

Darshan University

A Project Report on

**“Blood Bank Management System”**

Under the subject

**Software Engineering (2301CS405)**

B. Tech, Semester – IV

Computer Science & Engineering Department

|  |  |
| --- | --- |
| Submitted By | |
| Student Name: Odedra Viraj Arjanbhai | Enrollment No.:23010101178 |
| Academic Year  (2024-2025) | |
| Internal Guide  Prof. R. B. Gondaliya  Darshan University | Dean-DIET  Dr. Gopi Sanghani  Darshan University |

|  |  |
| --- | --- |
|  | **Computer Science & Engineering Department**  **Darshan University** |

**DECLARATION**

We hereby declare that the SRS, submitted along with the **Software Engineering** **(2301CS405)** for entitled **“Blood Bank Management System”** submitted in partial fulfilment for the Semester-5 of **Bachelor Technology (B. Tech)** in **Computer Science and Engineering (CSE)** Departmentto Darshan University, Rajkot, is a record of the work carried out at **Darshan University, Rajkot** under the supervision of R. B. Gondaliya and that no part of any of report has been directly copied from any students’ reports, without providing due reference.

(Odedra Viraj Arjanbhai)

Student’s Signature

Date: 01/ 03/ 2025

|  |  |
| --- | --- |
|  | **Computer Science & Engineering Department**  **Darshan University** |

**CERTIFICATE**

This is to certify that the SRS on **“Blood Bank Management System” has** been satisfactorily prepared by **Odedra Viraj Arjanbhai**(23010101178) under my guidance in the fulfillment of the course **Software Engineering (2301CS405)** work during the academic year 2024-2025.

|  |  |  |
| --- | --- | --- |
| Internal Guide  Prof. R. B. Gondaliya  Darshan University |  | Dean-DIET  Dr. Gopi Sanghani  Darshan University |

**Acknowledgement**

I wish to express my sincere gratitude to my project guide Prof. R. B. Gondaliyaand all the faculty members for helping me through my project by giving me the necessary suggestions and advices along with their valuable co- ordination in completing this work.

I also thank my parents, friends and all the members of the family for their precious support and encouragement which they had provided in completion of my work. In addition to that, I would also like to mention the Darshan University personals who gave me the permission to use and experience the valuable resources required for the project from the University premises.

Thus, in conclusion to the above said, I once again thank the faculties and members of **Darshan University** for their valuable support in completion of the project.

Thanking You

**Odedra Viraj Arjanbhai**

**ABSTRACT**

Blood Bank Management System is a system which aims in developing a computerized system to maintain all the daily work of a blood bank. This system will act as a tool to transfer the traditional blood bank operations into a digital platform. This project has many features which are generally not available in general blood bank systems, such as the facility of user login, blood search, and donation tracking with a single click. It also has a facility of admin login through which the admin can monitor the whole system. This system will be designed with the basic features such as the admin can add/view/update/delete blood donor and recipient details. It also provides a feature where donors, after logging into their accounts, can see a list of their past donations, donation dates, and certificates. Recipients, after logging into their accounts, can request blood, track the status of their requests, and view available blood types. The admin, after logging into their account, can generate various reports such as donation reports, recipient requests, and blood stock reports.

Main purpose of this system is to reduce human efforts as much as possible.

**Table of Contents**

[List of Figures I](#_Toc191816326)

[List of Tables II](#_Toc191816327)

[1 Introduction 1](#_Toc191816328)

[1.1 Product perspective 1](#_Toc191816329)

[1.2 Product features 1](#_Toc191816330)

[1.2.1 There are four different users who will be using this product: 1](#_Toc191816331)

[1.2.2 The features required for the Admin are: 1](#_Toc191816332)

[1.2.3 The features that are required for Donor are: 1](#_Toc191816333)

[1.2.4 The features required for the Recipient are: 1](#_Toc191816334)

[1.2.5 The features required for the Hospital are: 1](#_Toc191816335)

[1.3 Functional Requirement 2](#_Toc191816336)

[1.3.1 Blood Donor 2](#_Toc191816337)

[1.3.2 Recipient 2](#_Toc191816338)

[1.3.3 Hospital 2](#_Toc191816339)

[1.3.4 Admin 2](#_Toc191816340)

[1.4 Non-Functional Requirement 3](#_Toc191816341)

[1.4.1 Usability: 3](#_Toc191816342)

[1.4.2 Accuracy: 3](#_Toc191816343)

[1.4.3 Availability: 3](#_Toc191816344)

[1.4.4 Maintainability: 3](#_Toc191816345)

[2 Design and Implementation Constraints 4](#_Toc191816346)

[2.1 Use case diagram 4](#_Toc191816347)

[2.2 Activity diagram and Swimlane diagram 5](#_Toc191816348)

[2.3 Sequence diagram 7](#_Toc191816349)

[2.4 State diagram 8](#_Toc191816350)

[2.5 Class diagram 9](#_Toc191816351)

[2.6 Data flow diagram 10](#_Toc191816352)

[2.6.1 Context diagram (level-0) 10](#_Toc191816353)

[2.6.2 DFD Level-1 11](#_Toc191816354)

[2.6.3 DFD Level-2 12](#_Toc191816355)

[3 External interface requirement (Screens) 13](#_Toc191816356)

[3.1 Screen-1: Blood Stock Availability 13](#_Toc191816357)

[3.2 Screen-2: Donor Feedback Form 14](#_Toc191816358)

[3.3 Screen-3: Blood Storage Unit 15](#_Toc191816359)

[3.4 Screen-4: Blood Camp Schedule 16](#_Toc191816360)

[3.5 Screen-5: Camp Registration 18](#_Toc191816361)

[3.6 Screen – 6: Donor Feedback 20](#_Toc191816362)

[3.7 Screen – 7: Blood Request Tracking 21](#_Toc191816363)

[3.8 Screen – 8: Hospital Complaints 24](#_Toc191816364)

[3.9 Screen – 9: Campaign Report Generation 26](#_Toc191816365)

[4 Database design 28](#_Toc191816366)

[4.1 List of Tables 28](#_Toc191816367)

[5 Stories and Scenario 30](#_Toc191816368)

[5.1 Story-1: Request Blood by Recipient 30](#_Toc191816369)

[5.1.1 Scenario# S1.1 30](#_Toc191816370)

[5.1.2 Scenario# S1.2 30](#_Toc191816371)

[5.1.3 Scenario# S1.3 30](#_Toc191816372)

[5.1.4 Scenario# S1.4 30](#_Toc191816373)

[5.1.5 Scenario# S1.5 31](#_Toc191816374)

[5.2 Story-2: Add New Blood Donor to the Blood Bank Database 31](#_Toc191816375)

[5.2.1 Scenario# S2.1 31](#_Toc191816376)

[5.2.2 Scenario# S2.2 32](#_Toc191816377)

[5.2.3 Scenario# S2.3 32](#_Toc191816378)

[5.3 Story-3: Update Blood Inventory Levels 32](#_Toc191816379)

[5.3.1 Scenario# S3.1 33](#_Toc191816380)

[5.3.2 Scenario# S3.2 33](#_Toc191816381)

[5.3.3 Scenario# S3.3 33](#_Toc191816382)

[6 Test cases 34](#_Toc191816383)

[7 References 41](#_Toc191816384)

# List of Figures

[Figure 2.1‑1 Use case diagram for Blood Bank management system 4](#_Toc191816042)

[Figure 2.2‑1 Activity Diagram for Blood Donor 5](#_Toc191816043)

[Figure 2.2‑2 Activity Diagram for Blood Recipient 5](#_Toc191816044)

[Figure 2.2‑3 Swimlane Diagram For Blood Recipient 6](#_Toc191816045)

[Figure 2.2‑4 Swimlane Diagram For Blood Donor 6](#_Toc191816046)

[Figure 2.3‑1 Sequence Diagram For Blood Donor 7](#_Toc191816047)

[Figure 2.3‑2Sequence Diagram for Blood Recipient 7](#_Toc191816048)

[Figure 2.4‑1 State Diagram For Blood 8](#_Toc191816049)

[Figure 2.4‑2 State Diagram for Blood Request 8](#_Toc191816050)

[Figure 2.5‑1 Class diagram for Blood Bank Management System 9](#_Toc191816051)

[Figure 2.6‑1 Context diagram for Blood Bank Management System 10](#_Toc191816052)

[Figure 2.6‑2 DFD level-1 for Blood Bank Management System 11](#_Toc191816053)

[Figure 2.6‑3 DFD level-2 for Issue book 12](#_Toc191816054)

[Figure 3.1‑1 Screen-1: Blood Stock Availability 13](#_Toc191816055)

[Figure 3.2‑1 Screen-2: Blood Donor Feedback Form 14](#_Toc191816056)

[Figure 3.3‑1 Screen-3: Blood Storage Unit 15](#_Toc191816057)

[Figure 3.4‑1 Screen-4: Search Donor Availability 16](#_Toc191816058)

[Figure 3.5‑1 screen-5 Blood Camp Registration 18](#_Toc191816059)

[Figure 3.6‑1 Screen - 6 Donor Feedback 20](#_Toc191816060)

[Figure 3.6‑2 Screen Element of Donor Feedback 21](#_Toc191816061)

[Figure 3.7‑1 Screen - 7 Blood Request Tracking 21](#_Toc191816062)

[Figure 3.7‑2 Screen - 7 Blood Request Tracking - 2 22](#_Toc191816063)

[Figure 3.7‑3 Screen Element of Blood Request Tracking - 1 22](#_Toc191816064)

[Figure 3.8‑1 Screen - 8 Hospital Complaints 24](#_Toc191816065)

[Figure 3.8‑2 Screen Element for Hospital Complaints 25](#_Toc191816066)

[Figure 3.9‑1 Screen - 9 Campaign Report Generation 26](#_Toc191816067)

[Figure 3.9‑2 Screen Elements for Campaign Report Generation 27](#_Toc191816068)

# List of Tables

[Table 3.1‑1 Screen element of Blood Stock Availability 13](#_Toc191816069)

[Table 3.2‑1 Screen element of Blood Donor Feedback Form 14](#_Toc191816070)

[Table 3.3‑1 Screen element of Blood Storage Unit 16](#_Toc191816071)

[Table 4.1‑1 Table: Donor 28](#_Toc191816072)

[Table 4.1‑2 Table: Recipient 28](#_Toc191816073)

[Table 4.1‑3 Table: Doctor 29](#_Toc191816074)

[Table 4.1‑4 Table: Blood Bank 29](#_Toc191816075)

# Introduction

## Product perspective

This project aims to update the traditional blood bank management system into an internet-based application, allowing users to easily access information about blood availability, donation status, and their account details. It is a multi-user system that efficiently handles essential functions of a blood bank, including blood donation, inventory management, recipient requests, and emergency handling. The system is designed to support all the basic operations of a blood bank, ensuring smooth management of blood donations, tracking blood stock, and improving communication between donors, recipients, and hospitals. It can effectively cater to the needs of a small to medium-sized blood bank.

## Product features

### There are four different users who will be using this product:

* Admin (Blood Bank Manager): Oversees all operations of the blood bank.
* Donor: Donates blood to the bank.
* Recipient: Requests blood based on medical needs.
* Hospital: Receives blood for patients who need it.

### The features required for the Admin are:

* Generate Reports: Create reports on blood donations, requests, and inventory status.
* Track Donor and Recipient Data: View history of donors and track blood requests from recipients.
* Approve Donors: Check if donors have the health requirements and approve them.
* Blood Requests: Monitor and restock blood requests from hospitals or recipients.
* Emergency Blood Requests: Handle urgent blood requests during emergencies.
* Issue Blood Units: Give blood to hospitals or recipients as needed.

### The features that are required for Donor are:

* Donate Blood: Donate blood if eligible and available at donation camps or blood banks.
* View Donation History: See a list of past donations and details of each donation.
* Create an Account: Create and check a profile to track donations and receive certificates.
* Request Donation Certificates: Get certificates as proof of blood donation.
* Track Donation Camps: View upcoming blood donation drives or events.

### The features required for the Recipient are:

* Request Blood: Request blood if needed for medical treatment.
* View Request History: See the list of past blood requests and their statuses.
* Create an Account: Create a profile for blood requests and track status.
* Receive Notifications: Get updates on blood availability and request status.
* Search Available Blood: Find available blood types in nearby blood banks.

### The features required for the Hospital are:

* Request Blood: Hospitals can request specific blood types for their patients.
* View Blood Usage History: See a list of blood units used by patients in the hospital.
* Check Blood Stock: Track and Check the blood received from the blood bank.

## Functional Requirement

### Blood Donor

* Registration: For registration donor needs to create an account that contains personal details.
* Eligibility: Verifies if the donor meets health and age criteria for donation.
* Donation History: Displays past donations, including dates and donation centers.
* Certificates: Provides downloadable certificates as proof of donation.
* Appointment: Enables donors to schedule or reschedule donation appointments.
* Update Profile: Allows updating personal information, contact details, and preferences.
* Event Donation: Facilitates participation in special donation drives or events.
* Feedback: Collects donor feedback to improve donation processes.
* Donation Analytics: Provides insights on the donor’s impact, such as the number of lives saved.

### Recipient

* Recipient Registration: Enables individuals to register as recipients by providing their personal and medical details.
* Search Blood Center: Allows recipients to locate nearby blood centers based on their location or specific blood type availability.
* Request Blood: Facilitates the submission of blood requests, including urgent requirements for specific blood groups.
* Check Status: Provides updates on blood request approvals, availability, and estimated delivery times.
* Check Donor History: Displays information about donors who have contributed.
* Give Feedback: Allows recipients to share their experiences with the blood center or donation process to enhance services.
* Track Delivery: Enables real-time tracking of blood units in transit to the recipient's location.
* Health Tips: Provides guidelines for recipients on post-transfusion care and maintaining health.

### Hospital

* Schedule Appointments: Schedule donor appointments for hospital blood.
* Verify Donor Eligibility: Check donor health details to ensure safe blood donations at the hospital.
* View Blood Storage: Enables hospitals to monitor available blood units by type and quantity in their storage.
* Check Feedback: Allows hospitals to review feedback from donors and recipients to improve services.
* Request Blood Stock: Submit requests for restock blood supplies based on current inventory levels.
* Resolve Complains: Hospitals can respond to complaints or queries from donor or recipients.

### Admin

* Access Privileges: Admins can control and assign access levels to different users (donors, recipients, hospitals) based on their roles.
* Check User Accounts: Admin can add, remove, or update user profiles to maintain a smooth workflow in the system.
* Respond to Complaints: Admin handles and resolves complaints or issues raised by users related to blood donation or blood bank services.
* Monitor Requests: Admin can track all blood requests made by recipients or hospitals and ensure timely they gate their needs.
* Generate Reports: Admin generates detailed reports on blood donations, stock levels, usage trends, and overall system performance.

## Non-Functional Requirement

### Usability:

* The UI should be simple enough for everyone to understand and get the relevant information without any special training. Different languages can be provided based on the requirements.

### Accuracy:

* The data stored about the books and the fines calculated should be correct, consistent, and reliable.

### Availability:

* The System should be available for the duration when the library operates and must be recovered within an hour or less if it fails. The system should respond to the requests within two seconds or less.

### Maintainability:

* The software should be easily maintainable and adding new features and making changes to the software must be as simple as possible. In addition to this, the software must also be portable.

# Design and Implementation Constraints

## Use case diagram

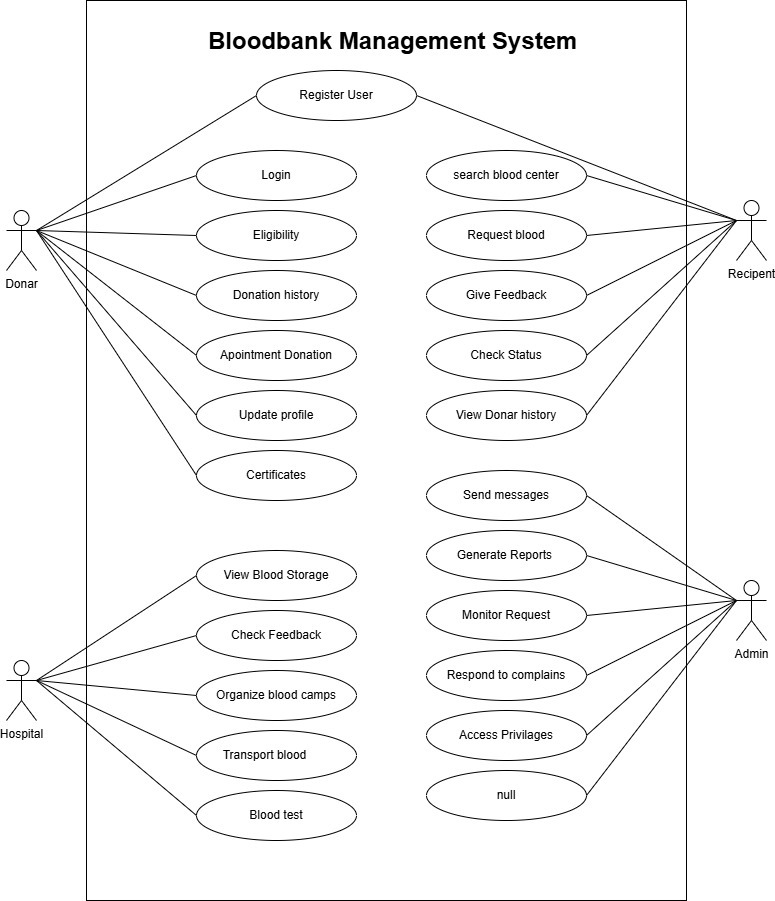


Figure ‑ Use case diagram for Blood Bank management system

## Activity diagram and Swimlane diagram

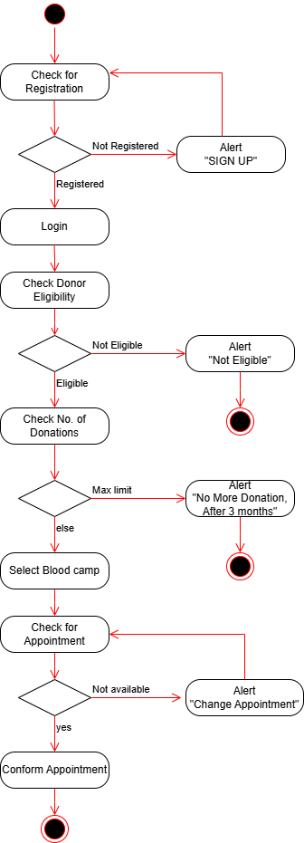


Figure ‑ Activity Diagram for Blood Donor

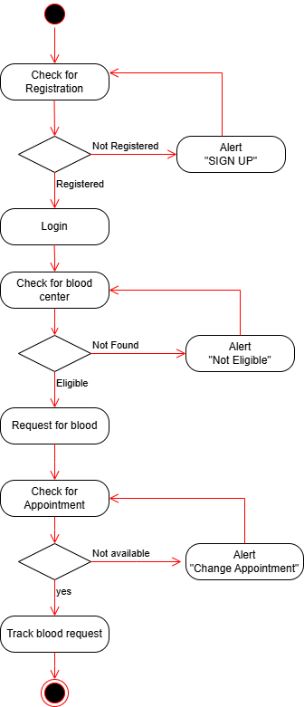


Figure ‑ Activity Diagram for Blood Recipient

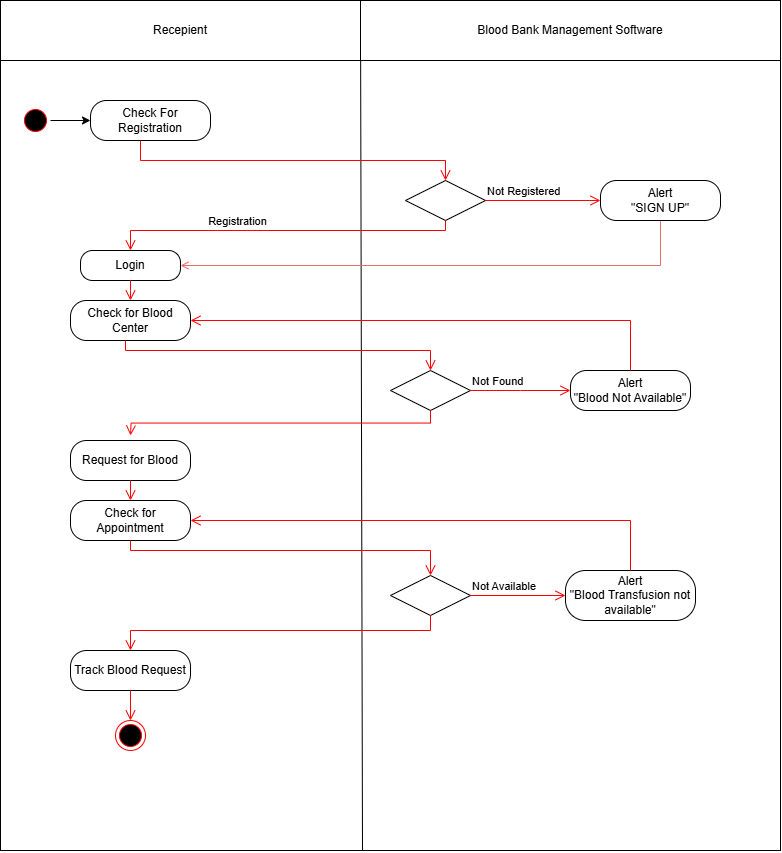


Figure ‑ Swimlane Diagram For Blood Recipient

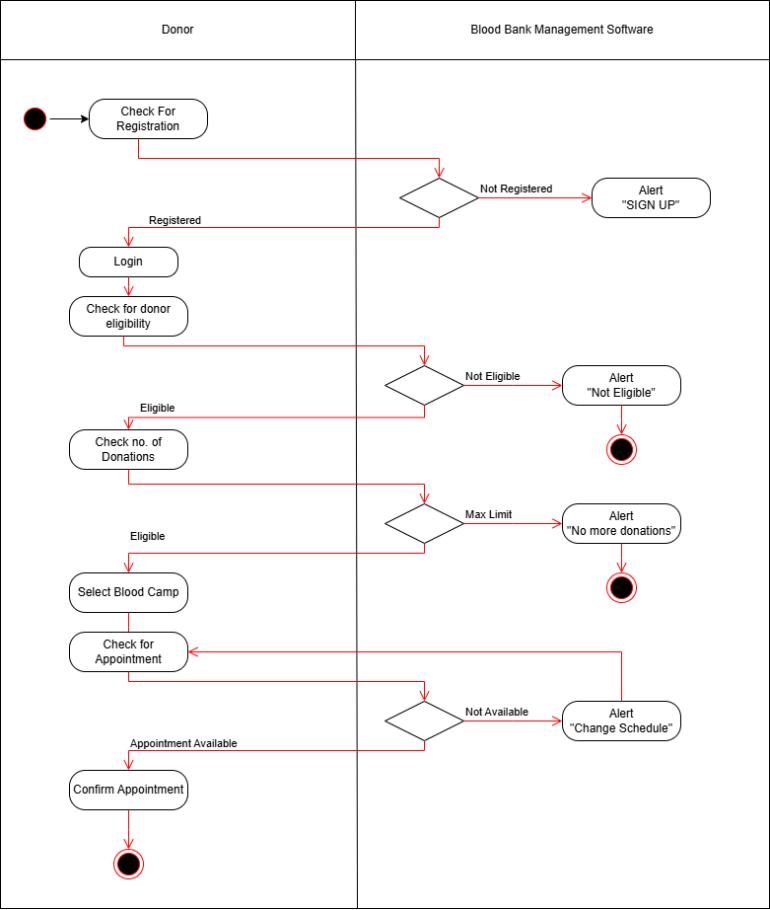


Figure ‑ Swimlane Diagram For Blood Donor

## Sequence diagram

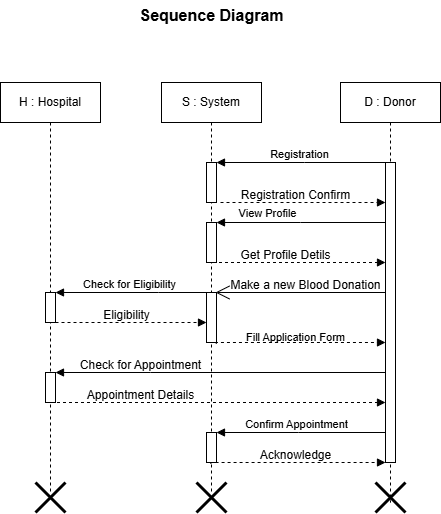


Figure ‑ Sequence Diagram For Blood Donor

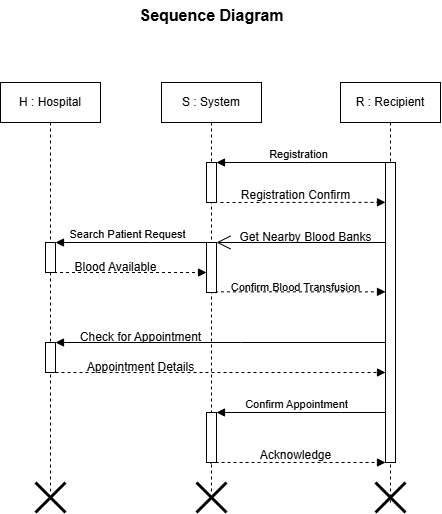


Figure ‑Sequence Diagram for Blood Recipient

## State diagram

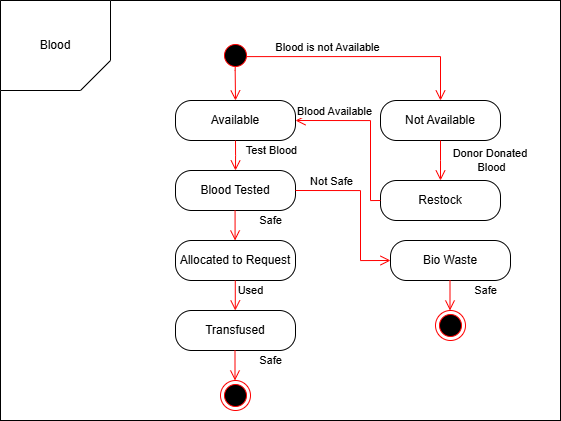


Figure ‑ State Diagram For Blood

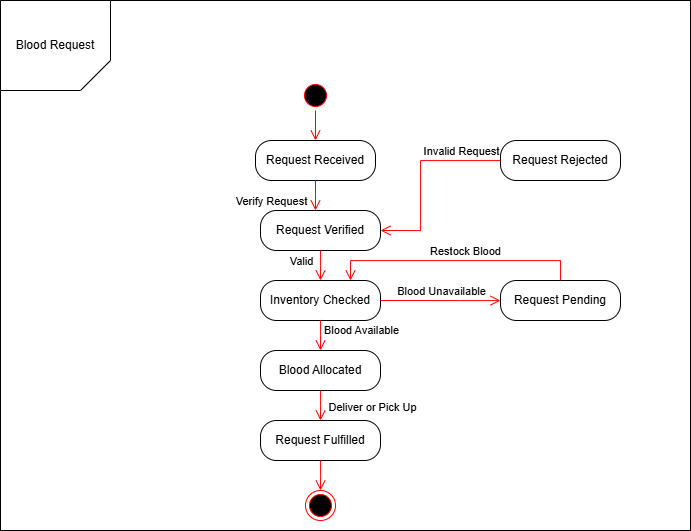


Figure ‑ State Diagram for Blood Request

## Class diagram

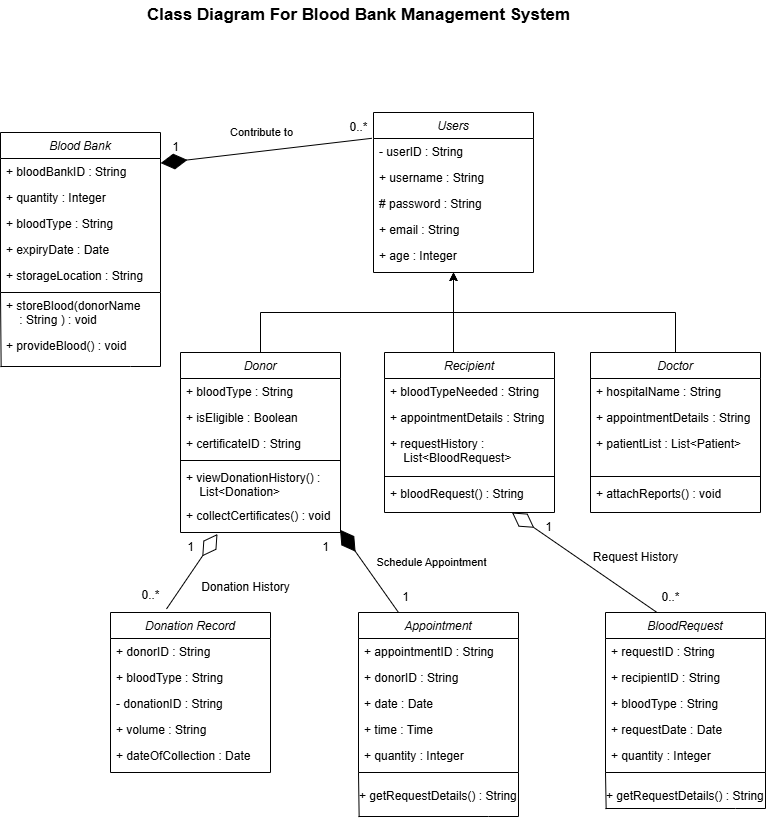


Figure ‑ Class diagram for Blood Bank Management System

## Data flow diagram

### Context diagram (level-0)

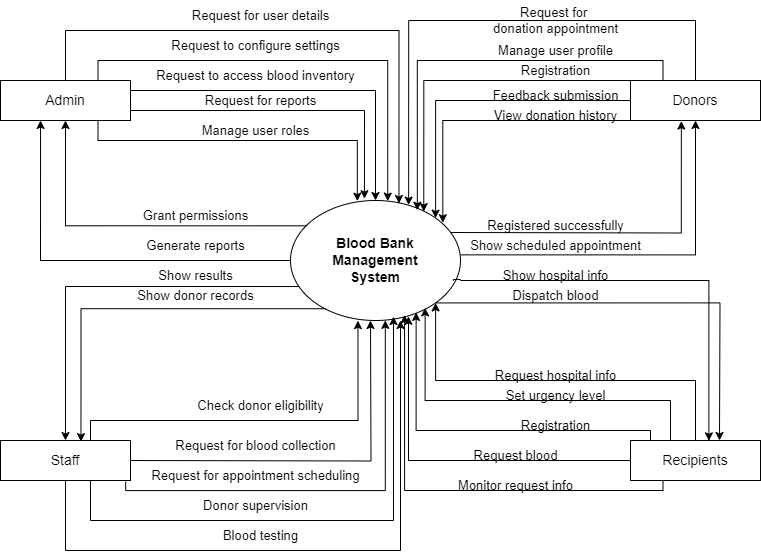


Figure ‑ Context diagram for Blood Bank Management System

### DFD Level-1

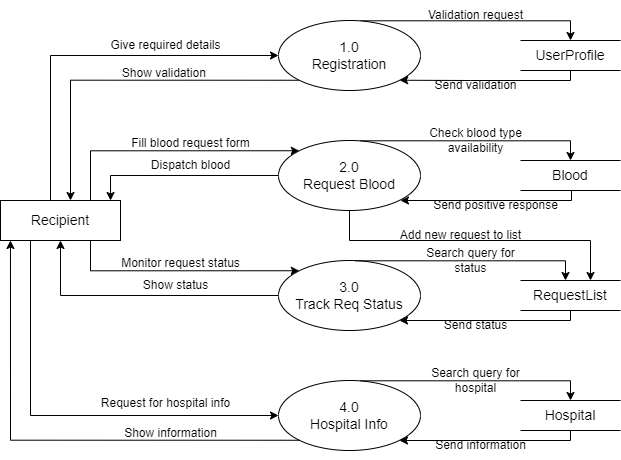


Figure ‑ DFD level-1 for Blood Bank Management System

# External interface requirement (Screens)

## Screen-1: Blood Stock Availability

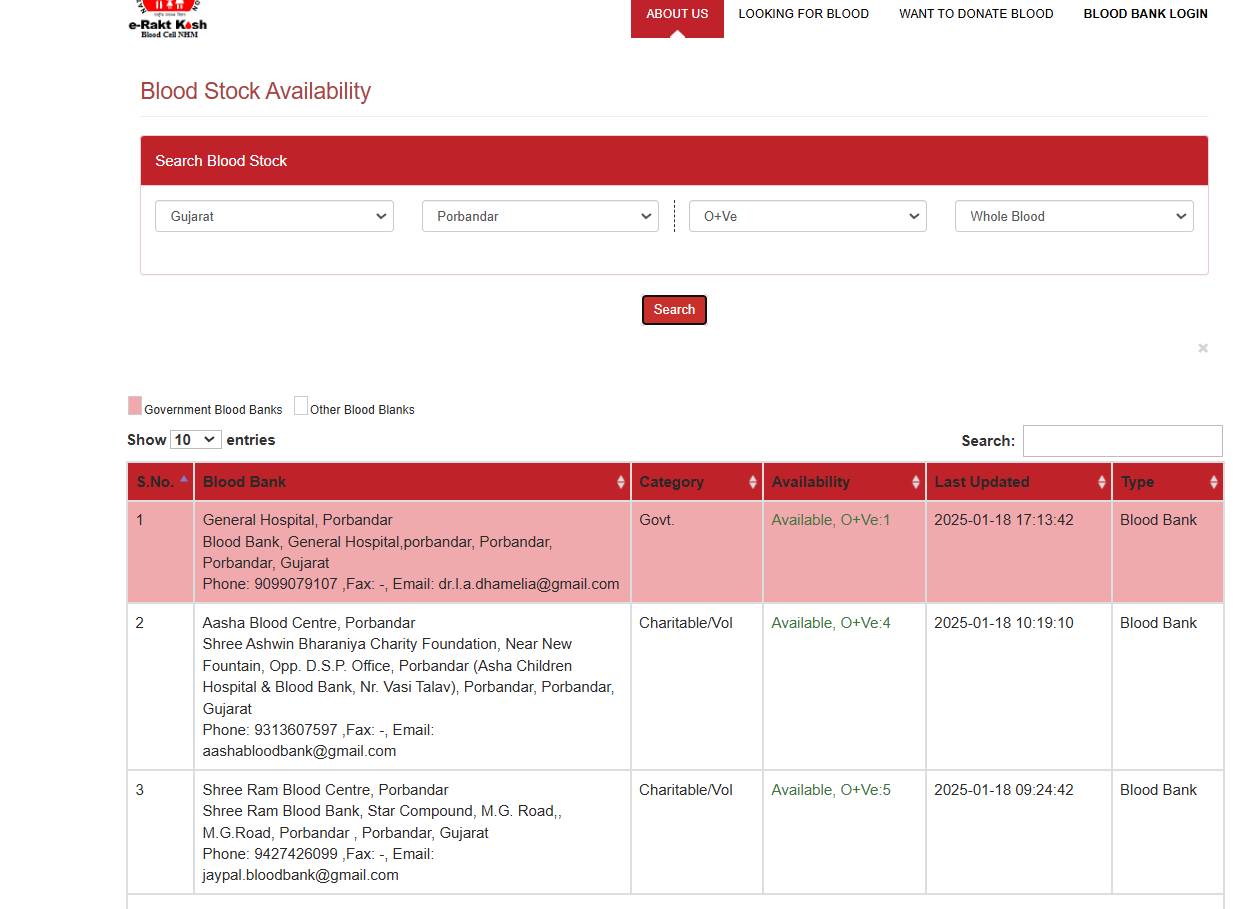


Figure ‑ Screen-1: Blood Stock Availability

**Purpose:** T This form will allow the target end-users to check the availability of blood type in the system. To check the availability, the following information will be encoded in the system.

Table ‑ Screen element of Blood Stock Availability

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Sr. | Screen Element | Input Type | O/M | 1/N | Description |
| 1 | State | Select | M | 1 | Select state for a blood bank in which you are looking for blood stock availability. |
| 2 | City | Select | M | 1 | Select city for a blood bank in which you are looking for blood stock availability. |
| 3 | Blood Group | Select | M | 1 | Select blood group for which you are looking for stock availability. |
| 4 | Blood  Component | Select | M | 1 | Select blood component for which you are  looking for stock availability. |
| 5 | Search | Button | M | 1 | Click on the search button to view the desired search result. |
| 6 | Search Result | Table | M | 1 | Search result table indicates blood bank details from desired inputs. |
| 7 | Title | Text | M | 1 | A title describes what a given screen or interface is about. |

## Screen-2: Donor Feedback Form

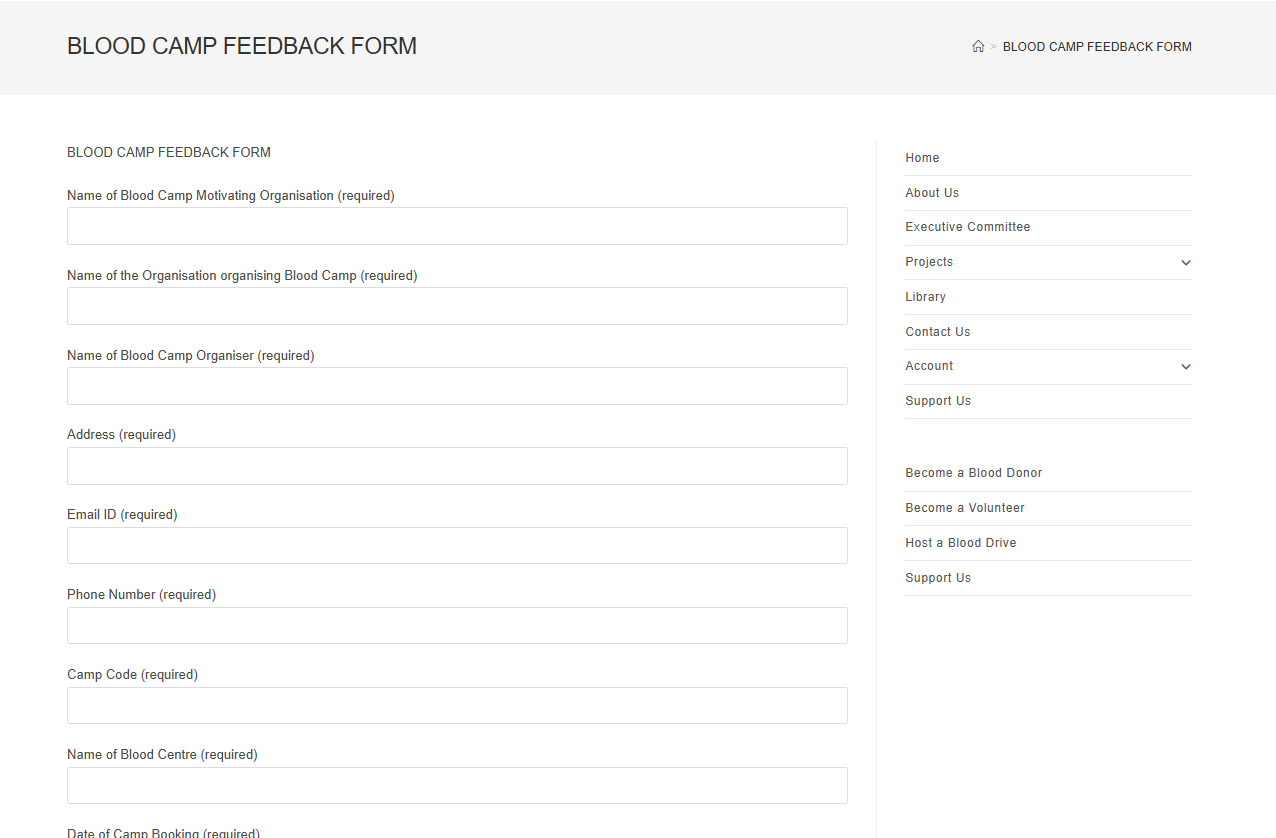


Figure ‑ Screen-2: Blood Donor Feedback Form

**Purpose:** This screen serves as a Blood Camp Feedback Form that enables users to provide feedback regarding blood camps, specifically for activities such as blood donation camps. It also collects essential information about organizations and individuals responsible for organizing these camps to improve service delivery and ensure better management.

Table ‑ Screen element of Blood Donor Feedback Form

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Sr. | Screen Element | Input Type | O/M | 1/N | Description |
| 1 | Name of Blood Camp Motivating Organisation | Text Field | M | 1 | Allows users to input the name of the motivating organization for the blood camp. |
| 2 | Name of the Organisation organising Blood Camp | Text Field | M | 1 | Allows users to input the name of the organization responsible for organizing the camp. |
| 3 | Name of Blood Camp Organiser | Text Field | M | 1 | Allows users to input the name of the individual organizing the camp. |
| 4 | Address | Text Field | M | 1 | Allows users to input the address related to the camp. |
| 5 | Email ID | Text Field | M | 1 | Allows users to input the email ID for communication purposes. |
| 6 | Phone Number | Text Field | M | 1 | Allows users to input the phone number for contact purposes. |
| 7 | Camp Code | Text Field | M | 1 | Allows users to input a unique code identifying the camp. |
| 8 | Name of Blood Centre | Text Field | M | 1 | Allows users to input the name of the blood center associated with the camp. |
| 9 | Date of Camp Booking | Date Picker/Calendar | M | 1 | Allows users to select the date on which the camp booking is made. |

## Screen-3: Blood Storage Unit

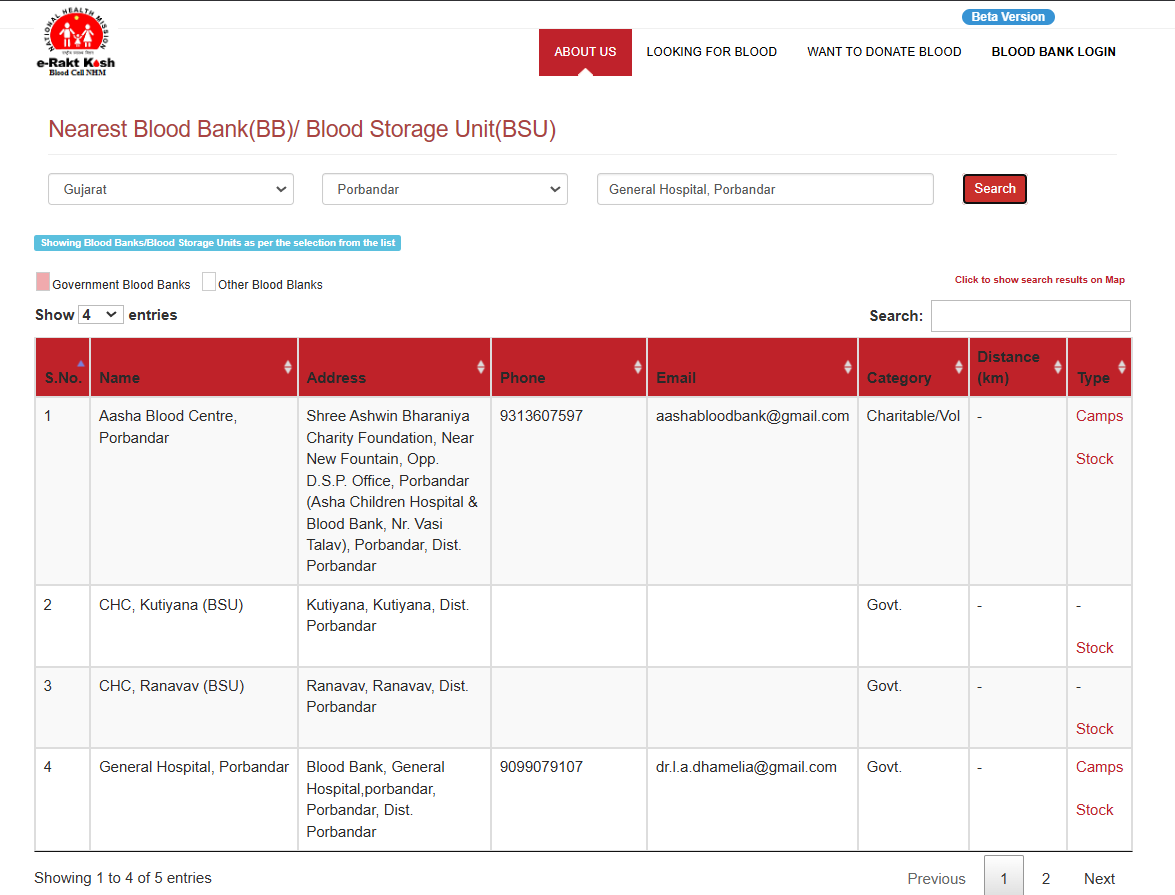


Figure ‑ Screen-3: Blood Storage Unit

**Purpose:** This form will allow the target end-users to view blood storage unit of nearest blood banks in the system. To view the storage units, the following information will be encoded in the system.

Table ‑ Screen element of Blood Storage Unit

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Sr. | Screen Element | Input Type | O/M | 1/N | Description |
| 1 | State | Select | M | 1 | Select state for a blood bank in which you are looking for blood stock availability. |
| 2 | City | Select | M | 1 | Select city for a blood bank in which you are looking for blood stock availability. |
| 3 | Blood Bank | Text Input | M | 1 | Write blood bank or hospital name for which you are looking for desired results. |
| 4 | Search | Button | M | 1 | Click on the search button to view the desired search result. |
| 5 | Search Result | Table | M | 1 | Search result table indicates blood bank details from desired inputs. |
| 6 | Logo | Image | M | 1 | Logo image represents the system. |
| 7 | Title | Text | M | 1 | A title describes what a given screen or interface is about. |

## Screen-4: Blood Camp Schedule

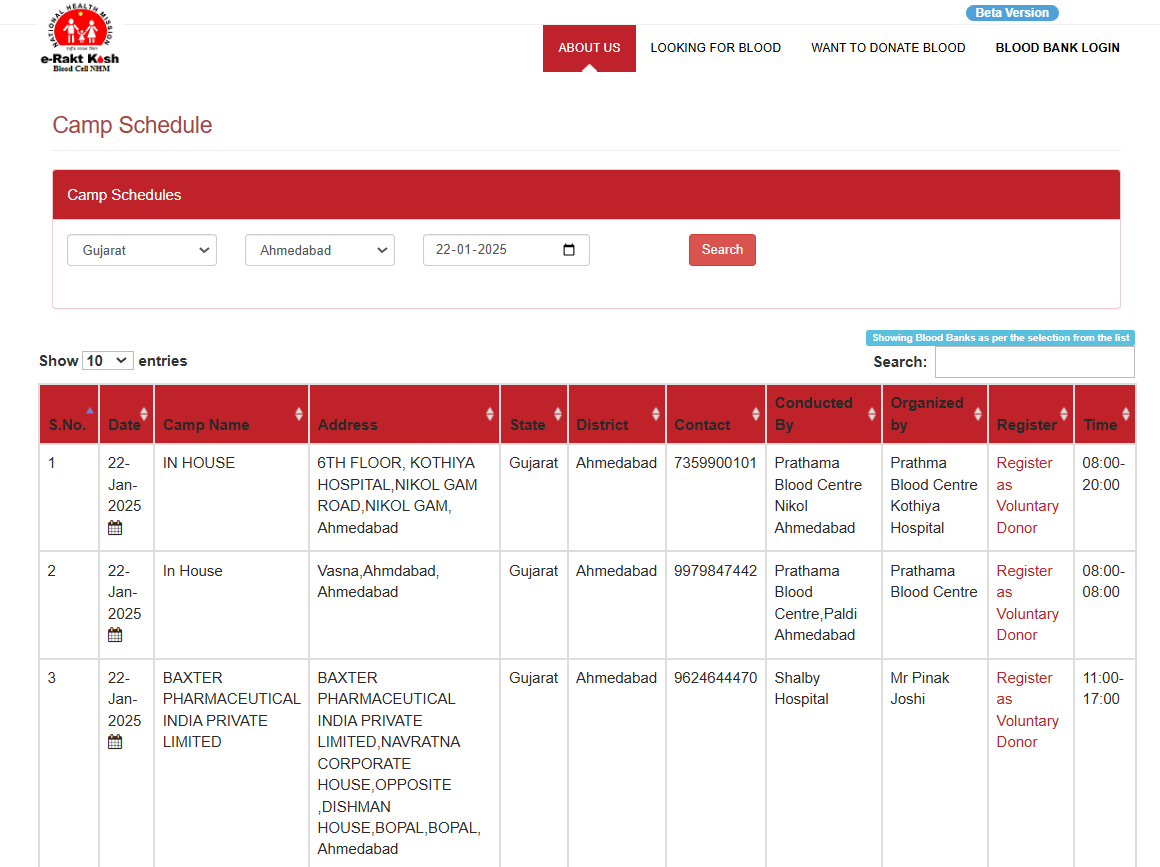


Figure ‑ Screen-4: Search Donor Availability

**Purpose:** This screen will allow the target end-users to view and search for scheduled blood donation camps in the system. To find a suitable camp, users can filter the information by state, district, and date.

Table 3.4-1 Screen element of Blood Camp Schedule

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Sr. | Screen Element | Input Type | O/M | 1/N | Description |
| 1 | S.No. | Text | M | 1 | Serial number of the table row. |
| 2 | Date | Date Picker | M | 1 | Date on which the camp is scheduled. |
| 3 | Camp Name | Text | M | N | Name of the blood donation camp. |
| 4 | Address | Text | M | N | Address where the camp is organized. |
| 5 | State | Dropdown | M | 1 | State where the camp is located. |
| 6 | District | Dropdown | M | 1 | District where the camp is located. |
| 7 | Contact | Number | O | 1 | Contact number of the camp organizer. |
| 8 | Conducted By | Text | M | 1 | Organization conducting the camp. |
| 9 | Organized By | Text | M | 1 | Organization managing the event logistics. |
| 10 | Register | Hyperlink | M | N | Link to register as a voluntary donor. |
| 11 | Time | Time Range | M | 1 | Timings during which the camp will operate. |

## Screen-5: Camp Registration

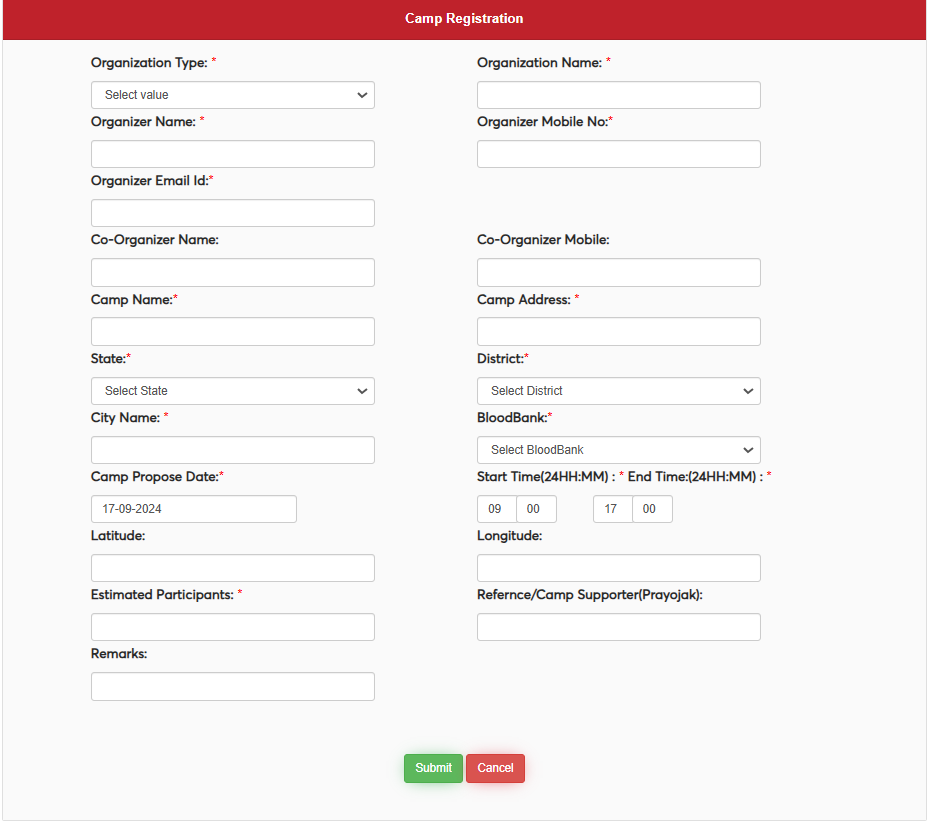


Figure ‑ screen-5 Blood Camp Registration

**Purpose:** This form will allow the target end-users to register and manage blood donation camps in the system. To register a camp, the following information will be encoded in the system.

Table 3.5-1 Screen Element of Blood Camp

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Sr. | Screen Element | Input Type | O/M | 1/N | Description |
| 1 | Organization Type | Dropdown | M | 1 | Select the type of the organizing entity. |
| 2 | Organizer Name | Text | M | 1 | Name of the primary organizer of the camp. |
| 3 | Organizer Mobile No. | Number | M | 1 | Contact number of the organizer. |
| 4 | Organizer Email Id | Email | M | 1 | Email address of the organizer. |
| 5 | Co-Organizer Name | Text | O | 1 | Name of the co-organizer of the camp. |
| 6 | Co-Organizer Mobile | Number | O | 1 | Contact number of the co-organizer. |
| 7 | Camp Name | Text | M | 1 | Name assigned to the blood donation camp. |
| 8 | Camp Address | Text | M | 1 | Full address of the camp location. |
| 9 | State | Dropdown | M | 1 | Select the state where the camp is organized. |
| 10 | District | Dropdown | M | 1 | Select the district where the camp is organized. |
| 11 | City Name | Text | M | 1 | Name of the city where the camp is located. |
| 12 | BloodBank | Dropdown | M | 1 | Select the associated blood bank. |
| 13 | Camp Propose Date | Date Picker | M | 1 | Proposed date for the camp. |
| 14 | Start Time (24HH:MM) | Time Picker | M | 1 | Time when the camp starts (24-hour format). |
| 15 | End Time (24HH:MM) | Time Picker | M | 1 | Time when the camp ends (24-hour format). |
| 16 | Latitude | Text | O | 1 | Latitude coordinates of the camp location. |
| 17 | Longitude | Text | O | 1 | Longitude coordinates of the camp location. |
| 18 | Estimated Participants | Number | M | 1 | Approximate number of participants expected at the camp. |
| 19 | Reference/Camp Supporter (Prayojak) | Text | O | 1 | Name of the camp sponsor or supporter. |
| 20 | Remarks | Text Area | O | 1 | Additional comments or remarks about the camp. |
| 21 | Submit Button | Button | M | 1 | Button to submit the form. |
| 22 | Cancel Button | Button | M | 1 | Button to cancel the form submission. |

## Screen – 6: Donor Feedback

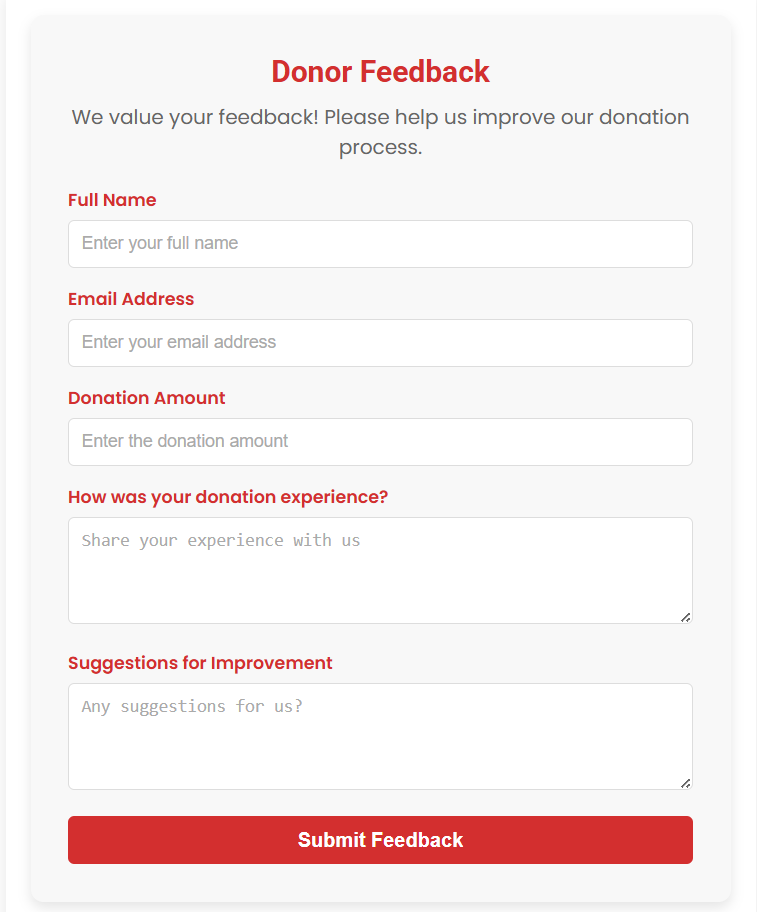


Figure ‑ Screen - 6 Donor Feedback

**Purpose:** This form will allow the target end-users, such as donors and camp organizers, to submit and manage feedback regarding the blood donation experience. The feedback collected will help evaluate the donation process, assess donor satisfaction, and identify areas for improvement in future blood donation camps.

Figure ‑ Screen Element of Donor Feedback

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| sSr. | Screen Element | Input Type | O/M | 1/N | Description |
| 1 | Full Name | Text | M | 1 | The donor's full name. This field is required to identify the person giving feedback. |
| 2 | Email Address | Email | M | 1 | The donor's email address for contacting and following up on their feedback. |
| 3 | Donation Amount | Number | M | 1 | The amount donated by the donor. This helps understand the scale of the donation. |
| 4 | Donation Experience | Textarea | M | 1 | The donor's personal experience while making the donation. |
| 5 | Suggestions for Improvement | Textarea | M | 1 | Feedback or suggestions provided by the donor to improve the donation process. |
| 6 | Submit Button | Button | M | 1 | A button to submit the form after filling in the required fields. |

## Screen – 7: Blood Request Tracking

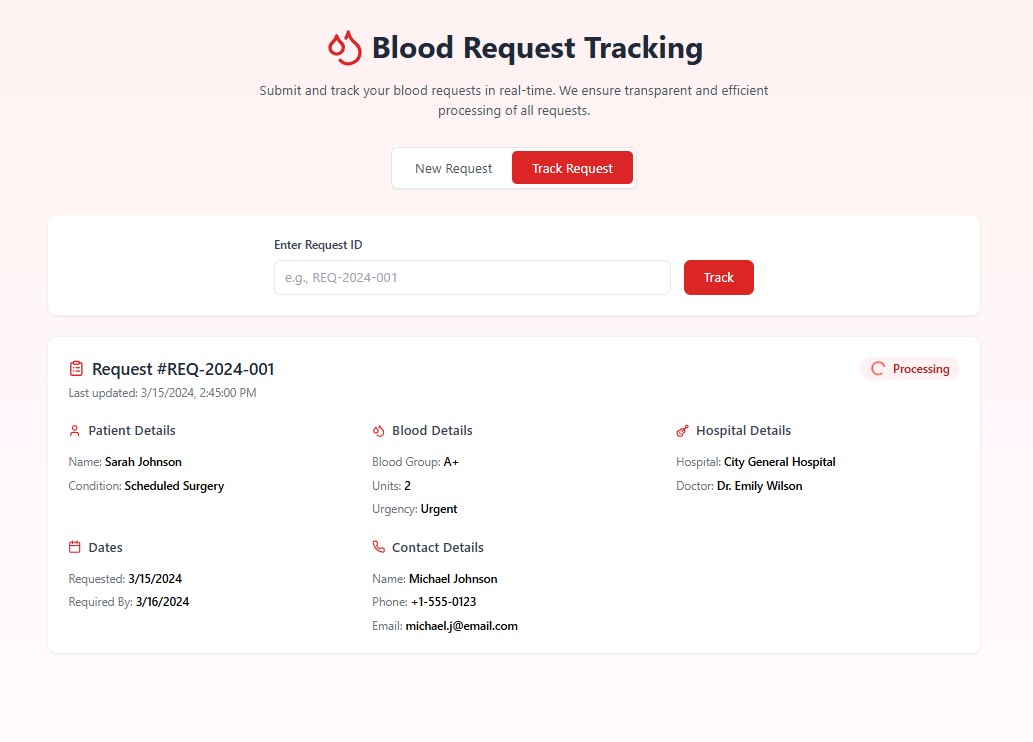


Figure ‑ Screen - 7 Blood Request Tracking

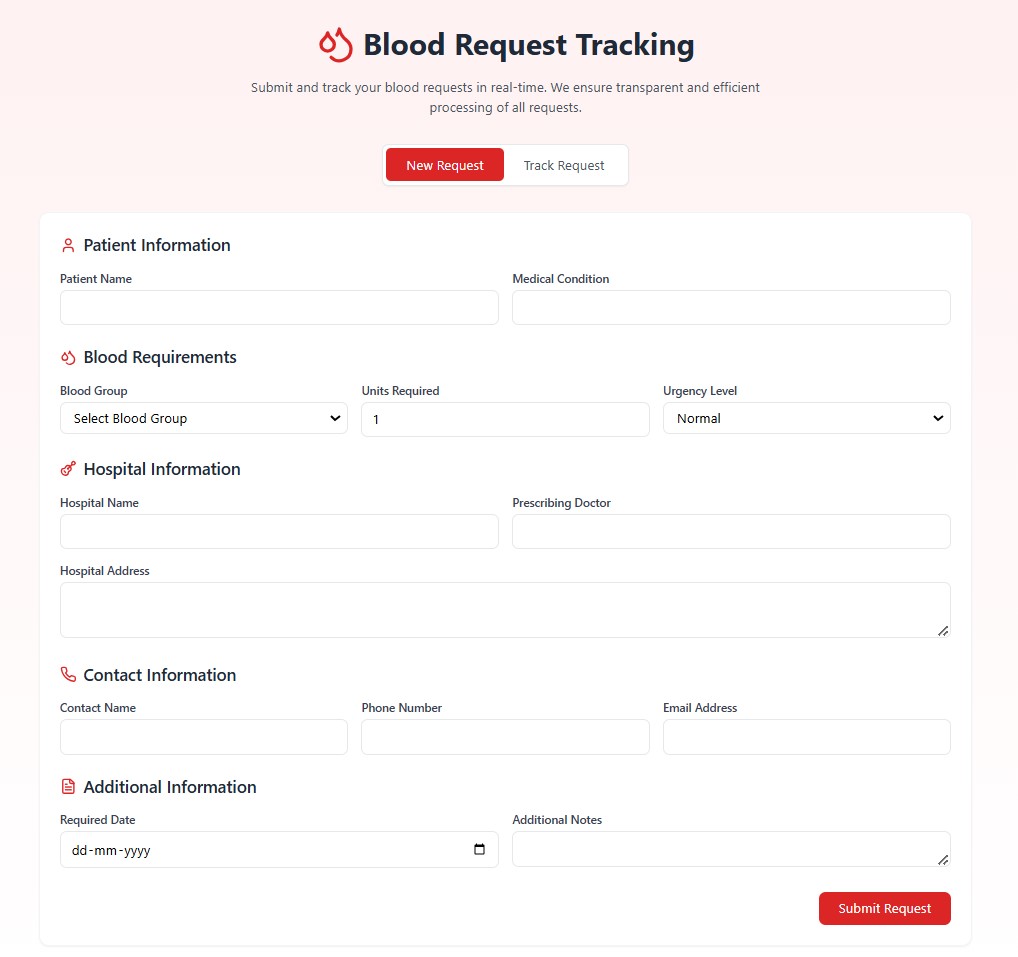


Figure ‑ Screen - 7 Blood Request Tracking - 2

**Purpose:** This form will allow users to submit and track blood requests, ensuring efficient management and timely fulfillment of blood supply needs.

Figure ‑ Screen Element of Blood Request Tracking - 1

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Sr. | Screen Element | Input Type | O/M | 1/N | Description |
| 1 | Patient Name | Text | M | 1 | Input field for the patient's name. |
| 2 | Medical Condition | Text | M | 1 | Input field for specifying the patient's medical condition. |
| 3 | Blood Group | Dropdown | M | 1 | Dropdown to select the required blood group. |
| 4 | Units Required | Number | M | 1 | Numeric input to specify the number of blood units needed. |
| 5 | Urgency Level | Dropdown | M | 1 | Dropdown to choose the urgency level (e.g., Normal, Urgent). |
| 6 | Hospital Name | Text | M | 1 | Input field for the hospital's name. |
| 7 | Prescribing Doctor | Text | O | 1 | Input field for the name of the prescribing doctor. |
| 8 | Hospital Address | Textarea | M | 1 | Text area for entering the hospital's address. |
| 9 | Contact Name | Text | M | 1 | Input field for the contact person's name. |
| 10 | Phone Number | Phone Number | M | 1 | Input field for the contact person's phone number. |
| 11 | Email Address | Email | M | 1 | Input field for the contact person's email address. |
| 12 | Required Date | Date Picker | M | 1 | Date picker to specify the required date for the blood. |
| 13 | Additional Notes | Textarea | O | 1 | Text area for adding any additional notes or comments. |
| 14 | New Request Button | Button | M | 1 | Button to navigate to the "New Request" form. |
| 15 | Track Request Button | Button | M | 1 | Button to navigate to the "Track Request" screen. |
| 16 | Request ID | Text | M | 1 | Input field for entering the request ID to track its status. |
| 17 | Track Button | Button | M | 1 | Button to initiate tracking of a request based on the entered request ID. |
| 18 | Request Details Card | Information Card | O | N | Displays detailed information about a specific blood request. |
| 19 | Status Icon (Processing) | Icon | O | 1 | Visual indicator showing the current status of the blood request (e.g., Processing). |

## Screen – 8: Hospital Complaints



Figure ‑ Screen - 8 Hospital Complaints

**Purpose:** This form will allow users to submit and manage complaints related to hospital services, helping to address issues and improve the overall quality of care provided.

Figure ‑ Screen Element for Hospital Complaints

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Sr. | Screen Element | Input Type | O/M | 1/N | Description |
| 1 | Complaint/Query ID | Text | M | 1 | Input field to enter the unique ID of the complaint or query. |
| 2 | Type | Dropdown | M | 1 | Dropdown to select the type of the query or complaint. |
| 3 | Complainant Type | Dropdown | M | 1 | Dropdown to select the type of complainant (e.g., Patient, Staff). |
| 4 | Complainant Name | Text | M | 1 | Input field for the name of the complainant. |
| 5 | Date Received | Date Picker | M | 1 | Date picker to specify the date the complaint/query was received. |
| 6 | Department | Dropdown | M | 1 | Dropdown to select the relevant department for the complaint or query. |
| 7 | Response | TextArea | M | 1 | Text area to provide a response or resolution to the complaint/query. |
| 8 | Respondent Name | Text | M | 1 | Input field for the name of the person responding to the complaint/query. |
| 9 | Submit Response Button | Button | M | 1 | Button to submit the response form. |

## Screen – 9: Campaign Report Generation

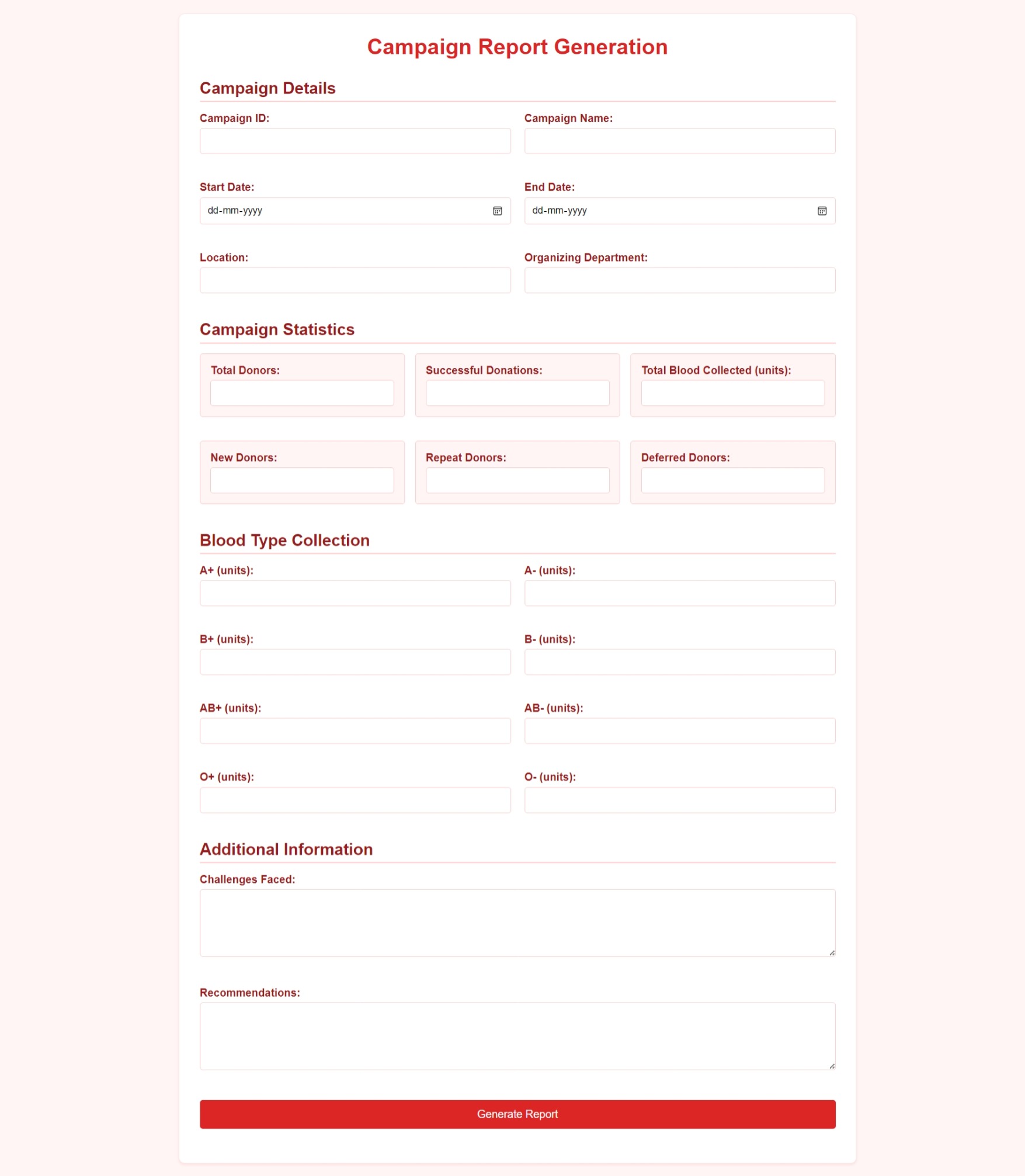


Figure ‑ Screen - 9 Campaign Report Generation

**Purpose:** This form will allow users to generate reports on blood donation campaigns, providing insights into donor participation, collected blood units, and overall campaign effectiveness.

Figure ‑ Screen Elements for Campaign Report Generation

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Sr. | Screen Element | Input Type | O/M | 1/N | Description |
| 1 | Campaign ID | Text Box | M | 1 | Unique identifier for the campaign |
| 2 | Campaign Name | Text Box | M | 1 | Name of the blood donation campaign |
| 3 | Start Date | Date Picker | M | 1 | Start date of the campaign |
| 4 | End Date | Date Picker | M | 1 | End date of the campaign |
| 5 | Location | Text Box | M | 1 | Location where the campaign is conducted |
| 6 | Organizing Department | Text Box | M | 1 | Department responsible for organizing the campaign |
| 7 | Total Donors | Text Box | O | 1 | Total number of donors who participated |
| 8 | Successful Donations | Text Box | O | 1 | Total successful blood donations |
| 9 | Total Blood Collected | Text Box | O | 1 | Total units of blood collected |
| 10 | New Donors | Text Box | O | 1 | Number of first-time donors |
| 11 | Repeat Donors | Text Box | O | 1 | Number of donors who have donated previously |
| 12 | Deferred Donors | Text Box | O | 1 | Number of donors who were deferred |
| 13 | A+ (units) | Text Box | O | 1 | Units of A+ blood collected |
| 14 | A- (units) | Text Box | O | 1 | Units of A- blood collected |
| 15 | B+ (units) | Text Box | O | 1 | Units of B+ blood collected |
| 16 | B- (units) | Text Box | O | 1 | Units of B- blood collected |
| 17 | AB+ (units) | Text Box | O | 1 | Units of AB+ blood collected |
| 18 | AB- (units) | Text Box | O | 1 | Units of AB- blood collected |
| 19 | O+ (units) | Text Box | O | 1 | Units of O+ blood collected |
| 20 | O- (units) | Text Box | O | 1 | Units of O- blood collected |
| 21 | Challenges Faced | Text Area | O | 1 | Any issues encountered during the campaign |
| 22 | Recommendations | Text Area | O | 1 | Suggestions for future improvement |
| 23 | Generate Report Button | Button | M | 1 | Generates the campaign report |

# Database design

## List of Tables

* Donor
* Recipient
* Doctor
* Blood Bank
* Blood Request

Table ‑ Table: Donor

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column | Data Type | Null | Keys & Constrains | Default Value & Description |
| DonorID | int | NN | PK (Auto Increment) |  |
| Name | varchar(100) | NN |  |  |
| Email | varchar(100) | AN |  |  |
| Blood Type | Varchar(50) | NN | CHECK (Blood Type in ('A+', 'A-', 'B+', 'B-', 'AB+', 'AB-', 'O+', 'O-')) |  |
| DOB | DateTime | AN |  |  |

Table ‑ Table: Recipient

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column | Data Type | Null | Keys & Constrains | Default Value & Description |
| RecipientID | int | NN | PK (Auto Increment) | Unique identifier for the recipient. |
| Name | Varchar(50) | NN |  |  |
| Email | Varchar(50) | NN | UNIQUE |  |
| Phone | Varchar(15) | NN |  |  |
| Address | Varchar(100) | NN |  |  |

Table ‑ Table: Doctor

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column | Data Type | Null | Keys & Constrains | Default Value & Description |
| DoctorID | int | NN | PK (Auto Increment) | Unique identifier for the Doctor |
| Name | varchar(50) | NN |  |  |
| Email | Varchar(50) | NN | UNIQUE |  |
| Phone | Varchar(15) | NN |  |  |
| Designation | varchar(100) | NN |  |  |

Table ‑ Table: Blood Bank

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column | Data Type | Null | Keys & Constrains | Default Value & Description |
| BankID | int | NN | PK (Auto Increment) | Unique identifier for the blood bank |
| Name | varchar(50) | NN |  |  |
| Email | Varchar(50) | NN | UNIQUE |  |
| Location | Varchar(100) | NN |  | Location of the blood bank |
| Phone | varchar(15) | NN |  |  |

Table ‑5 Table: Blood Request

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column | Data Type | Null | Keys & Constrains | Default Value & Description |
| RecipientID | int | NN | FK (Auto Increment) |  |
| RequestID | int | NN | PK(Auto Increment) | Unique identifier for the recipient request. |
| Recipient Name | Varchar(50) | NN |  |  |
| Blood Type | Varchar(50) | NN | CHECK (Blood Type in ('A+', 'A-', 'B+', 'B-', 'AB+', 'AB-', 'O+', 'O-')) |  |
| Request Date | DateTime | NN |  |  |
| Blood Quantity | int | NN |  |  |

# Stories and Scenario

## Story-1: Request Blood by Recipient

|  |  |  |
| --- | --- | --- |
| *Story # S1* | : | As a Recipient,  I want to request a specific type of blood  So that I can receive the blood I need for a medical procedure. |
| Priority | **:** | High |
| Estimate | **:** | L |
| Reason | **:** | Allowing recipients to request blood ensures timely fulfilment of critical medical needs. |

### Scenario# S1.1

|  |  |  |
| --- | --- | --- |
| *Scenario# S1.1* | : | Requesting Blood with Available Stock. |
| Prerequisite | **:** | Recipient is logged into their account. |
| Acceptance  Criteria | **:** | **Given:** The recipient navigates to the blood request page.  **When:** The recipient selects the blood type and quantity from available stock.  **Then:** The system confirms the request and prepares the blood for distribution. |

### Scenario# S1.2

|  |  |  |
| --- | --- | --- |
| *Scenario# S1.2* | : | Requesting Blood with Insufficient Stock. |
| Prerequisite | **:** | Recipient is logged into their account. |
| Acceptance  Criteria | **:** | **Given:** The recipient navigates to the blood request page.  **When:** The recipient selects a blood type, but the system shows insufficient stock.  **Then:** The system alerts the recipient and provides an estimated availability date. |

### Scenario# S1.3

|  |  |  |
| --- | --- | --- |
| *Scenario# S1.3* | : | Urgent Blood Request. |
| Prerequisite | **:** | Recipient is logged into their account. |
| Acceptance  Criteria | **:** | **Given:** The recipient indicates that the request is urgent.  **When:** The system prioritizes the request based on urgency.  **Then:** The system sends an alert to the blood bank staff to expedite the process. |

### 5.1.4 Scenario# S1.4

|  |  |  |
| --- | --- | --- |
| *Scenario#* ***S1.4*** | **:** | Modifying a Blood Request. |
| **Prerequisite** | **:** | Recipient has an existing blood request. |
| **Acceptance**  **Criteria** | **:** | **Given:** The recipient wants to modify the quantity or type of blood.  **When:** The recipient updates the request details.  **Then:** The system saves the changes and recalculates availability. |

### 5.1.5 Scenario# S1.5

|  |  |  |
| --- | --- | --- |
| *Scenario#* ***S1.5*** | **:** | Cancelling a Blood Request. |
| **Prerequisite** | **:** | Recipient has an existing blood request. |
| **Acceptance**  **Criteria** | **:** | **Given:** The recipient decides to cancel the request.  **When:** The recipient clicks the cancel option.  **Then:** The system cancels the request and frees up the blood stock. |

## Story-2: Add New Blood Donor to the Blood Bank Database

|  |  |  |
| --- | --- | --- |
| *Story # S2* | : | As a Blood Bank Administrator,  I want to add a new blood donor to the blood bank database  So that the donor's information is available for future blood donation and management |
| Priority | **:** | High |
| Estimate | **:** | XL |
| Reason | **:** | Adding a new donor is crucial for maintaining an up-to-date donor database and ensuring that the blood bank can contact donors for future blood drives. |

### Scenario# S2.1

|  |  |  |
| --- | --- | --- |
| *Scenario# S2.1* | : | Adding a New Donor with Valid Information. |
| Prerequisite | **:** | The administrator is logged into the Blood Bank Management System. |
| Acceptance Criteria | **:** | **Given:**  The administrator is navigated to the donor management page. Valid donor information, including name, contact details, blood type, and medical history, is available. **When**: The administrator selects the "Add New Donor" option  **And**: The administrator enters valid donor details **And**: The administrator clicks the "Save" button to add the donor to the database. **Then**: The system successfully adds the donor to the database and the administrator receives a confirmation message with the donor's unique identification number. |

### Scenario# S2.2

|  |  |  |
| --- | --- | --- |
| *Scenario# S2.2* | : | Adding a New Donor with Invalid Information. |
| Prerequisite | **:** | The administrator is logged into the Blood Bank Management System. |
| Acceptance Criteria | **:** | **Given:** The administrator is on the donor management page  **When**: The administrator selects the "Add New Donor" option  **And**: The administrator enters incomplete or incorrect donor details  **And**: The administrator clicks the "Save" button to add the donor to the database.  **Then**: The system displays error messages for the incorrect or missing information, and the donor is not added to the database. |

### Scenario# S2.3

|  |  |  |
| --- | --- | --- |
| *Scenario# S2.3* | : | Attempting to Add a Duplicate Donor. |
| Prerequisite | **:** | The administrator is logged into the Blood Bank Management System and is on the donor management page. |
| Acceptance Criteria | **:** | **Given**: The donor information, including name and contact details, is already present in the database. **When**: The administrator selects the "Add New Donor" option.  **And**: The administrator enters details that match an existing donor in the database **And**: The administrator clicks the "Save" button to add the donor. **Then**: The system detects the duplicate and prompts the administrator with a warning message, and the donor is not added again. |

## Story-3: Update Blood Inventory Levels

|  |  |  |
| --- | --- | --- |
| *Story # S3* | : | As a Blood Bank Technician, I want to update the blood inventory levels So that the blood bank can keep track of available blood types and quantities accurately. |
| Priority | **:** | High |
| Estimate | **:** | M |
| Reason | **:** | Accurate tracking of blood inventory is essential for ensuring that sufficient blood supplies are available for medical needs and for managing inventory effectively. |

### Scenario# S3.1

|  |  |  |
| --- | --- | --- |
| *Scenario# S3.1* | : | Updating Blood Inventory with Accurate Information. |
| Prerequisite | **:** | The technician is logged into the Blood Bank Management System. |
| Acceptance Criteria | **:** | **Given**: The technician is on the blood inventory management page. Accurate blood inventory information, including blood type, quantity, and expiration date, is available. **When**: The technician selects the "Update Inventory" option **And**: The technician enters valid inventory details **And**: The technician clicks the "Save" button to update the inventory. **Then**: The system successfully updates the blood inventory levels, and the technician receives a confirmation message indicating the updated inventory details. |

### Scenario# S3.2

|  |  |  |
| --- | --- | --- |
| *Scenario# S3.2* | : | Updating Blood Inventory with Inaccurate Information. |
| Prerequisite | **:** | The technician is logged into the Blood Bank Management System. |
| Acceptance Criteria | **:** | **Given**: The technician is on the blood inventory management page **When**: The technician selects the "Update Inventory" option **And**: The technician enters inaccurate or incomplete inventory details **And**: The technician clicks the "Save" button to update the inventory. **Then**: The system displays error messages for the inaccurate or incomplete information, and the inventory is not updated. |

### Scenario# S3.3

|  |  |  |
| --- | --- | --- |
| *Scenario# S3.3* | : | Attempting to Update Inventory with Expired Blood. |
| Prerequisite | **:** | The technician is logged into the Blood Bank Management System and is on the blood inventory management page. |
| Acceptance Criteria | **:** | **Given**: The blood inventory contains details of blood that is past its expiration date **When**: The technician attempts to update the inventory with quantities of expired blood **And**: The technician clicks the "Save" button.  **Then**: The system detects the expired blood and prompts the technician with a warning message, indicating that expired blood cannot be updated in the inventory, and the inventory is not updated with those quantities. |

# Test cases

|  |  |  |  |
| --- | --- | --- | --- |
| Project Name: | Blood Bank Management System | Test Designed by: | Viraj Odedra |
| Module Name: | **User Registration** | **Test Designed date:** | 20 – 02 - 2025 |
| Release Version: | **1.0** | **Test Executed by:** |  |
|  |  | **Test Execution date:** |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Pre-condition: Web application should be accessible | | | | |
| Test Case ID | **Test Title** | **Test Type** | **Description** | **Test Case ID** |
| TC\_001 | Register with valid details | Functional | Register a new user with valid details in the blood bank system | TC\_001 |
| TC\_002 | Register with invalid email format | Functional | Register a new user with an invalid email format | TC\_002 |
| TC\_003 | Verify registration page elements | GUI | Verify that all elements are present on the registration page | TC\_003 |
| TC\_004 | Register with missing required fields | Functional | Verify that all fields required are filled on the registration page | TC\_004 |

|  |  |
| --- | --- |
| **Test Case Title** | Login to web application with valid credential |
| **Test Type** | Functional |
| **Test Priority** | High |
| **Pre-condition** | Web application should be accessible |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Step** | **Test Case Description** | **Expected Result** | **Actual Result** | **Status** | **Comment** | **Data** | **BUG ID** |
| 1 | Access Web application URL | The site should launch properly | Site launched successfully | Pass |  | <https://example.com/register> |  |
| 2 | Enter valid Username in the username field | Username field should be editable and accept input | Username input accepted | Pass |  | Username: [john.doe@example.com](mailto:john.doe@example.com) |  |
| 3 | Enter valid Email in the email field | Email field should be editable and accept input | Email input accepted | Pass |  | Email: [john.doe@example.com](mailto:john.doe@example.com) |  |
| 4 | Enter valid Password in the password field | Password field should be editable and accept input | Password input accepted | Pass |  | Password: SecurePass123 |  |
| 5 | Click on the Register button | User should be registered and redirected to a confirmation page | User registered successfully and redirected | Pass |  |  |  |

|  |  |
| --- | --- |
| **Test Case Title** | Register with invalid email format |
| **Test Type** | Functional |
| **Test Priority** | Medium |
| **Pre-condition** | Web application should be accessible |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Step** | **Test Case Description** | **Expected Result** | **Actual Result** | **Status** | **Comment** | **Data** | **Bug ID** |
| 1 | Attempt registration with an invalid email format | Error message should be displayed for invalid email format | Error message displayed | Pass |  | Email: john.doe@com |  |
| 2 | Verify error message for missing domain part in email | Display an error message indicating missing domain | Error message displayed | Pass |  | Email: john.doe@ |  |
| 3 | Verify error message for missing "@" in email | Display an error message for missing "@" in email | Error message displayed | Pass |  | Email: john.doeexample.com |  |
| Step 4 | Verify error message for a blank email field | Display an error message for a blank email field | Error message displayed | Pass | Email field left blank | Email: (blank) | - |

|  |  |
| --- | --- |
| **Test Case Title** | Verify registration page elements |
| **Test Type** | GUI |
| **Test Priority** | Medium |
| **Pre-condition** | Web application should be accessible |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Step** | **Test Case Description** | **Expected Result** | **Actual Result** | **Status** | **Comment** | **Data** | **Bug ID** |
| 1 | Launch application with the given URL | The site should launch properly | Site launched successfully | Pass |  | <https://example.com/register> |  |
| 2 | Verify that the registration screen contains elements such as Username, Email, Password, and Register button | All listed controls displayed properly on the page | Registration page loaded successfully | Pass |  |  |  |
| 3 | Verify that cursor is focused on “Username” text box on page load | Cursor should be focused in Username textbox | Cursor focused in Username textbox | Pass |  |  |  |
| 4 | Verify that tab functionality is working properly | When tab is pressed, cursor should move to next control | Cursor moved to next control | Pass |  |  |  |
| 5 | Verify that all fields such as Username, Email have a valid placeholder | All text fields should have proper placeholders | All text fields have proper placeholders | Pass |  |  |  |
| 6 | Verify that labels float upward when the text field is in focus or filled (In case of floating label) | Labels should float above filled fields | Labels floated above filled fields | Pass |  |  |  |
| 7 | Verify that Register button is functioning correctly | Register button should submit the form | Register button functional. | Pass |  |  |  |

|  |  |
| --- | --- |
| **Test Case Title** | Register with missing required fields |
| **Test Type** | Functional |
| **Test Priority** | High |
| **Pre-condition** | Web application should be accessible |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Step** | **Test Case Description** | **Expected Result** | **Actual Result** | **Status** | **Comment** | **Data** | **Bug ID** |
| 1 | Attempt registration with missing Username field | Error message should be displayed for missing Username | Error message displayed | Pass |  | <https://example.com/register> |  |
| 2 | Attempt registration with missing Email field | Error message should be displayed for missing Email | Error message displayed | Pass |  |  |  |
| 3 | Attempt registration with missing Password field | Error message should be displayed for missing Password | Error message displayed | Pass |  |  |  |
| 4 | Verify that validation messages are properly shown for each missing field | Validation messages should be accurate and visible | Validation messages accurate | Pass |  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Project Name: | Blood Bank Management System | Test Designed by: | Viraj Odedra |
| Module Name: | **Hospital Complaints** | **Test Designed date:** | 20 – 02 - 2025 |
| Release Version: | **1.0** | **Test Executed by:** |  |
|  |  | **Test Execution date:** |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Pre-condition: Web application should be accessible, The user should be on the complaints section of the portal. | | | | |
| Test Case ID | **Test Title** | **Test Type** | **Description** | **Test Case ID** |
| TC\_001 | Submit Complaint with Valid Data | Functional | Test submitting a complaint with valid data (e.g., correct details and description) | TC\_001 |
| TC\_002 | Submit Complaint with Empty Description | Functional | Test submitting a complaint when the description field is left empty | TC\_002 |
| TC\_003 | Submit Complaint with Invalid Contact Info | Functional | Test submitting a complaint with an invalid contact number (e.g., missing digits or letters) | TC\_003 |
| TC\_004 | Verify Complaint Acknowledgment After Submit | Functional | Verify that a complaint acknowledgment is displayed after submission | TC\_004 |

|  |  |
| --- | --- |
| **Test Case Title** | Submit Complaint with Valid Data |
| **Test Type** | Functional |
| **Test Priority** | High |
| **Pre-condition** | Web application should be accessible, and the user should be on the "Submit Complaint" page. |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Step** | **Test Case Description** | **Expected Result** | **Actual Result** | **Status** | **Comment** | **Data** | **Bug ID** |
| 1 | Attempt registration with missing Username field | Error message should be displayed for missing Username | Error message displayed | Pass |  | <https://example.com/register> |  |
| 2 | Attempt registration with missing Email field | Error message should be displayed for missing Email | Error message displayed | Pass |  |  |  |
| 3 | Attempt registration with missing Password field | Error message should be displayed for missing Password | Error message displayed | Pass |  |  |  |
| 4 | Verify that validation messages are properly shown for each missing field | Validation messages should be accurate and visible | Validation messages accurate | Pass |  |  |  |

|  |  |
| --- | --- |
| **Test Case Title** | Submit Complaint with Empty Description |
| **Test Type** | Functional |
| **Test Priority** | Medium |
| **Pre-condition** | Web application should be accessible, and the user should be on the "Submit Complaint" page. The description field is mandatory. |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Step** | **Test Case Description** | **Expected Result** | **Actual Result** | **Status** | **Comment** | **Data** | **Bug ID** |
| Step 1 | Navigate to the "Submit Complaint" page | User successfully navigates to the "Submit Complaint" page | Page navigated successfully | Pass | Complaint submission form displayed | N/A | - |
| Step 2 | Enter name and issue, leave description blank | The description field should be mandatory and show an error | Error message displayed | Pass | Error displayed for missing description | Name: "John Doe", Issue: "Delay" | - |
| Step 3 | Leave contact information valid | Contact info should be accepted | Contact info accepted | Pass | Contact information correctly accepted | Phone: "123-456-7890", Email: "john@example.com" | - |
| Step 4 | Click on "Submit" button | The system should not allow submission without a description | Error message displayed | Pass | Form prevents submission | N/A | - |

# References

* http://www.w3schools.com/html/html\_intro.asp
* https://www.w3schools.com/php/default.asp
* <https://www.javatpoint.com/uml>
* <https://eraktkosh.mohfw.gov.in>s