

VIETNAM NATIONAL UNIVERSITY - HO CHI MINH CITY
INTERNATIONAL UNIVERSITY



WEB APPLICATION DEVELOPMENT THE ONLINE BOOKSTORE PROJECT

by.
Phan Quốc Thắng - ITCSIU21232
Đỗ Đình Phúc - ITCSIU21219

Advisors: Dr. Nguyen Van Sinh & Mr. Nguyen Trung Nghia

/ This report is submitted to the School of Computer Science and Engineering in satisfaction of the final project requirement for the Web Application Development course taken in Spring 2024. */*

Ho Chi Minh City, Vietnam

I. INTRODUCTION.....	4
a. About Us.....	4
b. About the Project.....	4
c. Development Process.....	4
d. Development Environment.....	5
e. Development Architecture.....	6
II. REQUIREMENT ANALYSIS & DESIGN.....	8
a. Requirement Analysis.....	8
Use Cases.....	8
Functional / Non-functional Requirements.....	8
b. Design.....	10
Database Showcase.....	10
Class Diagram.....	11
Sequence Diagram.....	21
III. IMPLEMENTATION.....	30
A. User.....	30
I. Account Management.....	30
Home.....	30
The Header.....	32
Login.....	32
Register / Sign Up.....	35
II. Browse Books.....	40
All Books.....	40
Book Card.....	43
Book Details.....	44
Comments.....	45
III. Cart.....	45
B. Admin.....	46
I. Admin Homepage.....	46
Books Management.....	46
Authors Management.....	49
Publishers Management.....	50
Comments Management.....	50
Carts Management.....	51
Orders Management.....	52
IV. DISCUSSION & CONCLUSION.....	52
1. Limitation.....	52
2. Future Improvement.....	52
V. REFERENCES.....	53
VI. CONTRIBUTION.....	53

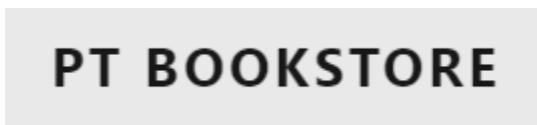
I. INTRODUCTION

a. About Us

The online bookstore web application is the ideal solution for Phuc and Thang to kickstart their vision of creating an innovative virtual book emporium. By harnessing the power of this modern platform, they can offer a diverse array of books to cater to various readers' preferences while providing a seamless and engaging shopping experience.

b. About the Project

PT Online Bookstore was created to foster a love of reading by providing an easy and enjoyable platform for browsing and purchasing publications. Our website caters to readers of all ages, offering a vast selection of books, magazines, and much more. Users can explore a curated collection and discover their next great read, all from the comfort of their homes. PT Online Bookstore strives to make the entire process, from browsing to giving reviews, a seamless and joyful experience which helps users discover publications that align with their interests. By providing a user-friendly platform and a curated selection of products, PT Online Bookstore aims to become the go-to destination for readers to explore and discover their next literary adventure.



PT BOOKSTORE

Figure 1.1. "PT BOOKSTORE" banner

c. Development Process

We decided to adopt **Agile** as our development methodology. This involved dividing the project into small, iterative "sprints." During each sprint, we continuously built and tested functionalities, as well as allowing for adjustments as development progressed. The looping process ensured a focus on delivering working features throughout the project lifecycle.

The following image illustrates an overview of the Agile method:

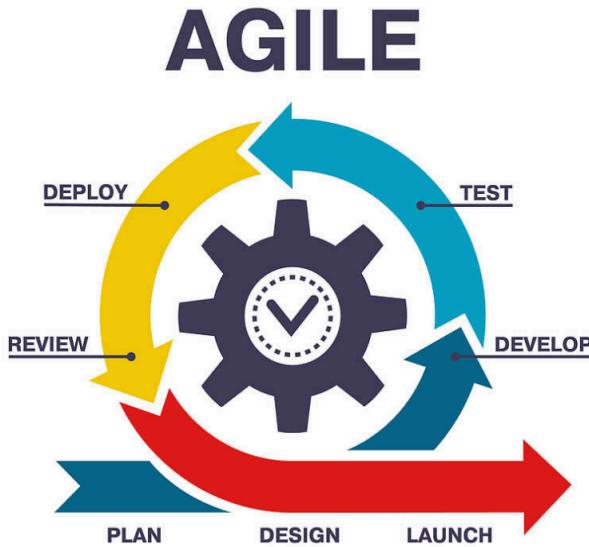


Figure 1.2. Agile Software Development life cycle

d. Development Environment

This is a Web application built to implement specific types of technologies and programming languages, based on the **MVC** (Model-View-Controller) architectural pattern. Each element handles their particular tasks:

- The UI (User Interface, including the visual look & feel of the page) is created by **HTML**, **CSS** (with the help of a front-end framework called **Bootstrap**) & **JavaScript**.
- Along with the above front-end handling elements, the *View*'s overall structure is mainly formed by **Java Server Page** (or **JSP**), which provides support in displaying data and adding interactivity.
- **Servlet** and some supporting libraries help handle server-side processes, work with user requests, and generate appropriate dynamic responses.

- To work with data, a database manager became an essential part, and **MySQL** was chosen for the job, which took part in storing data, relations between objects, and much more.
- **Git & GitHub** are used to control and track the progress.

Aside from the main tools that were discussed, additionally third-party applications and websites are brought into use to work with diagrams and workflow.

e. Development Architecture

In this project, the MVC architecture is used to handle the web server:

- The Model is represented for an object that acts as a place for storing all related data of a particular entity.
- The view is simply the interface that represents to the user.
- Controller is like the central nervous system that directs all the requests or activities between the View and the Model.

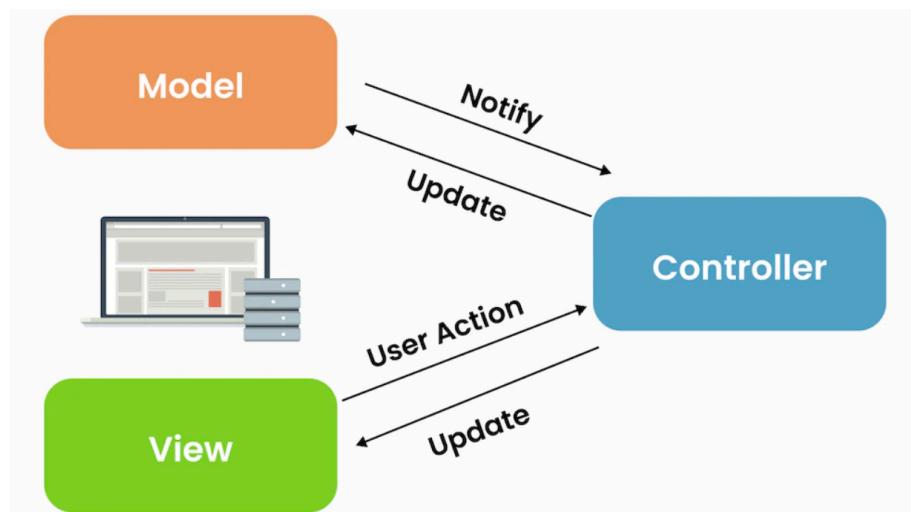


Figure 1.3. MVC Architecture

To connect to the database, the 3-tier architecture is applied for the MVC model:

- In the Presentation layer, all the requests to the lower level are performed by the Controller.
- The Business Logic layer plays an important role to handle, normalize all the data before transferring to the Data Access layer.

- Finally, the Data Access layer is the tier that directly connects to the database, executes the queries and commits the execution.

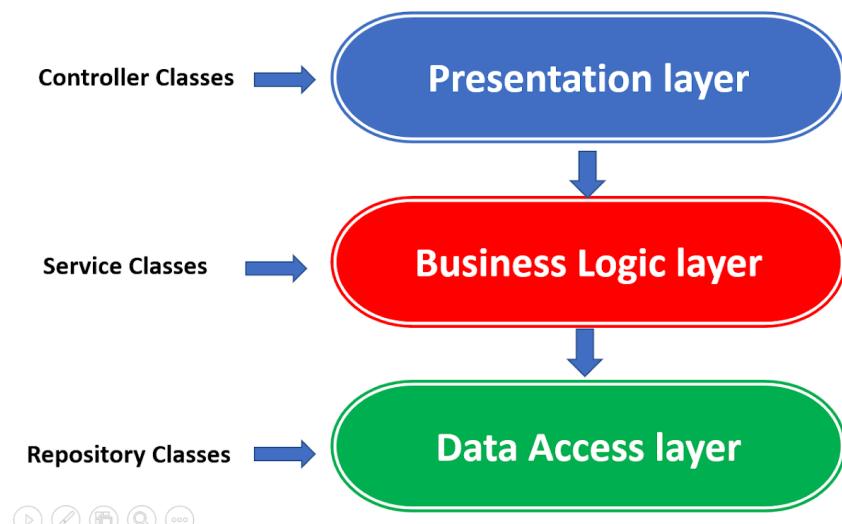


Figure 1.4. 3-tier Architecture

II. REQUIREMENT ANALYSIS & DESIGN

a. Requirement Analysis

Use Cases

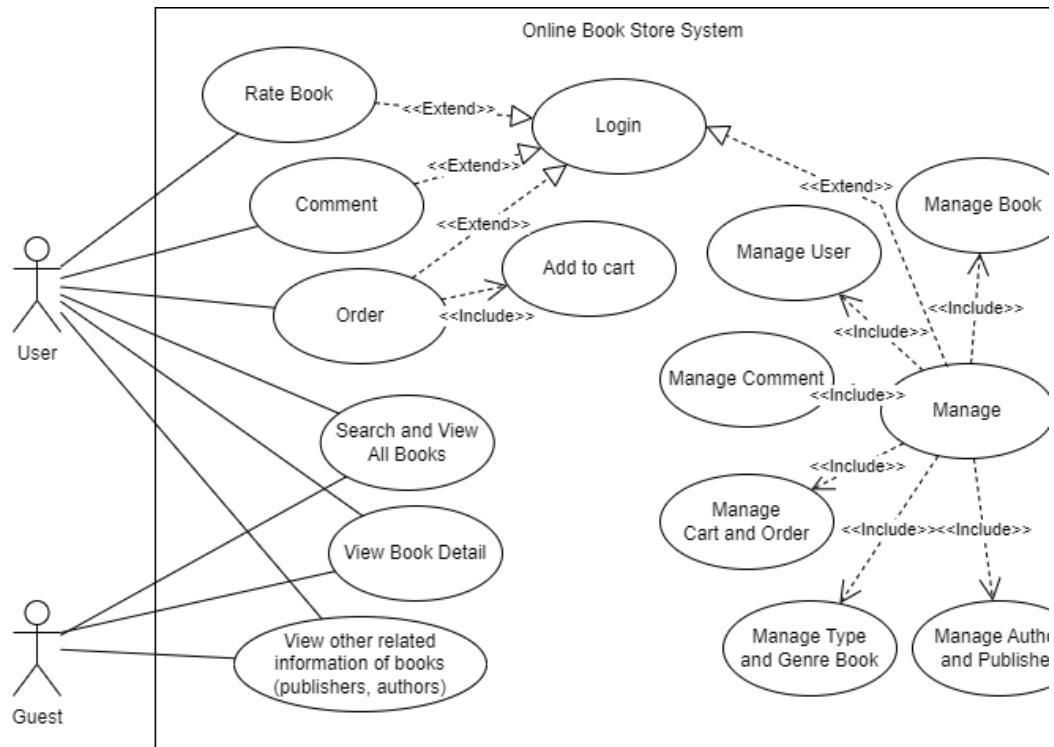


Figure 2.1. Use Case Diagram

Functional / Non-functional Requirements.

- *Functional requirements:*

1. The system must allow users to view a catalog of available books.
2. Users should be able to search for books based on title, author, genre, or keywords.
3. The system must provide detailed book descriptions including author, price, and availability.
4. Users should have the ability to add books to a shopping cart for purchase.

5. The system must allow registered users to create accounts and login with authentication and authorization.
6. Users should be able to rate, review and comment books on the website.
7. Users should be able to view their order history.
8. Users should be able to review and check the cart.
9. The website must allow users to manage their account settings, including password changes and email notifications.
10. The admin has full authorizations of reading and modifying all elements relating to the book store (book, user, comment, cart, order, type, genre, ...).

- *Non-functional requirements:*

1. The website should load within 3 seconds to provide a seamless user experience.
2. The system must handle a larger number of simultaneous users without performance degradation.
3. Regular backups of the website data must be performed to ensure data integrity and availability.
4. It must be designed to be scalable to handle increasing traffic and data storage demands.
5. The website should be designed with maintainability in mind, allowing for easy updates, modifications, and fixes.
6. The platform should provide a responsive and user-friendly interface, ensuring high usability for customers.
7. Security is vital, and the website must protect user data and transactions with robust cybersecurity measures.
8. The site should be performance-optimized to ensure quick response times and efficient resource consumption.

b. Design

Database Showcase

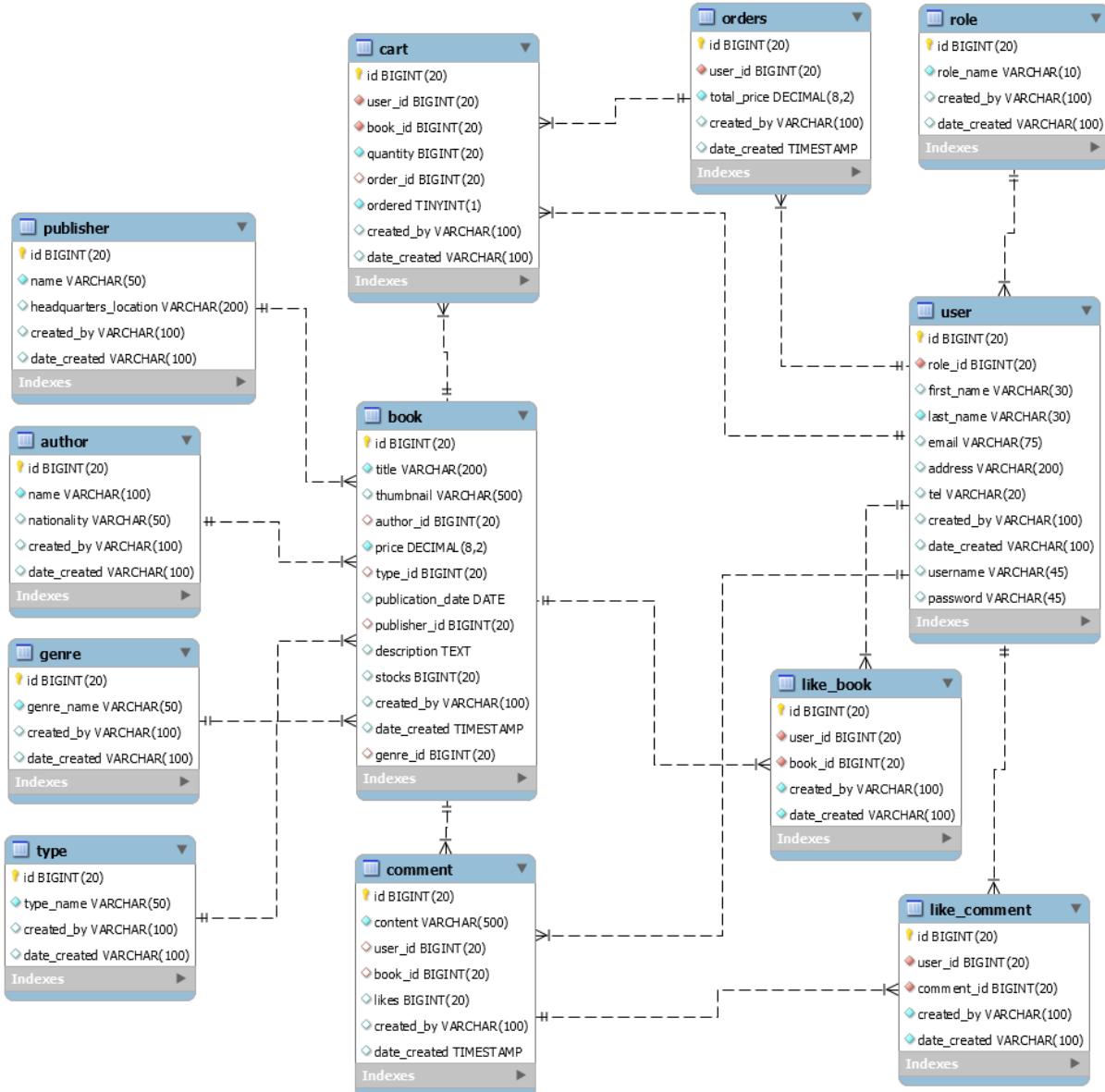


Figure 2.2. online_bookstore Database

Class Diagram

Package diagram and corresponding class diagrams:

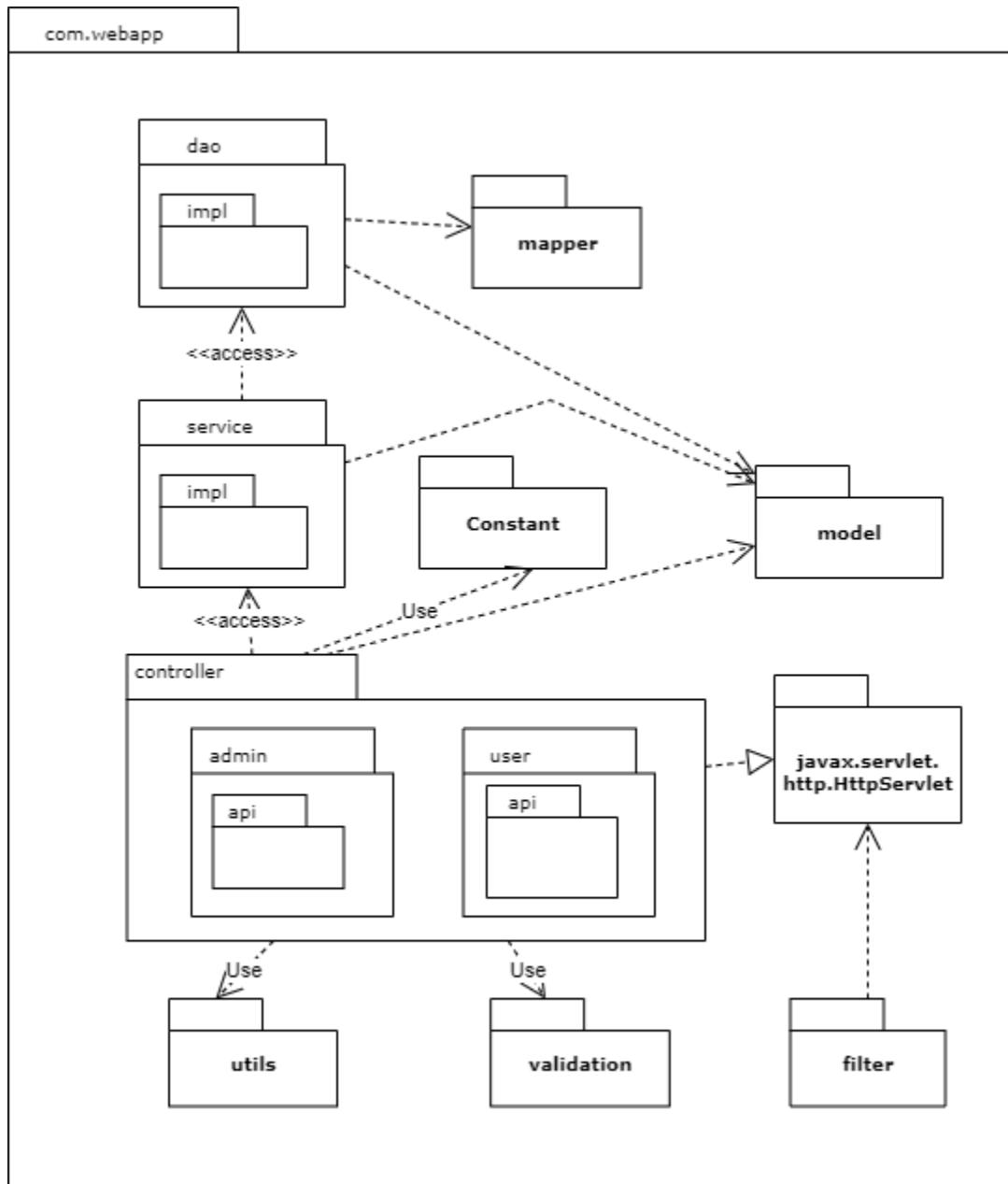


Figure 2.3. Class Diagram: Package Diagram



Figure 2.4. Class Diagram: Controller

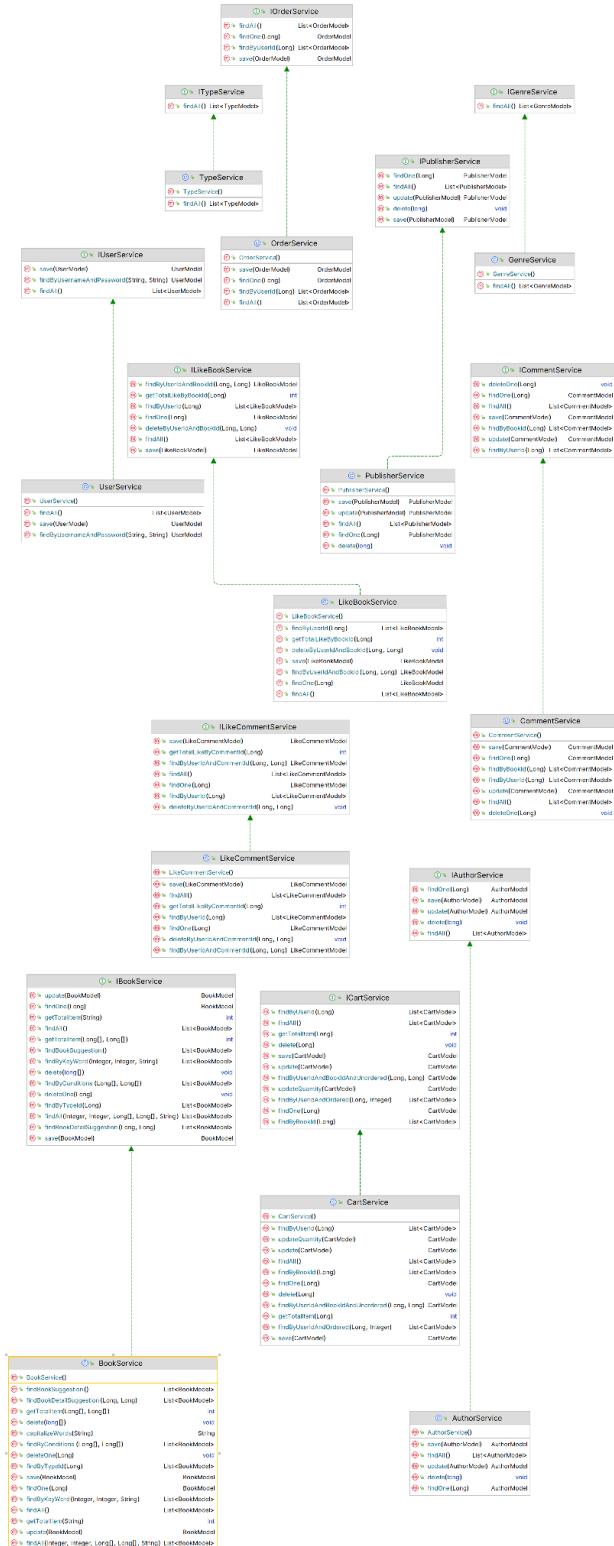


Figure 2.5. Class Diagram: Service

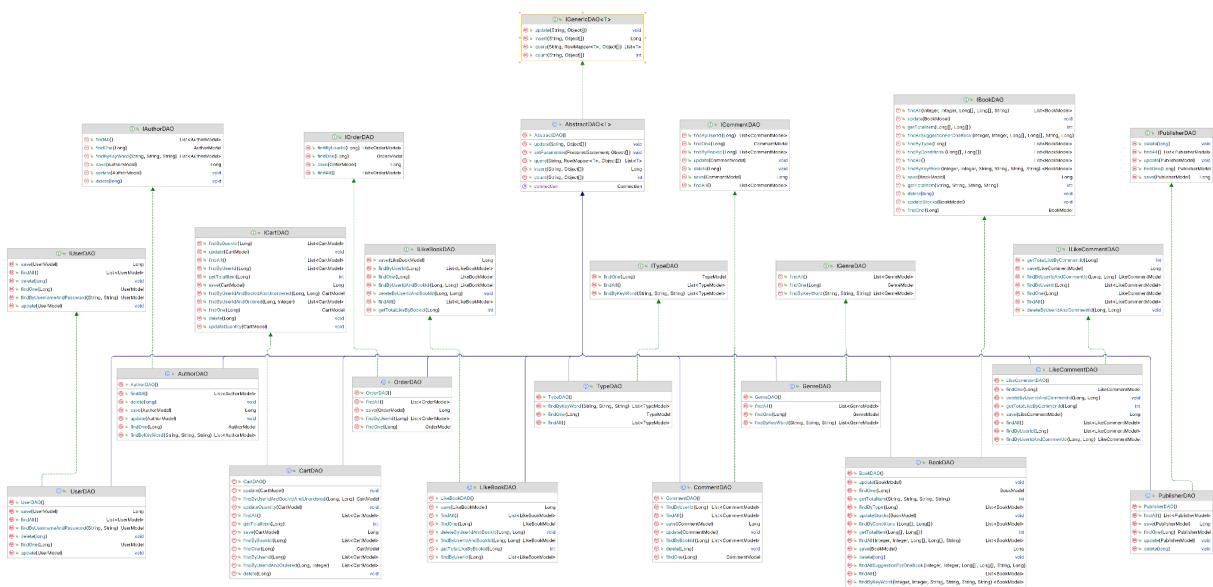


Figure 2.6. Class Diagram: DAO

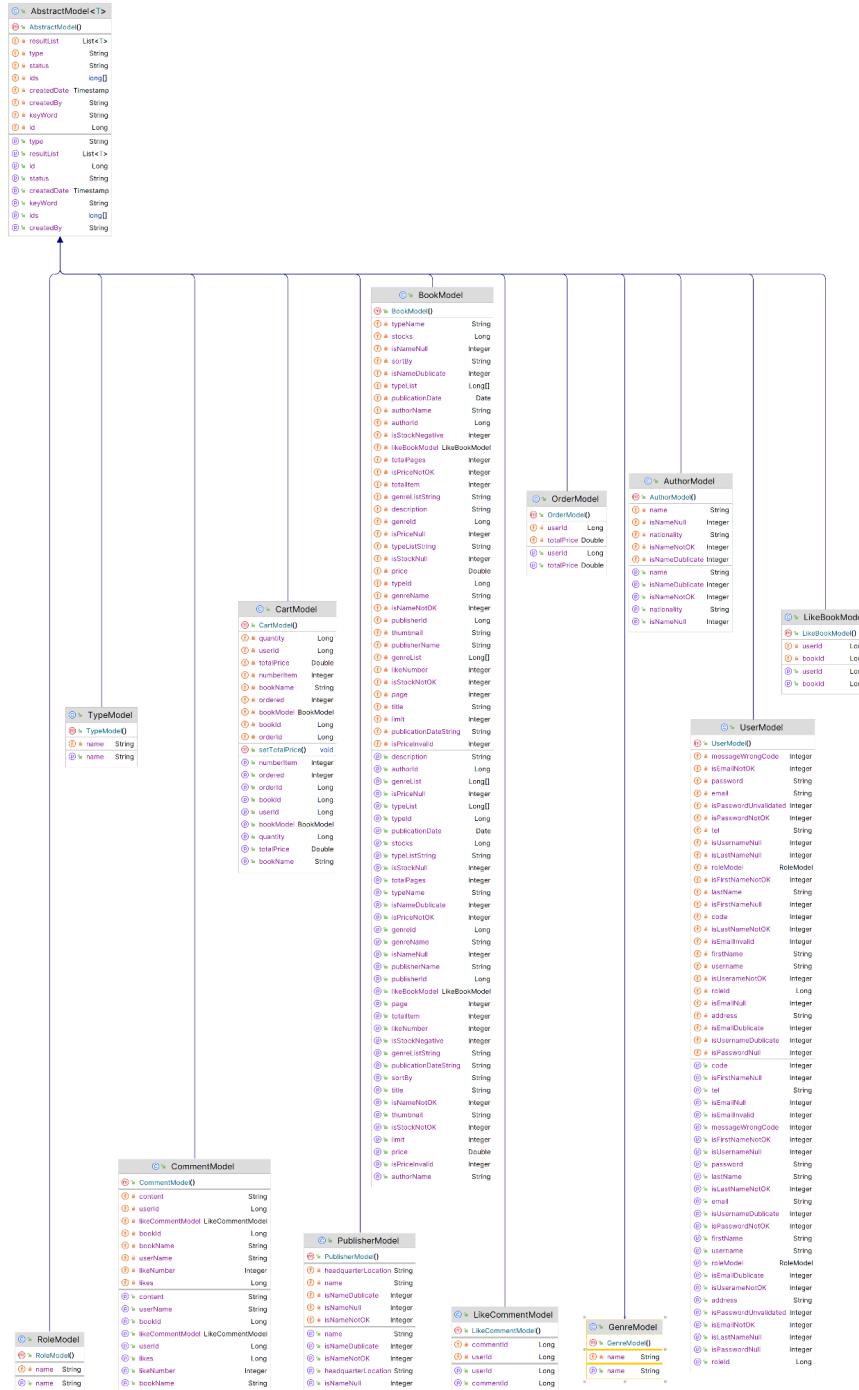


Figure 2.7. Class Diagram: Model

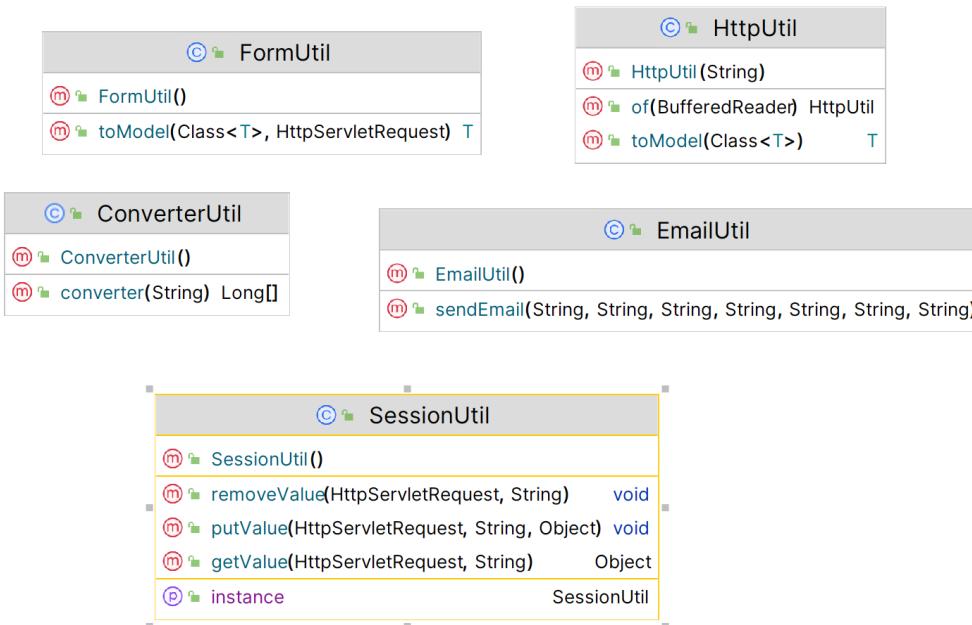


Figure 2.8. Class Diagram: Utils

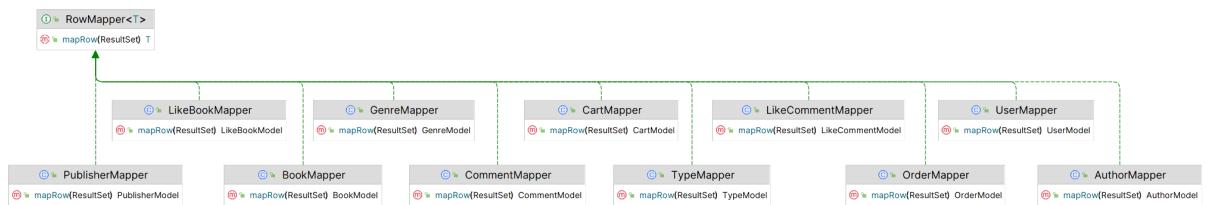


Figure 2.1. Class Diagram: Mappers

Class Explanation:

+ **Authorization:** the permission to access the url start with the “/webapp-javaweb/admin” is only for the role of admin, other accesses will be denied. All the user can access to the URLs which are not begin with “/webapp-javaweb/admin”.

```

public class Authorization implements Filter{
    private ServletContext context;
    @Override
    public void init(FilterConfig filterConfig) throws ServletException {
        this.context = filterConfig.getServletContext();
    }
    @Override
    public void doFilter(ServletRequest servletRequest, ServletResponse servletResponse, FilterChain chain)
            throws IOException, ServletException {
        HttpServletRequest request = (HttpServletRequest) servletRequest;
        HttpServletResponse response = (HttpServletResponse) servletResponse;
        String url = request.getRequestURI();
        //System.out.println(url);
        if (url.startsWith("/webapp-javaweb/admin")) {
            UserModel userModel = (UserModel) SessionUtil.getInstance().getValue(request, "USERMODEL");
            if (userModel!=null) {
                if (userModel.getRoleModel().getName().equals(SystemConstant.ADMIN)) {
                    chain.doFilter(servletRequest, servletResponse);
                } else {
                    response.sendRedirect(request.getContextPath() + "/login?action=login&message=not_permission&alert=danger");
                }
            } else {
                response.sendRedirect(request.getContextPath() + "/login?action=login&message=not_login&alert=danger");
            }
        } else {
            chain.doFilter(servletRequest, servletResponse);
        }
    }
}

```

Figure 2.9. Class Explanation: Authorization

- + **Sign up:** The user firstly goes to the main sign-up page to fill in the personal information. After filling the form, the server will check whether the information is valid or not; if the information is not valid, the user will be back to the sign-up page with the alert message to refill the form, otherwise, the server will send the user an OTP code (generated randomly by the server) via the email and lead the user to the OTP page; if there is any error in the sending email step, the user also turn back to the sign-up page. The OTP page only sends the API to create a new user in the database only when the user inputs the correct OTP. In the step the user can ask to resend the OTP message.

```

 SignUpController.java

38     if (userModel.getType().equals(SystemConstant.DISPLAY)) {
39         view = "/views/signup/signup.jsp";
40     } else if (userModel.getType().equals(SystemConstant.VALIDATE)) {
41         if (Validation.userValidate(userModel)) {
42             Random rand = new Random();
43             Integer code = 100000 + rand.nextInt(899999);
44             SessionUtil.getInstance().putValue(request, "CODE", code);
45             SessionUtil.getInstance().putValue(request, "USERSIGNUP", userModel);
46             String host = ResourceBundle.getString("host");
47             String port = ResourceBundle.getString("port");
48             String user = ResourceBundle.getString("user");
49             String pass = ResourceBundle.getString("pass");
50             String subject = ResourceBundle.getString("subject");

51             String recipient = userModel.getEmail();
52             String content = "Input the following code to sign up: \n" + code.toString() ;
53             try {
54                 EmailUtil.sendEmail(host, port, user, pass, recipient, subject, content);
55                 view = "/views/signup/sendemail.jsp";
56             } catch (Exception e) {
57                 userModel.setIsEmailNotOK(1);
58                 userModel.setIsEmailInvalid(1);
59                 view = "/views/signup/signup.jsp";
60             }
61         } else {
62             view = "/views/signup/signup.jsp";
63         }
64     } else if (userModel.getType().equals(SystemConstant.CHECKEMAIL)){
65         if (((Integer) SessionUtil.getInstance().getValue(request, "CODE")).equals(userModel.getCode())){
66             userModel = (UserModel) SessionUtil.getInstance().getValue(request, "USERSIGNUP");
67             view = "/views/signup/signupAPI.jsp";
68         } else {
69             userModel.setMessageWrongCode(1);
70             view = "/views/signup/sendemail.jsp";
71         }
72     } else {
73         view = "/views/signup/signupsuccess.jsp";
74     }
75 }
```

Figure 2.10. Class Explanation: Sign Up

- + **Login:** The login page requests to the page containing the code below. After getting the information from the login page, the server will check whether the account exists in the system or not. If there is no such account in the system, a message of “wrong username or password” will be sent to the user. If the account is valid, the server will save the status of the user by the command “SessionUtil.getInstance().putValue()”. If the user is admin, the controller will redirect the page to the admin-home page, otherwise, the user home page will be displayed.

```

    protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
        String action = request.getParameter("action");
        if (action!= null && action.equals("login")) {
            UserModel userModel = FormUtil.toModel(UserModel.class, request);
            userModel = userService.findByUsernameAndPassword(userModel.getUsername(), userModel.getPassword());
            if (userModel != null) {
                SessionUtil.getInstance().putValue(request, "USERMODEL", userModel);
                Integer cartNumber = new Integer(cartService.getTotalItem(((UserModel) SessionUtil.getInstance().getValue(request, "USERMODEL")).getId()));
                SessionUtil.getInstance().putValue(request, "CARTNUMBER", cartNumber);
                if (userModel.getRoleModel().getName().equals("User")) {
                    response.sendRedirect(request.getContextPath() + "/user-home");
                } else if (userModel.getRoleModel().getName().equals("Admin")){
                    response.sendRedirect(request.getContextPath() + "/admin-home");
                }
            } else {
                response.sendRedirect(request.getContextPath() + "/login?action=login&message=login_invalid&alert=danger");
            }
        }
    }
}

```

Figure 2.11. Class Explanation: Log In

- + **Access to database:** A big problem of querying the data is “How can we handle a huge number of queries to the database? Everywhen we want to query something in the database, we must go through the steps including initial connection, write the code of query data and bind data, ...”. We can not do such steps for a hundred queries we use in our web application. Java has a powerful tool that is ‘Generic type’ based on ‘Run time binding’. In the following piece of code, the method allows a string, a generic type called row mapper, and ‘some or no’ remain parameter(s).

```

@Override
public List<T> query(String sql, RowMapper<T> rowMapper, Object... objects) {
    List<T> results = new ArrayList<T>();
    Connection connection = null;
    PreparedStatement preparedStatement = null;
    ResultSet resultSet = null;
    try {
        connection = getConnection();
        preparedStatement = connection.prepareStatement(sql);
        setParameters(preparedStatement, objects);
        resultSet = preparedStatement.executeQuery();
        while (resultSet.next()) {
            results.add(rowMapper.mapRow(resultSet));
        }
        return results;
    } catch (SQLException e) {
        return null;
    } finally {
        try {
            if (connection != null) {
                connection.close();
            }
            if (preparedStatement != null) {
                preparedStatement.close();
            }
            if (resultSet != null) {
                resultSet.close();
            }
        } catch (SQLException e) {
            return null;
        }
    }
}

```

Figure 2.12. Class Explanation: Access To Database

All the classes extended from this code just need some lines to query the data! That is so convenient!

```

@Override
public List<BookModel> findByType(Long id) {
    String sql = "SELECT * FROM book WHERE type_id = ?";
    return query(sql, new BookMapper(), id);
}

```

Figure 2.13. Class Explanation: Access To Database

- + **Validation:** The following code is the sample code of validating books when updating or creating books. For example in this code, the title is verified by checking whether the title is blank or null and whether there is any same title in the system.

A false result will be returned if there is any condition not satisfying.

```

public static Boolean bookValidate(BookModel bookModel) {
    Boolean result = true;

    if (bookModel.getTitle() == null || bookModel.getTitle().equals("")) {
        bookModel.setIsNameNull(1); result = false;
    } else {
        for (BookModel book:bookModel.getResultList()) {
            if (book.getTitle().equals(bookModel.getTitle()) && !bookModel.getId().equals(book.getId())) {
                bookModel.setIsNameDuplicate(1); result = false; break;
            }
        }
    }
    if (bookModel.getIsNameDuplicate()==1 || bookModel.getIsNameNull()==1) bookModel.setIsNameNotOK(1);

    if (bookModel.getStocks() == null) {
        bookModel.setIsStockNull(1); result = false;
    } else {
        if (bookModel.getStocks()<0) {bookModel.setIsStockNegative(1); result = false;}
    }
    if (bookModel.getIsStockNegative()==1 || bookModel.getIsStockNull()==1) bookModel.setIsStockNotOK(1);

    if (bookModel.getPrice() == null) {
        bookModel.setIsPriceNull(1); result = false;
    } else {
        if (bookModel.getPrice()<=0) {bookModel.setIsPriceInvalid(1); result = false;}
    }
    if (bookModel.getIsPriceNull()==1 || bookModel.getIsPriceInvalid()==1) bookModel.setIsPriceNotOK(1);

    return result;
}

```

Figure 2.14. Class Explanation: Validation

Sequence Diagram

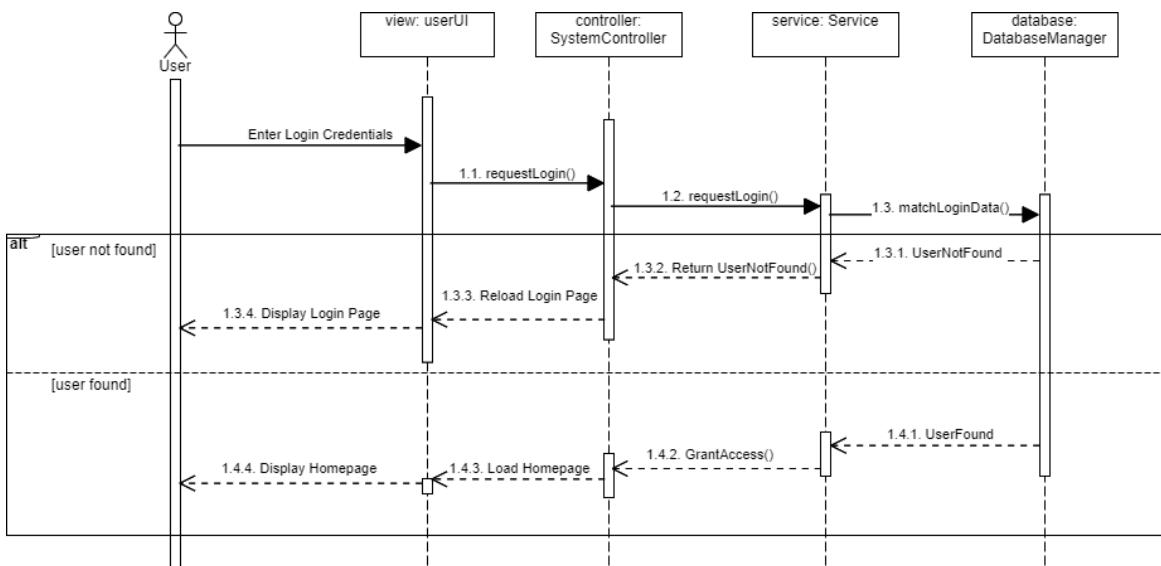


Figure. SQ - Login

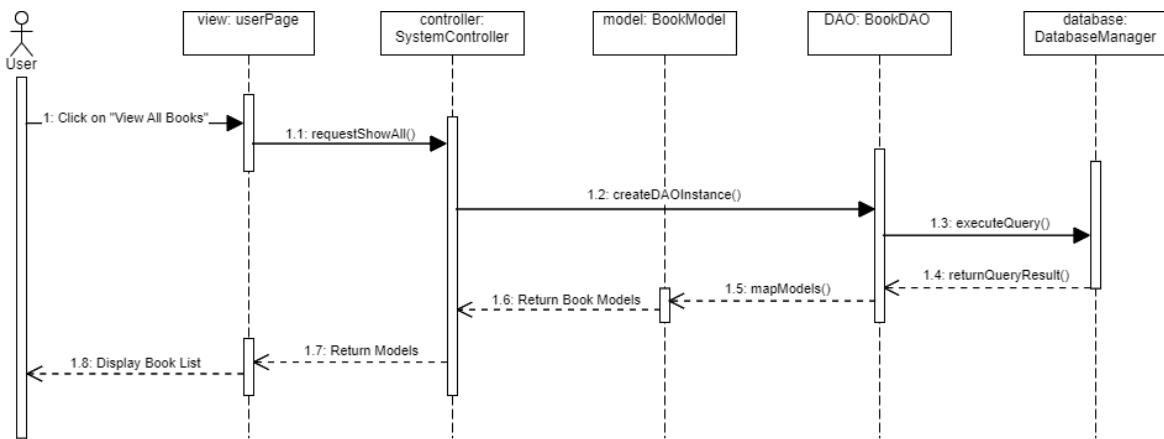


Figure. SQ - View All Books

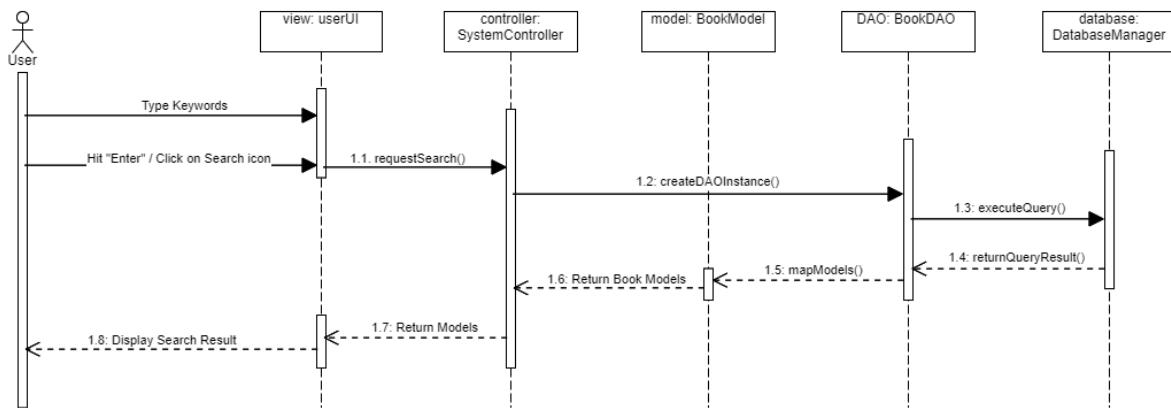


Figure. SQ - Search Books

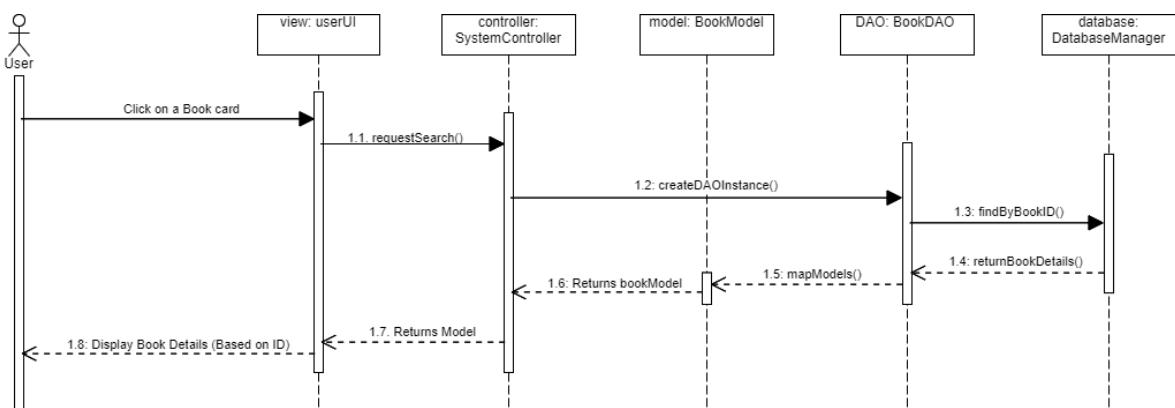


Figure. SQ - View Book Details

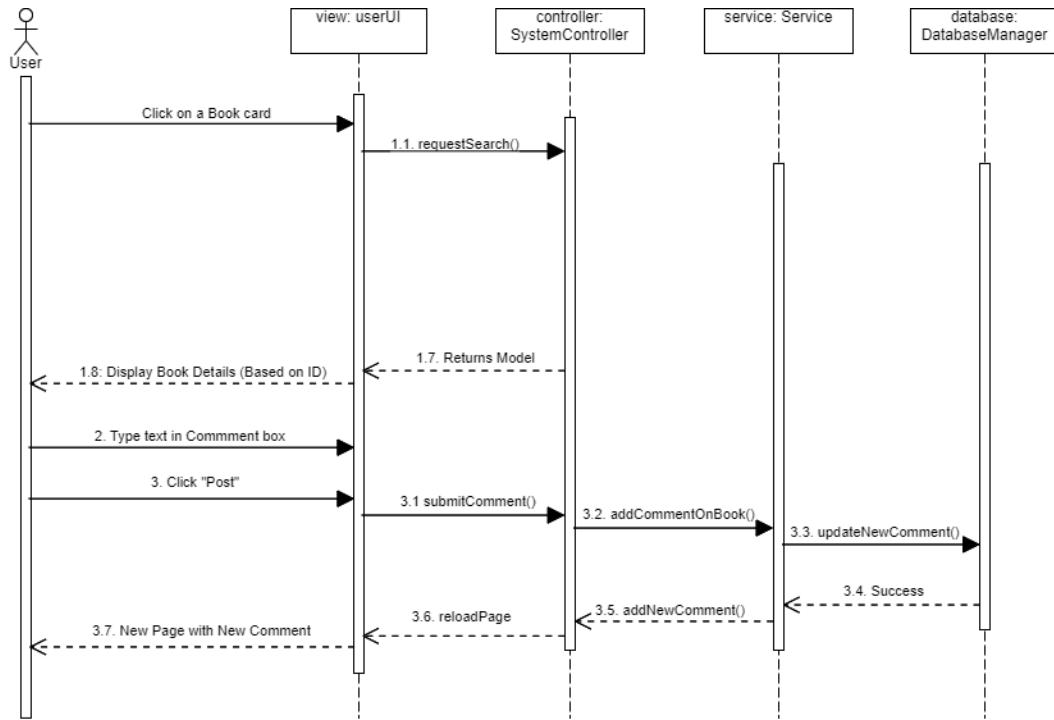


Figure. SQ - Comment On Books

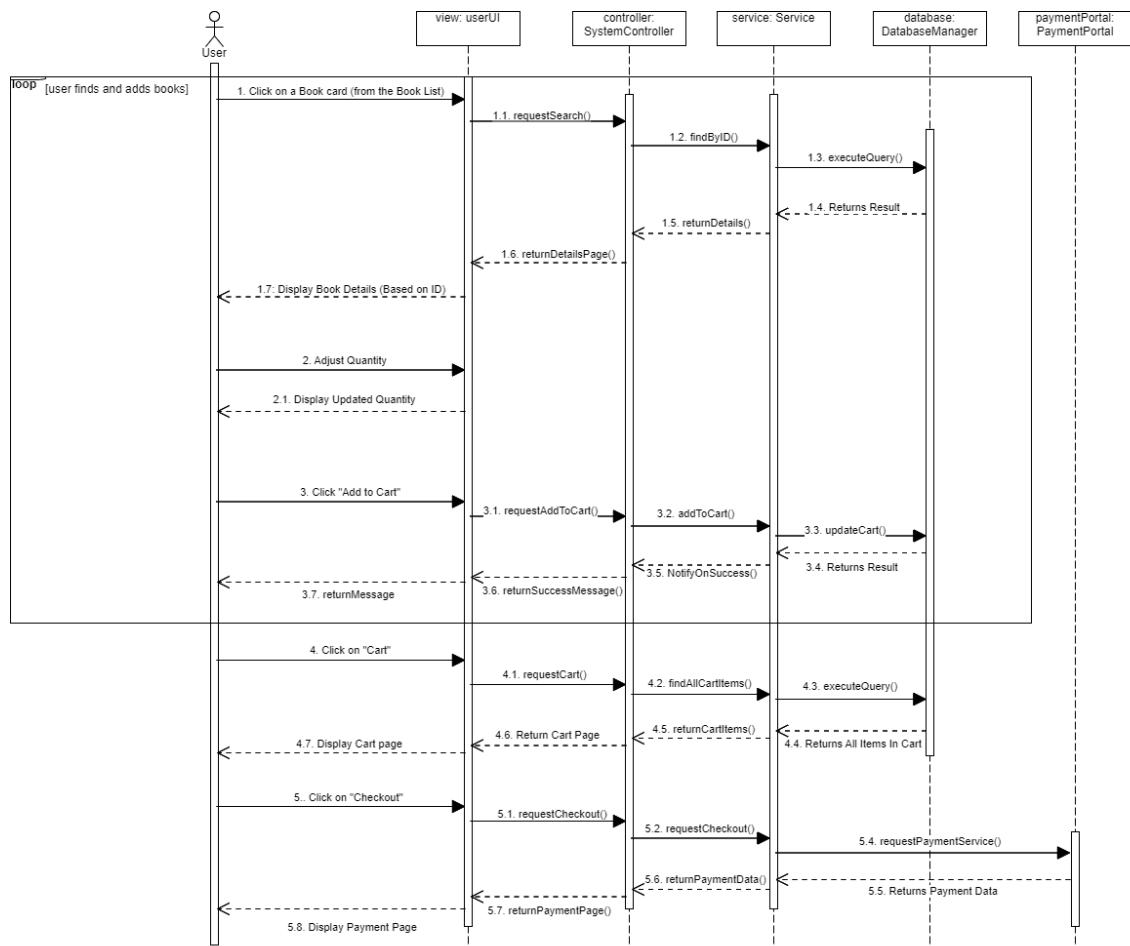


Figure. SQ - Order Books

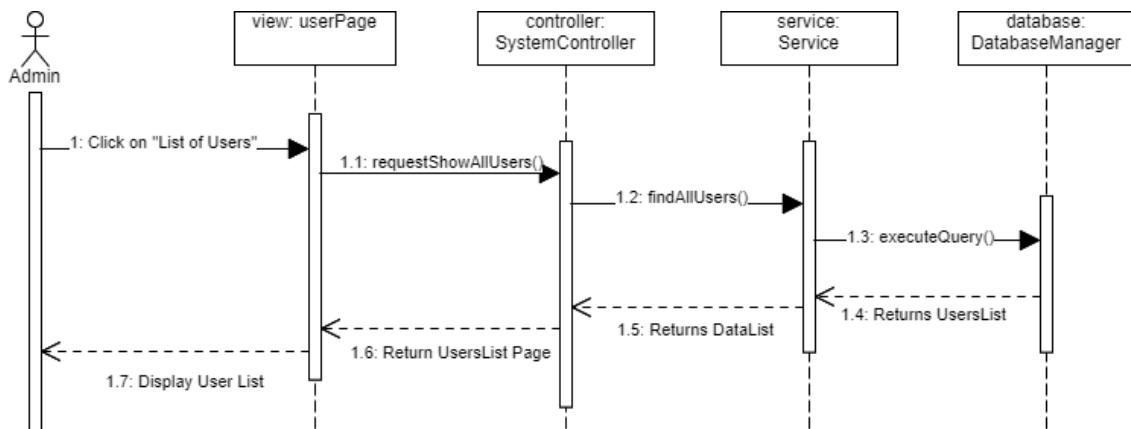


Figure. SQ - Admin: View Users

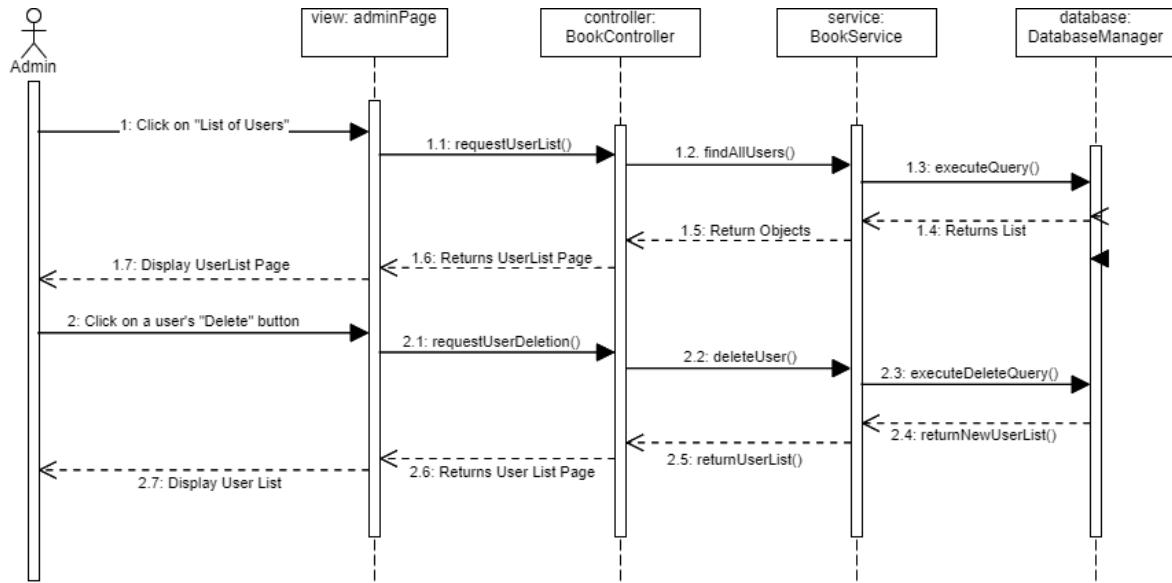


Figure. SQ - Admin: Delete Users

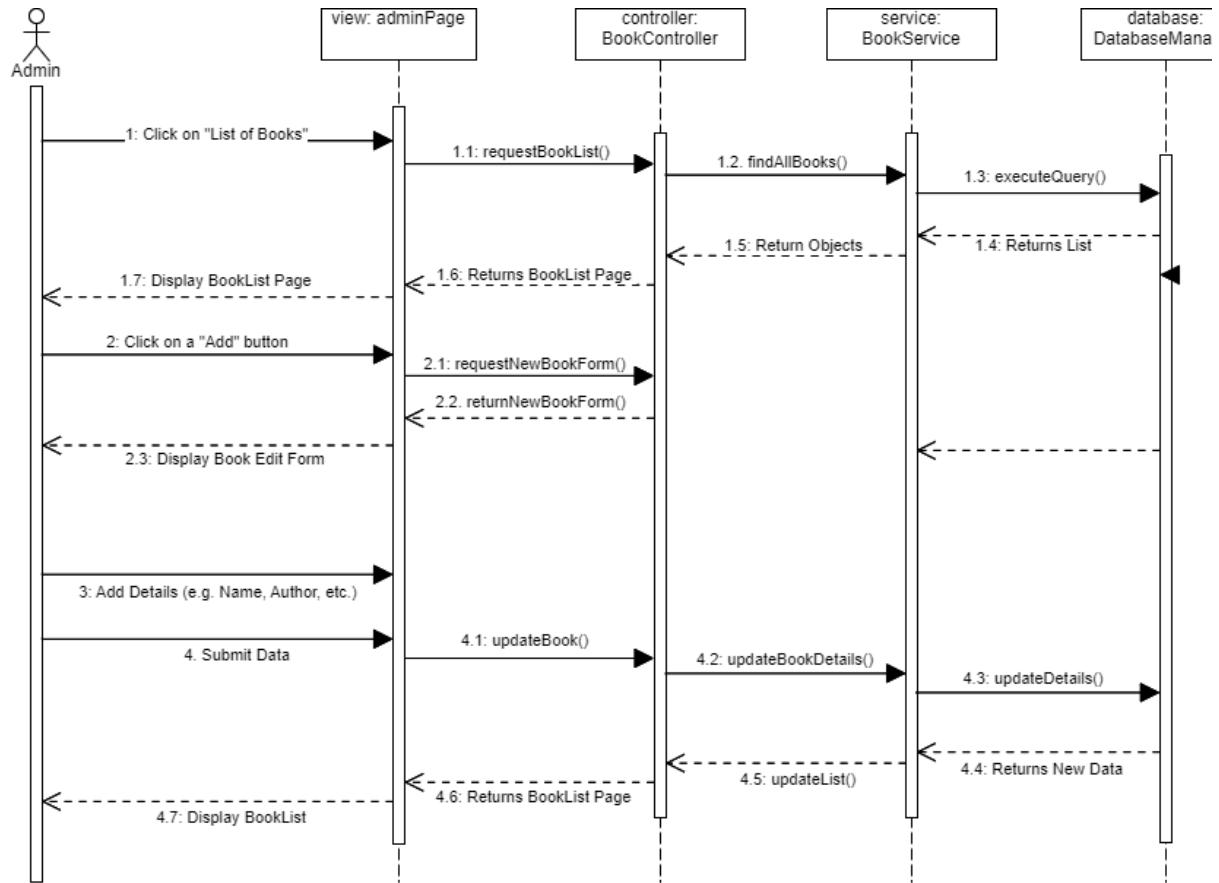


Figure. SQ - Admin: Add Books

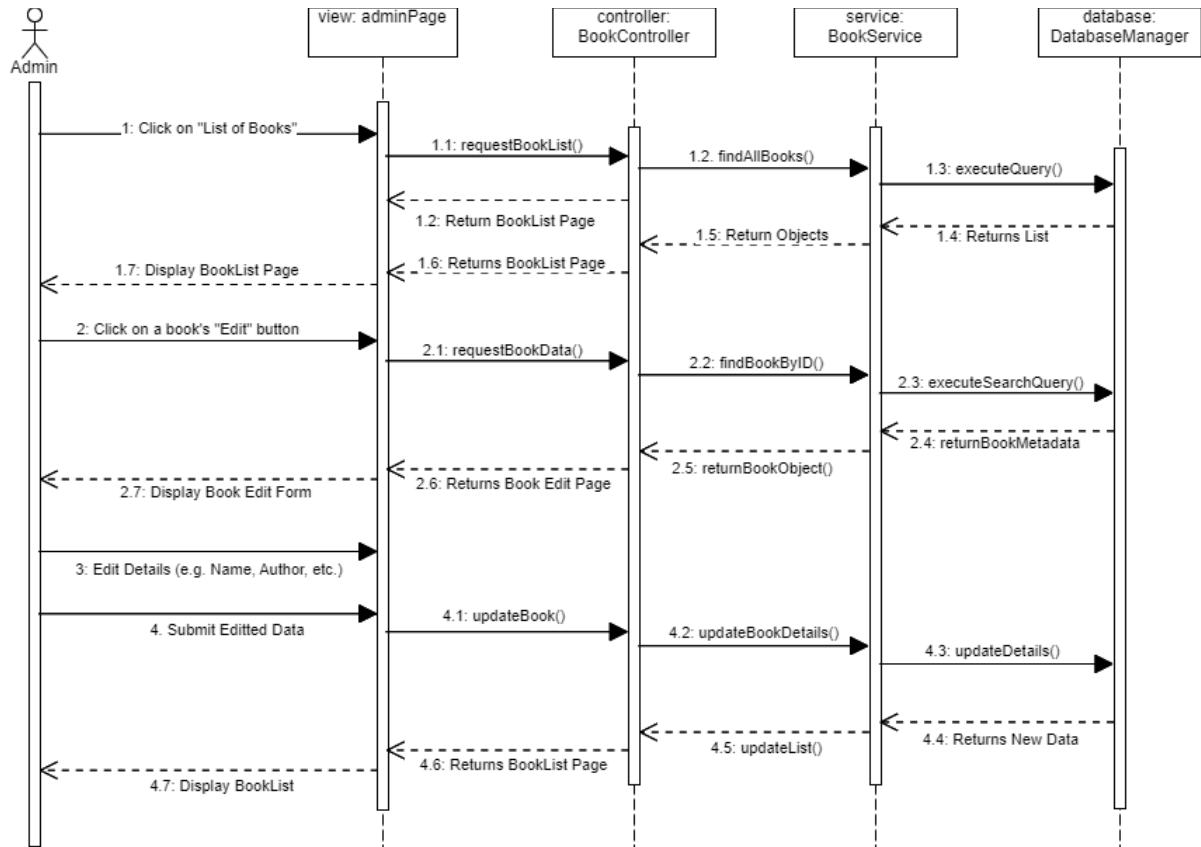


Figure. SQ - Admin: Edit Books

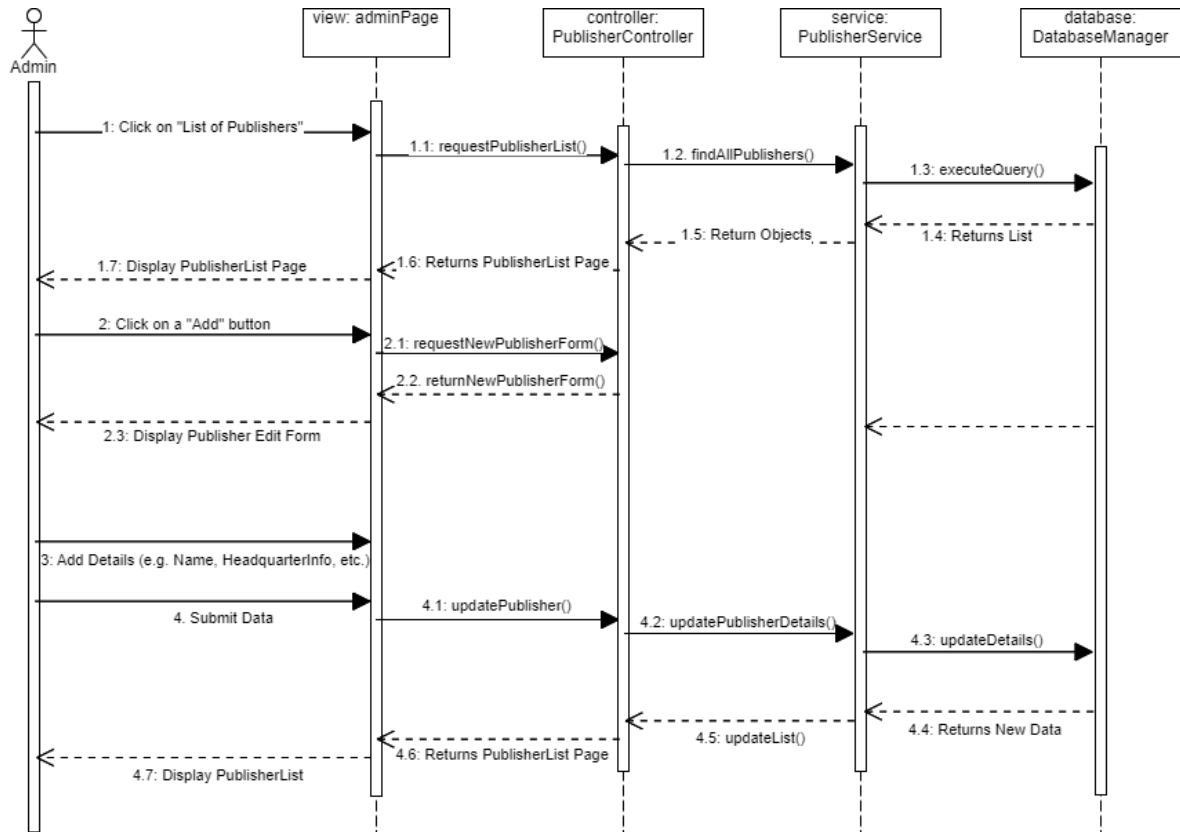


Figure. SQ - Admin: Add Publishers

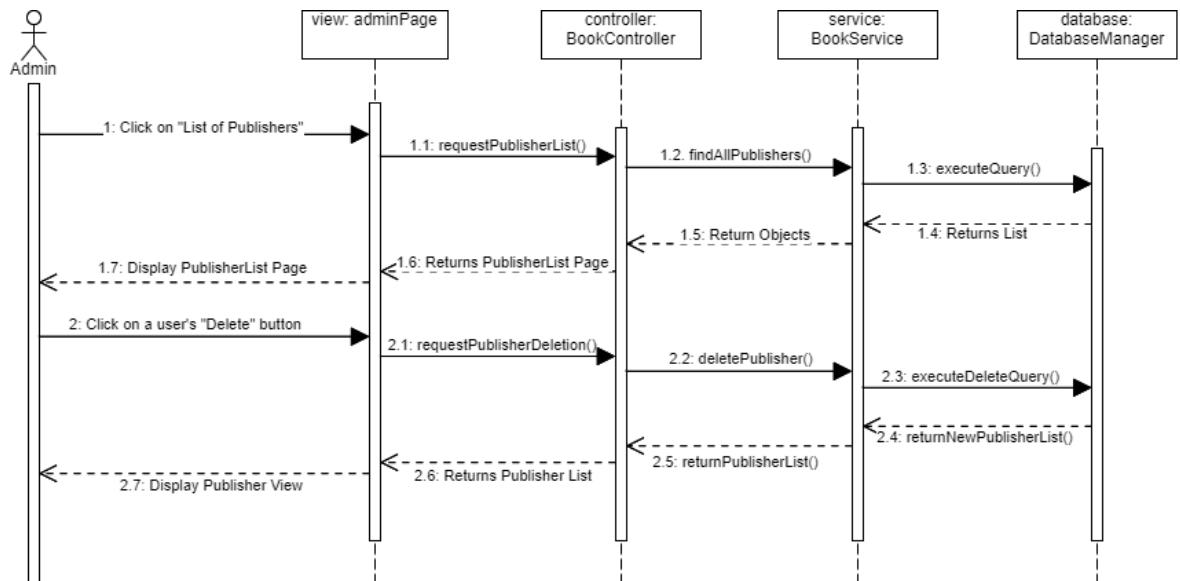


Figure. SQ - Admin: Delete Publishers

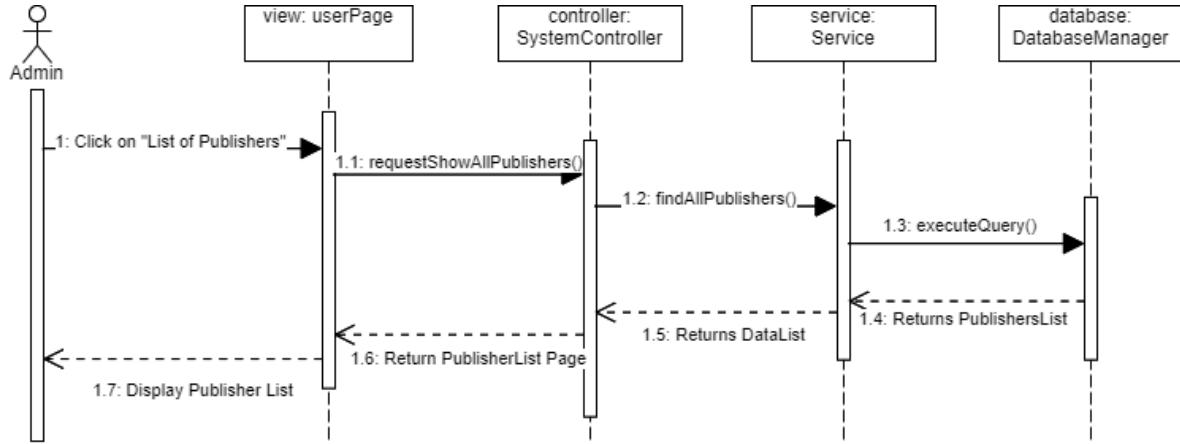


Figure. SQ - Admin: View Publishers

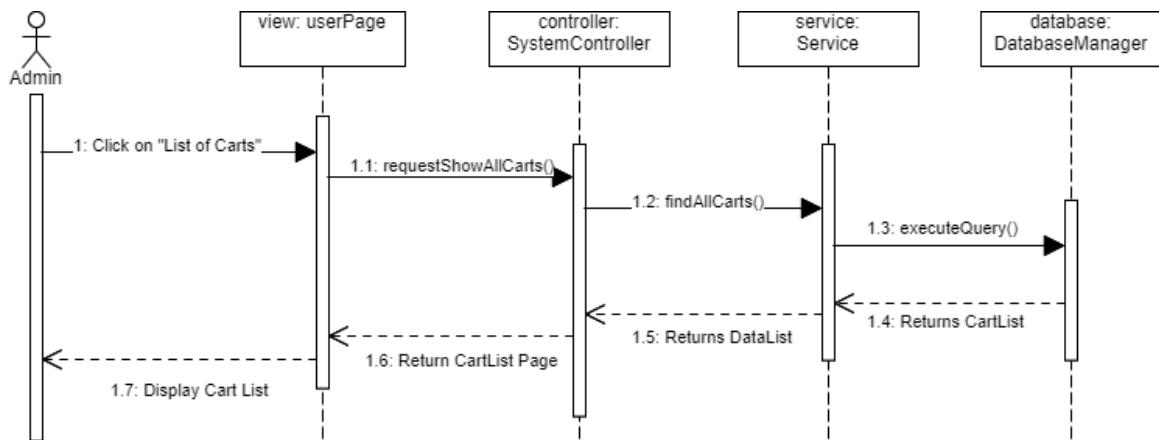


Figure. SQ - Admin: View Carts

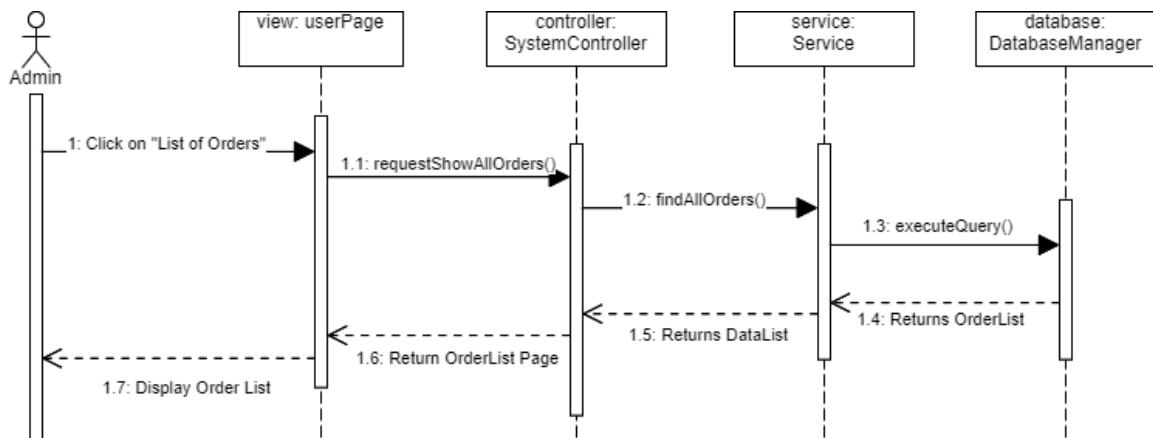


Figure. SQ - Admin: View Orders

III. IMPLEMENTATION

(Some functions are still under development - not everything shown in the screenshots will be mentioned. Future works are planned to be updated soon, and will be added to the later versions of the report.)

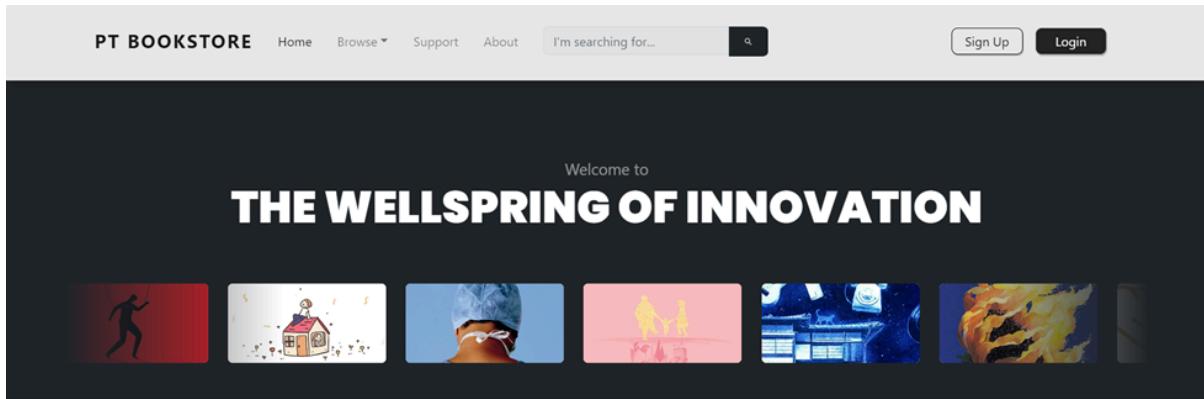
A. User

I. Account Management

Home

When a user first login the website, the homepage will be displayed (Figure ..). This plays as the head start for a visitor, where he or she can scan through the page to find what are currently on trend, daily book of choice, books that are highly rated by other readers, and so on.

The home page is divided into *sections*, each section introducing different categories of books and magazines that a user might find interesting.



New Arrivals



Figure 3.1. PT BOOKSTORE Homepage

Some sections can be mentioned: *New Arrivals*, *Today's Pick*.

New Arrivals



Figure 3.2. PT BOOKSTORE “New Arrivals” section

The Header

For better navigation, and to make it easy for a user to switch back and forth, the ***header*** is fixed at the top of *every* page of the website. This header let the user reach the books they want faster, and go to different sections effectively.

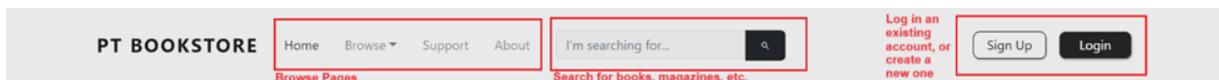


Figure 3.3. Header

A quick guide to the buttons/links on the header:

Home – Go to the home page.

Browse – Find all items available, or go to different sections.

Search Bar – Find items by keywords of choice.

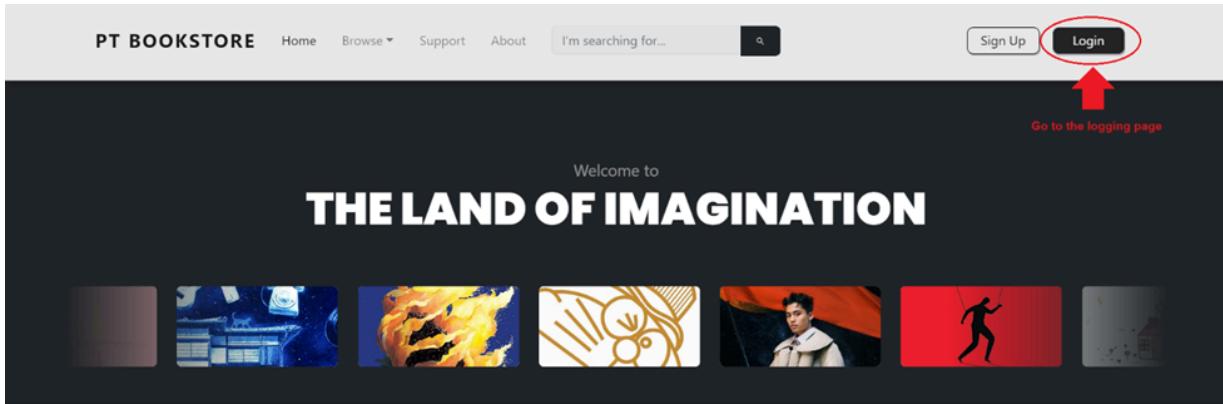
Log In – Log in an existing account.

Sign Up – Create a new account.

Login

By default, any visitors can find the books they want. But, for a personalizing experience, and most importantly, to be able to purchase products, every user needs to create and log in his or her account to pick items and pay for them. The login functionality is discussed first to create a better understanding of account registration and customization.

To start logging in, the button on the top-right corner can be clicked.



New Arrivals



Figure 3.4a. Click on the Login button

The following page will show up:

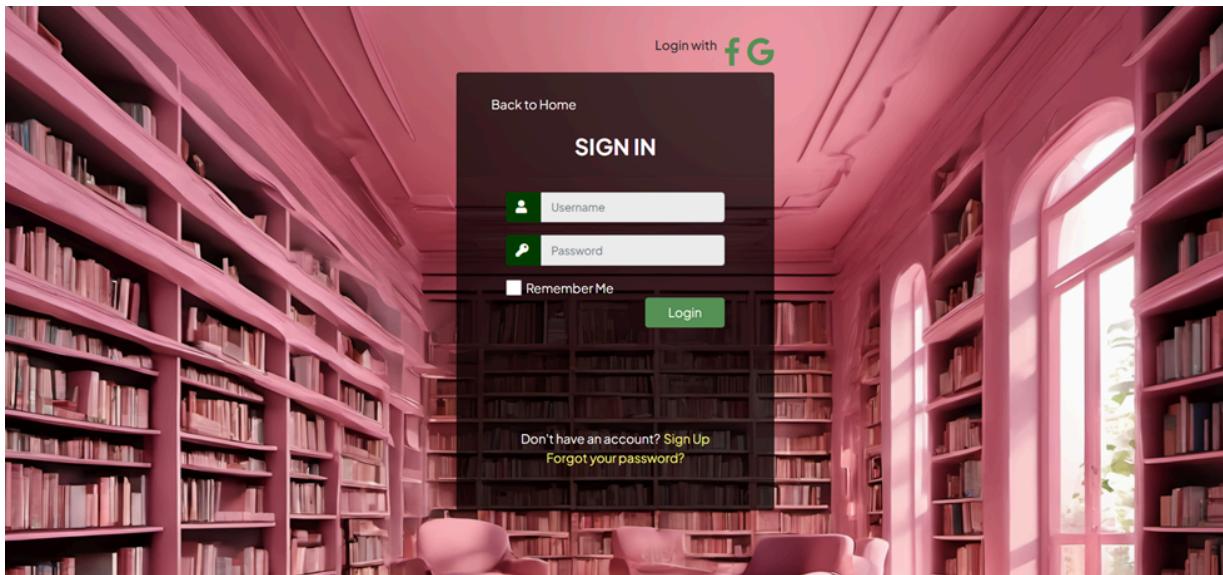


Figure 3.4b. Login form

The page will add a username and password. The user can click on each field and fill in the data.

Situation: *What if the user enters the wrong username and/or password.*

Solution: In this case, a message will pop up, reminding him or her to retype the credentials.

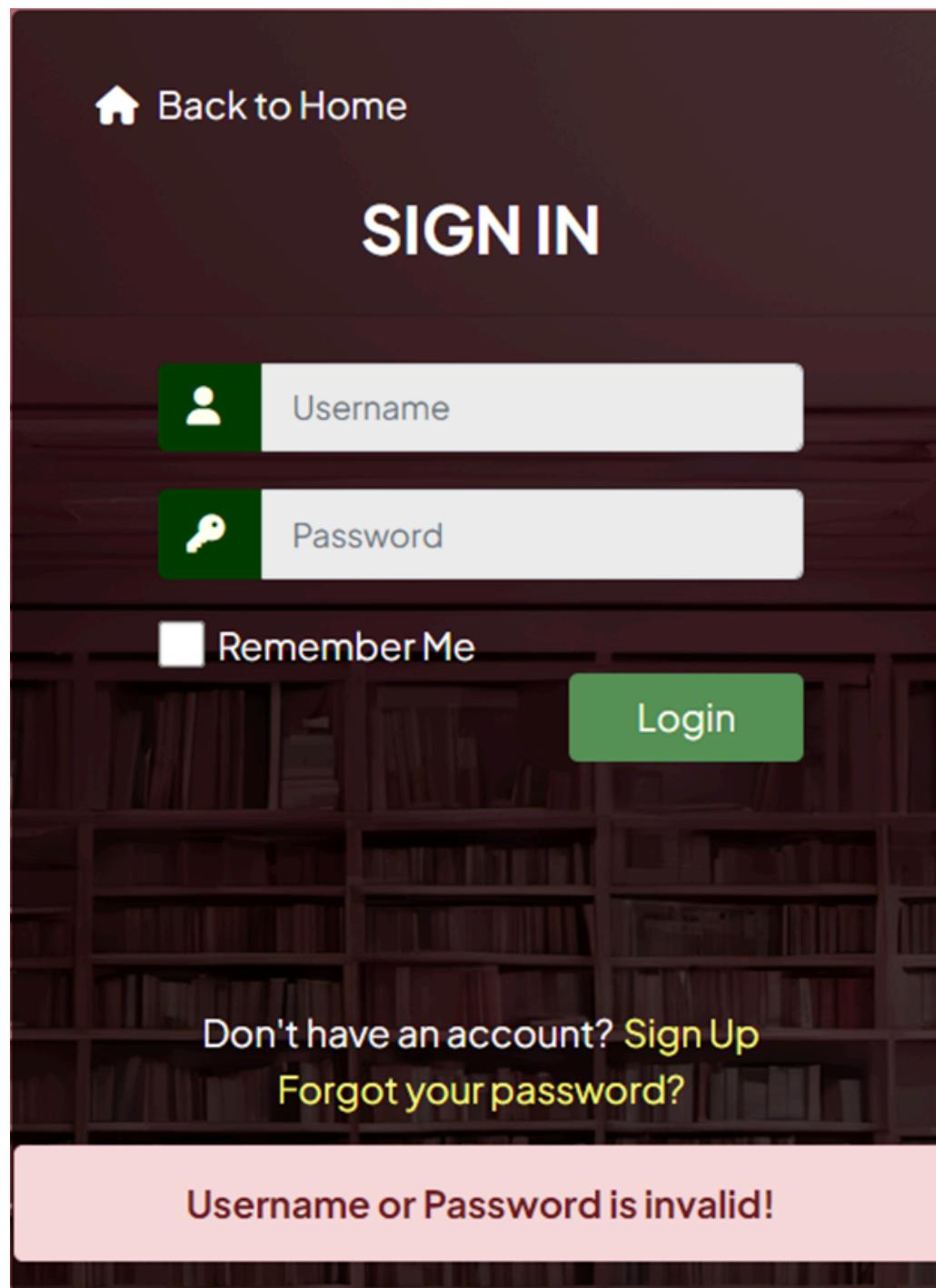


Figure 3.4c. Invalid credentials warning message

After logging into the website, the *Cart* and *Profile* menus will appear, allowing users to add books to the cart or managing account (Account Management will be updated later).



Figure 3.4d. Header: User logged in successfully

Register / Sign Up

To log in the website, an account is a must-have element. And a user needs to *create a new account* to log in the website.

Here is the breakdown of the “Sign Up” process:

1. A user needs to click on the “Sign Up” button on the top-right corner.



Figure 3.5a. Click on the Sign Up button

2. An registration form will be displayed.

[Back to Home](#)

SIGN UP

First Name*

Last Name*

Email*

Address

Phone Number*

Username*

Password*

Re-enter Password*

[Create](#)

Already have an account? [Log In](#)

Figure 3.5b. Sign Up form

3. The user can now enter the information required.

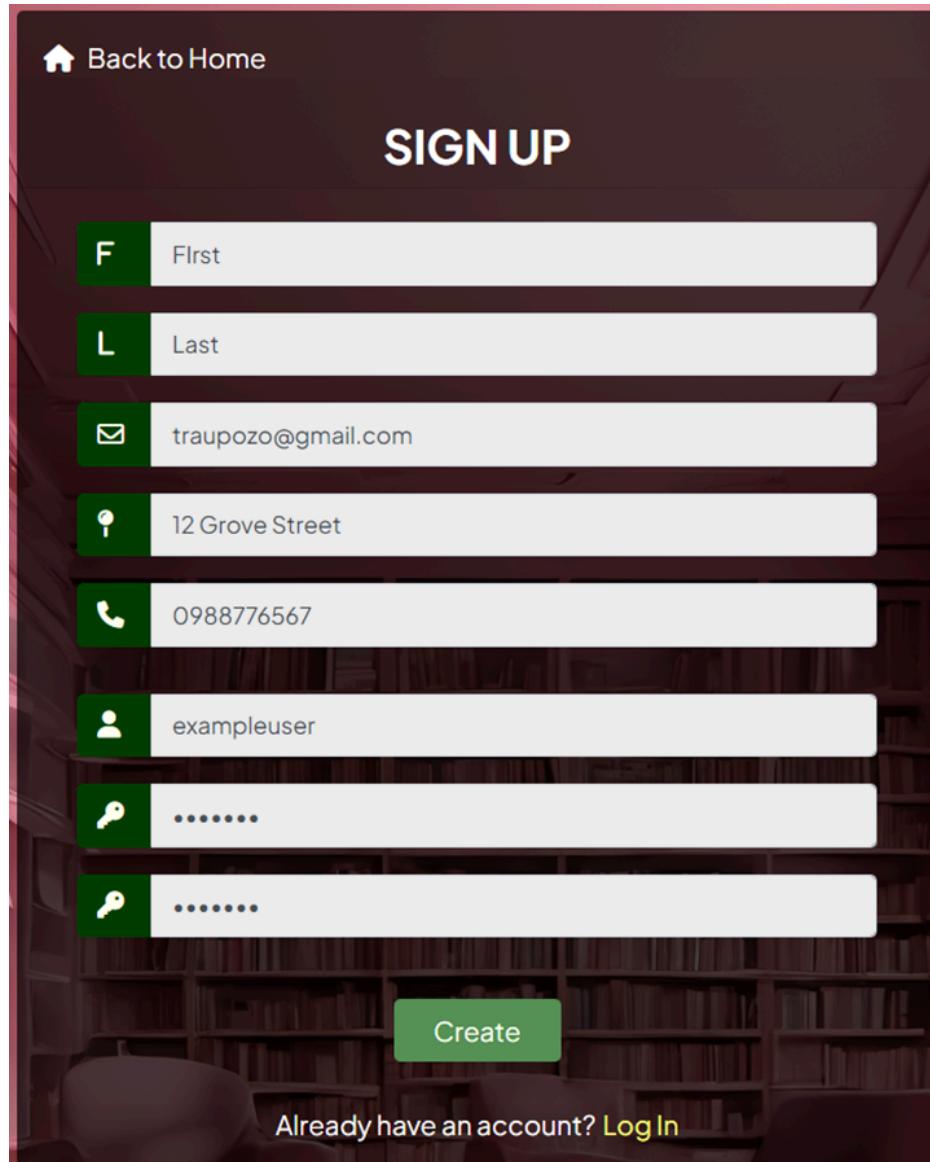


Figure 3.5c. Example of filling data

4. After hitting the “Create” button, the user now needs to *verify the account* with the entered email address. An email will be sent to that email address with a One-time Password (known as OTP). The user can check his or her mail’s inbox and grab the code.

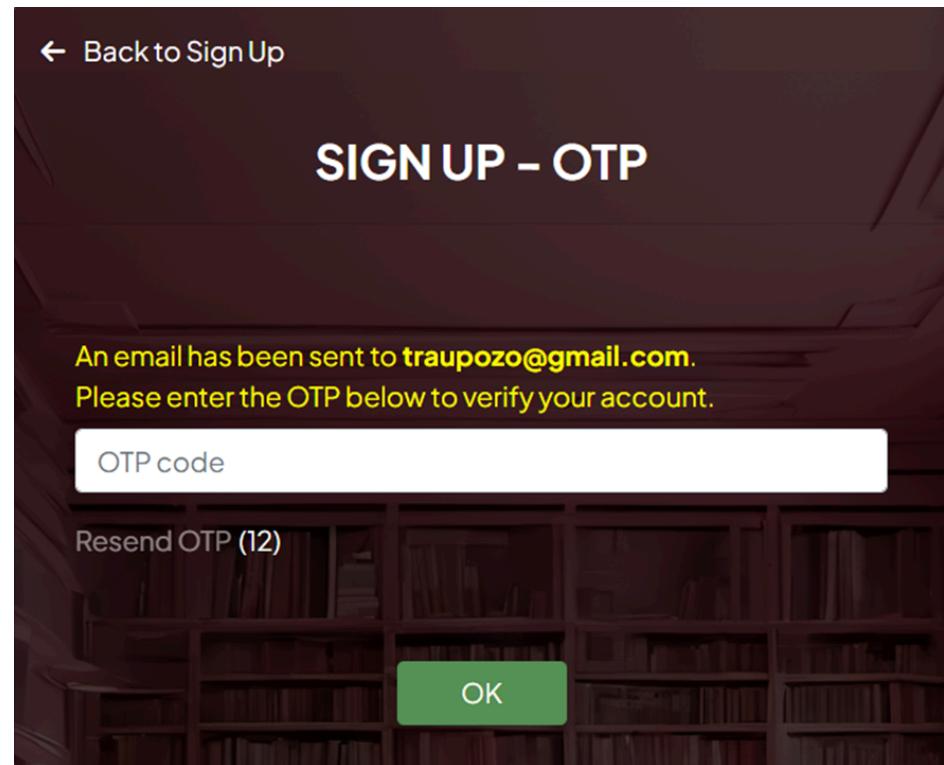


Figure 3.5d. OTP Window

Validation Code for Signing up PT Book Store

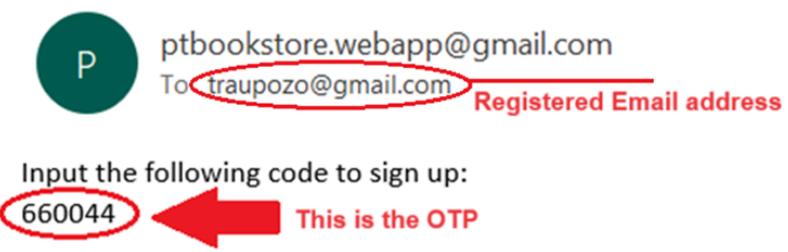


Figure 3.5e. The example email that is sent to the user

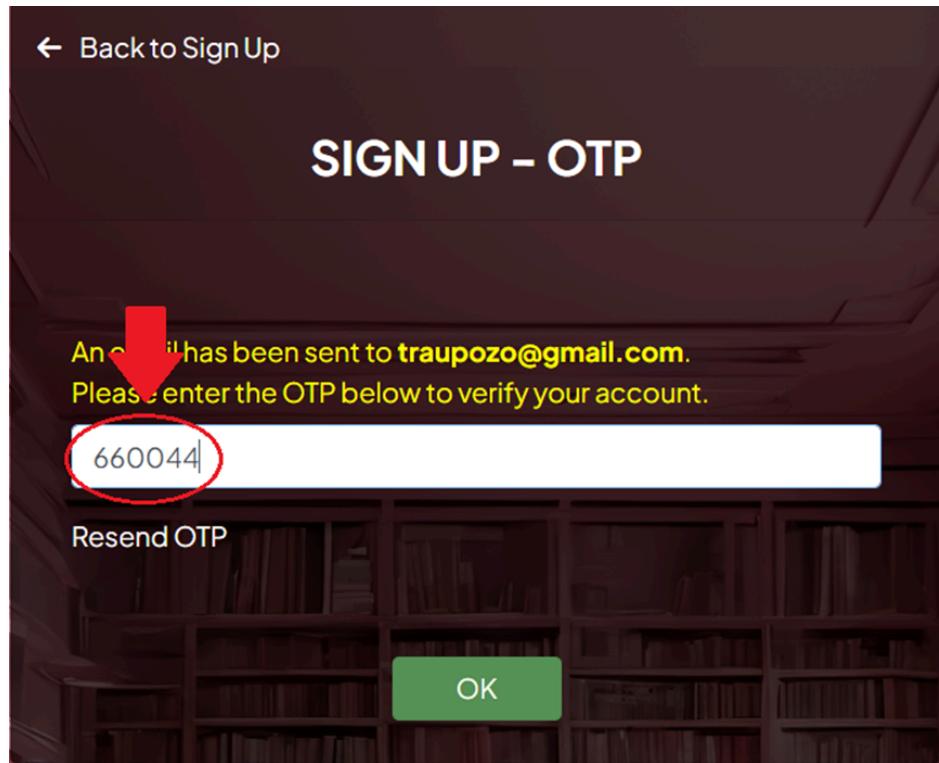


Figure 3.5f. The user enter the given OTP

5. The system will check if the OTP is matched. If not, an error message will be returned, then the user will either need to:
 - Click on “Resend OTP” to get a new code, or.
 - Refresh the page (this will also trigger the OTP resending protocol).
6. If everything is valid, an account with the provided information will be created.

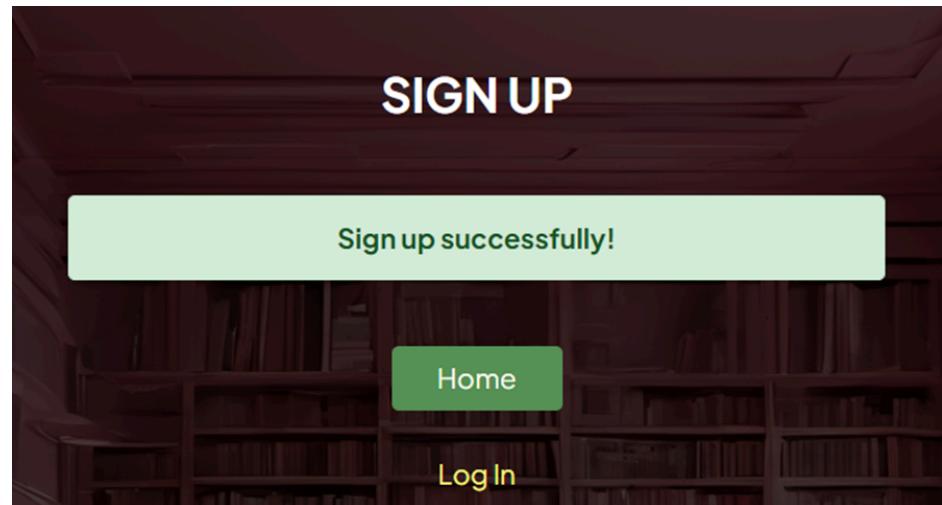


Figure 3.5g. New account confirmation

II. Browse Books

All Books

Along with the homepage (where the highlighted items are present), a user can find all available books on this website. Select **All Books** from the **Browse** dropdown menu, and it will lead to the big library.

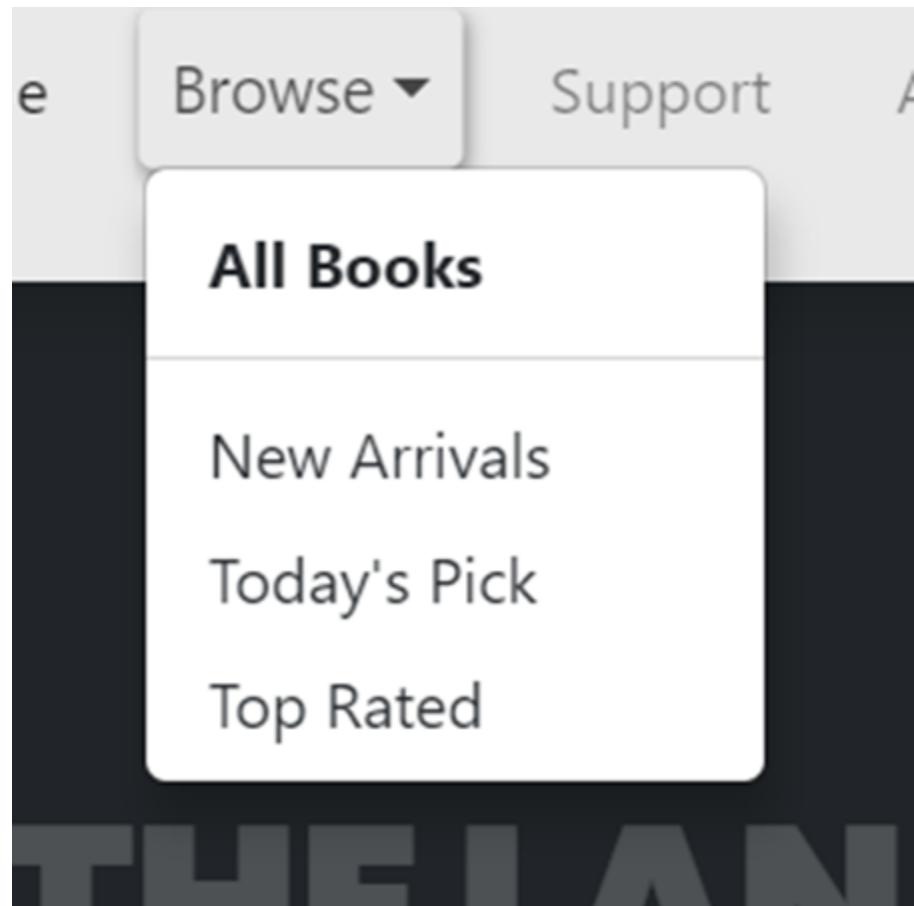


Figure 3.6. Browse Dropdown menu

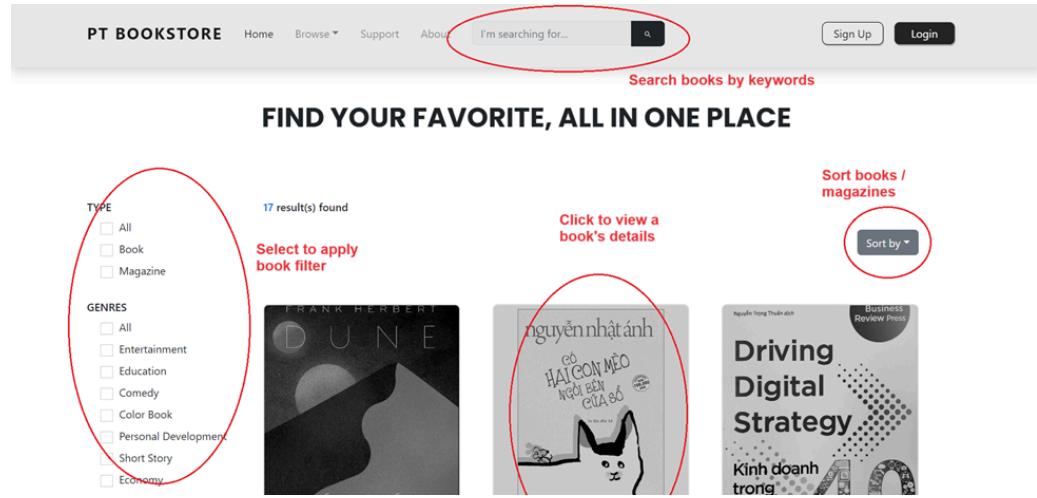


Figure 3.7. All Books page overview

Book Card - Click on a book card to view its details.

Search – The search bar on the header to find books by keywords.

Filter – This includes all conditions to browse desired books faster.

Sort – Sort book by conditions (sort by price, or by date added).

Book Card



Figure 3.8. A Book Card

Book Details

This is where the users can read a book in deep detail. He or she can see how many like the content, add the product to cart, and see similar content.

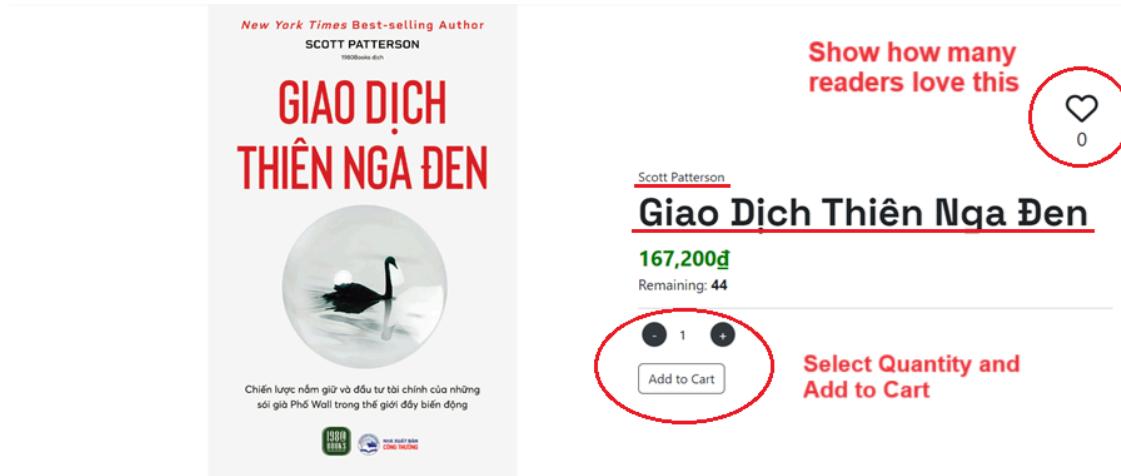


Figure 3.9. A Book Details page



Figure 3.10. Similar content

Comments

This is where the users can read comments about the book and share their opinions about the product, as well.

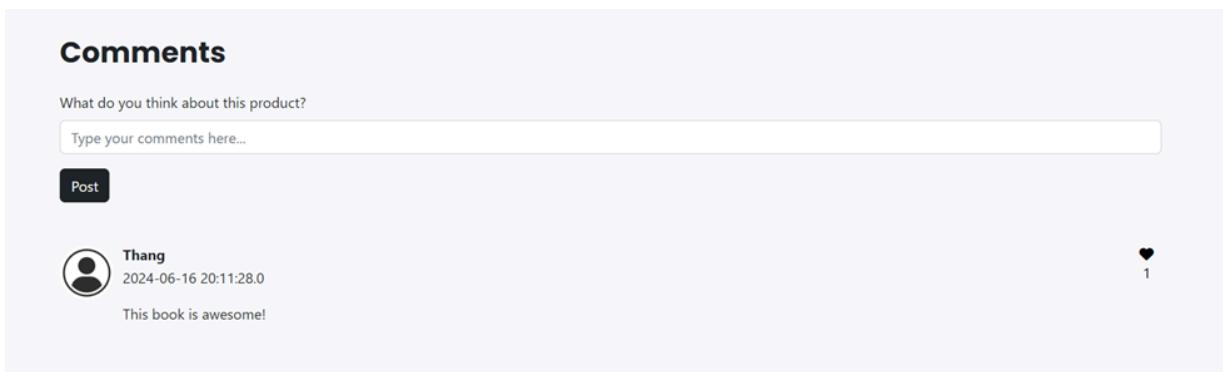


Figure 3.11. Comment section

III. Cart

This is where the users can read comments about the book and share their opinions about the product, as well.

A screenshot of a shopping cart interface. At the top, it says "Shopping Cart" and "1 item(s) in your cart". Below is a table with columns "Product", "Price", and "Quantity". One item is listed: "Giao Dịch Thiên Nga Đen" by Scott Patterson, priced at 167,200đ, with a quantity of 1. There are minus and plus buttons for adjusting quantity and a "Remove" button. At the bottom, it shows a subtotal of "167,200đ". Navigation buttons "Continue Shopping" and "Checkout" are at the bottom.

Figure 3.12. Shopping Cart interface

B. Admin

I. Admin Homepage

After the user logs in as an admin the system will lead to the homepage of the admin. The admin has full access to manage books, authors, publishers, carts, orders and comments. The admin can choose the corresponding management by clicking on the cards in the home page or the items in the list of the menu.

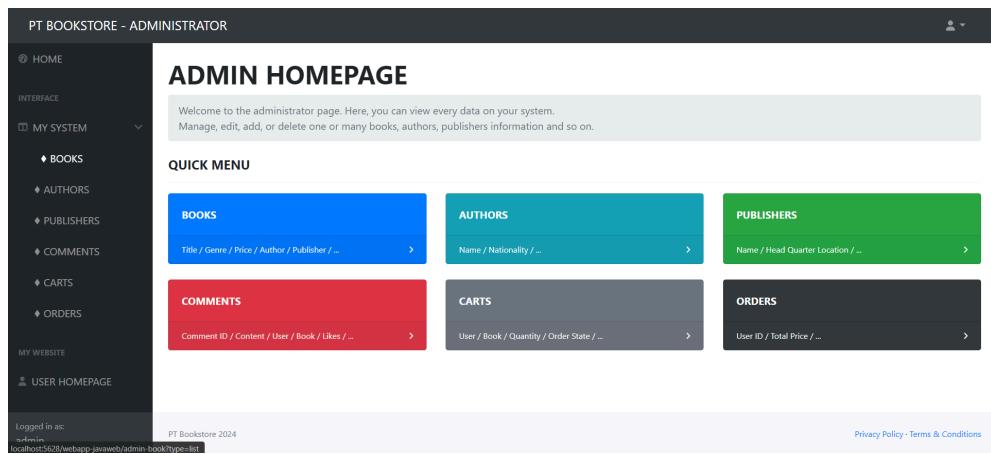


Figure 3.13. Admin Homepage

Books Management

The screenshot shows the 'BOOK MANAGER' section. The sidebar includes 'HOME', 'INTERFACE', 'MY SYSTEM' (with sub-options: BOOKS, AUTHORS, PUBLISHERS, COMMENTS, CARTS, ORDERS), 'MY WEBSITE', and 'USER HOMEPAGE'. It also shows 'Logged in as: admin'. The main area has a header 'BOOK MANAGER' with a sub-header 'Find, add, edit, and delete your books here.' A search bar at the top right contains 'Search for...' and a magnifying glass icon. Below is a table with the following numbered annotations:

- 1: 'Filter' button.
- 2: Filter dropdown menu.
- 3: 'Entries per page' dropdown.
- 4: Column headers for the table: ID, Title, Author, Type, Genre, Price, Publisher, Publication Date, Description, Stock.
- 5: 'Search...' input field.
- 6: Edit icon (pencil).
- 7: Delete icon (trash bin).

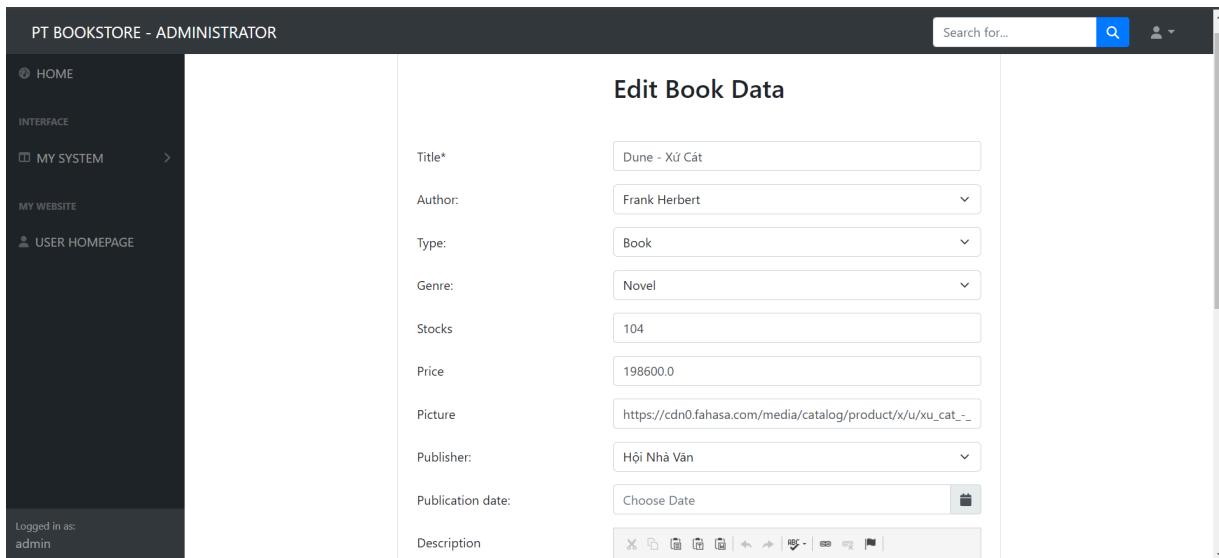
The table displays one book entry:

ID	Title	Author	Type	Genre	Price	Publisher	Publication Date	Description	Stock	Action	
3024356144	Dune - Xu Cát	Frank Herbert	Book	Novel	198.600	Hội Nhà Văn		Là một trong những cuốn tiểu thuyết khoa học giả tưởng bán chạy nhất mọi thời đại. Xu Cát không chỉ là lựa chọn đối với những tín đồ của Chúa nhẫn, Chiến tranh giữa các vì sao... mà còn chính phục độc giả dù mọi lứa tuổi, mọi tầng lớp và sở thích bởi sự đa dạng và phong phú của cốt truyện và không gian trong truyện.	104		

Figure 3.14a. List of Books

- (1) Search book by the filter.
- (2) Add a new book.
- (3) Adjust number of books per one page.
- (4) Sort by title, the other columns are the same.
- (5) Search by keyword.
- (6) Edit book.
- (7) Delete book.

In the edit-book function, the admin can change the properties of the book in the input field with the corresponding data type. If the admin input wrong format, the warning message will appear.



The screenshot shows the 'Edit Book Data' form in the PT Bookstore - Administrator interface. The left sidebar has a dark theme with white text and includes links for HOME, INTERFACE, MY SYSTEM, MY WEBSITE, and USER HOMEPAGE. It also shows 'Logged in as: admin'. The main area has a light background and contains the following fields:

Edit Book Data	
Title*	<input type="text" value="Dune - Xứ Cát"/>
Author:	<input type="text" value="Frank Herbert"/>
Type:	<input type="text" value="Book"/>
Genre:	<input type="text" value="Novel"/>
Stocks	<input type="text" value="104"/>
Price	<input type="text" value="198600.0"/>
Picture	<input type="text" value="https://cdn0.fahasa.com/media/catalog/product/x/u/xu_cat_-_.jpg"/>
Publisher:	<input type="text" value="Hội Nhà Văn"/>
Publication date:	<input type="text" value="Choose Date"/>
Description	<input type="text" value=""/>

Figure 3.14b. Edits Book Data

After finishing the input, the admin clicks ‘Update’ to complete the update.

PT BOOKSTORE - ADMINISTRATOR

Search for...

Publication date: Choose Date

Description

Created by: PT

Created date: 2024-06-14 16:07:38.0

Update Cancel

Figure 3.14c. Updates Book Data

The **add-new-book** screen looks like the edit-book one, but the information of the input fields are empty.

PT BOOKSTORE - ADMINISTRATOR

Search for...

Edit Book Data

Title*

Author: Select Author

Type: Select type
June 2024

Genre:

Stocks

Price

Picture

Publisher:

Publication date: Choose Date

Figure 3.14d. Add New Book

Delete book: After clicking the delete button, a message will appear to confirm whether the admin wants to delete or not.

PT BOOKSTORE - ADMINISTRATOR													
HOME													
INTERFACE													
MY SYSTEM													
MY WEBSITE													
USER HOMEPAGE													
Logged in as: admin		3032675993	Góc Nhỏ Có Nắng	Little Rainbow	Book	Color Book	52,000đ	NXB Thanh Niên	2024-06-14	truyện, bối cảnh đặc đáo và thú vị của khơi gợi kinh nghiệm như là một lời thú thật tình cảm gửi đến bạn: Thư giản và chữa lành: Với những hình ảnh đẹp mắt và gây ấn tượng, tóm tắt sẽ là một phương pháp hiệu quả...	99		
		3176930502	Cô Hai Con Mèo Ngôi Bên Cửa Số (2023 Edition)	Nguyễn Nhật Ánh					AI CON MÈO NGÔI BÊN CỦA SỐ là tác phẩm tác giả nhà văn Nguyễn Nhật Ánh viết theo thể đong thoại. Đặc biệt hơn nữa là viết về tình bạn hai loài vốn là thù địch của nhau mèo và chuột...	72			
		3265419780	Kinh Doanh Trong Thời Đại 4.0 - Driving Digital Strategy (2022)	Sunil Gupta			Quốc Dân		Đây là sách cập nhật nhất - "Kinh doanh trong thời đại 4.0", một tác phẩm của Sunil Gupta - giáo sư Quản trị Kinh doanh tại trường Kinh doanh Harvard, chuyên gia về chiến lược chuyển đổi kỹ thuật số sẽ dẫn...	33			
		3287820104	Giao Dịch Thiên Nga Bạc	Scott Patterson	Book	Economy	167,000đ	Công Thương	2024-06-14	Giao dịch Thiên nga Đen, cuốn sách best seller của Wall Street Journal, tiết lộ cho chúng ta chiến lược đầu tư tài chính của những nhà đầu tư Blue Chip.	43		

Figure 3.14e. Delete Book

Authors Management

The admin can manage the information of the author as same as managing books.

PT BOOKSTORE - ADMINISTRATOR														
HOME														
INTERFACE														
MY SYSTEM														
MY WEBSITE														
USER HOMEPAGE														
Logged in as: admin	localhost:5628/webapp/javaweb/admin/home	AUTHOR MANAGER												
		Find, add, edit, and delete authors.												
		Database: online-bookstore												
		<table border="1"> <thead> <tr> <th></th> <th>entries per page</th> <th>Search...</th> </tr> </thead> <tbody> <tr> <td>10</td> <td>▼</td> <td></td> </tr> </tbody> </table>								entries per page	Search...	10	▼	
	entries per page	Search...												
10	▼													
		ID	Name	Nationality	Action									
		130264	Phùng Thành Ngọc	Vietnam										
		146998	Chà	Vietnam										
		155478	Tony Crilly	Russia										
		186437	Edward Tse	China										
		276074	Frank Herbert	US										

Figure 3.15. Authors Manager

Publishers Management

Same as book management.

The screenshot shows the 'PUBLISHER MANAGER' section of the PT Bookstore Administrator interface. The left sidebar includes links for Home, Interface, My System (Books, Authors, Publishers, Comments, Carts, Orders), My Website, and User Homepage. The main area displays a table of publishers with columns for ID, Name, Head Quarter Location, and Action (trash and edit icons). The data shows five entries, all from Vietnam:

ID	Name	Head Quarter Location	Action
1189	One Pacific Holdings	Vietnam	[trash, edit]
1586	Hồng Đức	Vietnam	[trash, edit]
2584	ĐH Kinh Tế Quốc Dân	Vietnam	[trash, edit]
3553	Công Thương	Vietnam	[trash, edit]
4556	Thế Giới	Vietnam	[trash, edit]

Figure 3.15. Publishers Manager

Comments Management

The admin has the access to view and delete comments of users.

The screenshot shows the 'COMMENT MANAGER' section of the PT Bookstore Administrator interface. The left sidebar includes links for Home, Interface, My System (Books, Authors, Publishers, Comments, Carts, Orders), My Website, and User Homepage. The main area displays a table of comments with columns for ID, Content, User ID - User Name, Book ID - Book Name, Likes, Created date, and Action (trash icon). One entry is shown:

ID	Content	User ID - User Name	Book ID - Book Name	Likes	Created date	Action
39	Good book!	1 - phucbuidoi	3806869299 - Đinh Luật Peter	1	2024-06-14 16:25:24.0	[trash]

Showing 1 to 1 of 1 entries

Figure 3.15. Comments Manager

Carts Management

The user can view all the information of the available cart of all users.

The screenshot shows the 'PT BOOKSTORE - ADMINISTRATOR' interface. On the left, a sidebar lists 'INTERFACE' options: HOME, MY SYSTEM, MY WEBSITE, and USER HOMEPAGE. The 'HOME' option is selected. At the bottom of the sidebar, it says 'Logged in as: admin'. The main content area is titled 'CART MANAGER' and displays a table of cart entries. The table has columns: ID, User ID - User Name, Book ID - Book Name, Quantity, Order ID, Is ordered, and Created date. There are two entries:

ID	User ID - User Name	Book ID - Book Name	Quantity	Order ID	Is ordered	Created date
33	1 - phucbuidoi	3176930502 - Cố Hai Con Mèo Ngồi Bên Cửa Sổ (2023 Edition)	2	0	NO	2024-06-15 23:44:53.957
34	3 - sunphan	3287820104 - Giao Dịch Thiên Nga Đen	1	6	YES	2024-06-16 09:51:26.753

Below the table, a message says 'Showing 1 to 2 of 2 entries'. At the top right of the main area, there is a search bar with a magnifying glass icon and a user profile icon. The footer contains copyright information: 'Copyright © Your Website 2023' and links to 'Privacy Policy · Terms & Conditions'.

Figure 3.15. Carts Manager

Orders Management

The user can view all the information of the available orders of all users.

The screenshot shows the PT BOOKSTORE - ADMINISTRATOR interface. On the left, there's a sidebar with links: HOME, INTERFACE, MY SYSTEM, MY WEBSITE, and USER HOMEPAGE. The main area is titled 'ORDER MANAGER' and displays a table of orders. The table has columns for ID, User ID - User Name, Total Price, and Created date. There are two entries: one for user ID 1 (phucbuidoi) with a total price of 260000.0 and a creation date of 2024-06-13 19:54:08.0, and another for user ID 3 (sumphan) with a total price of 167200.0 and a creation date of 2024-06-16 09:51:46.0. At the bottom, it says 'Showing 1 to 2 of 2 entries'. The top right has a search bar and a user profile icon. The bottom right includes links for Privacy Policy and Terms & Conditions.

ID	User ID - User Name	Total Price	Created date
5	1 - phucbuidoi	260000.0	2024-06-13 19:54:08.0
6	3 - sumphan	167200.0	2024-06-16 09:51:46.0

Figure 3.15. Orders Manager

IV. DISCUSSION & CONCLUSION

1. Limitation

Full functions: The online bookstore needs a huge database to manage all the related components.

Potential bugs: The web needs long-term support to test all the available functions.

Illegal access via URL: There are some potential situations that the users can access to the invalid functions by URL.

System design: the design and the implementation is just in the new-born step.

2. Future Improvement

For new updates, the database will be updated and new functions will be added.

The maintenance phase will occur regularly and fix all the reported errors.

The authorization will be more strict to control the flow of accessing URLs.

For the long-term application, the structure of the class or the code organization will be restructured and improved.

V. REFERENCES

1. YouTube - Lập trình Java web - javaweb.

VI. CONTRIBUTION

1. Đỗ Đình Phúc - ITCSIU21219 : Back-end and Middle.
2. Phan Quốc Thắng - ITCSIU21232: Front-end and UI design.