CHAPTER 3

RESEARCH METHODOLOGY

This chapter presents the sequence of project development. It includes the discussion of methods and course of actions taken to develop the project, the collation of related information and development procedures, and the definition of a number of users on how the project was evaluated by the users and developers.

Research Design

The researchers used descriptive research methodology to evaluate the RaphaVets Pet Clinic: A Web-Based Management and Pet Care System Utilizing Multiple Regression. In order to assess the system's effectiveness and usability, this research design focuses on gathering data to measure its impact and overall effectiveness. This will provide us details of everyday interactions (registering, book, cancel, and use the chatbot and other features), capture the clinic's workflow changes (phone call volume, front-desk time spent on admin tasks), and measure basic technical performance (uptime, response time, errors) alongside simple accuracy checks for AI features such as the breed detector and regression predictions.

This approach is centered on describing the system's functions and analyzing user feedback to determine its performance. This descriptive methodology will be implemented for the researchers to evaluate the system through client feedback and technical testing. This will provide statistical data and analysis on the effectiveness of the software by objective measurement, which is essential for understanding user satisfaction

and system quality. Analysis will emphasize clear, simple accuracy rates for AI features, and qualitative feedback findings. Ethical safeguards include informed consent for interviews and surveys, secure storage of identifiable data, and a safety policy that frames all AI outputs as advisory only and requires clinician confirmation before any clinical action. This approach produces a human centered, realistic evaluation that caters directly to the clinic's needs and provides concrete recommendations for more improvement.

Project Development

The researchers used the Agile Software Development Life Cycle (SDLC) Methodology in developing the RaphaVets Pet Clinic system. This approach was chosen because it allowed the team to work in small, manageable steps and make improvements after every phase. Each stage of the Agile process helped ensure that the system was designed, tested, and improved based on the actual needs of the clinic and its clients.

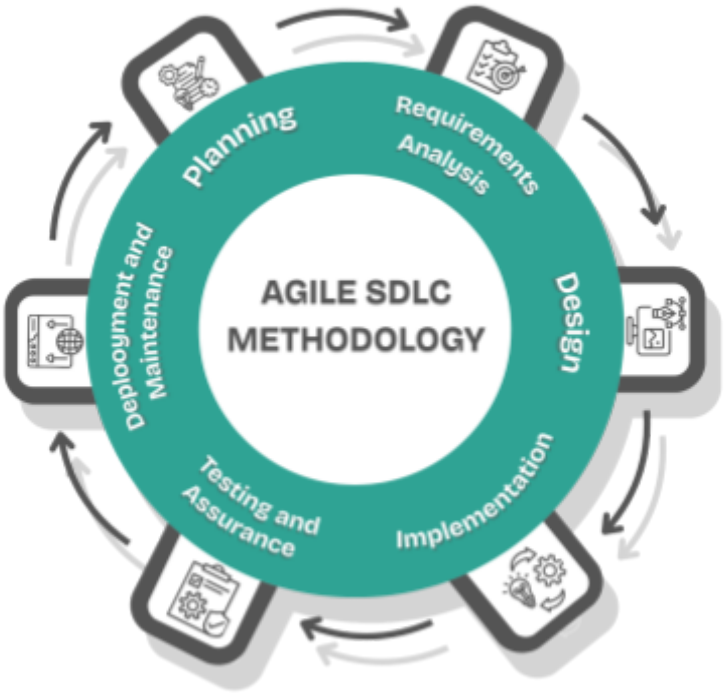

Figure 2.
Agile Diagram

Planning

In this phase, the researchers identified the main goals and features of the system. Meetings were conducted among the team members to discuss how the system could help the RaphaVets Pet Clinic improve its daily operations. The researchers also consulted the clinic staff to understand their problems with the current setup such as difficulty in organizing appointments and handling client records. From these, the team finalized the system's main features.

Requirements Analysis

After planning, the researchers gathered and analyzed all the information needed to define the system's requirements. This included identifying the types of users (admin and clients) and the functions each should be able to perform. The team studied how data would flow between features like account creation, appointment management, and report monitoring. The researchers also reviewed existing studies about veterinary systems and appointment platforms to ensure that the functions they included were both realistic and beneficial. At the end of this phase, a clear list of system requirements and technical specifications was created, which served as the basis for design and development.

Design

During this phase, the researchers focused on how the system would look and function. Using Figma, the team created a prototype that showed the layout of each page, including the home page, appointment form, chatbot interface, and pet

care tips section. The design was made to be clean and easy to navigate, using colors and icons that match the clinic's theme. The goal was to make the interface comfortable for both staff and clients, even those who are not highly familiar with online systems. The design phase also included planning the database structure to ensure smooth data management for accounts, appointments, and reports.

Implementation

The implementation phase was where the actual system creation took place. The researchers built the system using ReactJS for the frontend, NodeJs and Python for the backend, MySQL for the database, and Tailwind CSS for the design. Each feature was developed in small increments, or sprints, following the Agile Method. The first sprint focused on setting up user accounts, followed by the appointment booking module, chatbot integration, and later the breed detector and missing pet finder pages. After each sprint, the team tested the results and fixed any issues found before moving on to the next feature. This step-by-step approach helped ensure that every part of the system worked properly before combining everything.

Testing and Assurance

After implementation, the system was carefully tested to check its functionality and reliability. The researchers tested each feature, such as registering accounts, booking appointments, receiving SMS and email reminders, chatting with the AI assistant, trying the pet breed detector, and missing the pet

finder page. They also checked for errors, bugs, and interface issues. The testing was done repeatedly after every sprint, allowing early detection and correction problems. The clinic staff also tried using the system and gave feedback on its ease to use the design. This feedback was used to make adjustments and improvements to ensure the system met user expectations.

Deployment and Maintenance

Once the testing was complete, the system was deployed in a test environment to simulate actual use. The researchers observed how the clinic staff and clients interacted with the system. Minor bugs and performance issues found during this stage were fixed immediately. The team also evaluated the overall performance of the system using the ISO/IEC 25010 quality model, focusing on functionality, reliability, usability, and efficiency. After evaluation, the system was finalized and prepared for real deployment at RaphaVets Pet Clinic. This phase ensured that the system was stable, user-friendly, and ready for use in the clinic's daily operations.

Project Design

In developing the project, the researchers will undergo different activities shown in Figure 3. It identifies the analysis, resources, and procedures in designing and creating the project. The activities involved will be illustrated below:

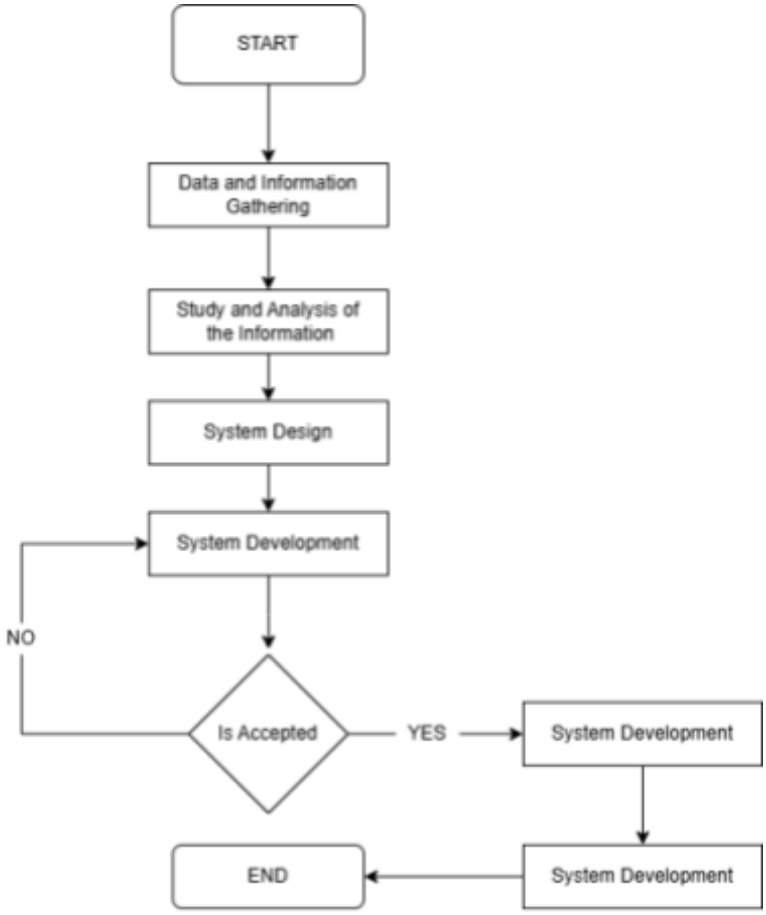

Figure 3.
Flowchart of the Project

Data Information Gathering

To make sure that the system being developed fits the needs of RaphaVets Pet Clinic, the researchers gathered information from both the staff and the clients of the clinic. The main goal of this process was to understand how the clinic currently operates, what problems they experience, and what improvements they want to see in their system.

The researchers first visited the clinic and talked to some of the staff and the doctor. Through simple interviews and observation, they found out that most of the clinic's records are done traditionally, and appointments are scheduled through walk-ins or phone calls. The staff also shared that it would be helpful if there was a system that could recognize appointments, send reminders, and help communicate easily with clients. Aside from interviews and observations, the researchers also used secondary sources such as articles, journals, and previous studies related to veterinary systems and online booking platforms. These sources helped the team learn about the features that other systems used and how they were designed.

The information gathered from all these sources as the basis for designing the RapahaVets Pet Clinic System. It guided the researchers in choosing features like online appointment booking, AI chatbot, pet care tips, breed detector, forum for missing pets, and staff monitoring tools. This process helped make sure that the system was not only useful but also practical and easy to use for both the clinic staff, doctor and the clients.

Study and Analysis of the Information

After gathering all the needed data from the clinic staff, and related studies, the researchers studied and analyzed the information to understand the main problems of RaphaVets Pet Clinic and how the proposed system could help solve them. The researchers carefully reviewed the notes from the interviews and observations to identify the most common issues and needs.

The findings showed that the clinic relied mostly on traditional methods of operation. Appointments and client information were done through Excel, and reminders were sent through phone calls or text messages. Because of this, the clinic sometimes experienced scheduling conflicts and miscommunication delays. These practices made the process time-consuming for both the staff and the clients.

Based on the analysis, the researchers concluded that the clinic needed a web-based system that could make their workflow faster, more organized, and easier to manage. The information gathered helped the researchers decide which features to include in the system. The online appointment booking was chosen as the main feature because it directly addresses the problem of scheduling and helps clients to book appointments anytime. The AI chatbot was added to assist users with basic pet concerns and common questions, reducing the staff's workload. The analysis also helped determine the most suitable tools and technologies for the system. Since the users were already familiar with basic online platforms, a web-based system using ReactJS, NodeJS, MySQL, and Tailwind CSS was found to be appropriate.

Through this study and analysis, the researchers were able to clearly define the system requirements, user roles, and features. This ensures that the proposed system would not only meet the needs of RaphaVets Pet Clinic but also be easy to use, reliable, and helpful for both staff and clients.

System Design

The system design phase serves as a blueprint for the development of the project. This stage translates the gathered requirements into a structured framework that defines how the system's components interact, process data, and deliver services to users. The design includes diagrams such as Context Diagram, Data Flow Diagrams (DFDs), Entity Relationship Diagrams (ERDs), Data Dictionary and User Story, which illustrate the system's flow of data and relationships between its users and internal processes. By creating these models, the researchers ensure that each component of the system functions coherently and aligns with the project objectives of improving veterinary service management, diagnostic accuracy, and client convenience.

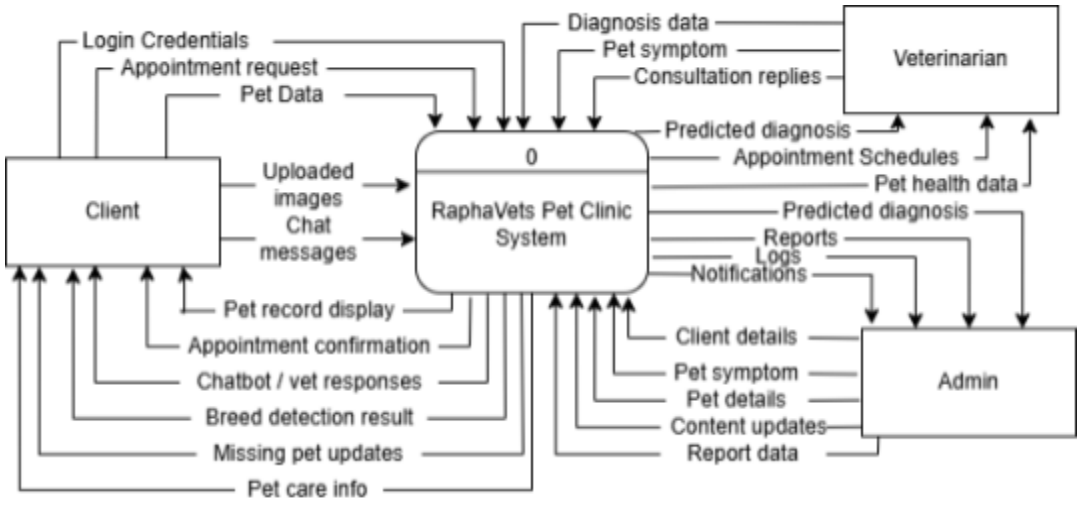

Figure 4.
Context Diagram

The figure above illustrates the overall data flow between the RaphaVets Pet Clinic System and its three main external entities, which are the client, veterinarian, and the admin. The system is represented by a single process at the center which serves as the main process that receives, processes, and sends data to and from these external actors. This diagram provides a high-level view of how the system functions as a whole and defines the boundary between what is inside the system and what exists externally.

The client represents the pet owner who interacts with the system to manage their pet-related activities. The client can log in, and provide necessary information. Through the system, the client can also request appointments, post or scroll missing pets, detect pet's breed, and view consultation results. In return, the system provides responses such as appointment confirmation, reminders, and notifications. The client can also access veterinary feedback and treatment instructions through the system. The veterinarian serves as the clinic professional who handles pet consultations and medical records. The veterinarian uses the system to view scheduled appointments, review pet records, and record diagnoses after each examination. The system provides the veterinarian with complete information about the client and pet prior to the appointment. Additionally, the system may assist the veterinarian by generating possible diagnoses using the diagnostic tool feature based on the symptoms provided by the client. The admin oversees all the operation within the system, including managing user accounts, veterinarian profiles, and overall clinic data. In return, the system provides the admin with summarized reports, activity logs, and notifications regarding appointments or user actions. The admin is also responsible for ensuring all stored data is accurate and secure.

Figure 5.
Level 1 DFD

The figure above illustrates the Level 1 Data Flow Diagram (DFD), which provides a detailed representation of how data moves between users, system processes, and data stores. It expands the main process into seven interconnected sub-processes that show how the system manages veterinary consultations, diagnostics, breed detection, appointments, forums, and reports. These processes interact with three external entities, the Client, Veterinarian, and Admin to ensure smooth data flow and system functionality.

The Manage User Account (1.0) process handles all functions related to user registration and login. Clients, veterinarians, and admins provide their login credentials, which are validated to ensure security and data accuracy. Once verified, the system stores

user details in User Data (D1). This process also sends error or success messages to users depending on the login result. The Appointment Request (2.0) process manages appointment scheduling between the client and veterinarian. Clients can send appointment requests, which the system records and forwards to the admin or veterinarian for confirmation. After approval, appointment confirmation messages are sent back to the client. All appointment details are saved in Appointment Records (D2). The Handle Diagnostic Tool (3.0) process manages pet diagnosis. Clients can provide pet symptoms, which are processed by the diagnostic tool and reviewed by the veterinarian to generate possible diagnostic results. These results are then stored in Pet Records (D3). The veterinarian also receives and reviews these results to ensure accurate diagnoses. The AI Chatbot/Consultation (4.0) process allows clients and veterinarians to communicate through chat messages. The AI chatbot provides automated replies or assists with initial consultations. Both clients and veterinarians can exchange chat messages through this module, and all conversation data are stored in Message Records (D3). The Breed Detection Module (5.0) process enables clients to upload pet images for automatic breed detection. The system analyzes the uploaded images and generates breed results, which are stored in Breed Records (D5). Clients then receive the detected breed result as feedback. The Lost and Found Forum (6.0) process serves as a community space where clients can post information about lost or found pets. The system records all posts in Forum Posts (D6), allowing admins and other users to view and respond to these entries to help locate missing pets. Lastly, the Content, Feedback, and Report Management (7.0) process is managed by the admin. This module handles user feedback, report generation, and content updates. The admin can review reports from different modules, manage

user-generated content, and ensure the quality of system data. All related information is stored in Reports (D7).

Figure 6.
Level 2 DFD: Process 1

The figure above illustrates Level 2 Data Flow Diagram for Process 1: Manage User Accounts, presents a more detailed breakdown of how user registration, login, verification, and account updates are handled within the system. This process ensures that only authorized users can access the system by maintaining accurate and verified account information. The main external entities involved in this process are client, veterinarian, and admin, while the main data store used is D1: user data, which keeps all user records securely stored for system operations.

Users can access the login user (1.1) sub-process by submitting their login details. The system retrieves the stored account details from user data (D1) to validate the provided information. If the credentials match, the system allows access to the client, veterinarian, or admin dashboard, depending on the user's role. Otherwise, an error message is displayed to inform the user of incorrect credentials. The update/delete account (1.2) sub-process allows the admin to manage existing user accounts. This function includes updating account information when users request changes or removing accounts that are no longer active or valid. The admin sends updated details or deletion requests to the system, which then modifies the stored records in the user data (D1) accordingly. After performing the action, the system returns a confirmation or error message, depending on whether the operation was successful.

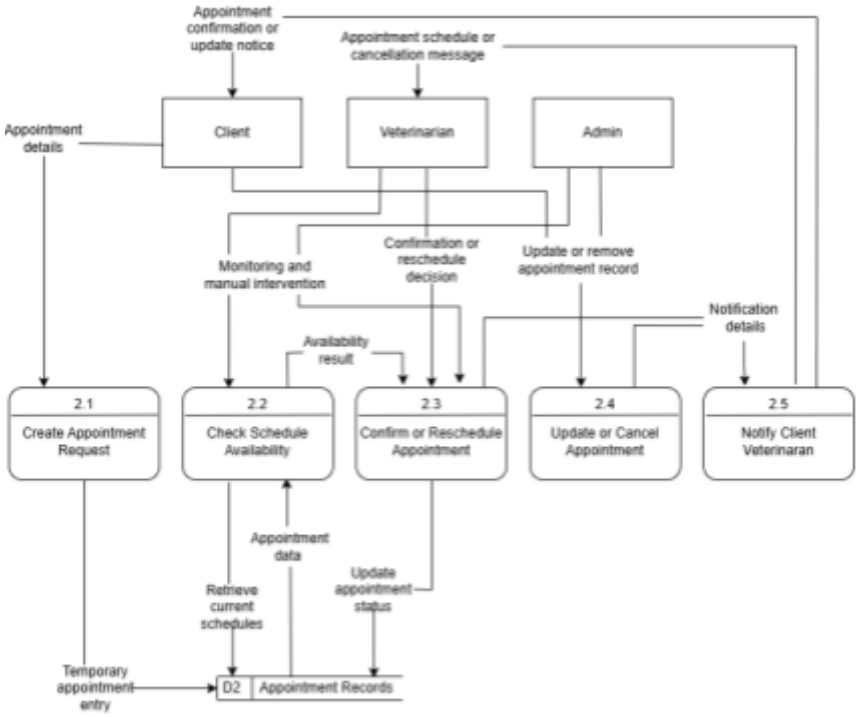

Figure 7.
Level 2 DFD: Process 2

The figure above illustrates Level 2 Data Flow Diagram for Process 2: Manage Appointments shows in detail how the system handles the entire appointment process, from when the client makes a request up to when the veterinarian confirms and the admin updates or cancels it. This process ensures organized scheduling and smooth coordination between the client, veterinarian, and admin. The main data store connected to this process is D2: appointment records, which keeps all appointment data, including schedules, confirmations, and cancellations.

The process starts when the client provides appointment details through the create appointment request (2.1) sub-process. The system records this temporary appointment entry in the appointment records (D2) and waits for validation. This request typically includes the client's preferred data, time, and pet information. Once submitted, the system proceeds to check if the chosen slot is available. The check schedule availability (2.2) sub-process retrieves the current schedules of veterinarians from appointment records (D2) to verify if the requested time is open. The system generated an availability result, indicating whether the slot can be booked or if another schedule should be chosen. This step ensures that overlapping or double-booked appointments are prevented, maintaining an efficient and accurate schedule for both clients and veterinarians. Once the availability is confirmed, the request moves to the confirm or reschedule appointment (2.3) sub-process. The veterinarian reviews the appointment details and either approves the request, reschedules it to another available slot, or rejects it when necessary. The veterinarian's decision is then updated in the appointment records (D2) to reflect the lates appointment status. The update or cancel appointment (2.4) sub-process is mainly

managed by the admin, who can modify or remove appointment records when there are changes or cancellations. The admin has the authority to correct scheduling errors, accommodate client requests, or remove outdated data. These updates are also stored in appointment records (D1), keeping all appointment data up to date. Lastly, the notify client and veterinarian (2.5) sub-process handles all communication related to appointment confirmation, rescheduling, or cancellation. The system sends notification details to both the client and the veterinarian, informing them of the status of the appointment. This ensures that all involved users are aware of any updates and can prepare accordingly.

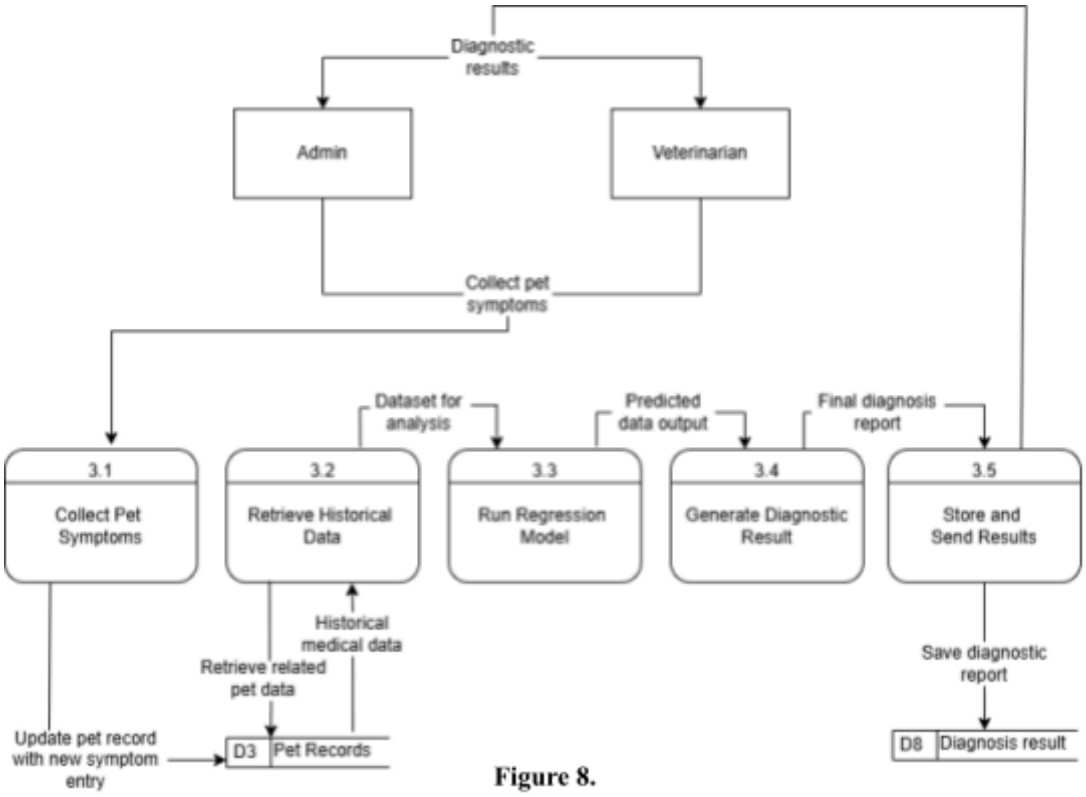

Figure 8.
Level 2 DFD: Process 3

The figure above illustrates DFD Level 2 for Process 3: Handle Diagnostic Tool shows how the system processes and analyzes health information using the diagnostic

tool to generate possible results. This process is designed by the veterinarian and admin in assessing the pet's health conditions efficiently based on symptoms provided by the client. The two main data stores involved are D3: pet records and D8: diagnosis result, which keep the system's medical history and generate diagnostic data.

The process starts with the collect pet symptom (3.1) sub process, ,where the veterinarian or admin collects the pet's symptoms from the system, either based on the client's given symptom through direct observation during consultation. The gathered information is then recorded and stored in pet records (D3), updating the existing pet profile with a new symptom entry. After the symptoms are collected, the retrieval of historical data (3.2) sub process takes place. Here, the system accesses pet records (D3) to retrieve relevant medical history or previous data of the pet. This step gathers datasets that can be used for comparison and analysis, ensuring the diagnosis considers not only current symptoms but also past medical trends and cases. The run regression model (3.3) sub-process represents the analytical core of this process. The system uses the multiple regression algorithm to analyze the dataset retrieved from historical records. By comparing input symptoms with past medical cases, the model generates predicted data outputs that suggest possible health conditions. One the diagnostic tool produces the predicted data. The generate diagnostic result (3.4) sub-process interprets the output and converts it into an understandable diagnostic report. The system compiles the predicted results, matching them with relevant medical information, and prepares a summary that includes possible conditions or recommendations. Lastly, the store and send results (3.5)

sub-process saves the final diagnosis report into the diagnostic result (D8). The system then sends the results to both the veterinarian and admin for review and verification.

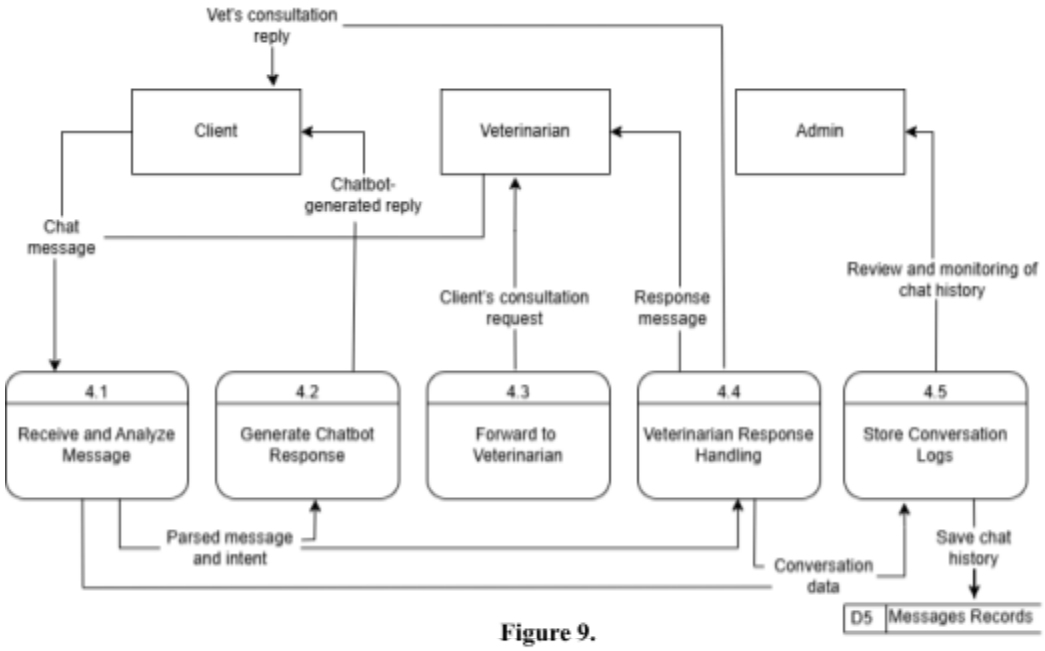

Figure 9.
Level 2 DFD: Process 4

The figure above represents Process 4: Manage Consultation through Chat, which details how communication occurs between the client, veterinarian, and admin within the system.

The process begins when the client sends a chat message to inquire about their pet's condition or to request a consultation. This message enters receive and analyze message (4.1), where the system interprets and identifies the intent of the message. After the analysis, the interpreted message is passed to generate chatbot response (4.2). At the stages, the chatbot attempts to generate an automatic reply if the query can be answered based on pre-programmed responses or general information about veterinary services. The chatbot-generated response is then sent back to the client. If the message requires

expert evaluation or cannot be handled by the chatbot, it is forwarded to the veterinarian (4.3). Here, the system sends the client's consultation request directly to the veterinarian for review. The veterinarian analyzes the request and sends a response message,which then moves to veterinarian response handling (4.4). This process ensures that the veterinarian's reply is properly formatted and delivered to the client, completing the consultation loop. All conversations data, including both chatbot and veterinarian responses, are sent to store conversation logs (4.5). This component saves the entire chat history to message records (D5), ensuring that all exchanges are archived for future reference. The admin has access to these stored records and can review and monitor chat history to ensure service quality, verify consultations, and analyze communication trends.

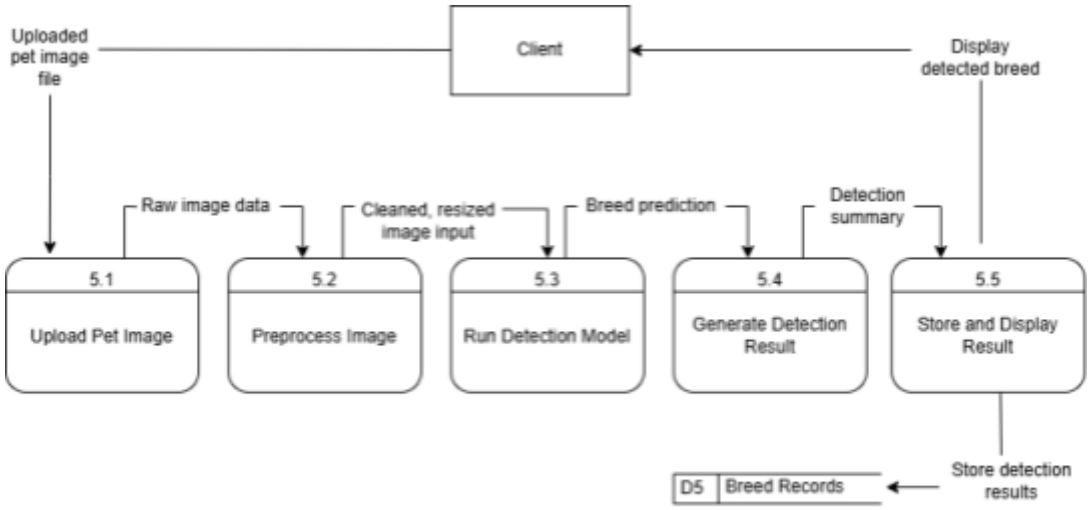

Figure 10.
Level 2 DFD: Process 5

The figure above represents Process 5: Pet Breed Detection, which illustrates how the system identifies a pet's breed through an image uploaded by the client. The process begins when the client uploads an image of their pet through the system interface. This uploaded file enters upload pet image (5.1), where the system receives and stores the raw

image data. Once the files is successfully uploaded, the data proceeds to preprocess image (5.2). In this stage, the image undergoes preprocessing, which includes cleaning, resizing and adjusting the image to ensure it meets the model's input requirements for accurate analysis. After preprocessing, the cleaned and resized image is sent to the run detection model (5.3). Here, the system uses a trained breed detection model to analyze the pet;s physical features and show the likely breed. The result of this analysis produces a breed prediction, which is passed on to generate detection result (5.4). In this stage, the breed prediction is formatted into a detection summary, which contains the identified breed along with relevant details or confidence scores. This result is then sent to store and display result (5.5), where the system both saves and displays the detected breed to the client. The detection data, including the image and its corresponding results, are stored in breed records (D5). Finally, the client can view the displayed detected breed directly from the system interface, allowing them to easily verify their pet's breed.

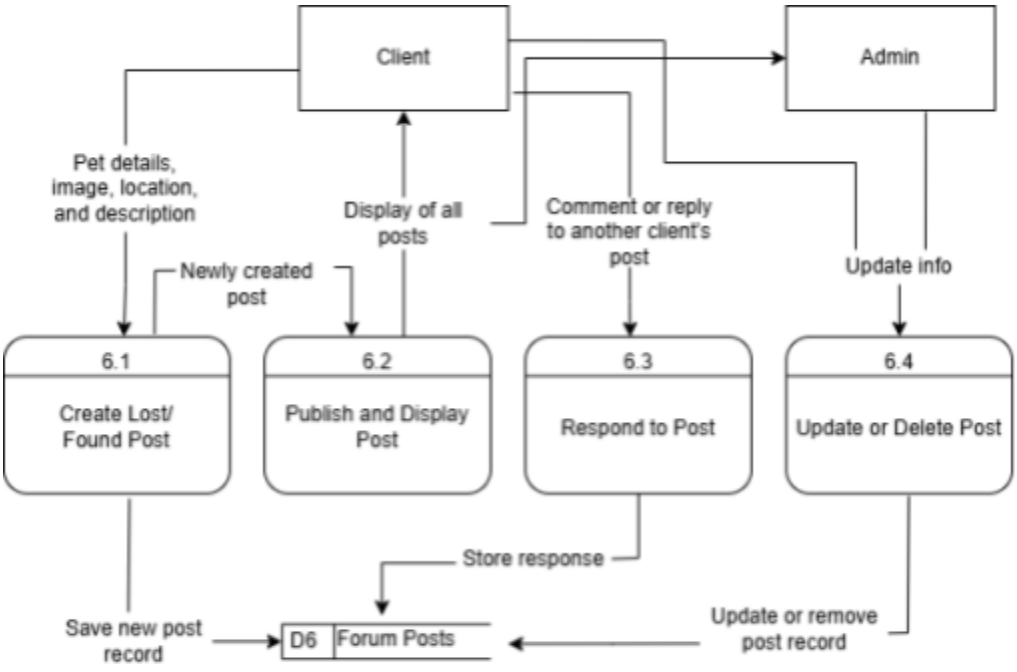

Figure 11.
Level 2 DFD: Process 6

The figure above illustrates Process 6: Lost and Found Forum Management, which details how the system facilitates the creation, publication, interaction, and maintenance of lost and found pet posts within the platform. The process starts when a client provides pet details such as the image, location, and description of a lost or found animal. This information is received by create lost/found post (6.1), where the system records the provided details to generate a new post entry. Once the post is created, it is stored in forum posts (D6) for future reference and accessibility. The newly created post then proceeds to publish and display post (6.2). Here, the system publishes the post on the forum, making it visible to all clients. The client can view all published posts on the interface, allowing them to browse or identify pets that match their reports. When users wish to interact with an existing post, their actions are handled by response to post (6.3). This process allows a client to comment on or reply to another client's post. These responses are stored and linked to the original forum post record within the forum posts (D6). Meanwhile, update or delete post (6.4) is responsible for maintaining the integrity of the forum content. Both the admin and the original client can initiate updates or deletions to correct information or remove irrelevant entries. When a change occurs, the system updates or removes the corresponding record in forum posts (D6).

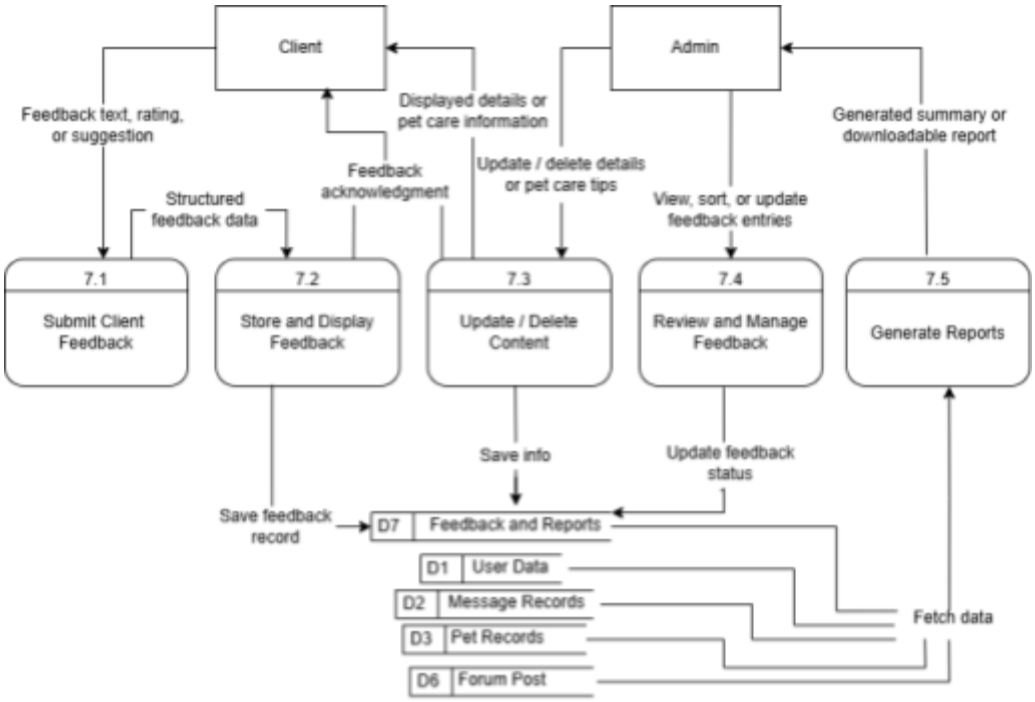

Figure 12.
Level 2 DFD: Process 7

The Level 2 illustrated Process 7: Feedback and Report Management, which represents how the system manages client feedback, pet care content updated, and administrative reporting. It shows the interaction between the client and admin in ensuring continuous service improvement and transparency through structured feedback and analytical reporting.

The process begins when a client provides feedback, rating, or suggestions through submitting client feedback (7.1). This stage involves collecting structured feedback data that reflect the client's experience with the system or veterinary services. Once submitted, the feedback is saved in feedback and reports (D7) for record-keeping and future references. Next, the feedback is processed and displayed in store and

displayed feedback (7.2). Here, the system organizes the feedback data, acknowledges receipt to the client, and displays relevant feedback content on the platform. This function ensures that users can view existing reviews or pet care tips that may help enhance their understanding and trust in the service. Meanwhile, update/delete content (7.3) allows both clients and admins to make necessary modifications to feedback entries or pet care information. Clients can edit or remove their previous submissions, while admins may update informational content to maintain accuracy and relevance. The system then saves any updated details back into feedback and reports (D7), ensuring that all data remains consistent and up to date. The admin also performs moderation and evaluation tasks in review and manage feedback (7.4). This usage involves reviewing, sorting, and managing feedback data to ensure compliance with the system policies. Admins can flag inappropriate comments, summarize general user sentiment, and track performance metrics. The updated feedback status is then reflected in the database. Finally, generate reports (7.5) is responsible for compiling insights from various data stores, including user data (D1), appointment records (D2), pet records (D3), forum posts (D6), and feedback and reports (D7), to produce summaries or downloadable reports. These reports help the admin evaluate system performance, user satisfaction, and operational trends.

Entity Relationship Diagram

Data Dictionary

User Story

User stories serve as a foundation for introducing the system's features from the perspective of its end users. This will provide a clear and concise description of how users will interact with each feature of the RaphaVets web-application. Each user story connects user experience and technical development by identifying specific goals and expected outcomes. This method helps the development team focus on delivering the functions of the features that directly address user needs. The table below presents the user stories categorized according to their corresponding features (Epic), and personas(User, Vet, and Admin) highlighting how each role engages with the application.

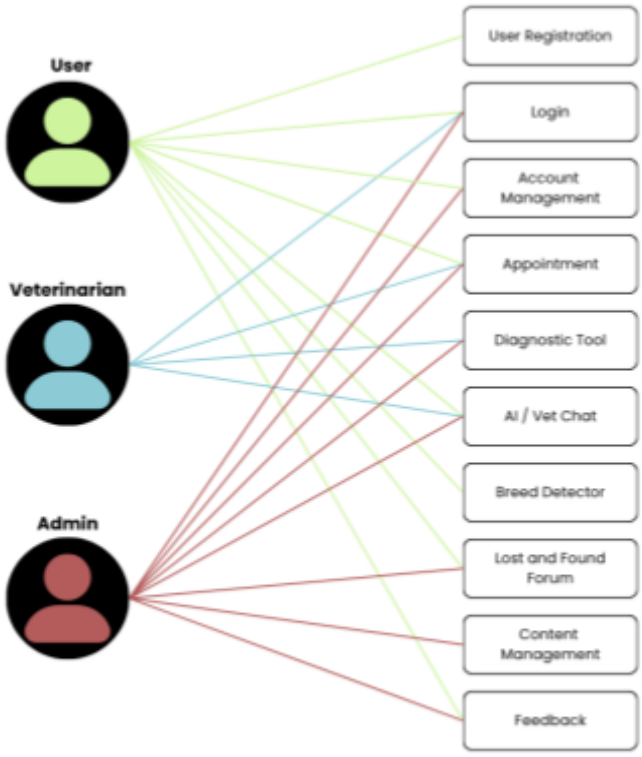

Figure 13.
User Story Diagram

Table 1. User Story Table

No	Epic	Persona	Goal	Expected result
1	User registration	As a user	I want to register an account.	To create an account, and gain access to RaphaVet's services
2	Login	As a user	I want to login my account.	To access my pets information, and gain access to RaphaVets services.
		As a vet	I want to login to the system using my credentials.	To access incoming appointments, and access the diagnostic tool.
		As an admin	I want to login with admin access	To oversee and manage users and system activities.
3	Account management	As an admin	I want to manage and verify user accounts	To secure that only authorized users can access the system
4	Appointment	As a user	I want to schedule, reschedule, or cancel an appointment with a veterinarian.	To plan my pet's checkup with ease
		As a vet	I want to view my scheduled appointments.	To prepare for upcoming consultations efficiently
		As an admin	I want to oversee the appointment schedules and manage reports	To ensure coordination between clients and veterinarians
5	Diagnostic tool	As a vet	I want to use a diagnostic tool model	To efficiently generate accurate and quick assessments on pet conditions.
		As an admin	I want to monitor diagnostic tool performance and manage data records	To maintain the accuracy and reliability of AI-generated results
6	Chatbot / Vet Chat	As a user	I want to ask questions to a	To receive initial pet care advice

			chatbot or consult a vet online	
		As a vet	I want to respond to client messages through chat	To provide guidance or follow-up consultation online
		As an admin	I want to monitor chat interactions	To ensure the platform is used properly and maintain user safety
7	Breed detector	As a user	I want to upload my pet image for breed detection	To identify my pets breed and related health information.
8	Lost and found forum	As a user	I want to post my lost pet	To seek help from others to locate or return lost pet
		As an admin	I want to moderate posts in the forum	To make the community safe and verified
9	Content management	As an admin	I want to update or remove content	So that users can reliable and current pet care information
10	Feedback	As a user	I want to submit feedback or suggestions about the system	To share my experience and help improve the platform
		As an admin	I want to review and analyze user feedback	To identify areas for improvement and enhance the system's quality

System Testing

Testing Procedures

An extensive set of tests was carried out to verify that the web application developed meets its specified functional requirements in full. During the test phase, it was ensured that the RaphaVets Web Application executes all the features intended effectively, such as scheduling appointments, responsiveness on all browsers and devices,

lost and found forum, real-time communication with the veterinarian or chatbot, and dog and cat breeder module. The following table provides the system components, testing stages, and comprehensive explanations of the procedures followed to assess the performance and reliability of each module.

Table 2: Application Functionality Testing Procedures

COMPONENT/PHASE	TEST TO BE CONDUCTED
Account management and user registration	A. Test if the registration module functions properly and successfully connects to the database. B. Verify that existing user accounts stored in the database can log in successfully using valid credentials. C. Conduct email verification testing through a One-Time Pin (OTP) sent to the user's registered email address. D. Test the password reset process, ensuring that reset instructions or verification links are sent correctly to the user's email.
Appointment booking	A. Test the scheduling, rescheduling, and cancellation of appointments to ensure accuracy and proper database updates. B. Verify that time conflicts between appointments are prevented. C. Check that appointment confirmations and details are displayed correctly to both users and administrators. D. Ensure that appointment data synchronizes correctly across browsers and devices.
Regression-based predictions	A. Verify that the module correctly receives and processes all input variables from the chatbot.

	B. Test the accuracy of the regression outputs by comparing predicted diagnoses against actual veterinary assessments. C. Ensure that the system provides predictions within acceptable accuracy or confidence levels (e.g., 80% or higher). D. Validate that results are clearly displayed in a user-friendly format for both pet owners and veterinarians.
Chatbot	A. Verify that the chatbot is available 24/7 and can respond to user queries without downtime. B. Test whether the chatbot can correctly interpret and respond to multiple pet symptoms entered by the user. C. Check that regression-based prediction provides reliable preliminary advice based on input symptoms. D. Confirm that after an appointment is booked, the chatbot's initial consultation details are automatically forwarded to the veterinarian.
Pet care tips module	A. Test that all pet care articles, videos, and resources load correctly on different devices and browsers. B. Ensure that users can easily navigate and access the pet care materials. C. Verify that all content displays with correct formatting and loads efficiently without errors.
Cat and dog breed detector	A. Test the accuracy and reliability of the breed detection feature using various pet images. B. Verify that the system correctly processes uploaded images and identifies the

	corresponding breed. C. Ensure that detection results are displayed clearly and that response time remains optimal. D. Check that image uploads follow the correct file type and size validation.
Lost and found forum	A. Test the posting, editing, and deleting of lost or found pet entries. B. Verify that uploaded images, pet details, and user information are saved and retrieved correctly. C. Ensure that users can search and view posts efficiently based on keywords or filters. D. Test that forum data updates dynamically and is properly stored in the database.
Profile management	A. Test updating, viewing, and deleting of user profile information. B. Verify that all changes are correctly reflected in real time within the database. C. Ensure that profile image uploads meet format and size requirements. D. Confirm that user data is stored securely and that unauthorized edits are prevented.
Content management	A. Verify that administrators can add, edit, and delete content through the system interface. B. Ensure that all content updates are displayed correctly across relevant pages. C. Test input validations to prevent errors or incomplete content submissions. D. Check database synchronization when content is modified or removed.

User interface and user experience design	A. Incorporates high-quality images and videos to enhance visual presentation and engagement. B. Features a user-friendly layout that promotes ease of use and accessibility. C. Provides an intuitive and well-structured navigation menu for effortless browsing. D. Utilizes an aesthetically appealing design to capture user attention and maintain interest.

Each component undergoes a series of tests to ensure the functionality and performance of the main features of the RaphaVets Application. In testing the features, registration, appointment booking, chatbot with AI, pet care tips module, regression-based prediction, lost and found forum, and content management are included in order to verify the accuracy, reliability, and usability of each module while making sure that data integrity and smooth user interaction are maintained. Any system errors or inconsistencies that have been found using these tests will serve as a basis for further optimization and enhancement by the developers.

Testing starts with the registration of accounts and follows through their successful login. The module for registration is put to test in areas such as proper database connection, handling duplicate entries, and validation of correct data entry. The email verification is checked through an OTP system for secure authentication, and the reset password feature must ensure the users can retrieve accounts through verification links sent to the email address registered with the website or app.

The appointment booking module is tested to verify that users can rightly schedule, reschedule, and cancel appointments. Also, the process confirms that all the data about appointment details are appropriately stored and updated in the database. Tests also confirm that appointment overlaps or conflicts are avoided, while confirmation details are accurately shown to users and administrators. Synchronization on different browsers and devices has been conducted to verify data consistency and proper functionality.

Testing procedure for chatbot component with AI integration focuses on ensuring the 24/7 availability an responsiveness to user queries. The chatbot is evaluated on its accuracy or confidence level that it can interpret inquiries provided by the users and to generate relevant and useful responses. It is also tested for the integration of the regression-based prediction system for proper processing of all fed variables into the model. Moreover, accuracy verification is performed by comparing the system outputs with real-life veterinary assessments. Performance will be verified regarding the reliability of prediction so that the system maintains acceptable levels of accuracy and show results clearly and in a user-friendly manner.

The cat and dog breed detector is tested for accuracy of system's ability to correctly process uploaded images and identify the corresponding pet breeds. The accuracy and response time of the detection process are assessed, while image uploads are validated to ensure file type and size compatibility is met. Results must be displayed clearly and consistently to maintain user satisfaction. Testing for the lost and found forum

focuses on ensuring that registered users can post entries regarding their missing or found pets, edit, and delete those entries. Uploaded images, descriptions, and contact details are checked for proper saving and retrieval from the database. It is also tested for keyword search and filtering functions to a user's capability to browse efficiently and locate relevant places.

The administrator can add, edit, and delete content using the platform's interface, and this is tested. Updates are checked for synchronization across related pages, and input validation is performed to prevent incomplete or incorrect data entries. Testing also ensures that any modification of content can be stored and reflected in the database correctly. This includes the checking of the user interface and the user experience design for visual appeal, intuitive, and ease of navigation. Checking for accessibility, consistency in layout, and responsiveness on devices; ensuring images, icons, and multimedia display as they should in terms of quality and how they load, all contribute to an engaging and user-friendly environment. Testing is also carried out in variable network conditions, device types, and browsers to ensure the stability of the application. Various simulated conditions include low internet connections, multiple users logging in at the same time, and disrupted sessions; these have all been tested to ensure the system maintains data integrity and errors are relayed back in clear messages. These procedures confirm that RaphaVets performs efficiently and consistently across all environments.

System Development

The development of the system required developers to create a well designed and user-friendly web design prototype and translate it into a functional well developed working web application using the chosen combination of technologies used to build and run a software application or website of the developers. This involves set of codes via programming languages or frameworks for each functionality. This includes debugging, testing, and improving written code ensuring proper functionality and responsiveness.

System Evaluation

The system development, testing, and integration using selected technology stack. The system was developed using React.js for the front-end, Node.js with Express for the backend, and MySQL for database management. A series of code debugging and refinement is made to address detected errors and improved overall funphase focuses on converting the system design and wireframes of the RaphaVets web-application into a functional software. This phase involved the implementation of features via coding, functionality.

Documentation

Documentation is the process of recording all information related to the development of the system. It includes written details about the design, codes, database, and instructions on how to use the system. Having complete documentation helps developers, users, and future researchers understand how the system works. It also makes

it easier to fix errors, update features, or improve the system later on. Good documentation provides clear and organized records that can serve as a guide for anyone who will maintain or study the system in the future.

Operation Procedure

The figure below shows the operating procedure of the proposed project of RaphaVets Pet Clinic System:

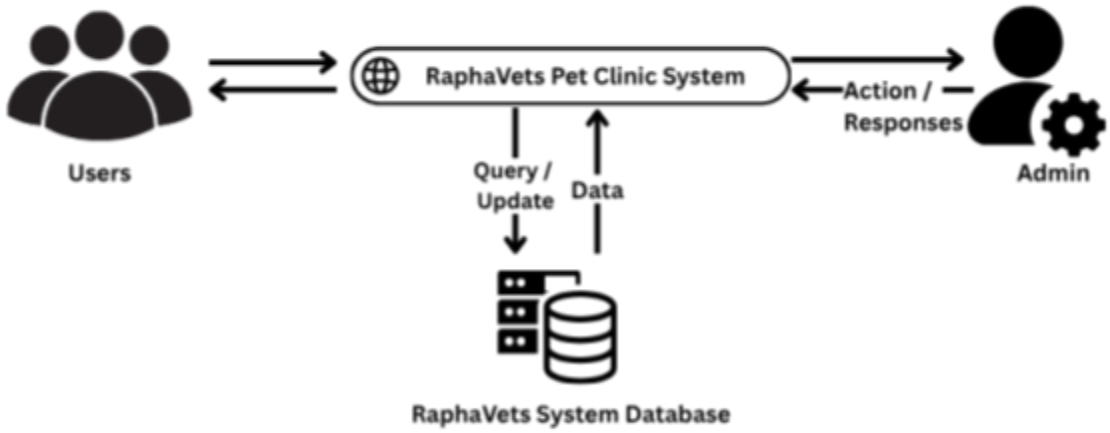

Figure 14.
RaphaVets Pet Clinic Platform

The RaphaVets operates as a user-friendly, clinic-centered web service that starts with simple, secure account creation and email verification. The RaphaVets platform allows different types of users. Users with no accounts using the website are considered guests and could continue to use the platform and other features using guest accounts, but with very limited access and cannot use the full capacity and features of the platform.

Second are the primary users, they have to create an account or their account must be created by the admin or veterinarian. They have full user side access, if they wanted to book then the system guides pet owners through an easy appointment flow: submit

preferred date/time, the system checks veterinarian availability, and the clinic confirms or proposes alternatives while sending timely SMS and email reminders, before consultations users can interact with an AI assistant that collects symptoms or use other features so veterinarians arrive informed, and all consultation notes, pet records, and any regression-based suggestions are stored in the clinic's electronic record for clinician review only; realtime chat enables brief triage or follow-up while a community lost-and-found forum amplifies local help through moderated posts.

Lastly, administrators manage schedules, moderate content, audit logs, and generate simple analytics to surface trends. Privacy and safety are built in — the chatbot and breed detector are explicitly advisory, image quality limits are acknowledged, and serious or ambiguous cases are escalated to clinicians; the system is deployed with repetitive testing, a maintenance plan, and a human in the loop governance model so RaphaVets improves over time without substituting professional veterinary judgment.

Evaluation Procedure

Preliminary Evaluation

Final Evaluation

Evaluation Instrument

Treatment of Data

This section addresses the statistical tools and procedures used in analyzing the data gathered from respondents. The functionality and usability of the RaphaVets Pet

Clinic System were evaluated using a four-point Likert scale as part of the survey instrument. Respondents were asked to rate each criterion on a scale of one (1) to four (4), corresponding to qualitative judgments such as *"Poor" to "Excellent"* or *"Strongly Disagree" to "Strongly Agree"*, depending on the type of question. Each response was assigned a corresponding numerical value, enabling the quantitative analysis of qualitative perceptions.

The data collected were subjected to statistical computation to determine the mean and overall mean, which serve as the basis for interpreting the level of user satisfaction and system performance. The mean represents the average score for each criterion, while the overall mean indicates the general performance across all criteria.

Mean Formula: $M = \sum fx/n$	Where: M = Mean f = frequency (number of respondents who chose a particular answer) x = score n = total number of responses on that criterion
Overall mean = Sum of X/n	X = sum of all scores in each criteria n = total number of criteria

The following scale was used as the basis for interpreting the computed mean values for each criterion:

Scale	Rating	Description	Mean Interpretation

4	**Excellent**	Greatly surpasses expectations	The mean value indicates a high level of user satisfaction and exceptional system performance.
3	**Good**	Meets expected standards	The mean value signifies satisfactory system performance and positive user experience.
2	**Fair**	Falls short of expectations	The mean value reflects moderate satisfaction, suggesting that certain areas need improvement.
1	**Poor**	Requires substantial improvement	The mean value indicates low user satisfaction and inadequate system performance.