4. DESIGN

4.1 INTRODUCTION

Design is the first step in the development phase for any engineered product or system. The designer's goal is to produce a model or representation of an entity that will later be built. Beginning, once system requirements have been specified and analyzed, system design is the first of the three technical activities – design, code and test that is required to build and verify software. In the technical point of view, design is comprised of four activities – architectural design, data structure design, interface design and procedural design.

4.2 UML DIAGRAMS

Unified Modeling Language (UML) is a standardized general-purpose modeling language in the field of object-oriented software engineering. The Unified Modeling Language includes a set of graphic notation techniques to create visual models of object-oriented software-intensive systems. UML plays an important role in defining different perspectives of a system. These perspectives are:

- Design
- Implementation
- Process
- Deployment

The center is the use case view which connects all these four. A Use case represents the functionality of the system. So the other perspectives are connected with use case.

- **Design:** It consists of classes, interfaces and collaboration. UML provides class diagram, object diagram to support this.
- **Implementation:** It defines the components assembled together to make a complete physical system. UML component diagram is used to support implementation perspective.

- **Process:** It defines the flow of the system. So the same elements as used in design are also used to support this perspective.
- **Deployment**: It represents the physical nodes of the system that forms the hardware. UML deployment diagram is used to support this perspective.

Modeling

It is important to distinguish between the UML model and the set of diagrams of a system. A diagram is a partial graphic representation of a system's model. The model also contains documentation that drives the model elements and diagrams.

UML diagrams represent two different views of a system model:

- **Static view:** It emphasizes the static structure of the system using objects, attributes, operations and relationships. The structural view includes class diagrams and composite structure diagrams.
- **Dynamic view**: It emphasizes the dynamic behavior of the system by showing collaborations among objects and changes to the internal states of objects. This view includes sequence diagrams, activity diagrams and state machine diagrams.

IMPORTANT DIAGRAMS

Structure Diagrams

Structure diagrams emphasize the things that must be present in the system being modeled. Structure diagrams represent the structure, they are used extensively in documenting the software architecture of software systems.

Component Diagram

Component diagrams emphasize the things that must be present in the system being modeled. It describes how a software system is split up into components and shows the dependencies among these components.

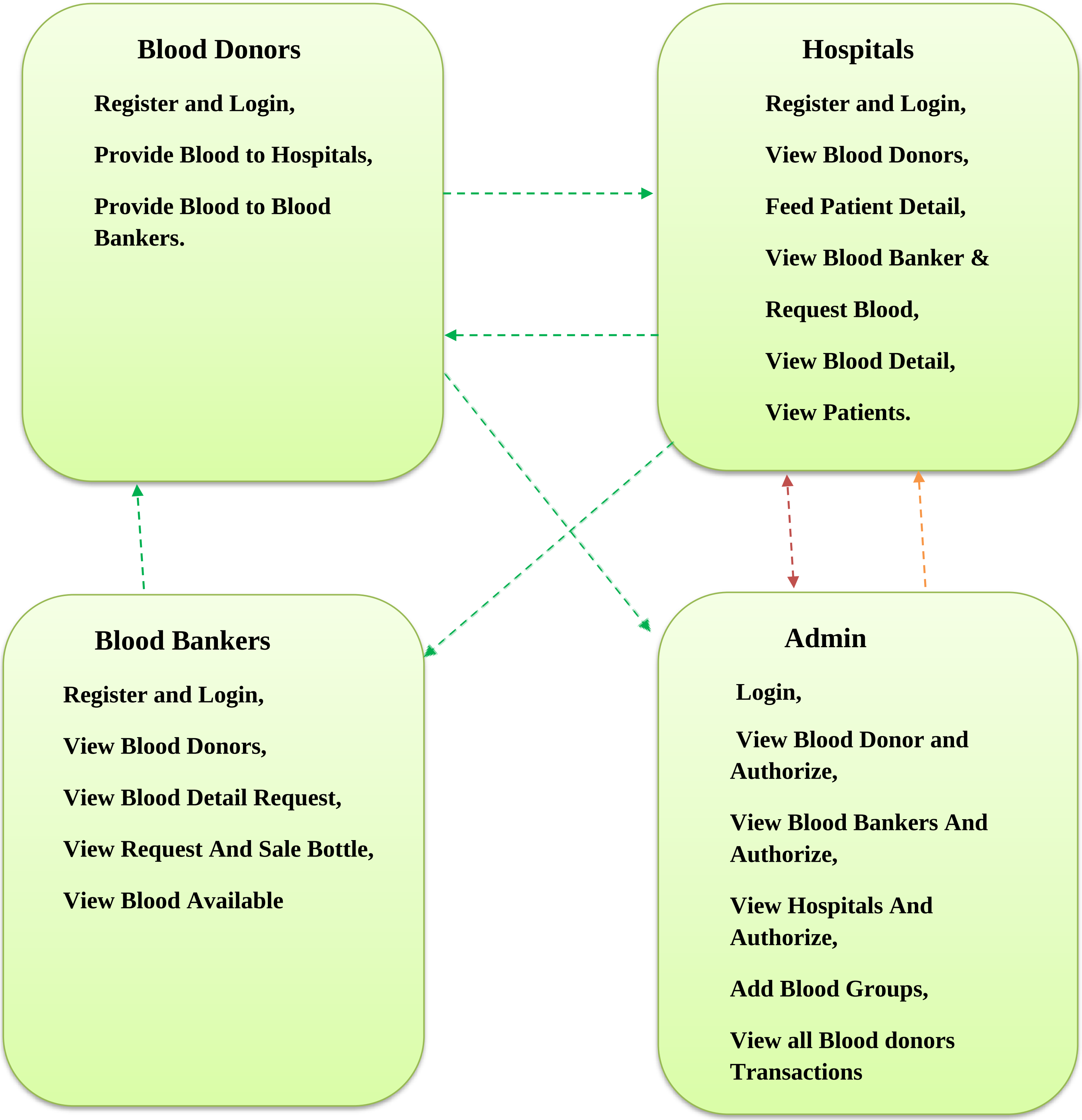


Fig 4.2.1: Component Diagram

Explanation: The above component diagram represent a set of components and their relationships. These components consists of Blood donor, Admin, Blood banker, Hospital.

The Component diagrams represent the implementation view of a system.

Data Flow Diagram of Admin

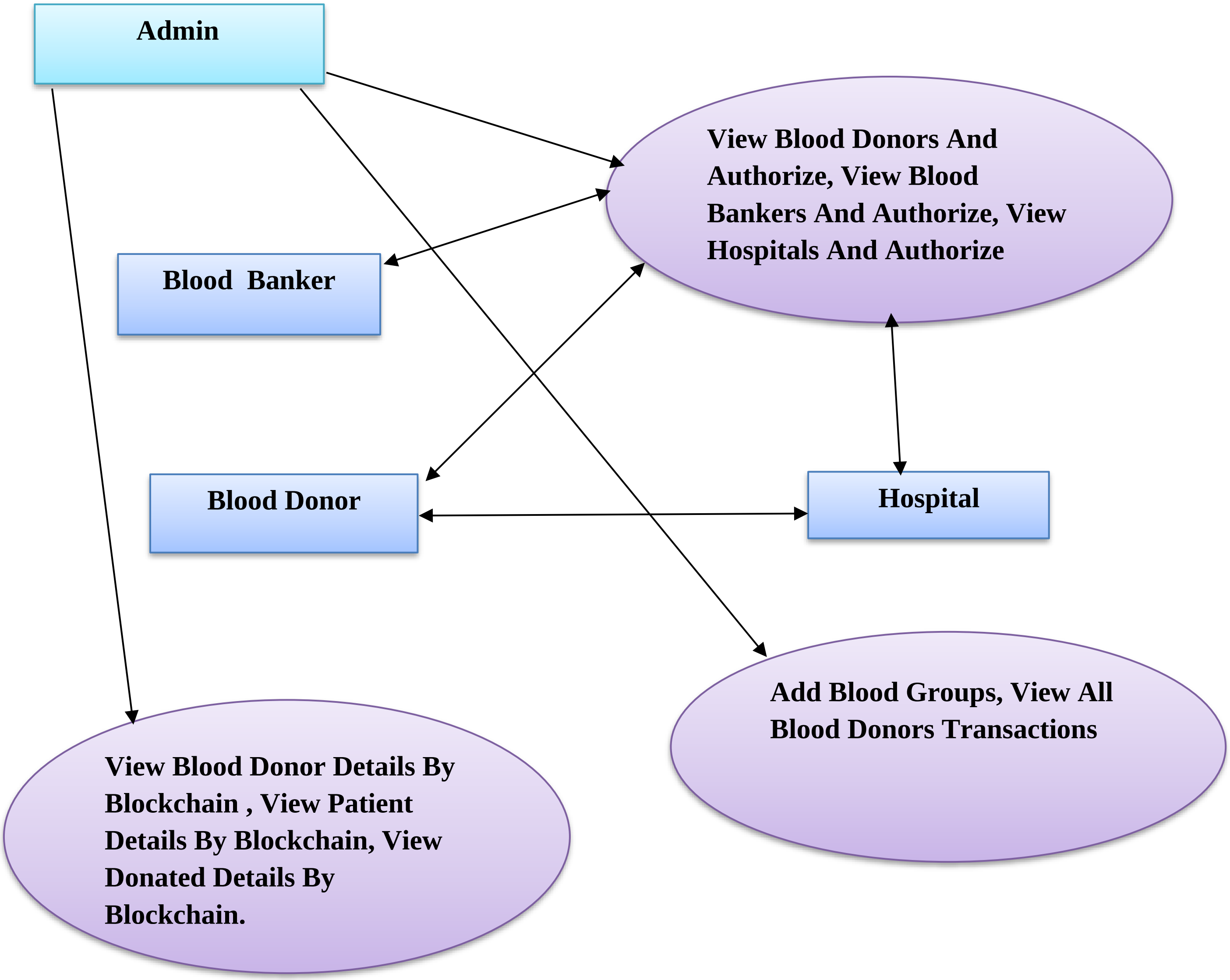

Admin
View Blood Donors And Authorize, View Blood Bankers And Authorize, View Hospitals And Authorize
Blood Banker
Blood Donor
Hospital
Add Blood Groups, View All Blood Donors Transactions
View Blood Donor Details By Blockchain , View Patient Details By Blockchain, View Donated Details By Blockchain.

Fig 4.2.2 Data Flow diagram of Admin Module

Explanation: This data flow diagram explains about how the admin module interact with remaining modules along with admin operations.

Level 0: This Data flow diagram explains about the Complete process of Blood donation process and also explains Admin, Blood banker, Blood donor and Hospital module.

Level 1: This Data flow diagram explains about the Blood donor process and explains the donor interactions.

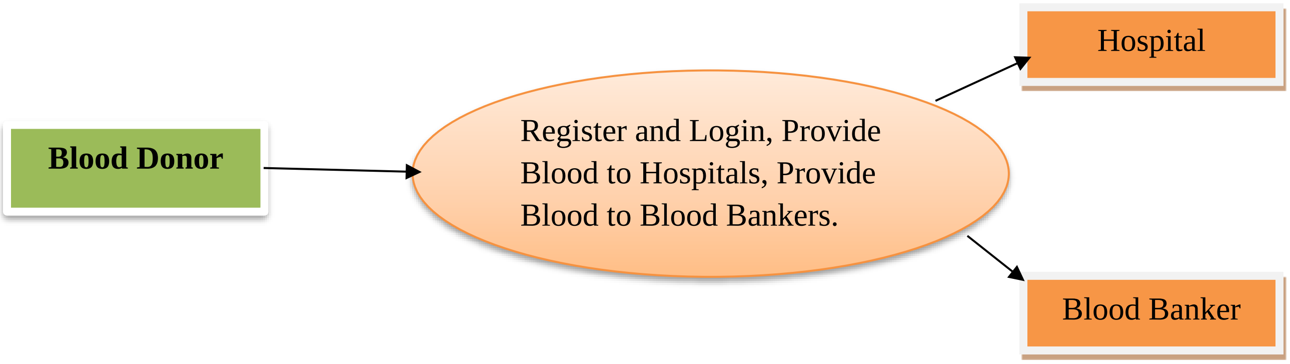

Hospital
Blood Donor
Register and Login, Provide Blood to Hospitals, Provide Blood to Blood Bankers.
Blood Banker

Fig 4.2.3 Blood Donor Module Operations

Explanation: This figure explains about the blood donor modules operations and explains how the blood donor module do operations with Hospital and Blood banker.

Structural things

It is fine the static part of the model. They represent physical and conceptual elements. Following are the brief descriptions of the structural things.

Behavioral things

A behavioral thing consists of the dynamic parts of UML models. Following are the behavioral things.

Interaction

It defined as a behavior that consists of a group of messages exchanged among elements to accomplish a specific task.

Dependency

Dependency is a relationship between two things in which change in one element also affects the other one.

----------->

DFD/UML DIAGRAMS

Context Level DFD

It is a simple graphical formalism that can be used to represent a system in terms of the input data to the system, various processing carried out on these data and the output data is generated by the system.

System Designs in this Projects:

The System designs are plays key roles for perform the various operations in various modules in the system. System designs follows the registrations process and various operations in each modules in this system.

It provides a structured approach to designing a blood donation system using blockchain technology. Each step contributes to building a secure, efficient and transparent platform for managing blood donations system. System design must have the following methods:

- **User Registration and Verification**
- **Smart Contracts**
- **Monitoring and Maintenance**

System designs of this project is mainly classified into four various designs. They are:

- **System Admin Design**
- **System Hospital Design**
- **System Blood Bankers Design**
- **System Blood Donors Design**

System Admin Design of the Project

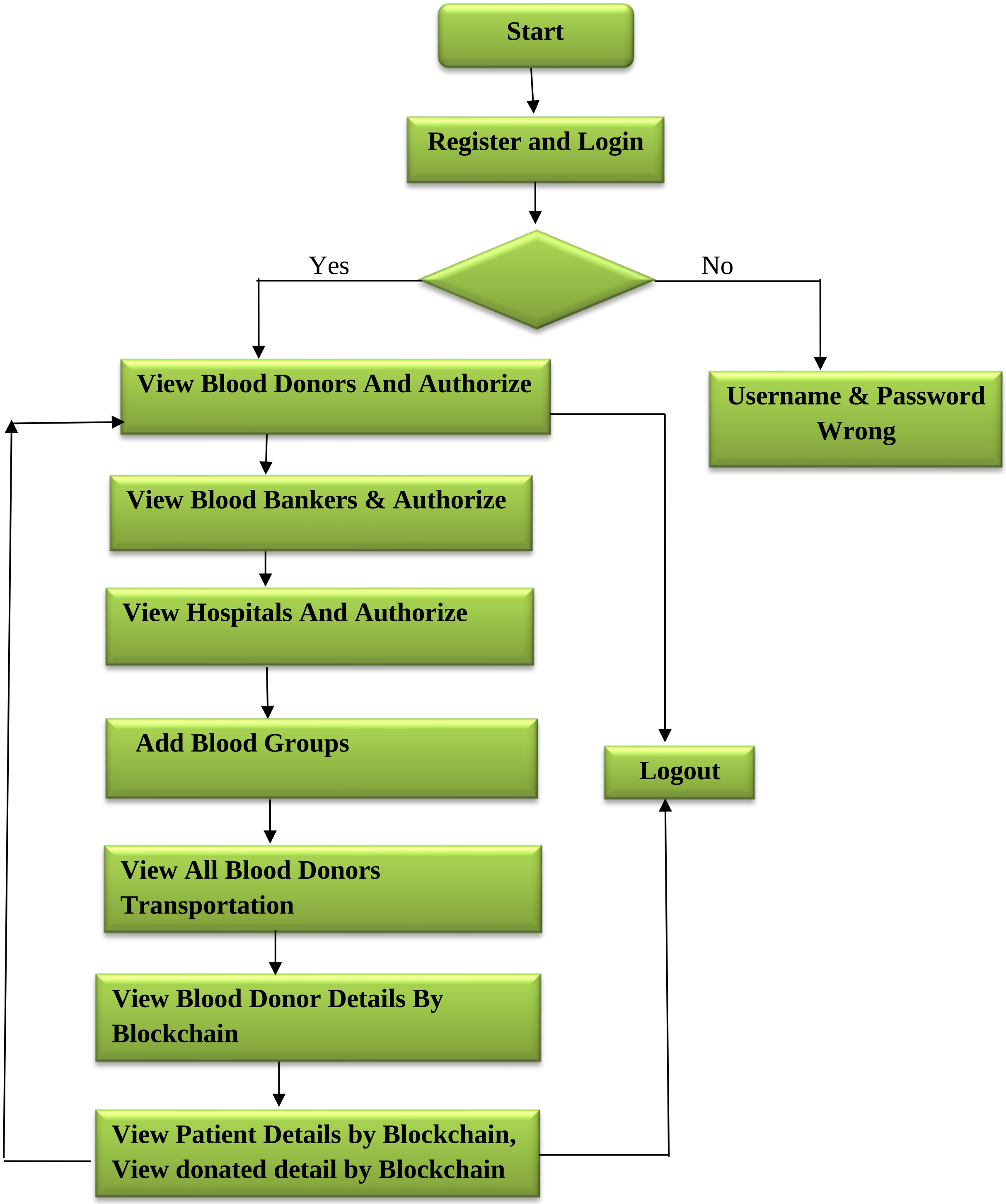

Start
Register and Login
Yes
No
View Blood Donors And Authorize
Username & Password Wrong
View Blood Bankers & Authorize
View Hospitals And Authorize
Add Blood Groups
Logout
View All Blood Donors Transportation
View Blood Donor Details By Blockchain
View Patient Details by Blockchain, View donated detail by Blockchain

Fig :4.2.4 System Admin Design

Explanation: The above figure explains about Admin system design and what type of operations admin can do in the system.

System Hospital Design of the Project

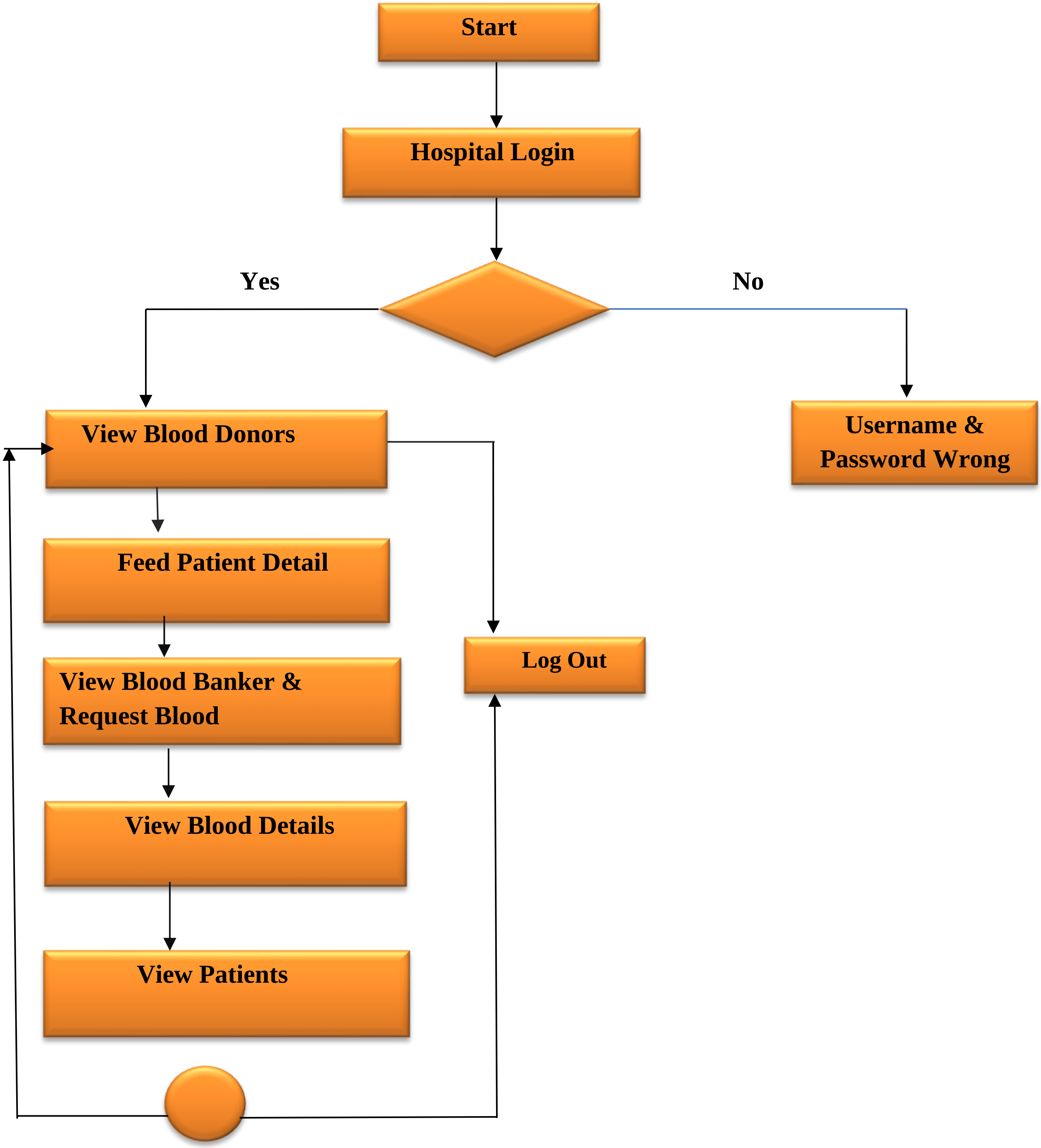


Fig :4.2.5 System Hospital Design

Explanation: The above figure explains about the hospital registration and login process and remaining operations in this system.

System Blood Bankers Design of the Project

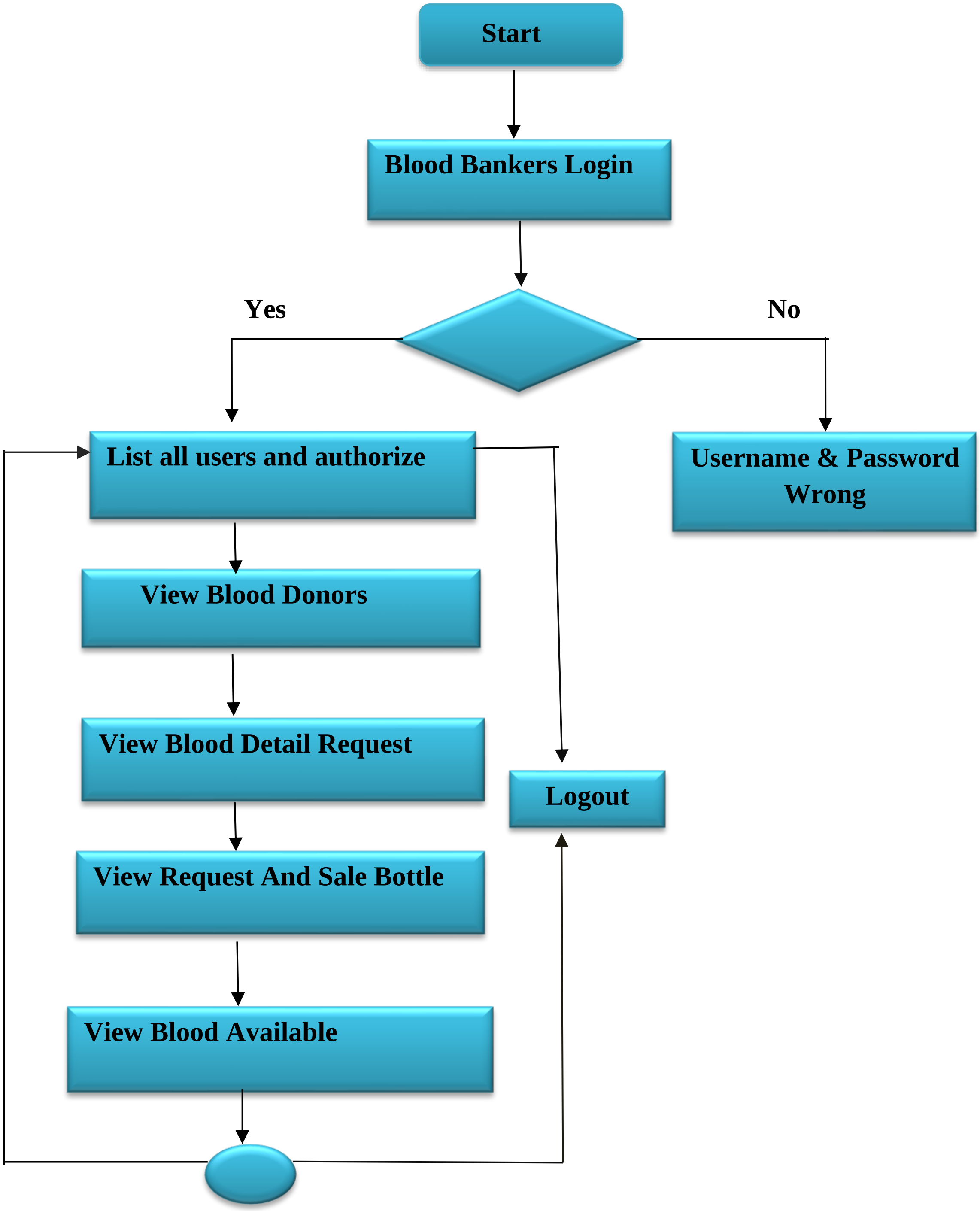

Start
Blood Bankers Login
Yes
No
List all users and authorize
Username & Password Wrong
View Blood Donors
View Blood Detail Request
Logout
View Request And Sale Bottle
View Blood Available

Fig :4.2.6 System Blood Bankers Design

Explanation: The above figure explains about the blood banker registration process and its operations.

System Blood Donor Design of the Project

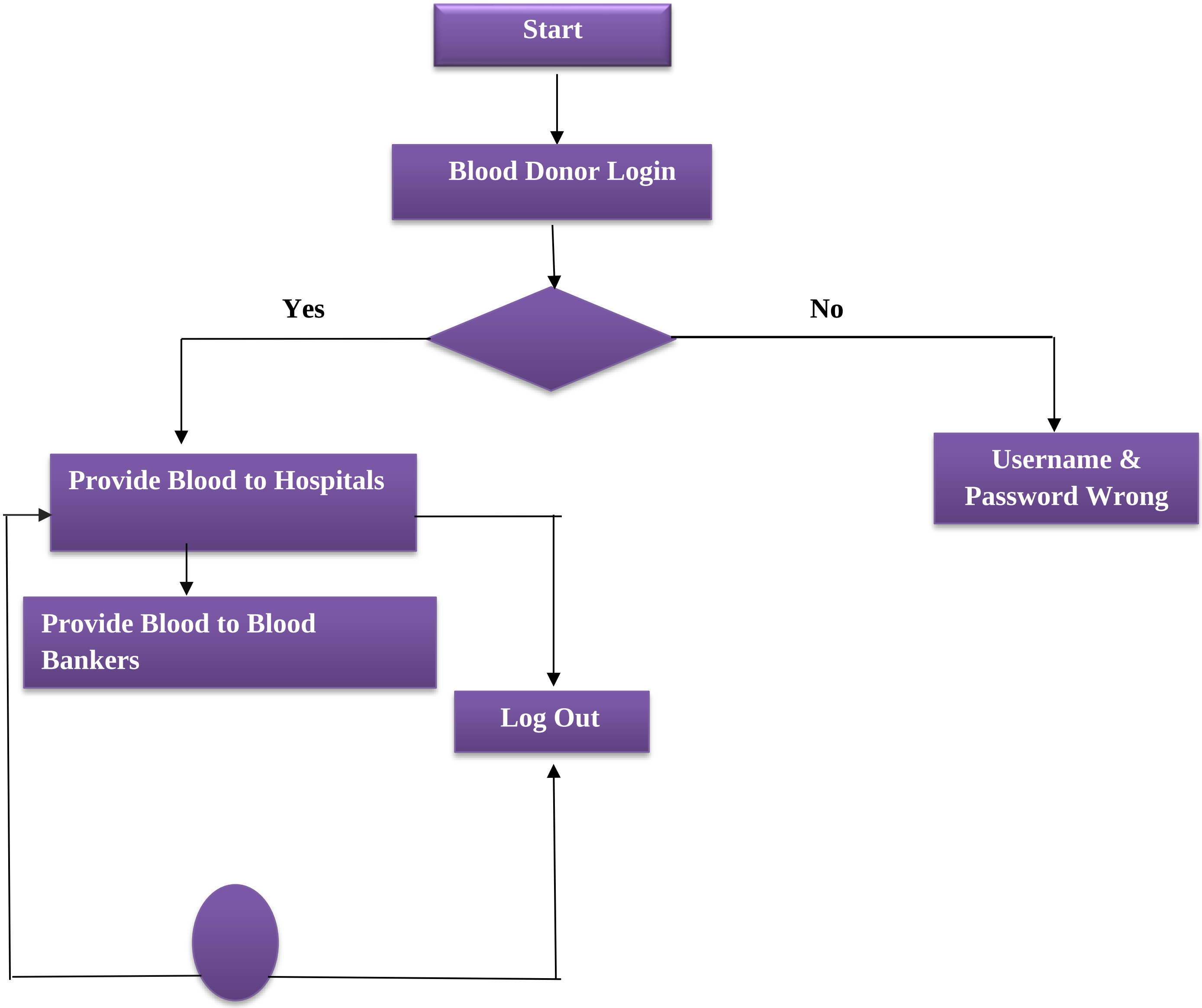
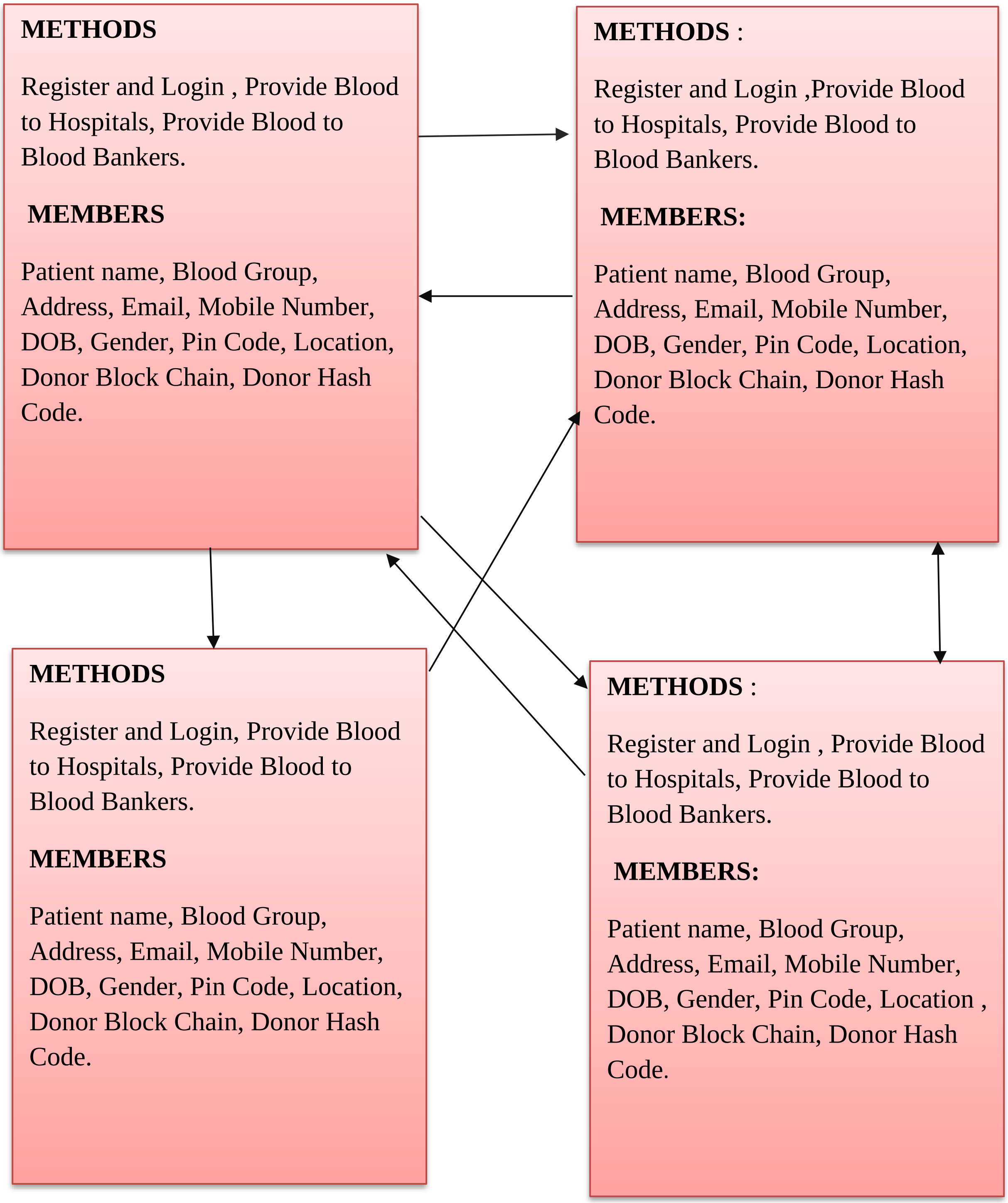

Start
Blood Donor Login
Yes
No
Provide Blood to Hospitals
Username & Password Wrong
Provide Blood to Blood Bankers
Log Out

Fig :4.2.7 System Blood Donor Design

Explanation: The above figure explains about the Blood donor registration and login process and also explains about the blood donor operations.

Class diagram: It describes the structure of a system by showing the system's classes, their attributes and the relationships among the classes.



METHODS

Register and Login , Provide Blood to Hospitals, Provide Blood to Blood Bankers.

MEMBERS

Patient name, Blood Group, Address, Email, Mobile Number, DOB, Gender, Pin Code, Location, Donor Block Chain, Donor Hash Code.

METHODS :

Register and Login ,Provide Blood to Hospitals, Provide Blood to Blood Bankers.

MEMBERS:

Patient name, Blood Group, Address, Email, Mobile Number, DOB, Gender, Pin Code, Location, Donor Block Chain, Donor Hash Code.

METHODS

Register and Login, Provide Blood to Hospitals, Provide Blood to Blood Bankers.

MEMBERS

Patient name, Blood Group, Address, Email, Mobile Number, DOB, Gender, Pin Code, Location, Donor Block Chain, Donor Hash Code.

METHODS :

Register and Login , Provide Blood to Hospitals, Provide Blood to Blood Bankers.

MEMBERS:

Patient name, Blood Group, Address, Email, Mobile Number, DOB, Gender, Pin Code, Location , Donor Block Chain, Donor Hash Code.

Fig 4.2.8: Class Diagram

Explanation: The above class diagram consists of Admin methods and Members, Blood donor methods and Members, Blood banker methods and Members, Hospital methods and Members.

Sequence Diagram: Sequence diagram describes the systems operations in different modules.

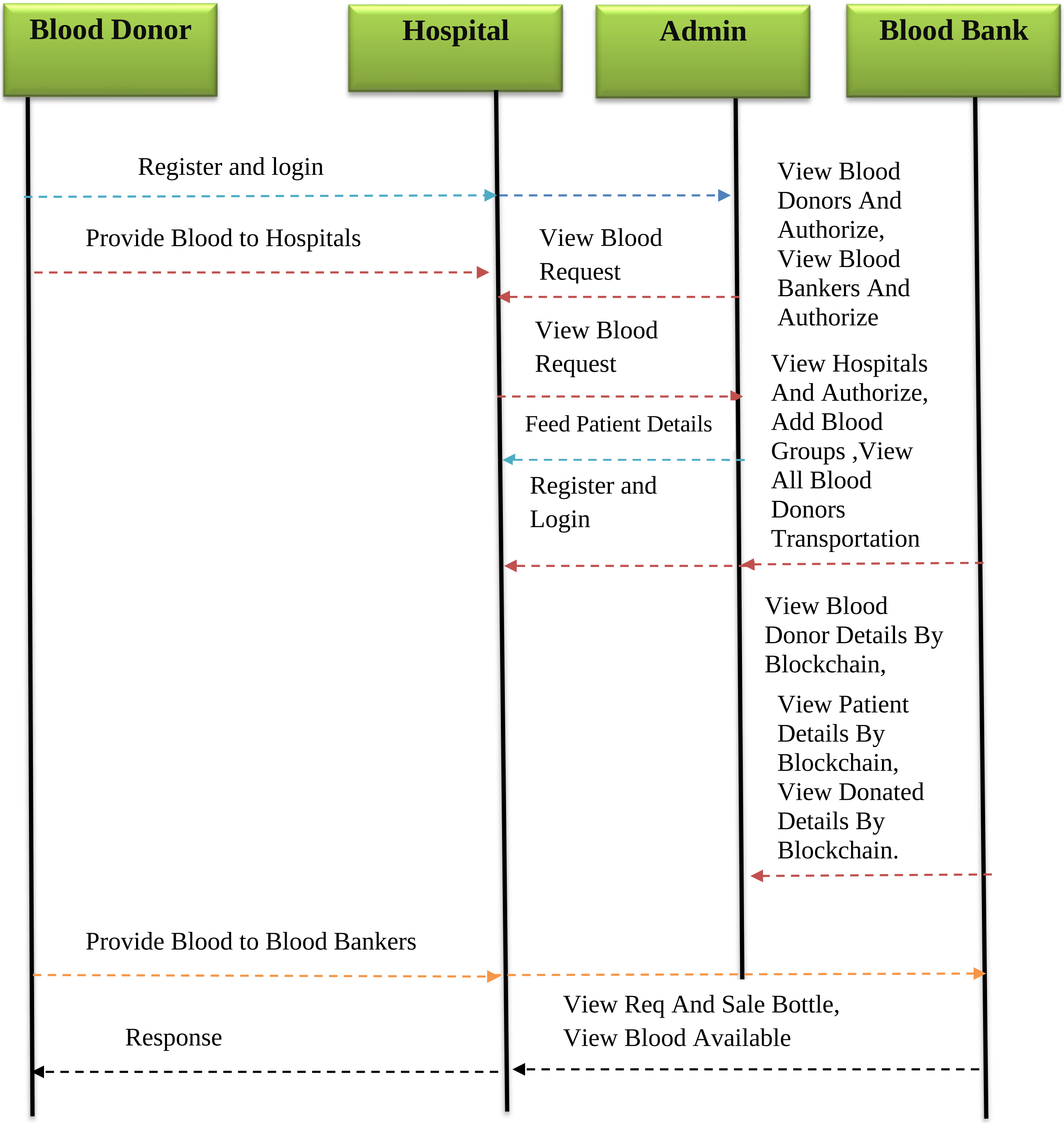

Blood Donor
Hospital
Admin
Blood Bank
Register and login
View Blood Donors And Authorize, View Blood Bankers And Authorize
Provide Blood to Hospitals
View Blood Request
View Blood Request
View Hospitals And Authorize, Add Blood Groups ,View All Blood Donors Transportation
Feed Patient Details
Register and Login
View Blood Donor Details By Blockchain,
View Patient Details By Blockchain, View Donated Details By Blockchain.
Provide Blood to Blood Bankers
View Req And Sale Bottle, View Blood Available
Response

Fig 4.2.9 : Sequence Diagram

Explanation: The above Sequence diagram explains how Admin communicate with each other in terms of a sequence of request and response way. It also consists of remaining modules methods and Members and also shows the interaction ways of admin with remaining Modules and Operations.

Use Case Diagram

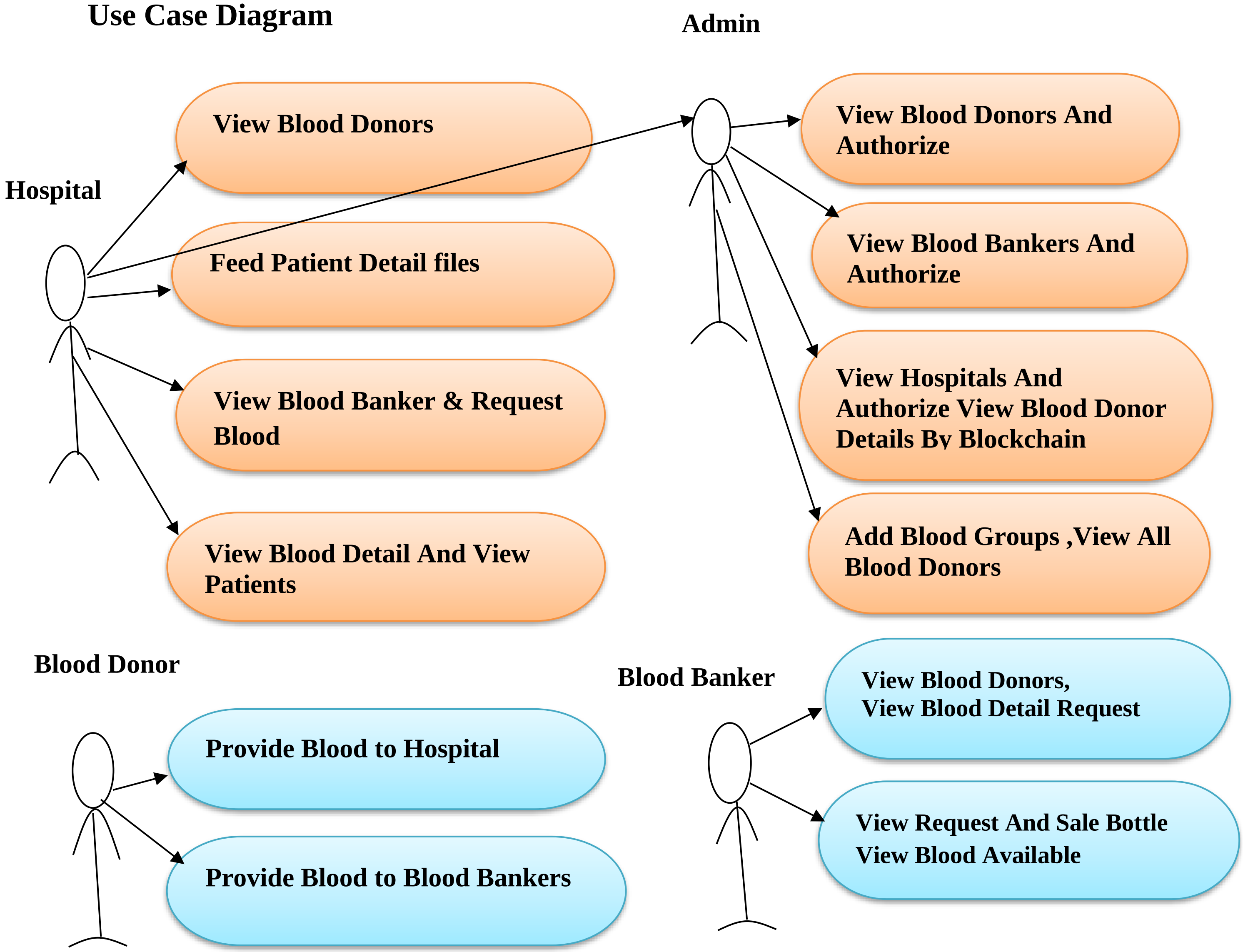

Admin
View Blood Donors
Hospital
Feed Patient Detail files
View Blood Banker & Request Blood
View Blood Detail And View Patients
View Blood Donors And Authorize
View Blood Bankers And Authorize
View Hospitals And Authorize View Blood Donor Details By Blockchain
Add Blood Groups ,View All Blood Donors
Blood Donor
Provide Blood to Hospital
Provide Blood to Blood Bankers
Blood Banker
View Blood Donors,
View Blood Detail Request
View Request And Sale Bottle
View Blood Available

Fig: 4.2.10 Use Case Diagram

Explanation: The above use case diagram consists of modules methods and Members in Hospital ,Admin, Blood banker and Blood donor and also shows the interaction ways of Admin with remaining Modules.

4.3 Module Design and Organization

Design is that first step in the development phase for any engineering product or system. The designer goals are to produce a model representation of any entity that will later be built. Beginning, once system requirement have been specified and analyzed, system design is the first of the three technical activities-design, code and test that is build and verify software. Design is the place where quality is fostered in software development. Design provides us with representations of software that can access for quality. During design, progressive refinement of data structure, program structure and procedural details are developed, reviewed and document. System design can be viewed from either technical or project management perspective.

4.4 Summary

The designer's goal is to produce a model or representations of an entity have will later be built. Beginning, once system requirement have been specified and analyzed, system design is the first of the three technical activities-designs, code and test is required to build and verify software.