

Visvesvaraya Technological University

Jnana Sangama, Belagavi - 590018



A Project Report on

“Dog Breed Identification with Product Classification and Assisted Services”

Project Report submitted in partial fulfilment of the requirement for the award of the degree of

BACHELOR OF ENGINEERING

IN

COMPUTER SCIENCE AND ENGINEERING

Submitted by

RUCHITHA G K	1KS17CS066
SPOORTHI R	1KS17CS081
VARUN ATTIGANAL VENKATESH	1KS17CS096

Under the guidance of

Dr. Swathi K

Assistant Professor

Department of Computer Science & Engineering

K.S.I.T, Bengaluru-560109



KSIT
K. S. INSTITUTE OF TECHNOLOGY

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

K. S. Institute of Technology

#14, Raghuvanahalli, Kanakapura Road, Bengaluru - 560109

2020 - 2021

K. S. Institute of Technology

#14, Raghuvanahalli, Kanakapura Road, Bengaluru - 560109

Department of Computer Science & Engineering



KSIT
K.S. INSTITUTE OF TECHNOLOGY

CERTIFICATE

Certified that the project work entitled “Dog Breed Identification with Product Classification and Assisted Services” is a bonafide work carried out by:

RUCHITHA G K

1KS17CS066

SPOORTHI R

1KS17CS081

VARUN ATTIGANAL VENKATESH

1KS17CS096

in partial fulfilment for VIII semester B.E., Project Work in the branch of Computer Science and Engineering prescribed by Visvesvaraya Technological University, Belagavi during the period of April 2021 to July 2021. It is certified that all the corrections and suggestions indicated for internal assessment have been incorporated in the report deposited in the department library. The Project Report has been approved as it satisfies the academic requirements in report of project work prescribed for the Bachelor of Engineering degree.

.....
Signature of the Guide

Dr. Swathi K
Assistant Professor

.....
Signature of the HOD

Dr. Rekha B. Venkatapur

.....
**Signature of the Principal/
Director**

Dr. DILIP KUMAR K

External Viva

Name of the Examiners

Signature with date

1.

2.

DECLARATION

We, the undersigned students of 8th semester, Computer Science & Engineering, KSIT, declare that our project work entitled “**Dog Breed Identification with Product Classification and Assisted Services**”, is a bonafide work of ours. Our project is neither a copy nor by means a modification of any other engineering project.

We also declare that this project was not entitled for submission to any other university in the past and shall remain the only submission made and will not be submitted by us to any other university in the future.

Place: *Banglore*
Date: *23/7/2021*

Name and USN

RUCHITHA G K (1KS17CS066)

Signature

Ruchita

SPOORTHI R (1KS17CS081)

Spoorthi

VARUN ATTIGANAL VENKATESH (1KS17CS096)

Varun

ACKNOWLEDGEMENT

The satisfaction and euphoria that accompany the successful completion of any task will be incomplete without the mention of the individuals, we are greatly indebted to, who through guidance and providing facilities have served as a beacon of light and crowned our efforts with success.

First and foremost, our sincere prayer goes to almighty, whose grace made us realize our objective and conceive this project. We take pleasure in expressing our profound sense of gratitude to our parents for helping us complete our project work successfully.

We take this opportunity to express our sincere gratitude to our college **K.S. Institute of Technology**, Bengaluru for providing the environment to work on our project.

We would like to express our gratitude to our **MANAGEMENT**, K.S. Institute of Technology, Bengaluru, for providing a very good infrastructure and all the kindness forwarded to us in carrying out this project work in college.

We would like to express our gratitude to **Dr. K.V.A Balaji**, CEO, K.S. Institute of Technology, Bengaluru, for his valuable guidance.

We would like to express our gratitude to **Dr. Dilip Kumar K**, Principal/Director, K.S. Institute of Technology, Bengaluru, for his continuous support.

We like to extend our gratitude to **Dr. Rekha. B. Venkatapur**, Professor and Head, Department of Computer Science & Engineering, for providing a very good facilities and all the support forwarded to us in carrying out this project work successfully.

We also like to thank our Project Coordinators, **Mr. K Venkata Rao**, Associate Professor, **Dr. Vaneeta M**, Associate Professor, **Mr. Raghavendrchar S**, Asst. Professor, **Dr. Aditya Pai H**, Asst. Professor, and **Mrs. Sneha K**, Asst. Professor, Department of Computer Science & Engineering for their help and support provided to carry out the project and complete it successfully.

Also, we are thankful to **Dr. Swathi K**, Assistant Professor, for being our Project Guide, under whose able guidance this project work has been carried out and completed successfully.

We are also thankful to the teaching and non-teaching staff of Computer Science & Engineering, KSIT for helping us in completing this project work.

RUCHITHA GK
SPOORTHI R
VARUN ATTIGANAL VENKATESH

ABSTRACT

Owning a furry companion can have many positive effects on a person's life. Not only do they get to live with a friend, but there are also some hidden benefits to keeping a pet in home. There are many health benefits of owning a pet. They can increase opportunities to exercise, get outside, and socialize. Regular walking or playing with pets can decrease blood pressure, cholesterol levels, and triglyceride levels. Pets can help manage loneliness and depression by giving us companionship. Most households in the United States have at least one pet. Studies have shown that the bond between people and their pets can increase fitness, lower stress, and bring happiness to their owners. Some of the health benefits of having a pet include: decreased blood pressure, decreased cholesterol levels, decreased triglyceride levels, decreased feelings of loneliness, increased opportunities for exercise and outdoor activities, increased opportunities for socialization.

And therefore the demand for having a pet at home has been increasing progressively for the past few years.

In order to help the pet owners we aim to develop a web application to provide the facility to identify the dog breed, categorize products and also a lost portal to help find lost dogs.

Keywords: Classification, Machine Learning, Pets

TABLE OF CONTENTS

Chapter No.	Title	Page No.
1.	INTRODUCTION	01
1.1	Overview	01
1.2	Purpose of the project	01
1.3	Scope of the project	02
1.4	Definitions	03
2.	LITERATURE SURVEY	06
2.1	Real-Time Object Detection with YOLO	04
2.2	You Only Look Once: Unifies, Real-Time Object Detection	05
2.3	Rethinking the Inception Architecture for Computer Vision	10
3.	PROBLEM STATEMENT	07
4.	SYSTEM REQUIREMENT SPECIFICATION	09
4.1	Hardware Requirements	09
4.2	Software Requirements	09
4.2.1	Python	10
4.2.2	Database	10
4.2.3	HTML and CSS	11
4.2.4	React	11
4.3	Non-Functional Requirements	11
5.	DESIGN	13

4.1	System Architecture	13
4.2	Model Architecture	16
6.	IMPLEMENTATION	18
6.1	Selecting a Programming Language	18
6.1.1	Visual Studio Code	19
6.1.2	Flexibility	19
6.1.3	Production	20
6.1.4	Performance	20
6.1.5	Support	20
6.2	Implementation of YOLO Module	21
6.3	Implementation of Classifier Module	23
6.4	Connectivity	28
6.5	Lost Portal	30
6.6	Product Categorization	31
7.	TESTING AND RESULTS	33
7.1	Unit Testing	33
7.2	Integration Testing	33
7.3	Functional Testing	34
7.4	White Box Testing and Black Box Testing	34
7.5	Testing the Breed Detector	34
8.	SNAPSHOTS	36
9.	CONCLUSION AND FUTURE ENCHANCEMENTS	39
	REFERENCES	40

LIST OF FIGURES

Fig. No.	Figure Name	Page No.
4.1	System Architecture	14
4.2	Model Architecture	16
6.2.1	Working of YOLO	21
6.2.2	Bounding Boxes and Grids	22
6.3.1	Feature Map	25
6.3.2	Deploying Inception v3	26
6.3.3	Freezing Classification Layer	26
6.3.4	Implementing Classifier	26
6.3.5	History of Loses and Accuracy	28
6.4	Connectivity	29
6.5	Lost Portal	30
6.6	Product Categorization	32
8.1	Dog Breed Identification	36
8.2	Lost Portal	36
8.3	Login Part 1	37
8.4	Login Part 2	37
8.5	Product Categorization	39

Chapter 1

INTRODUCTION

1.1 Overview

Owning a furry companion can have many positive effects on a person's life. Not only do they get to live with a friend, but there are also some hidden benefits to keeping a pet in home. There are many health benefits of owning a pet. They can increase opportunities to exercise, get outside, and socialize. Regular walking or playing with pets can decrease blood pressure, cholesterol levels, and triglyceride levels. Pets can help manage loneliness and depression by giving us companionship. Most households in the United States have at least one pet. Studies have shown that the bond between people and their pets can increase fitness, lower stress, and bring happiness to their owners. Some of the health benefits of having a pet include: decreased blood pressure, decreased cholesterol levels, decreased triglyceride levels, decreased feelings of loneliness, increased opportunities for exercise and outdoor activities, increased opportunities for socialization.

And therefore the demand for having a pet at home has been increasing progressively for the past few years. The population of pets was around 18 billion in the year 2018 and is said to reach over 30 billion by the year 2023. This increase in the population of pets has also automatically increased the value of the consumer's market. Over 210 million dollars are spent on pets in a country. With huge demand for pets and the necessary to have pets, our project holds the above discussed topics as a base to create something truly amazing for a pet owner.

1.2 Purpose of the Project

With the increase in the number of pets, pet owners look for some options that are really best for their pets. The first thing that pet owners need and consider very important is having a sound knowledge of the breed of the pet. This is a very important step because if pet owners need to provide the best for their tiny little companions, it's very necessary to know the breed. Wrong food/accessories can harm the pet in many ways. So we are aiming to provide a platform to our users where the users will be able to upload a picture of their pet, and our Machine Learning model will predict the breed for the users. It is to be noted that, in our project we are

aiming to detect only the breeds of various dogs. On uploading the picture of the dog, our system runs the Machine Learning Algorithms on the backend and returns the breed of the dog along with the accuracy count. This way, if at all any anytime the user is confused about the dog breed, they can refer our dog breed detection system.

Like it was discussed earlier, in our system we also provide a platform where we tell what products are good for the pets. All the products that are suitable for a particular breed are displayed to the user. This will be helpful because some users struggle regarding which product have to be selected for their pets and also wonder whatever products they are choose thing is right or wrong. Our product categorization system makes this easier for the user because we suggest what products or accessories could be good for a certain breed of dog. Thereby solving the major issue of confusion among pet uses. The pet owners could just look up a product categorization page and they will be able to find breeds of dogs along with the product categories according to the breed. Therefore we also aim to make the life of a pet owner quite easy.

Many pet owners usually lose their dogs. It might be when they are watching them or when the door is unlocked, some pets usually go away from the owner. Since this is also a major issue we are helping the pet owners solve this by providing something called as a lost and found portal only for the pets. In this, the users usually upload the picture of their Lost dog and any person who comes in contact with our system and if they have seen their dogs can contact the owner directly telling that they have found a pet therefore again making it very easier for the pet owners to find their pet. This is one of the assisted services that we are providing to our pet owners.

The purpose of a project is to make the life of a pet owner very easier by providing a complete understanding of their pet.

1.3 Scope of the Project

The scope of the project is as follows:

- 1) Provide the users with a web application that can detect up to 120 breeds of dogs.
- 2) This feature helps any dog owner to recognize the breed of the dog, which helps the dog owners to recognize their dogs better.

- 3) Product Categorization provides guidance and suggestions so that dog owners never go wrong in buying food or accessories to their beloved pets.
- 4) Lost portal enables users to upload a picture of the dog whenever their dog is lost.
- 5) We are aiming to make this centralized, all under one roof so that dog owners will not have to navigate multiple websites in order to find everything that they need for their pets.
- 6) Overall, pet owners will get a better understanding about their pets which is our main goal and also objective.

1.4 Definitions

1.4.1 Python

Python is an interpreted high-level programming language and also object-oriented. Python is used for many application programming. Python features a dynamic type system in automatic memory management and supports multiple Programming Paradigms object oriented, functional programming, and procedural styles. Python is also used by many machine learning algorithms. We have also used Python in order to implement machine learning algorithms and also other programs that are required to develop a complete web application.

1.4.2 Database

A database is used in order to store many information about the pets which will later help us when we have to deal with product categorization. A database in simple words means that it is nothing but a collection of data. In our case, the collection of data would be different types of foods and accessories related to a particular type of breed. We have used MongoDB as a database. This manages data and allows fast storage and retrieval of the data.

Chapter 2

LITERATURE SURVEY

2.1 Real-Time Object Detection with Yolo

Real time object detection with YOLO was written by Geethapriya S, N. Duraimurugan, and S.P. Chokkalingam. The objective of this paper is to detect any object in real time using an algorithm called You Only Look Once (YOLO). The authors talk about how this algorithm is more efficient than most of the machine learning algorithms that are used to identify in objects. Object detection is a method or technology that is used to detect in object of any classes in digital images and videos. The most common object that is easily detected using this algorithm is a car a motorcycle or a human. This algorithm is not limited to only these objects. It can also be used to identify many animals such as dog cat bird lion, etc. locating objects in the image something called as object localization is used. Convolutional neural network and recurrent neural network usually fall into classification algorithm category. These methods are very slow because a prediction for every selected region has to be run thus making is very slow. There are other categories of algorithms that are based on regressions. YOLO comes under this category. This algorithm is fast as compared to other classification algorithms. in real time this algorithm processes 45 frame per second thus making localization errors but predicts less false positives in the background. A complete approach to how to use the algorithm and how the algorithm functions in described in this paper. There is also a brief description about how the images is identified, divided into different grids, bounding boxes and matrixes in order to identify an object in real time. From this paper we are using the concept of real time object detection using YOLO.

2.2 You Only Look Once: Unified, Real-Time Object Detection

Look once: United, real time object detection was a whole new approach to object detection. Object detection in this paper is termed as a regression problem to spatially separated by bounding boxes in associated class probabilities. This architecture of object detection change the whole view of object detection using machine learning algorithms. This paper was written by Joseph Redmon, Santosh Divvala, Ross Girshick and Ali Farhadi. YOLO algorithm can basically

detect any type of object. There is no restriction to what kind of object has to be detected. This algorithm gave a whole new approach to customized object detection using machine learning algorithm which will deeply be used in a project. In this paper a complete explanation as to how images work, how pixels are divided, how YOLO works are explained very well. These authors define a unified architecture for the algorithm. Since the object detection pipeline is usually a single network, end-to-end directly on detection performance. This unified architecture that was defined by the authors of the paper is extremely fast and efficient. This model processes images in real time at 45 frames per second. A smaller version of this network is termed as fast YOLO processes about 155 frames per second while achieving double map of either real-time detectors. This is very astonishing. Which is the fact that it can process 155 frames per second is something unimaginable whenever object detection has to be used. This kind of speed and efficiency cannot be achieved by Convolutional neural networks. YOLO launch very general representations of the objects. Freedom makes more localization errors but is far less likely to predict false detections where nothing exists. This algorithm provides a model that is very simple to construct and can be trained directly on full images. Unlike classifier based approaches this algorithm is trained on a loss function that directly corresponds to detection performance and entire model is trained jointly that is together. This provided a whole new approach for object detection which made this entire concept much easier. Algorithm is also very fast accurate making it very ideal for computer vision applications. YOLO is known to you work very well with images it could be a real time image separate image.

We are particularly using this algorithm because we want our system to detect only dogs. This enables the classifier system to function more efficiently and accurately. This algorithm limits the images of only dogs to be passed into the classifier system. The entire functioning of YOLO, implementation and walking is explained in the further part of this report.

2.3 Rethinking the Inception Architecture for Computer Vision

Convolutional neural networks are the most important part of computer vision solutions for a wide variety of tasks. The development of Convolutional neural networks is not very easy and requires high level of programming in order to achieve the same. Rethinking the Inception Architecture for Computer Vision was a paper that was written by Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jonathan Shlens. This part of literature survey explains how a pre

trained model called Inception can is the task of implementing convolutional neural network in machine learning problems. Option is nothing but a pre trained model that can offer convolutional neural networks at a lower price. Implementing this free train model is very simple and any developer does not have to go through immense coding in order to use object detection. We have also implemented Inception pre trained model to implement convolutional neural network in our project. Father we have freeze the classification layer of this particular pertained model in order to deploy our own classifier system. This is also explain for the in this report.

The above data explains the literature survey of our project. These were the most important literature survey that had to be conducted in order to understand completely how our project is going to work. Literature survey has helped us understand completely as to what aspect are required in order to develop a very efficient project.

Chapter 3

PROBLEM STATEMENT

Every pet owner has one important duty, which is to provide the best food and accessories to their pets. A lot of online websites tell the users what they need to provide to their pets. But what they also lack is making this product breed specific. A lot of websites that exist currently are only specific to one feature. This feature could be either breed identification our product categorization or providing lost and found portal. Users who usually use these websites spend a lot of time navigating from one page to another in order to find the right feature that they are looking for. This can be very time consuming. This is also quite exhausting because it takes a lot of time and effort by the pet owner to find what they are looking for on the internet. We all also know that internet is quite vast and it is not easy to find anything quickly. If one website provides the information of food for pets then maybe on searching a lot on the Internet the fifth website would provide the accessories that are suitable for the pets. If we were to think about how much time and effort is invested in this then it would definitely be a lot. Thus we have come up with an idea which solves the main problem of time consumption along with providing the pet owners what exactly they are looking for. We want to provide the pet owners what is best for their pets. Our application is very user friendly and has a unique feature, which is it is centralized meaning everything is present right under one roof. Therefore we can define a problem statement as follows:

“To provide a centralized platform to the users where the breed of a dog is identified, products pertaining to the breed are categorized and assisted service is also given to the user.”

The very important concepts that a pet owner would look into all the following:

- To identify the breed of the dog
- To have a sound knowledge about the breed of the dog
- To make sure that they are buying the right food for their dogs

- To also ensure that they buy the right accessory for their dogs so that no kind of harm is done to the pet

The above factors by default are the main topics that are basically discussed and worried about by the pet owners. Our project is only limited to identify the breed of the dog and also perform other features based on the breed of the dog. We are aiming to provide all the dog owners with the above discussed features. This centralized platform that we are creating especially for pet owners will definitely create a significant impact. The whole problem of spending too much time looking for the right product, surfing throughout the internet to look for one tiny bit of information can be now reduced by implementing a web application. What makes our project stand out from the rest of the crowd is that, we are providing the users all the features all under one roof.

Chapter 4

SYSTEM REQUIREMENT SPECIFICATION

System requirement specification is a description of the purpose that is intended and the environment that has to be developed for the software that is under construction. System requirement specification describes what the function of the software is and how a user expects the software to function. Without system requirement specification the functioning of software will be very difficult to understand. This specification helps in the assessment of all the requirements that need to be present before the software can be designed. I should also provide a very realistic and a true basis for estimating the cost, risk, and any other factors of software. The software requirement specification is a whole process of documentation that basically enlists all the necessary requirements that are very much required in order to ensure a successful project development. In order to understand what these requirements are, we need to have a clear and a complete understanding of the software has to be developed. This is this verified, and communicated with the entire project team and uses until and unless the development of the software is complete. There are two main types of requirements specifications namely, hardware requirements and software requirements.

4.1 Hardware Specifications

The following are the hardware specifications that are required:

- A processor: Intel second generation and above
- RAM: 4GB and above
- Hard disk space: At least 500gb, preferably above that

4.2 Software Specifications

The following are the software requirements:

- Programming language: Python 3 and above

- Database: MongoDB
- Operating system: Windows or Ubuntu
- Front end Technologies: HTML, CSS, React, MaterialUI

4.2.1 Python

Python is an interpreted high-level programming language and also object-oriented. Python is used for many application programming. Python features a dynamic type system in automatic memory management and supports multiple Programming Paradigms object oriented, functional programming, and procedural styles. Python is also used by many machine learning algorithms. We have also used Python in order to implement machine learning algorithms and also other programs that are required to develop a complete web application. Python can be used for many applications success web development, machine learning, operating system, mobile application development, video games, etc. Python is a free programming language and is also completely open source. It is also one of the most accessible programming languages that have said to exist. The simplicity in the syntax of this programming language is very similar to that of English. There are not many rules and regulations that had to be followed in Python. Compare to other languages python does not impose any syntax or semantic rules while programming. Python is also faster than any other programming language. The main reasons as to why Python is preferred mainly when dealing with machine learning.

4.2.2 Database

A database is used in order to store many information about the pets which will later help us when we have to deal with product categorization. A database in simple words means that it is nothing but a collection of data. In our case, the collection of data would be different types of foods and accessories related to a particular type of breed. We have used MongoDB as a database. This manages data and allows fast storage and retrieval of the data.

MangoDB is a cross platform SQL database program. It is a document oriented database program. Latest licensed under Server-Side Public License. This no SQL database offers very fast query processing. This model is a very powerful way to store and retrieve the data quite fast. MangoDB also offers a great user experience for the developers. It is exceptionally very easy to add or change the fields in this database.

4.2.3 HTML and CSS

HTML stands for hypertext markup language. This programming language is basically used for or development of web pages. Very simple language and it does not follow very tough rules. The entire structure of the webpage can be easily designed by the use of this language. Furthermore CSS which stands for cascading style sheet is used to design the web pages. Since HTML does not offer any design options this CSS is used to style the web pages as and how the user wishes. HTML is still the main building block of all the websites. The simplicity of this language makes it very likeable.

4.2.4 React

React is an open source frontend JavaScript library. React is free library. Reactor is basically used for building user interfaces for UI components. React alerts the user usable UI components which is the best way to handle view layer for the web and mobile applications. This is one of the reasons as to why a react is considered the best. The purpose of React is to be fast, to be scalable, and to be quite simple. React is pure JavaScript.

4.3 Non-Functional Requirements

Apart from hardware and software requirements specification there is also something called as non functional requirements specification which is as important as hardware and software specification. This requirement specifies the criteria that can basically be used to judge the entire software that is being developed. It also focuses on certain specific behavior of the software that is under development. The non functional requirement should also be considered as important as the other requirements because to ensure a fully functioning software project all these things have to be considered. The non functional requirements specifications are as follows:

- Efficiency
- Maintainability
- Platform compatibility or platform independent
- Robust

- Reliability
- Scalability
- Quality
- Portability
- Testing
- Usability
- Extensibility

Even though while developing a software application these features might not be highlighted as much as how much Software and Hardware requirements should be stressed on. But toward the end of the completion of any software project these features have to be looked into in order to accept the software as a fully functioning product. Any user expects software to have the above discussed even though he does not specifically ask for it.

Therefore in order to complete an entire software project, not just the technical aspects but also the non technical aspects should be considered in order to make it the best product.

Chapter 5

DESIGN

Every software project or any project requires something called system design. System design is a very important part of software development. The design of any software or a system would define the components present in the system the working of the system the connections between the components of the system. This is very important because, in order to understand how an entire software works, this design is necessary for the developers to follow certain rules. System design is a process of defining the architecture and data, interfaces, components and data in order to fulfill the defined system requirements.

The importance of the system designing phase should be understood because it involves identifying the data sources, the nature of the data, and the type of data that is being made available in order to develop software. For example, in order to design a booking system, there will be a need of varying number of input such as customer details, hotel of choice, number of rooms to be booked, number of people staying in a room, location of the hotel, facilities provided by the hotel, etc. This facilitates an understanding of what kind of data will be made available to the system and what action it the supposed to take based on the data that is being made available by the system. This type of data we can also understand the various factors for the types of data that would fall under the hotel booking category. Therefore system designing becomes necessary.

4.1 System Architecture

System architecture is a conceptual design or model that basically define the structure of the model, behavior of the model and detail information about a particular system that has to be developed. And architecture description is a formal description and what is called as the representation of any software system. System architecture can have component that will work together in order to implement what most of the people look for that is an overall system. Since our project is also software based and also makes use of many system requirements we have also defined a system architecture for the same. Our system architecture includes various components

including a user face and give the clear Idea as to what exactly we want to deliver to the customer or uses. Figure 4.1 architecture of dog breed identification.

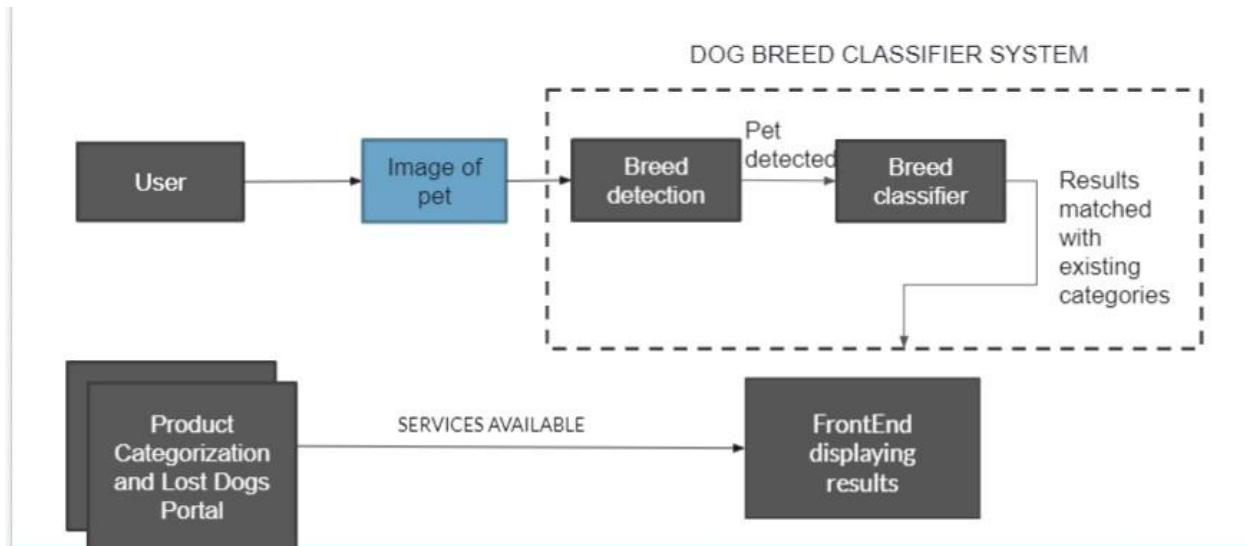


Fig 4.1: System Architecture

The architecture shown in figure 4.1 the following:

1. User input
2. Breed detection
3. Breed classifier
4. Product categorization
5. Lost portal
6. User interface

1. User Input

User phase is the first phase which is used to take input from the user. In our case the input that we will be asking the users to provide to our system would be a picture of a dog. The picture of the dog must be very clear and the user should also ensure that the user is uploading a picture that contains only one dog. After the picture of the dog has been successfully uploaded by the user our system takes it to the next days when the pet will be identified. On detecting the pet, our

system will take this in which forward to something called as dog breed classifier. Dog breed classifier has two main components that is explained below.

2. Breed Detection

Detection is the next important step of a project. Detection is important in order to understand which breed of the dog that we will have to identify. The breed detector takes the picture that has been uploaded and starts looking up for a similar breed. After identifying that the picture could be one of the breed that we are dealing with, the breed detector passes this image further onto the breed classifier where The Breed is clearly classified.

3. Breed Classifier

Breed classifier works on a machine learning algorithm in order to classify the breed. The Breed classifier predict the breed of the dog and also tell the probability score of of the dog breed matching with our classified breeds.

4. Product Categorization

We also aim on providing product categorization to the users to ease the task of choosing food or accessories for their pets. Product categorization is a portal where we suggest the pet uses what products can be used in order to make sure that they are making the right choice for their pet.

5. Lost Portal

The Lost portal allows the users to upload a picture of their dog and the description of the dog in case they have lost their pet. This portal is mainly being done as a social cause to help people find their lost pets.

6. User Interface

User interface is where the user will be interacting with our application that we are developing. The user will be able to access dog breed detector, product categorization and lost portal. The

only reason for providing the user interface is to make sure that the users can use the software and can also get their task done.

The above discussed concepts are regarding the system architecture, meaning a rough overview of what our project aims on doing and what we aim to provide to the user.

4.2. Model Architecture

Apart from the basic system architecture we have also have another architecture that is very much necessary for the dog breed protection in a project. This architecture has helped are successfully implement a dog breed identifier a clear description about out the architecture of the model implemented is shown below.

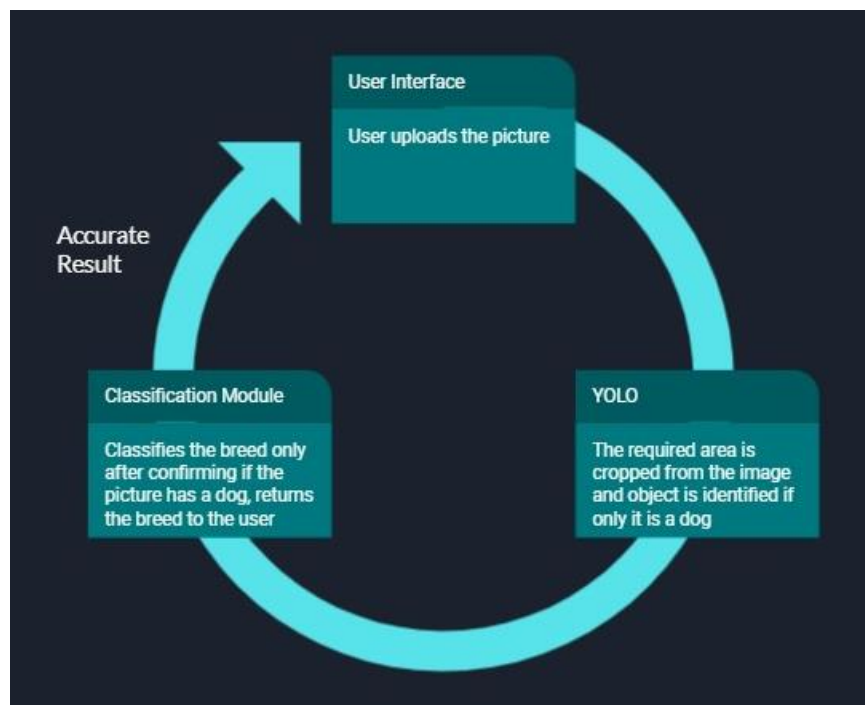


Fig 4.2 Model Architecture

The above diagram or figure show the model architecture that we have implemented in a project. This has three main components mainly, user interface, YOLO model, classifier model.

The user upload the picture of the dog. But there are some animals that resemble a dog. For example, Siberian wolf looks like a Husky. It would be very difficult for are classified as system to classify this use case as the breed of the dog. If user uploaded the picture of the Wolf and not the dog, the classify your would actually identify this wolf as a Husky. In this case we would get a wrong prediction. Therefore in a order to avoid this misconception we are implementing something called as the YOLO the model in the beginning. Therefore as soon as the user uploads the picture the picture is sent into the YOLO model where it checks if the uploaded picture is a dog. If the uploaded picture is definitely a dog then the picture is passed to the classifier system. If the uploaded picture is not a dog but any other animal then our system will return the picture to the user asking the user to upload the right picture which in our case is a dog. Therefore this model help in increasing the accuracy of a model. After the picture of the dog has been successfully passed to the classifier model then the classify the model will identify the breed of the talk which will be pretty much accurate since a long animal is not being passed into the classifier. The classifier has implemented machine learning algorithms and task will identify the breed of the dog that is present in the picture and return all most accurate result to the user. This would be the idea behind identifying The Breed of the dog.

Chapter 6

IMPLEMENTATION

When planning to implement a Software Solution we will have to choose different programming languages. It is very easy to get lost in the intricacies of each of the programming languages. The case of the programming language can depend on many factors but not one. If the project is for probably a personal hobby then we can basically settle down with any language that volatile aunt and that we find easy to work with. But if we have to develop a project that is used in large scale industry the language selection process becomes very crucial. We have to always make sure that we have to select two language with maximum efficiency which eases the process of development of the project.

6.1 Selecting a Programming Language

One factor that we have to think about when it comes to project. For example, c programming language is highly known and also acts as a base for many programming languages. Many of the syntax, semantics are derived from C language. But C language would not be a proficient choice in order to develop any machine learning project. Python is another object oriented programming language that is pretty straight forward and is also very simple to learn. Python also execute any program very quickly thereby saving the time complexity. It decreases the cost of program support. Factors such as performance and security and other factors such as a few line of code might be the first priority here. There will always be some kind of trade-offs when it comes to choosing a programming language. The choice of language is by far the most overlooked component at this point of preparation. The lines of the code and how fast the program can execute is important here because we want to get the solution first and then worry about neatness for performance of the program. Since we have to get the output first which means that the program must execute soon which is also directly proportional to the number of lines in the code. By selecting the right programming language the solutions that are produced are easy to debug, easy to extend, easy to fix and easy to document. Factors that have to be considered when selecting a programming languages are: The target platform, the time to production,

performance, flexibility of language, support for the language and the community for the language.

Python is very popular in machine learning because of many interrelated reasons. It is very simple, consistent, it has very less lines of code, flexible and Math like. Python code has been described as a readable pseudo code. It is very easy to pick up because of its consistent Syntax and its impact almost looks like sentences written in English language. Python mirrors human language and their mathematical counterparts. The mathematical counterparts is very important if we want to implement a machine learning algorithm of which the core would be just Optimization. Python provides all the necessary libraries to satisfy the above discussed details.

6.1.1 Visual Studio Code

Visual Studio code is the platform that we have selected. It is a portable code editor that includes features like debugging, task execution, etc. It attempts to provide developers only the tools that they need for fast code building the work cycle, leaving all the complicated processes to fully featured IDEs like visual Studio IDE. This platform is available for Mac OS, Windows and Linux. It is available for personal and both business usage. This leverage is internet services for a variety of functions, including downloading product update, locating, installing, and upgrading extensions along with natural language search within the settings editor. Visual Studio code provides with recommendations to finish lines of the code and fast solution for common errors as and when we start coding in this particular platform. This also has a debugger to go through each line of code and figure out what exactly going wrong in the code where it has to be corrected and so on. Therefore Visual Studio code would be the best choice.

6.1.2 Flexibility

Flexibility tells us how easy it is to add new features into an existing program. This may be obtained by creating a collection of functions or by modifying a present library to include new functionalities. Consider the questions:

- If potential available in a language library?

- If it's not a native function and not available in the library, what is the process to build functionalities from scratch?
- Before making a decision, user must know the decision of the program and what functionalities have been included in order to do any future enhancements.

6.1.3 Production

Production refers to the time taken for a program to go live that is when the code is ready for production and will perform as it is expected to perform. Servlets code may take months to master if at all the user is starting from scratch in both the languages.

6.1.4 Performance

Performance could be an important factor in any software development stage. Many research has been conducted to compare the speed of the programming languages in the same context. Give an ideal conditions, statistics in benchmark leads the user to believe that the program code in Java executes faster than python. But the simplicity that Python contains is not available in Java. Therefore the computing capacity will produce results if Python is used. Lines of code and maintainability will also be less thereby creating less complexities.

6.1.5 Support

Does excellent software requires a community to help it flourish, a language with an active forum is more likely to be popular than another language that lacks support. Python has been offered a lot of support from a lot of developers in order to develop anything

Therefore considering all these factors we have implemented python to be as a programming language.

6.2 Implementation of YOLO Module

When an image with other animals are uploaded. Our breed classifier returns the result as the closest looking dog breed. This reduces the accuracy as well as false classification of other animals as dogs.

To avoid this, we implemented yolo model, the image is first passed onto a yolo model that detects a dog in the image(cannot detect the breed) and returns only the cropped image of that dog to the breed classifier. The breed classifier can work on detection

Working

Image uploaded by the user is first resized to a standard 416×416 to process the image in batches using parallel GPUs . This reduces the time required to train the network. This makes the model faster and easier to implement.

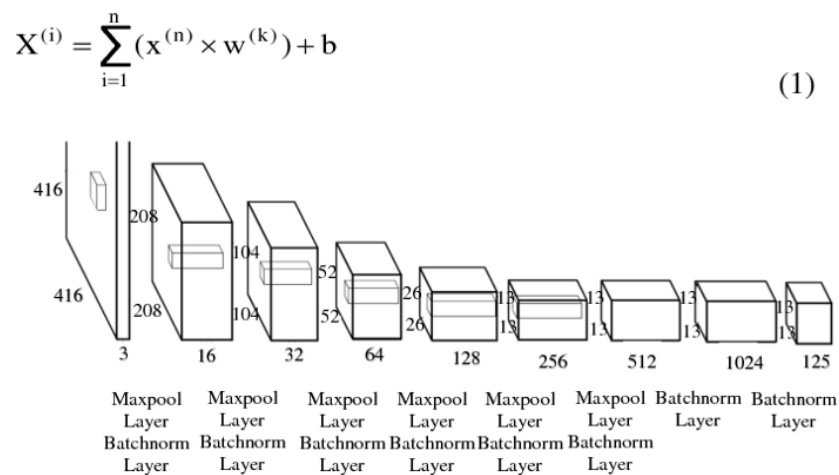


Fig 6.2.1: Working of YOLO

Multiple convolutions will be added to the image and in each layer a new feature will be added, such as color, shape, angle of inclination of the ear etc the model learns to detect a dog in this phase. The output of each layer is a output feature map, it is a 3D feature map of 32 stride. The final output is the combination of all the learnt features. The image passed on is divided into 13×13 grids depending on the stride mentioned. The current stride mentioned is 32 and if the

stride mentioned is 26 then the images divided into 16×16 . If the stride mentioned is 8 the grids are 52×52 the number of strides that the image has to be divided into depends on how small the required image is in the given image. we're expecting the user to upload the images of a dog, there are less chances of the dog being very small in the given image

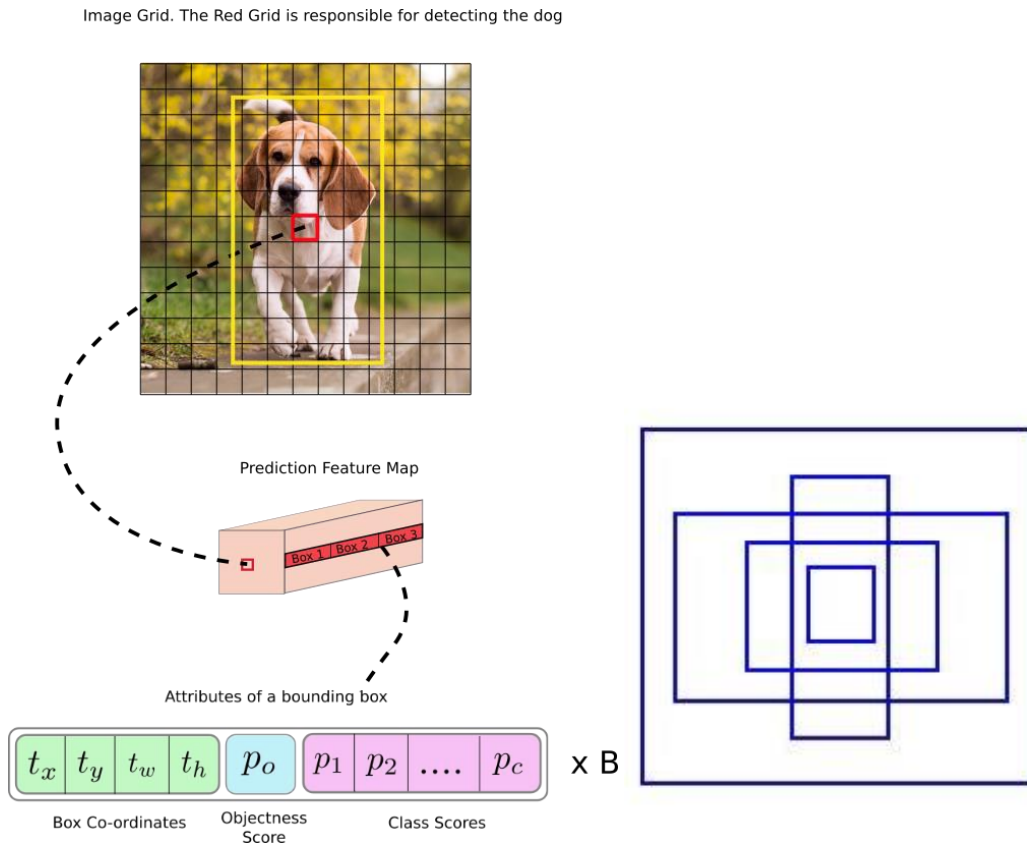


Fig 6.2.2: Bounding Boxes and Grids

Each grid contains 5 bounding boxes around it to detect the image, the entire image will fall into one of the bonding boxes depending on the shape. Therefore $13 \times 13 \times 5$ bounding boxes are present if the the stride is 32 and the bounding boxes increases with the increase in size. Therefore the chances of an image falling in the space of two bounding boxes is very high. To avoid leaving out any part of the image the two bounding boxes into which the image falls under are taken an undergo non max suppression. The area of overlap by total area of the bonding boxes gives the exact area of the image of the dog and this part is selected. The final cropped image is bounded by a box and send to the breed classifier to detect the breed. If a dog is not detected in the image the image is discarded and an error message occurs.

6.3 Implementation of Classifier Module

In order to implement the classifier module, we have taken a data set from Kaggle. Cackle is a very well-known site which contains a lot of data sets. Kaggle allows any user to publish the data that they have collected and it also allows to explore and build any type of data science projects.

In our classification module we are classified up to 120 breeds of dog. These 120 breeds of dogs are distinct to another and are present in the data set that we have downloaded. We have ensured that we are clearly classified these 120 breeds of dogs. Knowing the fact that there exists more than 120 breeds of dogs, we have decided to implement and scale are projects to more breeds of dogs in the near future.

Every image has to be pre-process it actually goes into the classifier module. Even in our project we are pre processing. The data set that we have downloaded does not have any false data or any bad data and therefore we have skipped the process of data cleaning since while downloading the data set itself the data cleaning was done to us through Kaggle. This is a huge advantage of using Kaggle. The only preprocessor we are applying to all the images in the data set is basically making it ready for training, testing and validation. For every picture that is present in the data state we are providing a label and an index to each of them. This is called as annotation where in a label is defined to any data set or values. This become very crucial when we are identifying and validating data.

The first very important step of classification module would be to create what is called as a bounding box. These bounding box is help us to identify the object that is dog in our case and crop them out and it save them to a folder. The cropping out of images and saving them to a folder is done according to ImageNet Standards. ImageNet if nothing but an image database in which each and every node that is present in the hierarchy is depicted by thousands and thousands of images. The cropping of the pictures according to image net standards involve a certain dimension and certain fixed normalization values. The part of the code snippet where we are implementing this image net standard is shown below.

```
transforms.Compose ([  
  
transforms.Resize(size=299) ,
```

```
transforms.CenterCrop(size=299),  
transforms.ToTensor(),  
transforms.Normalize([0.485, 0.456, 0.406], [0.229, 0.224, 0.225])  
1)
```

According to the image that standard the size of the object has to be 299X299 pixels along with the other fixed floating point numbers that are shown in the code above. After the cropping of the images take place according to the standards, these images are saved into two folders, training folder and testing folder. There are totally around 20000 that we are dealing with.

The next part of the classifier module is to use Convolutional Neural Network (CNN) to identify the features of a dog that are present in the image and then use the features identified to train the model and to identify the breed. Construction of CNN is quite complicated and will also require a lot of time. In order to simplify our process of construction of CNN we have implemented a pre-trained model in order to provide us with the feature of CNN thereby reducing the complexity of a project. Inception V3 is the pre trained model we have implemented in a machine learning classifier. Inception V3 is a popularly used pretrained model that can classify up to 1000 out of which dog is also one of those. Inception V3 provides the entire feature of CNN which is called as feature extraction. This is the first part of this pretrained model. The second part of the pre trained model would be classification. Since this model can classify up to 1,000 images we will not want to implement this classifier because of the number of images that we can classify. We will be freezing the classification layer of inception V3 in order to implement our own classifier. This pre trained model is known to print all the possibilities for an object but we need to only print breed of the dog not the possibilities but only one breed of the dog therefore we will be freezing the classification layer of this pre-trained model and implementing our own classifier. The feature extraction path which is also an important part of our project is entirely provided by Inception V3. The process in which the feature is extracted is shown in the figure below.

We will be looking at how a dog's picture after being past will undergo several levels of feature extraction in order for the machine to understand that is picture of the dog and then send it to the classifier.

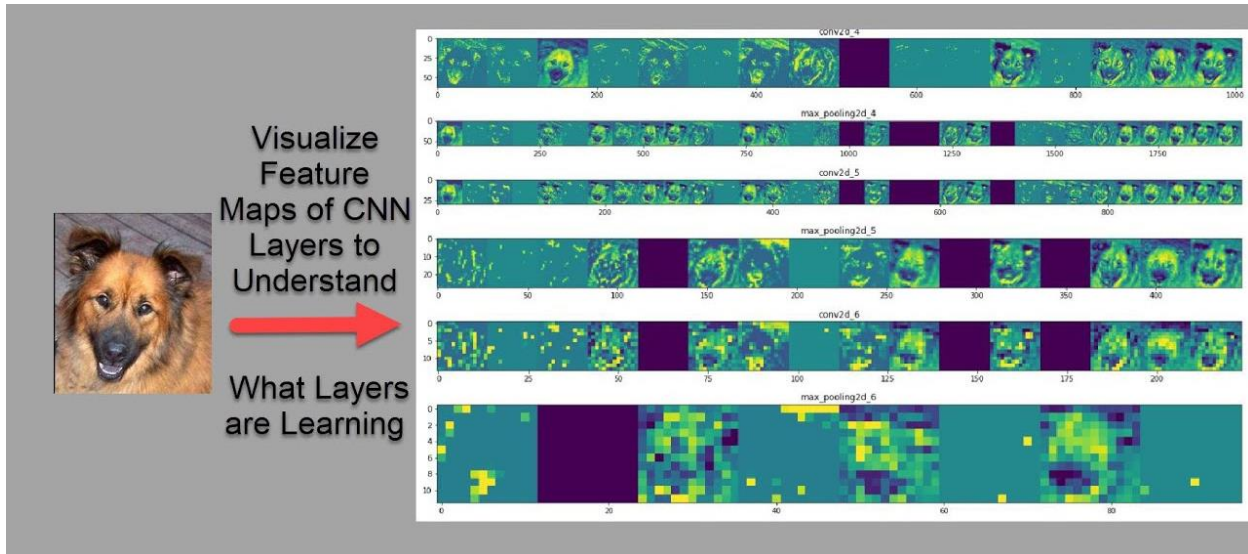


Fig 6.3.1: Feature Map

After the picture of the dog is uploaded what is called as a kernel which is a feature Extractor runs over the image of the dog not just once but multiple Times. This kernel on running multiple times will develop a feature map every time it runs over the image. In the feature map each and every feature of the dog will be identified as shown in the figure above. An entire view as to how every feature is being identified will be understood by the system itself in order to classify it as a dog. Every violet block that is present in the feature map is basically when the model is not learning anything at that stage. A method called as transfer learning is used to understand the features of CNN. Transfer learning means that every time the machine learn something it is transferred to the next day's thereby learning each and every distinct features of a dog. If we look closely into the feature map we can see that the eyes and ears and the face structure of the dark can be identified in the feature map. This goes on identifying the features until the system is clear about what features can be present in water breed of dog. As we move below the feature map it becomes a very complex that is it becomes unreadable by humans and can only be understood by the computer. This is how the entire process of the CNN actually works by implementing Inception V3.

Since Inception V3 is a pre trained model it can be easily deployed into a program just by using the following code snippet. This code snippet is written using Python and r model is set to Inception V3 and where term and the pretrained nature of it to be through which means that we want the CNN part of this Inception V3 to work. Thereby, implementing this pretrained model to

work with CNN which is the most complex feature of any machine learning classification problems.

```
[32] model = models.inception_v3(pretrained=True)
      model.aux_logits=False
      model
```

Fig 6.3.2: Deploying Inception

Already discussed that Inception V3 can classify up to 1000 we are going to freeze the classification layer of inception V3 by setting it to false, this part of the code snippet is shown below.

```
▶ for param in model.parameters():
    param.requires_grad = False
```

Fig 6.3.3: Freezing Classification Layer

After the classification layer of the pre trained model we will be implementing our own classifier with the sequential classifier. We are basically dealing with sequential classifier because we will be taking in one input from the user and also be providing one output to the user thereby we are implementing sequences classifier. The sequential classifier has a lot of variables that we are dealing with. Since we are identifying 120 breeds of dog the number of classes in our classifier will be 120. In order to prevent overfitting of data we have used Dropout. Since we will be dealing with probabilities we will be also implementing softmax layers for the same reason. The number 1024 is used because we are defining 1024 internal features for model. The code snippet that is used to implement the classifier is shown below.

```
▶ n_classes = 120
  n_inputs = model.fc.in_features

  model.fc = nn.Sequential(
    nn.Linear(n_inputs, 1024),
    nn.ReLU(),
    nn.Dropout(0.4),
    nn.Linear(1024, n_classes),
    nn.LogSoftmax(dim=1))

  model.fc
```

Fig 6.3.3: Implementing Classifier

After extracting all the features the next step would be breed detection that is done by training the data. In order to train the data we have considered epoch. We have defined up to 20 epoch.

The code snippet used for training is given below:

```
for data, label in val_loader:

    data, label = data.cuda(), label.cuda()

    output = model(data)

    loss = criterion(output, label)

    valid_loss += loss.item() * data.size(0)

    _, pred = torch.max(output, dim=1)

    correct_tensor = pred.eq(label.data.view_as(pred))

    accuracy = torch.mean(correct_tensor.type(torch.FloatTensor))

    valid_acc += accuracy.item() * data.size(0)

train_loss = train_loss / len(train_loader.dataset)

valid_loss = valid_loss / len(val_loader.dataset)

train_acc = train_acc / len(train_loader.dataset)

valid_acc = valid_acc / len(val_loader.dataset)

history.append([train_loss, valid_loss, train_acc, valid_acc])
```

First and initial step would be to give the labels and indexes. We have an Optimizer that calculates the error. Loss is determined by multiplying the previous loss with the batch size, in our case batch size is 128. The accuracy is determined by max log probability. The pred yields the index of the breed that has been detected. This predicted index is compared with the index of the breed, if the prediction is correct, 1 is returned to the user. Otherwise 0 meaning false is returned to the user. By estimating this and recording all the epoch values, we determined that our model was 79% accurate. The history of losses and accuracy is shown in the chart below. We considered the early stopping stage because our model did not observe much losses after a stage.

The accuracy and losses remained same and thus to prevent overfitting, we called in early stopping functions.

	train_loss	valid_loss	train_acc	valid_acc
0	2.850237	1.150930	0.431244	0.731293
1	1.182436	0.833909	0.689687	0.776968
2	0.968623	0.782745	0.727466	0.790573
3	0.882306	0.727162	0.744837	0.789602
4	0.818562	0.712987	0.762026	0.791545
5	0.778562	0.709375	0.769801	0.791059
6	0.741262	0.686897	0.780673	0.789602
7	0.718872	0.698174	0.785897	0.793489
8	0.691283	0.717993	0.790209	0.784257
9	0.664300	0.681601	0.797012	0.793489
10	0.636994	0.660161	0.805454	0.798348
11	0.620887	0.693265	0.807519	0.796890
12	0.607554	0.679501	0.812682	0.798834
13	0.588492	0.657882	0.813472	0.803693
14	0.566869	0.661825	0.821611	0.795432
15	0.561155	0.664027	0.824101	0.804665
16	0.549950	0.669909	0.826531	0.795918

Fig 6.3.4: History of Losses and Accuracy

6.4 Connectivity

We are using React js as our front end. React is a library or a framework which is fast and easy to develop the UI of the application. In our application we are using the mern stack.

Mern stands for mongo DB, express JS, react JS, node JS respectively. React JS was introduced by Facebook a while back and it is used to build single page application. We have three features in our application the first one being dog classification, the second being lost dog portal and the third one is the product categorization. To develop this we separated each of them into separate components. In react JS the advantages are that everything is basically considered as a component. There are functional components and class components.

We have used functional components in all the components that we have constructed in our application. Functional components are better than class components as we can use es6 features

and many other features that are used to our advantage in the application that we were building. Some of the components that I would like to mention would be the authentication component which is used to sign in login and use Google oauth and JSON web Token to do it so this is considered as one component. There are a lot more other components that were used in the application, that is being the classification form home navbar post details post and product components.

We have also used material UI and react bootstrap as well as CSS for styling our react frontend. This was quite helpful to make our UI look much better and more user friendly. Coming to the first feature that is the dog classification, we have created a simple user friendly page.

Here, we have an input to give the input of the image and by using some of the react features we are hitting the backend endpoint `api/infer` and we're using a flask server on the backend for the machine learning side. So once we input the dog file it goes to our backend and the server is how we are connecting the application and this flask server sends the photo to the model and the model detects the dog and then classifies the breed and the server sends it back to our front end. Here we capture it and show the detected breed to the user. This was quite simple.

We have also used redux for state management in our application. We are mainly using redux for our lost portal and product categorisation components.

```

12 app = Flask("__main__")
13 CORS(app)
14
15
16 @app.route("/api/infer", methods=['POST'])
17 def infer():
18
19     files_stream = request.files
20     if not 'image' in request.files:
21         return jsonify({
22             "success": False,
23             "result": "file not found..."
24         })
25
26     f = request.files['image']
27     f_name = f.filename
28
29     fpath = os.path.join(IMAGE_TEMP_FOLDER, f_name)
30     f.save(fpath)
31
32     yolo.predict(fpath)
33
34     yolo.clear()
35
36     with open(fpath, 'rb') as image:
37         imbase64 = base64.b64encode(image.read())
38
39     return imbase64
40
41

```

```

client > src > components > ClassificationHome > ClassificationHome.js
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68

```

```

request.open("POST", "http://127.0.0.1:9999/api/infer");
request.send(formData);
return (
  <div style={{
    display: "flex",
    flexDirection: "column",
    alignItems: "center",
    justifyContent: "center",
    margin: "30px",
  }} >
    <p style={{ fontSize: "1.5em", margin: "20px 0" }}>
      We recommend you to upload an image with a single dog in it for better accuracy.
    </p>
    <form id="uploadForm" onSubmit={uploadImage}>
      <input id="image-upload" type="file" name="image" />
      <button type="submit">Upload</button>
    </form>
    {isSuccess === true ? (
      <p style={{ color: "green", fontWeight: "bold", margin: "10px 0" }}>
        Success
      </p>
    ) : (
      <></>
    )}
    <div>{img}</div>
  </div>
);

```

Fig 6.4: Connectivity

6.5 Lost Portal

Now that we're talking about lost portal. We have multiple components, posts, forms and a single post. So we give you a form for you to put the name, message and image of your lost dog.

This consists of the form component. Once you hit on submit this is where our redux and redux think take place. Redux think is used to access or communicate with external resources. We can make http requests and other useful things with redux-thunk. So how the project works is to give some input and redux takes care next. Redux has a few things that we need to take note of firstly we have the actions, which are then sent to the reducer, the reducer updates whatever new value we have. So how do we get the new value, also we are using mongo DB as our database and we have value stored in it.

So this will be our new values whatever we fetch from our database. When we input some value we send an action, this is the redux action and then we use redux think to make http request to our server backend and fetch data from our database if we are facing some post and store it it was trailer and send this to our end user and the reducer will update our store. Store is where we take our values into each separate component.

We use the useSelector hook to get these values from our redux store. So the store contains some value and we take this value and put it in our component or we can say our react component and store it in our local state and this in turn can be displayed to our UI. We know how the MVC architecture works and react focus is just on the V part, V being the view of MVC architecture.

We can take an example of posting a lost dog. We give you a form to fill in the details as mentioned earlier, once you fill the details you hit on the submit button and then we use axios which is a package that we have installed to make API requests and responses much simpler.

We are using Express JS as our server for the lost dog portal and products categorization. Some of the important end points that will be hitting from the front-end to our server would be the post end point and all of this is mentioned in the routes of an Express server.

So once we have hit the endpoint we will be using Mongoose and we take the request that comes with the http and use mongoose to make a post to our database that is mongo DB. We successfully made a post of the Lost dog.

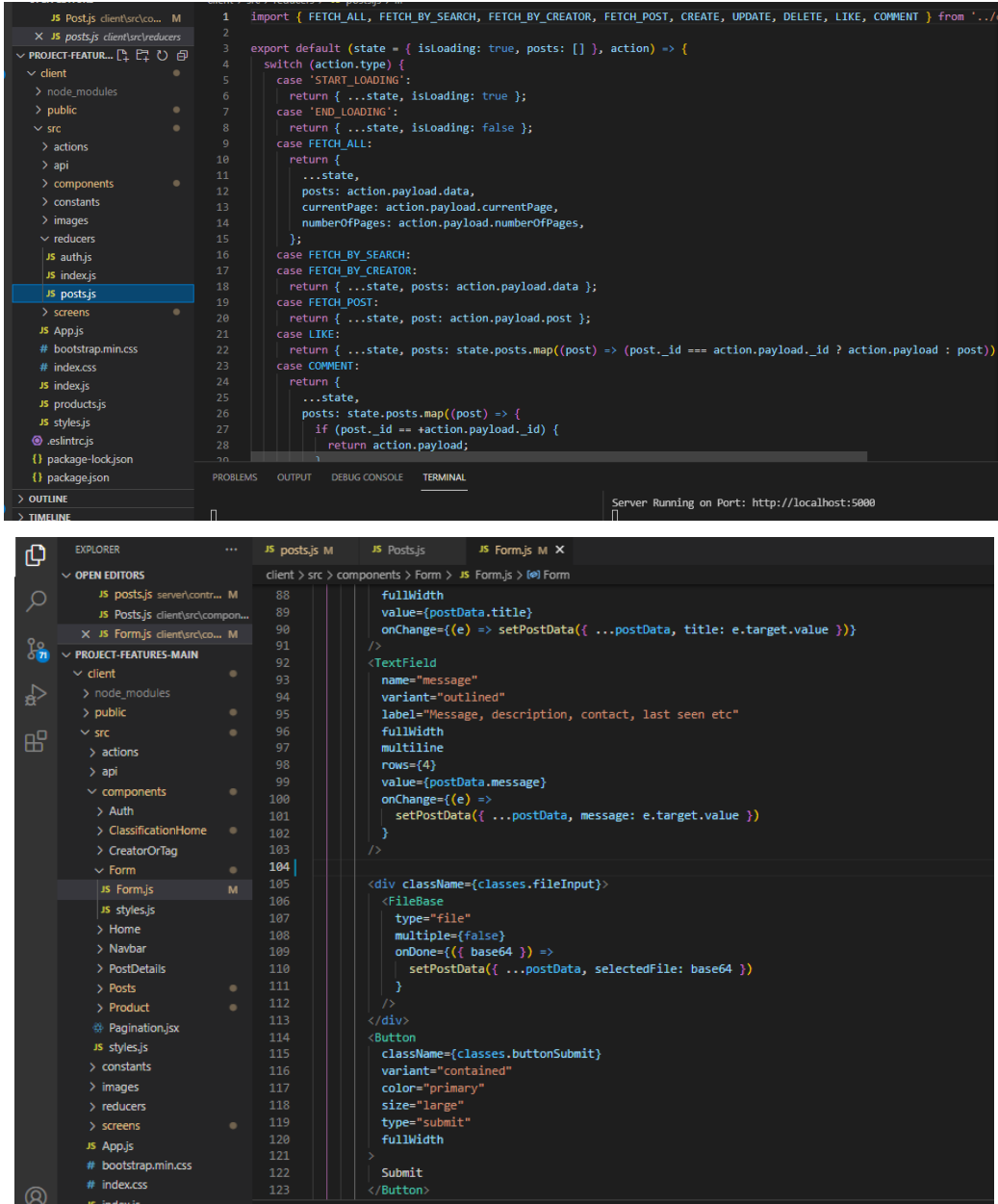


Fig 6.5: Lost Portal

6.6 Product Categorization

Similarly we have the product component. Share will have to fetch all the dogs, details of each of these dogs, products related to these dogs. We have already stored this in our database. We make a get request and fetch all the data from our database. This is done as soon as we enter the product page and we have used the useEffect hook to make the get request. One fits this data we store it in a local state of a component, i.e the products component and we display the

information to the user. Coming to the backend we have connected to our mongo DB cluster in our server index.js. We also have the route middleware and controllers folder in our server side. There we have also created models for the user, posts and products which can be used to do multiple functions with our database for example posting, fetching, editing, deleting etc. This is a brief of how we have connected our front end and back end of our application. To summarise we have used react JS, flask, mongo DB, express JS, redux and redux thunk to achieve the connectivity of a frontend and backend.

```

11  const app = express();
12
13  app.use(express.json({ limit: "30mb", extended: true }));
14  app.use(express.urlencoded({ limit: "30mb", extended: true }));
15  app.use(cors());
16
17  app.use("/posts", postRoutes);
18  app.use("/user", userRouter);
19
20  app.get("/api/products", (req, res) => {
21    res.json(products);
22  });
23  app.get("/api/products/:id", (req, res) => {
24    const product = products.find((p) => p._id === req.params.id);
25    res.json(product);
26  });
27
28  const CONNECTION_URL =
29    "mongodb+srv://varun:varun123@cluster0.bqgig.mongodb.net/myFirstDatabase?retryWrites=true&w=majority";
30  const PORT = process.env.PORT || 5000;
31
32  mongoose
33    .connect(CONNECTION_URL, { useNewUrlParser: true, useUnifiedTopology: true })
34    .then(() =>
35      app.listen(PORT, () =>
36        console.log(`Server Running on Port: http://localhost:\${PORT}`))
37    )
38    .catch((error) => console.log(`${error} did not connect`));
39
40  mongoose.set("useFindAndModify", false);
41

```

```

88  fullWidth
89  value={postData.title}
90  onChange={e} => setPostData({ ...postData, title: e.target.value })
91  />
92  <TextField
93    name="message"
94    variant="outlined"
95    label="Message, description, contact, last seen etc"
96    fullWidth
97    multiline
98    rows={4}
99    value={postData.message}
100   onChange={e} =>
101     setPostData({ ...postData, message: e.target.value })
102   />
103  />
104  <div className={classes.fileInput}>
105    <FileBase
106      type="file"
107      multiple={false}
108      onDone={({ base64 }) =>
109        setPostData({ ...postData, selectedFile: base64 })
110      }
111    />
112  </div>
113  <Button
114    className={classes.buttonSubmit}
115    variant="contained"
116    color="primary"
117    size="large"
118    type="submit"
119    fullWidth
120  />
121  Submit
122  </Button>
123

```

Fig 6.6: Product Categorization

Chapter 7

TESTING AND RESULTS

The whole purpose of testing any software that has been developed is to look for fault or error so that does not interrupt the user when he or she is using the program. Testing can also be defined as a way to discover any fault or weaknesses in a completely developed product. It checks for the entire program, development components, assemblies and verify that there are no error present in this. This testing process becomes very efficient in any software. This needs to be done so that the software does not fail when the user tries to use it or does not give any wrong output when the user wants to interpret or interact with the software. There are various types of test and each of these tests has to be incorporated into any software that has to be developed.

7.1 Unit Testing

Unit testing basically involves testing each and every unit and see if the logical up as properly and also use of these two not functioning properly and provides the required output to the user. Unit testing is the individual software unit testing in any kind of application that has been developed. Unit testing has to be done before the entire system is put into one which means basically before integrating it as a whole component. Unit testing becomes difficult after being integrated because to go back and check each and every component is a huge task and also very time-consuming. Unit test has to be performed at the earliest stage in order to avoid any complexity that can occur in the near future. Unit test ensure that each and every component is working fine so that they do not cause any kind of exceptions are error later.

7.2 Integration Testing

Integration testing would come after unit testing. Creation means that to bring together all the components of a system. We have ensure that each and every unit has been working just fine then only the system can be integrated into one component. Several test will also have be performed in integration test because even the every unit functions correctly when integrated as a whole it might not function the exact way it was expected to function. In order to get the single correct output integration testing becomes very essential in any software development stage.

7.3 Functional Testing


Functional testing is basically testing entire software if it is function in the right way. In the right way means that it is providing the desired output to the user and not the undesired one. After integrating and doing integration test there could be some use cases where are the output that is being given to a users will not be in the right for. We will have to perform something called as functional testing where every input output and test cases are function and this ensured minimal errors to the user.






7.4 White Box Testing and Black Box Testing

White box testing with type of testing where in the tester are the person who is sitting down to test the entire software will actually have no idea about what exactly is happening in the software components of the software or any other such information. Look for the functioning of the software and they do not worry about any other thing apart from this. White box testing users only look if they are getting the output that they want and do not really intended or write the kind of user cases or go in detail.

Black box testing is another type of software testing where in all the internal knowledge of the model and any other details regarding to this particular thing will actually be known to the tester and he or she is going to sit down and do the required documentation Research and other specification to see if the software is working just fine as predicted.

7.6 Testing the Breed Detector

GIVEN INPUT	EXPECTED OUTPUT	ACTUAL OUTPUT	REMARKS
	Afghan Hound	Afghan Hound	Pass

	Golden Retriever	Golden Retreiver	Pass
	Chihuahua	Chihuahua	Pass
	Labrador	Labrador Retriever	Fail
	Cardigan	Cardigan	Pass
	Terrier	Terrier	Pass

Chapter 8

SNAPSHOTS

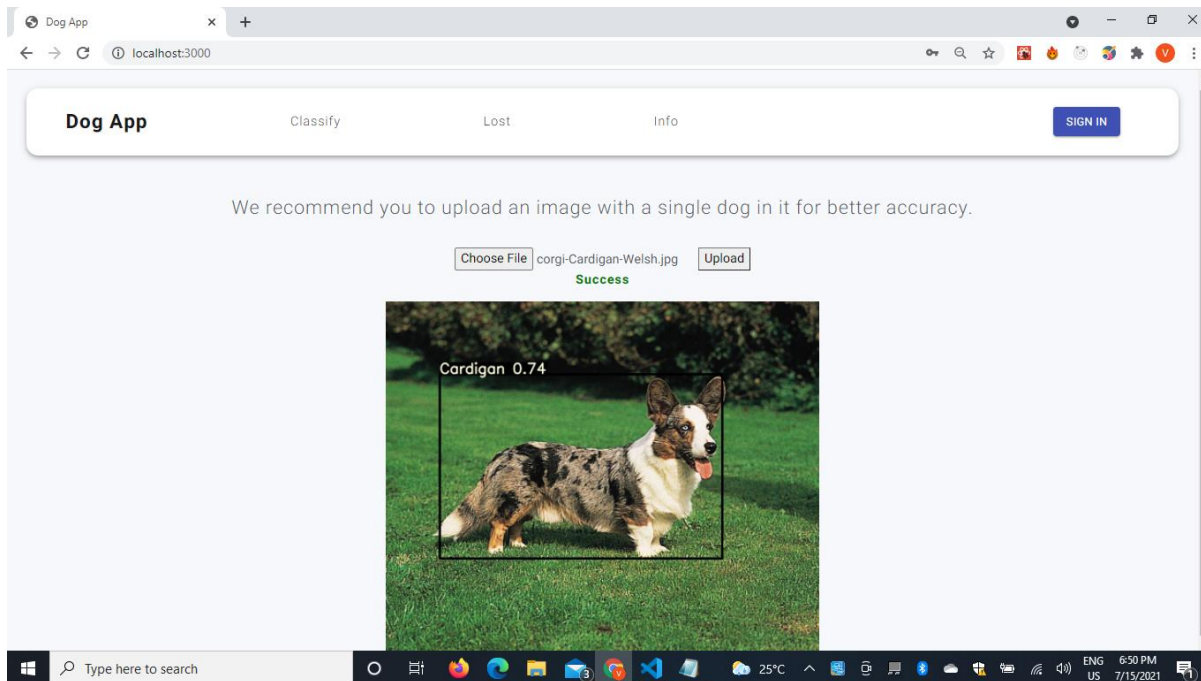


Fig 8.1: Dog Breed Identifier

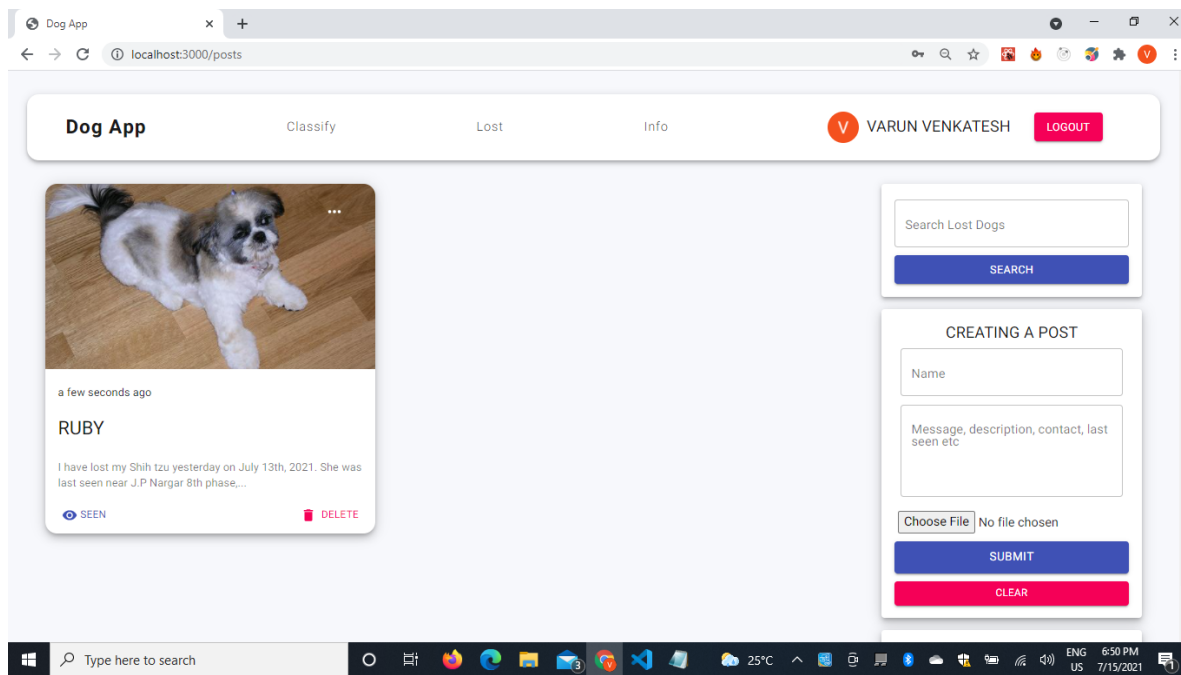


Fig 8.2: Lost Potral

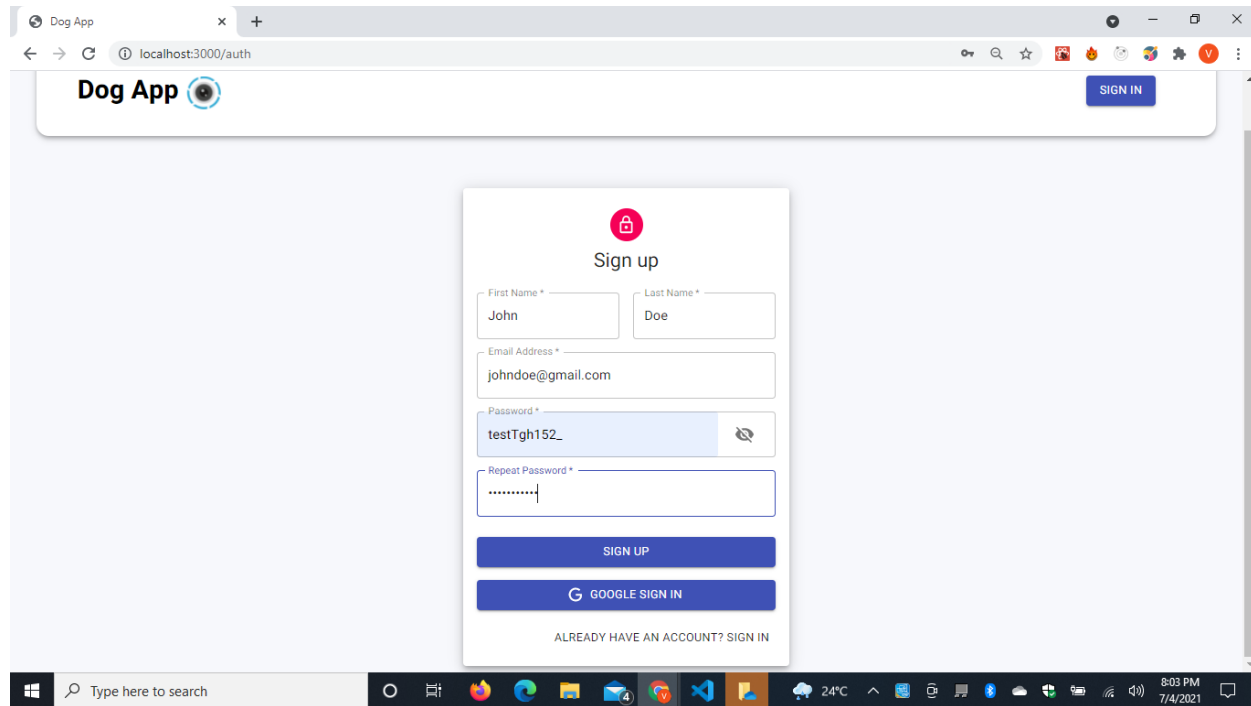


Fig 8.3: Login Part 1

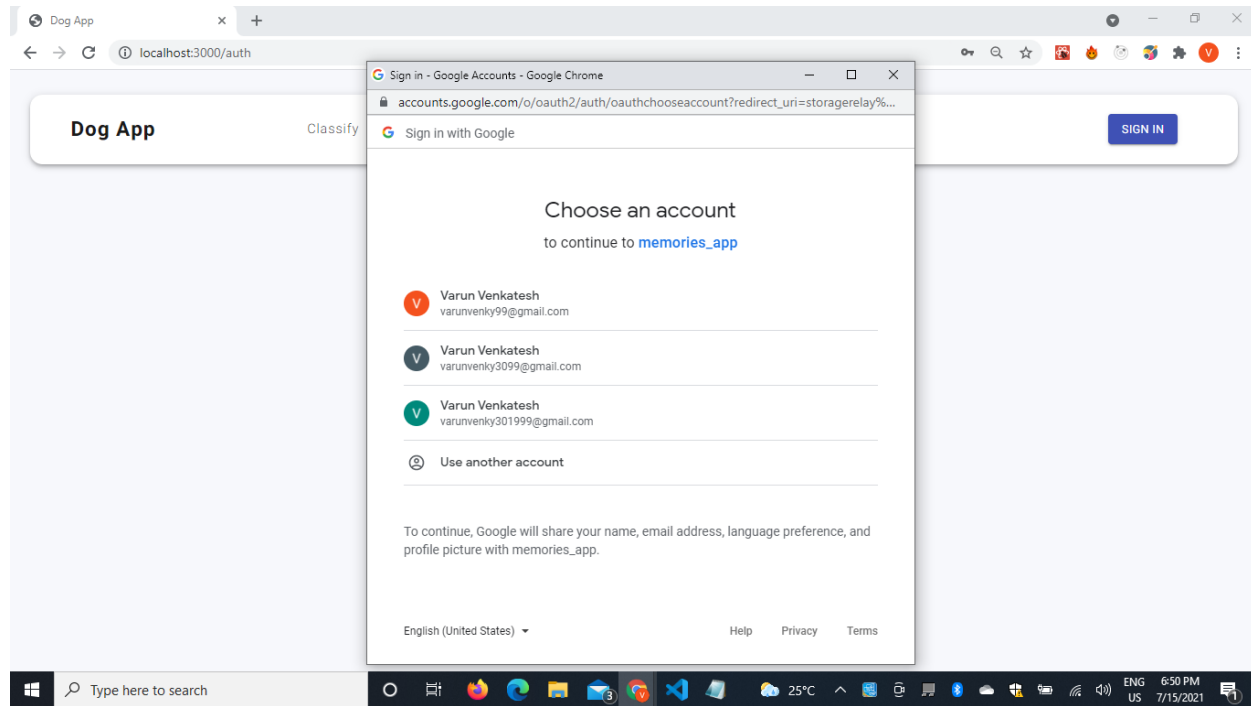


Fig 8.4: Login Part 2

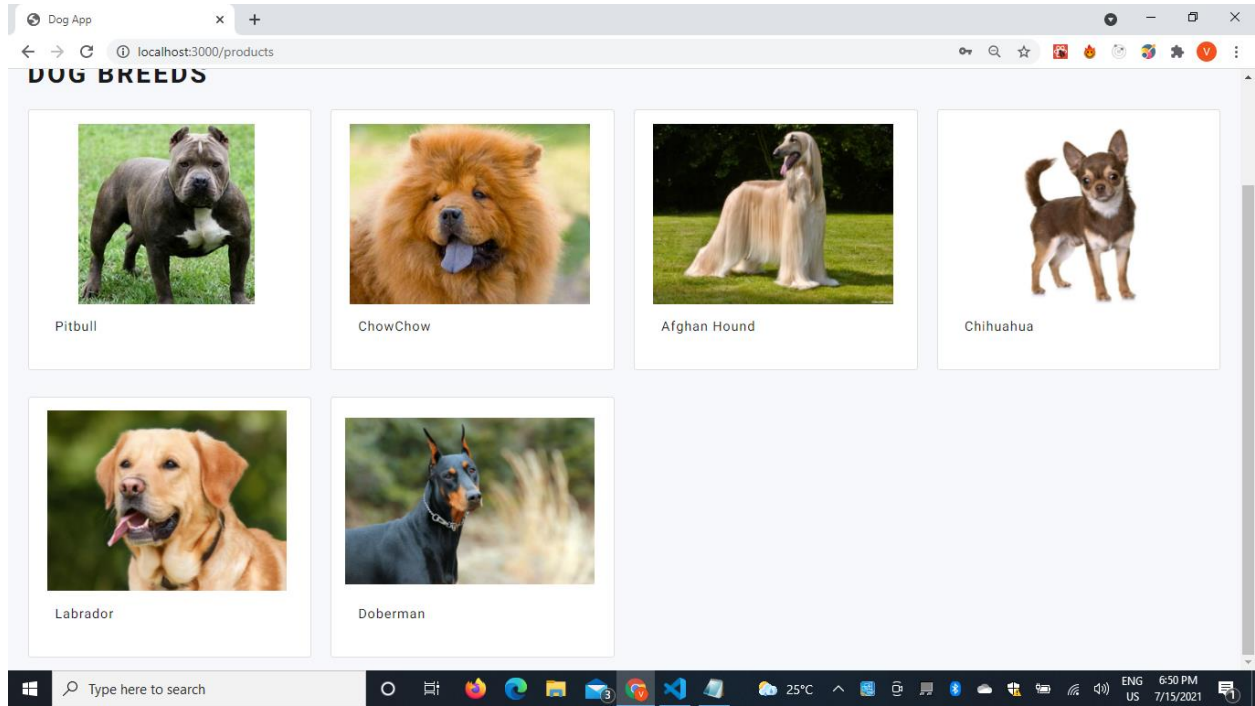


Fig 8.5: Product Categorization

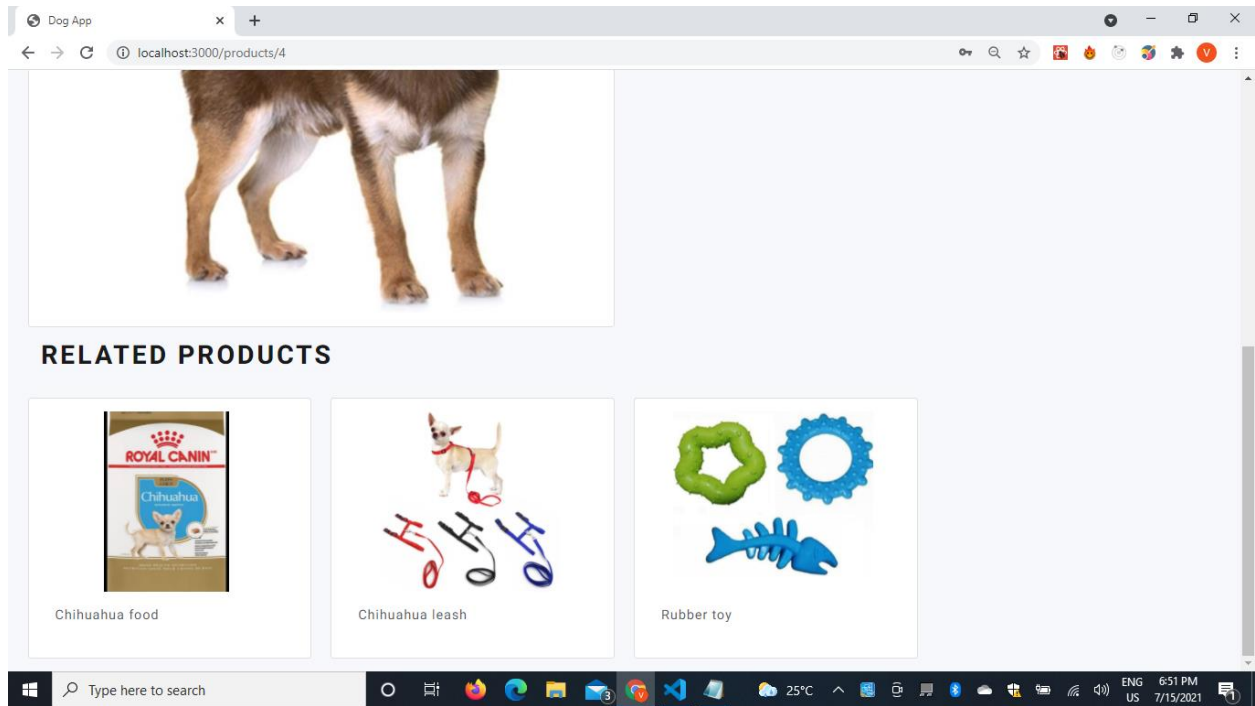


Fig 8.6: Breed Specific Products

CONCLUSION AND FUTURE ENHANCEMENTS

Throughout this project we aim to provide the user with the best experience in using our application. Application is developed only for the web we have made sure that we have incorporated all the features that can provide users the best experience. Where also reducing the navigation time of a window by providing all the features under one roof. Most of the application that are already present on the internet do not have all the features that are incorporated in a centralized way and we are providing in a centralized to a which could be a major advantage.

Detection of breed, product categorization gives users a sound knowledge about the pit and helps them to understand the breed of the dog and also guide them to choose the right product for their pets. This way they can provide what is best for their pets and also take care of them effectively and efficiently.

The Lost portal is an assisted service that is specifically provided to the user. Any time the user knows that their pet has been missing they can simply post their pm on to our website and all the people can view this thing and help the person find the missing. The last portal is not only specific to talk so that users can post any and pet that is lost.

The future enhancements of the project would include to detect more than 120 breeds of dogs. Model only detects the breed of the dog if the image contains only a single talk but we are planning to expand it to identify multiple dog breed in one single image itself. We are also planning to implement all the Lost dogs and filter them specific to your city and also include a location on the map. Intent on recommending more products for almost all the breed of dogs. To improve authentication so users can also have a safe experience when dealing with the application.

REFERENCES

- [1]. Geethapriya. S, N. Duraimurugan, S.P. Chokkalingam Real-Time Object Detection with Yolo. International Journal of Engineering and Advanced Technology (IJEAT) ISSN: 2249 – 8958, Volume-8, Issue-3S, February 2019.
- [2]. Joseph Redmon , Santosh Divvala, Ross Girshick , Ali Farhadi You Only Look Once: Unified, Real-Time Object Detection.
- [3]. Vincent Vanhoucke, Sergey Ioffe, Jonathon Shlens, Zbigniew Wojna University College London, “Rethinking the Inception Architecture for Computer Vision Christian Szegedy Google Inc.”, arXiv:1512.00567v3 [cs.CV] 11 Dec 2015
- [4]. <https://www.petsworld.in/> - the idea of food, accessories are referred from this website.
- [5]. Whitney LaRow, Brian Mittl, Vijay Singh, Dog Breed Identification
- [6]. <https://towardsdatascience.com/classify-any-object-using-pre-trained-cnn-model-77437d61e05f>

Dog breed identification with product classification and assisted services

ORIGINALITY REPORT

7%

SIMILARITY INDEX

6%

INTERNET SOURCES

3%

PUBLICATIONS

6%

STUDENT PAPERS

PRIMARY SOURCES

1	www.cdc.gov Internet Source	1%
2	www.arxiv-vanity.com Internet Source	1%
3	mafiadoc.com Internet Source	1%
4	mlwhiz.com Internet Source	<1%
5	segno-cakal.fun Internet Source	<1%
6	Submitted to Higher Education Commission Pakistan Student Paper	<1%
7	buyretinaonlinenoprescription.tk Internet Source	<1%
8	Submitted to National Institute of Technology, Kurukshetra Student Paper	<1%

9	Submitted to Corinthian Colleges Student Paper	<1 %
10	Submitted to Cornell University Student Paper	<1 %
11	www.jianshu.com Internet Source	<1 %
12	Submitted to Florida Institute of Technology Student Paper	<1 %
13	lup.lub.lu.se Internet Source	<1 %
14	Submitted to University of Southern California Student Paper	<1 %
15	Submitted to Middlesex University Student Paper	<1 %
16	Submitted to University of Central England in Birmingham Student Paper	<1 %
17	Jishnu U K, Indu V, K J Ananthakrishnan, Korada Amith, P Sidharth Reddy, Pramod S. "Voice Controlled Personal Assistant Robot for Elderly People", 2020 5th International Conference on Communication and Electronics Systems (ICCES), 2020 Publication	<1 %
18	machinethink.net Internet Source	

<1 %

19

artoftesting.com

Internet Source

<1 %

20

Submitted to University of Lancaster

Student Paper

<1 %

21

docplayer.net

Internet Source

<1 %

22

grras.com

Internet Source

<1 %

23

www.cv-foundation.org

Internet Source

<1 %

24

Therese Yamuna Mahesh, S S Parvathy, Shubin Thomas, Shilpa Rachel Thomas, Thomas Sebastian. "CICERONE- A Real Time Object Detection for Visually Impaired People", IOP Conference Series: Materials Science and Engineering, 2021

Publication

<1 %

25

github.com

Internet Source

<1 %

26

paperswithcode.com

Internet Source

<1 %

27

Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, Zbigniew Wojna. "Rethinking

<1 %

the Inception Architecture for Computer
Vision", 2016 IEEE Conference on Computer
Vision and Pattern Recognition (CVPR), 2016
Publication

Exclude quotes On

Exclude matches < 5 words

Exclude bibliography On