



**FACULTY OF ENGINEERING  
DEPARTMENT OF COMPUTER APPLICATIONS  
PROGRAM – MASTER OF COMPUTER APPLICATIONS**

**PES UNIVERSITY**

(Established under Karnataka Act No. 16 of 2013)

100-ft Ring Road, Bengaluru – 560 085, Karnataka, India

***6<sup>th</sup> Semester Project Report on***

**Pet Lookout**

*Submitted by*

**TRIPATHI SATISH DINESH (PES1201801831)**

**Jan – May, 2020**

**under the guidance of**

**Internal Guide**

Dr. A. Lekha

Associate Professor

Department of Computer Applications,  
PESU, Bengaluru – 560085



**FACULTY OF ENGINEERING  
DEPARTMENT OF COMPUTER APPLICATIONS  
PROGRAM – MASTER OF COMPUTER APPLICATIONS**

**CERTIFICATE**

*This is to certify that the project entitled*

**Pet Lookout**

*is a bonafide work carried out by*

**TRIPATHI SATISH DINESH– PES1201801831**

in partial fulfilment for the completion of 6<sup>th</sup> semester project work in the Program of Study MCA with specialization in Data Science under rules and regulations of PES University, Bengaluru during the period Jan. 2020 – May 2020. The project report has been approved as it satisfies the 6<sup>th</sup> semester academic requirements in respect of project work.

*Signature with date*

**Internal Guide**

Dr. A. Lekha  
Associate Professor,  
Department of Computer Applications,  
PES University, Bengaluru - 560085

*Signature with date & Seal*

**Chairperson**

Dr. Veena S

*Signature with date & Seal*

**Dean-Faculty of Engineering & Technology**

Dr. B K Keshavan

**Name and Signature of Examiners:**

*Examiner 1:*

*Examiner 2:*

*Examiner 3:*

## DECLARATION

I, **Tripathi Satish Dinesh**, hereby declare that the project entitled, ***Pet Lookout***, is an original work done by me under the guidance of **Dr. A Lekha** , Associate Professor, MCA Department and is being submitted in partial fulfilment of the requirements for completion of 6<sup>th</sup> Semester course work in the Program of Study **MCA**. All corrections/suggestions indicated for internal assessment have been incorporated in the report. The plagiarism check has been done for the report and is below the given threshold.

**PLACE:**

**DATE:**

*NAME AND SIGNATURE OF THE CANDIDATE*

## **ACKNOWLEDGEMENT**

I take this opportunity to express my greatest gratitude to the people who have helped me and supported me, whose guidance and encouragement has contributed immensely to the evolution of my project.

I express my deep sense of gratitude to the **Vice Chancellor of PES University, Dr. Suryaprasad J**, for providing me this valuable opportunity and **Dr. Veena S, Chairperson**, Department of Computer Applications for being the source of continual inspiration and overwhelming support.

I acknowledge with sincere thanks to the internal guide **Dr. A LEKHA, Associate Professor**, Department of Computer Applications, who is constant source of inspiration and provided with meticulous guidance, encouragement and constant personal support during the course of this project.

I also thank all the members of teaching staff for their help and suggestions. My acknowledgement will be incomplete if I do not thank my friends for being helpful and supportive.

Finally, I thank my, PARENTS for their blessings showered on me, at all times and in all circumstances.

**TRIPATHI SATISH DINESH**

**PES1201801831**

## **ABSTRACT**

Pet Lookout is a web based application that has following abilities:

For catering to clients who have pets such as dogs and cats.

To help pet lovers to register themselves as pet sitters, walkers and trainers and get hired for the same by creating their profile.

The application also provides car service booking for dogs helping owners to travel hassle free with them.

For other pet related services such as access to pet care tips through the posts made by the pet lovers and order varieties of cakes for both dogs and cats.

To apply for BBMP pet license without going out and reaching out to agents.

# CONTENTS

	Page No.
Abstract	
<b>1. Introduction</b>	<b>01</b>
1.1 Project Description	01
<b>2. Literature Survey</b>	<b>03</b>
1.2 Background Study	03
1.3 Feasibility Study	04
1.4 Tools and technologies	06
<b>3. Hardware and Software Requirements</b>	<b>08</b>
3.1 Hardware Requirements	08
3.2 Software Requirements	08
<b>4. Software Requirements Specifications</b>	<b>09</b>
4.1 Users	09
4.2 Functional Requirements	10
4.3 Non-Functional Requirements	11
<b>5. System Design</b>	<b>12</b>
5.1 Data Flow Diagram Level 0	12
5.2 Data Flow Diagram Level 1	13
<b>6. Detailed Design</b>	<b>15</b>
6.1 Use Case Diagram	15
6.2 Admin Activity Diagram	16
6.3 User Activity Diagram	17
6.4 Document Structure	19
<b>7. Implementation</b>	<b>21</b>
7.1 Source Code	21

# CONTENTS

7.2	Screenshots	39
<b>8.</b>	<b>Software Testing</b>	<b>59</b>
8.1	Test Cases	59
8.2	Selenium Test Cases	62
<b>9.</b>	<b>Conclusion</b>	<b>65</b>
<b>10.</b>	<b>Future Enhancements</b>	<b>66</b>
<b>Appendix A</b> Bibliography		<b>67</b>
<b>Appendix B</b> User Manual		<b>68</b>





# Introduction

# 1. INTRODUCTION

## 1.1 Project Description

“Pet sitting” is defined as “the act of caring for a pet in its own home while the owner is not present there. Walking is also a form of pet sitting since it involves coming to the pet’s home to provide exercise, care and companionship. With the increase in number of pet owners it becomes very important to have a dedicated website which thrives upon providing all the necessary requirements in need.

### 1.1.1 Problem Definition

Leaving pets alone at home at times can lead to a worrisome situation in which both the pet owner or the pet parent and the pet can feel very stressed. So, in these times it is always better to have someone look after the pet in the absence of the owner. With the increase in number of pet owners it becomes very important to have a dedicated website which thrives upon providing all the necessary requirements in need.

### 1.1.2 Proposed Solution

This is where Pet Lookout can turn out to very helpful to the pet parent and also for pet lover who can follow their passion for pets and can easily make some money out of it. To avoid such uneven circumstances of leaving pets alone and lack of pet related services one can lookout for a website which provides all the facilities and people who can be trusted and will provide utmost care to the pets so that their schedule and activities are not disturbed. The website provides the comfort of creation of your own profile through which any pet loving person can become a pet loving professional. The website is only dedicated for pets such as dogs and cats where pet parents can search for trusted people to look after their pets and also get access to the services provided on the website such as pet cab booking and ordering of cakes

### 1.1.3 Purpose

The main purpose of the system is that it provides pet owners to view a detailed view of the profile and posts, posted by the pet lovers and pet lovers too can do the same. The website also provides services such as pet car booking, where user, who can be a pet lover or pet parent can book a car service to travel alongside with their pet. Users will also be able to apply for BBMP license by entering all the details and uploading a record of their pet’s vaccinations and their address proof.

**1.1.4 Scope**

The application requires a person (admin) who can manage the users and other related aspects such as the license status , pet cab booking status and update the products and the order status of the users. The scope of the application is to use it for pet related services purpose only which can be used by anyone who loves pets or has a pet . The other aspect of the system is that it provides secure access to the system. Hence, the data remains secure and there is no data loss too.

# **Literature Survey**

## 2. LITERATURE SURVEY

The literature survey reveals the actuality of whether the development of an application is really needed.

### 2.1 BACKGROUND STUDY

Pet ownership is rapidly increasing, industry estimates have clearly shown that there are around 19 million pets in India (around 80 per cent among them are dogs, then followed by cats and then smaller animals like birds and fish), and on an average almost 6,00,000 pets are adopted every year.

Numerous types of pet services are available which offer pet sitting too as a service, but the people registered there are their own people and it primarily consists of the team of people which primarily belongs to the service provider.

There are certain websites such as [greatau.com](http://greatau.com) and [quickr.com](http://quickr.com) where people have been posting for pet sitting but it is not a dedicated site for pets only. Some of the currently running websites which have been providing pet sitting and many services have been discussed in the existing system.

#### Existing System

Some of the related websites that have been researched to make the newer development much easier and useful. Some applications are found much related to the current development. But with some extra functionalities to outperform the existing system. They are enlisted below.

**Waggle:** Waggle is an online community of genuine dog lovers who will board the pet when the owner travels or not around with the pet for long duration of time. Enjoys vacation without the stress of leaving their pet in a kennel or alone. With Waggle, their pet too, will look forward to a vacation. Waggle only provides house sitting as a service where one can leave their pet at the pet lover's residency. It does not provide the luxury of someone coming to the pet's house and looking after them.

**Petboro:** Petboro provides dog boarding, off-leash dog park, dog daycare, pet pickup and drop, dog training, and other pet related services. The user has to call and make an appointment for booking a car service, trainers and everything present on the website is contact based, where one has to call and book for the services being offered and without having any prior knowledge about

the service provider. The people working there cannot work individually and have to follow certain rules set up by the owner.

**Anvisinc:** Anvisinc is a Bengaluru based startup which provides pet sitters at the pet parents house by making an inquiry through their website. The pet sitters provided by them are their own trained people and they don't have their profile on the website for the pet parents to have a detail idea about them before hiring them. Also the login and registration functionality of the website is still under construction making it an incomplete work.

**Bark n' Treat Canine Bakery:** Bark n' Treat Canine Bakery is a startup passionate about providing healthy exclusive dog treats. Here users can order for cakes but only by sharing the details of their orders by WhatsApp or by calling them personally. No booking of cakes option is provided on the website.

## 2.2 Feasibility Study

Feasibility study is the analysing if the proposed project is technically, operationally and economically feasible.

### 2.2.1 Operational Feasibility

The person with bare minimum knowledge of using a website can comfortably use the application, the application will work with internet. The application is developed much user friendly way. It is operationally feasible because of its simple functionalities.

### 2..2. Technical Feasibility

Technical resources include tools which are used in this application are universally accepted. This application can run online in any web browser which is up to date. The technology that is used for implementation is freely available. However, the user does not require any software. the user of the application requires just a browser. The user is abstracted from the development view. Hence the application is technically feasible.

### 2.2.3. Economic Feasibility

The application may involve development cost and deployment cost. The application is developed with opens source tools, hence there is no cost involved in purchasing/licensing the

software. Hence no hardware cost is incurred for deployment. The people who use the application require basic computer operating skill. Hence there is no cost involved for training the end users.

## **2.3 ABOUT TOOLS AND TECHNOLOGIES**

### **2.3.1 Python**

Python is an interpreted, high-level, general purpose programming language. Python is very much powerful in supporting many ultra features that is very easily executed with different frameworks like Django. Python along with Django provide the comfort of web development with just few lines of codes.

### **2.3.2 Django**

“Django” is a python based high-level web framework which is highly used for developing web based applications for web development and clean, pragmatic design. With Django model forms, license application can be easily executed.and it's easy to learn. .It is open source and also simplifies the Django ORM which provides an elegant and powerful way to interact with the database.

### **2.3.3 MongoDB**

MongoDB is a cross-platform document-oriented database program. Classified as a NoSQL database program, MongoDB uses JSON-like (BSON) documents with schema.

Some of the main features of MongoDB are:

- Ad-hoc queries
- Indexing
- Replication
- Load balancing
- File Storage
- Aggregation
- Server-side JavaScript execution
- Capped collections
- Transactions

### **2.3.4 Bootstrap 4**

Bootstrap is the open source framework which is used for web page and web application front end development. It consist of HTML and CSS designs templates. This template can be accessed by using



class selectors. Using bootstrap it's is easier to design and format a web page or web application. It makes web development faster and easier. It uses HTML, CSS and optional JS extensions.

### **2.3.5 PyCharm Community Edition 2019.3.1**

For writing simple and efficient Python codes, an Integrated Development Environment (IDE) is need. PyCharm as an IDE for developing Python-based applications can be used. PyCharm easily runs on Windows, Linux, or Mac OS. Additionally, it contains modules and packages that help programmers develop software using Python in less time and with minimal effort.

### **2.3.6 MongoDB 4.2.2 Compass Community**

MongoDB Compass is the GUI for MongoDB. Compass allows to analyze and understand the contents of the data. It also allows to connect to the MongoDB Database to run queries, check queries execution plans, manage indexes, and create, drop/create collections and databases.

### **2.3.7 HTML**

It is a mark-up language used in the design of webpages. It uses tags such as <html>, <head>, <body>, <img>, <input> etc... These tags are used by the browsers to interpret the content of the page. Programs written in scripting languages such as JavaScript can be embedded in HTML. The different versions of HTML are HTML 2, HTML 3, HTML 4 and HTML 5.

**2.3.8 CSS – Cascading Style Sheets** is a style sheet language. It is used to define the look and layout of the content of the webpage. CSS has selectors which helps us to create custom styles for elements. We can use inline CSS or external CSS. Inline CSS is the styles applied in the HTML file itself using <style> tag. External CSS is that where we write the CSS in a separate file with extension .css. This file will be referenced in the HTML file inside

<head> as follows. <link href="style.css" rel="stylesheet" type="text/css">

# **Hardware and Software Requirements**

### 3. HARDWARE AND SOFTWARE REQUIREMENTS

#### 3.1 HARDWARE REQUIREMENTS

Table 3.1 Hardware Requirements

Hardware	Specification
Processor	2.6 GHz 6 -Core i7
RAM	8GB
Hard Disk	500GB
Keyboard and Mouse	Standard PS/2 Keyboard and Mouse

#### 3.2 SOFTWARE REQUIREMENTS

Table 3.2 Software Requirements

Software	Specification
Frontend	HTML5, CSS3, Bootstrap 4
Language	Python 3.8
Database	MongoDB 4.2.2
Framework	Django 2.2.9
IDE	PyCharm Community Edition 2019.3.1
Database GUI	MongoDB 4.2.2 Compass Community

# **Software Requirements Specification**

## **4. SOFTWARE REQUIREMENTS SPECIFICATION**

### **4.1 USERS**

This application has specific roles for various users. The users are categorized into

#### **4.1.1 Admin**

The admin in the web app is a superuser who could delete user profiles, update it. Admin can also assign other users as staff users if required. Admin can see all the bookings made by the users and also he will be receiving all the details filled in by users for License application, he can give the status about the booking and application. Admin also has the ability to add products such as cakes for dogs and cat ,he will go through all the orders made by the users and the users will be able to track the order after ordering and details will be provided by the admin.

#### **4.1.2 User**

The user here mainly consist of someone who has a pet or someone who loves pets , so once the user logs in, he can turn himself into a pet lover just by updating his profile which is automatically created after the user registers, if he/she doesn't wishes to be a pet lover , they can just leave updating their profile and continue to use the website for other purposes. Like ordering cakes, applying for BBMP pet license, pet car booking and view the posts posted by other users who are pet lovers and view their profiles so that if in near future they want someone to look after their pets in their absence or train their pet , they can contact them from the information available on the website.

## 4.2 FUNCTIONAL REQUIREMENTS

Pet Lookout application is a complete package of available services to ease out the stress of pet owners. System requirements specification is an organized collection of data incorporating of all the system requirements. It specifies all of the major functional and non-functional accepts of the system.

**Register and Login:** The user has to go through the registration process by providing valid username and password. Admin being superuser does not have to go through registration.

**Update Profile as Pet Lover:** User can create their profile and describe about themselves how they will look after the pets also about their experiences with pets.

**Pet Lover Posts:** Pet Lover will have to post about themselves with a picture of them with a pet

**.Update/Delete Posts:** Pet Lovers who have posted can update and delete the posts posted by them.

**View profiles through post:** User(Pet Parent) can view the profiles of Pet Lover through their posts.

**Apply For License:** User can apply for license by entering all the mandatory details and uploading a file of their address proof and vaccination record of pet.

**View License Details:** User once when applied for the BBMP pet license can view the details provided by the user and the status of application has to be updated by the user.

**Pet Cab Booking:** User can also book Pet Cab to travel from one place to another by mentioning the time and date.

**View Bookings and Cancel :** User can view their bookings and cancel it and Admin has to update the status of the booking.

**Order Cakes for Pets:** User can order cakes and other items available on the websites by going through categories relating to dogs and cats.

**Add Products to Cart :** User will be provided with a cart where he/she can add the items and remove the item .

**Place Order :** User once done with the addition of items into the cart can place the order by going to the checkout page and entering all the details like address , name , phone , zip and email.

**Track Order:** User can track their order after placing their order and can easily track the order by entering order id given after successful placement of the order and also by entering mail id provided during the checkout.

**User Management:** Admin can manage users and delete their profile if found not upto the mark of a pet guardian profile.

**View bookings/applications/orders:** Admin can also view the bookings made by the users for services like pet cab, license, food items.

**Update/Add Products:** Admin adds , delete and updates products details as required.

**Update Status:** Admin has to update the status of the order of the products made by the users, also update the status about the license application and pet cab service.

### 4.3. NON- FUNCTIONAL REQUIREMENTS

- **Performance:** This application provides quick result as its performance can be defined by the time required to complete the task. The application supports multiple users at a time. For example, it shouldn't take no longer than three seconds to load the original screen when the application is started. It should also be ensured that the app does not disrupts the user's input.
- **Scalability:** This is a simple web application based upon internet, app server and a database server. App server is of shared-nothing architecture. The database server alone needs memory and disk storage. so whenever, there is a requirement for larger data storage, the database server can be horizontally scaled to support enormous user requests.
- **Portability:** The complete application has to be uploaded in central server and can be used anywhere within the network across different devices.
- **Reliability:** The application should be reliable to perform the business, i.e. whenever the user performs some important functions it should be acknowledged with some confirmation.
- **Security:** The information is guaranteed because nobody can access the request without logging.

# **SYSTEM DESIGN**



## 5. SYSTEM DESIGN

### 5.1 DATA FLOW DIAGRAM LEVEL 0

The flow of information for any kind of system can be easily mapped in a data flow diagram. Clarified symbols can be used to demonstrate information entrances, exports, storage points and paths between each location, including rectangles, circles, arrows, and brief text Labels. Data flowcharts can range from simple, even manually drawn, process surveys to detailed, multi-level DFDs, which gradually deepen the way the information is treated.

#### LEVEL 0

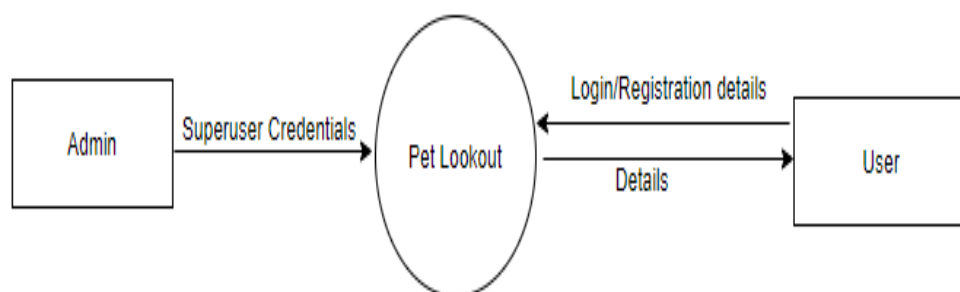


Figure 5.1 Data Flow Diagram Level 0

In the Fig 5.1.1, Admin and User are the users of the Pet Look out application All the users login with the username and password and Admin logs in through the superuser credentials and greatly interacts with the system.

## 5.2 DATA FLOW DIAGRAM LEVEL 1

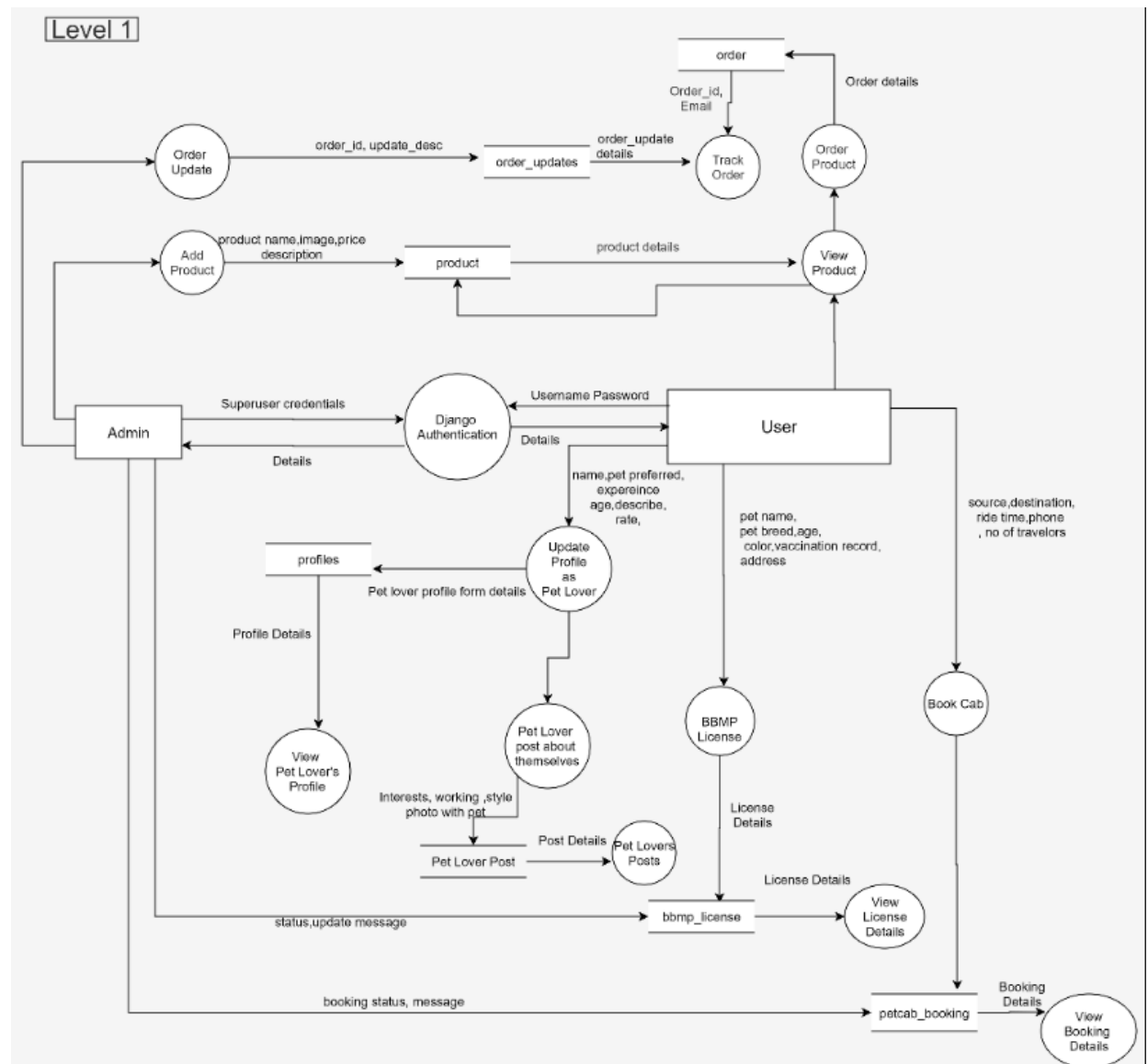


Figure 5.2 Data Flow Diagram Level 1

The above figure 5.2 describes the level 1 data flow of Pet Lookout. It mainly includes the processes which are update profile as pet lover, pet lover post about themselves, view pet lover's profile, apply for bbmp license, book cab, view booking details, add products, view products, order product and track order.

Admin adds product by entering all the details which includes product name, price, description and image of the product and these details are inserted into the product

collection from and when the user request to view he products it fetches all the details about the products.

Admin updates the status of the order by updating the message which gets inserted into the order updates collection.

User tracks the order by entering the order id and email entered while placing the order.

User will update profile by entering all the profile details such as name , pet preferred, age, experience, rate and describe yourself and all the information is inserted into profiles collection and from there when user request to view the profiles of other pet lovers all the details is presented.

Pet Lovers also posts about themselves by entering working style, interests and uploading a picture with a pet and all these details gets inserted into the pet lovers post collection.

For BBMP license User will enter the pet name, age, color, breed and upload vaccination record and address proof which will get inserted into the bbmp\_license collection and when the user request to view the details it fetches all the information and also the status of the application update by the Admin.

User books a pet cab by entering all the details such as source, destination ,ride time, ride date, phone and number of travelers in the booking form which gets inserted into the petcab\_booking collection and when the user request to view the status it fetches all the details about the booking status along with the message and status update by the Admin.

.

# **DETAILED DESIGN**

## 6. DETAILED DESIGN

### 6.1 USE CASE DIAGRAM

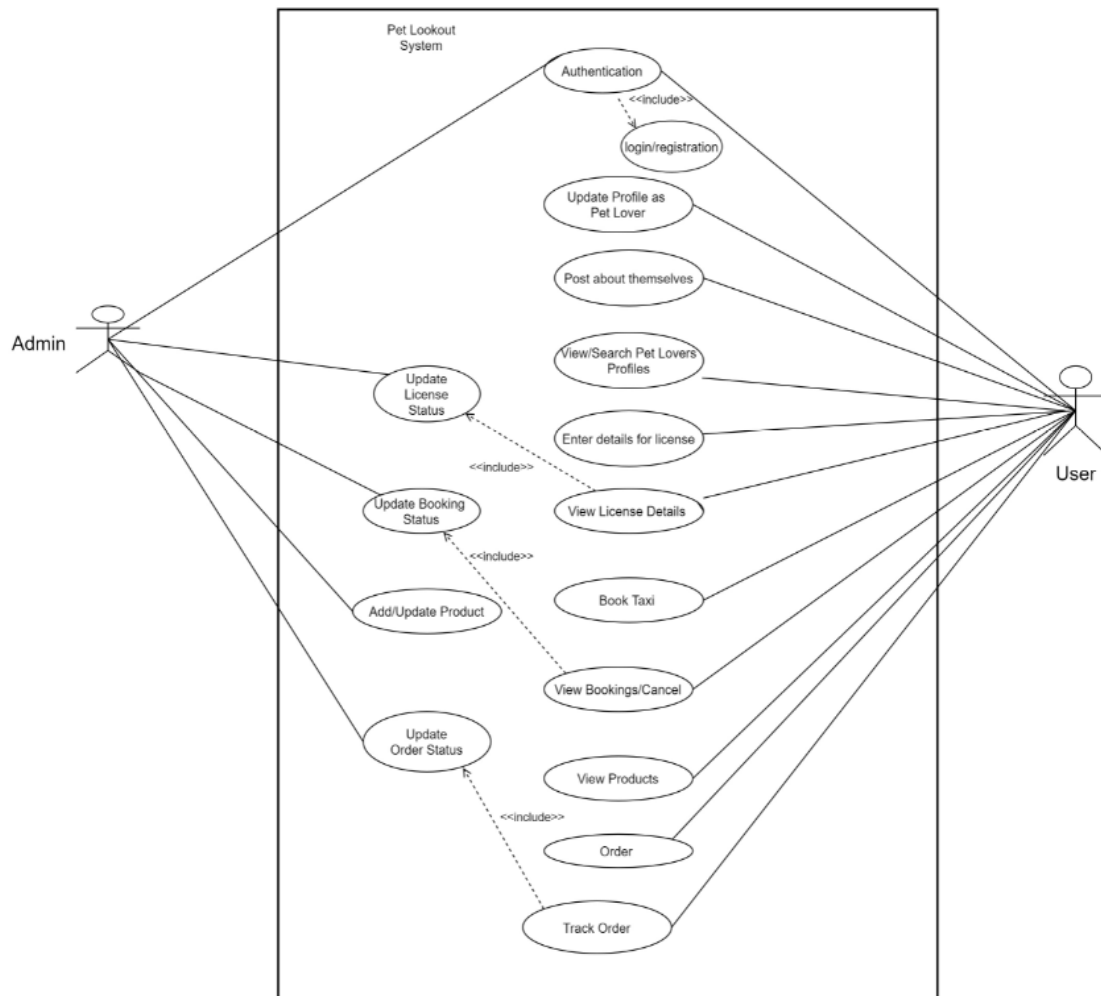


Figure 6.1 Use case Diagram

The above figure 6.1 explains the detailed process flow of Pet Lookout between the Admin and User. Admin and User are the two actors. Admin maintain all users. Admin updates the status of the bookings requested by the user such as for pet cab ,license and order of food items. Admin also adds the new products and updates it. Users can update their profile , create posts , view posts of theirs and other's ,view profiles of pet lovers ,search for different pet lovers, book cab for pets ,apply for license and keep track of the updates of pet cab and license. User can also order for the products and track their orders.

## 6.2 ADMIN ACTIVITY DIAGRAM

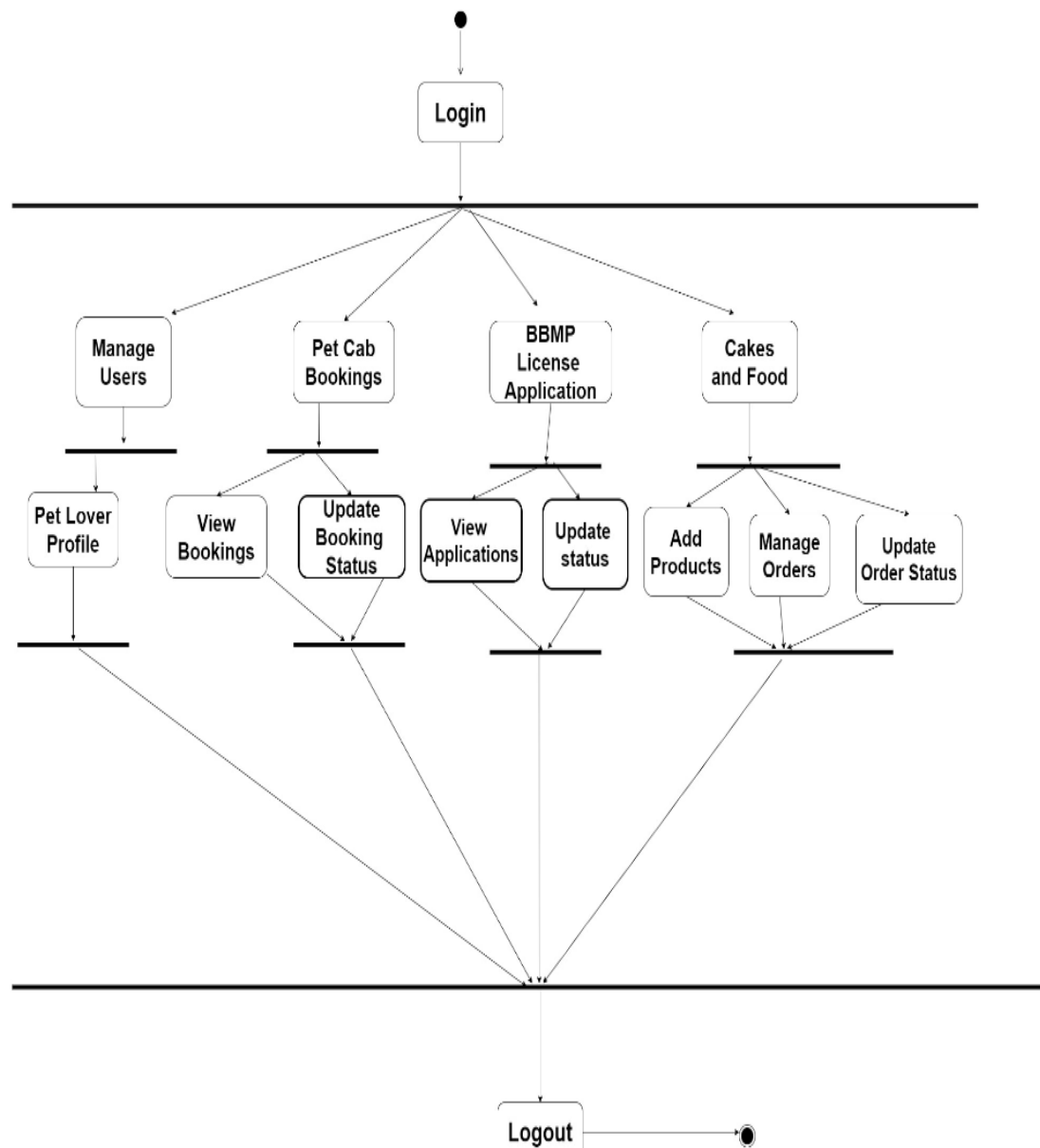


Figure 6.2 Admin Activity Diagram

The figure 6.2 explains the activity carried out by Admin. Admin manages all the registered users , pet cab bookings of the users, license applications of all the users and add products , manage orders of the users.

### 6.3 USER ACTIVITY DIAGRAM



Figure 6.3 User Activity Diagram

The figure 6.3 explains the activity carried out by User. Each user has their own profile but he/she can update it and become a pet lover. Once profile is updates user can post about themselves. They can also view the posts and profiles of others just like other normal users. User can also book pet cab and view the update of the booking ,apply for license and view the status of it too. User also browses through the food products and can make a purchase and keep track of it later.



## 6.4 DOCUMENT STRUCTURE

```

"name": "auth_user",
"fields": {
  "date_joined": {
    "type_code": "date"
  },
  "email": {
    "type_code": "string"
  },
  "first_name": {
    "type_code": "string"
  },
  "id": {
    "type_code": "int32"
  },
  "is_active": {
    "type_code": "boolean"
  },
  "is_staff": {
    "type_code": "boolean"
  },
  "is_superuser": {
    "type_code": "boolean"
  },
  "last_login": {
    "type_code": "date"
  },
  "last_name": {
    "type_code": "string"
  },
  "password": {
    "type_code": "string"
  },
  "username": {
    "type_code": "string"
  }
}
}

"name": "post_pet_lover",
"auto": {
  "field_names": ["id"],
},
"fields": {
  "address": {
    "type_code": "string"
  },
  "user_id": {
    "type_code": "int32"
  },
  "age": {
    "type_code": "int32"
  },
  "date_posted": {
    "type_code": "date"
  },
  "describe_yourself": {
    "type_code": "string"
  },
  "first_name": {
    "type_code": "string"
  },
  "hourly_rate": {
    "type_code": "int32"
  },
  "id": {
    "type_code": "int32"
  },
  "job_type": {
    "type_code": "string"
  },
  "last_name": {
    "type_code": "string"
  },
  "overnight_rate": {
    "type_code": "int32"
  },
  "pet_preference": {
    "type_code": "string"
  },
  "your_photo": {
    "type_code": "string"
  },
  "zip_code": {
    "type_code": "string"
  }
}
}

{
  "_id": {
    "Soid": "5e5eb18672f62814a39c2b9f"
  },
  "name": "post_petpost",
  "fields": {
    "date_posted": {
      "type_code": "date"
    },
    "interests": {
      "type_code": "string"
    },
    "pet_lover_id": {
      "type_code": "int32"
    },
    "petpost_id": {
      "type_code": "int32"
    },
    "photo_with_pets": {
      "type_code": "string"
    },
    "working_style": {
      "type_code": "string"
    },
    "id": {
      "type_code": "int32"
    }
  },
  "auto": {
    "field_names": ["id"],
  }
}

"name": "petcab_petcab",
"auto": {
  "field_names": ["id"],
},
"fields": {
  "depart_at": {
    "type_code": "date"
  },
  "destination": {
    "type_code": "string"
  },
  "id": {
    "type_code": "int32"
  },
  "message": {
    "type_code": "string"
  },
  "name": {
    "type_code": "string"
  },
  "no_of_travellers": {
    "type_code": "int32"
  },
  "pet_owner_id": {
    "type_code": "int32"
  },
  "phone_no": {
    "type_code": "string"
  },
  "ride_date": {
    "type_code": "date"
  },
  "source": {
    "type_code": "string"
  },
  "status": {
    "type_code": "string"
  }
}
}

"name": "shop_orders",
"auto": {
  "field_names": ["order_id"],
  "seq": {
    "$numberInt": "2"
  }
},
"fields": {
  "address": {
    "type_code": "string"
  },
  "amount": {
    "type_code": "int32"
  },
  "city": {
    "type_code": "string"
  },
  "email": {
    "type_code": "string"
  },
  "items_json": {
    "type_code": "string"
  },
  "name": {
    "type_code": "string"
  },
  "order_id": {
    "type_code": "int32"
  },
  "phone": {
    "type_code": "string"
  },
  "state": {
    "type_code": "string"
  },
  "zip_code": {
    "type_code": "string"
  }
}
}

"name": "shop_orderupdate",
"auto": {
  "field_names": ["update_id"],
},
"fields": {
  "order_id": {
    "type_code": "int32"
  },
  "timestamp": {
    "type_code": "date"
  },
  "update_desc": {
    "type_code": "string"
  },
  "update_id": {
    "type_code": "int32"
  }
}
}

"name": "shop_product",
"auto": {
  "field_names": ["id"],
  "seq": {
    "$numberInt": "3"
  }
},
"fields": {
  "category": {
    "type_code": "string"
  },
  "desc": {
    "type_code": "string"
  },
  "id": {
    "type_code": "int32"
  },
  "image": {
    "type_code": "string"
  },
  "price": {
    "type_code": "int32"
  },
  "product_name": {
    "type_code": "string"
  },
  "pub_date": {
    "type_code": "date"
  },
  "subcategory": {
    "type_code": "string"
  }
}
}

"name": "bbmp_license",
"auto": {
  "field_names": ["id", "id"],
},
"fields": {
  "address": {
    "type_code": "string"
  },
  "address_proof": {
    "type_code": "string"
  },
  "application_preference": {
    "type_code": "string"
  },
  "first_name": {
    "type_code": "string"
  },
  "id": {
    "type_code": "int32"
  },
  "last_name": {
    "type_code": "string"
  },
  "pet_age": {
    "type_code": "number"
  },
  "pet_name": {
    "type_code": "string"
  },
  "user_id": {
    "type_code": "int32"
  },
  "vaccination_record": {
    "type_code": "string"
  },
  "message": {
    "type_code": "string"
  },
  "old_license": {
    "type_code": "string"
  },
  "pet_breed": {
    "type_code": "string"
  }
}
}

"pet_colour": {
  "type_code": "string"
},
"vaccination_record": {
  "type_code": "string"
},
"message": {
  "type_code": "string"
},
"old_license": {
  "type_code": "string"
},
"pet_breed": {
  "type_code": "string"
},
"pet_colour": {
  "type_code": "string"
},
"pet_gender": {
  "type_code": "string"
},
"pet_type": {
  "type_code": "string"
},
"phone": {
  "type_code": "string"
},
"status": {
  "type_code": "string"
}
}


```

Figure 6.4 Document Structure

In the Fig 6.4 auth\_user, pet\_lover, post\_petlover, petcab\_petcab , bbmp\_license , shop\_product , shop\_order and shop\_orderupdate are the collections used for both inserting and fetching data.

# **IMPLEMENTATION**

## 7. IMPLEMENTATION

Implementation describes how the concept of the proposed system is converted into an executable working system. The implementation process refers to the design phase for building the system efficiently. The system implements all the requirements that are included in the design.

### 7.1 SOURCE CODE

#### **User registration:**

```
class UserRegisterForm(UserCreationForm):
    email = forms.EmailField()
class Meta:
    model = User
    fields = ['username', 'email', 'password1', 'password2']
class UserUpdateForm(forms.ModelForm):
    email = forms.EmailField()
class Meta:
    model = User
    fields = ['username', 'email']
```

#### **Views for Registration and Login:**

```
def register(request):
    if request.method == 'POST':
        form = UserRegisterForm(request.POST)
        if form.is_valid():
            form.save()
            username = form.cleaned_data.get('username')
            messages.success(request, "Your account has been
created! You are now able to log in.")
            return redirect('login')
```

```
    else:
        form = UserRegisterForm()
    return render(request, 'users/register.html', {'form':
form})
```

### **User Update Profile:**

```
class Profile(models.Model):
    user = models.OneToOneField(User, on_delete=models.CASCADE)
    image = models.ImageField(default='default.jpg',
upload_to='profile_pics')
    first_name = models.CharField(max_length=100,default="")
    last_name = models.CharField(max_length=100,blank=True)
    address = models.CharField(max_length=100,default="")
    age = models.IntegerField(null=True)
    zip_code = models.CharField(max_length=100,null=True)
    phone = models.CharField(max_length=10,
blank=True,null=True)
    describe_yourself = models.TextField(default="")
    experience = models.IntegerField(null=True)
    hourly_rate = models.IntegerField(null=True)
    overnight_rate = models.IntegerField(null=True)

    BOOL_CHOICES = ((True, 'Yes'), (False, 'No'))
    dog_firstaid_and_or_CPR =
models.BooleanField(choices=BOOL_CHOICES, blank=True,null=True)
    bol_choices = ((True, 'Yes'), (False, 'No'))
    member_of_any_organisation =
models.BooleanField(choices=bol_choices, blank=True,null=True)
```

```
organisation_name = models.CharField(max_length=50, blank=True)

PETSITTER = 'PETSITTING'

WALKER = 'WALKING'

TRAINER = 'TRAINING'
ALL = 'FULL'
    jo_type = [
        (PETSITTER, 'Pet Sitter'),
        (WALKER, 'Walker'),
        (TRAINER, 'Trainer'),
        (ALL, 'FULL'),
    ]
    job_type = models.CharField(max_length=10, blank=True,
choices=jo_type, default=ALL, )
    MALE = 'M'
    FEMALE = 'F'
    gen_type = [
        (MALE, 'Male  '),
    ]
    gender = models.CharField(max_length=10, blank=True,
choices=gen_type, default=MALE, )
    DOG = 'DOGS'
    CAT = 'CATS'

BOTH = 'DOGS&CAT'
    pet_prefer = [
        (DOG, 'Dog'),
        (CAT, 'Cat'),
```

```

        (BOTH, 'Both'),
    ]
    pet_preference = models.CharField(max_length=10, blank=True,
choices=pet_prefer, default=DOG, )
    date_posted = models.DateTimeField(default=timezone.now)
    your_photos_with_pets =
models.ImageField(upload_to='profile_pics', blank=True)
    def __str__(self):
        return f'{self.user.username} '
    def get_absolute_url(self):
        return reverse('profile-detail', kwargs={'pk': self.pk})
    def save(self, force_insert=False, force_update=False,
using=None, update_fields=None):
        super().save(force_insert, force_update, using,
update_fields)

```

**Pet Lover's profile Form:**class

ProfileUpdateForm(forms.ModelForm):

```

    class Meta:
        model = Profile
        fields =
['image','first_name','last_name','address','age','zip_code','ph
one','describe_yourself','experience','hourly_rate','overnight_r
ate','dog_firstaid_and_or_CPR','member_of_any_organisation','org
anisation_name','job_type','gender','pet_preference','date_poste
d','your_photos_with_pets']

```

**View for Pet Lover profile update:**

```
@login_required
def profile(request):
    if request.method == 'POST':
        u_form = UserUpdateForm(request.POST,
instance=request.user)
        p_form = ProfileUpdateForm(request.POST,
request.FILES,
instance=request.user.profile)
        if u_form.is_valid() and p_form.is_valid():
            u_form.save()
            p_form.save()
            messages.success (request, "Your account has been
updated!")
            return redirect('profile')
        else:
            u_form = UserUpdateForm(instance=request.user)
            p_form =
ProfileUpdateForm(instance=request.user.profile)

    context = {
        'u_form': u_form,
        'p_form': p_form
    }
    return render(request, 'users/profile.html', context)
```

**Pet Lover's Post:**

```
class PetPost(models.Model):
    interests = models.CharField(max_length=150)
```



```
        working_style = models.TextField(null=True, blank=True)
        pet_lover = models.ForeignKey(to=Profile,
on_delete=models.CASCADE,null=True,          blank=True)
        date_posted = models.DateTimeField(default=timezone.now)
        photo_with_pets =
models.ImageField(upload_to='profile_pics', blank=True)
    def ____ (self):
        return self.pet_lover

    def get_absolute_url(self):          return reverse('petpost-
detail', kwargs={'pk': self.pk})
```

#### **Create/Update/Delete Views for Pet Lover's Post:**

```
@login_required
def home(request):
    context = {
        'posts': PetPost.objects.all()
    }
    return render(request, 'post/home.html', context)
@method_decorator(login_required, name="dispatch")
class PetPostCreateView(LoginRequiredMixin, CreateView):
    model = PetPost
    fields = ["interests", "working_style", "photo_with_pets"]

    def form_valid(self, form):
        form.instance.pet_lover = self.request.user.profile
        return super().form_valid(form)
```

```
class PetPostListView(ListView):
    model = PetPost
    template_name = 'post/home.html' #
<app>/<model>_<viewtype>.html
    context_object_name = 'posts'
    ordering = ['-date_posted']
    paginate_by = 3

class PetPostUpdateView(UpdateView):
    model = PetPost
    fields = ["interests", "working_style", "photo_with_pets"]

    def form_valid(self, form):
        form.instance.pet_lover = self.request.user.profile
        return super().form_valid(form)
def test_func(self):
    petpost = self.get_object()
    if self.request.user == petpost.pet_lover:
        return True
    return False
@method_decorator(login_required, name="dispatch")
class PetPostDetailView(DetailView):
    model = PetPost
```

**Pet Cab Models:**

```
from django.core.validators import MinValueValidator,
MaxValueValidator
class Petcab(models.Model):
    pet_owner = models.ForeignKey(settings.AUTH_USER_MODEL,
```

```
on_delete=models.CASCADE)
    name = models.CharField(max_length=30)
    source = models.CharField(max_length=100)
    destination = models.CharField(max_length=100)
    depart_at = models.TimeField()
    ride_date = models.DateField()
    no_of_travellers =
models.IntegerField(validators=[MinValueValidator(2),
                                MaxValueValidator(4)])

    BOOKED = 'Booked'
    PENDING = 'Pending'
TICKET_STATUSES = ((BOOKED, 'Booked'),
                    (PENDING, 'Pending'),)

    status =
models.CharField(choices=TICKET_STATUSES,max_length=7,
default='Pending')
    message = models.CharField(max_length=1000, default="Please
Check Again After Some Time")
    phone_no = models.CharField(max_length=10, blank=True,
null=True)
    def __str__(self):
        return f'{self.pet_owner} Petcab'
    def get_absolute_url(self,**kwargs):
        return reverse('petcab-detail', kwargs={'pk': self.pk})
```

**Pet Cab Booking Form:**

```
from .models import Petcab
from bootstrap_datepicker_plus import DatePickerInput
```

```
class DateInput(forms.DateInput):
    input_type = 'date'
class TimeInput(forms.TimeInput):
    input_type = 'time'
class PetcabForm(forms.ModelForm):
    class Meta:
        model = Petcab
        fields = [ 'name','source',
'destination','depart_at','ride_date','no_of_travellers',
'phone_no']
    widgets = {
        'ride_date': DateInput(),
        'depart_at': TimeInput(),
    }
```

**Views for Pet Cab Booking and Cancel:**

```
def booking(request):
    context = {
        'petcabs': Petcab.objects.all(),
    }
    return render(request, 'petcab/booking.html', context)

class PetcabCreateView(SuccessMessageMixin, CreateView):
    model = Petcab
    form_class = PetcabForm
    success_url = '/'
    def form_valid(self, form):
        form.instance.pet_owner = self.request.user
        return super().form_valid(form)
```

```
def get_success_message (self, cleaned_data):
    print(cleaned_data)
    return "Your pet car has been booked!"

@method_decorator(login_required, name="dispatch")
class PetcabDeleteView(DeleteView):
    model = Petcab
    success_url = '/'

    def test_func(self):
        petcab = self.get_object()
        if self.request.user == petcab.pet_owner:
            return True
        return False
```

**Booking Status:**

```
@method_decorator(login_required, name="dispatch")
class PetcabDetailView(DetailView):
    model = Petcab

    def test_func(self):
        petcab = self.get_object()
        if self.request.user == petcab.pet_owner:
            return True
        return False
```

**Shop Models:**

```
class Product(models.Model):
    product_name = models.CharField(max_length=50)
    category = models.CharField(max_length=50, default="")
```

```
    subcategory = models.CharField(max_length=50, default="")
    price = models.IntegerField(default=0)
    desc = models.CharField(max_length=300)
    pub_date = models.DateField()
    image = models.ImageField(upload_to='profile_pics',
default="")

    def __str__(self):
        return self.product_name

class Orders(models.Model):
    order_id = models.AutoField(primary_key=True)
    items_json = models.CharField(max_length=5000)
    amount = models.IntegerField( default=0)
    name = models.CharField(max_length=90)
    email = models.CharField(max_length=111)
    address = models.CharField(max_length=111)
    city = models.CharField(max_length=111)
    state = models.CharField(max_length=111)
    zip_code = models.CharField(max_length=111)
    phone = models.CharField(max_length=111, default="")
    def __str__(self):
        return self.name

class OrderUpdate(models.Model):
    update_id  = models.AutoField(primary_key=True)
    order_id = models.IntegerField(default="")
    update_desc = models.CharField(max_length=5000)
    timestamp = models.DateField(auto_now_add=True)
```

```
def __str__(self):  
    return self.update_desc[0:7] + "..."
```

**Views for Shop:**

```
@login_required()  
def index(request):  
    allProds = []  
    catprods = Product.objects.values('category', 'id')  
    cats = {item['category'] for item in catprods}  
    for cat in cats:  
        prod = Product.objects.filter(category=cat)  
        n = len(prod)  
        nSlides = n // 4 + ceil((n / 4) - (n // 4))  
        allProds.append([prod, range(1, nSlides), nSlides])  
    params = {'allProds':allProds}  
    return render(request, 'shop/index.html', params)  
def searchMatch(query, item):  
    if query in item.desc.lower() or query in  
item.product_name.lower() or query in item.category.lower():  
        return True  
    else:  
        return False  
def search(request):  
    query = request.GET.get('search')  
    allProds = []  
    catprods = Product.objects.values('category', 'id')  
    cats = {item['category'] for item in catprods}  
    for cat in cats:  
        prodtemp = Product.objects.filter(category=cat)
```

```
        prod = [item for item in prodtemp if searchMatch(query,
item)]
        n = len(prod)
        nSlides = n // 4 + ceil((n / 4) - (n // 4))
        if len(prod) != 0:
            allProds.append([prod, range(1, nSlides), nSlides])
        params = {'allProds': allProds, "msg": ""}
        if len(allProds) == 0 or len(query)<4:
            params = {'msg': "Please make sure to enter relevant
search query"}
        return render(request, 'shop/search.html', params)

def aboutshop(request):
    return render(request, 'shop/aboutshop.html')
def productView(request, myid):
    product = Product.objects.filter(id=myid)
    return render(request, 'shop/prodView.html',
{'product':product[0]})
```

**Tracking Order:**

```
def tracker(request):
    if request.method=="POST":
        orderId = request.POST.get('orderId', '')
        email = request.POST.get('email', '')
        try:
            order = Orders.objects.filter(order_id=orderId,
email=email)
            if len(order)>0:
                update =
OrderUpdate.objects.filter(order_id=orderId)
```



```
        updates = []
        for item in update:
            updates.append({'text': item.update_desc,
'            'time': item.timestamp})
            response = json.dumps({"status": "success",
"updates": updates, "itemsJson": order[0].items_json},
default=str)
            return HttpResponse(response)
        else:
            return HttpResponse({'"status": "noitem"}')
    except Exception as e:
        return HttpResponse({'"status": "error"}')
    return render(request, 'shop/tracker.html')
```

**Checkout:**

```
def checkout(request):
    if request.method=="POST":
        items_json = request.POST.get('itemsJson', '')
        name = request.POST.get('name', '')
        amount = request.POST.get('amount', '')
        email = request.POST.get('email', '')
        address = request.POST.get('address1', '') + " " +
request.POST.get('address2', '')
        city = request.POST.get('city', '')
        state = request.POST.get('state', '')
        zip_code = request.POST.get('zip_code', '')
        phone = request.POST.get('phone', '')
        order = Orders(items_json=items_json, name=name,
email=email, address=address, city=city,
```

```
state=state, zip_code=zip_code, phone=phone, amount=amount)
    order.save()
    update = OrderUpdate(order_id=order.order_id,
update_desc="The order has been placed")
    update.save()
    thank = True
    id = order.order_id
    return render(request, 'shop/checkout.html',
{'thank':thank, 'id': id})
    return render(request, 'shop/checkout.html')
```

**BBMP License Models:**

```
class License(models.Model):
    user = models.ForeignKey(User, on_delete=models.CASCADE)
    first_name = models.CharField(max_length=100)
    last_name = models.CharField(max_length=100)
    pet_name = models.CharField(max_length=100)
    pet_colour = models.CharField(max_length=100)
    pet_breed = models.CharField(max_length=100)
    pet_age = models.FloatField()
    address = models.CharField(max_length=100)
    phone = models.CharField(max_length=10,
blank=True,null=True)
    MALE = 'MA'
    FEMALE = 'FE'
    pet_gen = [
        (MALE, 'Male '),
        (FEMALE, 'Female'),
    ]
```

```
    pet_gender = models.CharField(max_length=10, blank=True,
choices=pet_gen, default=MALE, )
    DOG= 'DG'
    CAT = 'CT'
    pettype_sel = (
        (DOG, "Dog"),
        (CAT, "Cat"),
    )
pet_type = models.CharField(max_length=2,
blank=True,choices=pettype_sel,default = DOG, )
    RENEW = 'RNW'
    NEW = 'NW'
    appl_prefer = [
        (RENEW, 'Renew'),
        (NEW, 'New'), ]
    application_preference = models.CharField(max_length=5,
blank=True, choices=appl_prefer, default=NEW, )
old_license = models.FileField(upload_to='media',blank=True)
    vaccination_record = models.FileField(upload_to='media')
    address_proof = models.FileField(upload_to='media')
    def __str__(self):
        return self.first_name

def get_absolute_url(self):
    return reverse('license-detail', kwargs={'pk':
self.pk})
```

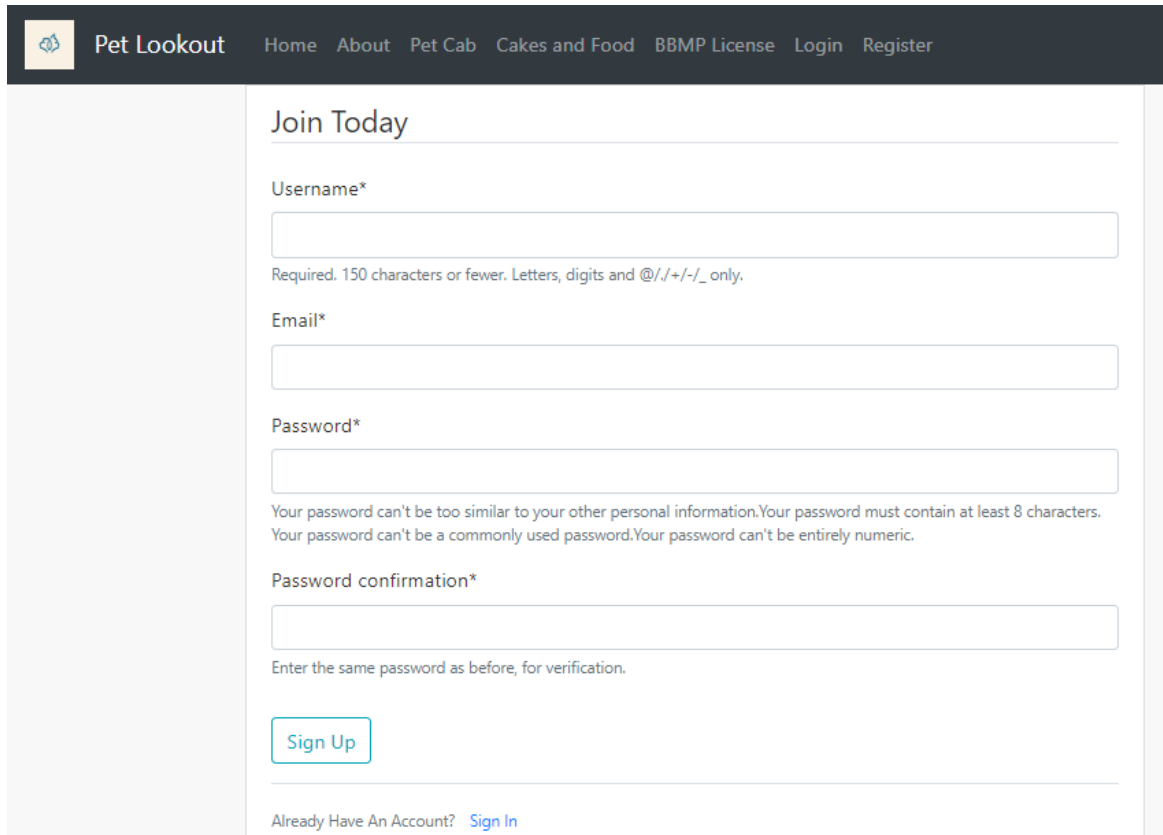
**License views:**

```
class LicenseCreateView(SuccessMessageMixin, CreateView):
    model = License
    fields =
['first_name', 'last_name', 'pet_name', 'pet_colour', 'pet_breed', 'p
et_age', 'address', 'phone', 'pet_gender', 'pet_type', 'application_p
reference',

'old_license', 'vaccination_record', 'address_proof']
    success_url = '/home/'
    def form_valid(self, form):
        form.instance.user = self.request.user
        return super().form_valid(form)

def get_success_message(self, cleaned_data):
    print(cleaned_data)
    return "Application Submitted Successfully, We will
contact you for further updates!"
```

## 7.2 SCREENSHOTS



The screenshot shows the 'Join Today' registration form on the Pet Lookout website. The form includes fields for Username, Email, Password, and Password confirmation. The Username field has a requirement of 150 characters or fewer, using letters, digits, and @/./+/\_ only. The Password field has requirements: it must be at least 8 characters, not too similar to other personal information, not a commonly used password, and not entirely numeric. The Password confirmation field requires the same password as before for verification. A 'Sign Up' button is at the bottom of the form. Below the form, there is a link 'Already Have An Account? Sign In'.

**Join Today**

Username\*

Required. 150 characters or fewer. Letters, digits and @/./+/\_ only.

Email\*

Password\*

Your password can't be too similar to your other personal information. Your password must contain at least 8 characters. Your password can't be a commonly used password. Your password can't be entirely numeric.

Password confirmation\*

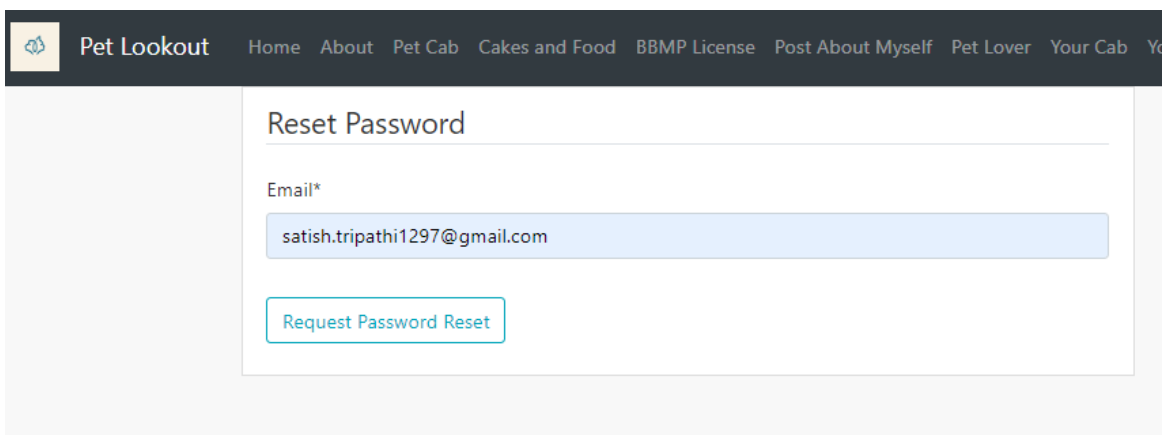
Enter the same password as before, for verification.

[Sign Up](#)

Already Have An Account? [Sign In](#)

Figure 7.2.1 Pet Lookout Registration Screen

The above figure 7.2.1 is Pet Lookout registration screen. New users can register from here and registered user can click on sign in now to login.



The screenshot shows the 'Reset Password' page on the Pet Lookout website. It includes a form with an 'Email\*' field containing the email address 'satish.tripathi1297@gmail.com'. Below the email field is a 'Request Password Reset' button. The page header shows the Pet Lookout logo and navigation links: Home, About, Pet Cab, Cakes and Food, BBMP License, Post About Myself, Pet Lover, Your Cab, and Your Profile.

**Reset Password**

Email\*

satish.tripathi1297@gmail.com

[Request Password Reset](#)

Figure 7.2.2 Password Reset page

The image shows two parts of a password reset process. The top part is an email from 'petlookout6918@gmail.com' to 'me'. The email body states: 'You're receiving this email because you requested a password reset for your user account at 127.0.0.1:8000. Please go to the following page and choose a new password: http://127.0.0.1:8000/password-reset-confirm/NDg/5gd-9bcf8f9e189d94354ee7/. Your username, in case you've forgotten: Rashford'. The bottom part is a web form titled 'Reset Password'. It has a header with 'Pet Lookout' and navigation links: Home, About, Pet Cab, Cakes and Food, BBMP License, Post About Myself, Pet Lover, Your Cab. The form contains two input fields: 'New password\*' and 'New password confirmation\*', both masked with dots. Below the first field is a message: 'Your password can't be too similar to your other personal information. Your password must contain at least 8 characters. Your password can't be a commonly used password. Your password can't be entirely numeric.' At the bottom of the form is a 'Reset Password' button. The footer of the web page shows 'PET LOVER' and 'PET SERVICES'.

**Pet Lookout** Home About Pet Cab Cakes and Food BBMP License Post About Myself

An email has been sent with instructions to reset your password

**petlookout6918@gmail.com**  
to me ▾

You're receiving this email because you requested a password reset for your user account at [127.0.0.1:8000](http://127.0.0.1:8000).

Please go to the following page and choose a new password:

<http://127.0.0.1:8000/password-reset-confirm/NDg/5gd-9bcf8f9e189d94354ee7/>

Your username, in case you've forgotten: Rashford

**Pet Lookout** Home About Pet Cab Cakes and Food BBMP License Post About Myself Pet Lover Your Cab

### Reset Password

New password\*

.....

Your password can't be too similar to your other personal information. Your password must contain at least 8 characters.  
Your password can't be a commonly used password. Your password can't be entirely numeric.

New password confirmation\*

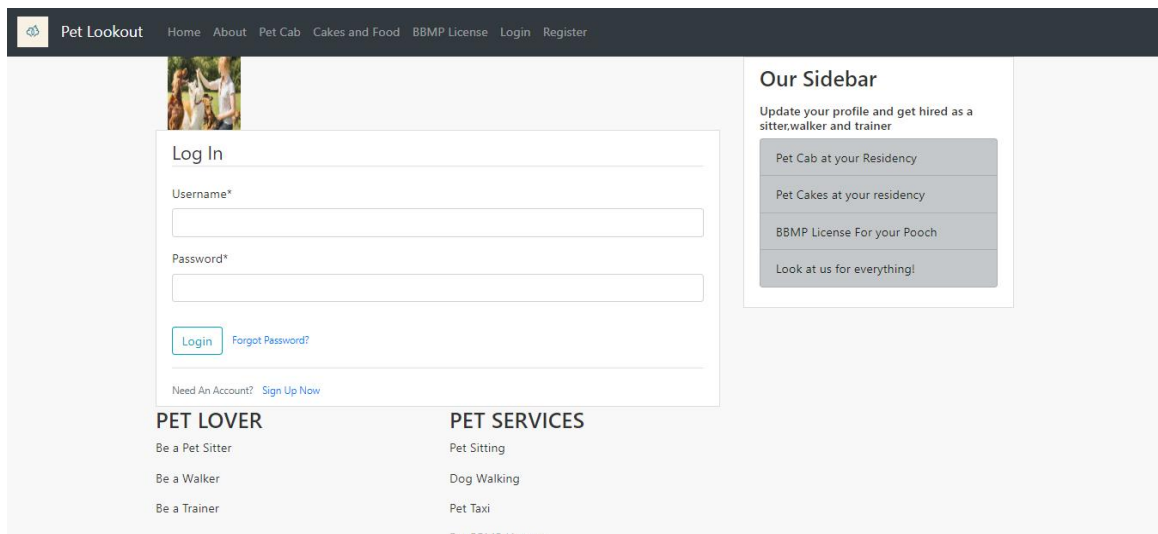
.....

Reset Password

PET LOVER PET SERVICES

Figure 7.2.3 Password reset screen.

Figure 7.2.3 shows how the user can reset the password by receiving a link to their email requesting them to reset the password.



The screenshot displays the Pet Lookout website's login interface. At the top, a dark navigation bar contains the site logo and links for Home, About, Pet Cab, Cakes and Food, BBMP License, Login, and Register. The main content area features a 'Log In' form with fields for 'Username\*' and 'Password\*', a 'Login' button, and a 'Forgot Password?' link. Below the form, there are links for 'Need An Account?' and 'Sign Up Now'. To the right, a 'Our Sidebar' section prompts users to 'Update your profile and get hired as a sitter,walker and trainer', with buttons for 'Pet Cab at your Residency', 'Pet Cakes at your residency', 'BBMP License For your Pooch', and 'Look at us for everything!'. At the bottom, two columns list services: 'PET LOVER' (Be a Pet Sitter, Be a Walker, Be a Trainer) and 'PET SERVICES' (Pet Sitting, Dog Walking, Pet Taxi, BBMP License).

Figure 7.2.4 Pet Lookout Login Screen

The registered users can login by entering the username and password used by them during the registration process.


Pet Lookout
Home About Pet Cabi Cakes and Food BOMP License Post About Myself Pet Lover Your Cabi Your License Logout

## About Us

In this day and age, life is understandably hectic for many pet parents. With commitments at work, time out with friends and family, full time attention to our pets are usually not a possibility. So whether you are having late nights at work, out on a holiday or simply need a break and need pet sitting, in home boarding, daycare, dog walking or even pet training, Pet Lookout is here to connect you with other pet lovers who are ready to help you take care of your pets like it were part of their family, belly rubs and hugs included.



Pet Lookout sitters and pet service providers are normal people who love pets, just like you. They are pet people who are dedicated to put time into giving your pet a day worth barking or meowing about, if you've got that sort of pet. Your pet stays in a home, free to roam and explore, rather than in the confines of a 3 foot cage typical in kennels or pet hotels. You wouldn't want to be locked in a cage when your other half is away! Pet Lookout sitters give your pet much deserved attention while you are away. Compare that to rows and rows of pets trying to seek attention from just a few attendants in a kennel or pet shop. Some of our sitters even go to great lengths, giving your pets' treats and delicious home cooked food.



### Cakes as Treats

Our furry friends bring us so much joy that we hardly need an excuse to spoil them! However when their birthday or a special holiday like Love Your Pet Day (February 20th) rolls around, many of us want to treat them to something special. That's where these dog cake recipes comes in! Dog cakes (or pupcakes, as some like to call them) are a tasty way to celebrate. They also make for a great photo op. Before you get started, we'll walk you through the ingredients that are healthiest for your pup.

[Go to Shop »](#)

### Joy for furry mate

Whether it's your dog's birthday, they just graduated from training school or you just want to treat them to a little something special, dog cakes will do the trick. While they're enjoying their pupcake, treat yourself something special out of a personalized pet mug. Even better, take it to work to remind you of your furry friend all day.

[View Shop »](#)

### Contact Us

Pet Lookout Inc. No.48, Mantri Elite, Jp nagar, Bangalore 560076, India.  
 Phone: +91 91484 51003 Phone Email: petlookout8915@gmail.com/  
 © 2020 Pet Lookout Inc. All Rights Reserved

Figure 7.2.5 Pet Lookout about page

Figure 7.2.5 briefs up user about how the website works and the services provided on the website and the process of working .



The screenshot displays the 'Pet Lookout' website interface. At the top, a navigation bar includes links for Home, About, Pet Cabs, Cakes and Food, BBMP License, Post About Myself, Pet Lover, Your Cab, Your License, and Logout. The main heading is 'Here are our Pet Lovers'. Below this, there's a search bar and a 'See Pet Lover's' button. The first post is by 'anthony' dated May 06, 2020, titled 'Dogs and Cats and Communication'. It discusses the importance of understanding pet behavior and includes a photo of a dog and a person. The second post is by 'Freshford' dated May 07, 2020, titled 'Cats', offering pet sitting and training services with a photo of a person holding a cat. The third post is by 'Jalke' dated May 07, 2020, titled 'Dogs Pet Sitting , Training.', also offering services with a photo of a person with a dog. To the right, an 'Our Sidebar' section prompts users to update their profiles and lists services like Pet Cab, Pet Cakes, BBMP License, and a 'Look at us for everything!' button. At the bottom, there are sections for 'PET LOVER' (Be a Pet Sitter, Be a Walker, Be a Trainer) and 'PET SERVICES' (Pet Sitting, Dog Walking, Pet Taxi, Pet BBMP License).

Figure 7.2.6 Posts posted by Pet Lovers

The page consists of all the posts posted by all the pet lovers in which they have posted about themselves which include their interests and the way in which they will be taking care of the pets and their experiences with other pet including a picture of themselves with a pet. Users can also move from one post page to another with the help of paginations and just by clicking on the name of the pet lover they will be directed to the detail view of their profile.

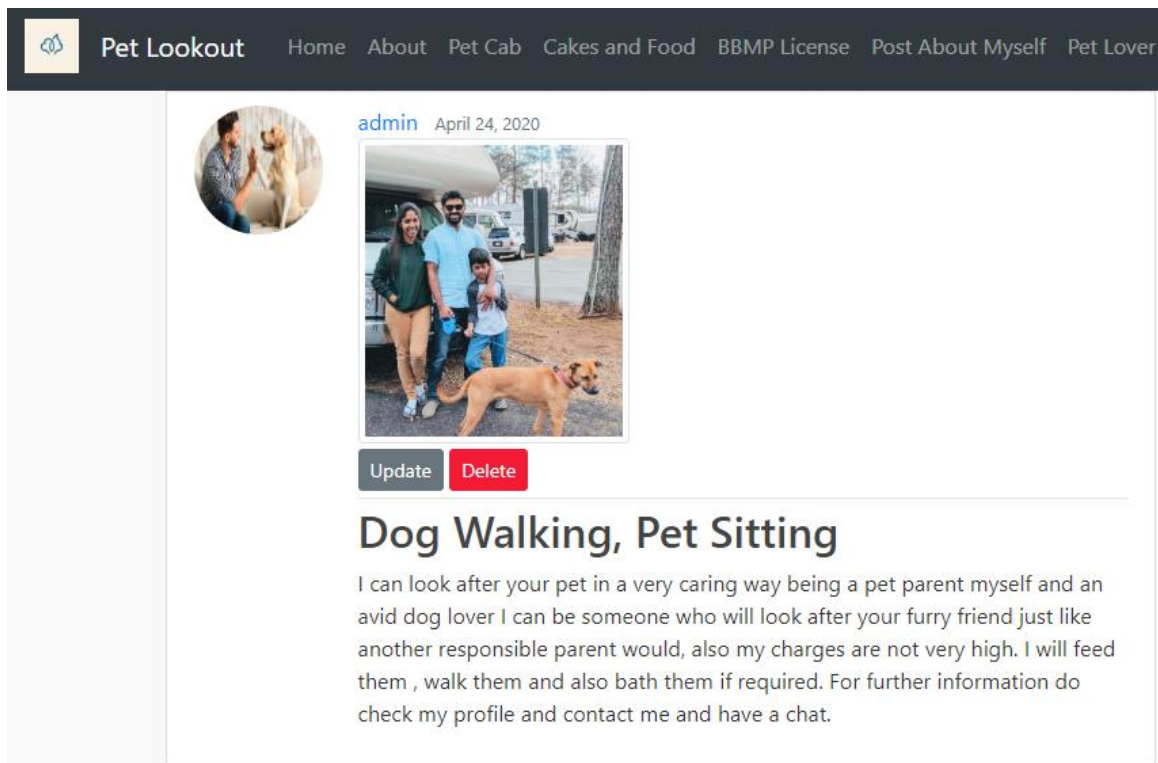
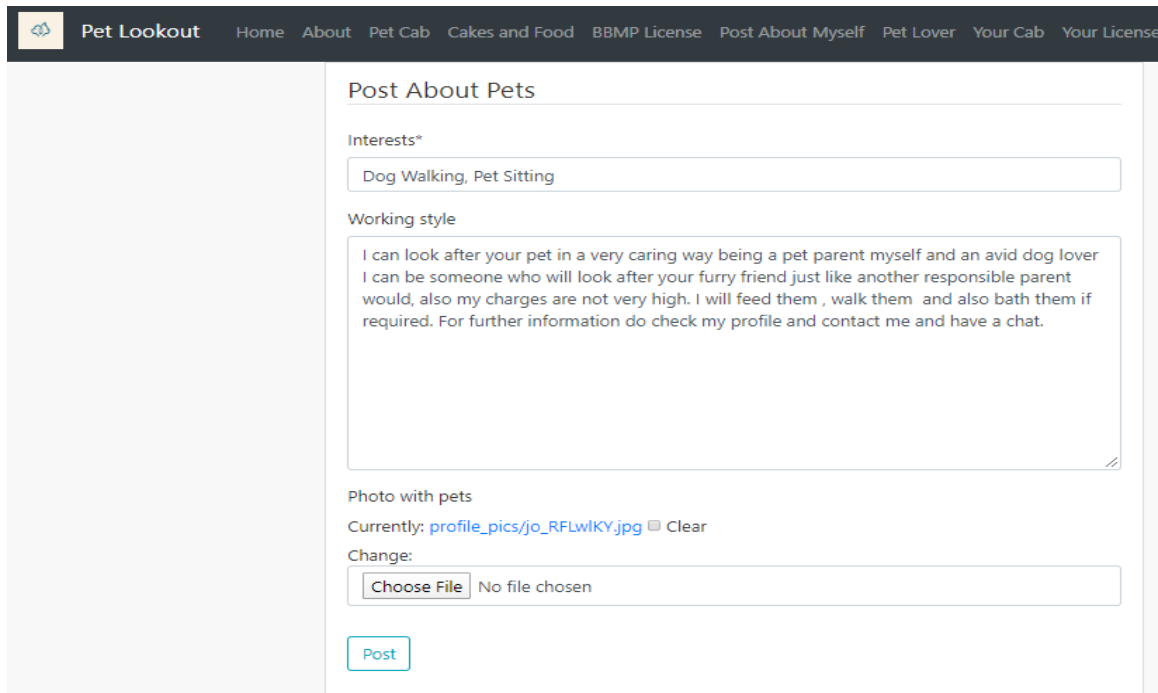


Figure 7.2.7 Pet Lovers individual post

In the figure above , pet lover can update and delete the post posted by him.



**Pet Lookout** Home About Pet Cab Cakes and Food BBMP License Post About Myself Pet Lover Your Cab Your License

### Post About Pets

Interests\*

Dog Walking, Pet Sitting

Working style

I can look after your pet in a very caring way being a pet parent myself and an avid dog lover I can be someone who will look after your furry friend just like another responsible parent would, also my charges are not very high. I will feed them , walk them and also bath them if required. For further information do check my profile and contact me and have a chat.

Photo with pets


Currently: [profile\\_pics/jo\\_RFLwIKY.jpg](#) Clear


Change:

No file chosen

Figure 7.2.8 Pet Lover posting post

In figure 7.2.8 Pet lover posts about pet related topics and attach a picture of pet


[Home](#)
[About](#)
[Pet Cab](#)
[Cakes and Food](#)
[BBMP License](#)
[Post About Myself](#)
[Pet Lover](#)



## Martial Anthony

anthony@gmail.com

DOGS&CAT

I will gladly walk or run a dog at your neighbourhood or parks if there are any closeby, let it do it's business and clean up. Walks will be 45 mins to 1 hour depending on the health and age of the dog. I will also ensure the dog is in a safe environment at all times as trust is utmost important when this service is being carried out. I grew up having dogs all my life, They are stress relievers and anyone of you out there who owns a pet knows you only want the best for them. They are as much humans and needs the best care and love. I will be more than happy to make a dogs day by giving him/her daily walks. I owned a German Shepard when I was growing up. After that several mix breeds and lastly a shih tzu. Just like many dog owners, I cried for days when it was time for them to cross the rainbow bridge. It's just too painful to endure losing them so I have decided it's better I only baby sit others' pet than owning my own for now.

### Hi there Pet Parent I am a DOGS&CAT Person

Age :: 24

Gender :: Female

Address :: C 205 Kripa Nagar

Job Type :: Walker

Phone No :: 953870661

Email :: anthony@gmail.com

Pet Preference :: DOGS&CAT

Session(Hourly) Charges :: 150


Organisation I joined :: Ekta

Overnight Rate :: 0

Organisation Member(If any):: True

Pet First Aid or CPR :: True

My Photo with my pet::



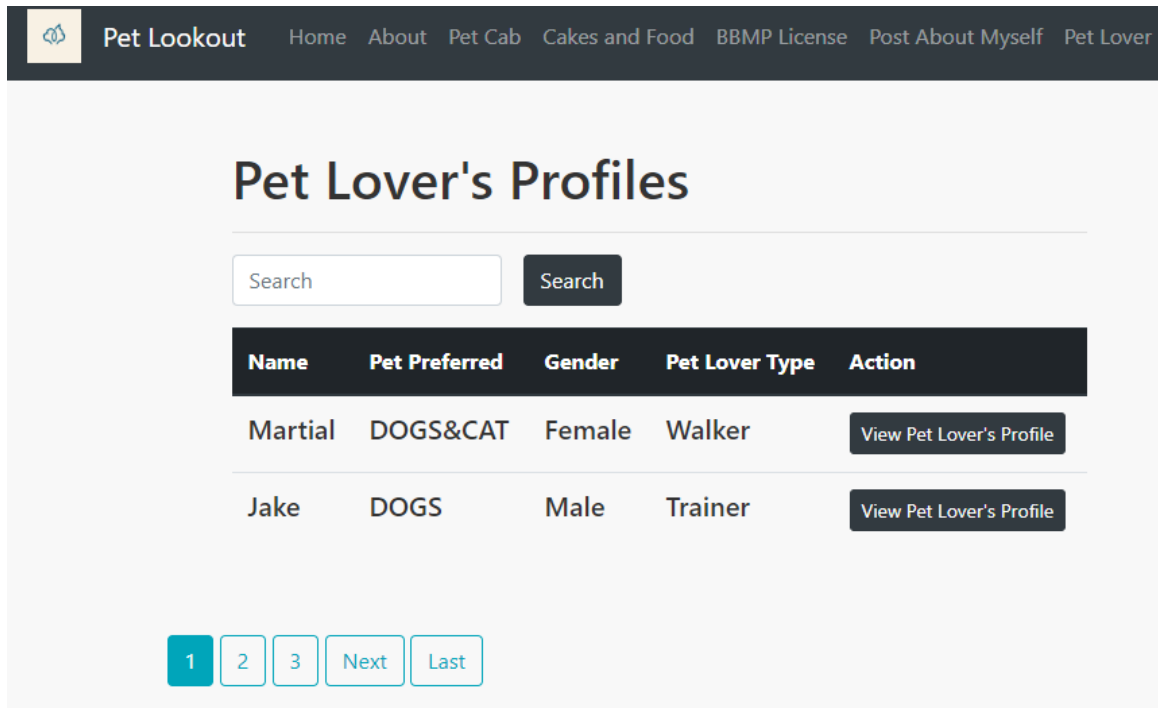
### General FAQs

Q: How does this work?

**Answer:**  
View the profile of our pet lovers and whomeveryou find it to be more appealing and good for pet contact them and give your requirements details and the availability of them.

Figure 7.2.9 Pet Lover's profile

Figure 7.2.9 shows the detail view of the pet lover describing his/her abilities and specialities.

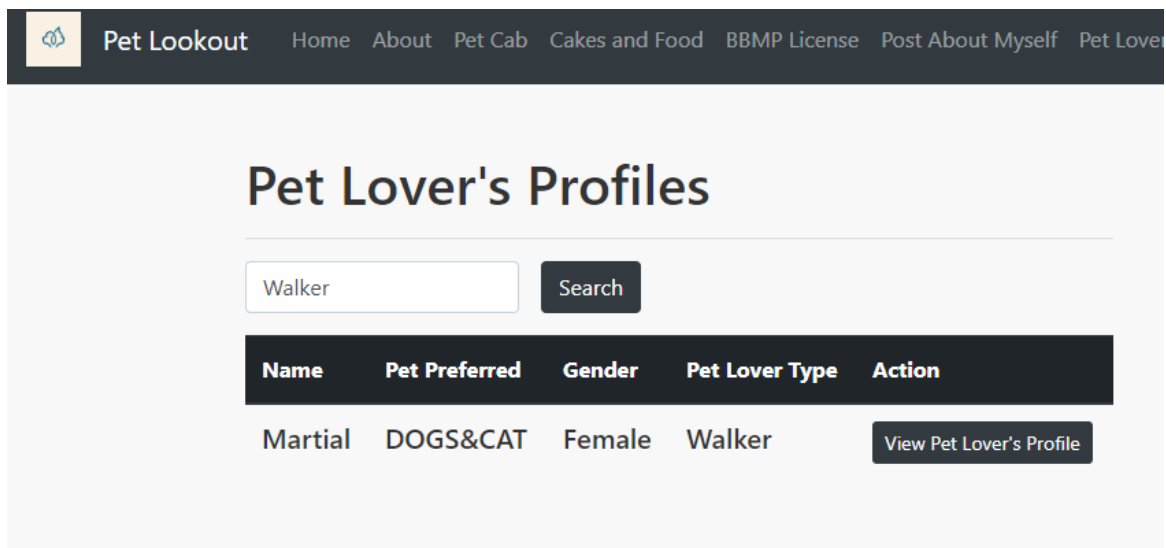


The screenshot shows the 'Pet Lookout' website with a dark navigation bar containing links: Home, About, Pet Cab, Cakes and Food, BBMP License, Post About Myself, and Pet Lover. The main heading is 'Pet Lover's Profiles'. Below it is a search bar with the text 'Search' and a 'Search' button. A table lists two profiles:

Name	Pet Preferred	Gender	Pet Lover Type	Action
Martial	DOGS&CAT	Female	Walker	<a href="#">View Pet Lover's Profile</a>
Jake	DOGS	Male	Trainer	<a href="#">View Pet Lover's Profile</a>

At the bottom, there are pagination controls: a teal button with '1', a white button with '2', a white button with '3', a white button with 'Next', and a white button with 'Last'.

Figure 7.2.10 List of Pet Lover's profiles




The screenshot shows the 'Pet Lookout' website with the same navigation bar. The search bar now contains the text 'Walker' and the 'Search' button is highlighted. The table below shows only one profile:

Name	Pet Preferred	Gender	Pet Lover Type	Action
Martial	DOGS&CAT	Female	Walker	<a href="#">View Pet Lover's Profile</a>

Figure 7.2.11 Pet Lover's profile

The above figures 7.2.10 and 7.2.11 comprises of the list of pet lovers and their profiles where user can search for them accordingly by pet lover's type for example a 'Walker' in this case.



**Jake**  
jake12@gmail.com

**Update Profile Info to become a professional**

**Username\***

Required: 150 characters or fewer. Letters, digits and @/./+/-/\_ only.

**Email\***

**Image\***

Currently: [profile\\_pics/jas\\_QDmchL2.jpg](#)

Change:

No file chosen

**Age\***

**Zip code\***

**Phone**

**Describe yourself\***

Since I have grown up with dogs , I can easily relate to them I can train then walk them feed them and also provide certain excercises to burn off those extra hyper activity of your pooch. I can do home sitting as well as pet sitting depending on the situation , you can easily get in touch in touch with me through the information I provide here on the website.I too have 2 dogs and can easily know how much it worries us to leave them alone and go out work. But with me your pooch doesn't have to worry.

**Experience\***

**Hourly rate\***

**Member of any organisation**

**Organisation name**

**Job type**

**Gender**

**Pet preference**

**Date posted\***

**Your photos with pets**

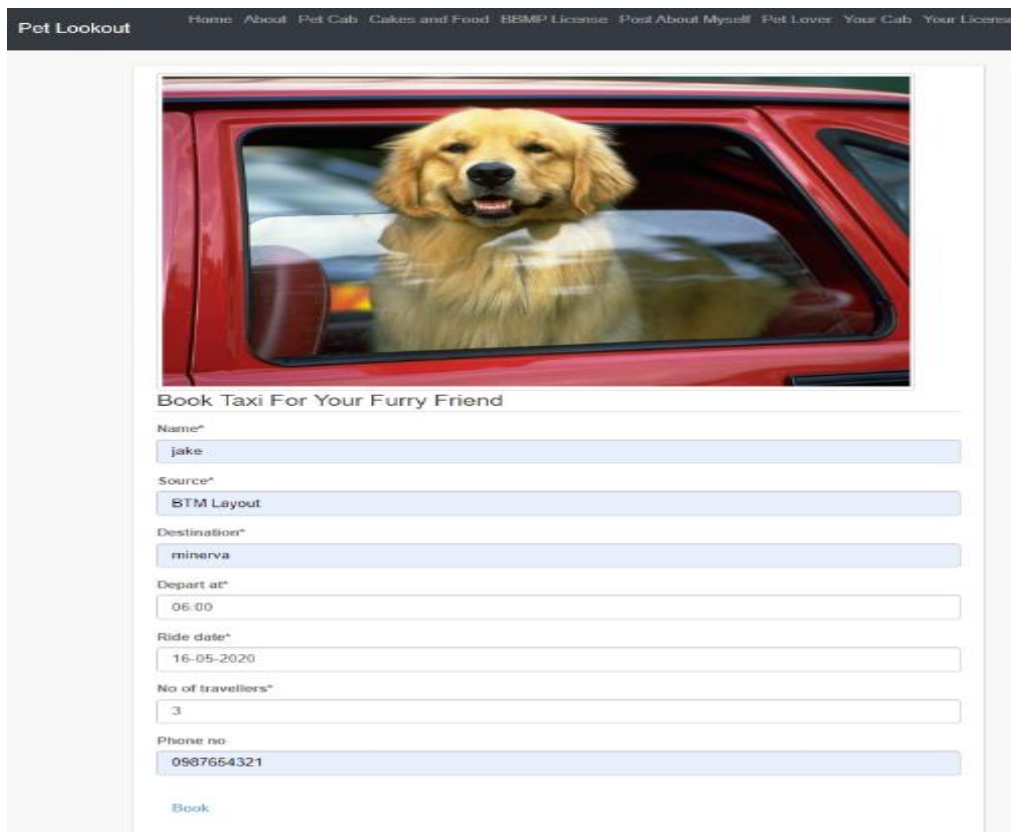
Currently: [profile\\_pics/jas\\_cX4iClu.jpg](#)

Change:

No file chosen

Figure 7.2.12 User updates profile to become pet lover

In figure 7.2.12 user who want to become a pet sitter, walker ,trainer can update his/her profile through filling in the details like .pet preference, job type, experience etc.



**Pet Lookout** Home About Pet Cab Cakes and Food BBMP License Post About Myself Pet Lover Your Cab Your License

**Book Taxi For Your Furry Friend**

Name\*  
jake

Source\*  
BTM Layout

Destination\*  
minerva

Depart at\*  
06:00

Ride date\*  
16-05-2020

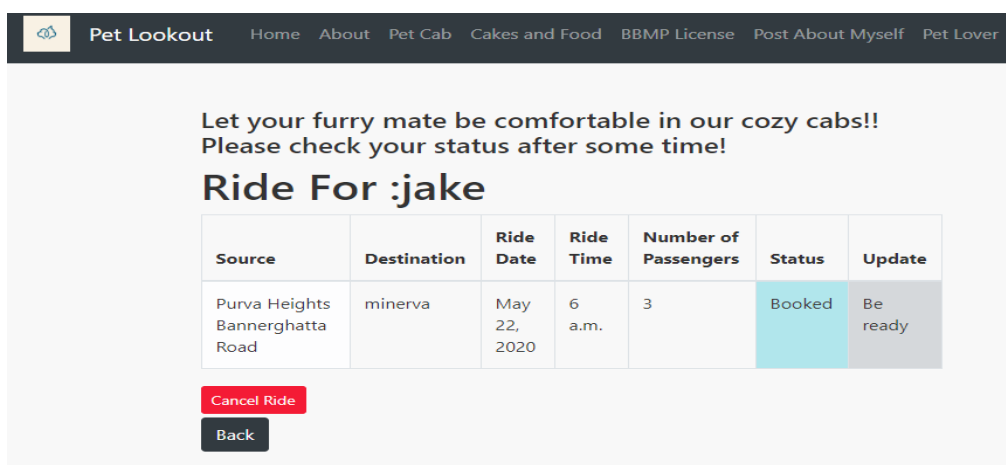
No of travellers\*  
3

Phone no.  
0987654321

Book

Figure 7.2.13 Pet cab booking screen

Figure 7.2.13 where the user can book a cab for their pet to travel with them by entering all the details like name, source, destination, number of travelers, ride date and ride time.



**Pet Lookout** Home About Pet Cab Cakes and Food BBMP License Post About Myself Pet Lover

Let your furry mate be comfortable in our cozy cabs!!  
Please check your status after some time!

**Ride For :jake**

Source	Destination	Ride Date	Ride Time	Number of Passengers	Status	Update
Purva Heights Bannerghatta Road	minerva	May 22, 2020	6 a.m.	3	Booked	Be ready

Cancel Ride

Back

Figure 7.2.14 Cab Booking Update

In the figure 7.2.14 user can see the status of their booking updated by the admin.

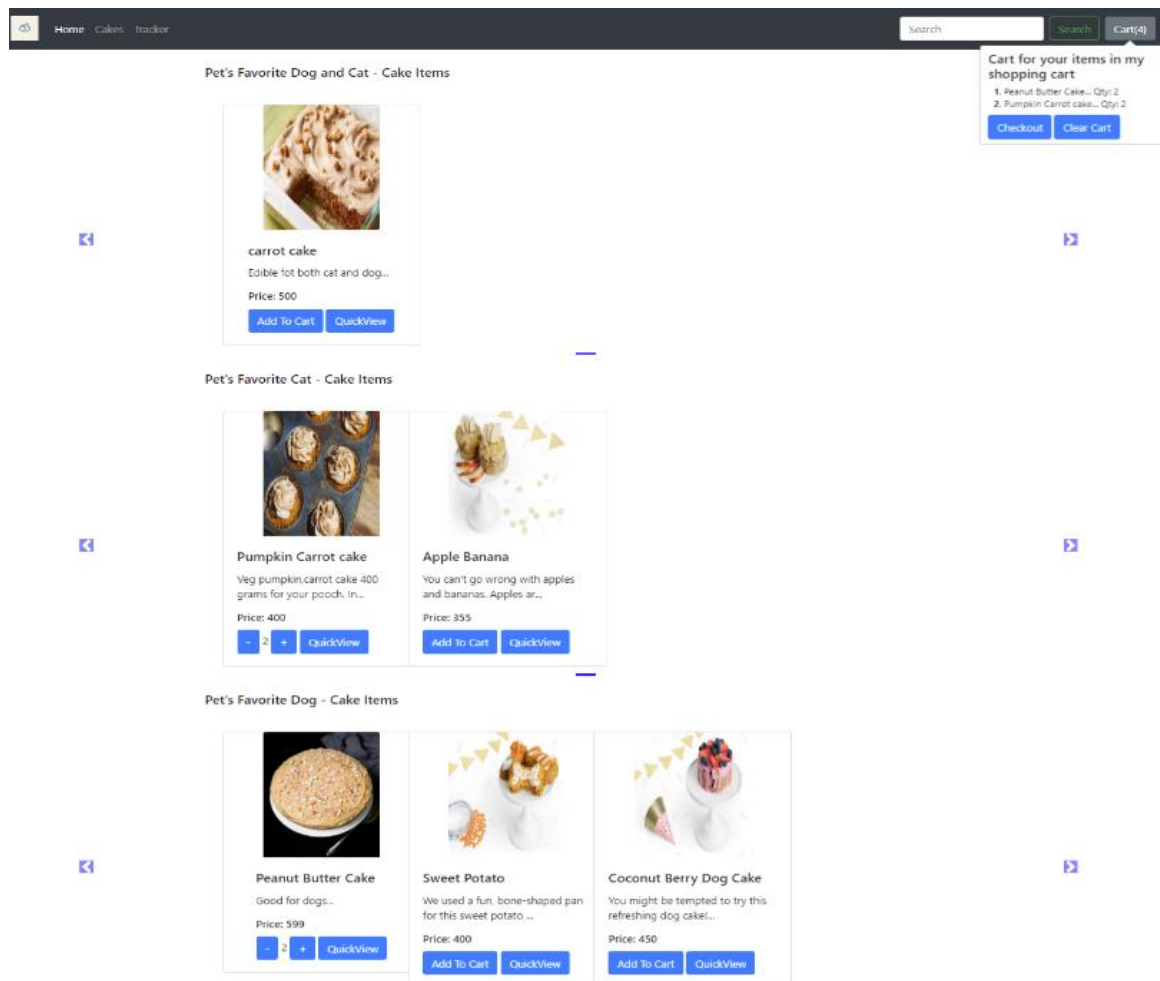


Figure 7.2.15 Cakes and Treats homepage

In the above figure all the varieties of cakes and treats for the pets can be seen categorized as cats and dogs food items , user can also add the items in their cart from here.



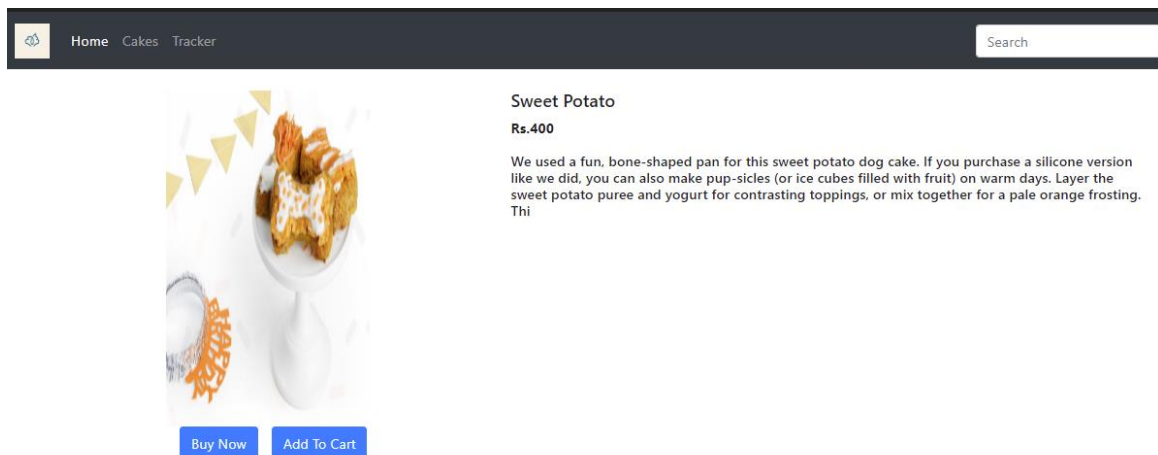


Figure 7.2.16 Items detail page

In the figure 7.2.16 the item can be viewed in detail composing of the description of food item.

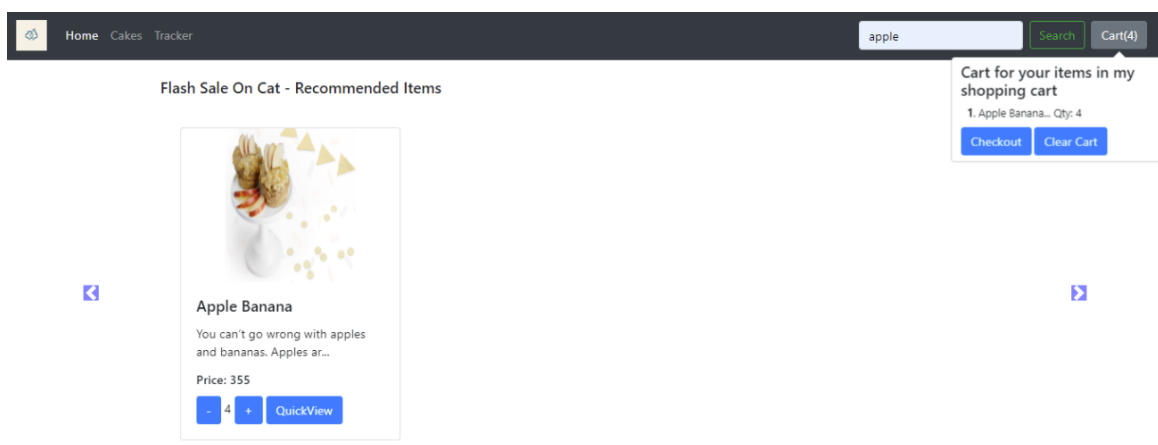


Figure 7.2.17 Item search and cart

In the figure 7.2.17 it can be clearly viewed the food item being searched and added to the cart.

**Step 1 - Pet Lookout Checkout - Review Your Cart Items**

Apple Banana	2
Pumpkin Carrot cake	3
carrot cake	1
Peanut Butter Cake	1

Your Cart Total Is ₹ 3009. Enter your details below & place your order. Thanks for using Pet Lookout ! Hope to serve you and your pup again.

**Step 2 - Enter Address & Other Details:**

Name

jake

Email

jake@gmail.com

Address

C 205 Kripa Nagar

Address line 2

bangalore

City

bengaluru

State

karnataka

Zip

560076

Phone Number

9880123451

Place Order

Figure 7.2.18 Checkout Page

Figure 7.2.18 depicts the checkout page which consist of all the order details and delivery details of the user like name , address, phone, email and the price of the items.

Cakes Tracker

Search

**Enter your Order Id and Email address to track your order**

Order Id

10

Email

jake@gmail.com

Track Order

**Your Order Status**


The order has been placed 2020-05-08


**Your Order Details**

Sweet Potato	1
Peanut Butter Cake	2

Figure 7.2.19 Track order page

In the above figure user can track the order status of theirs by entering the order id and email and the status is updated by the admin .


[Home](#)
[About](#)
[Pet Cab](#)
[Cakes and Food](#)
[BBMP License](#)
[Post About Myself](#)
[Pet Lover](#)
[Your Cab](#)
[Your License](#)
[Logout](#)



Send Your Application To Us and We will deliver it to you

First name\*

Last name\*

Pet name\*

Pet colour\*

Pet breed\*

Pet age\*

Address\*

Phone

Pet gender

Male

Pet type

Dog

Application preference

### Our Sidebar

Update your profile and get hired as a sitter,walker and trainer

Pet Cab at your Residency

Pet Cakes at your residency

BBMP License For your Pooch

Look at us for everything!

Pet gender

Male

Pet type

Dog

Application preference

New

Old license

Choose File No file chosen

Vaccination record\*

Choose File No file chosen

Address proof\*

Choose File No file chosen

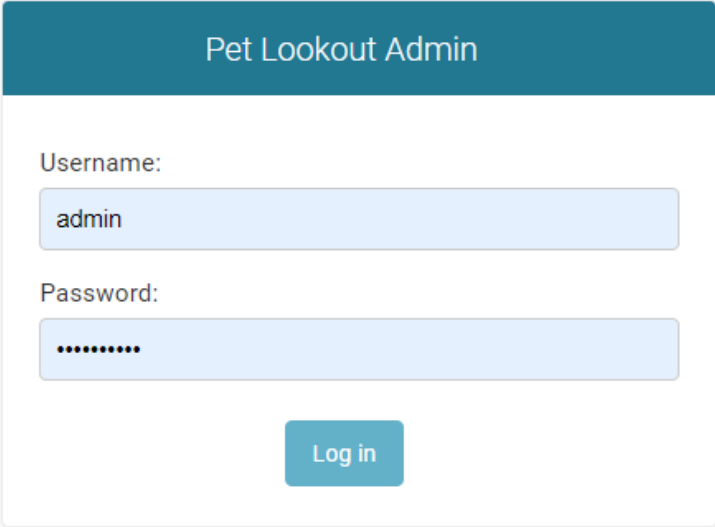
Send Application

### General FAQs

Q: How does this work?

Figure 7.2.20 BBMP license application page

In figure 7.2.20 user can apply for bbmp license by filling in all the mandatory details like pet name, pet color, pet breed ,license type, pet type etc.



The image shows a web form titled "Pet Lookout Admin" for logging in. It features two input fields: "Username:" with the text "admin" and "Password:" with masked characters (dots). Below the fields is a blue "Log in" button. The form is centered on a light gray background.

Figure 7.2.21 Admin login page

Figure 7.2.21 shows the admin login page where the admin who is the superuser of the system can login with the username and password.

## Site administration

## AUTHENTICATION AND AUTHORIZATION

Groups [+ Add](#) [Change](#)

Users [+ Add](#) [Change](#)

## BBMP

Licenses [+ Add](#) [Change](#)

## PETCAB

Petcabs [+ Add](#) [Change](#)

## POST

Pet posts [+ Add](#) [Change](#)

## SHOP

Order updates [+ Add](#) [Change](#)

Orders [+ Add](#) [Change](#)

Products [+ Add](#) [Change](#)

## USERS

Profiles [+ Add](#) [Change](#)

## Recent actions

## My actions

- ✖ The ord...  
Order update
- ✖ Orders
- ✓ Rashford  
Profile
- ✓ Coconut Berry Dog Cake  
Product
- ✓ Jake Petcab  
Petcab
- ✓ jake  
License
- ✖ Tester  
Orders
- ✖ anthony  
Orders
- ✖ jake  
Orders
- ✖ The ord...  
Order update

Figure 7.2.21 Admin's dashboard

Figure 7.2.21 shows the admin dashboard where admin can see all the users, profiles, booking made by the users, application for licenses, posts created by the users, orders and admin can also add and update products. Admin also has to update the status of bookings and orders made by the users.

**Pet Lookout Admin** WELCOME, SA

Home · Authentication and Authorization · Users

Select user to change

Q

Action:   0 of 8 selected

<input type="checkbox"/>	USERNAME	EMAIL ADDRESS	FIRST NAME	LAST NAME	STAFF STATUS
<input type="checkbox"/>	Jake	jake12@gmail.com			<span style="color: red;">✖</span>
<input type="checkbox"/>	Rashford	satish.tripathi1297@gmail.com			<span style="color: red;">✖</span>
<input type="checkbox"/>	admin	satish512tripathi@gmail.com	satish	tripathi	<span style="color: green;">✓</span>
<input type="checkbox"/>	anthony	anthony@gmail.com			<span style="color: red;">✖</span>
<input type="checkbox"/>	jack21	jach@gmail.com			<span style="color: red;">✖</span>
<input type="checkbox"/>	jake	jake@gmail.com			<span style="color: red;">✖</span>
<input type="checkbox"/>	proje	project@gmail.com			<span style="color: green;">✓</span>
<input type="checkbox"/>	satya123	satya@gmail.com			<span style="color: red;">✖</span>

8 users

Figure 7.2.22 List of users

In the above figure admin has the list of all the users registered on the website and also the admin has the access to delete the users and change the data.

The screenshot shows the 'Pet Lookout Admin' dashboard. At the top, there's a header with 'Pet Lookout Admin' and a welcome message for 'SATISH'. Below the header, a breadcrumb trail reads 'Home > Post > Pet posts'. The main content area is titled 'Select pet post to change' and includes an 'ADD PET POST +' button. There's an 'Action:' dropdown menu and a 'Go' button. A table lists four pet posts, each with a checkbox and a link to view the post. The posts are: 'PET POST', 'anthony Petpost', 'Rashford Petpost', 'Jake Petpost', and 'admin Petpost'. At the bottom, it says '4 pet posts'.

Checkbox	PET POST
<input type="checkbox"/>	<a href="#">anthony Petpost</a>
<input type="checkbox"/>	<a href="#">Rashford Petpost</a>
<input type="checkbox"/>	<a href="#">Jake Petpost</a>
<input type="checkbox"/>	<a href="#">admin Petpost</a>

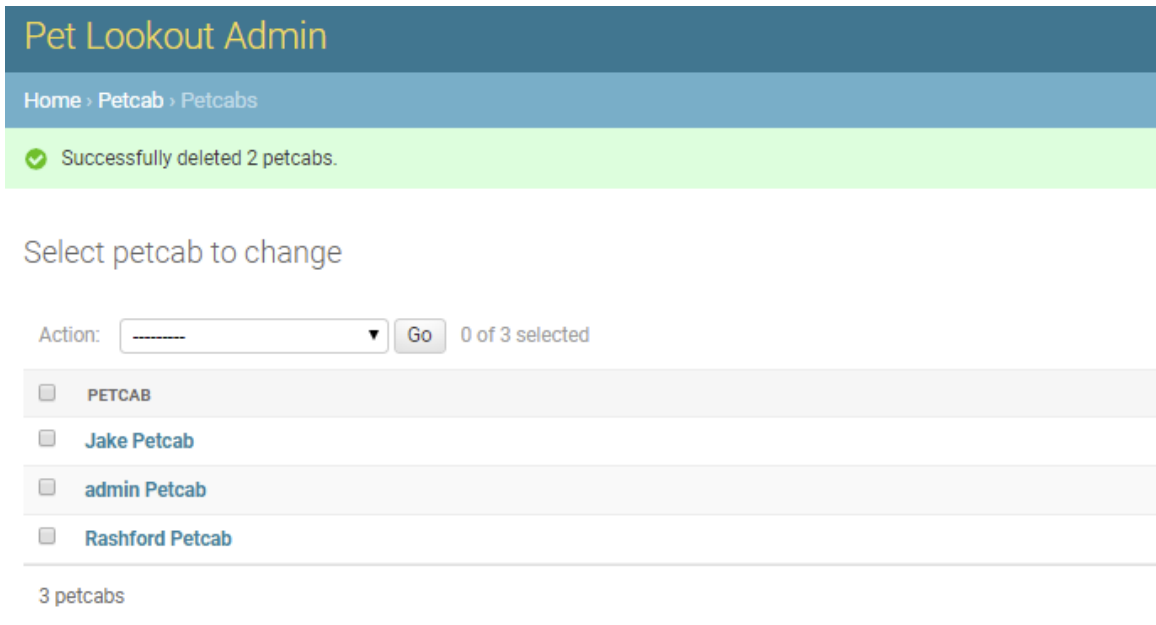
Figure 7.2.23 List of pet posts posted by pet lovers.

Figure 7.2.23 shows the list of pet post posted by the user , admin can view these posts and can also delete it if required.

The screenshot shows the 'Change license' form in the 'Pet Lookout Admin' dashboard. The header includes 'Pet Lookout Admin' and a welcome message for 'SATISH'. The breadcrumb trail is 'Home > Btamp > Licenses > jake'. The form is titled 'Change license' and has a 'HISTORY' button and a 'VIEW ON SITE' button. The form fields include: 'User' (Jake), 'First name' (jake), 'Last name' (Fernandes), 'Pet name' (jack), 'Pet colour' (Black), 'Pet breed' (german shepherd), 'Pet age' (2.0), 'Address' (c 205 Kripa nagar), 'Phone' (9880123451), 'Status' (Out for delivery), 'Message' (Information Received, Kindly Check Again Aft), 'Pet gender' (Male), 'Pet type' (Dog), and 'Application preference' (Renew). There are three sections for file uploads: 'Old license' (Currently: media/vaccine.pdf, Change: Choose File), 'Vaccination record' (Currently: media/address\_proof.pdf, Change: Choose File), and 'Address proof' (Currently: media/old.pdf, Change: Choose File). At the bottom, there are four buttons: 'Delete', 'Save and add another', 'Save and continue editing', and 'SAVE'.

Figure 7.2.24 License Application

The above figure depicts the license application applied by the user where admin can update the status by changing it to invalid, received or out for delivery.



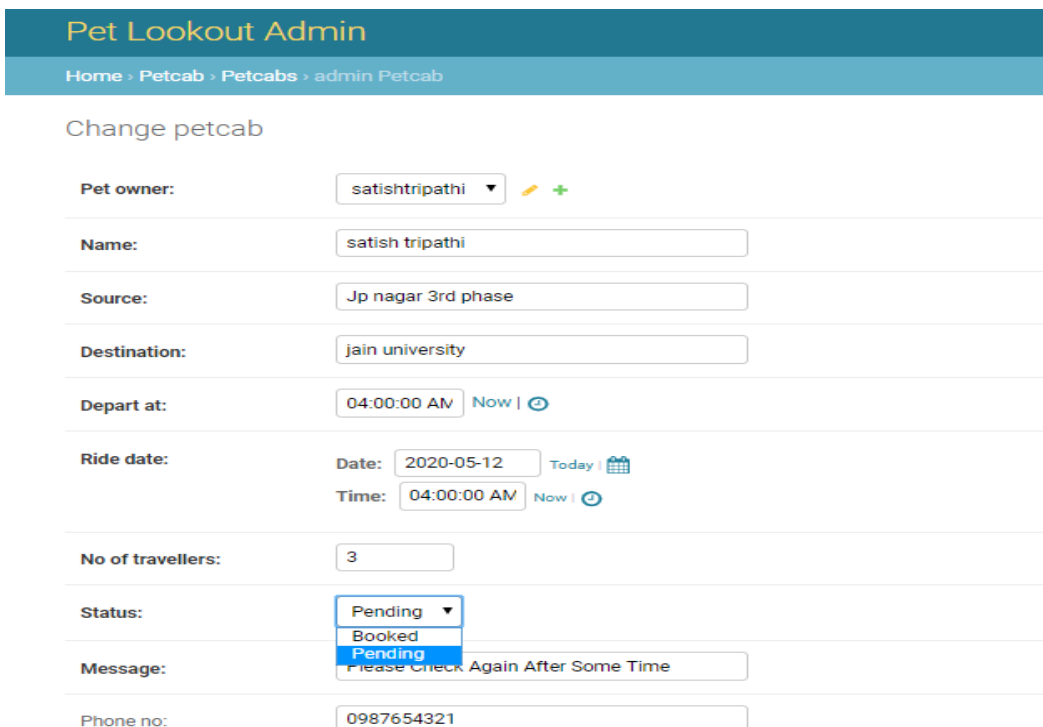
The screenshot shows the 'Pet Lookout Admin' interface. At the top, there's a header with the title 'Pet Lookout Admin' and a breadcrumb trail 'Home > Petcab > Petcabs'. Below the header, a green success message states 'Successfully deleted 2 petcabs.' The main section is titled 'Select petcab to change'. It features an 'Action:' dropdown menu, a 'Go' button, and a status '0 of 3 selected'. A table lists three petcabs: 'PETCAB', 'Jake Petcab', and 'admin Petcab', each with a checkbox. Below the table, it says '3 petcabs'.

PETCAB
<input type="checkbox"/> Jake Petcab
<input type="checkbox"/> admin Petcab
<input type="checkbox"/> Rashford Petcab

3 petcabs

Figure 7.2.25 Pet cab booking lists

It consist of the booking made by the users , the admin can update status of their bookings.



The screenshot shows the 'Pet Lookout Admin' interface for editing a booking. The breadcrumb trail is 'Home > Petcab > Petcabs > admin Petcab'. The section is titled 'Change petcab'. The form includes fields for 'Pet owner:' (satishtripathi), 'Name:' (satish tripathi), 'Source:' (Jp nagar 3rd phase), 'Destination:' (jain university), 'Depart at:' (04:00:00 AM), 'Ride date:' (Date: 2020-05-12, Time: 04:00:00 AM), 'No of travellers:' (3), 'Status:' (Pending), 'Message:' (Please Check Again After Some Time), and 'Phone no:' (0987654321). The 'Status:' dropdown menu is open, showing options: Pending, Booked, and Pending.

Pet owner:	satishtripathi
Name:	satish tripathi
Source:	Jp nagar 3rd phase
Destination:	jain university
Depart at:	04:00:00 AM
Ride date:	Date: 2020-05-12 Time: 04:00:00 AM
No of travellers:	3
Status:	Pending
Message:	Please Check Again After Some Time
Phone no:	0987654321

Figure 7.2.26 Pet cab user's booking

In the figure above admin confirms the booking of the user by changing the status to booked and updating the message.



Figure 7.2.27 Products list page

In above mentioned figure all the products which were added by the admin can be seen , admin can also update these products and add new products to this list.

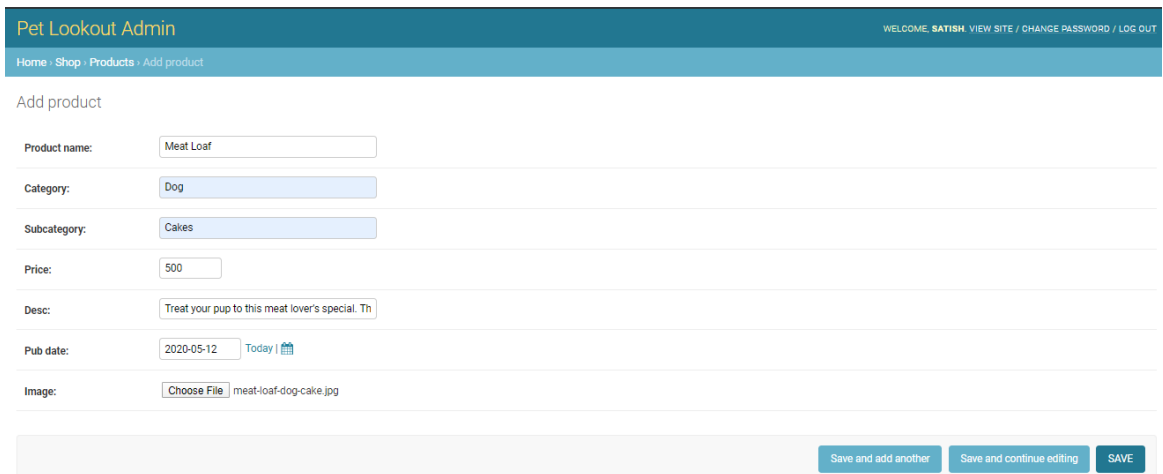


Figure 7.2.27 Add Product page

Figure 7.2.27 shows how the admin will add new products by entering the product name , description, price, categories and the image of the product.

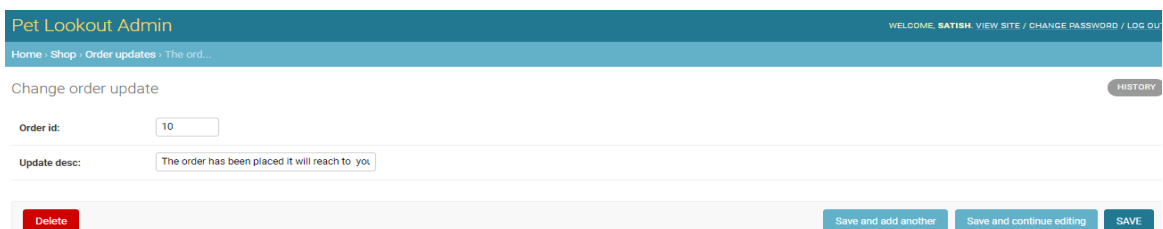


Figure 7.2.27 Order update page

In the above figure admin updates the status of the order by updating the update desc and entering the details of the order.



# **SOFTWARE TESTING**

## 8. SOFTWARE TESTING

Software testing is a process to evaluate the functionality of the software application. Test cases are prepared for software verification and software validation to determine if the product was built according to the requirements of the user.

### 8.1 TEST CASES

Test case no	Test Description	Required input	Expected Output	Actual Result	Status
1	Check User Login With valid data	valid username, password	user should be able to login and go to home page	Login is successful	Pass
2	Check user Login with Invalid data	Correct username, Incorrect password	Error message as Enter correct username and password.	Error message as Enter correct username and password.	Pass
3	Check User Login with Invalid data	Invalid username, Valid password	Error message as Enter correct username and password.	Error message as Enter correct username and password.	Pass
4	Check user login with invalid data	Invalid username, Invalid password	Error message as Enter correct username and password.	Error message as Enter correct username and password.	Pass
5	Check registration with different username and enter details	Different username .	Registered successfully.	Registered successful. You can now log in.	Pass

6	Check registration with same username	Same username	Error message as Invalid Credentials	Username already exists	Pass
7	Check Login with not activated user	Enter username and password	Error message as Enter correct username and password.	Error message as Enter correct username and password.	Pass
8	Admin Login	Invalid username, Invalid password	Error message not authorized.	Error message as, enter the correct username and password for a staff account.	Pass
9	Check Admin Login With valid data	valid superuser username, and password	Admin should be able to log in and go to admin dashboard.	Login is successful	Pass
10	Admin add Food Product.	Login as Admin. Enter Product Details. Click on Add	Product added successfully.	Added product should be displayed in product view.	Pass

11	Admin confirm Pet cab Booking	Select user from list of pet cab , Click on booking status option Select Booked. Click on save.	Pet cab booking status updated successfully.	Pet cab booking status updated successfully.	Pass
12	Pet Cab Booking	Enter number of passenger as 1. .	Error message	Error message as ensure this value is greater than or equal to 2.	Pass
13	Pet Cab Booking	Enter number of passengers as 6.	Error Message	Error message as ensure this value is greater than or equal to 4.	Pass
14	Pet Cab Booking	Enter number of passengers as 2.	Success message on booking.	Success message on booking.	Pass

## 8.2 SELENIUM TEST CASES

Test Case ID	1	Test Case Description	Check User Login with valid data		
Created By	Satish				
Tester's Name	Satish	Date Tested	May 8, 2020	Test Case (Pass/Fail/Not Executed)	Pass

S #	Prerequisites:
1	Access to Chrome Browser
2	
3	
4	

S #	Test Data
1	Username = jake
2	password = estophando
3	
4	

**Test Scenario** Verify on entering valid username and password user can login

Step #	Step Details	Expected Results	Actual Results	Pass / Fail / Not executed / Suspended
1	Navigate to http://127.0.0.1:8000/login/	Site should open	As Expected	Pass
2	Enter Username & Password	Credential can be entered	As Expected	Pass
3	Click Login	User is logged in	As Expected	Pass

Figure 8.2.1 User login with valid data test case

Test Case ID	2	Test Case Description	Check Admin Login with valid data		
Created By	Satish				
Tester's Name	Satish	Date Tested	May 8, 2020	Test Case (Pass/Fail/Not Executed)	Pass

S #	Prerequisites:
1	Access to Chrome Browser
2	
3	
4	

S #	Test Data
1	Username = admin
2	password = estophando
3	
4	

**Test Scenario** Verify on entering valid username and password the admin can login.

Step #	Step Details	Expected Results	Actual Results	Pass / Fail / Not executed / Suspended
1	Navigate to http://127.0.0.1:8000/admin/login/?next=/admin/	Site should open	As Expected	Pass
2	Enter username and password	Credential can be entered	As Expected	Pass
3	Click Login	Admin is logged in	As Expected	Pass

Figure 8.2.2 Admin login with valid data test case

Test Case ID	3	Test Case Description	Test the number of travellers in pet cab booking			
Created By	Satish					
Tester's Name		Satish	Date Tested	May 8, 2020	Test Case (Pass/Fail/Not Executed)	Pass
S #	Prerequisites:					
1	Access to Chrome Browser					
2						
3						
4						

Figure 8.2.3 Number of travelers test case

Test Case ID	4	Test Case Description	User registration with an existing username		
Created By	Satish				
Tester's Name	Satish	Date Tested	May 8, 2020	Test Case (Pass/Fail/Not Executed)	Pass
S #	Prerequisites:		S #	Test Data Requirement	
1	Access to Chrome Browser		1	username = jake	
2			2	email = jake@gmail.com	
3			3	password = estophando	
4			4	password confirmation = estophando	
<u>Test Conditions</u>					
Step #	Step Details	Expected Results	Actual Results		Pass / Fail / Not executed / Suspended
1	<a href="http://127.0.0.1:8000/register/">http://127.0.0.1:8000/register/</a>	site should open	As Expected		Pass
2	Enter username,email, password and password confirmation	Credentials can be entered.	As Expected		Pass
3	Click Sign Up	Error message as username already exists	As Expected		Pass

Figure 8.2.4 User registration with an existing username.

Test Case ID	5	Test Case Description	Track Ordered Order		
Created By	Satish				

Tester's Name	Satish	Date Tested	8 May , 2020	Test Case (Pass/Fail/Not Executed)	Pass
---------------	--------	-------------	--------------	------------------------------------	------

S #	Prerequisites:
1	Access to Chrome Browser

S #	Test Data
1	Order id = 10
2	Email = jake@gmail.com

Step #	Step Details	Expected Results	Actual Results	Pass / Fail / Not executed / Suspended
1	<a href="http://127.0.0.1:8000/track_order/">http://127.0.0.1:8000/track_order/</a>	Site should open	As Expected	Pass
2	Enter order id and Email	Data can be entered	As Expected	Pass
3	Click Track order	Order details and order status	As Expected	Pass

Figure 8.2.5 Track ordered order status test case

Test Case ID	6	Test Case Description	Search unavailable products		
Created By	Satish				

Tester's Name	Satish	Date Tested	8 May , 2020	Test Case (Pass/Fail/Not Executed)	Pass
---------------	--------	-------------	--------------	------------------------------------	------

S #	Prerequisites:
1	Access to Chrome Browser

S #	Test Data
1	product name = mango

Step #	Step Details	Expected Results	Actual Results	Pass / Fail / Not executed / Suspended
1	<a href="http://127.0.0.1:8000/">http://127.0.0.1:8000/</a>	Site should open.	As Expected	Pass
2	Enter product	product name can be entered	As Expected	Pass
3	Click search	Error message as Please make sure to enter relevant product name	As Expected	Pass

Figure 8.2.6 Search unavailable product test case

# **CONCLUSION**



## **9. CONCLUSION**

With the increase in number of pet owners every year , the developed web application will provide great assistance to all the pet owners with all the services provided and will give a head start to people who want to take responsibility of taking care of other's pet and it satisfies the much needed requirements.

# **FUTURE ENHANCEMENTS**

## **10. FUTURE ENHANCEMENTS**

With the advancement in technology every now and then, one can understand that this project opens with some of the future enhancements that are listed as below.

- Booking of pet lovers from the website and feedback system.
- Verification badges for certified for pet lovers.
- Variation in pet cab services.
- Booking of pet lovers from the website.
- Booking of pet professionals from the website.
- Adding varieties to the pet cab booking services .
- Auto Disable for users who are not using this application for longer time.

# APPENDIX

## **Appendix A: BIBLIOGRAPHY**

### **BOOK REFERENCES**

- [1] Django Web Framework for Python Django Book by Adrian Holovaty and Jacob Kaplan-Moss.
- [2] S.L. Pfleeger, Software Engineering – Theory and Practice, Shari Lawrence Pfleeger, 2nd Edition, Pearson Education.

### **WEB REFERENCES**

- [1] <https://www.stackoverflow.com>
- [2] <https://docs.djangoproject.com/>
- [3] <https://www.w3schools.com/php/>
- [4] <https://djangoreadthedocs.io/>
- [5] <https://www.businesstoday.in/magazine/the-hub/the-pet-bet/story/376801.html>
- [6] <https://www.petboro.in/>
- [7] <https://www.anvisinc.com/>
- [8] <https://barkntreat.com/>
- [9] <https://waggle.in/>

## **Appendix B – USER MANUAL**

### **[1] Pet Lookout-Login Page**

Step 1: Login page consists of two option login and Sign up now.

Step 2: For registration page, click on Sign up now.

Step 3: Enter email, username, enter password and confirm password.

Step 4: Click register button.

Step 5 : For login , enter username and password.

Step 6: Click Login.

### **[2] Pet Lookout- Manage Users (Admin)**

Step 1: Login as Admin.

Step 2: Click on users and view the registered users.

Step 3: Delete users if needed.

### **[3] Pet Lookout-Users Profile (Admin)**

Step 1: Login as Admin.

Step 2: Click on users profiles tab from top navigation bar and click on the users option.

Step 3: View the created profile and perform action button of deleting , updating the profile if needed.

### **[4] Pet Lookout-Pet Posts of Pet Lovers(Admin)**

Step 1: Login as Admin.

Step 2: Click on pet post tab from top navigation bar and click on pet lovers post.

Step 3: Click on action button of respective user and click on reset password.

Step 4: Enter new password and confirm password.

### **[5] Pet Lookout- Pet Cab Booking (Admin)**

Step 1: Login as Admin.

Step 2: Click on Petcabs tab from top navigation bar named PETCAB .

Step 3: Click on the user Petcab to update the status of the booking and change the update message of respective user.

Step 4: Click on Save.

**[6] Pet Lookout- License Booking (Admin)**

Step 1: Login as Admin.

Step 2: Click on Licenses tab from top navigation bar named BBMP .

Step 3: Click on the user to update the status of the application and change the update message of respective user.

Step 4: Click on Save.

**[7] Pet Lookout- Cakes and Food (Admin)**

Step 1: Login as Admin.

Step 2: Click on Products tab from top navigation bar named SHOP .

Step 3: Click on add product to add a new product or click on the any of the added product to update it.

Step 4: Click on Save.

**[8] Pet Lookout- Cakes and Food (Admin)**

Step 1: Login as Admin.

Step 2: Click on Products tab from top navigation bar named SHOP .

Step 3: Click on add product to add a new product or click on the any of the added product to update it.

Step 4: Click on order updates and update the order status.

Step 4: Click on Save.

**[9] Pet Lookout- Pet Lover Profile(User)**

Step 1: Login as User.

Step 2: Click on Pet lover tab from top navigation bar ..

Step 3: Enter all the details and upload your profile picture with a pet .

Step 4 : Click on Update.

**[10] Pet Lookout- Pet Lover Post(User)**

Step 1: Login as User.

Step 2: Click on my post tab from top navigation bar.

Step 3: Enter working style, interests and a picture with a pet.

Step 4: Click on post.

Step 5: To update posts go to your posts and click on the interests and update it again if needed.

#### **[11] Pet Lookout- View posts and pet lover's profiles(User)**

Step 1: Login as User.

Step 2: For post , type anything related to pets in the search box and click search.

Step 3: For pet lover's profiles click on Pet Lover's Profile and enter the type of Pet Lover as a trainer, walker and pet sitter.

Step 4 : For detail profile view click on the pet lover's name on their posts.

#### **[12] Pet Lookout- Pet Cab(User)**

Step 1: Login as User.

Step 2: For cab booking click on pet cab from top navigation bar.

Step 3: Enter details such as Name, Source, Destination , Ride day, Ride time and number of travelers.

Step 4: Click Book.

Step 5: To view booking status click on Your booking from top navigation bar and click on the bookings.

Step 6: To delete a particular booking , click on delete and then on confirmation select yes.

#### **[13] Pet Lookout- BBMP License(User)**

Step 1: Login as User.

Step 2: For License application click on BBMP License from top navigation bar.

Step 3: Enter details such as Name, Pet name, Pet type ,Pet age, Address and upload all your documents.

Step 4: Click on Send Application.

Step 5: To view license status click on Your license from top navigation bar and click on the licenses applied.

#### **[14] Pet Lookout- Cakes and Food(User)**

Step 1: Login as User.

Step 2: Click on cakes and food from top navigation bar.

Step 3: View the product and add the product to the cart if needed.

Step 4: To search for a product , type the product name on the search bar and click search.



Step 5: To get a quick view of the product , click on quick view.

Step 6: Click on checkout to place order.

Step 7: Enter details such as Name, address, phone no , email, zip .

Step 8: Click on Place order.

Step 9: To track the order, click on tracker on top of navigation bar.

Step 10: Enter order id and email.

Step 11: Click on track order.