

---

# **System Architectural Analysis**

**for**

## **Book Rental and Sales System**

**Version 1.6 approved**

**Prepared by Trinh Thai Linh  
Le Quang Chinh  
Pham Tung Duong  
Do Tien Dung  
Nguyen Hoang Anh Dung**

**Group 7**

**October 29, 2023**

# Table of Contents

<b>Table of Contents.....</b>	<b>ii</b>
<b>Revision History.....</b>	<b>v</b>
<b>1. Introduction.....</b>	<b>1</b>
1.1 Purpose.....	1
1.2 Intended Audience and Reading Suggestions .....	1
1.3 Product Scope .....	2
1.4 References.....	2
<b>2. Key Abstraction.....</b>	<b>3</b>
2.1 Diagram .....	3
2.2 Definitions .....	3
<b>3. Architectural Representation .....</b>	<b>4</b>
<b>4. Architectural Goals and Constraints .....</b>	<b>5</b>
<b>5. Use Case View.....</b>	<b>6</b>
5.1 Use Case Realizations.....	7
5.1.1 Guest Use Case .....	7
5.1.2 Member Use Case .....	7
5.1.3 Administrator Use Case .....	8
5.2 Significant Use Case Descriptions.....	9
<b>6. Logical View .....</b>	<b>11</b>
6.1 Overview.....	11
6.2 Architecture Significant Design Packages.....	11
6.2.1 Presentation Package.....	11
6.3 Application Package .....	13
6.3.2 Domain Package .....	14
6.3.3 Persistence Package .....	18
<b>7. Process View .....</b>	<b>20</b>

<b>8. Deployment View .....</b>	<b>21</b>
<b>9. Implementation View.....</b>	<b>22</b>

## Table of Figures

Figure 2-1: Key Abstractions .....	3
Figure 5-1: Use Case Model for Guest.....	7
Figure 5-2: Use Case Model for Member.....	8
Figure 5-3: Use Case Model for Administrator.....	9
Figure 6-1: Presentation Package Class Diagram .....	12
Figure 6-2: Application Package Class Diagram .....	14
Figure 6-3: Domain Package Diagram .....	15
Figure 6-4: Profile Package Diagram .....	16
Figure 6-5: Order Package Diagram.....	17
Figure 6-6: Reading Package Diagram.....	18
Figure 6-7: Persistence Class Diagram.....	19
Figure 7-1: Process View Diagrams.....	20
Figure 9-1: Implementation View Model.....	22

## Revision History

Name	Date	Reason For Changes	Version
Trinh Thai Linh	29-Oct-2023	Create Document Template	1.0
Trinh Thai Linh	29-Oct-2023	Add Introduction	1.1
Trinh Thai Linh	05-Nov-2023	Add Architecture Diagrams	1.2
Pham Tung Duong	05-Nov-2023	Add Use Case Realization	1.3
Pham Tung Duong	05-Nov-2023	Add Process View Diagram	1.4
Trinh Thai Linh	05-Nov-2023	Add Changes	1.5
Trinh Thai Linh	29-Nov-2023	Change Process View, Add Implementation View	1.6

# 1. Introduction

## 1.1 Purpose

This is a report prepared by Group 7 (Object-oriented Analysis and Design Class 20, 22-23) of Use Case Solution Analysis for Book Rental and Sales System and is written based on the reporting format “IEEE Std 830-1998 IEEE Recommended Practice for Software Requirements Specifications”.

The purpose of this report is to present the system properties, their structure and behaviour with the system’s architecture model. The contents include:

- Introduction: state the abstract of this document.
- Architecture Representation: describes what software architecture is used for the system and how it is represented.
- Key Abstraction: identify classes or objects that forms part of the vocabulary of the problem domain.
- Use Case View: represent use cases in the prespective of different actors.
- Logical View: describes the architecturally significant parts of the design models and how they are organized in different packages.

## 1.2 Intended Audience and Reading Suggestions

The different types of reader that the document is intended for are:

- **Designers:** Design the classes and the subsystems architecture that satisfies the requirements specified in this SRS.

- **Developers:** Implement codes from the design and these documents. They use this document as a blueprint, and it makes sure that, firstly, their code works properly, and secondly, it is in sync with others' codes.
- **Tester:** Use this document to understand the requirements. They have to create a test plan that outlines the scope of testing, testing objectives, test environment setup, resources required, and a schedule.
- **Documentation writers:** Ensure that the documentation aligns with the software's requirements as specified in the Software Requirements Specification (SRS). Verify that the documented information is accurate and complete.

### **1.3 Product Scope**

The software, “Book Rental and Sales System”, is designed to provide a platform for users to rent, purchase, read books in both physical and electrical form. Additionally, it includes a forum where users can discuss with other readers about the books they purchased, rented or simply ones that captured their interest. The system aims to enhance the reading experience and promote the reading community engagements.

### **1.4 References**

- [1] Form of presentation IEEE. IEEE Std 830-1998 IEEE Recommended Practice for Software Requirements Specifications. IEEE Computer Society, 1998.
- [2] System Requirements Specification Content and Format Standard, which specifies the content and format of this specification.
- [3] Previous Project of Object-Oriented Analysis and Design Course, The Cargo Carriage System.

## 2. Key Abstraction

### 2.1 Diagram

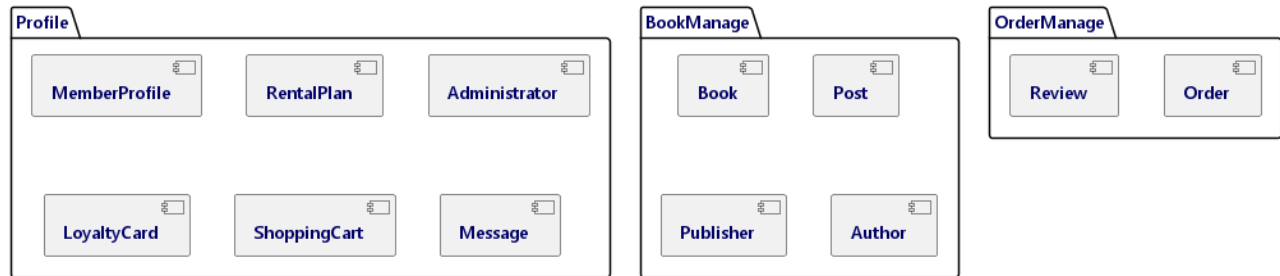


Figure 2-1: Key Abstractions

### 2.2 Definitions

- **MemberProfile:** acts as an actor to use sales, rental and reading services on the system.
- **Administrator:** acts as an actor to use management services on the system.
- **LoyaltyCard:** every Member has one Loyalty card to track membership on the system.
- **Order:** order on the system.
- **Message:** a message contains contents which a Member wants to ask Admin about.
- **Shopping Cart:** a cart to store the books that a Member wants to purchase.
- **Review:** record of the Member purchasing experience.
- **Post:** contains messages that are published on the system to record Member's thoughts of the book he purchased/read.
- **Book:** act as a book.
- **RentalPlan:** Members can register for a book rental plan. This plan can be either monthly or annual.



### **3. Architectural Representation**

This document presents the architectural as a series of views; use case view, process view, deployment view, and implementation view. These views are presented as Rational Rose Models and use the Unified Modeling Language (UML).

## **4. Architectural Goals and Constraints**

There are key requirements and system constraints that have some significant impact on the architecture:

- The architecture must be formed by considering all the functional requirements to be implemented and supported.
- All the non-functional requirements stated in the Supplementary Specification must be taken into consideration in the architecture.
- The architecture must make sure that the system is secure since transactions are carried out on the system.
- The architecture must take into consideration of the implementation environment (hardware, on what platforms, database management systems,...).

## **5. Use Case View**

The Use Cases in this system are listed below:

1. SignUp
2. SignIn
3. Logout
4. Edit Profile
5. Check Book's Detail
6. Search Books
7. Add Books to Favorite List
8. Add to Cart
9. View Member Card
10. Buy Books
11. Review Order
12. Request Refund
13. Read eBook
14. Manage Posts on Forum
15. Rent Books
16. Register for a Rental Plan
17. Ask Chat
18. Manage Order
19. Cancel Order
20. Update Books' Information
21. Sell Used Book

## 5.1 Use Case Realizations

### 5.1.1 Guest Use Case

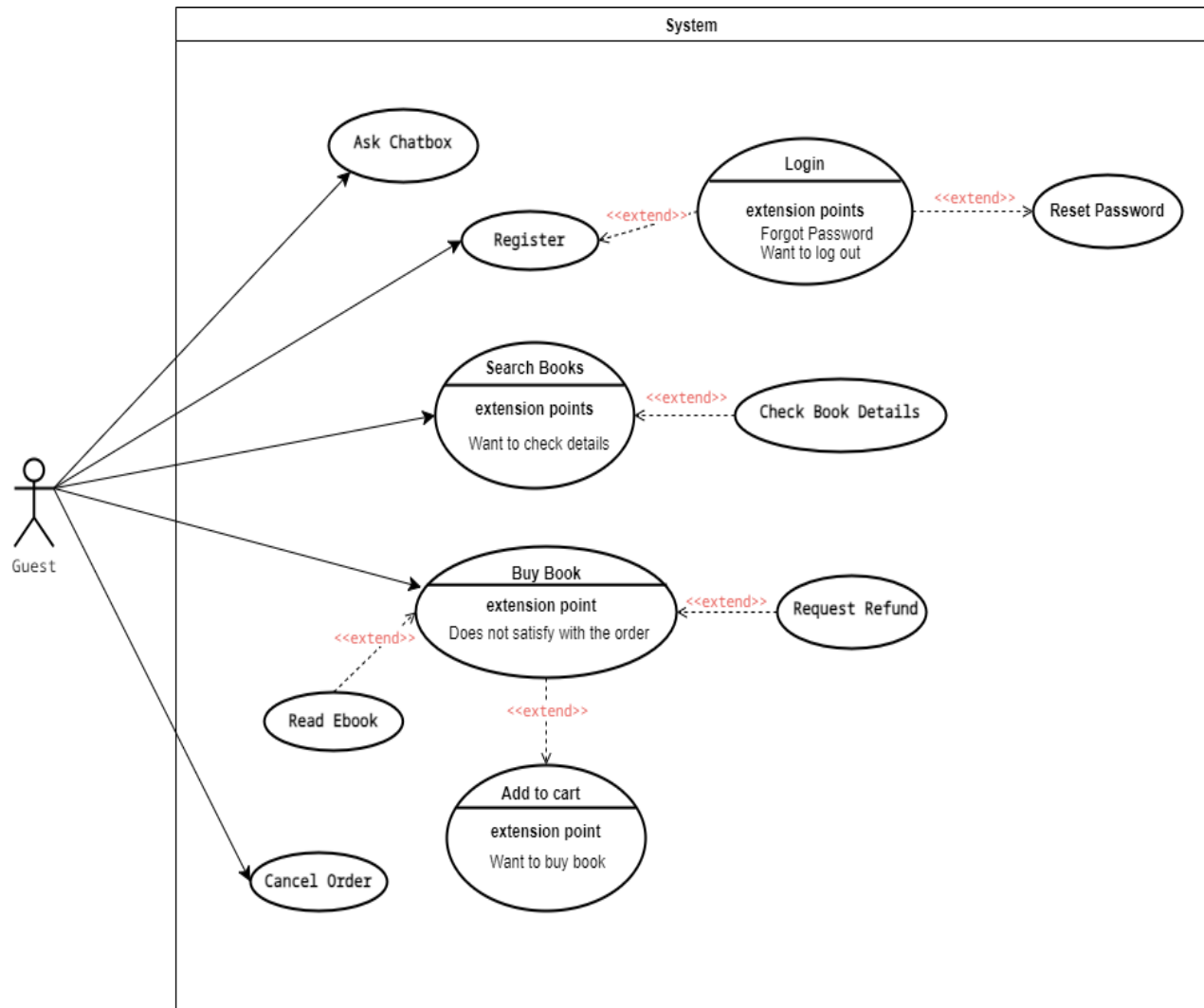


Figure 5-1: Use Case Model for Guest.

### 5.1.2 Member Use Case

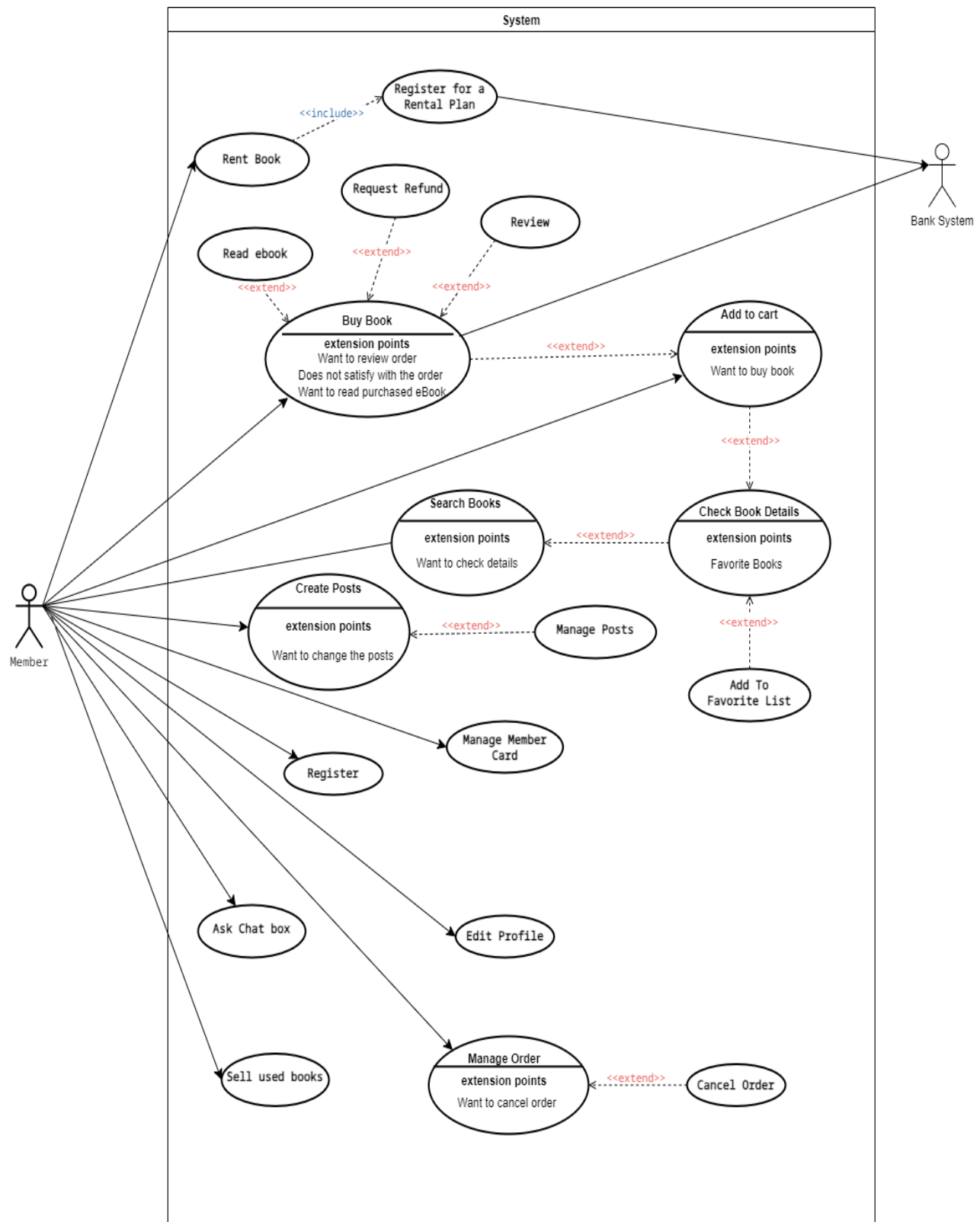


Figure 5-2: Use Case Model for Member

### 5.1.3 Administrator Use Case

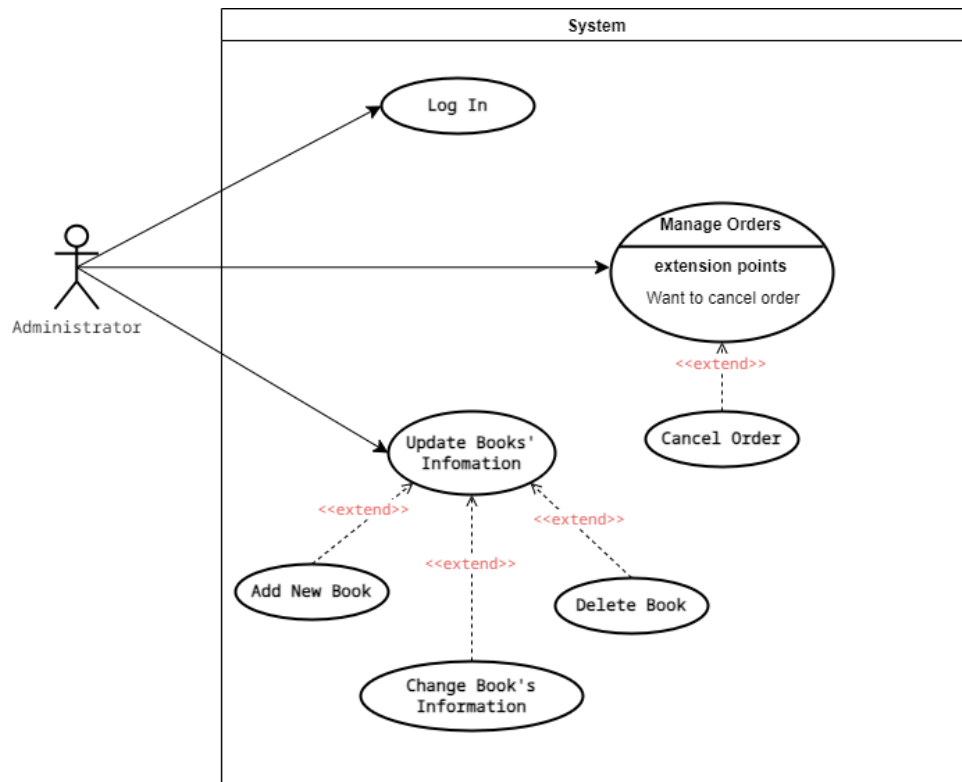


Figure 5-3: Use Case Model for Administrator

## 5.2 Significant Use Case Descriptions

- **Check Book's Details:** This Use Case that allows a Member to view detailed information of a book listed in the system.
- **Search Books:** This Use Case allows a Member to use searching functions to check if the book he wants to buy/rent is exist.
- **Update Cart:** This Use Case allows a Member to choose a product and add it to cart, delete it from cart or change product's quantity in the cart.
- **Buy Books:** The Use Case allows a Member to make purchases for selected book(s).
- **Review Order:** A Member can choose to provide feedback on the quality of the product or the quality of the order fulfillment process after completing an order on the system.
- **Request Refund:** This Use Case allows a Member to send a Refund Request to the Administrator.
- **Read eBook:** This Use Case allows a Member to read books online from his own library after he has purchased or chosen to rent any book.

- **Manage Posts on Forum:** This Use Case allows a Member to create, delete, close or edit his posts on the form.
- **Register for a Rental Plan:** This Use Case allows a Member join a Rental Plan on the System. Currently, this System is only provide a Monthly-Plan.
- **Manage Order:** This Use Case allows a Member or the Administrator to keep track of their orders.
- **Cancel Order:** This Use Case allows a Member to cancel an order.
- **Update Books' Information:** This Use Case allows the Administrator to add, edit or delete a book.
- **Sell Used Book:** This Use Case allows a Member to request to sell his used books on the system.

## 6. Logical View

### 6.1 Overview

This part represents the description of the logical view of the architecture, describe most important classes, their organization in service packages and subsystems, and how these subsystems are organized into layers. Class diagrams are included to visually illustrate the relationships between all the classes, packages, subsystems, and layers. The logical view of the Book Rental and Sales System has 5 main packages:

- **Presentation (UI Layer):** this layer handles the interactions that users have with the system and consists of all the Boundary Classes.
- **Application:** this layer handles the main programs of the architecture and how the functions of the application work. This layer contains all the Control Classes.
- **Domain (Business Layer):** this layer is where the application's business logic operates, determines the behaviour of the entire application and is based on the organization's guidelines.
- **Persistence (Data Access Layer):** this layer acts as a protective layer and is the layer used to access the database layer.
- **Database Layer:** this layer is where the system stores all the data.

### 6.2 Architecture Significant Design Packages

#### 6.2.1 *Presentation Package*

##### 6.2.1.1 *Brief Description*

This package contains all the boundary classes through which the Actors interacts with this System. Each Use Case has one boundary class to support displaying the information, forms, getting requests and inputs from the Actor.



## 6.2.1.2 Diagram

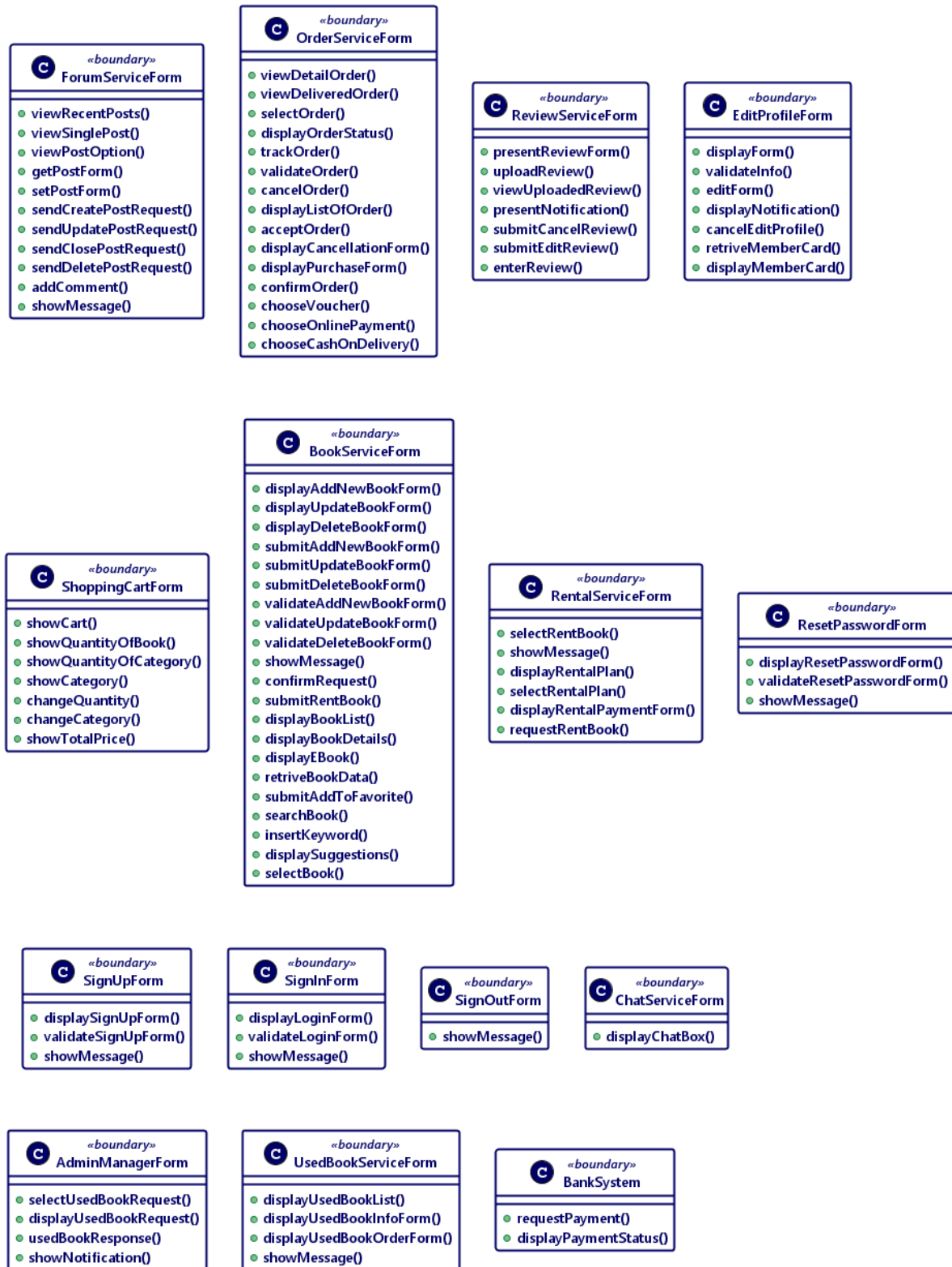


Figure 6-1: Presentation Package Class Diagram

## **6.3 Application Package**

### *6.3.1.1 Brief Description*

This package contains all the control classes which encapsulates functionalities of one or several use cases.

## 6.3.1.2 Diagram

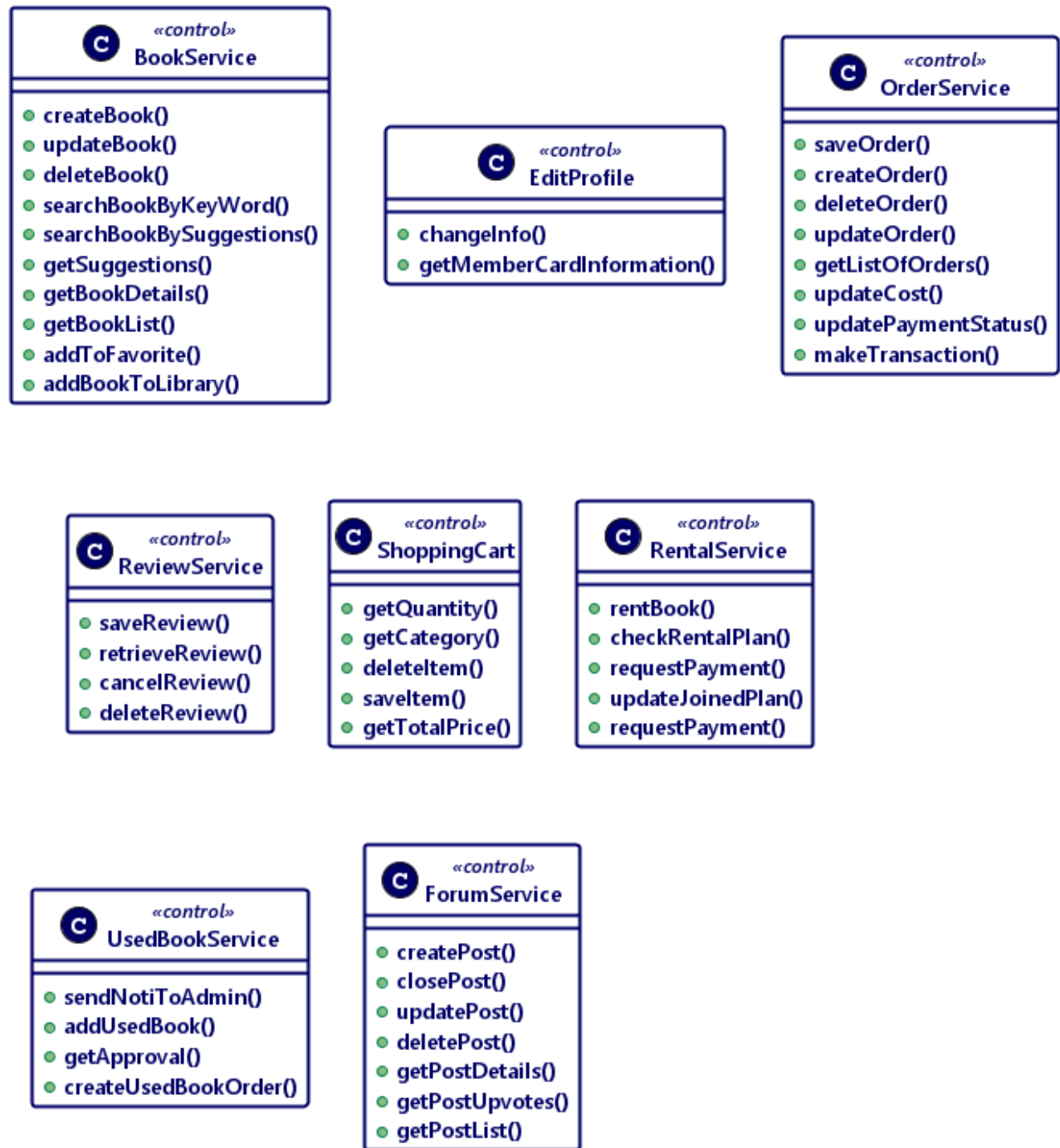


Figure 6-2: Application Package Class Diagram

## 6.3.2 Domain Package

### 6.3.2.1 Brief Description

This package contains packages related to business's logic and guidelines. In these packages are all the entity class to handles the data of the system:

- Profile Package: this package consists of classes related to user's profile (Member and Admin).
- Order Package: this package contains classes that handle data for order processing.
- Reading Package: this package contains classes that handle data for user's reading experience processing.

### 6.3.2.2 Diagram



Figure 6-3: Domain Package Diagram

## 6.3.2.3 Profile Package

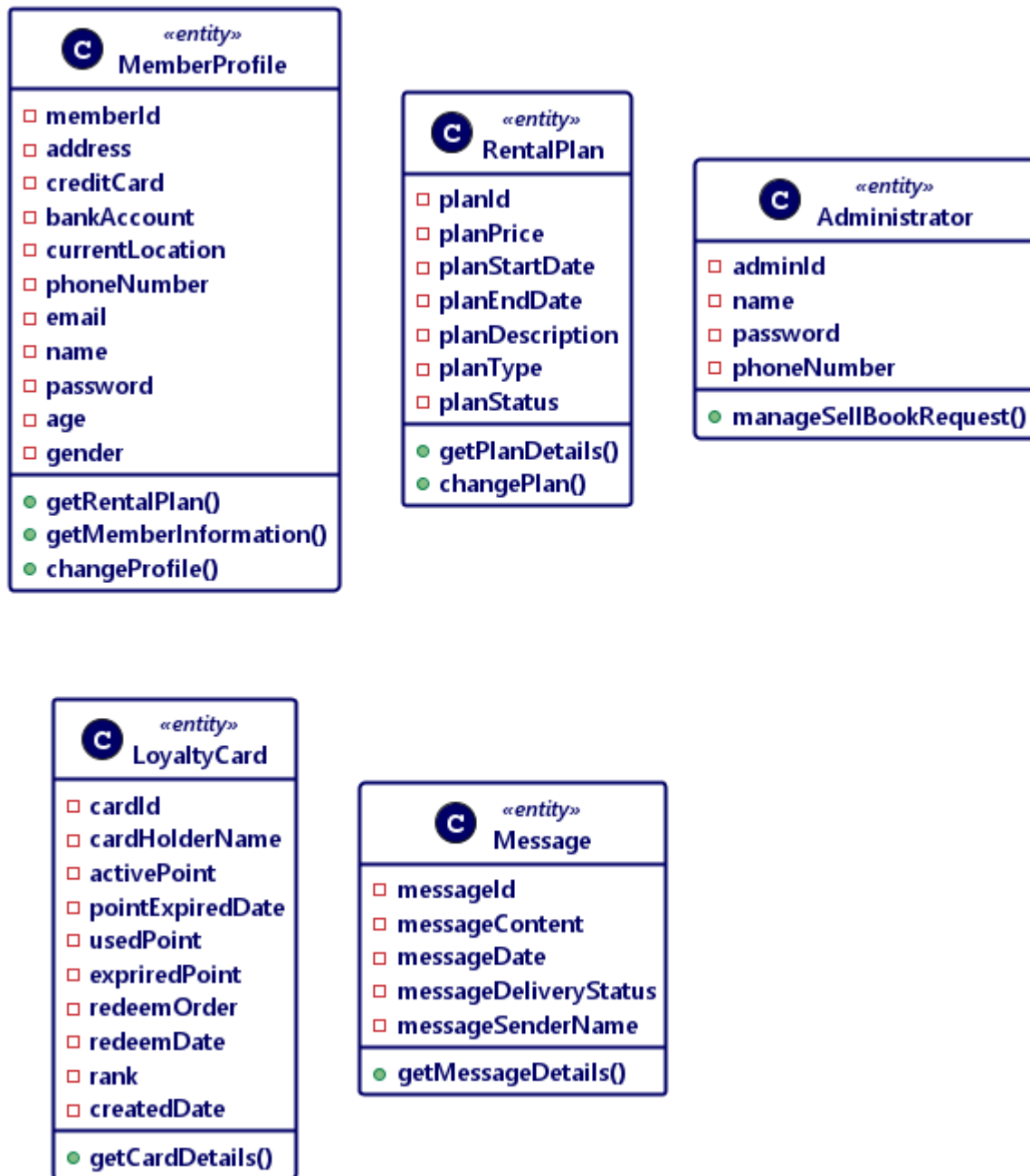


Figure 6-4: Profile Package Diagram

## 6.3.2.4 Order Package



Figure 6-5: Order Package Diagram

### 6.3.2.5 Reading Package

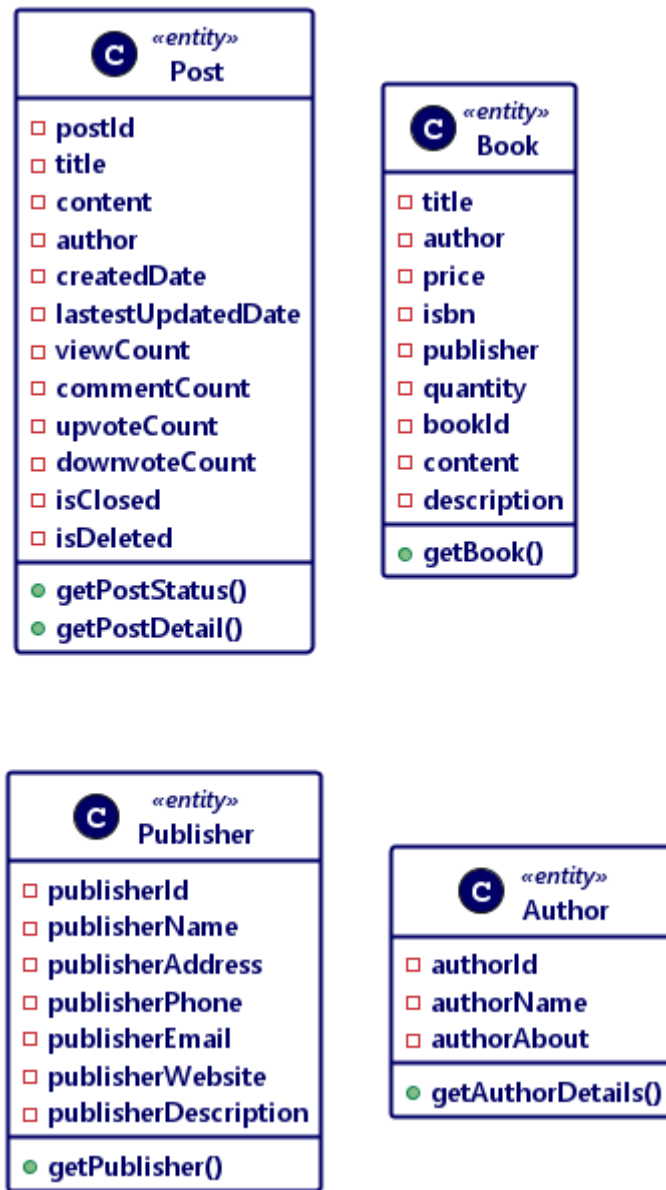


Figure 6-6: Reading Package Diagram

## 6.3.3 Persistence Package

### 6.3.3.1 Brief Description

This package acts as a protective package and is the package used to access to the databases.

## 6.3.3.2 Diagram

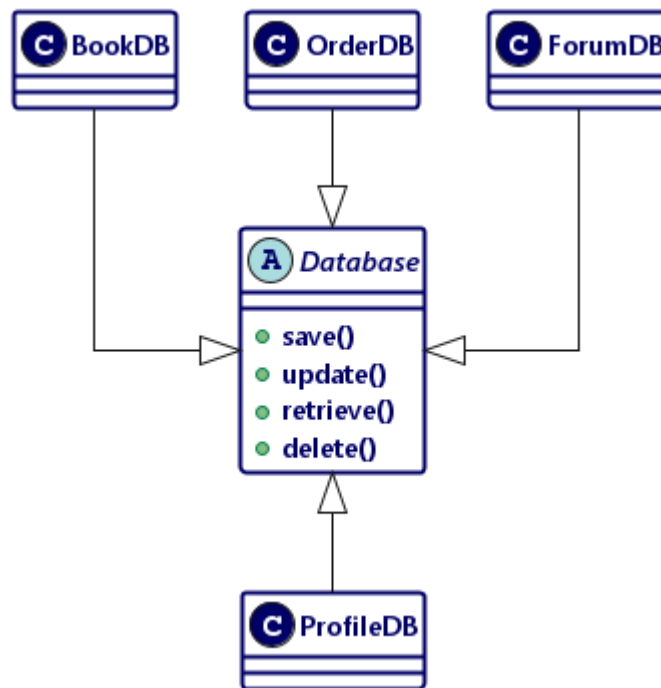
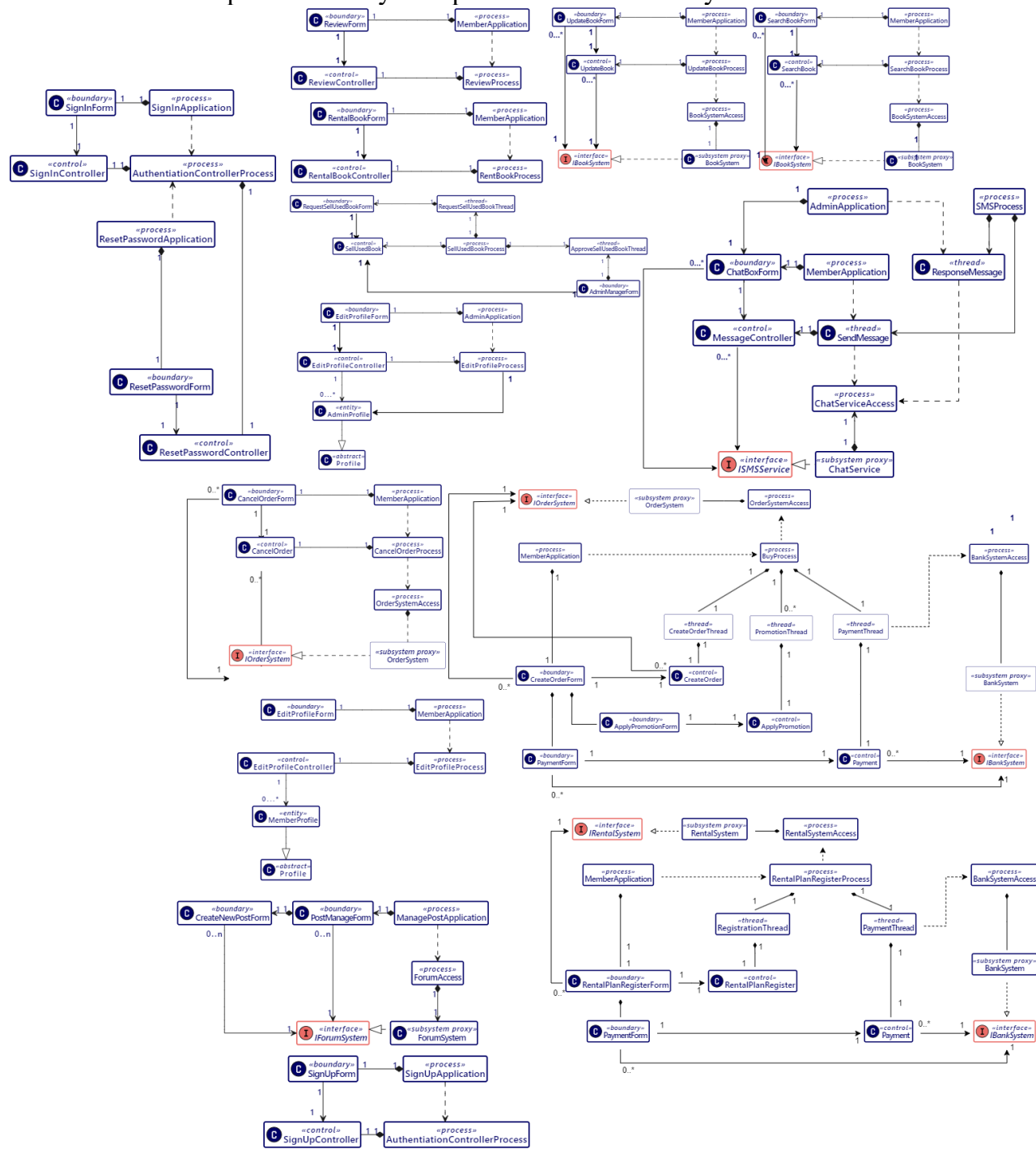


Figure 6-7: Persistence Class Diagram



## 7. Process View

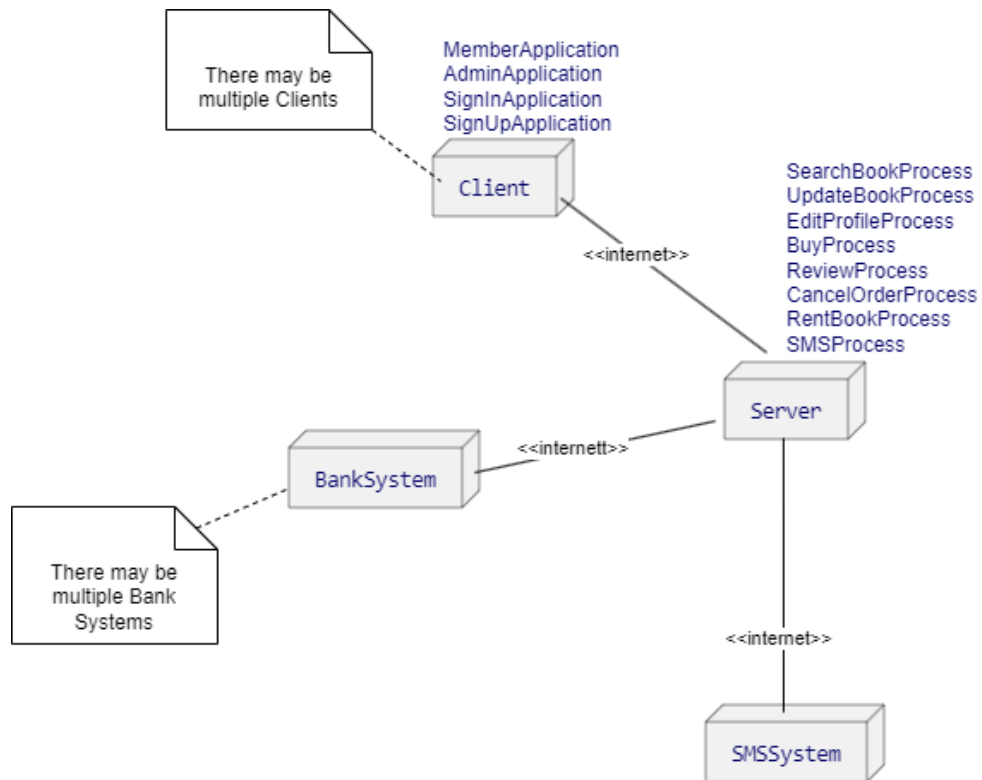
This section presents the system processes and how they interact.



*Figure 7-1: Process View Diagrams*

## 8. Deployment View

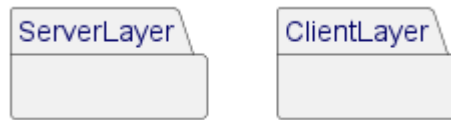
This section describes the configuration of run time processing nodes and the components that live on them through a deployment diagram.



Hình 8-1: Deployment Diagram.

## 9. Implementation View

The system contains two layers: Server Layer and Client Layer. These layers access through Internet Connection.



*Figure 9-1: Implementation View Model.*

Layer Definitions:

- **Server Layer:** Server layer supports several different application servers, where “application” includes static web pages. The existence of a server is known to the network server. In general servers are managed by the network server, but application programs may partly take over this responsibility.
- **Client Layer:** The client layer is where the user accesses the application. The server layer accepts requests through the internet connection from client layer and passes these requests to the appropriate agent. The server then relays the response from the agent back to the client layer. In this cases, the client is simply a browser.