Software Requirements Specification

For

Blood and Plasma Donation Platform

Prepared by

Aysha Siddika Methila (231014160)
Mahmudul Hasan Zeshan (231014050)
Abida Binte Asif (231014142)
Jumana Muntaha (231014125)
Tasneem Ahmed (231014146)

BPDP Group

Date: 16th of July, 2025

Table of Contents

Abstract

In emergency medical situations, timely access to compatible blood and plasma is critical. The current system in Bangladesh lacks a centralized and responsive platform for matching donors with recipients. This document outlines the functional and non-functional requirements for a proposed web-based system that aims to connect verified blood/plasma donors with hospitals and patients in real time. The software requirements are structured according to IEEE SRS standards.

Keywords

Blood donation, Plasma matching, Software Requirements Specification, Real-time system, Donor network, IEEE SRS

1. Introduction

1.1 Purpose

The purpose of this document is to define the software requirements for the Blood and Plasma Donation Platform. It provides a detailed functional and non-functional specification for the system, ensuring the availability and timely delivery of blood and plasma to patients in need. This document is intended for software developers, project stakeholders, and end users.

1.2 Scope

This software system aims to serve as a centralized platform that connects blood/plasma donors with recipients and hospitals. It provides real-time matching, donor tracking, secure communication, and notifications to enhance efficiency in emergency medical cases. The system will be accessible via web and mobile interfaces for different user roles including donors, recipients, hospitals, and system admins.

1.3 Definitions, Acronyms, and Abbreviations

- SRS: Software Requirements Specification
- UI: User Interface
- UX: User Experience
- RBAC: Role-Based Access Control
- JWT: JSON Web Token
- NID: National ID

1.4 References

[1] IEEE Std 830-1998, IEEE Recommended Practice for Software Requirements Specifications.
[2] WHO Blood Safety and Availability Fact Sheet.
[3] Bangladesh Red Crescent Society – Blood Services.

1.5 Overview

The rest of this document is structured as follows: Section 2 describes the overall system features. Section 3 lists the external interfaces. Section 4 provides specific functional and non-functional requirements.

2. Overall Description

2.1 Product Perspective

This product is an independent and self-contained system. It is not a part of any existing software suite. The system will be deployed as a cloud-based application accessible from desktops and mobile devices. Integration with third-party messaging and geolocation APIs will be required.

2.2 Product Functions

The main functions of the system include:
- User registration and role-based authentication
- Donor and recipient profile management
- Blood/plasma request submission
- Smart donor-recipient matching
- Real-time notification system
- Secure communication between users
- Identity verification and donation tracking

2.3 User Classes and Characteristics

- Donors: Individuals willing to donate blood or plasma.
- Recipients: Patients or their representatives requesting blood/plasma.
- Hospitals: Authorized medical institutions managing blood requests.
- Admins: System operators managing data integrity and verifying users.

2.4 Operating Environment

The system will run on standard web browsers (Chrome, Firefox, Safari). Backend services will be hosted on a cloud platform (e.g., AWS or Firebase) with secure database management.

2.5 Design and Implementation Constraints

The system must comply with data privacy laws in Bangladesh. All sensitive data must be encrypted. The user interface should be simple and responsive for low-bandwidth users.

2.6 User Documentation

A user manual and FAQ section will be provided for each user role. Online support and tutorials will be integrated within the application.

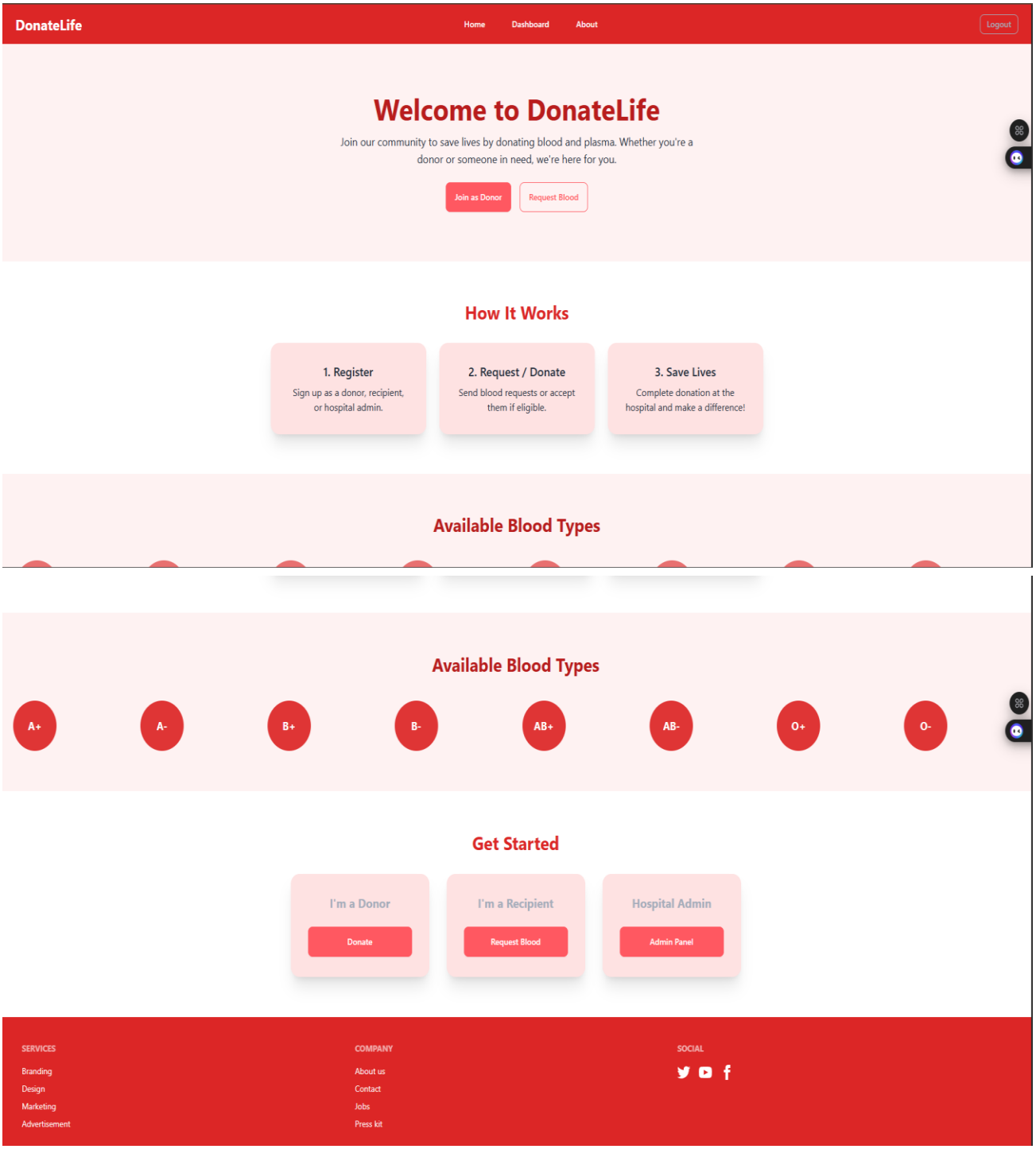
2.7 Assumptions and Dependencies

It is assumed that users have access to mobile devices or internet-enabled computers. The system depends on third-party APIs for sending notifications and verifying geolocation. Users must provide valid identification for verification.

3. External Interface Requirements

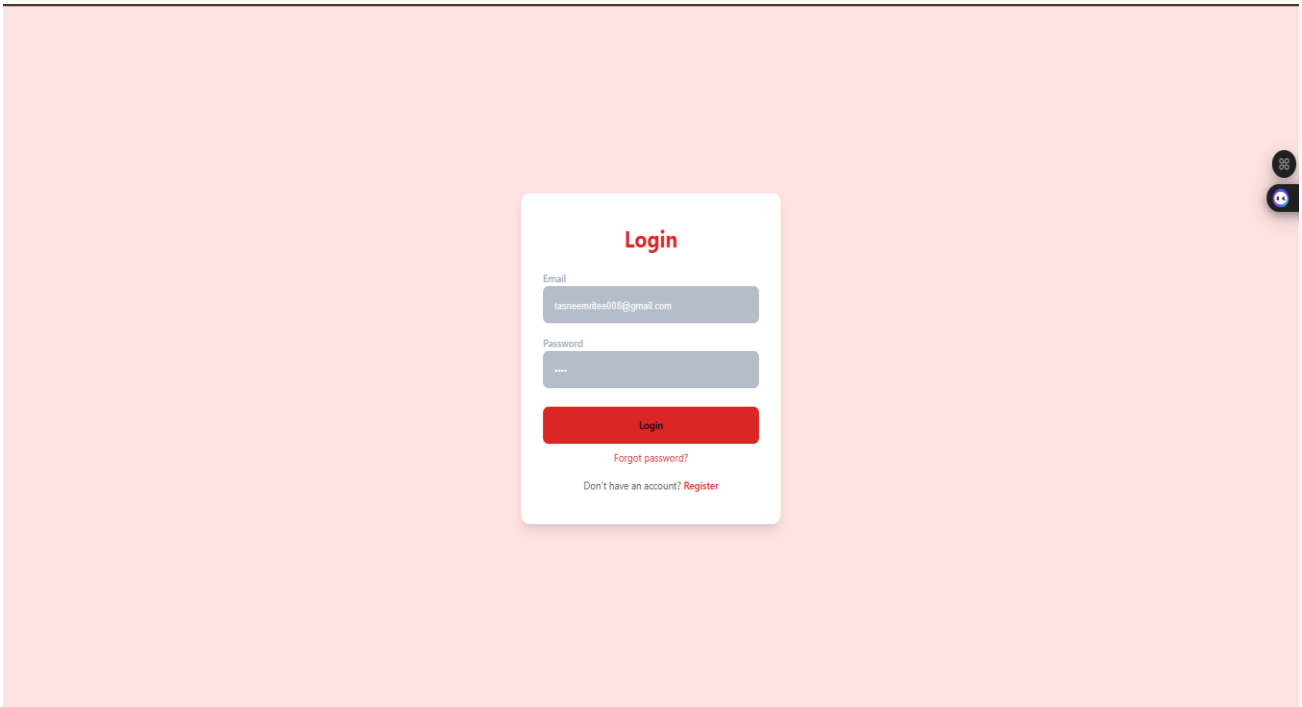
3.1 User Interfaces

The application will have a user-friendly graphical interface accessible via web browsers. Users will have different dashboards based on their roles (donor, recipient, hospital, admin). Features include registration, login, request submission, profile editing, chat, and viewing donation history.

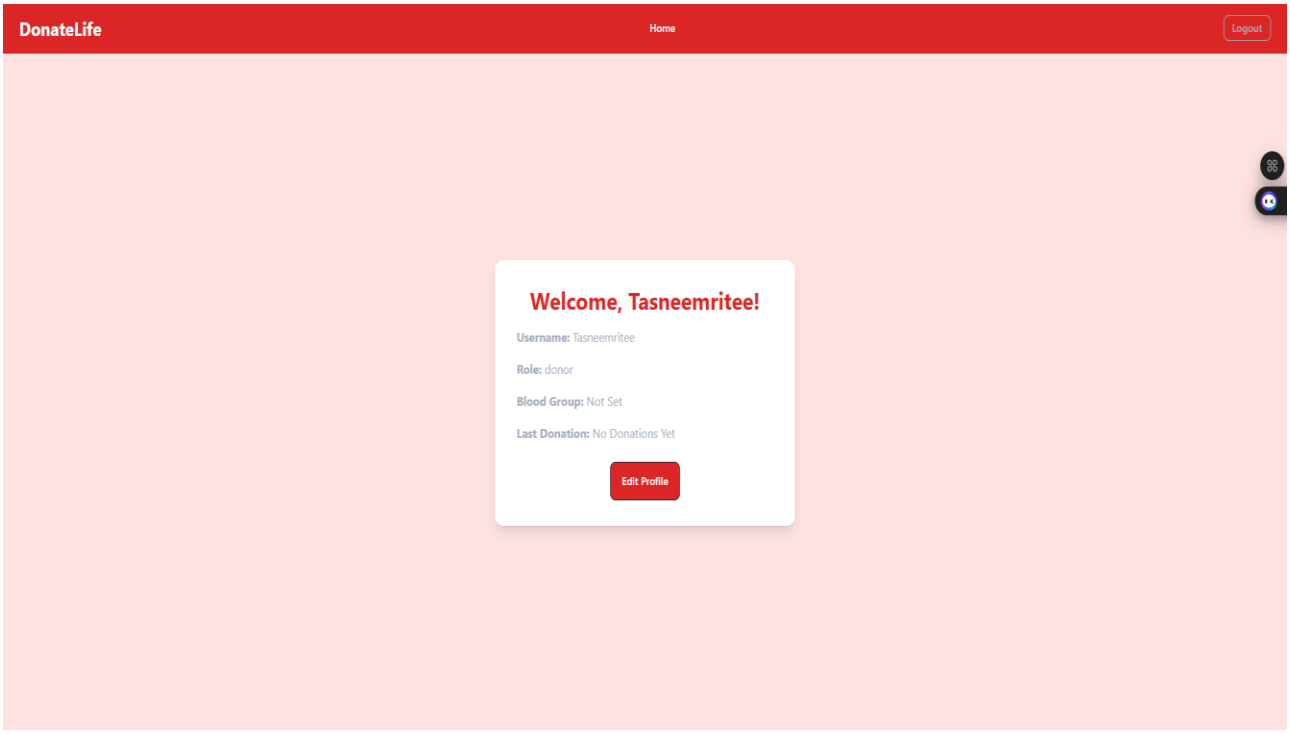


Home Page

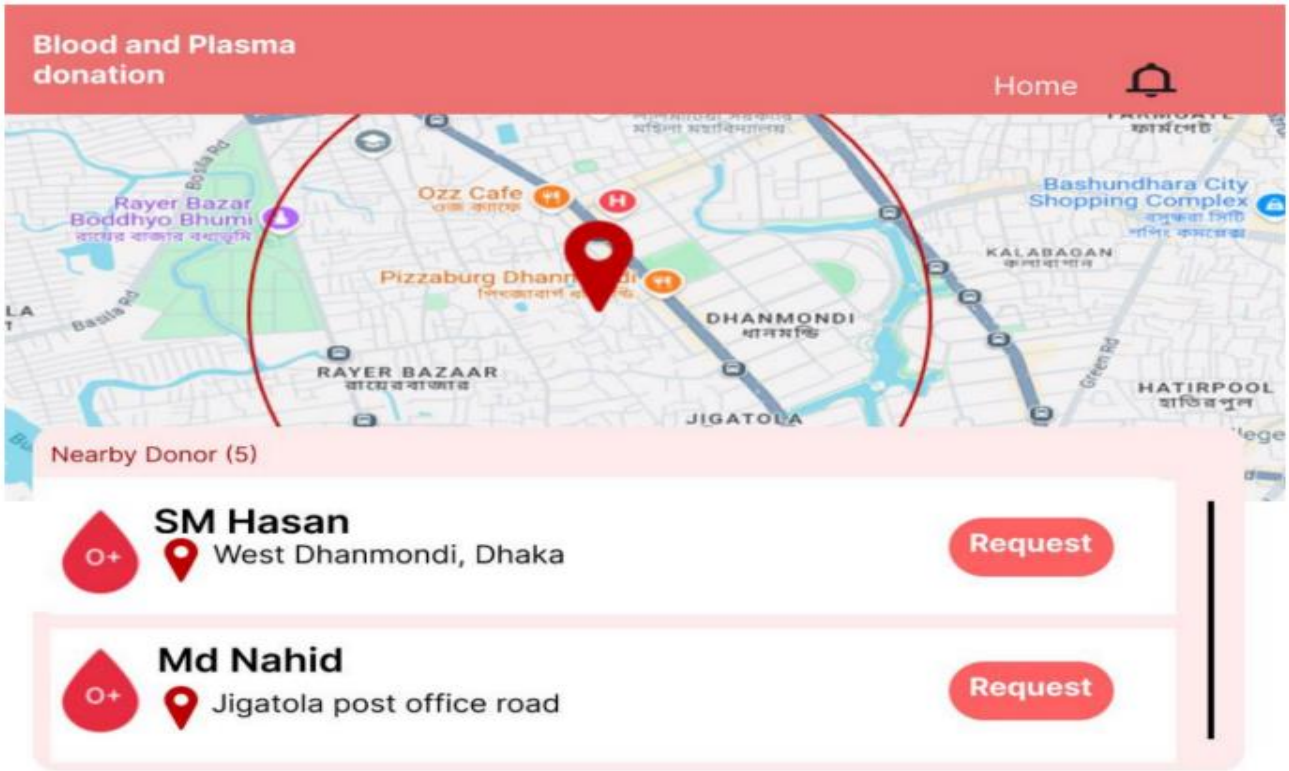
Register Page



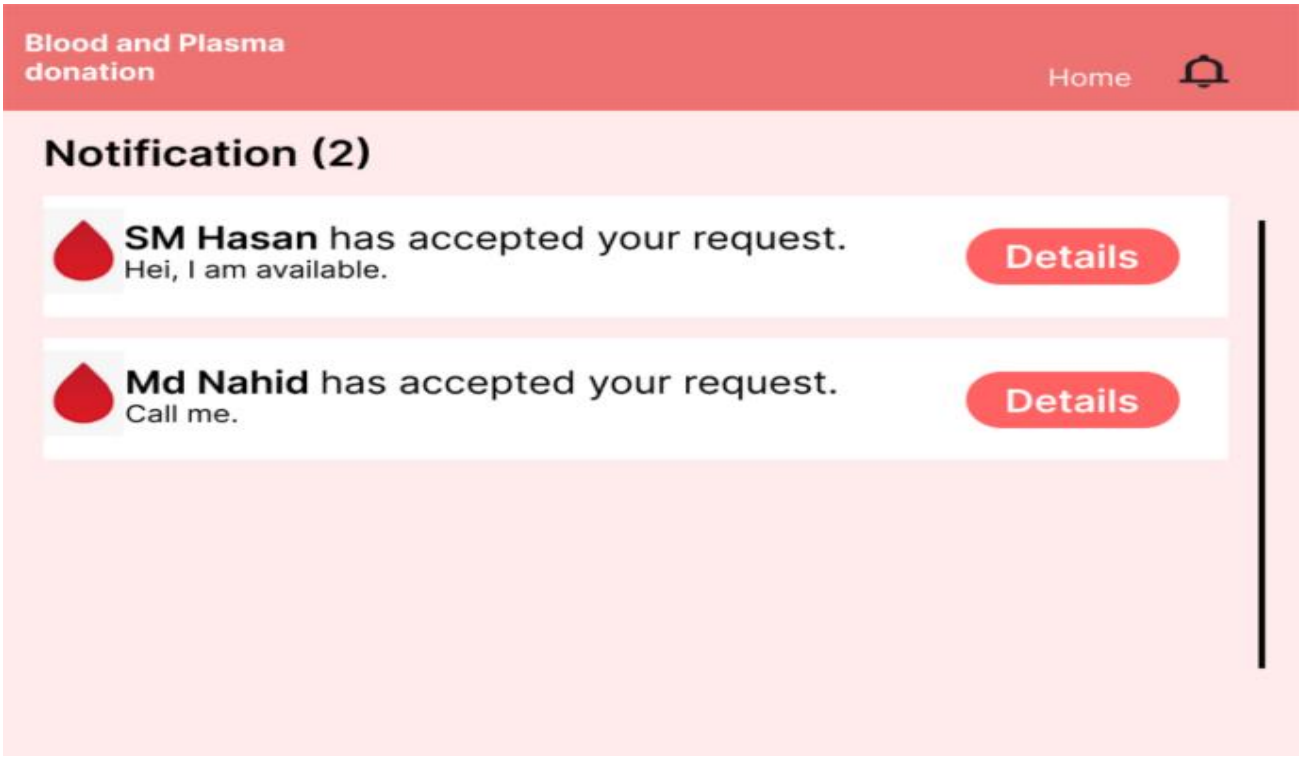
Login Page



Donor Dashboard



Request Submission Page



Notification Page

3.2 Hardware Interfaces

The system does not require any specialized hardware. It will operate on standard computing devices such as:
- Smartphones (Android/iOS)
- Tablets
- Desktop and Laptop Computers

Users only need a device capable of accessing the web or running the mobile app.

3.3 Software Interfaces

The system will interface with the following external software:
- Firebase or AWS (cloud backend and database)
- RESTful APIs for internal communication
- Google Maps API for location services
- SMS/Email APIs for notification delivery (e.g., Twilio, SendGrid)

3.4 Communication Interfaces

Communication between components and users will be handled via:
- HTTPS for secure web communication
- WebSocket or Firebase Realtime Database for instant messaging
- JSON format for API requests and responses

4. Specific Requirements

4.1 Functional Requirements

Each functional requirement is specified using the format: FR-x.y.z

4.1.1 User Registration and Authentication

Description and Priority: Users (donor, recipient, hospital, admin) securely register and log in using email or phone. (**High Priority)**

Stimulus/Response Sequences:

- User enters registration info → system validates → account created
- User enters credentials → system checks → login allowed
- User forgets password → system provides recovery option

Functional Requirements:

- FR-1.1.1: Allow registration using email/phone and password
- FR-1.1.2: Support role-based login (donor, recipient, hospital, admin)
- FR-1.1.3: Validate credentials
- FR-1.1.4: Provide password recovery via email/SMS

4.1.2 Donor/Recipient Profile Management

Description and Priority: Donors and recipients manage personal details and donation history. (**High Priority)**

Stimulus/Response Sequences:

- Donor edits profile → system updates info
- Recipient views donor → only after donor accepts request

Functional Requirements:

- FR-1.2.1: Donors update name, age, gender, blood type, location, contact info
- FR-1.2.2: View and edit profiles
- FR-1.2.3: Maintain donation history and check eligibility
- FR-1.2.4: Donors view recipient details only after request
- FR-1.2.5: Recipients view donor info post acceptance

4.1.3 Blood Request and Matching System

Description and Priority: Patients/hospitals request blood and get matched to donors. (**High Priority)**

Stimulus/Response Sequences:

- Request submitted → system searches → match found → notify receiver

Functional Requirements:

- FR-1.3.1: Submit blood/plasma request
- FR-1.3.2: Match based on blood type, location, availability, history
- FR-1.3.3: Store request history

4.1.4 Real-Time Notifications

Description and Priority: Match alerts and donor responses are sent instantly. (**High Priority)**

Stimulus/Response Sequences:

- Match created → donor notified
- Donor accepts → recipient notified

Functional Requirements:

- FR-1.4.1: Notify donors when a match is found
- FR-1.4.2: Notify recipients on donor response

4.1.5 Communication System

Description and Priority: Matched users and hospitals communicate via chat. (**Medium Priority)**

Stimulus/Response Sequences:

- Match made → chat enabled → users exchange messages

Functional Requirements:

- FR-1.5.1: In-app messaging between matched users
- FR-1.5.2: Hospitals chat with verified donors

4.2 Non-Functional Requirements

Each non-functional requirement is specified using the format: NFR-x.y.z

4.2.1 Performance Requirements

- NFR-2.1.1: The system shall respond to search and match queries within 2 seconds under normal load.
- NFR-2.1.2: Notifications shall be delivered within 1 second.

4.2.2 Scalability Requirements

- NFR-2.2.1: The system shall support at least 100,000 concurrent users.
- NFR-2.2.2: The system shall scale on cloud infrastructure (Firebase, AWS).

4.2.3 Security Requirements

- NFR-2.3.1: The system shall implement Role-Based Access Control (RBAC).
- NFR-2.3.2: The system shall use JWT/OAuth2 for secure login.
- NFR-2.3.3: The system shall optionally use Two-Factor Authentication (2FA).
- NFR-2.3.4: The system shall verify users using national ID or passport scan.

4.2.4 Privacy Requirements

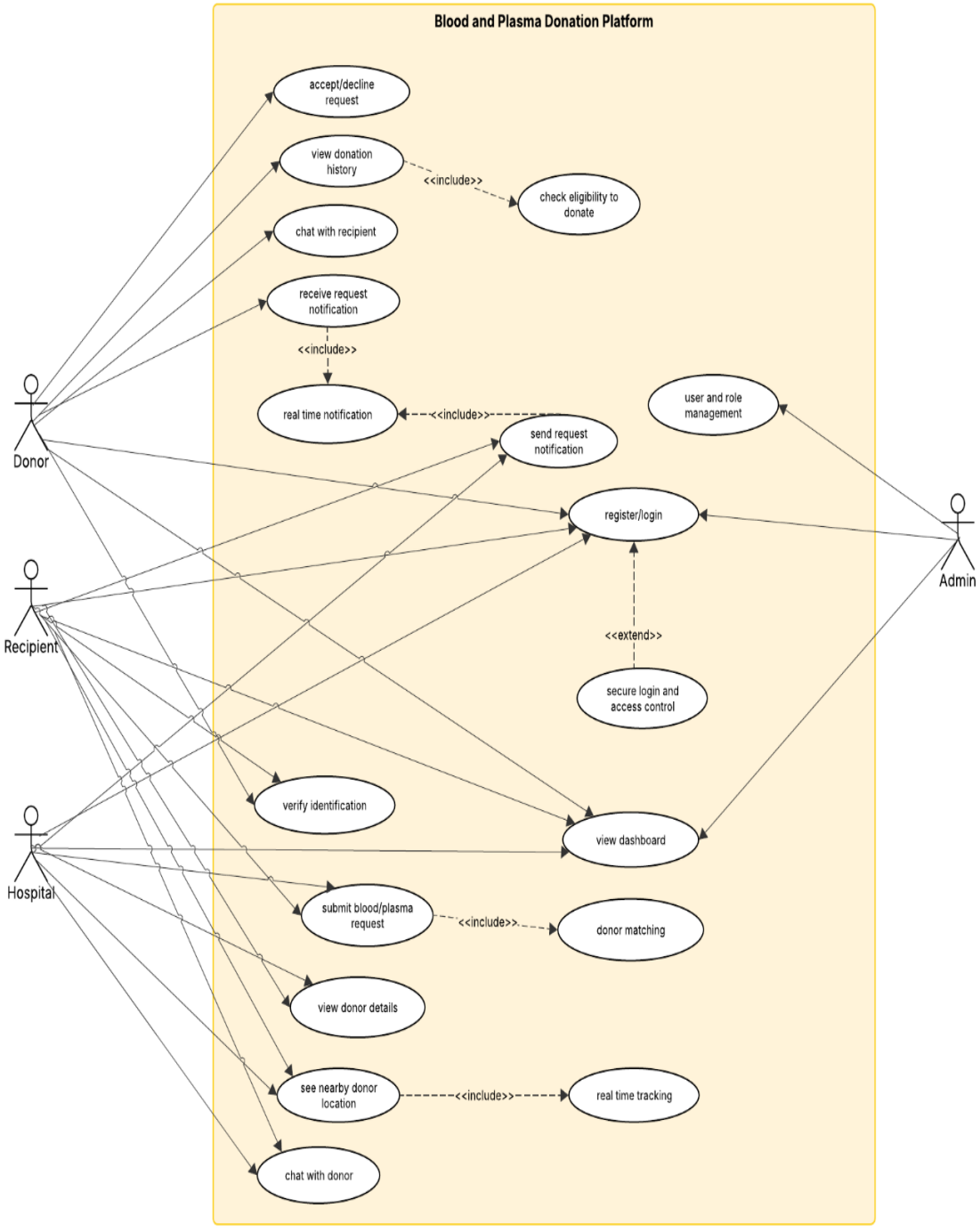
- NFR-2.4.1: All donor and patient data shall be encrypted.
- NFR-2.4.2: Only authorized roles may access personal/medical data.
- NFR-2.4.3: Users may choose anonymity if desired.

5. Appendix A: Glossary

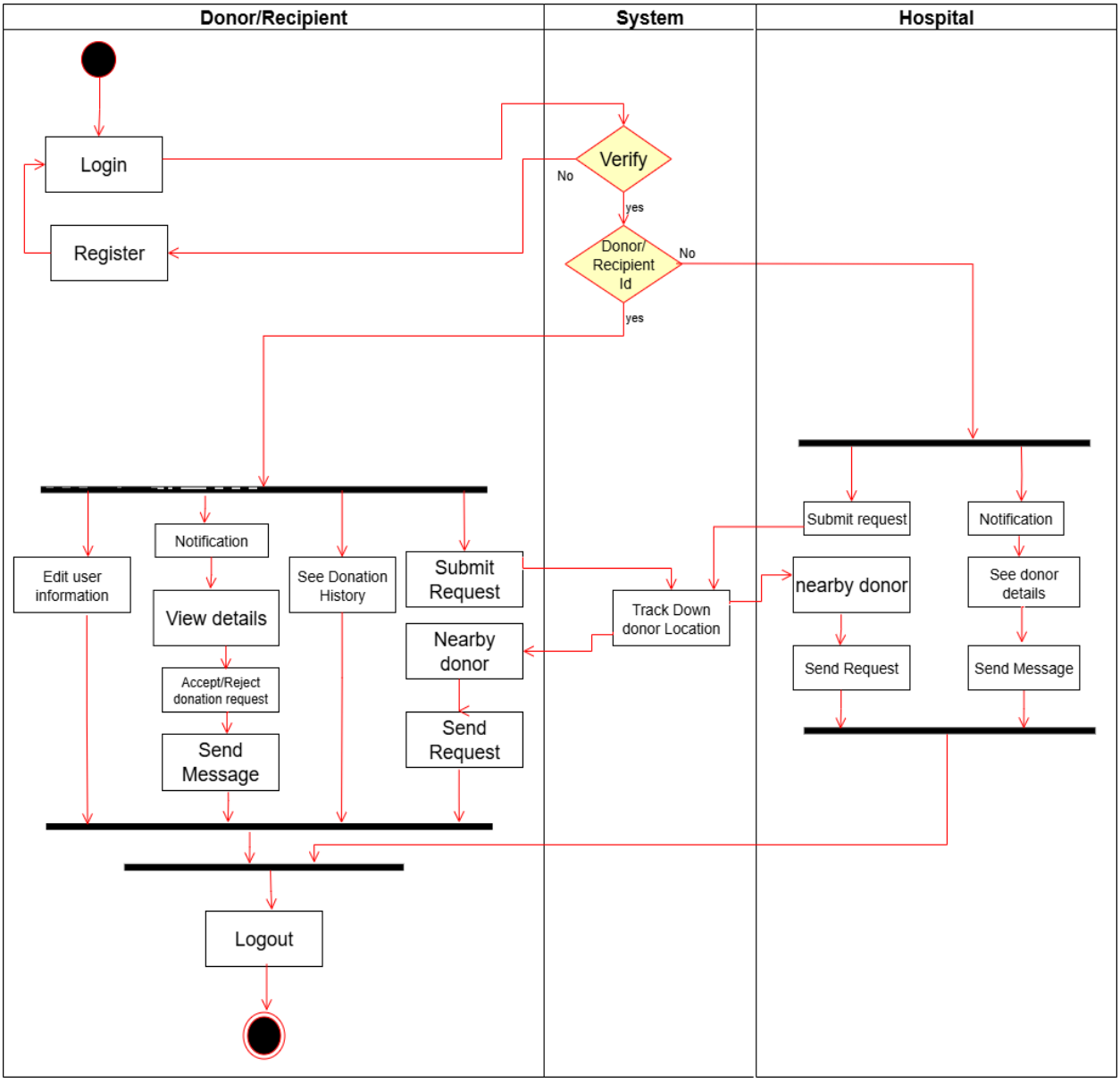
Term	Definition
Donor	A user who offers to donate blood or plasma
Recipient	A user in need of a blood or plasma transfusion
Hospital	Verified medical entity that can request donations and communicate with users
Admin	Authorized user responsible for managing the system
Request	Submission for blood/plasma by a recipient or hospital
Matching System	Algorithm that pairs donors and recipients based on criteria
Donation History	Record of past donations made by a user
Real-Time Notification	Immediate alerts sent to users regarding system events

6. Appendix B: Analysis Models

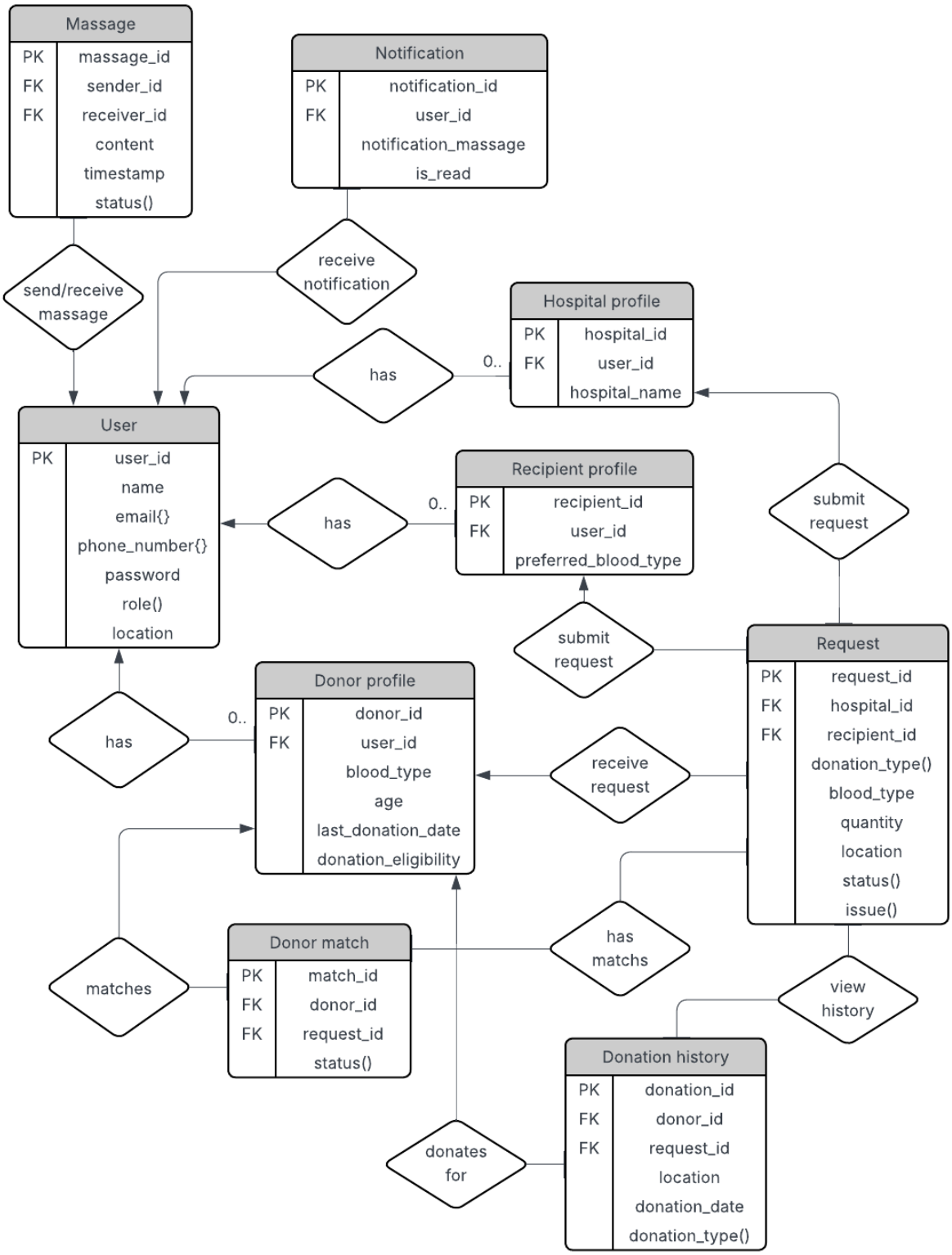
- ## Use Case Diagram



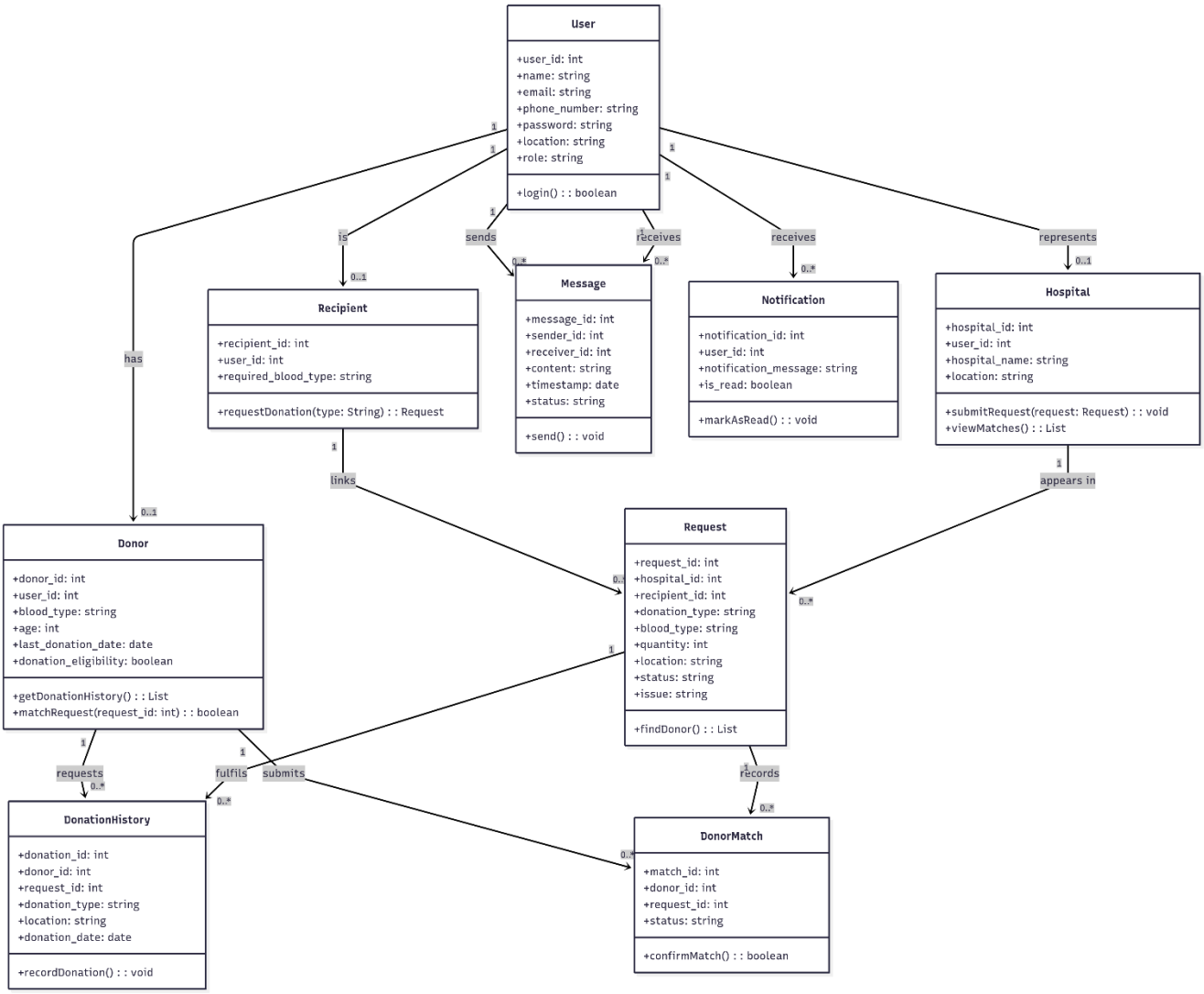
- ## Activity Diagram



- **ER Diagram**


Massage
PK massage_id
FK sender_id
FK receiver_id
content
timestamp
status()
Notification
PK notification_id
FK user_id
notification_massage
is_read
send/receive massage
receive notification
Hospital profile
PK hospital_id
FK user_id
hospital_name
has
0..
User
PK user_id
name
email{}
phone_number{}
password
role()
location
Recipient profile
PK recipient_id
FK user_id
preferred_blood_type
has
0..
submit request
submit request
Request
PK request_id
FK hospital_id
FK recipient_id
donation_type()
blood_type
quantity
location
status()
issue()
Donor profile
PK donor_id
FK user_id
blood_type
age
last_donation_date
donation_eligibility
has
0..
receive request
has matchs
matches
Donor match
PK match_id
FK donor_id
FK request_id
status()
view history
Donation history
PK donation_id
FK donor_id
FK request_id
location
donation_date
donation_type()
donates for

- **Class Diagram**



7. Appendix C: To Be Determined List (TBD)

- Choice of communication platform (WhatsApp, Telegram, Facebook group, etc.)
- Privacy policy and data protection protocols for member information
- Decision on developing a mobile app or continuing with manual systems