**SWE 632 Spring 2023 Project Checkpoint #2**

**Website URL**: <http://mason.gmu.edu/~mveludut/home.html>

**Project Argument**

**Why this website?**

Pets are an integral part of many people's lives, and they deserve to be treated with the best care possible. With an increasing number of pet owners worldwide, there is a growing need for a convenient and efficient platform that provides all the necessary pet care products and services in one place. Therefore, creating an e-commerce platform dedicated to pets (at present for dogs and cats, but will cater to other animals as well as and when deemed appropriate by the market) needs and accessories is essential to cater to this growing demand.

**What we are solving:**

The problem we are solving is the lack of a comprehensive e-commerce platform dedicated to pet care products and services. Many pet owners struggle to find high-quality and reliable products and services for their fur friends, which can be time-consuming and frustrating. Our platform, "Paws & Tails," aims to solve this problem by providing a one-stop-shop for all pet care needs, including pet food, toys, grooming products, and veterinary consultation services. Presently we are only focusing on Dogs and cat. In future we will expands to all Pet animals

**Target domain/ who needs it?**

Pet owners, animal lovers, and anyone looking for high-quality pet care products and services will benefit from our platform. The platform is designed to cater to the needs of pet owners with different types of pets. Presently exclusive for dogs and cats, making it accessible for these pet owners. In future we will expand to all kinds of pets.

**Our plan How to tackle this problem & objective:**

We plan to solve this problem by developing a high-quality user interface (UI) for our e-commerce platform that is user-friendly, visually appealing, and functional. Our platform will feature various hyperlinks, buttons, pop cards, and other components that provide affordance to the user, making it easy to navigate and find the necessary products and services. We have also incorporated signifiers in the form of buttons that change colors, navbar menus that change background colors, and cards that scale up on hovering, making the platform more engaging and interactive for the users. Additional features include autocomplete feature in FAQS which enable user to search their queries quickly. We will continue to improve and iterate on the platform to ensure that it remains user-friendly and meets the evolving needs of our users like minimal interaction and yet to be look beautiful.

**Objective**

Our objective for this project is to create a user interface (UI) for an online e-commerce platform dedicated to pet needs and accessories. The name of the website is "Paws & Tails". The main goal is to have a one stop platform which will cater to almost every need a pet and a pet parent has, including but not limited to Pet food, Pet toys, Grooming products, and even veterinary consultation services.

Users will be able to find a wide variety of pet products and services on our website, making it a one-stop shop for all their pet care needs. We plan to develop a high-quality website that will be user-friendly, visually appealing, and functional, providing users with a satisfying UI experience.

We aspire to make this website as user friendly as possible. Minimizing the Gulfs of Execution to the bare minimum is one of our primary goals. We plan to achieve this by keeping our page simple and providing an easy and smooth flow between components. No big knowledge of anything would be required to use the website.

Using bootstrap, we have tried to make the overall look and feel of the website visually appealing and pleasant to the users. We have provided various hyperlinks, buttons, pop cards and other stuff to provide affordance to the user while using them. We have provided signifiers in the form of buttons that change the colors, navbar menus that change the background colors and cards that scale up on hovering. We intend to make them more significant by bringing other effects and ideas in the further iterations of the project.

**Main Components/Pages/Features of our website:**

1. Interactive and Easy to navigate Landing page.
2. Exclusive Page to showcase offers and deals. (Part of next iteration)
3. Account login modal.
4. Account Registration page.
5. Password Recovery ( Part of next iteration)
6. Cart view of all products and checkout.
7. Tele Consultation for your pets from the comfort of your home.
8. Responsive Navbar containing various categories and options.
9. Footer containing links to various components and pages of the website.
10. Landing page showing top categories, top brands and a small glimpse of the offers and deals available.
11. Contact us page contains information useful to contact the owners.
12. About us page telling users about who we are, our story and what we aspire to do.

**Technologies being used:** HTML, CSS, Bootstrap, JavaScript, jQuery, JSON

**Testing Technologies :** Selenium, Jest

**Information flow diagrams:**

1. Login , Register and Forgot Password information flow diagram between the components.

Diagram

Description automatically generated

1. Pet health and Care Component information flow.

Diagram

Description automatically generated

1. Deals and Product shopping information flow.

Diagram

Description automatically generated

1. Cart and Payment.

Diagram

Description automatically generated

1. Order Tracking.

Diagram

Description automatically generated

**Architectural Diagram:**

MVC architecture diagram for our website is as below,

Diagram

Description automatically generated

**Sample Screenshots of the website:**

**Graphical user interface, application, website

Description automatically generatedAbout Us Page:**

**Text

Description automatically generatedContact info page:**

**Landing page:**

Graphical user interface, website

Description automatically generated

**FAQ Page**

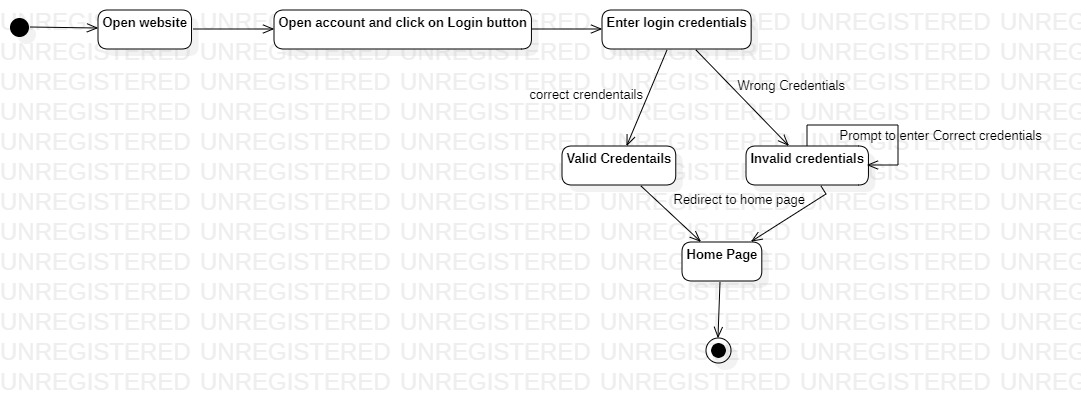
****

**Use Cases:**

1. **User login:** Implementation of user login use case has been performed using jquery. The main and the most important part of this use case is the json file names “data.json” which contains all the user data.

Using this use case any registered user can login to the application using their credentials. By logging in they gain access to some exclusive offers and deals. Also one has to login before they can view their cart and checkout and place order. It also gives them access to book a tele consultation session.

To implement this module, first the data is loaded from the Json file and then it is compared to the credentials entered by the user in the login form. Alerts are displayed for any incorrect credentials entered by the user. Upon entering the correct credentials and verification being performed, login success message is displayed onto the screen.



**Fig : User login state diagram**

Graphical user interface, application

Description automatically generated

Fig: Login Page

Text

Description automatically generated

**Fig: Implementation of login validation using Javascript**

Text

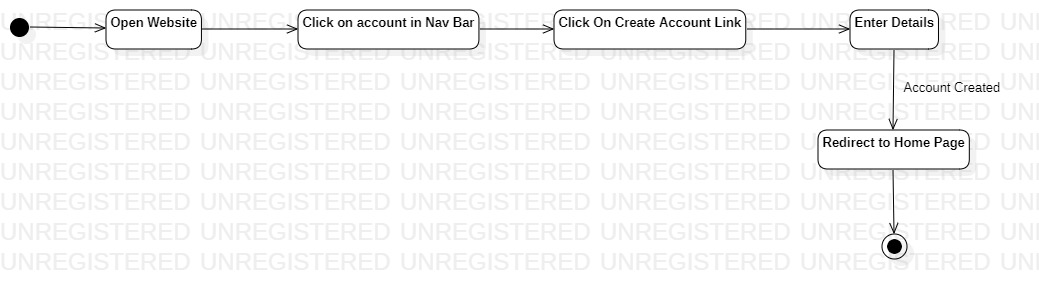
Description automatically generated

**Fig: Json data file for user login credentials**

1. **User registration:** The user registration use case of this application has been implemented using HTML, CSS, Bootsrap, JavaScript and jQuery.

Here in this use case any new user can register themselves onto the website by filling the user registration form. Upon successfully entering all the details one can consider themselves members or users of this application.

This use case also has custom form validations. The actual aim of this use case was to generate a Json file which would store the details of all the users. But this was practically not possible to implement owing to browser and http security issues. Instead we have mocked this step and are displaying the user with successful registration message if the user has entered all the details in a correct fashion. We are actively working on this and are looking to fix this in the future.



**Fig: User registration state diagram**

Graphical user interface

Description automatically generated

Fig: User Registration Page

1. **Shopping cart & checkout:** We have implemented the cart and checkout use case. We used JS to calculate the total by taking input from the quantity fields. First the values are taken from the page and the totals are computed and displayed on to the screen. The values are computed every time there is a change to the items in the cart.

After finalizing the cart the user can checkout using the button present at the bottom. Signifiers like hovering effect have been provided to indicate the presence of the button on the screen. Also the gulfs of execution have been reduced to a vast extent and the user need no knowledge of anything to use this feature.

Once the user clicks on the checkout button, the user is prompted through a modal to enter shipping information, also included in the same modal are payment details. Upon successful entering the details the system process the payment information in the backend( mocked this step) and displays a order placed message onto the screen.

Text

Description automatically generated

**Fig: JavaScript functions to calculate the total amount of cart items**

**Diagram

Description automatically generated**

**Fig: Shopping cart & checkout state diagram**

A screenshot of a computer

Description automatically generated with medium confidence

**Fig: Shopping cart with mocked data**

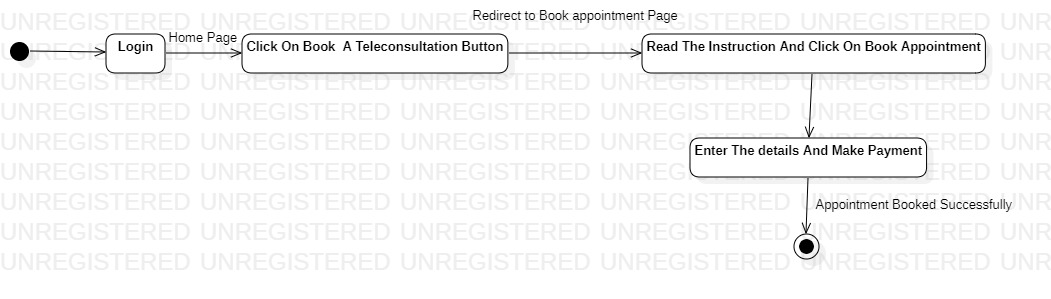
A screenshot of a computer

Description automatically generated with medium confidence

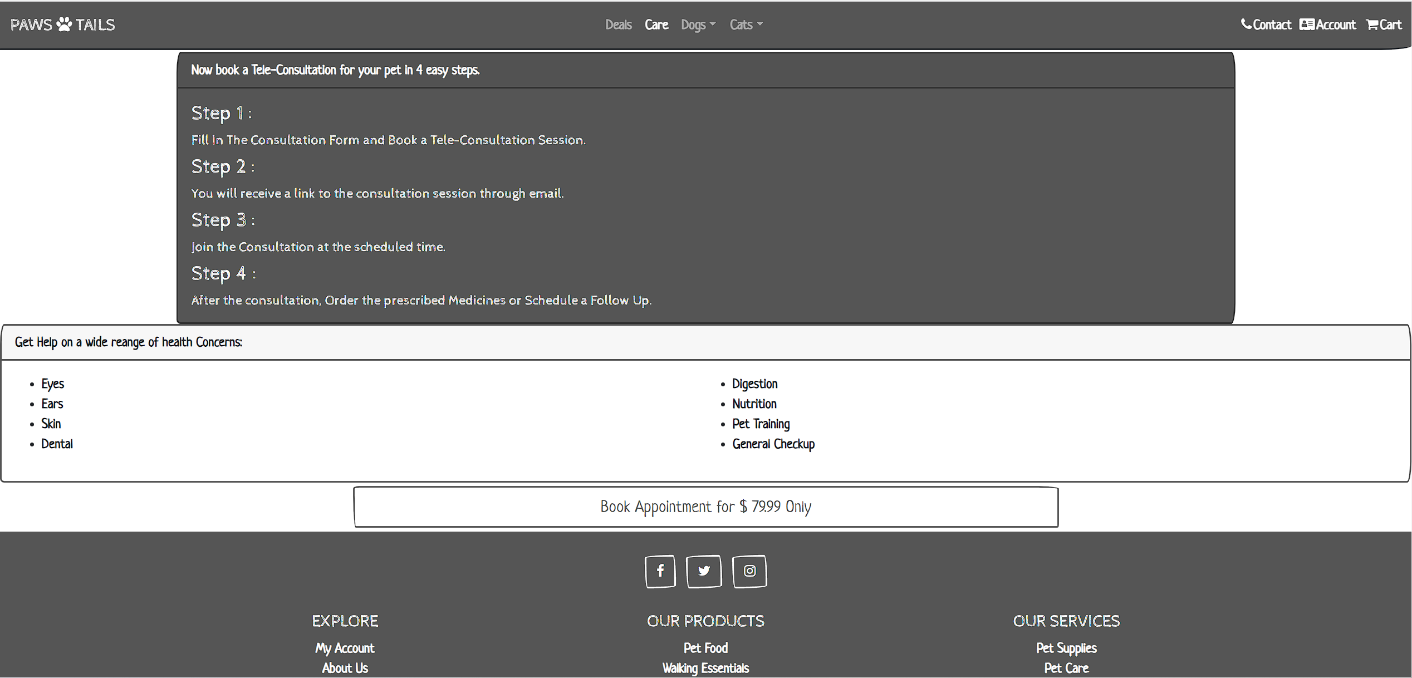
**Fig: Checkout page**

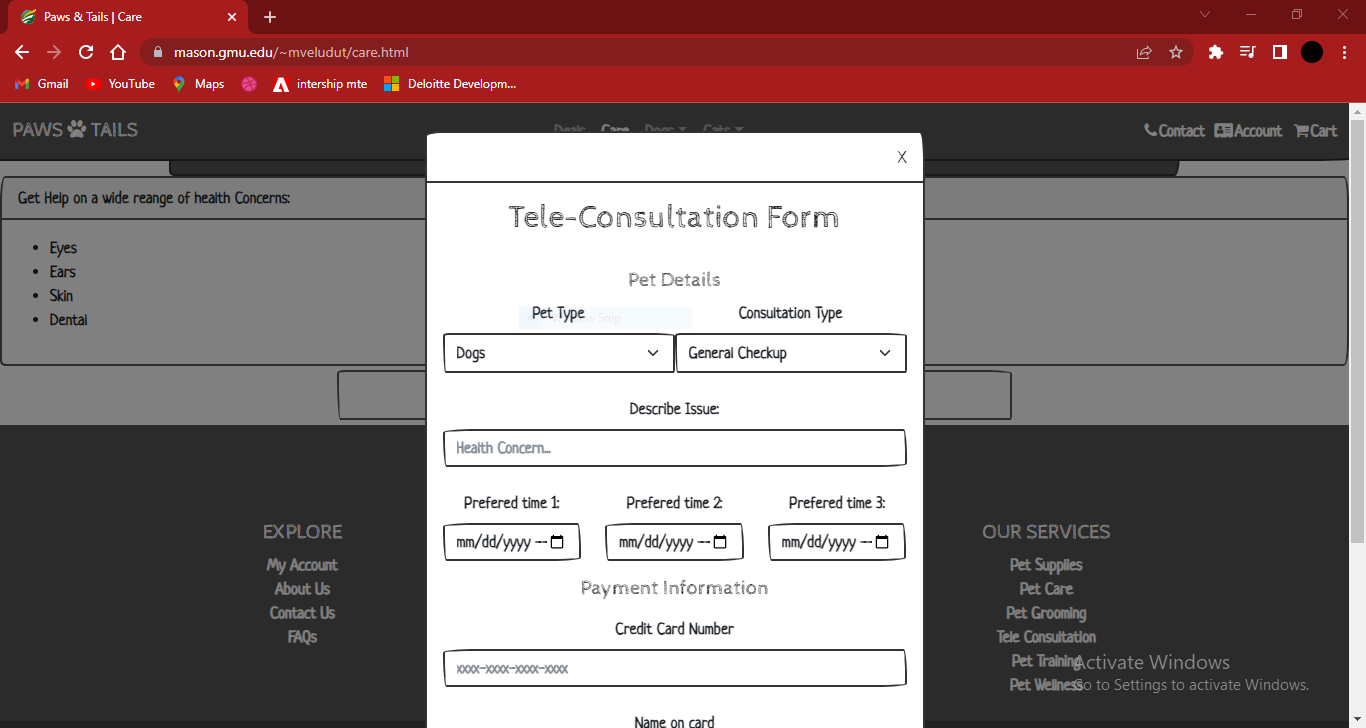
1. **Pet Care :** This use case is also one of our main use cases. Here the users get access to the care package of the company. The users can book a teleconsultation session for their pets. It covers a wide range of health concerns. It also provides advice on pet training, nutrition and other wellbeing components.

The user needs no extra knowledge, and the gulfs are also minimum in this use case. It is like the checkout feature. The user must give a brief summary on the issue or the agenda on which they seek advice and then select their available time slots and complete the payment portion. After successfully doing payment (mocked), the user will be receiving a email with all the details regarding the consultation. ( mocked, as this is just a front-end project).



**Fig: Book a Tele – Consultation State Diagram**

**Fig: Book Appointment Page**



**Fig: Book Appointment Form Pop Up**

**Graphical user interface, application

Description automatically generated**

**Fig : Appointment confirmation and Payment.**

**Testing outputs:**

1. **Selenium Testing:**

For Selenium testing, we have tested 2 of our use cases. This includes recording the test using the selenium IDE and then running the test script to check if it performs as expected.

The selenium test script files are included in the project folder as “SWE 632.SIDE”

Use case 1 Login : We have recorded the information and screen flow form the home screen. The user tries to login using their credentials and upon successful login receives gains access to certain sections of the website. The entire interaction between the end user and the system is recorded and tested. Below is the screen shot of the run of the test script and its successful passing.

Graphical user interface, text, application

Description automatically generated

**Fig: Testing output for the user login use case**

Use case 2 Cart and Checkout : We have recorded the information and screen flow form the home screen. The user first has to login using their credentials and upon successful login receives gains access to the cart and then they can view the items in cart modify them and proceed to checkout. The entire interaction between the end user and the system is recorded and tested. Below is the screen shot of the run of the test script and its successful passing.

Graphical user interface, text

Description automatically generated

**Fig: Testing output for shopping cart checkout**

1. **Jest JavaScript Testing:**

For jest testing, we have tested the cart and checkout use case of the project. We have checked if the totals, tax and subtotals are being computed properly and correctly onto the screen. We have tested this entire process by giving the function random quantities of items in the cart and checking if the correct total is generated.

Below is the screenshot of the code used for testing and the corresponding result of the various test cases included. As we may see that all the test cases have passed successfully.

Also included the file “compute.test.js” in the project folder which contains the code for this testing section.

Text

Description automatically generated

**Fig: Jest test case output for the total amount of cart items**

**Team Members:**

1. Haren Akula – G01387760.

2. Sagar Naidu Potana – G01355019.

3. Mahaveerasaiteja Veluduti – G01354854.