Malnad College of Engineering

Under the auspices of M.T.E.S ®
(An Autonomous Institution Affiliated to VTU, Belgaum)
P.B No. 21, Hassan-573 202, Karnataka

MINI PROJECT (23IS506)

"A Geo-Location Enabled Intelligent Blood Donation and Management System"

Submitted By:

Name	USN
A Y Kaustubha Sai	4MC23IS002
Bhuvanesh A U	4MC23IS020
Likhith Arya D	4MC23IS056
Manthan Nayak	4MC23IS057

Under the guidance of

Mrs. Tasmiya Anjum H N

Assistant Professor

Department of Information Science and Engineering
Malnad College of Engineering
Hassan - 573 202

2025-26

Malnad College of Engineering

Under the auspices of M.T.E.S ®
(An Autonomous Institution Affiliated to VTU, Belgaum)
P.B No. 21, Hassan-573 202, Karnataka

CERTIFICATE

Certified that the mini project work carried out by 4MC23IS002, 4MC23IS020, 4MC23IS056, 4MC23IS057 is a Bonafede work, submitted during academic year 2025-26, in partial fulfilment for the award of B.E degree in Information Science & Engineering. All the corrections suggested during the internal evaluation are incorporated in the project report. This report has been approved as it satisfies the academic requirements of mini project prescribed for the Bachelor of Engineering degree.

Mrs. Tasmiya Anjum H N
Assistant Professor

Dr. Ananda Babu J
Prof & Head of Dept

Name of The Examiners	Signature With Date
1.	
2.	

Department of Information Science and Engineering
Malnad College of Engineering
Hassan - 573 202
2025-26

ACKNOWLEDGEMENT

We express our sincere gratitude to everyone who contributed to the successful completion of our project titled “Geolocation-Enabled Blood Donation and Request Management System.” This project provided us with a meaningful opportunity to understand real-world challenges associated with emergency healthcare coordination and to develop a practical, technology-driven solution that addresses the critical issue of locating verified blood donors in time-sensitive situations.

We extend our heartfelt thanks to the Department of Information Science and Engineering for offering a supportive learning environment, continuous encouragement, and the essential academic guidance required to pursue this project. The department’s commitment to fostering technical innovation and problem-solving skills has played a significant role in shaping our direction and motivation throughout the development process.

We would like to express our deep gratitude to our Head of the Department, **Dr. Anand Babu J** whose leadership and academic vision inspired us to explore socially impactful systems. His constant support helped us refine our ideas and align our work with real-world community needs.

Our sincere appreciation goes to our project guide, **Mrs. Tasmiya Anjum H N** whose consistent guidance, technical insights, and constructive feedback were invaluable at every stage of development. His expertise helped us understand the nuances of system analysis, application design, geolocation integration, and database structuring. The clarity and direction provided during review sessions greatly enriched the quality and usability of the final system.

We are also thankful to the individuals who participated during requirement analysis, user-experience feedback sessions, and testing. Their observations helped us identify practical improvements related to donor workflows, emergency request handling, live-location tracking, and platform usability. These contributions greatly enhanced the system’s efficiency and real-world applicability.

Lastly, we extend our gratitude to our peers, friends, and family members for their encouragement, patience, and moral support throughout the development period. The experience and learning gained from this project will serve as a strong foundation for our future academic and professional endeavors.

ABSTRACT

The Geolocation-Enabled Blood Donation and Request Management System is a web-based application designed to address the persistent challenges associated with finding reliable and verified blood donors during medical emergencies. Traditional donor-search methods rely heavily on informal communication channels such as social media posts, manual donor lists, or word-of-mouth appeals. These approaches are often slow, unstructured, and ineffective during critical situations where timely access to compatible blood can make a significant difference.

This system introduces a centralized digital platform that connects blood donors and recipients through real-time geolocation services. Donors can register, update their availability, and allow their approximate location to be displayed on an interactive map. Patients or hospital coordinators can quickly generate a blood request, filter available donors by blood group, proximity, and availability status, and contact potential donors directly. The use of integrated mapping APIs and geospatial calculations enhances the system's ability to identify the nearest suitable donors, significantly reducing response time during emergencies.

Developed using Django, Python, HTML/CSS, JavaScript, and mapping technologies such as Leaflet.js or OpenStreetMap, the system provides secure authentication, structured workflows, automated request handling, and reliable record maintenance. A robust backend ensures that all donor information, request logs, and location data are stored securely while maintaining data integrity and privacy.

By digitalizing donor identification and request management, the system overcomes key limitations of manual methods, such as outdated donor lists, inaccurate contact information, and delayed responses. It offers a scalable, user-friendly, and socially impactful solution designed to enhance emergency healthcare support. The proposed platform not only improves the accessibility of life-saving resources but also promotes community engagement in voluntary blood donation.

TABLE OF CONTENTS

CHAPTER 4 – IMPLEMENTATION

CHAPTER 5 – TESTING

CHAPTER 6 – USER MANUAL

CHAPTER 7 – CONCLUSION

Category	Count
Total Tables	8
Total Images	7
Total Screen Shots	8

CHAPTER 1

INTRODUCTION

1.0 Introduction to the Area

Blood donation plays a crucial role in saving lives during medical emergencies, surgical procedures, and chronic illnesses. However, despite the critical need for timely blood availability, the process of identifying suitable donors remains largely dependent on informal and outdated communication channels. Families often struggle to locate donors quickly, especially during emergencies when every minute matters.

With the increasing adoption of digital technologies, there is a strong need for a unified platform that can automate donor search, streamline communication, and provide real-time updates on donor availability and proximity. A geolocation-enabled blood donation system that connects donors and recipients through a live interactive map can significantly improve response times and ensure timely medical support. Such a system brings accuracy, transparency, and efficiency into a domain that currently operates manually and lacks structure.

1.1 Problem Statement

The current method of finding blood donors is slow, unstructured, and unreliable. Patients depend on social media appeals, volunteer groups, or outdated donor lists, leading to delays and uncertainty. Even when potential donors exist nearby, recipients often fail to connect with them due to lack of real-time information.

The proposed project aims to build an automated, geolocation-enabled blood donation management system that identifies the nearest compatible donors, updates availability instantly, and displays all active requests and donors on a live map. This system reduces manual effort, avoids communication breakdowns, and increases the likelihood of timely transfusion support.

1.2 Existing System and Drawbacks

Current blood-donation coordination systems rely heavily on manual communication. Social media posts, WhatsApp broadcasts, and community donor lists are the most commonly used tools. While they help spread information quickly, they do not provide verified, structured, or real-time donor data. As a result, these systems often fail during emergencies when rapid decision-making is critical.

Key Drawbacks of Existing Systems

1. No Real-Time Donor Geolocation

- Existing platforms do not track the current location of donors.
- Recipients cannot determine whether a donor is physically close enough to reach a hospital quickly.

2. Outdated and Unverified Information

- Donor lists become obsolete within months.
- Availability status, health eligibility, and contact details are rarely updated.

3. No Automated Donor–Request Matching

- Current systems do not analyze distance, urgency, or compatibility.
- Request broadcasting is random and often reaches irrelevant groups.

4. Lack of Immediate Emergency Support

- Manual communication delays waste critical time.
- Donors may not receive timely notifications about nearby emergencies.

5. Inefficient and Unorganized Workflow

- Families must individually message donors or groups.
- No centralized platform exists to show all donors and requests in one place.

6. No Integrated Mapping or Visualization

- Existing systems do not use interactive maps to visualize donor distribution.
- Recipients cannot make informed decisions based on proximity.

These drawbacks highlight the urgent need for a technologically advanced system that provides accuracy, speed, and automated support.

1.3 Objectives of the Present Work

- To build a web-based blood donation platform with real-time geolocation features.
- To help users register as donors and manage their blood group, availability, and profile data.
- To allow recipients to create blood requests with urgency level, location, and required units.
- To automatically match donors and requests based on distance and compatibility.
- To display all donors and active requests on a live interactive map.
- To provide real-time updates of donor availability and location.
- To reduce emergency response time and improve blood accessibility.
- To ensure secure authentication and structured workflows for all users.

1.4 Platform & Tools Used

Category	Tools / Technologies Used
Frontend	HTML, CSS, JavaScript, TailwindCSS
Backend	Django (Python Web Framework)
Mapping Library	Leaflet.js, OpenStreetMap
Database	SQLite / MySQL
APIs Used	Browser Geolocation API
Version Control	GitHub
IDE	Visual Studio Code (VS Code)

CHAPTER 2

SYSTEM ANALYSIS

2.1 Literature Survey

Recent studies on digital health systems, blood donation management platforms, and geolocation-enabled emergency response emphasize the growing importance of real-time data, automation, and centralized coordination. Existing research highlights the challenges of manual donor identification, lack of structured workflows, and delays caused by traditional communication methods. Key works relevant to geolocation-enabled blood donation platforms are summarized below.

1. **"A Survey on Digital Blood Donor Management Systems Using Web and Mobile Technologies" – Priya S. et al. (2023)**

   This paper examines the evolution of blood donor management tools and highlights the limitations of traditional coordination methods. The authors analyze different models such as SMS-based alerts, online donor registries, and mobile-based donor applications. They emphasize the importance of structured databases, donor verification, and automated matching between donors and recipients. The study's findings support the need for a unified digital platform that reduces dependency on informal communication channels and enhances reliability—a key focus of the proposed system.

2. **"Emergency Response Enhancement Using Geolocation and Cloud-Based Health Platforms" – N. Harish et al. (2024)**

   This work introduces a geolocation-enabled emergency support system that integrates GPS data with cloud infrastructure to provide assistance during medical crises. The system uses real-time mapping and nearest-resource identification to improve response efficiency. The authors highlight challenges such as location inaccuracies, communication delays, and fragmented data. Their findings directly reinforce the importance of integrating mapping technologies and distance-based donor search into a blood donation application.

3. **"Smart Blood Bank System Using IoT and Automated Data Processing" – K. Thakur and M. Rao (2023)**

   The paper focuses on the use of IoT devices and automated workflows in hospitals to maintain blood inventories and donor records. Although its primary scope is internal hospital management, it highlights the need for real-time donor data, responsive systems, and automated alerts. It also discusses how manual donor lists become outdated quickly, leading to failures during emergencies. This research strengthens the necessity for the dynamic data update and donor availability tracking features adopted in the proposed system.

4. **"Mobile-Based Blood Donation Alert and Tracking System for Emergency Healthcare" – A. Fernandez et al. (2022)**

   This study presents a mobile-driven platform designed to send automated notifications to nearby donors during emergencies. The authors demonstrate that donors respond faster when notified based on distance and urgency. Their results highlight the importance of proximity-based matching, real-time donor engagement, and user-friendly mobile interfaces. These insights significantly influence the design decisions in this project, particularly the inclusion of map-based visualization and automated donor request broadcasting.

Summary of Literature Gaps

Across these literature sources, several gaps consistently emerge. Most existing systems lack real-time geolocation capabilities, preventing users from identifying the nearest compatible donor efficiently. The majority of donor platforms still rely on manual updates, resulting in outdated and unreliable information. Few systems provide automated workflows for donor–recipient matching, leading to delays during emergencies. Existing studies also emphasize the absence of unified platforms that combine donor registration, live mapping, emergency request posting, and donor availability tracking in a single interface. Furthermore, most applications fail to incorporate modern UI design principles, resulting in poor usability and limited adoption. These gaps underline the necessity for an integrated web application that intelligently automates and streamlines the entire blood donation process through geolocation support and real-time updates.

2.2 Findings of Analysis

The analysis of existing blood donation workflows reveals several critical issues that significantly affect the efficiency of emergency medical response. One of the most prominent findings is the heavy reliance on unsystematic communication channels such as WhatsApp groups, social media posts, and verbal networks. These channels lack structure, making it difficult to identify a suitable donor within a critical timeframe. Families often experience delays because donor data is scattered across multiple sources and lacks verification.

Another significant observation is the absence of real-time donor information. Manual donor lists typically do not reflect users' current location, availability, or eligibility status. As donors frequently move between locations or temporarily become unavailable, inaccurate information leads to miscommunication and wasted time. The inability to determine donor proximity results in longer travel durations, undermining the goal of quick transfusion support. The study also highlights a strong need for centralized platforms where users can access reliable donor and request data. Hospitals and volunteer groups benefit greatly from structured systems that offer transparency and consistency. A unified dashboard that displays donors, requests, and distances in real time simplifies coordination and decision-making.

It is also found that donors respond more effectively to structured alerts and clearly defined urgency levels. When emergencies are communicated in an organized digital format, donors feel more motivated and informed compared to random social media broadcasts. Users expect accurate distance calculations, intuitive interfaces, and instant updates, making automation essential.

Overall, the analysis emphasizes the need for a system that consolidates donor profiles, geolocation tracking, request management, and automated matching into a single, intelligent platform.

2.3 System Requirement Specification

Functional Requirements

1. **User Registration and Login:**
   The system must allow users to create accounts and authenticate securely before accessing donor or request features.
2. **Donor Profile Creation:**
   Registered users should be able to set up a donor profile containing blood group, contact number, availability status, and last donation date.
3. **Blood Request Submission:**
   Users must be able to create blood requests by specifying information such as required blood type, urgency level, units needed, and hospital or location details.
4. **Geolocation Capture:**
   The system should capture the user's real-time geolocation using browser APIs, subject to user permission.
5. **Map-Based Donor Visualization:**
   The application must display donors and requests as markers on a live map for easy tracking.
6. **Distance Calculation:**
   The system should calculate distances between donors and request locations using geospatial algorithms.
7. **Real-Time Donor Matching:**
   The system must list nearest donors based on blood group compatibility and physical proximity.
8. **Dashboard Analytics:**
   Users should be able to view key metrics such as active donors, active requests, and average donor distance.
9. **Profile Management:**
   Users must be able to view and update their personal details, blood group, availability, and login information.
10. **Request and Donor Listings:**
    The application must allow users to view structured lists of donors, recipients, and active emergency requests.

Non-Functional Requirements

1. **Portability:**
   The system should be accessible on desktops, laptops, and mobile devices without installation.
2. **Accuracy:**
   All distance calculations, location tracking, and donor–request matching must maintain high accuracy.
3. **Usability:**
   The interface should be intuitive, visually clear, and easy to use for non-technical individuals.
4. **Performance:**
   System responses, map loading, and updates must be fast and efficient, even during high-traffic periods.
5. **Security:**
   User credentials and sensitive donor information should be protected through secure authentication and encrypted storage.
6. **Data Privacy:**
   Donor location and personal details must only be shared with authorized users or during verified emergencies.
7. **Reliability:**
   The system should provide consistent availability and accurate updates without data loss.
8. **Scalability:**
   The architecture should support future expansion, such as hospital integration or SMS alert systems.
9. **Maintainability:**
   The codebase must be modular, documented, and easy to debug or extend.
10. **Ethical Compliance:**
    Collection and usage of location data should follow ethical guidelines and user consent practices.

Additional System Requirements Analysis

In addition to the functional and non-functional requirements listed above, it is essential that the system be designed for long-term sustainability and scalability. Given that the application will handle continuous geolocation updates and emergency-related data, the backend must remain robust under varying operational loads. A reliable database design is required to ensure that donor and request details are always accessible without delays.

Furthermore, the application must maintain a modular architecture that allows new features to be integrated without reworking existing components. This ensures that future enhancements—such as hospital dashboards, mobile app integration, or automated SMS notifications—can be seamlessly incorporated. The system must also comply with best practices in usability and accessibility so that users from diverse backgrounds can interact with it effectively.

Finally, because the application processes sensitive information such as blood type, phone numbers, and location details, stringent ethical and privacy considerations must guide data handling protocols. Users must maintain full control over what information they choose to share, ensuring trust and transparency.

CHAPTER 3

DESIGN

3.1 Database Design

Overview

The Geolocation-Enabled Blood Donation and Request Management System uses a structured relational database model designed to store donor information, blood requests, user accounts, and real-time geolocation data. The design ensures efficiency, reliability, and accuracy when retrieving or updating donor or request records. Django ORM supports seamless interaction between the application logic and the database tables while maintaining strong data integrity and referential consistency.

Since the system manages sensitive personal information—such as contact details, blood type, and location—data security and privacy considerations play an important role in the design. The database schema is normalized to reduce redundancy, improve query performance, and provide consistent updates across the platform. Relationships among entities are defined to support core system functionalities including donor registration, location tracking, and emergency request processing.

Primary Entities & Schemas

1. **User:**
   Stores fundamental authentication and profile details for individuals accessing the system. Each user can act as a donor, recipient, or both depending on their interaction with the platform.
2. **Donor Profile:**
   Contains donor-specific information such as blood group, last donation date, availability status, and geolocation coordinates. Each donor profile is linked to a unique user.
3. **Blood Request:**
   Represents an emergency request created by a user. It includes details such as required blood type, number of units, urgency, hospital location, and coordinates.
4. **Geolocation Data:**
   Stores live or last-known latitude and longitude values of donors and request creators. This data supports distance calculations and map visualization.
5. **Activity Logs / Notifications:**
   Optional table to track alerts, donor responses, and system-level interactions, helpful for future analytics and auditing.

Relationships & Data Flow

A **one-to-one** relationship exists between User and Donor Profile, allowing each user to maintain exactly one donor identity. A **one-to-many** relationship exists between User and Blood Requests, ensuring that individuals can create multiple emergency requests over time.

Geolocation Data is linked directly to users or donors using foreign keys, ensuring consistency and ease of updates during live tracking. Blood Requests include coordinates that help the system compute distances against donor locations dynamically. Through Django ORM, these relations enable smooth querying for donor–request matching and map rendering.

3.1.1 System Architecture Diagram

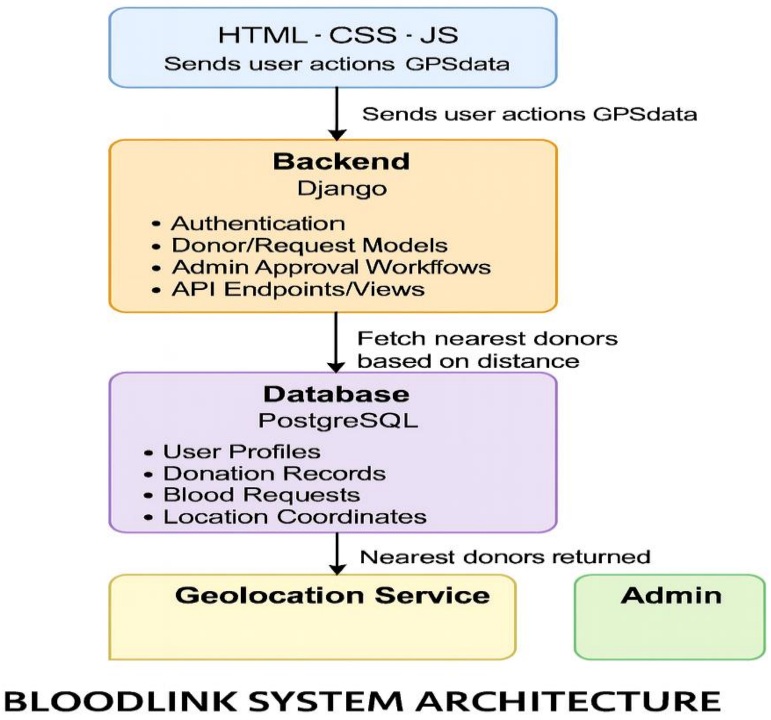


Figure 3.1.1 System Architecture

3.1.2 Data Flow Diagram

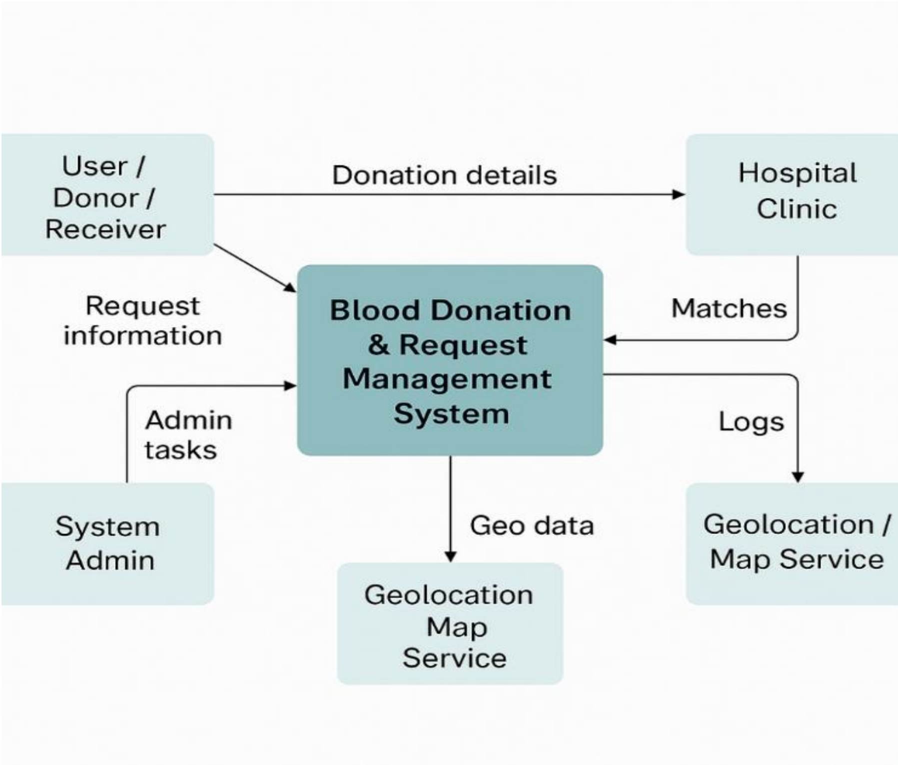


Figure 3.1.2 DFD Level 0

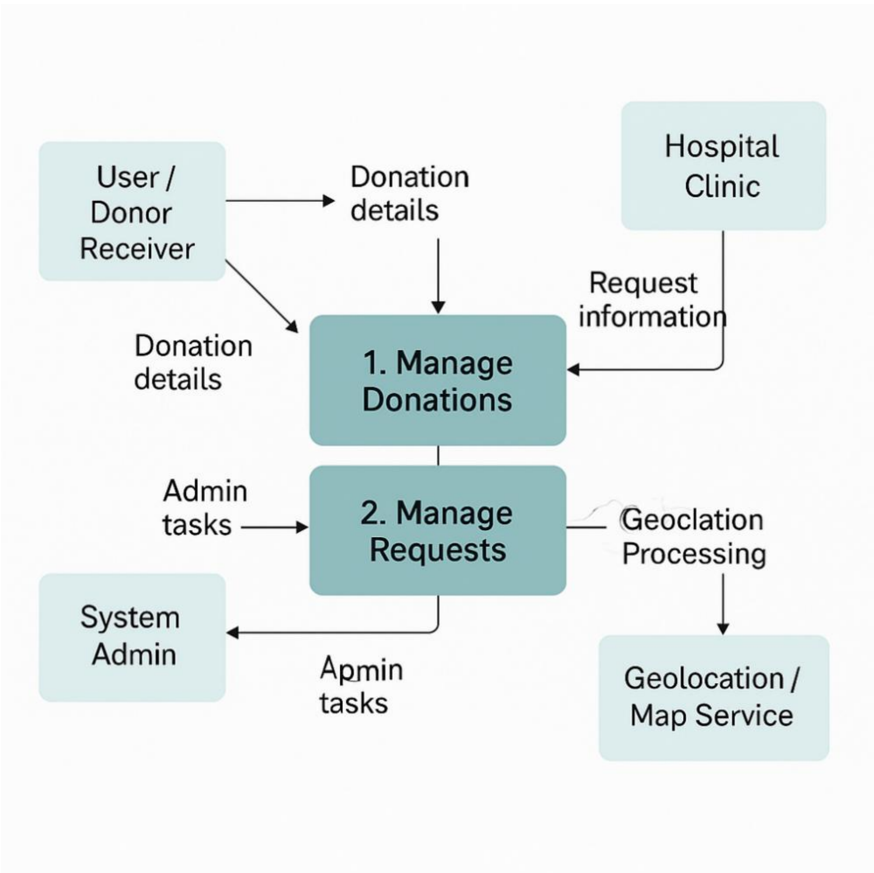


Figure 3.1.2 DFD Level 1

3.1.3 Schema Diagram

Figure 3.1.3 Schema Diagram

3.1.4 Entity Relationship Diagram

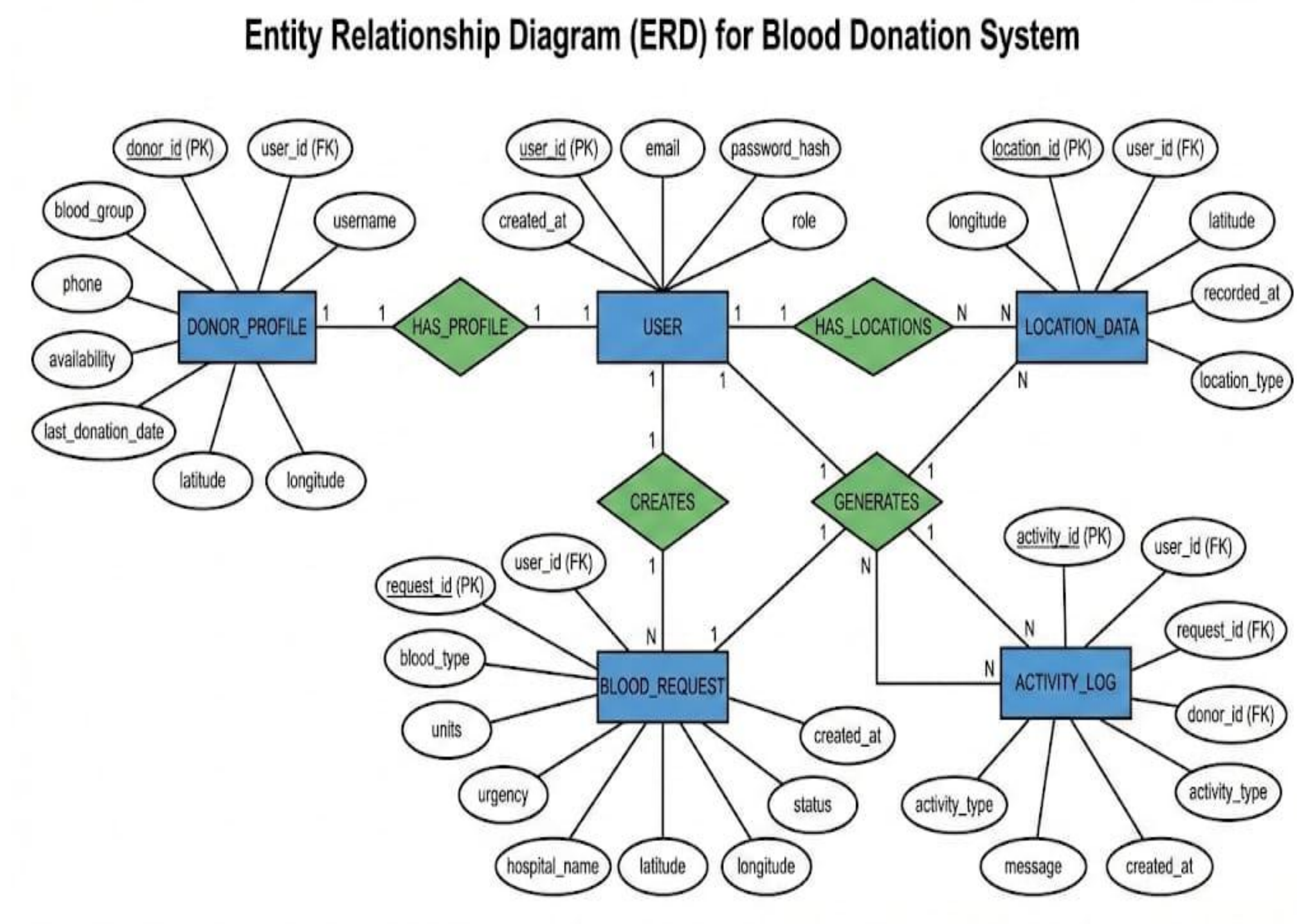


Figure 3.1.4 E R Diagram

3.1.5 Activity Diagram

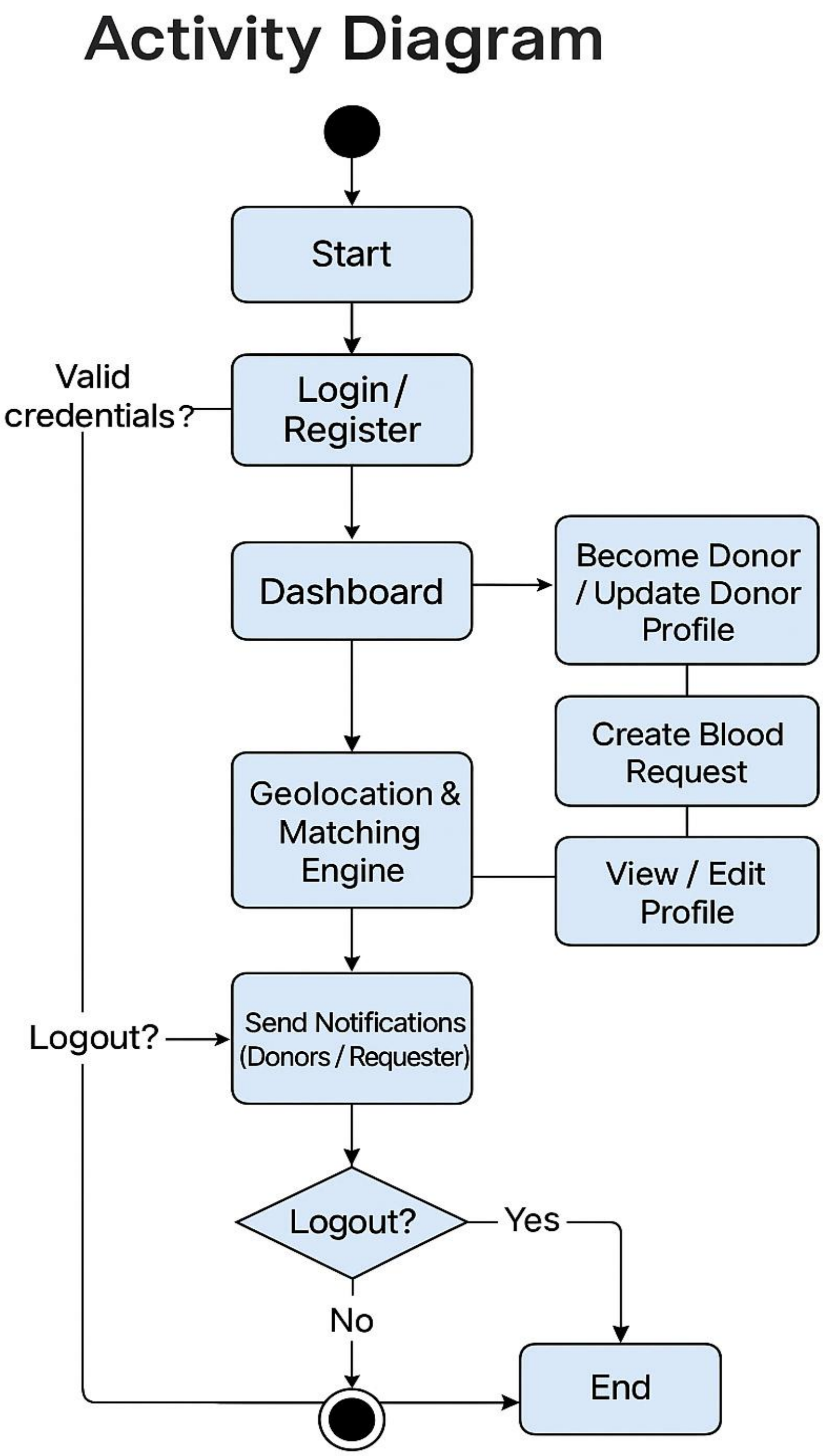


Figure 3.1.5 Activity Diagram

3.1.6 Flow Chart

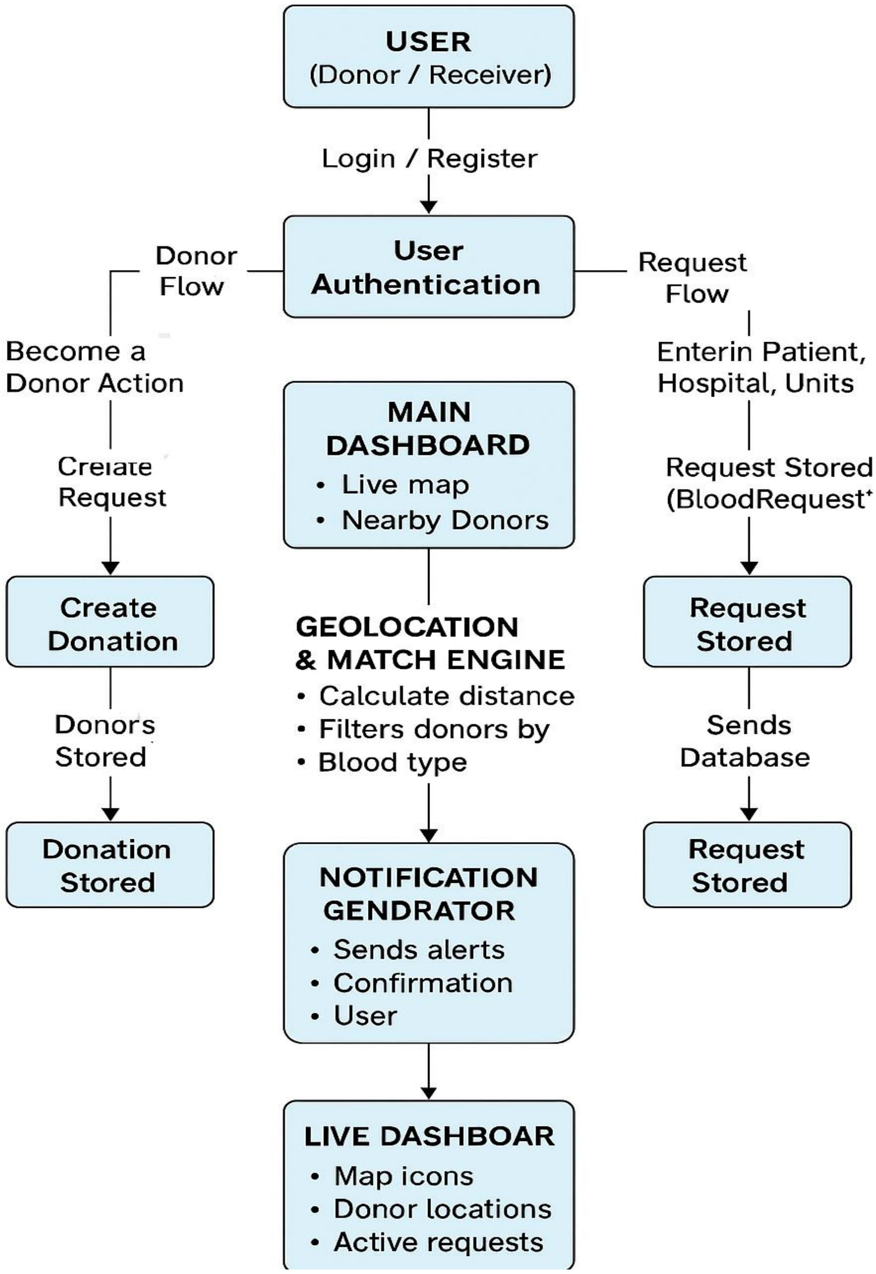


Figure 3.1.6 Flow Chart

3.2 Design of Functions

1. User Registration and Authentication

Users must authenticate before accessing the dashboard. Functions handle account creation, login, logout, password validation, error handling, and successful redirection to the home page.

2. Donor Profile Management

A function enables users to create donor profiles by entering their blood group, contact number, availability status, and last donation date. The system validates data, stores it securely, and updates donor markers on the map in real time.

3. Geolocation Tracking and Storage

The browser captures coordinates through the Geolocation API. Backend functions update donor positions, calculate distances using the Haversine formula, and store coordinates in the database for live map updates.

4. Blood Request Creation

Functions allow users to submit emergency requests by entering key details such as required units, urgency, and hospital address. The system geocodes the location, stores data, and triggers automated matching.

5. Donor–Recipient Matching

Matching functions compare donor blood groups with request requirements and compute proximity. The system ranks donors by distance, availability, and compatibility, enabling quick response during emergencies.

6. Interactive Map Rendering

Map functions integrate Leaflet.js and OpenStreetMap. They display donor and request markers, animate updates, and ensure the user sees real-time results. Popups show donor details, distance, and availability.

7. Dashboard Visualization

The dashboard function generates metrics such as total donors, active requests, average distance, and donor activity. This summary helps users and administrators gain clear insights.

8. Profile Editing and Updating

Users can update their profile, change availability, or modify contact or blood-group information. The system validates fields and synchronizes updated values with the donor map.

9. Notification and Alert System

The system can generate alerts for nearby donors or notify users of updates in request status. This enhances communication efficiency and donor engagement.

10. Optional: Logging and History Tracking

Functions store logs for donor actions, request responses, and historical interactions for future analytics, reporting, and system optimization.

3.3 Design of User Interface

The user interface is designed with clarity, consistency, and simplicity in mind. The interface ensures that users clearly understand each step, from logging in to navigating the live donor map.

1. The interface guides users through logical steps such as logging in, viewing the dashboard, and exploring the map.
2. Elements such as notifications, donor markers, and request details always appear in designated

sections of the screen.

3. All information, including alerts and messages, remains on screen for sufficient time so the user can read and interpret it.
4. Default values such as common blood groups and urgency levels are pre-filled to reduce user effort.
5. Form validation prevents users from submitting incomplete or invalid information during donor registration or request creation.
6. The system avoids technical or backend error messages. Instead, user-friendly alerts convey issues such as invalid form entries or unavailable features.

3.4 Design of Reports

The system generates structured and meaningful reports to support users, administrators, and healthcare personnel. These reports support decision-making and provide clear visibility into system operations.

1. Donor Activity Report

Shows total donors, availability statistics, recent donors, and last donation records.

2. Blood Request Report

Displays all active and closed requests, urgency status, required blood types, and response times.

3. Geolocation Coverage Report

Analyzes donor distribution on the map, identifies high-density and low-density regions, and helps optimize donor recruitment.

4. Distance & Matching Report

Summarizes donor–recipient matching efficiency, average distance calculations, and time taken for donor identification.

5. User Profile Summary Report

Includes user contact information, donor history, request activity, and availability logs.

6. System Utilization Report

Provides insights into login frequency, peak usage hours, request load, and map activity trends.

CHAPTER 4

IMPLEMENTATION

4.1 Modules Implemented

The Geolocation-Enabled Blood Donation and Request Management System consists of multiple integrated modules that work together to deliver real-time donor tracking, blood request management, and interactive map visualization. Each module is developed to ensure modularity, reliability, and ease of scalability. The primary modules implemented in the system are described below.

1. User Authentication Module

This module handles secure user registration and login functionalities.

Key functions include:

- Allowing users to create accounts using a username, email, and password.
- Authenticating users through Django’s in-built authentication framework.
- Ensuring secure password hashing and session management.
- Restricting dashboard access to authenticated users only.
- Redirecting unauthorized users to the login page.

2. Donor Profile Management Module

This module manages the creation and maintenance of donor profiles.

Key functions include:

- Allowing users to register as blood donors by providing blood group, phone number, and availability.
- Storing medical eligibility fields like last donation date.
- Updating availability status dynamically when a donor becomes active or inactive.
- Linking each donor profile with a unique registered user.
- Displaying donor details on the interactive map.

3. Blood Request Management Module

This module processes and stores all emergency blood requests created by recipients.

Key functions include:

- Enabling users to submit new blood requests with required blood group, units needed, urgency level, and hospital location.
- Automatically capturing request coordinates for map visualization.
- Updating the request status based on user actions or donor response.
- Displaying all active requests across the dashboard and map.

4. Geolocation Tracking Module

This module integrates browser geolocation services to track the real-time location of donors and request creators.

Key functions include:

- Capturing latitude and longitude through the Browser Geolocation API.
- Updating location values in the database during each session.
- Applying the Haversine formula to calculate distances between donors and requests.
- Improving response efficiency by identifying nearest donors automatically.

5. Interactive Map Visualization Module

One of the core components of the system, this module displays donors and requests on a live map for visual tracking.

Key functions include:

- Integrating Leaflet.js with OpenStreetMap for real-time map rendering.
- Plotting donor markers with blood group icons and availability colors.
- Plotting request markers with emergency indicators.
- Displaying popups showing donor details, distance, and status.
- Updating map markers dynamically when new data is added.

6. Dashboard Analytics Module

This module provides summarized insights and quick-access statistics for the user.

Key functions include:

- Displaying the total number of active donors.
- Showing the number of open requests.
- Calculating average distance between donors and requests.
- Presenting essential user data such as profile details and recent activity.

7. Profile Management & Editing Module

The module enables users to manage their personal and donor information conveniently.

Key functions include:

- Editing personal details such as name, phone number, and location.
- Updating donor-related data like availability, blood group, and last donation date.
- Ensuring changes reflect instantly on the map and donor listings.

8. Notification and Alerts Module (Optional Enhancement)

For real-time engagement, this module may notify donors of emergencies near their location.

Key functions include:

• Sending in-app or email notifications to donors within a certain radius.

• Alerting recipients when a donor responds or updates status.

4.2 TABLE CREATION

4.2.1 List of Logical Data Structures

Structure Name	Purpose
user	**Stores authentication details**
donor_profile	Stores donor-related medical and location data
blood_request	Represents emergency blood requests
location_data	Stores real-time or last-known user coordinates
activity_log	Logs system actions, updates, and notifications

4.2.2 Donor Profile Structure

Field	Description
user_id	Foreign key linking to User
blood_group	Donor blood type
phone	Contact number
availability	Donor active/inactive status
last_donation	Date of last donation
latitude	Current latitude of donor
longitude	Current longitude of donor

4.2.3 Blood Request Structure

Field	Description
user_id	Person requesting blood
blood_type	Required blood group
units	Number of units needed
urgency	Emergency level

Field	Description
hospital_name	Location of patient
latitude	Hospital’s latitude
longitude	Hospital’s longitude
status	Active/Closed

4.3 DATABASE INTEGRATION

The system uses a relational database (SQLite/MySQL) integrated with Django ORM. The database stores user profiles, donor details, blood requests, and real-time coordinates. Django handles migrations to create and maintain tables automatically while ensuring data consistency.

The system stores structured data that supports donor–recipient matching, map visualization, and historical tracking. Storing data in the backend ensures transparency and helps generate detailed reports.

4.3.1: USER Table

Column	Type	Description
id	INTEGER (PK)	User ID
username	VARCHAR	Login name
email	VARCHAR	Contact email
password	HASH	Secured password
created_at	DATETIME	Registration timestamp

4.3.2: DONOR_PROFILE Table

Column	Type	Description
id	INTEGER (PK)	Donor ID
user_id	INTEGER (FK)	Linked to user.id
blood_group	VARCHAR	Blood type
phone	VARCHAR	Contact number
availability	BOOLEAN	Available or not
last_donation	DATE	Medical eligibility

Column	Type	Description
latitude	FLOAT	Donor location
longitude	FLOAT	Donor location

4.3.3: BLOOD_REQUEST Table

Column	Type	Description
id	INTEGER (PK)	Request ID
user_id	INTEGER (FK)	Created by user
blood_type	VARCHAR	Required blood group
units	INTEGER	Units needed
urgency	VARCHAR	Emergency level
longitude	FLOAT	Requestor location
latitude	FLOAT	Requestor location
status	VARCHAR	Active/Closed
created_at	DATETIME	Request timestamp

CHAPTER 5

TESTING

5.1 TEST CASES AND RESULTS

The system was tested extensively to validate functionality, accuracy, and reliability. The following test cases ensure that donor registration, request creation, geolocation tracking, and dashboard operations work as intended.

Test ID	**Test Case Title**	**Condition**	**Expected Result**	**Observed Result**
T01	Invalid Login Attempt	Enter wrong username/password	Display "Invalid Credentials" error	Passed
T02	Donor Profile Creation	User enters valid donor details	Donor profile successfully saved	Passed
T03	Blood Request Submission	User submits request with required fields	Request displayed under "Active Requests"	Passed
T04	Geolocation Permission Test	User clicks "Allow Location Access"	System captures latitude & longitude	Passed
T05	Map Marker Rendering	Donor/request exists with coordinates	Donor and request markers appear on map	Passed
T06	Distance Calculation	Donor and request located in valid locations	Accurate distance displayed	Passed
T07	Dashboard Metrics Update	Add donor/request	Dashboard counts update in real time	Passed
T08	Profile Update	User edits phone/blood group	Profile updated successfully	Passed
T09	Logout Functionality	User clicks logout	Session terminated and redirected to login	Passed

CHAPTER 6

USER MANUAL

6.1 INSTALLATION PROCEDURE

1. Install Python & Django

The Blood Donation and Request Management System is developed using Python and Django.
• Install Python 3.10 or above from the official website.
• Ensure that Python is added to PATH.
• Install Django using the command:

pip install django

2. Install Required Python Libraries

Open the terminal and navigate to the project directory.

Run:

pip install -r requirements.txt

This installs:
• Django
• Leaflet integration libraries
• geopy (for distance calculations, optional)
• djangorestframework (if API extensions are used)

3. Set Up Django Project Folder

Download or clone the project folder into your local machine.
Ensure all folders such as:
• /templates
• /static
• /donation app folder
• /maps and JS files
are present.

4. Apply Database Migrations

Run the following commands:

python manage.py makemigrations

python manage.py migrate

This creates all tables such as User, DonorProfile, and BloodRequest.

5. Create Superuser (Optional for Admin Access)

python manage.py createsuperuser

Use admin panel to view donor profiles & requests.

6. Run the Application

python manage.py runserver

Open browser and visit:

http://127.0.0.1:8000/

You can now:
• Register an account
• Log in
• Update donor profile
• Create blood requests
• View donors/requests on the live map

7. **Testing the Application**

Perform the following actions to confirm the system is working correctly:
• Log in with registered account
• Allow geolocation access from browser
• Verify donor appears on map
• Create a sample blood request
• Check if markers update on map
• View dashboard metrics
• Edit profile and verify updates

8. **Troubleshooting**

• If server does not start, check package installation.
• If map does not load, ensure internet access for OpenStreetMap tiles.
• If geolocation fails, allow browser location permissions.
• For database errors, re-run migrations.

6.1.1 REQUIREMENTS

Software Requirements

• Python 3.10+
• Django 5.x
• SQLite/MySQL database
• Browser supporting geolocation (Chrome/Edge/Firefox)
• VS Code / PyCharm IDE

Hardware Requirements

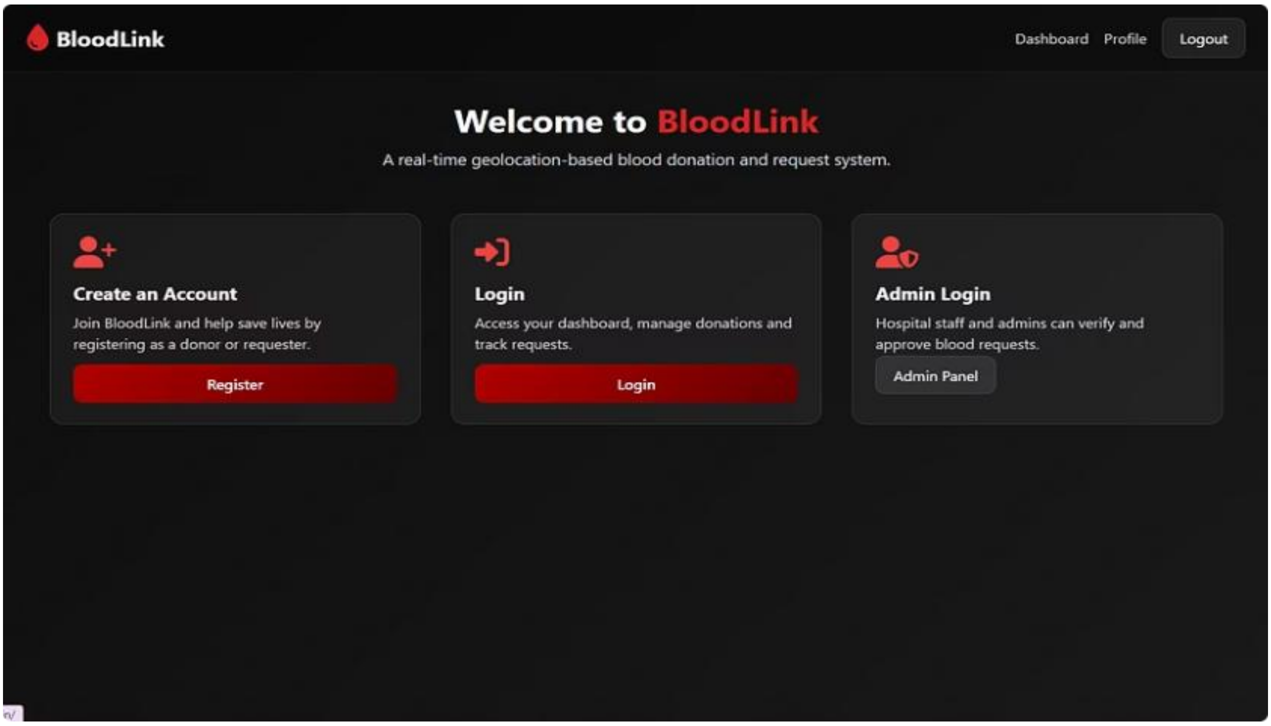
• Processor: Intel Core i3 or higher
• RAM: 4GB or more
• Storage: Minimum 1GB free space
• OS: Windows 10/11 / Linux / macOS

Project Requirements

• Complete Django project source code
• requirements.txt file
• Internet connection for map rendering
• Browser with Leaflet.js and JS enabled

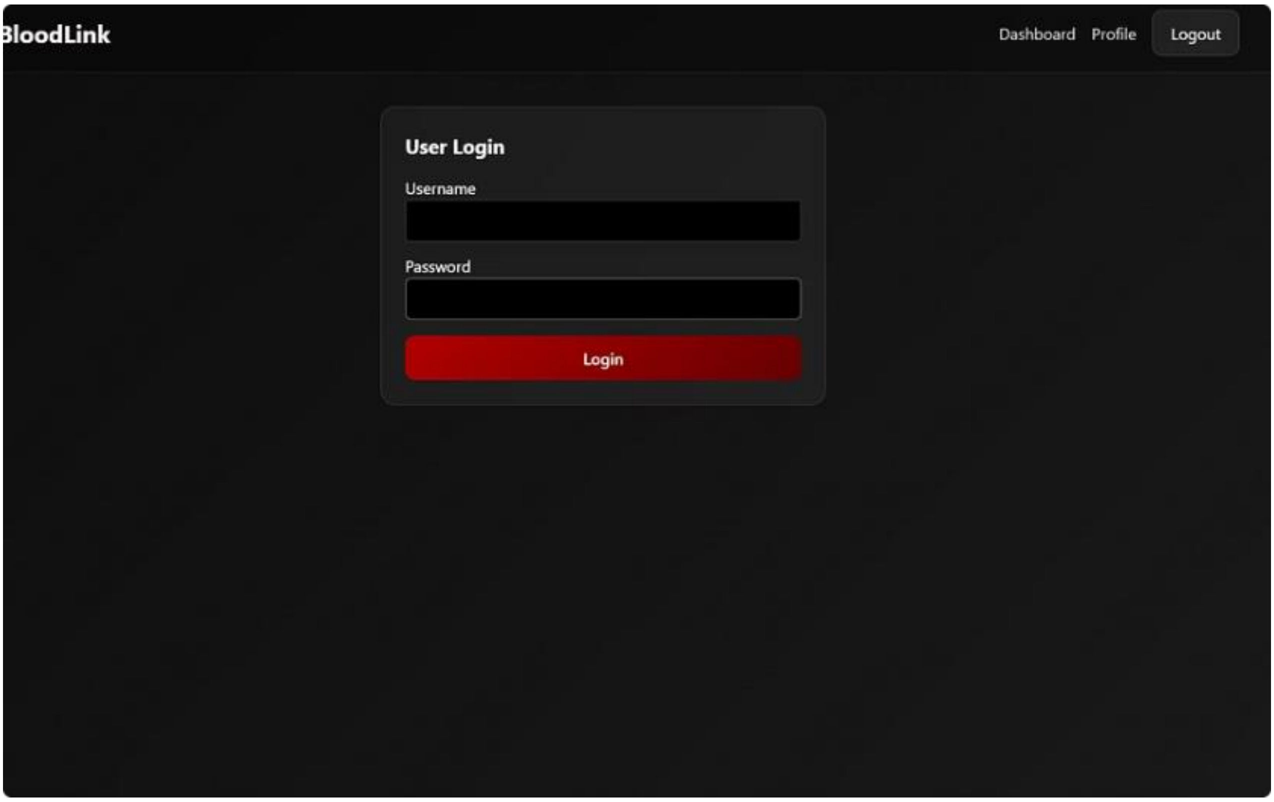
6.2 SNAPSHOTS OF THE INTERFACE

6.2.1 Home Page



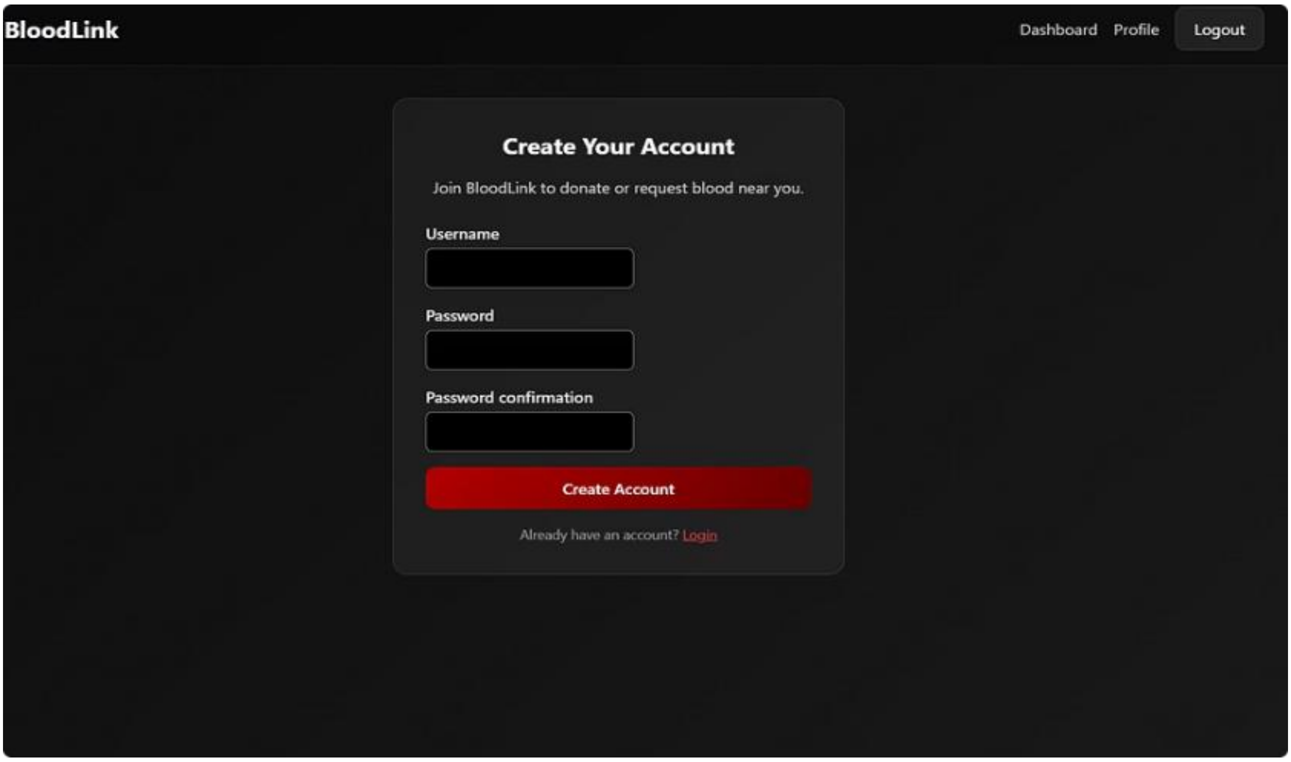
The initial entry point, welcoming users and providing immediate access to core functionalities.

6.2.2 Login Page



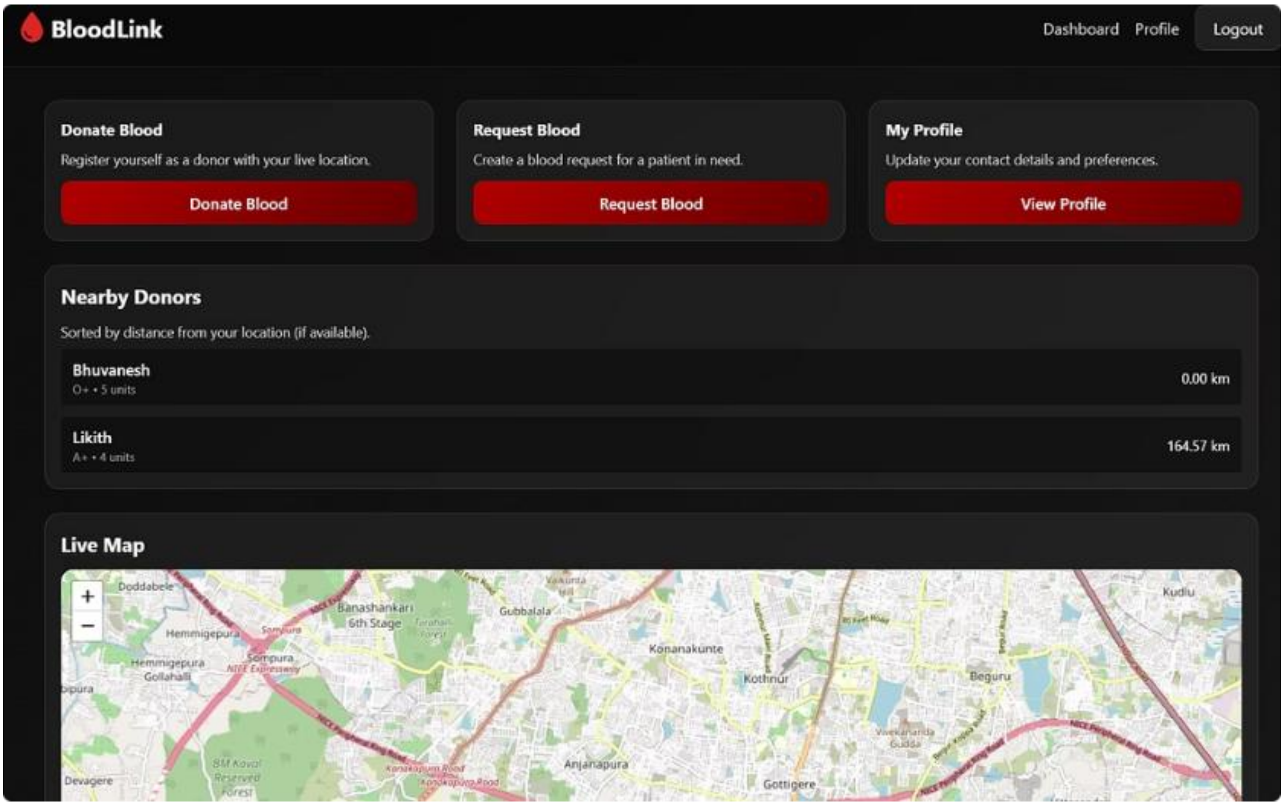
Secure authentication for existing users, ensuring personalized and safe access to their profiles.

6.2.3 Registration Page



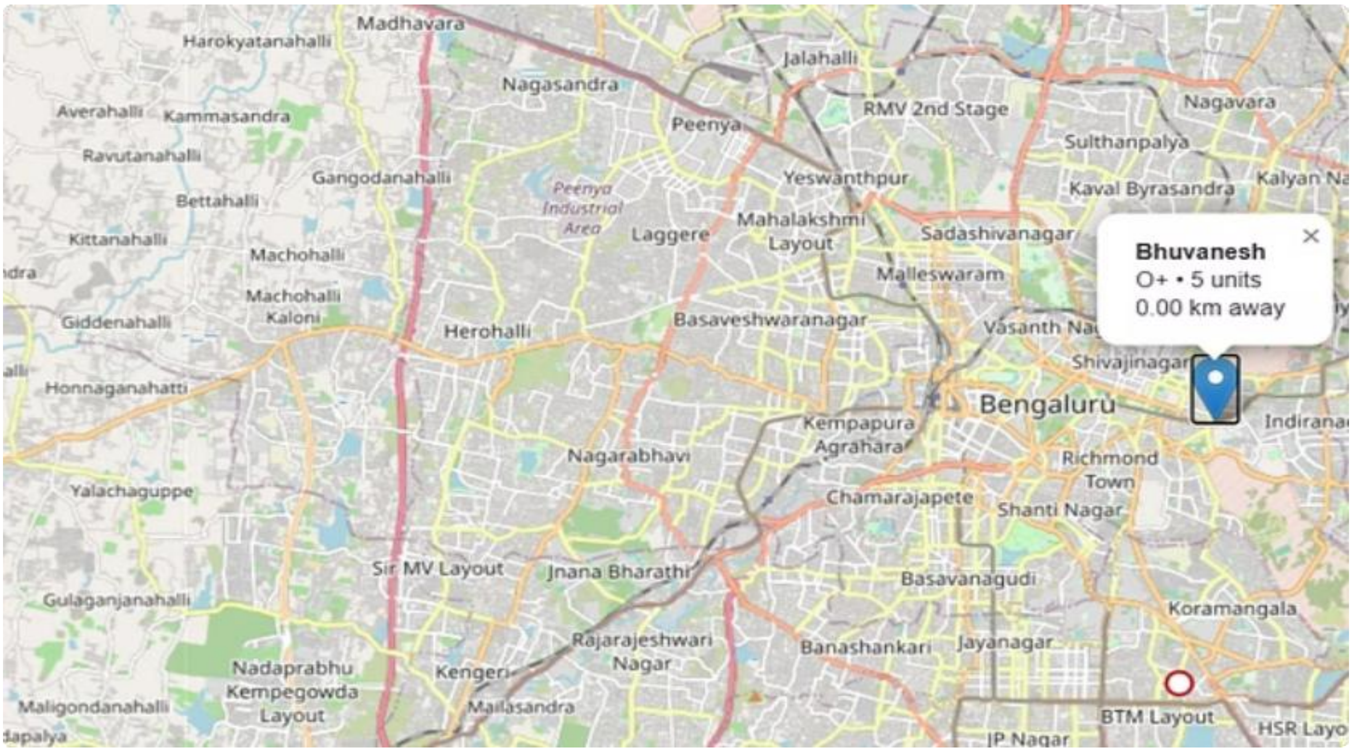
A simple and guided process for new users to create an account and join the BloodLink community.

6.2.4 Dashboard



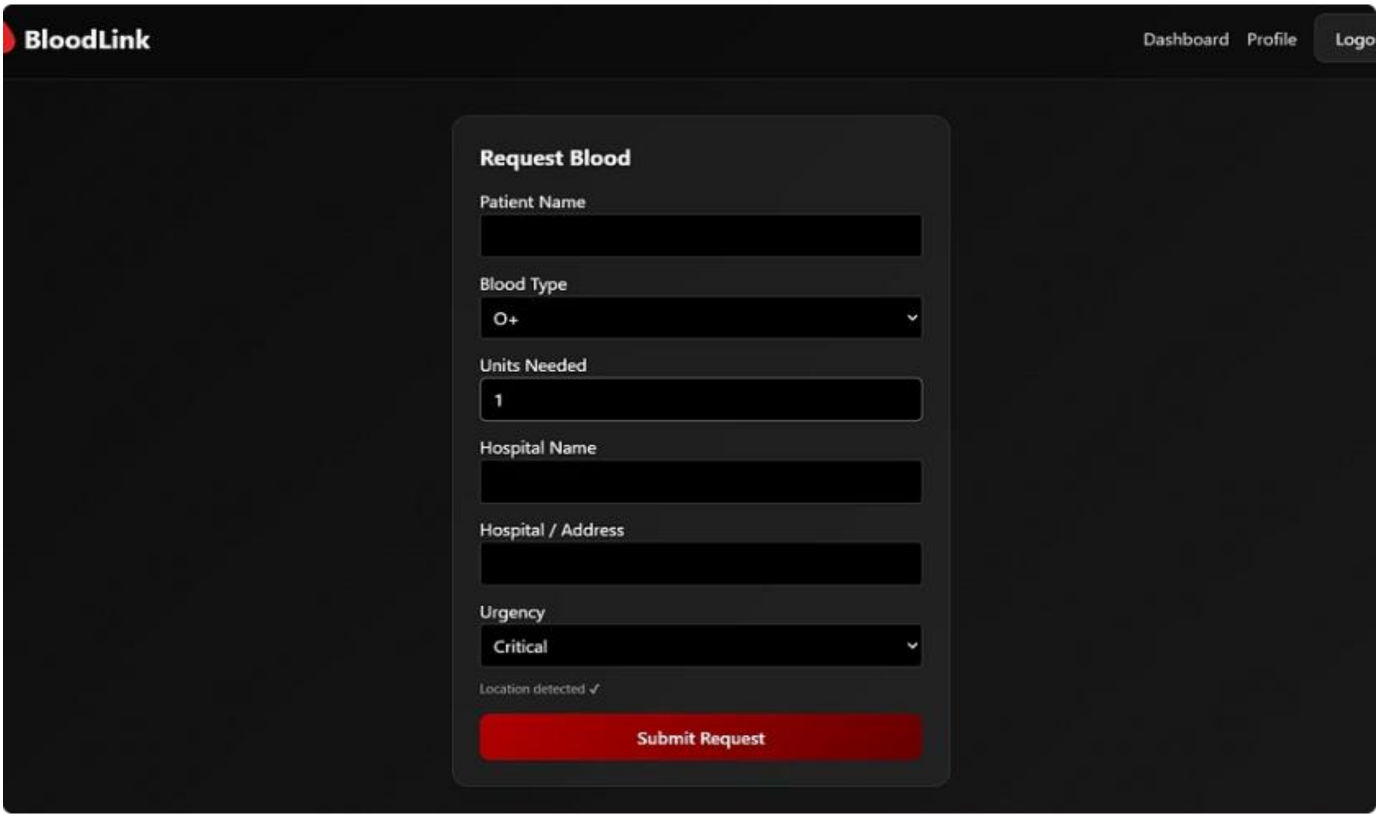
Personalized overview of user activity, including donation history, urgent requests, and quick actions.

6.2.5 Live Donor Map



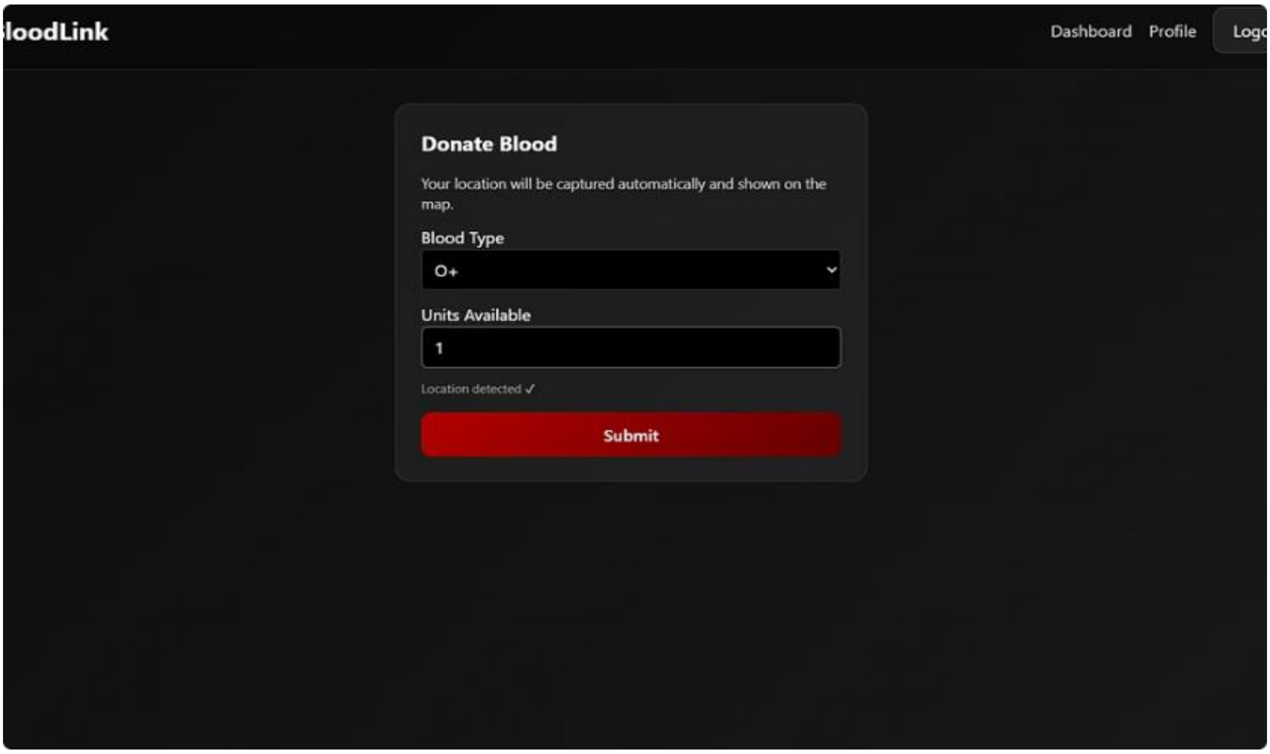
Interactive map visualizing real-time donor availability, blood bank locations, and urgent request areas.

6.2.6 Blood Request Form



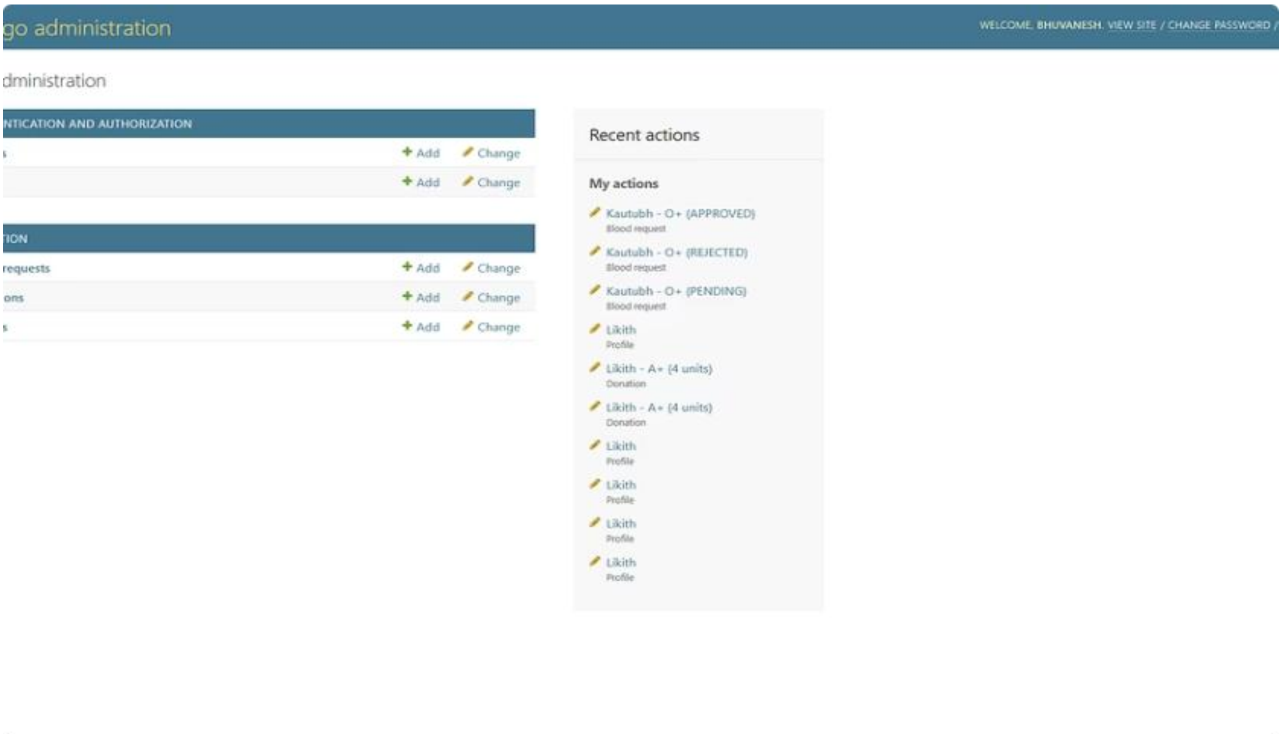
An intuitive interface for users to submit blood requests, clearly specifying their needs and location.

6.2.7 Donate Blood Form



Streamlined process to schedule a blood donation, providing clear steps and eligibility information.

6.2.8 Admin Panel



Comprehensive tools for administrators to manage users, approve requests, and monitor system health.

CHAPTER 7

CONCLUSION

The Geolocation-Enabled Blood Donation and Request Management System demonstrates how modern web technologies and location-based services can significantly improve the speed and reliability of emergency blood coordination by replacing manual, outdated methods with a centralized, automated digital platform. By integrating a Django-powered backend with real-time geolocation tracking and interactive mapping, the system enables donors to register, update availability, and share their live location while allowing recipients to instantly create requests that appear on a dynamic map, making it easier to find the nearest suitable donor. Through proximity-based matching, structured workflows, and an intuitive user dashboard, the platform ensures accurate information flow, faster response times, and greater transparency. Overall, the system streamlines the entire blood donation process, enhances user experience, and serves as an effective solution that can help save lives by reducing delays during medical emergencies.

7.1 FUTURE SCOPE

The system has strong potential for expansion, and several enhancements can be implemented to increase its impact and usability in the future.

One promising improvement is the integration of a hospital-side dashboard, enabling hospitals to directly manage donor interactions, track blood requests, and maintain real-time coordination with nearby donors. Similarly, implementing SMS or WhatsApp alert systems can instantly notify donors during emergencies, making the platform more responsive.

Artificial Intelligence can also be incorporated to predict donor availability, analyze donation trends, and recommend the nearest best donor based on compatibility, urgency, and travel feasibility. A dedicated mobile application would further enhance accessibility by offering real-time push notifications, background geolocation updates, and faster donor–recipient communication.

Additional enhancements may include linking the platform with government blood bank APIs to provide accurate blood inventory information, allowing users to check availability directly from hospitals. Multi-language support can increase inclusivity, enabling users from different regions to interact with the system comfortably. Finally, cloud-based storage and analytics modules can offer long-term data insights, improving decision-making for volunteers, hospitals, and emergency response teams.

With these advancements, the system has the potential to evolve into a comprehensive, nationwide blood donation and emergency response platform that strengthens healthcare support and saves more lives.

REFERENCES

1. Priya, S., Kumar, R., & Meenakshi, A. (2023). *A Survey on Digital Blood Donor Management Systems Using Web and Mobile Technologies*. International Journal of Health Informatics Research, 18(2), 45–52.
2. Harish, N., Varma, P., & Srinivas, M. (2024). *Emergency Response Enhancement Using Geolocation and Cloud-Based Health Platforms*. Journal of Emerging Computing and Applied Systems, 12(3), 112–120.
3. Thakur, K., & Rao, M. (2023). *Smart Blood Bank System Using IoT and Automated Data Processing*. International Journal of Medical Informatics and Smart Healthcare, 9(1), 66–73.
4. Fernandez, A., Joseph, B., & Thomas, R. (2022). *Mobile-Based Blood Donation Alert and Tracking System for Emergency Healthcare*. International Journal of Mobile Computing and Bioinformatics, 7(4), 89–98.
5. Django Documentation. (2024). *Django Web Framework — Official Documentation*. Retrieved from https://docs.djangoproject.com/
6. Leaflet.js Team. (2024). *Leaflet Maps – Open-Source JavaScript Mapping Library*. Retrieved from https://leafletjs.com/
7. OpenStreetMap Foundation. (2024). *OpenStreetMap — Collaborative Mapping Project*. Retrieved from https://www.openstreetmap.org/
8. Waze, Y., & Goldberg, R. (2021). *Haversine Formula and Distance Computation for Geolocation-Based Applications*. Journal of Geographic Information Systems, 14(2), 210–218.
9. Browning, S. (2023). *Designing Interactive Dashboards for Emergency Response Systems*. Proceedings of the International Conference on Human–Computer Interaction, 4(2), 55–63.
10. Sharma, V. (2022). *Web-Based Emergency Medical Support Systems: A Comprehensive Review*. International Journal of Computer Applications in Healthcare, 20(3), 131–145.
11. Python Software Foundation. (2024). *Python 3 Documentation*. Retrieved from https://www.python.org/doc/
12. HTML Living Standard. (2024). *WHATWG HTML Specification*. Retrieved from https://html.spec.whatwg.org/
13. Tailwind Labs. (2024). *TailwindCSS — Utility-First CSS Framework*. Retrieved from https://tailwindcss.com/
14. Mozilla Developer Network (MDN). (2024). *Geolocation API — Web APIs Documentation*. Retrieved from https://developer.mozilla.org/
15. Kumar, A., & Ranganath, S. (2022). *Analysis of Real-Time Donor–Recipient Matching Algorithms*. International Journal of Soft Computing and Intelligent Systems, 15(1), 74–81.