

RV COLLEGE OF ENGINEERING[®],
BENGALURU-560059
(Autonomous Institution Affiliated to VTU, Belagavi)

**DEPARTMENT OF COMPUTER SCIENCE AND
ENGINEERING**



Blood Bank Management System

Mini - Project Report

Submitted by

AKSHAT BANSAL **1RV19CS008**

AMISH CHOPRA **1RV19CS016**

ABHISHEK KUMAR **1RV19CS003**

in partial fulfillment for the requirement of 5th Semester

DBMS Laboratory Mini Project (18CS53)

Under the Guidance of

Dr. Shobha G, Professor, CSE, RVCE

Academic Year 2021- 2022

RV COLLEGE OF ENGINEERING®, BENGALURU - 560059
(Autonomous Institution Affiliated to VTU, Belagavi)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING



CERTIFICATE

Certified that the project work titled '**BLOOD BANK MANAGEMENT SYSTEM**' is carried out by **Amish Chopra(1RV19CS016)**, **Abhishek Kumar(1RV19CS003)**, **Akshat Bansal(1RV19CS008)**, who are bonafide students of RV College of Engineering®, Bengaluru, in partial fulfillment of the curriculum requirement of 5th Semester Database Design Laboratory Mini Project during the academic year **2020-2021**. It is certified that all corrections/suggestions indicated for the internal Assessment have been incorporated in the report deposited in the departmental library. The report has been approved as it satisfies the academic requirements in all respect laboratory mini-project work prescribed by the institution.

Signature of Faculty In-charge

Head of the Department
Dept. of CSE, RVCE

External Examination

Name of Examiners

Signature with date

1

2

2

ACKNOWLEDGEMENT

Any achievement, be it scholastic or otherwise does not depend solely on the individual efforts but on the guidance, encouragement and cooperation of intellectuals, elders and friends. A number of personalities, in their own capacities have helped me in carrying out this project work. I would like to take this opportunity to thank them all.

I deeply express my sincere gratitude to my guide **Internal Guide, Designation**, Department of CSE, RVCE, Bengaluru, for his able guidance, regular source of encouragement and assistance throughout this project

I would like to thank Dr.Ramakanth Kumar P, Head of Department, Computer Science & Engineering, R.V.C.E, Bengaluru, for his valuable suggestions and expert advice.

First and foremost I would like to thank **Dr. Subramanya. K. N**, Principal, R.V.C.E, Bengaluru, for his moral support towards completing my project work.

I thank my Parents, and all the Faculty members of the Department of Computer Science & Engineering for their constant support and encouragement.

Last, but not the least, I would like to thank my peers and friends who provided me with valuable suggestions to improve my project.

Abstract

The project blood bank management system is designed for the blood bank to gather blood from various sources and distribute it to the needy people who have high requirements for it. The software is designed to handle the daily transactions of the blood bank and search the details when required. The software application is designed in such a manner that it can suit the needs of all the blood bank requirements in the course of the future. The operation of the blood bank is still maintained in the manual system, thus it is time consuming, inefficient and tedious to handle all this data manually. Manual entering of data also makes it prone to errors and makes the maintenance of records difficult.

We aim to build an efficient blood bank management system using Python, Django, MySQL DB, HTML and CSS. The proposed system is designed to help the Blood Bank administrator to meet the demand of Blood by sending and/or serving the request for Blood as and when required. The proposed system gives the procedural approach of how to bridge the gap between Recipient, Donor, and Blood Bank.

This application is mainly designed to replace the existing system used in blood banks to keep track of the blood packets and the information of donors and recipients. Mainly it becomes the one stop place for storing of information and donors can register to donate blood. The patients in need of blood can place a request on the app making it a fast and seamless process. Also it is a web based application which can be easily accessed by the people in need.

Table of Contents

Acknowledgement	i
Abstract	ii
Table of Contents	iii
List of Figures	vi
List of Tables	vii
Glossary	
1. Introduction	Page No.
1.1 Objective	7
1.2 Scope	7
2. Software Requirement Specification	8
2.1 Software Requirements	
2.2 Hardware Requirements	
2.3 Functional Requirements	
3. ER Diagram	9
4. Detailed Design	
4.1 Data Flow Diagram	
4.1.1 DFD Level 0	10
4.1.2 DFD Level 1	11
5. Relational Schema and Normalization	12
6. Conclusion & Future Enhancement	14
References	15
Appendix: Snapshots	16

List of Figures

Figure No.	Figure Name	Page.No.
Fig 1	Entity Relationship Diagram	9
Fig 2	Data Flow Diagram-0	10
Fig 3	Data Flow Diagram-1	11
Fig 4	Relational mapping	12
A1	Homepage	15
A2	Patient Login Page	16
A3	Admin Dashboard	16
A4	Blood Request Page	17
A5	Patient Blood Request Form	17
A6	Admin - Donor Details Page	18
A7	Admin - Update Patient Page	18
A8	Logout Page	19

GLOSSARY

DB	:	Database
DFD	:	Data Flow Diagram
ER	:	Entity Relationship
NF	:	Normal Form
SRS	:	Software Requirement Specification
SQL	:	Structured Query Language

Chapter 1

Introduction

1.1 Objective:

The basic building aim is to provide blood donation services to the city. This project aims at maintaining all the information pertaining to blood donors, different blood groups available in blood banks and help them manage in a better way.

Also, the project aims to provide transparency in this field, make the process of obtaining blood from a blood bank hassle-free and corruption-free and make the system of blood bank management effective.

The objectives of this system includes the following:

- To ease the process of blood donation and reception.
- To improve the existing system.
- To develop a scalable system.
- To be highly available

1.2 Scope:

The aim of this software development project is to create a new web application called Blood Bank Management system

The proposed system (Blood Bank Management System) is designed to help the Blood Bank administrator to meet the demand of Blood by sending and/or serving the request for Blood as and when required. The scope of our project is:

- Ensure that all the functionalities of a manual blood bank are covered .
- Make sure the program is simple and easy to use

Chapter 2

Software Requirement Specification

A software requirements specification (SRS) is a description of a software system to be developed. The software requirements specification lays out functional and non-functional requirements, and it may include a set of use cases that describe user interactions that the software must provide to the user for perfect interaction. Software requirements specification is a rigorous assessment of requirements before the more specific system design stages, and its goal is to reduce later redesign. It should also provide a realistic basis for estimating product costs, risks, and schedules.

2.1 Hardware Requirements

- Processor: Intel core i5 or any other
- RAM: minimum 1GB
- Hard Disk: 60GB or above

2.2 Software Requirements

- Software: Python 3.9 or later
- Database: MySQLdb – 8.0.26
- Supported Browsers: Google Chrome/ Firefox/ Safari
- Editor: Visual Studio Code or any other
- Framework: Django 3.0.5
- Operating System: macOS, Windows, or Linux

2.3 Functional Requirements

1. Register a new user and collect all information of the user like Name, Blood group etc.
2. Maintain the statistics of the collected blood.
3. Lists the number of times a donor has donated blood.
4. Maintains all the data of the blood bank.
5. An Employee can see the list of all the blood of a specific blood type available.
6. A patient can see if the required blood is available at the blood bank.
7. Patients can contact the blood bank regarding any queries.

Chapter 3

ER Diagram

ER Diagram stands for Entity Relationship Diagram, also known as ERD is a diagram that displays the relationship of entity sets stored in a database. In other words, ER diagrams help to explain the logical structure of databases. ER diagrams are created based on three basic concepts: entities, attributes and relationships.

In our project 5 main entities were identified as follows:

1. BloodDonate
2. Donor
3. Stock
4. Patient
5. BloodRequest.

Their respective attributes are shown in the ER diagram and relationships between different entities are clearly identified and illustrated.

The relationships identified are:

1. registers_at - between Donor and BloodDonate
2. updates - between BloodDonate and Stock
3. requested_by - between Donor and Patient
4. has_availability - between Stock and BloodRequest

The ER diagram is illustrated in *Fig 1* below.

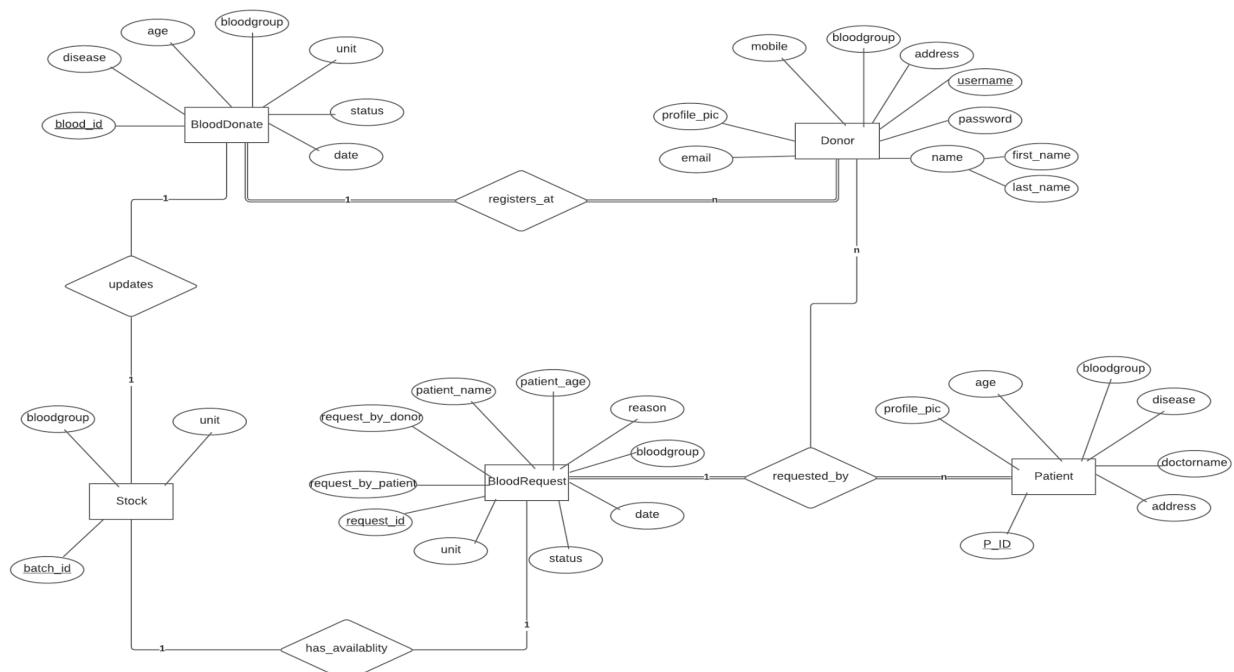


Fig 1 : Entity Relationship Diagram

Chapter 4

Detailed Design

A data flow diagram (DFD) maps out the flow of information for any process or system. They can be used to analyze an existing system or model a new one. The flow of data of a system or a process is represented by DFD. It also gives insight into the inputs and outputs of each entity and the process itself. DFD does not have control flow and no loops or decision rules are present. Specific operations depending on the type of data can be explained by a flowchart. Data Flow Diagrams can be represented in several ways. The DFD belongs to structured-analysis modeling tools. Data Flow diagrams are very popular because they help us to visualize the major steps and data involved in software-system processes.

4.1 DFD Level 0

DFD Level 0 is also called a Context Diagram. It's a basic overview of the whole system or process being analyzed or modeled. It's designed to be an at-a-glance view, showing the system as a single high-level process, with its relationship to external entities.

Three main entities were identified - Employee/admin, User/Patient and Donor. Their data flow diagram with the system is shown in the diagram below.

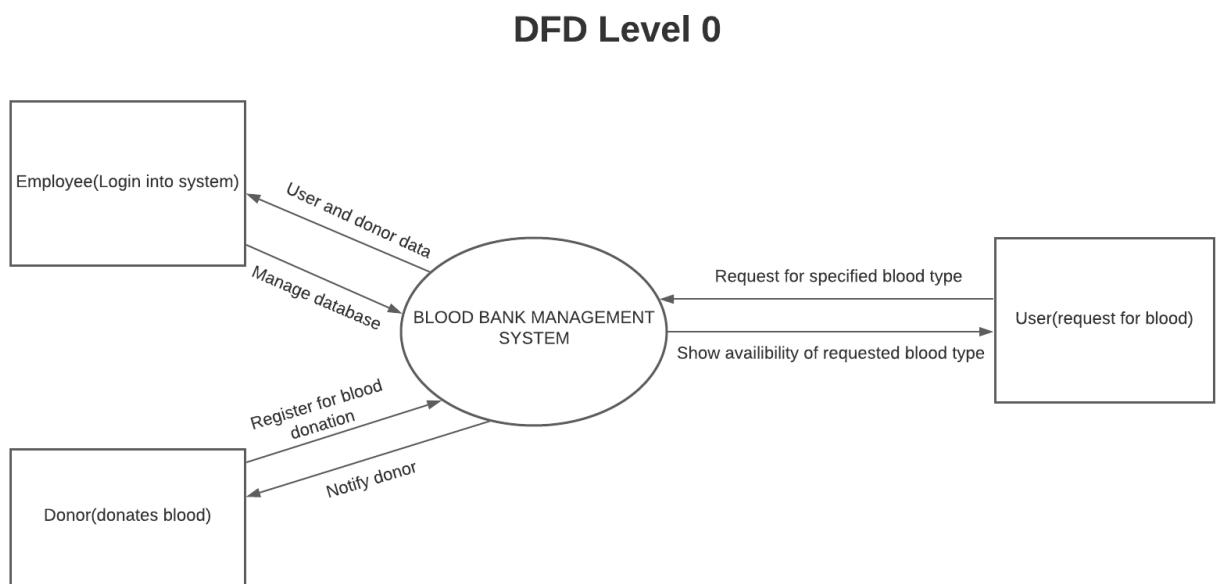


Fig 2 : Data Flow Diagram-0

4.2 DFD Level 1

DFD Level 1 provides a more detailed breakout of pieces of the Context Level Diagram. It highlights the main functions carried out by the system, as we break down the high-level process of the Context Diagram into its subprocesses.

The DFD level 1 is illustrated in *Fig 3* below.

DFD Level 1

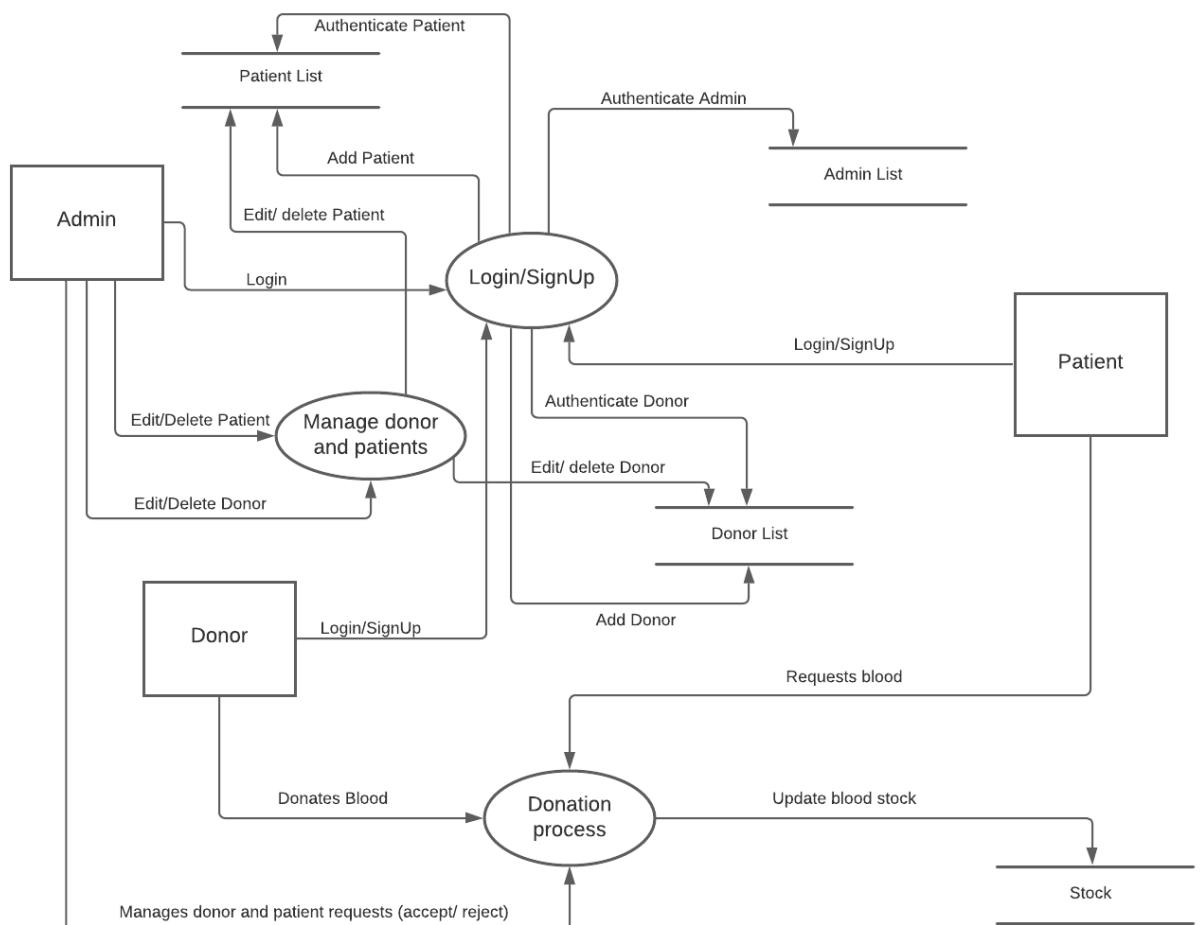


Fig 3 : Data Flow Diagram-I

Chapter 5

Relational Schema and Normalization

5.1 Entity Relationship Mapping

ER diagrams can be mapped to relational schema. This helps in identifying the primary keys, foreign keys etc. which enables us to maintain a proper database state at all times. This helps avoid violations of constraints like referential integrity constraints, null constraints etc.

The ER to relational mapping for the above depicted ER is shown below in figure 4.

Here, each schema is shown with a number of attributes along with their primary keys (underlined attributes). The foreign keys are also depicted here, by arrows.

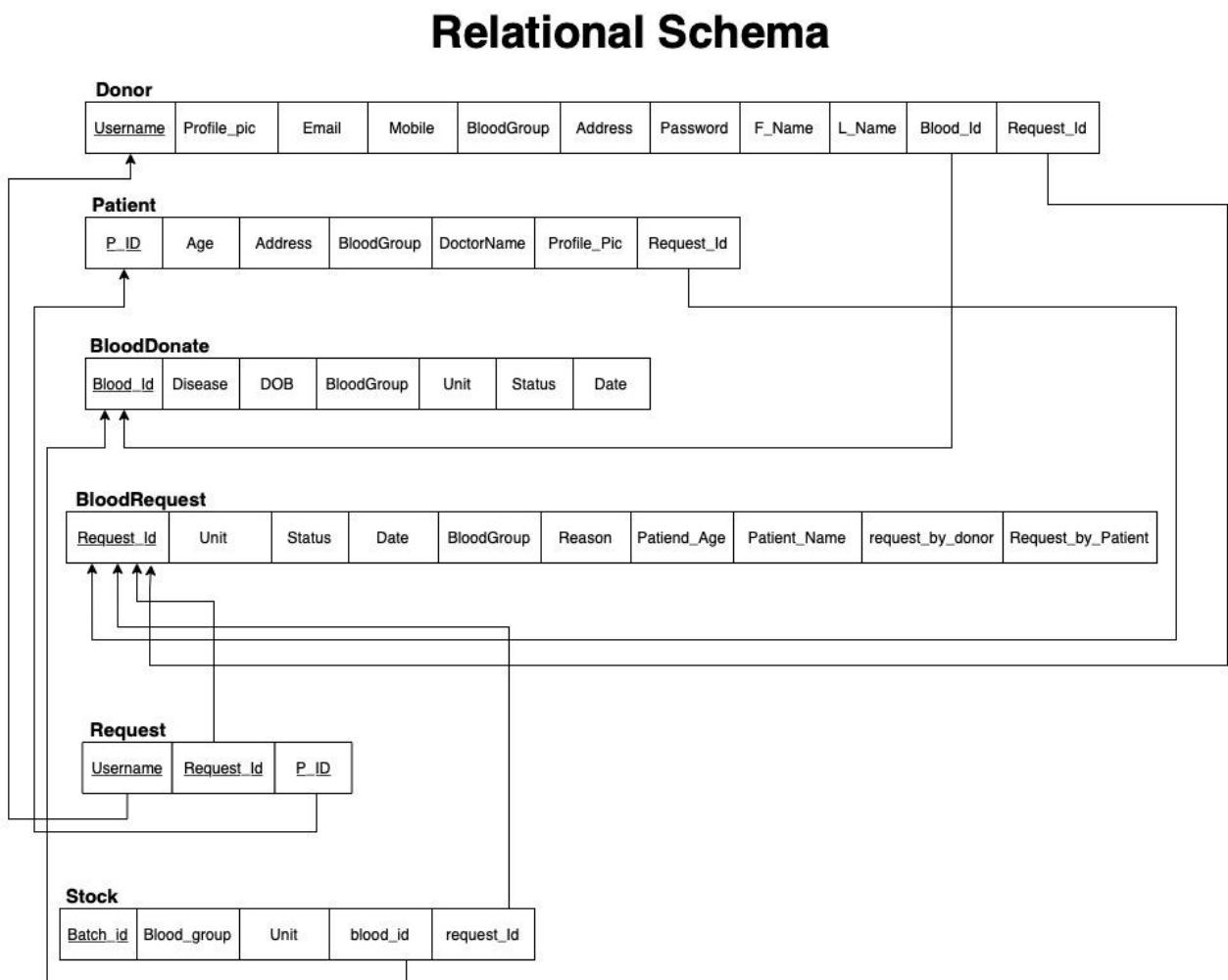


Fig 4 : Relational schema

5.2 Normalization

Normalization is the process of minimizing redundancy from a relation or set of relations. Redundancy in relation may cause insertion, deletion and updation anomalies. Normalization is the process of breaking down a table into smaller tables. So that each table deals with a single theme. There are three different kinds of modifications of anomalies and formulating the first, second and third normal forms (3NF) is considered sufficient for most practical purposes.

5.2.2 First Normal Form - 1NF

The schemas are in 1NF as there are no multivalued attributes.

5.2.3 Second Normal Form - 2NF

The schemas are in 2NF as, in every schema, each and every non prime attribute is fully functionally dependent on the primary key of that schema.

5.2.3 Third Normal Form - 3NF

The schemas are in 3NF form since there exists no transitive dependency. All the functional dependencies of the form $X \rightarrow Y$ exist such that Y is a prime attribute or X is super key.

5.2.4 Boyce–Codd Normal Form - BCNF

The schemas are in BCNF, as, in every functional dependency of the form $X \rightarrow Y$, X is always a super key (minimal super key, here considering primary key).

Chapter 6

Conclusion & Future Enhancement

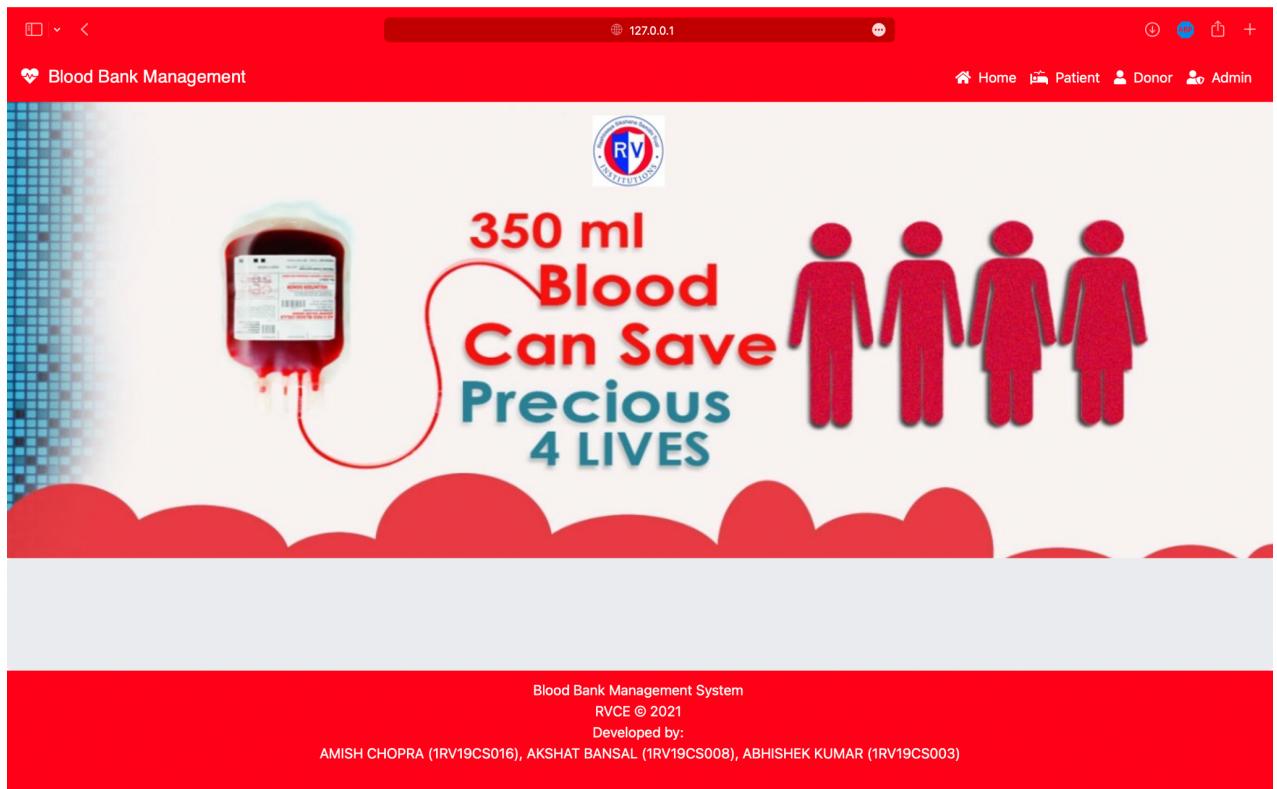
Technology is introducing new innovations day by day, thus reducing the time required to do things. The proposed system can be used to reduce the time required to deliver required blood to the needy in cases of emergency. The web application provides a way of communication and synchronization between the Donor and the blood bank. It also provides them with the facility of communicating with the donors in an emergency. The database is a vital aspect of the system. The database of the needy and the blood bank must be checked for consistency on a regular basis for smooth working of the system.

We have already entered the age of Information Technology, where all the paperwork / manually managed files are about to finish. Now with the help of this user friendly software all the files stored in the computer can be very well formatted. With little more modifications it will become a good software for Blood Bank. The present ‘Blood Bank’ project may be further developed for more complex transactions and to meet the requirements of modern day dynamic System Operation New options and their respective implementation may be done for this purpose. Support of various regional languages for better reach. The size of the database may increase exponentially, so our BBMS is made such that it is scalable and can be deployed on cloud storage systems like Amazon Elastic Compute Cloud (EC2) or Google’s Kubernetes Engine (GKE) after containerizing the application.

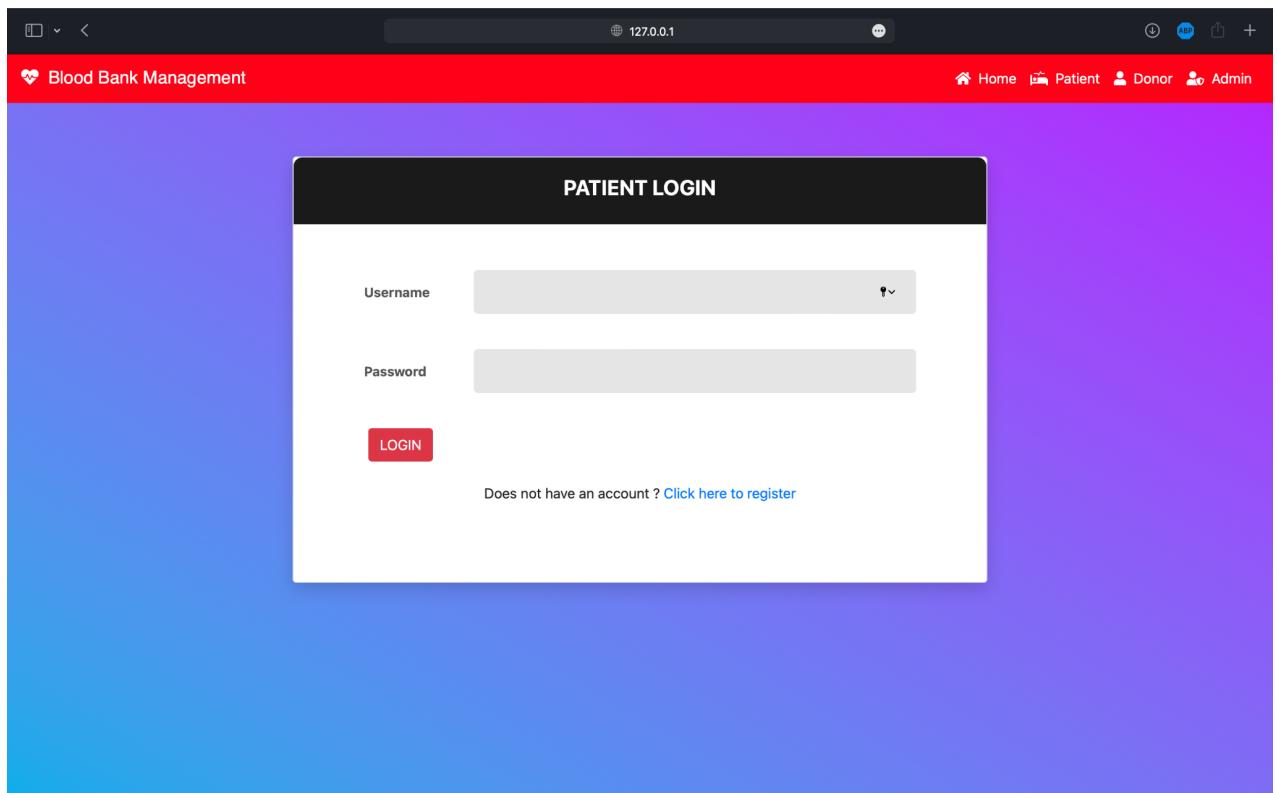
References

- [1] MySQL: The Complete Reference by Vikram Vaswani
- [2] Django for Professionals by William S. Vincent
- [3] Essentials of Blood Banking by Mehdi S.R.
- [4] <https://docs.python.org/3/tutorial/>
- [5] <http://www.mysqltutorial.org/>
- [6] <https://www.javatpoint.com/mysql-tutorial>
- [7] <https://www.coursera.org/learn/python-databases>

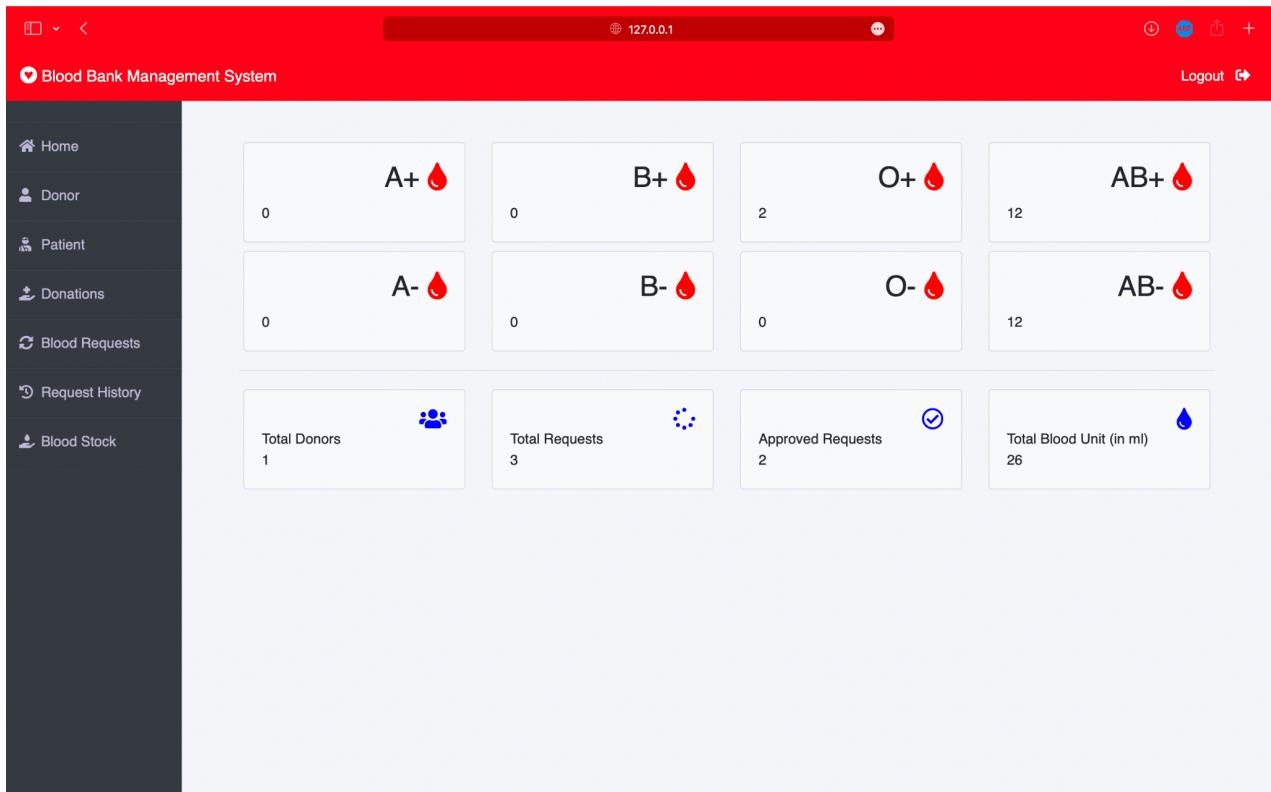
Appendix



A1 - Homepage: Provides Donor, Patient and Admin login links.



A2 - Patient Login Page: Patient enters his username and password to access the application.



A3- Admin dashboard: Administrator view of the dashboard to view the Blood Stock, Patient and Donor details and Blood Requests.

Blood Request History							
Patient Name	Age	Reason	Blood Group	Unit (in ml)	Date	Status	Stock Status
Akshat	22	Test	AB+	5	Dec. 6, 2021	Rejected	0 Unit Deducted From Stock
Akshat	22	Test2	AB+	2	Dec. 6, 2021	Approved	2 Unit Deducted From Stock
Akshat	22	test3	AB+	2	Dec. 7, 2021	Approved	2 Unit Deducted From Stock

A4 - Admin - Blood request page: The view from which the admin is able to see the blood requests made in the past.

The screenshot shows a web-based application titled "Blood Bank Management System". The main interface has a red header bar with the URL "127.0.0.1" and a "Logout" button. On the left, there's a dark sidebar with three options: "Home", "Make Request", and "Request History". The central area features a modal window titled "MAKE BLOOD REQUEST". Inside the modal, there are five input fields: "Patient Name" (text input), "Patient Age" (number input with a dropdown arrow), "Reason" (text input), "Blood Group" (dropdown menu with placeholder "Choose option"), and "Unit (in ml)" (number input with a dropdown arrow, currently set to 0). A large red "REQUEST" button is at the bottom of the modal.

A5- Patient-Blood Request form: The patient is able to request for blood by filling this form.

The screenshot shows a web-based application interface for a blood bank management system. The top navigation bar is red, displaying the URL '127.0.0.1' and various icons. The title 'Blood Bank Management System' is at the top left, and 'Logout' is at the top right. A sidebar on the left contains links: Home, Donor, Patient, Donations, Blood Requests, Request History, and Blood Stock. The main content area is titled 'DONOR DETAILS'. It features a table with one row for Amish Chopra. The columns are Name (Amish Chopra), Profile (small placeholder image), Blood Group (AB+), Address (BDA, Mysore Road, Bangalore), Mobile (9955177900), and Action (blue 'EDIT' button and red 'DELETE' button). The background of the main content area is light grey.

Name	Profile	Blood Group	Address	Mobile	Action
Amish Chopra		AB+	BDA, Mysore Road, Bangalore	9955177900	<button>EDIT</button> <button>DELETE</button>

A6- Admin-donor details page: The admin is able to view, edit and delete the details of the donor.

The screenshot shows a 'UPDATE PATIENT' form. The background of the main content area is purple. The form fields are: First Name (Akshat), Last Name (Bansal), Username (akshat1), Password (empty field), Age (empty field), Blood Group (dropdown menu with 'Choose option'), Disease(if any) (empty field), and Doctor (empty field). The sidebar on the left is identical to the one in the previous screenshot. The top navigation bar shows the user is 'ADMIN'.

A7- Admin-update patient page: The admin is able to edit the details of the patient like Blood Group, diseases, name of doctor etc.



You Have Been Logged Out

Thank you for using our website

Blood Bank Management System

RVCE © 2021

Developed by:

AMISH CHOPRA (1RV19CS016), AKSHAT BANSAL (1RV19CS008), ABHISHEK KUMAR (1RV19CS003)

A8- Logout page: The user is redirected to this page after logging out of his account.