

Book Exchange Django Application – Development Documentation

Overview

This Book Exchange web application is designed to facilitate a community-driven platform where users can share, discover, and manage books. The app targets students and general users interested in exchanging books in an organized, interactive environment.

App Purpose and Features

Core Functions

- Book Listings and Details: Users can browse posted books, view detailed information including price, images, posting user, and website links.
- User Registration & Authentication: New users can register accounts; existing users can log in to interact with features.
- Book Management: Users can post books they want to share and manage their posted list.
- Search Functionality: Users can search for books by name, streamlining discovery.
- User-centric Lists: Users have personal views showing their posted books, purchased books, and favorites.

Advanced Interactive Features

- Ratings: Users can rate each book from 1 to 5 stars. Ratings display anonymously except for the current user.
- Favorites: Users can add and remove books to a favorites list, enabling easy access to preferred books.
- Comments: Users can add comments on book details, and delete only their own comments, fostering community discussion.

- Shopping Cart & Checkout: Users can add books to a cart and check out, marking items as purchased.
 - Access Control: Certain views like "My Books" and "Checkout" require login; unauthorized access shows a custom "You need to login" message instead of redirect.
-

Student Required Features

1. About Us Page:
A dedicated informative page describing the app's purpose and development context.
 2. Book Search:
The search bar allows users to query books by name with immediate results.
 3. Navigation Reformat:
Responsive, user-friendly navigation bar styled with Bootstrap and CSS for fluid interaction.
-

Chosen Additional Features

(4) Message Box (Comment System):

Users can add comments to books and manage their own comments, improving user interaction and feedback.

(5) Shopping Cart:

Shopping cart functionality for adding multiple books and managing purchases easily.

(6) Ratings:

Star-rating system on books to provide qualitative user feedback integrated directly on book detail pages.

Key Challenges and Solutions

- Login Redirection 404 Issue:
Problem: Users accessing login-protected pages were redirected to a non-existent `/accounts/login/` causing 404 errors.
Solution: Removed Django's default `@login_required` redirect behavior. Added

manual `request.user.is_authenticated` checks in views, rendering a custom login-required message template to inform users without redirects.

- Anonymous Ratings Display:
Needed to show ratings for transparency but maintain privacy.
Solution: In templates, display "You" for current user's rating and "Anonymous" for others.
 - Favorites Toggle:
Allowing seamless add/remove without confusion.
Solution: Implemented toggle button with conditional template rendering to switch between "Add to Favorites" and "Remove from Favorites."
 - Comment Ownership & Delete Control:
Ensuring users can only delete their own comments.
Solution: Server-side checks in delete view with `HttpResponseForbidden` response if authorization fails.
-

Summary of Implemented Requirements

Requirement	Status	Notes
AboutUs	Completed	Informational page with CSULA logo image
Search a book	Completed	Case-insensitive search by name
Reformat Nav	Completed	Bootstrap/CSS responsive navigation bar
Message Box (Comments)	Completed	Add/delete comment on books with ownership control

Shopping Cart	Completed	Add to cart, view cart, check out
Ratings	Completed	Star rating system with anonymous display
Favorite List	Completed	User-specific favorites with toggle button

Usage Flow for Users

- Visit the homepage to browse or search books.
- Register or log in to access full features.
- Post books to share with others.
- Add books to the shopping cart and checkout.
- Rate and comment on books.
- Favorite/unfavorite books for quick access.
- Manage personal book lists (posted, purchased, favorites).
- Access control ensures only logged-in users perform certain actions, otherwise, they see friendly login reminder pages.