

Secure Cat Hotel and Grooming Booking System for Kayangan Pet House using OTP

Nur Ezattie Suhaime¹, Nordiana Rahim^{1*}

¹ *Fakulti Sains Komputer dan Teknologi Maklumat,
Universiti Tun Hussein Onn Malaysia, Parit Raja, Batu Pahat, 86400, MALAYSIA*

*Corresponding Author: nordiana@uthm.edu.my

DOI: <https://doi.org/10.30880/aitcs.2026.07.01.060>

Article Info

Received: 22 October 2025

Accepted: 18 June 2026

Available online: 30 June 2026

Keywords

Web Based System, Structure
Approach, Cat Booking System,
OTP

Abstract

The rapid development of information technology has revolutionized service industries, enabling efficient and user-friendly processes. In the pet care industry, specifically for cat hotels and grooming services, technology offers opportunities to enhance customer experience. The Secure Cat Hotel and Grooming Booking System for Kayangan Pet House, leveraging OTP authentication, addresses challenges in traditional service models. These include time-consuming manual processes, repetitive data entry, limited online booking accessibility, lack of transparency in grooming packages, and difficulties in scheduling and managing appointments. The proposed system is developed using PHP, MySQL, and XAMPP, with Notepad++ and Lucid chart aiding coding and workflow visualization. The platform supports three user roles: Administrators, Staff, and Users, providing features such as service management, package display, and appointment scheduling. By streamlining booking and management processes, the system enhances accessibility, saves time, and improves workflow efficiency for all users, ultimately delivering a more seamless and productive experience. Based on the testing result, all the module is passed from the test plan list and this system is accepted by the user based on the user acceptance testing.

1. Introduction

Kayangan Pet House is a stylish cat hotel and grooming service provider located in Taman Bunga Cempaka Biru, Parit Raja, Johor. It offers a range of services designed to pamper and care for cats, particularly when their owners are away on vacation or business trips. Over time, the concept of a boarding cattery has evolved to provide more comfort and specialized care for cats, ensuring they receive the best treatment during their stay. Kayangan Pet House began its operations on March 13, 2023, and has been managing customer information manually using Word and Excel. This approach is burdensome on their Central Processing Unit (CPU) storage, causing computers to slow down and potentially crash, leading to data loss and security issues [1]. The lack of a security system means unauthorized access to sensitive data is possible, and without proper backup, losing all customer information could lead to significant financial loss and operational disruption [2]. The current manual system also affects booking efficiency. With no real-time availability tracking, customers face delays and uncertainty, leading to dissatisfaction. The absence of a streamlined, secure booking system results in increased operational risks and customer inconvenience, highlighting the need for a more reliable, automated solution.

To address the issues stated above, the Cat Hotel and Grooming Booking System has been designed and developed. Several objectives are outlined to achieve the aim:

This is an open access article under the CC BY-NC-SA 4.0 license.



- 1) To design a web-based Cat Hotel and Grooming booking system for Kayangan Pet House using OTP authentication mechanism.
- 2) To develop Secure Cat Hotel and Grooming booking system for Kayangan Pet House using OTP.
- 3) To test the functionality, security, and responsiveness of the web-based system for Kayangan Pet House.

Three users are involved in the system, including the administrator, staff and pet owner. This system includes several functional modules such as a login/logout module, register module, user profile management module, services management module, service booking module, operation management module, staff management and report.

This paper is organized as follows: Section 2 focuses on a literature review of the related works and existing applications related to the topic. Next, Section 3 describes the methodology used to develop the system. Section 4 explains the analysis and design of the system. Section 5 covers development, deployment, testing. Lastly the final section summarizes the result, discussion and conclusion.

2. Literature Review

This section discusses the literature review conducted for the Cat Hotel and Grooming Booking System named Kayangan Pet House. The Booking Management System is a software tool to assist in managing bookings and appointments of businesses while allowing smooth operations. In the context of Kayangan Pet House, the Booking Management System coordinates not only the services provided but also revolutionizes the customer experience by introducing an unprecedented level of convenience and reliability in accessing and managing reservations. Real-time tracking of the availability of services enables pet owners to make informed decisions and secure bookings without unnecessary delays.

2.1 Current system for Kayangan Pet House

The current booking system of Kayangan Pet House is of a very basic nature and does not have any dedicated platform for managing pet accommodation and grooming. The system, in the present scenario, relies much on Google Docs for managing bookings, where the pet owners must fill out the form manually for requesting reservations for their cats. This system becomes prone to human error as there is no automated process regarding verification of availability, confirmation of bookings, or effective management of customer data. Also, most of the time, there is a need for supplementary calls after the initial booking to confirm some information about the reservation, request certain services, or to get answers about how their animals will be looked after and accommodated. In sum, all this process with the bookings takes up a lot of time both for the owners and for the staff because it introduces supplementary steps for making things clear and confirmed. It creates inefficiencies in that there are going to potentially be miscommunications, there might be some booking errors, and some delays in response time. Nor does it allow any scalability in the sense that the current system cannot handle an influx on bookings, nor real-time updates for customers.

2.2 Existing System

Three existing systems' features are investigated: Talisay Pet Care Center System [3], Lovely Cuts Pet Grooming System Web-Based System [4], and Boo Pet Groomers System [5]. The differences between existing and developed systems are divided into several features and aspects. The feature comparison is tabulated in Table 1. Compared to the three existing systems, all characteristics can be found in the developed system. The developed system has similar characteristics to Talisay Pet Care Center System, except there is an additional feature in the developed system which is responsive design.

Table 1 Comparison between Existing Systems and the Developed System

Features/ System	Talisay Pet Care Center [3]	Lovely Cuts Pet Grooming System Web-Based [4]	Lovely Cuts Pet Grooming System Web-Based [5]	Cat Hotel and Grooming Booking System for Kayangan Pet House
Easy to use	/	/	/	/
Easy to access	/	/	/	/
Provide Booking history	/	X	X	/
Provide available time slots	/	X	X	/

Table 1 (cont.)

Features/ System	Talisay Pet	Lovely Cuts Pet	Lovely Cuts Pet	Cat Hotel and
------------------	-------------	-----------------	-----------------	---------------

	Care Center [3]	Grooming System Web-Based [4]	Grooming System Web-Based [5]	Grooming Booking System for Kayangan Pet House
Provide available dates	/	X	X	/
An account is needed for booking	/	X	X	/
Responsive design	X	/	X	/
Pet owner details saved in the system	/	X	X	/
Provide profile update function	/	X	X	/
Provide a fast and simple booking process	/	/	/	/
Security implementation (strong password)	/	/	/	/

2.3 System of Cat Hotel and Grooming Booking System

The Kayangan Pet House Cat Hotel and Grooming Booking System is designed to make the process of booking easier and more manageable in terms of accessibility and user experience. Major features that will be integrated into this system include One-Time Password Authentication, responsive website design with a booking interface. This system will help alleviate some of the drawbacks present with manual booking and offer a seamless booking experience at par with the industry of pet care.

3. Methodology

This section presents the methodology used in this system, provides a comprehensive view of the system development workflow, offering insights into the sequential steps taken during development. Iterative Waterfall model is used as references for the development of Cat Hotel and Grooming Booking System for Kayangan Pet House using OTP. There are five steps of iterative waterfall model that are requirement analysis, system design, implementation, testing and maintenance. Also, this section provides a comprehensive view of the system development workflow, offering insights into the sequential steps taken during development.

3.1 Iterative Waterfall Model

In Iterative Waterfall Model development, the project can be developed in parts that result faster development and product can be seen earlier and as per the feedback the process can be repeated in newer versions [6]. This modified version has the most attractive aspect that it enables the phases to overlap when needed, unlike in a pure waterfall model where no overlapping was allowed. The iterative waterfall model comprises five phases which are Requirement Analysis, System Design, Implementation, Testing and Maintenance phase [7].

In the requirement analysis phases, in this phase, three similar existing systems were studied. Then, the comparison of the characteristics between the existing system and the booking system was conducted to analyze the characteristics of a similar system. A face-to-face interview with the owner of Kayangan Pet House was conducted at the Kayangan Pet House Store to understand more about the service reservation process. During the interview session, the Kayangan Pet House owner explained the flow of the reservation process and provided the details of the service provided by Kayangan Pet House, such as the service process, price range, estimated time needed for a service, and others. Besides that, the requirements of the booking system have also been analyzed.

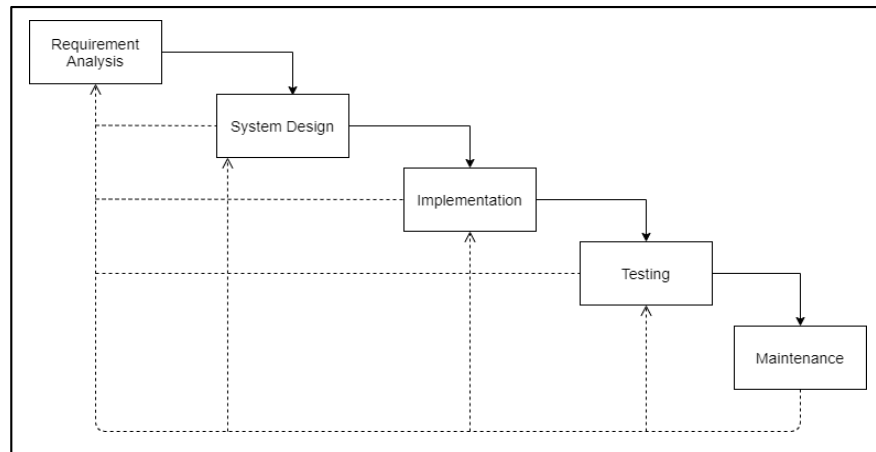


Fig. 1 Interactive Waterfall Model

In system design phase, a wireframe for the system user interface is created by using draw.io. Wireframes are mainly employed to translate abstractions into something tangible and to aid in product conception [8]. Besides that, a prototype of the system user interface is created using the Figma interface design tool. The prototype is an early design model of the final user interface of a website or application, complete with interactive user interfaces and animations [9]. Data Flow Diagram (DFD) and Entity Relationship Diagram (ERD) are also created in this phase. ERD is used to build a database system to store and manage data inserted by users into the system. At the same time, DFD shows the information flow for each process or system. In this phase, the source code is written according to the specifications. The physical design specifications are converted into functional code. The implementation phase will leverage diagrams from the design phase, such as entity relationship diagrams (ERDs) and data flow diagrams (DFDs), as well as blueprints, to solve problems, optimize solutions, and implement the components indicated in the need phase. The entire implementation step takes time to complete. The data acquired in the previous stage will be used to develop the Cat Hotel and Grooming Booking System. This phase requires implementing and completing the connection of database and system modules. The Cat Hotel and Grooming Booking System is built with several programming languages, including Hypertext Markup Language (HTML), Cascading Style Sheets (CSS), and JavaScript on the front end, and Hypertext Preprocessor (PHP) server-side scripting on the back end. In addition to coding, system error detection is a component of the implementation phase. If a system error occurs, the developer must rectify the code before continuing to the following coding section. These two crucial steps will continue until the system is completed.

Functionality testing is carried out in this system to ensure that all the requirements are satisfied. In this phase, test plan is used to test each functionality of the proposed system. There are four categories of test plan which are user interface testing, system functionality testing, security requirement testing and input validation testing. After the development of the proposed system, it will be tested by the users to ensure the system follows and meets the objectives and the requirements of the project.

In maintenance phase, maintenance is necessary for a system to operate smoothly to satisfy the users' needs. Therefore, problems must be detected so that the maintenance can be carried out on time. In this project, maintenance is executed on the system thus the system is up to date. Developers must fix the coding if imperfections and bugs are found during execution of this system. Using this way, the system can be ensured that it runs efficiently. The developer can visit the user to understand the issues errors created by the system to enhance the system. System modification is done according to change request of the user. However, due to the time limitation, this phase will not be performed in this project.

4. Analysis and Design

This section discusses the analysis and design of system development. This section includes a comprehensive view of the system development workflow, offering insights into the sequential steps taken during development. The functional and non-functional requirements of the developed system are explained and obtained in detail. In addition, diagrams such as Context Diagram, Data Flow Diagram (DFD) Level 0, Entity Relationship Diagram (ERD) and wireframe will be illustrated to identify the performance of the developed system.

4.1 System Development Workflow

Each phase of the waterfall model involves different task activities and processes, and all tasks must be completed on time. There is a total of five main phases in the waterfall model, which are the requirement analysis phase, system design phases, implementation phase, testing phase and maintenance phase. To ensure

that the system meets the requirements, enhancements and modifications have been made to fix the bugs and errors. Each project development phase has its task and deliverables, as illustrated in Table 2.

Table 2 *System Development Workflow*

Phase	Task	Output
Requirement Analysis Phase	- Define the project scope. - Preparing proposal. - Create Gant chart. - Identify problems and requirements. - Plan how to develop functional system requirements. - Plan a work schedule. - Interview with the owner. - Collect specific details.	- Proposal document. - Gantt chart. - Functional and non-functional requirements.
System Design Phase	- Design wireframes. - Design database. - Design prototype. - Illustrate Entity Relationship Diagram (ERD) and Data Flow Diagram (DFD).	- Wireframe. - User interface design. - Database design. - Entity Relationship Diagram (ERD). - Data Flow Diagram (DFD).
Implementation Phase	- Develop system module. - Integrate the system module. - Database connection.	- PHP and MySQL programming. - System with database connection.
Testing Phase	- System testing. - Evaluation by user. - Identify and fix problems and errors in the system.	- Test report. - User evaluation result.
Maintenance Phase	Fix the problem.	Final completed system.

Table 2 outlines key tasks and expected outputs for each phase in a typical system development life cycle. The main tasks of the Requirement Analysis Phase are the determination of the scope of the project, making a Gantt chart, identification of problems and requirements, and planning of the development process. During the System Design Phase, the designing of wireframes, databases, and creation of visuals such as ERDs and DFDs are done. Implementation Phase: Outlining the development of the system modules, integration of these modules, and establishment of database connections. Testing Phase: This involves testing the system, users' evaluation, and elimination of bugs. Maintenance Phase: This is the ongoing support of the system, encompassing problem fixing and updating so that the system can continue working.

4.2 System Requirement Analysis

The features and functionalities of a product are defined by its functional requirements [10]. All these features are required by the terms of the agreement and must be implemented into the system. These are stated as input to the system, operations to be carried out, and unexpected results. Unlike the less tangible non-functional criteria, these are the user-stated needs the finished product will meet. Table 3 shows the functional requirements of Pet's Village Pet Care Management System.

Table 3 *Functional Requirements of the Cat Hotel and Grooming Booking System*

No	Function	Functionalities
1.	User login	• Users need to enter the correct username and password. • An error message is displayed when the wrong username or password is entered.
2.	User logout	Users can log out their account
3.	User register	• Customers can create new accounts. • An error message will display when the input format is incorrect.
4.	Service Reservation	Customers can make an appointment for a service by selecting an available time
5.	User profile management	Customers can manage their profiles by editing their personal information
6.	Service management	Administrator can manage the service information by updating the current service information
7.	System Management	Administrator can manage the system by updating the site information

Besides that, non-functional requirements are a description of the general attributes of a system. So, these are the minimum requirements for system quality that the project contract mandates. Table 4 shows the non-functional requirement of Cat Hotel and Grooming Booking System.

Table 4 *Non-Functional Requirements of the Cat Hotel and Grooming Booking System*

No	Requirement	Description
1	Security	• Encrypt the password entered by the user. • Encrypt the password saved in database. • OTP (One-Time Password) for additional verification. • The user can only log into the system when the username and password are correct.
2	Usability	• Provide a user-friendly interface for the user. • This system should be easy to use and understand.
3	Availability	This system should be available and accessible 24 hours a day

The system should implement strict security measures, such as password encryption and secure authentication protocols, to safeguard user data. The system should be designed with usability in mind, featuring a user-friendly interface that is intuitive and easy to navigate for all users. The system should be highly available and accessible 24/7 to ensure seamless service delivery. These are essential non-functional requirements for a successful, reliable booking system that can meet the needs of both customers and administrators.

4.3 System Analysis

A context diagram illustrates the interactions between external entities and an internal software system. Its primary purpose is to help businesses comprehend the system's scope. Fig. 2 shows the context diagram of Cat Hotel and Grooming Booking System.

The Level 0 Data Flow Diagram (Level 0 DFD) outlines the order of the steps in the process based on their level. The purpose of this diagram is to clearly display the connections between all the primary operations taking place within the system, as well as the information flowing between them. For any given business process model, there will be exactly one DFD at level 0. The following elements, such as process, data flow, data store, and external entities, together with their respective functions, can be used to illustrate the functioning of a level 0 DFD Diagram. There are a total of five processes in the DFD level 0 diagram, which are register (process 1.0), login (process 2.0), manage pet owner details (process 3.0), manage service (process 4.0), booking detail, (process 5.0) and manage room details (process 6.0). Fig. 3 shows the DFD level 0 of the Cat Hotel and Grooming Booking System.

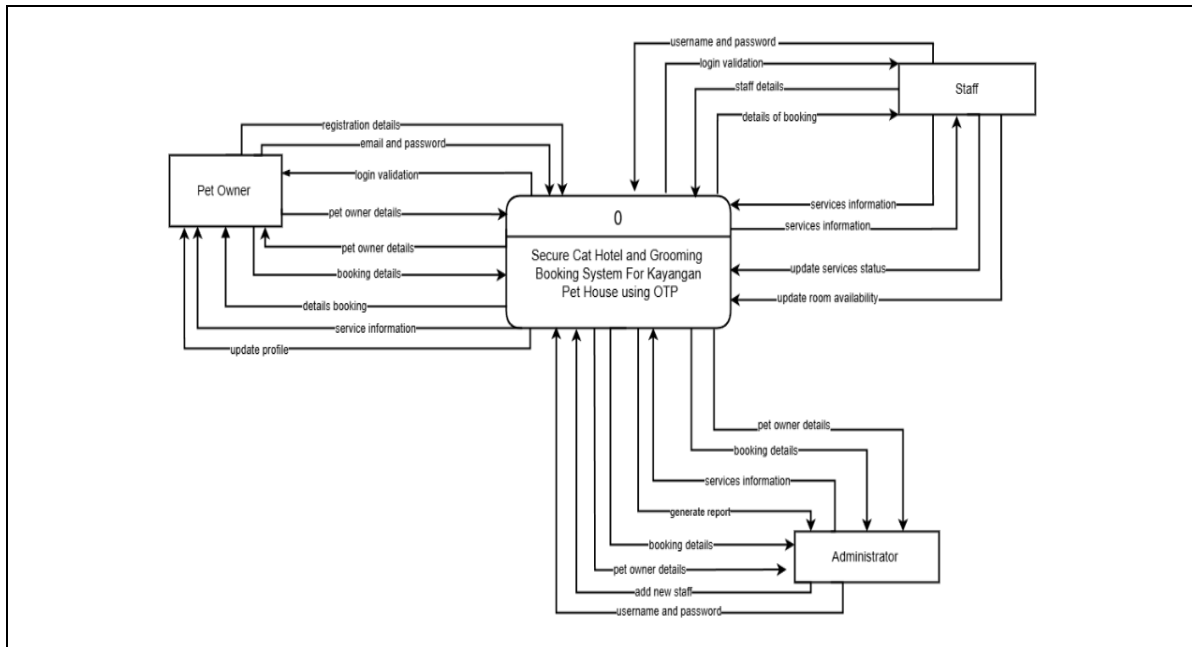


Fig. 2 Context diagram of Cat Hotel and Grooming Booking System

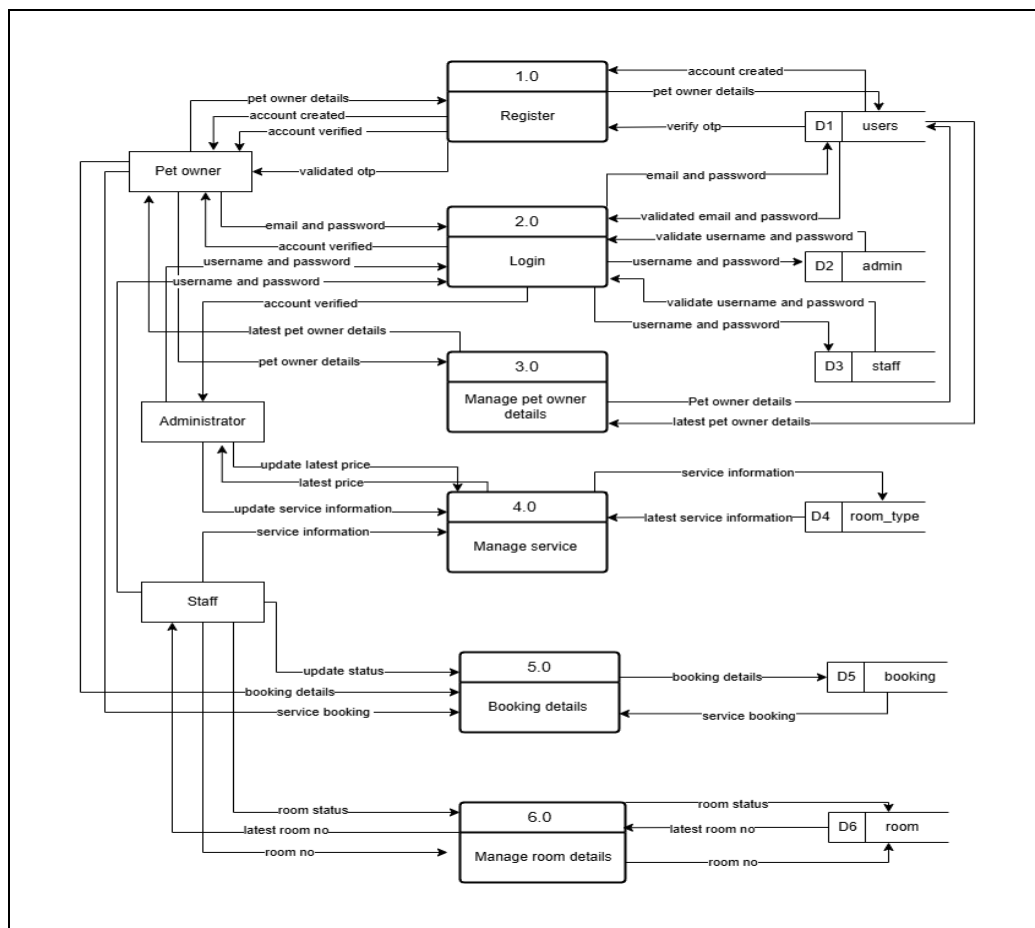


Fig. 3 DFD Level 0 of Cat Hotel and Grooming Booking System

The Entity Relationship (ER) Diagrams are flowcharts that depict the relationships between entities within a system. ER Diagrams are typically used to build or troubleshoot relational databases in business information systems, education, and software engineering. Fig. 4 shows the ERD of the Cat Hotel and Grooming Booking System.

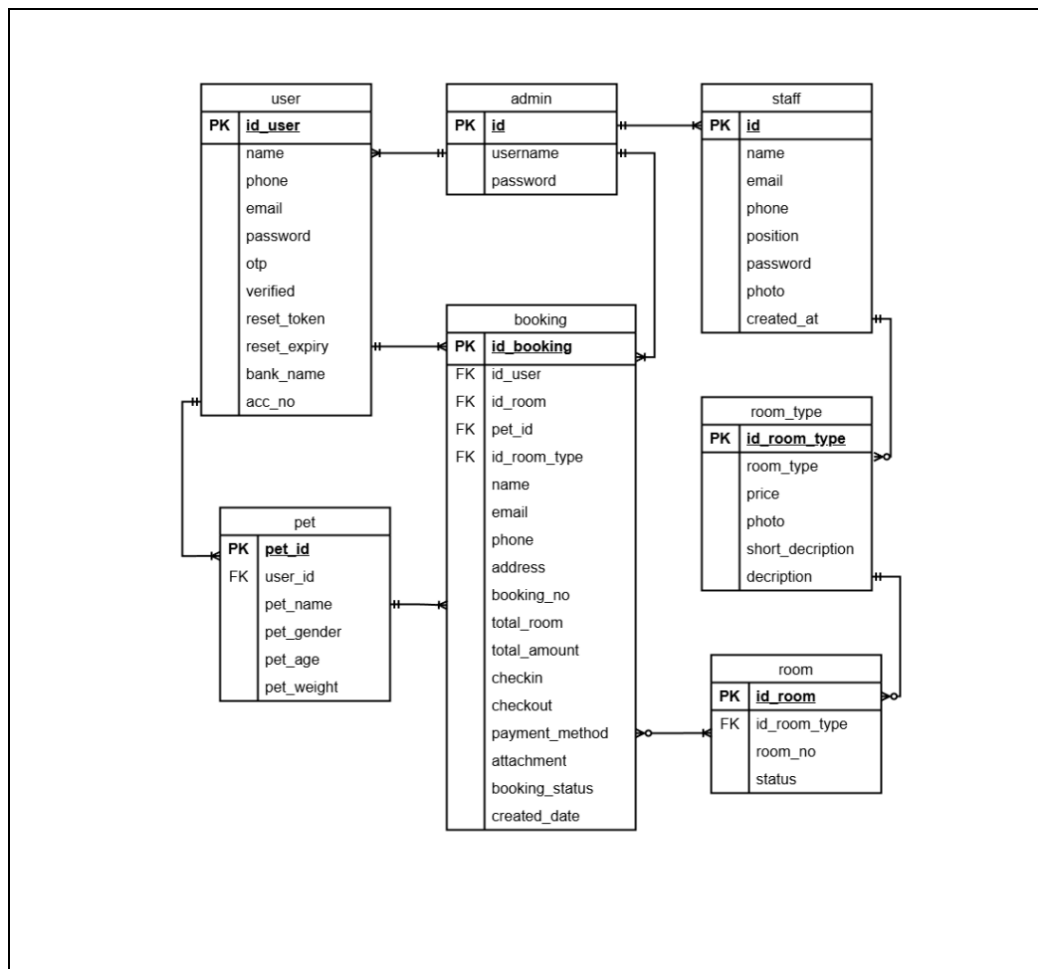


Fig. 4 ERD of Cat Hotel and Grooming Booking System

A flowchart is simply a graphical representation of steps. In this section, the flow chart which consists of flow chart for admin, staff and pet owner are outlined. The Flow chart for administrator is discussed in Fig. 5(a) and for staff in Fig. 5(b).

Fig. 5(a) the flow chart shows the interaction of the administrator with the system functionality involved in the booking system. It starts with the administrator input for both username and password. After a successful login, it displays the homepage. The administrator can then view all the users, view bookings, edit services, or add new services. At the end, the administrator can end the session. This flowchart describes the decision and action of a administrator in the system. Fig. 5(b), this flowchart depicts the basic staff interaction with the system, probably some booking management system.

Fig. 6, This flow chart describes the series of events a user goes through in a booking system: The user first checks whether they already have an account. If not, they will register and verify their details by getting a email that has six-digit OTP. After successful login, users land on the homepage, where they can choose to update their details, view bookings, or select services. In choosing services, there are a series of steps involved to give the user options to arrive at their choice. Once the service is chosen, users can proceed to a booking and payment for it. The session ends.

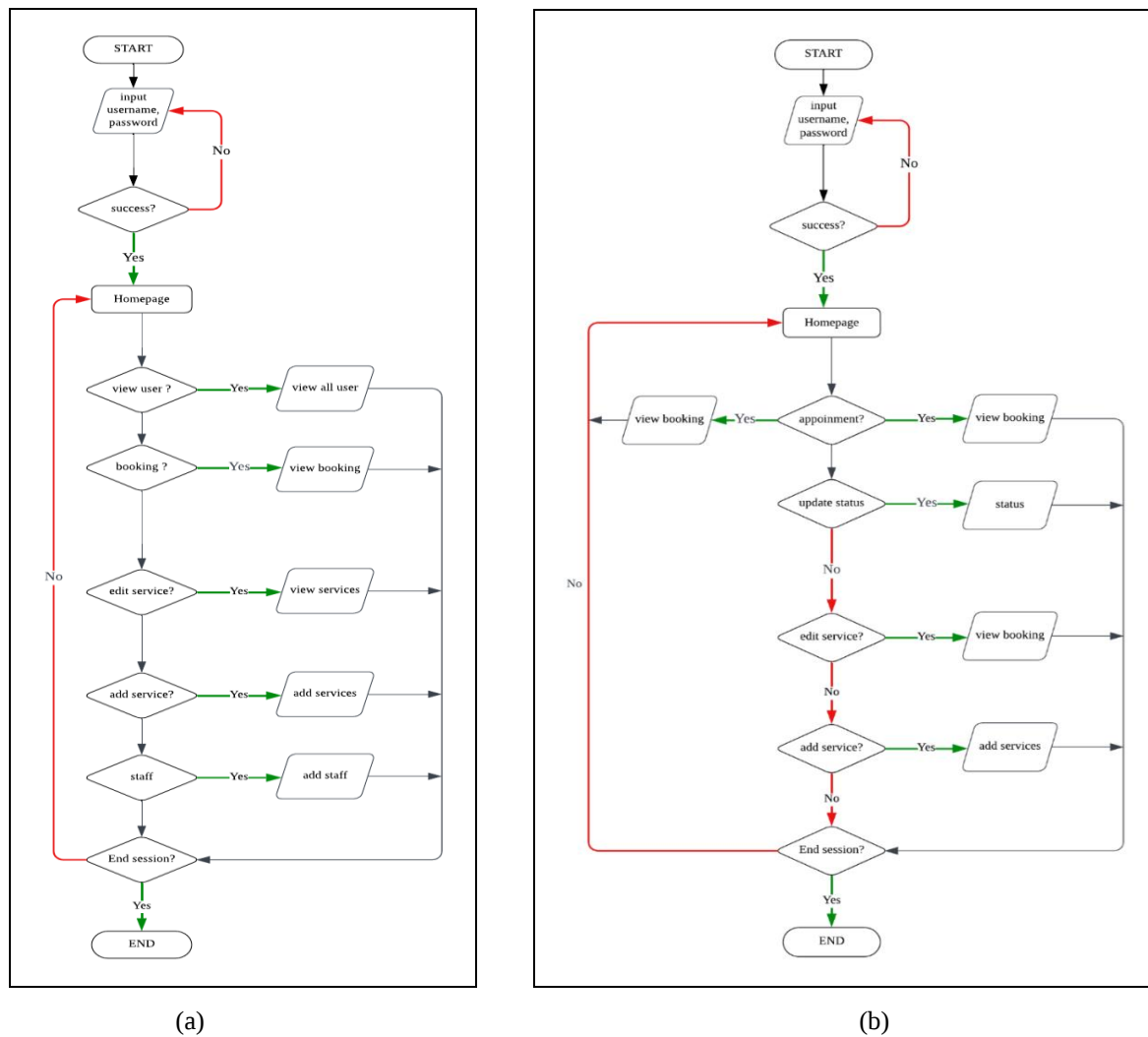


Fig. 5 Flow chart (a) Administrator; (b) Staff

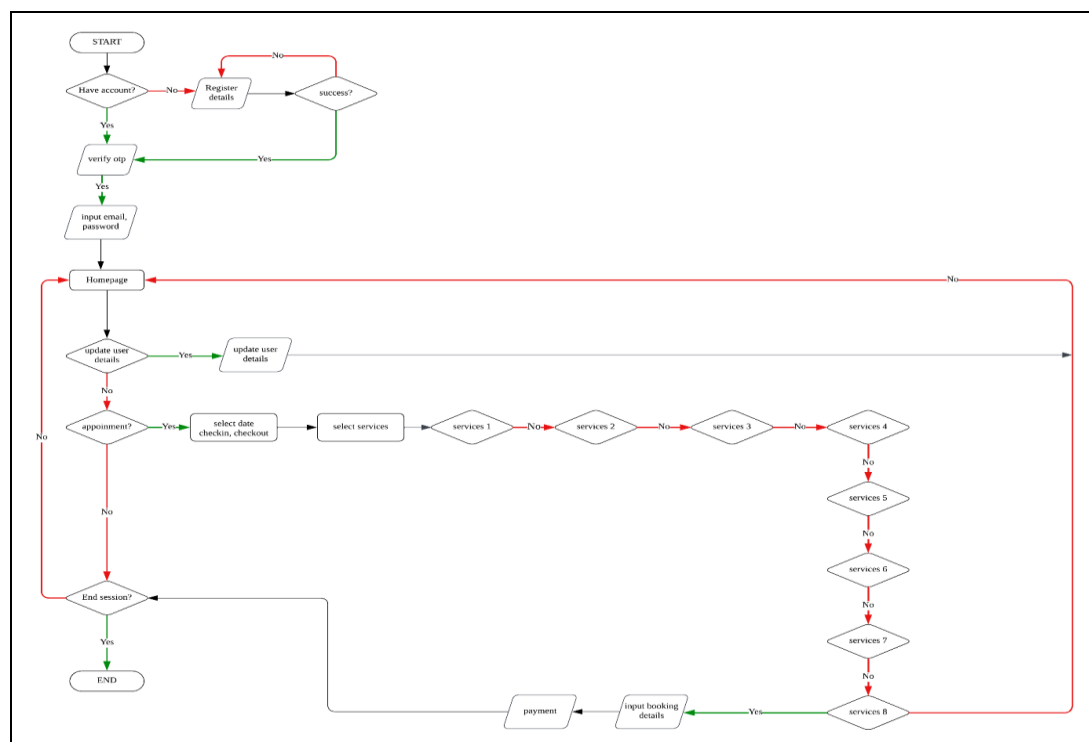


Fig. 6 Flow chart for Pet Owner

4.4 System Design

User interface (UI) design is how designers create interface in software or electronic devices emphasizing aesthetics or style [11]. Designers strive to design interfaces that are both easy to use and enjoyable for users. UI design encompasses graphical user interfaces as well as other forms.

Fig. 7(a) shows the login interface. The login module allows the customer to log in to their account. To log in successfully, the admin and customer require to provide the correct email and password. If the email or password provided by the users is wrong, they will fail to access their profile and reserve service. However, if the email provided by the user can be found in the database but the user has not verified their email when they create an account, a message "It looks like you have not still verified your email" will display to notify the user to verify their email is a must. Lastly, when the email and password input by the user is correct, they can access their profile and can make a reservation for service.

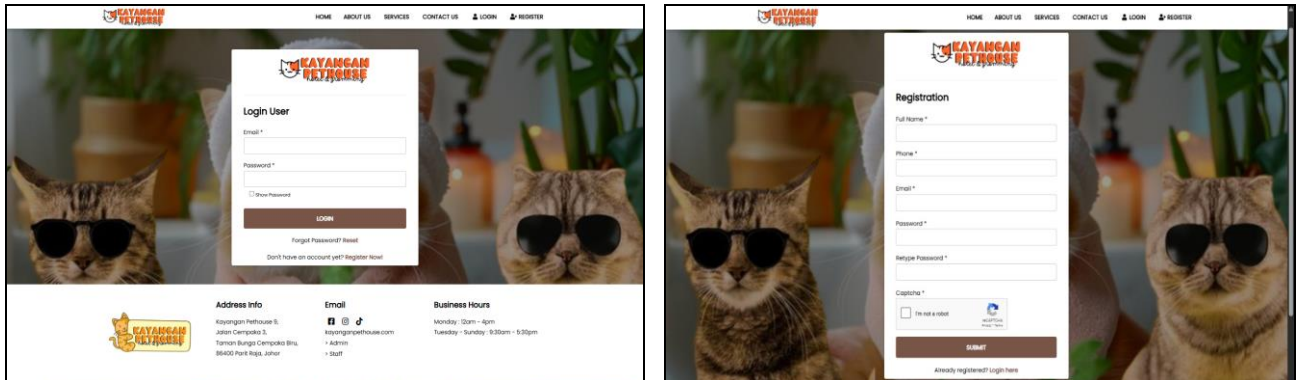


Fig. 7 Interface (a) Login; (b) Register

Fig. 7(b) shows the interface about the register module allows a new customer to create an account. To create a new account, the customer is required to provide their full name, phone, email, password and confirm password, and click "submit" button. Client-side form validation will be conducted after the sign-up button is clicked to make sure that the input has match the requirement such as the username field can only accept the alphabet, the email required to match the email format, the password must contain at least one number, uppercase, and lowercase and at least 8 or more character, the confirm password must match with the password input and the captcha must be ticked. If one of the input fields does not match the requirement, the form fails to submit, and the error message will be displayed.

If the inputs match the requirement, a verification code will be sent to the email that the customer registered, and the user will be redirected to the code verification page to enter the 6-digit number that they have received in the email. If the verification code that the user enters is correct, the account is created successfully but if the user enters the wrong verification code or skips this process, their account will fail to create. Fig. 8(a) shows the interface of OTP verify interface.

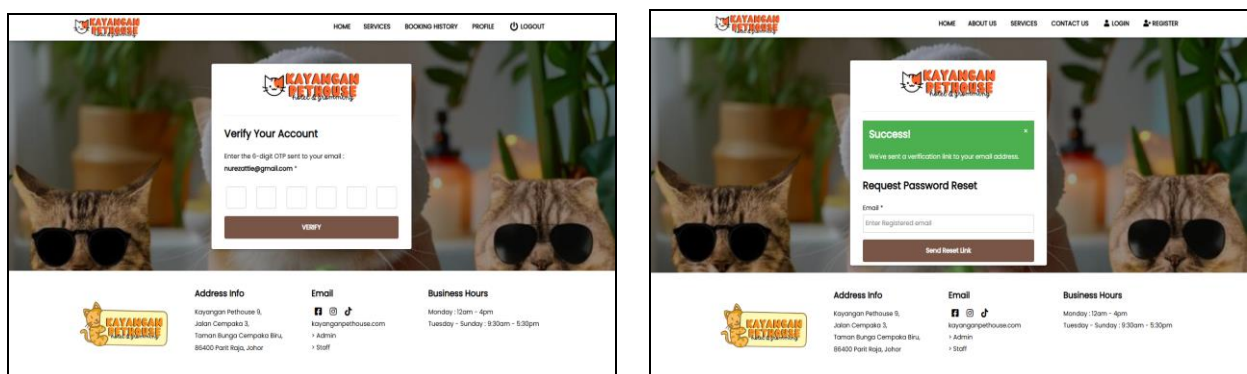


Fig. 8 Interface (a) Verify OTP; (b) Password Reset

Fig. 8(b) shows the interface of the reset password interface. The form is on the "Request Password Reset" page of the Kayangan Pet House website. There is a successful message showing that a verification link was sent to the user's registered email address. Below the successful message, there's also a form where the user can enter their email and make the password reset request.

This is the "Our Services" page of the Kayangan Pet House website, displaying their services in a visually attractive grid of service cards; each card includes an image that reflects the service provided, such as an adult

cat in a comfortable setting for "Boarding Adult" or a before-and-after photo for "Grooming Short Hair." Key information such as the service name, a brief description, and the daily rate is displayed next to each card. A "DETAIL" button prompts users to go into the details of each service, hence making it easy for the customer to understand and choose those services which would suit their needs. Fig. 9(a) shows the interface of services interface.

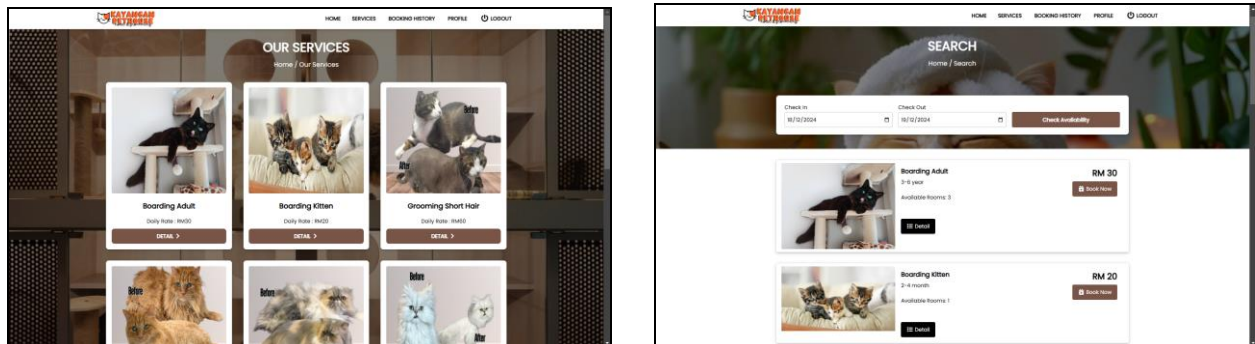


Fig. 9 Interface (a) Services; (b) Search

Fig. 9(b) is the Search Results Page of the Kayangan Pet House website. Users could input the desired check-in/check-out dates and then click the "Check Availability" icon to see the available service and room type. Results would then be displayed in a user-friendly manner with an image representing each service, a short description of each service offered, a daily rate, the target age of the cats, the number of available rooms, and a "Book Now" feature for convenient reservation. By using this page, it is clear what services the users require and how to find them, hence book them conveniently.

Fig. 10(a) shown a page message shows up as "Completed" and allows the user to have his booking number, which is 25137. Summary of details of booking will pop up, showing service booked, date of check-in and check-out, number of rooms booked in total, customer's contact information. It confirms that the booking has been made and gives essential details to the user for reference.

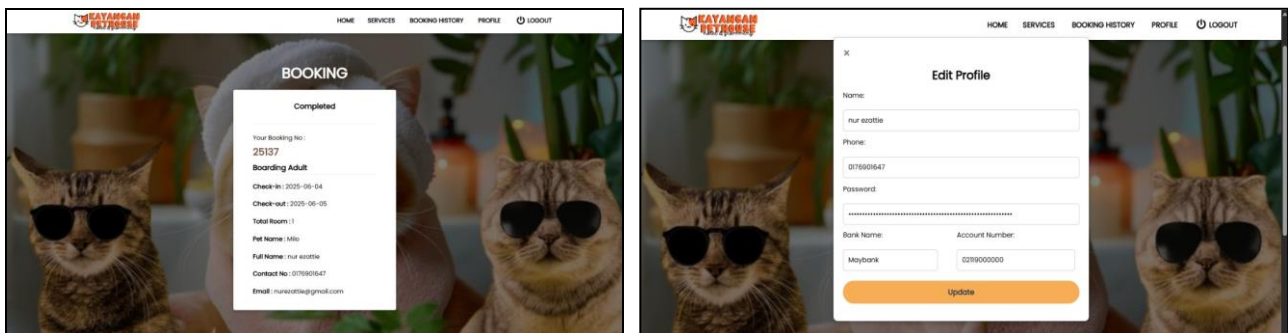


Fig. 10 Interface (a) Booking; (b) Profile

Fig. 10(b) shows the interface of the profile interface. The user profile management module allows users to manage their profile by editing their personal information. The user's personal information includes full name, contact no, and email. The user can edit their information by clicking the profile button that is displayed on the user navbar. When the user clicks on the edit button, a modal will display for the user to edit their information.

5. Implementation and Testing

For the part in the implementation and testing is to discuss and see if the system will run smoothly and each of the buttons will when to the correct page when click it. It will also test the system functionality to see if enter a wrong input, an error message will generate or not. So, if there is any problem when running the system, a solution can be found to uphold the problem.

5.1 System Implementation

This Cat Hotel and Grooming Booking System was built using php language. Php language is the mainly used to develop this system and CSS for designing the interfaces and layouts of the system for every module. MySQL and PhpMyAdmin also uses to connect with the database of the proposed system.

5.1.1 Database Configuration

To store and connect all the data of this Cat Hotel and Grooming Booking System with the database MySQL and PhpMyAdmin were needed. However, to run the system the XAMPP was needed because it will connect the proposed system with the database. Fig. 11 (Appendix A) shows the MySQL database connection of the proposed system.

5.1.2 Login Module

This module will be used for three users, which are an administrator, staff of the Kayangan Pet House and the pet owner of the pet house. All users need to fill in the correct email and password that already had been registered. Fig. 12 (Appendix A) shows the interface for the login module for pet owners and Fig. 13 (Appendix A) shows the code segment of login process.

5.1.3 Registration Module

Fig. 14 (Appendix A) shows the interfaces for registration module which is the first module for this proposed system. This module is necessary for the new user because pet owners need to register for the first time before login to the system. Customers need to fill in some details such as their full name, contact number, email and the important one, which is the password for login purposes. After the pet owners finishes filling in all the data, the data will be stored to the PhpMyAdmin for future use. Fig. 15 (Appendix A) shows the code segment for the register process.

5.1.4 Service Management Module

This module is designed for the administrator to manage and update the treatment packages provided by the Pet House. The admin can add, edit, or remove available services to ensure the information remains accurate and up to date. Although the services listed in the services are basic treatments for pets, pet owners can later view these services before making a booking based on the available options. This module is accessible only to administrators for management purposes. Fig. 16 (Appendix A) shows the interface for service management module and Fig. 17 (Appendix A) shows the code segment for services editing process.

5.1.5 Profile Module

This module allows pet owners to manage their personal profiles by editing and updating their information. The personal details include full name, contact number, email address, bank name, account number, and password. Fig. 18 (Appendix A) shows the interface for pet owner profile and Fig. 19 (Appendix A) shows the code segment for updating the user profile details. In addition, pet owners can also manage their pet details. Within this section, pet owners can add new pets, view existing pet records, and delete pet information. Fig. 20 (Appendix A) shows the interface for pet profile and Fig. 21 (Appendix A) shows the code segment for adding new pet.

5.1.6 Booking Module

This module allows pet owners to book a service by selecting an available date, the service name and category auto-fill based on the selected service, and the price is calculated accordingly. At the booking form pet owners need to select a pet and select a payment method. Fig. 22 (Appendix A) shows the interface for booking form and Fig. 23 (Appendix A) shows the code segment for storing the booking services.

5.1.7 Staff Module

This module allows the admin to manage staff using Create, Read, Update, and Delete (CRUD) operations. Admins can add staff by filling in details like name, email, contact, position, and photo. A table displays all staff information, with action buttons to view, edit, or delete. All actions are open in a modal. When deleting, a confirmation prompt ensures the action before removing the staff from the list and database. Fig. 24 (Appendix A) shows the interface for staff module and Fig. 25 (Appendix A) shows the code segment for staff adding process.

5.2 System Functionality Testing

System testing aims to evaluate the overall performance and functionality of a comprehensive and completely integrated software solution. This testing is performed after integration testing and before user acceptance testing. Besides that, it determines whether the system fulfils the specified requirements and is fit for distribution to end users. Therefore, testing is included in the project's objective and scope from the outset.

Table 5 *Test category of the system*

Test Category	Description
1	The functionality of the system that tests by administrator
2	The functionality of the system that tests by the pet owners
3	The functionality of the system that tests by the staff

Table 6 *Test plan of the system*

Modules	Test Category	Description	Expected Result	Actual Result
Log in	1,2,3	Log in to the system: • Insert email • Insert password • Click login button	User can only log in to the account if the input is valid	Pass
Logout	1,2,3	Logout form the system: • Click the logout button	User can logout from their account when they click on the logout button	Pass
Register	1,2	Register as new user: • Fill in the register form	Data is stored in database after clicking the register button and a anew account is created successfully	Pass
Staff	1	Manage staff list by: • Add new staff • View staff details • Update staff details • Delete Staff	1. New staff details are stored in the database after clicking on the add button and updating them in the staff list. 2. The staff details can be viewed. 3. The existing staff details can be updated by inserting new details and updating them in the database. 4. Confirmation message will pop up after clicking the delete button. 5. The staff details are deleted from the list after the "OK" option in the confirmation message is clicked.	Pass
Service	1,3	Manage service list by: • Update the service details. • View service details	1. The existing service details can be updated by inserting new details and update them in database. 2. The service details can be clearly viewed	Pass
User and cat profile	2	Manage the user profile by: - Update user personal information - Add new cat details	1. Users can update their personal information by inserting the new details and the database will be updated after the save button is clicked. 2. User can add a new cat detail	Pass
Booking	2	Booking service: - Select a service - Fill in the form - Confirm booking	1. The service name and category auto display according to user selection. 2. Available room and date are displayed to select. 3. The price is calculated correctly according to the selected service. 4. Booking details are stored in database after customers confirm the booking.	Pass
Booking Record	1,2,3	View the booking record	1. The administrator can view all the customer booking records. 2. Customer can view their booking record	Pass
Manage Room	1,3	Add available room	1. The administrator and staff can update the room that available to the system	Pass

All modules of the Cat Hotel and Grooming Booking System have passed the testing phase successfully, as shown in Table 6. Each function from user registration and login to booking services, managing staff, and updating user profiles performed as expected, with data correctly stored, updated, and displayed in the system. The test results confirm that the system is stable, reliable, and meets the intended functional requirements.

The positive outcomes of the testing phase demonstrate that the system is ready for deployment and can be used effectively by administrators, staff, and pet owners. Any actions performed, such as adding new records, viewing details, and managing data, return the correct results without errors. Overall, the test results validate that the Cat Hotel and Grooming Booking System is functional and provides a smooth user experience.

6. Results and Discussion

The Cat Hotel and Grooming Booking System for Kayangan Pet House has successfully fulfilled its objectives by transforming the manual record-keeping process into a digital system. This shift has significantly reduced the workload of staff, as customers can now independently register their pets, view available services, and make bookings through a user-friendly interface. The booking process is simple and can be completed within minutes by selecting the service, choosing a pet, picking an available date, and selecting a payment method. Customers can also view their booking records anytime without the need to contact staff. Additionally, the system includes essential security features such as password hashing and email verification using a 6-digit OTP to ensure the authenticity of user accounts. Administrators and staff can easily manage bookings and staff profiles, while customers can manage their personal and pet information, creating a seamless experience for all users.

Despite its advantages, the system has some limitations, such as only being accessible via a web browser and lacking a reminder feature to notify customers of upcoming bookings. To enhance its functionality, several improvements are proposed for future development. These include the implementation of a notification system to alert users about booking dates, promotions, or announcements, as well as a live chat feature to facilitate direct communication between customers and administrators. Moreover, developing a mobile application version of the system with OTP-secured login would further increase accessibility and convenience, allowing users to make bookings and manage information from any device at any time. In conclusion, while the current system effectively supports the needs of Kayangan Pet House, these enhancements could significantly improve the overall user experience and system performance in the future.

Acknowledgement

The authors would like to thank the Faculty of Computer Science and Information Technology, Universiti Tun Hussein Onn Malaysia for its support.

Conflict of Interest

Authors declare that there is no conflict of interest regarding the publication of the paper.

Author Contribution

This journal requires that all authors take public responsibility for the content of the work submitted for review. The contributions of all authors must be described in the following manner:

*The authors confirm contribution to the paper as follows: **study conception and design:** N. E. Suhaime, N. Rahim; **data collection:** N. E. Suhaime, N. Rahim; **analysis and interpretation of results:** N. E. Suhaime, N. Rahim; **draft manuscript preparation:** N. E. Suhaime, N. Rahim. All authors reviewed the results and approved the final version of the manuscript.*

References

- [1] Freist, R. (2024b, August 12). Is your PC slowing down? Here's how to find the problem. PCWorld. <https://www.pcworld.com/article/1936066/your-pc-is-getting-slow-how-to-find-the-cause.html>
- [2] Shackleton, T. (2023, August 10). The Financial Impact of a Data Breach in 2022. Six Degrees. <https://www.6dg.co.uk/blog/financial-impact-of-a-data-breach-2022/>
- [3] Pet, T. (2020). Book your appointment with talisay pet care center - pet services. Talisaypetcarecenter.setmore.com. <https://talisaypetcarecenter.setmore.com/>
- [4] Lovely. (2013). Dog grooming & cat grooming | deerfield beach | lovely cuts. Lovely.Cuts Pet Grooming. <https://lovelycuts.petgrooming.com/>
- [5] Boo Pet Groomers. (2020). Pet grooming & hotel | boo pet groomers. BooPetGroomers. <https://www.boopetgroomers.com/>
- [6] Deepak, N. & Kumar, S. (2015). Flexible Self-Managing Pipe-line Framework Reducing Development Risk to Improve Software Quality. International Journal of Information Technology and Computer Science, 7(7). 35-47. <https://doi.org/10.5815/ijitcs.2015.07.05>
- [7] SDLC - Waterfall Model. (2019). https://www.tutorialspoint.com/sdlc/sdlc_waterfall_model.htm
- [8] Osman, M. (2020). Prototype vs. Wireframe: How to Use Both in Design. Blog.hubspot.com. <https://blog.hubspot.com/website/prototype-vs-wireframe>
- [9] Khillar, S. (2021). Difference Between Wireframe and Prototype. Difference between Similar Terms and Objects. <http://www.differencebetween.net/technology/difference-between-wireframe-and-prototype>
- [10] Altexsoft. (2021). Functional and nonfunctional requirements: Specification and types. AltexSoft. <https://www.altexsoft.com/blog/business/functional-and-non-functional-requirements-specification-and-types/>
- [11] Visual Paradigm. (2019). *Requirement analysis techniques*. Visual-Paradigm.com. <https://www.visual-paradigm.com/guide/requirements-gathering/requirement-analysis-techniques/>
- [12] Tejaswini, B. (2017). A survey of password attacks and safe hashing algorithms. International Research Journal of Engineering and Technology (IRJET). <https://www.irjet.net/archives/V4/i12/IRJETV4112284.pdf>
- [13] "2023 AVMA annual report," American Veterinary Medical Association. <https://www.avma.org/about/annual-report/2023-avma-annual-report>
- [14] Prabhu, R. (2018). SHA-1 Hash. GeeksforGeeks. <https://www.geeksforgeeks.org/sha-1-hash-in-java/>
- [15] SDLC - Waterfall Model. (2019). https://www.tutorialspoint.com/sdlc/sdlc_waterfall_model.htm
- [16] Interaction Design Foundation. (2018). *What is user interface (UI) design?* The Interaction Design Foundation; UX courses. <https://www.interaction-design.org/literature/topics/ui-design>
- [17] Peterson, R. (2019). Database design tutorial: Learn data modeling. Guru99.com. <https://www.guru99.com/database-design.html>

Appendix A

```
//localhost
$dbHost = "localhost"; // Database host
$dbName = "kayanganpethouse_v3"; // Database name
$dbUser = "root"; // Database user
$dbPass = ""; // Database password

$con = mysqli_connect($dbHost,$dbUser,$dbPass,$dbName);
```

Fig. 11 Database Connection Code Segment

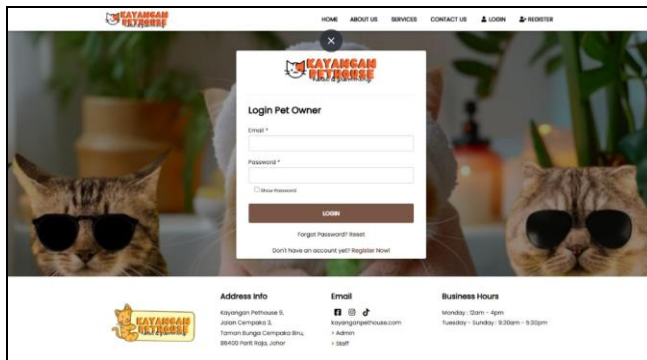


Fig. 12 Interface for the login module for pet owners

```
12 if ($act == "login") {
13     $email = (isset($_POST['email'])) ? trim($_POST['email']) : '';
14     $password = (isset($_POST['password'])) ? trim($_POST['password']) : '';
15
16     // Prepare a statement to prevent SQL injection
17     $stmt = $con->prepare("SELECT id_user, password FROM 'user' WHERE 'email' = ?");
18     $stmt->bind_param("s", $email);
19     $stmt->execute();
20     $result = $stmt->get_result();
21     $data = $result->fetch_assoc();
22
23     // Debugging output
24     if ($data) {
25         echo "User found. ";
26         if (password_verify($password, $data['password'])) {
27             echo "Password is correct. ";
28             $_SESSION['email'] = $email;
29             $_SESSION['id_user'] = $data['id_user'];
30             header("Location: search.php");
31             exit();
32         } else {
33             echo "Password is incorrect. ";
34         }
35     } else {
36         echo "User not found. ";
37     }
38 }
```

Fig. 13 Code segment of login process

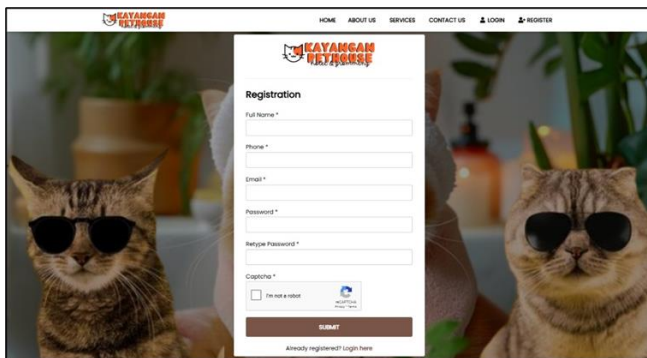


Fig. 14 Interface for the registration module

```
12 if ($act == "login") {
13     $email = (isset($_POST['email'])) ? trim($_POST['email']) : '';
14     $password = (isset($_POST['password'])) ? trim($_POST['password']) : '';
15
16     // Prepare a statement to prevent SQL injection
17     $stmt = $con->prepare("SELECT id_user, password FROM 'user' WHERE 'email' = ?");
18     $stmt->bind_param("s", $email);
19     $stmt->execute();
20     $result = $stmt->get_result();
21     $data = $result->fetch_assoc();
22
23     // Debugging output
24     if ($data) {
25         echo "User found. ";
26         if (password_verify($password, $data['password'])) {
27             echo "Password is correct. ";
28             $_SESSION['email'] = $email;
29             $_SESSION['id_user'] = $data['id_user'];
30             header("Location: search.php");
31             exit();
32         } else {
33             echo "Password is incorrect. ";
34         }
35     } else {
36         echo "User not found. ";
37     }
38 }
```

Fig. 15 Code segment of register process

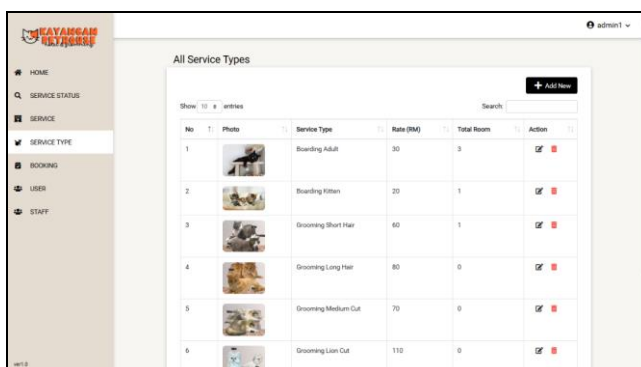


Fig. 16 Interface for services management module

```
72 if ($act == "edit") {
73     $SQL_update = "
74     UPDATE
75     'room_type'
76     SET
77     'room_type' = 'room_type',
78     'price' = 'price',
79     'short_description' = 'short_description',
80     'description' = 'description'
81     WHERE
82     id_room_type = $id_room_type
83     ";
84
85     $result = mysqli_query($con, $SQL_update);
86
87     // ----- Photo -----
88     if (isset($_FILES['photo'])) {
89         if ($_FILES['photo']['error'] == 0) && (isset($_FILES['photo'])) {
90
91             $file_name = $_FILES['photo']['name'];
92             $file_size = $_FILES['photo']['size'];
93             $file_tmp = $_FILES['photo']['tmp_name'];
94             $file_type = $_FILES['photo']['type'];
95
96             $fileNameCaps = explode(".", $file_name);
97             $file_ext = strtolower(end($fileNameCaps));
98
99             if (empty($errors) == true) {
100                 move_uploaded_file($file_tmp, "upload/" . $file_name);
101             }
102         }
103     }
```

Fig. 17 Code segment of services editing process

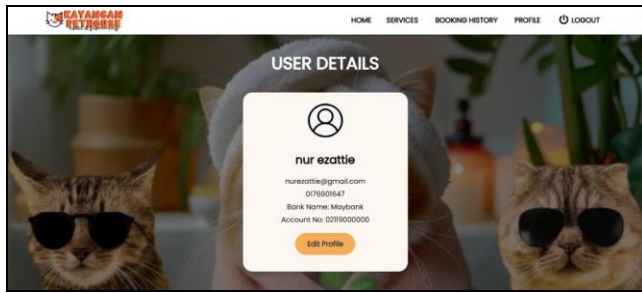


Fig. 18 Interface for pet owner profile

```

9  if ($_SERVER['REQUEST_METHOD'] === 'POST') {
10 // Update user details
11  if (isset($_POST['act']) && $_POST['act'] == "edit") {
12      $name = mysqli_real_escape_string($con, trim($_POST['name']));
13      $phone = trim($_POST['phone']);
14      $password = trim($_POST['password']);
15      $bank_name = mysqli_real_escape_string($con, trim($_POST['bank_name']));
16      $acc_no = mysqli_real_escape_string($con, trim($_POST['acc_no']));
17
18      $SQL_update = "UPDATE user SET
19          name = '$name',
20          phone = '$phone',
21          password = '$password',
22          bank_name = '$bank_name',
23          acc_no = '$acc_no'
24          WHERE email = '$email'";
25      mysqli_query($con, $SQL_update) or die("Error: " . mysqli_error($con));
26      $success = "User details updated!";
27  }

```

Fig. 19 Code segment of updating user profile

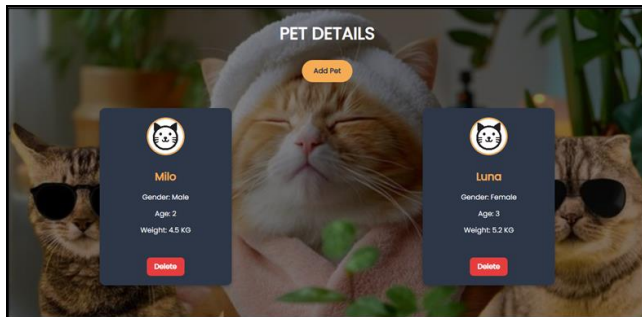


Fig. 20 Interface for pet profile

```

29 // Add pet
30
31 // Add pet
32 if (isset($_POST['act']) && $_POST['act'] == "add_pet") {
33     $pet_name = mysqli_real_escape_string($con, $_POST['pet_name']);
34     $pet_gender = $_POST['pet_gender'];
35     $pet_age = $_POST['pet_age'];
36     $pet_weight = $_POST['pet_weight'];
37
38     $query = "INSERT INTO pet (user_id, pet_name, pet_gender, pet_age, pet_weight)
39         VALUES ('$id_user', '$pet_name', '$pet_gender', '$pet_age', '$pet_weight')";
40     mysqli_query($con, $query) or die("Error: " . mysqli_error($con));
41     $success = "New pet added!";
42 }
43
44
45
46
47

```

Fig. 21 Code segment of adding new pet

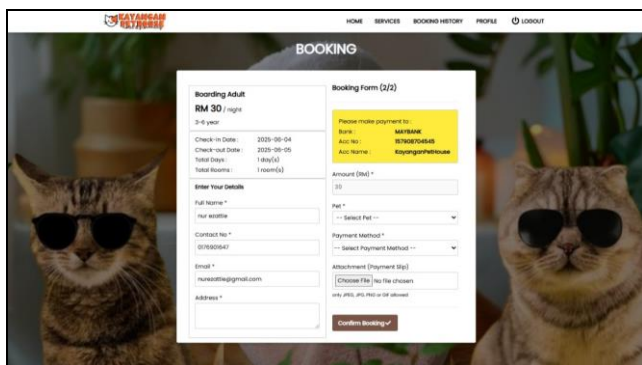


Fig. 22 Interface for booking form

```

13 // Get user's pets
14 $sql_pet = "SELECT pet_id, pet_name FROM pet WHERE user_id = '$id_user'";
15 $res_pet = mysqli_query($con, $sql_pet);
16 if (!$res_pet) {
17     die("Error in SQL query: " . mysqli_error($con));
18 }
19 $pets = [];
20 while ($row = mysqli_fetch_assoc($res_pet)) {
21     $pets[] = $row;
22 }
23
24 // Fetch form data
25 $act = $_POST['act'] ?? '';
26 $id_room_type = $_POST['id_room_type'] ?? '';
27 $checkin = $_POST['checkin'] ?? '';
28 $checkout = $_POST['checkout'] ?? '';
29 $total_room = $_POST['total_room'] ?? '1';
30
31 $name = $_POST['name'] ?? '';
32 $phone = $_POST['phone'] ?? '';
33 $email = $_POST['email'] ?? '';
34 $address = $_POST['address'] ?? '';
35 $pet_id = $_POST['id_pet'] ?? null;
36
37 $payment_method = $_POST['payment_method'] ?? '';
38 $booking_status = "Unpaid";
39
40 $name = mysqli_real_escape_string($con, $name);
41 $address = mysqli_real_escape_string($con, $address);

```

Fig. 23 Code segment of storing services booking

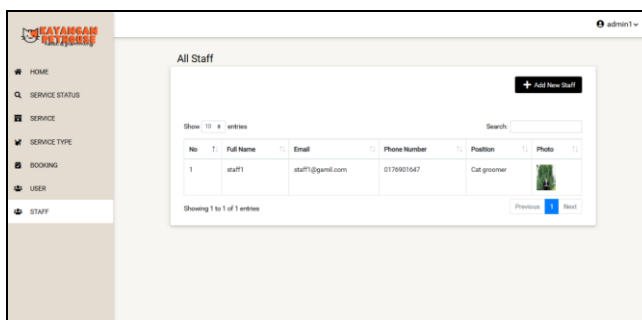


Fig. 24 Interface for staff module

```

24 if ($_SERVER['REQUEST_METHOD'] === 'POST' && isset($_POST['act']) && $_POST['act'] == "register") {
25     $name = mysqli_real_escape_string($con, $_POST['name']);
26     $email = mysqli_real_escape_string($con, $_POST['email']);
27     $phone = mysqli_real_escape_string($con, $_POST['phone']);
28     $position = mysqli_real_escape_string($con, $_POST['position']);
29     $password = password_hash($_POST['password'], PASSWORD_DEFAULT);
30
31 // Handle image upload
32 $allowed_types = ['jpg', 'jpeg', 'png'];
33 $photo = $_FILES['photo'];
34 $temp_name = $photo['tmp_name'];
35 $photo_ext = strtolower(pathinfo($photo, PATHINFO_EXTENSION));
36
37 if (in_array($photo_ext, $allowed_types)) {
38     $new_photo_name = uniqid() . $photo_ext;
39     $upload_path = "uploads/" . $new_photo_name;
40
41 if (move_uploaded_file($temp_name, $upload_path)) {
42     // Insert into database
43     $query = "INSERT INTO staff (name, email, phone, position, password, $new_photo_name)
44         VALUES ('$name', '$email', '$phone', '$position', '$password', '$new_photo_name')";
45     if (mysqli_query($con, $query)) {
46         echo "ScriptAlert('Staff added successfully!'); window.location='a-staff.php';/script";
47     } else {
48         echo "ScriptAlert('Database error: " . mysqli_error($con) . "');/script";
49     }
50 } else {
51     echo "ScriptAlert('Failed to upload photo.')./script";
52 }
53 }

```

Fig. 25 Code segment of staff adding process