BANGLADESH UNIVERSITY OF PROFESSIONALS (BUP)

UML Class Diagram

Project Title: Blood Donation Portal

	Student Name	Student ID
1.	Morium Chowdhury	2252421008
	Sadia Iffat	2252421012
	Rifah Zakiah	2252421024
	Hamza Bin Arif	2252421032
	Tafsir Un Nahiyan	2252421094
2.	Group:	2
3.	Name of the Department/Faculty:	Faculty of Science and Technology (FST), Department of Computer Science and Engineering (CSE)
4.	Name of the University:	Bangladesh University of Professionals
5.	Title of the Project:	Blood Donation Portal
6.	Course title:	Software Engineering Laboratory
7.	Course code:	3206

UML Class Diagram:

The class diagram for **Blood Donation Portal** represents a well-organized object-oriented structure that clearly defines how different users and entities interact within the system.

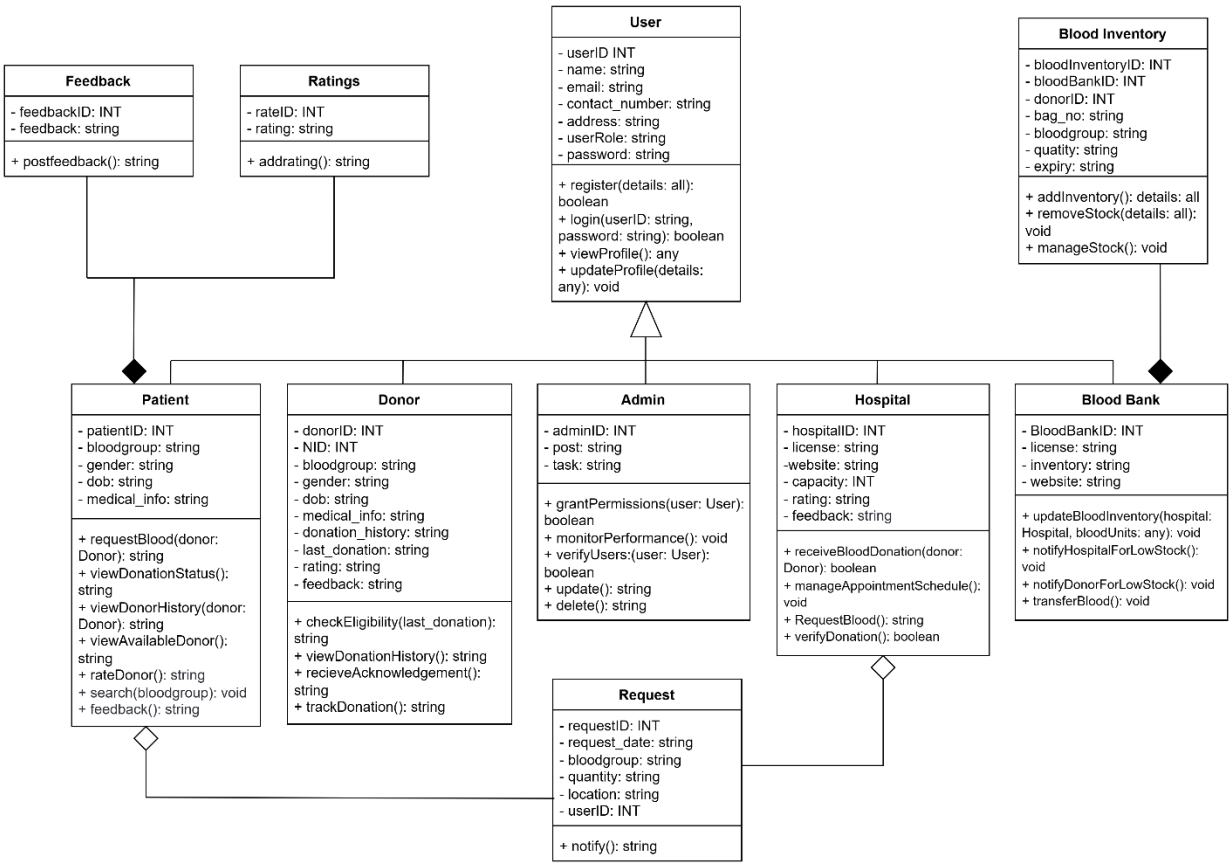

Fig. UML Class Diagram

At the core of this system is the **User class**, serving as the abstract base for all user types. This class defines the core user attributes and operations that are common across all roles—such as user_id, name, email, password, phone, address, gender, age, and blood_group. It also encapsulates essential operations like login(), register(), and update_profile().

From this generalized user class, five distinct user roles inherit functionality:

1. **Donor**
2. **Patient**
3. **Admin**
4. **Hospital**

5. **Blood Bank**

This inheritance setup ensures a consistent structure while allowing each subclass to extend or specialize its behavior.

Key Specialized Classes and Their Roles

Donor

The Donor class inherits from User and represents individuals who donate blood. Donors can track their donation, get acknowledgement for their services, and get ratings based on their services.

Patient

The Patient class, also inheriting from User, signifies individuals requesting blood. Patients have the ability to perform operations such as searching for donors or hospitals using the method search (bloodgroup: string): void. Patients are connected to both the Feedback and Ratings classes. This reflects that donors can leave feedback after donations and can also be rated based on their responsiveness, punctuality, or donation history. This creates a reputation system that fosters trust and reliability.

Admin

Admins oversee the system and are empowered with elevated privileges, allowing them to manage users, validate data, monitor requests, and resolve issues. Though not explicitly defined with methods or attributes in the diagram, the Admin class's role is central to maintaining system integrity.

Hospital

The Hospital class represents registered medical institutions. These can generate blood Requests and receive blood donations from donors. Hospitals, being part of the user hierarchy, also share attributes like contact details and blood group compatibility information.

Blood Bank

Blood banks are also modeled as users but have a unique association with the **Blood Inventory** class. This class likely holds data about available blood types, quantities, and expiry dates—critical for ensuring efficient blood matching and minimizing wastage. The diagram connects Blood Bank with Blood Inventory using composition, indicating a strong ownership relationship.

Associated Classes

Request

The Request class is associated with both Patient and Hospital, and potentially even Donor or Admin. It includes details such as blood type, urgency, request status, and the requester's identity.

Requests are fundamental to the system, enabling the communication of needs and triggering action by donors or institutions.

Feedback

Feedback is captured from patients, containing comments on their experience, satisfaction, or suggestions. This allows for continuous improvement of services and transparency in operations.

Ratings

The Ratings class extends the feedback mechanism by offering a quantifiable evaluation of donors (and possibly hospitals). This could be implemented as a numerical score or a star-based system, enhancing donor credibility and performance tracking, currently just set as a string input.