
Online Social Bookstore Application

Sprint Report

Kailidis Kyrillos AM:4680,

Petrogiannis Georgios AM:4779

VERSIONS HISTORY

Date	Version	Description	Author
5/4/2024	0.1	Initial Draft	Kailidis Kyrillos Petrogiannis Georgios
10/4/2024	0.2	Added Design and Use Cases	Kailidis Kyrillos Petrogiannis Georgios
15/4/2024	0.3	Start coding domain model package	Kailidis Kyrillos Petrogiannis Georgios
18/4/2024	0.4	Start coding services controllers	Kailidis Kyrillos Petrogiannis Georgios
28/4/2024	0.5	Set up config package	Kailidis Kyrillos Petrogiannis Georgios
30/4/2024	0.6	Set up the database	Kailidis Kyrillos Petrogiannis Georgios
10/5/2024	0.7	Start creating HTML pages	Kailidis Kyrillos Petrogiannis Georgios
12/5/2024	0.8	Fixing Bugs and Maintenance	Kailidis Kyrillos Petrogiannis Georgios
16/5/2024	0.9	Created Unit Tests	Kailidis Kyrillos Petrogiannis Georgios

18/5/2024	1.1	Final Version	Kailidis Kyrillos Petrogiannis Georgios

1 Introduction

This document provides information concerning the final sprint of the project.

1.1 Purpose

The purpose of this document is to provide a detailed report of the activities, progress, and design for the specified sprint of the Online Social Bookstore Application.

1.2 Document Structure

The rest of this document is structured as follows. Section 2 describes out Scrum team and specifies the this Sprint's backlog. Section 3 specifies the main design concepts for this release of the project.

2 Scrum team and Sprint Backlog

2.1 Scrum team

Product Owner	Kailidis Kyrillos
Scrum Master	Petrogiannis Georgios
Development Team	<ul style="list-style-type: none"> • Kailidis Kyrillos • Petrogiannis Georgios

2.2 Sprints

<List below the sprints that you performed and the user stories that have been realized in each Sprint>

Sprint No	Begin Date	End Date	Number of weeks	User stories
1	22/4/2023	29/4/2023	1	As a user, I want to create a new account, so that I have access to the functionalities of the social bookstore application.
2	22/4/2023	29/4/2023	1	As a user, I want login to my account, so that I have access to the functionalities of the social bookstore application.
3	22/4/2023	29/4/2023	1	As a user, I want to logout from my account, so as to terminate my interaction with the social bookstore application.
4	15/4/2023	13/5/2023	4	As a user, I want to create a profile that includes my full name, address, age, phone number, the categories of books that I prefer (e.g., Art, Comic, Fantasy, Fiction, Biographies, History, Science, Literature, Adventure, Crime, Other) and my favorite authors, to facilitate the search/recommendation of books that may be interesting for me.
5	15/4/2023	13/5/2023	4	As a user, I want to be able to add a book offer in personal list of book offers with a description that includes the book title, the author(s), the category that the book belongs to and a

				summary of the book, to facilitate the search/recommendation of books by/to other users.
6	15/4/2023	13/5/2023	4	As a user, I want to browse a list of requests from other users who are interested in a book offer that I have made, so that I can decide to whom I shall give the book.
7	15/4/2023	13/5/2023	4	As a user, I want to select a user who requested a book from my list and notify him that he can take the book. I also want to notify the rest of the users who requested the book that the book has been taken by another user
8	15/4/2023	13/5/2023	4	As a user, I want to have access to the contact information of a selected user who requested a book that I offer, so that I can contact him to arrange the delivery of the book.
9	15/4/2023	13/5/2023	4	As a user, I want to remove a book that is no longer available from my personal list of book offers and from the book request lists of other users who requested for this book, to enable the accurate book search/recommendation.
10	15/4/2023	13/5/2023	4	As a user, I want to be able to search for book offers with a certain title and authors to find interesting books. Besides specifying the search criteria, I want to choose between an exact or an approximate search strategy. Then, I want to be able to make a request for a book that is included in the search results to the user who offers the book so that he becomes aware that I want to get the book.
11	15/4/2023	13/5/2023	4	As a user, I would like to browse a list of recommended book offers from the social bookstore application, to find interesting books. It would be nice to be able to choose among various

				<p>recommendation strategies that consider information given in my profile. Then, I want to be able to make a request for a recommended book to the user who offers the book so that he becomes aware that I want to get the book.</p>
--	--	--	--	--

3 Use Cases

<Specify the concrete Use Cases that describe the interaction of the user with the applications, as derived from the abstract user stories. Give a **UML Use Case diagram** and the **detailed use case descriptions**.>

3.1 <Use Case 1> Create New Account

Use case ID	UC1
Actors	User
Pre conditions	The user does not already have an account.
Main flow of events	<ol style="list-style-type: none"> 1. The use case starts when the user navigates to the sign-up page. 2. The user enters their username, password, and role. 3. The system validates the entered details. 4. The system creates a new account for the user. 5. The system redirects the user to the login page.
Alternative flow 1	If the username is already taken, the system prompts the user to register again using different credentials.
Post conditions	A new user account is created

3.2 <Use Case 2> Login to Account

Use case ID	UC2
Actors	User

Pre conditions	The user has an existing account.
Main flow of events	<ol style="list-style-type: none"> 1. The use case starts when the user navigates to the login page. 2. The user enters their username and password. 3. The system validates the entered credentials. 4. The system logs the user in and redirects them to the main dashboard.
Alternative flow 1	If the username or password is incorrect, the system displays a login error message.
Post conditions	The user is logged in and has access to the functionalities of the social bookstore application.

3.3 <Use Case 3> Logout from Account

Use case ID	UC3
Actors	User
Pre conditions	The user is currently logged in.
Main flow of events	<ol style="list-style-type: none"> 1. The use case starts when the user clicks the logout button. 2. The system logs the user out and redirects them to the login page.
Post conditions	The user is logged out, and their session is terminated.

3.4 <Use Case 4> Create Profile

Use case ID	UC4
Actors	User
Pre conditions	The user is logged in.
Main flow of events	<ol style="list-style-type: none"> 1. The use case starts when the user navigates to the profile page.

	<ol style="list-style-type: none"> 2. The user enters their full name, address, age, phone number, preferred book categories, and favorite authors. 3. The system saves the user's profile information.
Post conditions	The user's profile is created, including their full name, address, age, phone number, preferred book categories, and favorite authors.

3.5 <Use Case 5> Manage Book Offers

Use case ID	UC5
Actors	User
Pre conditions	The user is logged in.
Main flow of events	<ol style="list-style-type: none"> 1. Add Book Offer: <ol style="list-style-type: none"> 1.1 The use case starts when the user navigates to the "Add Book Offer" page. 1.2 The user enters the book title, author and category 1.3 The system validates the entered details. 1.4 The system adds the book offer to the user's personal list of book offers. 2. Browse Book Requests: <ol style="list-style-type: none"> 2.1 The user navigates to the "Show Book Requests" page. 2.2 The system displays a list of requests from other users who are interested in the book offer. 2.3 The user browses the list of requests. 3. Select and Notify User for Book Offer: <ol style="list-style-type: none"> 3.1 The user selects a user from the list to accept the request and notify them that they can take the book. 3.2 The system sends a notification to the selected user. 3.3 The system sends notifications to the rest of the users who requested the book, informing them that the book has been taken by another user. 4. Access Contact Information of Selected User: <ol style="list-style-type: none"> 4.1 The system displays the contact information of the selected user.

	<p>4.2 The user views the contact information so we can accept the request.</p> <p>5. Remove Book Offer:</p> <p>5.1 The user selects a book offer to remove.</p> <p>5.2 The system removes the book offer from the user's list of book offers.</p> <p>5.3 The system removes the book offer from the request lists of other users who requested the book.</p> <p>5.4 The system displays a success message.</p>
Alternative flow 1	If any required fields are missing or incorrectly formatted during the addition of a book offer, the system displays an error message.
Post conditions	The book offer is managed correctly (added, browsed, selected for notification, contact information accessed, and removed as required).

3.6 <Use Case 6> Search for Book Offers and Make a Request

Use case ID	UC6
Actors	User
Pre conditions	The user is currently logged in.
Main flow of events	<ol style="list-style-type: none"> 1. The use case starts when the user navigates to the "Search Book Offers" page. 2. The user enters the search criteria (book title and author). 3. The user selects the search strategy (exact or approximate). 4. The system performs the search based on the specified criteria and strategy. 5. The system displays a list of book offers matching the search criteria. 6. The user selects a book offer from the search results. 7. The user makes a request for the selected book offer. 8. The system sends a notification to the user who offers the book, informing them of the request.
Alternative flow 1	If no book offers match the search criteria, the system displays a message indicating no results found.

Post conditions	The user has requested a book offer, and the offering user is notified of the request.
------------------------	--

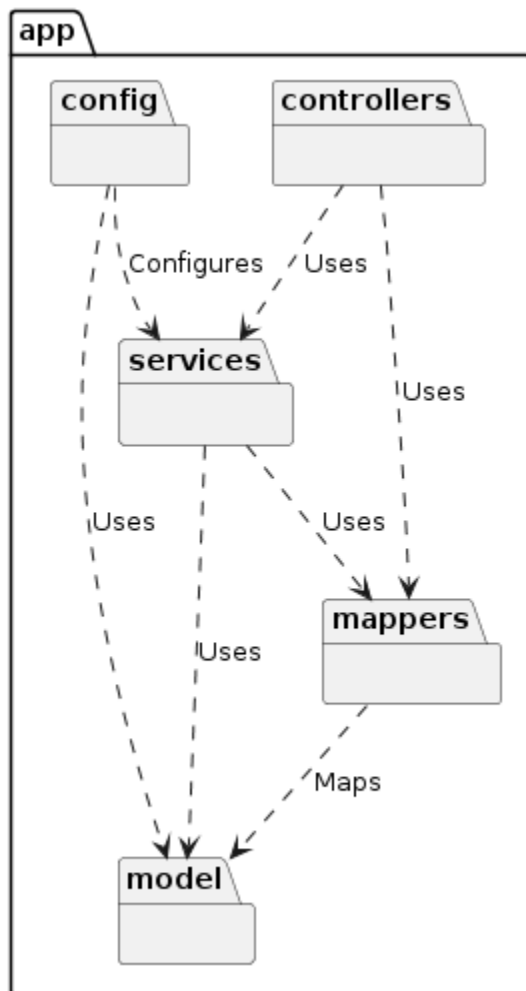
3.7 <Use Case 7> Browse Recommended Book Offers

Use case ID	UC7
Actors	User
Pre conditions	<ul style="list-style-type: none"> • The user is logged in. • The user has a completed profile with preferences for book categories and favorite authors.
Main flow of events	<ol style="list-style-type: none"> 1. The use case starts when the user navigates to the "View Recommendations" page. 2. The system generates a list of recommended book offers based on the user's profile information. 3. The system displays the list of recommended book offers.
Post conditions	The user gets to see a list of recommended book offers based on his favorites authors and book categories.

4 Design

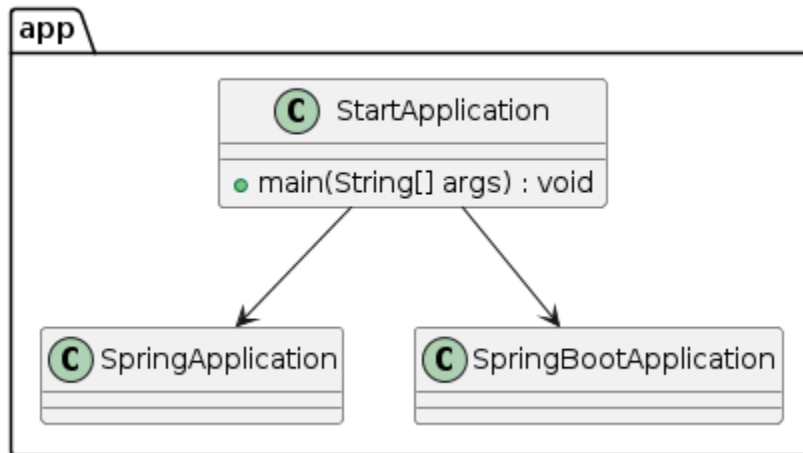
4.1 Architecture

UML package diagram:

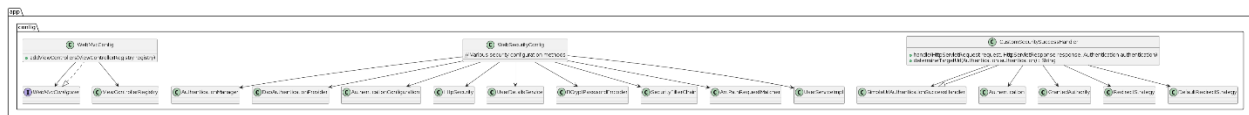


UML class diagrams:

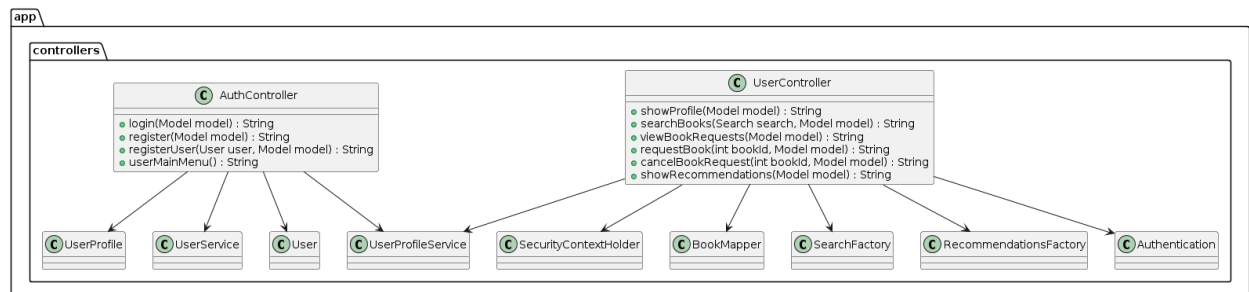
App Package:



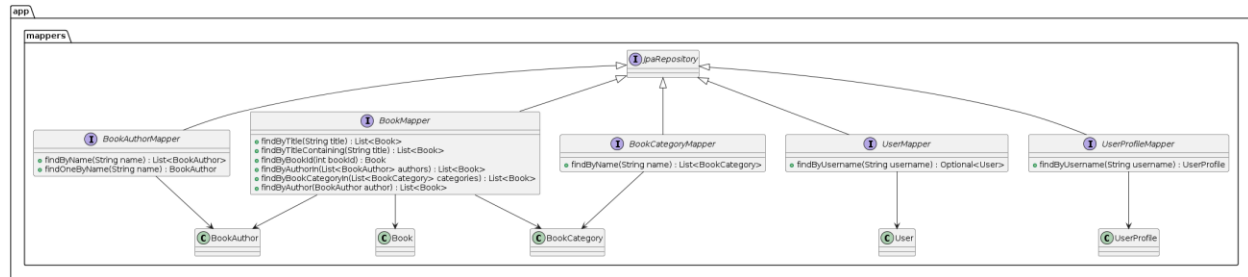
Config Package:



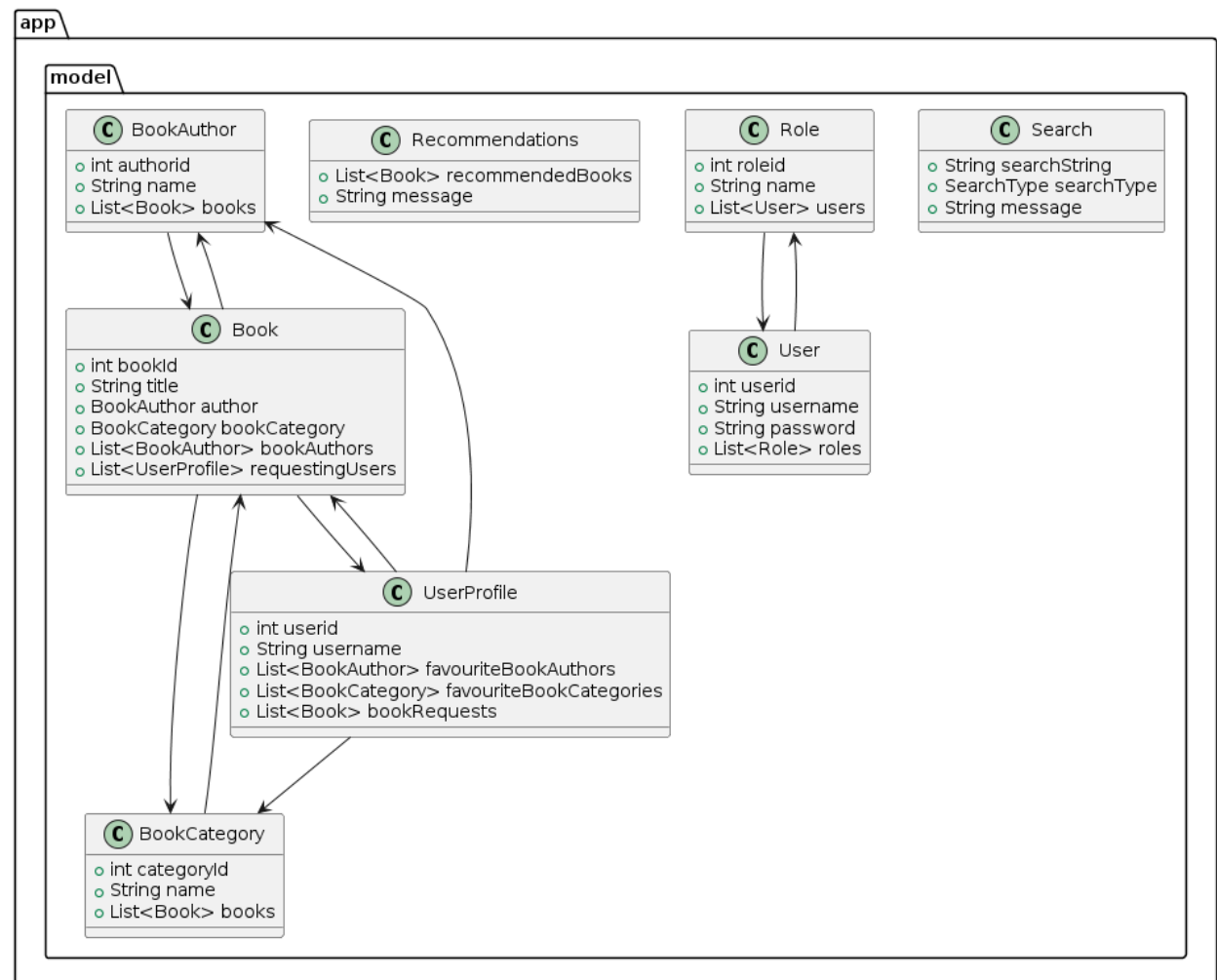
Controllers Package:



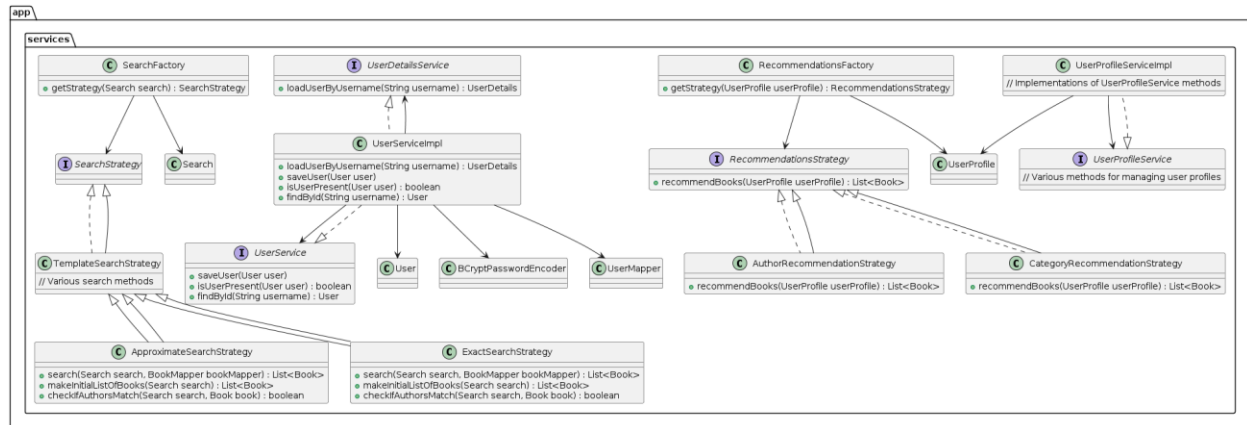
Mappers Package:



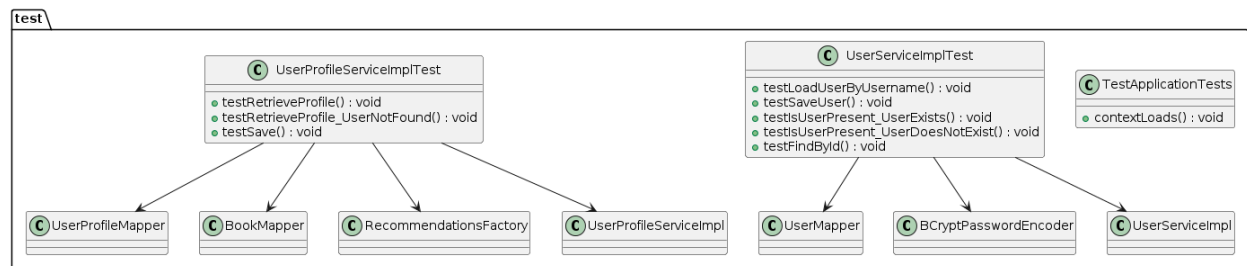
Model Package:



Services Package:



Test Package:



<Document the classes that are included in this release in terms of CRC cards according to the template that is given below.>

Class Name: BookAuthor	
Responsibilities: <ul style="list-style-type: none"> • Manage author ID. • Manage author name. • Manage books authored by the author. 	Collaborations: <ul style="list-style-type: none"> ▪ Collaborates with Book

Class Name: BookCategory	
Responsibilities: <ul style="list-style-type: none"> • Manage category ID. • Manage category name. • Manage books in the category. 	Collaborations: <ul style="list-style-type: none"> ▪ Collaborates with Book

Class Name: Recommendations

Responsibilities: <ul style="list-style-type: none"> • Manage recommended books. 	Collaborations: <ul style="list-style-type: none"> ▪ None
--	---

Class Name: Role	
Responsibilities: <ul style="list-style-type: none"> • Manage role ID. • Manage role name. • Manage users associated with the role. 	Collaborations: <ul style="list-style-type: none"> ▪ Collaborates with User

Class Name: Search	
Responsibilities: <ul style="list-style-type: none"> • Manage search string. • Manage search type. • Manage search messages. 	Collaborations: <ul style="list-style-type: none"> ▪ None

Class Name: User	
Responsibilities: <ul style="list-style-type: none"> • Manage user ID. • Manage username. • Manage password. • Manage roles associated with the user. 	Collaborations: <ul style="list-style-type: none"> ▪ Collaborates with Role

Class Name: UserProfile	
Responsibilities: <ul style="list-style-type: none"> • Manage user profile ID. • Manage username. • Manage favorite book authors. • Manage favorite book categories. • Manage book requests. 	Collaborations: <ul style="list-style-type: none"> ▪ Collaborates with Book ▪ Collaborates with BookAuthor ▪ Collaborates with BookCategory

Class Name: Book

Responsibilities: <ul style="list-style-type: none"> • Manage book ID. • Manage book title. • Manage author of the book. • Manage category of the book. • Manage authors of the book. • Manage users requesting the book. 	Collaborations: <ul style="list-style-type: none"> ▪ Collaborates with BookAuthor ▪ Collaborates with BookCategory ▪ Collaborates with UserProfile
--	--

Class Name: AuthController	
Responsibilities: <ul style="list-style-type: none"> • Handle login requests. • Handle registration requests. • Handle user registration. • Navigate to user main menu. 	Collaborations: <ul style="list-style-type: none"> ▪ Collaborates with User ▪ Collaborates with UserService ▪ Collaborates with UserProfileService ▪ Collaborates with UserProfile

Class Name: UserController	
Responsibilities: <ul style="list-style-type: none"> • Show user profile. • Search books. • View book requests. • Request book. • Cancel book request. • Show recommendations. 	Collaborations: <ul style="list-style-type: none"> ▪ Collaborates with UserProfileService ▪ Collaborates with Search ▪ Collaborates with Recommendations

Class Name: UserProfileServiceImplTest	
Responsibilities: <ul style="list-style-type: none"> • Test retrieval of user profile. • Test saving of user profile. • Test scenarios when user is not found. 	Collaborations: <ul style="list-style-type: none"> ▪ Collaborates with UserProfile ▪ Collaborates with UserProfileService

Class Name: UserServiceImplTest
--

Responsibilities:		Collaborations:	
<ul style="list-style-type: none"> • Test loading user by username. • Test saving user. • Test if user is present. • Test finding user by ID. 		<ul style="list-style-type: none"> ▪ Collaborates with UserService ▪ Collaborates with BCryptPasswordEncoder 	

Class Name: BookAuthorMapper			
Responsibilities:		Collaborations:	
<ul style="list-style-type: none"> • Find book authors by name. • Find a single book author by name. 		<ul style="list-style-type: none"> ▪ Collaborates with BookAuthor 	

Class Name: BookCategoryMapper			
Responsibilities:		Collaborations:	
<ul style="list-style-type: none"> • Find book categories by name. 		<ul style="list-style-type: none"> ▪ Collaborates with BookCategory 	

Class Name: BookMapper			
Responsibilities:		Collaborations:	
<ul style="list-style-type: none"> • Find books by title. • Find book by ID. • Find books by authors. • Find books by categories. • Find books by author. 		<ul style="list-style-type: none"> ▪ Collaborates with Book. ▪ Collaborates with BookAuthor. ▪ Collaborates with BookCategory. 	

Class Name: UserMapper			
Responsibilities:		Collaborations:	
<ul style="list-style-type: none"> • Find user by username. 		<ul style="list-style-type: none"> ▪ Collaborates with User 	

Class Name: UserProfileMapper			
--------------------------------------	--	--	--

Responsibilities: <ul style="list-style-type: none"> • Find user profile by username. 	Collaborations: <ul style="list-style-type: none"> ▪ Collaborates with UserProfile
---	--

Class Name: WebMvcConfig	
Responsibilities: <ul style="list-style-type: none"> • Add view controllers. 	Collaborations: <ul style="list-style-type: none"> ▪ Collaborates with WebMvcConfigurer. ▪ Collaborates with ViewControllerRegistry.

Class Name: WebSecurityConfig	
Responsibilities: <ul style="list-style-type: none"> • Configure web security settings. 	Collaborations: <ul style="list-style-type: none"> ▪ Collaborates with AuthenticationManager. ▪ Collaborates with DaoAuthProvider. ▪ Collaborates with AuthenticationConfiguration. ▪ Collaborates with HttpSecurity. ▪ Collaborates with UserDetailsService. ▪ Collaborates with BCryptPasswordEncoder. ▪ Collaborates with SecurityFilterChain. ▪ Collaborates with UserService.

Class Name: CustomSecuritySuccessHandler	
Responsibilities: <ul style="list-style-type: none"> • Handle successful authentication. • Determine target URL based on user role. 	Collaborations: <ul style="list-style-type: none"> ▪ Collaborates with Authentication. ▪ Collaborates with GrantedAuthority. ▪ Collaborates with RedirectStrategy.