Software Requirements Specification (SRS)

# Blood Donation Management System

Prepared by: Arpa Saha Barna  
Student ID: 1910628  
Date: May 2025

## 1. Introduction

1.1 Purpose

The purpose of the Blood Donation Management System is to create an online platform that connects blood donors with recipients. The system facilitates both donation and request processes for users who act as both donors and recipients. The system will manage user registration, blood donation requests, profile management, and include admin functionalities for managing users and viewing statistics.

1.2 Document Conventions

This document follows standard IEEE SRS formatting and uses the following conventions:  
- Functional requirements are prefixed with FR-  
- Non-functional requirements are prefixed with NFR-  
- Tables and diagrams are used for clarity where applicable.

1.3 Intended Audience and Reading Suggestions

This document is intended for:  
- Developers to understand system requirements  
- Testers to validate functionality  
- Project Managers to track system progress  
- End Users & Clients to review system features  
Readers should focus on Section 3 (System Features) for key functionalities and Section 5 (Nonfunctional Requirements) for system constraints and security considerations.

1.4 Project Scope

The Blood Donation Management System will provide a user-friendly platform where donors and recipients can both register and manage their profiles. Users will be able to search for donors and request blood as well. Admins will have the ability to view all registered users and delete user accounts if necessary.

1.5 References

IEEE Standard 830-1998 (Software Requirements Specification)  
PHP, MySQL, and JavaScript documentation  
Bootstrap/Tailwind CSS documentation for UI development

## 2. Overall Description

2.1 Product Perspective

The Blood Donation Management System is a web-based application that functions as a centralized platform for blood donation activities, where users can both donate and request blood. The system includes an admin panel for managing user data, monitoring donation trends, and handling user requests.

2.2 Product Features

The key features of the system are:  
- User registration for both donation and request functionalities.  
- Profile management with options to update availability and deactivate accounts.  
- Donors and recipients can search for blood donors based on blood type and location.  
- Blood request system where recipients can submit requests with urgency.  
- Admin panel for verifying donor profiles and managing user accounts.  
- Notifications for donation and request submissions.  
- History of past donations and requests.

2.3 User Classes and Characteristics

The system has three primary user roles:  
- User (Donor & Recipient): Can register, update profiles, and respond to requests. A user can both donate and request blood.  
- Admin: Can view all users, verify their profiles, manage requests, and delete users if necessary.  
- The system will also support data privacy and secure user authentication.

2.4 Operating Environment

Frontend: HTML, CSS (Bootstrap/Tailwind), JavaScript  
Backend: PHP (Vanilla)  
Database: MySQL  
Browser Compatibility: Chrome, Firefox, Edge

2.5 Design and Implementation Constraints

The system must use PHP and MySQL for the backend, ensuring data privacy and secure authentication.  
Data encryption should be implemented for sensitive user information.

2.6 User Documentation

User Guide (for donors and recipients)  
Admin Manual (for admin operations)

2.7 Assumptions and Dependencies

Users must have an internet connection.  
The system will rely on Google Maps API (optional) for donor location services.

## 3. System Features

### 3.1 Functional Requirements

FR-1: User Registration and Authentication  
Users can register by providing their name, email, password, blood group, and area. Passwords will be encrypted. Users can also log in securely.  
  
FR-2: Donor Profile Management  
Donors can update their availability and location. They can also deactivate their profile temporarily.  
  
FR-3: Search for Donors  
Recipients can search for donors by blood type and location. Donors can update their profile to show their availability.  
  
FR-4: Blood Request System  
Recipients can submit blood requests specifying urgency and hospital details. Donors will receive notifications about nearby requests.  
  
FR-5: Admin Dashboard  
Admins can view user profiles, approve or reject donation requests, and delete users if needed.  
  
FR-6: Donation History  
Users can track their past donations and requests.

3.2 Non-Functional Requirements

NFR-1: Performance Requirements  
The system should handle at least 100 concurrent users without performance degradation.  
  
NFR-2: Safety Requirements  
Regular backups of the database must be maintained.  
  
NFR-3: Security Requirements  
Role-based access control (RBAC) should be implemented for different user types. Data encryption must be applied to sensitive information.  
  
NFR-4: Software Quality Attributes  
The system must be modular and easily maintainable. It should also be scalable for future enhancements.

## 4. Monitoring Strategy

The system's performance will be monitored regularly. Key metrics will include user engagement, response times, and server load. Admin logs will be reviewed to ensure the integrity of user data, and any issues will be addressed proactively. Regular backups will ensure data safety.

## 5. Risk Management

The following risks have been identified:  
1. Data Breaches: Ensure data encryption and secure login systems to prevent unauthorized access.  
2. System Downtime: Implement redundant server systems and backup plans to minimize downtime.  
3. Scalability Issues: Build the system with modular architecture to handle future growth.  
4. User Errors: Implement clear user interfaces and error messages to guide users effectively.

## 6. Change Management Process

Changes to the system will be evaluated based on their impact on functionality and security. Any updates or changes will go through the following process:  
1. Request for Change (RFC) submission  
2. Impact analysis and review  
3. Development and testing of changes  
4. Deployment and monitoring of changes  
5. Post-deployment review