Bangabandhu Sheikh Mujibur Rahman Science and Technology University, Gopalganj



Requirement Analysis Report

On

‘Blood Bank Management System’

Web Application

Dept. Of Computer Science and Engineering, BSMRSTU

**Table of Contents**

1. **INTRODUCTION**.......................................................................................................... 4

1.1 System Purpose ........................................................................................................... 4

1.2 System Scope .............................................................................................................. 4

1.3 Goals of the Team ....................................................................................................... 4

1.4 Process Model ............................................................................................................. 5

1.5 Team Organization ...................................................................................................... 5

1. **RESEARCH** .................................................................................................................... 6

2.1 Market Research ......................................................................................................... 6

2.2 Target Audience ...........................................................................................................6

2.3 Technical Research ..................................................................................................... 6

2.3.1 Front-End Technology ................................................................................. 6

2.3.2 Back-End Architecture ................................................................................. 7

2.3.3 Database Systems ......................................................................................... 7

2.3.4 Security Practices ......................................................................................... 7

2.3.5 Tools and Services ........................................................................................7

1. **DESCRIPTION** ............................................................................................................... 8

3.1 Homepage ................................................................................................................... 8

3.2 Dashboard ................................................................................................................... 8

3.3 Registration ................................................................................................................. 8

3.4 Login ........................................................................................................................... 9

3.5 Logout ......................................................................................................................... 9

3.6 User Management ....................................................................................................... 9

3.7 Additional Features...................................................................................................... 9

1. **REQUIREMENTS**......................................................................................................... 10

4.1 Functional Requirements .......................................................................................... 10

4.1.1 User Roles ………...................................................................................... 10

4.1.2 Core Features ………................................................................................. 11

4.2 Performance Requirements ....................................................................................... 11

4.3 Software Requirements ............................................................................................ 12

4.4 Hardware Requirement ............................................................................................. 13

4.4.1 For Development ........................................................................................13

4.4.2 For Hosting .................................................................................................13

1. **SYSTEM ANALYSIS AND MODELING** ............................................................. 14

5.1 Functional Modeling ........................................................................................... 14

5.1.1 Level 0 Data Flow Diagram .................................................................. 14

5.1.2 Level 1 Data Flow Diagram ................................................................. 15

5.2 Use Case Analysis ................................................................................................ 16

5.2.1 Blood Donation Use Case .................................................................... 16

5.2.2 Request Fulfillment Use Case ............................................................... 16

1. **INTRODUCTION**

The Blood Bank Management System is a web-based platform developed to address the challenges in managing blood donations, inventory, and requests efficiently. The platform aims to simplify the workflow for blood banks, donors, hospitals, and patients by providing an interactive and transparent system. This document provides a comprehensive requirement analysis from ideation to implementation, covering both front-end and back-end development.

**1.1 System Purpose**

The Blood Bank Management System (BBMS) is a web-based solution developed to streamline the management of blood banks. It enables efficient handling of blood donations, inventory, and requests from hospitals and patients. The system bridges the gap between donors and recipients through a user-friendly platform, ensuring that life-saving resources are available when needed.

**1.2 System Scope**

The scope of the Blood Bank Management System includes:

* Registration and management of blood donors and hospitals.
* Real-time inventory tracking of blood stocks at participating blood banks.
* Processing and fulfillment of blood requests with transparent tracking mechanisms.
* Managing and providing real-time update of blood donation campaigns and location
* Notifications and reporting features for all stakeholders.

**1.3 Team Goals**

* Create a reliable system for managing blood inventory and requests.
* Enhance user experience (UX) with a secure and interactive interface (UI).
* Deliver a scalable and maintainable solution that minimizes downtime and operational inefficiencies.

**1.4 Process Model**

The project adopts the Agile Methodology, allowing iterative development with frequent feedback and incremental improvements.

The development process following the stages:

1. **Requirement Gathering:** Identifying user needs, app functionalities, and technical specifications.
2. **Design & Prototyping:** Creating wireframes, mockups, and prototypes for both the frontend and backend.
3. **Development & Implementation:** Dividing tasks into sprints, building components and APIs iteratively using React and Django.
4. **Testing & Debugging:** Conducting unit, integration, and user acceptance testing (UAT) to ensure quality and stability.
5. **Deployment:** Preparing the app for deployment on cloud platforms (e.g., Vercel, Heroku) and making it publicly accessible.
6. **Maintenance & Updates:** Continuously improving the app based on user feedback and monitoring its performance.

**1.5 Team Organization**

* Team Lead: Project management and stakeholder communication.
* Front-end Developer: UI/UX design, client-side programming.
* Back-end Developer: Database setup, API development, and server-side logic.

**2. RESEARCH**

**2.1 Market Research**

1. **Industry Trends:** Growing demand for safe blood, digital platforms for efficient management, and AI for demand prediction.
2. **Competitors:** Apps like Red Cross and BloodBD offer donor tracking, geolocation matching, and real-time updates.
3. **Opportunities:** Address rural blood shortages, AI-driven donor-recipient matching, and collaboration between hospitals and NGOs.
4. **Challenges:** Ensuring data security, donor retention, and logistics management.

**2.2 Target Audience**

1. **Primary:** Blood donors (aged 18–60), hospitals, and clinics requiring reliable blood supplies.
2. **Secondary:** Patients, caregivers, and NGOs organizing blood donation campaigns.
3. **Tertiary:** Local communities, volunteers, and government health agencies for policy support and awareness drives.

**2.3 Technical Research**

**2.3.1 Front-End Development**

**HTML5, CSS3, JavaScript:** For basic structure and styling.

**React.js:** Selected for its component-based architecture, enabling faster development and better maintainability.

**Bootstrap:** Ensures a responsive design that works across all devices.

**2.3.2 Back-End Development**

**Node.js:** For server-side scripting and API development.

**Express.js:** For building RESTful APIs.

**RESTful APIs:** Provide structured communication between the client and the server.

**2.3.3 Database Systems**

**MySQL:** Chosen for its reliability in handling relational data such as user profiles, blood inventory, and requests.

**Sequelize ORM:** To interact with the database efficiently.

**2.3.4 Security Practices**

Implementation of secure authentication (JWT) and data encryption techniques to safeguard sensitive data.

**2.3.5 Tools and Services**

**Version Control:** GitHub for source code management.

**Hosting:** AWS or Heroku for deploying the web application.

**Email Services:** Nodemailer for notification services.

**3. DESCRIPTION**

**3.1 Homepage**

The Homepage serves as the entry point for users, offering an intuitive interface to access essential features and information.

* **Search Functionality:** Users can search for nearby donors, blood banks, or donation drives.
* **Blood Availability:** Displays real-time data on available blood types in partnered banks.
* **Emergency Requests:** Allows patients or caregivers to raise urgent blood requests.
* **Upcoming Campaigns:** Highlights free blood donation camps and events organized by the system.

**3.2 Dashboard**

The Dashboard is a personalized hub for registered users.

* **Donor Dashboard:** Includes donation history, eligibility status, and reminders for future donations.
* **Admin Dashboard:** Enables hospitals and NGOs to manage blood inventories, view requests, and schedule donation drives.
* **Analytics:** Displays insights such as donor statistics, blood demand patterns, and campaign results.

**3.3 Registration**

The Registration feature enables new users to sign up and create an account on the platform. Key elements include:

* **User-Friendly Form:** A simple and intuitive form requiring basic details such as username, email, and password.
* **Email Verification:** Sends a confirmation email to verify the user’s identity.
* **Error Handling:** Provides clear feedback for invalid inputs or already registered email addresses.

**3.4 Login**

The Login feature allows users to access their accounts securely. Key aspects include:

* **Authentication:** Verifies user credentials against the stored database.
* **Remember Me Option:** Saves user login state for convenience on trusted devices.
* **Error Messages:** Displays messages for incorrect username or password inputs

**3.5 Logout**

The Logout feature ensures user sessions can be terminated securely. Features include:

* **Session Termination:** Ends the active session and clears authentication tokens.
* **Redirect to Home Page:** After logging out, users are redirected to the Home Page.

**3.6 User Management**

The User Management system encompasses functionalities for both regular users and admin users. Features include:

* **Profile Management:** Allows users to update their personal information, including profile picture, bio, and password.
* **Roles and Permissions:** Differentiates between regular users and administrators, with specific privileges for each.
* **Content Moderation Tools:** For administrators, tools to monitor and manage inappropriate content or user behavior.

**3.7 Additional Features**

* **Free Blood Donation Campaigns:** Displays information on upcoming donation drives and allows users to sign up as volunteers or participants.
* **Profile Management:** Users can update their personal information, including health history and donation preferences.
* **Notifications:** Sends alerts for emergencies, donation reminders, or campaign updates.
* **Contact Page:** Includes support details like helpline numbers, email, and live chat for queries.
* **Feedback and Ratings:** Users can provide feedback on campaigns, blood bank services, and overall app experience.

1. **REQUIREMENTS**

**4.1 Functional Requirements**

**4.1.1 User Roles**

* **Admin**
  + Manage user accounts (blood banks, donors, hospitals, employees).
  + Oversee blood inventory and generate system reports.
* **Blood Bank Staff**
  + Update blood inventory after donations or dispatches.
  + Manage donor details and hospital requests.
* **Donor**
  + Register and log in to their accounts.
  + Check donation history and receive eligibility reminders.
* **Receiver**
  + Request blood for patients.
  + Track status of requests.

**4.1.2 Core Features**

* **Blood Availability Tracking:** Real-time inventory management of blood types across partnered banks.
* **Search and Filter:** Allows users to search for blood donors, banks, or campaigns by location, blood type, and availability.
* **Donor Registration and Management:** Enables donors to register, update profiles, and track donation history.
* **Emergency Requests:** Facilitates urgent blood requests with notifications to nearby donors or blood banks.
* **Campaign Management:** Hospitals and NGOs can create, schedule, and manage free blood donation campaigns.
* **Notifications:** Sends reminders for donations, alerts for emergency needs, and updates about upcoming campaigns.
* **Reports and Analytics:** Provides data insights such as donation trends, blood type shortages, and campaign outcomes.

**4.2 Performance Requirements**

* **Page Load Speed:**
  + **Time to First Byte (TTFB):** Less than 200ms.
  + **Full Page Load:** Under 3 seconds on a 4G connection.
* **Scalability:** The system should handle at least 10,000 concurrent users and 1,000 requests per second during peak times.
* **Security:**
  + Data encryption for sensitive information like medical and personal details.
  + Secure authentication mechanisms, including multi-factor authentication.
* **Responsiveness:** Optimized for mobile, tablet, and desktop devices, ensuring a consistent user experience.
* **Reliability:** 99.9% uptime for critical functionalities like emergency requests and donor search.
* **Accessibility:** Adheres to accessibility standards (e.g., WCAG 2.1) to cater to differently-abled users.

**4.3 Software Requirements**

* **Frontend**
  + **Framework:** React.js or Angular.
  + **Styling:** Bootstrap or Tailwind CSS.
* **Backend**
  + **Framework:** Django (Python) or Node.js with Express.js.
  + **API Development:** Django REST Framework or GraphQL.
* **Database**
  + **Relational:** PostgreSQL or MySQL for structured data.
  + **NoSQL:** MongoDB for storing campaign details or logs.
* **Additional Tools**
  + **Version Control:** Git and platforms like GitHub or GitLab.
  + **Caching:** Redis for improving response times.
  + **Deployment:** Docker, Kubernetes, or CI/CD tools like GitHub Actions.
  + **Monitoring:** Prometheus and Grafana for performance tracking.
* **Third-Party Services:**
  + **Notifications:** Twilio for SMS, Firebase Cloud Messaging (FCM) for push notifications.
  + **Cloud Storage:** AWS S3 or Google Cloud for storing documents and reports.

**4.4 Hardware Requirements**

**4.4.1 For Development**

* **Processor:** Quad-core (e.g., Intel i5/i7 or AMD Ryzen 5/7).
* **RAM**: 8GB (16GB recommended for multitasking).
* **Storage**: 256GB SSD minimum.
* **OS:** Windows, macOS, or Linux.

**4.4.2 For Hosting**

* **Small Scale Deployment:**
  + **CPU:** 2 vCPUs.
  + **RAM:** 4GB.
  + **Storage:** 50GB SSD.
* **Medium to Large Scale Deployment:**
  + **CPU:** 4+ vCPUs.
  + **RAM:** 8GB or higher.
  + **Storage:** 100GB SSD or more.
* **Cloud Services:** AWS EC2, DigitalOcean, or Google Cloud with auto-scaling enabled.

This set of requirements ensures the system meets functionality, performance, and scalability expectations, catering to both end-users and administrators.