Implementation of a Pet Care Management System

D.M.M.P Gunaratne[1], S.M.D.S Bandara[2], H.M.T.S Sooriyabandara[3], M.G.D.P.M.B Chandrarathna[4], D. I. De Silva[5] and H.M.P.P.K.H Samarasekara[6]

[1]Faculty of Computing, Sri Lanka Institute of Information Technology, New Kandy Road, Malabe, SRI LANKA
[2]Faculty of Computing, Sri Lanka Institute of Information Technology, New Kandy Road, Malabe, SRI LANKA
[3]Faculty of Computing, Sri Lanka Institute of Information Technology, New Kandy Road, Malabe, SRI LANKA
[4]Faculty of Computing, Sri Lanka Institute of Information Technology, New Kandy Road, Malabe, SRI LANKA
[5]Faculty of Computing, Sri Lanka Institute of Information Technology, New Kandy Road, Malabe, SRI LANKA
[6]Faculty of Computing, Sri Lanka Institute of Information Technology, New Kandy Road, Malabe, SRI LANKA

[1]Corresponding Author: it20232368@my.sliit.lk

ABSTRACT

The meteoric rise of web applications has brought with it, a new layer of convenience and ease to nearly all walks of life and has been continuously enabling faster and more efficient alternatives to previously geriatric systems. Thus, the field of 'pet care', which has been stagnant in its accessibility is also currently seeing a new dawn of integration with modern technology. Now more than ever, pet owners are keenly aware of the amount of care that their pets require in order to be well looked after. However, being busy with day-to-day struggles, the common pet owner is in dire need of a convenient system to help manage their pets needs. As a country that has put relatively little emphasis on the field itself, the implementation of a functional and accessible pet care management web application would undoubtedly elevate the level of care that pets would receive, whilst also significantly increasing the ease of pet owners managing their beloved companions. This paper aims to detail the thought process behind the implementation of the application, the technology used, the services and functionalities provided, and the needs that both pet owners and pets alike would appreciate the app catering towards.

Keywords— Pet Care, Pet Care Management, Web Application

I. INTRODUCTION

Humans have long since learned to bond and share their lives with animal companions. The reasons and motivations may vary; however, it is quite clear that nearly always, both parties are involved in a mutually beneficial relationship. The human component of the relationship can experience an array of benefits, ranging from emotional support from domestic pets that would serve to lower the stress and strain of life's daily tribulations, steadfast security, and companionship, and even an irreplaceable aid in their livelihoods. Thus, it is clear that the human counterparts, or 'pet owners', are reaping many benefits from the relationships formed with domesticated animals. Likewise, the care, affection, and protection must be reciprocated to the 'pets.

As a pet owner who intends to prolong their pet's lifespan to the maximum possible age, ensure a constant level of health and well-being and keep them at their peak condition, several functions must be fulfilled for this application to be of practical use. A distillation of the possible needs of pets and pet owners has been catered to, employing the web application. Many 'pet owners', i.e., users of the web app, would require a service that would allow them to purchase any number of accessories and pet-related items. The application solves this need by having an interface that allows users to browse through a catalog of such items and purchase them within the application itself. Should the owner require a temporary keep for their pets to be boarded due to travel, or any other circumstance, the application is able to provide a list of available 'pet boarding' places, along with the information relevant to the boarding place (including but not limited to, contact information, location, and name of the establishment). This list of possible places can be filtered by the city where the places are located to ensure that the users of the application are able to easily find a convenient location.

Another common need for users would be to groom their companion, which may include services like clipping nails, bathing, etc. The application is also able to show establishments that would provide such services, which, similar to the boarding places, can be filtered by the location to allow for easy access to the user. Medical assistance and veterinary care are services that all users would require and as such the application is home to a plethora of established veterinary clinics which would be willing to provide care to animal companions. These services can be searched for, by the user, using keywords that would detail the services that they wish to procure. As an added functionality, users will be able to maintain a record of all the items, services, etc., they have used within the application and also maintain records of their companions to allow for a more convenient alternative to store the information relevant to a pet's health

e-ISSN: 2250-0758 | p-ISSN: 2394-6962
Volume-12, Issue-5 (October 2022)
https://doi.org/10.31033/ijemr.12.5.19

and wellbeing. Each of these functionalities has been implemented using MERN technology [1].

Considering these needs that the Pet Care System caters towards, this report will detail the process behind the design of the system, the thought process that went into how the various features were designed and implemented, the works which aided in the formulation of a cohesive idea for the system's function, the methodology behind the implementation of the system as well as a reflection on the process of implementing the system and any possible plans for the future of the system.

II. RELATED WORK

The care and love shown towards our animal companions have made it important for the services that this application provides to be easily accessible. This means that around the globe, many have tried and tested innumerable amounts of ways in which the average pet owner would be able to more conveniently and effectively care for their pets. These attempts have aimed at a plethora of different ways in which pet care could be enhanced and thus, these ideas have been adapted to better suit the needs of the users of the Pet Care Management Web Application. Owing to the strenuous daily life, filled to the brim with appointments and what have you, it may be easy to take the health and well-being of your pets for granted. However, constant check-ups and monitoring of the health of your pets is anything but a trivial task. Understanding this, while certain systems have chosen to focus on the veterinary clinic and medical center's points of view by providing a detailed medical history of the pet in question to the user [2], this system has chosen to prioritize the convenience and needs of the pet owner by simply providing details relevant to the appointments relevant to their pets health and well-being (Grooming, care, medical appointments, etc.).

The common problem of needing a safe and trusted place to be a temporary home for an owner's pet is also one that has been considered in alternative pet care systems, however, rather than leaving your companion at the hands of another pet owner who would be willing to house and feed them [3], it is also possible to use a pet boarding place. This alternative would potentially allow for better and more trusted care, considering that such places would have experience in sheltering pets of various kinds and also be available at any given time as opposed to being dependent on the willingness of another fellow pet owner. The information about these facilities and establishments would also be an asset to have provided for pet owners so that they may be able to browse through and select a place of their choosing, which may have a convenient location for them and their needs. This idea has further been extended to encompass the details and locations of veterinary and medical care centers, which would also be easy to search for and filter by means of the location.

Along with this, another useful function that has been considered by other developers with a concern for pet care is that pets and pet owners would require a near-endless supply of various kinds of products. Both perishable products such as specialized pet food and healthcare items along with equipment to aid in raising pets and even toys etc. are products that almost all pet owners would require at some point or another. Due to this undeniable need to be able to procure items that would be necessary to combat the daily challenges brought about by raising pets, certain systems have focused on pet shops as the beneficiaries of their functions. These systems would allow for easy cataloging, browsing, and recording of various pet-care items and goods [4]. The Pet Care Management System on the other hand would need to solve the same issue, whilst focusing on the perspective of the pet owners, thus leading to the implementation of a pet-care item purchasing function, which would allow users to browse through, locate and purchase their choice of an accessory or good for their own pets.

III. METHODOLOGY

"Pet Care Management System" consists of; React, a frontend JavaScript library for building web apps; MongoDB, a document based open-source database; NodeJS the runtime environment of JavaScript and Express, a web application framework for NodeJS.

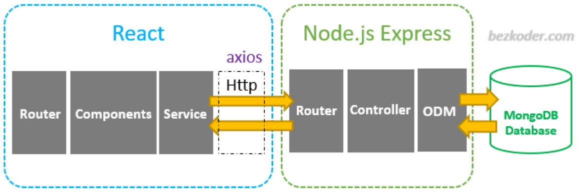


Figure 1: MERN Stack Development Architecture

In figure 1 it explains the main parts of a full stack application. Full stack application generally consists of two main parts, the frontend, and the backend. The frontend is usually the code that displays the application in the user's browser. The backend consists of all the logic that represents the application that is running on servers and connecting with the database. The frontend is generally developed with ReactJS. The developer writes everything that has to do with displaying data to the user, re-rendering things in the browser working with DOM elements, etc.

The backend server handles all of the HTTP requests coming from the frontend and loads data from the database. The database is a place where all of the application's data is stored in schemas.

e-ISSN: 2250-0758 | p-ISSN: 2394-6962
Volume-12, Issue-5 (October 2022)
https://doi.org/10.31033/ijemr.12.5.19

The 'Pet Care Management' project uses ReactJS [5] as the frontend. React is one of the most popular JavaScript libraries for building user interfaces. It is one of the fastest and most flexible JavaScript libraries to build apps and it has a strong community to help fellow developers should they face any problems. By using ReactJS developers can build the application based on the components. Programmers break down the complex code into individual pieces which means developers divide the entire user interface into reusable components. Then we merge all the separate components into parent components which are then rendered.

Programmers must have basic knowledge of HTML [6], CSS [7], and JavaScript [8] to learn ReactJS. HTML is used to develop web applications. It is comparable to the skeleton of the human body. CSS means Cascading Style Sheet. It is used to design and give styles to the website and can apply colors, fonts, and provides many other stylistic choices to the developers. Making interactive web pages is possible with JavaScript, a text-based computer language used both on the client-side and server-side.

After familiarizing themselves with these web development languages, programmers have to learn npm. Npm is a node package manager. It is used to install node modules and the package into the project.

Fundamentals of ReactJS like component architecture, state, props functional component, class, and how to connect API with React app, etc., are necessities when developing a ReactJS based web application. React router will help developers to load the content of specific interfaces and to redirect to specific pages. Server Render, and webpack help developers to maintain dependencies as a static file of the project. Webpack contains Loaders which help to run specific tasks in the project.

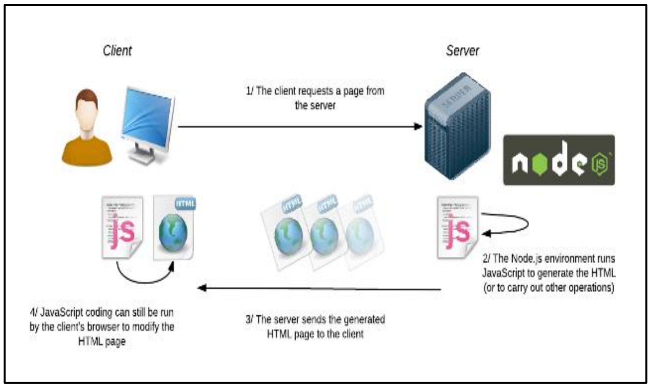

Client
Server
1/ The client requests a page from the server
2/ The Node.js environment runs JavaScript to generate the HTML (or to carry out other operations)
3/ The server sends the generated HTML page to the client
4/ JavaScript coding can still be run by the client's browser to modify the HTML page
node
JS
HTML

Figure 2: NodeJS: what is NodeJS for

In figure 2, NodeJS [9] is neither a language nor a framework. It is a free and open-source runtime for JavaScript. The JavaScript engine executes the code in the runtime environment just like NodeJS, however, it has some additional server-side modules. NodeJS is written in C, C++, and JavaScript, which is why NodeJS is extremely fast and has an amazing performance. NodeJS has great features, and that is why NodeJS is able to run the JavaScript code on the server.

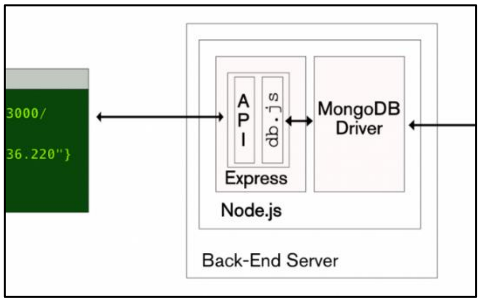


Figure 3: RESTful API using node.js and express.js

ExpressJS [10] is a NodeJS framework designed for building APIs, web applications and cross-platform mobile apps quickly, and easily. It has a very high performance, it is fast, lightweight, and unopinionated. Express does not force programmers to write code in a fixed manner. ExpressJS is used as a server-side scripting language. If programmers have to write a rest API, ExpressJS will reduce the time we have to code. That was the reason behind creating the ExpressJS framework for NodeJS. This is what demonstrates in the figure 3.

MongoDB [11] is a document-oriented, no sequel (NoSQL) database. In August 2009, it was first made available. When it comes to relational data structures, MongoDB uses documents in place of the rows that are typical in those models. Developers can work with changing data models due to its flexibility. MongoDB can describe complicated hierarchical relationships with a single record and supports embedded documents, arrays, and other document-based features. Since the document's defined keys are not fixed, it is also schema-free. Large-scale data migrations are therefore not an option.

Before programmers get started with the backend implementation, programmers have to install the following software,

- VS Code or any other editor
- Latest version of Node.js
- Postman API

Creating Frontend

In order to quickly set up a development environment for the application, first install Node and npm. Here [9] visit the NodeJS website and install version 16.18.0 LTS. Now developers are able to create the frontend of the project.

For that, the command run "create-react-app pet-care-management" can be used. This is going to create a folder and install the necessary packages in that folder to be able to run React. Type npm start within that folder, and this is going to launch a local server.

e-ISSN: 2250-0758 | p-ISSN: 2394-6962
Volume-12, Issue-5 (October 2022)
https://doi.org/10.31033/ijemr.12.5.19

Creating Backend

1. Create a new folder and name it and then open the folder in VS code and run the command “npm init -y” to initialize the project.
2. Run the command “npm i cors dotenv express mongoose” in the terminal to install the dependencies.
3. Change the main entry point to the server.js file.
4. Now create the folders called database, controllers, models, and routers. The figure 4 displays the above mentioned folder structure.

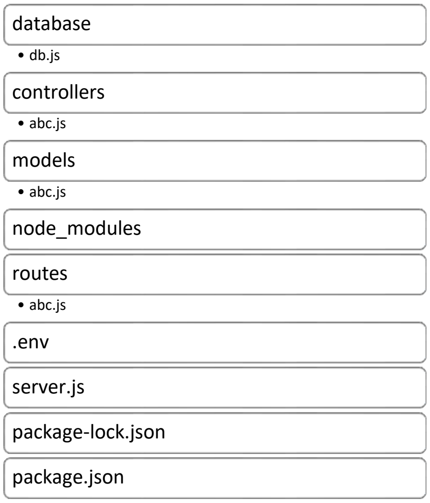
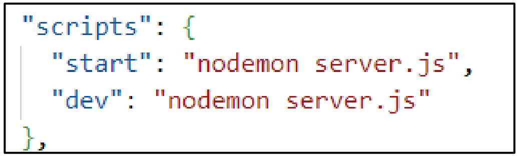


Figure 4: Folder structure of the backend

5. Change the scripts to the following shown in figure 5.



```
"scripts": {
  "start": "nodemon server.js",
  "dev": "nodemon server.js"
},
```

Figure 5: Backend package.json file

6. To set up the server, programmers have to import express, then initialize the app using express(), set up a get method for the endpoint “http://localhost:3000” using app.get(), and set the PORT to 3001 for the server to run.
7. Start the server using nodemon using the code “npm run dev”. If the server has started successfully then it should show a “server is running on port 3000” message on the terminal.
8. Get the MongoDB URL from the MongoDB database and paste it into the .env file. Replace the <username> and <password> with the database username and password.
9. Open the index.js file, which is in the database folder and import mongoose, import the database URL from the .env file, create the database connection method for connecting to the database and export the method and call it in the server.js file.
10. Create separate files in the models folder and define the database schemas for the services we are providing in this application such as boardingSchema, productSchema, etc.
11. Define the methods for the end points such as create, update, delete, and retrieve methods, etc.
12. Add the routes end points in the server.js
13. Test the created end points using Postman API [12].

Tools used for the Implementation

Azure board [13] is one of the main services of Azure DevOps which is used in a project to track the work with Kanban boards which makes it easier to complete the tasks as a team. Whenever programmers use Azure DevOps, developers will have an Azure board and in that Azure board developers can create and set up a Kanban board. A Kanban board is something which will have the details of all the work items which developers are going to deliver or work in a particular project.

In the Azure board, it is easy to track and work on the tasks that are in the project backlog. The backlog will have all the tasks which need to be delivered in a current or upcoming sprint. Azure boards will also provide different types of reports. In Kanban board developers have to update their tasks and their features as they progress with the project. Azure boards are useful in managing and tracking tasks, it provides a clear picture of work done/ being done by team members.

GitHub [14] is a hosting provider which offers a web-based graphical user interface. GitHub makes it simple for team members to collaborate, track, and update their work while enabling them to work on the project from any location. Therefore, projects continue to be open and on time. Using GitHub, the team can stay coordinated and on the same page.

Pull requests on GitHub assist teams in reviewing, enhancing, and suggesting new codes. Before making changes to the source code, the implementations and suggestions can be addressed.

SonarQube [15] basically runs over the code of a developer and finds mistakes early on. It is a static testing analysis program that is open source. Developers use it to control the consistency and quality of their source code. Potential flaws, design inefficiencies caused by code faults, duplication of code, inadequate test coverage, and other issues are some of the code quality checks.

Selenium [16], is a free (Open Source) automated testing platform for web applications across various browsers and platforms. Web-based application automation is Selenium's primary focus. Selenium is not just one tool but rather a group of programs that each address specific testing requirements of an enterprise.

IV. PROPOSED SYSTEM

e-ISSN: 2250-0758 | p-ISSN: 2394-6962
Volume-12, Issue-5 (October 2022)
https://doi.org/10.31033/ijemr.12.5.19

The Pet Care Management Application would run on an interactive system that engages the user in a way which would allow the user to navigate through all services that are available on the application. These services are,

- Browsing/purchasing pet products or accessories
- Finding details regarding pet boarding places (which can be filtered by city)
- Finding details regarding Veterinary and/or animal healthcare service establishments.
- Storing information regarding the user's pets as well as keeping track of their pet-related appointments.

The wireframe that has been created for the implementation of The Pet Care Management System is as follows:

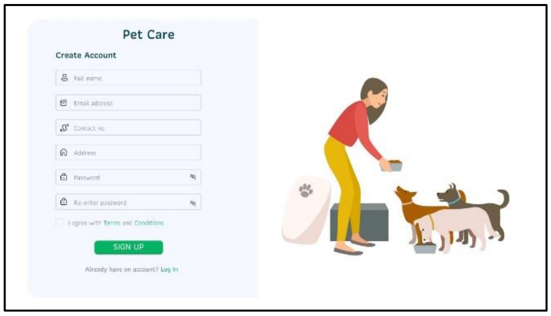


Figure 6: Wireframe of registration page

Here the user navigates to create an account through the figure 6 interface. User will enter their details and register to the system. Before registering the user must agree with the terms and conditions. Here, if the user already has an account, the user can go back to login page from the "already has an account" link.

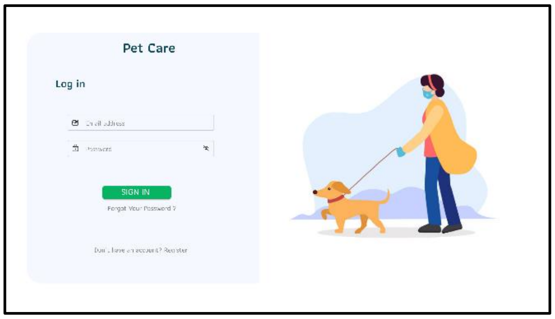


Figure 7: Wireframe of login page

On this login page shown in figure 7, the user can enter their email and password. If those details are correct, the user can log in to the system. If not, a message will display that the username or password is incorrect. If the user has not created an account yet, the user can click on the "Don't have an account? Register" link at the bottom. If the user forgot the password, the user can reset the password by clicking the "forgot password" link.

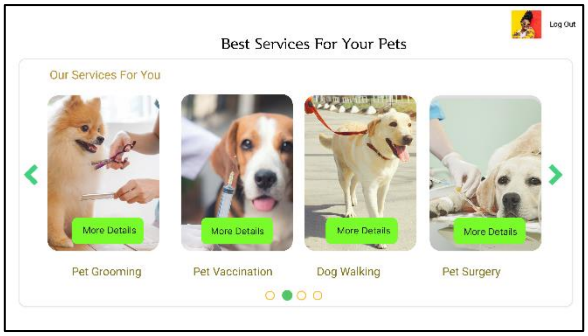
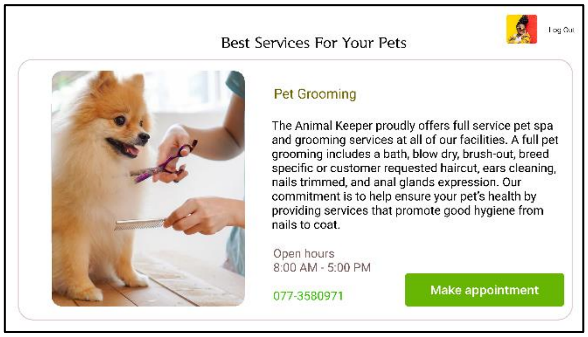


Figure 8: The pet services wireframe

The services entered by the admin will be added to the service's homepage. After that, pet owners can view those services. If the user wants to view more details, the user can click the relevant "more details" button represented in figure 8

After clicking the relevant "more details" button, the pet owner will navigate to the more services page the figure 9 interface will be displayed. At that interface, the user can view more services about the establishment. If the user wants to make an appointment, the user can book it by clicking the "make appointment" button. Then the user will navigate to the appointments page.



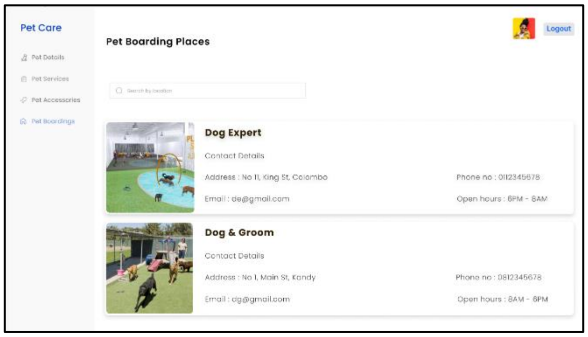
Pet Grooming

The Animal Keeper proudly offers full service pet spa and grooming services at all of our facilities. A full pet grooming includes a bath, blow dry, brush-out, breed specific or customer requested haircut, ears cleaning, nails trimmed, and anal glands expression. Our commitment is to help ensure your pet's health by providing services that promote good hygiene from nails to coat.

Open hours
8:00 AM - 5:00 PM

077-3580971

Figure 9: The pet services detailed wireframe



Pet Boarding Places

Dog Expert
Contact Details
Address : No 11, King St, Colombo | Phone no : 0112345678
Email : de@gmail.com | Open hours : 6PM - 8AM

Dog & Groom
Contact Details
Address : No 1, Main St, Kandy | Phone no : 0812345678
Email : dg@gmail.com | Open hours : 8AM - 6PM

Figure 10: Wireframe of pet boarding places

e-ISSN: 2250-0758 | p-ISSN: 2394-6962
Volume-12, Issue-5 (October 2022)
https://doi.org/10.31033/ijemr.12.5.19

Here the added pet boardings by the admin will be displayed to the pet owner. Pet owners can go through the details and if the user wants to get more details about the boarding, the user can contact the boarding place for more information. Pet owners can also search the boarding places by city. The figure 10 demonstrates the pet owner view of boarding places interface.

Once a pet owner logs in to the system user will be redirected to this interface. In figure 11 the he user can choose a product that they want to purchase by clicking on it.

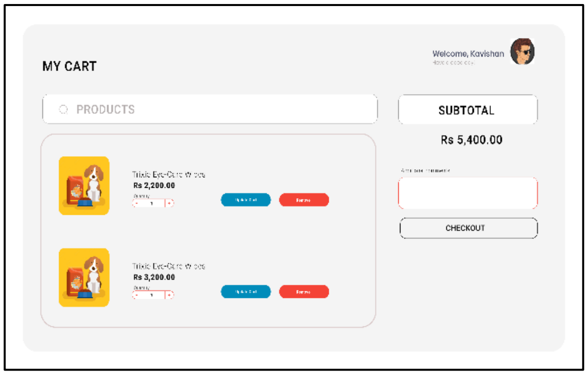
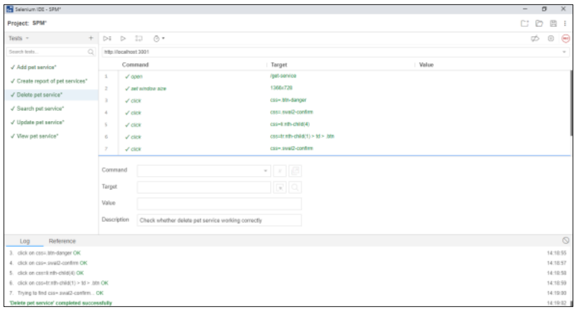


Figure 11: Wireframe of pet products



http://localhost:3001

Tests:

- ✓ Add pet service*
- ✓ Create report of pet services*
- ✓ Delete pet service*
- ✓ Search pet service*
- ✓ Update pet service*
- ✓ View pet service*

	Command	Target	Value
1	✓ open	/pet-service	
2	✓ set window size	1366x728	
3	✓ click	css=.btn-danger	
4	✓ click	css=.swal2-confirm	
5	✓ click	css=li:nth-child(4)	
6	✓ click	css=tr:nth-child(1) > td > .btn	
7	✓ click	css=.swal2-confirm	

Command:
Target:
Value:
Description: Check whether delete pet service working correctly

```
3.  click on css=.btn-danger OK                              14:18:55
4.  click on css=.swal2-confirm OK                           14:18:57
5.  click on css=li:nth-child(4) OK                          14:18:58
6.  click on css=tr:nth-child(1) > td > .btn OK              14:18:59
7.  Trying to find css=.swal2-confirm... OK                  14:19:00
'Delete pet service' completed successfully                  14:19:02
```

Figure 12: View cart wireframe

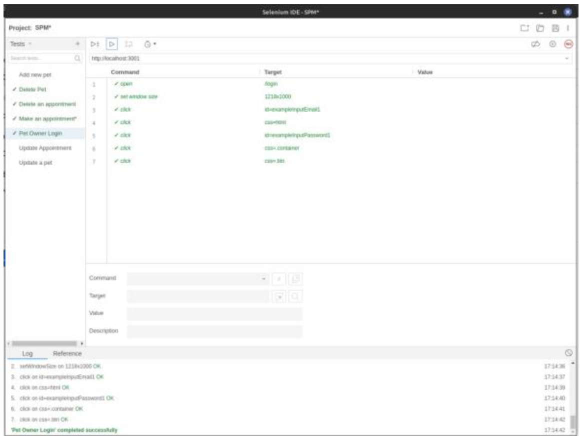
When the users choose to check out, they will be presented with a final list of items in the order, as well as options as to how they want to pay is displayed in figure 12.

V. DISCUSSION

This Pet Care System has been implemented with the intention of allowing pet owners to be able to provide for their pet needs with ease and convenience. With the help of the Selenium IDE test automation tool, all system functionalities have been thoroughly tested. It is an open-source test automation tool for web applications that can record and replay tests [16].

The main functionalities of the system such as pet user management, pet services, boardings, and accessories are well-tested with the automation tool to interact effectively with the user satisfaction. Selenium IDE has a script panel and log panel to give the result of the test case. First, create separate test cases for each of the functionalities to run the test cases.

When it comes to user management the main functions such as user login, registration, and updating profiles are tested. Among those functionalities, login and registration, functionalities ended with good results. Profile updating had one error. After the identification of that error and modification, the test case got a positive result which is shown in figure 13.



http://localhost:3001

Tests:

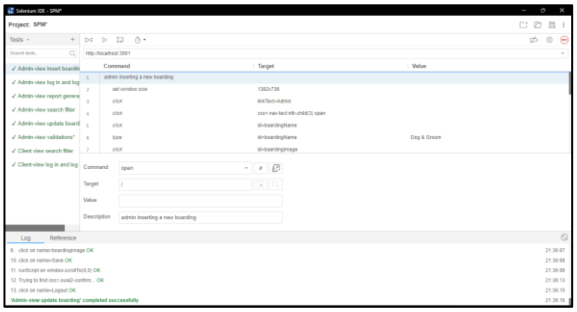
- Add new pet
- ✓ Delete Pet
- ✓ Delete an appointment
- ✓ Make an appointment*
- ✓ Pet Owner Login
- Update Appointment
- Update a pet

	Command	Target	Value
1	✓ open	/login	
2	✓ set window size	1218x1000	
3	✓ click	id=exampleInputEmail1	
4	✓ click	css=html	
5	✓ click	id=exampleInputPassword1	
6	✓ click	css=.container	
7	✓ click	css=.btn	

Command:
Target:
Value:
Description:

```
2.  setWindowSize on 1218x1000 OK                            17:14:36
3.  click on id=exampleInputEmail1 OK                        17:14:37
4.  click on css=html OK                                     17:14:39
5.  click on id=exampleInputPassword1 OK                     17:14:40
6.  click on css=.container OK                               17:14:41
7.  click on css=.btn OK                                     17:14:42
'Pet Owner Login' completed successfully                     17:14:42
```

Figure 13: User login test case with results



http://localhost:3001

Tests:

- ✓ Admin-view insert boardin
- ✓ Admin-view log in and log
- ✓ Admin-view report genera
- ✓ Admin-view search filter
- ✓ Admin-view update board
- ✓ Admin-view validations*
- ✓ Client view search filter
- ✓ Client-view log in and log

	Command	Target	Value
1	admin inserting a new boarding		
2	set window size	1382x736	
3	click	linkText=Admin	
4	click	css=.nav-text:nth-child(3) span	
5	click	id=boardingName	
6	type	id=boardingName	Dog & Groom
7	click	id=boardingImage	

Command: open
Target: /
Value:
Description: admin inserting a new boarding

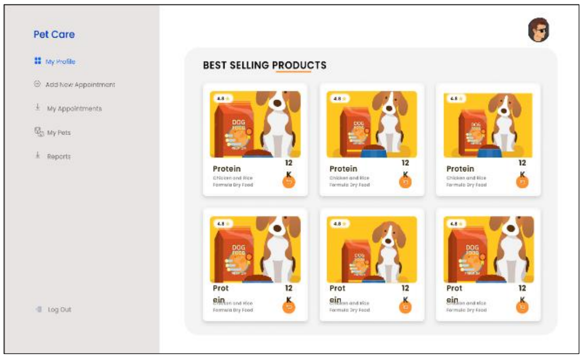
```
9.  click on name=boardingImage OK                           21:36:07
10. click on name=Save OK                                    21:36:08
11. runScript on window.scrollTo(0,0) OK                     21:36:09
12. Trying to find css=.swal2-confirm... OK                  21:36:13
13. click on name=Logout OK                                  21:36:15
'Admin-view update boarding' completed successfully          21:36:16
```

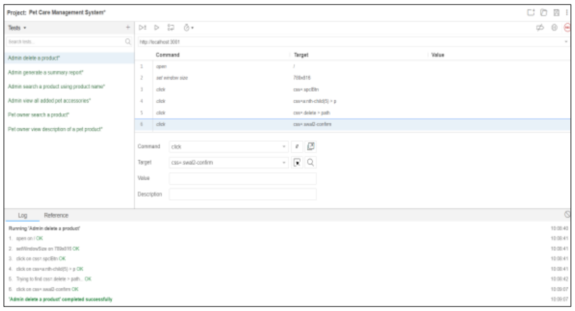
Figure 14: Delete pet service test case

The homepage and admin parts of pet services are tested to check the functionalities. Some of the functionalities exhibit errors. Successful modification of those functionalities comes up with successful test results in figure 14.

The main functions of the pet boardings, such as adding, updating, deleting, pet boarding view, and searching boardings are created with separate test cases to check whether those functions are working correctly. All the functions come up with positive test results, shown in figure 15.

e-ISSN: 2250-0758 | p-ISSN: 2394-6962
Volume-12, Issue-5 (October 2022)
https://doi.org/10.31033/ijemr.12.5.19




Project: Pet Care Management System*

Tests

- Admin delete a product*
- Admin generate a summary report*
- Admin search a product using product name*
- Admin view all added pet accessories*
- Pet owner search a product*
- Pet owner view description of a pet product*

http://localhost:3001

	Command	Target	Value
1	open	/	
2	set window size	789x816	
3	click	css= spcBtn	
4	click	css=a:nth-child(5) > p	
5	click	css= delete > path	
6	click	css= swal2-confirm	

Command: click
Target: css= swal2-confirm
Value:
Description:

```
Running 'Admin delete a product'                    10:08:40
1. open on / OK                                     10:08:41
2. setWindowSize on 789x816 OK                      10:08:41
3. click on css= spcBtn OK                          10:08:41
4. click on css=a:nth-child(5) > p OK               10:08:41
5. Trying to find css= delete > path... OK          10:08:42
6. click on css= swal2-confirm OK                   10:09:07
'Admin delete a product' completed successfully     10:09:07
```

Figure 15: Admin view update boarding test case

The main functions of the pet accessories such as add, view, update and delete are well-tested to check the functionality of the system. All the test cases are finished with a positive results which is shown in figure 16.

Figure 16: Admin delete product test case

VI. CONCLUSION

When analyzing this research, the developed pet care management system covers the overall sections of a pet's needs. That will provide good interaction to the user so that they can easily perform their pet's activities easily and accomplish them quickly.

In summary, the developed system can perform record handling of information relevant to the pet. That includes managing pet users, their boardings, services, and products. Users are notified about the status of pets. Administrators can monitor the records of pet details performed by pet owners. They can monitor how many pet users are registered, and how they perform in the various parts of the system. The system can generate separate analytical reports for each section to analyze the records.

We aim to enhance the system with IoT technologies which use sensors and wireless communication, such that it helps feed the pets. So that helps to feed the pets while the owners are absent at their homes. The system will be developed with weight sensors and timers so that users can set regular intervals of feeding time and amount of food based on their pets' weight [17]. The user can control the feeder through the system and see the status.

The system will enhance functionality to add a camera with a Raspberry Pi server. Its major interest is to monitor the movements of the pet, so that the pet owners can see the whole life of their pet throughout the system.

REFERENCES

[1] https://www.mongodb.com/mern-stack
[2] E. M. T. Abdalameer. (2016). Pet health care system. *Journal University of Kerbala*, *14*.
[3] V. Saswadkar. (2018). Pet care system based on android application. *International Journal for Research in Applied Science and Engineering Technology*, *6*(3), 1915–1919
[4] https://www.researchgate.net/publication/4376159_Pet_Shop_Management_System_for_Klinik_Veterinar_Surgeri_Jawhari.
[5] https://reactjs.org/.
[6] https://www.w3schools.com/html/.
[7] https://www.w3schools.com/css/.
[8] https://www.w3schools.com/Js/ .
[9] https://nodejs.org/en/.
[10] https://expressjs.com/.
[11] https://www.mongodb.com/.
[12] Postman. (2021). Postman | The collaboration platform for API Development. Available at: https://www.postman.com/.
[13] https://azure.microsoft.com/en-us/products/devops/boards/.
[14] https://github.com/.
[15] https://www.sonarqube.org/.
[16] https://www.selenium.dev/selenium-ide/.
[17] Kim1, S. (2016). Smart pet care system using internet of things. *International Journal of Smart Home*, *10*, 1–8.