
Software Requirements Specification

for

“Pick Me! Pet Store”

Version 1.0

Prepared by Tolgoi Gabriel

March 19, 2017

Table of Contents

1. Introduction.....	1
1.1 Purpose	1
1.2 Product Scope	1
1.3 References	1
2. Overall Description.....	2
2.1 Product Perspective	2
2.2 Product Function	3
2.3 User Classes and Characteristics	3
2.4 Operating Environment	3
2.5 Design and Implementation Constraints.....	4
2.6 User Documentation	4
2.7 Assumptions and Dependencies	4
3. External Interface Requirements	5
3.1 User Interfaces.....	5
3.2 Hardware Interfaces.....	7
3.3 Software Interfaces	7
3.4 Communications Interfaces	7
4. System Features	7
4.1 Account Registration	7
4.2 Account Login	8
4.3 Account Logout	8
4.4 Admin Add Member.....	8
4.5 Admin Edit Member.....	8
4.6 Admin Modify Resource	8
4.7 Visitor/Registered client product search.....	9
4.8 Client “Add to shopping cart”	9
4.9 Client “Delete from Shopping Cart”	9
4.10 Client checkout.....	9
4.11 Client adopt/buy a pet.....	9
5. Other Nonfunctional Requirements.....	10
5.1 Performance Requirements.....	10
5.2 Safety Requirements.....	10
5.3 Security Requirements.....	10
5.4 Software Quality Attributes	10
5.5 Business Rules.....	11
6. Other Requirements	11

1. Introduction

1.1 Purpose

The purpose of this document is to define the system requirements for the “Pick Me! Pet Store” online pet store. It will explain the purpose and features of the store, its interfaces, what it will do, the constraints under which it must operate and how it will interact with the users.

1.2 Product Scope

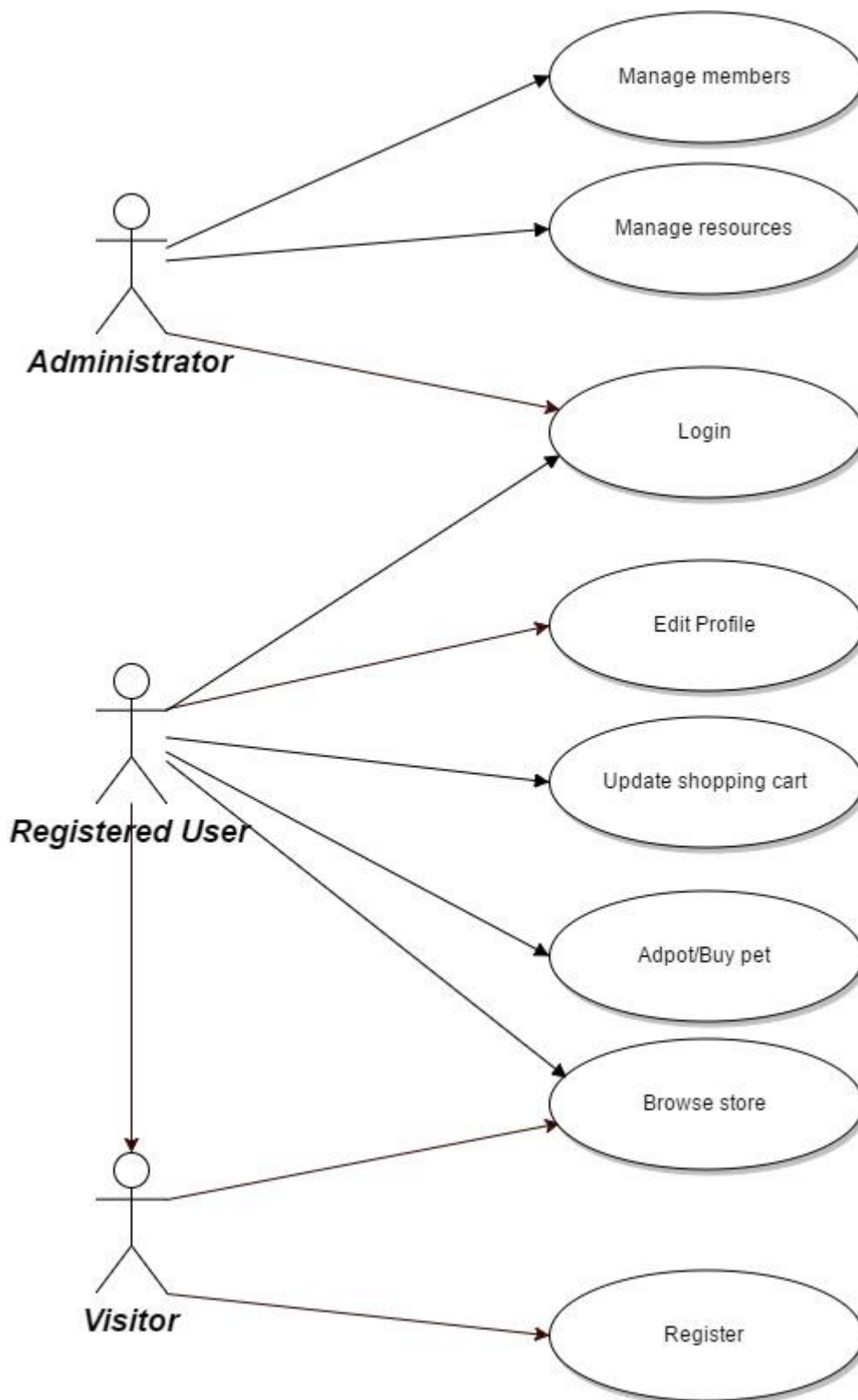
The objective of this project is to create and implement an online pet store website. The website will be used by anyone looking for a pet or just looking for pet-related supplies. The website will allow users to create and maintain an individual account, search the store database for pet food, accessories and animal medical supplies. The website’s administrators will be able to add/remove products, modify prices and remove users. The site will also offer the possibility of adopting or buying real pets. The website will be connected to a relational database containing a list of products related data, and user data.

1.3 References

IEEE. *IEEE Std 830-IEEE Recommended Practice for Software Requirements*

2. Overall Description

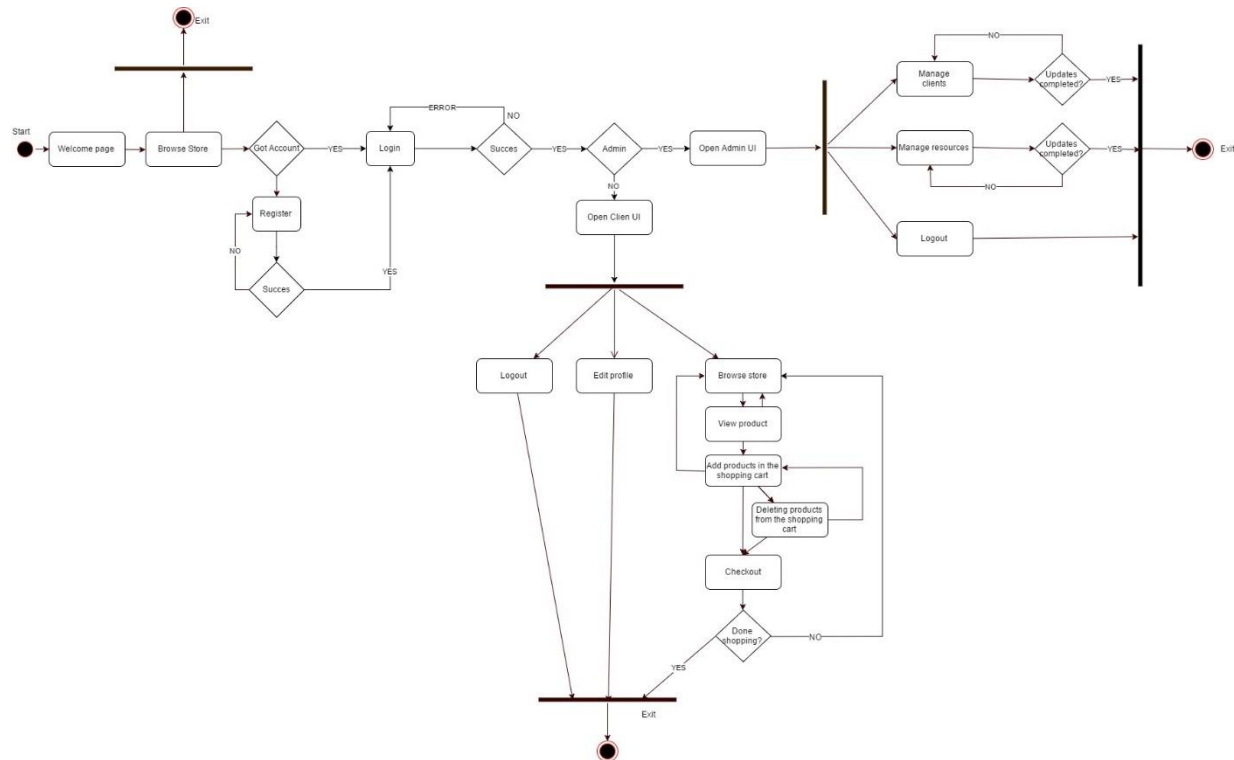
2.1 Product Perspective



The "Pick Me! Pet Store" is not a component of a larger system.

2.2 Product Function (activity diagram)

- Login
- New user registration
- New admin registration
- Admin manages user accounts
- Browsing the store & adding products to the shopping cart
- Managing payment
- Buying or adopting a real pet



2.3 User Classes and Characteristics

The application users will be store's employees which will have administrator privileges and citizens which will have regular member accounts. Users of the website must know how to navigate in a website.

2.4 Operating Environment

The "Pick Me! Pet Store" will require users to have access to a web browser on their workstation computer like Google Chrome (Version 56.0.2924 or later) or Mozilla Firefox (Firefox 50 or later).

2.5 Design and Implementation Constraints

Creating a user interface which is both effective and easily navigable will pose a difficult challenge. Other constraints such as a weak web-host and slow internet browsers (“Internet Explorer”) are also worth considering. “Pick Me! Pet Store” is meant to be quick and responsive, so each feature must be designed and implemented with efficiency in mind.

Backend language: Java

Frontend: HTML, CSS, JavaScript

Framework: Bootstrap

2.6 User Documentation

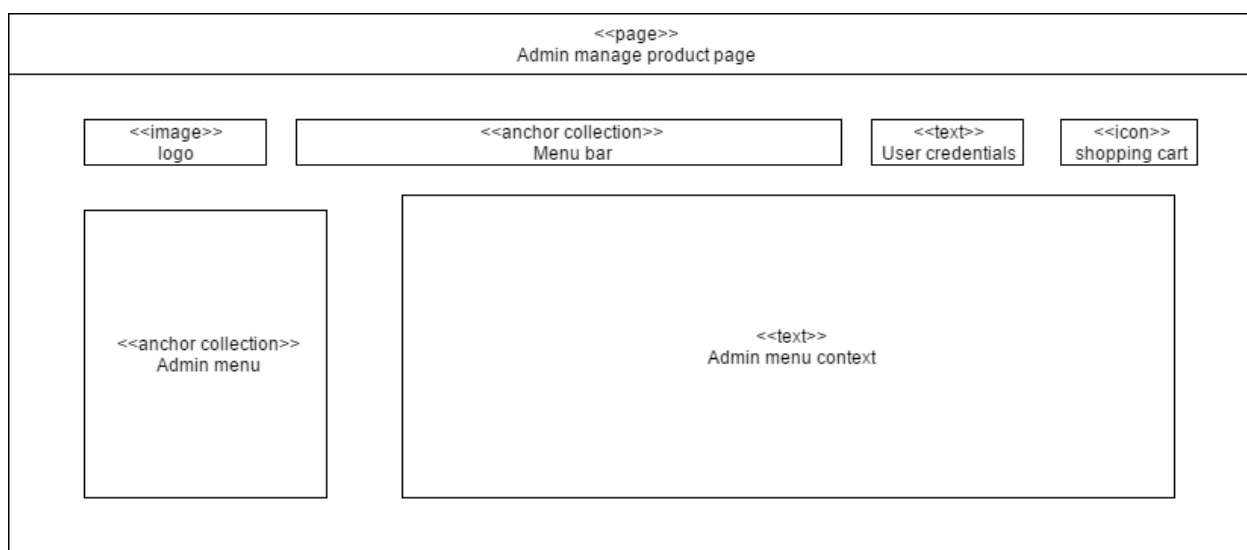
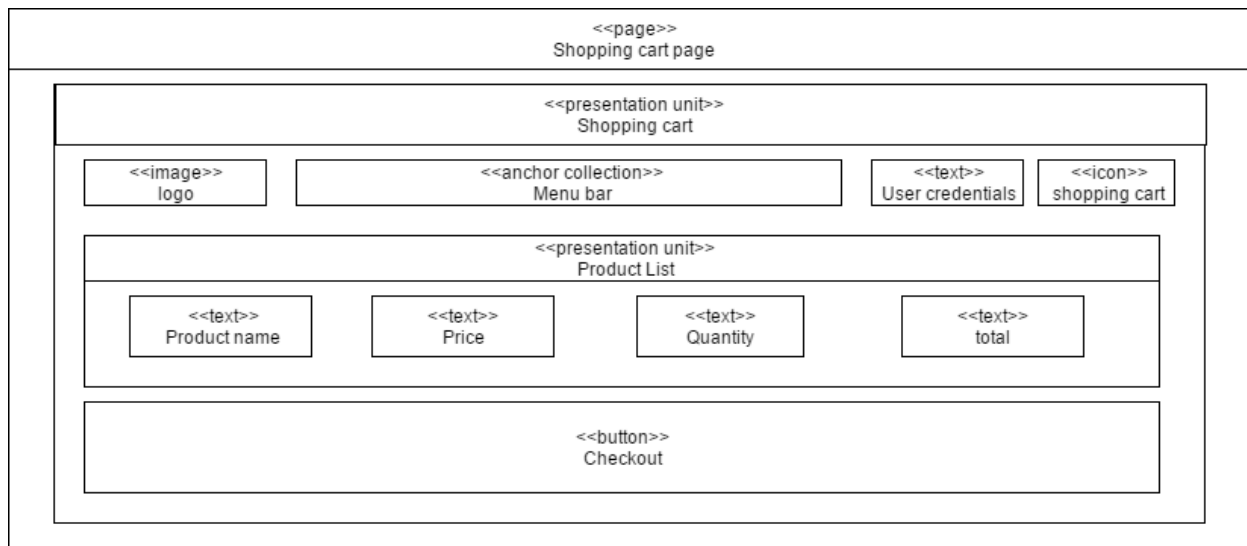
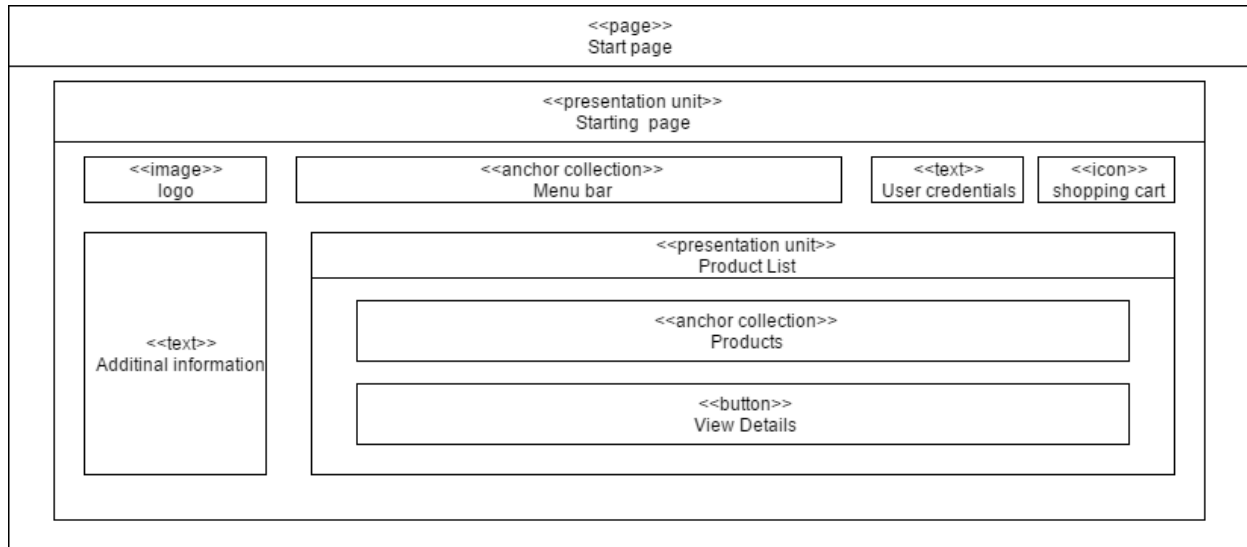
NA

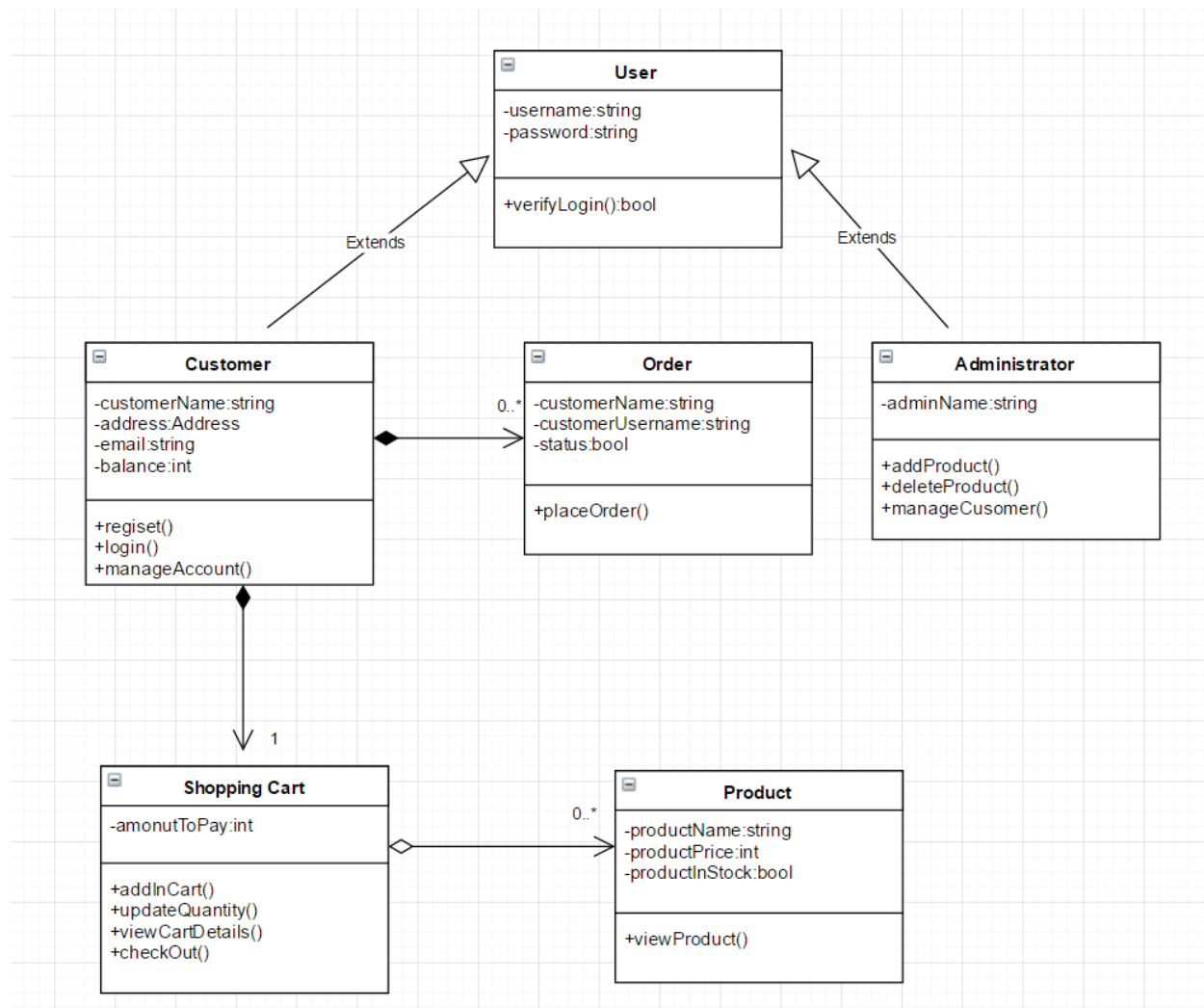
2.7 Assumptions and Dependencies

The web application assumes that the user has a computer with an Internet connection and a web browser to access “Pick Me! Pet Store”. The system may not behave correctly when used with internet browsers other than Firefox (Firefox 50 or later) and Google Chrome (Version 56.0.2924 or later).

3. External Interface Requirements

3.1 User Interfaces





The layout should be as follows:

- The header which displays:
 - the logo
 - the currently logged user
 - the “Logout” and “Manage Account” menu
- The body, which displays
 - the menu containing product categories
 - the content of the selected category (if no prior category was selected the content will be just the welcome page)
- The footer, which displays
 - contact
 - copyright

There will be three different user interfaces that will accompany this website: one for the visitors of the site, one for the registered clients and one for the administrators.

- Simple visitors will be able to only browse the products and view the pets currently available for sale/adoption.
- The administrators will be able to add or remove members, add new „resources” such as products, add new pets for sale/adoption, and also change the products prices.
- The registered clients will have all the features that simple visitors have. Additionally they will have the possibility to add items in their shopping carts, and perform the purchase. Also, registered users can adopt or buy pets.

3.2 Hardware Interfaces

NA

3.3 Software Interfaces

The application is connected to a database (MySQL). The database will communicate with the server, and the server will send the information to the client application. The information will consist of data regarding the quantity and price of products available for sale in the inventory.

3.4 Communications Interfaces

HTTP will be the communication protocol that “Pick Me! Pet Store” will use.

4. System Features

4.1 Account Registration

4.1.1 Functional Requirements

- The system shall allow a non-registered user to create a secure account.
- The system shall require the following information from the user:
 1. First name (max 20 chars)
 2. Last name (max 20 chars)
 3. Username (max 20 chars)
 4. Password (must contain uppercase letter, a number and a special character)
 5. Email (has to be a valid mail, <user>@<domain>)
- The system shall confirm the username, email and password are acceptable.
- The system shall store the information in the database.

All fields are mandatory and are to be written in the text boxes offered by the registration interface. If the request passes, the user will receive a validation email. He must confirm it

otherwise the account will be deleted in 1 day. If the request doesn't pass the user will be informed with the error and can try again.

4.2 Account Login

4.2.1 Functional Requirements

- The system shall allow a registered user to log-in to their account.
- The system shall require a username and password from the user.
- The system will verify the username and password, and the user will be considered "logged-in".

All fields are to be written in the text boxes offered by the login interface.

4.3 Account Logout

4.3.1 Functional Requirements

- The system shall allow the registered and logged-in user to exit his/her account, so that access to operations requiring a user to be logged in are now disabled

4.4 Admin Add Member

4.4.1 Functional Requirements

- The system shall allow a logged-in admin to create new member accounts and new admin accounts.
- Only admins can add new admins.
- The system shall require a username from the logged-in admin

4.5 Admin Edit Member

4.5.1 Functional Requirements

- The system shall allow a logged-in admin to edit a registered member
- The system shall require a username from the logged-in admin
- The system shall display all the information regarding that user with the possibility of modifying data such as user, password, privileges.

4.6 Admin Modify Resource

4.6.1 Functional Requirements

- The system shall allow a logged-in admin to modify an available resource
- The system shall require the name of the resource from the logged-in admin
- The system shall display all the information regarding that resource with the possibility of modifying its description and stock availability.

4.7 Visitor/Registered client product search

4.7.1 Functional Requirements

- The system shall allow a logged-in client or a simple visitor to search for products by name.
- The system shall require the name of the resource.
- The search results will include a picture of the product along with its description and price.

4.8 Client “Add to shopping cart”

4.8.1 Functional Requirements

- The system shall allow a registered and logged-in user to temporarily save products that are being considered for purchase into a list associated with their account.
- When viewing the shopping cart list, the system shall display the total price of the products in the cart.

4.9 Client “Delete from Shopping Cart”

4.9.1 Functional Requirements

- The system shall allow a registered and logged-in user to remove any unwanted products from their shopping cart.
- The system shall provide the user a way to select one of the products in his/her cart for deletion
- After the user has indicated the particular product to be deleted from their shopping cart, the stored list representing the shopping cart should be updated, and the display should be updated to show only the products remaining in the cart.

4.10 Client checkout

4.10.1 Functional Requirements

- The system shall allow a registered and logged-in user to purchase products that are in their shopping cart
- The user will receive a confirmation number
- The products (names, prices and quantities) will be displayed to the screen.

4.11 Client adopt/buy a pet

4.11.1 Functional Requirements

- The system shall allow a registered and logged-in user to adopt or purchase a real pet
- The system shall offer a list of “up for adoption” pets for the client to choose from

- Adopting a pet is free, the client will be given specific information about his next-to-be pet
- Selecting an “up for adoption” pet from the list will open a new page containing pictures, a video of the pet and other details such as its age and personality. If the client wants to adopt it he will have to add it in the shopping cart and proceed to checkout (total sum will be 0).
- When a client wants to buy a pet a list of pet breeds that are for sale will be displayed. Selecting a breed will just show generic images, not the actual pet(s) that will be delivered. As other products, the selected pet(s) will have to be added to the shopping cart. All the pets that are for sale come from professional breeders.

5. Other Nonfunctional Requirements

5.1 Performance Requirements

The product shall take initial load time depending on the internet connection strength which also depends on the media-device from which the product is run. The performance shall depend upon hardware components of the client/customer.

5.2 Safety Requirements

Information transmission should be securely transmitted to the server without any changes in information.

5.3 Security Requirements

The system shall use secure sockets in all transactions that include any confidential customer information. The system shall confirm all transactions with the customer's web browser.

5.4 Software Quality Attributes

Availability: the system is available 24 hours a day.

Portability: a user can log in to the system at any time.

Reliability: the system can be used by multiple users concurrently. Any user can access the system using even a medium performance PC.

Rule 1(required) Source code shall only use /* ... */ style comments.

Rule 2(advisory) No identifier name should be reused.

Rule 3(required) All automatic variables shall have been assigned a value before being used

Rule 4(advisory) The increment (++) and decrement (--) operators should not be mixed with

other operators in an expression

Rule 5(advisory) Write only one declaration per line

Rule 6(advisory) Use meaningful names for variables

Rule 7(required) All if ... else if constructs shall be terminated with an else clause.

Rule 8(required) All constants must be declared with capital letters and underline between words

Rule 9(required) The operands of a logical && or || shall be primary-expressions.

Rule 10(required) There shall be no unreachable code.

5.5 Business Rules

- Grievances of all members will be addressed only by prior notice the administrator.

6. Other Requirements

NA

Appendix A: Glossary

NA