University of Gondar

College of informatics

Department of information science

Web Based Blood Bank Management System

By

Group Members	**Id**
Samrawit Addis	GUR/40728/13
Natnael Dagnachew	GUR/23689/13
Samuel Abebe	GUR/40590/13
Adane Deneke	GUR/40670/13
Abel Sitotaw	GUR/40524/13

Gondar, Ethiopia

Date/Year: May,2024

Approval Sheet

This Group Project entitled “Web Based Blood Management System” has been read and approved as meeting the preliminary project requirements of the Department of Information Science in partial fulfillment for the award of a Bachelor of Science degree in Information Science, University of Gondar, Gondar, Ethiopia.

Approved by:

1. Name of Advisor: Prof. Assefa Signature:

2. Name of Advisor: Mr. Belay Signature:

3. Name of Project Coordinator: Mr. Cherenet Signature:

Date: 23/08/2015

ACKNOWLEDGMENT

Above everyone and anything else, we are grateful to our Almighty God for granting us Courage, strength, And Knowledge to complete this project

We want to sincerely thank and appreciate Prof. Assefa And Mr. Belay our advisors, for allowing us to work on this project. They have provided us with support, direction and helpful guidance.

We are also appreciative of Mr. Cherenet's leadership and direction in getting this project finished.

Finally, we would like to express our gratitude to our parents and friends for their invaluable advice and support throughout the course of the project.

Table Of Contents

List of figures

List of Table

ACRONYMS

FMOH: …….. Federal Ministry of Health

CDC: ……Central Disease Control and Prevention

ERCS: ……Ethiopia Red Cross Society

OOSD: ……Object Oriented System Development Methodology

REG.NO: …… Registration Number

ABO: ……..Blood Group

Exp.: ……,,,,,,……... Expired Date

FIG: …… Figure

PHP: …… ……. Hyper Text Preprocessor

Admin: ……Administrator

DB: …… Database

GUI: ……...Graphical User Interface

HTML: ……... Hypertext Markup Language

INFO: …… Information

MS: ……... Microsoft

MYSQL: …….. Structured Query Language

CSS: ……Cascading Style Sheet

WBBMS: ……. Web Based Blood Bank Management System

OS: ……... Operating System

USB: …….Universal Serial Bus

BR…….Business Rule

Executive project Summary

The Web-Based Blood Bank Management System is a comprehensive and innovative solution designed to streamline and enhance the management of blood donation processes, blood inventory, and hospital blood requests. This system is meticulously crafted to handle various critical functions of a blood bank, including donor registration, blood collection, testing, storage, and distribution, while ensuring the highest standards of safety, reliability, and availability of blood supplies. By integrating essential functionalities and automating routine tasks, the system significantly reduces manual processes and paperwork, which enables staff to focus on more critical, value-added tasks, thereby improving overall operational efficiency.

Additionally, the system provides a centralized platform that enhances accessibility and coordination between donors, hospitals, and the blood bank, ensuring seamless interaction and collaboration. Through efficient donor management, the system facilitates easy registration, updates of donor information, and eligibility checks, ensuring a robust and active donor base. The meticulous tracking and management of blood units—from collection through testing and storage to final distribution—ensure that all units meet stringent safety and quality standards.

Furthermore, the system streamlines hospital blood requests, making the process of requesting and receiving blood units more efficient and transparent. Enhanced inventory management features, such as real-time stock monitoring and automated report generation, optimize the use of available blood units, reducing wastage and ensuring timely availability for patients in need. Ultimately, the Web-Based Blood Bank Management System promises to modernize blood bank operations, enhancing the efficiency, safety, and reliability of blood donation and distribution processes, and contributing significantly to saving more lives through better resource management.

Chapter one: Introduction

The project in development is a web-based blood donation management system specifically designed for the Gondar blood bank. Its primary objective is to effectively handle and maintain the Gondar city blood bank management system while providing efficient transfusion services. The system oversees the entire blood transfusion process, including donor registration, blood collection, blood testing, blood storage in the blood stock, and distribution of blood to hospitals based on patient needs.

The project aims to provide essential information about the availability and locations of blood in emergency situations. It enables online reservation of blood donation dates and maintains donor statuses to gather volunteers for the blood bank. Hospitals can also request blood from the blood bank using patient names. The project ensures efficient management of the blood bank by tracking blood types, quantities, and expiry dates for each category of stored blood in the transfusion unit. Additionally, the system displays the total amount of blood in stock, enabling the inventory manager to efficiently manage the blood bank. It also alerts the manager about expired blood types for proper disposal.

1.1 Background of the project

In Ethiopia, the blood bank service was introduced in 1969 in 12 hospitals. Initially, the Ethiopian blood bank was supported by the Federal Ministry of Health (FMOH) and the Centers for Disease Control and Prevention (CDC). In 2004, it became a separate entity from the Ethiopian Red Cross Society (ERCS) [1].

The blood bank system in Ethiopia consists of a main bank located in Addis Ababa and 26 sub-branches spread across the country. The Gondar city blood bank is one of these 26 branches. also have 17 blood collection centers across the country.

The Gondar city blood bank was established in 2006 E.C. by the federal government and the Amhara regional state. Initially, it provided a limited capacity of blood services. The organization's objectives include delivering efficient services to users, increasing the capacity to provide blood for recipients, increasing the number of blood donors through public education, and facilitating blood donation services. Over time, the capacity of the blood bank has grown, but the organization still relies on a manual system. This manual system leads to challenges such as data redundancy and errors in data storage.

The primary purpose of the organization is to save the lives of individuals affected by various conditions such as accidents, cancer, sickle cell disease, prematurity, and surgical procedures.

Although there has been an increase in the number of blood donors in Ethiopia, the existing system is outdated and paper-based. Therefore, there is a strong interest in computerizing the system through the development of the "Gondar city blood donation management system."

1.2 Statement of the Problem and justification

The utilization of manual procedures in the blood bank to handle information concerning donors and recipients gives rise to a range of problems, including:

- The processes of donating require that the donors must be come and register at the blood bank.
- Less security.
- Wastage of resource and consuming storage space and time.
- Difficult to search, retrieve, update and delete the data about donors and other users of the system.
- The absence of electronic data storing mechanism it requires huge storage space.
- The current process requires high human-power.

Justification

For the problems we've stated above we have provided the solutions as follows

- By implementing an automated donor registration system with online capabilities, we can address the issue of donors having to physically come and register at the blood bank. This improves convenience and accessibility for the donors, making the registration process more efficient.
- By implementing an automated donor registration system with appropriate access controls and data encryption, we can enhance the security and confidentiality of donor and recipient information, resolving the problem of less security in the manual procedures.
- By implementing an automated donor registration system that stores data electronically, we can address the wastage of resources and storage space required by the current paper-based system.
- By implementing an automated donor registration system with electronic databases and management capabilities, we can enable easier searching, retrieval, updating, and deletion of donor and user data, addressing the difficulty in data management.
- By implementing an automated donor registration system with electronic data storage capabilities, we can overcome the limitations of the absence of an electronic data storing mechanism, providing a reliable and scalable solution.
- By implementing an automated donor registration system with streamlined processes, we can reduce the need for extensive human involvement, addressing the problem of the current process requiring high human-power.

1.3 Objective of the project

1.3.1 General Objective

The general objective of this project is to develop web-based Blood management system for Gondar city.

1.3.2 Specific objective

- Gathering requirements.
- Identifying and defining of the problem that the existing system have.
- Analyzing the existing system.
- Identifying functional and nonfunctional requirements.
- Design interactive user interfaces.
- Coding and testing.
- Implementation of the new system.
- Prepare the documentation and train the users.

1.4 Scope of the project

Donor Management: The system will facilitate the registration and management of blood donors, maintaining their personal information, medical history, donation records, and eligibility criteria. It will provide features for donor engagement, such as notifications, reminders, and rewards for regular donors.

Appointment Scheduling: The system will allow donors to schedule appointments for blood donation, providing them with available time slots and locations. It will enable donors to select a convenient date, time, and donation center, streamlining the process and minimizing wait times.

Donor Notification: The system will be designed to inform blood donors when their donated blood saves a patient's life, enhancing donor motivation.

Pre-Donation Assessment: The system will incorporate pre-donation assessments to ensure the eligibility and suitability of donors. It will collect information about potential health risks, travel history, and medication use. Based on the assessment, the system will determine if the donor meets the necessary criteria for donation.

Blood Inventory Management: The system will include features for blood banks to manage their inventory effectively. It will track and monitor blood units, including information on blood type, expiration dates, and storage locations. The system will provide alerts for low stock and facilitate efficient allocation of blood units to meet hospital demands.

Communication and Coordination: The system will enable seamless communication and coordination between blood banks, donors, and hospitals. It will facilitate real-time notifications, updates, and alerts regarding blood drives, urgent needs, and availability of blood units. It will also support communication channels for queries, requests, and feedback.

Reporting and Analytics: The system will generate comprehensive reports and analytics on blood donation activities, including donor demographics, donation trends, and inventory status. These reports will aid in decision-making, planning future blood drives, and assessing the overall efficiency of the blood donation system.

Security and Privacy: The system will prioritize the security and privacy of donor and patient information. It will incorporate robust data encryption, secure access controls, and compliance with relevant data protection regulations to safeguard sensitive data.

1.5 Limitation of the project

- The system cannot help people that have visual impairment because of lack of sound sensor machine.

1.6 System Development Methodology

1.6.1 System development Approach

In this project We will be using agile approach for the design.

Agile methodology is an iterative and flexible approach to project management and software development. It emphasizes collaboration, adaptability, and continuous improvement throughout the project lifecycle.

We choose this agile data model because: -

- **Flexibility:** Agile allows for changes in requirements throughout the project, which is crucial for a system like blood donation management where requirements may evolve based on user feedback, regulatory changes, or emerging needs.
- **Customer Collaboration:** Agile encourages frequent collaboration with stakeholders, including blood donation centers, donors, recipients, and healthcare professionals. This ensures that the system meets their needs and addresses any concerns or feedback they may have.
- **Iterative Development:** Agile's iterative approach allows for the delivery of usable increments of the system in short iterations, enabling stakeholders to provide feedback early and often. This helps in validating the system's functionality and making necessary adjustments as the project progresses.
- **Adaptability to Changing Needs:** Given the dynamic nature of healthcare systems and regulatory requirements, Agile's ability to adapt to changing needs and priorities is advantageous for a blood donation management system.
- **Continuous Improvement:** Agile promotes continuous improvement through regular retrospectives, where the team reflects on what went well, what didn't, and how to improve. This ensures that the system evolves over time to better serve its users and stakeholders.

1.6.2 System Development Tools

Activities	Tools
Client-side coding	HTML, CSS, JS
Database Server	MY SQL database
Web server Server-Side ScriptBrowser	Apache server PHP CHROME
Editor	VISUAL STUDIO CODE
Documentation	MS Word
Design	Edrawmax , draw.io

Table 1. 1 system development tools

1.7 Significance of the project

The development of an advanced blood donation management system holds immense significance in improving the efficiency, safety, and accessibility of blood donations. This project aims to address the following key areas: -

Saving Lives: The availability of an adequate and safe blood supply is crucial for medical procedures, emergency situations, and life-saving treatments. By implementing a streamlined blood donation management system, we can ensure that patients receive the necessary transfusions promptly, potentially saving countless lives.

Meeting Demand: The demand for blood is constant, driven by accidents, surgeries, and various medical conditions. However, managing this demand effectively can be challenging. Our project seeks to optimize the blood supply chain by automating processes such as donor registration, appointment scheduling, and inventory management. By doing so, we can reduce the risk of shortages and ensure that blood products are readily available when needed.

Enhancing Efficiency: Traditional manual processes in blood donation management are time-consuming, error-prone, and resource-intensive. Our digital blood donation management system will revolutionize the way blood banks and donation centers operate. By automating tasks, eliminating paperwork, and providing real-time information, we can significantly enhance operational efficiency, saving valuable time and resources for healthcare professionals.

Donor Engagement and Retention: Maintaining an active and engaged donor base is crucial for a sustainable blood supply. With our system, we will establish personalized communication channels with donors, providing them with updates, reminders, and appreciation for their contributions. By enhancing the donor experience, we aim to increase donor retention rates, ensuring a consistent supply of blood products.

Data Management and Analysis: Accurate data management is vital for effective decision-making and resource allocation. Our blood donation management system will securely store donor information, medical histories, and blood inventory data. By analyzing this data, we can identify

trends, optimize donation campaigns, and make informed decisions to meet the specific needs of different patient groups.

Safety and Quality Assurance: Ensuring the safety and quality of donated blood is of utmost importance. Our system will incorporate robust features such as comprehensive donor eligibility screening, infectious disease testing, and traceability of blood units. These measures will minimize the risk of transmitting infections through transfusions, ensuring the highest standards of safety and quality assurance.

Minimized Blood Wastage: Efficient inventory management reduces unnecessary discards, optimizing blood utilization and minimizing wastage.

Enhanced Donor Accessibility: The system promotes broader accessibility to blood donation, encouraging more individuals to become donors and contribute to the blood supply.

Improved Communication and Coordination: Effective communication streamlines interactions among blood banks, donors, and hospitals, ensuring seamless coordination and timely updates.

Transparent and Reliable Records: Accurate records enable traceability, analytics, and compliance, ensuring reliable information for effective management.

1.8 Beneficiaries of the project

The implementation of an advanced blood donation management system offers numerous benefits and positively impacts various stakeholders involved in the blood donation process. The following beneficiaries are at the forefront of our project:

Patients in Need: The primary beneficiaries of our blood donation management system are the patients who require blood transfusions. By streamlining the donation process, ensuring an adequate supply of blood products, and improving the efficiency of distribution, we aim to enhance patient care. Timely access to safe and compatible blood units can significantly improve treatment outcomes and contribute to the well-being and recovery of patients.

Blood Donors: Our project also focuses on providing a positive experience for blood donors. By implementing a user-friendly and efficient system, we aim to engage donors and make the donation process more convenient and accessible. Donors will benefit from features such as easy registration, appointment scheduling, and personalized communication. This encourages regular and voluntary blood donations, fostering a culture of giving and altruism.

Healthcare Professionals: The blood donation management system simplifies tasks for healthcare professionals involved in the blood collection and transfusion process. By automating administrative processes, such as donor registration, blood inventory management, and tracking of blood units, we reduce the burden of paperwork and streamline workflows. This allows healthcare professionals to focus more on patient care, improving overall efficiency and enhancing the quality of medical services.

Blood Banks and Donation Centers: Blood banks and donation centers play a crucial role in the blood donation ecosystem. Our project benefits these organizations by optimizing their operations.

The system facilitates efficient inventory management, ensuring an adequate supply of blood products while minimizing wastage. It also enhances traceability and improves the accuracy of data, enabling better planning, forecasting, and resource allocation for blood banks and donation centers.

Hospitals and Healthcare Facilities: Hospitals and healthcare facilities heavily rely on a stable and accessible blood supply to provide essential medical care. The blood donation management system ensures that hospitals have a reliable source of blood products readily available. By reducing the risk of shortages, delays in treatment, and complications related to blood transfusions, our system improves the overall quality of healthcare services and enhances patient safety.

Regulatory Authorities and Public Health Agencies: Regulatory authorities and public health agencies are responsible for enforcing standards and ensuring the safety and quality of blood donations. Our project benefits these entities by incorporating features that adhere to regulatory guidelines. These include robust donor eligibility screening, infectious disease testing, and traceability of blood units. By implementing these measures, we contribute to the maintenance of a safe and reliable blood supply.

Emergency Response Organizations: During emergencies and natural disasters, the need for blood products often surges. Our blood donation management system supports emergency response organizations by enabling efficient coordination and distribution of blood units. Real-time information on blood availability, donor status, and distribution networks empowers these organizations to respond swiftly and effectively to emergency situations, potentially saving more lives.

1.9 Feasibility study

1.9.1 Technical Feasibility

Compatibility and Scalability:

The system is designed to be compatible with major web browsers (Chrome, Firefox, Safari) and is responsive across various devices, ensuring accessibility for all users.

It features a scalable architecture capable of handling a growing user base and concurrent requests, making it robust for future expansion.

The system integrates seamlessly with existing healthcare systems through industry-standard protocols and APIs, ensuring smooth interoperability.

User-Friendly Interface:

Built with a focus on graphical user interface concepts, the application is intuitive and easy to use, even for novice users. This design choice minimizes the need for sophisticated training and supports immediate usability.

The system is customized to fit the specific information transfer needs of the North Gondar blood bank, ensuring that most of the required technology is already available and effectively utilized.

Its self-explanatory nature further enhances user experience, providing a straightforward and efficient way to manage blood donation processes without requiring additional technical support.

1.9.2 Economic Feasibility

The new system is developed with minimal cost and provides substantial benefits, such as improving service quality and reducing user workload. The organization avoids media advertising costs by making information available online, allowing everyone to access it via the website.

Tangible Benefits:

Tangible benefits are measurable in monetary terms and include:

- Increased flexibility in modifying the blood bank information system.
- Enhanced data backup capabilities through a well-designed database.
- Reduced resource requirements, such as paper and pens, and decreased advertising costs.
- Faster information retrieval.
- Improved blood bank management system performance.
- Timely information on blood transfusion services available 24/7.

Intangible Benefits:

Intangible benefits, which are not easily measured in monetary terms, include:

- Improved accuracy of blood bank information.
- Faster decision-making through efficient record searches.
- Enhanced security with authorized user access.
- Reduced workload for system users.
- Fewer errors in recording blood donation details.
- Increased efficiency of the blood bank management system.

Cost Reduction and Avoidance:

Current system costs:

Total workers:17

Average monthly salary per worker:7345br

Annual salary cost: 17*12*7345=1,498,380

Daily stationary expenses (Pen, paper, print etc.): 350

Monthly stationary expenses: 30*350=10,500

Annual stationary expenses: 10500*12=126,000

Total annual cost: Total stationary expense per year + Total money required for payment per year=126,000+1,498,380=1,624,380

New system costs:

Total workers needed:12

Average monthly salary per worker: 9000

Annual salary cost: 12*12*9000=1,296,000

Annual stationary expenses: 30,000

Total annual cost: Total stationary expense per year + Total money required for payment per year= 30,000+1,296,000=1,326,000

Cost reduction and avoidance: 1,624,380 – 1,326,000 = 298,380

Total cost reduction: 298,380

1.9.3 Operational Feasibility

The system ensures adequate throughput and delivers necessary information in a timely and well-formatted manner. It provides secure access by granting privileges to authorized users through account management. Additionally, the system includes user help descriptions for guidance and allows developers to perform technical modifications.

Impact on Processes and Workflows:

- Automates donor registration and appointment scheduling.
- Integrates with existing systems for seamless data transfer and reduced manual effort.
- Provides real-time notifications for timely response and coordination.

Tidiness and Training:

- Offers comprehensive training for blood bank staff, healthcare professionals, and donors.
- Features user-friendly interfaces and intuitive design for easy navigation.
- Includes detailed documentation and ongoing support for effective system use.

1.10 Project Schedule

Activity	Time schedule				
	Mar11-Apr8	Apr9-Apr24	Apr25-May25	Jun28-Agu30	Aug31-Sep15
Introduction and Description of existing system	★				
Requirement analysis and modeling		★			
Design			★		
Implementation				★	
Testing and installation					★

Table 1. 2 Project Schedule

1.11 Project Budget

Types of cost	Name	Quantity	Unit Price	Total price
Hardware cost	Computer	1	Free	Free
	Flash(8GB)	1	450	450
	Print	-	1000	1000
	Other material	-	300	300
Software cost	Xampp server	1	Free	Free
	Visual studio code	1	Free	Free
	Edrawmax, draw.io	1	Free	Free
	Microsoft office 2021	1	Free	Free
	Window 10	1	Free	Free
Total cost	--	--	--	1,750

Table 1. 3 Project Budget

Chapter Two: Requirement Analysis

2.1 Current System Description

According to the information collected by the team, the blood bank in Gondar city currently operates in a manual manner. This means that tasks such as documenting, writing, finding, and searching for specific blood-related information are all performed manually. Additionally, there is no logging function in place, which makes it impossible to trace each process or workflow from the database. Furthermore, the system does not display expired blood units. These types of systems force workers to record incorrect and redundant information, resulting in inefficiencies and consuming valuable time when completing specific tasks.

2.1.1 Major function of current system

Donor Recruitment: The system relies on community outreach programs, awareness campaigns, and word-of-mouth to recruit voluntary blood donors. It emphasizes educating potential donors about the importance of blood donation and the need for regular contributions.

Donor Registration: The system maintains manual records of registered donors. Donors provide their personal information, contact details, and medical history using paper forms or logbooks. This information is manually entered into a donor registry or database for future reference.

Donor Screening and Evaluation: Prior to blood donation, the system conducts a manual screening process. Donors are asked a series of questions regarding their health history, travel, and potential risk factors for infectious diseases. The screening process may involve simple physical examinations, such as checking vital signs like blood pressure and pulse rate.

Blood Collection: The system organizes blood donation drives at specific locations, such as hospitals or community centers. healthcare professionals manually collect blood from donors using sterile techniques and standard blood collection equipment, such as needles and blood bags.

Blood Testing and Processing: After collection, the donated blood samples are manually labeled and sent to a laboratory for testing. Manual laboratory procedures are employed to perform blood typing, cross-matching, and screening for infectious diseases. Test results are manually recorded and communicated to the appropriate personnel.

Blood Storage and Inventory Management: The system manually stores donated blood in refrigerators or other appropriate storage units. Manual inventory management methods, such as logbooks or inventory cards, are used to track blood units, monitor expiration dates, and ensure an adequate supply of blood products.

Distribution and Transfusion: When a hospital or healthcare facility requires blood, the system manually coordinates the distribution of blood units. This involves physically transporting the blood products to the required location, typically through designated personnel or couriers. Documentation, such as handwritten requisition forms, is used to track the distribution process.

Donor Recognition and Retention: The system manually recognizes and appreciates blood donors. This may include providing handwritten thank-you notes, certificates. Manual systems rely on personal communication to engage with donors, share updates, and encourage continued participation.

Data Management and Analysis: The system manually compiles and maintains records of donor information, donation details, and transfusion records. These records are typically stored in paper-based files or logbooks. Manual analysis of the data may be performed to identify trends or patterns, although it can be time-consuming and less efficient compared to automated systems.

2.2 Requirement Gathering

2.2.1 Requirement Gathering Methods

While collecting the essential data and information required for system analysis, we employed two fact-finding techniques derived from both primary and secondary sources. Regarding primary sources, the project team utilizes methodologies like interviews and observations. On the other hand, secondary sources involve obtaining data from various documents.

2.3.1.1 Interview

The project team conducted interviews with staff members and the manager of the Gondar city blood bank to gather essential information. These interviews served to identify the existing problems within the blood bank and gain an understanding of the current system. Based on the information obtained, the project team analyzed the blood bank's operations and gained an understanding of the fundamental concepts underlying its management in the current system. To establish a foundation, the team posed questions aimed at describing and providing background information on key aspects of the system.

2.3.1.2 Record review

This technique proved to be highly advantageous in the development of our database. We thoroughly examined the current file structure and analyzed the documents utilized and produced within the organization. In the process of registering each file, we made a concerted effort to determine its relevance, importance, and relationship with other data items. To illustrate, we made use of records such as manuals that were created by the organization.

2.3.1.3 Observation

To gather factual information, the team conducted observations of all activities and transactions involving files and documents. This approach enabled us to uncover the actual functioning of the system, distinguishing it from the ideal or desired state. Observation proved particularly valuable when we needed to witness firsthand how documents were handled, processes were executed, and whether prescribed steps were actually followed. During the observation of the existing system, it became evident that there were instances of improper handling of files, making them difficult to access and rendering decisions invalid. Additionally, the team gained insights into the challenges faced by both donors and collectors during the blood donation process.

2.2.2 Business Rules

This business rule establishes fundamental guidelines how the blood bank conducts its operations. The rules articulate the organization's policies and procedures, and provide a deeper understanding of the activities and processes within the existing blood donation system. Basic business rule of the blood bank are: -

BR1: The donor and hospital must be registered with the blood bank to participate.

BR2: The lab technician must test the newly added blood to ensure its safety and quality.

BR3: The donor must fulfill the required criteria, including their health status, age, and weight, to be eligible to donate blood.

BR4: The nurse must collect blood from donors to maintain a steady supply.

BR5: Hospitals that request blood must be registered members of the blood bank to ensure proper tracking and accountability.

BR6: The inventory manager is responsible for distributing the collected blood to the hospitals in need.

2.3 Proposed System Description

2.3.1 Overview

The objective of developing the proposed web-based Gondar City Blood Bank Management System is to address the issues outlined in the problem statement associated with the current system. The new system aims to be efficient in terms of registration, insertion, updating, accessing, searching, and generating reports. It offers various functions to replace the manual system effectively. One key feature is its robust database security, ensuring that unauthorized users cannot access it. Each member of the Gondar City Blood Bank Management System will have their own privileges to perform operations on the database. Additionally, the proposed system will facilitate resource control and enhance communication between users and customers. Overall, the proposed system aims to alleviate existing problems and reduce resource consumption by transitioning from a manual-based approach in blood bank activities.

2.3.2 Functional Requirements

Functional requirements encompass the desired behavior and capabilities of the system. In the case of the proposed system, it is designed to fulfill the following functionalities:

- Register and Keep record of donors and recipients in the database.
- Store new donors, hospital, blood group record in database, this is done by only privileged users of the bank.
- Retrieve blood donors and hospitals from a database; the user/employee searches this information whenever it is required.
- Renew blood donors, blood group, hospital and blood testing information; administrator can update this information by assigning the new value to the old value.
- Discard old information from a database and infectious blood.
- Generate of reports about the work of north Gondar blood bank management system by date per years per division of resource in a given time.
- support multiple languages, including English and Amharic.

2.3.3 Nonfunctional requirements

The non-functional requirement of the system pertains to the quality and performance aspects of the system in terms of providing satisfactory service to the user.

2.3.3.1 Performance

The system will provide fast response times for tasks such as donor registration, blood inventory management, and distribution to ensure efficient and timely operations.

It should be able to handle a high volume of concurrent users and processes without significant performance degradation.

The system will optimize data retrieval and processing to minimize delays and enhance user experience.

2.3.3.2 Scalability

The system will be designed to accommodate an increasing number of registered donors, blood units, and healthcare facilities without compromising performance.

It will be scalable both vertically (increasing resources within a single server) and horizontally (adding more servers) to handle growing demands.

2.3.3.3 Availability

The system will be available 24/7 to ensure uninterrupted access for donor registration, blood inventory management, and distribution.

It will have measures in place to minimize downtime, such as redundant servers, backup power supplies, and disaster recovery plans.

2.3.3.4 Reliability

The system will be reliable and operate without errors or failures that could disrupt critical processes.

It will be able to recover from system failures or crashes, ensuring data integrity and minimizing data loss.

2.3.3.5 Maintainability

The system will be designed with modular and well-structured code to facilitate ease of maintenance and updates.

It will have comprehensive documentation, including system architecture, codebase, and user manuals, to aid in system maintenance and troubleshooting.

2.3.3.6 Security

The system will have robust security measures to protect sensitive donor information, ensuring confidentiality, integrity, and availability of data.

It will include authentication and authorization mechanisms to control access to the system and ensure only authorized users can perform relevant tasks.

Data transmission and storage be encrypted to prevent unauthorized access or data breaches.

2.3.3.7 Environmental

The system will be designed to minimize its environmental impact, such as optimizing energy consumption and reducing carbon footprint.

It will comply with relevant environmental regulations and standards.

2.3.3.8 Usability

The system will have a user-friendly interface, with intuitive navigation and clear instructions, to ensure ease of use for both donors and healthcare professionals.

It will provide appropriate feedback and guidance to users during the interaction with the system.

2.3.3.9 Interoperability

The system will be able to integrate with other healthcare systems, such as electronic health record (EHR) systems, to enable seamless information exchange and improve overall efficiency.

It adheres to relevant interoperability standards and protocols to facilitate data sharing and interoperability with external systems.

Chapter 3: SYSTEM MODEL

3.1 Scenario

Scenario Name: Login

Flow of Events:

- The admin user navigates to the blood bank's login page.
- The admin enters their registered username and password into the login form.
- The blood bank's system validates the entered credentials against the stored admin account information.
- If the credentials are valid, the system authenticates the admin user and redirects them to the admin home page.
- The admin home page is displayed, showing the admin-level dashboard and access to all blood bank management functions.
- If the credentials are invalid, the system displays an error message and prompts the admin to try logging in again.
- The admin can attempt to log in again with the correct username and password.

Scenario Name: Account Management

Flow of Events:

- The administrator navigates to the blood bank's login page.

- The administrator enters their valid username and password, then clicks the login button.
- The system authenticates the administrator's credentials and redirects them to the admin dashboard page.
- On the admin dashboard, the administrator sees various links and options for account management functions.
- The administrator clicks the "Create New Account" link.
- The system displays a form for entering the required information to create a new account (e.g., name, contact details, blood type, etc.).
- The administrator fills out the form completely and accurately.
- The administrator submits the form.
- The system validates the input data and creates a new account in the system.
- The system displays a success message: "Account successfully created

Scenario Name: Post Info

Flow of Events:

- The administrator navigates to the blood bank's login page.
- The administrator enters their valid username and password, then clicks the login button.
- The system authenticates the administrator's credentials and redirects them to the admin dashboard page.
- On the admin dashboard, the administrator sees an "Add Info" menu option or link.
- The administrator clicks the "Add Info" option.
- The system displays a rich text box form for the administrator to fill out.
- The administrator enters the desired information into the rich text box.
- The administrator clicks the "Submit" button to save the new information.
- The system validates the submitted content.
- The system successfully adds the new information and displays a confirmation message: "Information added successfully."

Scenario Name: Hospital Enrollment

Flow of Events:

- The administrator navigates to the blood bank's login page.
- The administrator enters their valid username and password, then clicks the login button.
- The system authenticates the administrator's credentials and redirects them to the admin dashboard page.
- On the admin dashboard, the administrator sees a "Hospital Registration" menu option or link.
- The administrator clicks the "Hospital Registration" option.
- The system displays a hospital registration form for the administrator to fill out.
- The administrator enters all the required information about the hospital (e.g. name, address, contact details, etc.).

- The administrator clicks the "Register" button to submit the hospital registration.
- The system validates the submitted registration details.
- The system processes the hospital registration and displays a confirmation message: "Hospital registered successfully."

Scenario Name: Donation Request

Flow of Events:

- The donor navigates to the blood bank's login page.
- The donor enters their valid username and password, then clicks the login button.
- The system authenticates the donor's credentials and redirects them to the donor dashboard page.
- On the donor dashboard, the donor sees a "Request Donation" link or option.
- The donor clicks the "Request Donation" link.
- The system displays a donation request form for the donor to fill out.
- The donor provides the required information on the form (e.g. preferred donation date/time, blood type, contact details, etc.).
- The donor reviews the details and clicks the "Submit Request" button.
- The system validates the submitted donation request information.
- The system processes the donation request and displays a confirmation message: "Donation request submitted successfully."

Scenario Name: View Report

Flow of Events:

- The administrator navigates to the blood bank's login page.
- The administrator enters their valid username and password, then clicks the login button.
- The system authenticates the administrator's credentials and redirects them to the admin dashboard page.
- On the admin dashboard, the administrator sees a "View Report" menu option or link.
- The administrator clicks the "View Report" option.
- The system displays a report generation form, prompting the administrator to enter the desired report date range.
- The administrator fills out the report date field(s) on the form.
- The administrator clicks the "Generate Report" button.
- The system processes the report request using the provided date range.
- The system displays the generated report and a confirmation message: "Report viewed successfully."

Scenario Name: Approve Request

Flow of Events:

- The administrator navigates to the blood bank's login page.
- The administrator enters their valid username and password, then clicks the login button.
- The system authenticates the administrator's credentials and redirects them to the admin dashboard page.
- On the admin dashboard, the administrator sees an "Approve Requests" menu option or link.
- The administrator clicks the "Approve Requests" option.
- The system displays a list of pending donations or other requests that require administrator approval.
- The administrator reviews the details of each request in the list.
- If the administrator determines the request information is correct and valid, they click the "Approve" button for that request.
- The system processes the approval of the request.
- The system displays a confirmation message: "Request approved successfully.".

Scenario Name: View Comment

Flow of Events:

- The user navigates to the blood bank's login page.
- The user enters their valid username and password, then clicks the login button.
- The system authenticates the user's credentials and redirects them to their dashboard or homepage.
- On the user's dashboard, they see a "View Comments" menu option or link.
- The user clicks the "View Comments" option.
- The system retrieves and displays a list of feedback comments provided by customers.
- The user can browse through the list of comments.
- The user can read the details and contents of each comment.

Scenario Name: Give Comment

Flow of Events:

- The user navigates to the blood bank's website and logs in to their account.
- The user is directed to their user dashboard or homepage.
- On the user's dashboard, they see a "Give Feedback" or "Contact Us" link or menu option.
- The user clicks the "Give Feedback" or "Contact Us" option.
- The system displays a feedback/contact form for the user to fill out.
- The user provides the required information in the form fields, such as their name, email, comment/feedback text, etc.
- The user reviews the details they've entered and clicks the "Submit" button.
- The system validates the submitted feedback form.

- The system processes the user's feedback and displays a thank you message: "Thank you for your comment. We appreciate your feedback.

Scenario Name: Blood Request

Flow of Events:

- The hospital administrator navigates to the blood bank's login page.
- The administrator enters their valid username and password, then clicks the login button.
- The system authenticates the administrator's credentials and redirects them to the hospital admin dashboard.
- On the hospital admin dashboard, the administrator sees a "Request Blood" link or menu option.
- The administrator clicks the "Request Blood" option.
- The system displays a blood request form for the administrator to fill out.
- The administrator provides the required information on the form, such as the blood type needed, quantity, preferred delivery date, hospital details, etc.
- The administrator reviews the request details and clicks the "Submit Request" button.
- The system validates the submitted blood request information.
- The system processes the blood request and displays a confirmation message: "Blood request sent successfully."

Scenario Name: Collect Blood

Flow of Events:

- The nurse logs in to the blood bank's system using their valid credentials.
- The system authenticates the nurse's login and directs them to the nurse dashboard or homepage.
- On the nurse dashboard, the nurse sees a "Collect Blood" link or menu option.
- The nurse clicks the "Collect Blood" option.
- The system displays a blood collection form for the nurse to fill out.
- The nurse provides the required information on the form, such as the donor's details, blood type, volume collected, date/time of collection, etc.
- The nurse reviews the collected blood information entered on the form.
- The nurse clicks the "Submit Collection" button.
- The system validates the submitted blood collection details.
- If the information is correct, the system records the blood collection and displays a confirmation message: "Blood collection recorded successfully."

Scenario Name: Donor Registration

Flow of Events:

- The donor navigates to the blood bank's registration page.
- The donor fills out the registration form with the required information.
- The donor submits the registration form.

- The system validates the submitted information.
- If the information is correct, the system creates a new donor account.
- The system displays a confirmation message: "Donor registered successfully."

Scenario Name: Distribution

Flow of Events:

- The inventory manager logs in to the blood bank's system using their valid credentials.
- The system authenticates the manager's login and directs them to the inventory manager dashboard or homepage.
- On the inventory manager dashboard, the manager sees a "Distribute Blood" link or menu option.
- The manager clicks the "Distribute Blood" option.
- The system displays a blood distribution search interface, allowing the manager to filter and search for available blood inventory.
- The manager enters the relevant search criteria, such as blood type, quantity, expiration date, etc.
- The system retrieves and displays the matching blood inventory available for distribution.
- The manager reviews the blood inventory details and selects the appropriate units to distribute.
- The manager clicks the "Distribute" button for the selected blood units.
- The system processes the blood distribution, updates the inventory, and displays a confirmation message: "Blood distributed successfully."

Scenario Name: Manage Stock

Flow of Events:

- The inventory manager logs in to the blood bank's system using their valid credentials.
- The system authenticates the manager's login and directs them to the inventory manager dashboard or homepage.
- On the inventory manager dashboard, the manager sees a "Manage Stock" link or menu option.
- The manager clicks the "Manage Stock" option.
- The system presents the manager with three main actions for stock management: Update, Register, and Discard.
- If the manager chooses to discard expired blood:
- The system displays the current blood inventory, including expired units.
- The manager selects the expired blood units they want to discard, typically by selecting the pack number.
- The manager clicks the "Discard" button for the selected expired blood units.
- The system processes the blood discarding, updates the inventory records, and displays a confirmation message: "Expired blood discarded successfully."

Name of Scenario: Notify Blood Donor

Flow of Events:

- Blood donor visits the blood bank and donates blood.
- The blood bank records the donor's details (name, contact info, blood type, etc.) and assigns a unique ID to the donated blood unit.
- The donor's information and blood unit ID are stored in the blood bank's system.
- A hospital requests a blood transfusion for a patient.
- The blood bank retrieves the appropriate blood unit and sends it to the hospital, including the donor's ID.
- The hospital uses the donated blood for the patient's treatment.
- The hospital reports back to the blood bank on whether the transfusion was successful in saving the patient's life.
- The blood bank system matches the hospital's feedback to the original donor's ID.
- The blood bank system automatically generates a notification message to the donor, informing them that their donated blood was used to save a patient.
- The notification message is sent to the donor via their preferred contact method (email, SMS, etc.).
- The donor receives the message and feels gratified knowing their donation made a difference.

Name of Scenario: Send patient status

Flow of Events:

- The hospital administrator logs in to the blood bank's system using their valid credentials.
- The system authenticates the administrator's login and directs them to the hospital admin dashboard.
- On the hospital admin dashboard, the administrator sees a "Send Patient Status" link or menu option.
- The administrator clicks the "Send Patient Status" option.
- The system displays a patient status form for the administrator to fill out.
- The administrator enters the required information, such as patient ID, blood unit ID, status of the transfusion (successful/unsuccessful), and any additional comments.
- The administrator reviews the entered information for accuracy.
- The administrator clicks the "Submit" button to send the patient status.
- The system validates the submitted information.
- If the information is correct, the system records the patient status and displays a confirmation message: "Patient status sent successfully."

3.1.1 Use Case Model

Actor	Use case
Admin	➢ Login ➢ Approve Request ➢ Hospital Registration ➢ Manage Account ➢ Post Information

	➢ View Comment ➢ Generate Report ➢ Notify Donor
Donor	➢ Login ➢ Donation Request ➢ Give Comment ➢ View Report
Nurse	➢ Login ➢ Collect Blood ➢ View Report
Hospital	➢ Login ➢ Blood Request ➢ Send Patient Status ➢ Give Comment ➢ View Report
Inventory Manager	➢ Login ➢ Add Blood ➢ Discard Blood ➢ Distribute Blood

Table 3. 1 Use case model

3.1.2 **Use Case Diagrams**

The next step is to create the use case model, which is a graphical representation of how the system interacts with its external environment. The key actors that play a major role outside the system are: administrator, nurse, lab technician, inventory manager, hospital, and donor.

The use cases incorporated in the system include: -

- Login
- Manage account
- Add new information
- Approve
- View report
- Register donor
- Register hospital
- Donation request
- Add new blood
- Collect blood
- Blood request
- Blood distribution
- Searching process
- Send Patient Status
- Notify Blood Donor

The provided figure represents the use case model of the system, illustrating the relationships between these various use cases and the actors interacting with the system.

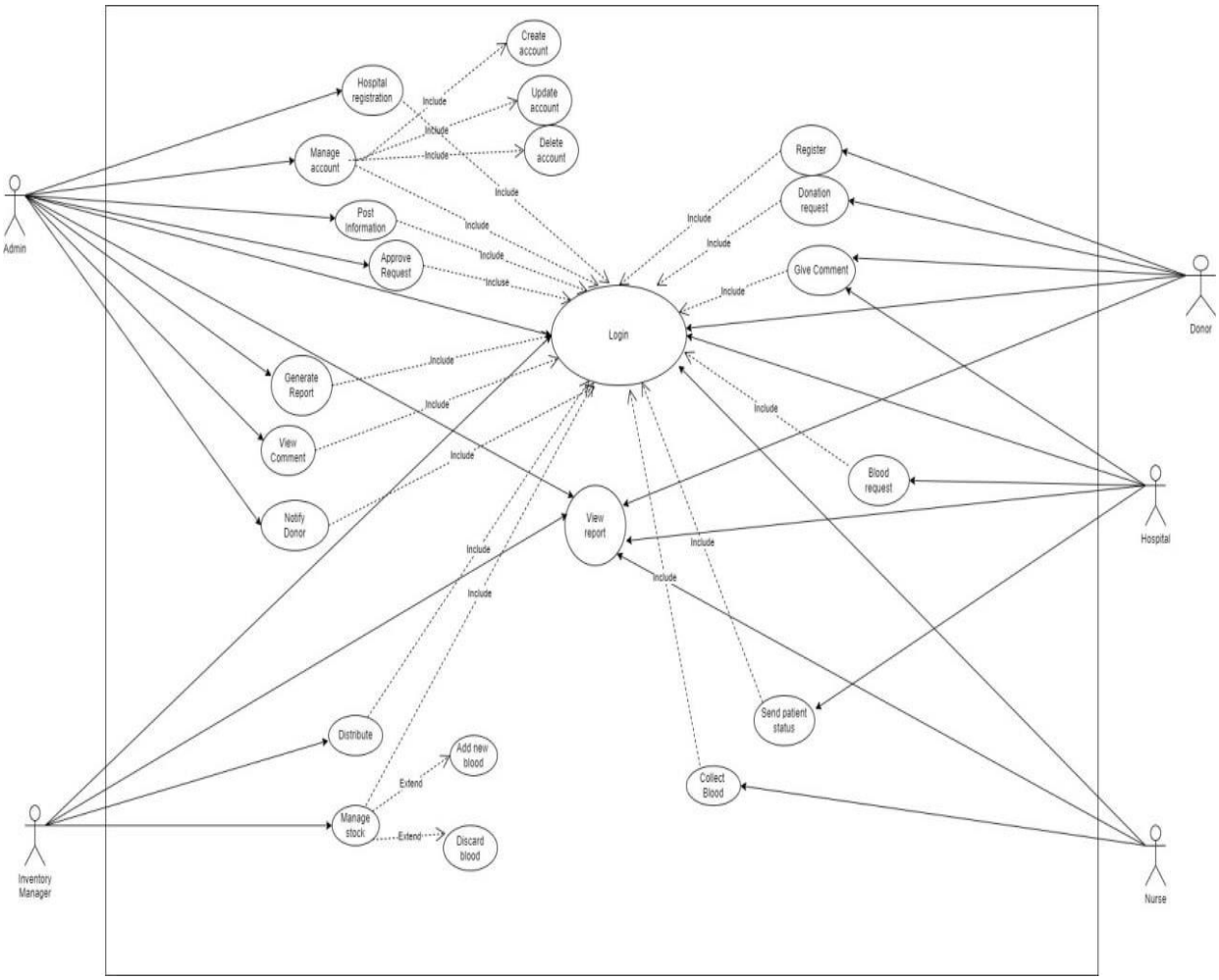

Figure 3. 1 Use case diagram

3.1.3 Description of use case model

Table 3. 2 Use case description for Login

Use case id	UC#1
Use case name	Login
Actors	Admin, Donor, Hospital, nurse, Inventory manager.

Description	The authentication for authorized users in the system to interact with the system Gondar web-based blood bank.
Goal	To be accessed by an authorized and trust system user
Precondition	Any user must have user name and password.
Post condition	User is authenticated and redirected to their respective dashboard. If authentication fails, an error message is displayed, and the user is prompted to try again.
Normal Flow	1. User navigates to the blood bank's login page. 2. User enters their username and password into the login form. 3. User clicks the login button. 4. System validates the entered credentials against the stored user information. 5. If credentials are valid, the system authenticates the user. 6. User is redirected to their respective dashboard (admin, donor, nurse, inventory manager, hospital administrator, or general user).
Alternative Flows	If credentials are invalid: 1. System displays an error message. 2. User is prompted to try logging in again. 3. User can attempt to log in again with the correct username and password.
Exceptions	If the system is down or unavailable, users cannot log in. If the user's account is locked or disabled, they cannot log in and will receive an appropriate message
Assumptions	Users know their valid login credentials. The system is operational and accessible.
Priority	High
Frequency of Use	Multiple times per day, as users log in to perform their respective tasks.
Business Rules	Only authenticated users can access the system Failed login attempts are tracked for security purposes.
Special Requirements	The system must ensure secure transmission of login credentials (e.g., using HTTPS). The system should support account recovery options (e.g., forgot password).

Notes and Issues	Consider implementing multi-factor authentication for enhanced security. The system should provide clear feedback for incorrect login attempts.

Table 3. 3 Use case description for manage account

Use case id	UC#2
Use case name	manage account
Actor	Admin
Precondition	The System administrator must login to control the account
Description	This activity is performed when the admin wants to manage the account
Goal	To control the system user's activity.
Preconditions	Admin is logged in and authenticated. Admin has appropriate permissions to manage accounts.
Postconditions	User accounts are created, updated, or deleted as required. Changes are reflected in the system and communicated to the relevant users
Normal Flow	1. Admin navigates to the blood bank's admin dashboard. 2. Admin clicks on the "Manage Accounts" link. 3. System displays the account management options. 4. Admin selects "Create New Account." 5. System displays a form for entering new account details (e.g., name, contact details, blood type, etc.). 6. Admin fills out the form completely and accurately. 7. Admin submits the form. 8. System validates the input data and creates a new account. 9. System displays a success message: "Account successfully created."
Alternative Flows	For updating an account: 1. Admin selects "Update Account."

	2. System displays a list of existing accounts. 3. Admin selects the account to update. 4. System displays the account details in an editable form. 5. Admin updates the necessary information and submits the form. 6. System validates the updated information and saves the changes. 7. System displays a success message: "Account successfully updated." - For deleting an account: 1. Admin selects "Delete Account." 2. System displays a list of existing accounts. 3. Admin selects the account to delete. 4. System prompts the admin for confirmation. 5. Admin confirms the deletion. 6. System deletes the account and displays a success message: "Account successfully deleted."
Exceptions	If the input data is invalid, the system displays an error message and prompts the admin to correct the information. If the system encounters an error while processing the request, it displays an error message and logs the issue for further investigation.
Assumptions	Admin is familiar with the account management process. The system is operational and capable of handling account management tasks.
Priority	High
Frequency of Use	Regularly, depending on the need to manage user accounts.
Business Rules	Only admins with appropriate permissions can manage accounts All user data must be validated before being stored in the system. Changes to accounts must be logged for auditing purposes.
Special Requirements	The system should support various roles and permissions for different types of users.

	The system must ensure data privacy and security for all user information.
Notes and Issues	Consider implementing additional security measures for account management actions (e.g., multi-factor authentication) The system should provide clear feedback for both successful and unsuccessful account management actions.

Table 3. 4 Use case description for post information

Use case id	UC#3
Use case name	Post information
Actor	Admin
Description	Post new information to create awareness and use full information
Goal	To post new information to the blood bank users and customers
Precondition	The System admin must login to post new information
Postconditions	Information or updates are posted and visible to relevant users Users are informed about the new information through appropriate channels.
Normal Flow	1. Admin navigates to the blood bank's admin dashboard. 2. Admin clicks on the "Post Information" or "Add Info" option. 3. System displays a rich text box form for entering the desired information. 4. Admin enters the information or update into the rich text box. 5. Admin clicks the "Submit" button to post the information. 6. System validates the submitted content. 7. System successfully posts the new information. 8. System displays a confirmation message: "Information added successfully."
Alternative Flows	None

Exceptions	If the input data is invalid, the system displays an error message and prompts the admin to correct the information. If the system encounters an error while processing the request, it displays an error message and logs the issue for further investigation.
Assumptions	Admin is authorized to post information. The system is operational and capable of handling information posting tasks.
Priority	Medium
Frequency of Use	Occasional, depending on the need to communicate updates or information.
Business Rules	Only admins with appropriate permissions can post information All posted information must adhere to the blood bank's guidelines and policies. Information should be clear, accurate, and relevant to users.
Special Requirements	The system should support formatting options for the posted content (e.g., rich text, images, links). Information posted should be archived for future reference
Notes and Issues	Consider implementing approval workflows for sensitive information postings. Admins should be trained on the proper use of the information posting feature to ensure consistency and quality of content

Table 3. 5 Use case description for view report

Use case id	UC#4
Use case name	View report
Actors	Donor, Hospital, nurse, Inventory manager.
Description	View the reported that is generated by admin
Goal	To view about all activities that have been done in organization
Precondition	The actors must be log in to the system
Normal Flow	1. Admin navigates to the blood bank's admin dashboard. 2. Admin clicks on the "View Report" or "Generate Report" option. 3. System displays a report generation form, prompting the admin to enter the desired report date range. 4. Admin fills out the report date field(s) on the form. 5. Admin clicks the "Generate Report" button. 6. System processes the report request using the provided date range. 7. System generates the report based on the specified parameters. 8. System displays the generated report for the admin to review.
Alternative Flows	None
Exceptions	If there is no data available for the specified date range, the system notifies the admin and prompts them to adjust the parameters. If the system encounters an error while processing the request, it displays an error message and logs the issue for further investigation.
Assumptions	Admin is authorized to view reports. The system is operational and capable of generating reports based on the specified parameters.

Priority	Medium
Frequency of Use	Periodically, depending on the need for data analysis and reporting.
Business Rules	Only admins with appropriate permissions can view reports. Report data should be accurate and up-to-date. Reports should be generated in a timely manner.
Special Requirements	The system should support various report formats (e.g., PDF, Excel). Reports should be customizable based on admin preferences and requirements.
Notes and Issues	Consider implementing scheduling options for automated report generation. Admins should be trained on interpreting and utilizing the information provided in the reports effectively.

Table 3. 6 Use case description for Approve

Use case id	UC#5
Use case name	Approve
Primary actors	Admin
Description	For Approving those who send donation request to donate blood.
Goal	Give decision for the donor appointment
Precondition	Admin to approve must be view the donor donation request
Postconditions	Request is approved, and necessary actions are taken based on the approval. User(s) associated with the request are notified about the approval status.
Normal Flow	1. Admin navigates to the blood bank's admin dashboard. 2. Admin clicks on the "Approve Requests" or similar option. 3. System displays a list of pending requests that require approval. 4. Admin reviews the details of each request in the list.

	5. If the request information is correct and valid, admin clicks the "Approve" button for that request. 6. System processes the approval of the request. 7. System displays a confirmation message: "Request approved successfully."
Alternative Flows	None
Exceptions	If the admin determines that the request information is incorrect or invalid, they do not approve the request and may provide feedback to the requester. If the system encounters an error while processing the approval, it displays an error message and logs the issue for further investigation
Assumptions	Admin is authorized to approve requests. The system is operational and capable of handling request approvals.
Priority	High
Frequency of Use	Varied, depending on the volume of requests received by the blood bank.
Business Rules	Only admins with appropriate permissions can approve requests Approval of requests should be done in a timely manner to avoid delays in operations. Approval decisions should be based on predefined criteria and guidelines.
Special Requirements	The system should provide clear and intuitive interfaces for reviewing and approving requests. Notification mechanisms should inform users about the status of their requests promptly.
Notes and Issues	Consider implementing workflow automation for streamlining the approval process. Admins should be trained on the criteria for approving requests to ensure consistency and compliance.

Table 3. 7 Use case description for view comment

Use case id	UC#6
Use case name	View comment
actor	Admin
Description	Users can see the comments that are submitted from the user, customer and other parties.
Goal	To view user feedback about the system.
Precondition	User is logged in and authenticated. User has appropriate permissions to view comments
Postconditions	User can access and review feedback comments. User can use the information from comments to improve services or address issues.
Normal Flow	1. User navigates to the blood bank's login page. 2. User enters their valid username and password, then clicks the login button. 3. System authenticates the user's credentials and redirects them to their dashboard or homepage. 4. On the user's dashboard, they see a "View Comments" menu option or link. 5. User clicks the "View Comments" option. 6. System retrieves and displays a list of feedback comments provided by customers. 7. User can browse through the list of comments. 8. User can read the details and contents of each comment.
Alternative Flows	None
Exceptions	If the system encounters an error while retrieving comments, it displays an error message and logs the issue for further investigation. If there are no comments available, the system displays a message indicating that no comments are found.
Assumptions	User is authorized to view comments. The system is operational and capable of retrieving and displaying comments.
Priority	Medium

Frequency of Use	Regularly, depending on the frequency of feedback comments provided by users or customers.
Business Rules	Only users with appropriate permissions can view comments. Comments should be displayed in a clear and organized manner for easy review.
Special Requirements	The system should support filtering and searching through comments for specific information. Comments should be timestamped and attributed to the respective users who provided them.
Notes and Issues	Consider implementing privacy measures to protect sensitive information within comments. Users should be encouraged to provide constructive feedback to improve the quality of services.

Table 3. 8 Use case description for donation request

Use case id	UC#7
Use case name	Donation Request
actor	Donor
Description	Allows a donor to request a donation appointment through the blood bank system
Goal	To enable donors to easily schedule and submit requests for blood donations.
Precondition	Donor is logged in and authenticated. Donor has a valid donor profile in the system.
Postconditions	The donation request is successfully submitted and logged in the system. The donor receives a confirmation of their request.
Normal Flow	1. Donor navigates to the blood bank's login page.

	2. Donor enters their valid username and password, then clicks the login button. 3. System authenticates the donor's credentials and redirects them to the donor dashboard page. 4. On the donor dashboard, the donor sees a "Request Donation" link or option. 5. Donor clicks the "Request Donation" link. 6. System displays a donation request form for the donor to fill out. 7. Donor provides the required information on the form (e.g., preferred donation date/time, blood type, contact details). 8. Donor reviews the details and clicks the "Submit Request" button. 9. System validates the submitted donation request information. 10. System processes the donation request and displays a confirmation message: "Donation request submitted successfully."
Alternative Flows	If the donor misses any required fields, the system prompts them to complete all required fields before submitting. If the entered information is invalid, the system displays an error message and prompts the donor to correct the information.
Exceptions	If the system encounters an error during request submission, it displays an error message and logs the issue for further investigation.
Assumptions	Donor is authorized to request donations. The system is operational and capable of processing donation requests.
Priority	High
Frequency of Use	Regularly, depending on the frequency of donor requests for donations.
Business Rules	Only authenticated and authorized donors can submit donation requests.

	All required fields in the request form must be completed accurately
Special Requirements	The system should support validation of contact details and other important information. The system should maintain an audit log of all donation requests for accountability.
Notes and Issues	Consider implementing automated reminders and notifications for scheduled donation appointments. Donors should be provided with information on donation eligibility and guidelines during the request process.

Table 3. 9 use case description for Hospital registration

Use case id	UC#8
Use case name Use case id	Hospital Registration
actor	Admin
Description	Allows the admin to register a hospital in the blood bank system.
Goal	To add new hospitals to the blood bank system, enabling them to request blood and participate in blood bank activities.
Precondition	Admin is logged in and authenticated. Admin has appropriate permissions to register hospitals.
Postconditions	The hospital is successfully registered and can access the blood bank system. The system confirms the registration and provides access credentials to the hospital.
Normal Flow	1. Admin navigates to the blood bank's admin dashboard. 2. Admin clicks on the "Hospital Registration" option. 3. System displays a hospital registration form for the admin to fill out.

	4. Admin enters all required information about the hospital (e.g., name, address, contact details). 5. Admin clicks the "Register" button to submit the hospital registration. 6. System validates the submitted registration details. 7. System processes the hospital registration. 8. System displays a confirmation message: "Hospital registered successfully."
Alternative Flows	If the admin misses any required fields, the system prompts them to complete all required fields before submitting. If the entered information is invalid, the system displays an error message and prompts the admin to correct the information.
Exceptions	If the system encounters an error during registration, it displays an error message and logs the issue for further investigation.
Assumptions	Admin is authorized to register hospitals. The system is operational and capable of processing hospital registrations.
Priority	High
Frequency of Use	Occasional, depending on the need to add new hospitals to the blood bank system.
Business Rules	Only authorized admins can register new hospitals. All required fields in the registration form must be completed accurately.
Special Requirements	The system should support validation of contact details and other important information. The system should maintain an audit log of all hospital registrations for accountability.
Notes and Issues	Consider implementing approval workflows for new hospital registrations to ensure accuracy and prevent unauthorized entries.

	Admins should be trained on the registration process to ensure consistency and compliance.

Table 3. 10 Use case description for generate report

Use case id	UC#9
Use case name Use case id	Generate report
actor	Admin
Description	Generating the report that of activities have been done.
Goal	To provide admins with the capability to generate and review reports for analysis and decision-making.
Precondition	Admin is logged in and authenticated. Admin has appropriate permissions to generate reports.
Postconditions	The report is successfully generated and displayed. Admin can review, save, or print the generated report.
Normal Flow	1. Admin navigates to the blood bank's login page. 2. Admin enters their valid username and password, then clicks the login button. 3. System authenticates the admin's credentials and redirects them to the admin dashboard page. 4. On the admin dashboard, the admin sees a "View Report" menu option or link. 5. Admin clicks the "View Report" option. 6. System displays a report generation form, prompting the admin to enter the desired report date range. 7. Admin fills out the report date field(s) on the form. 8. Admin clicks the "Generate Report" button. 9. System processes the report request using the provided date range.

	10. System displays the generated report and a confirmation message: "Report viewed successfully."
Alternative Flows	If the admin misses any required fields, the system prompts them to complete all required fields before generating the report. If the entered date range is invalid, the system displays an error message and prompts the admin to correct the date range.
Exceptions	If the system encounters an error during report generation, it displays an error message and logs the issue for further investigation.
Assumptions	Admin is authorized to generate reports. The system is operational and capable of processing report generation requests.
Priority	High
Frequency of Use	Regularly, depending on the need for reports for analysis and decision-making.
Business Rules	Only authorized admins can generate reports. Reports should be generated based on accurate and up-to-date data.
Special Requirements	The system should support exporting reports in various formats (e.g., PDF, Excel). The system should maintain an audit log of all report generation activities for accountability.
Notes and Issues	Consider implementing predefined report templates to streamline the report generation process. Admins should be trained on the criteria for generating and interpreting reports to ensure consistency and accuracy.

Table 3. 11 Use case description for Give comment

Use case id	UC#10
Use case name	Give Comment
actors	System user
Description	Comment the blood bank system about any thing
Goal	To enable users to share feedback or comments about their experience or any issues they encountered.
Precondition	User must have valid email address comment the system
Postconditions	The feedback or comment is successfully submitted and stored in the system. The user receives a confirmation of their submitted comment.
Normal Flow	1. User navigates to the blood bank's login page. 2. User enters their valid username and password, then clicks the login button. 3. System authenticates the user's credentials and redirects them to their dashboard or homepage. 4. On the user's dashboard, they see a "Give Feedback" or "Contact Us" link or menu option. 5. User clicks the "Give Feedback" or "Contact Us" option. 6. System displays a feedback/contact form for the user to fill out. 7. User provides the required information in the form fields, such as their name, email, comment/feedback text, etc. 8. User reviews the details they've entered and clicks the "Submit" button. 9. System validates the submitted feedback form. 10. System processes the user's feedback and displays a thank you message: "Thank you for your comment. We appreciate your feedback."
Alternative Flows	If the user misses any required fields, the system prompts them to complete all required fields before submitting. If the entered information is invalid, the system displays an error message and prompts the user to correct the information.

Exceptions	If the system encounters an error during feedback submission, it displays an error message and logs the issue for further investigation.
Assumptions	User is authorized to provide comments. The system is operational and capable of processing feedback submissions.
Priority	Medium
Frequency of Use	Regularly, depending on the frequency of user interactions and the need to provide feedback.
Business Rules	Only authenticated users can submit feedback. All required fields in the feedback form must be completed accurately.
Special Requirements	The system should support different types of feedback (e.g., general comments, issue reports). The system should maintain an audit log of all feedback submissions for accountability.
Notes and Issues	Consider implementing a feedback review process to ensure comments are addressed appropriately. Users should be encouraged to provide constructive feedback to help improve services.

Table 3. 12 Use case description for blood request

Use case id	UC#11
Use case name	blood request
Primary actors	Hospital
Description	Sending request for required blood unit, blood group. with the patient's name just for the identification of blood accepter
Goal	Asking blood from the blood bank for the patient.
Precondition	Hospital administrator is logged in and authenticated. Hospital has an active account registered in the blood bank system Hospital send request to blood bank.

Postconditions	The blood request is successfully submitted and logged in the system. The blood bank receives the request and processes it accordingly.
Normal Flow	1. Hospital administrator navigates to the blood bank's login page. 2. Administrator enters their valid username and password, then clicks the login button. 3. System authenticates the administrator's credentials and redirects them to the hospital admin dashboard. 4. On the hospital admin dashboard, the administrator sees a "Request Blood" link or menu option. 5. Administrator clicks the "Request Blood" option. 6. System displays a blood request form for the administrator to fill out. 7. Administrator provides the required information on the form (e.g., blood type needed, quantity, preferred delivery date, hospital details). 8. Administrator reviews the request details and clicks the "Submit Request" button. 9. System validates the submitted blood request information. 10. System processes the blood request and displays a confirmation message: "Blood request sent successfully."
Alternative Flows	If the administrator misses any required fields, the system prompts them to complete all required fields before submitting. If the entered information is invalid, the system displays an error message and prompts the administrator to correct the information.
Exceptions	If the system encounters an error during request submission, it displays an error message and logs the issue for further investigation. If the requested blood units are not available or insufficient, the blood bank notifies the hospital administrator accordingly.
Assumptions	Hospital administrator is authorized to request blood units. The blood bank system is operational and capable of processing blood requests.
Priority	High
Frequency of Use	Regularly, depending on the demand for blood units by hospitals.

Special Requirements	The system should support tracking and monitoring of blood requests and inventory levels. Notifications should be sent to the hospital administrator regarding the status of their blood request.
Notes and Issues	Consider implementing prioritization mechanisms for urgent blood requests. Hospital administrators should be informed about any delays or issues related to their blood requests.

Table 3. 13 Use case description for collect blood

Use case id	UC#12
Use case name	collect blood
Primary actors	Nurse
Description	Allows a nurse to record the collection of blood units from a donor in the blood bank system.
Goal	To accurately document the collection of blood units from donors for storage and distribution
Precondition	Nurse is logged in and authenticated. Nurse has the necessary permissions to record blood collections.
Postconditions	The blood collection is successfully recorded and stored in the system. The collected blood units are available for further processing and distribution.
Normal Flow	1. Nurse logs in to the blood bank's system using their valid credentials. 2. System authenticates the nurse's login and directs them to the nurse dashboard or homepage. 3. On the nurse dashboard, the nurse sees a "Collect Blood" link or menu option. 4. Nurse clicks the "Collect Blood" option.

	5. System displays a blood collection form for the nurse to fill out. 6. Nurse provides the required information on the form (e.g., donor's details, blood type, volume collected, date/time of collection). 7. Nurse reviews the collected blood information entered on the form. 8. Nurse clicks the "Submit Collection" button. 9. System validates the submitted blood collection details. 10. If the information is correct, the system records the blood collection and displays a confirmation message: "Blood collection recorded successfully."
Alternative Flows	If the nurse misses any required fields, the system prompts them to complete all required fields before submitting. If the entered information is invalid, the system displays an error message and prompts the nurse to correct the information.
Exceptions	If the system encounters an error during blood collection recording, it displays an error message and logs the issue for further investigation. If the collected blood is deemed unusable or contaminated, the nurse must follow appropriate protocols for disposal.
Assumptions	Nurse is authorized to record blood collections. The blood bank system is operational and capable of processing blood collection records.
Priority	High
Frequency of Use	Regularly, as blood collections occur throughout operational hours.
Business Rules	Only authorized nurses can record blood collections. Blood collection records should be accurate and complete for inventory management and donor tracking
Special Requirements	The system should support barcode scanning or other methods for accurate identification of blood units.

	Blood collection records should include donor consent status and any relevant notes from the collection process.
Notes and Issues	Training and protocols should be in place to ensure safe and hygienic blood collection practices. Nurses should be trained on the use of the blood collection form and the importance of accurate record-keeping.

Table 3. 14 Use case description for donor registration

Use case id	UC#13
Use case name	donor registration
actor	Donor
Description	This use case describes how a donor registers themselves in the blood bank system by filling out and submitting a registration form.
Precondition	The donor must have access to the internet and the blood bank's website.
Postconditions	A new donor account is created in the system if the registration information is valid.
Normal Flow	1. The donor navigates to the blood bank's registration page. 2. The donor fills out the registration form with the required information (e.g., name, contact details, age, gender, etc.). 3. The donor submits the registration form. 4. The system validates the submitted information. 5. If the information is correct, the system creates a new donor account. 6. The system displays a confirmation message: "Donor registered successfully."
Alternative Flows	None
Exceptions	If the information is invalid, the system displays an error message and prompts the donor to correct the information.
Assumptions	1. The donor provides accurate information during registration.

	2. The internet connection is stable throughout the registration process. 3. The registration form includes fields for necessary donor information. 4. Donors have basic computer literacy skills to navigate the registration process.
Priority	High
Frequency of Use	Regularly.
Business Rules	1. Donors must be of legal age to donate blood. 2. Donors must provide truthful information during registration. 3. Donor information is confidential and only accessible to authorized personnel.
Special Requirements	1. The registration form should be user-friendly and accessible on various devices. 2. The system should have measures in place to prevent duplicate registrations. 3. The system should encrypt and securely store donor information.
Notes and Issues	➢ Ensure that the registration form includes clear instructions and tooltips for each field to assist donors in providing accurate information. ➢ Regularly update the registration form based on feedback from donors to improve user experience. ➢ Conduct periodic reviews of registered donor accounts to ensure data accuracy and compliance with regulatory standards. ➢ Monitor system performance during peak registration periods to prevent potential bottlenecks or technical issues.

Table 3. 15 Use case description for blood distribution

Use case id	UC#14
Use case name	Distribution
Primary actors	Inventory manager
Description	Allows the inventory manager to distribute blood units to hospitals based on their requests.
Goal	To ensure timely and accurate distribution of blood units to hospitals for patient treatment.
Precondition	Inventory manager is logged in and authenticated. Adequate blood units are available in the inventory for distribution. Hospitals have submitted valid blood requests.
Postconditions	The requested blood units are successfully distributed to the respective hospitals. Inventory records are updated to reflect the distributed units
Normal Flow	1. Inventory manager logs in to the blood bank's system using their valid credentials. 2. System authenticates the manager's login and directs them to the inventory manager dashboard or homepage. 3. On the inventory manager dashboard, the manager sees a "Distribute Blood" link or menu option. 4. Manager clicks the "Distribute Blood" option. 5. System displays a blood distribution search interface, allowing the manager to filter and search for available blood inventory. 6. Manager enters the relevant search criteria (e.g., blood type, quantity, expiration date). 7. System retrieves and displays the matching blood inventory available for distribution. 8. Manager reviews the blood inventory details and selects the appropriate units for distribution. 9. Manager clicks the "Distribute" button for the selected blood units. 10. System processes the blood distribution, updates the inventory, and displays a

	confirmation message: "Blood distributed successfully."
Alternative Flows	no
Exceptions	If the system encounters an error during blood distribution, it displays an error message and logs the issue for further investigation. If the requested blood units are not available or insufficient, the inventory manager must inform the respective hospitals and coordinate alternative solutions
Assumptions	Inventory manager is authorized to manage stock levels. The blood bank system is capable of handling stock management tasks and updating inventory records in real-time.
Priority	High
Frequency of Use	Regularly, as hospitals require blood units for patient treatments.
Business Rules	Only authorized inventory managers can distribute blood units. Blood distribution should be based on accurate inventory levels and hospital requests. Distribution should prioritize urgent or critical requests.
Special Requirements	The system should support tracking and monitoring of blood distribution activities and inventory levels. Distribution records should include details such as recipient hospital, delivery date/time, and responsible personnel.
Notes and Issues	Inventory managers should coordinate closely with hospitals to ensure timely delivery and accurate fulfillment of blood requests. Any delays or issues in the distribution process should be communicated promptly to relevant stakeholders.

Table 3. 16 Use Case description for Manage stock

Use case id	UC#15
Use case name	Manage stock
Primary actors	Inventory manager
Description	Allows the inventory manager to manage the stock of blood units in the blood bank system, including updating, registering, and discarding units as necessary.
Goal	To ensure accurate tracking and maintenance of blood inventory levels for effective blood bank operations.
Precondition	Inventory manager is logged in and authenticated. Adequate permissions are granted to the inventory manager for stock management tasks. The blood bank system is operational and accessible.
Postconditions	Inventory records are updated to reflect changes in stock levels due to updates, registrations, or discards. Blood units are appropriately managed and accounted for within the system.
Normal Flow	1. Inventory manager logs in to the blood bank's system using their valid credentials. 2. System authenticates the manager's login and directs them to the inventory manager dashboard or homepage 3. On the inventory manager dashboard, the manager sees a "Manage Stock" link or menu option. 4. Manager clicks the "Manage Stock" option. 5. System presents the manager with three main actions for stock management: Update, Register, and Discard. 6. If the manager chooses to discard expired blood: a. System displays the current blood inventory, including expired units b. Manager selects the expired blood units they want to discard, typically by selecting the pack number

	c. Manager clicks the "Discard" button for the selected expired blood units 7. System processes the blood discarding, updates the inventory records, and displays a confirmation message: "Expired blood discarded successfully."
Alternative Flows	If the manager chooses to update stock levels or register new units, the system should provide appropriate forms or interfaces for entering the relevant information. If the manager encounters any discrepancies or issues in the stock management process, they must follow appropriate protocols for investigation and resolution. f the manager needs to search for specific blood units, the system should support filtering and searching functionalities.
Exceptions	If the system encounters an error during stock management tasks, it displays an error message and logs the issue for further investigation. If there are discrepancies between physical stock and system records, the manager must conduct a thorough inventory check and reconcile any differences.

Table 3. 17 Use case description for Notify Blood Donor

Use case ID	UC#16
Use Case Name	Notify Blood Donor
Actor	Admin
Description	The admin notifies donors when their blood is used in a successful transfusion.
Goal	To inform the donor that their blood has been used in a successful transfusion
Preconditions	Donor has donated blood. Donor's details are recorded in the system. Blood unit is assigned a unique ID. Hospital has reported the outcome of the blood transfusion.

Postconditions	Donor receives a notification about the successful use of their donated blood.
Normal Flow	1. Donor visits the blood bank and donates blood. 2. Blood bank records donor's details and assigns a unique ID to the blood unit. 3. Hospital requests blood for a transfusion. 4. Blood bank sends the appropriate blood unit to the hospital, including the donor's ID. 5. Hospital uses the blood for the patient's treatment. 6. Hospital reports back to the blood bank about the transfusion outcome. 7. Admin reviews the hospital's feedback and matches it with the original donor's ID. 8. Admin generates a notification message for the donor. 9. Admin sends the notification to the donor via their preferred contact method (e.g., email, SMS). 10. Donor receives the notification and is informed that their blood was used to save a life.
Alternative Flows	If the hospital does not report back, the admin does not send a notification to the donor.
Exceptions	If the donor's contact details are incorrect or missing, the notification cannot be sent. If the admin fails to match the hospital's feedback with the donor's ID, the notification is not generated.
Assumptions	Hospitals will report the outcome of blood transfusions accurately and in a timely manner. Donors' contact details are up-to-date and accurate
Priority	High
Frequency of Use	Multiple times per day, depending on the number of donations and transfusions.
Business Rules	Donors must be informed about the use of their donated blood.

	Notification must be sent promptly after receiving feedback from the hospital.
Special Requirements	The system must support multiple contact methods (email, SMS). The system must ensure data privacy and security for donor information.
Notes and Issues	Potential integration with hospital systems for automatic feedback reporting. Consideration for donors' preferences for receiving notifications (e.g., opt-in/opt-out options).

Table 3. 18 Use case description for: Send patient status

Use Case ID	UC#17
Use Case Name	Send Patient Status
Actor	Hospital Administrator
Description	Allows the hospital administrator to send the status of a patient who received a blood transfusion to the blood bank system.
Goal	To provide feedback to the blood bank on the outcome of blood transfusions for monitoring and record-keeping purposes.
Preconditions	- Hospital administrator is logged in and authenticated. The blood bank system is operational and accessible. The patient has received a blood transfusion using blood provided by the blood bank.
Postconditions	- The blood bank system records the patient's transfusion status. The blood bank system updates its records and may trigger notifications based on the provided status.
Normal Flow	1. Hospital administrator logs in to the blood bank's system using their valid credentials 2. System authenticates the administrator's login and directs them to the hospital admin dashboard. 3. On the hospital admin dashboard, the administrator sees a "Send Patient Status" link or menu option. 4. Administrator clicks the "Send Patient Status" option. 5. System displays a patient status form for the administrator to fill out.

	6. Administrator enters the required information, such as patient ID, blood unit ID, status of the transfusion (successful/unsuccessful), and any additional comments. 7. Administrator reviews the entered information for accuracy. 8. Administrator clicks the "Submit" button to send the patient status. 9. System validates the submitted information. 10. If the information is correct, the system records the patient status and displays a confirmation message: "Patient status sent successfully."
Alternative Flows	- If the administrator misses any required fields, the system prompts them to complete all required fields before submitting. If the entered information is invalid, the system displays an error message and prompts the administrator to correct the information.
Exceptions	- If the system encounters an error during the submission of patient status, it displays an error message and logs the issue for further investigation. If there are discrepancies in the patient or blood unit ID, the administrator must resolve these issues before submitting the status.
Assumptions	- Hospital administrator is authorized to send patient status updates. The blood bank system can receive and process patient status updates.
Priority	High
Frequency of Use	Regularly, as patient status updates are sent following each blood transfusion.
Business Rules	- Only authorized hospital administrators can send patient status updates. Patient status updates should be accurate and submitted promptly after the transfusion.
Special Requirements	- The system should support secure transmission and storage of patient status information. Patient status updates should be logged and auditable for compliance and monitoring purposes.
Notes and Issues	- Hospital administrators should be trained on the importance of accurate and timely patient status reporting. Regular reviews should be conducted to ensure the process of sending patient status updates is followed correctly and consistently.

3.1.4 **Activity diagram**

An activity diagram is a type of UML (Unified Modeling Language) diagram that is used to model the flow of activities or actions in a system. It shows the sequential and parallel activities performed as part of a business process or workflow [2]. We have provided activity diagram as follows: -

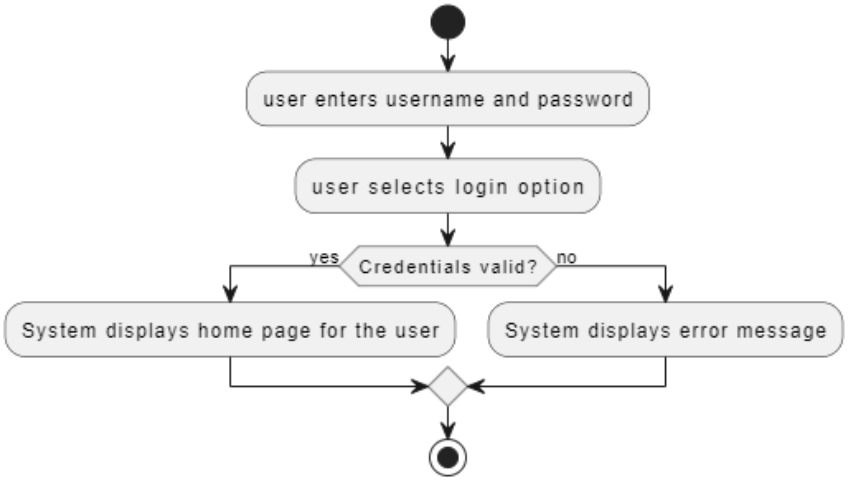

Figure 3. 2 Activity diagram for login

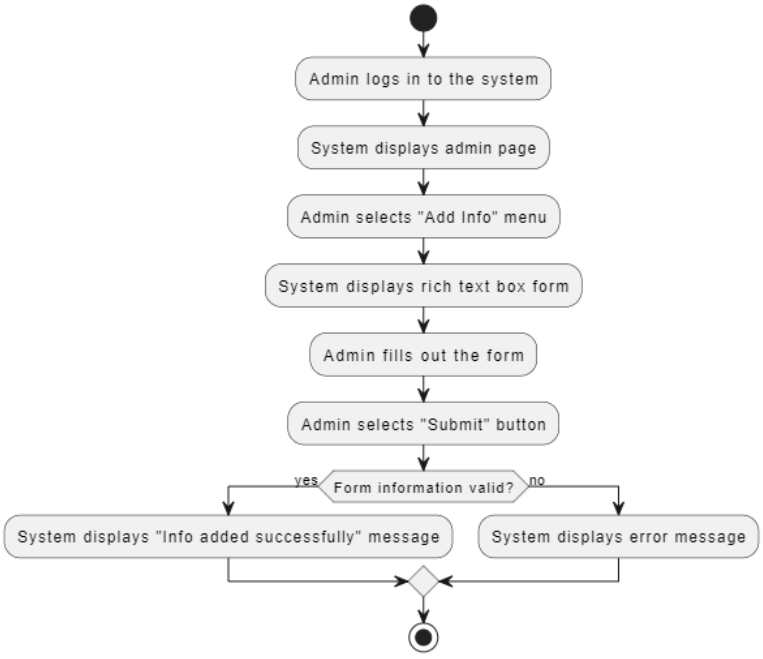

Figure 3. 3 Activity diagram for post info

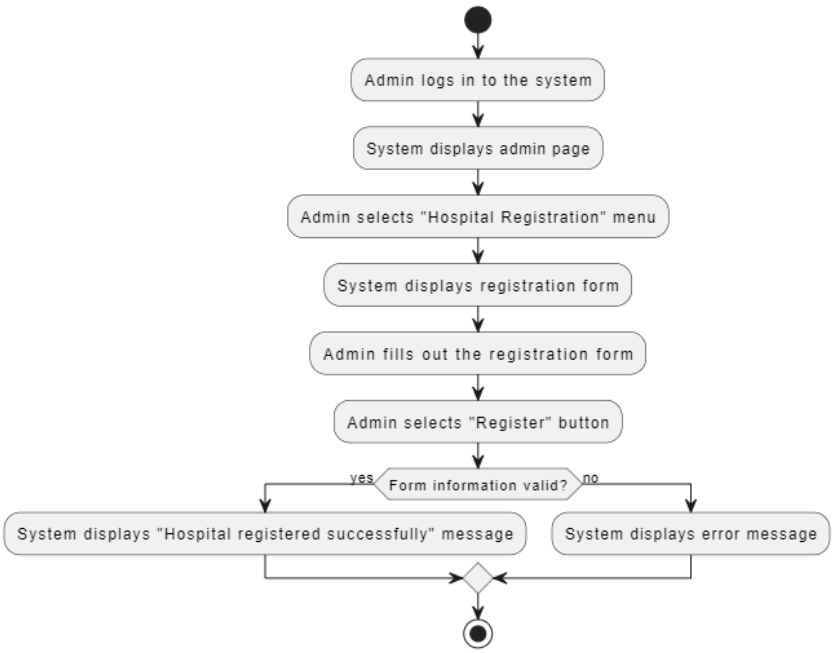

Figure 3. 4 Activity diagram for hospital registration

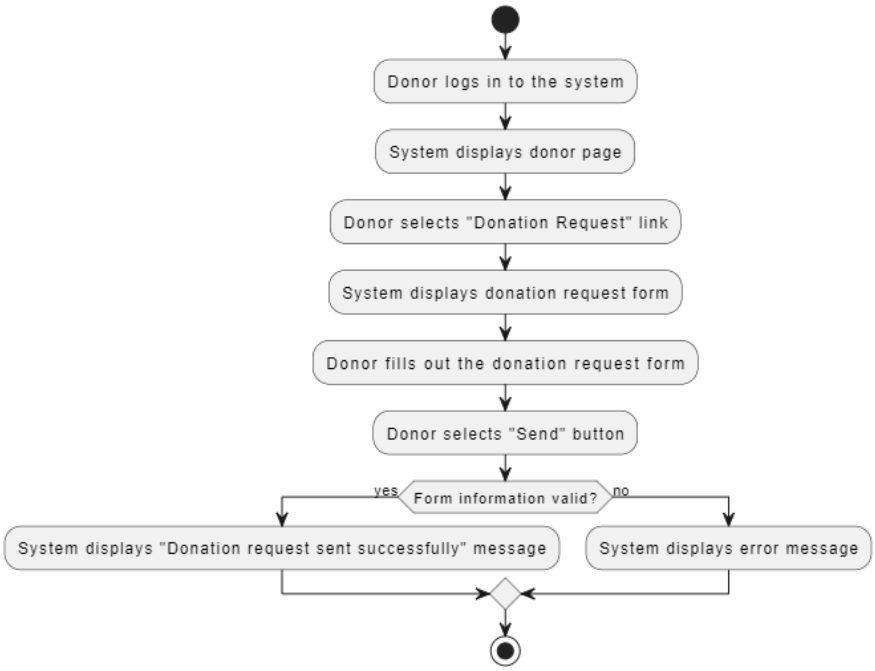

Figure 3. 5 Activity diagram for donation request

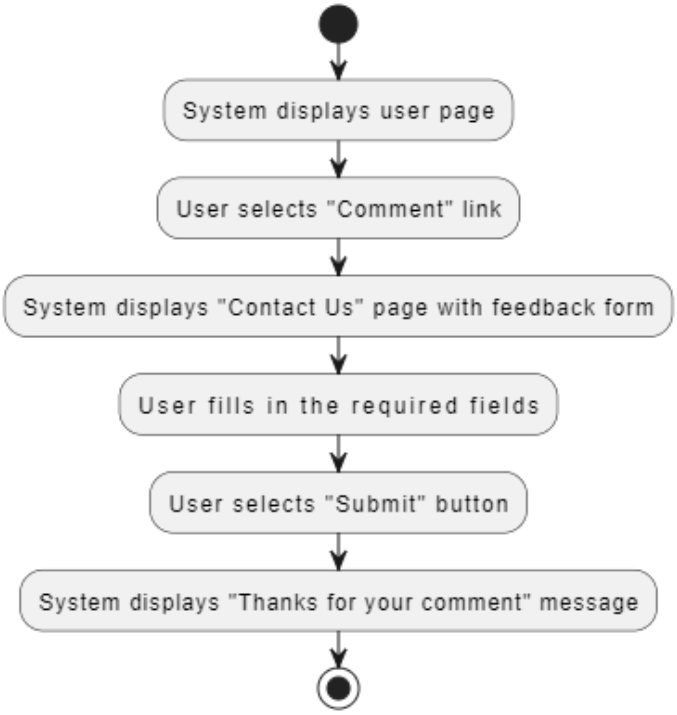

Figure 3. 6 Activity diagram for give comment

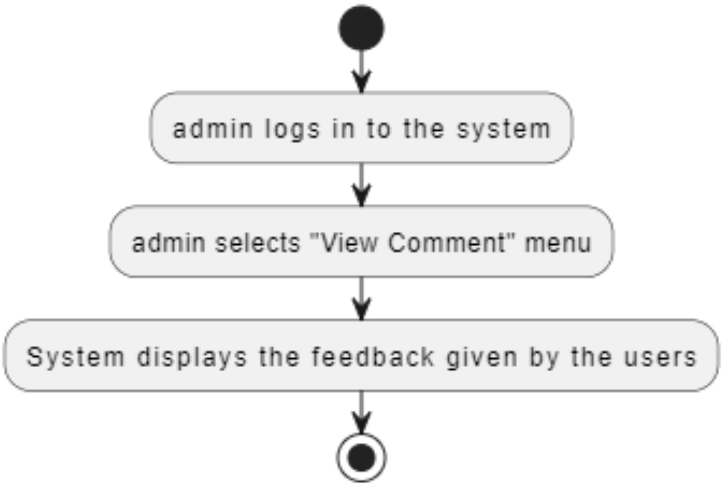

Figure 3. 7 **Activity diagram for view comment**

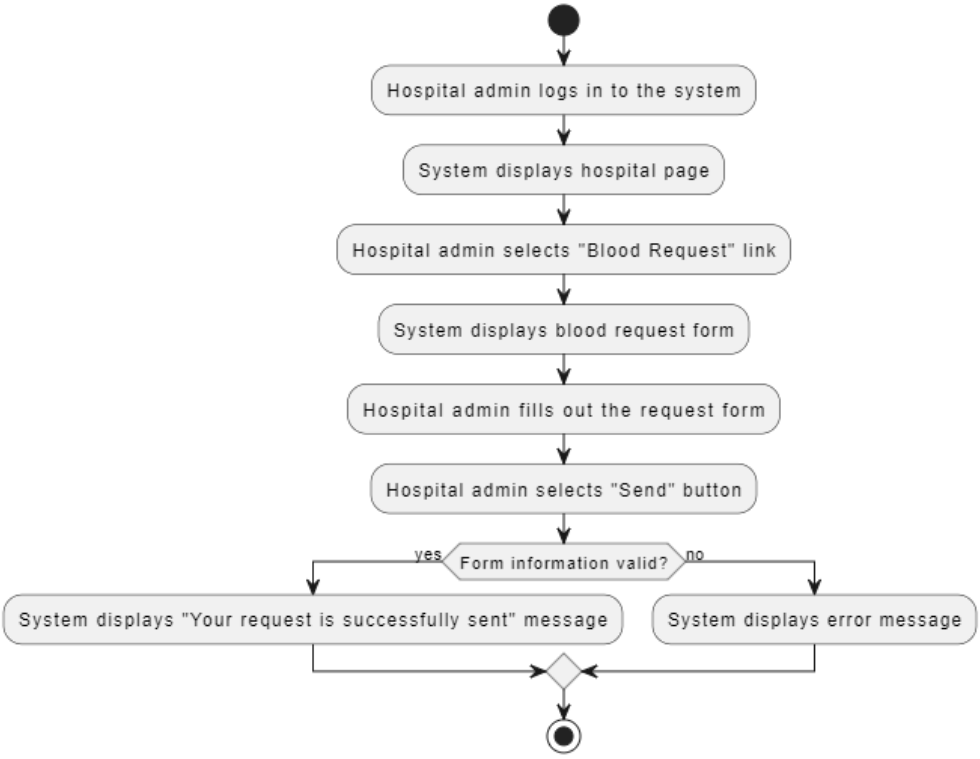

Figure 3. 8 **Activity diagram for blood request**

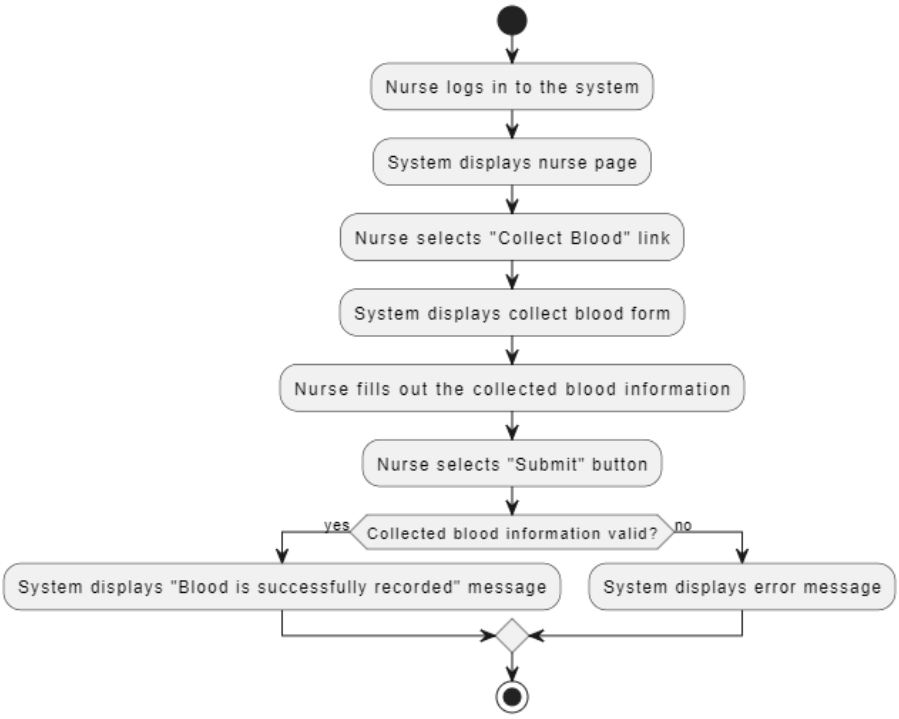

Figure 3. 9 **Activity diagram for collect blood**

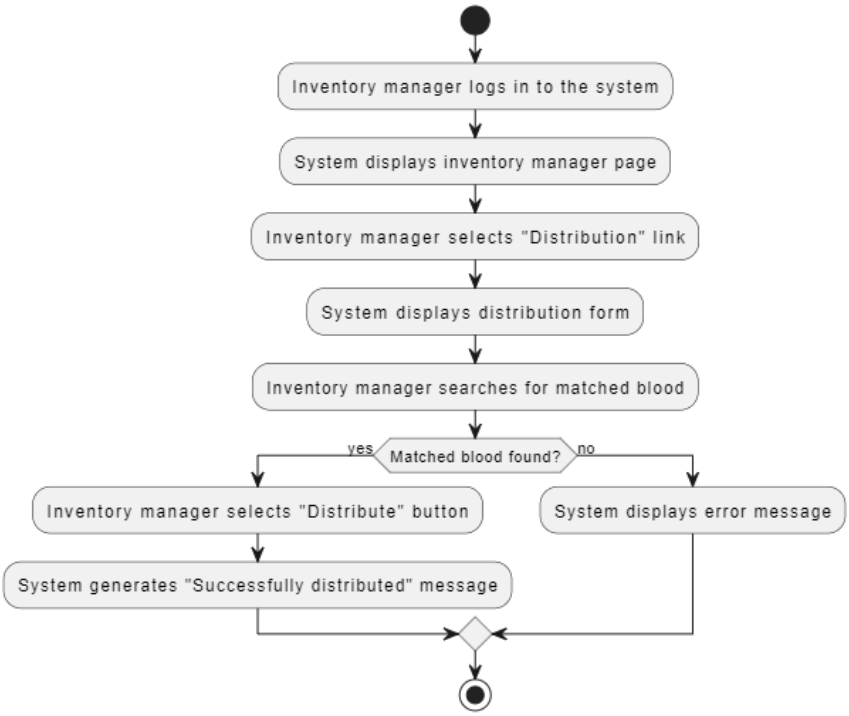

Figure 3. 10 **Activity diagram for distribution**

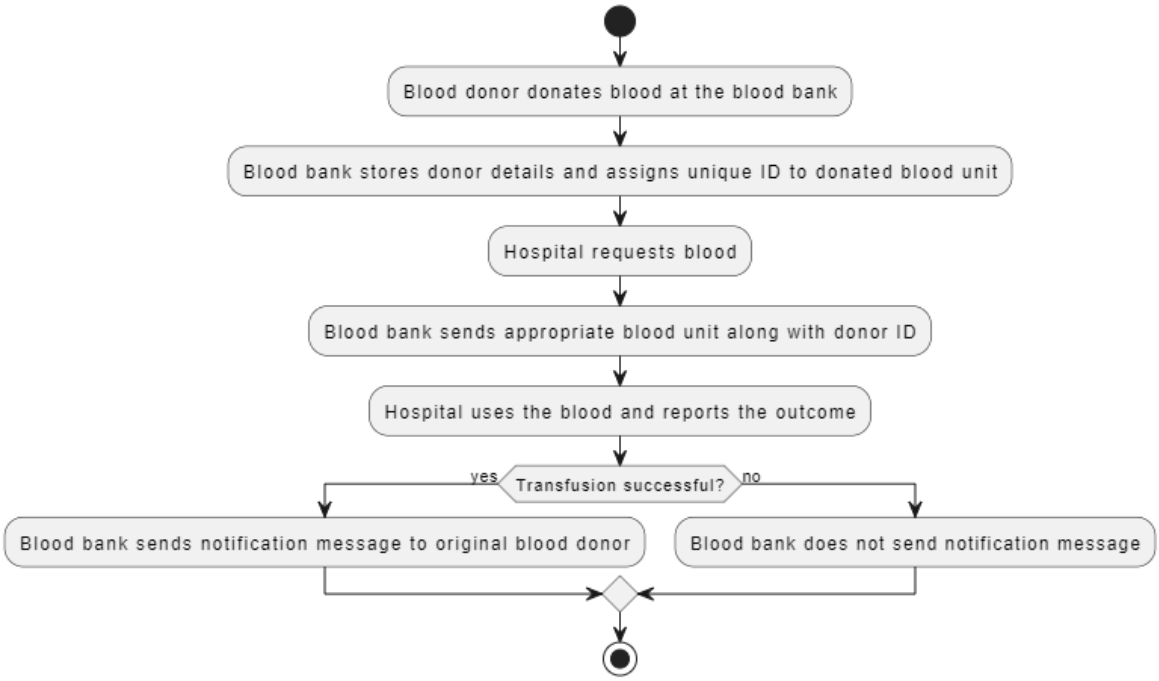

Figure 3. 11 **Activity diagram for notify blood donor**

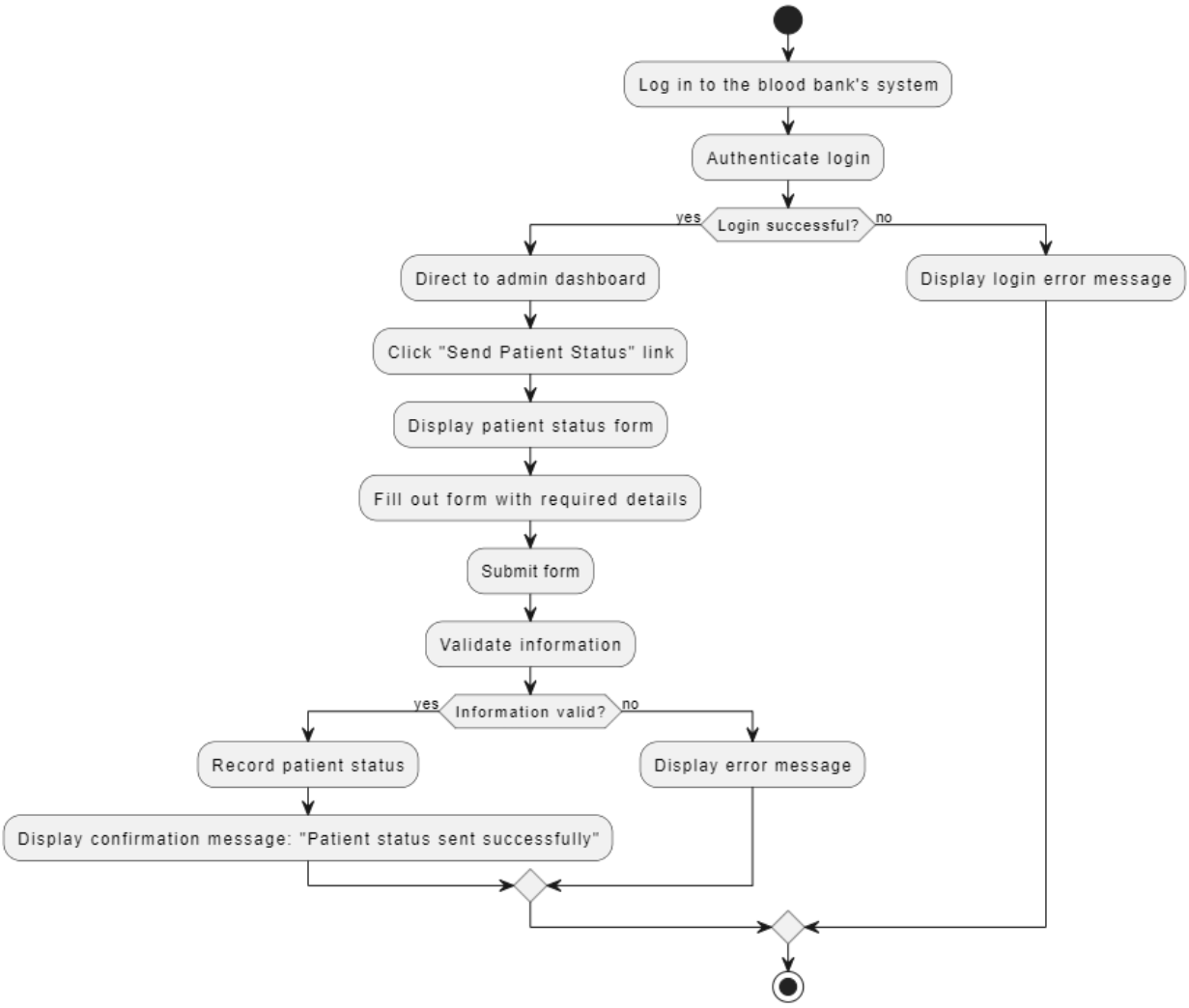

Figure 3. 12 **Activity Diagram for send patient status**

3.1.5 **Object model**

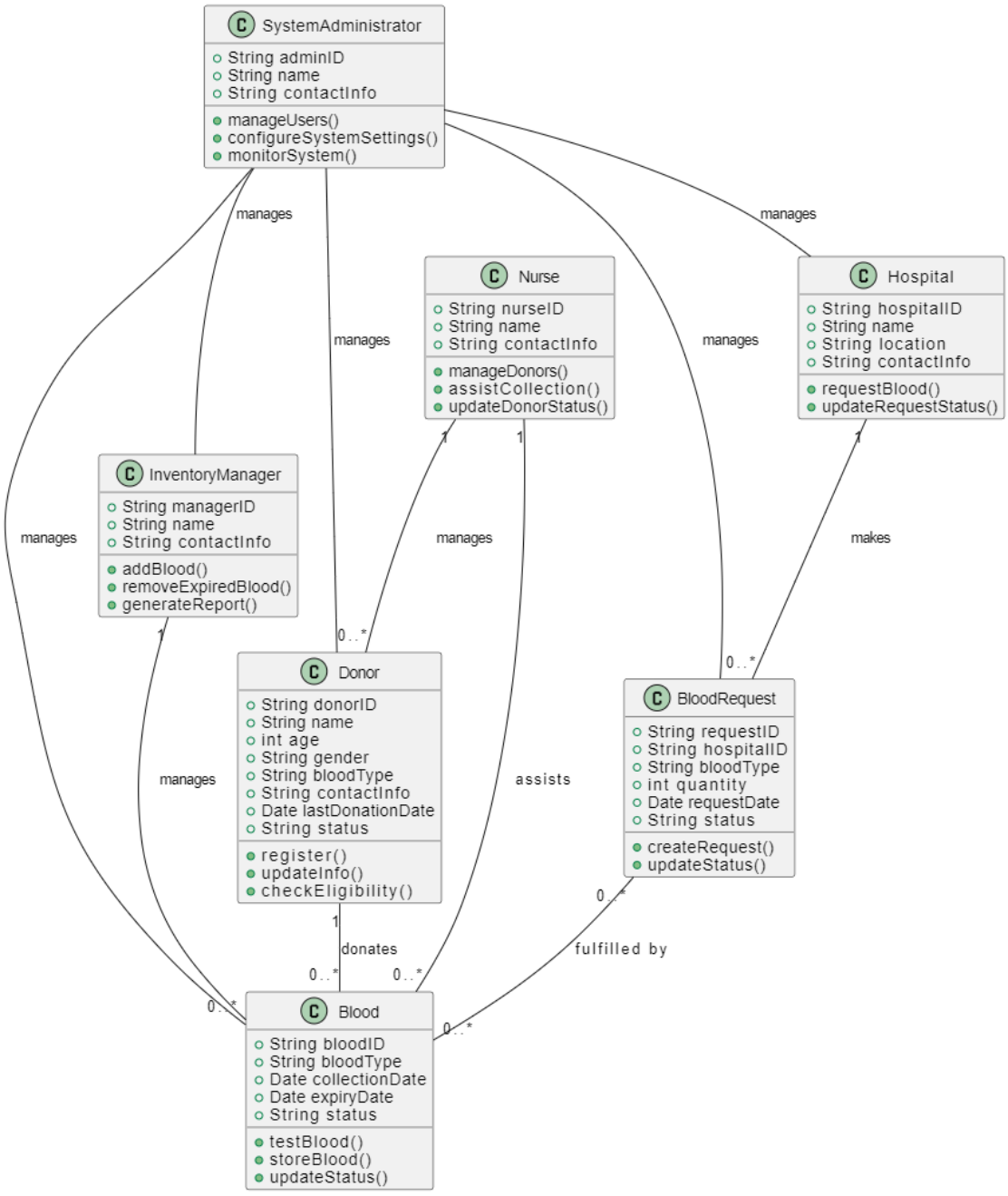

Figure 3. 13 Object Model

3.1.6 **Data dictionary**

Table 3. 19 Data dictionary for Admin

Primary/Foreign key	Field name	caption	Data type	Field size
P	ID	ID	string	5
	Fname	First Name	Text	20
	Lname	Last Name	Text	20
	Age	Age	Integer	2
	Email	Email	string	11
	Ph.no	Phone number	string	13

Table 3. 20 Data dictionary for Donor

Primary/Foreign key	Field name	Caption	Data type	Field size
P	ID	ID	String	5
	Fname	First name	Text	20
	Lname	Last name	Text	20
	Age	Age	Integer	2
	Date of birth	Date of birth	String	10
	Sex	Gender	Text	4
	Zone	Zone	String	10
	Woreda	Woreda	Text	10
	Kebele	Kebele	String	12
	Ph.no	Phone number	string	13
	City	City	String	15
	Email	Email	String	16

Table 3. 21 Data dictionary for Nurse

Primary/Foreign key	Field Name	Caption	Data Type	Field Size
P	ID	ID	String	5
	Fname	First Name	Text	13
	Lname	Last Name	Text	14
	Email	Email	String	18

Table 3. 22 Data dictionary for hospital

Primary/Foreign key	Field Name	Caption	Data Type	Field Size
P	ID	ID	String	5
	Hname	Hospital Name	Text	10
	Fax	Fax	Integer	18
	Ph.no	Phone number	String	13

Table 3. 23 Data dictionary for inventory manager

Primary/Foreign key	Field name	caption	Data type	Field size
P	ID	ID	string	5
	Fname	First Name	Text	20
	Lname	Last Name	Text	20
	Age	Age	Integer	2
	Email	Email	string	11
	Ph.no	Phone number	string	13

3.1.7 Class model

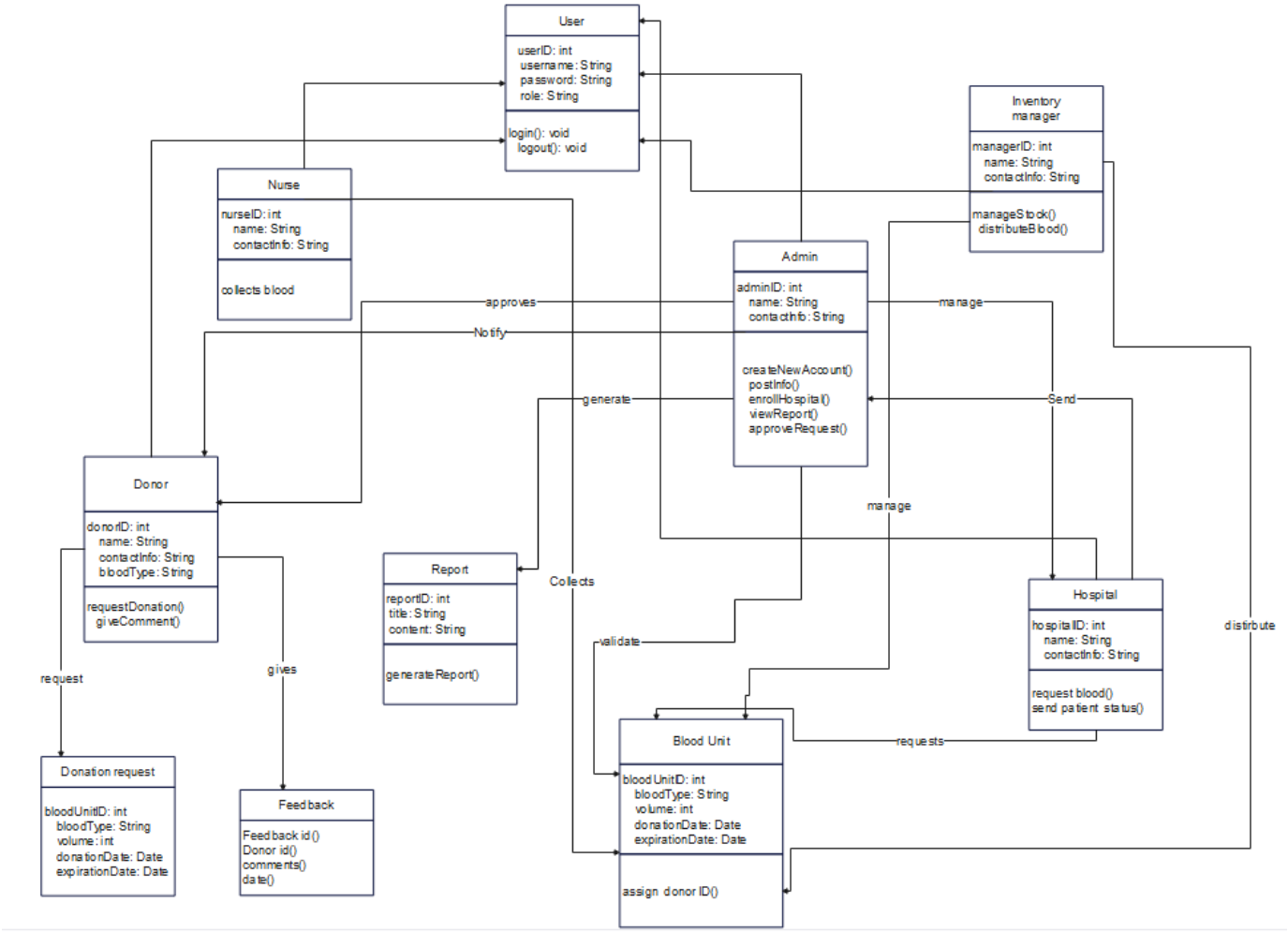

Figure 3. 14 class diagram

3.1.8 Dynamic modeling

The dynamic model is used to represent and capture the behavioral aspects of the system as it evolves over time [3].

State diagrams

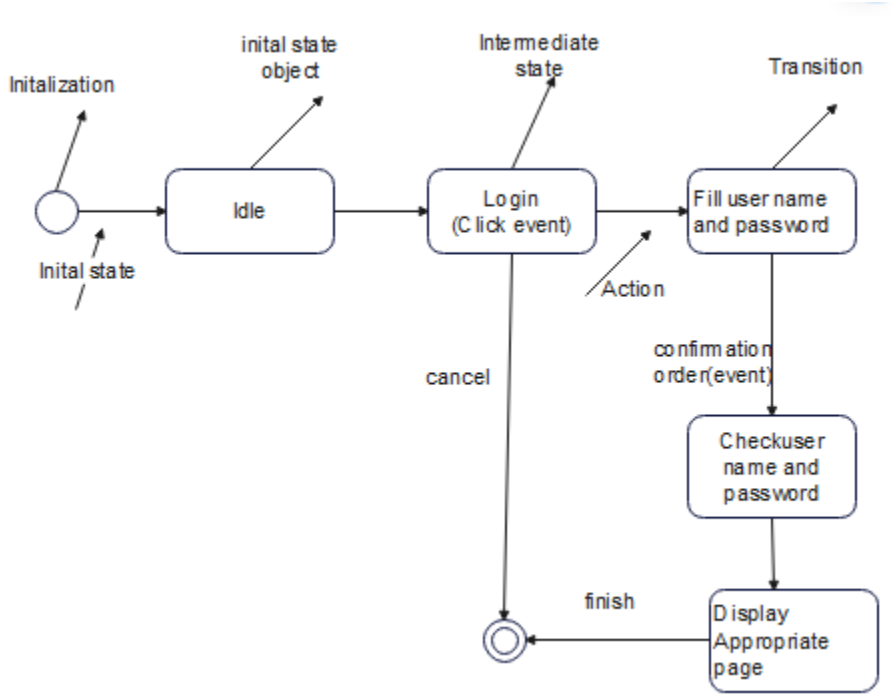

Figure 3. 15 State diagram for login

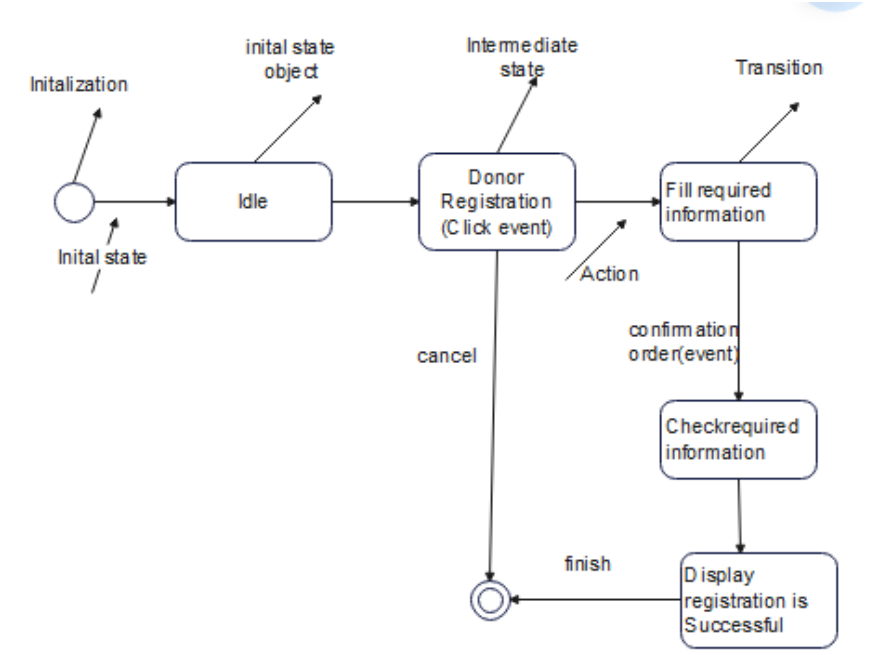

Figure 3. 16 State diagram for donor registration

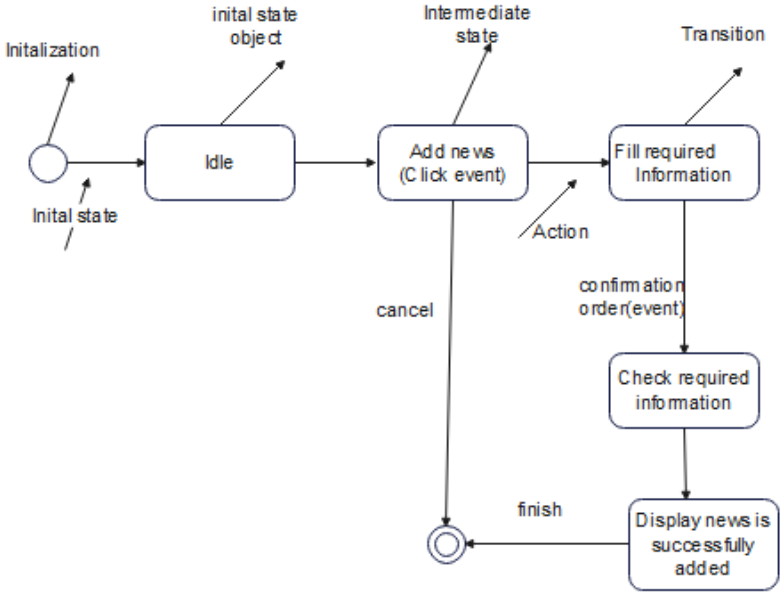

Figure 3. 17 State diagram for add new information

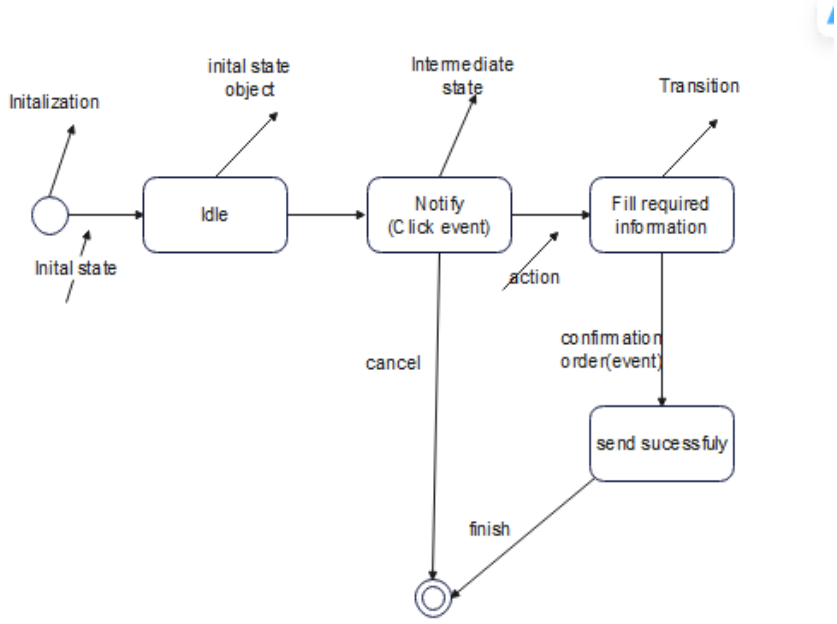

Figure 3. 18 State diagram for notify donor

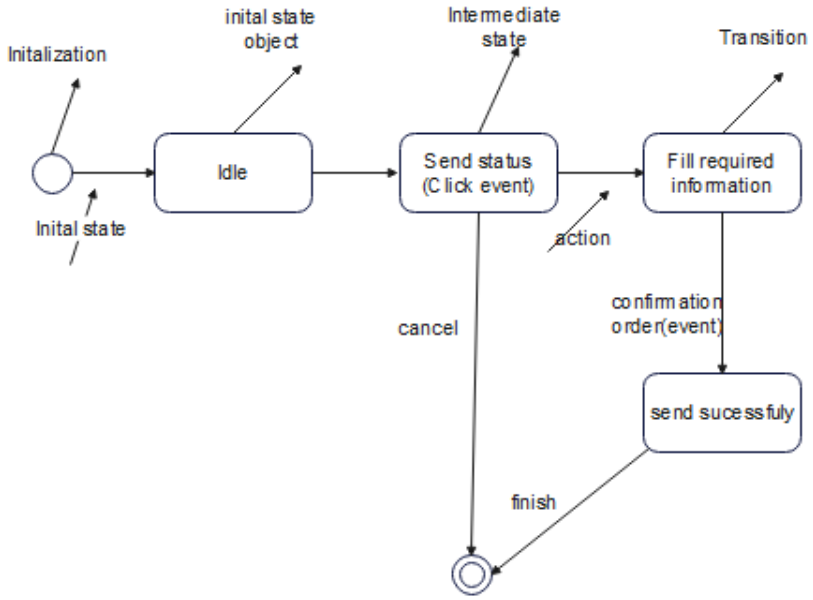

Figure 3. 19 State diagram for send patient status

Sequence diagram

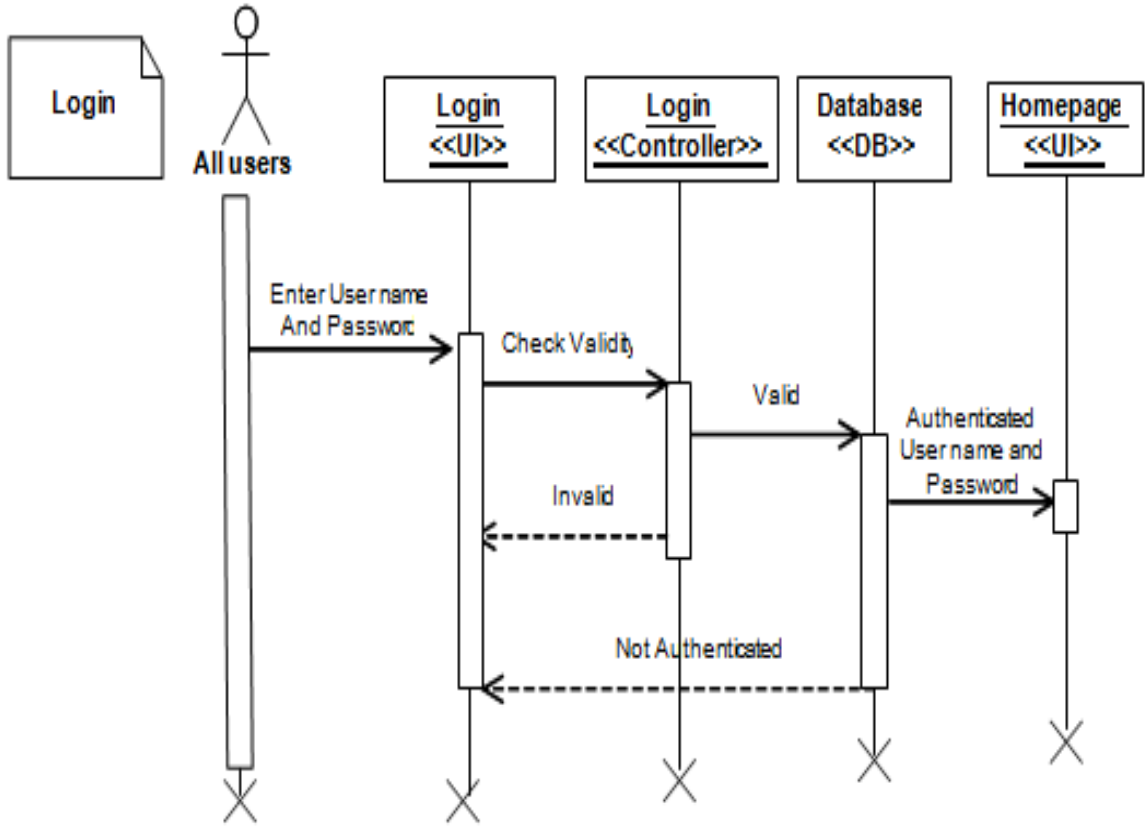

Figure 3. 20 Sequence diagram for login

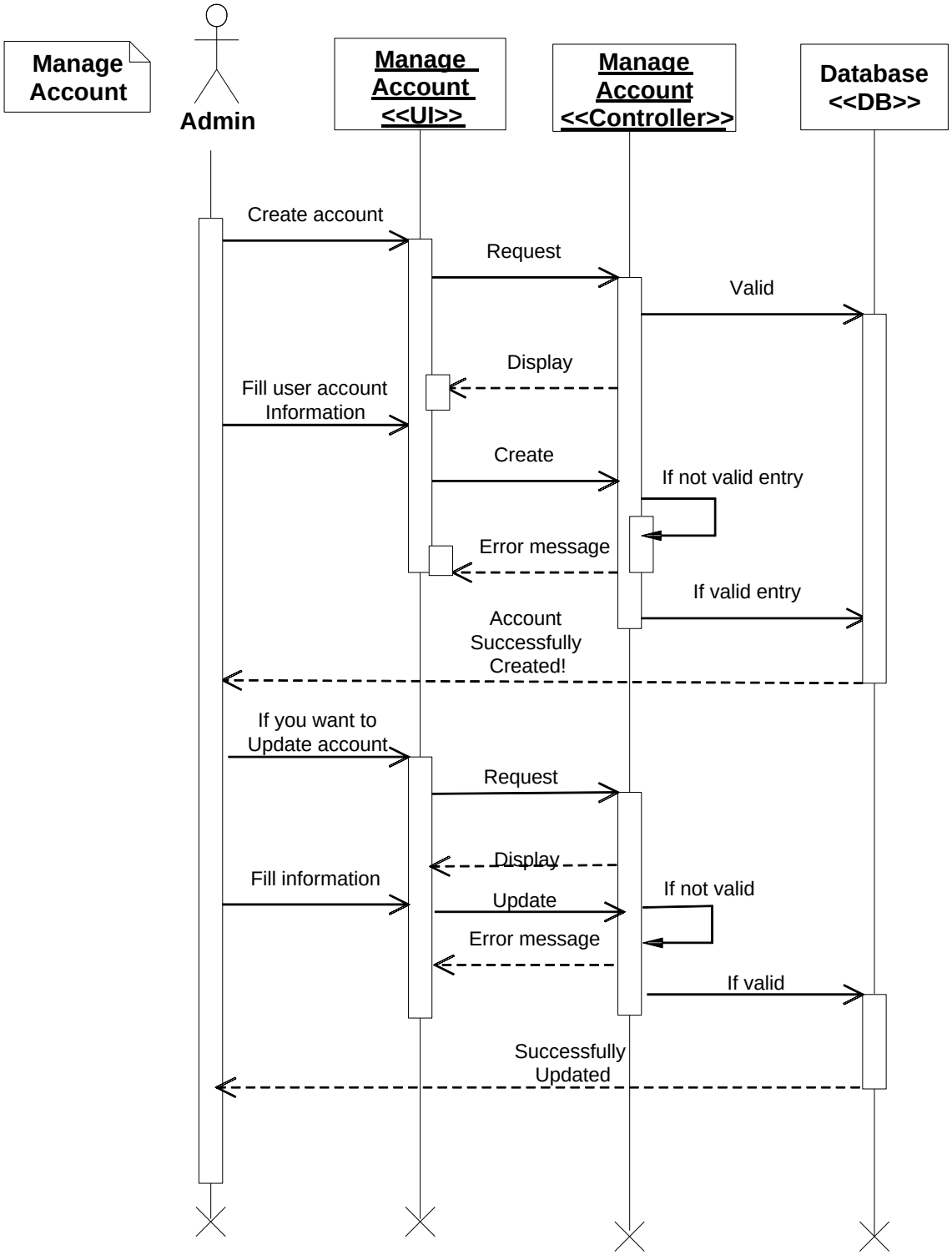

Figure 3. 21 Sequence diagram for manage account

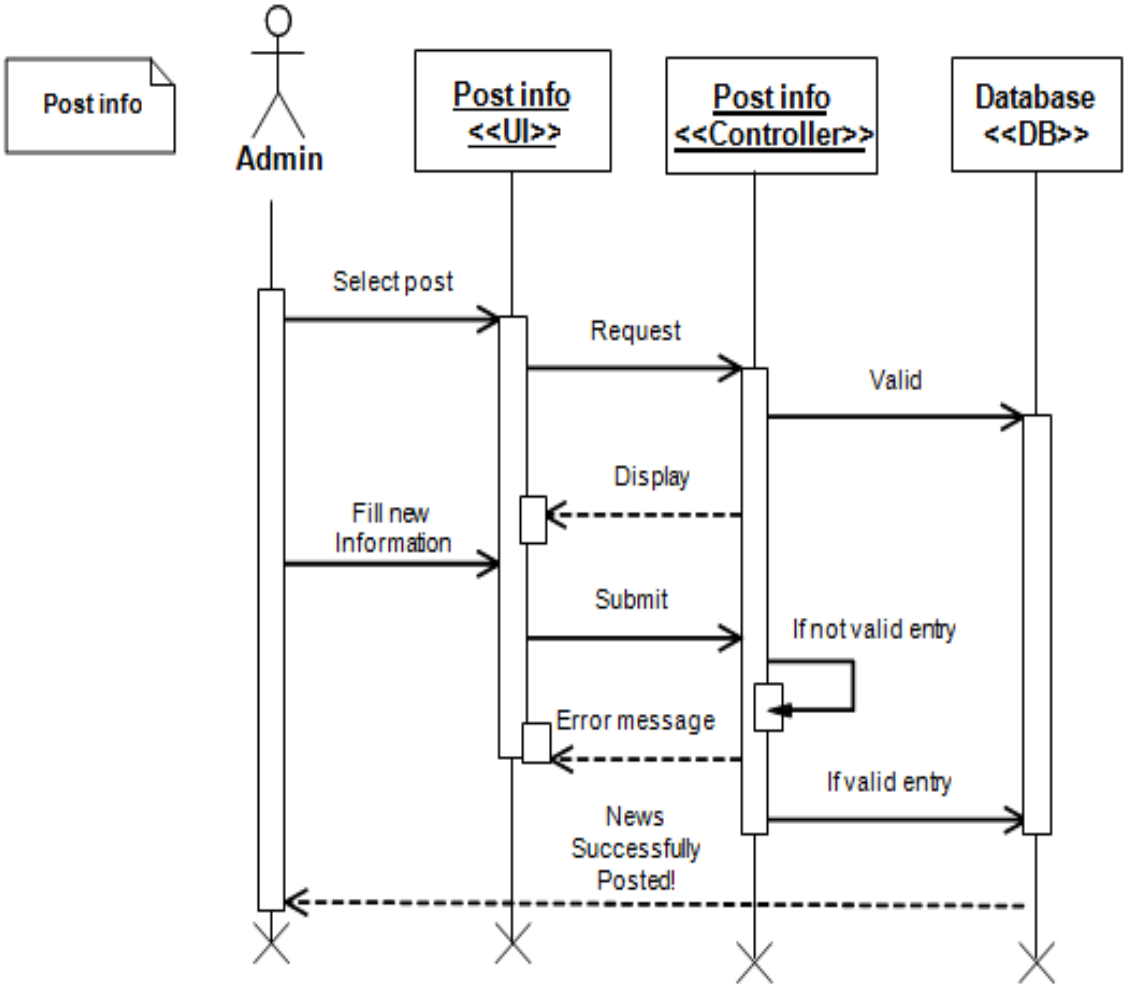

Figure 3. 22 Sequence diagram for post information

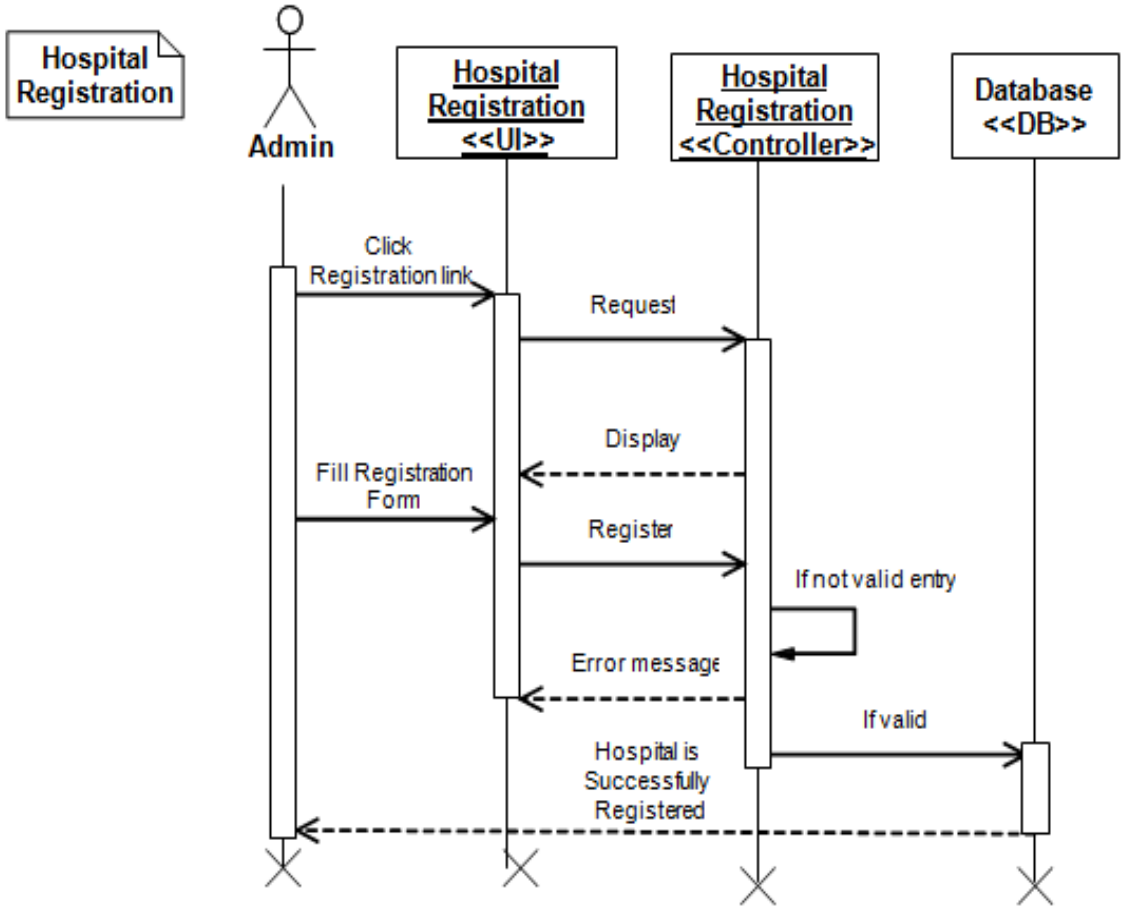

Figure 3. 23 Sequence diagram for hospital registration

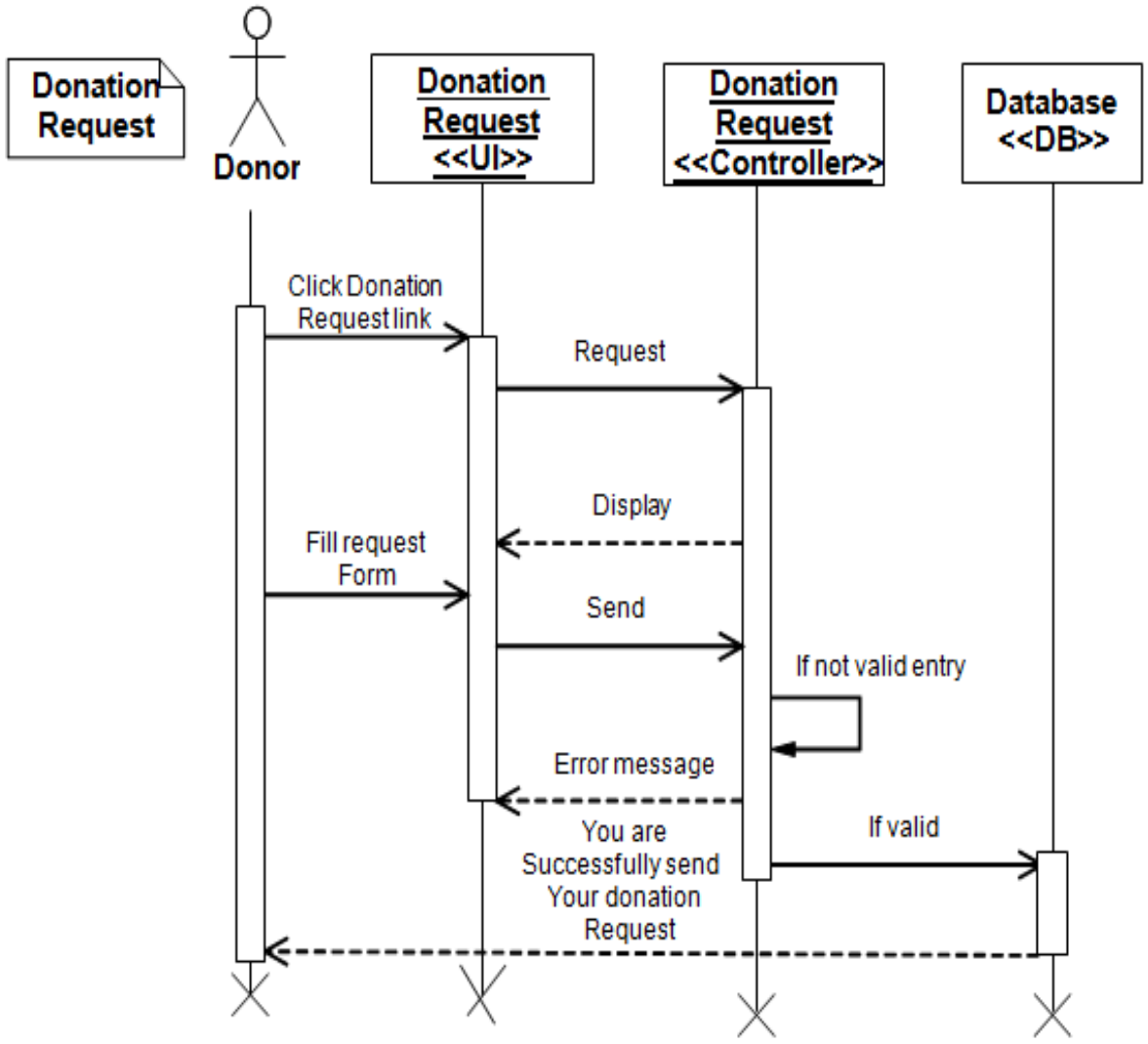

Figure 3. 24 Sequence diagram for donation request

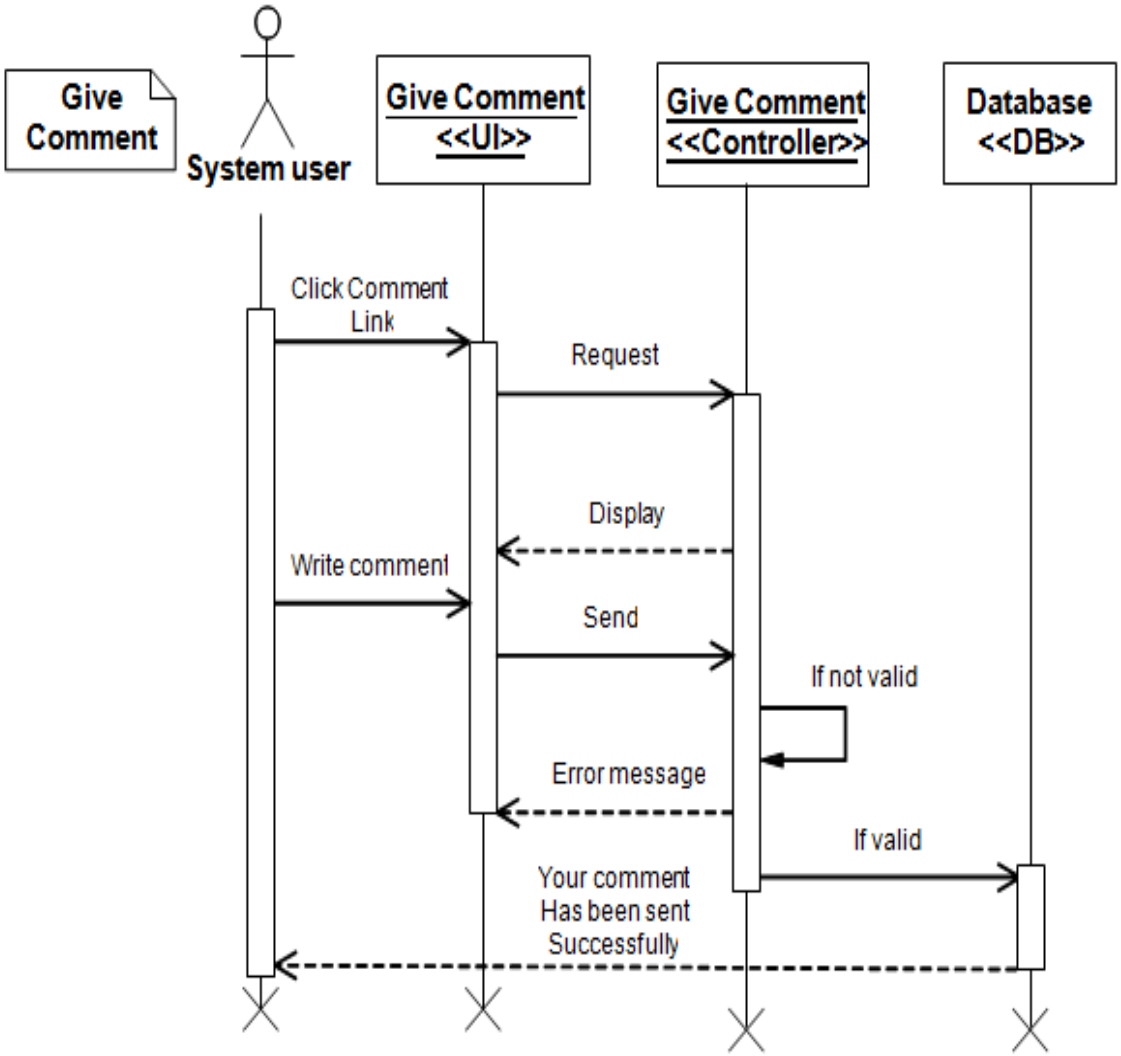

Figure 3. 25 Sequence diagram for give comment

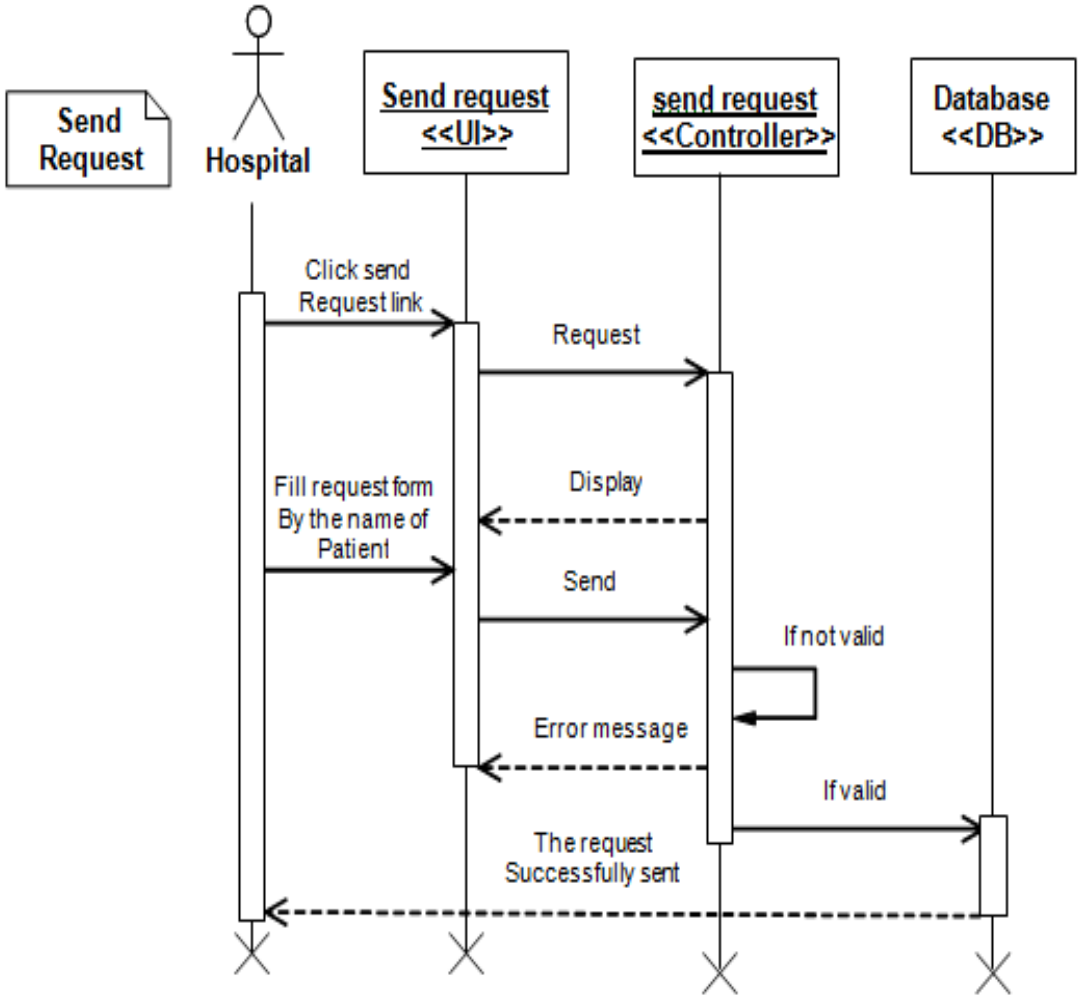

Figure 3. 26 Sequence diagram for send blood request

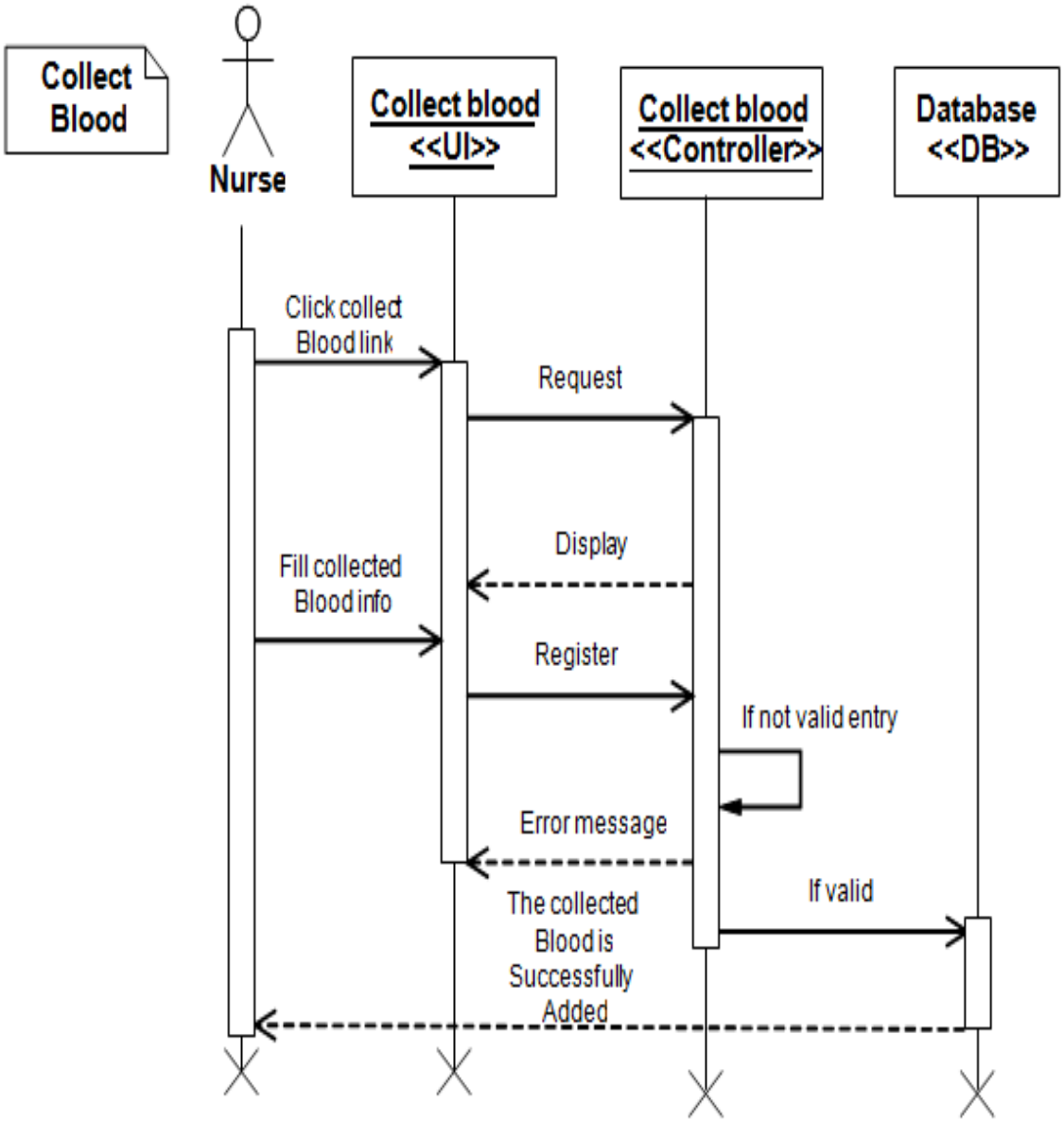

Figure 3. 27 Sequence diagram for collect blood

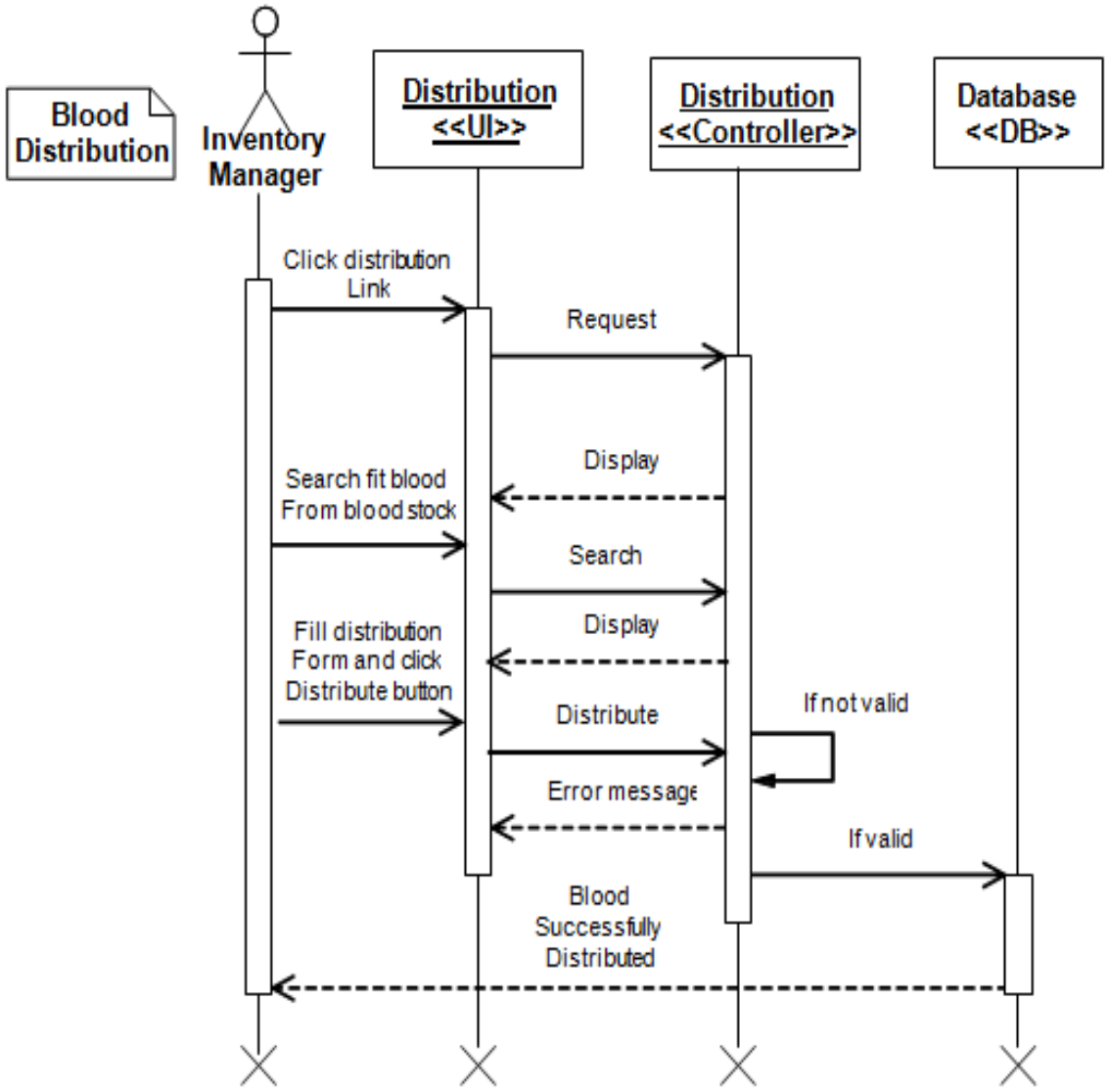

Figure 3. 28 Sequence diagram for blood distribution

3.1.9 User Interface

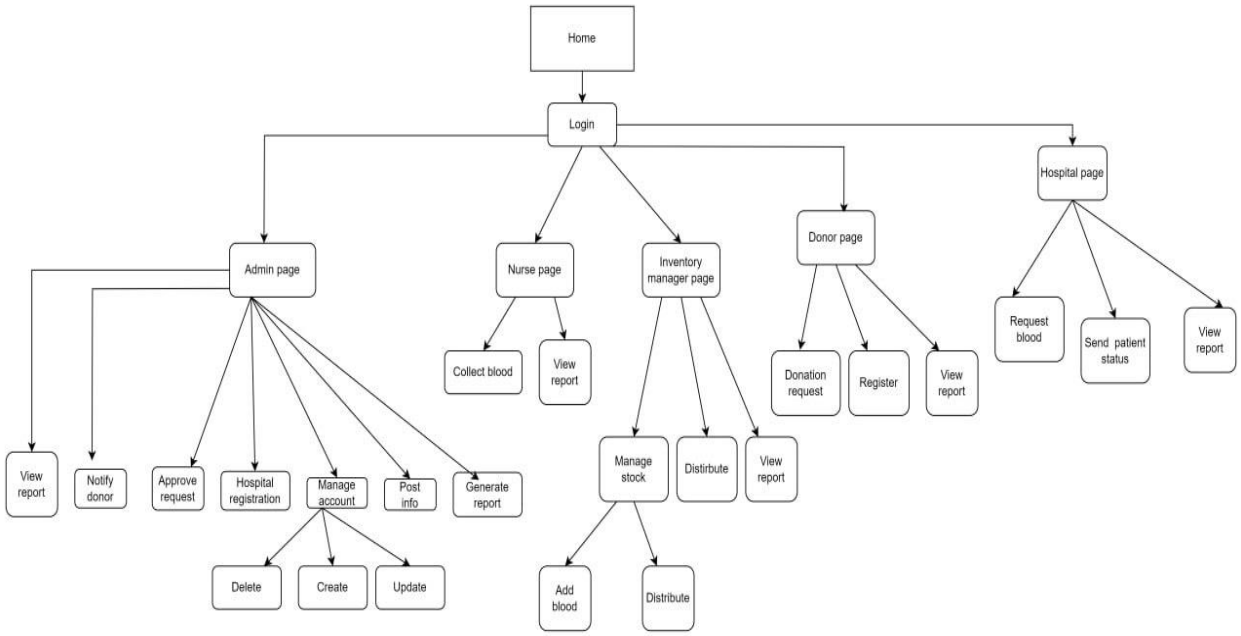

Figure 3. 29 User Interface

Chapter 4 system design

4.1 Introduction

In this chapter the overall procedures, activities and methods of execution during the implementation phase of the project are included

4.2 Current software architecture

The current Gondar blood bank system operates using a manual process, without any existing software architecture in place. Therefore, this proposal will focus solely on describing the software architecture for the new, proposed blood bank system, rather than any existing digital infrastructure.

In other words, since the current system is paper-based, there is no previous software system to build upon. This proposal will outline the architecture for an entirely new, digital blood bank management system.

4.3 Proposed software architecture

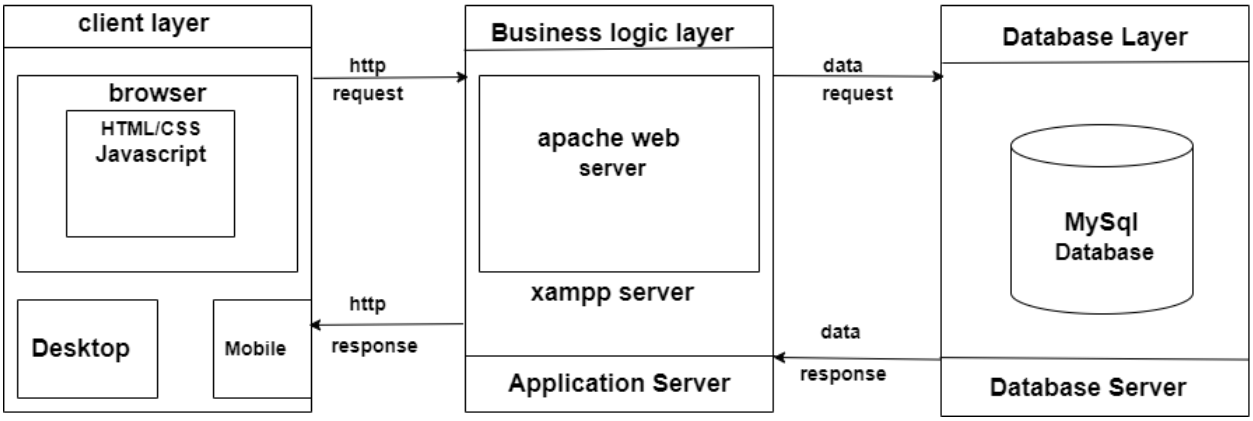

Figure 4. 1 Proposed Software architecture

4.3.1 System decomposition

system decompositions will help reduce the complexity of the system [4]. The systems can be considered as packages holding related classes/objects. These systems are further decomposed into other subsystems. The major subsystems identified are “Registration”, “Login”, “Screening”, “Donate Blood”, “Blood Distribution”, “Blood Collection”, “manage stock”, “notify donor” and” send patient status” subsystems. Users are classified in to roles. The “Login” subsystem authenticates a user to grant access based on the role of the user.

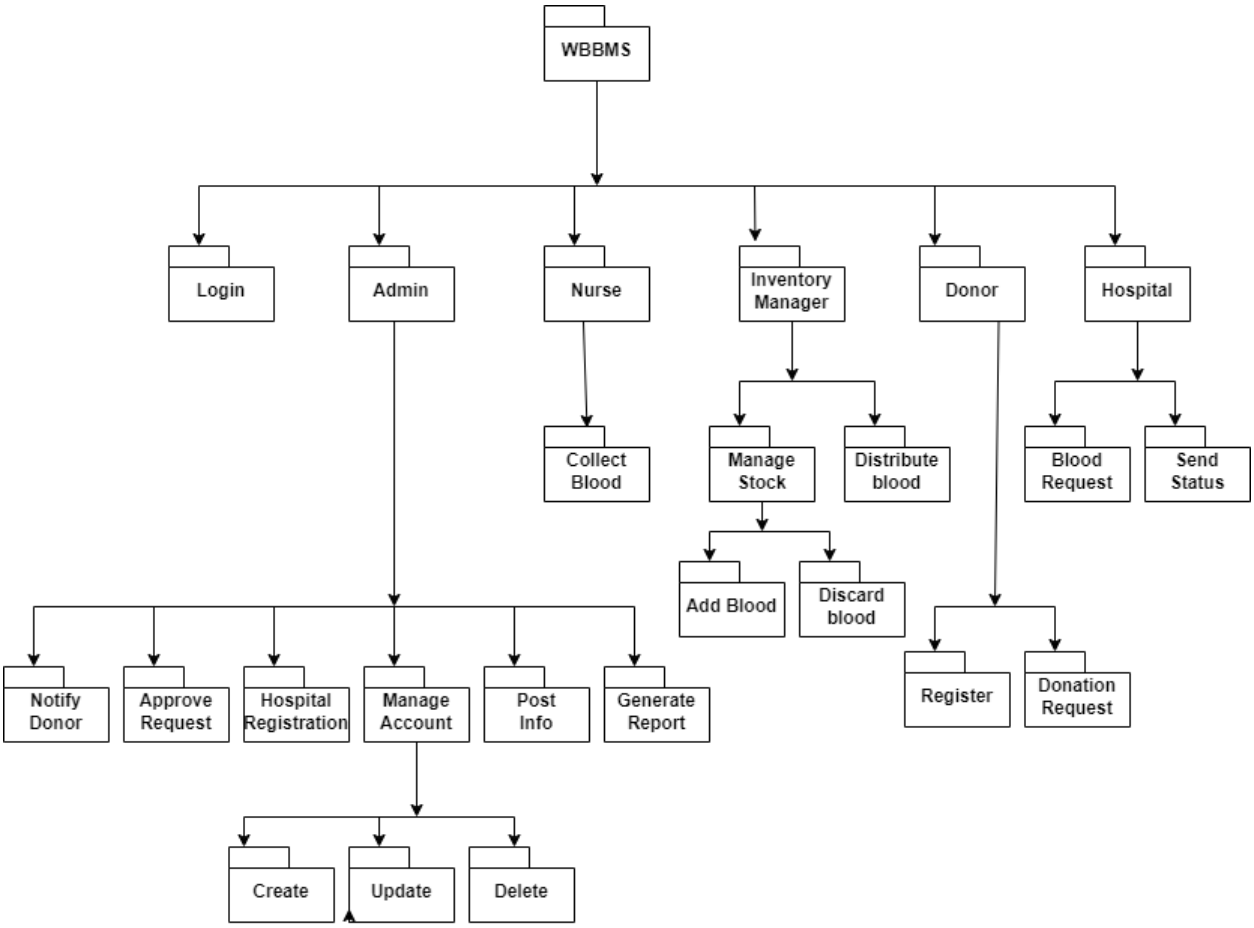

Figure 4. 2 System decomposition

4.3.2 **Component diagram**

A component diagram, a distinct type within the Unified Modeling Language (UML), serves a unique purpose compared to other UML diagrams [5]. Rather than illustrating the system's functionality, it describes the components essential for realizing those functionalities. Thus, component diagrams serve to visually represent the physical elements within a system.

Considered as a static implementation view, component diagrams offer insight into how components are organized within the system at a specific instance. It's important to note that a single component diagram cannot encapsulate the entirety of the system. Instead, a collection of these diagrams is employed to comprehensively depict the system's architecture and composition [6].

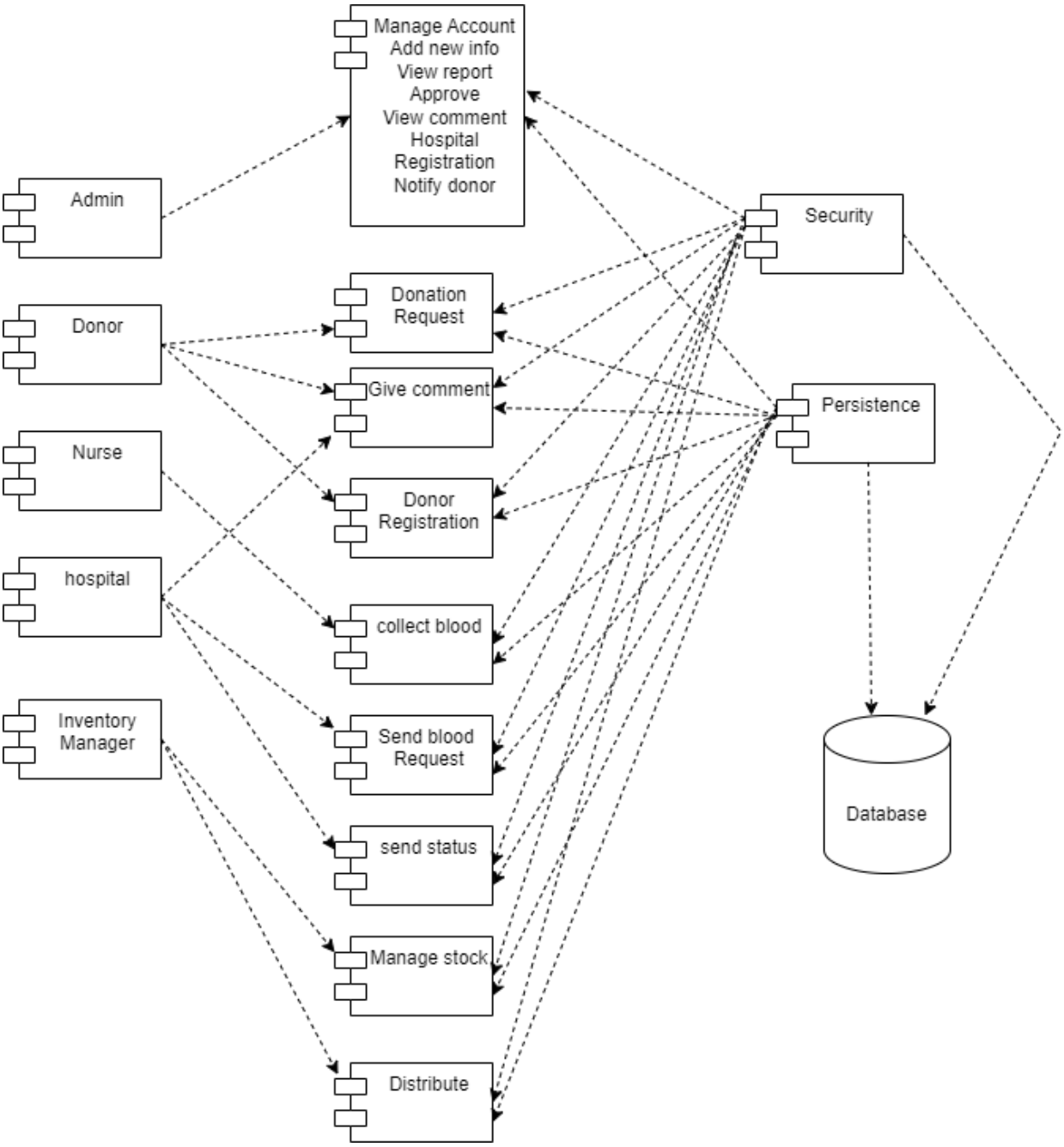

Figure 4. 3 Component diagram

4.3.3 **Hardware/ software mapping**

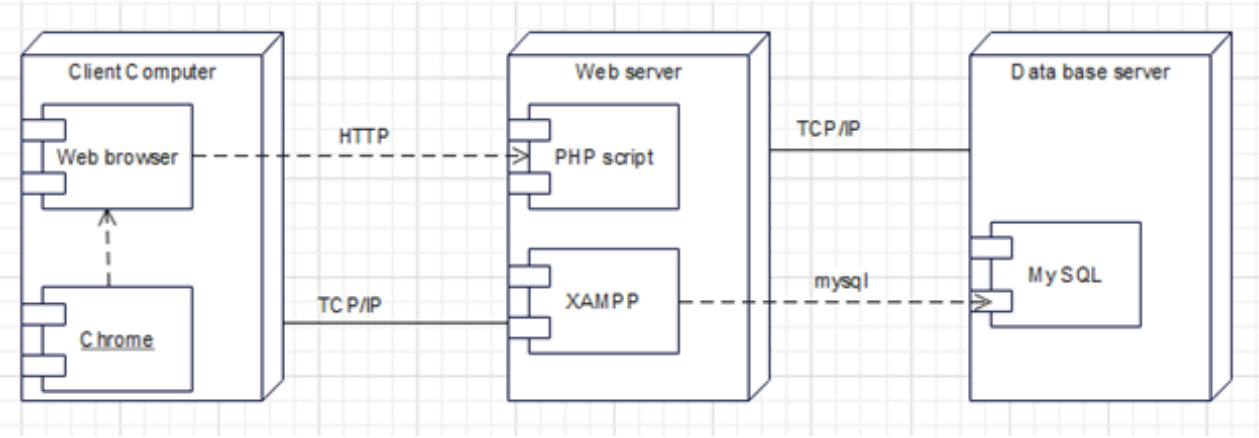

Figure 4. 4 Hardware /software mapping

4.3.4 **PERSISTENCE DATA MODELING**

Persistence data modeling, on the other hand, is the process of designing the conceptual, logical, and physical data models that represent the structure and relationships of the data that needs to be persisted [7].

The persistence data model for the blood management system revolves around the core entities of Users, Blood Donations, Blood Requests, Blood Inventory, Hospitals, and Patients.

let's go through the relationship details of the entities in the persistence data model:

User and Account:

One-to-many relationship: A user can have one or more accounts, but each account belongs to a single user.

Relationship type: User ||--o{ Account

User and Blood Donation:

One-to-many relationship: A user (as a donor) can make multiple blood donations, but each blood donation is associated with a single donor.

Relationship type: User ||--o{ Blood Donation

User and Blood Request:

One-to-many relationship: A user (as a requestor) can make multiple blood requests, but each blood request is associated with a single requestor.

Relationship type: User ||--o{ Blood Request

User and Feedback:

One-to-many relationship: A user can provide multiple feedback/comments, but each feedback/comment is associated with a single user.

Relationship type: User ||--o{ Feedback

Blood Donation and User:

Many-to-one relationship: Each blood donation is associated with a single donor (user), but a user can make multiple blood donations.

Relationship type: Blood Donation o--|| User

Blood Request and User:

Many-to-one relationship: Each blood request is associated with a single requestor (user), but a user can make multiple blood requests.

Relationship type: Blood Request o--|| User

Blood Inventory and Hospital:

One-to-many relationship: The blood inventory is managed by the blood bank, which can distribute blood to multiple hospitals.

Relationship type: Blood Inventory ||--|| Hospital

Patient and Hospital:

One-to-many relationship: A hospital can have multiple patients, but each patient is associated with a single hospital.

Relationship type: Patient ||--|| Hospital

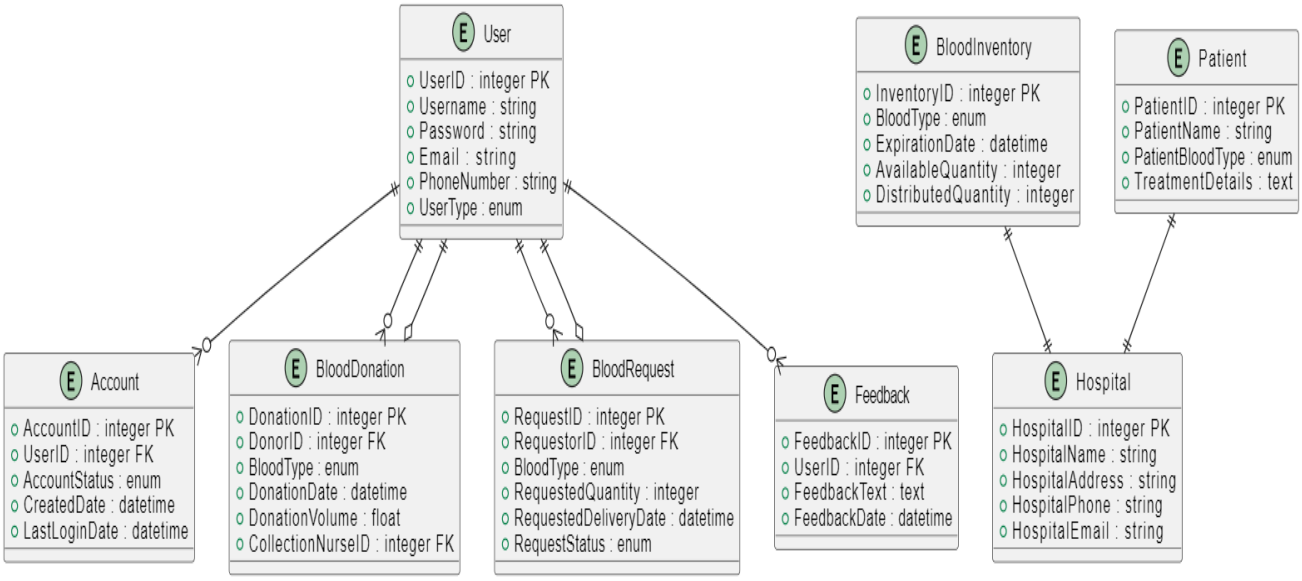

Figure 4. 5 Persistence Diagram

4.3.5 **ACCESS CONTROL AND SECURITY**

The system's user model is defined by an access and security matrix that controls user permissions [8]. When the user starts the application, they are presented with a login screen. The user must enter their username and password, which the system then verifies for validity.

If the login credentials are valid, the system grants the user access to the application based on the privileges associated with their user account. The specific access control list for the system is as follows:

The system has multiple user accounts, each with a defined set of permissions and privileges.

When a user logs in, the system checks their account and the access control list to determine the appropriate level of access to grant.

Depending on the user's assigned privileges, they may be able to view, edit, create, delete, or perform other actions on certain components or data within the application.

The access control and security list specify the exact permissions for each user role or account in the system.

This access control and security mechanism ensures that users can only perform actions that are authorized for their specific account, providing a secure and controlled environment for the application [9].

Table 4. 1 access control

Actor	Request	Post info	Registration	Distribution	Collect blood	Stock management	comment	Notify Donor	Send patient status
Donor	Send request Online for giving blood		Register				View comment		
Admin		adding new information	Register Hospital for being member	Control Distribution of Blood			View comment	Send message	
Nurse					Collect blood from donors		View comment		

Inventory manager				Distribute blood as ordered by Admin		Manage the Stock	View comment		
Hospital	Send request online for receive blood						Give comment		Send patient status

4.3.6 **DETAILED CLASS DIAGRAM**

The Class modeling diagram describes the attributes and operations of a class and also the constraints imposed on the system [10].

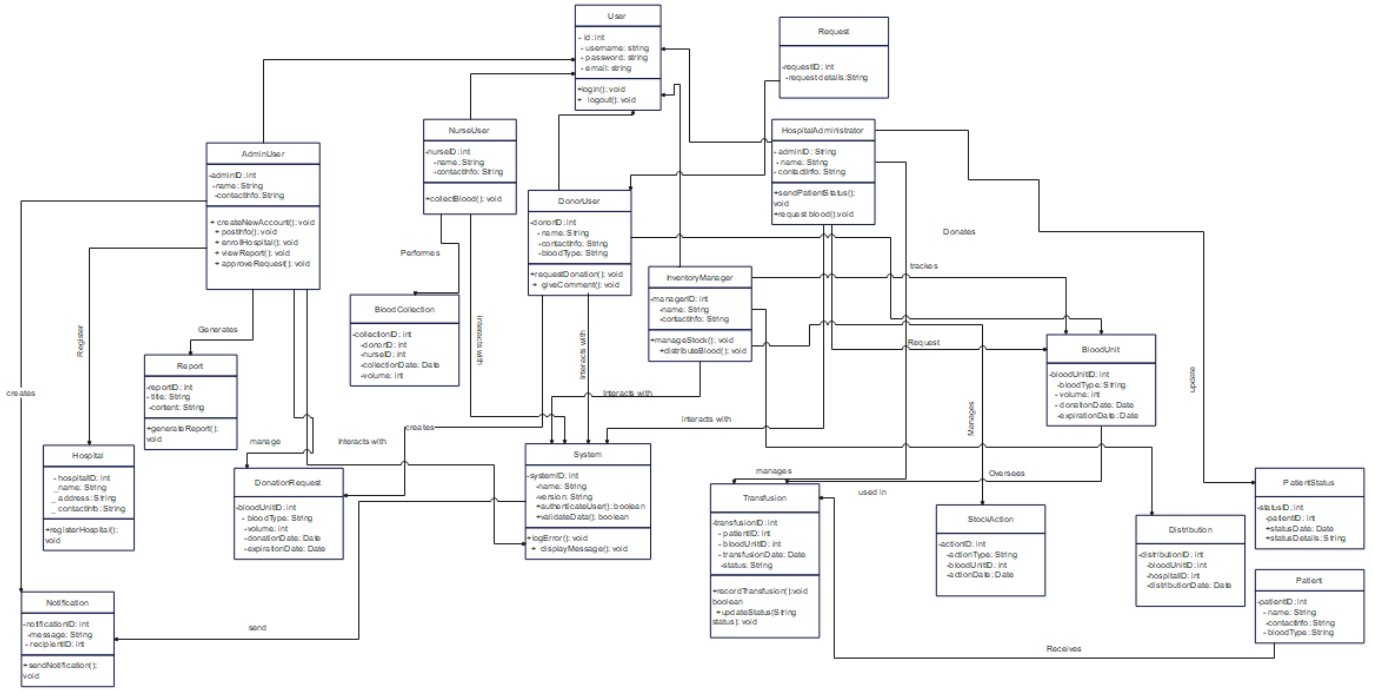

Figure 4. 6 Detailed Class Diagram

4.3.7 **PACKAGE DIAGRAM**

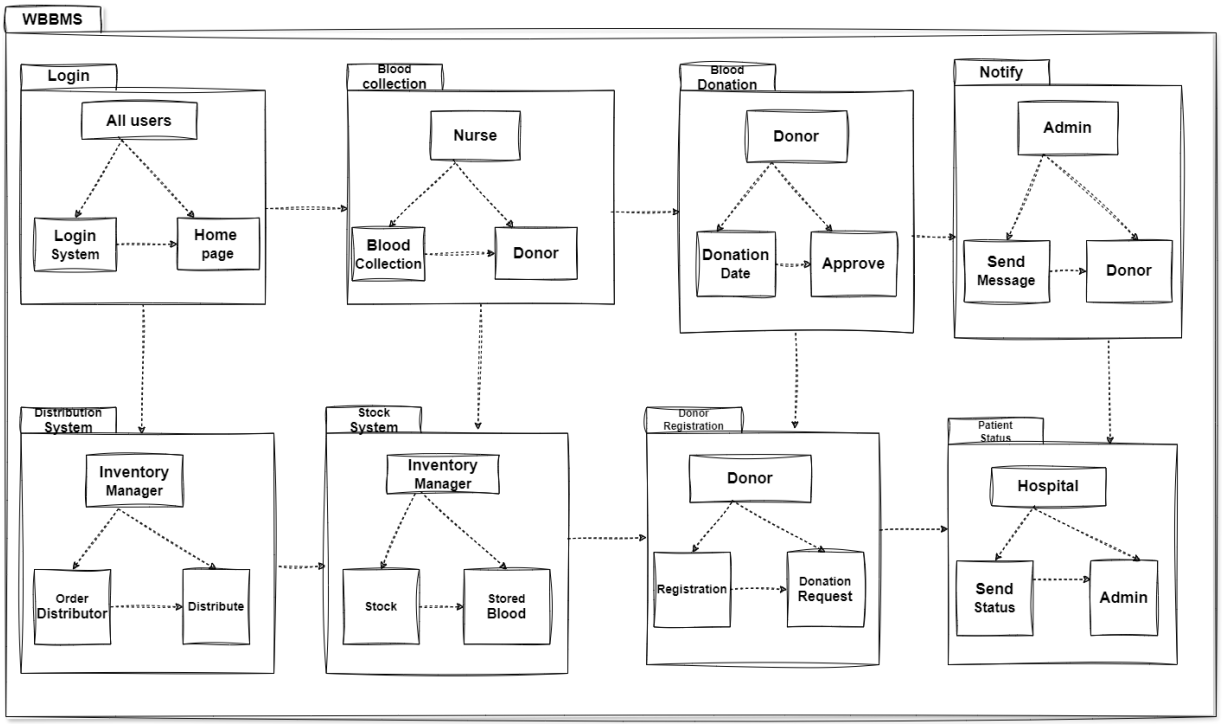

Figure 4. 7 Package Diagram

4.3.8 **DEPLOYMENT DIAGRAM**

The name Deployment itself describes the purpose of the diagram. Deployment diagrams are used for describing the hardware components where software components are deployed [11].

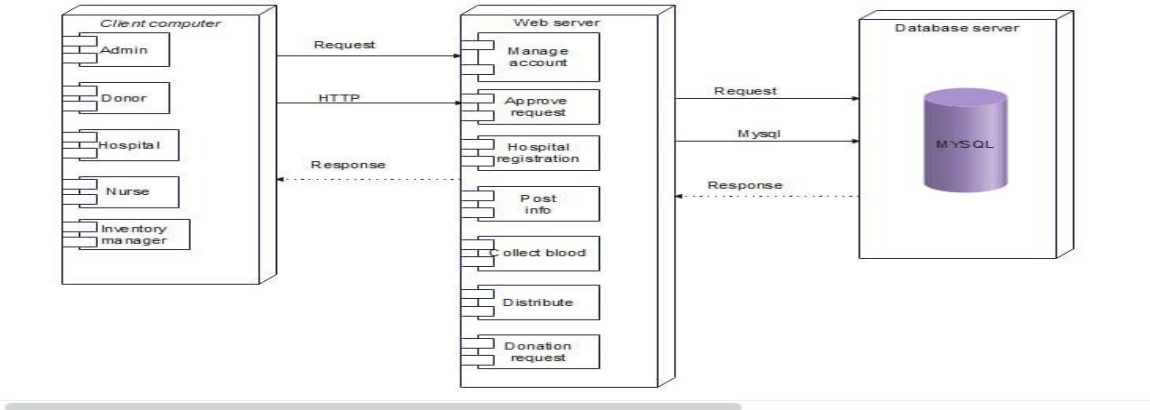

Figure 4. 8 Deployment Diagram

References

[1] "https://redcrosseth.org/blood-bank-service-in-ethiopia/.," [Online].

[2] "https://www.geeksforgeeks.org/unified-modeling-language-uml-activity-diagrams/," [Online].

[3] " R. S. Pressman, Software Engineering: A Practitioner's Approach, 7th ed. New York, NY, USA: McGraw-Hill Education, 2014".

[4] "M. Shaw and D. Garlan, "Software Architecture: Perspectives on an Emerging Discipline," Prentice Hall, Upper Saddle River, NJ, USA, 1996.".

[5] "A. G. Iacob and M. M. Jonkers, "A UML Profile for Component-Based Design," in Proceedings of the 27th Annual International Computer Software and Applications Conference (COMPSAC '03), pp. 57-62, Dallas, TX, USA, Nov. 2003".

[6] "M. Shaw and D. Garlan, "Software Architecture: Perspectives on an Emerging Discipline," Prentice Hall, Upper Saddle River, NJ, USA, 1996.".

[7] "T. Date, "An Introduction to Database Systems," Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 2000.".

[8] "S. Jajodia, P. Samarati, and V.S. Subrahmanian, "A Logical Language for Expressing Authorizations," in ACM Transactions on Database Systems (TODS), vol. 20".

[9] "G. C. S. Gupta and N. Gupta, "Role-Based Access Control: A Comprehensive Review," in International Journal of Computer Applications, vol. 171".

[10] "J. Rumbaugh, I. Jacobson, and G. Booch, "The Unified Modeling Language Reference Manual," Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 2005.".

[11] "H. Mili and A. Mili, "A Generic Framework for the Specification and Verification of Software Architectures," in IEEE Transactions on Software Engineering, vol. 21".