Bird Species Detection

A Major project Report submitted
in partial fulfillment of the requirements
for the award of the degree of

BACHELOR OF TECHNOLOGY

In

COMPUTER SCIENCE & ENGINEERING

By

1. Y. Sandhya	–	**18B01A0589**
2. M. Keerthi Priya	–	**18B01A0596**
3. K. Naga Priyanka	–	**18B01A05A4**
4. K.L.Prasanna Bhavani	–	**19B05A0512**

Under the esteemed guidance of
Mrs. P.R. Sudha Rani M.Tech(Ph.D.)
Associate Professor

VISHNU
UNIVERSAL LEARNING

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
SHRI VISHNU ENGINEERING COLLEGE FOR WOMEN(A)
(Approved by AICTE, Accredited by NBA & NAAC, Affiliated to JNTU Kakinada)
BHIMAVARAM – 534 202

2021 – 2022

SHRI VISHNU ENGINEERING COLLEGE FOR WOMEN(A)

(Approved by AICTE, Accredited by NBA & NAAC, Affiliated to JNTU Kakinada)
BHIMAVARAM – 534202

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

CERTIFICATE

*This is to certify that the Major Project entitled "**Bird Species Detection**", is being submitted by **Y.Sandhya**, **M.Keerthi Priya, K.Naga Priyanka, K.L.Prasanna Bhavani** bearing the **Regd. No. 18B01A0589, 18B01A0596, 18B01A05A4, 19B05A0512** in partial fulfillment of the requirements for the award of the degree of "**Bachelor of Technology** in **Computer Science & Engineering**" is a record of bonafide work carried out by them under my guidance and supervision during the academic year* **2021 – 2022** *and it has been found worthy of acceptance according to the requirements of the university.*

Internal Guide **Head of the Department** **External Examiner**

Acknowledgements

The satisfaction that accompanies the successful completion of any task would be incomplete without the mention of the people who made it possible and whose constant encouragement and guidance has been a source of inspiration throughout the course of this project. We take this opportunity to express our gratitude to all those who have helped us in this project.

We wish to place our deep sense of gratitude to **Sri. K. V. Vishnu Raju, Chairman of SVES**, for his constant support on each and every progressive work of ours.

We wish to express our sincere thanks to **Dr. G. Srinivasa Rao, Principal of SVECW** for being a source of an inspirational constant encouragement.

We wish to express our sincere thanks to **Dr. P. Srinivasa Raju, Vice-Principal of SVECW** for being a source of an inspirational constant encouragement.

We wish to place our deep sense of gratitude to **Dr. P. Kiran Sree, Head of the Department of Computer Science & Engineering**, for his valuable advices in completing this project successfully.

Our deep sense of gratitude and sincere thanks to **Mrs. P.R. Sudha Rani** **M.Tech(Ph.D.)** **, Associate Professor** for her unflinching devotion and valuable suggestion throughout our project.

Project Associates

Y. Sandhya	–	**18B01A0589**
M. Keerthi Priya	–	**18B01A0596**
K. Naga Priyanka	–	**18B01A05A4**
K.L.Prasanna Bhavani	–	**19B05A0512**

ABSTRACT

In our world, there are above 9000 bird species. Some bird species are being found rarely and if found also prediction becomes very difficult. In order to overcome this problem, we have an effective and simple way to recognize these bird species based on their features. Also, the human ability to recognize the birds through the images is more understandable than audio recognition.

BIRD behavior and population trends have become an important issue now a days. Birds help us to detect other organisms in the environment easily as they respond quickly to the environmental changes. But, gathering and collecting information about birds requires huge human effort as well as becomes a very costlier method. In such case, a reliable system that will provide large scale processing of information about birds and will serve as a valuable tool for researchers, governmental agencies, etc. is required. So, bird species identification plays an important role in identifying that a particular image of bird belongs to which species. Bird species identification means predicting the bird species belongs to which category by using an image.

The identification can be done through image, audio or video. An audio processing technique makes it possible to identify by capturing the audio signal of birds. But, due to the mixed sounds in environment such as insects, objects from real world, etc. processing of such information becomes more complicated. Usually, human beings find images more effective than audios or videos. So, an approach to classify bird using an image over audio or video is preferred. Bird species identification is a challenging task to humans as well as to computational algorithms that carries out such a task in an automatic fashion.

CONTENTS

1.INTRODUCTION

"Bird species Detection" means predicting the bird species belongs to which category by using an image. Identifying a bird can be a challenge, even for experienced birders. And if we are new to using field guides, it can be difficult to figure out where to even begin searching in the hundreds of pages of species. By some features like size, shape and colour birds can be classified.

These birds can be classified in two ways:

1. bird specie detection through voice
2. Bird specie detection using image

"**Bird specie detection through voice**" means that the bird is classified based on the sound we give as an user input. The voice signals of the sound is being classified and based on that the bird is being predicted. Birds interchange a variety of information through each vocal expression. Through the so-called calls, which we can hear more often, birds can transmit various warnings about a danger approaching, identify individuals in a flock etc. So basically we cannot predict a bird when it produce different sound since same bird can produce different sounds based on the situation it was present.

Disadvantages of bird classification through voice:

- Background noise- especially while using data recorded in a city.
- Multi-label classification problem-when there are many species singing at the same time.
- Different types of bird songs.
- Inter-species variance-there might be difference in bird song between the same species living in different regions or countries.
- Data set issues-the data can be highly imbalanced due to bigger popularity of one species over another, there is a large number of different species and recordings can have different length, quality of recordings.

- In this modern world , due to the heavy radiation many birds are disappearing. There are many bird conservation organisations which maintains a huge set of data about birds.
- Although bird classification can be done manually by domain experts, with growing amounts of data, this rapidly becomes a tedious and time-consuming process.
- So, by this Project **"Bird Specie Detection using Image"** we can identify the species of the birds accurately and in less time. Here in our project we give the bird image as input and we get the name of the Bird as Output.

Advantages of bird specie detection using image:

- Our project helps to identify the species of the birds by analysing an image.
- Saves a lot of time in predicting a bird.
- Our project maintains a huge dataset about birds which helps in finding the birds which are disappearing.

2. SYSTEM ANALYSIS

2.1 Existing System

- To identify the bird species there are many websites produces the results using different technologies. For suppose if we will give an input in those websites and android applications it gives us multiple results instead of single bird name.

- In Existing System, this Project is implemented using **KNN(K-Nearest Neighbour)** Algorithm.

DRAWBACKS OF EXISTING SYSTEM

1. In Existing System, there are less number of bird species.
2. Noisy Images are also present.

2.2 Proposed System

- So, our aim is to develop a project to produce better and accurate results. Convolution neural network algorithm is a multilayer perceptron that is the special design for the identification of two-dimensional image information.

- It has four layers: an input layer, a convolution layer, a sample layer, and an output layer. In a deep network architecture, the convolution layer and sample layer may have multiple. so, we have used Convolutional Neural Networks to classify the bird species.

- Also we will made a web application where we can upload any image of the bird.

ADVANTAGES

1. Easy to detect the type of species by using the image of bird.
2. Noisy Images are Eliminated.

2.3 Feasibility Study

Generally, the feasibility study is used for determining the resource cost, benefits and whether the proposed system is feasible with respect to the organization. The proposed system feasibility could be as follows. There are few types of feasibility which are equally important are:

- Technical Feasibility
- Economic Feasibility
- Behavioral Feasibility
- Operational Feasibility
- Social Feasibility

Technical Feasibility

Technical feasibility is the formal process of assessing whether it is technically possible to manufacture a product or service. Before launching a new offering or taking up a client project, it is essential to plan and prepare for every step of the operation. Technical feasibility helps determine the efficacy of the proposed plan by analyzing the process, including tools, technology, material, labor and logistics.

Technical feasibility deals with the existing technology, software and hardware requirements for the proposed system. The proposed system "BIRD SPECIE DETECTION" is planned to run on streamlit , machine learning technology. Thus, the project is considered technically feasible for the development. The work for the project can be done with current equipment, existing software technology and available personnel. Hence the proposed system is technically feasible.

Purpose of Technical Feasibility:

• To understand about the development of the product with the available technology.

• To have an idea that it is done with the trending technologies and whether it fulfilled all the features of the project.

Economic Feasibility

An evaluation of economic feasibility must include reliable estimates of the economic benefits and costs of the project. If the benefits generated by our project exceed project costs, then the project is considered to be economically feasible.

This method is most frequently used for evaluating the effectiveness of a python. It is also called as benefit analysis. In this project "BIRD SPECIE DETECTION" is developed on current equipment, existing software technology. Since the required hardware and software for developing the system is already available in the organization, it does not cost much developing the proposed system.

Behavioral Feasibility

It evaluates and estimates the user attitude or behavior towards the development of new system. It helps in determining if the system requires special effort to educate, retrain, transfer, and changes in employee's job status on new ways of conducting business. This proposed system "BIRD SPECIE DETECTION" has much behavioral feasibility because users are provided with a better facility.

Operational Feasibility

Operational feasibility is the measure of how well a proposed system solves the problems, and takes advantage of the opportunities identified during scope definition and how it satisfies the requirements identified in the requirements analysis phase of system development.

Social Feasibility

The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessary. The level of acceptance by the users solely depends on the methods that are employed to educate the user about the system and to make him familiar with it. His level of confidence must be raised so that he is also able to make some constructive criticism, which is welcomed, as he is the final user of the system.

3. System Requirements specification

3.1 Software Requirements:

The software requirements are descriptions of features and functionalities of the target system. Requirements convey the expectations of users from the software product. The requirements can be obvious or hidden, known or unknown, expected or unexpected from the client's point of view.

- **Editor:** Anaconda Navigator,Streamlit
- **Language:** Python(Version 3.0)
- **Operating System**: Windows

3.2 Hardware Considerations:

The most common set of requirements defined by any operating system or software application is the physical computer resources, also known as hardware, a hardware requirements list is often accompanied by a hardware compatibility list (HCL), especially in case of operating systems. An HCL lists tested, compatible, and sometimes incompatible hardware devices for a particular operating system or application. The following subsections discuss the various aspects of hardware requirements.

RAM	:	4 GB
Processor	:	Intel core i3
Hard disk space	:	500 GB
Operating System	:	Windows 10

User Requirements :

User requirements exclusively specifies the requirements that is needed to use and work with the application by the end user without any disturbances.

1. PC or Laptop which supports internet
2. Browser
3. Internet connectivity

4.System Design

4.1 INTRODUCTION

System design is the process of designing the elements of a system such as the architecture, modules and components, the different interfaces of those components and the data that goes through that system.

System Analysis is the process that decomposes a system into its component pieces for the purpose of defining how well those components interact to accomplish the set requirements. The purpose of the System Design process is to provide sufficient detailed data and information about the system and its system elements to enable the implementation consistent with architectural entities as defined in models and views of the system architecture.

The purpose of the design phase is to plan a solution of the problem specified by the requirement document. This phase is the first step in moving from problem domain to the solution domain. The design of a system is perhaps the most critical factor affecting the quality of the software, and has a major impact on the later phases, particularly testing and maintenance. The output of this phase is the design document. This document is similar to a blue print or plan for the solution, and is used later during implementation, testing and maintenance.

The design activity is often divided into two separate phase-system design and detailed design. System design, which is sometimes also called top-level design, aims to identify the modules that should be in the system, the specifications of these modules, and how they interact with each other to produce the desired results. At the end of system design all the major data structures, file formats, output formats, as well as the major modules in the system and their specifications are decided.

A design methodology is a systematic approach to creating a design by application of set of techniques and guidelines. Most methodologies focus on system design. The two basic principles used in any design methodology are problem partitioning and abstraction. A large system cannot be handled as a whole, and so for design it’s partitioned into smaller systems. Abstraction is a concept related to problem partitioning. When partitioning is used during design, the design activity focuses on one part of the system at a time. Since the part being designed interacts with other parts of the system, a clear understanding of the interaction is essential for property designing the part.

Software Design is a process to transform user requirements into some suitable form, which helps the programmer in software coding and implementation. For accessing user requirements, an SRS (Software Requirement Specification) document is created whereas for coding and implementation, there is a need of more specific and detailed requirements in software terms. The output of this process can directly be used into implementation in programming languages.

Software design is the first step in SDLC (Software Design Life Cycle), which moves the concentration from problem domain to solution domain. It tries to specify how to fulfil the requirements mentioned in SRS. Software analysis and design includes all activities, which help the transformation of requirement specification into implementation. Requirements specifications specify all functional and non-functional expectations from the software. These requirement specifications come in the shape of human readable and understandable documents, to which a computer has nothing to do. Software analysis and design is the intermediate stage, which helps human-readable requirements to be transformed into actual code.

SOFTWARE DESIGN LEVELS: Software design yields three levels of results:

Architectural Design:

The architectural design is the highest abstract version of the system. It identifies the software as a system with many components interacting with each other. At this level, the designers get the idea of proposed solution domain.

High-level Design:

The high-level design breaks the 'single entity-multiple component' concept of architectural design into less-abstracted view of sub-systems and modules and depicts their interaction with each other. High-level design focusses on how the system along with all of its components can be implemented in forms of modules. It recognizes modular structure of each sub-system and their relation and interaction among each other.

Detailed Design:

Detailed design deals with the implementation part of what is seen as a system and its sub-systems in the previous two designs. It is more detailed towards modules and their implementations. It defines logical structure of each module and their interfaces to communicate.

Modularization:

Modularization is a technique to divide a software system into multiple discrete and independent modules, which are expected to be capable of carrying out tasks independently. These modules may work as basic constructs for the entire software. Designers tend to design modules such that they can be executed and compiled separately and independently. Modular design unintentionally follows the rules of 'divide and conquer' problem-solving strategy this is because there are many other benefits attached with the modular design of software. Advantages of Modularization:

1) Smaller components are easier to maintain.

2) Program can be divided based on functional aspects.

3) Desired level of abstraction can be brought in the program.

4) Components with high cohesion can be re-used.

5) Concurrent execution can be made possible.

6) Desired from security aspect.

SYSTEM DESIGN IMPROVEMENTS:

- In Many websites the datasets that are taken into consideration are small datasets, i.e. less number of Birds Species.

- Also noise images are present.

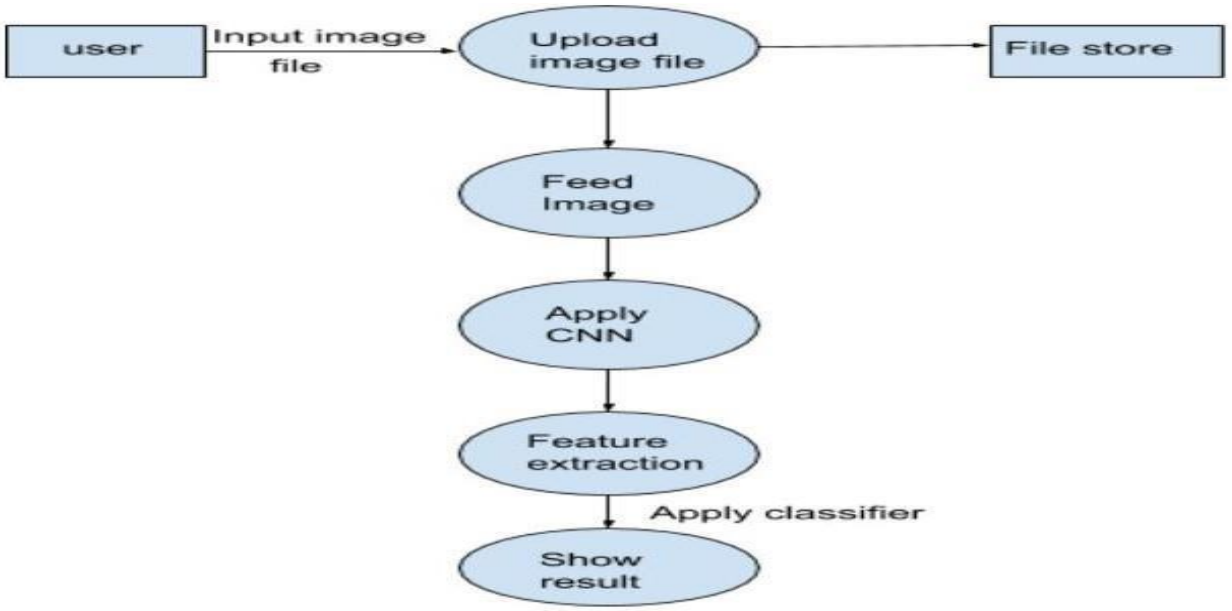
- So our aim is to improve these drawbacks and achieve the better accuracy.

- So for that we have used the CNN Algorithm for the better result to detect the type of species by using the image of bird.

- In our Project, We also implemented some other functionalities such as

    - Predicting the Scientific Name of the Bird
    - Characteristics of Bird
    - Lifespan of Bird
    - National Birds of a Country.

FLOW OF THE SYSTEM



Flow of System

The above figure represents the actual flow of the proposed system. To develop such a system a trained dataset is required to classify an image. The trained dataset consists of train results and test results. Whenever a user will upload an input file, the image is temporarily stored in the database. This input file is then fed to the system and given to CNN where CNN is coupled with a trained dataset. CNN consists of various convolutional layers. Various features such as head, color, body, shape, and the entire image of the bird are considered for classification to yield maximum accuracy. Each alignment is given through a deep convolutional network to extract features out from multiple layers of the network. Then an unsupervised algorithm called deep learning using CNN is used to classify that image.

Further, a grey scale method is used to classify the image pixel by pixel. These features are then aggregated and forwarded to classifier. Here, the input will be compared against 37 the trained dataset to generate possible results.

4.2 UML DIAGRAMS

UML Diagrams is a rich visualizing model for representing the system architecture and design. These diagrams help us to know the flow of the system.

Some of them are;

- Use case diagram
- Class diagram
- Sequence diagram
- Collaboration diagram
- Activity Diagram

USECASE DIAGRAM:

A Use Case Diagram in the Unified Modelling Language (UML) is a type of behavioural diagram defined by and created from a Use-case analysis. The main purpose of a use case diagram is to show what system functions are performed for which actor. Roles of the actors in the system can be depicted. Interaction among actors is not shown on the use case diagram.

Use cases:

A use case describes a sequence of actions that provide something of measurable value to an actor and is drawn as a horizontal ellipse.

Actors:

An actor is a person, organization, or external system that plays a role in one or more interactions with the system.

Four relationships among use cases are used often in practice.

Include:

In one form of interaction, a given use case may include another. "Include is a Directed Relationship between two use cases, implying that the behaviour of the included use case is inserted into the behaviour of the including use case.

This is useful for extracting truly common behaviours from multiple use cases into a single description. The notation is a dashed arrow from the including to the included use case, with the label "«include»".There are no parameters or return values.

Extend:

In another form of interaction, a given use case (the extension) may extend another. This relationship indicates that the behaviour of the extension use case may be inserted in the extended use case under some conditions. The notation is a dashed arrow from the extension to the extended use case, with the label "«extend

Identified Use Cases

The "user model view" encompasses a problem and solution from the preservative of those individuals whose problem the solution addresses. The view presents the goals and objectives of the problem owners and their requirements of the solution. This view is composed of "use case diagrams". These diagrams describe the functionality provided by a system to external integrators. These diagrams contain actors, use cases, and their relationships.

A single use case diagram captures a particular functionality of a system.

1. Functionalities to be represented as use case
2. Actors
3. Relationships among the use cases and actors

In our example the preliminaries are,

ACTORS :

- User
- ML Model

USECASES OF USER:

- Bird Specie Detection
- Common to Scientific Name
- Family Tree
- National Bird
- Characteristics
- Lifespan of Birds

USECASES OF ML MODEL :

- Display Options
- Displays the name of the bird
- Displays Scientific Name
- Displays Bird Images
- Displays Bird Name
- Displays an image with Details of Bird
- Displays an image with life span of birds.


select list of options
Display Options
Bird Specie Detection
Includes
Input an Image
Includes
Displays the name of the bird
Common to Scientific Name
Includes
Input a bird Name
Includes
Displays the sientific name
Family Tree
Includes
Input a Brd Name
Includes
Displays bird Images
National Birds
Includes
Input a country Name
Includes
displays the bird Name
Characteristics
Includes
Input a bird Name
Includes
Displays an Image with Details of Birds
Lifespan of Birds
Includes
Input a bird Name
Includes
dispays an image with life span of birds
USER
ml model

CLASS DIAGRAM:

In software engineering, a class diagram in the Unified Modeling Language is **a type of static structure diagram** that describes the structure of a system by showing the system's classes, their attributes, operations and the relationships among objects.

Purpose of Class Diagrams

- Shows static structure of classifiers in a system
- Diagram provides a basic notation for other structure diagrams prescribed by UML
- Helpful for developers and other team members too
- Business Analysts can use class diagrams to model systems from a business

A UML class diagram is made up of:

- A set of classes and
- A set of relationships between classes

About the class diagram below:

Classes:

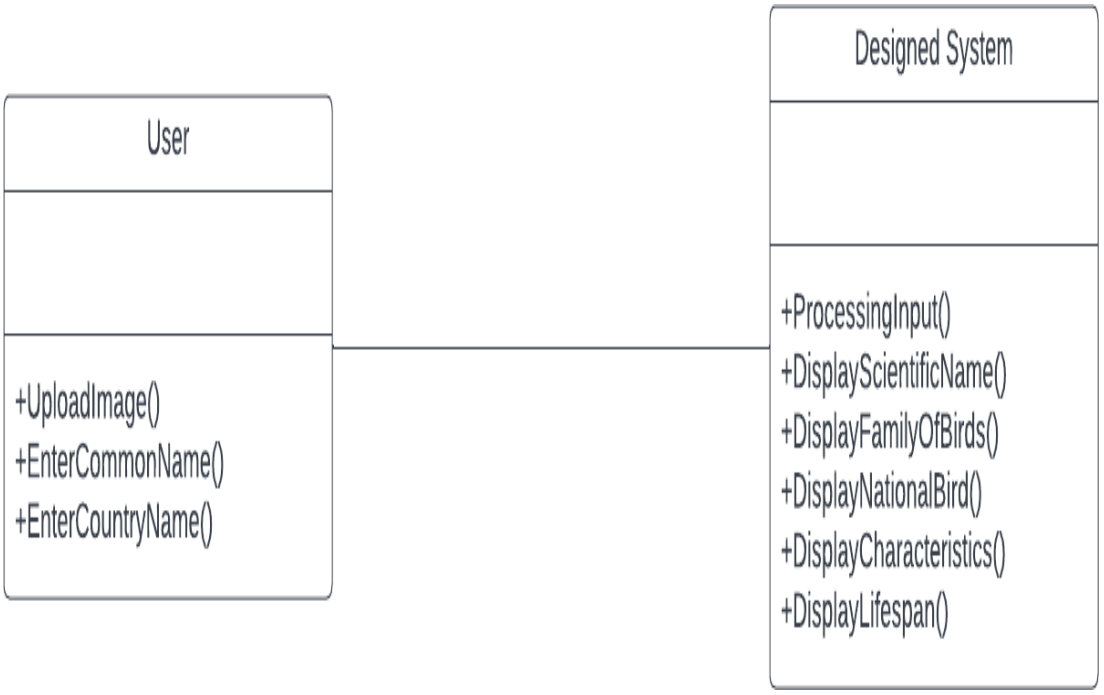
1. User
2. Designed System

Functions:

1. Functions of user class:
    1. UploadImage()
    2. EnterCommonName()
    3. EnterCountryName()

2. Functions of Designed System:
    1. ProcessingInput()
    2. DisplayScientificName()
    3. DisplayFamilyOfBirds()
    4. DisplayNationalBird()
    5. DisplayCharacteristics()
    6. DisplayLifespan()

CLASS DIAGRAM FOR BIRD SPECIES DETECTION:



SEQUENCE DIAGRAM

A Sequence diagram is an interaction diagram that emphasizes the time ordering of messages. It consists of a set of objects and their Relationships, including the messages that may be dispatched among them.

A **sequence diagram** shows object interactions arranged in time sequence. It depicts the objects and classes involved in the scenario and the sequence of messages exchanged between the objects needed to carry out the functionality of the scenario. Sequence diagrams are sometimes called **event diagrams** or **event scenarios**.

A sequence diagram shows, as parallel vertical lines (***lifelines***), different processes or objects that live simultaneously, and, as horizontal arrows, the messages exchanged between them, in the order in which they occur.

Sequence diagrams address the dynamic view of the system. Sequence diagrams are two dimensional in nature.

Sequence Diagram captures:

- The interaction that takes place in a collaboration that either realizes a use case or an operation.
- High-level interactions between user of the system and the system, between the system and other systems, or between subsystems.
- Model the high-level interaction between active objects in a system.

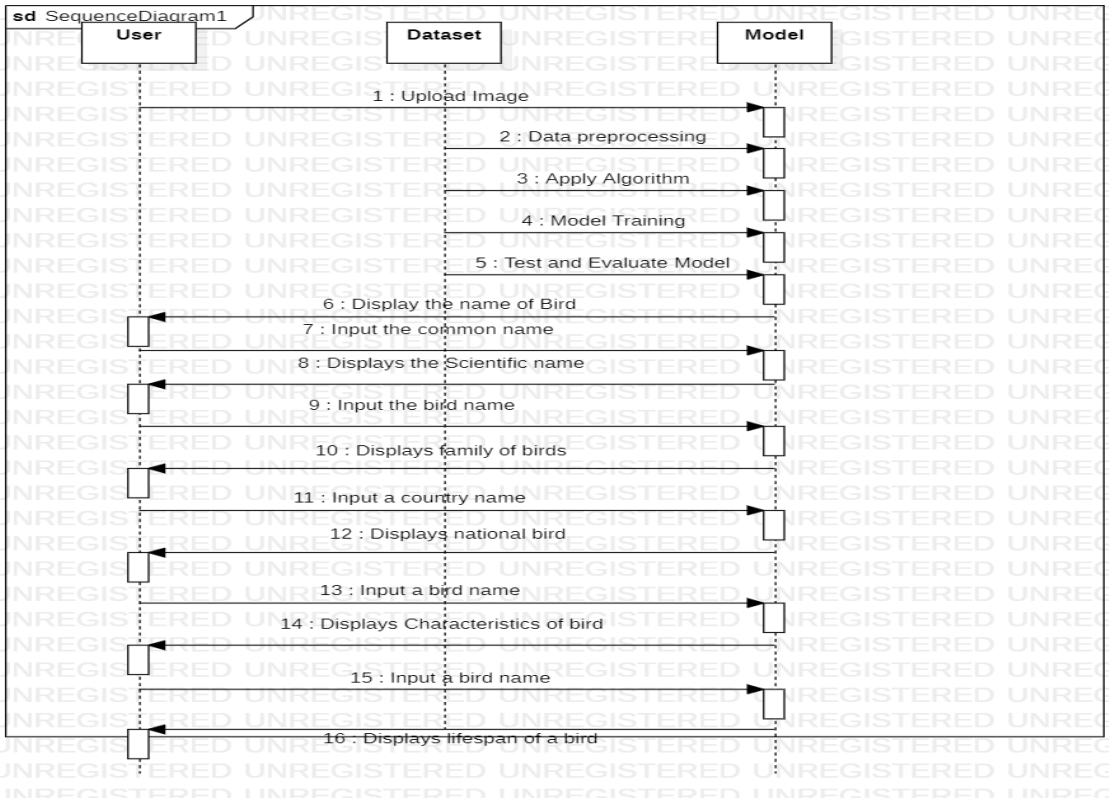
LIFE LINES:

1. User
2. Dataset
3. Model

Flow of Actions:

1. Upload Image
2. Data Preprocessing
3. Apply Algorithm
4. Model Training
5. Test and Evaluate Model
6. Display the name of Bird
7. Input the common Name
8. Displays the Scientific Name
9. Input the bird Name
10. Displays family of Birds
11. Input a country name
12. Displays national bird
13. Input a bird name
14. Displays Characteristics of bird
15. Input a bird name
16. Displays lifespan of a bird

SEQUENCE DIAGRAM FOR BIRD SPECIES DETECTION:



COLLABORATION DIAGRAM

A Communication Diagram shows the interactions between the objects or roles associated with lifelines and the messages that pass between lifelines.

A collaboration diagram is an interaction diagram that emphasizes the structural organization of the objects that send receive message.

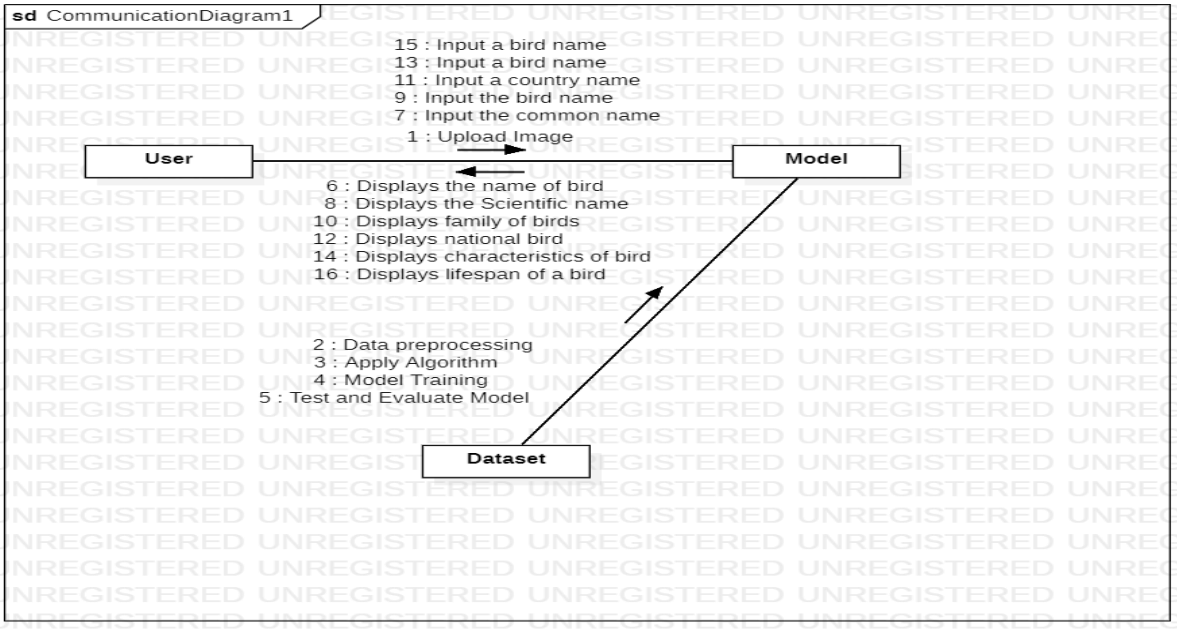
A collaboration diagram is very similar to sequence diagram. Collaboration diagram shows the objects and their association with other objects.

Sequence diagrams and collaboration diagrams shows same information but sequence diagram focus on the temporal aspect and collaboration diagram focus on communication between the objects of system.

This Diagram is an extension of object diagram that shows the objects along with the messages that travel from one to another. In addition to the associations among objects, communication diagram shows the messages the objects send each other.

They are used to understand the object architecture within a system rather than the flow of a message as in a sequence diagram.

COLLABORATION DIAGRAM FOR BIRD SPECIES DETECTION:



ACTIVITY DIAGRAM

Activity diagram is defined as a UML diagram that focuses on the execution and flow of the behavior of a system instead of implementation. It is also called object-oriented flowchart.

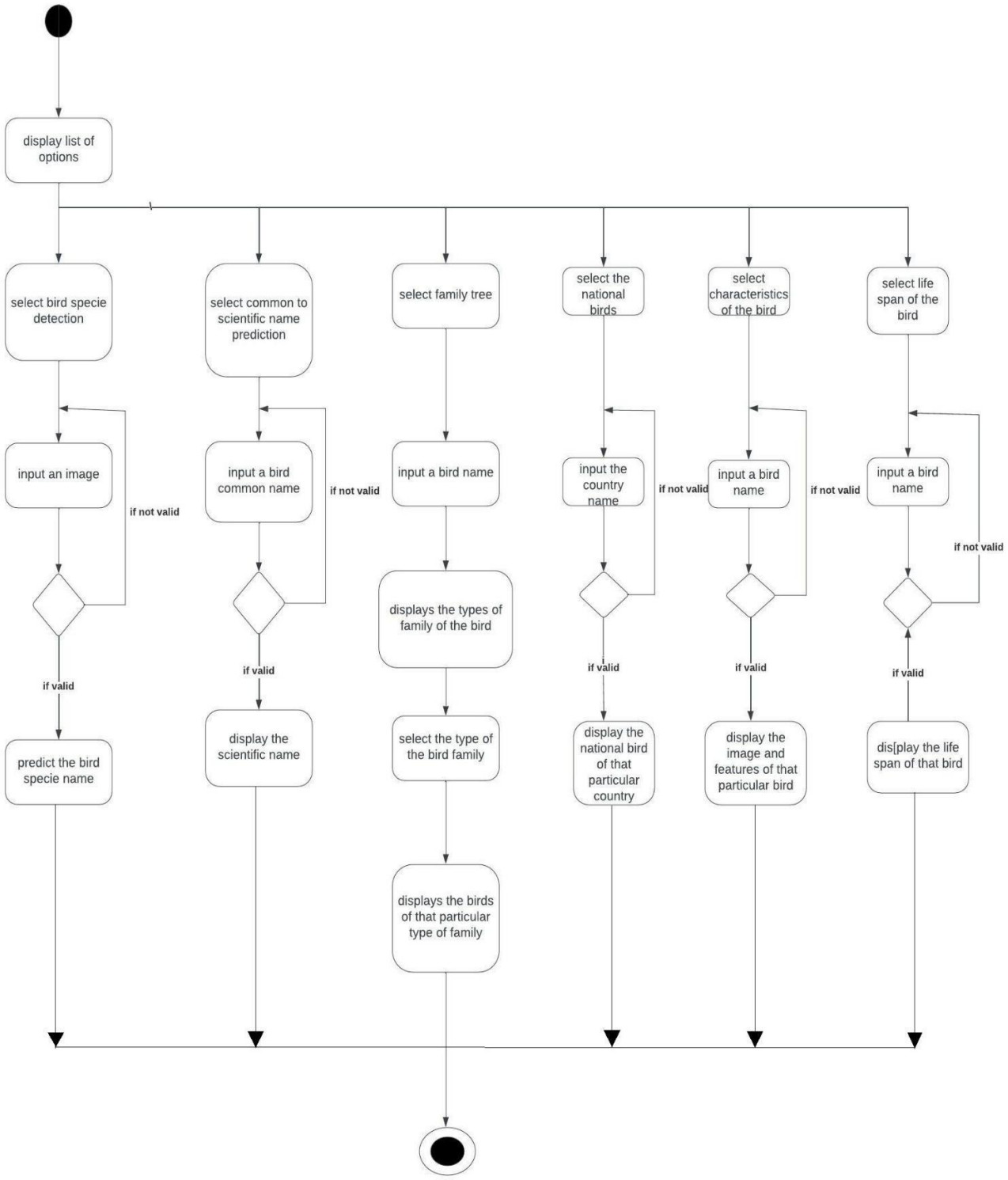
Activity diagrams consist of activities that are made up of actions which apply to behavioral modeling technology.

Activity diagrams are used to model processes and workflows. The essence of a useful activity diagram is focused on communicating a specific aspect of a system's dynamic behavior.

Activity diagrams capture the dynamic elements of a system. Activity diagram is similar to a flowchart that visualizes flow from one activity to another activity. Activity diagram is identical to the flowchart, but it is not a flowchart.

The flow of activity can be controlled using various control elements in the UML diagram. In simple words, an activity diagram is used to activity diagrams that describe the flow of execution between multiple activities.

ACTIVITY DIAGRAM FOR BIRD SPECIES DETECTION:



5. SYSTEM IMPLEMENTATION

5.1 INTRODUCTION

- Bird species Detection means predicting the bird species belongs to which category by using an image

- In our Project we created a web page where the user can upload an image so that the trained model of our project can predict the bird specie name and print the output on the web page.

- We also added some features such that the bird common name is being entered on our web page and the scientific name is being printed on the web page and Bird Life Span and Characteristics.

- The Main Objective of this Project is, We can easily identify the name of bird by using an image and also know about the characteristics of the Bird.

This Project Consists of different Modules. They are:

- Bird Species Detection
- Common Name to Scientific Name
- Family Tree
- Characteristics Of a Bird
- Finding National Birds
- Lifespan of Birds

5.2 PROJECT MODULES

1. **Bird Species Detection:**

- In this Bird Species Detection, the User enters any of the Bird Image.
- Then it displays the Name of the Bird as Output.

2. **Common Name to Scientific Name:**

- In this Module, the user has to enter any of the Common Name of the Bird.
- Then the Scientific Name of the Particular Bird is displayed on the Screen.

3. **Family Tree:**

- In this Module, User has to enter any of the Bird Name as the Input.
- Then it displays the list of birds belonging to that Particular Bird Species.
- If we Select any of the Bird from that list, it displays the images related to that particular Bird.

4. **National Birds:**

- In this Module, the User enters any of the Country Name as input.
- Then the National Bird of that Country is displayed on the Screen.

5. **Characteristics of Bird:**

- In this Module, the User has to enter any of the Bird name as Input.
- Then the Characteristics like Size, Colour etc are displayed on the Screen.

6. **Life Span:**

- In this Module, the User has to enter any of the Bird Name as Input.
- Then the lifesan of that particular Bird is displayed on the Screen.

5.2.1 ALGORITHM

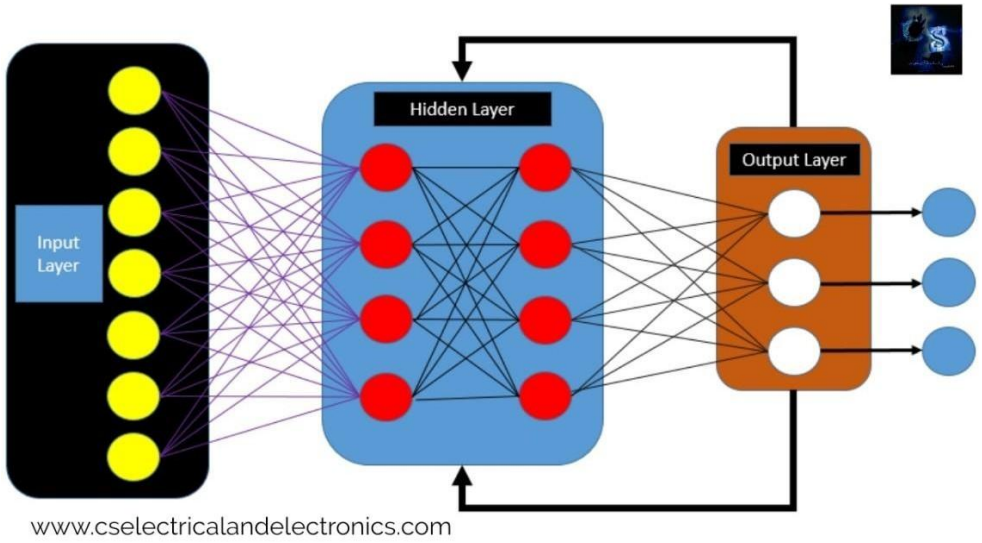
In this project, unsupervised learning algorithm has been used for developing the system, because the inputted image defined is not known. Also, the data which is given to unsupervised learning algorithm are not labeled, i.e. only the input variables(X) are given with no corresponding output variables. In unsupervised learning, algorithms discover interesting structures in the data themselves.

A Convolutional Neural Network (ConvNet/CNN) is a Deep Learning algorithm which can take in an input image, assign importance (learnable weights and biases) to various aspects/objects in the image and be able to differentiate one from the other. The pre-processing required in a ConvNet is much lower as compared to other classification algorithms. While in primitive methods filters are hand-engineered, with enough training, ConvNets have the ability to learn these filters/characteristics.

Next, we will apply a Pooling layer to our Convolutional layer, so that from every feature map we create a Pooled feature map as the main purpose of the pooling layer is to make sure that we have spatial invariance in our images. It also helps to reduce the size of our images as well as avoid any kind of overfitting of our data. After that, we will flatten all of our pooled images into one long vector or column of all of these values, followed by inputting these values into our artificial neural network. Lastly, we will feed it into the locally connected layer

To achieve the final output.

.



The CNN have two components:

1) **Feature extraction part**: Features are detected when network performs a series of convolutional and pooling operation.

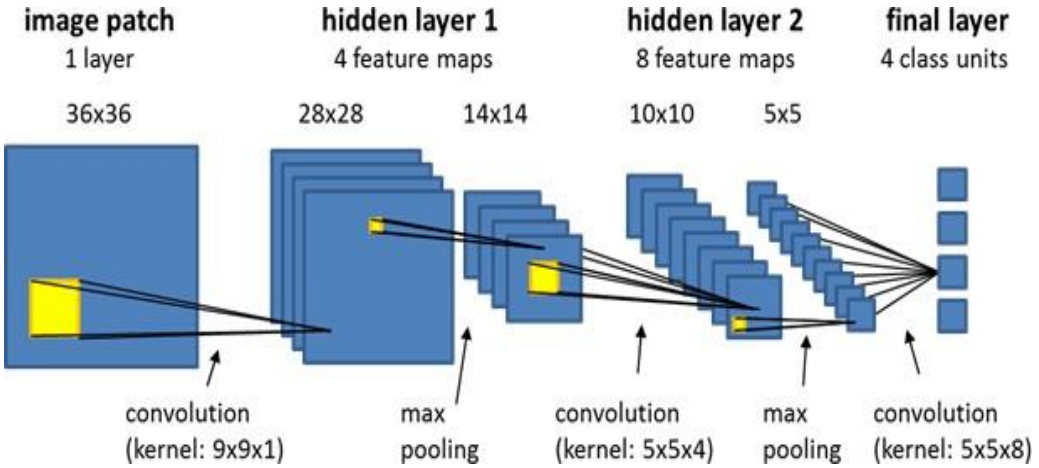
2) **Classification part**: Extracted features are given to fully connected layer which acts as classifier.

CNN consists of four layers:

- convolutional layer, activation layer, pooling layer and fully connected.

1. **Convolution Layer:**

   In convolution layer we take a small window size [typically of length 5*5] that extends to the depth of the input matrix. The layer consists of learnable filters of window size. During every iteration we slid the window by stride size [typically 1], and compute the dot product of filter entries and input values at a given position. As we continue this process we will create a 2-Dimensional activation matrix that gives the response of that matrix at every spatial position. That is, the network will learn filters that activate when they see some type of visual feature such as an edge of some orientation or a blotch of some colour.



2. Pooling Layer:

We use pooling layer to decrease the size of activation matrix and ultimately reduce the learnable parameters. There are two types of pooling: a. Max Pooling: In max pooling we take a window size [for example window of size 2*2], and only take the maximum of 4 values. Well lid this window and continue this process, so well finally get an activation matrix half of its original Size. b. Average Pooling: In average pooling, we take advantage of of all Values in a window.

There are three types of Pooling:

- Max Pooling
- Average Pooling
- Global Pooling.

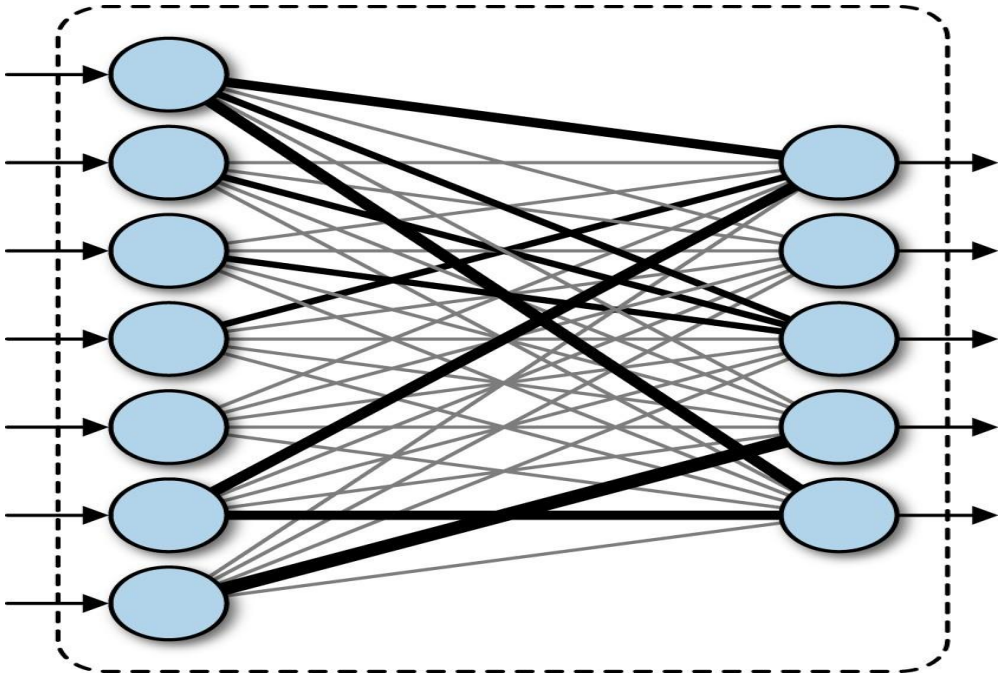
Max Pooling: Max pooling is a pooling operation that selects the maximum element from the region of the feature map covered by the filter. Thus, the output after maxpooling layer would be a feature map containing the most prominent features of the previous feature map. In Fig. 3 we perform maximum pooling operation by considering convoluted feature output obtained from convolution layer.

Average Pooling: Average pooling computes the average of the elements present in the region of feature map covered by the filter. Thus, while max pooling gives the most prominent feature in a particular patch of the feature map, average pooling gives the average of features present in a patch.

Global Pooling: Global pooling reduces each channel in the feature map to a single value. Thus, an nh x nw x nc feature map is reduced to 1 x 1 x nc feature map. This is equivalent to using a filter of dimensions nh x nw i.e. the dimensions of the feature map. Further, it can be either global max pooling or global average pooling

3. Fully Connected Layer:

In convolution layer, neurons are connected only to a local region, while in a fully connected region, we will connect all the inputs to neurons.



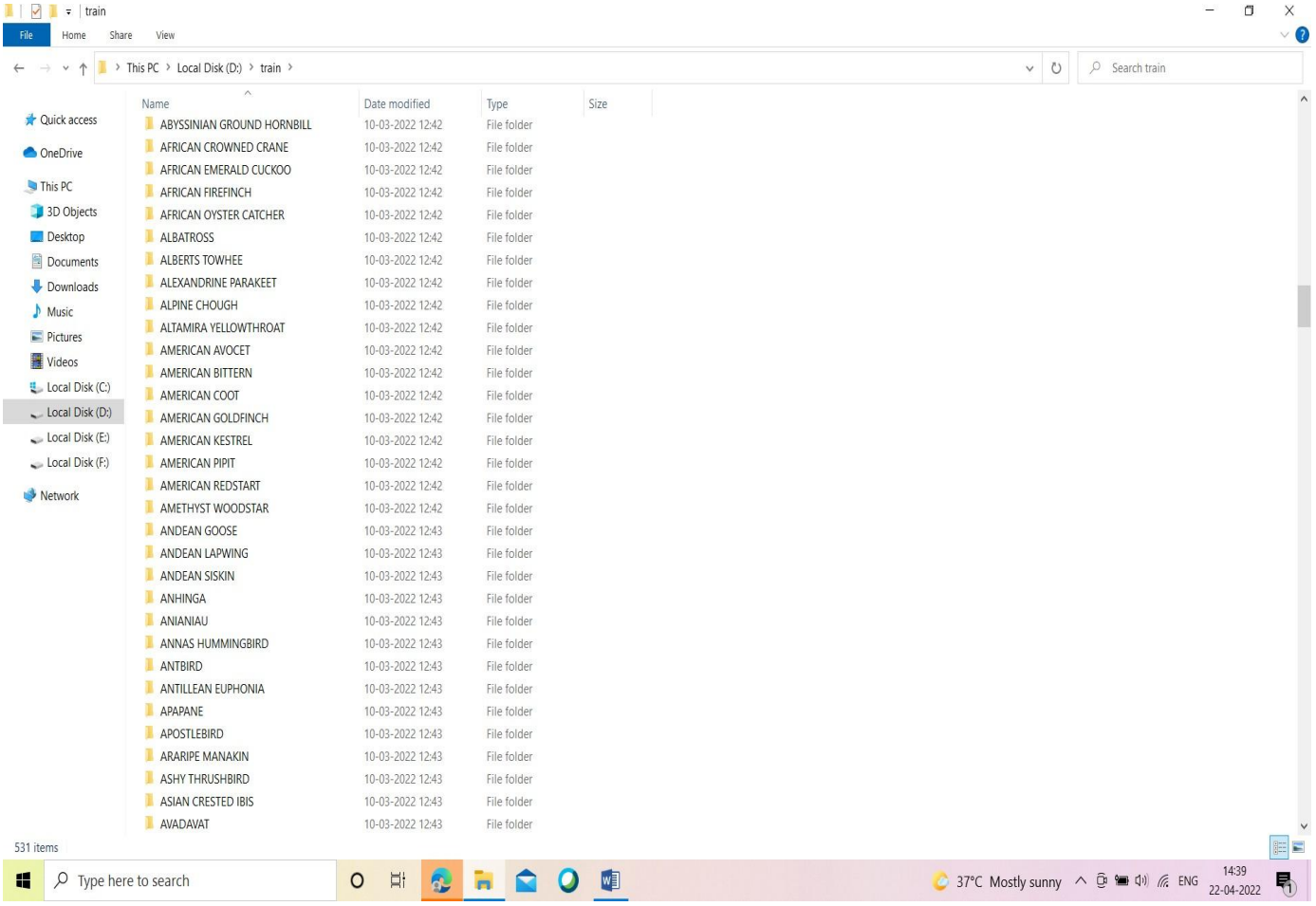
4.Final Output Layer:

After getting values from fully connected layer, we will connect them to the final layer of neurons [having count equal to total number of classes], that will predict the probability of each image to be in different classes.

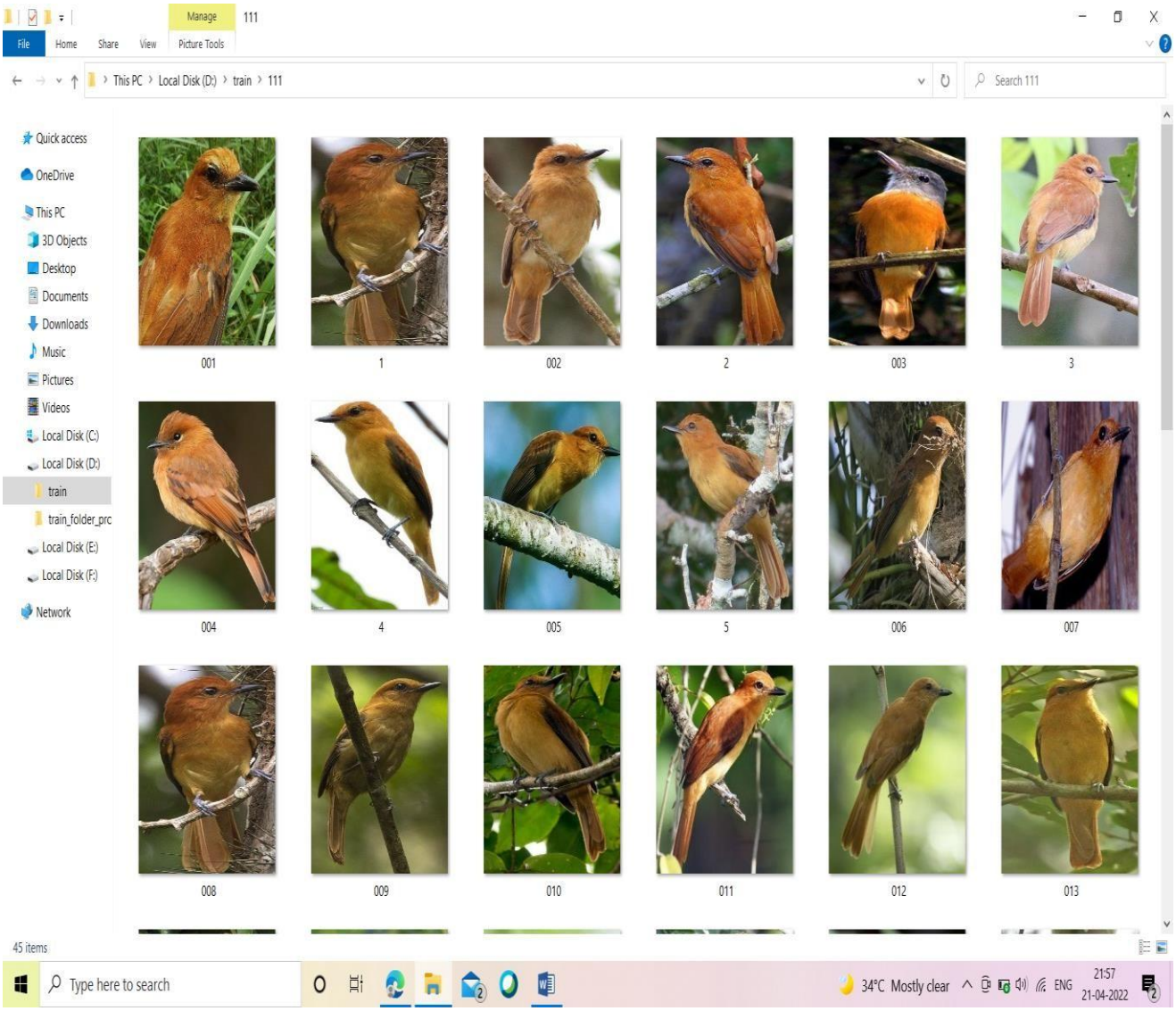
5.3 SCREENS

5.3.1 SCREENS OF DATASET:

1.SCREEN OF THE TRAIN FOLDER:



2. SCREEN OF THE FOLDER IN TRAIN DATASET:

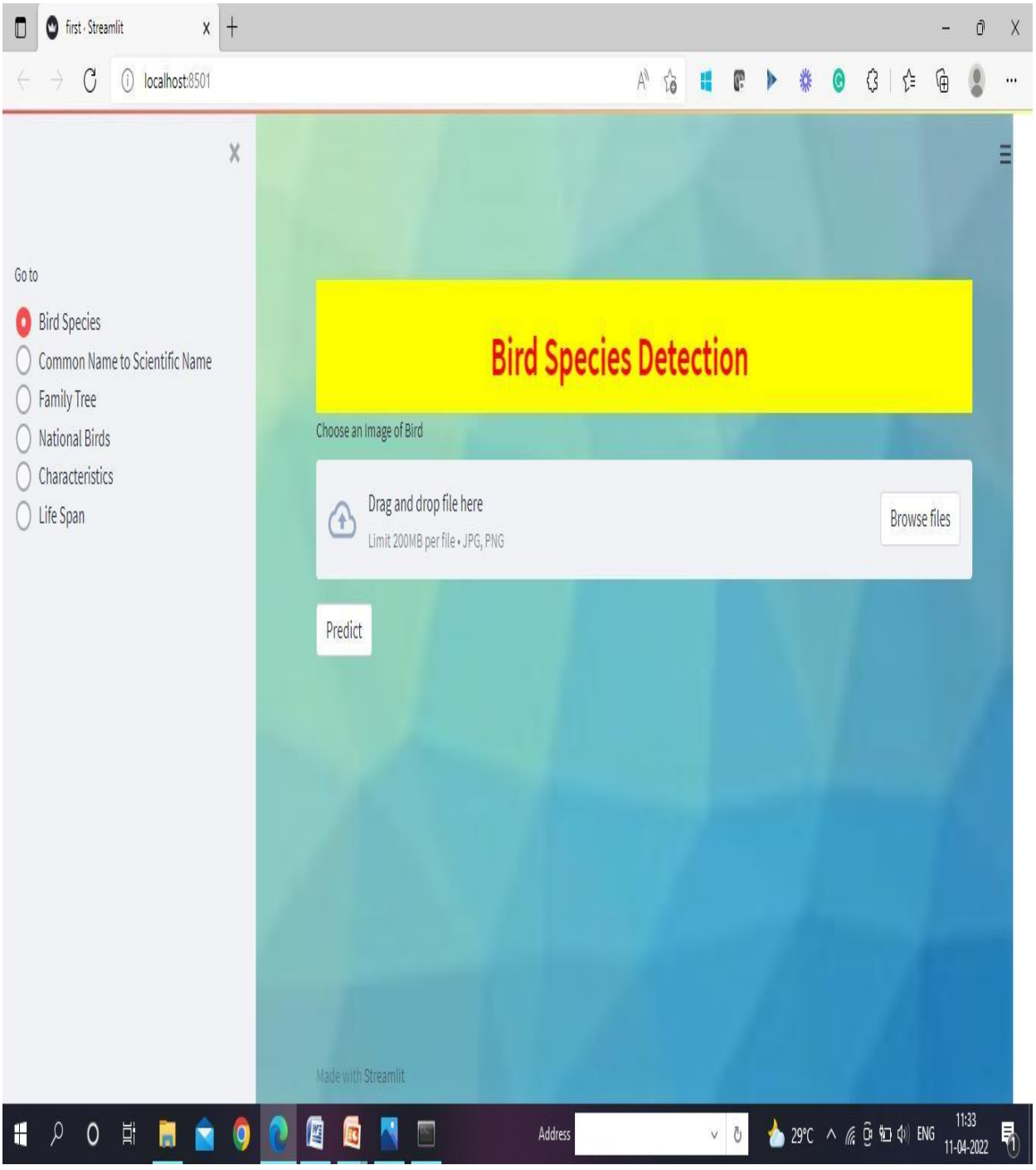


3.SCREEN OF THE FOLDER IN TEST DATASET:

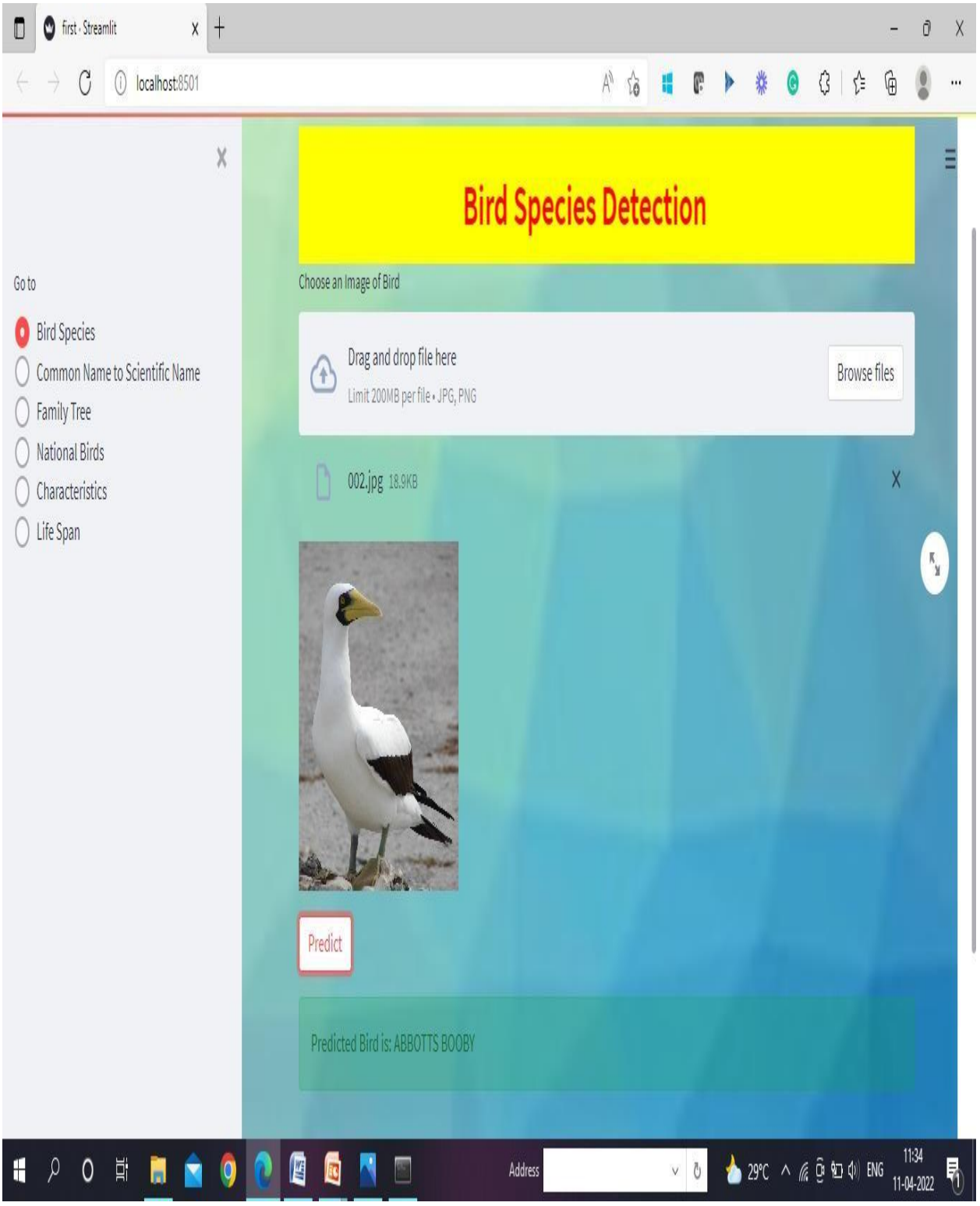
5.3.2 SCREENS OF THE PROJECT:

1.Bird Species Detection:

Before Image Uploaded:

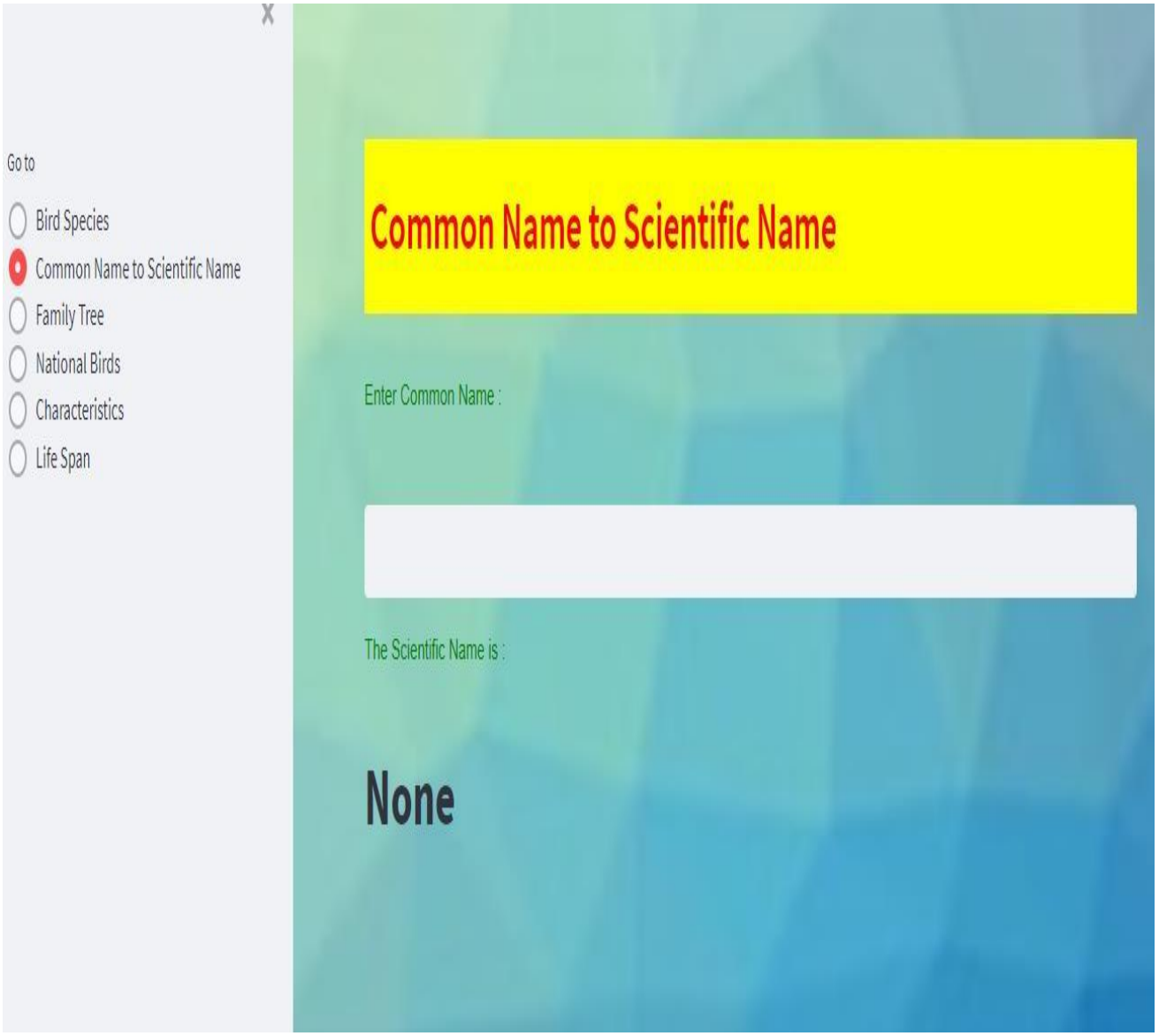


After Image Uploaded:



2. Common Name to Scientific Name:

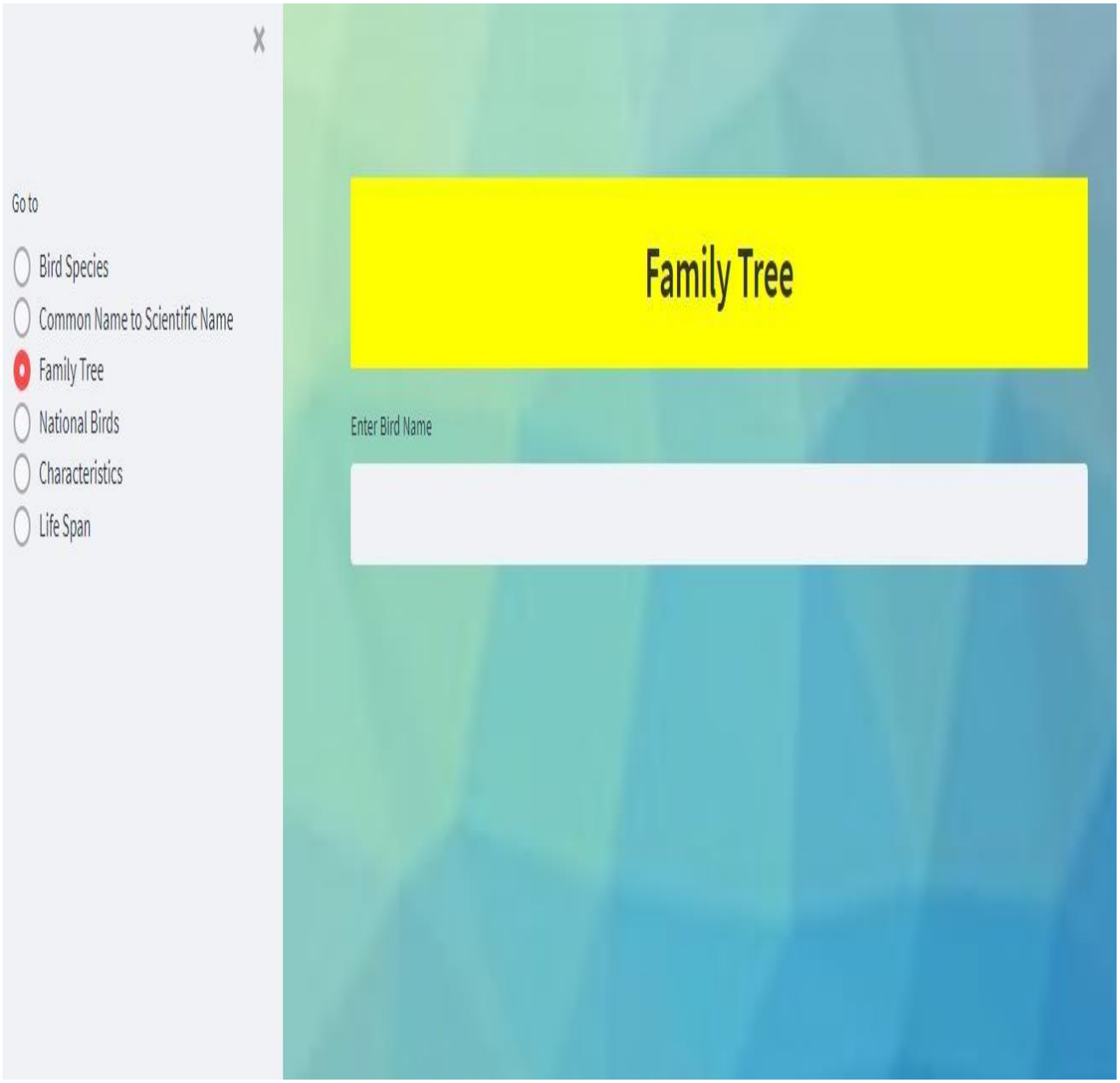
Before Common Name Entered:



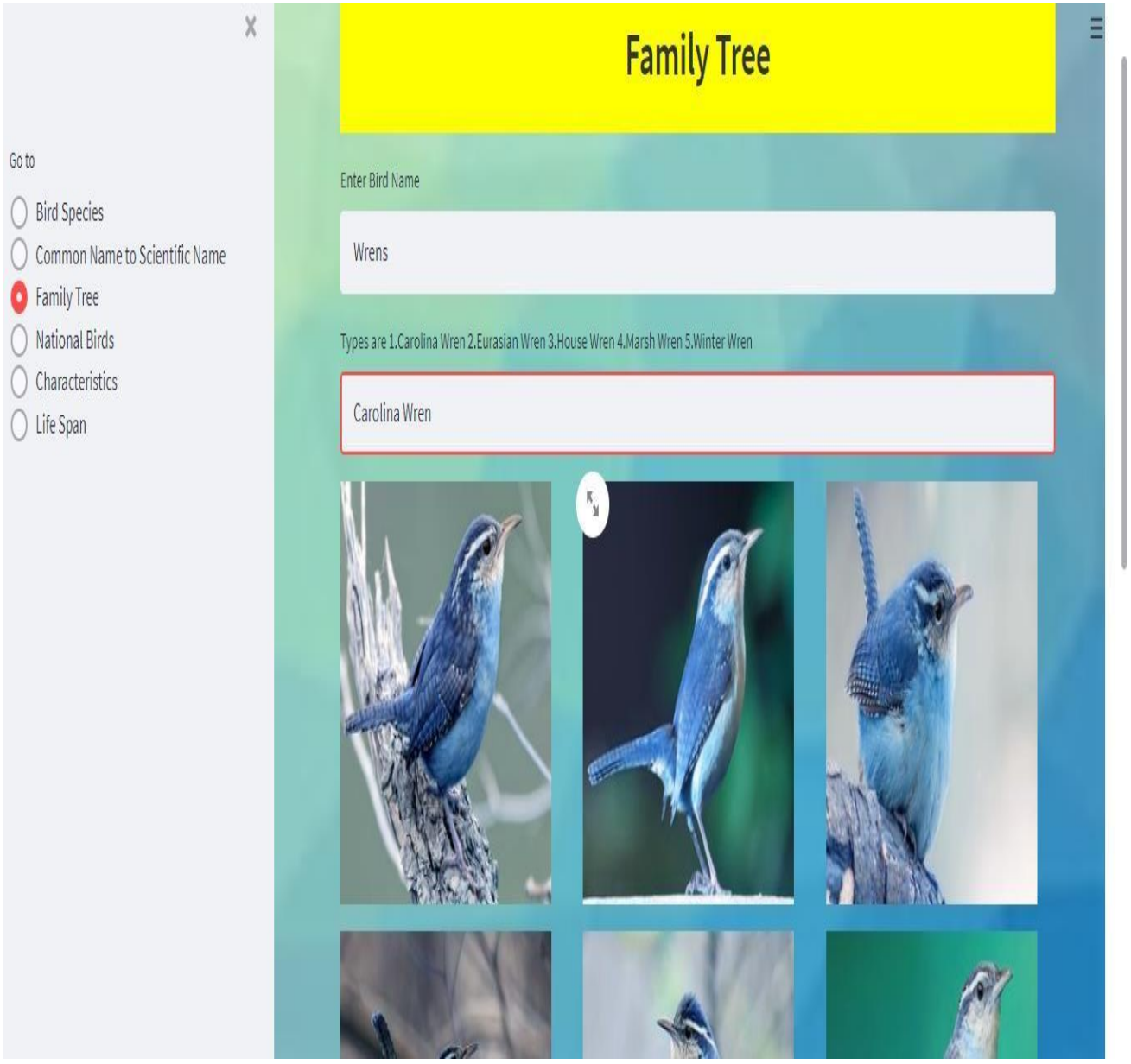
After Common Name Entered:

3. Family Tree

Before Bird Name entered:

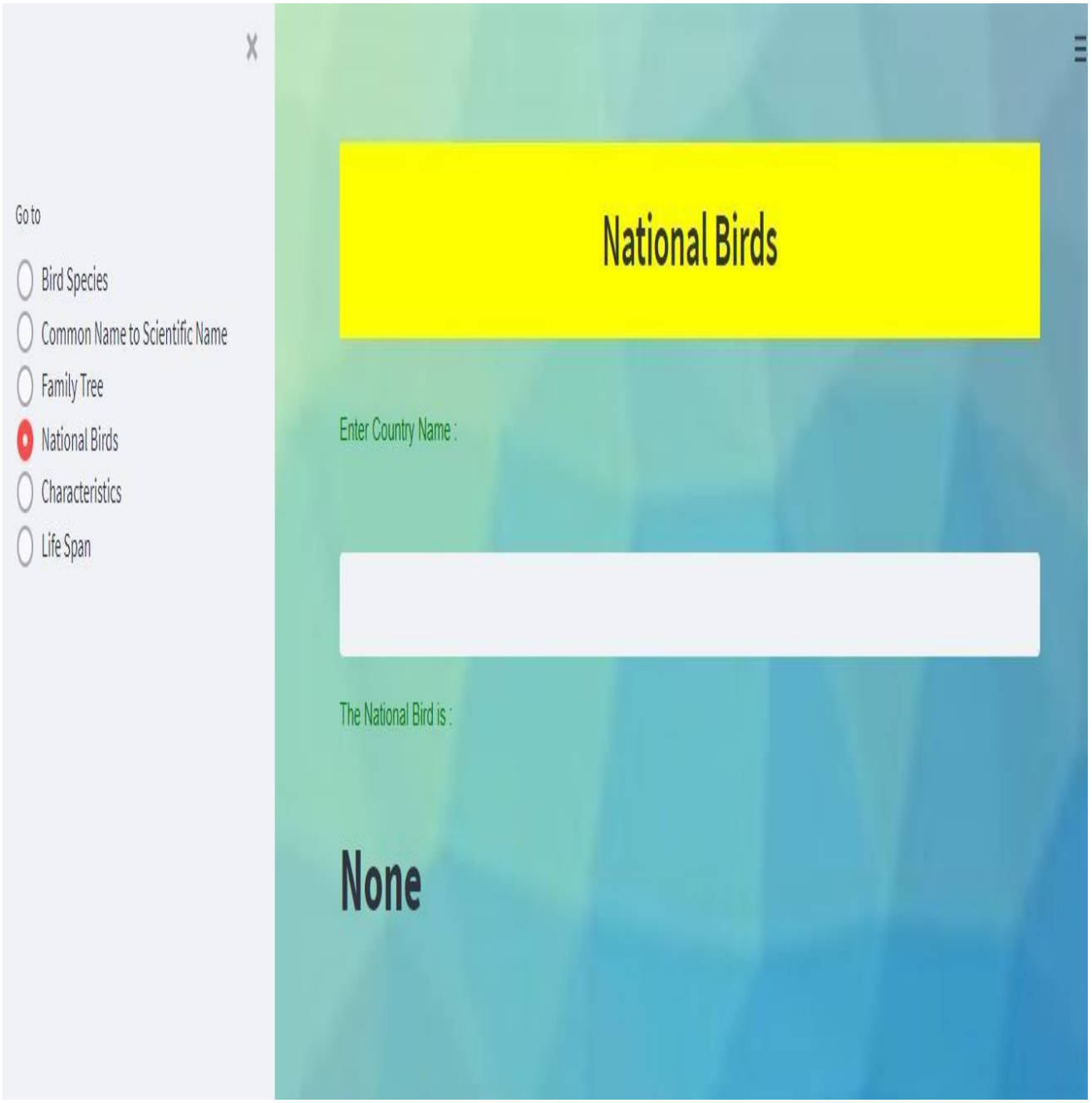


After Bird Name Entered:

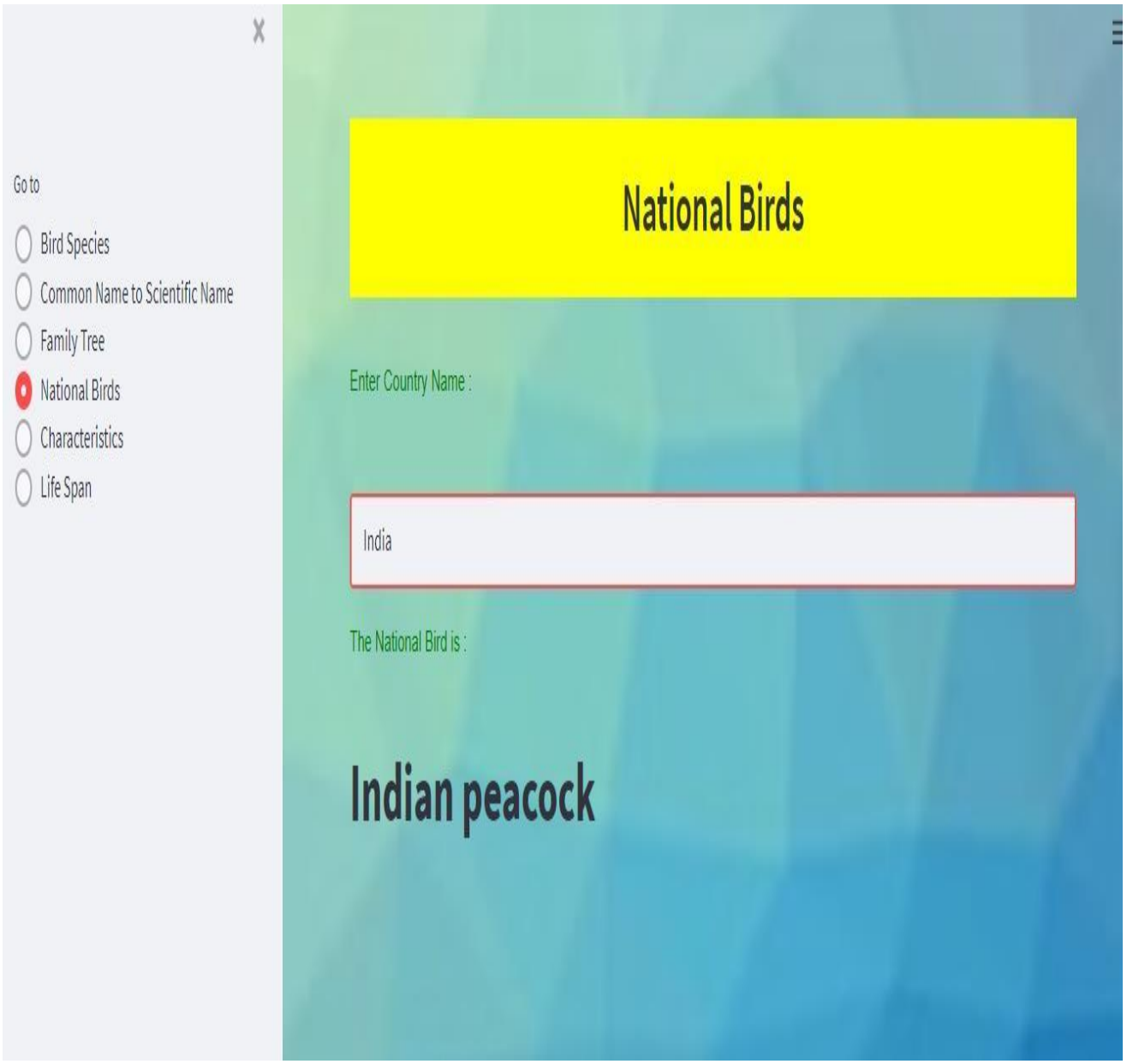


4. National Birds:

Before Country Name Entered:

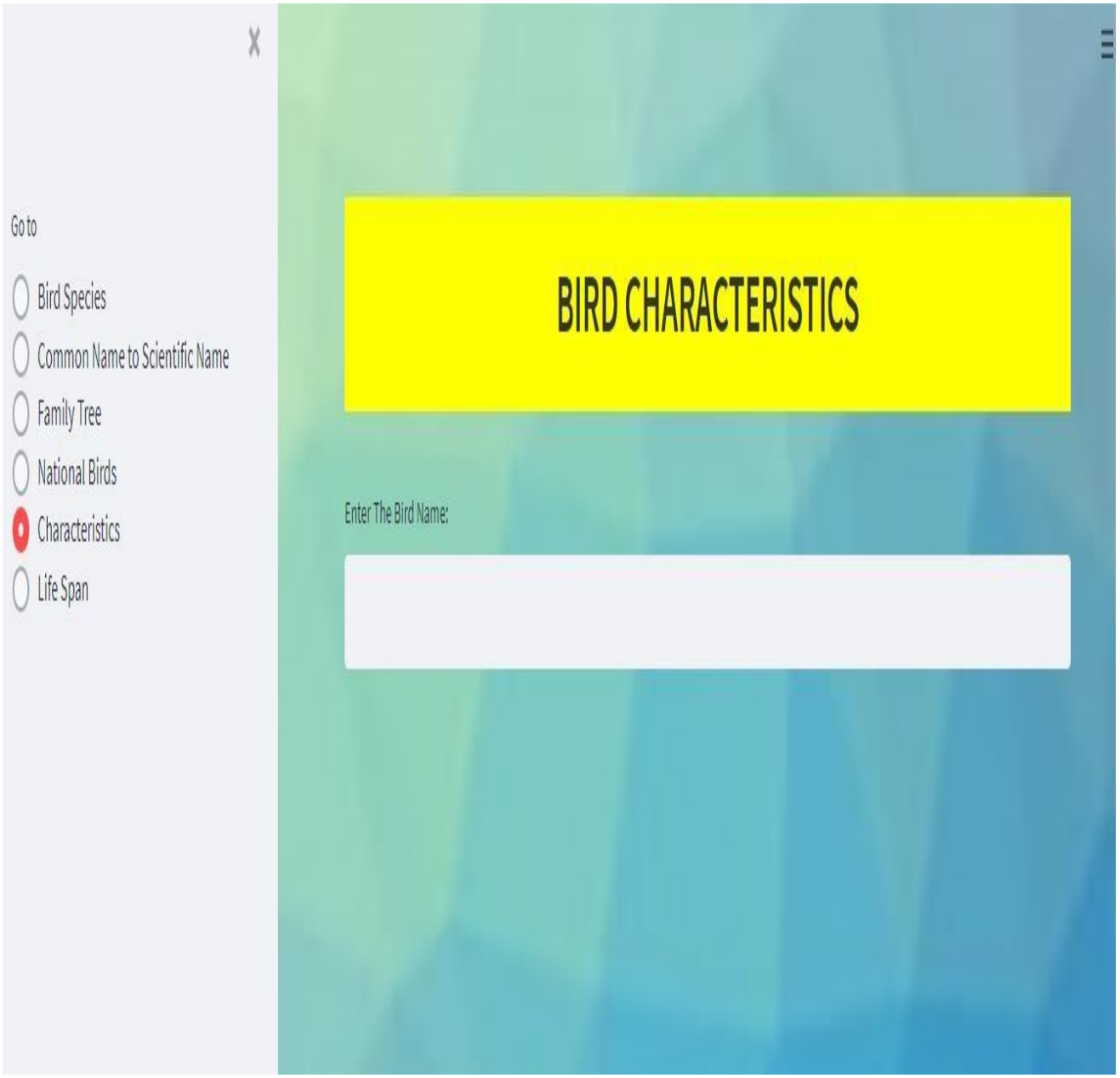


After Country Name Entered:

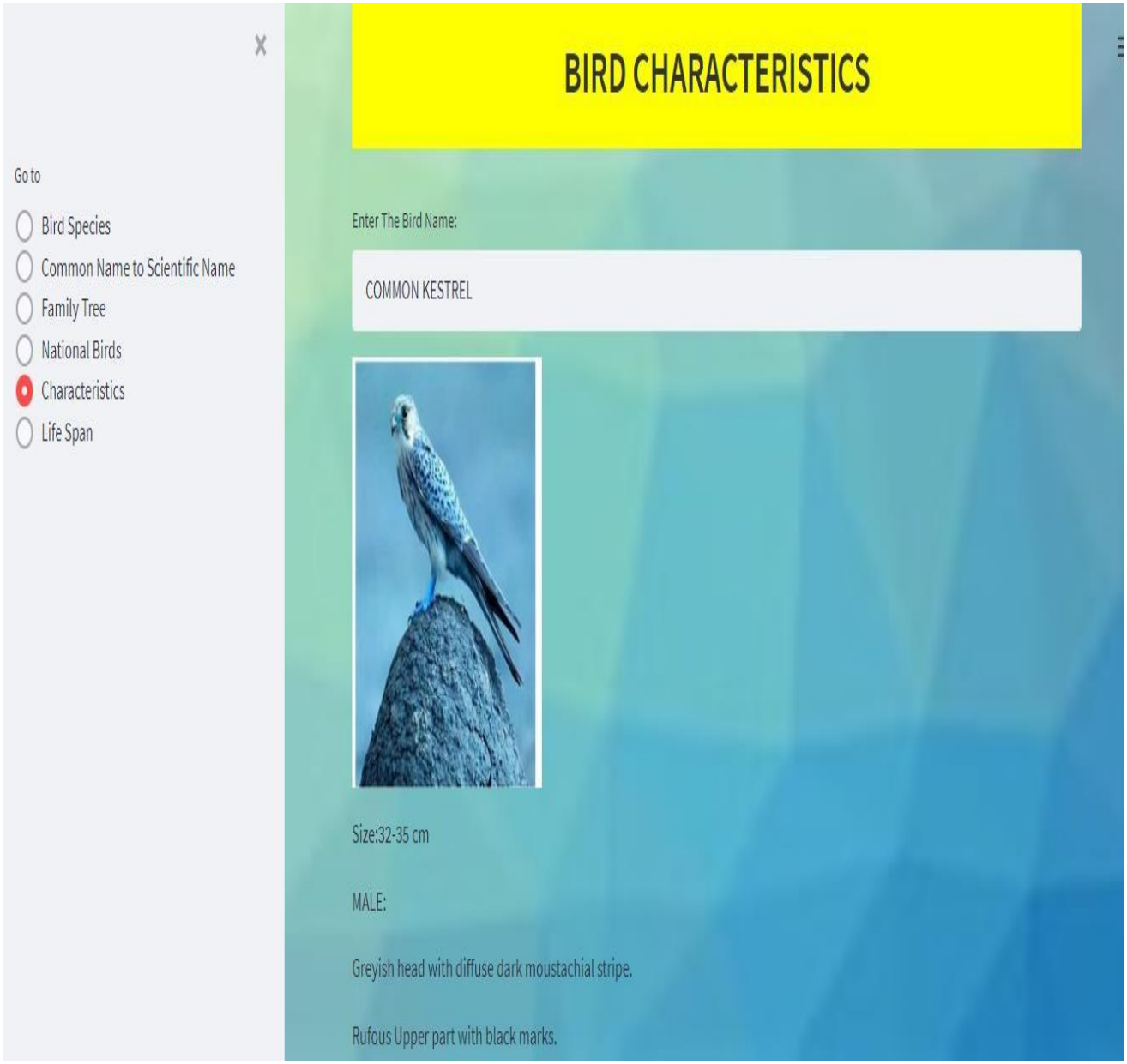


5. Characteristics Of Birds:

Before Bird Name Entered:

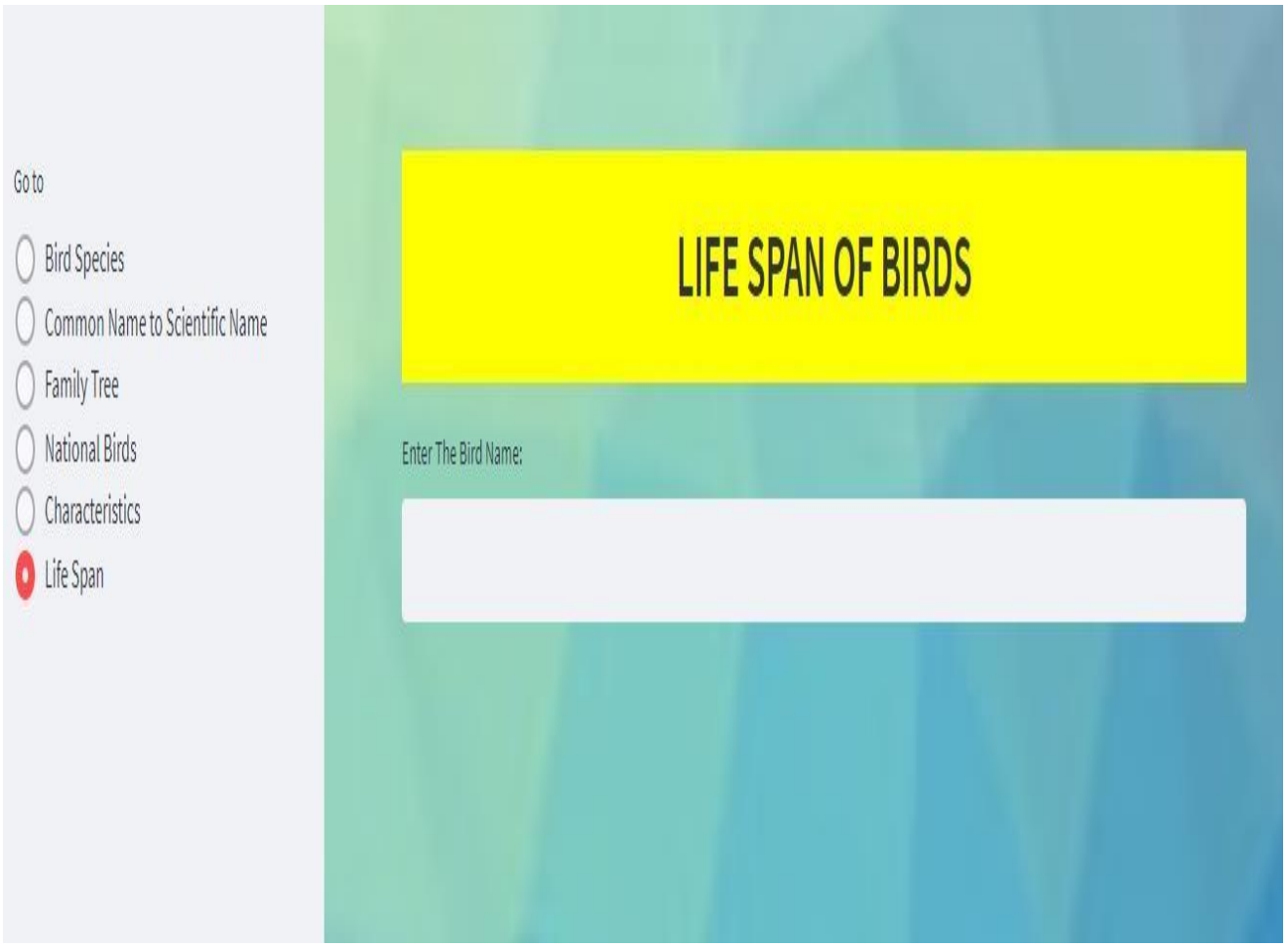


After Bird Name Entered:

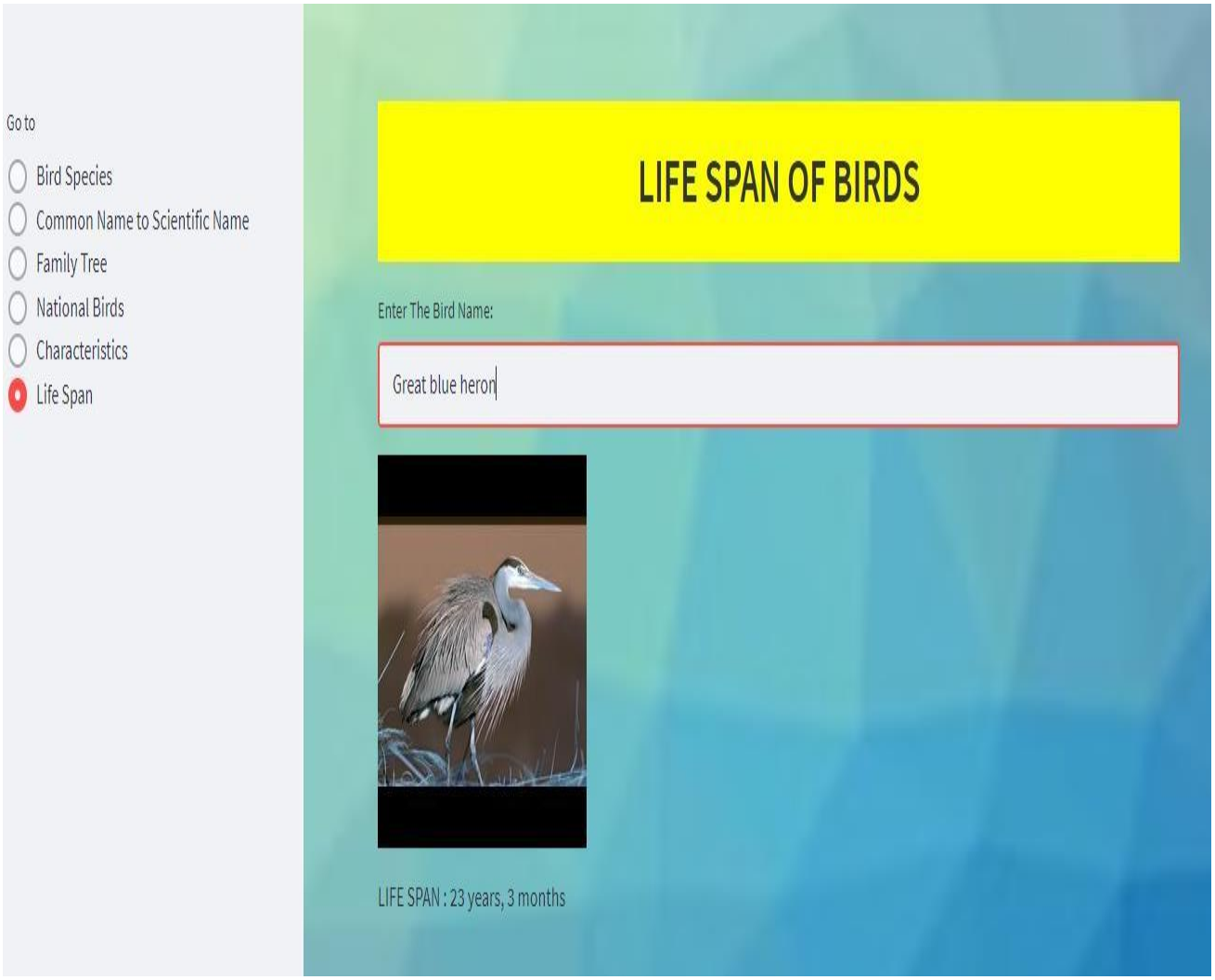


6. Life Span:

Before Bird Name Entered:



After Bird Name Entered:



6.SYSTEM TESTING

6.1. INTRODUCTION:

Software Testing is an important element of the software quality assurance and represents the ultimate review of specification, design and coding.The increasing feasibility of software as a system and the cost associated with the software failures are motivated
forces for III planned through testing.

TESTING OBJECTIVES

These are several rules that can save as testing objectives:

- Testing is a process of executing program with the intent of finding an error.
- A good test case is one that has a high probability of finding an undiscovered error.

Test Levels

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or darkness in a work product. It provides a way to check the functionality of components, subassemblies, assemblies and/or a finished product. Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of test. Each test type addresses a specific testing requirement.

6.2 TESTING METHODS

6.2.1 Unit Testing

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application.

6.2.2 Integration Testing

Integration tests are designed to test integrated software components to determine if they run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields.

6.2.3 Functional Testing

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals. Organization and preparation of functional tests is focused on requirements, key functions, or special test cases.

6.2.4 System Testing

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration oriented system integration test.

6.2.5 White Box Test

White Box Testing is a testing in which in which the software tester has knowledge of the inner workings, structure and language of the software, or at least its purpose. It is purpose. It is used to test areas that cannot be reached from a black box level.

6.2.6 Black Box Test

Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Black box tests, as most other kinds of tests, must be written from a definitive source document, such as specification or requirements document, such as specification or requirements document.

6.2.7 Unit Testing

Unit testing is usually conducted as part of a combined code and unit test phase of the software lifecycle, although it is not uncommon for coding and unit testing to be conducted as two distinct phases.

6.2.8 Integration Testing

Software integration testing is the incremental integration testing of two or more integrated software components on a single platform to produce failures caused by interface defects.

6.2.9 Acceptance Testing

User Acceptance Testing is a critical phase of any project and requires significant participation by the end user.

Test case 1	Giving a Bird image as input for bird specie detection.
Test requirement	To check whether a bird name is displayed.
Input description	The input image need to be predicted and the bird name should be displayed on the screen.
Sample Input	Input a bird image. Example: 
Sample Output	The bird name is displayed on the screen Example: **CROW**
Expected Output	The bird name should be displayed on the screen Example: **CROW**

Test case 2	Giving a Common name of the bird to display scientific name of the bird.
Test requirement	To check whether the scientific name of that bird is displayed.
Input description	The scientific name of the bird should be displayed when an common name of the bird is being given as an user input.
Sample Input	Input a common bird name: Example: **ABBOTTS BABBLER**
Sample Output	The scientific name of the bird is displayed Output: **Malacocincla abbotti**
Expected Output	The scientific name of the bird should be displayed. Output: **Malacocincla abbotti**

Test case 3	Giving a bird image as input to display the characteristics of the bird.
Test requirement	To check whether the characteristics of the bird is displayed on the screen.
Input description	The characteristics of the bird along with the bird image should be displayed on the screen when an bird name is being given as an user input.
Sample Input	Input a bird name: **EGYPTIAN VULTURE**
Sample Output	The bird image and its characteristics need to be displayed on the screen.  **Size-60-70cm** **Dirty white with yellow faceBlack flight feather** **Long and pointed wings** **Wedge shaped tail**
Expected Output	The bird image along with its characteristics need to be displayed on the screen.  **Size-60-70cm** **Dirty white with yellow faceBlack flight feather** **Long and pointed wings** **Wedge shaped tail**

Test case 4	Giving country name to print the national bird of that particular country.
Test requirement	To check whether the national bird of that particular country is printed on the screen when a country name is given.
Input description	The national bird of that particular country should be printed on the screen when a country name is given.
Sample Input	Enter country name: **Albania**
Sample Output	The national bird is: **Golden eagle**
Expected Output	The national bird should be displayed The national bird is: **Golden eagle**

Test case 5	Giving a invalid country name
Test requirement	To check whether the error message is displayed when an invalid country name is given .
Input description	The national bird of that particular country should be printed on the screen when a country name is given.
Sample Input	Enter country name: **Alfa**
Sample Output	**"Enter correct country name"**
Expected Output	Error message need to be displayed **"Enter correct country name"**

Test case 6	Giving a bird name to display the lifespan of that bird
Test requirement	To check whether the lifespan of the bird is being displayed.
Input description	The lifespan of the bird should be printed on the screen when a bird name is given.
Sample Input	Enter bird name: **Sandhill crane**
Sample Output	The lifespan of the bird is as follows:  **LIFE SPAN : 18 years, 6 months**
Expected Output	The life span of the bird need to be displayed. The life span of the bird is as follows:  **LIFE SPAN : 18 years, 6 months**

Test case 7	Giving a invalid bird name to display the lifespan of that bird
Test requirement	To check whether the error message is being displayed.
Input description	The lifespan of the bird should be printed on the screen when a bird name is given.
Sample Input	Enter bird name: **Sand**
Sample Output	Displays **"Enter correct bird name"**
Expected Output	The error message need to be displayed Displays **"Enter correct bird name"**

7. CONCLUSION AND FUTURE WORK

7.1. Conclusion:

The main idea behind developing the identification website is to build awareness regarding bird-watching, bird and their identification, especially birds found in India. It also caters to the need of simplifying the bird identification process and thus making bird-watching easier. The technology used in the experimental setup is Convolutional Neural Networks (CNN). It uses feature extraction for image recognition. The method used is good enough to extract features and classify images

The main purpose of the project is to identify the bird species from an image given as input by the user and we can also identify the different characteristics of a Bird. We used CNN because it is suitable for implementing advanced algorithms and gives good numerical precision accuracy.

7.2. Future Work:

1) Create an android/iOS app instead of website which will be more convenient to user.
2) System can be implemented using cloud which can store large amount of data for comparison and provide high computing power for processing (in case of Neural Networks)

8. BIBLIOGRAPHY

For Book References:

[1] Dipta Das, Sourya & Kumar, Akash "An Exploration of Computer Vision Techniques for Bird Species Classification", December 15, 2017.

[2] U.D.Nadimpalli, R.R.Price, S.G.Hall, and P.Bomma," A Comparison of image processing techniques for bird recognition", Biotechnology Progress, Vol. 22, no. 1,pp. 9-13,2006.

For Website References :

https://ieeexplore.ieee.org/document/9077750

https://www.researchgate.net/publication/348448453_Image_based_Bird_Species_Identification

9.APPENDIX

9.1 Introduction to Machine Learning

Machine Learning is the most popular technique of predicting or classifying information to help people in making necessary decisions. Machine Learning algorithms are trained over instances or examples through which they learn from past experiences and analyse the historical data.

Simply building models is not enough. You must also optimize and tune the model appropriately so that it provides you with accurate results. Optimization techniques involve tuning the hyperparameters to reach an optimum result.

As it trains over the examples, again and again, it is able to identify patterns in order to make decisions more accurately. Whenever any new input is introduced to the ML model, it applies its learned patterns over the new data to make future predictions. Based on the final accuracy, one can optimize their models using various standardized approaches. In this way, Machine Learning model learns to adapt to new examples andproduce better results.

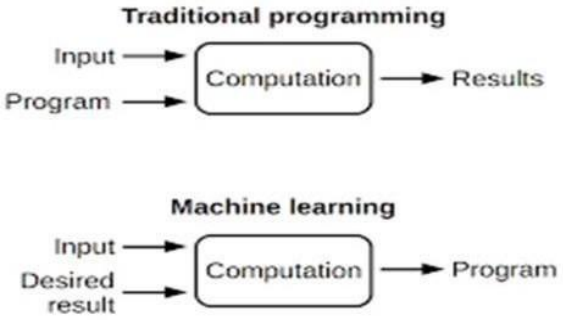


Fig 1. Machine Learning vs Traditional Programming

Types of Learnings

Machine Learning Algorithms can be classified into 3 types as follows:

1. Supervised learning
2.Unsupervised Learning
3.Reinforcement Learning

Supervised Learning

Supervised learning is the most popular paradigm for machine learning. It is the easiest to understand and the simplest to implement. It is the task of learning a function that maps an input to an output based on example input-output pairs. It infers a function from labelled training data consisting of a set of training examples. In supervised learning, each example is a pair consisting of an input object (typically a vector) and a desired output value (also called the supervisory signal). A supervised learning algorithm analyses the training data and produces an inferred function, which can be used for mapping new examples. Supervised Learning is very similar to teaching a child with the given data and that data is in the form of examples with labels, we can feed a learning algorithm with these example-label pairs one by one, allowing the algorithm to predict the right answer or not.

Most of the practical machine learning uses supervised learning. Supervised learning is where you have input variable (x) and an output variable (Y) and you use an algorithm to learn the mapping function from the input to the output.

$$Y = f(x)$$

The goal is to approximate the mapping function so well that when you have new input data (x) that you can predict the output variables (Y) for the data. It is called supervised learning because the process of an algorithm learning from the training dataset can be thought of as a teacher supervising the learning process

This is the learning type that you will most likely encounter, as it is exhibited in many of the common applications like Advertisement Popularity, Spam Classification, face recognition.

Two types of Supervised Learning are:

1. **Regression:**

Regression models a target prediction value based on independent variables. It is mostly used for finding out the relationship between variables and forecasting. Regression can be used to estimate/ predict continuous values (Real valued output). For example, given a picture ofa person then we have to predict the age on the basis of the given picture.

2. **Classification:**

Classification means to group the output into a class. If the data is discrete or categorical then it is a classification problem. For example, given data about the sizes of houses in the real estate market, making our output about whether the house "sells for more or less than the asking price" i.e. Classifying houses into two discrete categories.

Unsupervised Learning

Unsupervised Learning is a machine learning technique, where you do not need to supervise the model. Instead, you need to allow the model to work on its own to discover information. It mainly deals with the unlabelled data and looks for previously undetected patterns in a data set with no pre-existing labels and with a minimum of human supervision. In contrast to supervised learning that usually makes use of human- labelled data, unsupervised learning, also known as self-organization, allows for modelling of probability densities over inputs.

Unsupervised learning problems are classified into two categories of algorithms:

- **Clustering**: A clustering problem is where you want to discover the inherent groupings in the data, such as grouping customers by purchasing behaviour.

- **Association**: An association rule learning problem is where you want to discoverrules that describe large portions of your data, such as people that buy X also tend to buy Y.

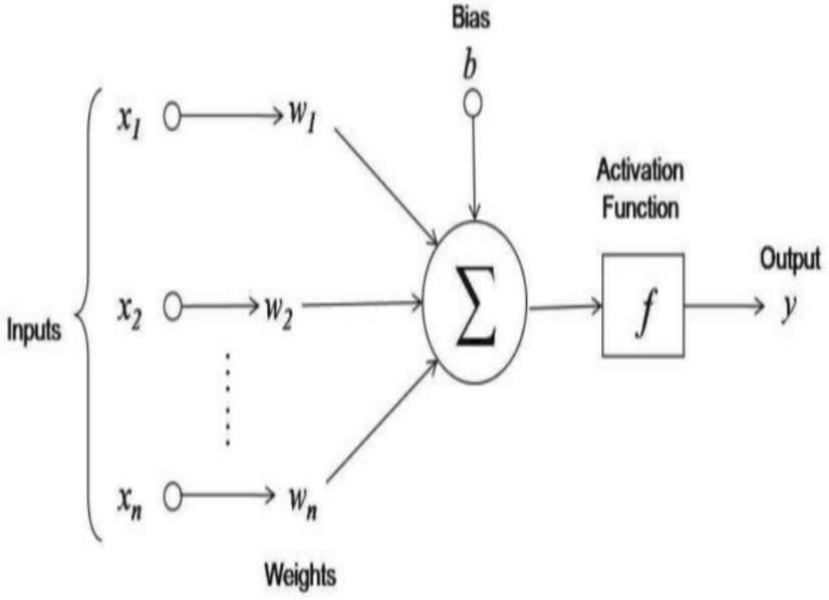
Reinforcement Learning

Reinforcement Learning (RL) is a type of machine learning technique that enables an agent to learn in an interactive environment by trial and error using feedback from its own actions and experiences. Machine mainly learns from past experiences and tries to perform best possible solution to a certain problem. It is the training of machine learning models to make a sequence of decisions. Though both supervised and reinforcement learning use mapping between input and output, unlike supervised learning where the feedback provided to the agent is correct set of actions for performing a task, reinforcement learning uses rewards and punishments as signals for positive and negative behaviour. Reinforcement learning is currently the most effective way to hint machine’s creativity.

9.2 Introduction to Neural Networks

Neural Network (or Artificial Neural Network) has the ability to learn by examples. ANN is an information processing model inspired by the biological neuron system. ANN biologically inspired simulations that are performed on the computer to do a certain specific set of tasks like clustering, classification, pattern recognition etc. It is composed of a large number of highly interconnected processing elements known as the neuron to solve problems. It follows the non-linear path and process information in parallel throughout the nodes. A neural network is a complex adaptive system. Adaptive means it has the ability to change its internal structure by adjusting weights of inputs.

Artificial Neural Networks can be best viewed as weighted directed graphs, where the nodes are formed by the artificial neurons and the connection between the neuron outputs and neuron inputs can be represented by the directed edges with weights. The ANN receives the input signal from the external world in the form of a pattern and image in the form of a vector. These inputs are then mathematically designated by the notations x(n) for every n number of inputs. Each of the input is then multiplied by its corresponding weights (these weights are the details used by the artificial neural networks to solve a certain problem). These weights typically represent the strength of the interconnection amongst neurons inside the artificial neural network. All the weighted inputs are summed up inside the computing unit (yet another artificial neuron).

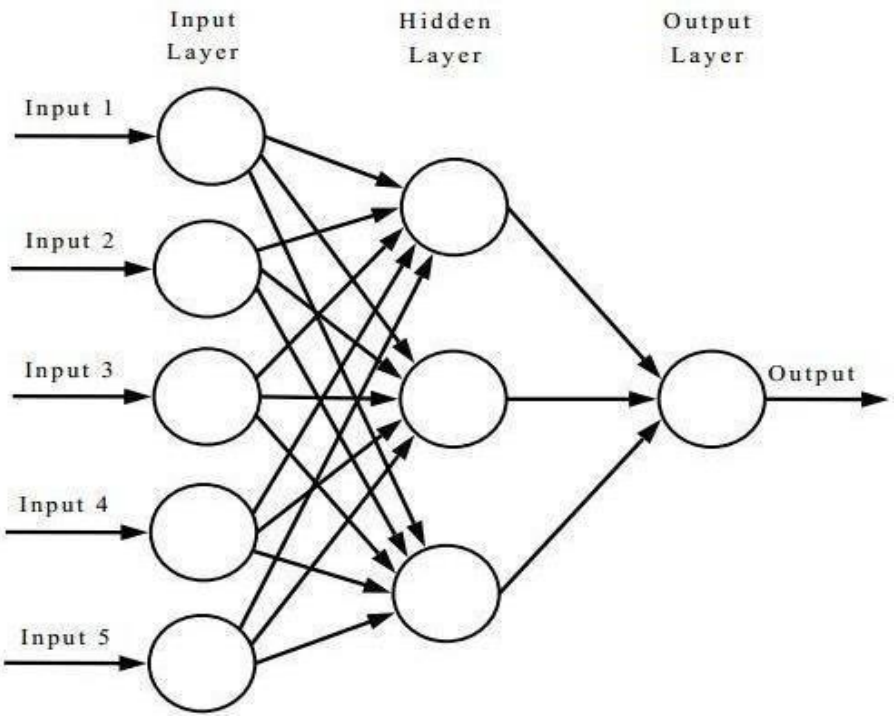


The Artificial Neural Network contains three layers

1. **Input Layer:** The input layers contain those artificial neurons (termed as units) which are to receive input from the outside world. This is where the actual learning on the network happens or corresponding happens else it will process.

2. **Hidden Layer:** The hidden layers are mentioned hidden in between input and the output layers. The only job of a hidden layer is to transform the input into something meaningful that the output layer/unit can use in some way. Most of the artificial neural networks are all interconnected, which means that each of the hidden layers is individually connected to the neurons in its input layer and also to its output layer leaving nothing to hang in the air. This makes it possible for a complete learning process and also learning occurs to the maximum when the weights inside the artificial neural network get updated after each iteration.

3. **Output Layer:** The output layers contain units that respond to the information that is fed into the system and also whether it learned any task or not.



Learning process of a neural network:

1. Start with values (often random) for the network parameters (*wij* weights and *bj* biases).

2. Take a set of examples of input data and pass them through the network to obtain their prediction.

3. Compare these predictions obtained with the values of expected labels and calculate the loss with them.

4. Perform the backpropagation in order to propagate this loss to each and every one of the parameters that make up the model of the neural network.

5. Use this propagated information to update the parameters of the neural network with the gradient descent in a way that the total loss is reduced, and a better model is obtained.

6. Continue iterating in the previous steps until we consider that we have a good model**.**

Deep Learning

Deep learning is a branch of machine learning which is completely based on artificial neural networks. Deep learning is an artificial intelligence function that imitates the workings of the human brain in processing data and creating patterns for use in decision making. Deep learning is a subset of machine learning in artificial intelligence (AI) that has networks capable of learning unsupervised from data that is unstructured or unlabelled. It has a greater number of hidden layers and known as deep neural learning or deep neural network.

Deep learning has evolved hand-in-hand with the digital era, which has brought about an explosion of data in all forms and from every region of the world. This data, known simply as big data, is drawn from sources like social media, internet search engines, e-commerce platforms, and online cinemas, among others. This enormous amount of data is readily accessible and can be shared through fintech applications like cloud computing. However, the data, which normally is unstructured, is so vast that it could take decades for humans to comprehend it and extract relevant information. Companies realize the incredible potential that can result from unravelling this wealth of information and are increasingly adapting to AI systemsfor automated support.

Deep Neural Network

It is a neural network with a certain level of complexity (having multiple hidden layers in between input and output layers). They are capable of modelling and processingnon-linear relationships.

9.3 Introduction to Streamlit:

Streamlit is an open-source python library for creating and sharing web apps for data science and machine learning projects. The library can help you create and deploy your data science solution in a few minutes with a few lines of code.

- For deploying the streamlit the following command is to be used :
  pip install streamlit

- For Title
  ```
  st.title("Title")
  ```

- For Header & Sub-Header
  st.header("This is a header")
  st.subheader("This is a subheader")

- For Writing onto the Interface
  ```
  st.write("Text with write")
  ```
- For displaying Images
  from PIL import Image
  img = Image.open("streamlit.png")
  ```
  st.image (img, width=200)
  ```

9.4 Introduction to Python

Python is an open source, high-level programming language developed by Guido van Rossum in the late 1980s and presently administered by Python Software Foundation. It came from the ABC language that he helped create early on in his career. Python is a powerful language that you can use to create games, write GUIs, and develop web applications.

It is a high-level language. Reading and writing codes in Python is much like reading and writing regular English statements. Because they are not written in machine-readable language, Python programs need to be processed before machines can run them. Python is an interpreted language. This means that every time a program is run, its interpreter runs through the code and translates it into machine readable byte code. Python is an object-oriented language that allows users to manage and control data structures or objects to create and run programs. Everything in Python is, in fact, first class. All objects, data types, functions, methods, and classes take equal position in Python.

Programming languages are created to satisfy the needs of programmers and users for an effective tool to develop applications that impact lives, lifestyles, economy, and society. They help make lives better by increasing productivity, enhancing communication, and improving efficiency. Languages die and become obsolete when they fail to live up to expectations and are replaced and superseded by languages that are more powerful. Python is a programming language that has stood the test of time and has remained relevant across industries and businesses and among programmers, and individual users

Readability

Python programs use clear, simple, and concise instructions that are easy to read even by those who have no substantial programming background. Programs written in Python are, therefore, easier to maintain, debug, or enhance.

Higher productivity

Codes used in Python are considerably shorter, simpler, and less verbose than other high-level programming languages such as Java and C++. In addition, it has well- designed built-in features and standard library as well as access to third party modules and source libraries.These features make programming in Python more efficient.

Less learning time

Python is relatively easy to learn. Many find Python a good first language for learning programming because it uses simple syntax and shorter codes. Python works on Windows, Linux/UNIX, Mac OS X, other operating systems and small form devices. It also runs on microcontrollers used in appliances, toys, remote controls, embedded devices, and other similar devices.

Installing Python in Windows

To install Python, you must first download the installation package of your preferred version from this link: https://www.python.org/downloads/ On this page, you will be asked to choose between the two latest versions for Python 2 and 3: Python 3.5.1 and Python 2.7.11.

Alternatively, if you are looking for a specific release, you can scroll down the page to find download links for earlier versions. You would normally opt to download the latest version, which is Python 3.5.1. This was released on December 7, 2015. However, you may opt for the latest version of Python 2, 2.7.11.

Your preferences will usually depend on which version will be most usable for your project. While Python 3 is the present and future of the language, issues such as third-party utility or compatibility may require you to download Python2

Jupyter Notebook

The Jupyter Notebook is an open source web application that you can use to create and share documents that contain live code, equations, visualizations, and text. Jupyter Notebook is maintained by the people at Project Jupyter. Jupyter Notebooks are a spin-off project from the IPython project, which used to have an IPython Notebook project itself. The name, Jupyter, comes from the core supported programming languages that it supports: Julia, Python, and R.

Jupyter ships with the IPython kernel, which allows you to write your programs in Python, but there are currently over 100 other kernels that you can also use. The Jupyter Notebook is not included with Python, so if you want to try it out, you will need to install Jupyter. There are many distributions of the Python language.