Software Requirements Specification

For

PET MED CARE

Prepared By

Dilshan Padukka

2024 October

Table of Contents

Table of Contents……..……...1

1. Introduction……...3

1.1 Purpose…….3
1.2 Document Conventions……....4
1.3 Intended Audience and Reading Suggestions ……4
1.4 Project Scope……5
1.5 References ……...5

2. Overall Description……...6

2.1 Product Perspective……6
2.2 Product Features……...……....6
2.3 User Classes and Characteristics…….……7
2.4 Operating Environment……7
2.5 Design and Implementation Constraints……..…8
2.6 User Documentation…….……8
2.7 Assumptions and Dependencies……...……8

3. System Features……..……...9

3.1 User Authentication and Authorization..……..9
3.2 Pet Management…….……...……....9
3.3 Appointment Scheduling…..……9
3.4 Medical Records Management……9
3.5 Reporting and Analytics…..……9

4. External Interface Requirements…….……10

4.1 User Interfaces……...……10
4.2 Hardware Interfaces…….……..10
4.3 Software Interfaces…….……...10
4.4 Communication Interfaces…….……..10

5. Other Nonfunctional Requirements……11
5.1 Performance Requirements……..……..11
5.2 Safety Requirements…..……..……..11
5.3 Security Requirements... ……..…...……...11
5.4 Software Quality Attributes…….. …….……...……...11
5.5 Business Rules…….…….……..……..11

6. Other Requirements……..…….……12

Appendix A: Glossary…….…….……12
Appendix B: Analysis Models……..……....13
Appendix C: Issues List……..……..15
Appendix D:Research Sheet….……..……..16
Appendix E:Competitor Details…….……16

1. Introduction

1.1 Purpose

The primary purpose of the **Pet Med Care** is to enhance the operational efficiency of veterinary clinics by automating various day-to-day tasks. The system is designed to help clinics manage their appointments, track pet medical histories, handle customer interactions, and manage billing processes seamlessly. It addresses the challenges of manual record-keeping and inefficient scheduling, ensuring that veterinarians, staff, and pet owners can interact with the clinic in a streamlined and user-friendly manner.

By utilizing a **microservices architecture**, the system offers a modular and scalable solution, allowing clinics to expand or modify the system as their needs grow. It ensures that sensitive data, such as medical records and billing information, are managed securely.

The specific goals of this system include:

1. **Improving efficiency**: Reducing administrative workload for staff, allowing more time to focus on patient care.
2. **Enhancing accuracy**: Minimizing errors in medical records, appointment scheduling, and billing processes.
3. **Facilitating user experience**: Offering a simple and intuitive interface for staff, veterinarians, and pet owners.
4. **Supporting scalability**: Allowing the system to grow with the clinic's needs, easily adding new features or services when required.
5. **Ensuring data security**: Protecting sensitive information, ensuring secure data storage and compliance with industry regulations.

In essence, the **Pet Med Care** is developed to digitize and streamline veterinary clinic operations, ensuring better customer service, improved clinic management, and ultimately, enhanced care for pets.

1.2 Document Conventions

This document follows a structured format to ensure consistency and clarity in the presentation of requirements and technical details. The following conventions have been applied throughout:

- **Requirement Prioritization**: Each requirement in this document is categorized based on its priority and impact on the system:
    - **Must Have (MH)**: These are critical features that are essential for the system's functionality and must be implemented in the initial release.
    - **Should Have (SH)**: These features are important but not critical. While the system will still function without them, their inclusion is highly recommended to enhance overall performance and user satisfaction.
    - **Could Have (CH)**: These are desirable features that are not essential but would improve the user experience or system capabilities if included. They may be deferred for future releases if necessary.
- **Technical Terms**:
    - The first use of any specialized or technical term in the document will be italicized and accompanied by a brief definition.
    - All such terms are consolidated and fully defined in the **Glossary** section at the end of the document.

This structure ensures that stakeholders can easily identify the importance of each requirement and understand the key technical terms used throughout the document.

1.3 Intended Audience and Reading Suggestions

Intended Audience:

- **Veterinarians & Clinic Managers**: Understand system functionalities to optimize clinic operations (appointments, medical records).
- **Software Developers**: Grasp system requirements and architecture for effective implementation.
- **Product Managers**: Review system features and ensure alignment with business needs.
- **System Administrators**: Familiarize with technical deployment, scaling, and maintenance.
- **QA/Testers**: Understand system features to create test cases and ensure reliability.

Reading Suggestions:

- **Veterinarians & Clinic Managers**: Focus on Overall Description and System Features.
- **Developers & Administrators**: Review Product Features, External Interfaces, and Nonfunctional Requirements.
- **QA/Testers**: Focus on System Features and Software Quality Attributes.

1.4 Project Scope

The **Pet Med Care** is a comprehensive veterinary practice management solution designed to streamline the daily operations of veterinary clinics. Its primary functionalities include:

- **Managing Pet and Owner Information**: The system allows for the creation, update, and retrieval of detailed pet and owner profiles, including medical history, species, breed, and contact details.
- **Handling Appointment Scheduling**: The system offers tools for scheduling, modifying, and canceling appointments. It also includes automated reminders for both clients and veterinarians to reduce missed appointments.
- **Maintaining Medical Records**: Veterinarians can document treatments, vaccinations, diagnoses, and procedures, ensuring that all medical history is securely stored and easily accessible for future reference.
- **Managing Veterinarian Schedules and Specialties**: The system helps manage clinic staff availability, specialty services offered by different veterinarians, and the assignment of appointments based on expertise and schedule.

The **Pet Med Care** is designed to enhance efficiency, reduce manual work, and provide a seamless experience for both veterinarians and pet owners, contributing to better overall clinic management and customer satisfaction.

1.5 References

➢ Spring Pet Clinic Reference Implementation. Retrieved from Spring Pet Clinic GitHub Repository.
➢ Veterinary Practice Management Standards. American Animal Hospital Association. (n.d.). Retrieved from AAHA Standards.
➢ Microservices Architecture Best Practices. Martin Fowler. (n.d.). Retrieved from martinfowler.com.

2. Overall Description

2.1 Product Perspective

The Pet Med Care is a web-based application built using a microservices architecture, requiring server access to provide a scalable and flexible solution for veterinary clinics. It consists of:

- **Frontend**: An Angular-based responsive user interface for users
- **Backend Microservices**:
    - **User Service**: Handles user authentication, registration, and profile management, ensuring secure access to the system.
    - **Pet Service**: Manages pet information, including breed, age, and medical history, allowing users to maintain detailed records of their pets.
    - **Appointment Service**: Facilitates the scheduling of appointments, ensuring efficient management of vet consultations and pet care services.
    - **Medical Records Service**: Stores and retrieves medical histories and treatment details for each pet, ensuring comprehensive tracking of their health status.

This architecture allows each microservice to be developed, deployed, and scaled independently, promoting flexibility and maintainability.

2.2 Product Features

The Pet Med Care offers the following key features:

- **User Management**: Supports different user roles, including pet owners, veterinarians, and administrators, allowing for tailored access and functionality based on user type.
- **Pet Information Management**: Enables users to create, update, and view detailed profiles for their pets, including essential information such as breed, age, and medical history.
- **Appointment Scheduling**: Provides an intuitive interface for users to book, modify, and cancel appointments with veterinarians, ensuring optimal time management and service delivery.
- **Medical Records Management**: Allows for the secure storage and retrieval of pets' medical histories, including vaccinations, treatments, and notes from veterinary visits, facilitating comprehensive health tracking.
- **Reporting and Analytics**: Offers insights through various reports and analytics tools, helping users and administrators to track system usage, appointment statistics, and pet health trends for informed decision-making.

2.3 User Classes and Characteristics

Pet Owners

- **Responsibilities**:
    - Register and manage their accounts.
    - Schedule appointments with veterinarians.
    - View and maintain their pets' medical history.

Veterinarians

- **Responsibilities**:
    - Manage and oversee appointment schedules.
    - Update and maintain medical records for pets.
    - Set their availability for consultations and appointments.

Administrators

- **Responsibilities**:
    - Manage system users, including pet owners and veterinarians.
    - Configure system settings to optimize functionality.
    - Generate reports for system usage, appointments, and other analytics.

2.4 Operating Environment

- user-friendly interface for all user classes.
- **Supported Browsers**: The application is compatible with the latest versions of major web browsers, including:
    - Chrome
    - Firefox
    - Safari
    - Edge
- **Backend**: Built on Spring Boot microservices, ensuring robust performance, scalability, and modularity in development and deployment.
- **Database**: Utilizes MySQL as the relational database management system, allowing efficient data storage and retrieval for user and pet information.
- **Deployment**: The application is deployed using Docker containers, enabling easy management, scaling, and portability across different environments.

2.5 Design and Implementation Constraints

- Frontend Framework: Angular
- Backend Framework: Spring Boot 2.5 or later
- Database: MySQL 8.0 or later
- ORM: Hibernate with JPA
- API Design: RESTful architecture
- Build Tool: Maven
- Version Control: Git

2.6 User Documentation

- Online help integrated into the application
- User manuals for pet owners, veterinarians, and administrators
- Quick start guides for each user role
- Video tutorials for key features

2.7 Assumptions and Dependencies

- The system assumes a stable internet connection for all users.
- The system depends on the availability and reliability of the chosen cloud hosting provider.
- It is assumed that all users have basic computer literacy and familiarity with web applications.
- The system depends on third-party libraries and frameworks, which are assumed to be maintained and updated regularly.

3. System Features

3.1 User Authentication and Authorization

- ➢ Secure user authentication with role-based access control for pet owners, veterinarians, and administrators.
- ➢ Features: User registration, login, password encryption (bcrypt), JWT-based authentication, password reset.
- ➢ Priority: Must Have (MH), Should Have (SH).

3.2 Pet Management

- ➢ Manage pet profiles (name, breed, age, weight, etc.) and link pets to owners.
- ➢ Features: Add, update, view pet info, upload pet photos, track profile changes, manage multiple pets.
- ➢ Priority: Must Have (MH), Should Have (SH).

3.3 Appointment Scheduling

- ➢ Schedule, modify, and cancel appointments with veterinarians while preventing conflicts.
- ➢ Features: Veterinarian availability, automated reminders, recurring appointments, wait-list management.
- ➢ Priority: Must Have (MH), Could Have (CH), Should Have (SH).

3.4 Medical Records Management

- ➢ Create and update medical records, including diagnoses, treatments, prescriptions, and vaccination tracking.
- ➢ Features: Attach documents, prescription management, view medical history.
- ➢ Priority: Must Have (MH), Should Have (SH).

3.5 Reporting and Analytics

- ➢ Generate reports on clinic operations, pet health trends, and business performance.
- ➢ Features: Standard/custom reports, data visualization, export functionality, automated reporting.
- ➢ Priority: Should Have (SH), Could Have (CH).

4. External Interface Requirements

4.1 User Interfaces

- Responsive Web Interface (MH): Design must adapt to various screen sizes (desktop, tablet, mobile).
- Intuitive Navigation (MH): Clear, user-friendly navigation.
- Consistent Styling (SH): Use Angular Material for uniform design.
- Accessibility Compliance (SH): Meet WCAG 2.1 Level AA standards.
- Localization Support (CH): Support for multiple languages with easy switching.

4.2 Hardware Interfaces

- Standard Input Devices (MH): Support for keyboards and mice.
- Mobile Device Compatibility (SH): Ensure touch-based compatibility for smartphones/tablets.
- Printer Integration (CH): Enable printing of reports and documents.

4.3 Software Interfaces

- Database: MySQL 8.0 (MH): Use MySQL for data management.
- Message Broker: RabbitMQ (SH): Facilitate microservice communication.
- Cache: Redis (SH): Use Redis for performance improvement.
- Container Platform: Docker (MH): Ensure deployment consistency with Docker.
- External Payment Gateway (CH): Integrate with Stripe or PayPal for payments.

4.4 Communications Interfaces

- HTTPS (MH): Secure data transmission via HTTPS.
- RESTful APIs (MH): Use REST APIs for frontend-backend communication.
- WebSocket (SH): Enable real-time updates (e.g., appointments).
- Email Notifications (SH): Send appointment reminders/updates via email.
- SMS Notifications (CH): Provide SMS alerts for critical updates.
- API Documentation (SH): Maintain documentation using Swagger/OpenAPI.
- API Versioning (SH): Ensure backward compatibility.
- Rate Limiting & Throttling (SH): Prevent API abuse.
- Health Check Endpoints (SH): Monitor service health.
- Logging and Monitoring (MH): Track system performance with Prometheus/Grafana.

5. Other Nonfunctional Requirements

5.1 Performance Requirements

- **Response Time (MH)**: 95% of interactions < 2 seconds; page load < 3 seconds.
- **Scalability (MH)**: Support for up to 1000 concurrent users with horizontal scaling.
- **Availability (MH)**: 99.9% uptime with disaster recovery and backup plans.
- **Database Performance (SH)**: Queries < 1 second, with indexing and optimization.
- **Caching (SH)**: Efficient caching to reduce database load.

5.2 Safety Requirements

- **Data Backup and Recovery (MH)**: Regular backups with a 4-hour recovery window.
- **Error Handling and Logging (MH)**: Comprehensive error handling and detailed logging.
- **Input Validation (MH)**: Prevent malicious input with validation and sanitization.
- **Secure File Uploads (SH)**: Validate file types and scan for viruses.
- **Audit Trails (SH)**: Tamper-proof logs for key activities.

5.3 Security Requirements

- **Authentication (MH)**: Multi-factor authentication and strong password policies.
- **Authorization (MH)**: Role-based access control (RBAC) with least privilege.
- **Data Encryption (MH)**: AES-256 for data at rest and TLS 1.3 for data in transit.
- **Session Management (SH)**: Secure sessions with timeouts and cookie protection.
- **API Security (SH)**: OAuth 2.0/JWT authentication and rate limiting.
- **Compliance (MH)**: GDPR, HIPAA compliance with data protection features.

5.4 Software Quality Attributes

- **Maintainability (SH)**: Modular architecture with clear documentation.
- **Testability (SH)**: Unit, integration, and end-to-end testing.
- **Usability (MH)**: Intuitive UI with usability testing.
- **Reliability (MH)**: Fault-tolerant design with circuit breakers.
- **Interoperability (SH)**: Adhere to industry standards for easy external integration.

5.5 Business Rules

- **Appointment Scheduling (MH)**: Enforce business hours, prevent overbooking.
- **Billing and Invoicing (SH)**: Automate invoices with flexible pricing models.
- **Prescription Management (MH)**: Check for drug interactions and dosage limits.
- **Data Retention (MH)**: Implement legal-compliant retention policies.
- **Reporting Requirements (SH)**: Standard reports and automated KPI generation.

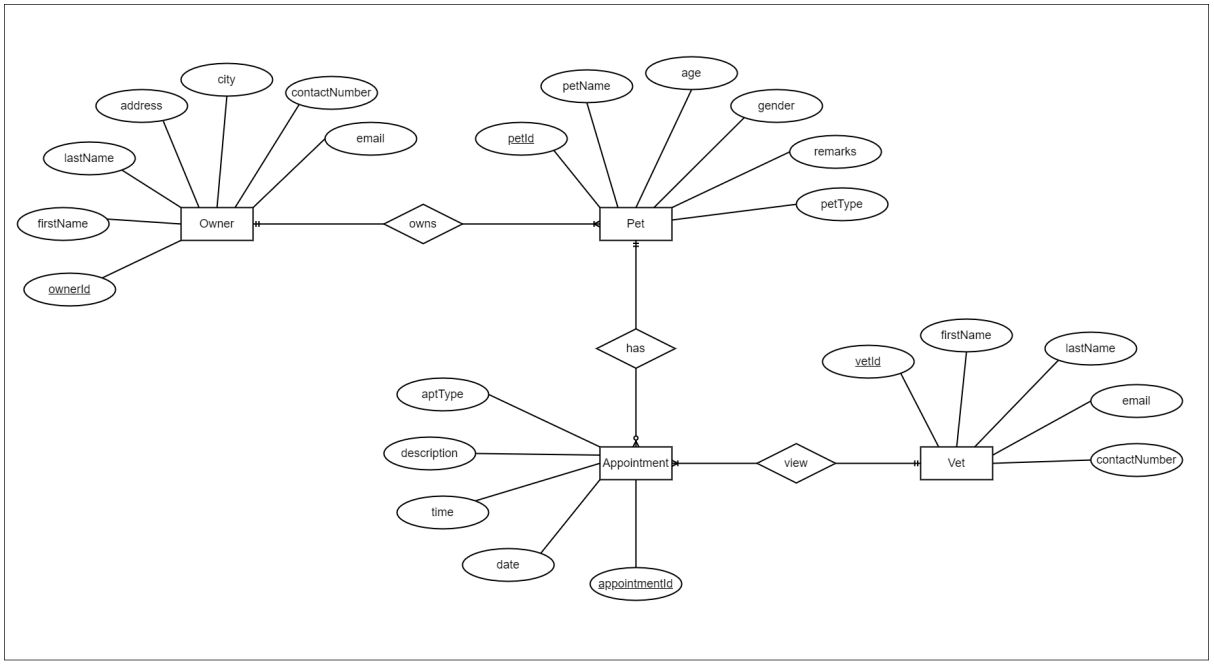
6. Other Requirements

Appendix A: Glossary

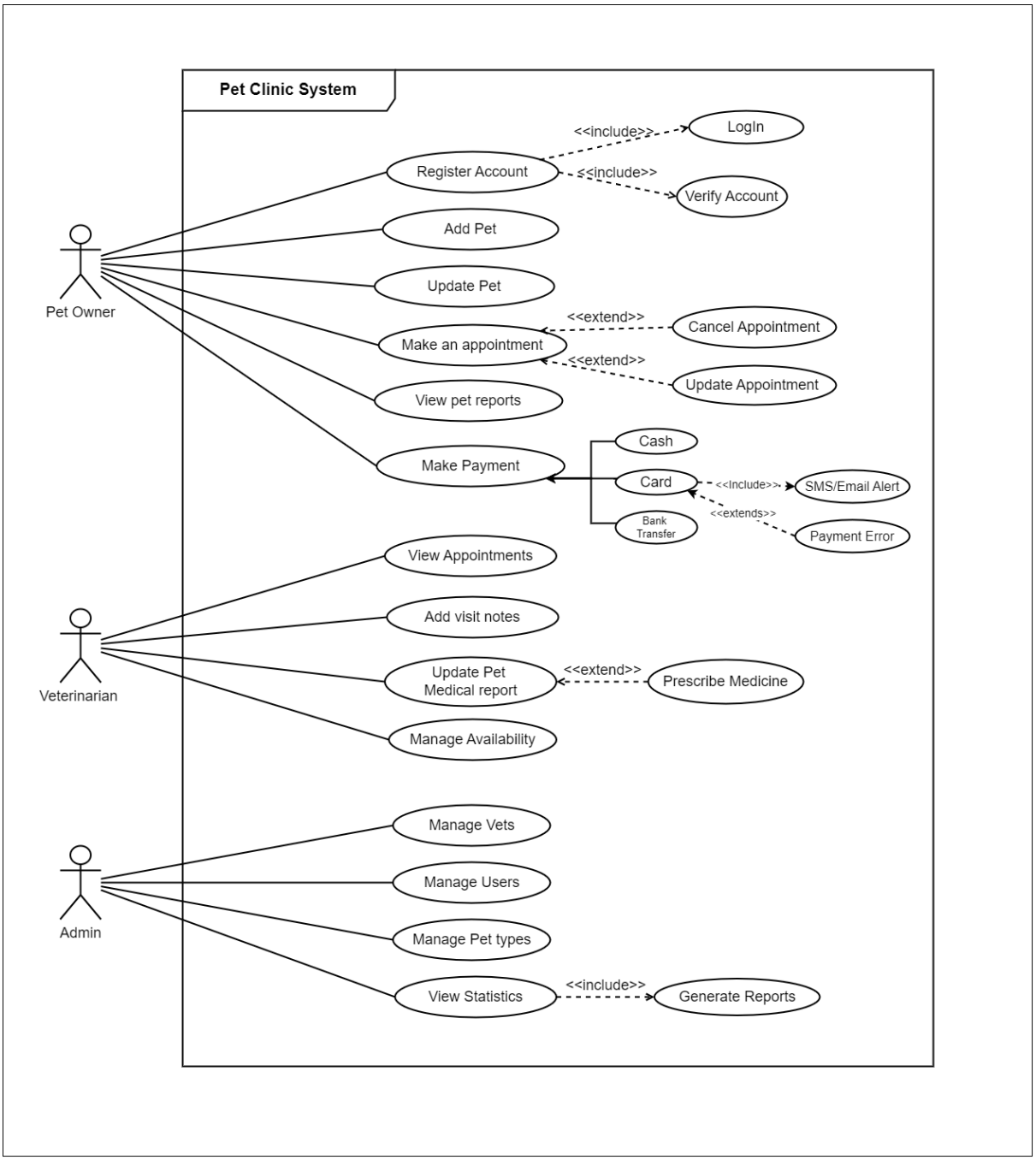
- **API**: Application Programming Interface
- **CCPA**: California Consumer Privacy Act
- **GDPR**: General Data Protection Regulation
- **HIPAA**: Health Insurance Portability and Accountability Act
- **JWT**: JSON Web Token
- **KPI**: Key Performance Indicator
- **OAuth**: Open Authorization
- **PCI DSS**: Payment Card Industry Data Security Standard
- **RBAC**: Role-Based Access Control
- **REST**: Representational State Transfer
- **SLA**: Service Level Agreement
- **SPA**: Single Page Application
- **SQL**: Structured Query Language
- **TLS**: Transport Layer Security
- **UI**: User Interface
- **UX**: User Experience
- **XSS**: Cross-Site Scripting

Appendix B: Analysis Models

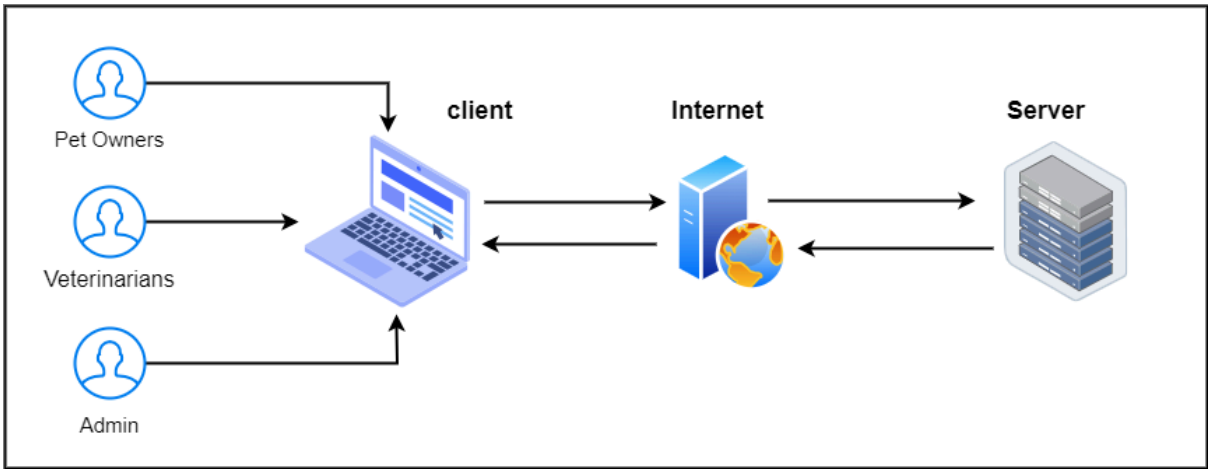
- **Entity Relationship Diagram**: A visual representation of the entities in the system and their relationships, helping to illustrate the data structure.



- **Use Case Diagram**: A diagram that identifies the interactions between users (actors) and the system, outlining the functional requirements.



- **System Architecture Diagram**: Displays the overall architecture of the application.



Appendix C: Issues List

- **Legacy System Integration**: Define data migration and integration with existing systems.
- **Compliance**: Ensure adherence to international veterinary standards.
- **Mobile App Development**: Decide on native app development vs. responsive web app.
- **AI and Predictive Analytics**: Explore AI for health analytics and personalized care.
- **Telemedicine**: Evaluate remote consultation features, pending regulatory approval.
- **Blockchain for Medical Records**: Investigate blockchain for secure records.
- **IoT Pet Monitoring**: Explore IoT for real-time pet health data collection.
- **Multi-Clinic Support**: Assess requirements for supporting multiple clinics.
- **Third-Party Integrations**: Prioritize connections with services like pet insurance and labs.
- **Data Migration**: Plan data migration, cleaning, and transformation strategies.

Appendix D: Research Sheet

This research sheet collects information on user needs and preferences relevant to the Pet Clinic Management System. For detailed insights, please refer to the following link

- ❖ Click Here : https://forms.gle/aorUzGMLU9ub2n3a6 / PET MED CARE

Appendix E: Competitor Details (Sri Lanka)

- **Vetstoria Sri Lanka**: Offers veterinary scheduling software with real-time booking, integration with clinic management systems, and telemedicine options.
- **VetOne Sri Lanka**: Provides cloud-based veterinary management, focusing on patient record-keeping, appointment scheduling, and billing for local veterinary clinics.
- **Zoetis Vet Solutions**: International brand with a presence in Sri Lanka, offering veterinary management software for clinics, focusing on medical records, diagnostics, and telemedicine.

***