PETZ-ONLINE PET SHOP

PROJECT REPORT

Submitted By

PAUL JACOB
Reg. No. CCASSCS017

for the award of the Degree of

Bachelor of Science (B.Sc.)

in Computer Science
(University of Calicut)

under the guidance of

Ms. Sini Thomas
Assistant Professor

BSc. COMPUTER SCIENCE
DEPARTMENT OF COMPUTER SCIENCE
CHRIST COLLEGE(Autonomous)
IRINJALAKUDA, KERALA
INDIA

March 2021

DEPARTMENT OF COMPUTER SCIENCE

CHRIST COLLEGE (AUTONOMOUS)

IRINJALAKUDA

CERTIFICATE

This is to certify that the project report entitled ***” Petz - Online Pet Shop”*** *is a bonfied record of the project work done by* ***Paul Jacob*** *in partial fulfillment of the requirement for the sixth semester of* ***Bachelor of Computer Science*** *in Department of Computer Science of* **CHRIST COLLEGE (AUTONOMOUS) IRINJALAKUDA**.

Ms. Sini Thomas
Assistant Professor,CS
Internal Guide

Ms. Viji Viswanathan
Head of the Department
Computer Science

EXTERNAL EXAMINER

INTERNAL EXAMINER

ABSTRACT

Petz-Online Pet Shop is an android application where the purchase and sale of pets took place. Today we came to know about the circumstances that most of the people depends on Local markets and nearby houses in order to buy a pet for their house. Through this project we are planning to digitalize this process by online marketing. Thus the individuals can also rely on other persons across the district in order to buy variety breeds of pets. People could also be able to come to know about the different price ranges for the same breed offered by different persons. Thus our app provides a platform where the users could buy or sell pets by providing the contact details of the buyers/sellers. Our project also includes a special feature for buying street dogs through the app whereas its maintenance and operation are carried out by different non-government organizations through a website. Through this, we desire to contribute something for reducing the count of street dogs in public places.

We are building the android application in a way that we creates a login for all users. After logging in, the user is directed towards the homepage where he/she could see the pets available for buying. Besides these, one could see options for selling pets and managing his/her account. The details of the pet, some of it's pictures, it's age(rough one) etc are all to be provided if one could sell a pet whereas the buyer can view pets classified into categories and can contact the seller if his desired pet is met. Moreover we are also providing a feature that allows the users to know about all the pet shops in the district so that they could buy pets or its food from there also. Besides normal users who can access the application, various NGOs could also login to our website to upload the details of street dogs so that they can be available for the users to buy. Thus our app makes a perfect choice for the ones who loves pets and seeks variant breeds of them.

DECLARATION

We hereby declare that this project work **"PETZ-ONLINE PET SHOP"** submitted to Christ College (Autonomous) Irinjalakuda, affiliated to Calicut University in partial fulfillment of the requirement for the award of the Bachelor of Computer Science, is a record of original work done by us, under the guidance of Ms. SINI THOMAS, Department of computer Science.

Place: Irinjalakuda

ASHIK C SURESH
MIBIN MURALEEDHARAN
PAUL JACOB
SREERAG M R

ACKNOWLEDGEMENT

First and foremost we like to thank Lord almighty for his providence and for being the guiding light throughout the project. We wish to express my sincere gratitude to our beloved Department head for giving me all the facilities for our project. We take this opportunity to express my gratitude to the class teacher Ms. SINI THOMAS and head of the department Ms. VIJI VISWANATHAN who has been supported us throughout the course of this project. We are thankful for her aspiring guidance and valuable advice during the project work. We express my sincere thanks to my project guide Ms. SINI THOMAS for supporting and guiding throughout the project.We would take this opportunity to specially thank all other faculty members for their constant and continuous motivation. Finally we would like to thank my family and friends for giving valuable advice and moral support throughout our project.

Contents

Chapter 1

1 Introduction

Our application ‘Petz-Online Pet Shop’ is a platform for buying and selling different categories of pets. Our project also enables the user for buying street dogs which are uploaded by various Non-Government Organisation. Through our project, users can directly contact the buyer/seller so that they are benefitted by removing the commission fee paid to agents while buying their favoured pets. Our project also promotes a mission to reduce the count of street dogs in public places as users have an option to buy them through Non-Government Organisations.

1.1 Overview

The objective of the Petz-Online Pet Shop is to design a simple and adaptable website and mobile application for purchase and sale of pets which helps the customers decrease the commission fee paid to the agents and also users can adopt streetdogs uploaded by various NGOs. It provides a unique page of mobile application for customers.The website includes many functions such as NGO registration, admin login,NGO login and some other features.

Chapter 2

2 System Analysis

2.1 Purpose

The main purpose of this application is to enable buying and selling of pets for users. It also aims at promoting the adoption of street dogs.

2.1.1 Existing System

Majority of the people still rely on local markets and dealers for buying their desired pets whereas a part of the money is to be sent to agents as commission. Currently the number of active mobile applications found in our area that provides purchase of pets are very limited. Many applications are available in foreign countries like USA, but it is rarely developed in our area. Proper and timely updation of data is not done in many such applications and therefore it is not recommended to use. Similarly, lack of good user interface is a main problem in these cases. Besides these, none of these apps consider street dogs. As of now there is no application where we can adopt street dogs and nurture them

- Inconsistency in data entry.
- Lack of security.
- Duplication of data entry.

2.1.2 Proposed System

Features and Advantages of Proposed system are:-

- Digitalize the process of pet marketing.
- Eliminating the commission paid to agents
- Computerization is important for every field because the data is secure in computer.
- This project reduces human efforts, saves time and resources to an extent.
- A special feature for buying street dogs
- Can locate nearby petshop through maps.
- An advanced search option for searching desired breeds of pets.
- Facility to send feedbacks and complaints.
- Attractive user interface.

2.2 Problem definition

A platform for users to purchase different categories of pets and to adopt street dogs.

2.3 Feasibility Study

Feasibility study is a process that identifies, describes and evaluates proposed system and selects the best system. During the study, the problem definition is solved and all aspects of problem to be included in the system are determined. Size of project, cost of benefits is also estimated with greater accuracy. A good feasibility study will show the strength and defects before the project is planned or budgeted.

2.3.1 Technical Feasibility

It determines whether the technology is needed for the proposed system is available and how it can be integrated with the government. Technical evaluation must assess whether the user have technical expert to understand and use the new system. This assessment is based on an outline design of system requirements, to determine whether proposal is technically and legally feasible. It is the evaluation of the hardware and software and how it meets the need of the proposed system.The technology used can be developed with the current equipment's and has the technical capacity to hold data required by the old system.

2.3.2 Economical Feasibility

The purpose of the economic feasibility assessment is to determine the positive economic benefits to the organization that the proposed system will provide. It includes quantification and identification. It identifies the financial benefits and cost associated with the development of the system. Economic feasibility is often known as the cost benefit analysis. To carry out an economic feasibility study it is necessary to estimate actual money value against activities needed for implementing the system.

While implementing our system we can ensure that the cost of prospective new venture will ultimately be profitable to the people. So we can say it is financially feasible.

2.3.3 Operational Feasibility

Operational feasibility is the measure of how well a proposed system solves the problems, and takes advantage of the opportunities identified during scope definition and how it satisfies the requirements identified in the requirement analysis phase of system development. It focuses on the degree to which the proposed development project fits in with the existing environment and objectives it regards to development schedule. Operational feasibility focuses on human, organizational and political.

2.4 Module Description

The system comprises of 3 major modules as follows:

2.4.1 Admin Module

Admin will be provided with the login ID and password in our website . Admin facilitates the registration of NGOs. The administrator is the trusted personality who can manage users and their information. The admin keeps the information as sensitive as possible. The users are validated and approved by the admin after verification. Similarly, pets posted for selling is also validated by the admin. In the admin side of the website, there is also an option to view complaints and feedbacks. The admin can initiate complaint management procedure whenever needed to block a particular person. For efficient management of user complaint and fraud activities, the user complaints will be validated and based on the authenticity; the fraud users will be banned from the application.

2.4.2 User Module

The user has to fill a registration form before using the application. Once registration, he has to wait till the administrator of the system to approve the registration. On successful approval from the admin , the user will be alerted and he can log in. On login, the user will be redirected to the home page in which different categories of pets are shown. By selecting any of these categories, the user could see the list of pets available for buying. There will be a search option where he can search for his area of interest in each categories. After selecting the category, the user could either search the breed name or select his desired one from the list of pets which containing their images and short details. Once the user selected a pet, he could be able to see its full details and also the contact details of its owner. From there, he could make a deal. Users also have an option to put the pets which he liked into a wishlist from where he can deal with the seller later by easy access. The users can add pets for selling in the same portal itself. He can select the category and add information and pictures about the pet for selling. Once the user submitted the details, it will be available in the list of buying after the admin validation. The user can also manage his account where he can edit or view his profile details. In it , a feature to show the pet shops nearby him is enabled so as he can buy pets/pet related items from there also. Users also have an option to send complaints to the admin in the case of fraudulent cases and also they can send feedbacks about the app to the admin.

2.4.3 NGO Module

The NGOs has to fill a registration form before accessing the website. Once registration, they had to wait till the administrator of the system to approve the registration. On successful approval from the admin, the NGOs will be alerted and he can log in. After logging in, the NGOs have an option to add street dogs into the application. Only various NGOs have the permission toad street dogs. There will be an option to list the added street dogs too.

Chapter 3

3 Software Requirement Specification

3.1 Purpose

The purpose of this document is to give a detailed description of the requirements for the Petz-Online Pet Shop Application. It will illustrate the purpose and complete declaration for the development of the system. It will also explain system constraints, interface and interactions with other external applications. This document is primarily intended to assist users to buy and sell pets online without commission fee and to adopt street dogs.

3.2 Scope

This is an Android based mobile application that is developed to help users for selling and buying pets. The mobile application with Android Java at the front end Php-MySQL at the backend, will record pet related activities in a secure manner. The concept of this Petz-Online Pet Shop is to sell and buy pets.

3.3 Overall Description

This section will give an overview of our system, Petz-Online Pet Shop. This project is designed for customers for selling and buying of different categories of pets and also street dogs provided through a unique page of mobile application. Besides these NGO's and admins can access our website to register and retrieve informations regarding the street dogs and other features. The website includes many functions such as NGO registration, admin login and some other features.

3.3.1 Product Perspective

Petz-Online Pet Shop is mainly used for pet lovers. It ensures greater profitability for them. The customer, NGOs and admin can view upload and update pet details.

3.3.2 Product Functionality

Through this system customers are able to buy and sell pets whereas admin can control the overall system like user and pet approval/removal. NGOs are able to upload the details of street dogs available.

3.3.3 Users and Characteristics

There are 2 types of users that interact with the site admin and NGOs .User can login through app. Each of these have different tasks which is performed.

Admin is able to remove users and pets. NGOs are able to add their street dogs. Users can buy farm pets and street dogs.

3.4 Specific Requirements

3.4.1 Hardware Requirements

Processor: Intel dual core or above
Processor speed :1.0GHZ or above
RAM :1 GB RAM or above
Hard Disk :20 GB hard disk or above

3.4.2 Software Requirements

Operating System : Any OS
Programming Language :PHP,Android
IDE: Android Studio
Database Server: SQL

3.5 Functional Requirements

Implementation is the stages of project when the theoretical design is turned into a working system. If the implementation stage is not properly planned and controlled, it can cause chaos. Thus it can be consider to be most crucial stage in achieving successful new system and in giving the users confidence that the new system will work and be effective. Normally this stage involves setting up a co-ordination committee, which will act as a sounding board of ideas, complaints and problems. The first task is implementation planning ie; of the methods and time scale to be adopted. Apart from planning, the two major task of repairing for implementation are education and training of administrator and testing of the system. After the implementation phase is completed and the user staff adjusted to the changes created by the candidate system, evaluation and maintenance is continue to bring the new system standards. The activities of the implementation phase can be summarized as;

- Implementation planning.
- Eduction and training
- System training

3.6 Non Functional Requirements

- **Performance requirements:**The response time should not vary with the increasing the size of the data storage.

- **Security requirements:** This application should not modify any vaccination details because the modules are authenticated.The username and password should be a unique token id
- **Training requirements:** Training has to be provided by the supervisor
- **Validation and verification:**Checks all the fields are filled and valid.The password is invisible by applying dots

3.7 Security Requirements

The system is implemented after through checking is done and if it is found working according to the specification. It involves careful planning investigation of the current system and constraints on implementation, design of methods to achieve. Two checking is done and if it is found working according to the major task of preparing the implementation are educating, training the administrator. The implementation process begins with preparing a plan for the implementation of the system. According to this plan, the activities are to be carried out, discussions made regarding the equipment, recourses and the additional equipment that has to be acquired to implement the new system. The most important thing in the implementation stage is, gaining the users confidence that the system will work effectively. The system can be implemented only after the testing is done. This method also offers the greatest security since the existing system can take over if the errors are found or inability to handle certain type of transaction while using the new system. Security involves both policies and mechanisms to protect the data and ensures that it is not accessed, altered or deleted without proper authorization. There are two dimensions for the protection of data in the data base. The logged user can only perform the operations. This ensures that the confidentiality of the data is maintained, second, the data must be protected from accidental or intentional corruption or destruction. The security features are considered while developing the system, so as to avoid the errors and omissions that may lead to serious problems. The system may have to face the unwanted events called threats. A threat to a computer system is any events that adversely affect the one or more assets or resources, which make up the system. An event can be any of the following:

- Interruption of communication
- Destruction of hardware
- Modification of software
- Removal of programs.
- Disclosure of information.

There are many methods for handling a threat.

- Avoid it by altering the design.

- Threat retention.
- Threat reduction that is the frequency of occurrence of a threat is reduced.

The security measures of a computer system should be specified at an early stage in the design of the system. During the system operation each user should understand the procedures required to keep the system secure.

There are many possible threats to the security and integrity of any system where more than one user is associated with the system. Software integrity has become increasingly important. The attribute measures a system's ability to withstand attacks, both accidental and intentional on its security. Attacks can be made on all the three components of software: programs date and documents.

In this project the data security, data validation checking methods are applied using a password authentication. All the data, which is entered by the administrator, will be validated.

3.8 Platform and Technologies Used

3.8.1 About Front End

Android Studio is the official Integrated Development Environment (IDE) for Google's Android operating system, built on JetBrains' IntelliJ IDEA software and designed specifically for Android development. It is available for download on Windows, macOS and Linux based operating systems. It is a replacement for the Eclipse Android Development Tools (ADT) as primary IDE for native Android application development.Each project in Android Studio contains one or more modules with source code files and resource files. Types of modules include:

- Android app modules
- Library modules
- Google App Engine modules

By default, Android Studio displays your project files in the Android project view. This view is organized by modules to provide quick access to your project's key source files. All the build files are visible at the top level under Gradle Scripts and each app module contains the following folders:

- manifests: Contains the AndroidManifest.xml file.
- java: Contains the Java source code files, including JUnit test code.
- res: Contains all non-code resources, such as XML layouts, UI strings, and bitmap images.

Chapter 4

4 Design Document

4.1 Purpose

The purpose of this document is to give a detailed description of the requirements for the PETZ-ONLINE PET SHOP. It will illustrate the purpose and complete declaration for the development of the system. It will also explain system constraints, interface and interactions with other external applications. This document is primarily intended to be proposed to users to buy and sell pets .

4.2 Scope

To develop an application with Android Technology at the front end and PHP-MYSQL at the back end.

4.3 Over View

System design transforms a logical representation what the system is required to do into the physical specification. The specifications are converted into a physical reality during the development. Design forms a blueprint of the system and adds how the components relates to each other. The design phase proceeds accordingly to an ordinary sequence of steps, beginning with review and assigning of task and ending with package design. Design phase is the lifecycle phase in which the detailed design of the system selected in the study phase is accomplished. A smooth transition form the study phase to design is necessary because the design phase continues the activities in the earlier phase. Simplicity is the most important criteria of design phase. The most creative and challenging phase of the system life cycle is system design . The term design describes the final system and the process by which it is developed. The first step in design is to determine how the output is to be produced and in what format . Second the formats of input screens are to be determined. The input data and the master files have to be designed to meet the requirements of the proposed output.

4.3.1 Logical Design

The part of the design process that is independent of any specific hardware or software platform is referred to as logical design. During the logical design, all functional features of the system chosen for development in analysis phase are described independently of any computer platform.

4.3.2 Physical Design

Physical design is the part of the design in which the logical specification of the system from logical design are transferred into technology-specific details from which all programming and system construction can be accomplished. The system perform information output

4.3.3 Database Design

The objective of database design is to provide auxiliary storage and to contribute to be overall efficiency of the program component one auxiliary storage medium must provide efficient access to the data . The concept behind a database is an integrated collection of data and provides centralized access to the data from program. A database is an collection of logically related data stored with minimum redundancy to serve many users quickly and efficiently.

4.3.4 Primary key

A primary key is a special relational database table column (or combination of columns) designated to uniquely identify all table records. A primary key's main features are: It must contain a unique value for each row of data. It cannot contain null values.

4.3.5 Foreign key

A foreign key is a key used to link two tables together. This is sometimes also called as a referencing key. A Foreign Key is a column or a combination of columns whose values match a Primary Key in a different table.

4.4 Normalization

Normalization is a systematic approach of decomposing tables to eliminate data redundancy(repetition) and undesirable characteristics like Insertion, Update and Deletion Anomalies. It is a multi-step process that puts data into tabular form, removing duplicated data from the relation tables.

4.5 Tables

registration

Name	DataType	Constraints
id	int(11)	Primarykey
name	varchar(30)	Notnull
address	varchar(50)	Notnull
username	varchar(20)	Notnull
password	varchar(50)	Notnull
mobilenumber	varchar(20)	Notnull
gender	varchar(20)	Notnull

Table 1: registration

pets_uploads

Name	DataType	Constraints
pe_id	int(11)	Primarykey
item_category	varchar(20)	Notnull
item_breed	varchar(20)	Notnul
life_expectancy	varchar(10)	Notnull
item_price	varchar(20)	Notnull
item_image	varchar(20)	Notnull
userid	varchar(25)	Notnull
contact	varchar(20)	Notnull

Table 2: pets_uploads

streetdog

Name	DataType	Constraints
id	int(11)	primarykey
dogname	varchar(20)	Notnull
breed	varchar(20)	Notnull
age	varchar(20)	Notnull
features	varchar(200)	Notnull
price	varchar(20)	Notnull
dogimage	varchar(50)	Notnull
certificate	varchar(50)	Notnull
contact	varchar(30)	Notnull

Table 3: streetdog

wishlist

Name	DataType	Constraints
s_id	varchar(50)	Notnull
item_id	varchar(50)	Notnull
user_id	varchar(50)	Notnull
sale_date	varchar(50)	Notnull
item_category	varchar(50)	Notnull
item_breed	varchar(25)	Notnul
life_expectancy	varchar(10)	Notnull
item_price	varchar(20)	Notnull
item_image	varchar(20)	Notnull
userid	varchar(25)	Notnull
contact	varchar(25)	Notnull

Table 4: wishlist

complaints

Name	DataType	Constraints
c_id	int(11)	Primarykey
email	varchar(30)	Notnull
complaints	varchar(100)	Notnull

Table 5: complaints

feedback

Name	DataType	Constraints
f_id	int(11)	Primarykey
email	varchar(20)	Notnull
feedback	varchar(100)	Notnull

Table 6: feedback

ngo

Name	DataType	Constraints
name	int(30)	Primarykey
place	varchar(50)	Notnull
contact	varchar(30)	Notnull
password	varchar(20)	Notnull

Table 7: ngo

Chapter 5

5 System Testing

Software testing is a critical element of software quality assurance and represents the ultimate review of specification ,design and coding. Testing represents an interesting anomaly or the software. Testing is vital to the success of the system. Errors can be injected at any state during development. System testing makes a logical assumption that all the part of the system is correct,the goal will be successfully achieved. During testing the program to be tested is executed with set of data and the output of the program for the test data is evaluated to determine if the program is performing as expected. A series of testing are performed for the proposed system before the system is ready for user acceptance testing.

5.1 Unit Testing

Unit testing focuses verification effort on the smallest unit of the software design,the module this s known as module testing. Since the proposed system has modules the testing is individually performed on each module. Using the detailed design description as a guide, important control paths are tested to uncover errors within the boundary of the module. This testing was carried out during programming stage itself. In this testing step, each module is found to be working satisfactorily as regards to the expected output from the module.

5.2 Integration Testing

Data can be test across an interface,one module can have effect on another, sub function when combined may not produce the desired function. Integration testing is a systematic technique for constructing the program structure while at the same time conducting test to uncover errors associated within the interface.

5.3 Validation Testing

Validation testing can be defined in many ways, but a simple definition is that validation succeeds the software functions in manner that is reasonably expected by the customer. Software validation is achieved through a series of black box tests that demonstrates conformity with requirements.

5.4 Output Testing

After performing the validation testing, the next step is output testing of the proposed system. Since no system could be useful if it does not produce the required output in the specific format. The output generator is displayed by the system under consideration is tested by asking the users about the format required by them. Here the output is considered in two ways; One is on screen and the other is printed format. The output format on the screen is found to be correct as the format was designed in the system design phase according to the user needs. As far as hardcopies are considered, it goes in terms with the user requirement. Hence output testing does not results any correction in the system.

5.5 White Box Testing

White Box testing is the test case design method that uses the control structure of the procedural design to derive test cases. White Box testing of software is predicted on close examination of procedural details.

5.6 Black Box Testing

Black Box testing focuses on the functional requirements of the software. It helps to find out errors in incorrect or missing functions, interface errors, errors in data structures, performance errors and initialization and termination errors. The Black Box testing is applied during the later stages for the functional requirement evaluation.

Chapter 6

6 System Maintenance and Security

6.1 Maintenance

The maintenance phase of the software cycle is the time in which a software product forms useful work. After a system is successfully implemented, it should be maintained in a proper manner. System maintenance is an important aspect in the software development life cycle. The need for maintenance is to make it adaptable to the changes in the environment. There may be social, technical, and other environmental changes, which effects a system to be implemented. Software produce enhancement that may provide new functional capabilities, improving administrator displays and mode of interaction, upgrading the performance characteristics of the system. So only through proper system maintenance procedures, the system can be adapted to cop up with these changes. Software maintenance is of course far more than “Finding mistakes”. We may define maintenance by describing four activities. Those are undertaken after a program is released for use. The first maintenance activity occurs because it is unreasonable to assume that software testing will uncover all latent errors in a large software system. During the use of any large program, errors will occur and be reported to the developer. The process that include that the diagnosis and correction of one or more errors is called corrective maintenance. The second activity that contributes to a definition of maintenance occurs because of the rapid change that is encountered in every aspects of computing. Therefore adaptive maintenance is an activity that modifies software interface with a changing environment is so common. After a system is successfully implemented, it should be maintained in a proper manner. System maintenance is performing to make the system adaptable to the changing environment.

6.1.1 Corrective Maintenance

In software testing user will not uncover the errors in the system. During the use of any large program errors will occur and reported to the developer.

6.1.2 Adaptive Maintenance

It is an activity that modifies the entire system to properly interact with the changing environment.

6.1.3 Perfective Maintenance

It is performed to enhance the system by making modifications in the program to meet the users changing needs.

6.2 System Security

Security involves both policies and mechanisms to protect the data and ensures that it is not accessed, altered or deleted without proper authorization. There are two dimensions for the protection of data in the data base. The logged user can only perform the operations. This ensures that the confidentiality of the data is maintained, second, the data must be protected from accidental or intentional corruption or destruction. The security features are considered while developing the system, so as to avoid the errors and omissions that may lead to serious problems. The system may have to face the unwanted events called threats. A threat to a computer system is any events that adversely affect the one or more assets or resources, which make up the system. An event can be any of the following:

- Interruption of communication
- Destruction of hardware
- Modification of software
- Removal of programs.
- Disclosure of information.

There are many methods for handling a threat.

- Avoid it by altering the design.
- Threat retention.
- Threat reduction that is the frequency of occurrence of a threat is reduced.

The security measures of a computer system should be specified at an early stage in the design of the system. During the system operation each user should understand the procedures required to keep the system secure.

There are many possible threats to the security and integrity of any system where more than one user is associated with the system. Software integrity has become increasingly important. The attribute measures a system's ability to withstand attacks, both accidental and intentional on its security. Attacks can be made on all the three components of software: programs date and documents.

In this project the data security, data validation checking methods are applied using a password authentication. All the data, which is entered by the administrator, will be validated.

Chapter 7

7 Conclusion and Future Scope

7.1 Conclusion

As part of the project detailed study has been made about the designing and development aspects of the project “Petz-Online Pet Shop” which is an opening to the component world of computerization. We have tried our best to achieve our goals. The project has to meet all the requirements that were collected during analysis and designing phase. In this project, we will be designing a simple platform for buying and selling pets. Besides these we also provide users with an option to adopt street dogs with the help of NGOs. The main objectives are to avoid the middle man in dealings and to decrease the count of street dog in public places. The project is developed in such a way that it is able to undergo future enhancement in reliable, secure manner. The successful completion of this project has expanded my boundaries of imagination, invoked confidence, raised my creativity and has provided with knowledge and experience.

7.2 Future Scope

Following features can be upadated in Future

- To include all varieties of breeds.
- Home delivery of pet accessories.
- An in-app chat option between the seller and buyer..

Appendix

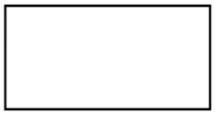
A Data Flow Diagram

Data flow is the one of the best way of documenting the entire functionality of the system.For the system ,which will have data flows in and have some processing inside and then some data flow out from the system can be documented or represented effectively by means of data flow out from the system can be documented or represented effectively by means of data flow diagrams. The data flow diagram are a diagrammatic representation of the system,which has input,process and output.Once any system is represented using a data flow diagram we can identify the following things easily:

- Various entities interacting with the system are identified
- Flow of data from one entity to another is identified
- The various processes involved in between the interaction of two or more entities in the system are clearly pointed out
- The various data stores which hold the data in between the process,are clearly identified

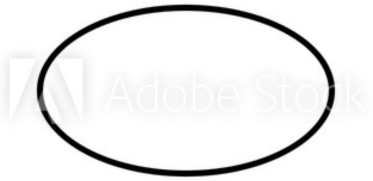
Some Data Flow Diagram charting forms:

A.1 External source or receiver



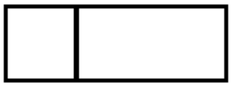
A source or sink is a person or part of organization ,which enters or receives information from the system,but is considered to be outlining the contest of data flow model.

A.2 Transform process



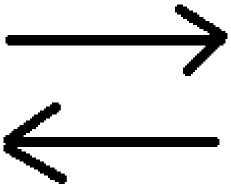
A process represents transformation where incoming data flowsare changed into outgoing data flow

A.3 Data Store



A data store is repository of data that is to be stored for use by one or more process may be as simple as buffer or queue or sophisticated as relational database. They should have clear names.If a process merely uses the contest of store and does not alter it,the arrowhead goes only form the store to the process. If a process alters the details in the store then double-headed arrow is used.

A.4 Data flow



A data flow is a route, which enable packets of data to travel from one point to another.Data may flow,with arrowhead pointing in the direction of the flow

B Data Flow Diagrams

B.1 Level 0

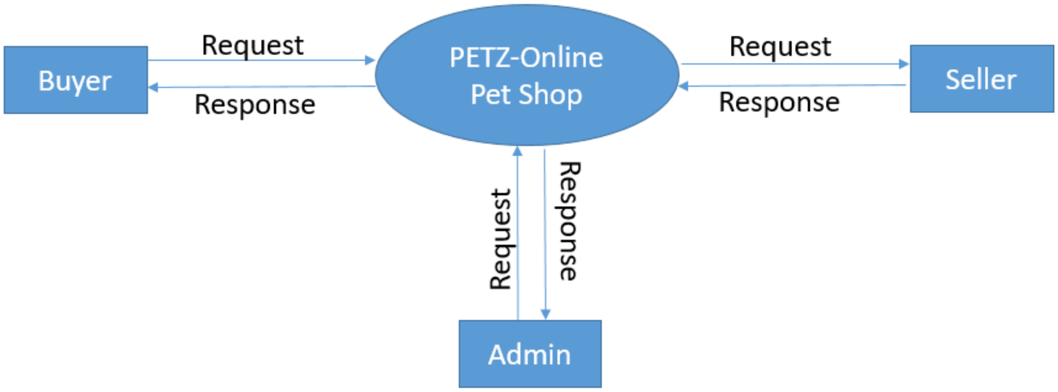


Figure 1: Level 0

B.2 Level 1-Seller

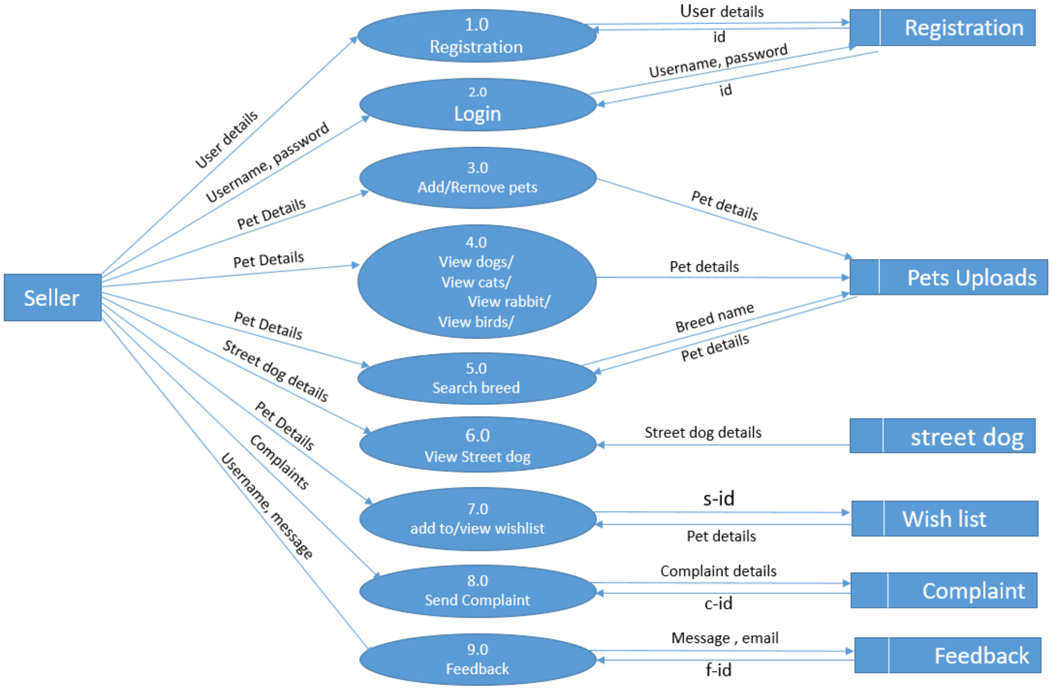


Figure 2: Level 1-Seller

B.3 Level 1-Buyer

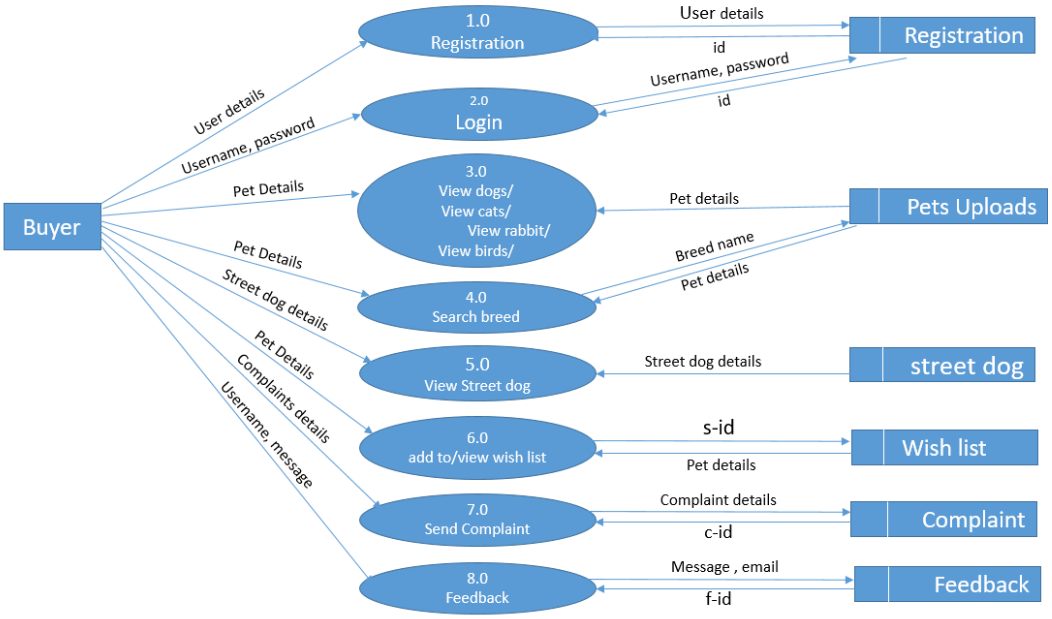


Figure 3: Level 1-Buyer

B.4 Level 1-Admin

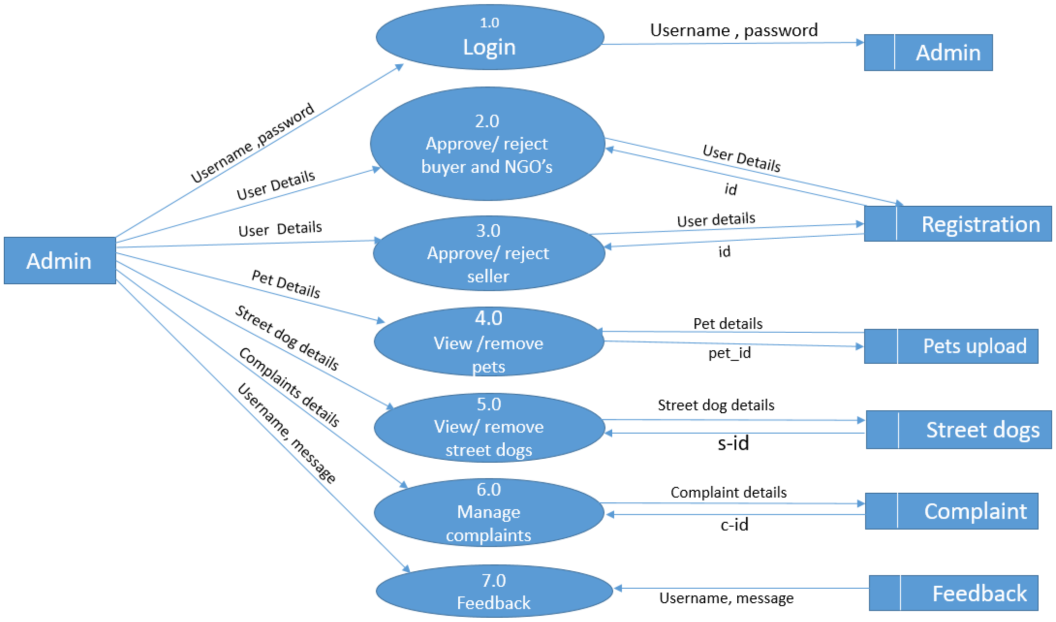


Figure 4: Level 1-Admin

B.5 Level 1-NGO

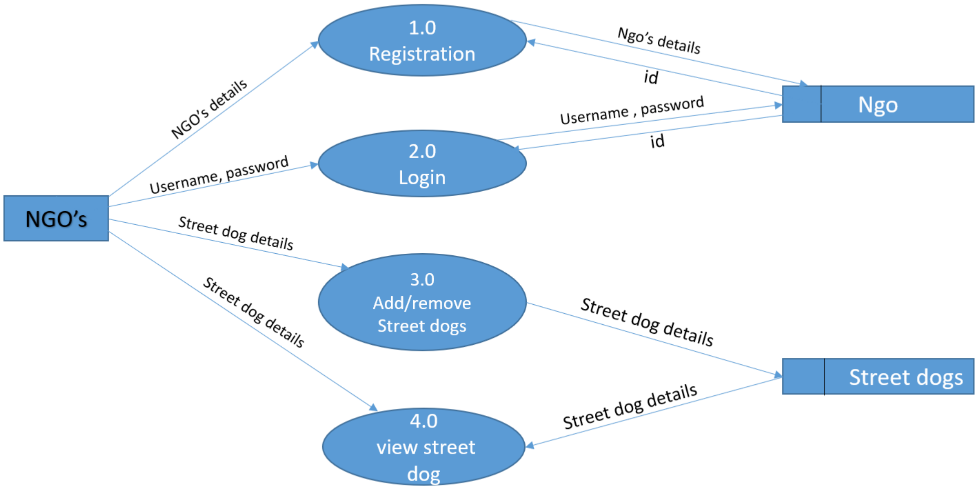


Figure 5: Level 1-NGO

E R Diagram

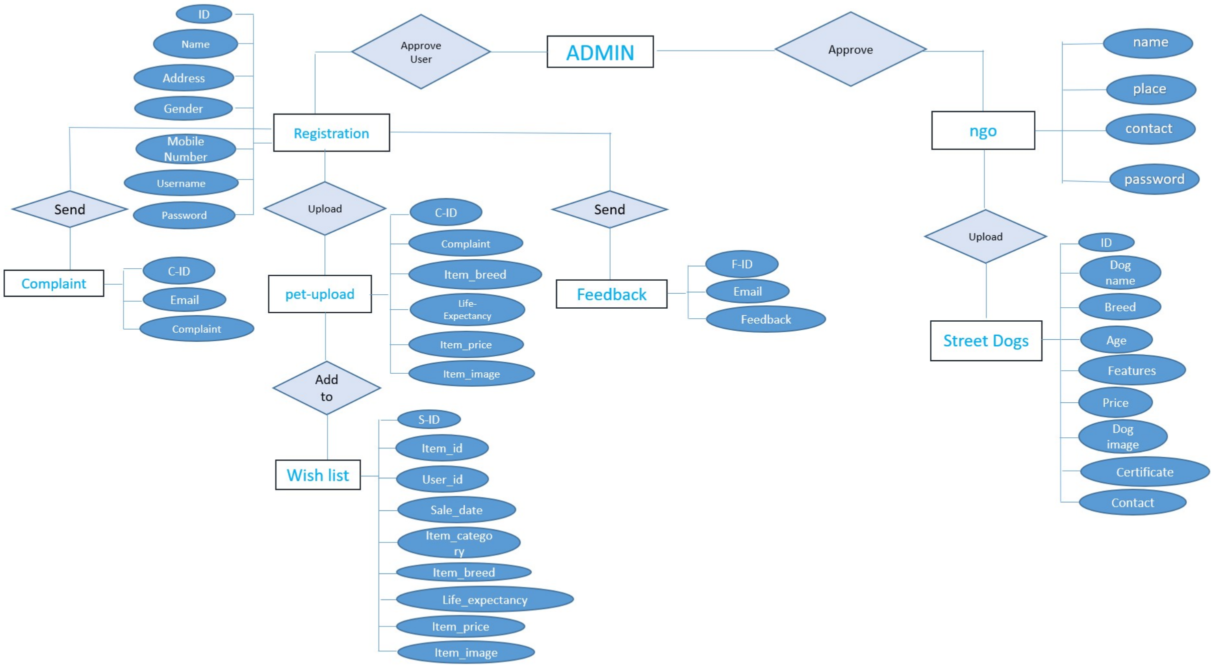
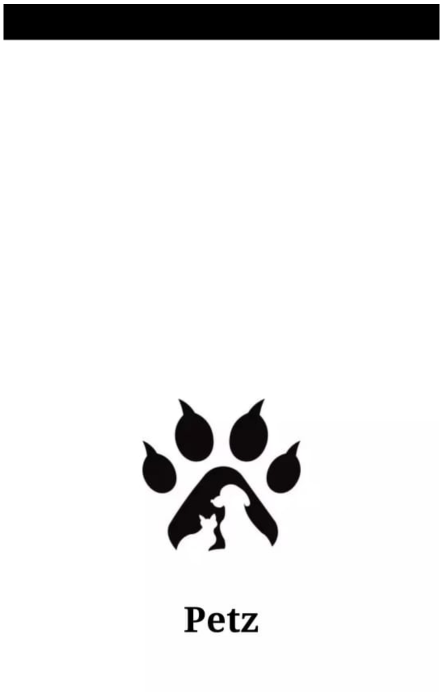
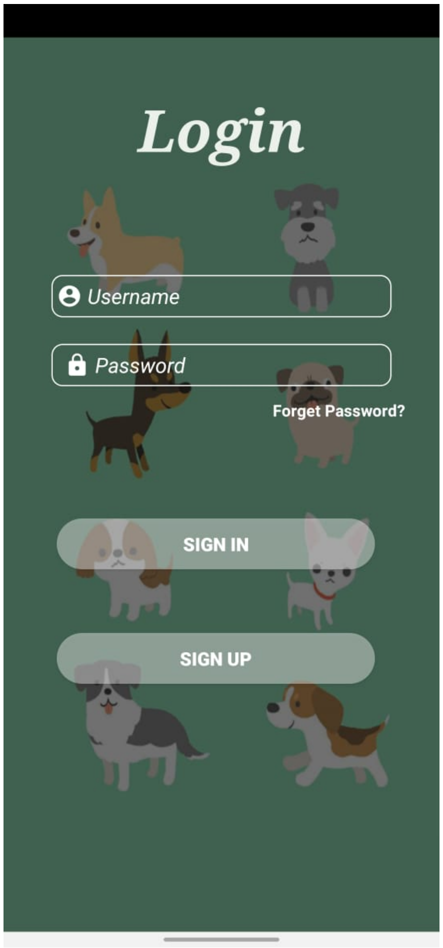
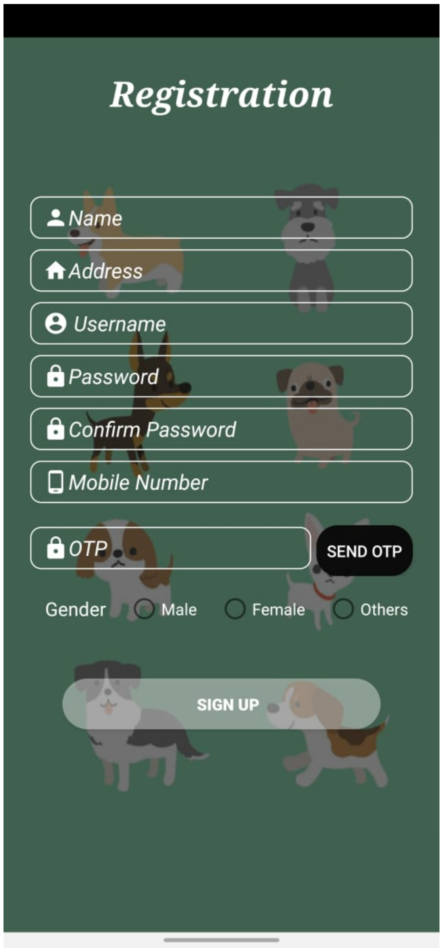


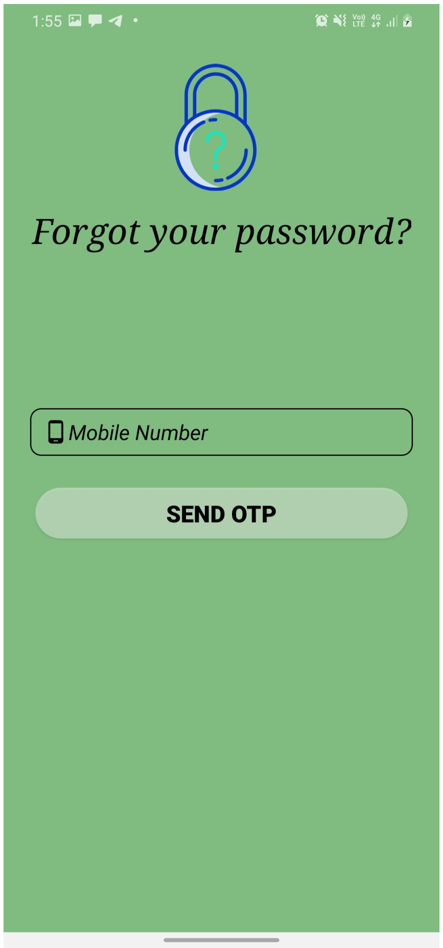
Figure 6: ER diagram

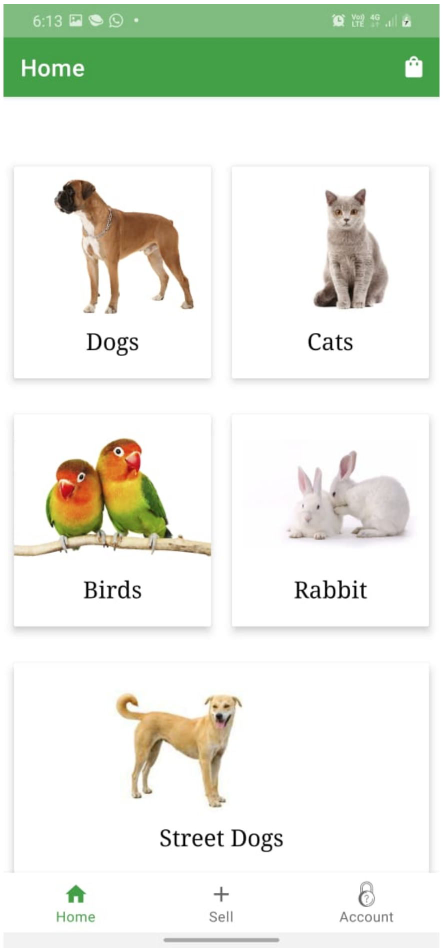
C User Interface

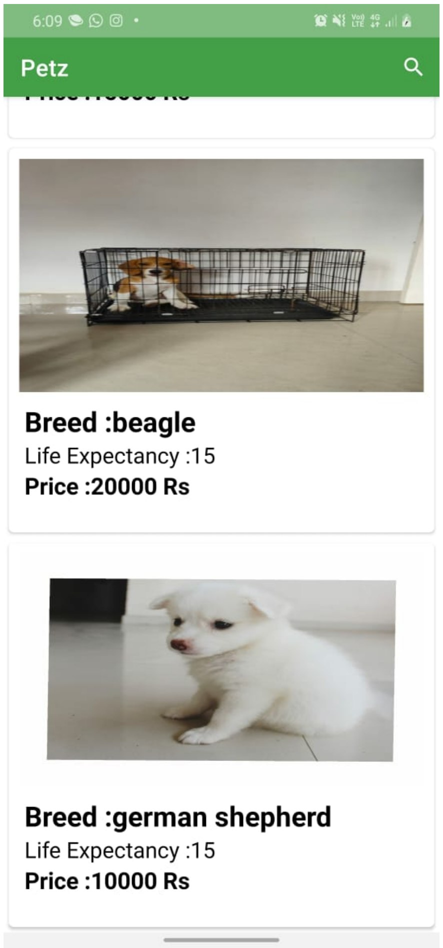





Registration
Name
Address
Username
Password
Confirm Password
Mobile Number
OTP
SEND OTP
Gender
○ Male
○ Female
○ Others
SIGN UP

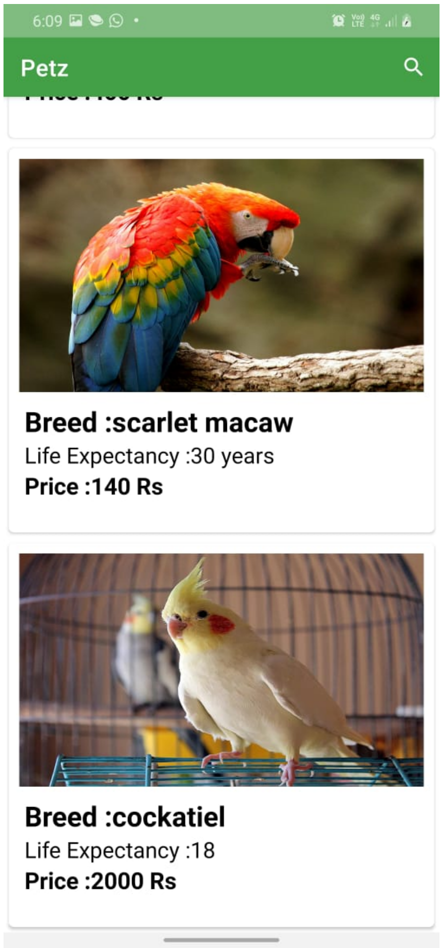

1:55
Forgot your password?
Mobile Number
SEND OTP





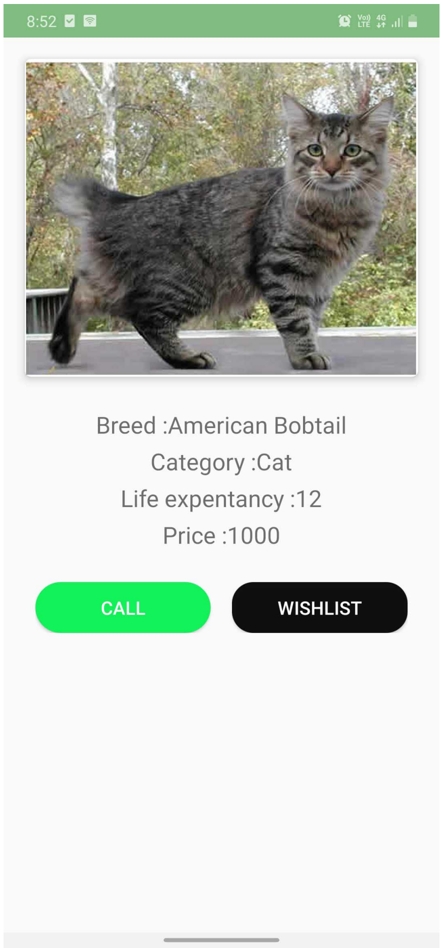
Breed :beagle
Life Expectancy :15
Price :20000 Rs

Breed :german shepherd
Life Expectancy :15
Price :10000 Rs

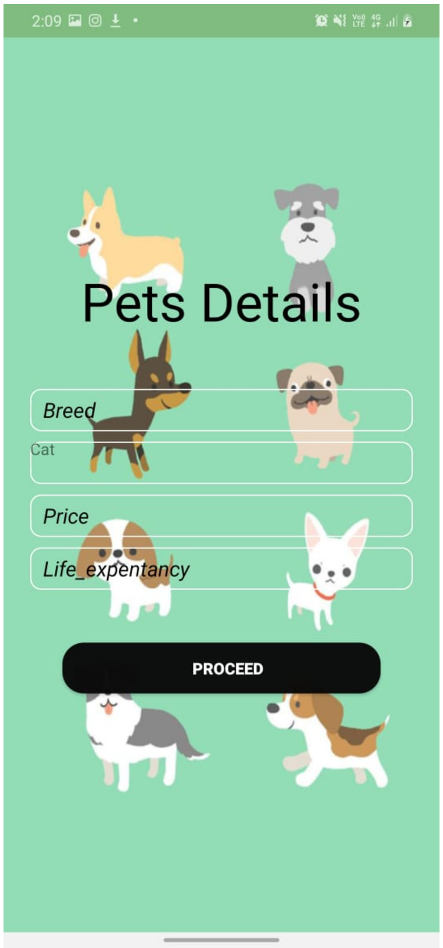


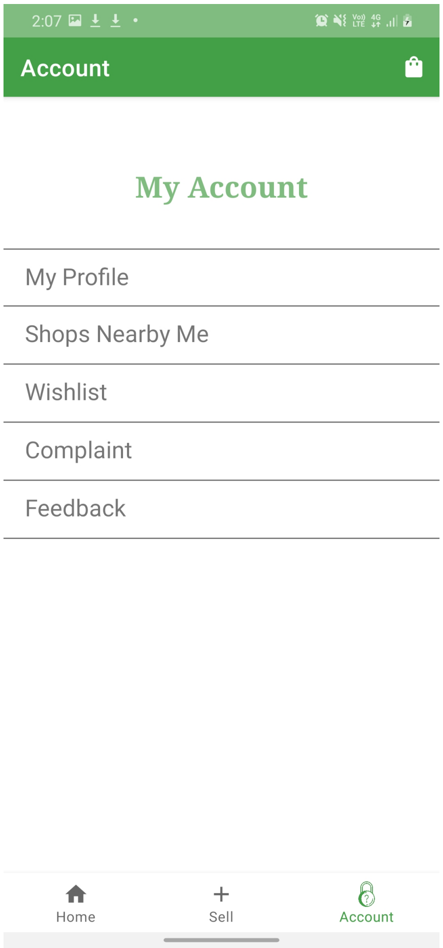
Breed :scarlet macaw
Life Expectancy :30 years
Price :140 Rs

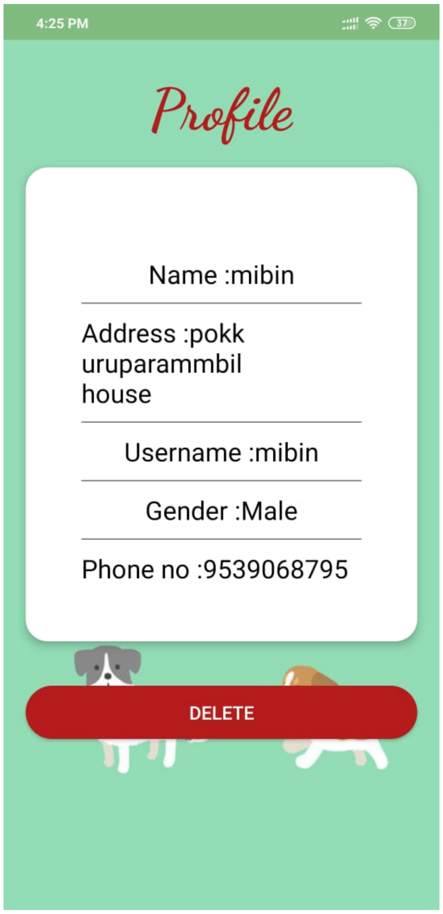
Breed :cockatiel
Life Expectancy :18
Price :2000 Rs



Breed :American Bobtail
Category :Cat
Life expentancy :12
Price :1000


2:09
Pets Details
Breed
Cat
Price
Life_expentancy
PROCEED


2:07
Account
My Account
My Profile
Shops Nearby Me
Wishlist
Complaint
Feedback
Home
Sell
Account



Profile

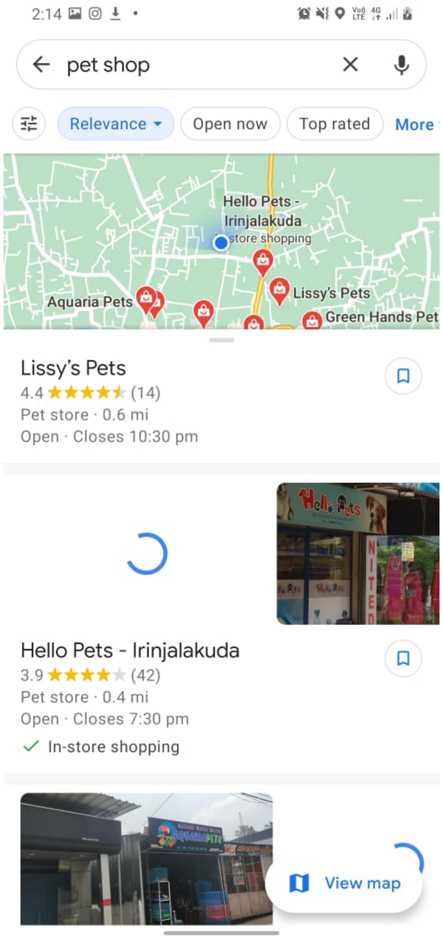
Name :mibin

Address :pokk uruparammbil house

Username :mibin

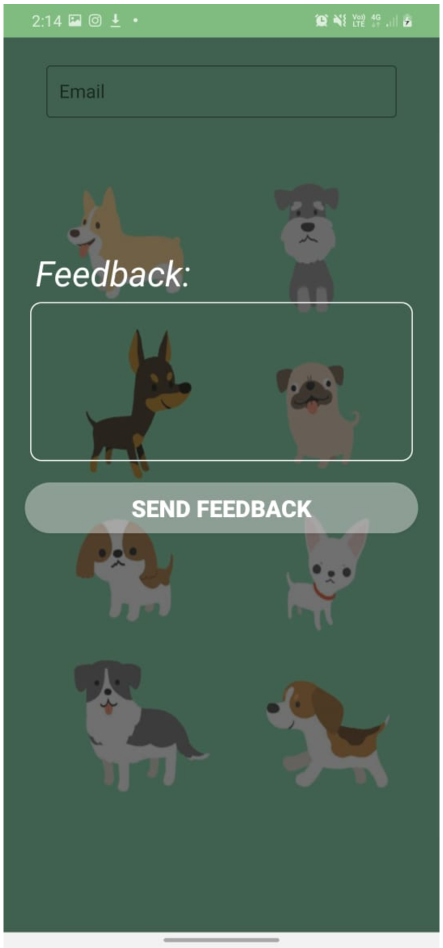
Gender :Male

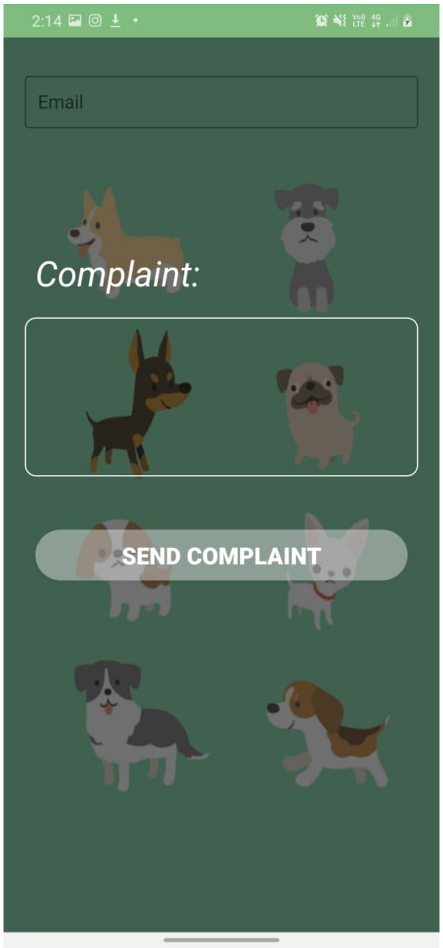
Phone no :9539068795

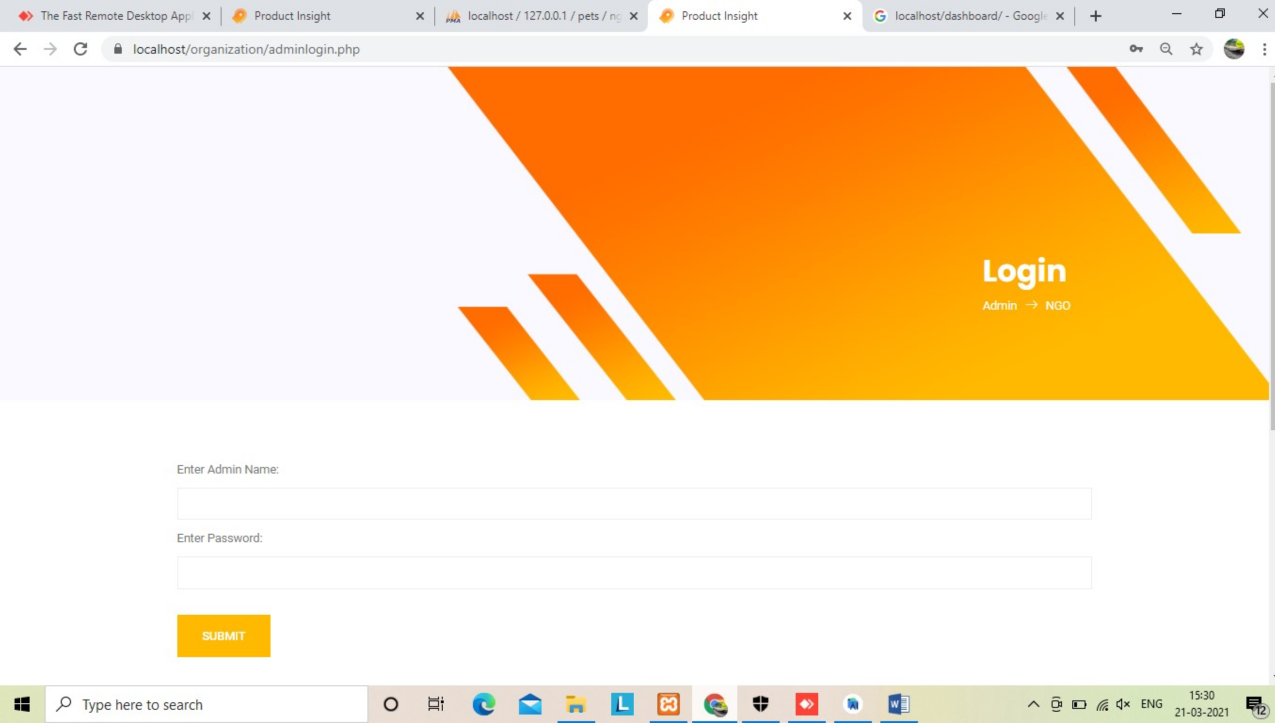


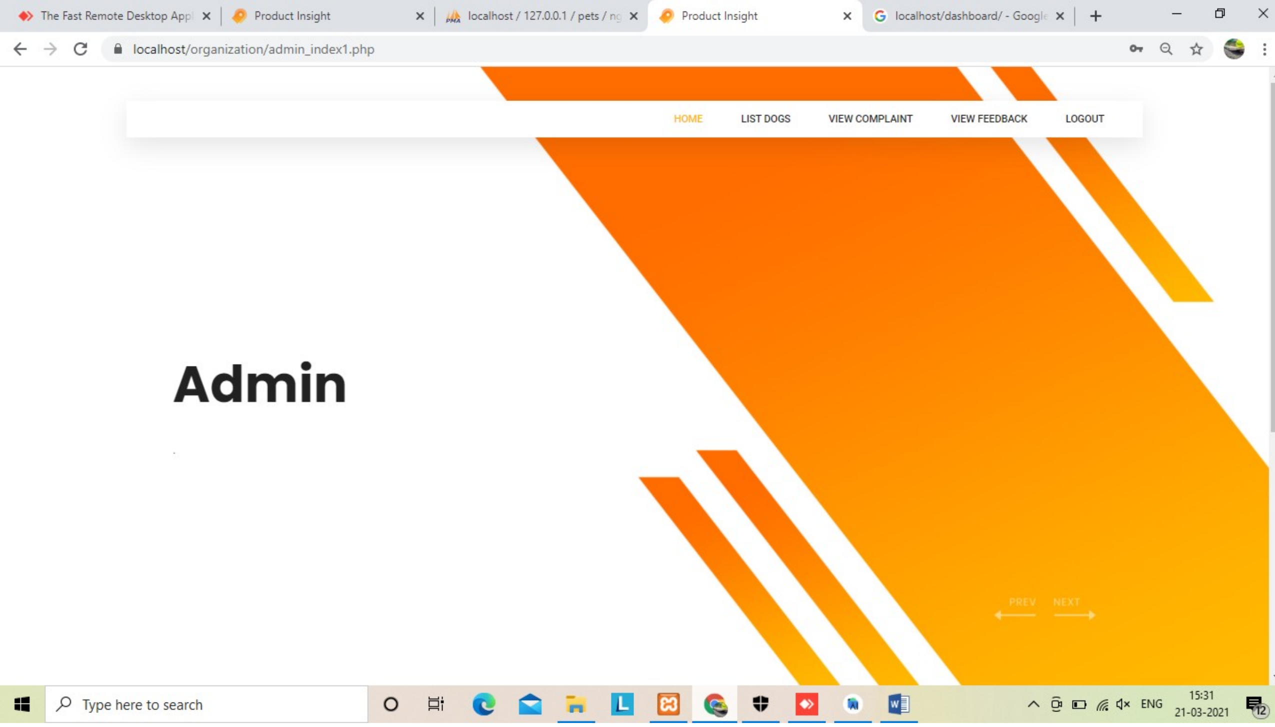
Lissy's Pets
4.4 ★★★★★ (14)
Pet store · 0.6 mi
Open · Closes 10:30 pm

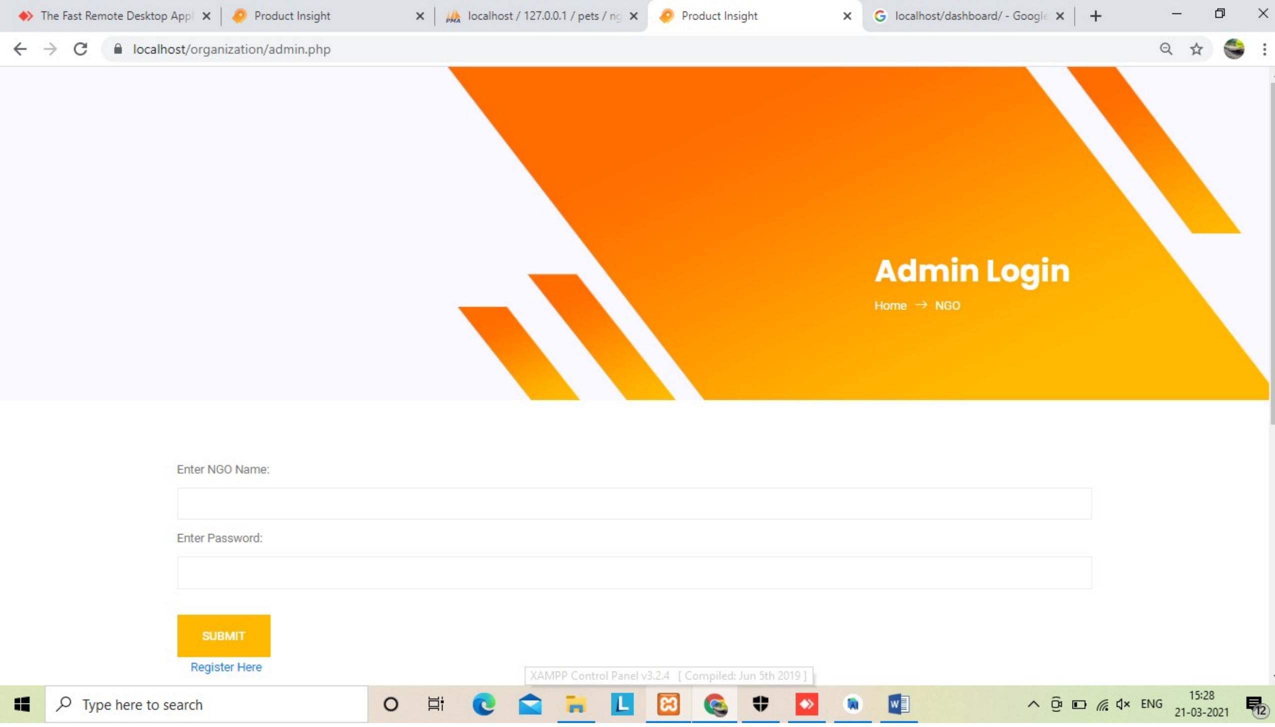
Hello Pets - Irinjalakuda
3.9 ★★★★☆ (42)
Pet store · 0.4 mi
Open · Closes 7:30 pm
✓ In-store shopping


2:14
Email
Feedback:
SEND FEEDBACK


2:14
Email
Complaint:
SEND COMPLAINT


The Fast Remote Desktop Appl
Product Insight
localhost / 127.0.0.1 / pets / ng
Product Insight
localhost/dashboard/ - Google
localhost/organization/adminlogin.php
Login
Admin → NGO
Enter Admin Name:
Enter Password:
SUBMIT
Type here to search
ENG
15:30
21-03-2021


The Fast Remote Desktop Appl
Product Insight
localhost / 127.0.0.1 / pets / ng
Product Insight
localhost/dashboard/ - Google
localhost/organization/admin_index1.php
HOME
LIST DOGS
VIEW COMPLAINT
VIEW FEEDBACK
LOGOUT
Admin
PREV
NEXT
Type here to search
ENG
15:31
21-03-2021



The Fast Remote Desktop Appl
Product Insight
localhost / 127.0.0.1 / pets / ng
Product Insight
localhost/dashboard/ - Google
localhost/organization/admin.php

Admin Login

Home → NGO

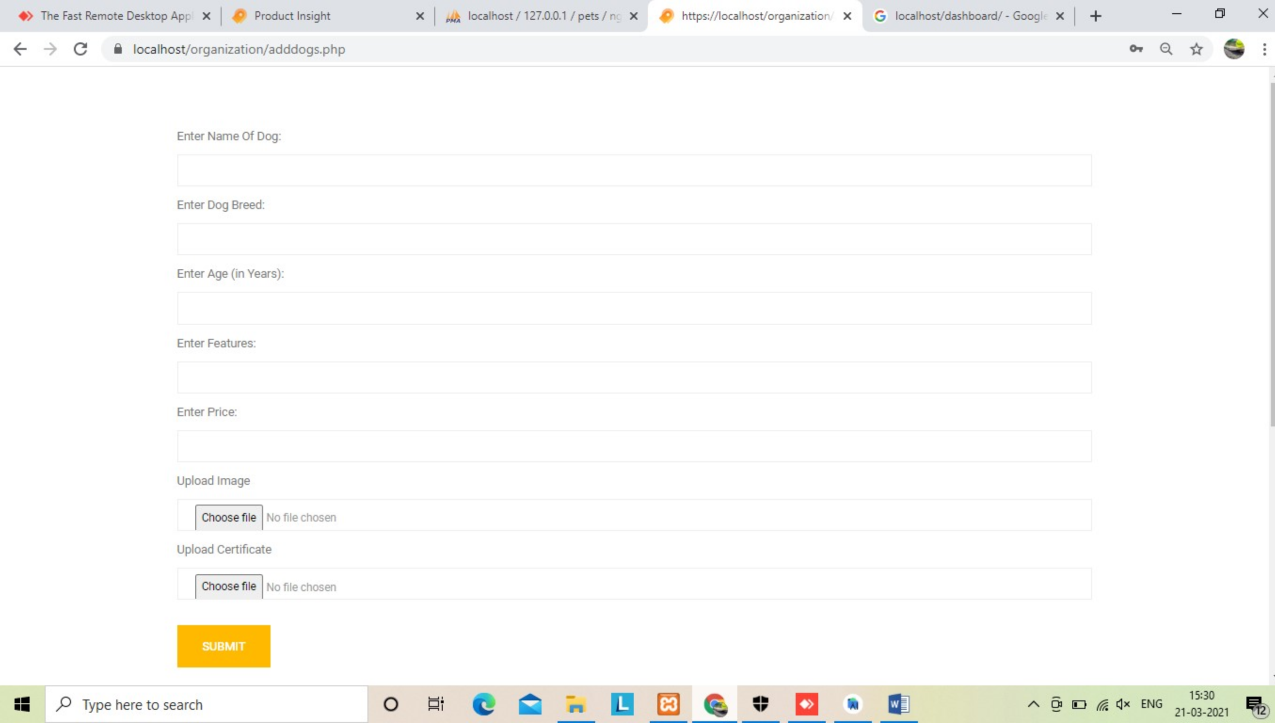
Enter NGO Name:

Enter Password:

SUBMIT

Register Here

XAMPP Control Panel v3.2.4 [Compiled: Jun 5th 2019]
Type here to search
ENG
15:28
21-03-2021



localhost/organization/adddogs.php

Enter Name Of Dog:

Enter Dog Breed:

Enter Age (in Years):

Enter Features:

Enter Price:

Upload Image

Choose file No file chosen

Upload Certificate

Choose file No file chosen

The Fast Remote Desktop Appl
Product Insight
localhost / 127.0.0.1 / pets / ng
https://localhost/organization/
localhost/dashboard/ - Google
SUBMIT
Type here to search
ENG
15:30
21-03-2021

D CODE

admin.php

```
[breaklines=true]
\begin{verbatim}
<body>
<header class="header_area sticky-header">
<div class="main_menu">
</div>
<div class="search_input" id="search_input_box">
<div class="container">
<form class="d-flex justify-content-between">
<input type="text" class="form-control" id="search_input" placeholder="Search Here">
<button type="submit" class="btn"></button>
<span class="lnr lnr-cross" id="close_search" title="Close Search"></span>
</form>
</div>
</div>
</header>
<section class="banner-area organic-breadcrumb">
<div class="container">
<div class="breadcrumb-banner d-flex flex-wrap align-items-center justify-content-end">
<div class="col-first">
<h1>Admin Login</h1>
<nav class="d-flex align-items-center">
<a href="index.html">Home<span class="lnr lnr-arrow-right"></span></a>
<a href="admin.php">Admin</a>
</nav>
</div>
</div>
</div>
</section>
<section class="contact_area section_gap_bottom">
<div class="container">
<br><br>
<br>
<div class="col-lg-12">
<form class="row contact_form" action="adminloginprocess.php" method="GET"
 id="contactForm" novalidate="novalidate">
<div class="col-md-12 text-left">
<div class="form-group">
Enter Username:
</div>
<div class="form-group">
<input type="text" class="form-control" id="name" name="name" required/>
```

```
</div>
<div class="form-group">
Enter Password:
</div>
<div class="form-group">
<input type="password" class="form-control" id="passwords"
 name="passwords"  required/>
</div>
<input type="submit" value="submit" class="primary-btn">
</div>
</form>
</div>
</div>
</div>
</section>
<footer class="footer-area section_gap">
<div class="container">
<div class="row">
<div class="col-lg-3  col-md-6 col-sm-6">
<div class="single-footer-widget">
<h6>About Us</h6>
<p>
Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod
 tempor incididunt ut labore dolore
magna aliqua.
</p>
</div>
</div>
<div class="col-lg-4  col-md-6 col-sm-6">
<div class="single-footer-widget">
<h6>Newsletter</h6>
<p>Stay update with our latest</p>
<div class="" id="mc_embed_signup">
<form target="_blank" novalidate="true"
 action="https://spondonit.us12.list-manage.com/subscribe/post?
u=1462626880ade1ac87bd9c93a&amp;id=92a4423d01"
method="get" class="form-inline">
<div class="d-flex flex-row">
<input class="form-control" name="EMAIL" placeholder="Enter Email"
onfocus="this.placeholder = ’’" onblur="this.placeholder = ’Enter Email ’"
required="" type="email">
<button class="click-btn btn btn-default"><i class="fa fa-long-arrow-right"
 aria-hidden="true"></i></button>
<div style="position: absolute; left: -5000px;">
<input name="b_36c4fd991d266f23781ded980_aefe40901a"
tabindex="-1" value="" type="text">
```

```
</div>
<button class="bb-btn btn"><span class="lnr lnr-arrow-right"></span></button>
</div>  -->
</div>
<div class="info"></div>
</form>
</div>
</div>
</div>
<div class="col-lg-3  col-md-6 col-sm-6">
<div class="single-footer-widget mail-chimp">
<h6 class="mb-20">Instragram Feed</h6>
<ul class="instafeed d-flex flex-wrap">
<li><img src="img/i1.jpg" alt=""></li>
<li><img src="img/i2.jpg" alt=""></li>
<li><img src="img/i3.jpg" alt=""></li>
<li><img src="img/i4.jpg" alt=""></li>
<li><img src="img/i5.jpg" alt=""></li>
<li><img src="img/i6.jpg" alt=""></li>
<li><img src="img/i7.jpg" alt=""></li>
<li><img src="img/i8.jpg" alt=""></li>
</ul>
</div>
</div>
<div class="col-lg-2 col-md-6 col-sm-6">
<div class="single-footer-widget">
<h6>Follow Us</h6>
<p>Let us be social</p>
<div class="footer-social d-flex align-items-center">
<a href="#"><i class="fa fa-facebook"></i></a>
<a href="#"><i class="fa fa-twitter"></i></a>
<a href="#"><i class="fa fa-dribbble"></i></a>
<a href="#"><i class="fa fa-behance"></i></a>
</div>
</div>
</div>
</div>
<div class="footer-bottom d-flex justify-content-center align-items-center flex-wrap">
<p class="footer-text m-0">
Copyright &copy;<script>document.write(new Date().getFullYear());
</script> All rights reserved | This template is made with
 <i class="fa fa-heart-o" aria-hidden="true"></i> by
 <a href="https://colorlib.com" target="_blank">Colorlib</a>
</p>
</div>
</div>
```

```
</footer>
<div id="success" class="modal modal-message fade" role="dialog">
<div class="modal-dialog">
<div class="modal-content">
<div class="modal-header">
<button type="button" class="close" data-dismiss="modal" aria-label="Close">
<i class="fa fa-close"></i>
</button>
<h2>Thank you</h2>
<p>Your message is successfully sent...</p>
</div>
</div>
</div>
</div>
<div id="error" class="modal modal-message fade" role="dialog">
<div class="modal-dialog">
<div class="modal-content">
<div class="modal-header">
<button type="button" class="close" data-dismiss="modal" aria-label="Close">
<i class="fa fa-close"></i>
</button>
<h2>Sorry !</h2>
<p> Something went wrong </p>
</div>
</div>
</div>
</div>
</body>
</html>
```

adminloginprocess.php

```
<?php
session_start();
$myusername=$_GET["name"];
$mypassword=$_GET['passwords'];
if($myusername=="admin"&&$mypassword=="admin")
{
$_SESSION['admin']=$myusername;
?>
<script language="javascript">alert('Login Success');
window.location.replace('admin_index.php');</script>
<?php
}
else {
echo "login failed";
}
?>
```

registration.php

```
<?php
$con = new mysqli("localhost", "root", "") or die("connection error");
$db = mysqli_select_db($con, "pets") or die("error in database");
$name= $_POST["name"];
$address = $_POST["address"];
$username = $_POST["username"];
$password = $_POST["password"];
$mobilenumber= $_POST["mobilenumber"];
$gender = $_POST["gender"];
$s="select * from registration where mobilenumber='$mobilenumber'";
$t=mysqli_query($con, $s) or die(mysqli_error());
if(mysqli_num_rows($t)>0)
{
$response['status'] = "0";
$response['message'] = "phone number already registered";
}
else
{
$q = "INSERT INTO registration VALUES
('', '$name', '$address', '$username', '$password', '$mobilenumber', '$gender')";
$result = mysqli_query($con, $q);
if ($result) {
$response['status'] = "1";
$response['message'] = "Registration successful";
}
else {
$response['status'] = "0";
$response['message'] = "Registration failed. Please try again!";
}
}
# Converting to JSON (JavaScript Object Notation) Format
echo json_encode($response);
?>
```

login.php

```
<?php
    $con = new mysqli("localhost", "root", "") or die("connection error");
    $db = mysqli_select_db($con, "pets") or die("error in database");
    $username = $_POST["username"];
    $password = $_POST["password"];
    $q = "SELECT * from registration WHERE
    username='$username' && password='$password' ";
    $result = mysqli_query($con, $q) or die("error in query");
    $row=mysqli_fetch_row($result);
    if (mysqli_num_rows($result) > 0) {
        $response['status'] = "1";
        $response['message'] = "Login successful";
        $response['id']=$row[0];
        $response['name']=$row[1];
        $response['address']=$row[2];
        $response['username']=$row[3];
        $response['mobile_no']=$row[5];
        $response['gender']=$row[6];
    }
    else {
        $response['status'] = "0";
        $response['message'] = "Incorrect username or password!";
        $response['id']="";
        $response['name']="";
        $response['address']="";
        $response['username']="";
        $response['mobile_no']="";
        $response['gender']="";
    }
     # Converting to JSON (JavaScript Object Notation) Format
    echo json_encode($response);

?>
```

ngo_reg.php

```
<body>
<header class="header_area sticky-header">
<div class="main_menu">
</div>
<div class="search_input" id="search_input_box">
<div class="container">
<form class="d-flex justify-content-between">
<input type="text" class="form-control" id="search_input" placeholder="Search Here">
<button type="submit" class="btn"></button>
<span class="lnr lnr-cross" id="close_search" title="Close Search"></span>
</form>
</div>
</div>
</header>
<section class="contact_area section_gap_bottom">
<div class="container">
<br><br>
<br>
<div class="col-lg-12">
<form class="row contact_form" action="registerprocess.php" method="post">
<div class="col-md-12 text-left">
<div class="form-group">
Enter Name Of NGO:
</div>
<div class="form-group">
<input type="text" class="form-control" id="ngoname" name="ngoname" required/>
</div>
<div class="form-group">
Enter Place:
</div>
<div class="form-group">
<input type="text" class="form-control" id="place" name="place" required/>
</div>
<div class="form-group">
<div class="form-group">
Enter Contact:
</div>
<div class="form-group">
<input type="text" class="form-control" id="contact" name="contact"  required/>
</div>
<div class="form-group">
```

```
Enter Password:
</div>
<div class="form-group">
<input type="text" class="form-control" id="password" name="password"  required/>
</div>
<input type="submit" value="submit" class="primary-btn">
</div>
</form>
</div>
</div>
</div>
</section>
<footer class="footer-area section_gap">
<div class="container">
<div class="row">
<div class="col-lg-3  col-md-6 col-sm-6">
<div class="single-footer-widget">
<h6>About Us</h6>
<p>
Lorem ipsum dolor sit amet, consectetur adipisicing elit,
 sed do eiusmod tempor incididunt ut labore dolore
magna aliqua.
</p>
</div>
</div>
<div class="col-lg-4  col-md-6 col-sm-6">
<div class="single-footer-widget">
<h6>Newsletter</h6>
<p>Stay update with our latest</p>
<div class="" id="mc_embed_signup">
<form target="_blank" novalidate="true"
action="https://spondonit.us12.list-manage.com/subscribe/post?
u=1462626880ade1ac87bd9c93a&amp;id=92a4423d01"
method="get" class="form-inline">
<div class="d-flex flex-row">
<input class="form-control" name="EMAIL"
placeholder="Enter Email" onfocus="this.placeholder = ’’"
 onblur="this.placeholder = ’Enter Email ’"
required="" type="email">
<button class="click-btn btn btn-default">
<i class="fa fa-long-arrow-right" aria-hidden="true"></i></button>
<div style="position: absolute; left: -5000px;">
<input name="b_36c4fd991d266f23781ded980_aefe40901a"
 tabindex="-1" value="" type="text">
</div>
<!-- <div class="col-lg-4 col-md-4">
```

```
<button class="bb-btn btn"><span class="lnr lnr-arrow-right">
</span></button>
</div>  -->
</div>
<div class="info"></div>
</form>
</div>
</div>
</div>
<div class="col-lg-3  col-md-6 col-sm-6">
<div class="single-footer-widget mail-chimp">
<h6 class="mb-20">Instragram Feed</h6>
<ul class="instafeed d-flex flex-wrap">
<li><img src="img/i1.jpg" alt=""></li>
<li><img src="img/i2.jpg" alt=""></li>
<li><img src="img/i3.jpg" alt=""></li>
<li><img src="img/i4.jpg" alt=""></li>
<li><img src="img/i5.jpg" alt=""></li>
<li><img src="img/i6.jpg" alt=""></li>
<li><img src="img/i7.jpg" alt=""></li>
<li><img src="img/i8.jpg" alt=""></li>
</ul>
</div>
</div>
<div class="col-lg-2 col-md-6 col-sm-6">
<div class="single-footer-widget">
<h6>Follow Us</h6>
<p>Let us be social</p>
<div class="footer-social d-flex align-items-center">
<a href="#"><i class="fa fa-facebook"></i></a>
<a href="#"><i class="fa fa-twitter"></i></a>
<a href="#"><i class="fa fa-dribbble"></i></a>
<a href="#"><i class="fa fa-behance"></i></a>
</div>
</div>
</div>
</div>
<div class="footer-bottom d-flex
justify-content-center align-items-center flex-wrap">
<p class="footer-text m-0"><!-- Link back to Colorlib can't be removed.
 Template is licensed under CC BY 3.0. -->
Copyright &copy;<script>document.write(new Date().getFullYear())
;</script> All rights reserved | This template is made with
 <i class="fa fa-heart-o" aria-hidden="true"></i> by
 <a href="https://colorlib.com" target="_blank">Colorlib</a>
<!-- Link back to Colorlib can't be removed.
```

```
 Template is licensed under CC BY 3.0. -->
</p>
</div>
</div>
</footer>
<div id="success" class="modal modal-message fade" role="dialog">
<div class="modal-dialog">
<div class="modal-content">
<div class="modal-header">
<button type="button" class="close" data-dismiss="modal" aria-label="Close">
<i class="fa fa-close"></i>
</button>
<h2>Thank you</h2>
<p>Your message is successfully sent...</p>
</div>
</div>
</div>
</div>
<div id="error" class="modal modal-message fade" role="dialog">
<div class="modal-dialog">
<div class="modal-content">
<div class="modal-header">
<button type="button" class="close" data-dismiss="modal" aria-label="Close">
<i class="fa fa-close"></i>
</button>
<h2>Sorry !</h2>
<p> Something went wrong </p>
</div>
</div>
</div>
</div>
<script src="js/vendor/jquery-2.2.4.min.js"></script>
<script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/
1.11.0/umd/popper.min.js" integrity="sha384-b/U6ypiBEHpOf/4+1nzFpr53nxSS+
GLCkfwBdFNTxtclqqenISfwAzpKaMNFNmj4"
crossorigin="anonymous"></script>
<script src="js/vendor/bootstrap.min.js"></script>
<script src="js/jquery.ajaxchimp.min.js"></script>
<script src="js/jquery.nice-select.min.js"></script>
<script src="js/jquery.sticky.js"></script>
<script src="js/nouislider.min.js"></script>
<script src="js/jquery.magnific-popup.min.js"></script>
<script src="js/owl.carousel.min.js"></script>
<!--gmaps Js-->
<script src="https://maps.googleapis.com/maps/api/js?
key=AIzaSyCjCGmQ0Uq4exrzdcL6rvxywDD0vfAu6eE"></script>
```

```
<script src="js/gmaps.min.js"></script>
<script src="js/main.js"></script>
</body>

</html>
```

Add Streetdogs.html

```
</head>
<body>
<header class="header_area sticky-header">
<div class="main_menu">
</div>
<div class="search_input" id="search_input_box">
<div class="container">
<form class="d-flex justify-content-between">
<input type="text" class="form-control" id="search_input"
placeholder="Search Here">
<button type="submit" class="btn"></button>
<span class="lnr lnr-cross" id="close_search" title="Close Search"></span>
</form>
</div>
</div>
</header>
<section class="contact_area section_gap_bottom">
<div class="container">
<br><br>
<br>
<div class="col-lg-12">
<form class="row contact_form" action="adddogprocess.php"
method="post" enctype="multipart/form-data"
id="contactForm" novalidate="novalidate">
<div class="col-md-12 text-left">
<div class="form-group">
Enter Name Of Dog:
</div>
<div class="form-group">
<input type="text" class="form-control" id="dogname"
 name="dogname" required/>
</div>
<div class="form-group">
Enter Dog Breed:
</div>
<div class="form-group">
<input type="text" class="form-control" id="breedname"
name="breedname" required/>
</div>
<div class="form-group">
Enter Age (in Years):
</div>
```

```
<div class="form-group">
<input type="text" class="form-control" id="age"
name="age" required/>
</div>
<div class="form-group">
Enter Features:
</div>
<div class="form-group">
<input type="text" class="form-control" id="features"
 name="features"  required/>
</div>
<div class="form-group">
Enter Price:
</div>
<div class="form-group">
<input type="text" class="form-control" id="price"
name="price"  required/>
</div>
<div class="form-group">
Upload Image
</div>
<div class="form-group">
<input type="file" class="form-control" id="image"
 name="image"  required/>
</div>
<div class="form-group">
Upload Certificate
</div>
<div class="form-group">
<input type="file" class="form-control" id="cert"
name="cert"  required/>
</div>
<input type="submit" value="submit" class="primary-btn">
</div>
</form>
</div>
</div>
</div>
</section>
<footer class="footer-area section_gap">
<div class="container">
<div class="row">
<div class="col-lg-3  col-md-6 col-sm-6">
<div class="single-footer-widget">
<h6>About Us</h6>
<p>
```

```
Lorem ipsum dolor sit amet, consectetur adipisicing elit,
 sed do eiusmod tempor incididunt ut labore dolore
magna aliqua.
</p>
</div>
</div>
<div class="col-lg-4  col-md-6 col-sm-6">
<div class="single-footer-widget">
<h6>Newsletter</h6>
<p>Stay update with our latest</p>
<div class="" id="mc_embed_signup">
<form target="_blank" novalidate="true"
action="https://spondonit.us12.list-manage.com/subscribe/post
?u=1462626880ade1ac87bd9c93a&amp;id=92a4423d01"
method="get" class="form-inline">
<div class="d-flex flex-row">
<input class="form-control" name="EMAIL" placeholder="Enter Email"
onfocus="this.placeholder = ''" onblur="this.placeholder = 'Enter Email '"
required="" type="email">
<button class="click-btn btn btn-default">
<i class="fa fa-long-arrow-right" aria-hidden="true"></i></button>
<div style="position: absolute; left: -5000px;">
<input name="b_36c4fd991d266f23781ded980_aefe40901a"
 tabindex="-1" value="" type="text">
</div>
<!-- <div class="col-lg-4 col-md-4">
<button class="bb-btn btn"><span class="lnr lnr-arrow-right">
</span></button>
</div>  -->
</div>
<div class="info"></div>
</form>
</div>
</div>
</div>
<div class="col-lg-3  col-md-6 col-sm-6">
<div class="single-footer-widget mail-chimp">
<h6 class="mb-20">Instragram Feed</h6>
<ul class="instafeed d-flex flex-wrap">
<li><img src="img/i1.jpg" alt=""></li>
<li><img src="img/i2.jpg" alt=""></li>
<li><img src="img/i3.jpg" alt=""></li>
<li><img src="img/i4.jpg" alt=""></li>
<li><img src="img/i5.jpg" alt=""></li>
<li><img src="img/i6.jpg" alt=""></li>
<li><img src="img/i7.jpg" alt=""></li>
```

```
<li><img src="img/i8.jpg" alt=""></li>
</ul>
</div>
</div>
<div class="col-lg-2 col-md-6 col-sm-6">
<div class="single-footer-widget">
<h6>Follow Us</h6>
<p>Let us be social</p>
<div class="footer-social d-flex align-items-center">
<a href="#"><i class="fa fa-facebook"></i></a>
<a href="#"><i class="fa fa-twitter"></i></a>
<a href="#"><i class="fa fa-dribbble"></i></a>
<a href="#"><i class="fa fa-behance"></i></a>
</div>
</div>
</div>
</div>
<div class="footer-bottom d-flex justify-content-center
align-items-center flex-wrap">
<p class="footer-text m-0">
Copyright &copy;<script>document.write(new Date().getFullYear());</script>
 All rights reserved | This template is made with <i class="fa fa-heart-o"
aria-hidden="true">
</i> by <a href="https://colorlib.com" target="_blank">Colorlib</a>
</p>
</div>
</div>
</footer>
<div id="success" class="modal modal-message fade" role="dialog">
<div class="modal-dialog">
<div class="modal-content">
<div class="modal-header">
<button type="button" class="close" data-dismiss="modal" aria-label="Close">
<i class="fa fa-close"></i>
</button>
<h2>Thank you</h2>
<p>Your message is successfully sent...</p>
</div>
</div>
</div>
</div>
<div id="error" class="modal modal-message fade" role="dialog">
<div class="modal-dialog">
<div class="modal-content">
<div class="modal-header">
<button type="button" class="close" data-dismiss="modal" aria-label="Close">
```

```
<i class="fa fa-close"></i>
</button>
<h2>Sorry !</h2>
<p> Something went wrong </p>
</div>
</div>
</div>
</div>
</body>
</html>
```

Birds List.php

```
<?php
define('DB_HOST', 'localhost');
define('DB_USER', 'root');
define('DB_PASS', '');
define('DB_NAME', 'pets');
$conn = new mysqli(DB_HOST, DB_USER, DB_PASS, DB_NAME);
if (mysqli_connect_errno()) {
echo "Failed to connect to MySQL: " . mysqli_connect_error();
die();
}
$stmt = $conn->prepare("SELECT pe_id,item_category,item_breed,life_expentancy,
item_price,item_image,contact FROM pets_uploads WHERE item_category ='Bird';");
$stmt->execute();
$stmt->bind_result($id,$cat,$breed,$le,$price,$img,$contact);
$products = array();
while($stmt->fetch()){
$temp = array();
$temp['pe_id'] = $id;
$temp['item_category'] = $cat;
$temp['item_breed'] = $breed;
$temp['life_expentancy'] = $le;
$temp['item_price']= $price;
$temp['item_image'] = $img;
$temp['contact'] = $contact;
array_push($products, $temp);
}
echo json_encode($products);
?>
```

Cats List.php

```
<?php
define('DB_HOST', 'localhost');
define('DB_USER', 'root');
define('DB_PASS', '');
define('DB_NAME', 'pets');
$conn = new mysqli(DB_HOST, DB_USER, DB_PASS, DB_NAME);
if (mysqli_connect_errno()) {
echo "Failed to connect to MySQL: " . mysqli_connect_error();
die();
}
$stmt = $conn->prepare("SELECT pe_id,item_category,item_breed,life_expentancy,
item_price,item_image,contact FROM pets_uploads WHERE item_category ='Cat';");
$stmt->execute();
$stmt->bind_result($id,$cat,$breed,$le,$price,$img,$contact);
$products = array();
while($stmt->fetch()){
$temp = array();
$temp['pe_id'] = $id;
$temp['item_category'] = $cat;
$temp['item_breed'] = $breed;
$temp['life_expentancy'] = $le;
$temp['item_price']= $price;
$temp['item_image'] = $img;
$temp['contact'] = $contact;
array_push($products, $temp);
}
echo json_encode($products);
?>
```

Rabbits List.php

```
<?php
define('DB_HOST', 'localhost');
define('DB_USER', 'root');
define('DB_PASS', '');
define('DB_NAME', 'pets');
$conn = new mysqli(DB_HOST, DB_USER, DB_PASS, DB_NAME);
if (mysqli_connect_errno()) {
echo "Failed to connect to MySQL: " . mysqli_connect_error();
die();
}
$stmt = $conn->prepare("SELECT pe_id,item_category,item_breed,life_expentancy
,item_price,item_image,contact FROM pets_uploads WHERE item_category ='Rabit';");
//executing the query
$stmt->execute();
$stmt->bind_result($id,$cat,$breed,$le,$price,$img,$contact);
$products = array();
//traversing through all the result
while($stmt->fetch()){
$temp = array();
$temp['pe_id'] = $id;
$temp['item_category'] = $cat;
$temp['item_breed'] = $breed;
$temp['life_expentancy'] = $le;
$temp['item_price']= $price;
$temp['item_image'] = $img;
$temp['contact'] = $contact;
array_push($products, $temp);
}
echo json_encode($products);
?>
```

Dog List.php

```
<?php
define('DB_HOST', 'localhost');
define('DB_USER', 'root');
define('DB_PASS', '');
define('DB_NAME', 'pets');
$conn = new mysqli(DB_HOST, DB_USER, DB_PASS, DB_NAME);
if (mysqli_connect_errno()) {
echo "Failed to connect to MySQL: " . mysqli_connect_error();
die();
}$stmt = $conn->prepare("SELECT pe_id,item_category,item_breed,life_expentancy,
item_price,item_image,contact FROM pets_uploads WHERE item_category ='Dog';");
$stmt->execute();
$stmt->bind_result($id,$cat,$breed,$le,$price,$img,$contact)
$products = array();
while($stmt->fetch()){
$temp = array();
$temp['pe_id'] = $id;
$temp['item_category'] = $cat;
$temp['item_breed'] = $breed;
$temp['life_expentancy'] = $le;
$temp['item_price']= $price;
$temp['item_image'] = $img;
$temp['contact'] = $contact;
array_push($products, $temp);
}
echo json_encode($products);
?>
```

Street dog list.php

```
<?php
//database constants
define('DB_HOST', 'localhost');
define('DB_USER', 'root');
define('DB_PASS', '');
define('DB_NAME', 'pets');
$conn = new mysqli(DB_HOST, DB_USER, DB_PASS, DB_NAME);
if (mysqli_connect_errno()) {
echo "Failed to connect to MySQL: " . mysqli_connect_error();
die();
}$stmt = $conn->prepare("SELECT id,dogname,breed,age,features,
price,dogimage,certificate,contact FROM streetdog");
$stmt->execute();
$stmt->bind_result($id,$dogname,$breed,$age,$features,
$price,$dogimage,$certificate,$contact);
$products = array();
while($stmt->fetch()){
$temp = array();
$temp['id'] = $id;
$temp['dogname'] = $dogname;
$temp['breed'] = $breed;
$temp['age'] = $age;
$temp['features']= $features;
$temp['price'] = $price;
$temp['dogimage'] = $dogimage;
$temp['certificate'] = $certificate;
$temp['contact'] = $contact;
array_push($products, $temp);
}
echo json_encode($products);
?>
```

Upload Pets.php

```
<?php
 if($_SERVER['REQUEST_METHOD']=='POST'){
 echo $_SERVER["DOCUMENT_ROOT"];
 include_once("config.php");
$con = new mysqli("localhost", "root", "");
$db = mysqli_select_db($con, "pets");
$_FILES['image']['name']   give original name from parameter
where 'image' == parametername eg. city.jpg
$_FILES['image']['tmp_name']  temporary system generated name
$originalImgName= $_FILES['filename']['name'];
$tempName= $_FILES['filename']['tmp_name'];
$folder="upload/";
$folder = "http://192.168.100.66/organization/upload/".$originalImgName;
$item_category= $_POST["item_category"];
$item_breed = $_POST["item_breed"];
$life_expendancy = $_POST["life_expendancy"];
$item_price = $_POST["item_price"];
$user_id = $_POST["user_id"];
$contact = $_POST["contact"];
if(move_uploaded_file($tempName,$folder.$originalImgName)){
$query = "INSERT INTO pets_uploads(item_category,item_breed,
life_expentancy,item_price,item_image,user_id,contact) VALUES
 ('$item_category','$item_breed','$life_expendancy','$item_price',
'$originalImgName','$user_id','$contact')";
if(mysqli_query($con,$query))
{
$response['status'] = "1";
$response['message'] = "Uploading successful";
$query= "SELECT * FROM upload_video WHERE path='$url'";
$result= mysqli_query($con, $query);
$emparray = array();
   if(mysqli_num_rows($result) > 0){
 while ($row = mysqli_fetch_assoc($result)) {
 $emparray[] = $row;
  }
 echo json_encode(array( "status" => "true",
"message" => "Successfully file added!" , "data" => $emparray) );
}
else{
echo json_encode(array( "status" => "false","message" => "Failed!") );
  }
}
else{
```

```
echo json_encode(array( "status" => "false","message" => "Failed!") );
$response['status'] = "0";
$response['message'] = "Uploading failed. Please try again!";
}
echo "moved to ".$url;
}
else{
$response['status'] = "0";
$response['message'] = "Uploading failed. Please try again!";
echo json_encode(array( "status" => "false","message" => "Failed!") );
}
echo json_encode($response);
 }
?>
```

Login.java

```
public class Login extends AppCompatActivity {
    Button signUp, signIn;
    EditText etUsername, etPassword;
    String username, password;
    TextView textForgot;
    String url = Config.BaseURL+"login.php";
    String status, error,id,name,mobile_no,address,gender;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        getWindow().setFlags(WindowManager.LayoutParams.FLAG_FULLSCREEN,
        WindowManager.LayoutParams.FLAG_FULLSCREEN);
        setContentView(R.layout.login_activity);

        signUp = findViewById(R.id.signup_button);
        signIn = findViewById(R.id.login_button);
        etUsername = findViewById(R.id.Username);
        etPassword = findViewById(R.id.password);
        textForgot = findViewById(R.id.textForgot);

         textForgot.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                startActivity(new Intent(getApplicationContext(),Forgotpswrd.class));
            }
        });
        signUp.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Intent intent = new Intent(Login.this, registration.class);
                startActivity(intent);
            }
        });
       signIn.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Login();
            }
        });
        }


    private void Login() {
```

```
username = etUsername.getText().toString();
password = etPassword.getText().toString();

if (TextUtils.isEmpty(username)) {
    etUsername.setError("Please enter username");
    etUsername.requestFocus();
    return;
}
if (TextUtils.isEmpty(password)) {
    etPassword.setError("Please enter password");
    etPassword.requestFocus();
    return;
}

StringRequest request = new StringRequest(Request.Method.POST,
url, new Response.Listener<String>() {
    @Override
    public void onResponse(String response) {
        try {
            JSONObject s = new JSONObject(response);
            status = s.getString("status");
            error = s.getString("message");
            id = s.getString("id");
            address = s.getString("address");
            mobile_no = s.getString("mobile_no");
            gender = s.getString("gender");
            name = s.getString("name");
        } catch (JSONException e) {
            e.printStackTrace();
        }
        if (status.equals("0")) {
            Toast.makeText(Login.this, error, Toast.LENGTH_SHORT).show();
        } else {
        Toast.makeText(Login.this, "Login successful", Toast.LENGTH_SHORT).show();

            new SessionManager(Login.this).createLoginSession(id,username,
            name,address,gender,mobile_no);

            Intent intent = new Intent(Login.this, HomeActivity.class);
            intent.putExtra("id",id);
            intent.putExtra("name",name);
            intent.putExtra("address",address);
            intent.putExtra("gender",gender);
            intent.putExtra("username",username);
            intent.putExtra("mobile_no",mobile_no);
            startActivity(intent);
```

```
                    finish();
                }
            }
        }, new Response.ErrorListener() {
            @Override
            public void onErrorResponse(VolleyError error) {
                Toast.makeText(Login.this, error.toString(), Toast.LENGTH_SHORT).show();
                }
        }) {
            @Override
            protected Map<String, String> getParams() throws AuthFailureError {
                Map<String, String> m = new HashMap<>();

                m.put("username", username);
                m.put("password", password);
                return m;
             }
        };

        RequestQueue queue = Volley.newRequestQueue(this);
        queue.add(request);

    }
}
```

Registration.java

```
public class registration extends AppCompatActivity {

    EditText etName,etAddress,etUsername,etPassword,etCpassword,etMobnumber,etOtp;
    Button btnRegister,btnOtp;
    RadioGroup rgGender;
    RadioButton rdbtn;
    String name,address,username,password,confirmpassword,mobilenumber,otp,gender;
    int id;

    String url=Config.BaseURL+"registration.php";
    String status,error;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        getWindow().setFlags(WindowManager.LayoutParams.FLAG_FULLSCREEN
        ,WindowManager.LayoutParams.FLAG_FULLSCREEN);
        setContentView(R.layout.activity_registration);

        etName=findViewById(R.id.regname);
        etAddress=findViewById(R.id.adress);
        etUsername=findViewById(R.id.name);
        etPassword=findViewById(R.id.pass);
        etCpassword=findViewById(R.id.confirmpassword);
        etMobnumber=findViewById(R.id.mobile);
        etOtp=findViewById(R.id.otp);
        btnRegister=findViewById(R.id.login_button);
        rgGender=findViewById(R.id.gender);
        btnOtp=findViewById(R.id.send);

        btnOtp.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                checkPermission();
            }
        });

        btnRegister.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                registration();

            }
        });
```

```
}
private void registration()

{
    name = etName.getText().toString();
    address = etAddress.getText().toString();
    username = etUsername.getText().toString();
    password = etPassword.getText().toString();
    confirmpassword = etCpassword.getText().toString();
    mobilenumber = etMobnumber.getText().toString();
    otp = etOtp.getText().toString();


    if (TextUtils.isEmpty(name))
    {
        etName.setError("Please enter name");
        etName.requestFocus();
        return;
    }
    if (TextUtils.isEmpty(address))
    {
        etAddress.setError("Please enter address");
        etAddress.requestFocus();
        return;
    }
    if (TextUtils.isEmpty(username))
    {
        etUsername.setError("Please enter username");
        etUsername.requestFocus();
        return;
    }
    if (TextUtils.isEmpty(password))
    {
        etPassword.setError("Please enter password");
        etPassword.requestFocus();
        return;
    }
    if (TextUtils.isEmpty(confirmpassword))
    {
        etCpassword.setError("Please enter confirmpassword");
        etCpassword.requestFocus();
        return;
    }
    if (TextUtils.isEmpty(password)) {
        etMobnumber.setError("Please enter mobilenumber");
        etMobnumber.requestFocus();
        return;
    }
    if (TextUtils.isEmpty(otp))
    {
        etOtp.setError("Please enter OTP");
        etOtp.requestFocus();
        return;
    }

    id = rgGender.getCheckedRadioButtonId();
    rdbtn = findViewById(id);
    gender = rdbtn.getText().toString();


    StringRequest request=new StringRequest(Request.Method.POST, url,
    new Response.Listener<String>() {
        @Override
        public void onResponse(String response) {
            try {
                JSONObject s = new JSONObject(response);
                status = s.getString("status");
                error = s.getString("message");
            } catch (JSONException e) {
                e.printStackTrace();
            }
            if (status.equals("0")) {
                Toast.makeText(registration.this, error, Toast.LENGTH_SHORT).show();
            } else {
                Toast.makeText(registration.this, "Registration successful",
                Toast.LENGTH_SHORT).show();
                Intent intent = new Intent(registration.this,Login.class);
                startActivity(intent);
                finish();
            }
        }
    }, new Response.ErrorListener() {
        @Override
        public void onErrorResponse(VolleyError error) {
            Toast.makeText(registration.this, error.toString(),
            Toast.LENGTH_SHORT).show();
        }
    })
    {
        @Override
        protected Map<String, String> getParams() throws AuthFailureError {
            Map<String,String> m=new HashMap<>();
```

```
                m.put("name",name);
                m.put("address",address);
                m.put("username",username);
                m.put("password",password);
                m.put("mobilenumber",mobilenumber);
                m.put("gender",gender);

                return m;
            }
        };


        RequestQueue q= Volley.newRequestQueue(this);
        q.add(request);

    }


    public void sendOTP() {
        String num = etMobnumber.getText().toString();
        if(TextUtils.isEmpty(num))
        {
            etMobnumber.setError("Please enter mobilenumber");
            etMobnumber.requestFocus();
            return;
        }
        Random r = new Random();
        int otp = r.nextInt((9999 - 1000) + 1) + 1000;

        String msg = "Welcome to pets care system.
        Your OTP for registration is " + otp;

        SmsManager sms = SmsManager.getDefault();
        sms.sendTextMessage(num, null, msg, null, null);

    }

 // Function to check and request permission.
    public void checkPermission()
    {
        if (ContextCompat.checkSelfPermission(registration.this,
        Manifest.permission.SEND_SMS) == PackageManager.PERMISSION_DENIED) {

            // Requesting the permission
            ActivityCompat.requestPermissions(registration.this, new String[]
           { Manifest.permission.SEND_SMS }, 1);
        }
        else {
            sendOTP();
        }
    }

    @Override
    public void onRequestPermissionsResult(int requestCode,
    @NonNull String[] permissions, @NonNull int[] grantResults)
    {
        super.onRequestPermissionsResult(requestCode, permissions, grantResults);

        if (requestCode == 1) {
            if (grantResults.length > 0 && grantResults[0] ==
                   PackageManager.PERMISSION_GRANTED) {
                sendOTP();
            }
            else {
                Toast.makeText(registration.this, "SMS Permission Denied",
                Toast.LENGTH_SHORT).show();
            }
        }
    }
```

ProfileActivity.java

```
public class ProfileActivity extends AppCompatActivity {
    TextView profileName,profileAddress,proUsername,profileGender,profileNumber;
    Button btnDelete;

    String mobile_no,name,username,address,gender,statusID,error,id;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_profile);

        Intent intent=getIntent();
        id=intent.getStringExtra("id");
        mobile_no=intent.getStringExtra("mobile_no");
        name=intent.getStringExtra("name");
        username=intent.getStringExtra("username");
        address=intent.getStringExtra("address");
        gender=intent.getStringExtra("gender");

        profileName = findViewById(R.id.profileName);
        profileAddress = findViewById(R.id.profileAddress);
        proUsername = findViewById(R.id.proUsername);
        profileGender = findViewById(R.id.profileGender);
        profileNumber = findViewById(R.id.profileNumber);
        btnDelete = findViewById(R.id.btnDelete);

        profileName.setText("Name :"+name);
        profileAddress.setText("Address :"+address);
        proUsername.setText("Username :"+username);
        profileGender.setText("Gender :"+gender);
        profileNumber.setText("Phone no :"+mobile_no);

        btnDelete.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                deleteUser();
            }
        });

    }
    private  void deleteUser(){

        String url = Config.BaseURL + "user_delete.php";
        StringRequest sr=new StringRequest(Request.Method.POST, url,
                new Response.Listener<String>() {
                    @Override
                    public void onResponse(String response) {
                        JSONObject c;
                        try {
                            c = new JSONObject(response);
                            statusID = c.getString("StatusID");
                            error = c.getString("Error");
                        } catch (JSONException e) {
                            e.printStackTrace();
                        }
                        if (statusID.equals("0")) {
                            Toast.makeText(getApplicationContext(), error,
                            Toast.LENGTH_SHORT).show();
                        } else {
                            Toast.makeText(getApplicationContext(), "deleted successfully",
                             Toast.LENGTH_SHORT).show();

                            profileName.clearFocus();
                            profileNumber.clearFocus();
                            profileGender.clearFocus();
                            proUsername.clearFocus();
                            profileAddress.clearFocus();
                            Intent intent=new Intent(getApplicationContext(),Login.class);
                            startActivity(intent);
                        }

                    }
                }, new Response.ErrorListener() {
            @Override
            public void onErrorResponse(VolleyError error) {
                Toast.makeText(getApplicationContext(), error.toString(),
                Toast.LENGTH_SHORT).show();
            }
        })
        {
            @Override
            protected java.util.Map<String, String> getParams() throws AuthFailureError {
                Map<String,String> n=new HashMap<>();
                n.put("id",id);
                return n;
            }
        };
        RequestQueue Var1= Volley.newRequestQueue(this);
        Var1.add(sr);
    }
}
```

PetsDetailsActivity.java

```
public class PetsDetailsActivity extends AppCompatActivity {

    Button buy,call,cht;
    String breed,price,le,id,cat,img,user_id,date,contact;
    ImageView pets;
    TextView tvbreed,tvprice,tvle,tvcat;
    String status,error;
    String url=Config.BaseURL+"sale.php";


    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_pets_details);

        buy=findViewById(R.id.buy1);
        call=findViewById(R.id.interest);

        pets = findViewById(R.id.pets1);

        user_id = new SessionManager(this).getUserDetails().get("user_id");
        Intent i = getIntent();
        id = i.getStringExtra("id");
        cat = i.getStringExtra("cat");
        breed = i.getStringExtra("breed");
        le = i.getStringExtra("le");
        price = i.getStringExtra("price");
        contact=i.getStringExtra("contact");
        img=i.getStringExtra("img");

        //Toast.makeText(PetsDetailsActivity.this,id, Toast.LENGTH_SHORT).show();

        tvbreed = findViewById(R.id.breed1);
        tvprice = findViewById(R.id.price2);
        tvle = findViewById(R.id.le1);
        tvcat = findViewById(R.id.cat);
        tvbreed.setText("Breed :"+breed);
        tvprice.setText("Price :"+price);
        tvle.setText("Life expentancy :"+le);
        tvcat.setText("Category :"+cat);


        if (!TextUtils.isEmpty(img)) {
            String ImgURL = Config.StreetURL + img;
            Picasso.get().load(ImgURL).into(pets);

        }

        buy.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                registration();

            }

        });

        call.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {

                Intent intent = new Intent(Intent.ACTION_DIAL);
                intent.setData(Uri.parse("tel:"+contact));
                startActivity(intent);
            }


        });
        @SuppressLint("SimpleDateFormat") SimpleDateFormat df =
        new SimpleDateFormat("yyyy-MM-dd");
        date = df.format(new Date());

    }


        private void registration()

        {

            StringRequest request=new StringRequest(Request.Method.POST,
            url, new Response.Listener<String>() {
                @Override
                public void onResponse(String response) {
                    Toast.makeText(PetsDetailsActivity.this, "Added to wishlist",
                    Toast.LENGTH_LONG).show();
                    try {
                        JSONObject s = new JSONObject(response);
                        status = s.getString("status");
                        error = s.getString("message");
                    } catch (JSONException e) {
                        e.printStackTrace();
                    }
                }
            }, new Response.ErrorListener() {
                @Override
                public void onErrorResponse(VolleyError error) {
                    Toast.makeText(PetsDetailsActivity.this, error.toString(),
                    Toast.LENGTH_SHORT).show();


                }
            })
            {
                @Override
                protected Map<String, String> getParams() throws AuthFailureError {
                    Map<String,String> m=new HashMap<>();

                    m.put("item_id",id);
                    m.put("user_id",user_id);
                    m.put("sale_date",date);
                    m.put("item_category",cat);
                    m.put("item_breed",breed);
                    m.put("life_expentancy",le);
                    m.put("item_price",price);
                    m.put("item_image",img);
                    m.put("contact",contact);



                    return m;
                }
            };

            RequestQueue q= Volley.newRequestQueue(this);
            q.add(request);

        }
}
```

HomeActivity.java

```
public class HomeActivity extends AppCompatActivity {

    String name,username,address,gender,mobile_no,id;
    int count=0;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_home);

        Intent intent=getIntent();
        id=intent.getStringExtra("id");
        mobile_no=intent.getStringExtra("mobile_no");
        name=intent.getStringExtra("name");
        username=intent.getStringExtra("username");
        address=intent.getStringExtra("address");
        gender=intent.getStringExtra("gender");
        BottomNavigationView navView = findViewById(R.id.nav_view);
        AppBarConfiguration appBarConfiguration =
        new AppBarConfiguration.Builder(
                R.id.navigation_home, R.id.navigation_dashboard,
                R.id.navigation_notifications)
                .build();
        NavController navController =
        Navigation.findNavController(this, R.id.nav_host_fragment);
        NavigationUI.setupActionBarWithNavController(this,
        navController, appBarConfiguration);
        NavigationUI.setupWithNavController(navView, navController);
    }

    @Override
    public void onBackPressed() {
        if (count == 0) {
            showExitAlert();
        } else {
            System.exit(0);
        }
    }
    private void showExitAlert() {
        count=1;
        new AlertDialog.Builder(this)
                .setTitle("Logout")
                .setMessage("Are you sure you want to logout from your account?")
                .setPositiveButton("Yes", new DialogInterface.OnClickListener() {
```

```
                @Override
                public void onClick(DialogInterface dialog, int which) {
                    new SessionManager(HomeActivity.this).logoutUser();
                    startActivity(new Intent(HomeActivity.this, Login.class));
                    finish();
                }
            })
            .setNegativeButton("Cancel", new DialogInterface.OnClickListener() {
                @Override
                public void onClick(DialogInterface dialog, int which) {
                }
            })
            .show();
}
@Override
public boolean onCreateOptionsMenu(Menu menu) {
    MenuInflater inflater=getMenuInflater();
    inflater.inflate(R.menu.wish_menu,menu);
    return true;
}

@Override
public boolean onOptionsItemSelected(@NonNull MenuItem item) {

    switch (item.getItemId())
    {
        case R.id.wishmenu:
           startActivity(new Intent(getApplicationContext(),WishViewActivity.class));

            return true;

        default:
            return super.onOptionsItemSelected(item);
    }

}
```

ComplaintsActivity.java

```
public class ComplaintsActivity extends AppCompatActivity {
   EditText edtEmail,edtComplaint;
   Button btnComplaint;

   String status,error,email,complaint;
   String url = Config.BaseURL+"complaint.php";

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_complaints);

        edtEmail=findViewById(R.id.edtEmail);
        edtComplaint=findViewById(R.id.edtComplaint);
        btnComplaint=findViewById(R.id.btnComplaint);

        btnComplaint.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                addComplaint();
            }
        });


    }
    private void addComplaint()
    {
        email=edtEmail.getText().toString();
        complaint=edtComplaint.getText().toString();

        if (TextUtils.isEmpty(email))
        {
            edtEmail.setError("Email is required");
            edtEmail.requestFocus();
            return;
        }
        else if (TextUtils.isEmpty(complaint)){
            edtComplaint.setError("Complaint is required");
            edtComplaint.requestFocus();
            return;
        }

        StringRequest request=new StringRequest(Request.Method.POST,
        url, new Response.Listener<String>() {
            @Override
            public void onResponse(String response) {
                try {
                    JSONObject s = new JSONObject(response);
                    status = s.getString("status");
                    error = s.getString("message");
                } catch (JSONException e) {
                    e.printStackTrace();
                }
                if ("0".equals(status)) {
                    Toast.makeText(ComplaintsActivity.this, error,
                    Toast.LENGTH_SHORT).show();
                } else {
                    Toast.makeText(ComplaintsActivity.this,error,
                    Toast.LENGTH_SHORT).show();
                    Intent intent = new Intent(ComplaintsActivity.this, HomeActivity.class);
                    startActivity(intent);
                    finish();
                }
            }
        }, new Response.ErrorListener() {
            @Override
            public void onErrorResponse(VolleyError error) {
                Toast.makeText(ComplaintsActivity.this, error.toString(),
                Toast.LENGTH_SHORT).show();
            }
        })
        {
            @Override
            protected Map<String, String> getParams() throws AuthFailureError {
                Map<String,String> m=new HashMap<>();
                m.put("email",email);
                m.put("complaints",complaint);

                return m;
            }
        };


        RequestQueue q= Volley.newRequestQueue(this);
        q.add(request);
    }
}
```

FeedbackActivity.java

```
public class FeedbackActivity extends AppCompatActivity {

    EditText email,feedback;
    Button btnFeedback;

    String status,error,email1,feedback1;

    String url=Config.BaseURL+"feedback.php";

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_feedback);

        email=findViewById(R.id.edtFmail);
        feedback=findViewById(R.id.edtFeedback);
        btnFeedback=findViewById(R.id.btnFeedback);

        btnFeedback.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                addFeedback();
            }
        });
    }

    private void addFeedback(){

        email1 = email.getText().toString();
        feedback1 = feedback.getText().toString();

        if (TextUtils.isEmpty(email1))
        {
            email.setError("Email is required");
            email.requestFocus();
            return;
        }
        else if (TextUtils.isEmpty(feedback1)){
            feedback.setError("Complaint is required");
            feedback.requestFocus();
            return;
        }

        StringRequest request=new StringRequest(Request.Method.POST,
        url, new Response.Listener<String>() {
            @Override
            public void onResponse(String response) {
                try {
                    JSONObject s = new JSONObject(response);
                    status = s.getString("status");
                    error = s.getString("message");
                } catch (JSONException e) {
                    e.printStackTrace();
                }
                if (status.equals("0")) {
                    Toast.makeText(FeedbackActivity.this, error, Toast.LENGTH_SHORT).show();
                } else {
                    Toast.makeText(FeedbackActivity.this, "Registration successful",
                    Toast.LENGTH_SHORT).show();
                    Intent intent = new Intent(FeedbackActivity.this, HomeActivity.class);
                    startActivity(intent);
                    finish();
                }
            }
        }, new Response.ErrorListener() {
            @Override
            public void onErrorResponse(VolleyError error) {
                Toast.makeText(FeedbackActivity.this, error.toString(),
                Toast.LENGTH_SHORT).show();
            }
        })
        {
            @Override
            protected Map<String, String> getParams() throws AuthFailureError {
                Map<String,String> m=new HashMap<>();
                m.put("email",email1);
                m.put("feedback",feedback1);

                return m;
            }
        };


        RequestQueue q= Volley.newRequestQueue(this);
        q.add(request);
    }
}
```

WishlistViewActivity.java

```
public class WishViewActivity extends AppCompatActivity {

    private String URLstring = Config.BaseURL + "wishlist_view.php";

    private static ProgressDialog mProgressDialog;
    ArrayList<WishDataModel> dataModelArrayList;
    WishAdapter rvAdapter;
    RecyclerView recyclerView;
    String mobile,user_id;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_wish_view);

        user_id = new SessionManager(this).getUserDetails().get("user_id");
        recyclerView = findViewById(R.id.recyclerWish);
        fetchingJSON();
    }
    private void fetchingJSON() {

        showSimpleProgressDialog(this,"Loading...","Fetching Json",false);

        StringRequest stringRequest = new StringRequest(Request.Method.POST, URLstring,
                new Response.Listener<String>() {
                    @Override
                    public void onResponse(String response) {
                        //Toast.makeText(WishViewActivity.this, response,
                       Toast.LENGTH_SHORT).show();

                        try {
                            removeSimpleProgressDialog();

                            dataModelArrayList = new ArrayList<>();

                            JSONArray array = new JSONArray(response);

                            for (int i = 0; i < array.length(); i++) {

                                JSONObject dataobj = array.getJSONObject(i);

                                dataModelArrayList.add(new WishDataModel(
                                       dataobj.getString("s_id"),
                                       dataobj.getString("user_id"),
                                        dataobj.getString("item_category"),
                                        dataobj.getString("item_breed"),
                                        dataobj.getString("life_expentancy"),
                                        dataobj.getString("item_price"),
                                        dataobj.getString("item_image"),
                                        dataobj.getString("contact")
                                ));

                            }
                            setupRecycler();

                        } catch (JSONException e) {
                            e.printStackTrace();
                        }
                    }
                },
                new Response.ErrorListener() {
                    @Override
                    public void onErrorResponse(VolleyError error) {
                        Log.e("Log", "inside onErrorResponse");
                        //displaying the error in toast if occurrs
                        Toast.makeText(getApplicationContext(), error.getMessage(),
                        Toast.LENGTH_SHORT).show();
                    }
                }){
            @Override
            protected Map<String, String> getParams() throws AuthFailureError {
                Map<String,String> map=new HashMap<>();
                map.put("user_id",user_id);
                return map;
            }
        };

        // request queue
        RequestQueue requestQueue = Volley.newRequestQueue(this);
        requestQueue.add(stringRequest);
    }

    private void setupRecycler(){

        rvAdapter = new WishAdapter(this, dataModelArrayList);
        recyclerView.setHasFixedSize(true);
        recyclerView.setAdapter(rvAdapter);
        recyclerView.setLayoutManager(new LinearLayoutManager
       (this, LinearLayoutManager.VERTICAL, false));
    }
```

```
    public static void removeSimpleProgressDialog() {
        try {
            if (mProgressDialog != null) {
                if (mProgressDialog.isShowing()) {
                    mProgressDialog.dismiss();
                    mProgressDialog = null;
                }
            }
        } catch (IllegalArgumentException ie) {
            Log.e("Log", "inside catch IllegalArgumentException");
            ie.printStackTrace();

        } catch (RuntimeException re) {
            Log.e("Log", "inside catch RuntimeException");
            re.printStackTrace();
        } catch (Exception e) {
            Log.e("Log", "Inside catch Exception");
            e.printStackTrace();
        }

    }

    public static void showSimpleProgressDialog(Context context, String title,
                                                String msg, boolean isCancelable) {
        try {
            if (mProgressDialog == null) {
                mProgressDialog = ProgressDialog.show(context, title, msg);
                mProgressDialog.setCancelable(isCancelable);
            }

            if (!mProgressDialog.isShowing()) {
                mProgressDialog.show();
            }

        } catch (IllegalArgumentException ie) {
            ie.printStackTrace();
        } catch (RuntimeException re) {
            re.printStackTrace();
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
}
```

ForgotPassword.java

```
public class Forgotpswrd extends AppCompatActivity {
    EditText etMob;
    Button bt1;
    //String fgfdh;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_forgotpswrd);
        etMob = findViewById(R.id.mobileForgot);
        //etOtp = findViewById(R.id.otpForgot);
        bt1 = findViewById(R.id.sendForgot);
        //bt2 = findViewById(R.id.verify);

        bt1.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                checkPermission();
            }
        });
    }

    public void sendOTP() {
        String num = etMob.getText().toString();
        Random r = new Random();
        int otp = r.nextInt((9999 - 1000) + 1) + 1000;

        String msg = "Welcome to petz Management System.
        Your OTP for changing the password is " + otp;

        SmsManager sms = SmsManager.getDefault();
        sms.sendTextMessage(num, null, msg, null, null);

        Intent i = new Intent(Forgotpswrd.this, Verify.class);
        i.putExtra("otp", Integer.toString(otp));
        i.putExtra("phone", num);
        startActivity(i);
        finish();
    }


    // Function to check and request permission.
    public void checkPermission()
    {
        if (ContextCompat.checkSelfPermission(Forgotpswrd.this,
        Manifest.permission.SEND_SMS) == PackageManager.PERMISSION_DENIED) {

            // Requesting the permission
            ActivityCompat.requestPermissions(Forgotpswrd.this, new String[]
           { Manifest.permission.SEND_SMS }, 1);
        }
        else {
            sendOTP();
        }
    }
    public void onRequestPermissionsResult(int requestCode,
    @NonNull String[] permissions, @NonNull int[] grantResults)
    {
        super.onRequestPermissionsResult(requestCode, permissions, grantResults);

        if (requestCode == 1) {
            if (grantResults.length > 0 &&
            grantResults[0] == PackageManager.PERMISSION_GRANTED) {
                sendOTP();
            }
            else {
                Toast.makeText(Forgotpswrd.this, "SMS Permission Denied",
                Toast.LENGTH_SHORT).show();
            }
        }
    }


    @Override
    public void onRequestPermissionsResult(int requestCode,
    @NonNull String[] permissions, @NonNull int[] grantResults)
    {
        super.onRequestPermissionsResult(requestCode, permissions, grantResults);

        if (requestCode == CAMERA_PERMISSION_CODE) {
            if (grantResults.length > 0 && grantResults[0] ==
            PackageManager.PERMISSION_GRANTED) {
                sendOTP();
            }
           else {
                Toast.makeText(Forgotpswrd.this, "Camera Permission Denied",
                Toast.LENGTH_SHORT).show();
            }
        }
}
}
```