Blood Donation Management System

Introduction

Blood is one of the most vital components of human life. It plays a crucial role in saving lives during medical emergencies such as accidents, surgeries and childbirth. In such critical situations, the availability of the right blood group at the right time can make the difference between life and death. Therefore, blood donation is a noble and essential act that helps maintain an adequate supply of blood in hospitals and healthcare centers.

Despite its importance, many patients and hospitals often face difficulties in locating suitable donors or checking the availability of specific blood groups in nearby blood banks. The lack of proper coordination and real-time information leads to delays, confusion, and sometimes even loss of life.

To overcome these challenges, there is a strong need for a digitalized management system that can store and manage donor details, track blood bank inventory, and process blood requests efficiently. Traditional methods like maintaining paper registers or relying solely on phone calls are often slow, prone to errors, and inefficient in emergency situations.

In contrast, a well-structured database system can provide fast access to donor information, maintain accurate records of blood stock, and update inventory in real time. Such a system benefits hospitals, patients, and donors by ensuring that blood is available when and where it is needed.

This project, **"Blood Donation Management System"**, aims to provide a simple yet effective database solution using **MySQL**. The system will help store donor information, manage blood availability across different banks, and process requests for blood in an organized and reliable manner. This initiative promotes better coordination between donors and hospitals, ultimately contributing to saving lives more efficiently.

Objective

The primary objective of this project is to design and implement a **Blood Donation Management System** using MySQL that efficiently stores and manages all information related to donors, blood banks, and patient requests. The system is developed with the aim of overcoming the limitations of manual record-keeping and ensuring quick access to critical information during emergencies.

The key objectives of this project are:

- **To store donor details** such as name, age, gender, blood group, contact information, and last donation date in an organized manner for easy retrieval.

- **To manage blood bank records**, including their location and the stock of different blood groups available at any given time.

- **To handle patient requests for blood** by recording their requirements and updating the blood inventory once the request is approved.

- **To simplify the search process** for hospitals and patients by enabling quick identification of suitable donors or the availability of required blood groups in specific banks.

- **To ensure accuracy and reliability** in record-keeping, reducing the chances of errors common in traditional manual systems.

Through this project, we aim to create a small-scale model that demonstrates how technology can support the healthcare system and ensure timely access to blood for patients in need.

Scope

The scope of this project is to design a small-scale database system for managing blood donation information. It includes storing donor details, maintaining hospital and blood bank records, and handling patient requests for blood. The system updates the blood inventory when donations are made or requests are approved. Advanced features like online registration, notifications, or large-scale hospital integration are not included. This project mainly serves as a simple model to show how MySQL can efficiently manage essential blood donation data.

Technology Used

This project is implemented using **MySQL**, a relational database management system. MySQL is used to create tables for donors, blood banks, blood inventory, and requests, and to perform operations like adding, updating, and retrieving data. It helps manage donor information, track blood stock, and process patient requests efficiently in a structured and reliable way

Database Design and Tables

Tables:

- **Donors:** Stores donor information like name, age, blood group, contact, and last donation date.

- **BloodBanks:** Contains details of hospitals or blood banks including name, location, and contact.

- **BloodInventory:** Tracks the number of blood units available for each blood group in each blood bank.

- **Requests:** Records patient requests for blood, including required blood group, units, hospital, and status.

MySQL Implementation

1. Table Creation Scripts (CREATE TABLE)

- **Shows how the tables for donors, blood banks, blood inventory, and requests are created in MySQL.**

 -- Create a Donor table

create table donors(donor_id int AUTO_INCREMENT PRIMARY KEY,

name varchar(50),age int, gender varchar(9),blood_group varchar(5),

contact varchar(50),last_donation_date date);

describe donors;

-- Create a Blood Bank table

create table bloodbanks(bank_id int AUTO_INCREMENT PRIMARY KEY,

name varchar(50),location varchar(100),contact varchar(50));

describe bloodbanks;

-- Create a Blood Inventory table

create table bloodinventory (inventory_id int AUTO_INCREMENT PRIMARY KEY,

bank_id int,blood_group varchar(5),units_available int, FOREIGN KEY(bank_id) REFERENCES bloodbanks(bank_id));

describe bloodinventory;

-- Create a Requests table

create table requests (request_id INT AUTO_INCREMENT PRIMARY KEY, patient_name varchar(50),blood_group varchar(5),units_needed int,

hospital varchar(100),status varchar(20) DEFAULT 'Pending');

describe requests;

2. Sample Data Insertion (INSERT INTO)

- **Demonstrates adding example records into each table to test the system.**

  -- Inserting Records into donors table

  insert into donors (name, age, gender, blood_group, contact, last_donation_date)

  values ('Priya', 22, 'Female', 'A+', '9876543210', '2025-06-10'),

      ('Jai', 28, 'Male', 'O-', '9123456789', '2025-07-15');

  -- Inserting Records into bloodbanks table

  insert into bloodbanks (name, location, contact)

  values('City Blood Bank', 'Madurai', '9988776655');

  -- Inserting Records into bloodinventory table

  insert into bloodinventory (bank_id, blood_group, units_available)

  values (1, 'A+', 10), (1, 'O-', 5);

  -- Inserting Records into requests table

  insert into requests (patient_name, blood_group, units_needed, hospital)

  values('Kumar', 'A+', 2, 'Apollo Hospital');

  -- View the inserted values using SELECT statement:

  select * from donors;

  select * from bloodbanks;

  select * from bloodinventory;

  select * from requests;

3. Queries for Real Use Cases

-- Find all donors of a particular blood group

```sql
select  name, contact
from donors
where blood_group = 'A+';
```

-- Check available blood stock in a blood bank

```sql
select blood_group, units_available
from bloodinventory
where bank_id = 1;
```

-- Add stock after a new donation

```sql
update bloodinventory
set units_available = units_available + 1
where bank_id = 1 AND blood_group = 'A+';


select * from bloodinventory;
```

-- ApproveRequest_UpdateStock

```sql
update bloodinventory
set units_available = units_available - 1
where bank_id = 1 AND blood_group = 'A+';


select * from bloodinventory;
```

-- Approve a blood request

```sql
update requests

set status = 'Approved'

where request_id = 1;


select * from requests;
```

-- List pending requests

```sql
select * from requests where status = 'Pending';
```

-- Report – Count total donors by blood group

```sql
select blood_group, COUNT(*) AS total_donors

from donors

GROUP BY blood_group;
```

Live SQL Demo:
The database schema, sample data, and queries for this project have been implemented and tested online. You can view and run them directly using the following link:

https://sqlfiddle.com/mysql/online-compiler?id=46b092c9-9080-4272-b9b7-f8fbb23db6a9

Output1

Create a donors table

Field	Type	Null	Key	Default	Extra
donor_id	int	NO	PRI	NULL	auto_increment
name	varchar(50)	YES		NULL	
age	int	YES		NULL	
gender	varchar(9)	YES		NULL	
blood_group	varchar(5)	YES		NULL	
contact	varchar(50)	YES		NULL	
last_donation_date	date	YES		NULL	

Create a bloodbank table

Field	Type	Null	Key	Default	Extra
bank_id	int	NO	PRI	NULL	auto_increment
name	varchar(50)	YES		NULL	
location	varchar(100)	YES		NULL	
contact	varchar(50)	YES		NULL	

Create a bloodinventory

Field	Type	Null	Key	Default	Extra
inventory_id	int	NO	PRI	NULL	auto_increment
bank_id	int	YES	MUL	NULL	
blood_group	varchar(5)	YES		NULL	
units_available	int	YES		NULL	

Create a Create a requests table

Field	Type	Null	Key	Default	Extra
request_id	int	NO	PRI	NULL	auto_increment
patient_name	varchar(50)	YES		NULL	
blood_group	varchar(5)	YES		NULL	
units_needed	int	YES		NULL	
hospital	varchar(100)	YES		NULL	
status	varchar(20)	YES		Pending	

Find all donors of a particular blood group

name	contact
Priya	9876543210

Check available blood stock in a blood bank

blood_group	units_available
A+	10
O-	5

Add stock after a new donation

inventory_id	bank_id	blood_group	units_available
1	1	A+	11
2	1	O-	5

ApproveRequest_UpdateStock

inventory_id	bank_id	blood_group	units_available
1	1	A+	10
2	1	O-	5

Approve a blood request

request_id	patient_name	blood_group	units_needed	hospital	status
1	Kumar	A+	2	Apollo Hospital	Approved

Report – Count total donors by blood group

blood_group	total_donors
A+	1
O-	1

Conclusion

The Blood Donation Management System provides an efficient way to store donor details, manage blood bank inventories, and handle patient requests in a reliable manner. By using MySQL, the system ensures accuracy, quick data retrieval, and real-time updates of blood availability, which helps hospitals and patients during critical emergencies.

This project highlights the importance of digital solutions in healthcare management. It not only reduces manual errors but also simplifies the process of locating donors and tracking blood stock. With further improvements, such as integrating a user-friendly interface or connecting with hospital networks, this system can play a significant role in saving lives and strengthening healthcare services.