Emergency Blood Platform

Project submitted to the

SRM University – AP, Andhra Pradesh

for the partial fulfillment of the requirements to award the degree of

Bachelor of Technology In

Computer Science and Engineering

School of Engineering and Sciences

Submitted by

AP23110010638 – M.Srinivas

AP2311 – K.Venu

AP2311 – K.Venkat

AP23110010906 – R.Jaswanth

Under the Guidance of

Yatharth Shahrawat

SRM University–AP

Neerukonda, Mangalagiri, Guntur

Andhra Pradesh – 522 240

[December, 2025]

Acknowledgement

We express our sincere gratitude to **Mr. Yatharth Shahrawat**, our respected faculty, for his continuous guidance, valuable insights, and encouragement throughout the course, which greatly helped us in developing the **Emergency Blood Donation & Management Platform**. His expertise and support played an important role in helping us understand full-stack development concepts and successfully implement this system.

We would also like to thank our department and institution for providing the necessary resources and a supportive learning environment that enabled us to work effectively on this project. Finally, we extend our appreciation to our peers and everyone who contributed directly or indirectly to the completion of this work.

This project has been a valuable learning experience, allowing us to gain practical knowledge in **frontend development, backend API handling, Tailwind CSS, state management, and modern web development practices**, while also understanding how technology can support real-time emergency services.

Team members

AP23110010638 – M.Srinivas

AP2311 – K.Venu

AP2311 – K.Venkat

AP23110010906 – R.Jaswanth

Table of contents

1. Introduction

In critical medical situations, timely access to the right blood type can make the difference between life and death. Yet, despite advancements in technology, many people still struggle to find available donors quickly. Traditional processes often involve contacting multiple blood banks, waiting for availability updates, or relying on manual communication, which leads to delays, inefficiency, and in many cases, serious medical risks.

The rapid growth of web applications has created opportunities to streamline these essential services and build systems that connect donors, patients, and hospitals in real time. However, many existing platforms lack intuitive interfaces, real-time updates, or efficient matching algorithms. Some provide only basic donor listings, while others overwhelm users with unorganized data and poor navigation. Very few systems offer a complete experience that includes emergency requests, donor verification, hospital coordination, and real-time notifications within a single, user-friendly platform.

To address these challenges, the **Emergency Blood Donation & Management Platform** has been developed as a modern, reliable, and efficient web application designed to save time and lives. Built with a responsive frontend and a robust backend structure, the platform provides seamless donor–receiver interaction, instant request posting, donor matching, and secure data management.

This project demonstrates how thoughtful UI/UX design, modular components, and modern full-stack technologies can transform the traditional blood donation process into a faster, more accessible, and life-saving digital solution. The platform establishes a foundation for a scalable emergency support system that can expand into hospital integrations, automated donor alerts, blood inventory tracking, and nationwide blood availability networks in the future.

2. Scenario-Based Introduction

Imagine a patient in a hospital who urgently requires a specific blood type for surgery or an accident victim who needs immediate transfusion. Their family members, unsure where to find donors quickly, open the **Emergency Blood Donation & Management Platform**. Within seconds, they can view available donors nearby, filter by blood group, location, or urgency, and instantly send a request to compatible donors. Each donor's profile displays verified details, availability status, and contact options, enabling fast and informed decision-making during emergencies.

Meanwhile, registered donors can receive instant notifications when someone nearby needs their blood type. They can view request details, check distance, accept or decline the request, and coordinate directly through the platform. Features like real-time updates, search filters, and responsive design ensure that life-saving information is accessible anytime, on any device. Secure login ensures privacy, personalized dashboards, and access to donation history for both donors and recipients.

Hospitals and blood banks can also use the system to update blood availability, respond to urgent requests, and manage donor records efficiently through the backend. Notification alerts help them track pending requests, confirm availability, or report shortages immediately. This creates a continuous and reliable communication flow between hospitals, donors, and recipients.

The **Emergency Blood Donation & Management Platform** simulates a real-world emergency response system and provides an integrated digital solution for finding donors, posting emergency requests, and managing blood availability quickly and efficiently. By combining modern web technologies, thoughtful UI/UX design, and real-time interaction, it transforms the traditional blood donation process into a faster, more organized, and life-saving digital experience.

3. Target Audience

The Emergency Blood Donation & Management Platform is designed for a wide range of users involved in emergency medical care and blood donation activities. The key target audiences include:

1. Patients & Families
Individuals who urgently require a specific blood type for surgeries, accidents, or medical treatments.
They need a fast and reliable platform to find nearby donors, post emergency requests, and receive quick responses.

2. Voluntary Blood Donors
People who are willing to donate blood and want an organized way to stay informed about ongoing needs.
The platform helps donors view requests, receive notifications, and respond instantly based on their availability.

3. Hospitals & Blood Banks
Healthcare professionals responsible for managing blood stock and coordinating transfusions.
They use the system to update availability, verify donor details, and handle emergency cases more efficiently.

4. Health Volunteers & NGOs
Organizations or volunteers who support blood donation drives and emergency awareness. They can track requests, share alerts, and assist in connecting donors with recipients.

5. Tech-Savvy Users
Users who prefer modern, mobile-friendly applications with real-time updates and smooth navigation.
They benefit from features like instant alerts, responsive UI, and easy access to information.

By addressing the needs of these groups, the Emergency Blood Donation & Management Platform ensures faster communication, improved transparency, and a more efficient process for saving lives.

4. Project Goals and Objectives

The **Emergency Blood Donation & Management Platform** is designed to provide a fast, reliable, and user-friendly system for connecting blood donors, patients, and hospitals during emergencies. The main goals and objectives are:

Seamless User Experience
Create an intuitive and responsive interface that allows users to quickly search for donors, post emergency requests, and access essential information effortlessly.
The platform ensures smooth navigation, clear data presentation, and accessibility across all devices.

Smart Donor Matching and Emergency Handling
Enable users to search for donors by blood group, location, and availability.
The system supports real-time notifications, emergency request posting, and quick donor responses to ensure timely blood assistance.

Efficient Admin and Data Management
Provide hospitals and administrators with tools to manage donor records, update blood availability, and monitor ongoing requests through a structured backend.
This ensures accurate, up-to-date information for patients, donors, and medical staff.

Modern Web Technologies
Use a modular frontend with modern technologies such as Vite, Tailwind CSS, and JavaScript, combined with a backend that handles secure data flow and API operations.
Features like real-time updates and responsive UI contribute to a scalable and maintainable platform.

By achieving these objectives, the **Emergency Blood Donation & Management Platform** aims to simplify the blood donation process, improve transparency, and support faster response during critical medical situations.

5. Prerequisites

Before running the Emergency Blood Donation & Management Platform, certain tools, software, and technical knowledge are required. The project uses Vite, JavaScript, Tailwind CSS, and a backend API, making the setup simple, lightweight, and beginner-friendly.

A suitable code editor is essential for working with the project files. Visual Studio Code is recommended because it supports modern JavaScript development, Tailwind CSS, and API integration effectively. Extensions such as Prettier, ESLint, Tailwind CSS IntelliSense, and Live Server enhance productivity and help maintain clean code.

A modern web browser is needed to run and debug the application. Google Chrome is preferred due to its powerful developer tools, while Firefox and Microsoft Edge can also be used for cross-browser testing.

Since the platform uses JavaScript and Vite on the frontend, developers should have a basic understanding of modular code structure, DOM manipulation, event handling, fetch API usage, and Tailwind CSS for styling. Knowledge of modern JavaScript concepts such as arrow functions, array methods, template literals, and async/await is helpful for managing different sections of the application.

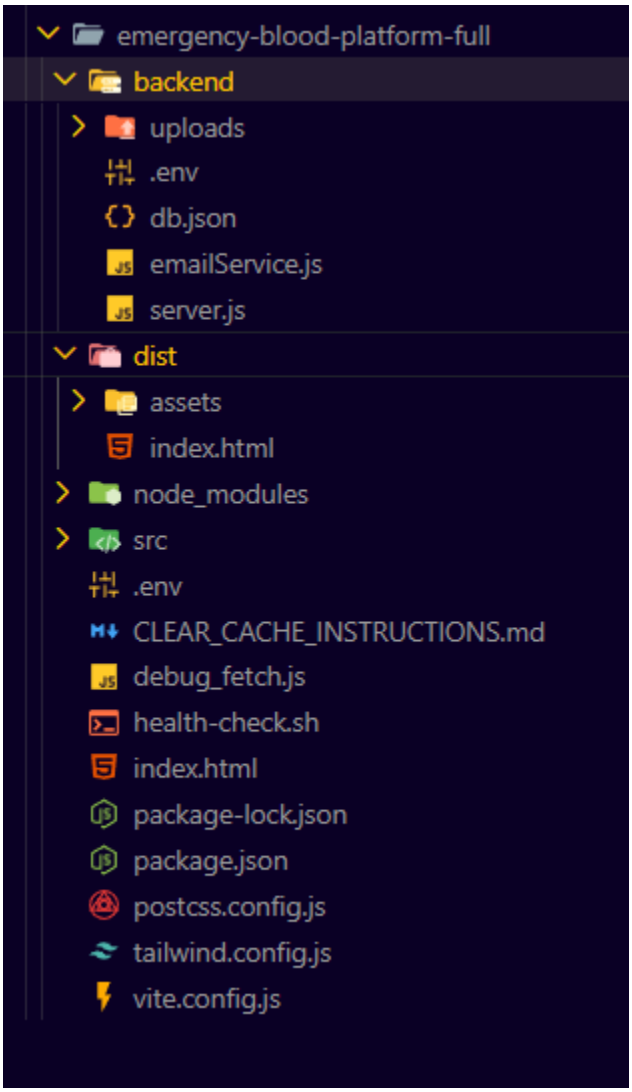
The platform interacts with backend APIs, so developers must know how to handle JSON data, make API calls using fetch, and perform essential CRUD operations such as fetching donor data, creating emergency requests, updating request status, or deleting unnecessary records.

A basic understanding of HTML and CSS is also required. Although Tailwind CSS handles most of the layout and styling, familiarity with HTML structure, utility classes, and responsive design principles helps in building forms, dashboards, and user interfaces effectively.

To run the system smoothly, the device should have at least 4GB RAM (8GB recommended), a dual-core processor, and around 5GB of free disk

space. A stable internet connection is required for installing project dependencies and testing backend API responses.

6. Project Structure



The Emergency Blood Donation & Management Platform is organized into frontend and backend components to ensure modularity, scalability, and easy maintenance. Each part of the system plays a specific role in managing the user interface, application logic, data handling, and server operations.

Frontend (Vite + JavaScript + Tailwind CSS)

The frontend contains all user interface elements, pages, and interaction logic for donors, patients, and hospitals. It includes the main HTML structure, style configuration, and scripts required for the platform.

The **index.html** file serves as the entry point of the application, while the **src** folder contains scripts, UI layouts, and logic for handling features such as donor search, emergency request forms, user dashboards, and real-time updates. Tailwind CSS configuration is handled through tailwind.config.js and postcss.config.js, enabling responsive and utility-based styling across the application.
The **dist** folder stores the production build files generated by Vite for deployment.

Backend (Node.js + JSON Storage)

The backend manages all server requests, API routes, and data processing for the application.

The backend folder includes:
A **server.js** file that handles API endpoints for creating emergency requests, managing donors, updating availability, and processing form submissions.
A **db.json** file that stores donor data, blood requests, and other essential records used during development and testing.
An **emailService.js** file responsible for sending automated email notifications to donors or recipients when a request is created or updated.
An **uploads** folder for storing any files or documents submitted through the platform.
An **.env** file for storing sensitive environment variables such as email credentials or API keys.

Assets and Build Files

The **assets** folder inside the dist directory contains images, icons, and static files used in the user interface.

The **health-check.sh** and **debug_fetch.js** files are included for troubleshooting, debugging API routes, and ensuring server stability.

Configuration Files

The project includes several configuration files that help manage dependencies, scripts, and development settings.

The **package.json** file stores project dependencies and scripts required to run both the frontend and backend.

The **vite.config.js** file contains build settings and development server configurations for Vite.

The **.env** file at the root level holds environment variables for frontend operations.

The project structure ensures that all components—frontend, backend, assets, and configuration files—work together smoothly, making the platform scalable, maintainable, and suitable for real-world emergency scenarios.

7. Project Flow

Start → User Visits Website → Choose Role (Donor/Reciver)
→ If **New User → Sign up → Login**
→ If **Existing User → Login**
→ System Verifies Credentials & Role

If Donor: Choose requests→ View Details → Confirn request → complete donation → get reward → Logout

If Reciver: Send new request → Choose Blood group → Complete Reciver
→ Manage Dashboard
→ Logout → End

8. Code Description - App.jsx

Imports and Essential Scripts
The project begins by importing essential JavaScript modules, configuration files, and utility functions that support the overall functionality of the application. These include API communication scripts, form-handling logic, emergency request functions, and UI-related utilities that help create a smooth and interactive user experience.
These core scripts are responsible for handling tasks such as sending requests, validating user input, updating the interface dynamically, and connecting the frontend to the backend server.

UI Components
The platform consists of multiple interface sections that appear on different pages of the website:
Navbar:
Displayed at the top of every page and provides navigation links such as Home, Donor Registration, Emergency Request, Dashboard, and Contact.
Footer:
Displayed at the bottom of all pages and contains contact information, emergency helpline details, and additional links for users.

Pages of the Emergency Blood Platform
Each page of the platform acts as a dedicated screen for a specific function, similar to how separate views work in a single-page application. These pages include:
Home Page:
Displays the purpose of the platform, emergency highlights, statistics, and quick-access buttons to register or request blood.
Donor Registration Page:
Allows new donors to create a profile by entering their blood type, location, availability, and contact details.
Login Page:
Provides a secure login option for hospitals, administrators, or previously registered donors.
Emergency Request Page:
Enables users to submit urgent blood requests by specifying blood type, location, patient details, and urgency level.
Dashboard Page:

Displays submitted requests, donor responses, and history for logged-in users or administrators.

Hospital Panel (if applicable):

Helps hospitals view donor availability, verify donor records, and manage active cases.

Each page is loaded through standard routing handled directly by Vite and browser navigation.

Application Structure Flow

Outer Wrapper (Main HTML Layout):

The index.html file provides the structural skeleton for the entire platform. All dynamic JavaScript modules and styles are injected into this layout to form the complete UI.

Toast Notification System:

The platform uses custom JavaScript alerts or UI pop-ups to inform users about important actions such as:

- Emergency request submitted successfully
- Donor registered
- Request failed due to incorrect data
- Server issues or validation errors

These notifications improve user experience and provide immediate feedback.

Router and Navigation Logic

Although the project does not use React Router, navigation is handled through separate HTML pages, anchor-based routing, and JavaScript event handling. When a user clicks a link on the Navbar, the browser loads the corresponding page where JavaScript files attach logic dynamically.

Main Content Section

Each page contains a main content region where donor forms, emergency requests, listings, and dashboards are displayed. JavaScript fetches relevant data from the backend and updates the sections dynamically.

Footer Section

Located at the bottom of every page, displaying:

- Emergency contact details
- Social media or awareness links
- Platform or hospital information

9. Project Execution

To run the project locally:

1.Install dependencies

```
npm install
```

2. Start the backend mock server

```
npm run server
```

3. Start Frontend

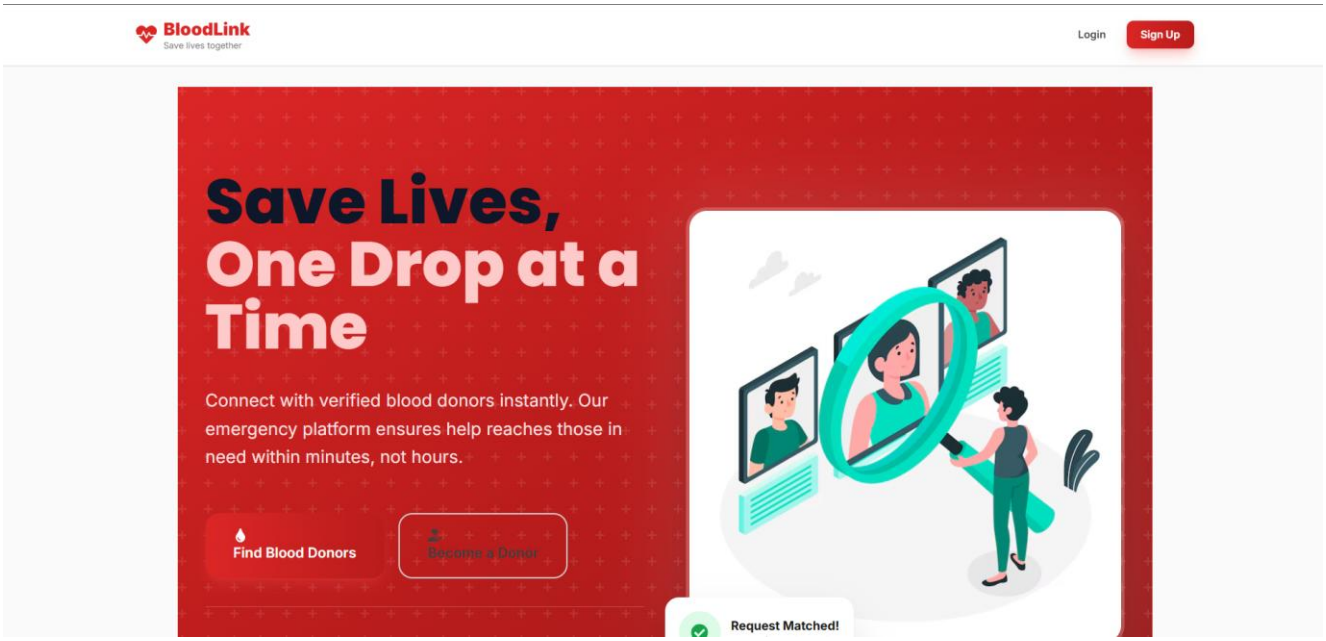
```
npm run dev
```

4. Access the platform in browser:

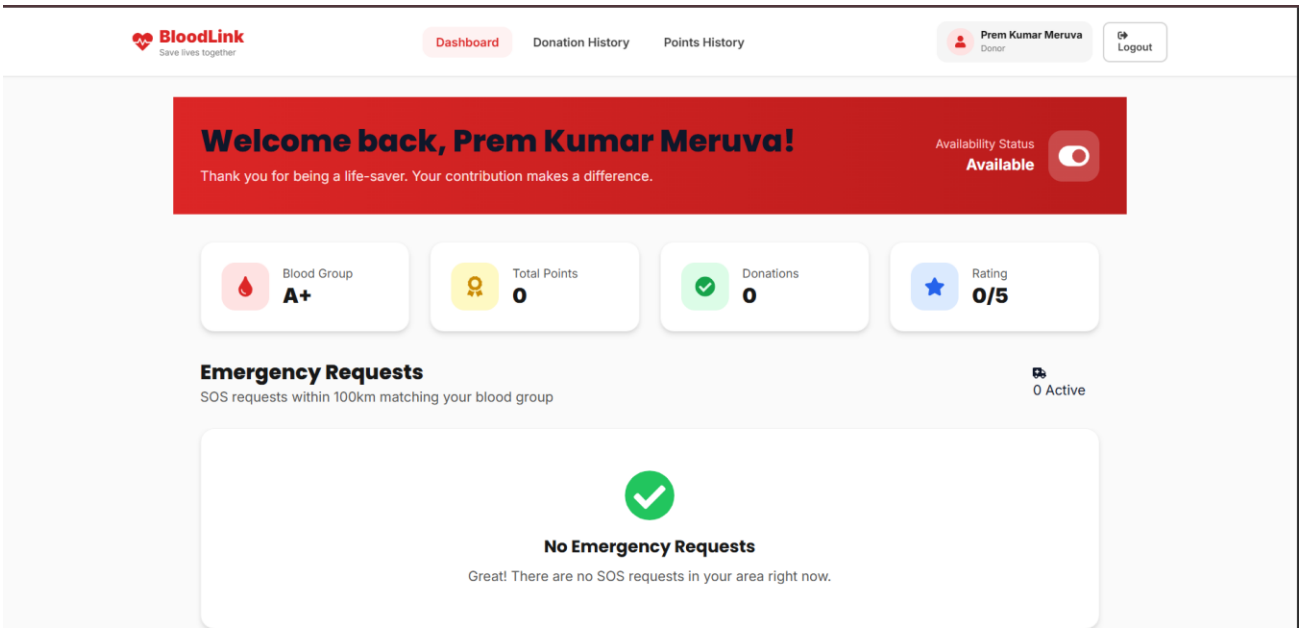
```
http://localhost:5173
```

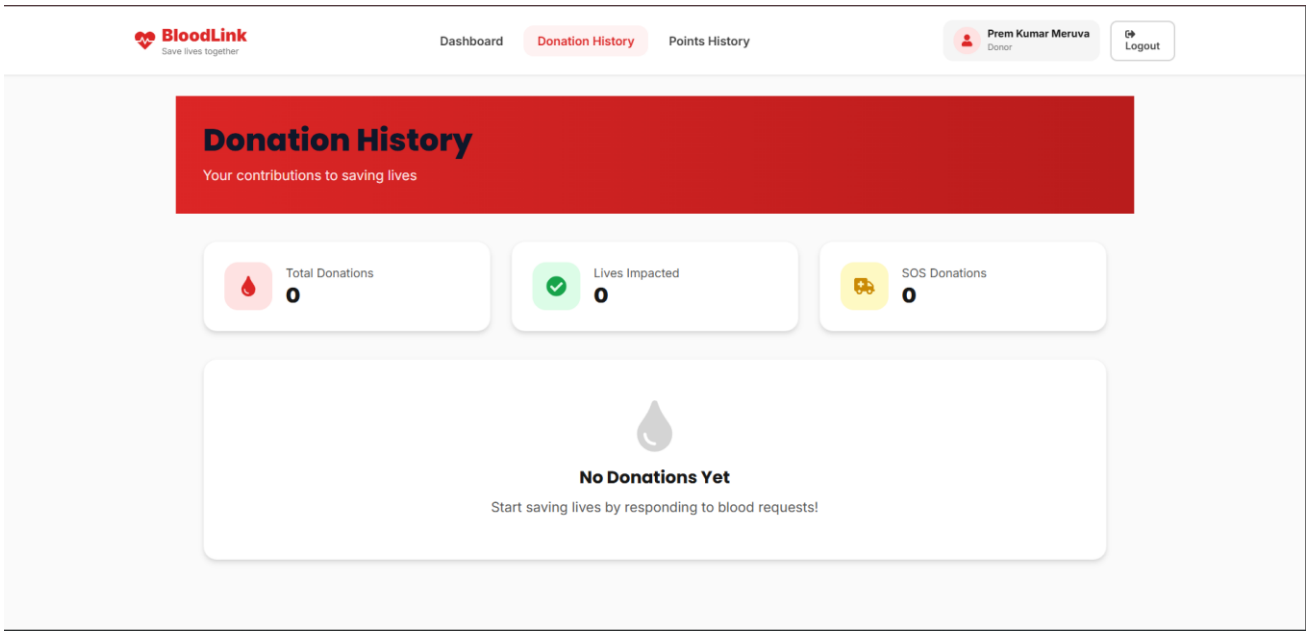
10. Outputs

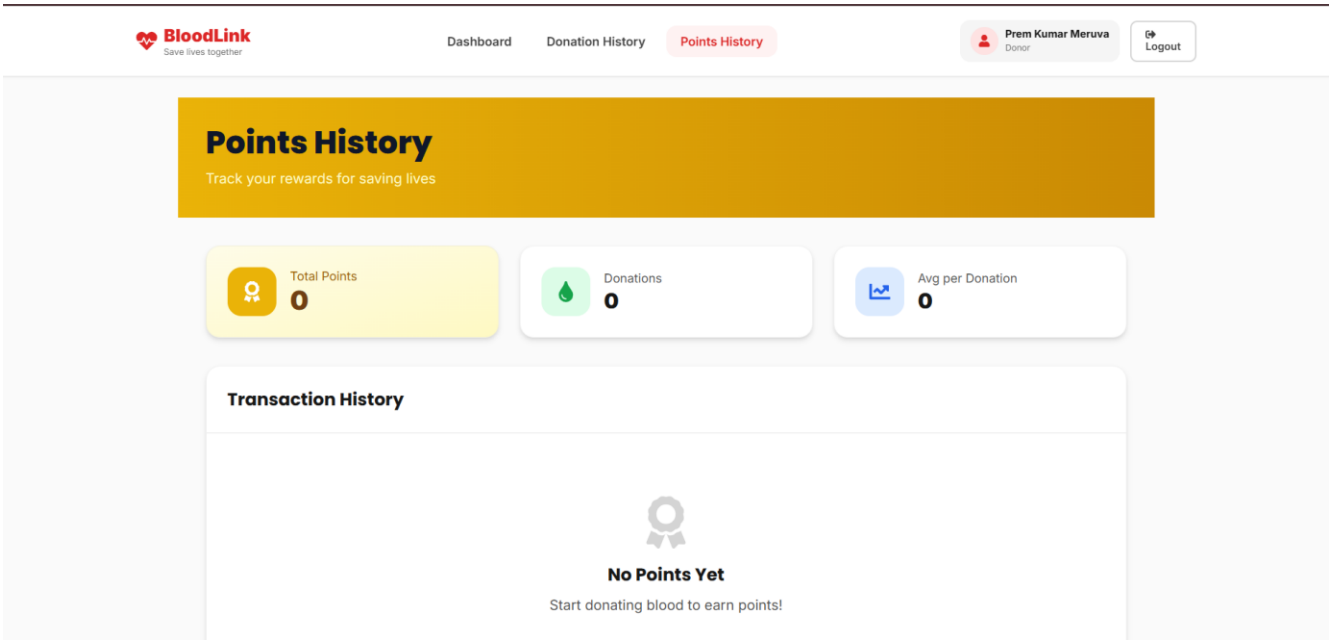
Landing Page:



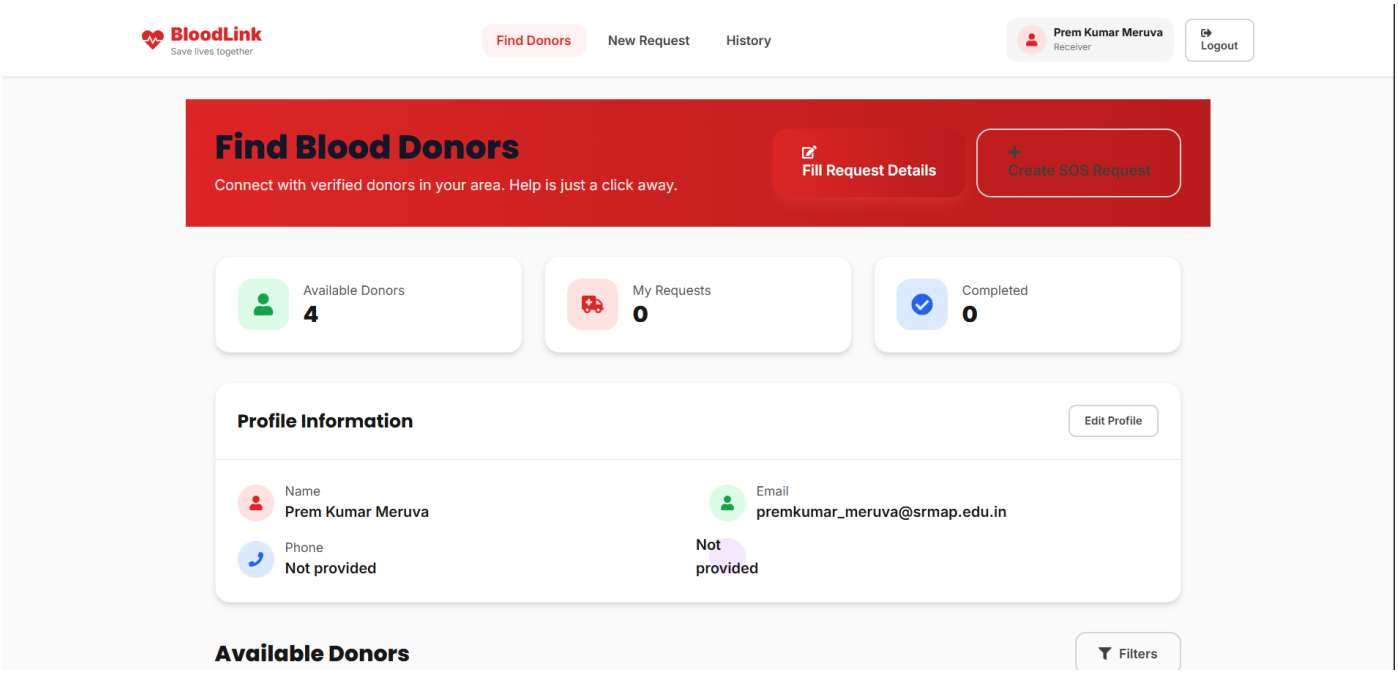
Donor Page:

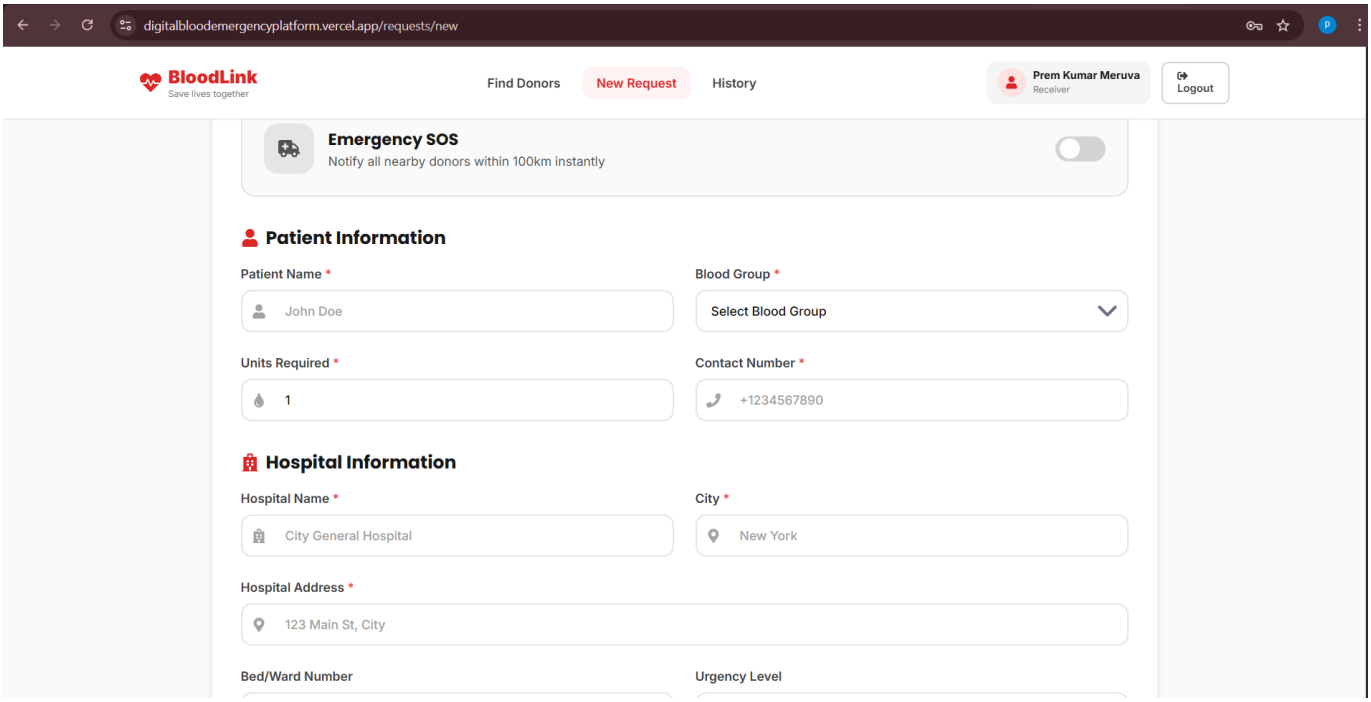

BloodLink
Save lives together
Dashboard
Donation History
Points History
Prem Kumar Meruva
Donor
Logout
Welcome back, Prem Kumar Meruva!
Thank you for being a life-saver. Your contribution makes a difference.
Availability Status
Available
Blood Group
A+
Total Points
0
Donations
0
Rating
0/5
Emergency Requests
SOS requests within 100km matching your blood group
0 Active
No Emergency Requests
Great! There are no SOS requests in your area right now.

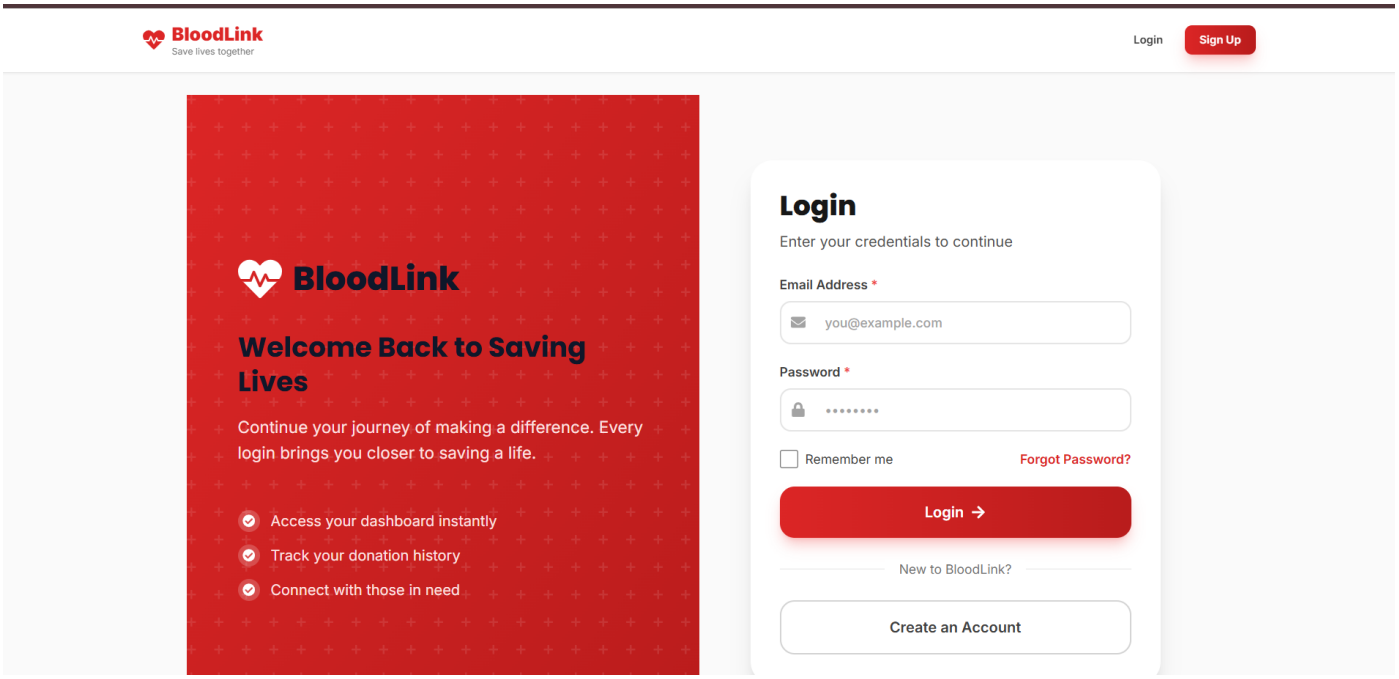

BloodLink
Save lives together
Dashboard
Donation History
Points History
Prem Kumar Meruva
Donor
Logout
Donation History
Your contributions to saving lives
Total Donations
0
Lives Impacted
0
SOS Donations
0
No Donations Yet
Start saving lives by responding to blood requests!

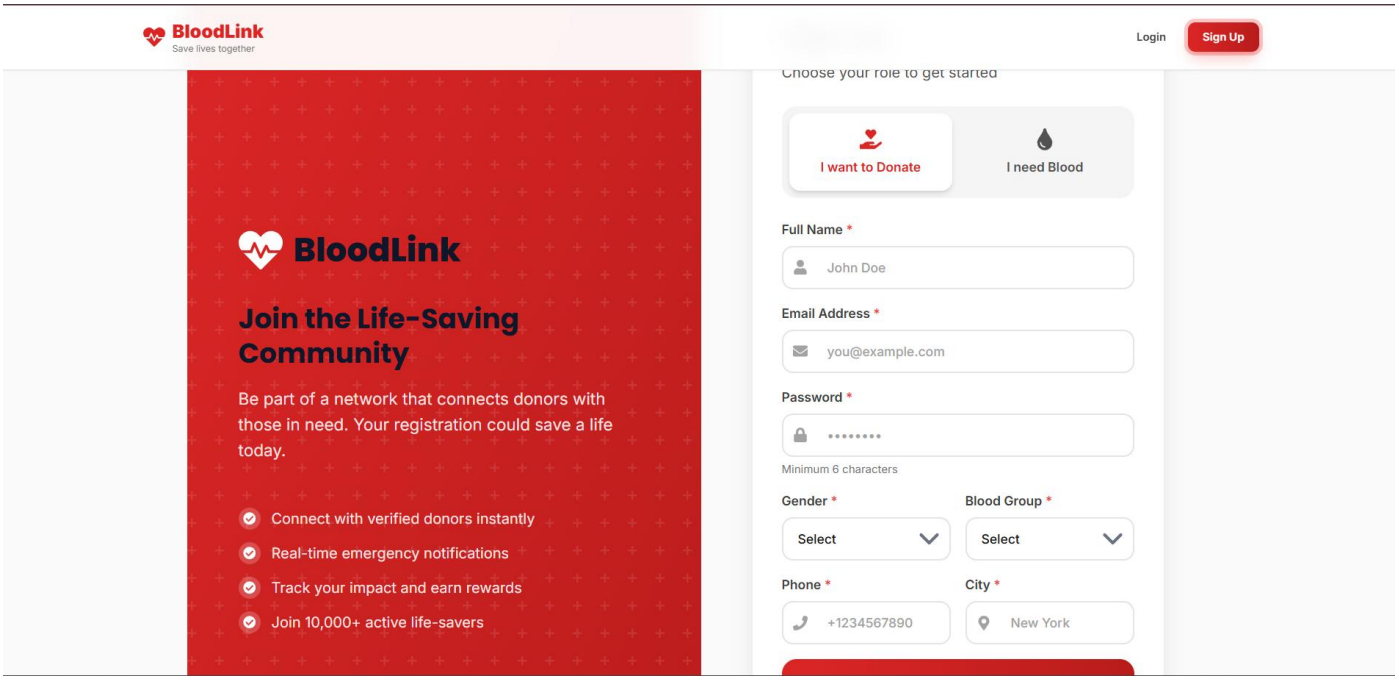

BloodLink
Save lives together
Dashboard
Donation History
Points History
Prem Kumar Meruva
Donor
Logout
Points History
Track your rewards for saving lives
Total Points
0
Donations
0
Avg per Donation
0
Transaction History
No Points Yet
Start donating blood to earn points!

Reciver Page:






BloodLink
Save lives together
Login
Sign Up
BloodLink
Welcome Back to Saving Lives
Continue your journey of making a difference. Every login brings you closer to saving a life.
Access your dashboard instantly
Track your donation history
Connect with those in need
Login
Enter your credentials to continue
Email Address *
you@example.com
Password *
Remember me
Forgot Password?
Login →
New to BloodLink?
Create an Account


BloodLink
Save lives together
Login
Sign Up
BloodLink
Join the Life-Saving Community
Be part of a network that connects donors with those in need. Your registration could save a life today.
Connect with verified donors instantly
Real-time emergency notifications
Track your impact and earn rewards
Join 10,000+ active life-savers
Choose your role to get started
I want to Donate
I need Blood
Full Name *
John Doe
Email Address *
you@example.com
Password *
Minimum 6 characters
Gender *
Select
Blood Group *
Select
Phone *
+1234567890
City *
New York

11. Conclusion

The Emergency Blood Donation & Management Platform successfully demonstrates the development of a modern, responsive, and user-friendly system for connecting blood donors, patients, and hospitals during critical situations. Through this project, we implemented essential features such as donor registration, emergency request submission, real-time notifications, secure data handling, and a structured backend for managing donor information and user requests. The use of modular JavaScript, Tailwind CSS for responsive design, and backend API integration ensured scalability, maintainability, and smooth user interaction across the platform.

Working on this project allowed us to gain valuable practical knowledge in full-stack development, client–server communication, data management, and UI/UX design. The system simplifies the process of finding blood donors, improves communication between users and hospitals, and enhances the overall efficiency of emergency medical support.
Overall, the project meets its objectives by offering a reliable, fast, and intuitive platform for managing blood donation needs. It serves as a strong foundation for future enhancements such as automated donor matching algorithms, live blood bank inventory tracking, hospital dashboards, SMS/email emergency alerts, and integration with real medical databases.

12. References

React.js Official Documentation – https://react.dev/

React Router Documentation – https://reactrouter.com/

Tailwind CSS Documentation – https://tailwindcss.com/

JSON Server Guide – https://github.com/typicode/json-server

https://digitalbloodemergencyplatform.vercel.app/