

Veterinary Management System Requirements Specification

Final Version
June, 2025

Veterinary Management System Requirements Specification

Table of Contents

1. Executive Summary
- 1.1 Project Overview
2. Product/Service Description
 - 2.1 Product Context
 - 2.2 User Characteristics
 - 2.3 Assumptions
 - The veterinary clinic will have computers, barcode scanners (for medication scanning and prescriptions), and printers available for use.
 - 2.4 Constraints and Dependencies
 1. Constraints
 - 1.1 System Integration Constraints
 - 1.2 Security and Compliance Constraints
 - 1.3 Performance and Scalability Constraints
 - 1.4 User Accessibility Constraints
 - 1.5 Hardware and Network Constraints
 - 1.6 Operational Constraints
 2. Dependencies
 - 2.1 Parallel Operation with Legacy Systems
 - 2.2 Dependency on External Modules
 - 2.3 Data Synchronization Dependencies
 - 2.4 Training and Adoption Dependencies
 3. Requirements
 - 3.1 Functional Requirements
 - 3.2 Non-Functional Requirements
 - Product Requirements
 - Client Registration

The system shall allow users to register clients by entering their personal details, such as full name, address, contact information, and preferred method of communication. The system should also support updating and deleting client information.
 - Pet Registration

Each pet shall be linked to a registered client. The system shall capture pet details such as name, species, breed, gender, age, medical history, vaccination records, and microchip number (if available).
 - Appointment Scheduling

Users shall be able to schedule, reschedule, or cancel appointments. The system shall send automated reminders to clients through SMS or email. Additionally, it shall prevent scheduling conflicts by checking for available time slots.
 - Medical Records Management

The system shall maintain a complete medical history for each pet, including previous illnesses, surgeries, treatments, and prescribed medications. Veterinary staff shall be able to update and retrieve medical records as needed.
 - Billing and Invoicing

Veterinary Management System Requirements Specification

The system shall generate invoices for veterinary services and products purchased. It should support different payment methods, including cash, credit/debit cards, and online payments. Users shall also be able to track outstanding balances and payment history.

3.2.1.1 Usability Requirements

3.2.1.4 Security

3.2.1 Organizational Requirements

3.2.2 External Requirements

4. User Scenarios/Use Cases

- Use Case 1
- Use Case 2
- Use Case 3
- Use Case 4
- Use Case 5
- Use Case 6
- Use Case 7
- Use Case 8
- Use Case 9
- Use Case 10
- Use Case 11
- Use Case 12
- Use Case 13
- Use Case 14
- Use Case 15
- Use Case 16
- Use Case 17
- Use Case 18
- Use Case 21
- Use Case 22
- Use Case 23
- Use Case 24
- Use Case 25
- Use Case 26
- Use Case 27
- Use Case 28
- Use Case 29

Who Does What?

Actors and Their Responsibilities

1. System Administrator (SA)

2. Veterinarian (Vet)

3. Receptionist (Rec)

4. Pet Owner (PO)

5. Finance Officer (FO)

5. Diagrams

6. Design Patterns

1. Executive Summary Era

1.1 Project Overview

The **veterinary management system (VMS)** ensures smooth communication between veterinarians and pet owners, provides an efficient storage solution for patient information, lays out the patient appointment scheduling in a simple matter, making it so the user experience is pleasant and easy.

2. Product/Service Description

The **Veterinary Management System** is designed to enhance the efficiency and organization of veterinary clinics by providing a centralized platform for managing patient records, appointments, billing, and medical history. The system is intended to **streamline daily operations, reduce paperwork, minimize errors, and improve the overall experience for veterinarians, staff, and pet owners**(patients/users).

2.1 Product Context

The Veterinary Management System is a self-contained(*containing in oneself or itself all that is necessary; independent*) software but also interfaces with external systems, such as:

- **Payment Gateways** (for online payment processing)
- **Insurance Systems** (for verifying pet insurance coverage, if needed/available)
- **Pharmacy Databases** (for medication availability and prescription tracking)
- **Laboratory Information Systems** (for real-time diagnostic test results integration)

The system can be implemented as a standalone desktop/web application or integrated with a **larger hospital management system** if needed.

2.2 User Characteristics

There are many users which are not included, such as general staff, veterinary assistants, etc, however we consider these 4 to be the key users to the application.

User Type	Experience Level	Technical Expertise	General Characteristics
<i>Veterinary</i>	High (Expected to have finished higher education in the field)	High	Diagnosing and treating pets, prescribing medication.
<i>Receptionists</i>	Medium to low	Medium (A receptionist should know how to navigate various software.)	Scheduling appointments, handling payments, noting down information regarding pets if needed.
<i>Pet owners</i>	None	Low (Should know how to operate around the website/application, which is why the website/application should be easy for anybody to use.)	Booking appointments, accessing pet information and records online.
<i>System Administrator</i>	High	High	Managing user access, adding and removing users, system updates and security, ensuring pet records are encrypted and safe.
<i>Finance Officer</i>	Medium to high	Medium (A finance officer should know how to navigate various software, according to their job.)	Oversees financial transactions, generates financial reports, and ensures

Veterinary Management System Requirements Specification

		compliance with accounting regulations. Requires seamless integration with payment systems and accounting software for accurate financial tracking.
--	--	--

2.3 Assumptions

- The veterinary clinic will have computers, barcode scanners (for medication scanning and prescriptions), and printers available for use.
- The system will be compatible with **Windows, macOS, and Linux** operating systems.
- If the clinic requires a mobile application, a separate version must be developed for **iOS** and **Android**.
- If deployed as a **cloud-based** system, an **internet connection** is required for features like online booking, cloud storage, and insurance verification.
- For an **on-premises** version, the system should function offline, with local database storage.
- The system assumes that **role-based access control (RBAC)** will be implemented, ensuring that veterinarians, staff, and clients can only access permitted data.

2.4 Constraints and Dependencies

1. Constraints

1.1 System Integration Constraints

- The VMS **must integrate** with third-party services such as **payment gateways, insurance verification systems, and laboratory databases**.
- If external APIs (e.g., **pet insurance providers, pharmacy databases**) are unavailable or changed, certain functionalities may be **delayed or modified**.

1.2 Security and Compliance Constraints

- The system must comply with **data protection laws** to protect patient and pet owner records.
- **Role-based access control (RBAC)** must be enforced to prevent unauthorized access to sensitive medical and financial data.
- The system **must support audit logging** for tracking changes in medical records and financial transactions.

1.3 Performance and Scalability Constraints

- The system **must handle multiple concurrent users**, including veterinarians, assistants, and receptionists.
- **Database performance** should allow fast retrieval of patient records even when storing **thousands of entries**.
- If the clinic expands, the system **should scale** to support **multiple branches** or additional features without major rework.

1.4 User Accessibility Constraints

- The **web portal must be mobile-friendly** to accommodate pet owners booking appointments online.
- The system must be designed for users **with minimal technical expertise**, ensuring an intuitive and user-friendly interface.

Veterinary Management System Requirements Specification

1.5 Hardware and Network Constraints

- The system assumes **reliable internet access** for cloud-based operations. If unavailable, an **offline mode** should be provided for essential functionalities.
- The system should support **barcode scanners** for tracking medications and prescriptions.

1.6 Operational Constraints

- **Clinic operating hours** impact system maintenance schedules—updates should occur **outside peak working hours** to avoid disruptions.
- The system should provide **backup and recovery options** to prevent data loss in case of failures.

2. Dependencies

2.1 Parallel Operation with Legacy Systems

- If the clinic is already using an **older management system**, the VMS may need to **operate in parallel** until full migration is completed.
- Data from the old system **must be imported** into the new VMS without losing nor corrupting patient histories and medical records.

2.2 Dependency on External Modules

- The **appointment scheduling module must be completed** before the notification and reminder system can be developed.
- **Billing and invoicing features depend on the successful integration** of payment gateways.

2.3 Data Synchronization Dependencies

- If the system supports **cloud storage**, data synchronization between local and cloud databases must be handled efficiently.

2.4 Training and Adoption Dependencies

- Staff must undergo **training sessions** before transitioning to the new system.
- Full implementation depends on **clinic staff adoption and feedback**, which may influence additional modifications before deployment.

3. Requirements Era Emi

3.1 Functional Requirements

For the **Veterinary Management System (VMS)**, the most effective way to organize the requirements would be **By User Class** since the system provides different functionalities for veterinarians, receptionists, administrators, and pet owners. Organizing by features could work as well, and will be considered in future drafts of the requirements.

ADMIN FUNCTIONAL REQUIREMENTS

Req#	Requirements	Comments	Priority	Date Reviewed	Reviewed/Approved by
-------------	---------------------	-----------------	-----------------	----------------------	-----------------------------

Veterinary Management System Requirements Specification

<i>ADM_FR_1</i>	The system shall authenticate users based on email and password.	Essential for secure login functionality.	1	10/03/2025	
<i>ADM_FR_2</i>	Administrators shall be able to add new users with specific roles.	New users must be added to manage system access, as well as new employees of any rank.	1	10/03/2025	
<i>ADM_FR_3</i>	Administrators shall be able to delete or deactivate user accounts.	Necessary for handling events where staff leaves, or any other such event occurs where deactivation is necessary.	2	10/03/2025	
<i>ADM_FR_4</i>	Administrators shall assign or revoke user permissions based on roles.	Ensures appropriate access levels for each user class. (For example, a veterinarian can access all patient information, while a receptionist can only see surface level information, such as a patient's name and DOB.)	1	10/03/2025	
<i>ADM_FR_5</i>	Administrators shall be able to generate reports on clinic activity.	Administrators need reports to manage and evaluate the clinic's performance and operations.	3	10/03/2025	
<i>ADM_FR_6</i>	Monitor inventory	Track stock levels of medication.	3	11/03/2025	
<i>ADM_FR_7</i>	Set Pricing and Billing Policies	Adjust prices in coordination with veterinarians and online information.	1	11/03/2025	
<i>ADM_FR_8</i>	Manage Discounts and Promotions	Apply discounts and show promotions as advised by veterinary staff.	3	11/03/2025	
<i>ADM_FR_9</i>	Backup and restore data	Ensure data integrity and make sure it is safe and backed up at all times.	2	11/03/2025	

VETERINARIAN FUNCTIONAL REQUIREMENTS

Req#	Requirements	Comments	Priority	Date Reviewed	Reviewed/Approved by
<i>VET_FR_1</i>	Veterinarians shall be able to view pet medical records.	Important for veterinarians to assess treatment and history of the pets.	1	10/03/2025	
<i>VET_FR_2</i>	Veterinarians shall be able to add treatment and information records to pet medical files.	This is essential for keeping track of treatments, diagnoses, and medication.	1	10/03/2025	
<i>VET_FR_3</i>	Veterinarians shall be able to prescribe medications for pets.	Needed to facilitate treatment processes for pets.	1	10/03/2025	
<i>VET_FR_4</i>	Veterinarians shall be able to schedule follow-up appointments.	If necessary, a veterinarian may schedule an appointment date after finishing a current appointment with a patient.	2	10/03/2025	
<i>VET_FR_5</i>	Access Lab Reports	View and upload test results	2	10/03/2025	
<i>VET_FR_6</i>	Record and Track treatments	Document and track the medication, making sure it is refilled if it runs out.	3	11/03/2025	
<i>VET_FR_7</i>	Generate Health certificates	Provide health certificates for legal or travel purposes.	3	11/03/2025	
<i>VET_FR_8</i>	Communicate with clients	Through means of the software, communicate with the client regarding follow-up appointments or any other cause.	2	11/03/2025	

Veterinary Management System Requirements Specification

<i>VET_FR_9</i>	Refer case to specialists	Cases where the pet might have an unique illness.	3	11/03/2025	
-----------------	---------------------------	---	---	------------	--

RECEPTIONIST FUNCTIONAL REQUIREMENTS

Req#	Requirements	Comments	Priority	Date Reviewed	Reviewed/Approved by
<i>REC_FR_1</i>	Receptionists shall be able to schedule appointments for pets.	Appointment scheduling is a core feature of the system for managing client visits.	1	10/03/2025	
<i>REC_FR_2</i>	Receptionists shall be able to update appointment status (completed, cancelled).	Receptionists need to update the system with appointment status for tracking purposes.	2	10/03/2025	
<i>REC_FR_3</i>	Receptionists shall be able to process payments for services.	Necessary for transaction processing in the clinic.	1	10/03/2025	
<i>REC_FR_4</i>	Check-in and check-out patients	Use the website to track when the client arrives at the appointment.	1	11/03/2025	
<i>REC_FR_5</i>	Handle walk-in patients	Manage unscheduled patients who may need urgent care.	2	11/03/2025	
<i>REC_FR_6</i>	Register new patients	Use the software efficiently to register new patients/pets and their medical history.	1	11/03/2025	
<i>REC_FR_7</i>	Coordinate with veterinarians	Notify the veterinarians in real-time regarding appointments, assign patients to certain veterinarians	1	11/03/2025	

Veterinary Management System Requirements Specification

Finance Officer Functional Requirements

Req#	Requirements	Comments	Priority	Date Reviewed	Reviewed/Approved by
<i>FIN_FR_1</i>	The finance officer shall process invoices to ensure timely payments.	Ensuring invoices are processed efficiently helps maintain healthy cash flow and avoids late payment penalties.	1	24/03/2025	
<i>FIN_FR_2</i>	The finance officer shall track revenue and expenses.	Monitoring revenue and expenses ensures accurate financial reporting, aids in budgeting and forecasting, helps identify financial trends, and supports strategic decision-making by providing a clear picture of the company's financial health.	1	24/03/2025	
<i>FIN_FR_3</i>	The finance officer shall generate financial reports.	Regular financial reporting assists in financial planning, helps management make data-driven decisions, ensures compliance with regulatory requirements, and provides stakeholders with insights into the company's performance and financial stability.	2	24/03/2025	
<i>FIN_FR_4</i>	The finance officer shall prepare, organize, and maintain tax-related documentation in compliance with financial regulations.	Proper tax documentation management is crucial for ensuring compliance with financial laws, avoiding penalties, and facilitating smooth audits. Accurate tax records also help in financial forecasting and decision-making.	2	24/03/2025	
<i>FIN_FR_5</i>	The finance officer shall integrate financial data with accounting software.	Automating financial processes through integration with accounting software improves efficiency, minimizes human error, enhances data security, and ensures real-time financial tracking, making it easier to generate reports and	3	24/03/2025	

Veterinary Management System Requirements Specification

		comply with financial regulations.			
FIN_FR_6	The finance officer shall manage invoice records and track their status.	Keeping an organized record of invoices helps in tracking pending, paid, and overdue invoices.	1	24/03/2025	

PATIENT FUNCTIONAL REQUIREMENTS

Req#	Requirements	Comments	Priority	Date Reviewed	Reviewed/Approved by
PAT_FR_1	Pet owners shall be able to view their pet's medical record.	Owners should have access to their pet's health history for transparency.	1	10/03/2025	
PAT_FR_2	Pet owners shall be able to receive reminders for scheduled appointments.	Automated reminders through various means ensure clients do not miss their appointments.	3	10/03/2025	
PAT_FR_3	Pet owners shall be able to update their contact information (phone number, address).	Required to ensure up-to-date contact records.	2	10/03/2025	
PAT_FR_4	Pet owners shall be able to reschedule or cancel their appointments.	If owners cannot make the appointment, there should be means of cancelling the appointment or rescheduling it.	2	10/03/2025	
PAT_FR_5					
PAT_FR_6					

Veterinary Management System Requirements Specification

3.2 Non-Functional Requirements Erla Laert

ADMIN NON-FUNCTIONAL REQUIREMENTS

Req#	Requirements	Comments	Priority	Date Reviewed	Reviewed/Approved by
<i>ADM_NFR_1</i>	The system shall process administrative tasks (e.g., user management, system updates) within 2 seconds.	Ensures that administrators experience minimal delays when managing system functions	1	10/03/2025	
<i>ADM_NFR_2</i>	The admin dashboard shall have a user-friendly and intuitive design.	Simplifies management tasks and reduces the learning curve for new administrators.	1	10/03/2025	
<i>ADM_NFR_3</i>	The system shall log all administrative actions with accurate timestamps.	Provides a comprehensive audit trail for security and troubleshooting purposes.	1	10/03/2025	
<i>ADM_NFR_4</i>	The system shall support simultaneous administrative sessions without performance degradation.	Allows multiple administrators to work concurrently without slowing down system performance.	2	10/03/2025	
<i>ADM_NFR_5</i>	The system shall be compatible with Windows, macOS, and Linux for administrative tasks.	Ensures accessibility and flexibility for administrators using various platforms.	2		

VETERINARIAN NON-FUNCTIONAL REQUIREMENTS

Req#	Requirements	Comments	Priority	Date Reviewed	Reviewed/Approved by
<i>VET_NFR_1</i>	The system shall load pet medical records within 1 second.	Provides quick access to critical patient data during consultations.	1	10/03/2025	

Veterinary Management System Requirements Specification

<i>VET_NFR_2</i>	The veterinarian interface shall display data in a clear, concise, and	Ensures that vital medical information is easily accessible, even in emergencies.	1	10/03/2025	
------------------	--	---	---	------------	--

Veterinary Management System Requirements Specification

	well-organized format.				
VET_NFR_3	The system shall auto-save treatment records and updates in real time.	Prevents data loss during active sessions and critical operations.	1	10/03/2025	
VET_NFR_4	The system shall maintain an uptime of 99.9% to ensure continuous availability during clinic hours	Guarantees high reliability, especially during high-demand periods or emergencies.	1	10/03/2025	
VET_NFR_5	The system shall enforce multi-factor authentication for veterinarian logins.	Enhances security by ensuring that only authorized personnel can access sensitive patient data.	2		

RECEPTIONIST NON-FUNCTIONAL REQUIREMENTS

Req#	Requirements	Comments	Priority	Date Reviewed	Reviewed/Approved by
REC_FR_1	The appointment scheduling interface shall load within 2 seconds.	Ensures rapid access for receptionists when booking or updating appointments.	1	10/03/2025	
REC_FR_2	The system shall synchronize appointment and payment data in real time.	The system shall synchronize appointment and payment data in real time.	1	10/03/2025	
REC_FR_3	The receptionist interface shall be designed for ease-of-use with minimal training required.	Minimizes errors and accelerates user adoption, especially for less technically skilled staff.	1	10/03/2025	
REC_FR_4	The system shall support concurrent sessions for multiple receptionists.	Allows the clinic to operate smoothly even when several receptionists are active at the same time.	2	10/03/2025	
REC_FR_5	Maintenance and updates shall be scheduled outside of peak clinic hours.	Minimizes disruptions to daily operations and ensures continuity of services during busy periods.	2	10/03/2025	

Veterinary Management System Requirements Specification

PET OWNER NON-FUNCTIONAL REQUIREMENTS

Req#	Requirements	Comments	Priority	Date Reviewed	Reviewed/Approved by
<i>PAT_NFR_1</i>	The pet owner portal shall load within 3 seconds.	Enhances user experience by reducing waiting times, especially on slower connections.	1	10/03/2025	
<i>PAT_NFR_2</i>	The portal shall be mobile-responsive across various devices (smartphones, tablets, desktops).	Ensures a consistent and accessible user experience regardless of the device used.	1	10/03/2025	
<i>PAT_NFR_3</i>	The system shall include accessibility features (e.g., scalable text, screen reader support).	Improves navigation and readability for users with disabilities, ensuring broader usability.	2	10/03/2025	
<i>PAT_NFR_4</i>	All pet owner data shall be encrypted during transmission and storage.	Protects sensitive personal and pet-related information from unauthorized access.	1	10/03/2025	
<i>PAT_NFR_5</i>	The interface shall allow new users to learn basic navigation and functionalities within one minute.	Enhances user adoption by ensuring that even non-technical pet owners can quickly become comfortable with the system.	2	10/03/2025	

Veterinary Management System Requirements Specification
FINANCE OFFICER NON-FUNCTIONAL REQUIREMENTS

Req#	Requirements	Comments	Priority	Date Reviewed	Reviewed/Approved by
<i>FIN_NFR_1</i>	The system shall generate financial reports within 3 seconds.	Improves efficiency.	1		
<i>FIN_NFR_2</i>	Financial data shall be encrypted in storage and transmission.	Ensures security.	1		
<i>FIN_NFR_3</i>	The finance module shall support role-based access.	Prevents unauthorized changes.	2		
<i>FIN_NFR_4</i>	The finance officer dashboard shall provide real-time updates.	Ensures accurate monitoring.	2		
<i>FIN_NFR_5</i>	The system shall support financial auditing tools.	Ensures compliance with regulations.	3		

Product Requirements Era Nadir

Client Registration

The system shall allow users to register clients by entering their personal details, such as full name, address, contact information, and preferred method of communication. The system should also support updating and deleting client information.

Pet Registration

Each pet shall be linked to a registered client. The system shall capture pet details such as name, species, breed, gender, age, medical history, vaccination records, and microchip number (if available).

Appointment Scheduling

Users shall be able to schedule, reschedule, or cancel appointments. The system shall send automated reminders to clients through SMS or email. Additionally, it shall prevent scheduling conflicts by checking for available time slots.

Medical Records Management

The system shall maintain a complete medical history for each pet, including previous illnesses, surgeries, treatments, and prescribed medications. Veterinary staff shall be able to update and retrieve medical records as needed.

Billing and Invoicing

The system shall generate invoices for veterinary services and products purchased. It should support different payment methods, including cash, credit/debit cards, and online payments. Users shall also be able to track outstanding balances and payment history.

3.2.1.1 Usability Requirements

Include any specific usability requirements, for example,

- Learnability
- The user documentation and help should be complete
- The system should be easy to learn

3.2.1.2 Performance Requirements

Static Numerical Requirements:

- The system shall support up to 25 concurrent terminals within a veterinary clinic.
- The system shall support at least 100 simultaneous users, including veterinarians, receptionists, and administrative staff.

Veterinary Management System Requirements Specification

- The system shall handle data related to:
 - 10,000 animal records
 - 5,000 client profiles
 - 20,000 appointment entries
 - 3,000 inventory items including medicine and equipment.
- Real-time access to digital medical histories, prescriptions, and vaccination schedules.

Dynamic Numerical Requirements:

- The system shall be capable of processing 90% of standard transactions (e.g., appointment booking, record lookup) in under 1 second during normal conditions.
- Under peak load (e.g., start of day or emergency influx), the system shall process at least 80% of transactions in under 2 seconds.
- The system shall handle at least 500 transactions per hour during normal operation.
- The database shall be able to store daily logs of operations for a minimum of 3 years before archiving is required.
- Daily backups must complete within 15 minutes, without interrupting system operations.

3.2.1.3 Availability

- **System Availability:**

The Veterinary Management System shall be available 99.5% of the time, equivalent to a maximum downtime of approximately 3.65 hours per month.

- **Geographical Coverage:**

The system shall support multiple branches of a veterinary network, ensuring seamless synchronization and access in different geographic regions, including rural and urban clinics.

- **Impact of Downtime:**

Downtime during working hours will negatively impact appointment scheduling, record access, and billing, leading to delays in patient treatment.

- In case of system failure, critical operations must switch to read-only offline mode within 2 minutes, using cached data.

- **Maintenance Impact:**

Scheduled maintenance shall be communicated to users at least 48 hours in advance via in-system notifications and email.

- Maintenance tasks must be scheduled during off-peak hours (e.g., weekends or 10 PM–5 AM).

Unscheduled maintenance must trigger an automatic failover system within 30 seconds to ensure continuity of service.

- **Reliability:**

- The system must have a Mean Time Between Failures (MTBF) of at least 1,000 hours.

The system must not experience more than 2 failures per month, and each failure must be logged and analyzed.

- A self-monitoring tool shall be integrated to log uptime and send alerts in case of abnormal behavior.

3.2.1.4 Security

Ensuring confidentiality, integrity, and availability of data within the Veterinary Management System (VMS) is a top priority. The following security measures will be implemented to safeguard sensitive information and maintain system reliability:

1. Authentication and Access Control

- The system will enforce **Role-Based Access Control (RBAC)**, ensuring that veterinarians, receptionists, pet owners, finance officers, and administrators have appropriate access levels.

Veterinary Management System Requirements Specification

- **Multi-Factor Authentication (MFA)** will be required for high-privilege users, including veterinarians, administrators, and finance officers, to enhance security.
- Strong password policies will be enforced, requiring a minimum of 12 characters, a mix of uppercase and lowercase letters, numbers, and special characters.

2. Data Encryption and Protection

- All sensitive data, including pet medical records and financial transactions, will be encrypted both in transit (TLS 1.2 or higher) and at rest (AES-256 encryption).
- Payment transactions will comply with **PCI DSS** standards to ensure secure processing.

3. Audit Logging and Monitoring

- A comprehensive audit trail will be maintained, logging all critical activities such as user login, modifications to medical records, and financial transactions.
- Logs will be immutable and securely stored for a minimum of five years to support security audits and compliance.

4. Data Integrity and Backup Procedures

- Automated daily backups will be implemented, with encrypted offsite and cloud-based storage to prevent data loss.
- Regular integrity checks will be conducted to detect and prevent data corruption.

5. Threat Detection and Prevention

- An **Intrusion Detection System (IDS) and Intrusion Prevention System (IPS)** will be employed to identify and mitigate security threats in real-time.
- The system will automatically lock user accounts after five failed login attempts, alerting administrators to potential security breaches.

6. Regulatory Compliance

- The VMS will comply with **GDPR, HIPAA** (if applicable), and other relevant data protection regulations.
- Regular security audits and vulnerability assessments will be conducted to ensure continued compliance and protection.

3.2.1 Organizational Requirements

To ensure the efficient and seamless operation of the Veterinary Management System, the following organizational policies and procedures will be followed:

1. Process Standards and Best Practices

- The system will be developed and maintained using **Agile** and **DevOps** methodologies, ensuring continuous improvement and timely updates.
- Regular system updates and security patches will be scheduled outside peak business hours to minimize disruption.

2. User Training and Support

- All clinic staff, including veterinarians, receptionists, and finance officers, will undergo comprehensive training on system usage and data security best practices.
- A dedicated support team will be available to provide technical assistance and troubleshoot issues as needed.

3. Operational Guidelines

- Standard Operating Procedures (SOPs) will be established for managing pet records, scheduling appointments, and handling financial transactions.

Veterinary Management System Requirements Specification

- Strict confidentiality agreements will be enforced to protect sensitive client and patient information.
- 4. **Change Management**
 - Any changes or updates to the system will undergo rigorous testing and approval before deployment.
 - A structured rollback plan will be in place to quickly restore previous system versions in case of unforeseen issues.

3.2.2 External Requirements

The Veterinary Management System must integrate with external systems and comply with industry standards to ensure seamless operations and regulatory adherence.

- 1. **Compliance with Legal and Industry Standards**
 - The system will align with regulatory requirements such as:
 - **GDPR** (General Data Protection Regulation) for handling user data in Europe.
 - **HIPAA** (Health Insurance Portability and Accountability Act) for managing medical data in the U.S.
 - Local veterinary board regulations governing record-keeping and confidentiality.
- 2. **Interoperability and System Integration**
 - The VMS will integrate with third-party services, including:
 - **Payment Gateways** (e.g., PayPal, Stripe, Square) for secure online transactions.
 - **Pharmacy Databases** for real-time medication availability and prescription tracking.
 - **Insurance Systems** to verify pet insurance coverage and streamline claims processing.
 - **Laboratory Information Systems** for seamless retrieval of diagnostic test results.
- 3. **Third-Party Dependencies and Data Synchronization**
 - The system will support seamless integration with existing veterinary hospital management software and facilitate data migration.
 - All external API integrations will adhere to standardized formats (e.g., **RESTful APIs, FHIR**) to ensure compatibility and scalability.
- 4. **Performance and Service Level Agreements (SLAs)**
 - External service providers, including cloud hosting and payment processing solutions, will be required to maintain a minimum **99.9% uptime**.
 - Automated fallback mechanisms will be in place to minimize disruptions caused by third-party service outages.

4. User Scenarios/Use Cases

Use Case 1 (Enea Cane 01-04)

UC Name	UC-ADM-01: Authenticate User
Summary	System Administrator logs into the system securely.
Dependency	None.
Actors	System Administrator (Primary).
Preconditions	1. Administrator has valid credentials. 2. System is operational.
Description of the Main Sequence	<ul style="list-style-type: none"> ● Step 1: Admin navigates to the login page. ● Step 2: Enters email and password. ● Step 3: System validates credentials. ● Step 4: System grants access to the admin dashboard.
Description of the Alternative Sequence	<ul style="list-style-type: none"> ● Step 1: System detects invalid email/password. ● Step 2: Displays error message and prompts re-entry.
Non-functional Requirements	<ul style="list-style-type: none"> - Authentication completes within 2 seconds (ADM_NFR_1). - Multi-factor authentication optional (VET_NFR_5).
Postconditions	Admin is logged into the system.

Use Case 2

UC Name	UC-ADM-02: Add New User
Summary	Admin creates a new user account with assigned role.
Dependency	UC-ADM-01 (Admin must be authenticated).
Actors	System Administrator (Primary).
Preconditions	1. Admin is logged in. 2. New user details are available.
Description of the Main Sequence	<ul style="list-style-type: none"> ● Step 1: Admin selects "Add User." ● Step 2: Enters user's name, email, and role. ● Step 3: System validates email uniqueness. ● Step 4: Sends temporary password to the user's email.
Description of the Alternative Sequence	<ul style="list-style-type: none"> ● Step 1: System detects duplicate email. ● Step 2: Alerts admin to use a different email.
Non-functional Requirements	- User creation completes within 5 seconds (ADM_NFR_1).
Postconditions	New user account is created and visible in the user list.

Use Case 3

UC Name	UC-ADM-03: Deactivate User
Summary	Admin deactivates a user account (e.g., staff resignation).
Dependency	UC-ADM-01.
Actors	System Administrator.
Preconditions	<ol style="list-style-type: none">1. Admin is logged in.2. Target user exists in the system.
Description of the Main Sequence	<ul style="list-style-type: none">• Step 1: Admin searches for the user.• Step 2: Selects "Deactivate Account."• Step 3: System revokes user access.
Description of the Alternative Sequence	<ul style="list-style-type: none">• Step 1: System displays "User not found" error.
Non-functional Requirements	- Audit log entry created (ADM_NFR_3).
Postconditions	User account is deactivated and cannot log in.

Use Case 4

UC Name	UC-ADM-04: Generate Clinic Activity Report
Summary	Admin generates a report on appointments, payments, and inventory.
Dependency	UC-ADM-01.
Actors	System Administrator.
Preconditions	1. Admin is logged in. 2. Data exists for the selected time period.
Description of the Main Sequence	<ul style="list-style-type: none"> ● Step 1: Admin selects "Generate Report" and specifies date range. ● Step 2: System compiles data into a PDF/Excel file. ● Step 3: Admin downloads or prints the report.
Description of the Alternative Sequence	<ul style="list-style-type: none"> ● Step 1: System displays "No records found" message.
Non-functional Requirements	- Report generation completes within 3 seconds (ADM_NFR_1).
Postconditions	Report is saved to the admin's device.

Use Case 5 (Laert 05-09)

UC Name	UC-VET-01: View Pet Medical Record
Summary	Veterinarian accesses a pet's full medical history.
Dependency	UC-SYS-01 (User must be logged in).
Actors	Veterinarian (Primary).
Preconditions	1. Veterinarian is logged in. 2. Pet is registered in the system.
Description of the Main Sequence	<ul style="list-style-type: none"> ● Step 1: Vet searches for the pet by name/ID. ● Step 2: System displays medical history.
Description of the Alternative Sequence	<ul style="list-style-type: none"> ● Step 1: System alerts "No records found."
Non-functional Requirements	- Records load within 1 second (VET_NFR_1).
Postconditions	Vet views the pet's medical record.

Use Case 6

UC Name	UC-VET-02: Prescribe Medication
Summary	Vet prescribes medication via integration with pharmacy databases.
Dependency	UC-VET-01 (Medical record must be open).
Actors	Veterinarian, Pharmacy Database (Secondary).
Preconditions	1. Vet is logged in. 2. Medication is available in the pharmacy database.
Description of the Main Sequence	<ul style="list-style-type: none"> ● Step 1: Vet selects "Prescribe Medication" in the pet's record. ● Step 2: Enters medication name, dosage, and duration. ● Step 3: System checks pharmacy database. ● Step 4: Prescription is sent to the pharmacy.
Description of the Alternative Sequence	<ul style="list-style-type: none"> ● Step 1: System alerts vet to choose an alternative.
Non-functional Requirements	- Prescription data encrypted during transmission (PAT_NFR_4).
Postconditions	Prescription is recorded and sent to the pharmacy.

Use Case 7

UC Name	UC-VET-03: Schedule Follow-Up Appointment
Summary	Vet schedules a follow-up after a consultation.
Dependency	UC-VET-01.
Actors	Veterinarian.
Preconditions	1. Vet is logged in. 2. Pet has an active appointment.
Description of the Main Sequence	<ul style="list-style-type: none"> ● Step 1: Vet selects "Schedule Follow-Up." ● Step 2: Chooses date/time and confirms. ● Step 3: System updates the calendar and notifies the pet owner.
Description of the Alternative Sequence	<ul style="list-style-type: none"> ● Step 1: System suggests alternative slots.
Non-functional Requirements	- Appointment reminders sent via SMS/email (PAT_FR_2).
Postconditions	Follow-up appointment is added to the calendar.

Use Case 8

UC Nam	UC-REC-01: Schedule Appointment
Summary	Receptionist books an appointment for a pet.
Dependency	UC-SYS-01.
Actors	Receptionist, Pet Owner (Secondary).
Preconditions	1. Receptionist is logged in. 2. Pet owner's contact details are registered.
Description of the Main Sequence	<ul style="list-style-type: none"> ● Step 1: Receptionist selects "New Appointment." ● Step 2: Enters pet owner's name, pet's name, and reason for visit. ● Step 3: System checks vet availability. ● Step 4: Appointment confirmation is sent.
Description of the Alternative Sequence	<ul style="list-style-type: none"> ● Step 1: System suggests next available slot.
Non-functional Requirements	- Interface loads within 2 seconds (REC_FR_1).
Postconditions	Appointment is added to the calendar.

Use Case 9 (Erla 09-12)

UC Name	UC-REC-02: Process Payment
Summary	Receptionist processes payment for services.
Dependency	UC-REC-01 (Appointment must exist).
Actors	Receptionist, Payment Gateway (Secondary).
Preconditions	1. Appointment is marked "Completed." 2. Invoice is generated.
Description of the Main Sequence	<ul style="list-style-type: none"> ● Step 1: Receptionist selects "Process Payment." ● Step 2: Enters payment method (cash/card). ● Step 3: System connects to payment gateway. ● Step 4: Receipt is printed/mailed.
Description of the Alternative Sequence	<ul style="list-style-type: none"> ● Step 1: System alerts receptionist to retry or use another method.
Non-functional Requirements	- Payment processing within 5 seconds (REC_FR_2).
Postconditions	Invoice status is updated to "Paid."

Use Case 10

UC Name	UC-REC-03: Register New Patient
Summary	Receptionist adds a new pet and owner to the system.
Dependency	None.
Actors	Receptionist.
Preconditions	1. Receptionist is logged in. 2. Pet owner provides valid ID and contact info.
Description of the Main Sequence	<ul style="list-style-type: none"> ● Step 1: Receptionist selects "Register New Patient." ● Step 2: Enters owner's name, phone, address, and pet's details. ● Step 3: System assigns a unique patient ID.
Description of the Alternative Sequence	<ul style="list-style-type: none"> ● Step 1: System highlights missing fields (e.g., pet's breed).
Non-functional Requirements	- Mobile-responsive form (PAT_NFR_2).
Postconditions	Pet and owner are registered in the system.

Use Case 11

UC Name	UC-PAT-01: Reschedule Appointment
Summary	Pet owner changes an existing appointment.
Dependency	UC-SYS-01 (Owner must be logged in).
Actors	Pet Owner.
Preconditions	1. Appointment is scheduled. 2. Rescheduling is allowed (>24 hours before).
Description of the Main Sequence	<ul style="list-style-type: none"> ● Step 1: Owner logs into the portal and selects "My Appointments." ● Step 2: Chooses an appointment and selects "Reschedule." ● Step 3: Picks a new slot and confirms. ● Step 4: System updates the calendar.
Description of the Alternative Sequence	<ul style="list-style-type: none"> ● Step 1: System charges a fee if <24 hours' notice.
Non-functional Requirements	- Portal loads within 3 seconds (PAT_NFR_1).
Postconditions	Appointment is updated in the system.

Use Case 12

UC Name	UC-PAT-02: View Pet Health Certificate
Summary	Pet owner accesses a health certificate for travel.
Dependency	UC-VET-07 (Vet must generate the certificate).
Actors	Pet Owner.
Preconditions	<ol style="list-style-type: none">1. Certificate is issued by a vet.2. Owner is logged in.
Description of the Main Sequence	<ul style="list-style-type: none">● Step 1: Owner navigates to "Medical Records."● Step 2: Selects "Health Certificates."● Step 3: Downloads/prints the PDF certificate.
Description of the Alternative Sequence	<ul style="list-style-type: none">● Step 1: System alerts owner to schedule a vet visit.
Non-functional Requirements	- Certificate PDF is watermarked for security (PAT_NFR_4).
Postconditions	Owner obtains the health certificate.

Use Case 13 (Eridjon 13-16)

UC Name	UC-FIN-01: Generate Financial Report
Summary	Finance Officer creates a monthly revenue report.
Dependency	UC-SYS-01.
Actors	Finance Officer.
Preconditions	1. Officer is logged in. 2. Financial data exists for the selected period.
Description of the Main Sequence	<ul style="list-style-type: none"> ● Step 1: Officer selects "Generate Financial Report." ● Step 2: System compiles data into a CSV/PDF file. ● Step 3: Officer exports the financial report.
Description of the Alternative Sequence	<ul style="list-style-type: none"> ● Step 1: System displays "No financial data available."
Non-functional Requirements	- Reports encrypted during export (FIN_NFR_2).
Postconditions	Report is saved and ready for auditing.

Use Case 14

UC Name	UC-FIN-02: Reconcile Payments
Summary	Finance Officer matches payments with invoices.
Dependency	UC-REC-02 (Payments must be processed).
Actors	Finance Officer.
Preconditions	1. Officer is logged in. 2. Daily transactions are completed.
Description of the Main Sequence	<ul style="list-style-type: none"> ● Step 1: Officer selects "Reconcile Payments." ● Step 2: System cross-checks invoices and payment records. ● Step 3: Flags discrepancies.
Description of the Alternative Sequence	<ul style="list-style-type: none"> ● Step 1: System highlights mismatches for manual review.
Non-functional Requirements	<ul style="list-style-type: none"> - Real-time data sync with payment gateways (REC_FR_2).
Postconditions	All payments are reconciled.

Use Case 15

UC Name	UC-SYS-01: Backup Database
Summary	System Administrator performs a nightly backup.
Dependency	UC-ADM-01.
Actors	System Administrator.
Preconditions	1. Backup storage is available (cloud/local).
Description of the Main Sequence	<ul style="list-style-type: none">● Step 1: System triggers automatic backup at 2:00 AM.● Step 2: Data is encrypted and copied.● Step 3: Admin receives confirmation email.
Description of the Alternative Sequence	<ul style="list-style-type: none">● Step 1: System retries after 15 minutes.
Non-functional Requirements	- 99.9% uptime (VET_NFR_4).
Postconditions	Latest database backup is stored securely.

Use Case 16

UC Name	UC-SYS-02: Restore Data
Summary	Admin restores data from a backup after a failure.
Dependency	UC-SYS-01 (Backup must exist).
Actors	System Administrator.
Preconditions	1. Backup file is available.
Description of the Main Sequence	<ul style="list-style-type: none">● Step 1: Admin selects "Restore Backup" and chooses a date.● Step 2: System validates backup integrity.● Step 3: Data is restored.
Description of the Alternative Sequence	<ul style="list-style-type: none">● Step 1: System alerts admin to use an older backup.
Non-functional Requirements	- Restoration completes within 5 minutes (UC-SYS-02 Success Metric).
Postconditions	System returns to the state at the backup time.

Use Case 17 (Emi 17-20)

UC Name	UC-LAB-01: Upload Lab Results
Summary	Lab technician uploads diagnostic test results.
Dependency	UC-VET-01 (Pet record must exist).
Actors	Lab Technician (Secondary).
Preconditions	1. Lab test is completed. 2. Results are in digital format.
Description of the Main Sequence	<ul style="list-style-type: none"> ● Step 1: Technician logs into the lab portal. ● Step 2: Selects the pet's record and uploads the file. ● Step 3: System notifies the vet via email.
Description of the Alternative Sequence	<ul style="list-style-type: none"> ● Step 1: System compresses the file before uploading.
Non-functional Requirements	- File encryption during upload (PAT_NFR_4).
Postconditions	Lab results are attached to the pet's medical record.

Use Case 18

UC Name	UC-INV-01: Reorder Medication
Summary	System triggers automatic medication reorder.
Dependency	UC-ADM-06 (Inventory tracking enabled).
Actors	System (Primary), Pharmacy Database (Secondary).
Preconditions	1. Medication stock falls below the threshold.
Description of the Main Sequence	<ul style="list-style-type: none">● Step 1: System detects low stock.● Step 2: Automatically sends a reorder request.● Step 3: Pharmacy confirms shipment.
Description of the Alternative Sequence	<ul style="list-style-type: none">● Step 1: System alerts the admin to contact another supplier.
Non-functional Requirements	- Real-time inventory sync (ADM_FR_6).
Postconditions	Medication is restocked.

Use Case 19

UC Name	UC-EMG-01: Triage Emergency Case
Summary	Receptionist prioritizes a critical patient.
Dependency	UC-REC-03 (Walk-in handling).
Actors	Receptionist, Veterinarian.
Preconditions	1. Emergency patient arrives at the clinic.
Description of the Main Sequence	<ul style="list-style-type: none">● Step 1: Receptionist selects "Emergency Triage."● Step 2: System alerts all available vets.● Step 3: Vet accepts the case.
Description of the Alternative Sequence	<ul style="list-style-type: none">● Step 1: System suggests nearby emergency clinics.
Non-functional Requirements	- Notifications sent in <10 seconds (REC_FR_3).
Postconditions	Emergency case is assigned to a vet.

Use Case 20

UC Name	UC-COM-01: Send Appointment Reminder
Summary	System sends automated reminders to pet owners.
Dependency	UC-REC-01 (Appointment must exist).
Actors	System (Primary), Pet Owner (Secondary).
Preconditions	1. Appointment is scheduled. 2. Reminder is due (24 hours before).
Description of the Main Sequence	<ul style="list-style-type: none"> ● Step 1: System checks the calendar daily. ● Step 2: Sends SMS/email reminder.
Description of the Alternative Sequence	<ul style="list-style-type: none"> ● Step 1: System skips reminders if owner unsubscribes.
Non-functional Requirements	- 95% reminders delivered successfully (PAT_FR_2).
Postconditions	Owner receives the reminder.

Use Case 21 (Nadir 21-24)

Section	Details
UC Name	UC-003 Cancel an Appointment
Summary	Allows users to cancel or reschedule an existing appointment.
Dependency	UC-002 (requires a booked appointment).
Actors	Primary: Pet Owner Secondary: Receptionist
Preconditions	- The appointment exists in the system. - The user is authenticated.
Description of the Main Sequence	1. The user navigates to the "My Appointments" section. 2. The user selects the appointment to cancel. 3. The system displays a cancellation confirmation dialog. 4. The user confirms cancellation. 5. The system updates the appointment status to "Cancelled" and frees the slot. 6. The system notifies the clinic staff and updates the pet owner via email/SMS.
Description of the Alternative Sequence	- If the cancellation occurs within 24 hours of the appointment: - The system applies a cancellation fee (e.g., 20% of service cost). - The user must confirm acceptance of the fee before proceeding. - The fee is added to the pet owner's invoice.
Non-functional requirements	- Immediate update of appointment status to prevent double-booking. - Cancellation fee calculations must comply with clinic policies.
Postconditions	- The appointment is removed from the schedule. - The slot becomes available for other users.

Use Case 22

Section	Details
UC Name	UC-007 Assign a Veterinarian to a Case
Summary	Assigns a veterinarian to handle a specific pet's medical case.
Dependency	UC-001 (registered pet), UC-005 (medical history accessible).
Actors	Primary: Receptionist Secondary: Administrator
Preconditions	- The pet has a diagnosed condition requiring specialized care. - At least one veterinarian is listed in the system.
Description of the Main Sequence	1. The receptionist selects the pet's case from the "Active Cases" list. 2. The system displays available veterinarians with relevant expertise (e.g., surgery, dermatology). 3. The receptionist assigns a vet and selects a priority level (e.g., urgent, routine). 4. The system updates the vet's schedule and sends a notification. 5. The vet acknowledges the assignment via the system.
Description of the Alternative Sequence	- If no vets are available: - The system suggests the nearest available slot (e.g., next 48 hours). - The receptionist negotiates with the pet owner for rescheduling.
Non-functional requirements	- Notifications to vets must include case priority tags (e.g., red for emergency). - Vet schedules must refresh in real-time to prevent overbooking.
Postconditions	- The vet is assigned to the case. - The pet owner receives an updated treatment plan.

Use Case 23

Section	Details
UC Name	UC-008 Manage User Roles & Permissions
Summary	Modifies user access levels and permissions within the system.
Dependency	None.
Actors	Primary: Administrator
Preconditions	- The administrator has superuser privileges. - The target user exists in the system.
Description of the Main Sequence	1. The administrator navigates to the "User Management" dashboard. 2. The system displays a list of all users with their current roles. 3. The administrator selects a user and edits their permissions (e.g., grant billing access, revoke edit rights). 4. The administrator confirms the changes. 5. The system logs the modification in the audit trail. 6. The affected user receives an email notification about the update.
Description of the Alternative Sequence	- If the user does not exist: - The system displays: "User not found. Check the email or ID." - The administrator cancels the action or creates a new user (UC-001).
Non-functional requirements	- Role changes must be applied immediately with no system downtime. - Audit logs must record IP address, timestamp, and admin ID for each change.
Postconditions	- The user's permissions are updated. - An audit entry is created.

Use Case 24

Section	Details
UC Name	UC-013 Schedule Surgery
Summary	Books a surgical procedure for a pet and allocates resources.
Dependency	UC-001 (registered pet), UC-005 (medical history reviewed).
Actors	Primary: Veterinarian
Preconditions	<ul style="list-style-type: none"> - The pet requires surgery (diagnosed via UC-011). - Operating room availability is confirmed.
Description of the Main Sequence	<ol style="list-style-type: none"> 1. The veterinarian selects the pet's profile. 2. The system displays surgical prerequisites (e.g., fasting, pre-op tests). 3. The veterinarian enters: <ul style="list-style-type: none"> - Surgery type (e.g., "Spay," "Tumor removal") - Anesthesia plan - Surgeon and assistant assignments 4. The system checks for scheduling conflicts and reserves the OR. 5. The pet owner receives a pre-surgery checklist via email/SMS.
Description of the Alternative Sequence	<ul style="list-style-type: none"> - If no operating rooms are available: <ul style="list-style-type: none"> - The system suggests the next three available slots. - The veterinarian consults the owner to reschedule.
Non-functional requirements	<ul style="list-style-type: none"> - OR schedules must sync with staff shift rotations. - Pre-op checklists must be printable and digitally signable.
Postconditions	<ul style="list-style-type: none"> - The surgery is added to the clinic's calendar. - Pre-op instructions are sent to the owner.

Use Case 25 (Era 25-29)

Section	Details
UC Name	UC-016 Verify Pet Insurance
Summary	Validates a pet's insurance coverage for billing or treatment purposes.
Dependency	UC-001 (registered pet), UC-004 (payment processing).
Actors	Primary: Receptionist
Preconditions	<ul style="list-style-type: none"> - The pet owner provides insurance policy details. - The clinic has API access to insurance providers.
Description of the Main Sequence	<ol style="list-style-type: none"> 1. The receptionist selects the pet's profile. 2. The system displays the "Verify Insurance" option. 3. The receptionist enters: <ul style="list-style-type: none"> - Policy number - Insurance provider - Policyholder's name 4. The system connects to the provider's API to validate coverage. 5. The system displays coverage details (e.g., deductible, covered services). 6. The pet owner is informed of the verification result.
Description of the Alternative Sequence	<ul style="list-style-type: none"> - If the policy is expired or invalid: <ul style="list-style-type: none"> - The system flags the policy as "Unverified." - The receptionist notifies the pet owner to update or provide alternative payment.
Non-functional requirements	<ul style="list-style-type: none"> - Insurance API responses must occur within 8 seconds. - Policy data must be encrypted during transmission (TLS 1.3).
Postconditions	<ul style="list-style-type: none"> - Insurance status is recorded in the pet's billing record. - The pet owner is aware of coverage limitations.

Use Case 26

Section	Details
UC Name	UC-020 Process Refunds
Summary	Handles reimbursement requests for overpayments or canceled services.
Dependency	UC-004 (payment records), UC-003 (canceled appointments).
Actors	Primary: Finance Officer
Preconditions	<ul style="list-style-type: none"> - A refund request is submitted (e.g., via form, email). - The original payment is verified in the system.
Description of the Main Sequence	<ol style="list-style-type: none"> 1. The finance officer retrieves the refund request from the queue. 2. The system displays the original transaction details: <ul style="list-style-type: none"> - Payment method - Amount paid - Service date 3. The officer approves the refund and selects a method: <ul style="list-style-type: none"> - Credit card reversal - Check issuance - Cash (in-clinic only) 4. The system updates the payment status to "Refunded" and logs the action. 5. The pet owner receives a confirmation and ETA for the refund.
Description of the Alternative Sequence	<ul style="list-style-type: none"> - If the payment method is invalid (e.g., expired card): <ul style="list-style-type: none"> - The system notifies the officer to contact the pet owner for alternate details. - The refund is paused until resolved.
Non-functional requirements	<ul style="list-style-type: none"> - Refunds must comply with PCI-DSS and GDPR standards. - Refund status must be visible to pet owners via the portal.
Postconditions	<ul style="list-style-type: none"> - The refund is processed and recorded. - The pet owner is notified of completion.

Use Case 27

Section	Details
UC Name	UC-021 Request Home Visit
Summary	Enables pet owners to schedule a veterinarian home visit for their pet.
Dependency	UC-001 (registered pet), UC-007 (vet availability).
Actors	Primary: Pet Owner
Preconditions	<ul style="list-style-type: none"> - The clinic offers home visit services. - The pet owner's location is within the clinic's service radius.
Description of the Main Sequence	<ol style="list-style-type: none"> 1. The pet owner selects "Request Home Visit" in the system. 2. The system displays available dates/times and eligible veterinarians. 3. The owner selects a slot and provides: <ul style="list-style-type: none"> - Visit reason (e.g., "Check-up," "Vaccination") - Address with geolocation pin 4. The system assigns a vet based on proximity and expertise. 5. The system sends a confirmation to the owner and vet.
Description of the Alternative Sequence	<ul style="list-style-type: none"> - If no vets are available for the selected date: <ul style="list-style-type: none"> - The system suggests alternative slots within the next 72 hours. - The owner chooses a new slot or cancels the request.
Non-functional requirements	<ul style="list-style-type: none"> - Geolocation tracking must be accurate within 100 meters. - Vet assignments prioritize proximity and workload balance.
Postconditions	<ul style="list-style-type: none"> - The home visit is scheduled. - The vet receives the assignment details.

Use Case 28

Section	Details
UC Name	UC-023 Rate and Review Services
Summary	Allows pet owners to submit feedback on clinic services.
Dependency	UC-002 (booked appointment), UC-004 (completed payment).
Actors	Primary: Pet Owner
Preconditions	<ul style="list-style-type: none"> - The pet owner has used at least one clinic service. - The service was completed within the last 30 days.
Description of the Main Sequence	<ol style="list-style-type: none"> 1. The owner navigates to the "Review" section. 2. The system displays a form with: <ul style="list-style-type: none"> - Star rating (1–5 stars) - Free-text comments (500-character limit) - Optional photo upload (e.g., pet recovery) 3. The owner submits the review. 4. The system publishes it after moderation.
Description of the Alternative Sequence	<ul style="list-style-type: none"> - If the review contains inappropriate language: <ul style="list-style-type: none"> - The system flags it for manual moderation. - The clinic staff reviews and approves/rejects it.
Non-functional requirements	<ul style="list-style-type: none"> - Reviews must be anonymized for public display (owner name hidden). - Moderation alerts resolved within 24 hours.
Postconditions	<ul style="list-style-type: none"> - The review is published on the clinic's platform. - The clinic staff can respond via the system.

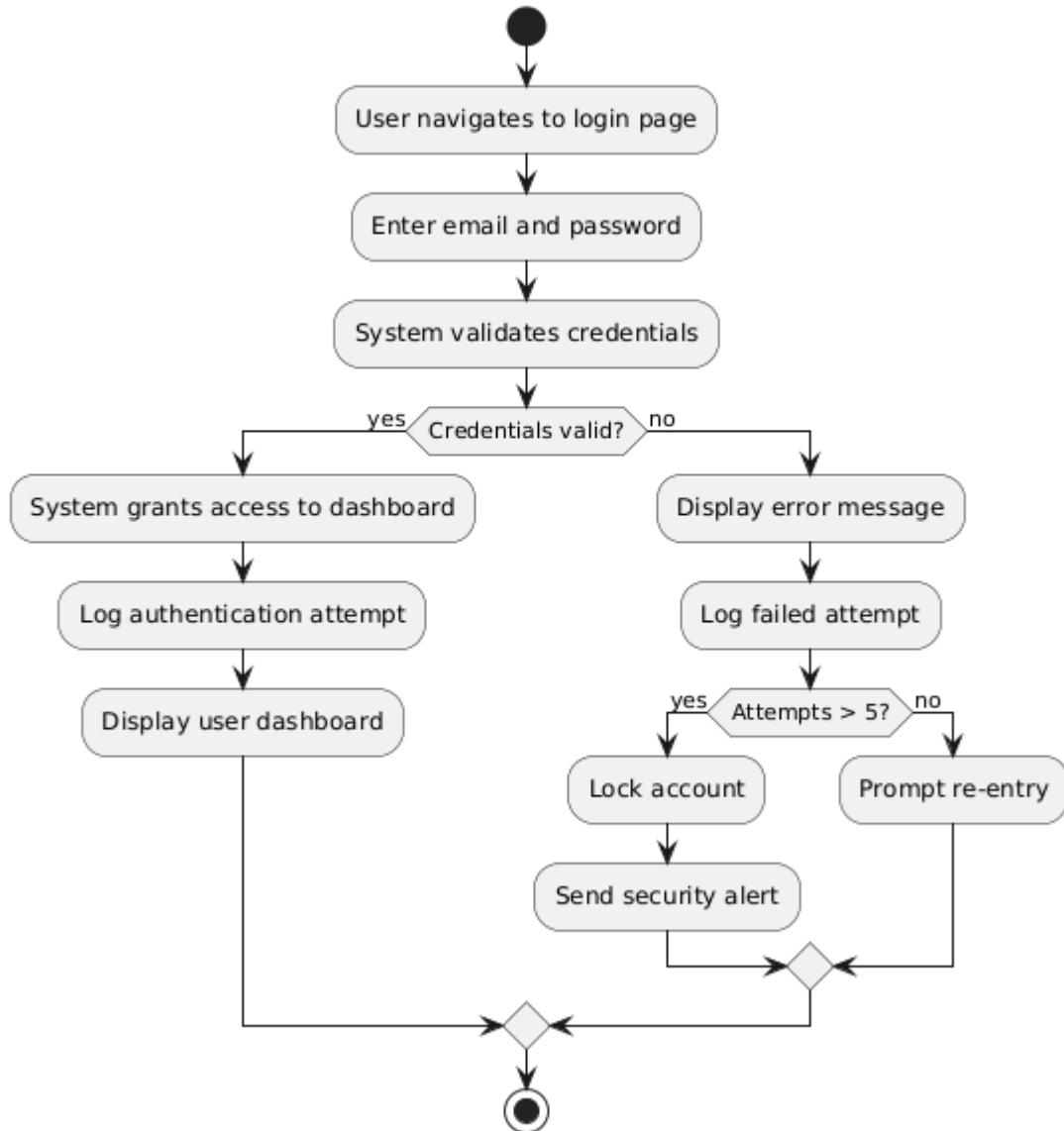
Use Case 29

Section	Details
UC Name	UC-023 Track Pet Microchip Information
Summary	Allows pet owners to view and manage microchip details for their pets.
Dependency	UC-006 (Vaccination records), UC-011 (Medical history).
Actors	Primary: Pet Owner
Preconditions	<ul style="list-style-type: none"> -The pet has a microchip registered in the system. - The pet owner has viewing permissions.
Description of the Main Sequence	<ol style="list-style-type: none"> 1. The pet owner navigates to the pet's profile. 2. The system displays: <ul style="list-style-type: none"> • Microchip ID • Microchip registration date • Microchip provider • Contact details of the microchip registry (if available) 3. The pet owner can view and update the microchip information, such as changing the contact details associated with the microchip. 4. The system provides an option for the owner to receive an SMS/email reminder if the microchip registration details are outdated or need to be updated.
Description of the Alternative Sequence	<ul style="list-style-type: none"> • If no microchip details are recorded: <ul style="list-style-type: none"> • The system displays: "No microchip information found. Would you like to register a microchip?" • The owner is redirected to UC-REC-01 (Schedule an Appointment) to schedule a microchip implantation appointment.
Non-functional Requirements	<ul style="list-style-type: none"> • Data must comply with local and international pet registration standards. • Microchip information must be updated in real-time.
Postconditions	<ul style="list-style-type: none"> • The pet owner is informed of the microchip registration details. • Reminder notifications are activated (if selected).

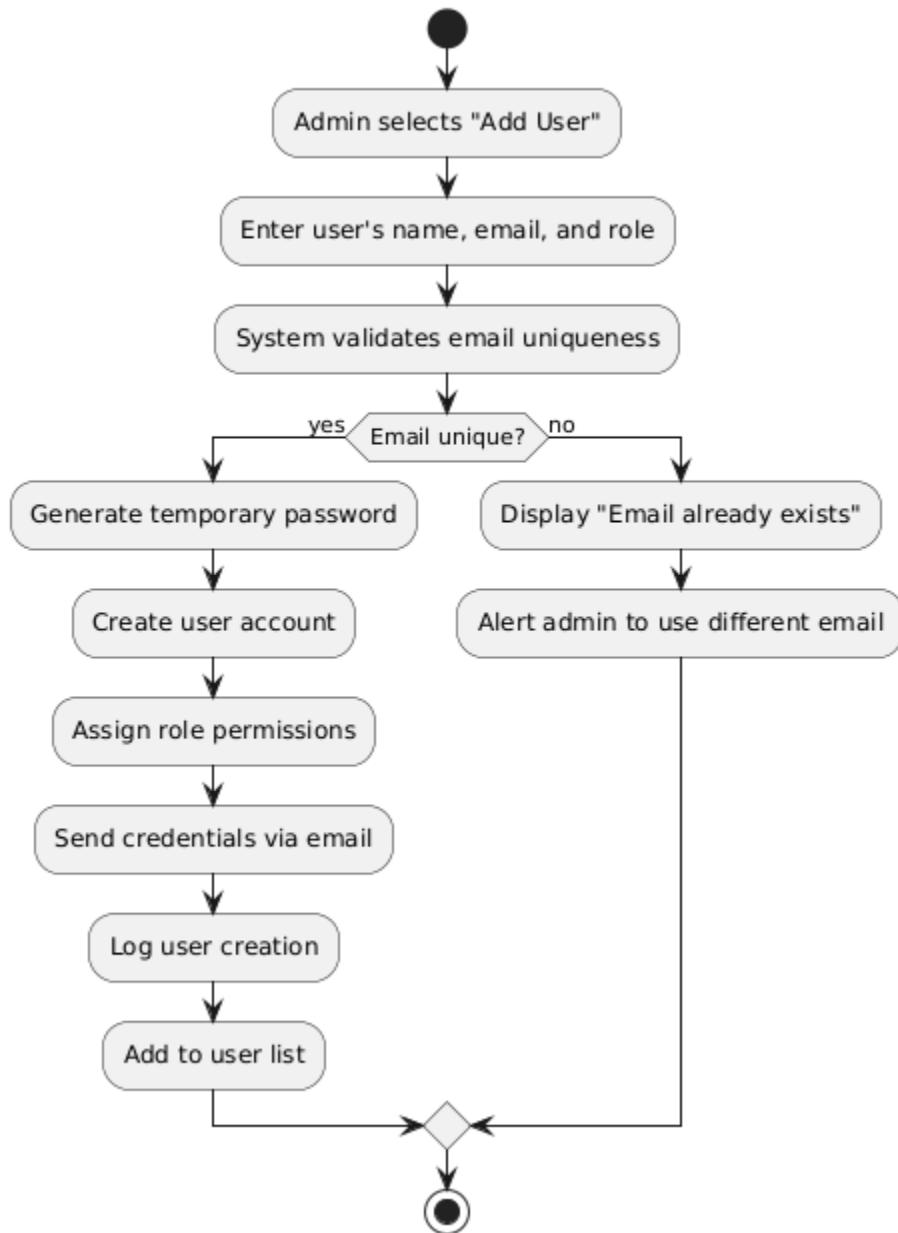
NEW Activity diagrams done by Era Erla Emi Enea Laert.

Enea 01-05

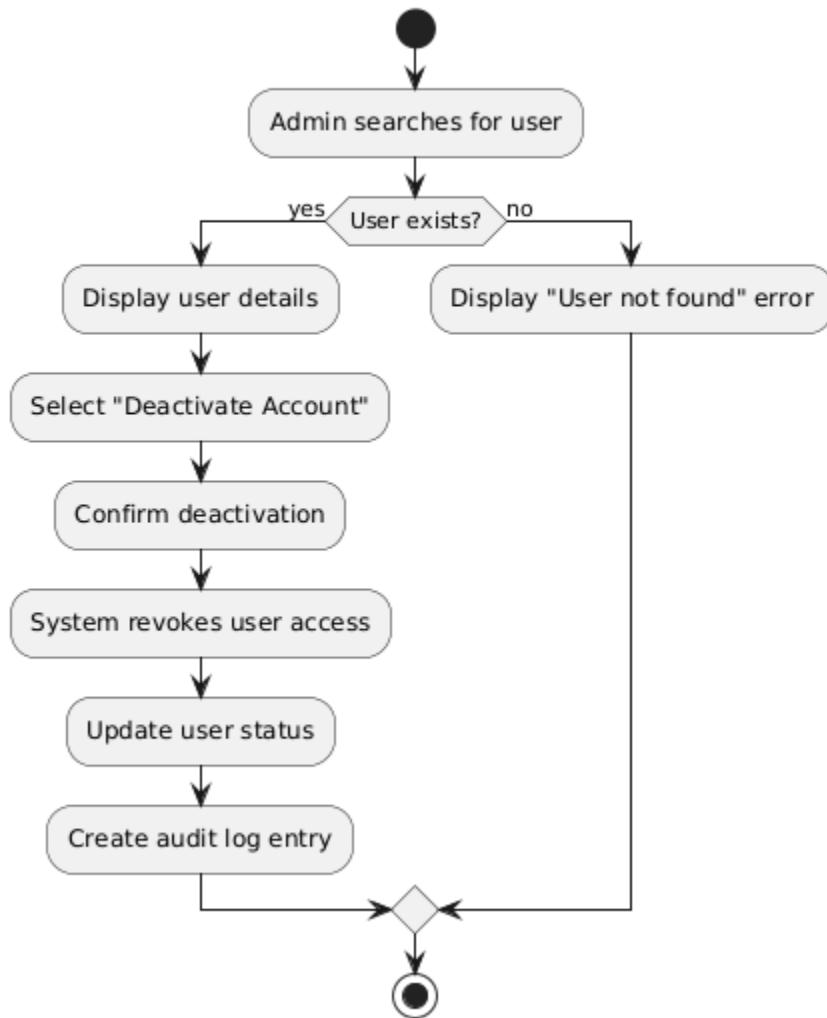
Activity Diagram - UC-ADM-01: Authenticate User



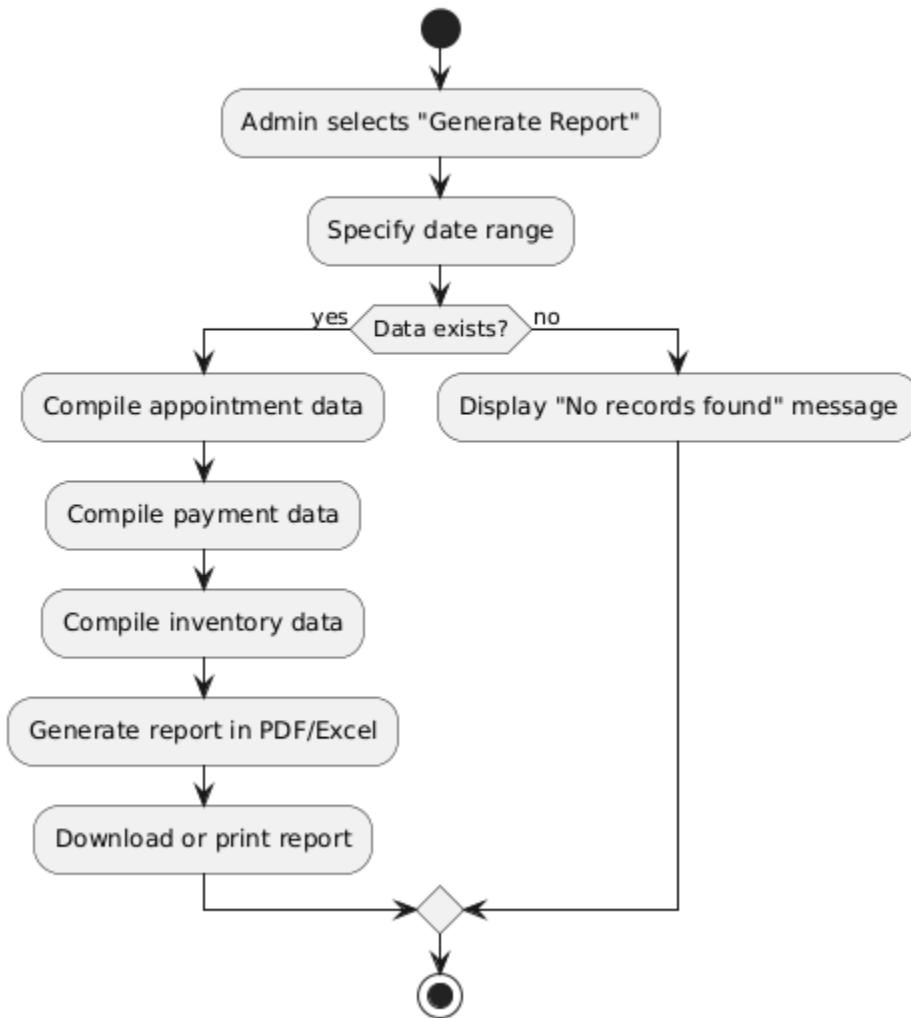
Activity Diagram - UC-ADM-02: Add New User



Activity Diagram - UC-ADM-03: Deactivate User



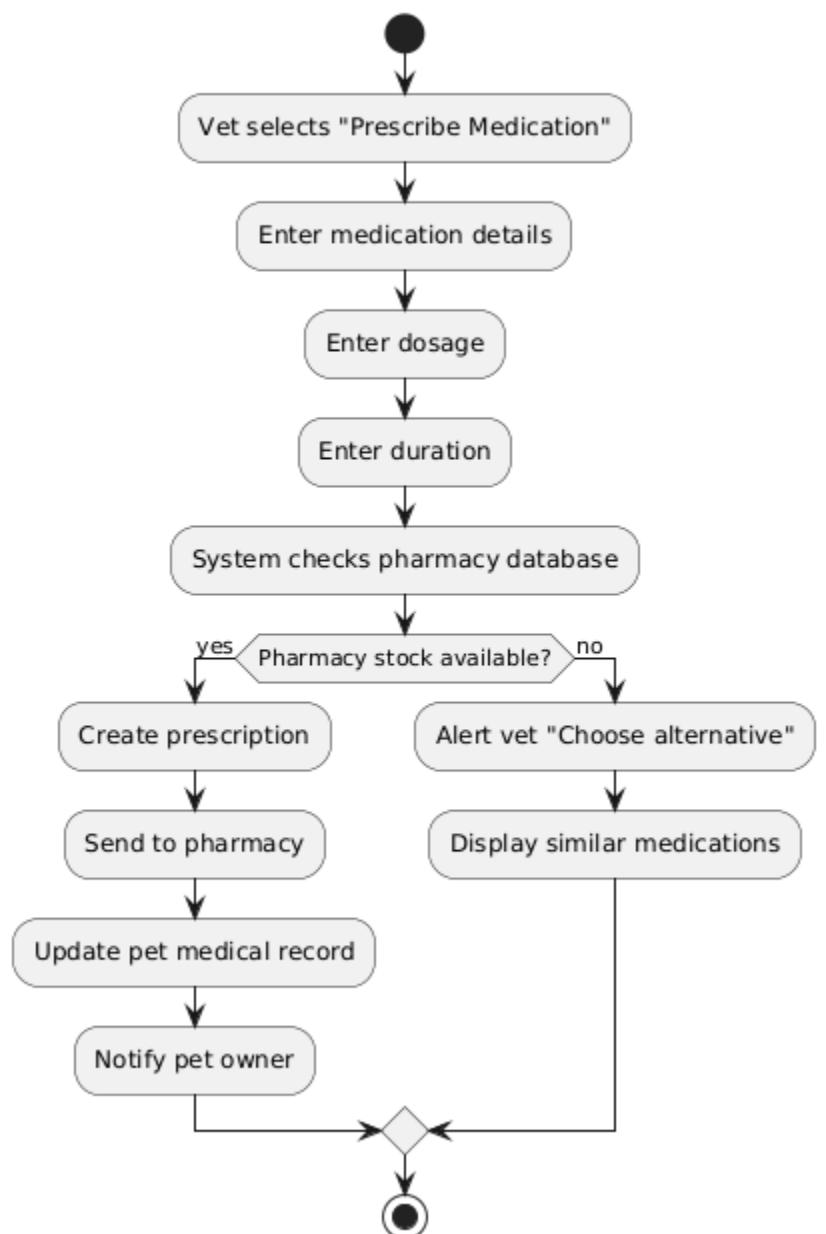
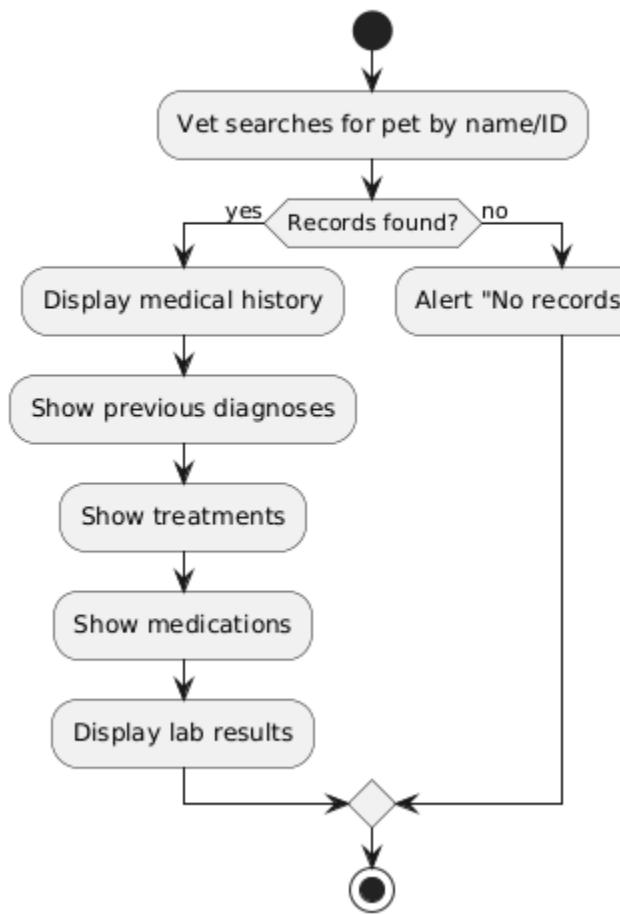
Activity Diagram - UC-ADM-04: Generate Clinic Activity Report



Laert 06-10

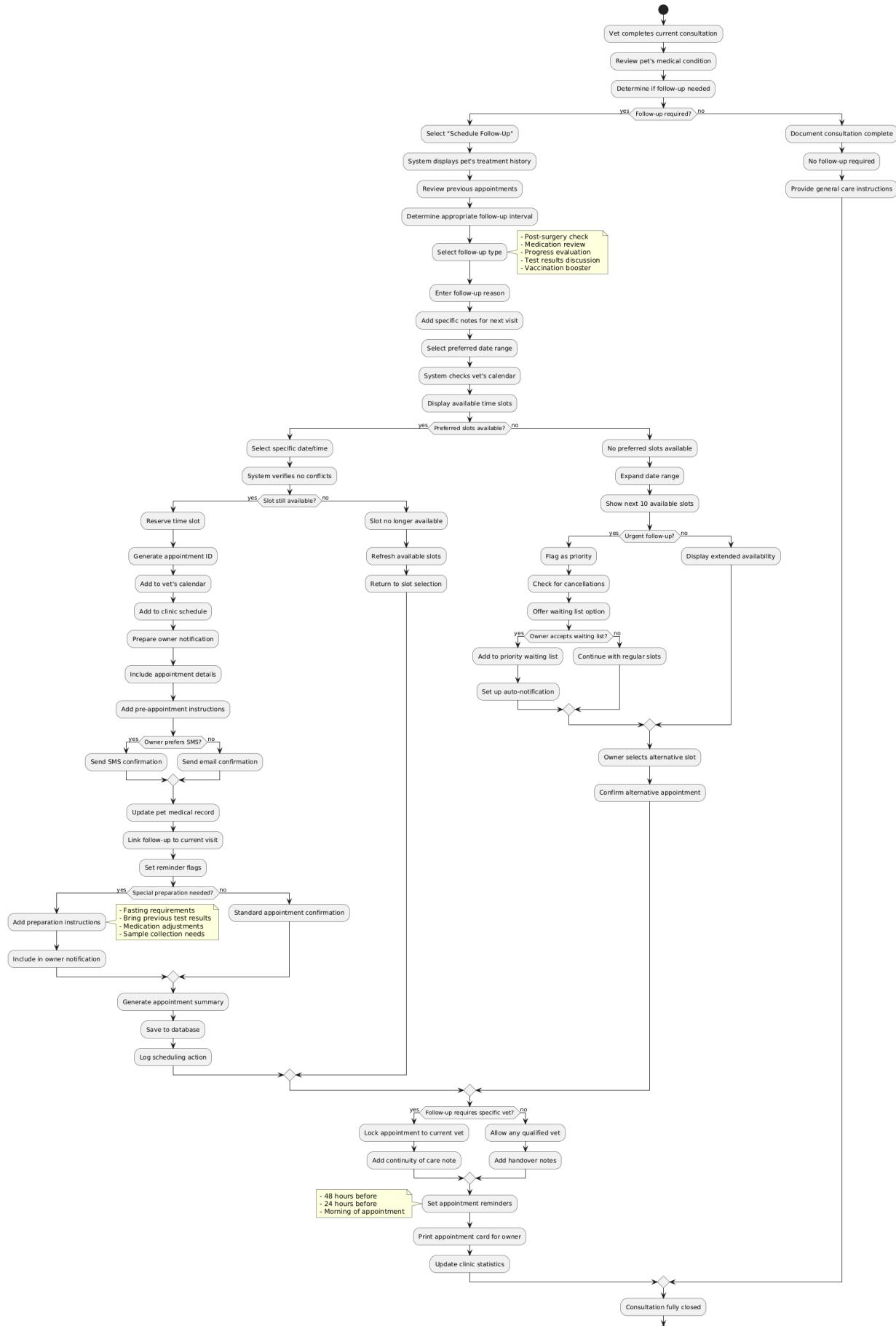
Activity Diagram - UC-VET-06: Prescribe Medication

Activity Diagram - UC-VET-05: View Pet Med



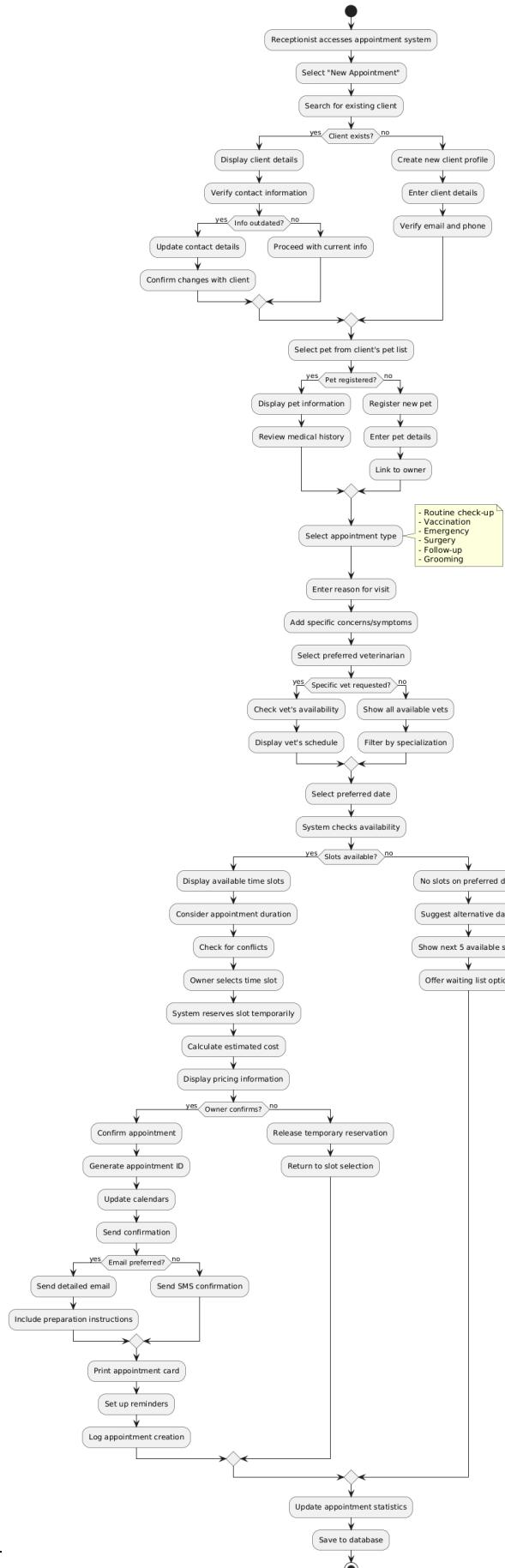
Veterinary Management System Requirements Specification

Activity Diagram - UC-VET-07: Schedule Follow-Up Appointment

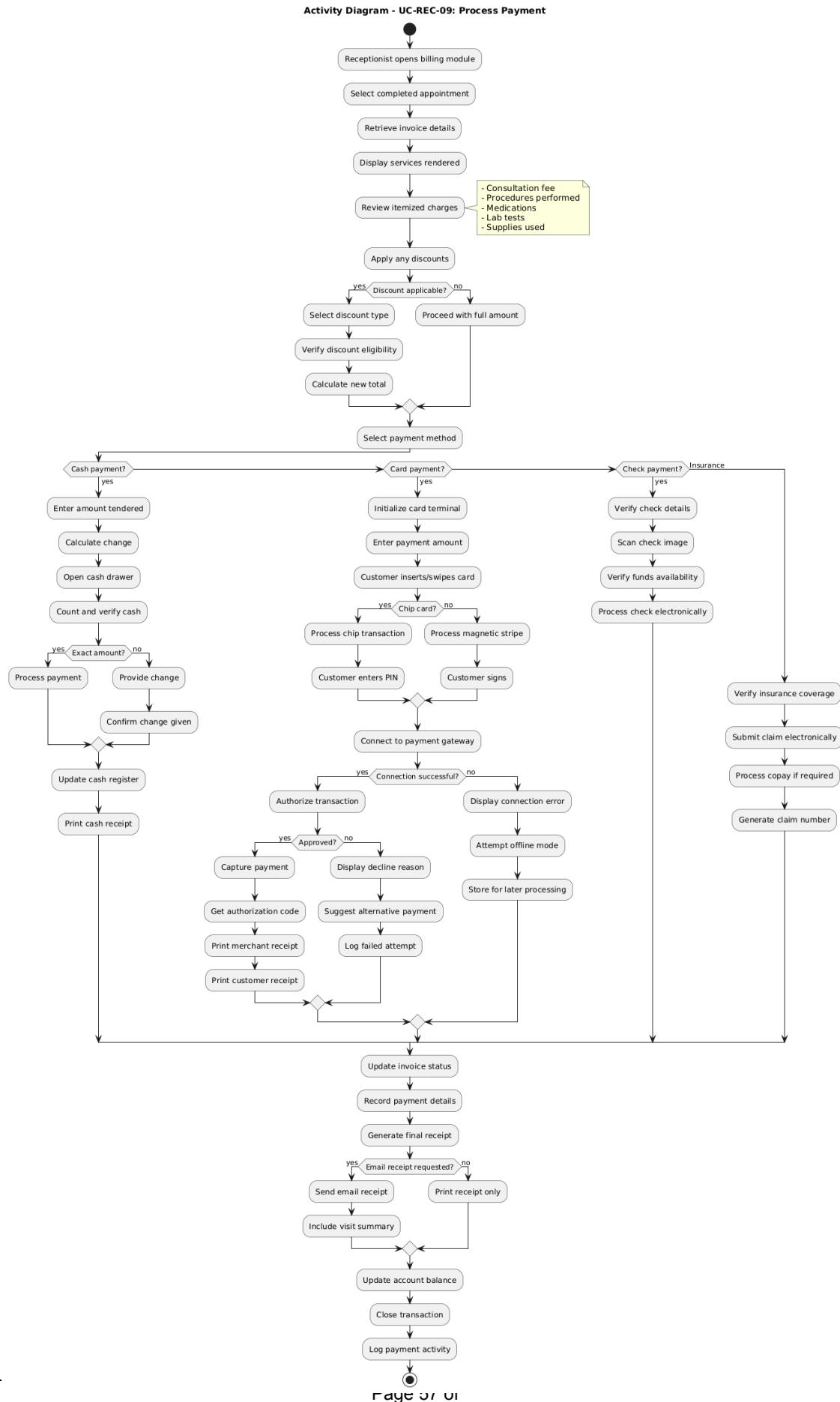


Veterinary Management System Requirements Specification

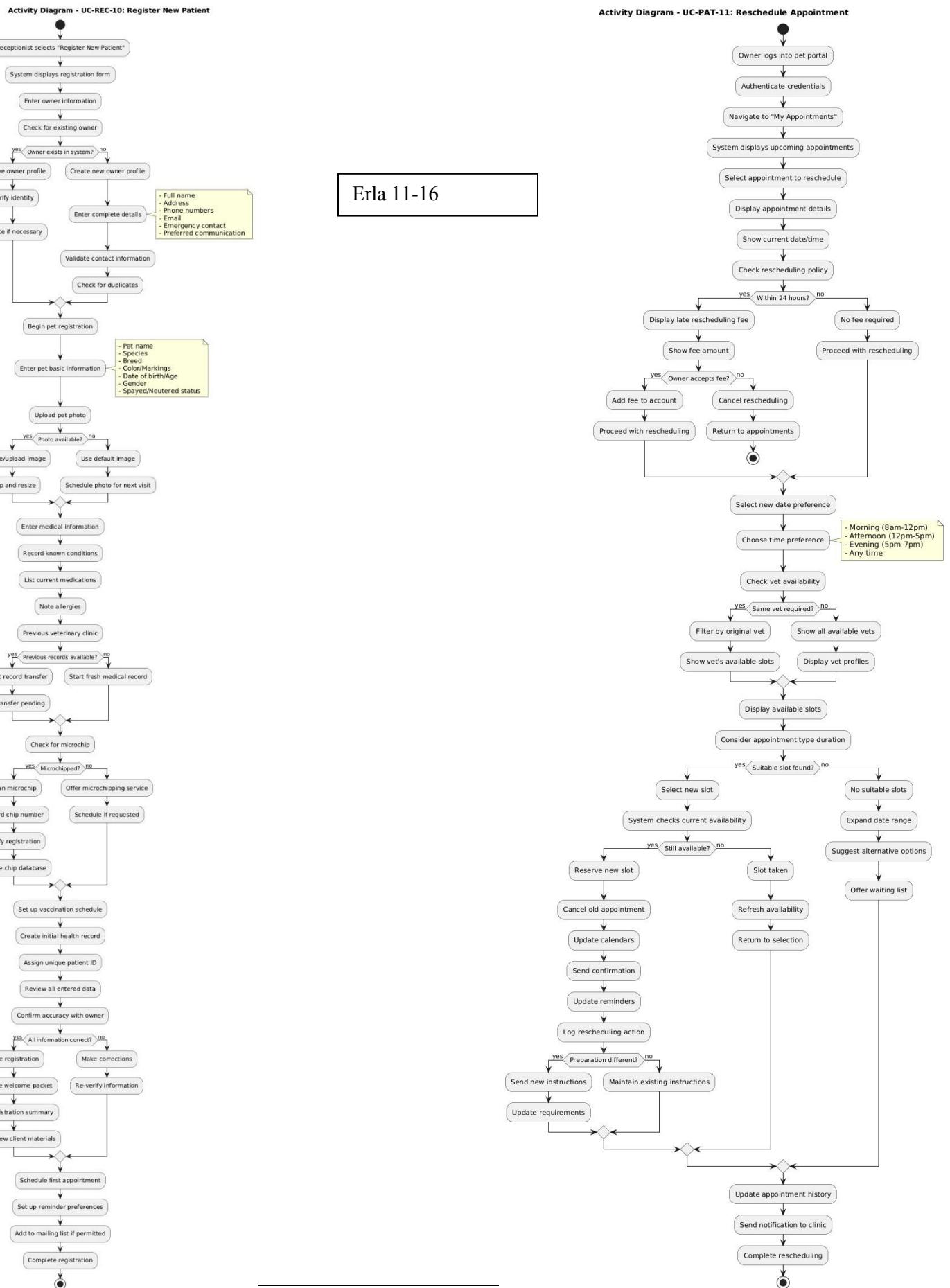
Activity Diagram - UC-REC-08: Schedule Appointment (Detailed)



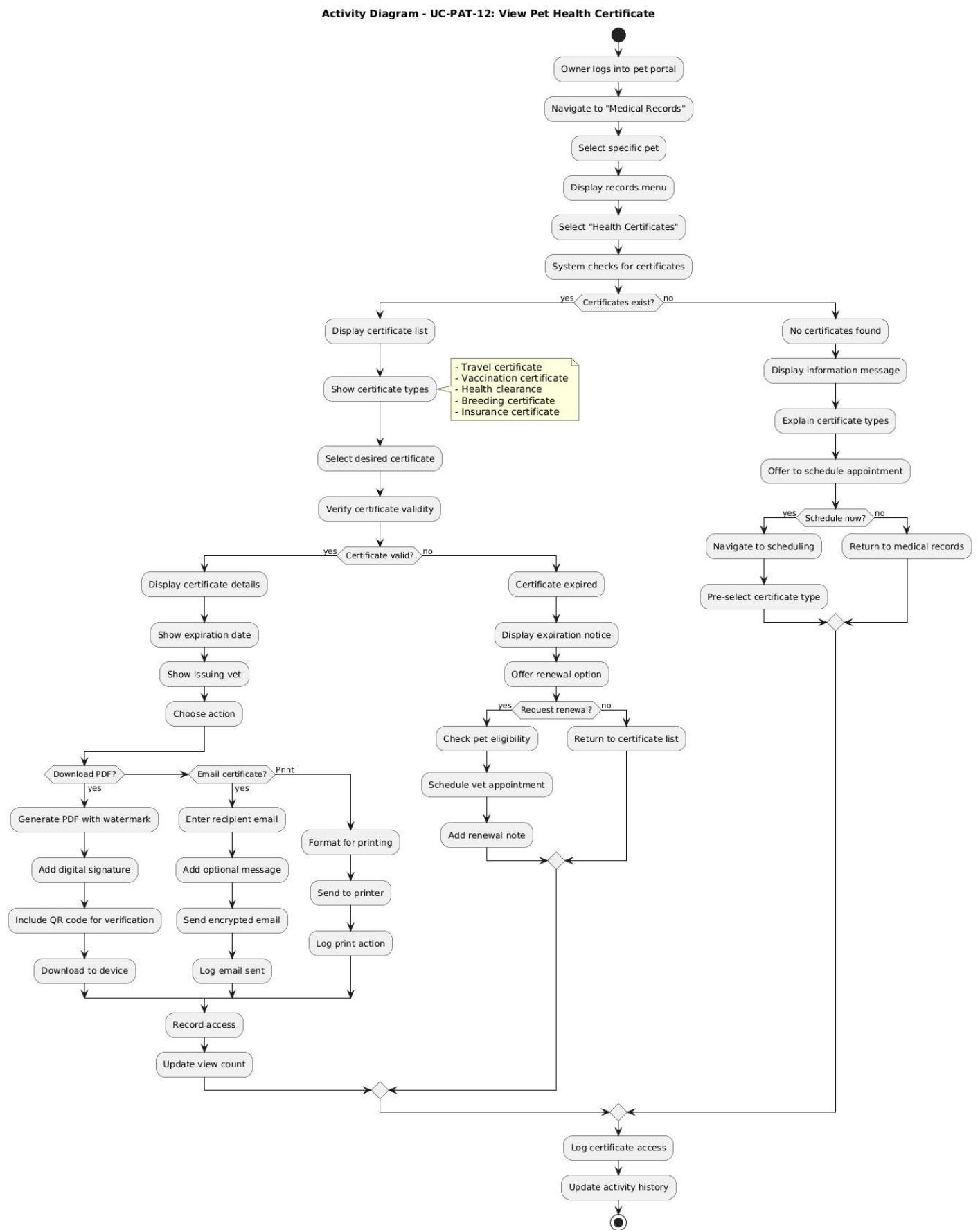
Veterinary Management System Requirements Specification



Veterinary Management System Requirements Specification

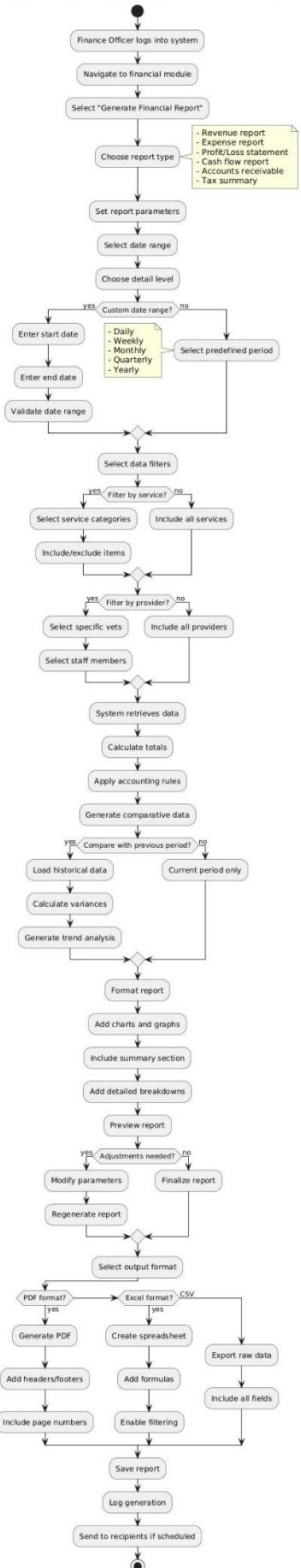


Veterinary Management System Requirements Specification

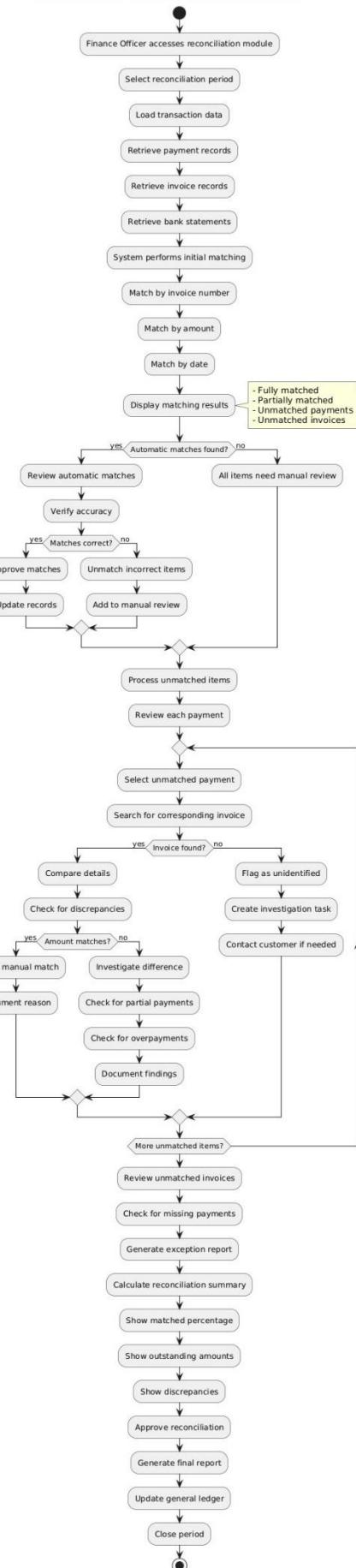


Veterinary Management System Requirements Specification

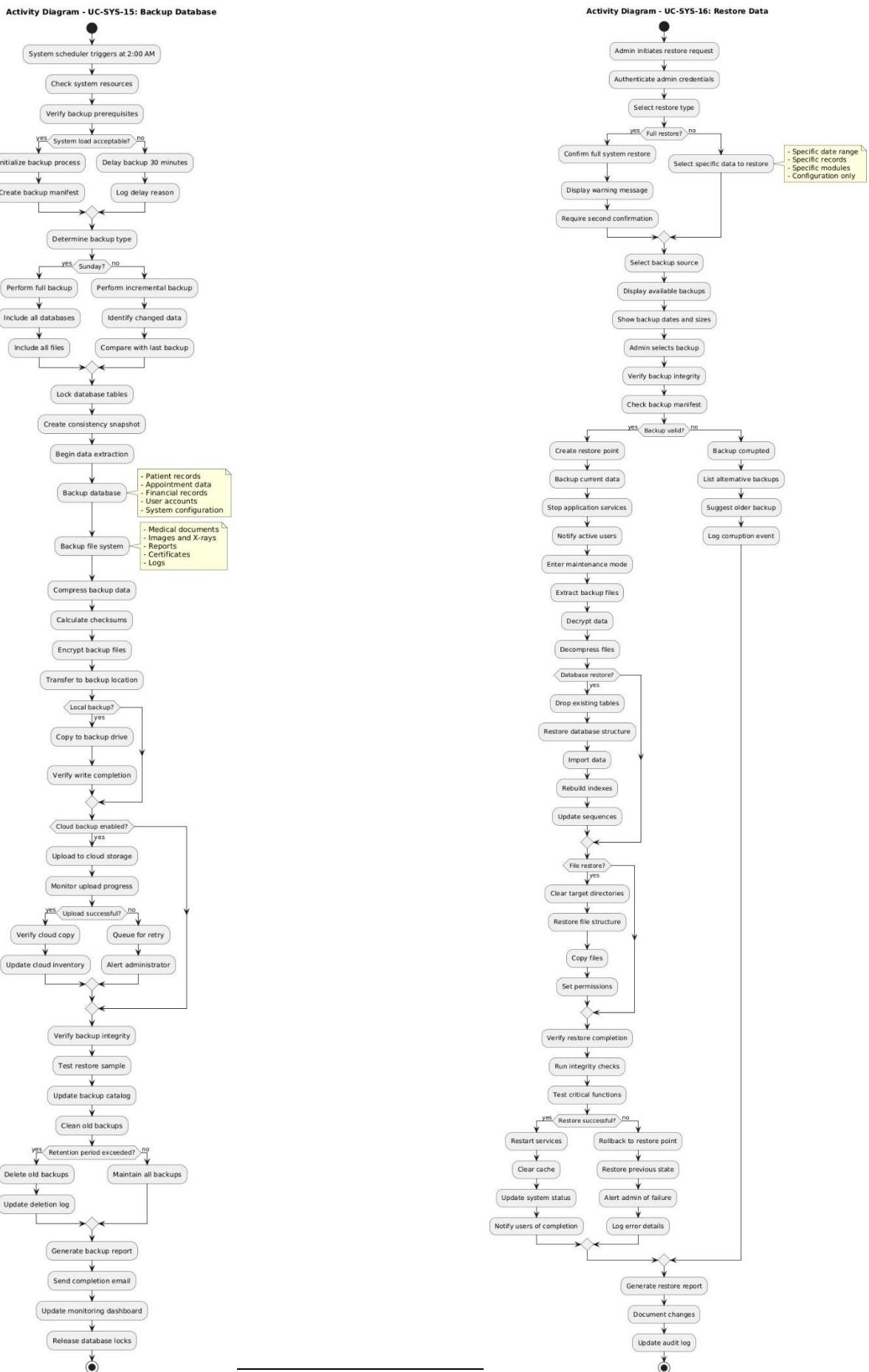
Activity Diagram - UC-FIN-13: Generate Financial Report



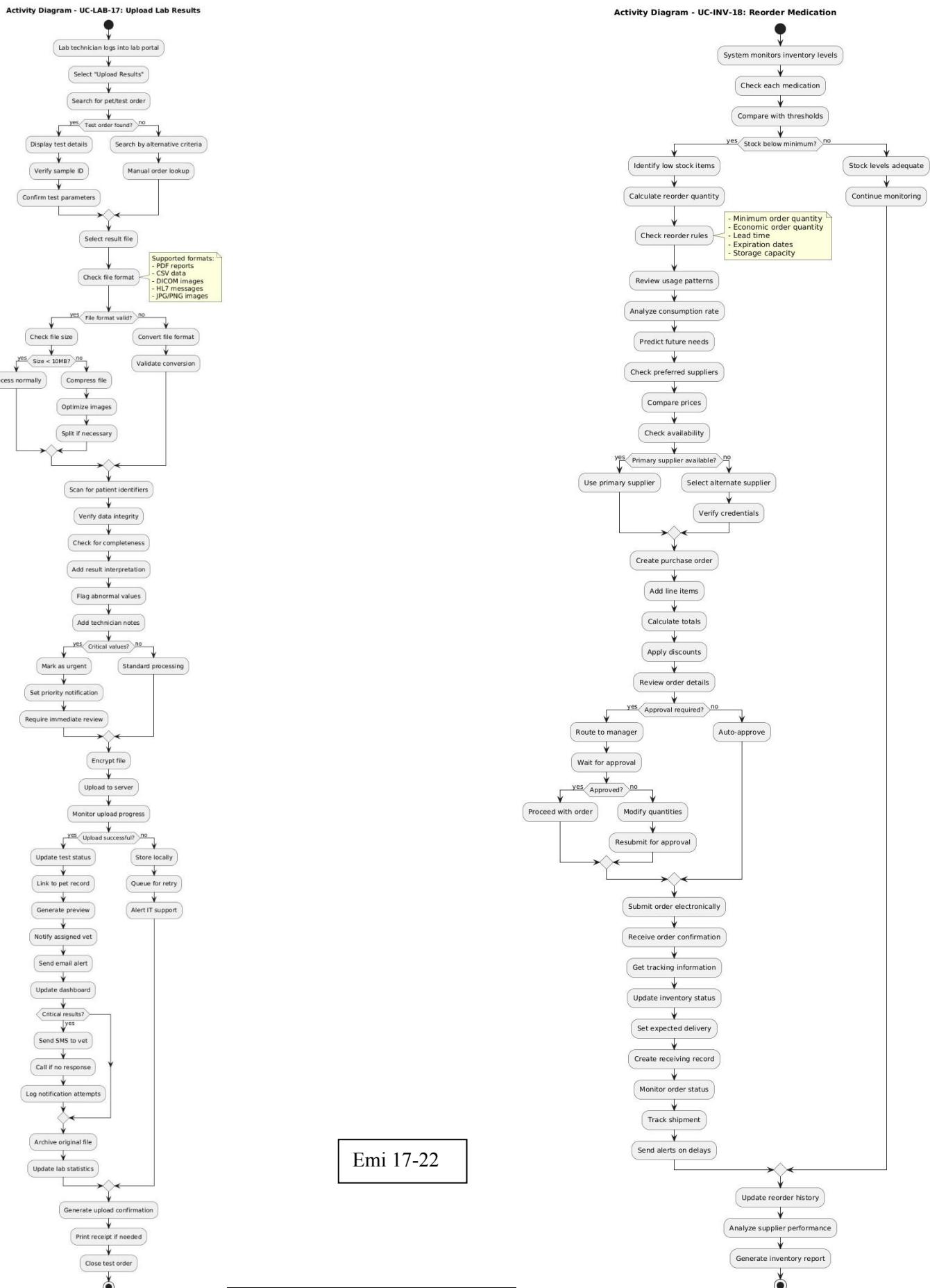
Activity Diagram - UC-FIN-14: Reconcile Payments



Veterinary Management System Requirements Specification

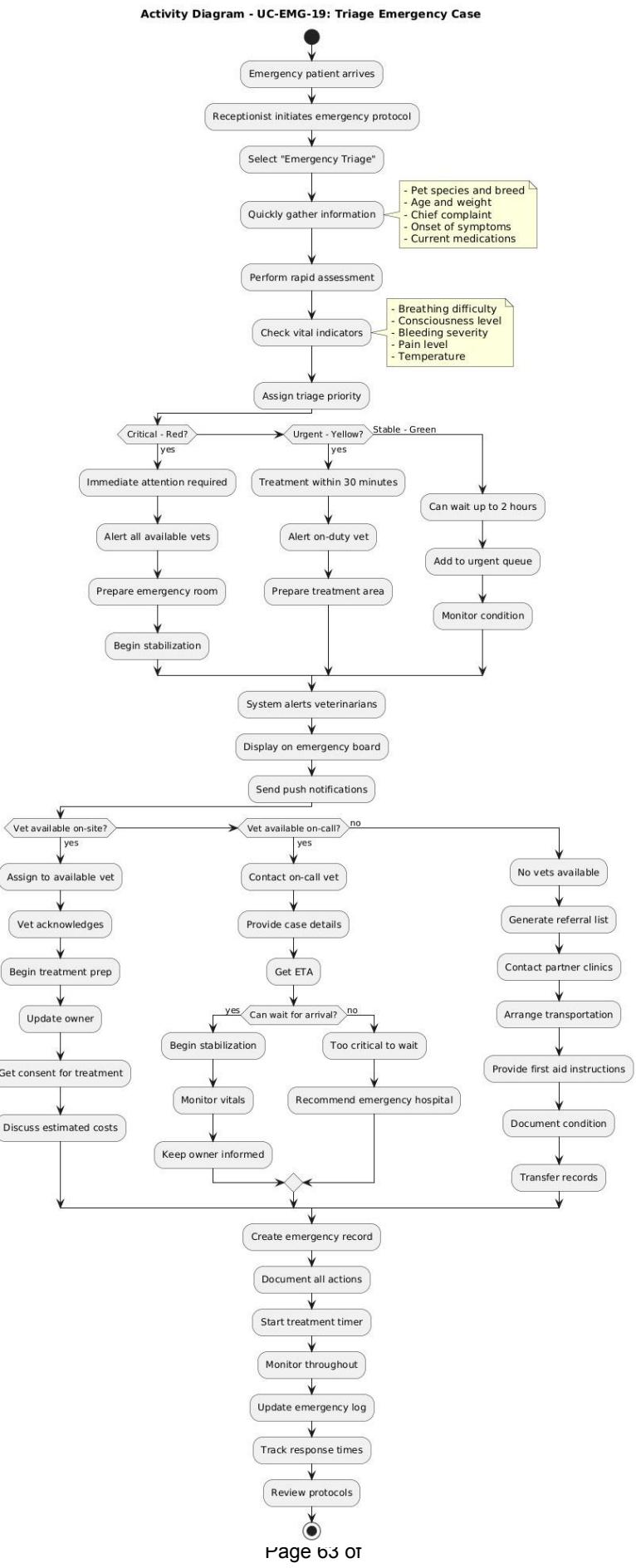


Veterinary Management System Requirements Specification

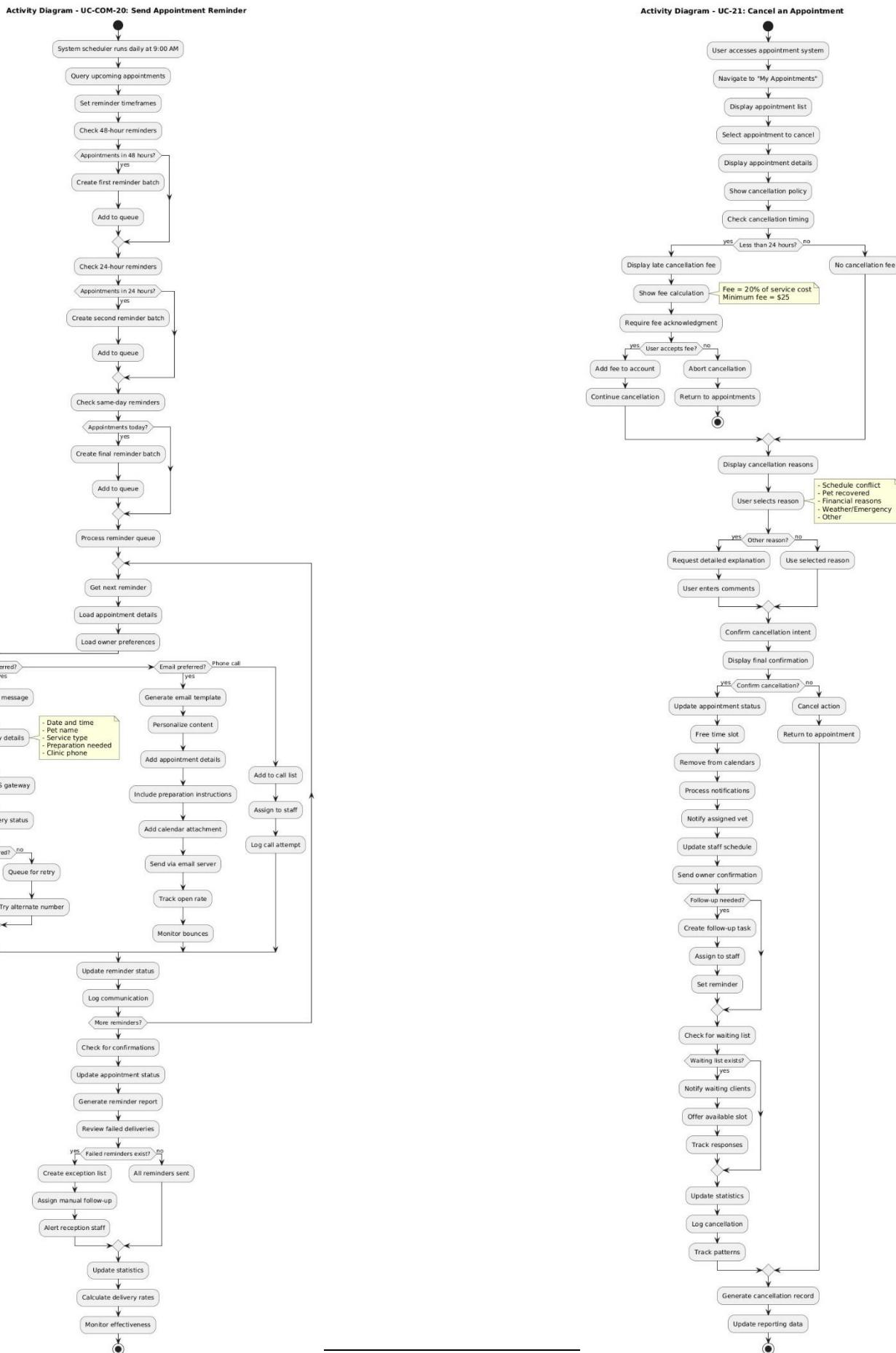


Emi 17-22

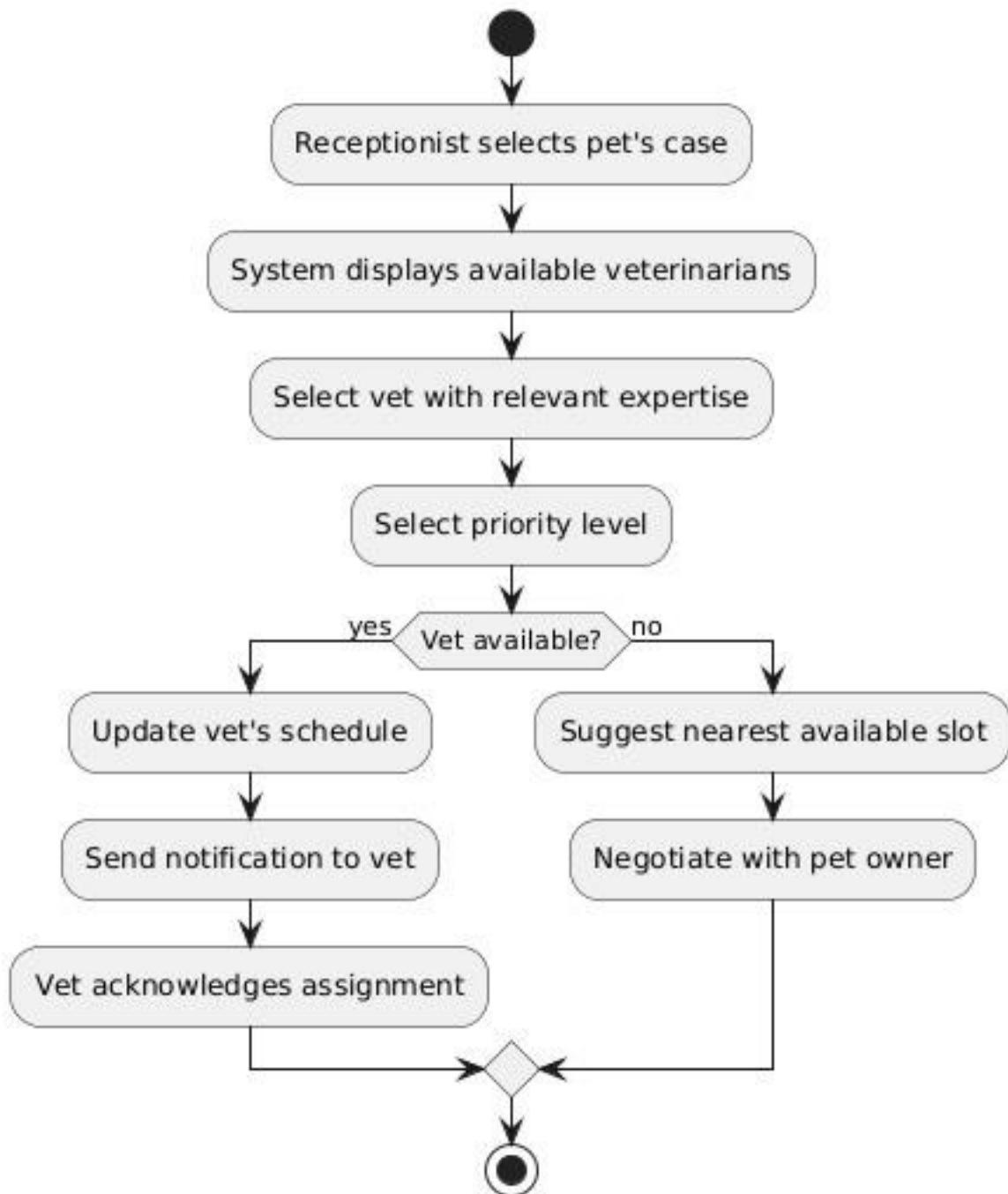
Veterinary Management System Requirements Specification



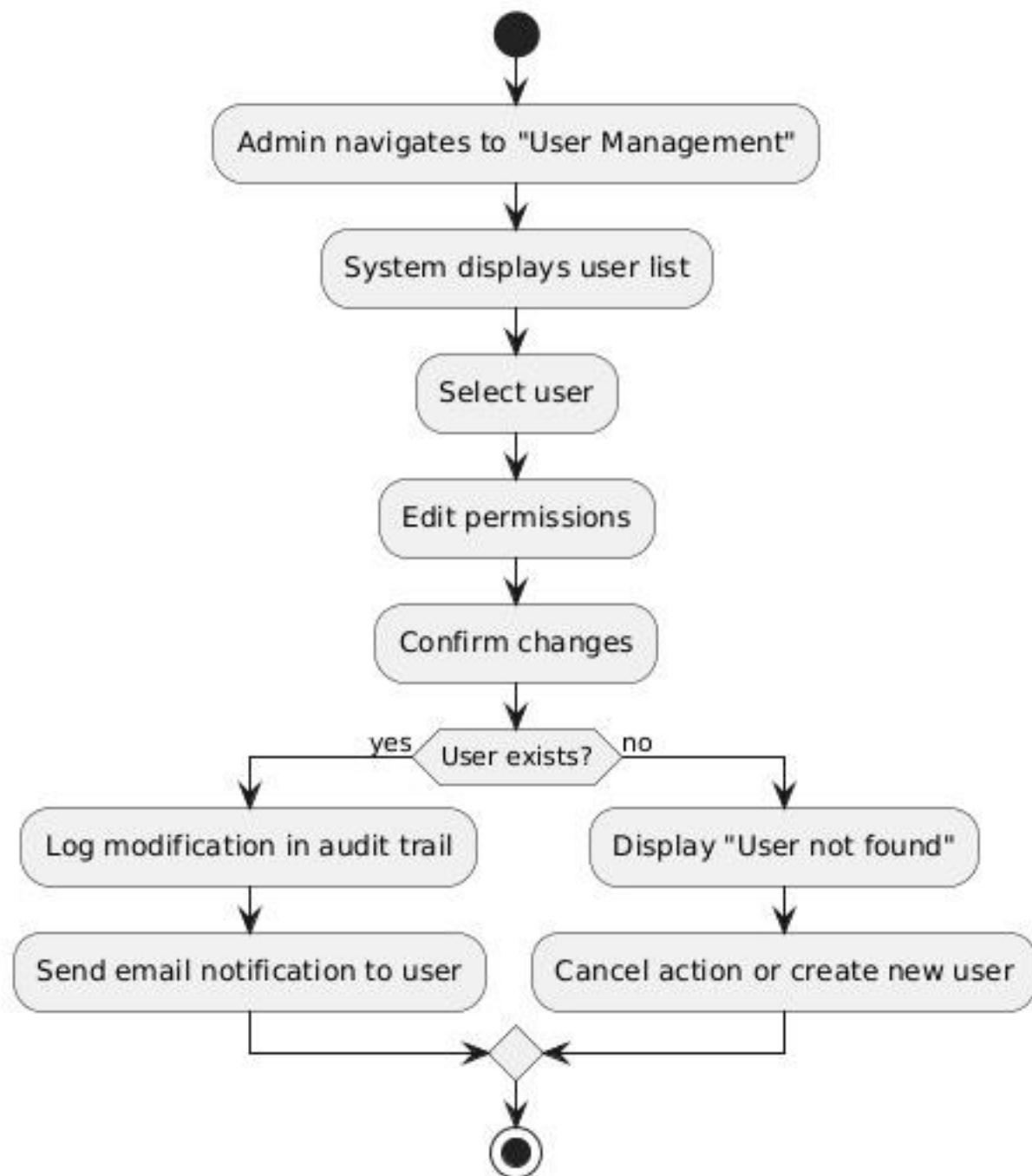
Veterinary Management System Requirements Specification



Activity Diagram - UC-22: Assign a Veterinarian to a Case

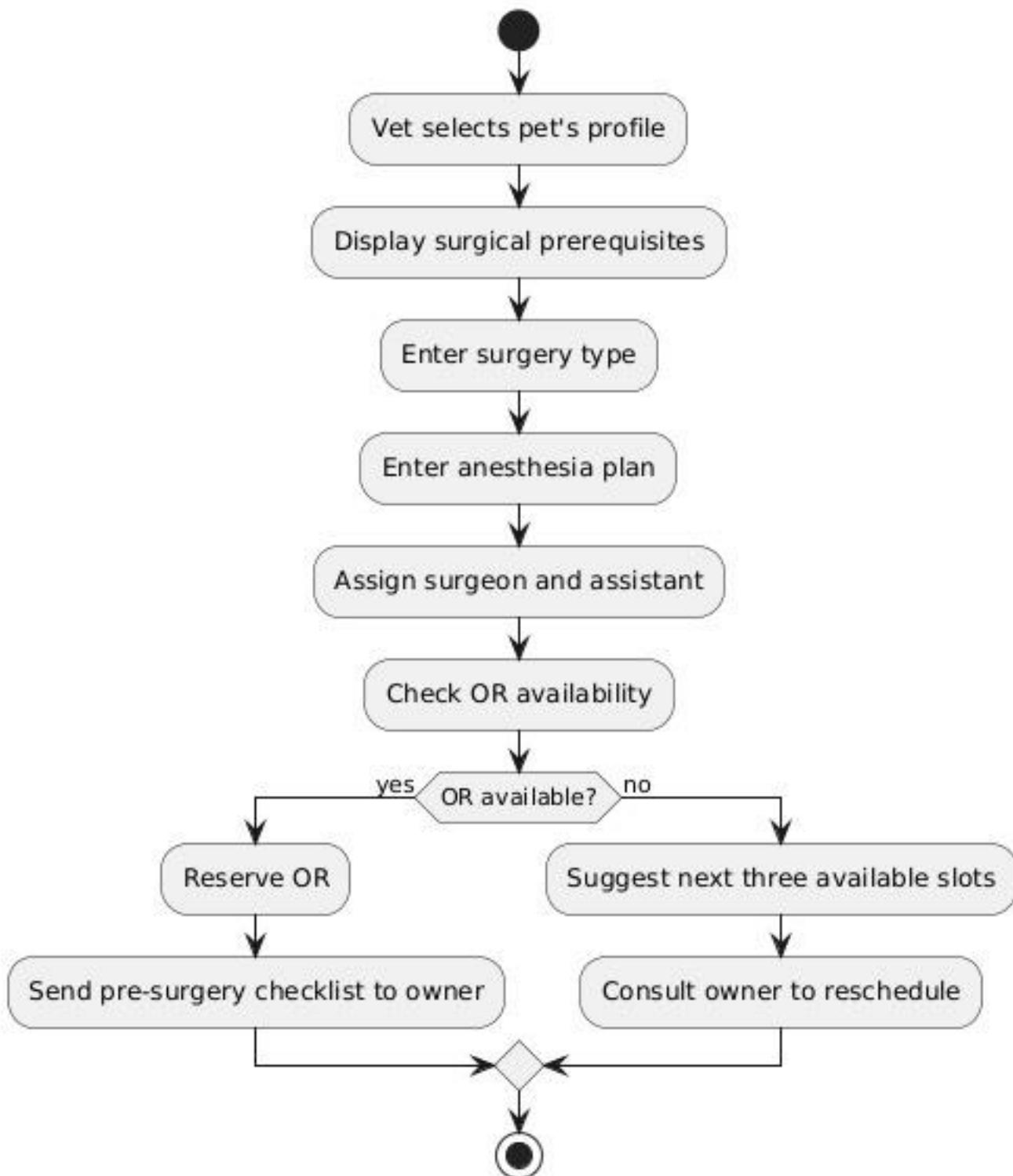


Activity Diagram - UC-23: Manage User Roles & Permissions



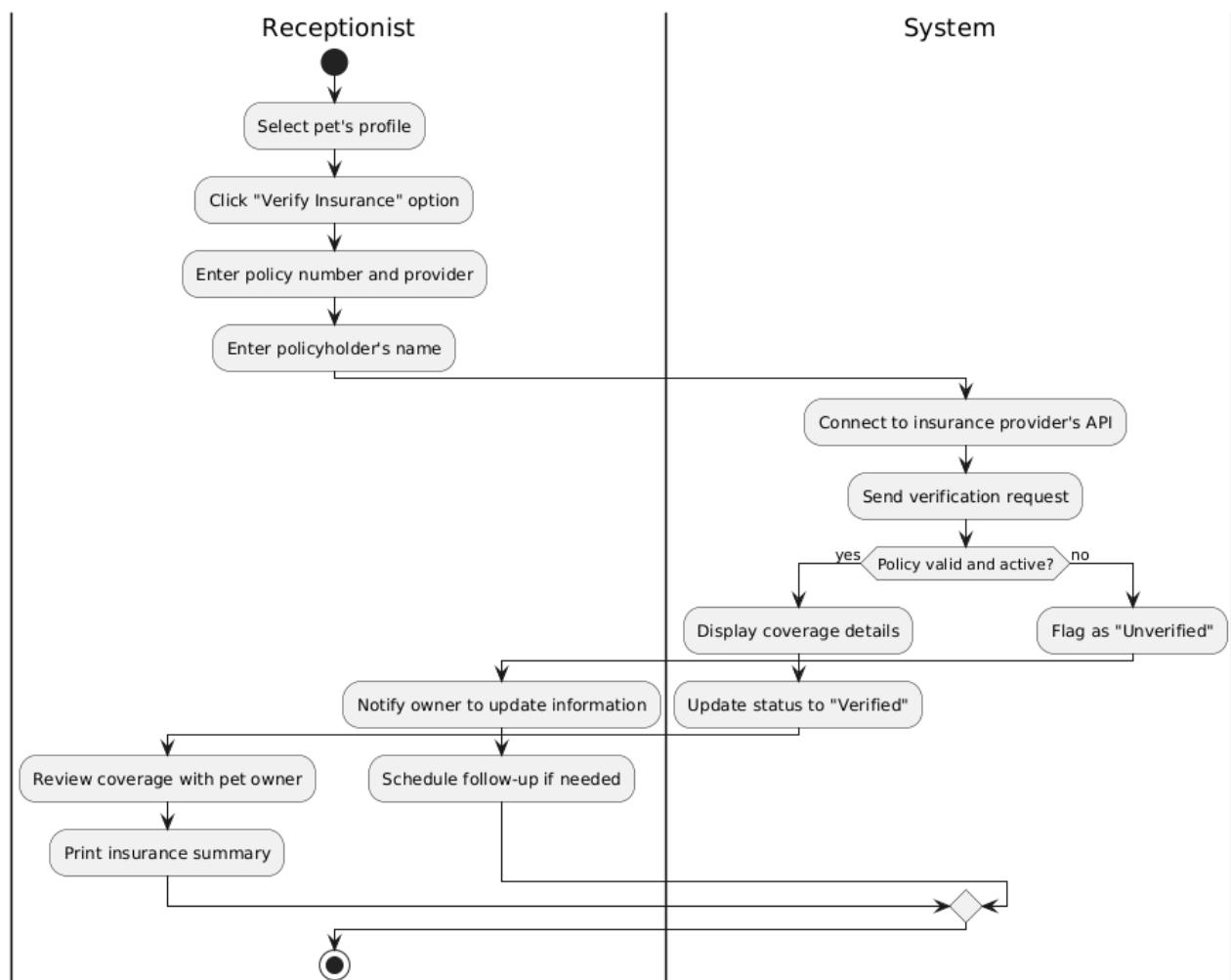
Era 23-29

Activity Diagram - UC-24: Schedule Surgery

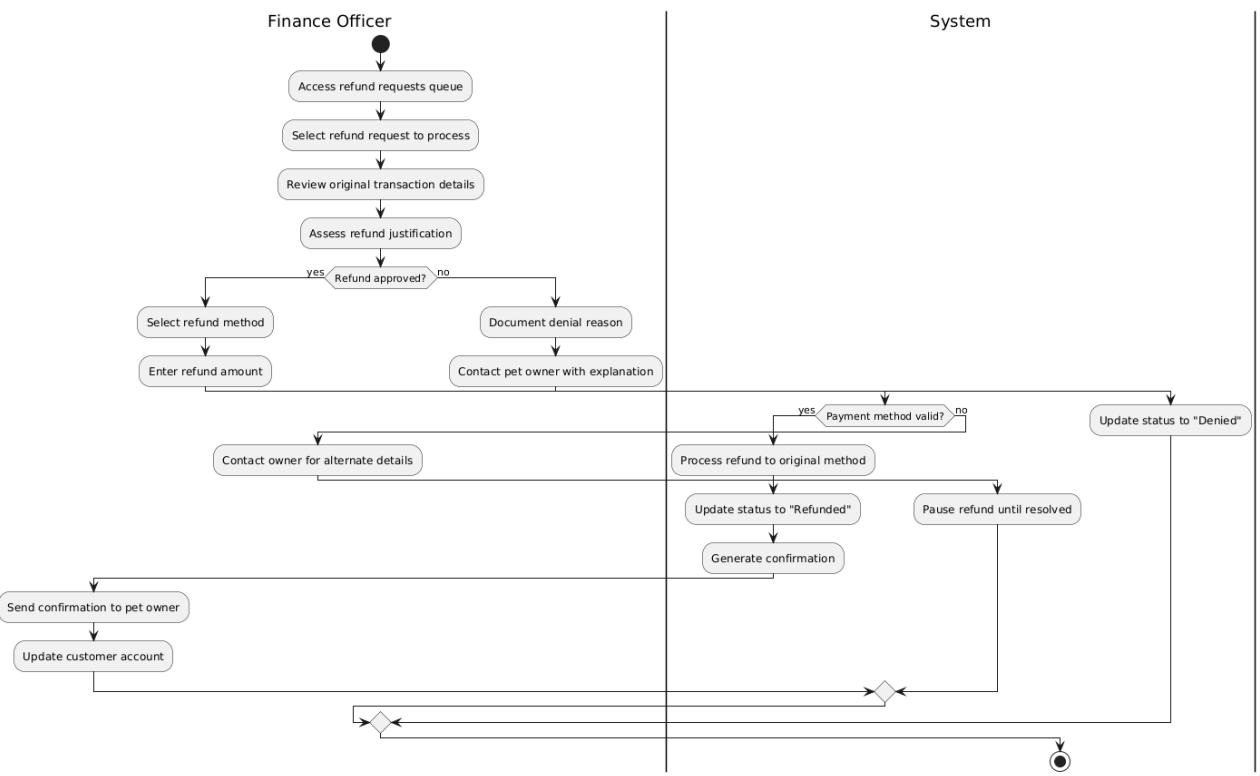


Veterinary Management System Requirements Specification

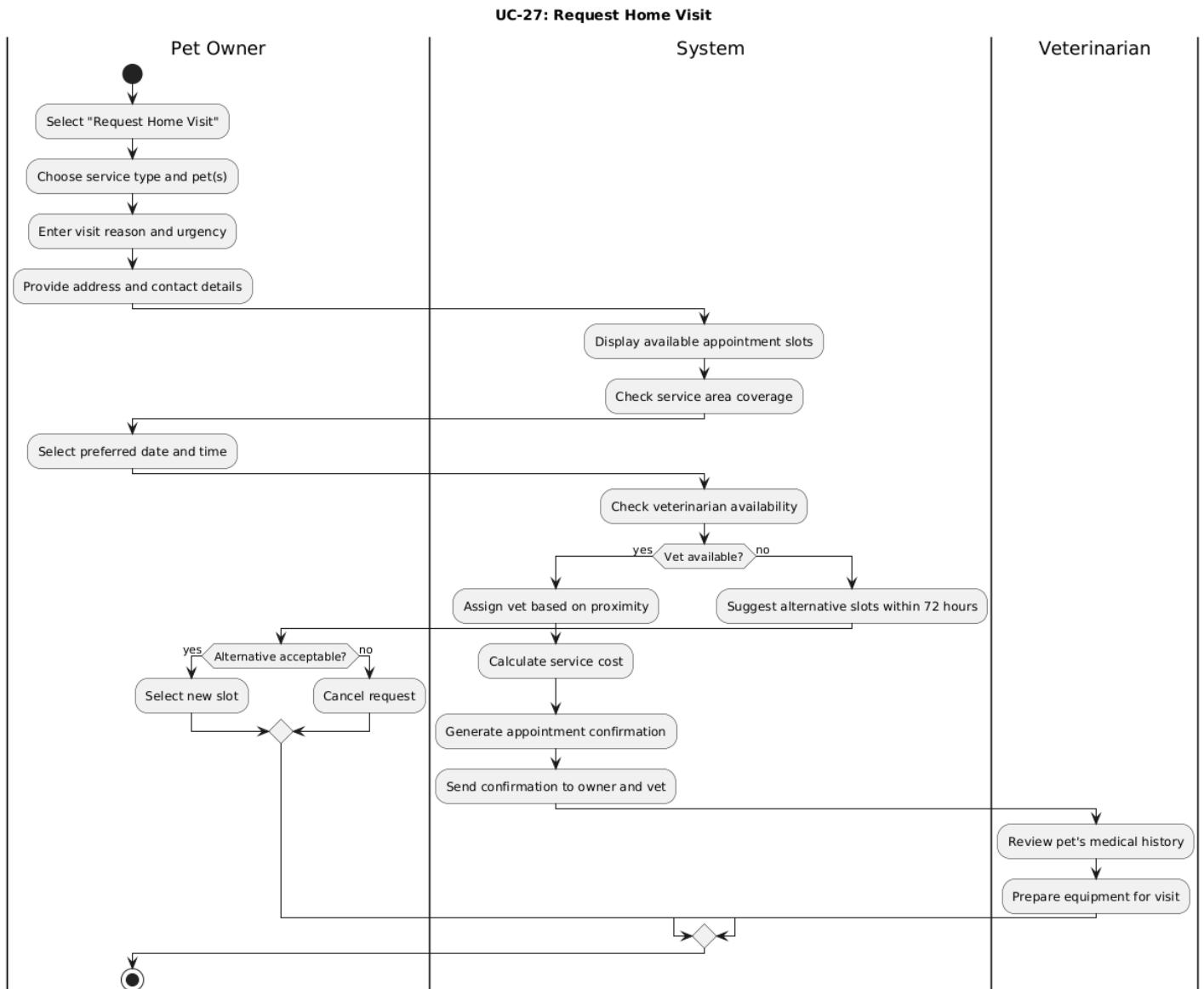
UC-25: Verify Pet Insurance



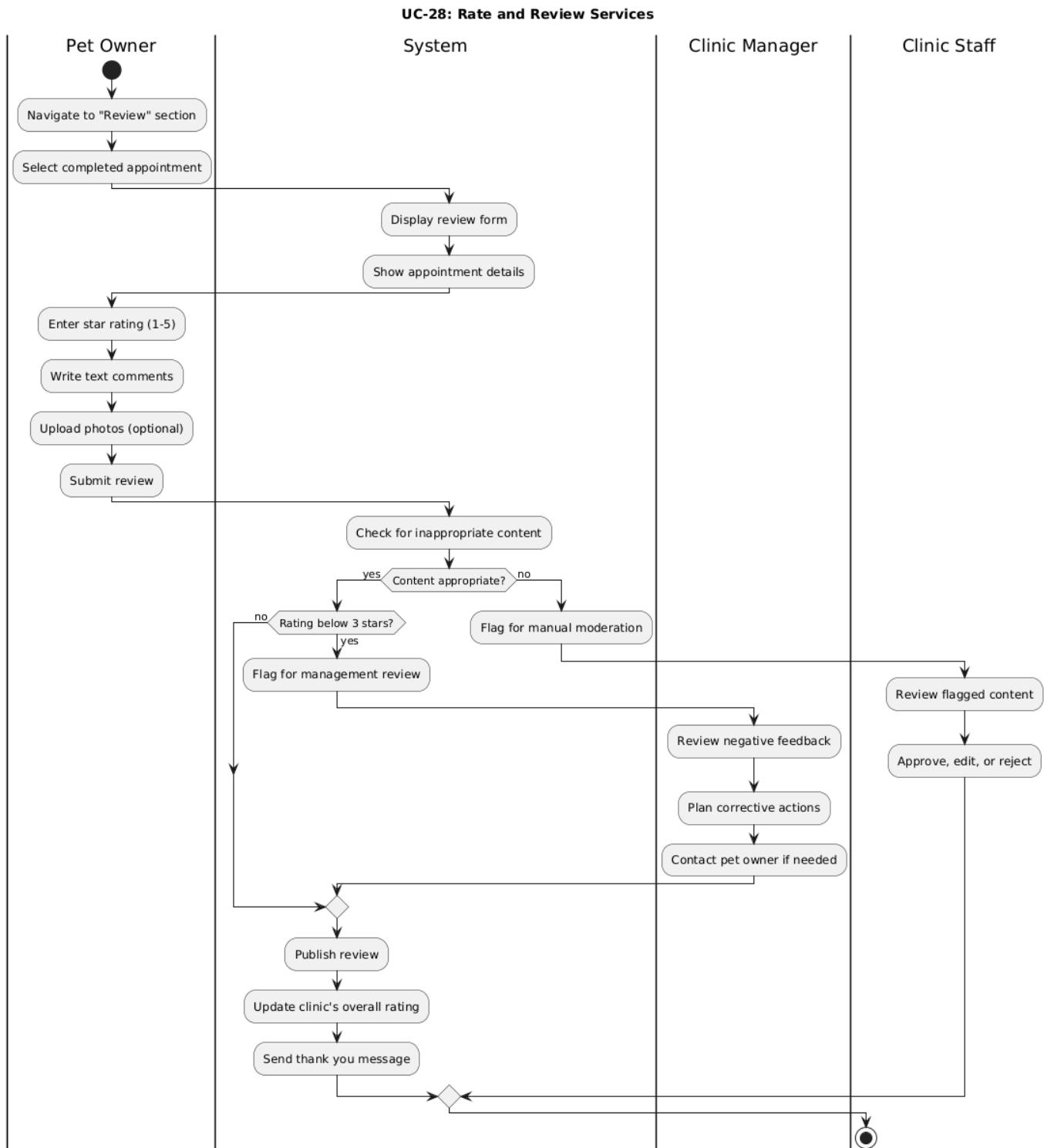
UC-26: Process Refunds



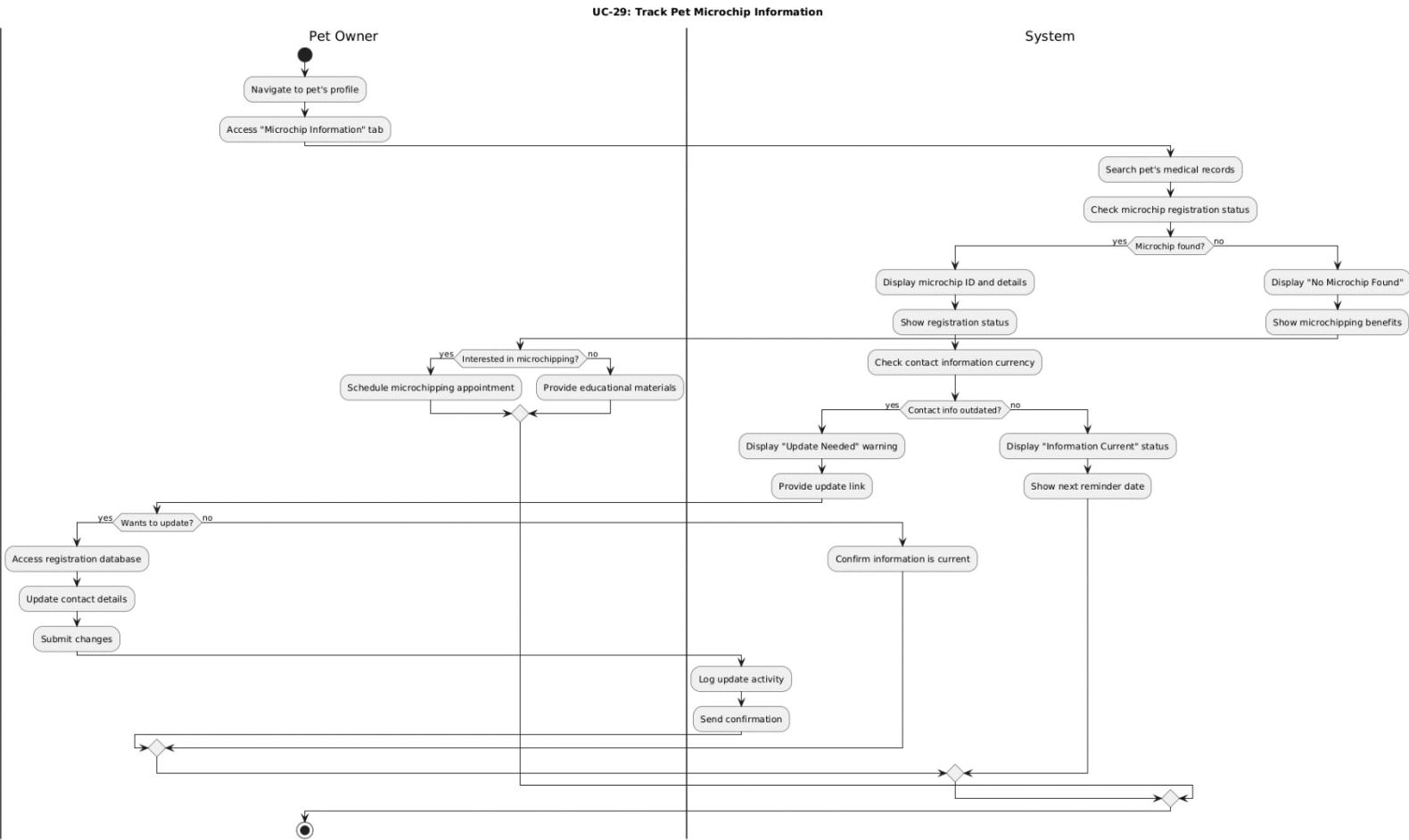
Veterinary Management System Requirements Specification



Veterinary Management System Requirements Specification



Veterinary Management System Requirements Specification



Swimlane Activity Diagram Era

Based on all the short activity diagrams, we made a swimlane activity diagram. It represents the connection that some activity diagrams have with each other.

↳ Who Does What?

- **Admin:** Handles user management, login, and backup restore.
- **Receptionist:** Manages appointments, new registrations, emergency handling, and insurance.
- **Vet:** Prescribes medications, follows up, does surgeries, and home visits.
- **Pet Owner:** Views records, schedules or cancels appointments, submits feedback.
- **Finance Officer:** Generates and reconciles financial reports.
- **System:** Automates backups, reminders, low stock alerts.
- **Lab Tech:** Uploads lab results and manages confirmations.

Use Case Diagram

👉 Actors and Their Responsibilities

1. System Administrator (SA)

Handles the **overall system control**, including user authentication, managing accounts, system maintenance, and generating reports.

Key Use Cases:

- **Authenticate User** – Includes entering credentials, validation, logging attempts; extends to multi-factor auth and password reset.
- **Manage Accounts** – Includes adding/deactivating users and managing roles.

Veterinary Management System Requirements Specification

- **System Maintenance** – Includes backing up and restoring data.
 - **Generate Reports** – Includes clinic activity and financial reports.
-

2. Veterinarian (Vet)

Focuses on **clinical care** for pets.

Key Use Cases:

- **View Pet Medical Record**
 - **Prescribe Medication**
 - **Schedule Follow-Up Appointment**
 - **Schedule Surgery**
-

3. Receptionist (Rec)

Manages **appointments, patient registration, billing**, and vet assignment.

Key Use Cases:

- **Register New Patient**
 - **Schedule / Cancel Appointment**
 - **Process Payment** – Includes selecting payment method, authorizing transactions, generating receipts; may extend to handling errors.
 - **Assign Veterinarian to a Case**
 - **Verify Pet Insurance**
-

4. Pet Owner (PO)

Interacts with the system for **services and feedback**.

Key Use Cases:

- **Reschedule Appointment**
 - **View Pet Health Certificate**
 - **Request Home Visit**
 - **Rate and Review Services**
 - **Track Pet Microchip Information**
-

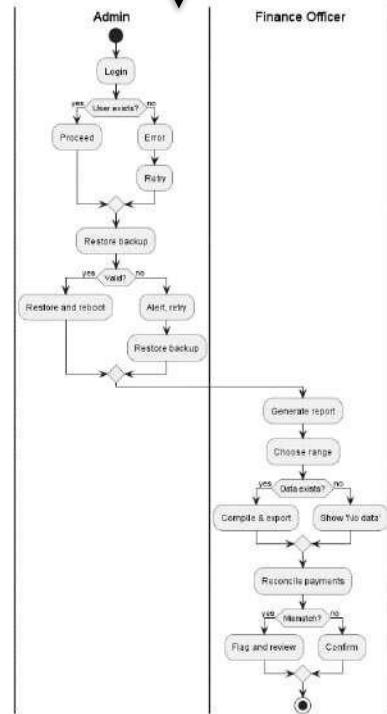
5. Finance Officer (FO)

Handles **financial tracking and refunds**.

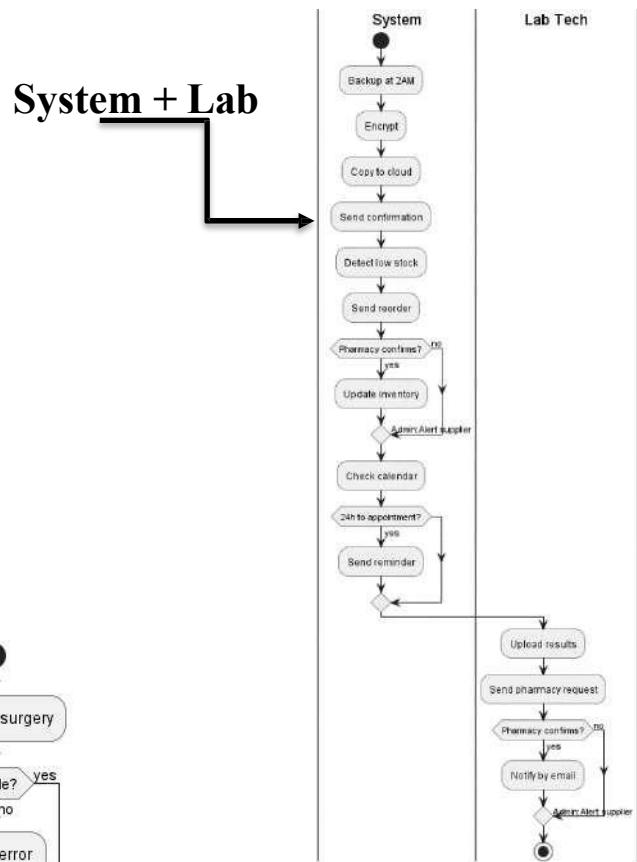
Key Use Cases:

- **Generate Financial Report**
 - **Reconcile Payments** – Includes fetching records, matching payments, identifying/reviewing discrepancies, and updating status.
 - **Process Refunds**
-

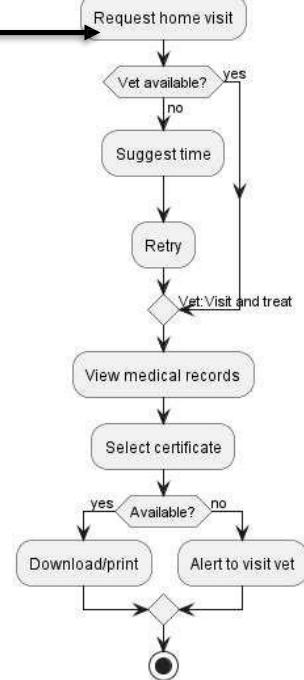
Admin + Finance Officer



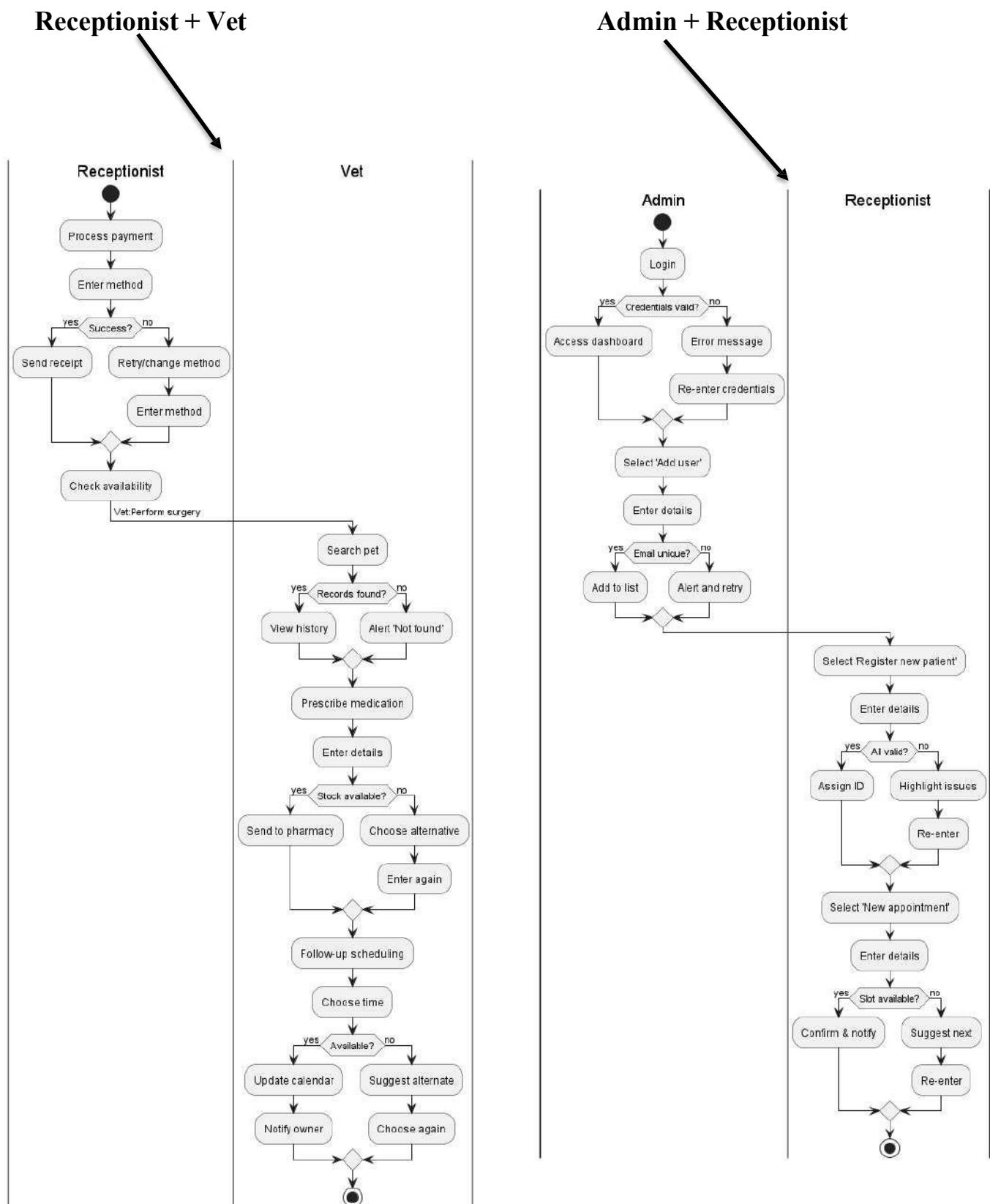
System + Lab



Owner + Vet



Veterinary Management System Requirements Specification



Veterinary Management System Requirements Specification

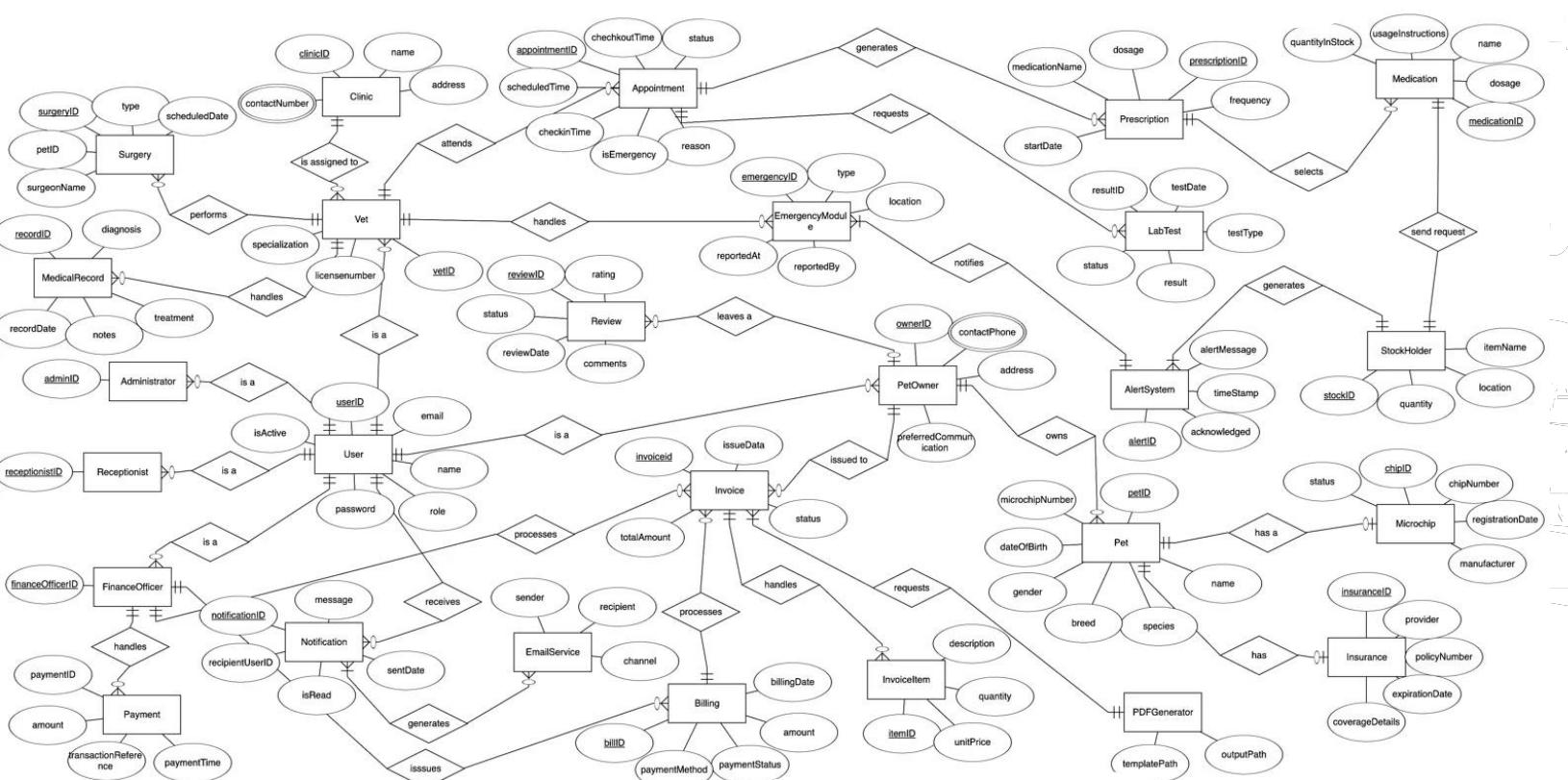


Veterinary Management System Requirements Specification

5. Diagrams

In this section you are going to place all of the diagrams that you build throughout to the course, in following with the slides presented throughout the weeks.

5.1 ER Diagram Erla + Era



//5.2 Use Case Diagram (general) **DONE**//

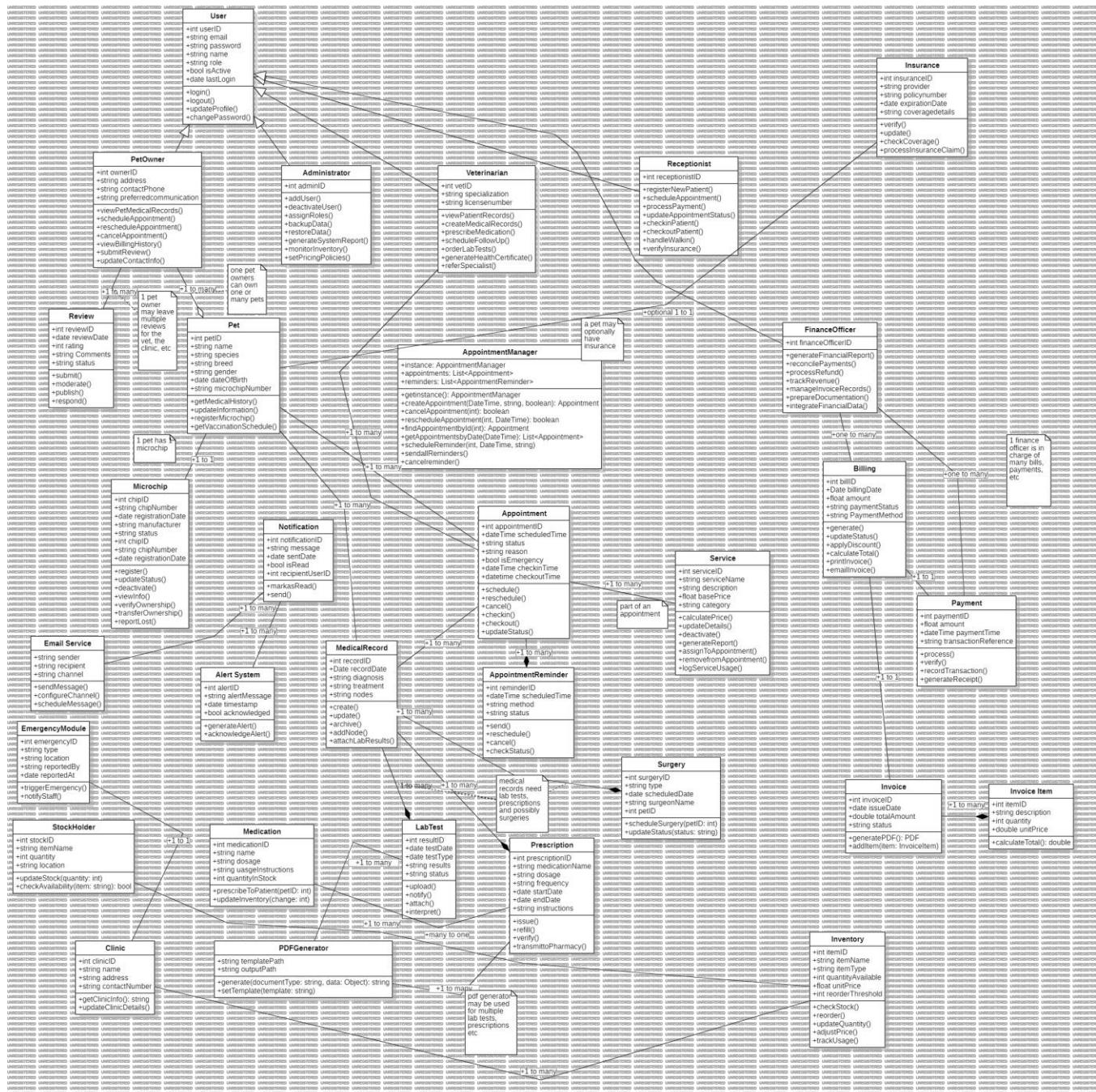
//Use Case Diagram (only one, with all the use cases). **DONE**//

//5.3 Activity Diagram **DONE**//

//Each Activity Diagram should be associated with an use case, associated with a particular requirement which is further associated with a particular use-case. E.g BR_01 which becomes UC_01 which becomes AC_01. **DONE**//

Veterinary Management System Requirements Specification

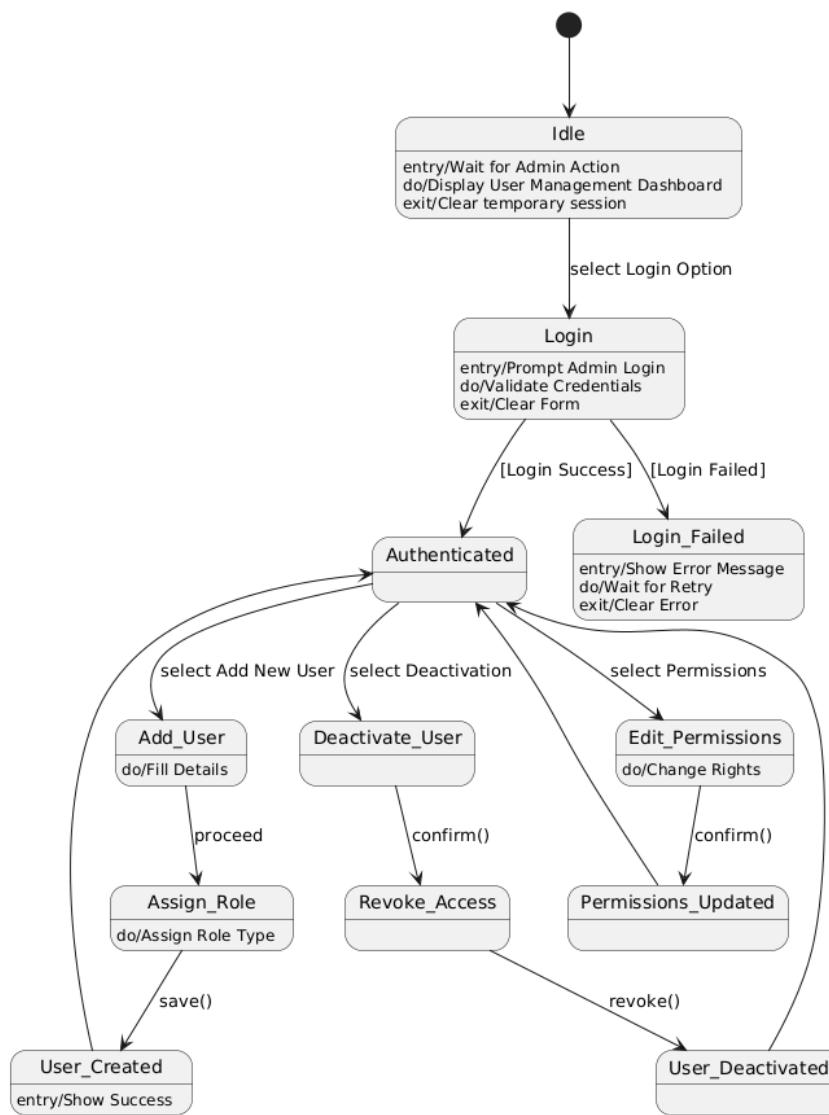
5.4. Class diagram. Emi + Era + Laert



One class diagram (general) for all the classes.

Veterinary Management System Requirements Specification

State Diagram 01 - User Account Management Lifecycle



5.5 State diagram

1. **User Account Management Emi** Place all the relevant state diagrams here. (Covers UC-ADM-01, UC-ADM-02, UC-ADM-03, UC-008)

present it like this :

"This is the state diagram for the User Account Management feature.

We begin at the initial state [*], and move to StartScreen, which represents the system waiting for the admin to log in.

Once the admin enters their credentials, the system transitions to Login, then either:

Moves to Authenticated if the login is successful, or

Returns to Login through Login_Failed if not.

From the Authenticated state, the admin can:

Add a new user → assign a role → and complete at User_Created

Deactivate a user → revoke access → and end at User_Deactivated

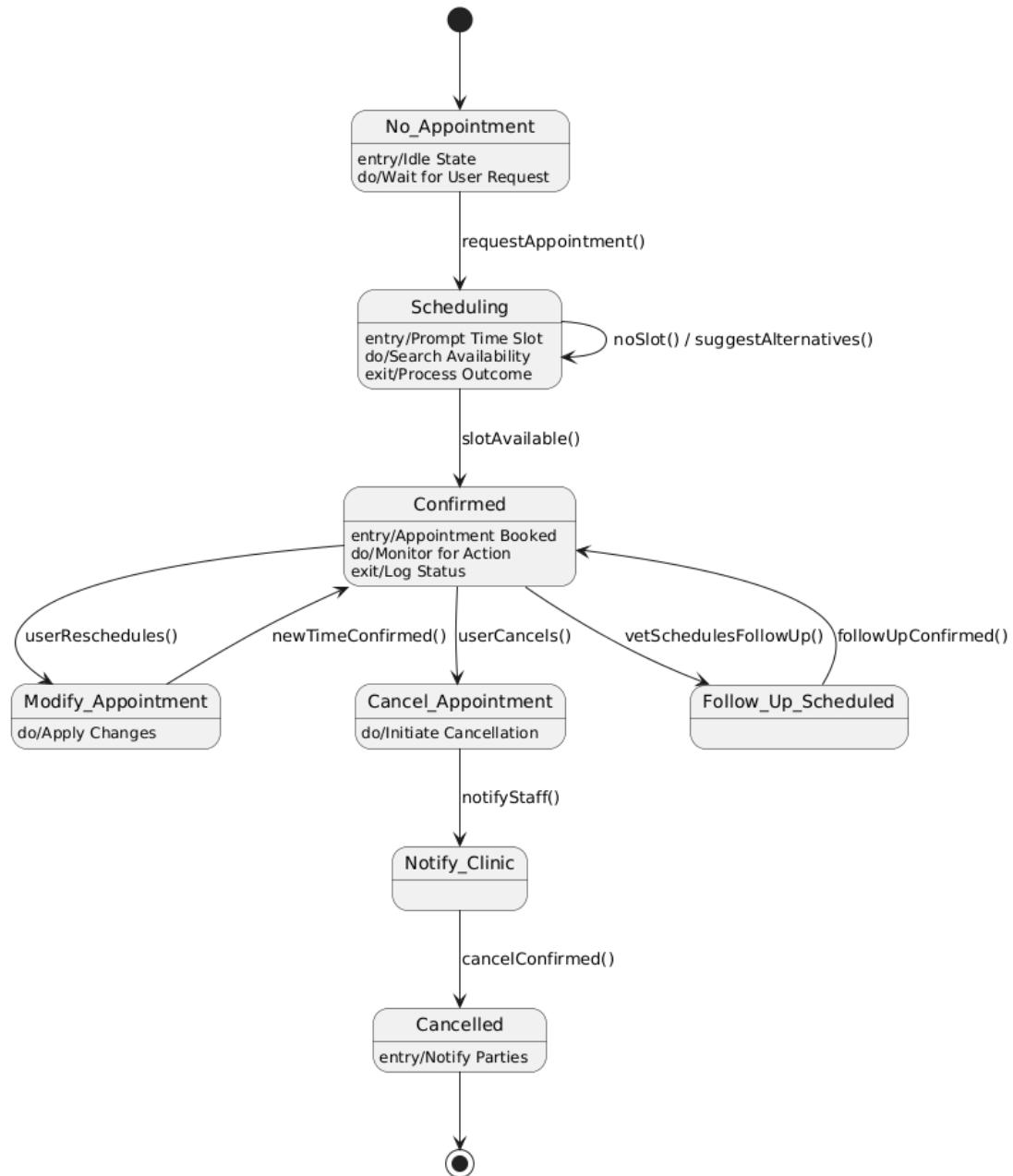
Edit permissions → and finish at Permissions_Updated

Each of these actions returns the system to an idle state, ready for the next admin task.

This diagram summarizes how the system handles all user management tasks through a clear flow of state transitions."

Veterinary Management System Requirements Specification

State Diagram 02 - Appointment Lifecycle



2. Appointment Lifecycle Era (Covers UC-REC-01, UC-PAT-01, UC-003, UC-VET-03)
present it like this :

"This is the state diagram for the Appointment Lifecycle. It starts at No_Appointment, where no booking exists. From there, a user can schedule an appointment, moving into the Confirmed state.

From Confirmed, the appointment can be:

Modified (rescheduled), which loops back to Confirmed, Cancelled, which goes to Cancelled, or

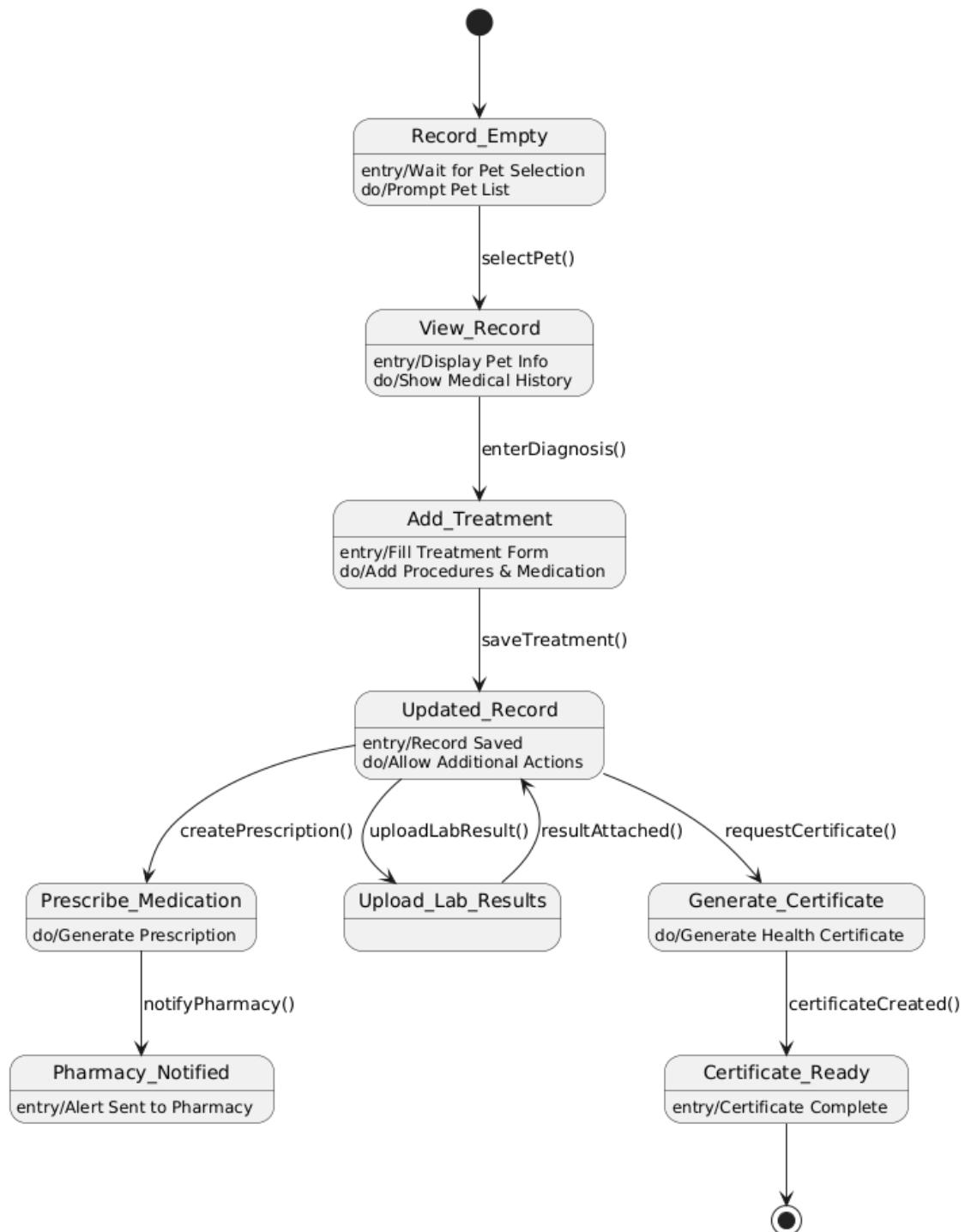
A follow-up can be scheduled, which also returns to Confirmed.

We don't connect Confirmed to the end state because the appointment remains active until further action is taken."

Veterinary Management System Requirements Specification

3. **Pet Record Era (Covers UC- VET-01, UC-VET-02, UC-LAB-01, UC-VET-06, UC-VET-07)**

State Diagram 03 - Pet Record Management



Veterinary Management System Requirements Specification

State Diagram 04 - Payment & Reconciliation

4. Enea(Covers UC-REC-02, UC-FIN-01, UC-FIN-02, UC-020)

explain like this:

"breakdown of the diagram:

Starting State: The process begins in the Invoice_Pending state, where the invoice is awaiting payment.

Payment Processing: From Invoice_Pending, the system moves to Payment_Processing where it attempts to process the payment.

Success or Failure:

If the payment is successful, the process moves to Paid.

If the payment fails, the process transitions to Retry, allowing another attempt to process the payment, and then goes back to Payment_Processing.

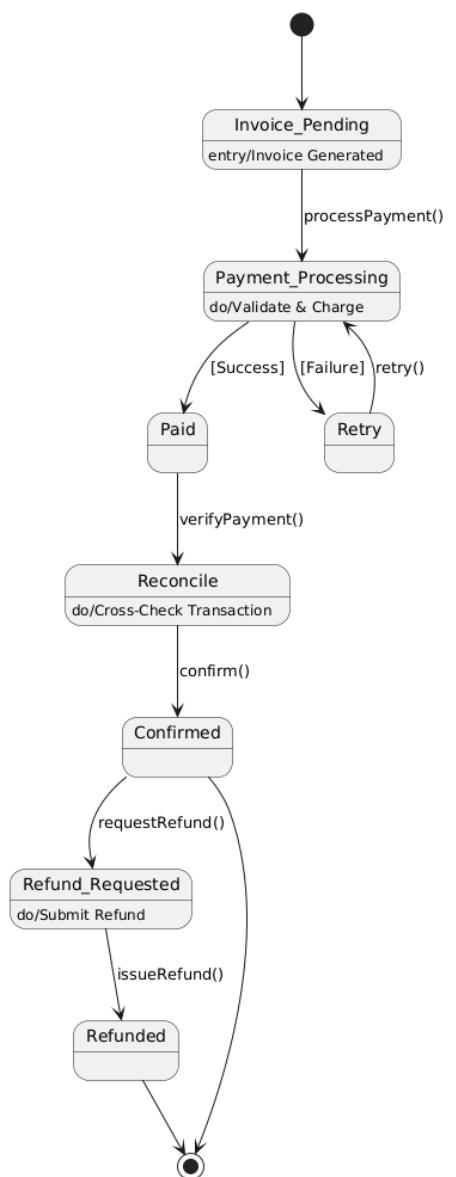
Reconciliation: After payment is successful, the system enters the Reconcile state, where it verifies the payment details. Confirmed: Once reconciliation is complete, it moves to Confirmed indicating everything is in order.

Refund Process: If there's a cancellation or overpayment, the system transitions to Refund_Requested and, once approved, moves to Refunded.

Final State: After a refund is issued, the process ends in the [*] state.

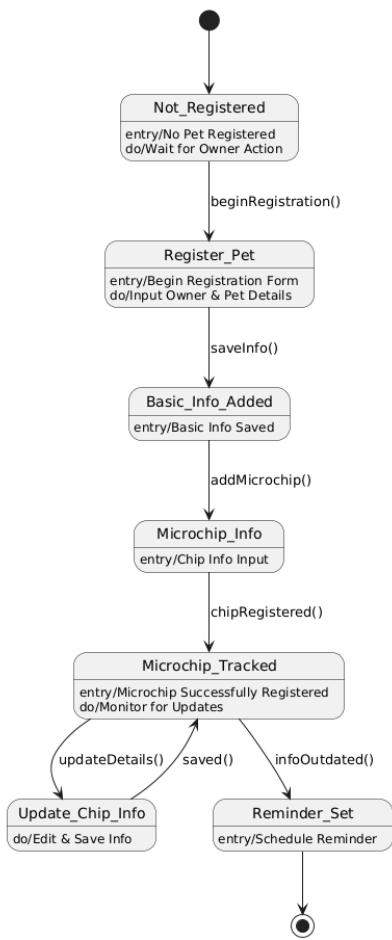
In essence, it's a payment processing flow that handles retries on failure and allows for refunds when necessary."

5. Laert(Covers UC-REC-03, UC-029)



Veterinary Management System Requirements Specification

State Diagram 05 - Pet Registration and Microchip Tracking



explain like this :

"explanation of the diagram:

Starting State: The process begins with the Not_Registered state, meaning the pet is not yet registered.

Registering Pet: From Not_Registered, the system moves to Register_Pet, where the pet registration process starts.

Adding Basic Info: Once the pet is registered, the system moves to Basic_Info_Added to input essential information like the pet's name, breed, etc.

Microchip Info: Next, the system transitions to Microchip_Info, where the pet's microchip details are added.

Microchip Tracked: Afterward, it enters the Microchip_Tracked state, where the microchip information is stored and monitored.

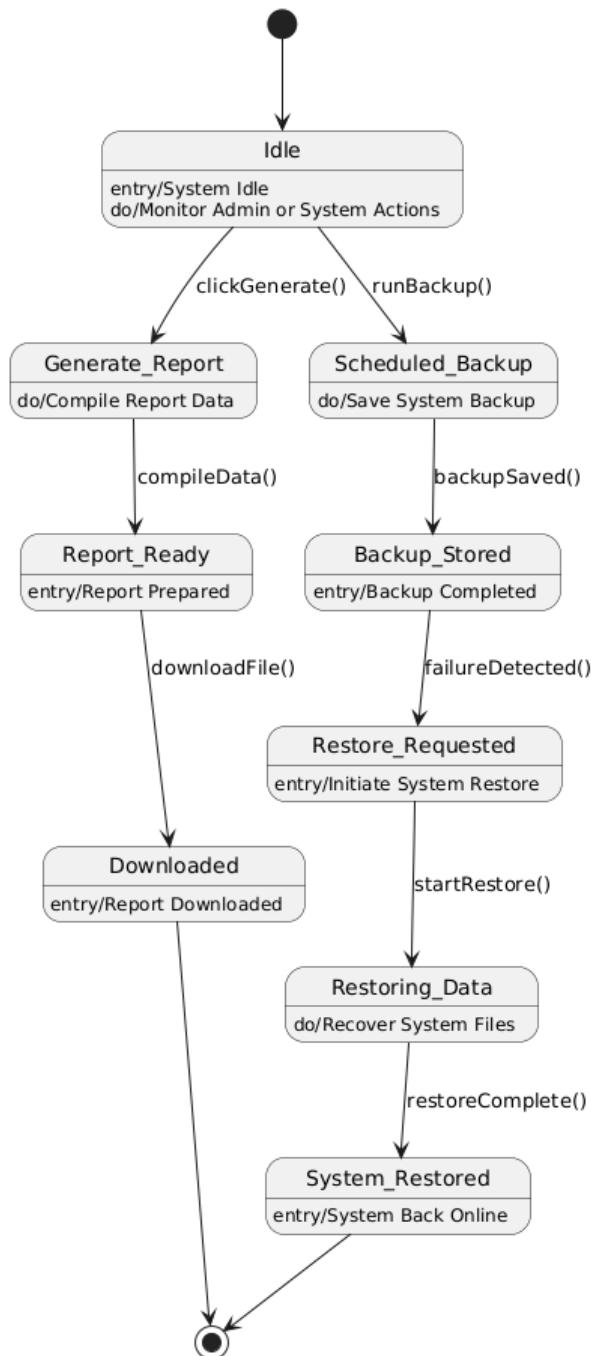
Updating Chip Info: If the owner updates the microchip info, the system goes to Update_Chip_Info and then returns to Microchip_Tracked to keep the data up-to-date.

Reminder: If the microchip info becomes outdated or needs updating, the system moves to Reminder_Set, where a reminder is scheduled for the owner to update the information.

Final State: After the reminder is set, the process ends in the [*] state."

Veterinary Management System Requirements Specification

State Diagram 06 - Clinic Reporting Lifecycle



6. Clinic Reporting Erla (Covers UC-ADM-04, UC-FIN-01, UC-SYS-02)

breakdown of the diagram:

Starting State: The process starts in the Idle state, meaning the system is not actively performing any tasks.

Generating Report: If an Admin or Finance Officer requests it, the system moves to Generate_Report, where a report is created.

Report Ready: Once the report is generated, the system enters the Report_Ready state, indicating the report is prepared for download.

Downloaded: When the report is downloaded, the process finishes and moves to the [*] state.

Scheduled Backup: While idle, the system can also initiate a Scheduled_Backup to back up the data.

Backup Stored: After the backup is completed, the system transitions to Backup_Stored, where the backup is safely stored.

Restoring Data: If there's a System Failure, the process moves to Restore_Requested. The system then begins restoring the data in Restoring_Data.

System Restored: After the restoration is complete, the system enters the System_Restored state, indicating the system is back to normal.

Final States: The process ends in the [*] state after the report is downloaded or the system is restored.

Veterinary Management System Requirements Specification

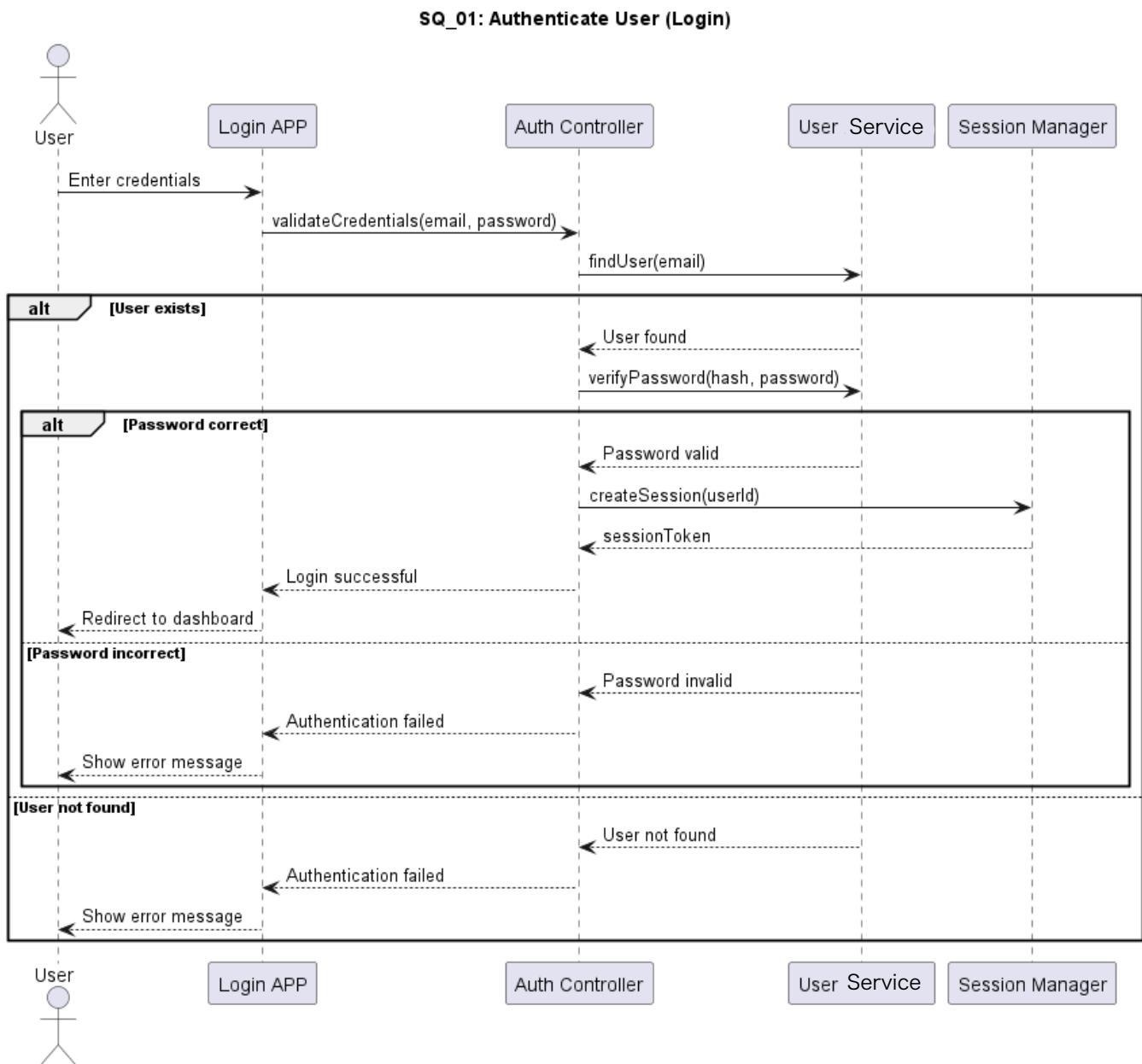
5.6 Sequence diagram.

All sequence diagrams are associated with an Activity Diagram. A Sequence Diagram is built based on an activity diagram. If the activity diagram is named AC_07, the Sequence Diagram will be named SC_07.

1	Authenticate User (Login)	Admin, Receptionist, Vet, Finance Officer, Pet Owner	
2	Add New User	Admin	
3	Deactivate User	Admin	
4	Generate Clinic Activity Report	Admin	
5	View Pet Medical Record	Vet, Receptionist, Pet Owner	
6	Prescribe Medications	Vet	
7	Schedule Follow-Up Appointment	Vet	
8	Schedule Appointment	Receptionist, Pet Owner	
9	Process Payment	Receptionist, Pet Owner	
10	Register New Patient (Pet)	Receptionist	
11	Reschedule Appointment	Receptionist, Pet Owner	
12	View Pet Health Certificate	Pet Owner	
13	Generate Financial Report	Finance Officer	
14	Reconcile Payments	Finance Officer	
15	Backup Database	Admin	
16	Restore Data	Admin	
17	Upload Lab Results	Lab Technician	
18	Reorder Medication	Pharmacy Staff	
19	Triage Emergency Case	Vet	
20	Send Appointment Reminder	System (automated)	
	Cancel an Appointment	Receptionist, Pet Owner	21
	Assign a Vet to a Case	Receptionist	22
	Manage User Roles & Permissions	Admin	23
	Schedule Surgery	Vet	24
	Verify Pet Insurance	Receptionist	25
	Process Refunds	Finance Officer	26
	Request Home Visits	Pet Owner	27
	Rate and Review Services	Pet Owner	28
	Track Pet Microchip Info	Vet, Receptionist	29

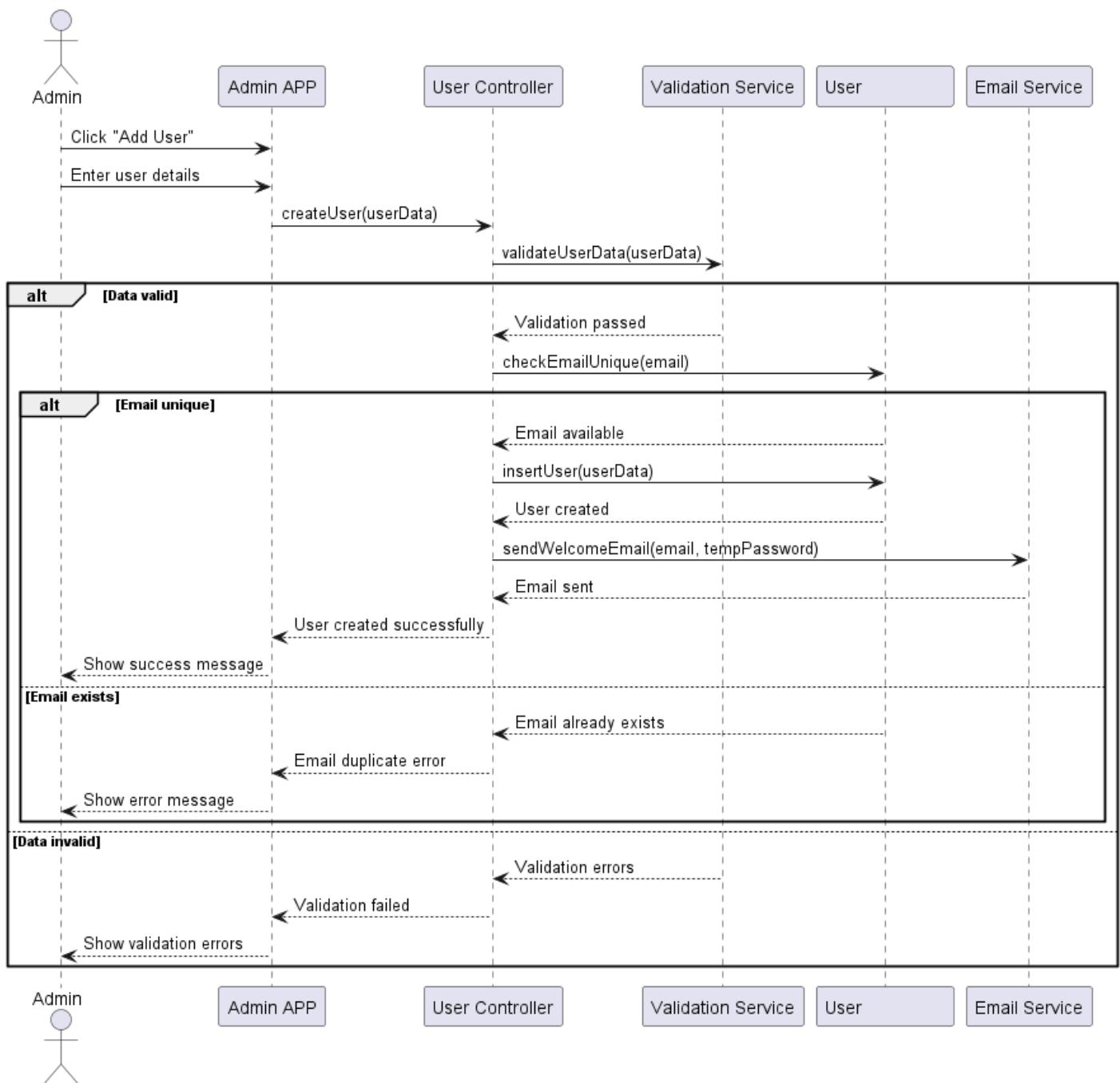
Veterinary Management System Requirements Specification

01-05 Enea Cane



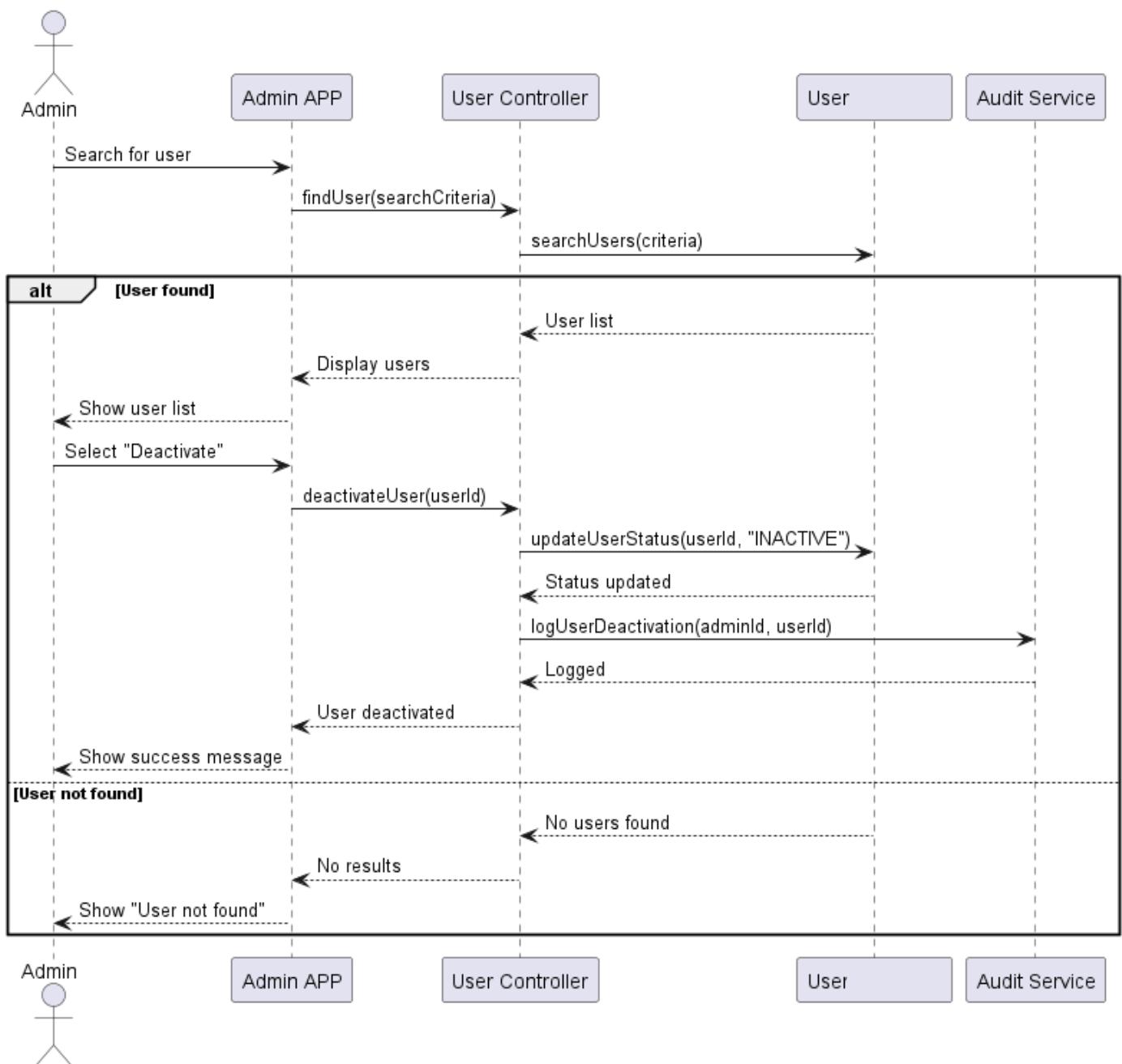
Veterinary Management System Requirements Specification

SQ_02: Add New User



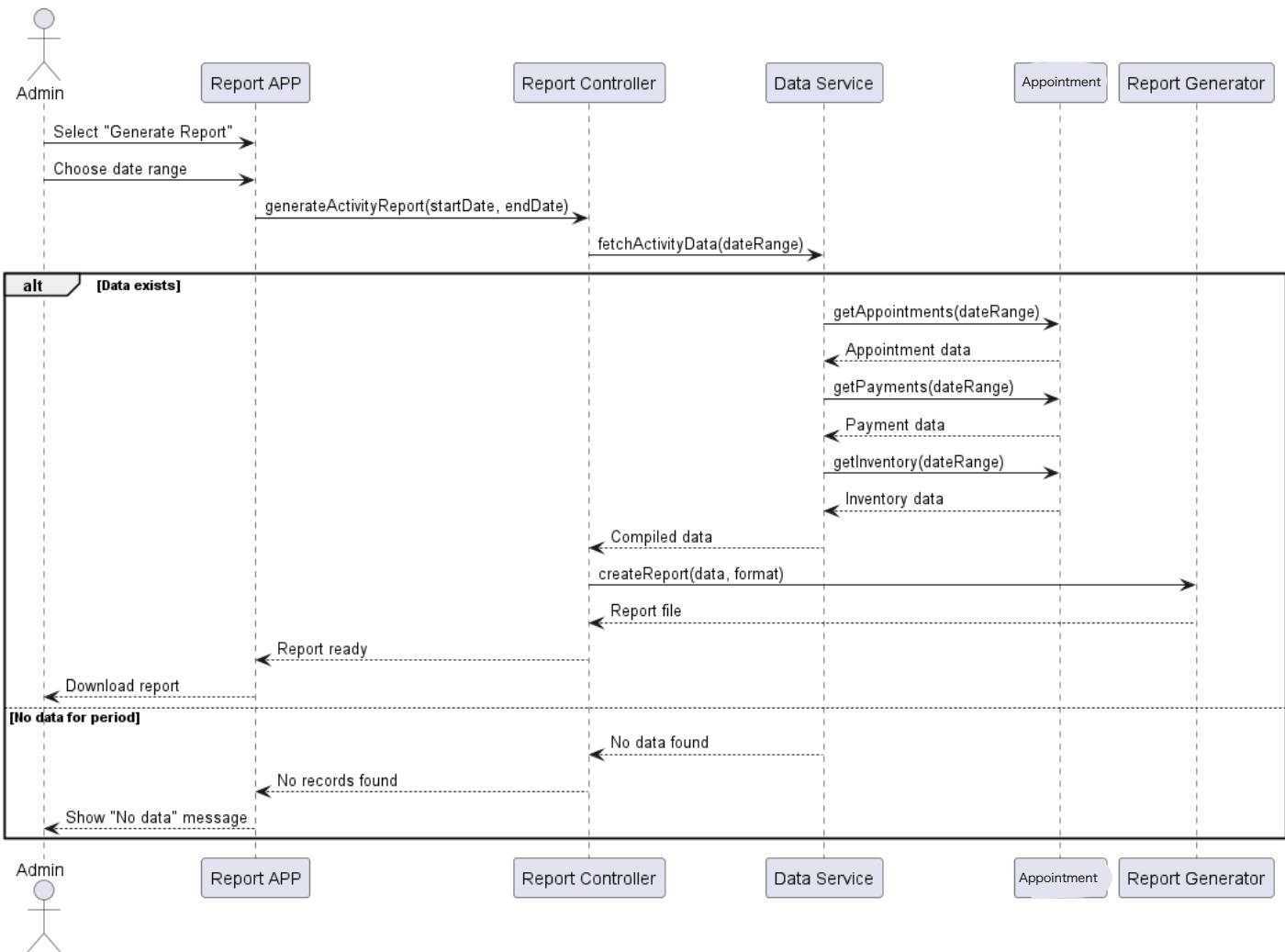
Veterinary Management System Requirements Specification

SQ_03: Deactivate User



Veterinary Management System Requirements Specification

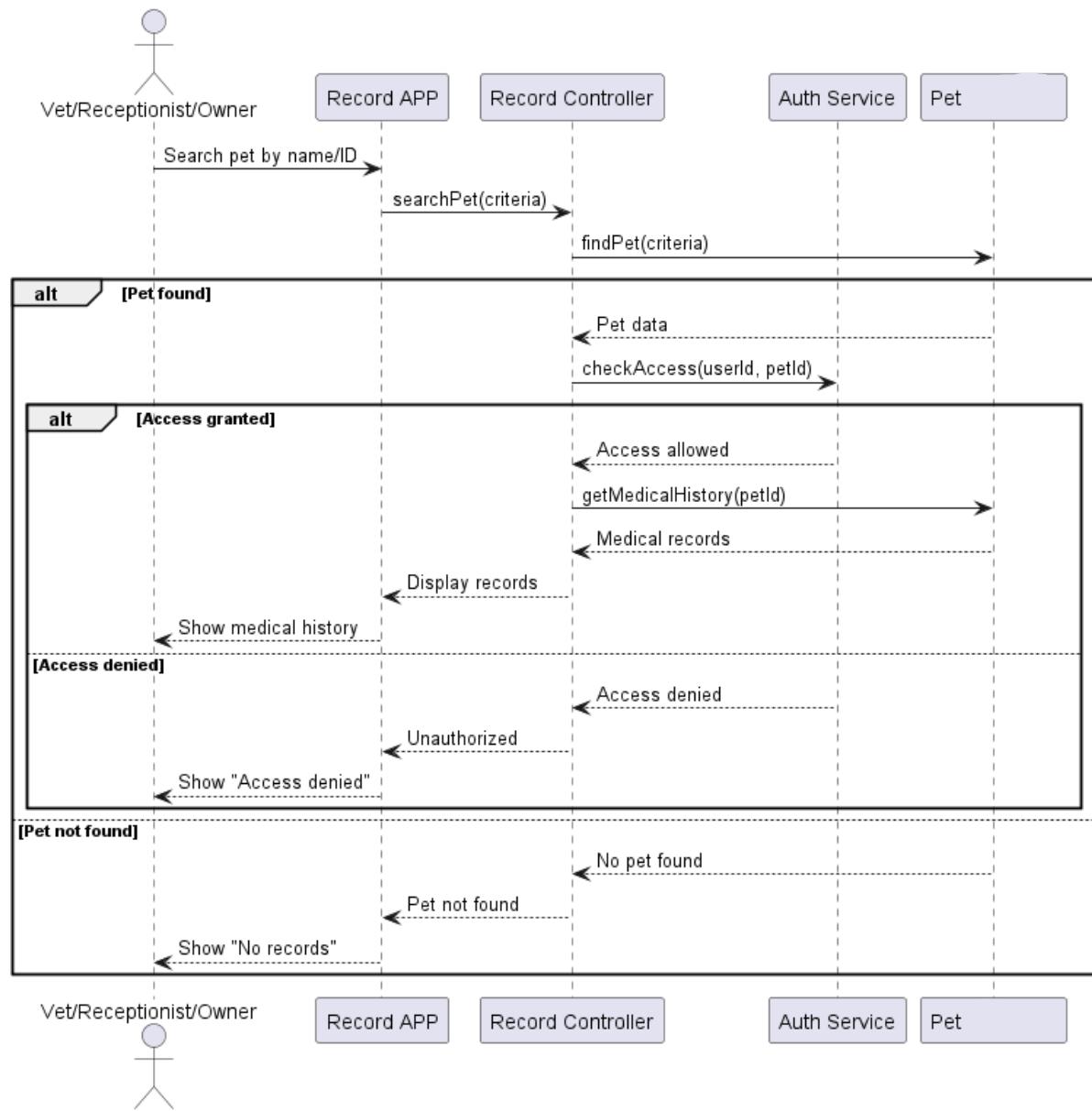
SQ_04: Generate Clinic Activity Report



05-

Veterinary Management System Requirements Specification

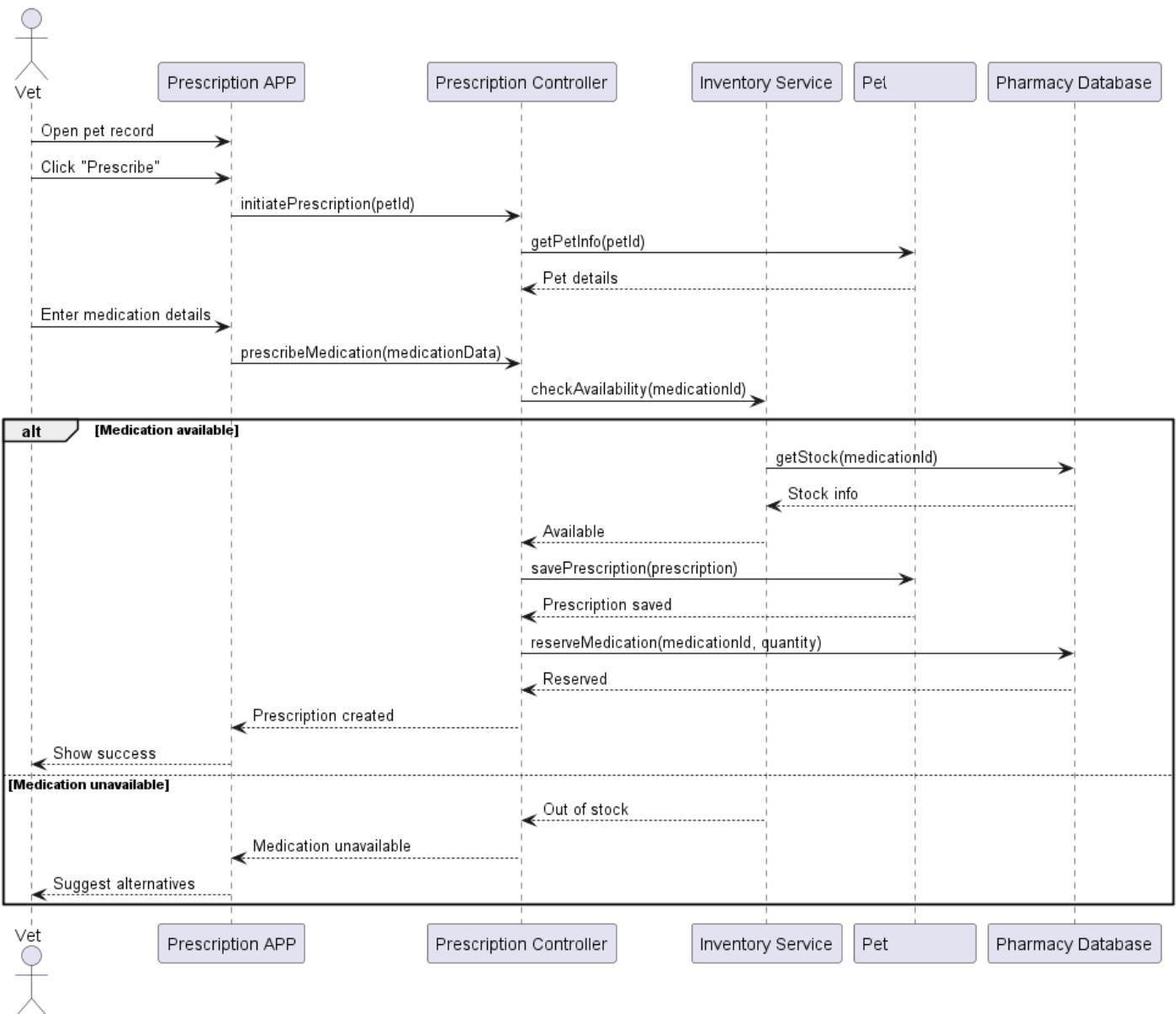
SQ_05: View Pet Medical Record



06-10 Laert

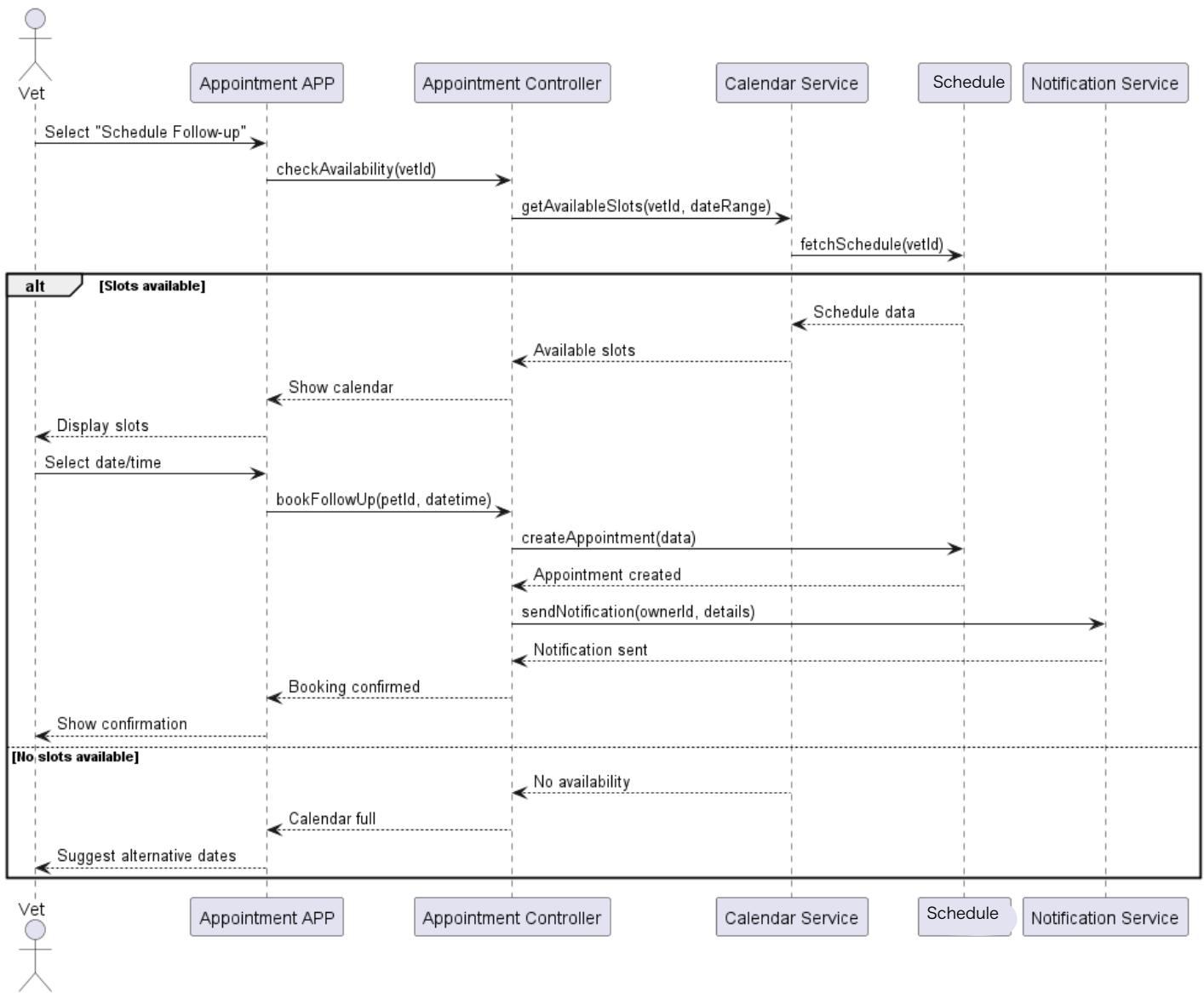
Veterinary Management System Requirements Specification

SQ_06: Prescribe Medications



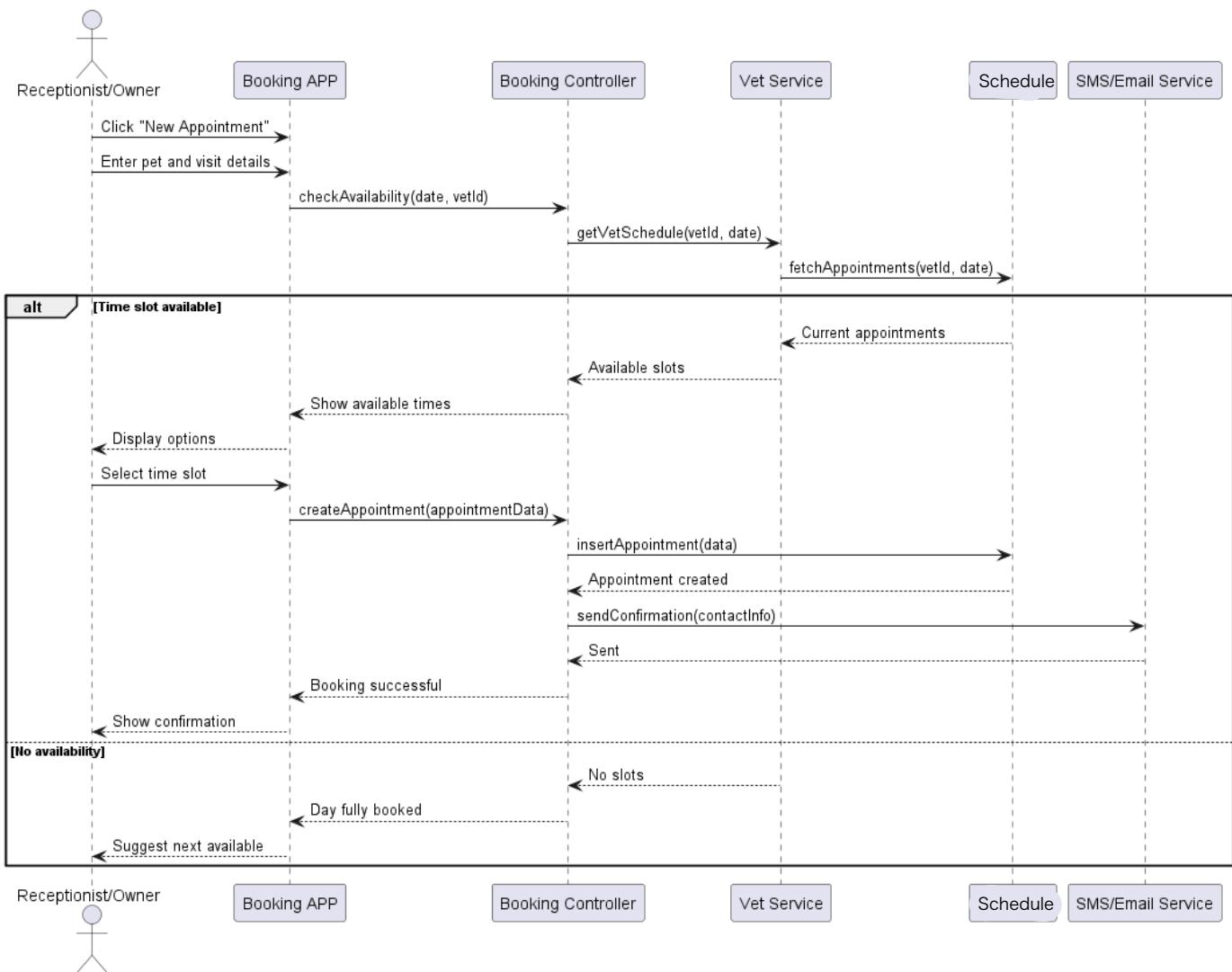
Veterinary Management System Requirements Specification

SQ_07: Schedule Follow-Up Appointment



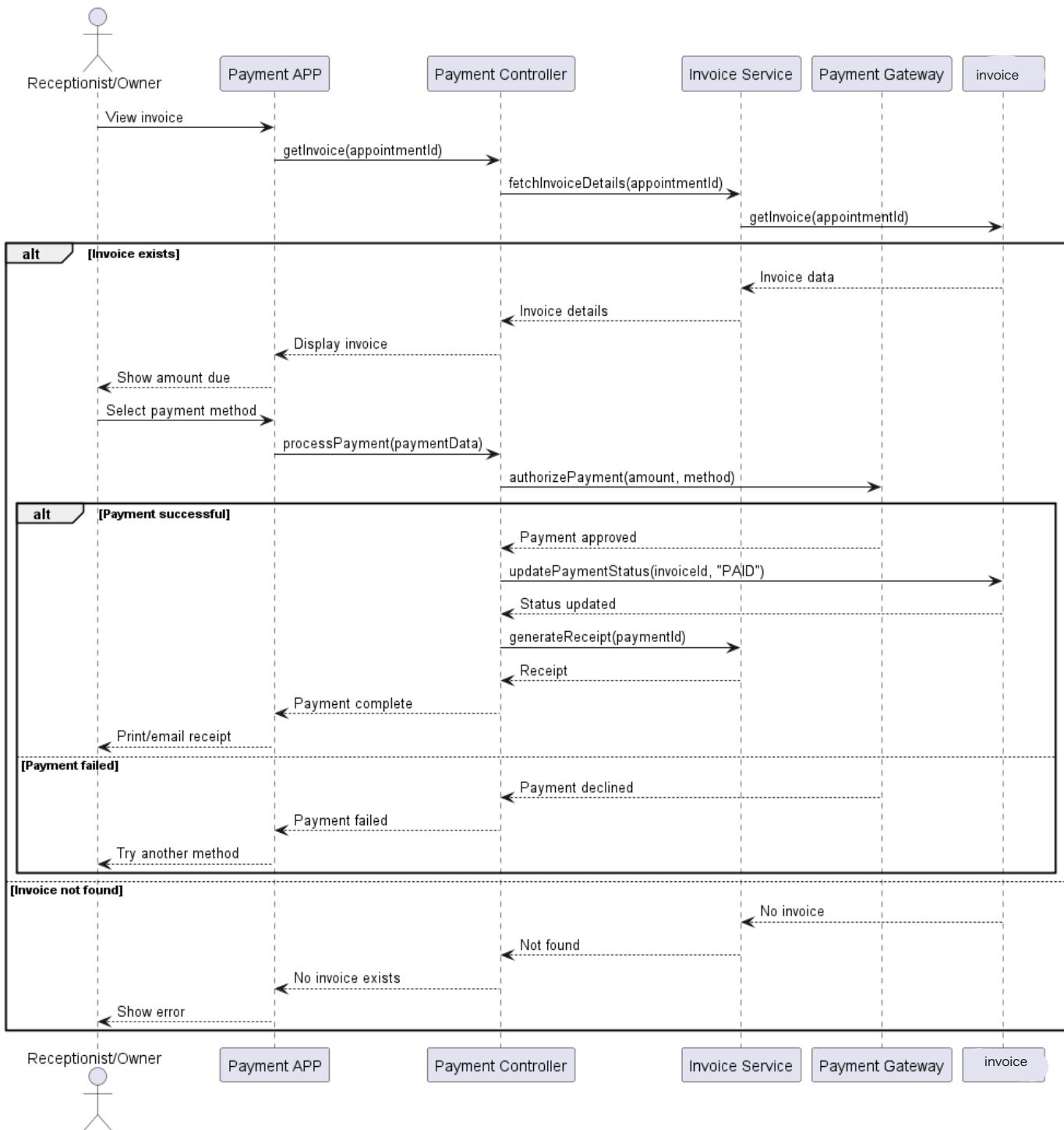
Veterinary Management System Requirements Specification

SQ_08: Schedule Appointment



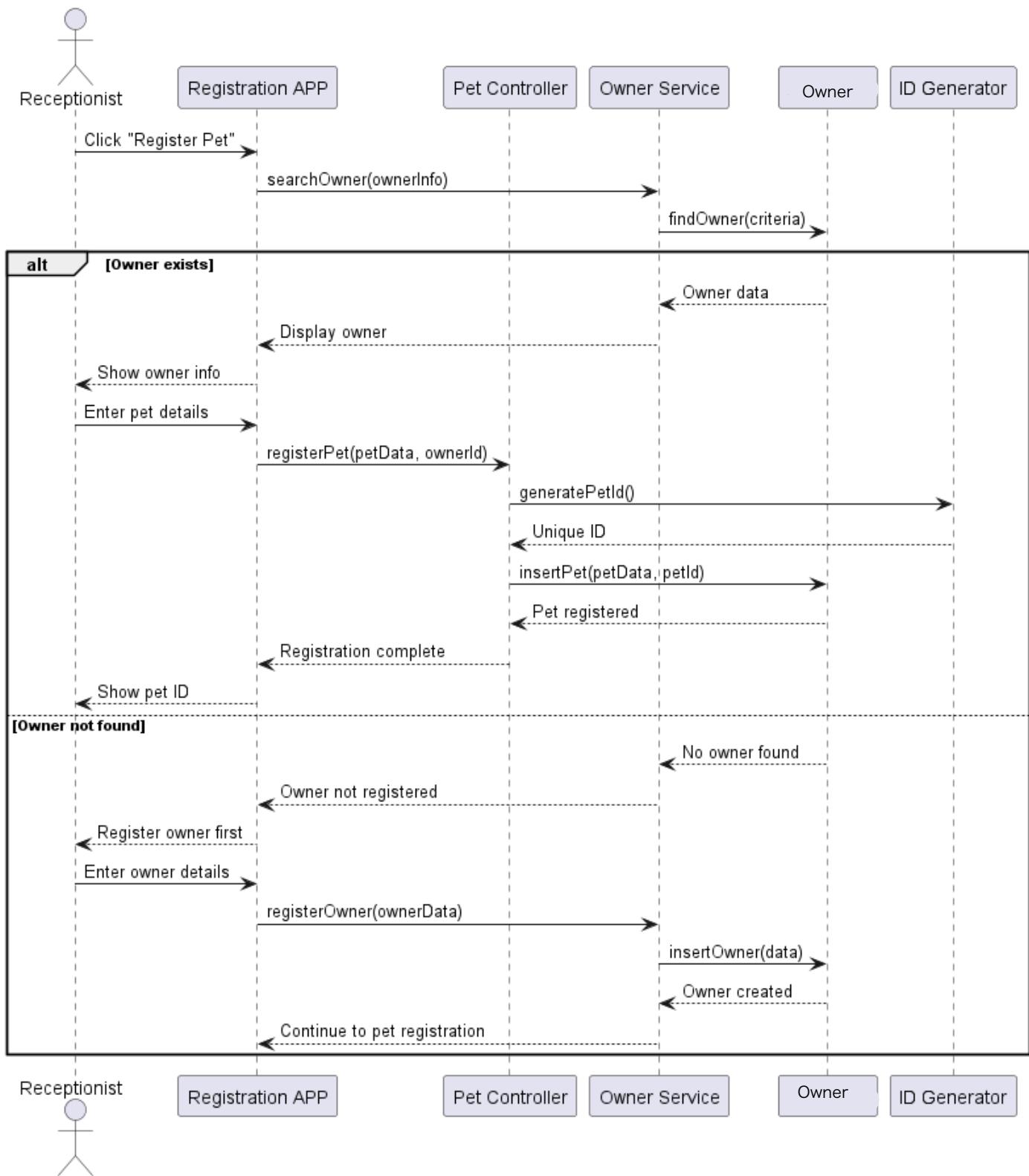
Veterinary Management System Requirements Specification

SQ_09: Process Payment



Veterinary Management System Requirements Specification

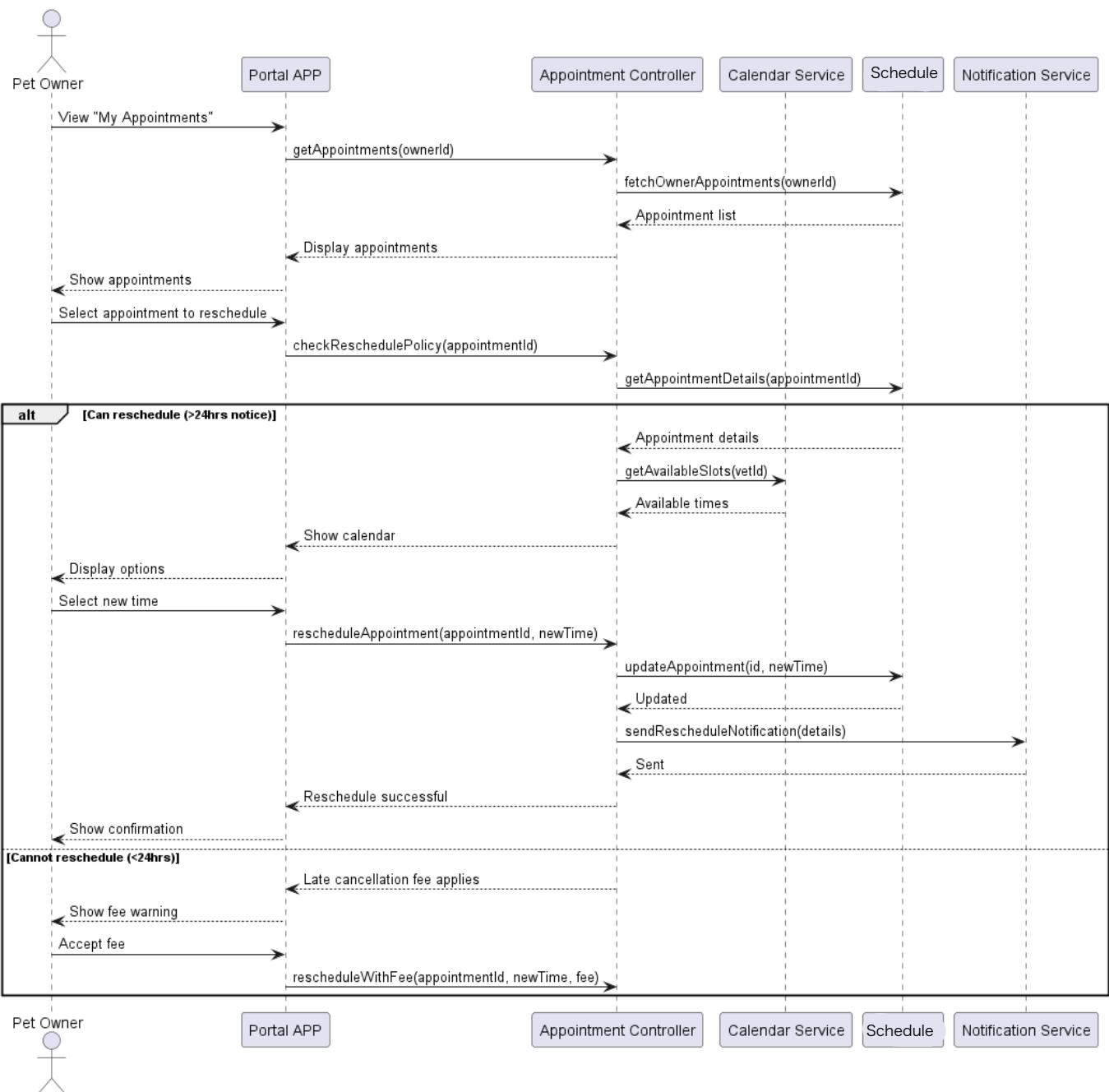
SQ_10: Register New Patient (Pet)



11-16 Erla

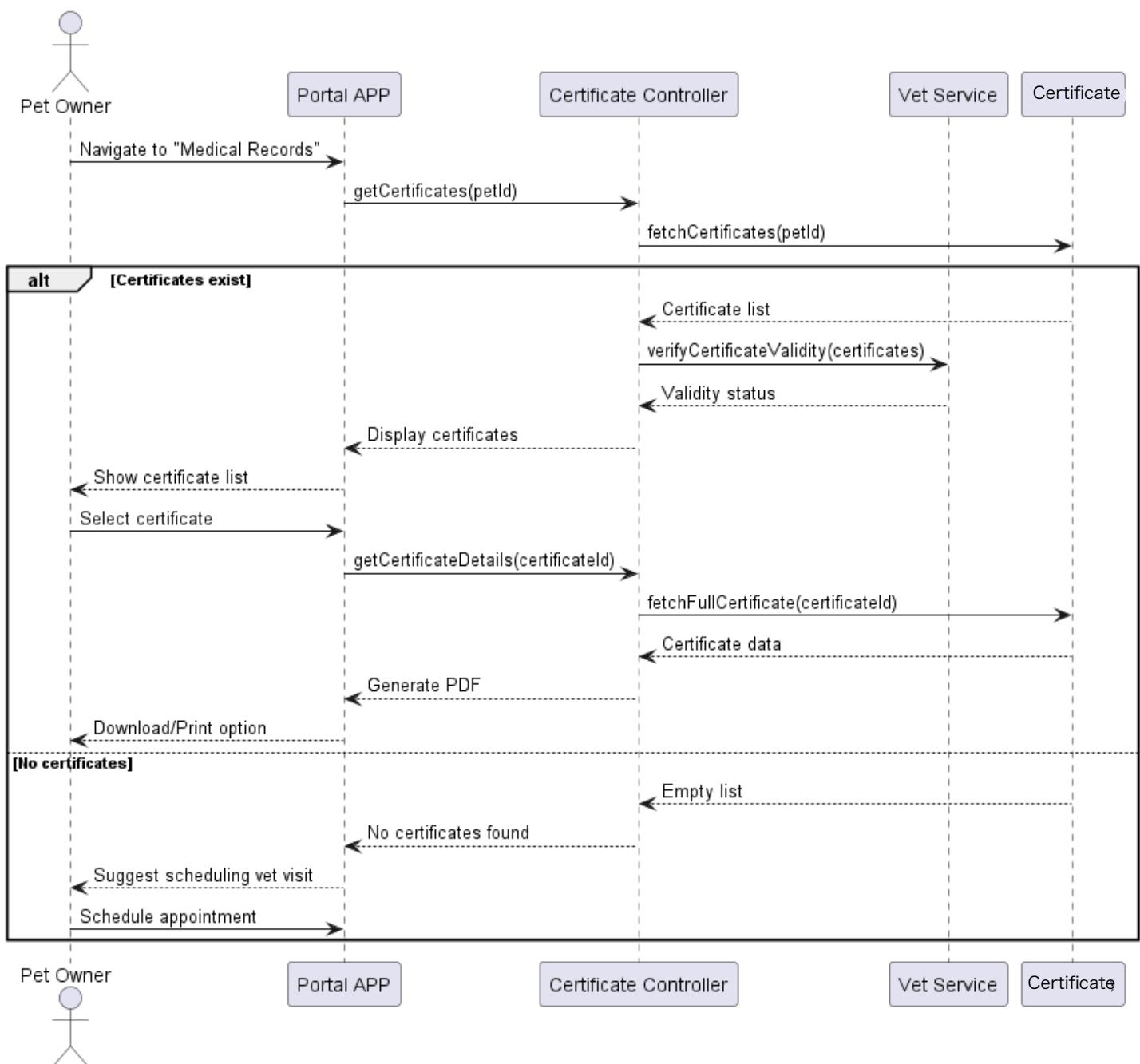
Veterinary Management System Requirements Specification

SQ_11: Reschedule Appointment



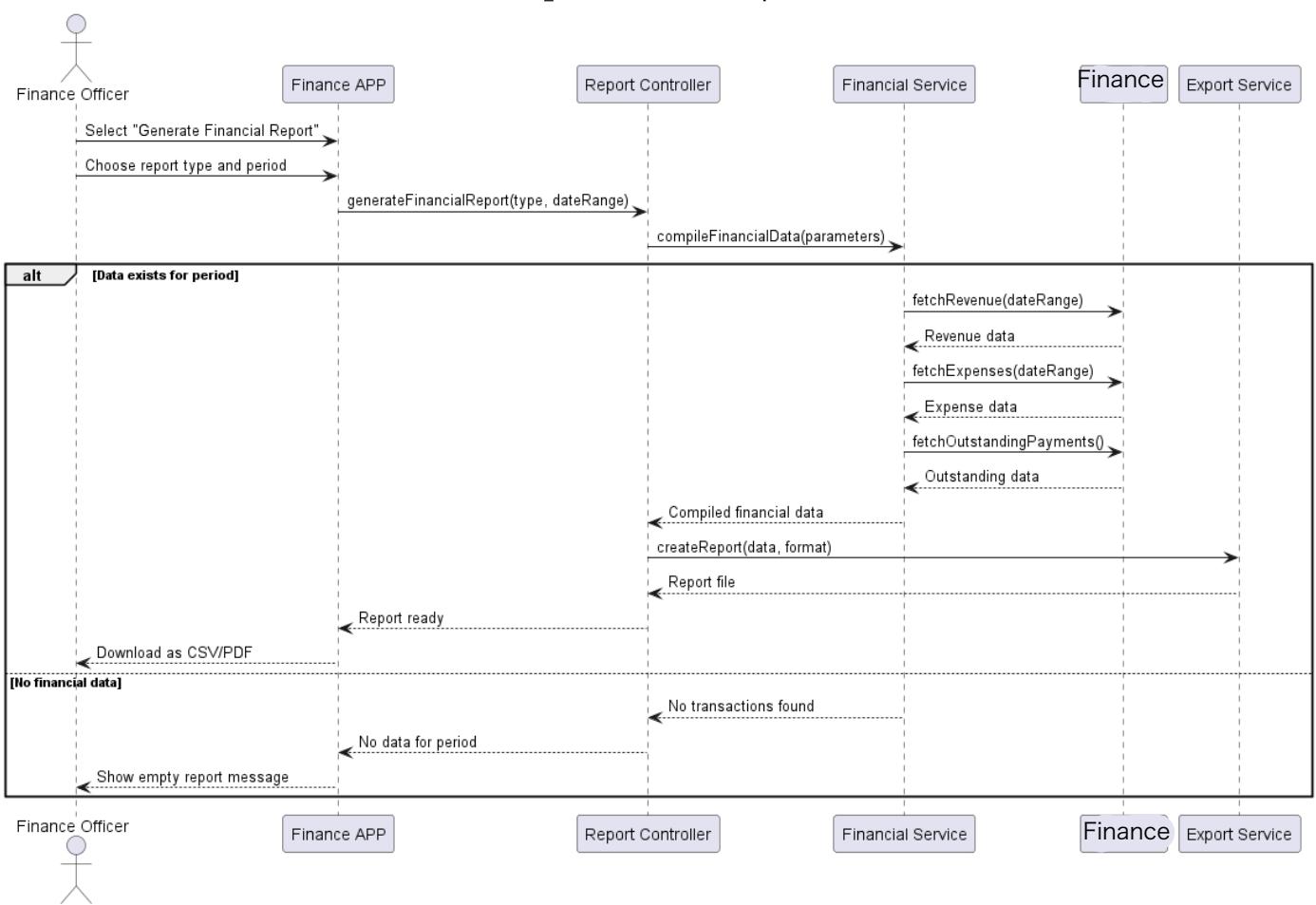
Veterinary Management System Requirements Specification

SQ_12: View Pet Health Certificate



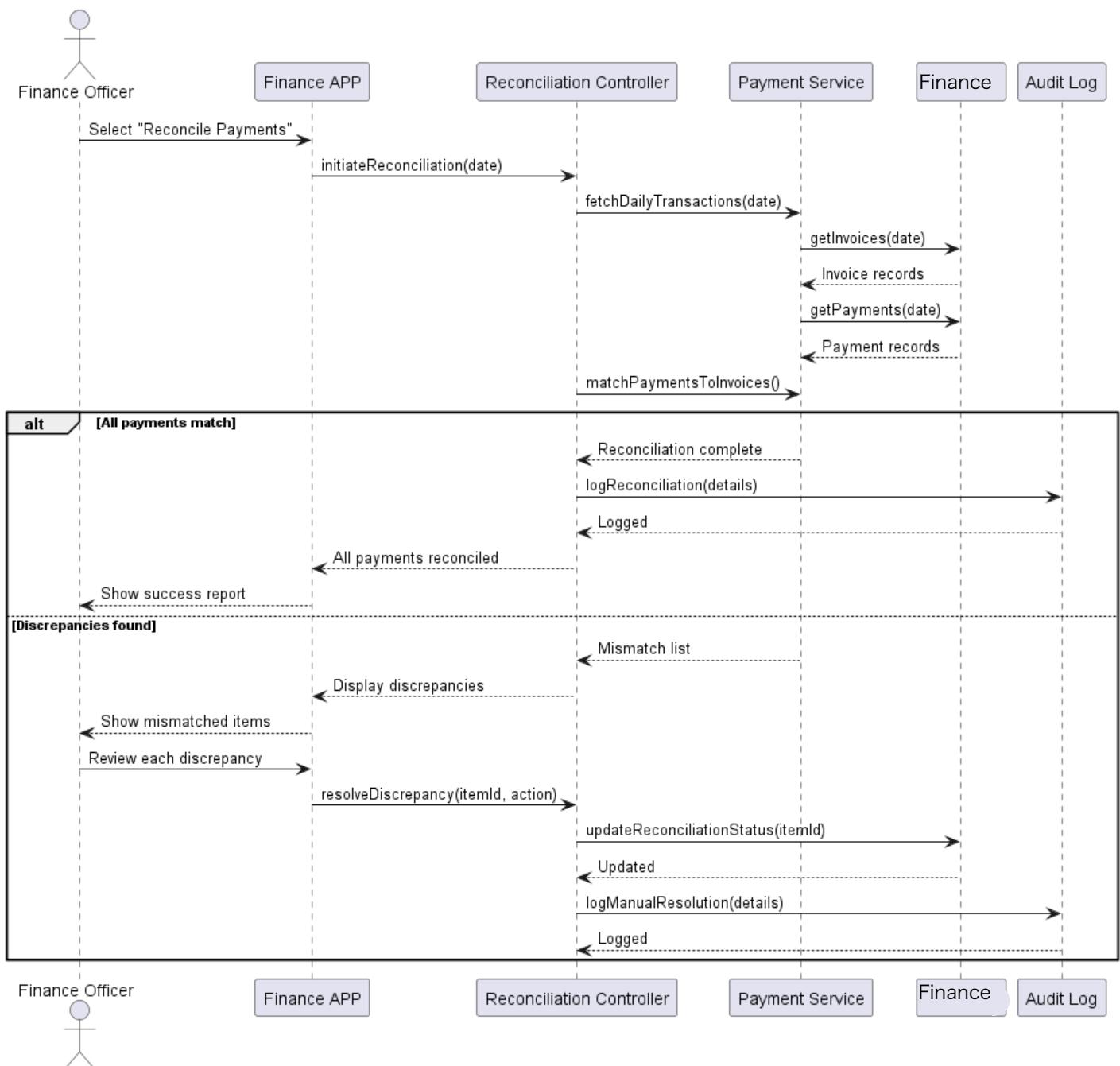
Veterinary Management System Requirements Specification

SQ_13: Generate Financial Report



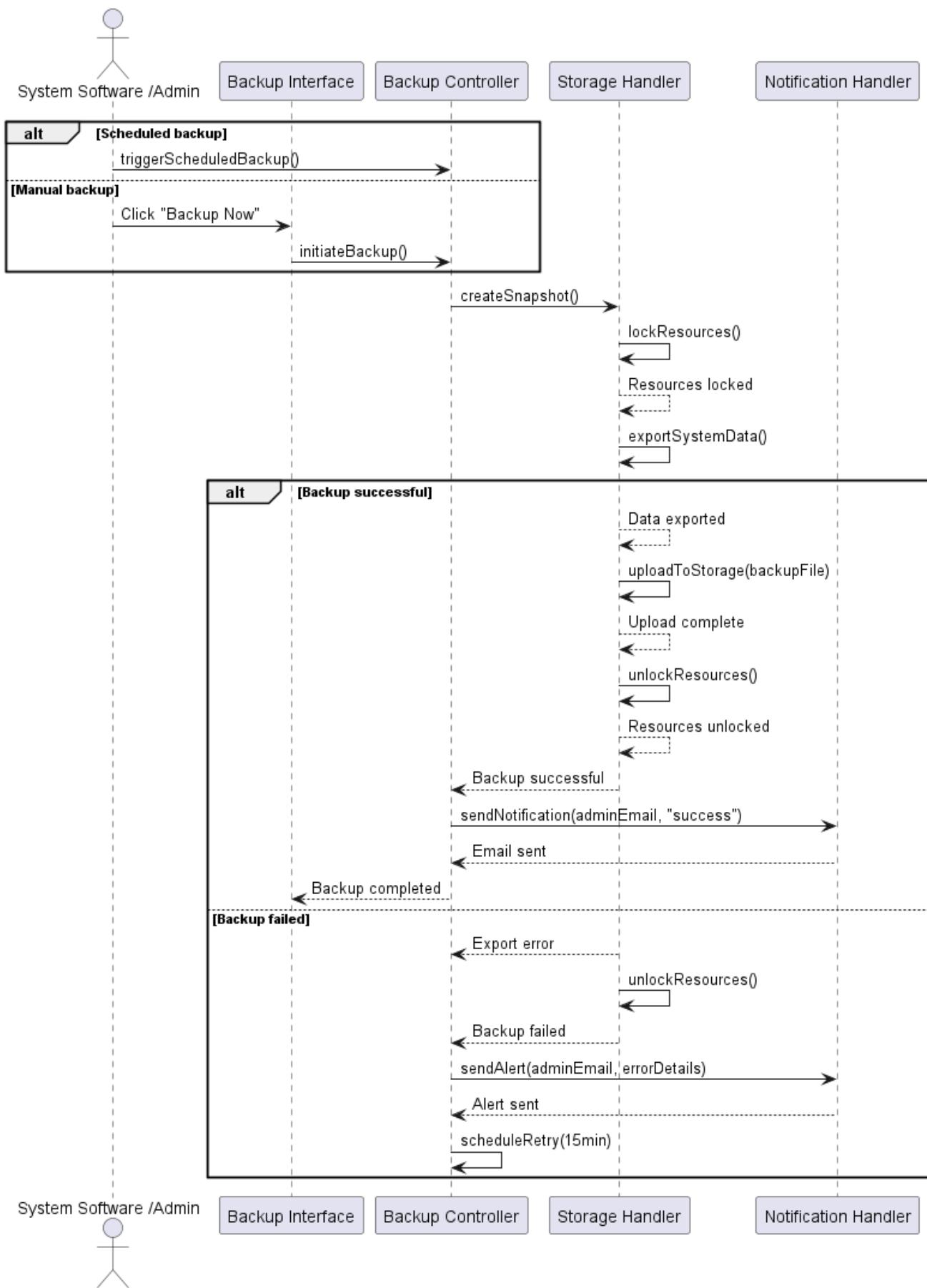
Veterinary Management System Requirements Specification

SQ_14: Reconcile Payments



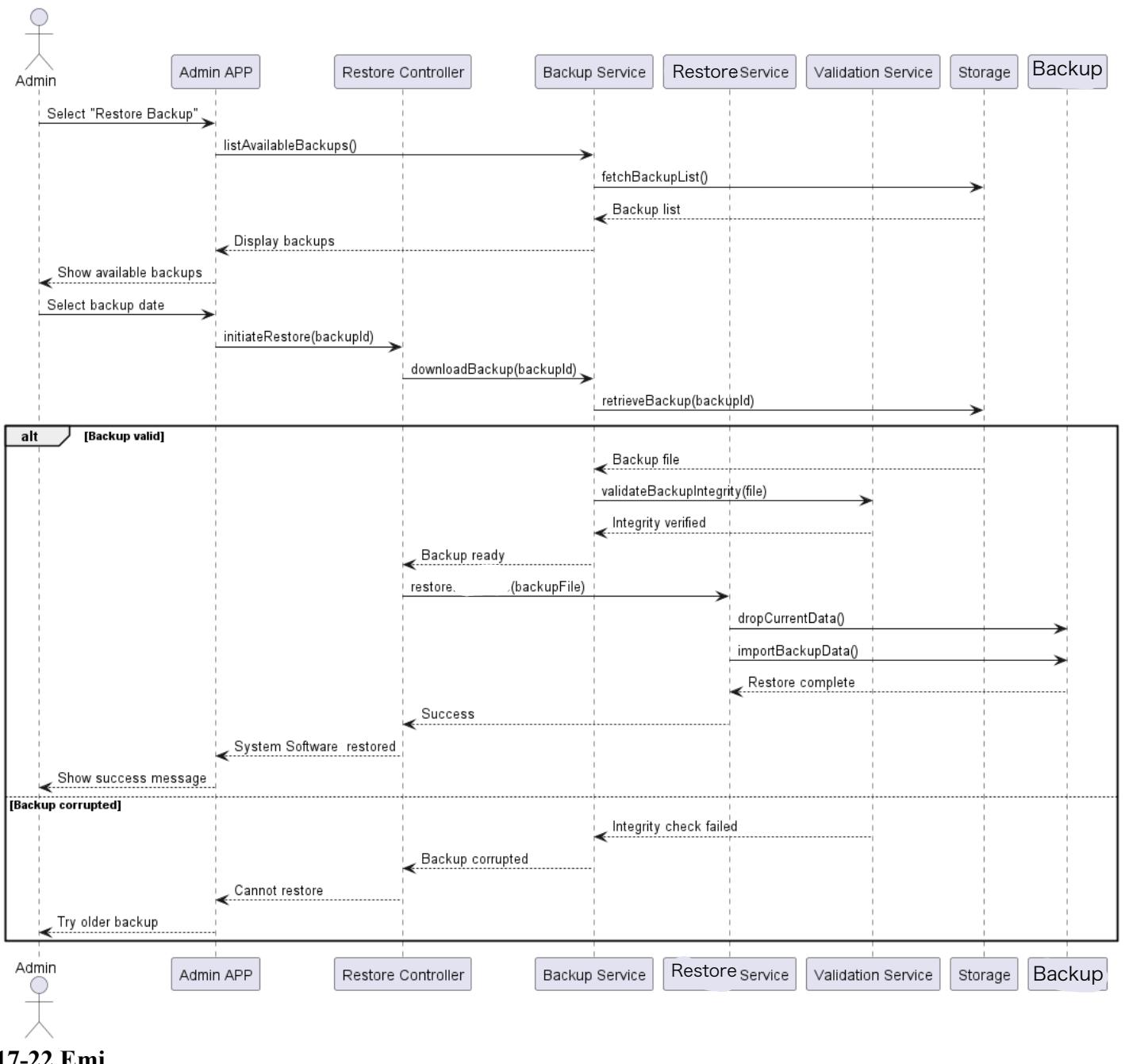
Veterinary Management System Requirements Specification

SQ_15: Backup System Data



Veterinary Management System Requirements Specification

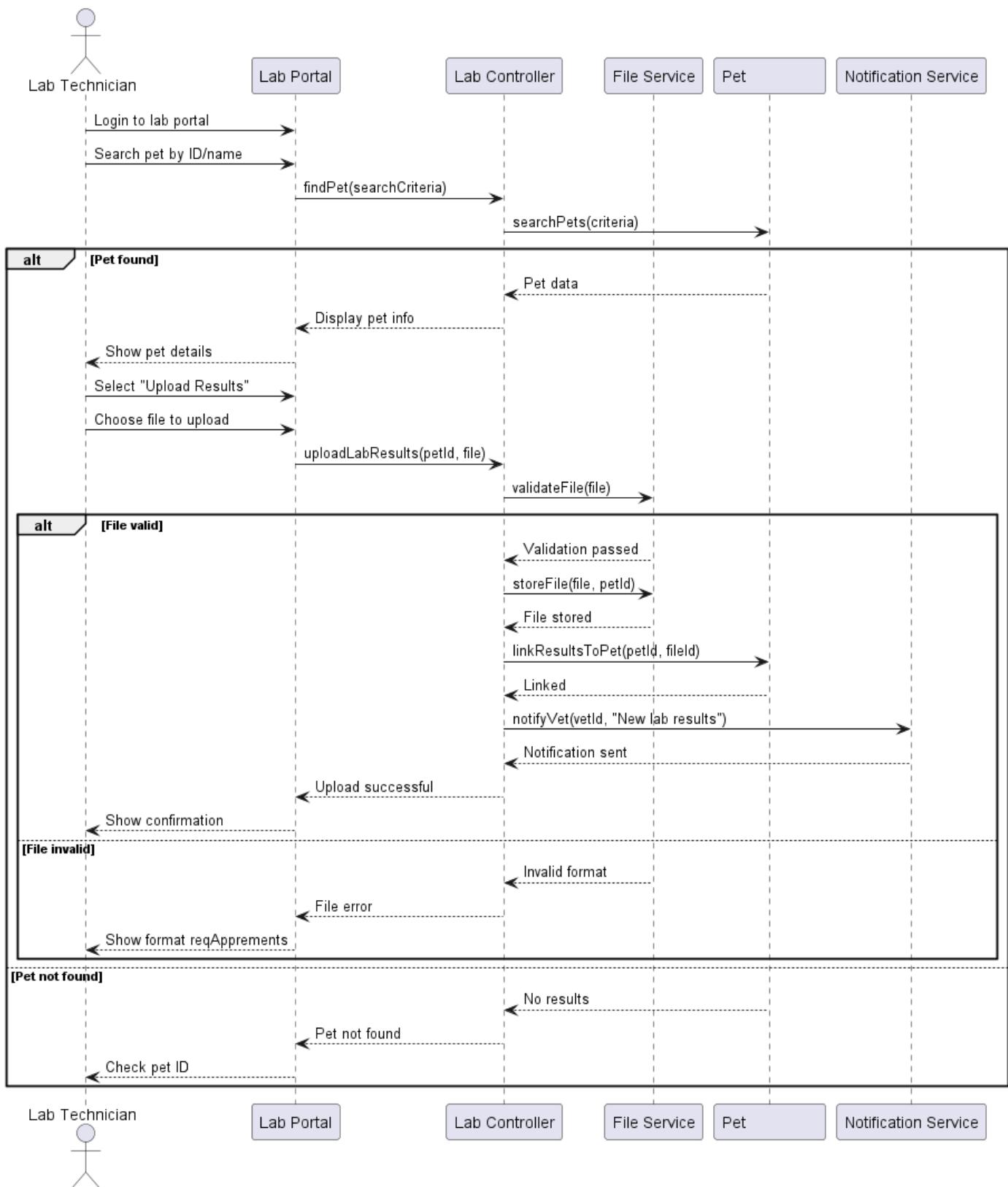
SQ_16: Restore Data



17-22 Emi

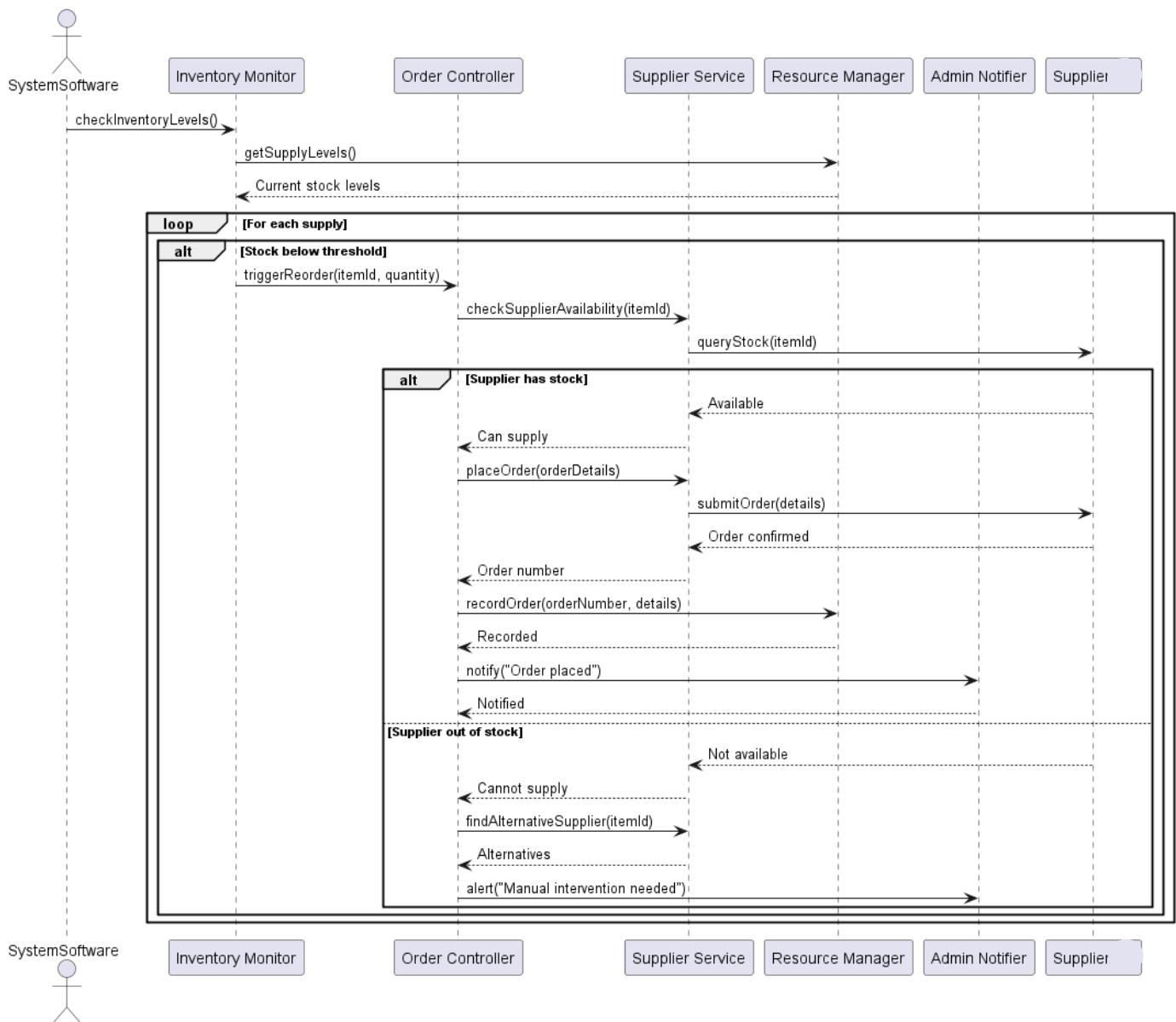
Veterinary Management System Requirements Specification

SQ_17: Upload Lab Results



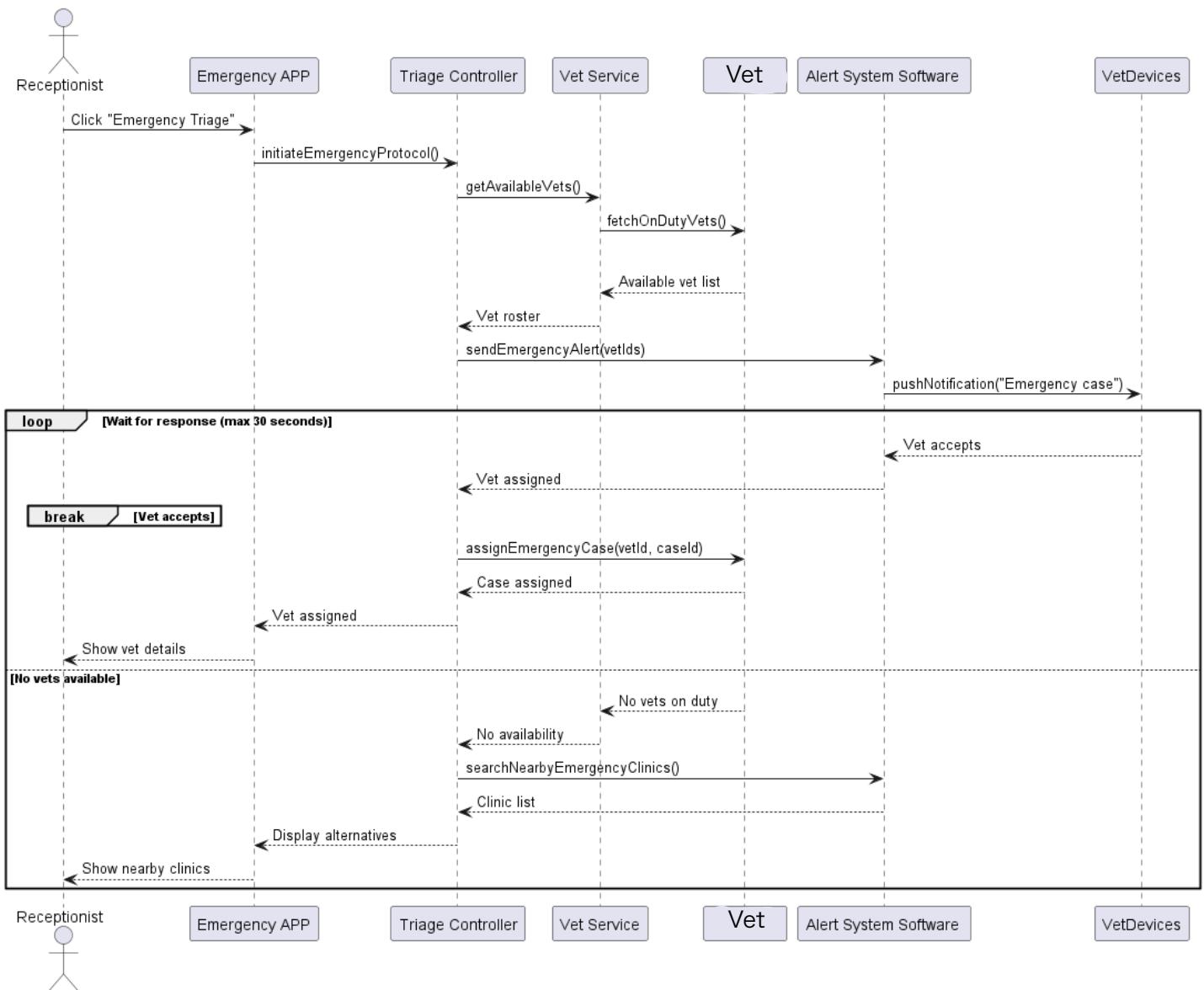
Veterinary Management System Requirements Specification

SQ_18: Reorder Medical Supplies



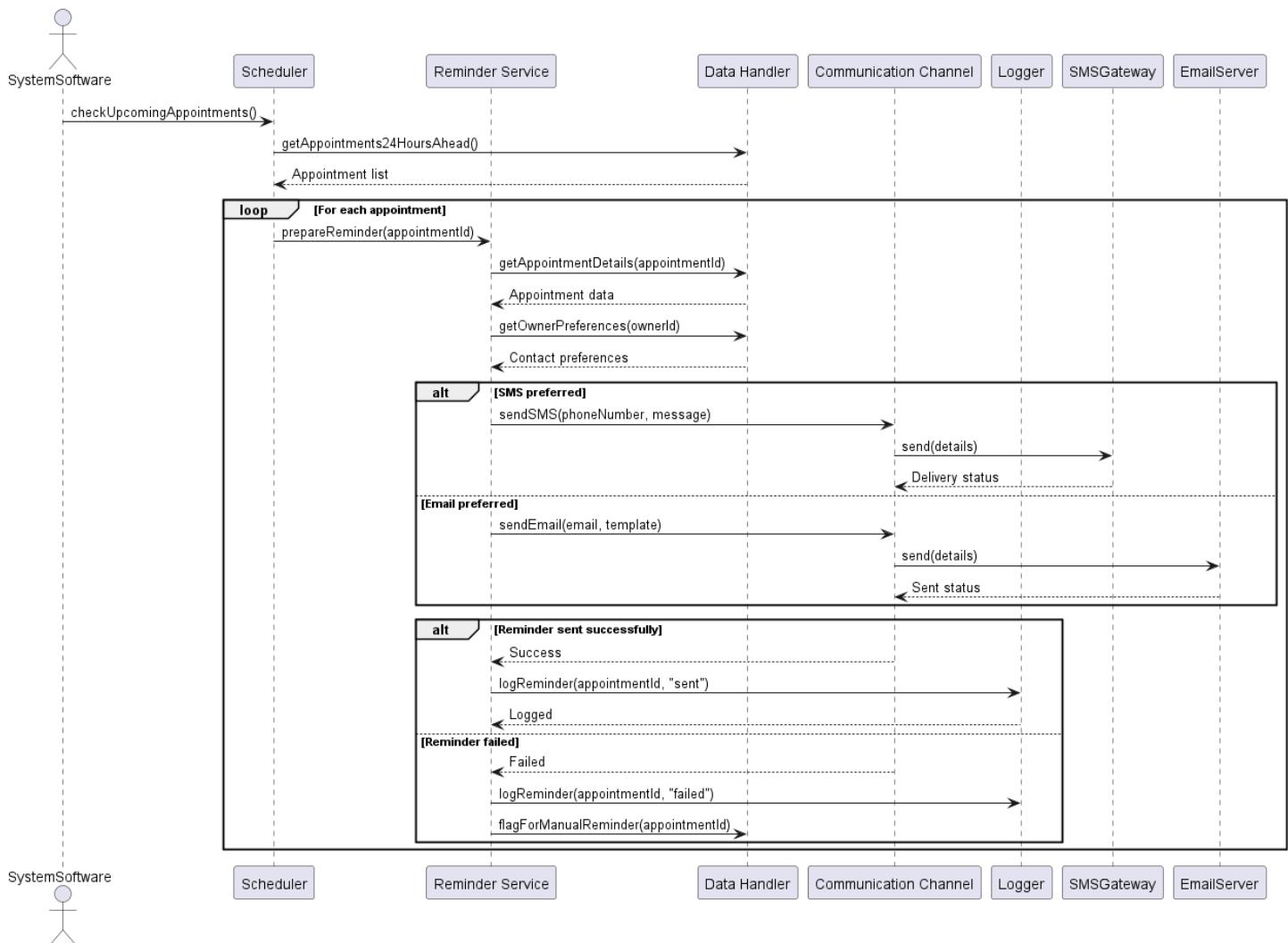
Veterinary Management System Requirements Specification

SQ_19: Triage Emergency Case



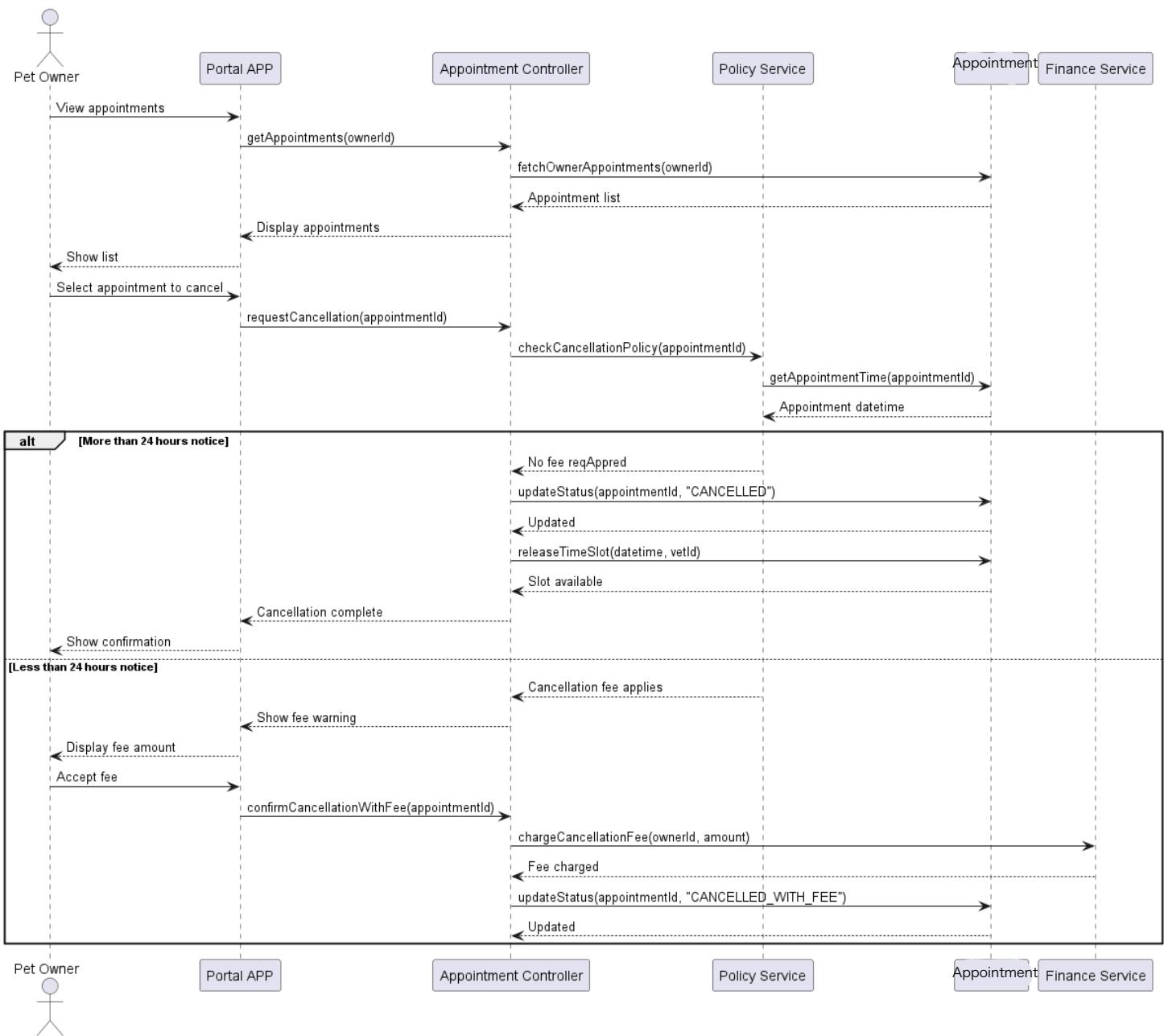
Veterinary Management System Requirements Specification

SQ_20: Send Appointment Reminder



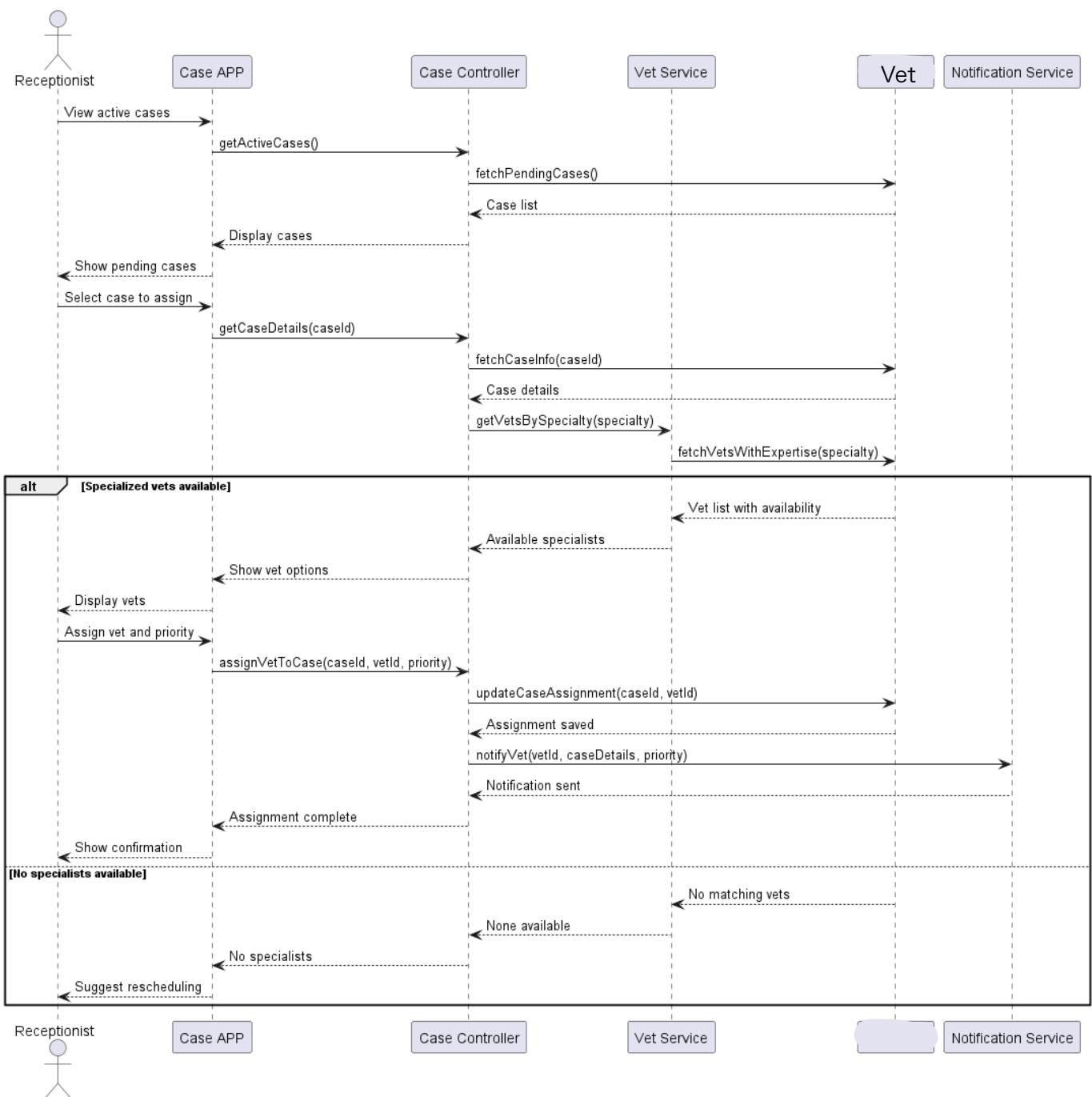
Veterinary Management System Requirements Specification

SQ_21: Cancel an Appointment



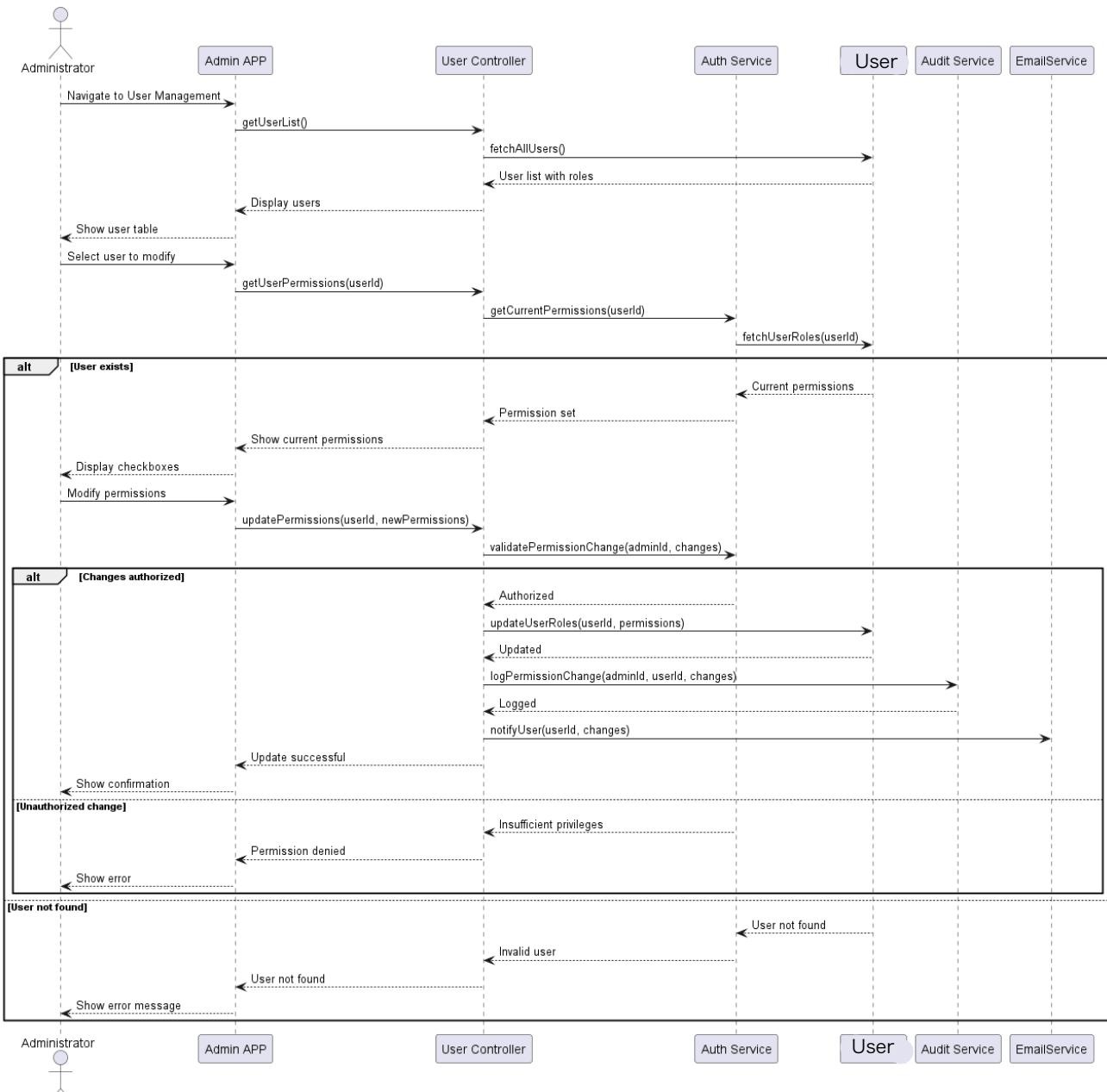
Veterinary Management System Requirements Specification

SQ_22: Assign a Veterinarian to a Case



Veterinary Management System Requirements Specification

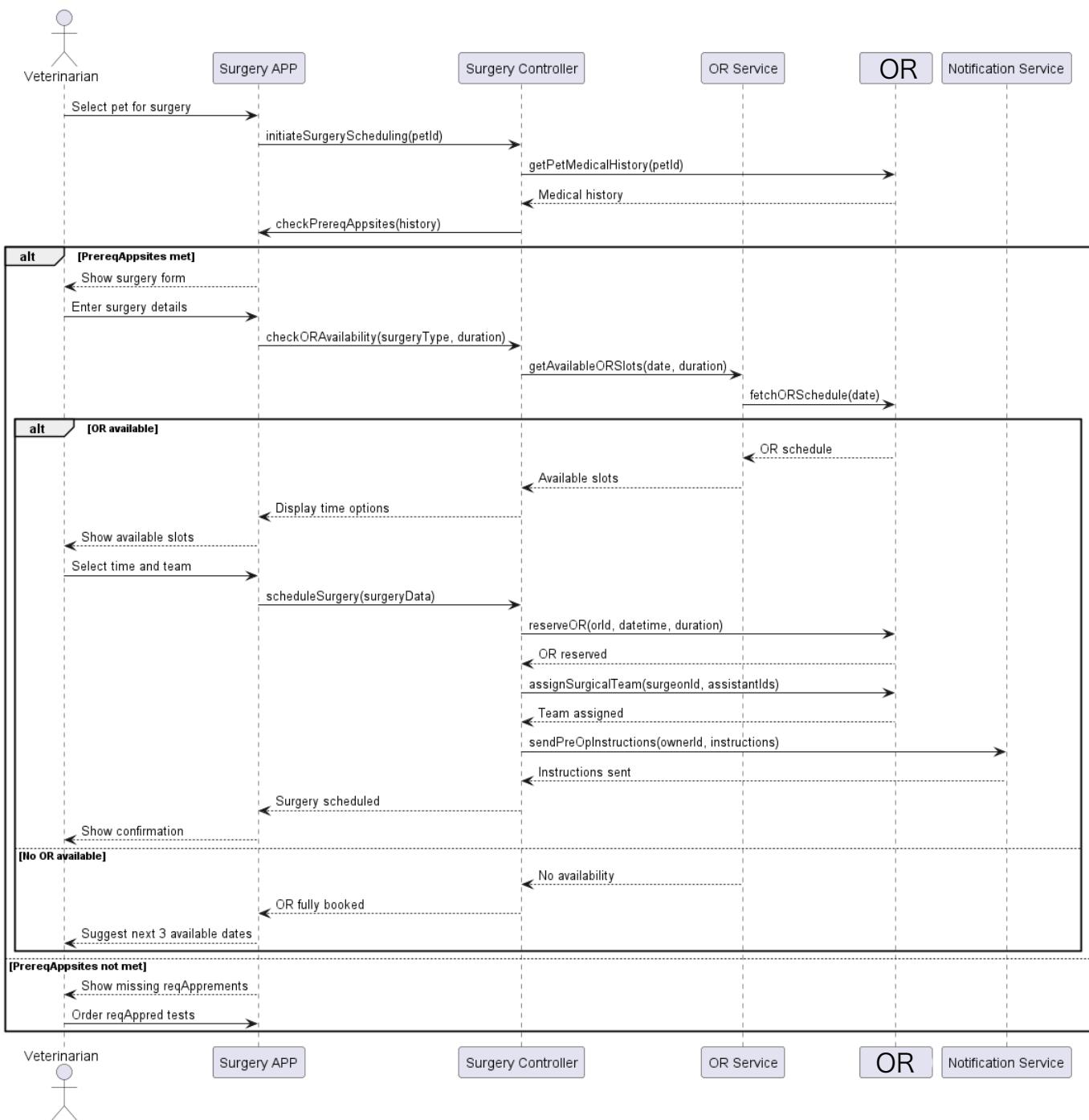
SQ_23: Manage User Roles & Permissions



23-29 Era

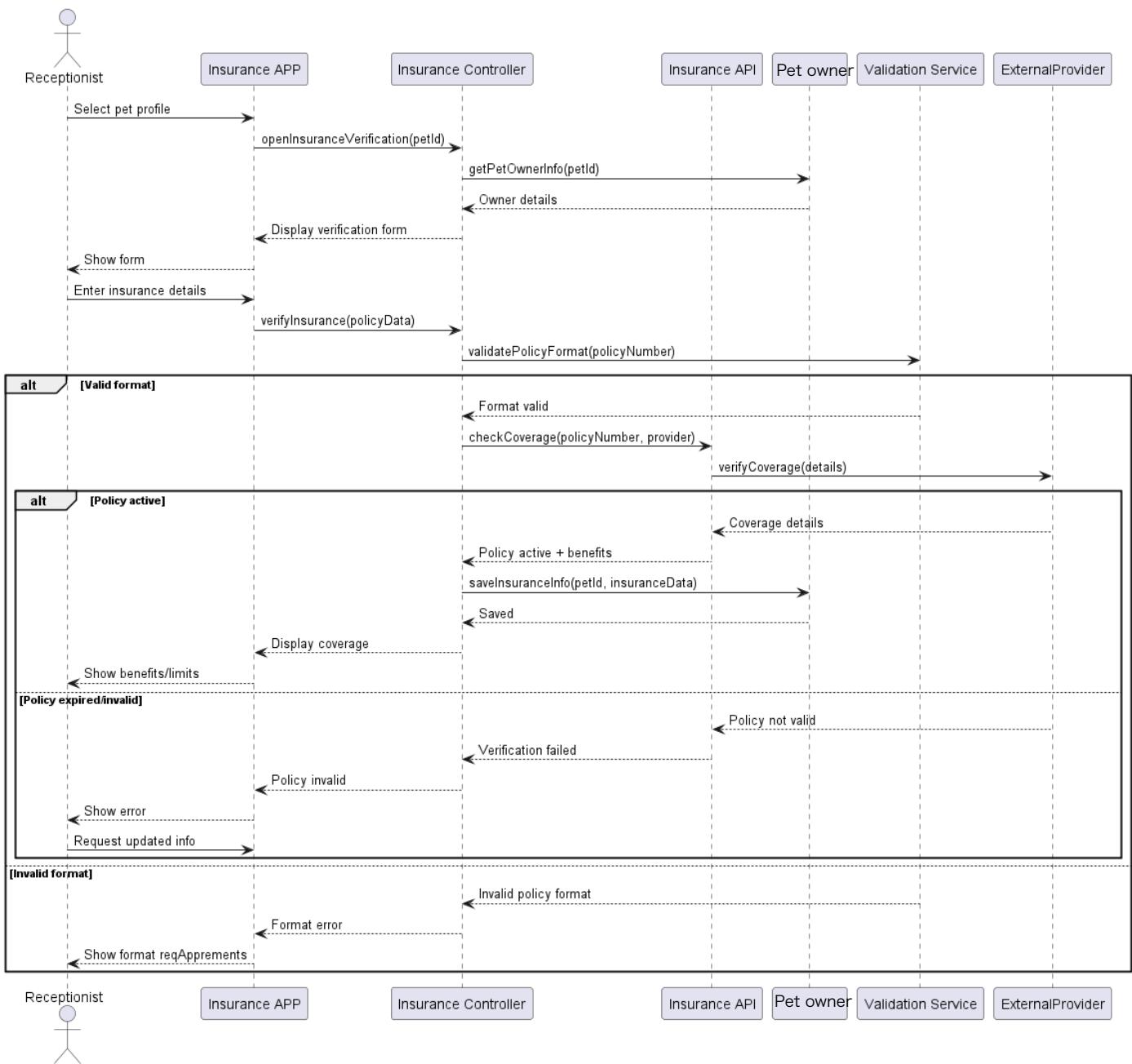
Veterinary Management System Requirements Specification

SQ_24: Schedule Surgery



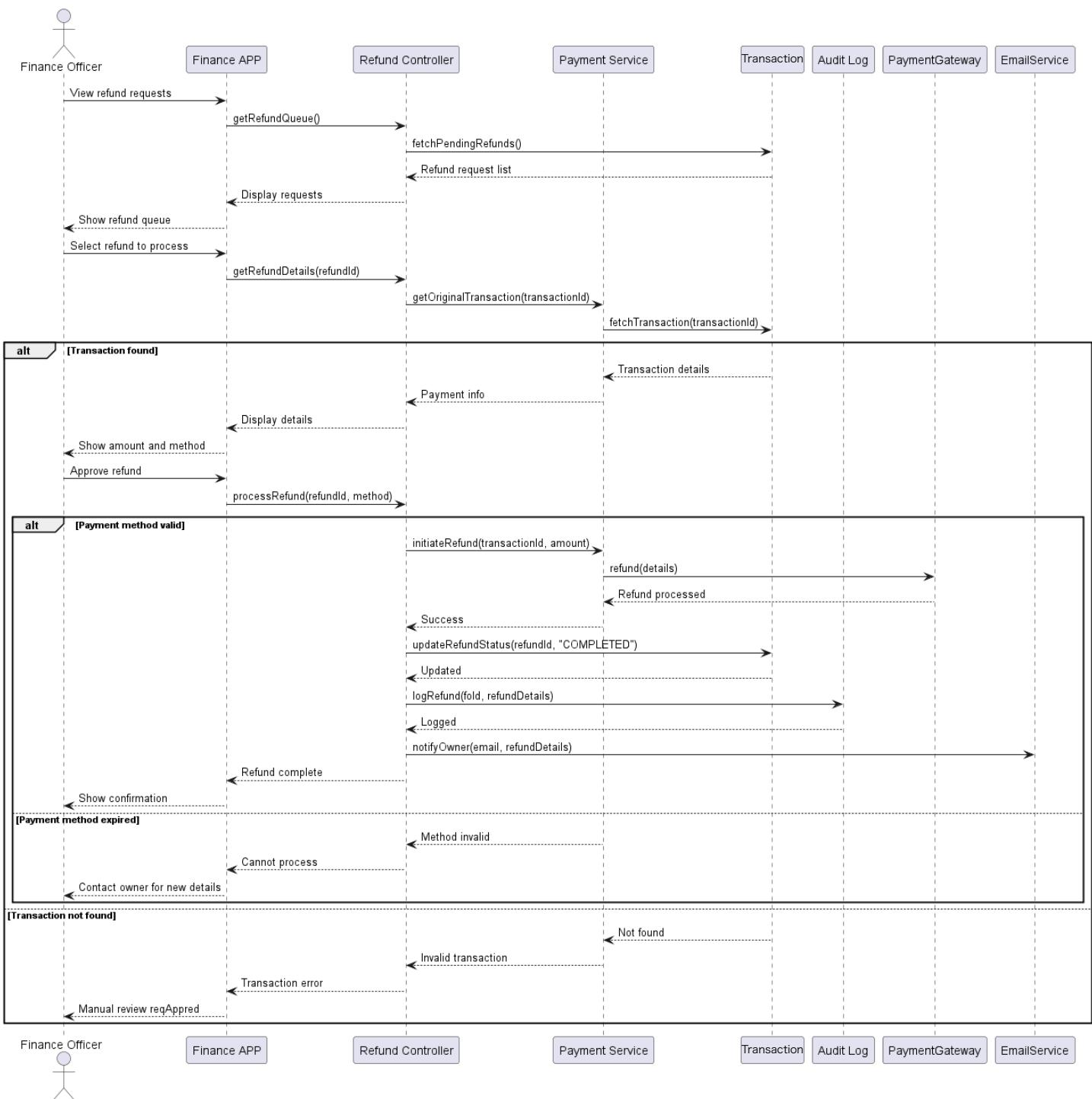
Veterinary Management System Requirements Specification

SQ_25: Verify Pet Insurance



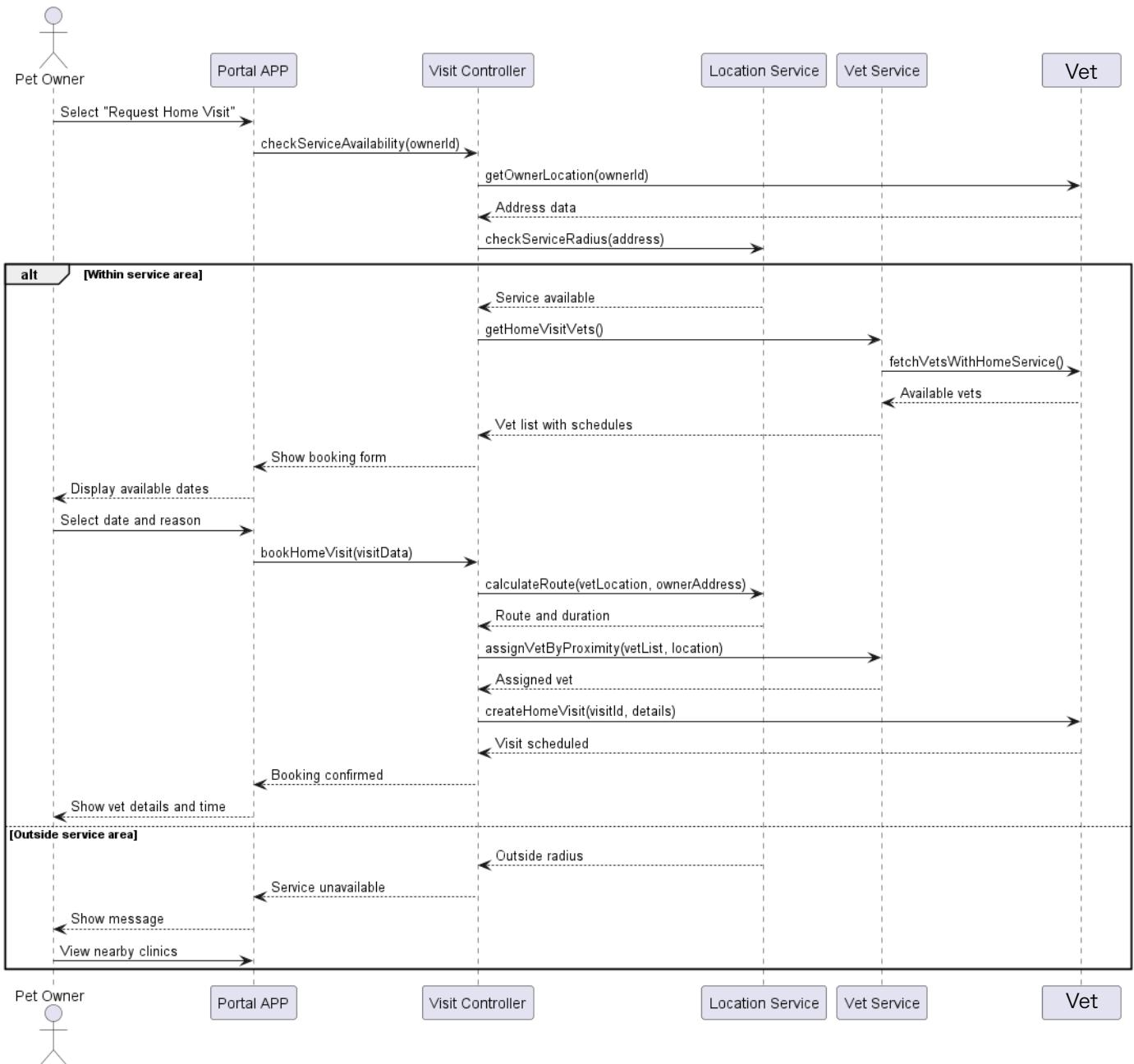
Veterinary Management System Requirements Specification

SQ_26: Process Refunds



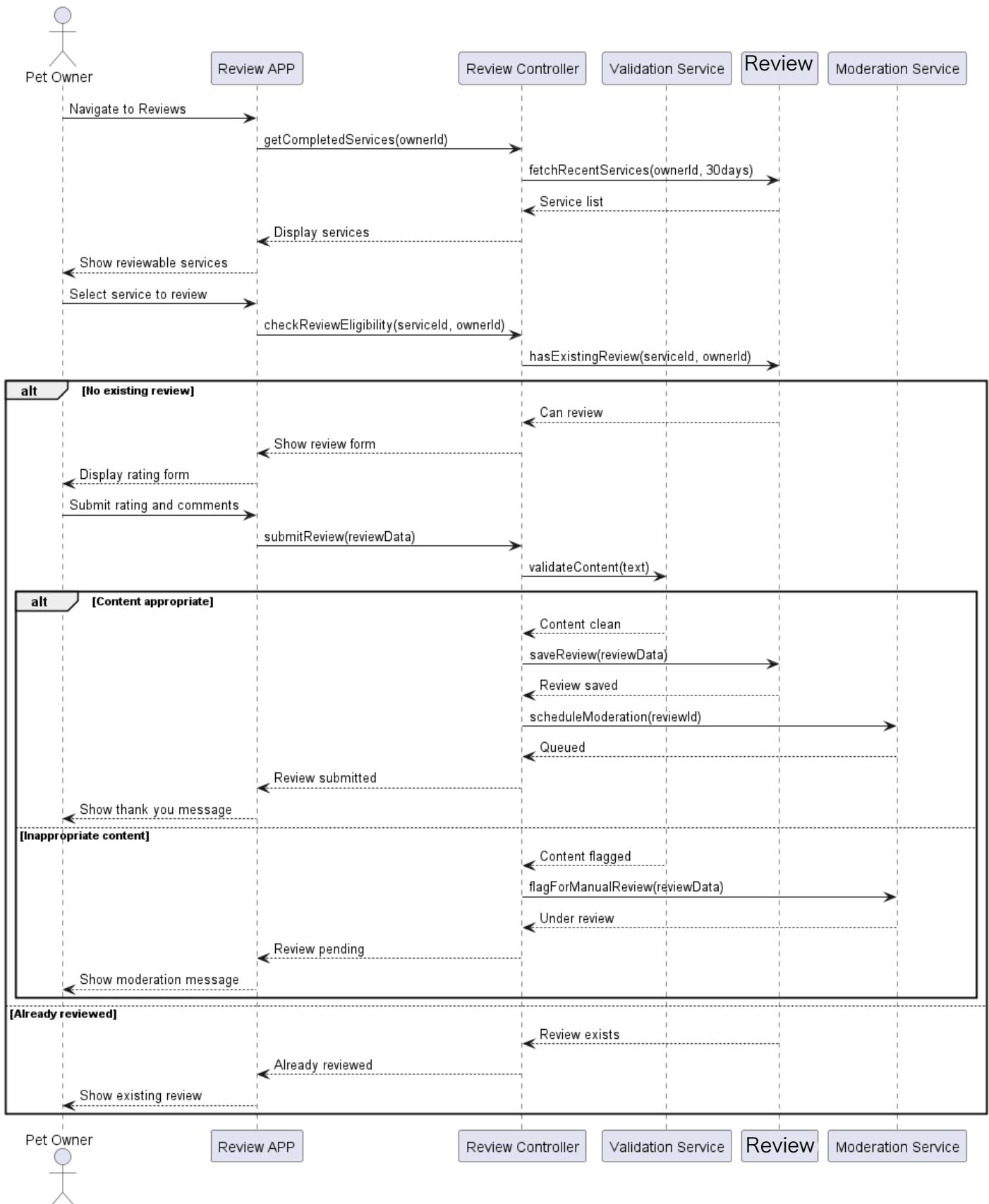
Veterinary Management System Requirements Specification

SQ_27: Request Home Visit



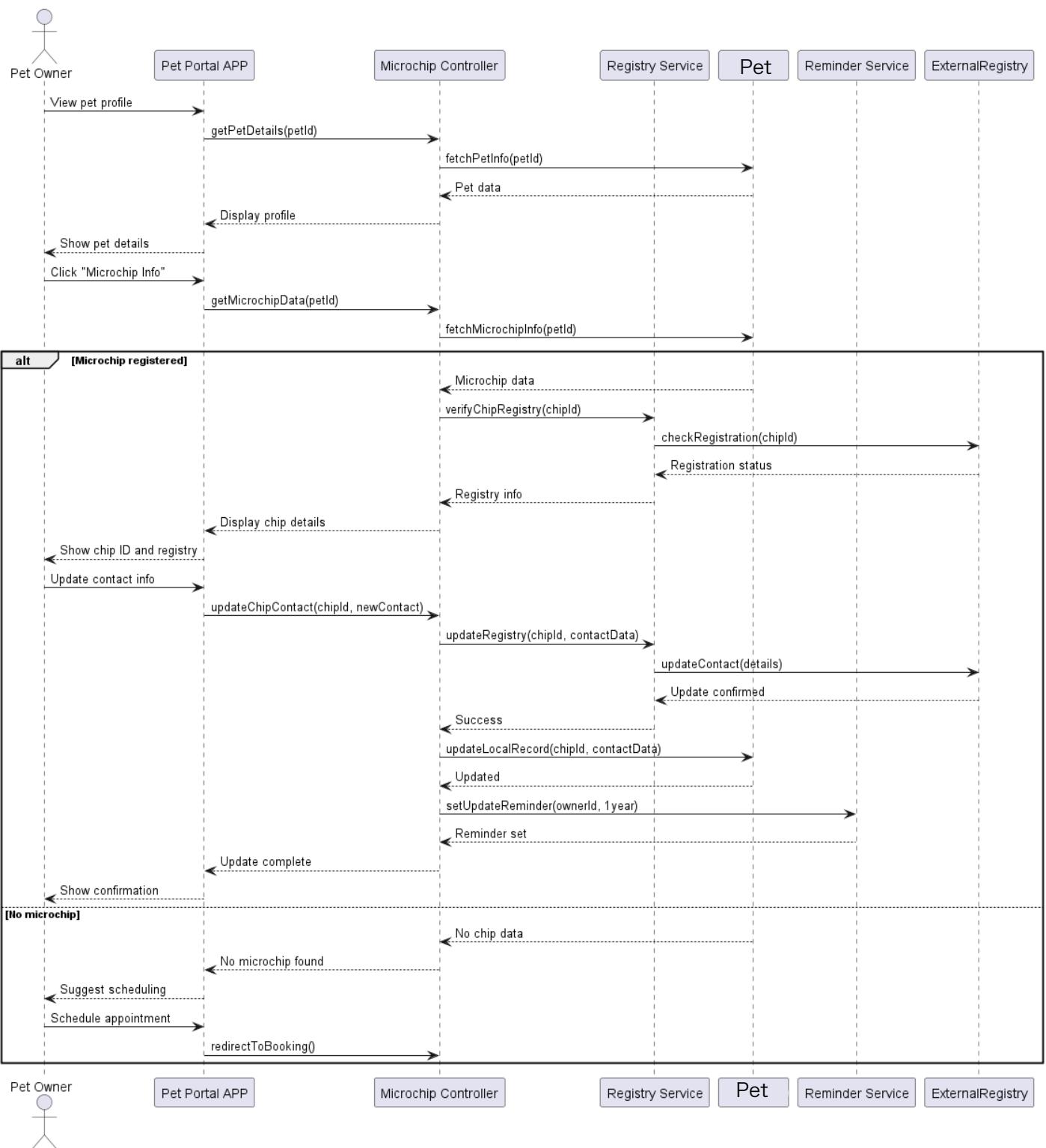
Veterinary Management System Requirements Specification

SQ_28: Rate and Review Services



Veterinary Management System Requirements Specification

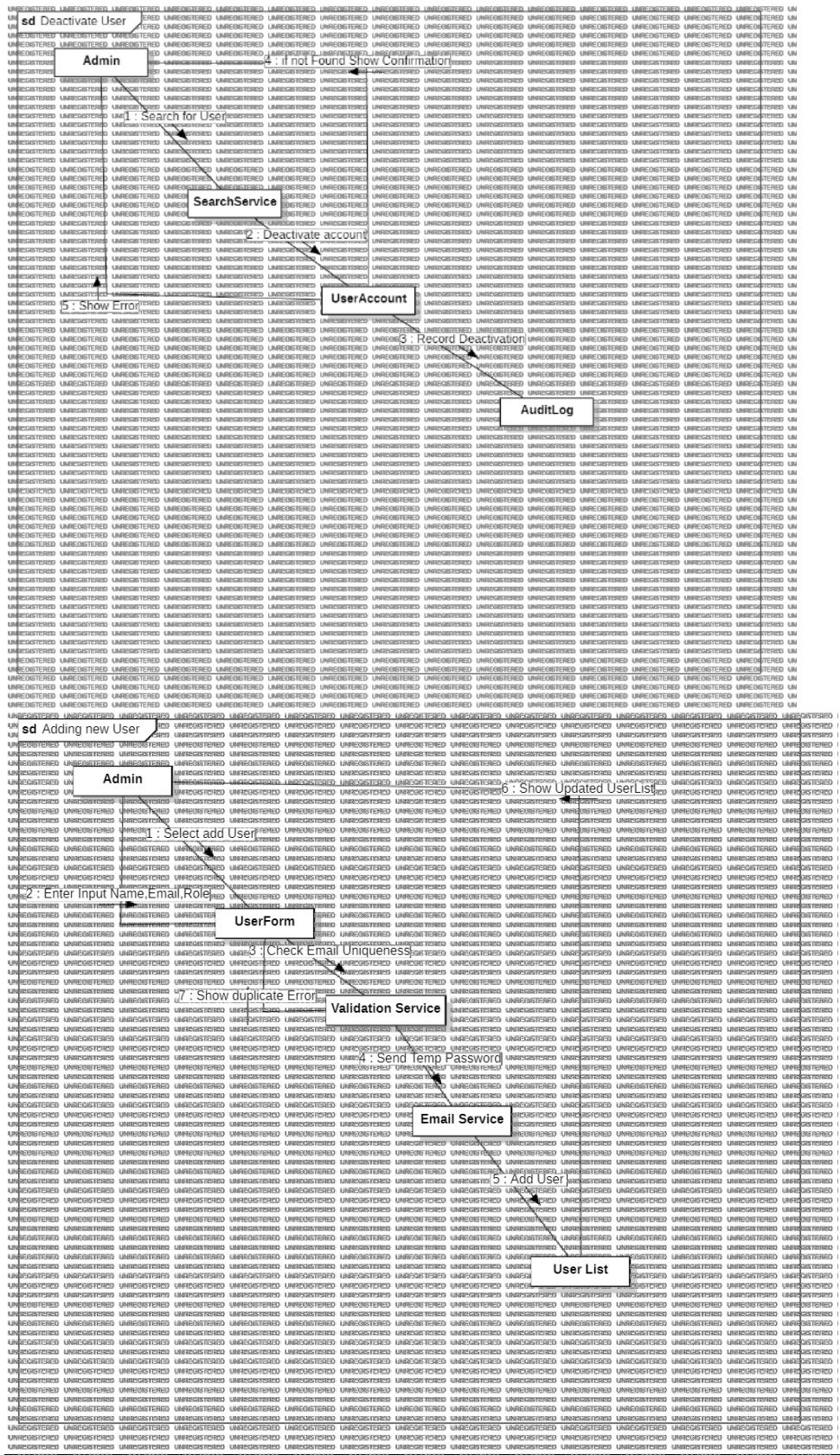
SQ_29: Track Pet Microchip Information



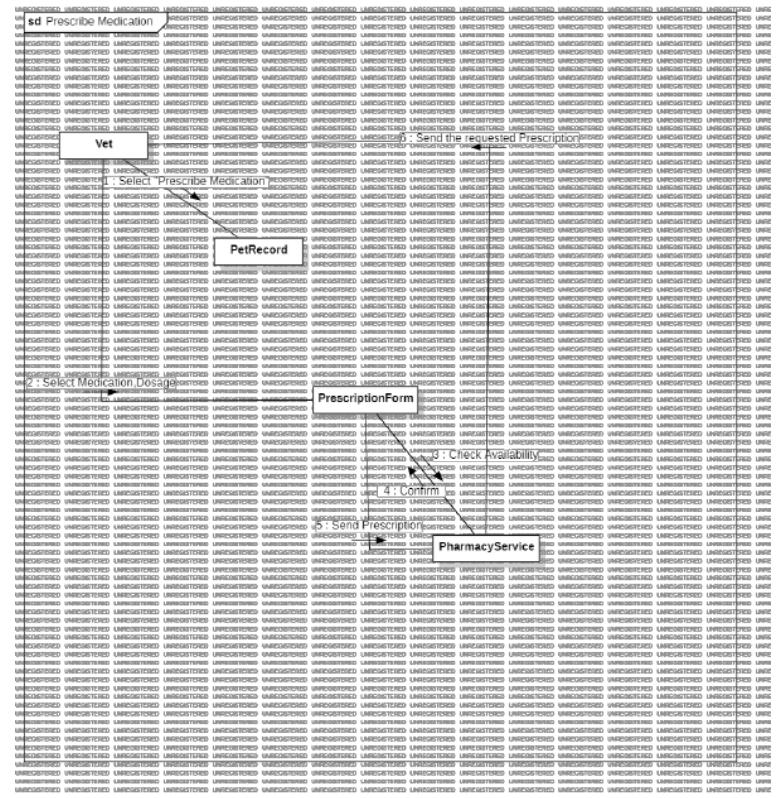
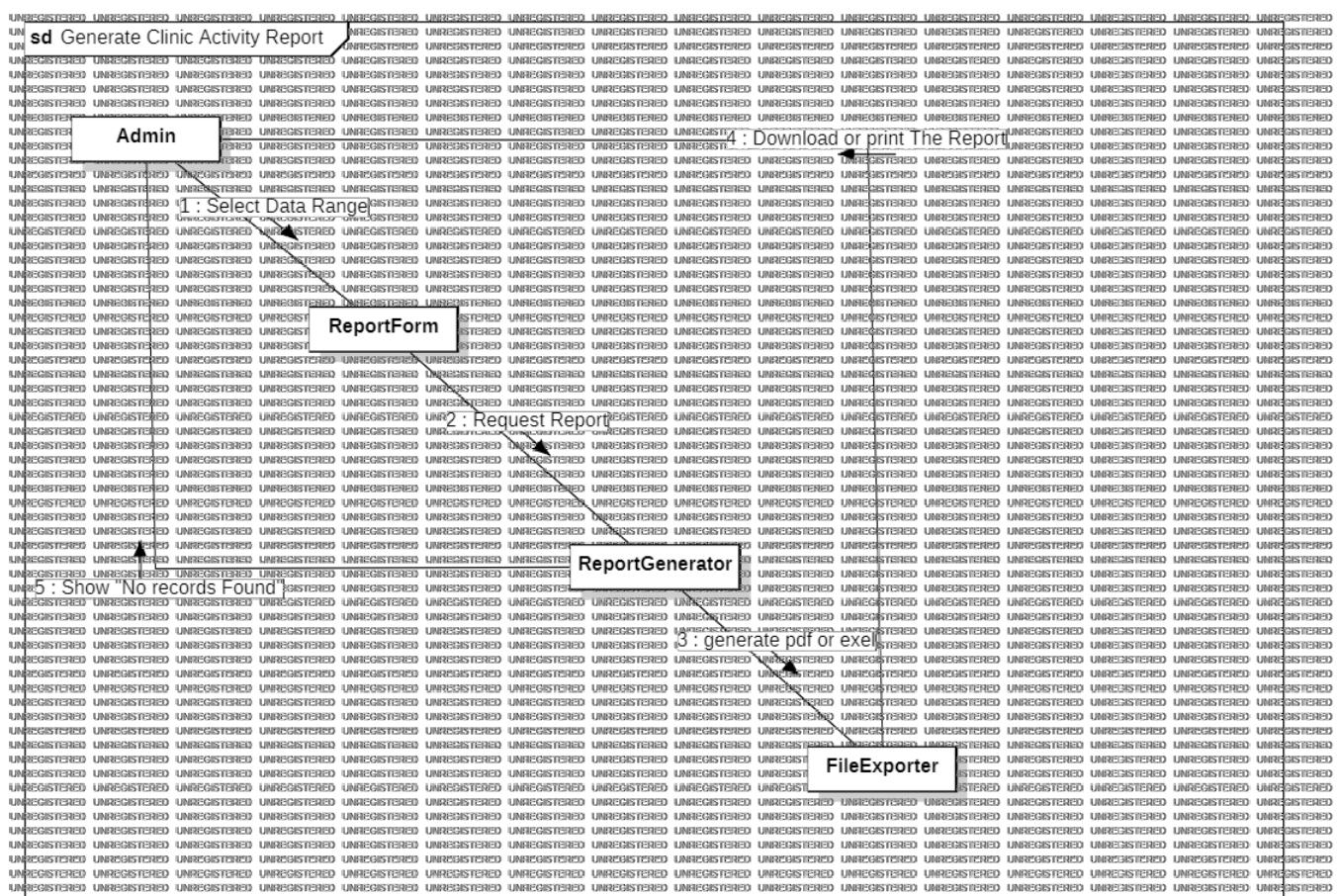
Veterinary Management System Requirements Specification

5.7. Collaboration diagram

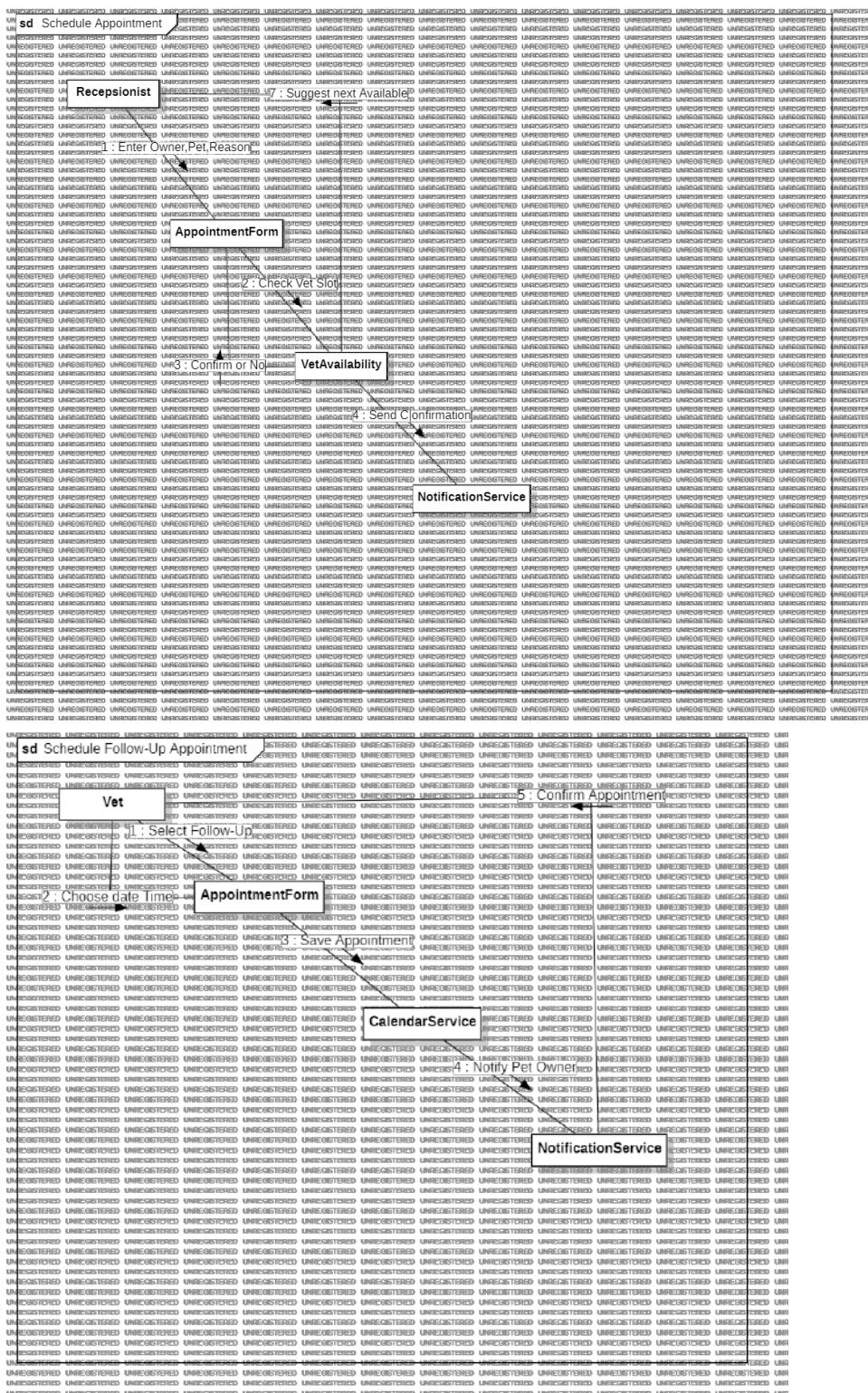
All collaboration diagrams directly relate to a sequence diagram. If a sequence diagram is named SC_07, then the collaboration diagram is named CC_07. Eridjon 01-04



Veterinary Management System Requirements Specification

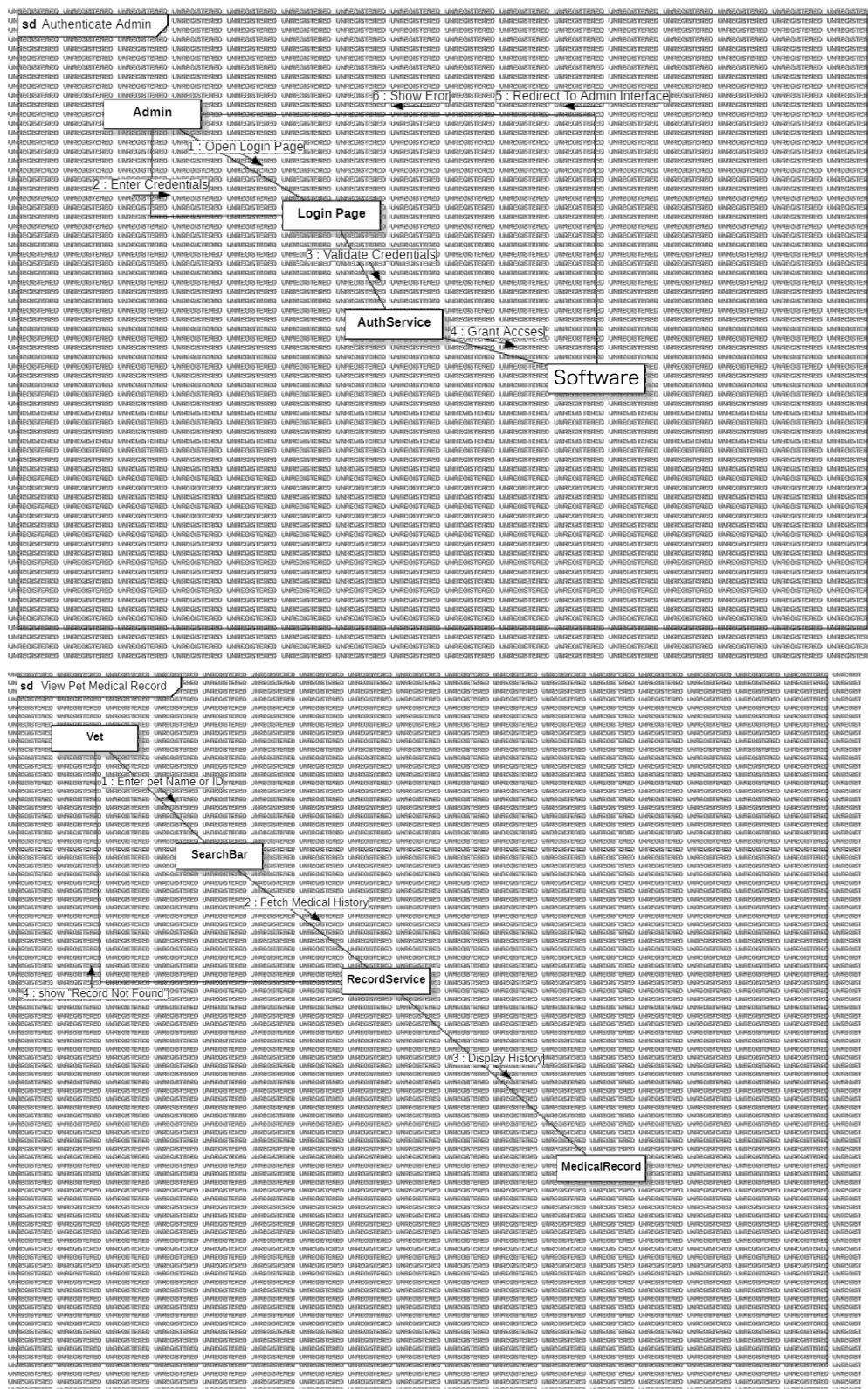


Veterinary Management System Requirements Specification

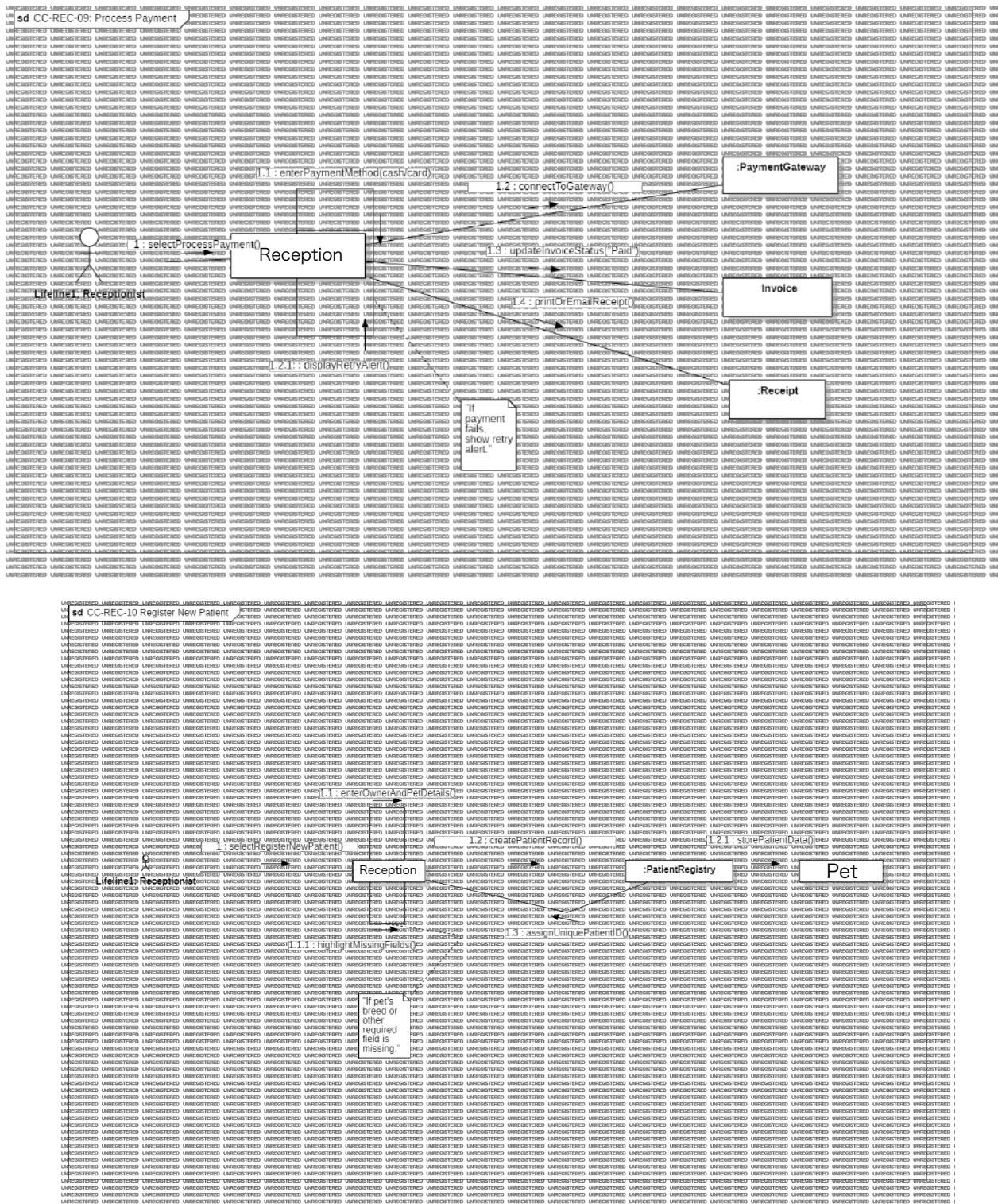


Enea 05-09

Veterinary Management System Requirements Specification

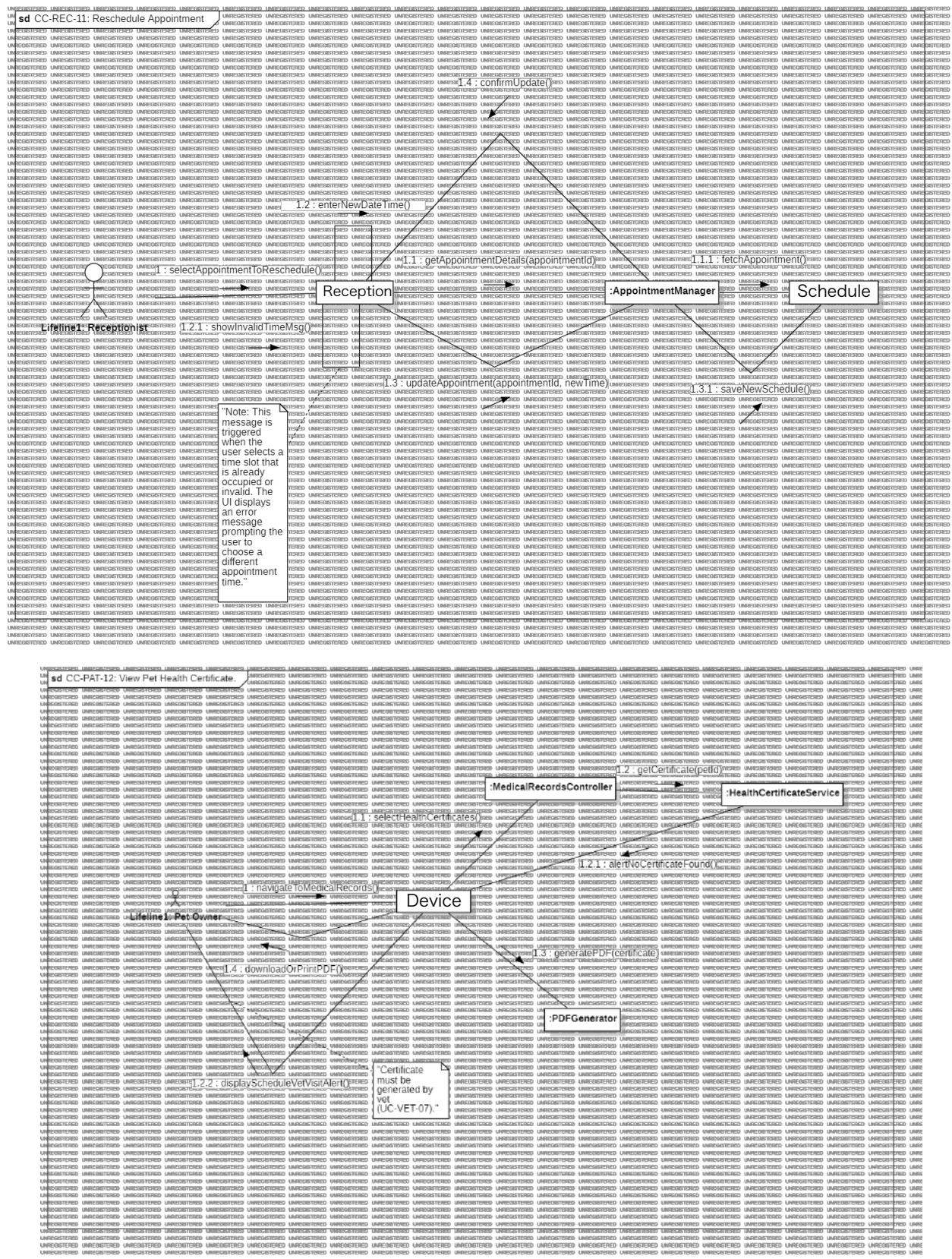


Veterinary Management System Requirements Specification

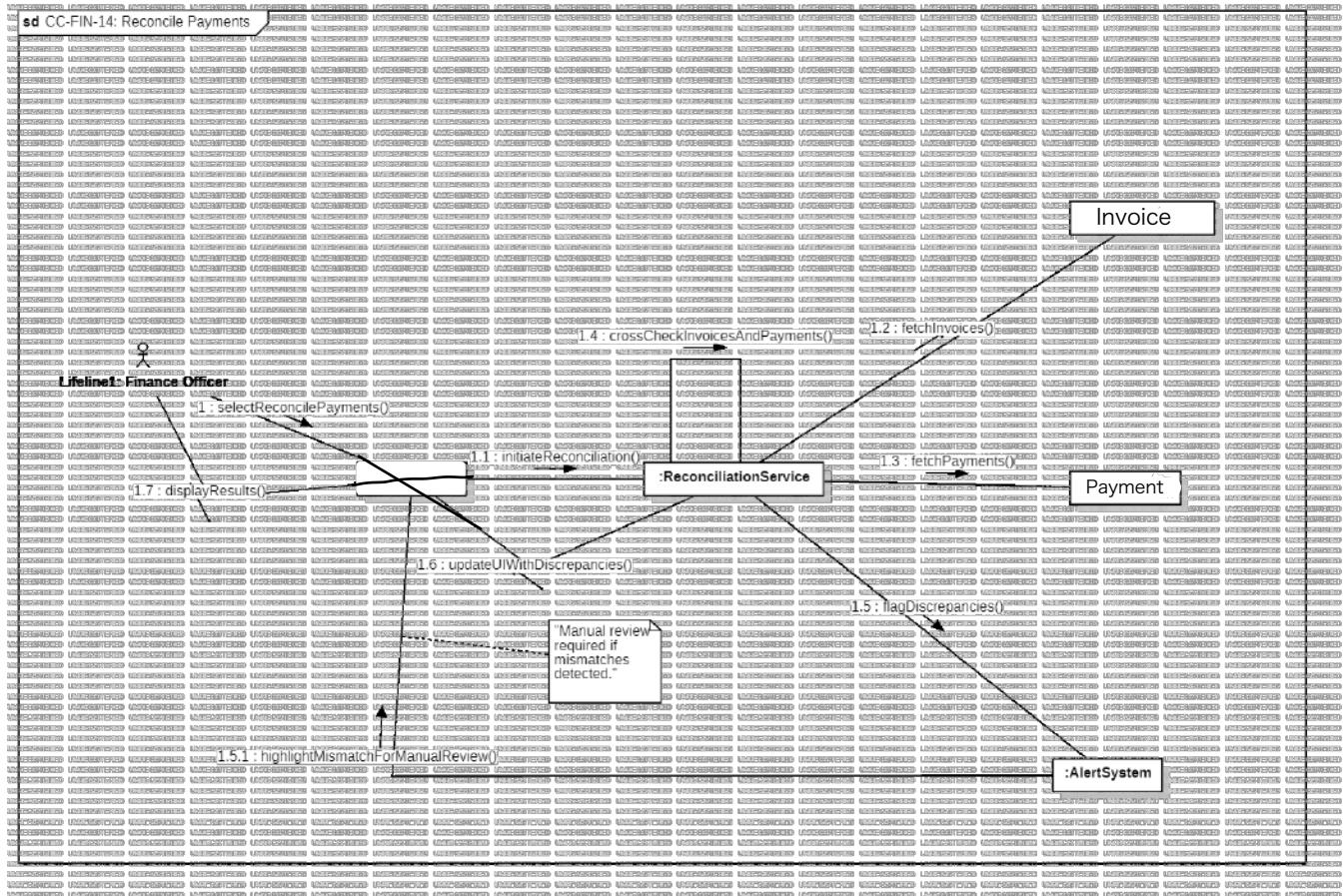
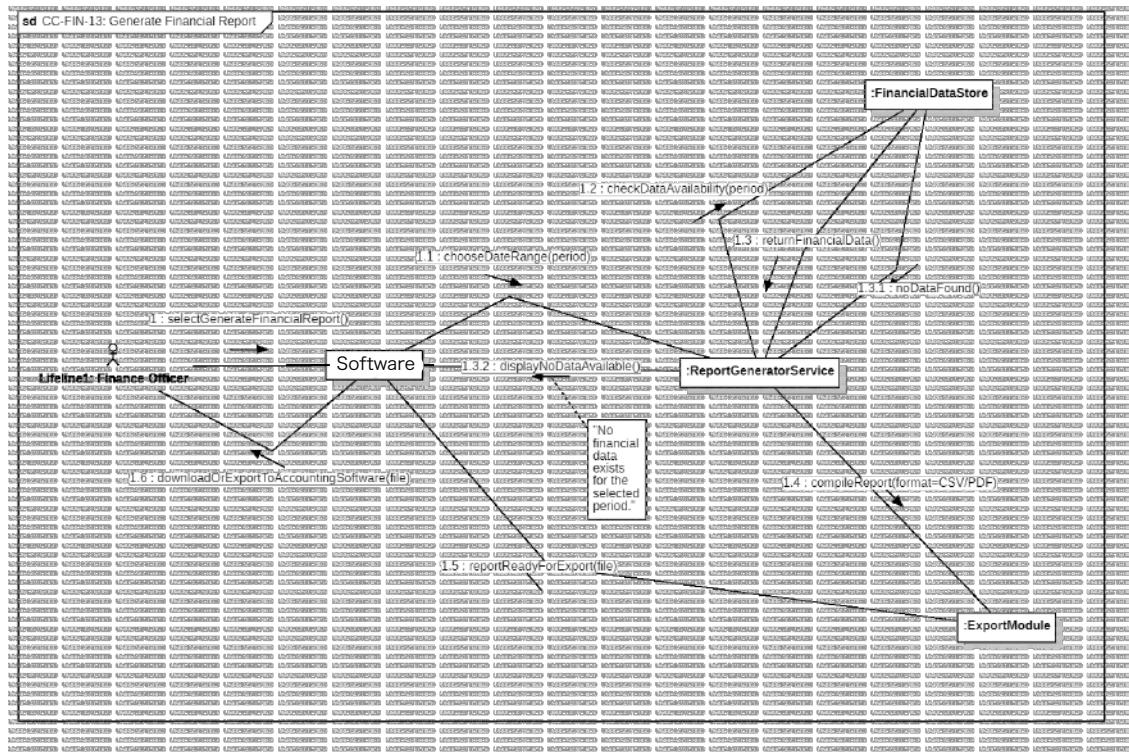


Laert 10-13

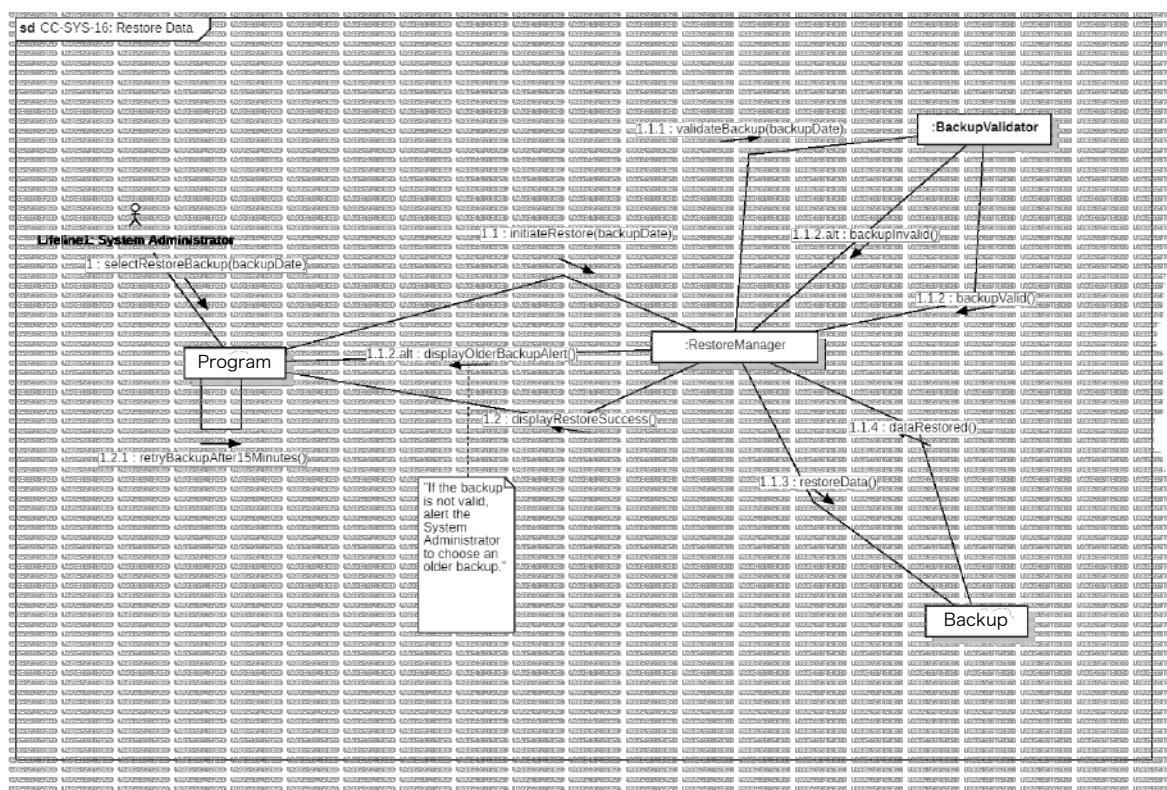
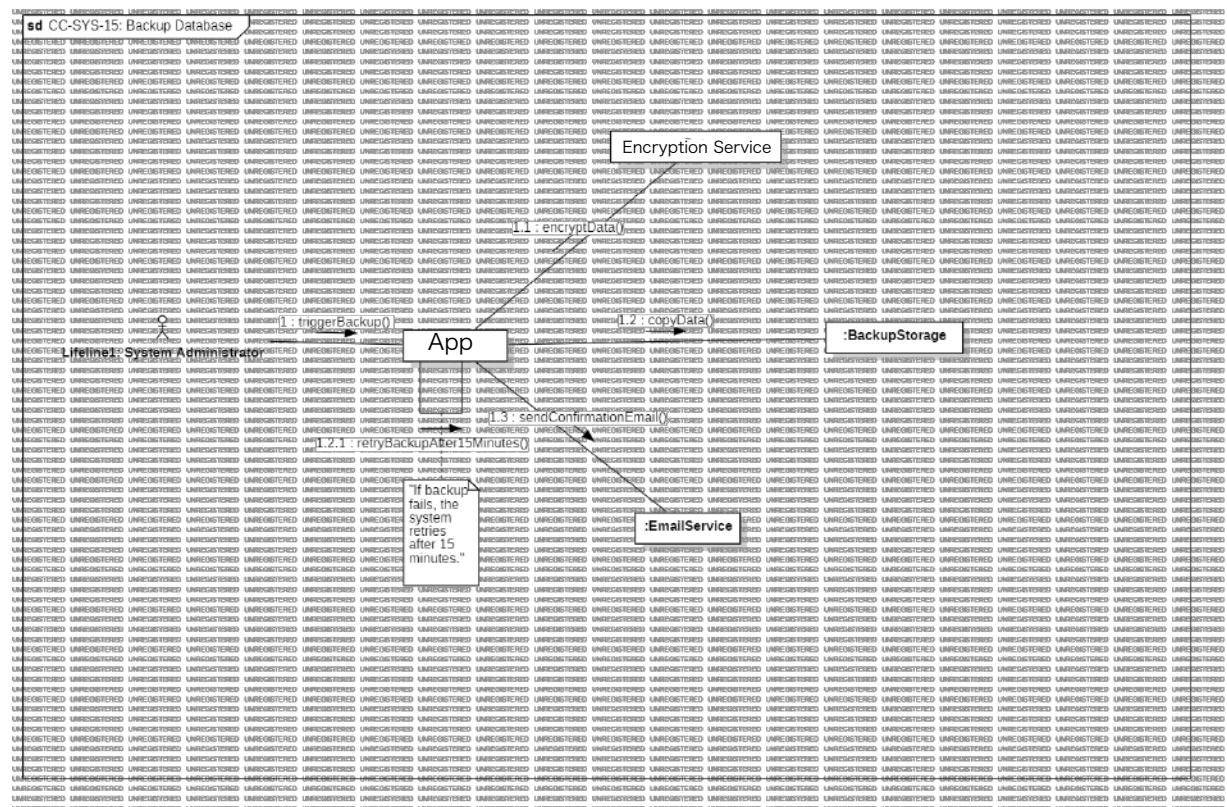
Veterinary Management System Requirements Specification



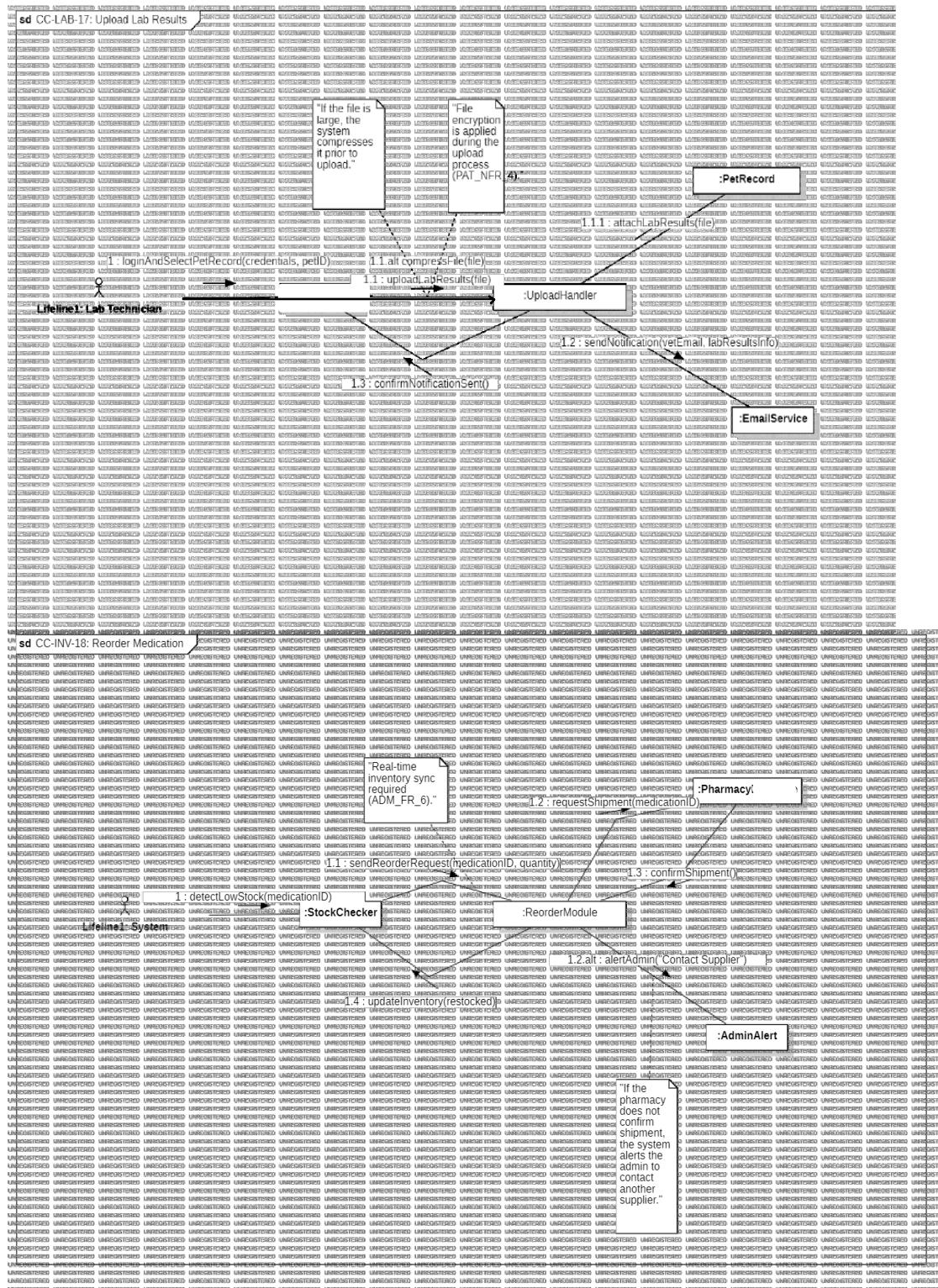
Veterinary Management System Requirements Specification



Veterinary Management System Requirements Specification

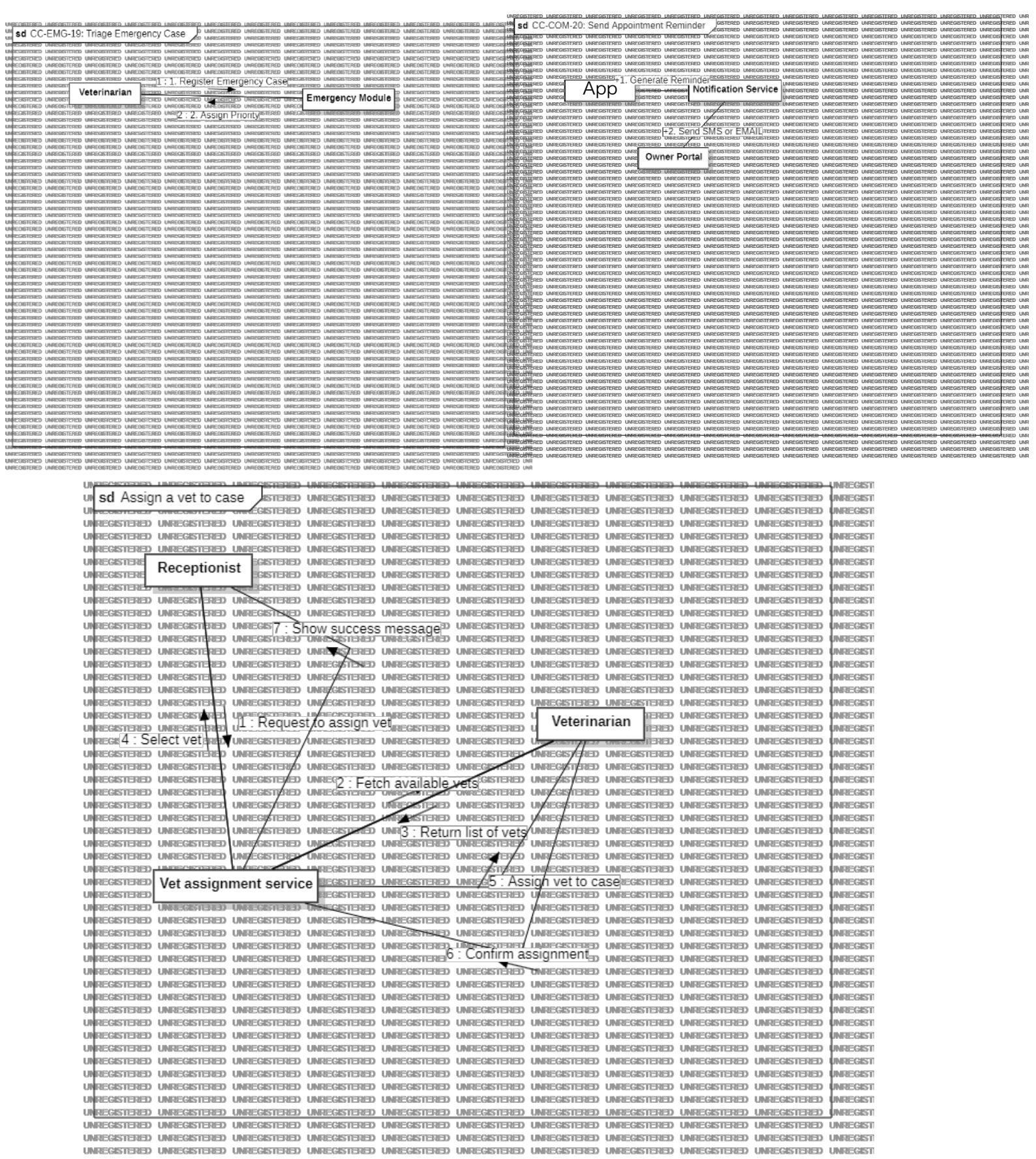


Veterinary Management System Requirements Specification

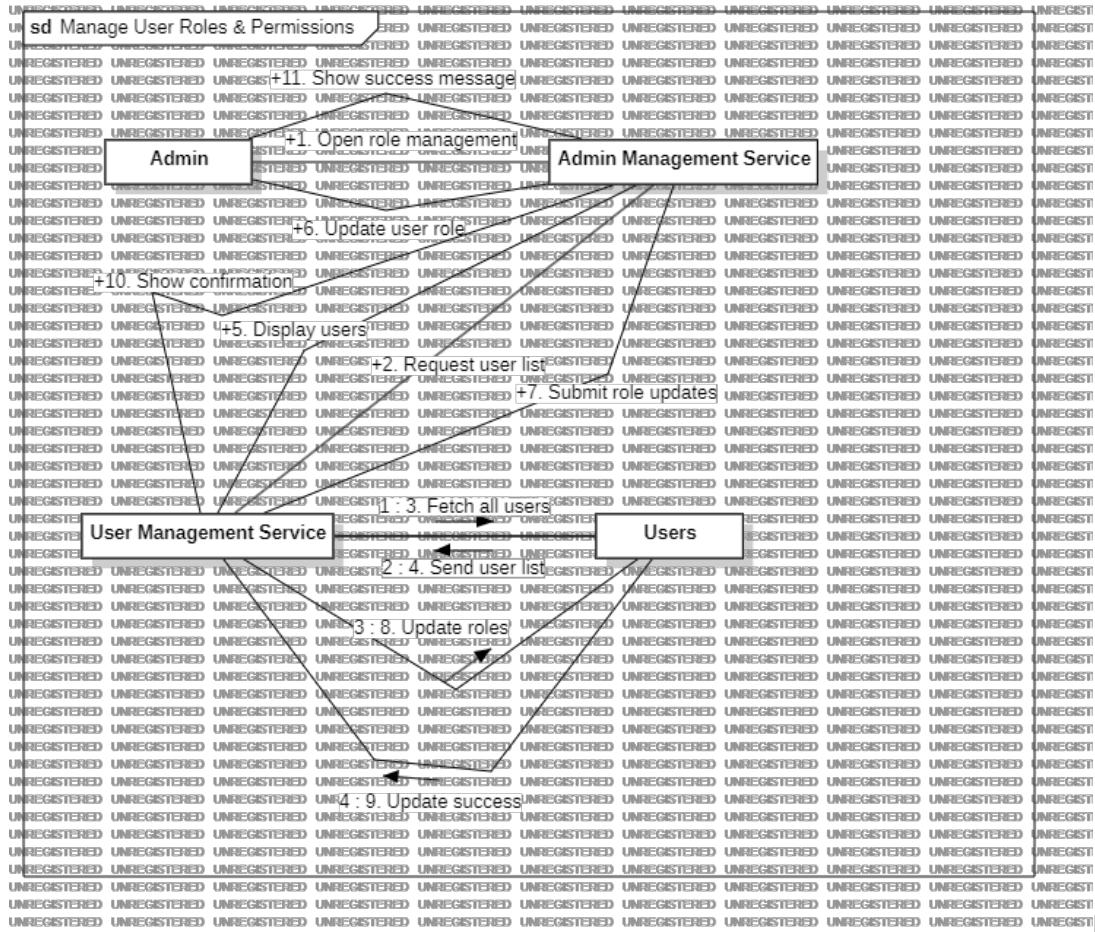


Emi 18-21

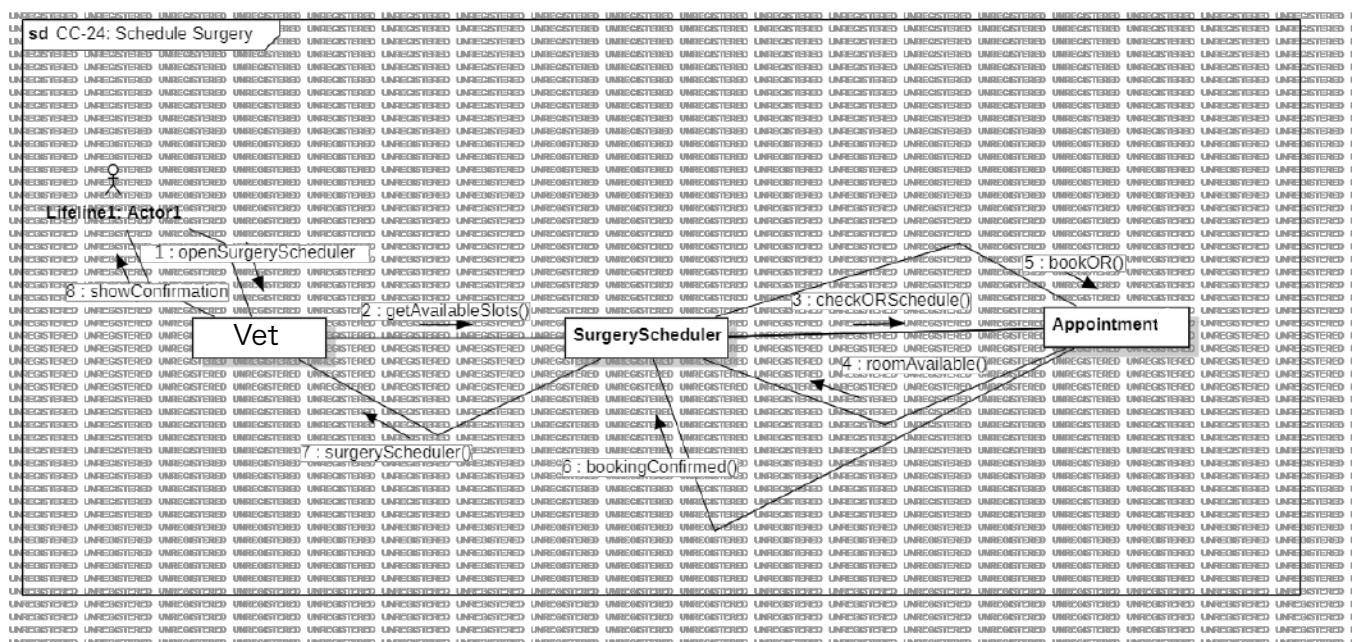
Veterinary Management System Requirements Specification



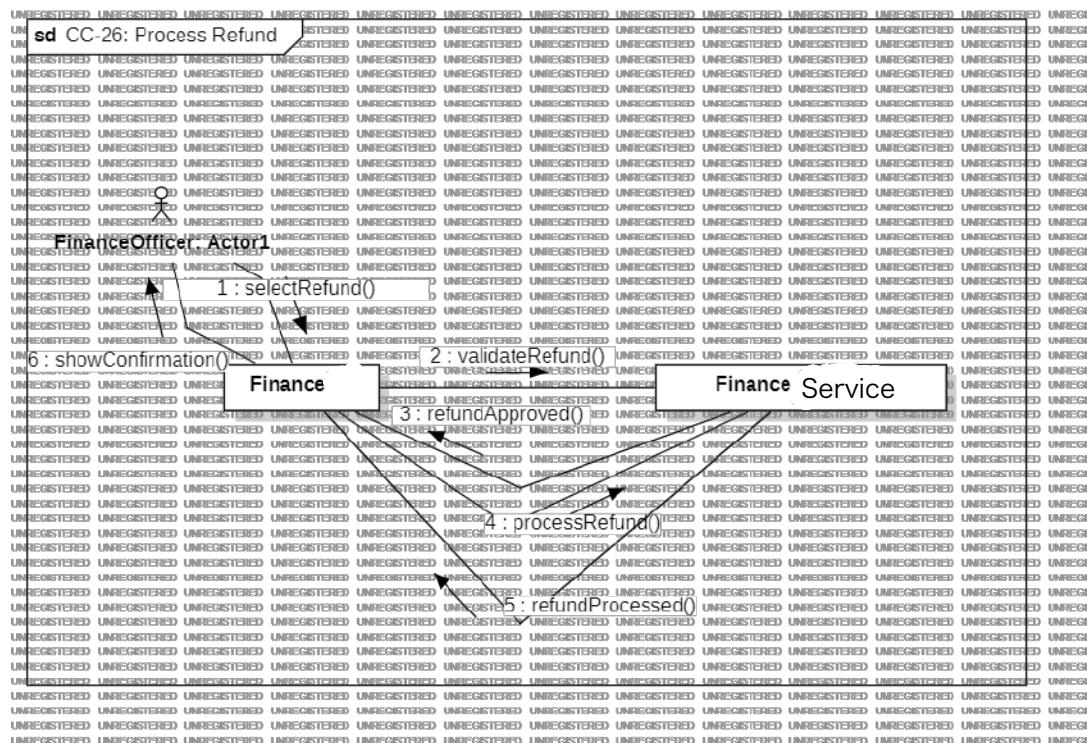
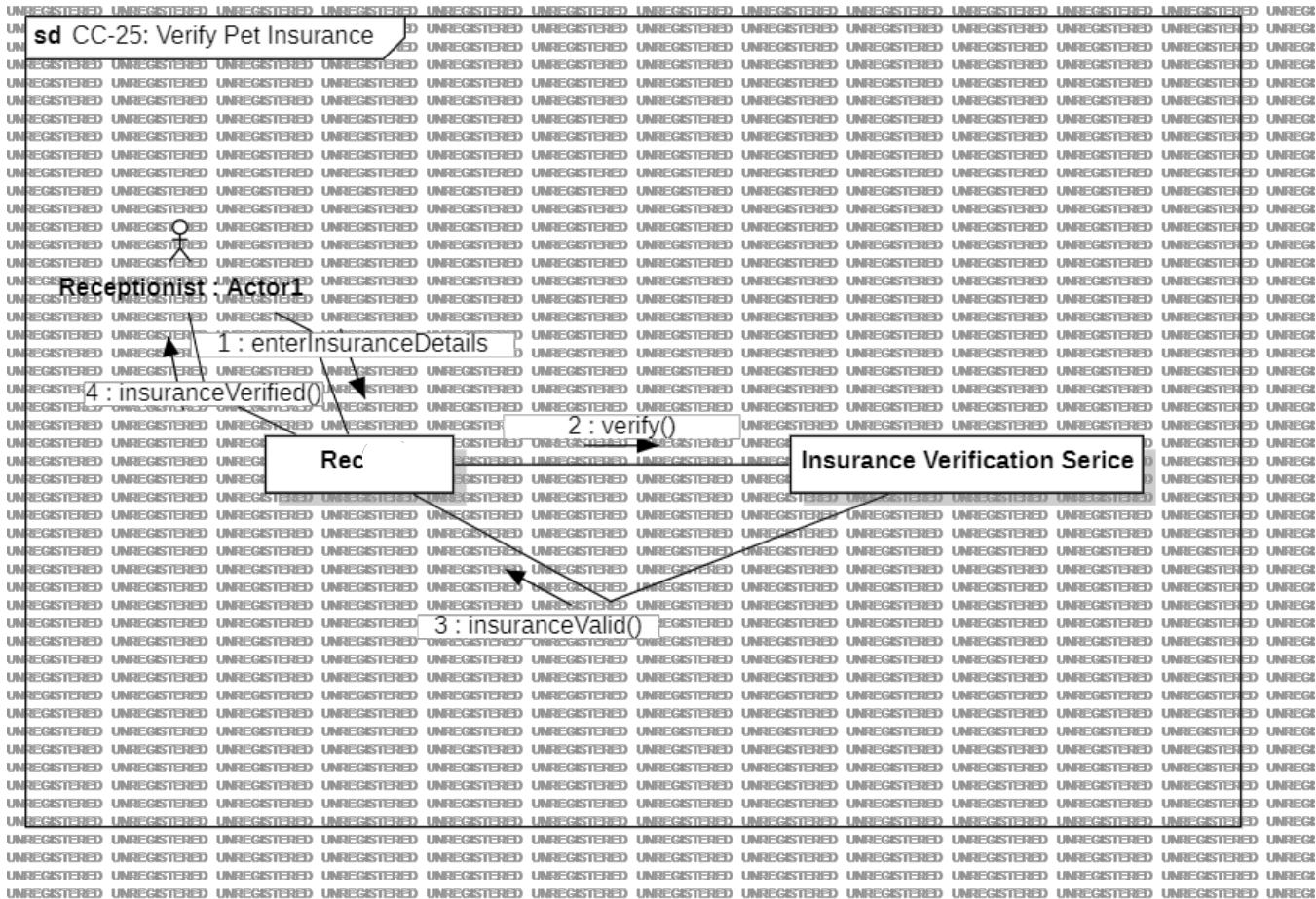
Veterinary Management System Requirements Specification



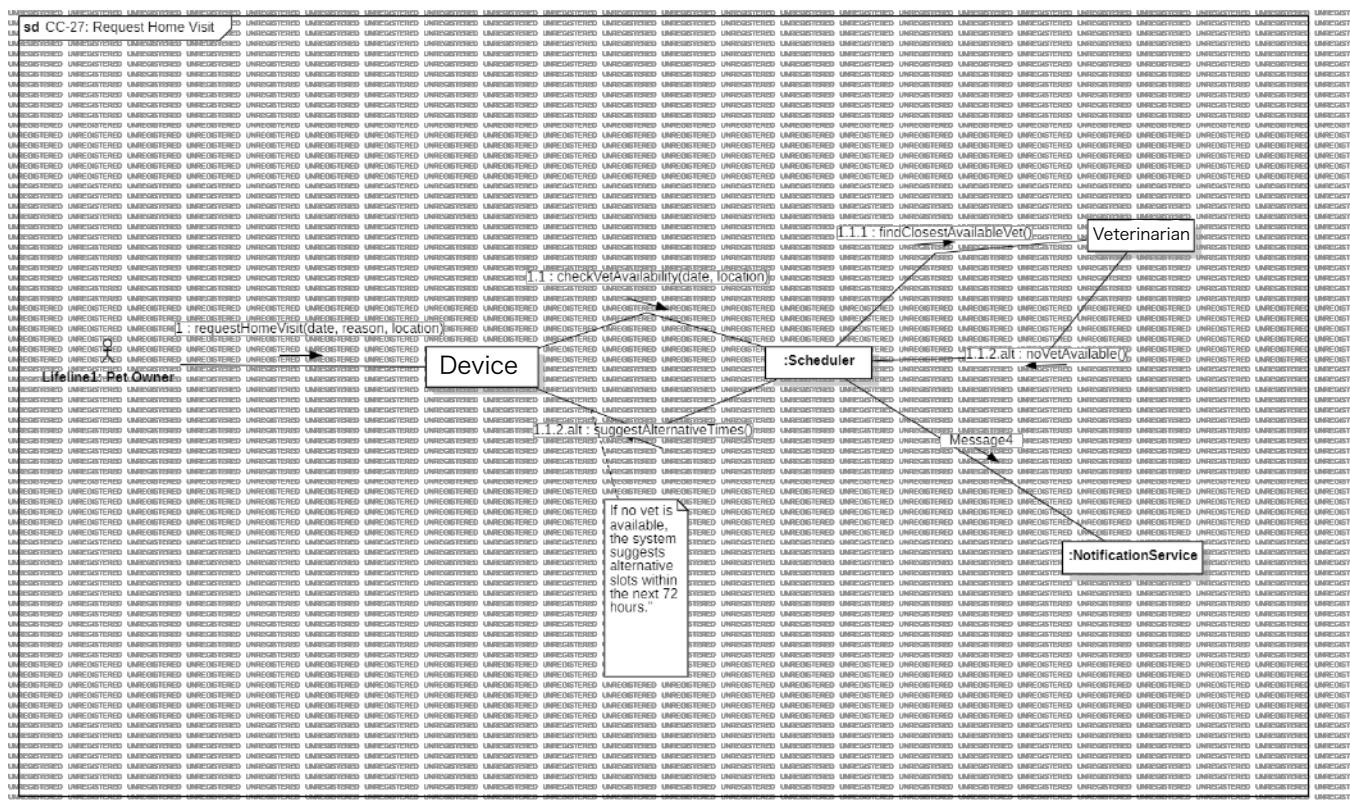
Era 22-29



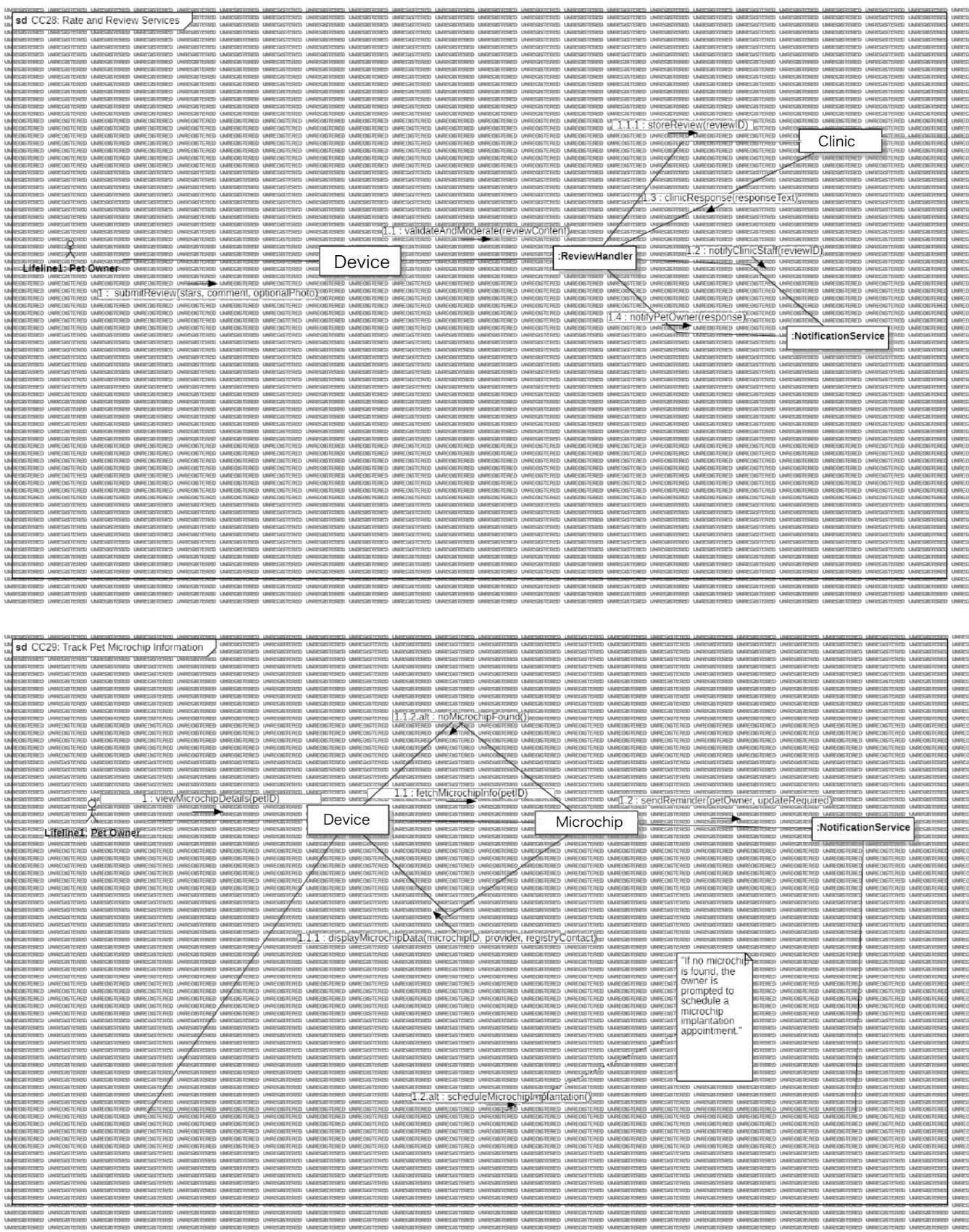
Veterinary Management System Requirements Specification



Veterinary Management System Requirements Specification



Veterinary Management System Requirements Specification



6. Design Patterns

Choose the relevant design patterns for your project. For each, give a reasoning and the associated class and sequence diagram. These are NOT part of the above diagrams, and need not carry the following naming scheme.

(Emi)The Singleton Pattern ensures that a class has only one instance throughout the application and

provides a global point of access to that instance. Singleton is a creational design pattern that lets you ensure that a class has only one instance, while providing a global access point to this instance..

AppointmentManager is a perfect candidate for Singleton because:

1. Central Coordination:

It manages all appointments and reminders in the system. Having multiple managers could lead to inconsistencies (duplicate or conflicting appointments).

2. Shared State:

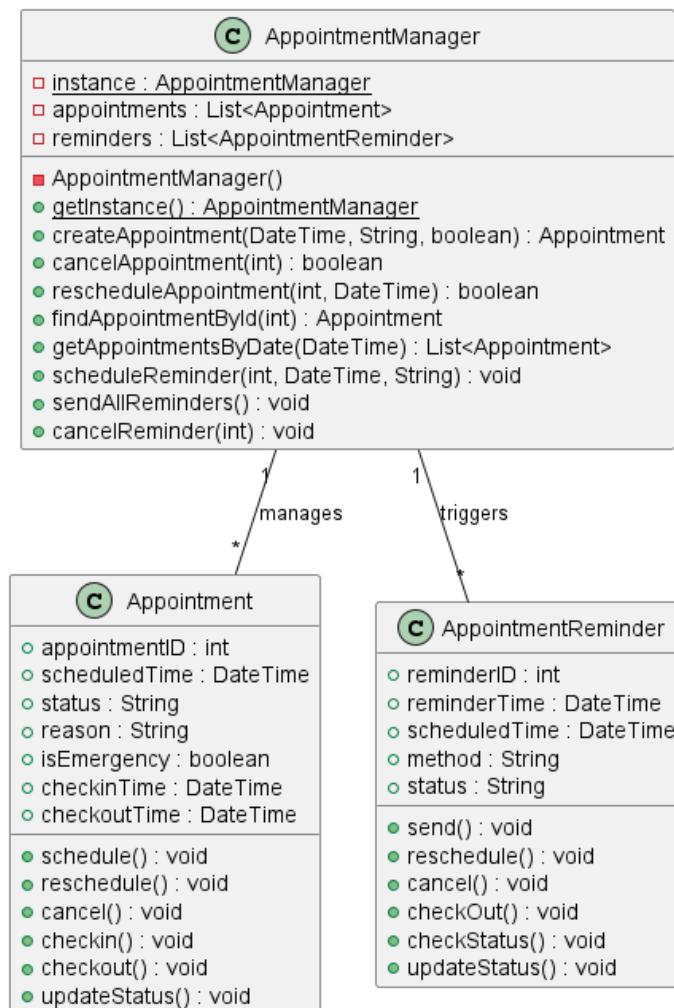
All parts of the system (Admin, Receptionist, Doctor) should work with the same set of appointments. The Singleton pattern guarantees a shared, consistent state.

3. Controlled Access:

By centralizing appointment creation, rescheduling, and cancellation, you gain better control over appointment logic (like conflict checking or notification triggers).

4. Avoids Duplication:

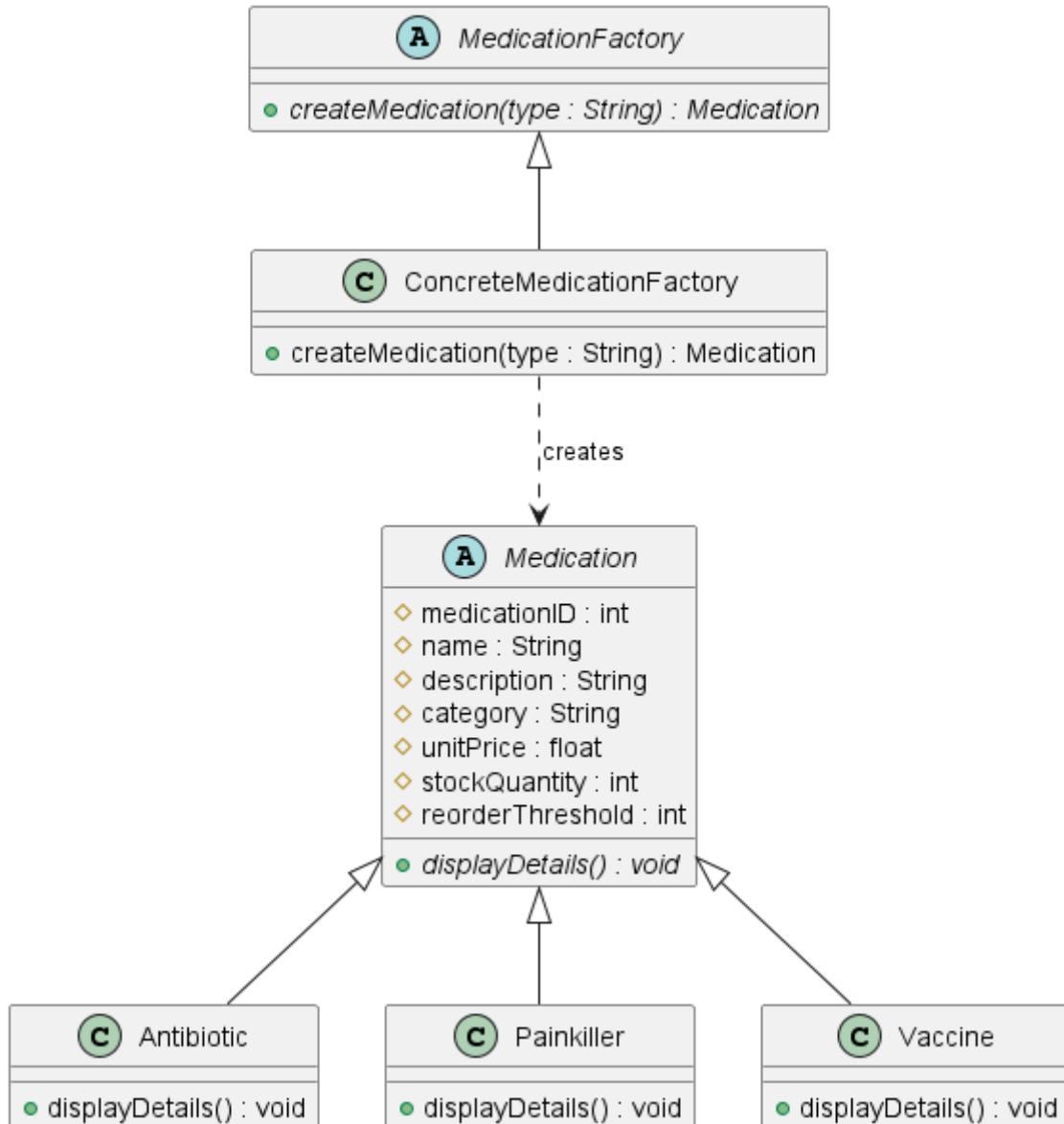
Without Singleton, different components might create their own AppointmentManager objects and store separate appointment lists — causing data fragmentation.



Veterinary Management System Requirements Specification

(Era) **Command** is a behavioral design pattern that turns a request into a stand-alone object that contains all information about the request. This transformation lets you pass requests as a method arguments, delay or queue a request's execution, and support undoable operations.

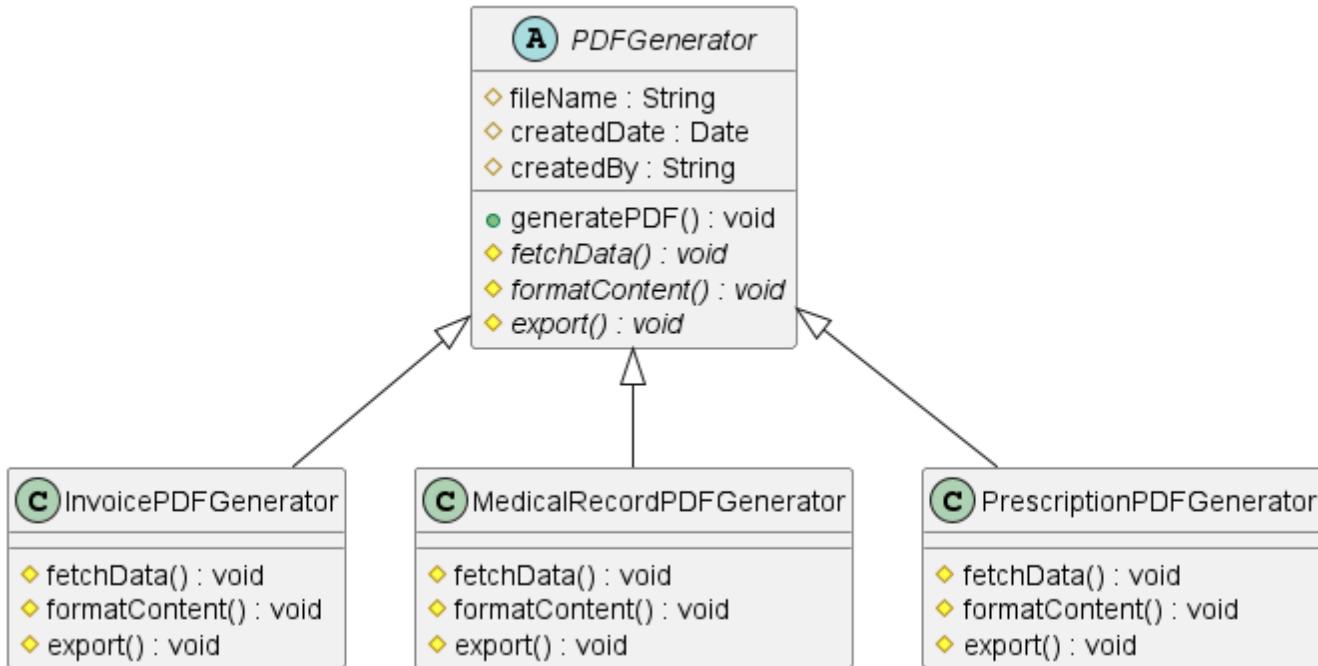
Command Pattern for Prescription Management in VMS The Command Pattern in VMS encapsulates prescription actions—such as prescribing, refilling, and canceling—as standalone command objects that implement execute(), undo(), and validate() methods. These are processed by a central Invoker, which manages execution, queuing, validation, retries, rollback, and logging, while core business logic resides in Receivers. This decouples the UI from domain logic, supports asynchronous and delayed execution, and ensures transactional integrity, making the system more modular, auditable, and maintainable. Components like SecurityManager, CommandLogger, and TransactionManager enforce access control, maintain audit trails, and ensure safe rollback. Built for resilience, the system uses thread-safe queues, command status tracking, and retry mechanisms to handle failures gracefully. Commands are persisted for recovery, validated for integrity, and sanitized for security. Business rules such as dosage limits, inventory checks, and prescription validation are enforced before execution. The architecture is extensible, testable, and integrates seamlessly with external systems like pharmacies and insurers, while supporting CI/CD, observability, and fault tolerance—making it a robust fit for enterprise healthcare workflows.



Veterinary Management System Requirements Specification

(Enea) The Template Method. Template Method is a behavioral design pattern that defines the skeleton of an algorithm in the superclass but lets subclasses override specific steps of the algorithm without changing its structure.

Why Template Method in VMS? In VMS, report generation (for medical records, inventory, finance, etc.) shares a common process: initialization → data retrieval → formatting → export. The Template Method allows us to define the general workflow in a BaseReport class, while letting subclasses like MedicalReport, InvoiceReport, and InventoryReport implement the specific steps. The base class BaseReport defines the structure of the reporting process with hooks and abstract methods. Each subclass customizes only what's relevant.



Veterinary Management System Requirements Specification

(Era) Why Factory Method for Medication? Factory Method is a creational design pattern that provides an interface for creating objects in a superclass, but allows subclasses to alter the type of objects that will be created. Use Case Context: The Medication entity in your ERD has subclasses or categories such as Antibiotic, Painkiller, and Vaccine.

These types share common fields (name, category, stock quantity, unit price) but differ in initialization logic, usage protocols, and potentially stock control behavior.

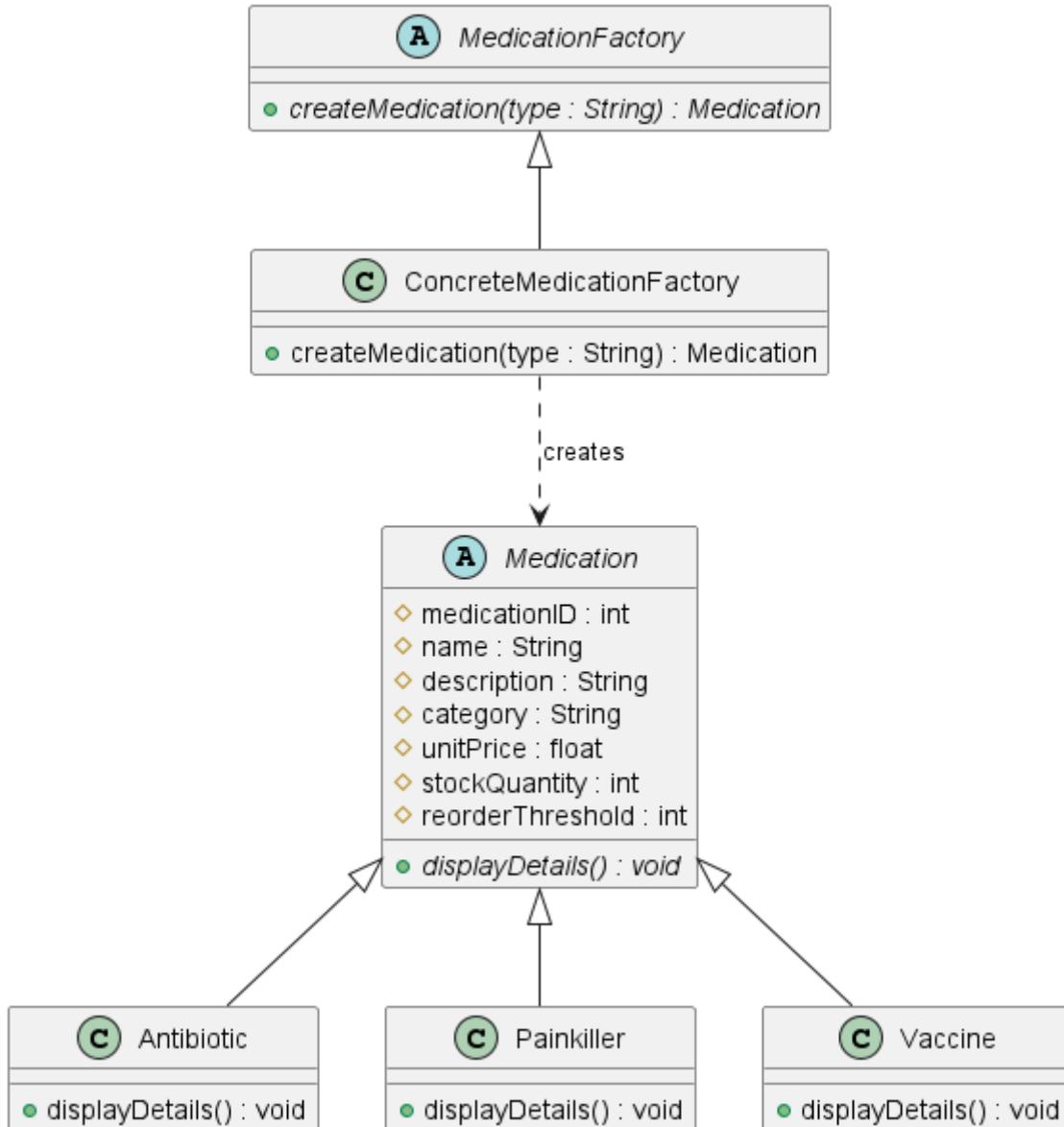
Why Factory Method:

You need to instantiate medication objects based on dynamic input (e.g., prescription type or user input).

The Factory Method pattern provides a single point of creation while hiding subclass instantiation logic.

Makes the system extensible: you can easily add new medication types (e.g., Dewormer, Steroid) without changing existing creation logic.

Benefits are: Promotes open/closed principle: open for extension, closed for modification & simplifies medication creation in a large-scale system with inventory control and prescription tracking.



Veterinary Management System Requirements Specification

(Erla) Why Proxy for MedicalRecord. Proxy is a structural design pattern that lets you provide a substitute or placeholder for another object. A proxy controls access to the original object, allowing you to perform something either before or after the request gets through to the original object. Use Case Context: MedicalRecord contains sensitive data: diagnosis, vet notes, treatment plans. Access to these records must be controlled, logged, and potentially filtered based on user role (vet, admin, pet owner).

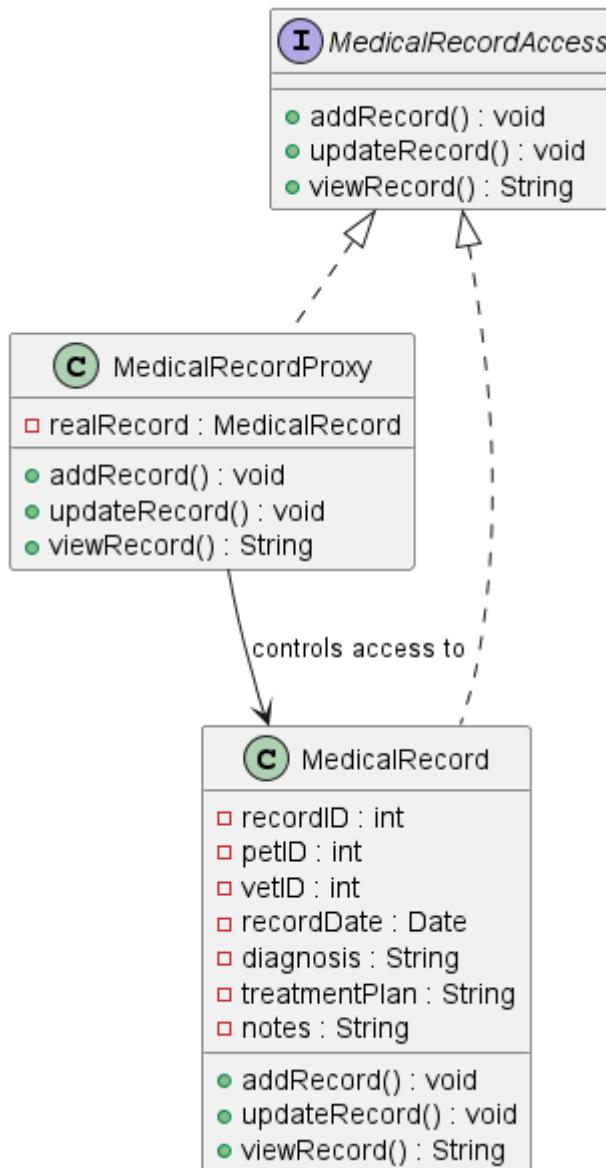
Why Proxy: You need to control access to medical data based on authorization.

The Proxy acts as a security layer between the request and the actual data.

It can perform role checks, audit logging, and even return redacted views when needed.

Benefits are: Enforces secure access and separation of concerns.

Avoids direct exposure of the data model, helping with compliance (GDPR/HIPAA).



Veterinary Management System Requirements Specification

BONUS DIAGRAM: (Erla) **Observer Pattern** for the notification system. Observer is a behavioral design pattern that lets you define a subscription mechanism to notify multiple objects about any events that happen to the object they're observing. This pattern is perfect for handling appointment reminders, payment notifications, and medical record updates where multiple components need to be notified of changes.

Why Observer Pattern for VMS?

Use Case Context: The VMS needs to notify multiple stakeholders when certain events occur:

Appointment reminders (SMS, Email, In-app notifications)

Payment confirmations

Medical record updates

Emergency alerts

Inventory low-stock warnings

Why Observer? Decouples notification logic from core business operations, allows dynamic subscription/unsubscription of notification channels, supports multiple notification types simultaneously & makes the system extensible for new notification methods

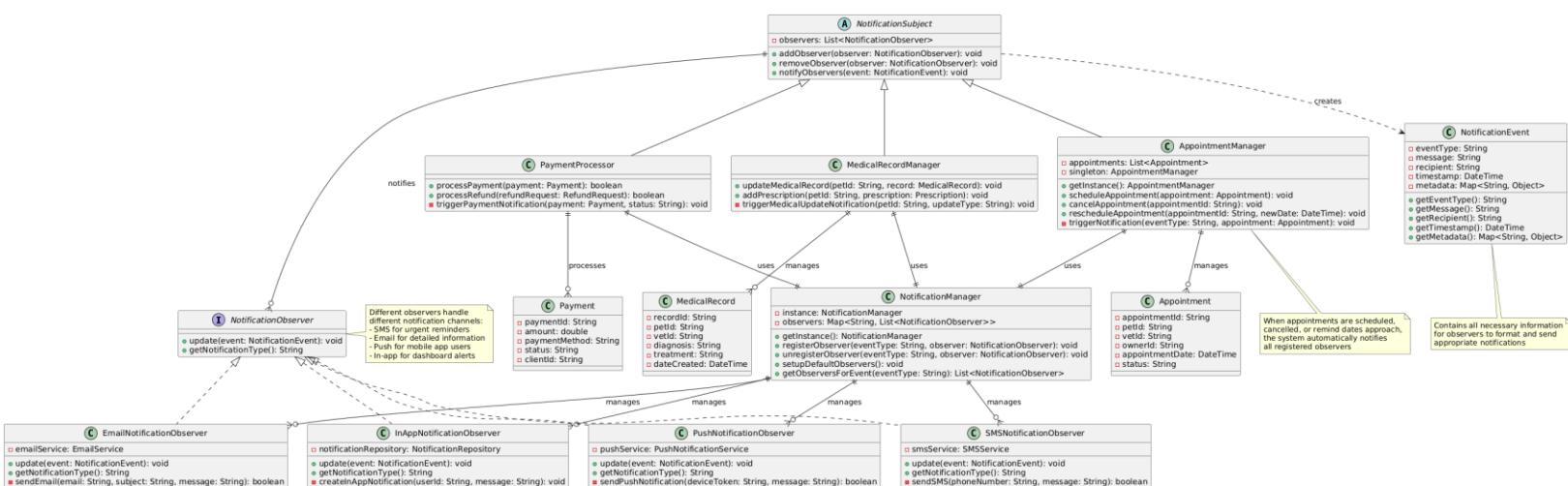
Benefits:

Loose coupling between event sources and notification handlers

Easy to add new notification channels without modifying existing code

Centralized notification management

Supports real-time updates across the system



Use Cases and their respective student

Emi Brahaj	17-20
Era Malaj	25-29
Enea Cane	01-14
Erla Luzi	09-12
Eridjon Krrashi	13-16
Laert Plangarica	05-08
Nadir Alani	21-24