SOFTWARE REQUIREMENT SPECIFICATION

2.1 Introduction

PawPal is a dynamic web-based platform designed to revolutionize the way pet owners and animal welfare organizations interact. It simplifies the rescue and care process by providing an integrated suite of tools for adoption, medical information, and emergency assistance all in one intuitive interface. PawPal empowers users to streamline the adoption of street dogs and pets, access reliable health facts, and coordinate immediate rescues to save animal lives. The platform is a dynamic web-based platform designed to revolutionize the way pet owners and animal welfare organizations interact. It simplifies the rescue and care process by providing an integrated suite of tools for adoption, medical information, and emergency assistance all in one intuitive interface. PawPal empowers users to streamline the adoption of street dogs and pets, access reliable health facts, and coordinate immediate rescues to save animal lives.

2.2 Purpose

The purpose of this Software Requirement Specification (SRS) is to provide a detailed and structured description of the PawPal system. It defines system functionality, constraints, and requirements to ensure clarity among developers, testers, and stakeholders. This document acts as a baseline for design, development, testing, and future maintenance. The purpose of this Software Requirement Specification (SRS) is to provide a detailed and structured description of the PawPal system. It defines system functionality, constraints, and requirements to ensure clarity among developers, testers, and stakeholders. This document acts as a baseline for design, development, testing, and future maintenance.

2.3 Scope

The scope of the PawPal application includes user registration, vet and rescuer registration, adoption management, rescue reporting, and administrative monitoring. The system supports scalability and future enhancements. The scope of the PawPal application includes user registration, vet and rescuer registration, adoption management, rescue reporting, and administrative monitoring. The system supports scalability and future enhancements.

2.4 Definitions, Acronyms, and Abbreviations

- **SRS**: Refers to Software Requirement Specification, the document detailing all project needs.
- **PHP**: Hypertext Preprocessor, the server-side scripting language used for retrieval and analysis.
- **XAMPP**: A cross-platform web server package containing Apache and MySQL.
- **MySQL**: The relational database management system used for secure storage of rescue records.
- **CFD**: Context Flow Diagram, representing the system's boundary and interactions with external entities.

- **ER Diagram**: Entity-Relationship Diagram, used to model the database connectivity for user profiles.
- **GPS**: Global Positioning System, utilized for setting specific locations on the helpline map.
- **HTTP/HTTPS**: Hypertext Transfer Protocol (Secure) for data transmission between client and server.
- **UI/UX**: User Interface and User Experience, focused on providing an intuitive interface for animal welfare.
- **RBAC**: Role-Based Access Control, ensuring the admin has complete control over specific modules.
- **API**: Application Programming Interface, allowing different software parts to communicate.
- **Admin**: The system administrator with elevated privileges to supervise helpline activities.

2.5 Overview

This document outlines the overall requirements and constraints of the PawPal platform. It describes system interactions, user roles, and operational flow. This document outlines the overall requirements and constraints of the PawPal platform. It describes system interactions, user roles, and operational flow.

2.6 Overall Description

PawPal functions as a centralized platform that allows users to report rescues and adopt pets while enabling vets and rescuers to manage their profiles and locations. Administrators oversee system integrity and supervise helpline activities. PawPal functions as a centralized platform that allows users to report rescues and adopt pets while enabling vets and rescuers to manage their profiles and locations. Administrators oversee system integrity and compliance.

2.7 Product Perspective

The system follows a client-server architecture implemented using the PHP and MySQL stack. It integrates rescue materials, pet profiles, and medical bibliographies into a single solution. The system follows a client-server architecture implemented using the PHP and MySQL stack. It integrates rescue materials, pet profiles, and medical bibliographies into a single solution.

2.8 Product Functions

Key functions include authentication, vet credential verification, pet adoption listings, rescue reporting, medical resource access, and administrative control. Key functions include authentication, vet credential verification, pet adoption listings, rescue reporting, medical resource access, and administrative control.

2.9 User Classes and Characteristics

The system supports general users, professional vets, pet rescuers, and administrators. Each user class has distinct permissions and usage patterns. The system supports general users, professional vets, pet rescuers, and administrators. Each user class has distinct permissions and usage patterns.

2.10 General Constraints

System operation depends on internet availability, database connectivity, and browser compatibility. Security and data protection regulations for user profiles and rescue records must be followed. System operation depends on internet availability and browser compatibility. Security and data protection regulations for user profiles and rescue records must be followed.

2.11 Assumptions and Dependencies

It is assumed that users possess basic technical literacy and access to compatible devices like desktops or mobiles. Vets and rescuers are expected to submit valid credentials and expertise information. It is assumed that users possess basic technical literacy and access to compatible devices. Vets and rescuers are expected to submit valid documentation.

2.12 Specific Requirements

Specific requirements define the system's technical and operational needs for managing pet welfare and rescues. Specific requirements define the system's technical and operational needs for managing pet welfare and rescues.

2.12.1 Software Requirements

The specific software components required for the development and operation of PawPal include:

- **Operating System**: Windows, Linux, or macOS.
- **Front-end Development**: HTML, CSS, and JavaScript.
- **Back-end Scripting**: PHP.
- **Database Management**: MySQL.
- **Web Server**: Apache via XAMPP.
- **Development Tools**: Visual Studio Code or Sublime Text Editor.
- **Client Interface**: Latest versions of Web Browsers such as Google Chrome, Firefox, or Safari .

2.12.2 Hardware Requirements

The system requires the following minimum hardware specifications to ensure stable performance:

- **Processor**: Intel Core i3 or higher.
- **Memory (RAM)**: 4GB or more.
- **Storage Space**: 50GB HDD/SSD minimum.
- **Device Support**: Compatible with Desktop, Mobile, or Tablet devices.

2.12.3 Communication Interface

The application uses optimized PHP scripts for retrieval and analysis to ensure high performance in emergency helpline situations. The application uses optimized PHP scripts for retrieval and analysis to ensure high performance in emergency helpline situations.

2.13 Functional Requirements

The system shall support registration, vet/rescuer verification, adoption management, rescue reporting, and administrative operations. The system shall support registration, vet/rescuer verification, adoption management, rescue reporting, and administrative operations.

2.14 Performance Requirements

The platform should optimize database queries and PHP scripts to handle emergency helpline situations efficiently with low latency. The platform should optimize database queries and PHP scripts to handle emergency helpline situations efficiently with low latency.

2.15 Design Constraints

The design follows a web-based architecture using PHP/MySQL and ensures responsiveness across multiple browser platforms. The design follows a web-based architecture using PHP/MySQL and ensures responsiveness across multiple browser platforms.

2.16 System Attributes

Key attributes include performance optimization for rescue response, secure storage of rescue records, and overall maintainability. Key attributes include performance optimization for rescue response, secure storage of rescue records, and overall maintainability.

2.17 Other Requirements

Additional requirements cover safety and security considerations for animal welfare data and user profiles. Additional requirements cover safety and security considerations for animal welfare data and user profiles.

2.17.1 Safety Requirements

The system ensures safe handling of user profiles, rescue materials, and health data. The system ensures safe handling of user profiles, rescue materials, and health data.

2.17.2 Security Requirements

Security measures include strong database connectivity for secure storage and role-based access control managed by the admin. Security measures include strong database connectivity for secure storage and role-based access control managed by the admin.