**FINAL YEAR PROJECT**

**BLOOD DONATION APP**



***Govt. Graduate College, Civil Lines, Sheikhupura***

*Supervisor* ***Hasan Raza***

*Submitted By*

Manahil Saeed 068420/2021-KS-482

***Project ID:*** *21-KS-BSIT-11*

*Bachelor of Science in Information Technology (2021-2025)*

**Declaration**

The work reported in this project was carried under the supervision of Project Supervisor, Hasan Raza, at Govt. Graduate College, Civil Lines, Sheikhupura.

I am here declaring that this project and the contents of the project are the product of my research and no part has been copied from any other written source

I further declare that this work has not been submitted for the award of any other degree diploma. The university may act if the provided information is found inaccurate at any stage.

|  |  |  |
| --- | --- | --- |
| **Student Name** | **Registration No.** | **Signature** |
| Manahil Saeed | 2021-KS-482 |  |

**STATEMENT OF SUBMISSION**

This to certify that following student has completed the final project named as: **“Blood Donation App”** at **Govt. Graduate College, Civil Lines, Sheikhupura**to fulfill the partial requirement of the degree of Bachelors of Information Technology.

|  |  |  |
| --- | --- | --- |
| PU Roll No | Name | Registration no. |
| 068420 | Manahil Saeed | 2021-KS-482 |

**Head of Department Supervisor**

**Muhammad Ali Waqas Hasan Raza**

Associate Professor of Computer Science Lecturer in Computer Science

Govt. Graduate College, Govt. Graduate College,

Civil Lines, Sheikhupura Civil Lines, Sheikhupura

**Project Coordinator**

**Hasan Raza**

Lecturer in Computer Science

Govt. Graduate College,

Civil Lines, Sheikhupura

**PROOFREADING CERTIFICATE**

This certify that Manahil Saeed (PU Roll # 068420) have completed their project “Blood Donation App”. This document does not contain any spelling, punctuation or grammatical mistakes.

**Supervisor**

**Hasan Raza**

Lecturer in Computer Science

Govt Graduate College,

Sheikhupura.

**ACKNOWLEDGEMENT**

In the name of ALMIGHTY ALLAH, the most merciful, the most beneficent. He is the entire source of knowledge of wisdom and Who blessed us with the ability to do work. I am grateful to the Prophet Muhammad (PBUH) who gave me the spirit to learn. No doubt in it because of ALLAH ALMIGHTY that today I am able to complete my project. I am also thankful to our honorable Supervisor Hasan Raza Department of Computer Science of Govt. Graduate College, Civil Lines, Sheikhupura for his guidance, encouragement, support, and admirable help. In fact, without his efforts and guidance, it might not have been possible to complete the work. I would like to thank all the faculty of Information Technology of Govt. Graduate College, Civil Lines, Sheikhupura for this great support and help for the proper completion of my project work. I should feel it necessary to express my love for my parents and my husband for their support, motivation and contribution to project completion. May all they live, long to see my dreams being fulfilled. Ameen!

**CERTIFICATE OF APPROVAL**

It is to certify that the final year project of BS (IT) “**Blood Donation App**” was developed by:

|  |  |  |
| --- | --- | --- |
| **Student Name** | **Registration No** | **Roll No** |
| Manahil Saeed | 2021-KS-482 | 068420 |

Under the supervision of “Hasan Raza” and that in our opinion; it is fully adequate, in scope and quality for the degree of Bachelor of Science in Information Technology.

**Supervisor**

Hasan Raza

Govt. Graduate College,  
Civil Lines, Sheikhupura.

**DATED:** \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Head of Department**

Muhammad Ali Waqas

Govt. Graduate College,  
Civil Lines, Sheikhupura.

**DATED:** \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**PROJECT INFORMATION**

Project Title Blood Donation App

Objective This App is created to make bridge between donors & recipients

Undertaken by Manahil Saeed

Supervised by Hasan Raza

Starting date 11/03/2024

Completion date 15/08/2025

Tools used VS. code, Draw.io

Table of Contents

[1. INTRODUCTION 14](#_Toc206759054)

[**1.1** Scope 14](#_Toc206759055)

[**1.2** Purpose 14](#_Toc206759056)

[2. PROBLEM DEFINATION 15](#_Toc206759057)

[**2.1** Problem Statement 15](#_Toc206759058)

[**2.2** Deliverable and Development Requirements 15](#_Toc206759059)

[**2.2.1** Development required tools 16](#_Toc206759060)

[**2.2.2** Languages 16](#_Toc206759061)

[3. FEASIBILITY REPORTS 17](#_Toc206759062)

[**3.1** Project Feasibility Report 17](#_Toc206759063)

[**3.2** Technical Feasibility 17](#_Toc206759064)

[**3.3** Operational Feasibility 17](#_Toc206759065)

[**3.4** Economic Feasibility 18](#_Toc206759066)

[**3.5** Schedule Feasibility 18](#_Toc206759067)

[**3.6** Specification Feasibility 18](#_Toc206759068)

[**3.7** Legal & Ethical Feasibility 18](#_Toc206759069)

[4. QUALITY ATTRIBUTES 19](#_Toc206759070)

[**4.1** Reliability 19](#_Toc206759071)

[**4.2** Usability 19](#_Toc206759072)

[**4.3** Security 19](#_Toc206759073)

[**4.4** Performance 19](#_Toc206759074)

[**4.5** Availability 19](#_Toc206759075)

[5. SPECIFIC REQUIREMENTS 20](#_Toc206759076)

[**5.1** User Registration and Authentication: 20](#_Toc206759077)

[**5.2** Profile Creation: 20](#_Toc206759078)

[**5.3** Donor Search: 20](#_Toc206759079)

[**5.4** Donor Profile: 20](#_Toc206759080)

[**5.5** Communication Tools: 20](#_Toc206759081)

[6. EXTERNAL INTERFACE REQUIREMENT 21](#_Toc206759082)

[6.1 User Interfaces 21](#_Toc206759083)

[6.2 Software Interfaces 21](#_Toc206759084)

[6.3 Hardware Interfaces 21](#_Toc206759085)

[7. REQUIREMENT IDENTIFYING TECHNIQUES 22](#_Toc206759086)

[**7.1** User Use Case: 22](#_Toc206759087)

[**7.2** USER USE CASE DESCRIPTION 24](#_Toc206759088)

[**7.2.1** User Sign up 24](#_Toc206759089)

[**7.2.2** Create Profile 24](#_Toc206759090)

[**7.2.3** User Login 25](#_Toc206759091)

[**7.2.4** Forget password 25](#_Toc206759092)

[**7.2.5** Need Blood 26](#_Toc206759093)

[**7.2.6** Profile 26](#_Toc206759094)

[**7.2.7** User logout 27](#_Toc206759095)

[**7.3** Admin Use Case 28](#_Toc206759096)

[**7.4** Admin Use Case Description 30](#_Toc206759097)

[**7.4.1** Admin Login 30](#_Toc206759098)

[**7.4.2** Forgot password 30](#_Toc206759099)

[**7.4.3** Change password **Error! Bookmark not defined.**](#_Toc206759100)

[**7.4.4** Change email **Error! Bookmark not defined.**](#_Toc206759101)

[**7.4.5** Blocked account 31](#_Toc206759102)

[**7.4.6** Logout 31](#_Toc206759103)

[8. DESIGN MODEL 32](#_Toc206759104)

[**8.1** User procedure 32](#_Toc206759105)

[**8.2** Admin procedure 33](#_Toc206759106)

[**8.3** Class Diagram 34](#_Toc206759107)

[**8.4** User Activity Diagram 35](#_Toc206759109)

[**8.5** Admin Activity Diagram 36](#_Toc206759110)

[**8.6** User State Transition Diagram 37](#_Toc206759111)

[**8.7** Admin State Transition Diagram 38](#_Toc206759112)

[**8.8** User Sequence Diagram 39](#_Toc206759113)

[**8.8.1** Sign up 39](#_Toc206759114)

[**8.8.2** Create profile 40](#_Toc206759115)

[**8.8.3** Login 40](#_Toc206759116)

[**8.8.4** Forget password 41](#_Toc206759117)

[**8.8.5** Need Blood 42](#_Toc206759118)

[**8.8.6** Profile 43](#_Toc206759119)

[**8.8.7** Log out 44](#_Toc206759120)

[**8.9** Admin Sequence Diagram 45](#_Toc206759121)

[**8.9.1** Log in 45](#_Toc206759122)

[**8.9.2** Forgot password 46](#_Toc206759123)

[**8.9.3** Blocked Account 47](#_Toc206759124)

[**8.9.4** Log out 47](#_Toc206759125)

[**8.10** User Collaboration Diagram 48](#_Toc206759126)

[**8.10.1** Sign up 48](#_Toc206759127)

[**8.10.2** Create Profile 49](#_Toc206759128)

[**8.10.3** Sign In 49](#_Toc206759129)

[**8.10.4** Forget Password 50](#_Toc206759130)

[**8.10.5** Need Blood 51](#_Toc206759131)

[**8.10.6** Profile 52](#_Toc206759132)

[**8.10.7** Log out 52](#_Toc206759133)

[**8.11** Admin Collaboration Diagram 53](#_Toc206759134)

[**8.11.1** Login 53](#_Toc206759135)

[**8.11.2** Forgot password 54](#_Toc206759136)

[**8.11.3** Blocked Account 55](#_Toc206759137)

[**8.11.4** Logout 55](#_Toc206759138)

[**8.12** DFD Diagram 56](#_Toc206759139)

[9. PROJECT COSTING 57](#_Toc206759140)

[**9.1** Project Cost Estimation by Function Point Analysis 57](#_Toc206759141)

[**9.1.1** External inputs (EI) 57](#_Toc206759142)

[**9.1.2** External outputs (EO) 57](#_Toc206759143)

[**9.1.3** External inquiries (EI) 57](#_Toc206759144)

[**9.1.4** Internal logical files (ILF) 57](#_Toc206759145)

[**9.1.5** External logical files (ELF) 57](#_Toc206759146)

[**9.2** Weights of Five Functional Point Attributes 58](#_Toc206759147)

[**9.3** Calculating Function Point: 58](#_Toc206759148)

[**9.4** Calculating Factors 59](#_Toc206759149)

[**9.5** General System Characteristics (GSCs): 59](#_Toc206759150)

[**9.6** Complexity Adjusted Factors 60](#_Toc206759151)

[**9.7** Adjusted function point 60](#_Toc206759152)

[**9.8** Cost & Effort Estimation 60](#_Toc206759153)

[**9.8.1** Effort estimation 61](#_Toc206759154)

[**9.8.2** 17.7 Total Cost 61](#_Toc206759155)

[10. CRITICAL PATH METHOD 62](#_Toc206759156)

[**10.1** Involved Activities 62](#_Toc206759157)

[**10.2** Determine the sequence of the activities 62](#_Toc206759158)

[**10.3** 13.4 Estimate Activity Completion Time 63](#_Toc206759159)

[**10.4** Network Diagram 64](#_Toc206759160)

[**10.5** Identify the Critical Path 64](#_Toc206759161)

[**10.6** Gantt Chart 64](#_Toc206759162)

[11. APPLICATTION ARCHITECTURE 65](#_Toc206759163)

[12. TESTING 66](#_Toc206759164)

[**12.1** User Test Case Description 66](#_Toc206759165)

[13. SCREEN IMAGES 68](#_Toc206759166)

[**13.1** Welcome screen 68](#_Toc206759167)

[**13.2** Home screen 68](#_Toc206759168)

[**13.3** Admin login screen 69](#_Toc206759169)

[**13.4** Sign up screen 69](#_Toc206759170)

[**13.5** Login in screen 70](#_Toc206759171)

[**13.6** OTP screen 70](#_Toc206759172)

[**13.7** Forget password screen 71](#_Toc206759173)

[**13.8** Create profile screen 71](#_Toc206759174)

[**13.9** User Dashboard screen 72](#_Toc206759175)

[**13.10** Admin Dashboard screen 72](#_Toc206759176)

[**13.11** Block profile list 73](#_Toc206759177)

[**13.12** User profile screen 73](#_Toc206759178)

[**13.13** Search donor screen 74](#_Toc206759179)

[**13.14** Donor list screen 74](#_Toc206759180)

[**13.15** Donor profile screen 75](#_Toc206759181)

[14. TRACEABILITY MATRIX 76](#_Toc206759182)

List of Figure

[Figure 1: User Use Case 22](#_Toc206759218)

[Figure 2: admin Use Case 28](#_Toc206759219)

[Figure 3: user procedure 32](#_Toc206759220)

[Figure 4: admin procedure 33](#_Toc206759221)

[Figure 5: class diagram 34](#_Toc206759222)

[Figure 6: user activity diagram 35](#_Toc206759223)

[Figure 7: admin activity diagram 36](#_Toc206759224)

[Figure 8: user state transition diagram 37](#_Toc206759225)

[Figure 9: admin state transition diagram 38](#_Toc206759226)

[Figure 10: sequence sign up 39](#_Toc206759227)

[Figure 11: sequence create profile 40](#_Toc206759228)

[Figure 12: sequence login 40](#_Toc206759229)

[Figure 13: sequence forget password 41](#_Toc206759230)

[Figure 14: sequence need blood 42](#_Toc206759231)

[Figure 15: sequence Profile 43](#_Toc206759232)

[Figure 16: sequence log out 44](#_Toc206759233)

[Figure 17: sequence login 45](#_Toc206759234)

[Figure 18: sequence forgot password 46](#_Toc206759235)

[Figure 19: sequence Blocked account 47](#_Toc206759236)

[Figure 20: sequence logout 47](#_Toc206759237)

[Figure 21: collaboration sign up 48](#_Toc206759238)

[Figure 22: collaboration create profile 49](#_Toc206759239)

[Figure 23: collaboration sign in 49](#_Toc206759240)

[Figure 24: collaboration forget password 50](#_Toc206759241)

[Figure 25: collaboration need blood 51](#_Toc206759242)

[Figure 26: collaboration profile 52](#_Toc206759243)

[Figure 27: collaboration logout 52](#_Toc206759244)

[Figure 28: collaboration login 53](#_Toc206759245)

[Figure 29: collaboration forgot password 54](#_Toc206759246)

[Figure 30: collaboration Block account 55](#_Toc206759247)

[Figure 31:collaboration logout 55](#_Toc206759248)

[Figure 32: data flow diagram 56](#_Toc206759249)

[Figure 34: network diagram 64](#_Toc206759250)

[Figure 35: Gantt Chart 64](#_Toc206759251)

[Figure 36: architecture 65](#_Toc206759252)

# INTRODUCTION

“Blood Donation App” is designed to facilitate the process of blood donation and ensuring that the blood is available in emergencies. The Blood Bank Application is a platform designed to make the process of blood donation easy for everyone, also ensuring timely access to blood during emergencies. This app creates bridges between blood donors and recipients by using modern technology to create efficient, and user-friendly experience. The application allows users to register as donors or recipients, create detailed profiles. Donors can update their availability. Recipients can also search for their matching blood type donors. The app provides services to connect donors with those in need, ensuring quick and accurate matches during critical situations. This app simplifies the blood donation process; the Blood Bank Application aims to increase blood donation rates and save lives. This app is a step forward in using technology for social good, making blood donation more easy, efficient, and impactful.

## Scope

The Blood Bank Application is designed to address the challenges of blood donation and distribution. Its scopes key features are written below:

* Enable donors to create and update their profiles with details like name, blood type, location, and contact information.
* Provide a search feature for recipients to find donors based on blood type and city.
* Provide calling features for direct communication between donors and recipients.
* Ensure the app is accessible in both urban and rural areas.

## Purpose

The primary purpose of the Blood Bank Application is to save lives by ensuring timely access to blood during emergencies.

* Provide recipients with a quick and efficient way to find donors.
* Reduce the time and effort required to locate and connect with donors.
* Raise awareness about the importance of regular blood donation.

# PROBLEM DEFINATION

## Problem Statement

Timely access to blood in emergencies is very difficult especially during situations such as accidents, surgeries, or medical conditions like anemia, thalassemia, and cancer. However, the current system for blood donation faces significant challenges, creating a gap between the demand for blood and its availability. Blood banks and hospitals often rely on manual processes, which are inefficient and time-consuming, leading to delays in critical situations. Additionally, finding a donor with the right blood type and location during emergencies is a very difficult task, often resulting in delays for patients in need. Another challenge is the low participation of people in donation of blood, as many people are unaware of the urgent need for blood. Communication between donors, recipients, and healthcare providers is also uncoordinated, there is no convenience for donors to respond to urgent requests or for recipients to contact donors directly. This lack of coordination often leads to missed opportunities for life-saving donations. These challenges highlight the need for an innovative solution that simplifies the blood donation process, improves communication, and ensures timely access to blood. The Blood Bank Application aims to address these issues.

## Deliverable and Development Requirements

* **Front-end Development:** The applications front end requires expertise in app development languages such as React Native, CSS and JavaScript. I develop visually appealing interfaces and responsive layouts too enhance user experience.
* **Back-end Development:** the backend development requires in server-side programming languages e.g. Python, Django. The backend team implements different functions such as authentication.
* **Database Management:** The app needs an effective database management system to store and retrieve data. The development team selected appropriate database technology such as MySQL DB.
* **Deliverables and Quality Assurance:** Testing can be conducted in various ways and various stages to identify and resolve any function issue. Quality Assurance process is implemented to ensure that the application meets the highest standards of performance and experience.

### Development required tools

The development required tools that are being used in this application development are given below:

1. **Design tools:**
   * MS Word
   * MS PowerPoint
   * MS Project
   * Draw.io
2. **Development tool**:
   * VS. code

### Languages

* JavaScript
* HTML
* CSS
* Django
* React Native

# FEASIBILITY REPORTS

## Project Feasibility Report

The blood bank app enables users to communicate with donors and recipients without any problem. The app serves as an essential platform for the efficient collection, recording and distribution of essential information regarding blood donation. It includes features like user registration, user profile, donor search and updating donor data.

* Technical Feasibility
* Operational Feasibility
* Economic Feasibility
* Schedule Feasibility
* Specification Feasibility
* Legal & Ethical Feasibility

## Technical Feasibility

* The technologies required to develop the Blood Bank Application are available and widely used. These include:
* Frontend: HTML, CSS, JavaScript, React Native.
* Backend: Django, Python
* Database: MySQL.

## Operational Feasibility

The app simplifies the process of connecting blood donors with recipients, making it highly likely to be accepted and adopted by users, including donors, recipients, and healthcare providers. Its user-friendly features, such as easy registration and donor search enhance the overall user experience and encourage participation. By using technology, we can make bridge between blood supply and demand, the app is not only practical but also has the potential to save lives.

## Economic Feasibility

Costs and benefits make up the two basic components of economic feasibility. Economic feasibility determines cost benefit in developing this application. For calculating the development cost, we analyzed cost categories through personal cost, computer usage, supply and equipment expenses. The app has potential to save lives and improve healthcare efficiency, the investment is justified, making the project economically viable and sustainable in the long term.

## Schedule Feasibility

Schedule to perform different tasks has been created and it is predicted that each activity will be perform on its specified time. I hopeful that my project is completed on its scheduled time. The project can be divided into clear phases, such as requirement gathering, planning, designing, coding, integration, testing and deployment. With proper project management we can accomplish our tasks on time.

## Specification Feasibility

By keeping in mind that the specification feasibility it is assured that my project also meets the essential specifications for the development of our project. All the requirements should be clear and well-defined so application can be developed easily. The hardware and software specification for this project are also achievable.

## Legal & Ethical Feasibility

After the project is finished, there won’t be any unlawful or immoral activities. The project is definitely legal because it won’t cause any issue for others.

# QUALITY ATTRIBUTES

## Reliability

The app should have work consistently without any error and crashes. The app interface data accurately stored and retrieve data.

## Usability

Usability deals in this application. It is easy to understand and use for user. The app has user friendly and simple interface.

## Security

User’s data and personal information must be protected through encryption and OTP authentication. Unauthorized user can’t get any access to user personal data.

## Performance

The quality of app performance should be successful and error free. It should complete its functions accurately and effectively. Application must work according to its requirement.

## Availability

The app should be accessible 24/7, especially during emergencies. Server downtime should be minimized and user can access the system whenever they needed.

# SPECIFIC REQUIREMENTS

## **User Registration and Authentication:**

In this app users (donors, recipients) can sign up using email and password. And also, users do OTP-based verification for secure authentication that no unknown user can access to personal information

## **Profile Creation:**

In this application users can create their own profiles and enter their personal information in profile form like Name, age, blood type, Address and contact information. Then user’s profiles data get stored in a secure database.

## **Donor Search:**

Recipients can search for donors based on: Blood type, city and availability. Then the system shows filtered data of matching donor profile with all details. Then the user can select any profile from that donor list.

## Donor Profile:

The donor can also make updating and addition in his profile. The donor can update its availability as he is available to donate blood or not. Then he can add or update his reports. Also, he can change his last donation date.

## **Communication Tools:**

Recipients can contact donors directly via call. Donors contact information is given in donor profile recipient can make contact with donor any time. Hospitals can coordinate with donors for blood collection.

# EXTERNAL INTERFACE REQUIREMENT

## 6.1 User Interfaces

The interface has simple and user-friendly interface for easy navigation of user and system. The languages that are used to create user friendly are these HTML, CSS, and JavaScript.

## 6.2 Software Interfaces

Visual studio code is used as software interface to design this app. The App is developed using Django and React Native ensuring scalable architecture. MySQL manages data storage and retrieval.

## 6.3 Hardware Interfaces

The Blood Bank App can be accessed on various devices, including:

* Tablets
* Smartphones

# REQUIREMENT IDENTIFYING TECHNIQUES

## User Use Case:

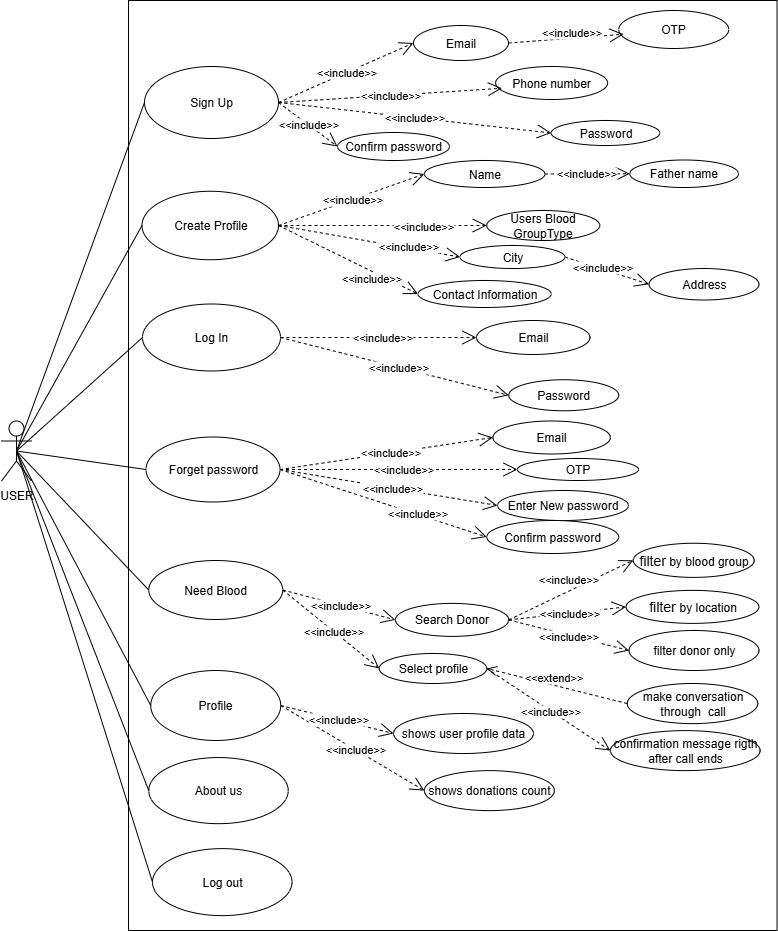


Figure 1: User Use Case

List of possible use cases of User:

* UC\_Sign up
* UC\_Create profile
* UC\_Log in
* UC\_Forget password
* UC\_Need blood
* UC\_ Profile
* UC\_Logouts

## USER USE CASE DESCRIPTION

|  |  |
| --- | --- |
| Use Case ID: | UC-1 |
| Use Case Name: | Sign up. |
| Description: | This Use Case is about user should sign up first. |
| Priority: | High |
| Actor: | User |
| Pre-Condition: | The user signs up first to login an account. |
| Post-Condition: | The user registered successfully. |
| Basic-Flow: | * Enter user name. * Enter email. * Enter OTP. * Enter password. * Submit. |
| Exceptional-Flow: | * Entered email is not valid. * Username has already been registered. * Password is too weak. * Error message displayed to the user. |
| Business-Rules: | None. |

### User Sign up

|  |  |
| --- | --- |
| Use Case ID: | UC-2 |
| Use Case Name: | Create profile. |
| Description: | This Use Case is used to make user profile. |
| Priority: | High |
| Actor: | User |
| Pre-Condition: | The user gives personal details to develop profile. |
| Post-Condition: | User profile generated successfully. |
| Basic-Flow: | * Enters the info in the given form. * System stores the information of the user. * Use case ends successfully. |
| Exceptional-Flow: | The form is not filled properly. |
| Business-Rules: | None |

### Create Profile

### User Login

|  |  |
| --- | --- |
| Use Case ID: | UC-3 |
| Use Case Name: | Log in |
| Description: | This Use Case describe user login. |
| Priority: | High |
| Actor: | User |
| Pre-Condition: | User fill the credentials to login. |
| Post-Condition: | Access in granted. |
| Basic-Flow: | * Donor enters email and password. * If authenticated then successfully logged in. * Use case ends successfully. |
| Exceptional-Flow: | * Invalid credentials. * Internet connectivity issue. |
| Business-Rules: | Authorized user has right to login. |

### Forget password

|  |  |
| --- | --- |
| Use Case ID: | UC-4 |
| Use Case Name: | Forget password |
| Description: | This Use Case is about changing user account password. |
| Priority: | Medium |
| Actor: | User |
| Pre-Condition: | The user email must be registered in system. |
| Basic-Flow: | * User clicks on “forget password”. * User entered registered email. * System sent OTP to that email. * User enter OTP and set new password. * System confirm password and reset it. |
| Post-Condition: | The user password is updated. |
| Exceptional-Flow: | * If the email invalid, system show error. * If OTP is not correct, system shows re-enter. |
| Business-Rules: | None. |

### Need Blood

|  |  |
| --- | --- |
| Use Case ID: | UC-5 |
| Use Case Name: | Need blood |
| Description: | This Use Case is about need blood. |
| Priority: | High |
| Actor: | User |
| Pre-Condition: | User click on need blood to open page. |
| Post-Condition: | Successfully shows the page. |
| Basic-Flow: | * Search for blood donor by city. * Search by blood type. * Select any donor profile. * Shows donor profile with all details. * Initiate through chat * Confirm blood is donated |
| Exceptional-Flow: | * Could not find matching donor profile. |
| Business-Rules: | None. |

### Profile

|  |  |
| --- | --- |
| Use Case ID: | UC-6 |
| Use Case Name: | Profile |
| Description: | This Use Case is used to show personal detail and request count. |
| Priority: | High |
| Actor: | User |
| Pre-Condition: | User open the use case. |
| Post-Condition | Page loaded successfully. |
| Basic-Flow | * User can see the personal details. * User can also check blood request count. |
| Exceptional-Flow | Data could not show or any error occur. |
| Business-Rules | None. |

### User logout

|  |  |
| --- | --- |
| Use Case ID: | UC-7 |
| Use Case Name: | Logout |
| Description: | This Use Case describe the phenomena of logout. |
| Priority: | High |
| Actor: | User |
| Pre-Condition: | User must have logged in the system. |
| Post-Condition | Successfully logged out |
| Basic-Flow | * User start logout task by clicking on “Logout” button. * The system will display message to ensure the logout. * Use case ends successfully. |
| Exceptional-Flow | * Internet connection issue. |
| Business-Rules | None. |

## Admin Use Case

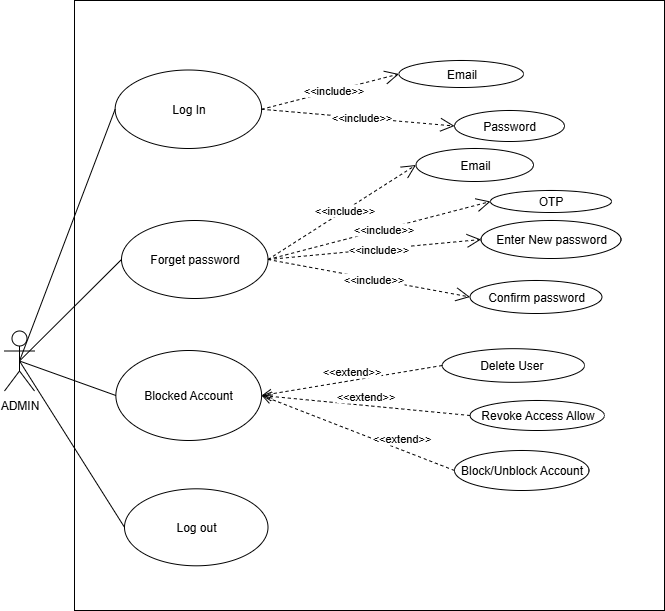


Figure 2: admin Use Case

List of admins use case:

UC\_LogIn

UC\_Forgot password

UC\_Blocked accounts

UC\_Logout

## Admin Use Case Description

### Admin Login

|  |  |
| --- | --- |
| Use Case ID: | UC-1 |
| Use Case Name: | Log in |
| Description: | This Use Case describe Admin login. |
| Priority: | High |
| Actor: | Admin |
| Pre-Condition: | Admin fill the credentials to login. |
| Post-Condition: | Access in granted. |
| Basic-Flow: | * Admin enters email and password. * If authenticated then successfully logged in. * Use case ends successfully. |
| Exceptional-Flow: | * Invalid credentials. * Internet connectivity issue. |
| Business-Rules: | None |

|  |  |
| --- | --- |
| Use Case ID: | UC-2 |
| Use Case Name: | Forget password |
| Description: | This Use Case is about changing Admin account password. |
| Priority: | Medium |
| Actor: | Admin |
| Pre-Condition: | The admin email must be registered in system. |
| Basic-Flow: | * Admin clicks on “forget password”. * Admin entered registered email. * System sent OTP to that email. * Admin enter OTP and set new password. * System confirm password and reset it. |
| Post-Condition: | The admin password is updated. |
| Exceptional-Flow: | * If the email invalid, system show error. * If OTP is not correct, system shows re-enter. |
| Business-Rules: | None. |

### Forgot password

### Blocked account

|  |  |
| --- | --- |
| Use Case ID: | UC-3 |
| Use Case Name: | Blocked account |
| Description: | This Use Case is about admin panel and admin features. |
| Priority: | Medium |
| Actor: | Admin |
| Pre-Condition: | The admin can make changes in blocked users’ account. |
| Basic-Flow: | * Admin clicks on “delete user” and can del user from app permanently. * Admin clicks on “revoke/allow access to give access to user and revoke access. * Admin click on “block/unblock user” to block and unblock user. |
| Post-Condition: | Admin changes will show on screen |
| Exceptional-Flow: | * Changes will be stored in database. |
| Business-Rules: | None. |

### Logout

|  |  |
| --- | --- |
| Use Case ID: | UC-4 |
| Use Case Name: | Logout |
| Description: | This Use Case describe the phenomena of logout. |
| Priority: | High |
| Actor: | Admin |
| Pre-Condition: | Admin must have logged in the system. |
| Basic-Flow | * start logout task by clicking on “Logout” button. * The system will display message to ensure the logout. * Use case ends successfully. |
| Post-Condition | Successfully logged out. |
| Exceptional-Flow | * Internet connection issue. |
| Business-Rules | None. |

# DESIGN MODEL

## User procedure

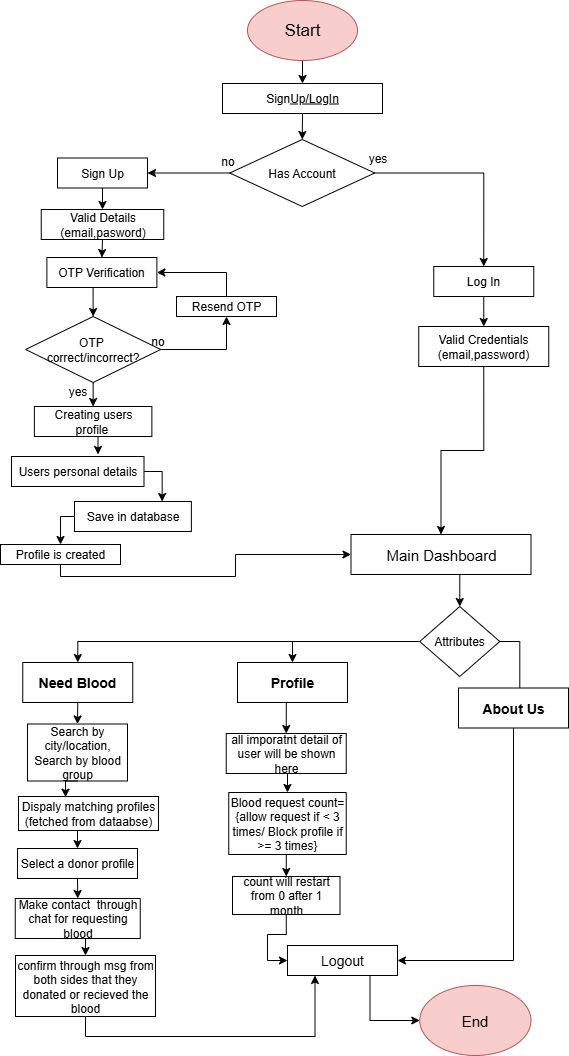


Figure 3: user procedure

## Admin procedure

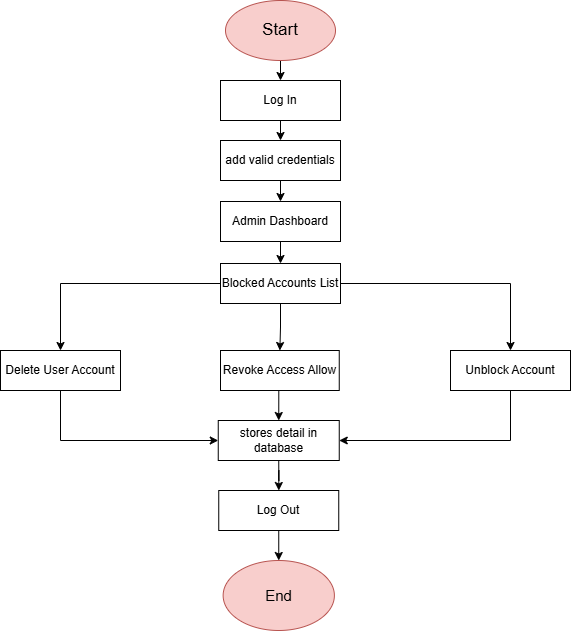


Figure 4: admin procedure

## Class Diagram

## 

Figure 5: class diagram

## User Activity Diagram

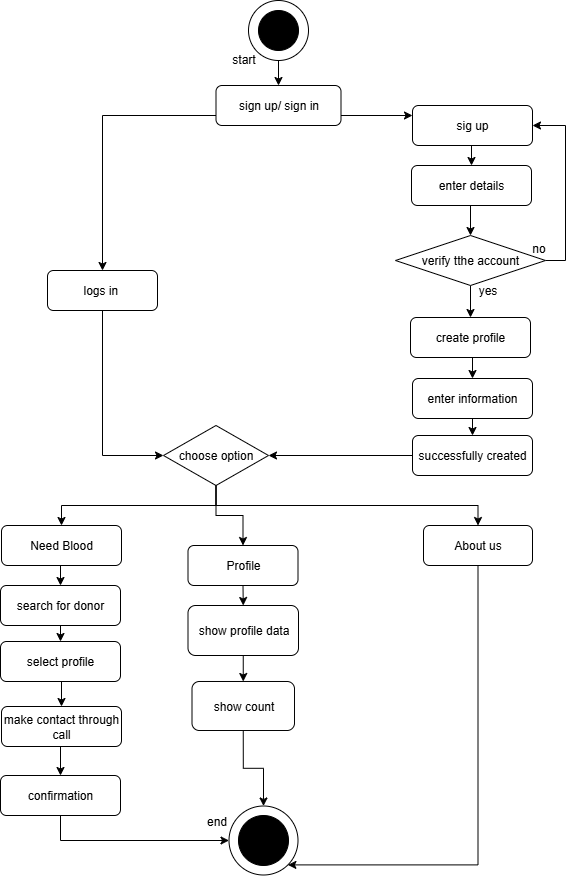


Figure 6: user activity diagram

## Admin Activity Diagram

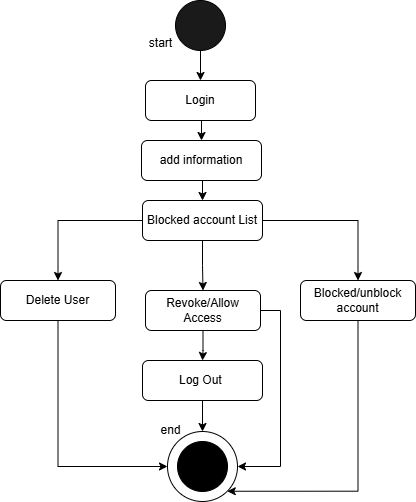


Figure 7: admin activity diagram

## User State Transition Diagram

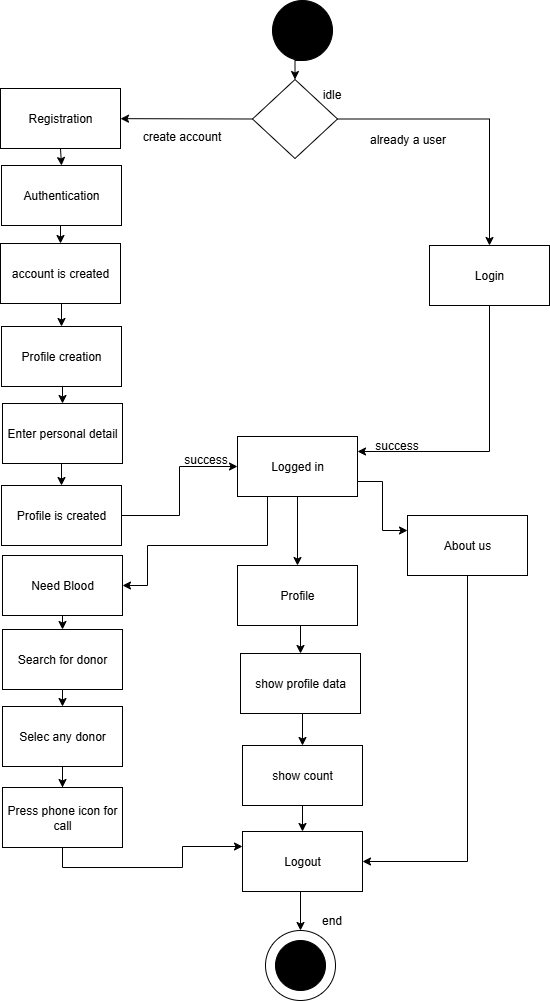


Figure 8: user state transition diagram

## Admin State Transition Diagram

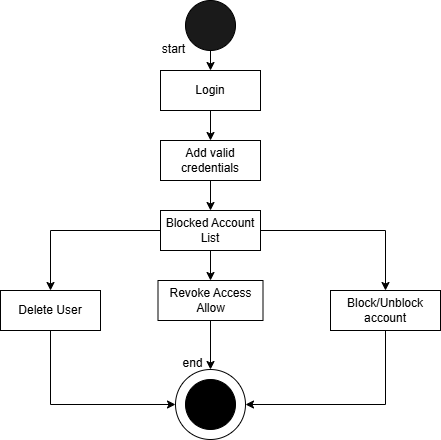


Figure 9: admin state transition diagram

## User Sequence Diagram

### Sign up

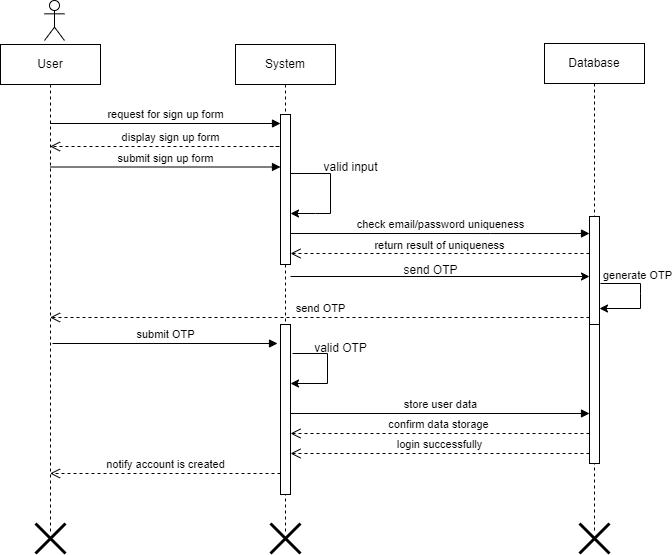


Figure 10: sequence sign up

### Create profile

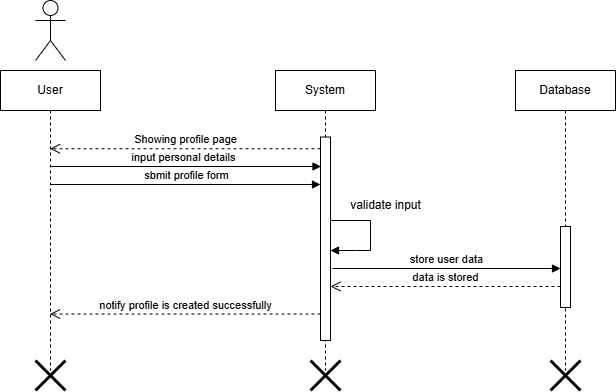


Figure 11: sequence create profile

### Login

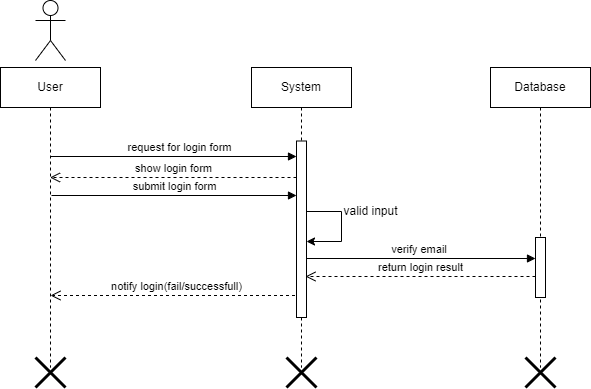


Figure 12: sequence login

### Forget password

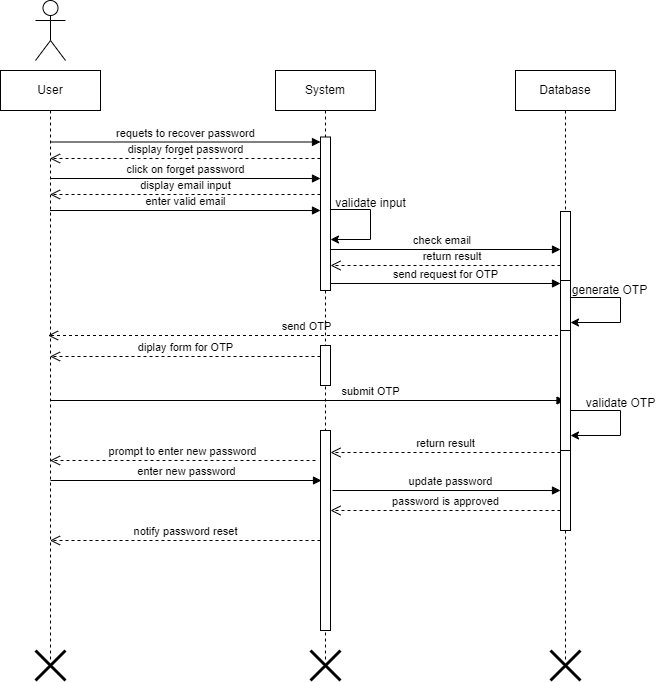


Figure 13: sequence forget password

### Need Blood

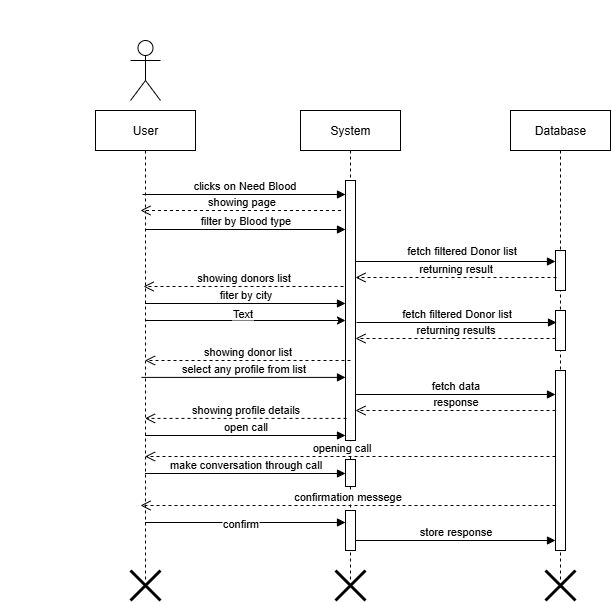


Figure 14: sequence need blood

### Profile

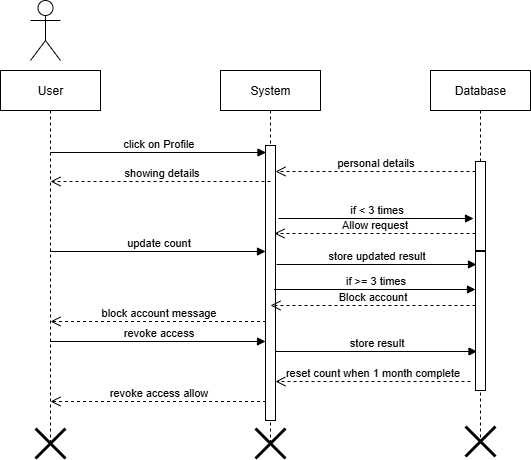


Figure 15: sequence Profile

### Log out

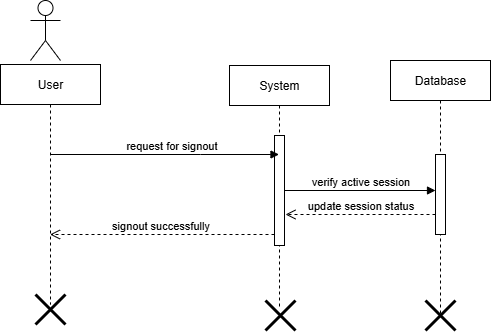


Figure 16: sequence log out

## Admin Sequence Diagram

### Log in

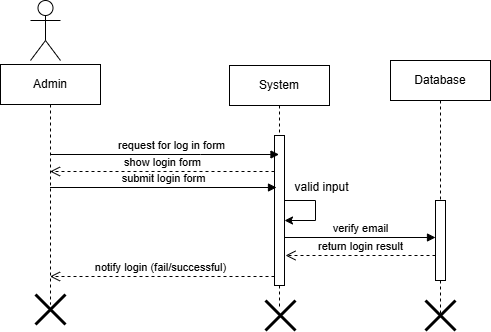


Figure 17: sequence login

### Forgot password

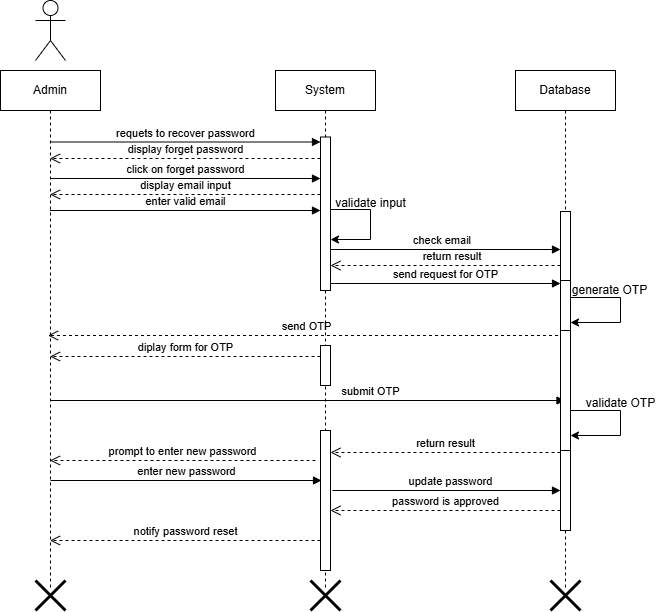


Figure 18: sequence forgot password

### Blocked Account

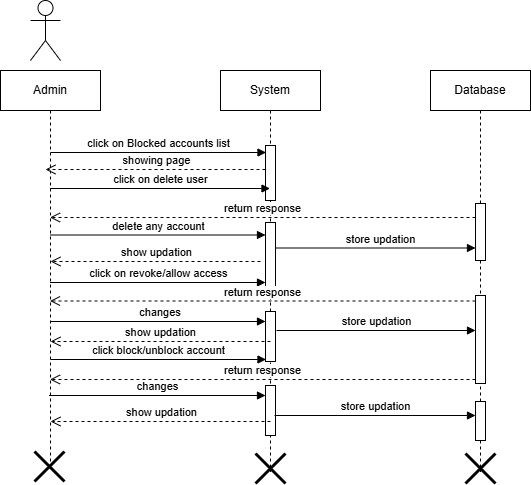


Figure 19: sequence Blocked account

### Log out

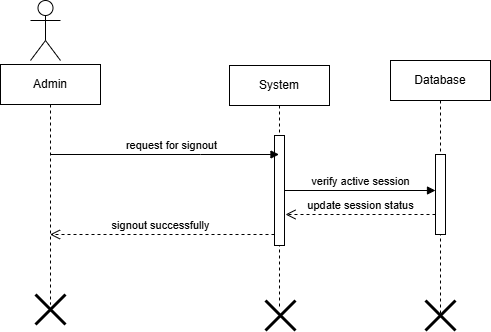


Figure 20: sequence logout

## User Collaboration Diagram

### Sign up

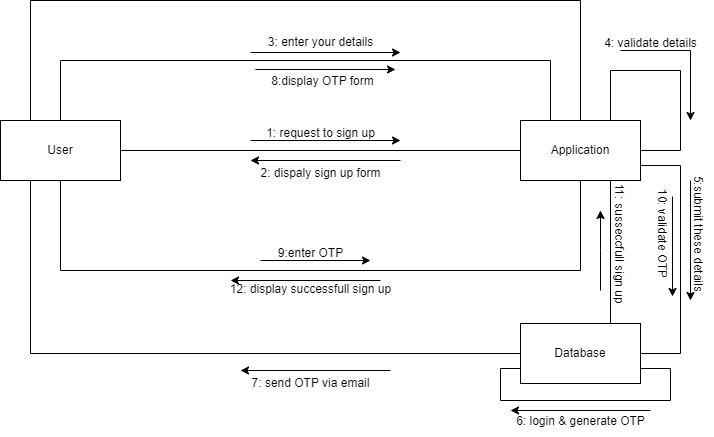


Figure 21: collaboration sign up

### Create Profile

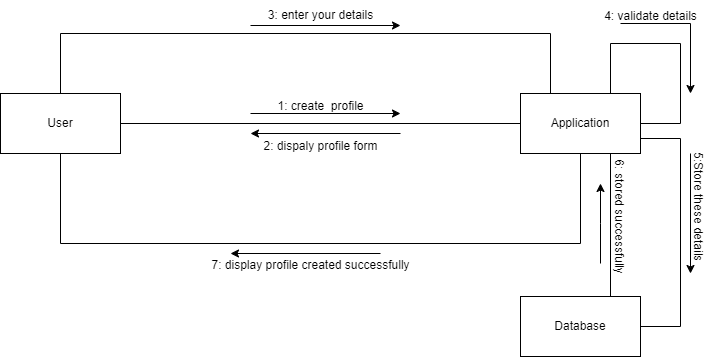


Figure 22: collaboration create profile

### Sign In

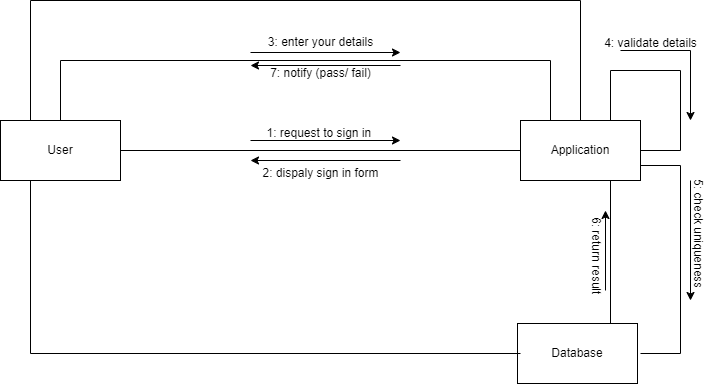


Figure 23: collaboration sign in

### Forget Password

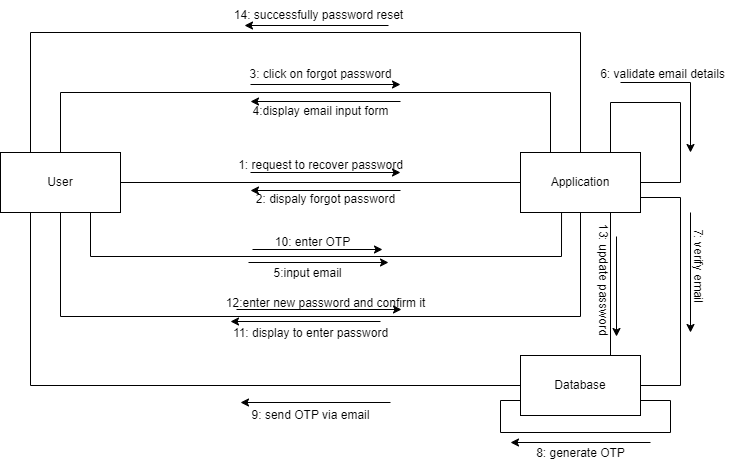


Figure 24: collaboration forget password

### Need Blood

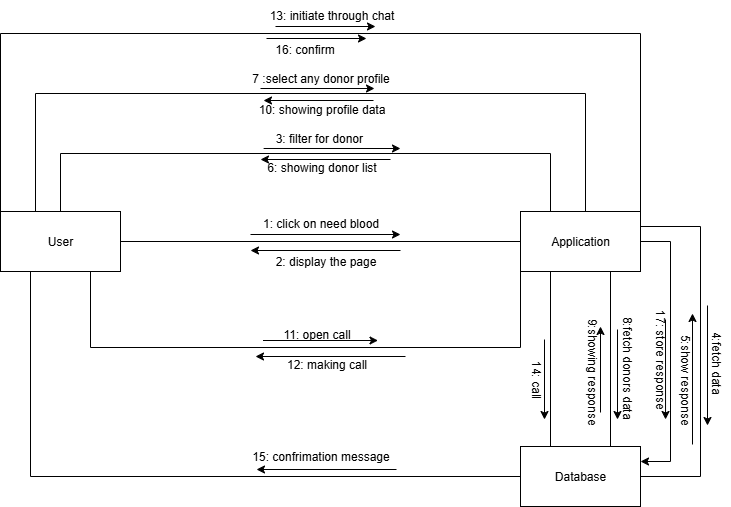


Figure 25: collaboration need blood

### Profile

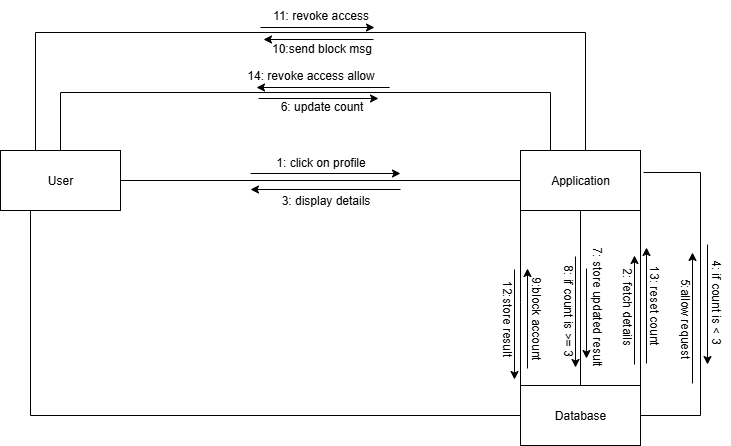


Figure 26: collaboration profile

### Log out

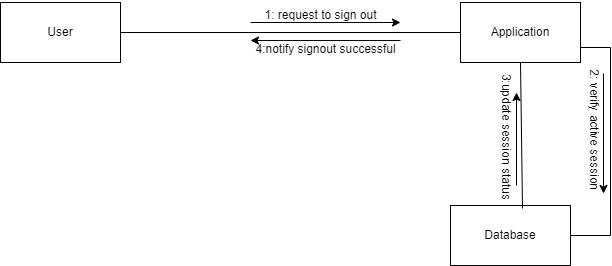


Figure 27: collaboration logout

## Admin Collaboration Diagram

### Login

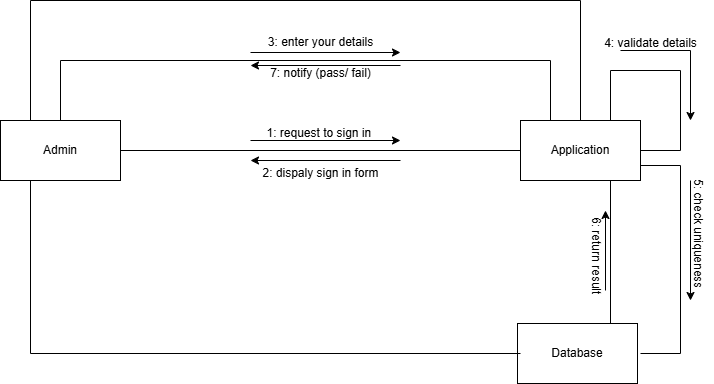


Figure 28: collaboration login

### Forgot password

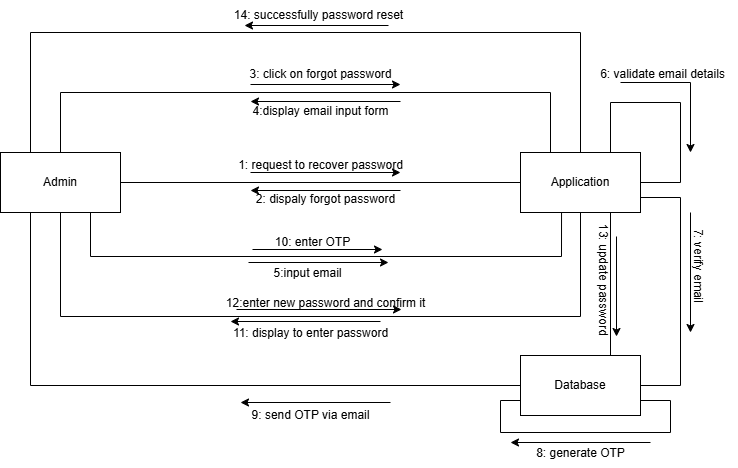


Figure 29: collaboration forgot password

### Blocked Account

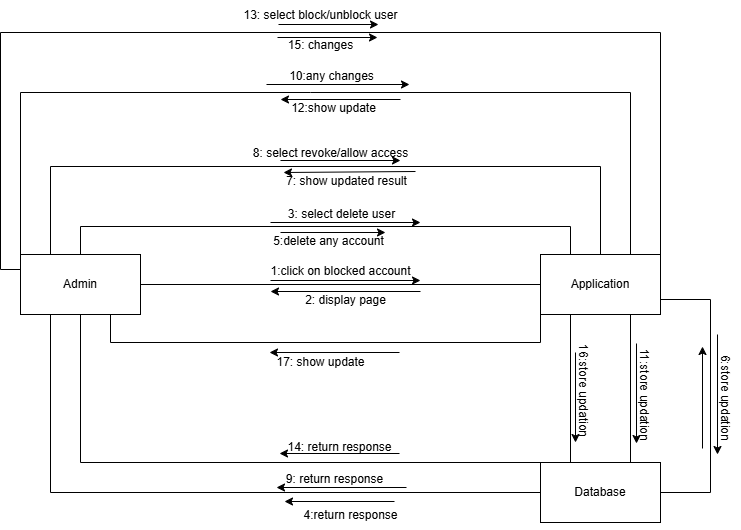


Figure 30: collaboration Block account

### Logout

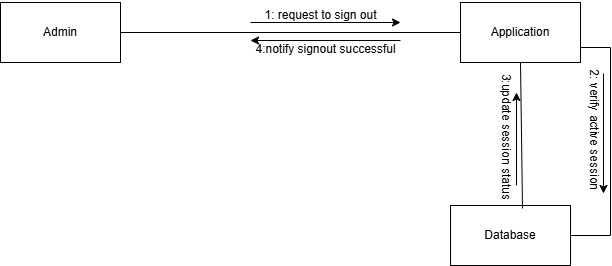


Figure 31:collaboration logout

## DFD Diagram

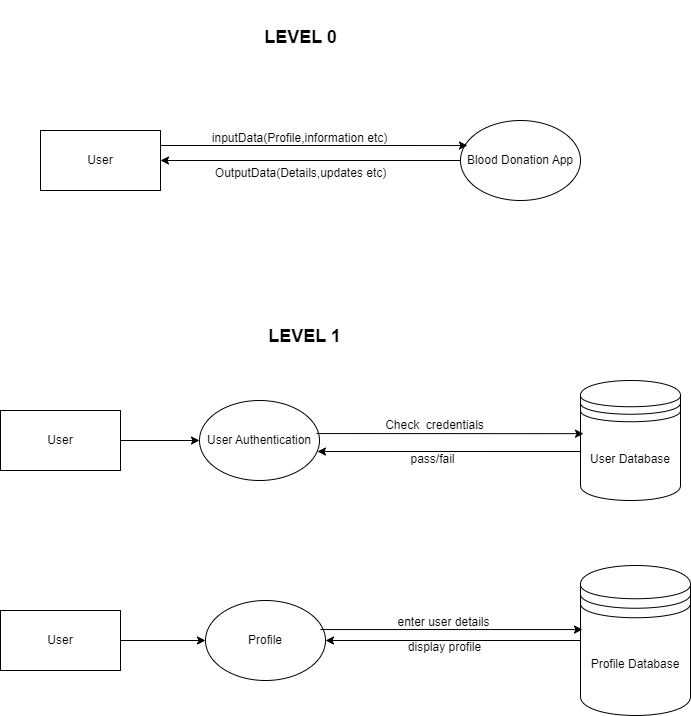


Figure 32: data flow diagram

# PROJECT COSTING

## Project Cost Estimation by Function Point Analysis

Values are defined below:

### External inputs (EI)

* User login
* Profile creation
* Update/add reports
* Update date
* Update availability

### External outputs (EO)

* Search for donor
* Display donor profile
* Display blood reports

### External inquiries (EI)

* Searching
* View donor profile
* View blood reports

### Internal logical files (ILF)

* User accounts
* Donor profile
* Blood reports

### External logical files (ELF)

* OTP service (external email)
* Call integration (external phone service)

## Weights of Five Functional Point Attributes

|  |  |  |  |
| --- | --- | --- | --- |
| Measurement  Parameters | Low | Average | High |
| Number of external inputs | 3 | 4 | 6 |
| Number of external outputs | 4 | 5 | 7 |
| Number of external inquiries | 3 | 4 | 6 |
| Number of internal files | 7 | 10 | 15 |
| Number of external interfaces | 5 | 7 | 10 |

## Calculating Function Point:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Type of component | Count | Complexity |  | |
| Low | Average | High |
| External inputs | 5 | 2\*3=6 | 1\*4=4 | 2\*6=12 |
| External outputs | 3 | 1\*4=4 | 1\*5=5 | 1\*7=7 |
| External inquires | 3 | 1\*3=3 | 2\*4=8 | 0\*6=0 |
| Internal files | 3 | 1\*7=7 | 1\*10=10 | 1\*15=15 |
| External files | 2 | 0\*5=0 | 2\*7=14 | 0\*10=0 |

|  |  |
| --- | --- |
| External inputs | 22 |
| External outputs | 16 |
| External inquires | 11 |
| Internal files | 32 |
| External files | 14 |
| Count total UFP: | 95 |

## Calculating Factors

|  |  |
| --- | --- |
| Sca | ling |
| 0 | Not applicable |
| 1 | Incidental |
| 2 | Moderate |
| 3 | Average |
| 4 | Significant |
| 5 | Strong influence |

## General System Characteristics (GSCs):

|  |  |  |
| --- | --- | --- |
| No. | Value of adjustment factors | Values |
| 1 | Data communications | 4 |
| 2 | Distributed data processing | 3 |
| 3 | Performance | 4 |
| 4 | Heavily used configuration | 3 |
| 5 | Transaction rate | 3 |
| 6 | On-Line data entry | 2 |
| 7 | End-user efficiency | 4 |
| 8 | On-Line update | 3 |
| 9 | Complex processing | 2 |
| 10 | Reusability | 4 |
| 11 | Installation ease | 4 |
| 12 | Operational ease | 4 |
| 13 | Multiple sites | 3 |
| 14 | Facilitate change | 3 |
| Total | ∑(Fi) | 46 |

Finally, calculations are given below

## Complexity Adjusted Factors

CAF = 0.65 + (0.01 \* GSC rating)

= 0.65 + (0.01 \* 46)

CAF = 1.11

## Adjusted function point

AFP = UFP \* CAF

= 95 \* 1.11

=105.45

~ 105

## Cost & Effort Estimation

Effort estimation using productivity factors.

Low (10 hours per FP)

Medium (7 hours per FP)

High (5 hours per FP)

Assuming Medium productivity rate.

### Effort estimation

Formula:

Effort = AFP \* Productivity rate

= 105 \* 7

=735 hours

### 17.7 Total Cost

Assuming hourly developer cost

Cost in Pakistan = 500 PKR/hour

Total cost:

Formula:

Total cost = Effort \* Cost per hour

= 735 \* 500

Total cost = 367,500 PKR

# CRITICAL PATH METHOD

## Involved Activities

The following are the activities involved in the project.

* Planning
* Requirement gathering
* Analysis
* Designing
* Coding
* Integration
* Testing
* Deployment

## Determine the sequence of the activities

There are many activities that are dependent on the completion of other activities. The dependencies are given below.

Planning None

Requirement gathering Planning

Analysis Planning, Requirement gathers

Design Analysis

Coding Design

Integration Coding

Testing Integration

Deployment Testing

|  |  |
| --- | --- |
| Activity ID | Dependency |
| A | None |
| B | A |
| C | A, B |
| D | C |
| E | D |
| F | E |
| G | F |
| H | G |

## 13.4 Estimate Activity Completion Time

|  |  |  |
| --- | --- | --- |
| Activity ID | Dependency | Duration (weeks) |
| A | None | 2 |
| B | A | 3 |
|  |  |  |
| C | A, B | 3 |
| D | C | 4 |
| E | D | 6 |
| F | E | 3 |
| G | F | 2 |
| H | G | 1 |

## Network Diagram

Figure 33: network diagram

## Identify the Critical Path

The critical path is given below:

A B C D E F G = 2+3+3+4+6+3+2 = 23

## Gantt Chart

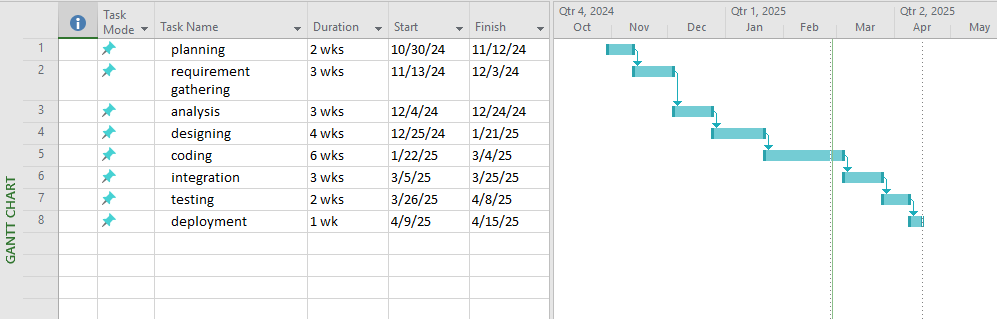


Figure 34: Gantt Chart

# APPLICATTION ARCHITECTURE

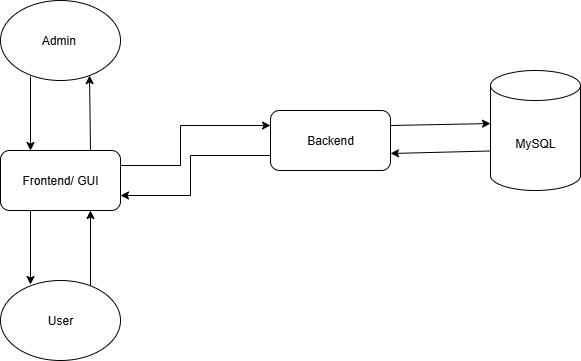


Figure 35: architecture

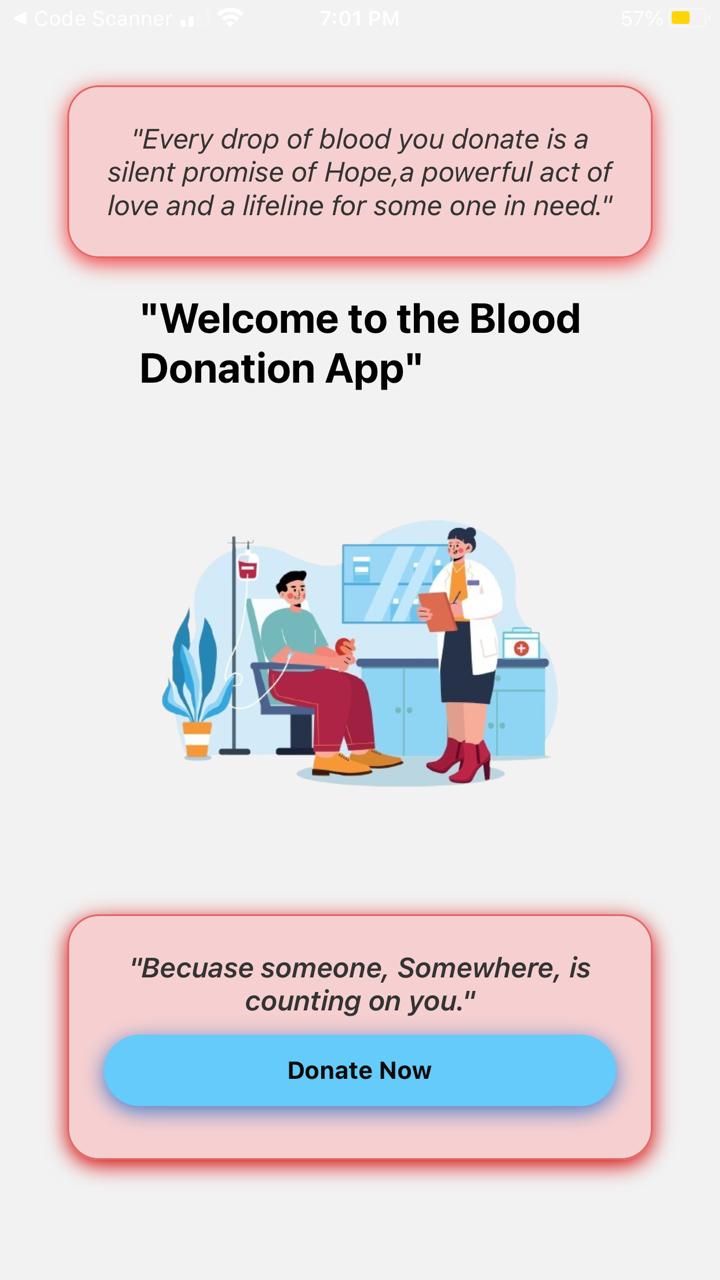
# TESTING

## User Test Case Description

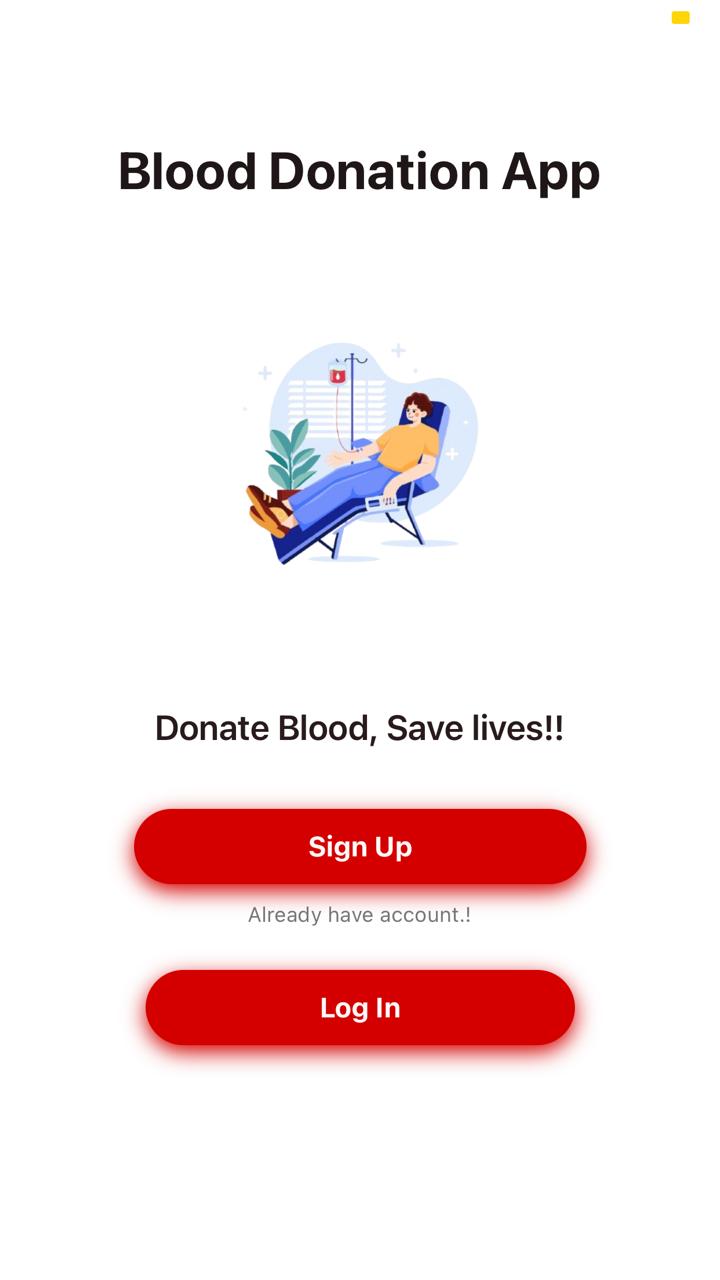
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test Case No:** | **Test Cases** | **Pre-Conditions** | **Expected Result** | **Actual result** | **Pass / Fail** |
| TC\_1 | Sign up | User must open screen to enter credentials. | Fill all requirements efficiently for sign up. | User registration is successful. | Pass. |
| TC\_2 | Sign in | User must register first for sign in. | Entre valid credentials. | User login is successful. | Pass. |
| TC\_3 | Create Profile | User must create his profile. | User fill all forms with valid data. | User profile created successfully. | Pass. |
| TC\_4 | Forget password | Enter registered email and enter OTP. | Enter new password and confirm it. | Password changed successfully. | Pass. |
| TC\_5 | Need blood | Search for donor. | Matching user not found. | Search again. | Pass. |
| Search for donor. | Matching donor list displayed. | Select any user. | Pass. |
| TC\_6 | Profile | Open profile screen. | Show users personal data  And counter for blood request. | Profile load successfully. | Pass. |
| TC\_7 | Blood request. | Make call and do contact with donor. | If he said yes select yes. | Count will update successfully. | Pass. |
| Make call but receiver didn’t pick it. | No contact with donor message will show. | Count will not be updated. | Pass. |
| TC\_8 | Logout | Click Logout button. | User logged out and redirected to sign in screen. | User successfully logged out. | Pass. |

# SCREEN IMAGES

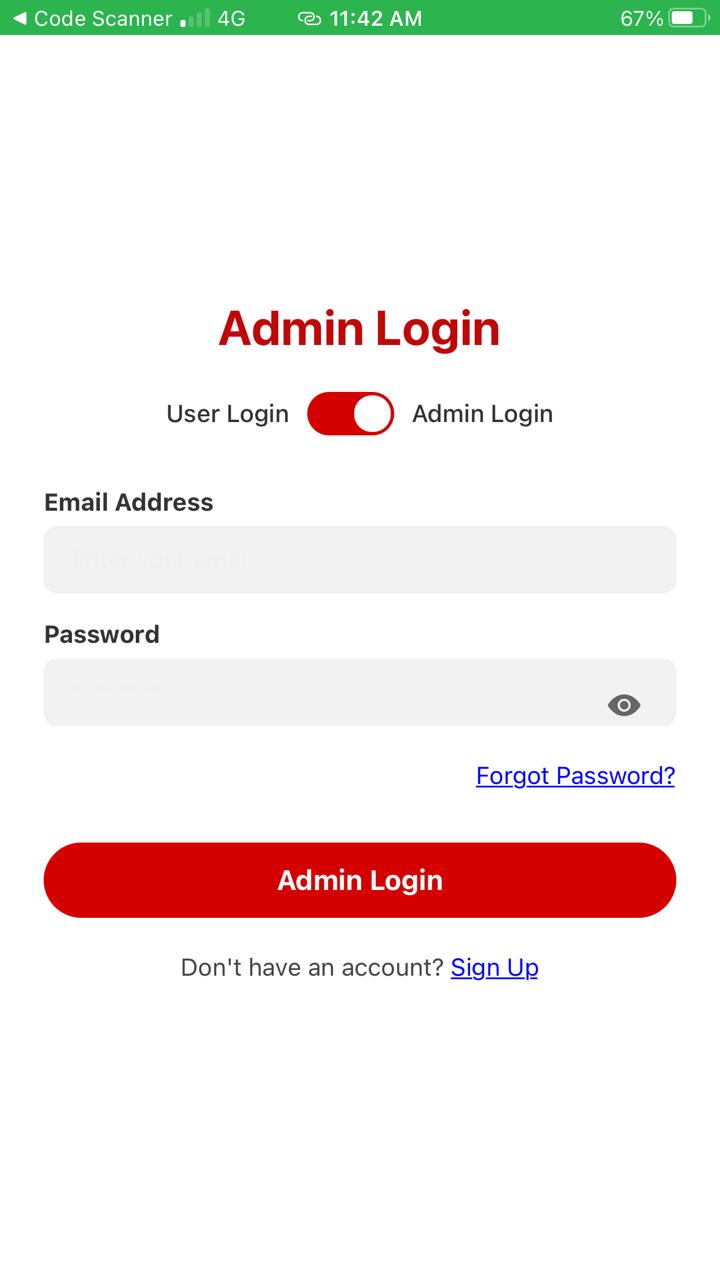
## Welcome screen



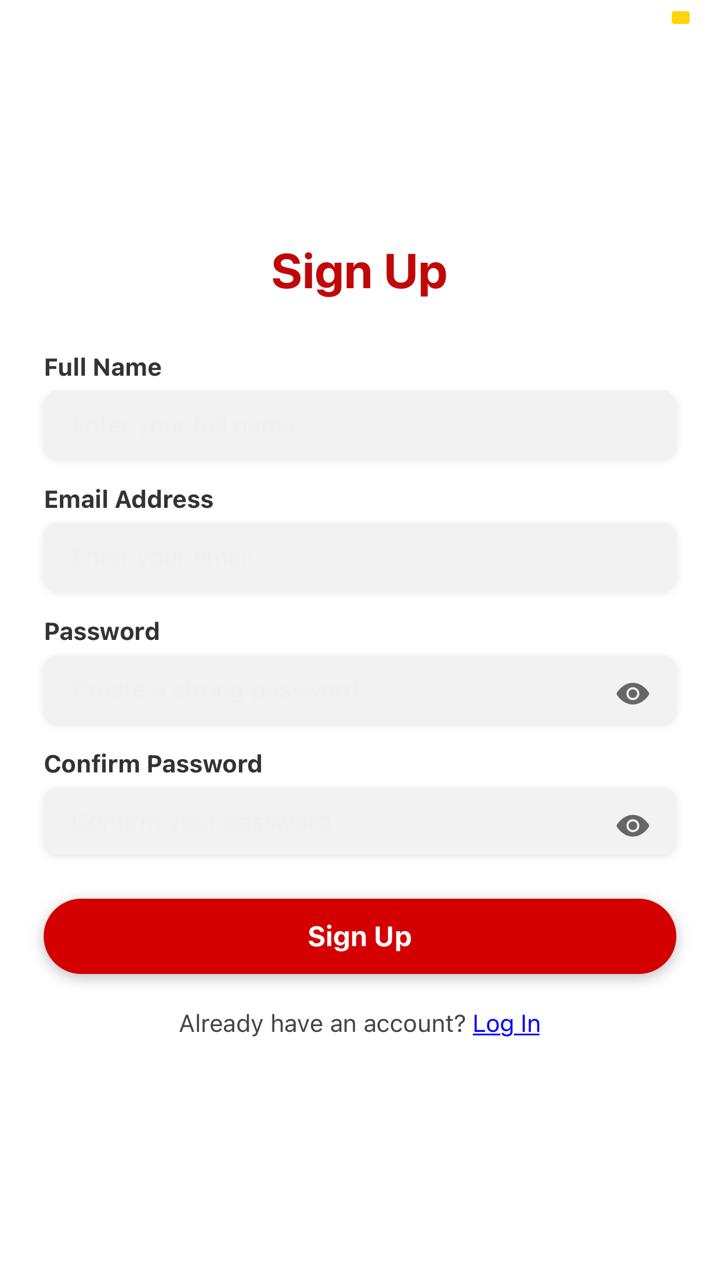
## Home screen



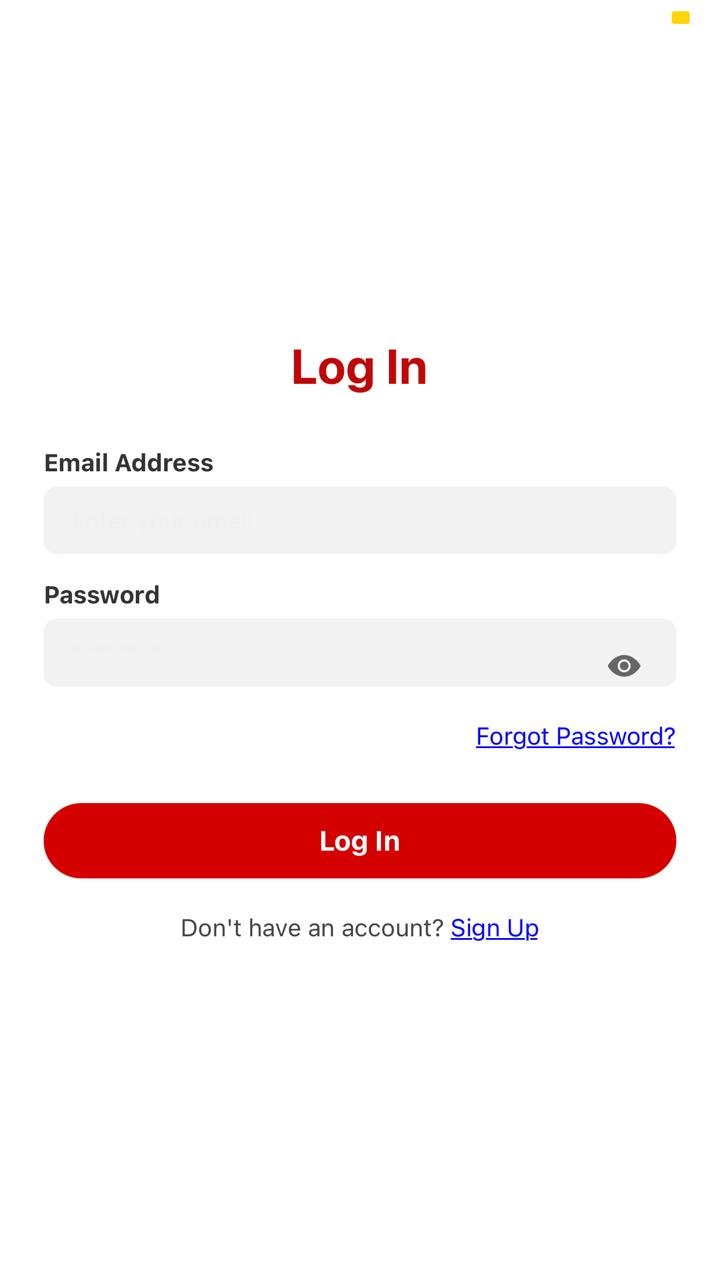
## Admin login screen



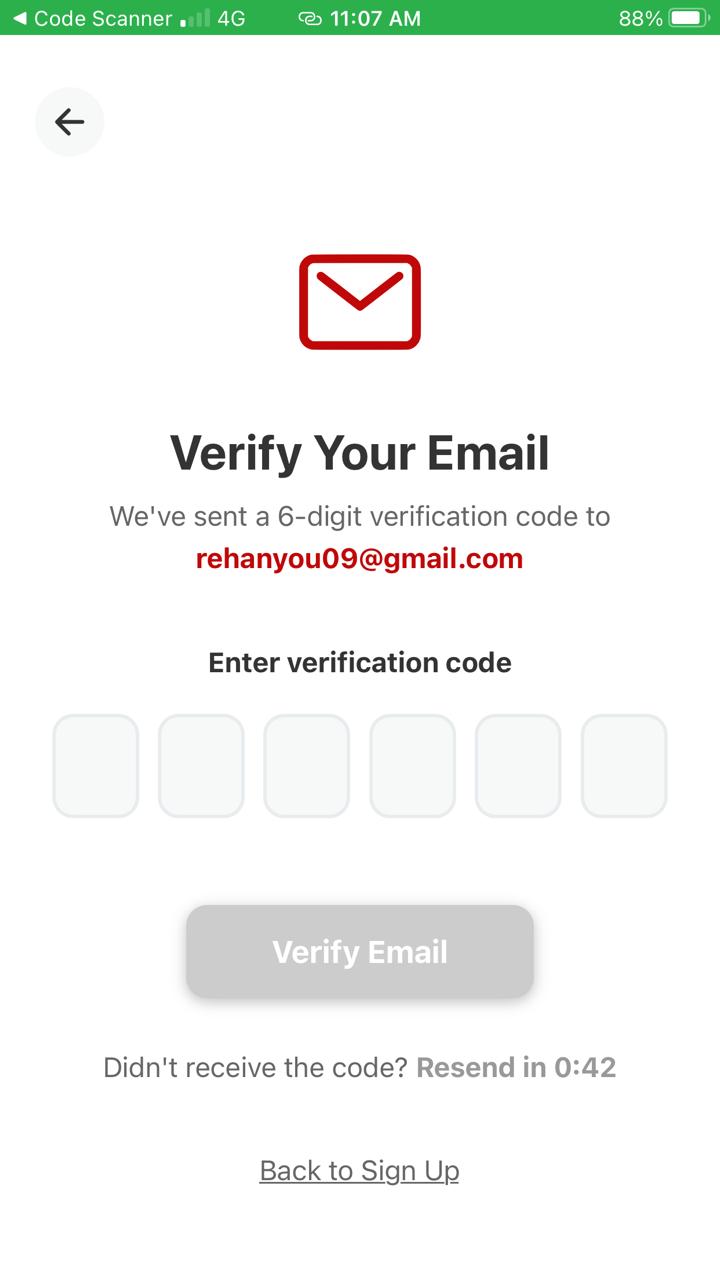
## Sign up screen



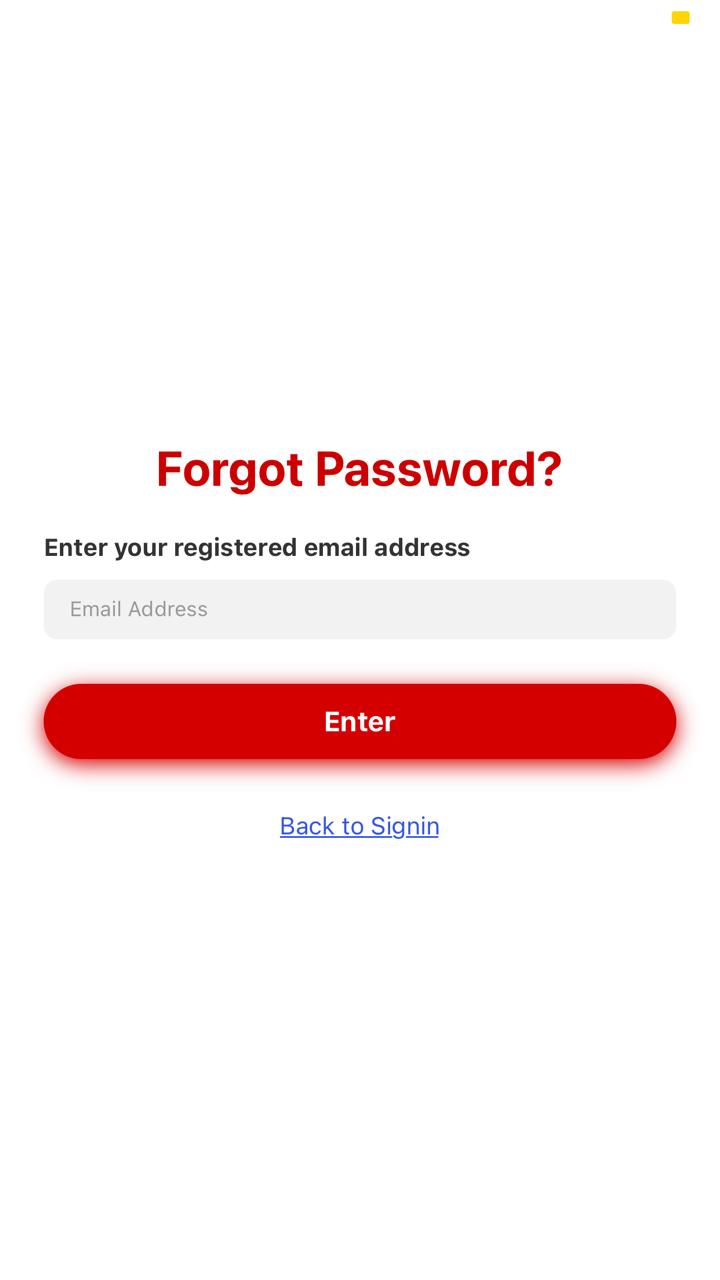
## Login in screen



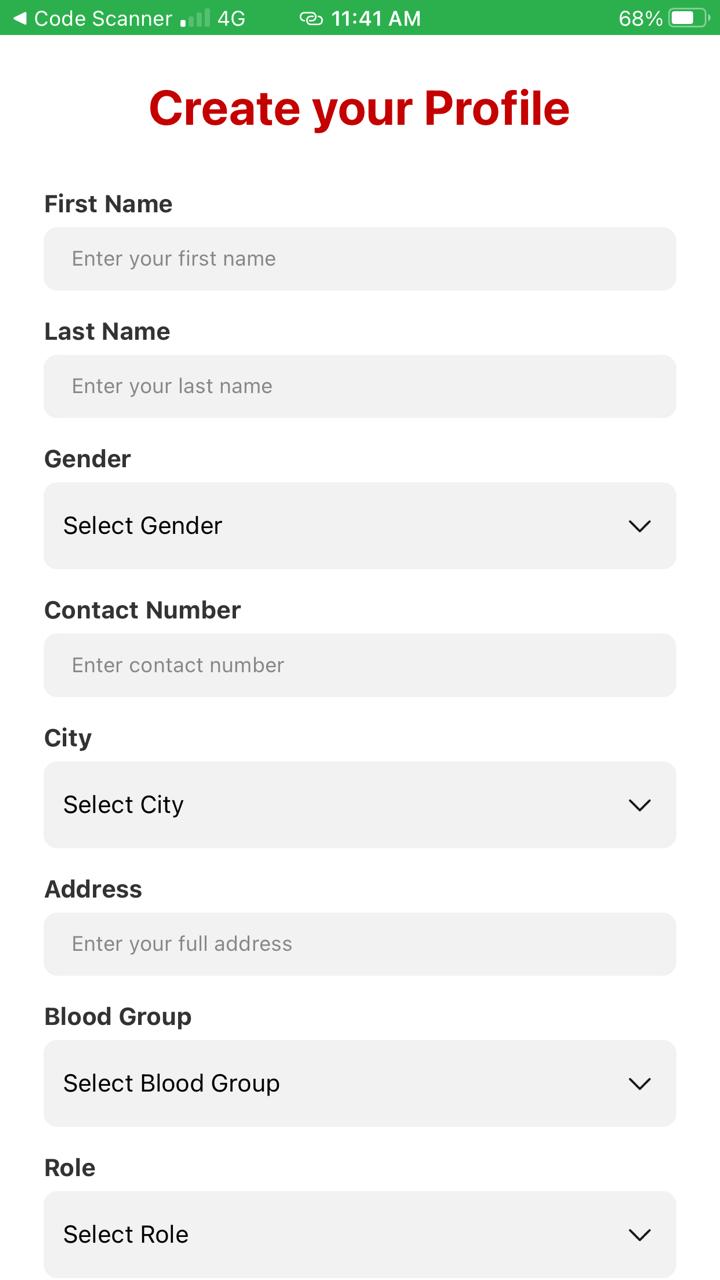
## OTP screen



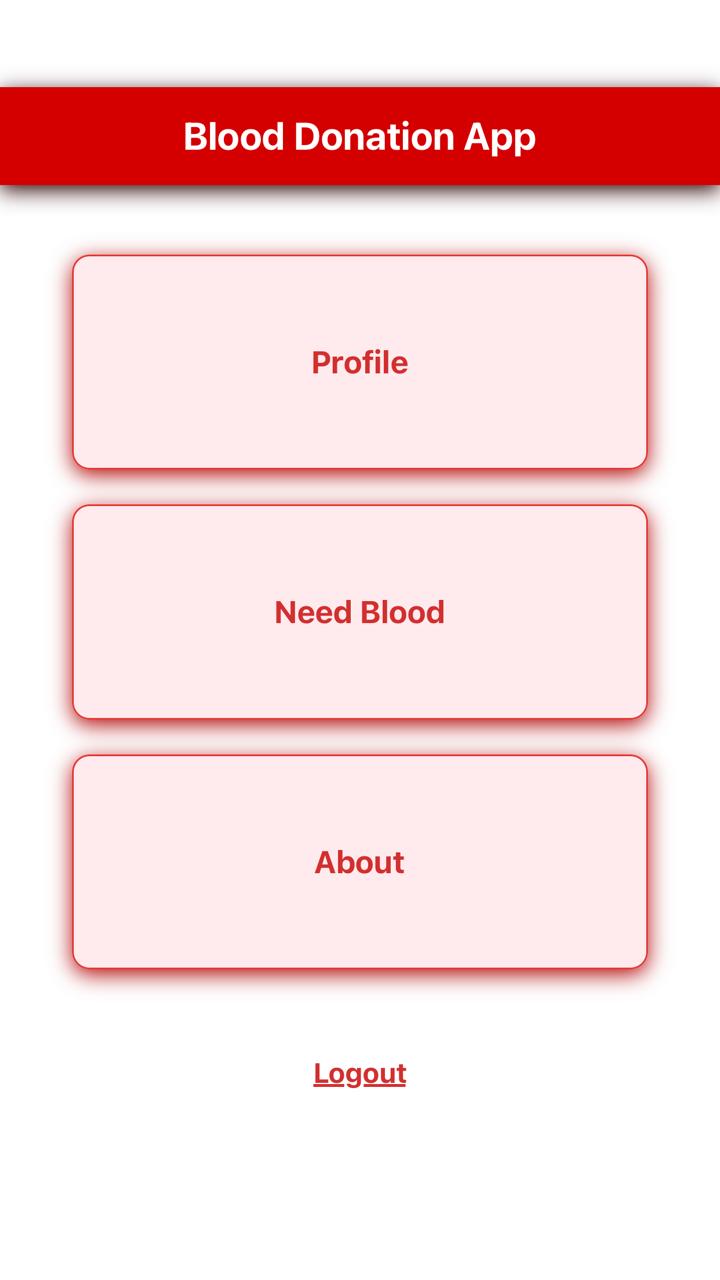
## Forget password screen



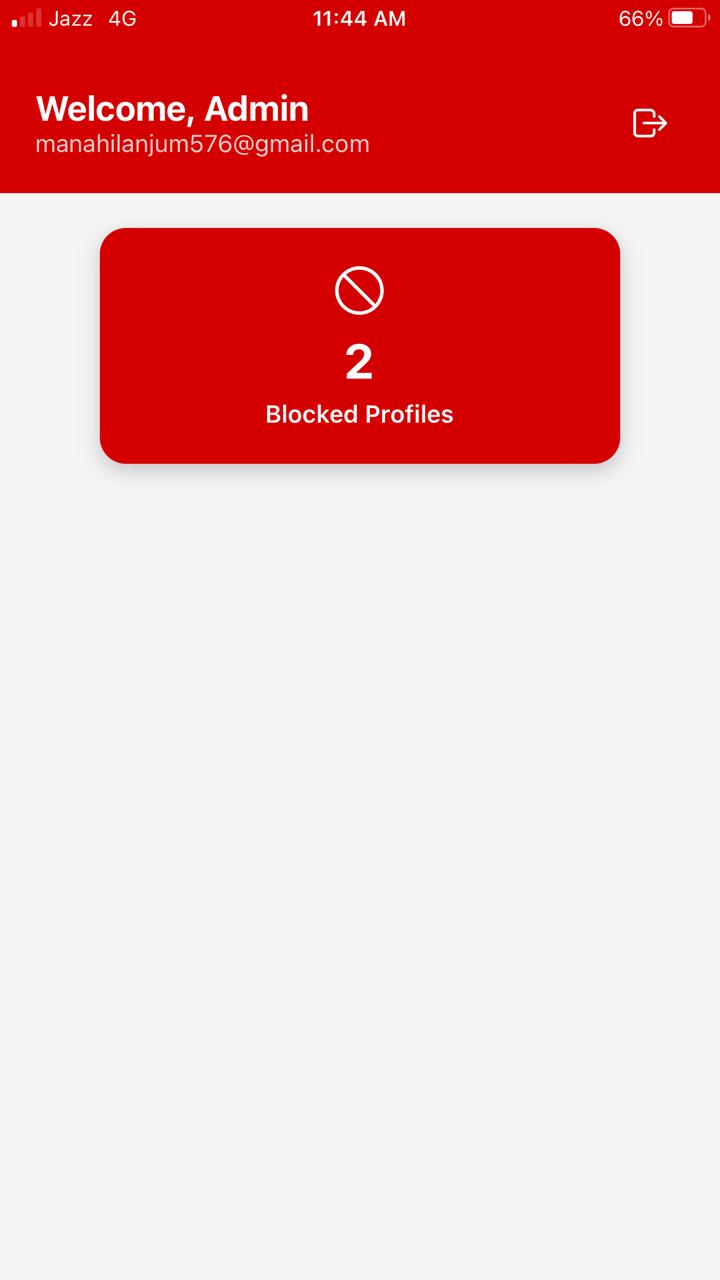
## Create profile screen



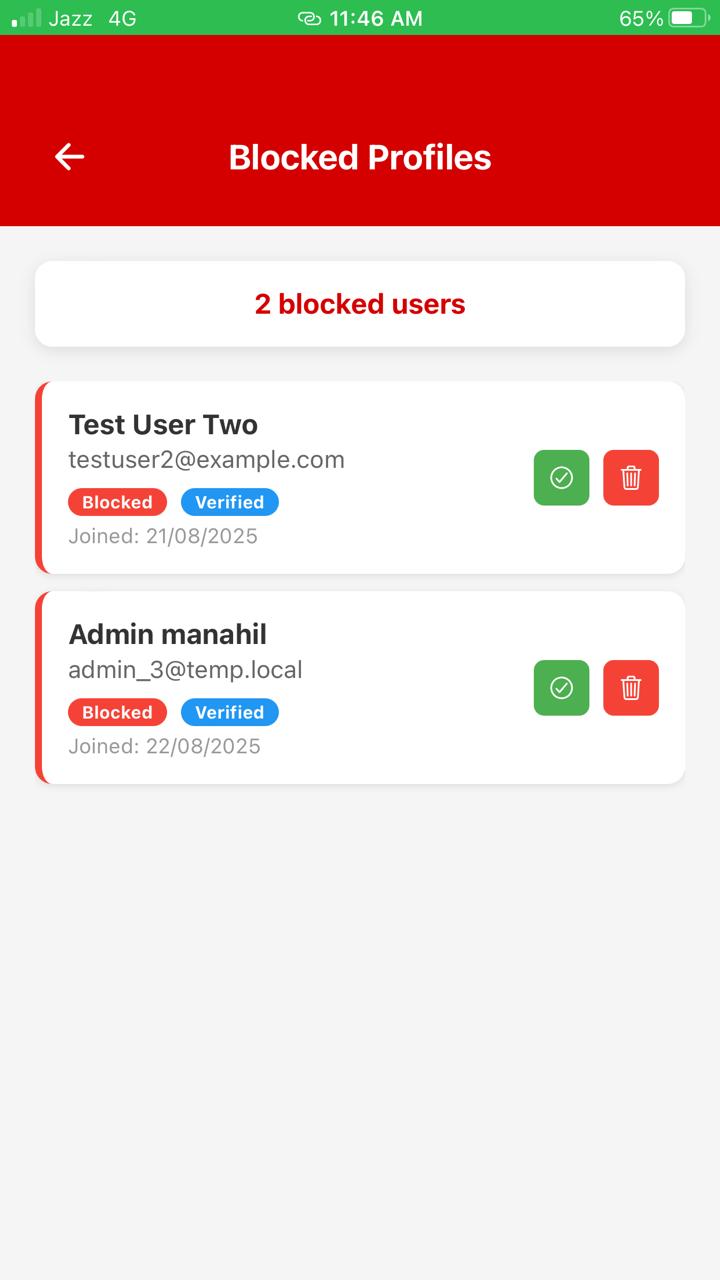
## User Dashboard screen



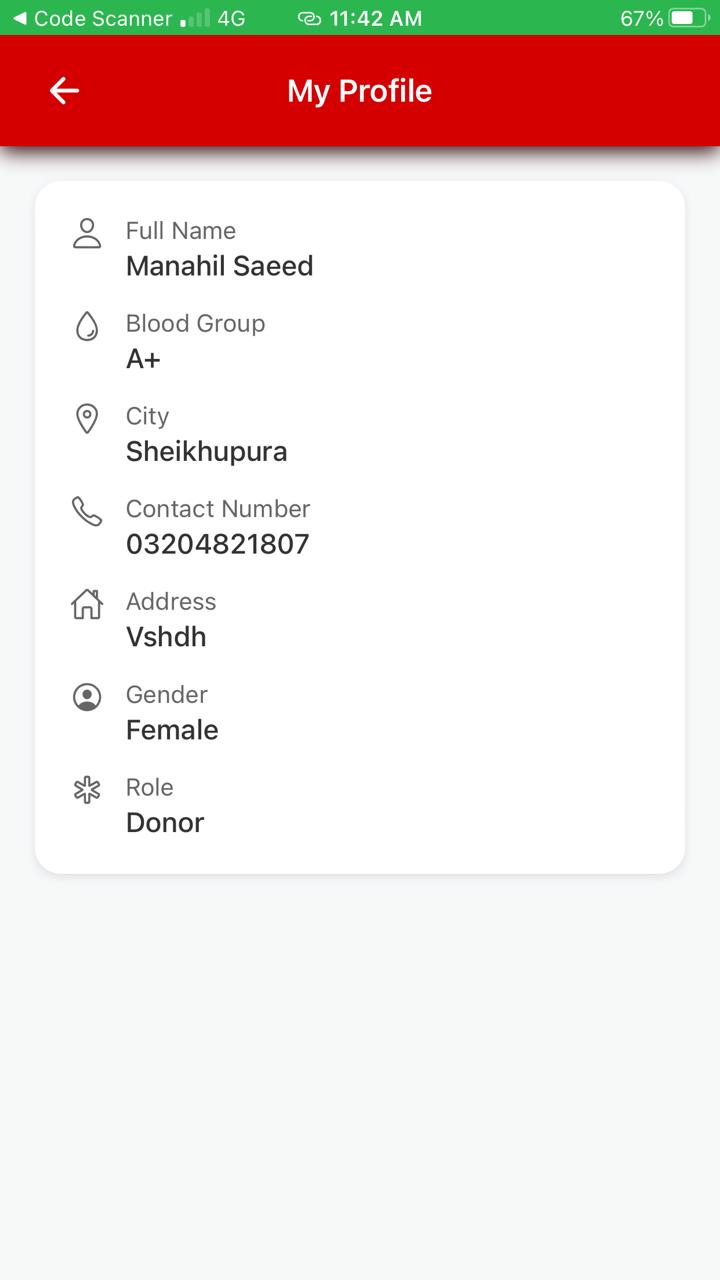
## Admin Dashboard screen



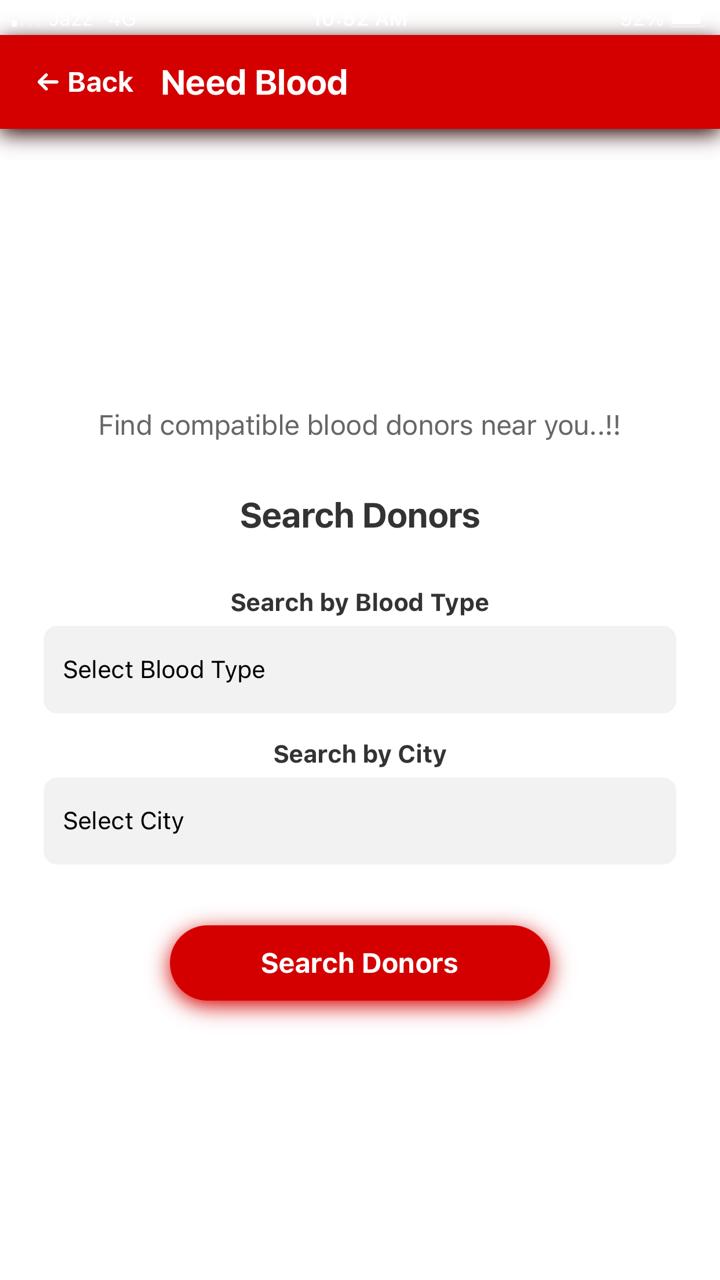
## Block profile list



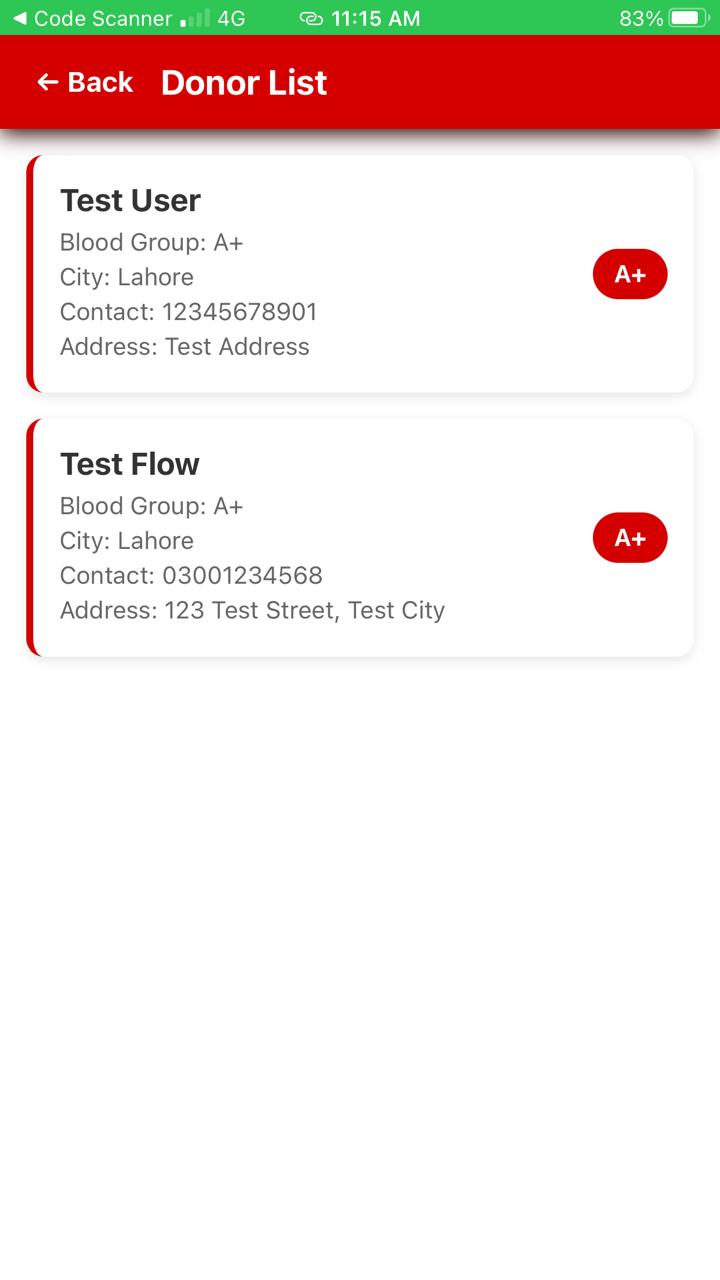
## User profile screen



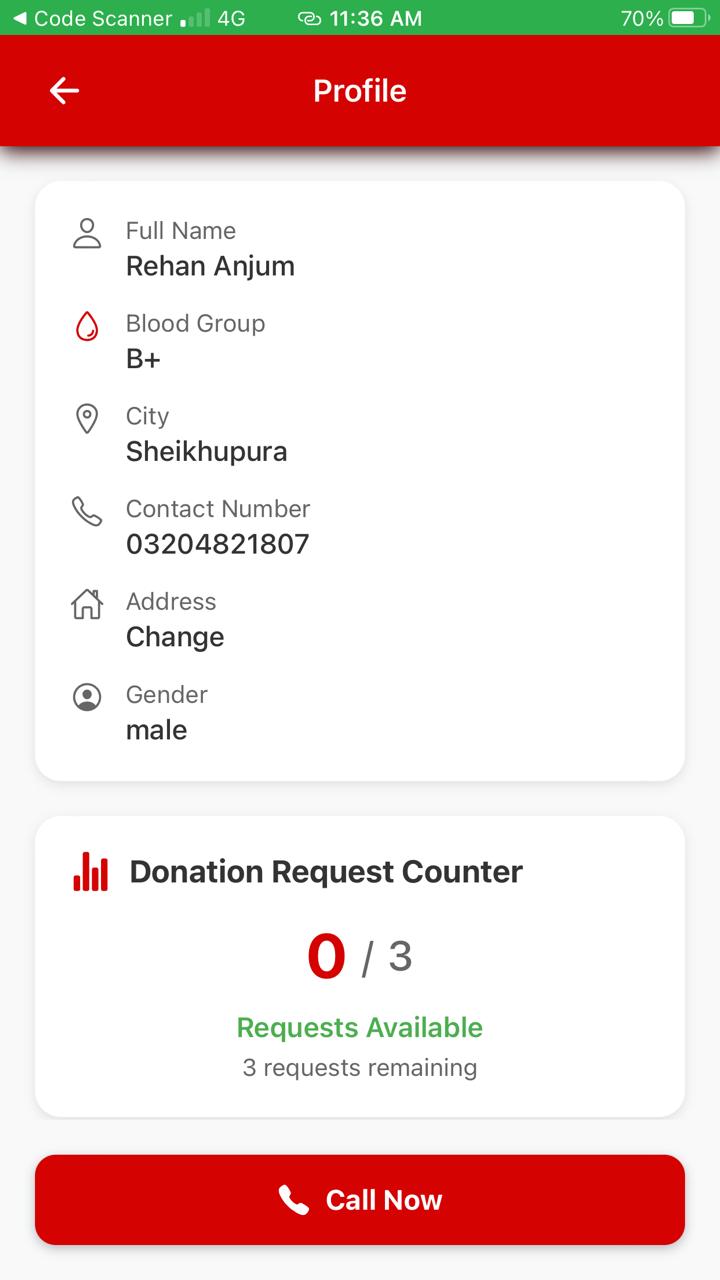
## Search donor screen



## Donor list screen



## Donor profile screen



# TRACEABILITY MATRIX

|  |  |  |  |
| --- | --- | --- | --- |
| **Sr. #** | **System Specification Text** | **Use Case Name** | **Category** |
| 1 | A user “shall” register to the system. | UC\_signin | Online |
| 2 | The system “shall” provide the registration process to users. | UC\_login | Online |
| 3 | The user and admin “shall” login to the system after successful registration. | UC\_Login | Online |
| 4 | System “shall” verify the login credentials of user and admin and on successful login system “shall” provide the user and admin access to his/her account. | UC\_Login | Online |
| 5 | User “shall” be able to search for donor in the city according to blood group. | UC\_Need blood | Online |
| 6 | System “shall” be able to make call to that donor | UC\_call info about detect disease | Offline |

CONCLUSION

The **Blood Donation App** provides an easy and reliable way to connect donors and recipients in emergencies. It reduces delays by allowing quick donor searches, secure profile management, and direct communication. The app is simple to use, technically feasible, and tested to work effectively. Overall, this project shows how technology can play an important role in saving lives by making the blood donation process faster and more efficient.

FUTURE WORK

In the future, the Blood Donation App can be enhanced with features such as real-time location tracking, push notifications for urgent requests, and integration with hospitals and blood banks to ensure better coordination. Artificial intelligence can be used to suggest the most suitable donors based on location, blood type, and availability. Multi-language support and an emergency SOS system can make the app more accessible and effective in critical situations. Additionally, health monitoring tools and nationwide scalability will further improve the app’s impact, making it a comprehensive solution for blood donation management.

REFERENCE

[1] Django Software Foundation. (2024). *Django Documentation*.

[2] Facebook Inc. (2024). *React Native Documentation*.

[3] Oracle Corporation. (2024). *MySQL Documentation*.

[4] Hendra wan, E., Meisel, M., & Sari, D. N. (2023). Analysis and Implementation of Computer Network Systems using Software Draw.io. *Asia Information System Journal*, 2(1).

[5] Erickson, J., Lyytinen, K., & Siau, K. (2005). Agile modeling, agile software development, and extreme programming: the state of research. *Journal of Database Management (JDM)*, 16(4), 88-100.

[6] Srivastava, A., Bhardwaj, S., & Saraswat, S. (2017, May). SCRUM model for agile methodology. In *2017 International Conference on Computing, Communication and Automation (ICCCA)* (pp. 864-869). IEEE.

[7] World Health Organization. (2023). *Blood safety and availability*.

[8] Nurre, S. G., & Weir, J. D. (2017). Interactive Excel-based Gantt chart schedule builder. *INFORMS Transactions on Education*, 17(2), 49-57.