

1. Introduction

Provide an overview of the application:

- **Name:** Bookstore Inventory Management System
- **Purpose:** Manage inventory of books, authors, and orders for a bookstore.

2. Technologies Used

Outline the technologies and tools utilized in the project:

- **Framework:** Django
- **Database:** SQLite (for development), PostgreSQL (for production)
- **Frontend:** HTML, CSS, JavaScript, Bootstrap
- **Payment Integration:** Stripe (for payment processing)

3. Application Structure

Describe the structure of your Django application:

- **Project:** bookstore_inventory
 - **Apps:**
 - cheussopkeng: Main app for bookstore functionality.
 - order: App for managing user orders and payments.
 - author_book: App for managing authors and their books.
 - **Templates:** Organized into folders (panels, book_manage, author_book, order) for different sections of the application.
 - **Static files:** CSS, JavaScript, and image files stored under static directory.

4. Features

Detail the key features and functionalities of the application:

- **Admin Panel:**
 - CRUD operations for books and authors.
 - Customized admin interface.
- **User Panel:**
 - User authentication (login, register, logout).
 - Ability to view books and place orders.
 - Integrated payment gateway (Stripe) for order processing.
- **Book Management:**
 - Create, read, update, and delete books.
 - View book details including title, author, ISBN, price, and stock quantity.
- **Author Management:**
 - Create, read, update, and delete authors.
 - View author details including name and biography.

- **Order Processing:**
 - Users can add books to cart and place orders.
 - Integration with Stripe for secure payment processing.

5. Implementation Details

Discuss specific implementation aspects:

- **Models:** Book, Author, Order, OrderItem with relationships defined using Django ORM.
- **Views:** Views implemented for rendering HTML templates and handling business logic.
- **Forms:** BookForm and AuthorForm for data validation and user input.
- **Authentication:** Custom user authentication system using Django's built-in User model.
- **Templates:** HTML templates styled with Bootstrap, CSS, and JavaScript for responsive design and interactivity.
- **Testing:** test.py script implemented to test all endpoints ensuring functionality and correctness.
- **Deployment:** Preparation for deployment on a production server, including configuration for PostgreSQL database, static file management, and security measures.

6. Conclusion

Summarize the project's achievements, challenges overcome, and future enhancements:

- **Achievements:** Successfully implemented a comprehensive bookstore inventory management system with CRUD operations, user authentication, and payment integration.
- **Challenges:** Addressed challenges in data modeling, frontend integration, and ensuring secure payment processing.
- **Future Enhancements:** Plan for adding features such as advanced search, reviews, user profiles, and enhanced reporting capabilities.