

**VISVESVARAYA TECHNOLOGICAL UNIVERSITY
BELAGAVI - 590 018, KARNATAKA**



A Mini Project Report on

“BLOOD BANK MANAGEMENT SYSTEM”

Submitted in the partial fulfillment for the requirements for the FS Lab with Mini Project (18ISL67)

in

INFORMATION SCIENCE AND ENGINEERING

By

Mr. YOUR NAME

USN: YOUR USN

Under the guidance of

Dr. Shanthi D L

Assistant Professor

Department of ISE, BMSIT&M.



**DEPARTMENT OF INFORMATION SCIENCE AND ENGINEERING
BMS INSTITUTE OF TECHNOLOGY & MANAGEMNT
YELAHANKA, BENGALURU-560064**

2022-2023

**VISVESVARAYA TECHNOLOGICAL UNIVERSITY
BELAGAVI – 590 018, KARNATAKA**

**BMS INSTITUTE OF TECHNOLOGY & MANAGEMENT
YELAHANKA, BENGALURU-560064**

DEPARTMENT OF INFORMATION SCIENCE & ENGINEERING



CERTIFICATE

This is to certify that the Mini Project work entitled “**Blood Bank Management System**” is a bonafide work carried out by **Mr. YOUR NAME AND USN**)in partial fulfillment of File structures Lab with Mini Project (18ISL67) for the award of **Bachelor of Engineering Degree in Information Science and Engineering** of the Visvesvaraya Technological University, Belagavi during the year 2022-23. It is certified that all corrections/suggestions indicated for Internal Assessment have been incorporated in this report. The project report has been approved as it satisfies the academic requirements in respect of Mini Project work for the B.E Degree.

Signature of the Guide

Dr. Shanthi D L
Assistant Professor
Department of ISE

Signature of the HOD

Dr. Pushpa S K
Professor and Head
Department of ISE

EXTERNAL EXAMINERS

Name of the Examiners

- 1.
- 2.

Signature with Date

ACKNOWLEDGEMENT

We are happy to present this mini project after completing it successfully. This mini project would not have been possible without the guidance, assistance and suggestions of many individuals. We would like to express our deep sense of gratitude and indebtedness to each and every one who has helped us make this mini project a success.

We heartily thank our **Principal, Dr. Mohan Babu G.N, B M S Institute of Technology & Management** for his constant encouragement and inspiration in taking up this mini project.

We heartily thank our **Head of Department Dr. Pushpa S K, Dept. of Information Science and Engineering, B M S Institute of Technology& Management** for his constant encouragement and inspiration in taking up this mini project.

We gracefully thank our Project guide, **Dr. Shanthi D L, Asst. Professor, Dept. of Information Science and Engineering**, for her encouragement and advice throughout the course of the mini project work.

Special thanks to all the staff members of Information Science Department for their help and kind co-operation.

We also thank our parents and friends for their unconditional love and encouragement and support given to us in order to finish this precious work.

Last but not the least we would like to thank God for giving us the strength and motivation through the course of this Project.

By,

YOUR NAME

ABSTRACT

The Blood Bank Management System (BBMS) is an application that stores, processes, retrieves, and analyzes data about blood bank administration. It also supervises the blood inventory management and other blood bank related activities.

The major goal of the blood bank management system is to keep track of blood, donors, seekers, blood groups, and stock information. The project is done at administration level, only administrator can see it.

A blood donation is a process whereby a person voluntarily has blood drawn to be used for future transfusions when in need at hospitals for treatment procedures that require them. Blood often participates in the process of collection of blood and other procedures like managing stocks, approving blood requests, updating donation information.

The project Blood Bank Management system includes registration of donors, seekers, storing their details into the system and also the details regarding the blood groups. It includes a search facility to know the donors available. Users can search details of donors using the primary key (username). The user can generate the report. The donors, seekers details can be registered by filling the register form and create a username, password which can be used to modify the details. The Blood Bank Management system can be entered using a username and password. It is accessible either by administrator or receptionist. Only they can add data into the database. The data can be retrieved easily. The interface is very user-friendly. The data is well protected for personal use and makes the data processing very fast.

TABLE OF CONTENTS

Chapter 1 Introduction	1-2
1.1 Outline	1
1.2 Motivation and Scope	1
1.3 Problem Statement	2
1.4 Objectives	2
1.5 Limitations	2
 Chapter 2 Requirements Specification	 3-4
2.1 Functional Requirements	3
2.2 Non-Functional Requirements	4
2.3 Hardware Requirements	4
 Chapter 3 Requirements and System Analysis	 5-8
3.1 Overall System Design	5
3.2 Components/Subsystem Design	6
 Chapter 4 System Design	 9-11
4.1 UI Logic Interface/Interaction Details	9
 Chapter 5 Implementation	 12-13
5.1 Description of Frameworks Used	12
5.3 Description of Integrated Development	13
 Chapter 6 Testing	 14-18
6.1 Component Tests	14
6.2 System Tests	18
 Chapter 7 Interpretation of Results	 19-23
Conclusion	24
Applications	24
References	25

LIST OF FIGURES

3.1.1. Overall System Design	5
3.2.1: Blood Donor's Panel Flow Diagram	6
3.3.1: Blood Seeker's Panel Flow Diagram	7
3.4.1: Inventory's Panel Flow Diagram	8
3.5.1: Report Panel Flow Diagram	8
4.1.1: Home Page of web application	9
4.1.2: Donors Tab	9
4.1.3: Seeker Menu panel	10
4.1.4: Report Menu panel	10
4.1.5: Blood Inventory Tab	11
7.1: Login page of Blood Bank Management System	19
7.2: Home Page and Donor's Tab	19
7.3: Seeker's Tab	20
7.4: Inventory Tab	20
7.5: Add Donor	21
7.6: Modify Donor	21
7.7: Add Seeker	22
7.8: Modify Seeker	22
7.9: Seeker Report	22

LIST OF TABLES

6.1.1: Blood Donor's Module Tests Table	14
6.1.2: Blood Donor's Module Tests Table	15
6.1.3: Blood Seeker's Module Tests Table	16
6.1.4: Blood Seeker's Module Tests Table	17
6.2.1: Complete System Tests Table	18

Chapter 1

INTRODUCTION

1.1 Outline

This project is a software application that provides details of doctors and seekers with various functionalities. And also details pertaining to the stock information regarding blood that are present in the blood bank. In this project we have implemented file management operations along with the concept of primary indexing implementation in C++ so that the application is robust and managing the system becomes easier too.

Typical Features include:

- Adding new Records – New Records can be added to the donors field, and the seekers field. And their respective details regarding their personal details can also be added.
- Searching individual Records – A user can search a specific record by keying in the Roll Id for the required donor/seeker they are looking for. It displays all the details regarding that record.
- Listing all Records under a category – If a user likes to list out all the records under the donor/seeker category, then they can do so by specifying the category.
- Updating Records – A user can update a specific record by keying in the Roll Id for the required donor/seeker they are looking for.
- Report Generation – The billing details of the seeker who received the blood from the particular donor details, the type of blood, the number of bottles of blood, the amount the seeker has to pay can also be retrieved in a report format.
- Manage Inventory- A user Manage the inventory by adding the stock details like blood groups and their quantity that are available in the blood bank.

1.2 Motivation and Scope

Our present modern information system makes use of computers for the execution, each of them connected through an optimized network. Blood transfusion is an essential component of the health care system of every country and patients who require blood transfusion service as a part of clinical management of their condition has the right to expect that sufficient and safe blood will be available to meet their needs. However, this is not always the case, especially in developing countries. To recruit and retain adequate regular volunteer non enumerated blood donors the motivators and the barriers of blood donors must be understood. Equally important is the knowledge of blood donors. Maintaining the details of donors, seekers and the stock information of blood records safely is tedious. To reduce this type of burdens and to manage the financial, blood bank administration and stock aspects, Blood Bank management system came into existence.

1.3 Problem Statement

Design and develop a management System for a Blood Bank to perform all required details processing operations using Files to store, retrieve and render from. Such operations can include Creation of files where the record are inserted, deletion of records from the files, Modification of file content, finding the required contents from the file, Report generation.

1.4 Objectives

- Develop a user-friendly and responsive web application for blood bank management using React and Flask technologies.
- Implement a comprehensive blood donor registration module to collect and maintain donor information, including personal details, medical history, and blood type.
- Enable efficient blood inventory management, allowing administrators to track available blood units, their types, quantities, and expiration dates.
- Conduct thorough testing and debugging to ensure the system is reliable, stable, and scalable.
- Provide comprehensive documentation to assist users in understanding the system's functionality, features, and usage.
- Collaborate with blood banks and other stakeholders to gather feedback and make improvements to the system based on user needs and requirements.

1.5 Limitations

Compatibility: As the project involves using React and Flask, it's essential to consider the compatibility of the system across different browsers and devices. Ensuring responsiveness and compatibility with popular browsers and mobile devices may require additional development and testing efforts.

User Training and Adoption: Introducing a new system to blood bank staff and users may require training and familiarization. The learning curve and resistance to change could potentially affect the adoption and acceptance of the system.

Integration with Existing Systems: Blood banks may already have existing systems in place for managing inventory, donors, and other processes. Integrating your blood bank management system with these existing systems could be a challenge.

Maintenance and Support: Once the system is deployed, ongoing maintenance and support will be essential to address any bugs, perform updates, and provide technical assistance. Planning for a sustainable maintenance and support strategy is crucial to ensure the long-term success of the system.

Chapter 2

REQUIREMENTS SPECIFICATION

2.1 Functional Requirements

Creating a donor/seeker record

This requires the user to create a new record with the help of B-tree nodes, using this, the data in the file, to enter its contents into a buffer and save the buffer contents to memory when creating it, basically this buffer is used as a node.

Searching for a specific existing record

Here, the user can search for specific doctor/patient details. They can also look for a particular doctor's details like the qualifications, area of expertise and other details. They can also look for a patient's details like the ailment's history, current treatment and other details.

Search for a list of records under a category

Here the user can access the data of all the doctors and patients working for or admitted in the hospital. They can retrieve the data based on certain parameters which will help them to identify the required doctor who fulfills the requirements of the patient.

Deleting a specific record

Once the patient is discharged from the hospital, the patient's record can be deleted using the patient ID. The same can be done for a doctor if they leave the hospital. When a patient is discharged, the vacated room which was allocated to the patient would be up for new engagement.

Modify a specific record

If the user chooses to edit an existing record, they can do so with the help of the ID that is issued to the specific doctor/ patient. This will help in keeping the data UpToDate.

Listing out the details

The required details of certain facilities that are available at the hospital will be listed for the ease of getting the information that is in need. Details of the occupied rooms and also that of the once that are vacant will also be shown to the user.

2.2 Non-Functional Requirements

Performance

Performance of this System should always vary between a few hundred milliseconds. Time taken to create files, open existing files, add a record, delete a record, update a record, flush contents of buffer onto disk when user hits save, find keywords within the file, replacing keywords within the file should be minimal.

Reliability

It shall always provide fast and flexible creation, insertion, deletion, updating and traversal within the File. It has been tested multiple times with various inputs. It shall be able to recover from hardware failures, power failures and other natural catastrophes and rollback the files to their most recent valid state.

Usability

It is quite simple to use, as it runs on web browser when sent a request for specified url. This demands users with no expertise in any field whatsoever. Being a web application, it's quite easy to navigate through its functionalities. The status of insertion, deletion, search result is clearly displayed on the Screen as “Success” or “Failure” using Toasts.

Integrity

Since there are separate files stored for Donors, Seekers and Inventory Availability and the status of a change can either result in a success which leads to permanent change in the respective file or may lead to a failure which eventually leads to the retainment of the most recent stable & accepted state the data is free from corruption.

Interoperability

Since it is a web application, by default, makes it platform independent. It can be opened in any environment and browser and it will continue to operate without any glitches.

2.3 Hardware Requirements

Hardware limitations: Should have 4GB RAM

Control functions: The software must be very user-friendly and display appropriate error messages.

Dependencies: Requires VS CODE. **Parallel operations:** It must support many file operations simultaneously.

Parallel operations: It must support many file operations simultaneously.

Chapter 3

SYSTEM/REQUIREMENTS ANALYSIS

3.1 Overall System Description

The overall system description describe abt how the system is designed and the below figure 3.1 shows the flow chat of overall system design.

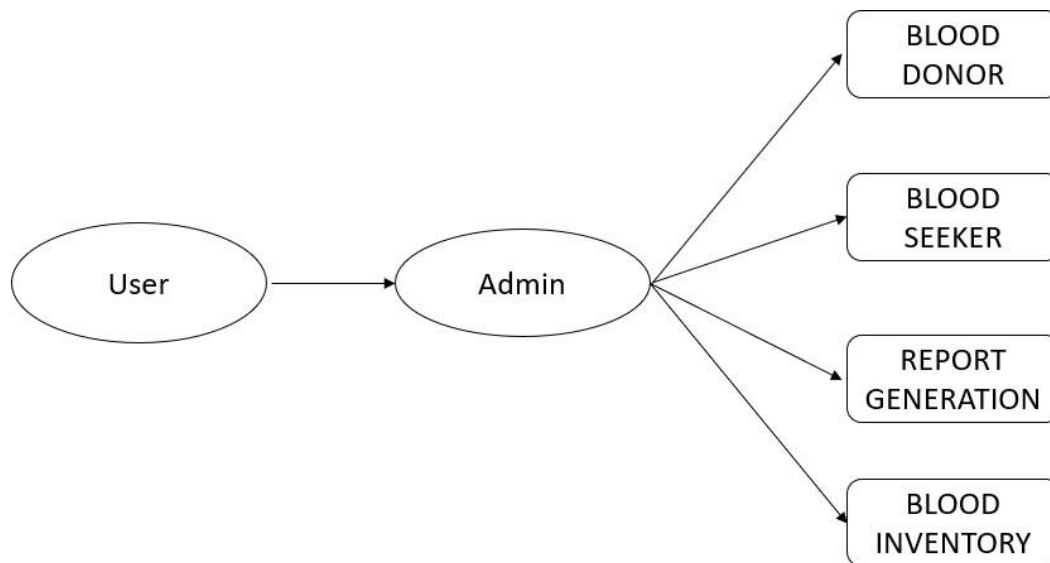


Fig. 3.1.1: Overall System Design

The overall description of the system is as follows:

The user is first presented with the Main Panel. Here the user has a set of four options.

Blood Donor: In this Module, users can add a donor record, can view all donors, can search a particular donor record, can update an existing donor record, and can go back to the home page.

Blood Seeker: In this Module, users can add a seeker record, can view all seeker, can search a particular seeker record, can update an existing seeker record, and can go back to the home page.

Report Generation: In this Module, user can fill the details to generate the report and exit the program with the help of any key.

Blood Inventory: In this Module, user can view all the stock available in the inventory, can search for specific blood group, can update the record and can go back to the home page.

3.2 Components/Subsystem Design

In the Donor's module, the user is prompted with 5 options; Add panel, Display panel, Searchpanel, delete panel, modify panel and go to the home page. Figure 3.2.1 shows the donors panel flow diagram

Add panel: User can enter the donor record by filling in all the details

Display Panel: User can list all donors in the file system would be shown upon selecting this option.

Search Panel: User is prompted to enter a unique ID for the donor that is to be searched and upon entering, the program outputs the required information about the donor.

Update Panel: Users can delete records based on the unique ID assigned against each donor record.

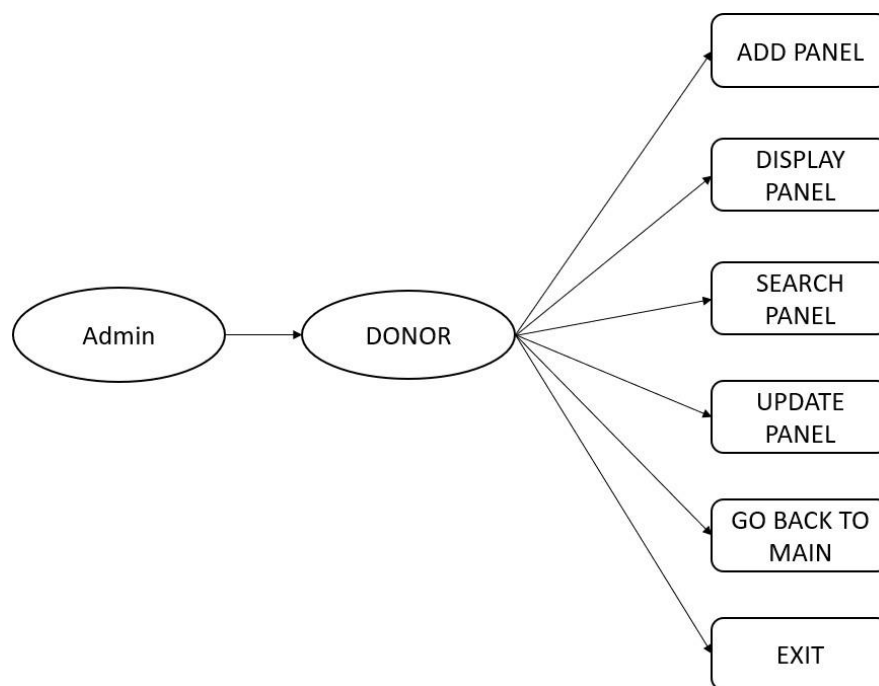


Fig. 3.2.1: Blood Donor's Panel Flow Diagram

3.3: Seeker's Module

In the Seeker's module, the user is prompted with options; Add panel, Display panel, Search panel, update panel, and go to the home page. Figure 3.3.1 below shows the seekers panel flow diagram

Add panel: User can enter the seeker record by filling in all the details.

Display Panel: User can list all seekers in the file system would be shown upon selecting this option.

Search Panel: User is prompted to enter a unique ID for the seeker that is to be searched and upon entering, the program outputs the required information about the seeker.

Update Panel: Users can delete records based on the unique ID assigned against each seeker.

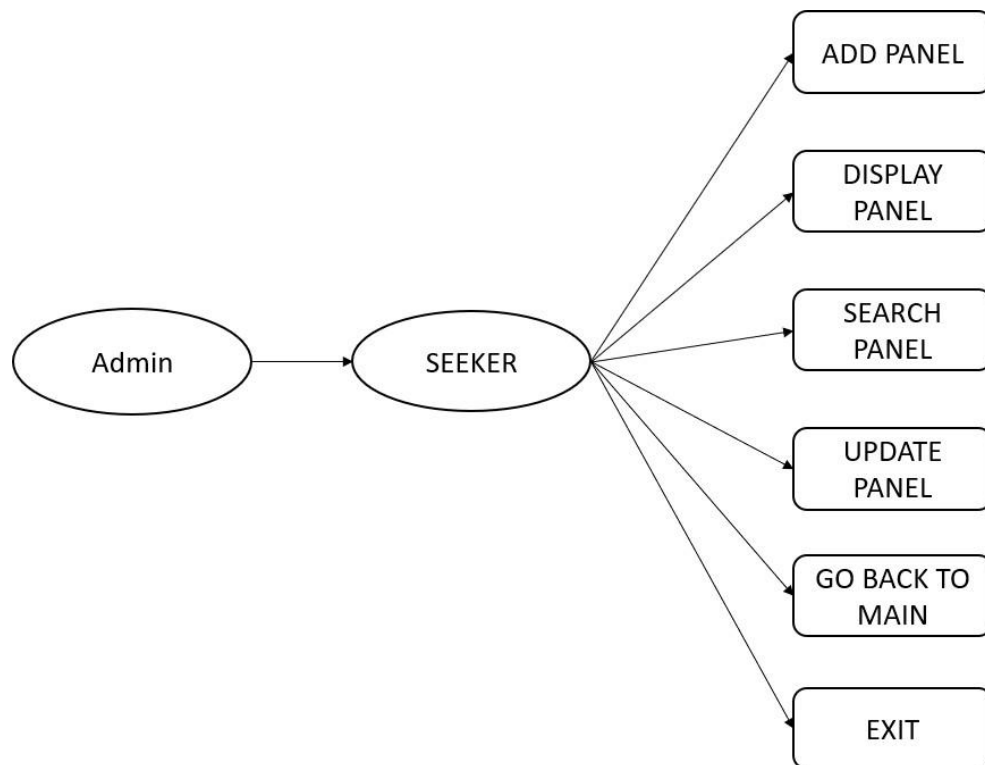


Fig. 3.3.1: Blood Seeker's Panel Flow Diagram

3.4: Inventory Module

In the inventory module, the user is prompted with options; Add panel, Display panel, Search panel, update panel, and go to the home page. Figure 3.4.1 shows the Inventory panel flow diagram.

Add panel: User can enter the stock information by filling in all the details.

Display Panel: User can list all blood available in the file system would be shown upon selecting this option.

Search Panel: User is prompted to enter a Blood Group that is to be searched and upon entering, the program outputs the required information about the Blood Group.

Update Panel: Users can update records based on the Blood Group available.

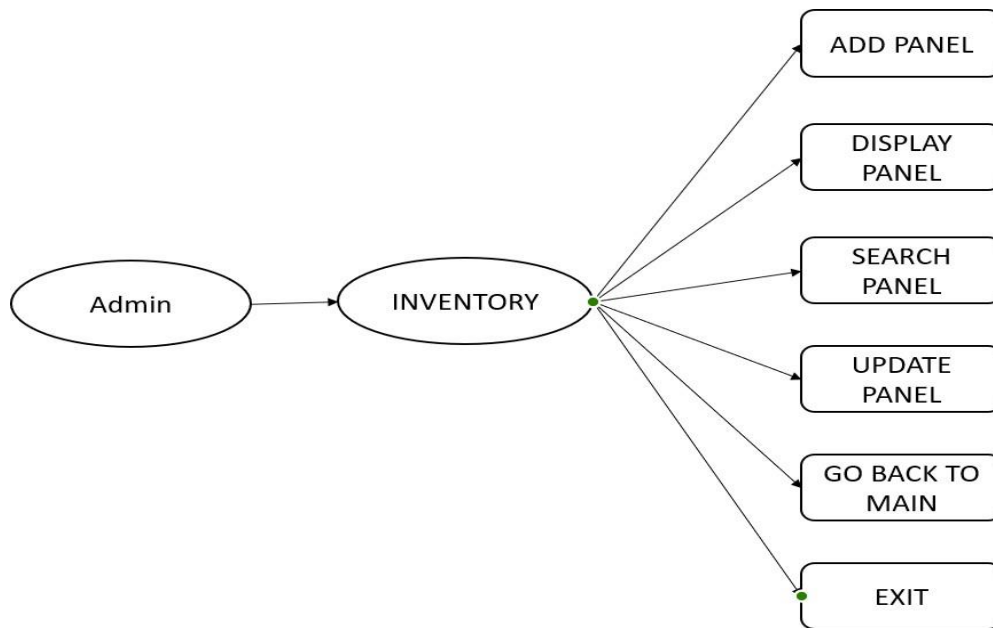


Fig. 3.4.1: Inventory's Panel Flow Diagram

3.5: Report Module

In the Report module, the user is prompted with options; Fill the details, go to the home page, exit program. Figure 3.5.1 shows the Report panel flow diagram.

Fill the details: User can enter the fill in all the details and generate the report.

Home page: User can go to the home page upon selecting this option.

Exit: User can exit the program upon selecting this option.

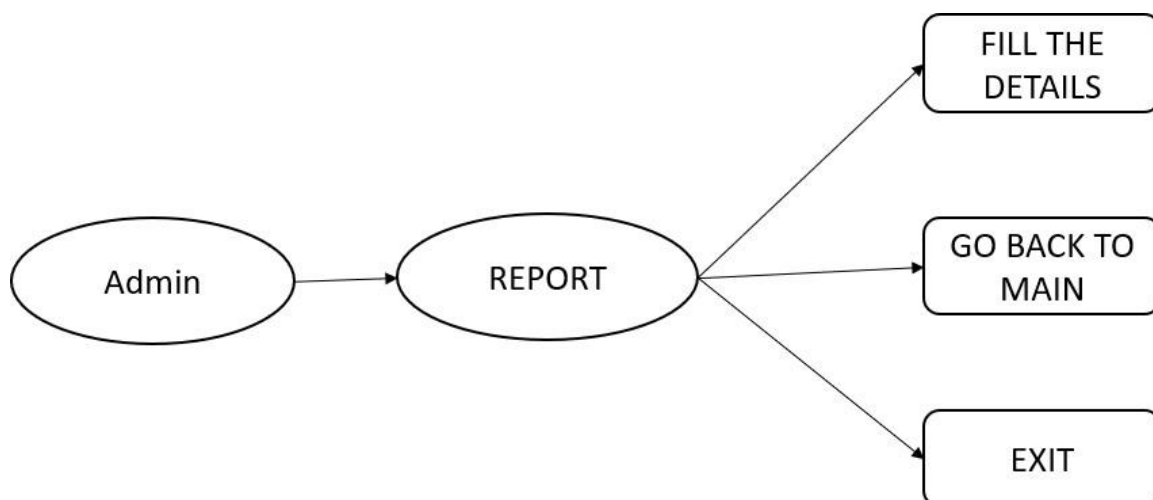


Fig. 3.5.1: Report Panel Flow Diagram

Chapter 4

SYSTEM DESIGN

4.1 UI Logic Interface/Interaction Details

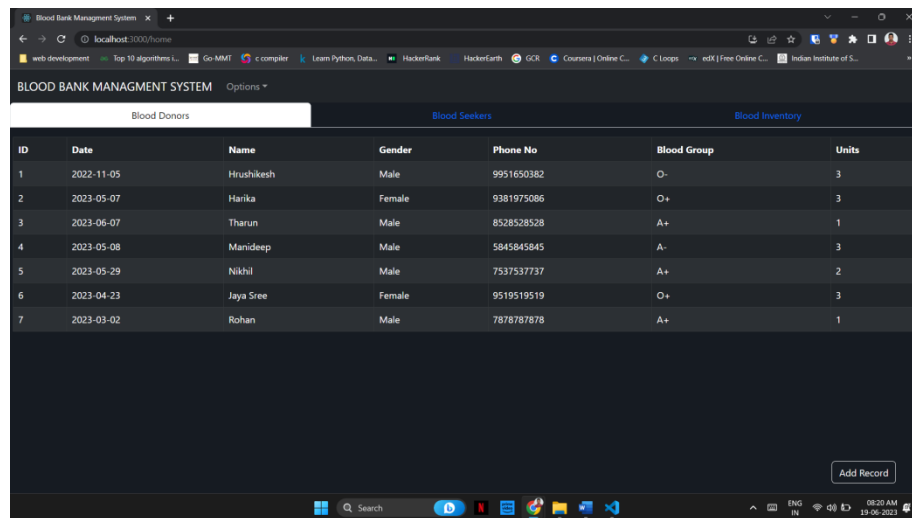


Fig 4.1.1.: Home Page of web application

Fig 4.1 shows the home page of the hospital management System. From here you can navigate to BloodDonors', Blood Seeker', Repot Generation', or Blood Inventory' Tab.

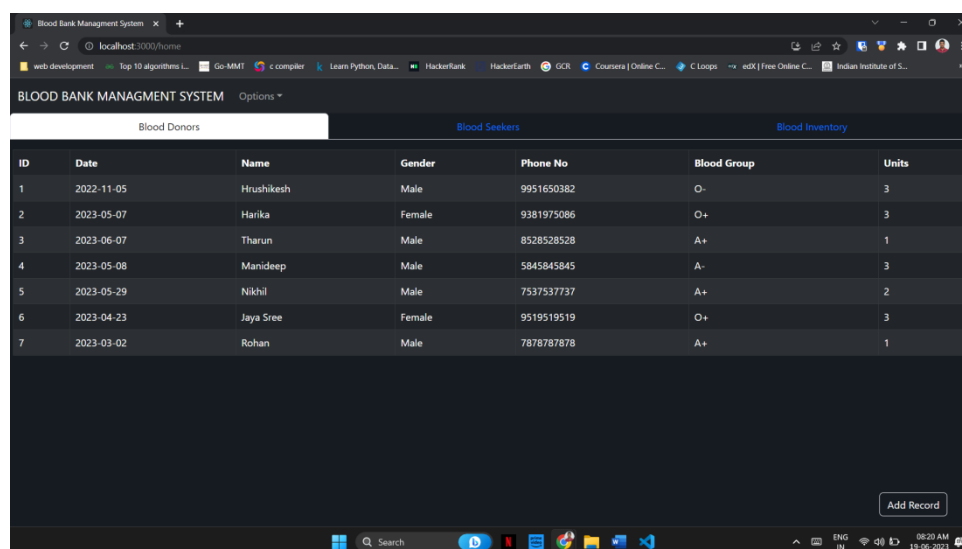
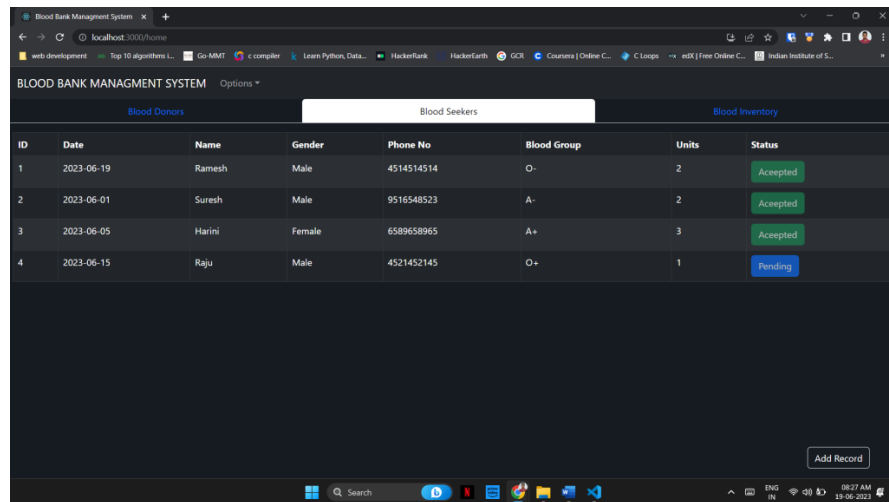


Fig 4.1.2: Donors Tab

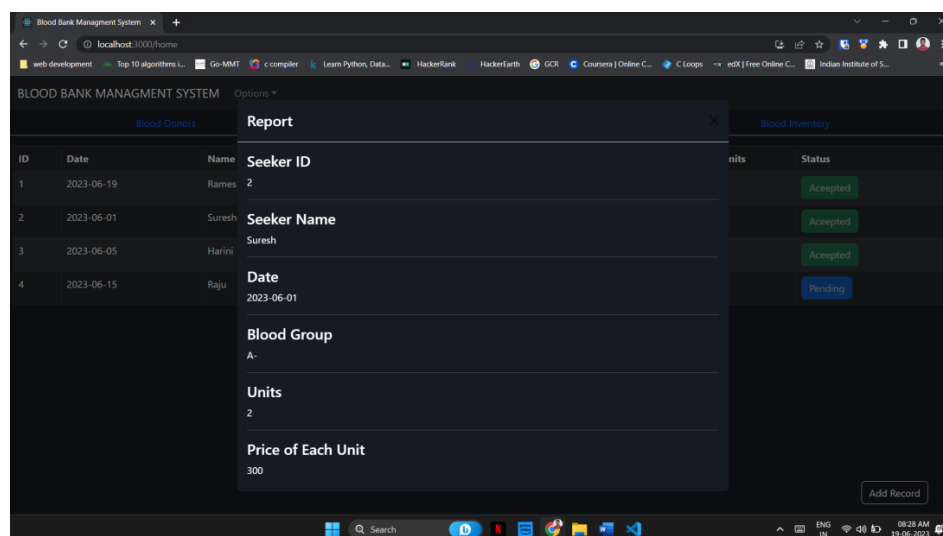
Fig 4.2 shows the Donor tab of the Blood bank management System. From here you can navigate to search adonor, add or to update a donor record. Or to list out all the donor records that were present.



ID	Date	Name	Gender	Phone No	Blood Group	Units	Status
1	2023-06-19	Ramesh	Male	4514514514	O-	2	Accepted
2	2023-06-01	Suresh	Male	9516548523	A-	2	Accepted
3	2023-06-05	Harini	Female	6589658965	A+	3	Accepted
4	2023-06-15	Raju	Male	4521452145	O+	1	Pending

Fig 4.1.3: Seeker Menu panel

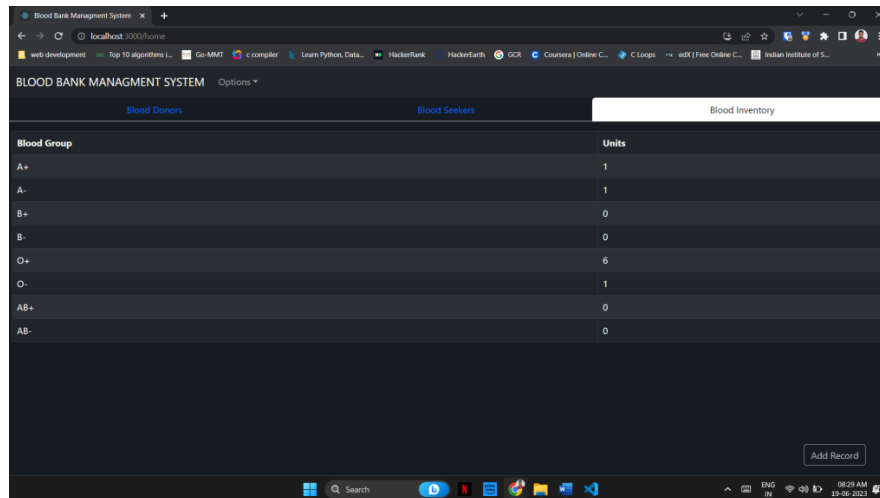
Fig 4.3 shows the Seeker tab of the Blood bank management System. From here you can navigate to search a Seeker, add or to update a seeker record. Or to list out all the seeker records that were present.



ID	Date	Name	Status
1	2023-06-19	Ramesh	Accepted
2	2023-06-01	Suresh	Accepted
3	2023-06-05	Harini	Accepted
4	2023-06-15	Raju	Pending

Fig 4.1.4: Report Menu panel

Fig 4.4 shows the Report's modal of the Blood bank management System. Here you can see all the bill details and can generate the report.



The screenshot shows a web application titled "Blood Bank Management System" with three tabs: "Blood Donors", "Blood Seekers", and "Blood Inventory". The "Blood Inventory" tab is active, displaying a table with two columns: "Blood Group" and "Units". The table contains the following data:

Blood Group	Units
A+	1
A-	1
B+	0
B-	0
O+	6
O-	1
AB+	0
AB-	0

At the bottom right of the table, there is an "Add Record" button. The browser's address bar shows "localhost:3000/home".

Fig 4.1.5: Blood Inventory Tab

Fig 4.5 shows the Blood inventory tab of the Blood bank management System. From here you can navigate to search a specific Blood group, add or to update a Blood quantity record. Or to list out stock of the Blood that were present.

Chapter 5

IMPLEMENTATION

5.1 Description of Frameworks Used

5.1.1 React:

React is a popular JavaScript library for building user interfaces. It allows developers to create reusable UI components and efficiently manage application state. React uses a virtual DOM to efficiently update and render UI components. It follows a component-based architecture, where UI elements are encapsulated into modular components that can be composed to build complex interfaces. React also promotes a declarative programming style, allowing developers to describe the desired UI state, and React takes care of updating the DOM accordingly. React is known for its performance optimizations and efficient rendering, making it a popular choice for building interactive and dynamic web applications.

5.1.2 Flask:

Flask is a lightweight and flexible web framework written in Python. It follows the "micro" framework philosophy, providing a minimalistic set of tools and features. Flask is designed to be simple and easy to use, allowing developers to quickly build web applications. It provides basic functionalities, such as routing, request handling, and template rendering. Flask also integrates well with other libraries and tools, giving developers the flexibility to choose the components they need for their specific project requirements. Flask does not enforce a specific project structure or impose unnecessary dependencies, allowing developers to have greater control over their application's design and architecture. It is often used for smaller to medium-sized projects or when a lightweight framework is preferred.

By combining React and Flask in your project, you leverage the strengths of both frameworks. React helps in building a dynamic and interactive user interface, while Flask provides a lightweight and flexible backend framework for handling server-side operations and APIs. This combination allows you to create a responsive and scalable web application with a modern UI and efficient server-side functionality.

5.2 Description of Integrated Development Environment

VS Code: Visual Studio Code is a popular and versatile Integrated Development Environment (IDE) that provides a range of powerful features for web development projects.

1. **Lightweight and Cross-Platform:** VS Code is known for its lightweight nature, making it fast and responsive even on lower-spec machines. It is available for Windows, macOS, and Linux, allowing developers to work seamlessly across different operating systems.
2. **Extensive Language Support:** VS Code offers excellent language support for a wide range of programming languages, including JavaScript (used in React), Python (used in Flask), HTML, CSS, and more. It provides syntax highlighting, code completion, and linting, making it easier to write clean and error-free code.
3. **Integrated Terminal:** VS Code includes a built-in terminal that allows developers to execute commands, run scripts, and interact with the command-line interface without switching to external terminal applications. This feature is particularly useful for running Flask servers, managing Git operations, and running build scripts.
4. **Debugging Capabilities:** VS Code offers robust debugging capabilities, allowing developers to set breakpoints, step through code execution, inspect variables, and analyze the program flow. This feature is valuable during development, as it helps identify and resolve issues efficiently.
5. **IntelliSense and Code Refactoring:** VS Code provides intelligent code suggestions and completion through its IntelliSense feature. It helps developers write code faster and with fewer errors. Additionally, VS Code supports code refactoring, allowing for efficient restructuring and reorganization of code.
6. **Live Share Collaboration:** VS Code includes a powerful Live Share extension that enables real-time collaboration. It allows multiple developers to work on the same project simultaneously, sharing code, terminal sessions, and debugging sessions. This feature fosters effective teamwork and facilitates pair programming or remote collaboration.

Chapter 6

Testing

6.1 Component Test

Main Panel

The Main Panel is what the user first views when they open the application. The functions available in it are shown in Table 6.1.1

Table 6.1.1: Blood Donor's Module Tests

TEST UNIT	TEST CASE	RESULT
Blood Donor Panel	Blood Donors option is selected	The program shows another menu under Donor panel
Blood Seekers Panel	Blood Seeker option is selected	The program shows another menu under Seeker panel
Report Panel	Report option is selected	The program shows another menu under Report panel
Blood Inventory Panel	Blood Inventory option is selected	The program shows another menu under Inventory panel

Blood Donor's Module

The Blood donors module is the tab where admin can get the list of all the donors. The below table 6.1.2 shows the functions of Blood Donor's Module.

Table 6.1.2: Blood Donor's Module Tests

TEST UNIT	TEST CASE	RESULT
Add Donor	Adding a Donor Record	System Inserts the Record in the File. Errors, if any, aborts the operation.
Update Donor	Updating a Donor Record	The system finds the Record matching the given Name and updates the record. If not found throws Record Not Found Error.
Display Donor	Displaying the Donor Record	Displays the Records of all Donors.
Search Donor	Searching a Donor Record	The system Searches the Record matching the given Name. If not found throws Record Not Found Error.
Go to home Page	Go to home Page	On selection, takes the user back to the main page.

Blood Seeker's Module

The Blood seekers module is the tab where admin can get the list of all the seekers. The below table 6.1.3 shows the functions of Blood Seeker's Module

Table 6.1.3: Blood Seeker's Module Tests

TEST UNIT	TEST CASE	RESULT
Add Seeker	Adding a Seeker Record	System Inserts the Record in the File. Errors, if any, aborts the operation.
Display Seeker	Displaying the Seeker Record	Displays the Records of all Seekers.
Update Seeker	Updating a Seeker Record	The system finds the Record matching the given Name and updates the record. If not found throws Record Not Found Error
Search Seeker	Searching a Seeker Record	The system Searches the Record matching the given Name. If not found throws Record Not Found Error.
Go to home Page	Go back to Main panel	On selection, takes the user back to the main page.
Back to Main Panel	Return back to the main page.	Successfully Returns back to main page

Blood Inventory Module

The Blood Inventory module is the tab where admin can get the details of blood available in inventory. The below table 6.1.4 shows the functions of Blood Inventory's Module

Table 6.1.4: Blood Inventory's Module Tests

TEST UNIT	TEST CASE	RESULT
Add Seeker	Decrease blood availability in inventory	System Inserts the Record in the File. Errors, if any, aborts the operation.
Add Donor	Increase blood availability	System Inserts the Record in the File. Errors, if any, aborts the operation..
Go to home Page	Go back to Main panel	On selection, takes the user back to the main page.
Back to Main Panel	Return back to the main page.	Successfully Returns back to main page

6.2 System Testing

Table 6.2.1: Complete System Tests

TEST UNIT	TEST CASE	RESULT
Inserting New Records	Choose add a new record	Creates a new record with the details entered by the user and stores the same
View all Records	Choose view all records	Opens file with all records and displays the same on the screen
View a Specific Record	Choose a specific record with a key	Opens file containing the record and searches the file for the key entered and displays the same
Update a Specific Record	Choose Update a record	Open file containing the record and Update the record
Go back to Home Page	Choose Go back to Home page	Returns back to the Home Page with the menu
Exit	Choose Exit	Closes the Program

Chapter 7

INTERPRETATION OF RESULTS

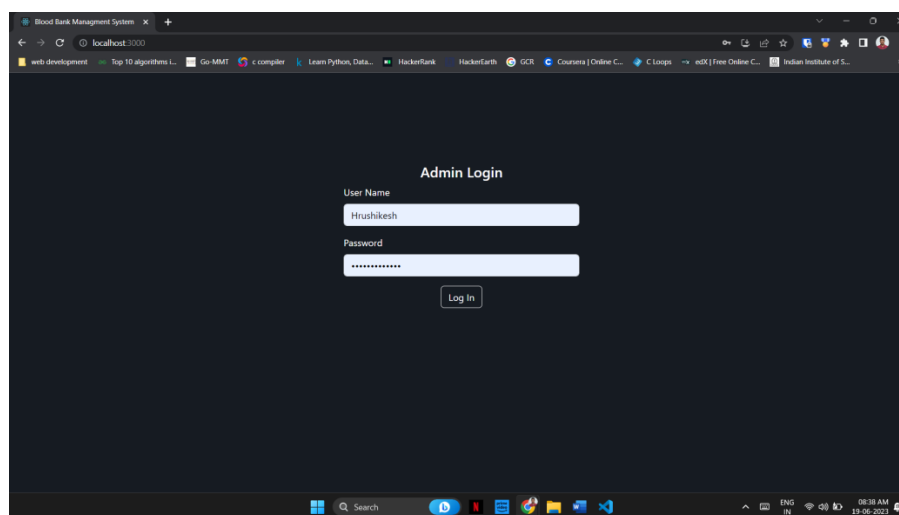


Fig 7.1: Login page of Blood Bank Management System

Fig 7.1 shows the login page of Blood Bank Management System. Here Admin and other users can use their credentials to log in.

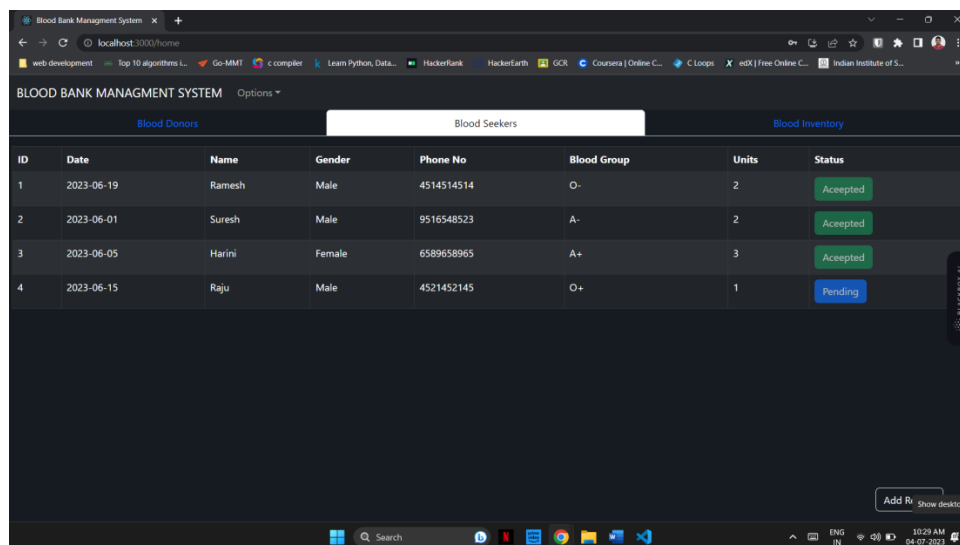
The screenshot shows the 'Blood Bank Management System' home page. The 'Blood Donors' tab is selected, displaying a table of donor information. The table has 7 columns: ID, Date, Name, Gender, Phone No, Blood Group, and Units. The data is as follows:

ID	Date	Name	Gender	Phone No	Blood Group	Units
1	2022-11-05	Hrushikesh	Male	9951650382	O-	3
2	2023-05-07	Harika	Female	9381975086	O+	3
3	2023-06-07	Tharun	Male	8538538538	A+	1
4	2023-05-08	Manideep	Male	5845845845	A-	3
5	2023-05-29	Nikhil	Male	7537537737	A+	2
6	2023-04-23	Jaya Sree	Female	9519519519	O+	3
7	2023-03-02	Rohan	Male	7878787878	A+	1

An 'Add Record' button is located at the bottom right of the table. The browser's taskbar at the bottom shows the system clock indicating 08:20 AM on 19-06-2023.

Fig 7.2: Home Page and Donor's Tab

Fig 7.2 show the home page of the Blood bank management system here users can see the details of the Blood Donors.

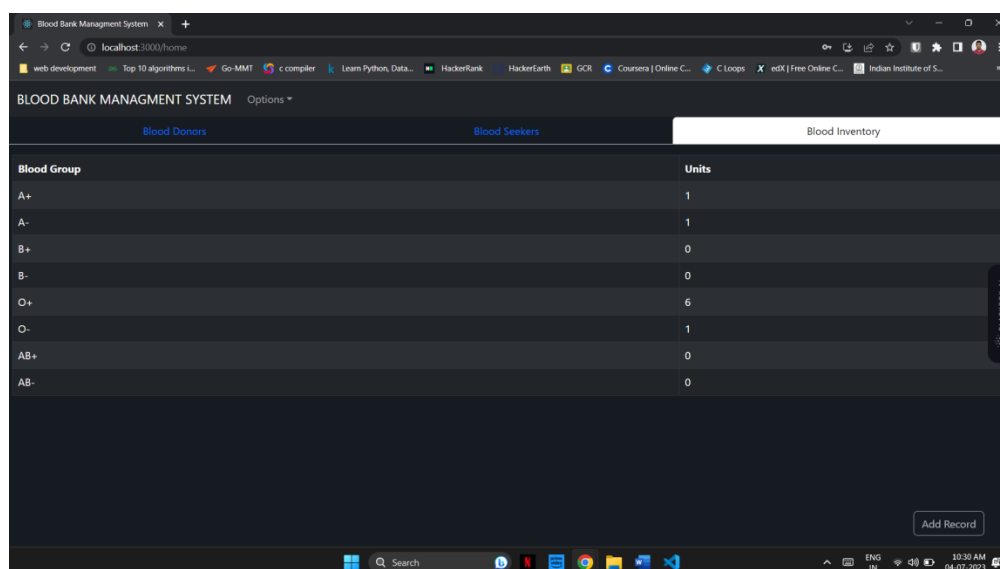


The screenshot shows a web browser displaying the 'Blood Bank Management System' interface. The 'Blood Seekers' tab is active, showing a table with the following data:

ID	Date	Name	Gender	Phone No	Blood Group	Units	Status
1	2023-06-19	Ramesh	Male	4514514514	O-	2	Accepted
2	2023-06-01	Suresh	Male	9516548523	A-	2	Accepted
3	2023-06-05	Harini	Female	6589658965	A+	3	Accepted
4	2023-06-15	Raju	Male	4521452145	O+	1	Pending

Fig 7.3: Seeker's Tab

Fig 7.3 shows the Blood Seekers Tab where user can find the request and the status of the request of the blood seekers.



The screenshot shows the 'Blood Inventory' tab of the 'Blood Bank Management System'. The table displays the following data:

Blood Group	Units
A+	1
A-	1
B+	0
B-	0
O+	6
O-	1
AB+	0
AB-	0

Fig 7.4: Inventory Tab

Fig 7.4 shows the Blood Inventory tab here users can see the type of blood and quantity of the blood available in the inventory.

BLOOD BANK MANAGEMENT SYSTEM

Add Blood Donors

ID	Date
1	2022-11-05
2	2023-05-07
3	2023-06-07
4	2023-05-08
5	2023-05-29
6	2023-04-23
7	2023-03-02

ID: 8

Date: 04-07-2023

Name: Rajesh

Gender: Male

Phone No: 9704888288

Blood Group: Select

Units:

Add Record

Fig 7.5: Add Donor

Fig 7.5 shows the add modal where user can add a new donor into the database. Here we can fill all the details of user.

BLOOD BANK MANAGEMENT SYSTEM

Modify Blood Donors

ID	Date
1	2022-11-05
2	2023-05-07
3	2023-06-07
4	2023-05-08
5	2023-05-29
6	2023-04-23
7	2023-03-02

ID: 1

Date: 05-11-2022

Name: Hrushikesh

Gender: Male

Phone No: 9951650382

Blood Group: O-

Units:

Add Record

Fig 7.6: Modify Donor

Fig 7.6 shows the modify donor tab where user can modify the details of the donor and makes changes to data in files.

BLOOD BANK MANAGEMENT SYSTEM

Options

Blood Donors

Add Blood Seekers

ID	Date	Name
1	2023-06-19	Ramesh
2	2023-06-01	Suresh
3	2023-06-05	Harini
4	2023-06-15	Raju

ID: 5

Date: 14-07-2023

Name: Rakesh

Gender: Male

Phone No: 8978978978

Blood Group: Select

Units:

Add Record

Blood Inventory

Units	Status
	Accepted
	Accepted
	Accepted
	Pending

Fig 7.7: Add Seeker

Fig 7.7 show the add seeker modal where use can fill the details of the seeker and save the data in the files.

BLOOD BANK MANAGEMENT SYSTEM

Options

Blood Donors

Modify Blood Seekers

ID	Date	Name
1	2023-06-19	Ramesh
2	2023-06-01	Suresh
3	2023-06-05	Harini
4	2023-06-15	Raju

ID: 4

Date: 15-06-2023

Name: Raju

Gender: Male

Phone No: 4521452145

Blood Group: O+

Units:

Add Record

Blood Inventory

Units	Status
	Accepted
	Accepted
	Accepted
	Pending

Fig 7.8: Modify Seeker

Fig 7.8 show the modify seeker modal where use can edit the details of the existing seeker and make changes in files.

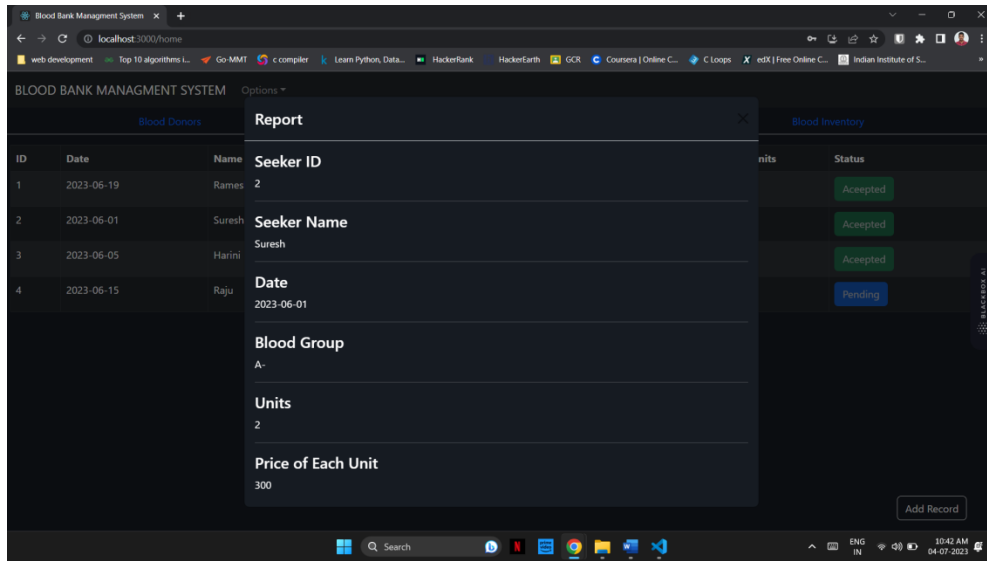


Fig 7.9: Seeker Report

Fig 7.9 show the report of the seeker who's status got accepted. In this we can see the details of the donor and billing details for the seeker.

Conclusion

Blood Bank Management System allows users to perform all functionalities such as adding donor/seeker records, searching donor, seeker, blood records and modifying donor, seeker, blood records and user can buy blood and total revenue is generated depending on the total quantity of blood.

- Once the project is completed it offers users the following functionalities:
- Allows users to easily create a new donor/seeker record and also allow them to modify it.
- Allows users to search for a specific record as well which displays the requested record with the latest details.
- All records with the latest details can be displayed whenever the user needs it.
- Finally, it provides the information about the revenue generated for Blood Bank System.
- Overall, it provides the necessary features for managing blood bank database in the form of files. This application is a Cross Platform and Architecture independent and can run efficiently on Windows, Mac or Linu as long as C++ installation files are present on the system.

Applications

Blood bank management system can be using in various fields to make their complex manual management simpler and easier. Here some fields where we can implement blood bank management system:

- Hospitals
- Blood banks
- Charity
- Trusts

References

- [1] Website:- https://www.researchgate.net/publication/329519303_Blood_Bank_System

- [2] IEEE Intelligent Blood Bank Management System
<https://ieeexplore.ieee.org/abstract/document/8973008>.

- [3] IEE A Computerized Central Blood Bank Management System
<https://ieeexplore.ieee.org/document/8515789>.

- [4] IEEE Hridoy Deb Das and Nurunnahar Smrity, A geo-localized Blood Donor Management System using Mobile Crowdsourcing.