

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: \_\_\_\_\_\_\_\_\_\_

|  |  |
| --- | --- |
|  | **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](#_Toc188465596)

[List of Tables II](#_Toc188465597)

[1 Introduction 1](#_Toc188465598)

[1.1 Product perspective 1](#_Toc188465599)

[1.2 Product features 1](#_Toc188465600)

[1.2.1 There are four different users who will be using this product: 1](#_Toc188465601)

[1.2.2 The features required for the Admin are: 1](#_Toc188465602)

[1.2.3 The features that are required for Donor are: 1](#_Toc188465603)

[1.2.4 The features required for the Recipient are: 1](#_Toc188465604)

[1.2.5 The features required for the Hospital are: 1](#_Toc188465605)

[1.3 Functional Requirement 2](#_Toc188465606)

[1.3.1 Blood Donor 2](#_Toc188465607)

[1.3.2 Recipient 2](#_Toc188465608)

[1.3.3 Hospital 2](#_Toc188465609)

[1.3.4 Admin 2](#_Toc188465610)

[1.4 Non-Functional Requirement 3](#_Toc188465611)

[1.4.1 Usability: 3](#_Toc188465612)

[1.4.2 Accuracy: 3](#_Toc188465613)

[1.4.3 Availability: 3](#_Toc188465614)

[1.4.4 Maintainability: 3](#_Toc188465615)

[2 Design and Implementation Constraints 4](#_Toc188465616)

[2.1 Use case diagram 4](#_Toc188465617)

[2.2 Activity diagram and Swimlane diagram 5](#_Toc188465618)

[2.3 Sequence diagram 7](#_Toc188465619)

[2.4 State diagram 8](#_Toc188465620)

[2.5 Class diagram 9](#_Toc188465621)

[2.6 Data flow diagram 10](#_Toc188465622)

[2.6.1 Context diagram (level-0) 10](#_Toc188465623)

[2.6.2 DFD Level-1 11](#_Toc188465624)

[2.6.3 DFD Level-2 12](#_Toc188465625)

[3 External interface requirement (Screens) 13](#_Toc188465626)

[3.1 Screen-1: Blood Stock Availability 13](#_Toc188465627)

[3.2 Screen-2: Thalassemia Request 14](#_Toc188465628)

[3.3 Screen-3: Blood Storage Unit 16](#_Toc188465629)

[3.4 Screen-4: Blood Camp Schedule 17](#_Toc188465630)

[3.5 Screen-5: Camp Registration 18](#_Toc188465631)

[4 Database design 20](#_Toc188465632)

[4.1 List of Tables 20](#_Toc188465633)

[5 Stories and Scenario 22](#_Toc188465634)

[5.1 Story-1: Add New Book in Library Catalogue 22](#_Toc188465635)

[5.1.1 Scenario# S1.1 22](#_Toc188465636)

[5.1.2 Scenario# S1.2 22](#_Toc188465637)

[5.1.3 Scenario# S1.3 22](#_Toc188465638)

[5.2 Story-2: Search Book 23](#_Toc188465639)

[5.3 Story-3: Manage due date for borrowed book 23](#_Toc188465640)

[5.4 Story-4: Renew book 23](#_Toc188465641)

[5.5 Story-5: Generate a report on book usage and availability 23](#_Toc188465642)

[6 Test cases 24](#_Toc188465643)

[7 References 28](#_Toc188465644)

# List of Figures

[Figure 2.1 Use case diagram for Blood Bank management system 4](#_Toc188465647)

[Figure 2.2 Activity Diagram for Blood Donor 5](#_Toc188465648)

[Figure 2.3 Activity Diagram for Blood Recipient 5](#_Toc188465649)

[Figure 2.4 Swimlane Diagram For Blood Recipient 6](#_Toc188465650)

[Figure 2.5 Swimlane Diagram For Blood Donor 6](#_Toc188465651)

[Figure 2.6 Sequence Diagram For Blood Donor 7](#_Toc188465652)

[Figure 2.7Sequence Diagram for Blood Recipient 7](#_Toc188465653)

[Figure 2.8 State Diagram For Blood 8](#_Toc188465654)

[Figure 2.9 State Diagram for Blood Request 8](#_Toc188465655)

[Figure 2.10 Class diagram for Blood Bank Management System 9](#_Toc188465656)

[Figure 2.11 Context diagram for Blood Bank Management System 10](#_Toc188465657)

[Figure 2.12 DFD level-1 for Blood Bank Management System 11](#_Toc188465658)

[Figure 2.13 DFD level-2 for Issue book 12](#_Toc188465659)

[Figure 3.1 Screen-1: Blood Stock Availability 13](#_Toc188465660)

[Figure 3.2 Screen-2: Thalassemia Request 14](#_Toc188465661)

[Figure 3.3 Screen-3: Blood Storage Unit 16](#_Toc188465662)

[Figure 3.4 Screen-4: Search Donor Availability 17](#_Toc188465663)

[Figure 3.5 screen-5 Blood Camp Registration 18](#_Toc188465664)

[Figure 3.6-1 Screen-5: Schedule an appointment 19](#_Toc188465665)

# List of Tables

[Table 3.1‑1 Screen element of Blood Stock Availability 13](#_Toc188465666)

[Table 3.2‑1 Screen element of Thalassemia Request 14](#_Toc188465667)

[Table 3.3‑1 Screen element of Blood Storage Unit 16](#_Toc188465668)

[Table 4.1‑1 Table: Book 20](#_Toc188465669)

[Table 4.1‑2 Table: Borrower 21](#_Toc188465670)

[Table 4.1‑3 Table: Staff 21](#_Toc188465671)

[Table 4.1‑4 Table: Student 21](#_Toc188465672)

# 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.
* Generate Reports: Create detailed reports on blood usage, stock trends, and hospital contributions.

### 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

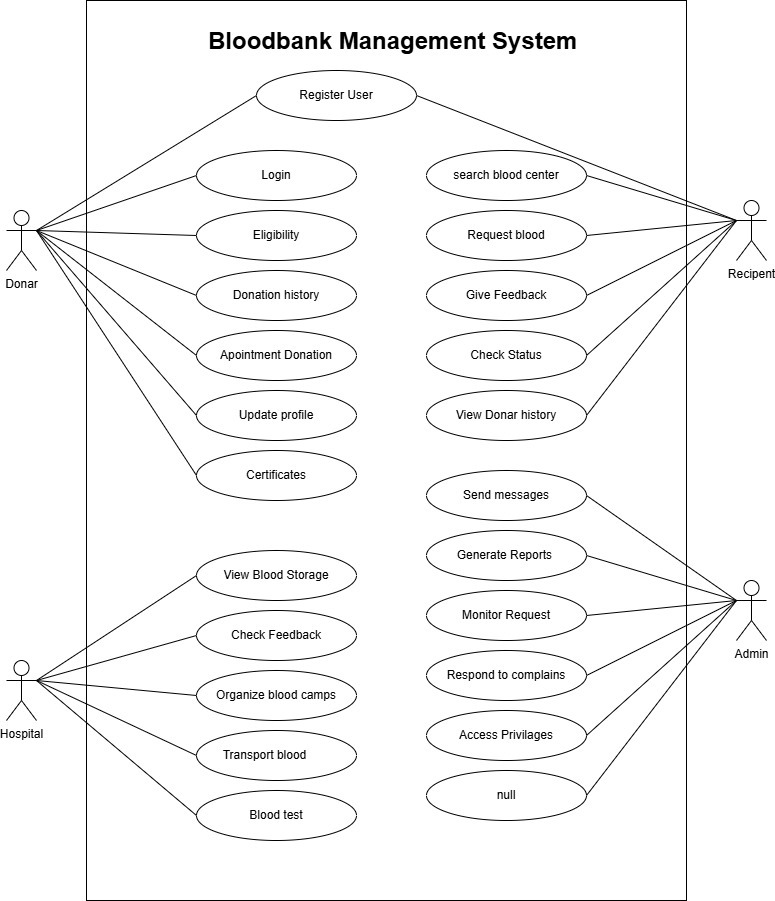


Figure 2.1 Use case diagram for Blood Bank management system

## Activity diagram and Swimlane diagram

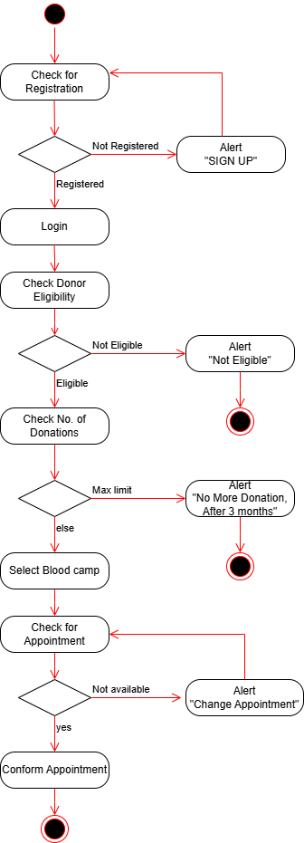


Figure 2.2 Activity Diagram for Blood Donor

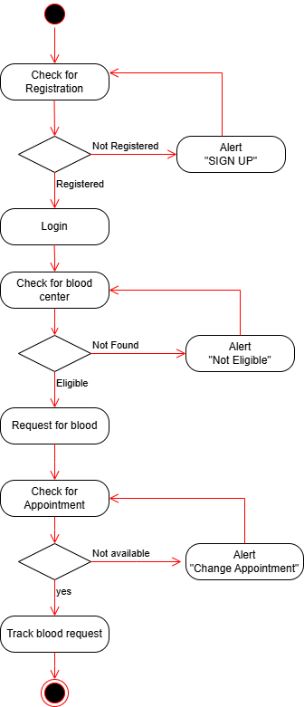


Figure 2.3 Activity Diagram for Blood Recipient

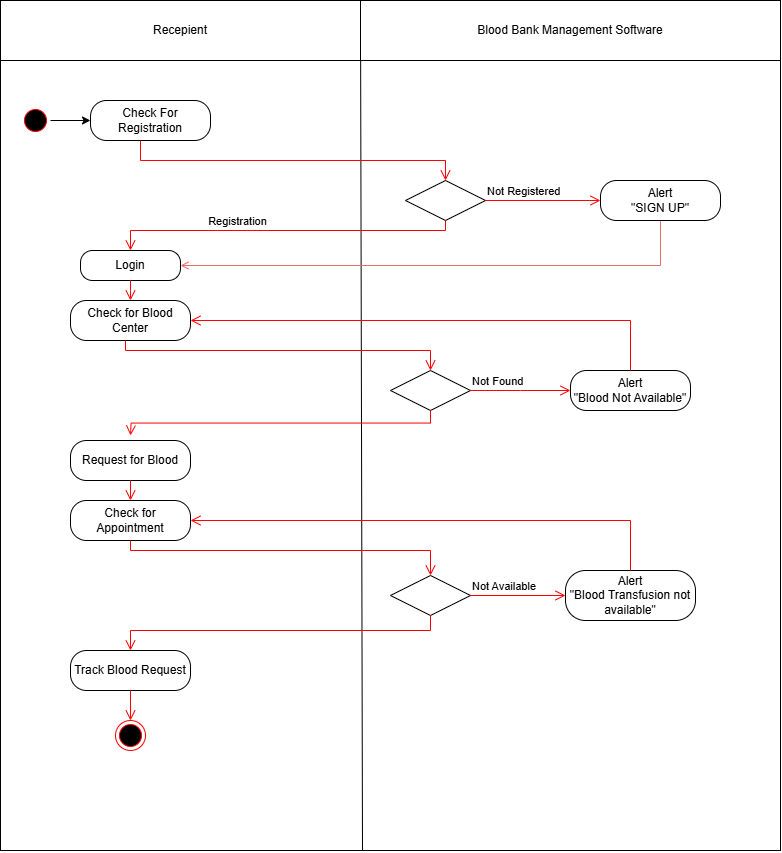


Figure 2.4 Swimlane Diagram For Blood Recipient

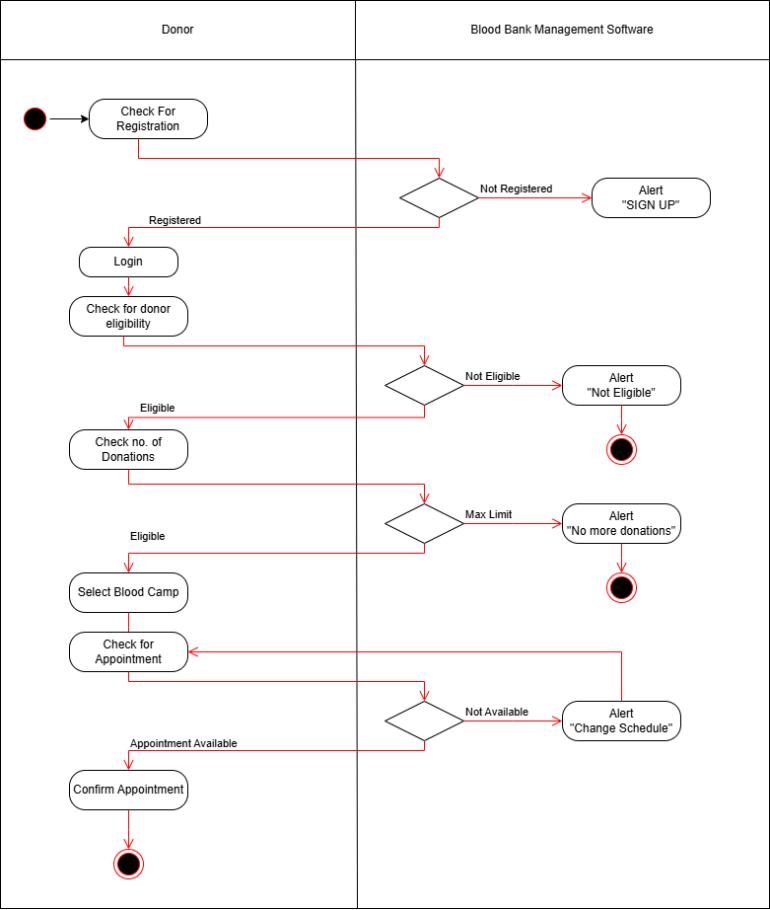


Figure 2.5 Swimlane Diagram For Blood Donor

## Sequence diagram

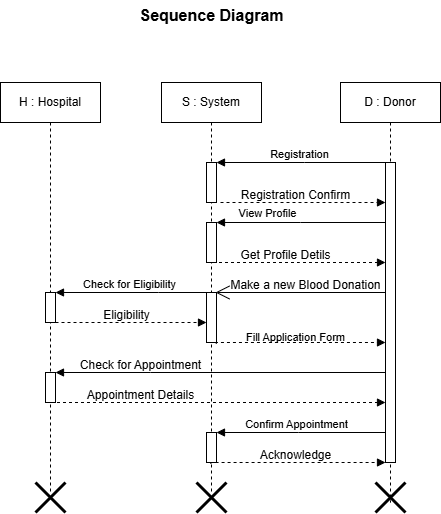


Figure 2.6 Sequence Diagram For Blood Donor

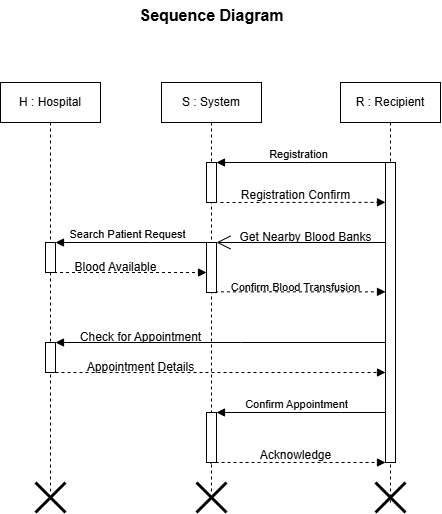


Figure 2.7Sequence Diagram for Blood Recipient

## State diagram

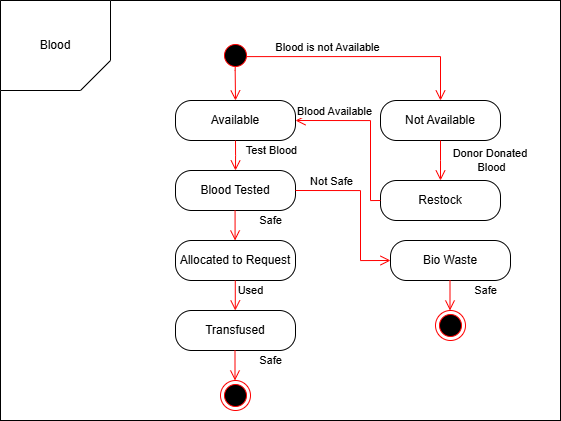


Figure 2.8 State Diagram For Blood

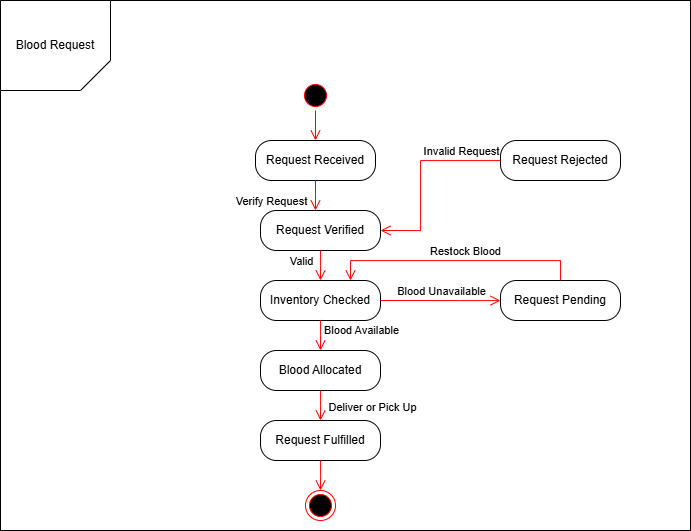


Figure 2.9 State Diagram for Blood Request

## Class diagram

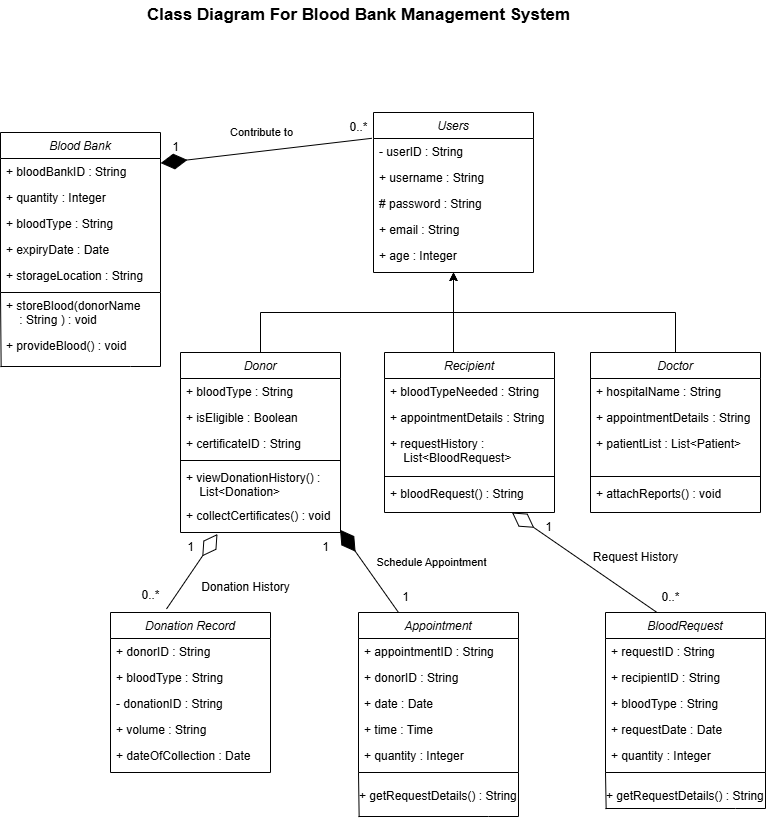


Figure 2.10 Class diagram for Blood Bank Management System

## Data flow diagram

### Context diagram (level-0)

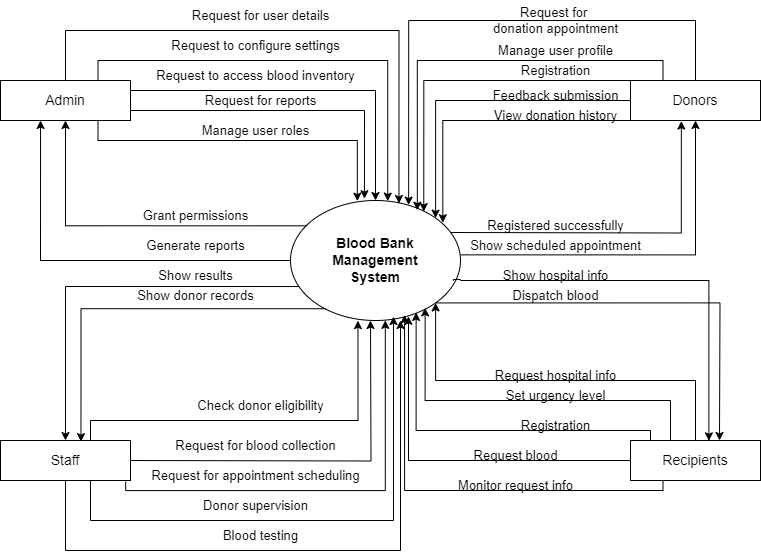


Figure 2.11 Context diagram for Blood Bank Management System

### DFD Level-1

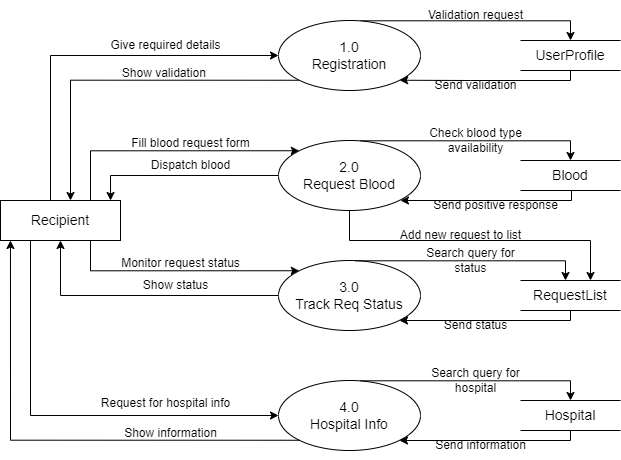


Figure 2.12 DFD level-1 for Blood Bank Management System

### DFD Level-2



Figure 2.13 DFD level-2 for Issue book

# External interface requirement (Screens)

## Screen-1: Blood Stock Availability

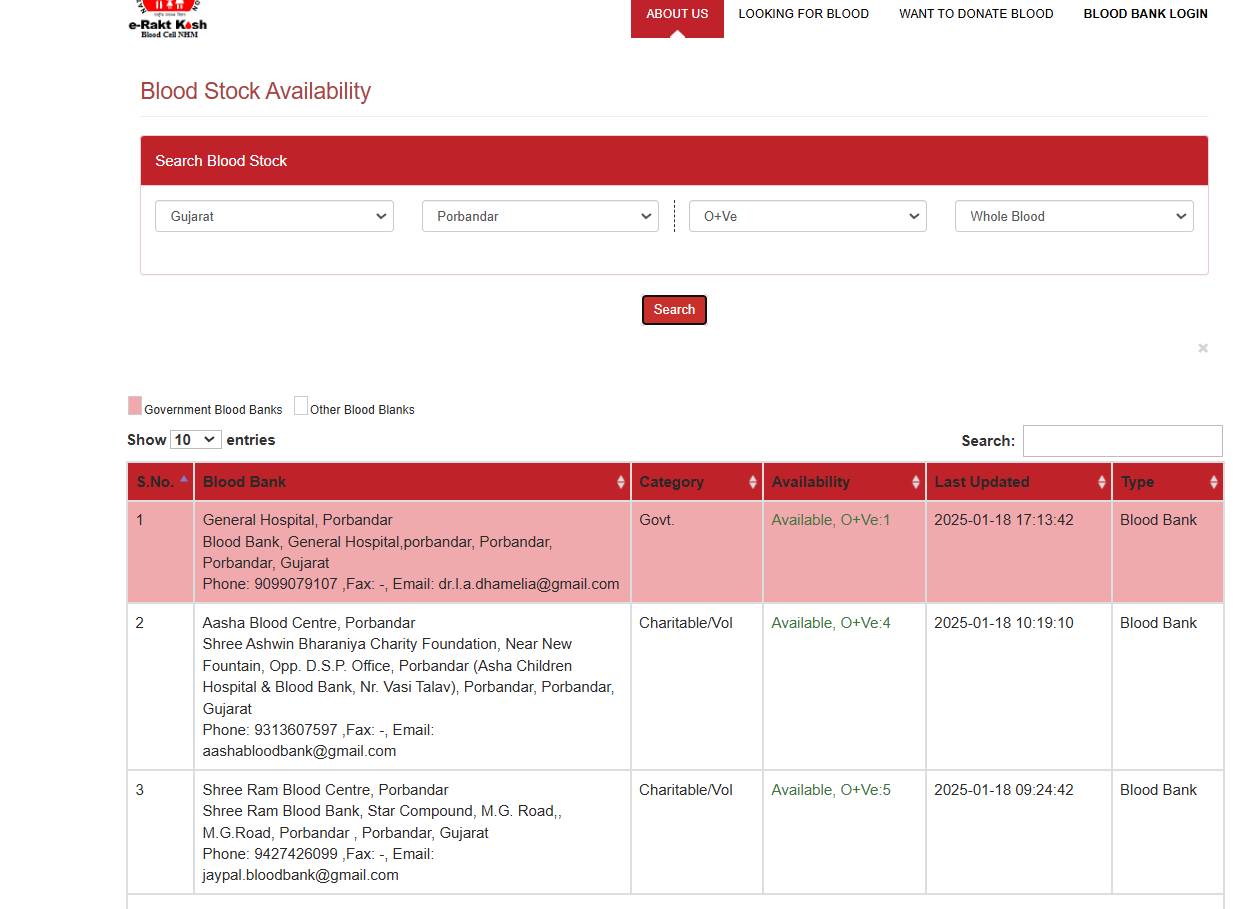


Figure 3.1 Screen-1: Blood Stock Availability

**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 3.1‑1 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: Thalassemia Request

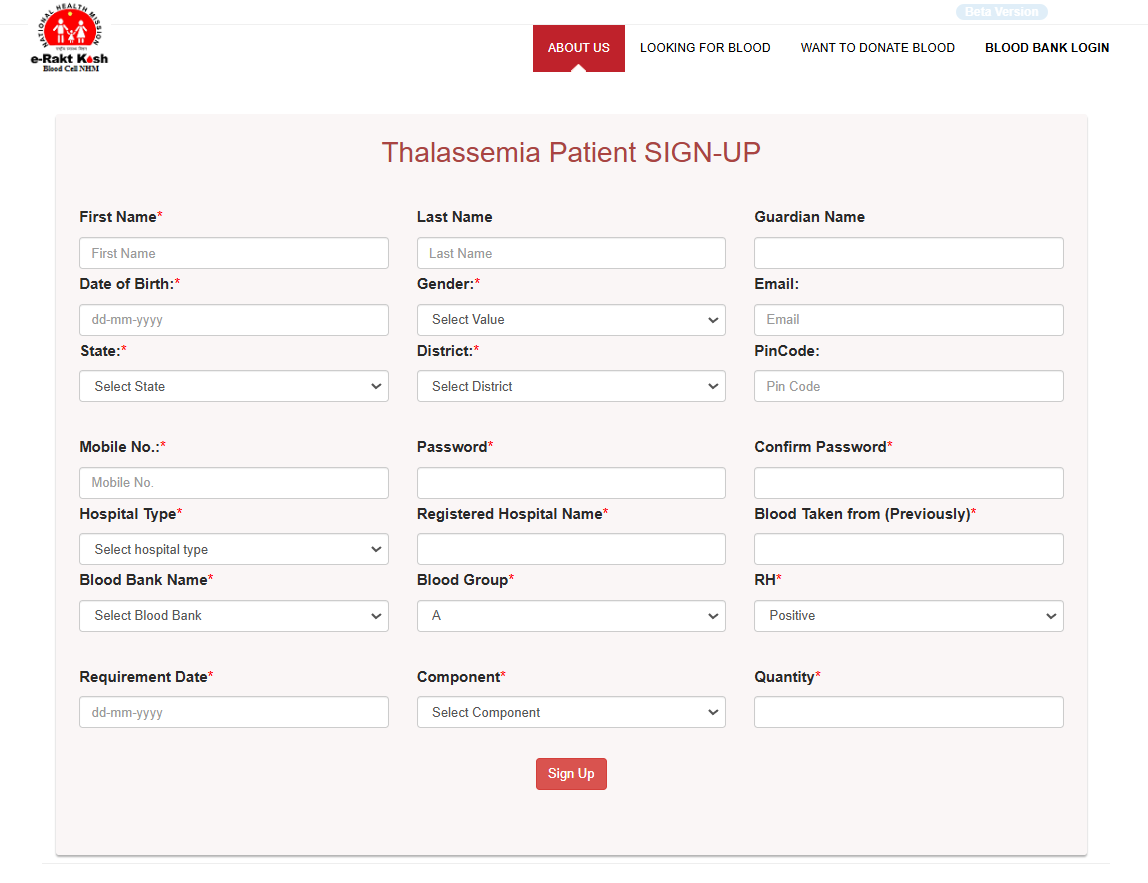


Figure 3.2 Screen-2: Thalassemia Request

**Purpose:** This form will allow the target end-users to request for thalassemia treatment at nearest blood banks in the system. To make request for thalassemia treatment, the following information will be encoded in the system.

Table 3.2‑1 Screen element of Thalassemia Request

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Sr. | Screen Element | Input Type | O/M | 1/N | Description |
| 1 | First Name | Text Input | M | 1 | Enter first name of patient. |
| 2 | Last Name | Text Input | O | 1 | Enter last name of patient. |
| 3 | Guardian Name | Text Input | O | 1 | Enter guardian/parents name of patient. |
| 4 | Date of Birth | Date Input | M | 1 | Enter Date of Birth of patient. |
| 5 | Gender | Select | M | 1 | Select gender of patient. |
| 6 | Email | Text Input | O | 1 | Enter email of patient. |
| 7 | State | Select | M | 1 | Select state for a blood bank in which you are looking for thalassemia treatment. |
| 8 | District | Select | M | 1 | Select district for a blood bank in which you are looking for thalassemia treatment. |
| 9 | Pin Code | Text Input | O | 1 | Enter Pin code of selected district. |
| 10 | Mobile No | Text Input | M | 1 | Enter mobile number of patient. |
| 11 | Password | Text Input | M | 1 | Enter password. |
| 12 | Confirm password | Text Input | M | 1 | Enter confirm password. |
| 13 | Hospital Type | Select | M | 1 | Select hospital type. |
| 14 | Registered  Hospital Name | Text Input | M | 1 | Enter registered hospital name. |
| 15 | Blood Taken  From | Text Input | M | 1 | Enter record of previously taken blood. |
| 16 | Blood Bank  Name | Text Input | M | 1 | Write blood bank or hospital name for which you are looking for desired results. |
| 17 | Blood Group | Select | M | 1 | Select blood group for which you are making thalassemia request. |
| 18 | RH | Select | M | 1 | Select RH for which you are making thalassemia request. |
| 19 | Requirement Date | Date Input | M | 1 | Enter date of requirement. |
| 20 | Component | Select | M | 1 | Select blood component for which you are making thalassemia request. |
| 21 | Quantity | Number  Input | M | 1 | Enter quantity of requirement. |
| 22 | Sing Up | Button | M | 1 | Click on sign up button to submit request. |

## Screen-3: Blood Storage Unit

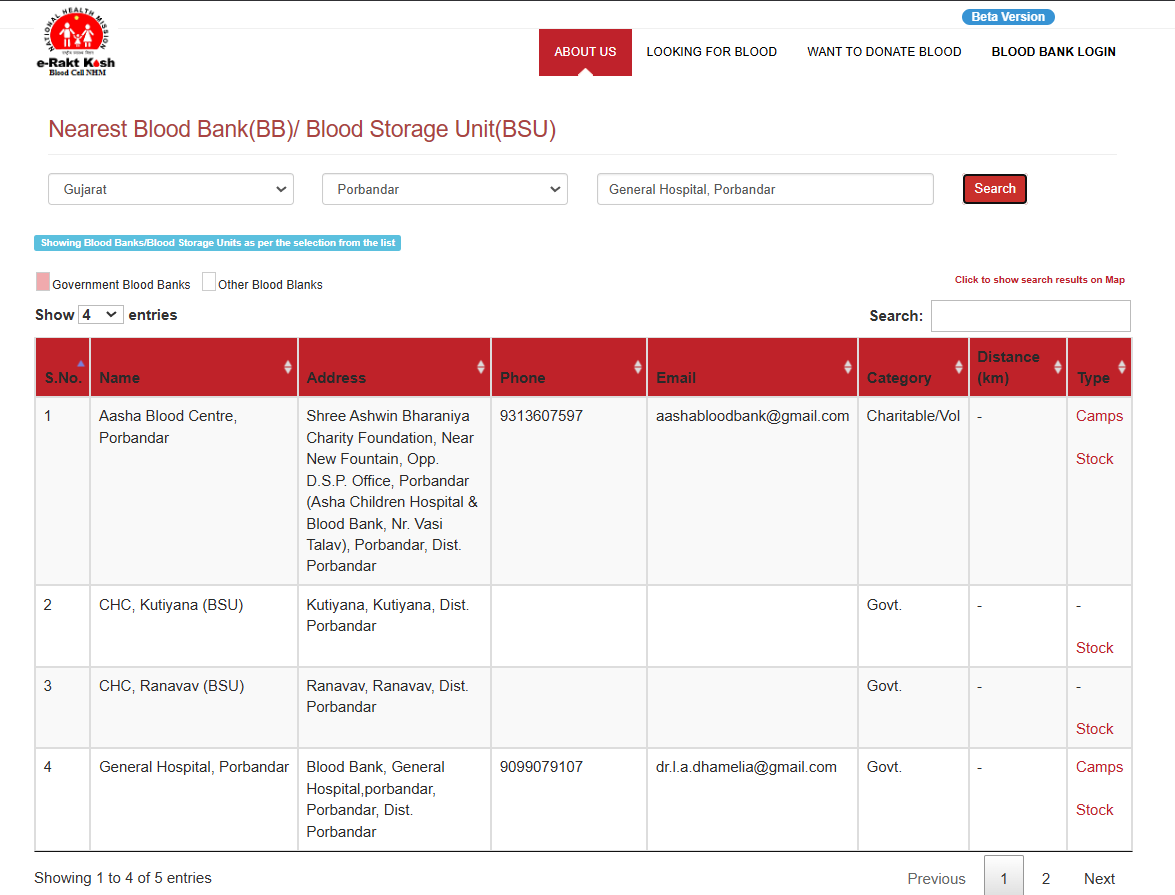


Figure 3.3 Screen-3: Blood Storage Unit

**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 3.3‑1 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

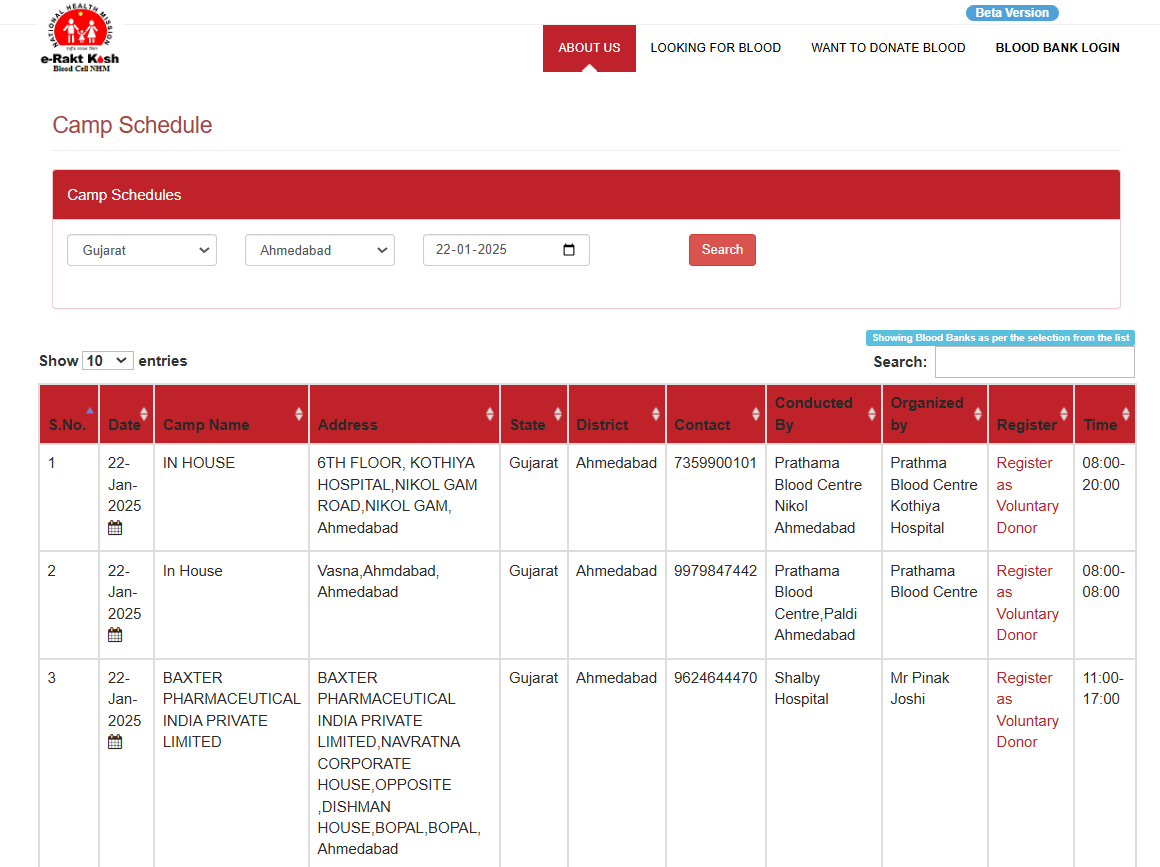


Figure 3.4 Screen-4: Search Donor Availability

**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. |

## 3.5 Screen-5: Camp Registration

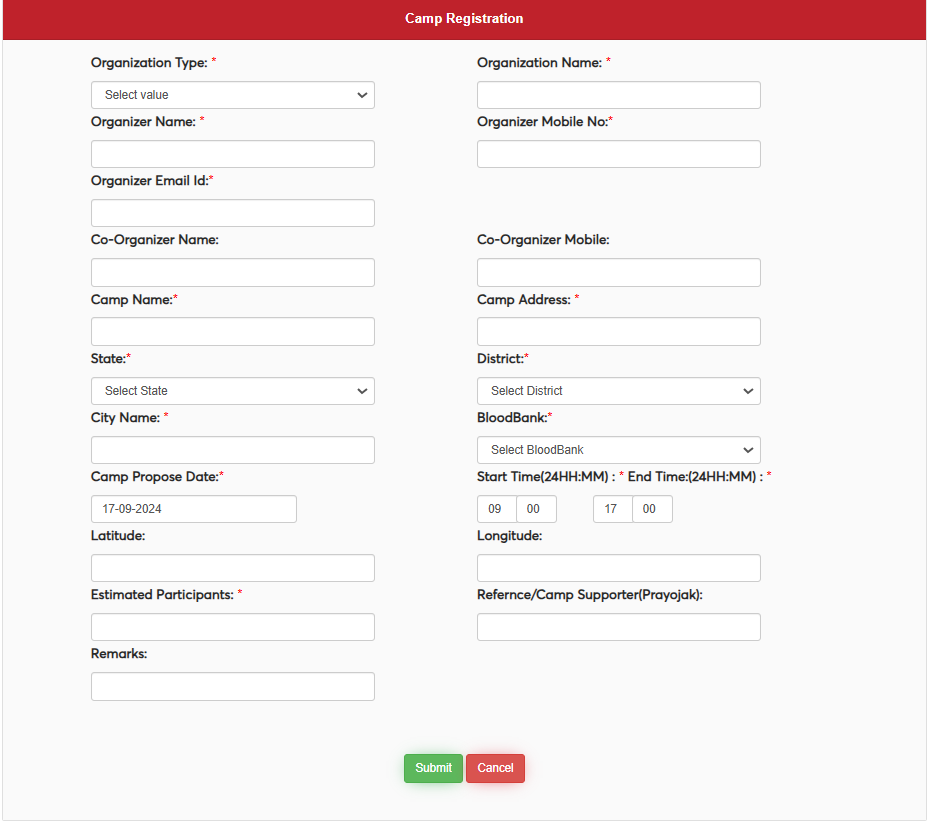


Figure 3.5 screen-5 Blood Camp Registration

**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. |

Figure 3.6-1 Screen-5: Schedule an appointment

# Database design

## List of Tables

* Book
* Borrower
* Student
* Staff

Table 4.1‑1 Table: Book

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column | Data Type | Null | Keys & Constrains | Default Value & Description |
| BookID | int | NN | PK (Auto Increment) |  |
| BookName | varchar(100) | NN |  |  |
| ISBN | varchar(100) | AN |  |  |
| Publication\_year | int | AN |  |  |
| Language | varchar(50) | AN |  |  |

Table 4.1‑2 Table: Borrower

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column | Data Type | Null | Keys & Constrains | Default Value & Description |
| BorrowerID | int | NN | PK (Auto Increment) |  |
| BookID | varchar(100) | NN | FK | Reference of Book Table |
| BorrowedFromDate | DateTime | AN |  |  |
| BorrowedToDate | DateTime | AN |  |  |
| ActualReturnDate | DateTime | AN |  |  |
| IssuedBy | int | NN | FK | Reference of Student and Staff Table |

Table 4.1‑3 Table: Staff

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column | Data Type | Null | Keys & Constrains | Default Value & Description |
| StaffID | int | NN | PK (Auto Increment) |  |
| StaffName | varchar(100) | AN |  |  |
| IsAdmin | Boolean | AN |  |  |
| Designation | varchar(100) | AN |  |  |

Table 4.1‑4 Table: Student

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column | Data Type | Null | Keys & Constrains | Default Value & Description |
| StudentID | int | NN | PK (Auto Increment) |  |
| StudentName | varchar(100) | AN |  |  |
| Gender | varchar(100) | AN |  |  |
| DOB | DateTime | AN |  |  |
| Department | varchar(100) | AN |  |  |
| Contact | number(10,0) | AN |  |  |

# Stories and Scenario

## Story-1: Add New Book in Library Catalogue

|  |  |  |
| --- | --- | --- |
| *Story # S1* | : | As a Librarian,  I want to add a new book in library catalogue  So that everyone can easily find and borrow it. |
| Priority | **:** | High |
| Estimate | **:** | XL |
| Reason | **:** | The addition of a new book to the library catalogue is crucial for ensuring that the library's collection is up-to-date and accessible to everyone. |

### Scenario# S1.1

|  |  |  |
| --- | --- | --- |
| *Scenario# S1.1* | : | Adding a New Book with Valid Information |
| Prerequisite | **:** | Librarian is logged in to the Library management system. |
| Acceptance Criteria | **:** | **Given:**  The Librarian is navigated to the library catalog management page. Valid book information, including title, author, ISBN, and other relevant details is added.  **When:**  The librarian selects the "Add New Book" option  And The librarian enters valid book details  The librarian clicks the "Save" button to add the book to the catalog.  **Then t**he system successfully adds the book to the catalog and the librarian receives a confirmation message with the book's identification number. |

### Scenario# S1.2

|  |  |  |
| --- | --- | --- |
| *Scenario# S1.2* | : | Adding a New Book with Invalid Information. |
| Prerequisite | **:** | The librarian is logged into the library management system. |
| Acceptance Criteria | **:** | **Given:** The librarian is on the library catalogue management page  **When:** The librarian selects the "Add New Book" option and the librarian enters an incomplete or incorrect book details and librarian clicks the "Save" button to add the book to the catalogue.  **Then t**he system displays error messages for the incorrect or missing information and the book is not added to the catalogue. |

### Scenario# S1.3

|  |  |  |
| --- | --- | --- |
| *Scenario# S1.3* | : | Attempting to Add a Duplicate Book |
| Prerequisite | **:** | The librarian is logged into the library management system and the librarian is on the library catalogue management page |
| Acceptance Criteria | **:** | **Given**: The book information, including title, author, ISBN, and other relevant details, is available and the book with the same ISBN is already in the catalogue.  **When**: User Clicks on “Add book” button. Enter a number of copies with the same book detail mentioned in the field.  **Then**: Generate unique book id, barcode and spine label for various book of same title. |

## Story-2: Search Book

|  |  |  |
| --- | --- | --- |
| *Story # S2* | : | As a Librarian or member,  I want to search for books by title, author, or keyword,  So that I can quickly find books that match my interests. |
| Priority | **:** | High |
| Estimate | **:** | M |
| Reason | **:** | Implementing a search functionality is essential for enhancing the user experience, as it allows librarian and member to efficiently discover and access the library's resources. |

## Story-3: Manage due date for borrowed book

|  |  |  |
| --- | --- | --- |
| *Story # S3* | : | As Librarian,  I want to manage due dates for borrowed books  So that I can ensure that books are returned on time and avoid overdue fines. |
| Priority | **:** | High |
| Estimate | **:** | M |
| Reason | **:** | Proper due date management is crucial for maintaining the library's collection and ensuring that books are available for all members. |

## Story-4: Renew book

|  |  |  |
| --- | --- | --- |
| *Story # S3* | : | As Librarian,  I want to renew a book that I have borrowed  So that I can extend my borrowing period if needed. |
| Priority | **:** | Medium |
| Estimate | **:** | M |
| Reason | **:** | Book renewal functionality is a convenience feature for librarian, allowing them to keep a book for an extended period if no one else has requested it. |

## Story-5: Generate a report on book usage and availability

|  |  |  |
| --- | --- | --- |
| *Story # S3* | : | As Librarian,  I want to generate reports on library usage and book availability  So that I can make informed decisions about library’s books. |
| Priority | **:** | Medium |
| Estimate | **:** | L |
| Reason | **:** | Reporting functionality helps librarians track the usage of library resources and make data-driven decisions to improve services and collections. |

# Test cases

|  |  |  |  |
| --- | --- | --- | --- |
| Project Name: | EMI Calculator | Test Designed by: | P. U. Jadeja |
| Module Name: | **Login** | **Test Designed date:** | 01-10-2023 |
| Release Version: | **1.0** | **Test Executed by:** | **R. B. Gondaliya** |
|  |  | **Test Execution date:** | 15-01-2023 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Pre-condition: Web application should be accessible | | | | |
| Test Case ID | **Test Title** | **Test Type** | **Description** | **Test Case ID** |
| TC\_001 | Login to web application with valid credential | Functional | Login to Library management system web application through valid credential | TC\_001 |
| TC\_002 | Login to web application with invalid credential | Functional | Login to Library management system web application through invalid credential | TC\_002 |
| TC\_003 | Varify login page elements | GUI | varify that all elements are availabe on login page | TC\_003 |

|  |  |
| --- | --- |
| **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 launched properly | Site launched successfully | Pass |  | <https://accounts.google.com/ServiceLogin> |  |
| 2 | Enter valid Username in username field | Username field should be editable and accept the Username | Username input accepted | Pass |  | Username:  Rbgondaliya@gmail.com |  |
| 3 | Enter valid Password in Password field | Password field should be editable and accept the password and display as star or dot | Password input displayed in dot and accepted | pass |  | Password: rbgondaliya |  |
| 4 | Enter valid captcha code in captch field | Captch field should editable and accept captcha and captcha is case sensitive | Captcha input accepted | Pass | Step required when human action validation perform | get captcha from image which is near by captcha field |  |
| 5 | Click on login button | User should login into site and navigated to dashboard | User navigated to dashboard and username should br display in top of the right side. | pass |  |  |  |

|  |  |
| --- | --- |
| **Test Case Title** | Login to web application with invalid credential |
| **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 | Verify that User is not able to Login with invalid Username and invalid Password | Should be display an error message enter wrong username or password | Display an error of wrong username and password | Pass |  |  |  |
| 2 | Verify that User is not able to Login with Valid Username and invalid Password | Should be display an error message enter wrong password | Display an error of wrong password | Pass |  |  |  |
| 3 | Verify that User is not able to Login with invalid Username and Valid Password | Should be display an error message User not found | Display an error Username not found | Pass |  |  |  |
| 4 | Verify that User is not able to Login with blank Username or Password | Set required field validation message for Username and Password | Display an error of wrong username and password | Fail | Not performa a validation function fix it |  | Bug\_002 |

|  |  |
| --- | --- |
| **Test Case Title** | Varify login 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 launched properly | Site launched successfully | Pass |  | <https://accounts.google.com/ServiceLogin> |  |
| 2 | Verify that the login screen contains elements such as Username, Password, Sign in button, Remember password check box, Forgot password link, and Create an account link. | All listed control displayed properly on the page | Login page loaded successfully | Pass |  |  |  |
| 3 | Verify that cursor is focused on “Username” text box on the page load | Cursor is focused in Username textbox | Cursor focus in Username textbox | Pass |  |  |  |
| 4 | Verify that tab functionality is working properly or not | When tab pressed cursor move in next control | Cursor moving in next control | Pass |  |  |  |
| 5 | Verify that all the fields such as Username, Password has a valid placeholder | All text fields have proper placeholder | All text fields have proper placeholder | Pass |  |  |  |
| 6 | Verify that the labels float upward when the text field is in focus or filled (In case of floating label) | When field is focused or filled, label display on top of the filled | When field is focus or filled, label display on top of the filled | Pass | step required when fields with floating label |  |  |
| 7 | verify that forgot password link working properly | when click on forgot password load forgot passworg page | forgot password link not working | Fail |  |  |  |

# References

* http://www.w3schools.com/html/html\_intro.asp
* https://www.w3schools.com/php/default.asp
* https://www.javatpoint.com/uml