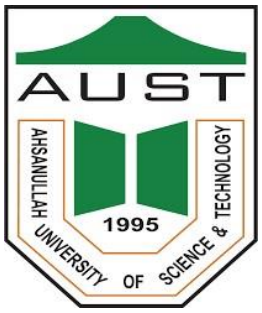
Ahsanullah University of Science & Technology

Department of Computer Science & Engineering

Semester Fall 2021



Course No: CSE 4126

Course Title: Distributed Database Systems Lab

Project Name: Blood Bank Management System

Submitted To:

Mr. G.M. Shahariar
Lecturer, CSE, AUST
Sanzana Karim Lora
Lecturer, CSE, AUST

Submitted By:

Name: S. M. Tasnimul Hasan
ID:18.02.04.142

Teammate:
Name: Nurul Amin
ID: 18.02.04.130

Introduction:

Our project 'Blood Bank Management System' is created for the blood bank to gather blood from various sources and distribute it to needy people who have high requirements for it. Almost every day people face situations where they require the blood of different groups. Using this system, a user can search for a blood group and get the contact information of the donor with the same blood group needed. The prime benefit of this system is that it can provide information on available Donors. So, using a system like this can ease the searching hassles.

Software:

- Oracle Database 10g Express Edition

Language:

- Oracle PL/SQL Procedure Language

Project Description:

We have developed our system based on Oracle PL/SQL procedure language. All the codes run in the SQL plus command prompt. As our system is based on distributed database concept here, we have used 1 Server site and 1 host site.
We have 5 tables in total for storing detailed data.

- ❖ The "Donor" table holds all the required information of a donor who has donated blood to a recipient.
- ❖ In the "Recipient" table the information of the recipients is stored.
- ❖ The "Blood Inventory" table saves the value of the bag numbers of the blood donated by a donor, hemoglobin and platelets number of that corresponding blood bag.
- ❖ In the "Donation Details" table, details of any blood donation event like the hospital at which the event occurred, the amount of blood that was received and the date when the blood was given.
- ❖ In the "Blood Group" table, the number of bags for each blood group is stored.

Global Schema:

DONOR (DID, Dname, Dage, Dgender, Dbloodgroup, Dcity, Dphnum, Deligibility)

RECIPIENT (RID, Rname, Rage, Rgender, Rbloodgroup, Rcity, Rphnum, DID)

BLOOD_INVENTORY (DID, bagnumber, heamoglobin, platelets)

DONATION_DETAILS (DID, donationnumber, hospital, amount, givenat)

BLOOD_GROUP (DID, bloodGroup, numOfBag)

Fragmentation Schema:

$DONOR_1 = SL_{DID <= 1100} DONOR$
$DONOR_2 = SL_{DID > 1100} DONOR$

$RECIPIENT_1 = SL_{RID <= 2100} RECIPIENT$
$RECIPIENT_2 = SL_{RID > 2100} RECIPIENT$

$BLOOD\_INVENTORY_1 = SL_{bagnumber <= 5100} BLOOD\_INVENTORY$
$BLOOD\_INVENTORY_2 = SL_{bagnumber > 5100} BLOOD\_INVENTORY$

$DONATION\_DETAILS_1 = SL_{donationnumber <= 7100} DONATION\_DETAILS$
$DONATION\_DETAILS_2 = SL_{donationnumber > 7100} DONATION\_DETAILS$

Functionalities:

- Insert information of donor into DONOR table.
- Delete donor from DONOR table.
- Update information of donor into DONOR table.
- Search donors from DONOR table by donor id.
- Search donors from DONOR table by blood group.
- Search donors from DONOR table by city.
- Search donor from DONOR table by the eligibility of donor.
- Count total number of bags of a specific blood group.
- Transfer blood from one site to another site

Packages and Functions:

1. Package myPack – Consists of function countBagNums
2. Function countBagNums – Consists of function countBagNums
3. Procedure transfer – Transfer blood from one site to another site.

Triggers:

1. trigInsertDonor - Trigger for donor insert.
2. trigUpdateDonor - Trigger for donor update.
3. trigDeleteDonor - Trigger for donor delete.

```
20
21  create or replace trigger trigInsertDonor
22  after insert on DONOR
23
24  declare
25
26  begin
27
28      dbms_output.put_line('Data Inserted!');
29
30  end;
31  /
32
33  commit;
```

Exception:

```
declare

    id_to_delete number;
    myExp EXCEPTION;

begin

    id_to_delete := &id;

    delete from DONOR where DID = id_to_delete;

    IF id_to_delete < 0 THEN
        RAISE myExp;
    END IF;

    EXCEPTION
        WHEN myExp THEN
            DBMS_OUTPUT.PUT_LINE('ID Cannot be Negative!');
        WHEN OTHERS THEN
            DBMS_OUTPUT.PUT_LINE('Others Errors!');

end;
/
```

Contribution:

- Insertion of Donor
- Insertion of Recipient
- Trigger for Donor insert, update and delete.
- Cursor for Search Donor by ID & Search Donor by blood.
- Exception for donor delete.
- Procedure for transfer of blood.

Conclusion:

Finally, it can be concluded that we are able to create a “Blood Bank Management System”. By using this system searching for available blood becomes easy and saves a lot of time. This system allows us to insert, update, delete & search the information. This is very helpful management system for blood recipients.