

Blood Donation

Database Management System

Group 12 - Report

Fedija Ben Salem 210302153

Shaima Salameh 210302172

Dželila Husić 210302146

**Contents**

Introduction …............................................................................................................................... 3

ER model ….................................................................................................................................. 4

EER model …............................................................................................................................... 5

UML model ……………………………………………………………………………………...6

SQL implementation …................................................................................................................. 7

SQL – creating tables………………………………………………………………………….... 8

Table for patients ……………………………………………………………………...... 8

Table for hospital ……………………………………………………………………...... 9

Table for donors …………………………………………………………………….......10

Table for blood bank …………………………………………………………….…...... 11

Table for blood ……………………………………………………………………........ 12

Table for bloodstock ………………………………...………………...…………….… 13

Queries ….................................................................................................................................... 14

Referential integrity ………………………………………………………...…………………. 15

Transaction ……………………………………………………………………..……………... 16

SQL- views ……………………………………………….……………………...……………. 17

**Introduction**

As we all know blood donation is important because it can save somebody’s life or help those in need, that is why we decided to make a blood donation database system. The system might be a huge aid to the hospitals and blood donation centers to be on track. The system keeps track of patient data, blood requirements, donor data, blood type, quantity and quality of the blood, and anything else that doctors need to monitor.

Hospitals, blood donation facilities, and other healthcare organizations that require blood can use the system. The system may also include tools by which it notifies donors when blood is needed and reminds them to donate since there is a constant need for blood.

The blood donation database system will cover donors' information, bloodstock and bank, hospital and patients' information, and their blood needs. The system will show the hospitals which blood group is available in which blood bank, which will also save them time.

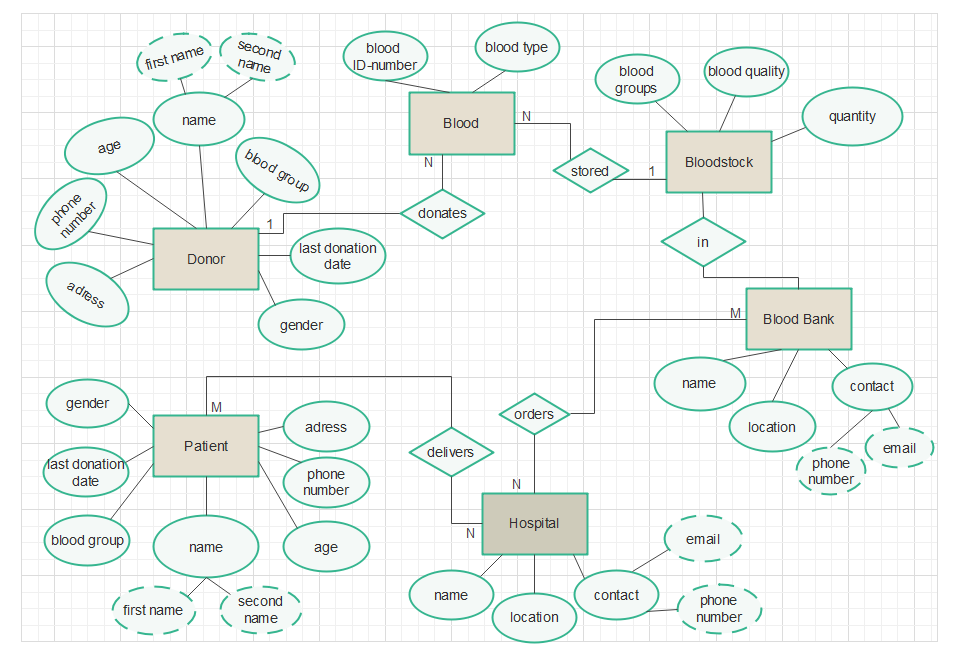
This project will concentrate on creating a system for blood donation utilizing different diagrams and SQL. To identify the key entities in the system, we will begin by developing an Entity-Relationship (ER) diagram. With SQL we are going to build the necessary tables for entities, to store all of the data.

**ER-diagram (Entity Relationship diagram):**

An Entity relationship diagram is a graphic description of the connections between different data system elements. An ER diagram can be used to represent the connections between donors, patients, blood banks, bloodstock, and hospitals in the context of a blood donation database system.

A blood donation database system's goal is to successfully manage the entire blood donation process, from the moment a donor arrives for a donation to the moment the blood is used in a patient's transfusion. An idea for creating a database system that can manage all the information required for the blood donation procedure is provided by this ER diagram.

The entities and relationships identified in the ER diagram are important because they help to define the scope of the database system and the types of queries that the system will be able to support. For example, a blood bank could use this database system to track the availability of different blood types and to manage their blood stock levels accordingly.

Overall, this ER diagram provides a useful starting point for designing a comprehensive blood donation database system that can help to streamline the blood donation process, ensure the safety of donors and patients, and efficiently manage the storage and distribution of bloodstock.

**EER- diagram (The Enhanced Entity Relationship diagram)**

An addition to the ER model is the Enhanced Entity Relationship model or EER (Enhanced Entity Relationship) model. In addition to the three new extra semantic data modeling ideas of specialization/generalization, classification, and aggregation, it incorporates all the modeling concepts (entity types, attribute types, relationship types) of the ER model. In the subsections that follow, we go into further depth on each of them. An effective model for a blood donation system should incorporate various components and functionalities to ensure a smooth and efficient process. In order to show how various entities, interact and share information inside the system, relationships are developed. A picture containing sketch, line art, white, diagram

Description automatically generated

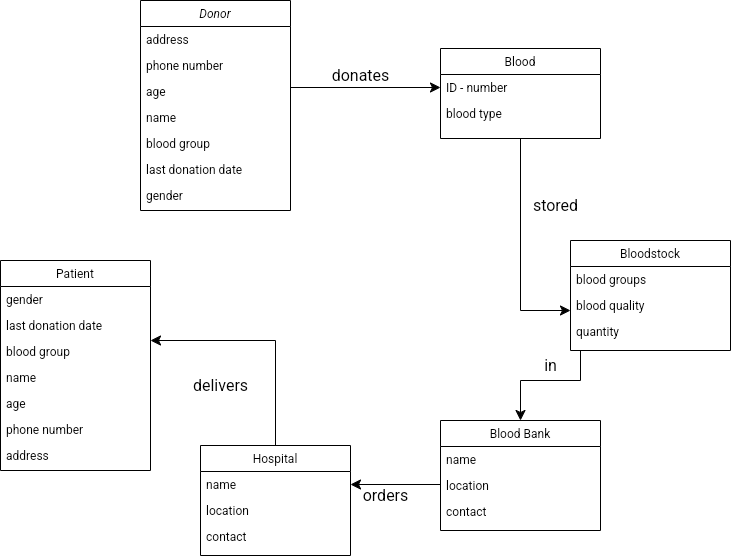
**UML – diagram (Unified Modeling Language)**

A UML diagram is a diagram based on the Unified Modeling Language with the purpose of visually representing a system along with its main actors, roles, actions, artifacts or classes, in order to better understand, alter, maintain, or document information about the system.

Blood donation UML diagram can be used to analyse blood donation software, or to plan blood donation software development. This diagram would be very useful to visualise blood donation software development idea.

The fact that UML diagrams are made in a standardised way makes it easier to understand them. It is easy to understand who are the “actors” in this blood donation system, and it doesn’t matter if you are developing a software or making a database system.

Entities and relations in this diagram are the same as the entities in ER diagram, and it provides information about the system and the functions in it.



**MySQL implementation**

In this part of our report, we will be presenting the defining database structure (DDL) part of SQL. We have created 6 tables in our Blood donation database management system.

1. Patients

2. Hospital

3. Donor

4. Blood bank

5. Blood

6. Bloodstock

Tables are basic units of the database, and they consist of columns and rows. In these tables for each column, we have to create variables, and for each variable, we must assign a data type. When defining a variable, it is important to declare its nullability property since if we remain with the SQL default, it may result in various problems. Also, at each table, we should assign one or more primary keys. To track the most important variable in the table, we use the primary key. For instance, in our table the primary key is blood group, it is a unique way to identify the row in a table.

**SQL** **– creating tables**

**TABLE FOR PATIENTS:**

* As you can see in the picture belove, the table patient has 8 rows which consist of ID, name, surname, gender, contact, birth date, blood group, and last donation date.

A screenshot of a computer

Description automatically generated with medium confidenceIn this table our primary key is ID, so ID cannot contain NULL values. The ID, patient contact, birth date, and last donation day are of type int and will hold an integer, while others are of type varchar and will hold characters, and the maximum length of these characters is going to be 100.

* **A screenshot of a computer

  Description automatically generated with low confidenceA screenshot of a computer

  Description automatically generatedAfter creating a table, the following SQL statement inserts data into the table:**
* **The result is:**

**TABLE FOR HOSPITAL:**

- Next table is the hospital and it comprises three rows including hospital ID, hospital name, and address. ID is of type integer and serves as the table’s primary key, while hospital name and address are of type varchar and the maximum of characters is 100.

A picture containing text, font, screenshot, line

Description automatically generated

* **Inserting data to the table (we specify both the column names and values to be inserted):**

A picture containing text, font, screenshot

Description automatically generated

* **The result is:**

A screenshot of a computer

Description automatically generated with medium confidence

**TABLE FOR DONORS:**

- There are seven columns in the donor table: name, surname, gender, birthdate, blood type, address, and last donation date. The date of birth and last donation date is of type integer, whereas all others are of type varchar, with a character limit of 100. The blood group cannot be null, since it is the primary key of the table.

A picture containing text, screenshot, font

Description automatically generated

* **A screenshot of a computer

  Description automatically generated with low confidenceInserting data in the table:**

* **A screenshot of a computer

  Description automatically generated with medium confidenceThe result is:**

**TABLE FOR BLOOD BANK**

- Blood bank name, address, and contact information are listed in three rows of the table. The name and address of the blood bank are of the varchar type, with a character limit of 100; contact information is of the integer type. In this table, the primary key is the name of the blood bank.

A picture containing text, screenshot, font, line

Description automatically generated

A picture containing text, font, screenshot

Description automatically generated

* **Inserting data in the table:**
* **The result is:**

A screenshot of a computer

Description automatically generated

**TABLE FOR BLOOD**

* Blood table consists of only 2 rows, blood ID and blood type. Blood ID is of type integer and primary key, while blood type is of type varchar, with a maximum of 100 characters. Blood ID is also a primary key of the table, so it cannot be null.

A picture containing text, font, screenshot, line

Description automatically generated

* **Inserting data in the table:**

A screenshot of a computer program

Description automatically generated with low confidence

* **The result is:**

A screenshot of a computer

Description automatically generated

**TABLE FOR BLOODSTOCK**

* The last table is bloodstock, which consists of 3 rows: blood quantity, type, and quality. Blood type is the primary key, it cannot be null, and it is of type varchar, blood quality is also of type varchar with a limit of 100 characters. Blood quantity is of type integer.

A picture containing text, font, screenshot, line

Description automatically generated

* **Inserting data in the table:**

A screenshot of a computer

Description automatically generated with medium confidence

* **The result is:**

A screenshot of a computer

Description automatically generated with medium confidence

# **SQL QUERIES**

* A database table or group of tables may be requested for data or information via a query. It can be either a select query or an action query in the context of database queries. An action query, on the other hand, requests extra actions on the data, such as insertion, updating, or deletion. While a select query, as its name implies, chooses data, and gets it.

Here we have 2 examples of select queries:

A picture containing text, font, screenshot, algebra

Description automatically generated

In this example, we select a blood quantity bigger than 25 from the bloodstock table.

And here is the result:

A screenshot of a computer

Description automatically generated with medium confidence

Our next example selects blood group “A+” from the patient table:

A picture containing text, font, screenshot

Description automatically generated

Since there is no “A+” in this table, the result is null:

# A picture containing text, font, screenshot, rectangle Description automatically generated

# **Referential integrity**

Referential integrity refers to the connection between tables. Each table in the database has a primary key, this same primary key can also appear in other tables because of its relationship to data within those tables. When the same primary keys are found in tables, it is called a foreign key. Foreign keys join tables together and make them dependent on each other. For example, deleting a row that also serves as a primary key render any subsequent rows that use that value as a foreign key meaningless.

A picture containing text, font, screenshot

Description automatically generated

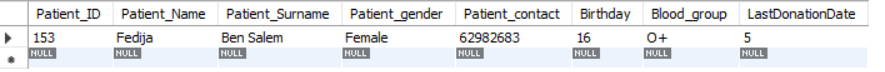
## **Transaction**

A transaction is a logical unit of work, when you are creating, updating, or deleting a variable in a table it is known as a transaction performing. The transaction has 4 standard features, short form ACID. A is for atomicity, ensures that all tasks are done, if it is not done then the transaction fails. C for consistency ensures that database changes. I for isolation, which helps transactions to be independent, and D for durability, ensure that results remain even in case of failure. The transaction has 3 main commands of the transaction are DELETE, INSERT and UPDATE.

A picture containing text, font, screenshot

Description automatically generatedHere is one example of a DELETE command:

In this case, we wanted to delete a patient ID that is bigger than 170, from the patient table. In the result ID bigger than 170 was deleted and only one is left:



**SQL- views**

A view is a virtual table based on a query result. It contains everything just like a table. It contains fields from one or more tables, it can also be extended with SQL functions. It shows the data as if it all were on one single table. A view is created with the INSERT VIEW command. It also has two more commands crate or replace and drop.

This view is specified for helping the blood banks to know immediately the name and ID of the hospitals which need blood for their patients.

A screenshot of a computer

Description automatically generated with low confidence

The result is:

A screenshot of a computer

Description automatically generated with medium confidence

A screenshot of a computer

Description automatically generated with medium confidenceThis view is specified to help the blood banks also to have the information about types of blood and their quantity in their bloodstock.

A screenshot of a computer

Description automatically generated with medium confidenceThe result is: