BLOOD DONATION MANAGEMENT SYSTEM

### A PROJECT REPORT

Submitted in partial fulfillment of the requirements for the award of the degree of

### Bachelor of Technology

*In*

### COMPUTER SCIENCE AND ENGINEERING

###### BY

###### KANTUBHUKTA HEMANTH Regd No - 19331A0573

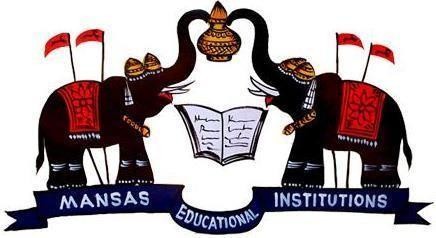
###### MD. AQUIB AKTHAR Regd No - 19331A05A6

###### M AKSHAYA Regd No - 19331A05A9

###### M KRANTHI Regd No - 19331A05B0

###### P KISHORE Regd No - 19331A05C8

**Under the Supervision of** Mr.R.Ravi Kumar **Assistant Professor**



**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING MVGR COLLEGE OF ENGINEERING (Autonomous) VIZIANAGARAM-535005, AP (INDIA)**

###### (Accredited by NBA, NAAC, and Permanently Affiliated to Jawaharlal Nehru Technological University Kakinada)

###### DECEMBER 2021

### Maharaj Vijayaram Gajapathi Raj (MVGR) College of Engineering (Autonomous)

### CERTIFICATE



This is to certify that the project report entitled “**BLOOD DONATION MANAGEMENT SYSTEM** ”is being submitted by **K.BHARATHRAM,K.VARSHINI,MOHAMMED IMRAN,N.NAVYA**

**GEETHIKA** bearing registered numbers **19331A0567, 19331A0572, 19331A05C0, 20335A0509** respectively, in partial fulfillment for the award of the degree of “**Bachelor of Technology” in Computer Science and Engineering** is a record of bonafide work done by them under my supervision during the academic year 2021-2022.

#### Dr P Ravi Kiran Varma Mr.R.Ravi Kumar

Head of the Department Assistant Professor

Dept. of CSE Dept. of CSE

## Contents

[ABSTRACT 4](#_Toc93612349)

[Design Methodology 5](#_Toc93612350)

[Entity-Relationship (ER) Diagram 5](#_Toc93612351)

[LOGICAL DATABASE DESIGN 9](#_Toc93612352)

[CONCLUSION 11](#_Toc93612353)

# ABSTRACT

### General view:-

Our project is to create and handle the database which has the collection of inter-related data about the blood donation activities. In general, various languages (eg: html,php,javascript,sql) to create an interface and interact with user effectively. We have to handle both the front-end design and the back-end design to make our system work more elegantly, effectively and properly.

### Our Idea :-

Our main idea is to handle the database using three tables. DONOR table (table which consists of donors who are ready to donate blood), RECIPIENT table(table which consists of patients(requirers) who are currently in need of blood), DONATION table(table which consists of a donor and his corresponding recipient whom he has donated the blood). We are using only the Structured Query Language(SQL) to implement the above and the application used in MySQL software. However, to design a user interface, we can create a web application or a native application using other languages.

The Main aim of Blood Bank Management DBMS project is to ease the process of blood donation and allowance at blood bank. We aim to demonstrate the use of create, read, update and delete MySQL operations through this project. This project starts by adding details of doctor. Once, registration is done doctor can add blood donor details with blood group, quantity, age etc. Now, when patient is in need of blood, doctors can easily check the blood group available and provide it to the blood receiver.

### Introduction :-

The table or Entity “Donor” has five attributes (Name, Donor id, Blood group, Address and contact number). In this table, the Donor id is considered as the primary key. The table “Recipients” has five attributes (Name, Recipient id, Blood group, Address, contact and done). Here, Recipients id is considered as primary key. Primary key is a unique identifier for a record in a relation.

The people who come up voluntarily to donate blood, their details are collected stored in the table “Donor”. Hence, we have the details of people who are ready to donate blood. The ones who are in need of blood, their details are collected and are stored in

the table “Recipient”. The time when hospital finds the person who is in “Donor” table donates blood to a person who is in “Recipient” table, then the details of donation are inserted into the ‘Donation” table and done field in Recipient table is then updated to indicate who received the blood are who are still in need of blood.

It is a binary relationship with partial participation of donors and recipients in the “Donation” table. The donors in the “Donor” table are uniquely identified with a given donor id. The recipients in the “Recipient” table are uniquely identified with a given recipient id. The donor id and recipient id function as foreign key in the “Donation” table. The combination of donor id and recipient id will be the Primary key for the “Donation” table.

# er_model.PNGDesign Methodology

# Entity-Relationship (ER) Diagram

Here, we have two entities, i.**e DONOR entity** and the **RECIPIENT entity**. We have a **DONATION** relationship existing between the above two entities.

**Representations** :-

* + 1. **Entity** – An entity is a real world object that can be easily identified in the database. Entity is represented by

Entity name

* + 1. **Relationship** – The association existing among several entities is called a Relationship. It is represented by

name

* + 1. **Attribute** – Attributes are the properties that define the entity type. These can be compared with columns in table. It is represented by

Attribute

* + 1. **Key attribute** – The attribute which uniquely identifies each entity in the entity set is called the key attribute. It is represented by

Attribute

* + 1. **Binary relationship** – When there are two entities participating in a relation, then the relationship is called a binary relationship.
    2. **Partial participation** – The entity in the entity set may or may not participate in a

relationship. It is represented by a

* + 1. **Foreign key** – It is a field or collection of fields in one table that refers to the primary key in another table. The table containing the foreign key is called child table and the table containing the primary key is called the master/parent table.

**Implementation :**

So, the detailed procedure in which these works are done are as follows :

**STEP 1** : Identifying the entities

1. Donor
2. Blood

3. Receptionist  
4. Blood Bank  
5. Blood Bank Manager  
6. Hospital

**STEP 2** : Identify the key attributes.

1. Donor – donor id(DID)  
2.Blood – code  
3.Receptionist – employee id(empid)  
4.Blood Bank–Blood Bank number(BNO)  
5.Blood Bank Manager – employee id(empid)  
6.Hospital – name, phone number(phno)

**STEP 3** : Identify other relevant attributes

1. Donor – name, age, sex, address, phone number(phno)  
2. Blood – blood type, cost  
3. Receptionist – name, address, phone number(phno)  
4. Blood Bank– issues, orders , blood type  
5. Blood Bank Manager – name ,email \_id , phone number(phno)  
6. Hospital-address

**STEP 4**: Complete the ER Diagram

**1) Requirement Analysis :**

The Requirement Analysis deals with understanding the data to be stored , the way of obtaining data through the database and measures that are taken for obtaining the better performance. The Blood Donation Management Database usually contain the details of Doctors, Patients, Staff, etc. These details are to be taken into account while preparing the database of a hospital.

**⇒ Doctors**

This section includes the list of the doctors and their details such as IDs and emergency numbers. The doctor can check his schedule and that of other doctors too. This helps a doctor to edit his schedule accordingly. The patient can be given an appointment referring to the doctor’s availability of blood required.

**⇒ Patient**

New patients can be registered in the system. An electronic medical record system is in-built which stores all the basic and medical details of the patient. One can know the room allotted to a particular patient along with their admission and discharge dates.

**⇒ Rooms**

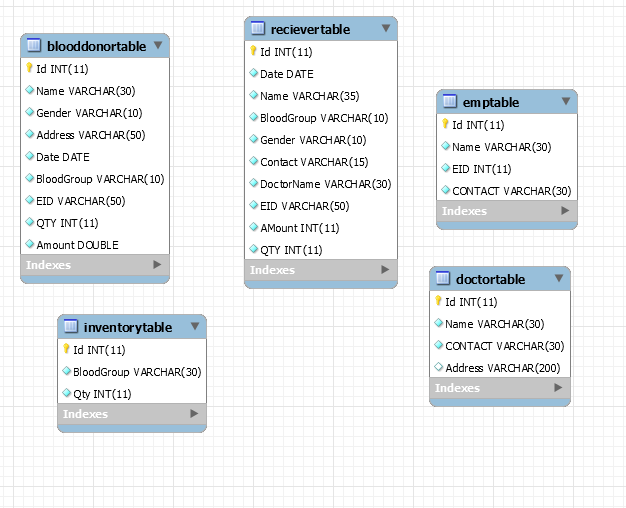
This section deals with the rooms in the hospital their allocation and one can also obtain details of a patient to whom the room is allocated.

**2) Conceptual Database Design :**

This step gives the higher-level description of data stored in the database. It should represent various fields of each relationship. So, for this ER models are used for easy understanding and implementation

This Blood Donation Management Database also dealt with the ER model diagrams to show the relationships among various entities of the database.

**3) Tabular Representation :**



###### DONOR : RECIPIENT :

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Name Did | Blood | Address | Contact | Name | Rid | Blood | Address | Contact | Done |
|  | Group |  |  |  |  | Group |  |  |  |

(Entity1) (Entity2)

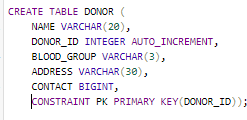
###### DONATION :

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Donor id | Recipient id | Date | Blood Donated (in ml) | Hospital |
|  |  |  |  |  |

(Relationship )

* Here, Did (Donor id) field acts as primary key for “Donor” table.
* The field Rid (Recipient id) acts as primary key for “Recipient” table.
* The DONOR table will be filled when people register themselves as a blood donor.
* The RECIPIENT table will be filled by the hospital staff or some other who post that a particular person is currently in need of blood.
* The DONATION table will be filled when blood donation takes place from a donor to recipient. At the same time, the done field in RECIPIENT table is updated to 1 which indicates which patients have received blood and which patients are still in need of blood.
* The “done” field in the “RECIPIENT” table will indicate the patient records who
* are still in need of blood and who are the ones who got blood donated.

# LOGICAL DATABASE DESIGN

* 1. *Creating Donor table :-*

CREATE TABLE DONOR ( NAME VARCHAR(20), DONOR\_ID INTEGER AUTO\_INCREMENT,

BLOOD\_GROUP VARCHAR(3), ADDRESS

VARCHAR(30), CONTACT BIGINT,

CONSTRAINT PK PRIMARY KEY(DONOR\_ID));

* 1. *Inserting data into Donor table :-*

INSERT INTO DONOR VALUES ( ‘RAM’,

1, ’A+’,

‘AKKAYYAPALEM, VISAKHAPATNAM’, 9988112255 );

INSERT INTO DONOR VALUES ( ‘RAVI’,

5, ‘O-’,

‘CLOCK TOWER, VIZIANAGARAM’, 9944556611 );

INSERT INTO DONOR VALUES ( ‘JEEVA’,

2, ‘B-’,

‘KANCHARAPALEM’, 5566771122 );

INSERT INTO DONOR VALUES ( ‘RAHUL’,

3, ‘O+’,

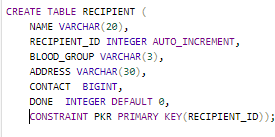
‘GANDHI ROAD, BANAGALORE’, 8899112277 );

##### donor_input.PNG

### Fetching data from Donor table :-

##### SELECT \* FROM DONOR;

##### donor_output.PNG Output :-

* 1. *Creating Recipient table :-*

CREATE TABLE RECIPIENT ( NAME VARCHAR(20), RECIPIENT\_ID INTEGER

AUTO\_INCREMENT,BLOOD\_GROUP VARCHAR(3),

ADDRESS VARCHAR(30), CONTACT BIGINT,

DONE INTEGER DEFAULT 0,

CONSTRAINT PKR PRIMARY KEY(RECIPIENT\_ID));

* 1. *Inserting data into Recipient table :-*

INSERT INTO RECIPIENT VALUES ('MOHAN', 3 ,

'B-' , 'JONNADA, VZM' , 9042123192);

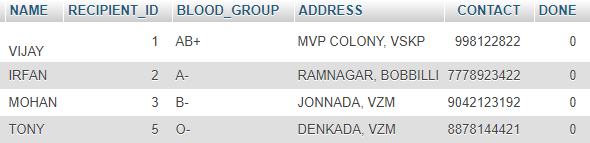
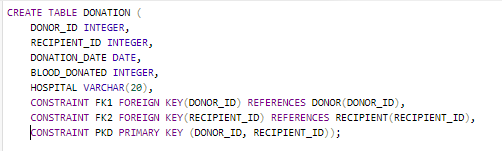
INSERT INTO RECIPIENT VALUES ('IRFAN',2 ,

'A-', 'RAMNAGAR, BOBBILLI' , 7778923422);

INSERT INTO RECIPIENT VALUES ('TONY', 5 ,

'O-' , 'DENKADA, VZM' , 8878144421)

INSERT INTO RECIPIENT VALUES ('VIJAY',1,AB 'MVP COLONY,



VSKP' ,998122822);

recipient_input.PNG

* 1. *Fetching data from Recipient table :-*

SELECT \* FROM RECIPIENT;

#####  Output :-

* 1. *Creating Donation table :-*

CREATE TABLE DONATION ( DONOR\_ID INTEGER, RECIPIENT\_ID INTEGER, DONATION\_DATE DATE, BLOOD\_DONATED INTEGER,HOSPITAL VARCHAR(20),

CONSTRAINT FK1 FOREIGN KEY(DONOR\_ID) REFERENCES DONOR(DONOR\_ID), CONSTRAINT FK2 FOREIGN KEY(RECIPIENT\_ID) REFERENCES RECIPIENT(RECIPIENT\_ID),CONSTRAINT PKD PRIMARY KEY (DONOR\_ID, RECIPIENT\_ID));

* 1. *Inserting data into Donation table :-*

INSERT INTO DONATION VALUES(2,

3,

'2020-02-20',

250,

'KGH, VSKP');

INSERT INTO DONATION VALUES (5,

5,

'2020-05-11',

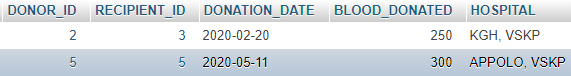
300,

'APPOLO, VSKP');

donation_input.PNG

* 1. *Fetching data from Donation table :-*

SELECT \* FROM DONATION;

 Output :-

#####  After a recipient in the “Recipient” table gets blood donated from a person in the “Donor” table, the ‘done’ field in recipient record is updated in “Recipient” table.

* 1. *Updating records from Recipient table who got blood donated :-*

UPDATE RECIPIENT SET DONE=1

WHERE RECIPIENT\_ID=(SELECT RECIPIENT\_ID FROM DONATION);

##### recipient_update.PNG

* 1. Viewing the patients who got blood donated and who are still in need of blood :-

SELECT \* FROM RECIPIENT ;

##### recipient_new.PNG Output :-

 Here, the value of 1 in “Done” field indicates that the patient in that record has already got blood donated by a person from “DONOR” table.

 The value of 0 in “Done” field indicates that the patient in that record is still in need of blood.

 Hence, we can differentiate the records of patients who are still in need of blood and who are the ones who got blood donated.

# CONCLUSION

* + - We has developed this project using MYSQL for writing various queries and designing the logical database.
    - The project uses a simple Binary relationship with partial participation of “Donor” entity and “Recipient” entity.
    - The updation of “Recipient” table needs to occur as soon as a record in inserted into “Donation” table which indicates that blood donation event has occurred.
    - The “Recipient” table will let us know the patients who are still in need of blood and who are the ones who got blood donated.
    - Here, we conclude our project which is the back-end database design for Blood Donation Management System.