**System Requirements**

**P10: Petswala**

**Muhammad Umar Tajjmal 22100291**

**Ali Hassan 22100262**

**Taha Asim 22100264**

**Ahmad Talal 22100174**

|  |  |  |
| --- | --- | --- |
| **Content** | **Totals** | **Obtained** |
| Introduction | 5 | 3 |
| System actors | 10 | 5 |
| Functional Requirements | 40 | 25 |
| Non-functional requirements | 20 | 15 |
| Who did what | 5 | 5 |
| Review checklist | 5 | 5 |
| Overall formatting/template | 5 | 5 |
| Github folder structure | 10 | 0 |
| Late submission penalty | -20 |  |
| **Grand Total** | **100** | **63** |

**Table of Contents**

[1.](#_heading=h.gjdgxs) Introduction 3

[2.](#_heading=h.30j0zll) System Actors 4

[3.](#_heading=h.1fob9te) Functional Requirements 5

[4.](#_heading=h.3znysh7) Non-functional Requirements / Quality Attributes 6

[5.](#_heading=h.2et92p0) Who Did What? 7

[6.](#_heading=h.tyjcwt) Review checklist 7

# Introduction

<Give an overview of the project here. The overview must highlight the overall objectives of the project and its potential users.>

This project will be an online platform for people having pets. People will be able to find and connect with veterinary doctors, as well as find and offer grooming related services, accessories, and helping articles related to their pets. The potential users for this app will be those owning pets, specifically those seeking veterinary assistance and services for their pets, or simply someone looking for some pet advice. Since there are not many existing apps that are similar to this app, there will be plenty of room to gain a significant user base.

[The introduction does not clarify overall goal/idea of your project.]

# System Actors

<List down the actor names and give a 2-3 lines description of the role of each actor>

[In actor description, you should also explain what each actor really wants from the system. For instance, who is a buyer or who is a service provider. Merely listing the access rights does not clarify the role of an actor.]

|  |  |
| --- | --- |
| **Actor Name** | **Description** |
| Admin | Sign in.  Dashboard.  Add veterinary doctors.  Edit/Delete veterinary profiles.  Approve/Decline service providers requests after filtering spam.  View complaints/feedback about service providers and doctors.  Send notifications and posts of pets to the rescue team’s dashboard.  View log of completed orders of pets and accessories.  View log of users of the app.  Log out of the app. |
| Buyer | Sign up.  Sign in.  User profile.  Edit user profile.  Search for a specific seller of pets and accessories.  Chat with the seller after he/she has accepted the order.  Cancel the order.  Filter the list of veterinary doctors using location and their services.  View veterinary doctors timings,reviews and ratings.  Give feedback,reviews and ratings to veterinary profiles.  Search for a specific service provider.  Place order to a specific service provider.  Give feedback,reviews and ratings to service providers.  Create blogs for pets needs,grooming etc.  Send notifications and posts of pets to the rescue team’s dashboard.  Post pet related events.  Log out of the app. |
| Pet Seller | Sign up.  Sign in.  User profile.  Edit user profile.  View orders from the buyers.  Accept , deny orders.  Chat with the users of the orders that have been accepted.  Access to buyer’s location.  Review and verify the payment received after completion of the order.  Filter the list of veterinary doctors using location and their services.  View veterinary doctors timings,reviews and ratings.  Give feedback,reviews and ratings to veterinary profiles.  Search for a specific service provider.  Place order to a specific service provider.  Give feedback,reviews and ratings to service providers.  Create blogs for pets needs,grooming etc.  Send notifications and posts of pets to the rescue team’s dashboard.  Post pet related events.  Log out of the app. |
| Service Provider | Sign up (will be registered as a service provider after being approved by the admin).  Sign in.  See profile.  Edit profile.  Add/Edit pets services.  View ordered services.  Accept , deny services.  Chat with users seeking grooming services.  Access to client’s location.  View the list of completed services.  Review and verify the payment received after completion of the provided service.  Create service related blogs.  Send notifications and posts of pets to the rescue team’s dashboard.  Post pet related events.  Log out of the app. |
| Rescue team | Sign up.  Sign in.  DashBoard.  View reports and posts of pets to rescue.  Log out of the app. |
| Veterinary Doctor | Register as a veterinary doctor.  sign in.  Dashboard.  View and contact the customers.  Log out of the app. |

# Functional Requirements

<Write system requirements from users’ (actors) perspective. Actor names have been highlighted in the sample requirements below. You may group requirements according to actors or modules>

[You need to review functional requirements completely. They are not clearly articulated. I would suggest that you review both the sections System Actors and Functional Requirements and then list down requirements in this section. For instance, for the information that *who writes blogs or gives feedback,* I need to look in the System Actors section as well.]

|  |  |
| --- | --- |
| **Requirements** | |
| **Sr#** | **Requirement** |
| 1 | Admins will be able to Add/Edit/Delete veterinary profiles. |
| 2 | There will be profile authentication for normal users |
| 3 | Buyers will be able to find nearest veterinary doctors based on their location and services being provided by them, as well as their timings and some reviews and ratings. |
| 4 | All users can view veterinary related feedbacks, ratings and reviews |
| 5 | Pet sellers and service providers can add/edit pets or accessories to sell. |
| 6 | Users can search for pets or accessories to purchase.  [Which of the users can do this?] |
| 7 | Setup flow through which the buyer can contact the seller.  [Which actor is doing this? What does it mean to setup flow?] |
| 8 | Setup flow to register as a service provider.  [Which actor is doing this?] |
| 9 | Service providers will be able to add/edit your services e.g. pets grooming, boarding, daycare, dog walking, etc. as a service provider. |
| 10 | Admin can verify and approve service provider changes in order to filter spam services. |
| 11 | All users can view for services related feedbacks, ratings and reviews |
| 12 | All users can view Blogs section where registered users create posts for articles and videos. |
| 13 | Setup platform for rescue services where rescue teams can register themselves. |
| 14 | Setup flow where anyone can report pets to rescue. |
| 15 | Send notifications / create posts which will be shown on the rescue teams dashboard. [Actors?] |
| 16 | Pets related events calendar where users can publish pets related events in their surroundings. [Actors?] |

# Non-functional Requirements / Quality Attributes

<Requirements must be testable>

Since the system is a client-server-based web application, the performance will depend on the client's hardware and a suitable server environment. Assuming these requirements are fulfilled the system should behave in the following manner.

|  |  |
| --- | --- |
| **Non-functional Requirements** | |
| **Sr#** | **Requirements** |
| 1 | Any get requests for the page should not take more than 10 seconds to load. |
| 2 | Upon loading of the page, the end user should be prompted to give access to their location within 5 seconds. The end user needs to have functional GPS in their device for the system to be able to filter the veterinary profiles around their location. |
| 3 | The system should be able to cater to a high number of simultaneous users (Up to 10000 users). Any more users would depend on the performance of the server the software is deployed on. |
| 4 | The Service Provider/Seller dashboard should be updated with information of any new orders placed within 15 seconds of order placement. |
| 5 | The data returned by any search filter used by the end user should be available within 3 seconds and should be visible in the end user view within 15 seconds. |
| 6 | After placing the reset password request, the user should receive a reset code on their primary contact number/email within 2 minutes. |
| 7 | As per standard memory consumption, our application will consume (130 - 450) MB of memory. |
| 8 | The system uses a password system for authenticating the users. A user account can only be accessed if a person knows the user id and password. The safety of the secret is also dependent on the end user. Constraints will be placed when the user chooses a password to ensure that the password is strong and secure against brute force attacks. |
| 9 | The system has different types of users interacting with the data. Constraints will  be placed to ensure that only authorized users can read or modify certain data. |
| 10 | The form data sent via post request to the server, password stored in the database, and users’ location & address will be encrypted. This step will ensure that the data will be unusable for any unauthorized party. |
| 11 | The system will be robust against web security attacks such as SQL injection and  CSRF attacks. All the input fields from user will be filtered to check for any alicious  strings. |
| 12 | A backup of current version of database will be maintained to cater any data  loss and to insure availability in case of any system crash or hard drive failure.  [What will be the schedule of backups?] |
| 13 | Usability is the most critical aspect for this software. The website is designed to be used by an average citizen and has a minimalistic & intuitive user interface. It should be easy to navigate and understand the use of different elements for an average internet user. More than 80 percent of the average computer user should be able to complete their function without any confusion.  [Quantify this requirement. For instance, in how much time or in how many steps a user would be able to complete his/her tasks with or without training.] |
| 14 | This software is highly portable and can be deployed through any platform or operating system that can support versions of JavaScript libraries. On the other hand, the user of the software only requires a modern browser that is supported on all major operating systems. [Are you planning to support all the browsers? Will you test on browsers on desktops and mobile phones? Which versions of JavaScript libraries would your system need?] |
| 15 | The reusability of the code is important to maintain the website and upgrade some features of the software that client intends. To make the code highly reusable, a design pattern such as MVC will be followed. Also, good modularity in templates and high cohesion of elements will be ensured. |
| 16 | The software will cater all the cases for all functions to safeguard against any system faults and ensure high availability. Also, the design will incorporate the possibility that a fault can occur, but it does not affect the functionality for other users. On average, the system should not be unavailable for more than a few hours in a month. [This is too broad requirement. Quantify it.] |

s

# Who Did What?

|  |  |
| --- | --- |
| **Name of the Team Member** | **Tasks done** |
| Umer | System actors, Non Functional Requirements |
| Ali | Introduction, Functional Requirements |
| Taha | Introduction, Functional Requirements |
| Ahmad Talal | System actors, Non Functional Requirements |

# Review checklist

Before submission of this deliverable, the team must perform an internal review. Each team member will review one or more sections of the deliverable.

|  |  |
| --- | --- |
| **Section** **Title** | **Reviewer Name(s)** |
| Introduction | Ahmad Talal |
| Actors | Taha Asim |
| Functional Requirements | Muhammad Umar Tajjmal |
| Non-functional requirements | Ali Hassan |