
Use Cases

for

Online Bookstore

Version 1.0 approved

Prepared by B4

Team members

Edward Cho, David DiGioia, Logan Hutcheson, Stefan Tobler, Steven Xu

Revision History

Name	Date	Reason For Changes	Version
Edward Cho	9-14	Created document	1.0
Stefan Tobler	9-14	Added some Primary Actors	1.1
David DiGioia	9-15	Re-organized list to match instructions from eLC	1.2
Stefan Tobler Edward Cho Logan Hutcheson David DiGioia Steven Xu	9-16	Added details to tables	1.3
David DiGioia	9-17	Added use case diagram	2.0

Use Case List

ID#	Primary Actor	Use Cases
1	Web user	Create Account
2	Registered User	Login
3	Logged-in User	Edit Profile
4	Web user, logged-in user	Search and Filter
5	Web user, logged-in user	View Book details
6	Web user, logged-in user	Add Book to cart
7	Logged-in user	View order history
8	Logged-in user	Checkout
9	Logged-in user	Add Promotion
10	Logged-in user	Reset password
11	Logged-in user	Return Book(s)
12	Admin	Manage Books
13	Admin	Add New Book
14	Admin	Delete Book
15	Admin	Edit Book
16	Admin	Manage Promotions
17	Admin	Add New Promotion
18	Admin	Edit Existing Promotion
19	Admin	Delete Existing Promotion
20	Admin	Manage Users
21	Admin	Edit Prices
22	System	Display Order Total
23	Web user, Logged-in User	Enter Payment Info
24	System	Calculate order total payment
25	System	Send order confirmation email
26	System	Send account verification email
27	Logged-in user	Reorder Book
28	Admin	Promote User
29	Admin	Demote Admin
30	Admin	Suspend User
31	Web user, Logged-in User	Enter personal info
32	Web user, Logged-in User	Enter address info
33	Registered User	Unsubscribe from promotions
34	Web user, Logged-in User	Manage shopping cart
35	Admin	Archive books

Actors:

Web User: Any user; Can browse for books but cannot purchase without an account

Registered User: Web user that has created an account; Must login with credentials

Logged-in User: Web user logged into the system; Can purchase books

Admin: Can enter/remove information about books, employees, users, and promotions into system

System: Automated system; Handles internal tasks

Create Account

Use Case ID:	1		
Use Case Name:	Create Account		
Created By:	Edward Cho	Last Updated By:	Edward Cho
Date Created:	9/14	Date Last Updated:	9/17

Actors:	Web user	
Brief Description:	User creates an account to access the system	
Preconditions:	1. User does not have an account	
Postconditions:	1. User receives a verification email to activate their account	
Related use cases	Enter Address Info, Enter Payment Info, Enter Personal Info, Send Verification Email	
Normal Flow:	Actor	System
	<ol style="list-style-type: none"> 1. User enters personal information into registration form 2. User can enter address info. 3. User can enter payment info. 4. User selects Save 	<ol style="list-style-type: none"> 1.1. System adds a new user to the database 1.2. System invokes Enter Personal Info 2.1. System invokes Enter Address Info 3.1. System invokes Enter Payment Info 5.1. System invokes Send Verification Email
Alternative Flows:	none	
Exceptions:	Account already exists	
Priority:		
Special Requirements:		
Assumptions:		
Notes and Issues:		

Login

Use Case ID:	2		
Use Case Name:	Login		
Created By:	David DiGioia	Last Updated By:	Stefan Tobler
Date Created:	9-15	Date Last Updated:	9-17

Actors:	Registered User	
Brief Description:	User enters credentials to login to the website	
Preconditions:	User has previously created an account User enters correct credentials	
Postconditions:	User is logged in on his/her browser	
Related use cases	Create Account	
Normal Flow:	Actor	System
	1. User enters account ID or email address 2. User enters password 3. User clicks login	3.1. System checks validity of credentials 3.2. System logs in user
Alternative Flows:	If credentials are incorrect, the system will give the user an error message	
Exceptions:	User is suspended, account does not exist, user has not activated account via email	
Priority:		
Special Requirements:		
Assumptions:	User is active, not suspended, and has the correct credentials	
Notes and Issues:		

Edit Profile

Use Case ID:	3		
Use Case Name:	Edit Profile		
Created By:	David DiGioia	Last Updated By:	David DiGioia
Date Created:	9-16	Date Last Updated:	9-16

Actors:	Logged-in User	
Brief Description:	Users may change details about their account such as name, phone number, password, shipping address, and payment info.	
Preconditions:	User must be logged in to a valid account Account must be verified from the verification email	
Postconditions:	Changes to profile will be updated immediately.	
Related use cases	Create Account, Login	
Normal Flow:	Actor	System
	1. User clicks edit profile 2. User clicks the field they wish to change 3. User enters updated information 4. User Clicks 'Save changes'	1.1. System displays the 'edit profile' page 2.1. Field becomes selected and editable 4.1. The changes are saved in the database and the changes are visible to the user immediately
Alternative Flows:		
Exceptions:		
Priority:		
Special Requirements:	User can exit choosing to save or not to save their changes	
Assumptions:		
Notes and Issues:		

Search and Filter

Use Case ID:	4		
Use Case Name:	Search and Filter		
Created By:	David DiGioia	Last Updated By:	David DiGioia
Date Created:	9-16	Date Last Updated:	9-16

Actors:	Web user, logged-in user	
Brief Description:	Users can search for keywords to find books. Also filters such as date, author, genre, and price can be applied when searching.	
Preconditions:		
Postconditions:	The system returns relevant search results to the user.	
Related use cases	View Book Details	
Normal Flow:	Actor	System
	1. User enters search terms 2. User selects filters they wish to apply 3. User presses 'search' button	3.1. System displays results
Alternative Flows:	User may choose to enter only search terms or only filters.	
Exceptions:	If there are no results for a particular search, the system will display an error message	
Priority:		
Special Requirements:		
Assumptions:		
Notes and Issues:		

View Book Details

Use Case ID:	5		
Use Case Name:	View Book Details		
Created By:	David DiGioia	Last Updated By:	Stefan Tobler
Date Created:	9-16	Date Last Updated:	9-18

Actors:	Web user, Logged-in user	
Brief Description:	Upon clicking a book title, the user sees details such as ISBN, category, author's name, title, cover page, edition, publisher, publication year, price	
Preconditions:	The bookstore is currently selling the desired book	
Postconditions:	The system displays book details to the user	
Related use cases	Search and Filter, Manage Books	
Normal Flow:	Actor	System
	1. User clicks on book title from search page	1.1 System displays new webpage with details about the book
Alternative Flows:		
Exceptions:	If the bookstore is not longer selling the book, an error page will be shown instead of the old details page	
Priority:		
Special Requirements:		
Assumptions:	The bookstore is currently selling the desired book	
Notes and Issues:		

Add Book to Cart

Use Case ID:	6		
Use Case Name:	Add Book to Cart		
Created By:	David DiGioia	Last Updated By:	Stefan Tobler
Date Created:	9-16	Date Last Updated:	9-18

Actors:	Web user, logged-in user	
Brief Description:	User may add book to cart so that they may easily purchase it later	
Preconditions:	The book must not be archived and the book must be in stock	
Postconditions:	The book is in the user's cart	
Related use cases	Checkout	
Normal Flow:	Actor	System
	1. User clicks add to cart by book's title	1.1. System adds book to user's cart
Alternative Flows:		
Exceptions:	The book is archived or is out of stock.	
Priority:		
Special Requirements:		
Assumptions:	The book is in stock and not archived	
Notes and Issues:		

View Order History

Use Case ID:	7		
Use Case Name:	View Order History		
Created By:	David DiGioia	Last Updated By:	Stefan Tobler
Date Created:	9-16	Date Last Updated:	9-18

Actors:	Logged-in user	
Brief Description:	User may view a list of all books they have ordered in the past, and see the status of books which have not yet been delivered	
Preconditions:	User is logged in to an active account	
Postconditions:	List of previously ordered books is displayed	
Related use cases	Checkout, Login	
Normal Flow:	Actor	System
	1. User clicks 'view order history' 2. User clicks specific order to view details or status	1.1. System displays page listing all previous orders 2.1. System displays details of order
Alternative Flows:	If the order selected hasn't been delivered yet, the system will display its delivery status	
Exceptions:	User has no order history	
Priority:		
Special Requirements:		
Assumptions:		
Notes and Issues:		

Checkout

Use Case ID:	8		
Use Case Name:	Checkout		
Created By:	David DiGioia	Last Updated By:	Stefan Tobler
Date Created:	9-16	Date Last Updated:	9-18

Actors:	Logged-in user	
Brief Description:	User purchases all the items in their cart	
Preconditions:	At least one item must be in the user's cart User enters valid payment information User enters valid shipping address	
Postconditions:	System sends order confirmation email and order status is updated	
Related use cases	Send Order Confirmation Email, Calculate Order Total Payment, Display Order Total, Add to Cart	
Normal Flow:	Actor	System
	1. User goes to cart 2. User clicks 'checkout' 3. User enters payment and shipping info 4. User clicks 'place order'	1.1. System calculates order total payment 1.2. System displays order total payment 2.1. System displays page to enter payment and shipping info 4.1. System charges money from credit card 4.2. System shows page saying order is successful and giving details and arrival date 4.3 Send confirmation email 4.4 Decrease book's stock by purchase amount
Alternative Flows:	If the user has saved shipping address or payment info, those steps may be skipped	
Exceptions:	If the book is out of stock, the system will not charge the user's card and will instead show an error message If there is an error in processing the payment, the order will not be placed and instead an error message will be shown If the book is archived, the system will show an error	
Priority:		
Special Requirements:		
Assumptions:	The books in the user's cart are in stock	
Notes and Issues:		

Add Promotion

Use Case ID:	9		
Use Case Name:	Add Promotion		
Created By:	Stefan Tobler	Last Updated By:	Stefan Tobler
Date Created:	9/16	Date Last Updated:	9/18

Actors:	Admin	
Brief Description:	Admin wants to add a new promotion that gets sent to customers.	
Preconditions:	Must be logged in as admin	
Postconditions:	Users get sent promotions	
Related use cases		
Normal Flow:	Actor	System
	1. Admin clicks on button to add new promotion 2. Admin fills out promo-code, percentage off, start-date, end-date, and expiration date 3. Admin confirms new promotion and sends it to customers.	3.1 System takes promotional information and stores it in a database. 3.2 System sends the promotional email to subscribed users
Alternative Flows:		
Exceptions:	The promotion already exists	
Priority:		
Special Requirements:		
Assumptions:	Customer's that are receiving promotions have signed up	
Notes and Issues:	Should users who are subscribed to promotions but have a deactivated account still be sent promotions?	

Reset Password

Use Case ID:	10		
Use Case Name:	Reset Password		
Created By:	Stefan Tobler	Last Updated By:	Stefan Tobler
Date Created:	9/16	Date Last Updated:	9/18

Actors:	Registered-user	
Brief Description:	If a user forgets his/her password, he/she may want to reset it.	
Preconditions:	The user has registered email and an active account	
Postconditions:	The system updates the user's password	
Related use cases	Register	
Normal Flow:	Actor	System
	1. User navigates to the login page 2. User clicks "Forgot password?" button 3. User enters the email he or she registered with 4. User receives password reset email 5. User clicks on the link in the password reset email and is prompted to enter a new password 6. User submits a new password and can not login.	3.1. System checks if the entered email is valid 4.1. System emails user with a unique password reset link with a set time to live 6.1. When a user submits a new password the system will update the database with that password
Alternative Flows:	If the system determines that the email does not exist in the system as a registered user	
Exceptions:	Not registered user	
Priority:	Medium	
Special Requirements:		
Assumptions:	User has access to email account	
Notes and Issues:		

Return Book(s)

Use Case ID:	11		
Use Case Name:	Return Book(s)		
Created By:	Stefan Tobler	Last Updated By:	Stefan Tobler
Date Created:	9/16	Date Last Updated:	9/16

Actors:	Logged-in User, Admin	
Brief Description:	User gets a book and decides that he or she does not want it; Made an order by mistake.	
Preconditions:	Book ordered	
Postconditions:	System initiates a return or sets the order status to canceled	
Related use cases	View Order History	
Normal Flow:	Actor	System
	1. User finds “return book” option in “Order History” 2. User fills in required feedback about why book is being returned 3. Customer receives confirmation email 4. The customer sends the book back. 5. Once book has been received Admin marks it in the system	2.1 System notes the reason for return 3.1 System marks the book as being returned 4.1 After admin marks the book as returned, system notifies customer that return has been received. 4.2 System refunds the customer 4.3 System updates stock
Alternative Flows:	The book has not been shipped, cancel the order, go straight to 4.1 in system	
Exceptions:	The book is damaged, destroyed, or lost	
Priority:		
Special Requirements:		
Assumptions:		
Notes and Issues:		

Manage Books

Use Case ID:	12		
Use Case Name:	Manage Books		
Created By:	Edward Cho	Last Updated By:	Stefan Tobler
Date Created:	9/16	Date Last Updated:	9/18

Actors:	Admin	
Brief Description:	The system administrator can add, delete, or edit book information into the system	
Preconditions:	Admin is logged in	
Postconditions:	The database saves book changes	
Related use cases	Add New Book, Delete Book	
Normal Flow:	Actor	System
	1. Admin selects Add Book 2. Admin selects Delete Book 3. Admin selects Edit Book	1.1. System invokes Add New Book 2.1. System invokes Delete Book 3.1. System invokes Edit Book
Alternative Flows:	Admin can select any of steps 1-3, in any order	
Exceptions:		
Priority:		
Special Requirements:		
Assumptions:	Inputs are valid	
Notes and Issues:		

Add New Book

Use Case ID:	13		
Use Case Name:	Add New Book		
Created By:	Edward Cho	Last Updated By:	Stefan Tobler
Date Created:	9/16	Date Last Updated:	9/18

Actors:	Admin	
Brief Description:	The administrator can add a new book by recording all necessary information into the system	
Preconditions:	The admin is logged in	
Postconditions:	The database adds the new book	
Related use cases	Manage Books	
Normal Flow:	Actor	System
	1. Admin enters following information into system: <ul style="list-style-type: none"> • ISBN • category • authors' names • title • cover picture • edition • publisher • publication year • quantity-in-stock • minimum threshold • buying price • selling price 2. Admin selects "Save"	1.1. System checks that all fields are filled 2.1. System only updates book information specified by the admin in the database
Alternative Flows:	none	
Exceptions:		
Priority:		
Special Requirements:	Users can exit choosing to save or not save the book information	
Assumptions:	Information entered by admin is valid	
Notes and Issues:		

Delete Book

Use Case ID:	14		
Use Case Name:	Delete Book		
Created By:	Edward Cho	Last Updated By:	Stefan Tobler
Date Created:	9/16	Date Last Updated:	9/18

Actors:	Admin	
Brief Description:	The administrator can remove a book from the system	
Preconditions:	The admin is logged in	
Postconditions:	The book is removed from the database	
Related use cases	Manage Books	
Normal Flow:	Actor	System
	1. Admin chooses from a list of books 2. Admin provides confirmation to delete book	1.1. System prompts admin with a confirmation window 2.1. System sets the book to archived
Alternative Flows:	none	
Exceptions:	The book is already archived	
Priority:	low	
Special Requirements:		
Assumptions:	The book is not archived	
Notes and Issues:		

Edit Book

Use Case ID:	15		
Use Case Name:	Edit Book		
Created By:	Edward Cho	Last Updated By:	Edward Cho
Date Created:	9/16	Date Last Updated:	9/16

Actors:	Admin	
Brief Description:	The administrator can update book information	
Preconditions:	The admin is logged into the system	
Postconditions:	The database updated the changes for the book	
Related use cases	Manage Books	
Normal Flow:	Actor	System
	1. Admin chooses from a list of books 2. Admin edits desired information 3. Admin selects "Save"	1.1. System displays form with fields for all book info 3.1. System updates information in database
Alternative Flows:		
Exceptions:		
Priority:	high	
Special Requirements:	Admin can exit choosing to save or not save the book information	
Assumptions:	Information entered by admin is valid	
Notes and Issues:		

Manage Promotions

Use Case ID:	16		
Use Case Name:	Manage Promotions		
Created By:	Logan Hutcheson	Last Updated By:	Stefan Tobler
Date Created:	9/16	Date Last Updated:	9/18

Actors:	Admin	
Brief Description:	The admin can create and interact with existing promotions.	
Preconditions:	The admin must be logged into the system.	
Postconditions:	Promotion changes are saved into the system.	
Related use cases	Add New Promotion, Edit Existing Promotion, Delete Existing Promotion.	
Normal Flow:	Actor	System
	1. Admin selects Add New Promotion 2. Admin selects Edit Existing Promotion 3. Admin selects Delete Existing Promotion	1.1. System invokes Add New Promotion 2.1. System invokes Edit Existing Promotion 3.1. System invokes Delete Existing Promotion
Alternative Flows:	Admin can select any of steps 1-3	
Exceptions:		
Priority:	high	
Special Requirements:		
Assumptions:	Valid inputs are checked by the invoked use cases	
Notes and Issues:		

Add New Promotion

Use Case ID:	17		
Use Case Name:	Add New Promotion		
Created By:	Logan Hutcheson	Last Updated By:	Logan Hutcheson
Date Created:	9/16	Date Last Updated:	9/16

Actors:	Admin	
Brief Description:	The admin can create a new Promotion.	
Preconditions:	The admin is logged into the system.	
Postconditions:	The new promotion is stored in the database.	
Related use cases	Manage Promotions	
Normal Flow:	Actor	System
	1. Admin enters following information into system: <ul style="list-style-type: none"> Promo-code Percentage Start date Expiration date 2. Admin selects "Save"	2.1 System prompts admin with a confirmation window 2.2. System checks that all inputs are provided and valid 2.3. System enters new promotion into database 2.4. System sends email with the promotion information to subscribed users 2.5. Systems disables editing and deletion of the promotion
Alternative Flows:	none	
Exceptions:	Promotion title or code already exists	
Priority:	high	
Special Requirements:		
Assumptions:	Promotion does not already exist	
Notes and Issues:		

Edit Existing Promotion

Use Case ID:	18		
Use Case Name:	Edit Existing Promotion		
Created By:	Logan Hutcheson	Last Updated By:	Logan Hutcheson
Date Created:	9/16	Date Last Updated:	9/16

Actors:	Admin	
Brief Description:	The admin can change values for an existing promotion.	
Preconditions:	The admin is logged into the system.	
Postconditions:	The new values for the promotion are stored in the database.	
Related use cases	Manage Promotions	
Normal Flow:	Actor	System
	1. Admin selects a promotion from a list of existing promotions 2. Admin makes changes from existing fields 3. Admin selects "Save".	1.1 Systems displays editable fields for selected promotion 3.1. System checks that all values are filled and valid 3.2. System stores new values for the selected promotion in the database
Alternative Flows:	none	
Exceptions:	Promotion is locked	
Priority:	high	
Special Requirements:		
Assumptions:	Promotion exists	
Notes and Issues:		

Delete Existing Promotion

Use Case ID:	19		
Use Case Name:	Delete Existing Promotion		
Created By:	Logan Hutcheson	Last Updated By:	Logan Hutcheson
Date Created:	9/16	Date Last Updated:	9/16

Actors:	Admin	
Brief Description:	The admin can delete an existing promotion from the system.	
Preconditions:	The admin is logged into the system.	
Postconditions:	The selected promotion is deleted from the database.	
Related use cases	Manage Promotions	
Normal Flow:	Actor	System
	1. Admin selects a promotion from a list of existing promotions. 2. Admin provides confirmation to delete promotion.	1.1 System prompts admin for confirmation to delete selected promotion. 2.1 System removes the selected promotion from the database.
Alternative Flows:	none	
Exceptions:	Promotion is locked	
Priority:		
Special Requirements:		
Assumptions:	Promotion exists	
Notes and Issues:		

Manage Users

Use Case ID:	20		
Use Case Name:	Manage Users		
Created By:	Stefan Tobler	Last Updated By:	Edward Cho
Date Created:	9/16	Date Last Updated:	9/18

Actors:	Admin	
Brief Description:	Admin can promote, demote(admin), or suspend users	
Preconditions:	Admin is logged in	
Postconditions:	Status changed of selected user	
Related use cases	Promote User, Demote Admin, Suspend User	
Normal Flow:	Actor	System
	1. Admin can see all the users and other admins 2. Admin can check to promote user or suspend user 3. Admin can choose to reinstate user account	2.1 System handles selected operation and saves new user status in database
Alternative Flows:	Admin chooses to demote user	
Exceptions:		
Priority:		
Special Requirements:		
Assumptions:	User is an admin	
Notes and Issues:		

Edit Prices

Use Case ID:	21		
Use Case Name:	Edit Prices		
Created By:	Stefan Tobler	Last Updated By:	Stefan Tobler
Date Created:	9/16	Date Last Updated:	9/16

Actors:	Admin	
Brief Description:	Admin can edit the price for a book	
Preconditions:	Book has price X	
Postconditions:	Book's price is updated in everyone's cart and on it's page	
Related use cases	Edit Book	
Normal Flow:	Actor	System
	1. Admin opens the edit book view 2. Admin updates the price 3. Admin clicks "Save"	3.1 System updates the price of the book 3.2 Notification is sent out to those who have the book in their cart that the price has been changed.
Alternative Flows:		
Exceptions:	Entered price is negative	
Priority:		
Special Requirements:		
Assumptions:	Book is in stock	
Notes and Issues:		

Display Order Total

Use Case ID:	22		
Use Case Name:	Display Order Total		
Created By:	Steven Xu	Last Updated By:	Steven Xu
Date Created:	9/16	Date Last Updated:	9/16

Actors:	System	
Brief Description:	System outputs total cost of the order in dollars	
Preconditions:	The order consists of at least one item	
Postconditions:	The total price of all items in the order is displayed	
Related use cases	Checkout, Calculate order total payment	
Normal Flow:	Actor	System
	1. User goes to cart	1.1 System displays order total
Alternative Flows:		
Exceptions:		
Priority:		
Special Requirements:		
Assumptions:	The user is logged in	
Notes and Issues:		

Enter Payment Info

Use Case ID:	23		
Use Case Name:	Enter Payment Info		
Created By:	Steven Xu	Last Updated By:	Steven Xu
Date Created:	9/16	Date Last Updated:	9/16

Actors:	Web user, Logged-in user	
Brief Description:	User fills in payment info to be used to make orders	
Preconditions:	User has created an account or in the process of doing so	
Postconditions:	Payment info has been verified and saved in the system	
Related use cases	Create Account, Checkout	
Normal Flow:	Actor	System
	<ol style="list-style-type: none"> 1. User enters payment info 2. User agrees to save payment info 	<ol style="list-style-type: none"> 1.1 System verifies payment info is valid 1.2 System prompts the user to save payment info 2.1 User's payment info stored under the account in the database
Alternative Flows:		
Exceptions:		
Priority:		
Special Requirements:		
Assumptions:		
Notes and Issues:	System may invoke while a user is creating an account	

Calculate Order Total Payment

Use Case ID:	24		
Use Case Name:	Calculate Order Total Payment		
Created By:	Steven Xu	Last Updated By:	Steven Xu
Date Created:	9/16	Date Last Updated:	9/16

Actors:	System	
Brief Description:	System adds up price of all item(s) in cart	
Preconditions:	At least one item is in cart	
Postconditions:	Added price of all item(s)	
Related use cases	Display Order Total, Checkout	
Normal Flow:	Actor	System
	1. User goes to cart	1.1 System adds order total
Alternative Flows:		
Exceptions:		
Priority:		
Special Requirements:		
Assumptions:	The user is logged in	
Notes and Issues:		

Send Order Confirmation Email

Use Case ID:	25		
Use Case Name:	Send Order Confirmation Email		
Created By:	Steven Xu	Last Updated By:	Steven Xu
Date Created:	9/16	Date Last Updated:	9/16

Actors:	System	
Brief Description:	System sends a confirmation email once the order has been placed	
Preconditions:	The user has placed an order	
Postconditions:	An email confirmation about the order is sent to the user who made it	
Related use cases	Checkout	
Normal Flow:	Actor	System
	<ol style="list-style-type: none"> 1. User successfully checkout an order 2. User receives confirmation email 	<ol style="list-style-type: none"> 1.1 System sends out an email containing all the necessary info to the specific user
Alternative Flows:		
Exceptions:		
Priority:		
Special Requirements:		
Assumptions:	User has the correct email address	
Notes and Issues:		

Send Account Verification Email

Use Case ID:	26		
Use Case Name:	Send Account Verification Email		
Created By:	Steven Xu	Last Updated By:	Edward Cho
Date Created:	9/16	Date Last Updated:	9/17

Actors:	System	
Brief Description:	System sends a verification email during account creation	
Preconditions:	User has provided an email address; User status is Inactive	
Postconditions:	An email with verifications sent to the given email address; User status is Active	
Related use cases	Create Account	
Normal Flow:	Actor	System
	<ol style="list-style-type: none"> 1. User attempts to register to the system with a given email address 2. User receives the verification code and enters it into system 	<ol style="list-style-type: none"> 1.1. System sends an email with verification code to the specific email address 2.1. System assigns user a unique account ID and sets user status to Active; Saves new info in database
Alternative Flows:		
Exceptions:		
Priority:		
Special Requirements:		
Assumptions:	User has provided a correct email address	
Notes and Issues:		

Reorder Book

Use Case ID:	27		
Use Case Name:	Reorder Book		
Created By:	Stefan Tobler	Last Updated By:	Stefan Tobler
Date Created:	9/17	Date Last Updated:	9/17

Actors:	Logged-in User	
Brief Description:	A logged-in user should be able to reorder books he or she has ordered in the past.	
Preconditions:	User has ordered books in the past	
Postconditions:	New book order has been placed	
Related use cases	Checkout, View Order History, Add book to cart	
Normal Flow:	Actor	System
	1. User goes to his/her order history 2. User clicks on the order details of an order he/she has completed in the past. 3. User has the option to add the book to cart again	3.1 System adds book to cart where user can then checkout with the book
Alternative Flows:		
Exceptions:	Book is not in stock	
Priority:		
Special Requirements:		
Assumptions:	Book is in stock and user has order history	
Notes and Issues:		

Promote User

Use Case ID:	28		
Use Case Name:	Promote User		
Created By:	Stefan Tobler	Last Updated By:	Stefan Tobler
Date Created:	9/17	Date Last Updated:	9/17

Actors:	Admin	
Brief Description:	An admin should be able to promote a user to an admin.	
Preconditions:	Desired user has an active account	
Postconditions:	User is now an Admin	
Related use cases	Manage Users	
Normal Flow:	Actor	System
	1. Admin navigates to users page 2. Admin selects the user he/she wants to promote 3. Admin selects and confirms user promotion	3.1 User's status is changed in the system to reflect that of an administrator
Alternative Flows:		
Exceptions:	User is inactive/suspended, account is already an admin	
Priority:		
Special Requirements:		
Assumptions:	User is active	
Notes and Issues:	Consider two forms of confirmation for admin upgrade, (i.e. click button then type the user's name in a box)	

Demote User

Use Case ID:	29		
Use Case Name:	Demote User		
Created By:	Stefan Tobler	Last Updated By:	Stefan Tobler
Date Created:	9/17	Date Last Updated:	9/17

Actors:	Admin	
Brief Description:	An admin should be able to demote other admins to normal users.	
Preconditions:	Desired account is an admin	
Postconditions:	Admin is now a user	
Related use cases	Manage Users	
Normal Flow:	Actor	System
	1. Admin should navigate to the user page 2. Admin selects another admin to demote 3. Admin confirms demotion	3.1 User's status is changed in the DB to reflect that he/she is now a normal user
Alternative Flows:		
Exceptions:	Selected user is not an admin	
Priority:		
Special Requirements:		
Assumptions:	Selected user is an admin	
Notes and Issues:	Consider two forms of confirmation for admin demotion, (i.e. click button then type the user's name in a box)	

Suspend User

Use Case ID:	30		
Use Case Name:	Suspend User		
Created By:	Stefan Tobler	Last Updated By:	Stefan Tobler
Date Created:	9/17	Date Last Updated:	9/17

Actors:	Admin	
Brief Description:	Admin should be able to suspend regular users	
Preconditions:	Selected user is not suspended and active	
Postconditions:	Selected user is now suspended	
Related use cases	Manage Users	
Normal Flow:	Actor	System
	1. Admin navigates to user page 2. Admin selects user to suspend 3. Admin confirms user suspension	3.1 System updates that user to “suspended” status
Alternative Flows:	If the selected user is an admin, that admin should be demoted then suspended.	
Exceptions:	User is already suspended, user is an admin, user account is not active	
Priority:		
Special Requirements:		
Assumptions:	User is not an admin and has an active account	
Notes and Issues:	Consider two forms of confirmation for user suspension, (i.e. click button then type the user’s name in a box)	

Enter Personal Info

Use Case ID:	31		
Use Case Name:	Enter Personal Info		
Created By:	Edward Cho	Last Updated By:	Edward Cho
Date Created:	9/16	Date Last Updated:	9/18

Actors:	Web user, Logged-in user	
Brief Description:	The user enters their personal information, which is stored in the system	
Preconditions:	User has created an account or in the process of doing so	
Postconditions:	Personal info has been verified and stored in the system	
Related use cases		
Normal Flow:	Actor	System
	<ol style="list-style-type: none"> 1. User enters name, phone number, email, password 2. User agrees to save payment info 	<ol style="list-style-type: none"> 1.1. System verifies that phone number, email, and password are valid 1.2. System prompts the user to save personal info 2.1 User's personal info stored under the account in the database
Alternative Flows:	none	
Exceptions:	Email address already exists in system	
Priority:		
Special Requirements:	Passwords should be encrypted when stored in database	
Assumptions:		
Notes and Issues:	System may invoke while a user is creating an account	

Enter Address Info

Use Case ID:	32		
Use Case Name:	Enter Address Info		
Created By:	Edward Cho	Last Updated By:	Edward Cho
Date Created:	9/17	Date Last Updated:	9/18

Actors:	Web user, Logged-in user	
Brief Description:	The user enters their shipping address information, which will be saved in the system and used for orders	
Preconditions:	User has created an account or in the process of doing so	
Postconditions:	Address information has been verified and stored in system	
Related use cases		
Normal Flow:	Actor	System
	<ol style="list-style-type: none"> 1. User enters their street, city, state, and zip code 2. User agrees to save address info 	<ol style="list-style-type: none"> 1.1 System prompts the user to save address info 2.1 User's address info stored under the account in the database
Alternative Flows:	none	
Exceptions:		
Priority:		
Special Requirements:		
Assumptions:	Address information is valid	
Notes and Issues:	System may invoke while a user is creating an account	

Unsubscribe from Promotion

Use Case ID:	33		
Use Case Name:	Unsubscribe from Promotion		
Created By:	David DiGioia	Last Updated By:	David DiGioia
Date Created:	9/17	Date Last Updated:	9/17

Actors:	Registered User	
Brief Description:	The user unsubscribes from promotional emails, so they will no longer receive promotion codes and marketing emails.	
Preconditions:	User has a registered account and is subscribed to promotions.	
Postconditions:	User will no longer receive marketing emails.	
Related use cases	Add Promotion, Add New Promotion	
Normal Flow:	Actor	System
	1. User clicks 'unsubscribe' button in promotional email.	1.1. System takes user to confirmation webpage, indicating success.
Alternative Flows:	none	
Exceptions:	If the webpage is unavailable, show an error message. If the user is already unsubscribed show an error message.	
Priority:		
Special Requirements:		
Assumptions:	User is already subscribed.	
Notes and Issues:		

Manage Shopping Cart

Use Case ID:	34		
Use Case Name:	Manage Shopping Cart		
Created By:	David DiGioia	Last Updated By:	David DiGioia
Date Created:	9/17	Date Last Updated:	9/17

Actors:	Web user, Logged-in user	
Brief Description:	The user may add or remove items from the shopping cart and view the items currently in the cart.	
Preconditions:	None	
Postconditions:	The shopping cart will remain in the updated state.	
Related use cases		
Normal Flow:	Actor	System
	1. The user clicks the 'remove' button to remove an item from the cart. 2. The user can click 'add to cart' to add a book to the cart.	1.1. The system removes the item from the user's cart. 2.1. The system invokes 'Add to Cart' use case.
Alternative Flows:	The user may click a 'remove all' button to remove all items from the cart.	
Exceptions:	The user tries to add an item to the cart that is no longer sold or is out of stock.	
Priority:		
Special Requirements:		
Assumptions:		
Notes and Issues:		

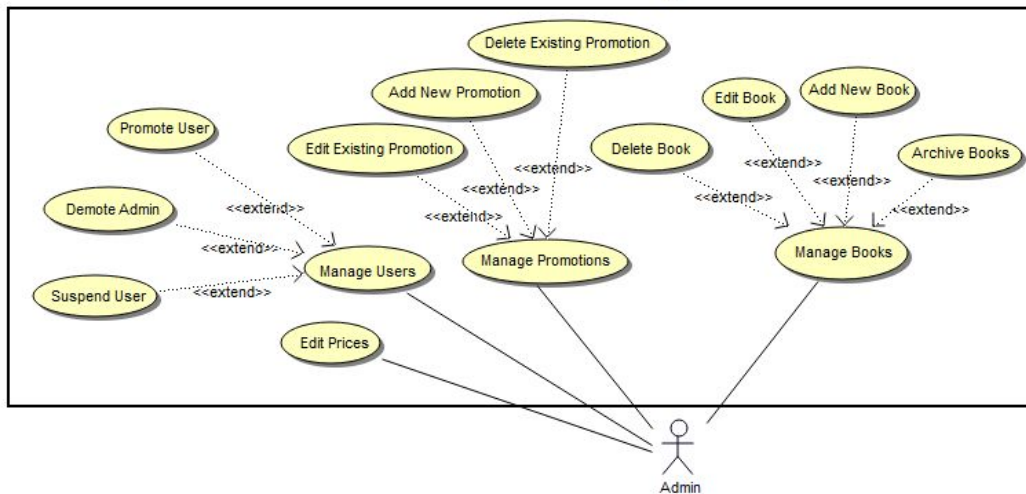
Archive Books

Use Case ID:	35		
Use Case Name:	Archive Books		
Created By:	David DiGioia	Last Updated By:	David DiGioia
Date Created:	9/17	Date Last Updated:	9/17

Actors:	Admin	
Brief Description:	The admin may archive books to mark them as not being actively sold, while still allowing people to finish checking out with them.	
Preconditions:	The book is currently not archived	
Postconditions:	The book is archived	
Related use cases		
Normal Flow:	Actor	System
	1. Admin selects 'archive' button by book title. 2. Admin confirms that they want to archive the book.	1.1. System prompts the admin to confirm permission to archive. 2.1. The book is archived.
Alternative Flows:	The admin may deny confirmation when prompted which will terminate the request.	
Exceptions:	The admin tries to archive a book which is already archived.	
Priority:		
Special Requirements:		
Assumptions:		
Notes and Issues:		

Use Case Diagrams

Admin Use Case Diagram



User and System Use Case Diagram

