WWW.IJRPR.COM

International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

BloodBridge: A Real-Time Blood Donation and Request Matching System Using Django

Kavya Vemuri[1], Mr. Raman K Rajagopalan[2]

[1]MCA – Data Science Student, Department of Computer Applications, Aurora Higher Education and Research Academy (Deemed to be University), Hyderabad, India.

[2]Associate Professor, School of Informatics, Aurora Higher Education and Research Academy (Deemed to be University), Hyderabad, India.

ABSTRACT:

Blood shortages during emergencies can lead to significant delays in providing timely patient care. While many existing platforms are capable of storing donor information and managing blood inventory, they often lack essential features such as real-time matching between donors and patients, health eligibility checks, awareness of geographic locations, and immediate alerts. This paper introduces BloodBridge, a web-based system developed using Django and MySQL that facilitates real-time connections between hospitals, patients, and donors. BloodBridge enables the matching of donors to blood requests based on blood group compatibility and proximity. It also verifies donor eligibility through a simple check that includes age, last donation date, and basic health indicators. During urgent situations, the system sends SMS or email alerts to notify relevant parties. The system is designed with a modular architecture that includes components for user management, request intake, matching services, notification services, and audit and analytics. A mobile-friendly user interface, built using HTML, CSS, and Bootstrap, enhances accessibility. The paper also outlines an evaluation plan that focuses on key performance metrics such as time-to-match, notification latency, accuracy and completeness of matches, and user satisfaction scores. By integrating matching logic, alert systems, and location-based filtering within a secure, role-based environment, BloodBridge seeks to minimize response times, improve coordination among healthcare providers and donors, and build a more dependable blood donation network. Although the implementation is still in progress, the proposed solution addresses important shortcomings in current systems and offers a feasible approach toward achieving quicker, safer, and more transparent emergency blood donation processes.

Keywords: Blood donation; real-time matching; Django; emergency alerts; health eligibility; GPS/location; online blood bank.

1. Introduction:

Millions of lives are saved annually by blood transfusions during surgery, trauma care, cancer therapy, and life-threatening illnesses, making it a vital component of healthcare. However, finding compatible blood during emergencies is still a problem worldwide. we can see delays in finding a suitable donor frequently lead to serious consequences, including mortality in some cases. Traditional methods such as paper-based records, manual communication, and maintaining offline donor registries are ineffective, error-prone, and unable to address patients' pressing needs (World Health Organization, 2022). These restrictions emphasize the need for digital platforms that may increase the efficiency and dependability of blood donation procedures.

By creating online blood donation platforms, several researchers have tried to solve this problem. For example, Abdul-Gafaru et al. (2024) suggested an online platform for managing blood requests and registering donors, but it did not include geographic search or real-time notifications. characteristics. Gayathri et al. (2024) unveiled a compatibility checker that increased the precision of the match, but it lacked immediate alerts or donor health eligibility checks. In a similar vein, Tarun et al. (2021) created a blood bank management system that made it easier for donors and recipients to connect but lacked location awareness and emergency notifications. Additional contributions, such those by Sankareshwari et al. (2022) and Manika et al. (2023), emphasized web analytics and design but lacked full functionality. such as a donor search with GPS and health verification.

This review makes it plain that although current methods offer partial answers to the urgent requirements of emergency blood donation, they don't completely satisfy them. The majority lack real-time communication tools, automated donor-patient matching, and eligibility verification. Furthermore, their usefulness in emergency situations is diminished by their limited mobile accessibility and broken designs.

This article suggests BloodBridge, an online application created with Django and MySQL, as a solution to these problems. The platform offers a single solution that links patients, donors, and hospitals in real time. Automated donor-patient matching according to blood type and location, straightforward health eligibility checks, and rapid SMS and email alerts are among the essential features. To ensure accessibility across devices, an interactive interface using HTML, CSS, and Bootstrap was created. By integrating matching logic, alert mechanisms, and location-based search into a secure, BloodBridge seeks to minimize delays, enhance coordination, and build a reliable donor base for emergency medical care through role-based

architecture.

2. Literature Review

Numerous studies have attempted to enhance the effectiveness and dependability of blood donation and management systems. Abdul-Gafaru et al. (2024) have made a noteworthy contribution by concentrating on creating an online platform for blood donation. The web interface facilitated donor registration and inquiries, but lacked crucial functionalities such as real-time communication, location-based donor coordination, and emergency alerts.
With a blood type compatibility tester, Gayathri et al. (2024) suggested an integrated blood bank management system. The system reduces manual mistakes during donor provider adjustments and increases accuracy by making sure that blood type compatibility guidelines are applied automatically. Nevertheless, the platform lacked GPS-based donor searches, immediate alerts, and donor health features. This decreased the efficiency of emergency services. Using PHP and MySQL, Tarun et al. (2021) created a blood bank management system in a similar manner. The platform had features like donor registration, queries, and administrative clearance. Despite the fact that we streamlined data administration and record keeping, the system lacked features like real-time donor coordination, automated health testing, and alerts.

The front-end web design for online blood donor management was the subject of another study by Sankareshwari et al. (2022). Using HTML, CSS, JavaScript, and bootstraps, the author developed a response-first, user-friendly interface that enhances usability. Nevertheless, the emphasis on design resulted in a restricted functional depth in the absence of regulations, warning systems, or location-related filtering to assess the donor's health.
Using a more advanced methodology, Manika et al. (2023) created a Bloodbank management system based on Django that integrates backend analytics utilizing technologies like Pandas and Matplotlib. Although the system improved database management and analytical skills, it was unable to verify real-time emergency warnings, GPS-based donor searches, or health clearances.
These research demonstrate that the majority of current systems only offer partial solutions. They handle donor registration, compatibility testing, and data management, but they are not a complete feature that is particularly useful in emergencies. Common gaps include the absence of GPS-based site search, automated health certification screening, immediate warnings, and real-time donor matching. These restrictions underscore the necessity of a consistent, scalable, and real-time system. Many current blood donation systems rely on manual procedures like phoning or emailing people and searching hospital records, which can frequently lead to delays. In contrast, the suggested BloodBridge system uses real-time donor matching, automates eligibility verification, and sends immediate notifications to reduce delays. Figure 1 shows a conceptual comparison between current systems and the suggested BloodBridge platform.

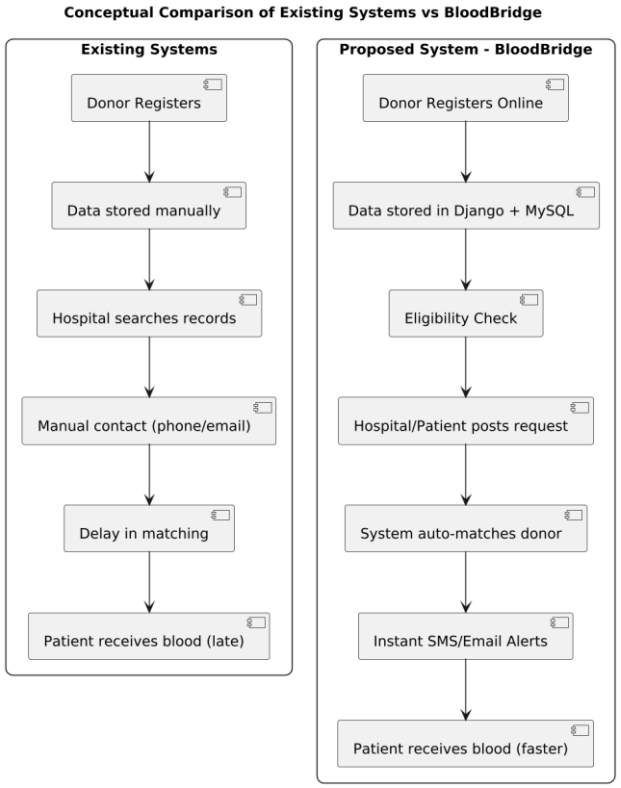

Figure 1. Conceptual comparison of existing systems and the proposed BloodBridge system

Figure 1 shows a conceptual comparison between existing blood donation systems and the proposed BloodBridge platform. While current systems rely heavily on manual processes, lack real-time donor–recipient matching, and depend on delayed communication, the proposed system automates eligibility checks, matches donors instantly, and triggers real-time alerts to ensure timely response.

3. Methodology

The development of the proposed system, BloodBridge, adopts a structured methodology to ensure scalability, reliability, and real-time performance. The methodology focuses on designing a web-based platform that automates donor–recipient matching, incorporates health eligibility checks, and provides instant alerts during emergencies.

3.1 System Architecture

The system follows a three-tier architecture consisting of:

- **Presentation Layer –** A responsive web interface that allows interaction for donors, patients, and hospitals.
- **Application Layer –** Django framework handles business logic, including blood request processing, donor–recipient matching, eligibility verification, and notification services.
- **Database Layer –** MySQL stores donor details, hospital information, patient requests, donation history, and system logs.

The BloodBridge platform is designed using a three-tier architecture, which separates the presentation, application, and database layers to ensure scalability, modularity, and secure data management. The overall system architecture of the proposed platform is depicted in Figure 2.

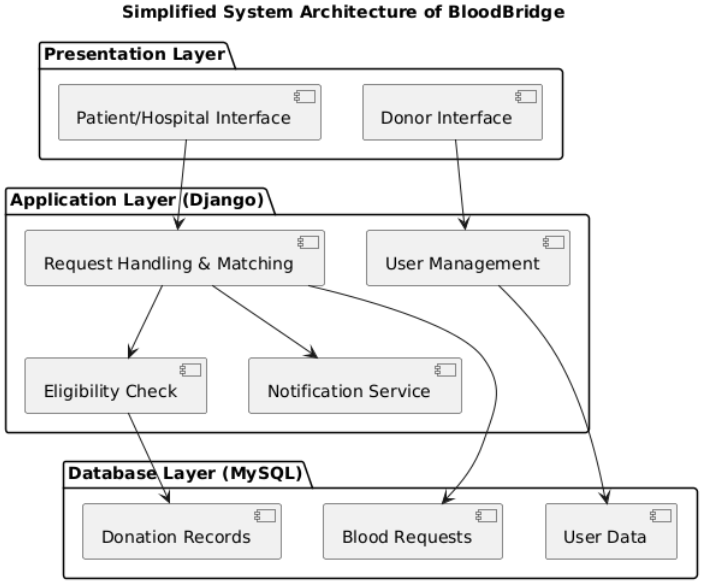

Figure 2. Simplified System Architecture of the Proposed BloodBridge Platform.

From Figure 2, the architecture consists of: (i) a Presentation Layer providing interfaces for donors, patients, and hospitals; (ii) an Application Layer built on Django for request handling, donor–patient matching, eligibility checking, and notifications and (iii) a Database Layer implemented in MySQL to store donor details, hospital information, patient requests, and system logs.

3.2 Workflow of the System

The workflow of the system can be described in the following steps:

- Donors, patients, and hospitals register on the platform.
- Hospitals or patients raise a blood request with details such as blood group, urgency, and location.
- The system verifies donor eligibility using parameters like age, last donation date, and health status.
- The matching engine identifies compatible donors based on blood group and geographic proximity.
- The notification service sends real-time SMS/Email alerts to eligible donors.
- Hospitals confirm the donor and proceed with the donation process.

The proposed system automates the entire process of donor–recipient matching, starting from donor registration to emergency notification. The step-by-step workflow of the BloodBridge system is illustrated in Figure 3.

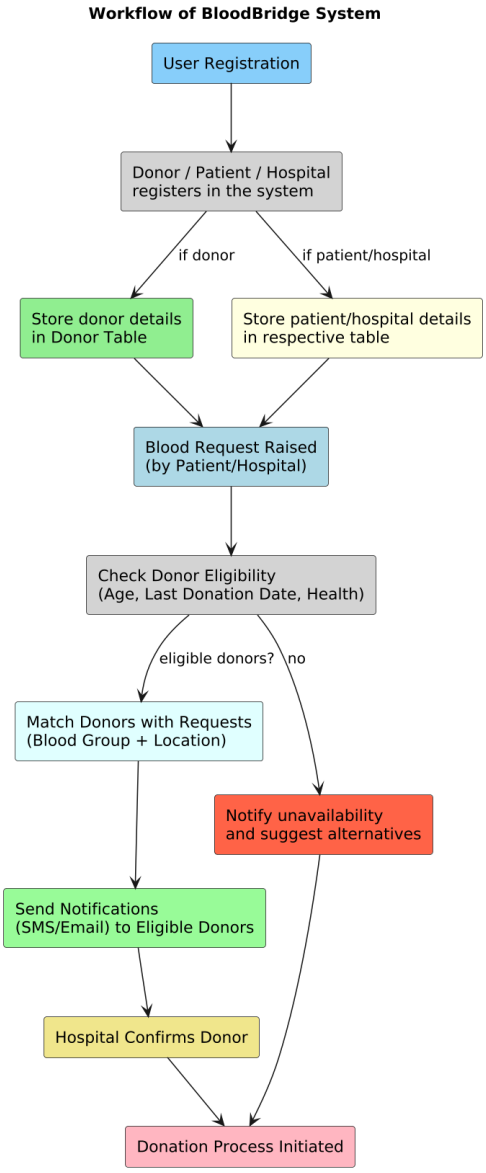

Figure 3. Workflow of the Proposed BloodBridge System.

As shown in Figure 3, the workflow begins with donor registration and health eligibility verification. Hospitals or patients then post requests, which are automatically matched with eligible donors. Once a match is found, instant SMS/Email alerts are triggered to notify donors, ensuring faster response and timely blood availability.

3.3 Functional Modules

The system is divided into several functional modules:

- User Management Module – Registration, login, and role-based access for donors, hospitals, and patients.
- Blood Request Module – Allows patients and hospitals to create and manage blood requests.
- Eligibility Verification Module – Screens donors based on age, health, and last donation date.
- Matching Engine – Performs automated donor–recipient matching in real time.
- Notification Module – Sends SMS/Email alerts during urgent requests.
- Analytics Module – Generates reports on donation history, donor activity, and request fulfillment rate.

3.4 Technology Stack

- Backend: Django Framework (Python)
- Database: MySQL

- Frontend: HTML, CSS, Bootstrap (for responsive design)
- Notification Services: SMS and Email API integration
- Deployment: Web-based application, accessible via browser and mobile devices

4. Implementation

The implementation of the proposed BloodBridge system focuses on transforming the conceptual design into a working solution using Django and MySQL. Unlike the methodology, which presented the overall system architecture and workflow, this section highlights the practical realization of the system through database creation, modular development, user interfaces, and integration.

4.1 Database Implementation

The backend database is implemented in MySQL, which stores and manages information about donors, patients, hospitals, and requests. The database schema is designed for scalability and consistency. The key entities are:

- Donor Table – Stores donor details such as donor ID, name, age, gender, blood group, contact number, location, and last donation date.
- Patient/Request Table – Contains patient information and details of raised blood requests, including required blood group, urgency, and request date.
- Hospital Table – Stores hospital records such as hospital ID, name, address, and contact information.
- Request–Donor Mapping – Maintains the relationship between requests and matched donors for tracking donation history.

The database design of BloodBridge ensures proper relationships between entities such as donors, patients, hospitals, and blood requests. The Entity–Relationship (ER) diagram of the proposed system is shown in Figure 4.

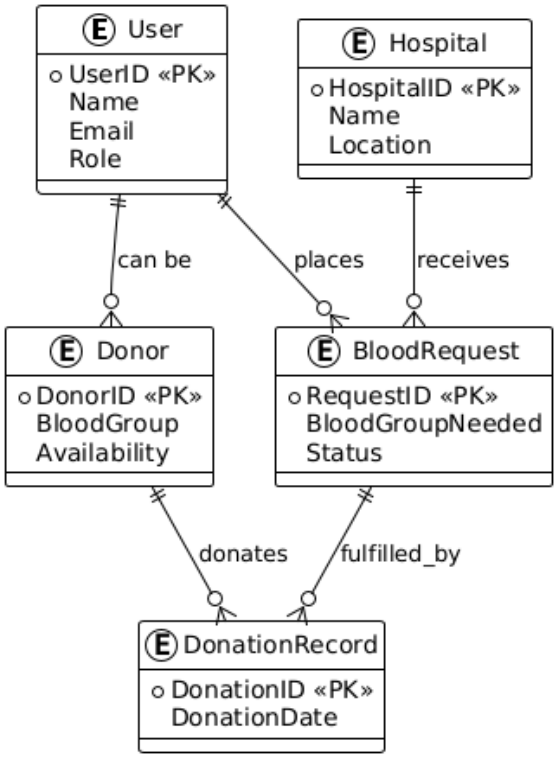

Figure 4. Entity–Relationship (ER) diagram of the BloodBridge system

As shown in Figure 4, the system consists of key entities including User, Donor, Hospital, BloodRequest, and DonationRecord. The relationships among these entities ensure accurate tracking of donor availability, hospital requests, and donation records, supporting reliable and scalable database management.

4.2 Module Implementation

The system is divided into multiple modules, each implemented in Django to ensure modularity and maintainability:

1. **User Management**
   Handles donor, patient, and hospital registrations using Django authentication and role-based access.
2. **Blood Request Module**
   Enables patients and hospitals to raise new requests through forms. Data is stored in MySQL for processing.
3. **Eligibility Verification Module**
   Checks donor age, last donation date, and health conditions before confirming eligibility.
4. **Matching Engine**
   Uses Django ORM queries to identify compatible donors based on blood group and geographic proximity.
5. **Notification Service**
   Sends real-time SMS/Email alerts to eligible donors when a request is raised.
6. **Reports and Analytics**
   Provides hospitals with statistics such as donor activity, fulfilled requests, and donation frequency.

4.3 User Interface Implementation

HTML, CSS, and Bootstrap are used in the frontend design, and Django templates are integrated. Interfaces are created to guarantee a seamless user experience:

- Donor Registration Page – Donors can register, update their information, and check their eligibility status on the donor registration page.
- Blood Request Form – Allows hospitals and patients to submit blood requests along with location and urgency information.
- Hospital/Admin Dashboard – Hospitals can manage donor lists, verify requests, and track donation history with the help of the Hospital/Admin Dashboard.

4.4 System Integration

Django's Model-View-Template (MVT) architecture is used to integrate the database and system modules.

- The donor, patient, hospital, and request entities that are kept in MySQL are represented by models.
- Views manage notifications, donor matching, and user requests.
- Bootstrap is used by templates to create responsive web pages.
- For real-time email and SMS notifications, external APIs are utilized.

The objective of real-time donor-recipient matching is supported by this integration, which guarantees seamless communication between the frontend, backend, and database.

5.Results

The proposed Bloodbridge system is designed to address the limitations of existing online blood donation platforms. The system is still under development, but the expected results are as follows:

- **Real-time donor and patient matching:** A proper engine is expected to significantly reduce the time required to identify compatible donors in emergencies compared to manual search processes.
- **Health Eligiblity Test:** By including donor age, last day of donation and health status, the system aims to improve the safety and reliability of blood transfusions.
- **Immediate Warnings and Notifications:** SMS and Email Notifications are expected to ensure that you minimize communication delays and ensure that you contact legitimate donors immediately for emergency enquiries.
- **Location-based search:** The integration of location-based donor searches is expected to improve the likelihood of finding nearby donors and thus reduce the time of blood availability.
- **Role-based access:** Patients, hospitals and donors interact with the system through safe and different interfaces to improve user friendliness and data management.
- **Mobile - Friendly Design:** The interface is responsive, allowing the system to be accessed across the device. This means that it is more reliable for users even in emergency situations.

The prototype interfaces of the BloodBridge system are shown in Figures 5–11. The homepage provides easy navigation with an awareness banner, while the services page highlights core features like donor search and emergency requests. The donor registration page collects details such as blood group, location, and last donation date, and the login page ensures secure access for registered users. The blood request form allows patients or hospitals to submit requests with location mapping, and the Contact Us page enables users to send queries or feedback. Finally, the donor awareness and registration page educates users about blood compatibility while recording donor details. These interfaces together demonstrate the system's usability and effectiveness for real-time blood donation management.

Figure 5. Home page of BloodBridge with navigation and awareness banner.

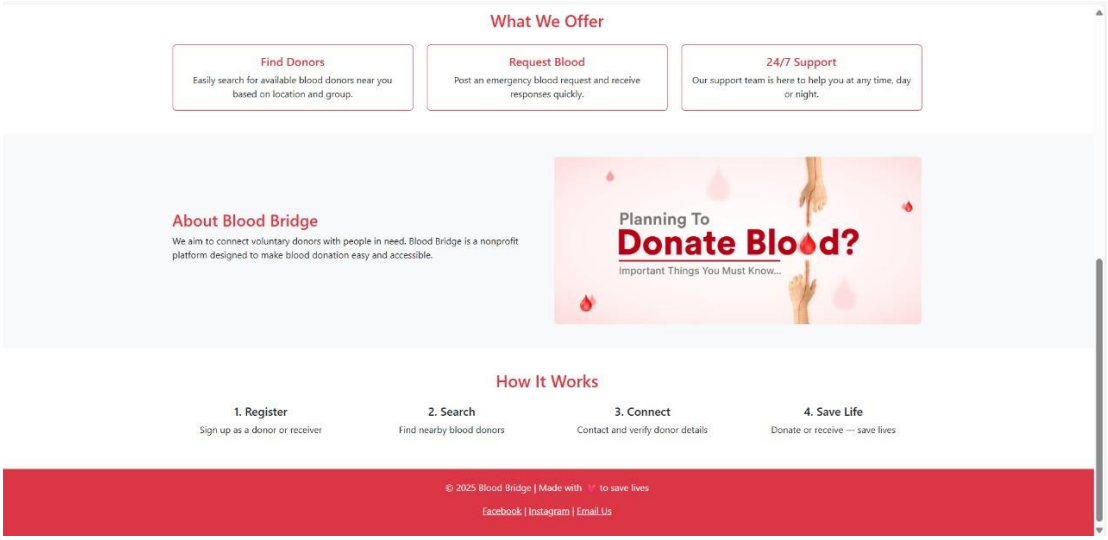

Figure 6. Services and workflow page of BloodBridge.

Figure 7. Donor registration interface for structured data collection.

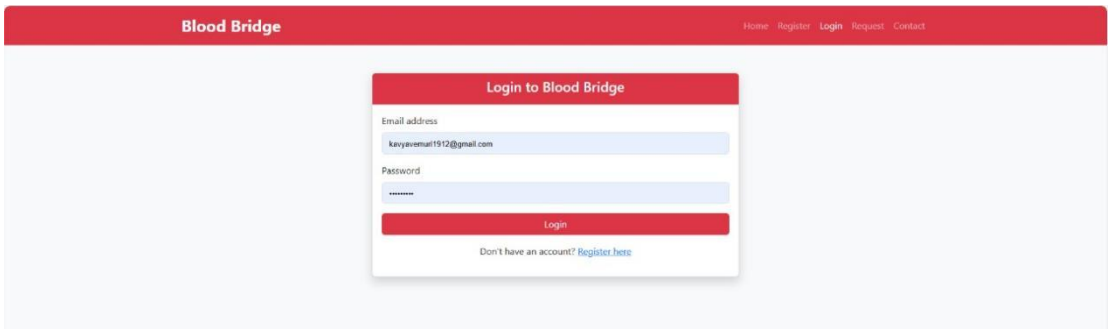

Figure 8. Login interface of the BloodBridge system

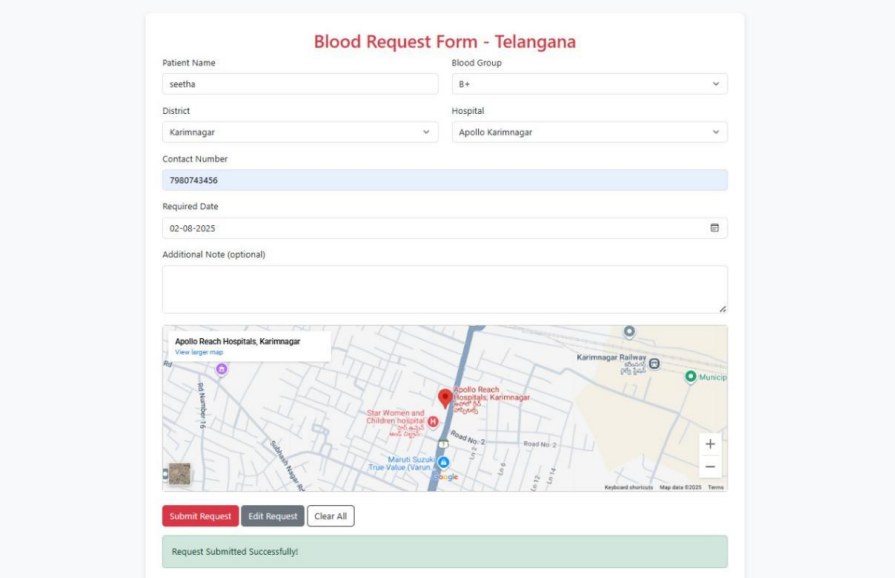

Figure 9. Blood request form of the BloodBridge system with integrated hospital location

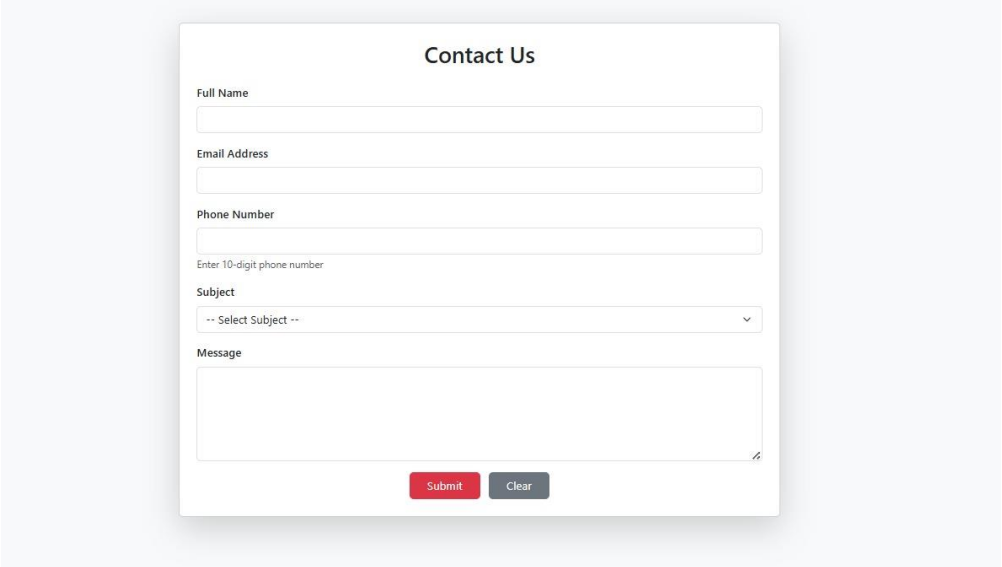

Figure 10. Contact Us interface of the BloodBridge system

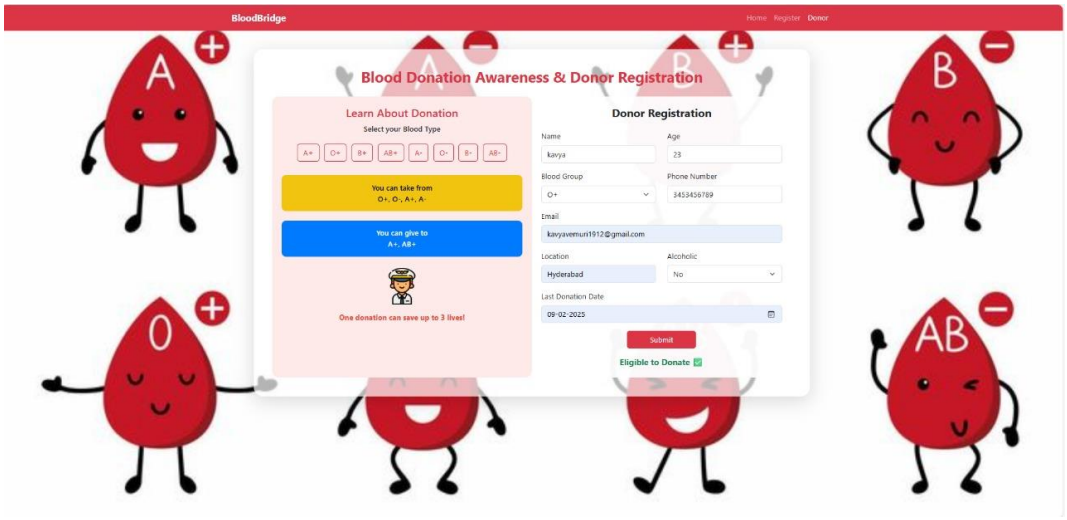

Figure 11. Donor registration and awareness interface of the BloodBridge system

Conclusion and Future

The proposed BloodBridge system aims to address the limitations of existing online blood donation platforms by providing a comprehensive, real-time, and reliable solution for emergency blood requests. Unlike conventional systems that are restricted to donor registration and inventory management, BloodBridge integrates donor–patient matching, eligibility verification, location-based search, and instant alerts into a unified framework.

The system is designed with a modular three-tier architecture using Django and MySQL, ensuring scalability, maintainability, and security. The adoption of a mobile-friendly interface built with HTML, CSS, and Bootstrap further enhances accessibility, allowing patients, hospitals, and donors to interact seamlessly across devices. Once implemented, the platform is expected to reduce response times, improve donor safety, and strengthen the overall reliability of emergency healthcare services.

From the literature survey, it is evident that existing systems provide only partial solutions, often lacking features such as automated health checks, location awareness, and real-time communication. BloodBridge is proposed as a more complete and efficient alternative, capable of minimizing delays and improving coordination among stakeholders during emergencies.

However, the current work is limited to the proposed design and partial implementation stages. Large-scale deployment and integration with national blood bank systems are yet to be achieved. The platform also depends on registered users and continuous internet connectivity, which may limit adoption in rural or resource-constrained areas.

Future work will focus on the following directions:

- Development of a dedicated mobile application for wider adoption and real-time accessibility.
- Integration of AI-based donor prediction models to recommend potential donors based on past behavior and location patterns.
- Establishing connections with hospital and government blood bank databases to synchronize inventory in real time.
- Addition of multi-language support to improve usability for users across different regions.
- Implementation of advanced analytics dashboards for monitoring donor activity, request fulfillment rates, and predicting future blood demand trends.

REFERENCES:

1. World Health Organization. “Blood safety and availability.” World Health Organization, 2022. Available: https://www.who.int/news-room/fact-sheets/detail/blood-safety-and-availability
2. Abdul-Gafaru, T., Laar, D. S., & Kubuga, K. K., “Design and implementation of an online blood donation system,” Asian Journal of Research in Computer Science, vol. 17, no. 4, 2024. Available: https://doi.org/10.9734/AJRCOS/2024/v17i4429
3. Gayathri, D., Mavushiga Shree, N., Nathanya Siphora, & Pooja, “Integrated blood bank management system with blood group compatibility checker,” International Journal of Research Publication and Reviews (IJRPR), vol. 5, no. 5, 2024. Available: https://ijrpr.com/uploads/V5ISSUE5/IJRPR28655.pdf

4. Tarun, M. S., Kishan, S. R., Suraz Basha, S. A., Ahammad, S. R., Chandrasekhar, U., & Bagga, N., "Blood bank management system," Journal of Emerging Technologies and Innovative Research (JETIR), vol. 8, no. 5, 2021. Available: https://www.jetir.org/papers/JETIR2105429.pdf
5. Sankareshwari, S., Dali, M., Mhaskar, A., & Gawas, S., "Web designing for online blood donor management system," International Journal of Creative Research Thoughts (IJCRT), vol. 10, no. 2, 2022. Available: https://ijcrt.org/papers/IJCRTO020009.pdf
6. Manika, G., Reddy, V. S., Vamshi, K. S., & Prashanthi, M., "Development and implementation of a web-based blood bank management system for efficient blood donation and distribution," International Journal for Research in Applied Science and Engineering Technology (IJRASET), vol. 11, no. 1, 2023. Available: https://doi.org/10.22214/ijraset.2023.50493
7. Babu, P. S., Arunkumar, M. V., Priya, P. K., Soundarya, N., & Meenakshi, Y., "Blood bank and donation management system," ResearchGate, 2025. Available: https://www.researchgate.net/publication/390756054_BLOOD_BANK_AND_DONATION_MANAGEMENT_SYSTEM
8. Noori, M. A., Hussien, S. A. S., & Al-Janabi, T. A., "Blood donors appointment booking and managing system using PC and mobile web browsers in current pandemic (COVID-19)," Indonesian Journal of Electrical Engineering and Computer Science, vol. 23, no. 1, 2021. Available: https://www.researchgate.net/publication/353485434