PROJECT NAME- Blood Bank Management System

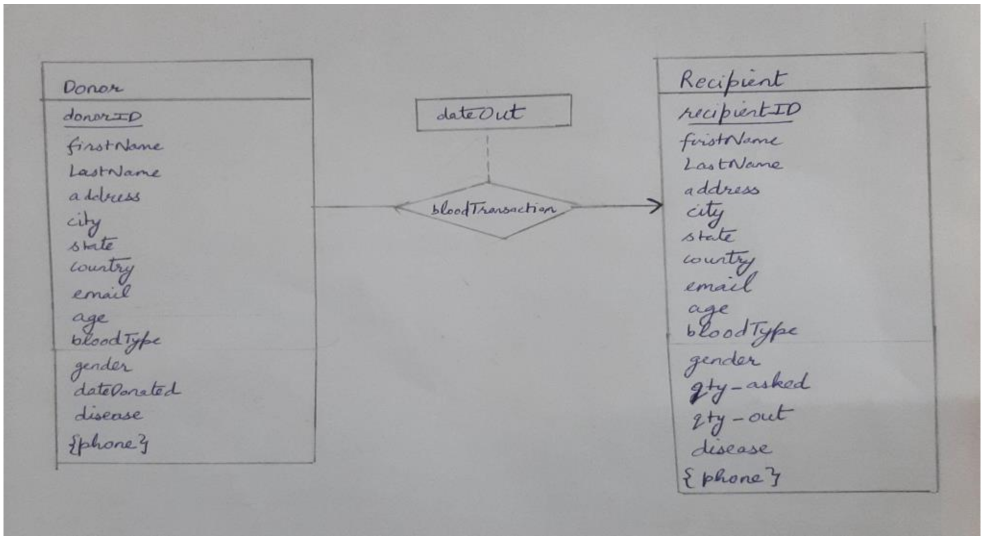
1a. Data Requirements:

- The age limit to donate is kept between 18 and 65.
- The donor would only be allowed to donate iff disease=0(No disease).
- The phone numbers for donors and recipients can be multi-valued.
- **At a time, a donor can donate only 1 unit of blood. This also means that if the same donor would donate again, he will be assigned a different donorID.**
- **Matching can be made based on either only blood type or based on blood type, city, state, and country**.
- If any donor or recipient has multiple phone numbers, they would still have the same donorID or recipient respectively.

Functional Requirements:

- As per the attributes, the only functional dependencies we have are the primary key constraint.

2a. ER DIAGRAM- SEE BELOW



- ❖ The cardinality is N:1.
- ❖ Both the entities have partial participation in the bloodTransaction relationship.

2b. SCHEMA DESIGN:

- ENTITIES:
    1. DONOR: donorID, firstName, lastName, address, city, state, country, email, age, bloodType, gender, dateDonated, disease, phone(multi-valued).
    2. RECIPIENT: recipientID, firstName, lastName, address, city, state, country, email, age, bloodType, gender, qty_asked, qty_out, disease, phone(multi-valued).
- RELATIONSHIPS:
    1. Bloodtransaction: donorID, recipientID, dateOut.

❖ The only Functional Dependencies we have here is the primary key constraint.

2c. DATA NORMALIZATION:

- **1NF:**
    1. DONOR: donorID, firstName, lastName, address, city, state, country, email, age, bloodType, gender, dateDonated, disease.
    2. Donor_phone: donorID, phone.
    3. RECIPIENT: recipientID, firstName, lastName, address, city, state, country, email, age, bloodType, gender, qty_asked, qty_out, disease.
    4. Recipient_phone: recipientID, phone.
    5. Bloodtransaction: donorID, recipientID, dateOut.
- **2NF:**
    1. DONOR: donorID, firstName, lastName, address, city, state, country, email, age, bloodType, gender, dateDonated, disease.
    2. Donor_phone: donorID, phone.
    3. RECIPIENT: recipientID, firstName, lastName, address, city, state, country, email, age, bloodType, gender, qty_asked, qty_out, disease.
    4. Recipient_phone: recipientID, phone.
    5. Bloodtransaction: donorID, recipientID, dateOut.

  The above form(1NF) is itself in 2NF since for the schemas 1,3,5 we have only one primary key and for schemas 2,4 we do not have any non-prime attribute.
- **3NF:**
    1. DONOR: donorID, firstName, lastName, address, city, state, country, email, age, bloodType, gender, dateDonated, disease.
    2. Donor_phone: donorID, phone.
    3. RECIPIENT: recipientID, firstName, lastName, address, city, state, country, email, age, bloodType, gender, qty_asked, qty_out, disease.
    4. Recipient_phone: recipientID, phone.
    5. Bloodtransaction: donorID, recipientID, dateOut.

The above form(1NF) is also in 3NF itself and the same reasoning goes for 3NF, as no non-prime attribute is dependent on the non-prime attribute(i.e No transitive dependency exists).

2d. List of tables required:

1. Bloodstock (VIEW): Blood Type, In Stock
   This view would show the stock of various blood types in the blood bank.
2. Bloodtransaction: This would show all the blood transactions that occurred till now between the donors and the recipients.
3. Donor: This would show the details of all the donors associated with the blood bank.
4. Donor_available (VIEW): donorID, firstName, lastName, address, city, state, country, email, age, bloodType, gender, dateDonated, disease.
   This view shows the table for the donor yet to be matched.
5. Donor_phone: This table shows all of the phone numbers associated with each of the donors.
6. Donor list (VIEW): donorID, firstName, lastName, address, city, state, country, email, age, bloodType, gender, dateDonated, disease, phone.
   This view would showcase the complete details(including phone numbers) of all the donors associated with the blood bank.
7. Recipient: This would show the details of all the recipients associated with the blood bank.
8. Recipient_pending (VIEW): recipientID, firstName, lastName, address, city, state, country, email, age, bloodType, gender, qty_asked, qty_out, disease.
   This view would show the recipients whose requests are not yet fulfilled.
9. Recipient_phone: This table shows all of the phone numbers associated with each of the recipients.
10. recipient _resolved (VIEW): recipientID, firstName, lastName, address, city, state, country, email, age, bloodType, gender, qty_asked, qty_out, disease.
    This view would show the recipient whose requests are fulfilled by the blood bank.
11. recipient list (VIEW): recipientID, firstName, lastName, address, city, state, country, email, age, bloodType, gender, qty_asked, qty_out, disease, phone.
    This view would showcase the complete details(including phone numbers) of all the recipients associated with the blood bank.

2e. Additional components

- **Procedures:**
  1. Balance_blood_transaction(recipientID int): This procedure finds the match for the input recipientID and output all the transaction(if occurred any due to this call) of the given recipientID in the current date. The donor matching would be based on the blood type only.
  2. Balance_blood_transaction_all(recipientID int): The only difference in this procedure is(as compared to the above), the donor matching would be based on- blood type, city, state, and country.

3. Days_after_donation(donorID int): This would showcase the number of days(till the current date) after the blood was donated for the input donorID.
4. Delete_donor_phone(donorID int, phone varchar(20)): This would delete the entry corresponding to the input phone number for the input donorID iff the input donorID has more than 1 phone number.
5. Delete_recipient_phone(recipientID int, phone varchar(20)): Same as above.
6. Donor_available_bloodtype(bloodType varchar(3)): This would showcase the number of donors available for the input bloodType.
7. Donors_by_date(dateDonated date): This would showcase the list of donors who donated on the dateDonated (input).
8. Donors_of_recipient(recipientID int): This would show the list of donors(matched till now) for the given input recipientID.
9. Recipient_by_date(dateOut date): This would showcase the list of recipients who received the blood on the dateOut(input).
10. Recipient_of_donor(donorID int): This would showcase the matched recipient for the given donorID.
11. Recipient_pending_bloodtype(bloodType varchar(3)): This would showcase the recipients yet with pending request with the given input bloodType.
12. Update_donor_credential(donorID int, email varchar(100),oldphone varchar(20),newphone varchar(20)):
    This would would showcase the updated details of the input donorID.
13. update_recipient_credentials(recipientID int, qty_asked int, disease int, address varchar(60), city varchar(30), state varchar(30), country varchar(30),email varchar(100),oldphone varchar(20),newphone varchar(20)):
    This would would showcase the updated details of the input recipientID.