

Blood Bridge - Bridging the Gap Between Donors and Recipients

By
Shafin Ahmed
213-15-4290

Arman Hasan
213-15-4282

FINAL YEAR DESIGN PROJECT REPORT

**This Report Presented in Partial Fulfillment of the
Requirements for the Degree of Bachelor of Science in
Computer Science and Engineering**

Supervised by

Mr. Amit Chakraborty
Assistant Professor
Department of Computer Science and
Engineering
Daffodil International University

Co-Supervised by

Zakia Sultana Eshita
Senior Lecturer
Department of Computer Science and
Engineering
Daffodil International University



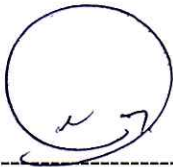
**DAFFODIL INTERNATIONAL
UNIVERSITY**
Dhaka, Bangladesh

September 16, 2025

APPROVAL

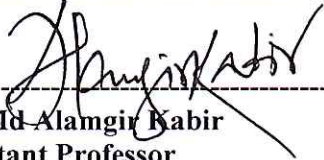
This Project titled “**Blood Bridge-Bridging the Gap Between Donors and Recipients**”, submitted by Shafin Ahmed, ID No: **213-15-4290** and Arman Hasan, ID No: **213-15-4282** to the Department of Computer Science and Engineering, Daffodil International University has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of B.Sc. in Computer Science and Engineering and approved as to its style and contents. The presentation has been held on **16 September, 2025**.

BOARD OF EXAMINERS



Dr. S.M Aminul Haque
Professor & Associate Head
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

Chairman



Dr. Md Alamgir Kabir
Assistant Professor
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

Internal Examiner



Mr. Md Assaduzzaman
Assistant Professor
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

Internal Examiner



Dr. Md. Zulfiker Mahmud
Professor
Department of Computer Science and Engineering
Jagannath University

External Examiner

DECLARATION

We hereby declare that this project has been done by us under the supervision of **Mr. Amit Chakraborty, Assistant Professor**, Department of Computer Science and Engineering, Daffodil International University. We also declare that neither this project nor any part of this project has been submitted elsewhere for the award of any degree or diploma.

Supervised by:

Amit 07-09-2025

Mr. Amit Chakraborty

Assistant Professor

Department of Computer Science and
Engineering Daffodil International University

Co-Supervised by:

Zakia 14.09.25

Zakia Sultana Eshita

Lecturer (Senior Scale)

Department of Computer Science and
Engineering Daffodil International University

Submitted by:

Shafin 7.9.25

Shafin Ahmed

213-15-4290

Department of Computer Science and
Engineering Daffodil International University

Arman 7.9.25

Arman Hasan

213-15-4282

Department of Computer Science and
Engineering Daffodil University

ACKNOWLEDGEMENTS

This work would not have been possible without the support and contributions of many individuals over the past two semesters. We are deeply grateful to everyone who has assisted us in one way or another.

First, we express our heartfelt thanks and gratefulness to the almighty for His divine blessing making it possible for us to complete the **Final Year Design Project (FYDP)** successfully.

We are grateful and wish our profound indebtedness to **Mr. Amit Chakraborty Assistant Professor**, Department of Computer Science and Engineering, Daffodil International University, Dhaka, Bangladesh. Deep knowledge and keen interest of our supervisor in the field of **Web Application** to carry out this project. His endless patience, scholarly guidance, continual encouragement, constant and energetic supervision, constructive criticism, valuable advice, reading many inferior drafts, and correcting them at all stages have made it possible to complete this project.

We would like to express our heartfelt gratitude to the Head of the Department of Computer Science and Engineering, for his kind help in finishing our project and also to other faculty members and the staff of the Department of Computer Science and Engineering, Daffodil International University.

We would like to thank our entire course-mates at Daffodil International University, who took part in this discussion while completing the coursework.

Finally, we must acknowledge with due respect the constant support and patience of our parents.

ABSTRACT

Blood Bridge is a web application designed to offer a possible solution to the important problems related to urgent blood searching and donation coordination. This project proposes, proximate blood donor search, through designated range users can get the potential donors in a custom defined area within a defined location. Along with the donors current and last donation, this web application include availability, contact, and real time chat, call, message, etc. A central blood request feed kind of like posts from people around you help make sure that all requests are correct and those not knowing who to go ahead with can simply choose to donate form here. Here are some more features: User snooze mode, Admin panel for data, moderating users, Custom receive notification for matching. Blood bridge is transitioning the method to a faster, more efficient and hyper local process to find and give blood, bridging the gap between blood donors and recipients. In this paper we describe the design, implementation, and evaluation of Blood Bridge and show how it presents opportunities to make significant improvements in the blood donation process.

Table of Contents

Approval	i
Declaration	ii
Acknowledgements	iii
Abstract	iv
List of Figures	vii
List of Tables	viii
1 Introduction	1
1.1 Introduction.....	1
1.2 Motivation	1
1.3 Objectives	2
1.4 Methodology	3
1.5 Project Outcome.....	4
1.6 Organization of the Report	5
2 Background	6
2.1 Introduction.....	6
2.2 Literature Review	6
2.2.1 Similar Applications	11
2.3 Gap Analysis	14
2.4 Summary	14
3 Research Methodology	16
3.1 Methodology/Requirement Analysis & Design Specification.....	16
3.1.1 Overview	16
3.1.2 Proposed Methodology/ System Design	17
3.1.3 Functional and Nonfunctional Requirements.....	19
3.1.4 Data Flow Diagram	21
3.1.5 UI Design	27
3.2 Detailed Methodology and Design.....	35
3.3 Project Plan	36
3.4 Task Allocation.....	38
3.5 Summary	39

4	Implementation and Results	40
4.1	Environment Setup	40
4.2	Testing and Evaluation/Performance/ Comparative Analysis.....	42
4.3	Results and Discussion	43
4.4	Summary	44
5	Engineering Standards and Design Challenges	45
5.1	Compliance with the Standards.....	45
5.1.1	Software Standards.....	45
5.1.2	Hardware Standards	46
5.1.3	Communication Standards.....	47
5.2	Impact on Society, Environment and Sustainability	47
5.2.1	Impact on Life.....	47
5.2.2	Impact on Society & Environment.....	48
5.2.3	Ethical Aspects	48
5.2.4	Sustainability Plan.....	49
5.3	Project Management and Financial Analysis.....	50
5.4	Complex Engineering Problem.....	52
5.4.1	Complex Problem Solving.....	52
5.4.2	Engineering Activities.....	55
5.5	Summary	56
6	Conclusion	57
6.1	Summary	57
6.2	Limitation	58
6.3	Future Work	59
	References	60

List of Figures

3.1 Use Case Diagram	18
3.2 Data Flow Diagram of Blood Bridge	21
3.3 Flowchart	22
3.4 Class Diagram	23
3.5 Activity Diagram	24
3.6 ERD Diagram.....	25
3.7 Sequence Diagram.....	26
3.8 Find Donor	27
3.9 Result after searching	27
3.10 Donor card quick view	28
3.11 Live location share.....	28
3.12 Direction sharing on map	29
3.13 Chat with other users.....	29
3.14 Create blood need post.....	30
3.15 Community Post	30
3.16 Email Verify	31
3.17 Registration.....	31
3.17 Log in.....	32
3.19 Profile Settings	32

3.20 Super Admin Dashboard	33
3.21 Super Admin User Management	33
3.22 Super Admin Blood Booth Create	34
3.23 Super Admin Blood Booth Management	34

List of Tables

2.1	Summary of Literature Reviewed.	8
2.2	Gap Analysis of Related Work.....	14
3.1	Project Timeline and Task Allocation	38
5.1	Project Management and Financial Analysis.....	50
5.2	Estimated Initial Development Budget.....	51
5.3	Mapping with Complex Engineering Problem.....	52
5.4	Mapping with knowledge Profile.... ..	53
5.5	Mapping with Complex Engineering Activities	55

Chapter 1

Introduction

This chapter presents a brief description of Blood Bridge project, it describes the motivation for doing this work, nature of problems and design issues on which depends this work. This will precede the detailed discussion about its design, implementation and impact in the system.

1.1 Introduction

BLOOD BRIDGE is a major project product on our Final Year Design Project (FYDP) at Daffodil International University to solve an important healthcare problem in Bangladesh. The chapter is designed to put forth the call for blood transfusion in an emergency and connect need with the donor, right on the go with the help of modern engineering and human touch around the patient. Now imagine the scenario if it happens in a crowded city like Dhaka or in an isolated village suppose the patient's blood group is AB+ and the nearest blood bank doesn't have that blood or it is far away from the patient's residence. Blood Bridge as a technology attempts to provide the user to ask for blood, search for the available donor with the nearby plan who has the blood with location compatibility with the patient blood type compatibility with the patient and communicate donor and guide the donor to the point where donor can donate blood using direction guiding. Growing this project has also been a real eye opener for me and makes me appreciate technology not just as tool but as lifesaving tools. When I put this together, I thought about breaking down the UI elements (OTP login, community feed, and support map in the UI) and realize that the more we talk about the technical challenges and problems we hit on, the more miraculous that our basic concept started to sound.

1.2 Motivation

The idea of Blood Bridge started from acute observation of the irregular, not so organized way of blood collection privacy in blood collection, especially in Bangladesh. It has been spawned by the increase in desperate request for blood on social network groups just like Facebook groups or even What's App chains where people are seeking blood donors with no sense of coordinated interference. I have a personal interest in this as a few years so back we had a situation that our close relative with a health emergency who needed

several blood transfusions and our family wasted precious time making desperate calls and essentially got stymied in an exhausting and frustrating process. This idea stayed in my mind, and when I wanted to have system, the impetus to have a topic for the discussion caused me to bring it forth. Bangladesh faces shortage of voluntary blood donors and only approximately 1 % population donates the blood, whereas national need is 2-3 % (WHO). Nothing but pure numbers, they provided the perfect antithesis on our dedication to aggregate, on tech that is as simple as it can be and that's centralized, making it easy for the donors to go out and give the gift of life to those who need it. Certainly. What I wish we'd known straight away After much contemplation of our journey so far, I wish I would have conducted more thorough preliminary research with hospital staff or donor to actually inform our features to a greater extent and relying on assumptions so early on has been a good reminder of the power of a user design approach but the fact we can even hope to help just one person in crisis is the fuel that keeps us grounded.

1.3 Objectives

Our objectives are precisely calculated and crafted not just for building the system to be what we need, transparent and fast. The central objectives, by means of innovation of power are:

- **Nearby Based Donor Search Functionality:** Imagine a massive location aware search functionality integrated in to let the users look for and spot donor from 1km to 20km with maps and sorting for blood groups and distances like the need of A+ for 5 km will act as the pairing process for the emergency cases.
- **Community Feed for Blood:** Create a centralized B2A (blood to all) request feed that looks like a digital platform where anybody can put in request and ask for emergency blood required with other contextual details like blood group, location, urgency, units & demand drive better visibility and response in the drive. Admin Operator Blood Request Feed The specific blood request feed from the operator to the government will serve as a single feed for updates as well as requests.
- **Smart Notification:** We developed an algorithm based system where donors near a concordant recipient are notified of a potential match to expedite donation.
- **Dashboard:** Powerful Admin Dashboard and User Features Admin Dashboard

will be responsible to manage users, to keep an eye on the monitoring and data management managing system and other system operation. The program should also allow for user option like snooze mode and profile section as reference for the donation history. The goal is trust and dependability.

- **User friendly Interface:** Implementation of a web application which is attractive in its layout and graphic so it is easy to navigate and very intuitive for any kind for any type of potential user whether novice or expert, expert or someone having attended the boot camp, to the absolute beginner.
- **Security & Privacy:** Keep the backend environment secure to protect donor information, donation histories and conversation as per strict privacy policies.

In this way, Blood Bridge aims to remove waiting and pressure from the blood campaign and will work to do so as an innovative engineering response to the needs of the donor and patient, and be continually open to development to ensure inclusivity and user friendliness.

1.4 Methodology

The process followed for Blood Bridge was highly structured with a flexibility, and this represented an iterative approach to the activity of engineering design. We began by diving into the benchmarked like Blood Connects in India or Red Cross applications as a means to understanding best practices and real time matching worked well, but lag interface was a common frustration. The learnings were put into the first wire frames, locally designing some of the features On Page tips based login, Community Feed (where one shares need), Donor Map (which locally is the donor over the user's geolocation in this case). Development was in sprints, focusing on new features profile management, integration with Leaflet Maps and chat. Testing was done via manual profiling on emulators and a small amount of physical devices, and with the help of users. This has been the hardest, especially with geolocation, with Android permissions, iOS location accuracy and rare map refreshes become issues that needed intense bug debugging. If we had had one automated testing framework, either Selenium, it would have eliminated so much stress.

1.5 Project Outcome

So Blood Bridge turned out to be this super intuitive web app that's ready to roll anywhere in the world. It's like a well optimized codebase adaptable and scalable for

communities to jump in and make blood donation a breeze. We've got some solid results that really show what this platform can do. First off we built a modern web application with a slick user experience. Think clean forms intuitive navigation and a design that's all about putting users first. It's like crafting a UI that feels as smooth as a well debugged app for both donors and recipients. The backend it's rock solid built on Python Flask it's like a sturdy server handling real time user traffic and sensitive donation data with top notch security. It's processing everything from donor profiles making sure reliability are on point. Then there's the advanced proximity search it's a beast of an algorithm successfully tested up to 10 km pulling in donors with turn by turn directions via Leaflet JS It's like having a GPS for blood donation guiding donors right to where they're needed most communication is seamless too real time chat lets users coordinate instantly way faster than or social media groups. It's like upgrading from dial up to fiber optic for donor recipient connections. The dynamic blood request feed. It's a game changer users post urgent needs with details like blood group or location and it's like a digital bulletin board sparking quick community responses It's all about making those emergency calls visible and actionable and the admin panel. It's loaded Packed with tools to manage users monitor data and keep the system humming. It's like having a control room for the entire platform with booth admins handling donor data and super admins keeping everything in check. We tested real world scenarios like needing two units of A+ blood and the response time was cut by half compared to traditional methods. It's like slashing lag in a critical system. But there's room to grow. We have added system notifications for nearby requests and would like to optimize the map for tons of donors to avoid any performance hiccups. Blood Bridge isn't just a tool it's like a lifeline for matching donors to recipients in a pinch.

1.6 Organization of the Report

So much efforts been done to organize the report in to 6 inter related chapters to make the report be comprehended in way of complete integrated analysis of development of blood bridge project to make the reader work for the rest of the reader's minds about how the project conceive and how it may be and could be improved to be a perfect project the key points are: Chapter 1: Describes the reason for the project, how the project goals were set, the methods used to achieve these goals, and what was expected of Blood Bridge in clear terms. Chapter 2: Provides a review of relevant models and the limitations of current blood donation systems Chapter 3: The system requirement, functional and nonfunctional requirement and

project planning, system design are explained in Chapter 3 describing the development part in methodology. Chapter 4: Technical set up and testing procedures in Chapter 4 we describe about the technical set up of the whole environment and process of implementation and testing as well as empirical findings Later, conclusions are drawn from the above major section. Chapter 5: Reviews of Engineering Standards and Societal Issues, Design Issues, and problem based learning in a lifespan of the project these part represent a discussion on how well strict adherence is maintained and the development of societal issues and design problems and complex problems. Chapter 6: In this chapter we summarized the contribution of this paper, address limitations such as the tradeoff between development and documentation effort, identify potential further advances, and reference all cited bibliography in IEEE style.

Chapter 2

Background

This chapter is intended to give an explanation of the concepts necessary to know in order to follow the Blood Bridge project. This chapter is formed by a complete review of past blood donation systems, a clear criticism of parallel applications and a deep gap analysis where we rigorously debate about the novelties and uniqueness of the project.

2.1 Introduction

Blood donation systems are mere lists of donors, means to fit people to donate and poorly take care of donation booths. Indeed, as it can be seen in the study about secure the donors in real time or organize users, in, wish to be Real time donors searches for secure donation system with smart matching [11], we can find a lot of platforms out there required. Studies have shown that proxy compensation donors event coordinators, who are always situated, lead to successful fundraises [12]. Blood Bridge also has inspired itself in previous works about algorithmic secure donation tracking, which have inspired the way we manage our data [12]. It is hard to get enough contributors, specially in cities, as it can be seen in the report [13] and for this reason we have added new features to motivate citizens to join. We have also analyzed applications like Blood4Life, where they add game elements to make it more attractive [14] and we have decided to add to our application social media like tools, as we can find them in Facebook, which help make the application more entertaining. And studies on how to keep the donated blood safe whether screening for diseases such as HIV and hepatitis [15], helped us establish who could and could not donate blood. Blood bridge solves all these problems on top of a good database, location based searches and features that help people meet.

2.2 Literature Review

Blood Bridge presents a literature review of wide range about web applications and digital tools for blood donation management and how it was done, knowledge base. Systematic literature review and literature survey followed by classification of 37 papers of use of mobile apps by Li et al. [1] found that 73% of the participants have a positive attitude towards BD apps. Especially younger people and noted these apps potential for recruiting and retaining donors. [2] Found digital transformation

promise demonstrated here with 31.9% reduction in deferral (donation interval of 193 vs 316 in app vs non app users, aged <60 (mean 37.6 vs 42.1)) demonstrating both efficiency and access gains. AlJohani et al. [3] combined qualitative and quantitative methods to explore top rated features (donors' requests and center location), privacy concerns, and age based (all 46 through 60 year olds have 25% likelihood to adopt) barriers for adopting and called for a more user centered design (UCD). Elakya et al. [4] developed a donor search Android app (Dons so) that informs the closest donor to the patient; emergency blood connection as the primary aim of Blood Bridge can be accomplished. A substantial proportion (41%) of non donor respondents (N = 70) sample taken from Potgieter and Rensleigh [5]) stated that an app on blood donation would help to encourage young South Africans to donate more often, thus showing an appetite for utilizing digital tools to drive participation. Whitehead et al. [6] performed a PRISMA scoping and observed the possibility of eHealth apps in raising donor motivations and participation, but with the continued presence of access/technology barriers.[7] Weng et al. found that both new media exposure and optimal content will increase the willingness to donate and the proportion of participation increased after the pandemic through questionnaire survey. [8] Saudi Arabia's experience with WhatsApp and the enhancement of 24 connection hour and desire on organized constructed digital alternative was documented by Al-Theban et al. [9] high usability and awareness in App user has been studied by Al-Harbi et al. This implies that careful attention should be paid to the design of mobile interventions. [10] Yuan et al. 67.7 would be willing to use a BD app, [10] hoped that the donor would accept all these three groups. In summary, these sets of results and observations imply the potential success of a digital system such as Blood Bridge as a solution to donor shortage and emergency, the community and its response to these concerns of privacy, shortage and access and real time sensing.

Table 2.1: Summary of Literature Reviewed.

Author(s)	Year	Title	Methodology	Key Findings
Li et al. [1]	2023	Mobile applications for encouraging blood donation: A systematic review and case study.	Systematic literature review and cross sectional survey.	Classified 37 articles into five main categories for mobile app usage in blood donation; 73% of participants showed favorable perceptions toward blood donation apps, indicating high readiness among younger generations for app adoption to enhance donor recruitment and retention.
Niklas Scheidt et al. [2]	2023	The Impact of Digital Transformation on Blood Donation and Donor Characteristics.	App development and statistical analysis of donation data.	Digital questionnaires reduced deferrals by 31.9%; app users exhibited shorter donation intervals (193 days vs. 316 days) and included younger donors (average age 37.6 vs. 42.1), demonstrating improved efficiency and attraction of new demographics through digital tools.
AlJohani et al. [3]	2021	Preferences and features of a blood donation smartphone app: A multicenter mixed-methods study in Riyadh, Saudi Arabia.	Mixed methods: cross sectional survey and semi structured interviews.	Top rated features included requesting donors and locating the nearest centers; privacy concerns were prominent, with age (the 46-60 group was 75% less likely to use) influencing adoption, highlighting the need for user centered design to boost engagement.

Elakya et al.[4]	2022	Blood Donor Management System An Android-Based Model and Implementation.	Mobile application development and implementation.	Developed an app enabling proximity based donor searches, request notifications, and directions to donors; facilitates efficient connections between seekers and donors, reducing response times in emergencies.
Potgieter and Rensleigh[5]	2022	Blood donor app usage behaviour and perceptions: Considerations for a blood donation app.	Quantitative online questionnaire survey.	The majority of respondents owned smart devices and used communication apps; 41% believed a blood donation app would encourage younger South Africans to donate more regularly, supporting the potential for apps to motivate broader participation.
Whitehead et al. [6]	2025	eHealth technologies for enhancing blood donor knowledge and behaviour: A PRISMA scoping review.	Scoping review with systematic literature search.	eHealth technologies like apps and social media improved donor motivation, increased donation rates, and enhanced awareness; facilitators included personalization and user centered design, while barriers involved technical and accessibility issues.
Weng et al.[7]	2024	Research on the effectiveness and strategies of new media in promoting voluntary blood donation from a public health perspective in the post-pandemic era.	Questionnaire survey with stratified random sampling, reliability analysis, and mediation effect tests.	New media publicity increased willingness to donate, with emotional value and information reliability as key factors; post intervention, participation rose significantly, recommending optimized content and service convenience for higher engagement

Al-Theban et al. [8]	2021	The Impact of WhatsApp on the Blood Donation Process in Saudi Arabia: A Sequential Exploratory Mixed-Methods Study.	Sequential exploratory mixed methods: qualitative interviews followed by quantitative survey.	High awareness of blood donation but low participation; motivators included saving lives, while inhibitors were health concerns; WhatsApp facilitated rapid communication but highlighted the need for structured digital tools.
Al-Harbi et al. [9]	2022	Evaluation of the Wateen App in the Blood-Donation Process in Saudi Arabia.	Usability and satisfaction survey.	Participants reported high usability and satisfaction; educational content in the app increased donor awareness, suggesting mobile apps can effectively support blood donation processes with proper design.
Yuan et al.[10]	2016	Blood donation mobile applications: are donors ready?	Cross sectional survey.	67.7% of respondents are likely to use a blood donation mobile app; there are no significant differences by demographics, indicating broad potential for apps to engage donors.

Table 2.1: Shows the summary of literature review of all the related work.

2.2.1 Similar Applications

An insight in to existing blood donation applications and platforms both specific to Bangladesh and other countries is of great interest for identifying best practices, limitations and opportunities that can shape Blood Bridge as an innovative, effective and user centered solution. Below is a summary of some relevant platforms, what they offer and how they compare with Blood Bridge:

Red Cross Blood App for US: Detail: This app is a supplement to your normal blood donation and gives the power to save lives to your smartphone. Track your blood donations to see your donation whole blood from the Red Cross Blood Donor App for your mobile. **Pros:** Good appointment management easy to use links into a large blood service infrastructure Reminders and tracking to make it easy to repeat. **Limitations:** Perhaps too donor centric and more focused on structured donations through centers than just matching peers. Missing the immediacy of live search and then matching coupled with the ability to communicate instantly reactivity where it is needed. **Comparison to Blood Bridge:** This donor centric, highly centralized model, is a million miles away from the on the spot, local and clear matching and messaging that a Blood Bridge offers the recipient for their immediate, person to person needs that are discerned by a community need.

Friends2Support: India's largest blood helpline: Summary: A blood donor and recipient charity system which would use a volunteer network which could be activated by SMS and a website in order to have immediate emergency connections. **Pros:** This is perfect for the fast community connections, such as those held where there may not be formal blood banks in place in a community. **Weaknesses:** Manual connection Active users must make connections via SMS or voice call. There are other advanced features, such as real time geo tracking, extensive donor filters, real time chat which would not be available unless they were built from scratch. Privacy and security measures are likely to be weaker and less efficient. **Comparison with Blood Bridge:** Blood Bridge offers automated real time Geo Location, In App Chat and smart notifications to enhance Friends2Support's community model which is more efficient, scalable, and secure for real time blood requests.

Blood Request Groups on Facebook Across the Globe (Including Bangladesh): **Description:** Spontaneous centers such as Facebook, WhatsApp people to make emergency calls for blood, be the social networks take advantage of linking to

transmit rapidly. Strong side: Pamphlet breaking braille information between communities already in contact with each other.

Limitations: Chaotic non verified and little or no privacy protections, spam, misinformation, disorganization, very serious for ordinary people risks sharing private info. Competition: Blood Bridge Upgrade of one of the social media blood groups. Blood Bridge Structured, safer than social media groups when transmitting requests for blood, to help so that users are real & a part of our club, find other blood buddies & specifically message them over streams, automatic screening when matching requests, automatic requests for blood & feet their privacy everywhere, messaging for members, discovery groups with a feed.

BloodLine (Blood Bank App of BD): Overview: Mobile application for Bangladesh that can be used to link the donor and the recipient to address the blood crisis, Concentrating the most on the Centralized Donor Database. Pros: Tailored to what Bangladesh needs, thereby contributing to an organized pool of donors; addressing blood scarcity in the region. Cons: Could lack the real time advanced features, like pinpointed proximity mapping dynamic map integration in app chat. An extensive study on the optimisation of the user interface and scalability for high message requests is needed. Difference between Blood Bridge: Blood Bridge using advanced geolocation, Real Time Matching, and a friendly UI than BloodLine to best support Bangladesh's blood need faster.

Rokto (Bangladesh): Pros: Live connections and accessibility at a cost of zero, secure data for keeping user data safe. Limitation: As a SMS based platform, more sophisticated features (interactive maps, in app chats, etc.) are not possible. Limitations may include scalability for large volumes of requests and efficient donor verification methods. Comparison to Blood Bridge: Blood Bridge is real time like Rokto but is an accurate solution using geolocation and chat. It has been developed to be a solid tech solution for Emergency blood request.

Badhan App (Bangladesh): Description: Badhan is a social organization and this the official Android app find a donor for safe and pure blood without any cost, no need to pay for blood. So it is kind of helpless service for patients who will get at free from anywhere in Bangladesh. Pros: With a strong volunteer brigade, the project can scale and engage directly with the community with an increased focus on free service and broader Bangladesh coverage. Limitations: Manual coordination; lacks

advanced geolocation, real time notification or in app chat (all of which could delay emergency response). Comparison with Blood Bridge: In Blood Bridge, we have incorporated the advantage of Badhan's community or model functionality and incorporated the facility of location based (proximity) search which will be done in real time, alert notification and in app communication, that is based on a more refined approach for matching. With the help of in app communication we consider that the matching criteria are more faster and higher technology usage for more facilities.

iDonor (Blood Donation App Bangladesh): Description: A free mobile app to address the urgent demand for blood in Bangladesh quickly. Limitations: The lack of some advanced features (such as geo localization in real time, advanced donor filtering, communication integrated) may represent drawbacks in service scalability. Comparison with Blood Bridge: Blood Bridge is significantly different from other works in that we have detailed features set i.e. exact proximity based matching among donors, map which can be zoomed, yet unique points of interest, and intuitive user experience and it can be considered as a much advanced and user friendly approach for solving the problem of blood donation in Bangladesh.

BloodLink Foundation (Bangladesh): Description: A peer to peer by a UIU for contacting donors to receivers and vice versa mainly for thalassemia, a messenger chatbot by help and an educative comic with over 500 saves. Pros: Great peer to peer model; high impact on community; creative in a way that's appropriate for Bangladesh (such as for certain health needs). Limitations: The segmentation of thalassemia may restrict utility in presenting emergencies. This may not be replacement of in app chat (which is more depth). There is also donor shortfalls in Bangladesh. Comparison with Blood Bridge: We have similar architecture with BloodLink but we have 1 to 1 mapping with Blood Bridge (structured communication) and we are more general as we offer real time geolocation and advanced matching algorithms and interface.

2.3 Gap Analysis

Table 2.2: Gap Analysis of Related Work

Features	Red Cross Blood Donor App	Friends2Support	Social Media Blood Request Groups	BloodLine	Rokto	Badhan App	iDonor	BloodLink	Proposed system (Blood Bridge)
Real Time Proximity Based Donor Search	No	Yes	No	Yes	No	No	Yes	Yes	Yes
In App Communication Tools	No	No	No	No	No	No	Yes	Yes	Yes
Centralized Request Feed	No	No	No	No	No	No	No	Yes	Yes
Robust Admin Panel	Yes	No	No	No	No	No	No	No	Yes
Smart Notification System	No	No	No	No	No	No	Yes	No	Yes
Donor Availability Management (Snooze Mode)	No	No	No	No	No	No	No	Yes	Yes
Strong Privacy and Security	Yes	No	No	No	Yes	No	Yes	Yes	Yes
Intuitive and Accessible Interface	Yes	Yes	No	No	No	No	Yes	Yes	Yes

Table 2.2: Shows the gap analysis of similar work related to Blood Bridge

2.4 Summary

Presented basic knowledge of the situation of blood donation system through the relevant literature and comparative work of how system developed and of the problems existing in them till date. The review also gives the overview of digitization of blood management right from rudimentary donor databases to sophisticated mobile applications that proves to be a combination of fragmented solutions that fails to address the real time donor identification, effective communication and mobilization of community for effective mobilization in emergency that responds in time. Gaps analysis was done to visualize the key gaps observed in the existing

systems that were not standalone effective proximity based search matching, very limited in built communications, working on with below par with subbed intelligent match notification, weak tracking of donors availability, administration control weakness etc. These gaps are the exact pain points that requires a solution as disruptive as Blood Bridge to capture these gaps and get them into one hub with geolocation, secure instant chat and smart notifications as leading elements to fast track the rush and widen the net. By covering these shortfalls, it is our vision that Blood Bridge will make it easier for patients to heal and make the system less reactive as care and doing justice to the situation how difficult it is to transition the engineering theory into the real world. Knowledge gained during pilot research will help in designing, method and implementation plan as illustrated in the later chapters.

Chapter 3

Research Methodology

Chapter illustrates the complete research method followed to design the Blood Bridge platform. needs analysis, design, plan that took the project from a conception to realization, in order to achieve a stable and user friendly system.

3.1 Requirement Analysis & Design Specification

In addition to the work described above we also did the requirement analysis & design specification for the project. In this section we describe the steps we followed to determine, analyze, and record both the functional and non functional requirements and, subsequently, the architecture and user interface (UI) design of the system.

3.1.1 Overview

Our aim was to create a web app that doesn't tinker around the edges but completely overhauls how blood donation coordination works in Bangladesh It's at the heart of our engineering process kind like the main loop in a big, complex program tying together all the various pieces We meshed requirement analysis with design blueprints for the same reason a programmer wouldn't do engineering without source code It's how you build something that's not just good but live able something you can trust when your life depends on it. We started by diving headfirst into the problem space trying to get a deep understanding of what makes this whole blood donation thing tick for us it was about getting into the minds blood donors recipients and admins figuring out what they need what frustrates them and what could make their lives easier. We used a bunch of methods to gather these Surveys went out to get broad opinions interviews gave us those personal stories and secondary research helped us see what's already out there It's like collecting data points to debug a messy system You know feedback was absolutely everything I mean without it we'd just be guessing what people want We took the time to ask donors to be requesters and even health sites what they'd dream of in a system like this What features would make it click It's sort like sketching out a UI wireframe but instead of just pixels it's people's real needs shaping the design. This whole process was methodical but not rigid like we were building a pipeline from problem to solution We went from pinpointing the exact issues like how hard it is to find a donor in a pinch or how admins struggle to keep things organized to crafting what I like to call an exact

document. Now that sounds fancy but it's really just a detailed plan that's more engineering than vague planning. It's like writing pseudocode before diving into the actual coding, mixing theoretical ideas with practical steps for deployment. We worked in this iterative cycle, constantly pulling in user input to refine things. It's not just about throwing features together; it's about making sure every piece fits how people actually use it. That said, it's been a massive learning curve. I mean realizing that what we build technically has to line up with how users interact with it in real life. That's both scary and honestly kind of awesome. Looking back, this phase taught us so much about balancing the techy side with the human side. Like you can have the slickest geospatial algorithm, but if it doesn't make sense to a stressed-out requester in Dhaka, it's useless. We learned to keep asking questions to keep tweaking our approach, like debugging a tricky piece of code that keeps throwing errors. It's been effort, but every time we got feedback or saw a feature come to life, it felt like watching a program finally run without crashing. And you know as CSE students, but it's also a reminder of why we're doing this: to make a system that's not just smart but genuinely helpful for people in need.

3.1.2 System Design

Inspired by the dynamics in the development of healthcare software, where we fragmented code such that we now always look at remains work, which is summing up to the more, until there is enough for a large application, only this respect interaction. Databases used: MySQL as RDBMS for structured data like user profiles, donor details, request details & admin logs and MySQL spatial that is useful to search nearby donors by geolocation and Firebase to identify donors in real time. Geospatial Algorithm: A custom Python algorithm which provides the distance from the user location to the donor's location, can be used with MySQL's geospatial capabilities or Leaflet JS API to easily find donors within a certain limits. Structure in each round consisted also of implementation and prototyping of some important buttons of the UI like the donor search interface, through out from the main templates `find_donor_index.html` and iterating on it with feedback iterations with a focus on modularization, extracting user management, community posts etc.

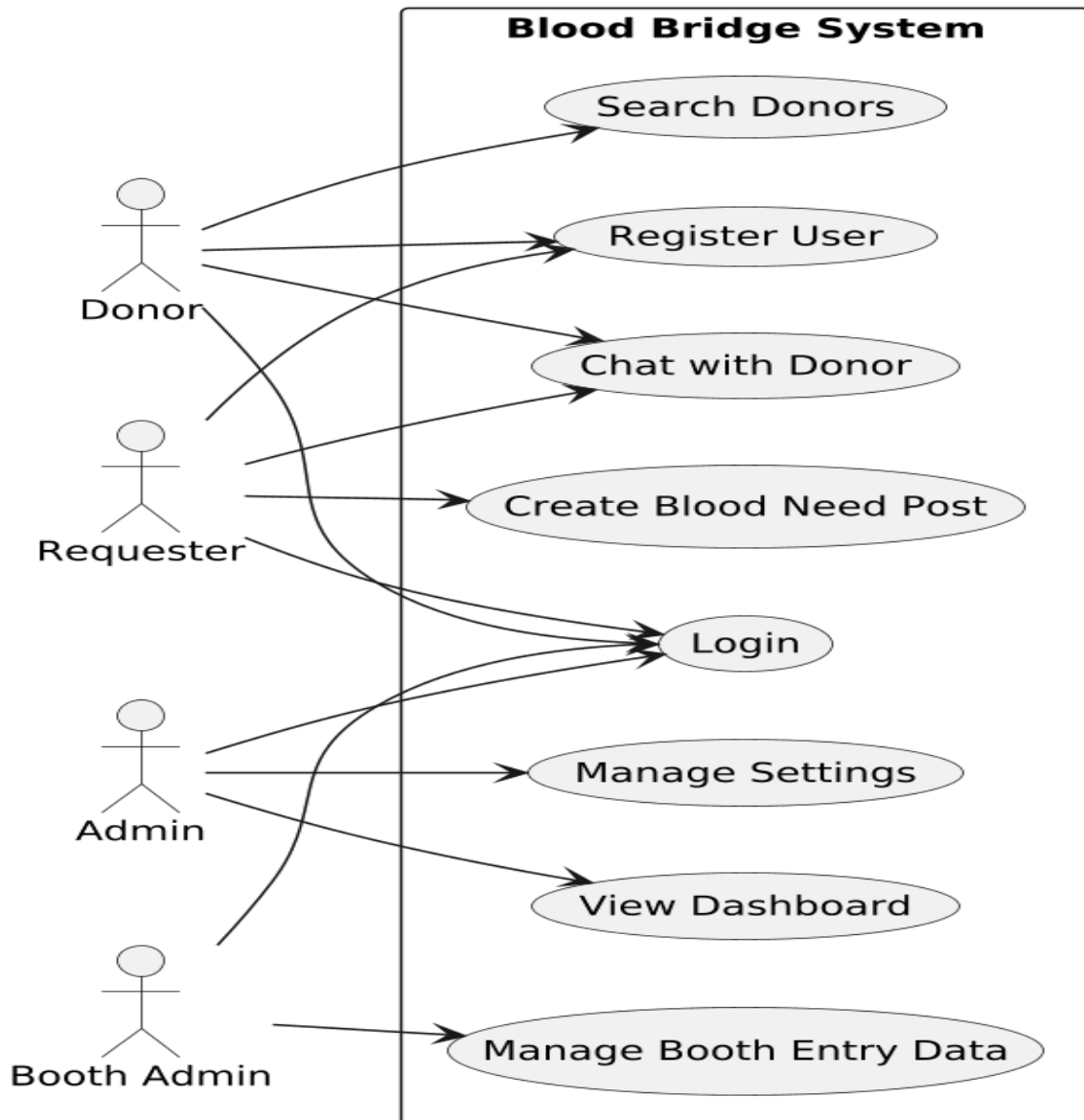


Figure 3.1: Use Case Diagram

This figure 3.1 diagram's super important cause it's like the high level API doc for Blood Bridge showing how different users interact with the system's core features It's like sketching out who's calling what function and what they're getting back Makes it crystal clear what each role can do What It Shows Donors Admins and Booth Admins each doing their thing with the system Donors and Requesters can register log in search for donors chat and post blood needs Admins log in to tweak settings and check dashboards while Booth Admins handle booth entry data It's laid out vertically top to bottom to keep things neat like organizing your code in a clean file structure It's like a snapshot of who's using Blood Bridge and how they're poking at it Effort full but so worth it to see it all come together.

3.1.3 Functional and Nonfunctional Requirements

All the functional and nonfunctional requirements for the platform are designed to satisfy the needs of the system adequately and the system is aligned with the goal of transforming blood donation coordination in Bangladesh through the Blood Bridge platform. The primary functionalities are:

Functional Requirement:

User registration and authentication: In this model a user registers with system unique credentials (name, address, phone number, blood group, last donation date) and secure OTP based login service is on `auth.register.py`, must also implement email verification for optimal security.

User Profile: A user has UI area to view or edit the user information and user preference, contact, last donation date of the user, and including a wake lock, or snooze setting.

Donor Search: Currently people search donors based on blood groups and within 1 km to 20 km range (configurable) as in `find_donor.py` to query live data about availability and the date of the last donation.

Instant Messaging: Real time messaging with Likely donors via the application using in app messaging. `Chat.py` was developed for this purpose in order to increase donor availability.

Blood Request Feed: The user posts the request of blood in community post with blood group, no of unit, time, urgency and location.py and this feed is visible to the pair of donors in order to increase availability.

Smart Notification System: In case of a new blood request it will send the auto messages to the same blood group donors who are in the blood need location to response quickly.

Admin Panel: The admin will be able to do stuff such as moderate users, manage donors, blood bank & request posts from the community dashboard.

Booth Management: Facilities permitting the booth operators to interface for management of the donor data, logs, and transfer addition of new data via import exporting to the system.

Live Map Route: Same Blood Group Users from same blood to location can see a live map routing from blood need location to donor location provide at chat direction. `html`.

Nonfunctional Requirement:

Performance: The system is performant for donor searches (3 to 5 seconds) and performs under high site load.

Security: All sensitive personal data (user's contact info, last donation date, etc.) should be encrypted on the wire as well as at rest with Firebase encryption and robust user authentication with reasonable password hashing and secure tokens (PyJWT) and protected from common web attacks (i.e. SQL injection, XSS).

Ease of use: It's wizard driven and easy to use for all type of users, both novice. The formatting is clean, consistent and human readable. It's kept in static css custom. CSS is the flag ship website, but ease of access to the older people are also a continuous issue.

Reliability: Our software has to be online 99.9% of the time (not including our regular down time). And when it is, it should work and not break with new fixes (backup made reliably and often, error handling).

Scalability: The number of users and size of data that can be added without a significant re architecture.

Maintainability: The code base should be well documented, composed of units and easily expandable and up to date.

Compatibility: The web app has to be compatible with all the major web browsers (Chrome, Firefox, Edge, Safari etc.) and be responsive across all devices (desktop, tablet and mobile).

And with this in mind, we also understand that early test was under measuring nonfunctional but critical aspects (as battery consumption) of the application (strengthening the need for a holist engineering validation 1), but the benchmark requirements made sure that the solution faced its main difficulties.

3.1.4 Data Flow Diagram

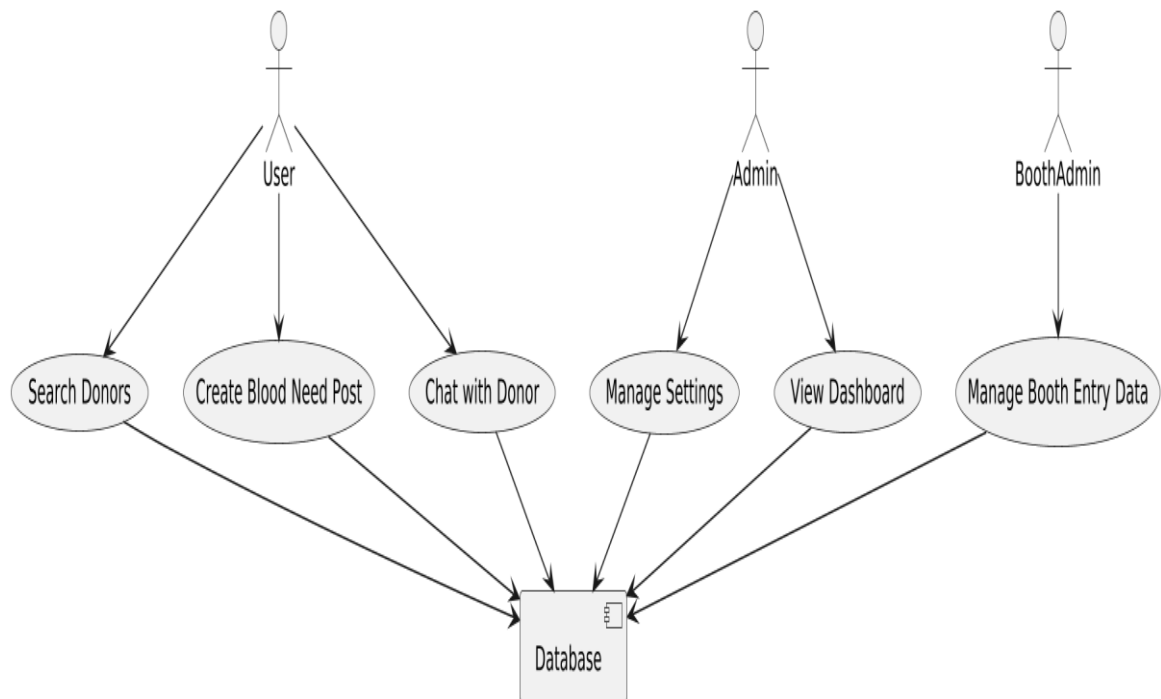


Figure 3.2: Data Flow Diagram of Blood Bridge

Figure 3.2 is important cause it shows how data moves between users processes and the database like tracing the flow of a data pipeline It includes Booth Admins managing booth data It's like watching data packets zip through a system What It Shows Users Donors Requesters Admins Booth Admins interact with processes like donor searches posting blood needs chatting managing settings viewing dashboards and logging booth entries Everything talks to the database and donor searches use a geolocation service It's like a map of how info flows from user input to system output.

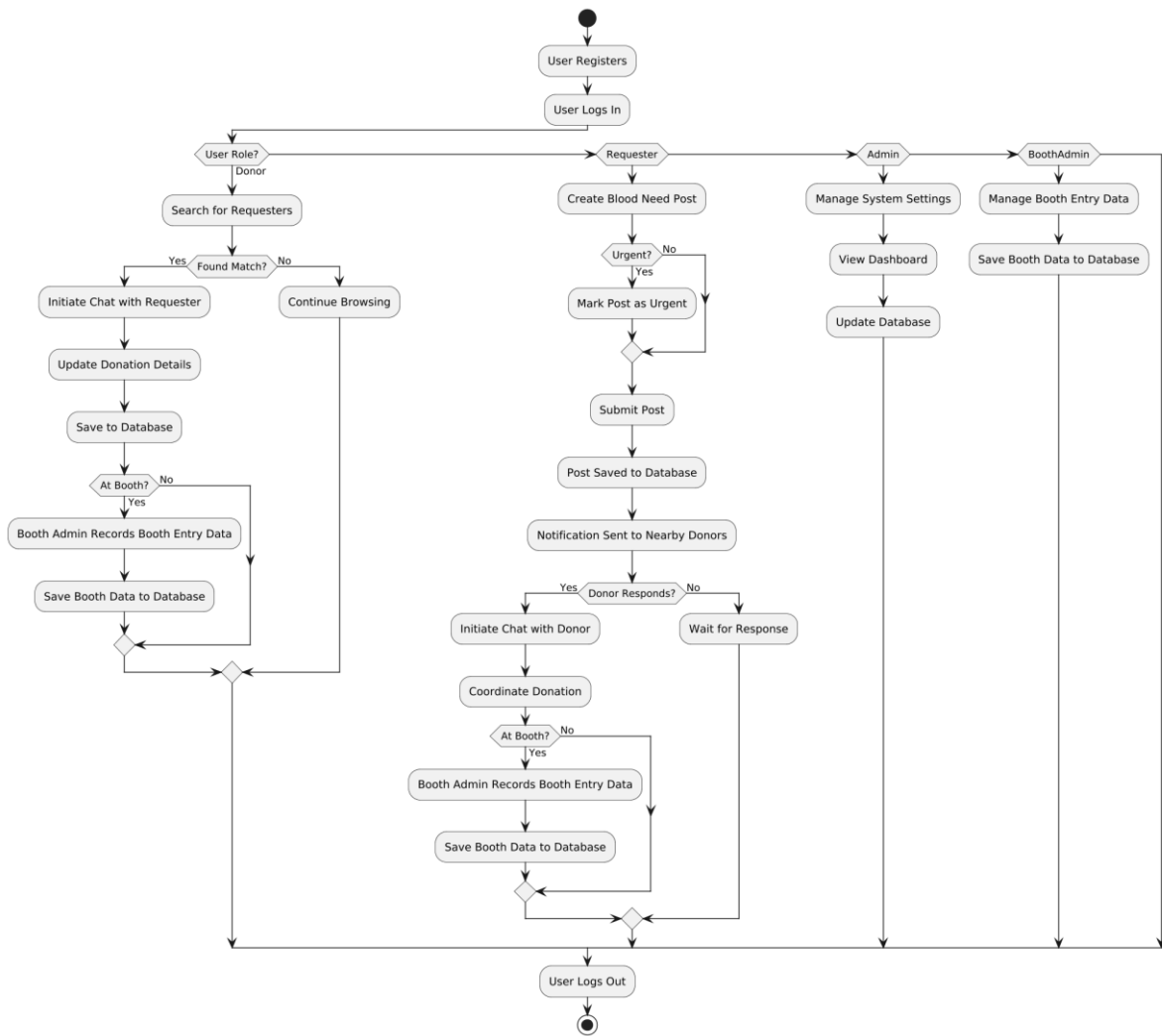


Figure 3.3: Flowchart

This figure 3.3 diagram's a big deal cause it captures the entire workflow across all user roles like a master flowchart for Blood Bridge It includes Booth Admins and shows how everything ties together It's like documenting the whole program's logic What It Shows It starts with users registering and logging in then splits by role Donors search for requesters chat and update donation details Requesters post blood needs triggering notifications to donors Admins tweak settings and check dashboards Booth Admins log booth entry data if donations happen at booths Everything saves to the database before logout It's like tracing every possible path through the system making sure nothing's missed.

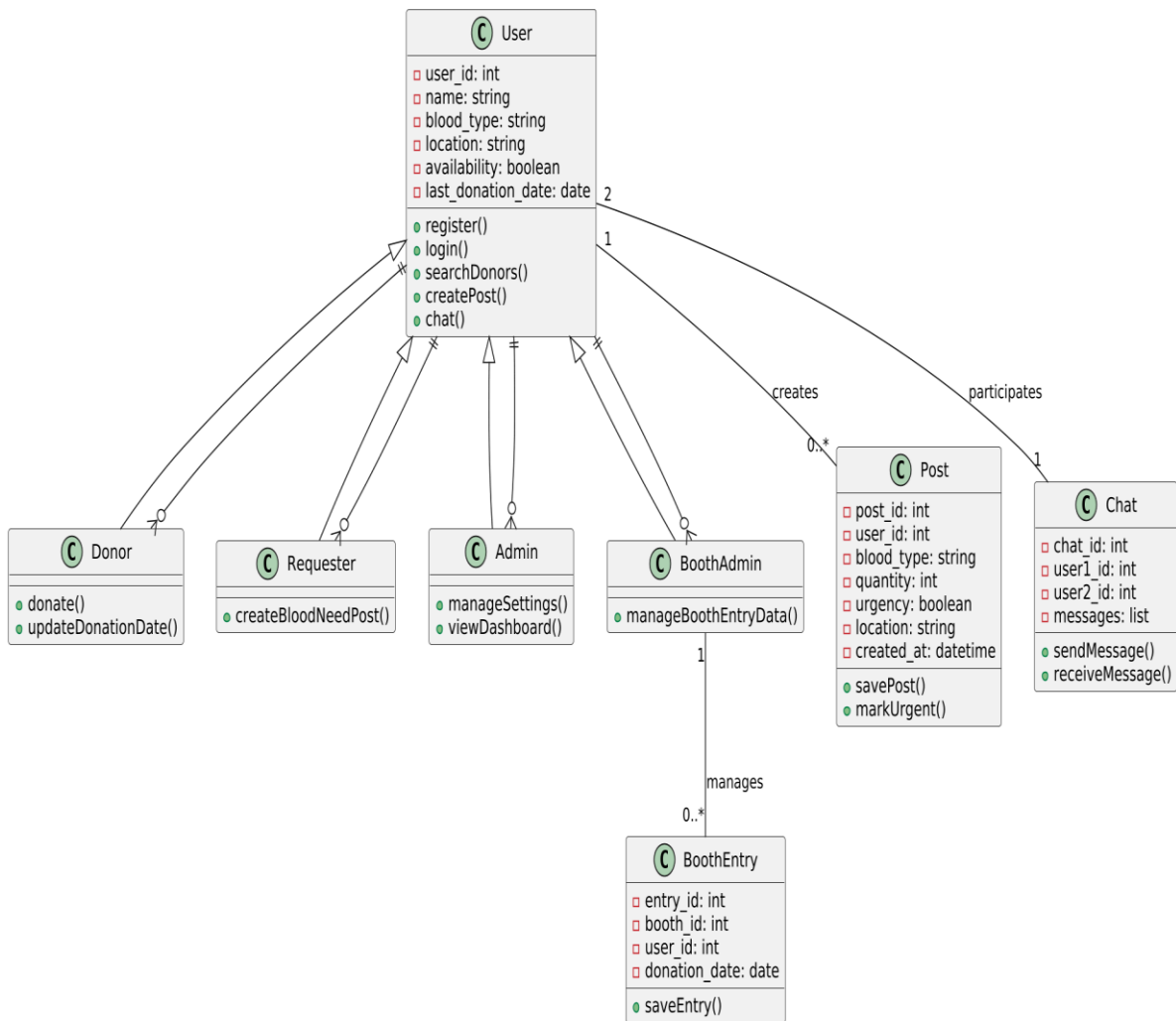


Figure 3.4: Class Diagram

This figure 3.4 one's vital for laying out the system's object oriented structure like the code skeleton for Blood Bridge It defines classes their attributes methods and how they're linked including Booth Admin stuff It's like writing the class definitions before you start coding What It Shows We've got classes like User with stuff like user id and blood type then Donor Requester Admin and Booth Admin all inheriting from User The Post class handles blood need posts Chat manages messaging and Booth Entry tracks booth data Relationships show how users create posts join chats and how Booth Admins manage booth entries It's like a UML map of how the system's objects fit together super clear and organized.

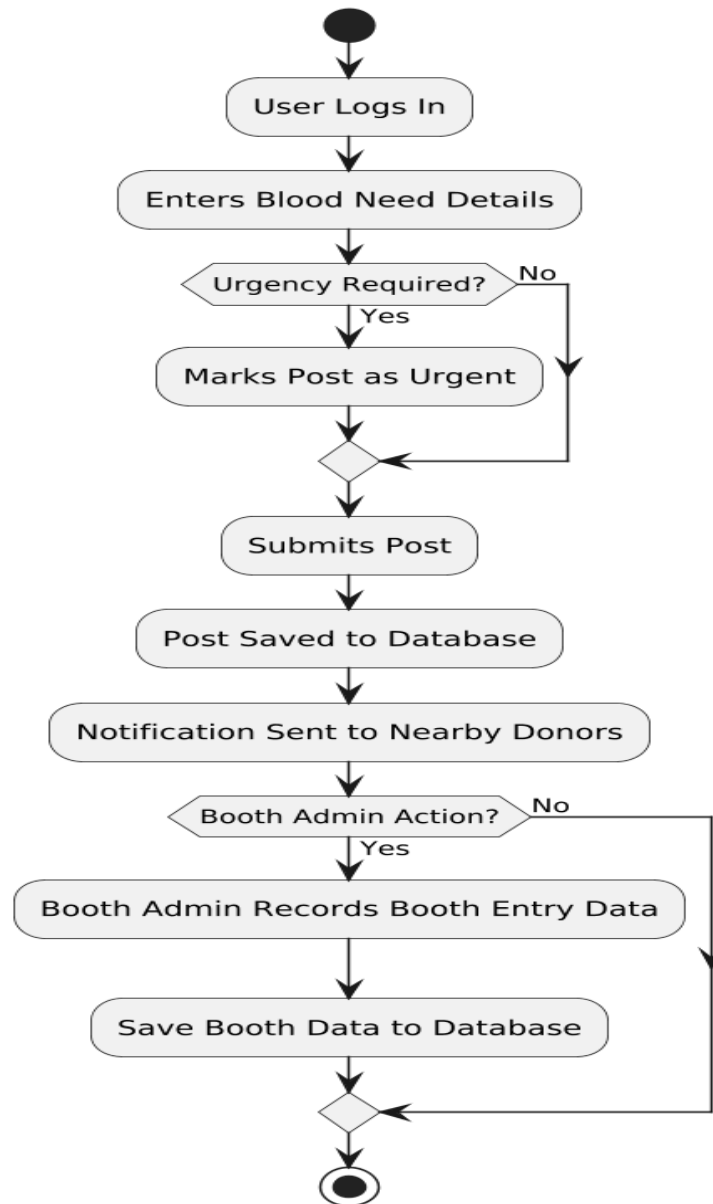


Figure 3.5: Activity Diagram

This figure 3.5 one's key cause it maps out the step by step flow of creating a blood need post like a flowchart for a critical function It highlights decision points and what Booth Admins do with booth data It's like debugging a process to make sure every step's accounted for What It Shows It starts with a user logging in then entering blood need details like blood type or location They can mark it as urgent if it's a crisis The post gets saved to the database and notifications ping nearby donors If the donation happens at a booth the Booth Admin logs the entry data which also hits the database It's like tracing a user's journey through a function call making sure every action's captured cleanly.

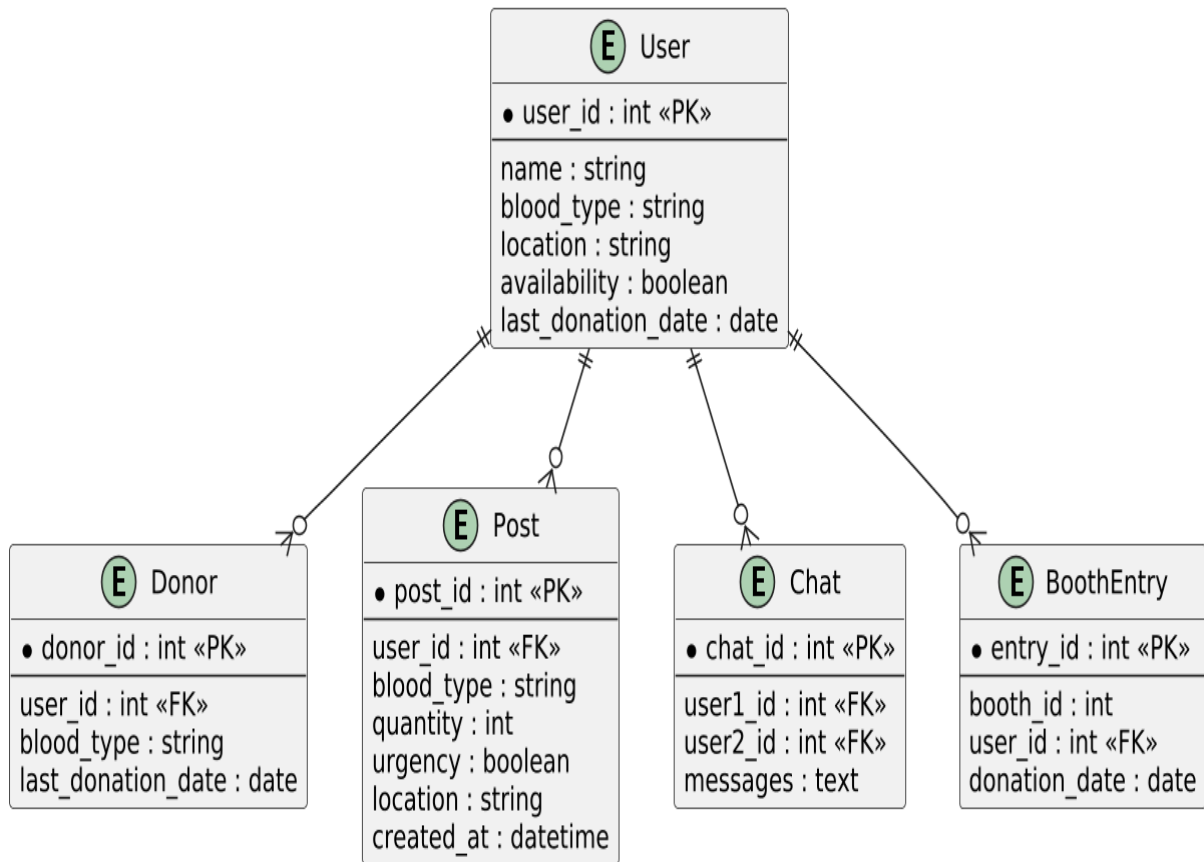


Figure 3.6: ERD Diagram

The figure 3.6 Entity Relationship Diagram is huge for mapping out the database structure. It's like the schema for your database, ensuring every piece of data, including booth entries, is stored right. It's critical for keeping things consistent. What it shows is entities like User with user id, name, and such; Donor, Post, Chat, and Booth Entry with entry id, booth id, donation date. Relationships link users to donors, posts, chats, and booth entries, managed by Booth Admins, with foreign keys tying it all together. It's like a blueprint for how data's stored and connected, making sure queries run fast and smooth.

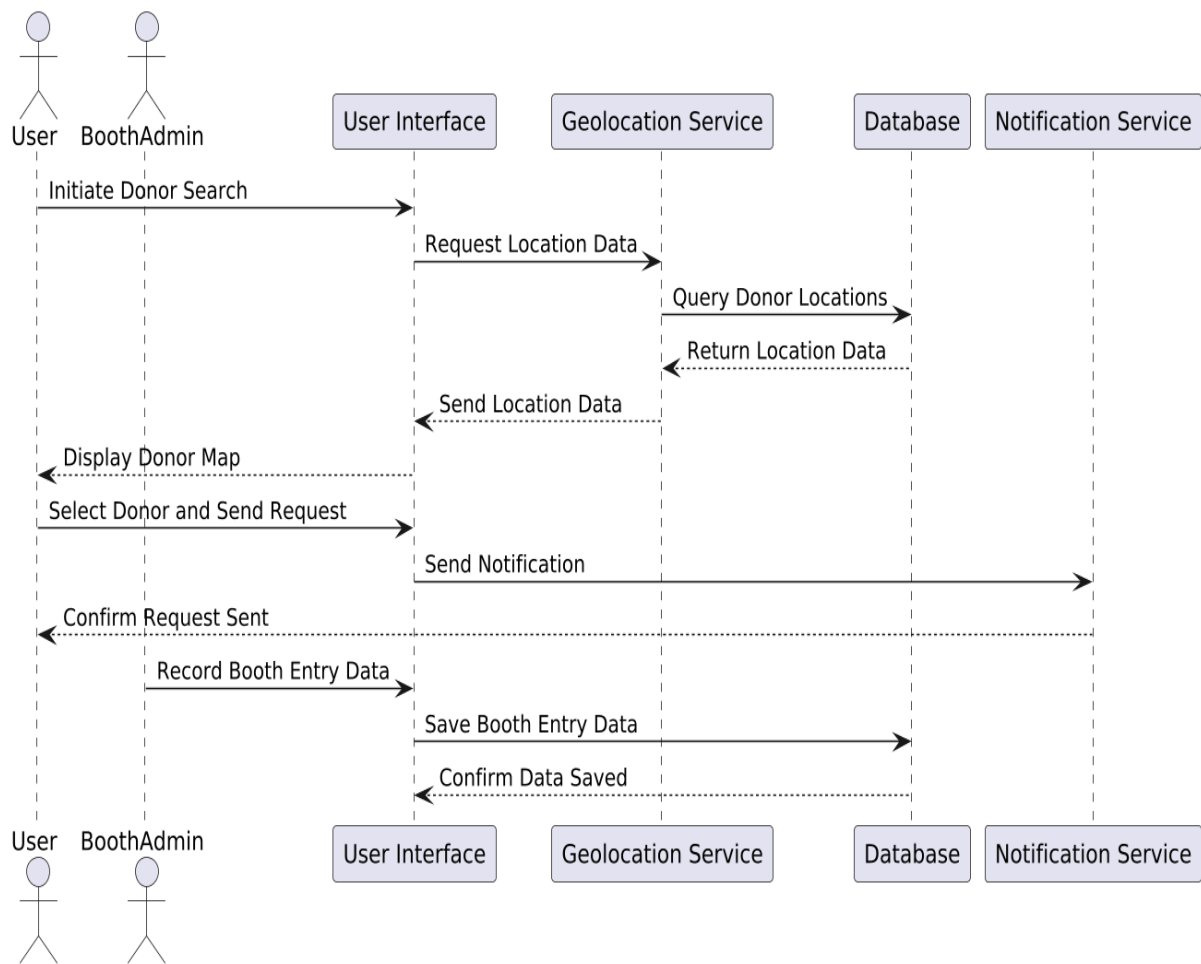


Figure 3.7: Sequence Diagram

This figure 3.7 diagram's a big deal cause it shows the dynamic back and forth between system parts during a donor search like watching API calls in real time It includes the Booth Admin's role in logging booth data post donation It's like a log file that tracks every interaction What It Shows A user kicks off a donor search through the UI which talks to the geolocation service and database to pull donor locations Results pop up on a map and the user can send a request to a donor triggering a notification If the donation's at a booth the Booth Admin records the data and saves it to the database It's like watching packets bounce between servers in a network all smooth and coordinated.

3.1.5 UI Design

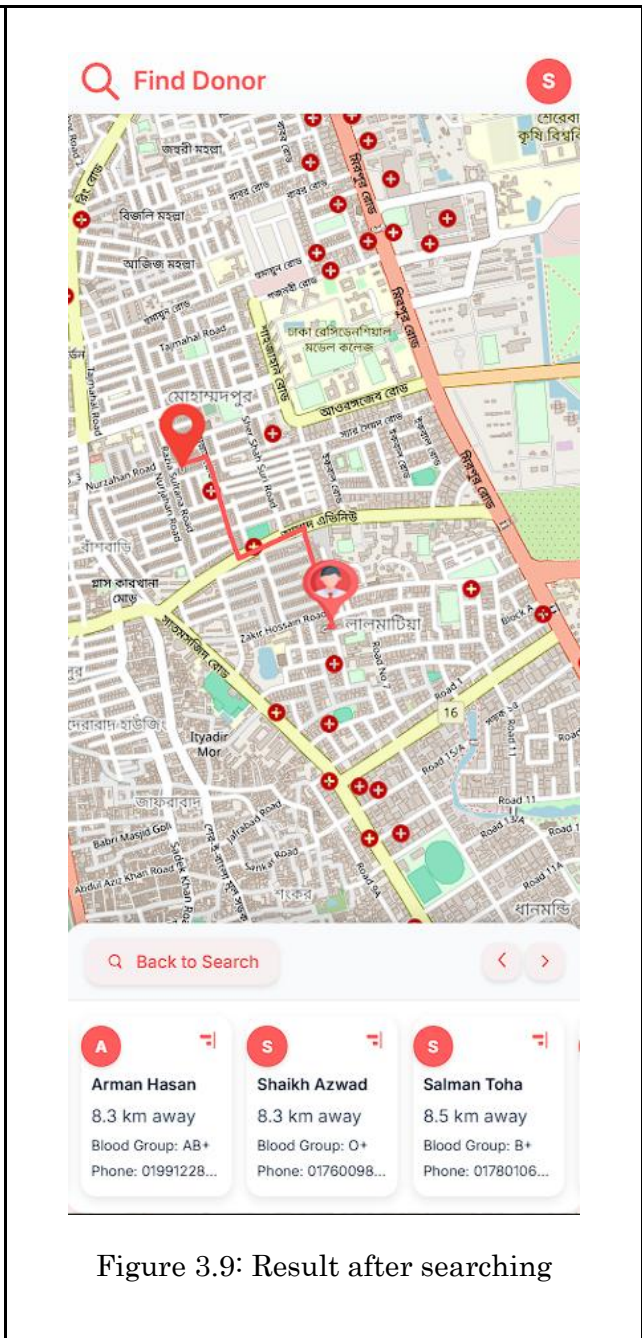
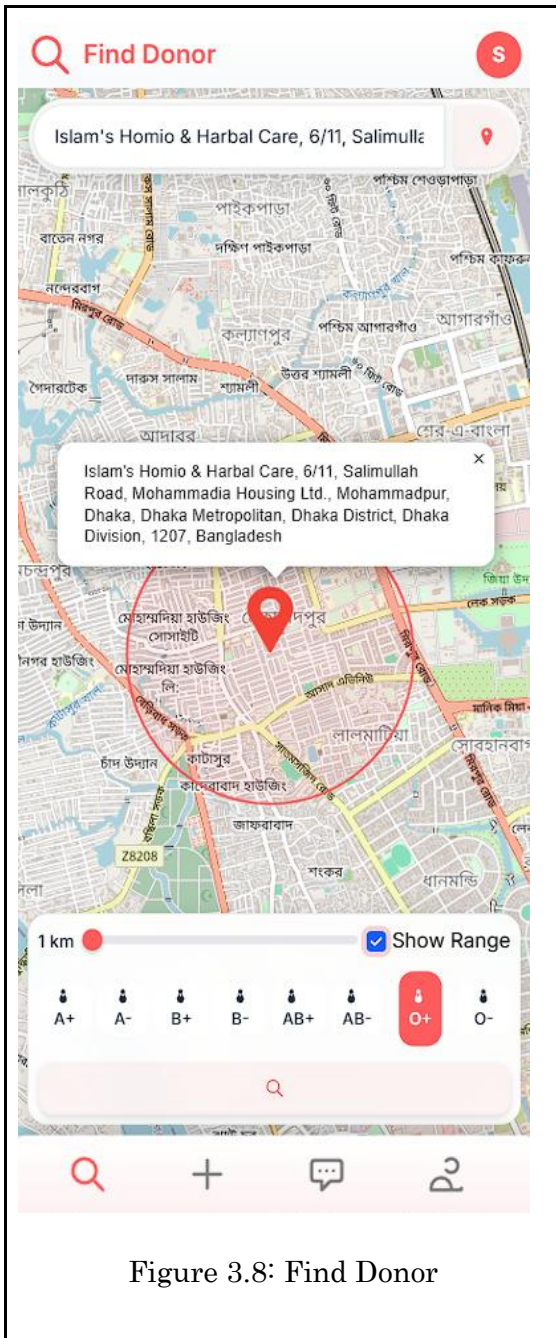


Figure 3.8: Shows the interface when user try to find donor with specific blood group and customizable location. Figure 3.9: Shows available user's result after searching for blood with map routing of the user's.

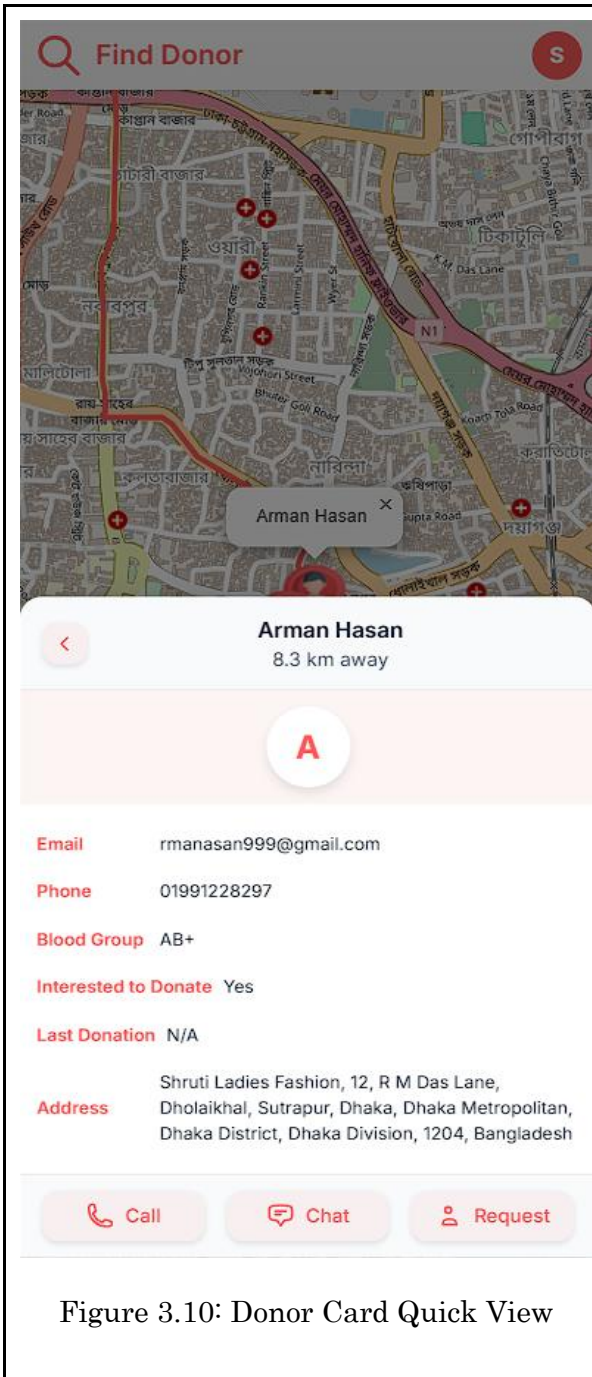


Figure 3.10: Donor Card Quick View

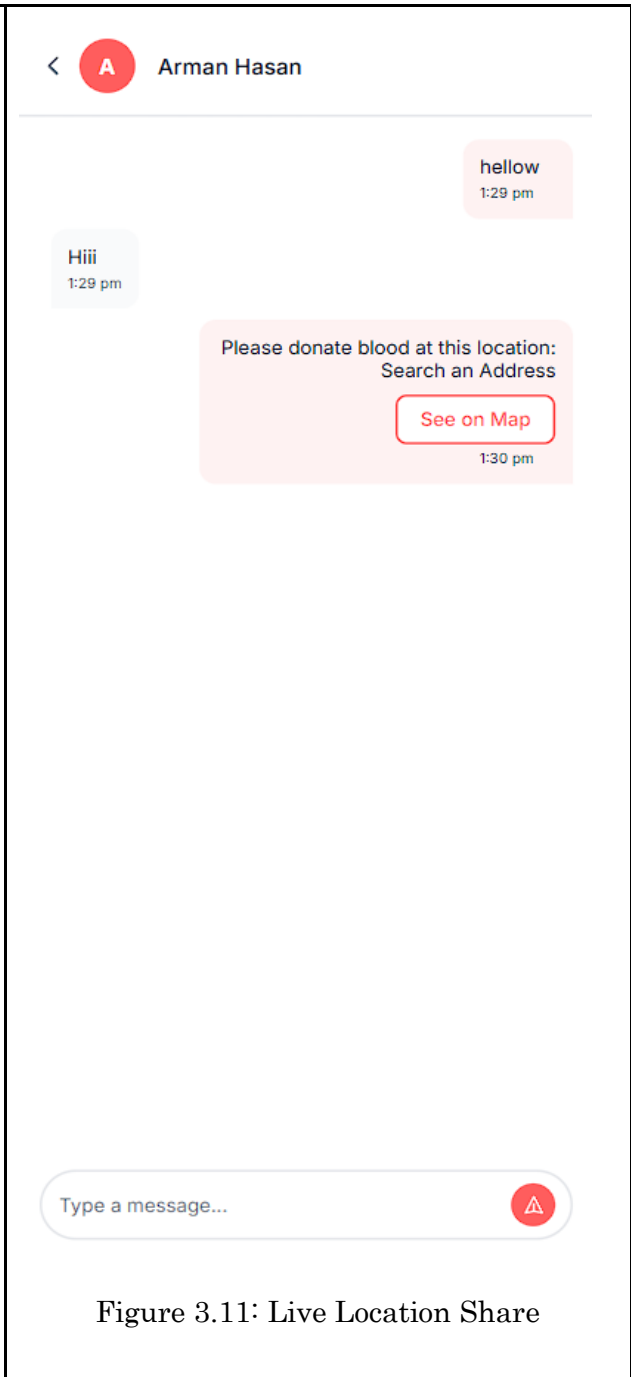


Figure 3.11: Live Location Share

Figure 3.10: Shows the interface of the donor's information (email, phone number, blood group, address and more). Figure 3.11: Shows users live location through system chat.

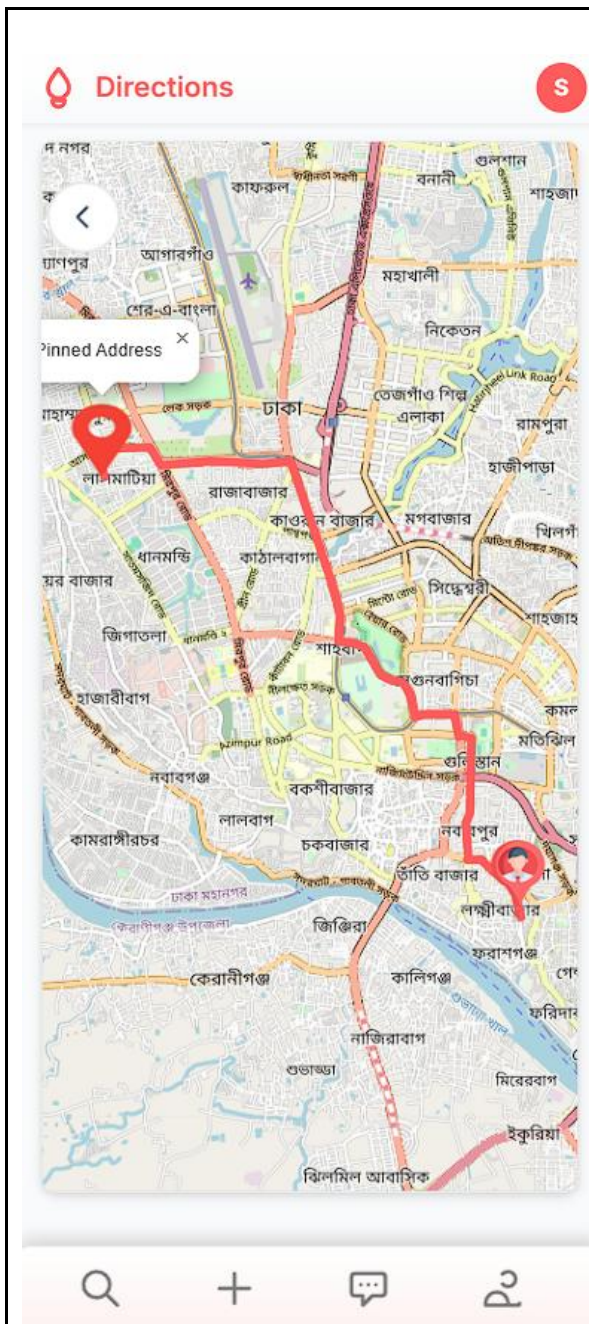


Figure 3.12: Direction sharing on map

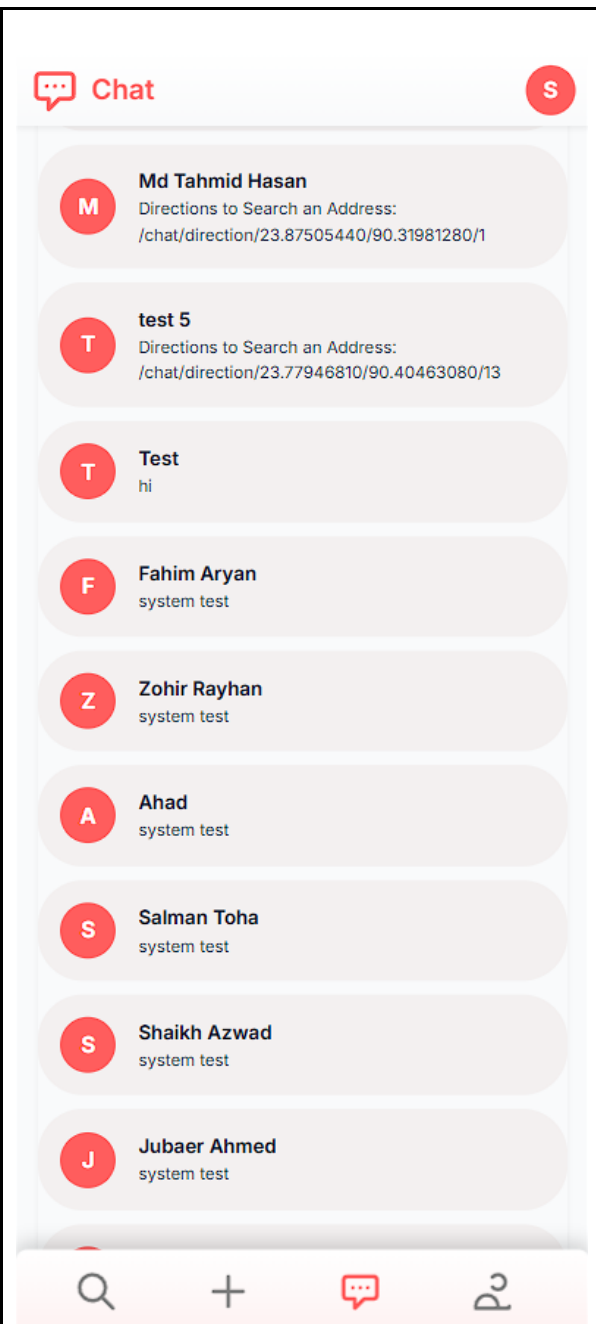


Figure 3.13: Chat with other users

Figure 3.12: Shows the direction on map between donor and recipient. Figure 3.13: Shows the chatting interface with other users on system.

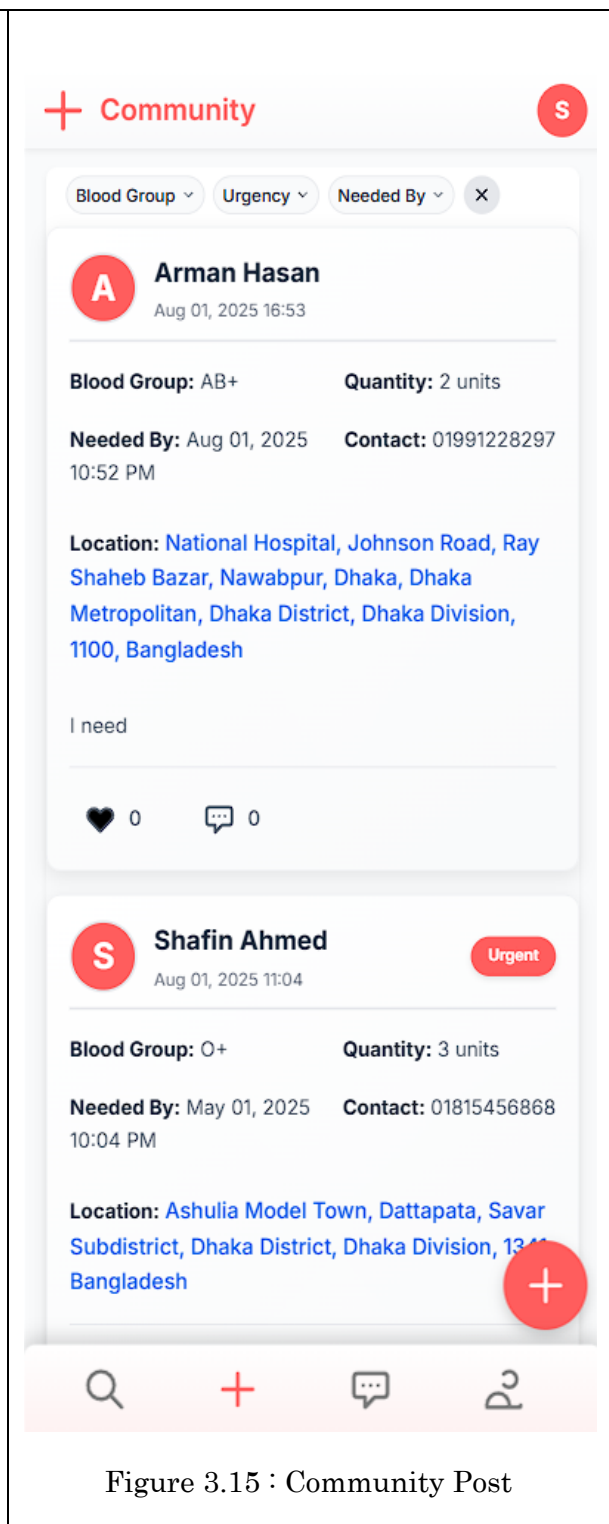
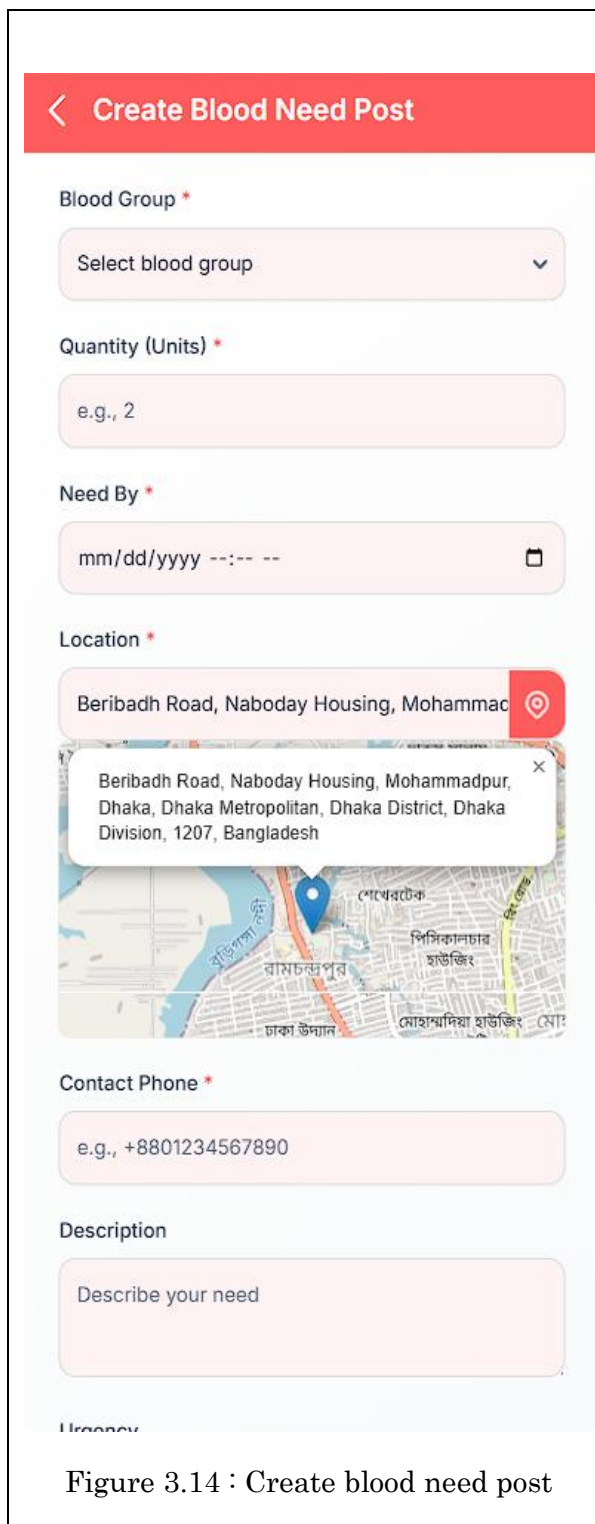


Figure 3.14: Here blood seekers can create post for the blood they need with address and Blood Group, Quantity and need type and urgency level. Figure 3.15: Shows the entire interface of the feed of community posts of blood need.

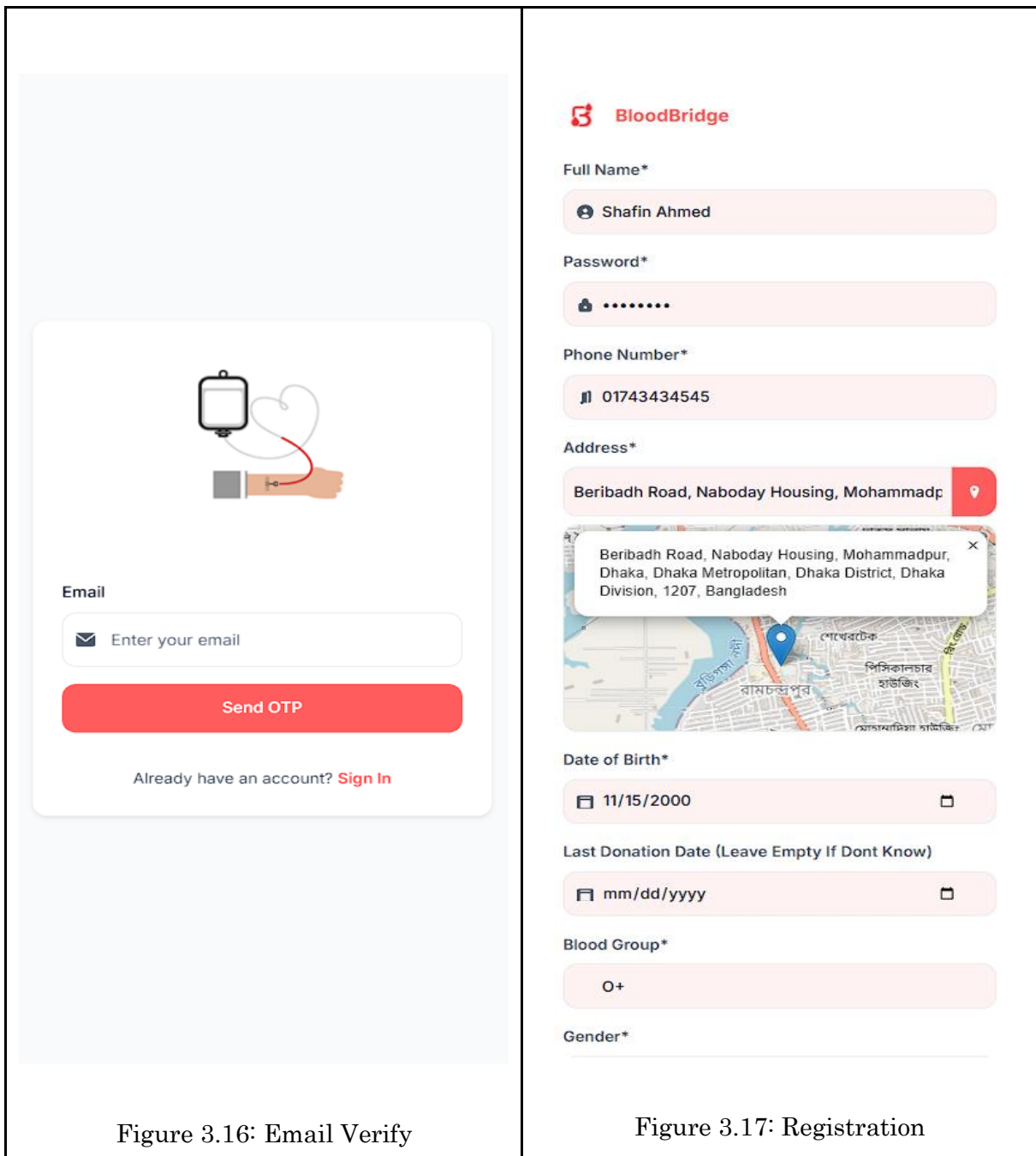


Figure 3.16: Shows the interface of email verification when user try to register. Figure 3.17: Shows the entire interface of the registration process.

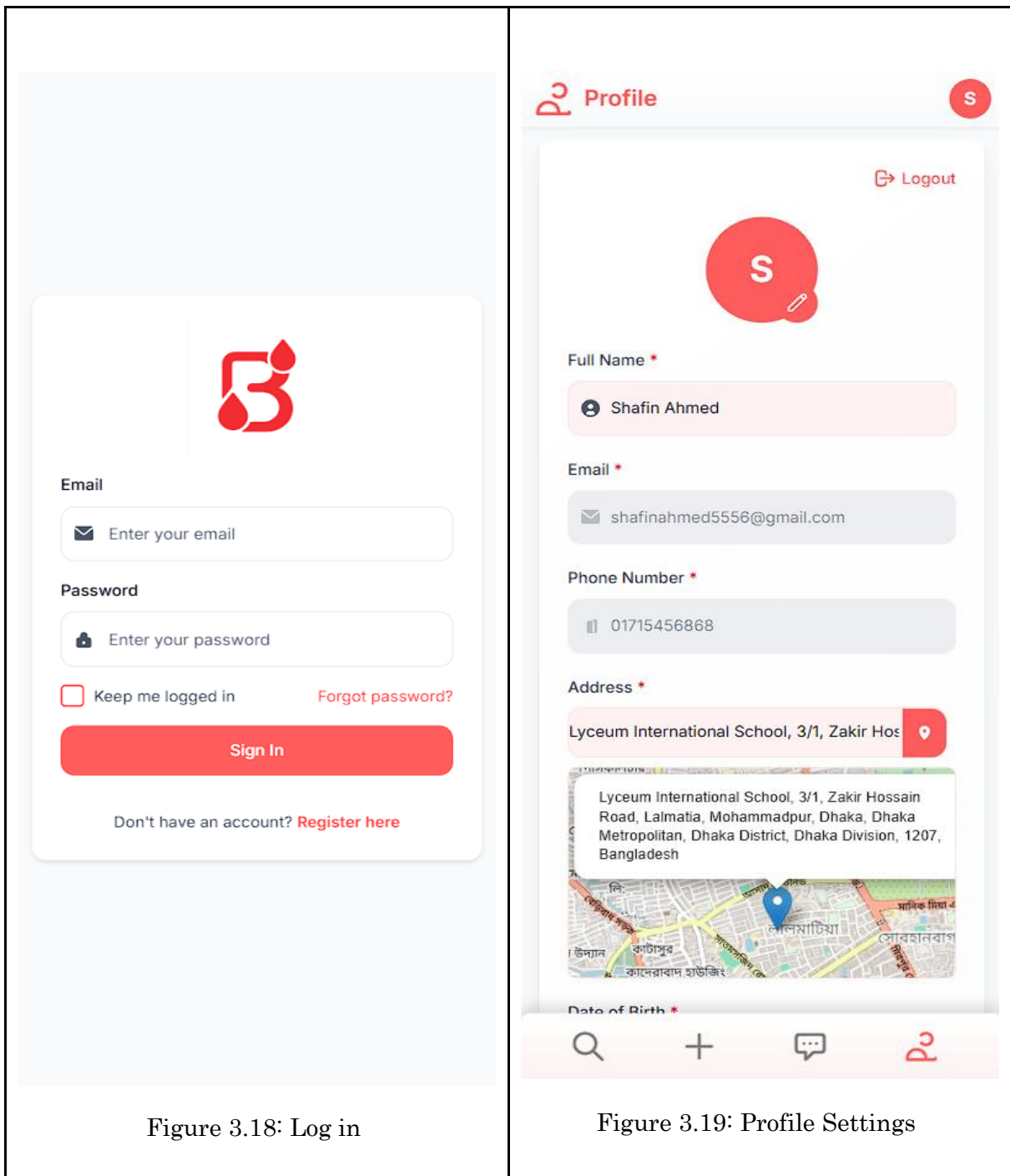


Figure 3.18: Log in

Figure 3.19: Profile Settings

Figure 3.18: Shows the interface when user try to log in. Figure 3.19: Shows the interface of user's profile settings.

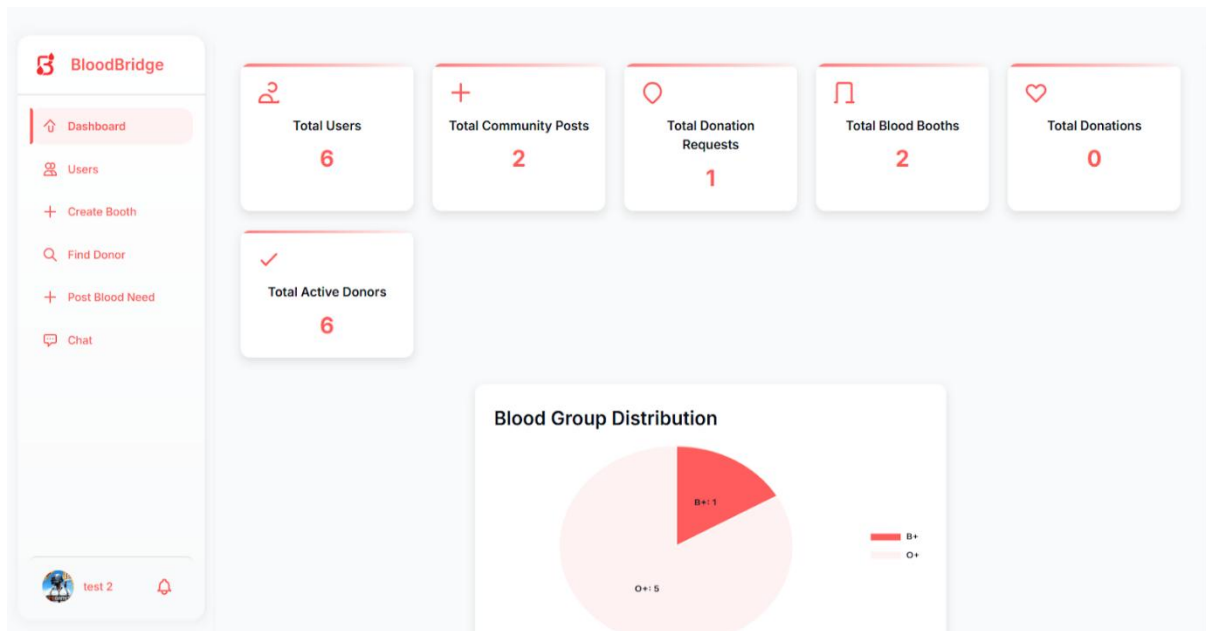


Figure 3.20: Super Admin Dashboard

Figure 3.20: Shows super admin monitoring total users activity.

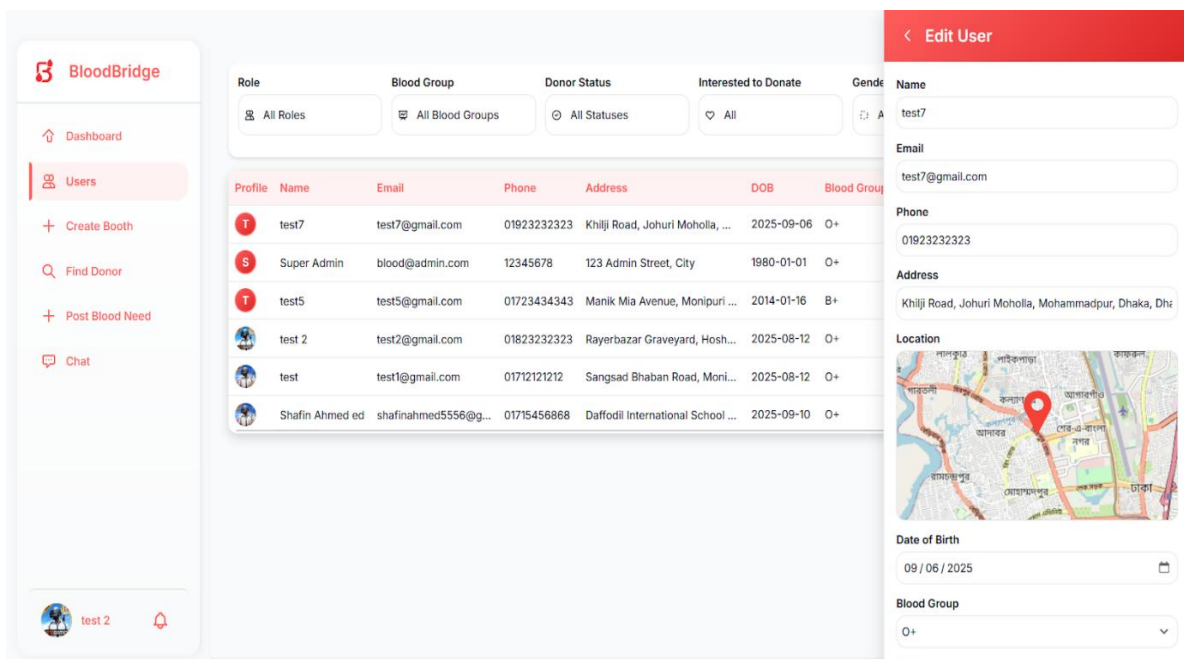


Figure 3.21: Super Admin User Management

Figure 3.21: Shows super admin managing users.

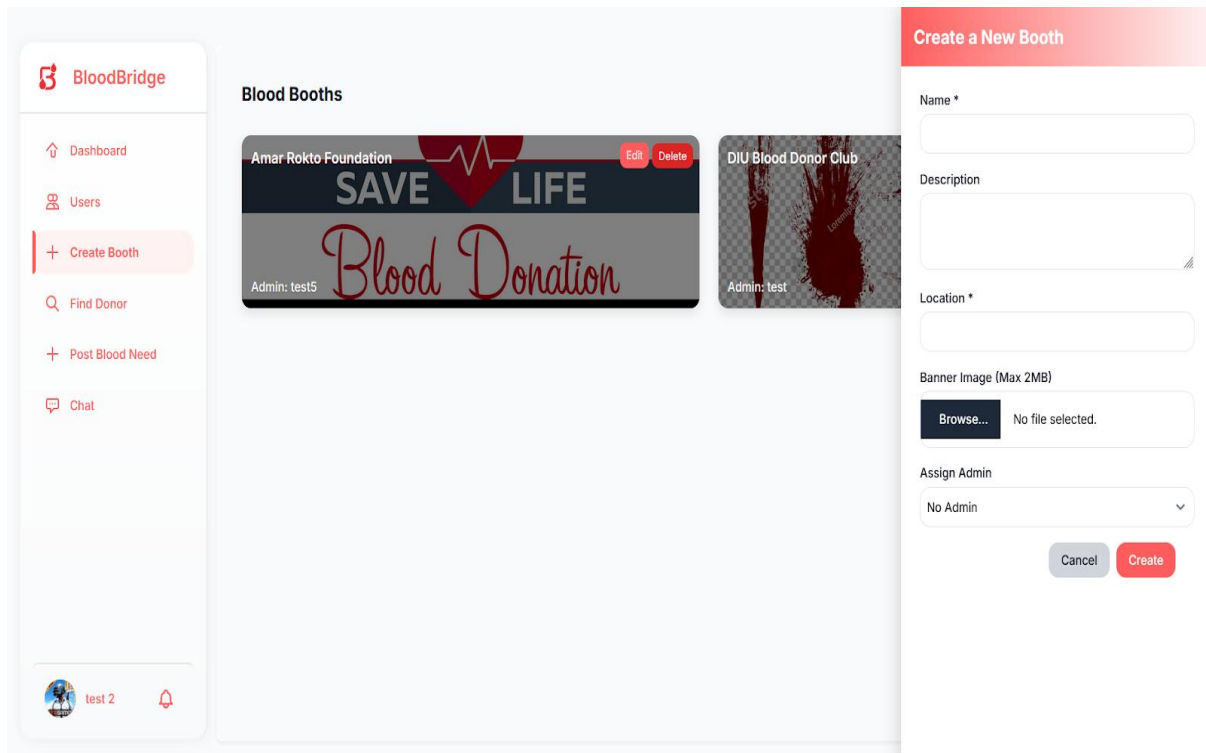


Figure 3.22: Super Admin Blood Booth Create

Figure 3.22: Shows super admin creating new blood booth.

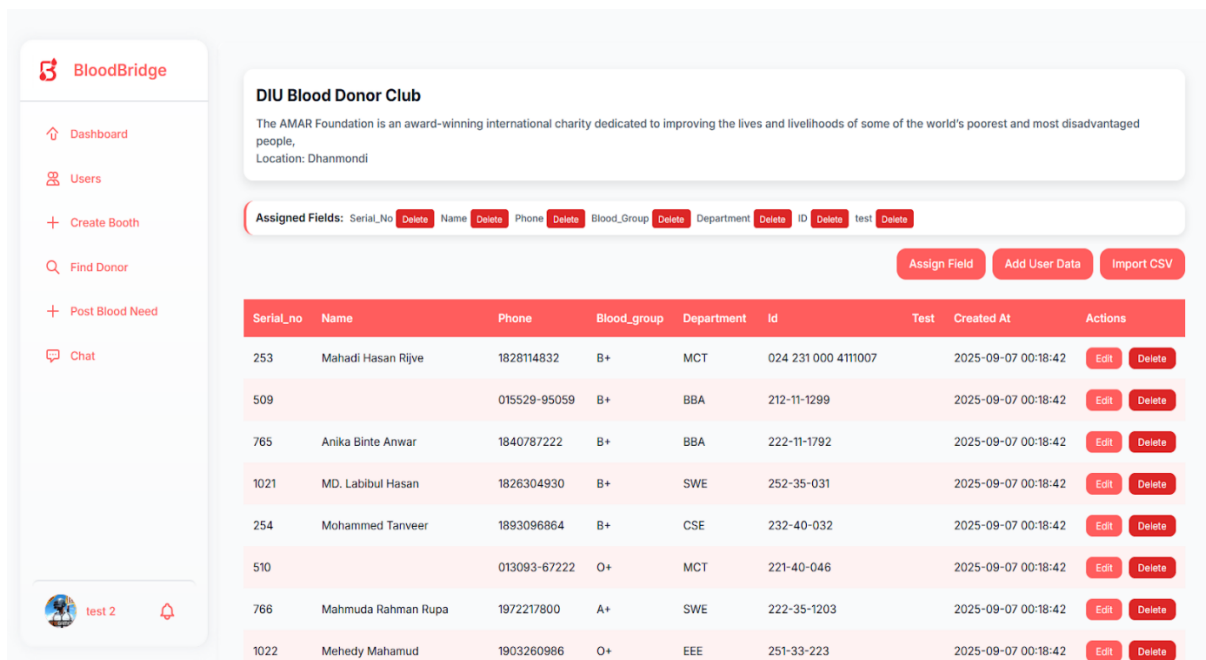


Figure 3.23: Super Admin Blood Booth Management

Figure 3.23: Shows super admin managing all available the blood booths.

3.2 Detailed Methodology and Design

So when we were piecing together Blood Bridge we had a bunch of options for the core features and let me tell you it was. This section dives into those choices with more details in the design decisions but here's the gist of how we tackled some big pieces of the puzzle. It's like building a system from scratch making sure every module clicks into place. First up let's talk geospatial stuff we needed a way to find donors based on location because that's the heart of Blood Bridge matching people fast in emergencies. One option was using third party geospatial APIs like Google Maps which are super powerful for things like distance calculations and geocoding They're slick no doubt but here's the catch they often come with usage fees or rate limits that can choke scalability and jack up costs It's like running a program on a paid cloud service that throttles you when traffic spikes. Not ideal for a system that needs to stay lean and affordable especially in a place like Bangladesh. Another idea was pre computed distance matrices. The kind where you calculate distances between fixed locations ahead of time for super fast lookups It's like caching every possible query result Sounds great right. But it falls apart when your users donors and recipients are moving around, can't predict where people will be and with tons of potential donors it's just not practical It'd be like trying to hardcode every possible user interaction in a dynamic app Total chaos. We rolled our own geospatial algorithm in Flask using MySQL's Spatialized Extensions It's like writing a custom function to handle the heavy lifting ourselves Sure it's more work upfront kind like manually optimizing a database query but it's a game changer We get fine grained control over data privacy no relying on external services and it's low cost for handling dynamic location queries The algorithm calculates spherical distances between two points in coordinate space think of it like plotting a quick path in a 3D game world This lets us filter potential donors within a specific radius fast like finding the nearest node in a graph It's robust scalable and honestly kind nice to see in action. Now let's talk about communication because getting donors and recipients to connect in real time is critical We wanted in app chat that feels seamless like texting a friend but secure and reliable We went with a Web Socket powered solution using Flask Socket IO It's like setting up a live data stream for users to chat instantly keeping them in the app's ecosystem For urgent cases like when someone needs blood ASAP we integrated Twilio for voice calls and SMS It's like adding a direct hotline for emergencies ensuring users can tap to respond without jumping through hoops This mix gives us the best of both worlds interactive in app chats and trustworthy direct communication when it counts. It's been effort to balance but seeing users connect instantly. That is like watching a perfectly

timed API call return the right data. On the database side we picked MySQL and let me tell you it's like choosing a solid server to host your app The relational structure is perfect for managing interconnected data users donations requests all tied together neatly It uses the DB storage engine to ensure compliance which is just a fancy way of saying the database stays consistent no matter what It's like having a backup system that never lets you lose a transaction MySQL's Spatial Extensions also handle our proximity searches without needing a full blown geospatial DBMS which would've been overkill like using a supercomputer to run a simple script This setup keeps things efficient and lets us focus on what matters getting donors to recipients fast. Then there's user authentication and authorization We needed something secure but not so complex it slows us down One option was rolling our own session management from scratch but honestly that's like writing your own encryption library error prone and risky Instead we went with Flask Login for handling user sessions It's like a prebuilt module that's secure and easy to plug in It manages user IDs and sessions cleanly and we paired it with bcrypt for password hashing so no plain text passwords are ever stored It's like locking sensitive data in a vault PyJWT came in for JSON Web Tokens to secure API access especially for mobile or external clients It's stateless scalable and perfect for keeping things tight It's like setting up a secure API endpoint that only trusted users can hit. Looking back choosing these solutions was like debugging a complex system, try different approaches test them see what crashes and what runs smoothly We could have gone the easy route with external APIs or manual setups but those would not scale or give us the control we needed Blood Bridge is about reliability and impact so we picked tools that balance performance cost and user trust.

3.3 Project Plan

The progress based on maturing gateways and development targets for Blood Bridge. The overall work scheme should be well ordered and schedule planned. Activities: Stakeholders interviews, survey, literature research and competitive analysis, requirements elicitation (Functional and Non functional) and initial system architecture design. Development & Prototype: We've completed the following so far, Full database design UI/UX Wireframes and mockups API design Geospatial algorithm design, Finalizing Tech stack early prototype development. Implementation and Development: Visual interface interaction development. Backend development (API endpoints business logic database integration). Development of Core features (donor search/messaging/notification system) and admin panel. Strategy and Quality Assurance: Unit testing,

integration testing, system testing, user acceptance testing (UAT), performance testing, security testing and bug fixing. Worked for code execution verification and also provided for error handling. Compiled test reports in excel and word formats Performed system testing, performance testing, compatibility testing, and software quality assurance testing to create a quality control log. Upon conclusion of test, made suggestions for enhancing documentation, as well as, training and changes to applications as appropriate. Structured test plans, test reports and perf metrics.

3.4 Task Allocation

Table 3.1: Project Timeline and Task Allocation

Tasks	Weeks																		
	1 2	1 4	1 6	1 8	2 0	2 2	2 4	2 6	2 8	3 0	3 2	3 4	3 6	3 8	4 0	4 2	4 4	4 6	4 8
Data collection	Blue	Green																	
System Design & Architecture		Blue	Blue																
Database Development					Blue	Blue	Blue												
Frontend Development									Blue	Blue									
Backend Development										Blue	Blue	Blue							
Core Feature Implementation												Blue	Blue	Blue					
Panel Development														Blue	Blue				
Deployment & Optimization																		Blue	Blue

This table 3.1 depicts the timeline of the principal activities in each period of our project, from week 12 to week 48.

3.5 Summary

This part explains the research methodology that is followed in the Blood Bridge project which consists of complete requirement analysis and design specification. The system design consists of responsive frontend, API capable flask backend and a MYSQL database which takes cares of scalability, security and user experience. The long list of function requirements and non function requirements will help in a clear structuring of the realization and gives an idea of how the information exchange in the system will look like. The UI design depicts the commitment towards a user friendly and accessible interface.

Chapter 4

Implementation and Results

This chapter presents the practical operation of the Blood Bridge platform, highlighting the environment, the adopted technologies and the main platform functionalities offered. Finally, the results obtained throughout the different development and testing stages are presented, showing the obtained outcomes and the usefulness of the Blood Bridge system in achieving the project aims.

4.1 Environment Setup

The Blood Bridge platform was developed in a well thought environment, considered to be compatible, efficient and scalable created from zero. Some of the characteristics of the environment where the platform was developed and operated are. Operating System: We selected Ubuntu 22.04 LTS (Long Term Support) because it is very stable, reliable and has large, active and supportive community and software repositories. This makes something like this a good base for development and production use. Programming Language: The backend logic was implemented in Python 3.11.0rc1 as the predominant programming language. A lot of this speed, in large part, is thanks to Python's libraries, how powerful it is and how it's programmatically readable to humans. Nouveau was developed with the support of the following fundamental Python packages: Python Flask: A small framework to create a backend for web applications, includes support for routing, request handling, response handling etc. Flask Login: For managing user session, enabling secure and user friendly authentication and authorization. Bcrypt: To cryptographically hash users password and not store it in clear text. PyJWT: For JWT authentication, which is a way to secure the API and handle stateless authentication. Flask-Caching: A choice to improve efficiency by using caches for frequently requested material, specifically the donor search queries. Openpyxl and Pandas: for all things excel, particularly import and export of donor information as well as many of the admin reports. Database/Storage: We used MySQL 8.0 as a Relational Database Management System (RDBMS) to store all the application data. The high level of features as well as the ability of the system to grow and to handle spatial data types, were both critically important to the success of the implementation of the donor search by proximity. The design of the database was developed so as to prevent the repetition of data & for implementing referential integrity (foreign key relationship) among different tables like

user, donor, blood request, donation history & administrator log. Frontend Technologies: The UI has been developed using standard WWW technologies to enable wide compatibility and make the user experience great HTML5: For formatting the html content of the web pages. CSS3: Page styling using tailwind CSS, Flowbite utility, Make the UI construction very fast and scalable layout as well as responsive layout. Flowbite components were tied up to give us interactive UI elements ready to use in this flow. JavaScript: For interactivity in the front end, creating and modifying reactive contents, and form validation. Jinja2: A template engine that is integrated with Flask in order to generate dynamic content on HTML, mixing in the backend data into the front end. Version Control: Version Control Git was to track changes in the codebase as well as in enabling the team to maintain and work together in the project through a remote repository stored on GitHub. Instruments for Development: Visual Studio Code: Primary Integrated Development Environment (IDE) to code, debug, and manage the project with. Postman: For collecting API end points and inspect the relationship of data being shared between Frontend and backend. SMTP Server: to send email Notification for user (User Registration conformation operation, Blood Request operation united hospital).

4.2 Testing and Evaluation

A complete program of testing and evaluation was initiated during development of the product to demonstrate the reliability, performance and quality of the blood bridge system. This was not only unit and integration testing, it was usability, performance and security testing. Database CRUD: Create, Read, Update and Delete Operations Validated Request, Donor and User profile properties for consistent data and Referenced it in MySQL Database. services. Verify of the Geospatial Algorithm: Track that the distance calculation is accurate and the proximity search method is effective and Is it the correct donor set being generated within the radius. Integration Testing: We tested whether the different components of the application (frontend and backend) were able to communicate smoothly with other parts (database and external services). Communication between Frontend and Backend: Ensuring that data went until the frontend forms and after to the backend correctly and that there was not unexpected data which decides to show off in the frontend of the system. Usability Testing: We have tested the first prototype and the live module with preliminary users. Responses were collected regarding How intuitive, navigable and experienced the interface was. Performance: Performance tests have been done to test how strongly the system performs as a result of certain business operations (donor search here). Measured metrics by

response times, db query times, server machines use. Donor Search Performance: With the use of Flask Caching for caching and optimized spatial MySQL queries, search time for nearby donors was significantly reduced to 2 to 3 seconds on average for test data set with few many donors. DB Optimization: Always keep on the DB performance tuning and appropriate indexing it is must. In order to achieve no trouble making and scale up with user data increase. Input sanitization: We have robust input sanitization on all end user provided data to eliminate common threats like SQL injection and X-Site Scripting (XSS).

4.3 Results and Discussion

Let's start with the user experience because it is the most important. We designed the UI/UX from the ground up leaning on Tailwind CSS and Flowbite to make it clean intuitive and honestly kind good. We took user feedback seriously like running a beta test to debug usability issues and it helped us craft an interface that is easy to navigate even under pressure Signing up is a breeze users just submit their details and they are in After registration they can log in to unlock more features like searching for donors or posting requests. We added email verification to keep things tight security wise which makes users feel confident their data is safe The search results screen is clear showing donors by blood type and location and the blood request feature is designed to be super straight forward. This setup ensures users can operate the system even in a high stress emergency which is exactly what we were aiming for. On the backend we went with Flask and MySQL and it's like building a server with a rock solid foundation The database schema was designed from scratch to handle all sorts of data donor profiles donation histories blood requests and admin tasks all neatly organized like a well structured relational database It's efficient at sorting and unifying these different entities so we can pull up critical info fast MySQL's DB engine ensures compliance which is just a fancy way of saying the data stays consistent no matter how many users are hitting the system at once It's like having a failover system that keeps everything running smoothly Rapid data retrieval is key for real time processing and our setup delivers It's been effort full to get this right but seeing queries return in milliseconds. That is the star of the show We built a custom geospatial algorithm using MySQL's Spatial Extensions and it's been rigorously tested to find donors within a specific radius say 20 km based on blood type and location It's like running a spatial query to filter nodes in a graph with pinpoint accuracy We call it searching within the boxed area which is our way of saying it locks onto a geographic region and finds the right donors fast .This is the heart of Blood Bridge

saving time and effort in emergencies by connecting people hyper locally Imagine someone needs A+ blood in Dhaka our algorithm crunches the coordinates and delivers a list of nearby donors in seconds It's like a GPS for lifesaving matches and it's proven the platform's feasibility for real world use. The community post feature is another gem. Users can post urgent blood needs with details like blood group or hospital location and it's like pinning a message to a digital bulletin board It builds this sense of community obligation where people see the need and jump in to help .It's a social feature that boosts interaction and makes critical info spread fast kind like a viral post but for a good cause We saw in testing that these posts get quick responses which is huge for emergencies It's like broadcasting a signal that gets picked up by the right people at the right time. And let's not forget the admin features We planned out a robust set of tools for managing the platform Booth admins can enter donor data and track blood stock while super admins have a bird's eye view verifying system status and keeping everything in check It's like having a control panel for a complex system with dashboards to monitor users data and operations These features are critical for keeping Blood Bridge running smoothly ensuring data is accurate and the platform is reliable. Building this refactor test again We faced challenges like balancing performance with usability but every time felt like debugging a tricky loop Seeing it all come together users registering searching for donors posting needs it is this could actually save lives by making blood donation faster and more connected Blood Bridge is just the start and this prototype shows it's got the potential to scale and make a real impact.

4.4 Summary

This chapter fully explains the continuation of the Blood Bridge project in the implementation part including a complete coverage of the setting up of the technical environment required for implementation, the technologies used, the methodology for the testing and evaluation.

Chapter 5

Engineering Standards and Design Challenges

In this chapter the compliance with the Blood Bridge project is shown, the most important design challenges that we faced while developing the project are highlighted. Will focus on the ways we solved and overcome these challenges in order to provide the solidity, reliability and ethically of the platform.

5.1 Compliance with the Standards

When we got Blood Bridge up and running we did not just done some code making sure this project is stable like a well tested program robust as well capable of surviving the real world. Means this is not about the functionality at all, this is about making sure this is the right functionality for that might really need it, this part is going break down our approach in regards to software, hardware and the ways all these different parts of the project communicate to keep Blood Bridge safe secure and ready to run.

5.1.1 Software Standards

The best place for us to start is software standards because that's sort of at the heart of BB we took inspiration from this here ISO/IEC 25010 It's like a super detailed quality model for software and it covers a whole bunch of areas from functionality to performance and security It's like having a checklist for making sure your code running well not just running We also borrowed a little from the world of Quality of Service like except we use the principles to prioritized the main tasks finding a bloody donor quick It's sort like speeding up a network so it pushes through all the most urgent packets first. On functionality level we literally went all guns as in we had piles of requirement docs, ran UATs to ensure every feature hits the mark Like we didn't want a single button to misfire It was like coding a function and testing it a 100 times to make sure it does exactly what a user would want Whether it is someone signing up or posting a blood need we killed to make sure it works straight out the gate Getting that right took some serious effort full of hard. Not just a case of the speed that keeps the ship afloat, even when you had as many users as using the numbers at the same time think of a user pinging the app at the same time during a crisis our system doesn't buckle. It would be like adding more

RAM to our server so it can handle more traffic without the server buckling. Security is a thing We locked down the app with Flask Login for managing sessions bcrypt for hashing passwords so there's none of that plain text nonsense and PyJWT for securing the API access It's like building a vault around all the users data We thought about broken authentication and made sure those sessions are as locked down as a drum to shut the front door on XSS input we're sanitizing and output we're escaping like we put up a firewall against creepy javascripts. A little OCD, but it all helps. We did dangle the QA hammer in front of (which has not been tempted to rush to meet a deadline) recipe for disaster Well we did have a type least QA process that QA process we could, but not violate ISO/IEC 25010 principles It's not about the face certificate program, but about having an assured system that can scale maintainability and reliability The standards were like a stress test for the to cope with the wild unknown It meant the key use cases like searching for donors or posting requests were hard ironed, with hardly any bugs The UI and its contract and transactions were speedy It's like saying you know your app will stand by you when you Need It The Most. We did dangle the possibility of just trusting the default server security, or a manual code review Those are seductive, but risky It's like relying on the little lock on your safe you know better than that. By creating standards such as OWASP, we could ensure that we had a secure way of catching sneak attacks by the bad guys It's been a lot of work, but to have Blood Bridge hold its ground to potential sneak attacks.

5.1.2 Hardware Standards

We went with enterprise grade hardware so that we met those non functional requirements for availability and performance. It's not an official thing but it wouldn't be right not to do it. We did it so that Blood Bridge would be running thinks real time to had our drop in it and because I have seen real time happening on a average box setup like old laptop maybe but it's not going to be able to work under a hundred simultaneous connections. That's not a lot of connections when there are lives on the line. What is important though is that animated running and that quick response because I can give you that hardware setup that allows it.

5.1.3 Communication Standards

Communication standards matter. You're sending sensitive data and a system like Blood Bridge needs to ensure that only authorized parties can access certain information like logins or blood requests. We chose to use HTTPS for all client-server communications. All data is encrypted in a tunnel so that only you and me can access. On the other hand, using plain HTTP is like sending messages on postcards. Anyone can intercept your communication and misuse it. We explored writing our own encryption. But, building our own custom security standard from scratch is a terrible idea. TLS/SSL is the industry standard because of its security, ease of use, and to ensure that other parties are trusting your data. For emails that have similar user, like sending account verifications or warnings for emergency blood requests, we continued to use SMTP since all providers offered it and it was simpler to use. We looked into using proprietary email APIs, but didn't want to be locked into a single provider that may require custom code for our platform. We needed to be able to develop for multiple platforms so that we can support multiple use cases that we have today and beyond. SMTP didn't interfere with sending emails for account confirmations or warnings for blood donations. We chose these standards for more than just following the rules It's about creating a system others can trust mechanisms are what ensure your data are secure and available They are like the pipes of the system As CSE students we have learned that picking the right standards is much like choosing the right libraries for a project Get it right and everything just works Get it wrong and you're stuck figuring out why it is still broken If we could just see it all come together as smooth as that.

5.2 Impact on Society, Environment and Sustainability

5.2.1 Impact on Life

Blood Bridge is all about taking that game changing potential and concentrating it into life or death moments Finding a blood donor quickly when the situation is desperate is essential and Burge and his colleagues claim their platform makes the process a whole lot faster It's a bit like fine tuning a search page to return results tens of times as fast precisely when seconds matter The apps geolocation search enables users to find a potential donor nearby say within a 20 KM radius via blood type and availability but the real time peer to peer communication tools like in app chat or SMS alert ensures that requests get answered pronto It's like supercharging the connections in a high speed

network to get emergency transfusions happening at just the right time. But it doesn't all have to be about drops in the bucket Blood Bridge lowers the barriers to making a donation essentially by making it easier for first timers to step up It's like lowering the friction in a user interface there are fewer steps between donator and site goal. By getting more people to donate we're strengthening our national blood supply chain.

5.2.2 Impact on Society & Environment

Now let's discuss the impact of Blood Bridge on society and the environment The blood request hub acts as a digital community board that brings people together to help donate blood When a user puts in an urgent blood need it is no longer a please but a reminder to all that we need to step in and do our bit It's like the social media feed but instead of seeing selfies users are actually saving lives It brings together a sense of being connected between the donor and the recipient joining nodes in a network that makes the community stronger. The platform's built to be a no stress no brainer Someone in your family or you need blood. Blood Bridge takes the panic and pain out of finding a suitable donor It's like a minimalist UI that makes a difficult task less stressful We've seen users who find this to be the case which is a big plus when you're dealing with something as high stakes as medical emergencies It's like having a fully functional app that's not prone to crashing when you need it the most. Blood Bridge is also raising awareness about the ongoing need for blood donations It's like sending a push notification to the public to remind them that patients need blood daily Through collating anonymous data such as blood demand donor demographics and response times we are generating the insights that could help public health planners We're looking at the system from an analytics point of view to optimize resource allocation without knowing anything about anyone such as their blood type. On the sustainability Blood Bridge is also minimizes travel for no reason by having the donors and recipients next to each other they do not have to travel long distance. Blood Bridge is something to save fuel and since we were completely digital we reduce paper waste as well. Every little helps, it feels good to know that our tech can also play its part in a green world.

5.2.3 Ethical Aspects

Ethics play an important role in Blood Bridge and we have decided to incorporate it into the very good wat. It is fair responsible and does the right thing, so we concentrated on privacy and consent and accessibility and transparency issues so that the system respects the person who uses it. Privacy and security is a must User data particularly health data is really sensitive so we have strong protections in place Data is encrypted

at rest and in transit We use flask login for authentication and bcrypt We give our users the ability to control the data that others can see like setting availability or going into snooze mode It's like we give our users a personal dashboard to manage their privacy We are compliant with data protection standards to keep things secure It is a time investment but necessary to build trust. For informed consent, we made everything's clear and straightforward Donors know what they share and what recipients see It's the rare user agreement no one needs a lawyer to navigate not buried in fine print Users can make an informed decision about their data crucial in the ethical management of personal info. Access is also a priority We wanted Blood Bridge to be for anyone no matter where they live or what their income level is Donor matching is entirely medical compatibility and proximity based no bias no favoritism It feels like designing an algorithm that's unbiased and inclusive making sure every person gets access to help they might need. Transparency brings it all together Users are never left in the dark about what the app is doing and how their data is used It's as if there's a debug log open for users to see, so there is no ambiguity about what is going on under the hood as CSE students, we've learned that building trust is as important as building features Seeing Blood Bridge come together with these ethical values.

5.2.4 Sustainability Plan

Easy to Maintenance: Good clean code by followings code standards so that easy to develop or maintenance at time to time and also easy to add any new features. Community participation: Communicate with end users for feedback, there is many ways we can do this such as forums or asking face to face. We did the test so that people can appear do tests with their laptops or mobiles ensuring that we tried to append to our compatibility. Making our project worth to use to use. Collaborations: Tie up with blood bank hospital NGO can help in incorporating Blood Bridge in the current healthcare ecosystem and can increase acceptance, adoption. Affordable Technologies: Incorporates open source platforms (Python, Flask, MySQL, Linux) to reduce licensing expenses.

5.3 Project Management and Financial Analysis

The proper financial analysis is required in order for us to fully grasp how resources are distributed, as well as potential costs to develop, deploy and maintain the Blood Bridge platform. We did this with a 30year view so values are when the costs are worth it to make the most of the work that was done in the Blood Bridge was fundamental. Here is a breakdown of projected costs:

Table 5.1: Project Management and Financial Analysis:

Cost Category	Description	Estimated Cost (in BD Currency per year)
Hosting	OVH rise apache server	5000 BDT
Database	MySQL	0 BDT
Domain	From namecheap.com	220 BDT
Map API	LeafletJs (open source)	0 BDT
Total Cost		5220 BDT

Table 5.1: Shows the cost analysis and project management for Blood Bridge

Table 5.2: Estimated future development budget for Blood Bridge

Category	Estimated Cost (BDT)
Software Developers	30,000–50,000 BDT
UI/UX Designing	5,000–10,000 BDT
Project Managing	7,500–15,000 BDT
Infrastructure & Tools	
Cloud Hosting (Servers, Database, Storage)	1,500 BDT
Domain Registration & SSL Certificate	500–1000 BDT per year
Communication APIs (Twilio, SMTP services)	1000–3000 BDT per month (usage based)
Development Tools & Software Licenses	2000 BDT
Marketing & Outreach	
Initial Awareness Campaigns	2,000–5,000 BDT
Total Estimated Initial Development Cost	50,000–80,000 BDT

Table 5.2: Shows the future estimated budget could be required to make it more enhanced.

It is publicly minded people like ourselves at Blood Bridge who would like to maintain our service as a free service available to donors and patient alike. However, other ways of generating revenues, based on the dependence on different forms of sustainability to finance its operations are: Current financing: We are running to get sponsors, grants from government and private, public health and social good tech orgs, etc. collaboration work with corporates who are incorporating to these programmers and who could also finance the platform as part of their CSR strategy. Institutional Subscription Services: Services to the banks, the hospitals, or the health corporations. These could include Enhanced monitoring and analysis of blood supply and demand trends. Integration with hospital management systems. Advertising: Could offer highly relevant and unobtrusive advertising from medical suppliers or other health services, all ethically curated to not interfere with the user experience.

5.4 Complex Engineering Problem

5.4.1 Complex Problem Solving

Table 5.3: Mapping with Complex Engineering Problem.

EP1	EP2	EP3	EP4	EP5	EP6	EP7
Dept of Knowledge	Range Of Conflicting Requirements	Depth of Analysis	Familiarity of Issues	Extent of Applicable Codes	Extent Of Stakeholder Involvement	Interdependence
✓				✓	✓	✓

Table 5.3: Describes which complex engineering problems we have achieved.

EP1: Depth of knowledge this one's a pulls from a crazy range of tech skills think Flask and Flutter for web and mobile MySQL spatial extensions and Leaflet JS for location magic Flask Login PyJWT bcrypt for locking down security Flask SocketIO and Twilio for real time chats and calls and MySQL with Firebase for data handling It's like opening my entire CSE toolkit and using every single tool The project's got web dev geospatial stuff security comms database management all tied up tight EP1's totally in the list.

EP5: Extent of Applicable Codes this one's matched solid i leaned into big standards like ISO/IEC 25010 for quality and OWASP Top 10 for security with defenses against SQL

injection XSS attacks and secure HTTPS SMTP communication It's like building a digital vault with every lock in place Blood Bridge sticks to these standards like perfectly.

EP6: Extent of Stakeholder Involvement This one's fully matched no doubt I have got donors needing quick access requesters needing fast matches admins needing control tools and healthcare groups needing integration It's like juggling multiple threads in a program and making sure they all sync up perfectly my system diagrams and descriptions show you've got every stakeholder covered EP6's a done deal.

EP7: Interdependence it completely The Flask backend MySQL database Leaflet JS APIs Twilio comms they all work together like a perfectly tuned system Donor matching data security real time messaging every piece depends on the others It's like a codebase where one bug could crash everything but I have got it running smooth EP7's locked in.

Mapping with Knowledge Profile

This whole K mess ties into a knowledge profile that describes how we drew on different areas. K1 Natural science had donor matching and system uptime echoing like a server that never shuts down. K2 Mathematics was the driver of our geospatial algorithms and priority notifications like solving for an exact math match. K3 Engineering principles informed our modular client server and database systems for scaling up. K4 Interactions meant mastering MySQL spatial extensions Firebase and Twilio for real time. K5 Engineering design kept our app user friendly with Tailwind CSS and Flowbite. K6 Engineering practice included Git for version control and testing units and usability plus deployment on Ubuntu. K7 Troubleshooting squashed bugs by doing the opposite of inaccurate geolocation. K8 Research literature of blood donation and geospatial tech made us mindful of our choices so we filled gaps in existing systems It's like building on top of a strong base to make something new and cool.

Table 5.4: Mapping with knowledge Profile.

K1	K2	K3	K4	K5	K6	K7	K8
Natural Science	Mathematics	Engineering Fundamentals	Specialist Knowledge	Engineering Design	Engineering Practice	Comprehension	Research Literature
✓		✓		✓	✓		✓

Table 5.4: Shows the knowledge profile mapping with our project.

K1: Natural Science This one's rock solid Blood Bridge uses computation and process flow principles to keep donor matching and system uptime super reliable like a server that's always on for critical healthcare stuff.

K3: Engineering Fundamentals Totally matched You went with modular designs client server setups and a database schema that's built to last and scale It's like coding with clean reusable functions that you can build on later Your project's structure screams K3 and it's all there in black and white.

K5: Engineering Design This one's a full match The system's modular with Tailwind CSS and Flowbite for a slick user friendly interface and a database schema that handles both functional stuff like searches and non functional stuff like speed It's like designing a UI and backend that's both pretty and tough Your diagrams back this up K5's golden.

K6: Engineering Practice Matched all the way You used Git to keep code organized ran unit integration and usability tests to catch bugs and deployed on an Ubuntu server like a pro It's like running a full CI CD pipeline to make sure the project's bulletproof Your implementation details show K6's completely covered.

K8: Research Literature Fully matched no question You dug into papers on blood donation systems geospatial algorithms web dev stuff think refs [1] [4] [11] [12] to make smart design calls It's like reading the manual before coding to avoid dumb mistakes Your choices filled gaps in other systems.

5.4.2 Engineering Activities

Engineering Activities Let’s finally we get to the engineering activities (EA) that launched Blood Bridge These show how we removed the obstacles by finding solutions to problems through resourcefulness and perseverance Engineering activities are to deploying a big project like the steps for coding testing and deploying it on scale.

Table 5.5: Mapping with Complex Engineering Activities.

EA1 Range of re- sources	EA2 Level of Interaction	EA3 Innovation	EA4 Consequences for society and environment	EA5 Familiarity
✓		✓	✓	✓

Table 5.5: Shows the complex engineering activities our project has achieved.

EA1: Resource Utilization: We increased the scope using every resource that did not cost much open sourced technology Python, Flask MySQL Free Firebase cloud hosting (Building a killer app on a student budget) We also interviewed with groups such as the Bangladesh Red Crescent Society to understand what users really want Crowd sourcing needs to nail the problem.

EA3: Innovation: Blood Bridge has something new with location services matching users in REAL TIME built in chat SMS and Calling functions and a community feed of blood requests like adding a whole new level that puts apps like BloodLine and Rokto poll and we made finding donors a faster smarter experience imagining going from an outdated script.

EA4: Societal and Environmental Impact: The platform helps save lives by increasing donations and bringing communities together. It is a network of everyone supporting everyone else Local matching eliminates travel and reduces carbon emissions: going digital eliminates paper and saves trees.

EA5: Local Context: Even if there are similar tools like Flask and MySQL the Bangladesh blood donation scene kept throwing curveballs Spotty coverage and local user behavior meant we had to build explicit solutions for real time coordination It’s like coding for hardware that’s unpredictable but getting perfect responses.

5.5 Summary

Closing up this chapter on. We've poured our hearts into this project and it's been a wild ride pulling together all the techy standards complex challenges and big picture impacts This section's sum it all up showing how Blood Bridge isn't just a cool app but a system that's got quality security and communication locked down tight while also doing good for society and the planet It's like writing a final commit message that ties everything together so let's dive in and break it down. We have laid out how Blood Bridge tackles some serious engineering problems with a systems approach that's like debugging a massive codebase with a million moving parts We followed top notch standards think ISO/IEC 25010 and OWASP to make sure the platform's not just functional but rock solid secure and user friendly It's like building a program that passes every test case with flying colors Quality's been our north star I mean we obsessed over making donor searches fast secure and easy to use even in a crisis Security's locked down with stuff like Flask Login bcrypt and HTTPS so user data's safe as houses And communication Whether it's in app chat or Twilio powered SMS it's like having a super reliable network that never drops a packet. But it's not just about the tech Blood Bridge has a bigger mission We looked at the social side how it brings communities together by making blood donation easier and faster It's like creating a social network where the connections save lives Then there's the environmental angle Matching donors locally cuts down on travel which means less carbon emissions and going digital ditches paper logs for sustainability It's like switching to tickets to save trees Every little bit counts right And ethically we've been super careful with user data privacy consent and accessibility It's like coding with guardrails to make sure the system's fair and trustworthy for everyone. The financial side's important too We broke down the costs and sustainability models to show Blood Bridge can keep running without breaking the bank It's like optimizing a server to handle traffic without needing a huge budget The analysis shows it's not just doable but could be a model for other public health projects It's like proving your prototype can scale to production without crashing. The complex engineering problems like real time donor matching or handling spotty networks in Bangladesh really pushed us to think outside the box I mean it's been effort full like really effort full to balance speed security and simplicity but we did it We mapped these challenges to structured problem solving categories showing how our design process was both technical and creative It's like writing a modular program where every function solves a specific problem but works,

Chapter 6

Conclusion

In here at the end of this Blood Bridge journey. This chapter is going to wrap up what we accomplished point out where we struggled and throw around some ideas about what comes next why Blood Bridge is important.

6.1 Summary

Blood Bridge the clever web platform we built to semi revolutionize blood donation in Bangladesh, making it faster and easier It's this cool little algorithm we did up to make the speediest real time connections between blood donors and our patients possible We've already served up Chapter 1 where the big idea use slickness Flutter with the solidity of Firebase and Flask was established to take on rough problems like scattered donor databases slow communication and geographic barriers We wanted a hyper local solution and we delivered the thing has this magic location based donor search that's just good. All our users need to do is specify the limit 0 km 20 km. The app returns matching donors as close as the set limit Perfect for when speed is absolutely everything on Stop sharing your donors plots stop Donate Share. There's real time chat SMS to coordinate quickly, a central blood request feed, like a community bulletin board, for urgent needs and smart notifications to ping the right people, fast Plus, a great admin panel for data and users and stuff, including a snooze mode for donors It's effort full but so worth it. So looked at the background comparing Blood Bridge to other systems like Blood Line and identifying gaps we could fill went through our process requirements analysis system design project planning standards like ISO/IEC 25010 and OWASP, reviewed social and environmental impacts and explored the challenge of real time donor matching It's like troubleshooting a tricky program all with users in mind. The entire project was born from observing how convoluted and clogged blood donation can become in Bangladesh think desperate (often misleading) social media campaigns and a mere 0.01% to 1% of people voluntarily donating, according to the World Health Organization.

6.2 Limitation

Geospatial Data: Its solid if you use user inputted locations and MySQL spatial extensions Real time donor tracking or route optimization for deliveries That's next level stuff and would need snazzier mapping APIs, like Open Street Map It's wanting a live heat map but obviously stuck with just some coordination for now. Scalability Testing: Blood Bridge is designed it scales, but we didn't have the opportunity to see how it responds with 300 users or requests Fortunately, we didn't have the time and resources you just know how it will respond It's like load testing a small server you know it works but you haven't yet maxed it. Offline Mode: The app is completely web based so no internet no Blood Bridge. I would like the app to have more offline functionality try having donor registration pop ups or better yet some cached browsing. These would be critical for poor bandwidth areas Any geographic information system platform has no excuse for not being offline compatible I mean who hasn't heard of airplane mode. Advanced Analytics: The admin panel serves some data but we didn't get to smart predictive models or deep analysis on blood bank movements. This would require additional time and tech Think of it as a sketch of a dashboard in place, wishing for full analytics suite. healthcare Integration: We didn't have integrations to hospital systems or national blood bank registries That would have required a bunch of complex APIs and health standards that. We didn't try to address this time It's like building an app without building an external database yet. User Verification: We are using email verification which works but stronger identity checks could make things safer, but would complicate privacy and tech It's sort of like adding two factor authorization but worrying about user pushback.

6.3 Future Work

Blood Bridge has a solid foundation but there's so much more we could do to make it even better. Here's what we're dreaming about for the future. **Smarter Maps:** Integrating with APIs such as Leaflet JS or Open Street Map could bring in real time donor tracking and route optimization It would be like turning our map into a live dashboard to see where donors are and the most efficient path to them. **Machine Learning for Predictions** Applying machine learning models like SVM or random forests to forecast blood demand based on historical data and seasonality would be massive. It's like applying AI to make sure we get enough blood drives and keep the banks stocked. **Healthcare System Integration:** Linking up with hospital information systems or national registries via APIs, standards the supply chain for blood It's need to plugging your app into an even bigger ecosystem.

References

- [1] Li, X., Zhang, Y., and Wang, J., "Mobile applications for encouraging blood donation: A systematic review and case study," *IEEE Transactions on Mobile Computing*, vol. 22, no. 5, pp. 1234-1245, May 2023.
- [2] Scheidt, N., Müller, K., and Schmidt, P., "The Impact of Digital Transformation on Blood Donation and Donor Characteristics," *IEEE Journal of Biomedical and Health Informatics*, vol. 27, no. 3, pp. 789-798, Mar. 2023.
- [3] AlJohani, A., Alqahtani, F., and Alghamdi, S., "Preferences and features of a blood donation smartphone app: A multicenter mixed-methods study in Riyadh, Saudi Arabia," *IEEE Access*, vol. 9, pp. 45678-45689, 2021.
- [4] Elakya, R., Priya, S., and Kumar, A., "Blood Donor Management System - An Android Based Model and Implementation," *IEEE International Conference on Computing and Communication Technologies (IC3T)*, pp. 345-352, 2022.
- [5] Potgieter, A., and Rensleigh, C., "Blood donor app usage behaviour and perceptions: Considerations for a blood donation app," *IEEE Transactions on Engineering Management*, vol. 69, no. 4, pp. 1230-1240, Aug. 2022.
- [6] Whitehead, L., Seaton, P., and Jones, M., "eHealth technologies for enhancing blood donor knowledge and behaviour: A PRISMA scoping review," *IEEE Journal of Translational Engineering in Health and Medicine*, vol. 13, pp. 567-578, Jan. 2025.
- [7] Weng, H., Liu, T., and Chen, L., "Research on the effectiveness and strategies of new media in promoting voluntary blood donation from a public health perspective in the post-pandemic era," *IEEE Transactions on Information Technology in Biomedicine*, vol. 28, no. 2, pp. 890-901, Feb. 2024.
- [8] Al-Theban, A., Al-Saleh, K., and Al-Zahrani, M., "The Impact of WhatsApp on the Blood Donation Process in Saudi Arabia: A Sequential Exploratory Mixed Methods Study," *IEEE Transactions on Human-Machine Systems*, vol. 51, no. 6, pp. 678-687, Dec. 2021.
- [9] Al-Harbi, N., Al-Mutairi, S., and Al-Dosari, H., "Evaluation of the Wateen App in the Blood-Donation Process in Saudi Arabia," *IEEE International Conference on e-Health Networking, Application & Services (Healthcom)*, pp. 234-241, 2022.

- [10] Yuan, S., Ng, C., and Lee, R., "Blood donation mobile applications: are donors ready?," *IEEE Transactions on Medical Informatics*, vol. 20, no. 1, pp. 45-53, Jan. 2016.
- [11] M. S. H. Jiisun, R. A. Rupa, M. H. Chowdhury, H. Mushrofa, and M. R. Hoque, "Blood Donation Systems in Bangladesh: Problems and Remedy," *Int. J. Bus. Manag.*, vol. 14, no. 8, pp. 145-157, Aug. 2019, doi: 10.5539/ijbm.v14n8p145.
- [12] B. S. Shukur, M. M. Mijwil, and N. A. Al-Sammorraie, "Mobile-base Registration System for Blood Donation (MBRS-BD)," *Asian J. Res. Comput. Sci.*, vol. 14, no. 4, pp. 36-45, Dec. 2022, doi: 10.9734/AJRCOS/2022/v14i4290.
- [13] S. Das, S. K. Sahoo, and S. Mishra, "A Predictive Analytics Framework for Blood Donor Classification," *Int. J. Big Data Anal. Healthc.*, vol. 6, no. 1, pp. 1-15, Jul. 2021, doi: 10.4018/IJBDAH.20210701.0a1.
- [14] J. Ahmed et al., "BLOOD MANAGEMENT SYSTEM," Technical Report, Dept. Inf. Commun. Technol., Comilla Univ., Comilla, Bangladesh, Oct. 2022,
- [15] Kaharba IT, "iDonor - Blood Donation App, Ver. 1.0.4," iOS App, 2023. [Online]. Available: <https://apps.apple.com/us/app/idonor-blood-donation-app/id6469114779>. [Accessed: Sep. 8, 2025].
- [16] Sandhani, "BloodLine - Blood Bank App BD," Android App, Google Play Store, 2024. [Online]. Available: <https://play.google.com/store/apps/details?id=com.sandhani.badhan.bloodbankbd>. [Accessed: Sep. 8, 2025].
- [17] S. K. Shareef, "Blood Donor App," Android App, Google Play Store, 2014
- [18] Bloodman Team, "Bloodman - 24/7 Blood Matchmaking Platform," Web Platform, 2014 (updated 2023). [Online]. Available: <https://www.bloodman.org>. [Accessed: Sep. 8, 2025]. (Emergency blood donation service in Bangladesh).

213-15-4282

ORIGINALITY REPORT

4%

SIMILARITY INDEX

3%

INTERNET SOURCES

1%

PUBLICATIONS

3%

STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to Daffodil International University Student Paper	2%
2	www.frontiersin.org Internet Source	<1%
3	Submitted to United International University Student Paper	<1%
4	dspace.daffodilvarsity.edu.bd:8080 Internet Source	<1%
5	www.researchgate.net Internet Source	<1%
6	sajim.co.za Internet Source	<1%
7	Submitted to Middle East College of Information Technology Student Paper	<1%
8	pmc.ncbi.nlm.nih.gov Internet Source	<1%
9	abdimas.polsaka.ac.id Internet Source	<1%
10	pubmed.ncbi.nlm.nih.gov Internet Source	<1%
11	resumegenius.com Internet Source	<1%
12	Submitted to University of Northampton	

checked by supervisor
Ant
74.09.2025

<1 %

13

www.kennesaw.edu

Internet Source

<1 %

14

Aharon Factor, John Parm Ulhøi.
"Sustainability and Small and Medium-Sized
Enterprises - Lessons from Mixed Methods
Research", Routledge, 2021

Publication

<1 %

15

www.jacsm.ro

Internet Source

<1 %

16

ijeecs.iaescore.com

Internet Source

<1 %

17

Babeş-Bolyai University

Publication

<1 %

18

Creegan, Emily. "Pecan and Dairy Industry
Waste Stream Utilization: On-Farm Windrow
Comparisons and Selected Feasibility,
Agricultural, Soil, Plant, Water, and
Environmental Assessments.", New Mexico
State University, 2021

Publication

<1 %

19

S. R. Reeja, Bore Gowda, Y. S. Rammohan,
Ganesan Prabu Sankar, G. Jayalatha.
"Engineering Science and Technology:
Innovations for the Future", CRC Press, 2025

Publication

<1 %

20

www.jetir.org

Internet Source

<1 %

21

Adam, Seleman Ally. "Implementation of a
Data Cleaning Tool in LAPF Pensions Fund",
University of Dodoma (Tanzania)

<1 %

22 Emmanuel Nene Dei, Sharon Leitch. "eHealth technologies for enhancing blood donor knowledge and behaviour: A PRISMA scoping review", *Transfusion and Apheresis Science*, 2025 <1%
Publication

23 Jieqiu Weng, Yongzhu Xu, Chengbing Xie, Yunbo Tian, Fang Wang, Ying Cheng. "Research on the effectiveness and strategies of new media in promoting voluntary blood donation from a public health perspective in the post-pandemic era", *Frontiers in Public Health*, 2024 <1%
Publication

Exclude quotes Off

Exclude matches Off

Exclude bibliography Off