

**PROFORMA FOR THE APPROVAL PROJECT PROPOSAL**

*(Note: All entries of the proforma of approval should be filled up with appropriate and complete information. Incomplete proforma of approval in any respect will be summarily rejected.)*

PNR No. : …………………… Roll No. : ……………..

1. Name of the Student

…………………………………………………………………………………...

1. Title of the Project

…………………………………………………………………………………...

1. Name of the Guide

…………………………………………………………………………………...

1. Teaching experience of the Guide ………………………………………………

1. Is this your first submission? Yes …. No ….

Signature of the Student Signature of the Guide

Date: …………………. Date: …………………. Signature of the Coordinator

Date: …………………...

# ANIMAL ADOPTION

**A Project Report**

Submitted in partial fulfilment of the

Requirements for the award of the Degree of

**BACHELOR OF SCIENCE (INFORMATION TECHNOLOGY)**

**By**

Yash Gadodia

1920518

**Under the esteemed guidance of**

**Mrs. Nidhi Singh**

**Professor.**

**DEPARTMENT OF INFORMATION TECHNOLOGY**

**LALA LAJPAT RAI COLLEGE OF COMMERCE AND ECONOMICS.**

***(Affiliated to University of Mumbai)***

**MUMBAI-400 034.**

**MAHARASHTRA**

**YEAR 2021-22**

**LALA LAJPAT RAI COLLEGE OF COMMERCE AND ECONOMICS.**

***(Affiliated to University of Mumbai)***

**MUMBAI-400 034. MAHARASHTRA.**

**DEPARTMENT OF INFORMATION TECHNOLOGY**



**CERTIFICATE**

This is to certify that the project entitled, “**Animal Adoption**”, is bonafied work of **YASH GADODIA** bearing Seat. No. :- \_\_\_\_\_\_ and Roll No: **1920518** submitted in partial fulfilment of the requirements for the award of degree of BACHELOR OF SCIENCE in INFORMATION TECHNOLOGY from University of Mumbai.

**Subject Teacher Coordinator**

**Internal Guide External Examiner**

**Date:**

**College Seal**

## ABSTRACT

As a partial fulfilment of Final Year Project, the main aim of this project is to develop an platform for animal adoption. Right now, millions of cats and dog are waiting to be adopted. This is a project regarding the treatment of stray animals by people and how we can improve the condition. Our website can help to improve this current scenario. It is an online platform where people can adopt pets.

This platform is based on software and database to solve the problem occurs for public who want to adopt animal. A Dell laptop has been used as the hardware devices to create and modified the software. The software being used is Visual Studio as the developing platform.

## ACKNOWLEDGEMENT

I have a great pleasure in representing this project report entitled " **ANIMAL ADOPTION** " and I grab this opportunity to convey my immense regards towards all the distinguished people who have their valuable contribution in the hour of need.

I take this opportunity to thank Prof. Mrs. Nidhi Singh, Coordinator of the Department and all the professors of the Department of Information Technology of Lala Lajpat Rai College of Commerce & Economics, for giving me an opportunity to complete this project and the most needed guidance throughout the duration of the Programme.

I am extremely grateful to my project guide Prof. Mrs. Jayshri Parab her valuable guidance and necessary support during each phase of the project. She was the source of continuous encouragement as each milestone was crossed.

A special thanks to the University Of Mumbai for having prescribed this project work to me as a part of the academic requirement in the Final year of Bachelor of Science in Information Technology.

Sincere thank from,

YASH GADODIA

## DECLARATION

I here by declare that the project entitled, “**Animal Adoption**” done at **Mumbai,** has not been in

any case duplicated to submit to any other university for the award of any degree. To the best of

my knowledge other than me, no one has submitted to any other university.

The project is done in partial fulfillment of the requirements for the award of degree of

### BACHELOR OF SCIENCE (INFORMATION TECHNOLOGY) to be submitted

as final semester project as part of our curriculum.

**Name and Signature of the Student**

**TABLE OF CONTENTS.**

**Chapter 1: Introduction**

1.1 Background

1.2 Objectives

1.3 Purpose, Scope and Applicability

1.3.1 Purpose

1.3.2 Scope

1.3.3 Applicability

1.4 Achievements

**Chapter 2: Survey of Technologies**

**Chapter 3: Requirements and Analysis**

3.1 Problem Definition

3.2 Requirements Specification

3.3 Planning and Scheduling

3.4 Software and Hardware Requirements

3.5 Preliminary Product Description

3.6 Conceptual Models

**Chapter 4: System Design**

4.1 Basic Modules

4.2 Data Design

4.2.1 Data Integrity and Constrains

4.3 User Interface Design

4.4 Test Cases Design

4.5 Security Issues

**Chapter 5: Implementation and Testing**

5.1 Algorithm of important modules

**Chapter 6: Conclusion**

**Chapter 7: Bibliography**

7.1 References

## Chapter 1 Introduction

**1.1 BACKGROUND**

Animal adoption is the process whereby a person brings an animal, mostly cats and dogs, to their own care. These animals that are either lost or abandoned are taken by the animal control to be kept in animal shelters. Pets that are kept in the shelters for a long time are euthanized to reduce the number of overpopulation of unwanted animals.

The inspiration of this project is a web-based adoption site for people to be more aware of stray animals waiting to be adopted around India. The website is accessible for users and administrators where users can register themselves into the system to view the list of animals waiting to be adopted along with each animal’s records such as their (estimated) age, gender, and their description.

### 1.2 OBJECTIVES

The goal of this project is to develop a website for users to select animals to adopt conveniently. The main objectives of this website development can be defined as follows:

* To develop a system that provides functions to support users to view the animals in the shelters conveniently.
* To be able to view the list of animals in the shelters and their basic information order to make decisions to adopt.
* To maintain records of animals in the shelters and the users information in a centralized database system.
* To develop a system the user’s answers to the provided questionnaire
* The ultimate goal is to increase the rate of animals adopted.

.

### 1.3 PURPOSE, SCOPE AND APPLICABILITY

#### 1.3.1 Purpose

Following are some purposes why system is being designed:

* People can choose from a great selection of animals.
* To educate the community on Animal Welfare.
* Provide a better customer service platform.
* To find homes for rescued animals.
* Fast, easy and comfortable to use.

#### Scope

* The user has to send a request through the system which will then notify the admin of a

new adoption request.

* All animal’s records are input and updated by the administrators.
* It is mandatory for the user to fill in the questionnaire form.
* The user can view the records of animals in the lists.
* The administrator will send an approval E-mail to user.

• **Limitations**

Every user must go through a set of questionnaires provided by the system after signing up which will then be reviewed by the administrator when the user sends a request to adopt an animal. One of the limitations of this system is that the questionnaires can only be approved by the administrator and not approved automatically in the system, so the administrator has to manually go through each questionnaire done by the user to approve whether the user has the qualifications to adopt an animal or not.

#### 1.3.3 Applicability

* The customer satisfaction would be definitely higher when compared to the old manual system.
* Users do not have to contact the administrators to know the results of their adoption.
* Easy to handle even for a person with basic computer knowledge.
* User has mandatory to fill the questionnaire form.
* The admin can update the details of the pets.

### 1.4 ACHIEVEMENTS

* If some changes occurs than how it will be implemented without interfering either modules.
* How the changes are implemented in each module according to the future requirements.
* I learned how to design diagram in the Star UML by using different components.
* While making the project I understood how the modules are to be implemented.
* Implementation of interfaces methods used in the software
* Developing the software phase by phase.
* Managing the work and time.

## Chapter 2 Survey of Technologies

**MICROSOFT VISUAL STUDIO**

Microsoft Visual Studio is an integrated development environment (IDE) from Microsoft. It is used to develop computer programs, as well as websites, web apps, web services and mobile apps.

**Advantages of using Microsoft VS code.**

• Accurate Coding. With Visual Studio IDE, users are provided live coding assistance regardless of the programming language they are utilizing.

• Quick Debugging.

• Rigorous Testing.

• Team Collaboration.

• Customization Options.

### Front End:

Scripting language came about largely because of the development of internet as a communication tool. Types of scripting technology are ASP .NET, JSP, JavaScript, PHP and Python. In this project am using HTML language.

#### HTML (Hypertext Markup Language)

HTML (stands for Hypertext Markup Language) is a computer language that makes up most web pages and online applications. A hypertext is a text that is used to reference other pieces of text, while a markup language is a series of markings that tells web servers the style and structure of a document.

##### Features of HTML

* Images, videos, and audio can be added to a web page.
* It is easy to learn and easy to use.
* Hypertext can be added to the text.
* It is platform-independent.

#### CSS (Cascading Styling Sheet)

CSS stands for Cascading Styling Sheet. It gives an additional style to the HTML document. A cascading style sheet is a language that is designed to define the document formatting and look written in a markup language. Generally, CSS is applied with HTML documents to change various styles of user interface and web pages.

##### Features of CSS

* **Opportunity in Web designing** **:** If anyone wants to begin a career in web designing professionally it is essential to have knowledge of CSS and HTML .
* **Website Design :** With the use of CSS, we can control various styles, such as the text color, the font style, the spacing among paragraphs, column size and layout, background color and images, design of the layout, display variations for distinct screens and device sizes, and many other effects as well.
* **Web Control :** CSS has controlling power on the documents of HTML, so it is easy to learn. It is integrated with the HTML and the XHTML markup languages.

### Back End:

#### Python

Python is a high-level, general-purpose programming language. Its design philosophy emphasizes code readability with the use of significant indentation. Its language constructs and object-oriented approach aim to help programmers write clear, logical code for small- and large-scale projects.

##### Features of Python

* It is easy to learn and easy to use.
* Expressive Language.
* Free and Open Source.
* Python supports Object-Oriented Language.

### MySQL

MySQL is an open-source relational database management system (RDBMS). This is a freeware

hence best suited for the project environment to form the back end database.

MySQL can be used for a variety of applications, but is most commonly found on web servers. A website that uses MySQL may include web pages that access information from a database. These pages are often referred to as dynamic meaning the content of each page is generated from a database as the page loads. Website that use dynamic web pages are often referred to as database driver websites.

#### Features of SQL

##### • Relational Database System

Like almost all other database system on the market. MySQL is a relational database system.

##### • Client/Server Architecture

MySQL is a client/server system. There is a database server and arbitrarily many clients (Application programs). Which communicate with the server that is they query data save changes etc. The clients can run on another computer (communication via a local network or the internet). Almost all of the familiar large database systems (Oracle, Microsoft SQL Server, etc.) are client/server system. These are in contrast to the file server systems which include Microsoft Access, dBase and FoxPro. The decisive drawback of file server systems is that when run over a network they become extremely inefficient as the number of users grows.

##### • SQL Compatibility

MySQL supports as its database language as its name suggests SQL (Structure Query Language). SQL is a standardized language for querying and updating data and for the administration of a database. There are several SQL dialects (about as many as there are database systems). MySQL addresses to the current SQL standards (at the moment SQL: 2003), although with significant restrictions and a large number of extensions.

### TOOL

#### Star UML

Star UML is a UML tool by MK Lab. This software is very useful to draw the diagram like class diagram, activity diagram, and use case diagram in the project.

##### Features of Star UML

* Supports most of the diagrams specified in UML 2.0.
* Very rich features set and formatting options.
* Ability to generate source code from the UML diagram.
* Supported languages: Java, C, C#.
* Fast load time / execution time compared with other UML Tools.
* Supports exporting diagrams into JPG format.

### Technical Feasibility

* We are building a system that matches the organization's requirements.
* We have chosen Java and MySQL as this project is vast. Java is specialized for big-size applications. Also, a database server like MySQL can handle big database tables easily without causing any interruptions. And also, we are much familiar with these technologies.
* The next area we have concentrated on is operating system architecture. Currently, we are using Windows 10. To make the software architecture user-friendly and run-on multiple OS, we have chosen Java due to its cross-platform feature that makes the software function anywhere with less configuration.
* We are making the system's UI more user-friendly and interactive. With an automated system that will generate reports as per the specific need of the organization

### Operational Feasibility

The purpose of the project is to develop an online web system named **‘Animal Adoption’**. The aim of the project is to is to develop a website for users to select animals to adopt conveniently. The proposed of the system will make it easier for user to choose the pets according to their requirements.

The functionality of the operators are as follows:

1. Admin

2. User

**Admin**

* Admin can be able to create, update, delete and query the animal’s records in order to manage information.
* Admin can be able to create, update, delete, and query adoption records to manage information about adoptions made.
* Admin can be able to approve whether the user who wants to adopt an animal meets the qualifications to raise a pet .
* Admin can be able to confirm whether the user has come to adopt the animal or canceled their adoption request.

**User**

* To be able to register as a part of the system.
* To be filled a mandatory form.
* To be able to view the list of animals in the shelters and their basic information order to make decisions to adopt.
* To send a request to adopt an animal through the system.
* To be notified of the adoption results by e-mail.

### Economic Feasibility

* Being a web developer will have an associated hosting cost. As the system consists of multimedia data transfer, the bandwidth required for the operation of this application is not extremely costly.
* The system follows the freeware software standards. Hence, no cost will be charged from the potential clients. Bugfix and maintenance tasks will have an associated cost.
* Besides the associated cost, there will be many benefits for the customer. Especially the extra effort that is associated with managing data in a file system will be reduced, creating detailed statical reports will be eliminated since report generation will be completely autom

### COCOMO Model

The Constructive Cost Model (COCOMO) is a productive software cost estimation model developed by Barry W. Boehm. The model parameter are driver from fitting a regression formula using data from historical projects (16 projects for COCOMO 81 and 163 projects for COCOMO II).

Basic COCOMO computer software development effort (and cost) as a function of program size.

Program size is expressed in estimated thousands of source lines of code (SLOC, KLOC).

COCOMO applies to three classes of software projects:

* Relatively small, simple software projects.
* Small teams with good application experience work to a set of less than rigid requirements.
* Similar to the previously developed projects.
* Relatively small and requires little innovations.
* Developed in familiar, stable environment.

While considering economic feasibility it is checked in points like performance, information and outputs from the system. MySQL is a free software available and also compatible foe almost all versions of windows operating system. Thus, the cost incurred to develop the system is less and does not affect the cost to the project development process. This justifies economic feasibility of the system.

**Chapter 3 Requirements and Analysis**

### 3.1 Problem Definition

Defining a problem is one of the important activities of the project. The objective is to define precisely the business problem to be solved and thereby determined the scope of the new system.

Some limitation / problems occurred in the current existing system because of no automation been used are as follows:

* Time consuming: As records are manually maintained, more time is consumed.
* Paper work: Lots of paper work is involved as the records are maintained in files and registers.
* Storage requirements: As files and registers are being used, the storage space requirement increases.
* Less reliable: Use of paper for storing sensitive and valuable information is not at all reliable and secure.
* Accuracy: As the system is manual system, there are many chances of human errors to occur, such as miscalculation or mistakes or errors in maintaining records, etc.
* Difficulty in keeping new records: It becomes difficult to maintain new records entries of members, their transaction details related to their accounts and so on.

### 3.2 Requirements Specification

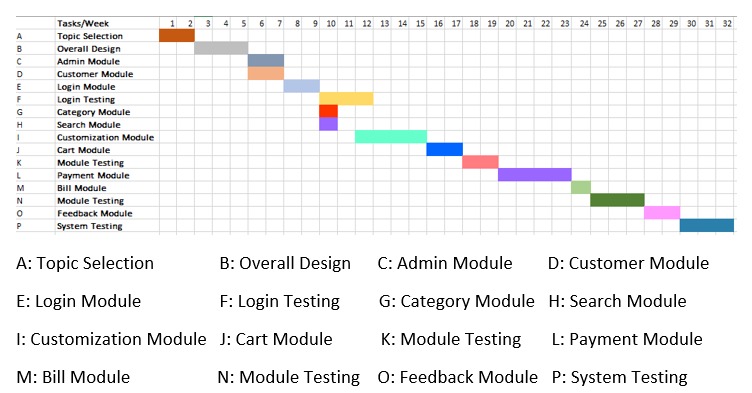
* In this system there is a registration page, a login page for the User.
* User has to fill the questionnaire about their background, interests, life style and history of raising a pet.
* The admin can update and delete the pets information when required.
* The admin will send an approval E-mail to the user whether they meet the qualifications of being a decent pet owner or not
* The system also keeps the record of pet adoptions.

### 3.3 Planning and Scheduling

#### 3.3.1 GANTT chart

A Gantt chart, commonly used in project management, is one of the most popular and useful ways of showing activities display against time. On the left of the chart is a list of activities and along the top is a suitable time scale. Each activity is represented by a bar; the position and length of the bar reflects the start date, duration and end date of the activity. This allows you to see at a glance:

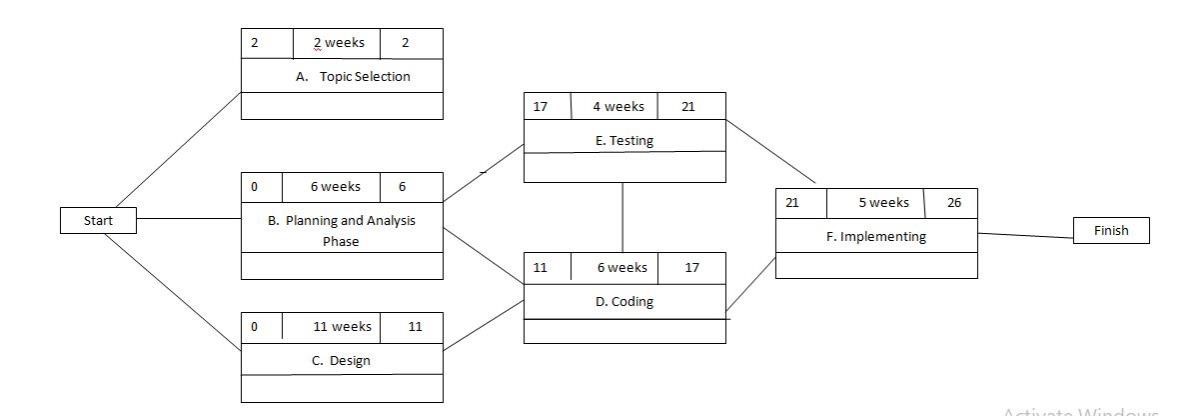
* What the various activities are
* When each activity begins and ends
* How long each activity is scheduled to last
* Where activities overlap with other activities, and by how much
* The start and end date of the whole project



#### 3.3.2 PERT chart

A Pert chart is a project management tool used to schedule, organize, and coordinate tasks within a project. PERT stands for program Evaluation review Technique, a methodology developed by U.S. Navy in 19540s to manage the Polaris submarine missile program. A similar methodology, the Critical Path Method (CPM) was developed for project management in the private sector at about the same time.

* Numbered rectangles are notes and represent events or milestones.
* Directional arrows directions indicates possible concurrent tasks.
* Dotted lines indicates dependent tasks that do not require resources.



### 3.4 Software and Hardware Requirements

**Hardware Requirements:**

* Minimum 5 GB HDD space.
* Pentium based processor.
* Minimum 128 MB of RAM.

**Software Requirements:**

* Microsoft windows.
* Visual Studio.
* MySQL.

### 3.5 Preliminary Product Description

The system Animal Adoption is the overall management of the pets.

* Users do not have to contact the administrators to know the results of their adoption. They can view their result of their adoption request through e-mail.
* The reports and information are kept in electronic form and can be easily maintained by the administrators and they can access the records whenever they want to.
* All reports are kept in an electronic files so that they may last longer and have less chance of being lost or damaged.
* Administrators can easily manage records such as the animal records, request records, and history records in the system.

### 3.6 Conceptual Models

#### ER Diagram

An entity relationship diagram (ERD) shows the relationships of entity sets stored in a database. An entity in this context is a component of data. In other words, ER diagrams illustrate the logical structure of databases.

**Entity:**

An entity is an object in the real world that is distinguishable from other objects. Examples: The address of the manager of the institution, a Person with unique name etc. It is often useful to identify a collection of similar entities. Such a collection is called as “Entity set”. Note that entity set need not be disjoint.

**Attributes:**

An entity is described using a set of attributes. All entities in a given entity set have the same attributes; this essentially what we mean by similar. Our choice of attributed reflects the level of detail at which we wish to represent information in crisis. For e.g. In this case we will store the name, registry umber, the course enrolled of the student and not his/her address or the gender.

**Relationships:**

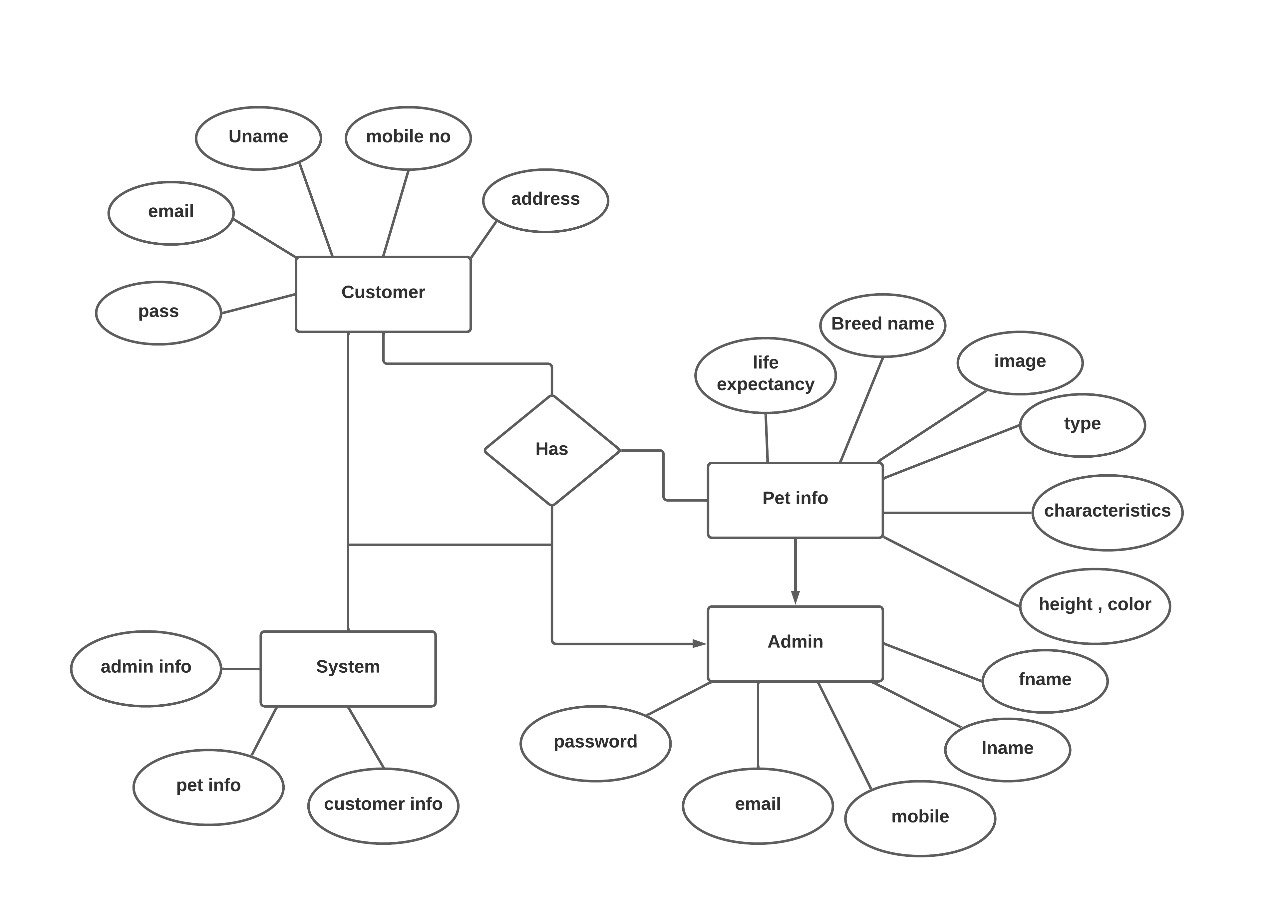
A Relationship describes relations between entities. There are 4 relationships of relational database and they are:

**One to One:** This type of relationship is rarely seen in real world.

**One to Many:** It reflects business rule that one entity is associated with many number of same entity. For example, one student can enrol for many courses, but one course will have one Student.

**Many to One:** It reflects business rule that many entities can be associated with just one entity. For example, Student enrols for only one Course but a Course can have many Students.

**Many to Many:** For example, many students can enrol for more than one course.

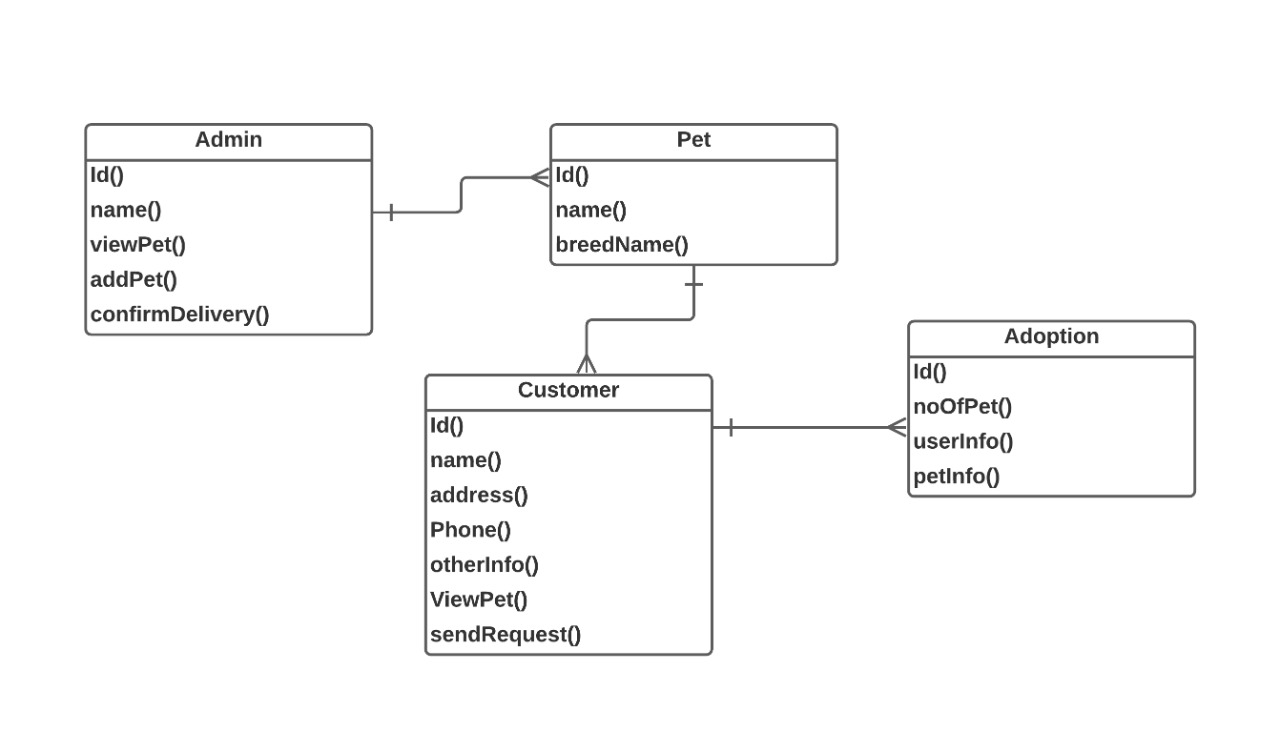


**Figure 3.1: ER Diagram**

##### Class Diagram

A class diagram is an illustration of the relationships and source code dependencies among classes in the Unified Modelling Language (UML). In this context, a class defines the methods and variables in an object, which is a specific entity in a program or the unit of code representing that entity. Class diagrams are useful in all forms of object-oriented programming (OOP). The concept is several years old but has been refined as OOP modelling paradigms have evolved.

In a class diagram, the classes are arranged in groups that share common characteristics. A class diagram resembles a flowchart in which classes are portrayed as boxes, each box having three rectangles inside. The top rectangle contains the name of the class; the middle rectangle contains the attributes of the class; the lower rectangle contains the methods, also called operations, of the class. Lines, which may have arrows at one or both ends, connect the boxes. These lines define the relationships, also called associations, between the classes.



**Figure 3.2: Class Diagram**

#### Activity Diagram

Activity diagram is another important diagram in UML to describe the dynamic aspects of the system. Activity diagram is basically a flowchart to represent the flow from one activity to another activity. The activity can be described as an operation of the system.

The control flow is drawn from one operation to another. This flow can be sequential, branched, or concurrent. Activity diagrams deal with all type of flow control by using different elements such as fork, join, etc.

• **Initial State or Start Point:**

A small filled circle followed by an arrow represents the initial action state or the start point for any activity diagram.



* **Activity or Action State:**

An action state represents the non-interruptible action of objects.



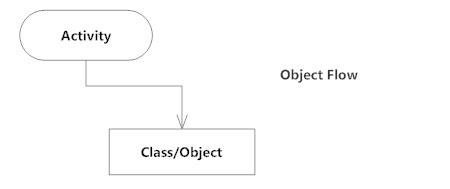
* **Action Flow:**

Action flows, also called edges and paths, illustrate the transitions from one action state to another. They are usually drawn with an arrowed line.



* **Object Flow:**

Object flow refers to the creation and modification of objects by activities. An object flow arrow from an action to an object means that the action creates or influences the object. An object flow arrow from an object to an action indicates that the action state uses the object.



* **Decisions and Branching:**

A diamond represents a decision with alternate paths. When an activity requires a decision prior to moving on to the next activity, add a diamond between the two activities. The outgoing alternates should be labeled with a condition or guard expression. You can also label one of the paths "else."

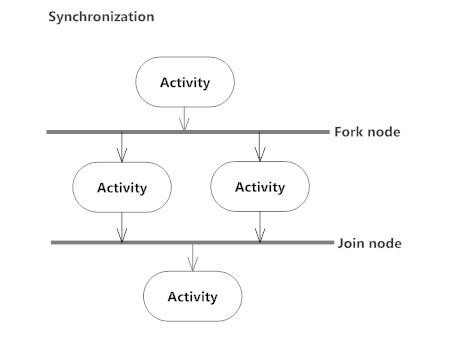


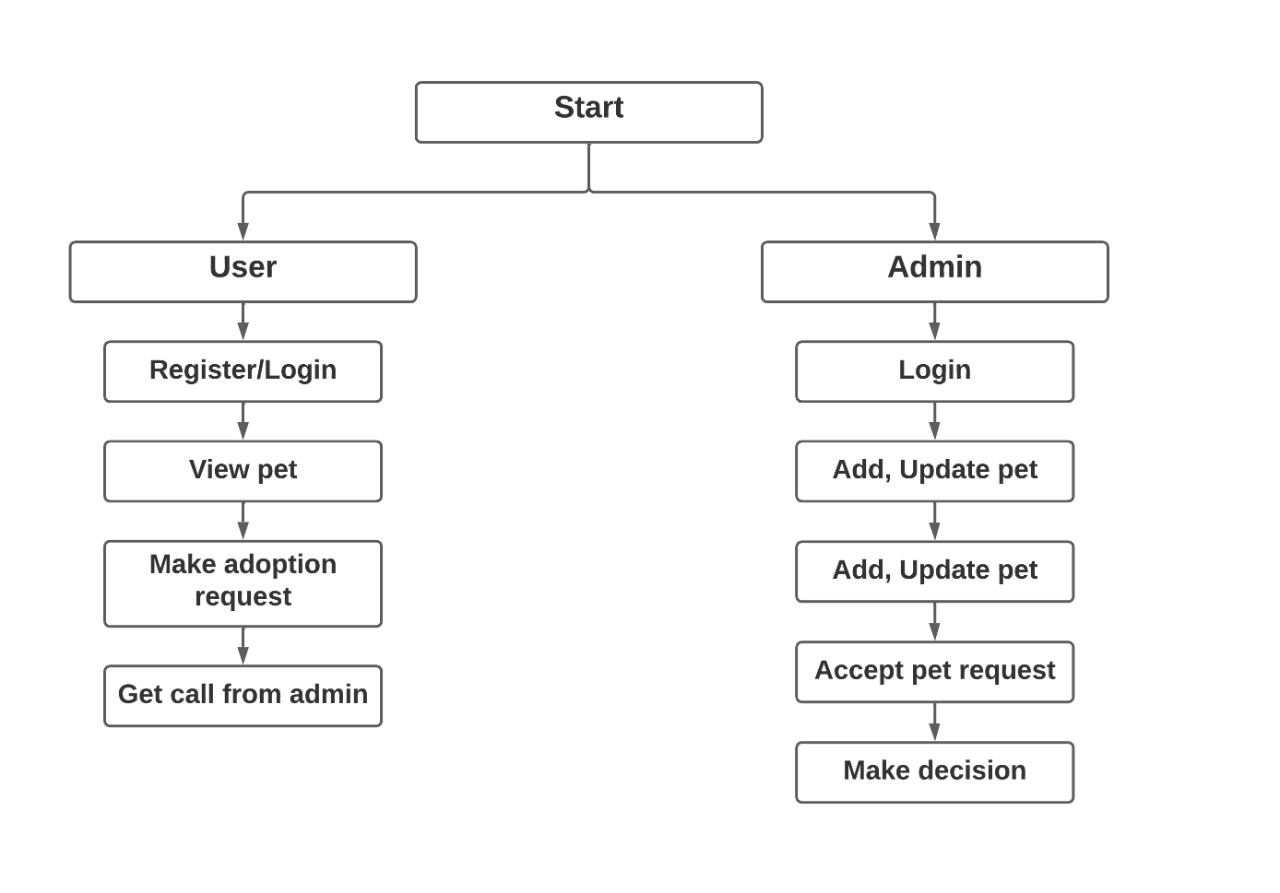
• **Synchronization:**

A fork node is used to split a single incoming flow into multiple concurrent flows. It is represented as a straight, slightly thicker line in an activity diagram.

A join node joins multiple concurrent flows back into a single outgoing flow.

A fork and join mode used together are often referred to as synchronization.



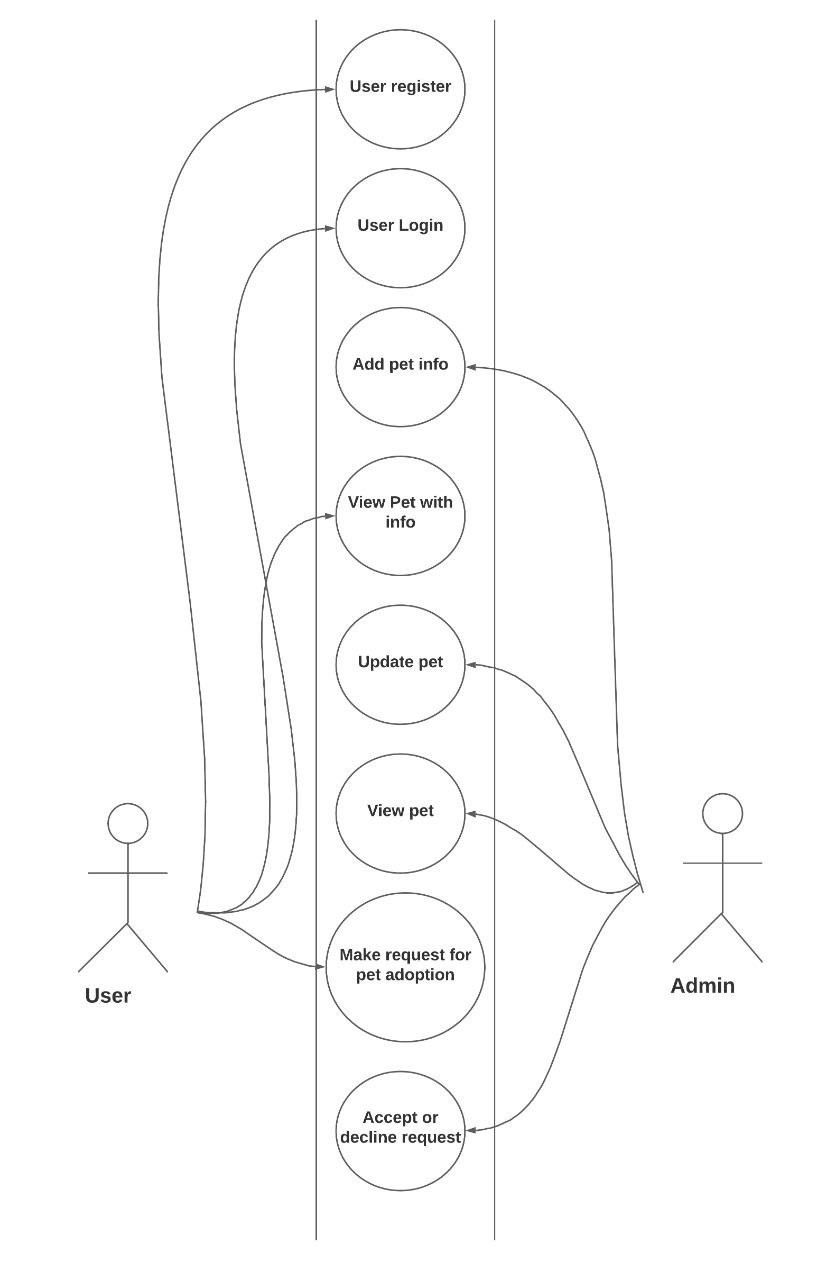
****

**Figure 3.3: Activity Diagram**

#### Use Case Diagram

Use case diagrams are usually referred to as behavior diagrams used to describe a set of actions (use cases) that some system or can perform in collaboration with one or more external user of the system (actors). Each use case should provide some observable and valuable result to the actors or other stakeholders of the system.

Use case diagrams are in fact twofold - they are both behavior diagrams, because they describe behavior of the system, and they are also structure diagrams - as a special case of class diagrams where classifiers are restricted to be either actors or use cases related to each other with associations.

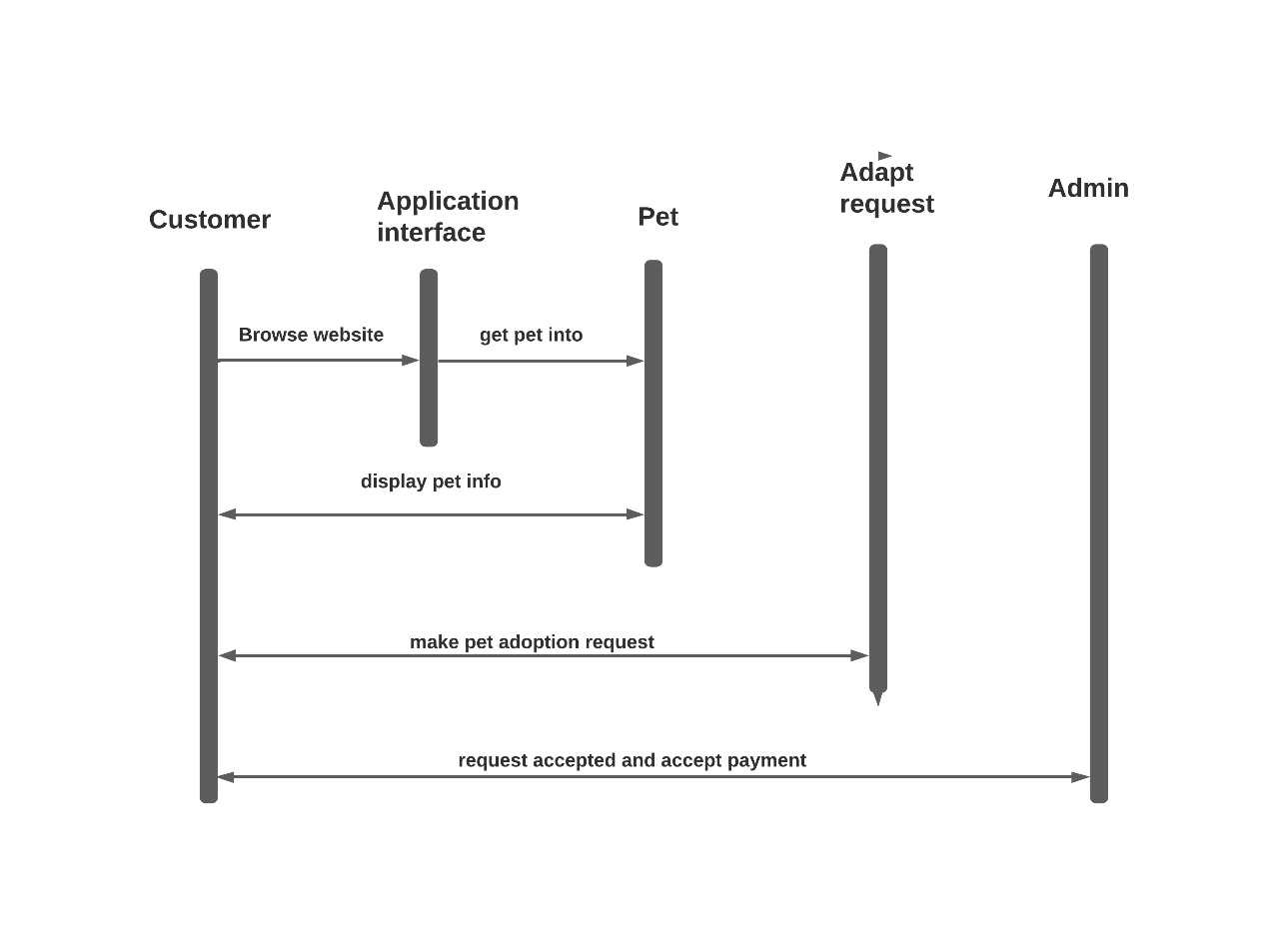


**Figure 3.4: Use Case Diagram**

#### Sequence Diagram

A sequence diagram shows object interactions arranged in time sequence. It depicts the objects and classes involved in the scenario and the sequence of messages exchanged between the objects needed to carry out the functionality of the scenario. Sequence diagrams are typically associated with use case realizations in the Logical View of the system under development. Sequence diagrams are sometimes called event diagrams or event scenarios.

A sequence diagram shows, as parallel vertical lines (*lifelines*), different processes or objects that live simultaneously, and, as horizontal arrows, the messages exchanged between them, in the order in which they occur. This allows the specification of simple runtime scenarios in a graphical manner.



**Figure 3.5: Sequence Diagram**

#### Deployment Diagram

Deployment diagrams are used to visualize the topology of the physical components of a system, where the software components are deployed. Deployment diagrams are used to describe the static deployment view of a system. Deployment diagrams consist of nodes and their relationships.

**DATABASE**

**Database Server**

**WEB SERVER**

**User**

**Admin**

Client Machine Web Browser

**Figure 3.6: Deployment Diagram**

## Chapter 4 System Design

### 4.1 Basic Module

A **work-breakdown structure** (**WBS**) in project management and systems engineering, is a deliverable -oriented breakdown of a project into smaller components. A work breakdown structure is a key project deliverable that organizes the team's work into manageable sections. The Project Management Body of Knowledge (PMBOK 5) defines the work-breakdown structure "A hierarchical decomposition of the total scope of work to be carried out by the project team to accomplish the project objectives and create the required deliverables."

A work-breakdown structure element may be a product, data, service, or any combination thereof. A WBS also provides the necessary framework for detailed cost estimating and control along with providing guidance for schedule development and control.

LOGIN/REGISTRATION

ADMIN

USER

SELECT PETS

SEND ADOPTION

REQUEST

UPDATE

SYSTEM

ADD/REMOVE

PETS

SEND MAIL

GET ADOPTION

REQUEST

**Figure 4.1: Basic Module**

### 4.2 Data Design

A database is an inherent collection of data with some inherent meanings, designed, built, and populated with data for a specific purpose. The following guidelines are been followed during the database design:

* Descriptive names for the tables, columns and indexes
* Singular names for tables and columns
* Proper data type for each column

This document describes the tables that are used to design the software, its attributes, data type, constraints, and relationship among these tables.

#### 4.2.1 Data Dictionary

##### Table name: Admin

##### 

|  |  |  |  |
| --- | --- | --- | --- |
| Column | Type | Null | Description |
| Email | Varchar (50) | No | Email |
| Password | Varchar (50) | No | Password |

**Table 4.1**

##### Table name: Login

|  |  |  |  |
| --- | --- | --- | --- |
| Column | Type | Null | Description |
| Email | Varchar (50) | No | Email |
| Password | Varchar (50) | No | Password |

**Table 4.2**

##### Table name: Animal

|  |  |  |  |
| --- | --- | --- | --- |
| Column | Type | Null | Description |
| ID | Integer (50) | No | Unique key to identify each ID |
| Type | Varchar (50) | No | Type |
| Breed Name | Varchar (50) | No | Name |
| Temperment | Varchar (50) | No | Temperment |
| About | Varchar (50) | No | About |
| Height | Integer (50) | No | Height |
| Weight | Integer (50) | No | Weight |
| Colour | Varchar (50) | No | Colour |
| Description | Varchar (100) | No | Description |

**Table 4.3**

##### Table name: Questionaire Form

|  |  |  |  |
| --- | --- | --- | --- |
| Column | Type | Null | Description |
| Id | Interger  (50) | No | Id |
| First Name | Varchar (50) | No | Unique key to identify each User.Username |
| Last.Name | Varchar (50) | No | User name |
| Email | Varchar (50) | No | Email |
| Phone | Integer (50) | No | Phone |
| Address | Varchar (50) | No | Address |
| Pet Breed | Varchar  (50) | No | PetBrees |
| Signature | Varchar  (50) | No | Signature |

**Table 4.5**

### 4.3 User Interface design

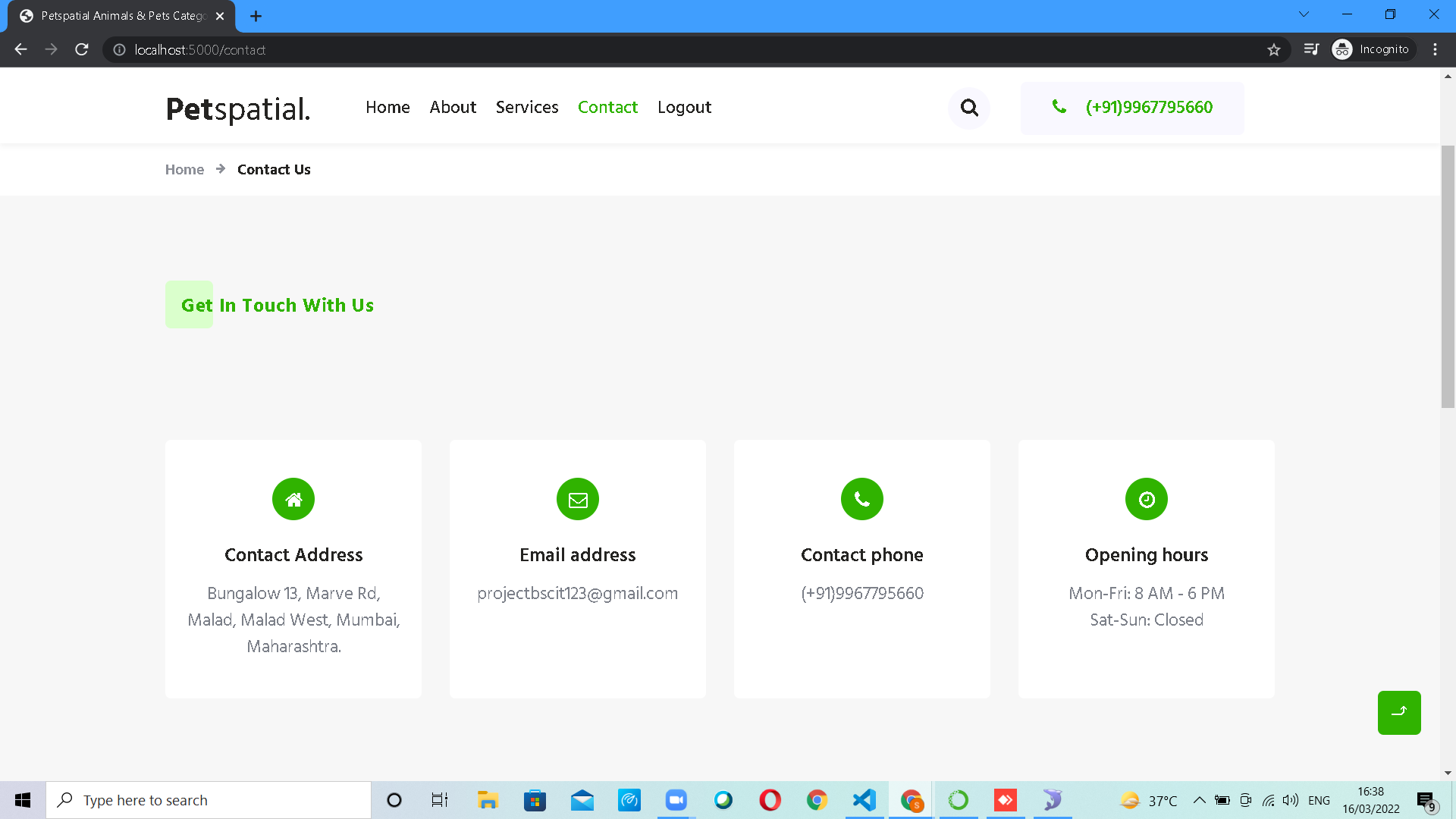
### User HomePage

### LoginPage

### Questionnaire Form

### Services

### ContactUs



### 4.4 Test cases design

System testing is designed to uncover the weaknesses that were not found in earlier test. In the testing phase, the program is executed with the explicit intention of finding errors. This includes forced system failures and validation of the system, as its user in the operational environment will implement it. For this purpose test cases are developed.

When a new system replaces the old one, such as in the present case, the organization can extract data from th e old system to test them on the new. Such data usually exist in sufficient volume to provide sample listings and they can create a realistic environment that ensures eventual system success. Regardless of the source of test data, the programmers and analyst will eventually conduct four different types of tests.

#### 4.4.1. Unit Testing

In computer programming, unit testing is a software testing method by which individual units of source code, sets of one or more computer program modules together with associated control data, usage procedures, and operating procedures, are tested to determine whether they are fit for use.

#### 4.4.2. System Testing

System testing is testing conducted on a complete integrated system to evaluate the system's compliance with its specified requirements.

System testing takes, as its input, all of the integrated components that have passed integration testing. The purpose of integration testing is to detect any inconsistencies between the units that are integrated together. System testing seeks to detect defects both within the "inter-assemblages" and also within the system as a whole. The actual result is the behavior produced or observed when a component or system is tested.

System testing is performed on the entire system in the context of either functional requirement specifications (FRS) or system requirement specification (SRS), or both. System testing tests not only the design, but also the behaviour and even the believed expectations of the customer. It is also intended to test up to and beyond the bounds defined in the software or hardware requirements specification.

#### 4.4.3. Acceptance Testing

Acceptance Testing is the final level of software testing. The main aim of this testing is to determine the working process of the system by satisfying the required specifications and it is acceptable for delivery. It is also known as End-User Testing. It also works under the Black Box Testing Method.

#### 4.4.4. Test Case Design

##### Login Verification

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sr. No** | **Input Values** | **Test case** | **Conditional being checked** | **Result** |
| 1 | Login ID | Empty | Please Enter valid Login ID. | Successful |
| 3 | Login ID | Already | Login ID should be unique. | Successful |
|  |  | Exists or not |  |  |
| 4 | Password | Empty | Please Enter valid Password. | Successful |
| 5 | Password | If wrong  Password | Enter Password. | Successful |
| 6 | Password | Length | Length should be less than or equal to 10 character. | Successful |

##### Registration verification

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sr. No** | **Input Values** | **Test case** | **Conditional being checked** | **Result** |
| 1 | Name | Empty | Name must not be empty. | Successful |
| 2 | Email ID | Empty | Minimum 8 characters required Enter valid Email ID. | Successful |
| 3 | Mobile Number | Length | Minimum 10 digits required Enter valid Email ID. | Successful |
| 4 | Password | Empty | Enter valid Password. | Successful |
| 5 | Confirm  Password | Empty | Password and confirmation password must be same. | Successful |

### 4.5 Security Issues

#### • OTP

A one-time password (OTP), also known as one time pin is a password that is valid for only one login session or transaction on a computer system or other digital device. OTP’s avoid a number of short comings that are associated with tradition (static) password based authentication by ensuring that one time password requires access to something a person has (such as a small key ring from device with the OTP calculator build into it, or a smart card or specific cell phone) as well as something a person knows (such as a PIN).

OTP generation algorithms typically make use of pseudo randomness or randomness to generate a shared key or seed, and cryptographic hash functions, which can be used to derive a value but are hard to reverse and therefore difficult for an attacker to obtain the data that was used for the hash. This is necessary because otherwise, it would be easy to predict future OTPs by observing previous ones.

OTPs have been discussed as a possible replacement for, as well as an enhancer to, traditional passwords. On the downside, OTPs can be intercepted or rerouted, and hard tokens can get lost, damaged, or stolen. Many systems that use OTPs do not securely implement them, and attackers can still learn the password through phishing attacks to impersonate the authorized user.

## Chapter 5 Implementation and Testing

### 5.1 Algorithm of Important Modules

### 5.1.1 Login.html

<!DOCTYPE html>

<html lang="en">

<head>

<title>Login V1</title>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1">

<!--===============================================================================================-->

<link rel="icon" type="image/png" href="../static/login\_images/icons/favicon.ico" />

<!--===============================================================================================-->

<link rel="stylesheet" type="text/css"

href="{{ url\_for('static',filename='login\_vendor/bootstrap/css/bootstrap.min.css') }}">

<!--===============================================================================================-->

<link rel="stylesheet" type="text/css"

href="{{ url\_for('static',filename='login\_fonts/font-awesome-4.7.0/css/font-awesome.min.css') }}">

<!--===============================================================================================-->

<link rel="stylesheet" type="text/css" href="{{ url\_for('static',filename='login\_vendor/animate/animate.css') }}">

<!--===============================================================================================-->

<link rel="stylesheet" type="text/css"

href="{{ url\_for('static',filename='login\_vendor/css-hamburgers/hamburgers.min.css') }}">

<!--===============================================================================================-->

<link rel="stylesheet" type="text/css"

href="{{ url\_for('static',filename='login\_vendor/select2/select2.min.css') }}">

<!--===============================================================================================-->

<link rel="stylesheet" type="text/css" href="{{ url\_for('static',filename='login\_css/util.css') }}">

<link rel="stylesheet" type="text/css" href="{{ url\_for('static',filename='login\_css/main.css') }}">

<!--===============================================================================================-->

</head>

<body>

<div class="limiter">

<div class="container-login100">

<div class="wrap-login100">

<div class="login100-pic js-tilt" data-tilt>

<img src="../static/login\_images/img-01.png" alt="IMG">

</div>

<form action="/login" method="post" class="login100-form validate-form">

<span class="login100-form-title">

Member Login

</span>

<div class="alert alert-success container" role="alert">

{{msg}}

</div>

<div class="wrap-input100 validate-input" data-validate="Valid email is required: ex@abc.xyz">

<input class="input100" type="email" name="email" placeholder="Email">

<span class="focus-input100"></span>

<span class="symbol-input100">

<i class="fa fa-envelope" aria-hidden="true"></i>

</span>

</div>

<div class="wrap-input100 validate-input" data-validate="Password is required">

<input class="input100" type="password" name="pass" placeholder="Password">

<span class="focus-input100"></span>

<span class="symbol-input100">

<i class="fa fa-lock" aria-hidden="true"></i>

</span>

</div>

<div class="container-login100-form-btn">

<button class="login100-form-btn">

Login

</button>

</div>

<div class="text-center p-t-12">

<span class="txt1">

For admin login

</span>

<a class="txt2" href="/adlogin">

click here!

</a>

</div>

<div class="text-center p-t-136">

To create your account<a class="txt2" href="/register">

Click Here!

<i class="fa fa-long-arrow-right m-l-5" aria-hidden="true"></i>

</a>

</div>

</form>

</div>

</div>

</div>

<!--===============================================================================================-->

<script src="{{ url\_for('static',filename='login\_vendor/jquery/jquery-3.2.1.min.js') }}"></script>

<!--===============================================================================================-->

<script src="{{ url\_for('static',filename='login\_vendor/bootstrap/js/popper.js') }}"></script>

<script src="{{ url\_for('static',filename='login\_vendor/bootstrap/js/bootstrap.min.js') }}"></script>

<!--===============================================================================================-->

<script src="{{ url\_for('static',filename='login\_vendor/select2/select2.min.js') }}"></script>

<!--===============================================================================================-->

<script src="{{ url\_for('static',filename='login\_vendor/tilt/tilt.jquery.min.js') }}"></script>

<script>

$('.js-tilt').tilt({

scale: 1.1

})

</script>

<!--===============================================================================================-->

<script src="{{ url\_for('static',filename='login\_js/main.js') }}"></script>

</body>

</html>

### 5.1.2 Home.html

<!doctype html>

<html lang="zxx">

<head>

<!-- Required meta tags -->

<meta charset="utf-8">

<meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">

<title>Petspatial Animals & Pets Category Bootstrap Responsive Web Template | Home :: W3Layouts </title>

<link href="//fonts.googleapis.com/css2?family=Hind+Siliguri:wght@300;400;500;600;700&display=swap" rel="stylesheet">

<!-- Template CSS -->

<link rel="stylesheet" href="{{ url\_for('static',filename='css/style-starter.css') }}">

</head>

<body>

<header id="site-header" class="fixed-top">

<section class="w3l-header-4">

<div class="container">

<nav class="navbar navbar-expand-lg navbar-light">

<h1> <a class="navbar-brand" href="/">

Pet<span class="sublog">spatial.</span>

</a></h1>

<button class="navbar-toggler collapsed" type="button" data-toggle="collapse" data-target="#navbarNav"

aria-controls="navbarNav" aria-expanded="false" aria-label="Toggle navigation">

<span class="fa icon-expand fa-bars"></span>

<span class="fa icon-close fa-times"></span>

</button>

<div class="collapse navbar-collapse" id="navbarNav">

<ul class="navbar-nav">

<li class="nav-item active">

<a class="nav-link" href="/">Home </a>

</li>

<li class="nav-item">

<a class="nav-link" href="#">About</a>

</li>

<li class="nav-item">

<a class="nav-link" href="#">Services</a>

</li>

<li class="nav-item">

<a class="nav-link" href="#">Contact</a>

</li>

<li class="nav-item">

<a class="nav-link" href="/logout">Logout</a>

</li>

</ul>

<ul class="navbar-nav search-right mt-lg-0 mt-2">

<li class="nav-item mx-xl-4"><a href="/login"

class="btn btn-primary btn-white d-none d-lg-block btn-style mr-2"> Login</a>

</li>

<li class="nav-item mx-xl-4"><a href="/register"

class="btn btn-primary btn-white d-none d-lg-block btn-style mr-2"> Register</a>

</li>

</ul>

<!-- search popup -->

<div id="search" class="pop-overlay">

<div class="popup">

<form action="#" method="GET" class="d-sm-flex">

<input type="search" placeholder="Search.." name="search" required="required" autofocus>

<button type="submit"><span class="fa fa-search"></span></button>

<a class="close" href="#">&times;</a>

</form>

</div>

</div>

<!-- /search popup -->

</div>

</nav>

</div>

</section>

</header>

<!--//header-->

<!-- main-slider -->

<section class="w3l-main-slider banner-slider" id="home">

<div class="owl-one owl-carousel owl-theme">

<div class="item">

<div class="slider-info banner-view banner-top1">

<div class="container">

<div class="banner-info">

<h3>Adopt Don't Shop.</h3>

<div class="banner-info-top">

<p>Approximately 1478 dogs & cats die every day on road in India.

The Petspatial is on a mission to provide every dog and cat a home before 2035.

It's just one of the many ways The Petspatial gives back and helps you become a part of something

larger. </p>

</div>

</div>

<div class="banner-info-top-grids">

<h5 class="text-lg-left"><strong>Our Location : </strong> Mumbai. </h5>

<h5 class="text-lg-right"><strong> Open Hours : </strong> Mon - Sat 8am - 6pm</h5>

</div>

</div>

</div>

</div>

<div class="item">

<div class="slider-info banner-view banner-top2">

<div class="container">

<div class="banner-info">

<h3>A little Changes Make a Big Difference.</h3>

<div class="banner-info-top">

<p>Take a look at our newest arrivals. Will one of these lovelies be coming to your home soon ? </p>

</div>

</div>

<div class="banner-info-top-grids">

<h5 class="text-lg-left"><strong>Our Location : </strong> Mumbai.</h5>

<h5 class="text-lg-right"><strong> Open Hours : </strong> Mon - Sat 8am - 6pm</h5>

</div>

</div>

</div>

</div>

<div class="item">

<div class="slider-info banner-view banner-top3">

<div class="container">

<div class="banner-info">

<h3>Save a life Adopt a Pet.</h3>

<div class="banner-info-top">

<p>Adoption appointments can now be made online. Meet the newest member of your family !</p>

</div>

</div>

<div class="banner-info-top-grids">

<h5 class="text-lg-left"><strong>Our Location : </strong> Mumbai. </h5>

<h5 class="text-lg-right"><strong> Open Hours : </strong> Mon - Sat 8am - 6pm</h5>

</div>

</div>

</div>

</div>

</div>

</section>

<!-- /main-slider -->

<!--#-section-->

<section class="w3l-index3" id="about">

<div class="midd-w3 py-5">

<div class="container py-lg-5 py-md-3">

<div class="row">

<div class="col-lg-6 mb-lg-0 mb-md-5 mb-4 align-self pr-lg-5">

<div class="title-content text-left">

<h6 class="title-subhny mb-2"><span>Love Your Pet</span></h6>

<h3 class="title-w3l">Why Should You Adopt a Dog or Cat?</h3>

</div>

<p class="mt-md-4 mt-3">Did you know that over 2000 people per hour in India run a search right here looking

to adopt a pet? Pet adoption is

becoming the preferred way to find a new pet. Adoption will always be more convenient than buying a puppy

for sale from

a pet shop or finding a kitten for sale from a litter. Pet adoption brings less stress and more savings!

So what are you

waiting for? Go find that perfect pet for home!</p>

</div>

<div class="col-lg-6 mt-lg-0 mt-4">

<div class="position-relative">

<img src="../static/images/b1.jpg" alt="" class="radius-image img-fluid">

</div>

</div>

<div class="col-lg-6 mt-5 pt-lg-4">

<div class="position-relative">

<img src="../static/images/b2.jpg" alt="" class="radius-image img-fluid">

</div>

</div>

<div class="col-lg-6 mt-5 align-self pl-lg-5 pt-lg-4">

<div class="title-content text-left">

<h6 class="title-subhny mb-2"><span>Pet care</span></h6>

<h3 class="title-w3l">Does everybody in your household agree with this adoption?</h3>

</div>

<p class="mt-md-4 mt-3">It is important that all members of your household are aware that you're adopting a

dog and have no issues with this,

otherwise there can be tension that can hurt the dog or result in many different problems.</p>

</div>

</div>

</div>

</div>

</section>

<!--/#-section-->

<!-- footer -->

<footer class="w3l-footer-29-main">

<div class="footer-29-w3l py-5">

<div class="container py-lg-4">

<div class="row footer-top-29">

<div class="col-lg-4 col-md-6 footer-list-29 footer-1 pr-lg-5">

<div class="footer-logo mb-3">

<a class="footer-brand-logo" href="/">Pet<span class="sublog">spatial.</span></a>

</div>

<p>Welcome to Petspatial. We're all about getting homeless pets into homes. Petspatial is an online,

searchable database of

animals who need homes.</p>

<div class="main-social-footer-29 mt-4">

<a href="#facebook" class="facebook"><span class="fa fa-facebook"></span></a>

<a href="#twitter" class="twitter"><span class="fa fa-twitter"></span></a>

<a href="#instagram" class="instagram"><span class="fa fa-instagram"></span></a>

<a href="#linkedin" class="linkedin"><span class="fa fa-linkedin"></span></a>

</div>

</div>

<div class="col-lg-4 col-md-6 footer-list-29 footer-2 mt-sm-0 mt-5">

<h6 class="footer-title-29">Subscribe Newsletter

</h6>

<form action="#" method="post" class="forms-25-info mt-4 mb-2">

<div class="forms-gds">

<input type="email" name="" placeholder="Enter your email" required="">

<button class="btn btn-style btn-primary">Subscribe</button>

</div>

<p class="mt-4 text-left">By submitting this form, you agree to the privacy policy and terms of use</p>

</form>

</div>

<div class="col-lg-4 col-md-6 footer-list-29 footer-4 mt-lg-0 mt-5 pl-lg-5">

<h6 class="footer-title-29">Instagram

</h6>

<ul class="w3linst-imgs row">

<li class="col-4"><a href="#"><img src="../static/images/b1.jpg" alt=""

class="radius-image img-fluid"></a>

</li>

<li class="col-4"><a href="#"><img src="../static/images/b2.jpg" alt=""

class="radius-image img-fluid"></a>

</li>

<li class="col-4"><a href="#"><img src="../static/images/b3.jpg" alt=""

class="radius-image img-fluid"></a>

</li>

<li class="col-4 mt-4"><a href="#"><img src="../static/images/b4.jpg" alt=""

class="radius-image img-fluid"></a>

</li>

<li class="col-4 mt-4"><a href="#"><img src="../static/images/b5.jpg" alt=""

class="radius-image img-fluid"></a>

</li>

<li class="col-4 mt-4"><a href="#"><img src="../static/images/b6.jpg" alt=""

class="radius-image img-fluid"></a>

</li>

</ul>

</div>

</div>

</div>

</div>

<!-- //footer -->

<!-- copyright -->

<section class="w3l-copyright">

<div class="container">

<div class="row bottom-copies">

<p class="col-lg-8 copy-footer-29">© 2021 Petspatial. All rights reserved.</p>

<div class="col-lg-4 footer-list-29">

<ul class="d-flex text-lg-right">

<li><a href="#careers"> Careers</a></li>

<li class="mx-lg-5 mx-md-4 mx-3"><a href="#privacymy-lg-0 my-4">Privacy Policy</a></li>

<li><a href="#">Contact us</a></li>

</ul>

</div>

</div>

</div>

</section>

<!-- move top -->

<button onclick="topFunction()" id="movetop" title="Go to top">

&#10548;

</button>

<script>

// When the user scrolls down 20px from the top of the document, show the button

window.onscroll = function () {

scrollFunction()

};

function scrollFunction() {

if (document.body.scrollTop > 20 || document.documentElement.scrollTop > 20) {

document.getElementById("movetop").style.display = "block";

} else {

document.getElementById("movetop").style.display = "none";

}

}

// When the user clicks on the button, scroll to the top of the document

function topFunction() {

document.body.scrollTop = 0;

document.documentElement.scrollTop = 0;

}

</script>

<!-- /move top -->

</footer>

<!-- //copyright -->

<!-- Template JavaScript -->

<script src="{{ url\_for('static',filename='js/jquery-3.3.1.min.js') }}"></script>

<script src="{{ url\_for('static',filename='js/theme-change.js') }}"></script>

<!-- owlcarousel -->

<!-- owl carousel -->

<script src="{{ url\_for('static',filename='js/owl.carousel.js') }}"></script>

<!-- script for banner slider-->

<script>

$(document).ready(function () {

$('.owl-one').owlCarousel({

loop: true,

margin: 0,

nav: false,

responsiveClass: true,

autoplay: true,

autoplayTimeout: 5000,

autoplaySpeed: 1000,

autoplayHoverPause: false,

responsive: {

0: {

items: 1

},

480: {

items: 1

},

667: {

items: 1

},

1000: {

items: 1

}

}

})

})

</script>

<!-- //script -->

<!-- script for tesimonials carousel slider -->

<script>

$(document).ready(function () {

$("#owl-demo2").owlCarousel({

loop: true,

nav: false,

margin: 50,

responsiveClass: true,

responsive: {

0: {

items: 1,

nav: false

},

736: {

items: 1,

nav: false

},

991: {

items: 2,

margin: 30,

nav: false

},

1080: {

items: 2,

nav: false

}

}

})

})

</script>

<!-- //script for tesimonials carousel slider -->

<!-- stats number counter-->

<script src="{{ url\_for('static',filename='js/jquery.waypoints.min.js') }}"></script>

<script src="{{ url\_for('static',filename='js/jquery.countup.js') }}"></script>

<script>

$('.counter').countUp();

</script>

<!-- //stats number counter -->

<!-- video popup -->

<script src="{{ url\_for('static',filename='js/jquery.magnific-popup.min.js') }}"></script>

<script>

$(document).ready(function () {

$('.popup-with-zoom-anim').magnificPopup({

type: 'inline',

fixedContentPos: false,

fixedBgPos: true,

overflowY: 'auto',

closeBtnInside: true,

preloader: false,

midClick: true,

removalDelay: 300,

mainClass: 'my-mfp-zoom-in'

});

$('.popup-with-move-anim').magnificPopup({

type: 'inline',

fixedContentPos: false,

fixedBgPos: true,

overflowY: 'auto',

closeBtnInside: true,

preloader: false,

midClick: true,

removalDelay: 300,

mainClass: 'my-mfp-slide-bottom'

});

});

</script>

<!-- //video popup -->

<!--/MENU-JS-->

<script>

$(window).on("scroll", function () {

var scroll = $(window).scrollTop();

if (scroll >= 80) {

$("#site-header").addClass("nav-fixed");

} else {

$("#site-header").removeClass("nav-fixed");

}

});

//Main navigation Active Class Add Remove

$(".navbar-toggler").on("click", function () {

$("header").toggleClass("active");

});

$(document).on("ready", function () {

if ($(window).width() > 991) {

$("header").removeClass("active");

}

$(window).on("resize", function () {

if ($(window).width() > 991) {

$("header").removeClass("active");

}

});

});

</script>

<!--//MENU-JS-->

<script src="{{ url\_for('static',filename='js/bootstrap.min.js') }}"></script>

</body>

</html>

## Chapter 6 Conclusion

The system **“Petspatial**” has been developed to satisfy the need for client. The entire system is highly interactive. The system provides maximum flexibility to user. So, all validity checks are taken care by the system during the entry.

I would like you to take a moment and think about how this project could really help dogs in shelters that are in need of a forever home. Understand that even though owning a dog has many benefits, to own a dog is also to have a lot of responsibility. You must take good care of a dog and make sure you have time to spend with them. Helping animals in need is a great thing to do, as well as giving knowledge to people about the health benefits of owning a dog. Please share with family and friends the new app you tried and tell people that it can help dogs in their local animals .

## Chapter 7 Bibliography

### 7.1 References

1) Wikipedia – [www.wikipedia.com](http://www.wikipedia.com)

(System design tools reference)

2)Stackoverflow – [www.stackoverflow.com](http://www.stackoverflow.com)

(Image storing and retrieval)

3)W3Schools – [www.w3schools.com](http://www.w3schools.com)

(CSS tutorial)

4)Bootstrap – [www.getbootstrap.com](http://www.getbootstrap.com)

(Page Layouts)