

Blood Donation Database Management System

Prepared by:

- | | |
|-------------------------|------------|
| 1. Vatsal Prakhar Mehta | - 20UCS225 |
| 2. Shruti Sharma | - 20UCS189 |
| 3. Ujjwal Agrawal | - 20UCS214 |
| 4. Shrey Kalpesh Parikh | - 20UCS185 |

Acknowledgment

We are thankful to our professor for providing us an opportunity to work on a project and for giving us their blessings and encouragement. It is a matter of pride and pleasure to express our warm gratitude to the computer staff for keen interest, unceasing resistance and criticism at every stage of this project work. I am also thankful to the Lnmiit college for including the project work in syllabus.

Introduction

Blood donation is a voluntary procedure that can help save the lives of others. Blood Donation and Procurement Organizations play a pivotal role in today's medical institutions. Such organizations are responsible for the evaluation and procurement of blood for blood donation. These organizations represent the front-line of blood procurement, having direct contact with the hospital and the donor.

Our project provides a database for such an organisation. It maintains a database of personal information and critical information like blood group, age, phone number etc of every person in the database design. Only the organisation staff can log in the system and maintain the database. Whenever a patient requiring blood is found, the hospital notifies the organisation of the same and the staff will enter the details of the patient. The list of donors is also maintained and whenever a match is found by the organization, the data of the donation is then entered into the database by the organization staff.

Project Resource Requirements

Tech Used: HTML, CSS, JavaScript, ReactJS
Requirements: MySQL, NodeJS

Login

<u>userID</u>	username	password	createdAt	updatedAt
---------------	----------	----------	-----------	-----------

donors

donors					
<u>donor ID</u>	first-name	last-name	Gender	ContactNumber	dob
bloodType	address1	address2	address3	medical-history	
Created At	updatedAt				

patients

<u>patientID</u>	first-Name	last-Name	Gender	ContactNumber	dob
bloodType	address1	address2	address3	medical-history	createdAt
updatedAt					

blood

<u>bloodID</u>	bloodType	donatedDate	expiryDate	available	donorID
bloodBankID	createdAt	updatedAt			

requirements

<u>requestID</u>	Reason	location	patientID	createdAt	updatedAt
------------------	--------	----------	-----------	-----------	-----------

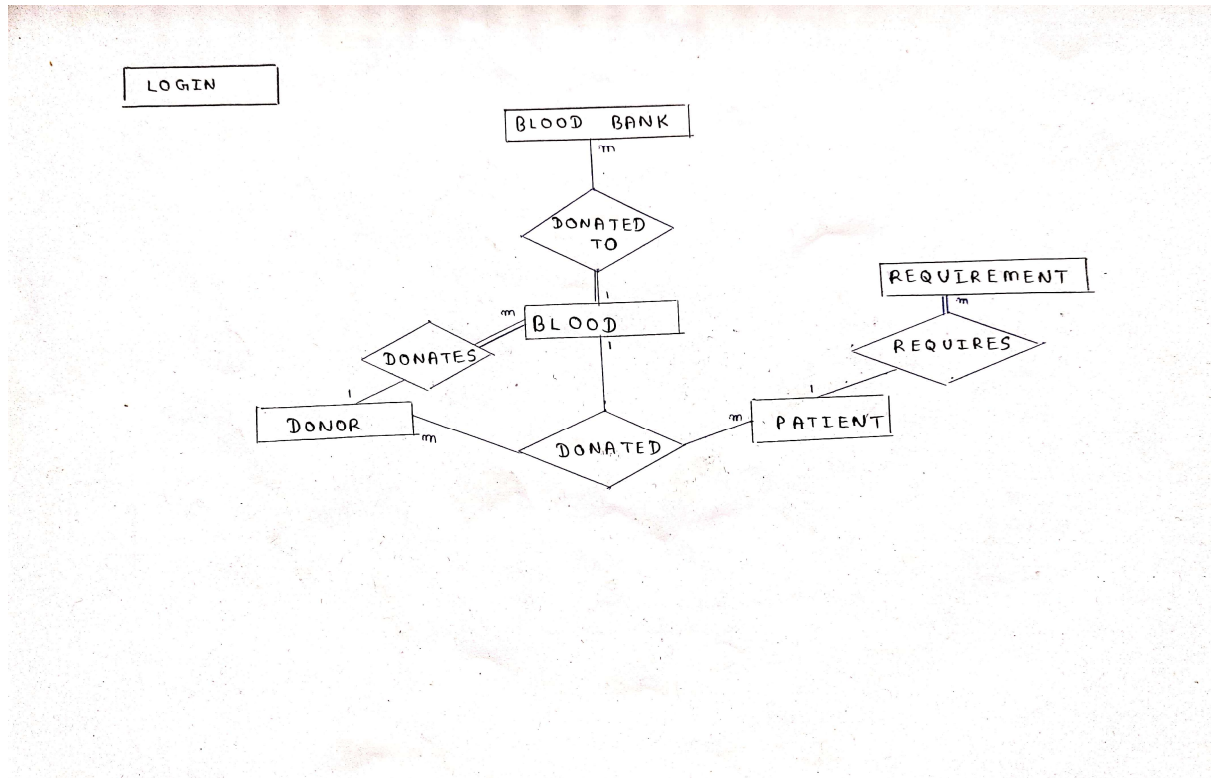
donatedBloods

<u>requestID</u>	reason	location	bloodID	patientID	donorID	createdAt
updatedAt						

bloodBanks

<u>bloodBankID</u>	name	location	createdAt	updatedAt
--------------------	------	----------	-----------	-----------

ER-Diagram



Description of Tables

Donors Table

Field	Type	Null	Key	Default	Extra
1 donorID	int	NO	PRI	<null>	auto_increment
2 First_Name	varchar(50)	NO		<null>	
3 Last_Name	varchar(50)	YES		<null>	
4 Gender	char(1)	YES		<null>	
5 ContactNumber	bigint	NO		<null>	
6 dob	date	NO		<null>	
7 bloodType	varchar(3)	NO		<null>	
8 address1	varchar(50)	NO		<null>	
9 address2	varchar(50)	YES		<null>	
10 address3	varchar(50)	YES		<null>	
11 medical_history	varchar(500)	NO		<null>	
12 createdAt	datetime	YES		CURRENT_TIMESTAMP	DEFAULT_GENERATED
13 updatedAt	datetime	YES		CURRENT_TIMESTAMP	DEFAULT_GENERATED

Patients Table

Field	Type	Null	Key	Default	Extra
patientID	int	NO	PRI	<null>	auto_increment
First_Name	varchar(50)	NO		<null>	
Last_Name	varchar(50)	YES		<null>	
Gender	char(1)	YES		<null>	
ContactNumber	bigint	NO		<null>	
dob	date	NO		<null>	
bloodType	varchar(3)	NO		<null>	
address1	varchar(50)	NO		<null>	
address2	varchar(50)	YES		<null>	
address3	varchar(50)	YES		<null>	
medical_history	varchar(500)	NO		<null>	
createdAt	datetime	YES		CURRENT_TIMESTAMP	DEFAULT_GENERATED
updatedAt	datetime	YES		CURRENT_TIMESTAMP	DEFAULT_GENERATED

Blood Table

Field	Type	Null	Key	Default	Extra
bloodID	int	NO	PRI	<null>	auto_increment
bloodType	varchar(3)	NO		<null>	
donatedDate	date	NO		<null>	
expiryDate	date	NO		<null>	
available	int	NO		1	
donorID	int	YES	MUL	<null>	
bloodBankID	int	YES	MUL	<null>	
createdAt	datetime	YES		CURRENT_TIMESTAMP	DEFAULT_GENERATED
updatedAt	datetime	YES		CURRENT_TIMESTAMP	DEFAULT_GENERATED

Blood Bank Table

Field	Type	Null	Key	Default	Extra
bloodBankID	int	NO	PRI	<null>	auto_increment
name	varchar(50)	NO		<null>	
location	varchar(50)	NO		<null>	
createdAt	datetime	YES		CURRENT_TIMESTAMP	DEFAULT_GENERATED
updatedAt	datetime	YES		CURRENT_TIMESTAMP	DEFAULT_GENERATED

Blood Requirements Table

Field	Type	Null	Key	Default	Extra
requestID	int	NO	PRI	<null>	auto_increment
reason	varchar(50)	YES		<null>	
location	varchar(50)	YES		<null>	
patientID	int	YES	MUL	<null>	
createdAt	datetime	YES		CURRENT_TIMESTAMP	DEFAULT_GENERATED
updatedAt	datetime	YES		CURRENT_TIMESTAMP	DEFAULT_GENERATED

Donated Blood Table

Field	Type	Null	Key	Default	Extra
requestID	int	NO	PRI	<null>	
reason	varchar(50)	YES		<null>	
location	varchar(50)	YES		<null>	
bloodID	int	YES		<null>	
patientID	int	YES	MUL	<null>	
donorID	int	YES	MUL	<null>	
createdAt	datetime	YES		CURRENT_TIMESTAMP	DEFAULT_GENERATED
updatedAt	datetime	YES		CURRENT_TIMESTAMP	DEFAULT_GENERATED

Logins Table

Field	Type	Null	Key	Default	Extra
userID	int	NO	PRI	<null>	auto_increment
username	varchar(255)	YES		<null>	
password	varchar(255)	YES		<null>	
createdAt	datetime	YES		CURRENT_TIMESTAMP	DEFAULT_GENERATED
updatedAt	datetime	YES		CURRENT_TIMESTAMP	DEFAULT_GENERATED

Relationship descriptions:

1. Donated Relation: A one to many relationship between entities donors and blood. There is a total participation from the blood side.
2. Requires: A one to many relationship between entities patient and blood. There is a total participation from the blood side.
3. Donated: A ternary relationship between donors, patients and blood. It is many on donors side, one on blood side, and many on the patients side.
4. Donated to: A one to many relationship existing between entities blood and bloodBanks. There is a total participation from the blood side.

SQL file:

```
create database BBMO;
```

```
use BBMO;
```

```
CREATE TABLE logins
```

```
(  
    userID int primary key auto_increment,  
    username varchar(255),  
    password varchar(255),  
    createdAt datetime default NOW(),  
    updatedAt datetime default NOW()  
);
```

```
CREATE TABLE IF NOT EXISTS donors
```

```
(  
    donorID int primary key auto_increment,  
    First_Name varchar(50) not null,  
    Last_Name varchar(50),  
    Gender char(1),  
    ContactNumber bigint not null,  
    dob date not null,  
    bloodType varchar(3) not null,  
    address1 varchar(50) not null,  
    address2 varchar(50),  
    address3 varchar(50),  
    medical_history varchar(500) not null,  
    createdAt datetime default NOW(),  
    updatedAt datetime default NOW()  
);
```

);

CREATE TABLE IF NOT EXISTS patients

```
(
    patientID    int primary key auto_increment,
    First_Name   varchar(50) not null,
    Last_Name    varchar(50),
    Gender       char(1),
    ContactNumber bigint    not null,
    dob          date       not null,
    bloodType    varchar(3) not null,
    address1     varchar(50) not null,
    address2     varchar(50),
    address3     varchar(50),
    medical_history varchar(500) not null,
    createdAt    datetime default NOW(),
    updatedAt    datetime default NOW()
);
```

CREATE TABLE IF NOT EXISTS blood

```
(
    bloodID    int primary key auto_increment,
    bloodType  varchar(3) not null,
    donatedDate date    not null,
    expiryDate date    not null,
    available  int(1)   not null default 1,
    donorID   int,
    bloodBankID int,
    createdAt  datetime default NOW(),
```

```
        updatedAt datetime default NOW()
    );
```

```
CREATE TABLE IF NOT EXISTS requirements
```

```
(
    requestID int primary key auto_increment,
    reason varchar(50),
    location varchar(50),
    patientID int,
    createdAt datetime default NOW(),
    updatedAt datetime default NOW()
);
```

```
CREATE TABLE IF NOT EXISTS donatedBloods
```

```
(
    requestID int primary key,
    reason varchar(50),
    location varchar(50),
    bloodID int,
    patientID int,
    donorID int,
    createdAt datetime default NOW(),
    updatedAt datetime default NOW()
);
```

```
CREATE TABLE bloodBanks
```

```
(
    bloodBankID int primary key auto_increment,
    name varchar(50) not null,
```



```
location varchar(50) not null,  
createdAt datetime default NOW(),  
updatedAt datetime default NOW()  
);
```

```
ALTER TABLE blood
```

```
ADD FOREIGN KEY (donorID) REFERENCES donors (donorID);
```

```
ALTER TABLE blood
```

```
ADD FOREIGN KEY (bloodBankID) REFERENCES bloodBanks (bloodBankID);
```

```
ALTER TABLE requirements
```

```
ADD FOREIGN KEY (patientID) REFERENCES patients (patientID);
```

```
ALTER TABLE donatedBloods
```

```
ADD FOREIGN KEY (patientID) REFERENCES patients (patientID);
```

```
ALTER TABLE donatedBloods
```

```
ADD FOREIGN KEY (donorID) REFERENCES donors (donorID);
```

```
ALTER TABLE donors
```

```
ADD CONSTRAINT CHECK (bloodType in ('O+', 'O-', 'A+', 'A-', 'B+', 'B-', 'AB+', 'AB-'));
```

```
ALTER TABLE patients
```

```
ADD CONSTRAINT CHECK (bloodType in ('O+', 'O-', 'A+', 'A-', 'B+', 'B-', 'AB+', 'AB-'));
```

```
ALTER TABLE blood
```

```
ADD CONSTRAINT CHECK (bloodType in ('O+', 'O-', 'A+', 'A-', 'B+', 'B-', 'AB+', 'AB-'));
```

```
ALTER TABLE donors
```

```
ADD CONSTRAINT CHECK (999999999 < ContactNumber < 10000000000);
```

```
ALTER TABLE patients
```

```
ADD CONSTRAINT CHECK (999999999 < ContactNumber < 10000000000);
```

```
CREATE TRIGGER clearRequest
```

```
AFTER INSERT
```

```
ON donatedBloods
```

```
FOR EACH ROW
DELETE
FROM requirements
WHERE requirements.requestID = NEW.requestID;
```

```
CREATE TRIGGER donateBloods
AFTER INSERT
ON donatedBloods
FOR EACH ROW
UPDATE blood
SET available=0
WHERE blood.bloodID = NEW.bloodID;
```

```
INSERT INTO donors (First_Name, Last_Name, Gender, ContactNumber, dob, bloodType,
                    address1, address2, address3, medical_history, createdAt, updatedAt)
VALUES ('Shrey', 'Parikh', 'M', 8745212365,
        '2012-1-31', 'AB+', 'Kholi No. 420', 'Prem Gali',
        'Rup Nagar', 'None', now(), now());
```

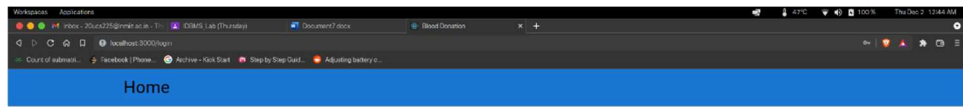
```
INSERT INTO patients (First_Name, Last_Name, Gender, ContactNumber, dob, bloodType,
                      address1, address2, address3, medical_history, createdAt,
                      updatedAt)
VALUES ('ftyvhgvhg', 'yfyvjhhvjy', 'F', 8796512235,
        '1953-4-12', 'AB-', 'yruegidfhjb', 'urtihgkdjf',
        'tywfhgdsa', 'oiuykjlgh', now(), now());
```

```
INSERT INTO bloodBanks (name, location, createdAt, updatedAt)
VALUES ('Maharana pratap singh', 'udaipur', now(), now());
```

```
INSERT INTO blood (bloodType, donatedDate, expiryDate, createdAt,  
                  updatedAt, bloodBankID, donorID)  
VALUES ('AB+', now(), '2022-1-12', now(), now(), 1, 1);
```

Working Project

Login page



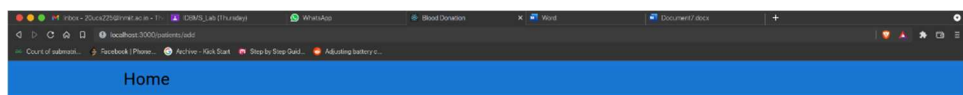
Login

Username
admin

Password
.....

LOGIN

Add patient



Add a new Patient

First Name * baymax	Last Name * xyz
Blood Group * A+	Gender * Male
Contact Number * 1234567891	Date of Birth * 12/02/2001
Address 1 * street 123,mg road	Address 2 * Phadke Road
Address 3 	Medical History * nil

ADD PATIENT

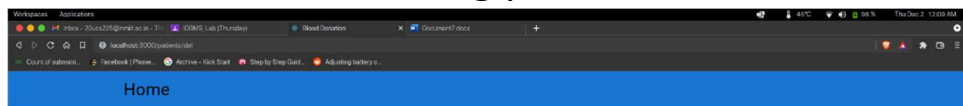
Displaying patients



Requesting blood

[illegible]

Deleting patient



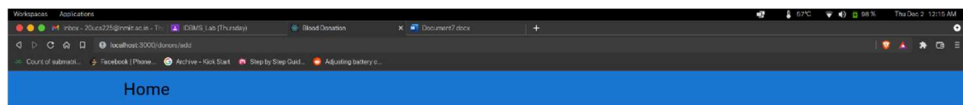
Enter the ID of patient to be deleted

Patient ID *

DELETE



Adding donor



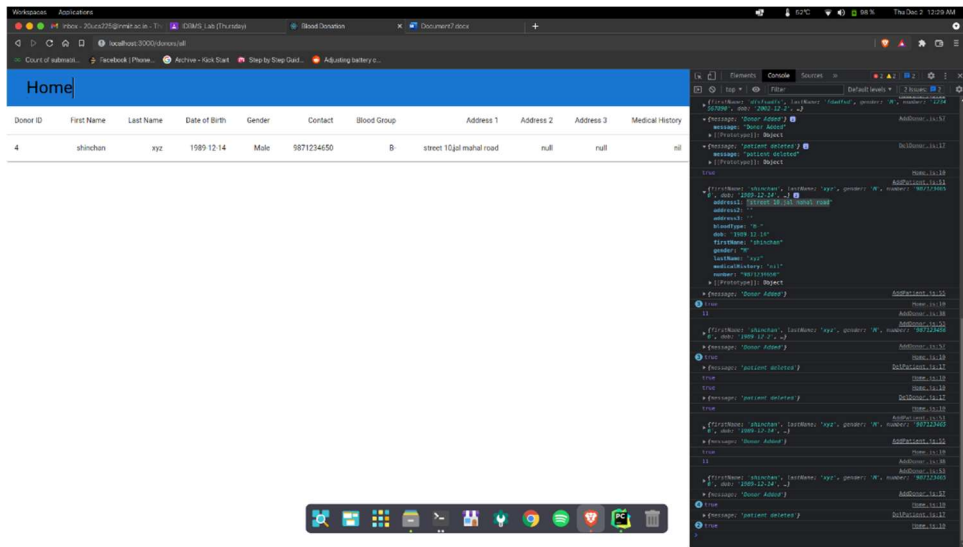
Add a new Donor

First Name *	shinchan	Last Name *	xyz
Blood Group *	B	Gender	
Contact Number *	9871234560	Date of Birth *	12/14/1989
Address 1 *	street 10,jal mahal road	Address 2	
Address 3		Medical History *	nil

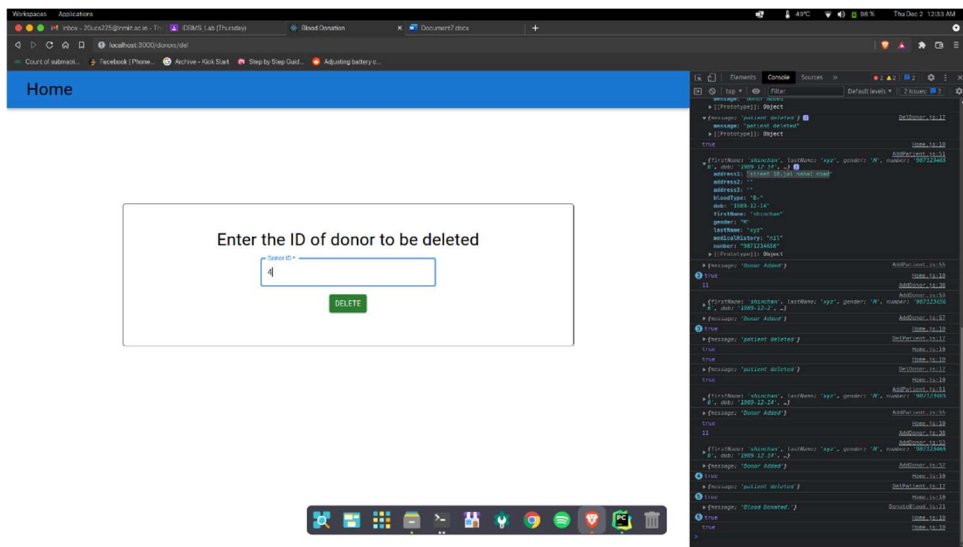
ADD DONOR



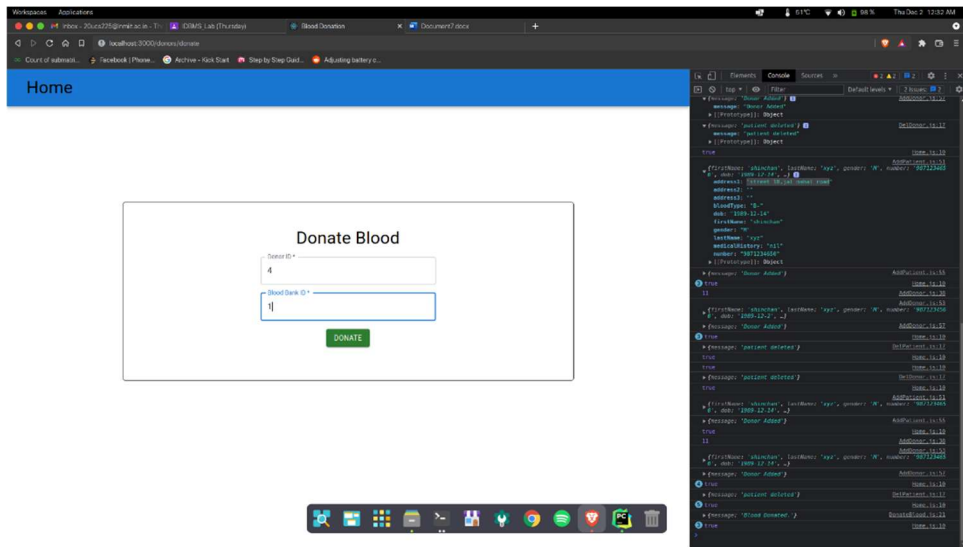
Display donor



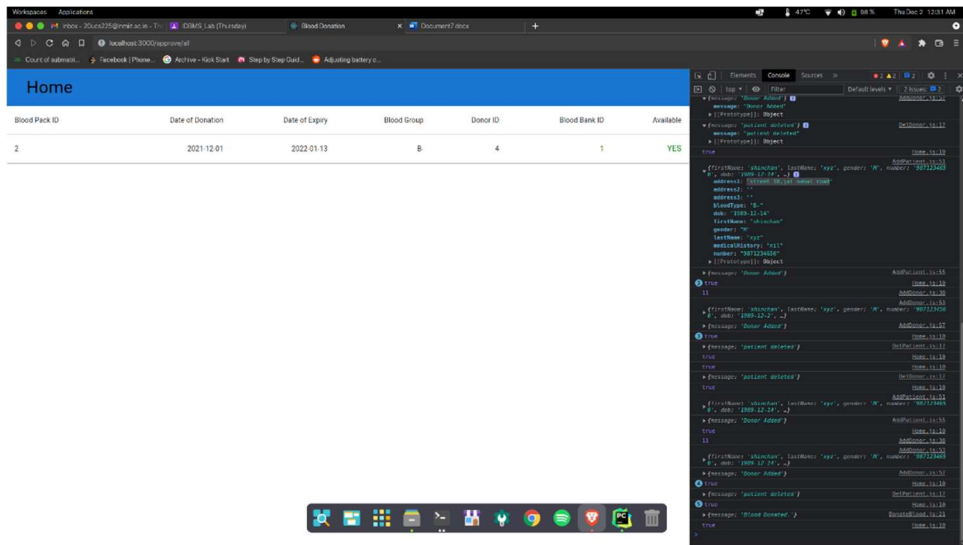
Delete donor



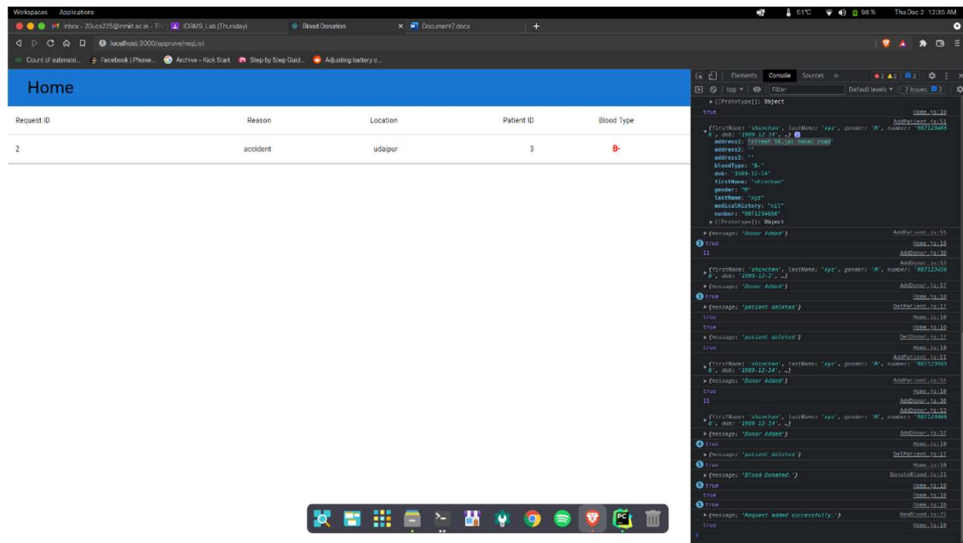
Donate blood



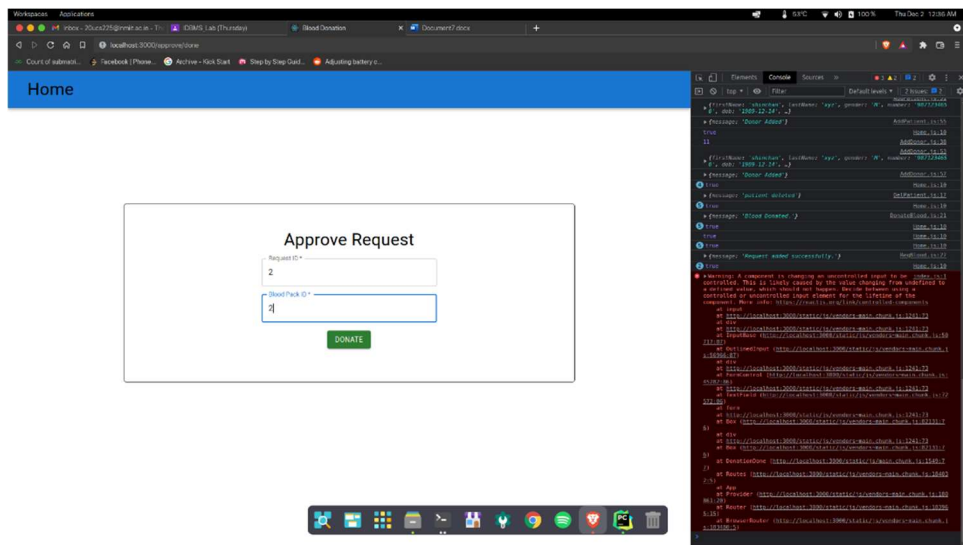
Available blood



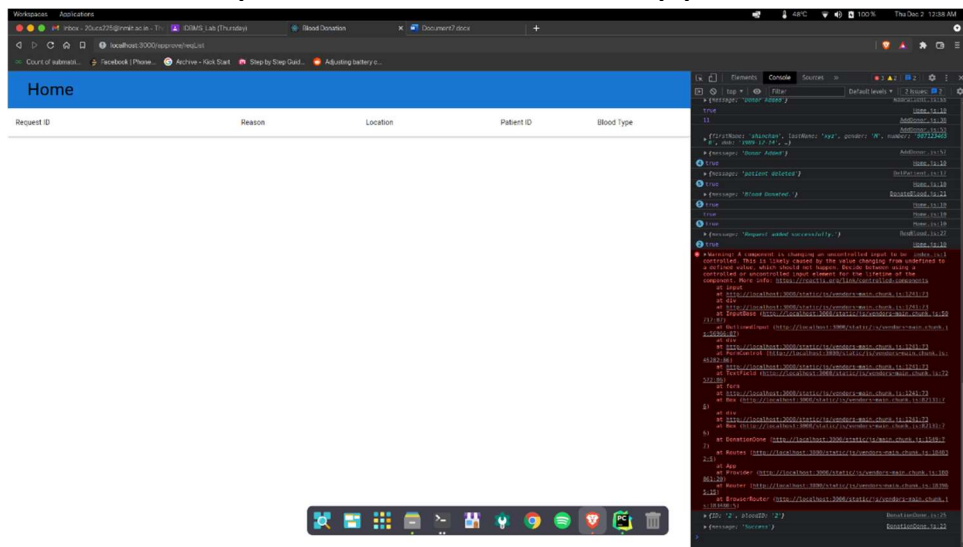
Present blood request



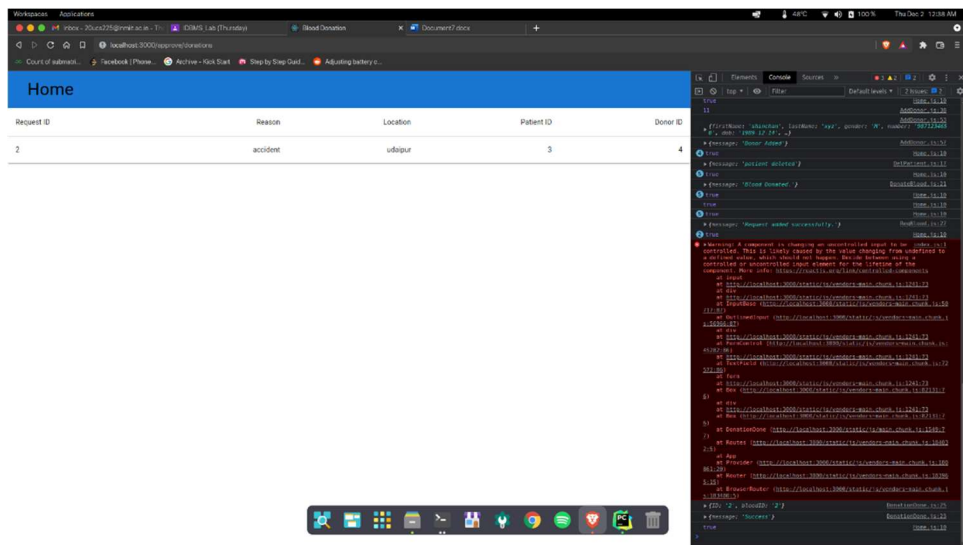
Approving blood request



Request removed after approval



And donated blood added to history



And in database availability changes to no

Blood Pack ID	Date of Donation	Date of Expiry	Blood Group	Donor ID	Blood Bank ID	Available
2	2021-12-01	2022-01-13	B	4	1	NO

Adding blood bank

Add new Blood Bank

Name* rane udai singh

Location* chittoor

ADD BLOOD BANK

Existing blood banks

The screenshot shows a web application interface with a table of existing blood banks. The table has three columns: Blood Bank ID, Name, and Location. There are two entries in the table.

Blood Bank ID	Name	Location
1	Maharaja pratap singh	udipur
2	rana udai singh	chittor

Below the table is a taskbar with various application icons. To the right of the web application is a terminal window showing a list of commands and their outputs, including commands like 'curl', 'http', 'python', and 'python3'.

Deleting blood bank

The screenshot shows a web application interface with a form to delete a blood bank. The form has a title 'Enter the ID of Blood Bank to be deleted' and a text input field with the value '1'. Below the input field is a green 'DELETE' button.

Below the form is a taskbar with various application icons. To the right of the web application is a terminal window showing a list of commands and their outputs, including commands like 'curl', 'http', 'python', and 'python3'.

Deleted from list

