CS-257
Database Information Systems -PROJECT

BLOOD BANK MANAGEMENT SYSTEM

Submitted by:
Jagruthi Patibandla(180001021)
Shravya Ramasahayam(180001052)

Abstract:

In this project, we created one application which is easy to access and user friendly. The database was developed using MySQL. For the user interface, we used PHP and HTML. There are two kinds of people who can use this application - unregister users, registered users and an administrator. A registered user can request for blood and also apply for a donation. The administrator is the only one who can add donations and recipients. An unregistered user can search for blood but cannot request or donate.

Introduction:

A Database is a collection of inter-related data containing information related to an enterprise. Databases are widely used in almost every field. One of that is Blood Bank Management System. Blood banks have massive and frequently used data to store.

Motivation:

The current system that is used by the blood bank is a manual system. With the manual method, there are problems in managing the donors' records. The records of the donor might not be kept safely, and they might be missing of donor's records due to human

error. Besides that, errors might occur when the staff holds more than one record for the same donor.

Also, in case of emergency, when one hospital fails to provide the required quantity of blood of a particular group, it becomes difficult to for that health centre to contact other hospitals for the same.

Without an automated management system, there are also problems in keeping track of the actual amount of every blood type in the blood bank. Besides, there is automatic deletion when the blood in the bank has expired. In the existing manual system, it takes a lot of time.

Aim:

This project aims to help the hospitals to do all such tasks more accurately and faster way. The purpose of the blood bank management system is to simplify and automate the process of searching for blood in case of emergency. Maintain the records of blood donations, recipients, users, login information, details of hospitals collaborated and incharge details in the bank so that users can request in case of emergency.

Requirements:

1. XAMPP
2. MySQL
3. Visual Studio Code

Languages: HTML, PHP, MySQL, CSS

Entity and Relationship Sets:

1. DONATIONS

A blood bank needs to maintain records of its donations so as to track the Donor details, which is the information about people donating blood and also the quantity donated. Each donation should have a unique id and contact details, and the blood group of the donor needs to be stored. Details regarding the donation are also stored like the place (It comes as a drop-down menu.) and date of donation so as to track its expiry. Date of donation is also used to track the most recent donation (if any) to check further eligibility to donate.

a. Donation ID -PRIMARY KEY
b. Email of donor
c. Blood group
d. Quantity of blood donated

e. Place of donation (hospital)
f. Date of donation
g. Availability

2. RECIPIENT LIST

Recipient details contain information regarding patients who use blood through our portal. His contact details need to be stored. Details about his blood group and quantity used are stored. Name of the hospital is used to record the place of blood donation. It comes as a drop-down menu. The date the blood was used is also stored.

a) Recipient ID -PRIMARY KEY
b) Blood group
c) Hospital admitted
d) Quantity of blood used
e) Email id

3. USERS

It contains information about registered users. These users can either donate or request for blood through the portal. His age, weight, date of birth and contact details are stored. This data (age and weight) is used to check if a donor is eligible to donate or not. The address is a composite attribute and includes street no., city and state. Age is a derived attribute derived from the date of birth.

a. Full name
b. Date of Birth
c. Gender
d. Age
e. Weight
f. Blood group
g. mobile number
h. email id -PRIMARY KEY
i. Address: Street number
City
State

4. HOSPITAL

Hospital entity is used to store details of the hospitals that have their partnership with the blood bank. Hospital id is used to uniquely track a hospital. Address of the hospital is a composite attribute and includes street no. , city and state. Phone number of the hospital is stored. Each hospital has an incharge whose ID is also stored.

a. Hospital id - PRIMARY KEY
b. Name of the hospital

c. Address
    Street number
    City
    State
d. Phone number
e. Associated incharge ID

5. INCHARGE

Incharge entity is used to store the details of incharges associated with various hospitals. Incharge ID is used to identify an incharge uniquely. His name and phone number are also stored. Each incharge has a unique userID and password. These login details are used by them to update donations and recipient list.

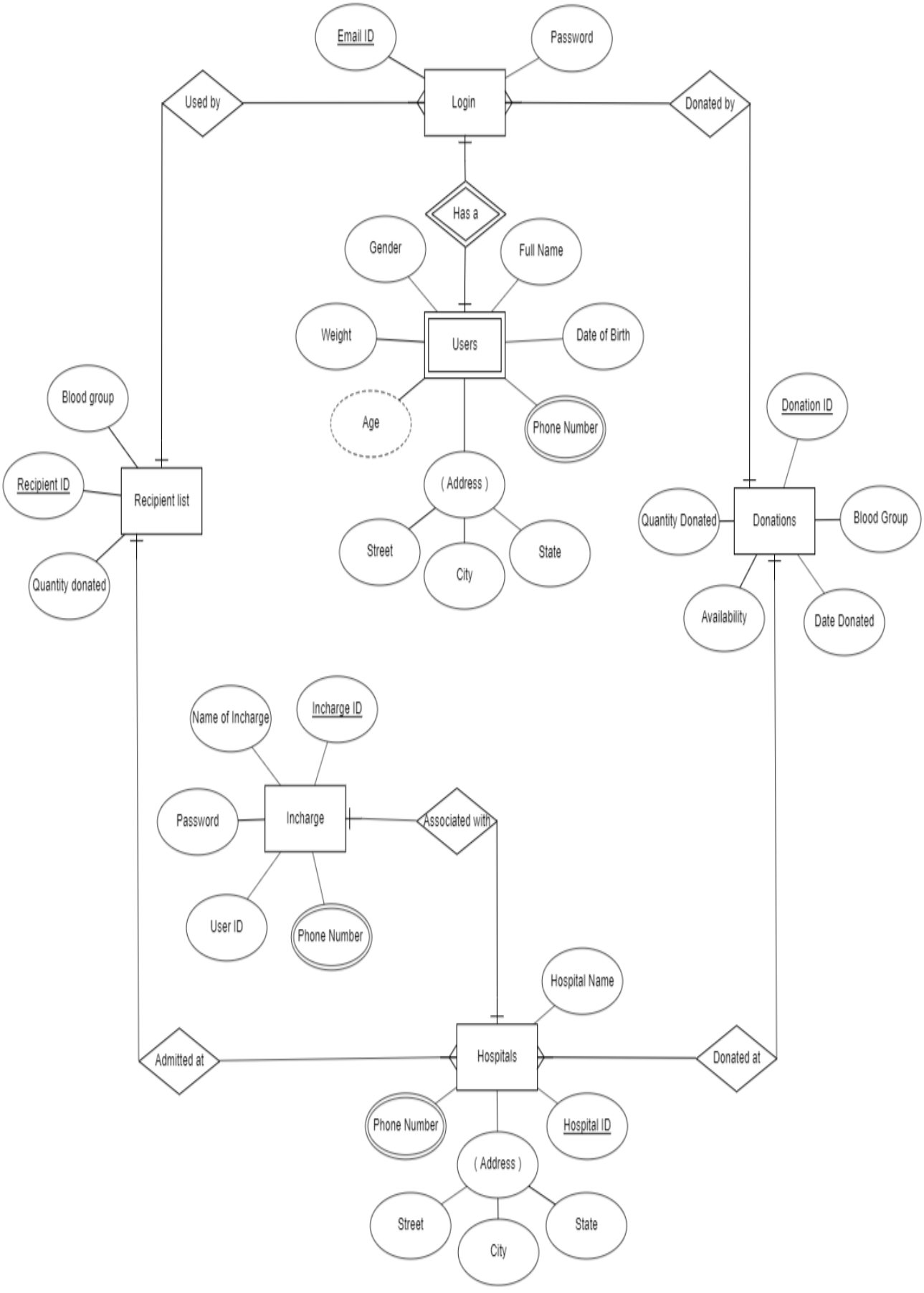
a. Incharge ID
b. Name of incharge
c. Phone number
d. UserID
e. Password

6. LOGIN

It contains emailID and password of a registered user. The login details are used by him to donate or request for blood. A logged-in user can see his previous activities and also there is provision for editing his existing details.

a. emailID
b. Password

ER Diagram:


Email ID
Password
Login
Used by
Donated by
Has a
Gender
Full Name
Weight
Users
Date of Birth
Age
Phone Number
(Address)
Street
City
State
Blood group
Recipient ID
Recipient list
Quantity donated
Donation ID
Quantity Donated
Donations
Blood Group
Availability
Date Donated
Name of Incharge
Incharge ID
Password
Incharge
Associated with
User ID
Phone Number
Hospital Name
Admitted at
Hospitals
Donated at
Phone Number
Hospital ID
(Address)
Street
City
State

ER Into Tables:

1.Users:

```
create table users(
username varchar(50),
dob date,
gender varchar(10),
age varchar(3),
weight varchar(3),
bloodgroup varchar(5),
phoneno varchar(10),
emailID varchar(50),
streetno varchar(30),
city varchar(30),
state varchar(30) default "Telengana",
primary key (donorid),
FOREIGN KEY (emailID) REFERENCES login(email) );
```

2.Donations:

```
create table donations(
donationid int(5) auto_increment,
email varchar(30),
bloodgroup varchar(3),
quanititydonated varchar(5),
hospitaldonated varchar(60),
dateofdonation date,
availability varchar(3) default "Yes",
primary key (donationid));
```

3.Recipient list:

create table recipientlist(

Rid int(5) auto_increment,

emailID varchar(30),

bloodgroup varchar(3),

quantityused varchar(5),

dateused date,

hospital varchar(50),

primary key (Rid));

4.Hospitals:

create table hospital(

hospitalid int,

hospitalname varchar(50),

phoneno varchar(10),

streetno varchar(30),

city varchar(30),

state varchar(30) default 'Telangana',

inchargeid int,

primary key (hospitalid),

FOREIGN KEY (inchargeID**)** REFERENCES inchargeID(incharge))**;**

5.Incharge:

create table incharge

(inchargeid int,

inchargename varchar(50),

phoneno varchar(10),

primary key (inchargeid));

6.Login:

```
create table login(
email varchar(30),
pass varchar(30)
primary key (email),
FOREIGN KEY email REFERENCES emailID(Users));
```

Triggers Used:

1) Age: age is automatically updated in users table whenever a new user registers himself using the date of birth entered
2) Availability: In donations table whenever a donation is added the available quantity is initialised to the amount donated

1) Finding Age with date of birth:

```
DELIMITER |
create TRIGGER agefind before INSERT on users for each row
BEGIN
set new.age=(year(curdate())-year(new.dob))
END
|
DELIMITER;
```

2) Updating quantity available:

```
DELIMITER |
create TRIGGER avail before INSERT on users for each row
BEGIN
set new.availability=new.quantitydonated;
END
|
DELIMITER ;
```

Views Used:

A view is created by performing inner join on donations table, incharge table and hospitals table. The common attribute between donations and hospitals table is the hospital donated. The common attribute between incharge and hospitals table is the incharge ID whenever a user searches or requests for blood, the details of the incharge along with hospital name are displayed.
Create view view_name as select d.availability, d.hospitaldonated, i.inchargename, i.phoneno from hospital as h inner join donations as d on h.hospitalname = d.hospitaldonated inner join incharge as i on h.inchargeid = i.inchargeid where d.bloodgroup = bloodgroup and d.availability > 0

There are 8 views for 8 blood groups:

1)mixabn:

2)mixabp

3)mixop

4)mixon

5)mixap

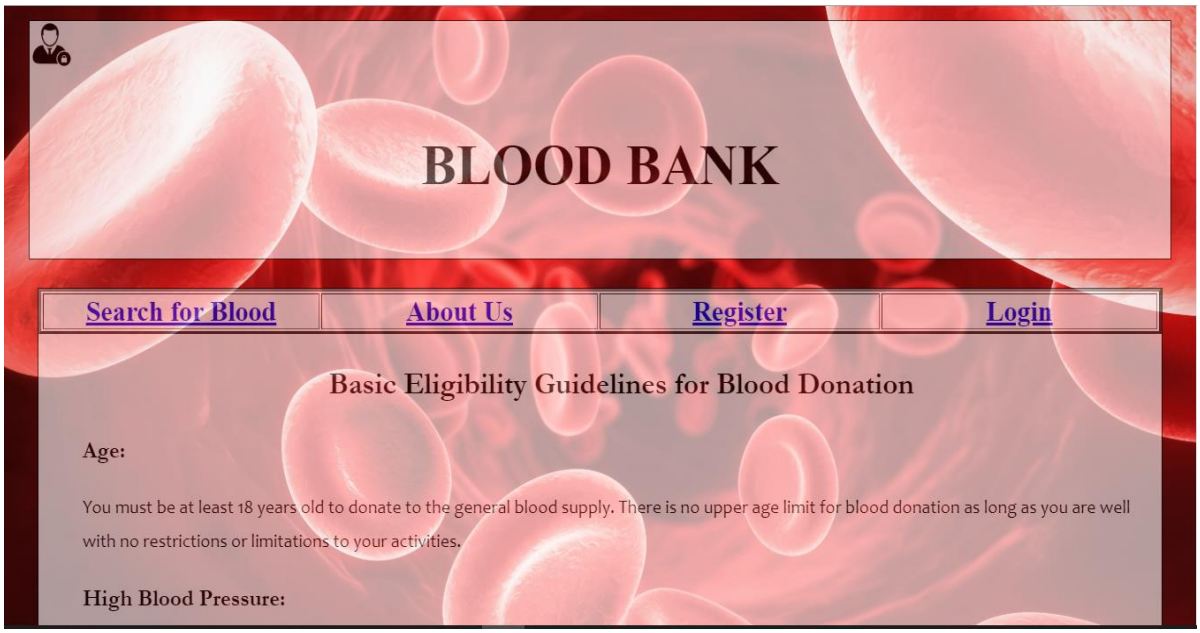
6)mixan

7)mixbp

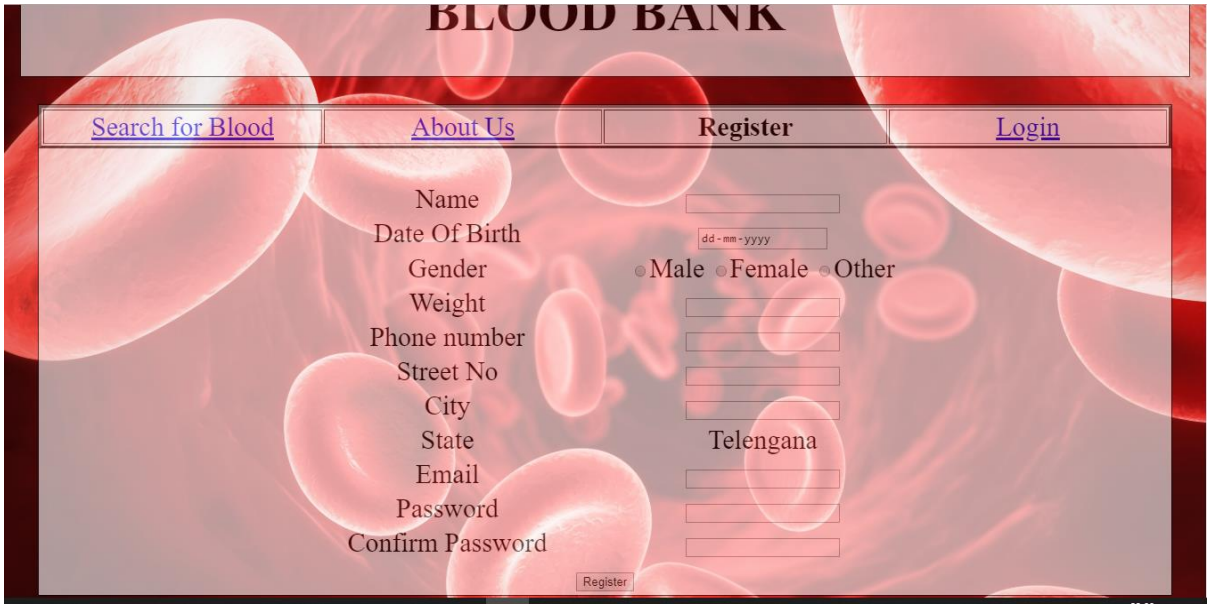
8)mixbn

Screenshots:

- HOME:

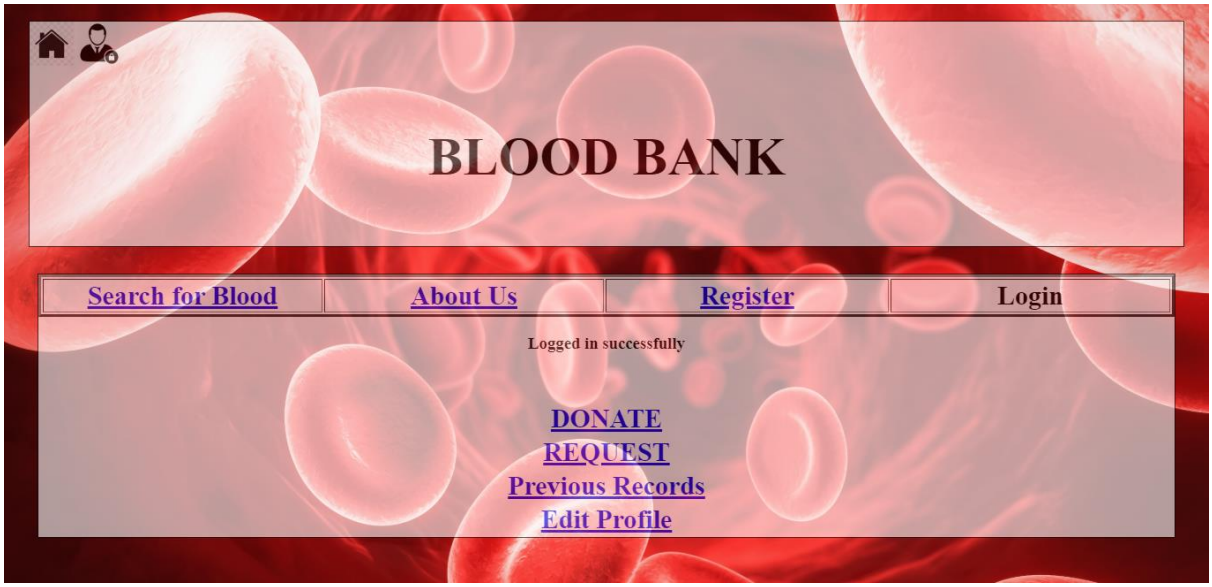


- REGISTER:

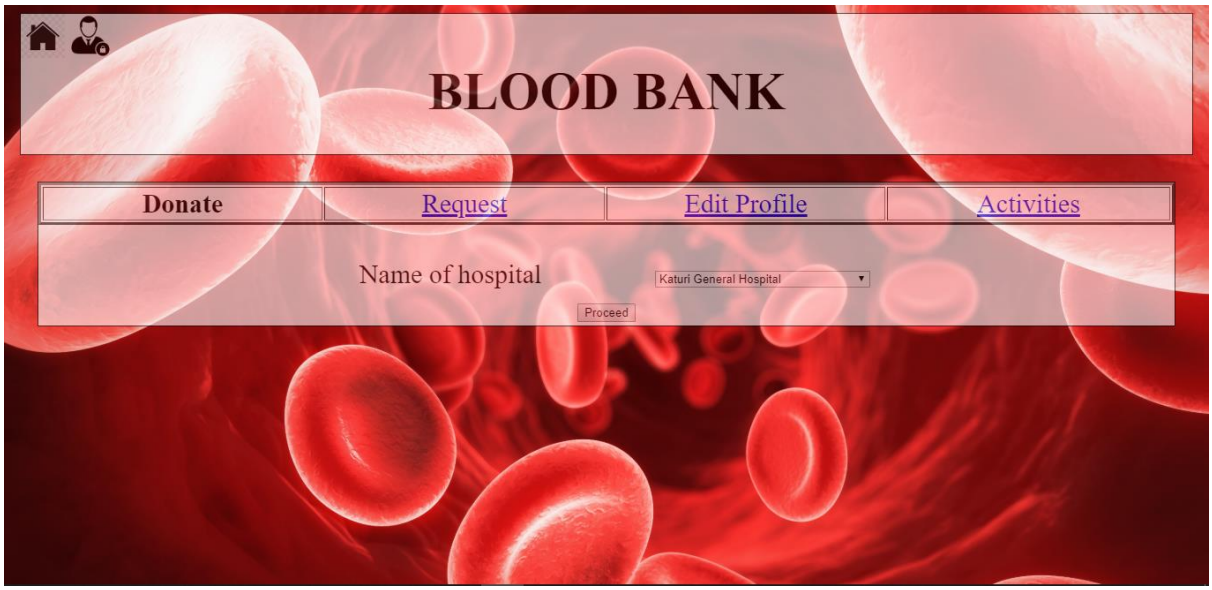


- LOGIN:

- AFTER LOGIN:



- LOGIN-DONATE:



- LOGIN-DONATE:

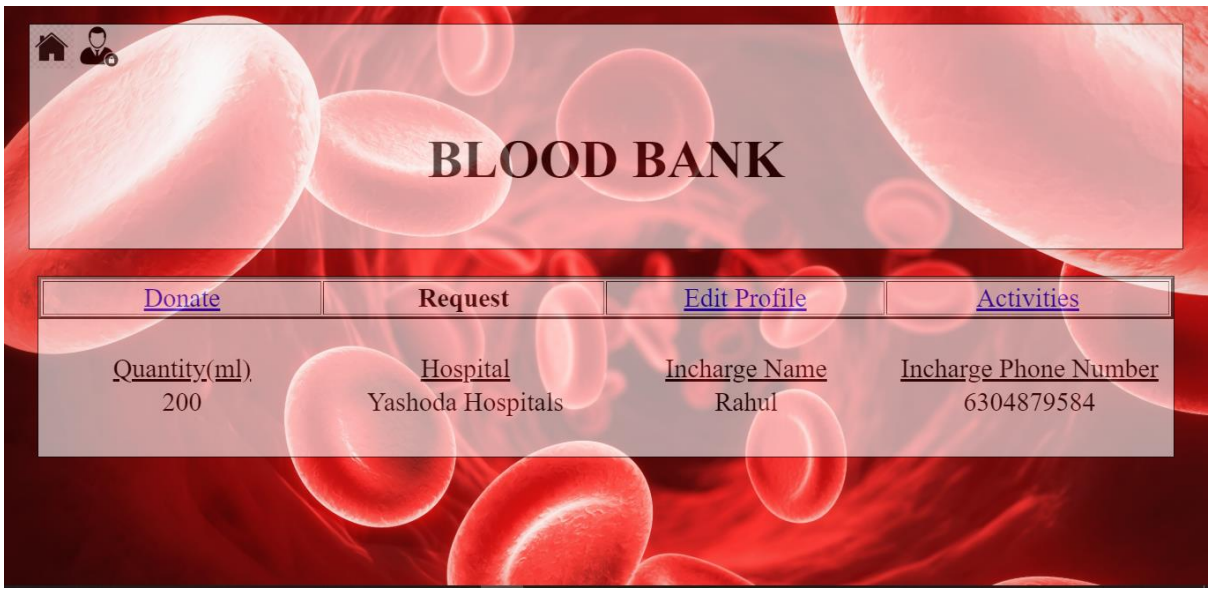
1.

2.

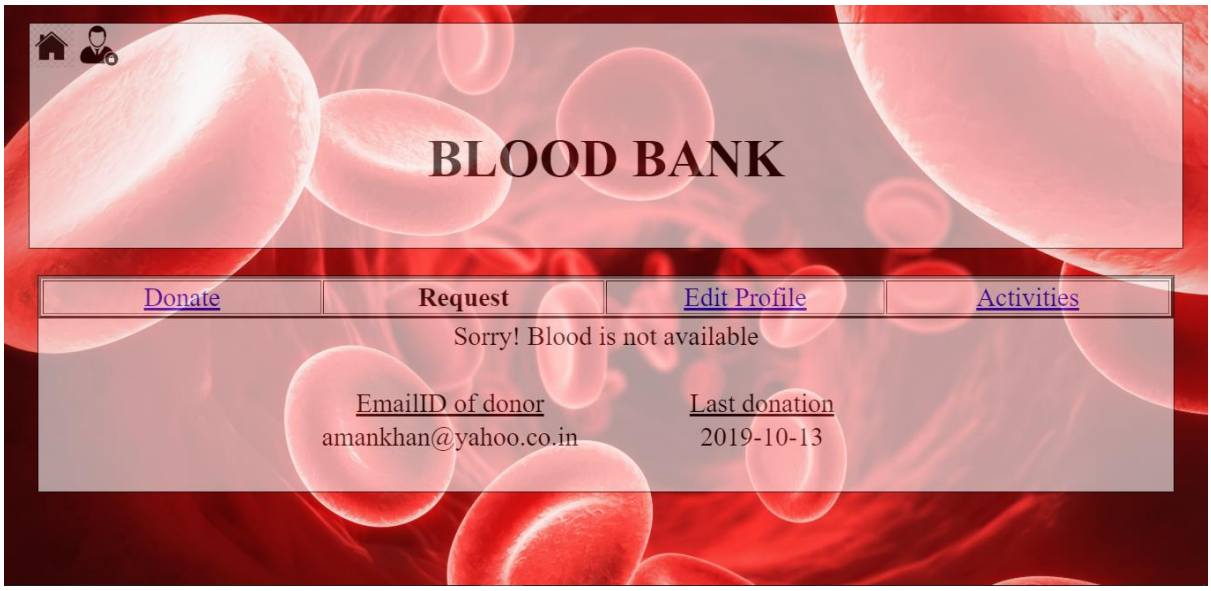
3.

- LOGIN->REQUEST:

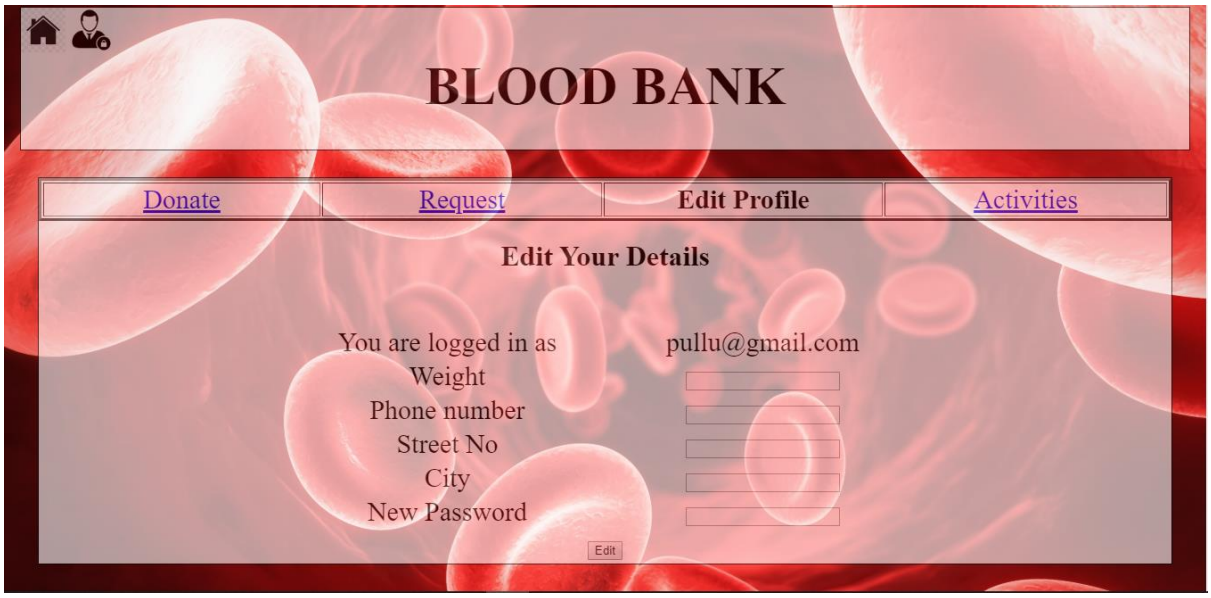
1.



2.



- LOGIN->EDIT PROFILE:

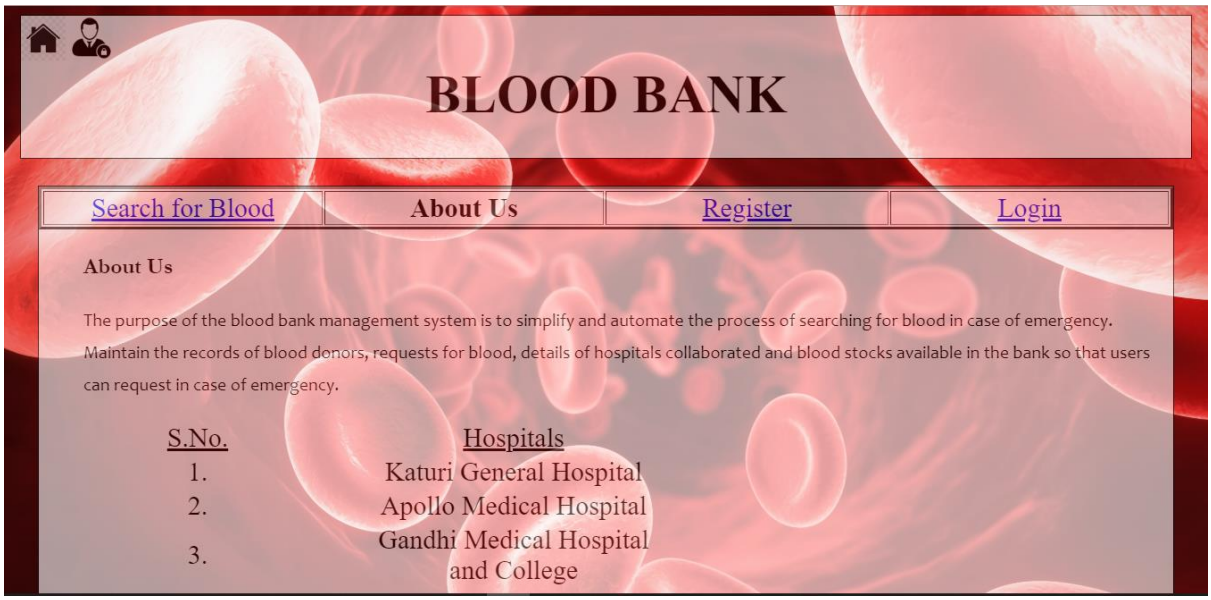


➢ SEARCH FOR BLOOD:

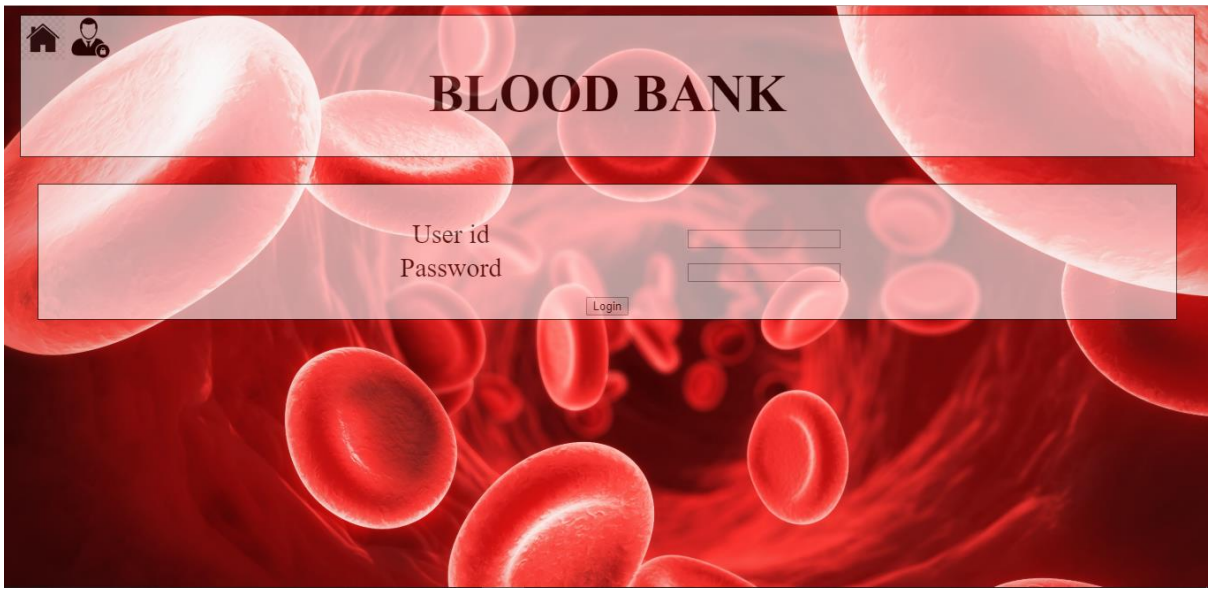
1.

2.BLOOD NOT AVAILABLE IN SEARCH:

- ABOUT US:

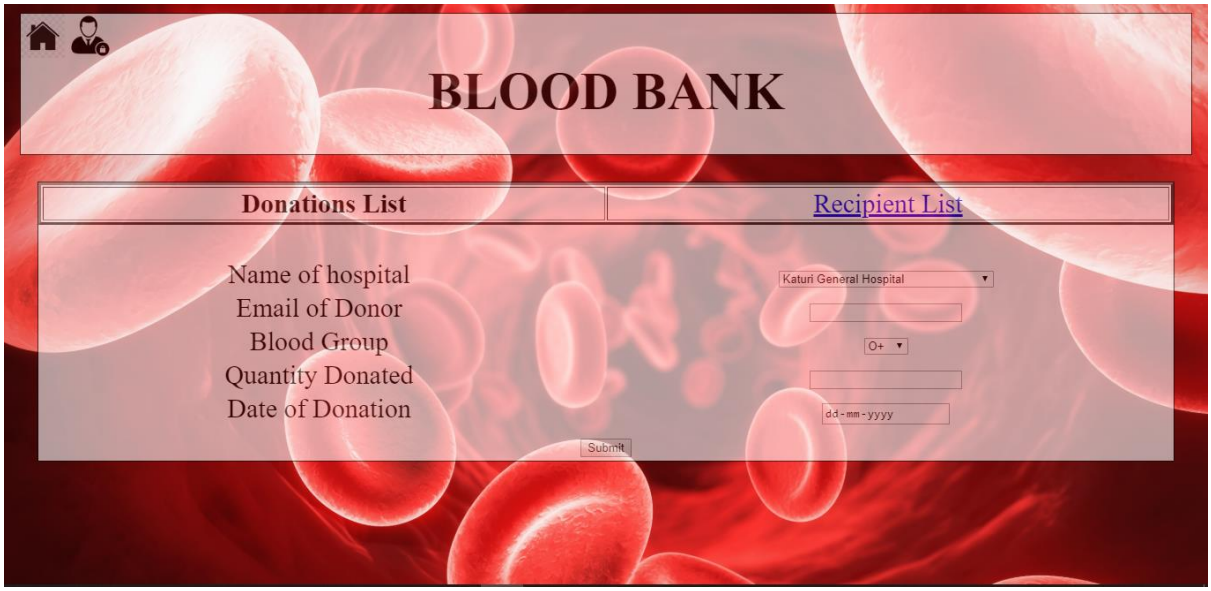


- INCHARGE LOGIN:

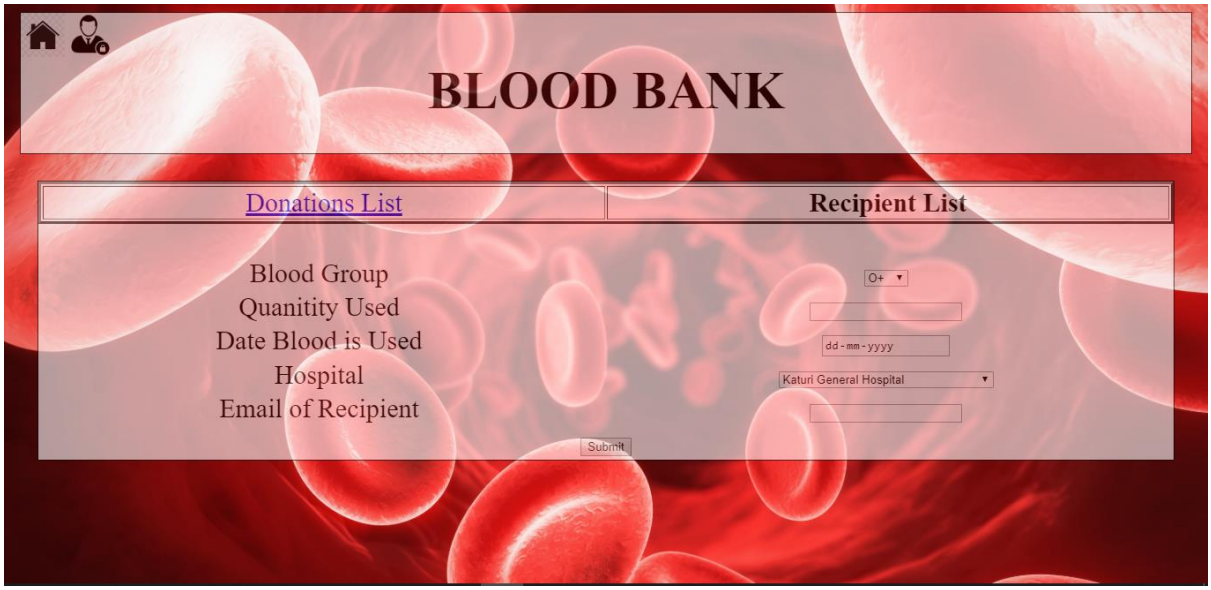


- AFTER LOGGING IN:

- DONATIONS LIST:



- RECIPIENT LIST:



GOALS ACHIEVED:

1) Awareness about blood donation. Do's and Don'ts through the home page.
2) Portal to search for blood availability.
3) Whenever a registered user applies for donating blood, his eligibility for donating blood is checked—for instance, his age, weight criteria along with the very recent donation activity.
4) In case a user is eligible for the donation, he provides contact details of incharge of the hospital where he wishes to donate.
5) If the user wants to request for blood, Availability of the blood stocks is shown. In case the blood is not available, the emailID's of donors of that group are displayed.
6) A logged-in user can view his previous activities and edit his profile (contact details, weight, ansd password).
7) There is a provision for the admin to edit donations and recipients list.