

# THE SUPERIOR COLLEGE LAHORE



Faculty of Computer Science

Final Year Project  
**PROJECT REPORT**

**Maweshi Mandi**

Project ID: **FYP-BSCS-F18-038**

## Project Team

Student Name	Student ID	Program	Contact Number	Email Address
Abdul Waheed	BCSM-F15-340	BS(CS)	03066734160	<a href="mailto:abdulwaheedch160@gmail.com">abdulwaheedch160@gmail.com</a>
Muhammad Furqan	BCSM-F15-088	BS(CS)	03024118047	<a href="mailto:muhammadfurqan133660@gmail.com">muhammadfurqan133660@gmail.com</a>
Khurram Shafique	BCSM-F15-114	BS(CS)	03034144964	<a href="mailto:ranakhurram114@gmail.com">ranakhurram114@gmail.com</a>

**Mr. Umair Ahmad**  
(Lecturer)

# Project Report

## Maweshi Mandi

### Change Record

Author(s)	Version	Date	Notes	Supervisor's Signature
Project Proposal	1.0	10/09/2018	Initial Version	
Project Proposal	1.1	12/11/2018	Changes Based on Feedback from Supervisor	
Project Proposal	1.2	02/01/2019	Changes Based on Feedback From Faculty	
Project Proposal Final Version	1.3	22/02/2019	Added Project Plan	
Project Report - 1	2.0	10/03/2019		
Project Report - 2	3.0		1. Changes Based on Feedback 2. Adding Testing Report	

## APPROVAL

---

### PROJECT SUPERVISOR

Comments: \_\_\_\_\_

---

Name: \_\_\_\_\_

Date: \_\_\_\_\_

Signature: \_\_\_\_\_

---

### PROJECT MANAGER

Comments: \_\_\_\_\_

---

Date: \_\_\_\_\_

Signature: \_\_\_\_\_

---

### HEAD OF THE DEPARTMENT

Comments: \_\_\_\_\_

---

Date: \_\_\_\_\_

Signature: \_\_\_\_\_



## **Dedication**

*This work is dedicated to my family, friends and everyone who showered me with their love and support. I am especially thankful to those who presented challenges to my work and I pray for them because without their resistance I would never have been able to achieve what I did.*

## **Acknowledgements**

I am really thankful to my supervisor who has guided us well throughout the semester. As well as the people that provided us with their honest opinions and reviews. Who made time for us for conducting their interview from their busy schedules. I would also like to thank my friends and family that provided us with the opportunity and support for developing this project.

## Executive Summary

This proposal is for our final year project, “Maweshi Mandi”. The purpose of this project is to create a platform where users can purchase and sales of animals. The growing use of Internet in Pakistan provides a developing prospect for online shopping. If E-marketers know the factors affecting online Pakistan behavior, and the relationships between these factors and the type of online buyers, then they can further develop their marketing strategies to convert potential customers into active ones, while retaining existing online customers.

This project is a part of study, and focuses on factors which online Pakistan buyers keep in mind while shopping online. This research found that information, perceived usefulness, ease of use perceived enjoyment and security/privacy are the five dominant factors which influence consumer perceptions of online purchasing. Consumer behavior is said to be an applied discipline as some decisions are significantly affected by their behavior or expected actions. The two perspectives that seek application of its knowledge are micro and societal perspectives. The online purchasing behavior of online shoppers and factor influencing online shopping behavior and its future perspective. Internet is changing the way consumers shop and buy goods and services, and has rapidly evolved into a global phenomenon.

Our project is based on online purchase and sales of animal's .It include all type of Animals like that Cow, Buffalo, Camel, Sheep’s and Goats etc. This solve the problem of peoples and farmers to go outside and spent his time for his desire animals. Some people can purchase the animals for meat and the skin of the body of an animal sale to others for Leather using online System. We provide the online platform to buy and sell their Animals which is easy for them it will save their time and cost. They will get their desire animals in sitting their home just in one click. It will easy for both seller and buyer. The many other online website can laying in the e-commerce market but our project only based on cattle related and other animals like that poultry forms etc.

The summary contains the following elements:

- This proposal would bestow the readers to understand the scope, need, estimation, budgeting and complete development cycle required to develop the project.
- This is a software project and requires technical expertise which will be fulfilled by the Supervisor and other staff members at their ease.
- The total anticipated budget.
- Other information deemed pertinent.

## Table of Contents

Dedication.....	v
Acknowledgements.....	vi
Executive Summary.....	vii
Table of Contents.....	viii
List of Figures.....	xi
List of Tables.....	xii
Chapter 1.....	1
Introduction.....	1
1.1. Background.....	2
1.2. Motivations and Challenges.....	2
1.3. Goals and Objectives.....	2
1.4. Literature Review/Existing Solutions.....	3
1.5. Gap Analysis.....	4
1.6. Proposed Solution.....	5
1.7. Project Plan.....	5
1.7.1. Work Breakdown Structure.....	6
1.7.2. Roles & Responsibility Matrix.....	8
1.7.3. Gantt chart.....	11
Chapter 2.....	12
Software Requirement Specifications.....	12
2.1. Introduction.....	13
2.1.1. Purpose.....	13
2.1.2. Document Conventions.....	13
2.1.3. Intended Audience and Reading Suggestions.....	13
2.1.4. Project Scope.....	14
2.1.5. References.....	15
2.2. Overall Description.....	16
2.2.1. Product Perspective.....	16
2.2.2. Product Functions.....	16
2.2.3. User Classes and Characteristics.....	18
2.2.4. Operating Environment.....	18
2.2.5. Design and Implementation Constraints.....	19
2.2.6. User Documentation.....	19
2.2.7. Assumptions and Dependencies.....	19
2.3. External Interface Requirements.....	20
2.3.1. User Interfaces.....	28
2.3.2. Hardware Interfaces.....	28
2.3.3. Software Interfaces.....	28
2.3.4. Communications Interfaces.....	28
2.4. System Features.....	29

2.4.1.	Admin Panel.....	30
2.4.1.1.	Description and Priority .....	30
2.4.1.2.	Stimulus/Response Sequences.....	30
2.4.1.3.	Functional Requirements.....	31
2.4.2.	Seller Panel .....	31
2.4.2.1.	Description and Priority .....	31
2.4.2.2.	Stimulus/Response Sequences.....	32
2.4.2.3.	Functional Requirements.....	32
2.4.3.	User Panel .....	33
2.4.3.1.	Description and Priority.....	33
2.4.3.2.	Stimulus/Response Sequences.....	33
2.4.3.3.	Functional Requirements.....	33
2.4.4.	Genuine Animals .....	34
2.4.4.2.	Stimulus/Response Sequences.....	34
2.4.4.3.	Functional Requirements.....	34
2.4.5.	Intelligent Reporting.....	35
2.4.5.1.	Description and Priority.....	35
2.4.5.2.	Stimulus/Response Sequences.....	35
2.4.5.3.	Functional Requirements.....	35
2.4.6.	CRUD Operations .....	36
2.4.6.1.	Description and Priority .....	36
2.4.6.2.	Stimulus/Response Sequences.....	36
2.4.6.3.	Functional Requirements.....	36
2.4.7.	Secure User Registration & Login/Logout .....	37
2.4.7.1.	Description and Priority.....	37
2.4.7.2.	Stimulus/Response Sequences.....	37
2.4.7.3.	Functional Requirements.....	37
2.4.8.	User & other Users Search Animal Filter .....	38
2.4.8.1.	Description and Priority .....	38
2.4.8.2.	Stimulus/Response Sequences.....	38
2.4.8.3.	Functional Requirements.....	38
2.5.	Other Nonfunctional Requirements .....	39
2.5.1.	Performance Requirements .....	39
2.5.2.	Safety Requirements.....	39
2.5.3.	Security Requirements.....	39
2.5.4.	Software Quality Attributes .....	39
2.5.5.	Business Rules .....	40
2.6.	Other Requirements .....	40
Chapter 3	.....	41
Use Case Analysis	.....	41
3.1	Use Case Mode .....	43
	.....	43
3.1.1	Visitor Use Case: .....	44
3.1.2	User Use Case: .....	44
3.1.3	Admin Use Case: .....	45
3.2	Fully Dressed Use Cases .....	45

Chapter 4 .....	56
System Design .....	56
4.1 Architecture Diagram: .....	57
4.2. Domain Model: .....	58
4.3. Entity Relationship Diagram with Data Dictionary .....	59
4.4. Class Diagram .....	64
4.5. Sequence / Collaboration Diagram .....	65
4.5.1 Visitor: .....	68
4.5.2 User:.....	68
4.6. Operation Contracts .....	69
OP1 .....	69
OP2.....	69
OP3.....	69
OP4.....	69
OP5.....	69
OP6.....	70
OP7.....	70
OP9.....	70
OP10.....	70
OP11 .....	71
OP12.....	71
OP13.....	71
OP14.....	71
4.7. Activity Diagram User.....	72
4.7.1 Activity Diagram Admin .....	73
4.8. State Transition Diagram .....	74
4.8.1 Visitor: .....	74
4.8.2 User:.....	75
4.9. Component Diagram.....	76
4.10. Deployment Diagram.....	77
4.11. Data Flow Diagram .....	78
0 Level: .....	78
Level 1: .....	79
Chapter 5 .....	80
Implementation .....	80
5.1. Important Flow Control/Pseudo codes.....	82
5.2. Components, Libraries, Web Services and stubs.....	84
5.3. Deployment Environment .....	84
5.4. Tools and Techniques .....	85
5.5. Best Practices / Coding Standards .....	85
Chapter 6 .....	86
Testing and Evaluation.....	86
6.1. Use Case Testing .....	87
6.2. Equivalence partitioning.....	96
6.3. Boundary value analysis .....	97
6.4. Data flow testing .....	97

6.4 Unit testing .....	102
6.6. Integration testing .....	103
6.7. Performance testing .....	111
6.8. Stress Testing .....	111
Chapter 7 .....	112
Summary, Conclusion and Future Enhancements .....	112
7.1. Project Summary .....	113
7.2. Achievements and Improvements .....	113
7.3. Lessons Learnt.....	113
7.4. Future Enhancements .....	114
Reference and Bibliography .....	115

## List of Figures

Figure 1 Project Plan.....	6
Figure 2 Gantt chart .....	11
Figure 3 Home Page of MM.....	20
Figure 4 Post Adds.....	20
Figure 5 Post Ads of MM.....	21
Figure 6 Visitor Use Case.....	44
Figure 7 User Use Case .....	44
Figure 8 Admin Use Case.....	45
Figure 9 Architecture Diagram .....	57
Figure 10 Domain Model .....	58
Figure 11 Entity Relationship Diagram with Data Dictionary .....	59
Figure 12 Class Diagram.....	64
Figure 13 Visitor Search/Review .....	65
Figure 14 User Ads /Buy & Sells Animals .....	66
Figure 15 User login/logout.....	66
Figure 16 Admin Manipulates Category/User Account/Transection/Payments/Check Reports & Reviews .....	67
Figure 17 Visitor .....	68
Figure 18 User .....	68
Figure 19 Activity Diagram.....	72
Figure 20 Admin .....	73
Figure 21 State Transition Diagram Consumer .....	74
Figure 22 State Transition Diagram Customer .....	75
Figure 23 Component Diagram .....	76
Figure 24 Deployment Diagram .....	77
Figure 25 Data Flow Diagram 0 level.....	78
Figure 26 Data Flow Diagram level 1 .....	79

## List of Tables

Table 1-0-1 Gap Analysis.....	4
Table 1-0-2 Work Breakdown Structure.....	8
Table 1-0-3 Roles & Responsibility Matrix.....	10
Table 2-0-1 Intended Audience and Reading Suggestions.....	14
Table 2-0-2. User Classes and Characteristics.....	18
Table 2-0-3 Hardware Interfaces.....	28
Table 2-0-4 Admin Panel Description and Priority.....	30
Table 2-0-5 Admin Panel Stimulus/Response Sequences.....	31
Table 2-0-6 Seller Panel Description and Priority.....	31
Table 2-0-7 Seller Panel Stimulus/Response Sequences.....	32
Table 2-0-8 Costumer Panel Description and Priority.....	33
Table 2-0-9 Costumer Panel Stimulus/Response Sequences.....	33
Table 2-0-10 Genuine Animals Description and Priority.....	34
Table 2-0-11 Genuine Animals Stimulus/Response Sequences.....	34
Table 2-0-12 Intelligent Reporting Description and Priority.....	35
Table 2-0-13 Intelligent Reporting Stimulus/Response Sequences.....	35
Table 20-14 CRUD Operations Description and Priority.....	36
Table 2-0-15 CRUD Operations Stimulus/Response Sequences.....	36
Table 2-0-16 Secure User Registration & Login/Logout Description and Priority.....	37
Table 2-0-17 Secure User Registration & Login/Logout Description and Priority.....	37
Table 2-0-18 User & other Users Search Animal Filter Description and Priority.....	38
Table 2-0-19 User & other Users Search Animal Filter Stimulus/Response Sequences.....	38
Table 3-0-1 Registration.....	46
Table 3-0-2 Login.....	47
Table 3-0-3 Logout.....	48
Table 3-0-4 Review Animal.....	48
Table 3-0-5 Manipulate Animal Adds.....	49
Table 3-0-6 Animals Seller.....	50
Table 3-0-7 Animals buyer.....	51
Table 3-0-8 Change Account Setting.....	51
Table 3-0-9 Registration.....	52
Table 3-0-10 Search Animal.....	52
Table 3-0-11 View Animals.....	53
Table 3-0-12 Manipulate Category.....	53
Table 3-0-13 Verify User Account.....	54
Table 3-0-14 Manage Product.....	55
Table 3-15 Check Report.....	55
Table 4-0-1 Admin.....	60
Table 4-0-2 User.....	60
Table 4-0-3 Visitor.....	61
Table 4-0-4 Posted Ads.....	61
Table 4-0-5 Promoted Ads.....	62

Table 4-0-6 Report .....	62
Table 4-0-7 Category .....	62
Table 4-0-8 Brand .....	62
Table 4-0-9 Animals .....	63
Table 4-0-10 Safety Tips .....	63
Table 5-0-1 Implementation .....	81

# Chapter 1

## **Introduction**

# Chapter 1: Introduction

In this chapter the reader will be introduced to the basic idea of the documented product name Maweshi Mandi, the motivation behind it and the problems it will solve. This chapter will also provide the reader with the project plan that will be carried out throughout the development process of this product. It will include the responsibilities matrix, work breakdown structure and the Gantt chart. Finally, the chapter concludes with a brief outline of the document to prepare the reader for what's to come and what to expect.

## 1.1. Background

In the past, people can purchase any animal or bird (etc.) from the market. The customer goes to the market and likes the animal and sees it in real life. After that, they purchase it. On the other side, the seller can go to the market to sell any animal. If they cannot sell, they go back home and waste time and money. So in the past, people purchased animals using a manual traditional system.

## 1.2. Motivations and Challenges

E-commerce sites have existed even before the 21st century. They've become really popular even here in Pakistan. This also includes C2C (customer to customer) dealing companies where a consumer can sell and purchase different products. So, no online platform for selling and purchasing animals in Pakistan. There are not any significant platforms in Pakistan which provide this kind of dealing between consumers. Then I've seen the selling and purchasing traditional system of animals, birds, so me and my group member were motivated and then decided to create an online platform for the cattle system and create "Maweshi Mandi".

## 1.3. Goals and Objectives

To solve the above-mentioned problem, it is our goal to create a single platform Online Maweshi Mandi System which will manage and handle all aspects of the user.

- Develop a website where owner can post an ad of sealer animal, and the buyer can view it and contact the owner.
- Also develop hybrid app for mobile users.

## **Objectives**

- The system will help to Seles and purchases animals
- Save time both end-user seller and buyers.

## **1.4. Literature Review/Existing Solutions**

### **1- Zameen.com**

Zameen.com is a very popular website that deals in real estate. It is the leading property website in all Pakistan and has been named among the best property based websites across the world. It deals in all kinds of property. Shops, houses, plots, apartments, flats and offices are all included in its dealings.

Besides property for sale, users can also search for property on rent. This website is already very mature and successful, making it one of the biggest threat to our online business.

### **2- Bolee.com**

Bolee.com working like that olx.com but it is a Pakistani site so it is not popular in the overall world. But it's working good and get thousands of user's views and earning with online though.

### **3- Olx.com**

OLX is known and used by a huge number of people, all around the globe. Through excellent marketing they are running it in successfully 45 different countries, including Pakistan. Only earning through online, they've become the most successful and well-known business of their kind.

Olx.pk provides ads of all sorts and deals in many different categories. Users can also find ads for property on rent and even for product exchanges.

### **4- Eidqurbani.com**

This site is only working for other peoples help for Religios base etc. We are a team of individuals who are working with the intention of bringing happiness to the poor and needy on the day of Eid ul Adha in Muslim countries that are poor; Muslims living there are in great need of support. Its work on Animals only eid days so its no suffecent working.

## 1.5. Gap Analysis

### Current Situation:

In the current situation people go to the Maweshi Mandi and purchase an animal. It is very difficult for the job holder's people and time consumer process and very expensive for customer. We can say it is a traditional or manually purchasing system.

### After Maweshi Mandi Situation:

User can save time, extra charges and get the good Animal at home by clicking and using this online platform. These are following features.

**Feature 1:** Consumer can directly deal with buyer 7/24 hours.

**Feature 2:** Dealings in every category of animals.

**Feature 3:** System failure reporting and feedbacks and respond quickly on the Query. And selling the multiple Products but not focused one brands.

Applications	Feature1	Feature 2	Feature 3
MaweshiMandi.com	✓	✓	✓
Bolee.com	✓	✓	X
Zameen.com	✓	X	X
Olx..com	✓	✓	X

Table 1-0-1 Gap Analysis

## 1.6. Proposed Solution

The purpose of this document is to collect analyze and define high level needs and features of the Maweshi Mandi System. It focuses on the online sales and purchasing of the Animals and the details that how system will fulfill the needs of seller, buyers and end-users. The main purpose of our system is to provide better online system for the Online Maweshi Mandi system. This will save the visitors' time and Cost. We get a solution for all of these people and form holder which love animals and do marketing for animals. It will be a best way to buy and sell animals. It will solve many problems of buyer and sellers which include time, Quality product and more.

- Develop a website where owner can post an ad of sealer animal, and the buyer can view it and contact the owner.
- Also develop hybrid app for Android mobile users.

## 1.7. Project Plan

A project plan is a formal document designed to guide the control and execution of a project. A project plan is the key to a successful project and is the most important document that needs to be created when starting any business project. These are following steps for completing project plan.

- **Requirement Gathering**
- **Analyses of Requirements**
- **Designing**
- **Development**
- **Testing & Deployment**

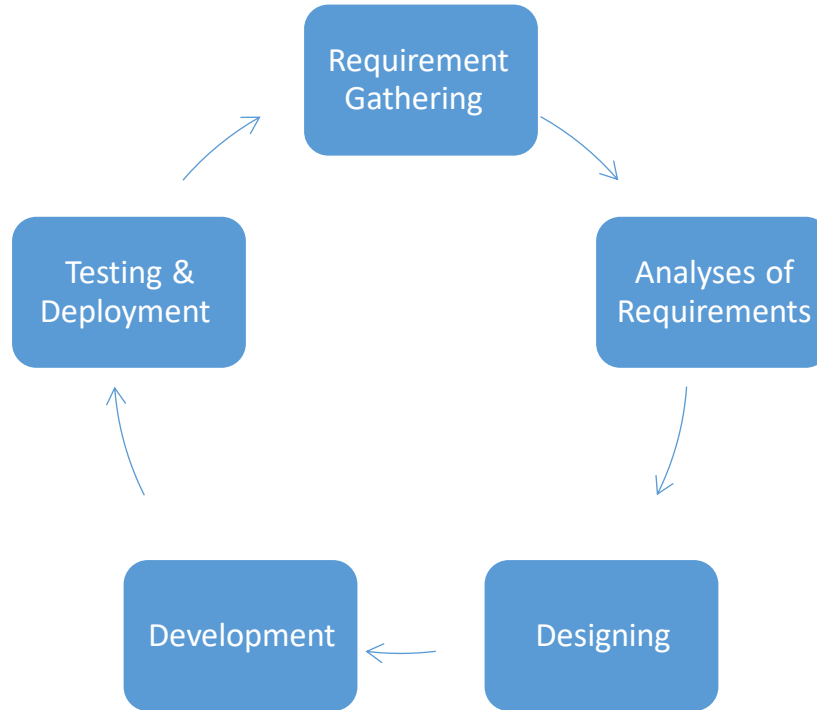


Figure 1 Project Plan

### 1.7.1. Work Breakdown Structure

WBS Name		WBS Name	
<b>Requirement Gathering</b>	Establish Design Technical specification Restricted areas Available solutions	<b>Product Management</b>	<b>Module Products:</b> Add new products Update products Delete Products View Products <b>Categories:</b> Add new Categories Update Categories Delete Categories View Categories <b>Brands:</b> Add new brands Update brands Delete brands View brands
<b>Establish Design</b>	<b>Website</b> Design elements Banner	<b>Documentation</b>	SRS SDS User Documentation

	<p>Footer Logo Color scheme Font usage Icons Slider Content Header <b>Mobile App</b> Design elements Banner Footer Logo Color scheme Font usage Icons Content Header</p>		
<b>Admin panel:</b>	<p>Admin dashboard Customer detail page Delete customer account Visit page List CRUD operation on products CRUD operation on categories CRUD operation on brands Verify customer account page.</p>	<b>Services</b>	REST web API Services
<b>Seller panel:</b>	<p>Account Page Account Dashboard Page Account seller Items Account my ads Account setting page</p>	<b>Testing</b>	<p>Performance testing Load testing Security testing Unit testing Integration testing System testing</p>
<b>Purchaser/Consumer panel:</b>	<p>Home/index page Search Page Product detail page Not Found Page. Login/ Signup Page Contact Us Page Help Page General safety tips Page A bit about us Page Product oriented safety tips</p>	<b>Roll out Site</b>	<p>Establish target date Create communication plan Make site live</p>

	Page Feedback Page (Both Consumer & buyer)		
<b>Database Module</b>	Design database ERD Perform crud operation on products.		

Table 1-0-2 Work Breakdown Structure

### 1.7.2. Roles & Responsibility Matrix

WBS #	WBS Deliverable	Activity #	Activity to Complete the Deliverable	Duration (# of Days)	Responsible Team Member(s) & Role(s)
1	Requirement gathering	1	<ul style="list-style-type: none"> <li>• Interviews</li> <li>• Research</li> <li>• Competitors Analysis</li> <li>• Purify feasible requirements.</li> </ul>	15 days	<b>Abdul Waheed</b>  <b>M Furqan</b>  <b>Khurram Shafique</b>
2	Establish Design Interface		<ul style="list-style-type: none"> <li>• Design Wire Layout</li> <li>• Try Different Color</li> <li>• Schemas</li> <li>• Design Logo, Icons, Header, Footer, Slider.</li> <li>• Implement design for responsive website and mobile apps</li> </ul>	20 days	<b>Khurram Shafique &amp; Abdul Waheed:</b> Do all designing work and also implement front-end for mobiles apps. <b>M Furqan:</b> Do implement that design through HTML, CSS, bootstrap. <b>Abdul Waheed:</b> Do implement that design through xml, Java, Android Stodio and all implemented Process od coding.
3	Admin Module		<ul style="list-style-type: none"> <li>• Design Admin Module</li> <li>• Design every activity of admin</li> </ul>	25 days	<b>M Furqan:</b> Do all Implementation through bootstrap and php.

			<ul style="list-style-type: none"> <li>• Implement of design and all activities</li> <li>• Apply and perform all operations in activities</li> </ul>		<b>Khurram Shafique &amp; Abdul Waheed:</b> Design the module and all activities.
4	Seller Module		<ul style="list-style-type: none"> <li>• Design seller Module</li> <li>• Design every activity of buyer</li> <li>• Implement of design and all activities.</li> <li>• Apply and perform all operations in activities</li> <li>• Perform all operations as a layperson</li> </ul>	25 days	<p><b>M Furqan:</b> Do all implementation through bootstrap and php.</p> <p><b>Khurram Shafique:</b> Design the module and all activities.</p> <p><b>Abdul Waheed:</b> Do perform all operations.</p>
5	Purchaser/Consumer Module		<ul style="list-style-type: none"> <li>• Design Consumer Module</li> <li>• Design every activity of consumer</li> <li>• Implement of design and all activities.</li> <li>• Apply and perform all operations in activities</li> <li>• Perform all operations as a layperson</li> </ul>	25 days	<p><b>M Furqan:</b> Do all implementation through bootstrap and php.</p> <p><b>Khurram Shafique:</b> Design the module and all activities.</p> <p><b>Abdul Waheed:</b> Do perform all operations.</p>
6	Database Module		<ul style="list-style-type: none"> <li>• Define data entities and their usage.</li> <li>• Identify relations between entities</li> </ul>	35 days	<p><b>M Furqan:</b> Define data entities and also define relations between them.</p> <p><b>Abdul Waheed:</b> Define relations between entities.</p>
7	Product Management Module		<ul style="list-style-type: none"> <li>• Design Product Module</li> </ul>	30 days	<b>M Furqan:</b> Do all implementation

			<ul style="list-style-type: none"> <li>• Design every activity for products, categories and brands.</li> <li>• Implement design and activities</li> </ul>		<p>through bootstrap and php.</p> <p><b>Khurram Shafique:</b> Design the module and all activities.</p> <p><b>Abdul Waheed:</b> Do perform all operations.</p>
8	Services		<ul style="list-style-type: none"> <li>• Identify and design services.</li> <li>• Implement services.</li> </ul>	20 days	<p><b>M Furqan &amp; Abdul Waheed:</b> Design and implement REST Services.</p>
9	Testing		<ul style="list-style-type: none"> <li>• Test every single module.</li> <li>• Integration test for every module.</li> <li>• Apply different security algorithms then check manually and through tools.</li> <li>• Check performance, load testing through tools.</li> </ul>	days	<p><b>M Furqan:</b> Do Unit and Integration testing.</p> <p><b>Abdul Waheed:</b> Do Security and Performance testing.</p> <p><b>Khurram Shafique:</b> Do load testing.</p>
10	Documentation		<ul style="list-style-type: none"> <li>• Document each and everything from the beginning of the project to the end.</li> <li>• SRS</li> <li>• SDS</li> <li>• Reported Bugs and issues</li> <li>• User Document</li> </ul>	60 days	<p><b>M Furqan:</b> Make SRS &amp; diagram.</p> <p><b>Khurram Shafique:</b> Make Design Documentation</p> <p><b>Abdul Waheed:</b> Reported Bugs and issues Testing Documentation and User documentation</p>
11	Roll out Site		<ul style="list-style-type: none"> <li>• Buy Hosting.</li> <li>• Buy Domain.</li> <li>• Upload website and mobile app.</li> <li>• Make website available online.</li> </ul>	10 days	<p><b>M Furqan:</b></p> <p><b>Abdul Waheed:</b></p> <p><b>Khurram Shafique:</b></p>

Table 1-0-3 Roles &amp; Responsibility Matrix

### 1.7.3. Gantt chart

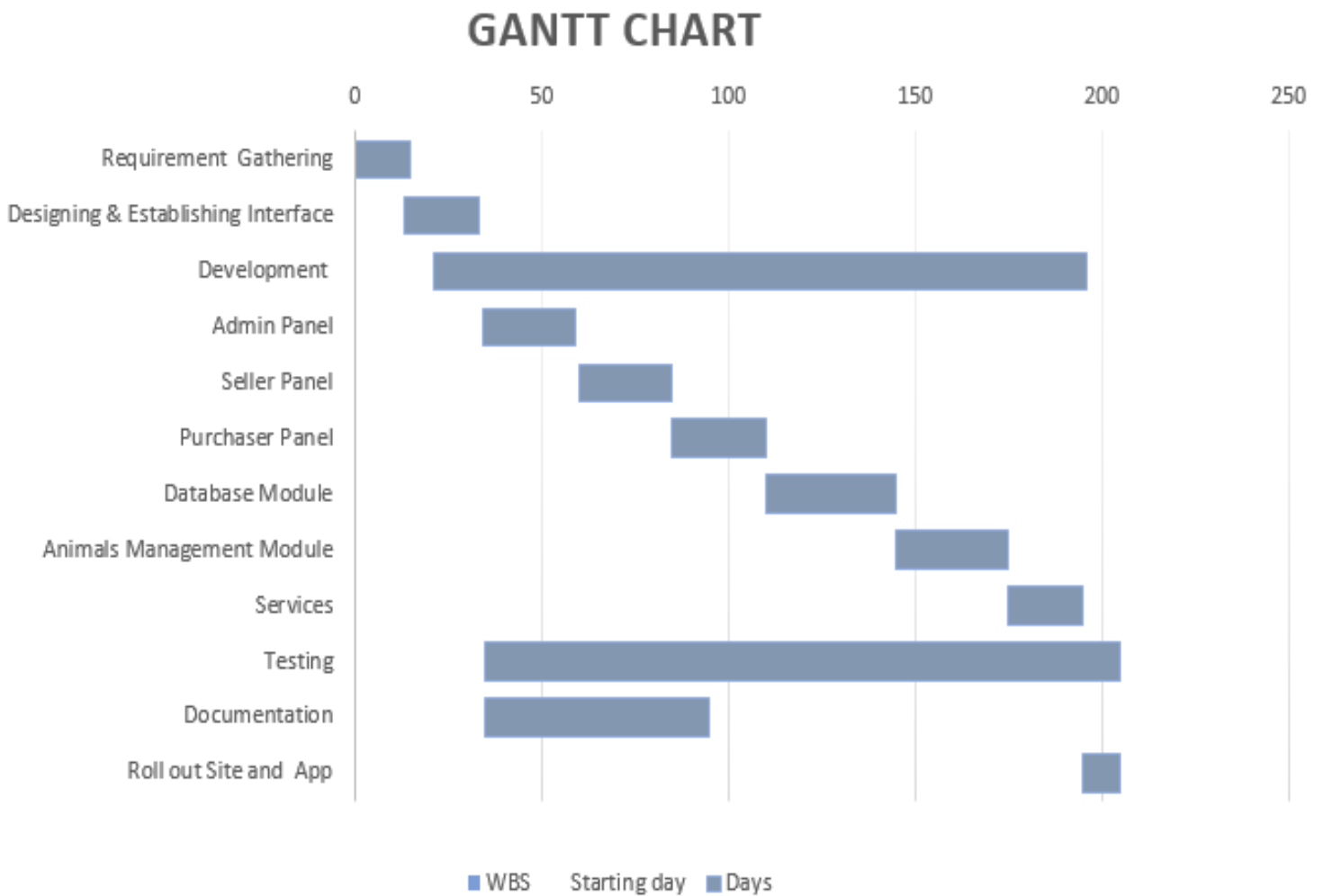


Figure 2 Gantt chart

# Chapter 2

## **Software Requirement Specifications**

## Chapter 2: Software Requirement Specifications

### 2.1. Introduction

#### 2.1.1. Purpose

The e-commerce sites have existed even before the 21st century. They've gotten really popular even here in Pakistan. This also includes C2C (customer to customer) dealing companies where a consumer can sell something that he owns to another consumer willing to buy it. The purpose of this document is to collect analyze and define high level needs and features of the Maweshi Mandi System. It focuses on the online sales and purchasing of the Animals and the details that how system will fulfill the needs of seller, buyers and end-users. The main purpose of our system is to provide better online system for the Online Maweshi Mandi system. This will save the visitors' time and Cost. We get a solution for all of these people and form holder which love animals and do marketing for animals. It will be a best way to buy and sell animals. It will solve many problems of buyer and sellers which include time, Quality product and more.

- Develop a website where owner can post an ad of sealer animal, and the buyer can view it and contact the owner.
- Also develop hybrid app for Android mobile users.

#### 2.1.2. Document Conventions

The format of this SRS is simple. Bold face and indentation is used on general topics and or specific points of interest. The remainder of the document will be written using the standard font, paragraph text 12 pt., Calibri, 1.5 Line Spacing, Justified.

#### 2.1.3. Intended Audience and Reading Suggestions

There are different types of readers for this document and it is helpful for all of that like:

Audience	Use
Developers	They will use this document for guidance in design and documentation.

Client	They can ensure by reading this document that this product met their needs or not.
Managers	Manager will read this document for check all constraints. For example time and cost are in limit or not.
Testers	They will use this document to test the implementation according to requirements.
Documentation Writer	They will use this document during the documentation of the project. It will be really helpful for them.

*Table 2-0-1 Intended Audience and Reading Suggestions*

#### 2.1.4. Project Scope

This project consists of a website and mobile app for mobile users.

We will be building a responsive user-friendly interface. The front-end of the website will use popular HTML, CSS, JS frameworks like Angular for enhancing the user experience and the front-end of the Android App will use popular XML and different library of Android Studio. It will use PHP and SQL as a back-end language and database for website and JAVA and SQLite as a back-end language and database for Android Mobile App.

The following are the key-features of this project.

➤ **Admin panel:**

- Monitor the activities of users
- Verify accounts
- Decision making (via graphs and notifications)
- Getting notified about various issues or important new e.g. too many reports on the user etc.

➤ **Seller panel:**

- Add products with their description in the defined schema as required by the website.
- Give feedback to the users he interacted with.
- View seller charts of the products on the sell.
- Seller report

- **User panel:**
  - View products according to city, categories, price range.
  - Filters for various attributes like condition, area proximity etc.
  - Report Seller of false advertisement.
  - Share experience with seller and the product.
- **Genuine Products:**
  - To ensure users uploads the actual pictures of their animal instead of some image they found on google.
- **Intelligent reporting:**
  - For administrator to see how and where the website is being used and the products are being sold. And for creating research data.
- **CRUD operations on all entities**
- **A secure user registration and login/logout**
- **Product search with filters**

### 2.1.5. References

This document is formed by helping from these sites and documents:

- Executive Summary | Online Shopping | E Commerce. [online] Scribd. Available at: <https://www.scribd.com/document/50093622/Executive-Summary> [Accessed 8 Oct. 2018].
- Status.net Business Templates. (2018). How to Write Project Report: Complete Step-By-Step Guide. [online] Available at: <https://status.net/templates/project-report/> [Accessed 9 Oct. 2018].
- ProjectManager.com. (2018). The Ultimate Guide to Gantt Charts - ProjectManager.com. [online] Available at: <https://www.projectmanager.com/gantt-chart> [Accessed 9 Oct. 2018].
- Edrawsoft.com. (2018). Architecture Diagram Overview. [online] Available at: <https://www.edrawsoft.com/architecture-diagram.php> [Accessed 9 Oct. 2018].
- Totallycommunications.com. (2018). How to define the scope of a project - Totally Communications. [online] Available at:

<https://www.totallycommunications.com/latest/how-to-define-the-scope-of-a-project/>  
[Accessed 9 Oct. 2018].

- Teamgantt.com. (2018). How to Write a Good Project Plan in 10 Steps | Project Management Guide. [online] Available at: <https://www.teamgantt.com/guide-to-project-management/how-to-plan-a-project> [Accessed 9 Oct. 2018].
- Scribd. (2018). SRS for Apni Dukan(an Onineline Shopping Website) | Online Shopping | Use Case. [online] Available at: <https://www.scribd.com/doc/40520541/SRS-for-Apni-Dukan-an-Onineline-Shopping->
- Anon,(2018).[online]Availableat:[http://www.cse.chalmers.se/~feldt/courses/reqeng/examples/srs\\_example\\_2010\\_g roup2.pdf](http://www.cse.chalmers.se/~feldt/courses/reqeng/examples/srs_example_2010_g roup2.pdf) [Accessed 24 Nov. 2018].
- IEEE standard document for software requirement specification.

## 2.2. Overall Description

### 2.2.1. Product Perspective

The product is supposed to be an online platform for users to sell products to others. There are going to be two main user roles in this i.e. the seller (the user that owns a product and provides for sell to others users) and the consumer (the user that is going to be paying the cost in order to purchase the product). It will basically work in the following manner.

1. Seller and Consumers will create their own respective accounts.
2. The seller will then post an ad, with all the pictures and details including condition of the product he is providing for selling.
3. The consumer will search for a product and see the ad. He/she will open the ad and request contact details of the seller of that product.
4. After contacting each other, both users will meet in a safe place (safety tips will be provided to both of them). The consumer will receive the product and return it within the specified period.

### 2.2.2. Product Functions

The following are the key-features of this project.

**❖ Admin panel:**

- Monitor the activities of users
- Verify accounts
- Decision making (via graphs and notifications)
- Getting notified about various issues or important new e.g. too many reports on the user etc.

**❖ Seller panel:**

- Add products with their description in the defined schema as required by the website.
- Give feedback to the users he interacted with.
- View seller charts of the sell Animals/birds
- Seller report

**❖ Visitor panel:**

- View products according to city, categories, price range.
- Filters for various attributes like condition, area proximity etc.
- Registration.
- Search Product.

**❖ Genuine Products:**

- To ensure users uploads the actual pictures of their product instead of some image they found on goggle.

**❖ Intelligent reporting:**

- For administrator to see how and where the website is being used and the products are being sold. And for creating research data.

**❖ CRUD operations on all entities****❖ A secure user registration and login/logout****❖ Product search with filters**

### 2.2.3. User Classes and Characteristics

Class	Characteristics
Admin	It is system administrative class. He will maintain the overall website.
Seller	It is a class of seller. Who create account and post adds.
Consumer	It is a class of visitors. Who search and buy products.
Product adds	It is class of product adds that can be store, update and remove by CRUD operations.
Category	It is class of category that can be store, update and remove by CRUD operations.
Trending adds	It is class of trending adds that can be store, update and remove by CRUD operations.
Sign up	It is sign up class. User can make his account through this class and post adds.
Damage report	It is product damage report class. Renter submit that class and share experience about consumer.
City	It is class of different cities that can be add, update and remove by CRUD operations.
Reports	It is a class of reports that can show user visits and about products on rent.
Paid adds	It is a class of paid adds that can be update, add and remove by CRUD operations.

Table 2-0-2. User Classes and Characteristics

### 2.2.4. Operating Environment

- This system will operate in browsers like Mozilla, Firefox, Opera and for their different version also.
- It can be open on window, android and iPhone.
- The processor should be core 2 duo or above.

- The processor's speed should be 2.8 GHz or greater.
- Ram should be or greater than 512 MB.

### **2.2.5. Design and Implementation Constraints**

- Developers have to complete this system in minimum time say 6 months.
- The languages which will be used in this project is PHP, Bootstrap3, Html, CSS3.
- The Case tool to implement this product is Bracket, Xampp, Adobe Photoshop, Illustrator, Visual studio.
- For testing New Relic and Load Runner tools will be used.
- Default language for this system is English (Easy and understandable for everyone).
- This software will follow all the copyrights cyber laws of PTA (Pakistan Telecommunication Authority).

### **2.2.6. User Documentation**

- We make a blog and help section in system for online help of customer.
- We will give safety disclaimers for both Seller and consumer.
- We will also give the tutorials video that how to use the website on social media like YouTube and Facebook.

### **2.2.7. Assumptions and Dependencies**

User already knows how to browse and shop on online e-commerce site.

## 2.3. External Interface Requirements

### Home Page

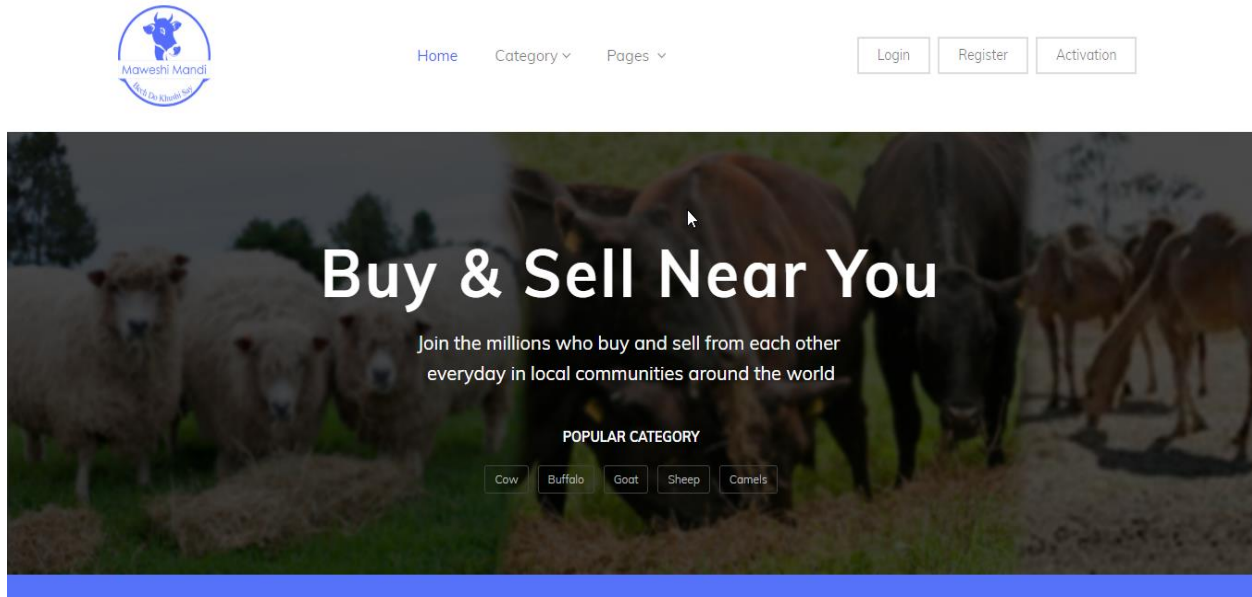


Figure 3 Home Page of MM

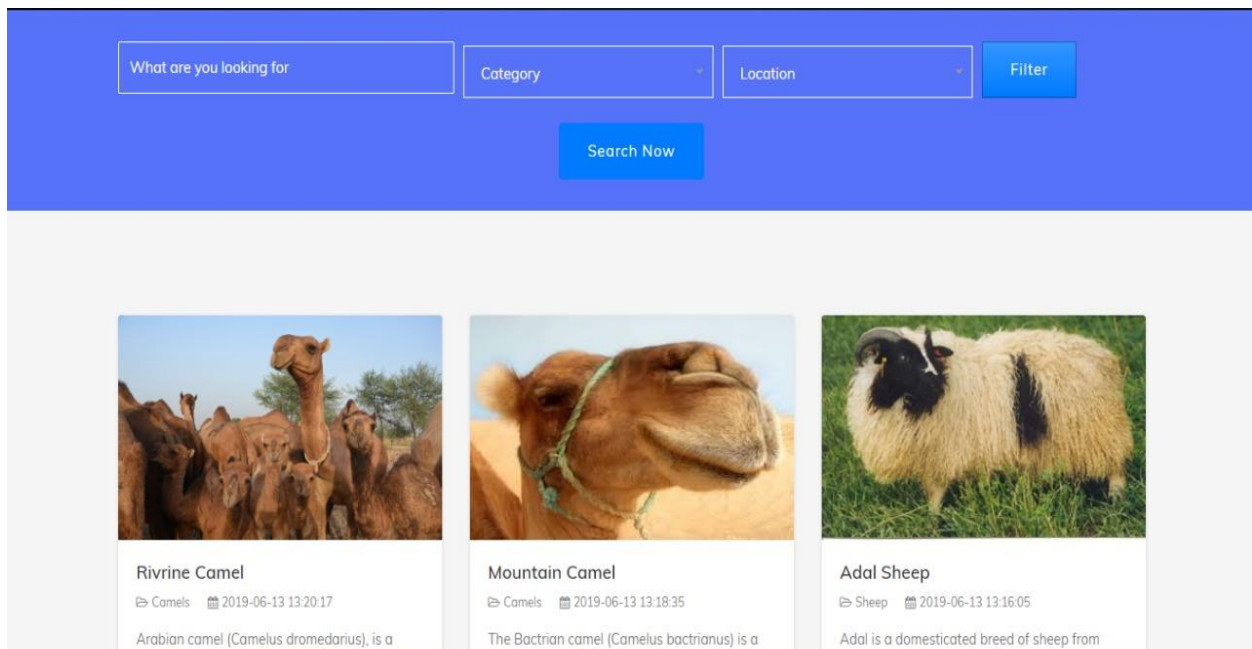


Figure 4 Post Adds

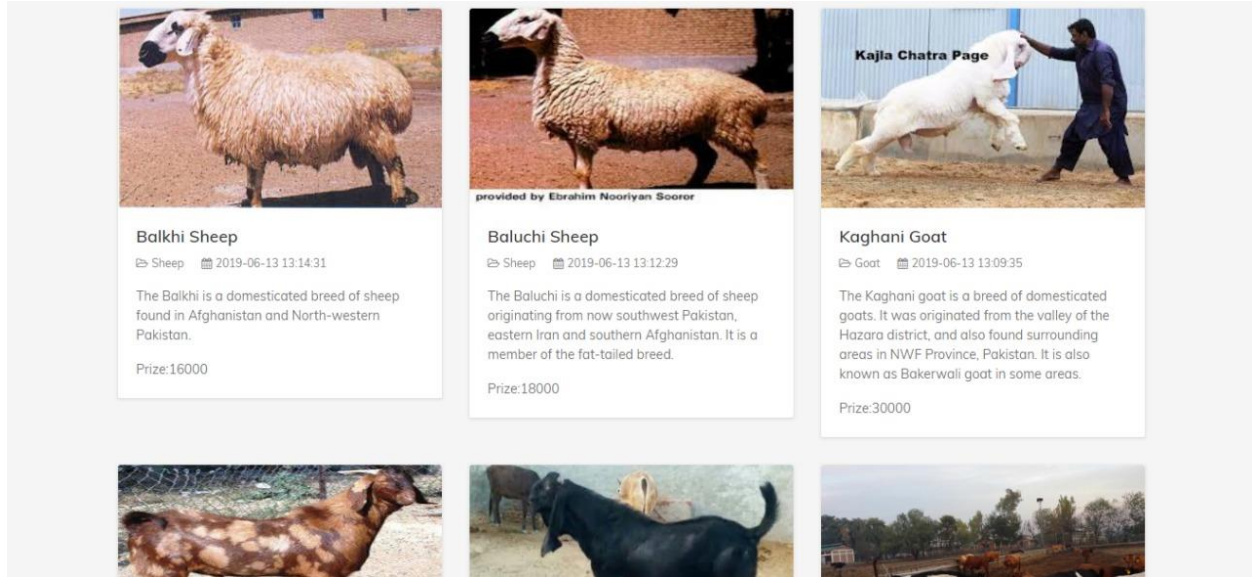
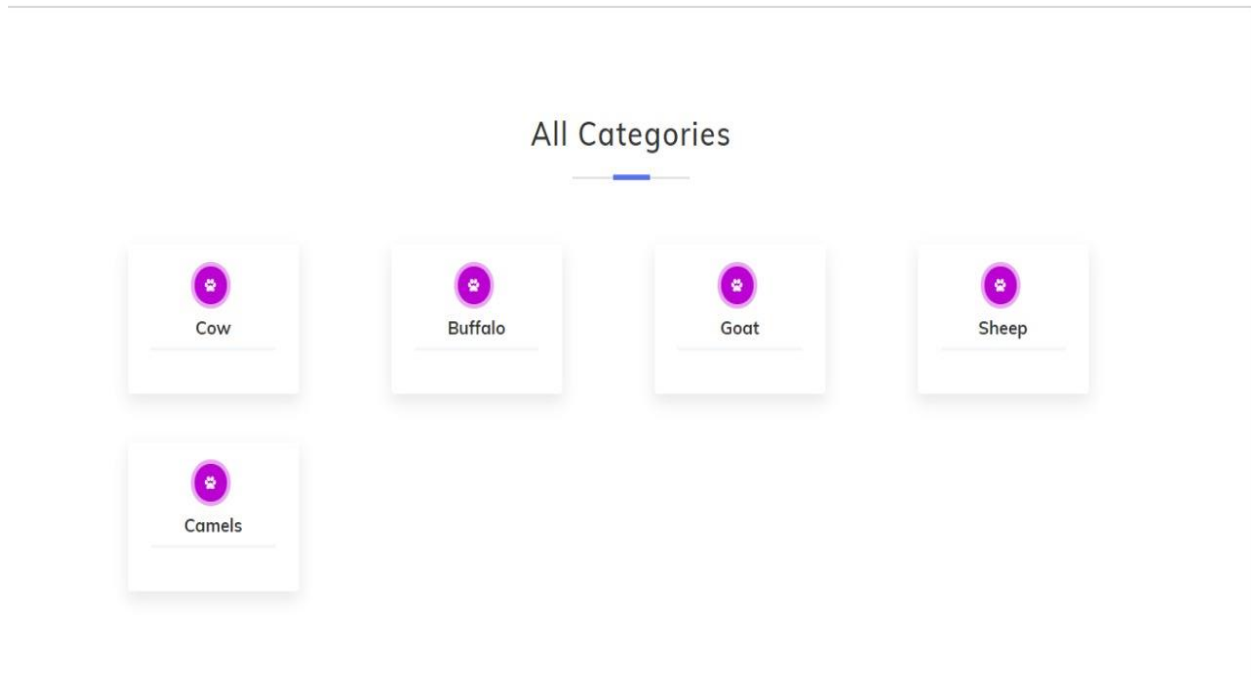
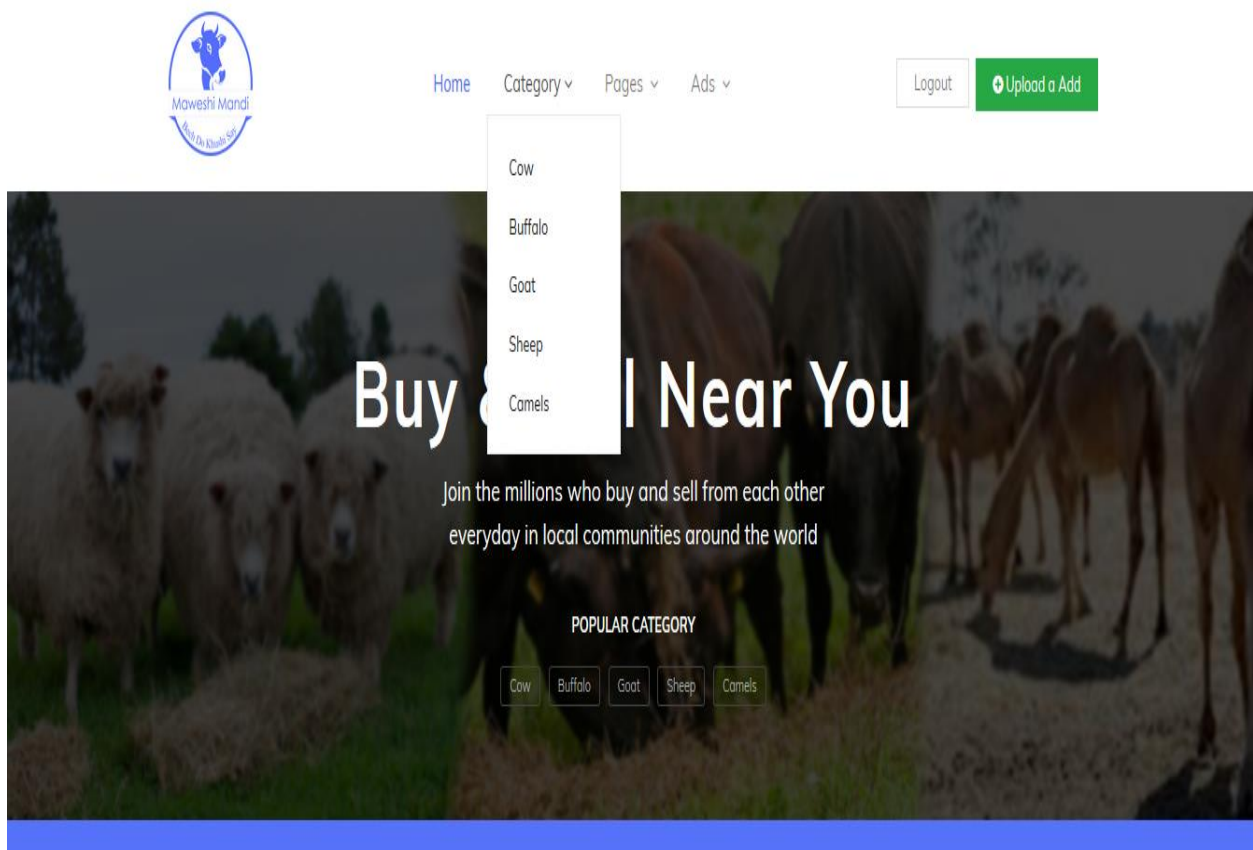
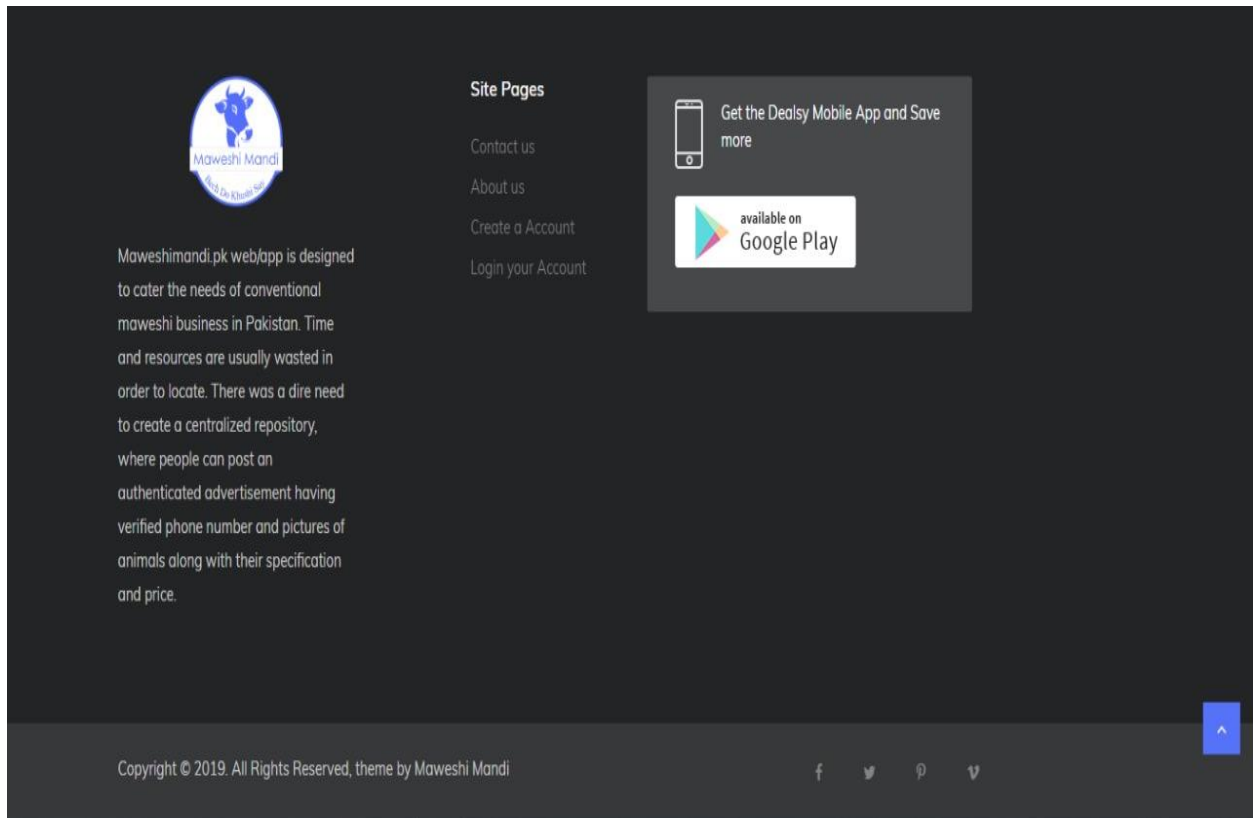
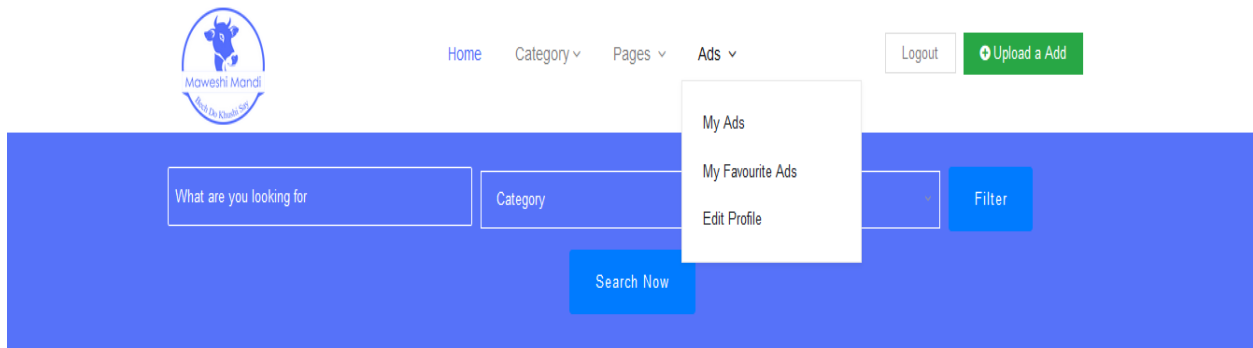
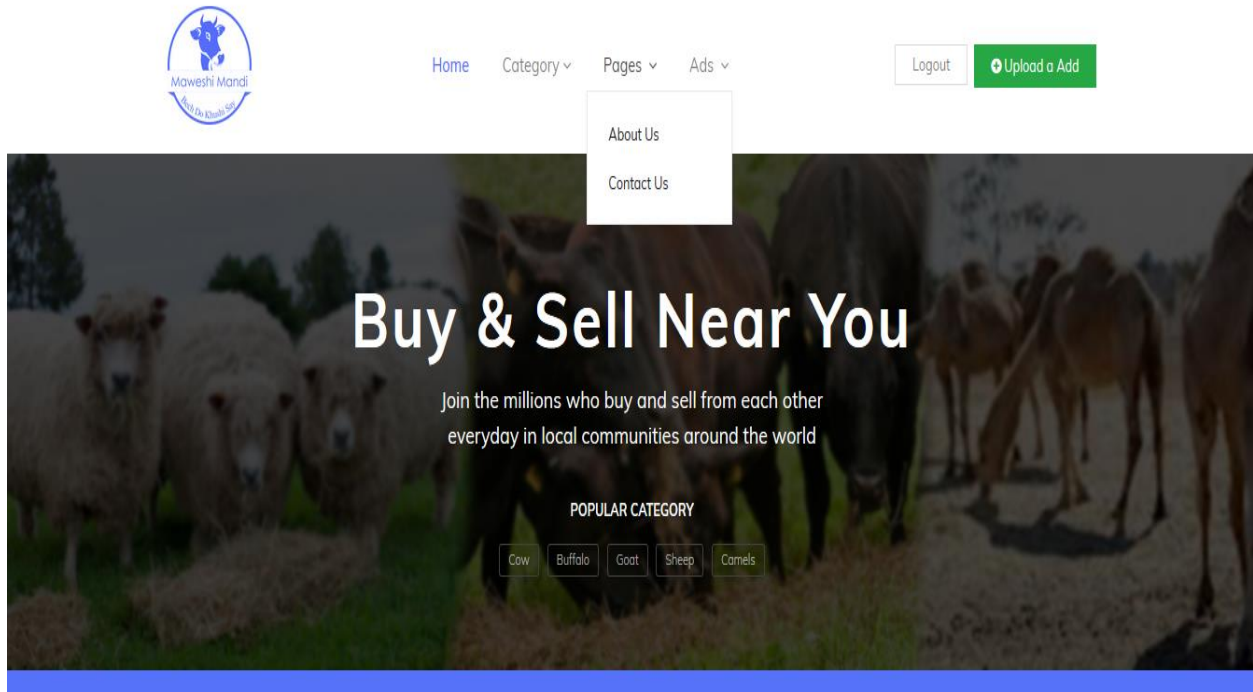


Figure 5 Post Ads of MM







### Create A Account

Name:

Phone:

(03XX-XXXXXXX)

## Create a New Account

Name:

Phone:  
(0300-XXXXXXX)

Email:

Password

City

Area

SUBMIT

FORGET PASSWORD



[Home](#) [Category](#) [Pages](#)

[Login](#)

[Register](#)

[Activation](#)

What are you looking for

Category

Location

Filter

Search Now

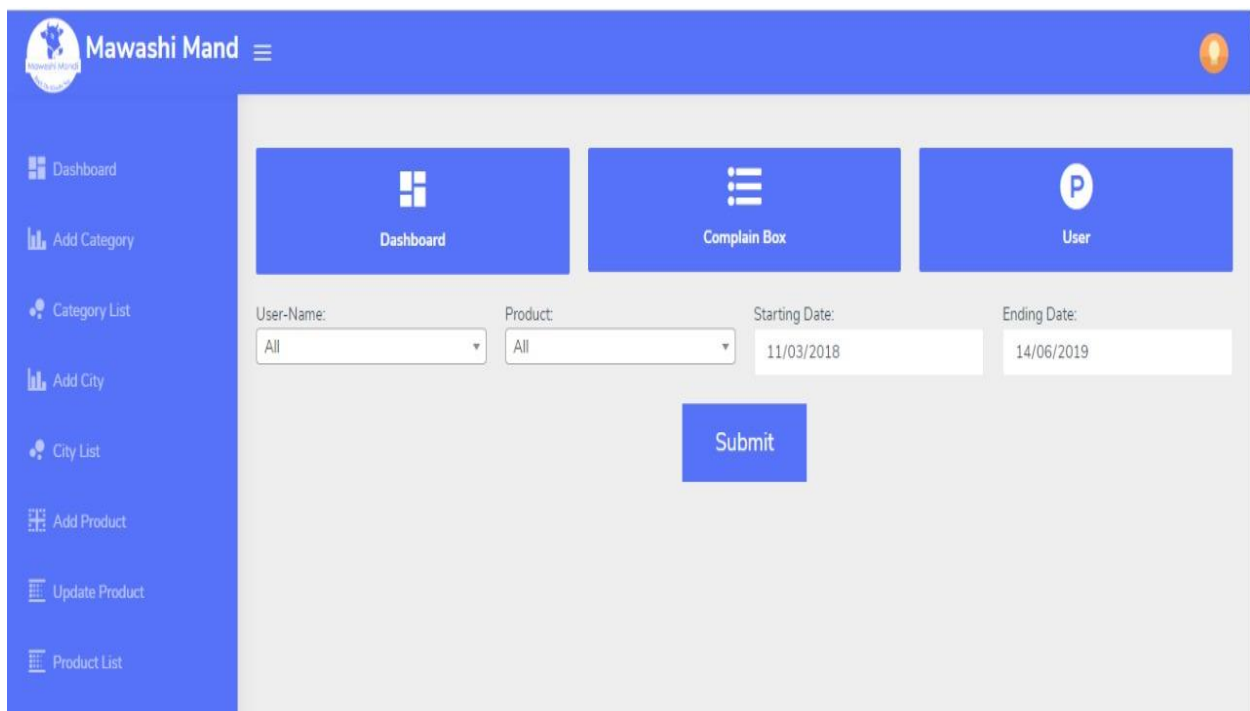
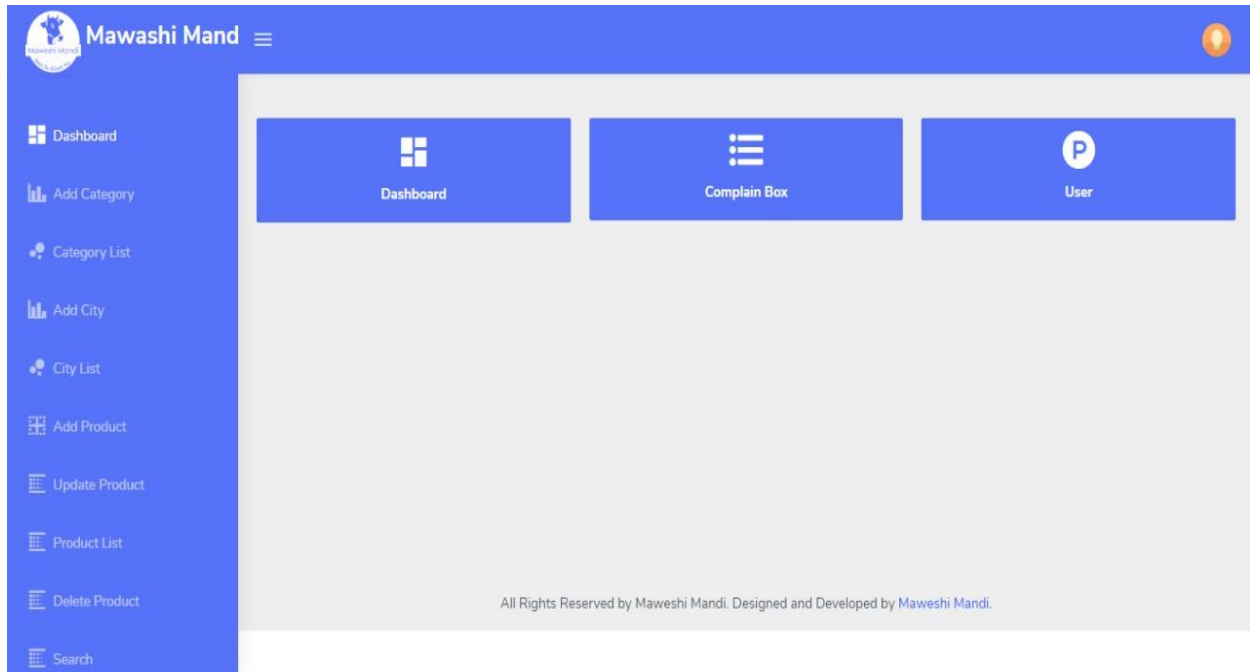
## Active your Account

Email:

User Code

SUBMIT

## Admin Page



**Mawashi Mandi**

- Dashboard
- Add Category
- Category List
- Add City
- City List
- Add Product
- Update Product
- Product List
- Delete Product
- Search

**Category List**

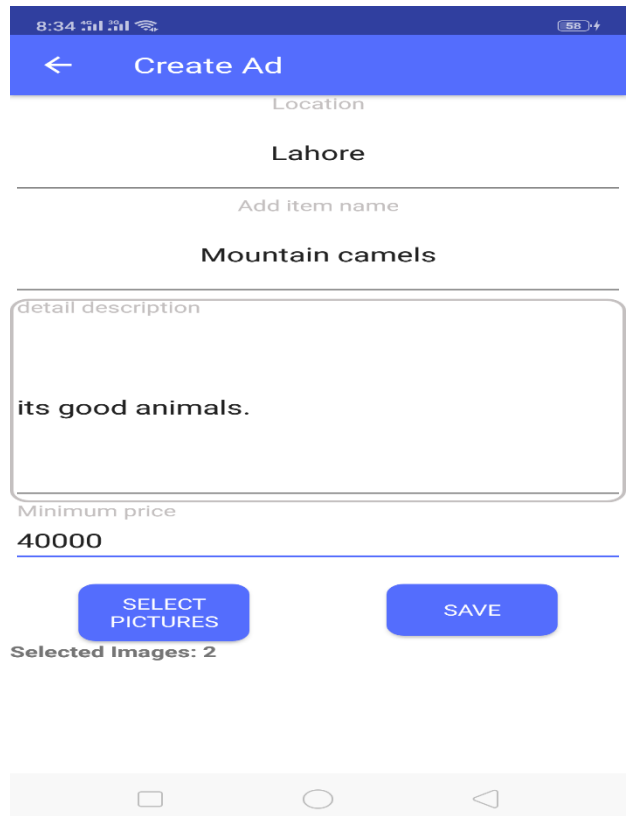
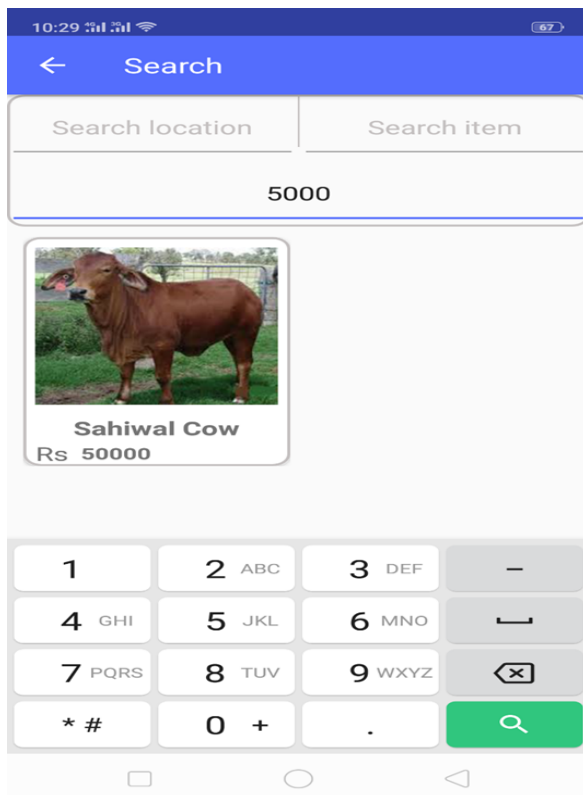
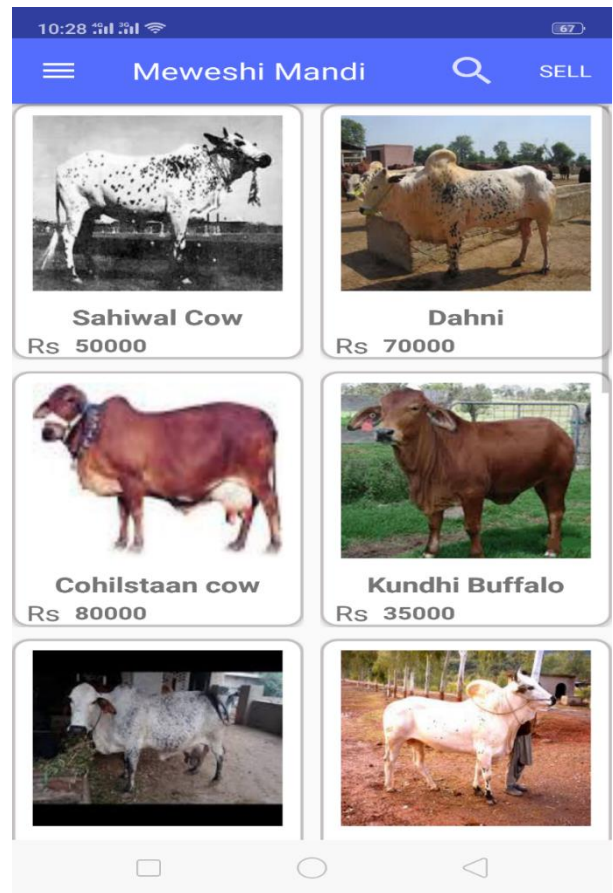
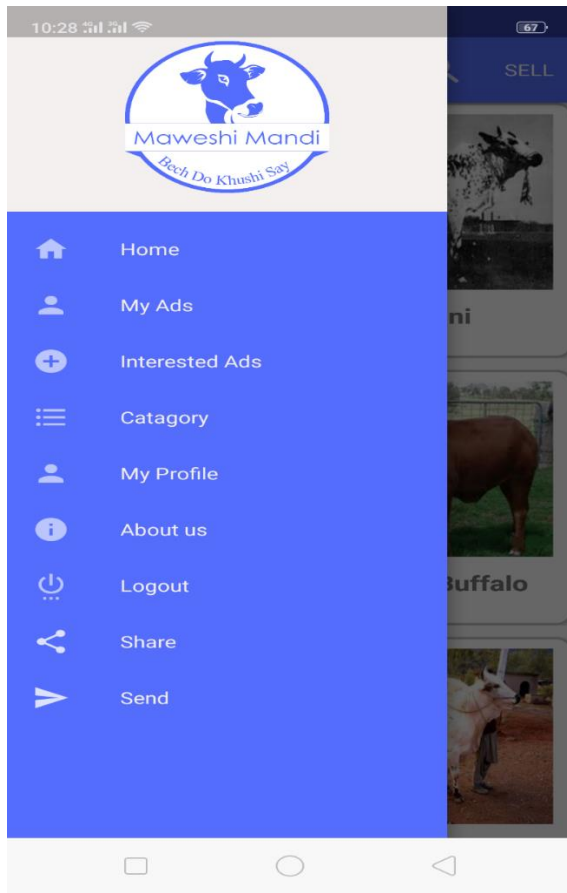
Number	Category Name	Action
1	Cow	Delete
2	Buffalo	Delete
3	Goat	Delete
4	Sheep	Delete
5	Camels	Delete

**Mawashi Mandi**

- Dashboard
- Add Category
- Category List
- Add City
- City List
- Add Product
- Update Product
- Product List
- Delete Product
- Search

**PRODUCT LIST**

Image	Category	Name	Action	Action
image	Cow	Sahiwal Cow	Update	Delete
image	Cow	Red Sindhi	Update	Delete
image	Cow	Dhanni	Update	Delete
image	Cow	Cohistani	Update	Delete
image	Buffalo	Kundi	Update	Delete



### 2.3.1. User Interfaces

Describe the logical characteristics of each interface between the software product and the users. This may include sample screen images, any GUI standards or product family style guides that are to be followed, screen layout constraints, standard buttons and functions (e.g., help) that will appear on every screen, keyboard shortcuts, error message display standards, and so on. Define the software components for which a user interface is needed. Details of the user interface design should be documented in a separate user interface specification.>

### 2.3.2. Hardware Interfaces

This is an online system so the hardware interface that are going to be used are the hardware components that are required for the internet access. These have been mentioned in the following table.

<b>WLAN</b>	<b>WAN</b>	<b>LAN</b>	<b>Ethernet cross-cable</b>	<b>Modem</b>	<b>Router</b>
-------------	------------	------------	-----------------------------	--------------	---------------

*Table 2-0-3 Hardware Interfaces*

### 2.3.3. Software Interfaces

#### External machine interfaces

External Machine interface is not required since computation will be done on Windows OS and Android OS.

#### External system interfaces

This will only include the MySQL database for the products

#### Human interface

The software will use a Graphical User Interface coded in HTML, CSS and JavaScript

### 2.3.4. Communications Interfaces

- Client (Seller) on Internet will be using HTTP/HTTPS protocol.
- Client (Customer) on Internet will be using HTTP/HTTPS protocol.

## 2.4. System Features

The system is going to consist of multiple modules, each separately developed with their own features.

**1. The first module is admin panel:**

- a. It will monitor the activities of users and verify accounts.
- b. It will also make decisions (via graphs and notifications)
- c. Another feature is getting notified about various issues or important new e.g. too many reports on the user.

**2. Seller panel module is going to handle all of seller's details and actions:**

- a. The seller can add products with their description in the defined schema as required by the website.
- b. Give feedback to the users he interacted with and give a damage report.
- c. He can also view all add

**3. Visitor panel can do a variety of things:**

Visitors can view products according to city, categories, price range and performs registration.

**4. Another feature called "Genuine Products" will ensure users uploads the actual pictures of their product instead of some image they found on google.**

**5. Our Intelligent reporting feature is for administrator to see how and where the website is being used and the products are being sold. And for creating research data.**

**6. User can create secure account**

## 2.4.1. Admin Panel

### 2.4.1.1. Description and Priority

This feature describes admin panel. It includes monitor the user's activities, verifying user accounts, manipulate add (add, update & delete), verifying add and product.

<b>Identifier</b>	SF-001
<b>Title</b>	Admin Panel
<b>Dependencies</b>	REQ-SF1-1, REQ-SF1-2, REQ-SF1-3, REQ-SF1-4, REQ-SF1-5, REQ-SF1-6, REQ-SF1-7, REQ-SF1-8, REQ-SF1-9, REQ-SF1-10, REQ-SF1-11, REQ-SF1-12, REQ-SF1-13.
<b>Priority</b>	High

Table 2-0-4 Admin Panel Description and Priority

### 2.4.1.2. Stimulus/Response Sequences

<b>Actor Action</b>	<b>System Response</b>
Admin request for login in admin panel.	System will check fields and generate the page.
Admin click on seller adds to check post details.	System will generate seller adds page.
Admin click on users to check seller detail.	System will generate user's information page.
Admin click on seller adds to check adds detail.	System will generate seller adds page.
Admin click accept or reject button for user account.	System will approve or disapprove user account on admin behavior.
Admin click on graph to check users visits.	System will generate a graph of users visits on website.
Admin click on add or remove or update for specific add.	System will add or update or remove product on admin action.
Admin click on categories to check Product categories.	System will generate categories detail page.

Admin click on add or remove or update for specific category.	System will add or update or remove category on admin action.
---	---

Table 2-0-5 Admin Panel Stimulus/Response Sequences

### 2.4.1.3. Functional Requirements

Function requirements for this feature are following:

- REQ-SF1-1: System do verify user.
- REQ-SF1-2: System do de-activate user account.
- REQ-SF1-3: System does add seller post.
- REQ-SF1-4: System does update seller post.
- REQ-SF1-5: System does remove rented post.
- REQ-SF1-6: System does add category.
- REQ-SF1-7: System does update category.
- REQ-SF1-8: System does remove category.
- REQ-SF1-9: System does show all products.
- REQ-SF1-10: System does show all users information.
- REQ-SF1-11: System do show all categories.
- REQ-SF1-12: System do show all graphs.

## 2.4.2. Seller Panel

### 2.4.2.1. Description and Priority

<b>Identifier</b>	SF-002
<b>Title</b>	Seller Panel
<b>Dependencies</b>	REQ-SF2-1, REQ-SF2-2, REQ-SF2-3, REQ-SF2-4, REQ-SF2-5, REQ-SF2-6, REQ-SF2-7, REQ-SF2-8, REQ-SF2-9, REQ-SF2-10.
<b>Priority</b>	High

Table 2-0-6 Seller Panel Description and Priority

### 2.4.2.2. Stimulus/Response Sequences

Actor Action	System Response
Seller request for login in admin panel.	System will check fields and generate the page.
Seller upload picture of product & detail and click to post add.	System will verify add data and then post it.
Seller change the account setting and click on update button.	System will verify user new information then change the account setting.
Seller click on MyAdds to check his adds detail.	System will generate MyAdds page.
Seller click on seller items to check seller items detail.	System will generate seller items page.
Renter can update add by re-submitting the add data.	System will verify the data then update the add.
Seller click on remove button to delete add.	System will remove the add.
Seller click on dashboard to check his account detail.	System will generate dashboard page.
Seller click on submit to submit product report.	System will save the product report.

Table 2-0-7 Seller Panel Stimulus/Response Sequences

### 2.4.2.3. Functional Requirements

Function requirements for this feature are following:

- REQ-SF2-1: System does Sign up
- REQ-SF2-2: System do login.
- REQ-SF2-3: System does add seller post.
- REQ-SF2-4: System does update seller post.
- REQ-SF2-5: System does remove seller post.
- REQ-SF2-6: System does update account detail.
- REQ-SF2-7: System does show all seller adds information.

REQ-SF2-9: System do show seller items detail currently and previously.

REQ-SF2-10: System do show chart of seller items.

REQ-SF2-10: System do submit product report.

### 2.4.3. User Panel

#### 2.4.3.1. Description and Priority

This feature describes user panel. It includes search product via city, price range and categories see seller view and filter product according to condition.

<b>Identifier</b>	SF-003
<b>Title</b>	User Panel
<b>Dependencies</b>	REQ-SF3-1, REQ-SF3-2, REQ-SF3-3, REQ-SF3-4, REQ-SF3-5, REQ-SF3-6.
<b>Priority</b>	High

*Table 2-0-8 Costumer Panel Description and Priority*

#### 2.4.3.2. Stimulus/Response Sequences

<b>Actor Action</b>	<b>System Response</b>
Consumer click on search button to search product.	System will generate list of searched products.
Consumer click on filter to filter product acc. to price, city & categories.	System will filter searched products according to consumer action.
Consumer click on product add to check product detail.	System will generate product detail page.

*Table 2-0-9 Costumer Panel Stimulus/Response Sequences*

#### 2.4.3.3. Functional Requirements

Function requirements for this feature are following:

REQ-SF3-1: System do search products.

REQ-SF3-2: System do filter product search according to city, price range and categories.

REQ-SF3-3: System does show product details.

REQ-SF3-6: System does show safety disclaimers.

## 2.4.4. Genuine Animals

### 2.4.4.1. Description and Priority

This feature ensure users uploads the actual pictures of their product instead of some image they found on Google.

<b>Identifier</b>	SF-004
<b>Title</b>	Genuine Animals
<b>Dependencies</b>	REQ-SF4-1, REQ-SF4-2, REQ-SF4-3, REQ-SF4-4.
<b>Priority</b>	High

*Table 2-0-10 Genuine Animals Description and Priority*

### 2.4.4.2. Stimulus/Response Sequences

<b>Actor Action</b>	<b>System Response</b>
Renter click on login button to his account.	System will generate renter account page.
Renter click on upload button to upload product picture and then submit add.	System will verify product picture if it is genuine then upload otherwise not.

*Table 2-0-11 Genuine Animals Stimulus/Response Sequences*

### 2.4.4.3. Functional Requirements

Function requirements for this feature are following:

REQ-SF4-1: System do login.

REQ-SF4-2: System do upload product pictures.

REQ-SF4-3: System does post add.

REQ-SF4-4: System does verify add detail and pictures of product.

## 2.4.5. Intelligent Reporting

### 2.4.5.1. Description and Priority

This feature describes that how administrative can see that where and how website is being used and the products are being sold.

<b>Identifier</b>	SF-003
<b>Title</b>	Intelligent Reporting
<b>Dependencies</b>	REQ-SF5-1, REQ-SF5-2, REQ-SF5-3.
<b>Priority</b>	High

*Table 2-0-12 Intelligent Reporting Description and Priority*

### 2.4.5.2. Stimulus/Response Sequences

<b>Actor Action</b>	<b>System Response</b>
Admin click to login administrative account.	System will generate the dashboard page.
Admin click on reports to check how website is being used and how products are being sold.	System will generate report of time.

*Table 2-0-13 Intelligent Reporting Stimulus/Response Sequences*

### 2.4.5.3. Functional Requirements

Function requirements for this feature are following:

REQ-SF5-1: System do login.

REQ-SF5-2: System do show reports of how website is being used.

REQ-SF5-3: System does show reports that how product on sell.

## 2.4.6. CRUD Operations

### 2.4.6.1. Description and Priority

This feature describes that CRUD operation will be done on all entities like product add, user registration and product categories.

<b>Identifier</b>	SF-003
<b>Title</b>	CRUD Operations
<b>Dependencies</b>	REQ-SF6-1, REQ-SF6-2, REQ-SF6-3.
<b>Priority</b>	High

*Table 20-14 CRUD Operations Description and Priority*

### 2.4.6.2. Stimulus/Response Sequences

<b>Actor Action</b>	<b>System Response</b>
Admin click to login administrative account.	System will generate the dashboard page.
Admin click on update or add or delete to manipulate the product add.	System will add or update or delete the product add according to admin action.
Admin click on update or add or delete to manipulate the product category.	System will add or update or delete the product category according to admin action.
Seller click to login his account.	System will generate the dashboard page.
Seller click on update or add or delete to manipulate the product add.	System will add or update or delete the product add according to admin action.

*Table 2-0-15 CRUD Operations Stimulus/Response Sequences*

### 2.4.6.3. Functional Requirements

Function requirements for this feature are following:

REQ-SF6-1: System do login.

REQ-SF6-2: System do manipulate product adds.

REQ-SF6-3: System do manipulate product categories.

## 2.4.7. Secure User Registration & Login/Logout

### 2.4.7.1. Description and Priority

This feature describes the secure registration and login/logout of renters.

<b>Identifier</b>	SF-003
<b>Title</b>	Secure user registration & Login/Logout
<b>Dependencies</b>	REQ-SF7-1, REQ-SF7-2, REQ-SF7-3.
<b>Priority</b>	High

*Table 2-0-16 Secure User Registration & Login/Logout Description and Priority*

### 2.4.7.2. Stimulus/Response Sequences

<b>Actor Action</b>	<b>System Response</b>
Seller fill the required fields and click on signup button.	System will verify user data and then register him as a seller and generate his account page.
Seller fill required fields and click on login button.	System will verify user information then login to his account.
Admin click on logout button.	System will logout Seller account.

*Table 2-0-17 Secure User Registration & Login/Logout Description and Priority*

### 2.4.7.3. Functional Requirements

Function requirements for this feature are following:

REQ-SF7-1: System do sign up.

REQ-SF7-2: System do login.

REQ-SF7-3: System do logout.

## 2.4.8. User & other Users Search Animal Filter

### 2.4.8.1. Description and Priority

This feature describes the secure registration and login/logout of renters.

<b>Identifier</b>	SF-003
<b>Title</b>	User & other users search animal Filter
<b>Dependencies</b>	REQ-SF8-1, REQ-SF8-2.
<b>Priority</b>	High

*Table 2-0-18 User & other Users Search Animal Filter Description and Priority*

### 2.4.8.2. Stimulus/Response Sequences

<b>Actor Action</b>	<b>System Response</b>
Consumer write the name of product in search bar and click search button.	System will generate the list of searched products.
Consumer click on filter to filter searched products according to city, price range and category.	System will filter the products according to consumer action.

*Table 2-0-19 User & other Users Search Animal Filter Stimulus/Response Sequences*

### 2.4.8.3. Functional Requirements

Function requirements for this feature are following:

REQ-SF9-1: System do search product.

REQ-SF9-2: System do filter the search products according to city, price and category.

## 2.5. Other Nonfunctional Requirements

### 2.5.1. Performance Requirements

- **Response time:** The home page loaded in 3-5 seconds.
- **Availability:** The site will available 97.99% of the time.
- **Throughput:** Page hits per second or orders per hour.
- **Capacity:** Maximum throughput that can be achieved with acceptable response time and availability.

### 2.5.2. Safety Requirements

- Once admin delete product adds or product categories or user account the data of that product add can't be retrieve. To handle this problem system gives warning before delete the product.
- Safety disclaimers will be given for both renter and consumer.
- If product picture is not original than product add will not uploaded and waning will be given to the user.
- If any user has some problem while using the site, so he can contact via mail to website maintainers.

### 2.5.3. Security Requirements

- Security is not free, but it's cheaper than getting hacked.
- This software keeps the information of renters safe.
- User will be registered after verification of user data.
- Secure login and logout. If user forget his account credential than he has to get new credential through his sign-up e-mail.
- Admin area will be secured.

### 2.5.4. Software Quality Attributes

- **Usability:** This website easy to use and easy to learn how to operate.

- **Correctness:** When user search something but spellings are not right the system search related to that spelling or give autocomplete search.
- **Portability:** This product can be used on different devices like tablets, mobiles, laptops etc.
- **Robustness:** This software can handle many problems itself like if user forgets his password so system gives it option to re-write name and user password.
- **Flexibility:** The functionality of this software can be increased after development because documentation is available.

### 2.5.5. Business Rules

- If user want to post add than he has to first sign up.
- Website will not responsible for raw data of product.
- User has to follow terms and conditions of website.

## 2.6. Other Requirements

- **Usability:** This website easy to use and easy to learn how to operate.
- **Correctness:** When user search something but spellings are not right the system search related to that spelling or give autocomplete search.
- **Portability:** This software can be used on different devices like tablets, mobiles, laptops etc.
- **Robustness:** This software can handle many problems itself like if user forgets his password so system gives it option to re-write name and user password.
- **Flexibility:** The functionality of this software can be increased after development because documentation is available.

# Chapter 3

## Use Case Analysis

### **Chapter 3: System Analysis**

This chapter describe about system use cases and how the different actors are interacting with the system. In use case main functionalities of the system are defined that are divided in different modules e.g. major use case in which all functionalities of the system are defined is divide into Customer, consumer and admin module and each actor interacting with module and performing his module functionalities.

### 3.1 Use Case Mode

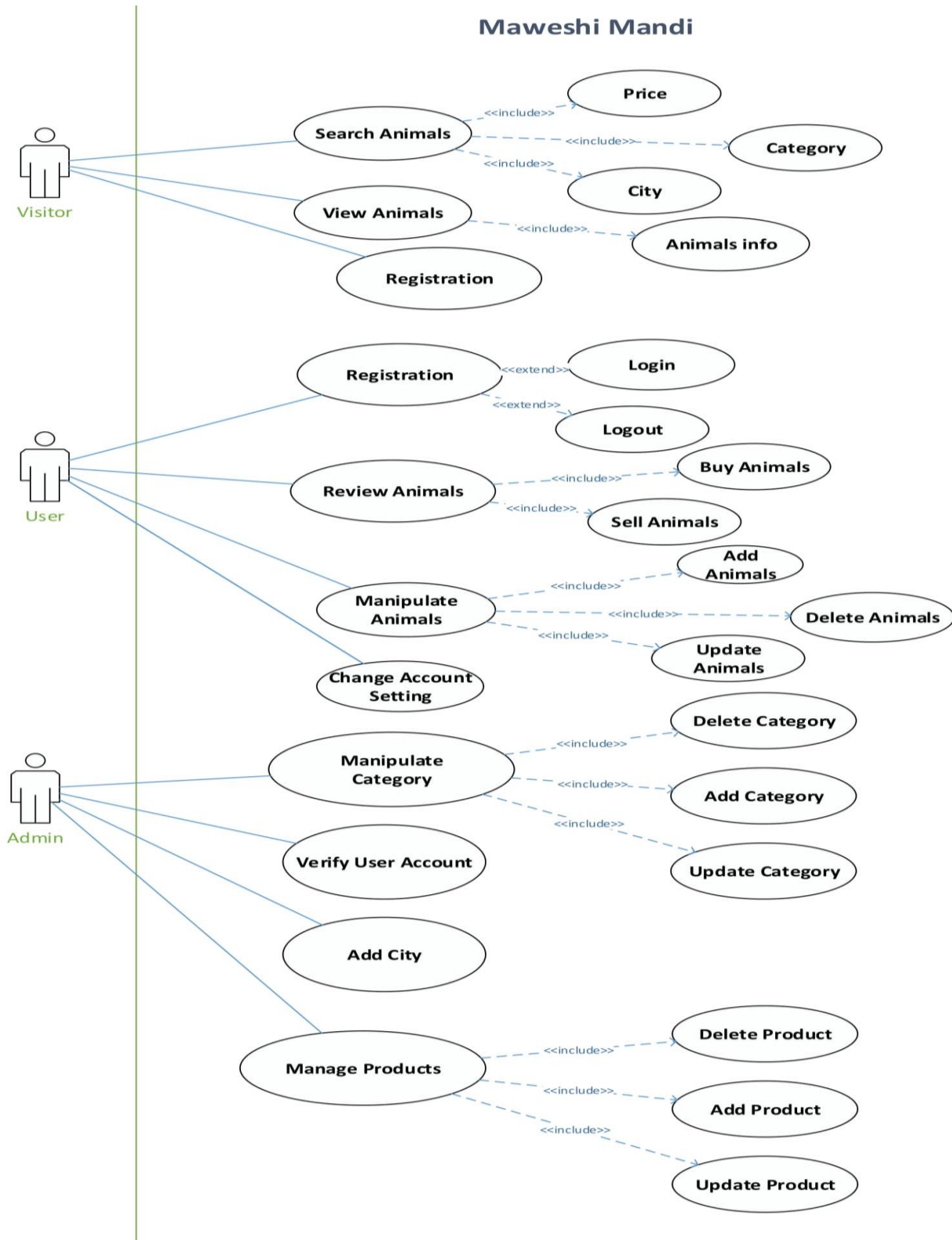


Figure 6 Use Case Model

### 3.1.1 Visitor Use Case:

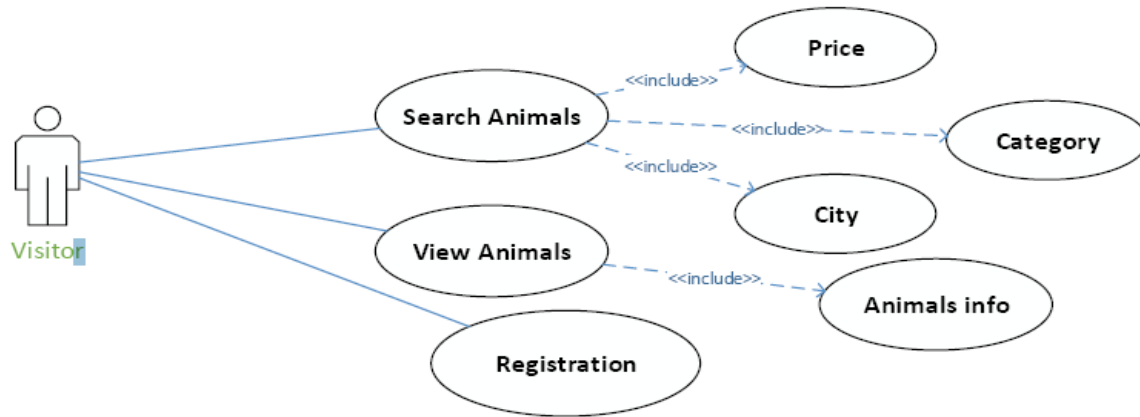


Figure 6 Visitor Use Case

### 3.1.2 User Use Case:

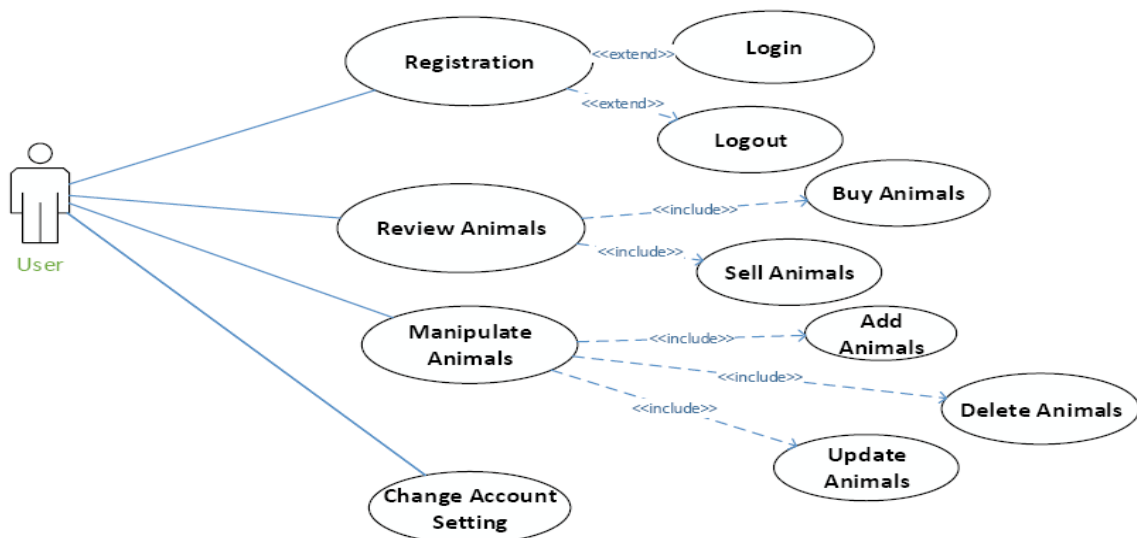


Figure 7 User Use Case

### 3.1.3 Admin Use Case:

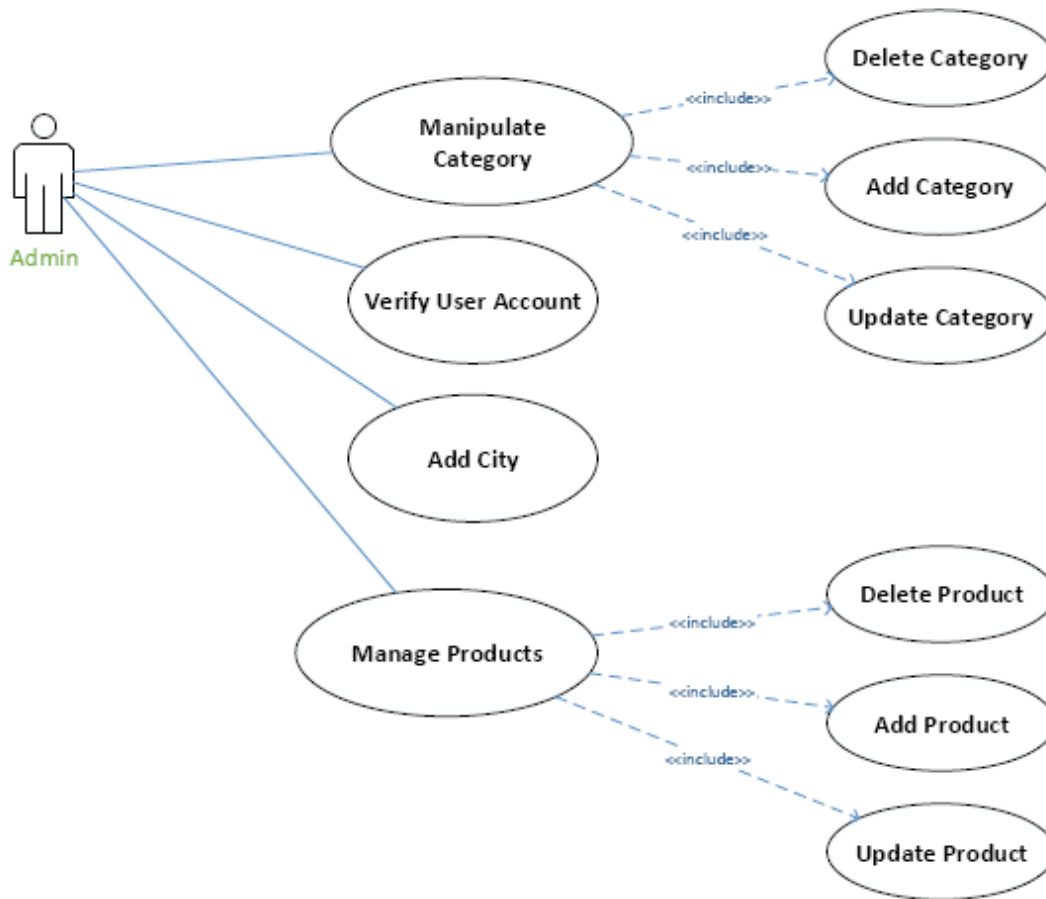
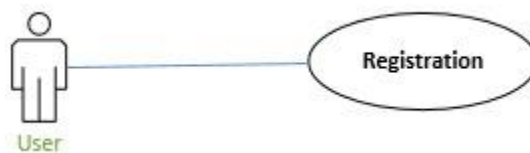


Figure 8 Admin Use Case

### 3.2 Fully Dressed Use Cases



Fully dressed use case

1.	<b>Use case name</b>	<b>Registration</b>
2.	<b>Scope</b>	Registration in the System
3.	<b>Level</b>	User goal (Seller/Buyer)
4.	<b>Primary Actor</b>	Customer
5.	<b>Stakeholder</b>	Consumer/customer & Admin
6.	<b>Precondition</b>	Null
7.	<b>Post condition</b>	Add in the System
8.	<b>Success Scenario</b>	Registered successfully
9.	<b>Extension</b>	View login task.

Table 3-0-1 Registration



Fully dressed use case		
1.	Use case name	Login
2.	Scope	Open all information of system pages
3.	Level	User goal (Customer)
4.	Primary Actor	Customer
5.	Stakeholder	Consumer/customer & Admin
6.	Precondition	Check it is registered or not
7.	Post condition	Login to system
8.	Success Scenario	Login to system successfully
9.	Extension	View the system as user perspective.

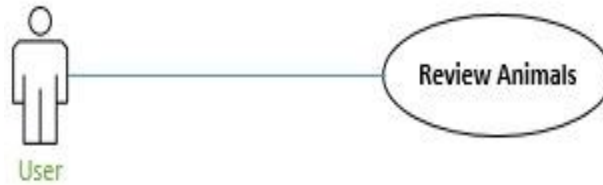
Table 3-0-2 Login



Fully dressed use case		
1.	Use case name	Logout
2.	Scope	Not Open all information of system pages
3.	Level	User goal (Customer)
4.	Primary Actor	Customer
5.	Stakeholder	Consumer/customer & Admin

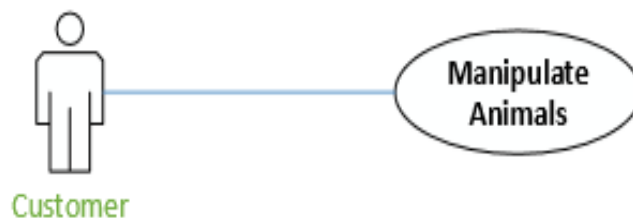
6.	<b>Precondition</b>	Check it is login or not
7.	<b>Post condition</b>	Null
8.	<b>Success Scenario</b>	Logout from the system successfully
9.	<b>Extension</b>	Exit from the System

Table 3-0-3 Logout



<b>Fully dressed use case</b>		
1.	<b>Use case name</b>	<b>Review Animal</b>
2.	<b>Scope</b>	View and Open all information of system pages/post
3.	<b>Level</b>	User goal (User)
4.	<b>Primary Actor</b>	User
5.	<b>Stakeholder</b>	Consumer/customer & Admin
6.	<b>Precondition</b>	Check it is login or not
7.	<b>Post condition</b>	Post review and comment on it
8.	<b>Success Scenario</b>	Review to system successfully
9.	<b>Extension</b>	Review the System

Table 3-0-4 Review Animal



Fully dressed use case		
1.	Use case name	Manipulate Animal Adds
2.	Scope	Customer manipulate the her post add/info
3.	Level	User goal (Customer)
4.	Primary Actor	Customer
5.	Stakeholder	Consumer/customer & Admin
6.	Precondition	Check it is login or not
7.	Post condition	Manipulate her Adds
8.	Success Scenario	Manipulate Adds successfully
9.	Extension	Manipulate from the System adds

Table 3-0-5 Manipulate Animal Adds



Fully dressed use case		
1.	Use case name	Animals Seller
2.	Scope	Customer sells her Animals by posing in the system
3.	Level	User goal (Customer)
4.	Primary Actor	Customer
5.	Stakeholder	Consumer/customer & Admin
6.	Precondition	Check it is login or not
7.	Post condition	Post Adds of sells Animal
8.	Success Scenario	Post Adds successfully
9.	Extension	Add Post in the System for others users

Table 3-0-6 Animals Seller



Fully dressed use case		
1.	Use case name	Animals buyer
2.	Scope	Customer buyer Animals by other user posing in the system
3.	Level	User goal (Customer)
4.	Primary Actor	Customer
5.	Stakeholder	Consumer/customer & Admin
6.	Precondition	Check it is login or not
7.	Post condition	Buyer buy the Animal
8.	Success Scenario	Buy the Animal successfully
9.	Extension	Buyer purchasing product from System

Table 3-0-7 Animals buyer



Fully dressed use case		
1.	Use case name	Change Account Setting
2.	Scope	Customer change the account setting of an profile privacy (Password, nickname & DB)
3.	Level	User goal (Seller/Buyer)
4.	Primary Actor	Seller/Buyer
5.	Stakeholder	Consumer/customer & Admin
6.	Precondition	Check it is login or not
7.	Post condition	Change the profile setting
8.	Success Scenario	Change Account Setting successfully
9.	Extension	Personal profile setting according to the user

Table 3-0-8 Change Account Setting



Fully dressed use case		
1.	Use case name	Registration
2.	Scope	Visitor can be perform registration
3.	Level	User goal (Visitor)
4.	Primary Actor	Visitor
5.	Stakeholder	Consumer/customer & Admin

6.	<b>Precondition</b>	Check it is login or not Visitor
7.	<b>Post condition</b>	Done Registration
8.	<b>Success Scenario</b>	Registration successfully
9.	<b>Extension</b>	Done Registration

Table 3-0-9 Registration



<b>Fully dressed use case</b>		
1.	<b>Use case name</b>	<b>Search Animal</b>
2.	<b>Scope</b>	Only search and view
3.	<b>Level</b>	User goal (Visitor)
4.	<b>Primary Actor</b>	Visitor
5.	<b>Stakeholder</b>	Consumer & Admin
6.	<b>Precondition</b>	Not Check (Null)
7.	<b>Post condition</b>	Search and view post add only others users
8.	<b>Success Scenario</b>	Search and view post add successfully
9.	<b>Extension</b>	Search post adds only system

Table 3-0-10 Search Animal



<b>Fully dressed use case</b>		
1.	<b>Use case name</b>	<b>View Animals</b>
2.	<b>Scope</b>	Consumer give review about system or add post
3.	<b>Level</b>	User goal (Visitor)

4.	<b>Primary Actor</b>	Visitor
5.	<b>Stakeholder</b>	Visitor & Admin
6.	<b>Precondition</b>	Check it is login or not
7.	<b>Post condition</b>	Submitted review done
8.	<b>Success Scenario</b>	Submitted review successfully
9.	<b>Extension</b>	Only review about system

Table 3-0-11 View Animals



<b>Fully dressed use case</b>		
1.	<b>Use case name</b>	<b>Manipulate Category</b>
2.	<b>Scope</b>	Admin Manipulate Category of all types Animals (products)
3.	<b>Level</b>	Admin Goal (Admin)
4.	<b>Primary Actor</b>	Admin
5.	<b>Stakeholder</b>	Consumer/customer & Admin
6.	<b>Precondition</b>	Null
7.	<b>Post condition</b>	Manipulate /Changed the category of (Products)
8.	<b>Success Scenario</b>	Changed the category successfully
9.	<b>Extension</b>	Manipulate /Changed the category of (Products)

Table 3-0-12 Manipulate Category



Fully dressed use case		
1.	<b>Use case name</b>	<b>Verify User Account</b>
2.	<b>Scope</b>	Admin Verify User Account and check it authorized user or not
3.	<b>Level</b>	Admin Goal (Admin)
4.	<b>Primary Actor</b>	Admin
5.	<b>Stakeholder</b>	Consumer/customer & Admin
6.	<b>Precondition</b>	Admin check user authorized if it true then
7.	<b>Post condition</b>	Verified Abc etc
8.	<b>Success Scenario</b>	Verified successfully
9.	<b>Extension</b>	Admin Verify User Account and check

Table 3-0-13 Verify User Account



Fully dressed use case		
1.	<b>Use case name</b>	<b>Manage Products</b>
2.	<b>Scope</b>	Admin Check it Transection done or not
3.	<b>Level</b>	Admin Goal (Admin)
4.	<b>Primary Actor</b>	Admin
5.	<b>Stakeholder</b>	Visitor/User & Admin
6.	<b>Precondition</b>	Admin check user login done or not
7.	<b>Post condition</b>	Product Done

<b>8.</b>	<b>Success Scenario</b>	Product to ABC successfully
<b>9.</b>	<b>Extension</b>	Admin Verify User transection both sides done

Table 3-0-14 Manage Product



<b>Fully dressed use case</b>		
<b>1.</b>	<b>Use case name</b>	<b>Check Report</b>
<b>2.</b>	<b>Scope</b>	Admin Check Report/Compline to user As feedback from System
<b>3.</b>	<b>Level</b>	Admin Goal (Admin)
<b>4.</b>	<b>Primary Actor</b>	Admin
<b>5.</b>	<b>Stakeholder</b>	Consumer/customer & Admin
<b>6.</b>	<b>Precondition</b>	Admin check user login, Report to user selling/buyer Animals
<b>7.</b>	<b>Post condition</b>	Report/Compline to user Done
<b>8.</b>	<b>Success Scenario</b>	Report/Compline to user ABC successfully
<b>9.</b>	<b>Extension</b>	Admin Verify User Report/Compline user both sides done

Table 3-15 Check Report

# Chapter 4

## System Design

## Chapter 4: System Design

This chapter contains the key-diagrams and models to visually represent the system. The system design has been briefed below:

The application will follow client server architecture. This is an online application and the server will manage the delivery of most of the resources. We will also develop hybrid mobile application. Since they are hybrid, the same servers that we are using for our website, will be holding the data for mobile applications. The user will need a connection to the internet for accessing our system and all its features. In a nutshell, we'll be:

- Using Client-Server Architecture
- Developing Hybrid Application

### 4.1 Architecture Diagram:

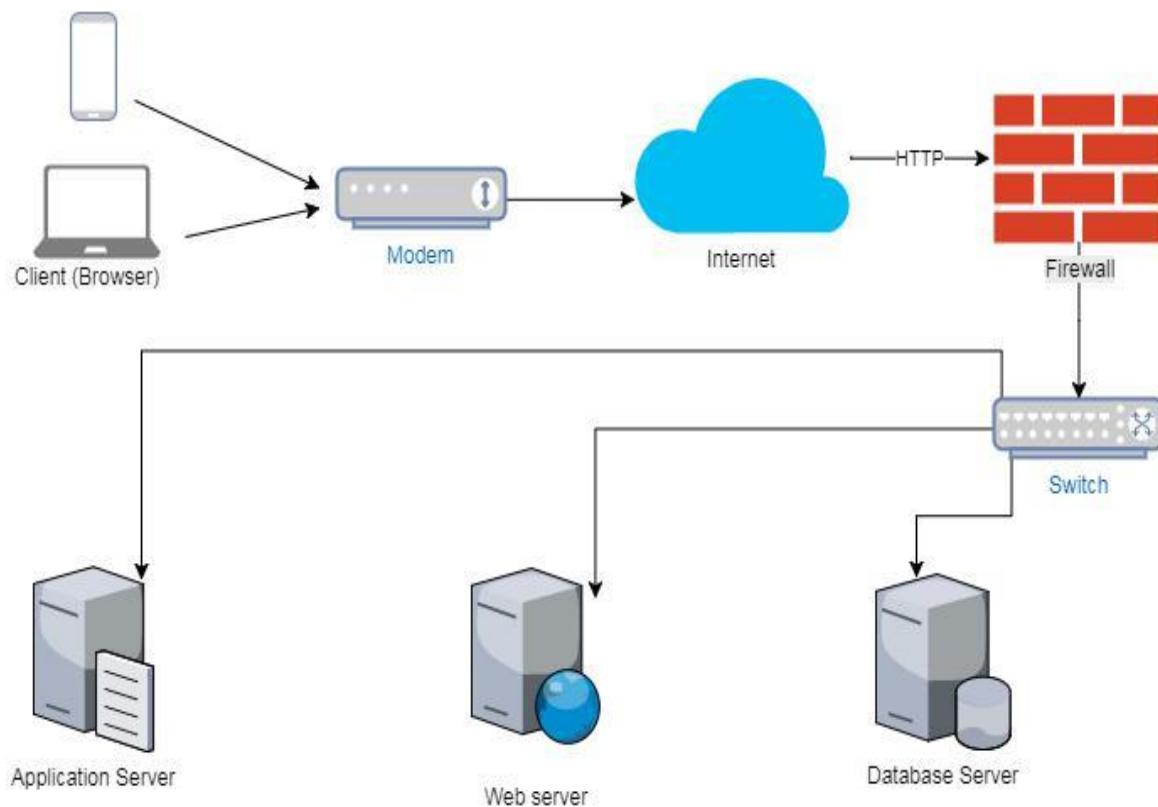


Figure 9 Architecture Diagram

## 4.2. Domain Model:

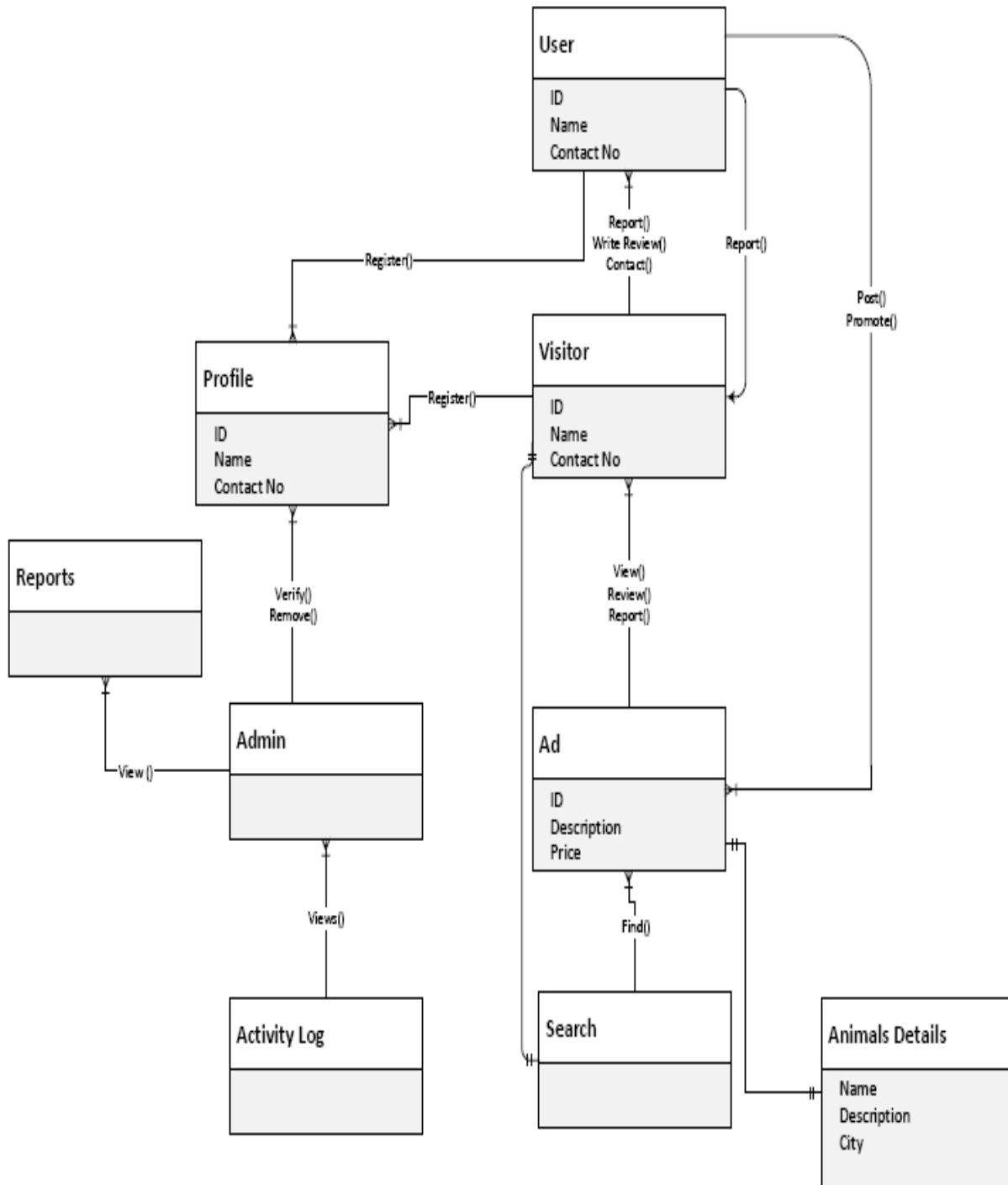


Figure 10 Domain Model

### 4.3. Entity Relationship Diagram with Data Dictionary

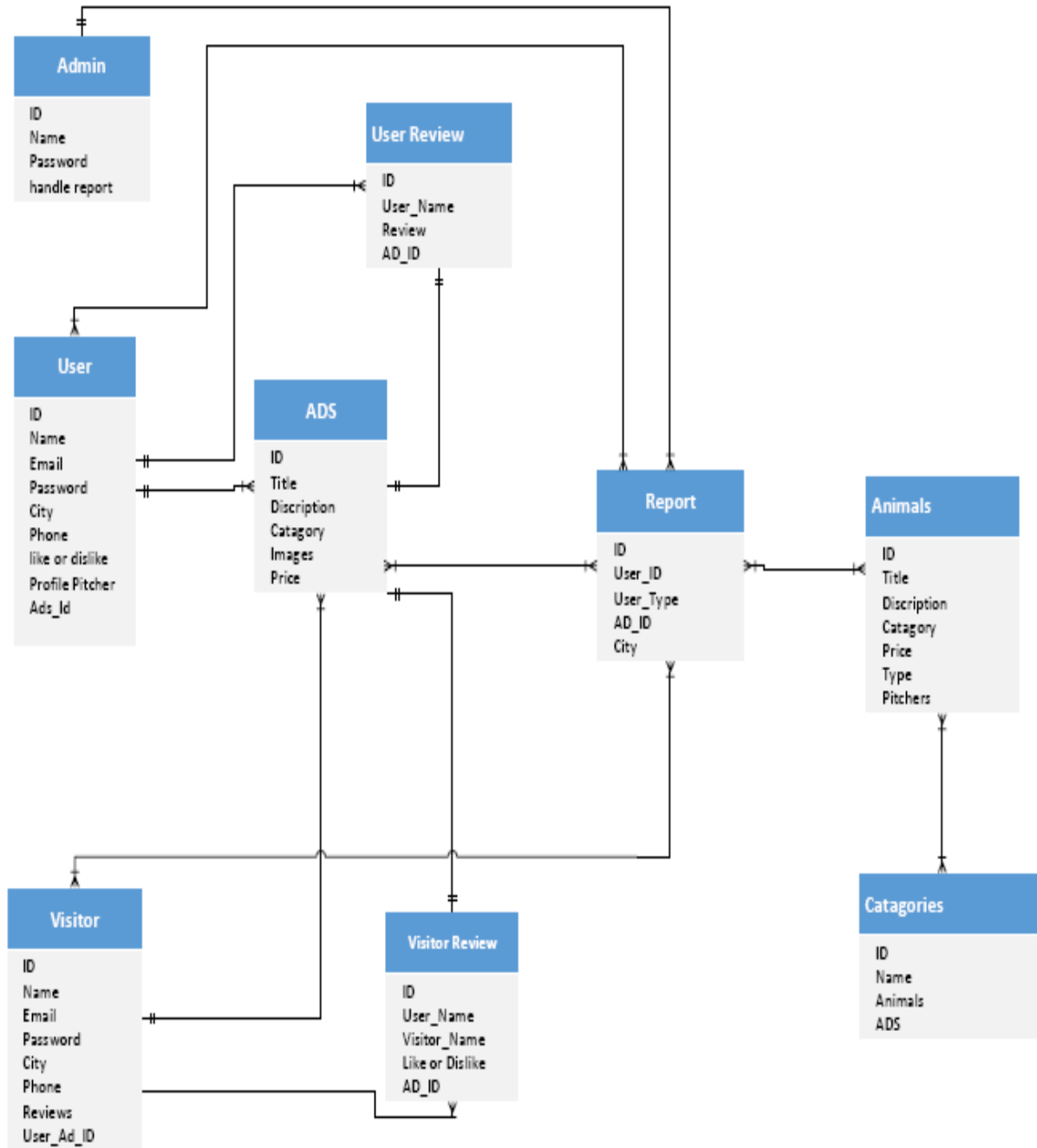


Figure 11 Entity Relationship Diagram with Data Dictionary

<b>Admin</b>			
<b>Attribute</b>	<b>Data Type</b>	<b>Nullable</b>	<b>Description</b>
ID	int	No	The identity number of this entity
Name	varchar		The name of this entity
Email	varchar		The email address of this entity
Password	varchar		The password required to login to admin account
Verification Code	varchar		Code needed to verify an account

Table 4-0-1 Admin

<b>User</b>			
<b>Attribute</b>	<b>Data Type</b>	<b>Nullable</b>	<b>Description</b>
ID	int	No	The identity number of this entity
Name	varchar		The name of this entity
Email	varchar		The email address of this entity
Password	varchar		The password required to login to account
City	varchar		The name of renter's city
Phone	small		Contact number of user/seller
Address	varchar		Address of user/seller
Date	datetime		Date he registered
IP_Address	varchar		IP address of the device this account was created from
Profile Picture	varchar		Profile picture of user/seller

Table 4-0-2 User

<b>Visitor</b>			
<b>Attribute</b>	<b>Data Type</b>	<b>Nullable</b>	<b>Description</b>
ID (PK)	int	No	The identity number of this entity

Name	varchar		The name of this entity
Address	varchar		Address of Visitor
Email	Email		The email address of this entity
Password	varchar		The password required to login to account
City	varchar		Name of consumer's city
Phone	small		Contact number of Visitor
Date	datetime		Date of registration

Table 4-0-3 Visitor

<b>Posted Ads</b>			
<b>Attribute</b>	<b>Data Type</b>	<b>Nullable</b>	<b>Description</b>
ID	int	No	The identity number of this entity
Title	varchar		The title of the ad
Description	varchar		The detailed description of the ad
animals(s)	varchar		The animals being for sells
Category(s)	varchar		Categories of the animals
Tags	varchar		Tags related to ad
Images	varchar		Images of the animals
Price	small		Price of the animals
Customer	varchar		The poster of the ad
Date	datetime		When the ad was posted

Table 4-0-4 Posted Ads

<b>Like/unlike Ads</b>			
<b>Attribute</b>	<b>Data Type</b>	<b>Nullable</b>	<b>Description</b>
ID	Int	No	The identity number of the Like/unlike ad
Title	Varchar		Title of the Like/unlike ad
Description	Varchar		Description of the promoted ad
Animal(s)	Varchar		animals being to sell

Category(s)	Varchar		Category of the animals
-------------	---------	--	-------------------------

Table 4-0-5 Promoted Ads

<b>Report</b>			
<b>Attribute</b>	<b>Data Type</b>	<b>Nullable</b>	<b>Description</b>
ID	Int	No	The identity number of this entity
Name	Varchar		Name of the reporter
Reporter Type	Varchar		Reporter Type (Customer, consumer, visitor)
Category	Varchar		Category of the report

Table 4-0-6 Report

<b>Category</b>			
<b>Attribute</b>	<b>Data Type</b>	<b>Nullable</b>	<b>Description</b>
ID	Int	No	The identity number of this entity
Animal	Varchar		Animals belonging to the category
Brand	Varchar		Brands belonging to the category

Table 4-0-7 Category

<b>Brand</b>			
<b>Attribute</b>	<b>Data Type</b>	<b>Nullable</b>	<b>Description</b>
ID	Int	No	The identity number of this entity
Animals	Varchar		Products belonging to the Brand
Categories	Varchar		Categories this brand provides

Table 4-0-8 Brand

<b>Animals</b>			
<b>Attribute</b>	<b>Data Type</b>	<b>Nullable</b>	<b>Description</b>
ID	int	No	The identity number of this entity
Title	varchar		The name of this entity
Description	varchar		Description/specs of animal
Category	varchar		Category of animal

Price	varchar		The price of the animal
Tags	varchar		Tags related to this animal
Brand	varchar		Brand this animal belongs to
Status	varchar		Available for sell
Date	datetime		Date this animal was registered
Condition	small		Condition (out of 10)
Pictures	varchar		Pictures of the animal

Table 4-0-9 Animals

<b>Safety Tips</b>			
<b>Attribute</b>	<b>Data Type</b>	<b>Nullable</b>	<b>Description</b>
ID (PK)	Int	No	The identity number of this entity
Safety Note	varchar		The note
Categories	varchar		The categories this note is for

Table 4-0-10 Safety Tips

### 4.4. Class Diagram

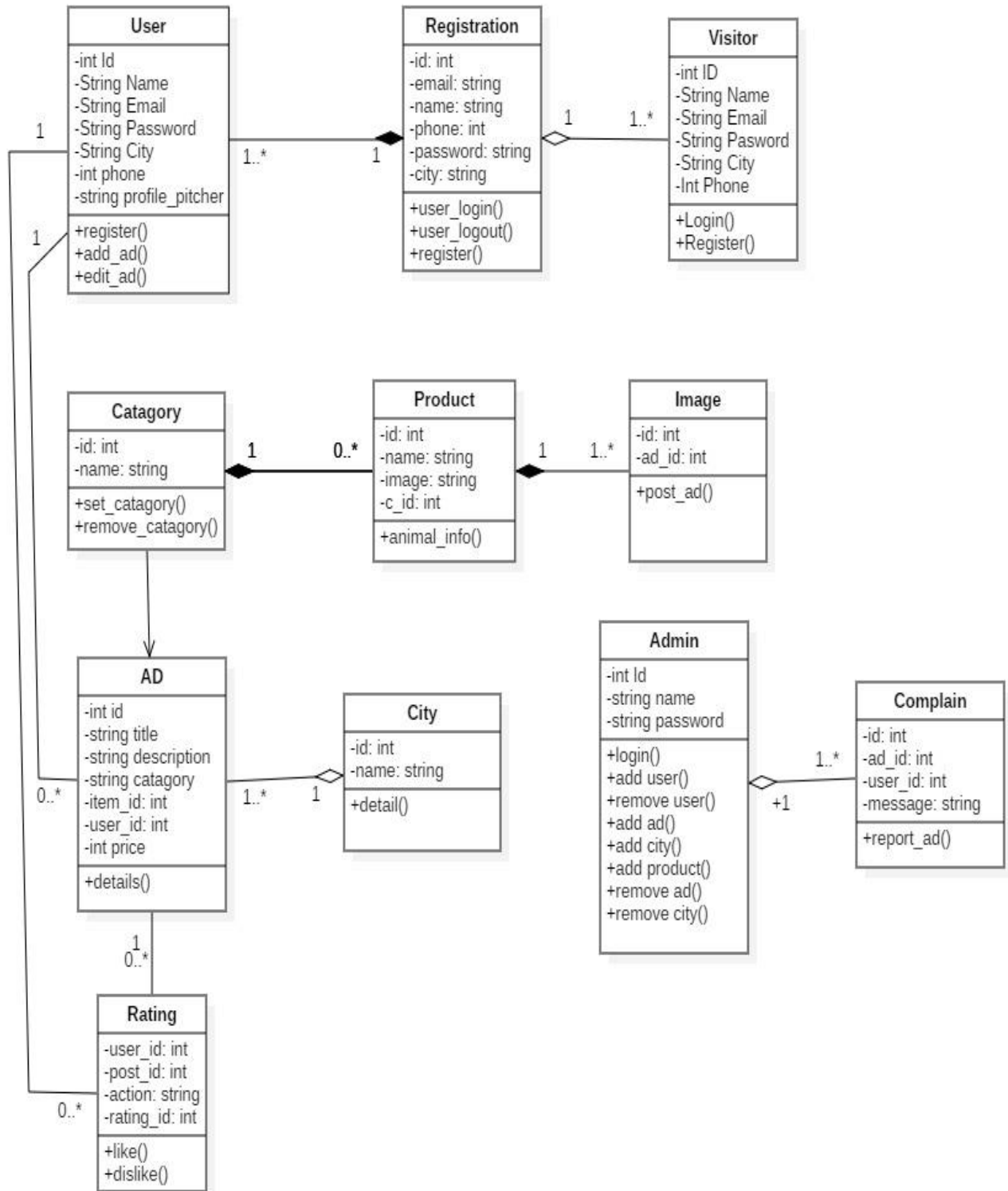


Figure 12 Class Diagram

### 4.5. Sequence / Collaboration Diagram

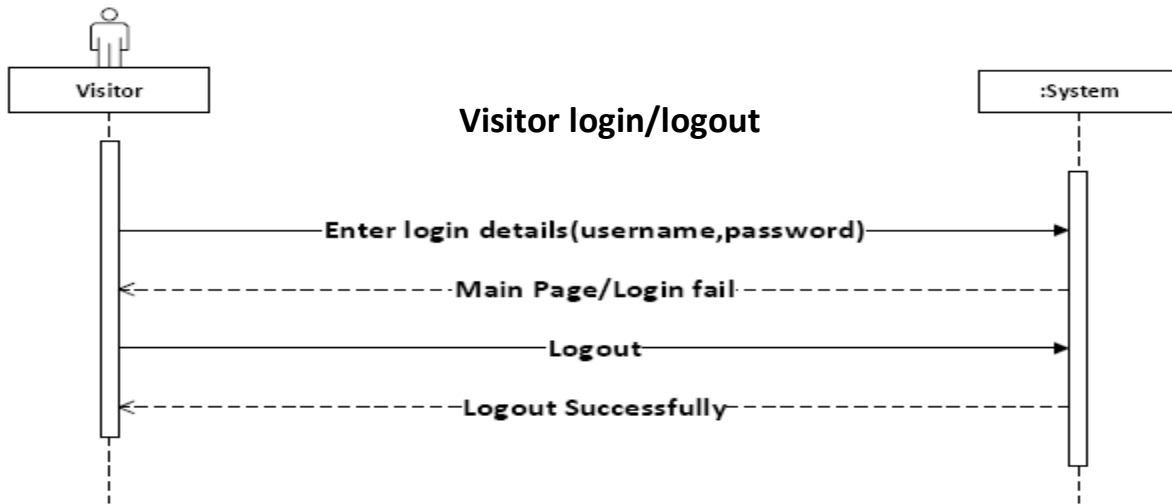


Figure 13 Sequence Visitor login/logout



Figure 13 Visitor Search/Review

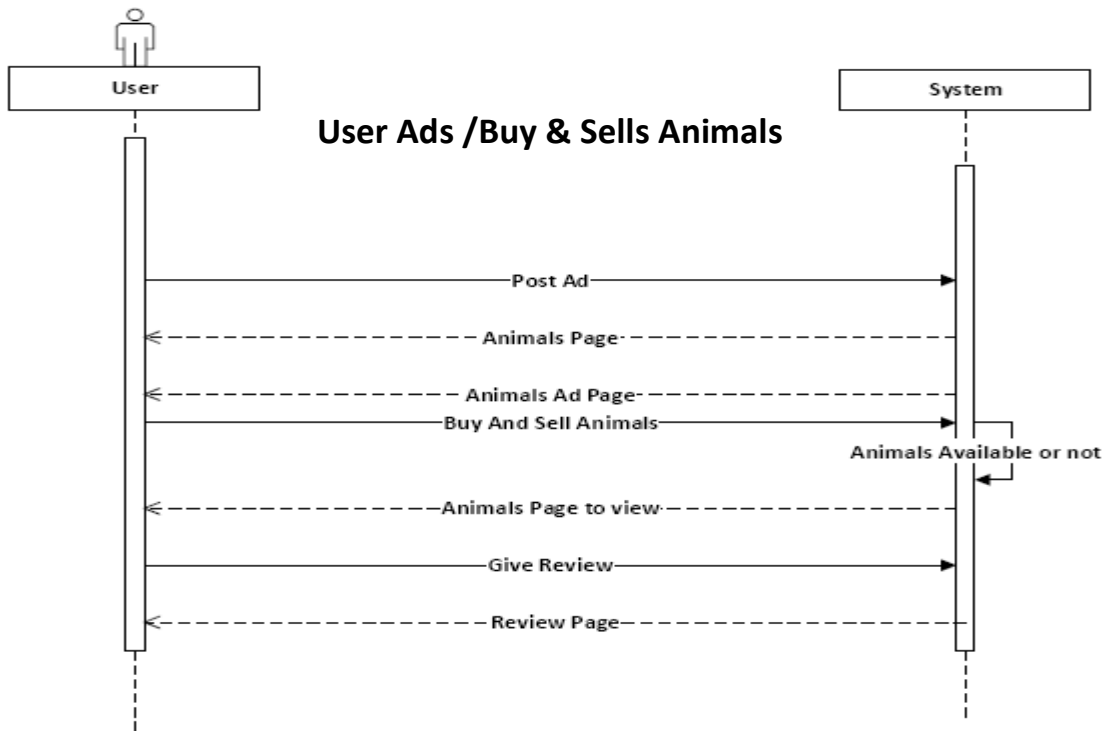


Figure 14 User Ads /Buy & Sells Animals

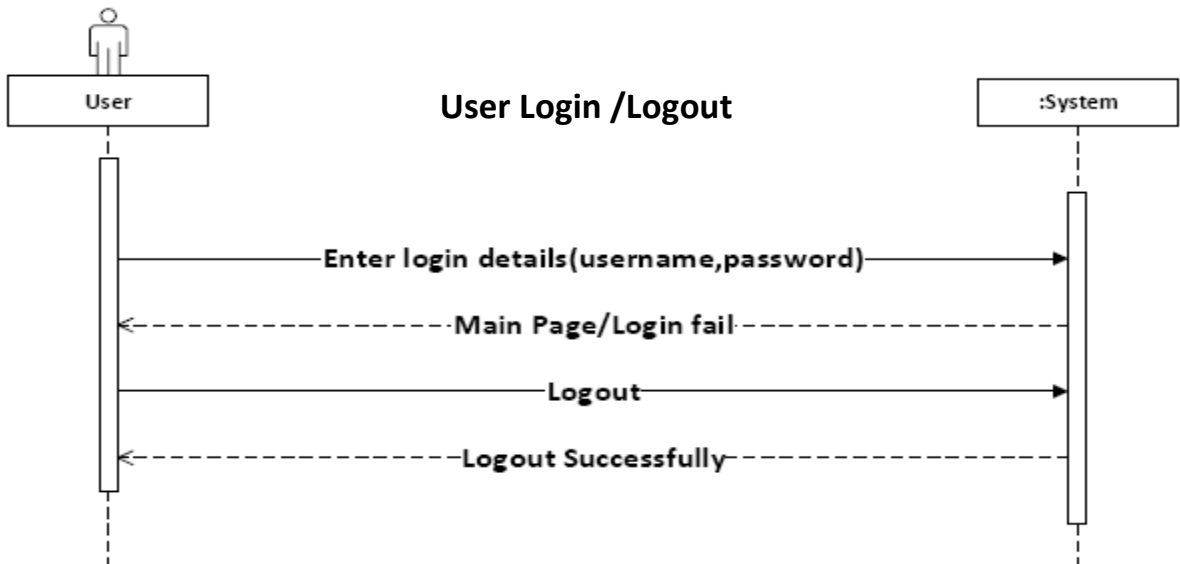


Figure 15 User login/logout

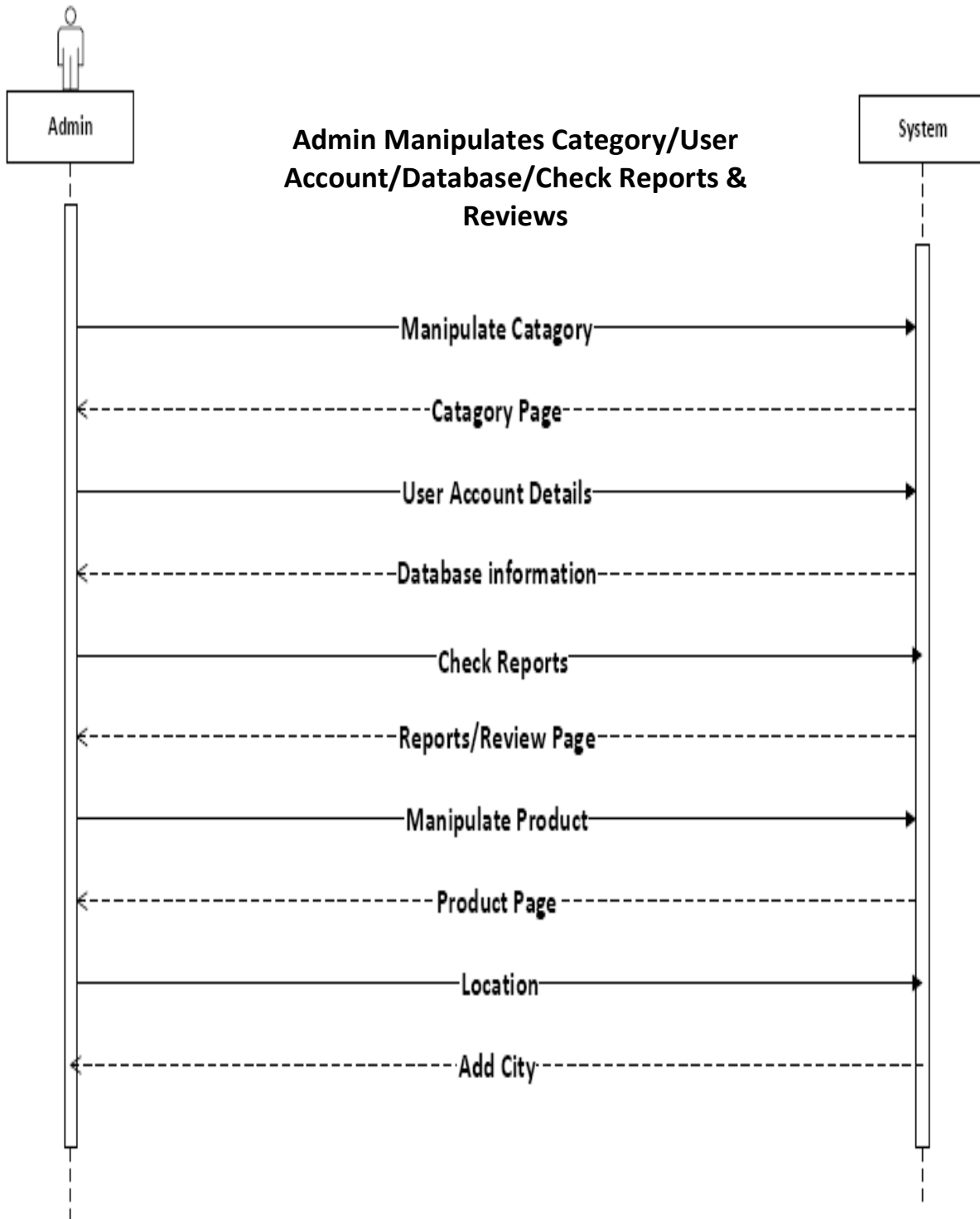


Figure 16 Admin Manipulates Category/User Account/Transection/Payments/Check Reports & Reviews

### 4.5.1 Visitor:

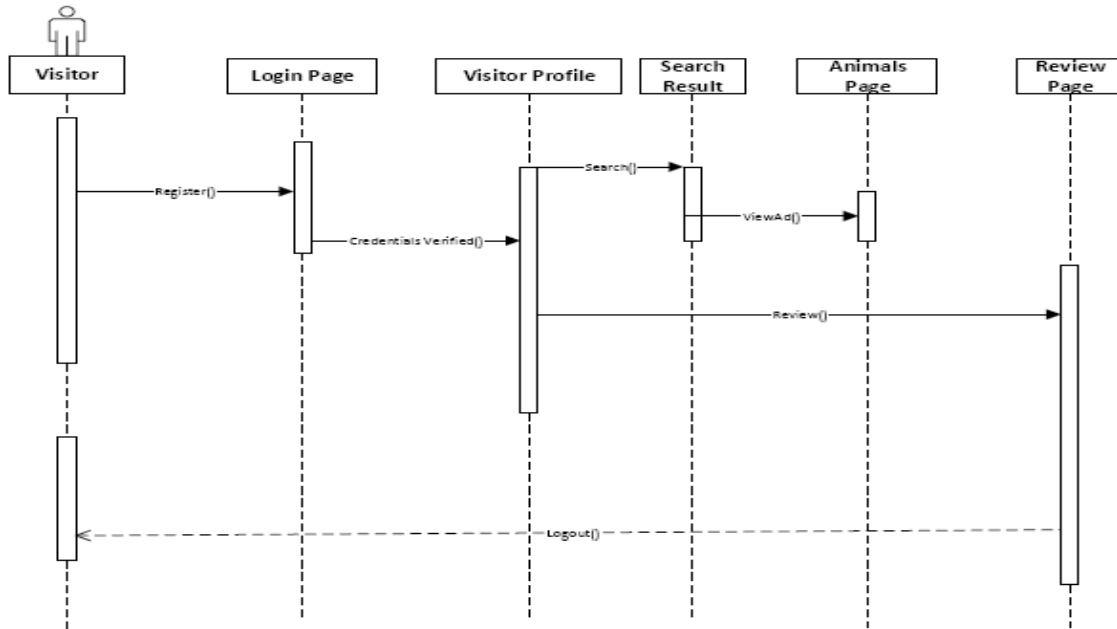


Figure 17 Visitor

### 4.5.2 User:

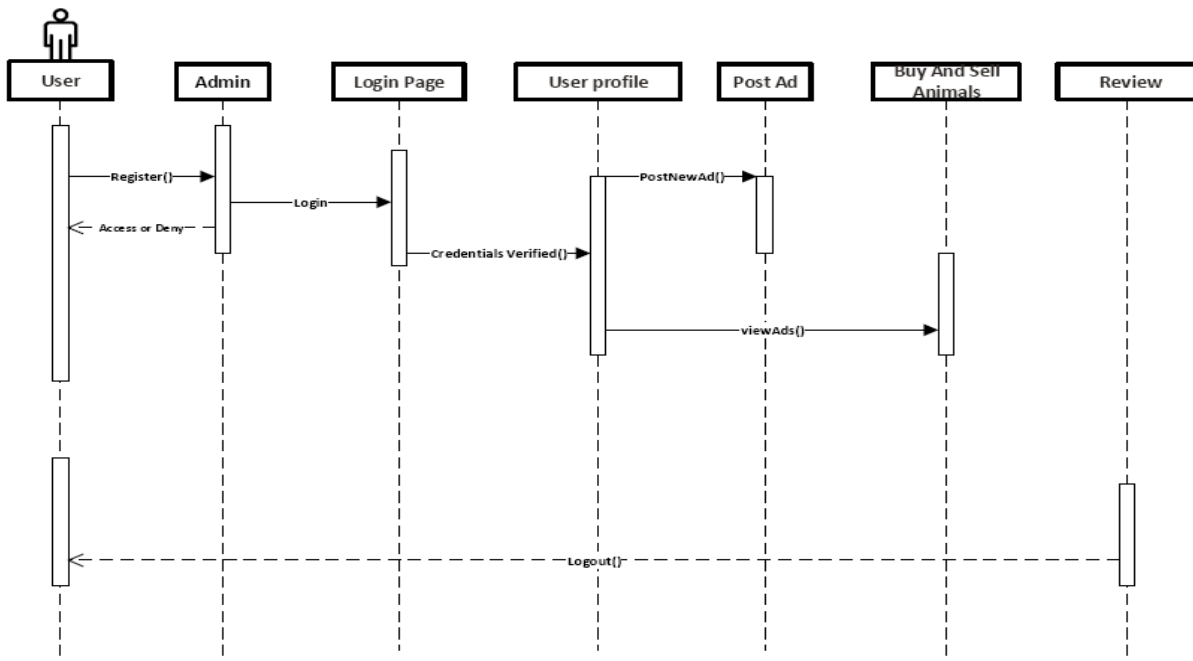


Figure 18 User

## 4.6. Operation Contracts

### OP1

**Operation Name:** Register

**Pre-conditions:**

- User has entered the data in registration fields

**Post-Conditions:**

- User is registered to the user accounts database

### OP2

**Operation Name:** Login

**Pre-conditions:**

- User has provided the correct credentials
- User is registered

**Post-Conditions:**

- User is logged into his account
- User can make changes and post ads

### OP3

**Operation Name:** Verify

**Pre-conditions:**

- Credentials are entered

**Post-Conditions:**

- Log in if the credentials are correct and redirect to profile page
- Notify if the credentials are not correct

### OP4

**Operation Name:** Create Ad

**Pre-conditions:**

- User is logged in as a Customer

**Post-Conditions:**

- User can create his own ad and publish it
- Publish Ad

### OP5

**Operation Name:**

**Pre-conditions:**

- User has entered the required information in create ad fields
- The information is valid

**Post-Conditions:**

- The ad is published on the website
- Other users can view this ad and contact the Customer

## **OP6**

**Operation Name:** Promote Ad

**Pre-conditions:**

- Ad has been created
- Promotion fee has been paid

**Post-Conditions:**

- The ad will appear on top of the list as featured ad
- The Customer will be charged monthly for promotion

## **OP7**

**Operation Name:** Animals received

**Pre-conditions:**

- Consumer has returned the Animals back

**Post-Conditions:**

- The ad is now available for renting
- Wish list consumers will be notified

## **OP9**

**Operation Name:** Logout

**Pre-conditions:**

- Renter of consumer is logged in

**Post-Conditions:**

- The user will be logged out of the system

## **OP10**

**Operation Name:** Search

**Pre-conditions:**

- User has entered at least 3 alphabets into the search bar
- **(OPTIONAL)** filters have been applied

**Post-Conditions:**

User will be redirected to the search results page

**OP11**

**Operation Name:** Browse Animals

**Pre-conditions:**

- User has been navigated to the browse products page
- User has clicked on a category or brand

**Post-Conditions:**

- Product results will be showed according to the clicked category or brand

**OP12**

**Operation Name:** Report Ad

**Pre-conditions:**

- Consumer has confirmed reporting the ad
- Consumer has entered reporting reasons and other data in the required fields

**Post-Conditions:**

- The Ad will be reported
- Categorized report will be sent to admin

**OP13**

**Operation Name:** Contact Customer

**Pre-conditions:**

- Consumer has sent contact Customer request

**Post-Conditions:**

- A request will be shown to the renter
- His contact details will be visible to the consumer if request accepted

**OP14**

**Operation Name:** View Profile

**Pre-conditions:**

- User is logged in
- User is not an admin

**Post-Conditions:**

- User will be redirected to his profile page

### 4.7. Activity Diagram User

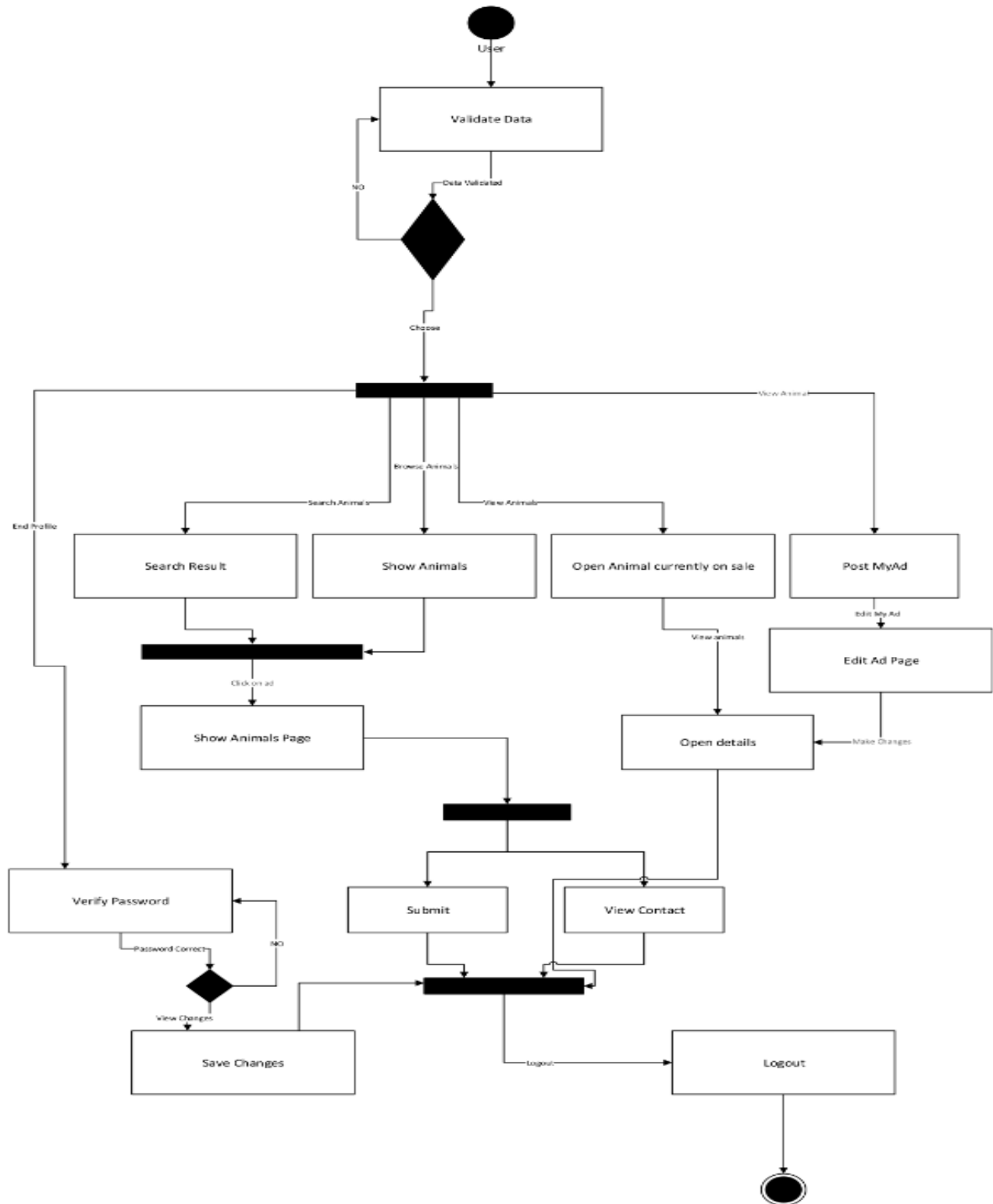


Figure 19 Activity Diagram

### 4.7.1 Activity Diagram Admin

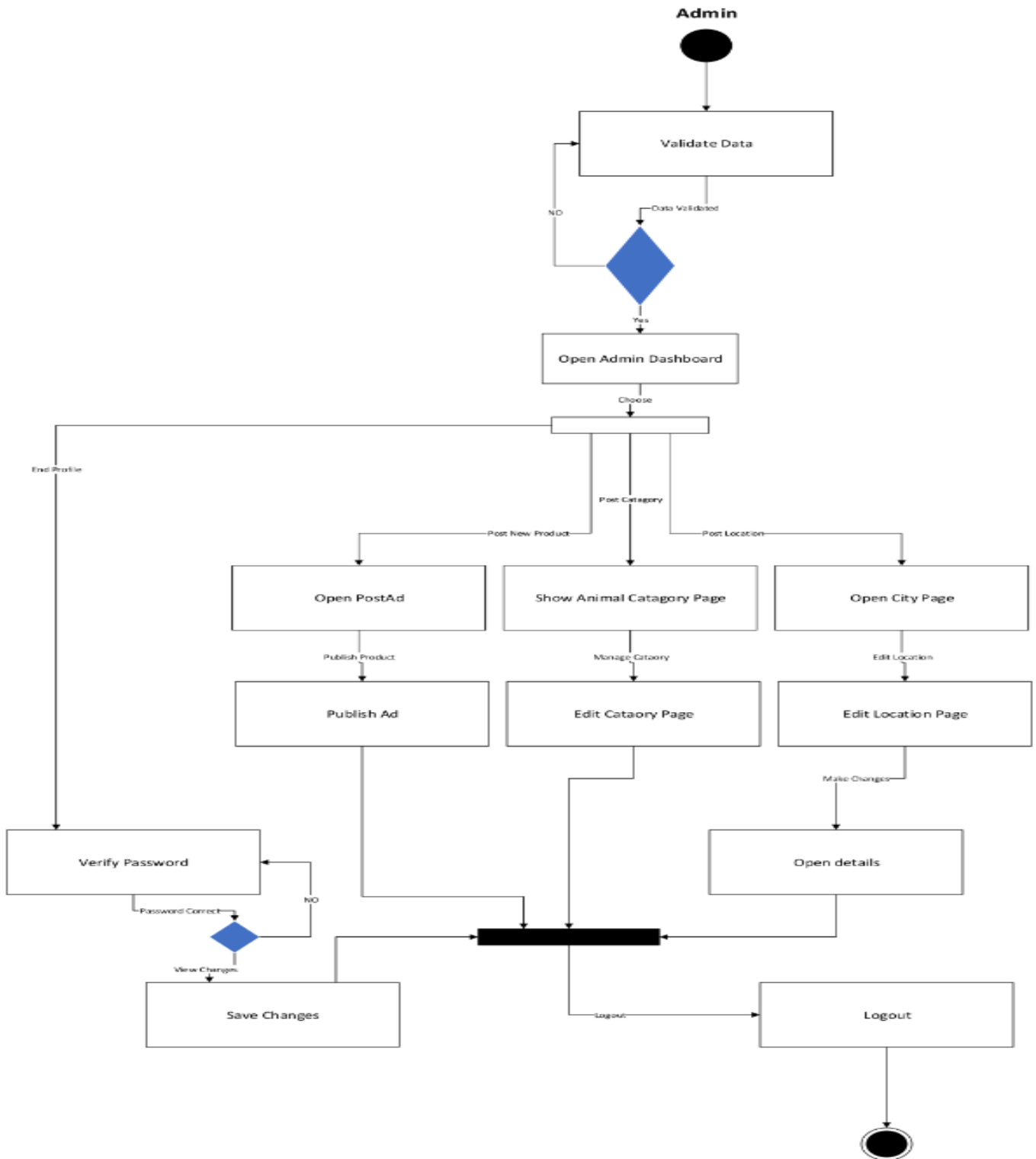


Figure 20 Admin

## 4.8. State Transition Diagram

### 4.8.1 Visitor:

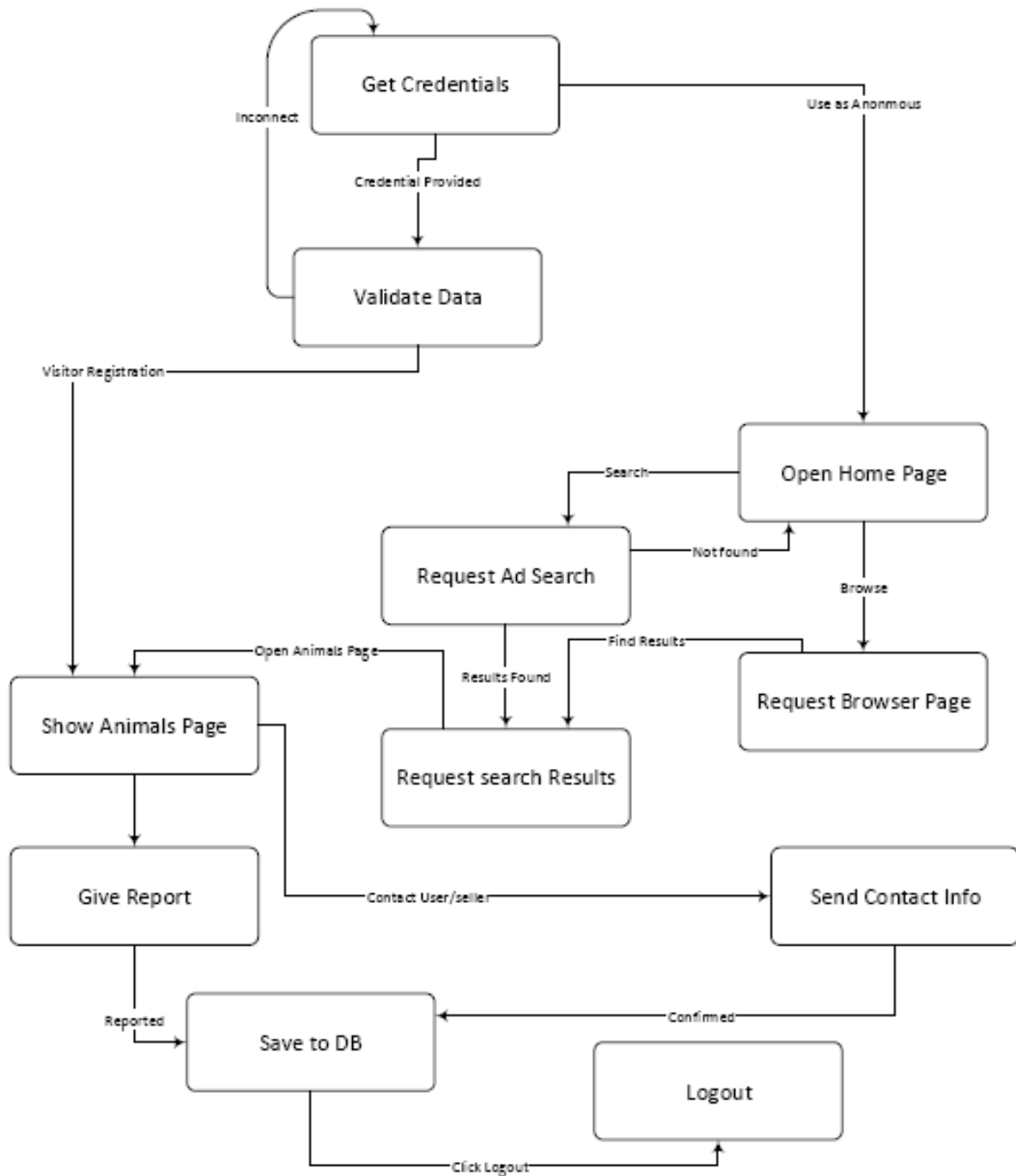


Figure 21 State Transition Diagram Consumer

## 4.8.2 User:

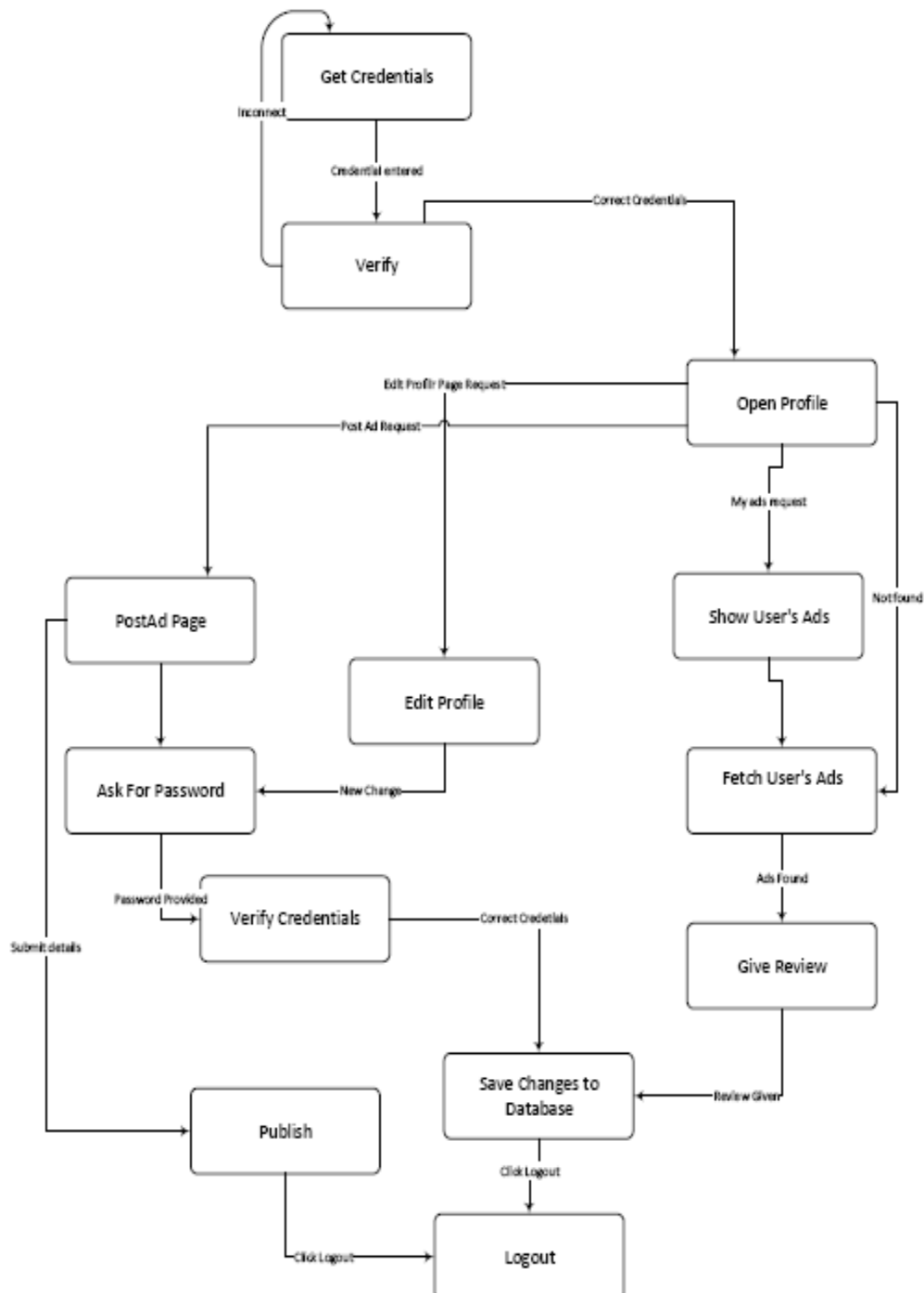


Figure 22 State Transition Diagram Customer

### 4.9. Component Diagram

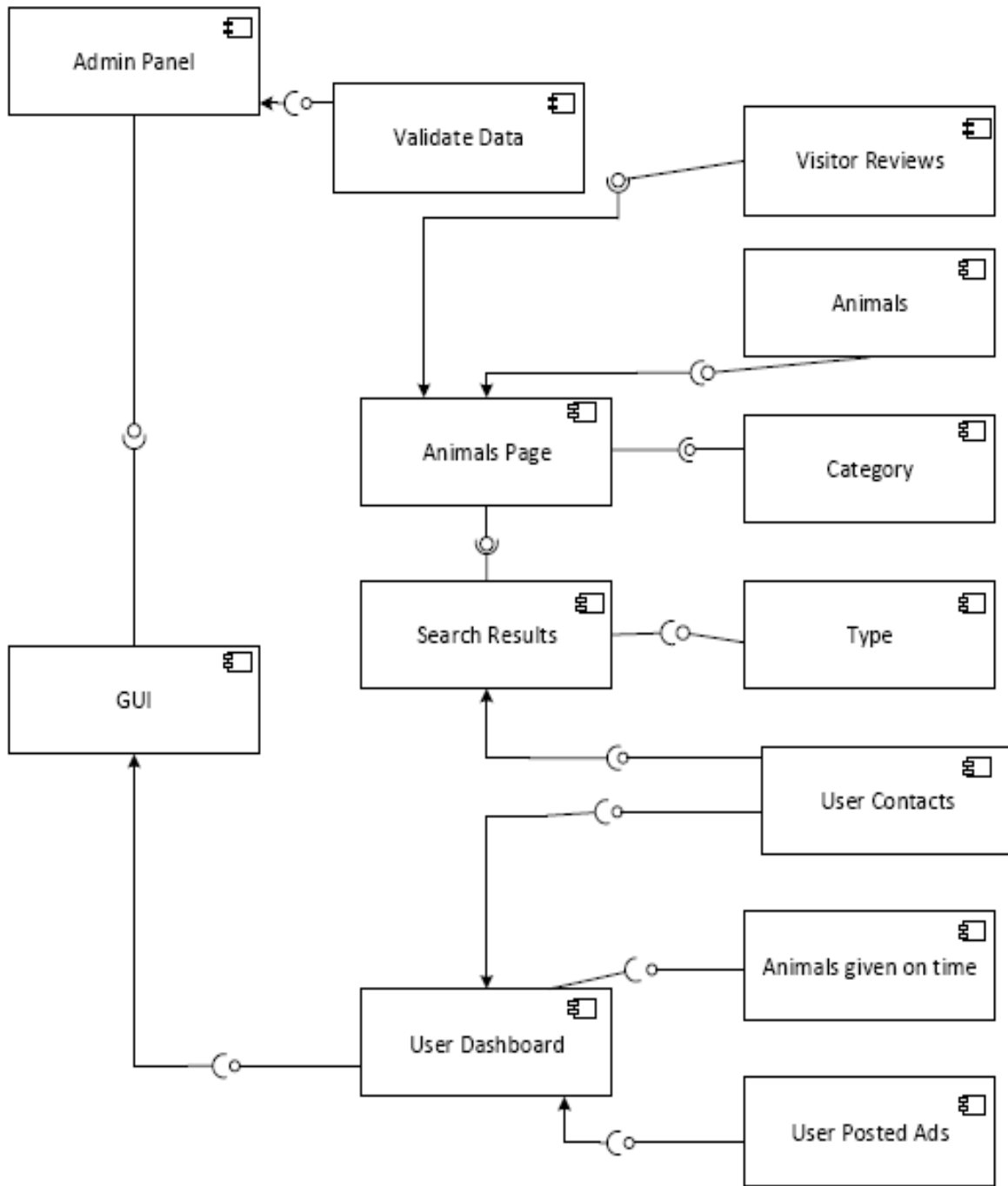


Figure 23 Component Diagram

### 4.10. Deployment Diagram

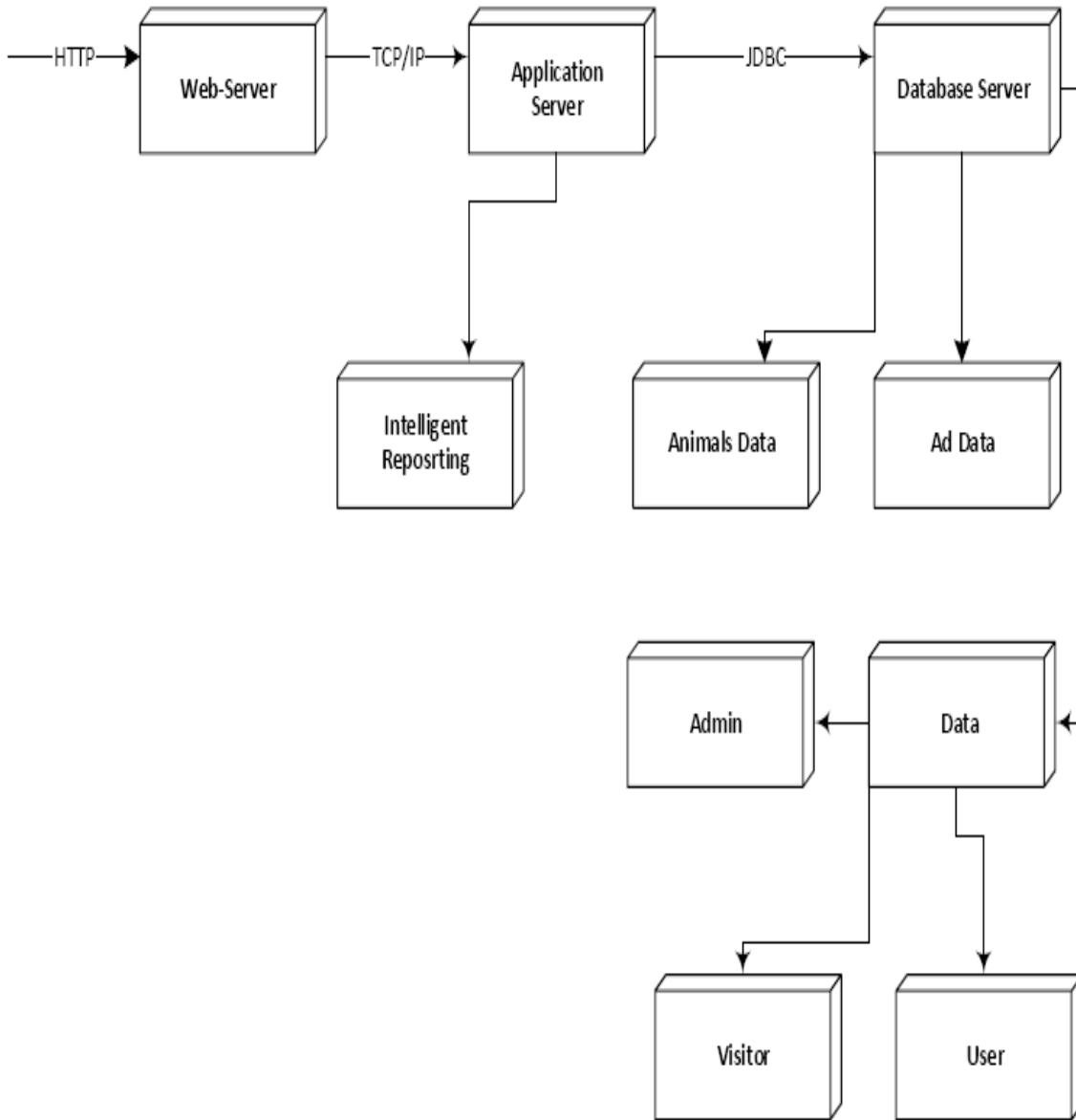


Figure 24 Deployment Diagram

### 4.11. Data Flow Diagram

#### 0 Level:

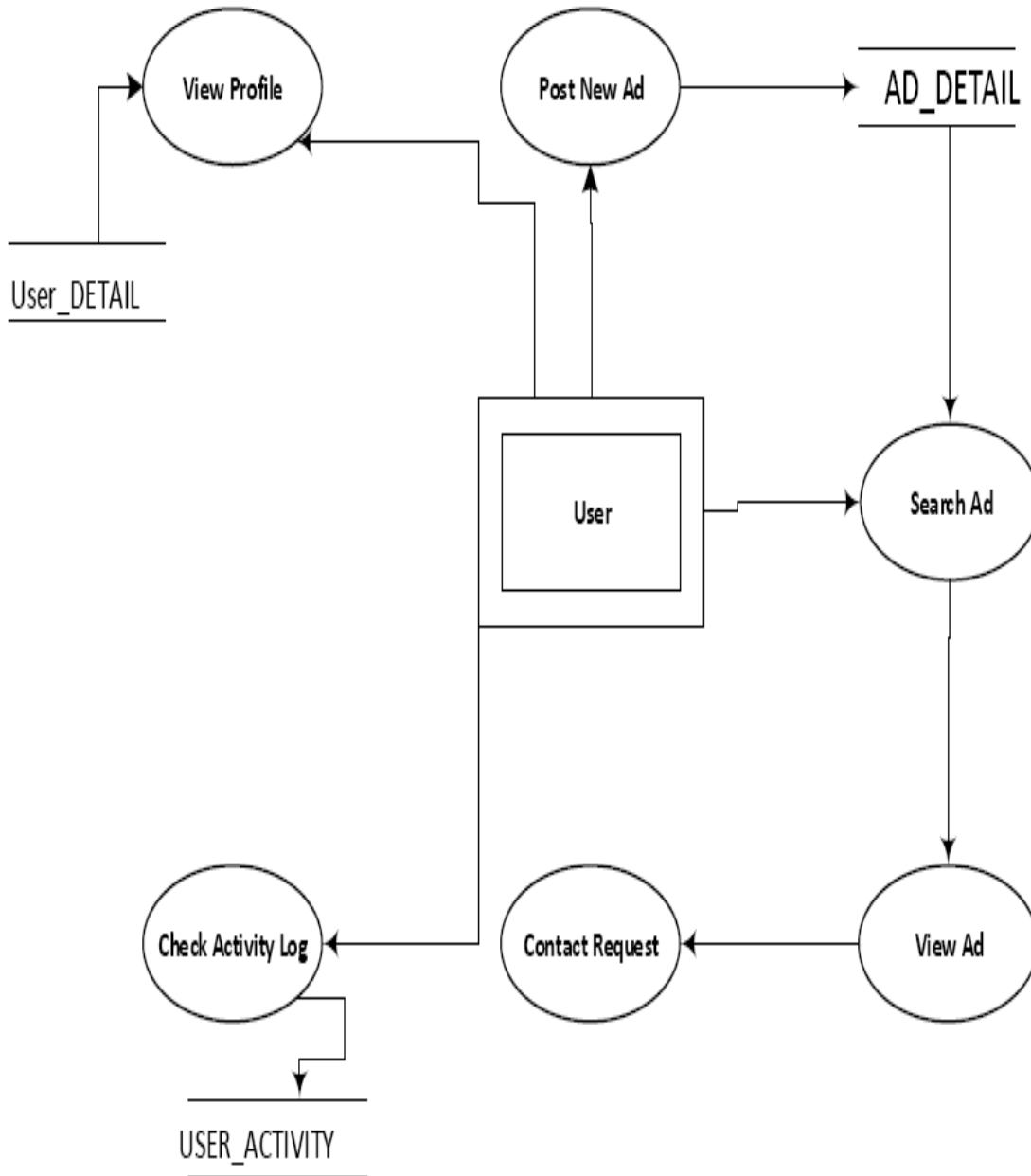


Figure 25 Data Flow Diagram 0 level

**Level 1:**

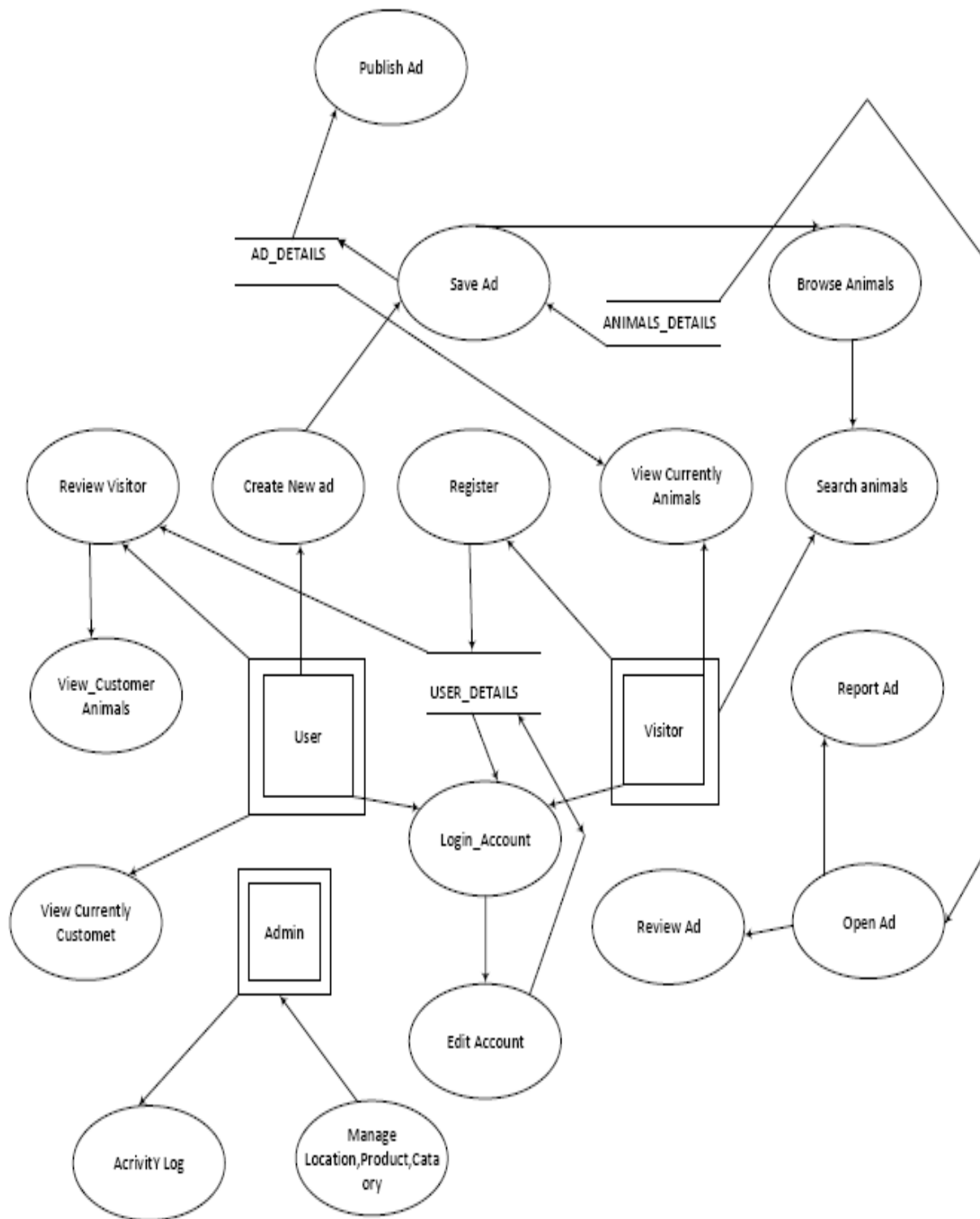


Figure 26 Data Flow Diagram level 1

# Chapter 5

## Implementation

## Chapter 5: Implementation

Waterfall methodology use for the Implementation of this system. We get all the necessary information of all the development phases including requirement gathering, analysis, design, implementation, testing and deployment. In implementation we develop the platform and we check either the system fulfills the requirements.

<b>Requirement Gathering &amp; Analysis</b>	In this phase all the requirements and assets necessary are ready for analyzing. Errors and bugs are pinpointed during analysis and fixed. And then the feasibilities in the different areas of implementation are formulated (including time and cost).
<b>Planning</b>	In this phase plan will be manifested on the basis on the analyzed data and requirements. The time and cost for each SDLC phase will also be decided.
<b>Design</b>	In this phase, development model will be chosen and database will be designed. It will be verified by Supervisor.
<b>Implementation</b>	After design phase implementation according to the requirements. During and after completion of this phase, V&V will be applied to check if the system fulfills the requirements.
<b>Testing</b>	Testing will be done to check requirements and functionality. Both dynamic and static testing will be performed. Dynamically through testing tools and statically to check every line of code manually. Static testing will reduce errors upon execution and allow us to find errors that otherwise are very difficult to find.
<b>Deployment</b>	After tests are done and no further errors are discovered, the system will be deployed. In case of discovery of errors or in case a change is made, regression testing will be performed.

*Table 5-0-1 Implementation*

## 5.1. Important Flow Control/Pseudo codes

```

<!DOCTYPE HTML>
<html lang="zxx">
  <head>
    <title>Maweshi Mandi| Home</title>

    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <script>
      addEventListener("load", function () {
        setTimeout(hideURLbar, 0);
      }, false);

      function hideURLbar() {
        window.scrollTo(0, 1);
      }
    </script>

    <link href="css/bootstrap.min.css" rel="stylesheet" type="text/css" media="all">
    <link href="css/fontawesome-all.min.css" rel="stylesheet" type="text/css" media="all">
    <link rel="stylesheet" href="css/flexslider.css" type="text/css" media="all" />
    <link href="css/jiSlider.css" rel="stylesheet">
    <link rel="stylesheet" href="css/shop.css" type="text/css" />
    <link href="css/style.css" rel='stylesheet' type='text/css' media="all">

    <link href="//fonts.googleapis.com/css?family=Sunflower:500,700" rel="stylesheet">
    <link href="//fonts.googleapis.com/css?family=Open+Sans:400,600,700" rel="stylesheet">
  </head>
  <body>
    <div class="header-outs" id="home">
      <div class="header-bar">
        <div class="info-top-grid">
          <div class="info-contact-agile">
            <ul>
              <li>
                <span class="fas fa-phone-volume"></span>
                <p>+91301569248</p>
              </li>
            </ul>
          </div>
        </div>
      </div>
    </div>
  </body>
</html>

```

```

        </li>
        <span class="fas fa-envelope"></span>
        <p><a href="mailto:maweshimandi@gmail.com">maweshimandi@gmail.com</a></p>
    </li>
    </li>
    </li>
</ul>
</div>
</div>
<div class="container-fluid">
    <div class="hedder-up row">
        <div class="col-lg-3 col-md-3 logo-head">
            <h1><a class="navbar-brand" href="index.html"></a></h1>
        </div>
        <div class="col-lg-5 col-md-6 search-right">
            <form class="form-inline my-lg-0">
                <input class="form-control mr-sm-2" type="search" placeholder="Search">
                <button class="btn" type="submit">Search</button>
            </form>
        </div>
        <div class="col-lg-4 col-md-3 right-side-cart">
            <div class="cart-icons">
                <ul>
                    <li>
                        <span class="far fa-heart"></span>
                    </li>
                    <li>
                        <button type="button" data-toggle="modal" data-target="#exampleModal"> <span class="far fa-user"></span></button>
                    </li>
                    <li class="toyscart toyscart2 cart cart_box_1">
                        <form action="#" method="post" class="last">
                            <input type="hidden" name="cmd" value="_cart">
                            <input type="hidden" name="display" value="1">
                            <button class="top_toys_cart" type="submit" name="submit" value="">
                            <span class="fas fa-cart-arrow-down"></span>
                            </button>
                        </form>
                    </li>
                </ul>
            </div>
        </div>
    </div>
    ...

```

### **Admin Panel:**

- Logs in
- Checks activity log
- Views intelligent report
- Decision Making, removes and verifies accounts

### **User (Buyer/Seller) Panel:**

- Logs in
- Views Animals/birds for sell
- Checks seller report
- Views active and inactive ads
- Creates new ad and publishes it
- Edits profile

### **Visitor Panel:**

Registration

Search for Animals

Browses Animals

Opens Ad

## **5.2. Components, Libraries, Web Services and stubs**

### **Languages and Libraries:**

Bootstrap4, PHP, JavaScript, HTML, CSS

And for Android App we use the XML, JAVA and android Studio libraries.

### **Web services:**

Rest API

Android API

## **5.3. Deployment Environment**

### **Browser**

- Safari

- Chrome
- Firefox
- MS edge

## Server

Bandwidth: 6GB

Space: 500MB

## 5.4. Tools and Techniques

- Visual Studio
- Visio Studio
- Xampp
- Adobe Photoshop
- Illustrator
- Bootstrap
- Android Studio

### Languages:

Php, Java, JavaScript, Html, CSS,Xml.

## 5.5. Best Practices / Coding Standards

- Files MUST use only `<? php` and `<? =` tags.
- Files MUST use only UTF-8 without BOM for PHP code.
- File should either declare symbols (classes, functions, constants, etc.) or cause side-effects (e.g. generate output, change settings) but should not do both.
- Namespaces and classes MUST follow an “autoloading”.
- Class names MUST be declared in Studly Caps.
- Class constants MUST be declared in all upper case with underscore separators.
- Method names MUST be declared in camelCase.

# Chapter 6

## Testing and Evaluation

## **Chapter 6: Testing and Evaluation**

In this chapter testing of all units and individually will done and if each unit or module is working correctly then integration testing will be done to check the effect when two or more than two modules are working with each other. After completion of unit and integration than performance and stress testing will be done.

### **6.1. Use Case Testing**

**6.1.1.**

Test Suite ID	TS001
Test Case ID	TC001
Test Case Summary	To verify that by clicking Signup button username, password, Email, phone# and address store in customer detail in database.
Related Requirement	RS001: User should able to Sign up.
Prerequisites	No
Test Procedure	<ol style="list-style-type: none"> <li>1. Select fields in Signup form.</li> <li>2. Enter user data in fields.</li> <li>3. Click Signup button.</li> </ol>
Test Data	Valid username: khurram shafique, KHURRAM SHAFIQUE Invalid username: 12khurram, ./khurram, khurram %^3 Valid password: 123Abc@5, Abc567\$%9 Invalid password: 1_2 khurram, _ khurram 12 Valid email: <a href="mailto:ranakhurram@gmail.com">ranakhurram@gmail.com</a> , <a href="mailto:12ali@yahoo.com">12ali@yahoo.com</a> Invalid email: ali.com, khurram@yahoo Valid phone# :03034144964 Invalid phone: @56rfgf7999, 2wstyyA
Expected Result	<ol style="list-style-type: none"> <li>1. If username, password, email and phone# are valid then store user data by clicking signup button in rener detail.</li> <li>2. If given inputs are invalid then display error message.</li> <li>3. If fields are empty then display then show warning message.</li> </ol>
Actual Result	<ol style="list-style-type: none"> <li>1. If name is valid, the result is as expected.</li> <li>2. If name is not valid then invalid message displayed.</li> <li>3. If fields are empty then warning message displayed.</li> </ol>
Status	Pass
Remarks	This test case is simple and easy.
Created By	Abdul waheed
Date of Creation	11/06/19
Executed By	Abdul waheed
Date of Execution	11/06/19
Text Environment	OS: Window 10 Browser: Google Chrome

**6.1.2.**

Test Suite ID	TS001
Test Case ID	TC002
Test case Summary	To verify Authentication or Login.
Related Requirement	RS002: User should able to Login.
Prerequisites	User should Signup first.
Test Procedure	<ol style="list-style-type: none"> <li>1. Select Username field and enter username.</li> <li>2. Select Password field and enter password.</li> <li>3. Click Login button.</li> </ol>
Test Data	Valid username: khurram shafique, KHURRAM SHAFIQUE Invalid username: 12khurram, ./khurram, khurram %^3 Valid password: 123Aabc@, abMc567\$% Invalid password: 1_2 khurram, _ khurram 12
Expected Result	<ol style="list-style-type: none"> <li>1. If username and password are valid then clicking the Login button user successfully login.</li> <li>2. If username and password are invalid then clicking the Login button invalid message display.</li> <li>3. If fields are empty then display warning message</li> </ol>
Actual Result	<ol style="list-style-type: none"> <li>1. If both fields are valid, the result is as expected.</li> <li>2. If both fields are not valid then invalid message displayed.</li> <li>3. If fields are empty then warning message displayed.</li> </ol>
Status	Pass
Remarks	This test case is simple and easy.
Created By	Abdul waheed
Date of Creation	11/06/19
Executed By	Abdul waheed
Date of Execution	11/06/19
Text Environment	OS: Window 10 Browser: Google Chrome

**6.1.3.**

Test Suite ID	TS002
Test Case ID	TC003
Testcase Summary	To verify that Posted Ad title, price, category and detail update in ad detail in database by clicking Update button.
Related Requirement	RS006: User should able to update ad.
Prerequisites	User should login.
Test Procedure	<ol style="list-style-type: none"> <li>1. Select fields in ad update form.</li> <li>2. Enter ad data in fields.</li> <li>3. Click update ad button.</li> </ol>
Test Data	Valid Title: Animals for sale. Invalid Title: hjg### , ////..._ Valid Animals: Cow, Beffalo, goat, Camel, Sheep Invalid Brand: ./ Cow, ./Beffalo,./ goat, ./Camel, ./Sheep Valid Category: Cow, Beffalo, goat, Camel, Sheep Invalid Category: 12 Cow, ./ Beffalo Valid Price: 120000, 100 Invalid Price: qamm555, 00op, AAA
Expected Result	<ol style="list-style-type: none"> <li>1. If title, category and price are valid then store update ad data by clicking add button in ad detail.</li> <li>2. If given inputs are invalid then display error message.</li> <li>3. If fields are empty then display warning message.</li> </ol>
Actual Result	<ol style="list-style-type: none"> <li>1. If fields are valid, the result is as expected.</li> <li>2. If fields are not valid then invalid message displayed.</li> <li>3. If fields are empty then warning message displayed.</li> </ol>
Status	Pass
Remarks	This test case is simple and easy.
Created By	Abdul waheed
Date of Creation	12/06/19
Executed By	Abdul waheed
Date of Execution	12/06/19
Text Environment	OS: Window 10 Browser: Google Chrome

6.1.4.	
Test Suite ID	TS002
Test Case ID	TC004
Testcase Summary	To verify that Ad delete from post ads detail in database.
Related Requirement	RS006: User can delete product.
Prerequisites	User should login.
Test Procedure	1- Click delete ad button.
Test Data	Valid: click button
Expected Result	2- If ad deleted by click button then message will appear.
Actual Result	3. If action is valid, the result is as expected.
Status	Pass
Remarks	This test case is simple and easy.
Created By	Abdul waheed
Date of Creation	12/06/19
Executed By	Abdul waheed
Date of Execution	12/06/19
Text Environment	OS: Window 10
	Browser: Google Chrome

**6.1.5.**

Test Suite ID	TS002
Test Suite ID	TS002
Test Case ID	TC005
Testcase Summary	To verify that Posted Ad title, price, brand, category and detail store in ad detail in database by clicking Post Ad button.
Related Requirement	RS006: User should able to post ad.
Prerequisites	User should login.
Test Procedure	<ol style="list-style-type: none"> <li>4. Select fields in post ad form.</li> <li>5. Enter ad data in fields.</li> <li>6. Click add ad button.</li> </ol>
Test Data	Valid Title: Animals for sale. Invalid Title: hjg### , /////... _ Valid Animals: Cow, Beffalo, goat, Camel, Sheep Invalid Brand: ./ Cow, ./Beffalo,./ goat, ./Camel, ./Sheep Valid Category: Cow, Beffalo, goat, Camel, Sheep Invalid Category: 12 Cow, ./ Beffalo Valid Price: 120000, 100 Invalid Price: qamm555, ooop, AAA Valid Title: Animals for sale.
Expected Result	<ol style="list-style-type: none"> <li>4- If title, category and price are valid then store ad data by clicking add button in ad detail.</li> <li>5- If given inputs are invalid then display error message.</li> <li>6- If fields are empty then display warning message.</li> </ol>
Actual Result	<ol style="list-style-type: none"> <li>5. If both fields are valid, the result is as expected.</li> <li>6. If both fields are not valid then invalid message displayed.</li> <li>7. If fields are empty then warning message displayed.</li> </ol>
Status	Pass
Remarks	This test case is simple and easy.
Created By	Abdul waheed
Date of Creation	12/06/19
Executed By	Abdul waheed

Date of Execution	12/06/19
Text Environment	OS: Window 10 Browser: Google Chrome

### 6.1.6.

Test Suite ID	TS003
Test Case ID	TC006
Test case Summary	To verify that Product is searched by clicking search product button.
Related Requirement	RS006: User should able to search product.
Prerequisites	Product/Animal is available.
Test Procedure	<ol style="list-style-type: none"> <li>7. Select search Product/Animal field.</li> <li>8. Enter product name in fields.</li> <li>9. Click search Product/Animal button.</li> </ol>
Test Data	Valid: Cow, Buffalo, goat, Camel, Sheep Invalid: ./ Sheep, ./goat, ./Camel
Expected Result	<ol style="list-style-type: none"> <li>4. If name is valid then show product with detail.</li> <li>5. If name is invalid then display error message.</li> <li>6. If field is empty then display warning message.</li> </ol>
Actual Result	<ol style="list-style-type: none"> <li>8. If name is valid, the result is as expected.</li> <li>9. If name is not valid then invalid message display.</li> <li>10. If field is empty then warning message display.</li> </ol>
Status	Pass
Remarks	This test case is simple and easy.
Created By	Abdul waheed
Date of Creation	12/06/19
Executed By	Abdul waheed
Date of Execution	12/06/19
Text Environment	OS: Window 10 Browser: Google Chrome

**6.1.7.**

Test Suite ID	TS003
Test Case ID	TC007
Testcase Summary	To verify that Product detail display by clicking product detail button.
Related Requirement	RS007: User should able to see product/Animals detail.
Prerequisites	Search product/Animals.
Test Procedure	<ol style="list-style-type: none"> <li>1. Search product/Animals.</li> <li>2. Click on product/Animals detail.</li> </ol>
Test Data	Valid: Cow, Buffalo, goat, Camel, Sheep Invalid: ./ Sheep, ./goat, ./Camel
Expected Result	<ol style="list-style-type: none"> <li>1. If product is valid then show product detail.</li> <li>2. If product is invalid then display error message.</li> </ol>
Actual Result	<ol style="list-style-type: none"> <li>1. If product is valid, the result is as expected.</li> <li>2. If product is not valid then invalid message display.</li> </ol>
Status	Pass
Remarks	This test case is simple and easy.
Created By	Abdul waheed
Date of Creation	12/06/19
Executed By	Abdul waheed
Date of Execution	12/06/19
Text Environment	OS: Window 10 Browser: Google Chrome

**6.1.8.**

Test Suite ID	TS002
Test Case ID	TC004
Testcase Summary	To verify that account delete from user accounts detail in database.
Related Requirement	RS006: Admin can delete product.
Prerequisites	Admin should login. Open users detail.
Test Procedure	7- Click delete button.
Test Data	Valid: click button
Expected Result	8- If ad deleted by click button then message will appear.
Actual Result	11. If action is valid, the result is as expected.
Status	Pass
Remarks	This test case is simple and easy.
Created By	Abdul waheed
Date of Creation	12/06/19
Executed By	Abdul waheed
Date of Execution	12/06/19
Text Environment	OS: Window 10 Browser: Google Chrome

## 6.2. Equivalence partitioning

### Derived Equivalence Classes

1. User name is alphabetic. (valid)
2. User name is not alphabetic. (invalid)
3. Password is equal to 8 or greater than 8 characters in length. (valid)
4. Password is 2 to 15 characters in length. (invalid)
5. Password include one uppercase letter & one lowercase letter. (valid)
6. Password include ‘\_’. (invalid)
7. Email without ‘@’ and ‘.com’. (invalid)
8. Email with ‘@’ and ‘.com’. (valid)
9. Phone# only be number. (valid)
10. Phone# with alphabetic. (invalid)
11. City name is alphabetic. (valid)
12. City name with numeric values. (invalid)
13. Price only be numeric. (valid)
14. Price is alphabetic. (invalid)
15. Search product with alphabetic. (valid)
16. Search product with special character (/,%^,#,@). (invalid)
17. No search product name entered. (invalid)
18. Brand name alphabetic (valid)
19. Brand name with special character. (invalid)
20. Empty field (invalid)

Black Box Test Cases based on the Equivalence Classes Above.

Sr.	Test Data	Expected Outcome	Classes Covered
1	Khurram, saem1	T	1
2	ss, 11233	T	2
3	123Ea84747, 12ABc4%^	T	3,5
4	_1234567, AV_we333	T	4,6
5	khurram @gmail.com	T	7
6	khurram.com, khurram@ali	T	8
7	0322494949	T	9,14
8	032222XXX12),1234	T	10
9	Lahore, Karachi	T	11
10	Lahore1	T	12
11	12000, 100	T	13

12	Rs.1200, 12\$	T	14
13	Cow, Sahiwal	T	15
14	Sheep, Sahiwal	T	18
15	..... (empty)	T	17,20

### 6.3. Boundary value analysis

Sr.		Partition 1	Partition 2	Partition 3
1	Password	Less than 8 character	1 – 8 character	9 – 12
2	Price	Less than 0	Greater than 1	Less than
3	Phone	<=0	1 - 11	>=12

### 6.4. Data flow testing

#### Read url

1. \$url = ' http://localhost/MM/templates/index.php ';
2. \$ch = curl\_init(\$url);
3. function getUserIpAddr(){
4. if(!empty(\$\_SERVER['HTTP\_CLIENT\_IP'])){
5. \$ip = \$\_SERVER['HTTP\_CLIENT\_IP'];
6. }elseif(!empty(\$\_SERVER['HTTP\_X\_FORWARDED\_FOR'])){
7. \$ip = \$\_SERVER['HTTP\_X\_FORWARDED\_FOR'];
8. }else{
9. \$ip = \$\_SERVER['REMOTE\_ADDR'];
10. }
11. return \$ip;
12. }
13. \$ip\_address = getUserIpAddr();
14. \$jsonData = array(

```

15.'name' => $name'city' => $city,
16. 'email' => $email,
17. 'password' => $password,
18. 'address' => $address,
19. 'phone' => $phone,
20. 'ip_address'=> $ip_address,
21. 'image_num' => $images_db
22. );
23. $jsonDataEncoded = json_encode($jsonData);
24. curl_setopt($ch, CURLOPT_POST, 1);
25. curl_setopt($ch, CURLOPT_POSTFIELDS, $jsonDataEncoded);

26. curl_setopt($ch, CURLOPT_HTTPHEADER, array('Content-Type: application/json'));
28.$result = curl_exec($ch);

```

Associations	All p-use	All c-use
(1,2,\$url)		
(2,25,\$ch)		
(2,26,\$ch)		
(2,27,\$ch)		
(2,28,\$ch)		
(4,(5,T),\$ _SERVER)	*	
(4,(7,F), \$ _SERVER)	*	
(5,11,\$ip)		
(7,11,\$ip)		
(13,21,\$ip_address)		
(14,24,\$jsonData)		

**Read \$url**

1. \$url = 'http://localhost/MM/templates/index.php ';
2. \$ch = curl\_init(\$url);

Associations	All p-use	All c-use
(1,2,\$url)		

**Read \$ch**

1. \$ch = curl\_init(\$url);
2. curl\_setopt(\$ch, CURLOPT\_POST, 1);
3. curl\_setopt(\$ch, CURLOPT\_POSTFIELDS, \$jsonDataEncoded);
4. curl\_setopt(\$ch, CURLOPT\_HTTPHEADER, array('Content-Type: application/json'));
5. \$result = curl\_exec(\$ch);
6. if(\$result)
7. {
8. echo "<script>alert('Account Created successfully !');</script>";
9. }

Associations	All p-use	All c-use
(1,2,\$ch)		
(1,3,\$ch)		
(1,4,\$ch)		
(1,5,\$ch)		*
(5,(6,T),\$ _SERVER)	*	
(5,(6,F), \$ _SERVER)	*	

**Read \$\_SERVER**

1. function getUserIpAddr(){
2. if(!empty(\$\_SERVER['HTTP\_CLIENT\_IP'])){\
3. \$ip = \$\_SERVER['HTTP\_CLIENT\_IP'];
4. }elseif(!empty(\$\_SERVER['HTTP\_X\_FORWARDED\_FOR'])){\
5. \$ip = \$\_SERVER['HTTP\_X\_FORWARDED\_FOR'];
6. }else{\
7. \$ip = \$\_SERVER['REMOTE\_ADDR'];
8. }
9. return \$ip;

Associations	All p-use	All c-use
(2,3,\$_SERVER)		
(2,5, \$_SERVER)		
(2,7, \$_SERVER)		
(3,9,\$ip)		
(5,9,\$ip)		
(7,9,\$ip)		

**Read \$ip\_address**

1. \$ip\_address = getUserIpAddr();
- 2.'ip\_address'=> \$ip\_address,

Associations	All p-use	All c-use
(1,2,\$ip_address)		

**Read \$jsonData**

1. \$jsonData = array(
2. 'name' => \$name,
3. 'city' => \$city,
4. 'email' => \$email,
5. 'password' => \$password,
6. 'address' => \$address,
7. 'phone' => \$phone,
8. 'ip\_address'=> \$ip\_address,
9. 'image\_num' => \$images\_db
10. );
11. \$jsonDataEncoded = json\_encode(\$jsonData);

Associations	All p-use	All c-use
(1,11,\$jsonData)		

**Read \$jsonDataencoded**

1. \$jsonDataEncoded = json\_encode(\$jsonData);
2. curl\_setopt(\$ch, CURLOPT\_POSTFIELDS, \$jsonDataEncoded);

Associations	All p-use	All c-use
(1,2,\$jsonDataEncoded)		

**Read \$user\_data**

1. \$user\_data = json\_decode(\$content3, false);
2. \$\_SESSION["user\_data"] = \$user\_data;

Associations	All p-use	All c-use
(1,2,\$user_data)		

**Read \$user\_exist**

1. \$user\_exist = count(\$content3);
2. if(\$user\_exist == 1 )
3. {
4. \$\_SESSION["user\_data"] = \$user\_data;
5. }

Associations	All p-use	All c-use
(1,(2,T),\$user_exist)		
(2,(2,F), \$user_exist)		

**Read \$city**

1. \$city = new City(\$db);
- \$stmt = \$city->search(\$keywords);

Associations	All p-use	All c-use
(1,2,\$city)		

**Read \$keyword**

1. \$keywords=isset(\$\_GET["s"]) ? \$\_GET["s"] : "";
- \$stmt = \$city->search(\$keywords);

Associations	All p-use	All c-use
(1,2,\$keywords)		

**6.4 Unit testing**

## 6.6. Integration testing

Integration ID	001
Unit ID	001, 002
Testcase Summary	To verify that user first signup and then login.
Related Requirement	User should signup. User should login.
Prerequisites	For login user should signup.
Test Procedure	<ol style="list-style-type: none"> <li>1. Select fields in post ad form.</li> <li>2. Enter user data in fields.</li> <li>3. Click add button.</li> </ol>
Test Data	Valid username: khurram, KHURRAM Invalid username: 12 khurram r, ./ khurram, khurram %^3 Valid password: 123Abc@5, Abc567\$%9 Invalid password: 1_2 khurram, _ khurram 12 Valid email: <a href="mailto:khurram@gmail.com">khurram@gmail.com</a> , 12ali@yahoo.com Invalid email: khurram.com, khurram@yahoo Valid phone#: 03123456578 Invalid phone: @56rfgf7999, 2wstyyA
Expected Result	<ol style="list-style-type: none"> <li>1. If username, password, email and phone# are valid then store user data by clicking signup button in renter detail.</li> <li>2. If given inputs are invalid then display error message. If fields are empty then display then show warning message.</li> <li>3. If email and password are valid by clicking login button then login.</li> <li>4. If given inputs are invalid then display error message.</li> <li>5. If fields are empty then display then show warning message.</li> </ol>
Actual Result	<ol style="list-style-type: none"> <li>1. If data is valid, the result is as expected.</li> <li>2. If data is not valid then invalid message displayed.</li> <li>3. If fields are empty then warning message displayed.</li> </ol>
Status	Pass
Remarks	This test case is simple and easy.
Created By	Abdul waheed

Date of Creation	13/06/19
Executed By	Abdul waheed
Date of Execution	13/06/19
Text Environment	OS: Window 10 Browser: Google Chrome

Integration ID	002
Unit ID	002, 003
Testcase Summary	To verify that user first login and then post ad.
Related Requirement	User should login. User should post ad.
Prerequisites	User should login first.
Test Procedure	<ol style="list-style-type: none"> <li>4. Select fields in form.</li> <li>5. Enter data in fields.</li> <li>6. Click add button.</li> </ol>
Test Data	Valid password: 123Abc@5, Abc567\$%9 Invalid password: 1_2ikhurram, _khurram12 Valid email: <a href="mailto:khurram@gmail.com">khurram@gmail.com</a> , 12ali@yahoo.com Invalid email: khurram.com, khurram@yahoo Valid Title: Product for sell Invalid Title: hjg### , ////..._ Invalid Brand: ./Dell Valid Brand: Dell, HP Valid Category: Cow, Sheep Invalid Category: 12cat, ./rabit Valid Price: 120000, 100 , Invalid Price: qamm555, ooop, AAA
Expected Result	<ol style="list-style-type: none"> <li>6. If email and password are valid by clicking login button then login.</li> <li>7. If title, brand, category and price are valid then store update ad data by clicking add button in ad detail.</li> <li>8. If given inputs are invalid then display error message.</li> <li>10. If fields are empty then display warning message.</li> </ol>
Actual Result	<ol style="list-style-type: none"> <li>4. If data is valid, the result is as expected.</li> <li>5. If data is not valid then invalid message displayed.</li> <li>6. If fields are empty then warning message displayed.</li> </ol>

<b>Status</b>	<b>Pass</b>
<b>Remarks</b>	<b>This test case is simple and easy.</b>
<b>Created By</b>	<b>Abdul waheed</b>
<b>Date of Creation</b>	<b>13/06/19</b>
<b>Executed By</b>	<b>Abdul waheed</b>
<b>Date of Execution</b>	<b>13/06/19</b>
<b>Text Environment</b>	<b>OS: Window 10 Browser: Google Chrome</b>

<b>Integration ID</b>	003
<b>Unit ID</b>	006, 007
<b>Testcase Summary</b>	To verify that user first search product and then show detail ad.
<b>Related Requirement</b>	User search product User should see product detail.
<b>Prerequisites</b>	User search product.
<b>Test Procedure</b>	<ol style="list-style-type: none"> <li>7. Select fields in form.</li> <li>8. Enter data in fields.</li> <li>9. Click add button.</li> </ol>
<b>Test Data</b>	Valid Title: Product for sell Invalid Title: hjg### , ////... _ Invalid Brand: ./cow Valid Brand: Cow, Sheep Valid Category: Cow, Sheep Invalid Category: 12 Cow , ./ Cow Valid Price: 120000, 100 Invalid Price: qamm555, ooop, AAA
<b>Expected Result</b>	<ol style="list-style-type: none"> <li>9. If user search product with valid data and then click on ad to show ad detail.</li> <li>10. If given inputs are invalid then display error message.</li> <li>11. If fields are empty then display warning message.</li> </ol>
<b>Actual Result</b>	<ol style="list-style-type: none"> <li>7. If data is valid, the result is as expected.</li> <li>8. If data is not valid then invalid message displayed.</li> <li>9. If fields are empty then warning message displayed.</li> </ol>
<b>Status</b>	Pass
<b>Remarks</b>	This test case is simple and easy.
<b>Created By</b>	Abdul Waheed
<b>Date of Creation</b>	13/06/19
<b>Executed By</b>	Abdul Waheed
<b>Date of Execution</b>	13/06/19

Text Environment	OS: Window 10 Browser: Google Chrome
------------------	---

Integration ID	004
Unit ID	003, 004
Testcase Summary	To verify that user first post ad and then update post ad.
Related Requirement	User should login. User should post ad. User should update ad.
Prerequisites	User should login first.
Test Procedure	10. Select fields in form. 11. Enter data in fields. 12. Click submit button.
Test Data	Valid password: 123Abc@5, Abc567\$%9 Invalid password: 1_2ikhurram, _khurram12 Valid email: <a href="mailto:khurram@gmail.com">khurram@gmail.com</a> , 12ali@yahoo.com Invalid email: khurram.com, khurram@yahoo Valid Title: Product on for sell Invalid Title: hjg### , /////..._ Invalid Brand: ./Dell Valid Brand: Cow shahiwal nasal, Sheep shahiwal nasal Valid Category: Sheep, Cow Invalid Category: cow, ./cow Valid Price: 120000, 100 Invalid Price: qamm555, ooop, AAA
Expected Result	12. If title, brand, category and price are valid then store update ad data by clicking add button in ad detail the update by clicking update button. 13. If given inputs are invalid then display error message. 14. If fields are empty then display warning message.
Actual Result	10. If data is valid, the result is as expected. 11. If data is not valid then invalid message displayed. 12. If fields are empty then warning message displayed.
Status	Pass
Remarks	This test case is simple and easy.

<b>Created By</b>	<b>Abdul Waheed</b>
<b>Date of Creation</b>	<b>13/06/19</b>
<b>Executed By</b>	<b>Abdul Waheed</b>
<b>Date of Execution</b>	<b>13/06/19</b>
<b>Text Environment</b>	<b>OS: Window 10 Browser: Google Chrome</b>

<b>Integration ID</b>	005
<b>Unit ID</b>	003, 005
<b>Testcase Summary</b>	To verify that user first post ad and then delete ad.
<b>Related Requirement</b>	User should login. User should post ad.
<b>Prerequisites</b>	User should login first. User post ad.
<b>Test Procedure</b>	13. Select fields in form. 14. Enter data in fields. 15. Click submit button.
<b>Test Data</b>	Valid Title: Product for sell Invalid Title: hjg###, /////... _ Invalid Brand: ./cow Valid Brand: cow, sheep Valid Category: Cow, Sheep Invalid Category: 12Cow, ./Cow Valid Price: 120000, 100 Invalid Price: qamm555, 00op, AAA
<b>Expected Result</b>	15. If title, brand, category and price are valid then store update ad data by clicking add button in ad detail then delete ad by clicking delete button. 16. If given inputs are invalid then display error message. 17. If fields are empty then display warning message.
<b>Actual Result</b>	13. If data is valid, the result is as expected. 14. If data is not valid then invalid message displayed. 15. If fields are empty then warning message displayed.
<b>Status</b>	Pass
<b>Remarks</b>	This test case is simple and easy.
<b>Created By</b>	Abdul Waheed
<b>Date of Creation</b>	13/06/19
<b>Executed By</b>	Abdul Waheed
<b>Date of Execution</b>	13/06/19
<b>Text Environment</b>	OS: Window 10

Browser: Google Chrome
------------------------

Integration ID	006
Unit ID	002, 008
Testcase Summary	To verify that admin login and then delete user account.
Related Requirement	User should login.
Prerequisites	User should login first.
Test Procedure	<ul style="list-style-type: none"> <li>16. Select fields in form.</li> <li>17. Enter data in fields.</li> <li>18. Click submit button.</li> </ul>
Test Data	Valid password: 123Abc@5, Abc567\$%9 Invalid password: 1_2 khurram, _ khurram 12 Valid email: <a href="mailto:khurram@gmail.com">khurram@gmail.com</a> , 12 khurram @yahoo.com Invalid email: khurram.com, khurram@yahoo
Expected Result	<ul style="list-style-type: none"> <li>18. If email and password data is valid then login.</li> <li>19. Delete user account by clicking delete button.</li> <li>20. If given inputs are invalid then display error message.</li> <li>21. If fields are empty then display warning message.</li> </ul>
Actual Result	<ul style="list-style-type: none"> <li>16. If data is valid, the result is as expected.</li> <li>17. If data is not valid then invalid message displayed.</li> <li>18. If fields are empty then warning message displayed.</li> </ul>
Status	Pass
Remarks	This test case is simple and easy.
Created By	Abdul Waheed
Date of Creation	13/06/19
Executed By	Abdul Waheed

Date of Execution	13/06/19
Text Environment	OS: Window 10 Browser: Google Chrome

## 6.7. Performance testing

## 6.8. Stress Testing

**Initial level:**

**Intermediate level:**

**Last Level:**

# **Chapter 7**

## **Summary, Conclusion and Future Enhancements**

## Chapter 7: Summary, Conclusion & Future Enhancements

### 7.1. Project Summary

Now that the project Maweshi Mandi has been completely developed, let us summarize it all. Maweshi Mandi is a platform where its users can sell and buy products like that Animals. The project included a website and its android mobile applications. They have been especially designed for the people of Pakistan since this is where it's going to operate.

Mobile application has been developed using react-native, a JavaScript framework. Since react native is multi-platform, the application (with slight changes) will run on android. React also supports Ajax based HTML requests, which was the main purpose of using this framework. It will give us a proper compiled interface where then we can use the data we are already using on our website.

### 7.2. Achievements and Improvements

The biggest achievements here is that we were able to compile all we learned in four years of studying BSCS and apply it to this project. We learned software architecture design techniques, UML modeling, project management, software quality assurance, testing and much more, and were able to apply it all in this project.

The next big achievement are the things we learnt during this project. New languages, frameworks, libraries, data handling techniques and API. All that will be useful for us in our futures. We were able to connect all our products via API. Change in data in one place will result in change in all of our products.

### 7.3. Lessons Learnt

#### What went well:

- Using a multi-platform language/framework for developing your application works well.

- Using an API to manage same data all across the system will be beneficial in the future

**What went wrong:**

- Using redux for small traditional non-complex applications will result in complicating the code for no reason.
- A project should be well managed and tasks completed within their timeframe, else they will affect the upcoming tasks.

## 7.4. Future Enhancements

As it has been already said, there is always room for further improvement. And since we plan to launch this website as our own startup, enhancements will keep coming.

The application has been built on react native, but at some point, we are definitely going to scale it up, for that we choose to use redux. The application will grow in size and complexity and handling changes in react native alone will start getting difficult. So, the best thing here is to use redux, which is kind of a state handler. It makes it easier for us manage a highly complex application, which we believe ours is eventually going to become.

Review and rating system is going to be added as a feature that will help improve the quality of service people provide to each other since this is a C2C business.

# Reference and **Bibliography**

## Reference and Bibliography

- Executive Summary | Online Shopping | E Commerce. [online] Scribd. Available at: <https://www.scribd.com/document/50093622/Executive-Summary> [Accessed 8 Oct. 2018].
- Status.net Business Templates. (2018). How to Write Project Report: Complete Step-By-Step Guide. [online] Available at: <https://status.net/templates/project-report/> [Accessed 9 Oct. 2018].
- ProjectManager.com. (2018). The Ultimate Guide to Gantt Charts - ProjectManager.com. [online] Available at: <https://www.projectmanager.com/gantt-chart> [Accessed 9 Oct. 2018].
- Edrawsoft.com. (2018). Architecture Diagram Overview. [online] Available at: <https://www.edrawsoft.com/architecture-diagram.php> [Accessed 9 Oct. 2018].
- Totallycommunications.com. (2018). How to define the scope of a project - Totally Communications. [online] Available at: <https://www.totallycommunications.com/latest/how-to-define-the-scope-of-a-project/> [Accessed 9 Oct. 2018].
- Teamgantt.com. (2018). How to Write a Good Project Plan in 10 Steps | Project Management Guide. [online] Available at: <https://www.teamgantt.com/guide-to-project-management/how-to-plan-a-project> [Accessed 9 Oct. 2018].
- Scribd. (2018). SRS for Apni Dukan(an Onineline Shopping Website) | Online Shopping | Use Case. [online] Available at: <https://www.scribd.com/doc/40520541/SRS-for-Apni-Dukan-an-Onineline-Shopping->
- Anon,(2018).[online]Availableat:[http://www.cse.chalmers.se/~feldt/courses/reqeng/examples/srs\\_example\\_2010\\_group2.pdf](http://www.cse.chalmers.se/~feldt/courses/reqeng/examples/srs_example_2010_group2.pdf) [Accessed 24 Nov. 2018].
- IEEE standard document for software requirement specification.