+*ANUDIP FOUNDATION*

A Project Report on

BLOOD BANK MANAGEMENT SYSTEM

By

Batch: ANP-D0453

Student ID: AF0477433

Name: Pratibha Sindkhed

Under the Guidance of

Mrs. Rajshri Chandrabhan Thete

BLOOD BANK MANAGEMENT SYSTEM

Introducing our Java-based Blood Bank Management System (BBMS):
Efficiently manage and oversee every aspect of blood donation and transfusion with our intuitive system. From maintaining donor records and tracking blood inventory to handling requests and ensuring timely distribution, our BBMS empowers healthcare facilities to streamline operations and save lives with efficiency and precision.

Entities

- ❖ Donor
- ❖ Blood
- ❖ Patient
- ❖ Employee Team
- ❖ Hospital
- ❖ Admin
- ❖ Blood Bank

ATTRIBUTES OF ENTITIES:

1. **Donor**
   - **Primary Key: Donor_ID**
   - **Attributes: Donor_name, Donor_Gender, Contact_No, DateOfDonation, BloodGroup, Donor_Address, Age**

2. **Blood**
   - **Primary Key: Blood_ID**
   - **Attributes: BloodGroup**

3. **Patient**
   - **Primary Key: Patient_ID**
   - **Attributes: Patient_name, Patient_Gender, Contact_No, Patient_Address, BloodGroup, DateOfIntake**

4. **EmployeeTeam**
   - **Primary Key: Emp_ID**
   - **Attributes: Emp_name**

5. **Hospital**
   - **Primary Key: Hospital_ID**
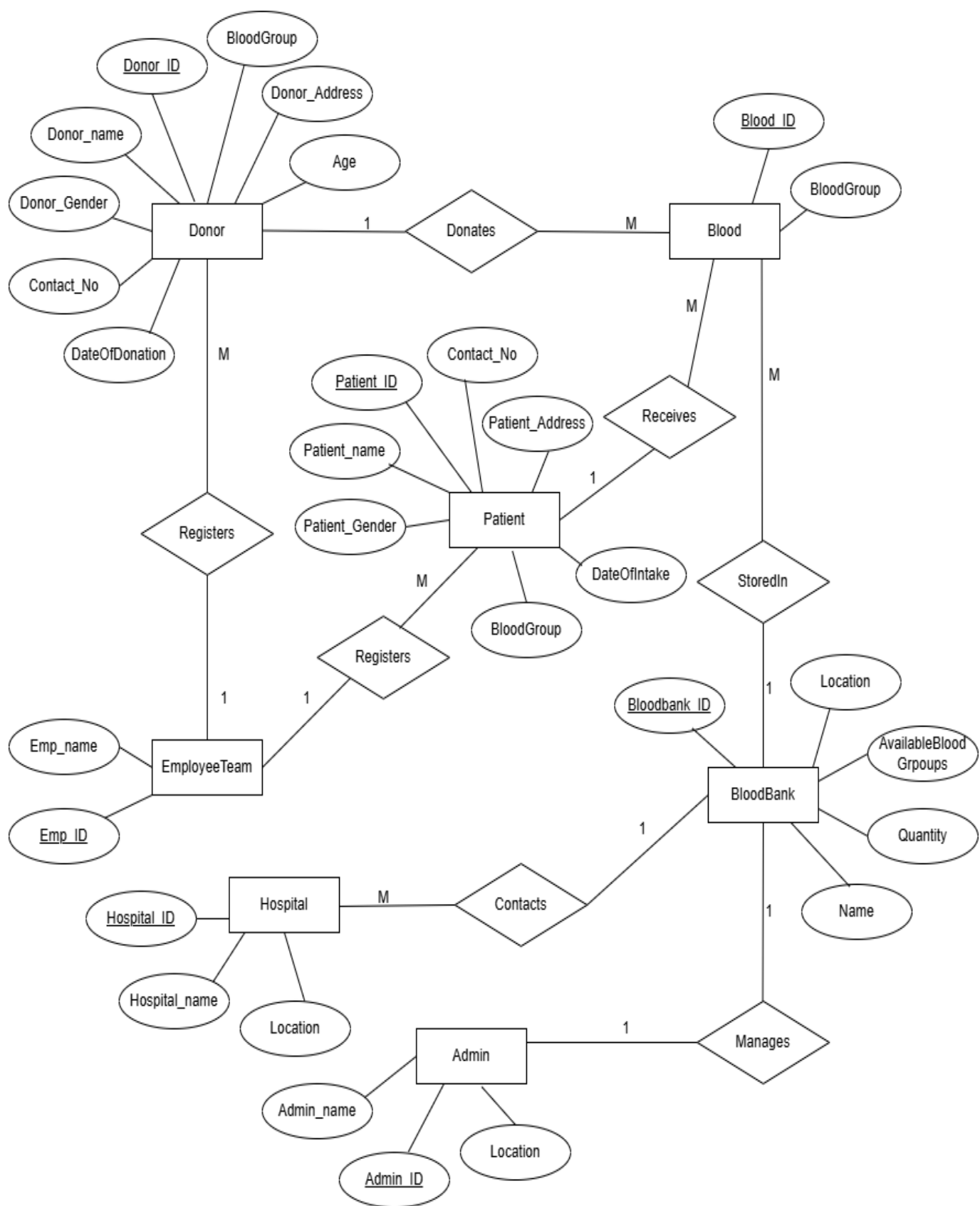   - **Attributes: Hospital_name, Location**

6. **Admin**
   - **Primary Key: Admin_ID**
   - **Attributes: Admin_name, Location**

7. **BloodBank**
   - **Primary Key: Bloodbank_ID**
   - **Attributes: Location, AvailableBloodGroups, Quantity, Name**

ENTITY RELATIONSHIP DIAGRAM – BLOOD BANK

MANAGEMENT SYSTEM

Entity-Relationship Diagram for Blood Bank Management System:

Donor (entity) with attributes: Donor_ID (key), BloodGroup, Donor_Address, Age, Donor_name, Donor_Gender, Contact_No, DateOfDonation

Blood (entity) with attributes: Blood_ID (key), BloodGroup

Patient (entity) with attributes: Patient_ID (key), Contact_No, Patient_Address, Patient_name, Patient_Gender, DateOfIntake, BloodGroup

EmployeeTeam (entity) with attributes: Emp_name, Emp_ID (key)

Hospital (entity) with attributes: Hospital_ID (key), Hospital_name, Location

Admin (entity) with attributes: Admin_name, Admin_ID (key), Location

BloodBank (entity) with attributes: Bloodbank_ID (key), Location, AvailableBlood Grpoups, Quantity, Name

Relationships:
- Donor 1 — Donates — M Blood
- Blood M — Receives — 1 Patient
- Donor M — Registers — 1 EmployeeTeam
- Patient M — Registers — 1 EmployeeTeam
- Blood M — StoredIn — 1 BloodBank
- Hospital M — Contacts — 1 BloodBank
- Admin 1 — Manages — 1 BloodBank

CONCLUSION:

       In summary, a Blood Bank Management System (BBMS) provides a comprehensive solution designed to optimize the management of blood donations, inventory, and distribution. By integrating crucial processes such as donor registration, blood tracking, and hospital requests, the system enhances efficiency, accuracy, and transparency in blood bank operations. It fosters seamless communication between donors, hospitals, and blood banks, ensuring the timely availability of safe and compatible blood for patients in need. Additionally, by offering real-time access to critical data, including blood stock levels, donor eligibility, and transfusion history, the system empowers administrators to make informed decisions that improve service quality and patient care. Overall, a BBMS plays a crucial role in modernizing blood bank operations, enhancing reliability, and ultimately contributing to life-saving efforts.

DATABASE CREATION QUERY:

1. Donor Table

CREATE TABLE Donor (

```sql
    Donor_ID INT PRIMARY KEY,

    Donor_name VARCHAR(100),

    Donor_Gender VARCHAR(10),

    Contact_No VARCHAR(15),

    DateOfDonation DATE,

    BloodGroup VARCHAR(5),

    Donor_Address TEXT,

    Age INT
);
```

2. Patient Table

```sql
CREATE TABLE Patient (

    Patient_ID INT PRIMARY KEY,

    Patient_name VARCHAR(100),

    Patient_Gender VARCHAR(10),

    Contact_No VARCHAR(15),

    Patient_Address TEXT,

    BloodGroup VARCHAR(5),

    DateOfIntake DATE
);
```

3. Blood Table

```sql
CREATE TABLE Blood (

    Blood_ID INT PRIMARY KEY,

    BloodGroup VARCHAR(5)
);
```

4. Blood Bank Table

```sql
CREATE TABLE BloodBank (
```

```
    Bloodbank_ID INT PRIMARY KEY,

    Location TEXT,

    AvailableBloodGroups VARCHAR(50),

    Quantity INT,

    Name VARCHAR(100)

);
```

5. Employee Team Table

```
CREATE TABLE EmployeeTeam (

    team_id INT AUTO_INCREMENT PRIMARY KEY,

    team_name VARCHAR(100) NOT NULL,

    bank_id INT,

    FOREIGN KEY (bank_id) REFERENCES BloodBank(bank_id) ON DELETE CASCADE

);
```

6. Hospital Table

```
CREATE TABLE Hospital (

    Hospital_ID INT PRIMARY KEY,

    Hospital_name VARCHAR(100),

    Location TEXT

);
```

7. Admin Table

```
CREATE TABLE Admin (

    Admin_ID INT PRIMARY KEY,

    Admin_name VARCHAR(100),

    Location TEXT

);
```