**SUPERIOR UNIVERSITY LAHORE**

|  |
| --- |
| Superior Logo |

**Faculty of Computer Science & IT**

**Final Year Project**

**PROJECT REPORT**

**[****Pet’s App]**

Project ID: **FYP-BCSM-S20-**

**Project Team**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Student Name** | **Student ID** | **Program** | **Contact Number** | **Email Address** |
| Hafiza Kainat Khalid | BCSM-S17-046 | BCS | 03054438203 | BCSM-S17-046@superior.edu.pk |
| Aqsa | BCSM-F16-198 | BCS |  | BCSM-F16-198@superior.edu.pk |
| Emran Hameed | BCSM-S17-068 | BCS |  | BCSM-S17-068@superior.edu.pk |

**Dr Sheheryar Malik**

[Dean]

**Project Report**

**Pet’s App**

**Change Record**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Author(s)** | **Version** | **Date** | **Notes** | **Supervisor’s Signature** |
|  | 1.0 |  | <Original Draft> |  |
|  |  |  | <Changes Based on Feedback from Supervisor> |  |
|  |  |  | <Changes Based on Feedback From Faculty> |  |
|  |  |  | <Added Project Plan> |  |
|  |  |  | <Changes Based on Feedback from Supervisor> |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

**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 our parents who support us to do such a great thing. And as well as our supervisor DR SHEHERYAR MALIK to help or train us to complete this work. We thankful to our parents and supervisor to help us.*

*.*

# 

# 

# Acknowledgements

I am really thankful to my supervisor who has . . . . . . . . . .

# Executive Summary

This app manages all of your pets’ details from one simple app. We are going to develop app at which we will facilitate the Pet owners to make appointment and provide collaborating environment between vet and Pet owners (chat) and house call, funerary services, disaster triage and pet’s owner create own and pet profile, medications and allergies, weight history, height history and more and pet’s owner and vet create profiles or login with face-book or google Account or scan QR code for login. The variety of customizable features make this a convenient and easy-to-use app. you can also chat with the vet tell and ask about your pet and working alongside with map vet tracking system, the map makes it easy to track where vet is if he or she near the house. Use the app view your pet’s activity levels in veterinary hospital know about your pet’s health. Pet app could save your animal’s life. This pet first aid app can help you figure out what’s wrong with your pet before rushing to the animal hospital. It also provides the information about your pet’s behavior mild, or do you need to rush him to the vet now? This app aims to solve these issues. You’ll even get access to resources like articles and videos that teach you how to care for your pet in certain situations, such as when choking. You can also record your pet’s medical information so you have it on-hand when you go to the vet. Have you ever wondered what types of plants in and around your home are toxic to your pets? Or maybe you’ve wondered what you can feed them without them getting sick. Now you don’t have to worry so much. With the pet app, you’ll receive information of over 100 items toxic to cats and dogs. This helps you decide what flowers to plant, if it’s okay to share your leftovers, and if you should contact the vet when your pet eat some-thing you’re unsure off. The Pet app is managing everything pet related. Among its extensive list of tools, you can create profiles for any type of pet and enter birth dates, feeding and walking schedules, and vet appointments. When you hire a pet sitter, simply share your pet’s profile via email. You can also use the app to search for nearby vet clinics.

# Table of Contents

[Dedication 4](#_Toc14122728)

[Acknowledgements 5](#_Toc14122729)

[Executive Summary 6](#_Toc14122730)

[Table of Contents 7](#_Toc14122731)

[List of Figures 10](#_Toc14122732)

[List of Tables 11](#_Toc14122733)

[Chapter 1 12](#_Toc14122734)

[Introduction 12](#_Toc14122735)

[1.1. Background 13](#_Toc14122736)

[1.2. Motivations and Challenges 13](#_Toc14122737)

[1.3. Goals and Objectives 14](#_Toc14122738)

[1.4. Literature Review/Existing Solutions 14](#_Toc14122739)

[1.5. Gap Analysis 15](#_Toc14122740)

[1.6. Proposed Solution 15](#_Toc14122741)

[1.7. Project Plan 16](#_Toc14122742)

[1.7.1. Work Breakdown Structure 17](#_Toc14122743)

[1.7.2. Roles & Responsibility Matrix 17](#_Toc14122744)

[1.7.3. Gantt Chart 18](#_Toc14122745)

[1.8. Report Outline 19](#_Toc14122746)

[Chapter 2 20](#_Toc14122747)

[Software Requirement Specifications 20](#_Toc14122748)

[2.1. Introduction 21](#_Toc14122750)

[2.1.1. Purpose 21](#_Toc14122751)

[2.1.2. Document Conventions 21](#_Toc14122752)

[2.1.3. Intended Audience and Reading Suggestions 22](#_Toc14122753)

[2.1.4. Product Scope 22](#_Toc14122754)

[2.1.5. References 24](#_Toc14122755)

[2.2. Overall Description 24](#_Toc14122756)

[2.2.1. Product Perspective 24](#_Toc14122757)

[2.2.2. Product Functions 25](#_Toc14122758)

[2.2.3. User Classes and Characteristics 26](#_Toc14122759)

[2.2.4. Operating Environment 26](#_Toc14122760)

[Design and Implementation Constraints 27](#_Toc14122761)

[2.2.5. User Documentation 27](#_Toc14122762)

[2.2.6. Assumptions and Dependencies 27](#_Toc14122763)

[2.3. External Interface Requirements 27](#_Toc14122764)

[2.3.1. User Interfaces 27](#_Toc14122765)

[The User interface will be implemented using local host. This interface will be faculty friendly.so that any kind of faculty can place the order easily. Faculty can also give feedback easily through it with some demo comments or if they are keen to write their review by own they can do it. 27](#_Toc14122766)

[ Login and sign up page 28](#_Toc14122767)

[2.3.2. Hardware Interfaces 29](#_Toc14122768)

[2.3.3. Software Interfaces 29](#_Toc14122769)

[2.3.4. Communications Interfaces 30](#_Toc14122770)

[2.4. System Features 30](#_Toc14122771)

[2.4.1. System Feature 1 31](#_Toc14122772)

[2.4.1.1. Description and Priority 31](#_Toc14122773)

[2.4.1.2. Stimulus/Response Sequences 31](#_Toc14122774)

[2.4.1.3. Functional Requirements 31](#_Toc14122775)

[1.1.1. Order Special Items 31](#_Toc14122776)

[2.4.1.4. Description and Priority 32](#_Toc14122777)

[2.4.1.5. Stimulus/Response Sequences 32](#_Toc14122778)

[2.4.1.6. Functional Requirements 32](#_Toc14122779)

[2.5. Other Nonfunctional Requirements 32](#_Toc14122780)

[2.5.1. Performance Requirements 32](#_Toc14122781)

[2.5.2. Safety Requirements 33](#_Toc14122782)

[2.5.3. Security Requirements 33](#_Toc14122783)

[2.5.4. Software Quality Attributes 33](#_Toc14122784)

[3.1.1. Business Rules 34](#_Toc14122785)

[3.2. Other Requirements 34](#_Toc14122786)

[Chapter 3 35](#_Toc14122787)

[Use Case Analysis 35](#_Toc14122788)

[3.1. Use Case Model 36](#_Toc14122789)

[3.2. Use Case Descriptions 37](#_Toc14122790)

[Chapter 4 44](#_Toc14122791)

[System Design 44](#_Toc14122792)

[4.1. Architecture Diagram 45](#_Toc14122793)

[4.2. Domain Model 46](#_Toc14122794)

[4.3. Entity Relationship Diagram with data dictionary 47](#_Toc14122795)

[4.4. Class Diagram 48](#_Toc14122796)

[4.5. Sequence / Collaboration Diagram 49](#_Toc14122797)

[4.6. Activity Diagram 50](#_Toc14122798)

[4.7. State Transition Diagram 52](#_Toc14122799)

[4.8. Component Diagram 53](#_Toc14122800)

[4.9. Deployment Diagram 54](#_Toc14122801)

[4.10. Data Flow diagram 55](#_Toc14122802)

[Chapter 5 56](#_Toc14122803)

[Implementation 56](#_Toc14122804)

[5.1. Important Flow Control/Pseudo codes 57](#_Toc14122805)

[5.2. Components, Libraries, Web Services and stubs 57](#_Toc14122806)

[5.3. Deployment Environment 58](#_Toc14122807)

[5.4. Tools and Techniques 58](#_Toc14122808)

[5.5. Best Practices / Coding Standards 58](#_Toc14122809)

[5.6. Version Control 59](#_Toc14122810)

[Chapter 6 60](#_Toc14122811)

[Testing and Evaluation 60](#_Toc14122812)

[6.1. Use Case Testing 61](#_Toc14122813)

[6.2. Equivalence partitioning 62](#_Toc14122814)

[6.3. Boundary value analysis 63](#_Toc14122815)

[6.4. Data flow testing 64](#_Toc14122816)

[6.5. Unit testing 64](#_Toc14122817)

[6.6. Integration testing 65](#_Toc14122818)

[6.7. Performance testing 65](#_Toc14122819)

[6.8. Stress Testing 67](#_Toc14122820)

[Chapter 7 68](#_Toc14122821)

[Summary, Conclusion and Future Enhancements 68](#_Toc14122822)

[7.1. Project Summary 69](#_Toc14122823)

[7.2. Achievements and Improvements 69](#_Toc14122824)

[7.3. Critical Review 69](#_Toc14122825)

[7.4. Lessons Learnt 69](#_Toc14122826)

[7.5. Future Enhancements/Recommendations 70](#_Toc14122827)

[Reference and Bibliography 72](#_Toc14122828)

[Index 74](#_Toc14122829)

# 

# List of Figures

[Figure 1 Work Breakdown Structure 17](#_Toc14122086)

[Figure 2 Gantt Chart 18](#_Toc14122087)

[Figure 3 Use Case Model 36](#_Toc14122088)

[Figure 4 System Architecture 45](#_Toc14122089)

[Figure 5 Domain Model 46](#_Toc14122090)

[Figure 6 Entity Relationship Diagram 47](#_Toc14122091)

[Figure 7 Class Diagram 48](#_Toc14122092)

[Figure 8 Sequence Diagram 49](#_Toc14122093)

[Figure 9 Sequence Diagram 51](#_Toc14122094)

[Figure 10 State Transition Diagram 52](#_Toc14122095)

[Figure 11 Component Diagram 53](#_Toc14122096)

[Figure 12 Deployment Diagram 54](#_Toc14122097)

[Figure 13 Data Flow Diagram 55](#_Toc14122098)

[Figure 14 Flow Control 57](#_Toc14122099)

[Figure 15 Version Control 59](#_Toc14122100)

[Figure 16 Use Case Testing 61](#_Toc14122101)

# List of Tables

[Table 1 Literature Review 15](#_Toc14122496)

[Table 2 Roles And Responsibility Matrix 18](#_Toc14122497)

[Table 3 Document Conventions 22](#_Toc14122498)

[Table 4 User Classes and Characteristics 27](#_Toc14122499)

[Table 5 User Interfaces 29](#_Toc14122500)

[Table 6 Use Case Descriptions 38](#_Toc14122501)

[Table 7 Use Case Testing 64](#_Toc14122502)

[Table 8 Equivalence Partition 65](#_Toc14122503)

[Table 9 Boundary Value Analysis 65](#_Toc14122504)

# Chapter 1

# Introduction

**Chapter 1:** Introduction

This app manages all of your pets’ details from one simple app. With Phone, you can create profiles for multiple pets detailing their vet appointments, medications and allergies, weight history, height history and more and pet’s owner and vet create profiles or login with face-book or google Account or scan QR code for login. The variety of customizable features make this a convenient and easy-to-use app. you can also chat with the vet tell and ask about your pet and working alongside with map vet tracking system, the map makes it easy to track where vet is if he or she near the house. Use the app view your pet’s activity levels in veterinary hospital know about your pet’s health. Pet app could save your animal’s life. This pet first aid app can help you figure out what’s wrong with your pet before rushing to the animal hospital. It also provides the information about your pet’s behavior mild, or do you need to rush him to the vet now? This app aims to solve these issues. You’ll even get access to resources like articles and videos that teach you how to care for your pet in certain situations, such as when choking. You can also record your pet’s medical information so you have it on-hand when you go to the vet. Have you ever wondered what types of plants in and around your home are toxic to your pets? Or maybe you’ve wondered what you can feed them without them getting sick. Now you don’t have to worry so much. With the pet app, you’ll receive information of over 100 items toxic to cats and dogs. This helps you decide what flowers to plant, if it’s okay to share your leftovers, and if you should contact the vet when your pet eat some-thing you’re unsure off. The Pet app is managing everything pet related. Among its extensive list of tools, you can create profiles for any type of pet and enter birth dates, feeding and walking schedules, and vet appointments. When you hire a pet sitter, simply share your pet’s profile via email. You can also use the app to search for nearby vet clinics.

## Background

### The background of this project is an idea that came from **“pet rescuers home again”** and **“Bonus: pet diabetes tracker”** that if these Apps can work for pets it can also protect pets. We saw the need of a separate system for pets and we got this idea for online **“pet rescuers home again”**. Many other platforms like **“Bonus: pet diabetes tracker”** have their single franchise and they are performing it very well. They have similarities like us by lack some additional features that we want to implement in this project. These are the system to create profiles of vets and Pet owners and also tell and ask about pet health via chat between vet and Pet owners. We are going to make a platform which will complete take care of pets and make them valuable.

## Motivations and Challenges

**Motivations:**

The main motivation for this project were pet apps because they worked long time ago for pet’s health and happiness. But many things are missing they were not getting the required facilities and needs in these apps. We often saw them standing and waiting for vet in vet-clinic. That had tremendous impact on their (Pet owners) time schedule.

**Challenges:**

There were not many challenges except some small ones. Because there has been no such system deployed in play store. Some challenges are in development because modules were tough enough for us.

## Goals and Objectives

Our goal is to save the Pet owners valuable time and provide their pets easy treatment and tell and ask about pet’s health through chat.

Our objective is to ease the manageability of pet app providing them with facilities like:

* Crete profile and upload status.
* Provide facility of house call
* Provide facility of funerary services.
* Provide facility of Rescue and disaster.
* To allow Botanist provide information of plants injurious for pet’s health.
* To allow Blogger upload videos.
* To allow chat between vet and Pet owners.

## Literature Review/Existing Solutions

This feature is specific for the pet’s app. When we choose this project before we have searched on the internet for such kind of app but have not succeeded in doing so. Pet’s app will not that much advanced in these matters and also do not provide the facilities to the pet or Pet owners. This project is specifically chosen for the Pet owners.

The products that we have looked forward to online are as follows.

Table 1 Literature Review

|  |  |
| --- | --- |
| **Websites** | **Description** |
| <https://play.google.com/store/apps/details?id=com.flightpath.merkdiabetes&hl=en> | This app is mainly focused on pet health. |
| <https://www.homeagain.com/petrescuers.html> | This app is mainly focused on Provide facility of house call,  funerary services,  and Rescue disaster etc. |
|  |  |

## Gap Analysis

Now we discuss the gap between our client environment and our software system. Customers and Company has lack of communication. They are coming and make appointment for pet which consumes a lot of time. They are not providing online make appointment. There’s big gap between the software system and client environment. For covering this gap, we have to implement our all system requirement

1. **Organization:**

To start a new product that has not been launch in play store to start a new trend.

1. **Business Directions:**

It will start a new trend in the market providing facilities to Pet owners.

## Proposed Solution

We are going to create app at which we will facilitate the Pet owners to create profile and make appointment and chat between vet and Pet owners and house call, funerary services and triage disasters.

**Solutions of the problems regarding pet treatment:**

* Pet owners will have the complete access to the app and his information. Pet owners will not have to wait in vet clinic for pet treatment.
* Pet owners will not have to wait for make appointment.
* Pet owners will have their reviews about the clinic or treatment and also about the pet health. Faculty will have an option to cancel and make appointments.
* Pet owners will also have an option to upload status.
* Pet owners will also have an option to create pet profile and edit and delete profile of pet.
* vet will also have an option of house call, funerary services and rescue disaster.
* vet will also have an option to update pet history.
* vet will also have an option to chat with pet parents.
* vet will also have an option to upload status.

## Project Plan

As we are going through a proper way to build out system so here is our project plan which is further explained in Gant chart.

1. Requirement
2. Design
3. Implementation
4. Testing
5. Maintenance

Estimated days of work are 41 days, which can be a little bit less or more.

We will collect requirements by using different techniques, which are interviews one by one, doing surveys, and questionnaires, which will help us to meet the requirements according to demand.

Designing is a difficult phase where we can’t do a single mistake as a single mistake will ruin our project. Same as designing, implementation is much difficult and public accepts make it difficult.

Testing is an initial phase where we will find some errors if occur. Maintenance will play an important role of success. Maintaining our project and update it according to need will make it successful.

## Work Breakdown Structure

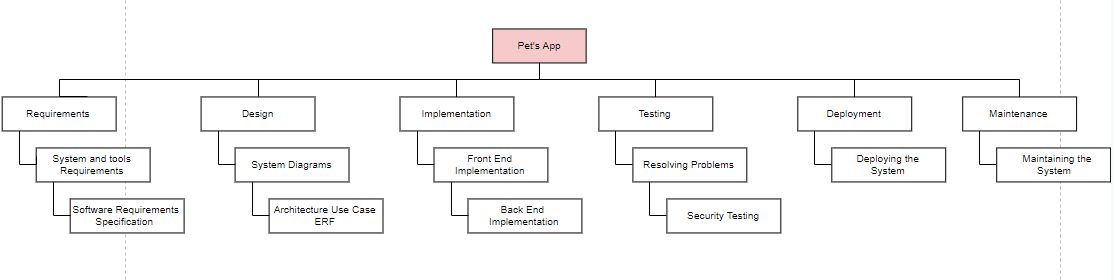


Figure 1 Work Breakdown Structure

## Roles & Responsibility Matrix

Table 2 Roles and Responsibility Matrix

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **WBS #** | **WBS Deliverable** | **Activity #** | **Activity to Complete the Deliverable** | **Duration**  **(# of Days)** | **Responsible Team Member(s) & Role(s)** |
| 01 | Project Proposal | 01 | Planning | 15 |  |
| 02 | Documentation | 02 | software requirements specification (SRS) | 8 |  |
| 03 | System analyzing | 03 | Market Analysis | 4 |  |
| Surveys | 3 |
| Research | 3 |
| 04 | System designing | 04 | Flow chart | 4 |  |
| Use case diagram | 5 |
| ERD | 2 |
| Sequence Diagram | 3 |
| 05 | User Interface Design | 05 | Diagrams | 7 |  |
| 06 | Front end Coding | 06 | XML | 20 |  |
| 07 | Back End Coding | 07 | Java Web API’s (PHP), | 40 |  |
| 08 | Testing | 08 | Tools for testing | 8 |  |
| 09 | Maintenance | 09 | Search defects | 8 |  |
| 10 | Final Documentation | 10 | MS word, MS Project, MS Visio documentation | 15 |  |

## Gantt Chart

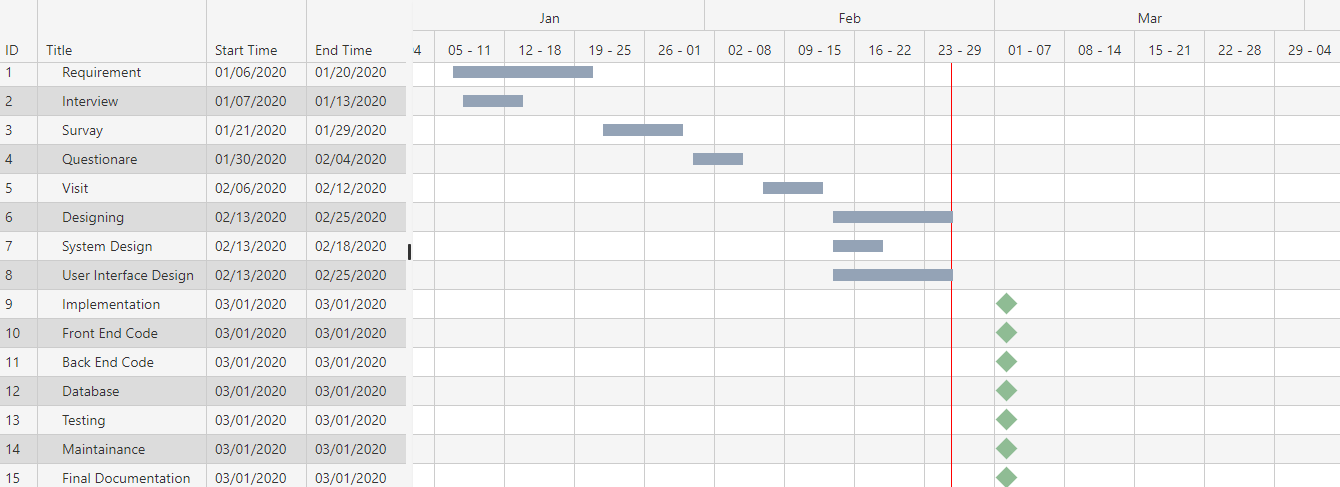


Figure 2 Gantt Chart

## Report Outline

* **Introduction**

What we are going to do?

* **Background**

What currently happening in the market.

* **Motivation & challenges**

Discussed in detail.

* **Goals & objectives**

Discussed how to achieve our goals.

* **Existing system**

Discussed what unique we are having. How could survive in market.

* **Gap Analysis**

Discussed Gap analysis & discussed what our planned strategy is.

* **Proposed Solution**

Discussed the solution of the problems.

* **Project Plan**

Made a project plan so we could complete our project in time.

* **Work Breakdown Structure**

Discussed in detail.

* **Rules & Responsibility**

Divided task in team members.

* **Gant Chart**

Made a chart so could complete our project in time.

# Chapter 2

# Software Requirement Specifications

**Chapter 2:** Software Requirement Specifications



## Introduction

## Purpose

This document presents the detailed explanation of the objective, feature, vet interface and pet owner’s interface and applications of Pet’s App in real life. It also describes how the system performs and under which condition it must operate. In this document there will be listed all the features and functions of the software and its requirements.

## Document Conventions

Table 3 Document Conventions

|  |  |
| --- | --- |
| **SHORT FORM** | **FULL FORM** |
| ERD | Entity relationship diagram |
| DB | Database |
| SRS | Software requirement specification |
| UML | Unified Modeling Language |
| DFD | Data flow diagram |
| WBS | Work breakdown structure |
| IDE | Integrated development environment |
| GUI | Graphical user interface |
| HTTP | Hypertext transfer protocol |
| FTP | File transfer Protocol |
| WAN | Wide Area network |
| LAN | Local area network |

## 

## Intended Audience and Reading Suggestions

It is restricted within the university premises. This has been implemented under the guidance of university professors. This project is useful for pet parents and as well as vet or vet clinic.

Reading suggestions are to start reading from the top till bottom and read that portion again if can’t understand what it means.

## Product Scope

This app manages all of your pets’ details from one simple app. With Phone, you can create profiles for multiple pets detailing their vet appointments, medications and allergies, weight history, height history and more and pet’s owner and vet create profiles or login with face-book or google Account or scan QR code for login. The variety of customizable features make this a convenient and easy-to-use app. you can also chat with the vet tell and ask about your pet. It working alongside with map vet tracking system, the map makes it easy to track where vet is if he or she near the house.

## References

* <https://krazytech.com/projects> we used this website to get help related to SRS.

## Overall Description

## Product Perspective

This app will be a replacement of all the existing solution that is currently providing solutions for the pet’s health problems. This solution is emphasizing on all the draw backs of the solution that occurs in different homes and vet clinics. These solutions are mostly un-customizable and mostly expensive and lack future support whether our solution will be more customizable, will be free of cost and will support future possibilities. By using our system, we can make online Appointment, tell and ask about pet’s health via chat, watch videos and read blogs for pet’s happiness and health and tracking vet easily.

## Product Functions

The whole functions will perform through this order:

* Log in: This function authenticates the pet parent.
* Vet tracking: this function will display nearest vet clinic of house.
* Make appointment: this function will display vet available time.
* Create pet profile: this function gave the fields to write pet name, age, specie, breed.
* chat: tell and ask about pet health.
* Disaster triage: This feature will allow to house call and funerary services.
* Plants information: This function display required or concerned plant details for pet health uploaded from botanist.
* Blogs and videos: This function show blogs and videos for pet health and happiness uploaded by bloggers.

## User Classes and Characteristics

Table 4 User Classes and Characteristics

|  |  |  |  |
| --- | --- | --- | --- |
| **Classes** | **Functionality Privilege** | **Technical/Non-Technical** | |
| Pet Owner’s | Create profile, edit profile, chat  And upload status, View homepage, view videos, view blogs, Make-appointment, cancel appointment and create pet profile. | Technical | |
| Blogger | Create profile, edit profile, chat  And upload status, View homepage | Technical | |
| Botanist | Create profile, edit profile, chat  And upload status, View homepage | Technical | |
| Vet | Create profile, edit profile, chat, confirm Appointment, view homepage, And upload status | Technical |

## Operating Environment

* For Database (Use for Record like add profile, delete profile, update profile) We Will Use Tool: SQLite.
* In the development of this app we are following Spiral Model.
* For front end and back end coding including user interface (server side, user side) and database connectivity with front end interfaces we used Android studio and java JDK.
* We used windows 7, 8, 8.1, 10 as operating systems.
* We used firebase to setup local servers.
* For this project, we used XML and Java as programming languages.
* Used Google chrome as web browser.

## Design and Implementation Constraints

* We should follow the IEEE standards.
* Default language will be English.
* Project will follow all the copyright and cyber lines of PTA (Pakistan Telecommunication Authority).

## User Documentation

It will provide specific guidelines to a pet parents for using the Pet’s app. Further video (slide show) will be provided which will represent the whole system functions and how it works.

## Assumptions and Dependencies

Pet’s app is a self-contained project, the only explicit dependencies are those stated in the Design and Implementation Constraints.

## External Interface Requirements

## 

## User Interfaces

## User interface is a key component of any system. User interface must be made easy to understand and use. User interfaces are simple and to the point, any irrelevant information is hidden from that interface. For that we use some standards to provide our users the best interface experience possible.

## Hardware Interfaces

The server is directly connected to the client system also the client has to access the database for accessing the account details and storing the log in time.

The client access to the database in the server is read only.

* Hardware requirements for Insurance on internet will be same for both the parties which are follows:
* Processor: Any
* RAM: - 2 GB or above.
* HD: - 20 GB or above.
* Only the recommended configuration (basic requirements of a computer system) no other specific hardware is required to run the software.

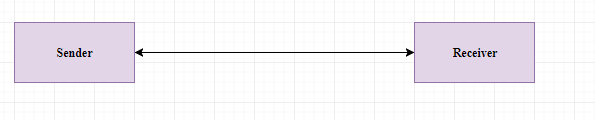
Our System will work through internet connectivity. We will use all those hardware devices and configuration that will use to connect with internet. We need the modems, wires, WAN, LAN networks and Ethernet Cross cables etc.

## Software Interfaces

The system will run on SQL server so it requires any scripting language like Java, XML etc. The systems require Data Base also for the store the data of the system like SQLite. User need web browser for interact with the system. Operating system must be with GUI supportability like windows (XP and higher versions), Linux, Macintosh (leopard or OSX version or higher), android OS (jellybean or higher). To access database required SQLite. Android studio.

## Communications Interfaces

Communication function required the internet protocol version 6 and it will follow HTTPS. It will use FTP for whole system with local server.



## System Features

* Log in: This function authenticates the pet’s owner.
* Vet tracking: this function will display nearest vet clinic of house.
* Make appointment: this function will display vet available time.
* Create pet profile: this function gave the fields to write pet name, age, specie, breed.
* chat: tell and ask about pet health.
* Disaster triage: This feature will allow to house call and funerary services.
* Plants information: This function display required or concerned plant details for pet uploaded from botanist.
* Blogs and videos: This function show blogs and videos for pet health and happiness uploaded by bloggers.

## System Feature 1

This feature is one of the main features listed. This will provide users to get themselves a rescue disaster and make appointment and then can order medicines.

## Vet

## Description and Priority

**User Action:** User must click on the app.

**Response:** system will be started and show the signup page.

**User Action:** If user is already registered then he will click on the login buttonotherwise he/she will have registered by entering required information.

**Response:** System will first check whether the user already registered into the system or not. If theuser already registered in then the user will login with system otherwise system will ask the user to

login into his google account or face book account and then it will show the home page after user logged in.

**User Action:** User can change his account settings and then click on save changing button **Response:** System will first validate the data according to system policy and then checkwhether the data has been taken by the existing user are not. If the data is valid and not already taken by the existing user then it will save changes that the user made.

**User Action:** User can accept the request and then click on confirm button.

**Response:** System will first validate the appointment and then save on the server and send the confirmation message to the user. If the data is already taken by the existing user then it will not save.

**User Action:** if user want to collaborate with each other and then he/she write message and click on send button.

**Response:** System will first validate the user and then send message to the user. If the user not existing then it will not send.

**User Action:** If user is already created profile then he/she will click on the edit buttonotherwise he/she will create his/her profile by entering required information.

**Response:** System will first check whether the user already create profile into the system or not. If theuser already created then the user will edit his/her profile otherwise system will ask the user to

Create profile and then it will able to use all features of app.

**User Action:** If user is already created pet history then he/she will click on the edit buttonotherwise he/she will create pet history by entering required information.

**Response:** System will first check whether the user already create pet history into the system or not. If theuser already created then the user will edit pet history otherwise system will ask the user to

Create pet history.

## Functional Requirements

* REQ-SF1-1: Validating vet Registration
* REQ-SF1-2: Sign In
* REQ-SF1-3: confirm appointment
* REQ-SF1-4: chat
* REQ-SF1-5: create profile update profile
* REQ-SF1-6: Vet update history of pet

## Pet’s owner

## Description and Priority

**User Action:** User must click on the app.

**Response:** system will be started and show the signup page.

**User Action:** If user is already registered then he will click on the login buttonotherwise he/she will have registered by entering required information.

**Response:** System will first check whether the user already registered into the system or not. If theuser already registered in then the user will login with system otherwise system will ask the user to

login into his google account or face book account and then it will show the home page after user logged in.

**User Action:** User can change his account settings and then click on save changing button **Response:** System will first validate the data according to system policy and then checkwhether the data has been taken by the existing user are not. If the data is valid and not already taken by the existing user then it will save changes that the user made.

**User Action:** User can make the appointment and then click on send button. And if user want to cancel appointment then click on the cancel button.

**Response:** System will first validate the appointment and then save on the server and send the message to the user. If the data is already taken by the existing user then it will not send. And if the appointment canceled by user in 15 min then system will not send the request to the required user otherwise it will send.

**User Action:** if user want to collaborate with each other and then he/she write message and click on send button.

**Response:** System will first validate the user and then send message to the user. If the user not existing then it will not send.

**User Action:** If user is already created profile then he/she will click on the edit buttonotherwise he/she will create his/her profile by entering required information.

**Response:** System will first check whether the user already create profile into the system or not. If theuser already created then the user will edit his/her profile otherwise system will ask the user to

Create profile and then it will able to use all features of app.

**User Action:** If user is already created pet profile then he/she will click on the edit buttonotherwise he/she will create pet profile by entering required information.

**Response:** System will first check whether the user already create pet profile into the system or not. If theuser already created then the user will edit pet profile otherwise system will ask the user to

Create pet profile.

## Functional Requirements

* REQ-SF5-1: Validating pet owner’s Registration
* REQ-SF5-2: Sign In
* REQ-SF5-3: Make appointment, cancel appointment
* REQ-SF5-4: chat
* REQ-SF5-5: create profile update profile
* REQ-SF5-6: Create pet profile, update profile

## Blogger

## Description and Priority

**User Action:** User must click on the app.

**Response:** system will be started and show the signup page.

**User Action:** If user is already registered then he will click on the login buttonotherwise he/she will have registered by entering required information.

**Response:** System will first check whether the user already registered into the system or not. If theuser already registered in then the user will login with system otherwise system will ask the user to

login into his google account or face book account and then it will show the home page after user logged in.

**User Action:** User can change his account settings and then click on save changing button **Response:** System will first validate the data according to system policy and then checkwhether the data has been taken by the existing user are not. If the data is valid and not already taken by the existing user then it will save changes that the user made.

**User Action:** If user is already created profile then he/she will click on the edit buttonotherwise he/she will create his/her profile by entering required information.

**Response:** System will first check whether the user already create profile into the system or not. If theuser already created then the user will edit his/her profile otherwise system will ask the user to

Create profile and then it will able to use all features of app.

**User Action:** If user is already uploaded video then he/she will click on the delete buttonotherwise he/she will be upload video.

**Response:** System will first check whether the user already upload video into the system or not. If theuser already uploaded video then the user will delete video otherwise not.

## Functional Requirements

* REQ-SF6-1: Validating blogger Registration
* REQ-SF6-2: Sign In
* REQ-SF6-3: upload videos, delete videos
* REQ-SF6-4: View home page
* REQ-SF6-5: create profile update profile

## Botanist

## Description and Priority

**User Action:** User must click on the app.

**Response:** system will be started and show the signup page.

**User Action:** If user is already registered then he will click on the login buttonotherwise he/she will have registered by entering required information.

**Response:** System will first check whether the user already registered into the system or not. If theuser already registered in then the user will login with system otherwise system will ask the user to

login into his google account or face book account and then it will show the home page after user logged in.

**User Action:** User can change his account settings and then click on save changing button **Response:** System will first validate the data according to system policy and then checkwhether the data has been taken by the existing user are not. If the data is valid and not already taken by the existing user then it will save changes that the user made.

**User Action:** If user is already created profile then he/she will click on the edit buttonotherwise he/she will create his/her profile by entering required information.

**Response:** System will first check whether the user already create profile into the system or not. If theuser already created then the user will edit his/her profile otherwise system will ask the user to

Create profile and then it will able to use all features of app.

**User Action:** If user is already uploaded images and description then he/she will click on the delete buttonotherwise he/she will be upload images and description.

**Response:** System will first check whether the user already upload images and description into the system or not. If theuser already uploaded images and description then the user will delete images and description otherwise not

## Functional Requirements

* REQ-SF7-1: Validating botanist Registration
* REQ-SF7-2: Sign In
* REQ-SF7-3: upload images, delete images
* REQ-SF7-4: View home page
* REQ-SF7-5: create profile update profile

## Other Nonfunctional Requirements

## Performance Requirements

* The product will be based on SQLite.
* The product will take the initial load time.
* App information will be fully secure through POS system.
* Faster Searching algorithms will be used to make searches appear faster.

## Safety Requirements

* The source code developed for the system shall be maintained in the Android studio.
* The whole system secured only vet or pet’s owner can access the all data.
* The system will use HTTPS because it is more secure.
* The system uses POS system.

## Security Requirements

The security feature from having a log in for all pet’s owner personal information. The log in detailed will be used in the app also. That situation the chances of app getting slow down.

## Software Quality Attributes

Correctness and reliability are the major quality attributes of this product. Correct fetching of data is high priority as everything is depending on it. Our product needs to be a reliable source as it is being used to manage a large amount of people. Even one error can play a huge role in the downfall of this project. Robustness and maintainability are also some attributes which would be kept in mind while preparing and installing this project as it has to use on daily basis.

## Business Rules

**Pet’s owner:**

A Pet’s owner can create profile, edit profile, chat, upload status, View homepage, view videos, view blogs, make appointment, cancel appointment and create pet profile.

**Vet:**

Vet can create profile, edit profile, chat, confirm appointment, view homepage, and upload status.

**Blogger:**

Blogger can create profile, edit profile, upload blogs and videos, View homepage.

**Botanist:**

Botanist can create profile, edit profile, upload images, View homepage.

## Other Requirements

1. Database which we will be using to save and record the data of the student should be reliable and big enough to store all the data and keep the record saved for a specified period of time.

# 

# 

# 

# Chapter 3

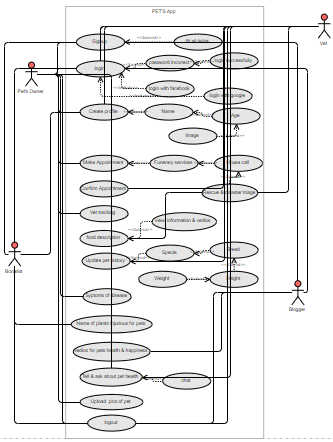
# Use Case Analysis

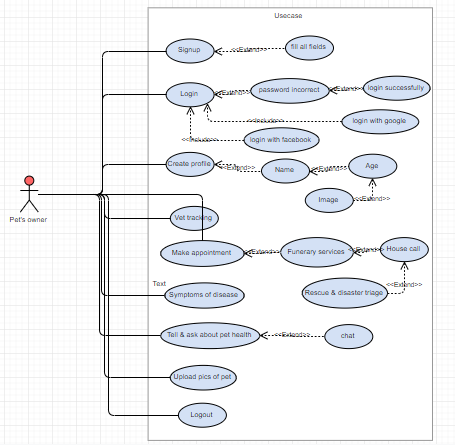
**Chapter 3:** System Analysis

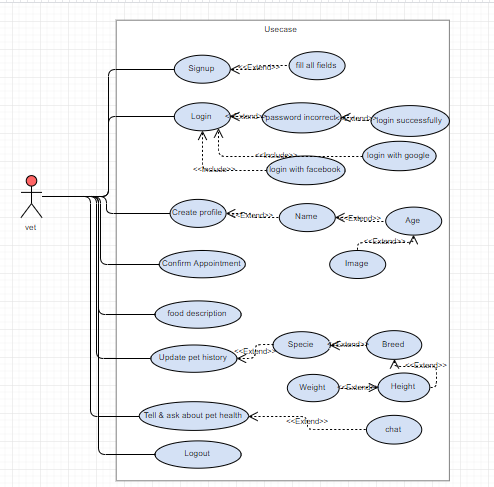
Creating use cases to describing the whole scenario how it is going to work. Describing actors & their purposes. How users will Log In, create accounts make purchases. And users can edit order within specific time limit.

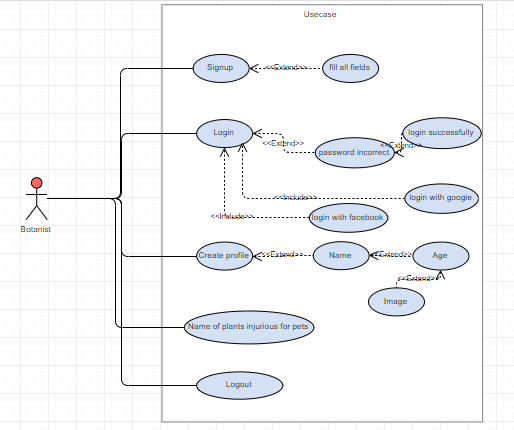
## Use Case Model

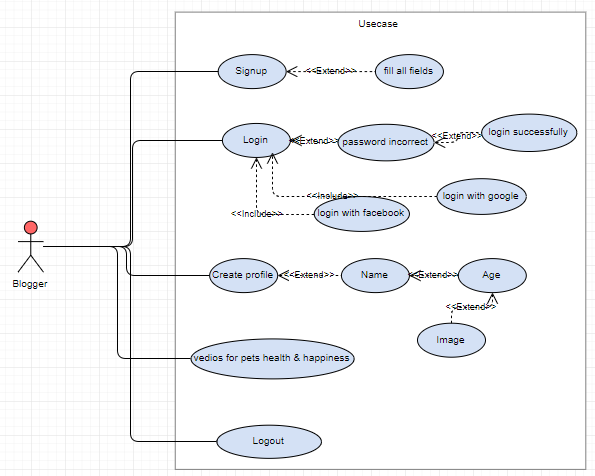
Figure 3 Use Case Model











## Fully Dressed Use Cases

|  |  |
| --- | --- |
| **Use case name**  **Use case id** | Signup  UC-01 |
| **Actors**  **Trigger events** | Vet, Pet’s owner, Botanist and Blogger  Vet, Pet’s owner, Botanist and Blogger will login successfully. |
| **Description** | The use case describes how the Vet, Pet’s owner, Botanist, Blogger Signup with app. |
| **Steps Performed (Main Path):** | 1. System requires Actor to enter required data. 2. Actor enters required data. 3. Actors will Insert/Update the signup information. 4. The use case ends successfully. |
| **Precondition** | Required signup with app or login with face book or google account. |
| **Post condition** | Actor’s login successfully. |

|  |  |
| --- | --- |
| **Use-case name**  **Use case id** | Login  UC-02 |
| **Actors**  **Trigger**  **events** | Vet, Pet’s owner, Botanist, Blogger  Vet, Pet’s owner, Botanist and Blogger will view the home page. |
| **Description** | This use case describes the Vet, Pet’s owner, Botanist, Blogger for not must sign up to the app it will login with google account or face book account. |
| **Steps Performed (Main Path):** | 1. System requires Actor to enter required data. 2. Actor enters required data. 3. Actors will Insert/Update the login data. 4. The use case ends successfully. |
| **Precondition** | Need login with system or login with google account or face book account. |
| **Post condition** | Actors view the home page. |

|  |  |
| --- | --- |
| **Use case name**  **Use case id** | Chat  UC-03 |
| **Actors**  **Trigger**  **events** | Vet, Pet’s owner  Vet and pet’s owner will chat with each other. |
| **Description** | This use case describes the vet and pet’s owner chat with each other to tell and ask about pet’s health. |
| **Steps Performed (Main Path):** | 1. System requires user profile to collaborate with each other. 2. Vet and pet’s owner can tell and ask about pet’s health. 3. The use case ends successfully. |
| **Pre**  **condition** | Need create profile for chat |
| **Post condition** | Vet and pet’s owner can tell and ask about pet’s health |

|  |  |
| --- | --- |
| **Use case name**  **Use case id** | Videos and images upload  UC-04 |
| **Actors**  **Triggering events:** | Vet, Pet’s owner, Botanist and Blogger  The botanist will upload plants name, images of plants injurious for pets and blogger will upload videos, vet and pet’s owner will upload status. |
| **Description** | This use case describes the botanist upload data for pet’s health and safety and plants name, images of plants injurious for pets and blogger upload videos for pet happiness, vet and pet’s owner upload status to share daily activity of pet with each other. |
| **Steps Performed (Main Path):** | 1. System requires profile to upload videos and images. 2. User can select videos and images in his system gallery. 3. Videos and images uploaded successfully. 4. The use case ends successfully. |
| **Precondition** | System need to create profile for upload videos and images. |
| **Post condition** | After create the profile botanist upload plants name, images of plants injurious for pets and blogger upload videos, vet and pet’s owner upload status. |

|  |  |
| --- | --- |
| **Use-case name**  **Use case id** | Make Appointment  UC-05 |
| **Actor** | Pet’s owner |
| **Triggering events** | Pet’s owner will make appointment for house call checkup of pet in vet clinic. |
| **Description** | This use case describes the pet’s owner make appointment for House call, Funerary services and rescue triage disaster. |
| **Steps Performed (Main Path):** | 1. System requires user profile. 2. Pet’s owner can make appointment for house call checkup in vet clinic. 3. The use case ends successfully. |
| **Precondition** | System need login and create profile. |
| **Post condition** | After create the profile Pet’s owner make appointment. |

|  |  |
| --- | --- |
| **Use case name**  **Use case id** | View home page  UC-06 |
| **Actors**  **Triggering events:** | Vet, Pet’s owner, Botanist and Blogger.  After login the Vet, Pet’s owner, Botanist and Blogger will view the home page. |
| **Description** | This use case describes the Vet, Pet’s owner, Botanist and Blogger can view the home page. |
| **Steps Performed (Main Path):** | 1. System requires login with system. 2. Vet, Pet’s owner, Botanist and Blogger can view the home page. 3. The use case ends successfully. |
| **Precondition**  **Post condition** | System required to user login with system.  After login the Vet, Pet’s owner, Botanist and Blogger can view the home page. |

|  |  |
| --- | --- |
| **Use case name**  **Use case id** | Edit Profile  UC-07 |
| **Actors** | Vet, Pet’s owner, Botanist and Blogger |
| **Triggering events** | Vet, Pet’s owner, Botanist and Blogger will edit the profile. |
| **Description** | This use case describes the Vet, Pet’s owner, Botanist and Blogger can edit his own profile. |
| **Steps Performed (Main Path)** | 1. System requires to create profile. 2. User can edit his own profile. 3. The use case ends successfully. |
| **Precondition** | System required to user create profile. |
| **Post condition** | Vet, Pet’s owner, Botanist and Blogger can edit the profile. |

|  |  |
| --- | --- |
| **Use case name**  **Use case id** | Order Medicine  UC-08 |
| **Actor name** | Pet’s owner |
| **Triggering events** | Pet’s owner will order medicines for pet. |
| **Description** | This use case describes the pet’s owner can order medicines. |
| **Steps Performed (Main Path):** | 1. System requires user profile. 2. Pet’s owner will order medicines for pet. 3. The use case ends successfully |
| **Precondition** | System requires user profile and then view all medicines. |
| **Post condition** | Pet’s owner can order medicines for pet. |

|  |  |
| --- | --- |
| **Use case name**  **Use case id** | Edit pet profile  UC-09 |
| **Actors** | Pet’s owner |
| **Triggering events** | Pet’s owner will edit the pet profile. |
| **Description** | This use case describes the Pet’s owner can edit his pet profile. |
| **Steps Performed (Main Path)** | 1. System requires to create pet profile. 2. User can edit his pet profile. 3. The use case ends successfully. |
| **Precondition** | System required to user create pet profile. |
| **Post condition** | Pet’s owner can edit his pet profile. |

|  |  |
| --- | --- |
| **Use case name**  **Use case id** | Edit pet history  UC-10 |
| **Actor** | Vet |
| **Triggering events** | Vet will edit the pet history. |
| **Description** | This use case describes the Vet can edit pet history. |
| **Steps Performed (Main Path)** | 1. System requires to create pet history. 2. User can edit pet history. 3. The use case ends successfully. |
| **Precondition** | System required to vet create pet history. |
| **Post condition** | Vet can edit the pet history. |

# Chapter 4

# System Design

**Chapter 4:** System Design

We are designing the whole system conceptually just to clear out how system will look like. We have made different diagrams, different designs, models ER-Diagram to see how our system will works & how we will manage it.

## Architecture Diagram

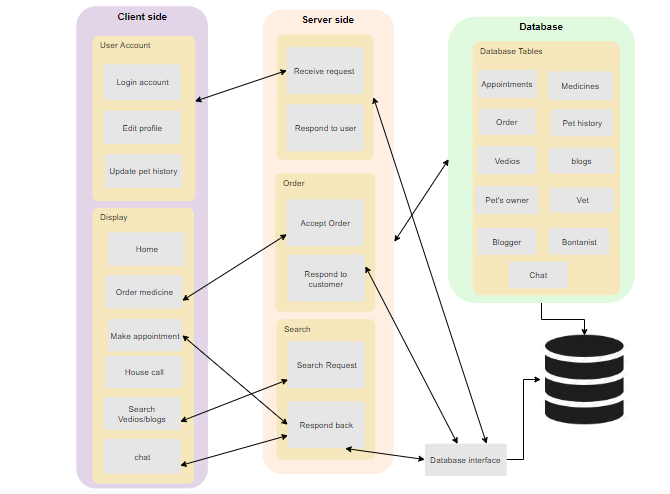


Figure System Architecture

## Domain Model

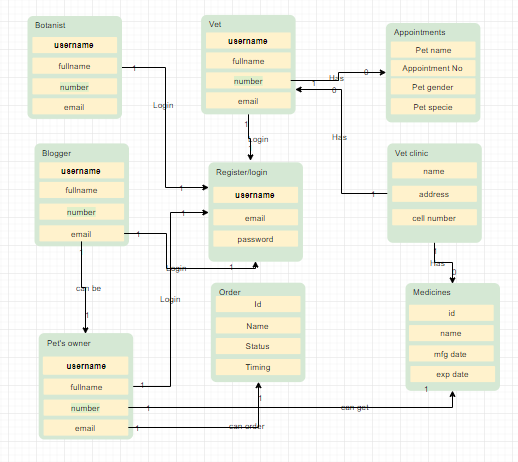


Figure Domain Model

## Entity Relationship Diagram with data dictionary

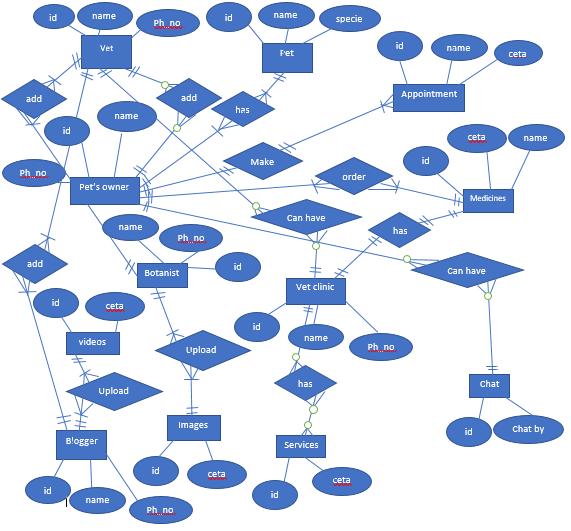


Figure Entity Relationship Diagram

## Class Diagram

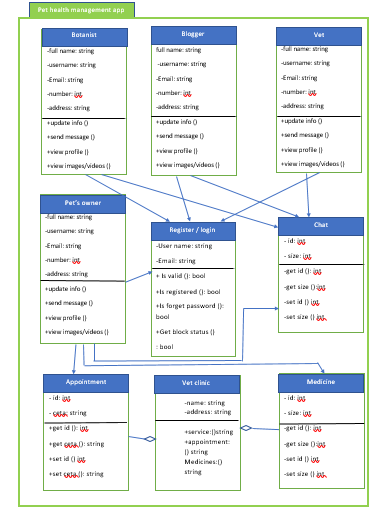


Figure Class Diagram

## Sequence / Collaboration Diagram

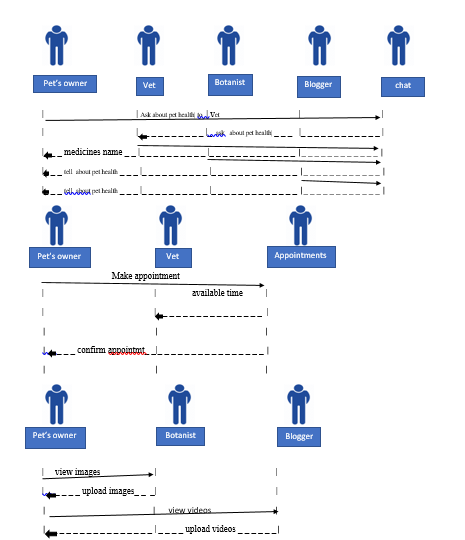


Figure Sequence Diagram

## Activity Diagram

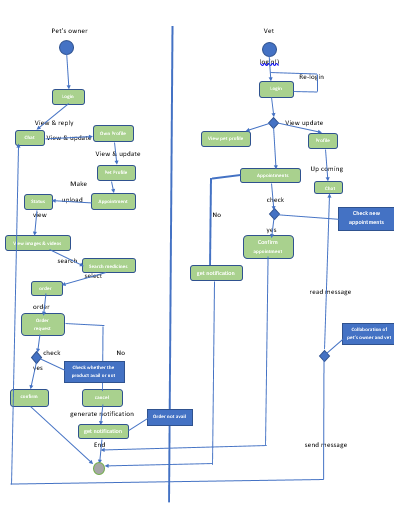


Figure Sequence Diagram

## State Transition Diagram

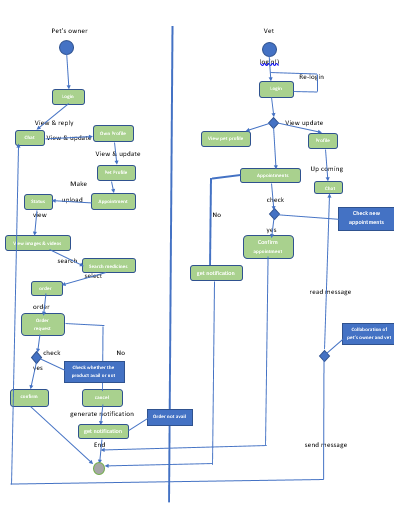


Figure State Transition Diagram

## Component Diagram

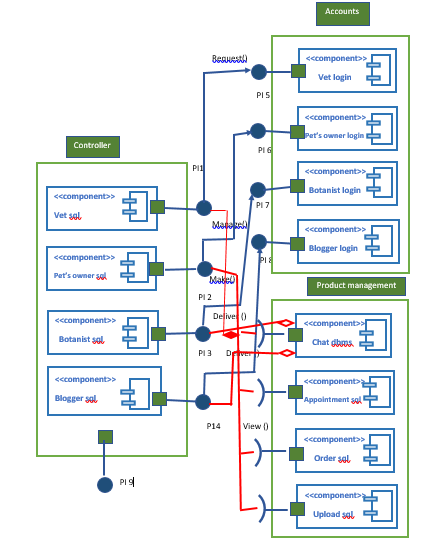


Figure Component Diagram

## Deployment Diagram

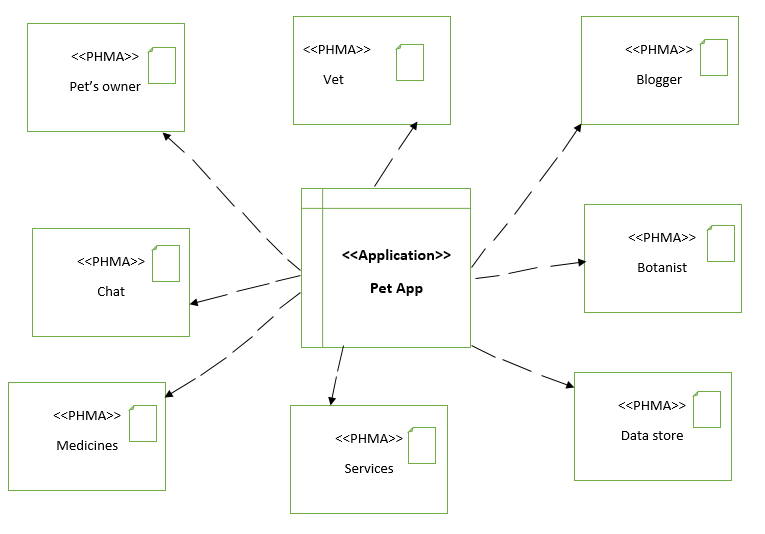


Figure Deployment Diagram

## Data Flow diagram

# 

Figure 13 Data Flow Diagram

# 

# Chapter 5

# Implementation

## Important Flow Control/Pseudo codes

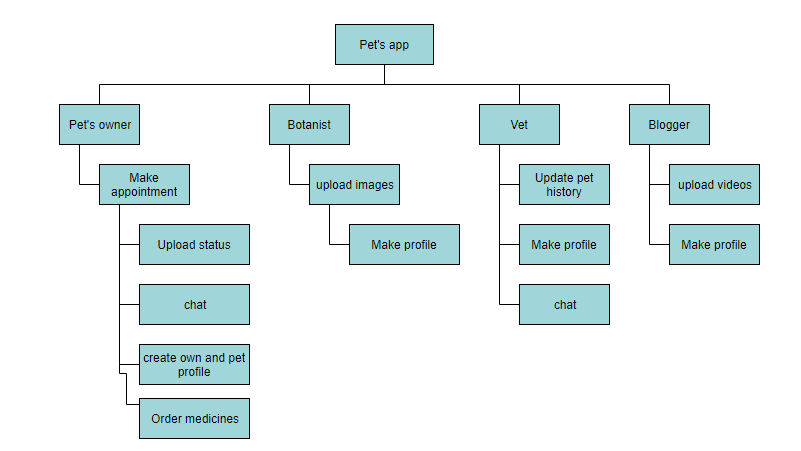


Figure Flow Control

## Components, Libraries, Web Services and stubs

Tool: Android Studio, Java JDK

Languages: java, xml, Php, JSON

Database: firebase for Scripting as Database Language.

Server: Local Serves SQLite

## Deployment Environment

* Hardware: Core 2 duo, 500 GB Hard Drive, 2.8 GHZ, 8 GB RAM
* Operating System: Window 10
* Application: Google Chrome
* Network: Modem, LAN WAN connections, Ethernet cables

## Tools and Techniques

* Android studio for front end coding
* Java JDK for back end coding
* Firebase will also be used for MYSQL database
* Spiral Model will be used for developing.
* PSR-1 Basic Coding Standard.
* Android Studio emulator for test the project

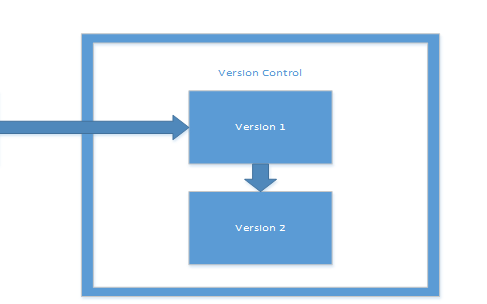
## Best Practices / Coding Standards

**Codding Standards**

* IEEE Oriented Programming coding standards for C#, C++ in parallel to deliver the optimal code.
* Published in:[Computational and Information Sciences (ICCIS), 2013 Fifth International Conference on](https://ieeexplore.ieee.org/xpl/mostRecentIssue.jsp?punumber=6642226)
* Date of Conference:21-23 June 2013
* Date Added to IEEE *Xplore*:24 October 2013

Electronic ISBN: 978-0-7695-5004-6

## Version Control



Pet’s app

Figure Version Control

* **Version 1**

Version 1 of our product will be released after alpha and beta testing. This will help us to identified major defects and prevent major failure. Version 1 will not be all the functionalities.

* **Version 2**

Version 2 of our production will be released after successful feedback from version 1. Version 2 will contain all the functionalities of the product.

# 

# Chapter 6

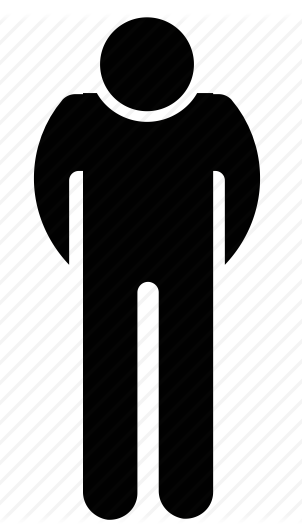
# Testing and Evaluation

**Chapter 6:** Testing and Evaluation

Software testing is very important because of the many reasons; software testing is required to point out the defects and errors that were made during the development phase. It’s essential since it makes sure of the customers reliability and their satisfaction in the website. It is very important to ensure the quality of the product.

## Use Case Testing

**User Actions**



Upload video/images

Delete video/images

Update pet history

Create pet history

Create profile

Update profile

View medicines

Order medicines

Make appointment

Cancel appointment

confirm appointment

Figure Use Case Testing

Table Use Case Testing

|  |  |  |
| --- | --- | --- |
| **Main success Scenario** | **Step** | **Description** |
| Vet, pet’s owner, botanist, blogger  Extension | 1  2  3  2a  2b | Enter Email & Password (to Log in).  Validate password  Allow Account access  **Password not valid**  Displays message and ask for re-try  **Password not valid 4 times** Can’t Login |

## Equivalence partitioning

Equivalence partitioning is a Test Case Design Technique to divide the input data of software into different equivalence data classes. Test cases are designed for equivalence data class.

A use of this method reduces the time necessary for testing software using less and effective test cases, while validating Sign up form we’ve set values:

* A text field permits only numeric characters (e.g. phone number)
* Length must be 11-14 characters long (e.g. phone number length)

In phone number length numbers from 11 to 13 are equivalent and are valid & numbers from 14 or above all are invalid. And numbers from 0-10 are invalid.

Table Equivalence Partition

|  |  |  |
| --- | --- | --- |
| **Test Scenario** | **Test Scenario Description** | **Expected Outcome** |
| 1 | Enter 0 to 11 characters | System should not accept |
| 2 | Enter 12-14 characters | System should accept |
| 3 | Enter 14-20 characters | System should not accept |

## Boundary value analysis

Boundary value analysis is a test case design technique to test boundary value between partitions (both valid boundary partition and invalid boundary partition).

A boundary value is an input or output value on the border of an equivalence partition, includes minimum and maximum values at inside and outside boundaries. While validating Sign up form we’ve set values:

* Email should only accept Gmail.
* Password should be with a valid length of 10-20.

In test case 9, 10, 11 are the minimum values which are Boundary Values. And 19, 20, 21 are maximum values of Boundary Values.

Table Boundary Value Analysis

|  |  |
| --- | --- |
| **Test Scenario Description** | **Expected Outcome** |
| Boundary Value = 9 | System should NOT accept |
| Boundary Value = 10 | System should accept |
| Boundary Value = 11 | System should accept |
| Boundary Value = 19 | System should accept |
| Boundary Value = 20 | System should accept |
| Boundary Value = 21 | System should NOT accept |

## Data flow testing

Data flow testing is a family of test strategies based on selecting paths through the program's control flow in order to explore sequences of events related to the status of variables or data objects.

Data Flow testing is very important that can help us to pinpoint many issues:

* A variable that is declared but never used within the program.
* A variable that is used but never declared.
* A variable that is defined multiple times before it is used.
* Deallocating a variable before it is used.

So, while data flow testing some undeclared variables shouldn’t be the part of our project. It is better to be removed.

## Unit testing

The unit testing was done after the coding phase. The purpose of the unit testing was to locate errors in the current module, independent of the other modules. Some changes in the coding were done during the testing phase. Finally, all the modules were individually tested following bottom to top approach, starting with smallest and lowest modules and then testing one at a time.

**Testing**

* Checked all the validations on each field.
* Wrong inputs in the fields of the forms.
* Database testing.
* Validation of java. Checked error e.g. syntax error.

**Benefits of Unit Testing:**

* Codes are more reusable. In order to make unit testing possible, codes need to be modular. This means that codes are easier to reuse.
* Unit testing increases confidence in changing/ maintaining code.

## Integration testing

**Integration testing** is a level of software testing where individual units are combined and tested as a group. The purpose of this level of testing was to expose faults in the interaction between integrated units. Integration testing*is an*important part of the testing cycle as it makes it easier to find the defect when two or more modules are integrated.

**Integration Test case:**

Verifying the interface link between the login page and the home page i.e. when a user enters the credentials and logs it should be directed to the homepage

## Performance testing

Performance testing is the process of determining the speed or effectiveness of a computer, network, software program or device.

Software Performance testing is type of testing perform to determine the performance of system to major the measure, validate or verify quality attributes of the system like responsiveness, Speed, Scalability, Stability under variety of load conditions. It is very important to interact with the system.

* **Processor Usage -** amount of time processor spends executing non-idle threads.
* **Memory use -** amount of physical memory available to processes on a computer.
* **Disk time -**amount of time disk is busy executing a read or write request.
* **Bandwidth -** shows the bits per second used by a network interface.
* **Private bytes -** number of bytes a process has allocated that can't be shared amongst other processes. These are used to measure memory leaks and usage.
* **Committed memory -** amount of virtual memory used.
* **Memory pages/second -** number of pages written to or read from the disk in order to resolve hard page faults. Hard page faults are when code not from the current working set is called up from elsewhere and retrieved from a disk.
* **Page faults/second -** the overall rate in which fault pages are processed by the processor. This again occurs when a process requires code from outside its working set.
* **CPU interrupts per second -** is the avg. number of hardware interrupts a processor is receiving and processing each second.
* **Network bytes total per second -** rate which bytes are sent and received on the interface including framing characters.
* **Response time -** time from when a user enters a request until the first character of the response is received.
* **Throughput -** rate a computer or network receives requests per second.
* **Amount of connection pooling -** the number of user requests that are met by pooled connections. The more requests met by connections in the pool, the better the performance will be.
* **Maximum active sessions -** the maximum number of sessions that can be active at once.
* **Hit ratios -** This has to do with the number of[SQL](https://www.guru99.com/sql.html)statements that are handled by cached data instead of expensive I/O operations. This is a good place to start for solving bottlenecking issues.
* **Hits per second -** the no. of hits on a web server during each second of a load test.
* **Rollback segment -** the amount of data that can rollback at any point in time.
* **Database locks -** locking of tables and databases needs to be monitored and carefully tuned.
* **Top waits -** are monitored to determine what wait times can be cut down when dealing with the how fast data is retrieved from memory
* **Thread counts -** An applications health can be measured by the no. of threads that are running and currently active.
* **Garbage collection -** It has to do with returning unused memory back to the system. Garbage collection needs to be monitored for efficiency.

**Performance testing tool**

**Web Load**: Web Load is a pioneer and leader in load testing, providing rich capabilities for managing large-scale performance tests in complex enterprise environments.

## Stress Testing

Stress testing is used to test the stability & reliability of the system. This test mainly determines the system on its robustness and error handling under extremely heavy load conditions.

The goal of stress testing is to analyze the behavior of the system after failure. For stress testing to be successful, system should display appropriate error message while it is under extreme conditions.

# 

# Chapter 7

# Summary, Conclusion and Future Enhancements

**Chapter 7:** Summary, Conclusion & Future Enhancements

## Project Summary

We made a project that can be helpful for pet’s owner. In busy schedule no one have a time to go vet clinic and wait for vet to appointment and wait for medicines. So just for the ease of pet owners we’ve made a app where they can order medicines and make appointment by using Log in to portal and place order get notification that their order is ready.

## Achievements and Improvements

* Learnt to work in a team.
* Learnt time management.
* Learnt how to survive in a competitive business environment.
* We will try to improve & take care of our project, we will try to make it more secure, maintain its availability and it will be more responsive at a time.

## 

## Critical Review

We know there is no our competitors and we’ve learnt how to survive in a market, for being stay in a market we must have to take a uniqueness to compete everyone in a business environment.

## Lessons Learnt

Learnt how to make a successful project with absolutely no chances of failure & learnt how to survive in a market.

## Future Enhancements/Recommendations

Expanding its operation in other major cities of Pakistan it is our main goal.

We will make website where our customers will able to buy medicines from our website with full transaction process.

# Reference and Bibliography

**Reference and Bibliography**

<https://www.uml-diagrams.org/examples/online-example.html>

<https://www.slideshare.net/sunilkumar710>

https://belitsoft.com/php-development-services/software-requirements-specification-document-example-international-standard

<https://www.softwaretestingclass.com/what-is-performance-testing/>

[Book]

Programming the World Wide Web, 4th Edition, by Robert Sebesta

http://www.javaengineeringprograms.com

**Index**