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 connect blood donors with recipients, allowing users to donate and request blood. The system handles registration, requests, and donation processes, with admin functionalities for managing users and viewing statistics.

1.2 Document Conventions

Functional requirements are prefixed with FR-, Non-functional requirements with NFR-, and relevant tables/diagrams are used for clarity.

1.3 Intended Audience and Reading Suggestions

This document is for developers, testers, project managers, and end users to understand system requirements.

1.4 Project Scope

The system allows users to register, donate, request blood, and track donations, with admins managing and viewing users.

1.5 References

IEEE Standard 830-1998, PHP, MySQL, JavaScript, Bootstrap/Tailwind CSS documentation.

## 3. System Features

### 3.1 Functional Requirements

FR-1: User Registration and Authentication  
Users can register with personal details and log in securely with password encryption.

FR-2: Donor Profile Management  
Donors can update availability, location, and deactivate their profile if needed.

FR-3: Search for Donors  
Recipients can search for donors by blood type and location.

FR-4: Blood Request System  
Recipients can submit requests with urgency; donors get notifications.

FR-5: Admin Dashboard  
Admins can manage users, verify profiles, and delete accounts as necessary.

FR-6: Donation History  
Users can track past donations and requests.

### 3.2 Non-Functional Requirements

NFR-1: Performance Requirements  
The system should handle at least 100 concurrent users.

NFR-2: Safety Requirements  
Regular backups of the database must be performed.

NFR-3: Security Requirements  
Role-based access control (RBAC) and data encryption are necessary.

NFR-4: Software Quality Attributes  
The system should be modular, scalable, and maintainable.

## 4. Requirements Analysis & Negotiation

4.1 Examined Against SMART Objectives

The SMART objectives for this system are:  
Specific: Clear user roles (donor, recipient, admin).  
Measurable: Track blood donations, requests, and user activity.  
Achievable: Using PHP, MySQL for the backend, and Tailwind for UI.  
Relevant: Addresses urgent needs for blood donations in various locations.  
Time-bound: System to be launched within 6 months.

4.2 Prioritized by MoSCoW

Must-have: Registration, Blood Request System, Admin Dashboard.  
Should-have: Search by blood type and location.  
Could-have: Google Maps integration for better donor location tracking.  
Won't-have: Mobile app for the first release.

## 5. Classification of Requirements

5.1 Functional Requirements

These define the core functionality of the system, including user registration, search for donors, blood request handling, and profile management.

5.2 Non-Functional Requirements

These ensure system performance, security, and scalability. Regular backups, encrypted data storage, and role-based access control are required.