



# **Requirement & Design Specification**

## **Book System**

**Version: 1.0**

## Record of Changes

Version	Date	A* M, D	In charge	Change Description
V1.0	17/03	A	DucTD	Create documents
V1.0	17/03	A	DucTD	Add template
V1.0	18/3	A	ToanLV	Add context diagram, actors
V1.0	18/3	M	DucTD	Change use case diagram, add screen, add use case list and description
V1.0	19/3	M	ToanLV	Update Overall Functionalities
V1.0	22/3	A	ToanLV	Add package diagram, add use case specifications
V1.0	23/3	M	DucTD	Final Report

\*A - Added M - Modified D - Deleted

## Contents

Record of Changes	2
I. Overview	5
1. Business Requirements	5
1.1 Product Visions	5
1.2 Scope	5
2. User Requirements	5
2.1 Actors	5
2.2 Use Cases	6
3. Overall Functionalities	6
3.1 Screens Flow	6
3.2 Screen Descriptions	7
3.3 Screen Authorization	7
3.4 Non-UI Functions	7
4. System High Level Design	8
4.1 Database Design	8
4.2 Code Packages	8
II. Screen Designs	15
1. <<Feature Name>>	15
1.1 <<SubFeature Name>>	15
1.2 System Access	15

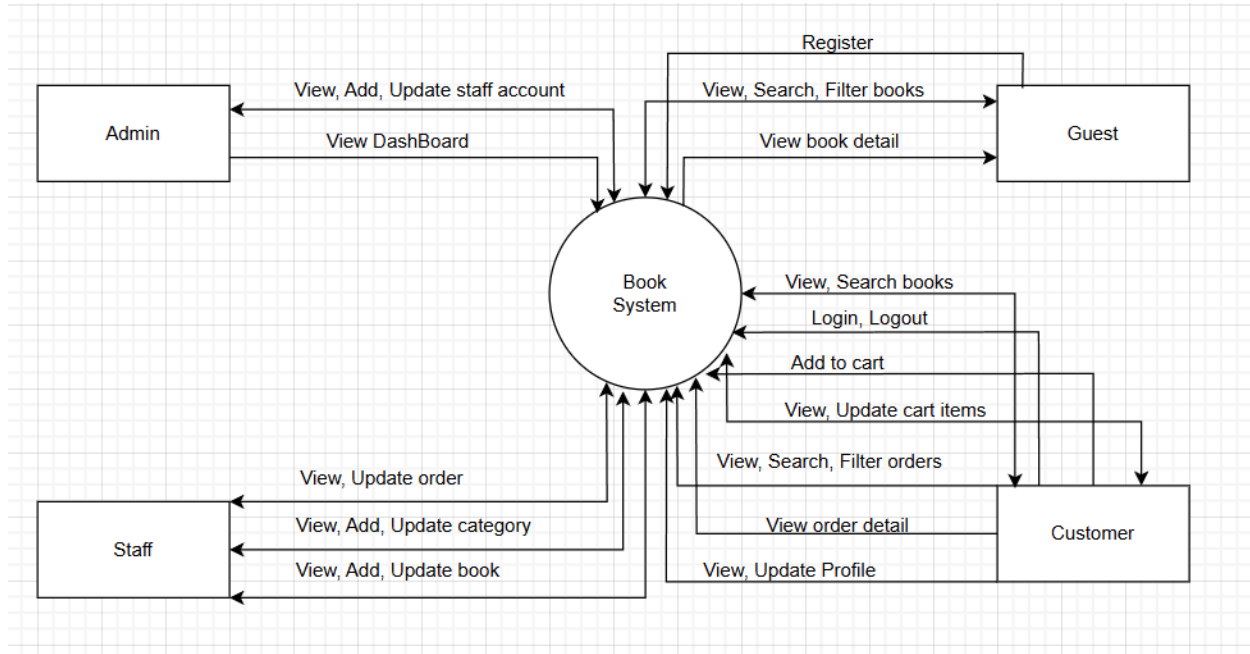
# I. Overview

## 1. Business Requirements

### 1.1 Product Visions

Create a simple and convenient online book shopping experience, helping users easily search, buy books and manage orders. At the same time, the system supports admins in managing book warehouses, categories and orders effectively.

### 1.2 Context Diagram



Context Diagram

## 2. User Requirements

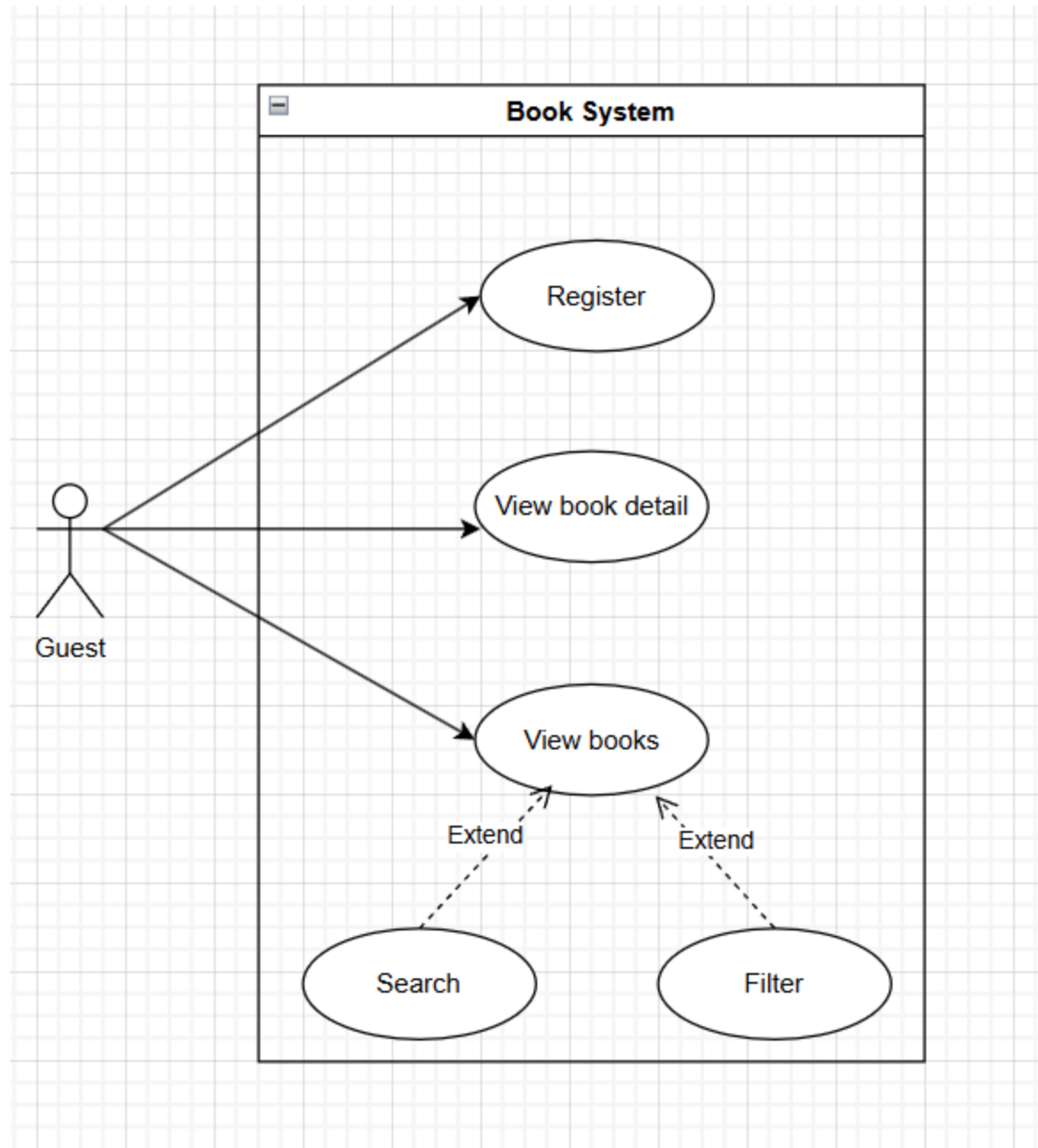
### 2.1 Actors

#	Actor	Description
1	Admin	Administrators have full control over books, categories, orders and users.
2	Staff	Staff assist with book management, cataloging and order tracking.
3	Customer	Registered users can purchase books, manage their shopping cart, personal profile and order history.
5	Guest	Users who are not logged in can view the book list and details.

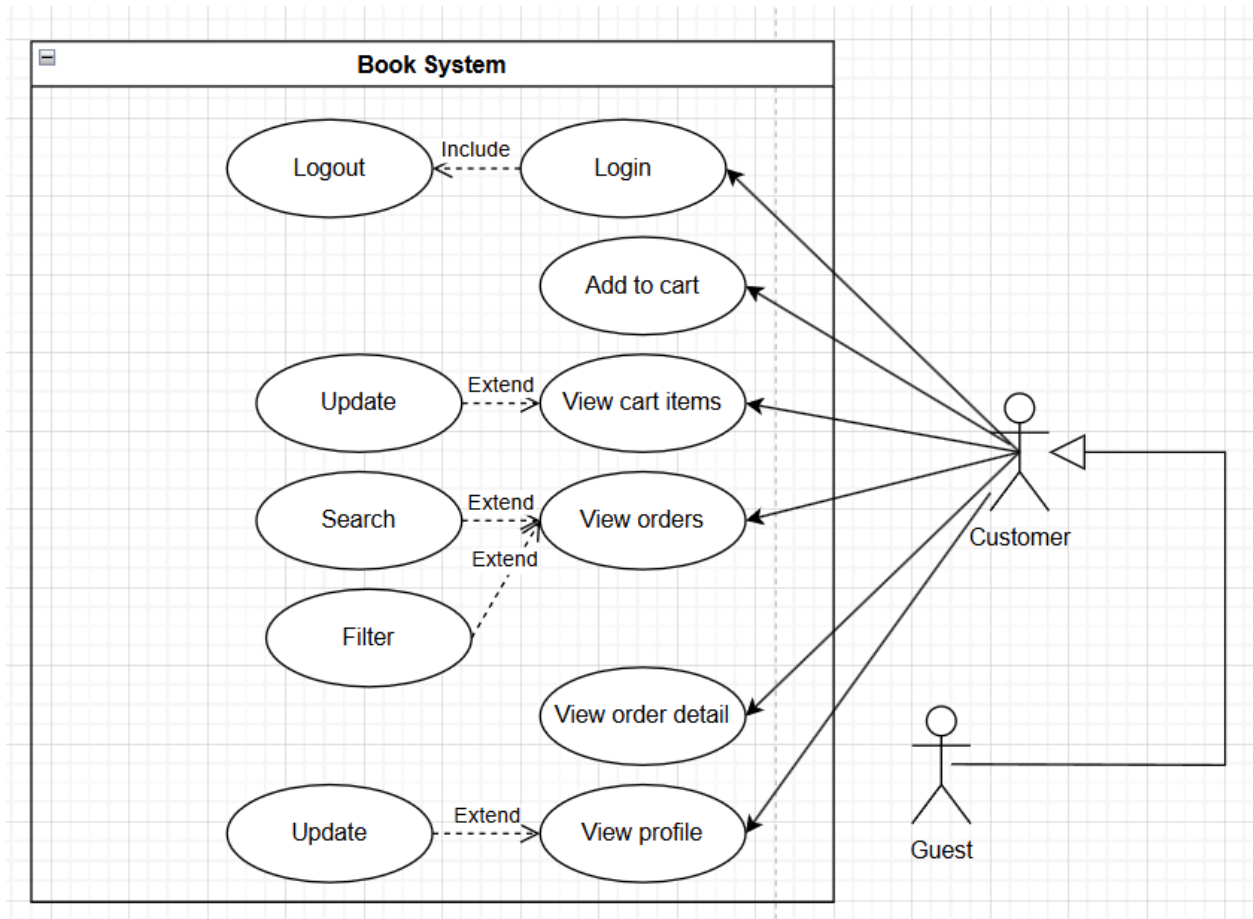
### 2.2 Use Cases

#### a. Diagram(s)

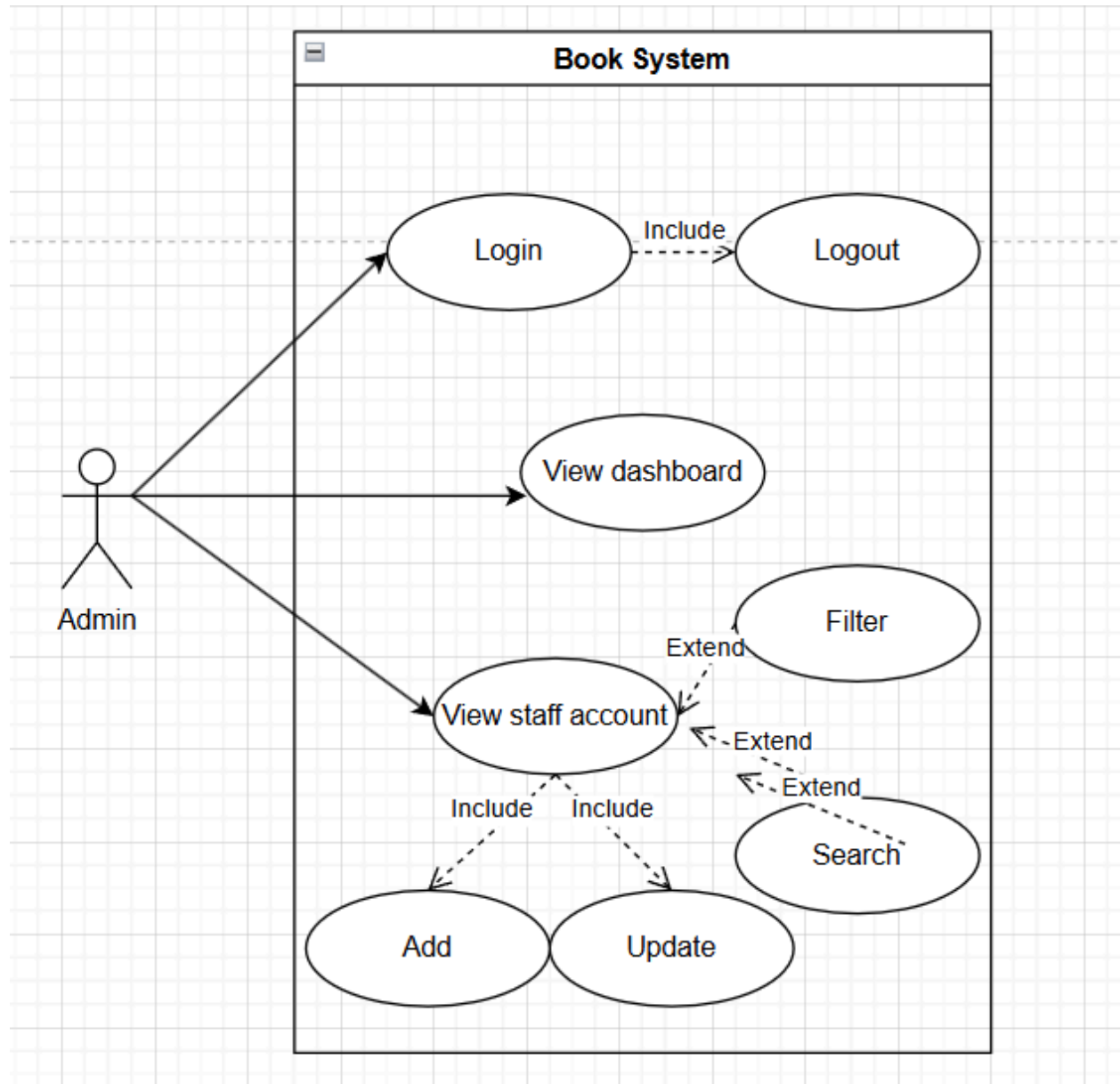
##### 1 Guest



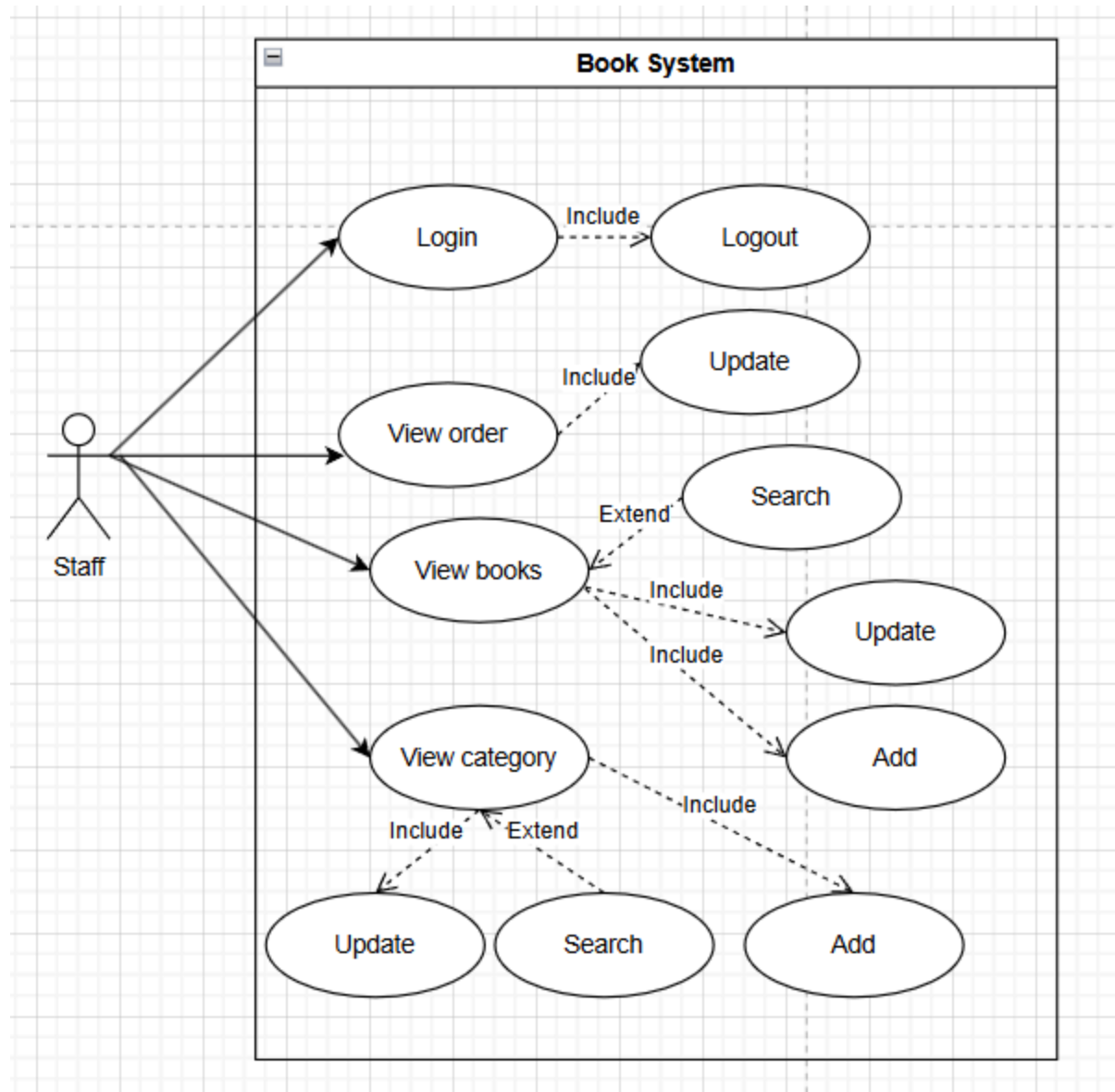
## 2 Customer



### 3 Admin



*4 staff*



### b. Descriptions

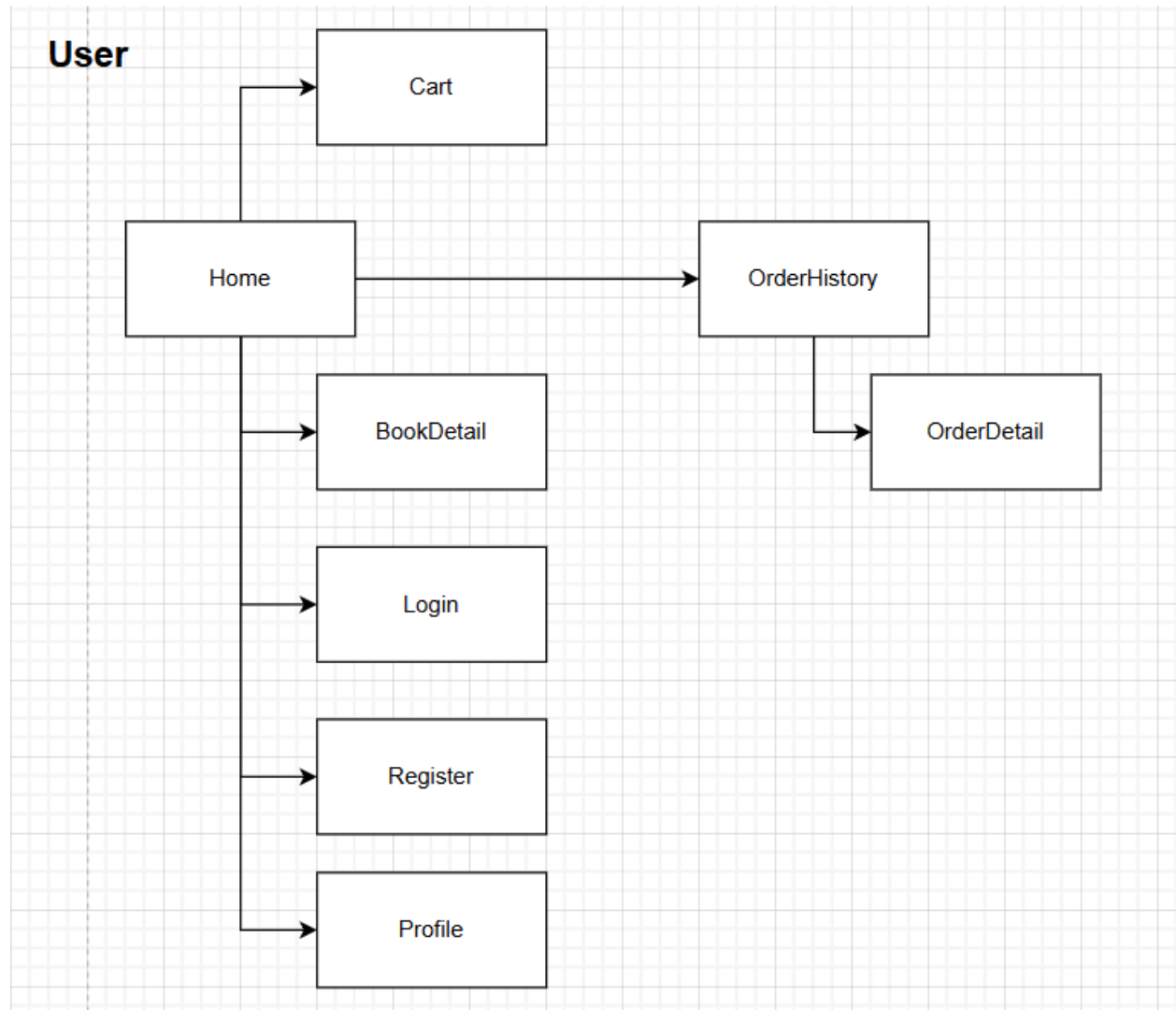
ID	Feature	Use Case	Use Case Description
01	Public	Register	Users create new accounts by entering personal information and confirming registration.
02	Public	View books	Users (guest or logged in) can view the list of books available in the system.
03	Public	View book detail	Users view detailed information about a book, including description, price, author, etc.
04	Common (Customer)	Login, Logout	Users log in to the system with their registered account or log out of the system.
05	Common (Customer)	Forgot password	User requests password reset when forgot login password.
06	Common (Customer)	View profile	Users view their personal information such as name, email, address,...

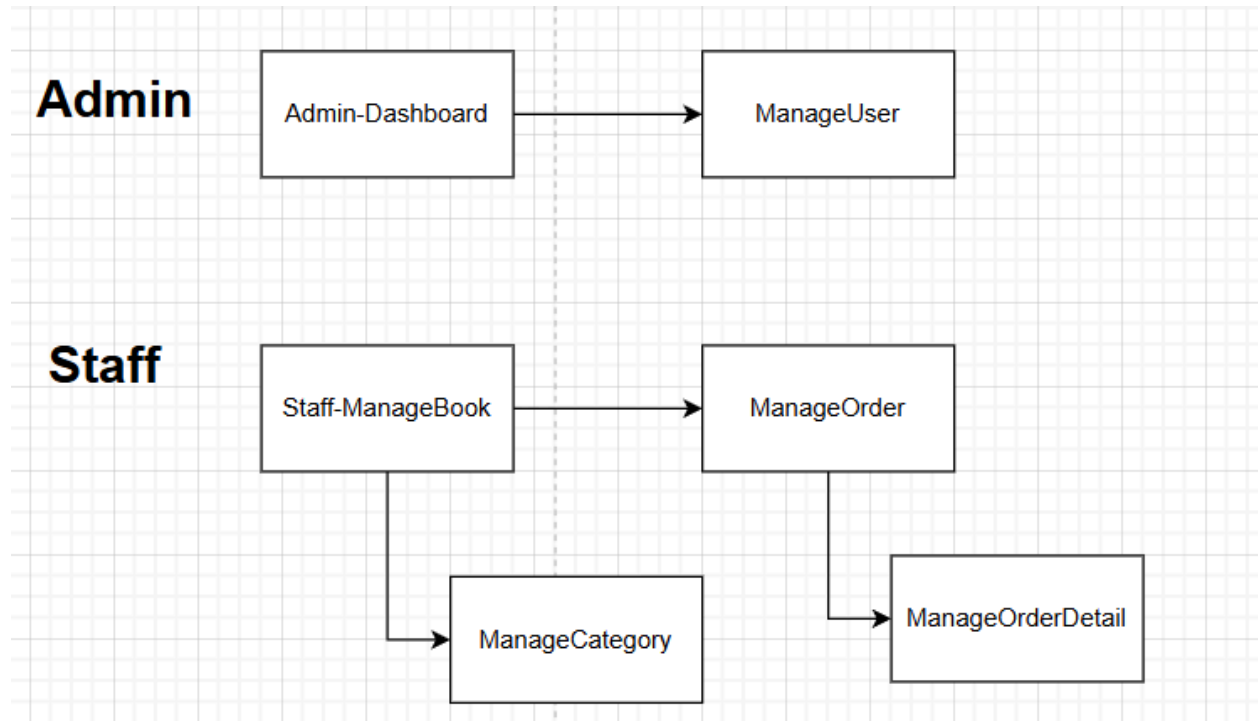


07	Common (Customer)	Add to cart	User adds book to cart to purchase later.
08	Common (Customer)	View cart items	Users view products added to cart, can update quantity or delete.
09	Common (Customer)	View order	User views list of placed orders.
10	Common (Customer)	View order detail	Users view details of an order, including status, products ordered, total amount,...
11	Admin	View Staff	Administrators view the list of employees in the system.
12	Admin	View Dashboard	Administrators view dashboards, statistics on sales, orders,...
13	Staff	View order	Staff view order list to process and support customers.
14	Staff	View books	Staff can view the list of books in the system, edit and delete.
15	Staff	View category	Staff can view the list of book categories in the system, edit and delete.

### 3. Overall Functionalities

#### 3.1 Screens Flow





### 3.2 Screen Descriptions

#	Feature	Screen	Description
1	View Homepage	Home	Displays the landing page of the application, showing a list of products (books), categories, or featured items. Users can browse products and navigate to other sections like login, register, or book details.
2	User Login	Login	Allows users to log in to their account by entering their username and password. After successful login, users are redirected to the homepage
3	User Registration	Register	Enables new users to create an account by providing details such as username, password, full name, email, and address. After successful registration, users are redirected to the login page
	View Book Details	BookDetail	Shows the user's shopping cart, listing all added products with their quantities and prices. Users can update quantities, remove items, or proceed to checkout to place an order.
	View Cart Item	Cart	Lists all past orders of the user, including order ID, date, total amount, and status. Users can click on an order to view its details.
	View Order Details	OrderDetail	Shows detailed information about a specific order, including the products ordered, quantities, unit prices, total amount, shipping

			address, and order status. Users can return to the order history or homepage.
	View Profile	Profile	Allows users to view and edit their profile information, such as full name, email, and address. Users can also log out from this screen.
	Dashboard	Dashboard	The main dashboard for admins/staff, providing an overview of the system (e.g., total users, orders, products). It includes links to manage users, products, categories, and orders.
	Manage Users	ManageUser	Allows admins to view, edit, or delete user accounts. Displays a list of users with details like username, full name, email, and role. Admins can return to the dashboard from here.
	Manage Books	ManageBook	Enables admins to add, edit, or delete products (books). Displays a list of products with details like title, author, price, stock, and category. Admins can return to the dashboard.
	Manage Categories	ManageCategory	Allows admins to add, edit, or delete product categories. Displays a list of categories with their names and descriptions. Admins can return to the dashboard.
	Manage Orders	ManageOrder	Enables admins to view and manage orders, including updating order statuses (e.g., mark as "Delivered"). Displays a list of orders with details like order ID, user, date, total amount, and status. Admins can view order details or return to the dashboard.
	Manage Order Details	ManageOrderDetail	Shows detailed information about a specific order, including the products, quantities, unit prices, total amount, and shipping address. Admins can return to the order management screen or dashboard.

### 3.3 Screen Authorization

Screen	Guest	Customer	Admin	Staff
Home	x	x		
Login	x	x	x	x
Register	x			
Profile		x		
Cart		x		
Book detail	x	x		
Order history		x		
Order detail		x		

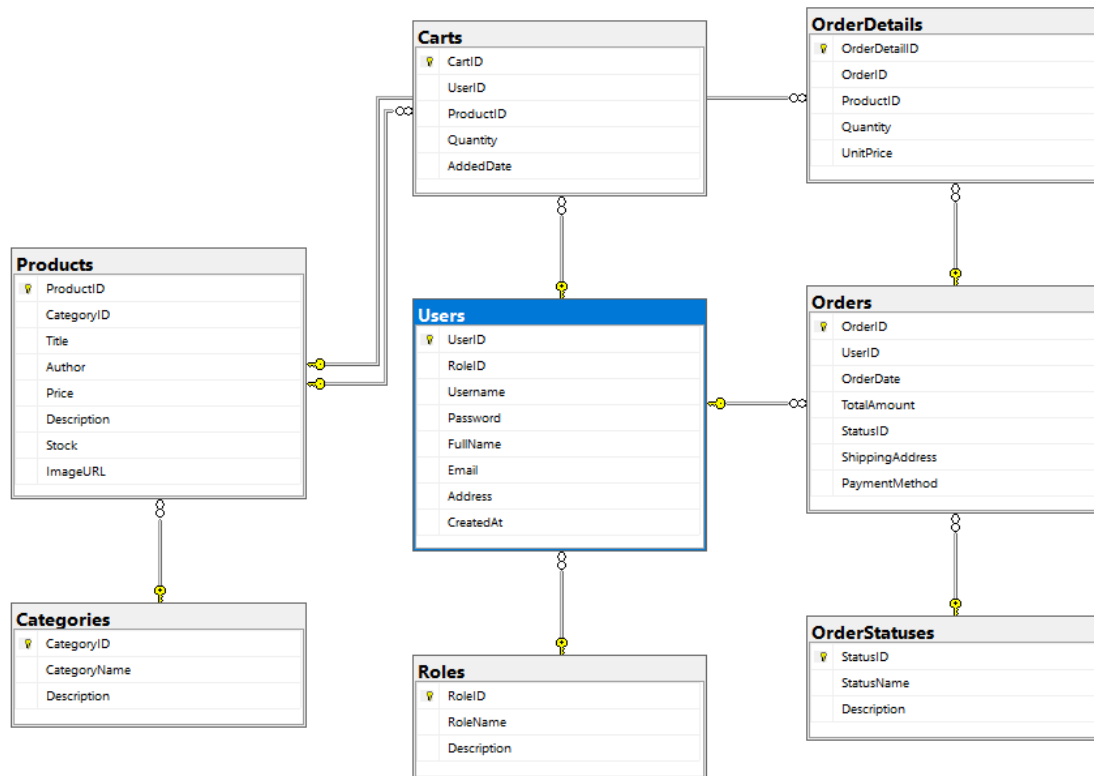
Dashboard			X	
ManageUser			X	
ManageBook				X
ManageOrder				X
ManageOrderDetail				X
ManageCategory				X

### 3.4 Non-UI Functions

#	Feature	System Function	Description
01	Authentication	User Authentication	Authenticate users when logging in, logging out, forgetting password.
02	Authorization	Role-Based Access Control	Role-based authorization (Customer, Staff, Admin) to control access.
03	Payment	Payment Processing	Process online payments via payment gateways.
04	Notification	Email Notification	Send email to confirm password change, purchase order

## 4. System High Level Design

### 4.1 Database Design



#### a. Table Descriptions

No	Table	Description
01	Users	<p><b>Description:</b> Stores information about the users of the system.</p> <p><b>Fields:</b></p> <ul style="list-style-type: none"><li>● UserID: Primary key, unique identifier for each user.</li><li>● RoleID: Foreign key, links to the Roles table to determine the user's role (e.g., admin, customer).</li><li>● Username: The user's login name.</li><li>● Password: The user's password (typically encrypted).</li><li>● FullName: The user's full name.</li><li>● Email: The user's email address.</li><li>● Address: The user's physical address.</li><li>● CreatedAt: The timestamp when the user account was created.</li></ul> <p><b>Relationships:</b></p>

		<ul style="list-style-type: none"> <li>● Linked to the Roles table via RoleID.</li> <li>● Linked to the Orders table via UserID (one user can have multiple orders).</li> <li>● Linked to the Carts table via UserID (one user can have multiple cart items).</li> </ul>
02	Roles	<p><b>Description:</b> Stores information about roles in the system (e.g., admin, customer).</p> <p><b>Fields:</b></p> <ul style="list-style-type: none"> <li>● RoleID: Primary key, unique identifier for each role.</li> <li>● RoleName: The name of the role.</li> <li>● Description: A description of the role.</li> </ul> <p><b>Relationships:</b></p> <ul style="list-style-type: none"> <li>● Linked to the Users table via RoleID (one role can be assigned to multiple users).</li> </ul>
03	Products	<p><b>Description:</b> Stores information about the products available for sale in the system.</p> <p><b>Fields:</b></p> <ul style="list-style-type: none"> <li>● ProductID: Primary key, unique identifier for each product.</li> <li>● CategoryID: Foreign key, links to the Categories table to determine the product's category.</li> <li>● Title: The title or name of the product.</li> <li>● Author: The author or brand of the product.</li> <li>● Price: The price of the product.</li> <li>● Description: A detailed description of the product.</li> <li>● Stock: The quantity of the product in stock.</li> <li>● ImageURL: The URL to the product's image.</li> </ul> <p><b>Relationships:</b></p> <ul style="list-style-type: none"> <li>● Linked to the Categories table via CategoryID (one category can contain multiple products).</li> <li>● Linked to the OrderDetails table via ProductID (one product can appear in multiple order details).</li> <li>● Linked to the Carts table via ProductID (one product can be added to multiple carts).</li> </ul>
04	Categories	<ul style="list-style-type: none"> <li>● <b>Description:</b> Stores information about product categories.</li> <li>● <b>Fields:</b> <ul style="list-style-type: none"> <li>○ CategoryID: Primary key, unique identifier for each category.</li> </ul> </li> </ul>

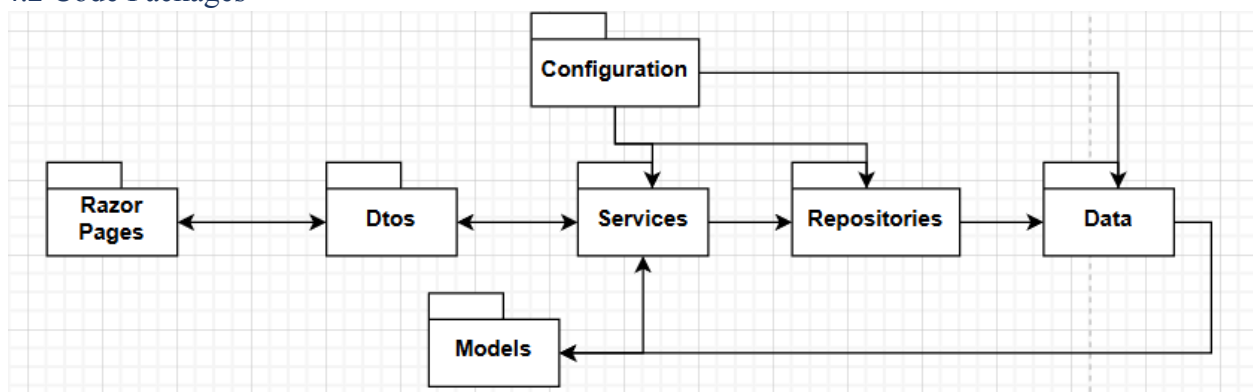
		<ul style="list-style-type: none"> <li>○ CategoryName: The name of the category.</li> <li>○ Description: A description of the category.</li> <li>● <b>Relationships:</b> <ul style="list-style-type: none"> <li>○ Linked to the Products table via CategoryID (one category can contain multiple products).</li> </ul> </li> </ul> <p><b>5. Table: Orders</b></p> <ul style="list-style-type: none"> <li>● <b>Description:</b> Stores information about orders placed by users.</li> <li>● <b>Fields:</b> <ul style="list-style-type: none"> <li>○ OrderID: Primary key, unique identifier for each order.</li> <li>○ UserID: Foreign key, links to the Users table to identify the user who placed the order.</li> <li>○ OrderDate: The date the order was placed.</li> <li>○ TotalAmount: The total amount of the order.</li> <li>○ StatusID: Foreign key, links to the OrderStatuses table to determine the order's status (e.g., delivered, processing).</li> <li>○ ShippingAddress: The shipping address for the order.</li> <li>○ PaymentMethod: The payment method used (e.g., credit card, cash).</li> </ul> </li> <li>● <b>Relationships:</b> <ul style="list-style-type: none"> <li>○ Linked to the Users table via UserID (one user can have multiple orders).</li> <li>○ Linked to the OrderStatuses table via StatusID (one status can apply to multiple orders).</li> <li>○ Linked to the OrderDetails table via OrderID (one order can contain multiple order details).</li> </ul> </li> </ul>
05	OrderDetails	<p><b>Description:</b> Stores detailed information about the products in each order.</p> <p><b>Fields:</b></p> <ul style="list-style-type: none"> <li>● OrderDetailID: Primary key, unique identifier for each order detail.</li> <li>● OrderID: Foreign key, links to the Orders table to identify the order.</li> <li>● ProductID: Foreign key, links to the Products table to identify the product.</li> <li>● Quantity: The quantity of the product in the order.</li> <li>● UnitPrice: The unit price of the product at the time of the order.</li> </ul> <p><b>Relationships:</b></p>



		<ul style="list-style-type: none"> <li>• Linked to the Orders table via OrderID (one order can have multiple details).</li> <li>• Linked to the Products table via ProductID (one product can appear in multiple order details).</li> </ul>
06	OrderStatuses	<p><b>Description:</b> Stores information about the possible statuses of an order.</p> <p><b>Fields:</b></p> <ul style="list-style-type: none"> <li>• StatusID: Primary key, unique identifier for each status.</li> <li>• StatusName: The name of the status (e.g., "Processing", "Delivered").</li> <li>• Description: A description of the status.</li> </ul> <p><b>Relationships:</b></p> <ul style="list-style-type: none"> <li>• Linked to the Orders table via StatusID (one status can apply to multiple orders).</li> </ul>
07	Carts	<p><b>Description:</b> Stores information about the products in a user's shopping cart.</p> <p><b>Fields:</b></p> <ul style="list-style-type: none"> <li>• CartID: Primary key, unique identifier for each cart item.</li> <li>• UserID: Foreign key, links to the Users table to identify the user who owns the cart.</li> <li>• ProductID: Foreign key, links to the Products table to identify the product in the cart.</li> <li>• Quantity: The quantity of the product in the cart.</li> <li>• AddedDate: The date the product was added to the cart.</li> </ul> <p><b>Relationships:</b></p> <ul style="list-style-type: none"> <li>• Linked to the Users table via UserID (one user can have multiple cart items).</li> <li>• Linked to the Products table via ProductID (one product can appear in multiple carts).</li> </ul>
08	Orders	<p><b>Description:</b> Stores information about orders placed by users.</p> <p><b>Fields:</b></p> <ul style="list-style-type: none"> <li>• OrderID: Primary key, unique identifier for each order.</li> <li>• UserID: Foreign key, links to the Users table to identify the user who placed the order.</li> <li>• OrderDate: The date the order was placed.</li> </ul>

		<ul style="list-style-type: none"> <li>• TotalAmount: The total amount of the order.</li> <li>• StatusID: Foreign key, links to the OrderStatuses table to determine the order's status (e.g., delivered, processing).</li> <li>• ShippingAddress: The shipping address for the order.</li> <li>• PaymentMethod: The payment method used (e.g., credit card, cash).</li> </ul> <p><b>Relationships:</b></p> <ul style="list-style-type: none"> <li>• Linked to the Users table via UserID (one user can have multiple orders).</li> <li>• Linked to the OrderStatuses table via StatusID (one status can apply to multiple orders).</li> <li>• Linked to the OrderDetails table via OrderID (one order can contain multiple order details).</li> </ul>
--	--	--

## 4.2 Code Packages



### Package descriptions

No	Package	Description
01	Models Package	Contains entity classes that represent the database structure in MongoDB. These classes (e.g., User, Product, Order) define the data schema and are used by the Data layer to map MongoDB collections and by the Services layer to handle business logic.
02	Dtos Package	Includes Data Transfer Objects (e.g., UserDto, ProductDto, OrderDto) used to transfer data between layers. Dtos are used by Razor Pages to send/receive data to/from Services and by Services to map data from Models, ensuring data encapsulation and security.
03	Repositories Package	Manages data access operations with MongoDB. This package includes interfaces (e.g., IProductRepository) and implementations (e.g., ProductRepository) that interact with the Data layer (MongoDbContext) to perform CRUD

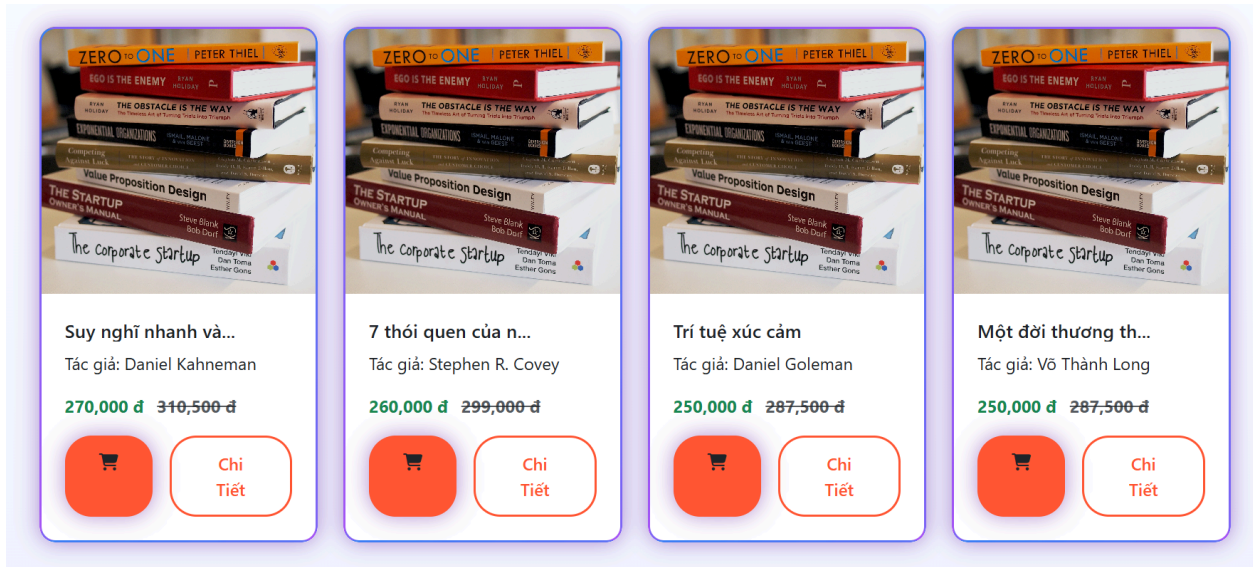
		operations on Models. Repositories are called by Services to access or manipulate data.
04	Services Package	Handles business logic and acts as an intermediary between Razor Pages and Repositories. This package includes interfaces (e.g., IProductService) and implementations (e.g., ProductService) that process requests, map Models to Dtos (and vice versa), and call Repositories for data operations.
05	Razor Pages Package	Contains the UI layer of the application, implemented as Razor Pages (e.g., Home.cshtml, Staff-ManageBook.cshtml). Razor Pages interact with Services to handle user requests, send Dtos to Services for processing, and receive Dtos to display data on the UI.
06	Data Package	Manages the connection to SQL Server and provides access to database tables. This package includes BookDbContext, which uses Models to map SQL Server tables and is used by Repositories to perform database operations. It relies on Configuration for the connection string.
07	Configuration Package	Handles application configuration and dependency injection (DI). This package includes Program.cs (for DI setup) and appsettings.json (for settings like the SQL Server connection string). It provides configuration to the Data layer and registers Services and Repositories for DI.

## II. Screen Designs

### 1. Feature User

#### 1.1 Add to Cart

##### a. Add to Cart



Field Name	Field Type	Description
ProductID	INT	ID of the product added to the cart
UserID	INT	ID of the user adding the product
Quantity	INT	Number of items added to the cart
AddedDate	DATETIME	Timestamp when the product is added

#### Database Access

Table	CRUD	Description
Cart	C	Insert product into the cart
..		

## 1.2 View Product Detail

### a. Product Detail

Hiệu Sách 4.0

Sách Mới


Bán Chạy

Giảm Giá

Tìm sách...

Q

Lê Văn Toàn



## Trí tuệ xúc cảm

Tác giả: Daniel Goleman

★★★★☆ (4.2/5 - 150 đánh giá)

**250,000 đ** ~~287,500 đ~~

**Thể loại:** Văn học

**Số trang:**

**Mô tả :** Tầm quan trọng của trí tuệ cảm xúc trong cuộc sống.

**Số lượng :** 5

-

1

+

Thêm Vào Giỏ Hàng

Mua Ngay

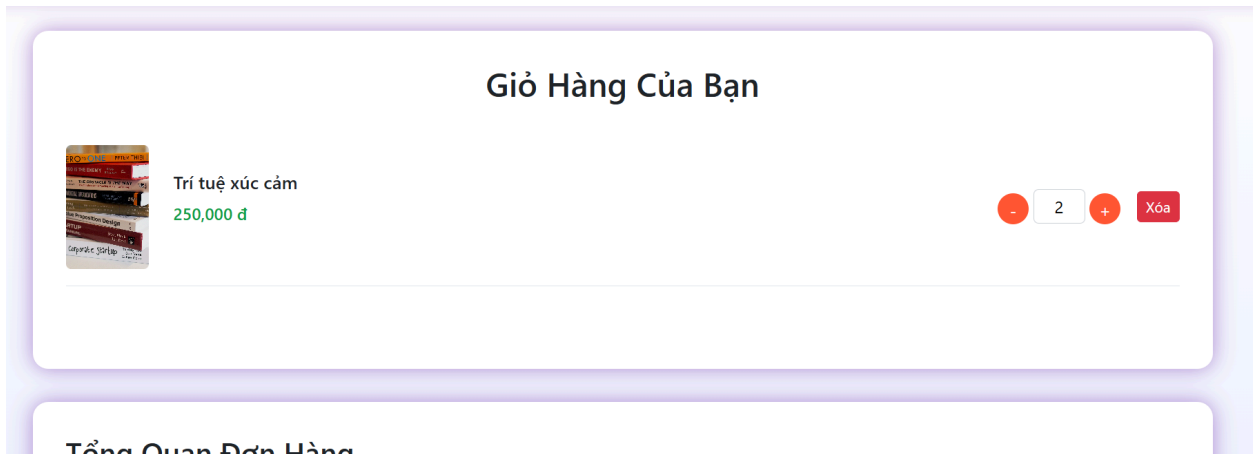
Field Name	Field Type	Description
ProductID	INT	ID of the product
Name	VARCHAR	Name Product
Price	DECIMAL	Product price
Description	TEXT	Product description
Stock	INT	Available stock quantity
ImageURL	VARCHAR	Product image URL

### Database Access

Table	CRUD	Description
Product	R	Retrieve product details from the database
..		

### 1.3 View Cart

#### a. View Cart



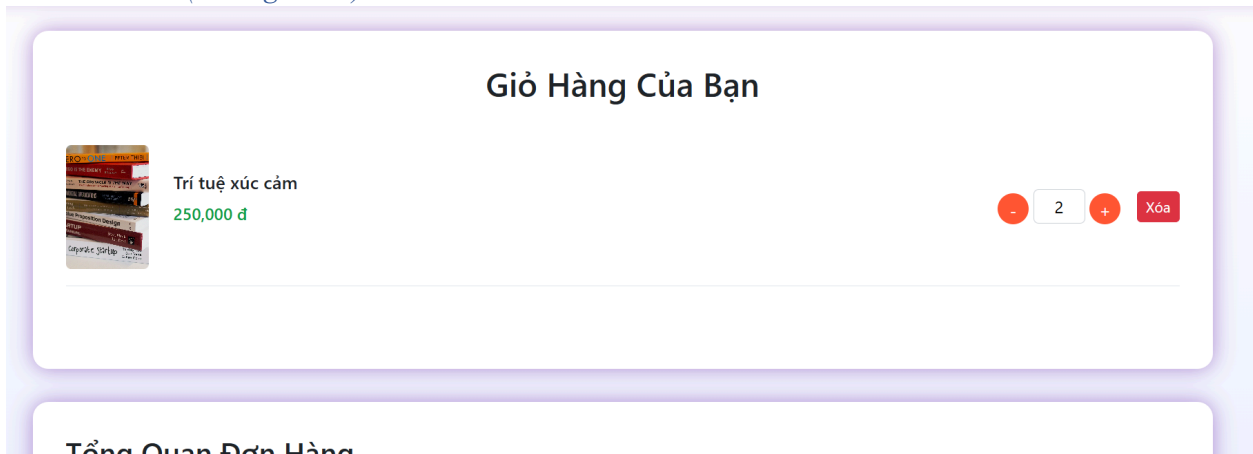
Field Name	Field Type	Description
CartID	INT	CartID
UserID	INT	UserID
ProductID	INT	Product ID in the cart
Quantity	INT	Quantity of the product in the cart
Price	DECIMAL	Price per product
TotalPrice	DECIMAL	Total cart amount

#### Database Access

Table	CRUD	Description
Cart	R	Retrieve the list of products in the cart
..		

## 1.4 CRUD Cart (Manage Cart)

### a. CRUD Cart (Manage Cart)



Field Name	Field Type	Description
CartID	INT	CartID
ProductID	INT	ProductID
Quantity	INT	Updated quantity of the product
UserID	INT	UserID

#### Database Access

Table	CRUD	Description
Cart	U	Update product quantity in the cart
Cart	D	Remove product from the cart

## 1.1 Checkout


### a. Checkout

**Tổng Quan Đơn Hàng**

Tạm tính (1 sản phẩm):  
Phí vận chuyển:  
Tổng cộng:

**Địa chỉ - Số điện thoại giao hàng:**  
Phú đồ, Nam từ liêm, Hà nội - 035572

**Liên Hệ**  
Email: contact@hieusach.com  
Phone: 0123-456-789



**Số tiền:** 500,000 đ  
**Địa chỉ giao hàng:** Phú đồ, Nam từ liêm, Hà nội - 0355725203  
**Nội dung chuyển khoản:** 10022503231705  
**Ngân hàng:** TPBank  
**Số tài khoản:** 00003480126  
**Chủ tài khoản:** Trần Đình Đức  
Thời gian còn lại: **02:55**  
Vui lòng quét mã QR hoặc chuyển khoản theo thông tin trên để

500,000 đ  
Free  
**500,000 đ**

**Dối**  
@

Field Name	Field Type	Description
OrderID	INT	Order ID
UserID	INT	UserID
CartID	INT	Cart ID
TotalPrice	DECIMAL	Total order amount
PaymentMethod	VARCHAR	Payment method
OrderStatus	VARCHAR	Order status (Pending, Completed, Canceled)
CreatedDate	DATETIME	Order creation date

#### Database Access

Table	CRUD	Description
Cart	D	Delete cart after successful checkout
Orders	C	Create a new order from the cart

## 2.1 System Access

### a. Register

This screen allows user to be authenticated to the system screens/functionalities.



## Create an account

Full name

John Doe

Email

example.email@gmail.com

Password

Enter at least 8+ characters



Re-type Password

Enter at least 8+ characters



[Forgot Password ?](#)

Sign up

Field Name	Field Type	Description
Email*	String(30)	This is for user to input valid email address for logging in
Password*	String(8-20)	This is for user to input password for logging in
Login	Button	User clicks to authenticate him/herself into the system with provided email & password
Register	Button	User clicks to redirect to the User Register page for registering new user account to access the system
Forgot Password?	Hyperlink	User clicks to redirect to the Password Reset page for resetting his/her forgot password
Login with Google	Hyperlink	Allow user to login with his/her Google account
Login with Facebook	Hyperlink	Allow user to login with his/her Facebook account

### Database Access

Table	CRUD	Description
User	R	Verify UserName & Password information
Setting, User	R	Specify the authorizations of the logged-in user

## 2.2 Book Management

### a. Book Management

Quản Lý Sách

Staff

Tìm kiếm sách...

Tìm

+ Thêm Sách

ID	Tên Sách	Tác Giả	Giá	Số Lượng	Hành Động
50	AI trong y t?	Đ?ng Th? E	390.00	8	Sửa
49	Docker và CI/CD	Ph?m Đình D	320.00	16	Sửa
48	Phát tri?n game v?i Unreal Engine	Lê Van C	340.00	28	Sửa
47	Phát tri?n game v?i Unity	Tr?n Th? B	280.00	35	Sửa
46	Xây d?ng h? th?ng phân tán	Nguy?n Van A	370.00	10	Sửa

1

2

3

4

5

6

7

8

9

10

Field Name	Field Type	Description
BookID	INT	Unique identifier for each book
Title	VARCHAR	Book title
Author	VARCHAR	Book author
Price	DECIMAL	Book price
Quantity	INT	Number of books in stock
Actions	BUTTON	Edit, Delete

#### Database Access

Table	CRUD	Description
Books	R	Retrieve all books for management
Books	U	Update book details
Books	D	Delete a book from the system

## 2.3 Category Management

### a. Category Management

Quản Lý Danh Mục

Staff

Tìm kiếm danh mục...

+

Thêm Danh Mục

ID	Tên Danh Mục	Mô Tả	Hành Động
1	Văn học	Các tác phẩm văn học kinh điển và hiện đại.	Sửa
2	Kinh doanh	Sách về kinh tế, tài chính và phát triển bản thân.	Sửa
3	Kỹ năng sống	Sách hướng dẫn phát triển kỹ năng cá nhân và tư duy.	Sửa
4	Khoa học & Công nghệ	Sách về khoa học, vũ trụ và công nghệ.	Sửa
5	Tâm lý & Triết học	Sách về tâm lý học, triết học và hành vi con người.	Sửa

Field Name	Field Type	Description
CategoryID	INT	Unique identifier for each category
Name	VARCHAR	Name of the category
Description	VARCHAR	Description
Actions	Button	Edit, Delete

### Database Access

Table	CRUD	Description
Categories	R	Retrieve all book categories
Categories	U	Update category details
Categories	D	Delete a category

## 2.4 Order Management

### a. Order Management

Quản Lý Đơn Hàng

Staff

Tìm kiếm đơn hàng...

Tất cả trạng thái

Tìm

Mã Đơn	Tổng Tiền	Ngày Đặt	Trạng Thái	Hành Động
1003	530000.00	3/17/2025 10:55:47 PM	Đặt hàng	Chi Tiết
1002	270350.00	3/17/2025 2:33:12 PM	Đặt hàng	Chi Tiết
2	1620000.00	3/12/2025 8:28:49 AM	Đặt hàng	Chi Tiết

Trước

1

2

3

Sau

Field Name	Field Type	Description
OrderID	INT	Unique identifier for each order
UserID	INT	ID of the customer who placed the order
TotalPrice	DECIMAL	Total order amount
Status	INT	Order status (Pending, Shipped, Completed, Canceled)
Actions	BUTTON	View, Update Status

#### Database Access

Table	CRUD	Description
Orders	R	Retrieve all orders
Orders	D	Update order status