

Software Requirements Specification for Database for an Animal Shelter

Prepared by

Kriti Bharadwaj(PES1UG22CS295)
and
Keya Kesani(PES1UG22CS285)

PES University, Bengaluru
BTech CSE
Software Engineering

28th August 2024

Contents

Revision History	1
1 Introduction	2
1.1 Purpose	2
1.2 Intended Audience	2
1.3 Intended Use	2
1.4 Product Scope	3
1.5 Risk Definition	3
1.6 References	3
2 Overall Description	4
2.1 Product Perspective	4
2.2 Product Functions	4
2.3 User Classes and Characteristics	4
2.4 Operating Environment	5
2.5 Design and Implementation Constraints	5
2.6 Assumptions and Dependencies	6
3 External Interface Requirements	7
3.0.1 User Interfaces	7
3.0.2 Software Interfaces	7
3.0.3 Communications Interfaces	7
4 Analysis Models	8
5 System Features	9
5.1 Animal Registration	9
5.2 Bath Notification	9
5.3 End of Month Expenditure Display	10
5.4 Health Check-Up Notification	10
5.5 Medicine Expiry Date Notification	10
6 Other Non-Functional Requirements	12

6.1	Performance Requirements	12
6.2	Safety Requirements	12
6.3	Security Requirements	12
6.4	Software Quality Attributes	12
6.5	Business Rules	13
6.6	Domain Requirements	13
6.7	Other Requirements	13
7	Requirement Traceability Matrix	14
7.1	Introduction	14
7.2	Matrix:	14
7.3	Notes	14

Revision History

Revision	Date	Author(s)	Description
1.0	28.08.2024	Kriti Bharadwaj, Keya Kesani	Chapter 1 - 7

Chapter 1

Introduction

1.1 Purpose

The purpose of this Software Requirements Specification (SRS) document is to outline the requirements for the development of a database system for an animal shelter. This database will be designed to manage and organize information about the animals, volunteers, staff, and operations within the shelter. The system aims to streamline shelter operations, improve data accessibility, and support efficient decision-making processes. This document covers all aspects of the system's development, including data management, user interfaces, and system interactions. This version of the SRS is intended for the initial release (version 1.0) of the system.

1.2 Intended Audience

This document is intended for a range of stakeholders involved in the development, implementation, and future maintenance of the animal shelter database system. The primary audience includes:

- **Users:** Shelter staff, volunteers, and management who will interact with the system and rely on it for daily operations.
- **Student Developers:** Software engineers and database administrators responsible for designing and implementing the software. To understand the functional requirements and design specifications during the development and testing phases.
- **Project Managers:** Faculty Supervisors, To oversee the project's progress, ensure that the requirements are met, and provide guidance on the project.
- **Organization's IT Staff:** To facilitate the implementation, integration, and ongoing maintenance of the system within the shelter's operations. To understand the system's design and codebase for making updates and improvements for any future developments.

1.3 Intended Use

The animal shelter database will be used to:

- Store detailed records of animals, including their health history, adoption status, and shelter stay duration.
- Manage volunteer and staff information, including roles, and contact details.
- Track shelter resources such as food, medicine, and equipment.

- Generate reports on shelter activities, including adoption rates, resource usage, and volunteer hours.
- Provide a centralized and accessible platform for shelter management to make informed decisions.

1.4 Product Scope

The animal shelter database software is designed to support the efficient management of an animal shelter's operations. Its primary purpose is to maintain a centralized repository of information related to the animals, staff, volunteers, and resources within the shelter. The system will enable shelter management to streamline processes, reduce manual errors, and improve overall operational efficiency.

1.5 Risk Definition

The development of the animal shelter database project carries several risks:

- **Data Security Risks:** There is a risk of unauthorized access to sensitive shelter information, which could lead to data breaches or misuse of information.
- **Technical Challenges:** The integration of various system components, such as the database, user interface, and reporting tools, may present technical challenges that could impact the project's timeline.
- **User Adoption:** Shelter staff and volunteers may resist adopting the new system due to unfamiliarity or perceived complexity, which could hinder the project's success.
- **Data Migration:** Migrating existing data from paper records or legacy systems to the new database could be time-consuming and error-prone, leading to potential data loss or inconsistencies.

1.6 References

IEEE Standard for Software Requirements Specifications (IEEE Std 830 1998)

Chapter 2

Overall Description

2.1 Product Perspective

The animal shelter database is a new, self-contained software product designed specifically to address the operational needs of animal shelters. This product is not a follow-on member of an existing product family, nor is it a replacement for any existing systems. It is being developed from the ground up to provide a comprehensive solution tailored to the unique requirements of an animal shelter.

The database will serve as a central hub for managing all shelter-related data, including animal records, staff and volunteer information, and resource tracking. If integrated with other systems, such as a shelter's website or accounting software, this database will interface with those external systems to share relevant data securely.

2.2 Product Functions

The animal shelter database will perform the following major functions:

- **Animal Management:** Store and manage detailed records for each animal, including health history, adoption status, and stay duration.
- **Staff and Volunteer Management:** Manage information about staff and volunteers, including schedules, roles, and contact details.
- **Resource Tracking:** Track shelter resources such as food, medicine, and equipment, and monitor their usage.
- **Adoption Process Management:** Facilitate the management of the animal adoption process, including application processing and donation amounts.
- **Reporting and Analytics:** Generate reports on various aspects of shelter operations, such as resource usage, health histories of animals and donations collected.
- **User Access Control:** Implement role-based access control to ensure that users have appropriate access to the data and functions relevant to their roles.

2.3 User Classes and Characteristics

The primary user classes anticipated for this product are:

- **Shelter Administrators:**

- **Characteristics:** High level of technical expertise, responsible for overall system management, including user administration and data integrity.
- **Usage:** Frequent users, primarily using the system for data management, reporting, and oversight of shelter operations.
- **Shelter Staff:**
 - **Characteristics:** Moderate level of technical expertise, responsible for daily operations, including animal care, record keeping, and resource management.
 - **Usage:** Regular users, interacting with the system to update animal records, manage resources, and track daily activities.
- **Volunteers:**
 - **Characteristics:** Varied levels of technical expertise, primarily responsible for assisting with animal care and shelter operations.
 - **Usage:** Occasional users, using the system to log hours, check schedules, and update task statuses.

This system is primarily designed to meet the needs of the shelter administrators and staff, with volunteers being secondary users.

2.4 Operating Environment

The animal shelter database software will operate in the following environment:

- **Hardware:** Servers with adequate processing power, storage capacity, and reliable network connectivity.
- **Operating System:** The software will be compatible with the following operating systems:
 - Linux (Ubuntu 20.04 LTS or later)
 - Windows Server 2019 or later
 - macOS (macOS 11 Big Sur or later)
- **Database Management System (DBMS):** The system will utilize MySQL as the primary database management system.
- **Web Server:** Apache or Nginx will be used as the web server for hosting the application interface.
- **Client-Side Requirements:** Users will access the system through modern web browsers such as Google Chrome, Mozilla Firefox, or Microsoft Edge.

2.5 Design and Implementation Constraints

The development of this system is subject to the following constraints:

- **Hardware:** The system must operate effectively on the shelter's existing hardware, which may have limited processing power or memory.
- **Software Interfaces:** The system must be capable of integrating with existing software solutions in use at the shelter, particularly for communication and financial operations.
- **Security Considerations:** The system must ensure the security of sensitive data, including personal information of adopters and animal medical records. Security measures like user authentication, data encryption, and secure backups must be implemented.

2.6 Assumptions and Dependencies

The following assumptions and dependencies are identified for the animal shelter database project:

- **Assumptions:**

- The shelter’s IT infrastructure will be upgraded to meet the hardware requirements before the software is deployed.
- The staff and volunteers will receive adequate training to use the new system effectively.
- The shelter will continue to have reliable internet access to facilitate cloud-based backups and remote access.

- **Dependencies:**

- The project depends on the availability and reliability of third-party software components, such as the selected DBMS and web server.
- The project relies on timely integration with external systems, including the shelter’s website and accounting software.
- The successful deployment of the software is contingent on the completion of a data migration plan that accurately transfers existing records into the new database.

Chapter 3

External Interface Requirements

3.0.1 User Interfaces

The software will provide a user-friendly interface for shelter staff to manage animal records, adoption processes, and volunteer information. The interface will include standard elements such as navigation menus, buttons for common actions, and form fields for data entry. Consistency will be maintained across all screens, with clear labeling and accessible help options. The design will follow accessibility guidelines with clear fonts, high-contrast colors, and standardized elements.

3.0.2 Software Interfaces

The system will interface with the shelter's existing database management system and any relevant third-party software used for payment processing or reporting. Data will be exchanged securely using standard protocols, ensuring compatibility and seamless integration with other software components.

3.0.3 Communications Interfaces

The software will support communication functions such as email notifications for adoption inquiries and alerts for upcoming vaccinations. It will use standard protocols like SMTP for email and HTTPS for secure web communication. All communications will be encrypted to protect sensitive information.

Chapter 4

Analysis Models

- Entity-Relationship Diagrams:

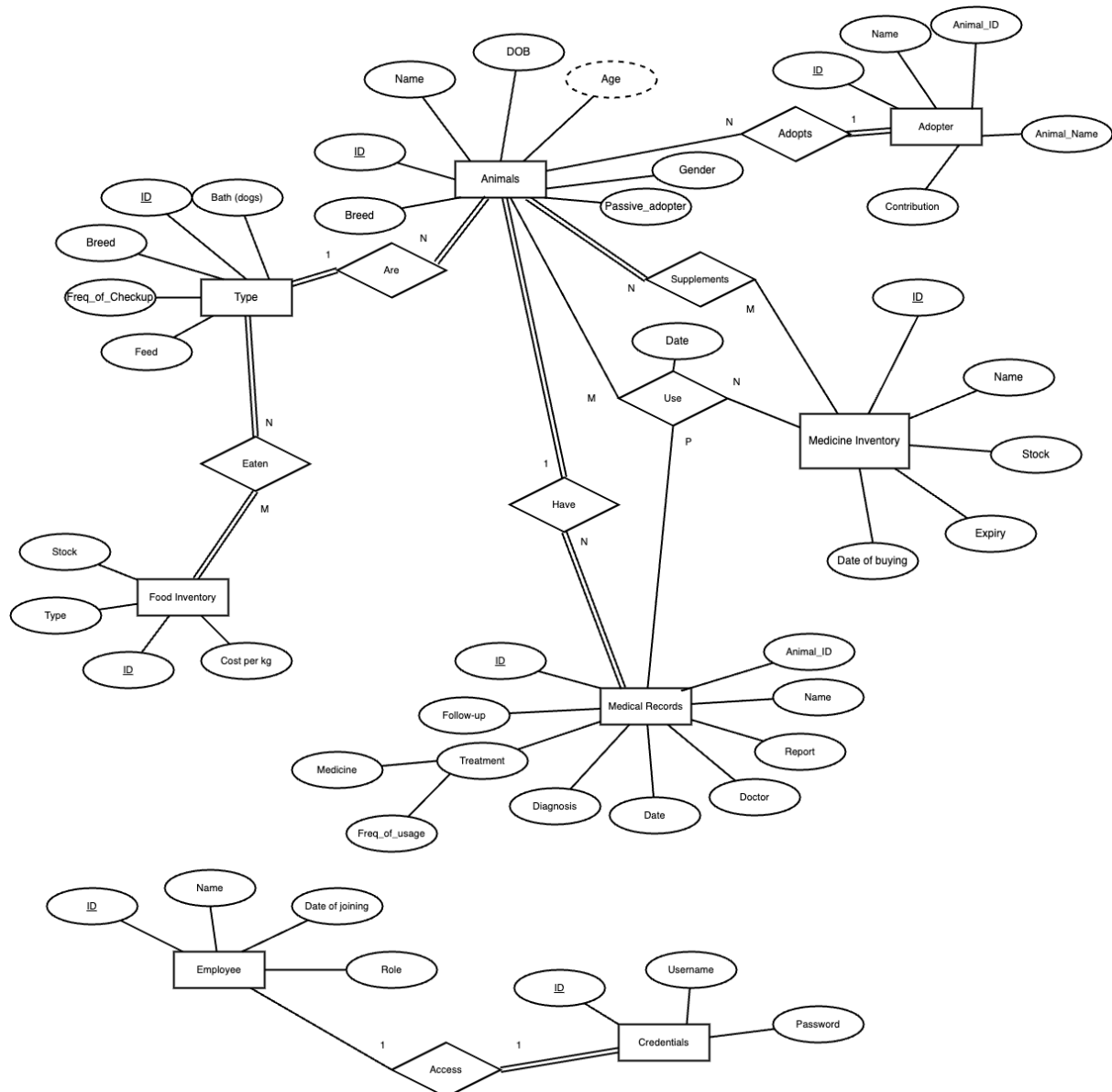


Figure 4.1: Entity-Relationship Diagram for the Animal Shelter Database

Chapter 5

System Features

5.1 Animal Registration

Description and Priority

This feature allows staff to register new animals into the system, recording details such as species, breed, age, and health status. **Priority: High**

Stimulus/Response Sequences

- **Stimulus:** A staff member enters the animal's details into the system.
- **Response:** The system saves the information and confirms successful registration.

Functional Requirements

- **REQ-1:** The system shall allow staff to enter and save animal details, including species, breed, age, and health status.
- **REQ-2:** The system shall validate all required fields before saving the animal record.
- **REQ-3:** The system shall display a confirmation message upon successful registration.

5.2 Bath Notification

Description and Priority

This feature sends notifications to staff when the dogs are due for a bath. **Priority: Medium**

Stimulus/Response Sequences

- **Stimulus:** The system detects that a certain period has passed since the last bath.
- **Response:** The system sends a notification to staff.

Functional Requirements

- **REQ-1:** The system shall track the last bath date for the dogs.
- **REQ-2:** The system shall allow staff to set a bathing schedule.
- **REQ-3:** The system shall send notifications when a bath is due.

5.3 End of Month Expenditure Display

Description and Priority

This feature displays the total expenditure for the shelter at the end of each month. **Priority: Medium**

Stimulus/Response Sequences

- **Stimulus:** The month ends, and the system aggregates all expenses.
- **Response:** The system displays a summary of the month's expenditure.

Functional Requirements

- **REQ-1:** The system shall track all expenses incurred by the shelter.
- **REQ-2:** The system shall automatically calculate the total expenditure at the end of the month.
- **REQ-3:** The system shall display a summary report of the expenditures.

5.4 Health Check-Up Notification

Description and Priority

This feature sends notifications to staff when an animal is due for a health check-up. **Priority: High**

Stimulus/Response Sequences

- **Stimulus:** The system detects that a certain period has passed since the last health check-up.
- **Response:** The system sends a notification to staff with the animal's details and recommended check-up schedule.

Functional Requirements

- **REQ-1:** The system shall track the last health check-up date for each animal.
- **REQ-2:** The system shall allow staff to set a health check-up schedule.
- **REQ-3:** The system shall send notifications when a check-up is due.

5.5 Medicine Expiry Date Notification

Description and Priority

This feature notifies staff when medicines are nearing their expiry date. **Priority: High**

Stimulus/Response Sequences

- **Stimulus:** The system detects that a medicine's expiry date is approaching.
- **Response:** The system sends a notification to staff with details of the medicine and its expiry date.

Functional Requirements

- **REQ-1:** The system shall track the expiry date of all medicines in the shelter.
- **REQ-2:** The system shall send notifications to staff before the expiry date of any medicine.
- **REQ-3:** The system shall allow staff to update or remove expired medicines from the inventory.

Note: This is not the final list of features. Additional features may be identified and added during the development process.

Chapter 6

Other Non-Functional Requirements

6.1 Performance Requirements

- The system shall process and respond to user inputs within 2 seconds under normal operating conditions.
- The system shall generate monthly expenditure reports within 5 minutes after the end of the month.

6.2 Safety Requirements

- The system shall ensure that all data is backed up daily to prevent loss of information.
- The system shall implement error-handling procedures to prevent data corruption in case of unexpected shutdowns.
- The system shall comply with relevant safety regulations and guidelines to ensure user safety during system operation.

6.3 Security Requirements

- The system shall require user authentication via username and password for access to sensitive features.
- The system shall encrypt sensitive data, including personal and medical information of animals.

6.4 Software Quality Attributes

- **Reliability:** The system shall achieve quality uptime over a year.
- **Maintainability:** The system code shall be modular and well-documented to facilitate easy updates and bug fixes.
- **Usability:** The user interface shall be intuitive, with a high user satisfaction rating.
- **Scalability:** The system shall be able to handle an increase in data and users without requiring significant rework.

6.5 Business Rules

- Only authorized staff members with appropriate roles can access the animal registration and health check-up notifications.
- All expenditures must be reviewed and approved by the finance department before being finalized in the system.
- Medicine expiry dates must be checked and updated by the inventory manager regularly.

6.6 Domain Requirements

- The system shall comply with local animal shelter regulations and standards for data management and reporting.
- The system shall integrate with existing shelter management tools and databases, as specified by the shelter's IT infrastructure.

6.7 Other Requirements

- The database shall be implemented using MySQL to ensure compatibility with existing systems.
- The system must comply with legal requirements for data protection and privacy.

Chapter 7

Requirement Traceability Matrix

7.1 Introduction

The Requirement Traceability Matrix (RTM) provides a structured approach to ensure that all requirements of the Animal Shelter Database project are addressed throughout the development lifecycle. It links each requirement to its corresponding design, development, and testing stages.

7.2 Matrix:

Requirement ID	Brief Description	Design Ref.	Development Task Ref.	Test Case ID
REQ-1	Allow staff to register new animals	DS1: Register Interface	DEV1: Develop registration function	T1
REQ-2	Validate animal record fields	DS2: Validation Rules	DEV2: Design validation logic	T2
REQ-3	Display registration confirmation	DS3: Confirmation Message	DEV3: Create confirmation display	T3
REQ-4	Notify staff for dog bath	DS4: Bath Notification System	DEV4: Configure bath notifications	T4
REQ-5	Display end-of-month expenditure	DS5: Expenditure Report	DEV5: Generate expenditure report	T5
REQ-6	Notify for animal health check-up	DS6: Health Check Notification	DEV6: Set up health check notifications	T6
REQ-7	Alert for medicine expiry	DS7: Medicine Expiry Alert	DEV7: Code medicine expiry alerts	T7

7.3 Notes

- The Design References column references specific design documents detailing how each requirement will be implemented.
- The Development Task References column indicates the tasks associated with each requirement.
- Test Case IDs list the corresponding test cases that will validate each requirement.