Group 08: Super Idol

SIBOOKS WEB
Use-case Specification

Version 2.1

Revision History

Date	Version	Description	Author
12/07/2024	1.0	Formatting the report.	Lê Phước Phát
15/07/2024	1.1	Doing the following use cases: Sign Up, Sign In, Sign In with Google, Recover Password, Verify by Email, and Browse Book.	Lê Phước Phát
17/07/2024	1.2	Doing Checkout, Calculate Total, Calculate Shipping, Payment (Pay by Cash, Pay with Wallet/ QR Code).	Thái Huyễn Tùng
17/07/2024	1.3	Doing View Book Details, Leave Book Feedback, Cart Management (Add to Cart, View / Update Cart), and Make Purchases.	Ngô Văn Khải
18/07/2024	1.4	Doing the following use cases: Manage Accounts, Manage Book, Manage Orders, Analyze Revenue, Create Sale-off Event	Bùi Lê Khôi
19/07/2024	1.5	Doing Manage User Account, Update Order, Create Order, View Order Detail / Status / History, Admin Sign-In, Manage Notifications	Tô Quốc Thanh
20/07/2024	1.6	Checking all the wrong issues and fixing them. Formatting the final report and releasing it.	Lê Phước Phát
03/08/2024	2.0	Merge 6 use-cases Checkout, Calculate Total, Calculate Shipping, {abstract} Payment, Pay by cash and Pay with Wallet / QR code into 1 use-case: Make Payment.	Ngô Văn Khải
03/08/2024	2.1	Formatting and releasing the final ver 2.0 of the Use-case Specification.	Lê Phước Phát

Table of Contents

Use-case Specification

1. Use-case model

1.1 Use-case Diagrams

1.1.1 Customer
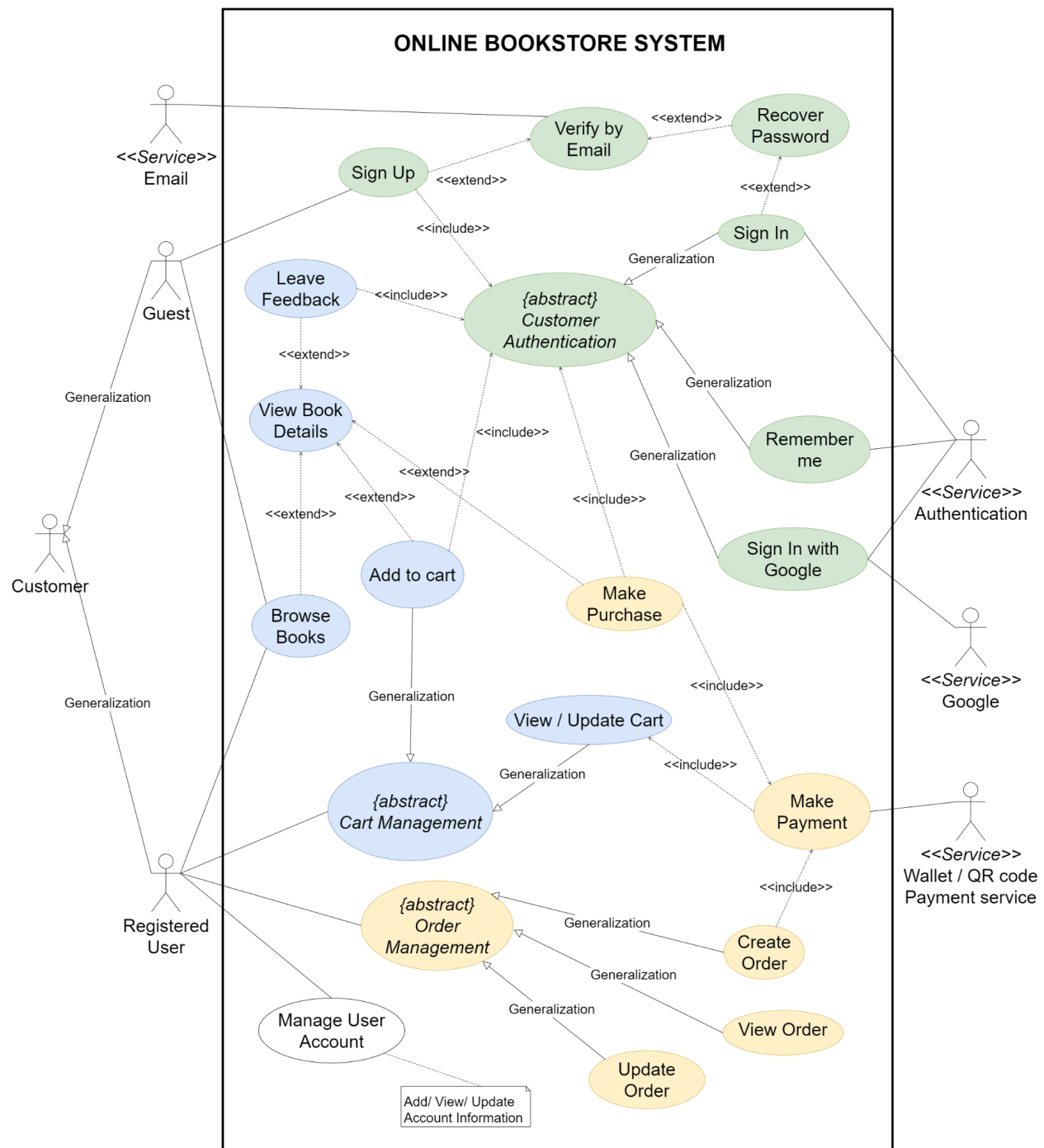
1.1.2 Admins
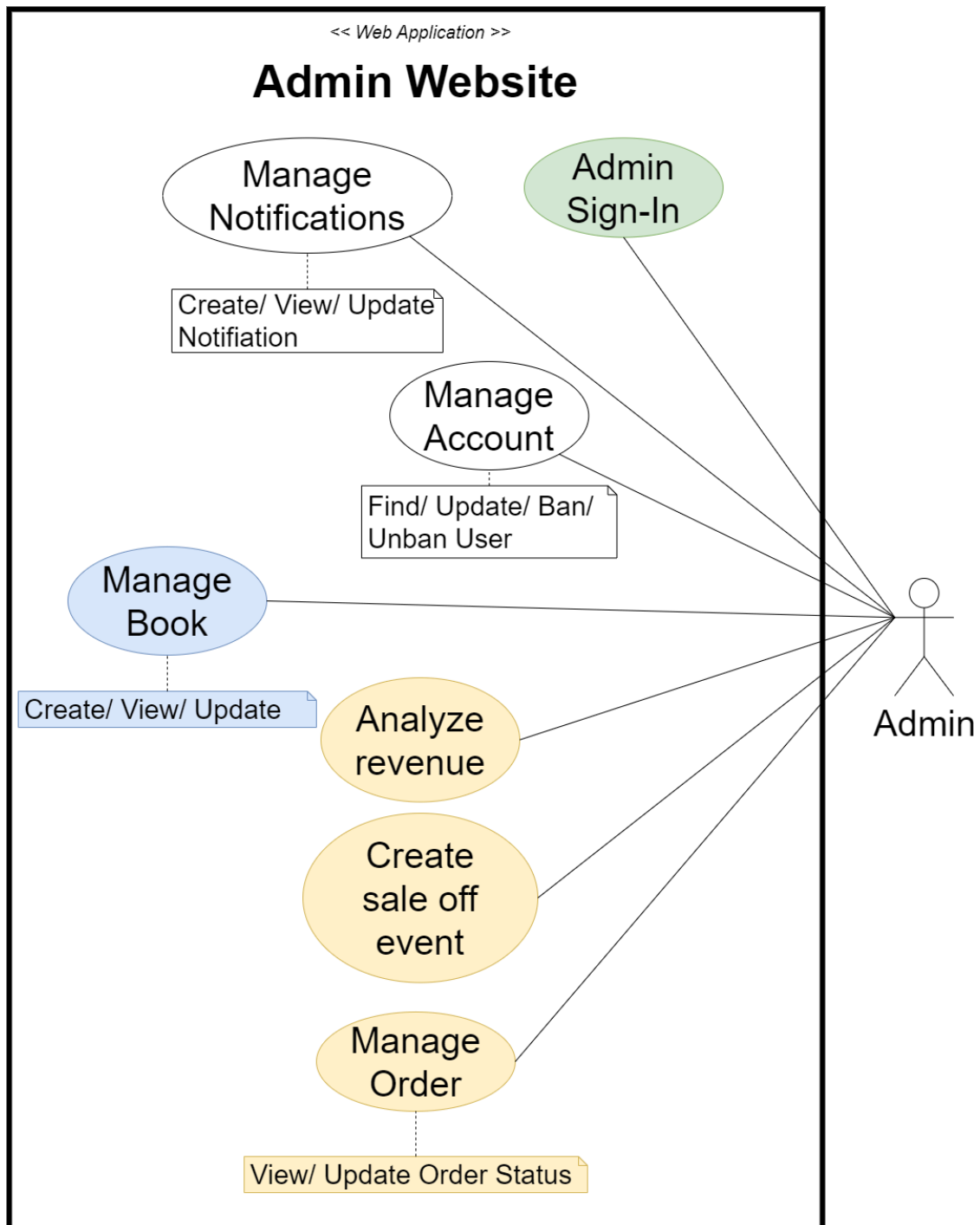
1.2 Actors

1.2.1 The list of the main actors

No.	Actors	Detailed description
1	Guests	Guests are customers who have not signed in to the website. They can sign up, browse the books, view books in detail, and view comments and ratings but can not buy, comment or rate them.
2	Registered Users	They are the customers who signed up for the account and inherit all the functionalities of the Guest, can sign in, recover passwords, sign in with Google, make purchases, leave feedback, manage their user account, view order details, and update orders.
3	Administrators	They will manage issues related to customers, directly resolve feedback that users have sent to the store, and provide effective support around difficulties that customers are facing

1.2.2 The list of the services

No.	Actors	Detailed description
4	<<Service>> Google	The **Google** service can help the customer sign in with a Google account to use all the services from the application. It integrates with Google's APIs to provide functionalities such as Google authentication.
7	<<Service>> Email	The **Email** service is responsible for sending and receiving emails. It is used for verifying the account after recovering the password account or signing up, and for various customer interactions that require email communication.
5	<<Service>> Authentication	The **Authentication** service manages user authentication, ensuring that only authorized users can access certain functionalities within the system. It handles user sign-in, sign-out, session management, cookie management, and user identity verification.
6	<<Service>> Wallet / QR code Payment service	This service facilitates digital payments through wallets or QR codes. It handles transactions, verifies payment information, and processes payments securely.

1.3 The list of use-case specification

No.	Use Case ID	Use Case Name
1	UC001	Sign Up
2	*UC002*	*<< Abstract>> Customer Authentication*
2.1	UC002_1	Sign In

2.2	UC002_2	Forgot Password
2.3	UC002_3	Remember me
2.4	UC002_4	Single Sign-On
3	UC003	Verify by Email
4	UC004	Browse Books
5	UC005	View Book Details
6	UC006	Leave Book's Feedback
7	*UC007*	*<<Abstract>> Cart Management*
7.1	UC007_1	Add to cart
7.2	UC007_2	View / Update Cart
8	UC008	Make Purchase
9	UC009	Make Payment
10	UC010	Manage User Account
11	*UC011*	*<<Abstract>> Order Management*
11.1	UC011_1	Update Order
11.2	UC011_2	Create Order
11.3	UC011_3	View Order Detail / Status / History
12	UC012	Admin Sign-In
13	UC013	Manage Notifications
14	UC014	Manage Account
15	UC015	Manage Book
16	UC016	Analyze revenue
17	UC017	Create sale off event
18	UC018	Manage Order

2.   Use-case specification

2.1   Use-case Sign Up

Use Case ID	UC001
Use Case Name	Use-case Sign Up
Brief Description	When we are the **guests**, we want to sign up to create a new account in the system to use more of the services from the application and store our data.
Actors	Guests
Pre-conditions	The user does not have an account in the system. The user accounts are unauthenticated.
Post-conditions	The user account is created and stored in the system. The user is automatically logged in. **Triggering related events:** ● Sending a welcome email
Basic Flows	1. The guest navigates to the **"Sign In"** page from the website's **"Home"** page or other relevant links. 2. The system would display a register form requiring information: ● Username ● Email ● Password ● Password confirmation 3. The guest would fill in the form and click the **"Sign Up"** button. ● The guest enters the required information and submits the form by clicking the **"Sign Up"** button. 4. The system would check the information: ● The username has not been used. ● The email has not been used ● The password and password confirmation match. 5. The system will store data about the new account information in the database. 6. The system will notify and display the notification about the user of a successful registration. 7. The system sends an account verification email to the user. 8. The user is automatically signed in and moved to the **"Home"** page.
Alternative Flows	**4.1. Email has already been used** ● The system detects that the email has already been used. ● The system displays an error dialogue and asks the user to enter a different email. **4.2. The username has already been used** ● The system detects that the username has already been used. ● The system displays an error dialogue and asks the user to enter a different username. **4.3. Password and password confirmation do not match** ● The system detects that the password and password confirmation do

	not match. ● The system displays an error dialogue and asks the user to re-enter the password.
Special Requirements (Nonfunctional Requirements)	● The system must handle up to 10000 concurrent sign-up requests. ● The sign-up process should not take longer than 5 seconds. ● Passwords must be stored using strong encryption algorithms.
Extension Points	Verify by Email

2.2 <<Abstract>> Customer Authentication

2.2.1 Use-case Sign-In

Use Case ID	**UC002_1**
Use Case Name	Use-case Sign-In
Brief Description	When we are **registered users**, we want to sign in to the system using a registered account with a username and password to use all the services from the website app.
Actors	Registered User
Pre-conditions	The user already has an account in the system with a username, password, and email. The user account has not been authorized. The user's device is connected to the Internet when signing in.
Post-conditions	The user is successfully signed in to the system, and their authentication status is marked as "authenticated". The system would display a welcome message (e.g., **"Welcome users, [Username] !!!"**) indicating successful login. A user session is created and maintained for the duration of the user's activity on the site, ensuring the user remains logged in until they log out or the session expires. The users are redirected to the **"Home"** page displaying the recommendations, website content, …
Basic Flows	1. The user navigates to the **"Sign In"** page from the website's **"Home"** page. 2. The user would choose the Sign-In form with the registered account. 3. The system would display the Sign- In form requiring information: 　● Username 　● Password 4. The user would fill in the username and password fields on the Sign-In form and click the "Sign In" button. 5. The system would validate the credentials 　● The system checks the entered username and password

	against the stored user data.
	● The system confirms the username is registered and the password matches the stored password for that registered account.
	6. Upon successful validation, the system logs the user in and sets a session for the user.
	7. The system redirects the user to the "Home" page or the last accessed page before signing out.
Alternative Flows	**6.1. If the credentials are incorrect:** ● The system detects that the username or password is incorrect. ● The system displays an error message indicating the invalid credentials. ● The user is prompted to re-enter the username and password. **6.2. If the user selects "Remember me":** ● The system would move to the use case **"Remember Me".** **6.3. If the user clicks on the "Sign-In with Google" button:** ● The system would move to the use case **"Single Sign-On".**
Special Requirements (Nonfunctional Requirements)	● The system must handle up to 10000 concurrent sign-in requests. ● The sign-in process should not take longer than 3 seconds. ● The system should implement strong security measures to protect user credentials, including encryption and secure transmission of data. ● Persistent login cookies should have a reasonable expiration period and be stored securely.
Extension Points	Forgot Password Remember me Sign In with Google

2.2.2 Use-case Recover Password

Use Case ID	UC002_2
Use Case Name	Use-case Recover Password
Brief Description	When we are **registered users**, we will recover the password account to continue to use all the services from the application because we forgot the password account.
Actors	Registered user
Pre-conditions	The user must have an available registered account. The user has forgotten their password. The user has a registered email address in the system.
Post-conditions	The user receives a password code on the email reset link. The user successfully resets the password account.
Basic Flows	1. The user accesses the **"Forgot Password"** page.

	2. The user enters their registered email address by filling in the email address field on the "Forgot Password" form.. 3. The System validates the email. • The system checks if the entered email address is registered. • The system confirms that the email address exists in the user database. 4. The system sends a password reset link to the email. • Upon successful validation, the system generates a unique, secure password reset link. • The system sends an email containing the password reset link to the user's registered email address. 5. The user follows the link to reset the password. • The user receives the email and clicks on the password reset link. • The link directs the user to a secure page to enter and confirm a new password. • The user fills in the new password fields and submits the form. 6. The system updates the password. • The system validates the new password (ensuring it meets security requirements). • The system updates the user's account with the new password. • The system notifies the user of a successful password reset and prompts them to log in with the new password.
Alternative Flows	**3.1. If the email is not registered:** • The system detects that the entered email address is not registered. • The system displays an error message indicating that the email is not found in the system. • The user is prompted to re-enter a registered email address or contact support for further assistance. **5.1. If the reset link is expired or invalid:** • The user attempts to use an expired or invalid password reset link. • The system displays an error message indicating that the link is expired or invalid. • The user is prompted to request a new password reset link by repeating the Forgot Password process.
Special Requirements (Nonfunctional Requirements)	• The system must handle up to 500 concurrent password reset requests. • The password reset email should be sent within 1 minute of the request. • The password reset link should be valid for a limited time (e.g., 24 hours) to enhance security. • The password reset process should comply with security best practices, including HTTPS for secure communication and strong password requirements.
Extension Points	Verify by Email

2.2.3 Use-case Remember me

Use Case ID	UC002_3
Use Case Name	Use-case Remember me
Brief Description	This use case helps to keep the user signed in across sessions.
Actors	Registered user
Pre-conditions	The user must access the "Sign In" page. The user selects the "Remember me" option during sign-in. The user's browser accepts cookies.
Post-conditions	The user remains signed in on subsequent visits to the website.

Basic Flows	1. The user checks the "Remember me" checkbox on the Sign-In form
	2. The system sets a persistent login cookie.
	    ● The system authenticates the user's credentials.
	    ● Upon successful authentication, the system generates a persistent login token.
	    ● The system sets a persistent login cookie in the user's browser with the login token and expiration date.
	    ● The cookie ensures that the user's session persists across browser restarts and visits to the site.
	3. The user remains signed in on subsequent visits.
	    ● When the user revisits the site, the system detects the persistent login cookie.
	    ● The system validates the login token in the cookie.
	    ● If the token is valid, the system logs the user in automatically without requiring them to enter credentials.
	    ● The user is redirected to the home page or their last accessed page.

Alternative Flows	**2.1. If cookies are disabled:**
	    ● The system attempts to set the persistent login cookie but detects that cookies are disabled in the user's browser.
	    ● The system displays a notification to the user, informing them that the "Remember me" feature is unavailable due to disabled cookies.
	    ● The user is advised to enable cookies in their browser settings to use the "Remember me" feature.
	3.1. If the login token is invalid or expired:
	    ● The system detects that the persistent login token is invalid or expired during a subsequent visit.
	    ● The system does not automatically log the user in.
	    ● The system prompts the user to sign in manually.

Special Requirements (Nonfunctional Requirements)	● The persistent login cookie should be encrypted and secure to prevent unauthorized access. ● The login token should have a reasonable expiration period (e.g., 30 days). ● The system should handle up to 1000 concurrent persistent login

	sessions.
	● The system should comply with data protection regulations regarding cookie usage and user consent.
Extension Points	None

2.2.4 Use-case Sign-In with Google

Use Case ID	UC002_4
Use Case Name	Use-case Sign In with Google
Brief Description	This use case allows a user to sign in using an external service which is Google service.
Actors	Guest, Registered user, Google;
Pre-conditions	The user has an account with Google service, which means that is an external service. The bookstore system is integrated with the external service for Single Sign-On (SSO).
Post-conditions	The user is successfully logged in to the bookstore system.

Basic Flows	1. The user accesses the "Sign In" Page and clicks the "Sign In with Google".
	 2. The system redirects the user to the external service's authentication page which is the Google Sign-In Page.
	 3. The user provides their credentials (email and password) for signing in with Google service.
	 4. External service authenticates the user and redirects back to the bookstore system.
	     ● The external service validates the user's credentials.
	     ● Upon successful authentication, the external service generates an authentication token and redirects the user back to the bookstore system with the token.
	 5. The system logs the user in.
	     ● The bookstore system receives the authentication token.
	     ● The system validates the token with the external service.
	     ● Upon successful validation, the system logs the user in and establishes a session.
	     ● The user is redirected to the home page or the last accessed page before signing out.

Alternative Flows	**4.1. If the external service authentication fails:** ● The user enters incorrect credentials or there is an issue with the external service. ● The external service fails to authenticate the user and displays an error message. ● The user is prompted to try again or use a different login method. ● The system redirects the user back to the bookstore system's Sign-In page with an error notification.
Special Requirements (Nonfunctional Requirements)	● The system must handle up to 500 concurrent Single Sign-On requests. ● The Sign-In with Google process should not take longer than 5 seconds. ● The integration with the external service should comply with security best practices, including HTTPS for secure communication and OAuth2 for token-based authentication. ● The system should provide clear error messages and guidance for users if the Sign-In with Account process fails.
Extension Points	None

2.3   Use-case Verify by Email

Use Case ID	**UC003**
Use Case Name	Use-case Verify by Email
Brief Description	When we are registered users, we will verify the account by Email address after signing up for a new account or recovering the password, ensuring that the email provided during registration is valid and that the user can be authenticated.
Actors	Registered user, Email service
Pre-conditions	The user has completed the sign-up process or the recovered password process and provided a valid email address. The system has generated a verification link and sent it to the provided email address.
Post-conditions	The user's email address is verified. The user's account is activated, allowing them to access features restricted to verified users. The password code will be sent from the email user.
Basic Flows	1. The user completes the sign-up process by providing a valid email address. 2. The system generates a verification link containing a unique token. 3. The system sends an email with the verification link to the user's email address. 4. The user receives the email and clicks on the verification link. 5. The system verifies the token in the verification link.

	6. If the token is valid, the system updates the user's status to verified. 7. The user receives a confirmation message indicating that their email has been verified and their account is now active.
Alternative Flows	**4.1 If the user does not receive the verification email:** ● The user checks their spam/junk folder and verifies that the email address provided is correct. ● If the email is still not received, the user requests the system to resend the verification email. ● The system sends a new verification email to the user. **6.1. If the token is invalid or expired when the user clicks on the verification link** ● The system displays an error message indicating that the token is invalid or expired. ● The user is prompted to request a new verification email.
Special Requirements (Nonfunctional Requirements)	● The verification link should be secure and contain a token that expires after a certain period. ● The verification email should be clear and contain instructions on what the user needs to do to verify their email. ● The email service should ensure that verification emails are sent promptly and reliably. ● The system should handle multiple verification requests without significant delay.
Extension Points	Resend Verification Email Customer Authentication

2.4 Use-case Browse Books

Use Case ID	UC004
Use Case Name	Use-case Browse Books
Brief Description	When we are guests and registered users, we want to browse through the list of available books for searching, listing, viewing, and filtering the detailed books.
Actors	Guests, Registered Users
Pre-conditions	The user wants to visit the website.
Post-conditions	The user can browse books, search a book by name, and filter the page to find the book he/she wants to buy.

Basic Flows	1. The user navigates to the **"Shop Books"** page from the **"Home"** page.
	2. The system would display a list of available books.
	   ● The system retrieves the list of available books from the database.
	   ● The system displays the list of books with relevant details

	such as title, author, genre, price, inventory quantity, and a brief description.
	3. The users can filter and sort the list by various criteria:
	● The user can apply filters to narrow down the list based on criteria:
	○ Categories
	○ Author
	○ Daily sale / Deal (percentage of sale off)
	○ Rating
	○ Price range
	○ Publisher
	○ Publication date
	● The user can sort the list by criteria such as popularity, best selling new book, rating, price (increasing or decreasing order), and publication date.
	● The system would dynamically update the displayed list based on the user's selected filters and sort options.
	4. The user can search for books by keywords
	● The user enters keywords such as book title, author, and ISBN into the search bar.
	● The system processes the search query and retrieves matching results from the database.
	● The system displays the search results and highlights the matched keywords.
Alternative Flows	**2.1. If no books are available:**
	● The system detects that there are no available books in the database.
	● The system displays a notification informing the user that no books match the selected criteria.
	● The user is prompted to adjust the filters to broaden the search.
	3.1. If the user applies filters that result in no matches:
	● The system detects that the applied filters result in no matching books.
	● The system displays a notification informing the user that no books match the selected criteria.
	● The user is prompted to adjust the filters to broaden the search.
	4.1. If the search query returns no results:
	● The system detects that the search query did not match any books in the database.
	● The system displays a notification informing the user that no books were found for the search query.
	● The user is prompted to adjust the search keywords or try different terms.
Special Requirements (Nonfunctional Requirements)	● The system must handle up to 10000 users accessing at the same time.
	● The system has to guarantee the security of the private information of registered users.
	● The book list and filtering/sorting/searching operations should be responsive and load within 3 seconds.
	● The user interface should be intuitive and user-friendly to enhance the browsing and searching experience.
	● The system should ensure that book data is kept up-to-date and

	accurate.
Extension Points	None

2.5 Use-case View Book Details

Use Case ID	UC005
Use Case Name	Use-case View Book Details
Brief Description	The user can know a book's information such as name, price, author, publisher and some photos about the book.
Actors	Guests, Registered Users
Pre-conditions	The user is entering the website. The user has clicked on a book's cover photo while browsing books.
Post-conditions	The user can see the book's information: name, author, translator, category, ISBN, publishing company, publishing date, price, language, size, the number of pages, cover type, description and some photos about the book. They can also know the book's rating and read comments from other user(s).
Basic Flows	1. The user browses books. 2. The user visits a page of a book
Alternative Flows	1. If the user clicks on the author name • The website displays a list of books written by that author (Use-case Browse Books). 2. If the user clicks on the translator name • The system displays a list of books translated by that translator (Use-case Browse Books). 3. If the user clicks on the publishing company name • The system displays a list of books published by that publishing company (Use-case Browse Books).
Special Requirements (Nonfunctional Requirements)	• The system must handle up to 10000 users accessing at the same time. • The system has to guarantee the security of the private information of registered users.
Extension Points	None

2.6 Use-case Leave Book's Feedback

Use Case ID	UC006
Use Case Name	Use-case Leave Book's Feedback

Brief Description	Registered users can comment or rate a book after buying it.
Actors	Registered Users
Pre-conditions	The user's account has to be authenticated. This registered user has bought and received the book.
Post-conditions	The user successfully posts a comment and/or rating the book from 1 to 5.
Basic Flows	1. The user has signed in. 2. The user browses books. 3. The user visits a page of a book which he/she has bought before. 4. The user can rate the book from 1 to 5 and/or post a comment about the book. The user can write something and add a photo(s) in the comment.
Alternative Flows	1. If the user has not signed in when leaving feedback ● The system displays an error message indicating that the user should sign in or create an account. 2. If the user has not bought the book ● The system displays a warning message indicating that the user cannot leave the book's feedback.
Special Requirements (Nonfunctional Requirements)	● The system has to guarantee the security of the private information of users.
Extension Points	None

2.7   <<Abstract>> Cart Management

2.7.1   Use-case Add to cart

Use Case ID	UC007_1
Use Case Name	Use-case Add to cart
Brief Description	This use case allows the registered user to add a book to the cart list.
Actors	Registered Users
Pre-conditions	The user's account has to be authenticated.
Post-conditions	The user successfully
Basic Flows	1. The user has signed in. 2. The user browses books. 3. The user visits a page of a book. 4. The user inputs the number to the 'Quantity' box (1 is the default number). Then, click the 'Add to Cart' button, and the book will be added to the personal cart with its quantity.
Alternative Flows	1. If the user has not signed in

	● The system displays an error message indicating that the user should sign in or create an account
Special Requirements (Nonfunctional Requirements)	● The system has to guarantee the security of the private information of users.
Extension Points	None

2.7.2 Use-case View / Update Cart

Use Case ID	UC007_2
Use Case Name	Use-case View / Update Cart
Brief Description	This use case allows the registered user to know which book has been added to the cart list and change the number of book(s).
Actors	Registered Users
Pre-conditions	The user's account has to be authenticated.
Post-conditions	Some books might be removed from the cart. The number of books might be changed.
Basic Flows	1. The user has signed in. 2. The user accesses the personal cart. 3. The user can change the number of any book in the cart. If any number is decreased to 0, that book will be removed from the cart list. 4. The user clicks the 'Update' button

Alternative Flows	1. If the user has not signed in
	  ● The system displays an error message indicating that the user should sign in or create an account.
	2. If the user didn't click the 'Update' button
	  ● The system keeps the current data

Special Requirements (Nonfunctional Requirements)	● The system has to guarantee the security of the private information of users.
Extension Points	None

2.8 Use-case Make Purchase

Use Case ID	UC008
Use Case Name	Use-case Make Purchase
Brief Description	The registered user wants to buy a book while viewing it.

Actors	Registered Users
Pre-conditions	The user's account has to be authenticated.
Post-conditions	The system considers that the user wants to checkout.
Basic Flows	1. The user has signed in. 2. The user browses books. 3. The user visits a page of a book. 4. The user inputs the number to the 'Quantity' box (1 is the default number). Then, click the 'Buy Now' button.
Alternative Flows	1. If the user has not signed in ● The system displays an error message indicating that the user should sign in or create an account.
Special Requirements (Nonfunctional Requirements)	● The system has to guarantee the security of the private information of users.
Extension Points	None

2.9   Use-case Make Payment

Use Case ID	**UC009**
Use Case Name	Use-case Make Payment
Brief Description	Users can pay for a book they want to buy or all books selected in the cart. Users can check out their bills.
Actors	Registered users
Pre-conditions	The user's account has to be authenticated. The user's cart is not empty. Complete the options when checking out: choosing payment methods, and delivery methods, and filling in the necessary information.
Post-conditions	Complete paying and display the successful message. Then, users can view their orders.
Basic Flows	1. User accesses a personal cart. 2. The system displays his/ her products and vouchers in the cart. 3. The user chooses one voucher and applies it to the bill. 4. The system displays short information about the amount of money before and after applying the voucher. 5. The user presses the checkout button to check out the bill. 6. The system displays a table that contains the necessary information for the order. 7. The user has to choose the address, payment method, and delivery method 7.1. The user chooses the address:

	● System displays the current address (user's personal address) and previous addresses. ● User chooses to add a new address button, the system will display a table where the user can select a city, district, or town or enter a specific address (house number, road name). 7.2. User chooses payment method: ● The system will display 2 options: cash on delivery and bank account. ● User chooses cash on delivery, he/ she can pay when receiving the order. ● The user chooses a bank account, he/ she can pay through a mobile wallet. 7.3. User chooses delivery method: ● System displays 3 types of delivery: standard delivery, quick delivery, and super fast delivery. ● User chooses one of three methods above. ● System adds delivery fees based on those types.. 8. System displays detailed information about the old price, applied voucher, delivery fee and total price. 9. The user presses the buy now button. 10. The system validates information and displays buy products successfully on the screen.
Alternative Flows	Step 7 is violated: ● User misses one of these selections, he/ she can not complete the checkout for the bill. Step 10 is violated: ● User's information is invalid, the system will refuse to complete the checkout step and send an error message to the user.
Special Requirements (Nonfunctional Requirements)	The system should ensure security during the checkout process for the payment method section. The system also ensures availability when there are a lot of users accessing at the same time to check out their bills.
Extension Points	None

2.10 Use-case Manage User Account

Use Case ID	UC010
Use Case Name	Use-case Manage User Account
Brief Description	Admin can view, update user accounts or ban him from the system for a defined time.
Actors	Admin.
Pre-conditions	Admin already Logged in.

	To view/ update/ ban a user, he must exist in the system.
Post-conditions	User accounts are successfully updated and banned. The system stores all changes to the database. **Trigger events:** ● Display a pop-up to notify that the action is successful or a failure ● Send a notification to the user about the actions of the admin
Basic Flows	The system displays the user account management interface. **View User Account:** 1. Admin selects a user account to view. 2. The system displays the detailed information of the selected user account. **Update User Account:** 1. Admin selects a user account to update. 2. The system displays the current details of the selected user account. 3. Admin changes the user details and submits the changes. 4. The system validates the changes and updates the user account. 5. The system sends a notification to the user about the account update. **Ban User Account:** 1. Admin selects a user account to ban. 2. Admin select time to ban and select the reason to ban the user. 3. The system change the user account status to 'banned' 4. The system sends a notification to the user about the account banned..
Alternative Flows	**Invalid Data Entry:** 1. Admin enters invalid information when updating a user account. 2. The system displays error messages . 3. Admin corrects the information and resubmits the form. 4. If the information is still invalid, the system displays a failure popup and does not proceed with the action. **Cancel Action:** 1. Admin decides to cancel the current action (view, update, ban). 2. Admin selects the cancel option. 3. The system discards the changes and returns to the user account management page.
Special Requirements (Nonfunctional Requirements)	**Security:**

	● Only authenticated and authorized admins can access and manage user accounts. ● Some of importance information of user as password, Wallet password,.... should not be show
Extension Points	**Password Reset:** ● Admin can create a reset link for users through email.

2.11   <<Abstract>> Order Management

2.11.1 Use-case Update Order

Use Case ID	UC011_1
Use Case Name	Use-case Update Order
Brief Description	Users can update the shipping address, phone number, and delivery times, or cancel the Order before it is shipped.
Actors	Registered User
Pre-conditions	The user account is authenticated. The order is created. The order is not on delivering or done or canceled (the order's status is not in 'delivering', 'done', or 'canceled').
Post-conditions	Update information of Order. **Triggering related events:** ● An email or notification is sent to the admin informing them of the order's change. ● The user is notified that their order has been updated successfully.
Basic Flows	1. The user accesses the "View Order" page. 2. For each order, the system displays an icon for users to edit and update order information. 3. The user is allowed to edit details such as the recipient's name, phone number, shipping address, etc., in edit mode or cancel the Order. 4. The user adjusts the information as desired. 5. The user clicks the "Update" button and confirms to save the updates. 6. The system saves the new order information.

	7. System notifies both the user and the admin about the changes. 8. System will display buttons for the user to move to the "Home" page or "View Order" page

Alternative Flows	2.1 Not allow to update Order
	If the order status is 'delivering', 'done', or 'canceled', the system will not display the button for the user to update the Order.
	
	3.1 Cancel Order
	If the user selects 'Cancel Order', the Order will be canceled. The system will send a notification to the admin. If the user paid for the Order by wallet, the system will refund for user.
	
	5.1 Do not change/ update Order
	If the user does not click the "Update" button or confirm, he will be able to keep the current data.
	
	8.1 User clicks on the **"Home"** button:
	   ● The system navigates to the "Home" page, and the user can continue shopping.
	
	8.2 User clicks on "View Order" button:
	   ● System navigates to the "View Order" page to see the order's details and status.

Special Requirements (Nonfunctional Requirements)	The system needs to automatically validate the validity of newly updated information. The editable information should be prominently displayed to make it easy for users to see, for example, by outlining it within the input box. If the system refunds the user, it needs to check that the order is validated, not a virtual order created by a hacker.
Extension Points	None

2.11.2 Use-case Create Order

Use Case ID	UC011_2
Use Case Name	Use-case Create Order
Brief Description	Users can create a new order after checkout and send the order to admin.
Actors	Registered User
Pre-conditions	The user account is authenticated. Check out step does not appear as any error as product quantity is not enough, fail on payment.
Post-conditions	A new order is created and stored in the system, and the order status is set to

	"Pending Approval". Update the cart of the user, and pop products which are in the order out of the cart. **Triggering related events:** ● An email or notification is sent to the admin informing them of the new order awaiting approval. ● The user is notified that their order has been created and is pending approval.
Basic Flows	1. The user would access his cart to view products there because this UC includes Check Out UC, and it includes View Cart UC. 2. On this page, the system displays information about product data in the cart such as total amount, payment method,... and a button named "Create Order". 3. The user would click on the button "Create Order", if Check Out UC can finish without any error, the system will create a new order and display a pop-up to notify that the order was created. 4. System store order to data and send it to admin for approval. 5. System update user's cart (pop out product which was added to Order). 6. System will display buttons for the user to move to the "Home" page or "View Order" page
Alternative Flows	3.1 Check Out UC appear error: Some errors could appear: ● Product quantity is not enough. ● Fail on payment. ● Can not find shipping company If the error appears, the user needs to update his cart to avoid it unless he can not create a new order 6.1 User clicks on the **"Home"** button: System navigates to the "Home" page, and the user can continue shopping. 6.2 User clicks on the "View Order" button: The system navigates to the "View Order" page to see the order's details and status.
Special Requirements (Nonfunctional Requirements)	In stable internet conditions (ping < 30ms), the time to create an order and notify the user must not exceed 5 seconds. Information about the newly created order needs to be kept confidential, and visible only to the admin and the user.
Extension Points	None.

2.11.3 Use-case View Order

Use Case ID	UC011_3
Use Case Name	Use-case View Order Detail / Status / History.
Brief Description	The user can view his order containing details, the status of the current Order and all orders that he made.
Actors	Registered User.
Pre-conditions	The user account is authenticated. Users access the "View Order" page.
Post-conditions	The system displays recent orders of the current users. The user can view the detailed information of each order, including items, prices, quantities, and delivery details. The user can filter or search through their order history.
Basic Flows	1. The user navigates to the "View Order" page. 2. The system displays a list of orders associated with the user, including details and current status for each order. 3. The user selects a specific order to view its detailed information. 4. The system shows detailed information such as order items, shipping details, payment status, and the current status of the order.
Alternative Flows	2.1 View Order History If the user continues to scroll down instead of selecting a specific order, the system will show his older orders. 4.1 View details of the product in Order If the user selects the icon of a product in Order, the system will navigate to the page of that product. If the product no longer exists, display a pop-up to the user
Special Requirements (Nonfunctional Requirements)	The system should ensure secure access to order information, visible only to authenticated users. Response time for loading order details should be optimized to provide a seamless user experience. The interface should be intuitive and user-friendly, facilitating easy navigation and information retrieval.
Extension Points	None

2.12 Use-case Admin Sign-In

Use Case ID	UC012
Use Case Name	Use-case Admin Sign-In

Brief Description	Admin can sign in to the system using a provided account with a username and password to manage all the services from the website app.
Actors	Admin
Pre-conditions	The admin already has an account in the system. The admin account has not been authorized. The admin access to the "Login" page for Admin.
Post-conditions	The admin is successfully logged in and can access admin functionalities. The system records the login time and other relevant details. The system provides access to the admin dashboard and management tools. **Trigger event:** ● System displays a pop-up to notify that logged in successfully
Basic Flows	1. System display login form 2. Admin will input name and password 3. The system validates the input of admin 4. If the input is correct, the system grants access. 5. System navigate admin to Dashboard
Alternative Flows	**Invalid Credentials:** 1. Admin enters incorrect username or password. 2. The system displays an error message. 3. Admin retries entering the correct input or opts to reset the password.
Special Requirements (Nonfunctional Requirements)	**Performance:** ● The system should authenticate the admin within 2 seconds. **Usability:** ● The login page should be able to be accessed from many kinds of devices. ● Error messages should be clear and helpful.
Extension Points	**Expire Time:** ● System automatically logs out after a defined time to increase the security.

2.13   Use-case Manage Notifications

Use Case ID	UC013
Use Case Name	Use-case Manage Notifications
Brief Description	Admin can send notification to the user about his order status or flash sale event in coming. If some information is incorrect, he can update the notification.
Actors	Admin
Pre-conditions	Admin already Logged in. To view/ update a notification, it must exist in the system. Users selected to send notifications already existed in the system.
Post-conditions	Notifications are sent to the specific user or all users. Notification details can be updated if needed. The system logs the notification activity. **Trigger event:** ● System displays a pop-up to notify that notification has been sent/ updated successfully
Basic Flows	1. The system displays the notification management interface. 2. The Admin will generate a new notification, and select the user to send or update a notification. 3. The system processes the notification and sends it to the specified users. 4. The system provides a confirmation message to the admin. 5. The system will navigate admin back to the Dashboard.
Alternative Flows	**Update Notification:** 1. Admin selects an existing notification to update. 2. The system displays the current details of the selected notification. 3. Admin changes the details of the notification. 4. Admin saves the changes. 5. The system updates the notification and sends the revised notification to the users. **Cancel Notification:** 1. Admin decides to cancel a notification before it is sent. 2. Admin selects the notification and cancels it. 3. The system removes the notification from the queue and confirms the cancellation.

Special Requirements (Nonfunctional Requirements)	Only authenticated and authorized admins can access and manage notifications. The system should be able to send notifications to multiple users without delays.
Extension Points	**Notification Templates:** ● The system should provide some templates for common notifications (e.g., order status updates, and flash sale announcements).

2.14 Use-case Manage Account

Use Case ID	**UC014**
Use Case Name	Use-case Find Account
Brief Description	Use-case allows the admin to find accounts created by users
Actors	Admin
Pre-conditions	Admin is logged in.
Post-conditions	The target account, its information and its status will be displayed and updated, or there will be a notification that the account does not exist.

Basic Flows	1. Admin selects the Manage User tab.
	2. Admin selects the search bar and searches the user account by given information (username, userID, profileName, …)
	3. The account information will be displayed for further action.
	4. Admin can have further actions to the account:
	  4.1. Update information:
	    - Admin chooses the specific information field(s).
	    - Admin changes the data.
	    - Admin presses the Save button to save the new information.
	  4.2. Change the account status:
	    - Admin chooses the account status field.
	    - Switch the status from unbanned to banned, or banned to unbanned.
	    - Admin presses the Save button to save the new information.
	  4.3. Admin presses the Close button to close the window.

Alternative Flows	3.1. If the account to be searched for does not exist, a notification about it will be displayed. 4.1.1. Admin cancel the process by pressing the Cancel button. 4.2.1. Admin cancel the process by pressing the Cancel button.
Special Requirements	- In good conditions (stable networks, the device operates normally),

(Nonfunctional Requirements)	the time to return search results should not exceed 5 seconds.
Extension Points	None

2.15 Use-case Manage Book

Use Case ID	**UC015**
Use Case Name	Use-case Manage Book
Brief Description	Admin can add new books to the database, and manage books' information and status.
Actors	Admin
Pre-conditions	Admin is logged in.
Post-conditions	A new book with its information will be added to the database. The chosen book's information will be displayed. The chosen book's information will be updated.

Basic Flows	1. Admin selects the Book tab.
	2. Admin can have further actions:
	   2.1. Admin selects the Add button to add new book:
	      - Admin fills information of the book (ISBN, Book title, publisher, etc.) in the corresponding fields.
	      - Click the Save button to save the data, or click the Cancel button to cancel the import process.
	      - A pop-up notification will be displayed, indicating that the data was imported successfully.
	   2.2. Admin has the option to find a book:
	      - Admin selects the search bar and searches the book by given information (ISBN/ Book title)
	      - System will display the results from the given information.
	   2.2.a. Admin can click on the Update button of a book to update its information:
	      + Admin chooses the specific information field(s).
	      + Admin changes the data.
	      + Admin presses the Save button to save the new information.
	      - Admin press the Close button to close the result window.

Alternative Flows	2.1.1. If the book is already in the database (based on ISBN) then a pop-up caution will be displayed. 2.2.1. If the book does not exist within the given information then a pop-up caution will be displayed. 2.2.a.1. Admin cancels the process by pressing the Cancel button.
Special Requirements (Nonfunctional Requirements)	- In good conditions (stable networks, the device operates normally), the time to return search results should not exceed 5 seconds. - 2.2.2: The system should be able to display all close results if

	possible, for example, when admin inputs a book title.
Extension Points	None

2.16   Use-case Analyze Revenue

Use Case ID	UC016
Use Case Name	Use-case Analyze Revenue
Brief Description	Admin can check information about the revenue made within a certain period.
Actors	Admin
Pre-conditions	Admin is logged in. The revenue data should be long enough to be statistically measurable.
Post-conditions	The revenue data and statistics within a certain period of time will be displayed.
Basic Flows	1. Admin accesses the feature by selecting the Revenue tab. 2. Admin chooses the options of displaying the revenue: 2.1. Admin can choose the time range in the last day, week, month and year. 2.2. Admin can use filters to choose which book should be investigated. 2.3. Admin can type in the search bar to find and choose a specific book or brand/publisher. 3. The system displays the revenue based on the given information.
Alternative Flows	3.1. The system displays a notification if the book/publisher does not exist.
Special Requirements (Nonfunctional Requirements)	- In good conditions (stable networks, the device operates normally), the time to return search results should not exceed 5 seconds.
Extension Points	None

2.17   Use-case Create sale off event

Use Case ID	UC017
Use Case Name	Use-case Create Sale-off Event
Brief Description	Admin can create a Sale-off Event for the website.
Actors	Admin
Pre-conditions	Admin is logged in.

Post-conditions	A Sale-off Event will be created.
Basic Flows	1. Admin accesses the feature by selecting the Event tab. 2. Admin presses the Create button to create an event. 3. Admin input the data (Name, description, books for sale, discount percentage, start and end times, etc.) 4. Admin presses the Save button to save the event or presses the Cancel button to cancel the process.
Alternative Flows	4.1. When the admin presses Save, if there is invalid data, a notification will be displayed and the event will not be saved/created.
Special Requirements (Nonfunctional Requirements)	None.
Extension Points	None.

2.18   Use-case Manage Order

Use Case ID	UC018
Use Case Name	Use-case Manage Order
Brief Description	Admin can handle each specific order, viewing and updating its information.
Actors	Admin
Pre-conditions	Admin is logged in. The order has to exist in the system.
Post-conditions	Order information will be displayed. Order information can be updated.

Basic Flows	1. Admin access the feature by selecting the Order tab.
	2. Admin can check specific orders by inputting orderID in the search bar.
	3. The order information will be displayed for further actions.
	4. Admin can update order information and status:
	   - Admin clicks on the Update button of the order.
	   - Admin chooses the specific information field(s) and changes the data.
	   - Admin presses the Save button to save the new information.

Alternative Flows	2.1. If the orderID does not exist then a pop-up caution will be displayed.
Special Requirements (Nonfunctional Requirements)	- In good conditions (stable networks, the device operates normally), the time to return search results should not exceed 5 seconds.
Extension Points	None.