Digital Naturalist - AI Enabled tool for Biodiversity Researchers

A PROJECT REPORT

Submitted by

Kaviya S	822719104027
Karthika S	822719104025
Gomathisankar S	822719104021
Georgebush A	822719104019

Date	19 November 2022
Team ID	PNT2022TMID47219
Project Name	Digital Naturalist - AI Enabled tool for Biodiversity Researchers

TABLE OF CONTENTS

1. INTRODUCTION

1.1 Project Overview

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 forconservation 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 toolfor them to capture, identify and share the beauty to the outside world.

In this project, we are creating a web application which uses a deep learning model, trained on different species of birds, flowers and mammals (2 subclasses in each for a quick understanding) and get theprediction of the bird when an image is been given.

1.2 Purpose

- Augmenting a dataset to virtually increase the size of small datasets in order to make our machinelearning models work better.
- Preprocess the images to a machine-readable format.
- Applying CNN algorithm on the dataset.
- How deep neural networks are predicting the class and subclass of a given image.
- You will be able to know how to find the accuracy of the model.
- You will be able to build web applications using the Flask framework.

2. LITERATURE SURVEY

2.1 Existing problem

Biases in our data arise in part from differences between the aims of the original data collectors (i.e., the photographers) and our aims as biodiversity researchers and ecologists. For example, the spatial distribution of our images was biased toward areas where extensive managed gardens or other displays exhibited large collections of flowering plants. These biases could be addressed by choosing alternative sources, changing the search terms used, or pre-filtering images. Images may also be biased taxonomically or in terms of certain traits, for example, toward species that are typically consideredmore photogenic due tolarge colorful flowers or leaves. Search terms could be modified to either focus on a specific sub-group, e.g., searching using scientific names, or to exclude non-target images, e.g., excluding images that include the words "show" or "garden" in their metadata. Finally, high-level imageclassifiers could be trained to remove images that are clearly not plants, e.g: removing images of animals, paintings. High-level classifiers developed to separate images that contain plants from those that do not, without looking to identify species, could be used to find images worthy of furtherexamination in large datasets that do not have metadata (such as titles and descriptions), removing the need for keyword searches, such as that used in this study.

2.2 References

[1] Aldhebiani AY (2018) Species concept and speciation. Saudi J Biol Sci 25:437–440.

[2] AI naturalists might hold the key to unlocking biodiversity data in social media imagery,

TA August, OL Pescott, A Joly, P Bonnet - Patterns, 2020 - Elsevier.

[3] Digitalization to achieve sustainable development goals: Steps towards a Smart Green Planet, ME Mondejar, R Avtar, HLB Diaz, RK Dubey… - Science of the Total …, 2021 - Elsevier

[4] The real-world use of big data, M Schroeck, R Shockley, J Smart, D Romero-Morales… - IBM Global Business …, 2012

2.3 Problem Statement Definition

- **Main Problem statement (common):**

I. How might we help both experienced and inexperienced user to identify species of plants andanimals and their characteristics with related information?

- **Specific problem statement:**

i) Inexperienced users need to know about poisonous plants and dangerous animals so that they canstay away from it.

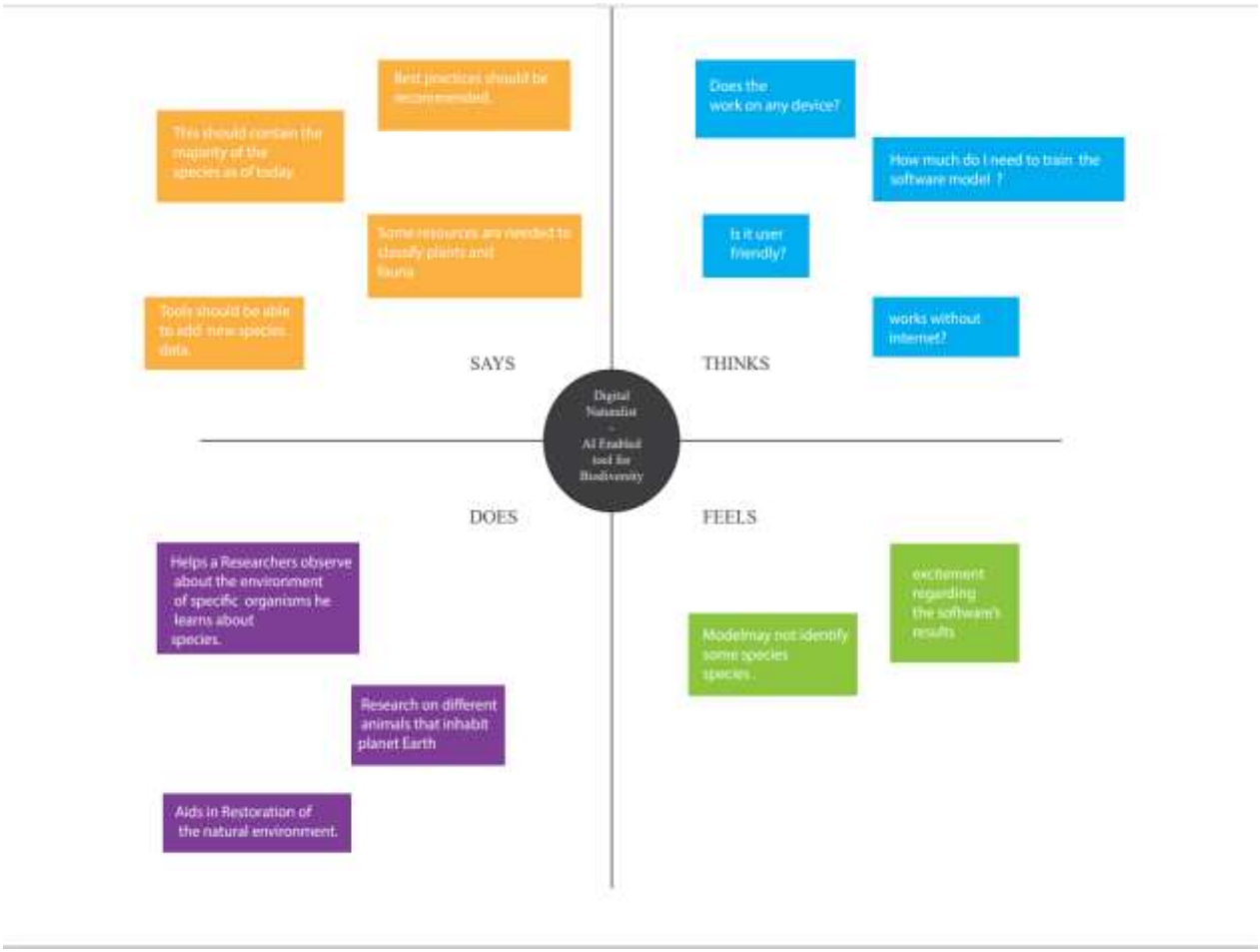
ii) Both experienced and inexperienced users need to know about the medicinal values of a plantbecause they need to use it in case of emergencies.

iii) All the users need to know the types of species of birds, plants and animals so that they can learnabout it in more detailed manner.

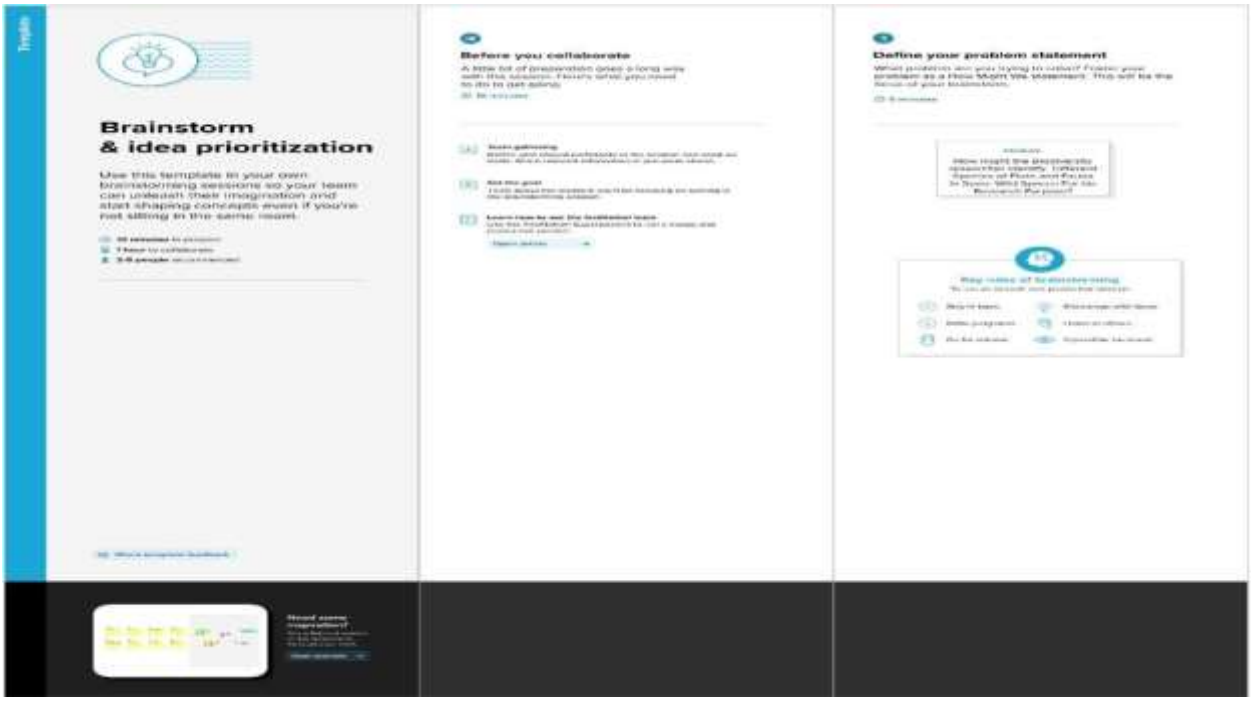
iv) All the users need to know about the rarity of the species of birds, animals or plants so that they canpreserve and save it.

3. IDEATION & PROPOSED SOLUTION

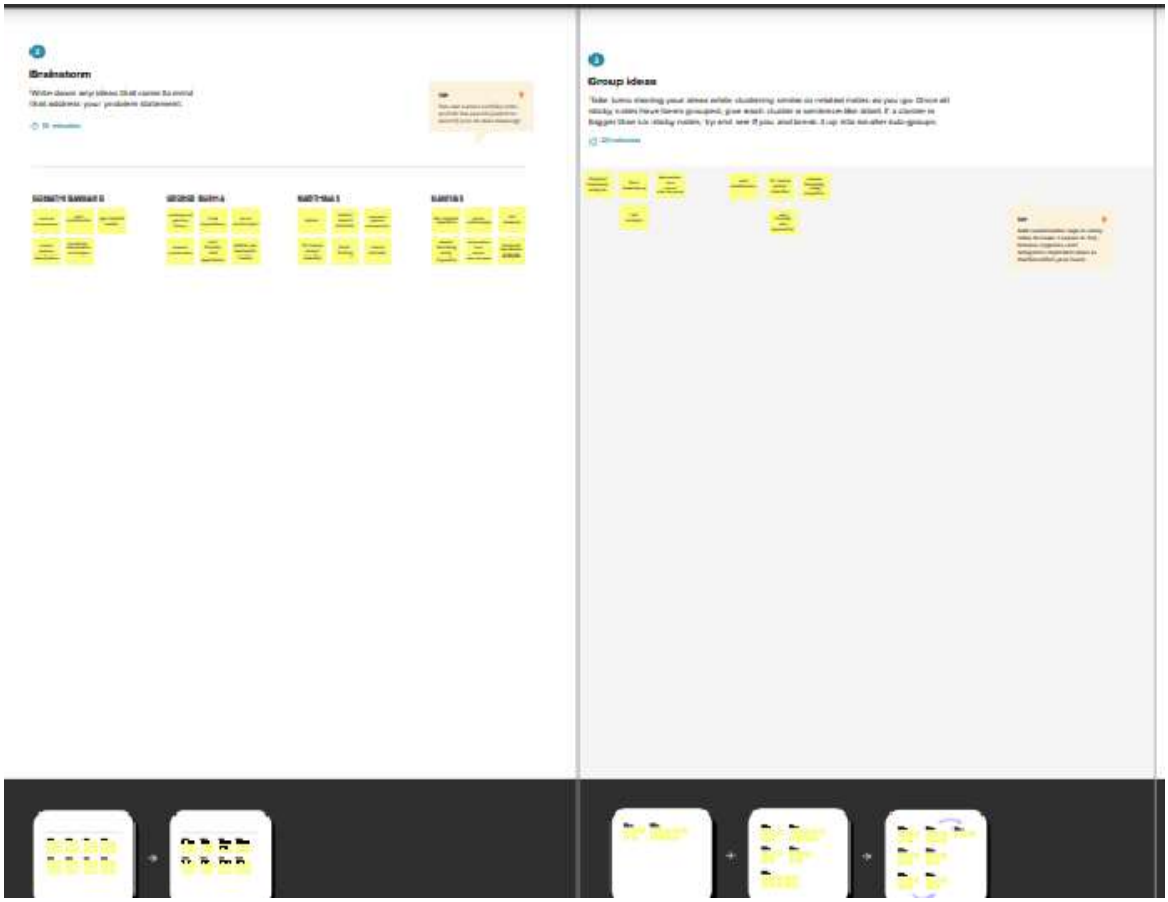
3.1 Empathy Map Canvas



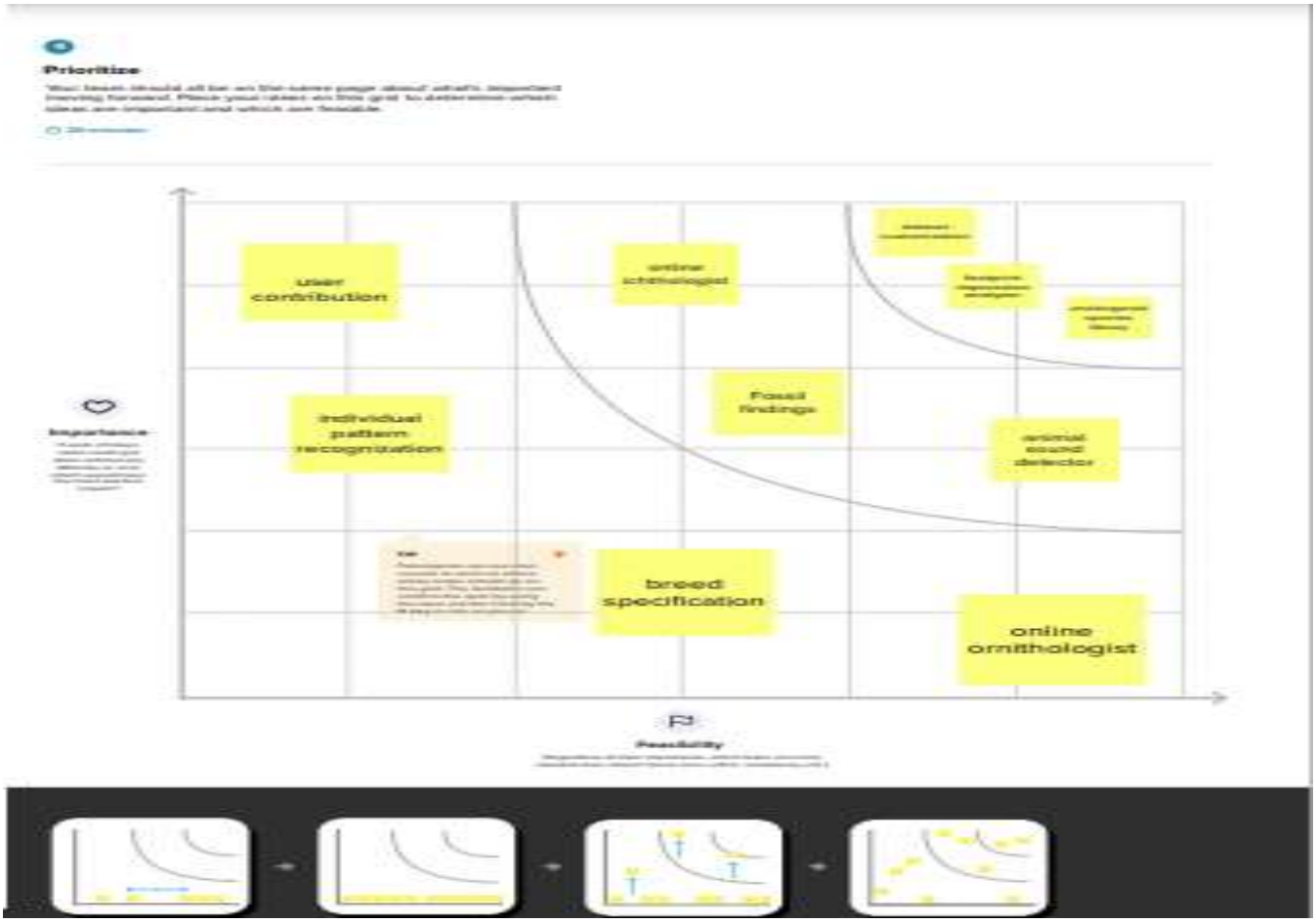
3.2 Ideation & Brainstorming



Step-1: Team Gathering, Collaboration and Select the Problem Statement



Step-2: Brainstorm, Idea Listing and Grouping

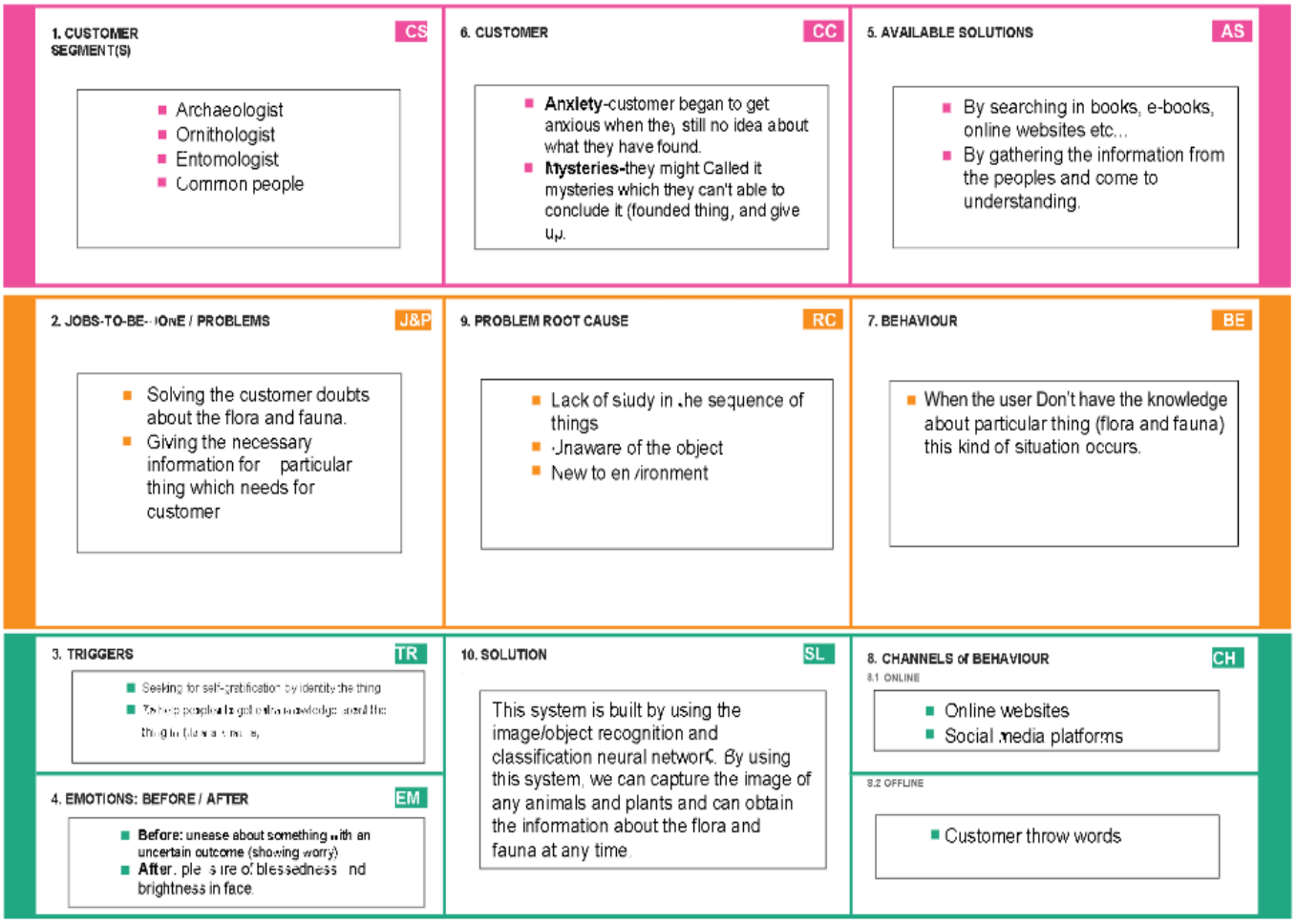


Step-3: Idea Prioritization

3.3 Proposed Solution:

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	Build an efficient AI-based image recognition tool that effectively capture flora and fauna provide information it.
2.	Idea / Solution description	This system is built using image/object recognition and classification using (CNN) convolutional neural networks to identify flora
3.	Novelty / Uniqueness	By using this system, you can capture images of all animals and plants, get information about animals and plants at anytime and anywhere
4.	Social Impact / Customer Satisfaction	The feasibility of this idea is medium as the system must meet the basic requirements of the customer and act as a bridge to more accurate forecasting and analytics. and is neither easy nor difficult. It takes of the captured images as input and provides output in terms of input images
5.	Business Model (Revenue Model)	By using this system, users can predict and analyse pictures of animals and plants without any need to pay for any experts who specialize in this field.
6.	Scalability of the Solution	The Identification database can be extended by training more data as the user capture more data. By this way it can be scaled to any limit.

3.4 Problem Solution fit



4. REQUIREMENT ANALYSIS

(Following are the functional & non-functional requirements of the proposed solution)

4.1 Functional requirement.

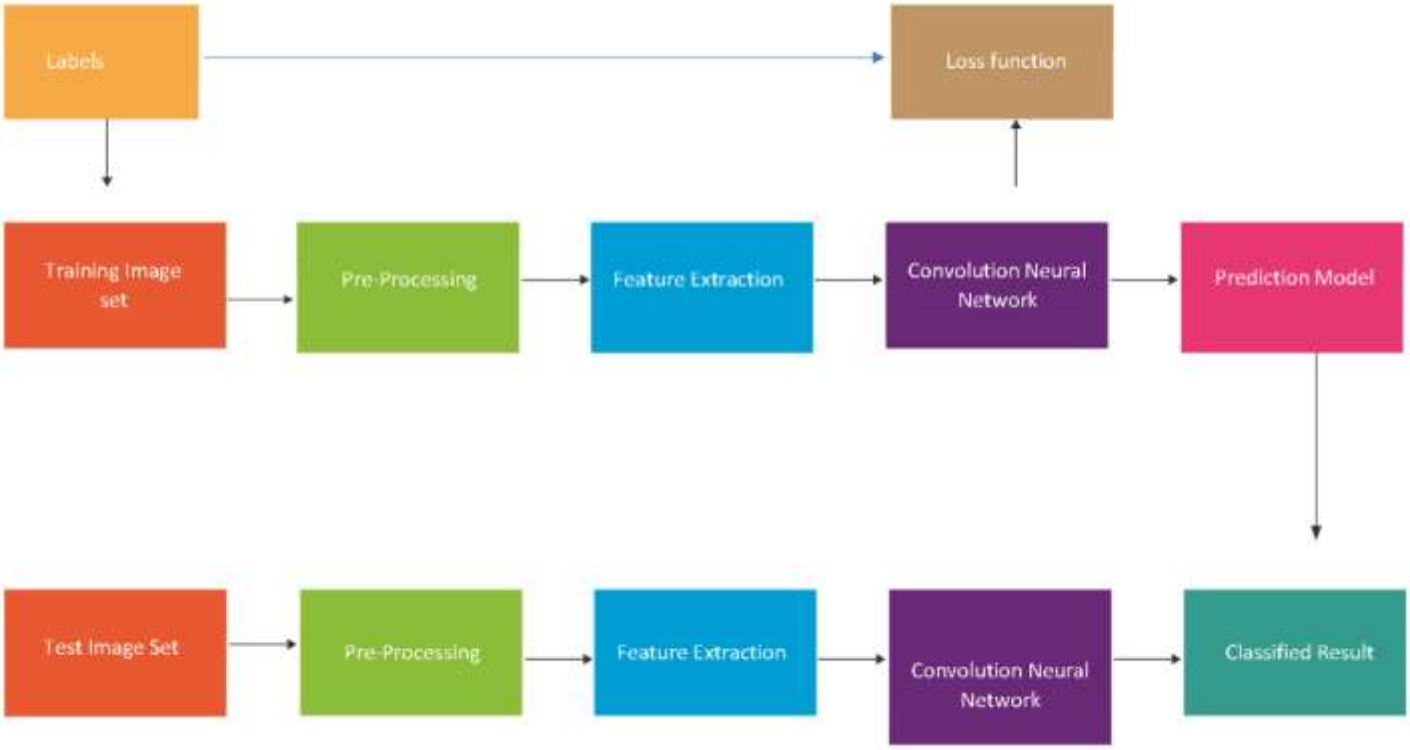
FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	Registration through Form Registration through Gmail Registration through LinkedIN
FR-2	User Confirmation	Confirmation via Email Confirmation via OTP
FR-3	Navigation Service	GPS
FR-4	Database	My SQL, IBM Cloud
FR-5	Premium features	Location sharing, Adding information of new data by User
FR-6	Updating and bug fixing	Updating the application based on user feedback
FR-7	Final Output	Final description of the image (species) captured.
FR-8	Alerts	System should alert about dangerous plants and animals

4.2 Non-functional Requirements.

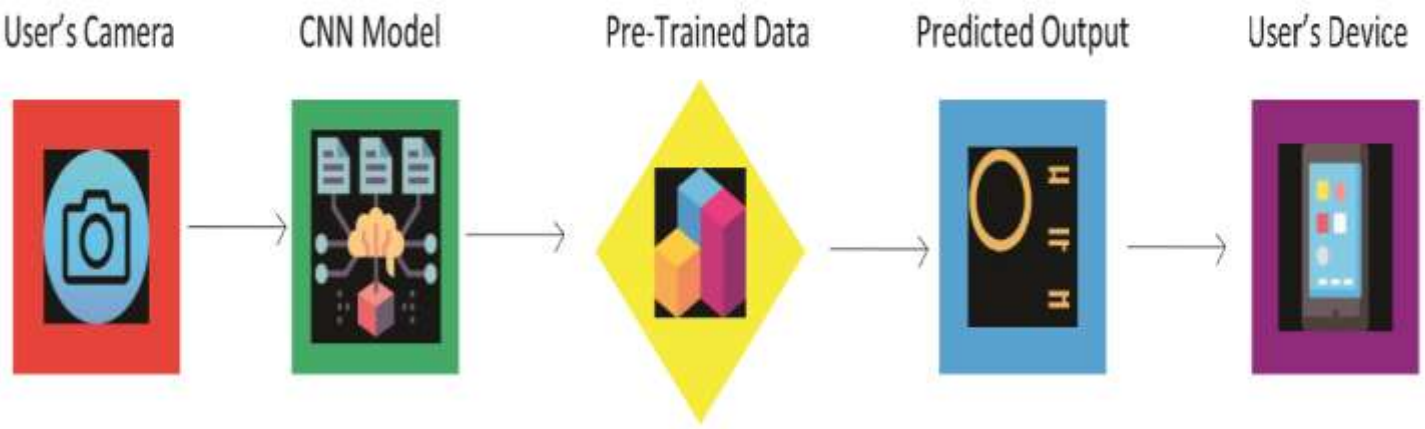
FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	The users must be able to use the application without any problems and difficulties. The app is easy to view and does not strain the eyes. All information are in simple terms. The error rate of the final output must not be more than 20%
NFR-2	**Security**	SHA-256, Encryptions, AES etc.
NFR-3	**Reliability**	The system must perform without failure in 80 percent of the time.
NFR-4	**Performance**	Under normal load, the system must show the results within 15 seconds, and under maximum it can take up to load 30 seconds
NFR-5	**Availability**	The application will be available 99 % of the time in a month.
NFR-6	**Scalability**	The system must be able to support 10,000 users while using it. As the usage and user base of this application grows, more features can be added like languages based on the geographical usage, premium or subscription model, etc.

5. PROJECT DESIGN

5.1 Data Flow Diagrams



Solution & Technical Architecture



5.2 User Stories

Use the below template to list all the user stories for the product.

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 plantlife, animals and birds	I can take photos whenrequired	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 shareoption	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 nameof the species	I must see the correctdata	High
		USN-6	As a user, I can see the characteristicsand alert messages	I must see the correctdata	High
Administrator	Manage	USN-1	As a admin I must add various data andedit information	I must edit the datapresent	High

6. PROJECT PLANNING & SCHEDULING

6.1 Sprint Planning & Estimation

Use the below template to create product backlog and sprint schedule

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Registration	USN-1	As a biogeography, I can register for the application by entering my Email, Password, and confirming my password	2	High	Kaviya S Karthika S Georgebush A Gomathisankar s
Sprint-	User	USN-2	As a biogeography, I will receive confirmation email once I have registered for the application	1	Medium	Kaviya S Karthika S Georgebush A Gomathisankar s
Sprint-1	Login	USN-3	As an biogeography, I can log into the application by entering email & password	2	High	Kaviya S Karthika S Georgebush A Gomathisankar s
Sprint-2	Data Collection	USN-1	Download the dataset used in Digital Naturalist – AI Enabled tools for Biodiversity Researchers	2	Medium	Kaviya S Karthika S Georgebush A Gomathisankar s
Sprint-	Image	USN-1	Improving the image data that	1	High	Kaviya S Karthika S Georgebush A Gomathisankar S
Sprint-	Getting	USN-1	Neural network are integral for high level of accuracy based on		Medium	Kaviya S Karthika S Georgebush A Gomathisankar S

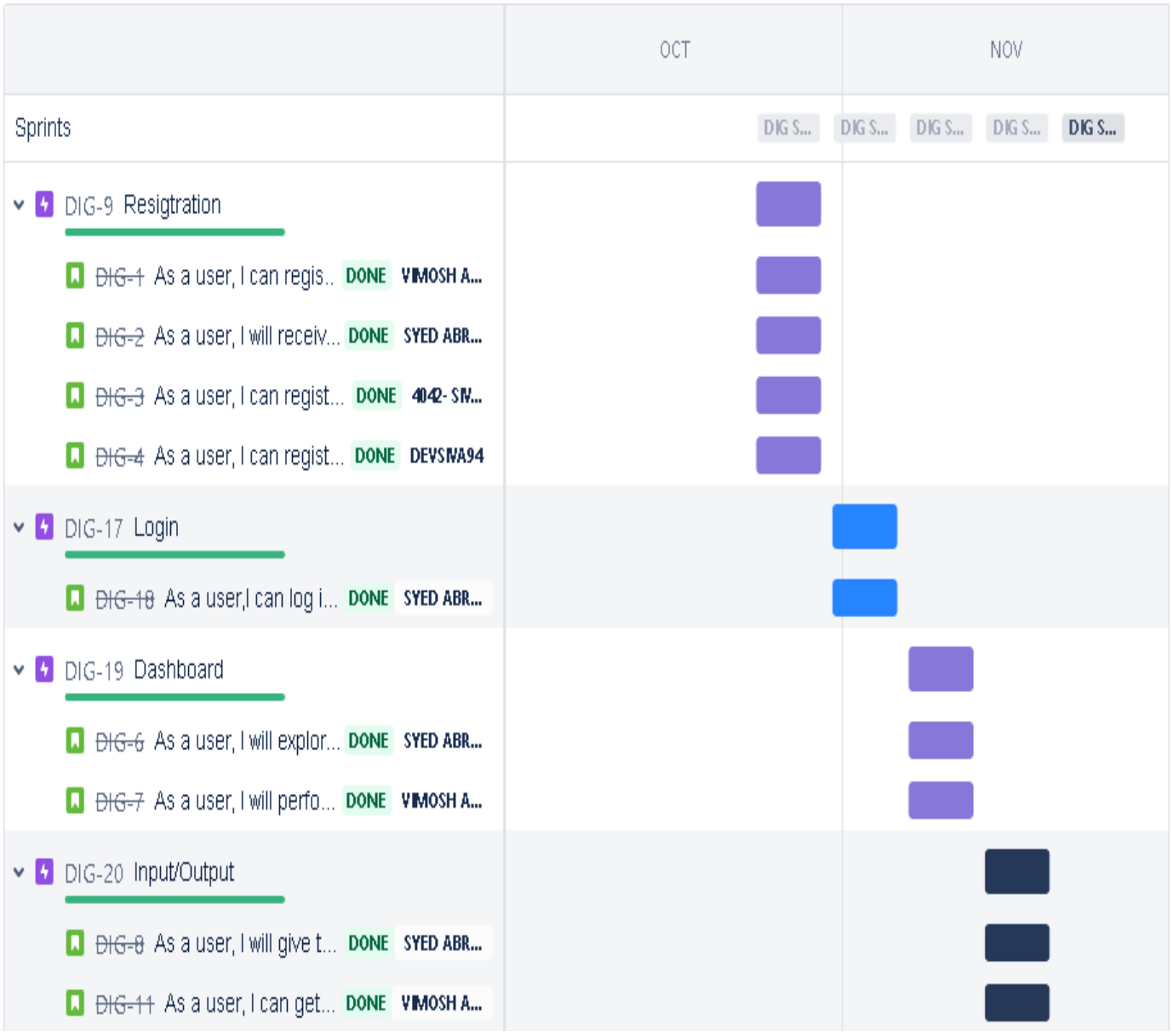
			data inputs.			
Sprint-	Evaluation and	USN-1	well a model behaves after each	1	Medium	Kaviya S Karthika S Georgebush A Gomathisankar S
Sprint-	Application	USN-2	After the model is built, we will be	1	High	Kaviya S Karthika S Georgebush A Gomathisankar S
Sprint-4	Train the Model on IBM	USN-2	Build Deep learning model and computer vision Using the IBM cloud.	2		Kaviya S Karthika S Georgebush A Gomathisankar S

6.2 Sprint Delivery Schedule

Project Tracker, Velocity & Burndown Chart:

Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned EndDate)	Sprint Release Date (Actual)
Sprint-1	20	4 Days	24 Oct 2022	27 Oct 2022	20	29 Oct 2022
Sprint-2	20	5 Days	28 Oct 2022	01 Nov 2022	20	04 Nov 2022
Sprint-3	20	7 Days	02 Nov 2022	09 Nov 2022	20	11 Nov 2022
Sprint-4	20	9 Days	10 Nov 2022	18 Nov 2022	20	19 Nov 2022

6.3 Reports from JIRA



7. CODING & SOLUTIONING

7.1 Feature 1:

- Display Botanical names
- Display alert messages for plants/animals using different colors
- small description about them
- Rarities of the species

7.2 Feature 2:

- What disease does the plant cure
- Providing alerts based on if a species is harmful or not

8.TESTING

8.1. Test Cases:

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).
   65
2. **Defect Analysis**
   This report shows the number of resolved or closed bugs at each severity level, and they were resolved

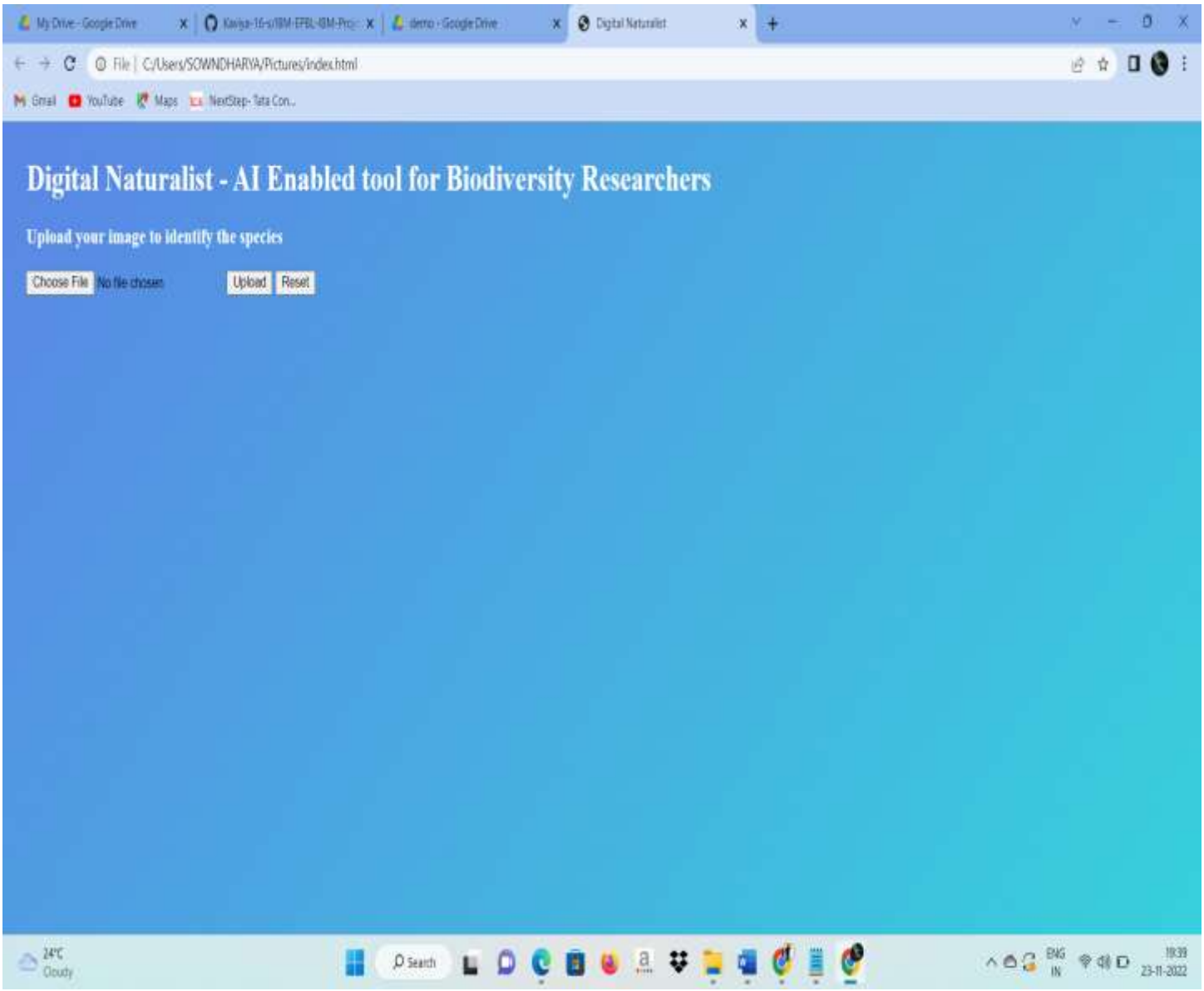
Resolution	**Severity 1**	**Severity 2**	**Severity 3**	**Severity 4**	**Severity 5**
By Design	2	0	0	0	2
Duplicate	1	0	0	0	1
External	0	1	0	0	1
Fixed	8	0	4	1	13
Not Reproduced	0	0	1	0	1
Skipped	1	0	0	0	1
Won't Fix	0	0	1	0	1
Totals	12	1	6	1	20

8.2. Acceptance Testing:

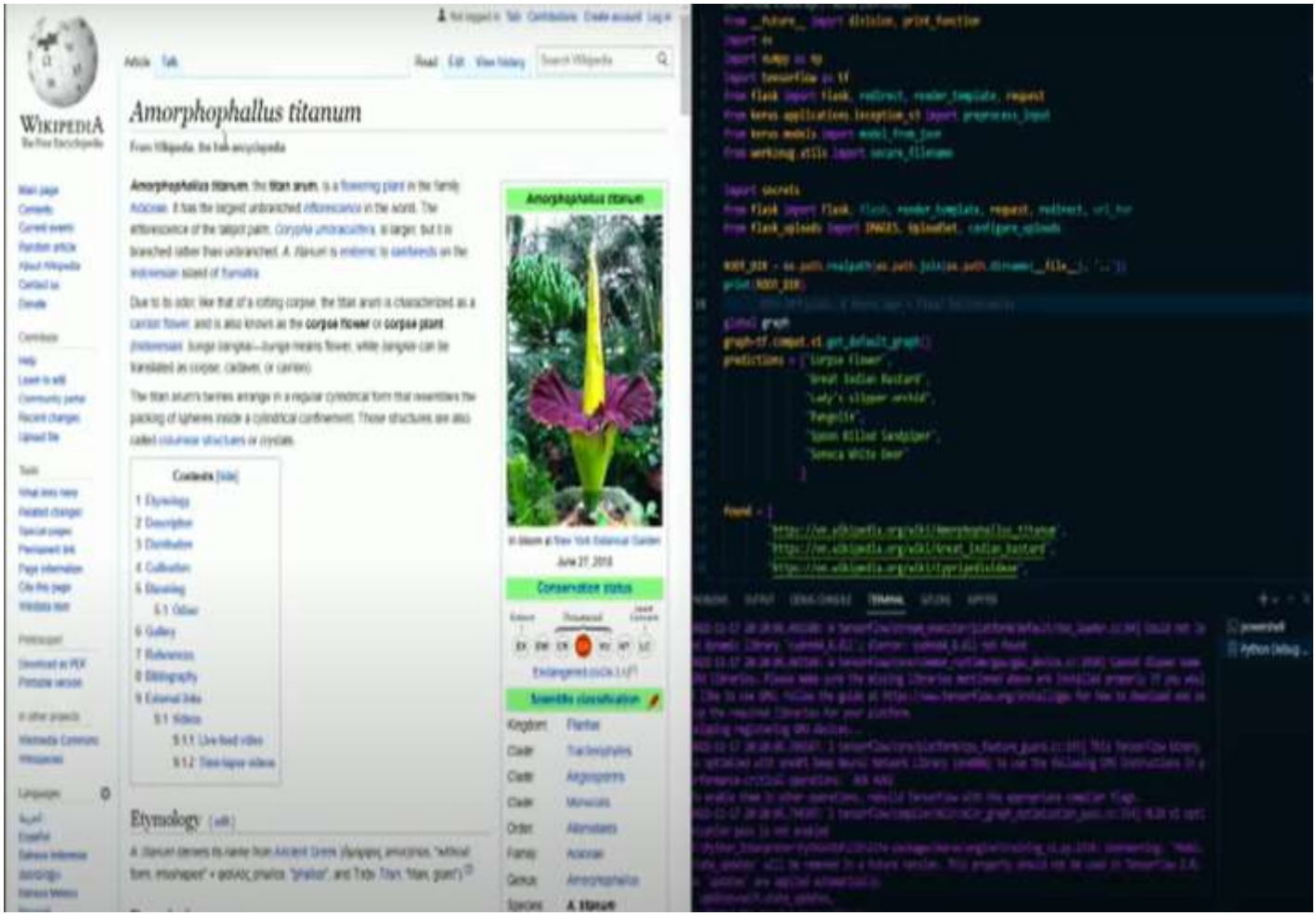
Section	Test cases	Not Tested	Fail	Pass
Print Engine	2	0	0	2
Client Application	1	0	0	1
Security	3	0	0	3
Final Report Output	4	0	0	4
Version Control	2	0	0	2

9.RESULTS

9.1 index.html



Output



10.ADVANTAGES & DISADVANTAGES

Advantages:

- It helps field biologists build their own experimental tools.
- It helps designers explore new interactions with nature.
- It paved the ways to discover new ways of experiencing the natural world.
- It helps to create sustainable world by saving the endangered species.
- Digital Naturalism unites biologists, designers, engineers, and artists to build and analyze new devices.
- User can get detailed description of any kind of species.
- It is a handy application for a person who is travel freak.

Disadvantages:

- Proper network should be maintained to avoid hitches.
- Difficult to classify sub-classes of same species.

11.CONCLUSION

Assessment of regional biodiversity based on global scientific consensus is a scientific basis for the whole society and a tool for local to international discussion and decision making. In the new era of extinction, people would understand the value of (intrinsic or otherwise) of our state's threatened biodiversity. Fewer observations, data points and discoveries would be made of the natural world to help us measure our impacts on it. It provides the opportunity to build a positive identity with science or recognize the value of more holistic ways of thinking such as traditional ecological knowledge. It helps us to engage in environmental stewardship behaviours ranging from resource conservation to building resilience among vulnerable communities. By acknowledging our origins in evolution, the naturalist perspective also enhances our feeling of kinship with the other species with which we share this planet, and our desire to sustain and nurture the planet itself. All sentient beings, including humanity, owe their existence to conditions that extend far beyond us in space and time. The model which was used for the detection of digital naturalists using the species images from the wild life and the species with flora part and with fauna part will be displayed as well. From the resultant graphs, it is proven that the accuracy of the model has reached good level. If it is deployed in the real-time scenario then it will help many people in distinguishing between both without wasting the money on various machines. If the image is confirmed by the model, then the person can know the feature of the species. It can be the best way of practice for people to save money. As we know that the data plays a crucial role in every deep learning model, if the data is more specific and accurate about the species then that can help in reaching greater accuracy with better results in real- time applications.

11.FUTURE SCOPE

- AI image classifiers can create biodiversity datasets from social media imagery
- Flickr hosts many images of plants; some can be accurately classified to speciesby AI
- Images are spatially aggregated around tourist sites and under-represent native species
- Images focused on a single, non-horticultural, plant are most reliably

13.APPENDIX

Source Code

<---------- # **trainmodel.py** ---------- >

```
import os, glob, random
from PIL import Image
import numpy as np
import keras.backend as K
from keras.optimizers import Adam
from keras.applications import *
from keras.preprocessing import *
from keras.callbacks import EarlyStopping
from keras.models import Sequential
from keras.layers import Dense, Conv2D, MaxPool2D, Flatten,Dropout
from keras.utils.np_utils import to_categorical
from sklearn.model_selection import train_test_split



ROOT_DIR = os.path.realpath(os.path.join(os.path.dirname(__file__), '..'))
print(ROOT_DIR)


def generateListofFiles(dirName):
    listOfFile = os.listdir(dirName)
    allFiles = list()
    for fol_name in listOfFile:
        fullPath = os.path.join(dirName, fol_name)
```

```
        allFiles.append(fullPath)
    return allFiles

def Configure_CNN_Model(output_size):
    K.clear_session()
    model = Sequential()
    model.add(Dropout(0.4,input_shape=(224, 224, 3)))
    model.add(Conv2D(256, (5, 5),input_shape=(224, 224, 3),activation='relu'))
    model.add(MaxPool2D(pool_size=(2, 2)))
    model.add(Conv2D(128, (3, 3), activation='relu'))
    model.add(MaxPool2D(pool_size=(2, 2)))
    model.add(Conv2D(64, (3, 3), activation='relu'))
    model.add(MaxPool2D(pool_size=(2, 2)))
    model.add(Flatten())
    model.add(Dense(512, activation='relu'))
    model.add(Dropout(0.3))
    model.add(Dense(256, activation='relu'))
    model.add(Dropout(0.3))
    model.add(Dense(128, activation='relu'))
    model.add(Dropout(0.3))
    model.add(Dense(output_size, activation='softmax'))
    return model

def PrepreocessData(subfolders):
    X_data,Y_data,found = [],[],[]
    id_no=0
    for paths in subfolders:
        files = glob.glob (paths + "/*.jpg")
        found.append((paths.split('\\')[-2],paths.split('\\')[-1]))


        for myFile in files:
            img = Image.open(myFile)
            img = img.resize((224,224), Image.ANTIALIAS)
            img = np.array(img)
            if img.shape == ( 224, 224, 3):
                X_data.append (img)
                Y_data.append (id_no)
        id_no+=1


    X = np.array(X_data)
    Y = np.array(Y_data)
```

```
    print("x-shape",X.shape,"y shape", Y.shape)

    X = X.astype('float32')/255.0
    y_cat = to_categorical(Y_data, len(subfolders))

    print("X shape",X,"y_cat shape", y_cat)
    print("X shape",X.shape,"y_cat shape", y_cat.shape)

    return X_data,Y_data,X,y_cat,found;
   def splitData():
    X_train, X_test, y_train, y_test = train_test_split(X, y_cat, test_size=0.2)
    print("The model has " + str(len(X_train)) + " inputs")
    return X_train, X_test, y_train, y_test



augumented_path = ROOT_DIR+"\\Final Deliverables\\Augumented Dataset\\"
Folders = generateListofFiles(augumented_path)
subfolders = []
for num in range(len(Folders)):
    sub_fols = generateListofFiles(Folders[num])
    subfolders+=sub_fols

X_data,Y_data,X,y_cat,found= PrepreocessData(subfolders)
X_train, X_test, y_train, y_test = splitData()



early_stop_loss = EarlyStopping(monitor='loss', patience=3, verbose=1)
early_stop_val_acc = EarlyStopping(monitor='val_accuracy', patience=3, verbose=1)
model_callbacks=[early_stop_loss, early_stop_val_acc]

model = Configure_CNN_Model(6)
model.compile(loss='categorical_crossentropy',optimizer=Adam(lr=0.001),metrics=['accuracy'])
weights = model.get_weights()
model.set_weights(weights)



image_number = random.randint(0,len(X_test))
predictions = model.predict([X_test[image_number].reshape(1, 224,224,3)])

for idx, result, x in zip(range(0,6), found, predictions[0]):
  print("Label: {}, Type : {}, Species : {} , Score : {}%".format(idx, result[0],result[1],
round(x*100,3)))
```

```
ClassIndex=np.argmax(model.predict([X_test[image_number].reshape(1, 224,224,3)]),axis=1)
print(found[ClassIndex[0]])


model_json = model.to_json()
with open(ROOT_DIR+"\\Final Deliverables\\DigitalNaturalist.json", "w") as json_file:
    json_file.write(model_json)
model.save_weights(ROOT_DIR+"\\Final Deliverables\\DigitalNaturalist.h5")
```

<--- **# 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

                import secrets
                from flask import Flask, flash, render_template, request, redirect, url_for
                from flask_uploads import IMAGES, UploadSet, configure_uploads

                ROOT_DIR = os.path.realpath(os.path.join(os.path.dirname(__file__), '..'))
                print(ROOT_DIR)

                global graph
                graph=tf.compat.v1.get_default_graph()
                predictions = ["Corpse Flower",
                           "Great Indian Bustard",
                           "Lady's slipper orchid",
                           "Pangolin",
                           "Spoon Billed Sandpiper",
                           "Seneca White Deer"
                          ]

                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__)
photos = UploadSet("photos", IMAGES)
app.config["UPLOADED_PHOTOS_DEST"] = ROOT_DIR+"\\Final
Deliverables\\Uploads\\"
app.config["SECRET_KEY"] = str(secrets.SystemRandom().getrandbits(128))
configure_uploads(app, photos)

@app.route('/', methods=['GET'])

def index():
    return render_template("index.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':

        photos.save(request.files["upload-btn"])
        f = request.files['upload-btn']
        img = tf.keras.utils.load_img(ROOT_DIR+"\\Final
Deliverables\\Uploads\\"+secure_filename(f.filename), target_size=(224, 224))
        x = tf.keras.utils.img_to_array(img)
        x = preprocess_input(x)
        inp = np.array([x])
        with graph.as_default():
            json_file = open('Final Deliverables\\DigitalNaturalist.json', 'r')
            loaded_model_json = json_file.read()
            json_file.close()
            loaded_model = model_from_json(loaded_model_json)
            loaded_model.load_weights("Final Deliverables\\DigitalNaturalist.h5")
            preds =  np.argmax(loaded_model.predict(inp),axis=1)
            print("Predicted the Species " + str(predictions[preds[0]]))
        text = found[preds[0]]
        return redirect(text)
```

```
if __name__ == '__main__':
    app.run(threaded = True,debug=True,port="8000")
```

GitHub & Project Demo Link

https://drive.google.com/file/d/1kZDDJ4_nZIv6lFilFAHmQUXGOMjj2LFi/view?usp=share_link

GitHub Link :

https://github.com/Kaviya-16-s/IBM-EPBL-IBM-Project-51140-1660973626

REFERENCES

[1] Aldhebiani AY (2018) Species concept and speciation. Saudi J Biol Sci 25:437–440.

[2] AI naturalists might hold the key to unlocking biodiversity data in social media imagery,

TA August, OL Pescott, A Joly, P Bonnet - Patterns, 2020 - Elsevier.

[3] Digitalization to achieve sustainable development goals: Steps towards a Smart Green Planet, ME

Mondejar, R Avtar, HLB Diaz, RK Dubey… - Science of the Total …, 2021 - Elsevier

[4] The real-world use of big data, M Schroeck, R Shockley, J Smart, D Romero-Morales…