5.1 Construction Phase

The **Construction Phase** is the third phase in the Rational Unified Process (RUP) software development methodology. It involves the actual development and implementation of the software system, based on the detailed architecture defined in the previous Elaboration Phase. During the Construction Phase, the development team focuses on writing code, testing and integrating components, and building the system incrementally. The development team will also continuously refine and improve the design as the system is being built. The goal of the Construction Phase is to deliver a working version of the system, ready for testing and validation by stakeholders. To achieve this, the development team will use tools and techniques such as version control, build automation, continuous integration and testing, and project management to ensure that the system is built to a high standard, on time and within budget. By the end of the Construction Phase, the development team should have a functioning system that can be demonstrated to stakeholders and stakeholders can provide feedback for further refinement.

During the Construction Phase, the development team will also carry out a range of activities to ensure the quality and reliability of the system, such as code reviews, testing, and debugging. The development team will also use design patterns, best practices, and established software engineering methodologies to ensure that the system is built to be scalable, maintainable, and secure. In addition, the development team will also use agile methodologies, such as Scrum or Kanban, to manage and track progress during the Construction Phase. This will help to ensure that the system is being built according to schedule, and that any issues or problems are identified and addressed in a timely manner. The Construction Phase is also an opportunity for the development team to refine and improve the design of the system, based on feedback from stakeholders and real-world usage. This can lead to a better understanding of the system requirements, and can result in a more robust, flexible and user-friendly system. Overall, the Construction Phase is a critical stage in the software development process, as it involves the actual creation of the software system. Successful completion of this phase sets the foundation for the next phase of the RUP process, the Transition Phase, where the system is deployed and transitioned into production.

The following are some **steps** that we should do in the Construction phase of the RUP methodology for our e-shop application:

1. **Use Case Diagram (v.3):** This diagram is used to represent the functional **requirements** of the system and the **interactions** between the system and its actors. It helps to identify the main functions of the system and the relationships between the actors and the system. In this **updated version**, we noticed that it was important to add **role** and **admin** classes and objects.

2. **Class Diagram (v.3):** This diagram is used to represent the **structure** of the system, including classes, their attributes, and their relationships. It provides a clear understanding of the objects and the relationships between them in the system. In this **updated version**, we noticed that it was important to add **role** and **admin** classes and objects.

3. **Object Diagram (v.2):** This diagram is used to represent the **instances** of classes and the relationships between them at a specific point in time. It helps to understand the behavior of the system at runtime.

4. **Collaboration or Communication Diagram (v.2):** This diagram is used to represent the flow of **messages** between objects or components in a system. It helps to understand how the objects interact and communicate with each other.

5. **Sequence Diagram (v.2):** This diagram is used to represent the **interaction between objects or components over time**, including the order of messages and the conditions under which they are sent. It helps to understand the flow of events in the system and the dependencies between the objects.

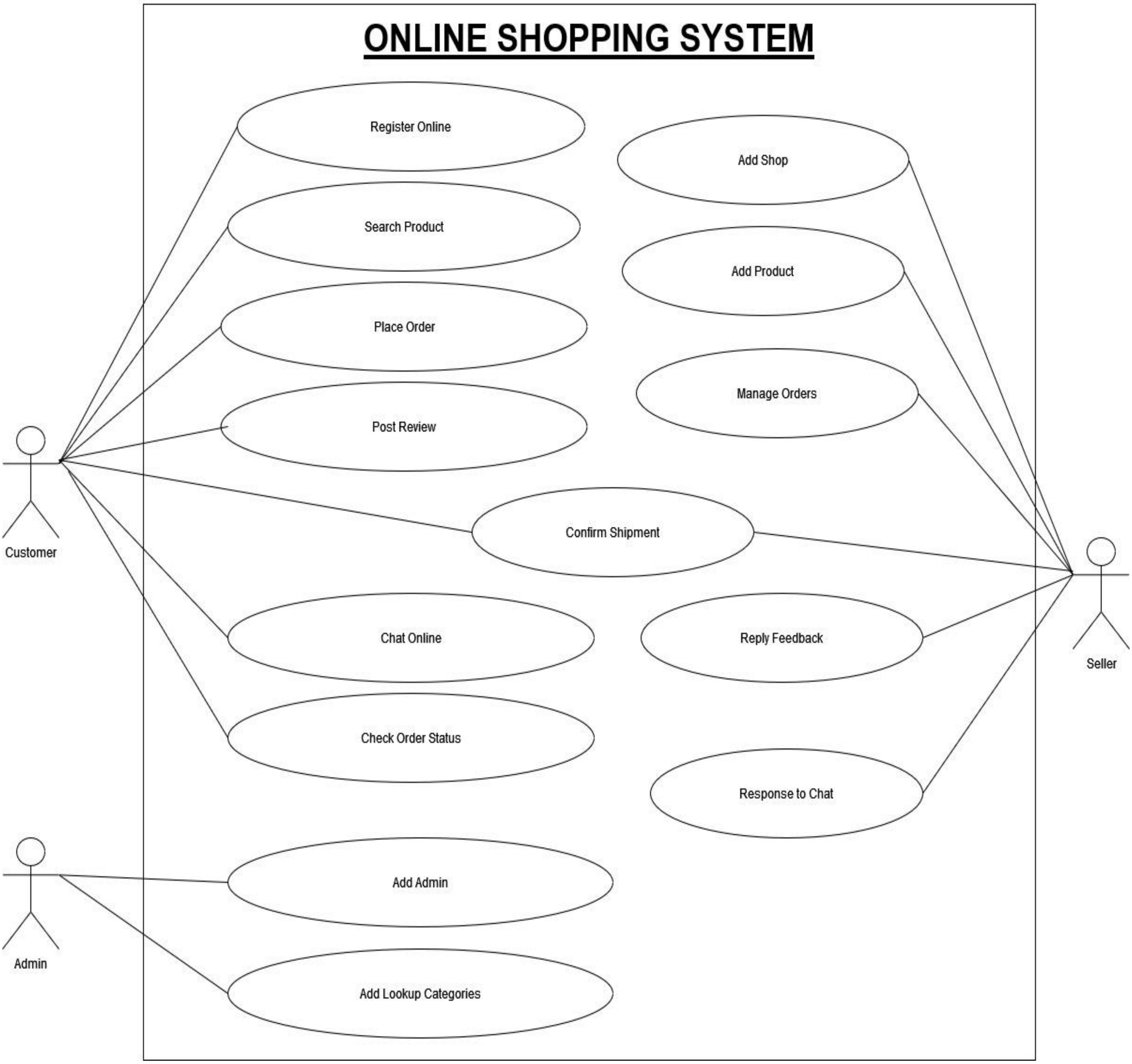
6. **Activity Diagram (v.2):** This diagram is used to represent the **flow of activities** within a system. It helps to understand the logic of a process and the relationships between the steps involved.

7. **State Chart Diagram (v.2):** This diagram is used to represent the **states and transitions of objects or components in a system**. It helps to understand the behavior of the system over time, including the possible states and events that trigger transitions between them.

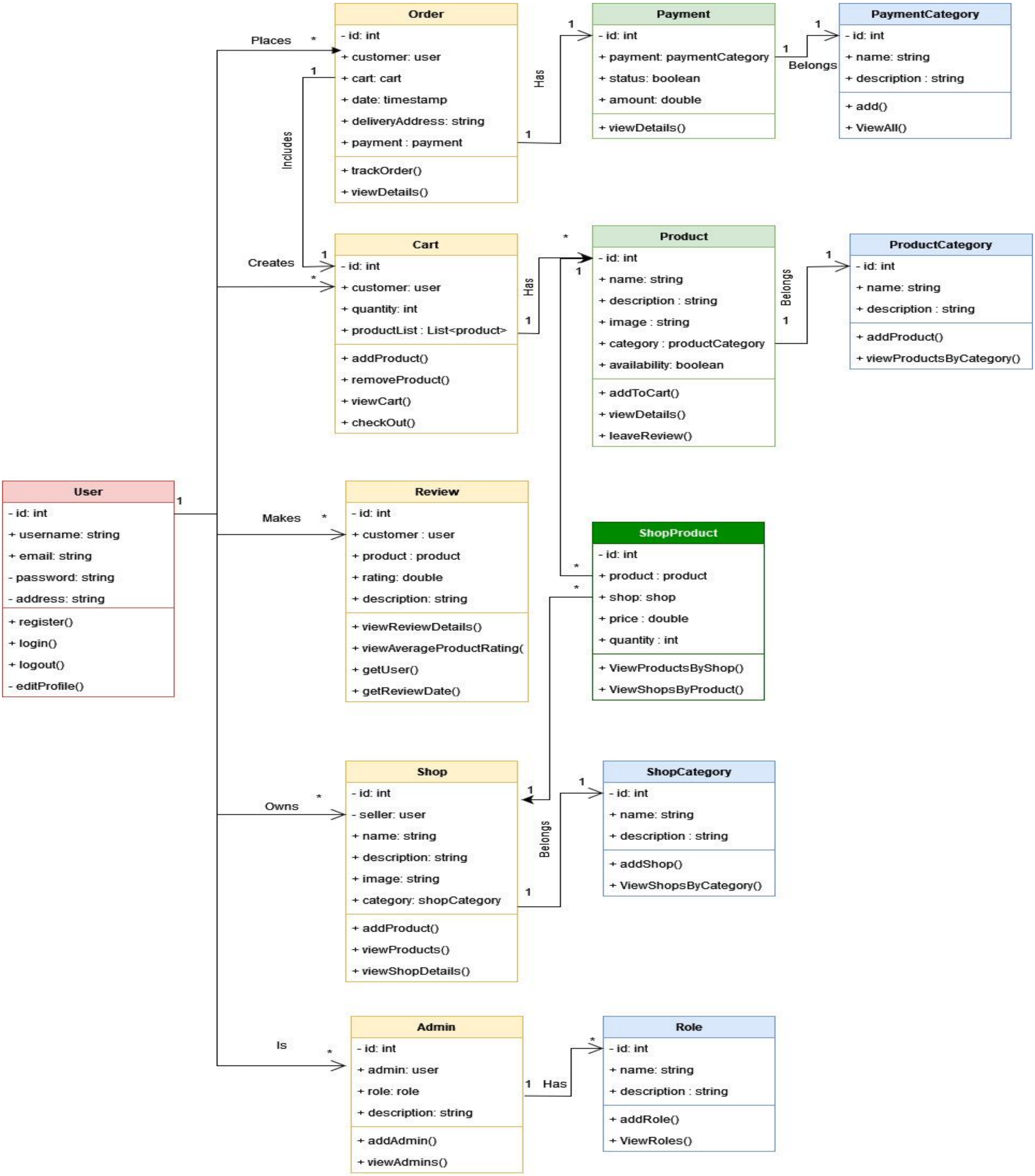
8. **Component Diagram (v.2):** This diagram is used to represent the components and the **relationships** between them in a system. It helps to understand the structure of the system and the dependencies between the components.

9. **Deployment Diagram (v.2):** This diagram is used to represent the **physical deployment** of the system, including the hardware and software components and their relationships. It helps to understand the environment in which the system will run and the dependencies between the components.

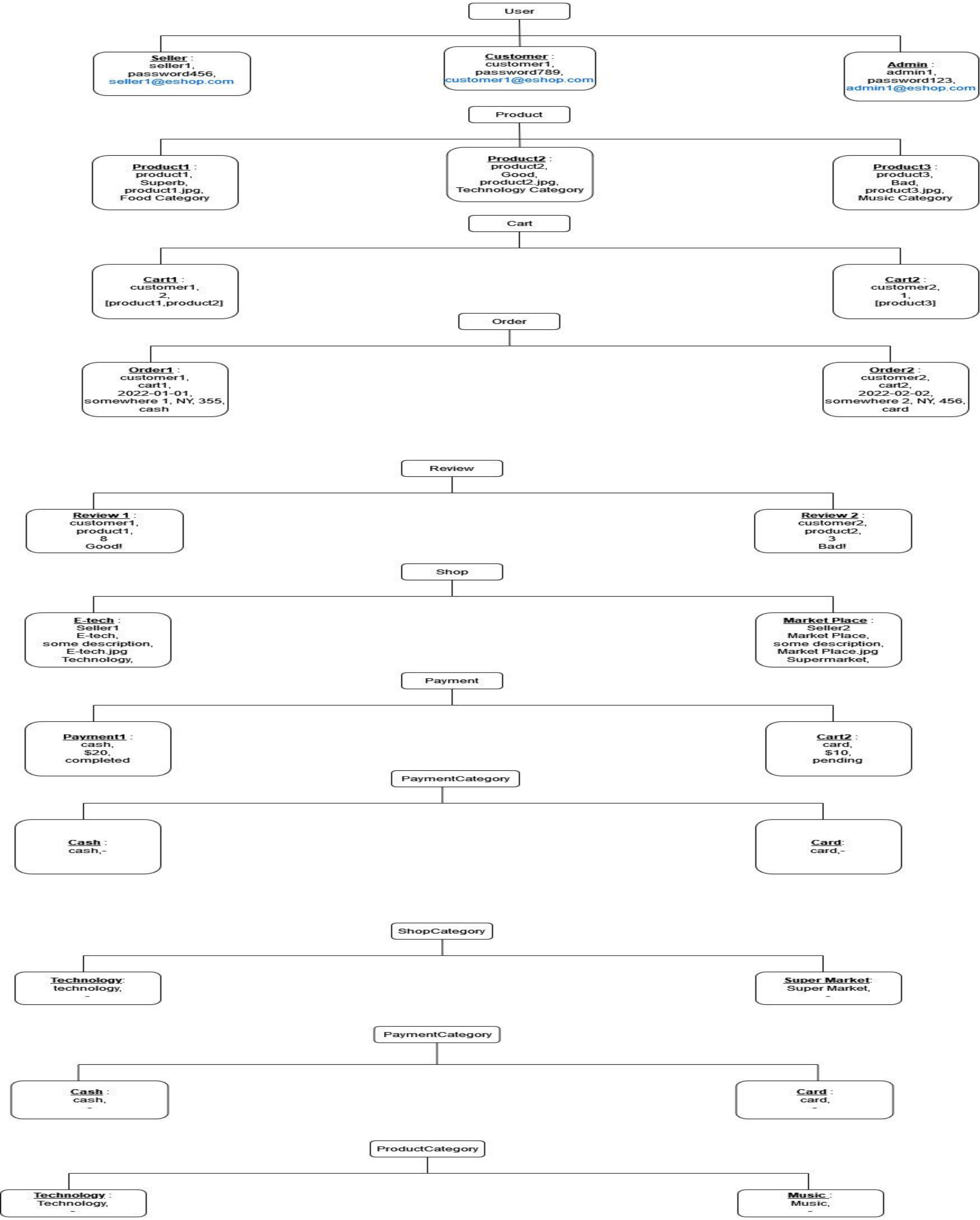
5.1.1 Use Case Diagram (3rd Version)

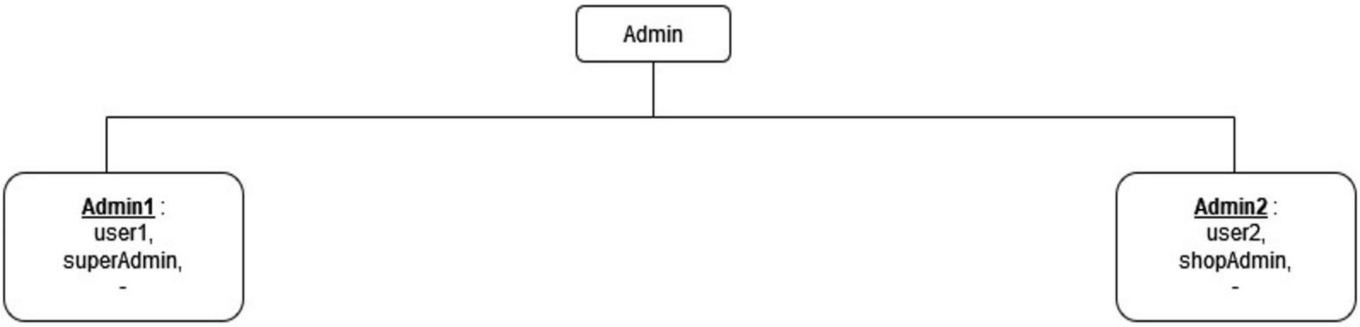


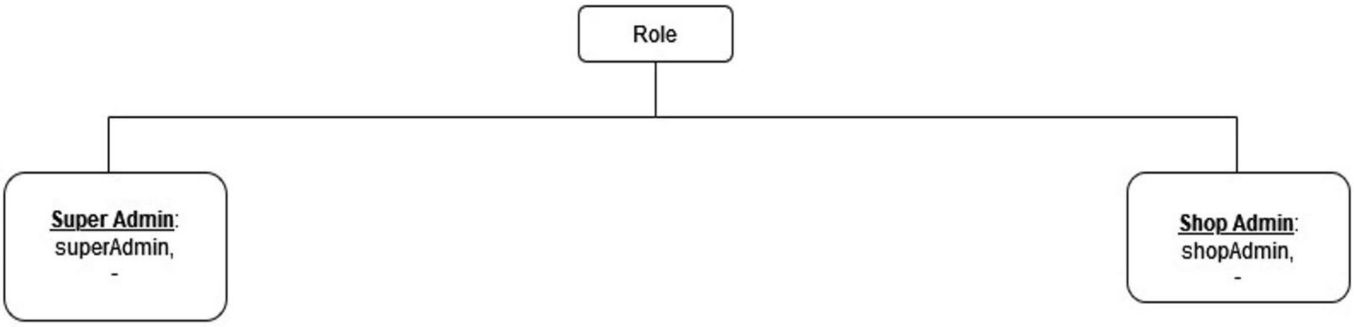
5.1.2 Class Diagram (3rd Version)



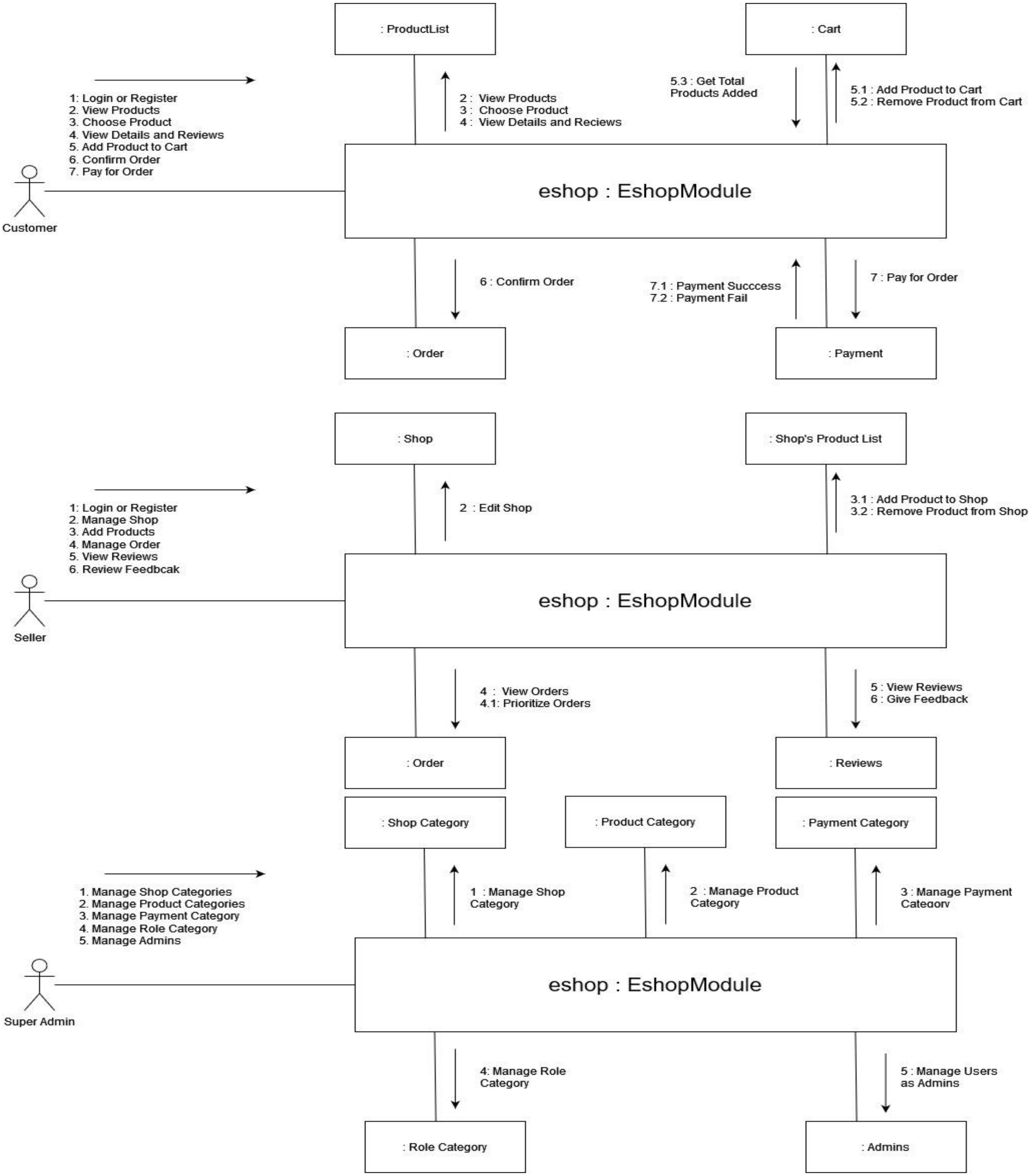
5.1.3 Object Diagram (2nd Version)


User
Seller : seller1, password456, seller1@eshop.com
Customer : customer1, password789, customer1@eshop.com
Admin : admin1, password123, admin1@eshop.com
Product
Product1 : product1, Superb, product1.jpg, Food Category
Product2 : product2, Good, product2.jpg, Technology Category
Product3 : product3, Bad, product3.jpg, Music Category
Cart
Cart1 : customer1, 2, [product1,product2]
Cart2 : customer2, 1, [product3]
Order
Order1 : customer1, cart1, 2022-01-01, somewhere 1, NY, 355, cash
Order2 : customer2, cart2, 2022-02-02, somewhere 2, NY, 456, card
Review
Review 1 : customer1, product1, 8 Good!
Review 2 : customer2, product2, 3 Bad!
Shop
E-tech : Seller1 E-tech, some description, E-tech.jpg Technology,
Market Place : Seller2 Market Place, some description, Market Place.jpg Supermarket,
Payment
Payment1 : cash, $20, completed
Cart2 : card, $10, pending
PaymentCategory
Cash : cash,-
Card: card,-
ShopCategory
Technology: technology, -
Super Market: Super Market, -
PaymentCategory
Cash : cash, -
Card : card, -
ProductCategory
Technology : Technology, -
Music : Music, -

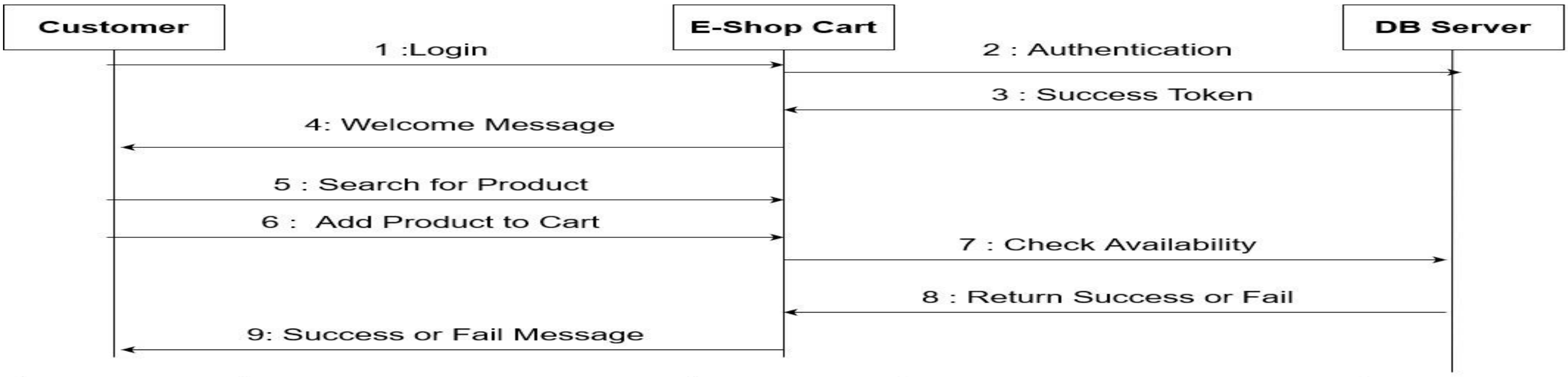

Admin
Admin1 :
user1,
superAdmin,
-
Admin2 :
user2,
shopAdmin,
-

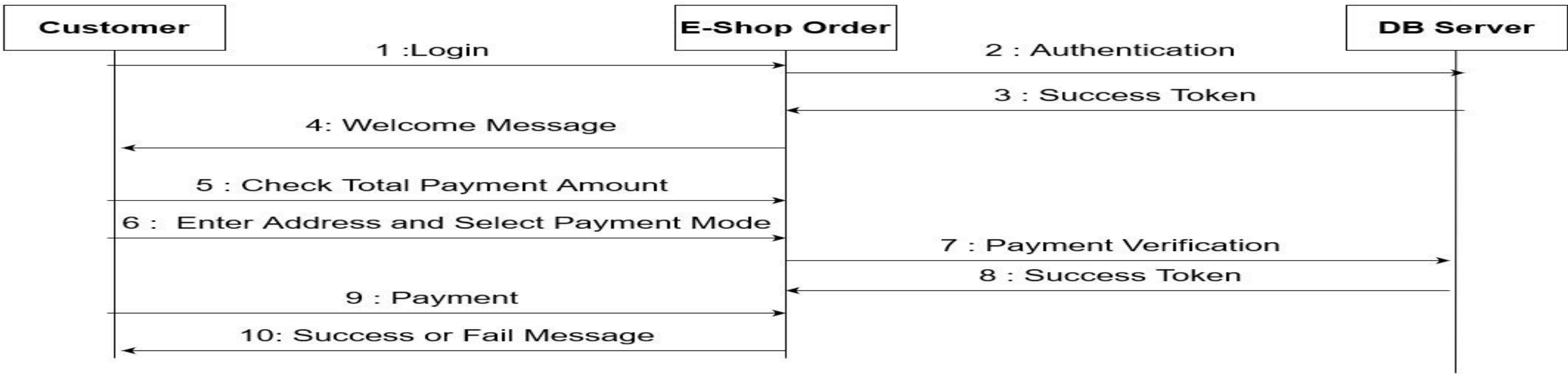

Role
Super Admin:
superAdmin,
-
Shop Admin:
shopAdmin,
-

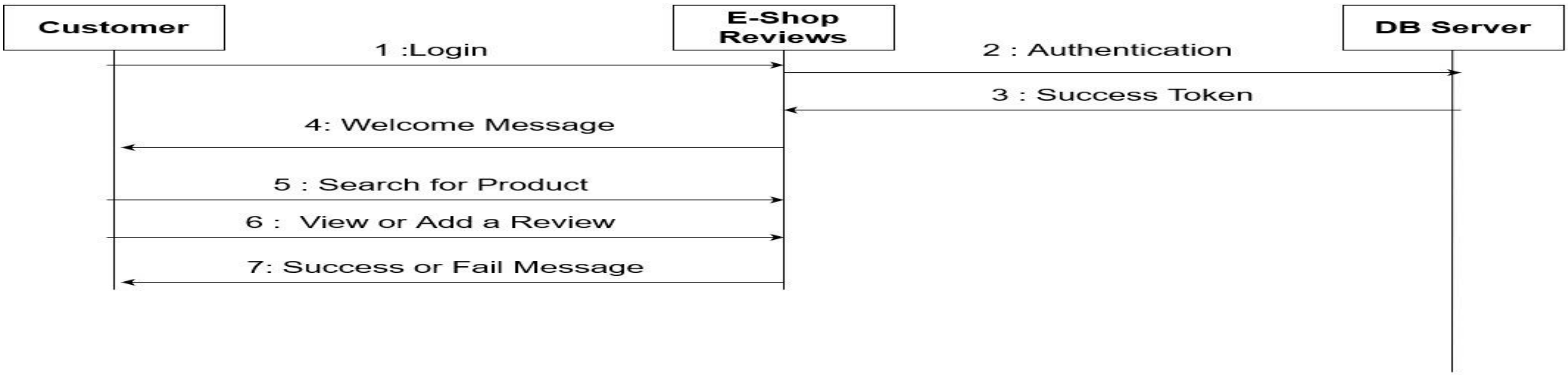
5.1.4 Collaboration or Communication Diagram (2nd Version)

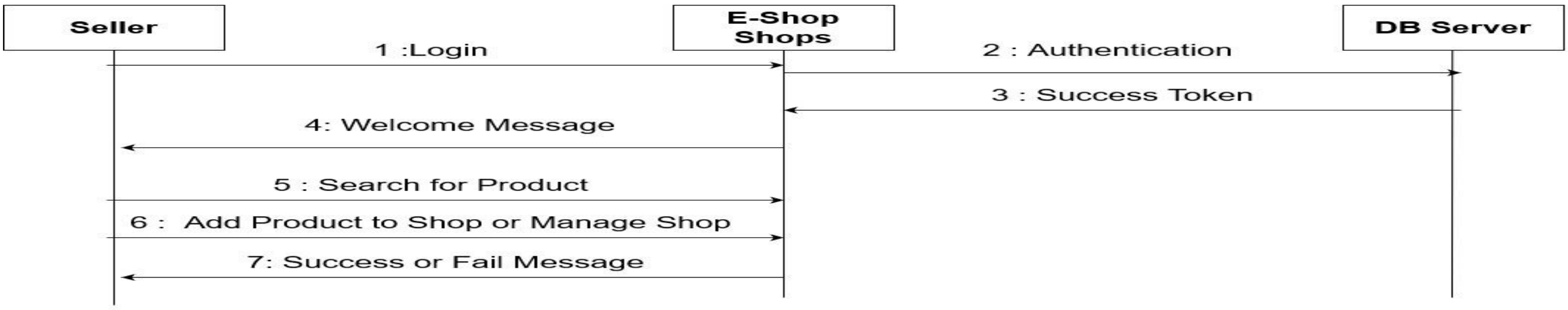


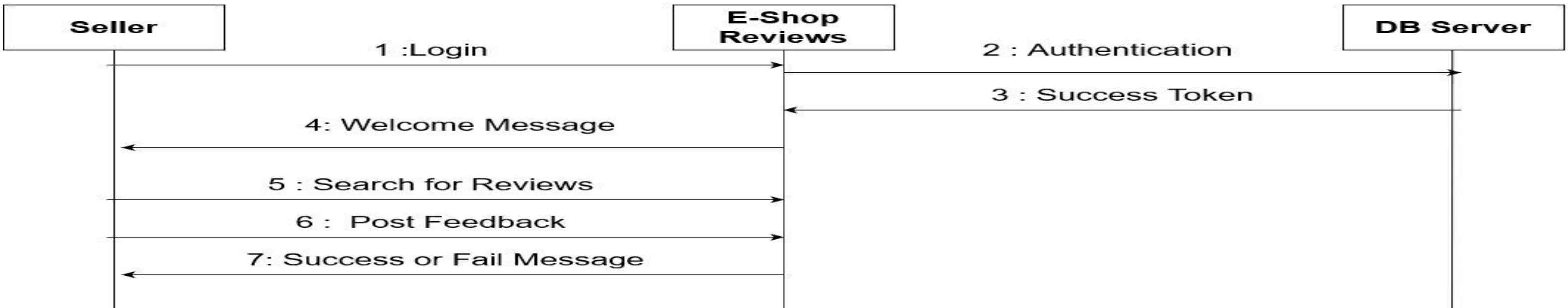
5.1.5 Sequence Diagram (2nd Version)

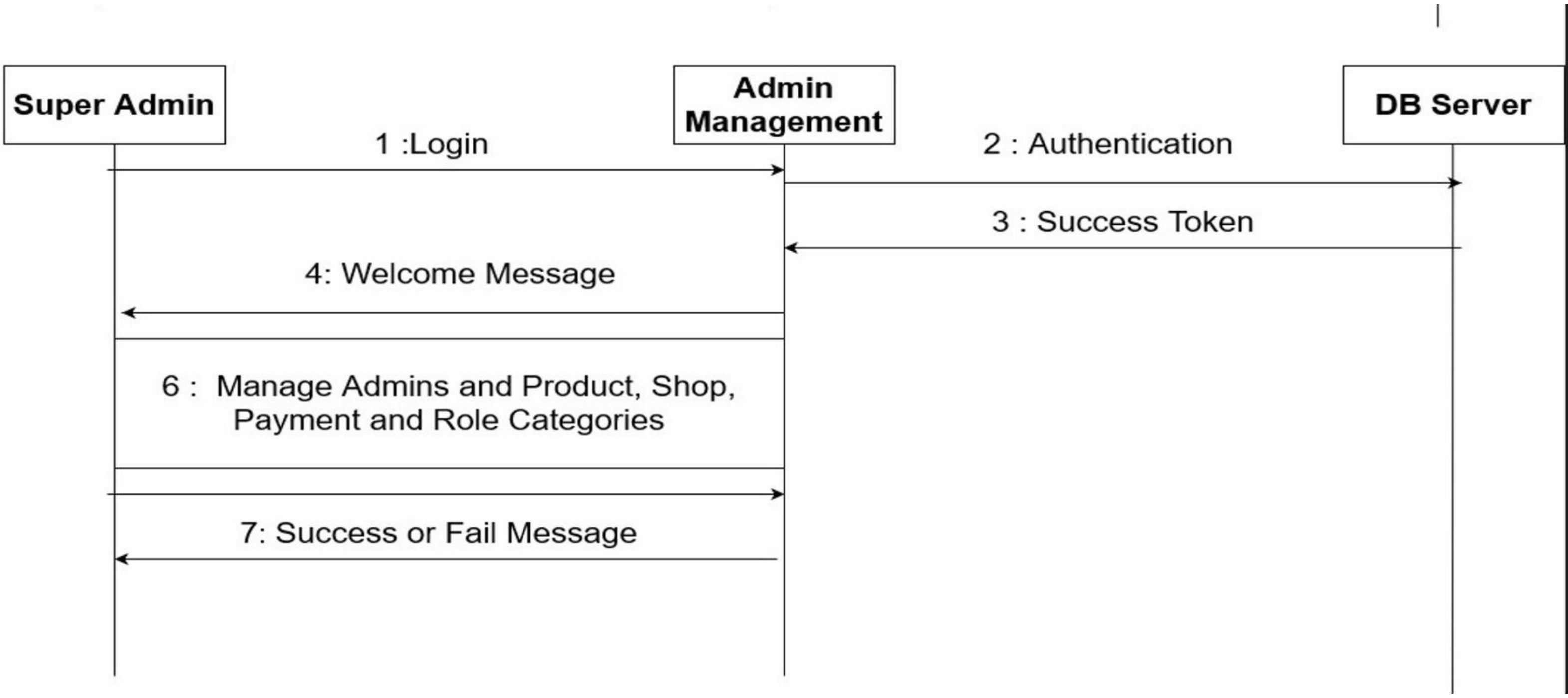






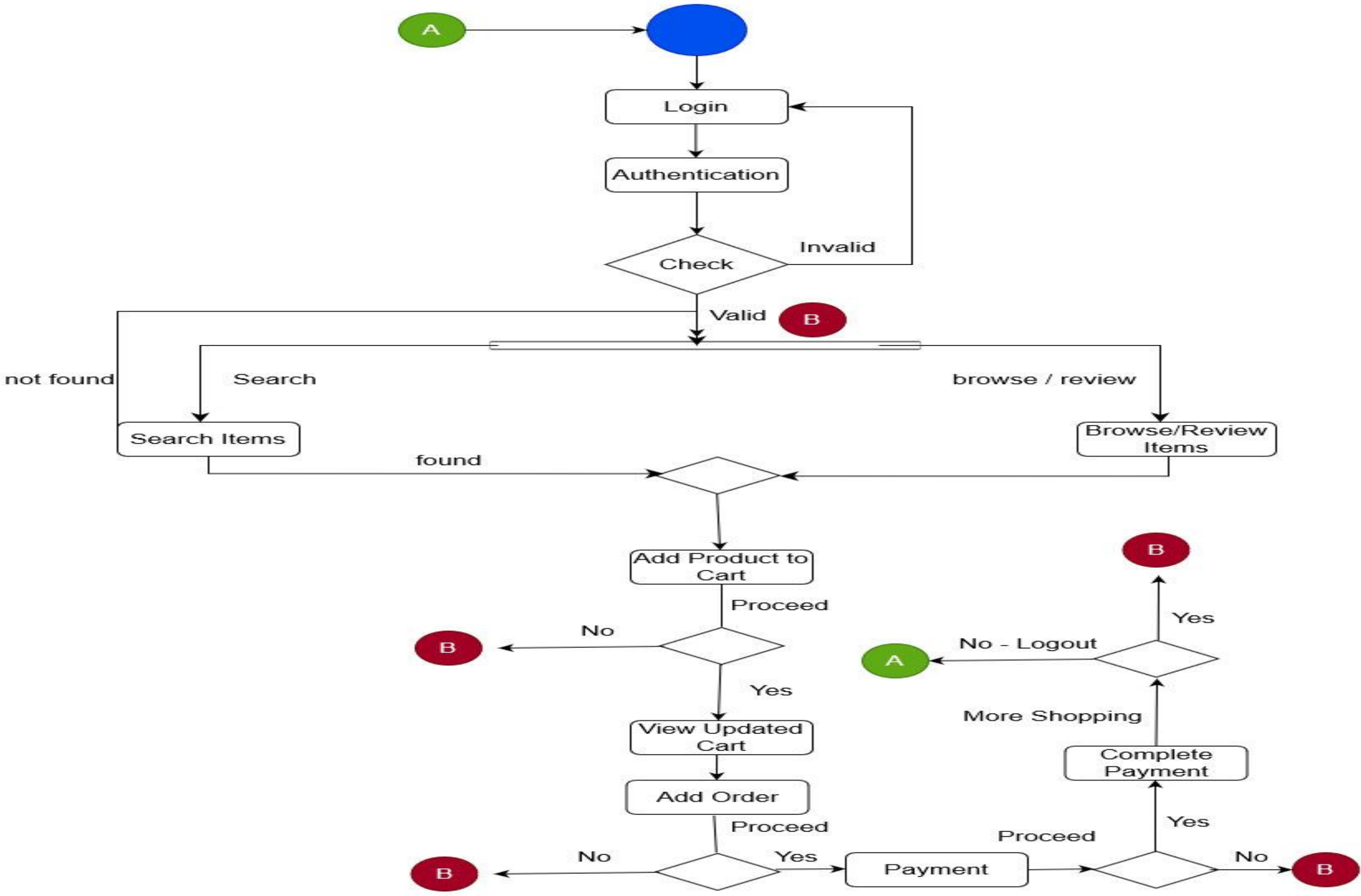




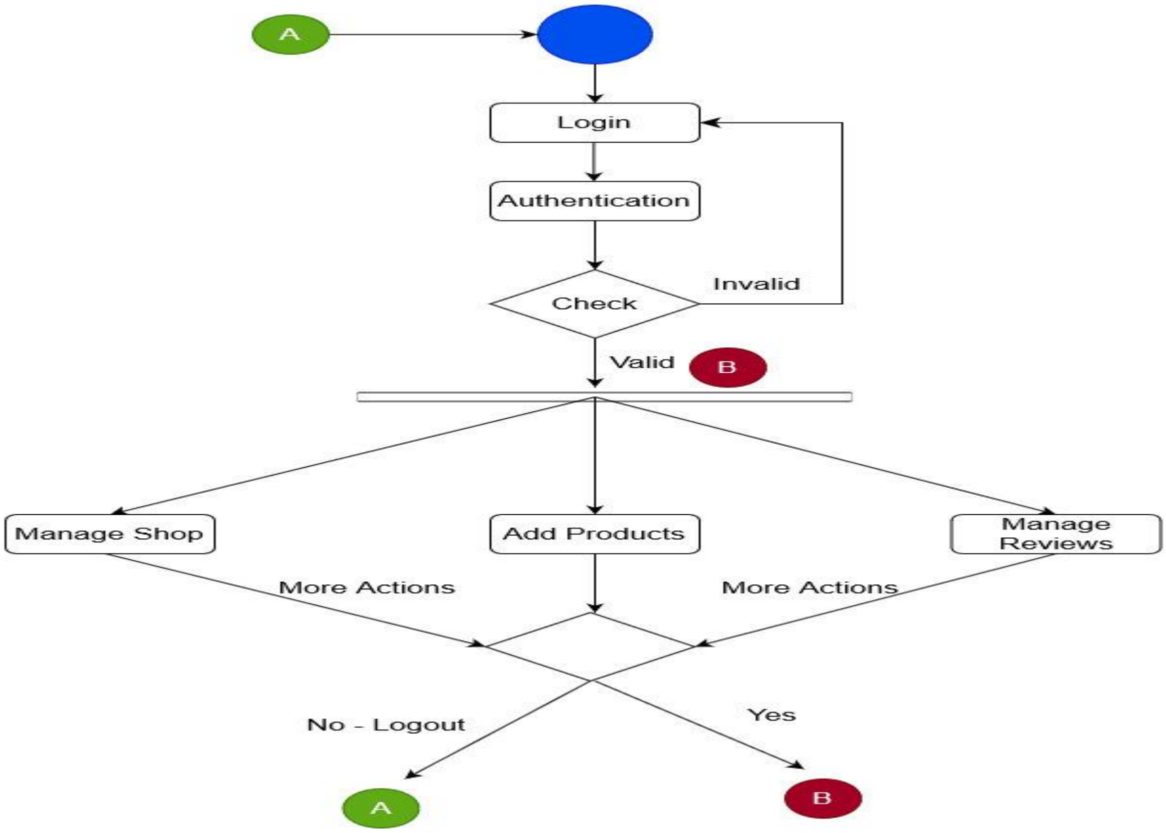

Super Admin
Admin Management
DB Server
1 :Login
2 : Authentication
3 : Success Token
4: Welcome Message
6 : Manage Admins and Product, Shop, Payment and Role Categories
7: Success or Fail Message

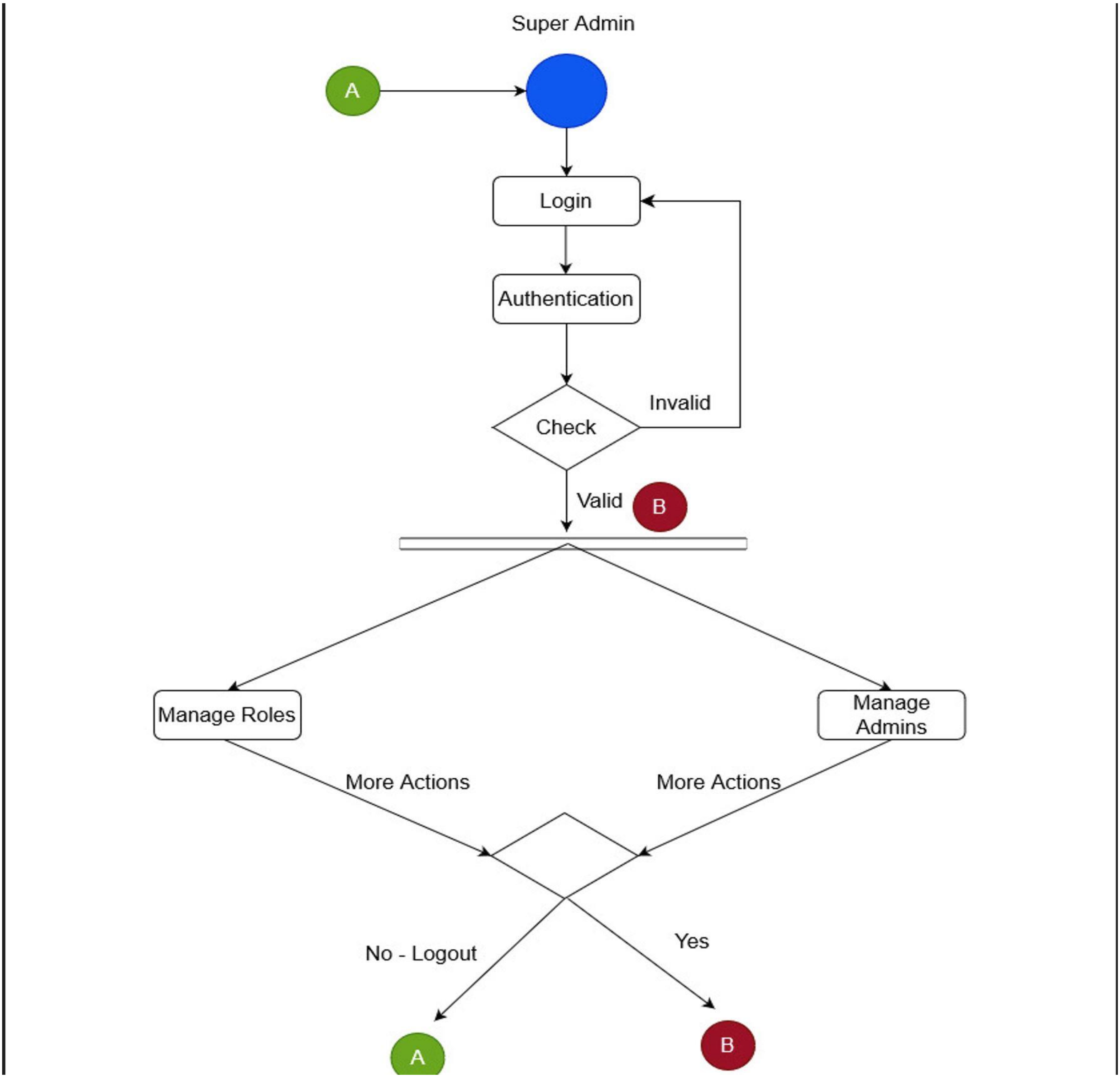
5.1.6 Activity Diagram (2nd Version)

Customer

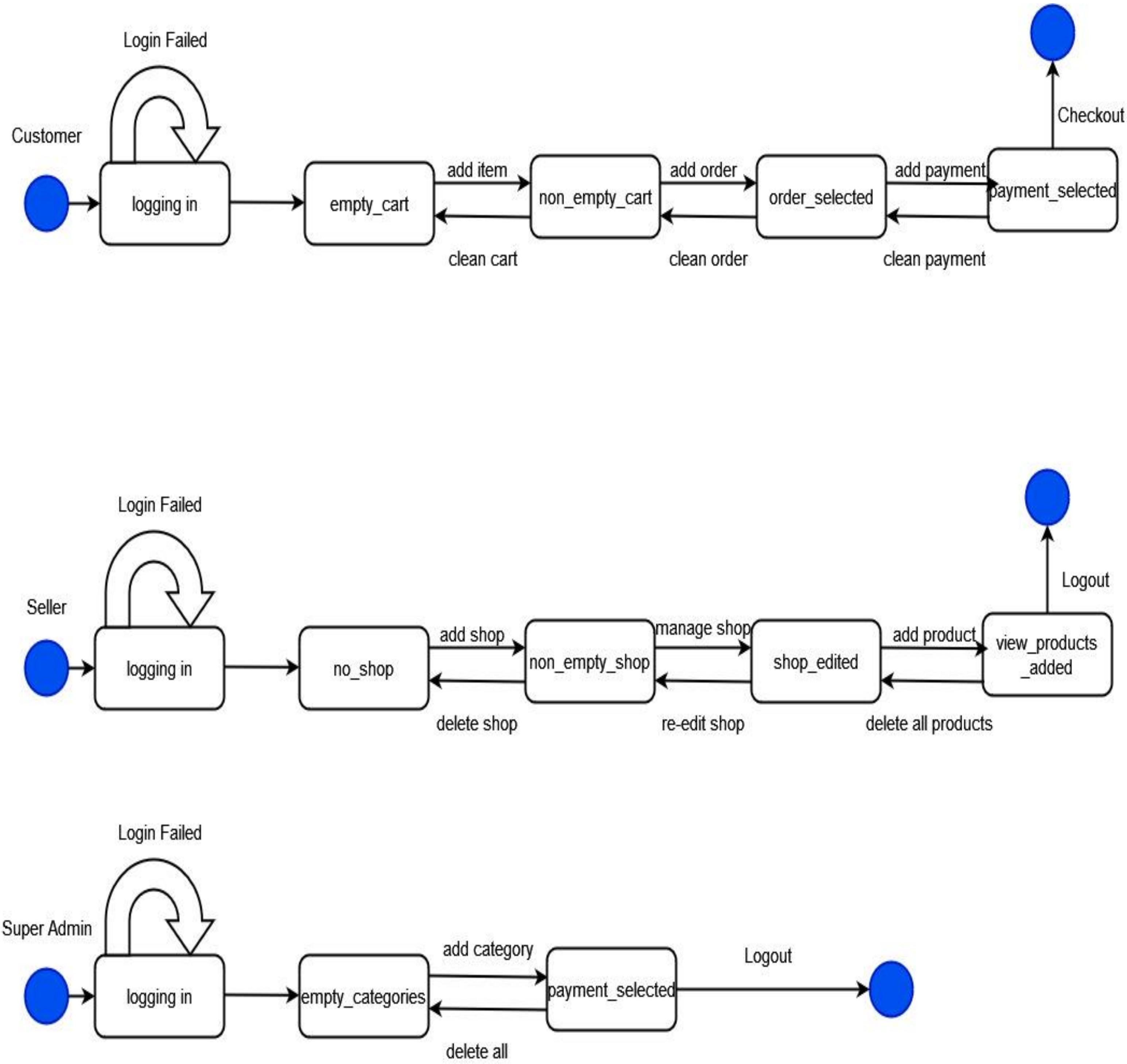


Seller

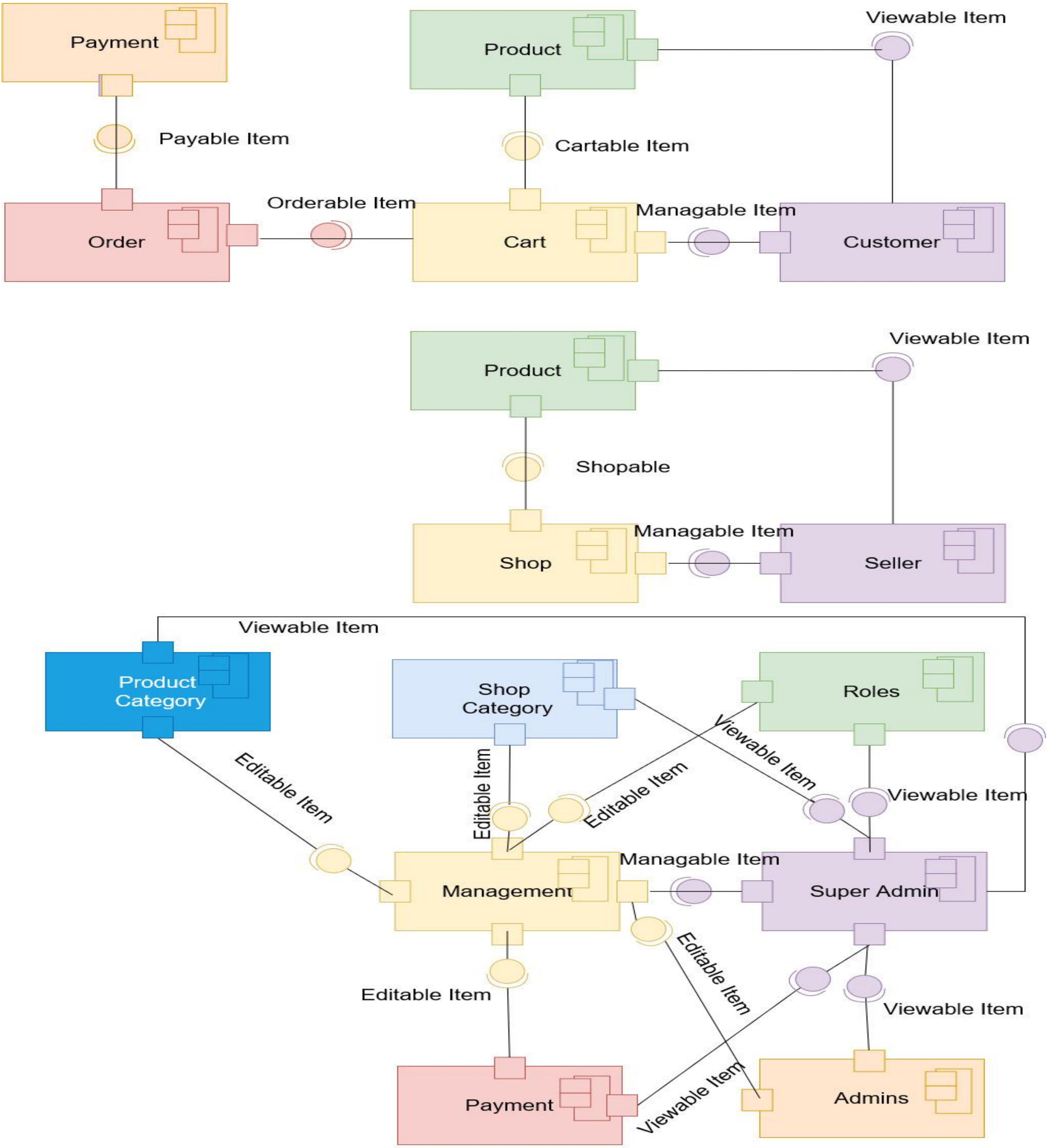



Super Admin
A
Login
Authentication
Check
Invalid
Valid
B
Manage Roles
Manage Admins
More Actions
More Actions
No - Logout
Yes
A
B

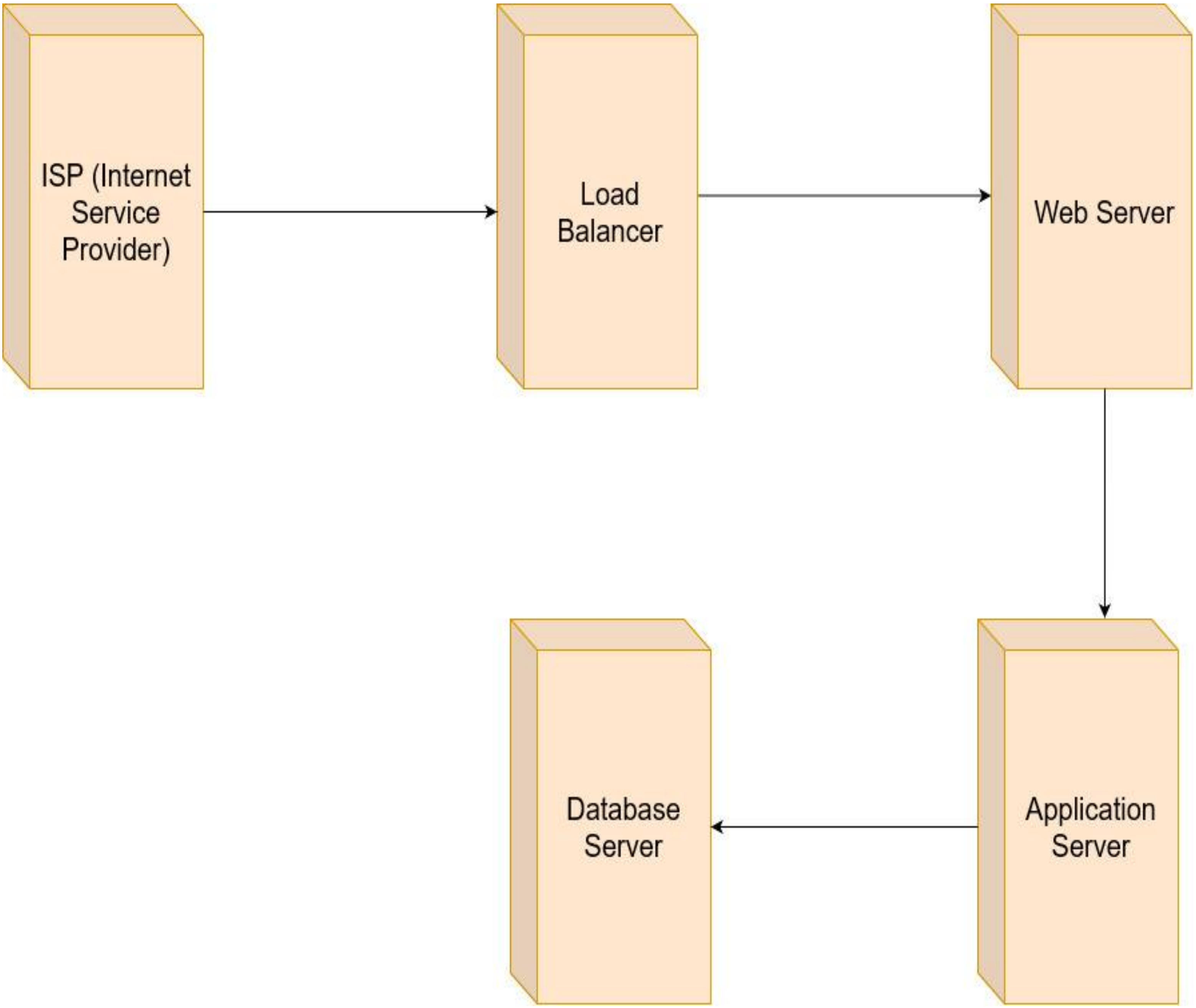
5.1.7 State Chart Diagram (2nd Version)



5.1.8 Component Diagram (2nd Version)



5.1.9 Deployment Diagram (2nd Version)


ISP (Internet Service Provider)
Load Balancer
Web Server
Database Server
Application Server