model.load_weights("DigitalNaturalist.h5")
            #predicting the image
            preds = np.argmax(loaded_model.predict(inp), axis=1)
            #logs are printed to the console
            print("Predicted the Species " + str(predictions[preds[0]]))
            text = found[preds[0]]

```

```
return redirect(text)

if __name__ == '__main__':
    #Threads enabled so multiple users can request simultaneously
    #debug is turned off, turn on during development to debug the errors
    #application is binded to port 8000
    app.run(threaded = True, debug=True, port="8000")
```

8.TESTING

8.1 Test Cases

				Unit Team ID	TY Nov-22 PNT2022TMDT8534							
				Project Name Maximum Marks	Project - Digital Naturalist - AI Enabled tool for Biodiversity Researchers 4 marks							
Test case ID	Feature Type	Component	Test Scenario	Pre-Requsite	Steps To Execute	Test Data	Expected Result	Actual Result	Status	Comments	TC for Automation(Y/N)	BUG ID Executed By
Webpage_TC_001	UI	Home Page	Verify the user is able to view the page	1. Latest web browser 2. Proper Internet Connection	1. Enter the url of the website and click go 2. Verify the webpage is loading or not	no test data required	The webpage should be visible to the user	The webpage is visible	Pass	The test case passed without any issues	Y	1 Shri Karan S
Webpage_TC_002	UI	Home Page	Verify the page is responsive for all devices	1. Mobile device 2. Desktop device 3. Tablet device 4. Web browser and internet connection	1. Enter the url of the website and click go 2. Verify the webpage is loading properly with proper alignment in all the devices	no test data required	The webpage should be visible to the user	The webpage is visible in all the devices	Pass	The test case passed without any issues	Y	2 Soundhararajan S
Webpage_TC_003	UI	Home Page	Verify the UI elements in upload work	1. Latest web browser 2. Proper Internet Connection	1. Enter the url of the website and click go 2. After the page loaded Successfully click the upload button	Sample Species's images for testing	The webpage should accept the image from the user	The webpage accepts the user input	Pass	The test case passed without any issues	Y	3 Shobanram R
Webpage_TC_004	UI	Home Page	Verify the page is responding for every user action	1. Latest web browser 2. Proper Internet Connection	1. Enter the url of the website and click go 2. Verify the webpage is loading and working properly during the upload and reset	Sample Species's images for testing	The webpage should be stable during the upload and predicting procedure	The webpage is responding stably	Pass	The test case passed without any issues	Y	4 Shankar Subramanyam G

Webpage_TC_005	UI	Home Page	Verify the app accepts only imageformats	1. Latest web browser 2. Proper Internet Connection	1. Enter the url of the website and click go 2. After page loading try to upload non image formats such as pdf, xml, or any audio or video file	non image data	The webpage should reject the user input and prompts the user to upload proper image data for predicting	The webpage prompted with an error message when wrong filetypes uploaded	Pass	The testcase passed without any issues	Y	5	Shri Kamal S
Flask_TC_001	Functional	Flask app	Verify the flask app use the saved model	1. Latest web browser 2. Proper Internet Connection	1. Enter the url of the website and click go 2. Verify the webpage is accepting inputs and predicting according to the category of the animal	Sample Species's images for testing	The webapp should predict the animal properly	The webapp predicts the animal accurately	Pass	The test case passed without any issues, but it requires more dataset to predict the animal accurately	Y	6	Shankar Subrama niyam G
Flask_TC_002	Functional	Flask app	Verify the uploaded image saves on the server	1. Latest web browser 2. Proper Internet Connection 3. Storage in the server for storing the uploaded image	1. Enter the url of the website and click go 2. After page loading try to upload the image and wait	Sample Species's images for testing	The website should accept the image data and save it locally on the server	The app stored the image successfully	Pass	The testcase passed without any issues, But storage will be a issue in future when the storage overflowed	Y	7	Shobanram R
Flask_TC_003	Functional	Flask app	Verify the uploaded image can be retrieved from the storage.	1. Latest web browser 2. Proper Internet Connection 3. Storage in the server where the uploaded image can be retrieved	1. Enter the url of the website and click go 2. Verify the webpage is accepting inputs and predicting according to the category of the animal	Sample Species's images for testing	The web app should be able to store and retrieve the image that is uploaded by the user	The app retrieved the image successfully	Pass	The testcase passed without any issues.	Y	8	Soundharanjan S
Flask_TC_004	Functional	Flask app	Verify the app redirects the user to appropriate species's wikipedia page after predicting	1. Latest web browser 2. Proper internet connection 3. Sample Species's images to test	1. Enter the url of the website and click go 2. Verify the page is redirecting to appropriate specie's webpage	Sample Species's images for testing	The web app should redirect to the appropriate species's wikipedia	The app redirected successfully	Pass	The testcase passed without any issues	Y	9	Soundharanjan S

8.2 User Acceptance Testing

1. Purpose of Document

The purpose of this document is to briefly explain the test coverage and open issues of the Digital Naturalist project at the time of the release to User Acceptance Testing (UAT).

2. Defect Analysis

This report shows the number of resolved or closed bugs at each severity level, and how they were resolved

Resolution	Low Severity	Medium Severity	High Severity	Subtotal of bugs
By UI	1	2	2	5
By Functionality	0	2	2	4
Duplicate	0	4	7	11
External	0	0	0	0
Fixed	1	4	4	9
Not Reproduced	0	0	0	0
Skipped	0	0	0	0
Won't Fix	0	0	0	0
Totals	2	16	15	29

9.RESULTS

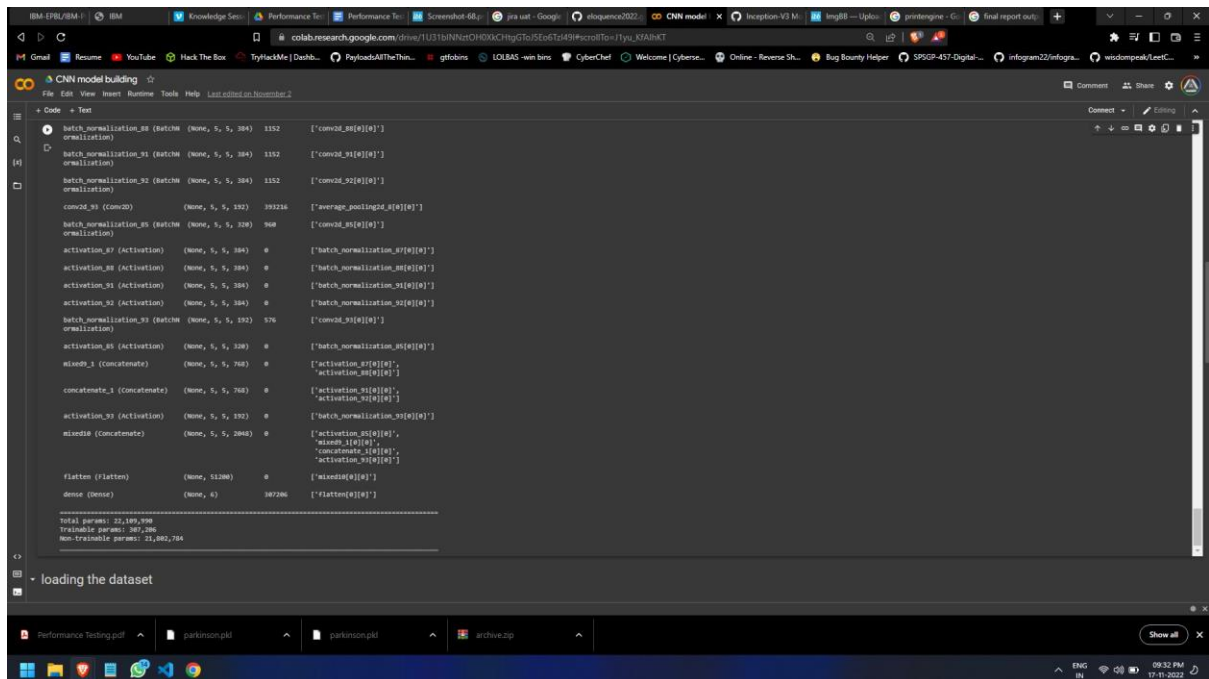
9.1 Performance Metrics

Model Performance Testing:

Project team shall fill the following information in model performance testing template

S.No.	Parameter	Values	Screenshot
1.	Model Summary	Total params: 22,109,990 Trainable params: 307,206 Non-trainable params: 21,802,784	Screenshot 1
2.	Accuracy	Training Accuracy - 92.8% Validation Accuracy - 85.6%	Screenshot 2

Screenshot 1 :



```
batch_normalization_0 (Batch Normalization) (None, 5, 5, 384) 1152 ['conv2d_0[0][4]']
batch_normalization_1 (Batch Normalization) (None, 5, 5, 384) 1152 ['conv2d_1[0][4]']
batch_normalization_2 (Batch Normalization) (None, 5, 5, 384) 1152 ['conv2d_2[0][4]']
conv2d_3 (Conv2D) (None, 5, 5, 32) 39136 ['average_pooling2d_0[0][4]']
batch_normalization_3 (Batch Normalization) (None, 5, 5, 32) 960 ['conv2d_3[0][4]']
activation_4 (Activation) (None, 5, 5, 384) 0 ['batch_normalization_0[0][0]']
activation_5 (Activation) (None, 5, 5, 384) 0 ['batch_normalization_1[0][0]']
activation_6 (Activation) (None, 5, 5, 384) 0 ['batch_normalization_2[0][0]']
activation_7 (Activation) (None, 5, 5, 384) 0 ['batch_normalization_3[0][0]']
batch_normalization_4 (Batch Normalization) (None, 5, 5, 32) 976 ['conv2d_3[0][4]']
activation_8 (Activation) (None, 5, 5, 32) 0 ['batch_normalization_4[0][0]']
mixed_0_1 (Concatenate) (None, 5, 5, 768) 0 ['activation_4[0][0]', 'activation_5[0][0]']
concatenate_1 (Concatenate) (None, 5, 5, 768) 0 ['activation_6[0][0]', 'activation_7[0][0]']
activation_9 (Activation) (None, 5, 5, 32) 0 ['batch_normalization_4[0][0]']
mixed_0 (Concatenate) (None, 5, 5, 2048) 0 ['activation_8[0][0]', 'mixed_0_1[0][0]', 'concatenate_1[0][0]', 'activation_9[0][0]']
flatten (Flatten) (None, 11200) 0 ['mixed_0[0][0]']
dense (Dense) (None, 4) 16720 ['flatten[0][0]']

Total params: 22,109,990
Trainable params: 307,206
Non-trainable params: 21,802,784
```

Screenshot 2 :


```

# Creating a model checkpoint which monitors the accuracy of the model and saves the checkpoint
mc = ModelCheckpoint(filepath = './model.h5',
                    monitor = 'accuracy',
                    verbose = 1,
                    save_best_only = True)

# Creating a earlystopping object which stop training once the model performance stops improving on a hold out validation dataset
es = EarlyStopping(monitor = 'accuracy',
                  min_delta = 0.01,
                  verbose = 1)

call_back = [mc, es]

# Fitting the model
modelHistory = model.fit(traindata, steps_per_epoch=64, epochs = 30, callbacks=call_back)

Epoch 1/30
10/10 [=====] - ETA: 4s - loss: 0.7579 - accuracy: 0.5280WARNING:tensorflow:Your input ran out of data; interrupting training. Make sure that your dataset or generator can generate at least 'steps_per_epoch * epochs' batches (in this case, 1800 batches). You may need to
Epoch 1: accuracy improved from 0.47963 to 0.52799, saving model to ./model.h5
4/10 [=====] - loss: 0.7580/step - loss: 0.7579 - accuracy: 0.5280

# Exporting the model

# Exporting the model to json
model_json = model.to_json()
with open('digitsModel.json', 'w') as json_file:
    json_file.write(model_json)

# Exporting the model weights
model.save_weights('digitsModel.h5')
print("Saved model to disk")

Saved model to disk

# Testing the model

```

10.ADVANTAGES & DISADVANTAGES

Advantages :

- Biodiversity contains a wealth of systematic ecological data that helps us understand the natural world and its origins. Understanding the importance of Biodiversity is achieved through the application.
- Additional information about each and every species is learnt.
- Instant information is available and searching with an image is also very advanced and easy to user or user-friendly.
- The Wikipedia website is like an encyclopedia for knowing details about the specific species that may be bird or flower or mammal.
- Since deep learning is used , the model is highly trained for achieving maximum accuracy with minimal losses.
- Therefore, the prediction will always be correct and can never go wrong.

Disadvantages :

- Our application does not have login or signup page such that the user can't register themselves for the application.
- Since we used flask framework, the need for database is less and also due to absence of login module, database is not used currently.
- Dataset is limited to only 6 species in our application, but in a real-time scenario, the datasets required for training the model will be extravagantly huge.
- Database or API can be used for fetching details/description about each and every species but in our application, we haven't used them.
-

11.CONCLUSION

A web application satisfying the needs of customers within a limited dataset is built successfully with all the code developed and also user acceptance testing is performed.

It is an application for the hikers to find the rare species of birds, flowers , and mammals by giving a picture taken by them.

Not only finding the name of the species , but also displaying all the possible description or the information about in every possible aspect.Species are classified into one of nine Red List Categories: Extinct, Extinct in the Wild, Critically Endangered, Endangered, Vulnerable, Near Threatened, Least Concern, Data Deficient and Not Evaluated.

Vulnerable, Endangered and Critically Endangered species are considered to be threatened with extinction.

These categories can be displayed to the user so that the user can learn the importance of the preserving the species in the realm of biodiversity chain.

Whether the species is dangerous or not, can also be predicted or displayed to the user.

If they are dangerous , the naturalist can probably stay away from them and if there are any poisonous flowers (i.e) plants , they can also take down a note of them and be alert. If there are any medical significances of the plants , and if they contribute to the cure of any disease in the world, the information provided to the naturalists can be helpful.

The unique characteristics of any species can also be noted down by the naturalist while displaying the information about them.

Thus, the web application is run with no hurdles and it predicts all the images that are classified in our dataset.

12.FUTURE SCOPE

The web application can be introduced with login module making the application catering to the needs of a wide variety of users.

Naturalists may need the application to be developed for almost all the discovered species of flora and fauna in the world to be predict with the help of a CNN model.

Database is also required but that can be handled with IBM Cloud.

The information displayed can also be more intense and deep for each and every species covering all the possible facts about them.

The information about medicinal plants can also be used for preparing vaccines and can be use of greater help to humankind.

13.APPENDIX

GITHUB LINK :

<https://github.com/IBM-EPBL/IBM-Project-26473-1660027397>

PROJECT DEMO LINK :

https://drive.google.com/file/d/1Jt7so0l-5t8i3e3oXfi2clr4ZteP0osS/view?usp=share_link