Book Alley

Use-Case Specification

Version 4.0

Revision History

| **Date** | **Version** | **Description** | **Author** |
| --- | --- | --- | --- |
| 09/11/2023 | 1.0 | Initial | Dang Ha Huy |
| 17/11/2023 | 2.0 | Review | Phung Sieu Dat |
| 25/11/2023 | 3.0 | Merge and Update important flows | Tran Minh Hai Uyen |
| 29/11/2023 | 4.0 | Review | Dang Ha Huy |
| 30/11/2023 | 5.0 | Adjust to suit the requirements | Chau Tan Kiet |

Table of Contents

[**Use-Case Specification: Login 5**](#_heading=h.gjdgxs)

[**1. Use-Case Name 6**](#_heading=h.30j0zll)

[**2. Flow of Events 6**](#_heading=h.3znysh7)

[**3. Special Requirements 7**](#_heading=h.4d34og8)

[**4. Preconditions 7**](#_heading=h.17dp8vu)

[**5. Postconditions 7**](#_heading=h.26in1rg)

[**6. Extension Points 8**](#_heading=h.35nkun2)

[**Use-Case Specification: Registration 9**](#_heading=h.wxzybmhuox8c)

[**1. Use-Case Name 9**](#_heading=h.d2vak3kar1x9)

[**2. Flow of Events 9**](#_heading=h.itu0zthu2sxi)

[**3. Special Requirements 10**](#_heading=h.q1ewn01ikceb)

[**4. Preconditions 10**](#_heading=h.wmpaki557knf)

[**5. Postconditions 10**](#_heading=h.qbuvx3feepk8)

[**6. Extension Points 11**](#_heading=h.6t12t5dck99i)

[**Use-Case Specification: Forgot password 12**](#_heading=h.55yy5hktvdts)

[**1. Use-Case Name 12**](#_heading=h.3svcmg431ewd)

[**2. Flow of Events 12**](#_heading=h.bwas0iagjv3i)

[**3. Special Requirements 13**](#_heading=h.wgr9lb0cc9z)

[**4. Preconditions 13**](#_heading=h.xn7bt2hpb8ho)

[**5. Postconditions 13**](#_heading=h.mht0ld593w8t)

[**6. Extension Points 14**](#_heading=h.qro3evxo4ny2)

[**Use-Case Specification: Manage accounts 15**](#_heading=h.8mfmnavq2ksn)

[**1. Use-Case Name 15**](#_heading=h.uejbb42soo4)

[**2. Flow of Events 15**](#_heading=h.lpsmqd3j4rfg)

[**3. Special Requirements 15**](#_heading=h.p9tgjshrj4wn)

[**4. Preconditions 16**](#_heading=h.oqt6d8lhbxj9)

[**5. Postconditions 16**](#_heading=h.9agqsbn72pq)

[**6. Extension Points 16**](#_heading=h.cv6yof8br3g0)

[**Use-Case Specification: Manage books 17**](#_heading=h.r2au73aakon6)

[**1. Use-Case Name 17**](#_heading=h.3x6wt0ev2wid)

[**2. Flow of Events 17**](#_heading=h.2fomz58cbx63)

[**3. Special Requirements 18**](#_heading=h.r2m5z26b4zvs)

[**4. Preconditions 19**](#_heading=h.wye2x1iuvbaa)

[**5. Postconditions 19**](#_heading=h.3strz0e3v1rm)

[**6. Extension Points 19**](#_heading=h.6e3llbqwm5gm)

[**Use-Case Specification: Manage Users 21**](#_heading=h.g5mkhxalexaa)

[**1. Use-Case Name 21**](#_heading=h.cvy3c5efk96a)

[**2. Flow of Events 21**](#_heading=h.fiqtdivpmdyd)

[**3. Special Requirements 22**](#_heading=h.6zp4cpwe7tdg)

[**4. Preconditions 23**](#_heading=h.t33t8mekm1le)

[**5. Postconditions 23**](#_heading=h.8g52cbx1ei1o)

[**6. Extension Points 23**](#_heading=h.lodckmrpfm8c)

[**Use-Case Specification: Manage orders 24**](#_heading=h.c3s7346yaut6)

[**1. Use-Case Name 24**](#_heading=h.lllk3jqvw86)

[**2. Flow of Events 24**](#_heading=h.inafmcka3ac0)

[**3. Special Requirements 25**](#_heading=h.qw02i653bekp)

[**4. Preconditions 26**](#_heading=h.uxgtbavvqx4u)

[**5. Postconditions 26**](#_heading=h.a0fn5624lgv9)

[**6. Extension Points 26**](#_heading=h.2rtut9a8ewi4)

[**Use-Case Specification: Browse books 28**](#_heading=h.bsqm2qag62l0)

[**1. Use-Case Name 28**](#_heading=h.p59vvtyvdbl9)

[**2. Flow of Events 28**](#_heading=h.rgjkeiqm3zt5)

[**3. Special Requirements 29**](#_heading=h.16mrqa6d1sz9)

[**4. Preconditions 29**](#_heading=h.h3028w55h4gu)

[**5. Postconditions 30**](#_heading=h.uzwv9jpe6qim)

[**6. Extension Points 30**](#_heading=h.61rls3izypx2)

[**Use-Case Specification: Add product to cart 32**](#_heading=h.3urb9cesddym)

[**1. Use-Case Name 32**](#_heading=h.vhp30x3r3cjv)

[**2. Flow of Events 32**](#_heading=h.qeovg0eyzm9n)

[**3. Special Requirements 33**](#_heading=h.gnthky7qnm5z)

[**4. Preconditions 33**](#_heading=h.2g5cmi4upxan)

[**5. Postconditions 34**](#_heading=h.kk379uw6k40o)

[**6. Extension Points 34**](#_heading=h.tjic4jewb8ga)

[**Use-Case Specification: Checkout 36**](#_heading=h.6z5z105xglg7)

[**1. Use-Case Name 36**](#_heading=h.80sk6bgw2nfv)

[**2. Flow of Events 36**](#_heading=h.9eeiwmyaksz0)

[**3. Special Requirements 37**](#_heading=h.8277r5a7l86)

[**4. Preconditions 38**](#_heading=h.o1v1i4ncgvn8)

[**5. Postconditions 38**](#_heading=h.ixm0g2cd1tn8)

[**6. Extension Points 39**](#_heading=h.9wx0lahr2h2i)

[**Use-Case Specification: Review and Rating 40**](#_heading=h.8h24arghyij5)

[**1. Use-Case Name 40**](#_heading=h.7tet0xajsn0)

[**2. Flow of Events 40**](#_heading=h.3pwcm6mxsevw)

[**3. Special Requirements 40**](#_heading=h.vs2h5naq32b8)

[**4. Preconditions 41**](#_heading=h.xmzwrhijiv0v)

[**5. Postconditions 41**](#_heading=h.pqyt91s1alws)

[**6. Extension Points 42**](#_heading=h.3fpuiivu23k8)

[**Use-Case Specification: Moderate reviews 43**](#_heading=h.2srxyl652ott)

[**1. Use-Case Name 43**](#_heading=h.2h3msr5ivncs)

[**2. Flow of Events 43**](#_heading=h.51acbp4hsdby)

[**3. Special Requirements 43**](#_heading=h.vbmf73wk3ep5)

[**4. Preconditions 44**](#_heading=h.dqsht4kadrjj)

[**5. Postconditions 44**](#_heading=h.ttdyvrxu7wo8)

[**6. Extension Points 45**](#_heading=h.4gm8sax65z25)

# Use-Case Specification: Login

## Use-Case Name

* 1. **Brief Description**

The login use case of Book Alley allows users with an account to securely sign in to their account and enables them to use other functions of the web application.

## Flow of Events

* 1. **Basic Flow**

1. User action: Accesses the website through an internet browser

System response: Displays the homepage with a login option in the top right corner

1. User action: Selects the account icon to initiate the login process

System response: Presents the login screen with fields for username and password

1. User action: Enters valid credentials

System response: Validates the credentials against the stored user account information in the database

System action (if credentials are valid): Grants the user access to their account and redirects them to the homepage

1. User action: Uses the website's features and accesses their personalized content
   1. **Alternative Flows**
      1. *Invalid credentials*
2. User action: Navigates to the login page
3. User action: Enters invalid credentials

System response: Detects that the credentials are invalid and displays an error message to the user

System response: Prompts the user to re-enter their credentials

1. User action: Attempts to login again
2. System response: Continues to validate credentials. The process repeats until the credentials are valid or the user abandons the login attempt.
   * 1. *Account lockout*
3. User action: Navigates to the login page
4. User action: Enters invalid credentials multiple times

System response: Detects multiple failed login attempts

System action: Locks the user's account after a predefined number of failed attempts

System response: Displays an account lockout message

System response: Informs the user that their account has been temporarily locked

System response: Provides instructions for initiating account recovery, such as resetting the password or contacting customer support

* + 1. *Two-factor authentication*

1. User action: Navigates to the login page
2. User action: Enters credentials

System response: Checks the entered credentials against the stored user account information in the database

System response: Prompts the user for a second authentication factor (2FA) if credentials are valid

1. User action: Enters the 2FA code

System response: Validates the 2FA code

System action (if 2FA code is valid): Grants the user access to their account

System action (if 2FA code is invalid): Prompts the user to retry for a maximum of 5 times, if the number of retries exceeded 5, the account will be locked for 30 minutes to retry again

The system validates the code. If it's correct, the user gains access to their account; otherwise, they are prompted to retry for a maximum of 5 times, if the number of retries exceeded 5, the account will be locked for 30 minutes to retry again.

## Special Requirements

* 1. **Security**
* Strong encryption protocols should be used to protect user information like username and password
  1. **User experience**
* The login page should be intuitive and easy to use, providing essential functions and offer a saving method like "Remember Me" option for convenience.
  1. **Performance**
* The login process should be fast and responsive, handling a high volume of concurrent requests.
* Cached data should be used to reduce database queries and improve performance.
  1. **Compatibility**
* Login functionality should be supported on popular browsers.
* The login page should be responsive and mobile-friendly.
* Multiple languages and character sets should be supported.

## Preconditions

* 1. **Internet connectivity**
* The user’s PC must have an internet connection to access the website and communicate with the website’s servers for authentication and other functions.
  1. **User account exists**
* There must be an existing user account for them to be in the system. The system must have stored the user's credentials and profile information.
  1. **Website is accessible**
* The website must be up and running, and the user must have access to it.
  1. **Compatibility**
* The customer's device should meet the compatibility requirements of Book Alley web application.
  1. **Account not locked**
* The user's account should not be locked or suspended due to multiple unsuccessful login attempts or any other reason.

## Postconditions

* 1. **Login successfully**
* If the user provided valid credentials then the website authenticates them and leads the user to the homepage.
  1. **Session Management**
* A unique identifier is generated for the user's session, used to track the user's activity and keep them logged in.
  1. **Account lockout**
* If the user enters their password incorrectly too many times, their account is locked.
* This prevents unauthorized access to the account.
  1. **Error Handling**
* If the login process encounters any errors or exceptions, appropriate error messages are displayed to the user, indicating the nature of the issue.
  1. **Logging in and security**
* Appropriate logging and security measures should be in place to track and protect the user's login information and the system's data. This might include measures like password encryption and secure communication.

## Extension Points

* 1. **Forgotten password**
* Allow users that have forgotten their password to reset it. The use case can be extended to include a "Password Reset" feature, allowing the user to regain access to their account.
  1. **Captcha**
* Include the use of CAPTCHA verification or other security challenges to prevent automated or brute-force login attempts
  1. **Two-step verification (2FA)**
* In scenarios where a user has enabled two-step verification for their account, this extension point can be triggered to accommodate the additional step in the login process.
  1. **Social media login**
* Allowing users to log in using their social media accounts such as Google, Facebook, Twitter, etc.

# Use-Case Specification: Registration

## Use-Case Name

* 1. **Brief Description**

The registration use case for Book Alley enables user to create a new account to gain access to the all of the website features

## Flow of Events

* 1. **Basic Flow**

1. User action: Navigates to the register page by clicking on "Register now" on the login page

System response: Presents the registration form with fields for username, email, and password

1. User action: Enters their information into the registration form and clicks "Submit"

System response: Validates the user's input by checking:

* Email address format: Ensures the email address is in a valid format
* Username uniqueness: Checks if the chosen username is already in use
* Password strength: Verifies that the password meets security requirements

System action (if input is valid):

* Creates a new user account in the database
* Stores the user's information securely

System response (if input is valid):

* Displays a confirmation message indicating successful registration
* Prompts the user to log in using their newly created account
  1. **Alternative Flows**
     1. *Invalid data*

1. System action (if invalid data is detected):

* Highlights the specific fields containing errors
* Provides clear error messages explaining the issues
* Prompts the user to correct the errors

1. User action: Corrects the errors and submits the registration form again

System response: Repeats the validation process until all data is valid

* + 1. *Password strength*

1. System action (if password strength is insufficient):

* Provides a message indicating the password does not meet security requirements
* Explains the specific password criteria that need to be met

1. User action: Chooses a stronger password and resubmits the registration form

System response: Repeats the validation process until the password meets the required strength

* + 1. *If user already have an account*

1. System action (if the email address is already registered):

* Detects that the email address is associated with an existing account
* Displays a message informing the user that the email is already in use

1. User action:

* Reviews the message and decides whether to proceed with registration using a different email address
* Navigates back to the login page to access their existing account

## Special Requirements

* 1. **Security**
* Employ strong encryption measures to protect user data and prevent unauthorized access.
  1. **User experience**
* Design a user-friendly registration process that caters to everyone, focusing on essential information and providing clear feedback to users throughout the process.
  1. **Performance**
* The registration process should be fast, responsive, and able to handle many users at once.
  1. **Compatibility**
* Maintain compatibility with major web browsers and ensure accessibility across devices.
* Implement a responsive design for seamless adaptation to different screen sizes.
  1. **Server Functionality**
* Utilize a robust, scalable, and highly available server infrastructure.
* Keep server software up-to-date for optimal performance, security, and compatibility.
* Proactively monitor and manage the server environment to maintain user experience.

## Preconditions

* 1. **No duplicate account**
* Prevent duplicate accounts using the same email or username.
  1. **Unique Identity**
* Prohibit multiple accounts with the same identifying information.
  1. **Required Fields**
* Clearly indicate mandatory fields on the registration page.
  1. **Data Uniqueness**
* Check for existing accounts using the provided email or username.
  1. **Error Messages**
* Display clear and informative error messages for invalid or missing information.

## Postconditions

* 1. **Account creation**
* Upon registration, create a new user account in the system's database.
  1. **Account activation**
* Activate the user's account immediately, allowing immediate access.
  1. **Account verification**
* Implement account verification processes, such as email or phone verification, if applicable.
* Enhance security and protect against spam, malicious activities, and unauthorized access.
  1. **User credentials**
* Provide the user with login instructions and safeguard their credentials securely.
* Store passwords in a hashed format to prevent plaintext disclosure.
* Implement measures to protect against unauthorized access or retrieval of user credentials.
  1. **Error handling**
* Display clear error messages in case of registration failures.
* Provide guidance for resolving issues.
* Formulate error messages in a clear, concise, and informative manner.

## Extension Points

* 1. **Social media integration**
* Enable social media logins for easy account creation and access.
  1. **Two-factor authentication**
* Implement 2FA for enhanced account security.
  1. **Multi-language support**
* Provide multi-language support to cater to a global audience.
  1. **Captcha**
* Utilize captchas to prevent automated bot registrations.
  1. **Third-party integration**
* Integrate with third-party services for seamless user experience.
  1. **User notification**
* Engage users with relevant notifications and prompts during registration.

# Use-Case Specification: Forgot password

## Use-Case Name

* 1. **Brief Description**

The forgot password use case for Book Alley allows users to recover their account when they forgot their password by using their recovery email.

## Flow of Events

* 1. **Basic Flow**

1. User action: Clicks on the "Forgot Password" button on the login page.

System response: Displays a prompt for the user to enter their recovery email address.

1. User action: Enters their recovery email address and submits the request.

System response: Validates the entered email address against the stored user account information in the database.

System action (if email is valid):

* Generates a secure password reset token
* Creates a password reset link containing the token
* Sends the password reset link to the user's registered email address

System response (if email is invalid):

* Displays an error message indicating that the email address is not associated with a registered account
* Prompts the user to re-enter their email address or contact customer support

1. User action: Checks their email and clicks on the provided password reset link.

System response: Verifies the validity of the password reset token and redirects the user to a secure password reset page.

1. User action: Enters a new password and submits the form.

System response: Validates the new password strength and ensures it meets security requirements.

System action (if password is valid):

* Updates the user's password in the database
* Sets the password reset token as invalid to prevent further use

System response (if password is invalid):

* Displays an error message indicating that the password does not meet security requirements
* Prompts the user to choose a stronger password

System response (upon successful password reset):

* Sends a confirmation message to the user's email address informing them that their password has been successfully reset
* Prompts the user to log in using their new password
  1. **Alternative Flows**
     1. *Invalid email address*

User action: Enters an invalid email address and submits the request.

System response:

* Detects that the entered email address is not associated with a registered account.
* Displays an error message indicating that the email address is not found or invalid.
* Prompts the user to re-enter their email address or contact customer support.
  + 1. *Expired or invalid reset link*

User action: Clicks on an expired or invalid password reset link.

System response:

* Detects that the password reset token is invalid or has expired.
* Displays a message informing the user that the password reset link is no longer valid.
* Provides the user with the option to request a new password reset link.
  + 1. *Account lockout*

1. User action: Makes multiple unsuccessful attempts to reset their password.

System response: Detects a pattern of failed password reset attempts.

System action (if predefined number of attempts is exceeded):

* Temporarily locks the user's account to prevent further unauthorized attempts
* Implements security measures, such as CAPTCHA, to prevent automated attacks

1. System response: Notifies the user that their account has been temporarily locked.

System response: Provides instructions on how to initiate account recovery, such as contacting customer support.

## Special Requirements

* 1. **Security**
* Utilize encryption throughout the password reset process.
* Generate unique and time-bound reset tokens.
* Implement safeguards against common security attacks.
  1. **Accessibility**
* The forgot password function can always be accessed in the login page so that the user can easily find it when they forget their password
  1. **Performance**
* Optimize response time for a smooth user experience and be able to handle a high volume of reset requests efficiently.
  1. **Regular backup**
* Implement a regular backup system for the user database to protect against data loss and ensure data recovery in case of system failures.

## Preconditions

* 1. **Internet connectivity**
* The user's device should have a working internet connection to access the website
  1. **User is registered**
* The user attempting to use the "Forgot Password" functionality must be registered and already have an account
  1. **Valid email address**
* The user provides a valid email address associated with their account during the password recovery process.
  1. **System availability**
* The forgot password page should be up and running

## Postconditions

* 1. **Password reset successful**
* After successfully completing the password reset process, the user's account password should be updated in the system.
* The new password should be immediately activated and allow the user to log in to their account.
  1. **Notification sent**
* A confirmation email should be sent to the user, notifying them that their password has been changed.
* The confirmation message can also be displayed on the password reset page..
  1. **Expired link**
* If a reset password link expires or becomes invalid, the system should inform the user and provide an option to request a new reset link.
  1. **User feedback**
* Provide clear and user-friendly feedback messages throughout the process, indicating the status of each step (e.g., whether the email was sent successfully, whether the reset link is valid, etc.).
  1. **Security audit**
* Conduct periodic security audits to ensure that the "Forgot Password" functionality remains secure and is not susceptible to common vulnerabilities, such as account enumeration or token misuse.

## Extension Points

* 1. **Custom email templates**
* Allow customization of password reset email content and appearance for branding consistency.
  1. **Integration with external identity providers**
* Integrate with external identity providers for password reset through social login or third-party authentication services.
  1. **Integration with Password Management Tools**
* Enable integration with popular password management tools for seamless password updates.
  1. **Two-Factor Authentication (2FA)**
* Provide an option to enable two-factor authentication for the forgotten password process.
  1. **Multi-Channel Password Reset**
* Extend the password reset functionality to support multiple channels, such as SMS or voice-based password reset options in addition to email.

# Use-Case Specification: Manage accounts

## Use-Case Name

* 1. **Brief Description**

The managed account use case for Book Alley allows users to customize their account to their liking.

## Flow of Events

* 1. **Basic Flow**

1. User action: Logs into their account using their valid credentials.

System response: Grants the user access to their account and redirects them to the homepage.

1. User action: Navigates to the account icon on the top right corner and selects "Profile" from the dropdown menu.

System response: Displays the user's profile page with options for managing their account information.

1. User action: Selects the specific information they want to update (e.g., profile picture, password, email address, phone number).

System response: Presents the corresponding update form based on the selected information.

1. User action: Enters the new information and clicks on "Confirm Change" to submit the updates.

System response: Validates the updated information to ensure it meets the required criteria (e.g., valid email format, strong password).

System action (if validation is successful):

* Updates the user's account information in the database
* Displays a confirmation message indicating that the changes have been successfully saved

System action (if validation fails):

* Displays an error message explaining the specific validation errors
* Prompts the user to correct the invalid information
  1. **Alternative Flows**
     1. *Wrong current password while changing password*

1. User action: Clicks on the "Change Password" option.

System response: Prompts the user to enter their current password and the new password they want to set.

System action: Validates the entered current password against the stored user account information.

System action (if current password is incorrect):

* Displays an error message indicating that the current password is incorrect
* Prompts the user to re-enter their current password

1. User action: Re-enters the current password until it is correct.
   * 1. *Invalid details*
2. User action: Clicks on the option to update their phone number or email address.

System response: Prompts the user to enter their new phone number or email address.

System action: Validates the entered new details against the existing user accounts in the database.

System action (if new details are already in use):

* Detects that the new details are associated with an existing account
* Displays an error message indicating that the new details are already in use

1. User action: Re-enters the details until they are unique and valid.

## Special Requirements

* 1. **Accessibility**
* Adhere to WCAG standards for keyboard navigation, alt text, and accessibility.
  1. **Data privacy**
* Detailed privacy policy and obtaining explicit user consent for data processing.
  1. **Security**
* Enhance account security during password change by implementing industry-standard measures CAPTCHA.
  1. **Confirm message**
* After each successful operation (e.g., profile update, password change), display clear confirmation messages confirming the action. In case of failures, provide specific error messages with guidance on how to resolve the issue..

## Preconditions

* 1. **Internet connectivity**
* For seamless access to the website, the user's device must maintain a stable and active internet connection.
  1. **User is authenticated**
* To utilize the "Manage account" function, the user is required to be successfully logged into their account.
  1. **Input verification**
* When updating information, the user is obligated to furnish accurate details, such as the correct current password when altering the password or a valid phone number.
  1. **System availability**
* The website must be operational and available for use; any interruptions in service should be minimal to ensure a consistent user experience.

## Postconditions

* 1. **Update user information**
* The customer's account details are successfully modified, ensuring that the latest information is reflected.
  1. **Updated communication preferences**
* Upon modification of communication preferences by the customer, the system accurately applies the changes to ensure adherence to the updated settings.
  1. **Changed password**
* If the customer opts to change their password, the system securely establishes the new password, rendering the old password invalid for enhanced security.

## Extension Points

* 1. **Profile customization**
* Provide extensive profile customization options, such as profile pictures, and layout personalization.
  1. **Advanced account recovery**
* Offer multiple verified channels or security questions for robust and flexible account recovery.

# Use-Case Specification: Manage books

## Use-Case Name

* 1. **Brief Description**

The manage books use case in Book Alley enables sellers to manage their books inventory such as add new book to sell, update information of their current book

## Flow of Events

* 1. **Basic Flow**

1. Seller Action: Logs into the e-commerce website using their valid credentials.

System Response: Validates the seller's credentials against the stored seller account information in the database.

System Action (if credentials are valid):

* Grants the seller access to their account.
* Redirects the seller to their dashboard.

1. Seller Action: Views their current inventory of books in the product page.

System Response:

* Retrieves the seller's inventory data from the database.
* Presents the seller with a list of their products, including details such as title, author, price, and quantity.

1. Seller Action: Selects the option to add a new book or update current books in the inventory.

System Response:

* Displays two options:
* "Add a new book".
* "Update current books".

1. Seller Action (if choosing to add a new book): Enter book details (title, author, translator, price, etc.) in the provided form.

System Response:

* Validates the entered book information.
* Ensures all required fields are filled and meet the defined criteria (e.g., valid price format).

System Action (if validation is successful):

* Creates a new book entry in the database.
* Adds the newly added book to the seller's inventory.
* Displays a confirmation message indicating successful book addition.

System Action (if validation fails):

* Displays an error message explaining the specific validation errors.
* Prompts the seller to correct the invalid information.

1. Seller Action (if choosing to update current books): Selects a book from their inventory.

System Response:

* Presents the selected book's details and allows the seller to edit relevant information.

Seller Action: Edits details such as price, quantity, or other relevant information.

System Response:

* Validates the updated book information.
* Ensures all required fields are filled and meet the defined criteria (e.g., valid price format, valid quantity).

System Action (if validation is successful):

* Updates the corresponding book entry in the database.
* Reflects the updated book details in the seller's inventory.
* Displays a confirmation message indicating successful book update.

System Action (if validation fails):

* Displays an error message explaining the specific validation errors.
* Prompts the seller to correct the invalid information.
  1. **Alternative Flows**
     1. *Invalid login*

1. Seller Action: Enters invalid credentials.

System Response:

* Detects that the credentials are invalid.
* Displays an error message indicating invalid credentials.

1. System Action: Prompts the seller to re-enter their login details.
   * 1. *Book information error*
2. Seller Action (during book addition): Enters erroneous book information.

System Response (during book addition):

* Detects errors in the entered book information.
* Displays an error message indicating the specific errors in the book information.

System Action (during book addition): Prompts the seller to correct the errors before submitting the information.

1. Seller Action (during book update): Enters erroneous updated book information.

System Response (during book update):

* Detects errors in the updated book information.
* Displays an error message indicating the specific errors in the updated book information.

System Action (during book update): Prompts the seller to correct the errors before submitting the updated information.

* + 1. *Out of stock*

1. Seller Action: Tries to update the quantity of a book to a value that exceeds the available stock.

System Response:

* Detects that the updated quantity exceeds the available stock.
* Displays an error message indicating that the desired quantity exceeds the available stock.
* System Action: Prompts the seller to enter a valid quantity that does not exceed the available stock.
  + 1. *Confirmation book removal*

1. Seller Action: Selects the option to remove a book from their inventory.

System Response:

* Displays a confirmation prompt to prevent accidental deletions.
* Ask the seller to confirm the book removal.

1. Seller Action: Confirms the book removal.

System Response:

* Removes the selected book from the seller's inventory and the database.
* Displays a confirmation message indicating successful book removal.

1. Seller Action (if canceling the removal): Declines the confirmation prompt.

System Response:

* Maintains the selected book in the seller's inventory and the database.
* Does not perform any removal action.

## Special Requirements

* 1. **Security measures**
  2. **Backup and recovery protocols**
* Establish regular data backup procedures for the book inventory to mitigate the risk of data loss due to system failures. Implement a reliable recovery mechanism to restore data in the event of accidental deletion or corruption, ensuring data integrity and continuity.
  1. **Scalability**
* Design the system with scalability in mind to accommodate the expanding number of books and sellers. This involves efficient database design and optimization for optimal performance as the system grows.
  1. **User-friendly interface**
* Develop an intuitive and user-friendly interface for managing books, prioritizing ease of navigation and action execution. This ensures that sellers can interact with the system seamlessly, reducing the likelihood of errors or confusion.
  1. **Audit trail**
* Incorporate an audit trail feature within the system to log all significant actions performed by sellers, including additions, updates, and removals of books.

## Preconditions

* 1. **Internet connectivity**
* For seamless website access, the user's device must maintain a reliable and active internet connection.
  1. **Seller is logged in**
* Access to the book management functionality is contingent upon the seller being authenticated and logged into the system.
  1. **System availability**
* The website must consistently be operational and available, ensuring uninterrupted access for users.
  1. **Valid book information**
* When adding or updating a book, the entered information must be accurate and comply with predefined formats and constraints.
  1. **Sufficient permissions**
* The logged-in seller must possess the requisite permissions to perform actions such as adding, updating, or removing books from their inventory, ensuring a secure and controlled environment.

## Postconditions

* 1. **Book addition confirmation**
* Upon successful addition, the added book must promptly appear in the seller's inventory with the specified details, ensuring real-time updates.
  1. **Book update validation**
* Following a successful update, the system should immediately reflect the changes in the book's details within the seller's inventory, maintaining accuracy and consistency.
  1. **Book removal validation**
* After successfully removing a book, the book should promptly disappear from the seller's inventory, ensuring that it is no longer visible or accessible.
  1. **Notification system**
* The system should generate appropriate notifications or messages for the seller, indicating the success or failure of each action (addition, update, removal) to keep them informed.
  1. **Data consistency assurance**
* Post each operation, the system must ensure data consistency, preventing scenarios where the data is left in an inconsistent or incomplete state, thus maintaining the integrity of the book inventory.
  1. **Secure log out**
* Upon completing book management tasks, the seller can securely log out of the system, ensuring the termination of the session in a manner that prioritizes security and user privacy.

## Extension Points

* 1. **Bulk operations**
* Enhance system capabilities to enable sellers to conduct bulk operations, such as simultaneous addition, updating, or removal of multiple books. This could involve functionalities like uploading a CSV file or utilizing a similar import/export mechanism for efficient management.
  1. **Advanced search and filters**
* Augment the search functionality to incorporate advanced filters, empowering sellers to search and filter books based on diverse criteria such as category, publication date, and ratings, enhancing the precision of book searches.
  1. **Seller analytics**
* Introduce analytics and reporting features within the system to provide sellers with valuable insights into their book performance. This includes tracking sales trends, customer preferences, and revenue generated, facilitating informed decision-making.
  1. **Integration with external systems**
* Enable seamless integration with external systems, such as third-party shipping services or accounting software, to streamline processes and offer additional functionalities, enhancing the overall efficiency of book management.
  1. **Internationalization and localization**
* Extend system support to include internationalization and localization features, allowing sellers to manage books in multiple languages and currencies. This ensures adaptability to diverse markets and provides a more inclusive platform.
  1. **Integration with social media**
* Incorporate features that enable sellers to share information about their books directly from the seller dashboard on social media platforms. This integration enhances marketing efforts and promotes broader visibility for the seller's book offerings.

# Use-Case Specification: Manage Users

## Use-Case Name

* 1. **Brief Description**

The "Manage Users" use case for an administrator involves the efficient and effective handling of user-related activities within a system or platform. Administrators are responsible for overseeing and controlling user accounts to ensure secure and organized access.

## Flow of Events

* 1. **Basic Flow**

***Create a new user***

1. Administrator action: Selects the option to create a new user.

System response: Presents a form for the administrator to enter the new user's information, such as username, email address, and password.

1. Administrator action: Enters the required information and submits the form.

System response: Validates the entered information to ensure it meets all security requirements.

System action (if validation is successful):

* Creates a new user account in the database.
* Stores the user's information securely.
* Displays a confirmation message indicating successful user creation.

System action (if validation fails):

* Displays an error message explaining the specific validation errors.
* Prompts the administrator to correct the invalid information.

***Assign a role and set appropriate permissions***

1. Administrator action: Selects a user from the list of existing users.

System response: Displays the selected user's profile page, including their current role and permissions.

1. Administrator action: Chooses the appropriate role and sets the relevant permissions for the selected user.

System response:

* Updates the user's role and permissions in the database.
* Displays a confirmation message indicating successful role and permission assignment.

***Reset a user's password***

1. Administrator action: Selects a user from the list of existing users.

System response: Displays a confirmation prompt to verify the administrator's intention to reset the selected user's password.

1. Administrator action (if confirming reset): Confirms the password reset.

System response:

* Generates a secure password reset token.
* Sends a password reset email to the user's registered email address, containing the password reset token.
* Displays a confirmation message indicating successful password reset initiation.

***Suspend or reactivate a user's account***

1. Administrator action: Selects a user from the list of existing users.

System response:

* Displays the selected user's profile page, including their current account status (active or suspended).

1. Administrator action (if suspending the account): Chooses the option to suspend the selected user's account.

System response:

* Suspends the user's account, preventing them from logging in.
* Updates the user's account status in the database.
* Displays a confirmation message indicating successful account suspension.

1. Administrator action (if reactivating the account): Chooses the option to reactivate the selected user's account.

System response:

* Reactivates the user's account, allowing them to log in.
* Updates the user's account status in the database.
* Displays a confirmation message indicating successful account reactivation.
  1. **Alternative Flows**
     1. *User Creation Errors*

1. Administrator action: Attempts to create a new user but encounters errors.

System response: Detects errors in the user creation process, such as:

* Duplicate username: Another user already exists with the same username.
* Invalid email address: The entered email address is not in a valid format.
* Missing required information: Essential fields, such as password or email address, were not filled in.
* Displays an error message indicating the specific error(s) that prevent user creation.

System action:

* Prompts the administrator to correct the erroneous or missing information.

System action (upon successful correction):

* Continues the user creation process and creates the new user account if all validation criteria are met.
  + 1. *Password Reset Challenges*

1. Administrator action: Notices that a user is facing difficulties with the password reset process.

System response: Detects potential issues with the password reset process, such as:

* Invalid password reset link: The user may have clicked on an expired or invalid password reset link.
* Multiple failed reset attempts: The user may have made numerous unsuccessful attempts to reset their password.

System response (if invalid reset link): Provides the administrator with options to:

* Generate a new password reset token and send it to the user.
* Manually reset the user's password.

System response (if multiple failed attempts): Triggers security measures, such as temporary account lockout or CAPTCHA, to prevent automated attacks.

System action: Prompts the administrator to choose the appropriate method to assist the user in regaining access securely.

System action (upon successful password reset): Notifies the administrator that the user's password has been successfully reset.

## Special Requirements

* 1. **Password Policy Enforcement**
* Enforce strong password complexity requirements to resist brute force attacks.
* Implement mechanisms to prevent password reuse across multiple accounts.
  1. **User Data Privacy**
* Encrypt sensitive user data at rest and in transit using industry-standard algorithms.
* Clearly inform users about data collection, usage, and obtain explicit consent.
  1. **Emergency Access Procedures**
* Establish a structured authorization and approval process for emergency access.

## Preconditions

* 1. **Authentication**
* Secure administrator login to safeguard user management functions.
* Utilize strong authentication protocols.
  1. **Administrator Privileges**
* Restrict user management access to authorized administrators with necessary permissions.

## Postconditions

* 1. **User Account Changes Persist**
* Ensure user account changes persist until explicitly reversed.
* Implement robust data storage mechanisms for reliable user data storage and retrieval.
  1. **User Role Assignment**
* Enable assigning or modifying roles for specific users.
* Implement a role management interface for administrators to select appropriate roles for each user.
* Enforce mechanisms to prevent unauthorized role changes.

## Extension Points

* 1. **User Authentication Extension Point**
* Integrate external authentication providers like OAuth and LDAP for diverse user preferences.
* Implement a mechanism for registering and configuring external authentication providers.
  1. **Custom User Fields**
* Add custom fields to user profiles without modifying core user management functionality.
* Provide a mechanism for defining and managing custom fields.

# Use-Case Specification: Manage orders

## Use-Case Name

* 1. **Brief Description**

The managed orders use case in Book Alley enables sellers to manage their orders more efficiently.

## Flow of Events

* 1. **Basic Flow**

1. Seller Action: Logs into their account using their credentials.

System Response: Validates the seller's credentials and grants access to their account.

1. Seller Action: Navigates to the orders dashboard.

System Response: Presents the seller with a list of orders, categorized by their current status (e.g., pending, processing, shipped, delivered).

1. Seller Action: Reviews the list of pending orders that need attention.

System Response: Highlights the pending orders that require action by the seller.

1. Seller Action: Clicks on a specific order to view its details.

System Response: Displays the order details page, including:

* Customer information (name, address, contact details)
* Book details (title, author, quantity, price)
* Order status (pending, processing, shipped, delivered)
* Payment information (payment method, transaction ID)

1. Seller Action: Updates the order status based on the current state.

System Response: Verifies that the updated order status is valid and consistent with the order's progress.

System Action (if status update is valid):

* Updates the order status in the database
* Reflects the updated status on the order details page
* Triggers notifications based on the new status (e.g., shipment confirmation for "shipped" status)

System Action (if status update is invalid):

* Displays an error message explaining the invalid status change
* Prompts the seller to select a valid order status

1. Seller Action (if generating an invoice): Requests to generate an invoice for the order.

System Response:

* Retrieves the order's details and billing information
* Generates an invoice document in the chosen format (e.g., PDF)
* Provides the seller with options to download or print the invoice

1. Seller Action (if marking the order as "shipped"): Marks the order as "shipped" and proceeds to pack the books.

System Response:

* Updates the order status to "shipped" in the database
* Triggers inventory updates to reflect the shipped items
* Generates a shipment confirmation email to the customer, including tracking information if available

1. Seller Action: Arranges for shipping of the packed books.

System Response (if shipping information is provided):

* Updates the order's shipping details in the database
* Provides a shipping label or tracking information to the seller

System Action (upon order delivery confirmation):

* Updates the order status to "delivered" in the database
* Archives the completed order for record-keeping purposes
* Sends a final notification to the customer confirming order delivery
  1. **Alternative Flows**
     1. *Inventory issue*

1. Seller Action: Notices a discrepancy between the ordered quantity and the available inventory.

System Response:

* Identifies the discrepancy and alerts the seller during order processing.
* Displays a notification indicating the specific book(s) with insufficient stock.

1. Seller Action: Updates the order status to "On Hold" to prevent further processing.

System Response:

* Changes the order status to "On Hold" in the database.
* Reflects the updated status on the order details page.

1. Seller Action: Contacts the customer to explain the situation and apologize for the inconvenience.

System Response:

* Provides the seller with contact information for the customer.
* Suggests a template message to inform the customer about the inventory issue and the expected restocking timeframe.

1. Seller Action: Takes steps to restock the book(s) in question.

System Response:

* Guides the seller through the restocking process, which may involve:
* Contacting the supplier to place a new order for the out-of-stock book(s).
* Monitoring the restocking status and updating the customer with estimated delivery dates.

1. Seller Action (once inventory is restocked): Updates the order status and notifies the customer of the new status.

System Response:

* Marks the order status as "Processing" to indicate that the inventory issue has been resolved.
* Sends a notification to the customer informing them that their order is now ready to be shipped.

1. Seller Action: Proceeds with packaging, shipping, and updating the inventory as per the normal flow.

System Response:

* Handles the remaining steps of the order processing workflow, including:
* Packing the ordered books and arranging for shipment.
* Updating the order status to "Shipped" and sending shipment confirmation to the customer.
* Updating the inventory levels to reflect the shipped items.

## Special Requirements

* 1. **Secured authentication**
* Implement robust authentication measures within the system to ensure secure access, preventing unauthorized entry into the seller's account and safeguarding sensitive information.
  1. **Real-time inventory update**
* Enable real-time updates to the inventory system, ensuring prompt and accurate reflections of stock levels.
  1. **Order notification**
* Establish a notification system to promptly inform the seller of new orders and any changes in order status.
  1. **Integration with shipping providers**
* Facilitate seamless integration with shipping providers, allowing for the creation of shipping labels, tracking information, and efficient order fulfillment.
  1. **Invoice generation**
* Provide support for the generation of invoices as needed by the seller for order transactions.

## Preconditions

* 1. **Internet connectivity**
* Ensure the user's device has a stable internet connection for accessing the website and utilizing its features without interruptions.
  1. **Seller is logged in**
* Authenticate and log in the seller before granting access to the book management functionality, enhancing the security of the system.
  1. **System availability**
* Maintain the website in an operational state, ensuring it is consistently up and running for seamless user access.
  1. **Valid seller account**
* Verify that the seller possesses a valid and active account on the e-commerce platform, ensuring legitimacy and authorized access to the system.
  1. **Updated product catalog**
* Grant the logged-in seller the necessary permissions to add, update, or remove books from their inventory, ensuring an accurate and up-to-date product catalog.
  1. **Sufficient inventory**
* Ensure that there is an ample quantity of inventory available for the books being sold, preventing issues related to stock shortages.
  1. **Authenticated session**
* Require the seller to have an authenticated session to access the orders management functionality, reinforcing security measures within the system.

## Postconditions

* 1. **Updated order status**
* Following order processing, promptly update the order status to accurately reflect the current state (e.g., processed, shipped, delivered), providing real-time information to both the seller and the customer.
  1. **Updated inventory**
* Ensure the inventory is promptly updated to mirror changes resulting from order processing, such as a decrease in stock after shipping.
  1. **Notification sent**
* Implement a notification system to inform the customer of any changes in the order status, including shipping confirmation or delivery updates, enhancing transparency and customer satisfaction.
  1. **Archived orders**
* Archive completed orders systematically for historical and record-keeping purposes, facilitating efficient order management and retrieval of past transaction details.
  1. **Invoice record**
* Maintain a record of generated invoices associated with the respective orders, providing a comprehensive overview of financial transactions for reference and auditing purposes.
  1. **Shipping documentation**
* Generate shipping labels and related documentation for orders marked as "shipped," streamlining the shipping process and contributing to effective order fulfillment.

## Extension Points

* 1. **Discounts and promotions**
* Enable sellers to create and apply custom discounts or promotions for their products by implementing a flexible discount engine.
  1. **Advanced analytics**
* Facilitate advanced analytics features for sellers to gain profound insights into sales performance, customer behavior, and inventory management.
  1. **Dynamic pricing strategies**
* Implement a framework that empowers sellers to apply dynamic pricing strategies based on factors such as demand, inventory levels, or market conditions.
  1. **Return and refund policies**
* Provide extension points for sellers to define and implement custom return and refund policies, allowing for flexibility to accommodate different business models and customer service approaches.

# Use-Case Specification: Browse books

## Use-Case Name

* 1. **Brief Description**

The browse book use case for Book Alley enables users to explore a wide range of book available on the website, view details information and purchase the book that they like

## Flow of Events

* 1. **Basic Flow**

1. User Action: Opens the website and lands on the homepage.

System Response:

* Retrieves the homepage content from the database, including the layout, banners, and featured products.
* Presents the homepage to the user, showcasing a curated selection of books and highlighting promotions or new releases.

1. User Action: Scrolls through the list of available books to explore more.

System Response:

* Detects the user's scrolling behavior and loads additional book listings dynamically to provide a seamless browsing experience.
* Ensures that the book listings are displayed in an engaging and visually appealing manner, using high-quality images and concise descriptions.

1. User Action: Applies filters (e.g., by genre, author, price) to narrow down the book selection.

System Response:

* Presents the user with a variety of filter options, allowing them to refine their search based on their preferences.
* Upon filter selection, updates the displayed book listings to match the specified criteria.
* Provides real-time updates to the book selection as the user applies or removes filters.

1. User Action: Selects a specific book to view its detail.

System Response:

* Retrieves the detailed information for the selected book from the database, including:
  + Full product description
  + Author biography
  + Customer reviews
  + Availability and pricing information
* Presents the book detail page to the user, providing a comprehensive overview of the selected book.

1. User Action: Decides to continue shopping or proceed to checkout.

System Response:

* Provides the user with clear navigation options, allowing them to continue exploring the website or initiate the checkout process.
* If the user chooses to continue shopping, maintains their current browsing position and filters.
* If the user proceeds to checkout, redirects them to the shopping cart page for order review and payment.
  1. **Alternative Flows**
     1. *Unauthenticated user access*

1. User Action: Opens the website and lands on the homepage.

System Response:

* Retrieves the homepage content from the database, including the layout, banners, and featured products.
* Presents the homepage to the user, showcasing a curated selection of books and highlighting promotions or new releases.

System Action (if user attempts to add to cart or wish list):

* Informs the user that they need to log in to perform actions like adding items to their cart or wish list.
* Provides a prompt for the user to log in or create an account.
  + 1. *Sorting and filtering*

1. User Action: Selects the genre filter at the left bar to browse books by genre.

System Response:

* Updates the displayed book listings to match the selected genre.
* Maintains the genre filter selection for subsequent browsing sessions.

1. User Action: Selects the Newest/Oldest filter on the top to sort books by their publication date.

System Response:

* Sorts the displayed book listings in ascending or descending order based on their publication date.
* Maintains the sorting preference for subsequent browsing sessions.

1. User Action: Continues browsing through the filtered book selection.

System Response:

* Loads additional book listings dynamically as the user scrolls, ensuring a seamless browsing experience.
* Provides real-time updates to the book selection as the user applies or removes filters.

## Special Requirements

* 1. **Multi-language support**
* Allow users to easily switch between languages and ensure that the multi-language support covers the whole user interface
  1. **Accessibility**
* Comply with accessibility standards such as WCAG (Web Content Accessibility Guidelines) to make the website accessible to users with disabilities.
  1. **Performance**
* Implement image optimization techniques to reduce load times without compromising quality.
* Use lazy loading for images and prioritize the loading of essential content first.
  1. **Security**
* Encrypt sensitive user information, such as passwords and payment details, using industry-standard encryption algorithms. Regularly update and patch server software to address potential security vulnerabilities.
  1. **Regular backup**
* Schedule automated regular backups of both book data and user preferences to ensure data recovery in the event of system failures or data loss.

## Preconditions

* 1. **Internet connectivity**
* The user's device should have a working internet connection to access the website
  1. **User authentication**
* The user must be logged into their account to gain access to the full library of book on the website
  1. **Previous user interaction**
* If the users have previously interacted with the website (e.g browsing, purchasing) then the website can generate personalized recommendations.
  1. **Product availability**
* Regularly update the inventory to ensure that all books displayed are in stock and available for purchase.
  1. **No conflicting processes**
* The user has not already initiated a different process within the website that conflicts with the browsing functionality (e.g., placing an order, viewing order history, accessing account settings).
  1. **No error**
* Implement effective error handling mechanisms, providing informative error messages and guidance on how users can resolve issues when they occur.

## Postconditions

* 1. **Filtered book list display**
* Provide the user with options to apply filters directly from the displayed book list, allowing for dynamic adjustments without returning to the main filter selection page.
  1. **User preferences retention**
* Enable users to customize and manage their saved preferences, ensuring a seamless experience across different sessions and devices.
  1. **Successful book details view**
* Implement a "Recently Viewed" section to help users revisit books they have previously clicked on.
  1. **Direct interaction options**
* Enable users to manage their shopping cart and wishlist directly from the browse page, allowing them to review or modify their selections without navigating away.
  1. **Product tags displayed**
* If the user view the books detail then the tags of that books are displayed and the user can click on those tags to view other books with the same tag
  1. **Activity logging**
* Enhance user engagement by incorporating a feature that dynamically updates tag suggestions based on the user's browsing history and preferences.

## Extension Points

* 1. **Review and rating**
* Allow users that have bought the book to provide reviews and ratings to enable other users to make an informed decision based on those reviews
  1. **Personalized recommendation**
* Implement an algorithm for personalized book recommendations to enhance user experience by suggesting books based on browsing history and preferences.
  1. **Social media integration**
* Enable users to create and share lists of their favorite books on their personal social media
* Implement social login options, making it easy for users to connect their social media accounts to their profile on the platform, streamlining the sharing process.
  1. **Book preview**
* Implement a feature that allows users to preview the first couple of pages of the book they are going to buy.

# Use-Case Specification: Add product to cart

## Use-Case Name

* 1. **Brief Description**

The "Add Product to Cart" use case in the Book Alley system allows registered users to conveniently select and add desired products to their shopping cart for subsequent purchase. This functionality enhances the overall shopping experience and streamlines the process of selecting items for checkout.

## Flow of Events

* 1. **Basic Flow**

1. User Action: Navigates to the product detail page of the book they want to purchase.

System Response:

* Retrieves the detailed information for the selected book from the database, including:
  + Full product description
  + Author biography
  + Customer reviews
  + Availability and pricing information

System Response: Presents the book detail page to the user, providing a comprehensive overview of the selected book.

1. User Action: Selects the desired quantity of the book using the quantity selector or input field.

System Response:

* Validates the entered quantity to ensure it is within the available stock and meets any minimum or maximum order requirements.
* Displays the updated cart subtotal based on the selected quantity.

1. User Action: Clicks on the "Add to cart" button on the product page.

System Response:

* Retrieves the user's cart information from the database or creates a new cart if the user does not have an existing cart.
* Adds the selected book to the user's cart along with the specified quantity.
* Updates the cart total to reflect the added item.
* Stores the updated cart information in the database.

System Action: Displays a confirmation message indicating that the selected book has been successfully added to the cart.

System Action: Presents the user with options to:

* Continue shopping: Redirects the user back to the homepage or the previous product listing page.
* Proceed to checkout: Redirects the user to the shopping cart page for order review and payment.

1. User Action: Chooses to proceed to checkout.

System Response: Redirects the user to the shopping cart page to review their order details and initiate the checkout process.

* 1. **Alternative Flows**
     1. *Insufficient stock*

1. User Action: Selects a book and enters a quantity that exceeds the available stock.

System Response:

* Detects that the selected quantity exceeds the available stock for the book.
* Displays a notification informing the user that the requested quantity is not available.

System Action: Provides the user with options to:

* Update the quantity to a value within the available stock.
* Choose an alternative product with sufficient stock.

1. User Action: Selects to update the quantity to a value within the available stock.

System Response:

* Updates the selected quantity to the specified value.
* Displays the updated cart subtotal based on the revised quantity.
* Provides the user with the option to proceed with adding the book to the cart with the updated quantity.

1. User Action: Selects to choose an alternative product.

System Response:

* Redirects the user back to the product listing page or the search results page.
* Highlights alternative products with similar features or from the same genre.
* Facilitates the user's search for a suitable replacement product.
  + 1. *Product unavailable*

1. User Action: Selects a book that is currently unavailable or discontinued.

System Response:

* Detects that the selected book is currently unavailable or discontinued.
* Displays a notification informing the user that the selected book is not currently available for purchase.

System Action: Provides the user with options to:

* Choose an alternative product with similar features or from the same genre.
* Contact customer support for further assistance or information about the product's availability.

1. User Action: Selects to choose an alternative product.

System Response:

* Redirects the user back to the product listing page or the search results page.
* Highlights alternative products with similar features or from the same genre.
* Facilitates the user's search for a suitable replacement product.

## Special Requirements

* 1. **Usability**
* The "Add to Cart" button must be prominently displayed on the product page. The language used on the button should also be clear and concise, such as "Add to Cart," to avoid any ambiguity about its purpose.
  1. **Accessibility**
* The web application must adhere to accessibility standards, ensuring that it is usable for people with disabilities.
  1. **Performance**
* The process of adding a product to the cart must be fast and responsive, with a delay of under 2-3 seconds. This ensures a seamless user experience and prevents frustration caused by slow interactions.
  1. **Security**
* The shopping cart can only be accessed by the people that have access to the account. Cart contents must be encrypted during transmission and security measures should be applied to prevent common threats
  1. **Reliability**
* The "Add to Cart" function must be highly reliable to ensure that products are added correctly and consistently remain in the cart until the user either completes the purchase or removes them.

## Preconditions

* 1. **Internet connectivity**
* The user's device should have consistent access to the internet to ensure a smooth experience during product selection and addition to the cart.
  1. **User authentication**
* Proper authentication ensures the security and personalization of the shopping experience.
  1. **Product page navigation**
* The user should be on the product page of the book they intend to add to the cart.
  1. **Product availability**
* The system should verify that there is enough stock of the selected book before allowing the user to add it to their cart.
  1. **No conflicting processes**
* The user should not have initiated a conflicting process within the website that interferes with the "Add to Cart" functionality like viewing orders and access to account setting
  1. **No error**
* The user should not have encountered any critical errors during the browsing or accessing of the product page that could impact the buying process.

## Postconditions

* 1. **Successful addition to cart**
* The system should provide immediate feedback to the user, announcing the successful addition. This notification should be clear and visible, confirming that the desired product has been added to the cart.
  1. **Update cart contents**
* Following a successful addition to the cart, the cart's contents should be updated to reflect the newly added product. The system should display the added product along with the chosen quantity, and the total cost of the cart must be recalculated and updated accordingly.
  1. **Shopping options**
* After adding a product to the cart, the user should be presented with options to either continue shopping or proceed to checkout.
  1. **Cart persistence**
* The shopping cart should remain accessible at all times when the user is on the website. Even if the user logs out and logs back in, the contents of the cart should persist.
  1. **Synchronize cart**
* To enhance the user experience, the shopping cart should remain consistent across multiple devices.
  1. **Quantity update**
* If the user add the same product again, that product within the cart will automatically update its quantity
* User can also update the quantity of any products that have been added to cart within the shopping cart page

## Extension Points

* 1. **Customized discount**
* To enhance the marketing and promotional aspects of the platform, the system should allow for the integration of custom discount and promotion modules.
  1. **Shipping integration**
* To cater to diverse user preferences, the system should allow for the integration of various shipping methods and carriers.

# Use-Case Specification: Checkout

## Use-Case Name

* 1. **Brief Description**

The checkout process at Book Alley is designed to provide a seamless and convenient experience for users looking to finalize their book purchases. Whether customers prefer traditional cash transactions or the ease of online payments, Book Alley caters to both preferences.

## Flow of Events

* 1. **Basic Flow**

1. User Action: Browses the website and adds desired books to their shopping cart by clicking on the "Add to Cart" button for each item.

System Response:

* Validates the selected books and ensures they are available in sufficient quantities.
* Adds the selected books to the user's shopping cart, along with the specified quantities.
* Updates the cart total to reflect the added items.
* Stores the updated cart information in the database.
* Provides the user with options to continue shopping or proceed to checkout.

1. User Action: Navigates to the shopping cart page to review their selected items and order details.

System Response:

Retrieves the user's cart information from the database and displays it on the shopping cart page.

Presents the user with a summary of their order, including:

* List of items with their titles, quantities, and prices
* Total order amount
* Applicable taxes and shipping charges

1. User Action: Clicks on the "Checkout" button to initiate the checkout process.

System Response:

* Checks if the user is logged in.
* If the user is not logged in: Prompts the user to log in using their existing account or create a new account.
* If the user is already logged in: Proceeds to the next step of the checkout process.

1. User Action: Provides shipping details, including their full name, shipping address, and preferred shipping method.

System Response:

* Validates the entered shipping information to ensure it is complete and accurate.
* Displays the estimated shipping cost based on the selected shipping method and shipping address.
* Updates the order total to reflect the shipping charges.

1. User Action: Enters payment details, such as cash, credit card, or online payment information.

System Response:

* Validates the entered payment information to ensure it is correct and secure.
* Processes the payment information and initiates payment authorization.
* Displays a confirmation message indicating successful payment authorization.

1. User Action: Reviews the order summary, including items, quantities, shipping information, and total cost.

System Response: Presents the user with a comprehensive order summary, including:

* Order details (items, quantities, prices)
* Shipping information (name, address, shipping method)
* Payment information (payment method, transaction details)
* Total order cost (including items, taxes, shipping)

1. User Action: Confirms the order and clicks on the "Place Order" button.

System Response: Processes the order, including:

* Creating an order record in the database
* Updating inventory levels for the ordered items
* Sending order confirmation emails to the user and the seller
* Triggering order fulfillment processes

System Action: Displays a confirmation message with the order details, including:

* Order number
* Order date
* Order summary (items, quantities, prices)
* Shipping information
* Payment information

System Action: Provides the user with options to:

* Track order status
* Print order confirmation
* Contact customer support for any further assistance
  1. **Alternative Flows**
     1. *Payment failure*

1. User Action: Enters payment details and clicks on the "Place Order" button.

System Response:

* Processes the payment information through the payment gateway.
* Detects that the payment transaction has failed due to reasons such as insufficient funds, declined card, or invalid payment details.

System Action:

* Displays an error message informing the user that the payment transaction has failed.
* Provides the user with options to:
  + Review and correct the entered payment information.
  + Try an alternative payment method (e.g., different credit card, cash on delivery).
  + Contact customer support for assistance with payment issues.

1. User Action: Chooses to retry payment by updating or changing the payment information.

System Response:

* + Allows the user to modify their payment details and re-enter them.
  + Processes the updated payment information through the payment gateway.

System Action:

* + If the payment transaction is successful, proceed with the checkout process from the order review step.
  + If the payment transaction fails again, prompt the user to contact customer support for further assistance.
    1. *Successful payment*

1. User Action: Enters valid payment details and clicks on the "Place Order" button.

System Response:

* Processes the payment information through the payment gateway.
* Receives a successful payment authorization response.

System Action:

* Displays a confirmation message indicating successful payment authorization.
* Proceeds with the checkout process from the order review step.

## Special Requirements

* 1. **Secure transaction**
* Prioritizes the security of user transactions by implementing industry-standard encryption protocols.
  1. **Inventory management**
* Book Alley's inventory management system operates in real-time, constantly updating the availability of books as purchases are made.
  1. **User authentication**
* Only authorized users (users that have logged into their account) can place orders and proceed to the checkout page.
  1. **Shipping options**
* The system should provide multiple shipping options with associated costs for the user to choose from during the checkout process.

## Preconditions

* 1. **Internet connectivity**
* Book Alley's online checkout system relies on a stable internet connection for the best experience.
  1. **User is authenticated**
* Users must be logged into their account in order to proceed with the checkout process.
  1. **Items in cart**
* To proceed with the checkout process, users must have items in their shopping cart.
  1. **System availability**
* The availability of the website is a critical precondition for a successful checkout process.
  1. **Valid address**
* Users are required to provide a valid shipping address during the checkout process.
  1. **Valid payment information**
* Valid payment information, such as credit card details or PayPal, is essential for completing the checkout process. Book Alley's system securely processes and stores payment information in compliance with industry standards, safeguarding user data against unauthorized access.

## Postconditions

* 1. **Order confirmation**
* Include a prominent call-to-action on the order confirmation page, confirming that their order have successfully been placed or have failed and need a replacement.
  1. **Payment authorization**
* Display a clear message on the payment authorization page, reassuring users that their payment has been successfully authorized and providing an estimated timeframe for fund capture.
  1. **Inventory update**
* After a successful purchase, the item that the user has bought will be automatically removed from their cart or their quantity will be updated
  1. **Confirmation email**
* Customize the confirmation email template to include personalized elements such as the user's name and a thank-you message for their purchase, enhancing the overall user experience.
  1. **Order tracking**
* Provide a dedicated and easily accessible order tracking page within the user's account, displaying real-time updates on the status of their order, expected delivery date, and tracking number.

## Extension Points

* 1. **Multiple payment methods**
* Offer a diverse range of payment options, including credit/debit cards, digital wallets, cash-on-delivery and other region-specific payment methods, catering to a broader audience.
  1. **Vouchers**
* Provide users with the ability to apply multiple vouchers or discount codes during the checkout process, fostering a sense of flexibility and increased satisfaction.
  1. **Address book**
* Allow users to save, edit, and delete multiple shipping addresses in their address book, implement an option for users to set a default shipping address, ensuring a faster and more convenient checkout experience.
  1. **Order customization**
* Integrate a user-friendly interface during the checkout process that allows users to add personalized messages, select gift wrapping options, or choose from other customization features.
  1. **Save shopping cart**
* Allow users to save their current shopping cart for future sessions, promoting a convenient and seamless shopping experience over multiple visits.

# Use-Case Specification: Review and Rating

## Use-Case Name

* 1. **Brief Description**

The review and rating use case for Book Alley enables user to leave a review for the book that they have purchased in order to give other customers a better idea of what product they are going to buy

## Flow of Events

* 1. **Basic Flow**

1. User Action: Locates the book they purchased and navigates to its product page.

System Response: Retrieves and displays the book's product page, including its details, reviews, and purchase options.

1. User Action: Scrolls down to the review section of the product page.

System Response: Displays the book's existing reviews, if any, and provides a prominent "Write a Review" or similar button.

1. User Action: Clicks on the "Write a Review" button to initiate the review process.

System Response: Opens a review form with fields for:

* Review title
* Review text
* Star rating (e.g., 1-5 stars)

1. User Action: Provides their review title, enters their review text, and selects a star rating based on their satisfaction with the book.

System Response: Validates the entered review information to ensure it meets any content guidelines or restrictions.

System Action (if validation is successful):

* Saves the user's review in the database, associating it with the book being reviewed.
* Displays a confirmation message indicating successful review submission.

System Action (if validation fails):

* Displays an error message explaining the specific validation errors.
* Prompts the user to correct the invalid information.

System Action (upon successful correction):

* Continues the review submission process and saves the validated review.
  1. **Alternative Flows**
     1. *Rejected review*

1. User Action: Writes a review and submits it for publication.

System Response:

* Performs content moderation on the submitted review to ensure it adheres to the community guidelines.
* Detects inappropriate content or potential spam violations in the review.

System Action:

* Rejects the submitted review due to the identified violations.
* Displays a message to the user informing them that their review has been rejected.
* Provides the user with an explanation of the reasons for rejection, highlighting the specific violations of the community guidelines.

1. User Action: Reviews the feedback and revises their review to comply with the community guidelines.

System Response: Allows the user to edit and resubmit their review.

System Action (if revised review meets guidelines):

* Accepts the revised review and publishes it on the product page.
* Sends a notification to the user informing them that their review has been approved.

System Action (if revised review still violates guidelines):

* Maintains the rejection status and prompts the user to further revise their review.
* Offers guidance and resources on how to write appropriate and compliant reviews.

## Special Requirements

* 1. **Spam protection**
* Implement robust spam protection measures. These could include reCAPTCHA, a widely used anti-bot system, content analysis to identify and flag potentially spammy reviews.
  1. **User authentication**
* To prevent anonymous reviews and ensure accountability, require users to sign in or create an account before submitting a review.
  1. **Review editing**
* Provide users with the flexibility to edit their reviews after submission, allowing them to correct any errors, refine their thoughts, or provide additional insights based on further experience with the product or service.
  1. **View existing reviews**
* Enable users to browse and read existing reviews submitted by other customers, gaining valuable insights into the product's quality, features, performance, and overall user experience.
  1. **Sort reviews**
* Enable users to browse and sort reviews submitted by other customers, gaining valuable insights into the product's quality, features, performance, and overall user experience..

## Preconditions

* 1. **Internet connectivity**
* A stable internet connection is essential for users to access the website and submit their reviews.
  1. **User is logged in**
* To ensure accountability and prevent anonymous reviews, users must be logged into their accounts before submitting their evaluations.
  1. **System availability**
* Reviews can only be submitted when the website is up and running. If the website is unavailable, users will encounter an error message and be unable to proceed with their review submission.
  1. **Book availability**
* To prevent reviews for non-existent or unavailable products, the book being reviewed must be present in the catalog and accessible to users.
  1. **Purchase History**
* For product reviews, users should have a verified purchase history or demonstrated experience with the product they are evaluating.
  1. **Review Frequency**
* To prevent spamming and maintain the authenticity of reviews, users should be limited in the frequency with which they can submit reviews for the same product or service.

## Postconditions

* 1. **Review displayed**
* Upon successful submission, the user's review is displayed prominently within the book's review section, making it visible to other users seeking insights into the book's content and overall quality.
  1. **Rating updated**
* The overall rating for the book is dynamically updated to reflect the newly submitted review, providing a more accurate representation of the book's perceived value and appeal to potential readers.
  1. **Notification**
* A confirmation notification is promptly delivered to the user's account, acknowledging the successful submission of their review and providing feedback on the review's processing timeline.
  1. **Moderation process**
* Initiated by the review's submission, the moderation process ensures that reviews adhere to community guidelines and maintain the platform's integrity.
  1. **Review editing**
* If the system allows review editing, users retain the ability to modify their submitted reviews after submission.
  1. **Sort & filter reviews**
* Users are provided with comprehensive sorting and filtering options to refine their review browsing experience.

## Extension Points

* 1. **Review comment system**
* Introduce a review comment system to encourage active engagement and meaningful discussions among users.
  1. **Localized review**
* Expand the platform's reach by supporting reviews in multiple languages, enabling users to contribute and consume reviews in their preferred language.
  1. **Review filtering**
* Empower users to personalize their review browsing experience by providing advanced sorting and filtering by rating, recency, helpfulness, and other relevant criteria.

# Use-Case Specification: Moderate reviews

## Use-Case Name

* 1. **Brief Description**

The moderate reviews for admin use cases empowers administrators to assess and filter customer reviews submitted for products. This ensures that the content aligns with community guidelines and provides valuable insights for potential customers.

## Flow of Events

* 1. **Basic Flow**

1. User Action: Submits a review after purchasing and reading a book.

System Response:

* Captures the user's review and stores it in the database.
* Automatically flags the new review for moderation based on predefined criteria (e.g., keyword detection, sentiment analysis).

System Action: Places the flagged review in a moderation queue, accessible to the admin.

1. Admin Action: Reviews each flagged review to assess its compliance with the community guidelines and policies.

System Response: Provides the admin with tools and resources to facilitate the review process, such as:

* Highlighting potential violations in the review text.
* Providing access to the user's profile and purchase history for context.
* Comparing the review to similar reviews for the same book or other products.

1. Admin Action: Makes a decision regarding the flagged review, selecting one of the following options:

* Approved: The review is deemed compliant and published on the website.
* Rejected: The review is found to violate the community guidelines and rejected.
* Pending: The review requires further investigation or clarification before a decision can be made.

System Action: Notifies the user of the status of their review.

* For approved reviews:
  + Sends a notification informing the user that their review has been published.
  + Updates the book's overall rating and review statistics.
* For rejected reviews:
  + Sends a notification to the user informing them that their review has been rejected.
  + Provides clear explanations for the rejection, citing specific violations of the community guidelines.
  + Offers the user an opportunity to appeal the decision or revise their review to comply with the guidelines.
* For pending reviews:
  + Sends a notification to the user informing them that their review is under review.
  + Requests additional information or clarification from the user if needed.
* Approved Reviews:
  + Are published on the website and become publicly visible.
  + Contribute to the book's overall rating and review statistics.
  + Provide valuable insights for other users considering purchasing the book.
  1. **Alternative Flows**
     1. *Appeal review*

1. User Action: Submits an appeal against the moderation decision for their review.

System Response:

* Captures the user's appeal and directs it to the appropriate reviewer, either the admin or a dedicated appeals team.
* Notifies the user that their appeal has been received and will be reviewed.
* Provides the user with a reference number or tracking link to monitor the status of their appeal.

1. Admin: Receives the user's appeal and reviews the flagged review again.

System Response:

* Provides the admin or appeals team with access to the original review, the moderation decision, and the user's appeal.
* Facilitates a thorough review of the review, considering the user's appeal, any additional context provided, and the community guidelines.

1. Admin: Makes a final decision regarding the review, either:

* Upholding the original decision: The review remains rejected or pending, and the admin provides a detailed explanation to the user, reiterating the reasons for the decision.
* Overturning the original decision: The review is approved and published on the website. The admin informs the user of the decision and provides an apology for any inconvenience caused.

System Action: Notifies the user of the final decision.

* If the review is approved:
  + Sends a notification to the user informing them that their appeal has been successful and their review has been published.
  + Updates the book's overall rating and review statistics.
* If the original decision is upheld:
  + Sends a notification to the user informing them that their appeal has been unsuccessful and explaining the reasons for upholding the original decision.
  + Provides the user with additional resources or guidance on how to write appropriate and compliant reviews.

## Special Requirements

* 1. **Security and privacy**
* Implement robust security measures that comply with all applicable data privacy regulations to protect user data, including encryption, access controls, and regular security audits.
  1. **Scalability**
* Design a streamlined review workflow that prioritizes high-risk content and ensures timely moderation
* Implement a scalable infrastructure that can handle increasing volumes of user-generated content without compromising performance and security.
  1. **User experience**
* Establish clear and easy-to-understand community guidelines that outline acceptable and prohibited behavior on the e-commerce platform..
* Provide a reporting mechanism for users to flag inappropriate or offensive content.
  1. **Transparency**
* Actively engage with the e-commerce platform community to gather feedback on the moderation process.
  1. **Adaptability**
* Continuously monitor evolving content trends and potential new challenges in user-generated content.
* Be prepared to adapt the moderation system to changes in user behavior.

## Preconditions

* 1. **Internet connectivity**
* The admin's device should have a working internet connection to access the website..
  1. **User authentication**
* Only authenticated users should be able to submit reviews. Only users who have logged in to their accounts should be able to submit reviews.
  1. **Review submission completeness**
* A review should have the required fields completed before submission, such as a rating and textual content. This ensures that the review is informative and helpful to other users.
  1. **Compliance with guidelines**
* Reviews should adhere to the community guidelines and policies specified by the e-commerce platform.
  1. **Content filtering**
* Implement content filtering mechanisms to catch common issues before manual review. This can include things like checking for profanity, spam, and other types of inappropriate content.

## Postconditions

* 1. **Review publication**
* After approval, the review is published on the website and contributes to the product's overall rating.
  1. **Notification to user**
* Users receive timely notifications about the status of their reviews, whether approved, rejected, or pending.
  1. **Continuous monitoring**
* Implement continuous monitoring of published reviews to address any issues that may arise post-publication.
  1. **Feedback loop**
* Establish a feedback loop for users to report issues with reviews even after they are published, ensuring a dynamic and responsive system.

## Extension Points

* 1. **User reporting tools**
* Implementing user reporting tools for maintaining a high-quality review system on an e-commerce platform. These tools flag inappropriate or irrelevant content for review and will significantly reduce the workload.
  1. **Custom notifications**
* Allow administrators to manage the flow of moderation-related alerts and ensure that they are notified promptly about critical issues.