Information Systems and Data Modeling – IT1090

Assignment

Title: **Blood Donation System**

 Batch Number: 09.02 Group Number: MLB_09.02_06

Declaration:

We hold a copy of this assignment that we can produce if the original is lost or damaged.

We hereby certify that no part of this assignment has been copied from any other group's work or from any other source. No part of this assignment has been written / produced for our group by another person except where such collaboration has been authorized by the subject lecturer/tutor concerned.

Group Members:

AHAMED M.N.I	IT23389656	
FERNANDO B.S.N	IT23382244	
BAGYA R.M.S	IT23394124	
PERERA K.D.S	IT23391468	
NETHSARANI P.A.T	IT23388284	

Submitted on: 03/05/2024.

Introduction

shortcomings of the manual procedures that were previously used for blood donation campaigns. The management of blood donation drives may be made more efficient and automated with the help of this technology. Additionally, blood banks and other organizations may effectively manage donors, blood inventories, and donation events with the help of this system.

Blood banks and donors can use this system to manage eligibility and history, plan appointments, track donor information, keep an eye on blood supply levels, and access donation records and payment histories. Blood banks can reduce the challenges posed by manual procedures and enhance the general effectiveness of blood donation management by implementing this automated approach. Donors, volunteers, coordinators, and system administrators are among the entities that can access the system using conveniently handle their work and maintain their own user accounts. A dependable and safe database system will be provided by the system's encrypted data logs and limited staff access to donor information. Furthermore, users may quickly retrieve chosen information thanks to simple data retrieval.

Data on blood donations is managed securely since only system administrators can access databases and the web. The availability of system administrators will make it simple, accurate, and effective to upload data about contribution events, as well as donor and volunteer details, to the database.

Blood banks and donors alike can gain from safe, easy-to-use, and effective procedures as well as a well-structured volunteer and donor management system by putting this new blood donation management system into place. Furthermore, overseeing the company's Compared to manual blood donation administration procedures, the database and web content will be more precise and efficient.

Hypothetical scenario

Blood donation system (BDS) facilitates the convenience of donors, health care professionals and blood recipients. DS has four main types of users who can login to the system, donors, recipients,

health care professionals and advocates. Donors can be either registered or unregistered. The system allows the unregistered donors to get registered to the system. If the donor is an unregistered donor, they must first meet the eligibility criteria to get registered.

The system provides ample amount facilities for registered donors, healthcare professionals, blood recipients and advocates. Once a registered donor enters the portal and wishes to donate blood, they must provide medication history data and meet the eligibility criteria. Once their eligibility is confirmed they can then proceed to donate blood.

When the recipients submit a request for transfusion, the system notifies the donors the and the request is carried out by the healthcare professional. When donors donate blood, health care professionals view their profiles through the system and select a donor for the transfusion and notify the blood recipient.

When a request for transfusion is submitted, if there are no available blood units in the inventory the system notes the advocates, and the advocate initiates a blood campaign and publish the poster in the system.

Requirement Analysis

Main Requirements

1) **First time donor**

user Requirements-

- First time donors can register to the system by providing required details for the registration.
- First time donors can log in to the system and check available appointment schedules.
- First time donors can book an appointment through the app at a nearby blood donation Centre.
- Donors can answer health related questions before the donation.
- First time donor can access information about the donation process, potential risks and provide their consent for donation.
- First time donors can access information on post donation care.

System Requirements-

- System should validate the information provided by the user.
- System should approve and record the details.
- System should provide user authentication and data encryption to secure user privacy.
- The system should support appointment booking, rescheduling, deleting appointments.
- The system should verify donor eligibility.
- The system should generate donor records to be accessed by relevant parties concerned.

2) **Blood recipient**

User requirements-

- Blood recipients can register to the system providing required information relevant to blood transfusion.
- User can request a blood transfusion.
- Users can search for potential donors who fit the transfusion criteria.
- Users can receive notifications and massages when a compatible donor is found and verified.
- Users can view donor profiles.
- Users can answer health related questions for transfusion.
- Users can access information about the process and risks.

System Requirements-

- System should validate the data provided by the recipient.
- The system should manage recipient records, blood requests.
- System should provide an efficient way to search and match compatible donors who meet the required criteria.
- System should provide user friendly interface for recipients and donors to communicate easily.
- System should provide data encryption to secure privacy.
- System should be scalable to handle large number of users.

3) **Health care professional**

User Requirements-

- Regular donors should undergo donor registration.
- Regular donors should book an appointment.
- Regular donors should participate in donor eligibility screening.
- Regular donors should provide donor history.
- Regular donors should receive notification and reminders.
- Regular donors should offer feedback and participate in surveys.
- Regular donors Should engage in account management.

System Requirements-

- System should handle donor registration and profile management.
- System Should manage Donation Scheduling.
- System should conduct donation eligibility screening.
- System should track donation history.
- System should provide donation education and resources.
- System should handle donor communication.
- System should manage rewards and recognition.

4) **<u>Advocate and Coordinator</u>**

User requirements-

- Coordinator can sign into the website by providing required login credentials.
- Coordinator should handle volunteer scheduling and task assignment.
- Coordinator should manage camp logistics and inventory.
- The coordinator should oversee blood collection tracking and integration with blood bank systems.
- The coordinator should handle communication and reporting for donors, volunteers, and staff.
- The coordinator should ensure data security and privacy compliance.
- The coordinator should oversee analytics and performance monitoring.
- The coordinator should manage Configurability and customization options.

System requirements-

- System should validate the user login credentials.
- System should update the details of the appointments of the donors.
- System should store the donor feedback, contact details, and reviews and display them.
- System should generate reports such as donation report, blood bank report.
- System should validate login credentials entered by the coordinator.

5) <u>Health care professional</u>

user Requirements

- Health care professional can log into the system and check pre donation section
- Health care professional can check blood recipient details and donor's form
- Health care professional should do analyzing data and assessing personal readiness.
- Health care professional should schedule appointments

System Requirements

- System should update the pre donation section including blood donor and blood recipient details.
- System should update doctor's details schedule and appointment.
- System should provide platform to view donor profile categorize them according to eligibility criteria
- System should provide data verification to eligible donor

Non-Functional Requirements

Quality qualities are also referred to as non-functional requirements. It explains the aspects of the system that don't immediately relate to a certain functionality. It's possible that non-functional requirements are more important than functional ones. The system might be ineffective if these goals are not met.

Usability

- Interface that is easy to use for campaign coordinators and donors alike.
- Simple navigation and user-friendly processes
- Features that are accessible to people with impairments.

Performance

- Quick response times for appointment booking and donor registration.
- Capacity to manage multiple users and heavy traffic during hours of peak donation.

Reliability

- prompt response times for scheduling appointments and registering donors.
- Ability to handle many users and a lot of traffic during peak donation hours.

Security

- Safe methods for authorization and authentication.
- Sensitive donor data, including contact information and medical history, is encrypted.
- adherence to data protection laws (such as GDPR and HIPAA)

Scalability

- Capacity to manage a growing number of campaigns and donors over time.
- support for blood donation centers and different locations.

Maintainability

- Modular architecture for simple improvements and upgrades
- code that is thoroughly documented for future upkeep.

Compatibility

- Compatibility across platforms and browsers
- Integration with third-party systems (such as databases from blood banks and medical records)

Localization

- Support for a variety of languages and cultural tastes.

Privacy

- strict observance of donor consent and privacy policies
- Donor data handling and storage that is secure.

Scalability

- extensive analytics and reporting capabilities.
- Real-time tracking of campaign advancement and donation targets

Data Requirements

Regular Donor

- Donor ID (Donor_id)
- Donor name (D_name)
- Donor blood type (B_type)
- Donor medical history (M_history)

Blood Recipient

- Blood recipient ID (BRID)
- Blood recipient name (B_Name)
- Age (B_age)
- Blood recipient contact number (B_contact)
- Recipient Blood type (B_bloodgrp)

Health Care

- Doctor ID (OCID)
- Doctor name (Ha_Name)

- Doctor contact number (Ha_Phonenumber)
- Doctor availability (professional_availability)
- Docter email (Ha_Email)

Blood Bank

- Blood bank ID (ID)
- Blood bank name (B_Name)
- Blood bank location (B_location)
- Blood bank number (B_contact)

Campaign

- Campaign ID (campaign_ID)
- Campaign Time (Campaign_time)
- Campaign Contact person (Contact_person)
- Campaign venue (Place)

Campaign Coordinator

- Coordinator name (cc_Name)
- Coordinator email (cc_Email)
- Coordinator ID (CAID)

Reception

- Reception Staff ID (staff_ID)
- number of donors (No_Of_Donor)
- date (R_date)
- time (R_time)

First Time Donor

- First time donor NIC (NIC)
- First time donor address (Address)
- First time donor name (F_Donor_name)
- First time donor gender (F_Donor_Gender)

Entity – Relationship Diagram
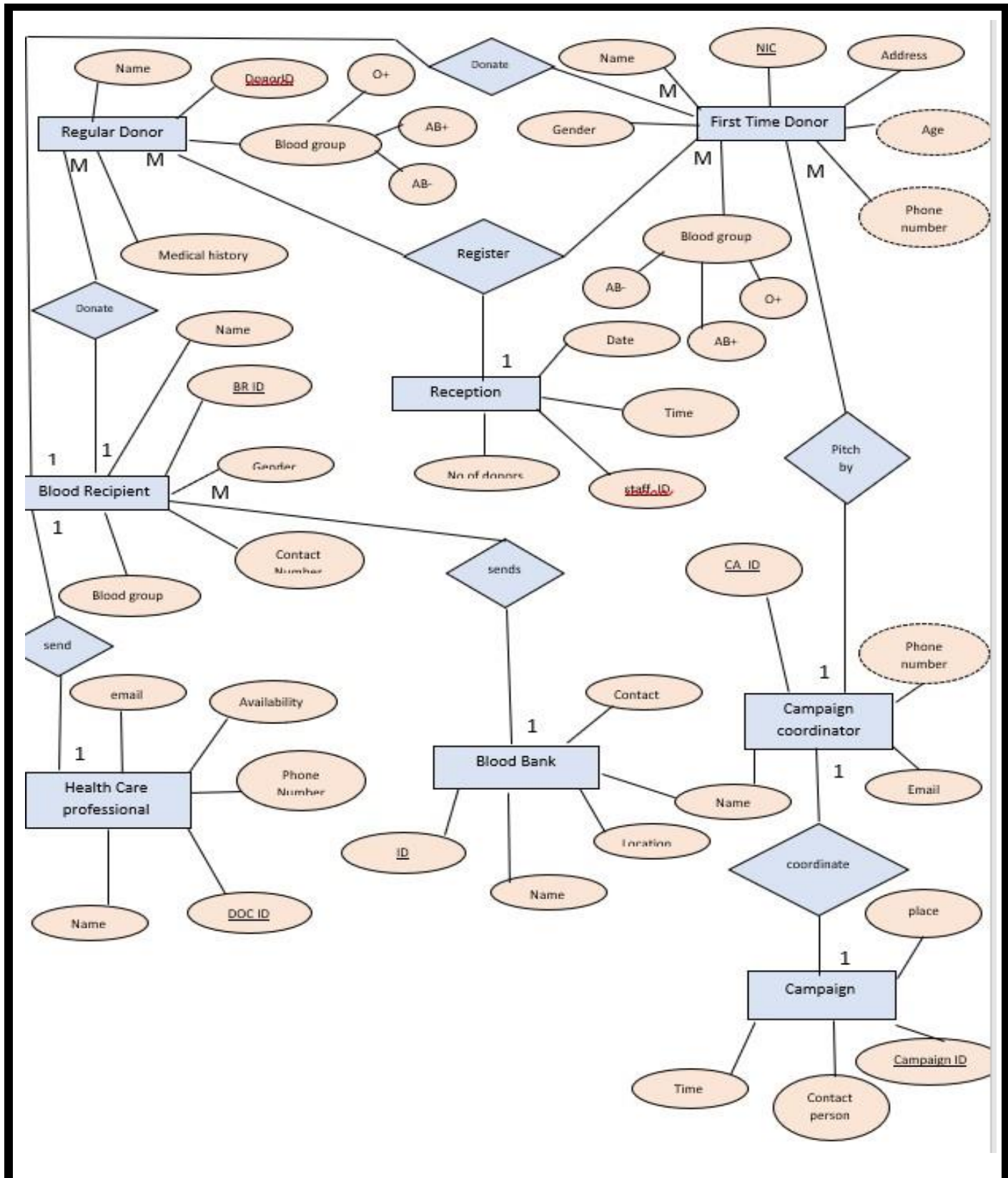
Relational Schema

Regular Donor

DonorID	name	Blood type	Medicalhistory	BRID	Staff ID

Blood Recipient

BRID	Name	Gender	Contact number	Blood group	ID

Health care

OCID	Name	Phone Number	Professional Availability	email

Blood bank

ID	Name	Location	Contact number

campaign

Campaign ID	time	Contact person	place

Campaign coordinator

Name	Email	CAID

Reception

Staff ID	No of donors	date	time

First Time Donor

NIC	Address	name	gender	BRID	staffID	CAID

Regular donor_First time donor

NIC	DonorID

First time donor-mobile

NIC	phone number

Campaign coordinator_mobile

CAID	Phone number

14

SQL Database Creation

Create Tables

```sql
USE [Blood_donation];

CREATE TABLE Blood_Recipient(
    BRID int PRIMARY KEY,
    Blood_Name varchar(30),
    B_age varchar(20),
    B_contact varchar(20),
    B_bloodgrp varchar(30),
        ID int,
);
ALTER  TABLE Blood_Recipient

ADD CONSTRAINT FK_ID
FOREIGN KEY (ID)
REFERENCES Blood_Recipient(ID)
ON DELETE CASCADE;

CREATE TABLE Campaign(
    campaign_ID int PRIMARY KEY,
    Campaign_time varchar(30),
    Contact_person varchar(20),
    Place varchar(20)
);

CREATE TABLE Regular_Donor(
    Donor_id int PRIMARY KEY,
    Donor_name varchar(30),
    Blood_type varchar(20),
    Medical_history varchar(20),
    BRID int,
        staff_ID int

);

ALTER  TABLE Regular_Donor

ADD CONSTRAINT FK_BRID
FOREIGN KEY (BRID)
REFERENCES Blood_Recipient(BRID)
ON DELETE CASCADE;

CREATE TABLE Reception(
    staff_ID int PRIMARY KEY,
    No_Of_Donor varchar(20),
    R_date varchar(20),
    R_time varchar(20)
);
```

```sql
CREATE TABLE Blood_bank(
    ID int PRIMARY KEY,
    B_Name varchar(20),
    B_location varchar(20),
    B_contact varchar(20)
);

CREATE TABLE Health_care(
    OCID int PRIMARY KEY,
    Ha_Name varchar(20),
    Ha_Phonenumber varchar(20),
    professional_availability varchar(30),
    Ha_Email varchar(30)
);

CREATE TABLE First_time_donor(
    NIC int PRIMARY KEY,
    Address  varchar(20),
    F_Donor_name varchar(20),
    F_Donor_Gender varchar(20),
    BRID int,
    staff_ID int,
    CAID int,
    Campaign_ID int,

);
ALTER  TABLE First_time_donor

ADD CONSTRAINT FK_BRID_
FOREIGN KEY (BRID)
REFERENCES Blood_Recipient(BRID)
ON DELETE CASCADE;

ALTER  TABLE First_time_donor

ADD CONSTRAINT FK_staff_ID
FOREIGN KEY (staff_ID)
REFERENCES Reception(staff_ID)
ON DELETE CASCADE;

ALTER  TABLE First_time_donor

ADD CONSTRAINT FK_campaign_ID
FOREIGN KEY (campaign_ID)
REFERENCES Campaign(campaign_ID)
ON DELETE CASCADE;

ALTER  TABLE First_time_donor

ADD CONSTRAINT FK_CAID
FOREIGN KEY (CAID)
REFERENCES campaign_coordinator(CAID)
ON DELETE CASCADE;

CREATE TABLE campaign_coordinator(
    CAID int PRIMARY KEY,
    cc_Email varchar(20),
```

```sql
    cc_Name varchar(20)
);

ALTER  TABLE Blood_Recipient

ADD CONSTRAINT FK_ID
FOREIGN KEY (ID)
REFERENCES Blood_bank(ID)
ON DELETE CASCADE;

ALTER  TABLE Regular_Donor

ADD CONSTRAINT FK_staff_ID_
FOREIGN KEY (staff_ID)
REFERENCES Reception(staff_ID)
ON DELETE CASCADE;
```

Insertion of sample data into tables

```sql
INSERT INTO Blood_bank VALUES ('1002','AB-','Asiri Hospital','0453456789');
INSERT INTO Blood_bank VALUES ('1051','AB+','Queens Hospital','0246785678');
INSERT INTO Blood_bank VALUES ('2021','O+','National Hospital','0112345678');
INSERT INTO Blood_bank VALUES ('1430','B-','MDK Hospital','0356784589');
INSERT INTO Blood_bank VALUES ('2140','B-','Navaloka Hospital','0453456789');


INSERT INTO Blood_Recipient VALUES('834567','Ijas
Ahamad','22','0771005129','AB+','1002');
INSERT INTO Blood_Recipient VALUES('678401','Nimasha Fernando','20','0713409871','AB-
','1051');
INSERT INTO Blood_Recipient VALUES('209500',' Sadumina
Bagya','24','0749071221','B+','2140');
INSERT INTO Blood_Recipient VALUES('100345','Dhananjana Perera
','26','078205189','B','1430');
INSERT INTO Blood_Recipient VALUES('7610982','Thanushi
Nethsarani','20','0739057211','O+','2140');


SELECT * FROM Blood_Recipient

DELETE FROM Blood_Recipient WHERE BRID='1';
DELETE FROM Blood_Recipient WHERE BRID='102';


INSERT INTO Blood_bank VALUES ('119','Sandalika','Rathnapura','0769655970');
INSERT INTO Blood_bank VALUES ('120','Thanushi','Malabe','0712525251');
INSERT INTO Blood_bank VALUES ('121','ijas','Maradhana','0701001010');


SELECT * FROM Blood_bank
SELECT * FROM Blood_bank WHERE B_Name LIKE 's%'

INSERT INTO Campaign VALUES ('222','09/04/2024 10.00 AM','0742897689','Gampaha');
INSERT INTO Campaign VALUES ('223','12/05/2024 10.00 AM','0723457877','Rathnapura');
INSERT INTO Campaign VALUES ('224','15/07/2024 10.00 AM','0784568932','Colombo');
INSERT INTO Campaign VALUES ('225','23/08/2024 10.00 AM','0713458972','Galle');
INSERT INTO Campaign VALUES ('226','09/10/2024 10.00 AM','0742897689','Kalutara');
```

```sql
SELECT * FROM Campaign

INSERT INTO campaign_coordinator VALUES ('207758','ruvini@gmail.com','Ruvini Dias');
INSERT INTO campaign_coordinator VALUES ('348902','sadalika@gmail.com','Sadalika
Nethmini');
INSERT INTO campaign_coordinator VALUES ('202561','shubmangill@gmail.com','Shubman
Gill');
INSERT INTO campaign_coordinator VALUES ('234896','adithya@gmail.com',' Adithya Shehan');

SElECT * FROM campaign_coordinator



INSERT INTO Health_care VALUES ('450567','Shawn
Mendis','0775623410','Monday','shawn123@gmail.com');
INSERT INTO Health_care VALUES ('905763','Naveen
Methshan','0724509812','Tuesday','naveenmethshan@gmail.com');
INSERT INTO Health_care VALUES ('678567',' Pamisha
peiris','0754693021','Wednesday','pamishaasiri@gmail.com');
INSERT INTO Health_care VALUES ('295671','Binura
Akash','0763451209','Monday','akashdissanayaka@gmail.com');
INSERT INTO Health_care VALUES ('928675','Malindu Sanketh
','071894567','Friday','malindusanketh@gmail.com');

SELECT * FROM Health_care

INSERT INTO Reception VALUES ('355908','10','02/04/2024','11.00 AM');
INSERT INTO Reception VALUES ('355200','15','08/05/2024','3.00 PM');
INSERT INTO Reception VALUES ('340077','8','10/07/2024','8.00 AM');
INSERT INTO Reception VALUES ('320078','21','10/08/2024','2.00 PM');
INSERT INTO Reception VALUES ('340100','24','02/10/2024','11.00 AM');

SELECT * FROM Reception

INSERT INTO Regular_Donor VALUES('44565','kapurubola','O+','No','834567','355908');
INSERT INTO Regular_Donor VALUES('67548','sudu menika','AB-','Yes','678401','355200');
INSERT INTO Regular_Donor VALUES('98567','johny papa','AB+','Yes','209500','340077');
INSERT INTO Regular_Donor VALUES('12754','maithiripala sirisena','O-
','No','209500','320078');
INSERT INTO Regular_Donor VALUES('89566','thomas alwa
edison','B','No','209500','340100');

SELECT * FROM Regular_Donor
```

Examples


```sql
INSERT INTO Blood_bank VALUES ('119','Sandalika','Rathnapura','0769655970');
INSERT INTO Blood_bank VALUES ('120','Thanushi','Malabe','0712525251');
INSERT INTO Blood_bank VALUES ('121','ijas','Maradhana','0701001010');


SELECT * FROM Blood_bank
SELECT * FROM Blood_bank WHERE B_Name LIKE 's%'

INSERT INTO Campaign VALUES ('222','09/04/2024 10.00 AM','0742897689','Gampaha');
INSERT INTO Campaign VALUES ('223','12/05/2024 10.00 AM','0723457877','Rathnapura');
INSERT INTO Campaign VALUES ('224','15/07/2024 10.00 AM','0784568932','Colombo');
INSERT INTO Campaign VALUES ('225','23/08/2024 10.00 AM','0713458972','Galle');
INSERT INTO Campaign VALUES ('226','09/10/2024 10.00 AM','0742897689','Kalutara');

SELECT * FROM Campaign
```

	campaign_ID	Campaign_time	Contact_person	Place
1	222	09/04/2024 10.00 AM	0742897689	Gampaha
2	223	12/05/2024 10.00 AM	0723457877	Rathnapura
3	224	15/07/2024 10.00 AM	0784568932	Colombo
4	225	23/08/2024 10.00 AM	0713458972	Galle
5	226	09/10/2024 10.00 AM	0742897689	Kalutara


```sql
ON DELETE CASCADE;

INSERT INTO Blood_bank VALUES ('1002','AB-','Asiri Hospital','0453456789');
INSERT INTO Blood_bank VALUES ('1051','AB+','Queens Hospital','0246785678');
INSERT INTO Blood_bank VALUES ('2021','O+','National Hospital','0112345678');
INSERT INTO Blood_bank VALUES ('1430','B-','MDK Hospital','0356784589');
INSERT INTO Blood_bank VALUES ('2140','B-','Navaloka Hospital','0453456789');


INSERT INTO Blood_Recipient VALUES('834567','Ijas Ahamad','22','0771005129','AB+','1002');
INSERT INTO Blood_Recipient VALUES('678401','Nimasha Fernando','20','0713409871','AB-','1051');
INSERT INTO Blood_Recipient VALUES('209500',' Sadumina Bagya','24','0749071221','B+','2140');
INSERT INTO Blood_Recipient VALUES('100345','Dhananjana Perera ','26','078205189','B','1430');
INSERT INTO Blood_Recipient VALUES('7610982','Thanushi Nethsarani','20','0739057211','O+','2140');


SELECT * FROM Blood_Recipient
```

	BRID	Blood_Name	B_age	B_contact	B_bloodgrp	ID
1	100345	Dhananjana Perera	26	078205189	B	1430
2	209500	Sadumina Bagya	24	0749071221	B+	2140
3	678401	Nimasha Fernando	20	0713409871	AB-	1051
4	834567	Ijas Ahamad	22	0771005129	AB+	1002
5	7610982	Thanushi Nethsarani	20	0739057211	O+	2140

Screen 1

```sql
SELECT * FROM campaign_coordinator


INSERT INTO Health_care VALUES ('450567','Shawn Mendis','0775623410','Monday','shawn123@gmail.com');
INSERT INTO Health_care VALUES ('905763','Naveen Methshan','0724509812','Tuesday','naveenmethshan@gmail.com');
INSERT INTO Health_care VALUES ('678567',' Pamisha peiris','0754693021','Wednesday','pamishaasiri@gmail.com');
INSERT INTO Health_care VALUES ('295671','Binura Akash','0763451209','Monday','akashdissanayaka@gmail.com');
INSERT INTO Health_care VALUES ('928675','Malindu Sanketh ','071894567','Friday','malindusanketh@gmail.com');

SELECT * FROM Health_care

INSERT INTO Reception VALUES ('355908','10','02/04/2024','11.00 AM');
INSERT INTO Reception VALUES ('355200','15','08/05/2024','3.00 PM');
INSERT INTO Reception VALUES ('340077','8','10/07/2024','8.00 AM');
INSERT INTO Reception VALUES ('320078','21','10/08/2024','2.00 PM');
INSERT INTO Reception VALUES ('340100','24','02/10/2024','11.00 AM');

SELECT * FROM Reception

INSERT INTO Regular_Donor VALUES('44565','kapurubola','O+','No','834567','355908');
INSERT INTO Regular_Donor VALUES('67548','sudu menika','AB-','Yes','678401','355200');
```

	OCID	Ha_Name	Ha_Phonenumber	professional_availability	Ha_Email
1	295671	Binura Akash	0763451209	Monday	akashdissanayaka@gmail.com
2	450567	Shawn Mendis	0775623410	Monday	shawn123@gmail.com
3	678567	Pamisha peiris	0754693021	Wednesday	pamishaasiri@gmail.com

Query executed successfully. DESKTOP-L9L88I7\SQLEXPRESS ... DESKTOP-L9L88I7\hp (58) Blood-donation 00:00:00 5 rows

Screen 2

```sql
SELECT * FROM Reception

INSERT INTO Regular_Donor VALUES('44565','ijas ahmed','O+','No','834567','355908');
INSERT INTO Regular_Donor VALUES('67548','sadumina','AB-','Yes','678401','355200');
INSERT INTO Regular_Donor VALUES('98567','nimasha','AB+','Yes','209500','340077');
INSERT INTO Regular_Donor VALUES('12754','thanushi','O-','No','209500','320078');
INSERT INTO Regular_Donor VALUES('89566','dananjana','B','No','209500','340100');

SELECT * FROM Regular_Donor

DELETE FROM Regular_Donor WHERE Donor_id='12754';
DELETE FROM Regular_Donor WHERE Donor_id='44565';
DELETE FROM Regular_Donor WHERE Donor_id='67548';
DELETE FROM Regular_Donor WHERE Donor_id='89566'
DELETE FROM Regular_Donor WHERE Donor_id='98567';;
```

	Donor_id	Donor_name	Blood_type	Medical_history	BRID	staff_ID
1	12754	thanushi	O-	No	209500	320078
2	44565	ijas ahmed	O+	No	834567	355908
3	67548	sadumina	AB-	Yes	678401	355200
4	89566	dananjana	B	No	209500	340100
5	98567	nimasha	AB+	Yes	209500	340077

Query executed successfully. DESKTOP-L9L88I7\SQLEXPRESS ... DESKTOP-L9L88I7\hp (58) Blood-donation 00:00:00 5 rows

Performance Considerations

- The system must be available 24 hours a day for a user to access the system.
- at any time without any obligations.
- Management (manager, advocate) can manage the eligible donors and recipients.
- Manager has access to track past donor interactions.
- Managers can manage system policies.
- health care professional can view donor and recipient profiles and records.
- The system should be able to handle a growing number of policies, records,
- and users without significant degradation in performance.
- System needs to integrate with external systems such as
- databases.

Security Requirements

- Ensure security measurements of the website.
- Restricted access for sensitive data in the system.
- Using cloud storage to ensure secure data backups and easy recovery.
- Have up-to-date antivirus software for every system.
- Have administrative accounts for advocates to easily manage data.
- Make eligibility criteria accessible only on need-to-know
- strategy.
- Record login time and logout time of each system user.
- Have secured login system for blood donation t system.
- Regular analysis of security measures of the system.
- Maintain security policies for the system and address them to all users.
- Use a strong password management system.
- Make exclusive members only pages hidden from public view.
- Use two factor authentication.
- Be cautious about unusual activities in the system.
- Using high level data encryption methods to prevent data breaching.