

Advanced Task Management System

Develop a simplified bookstore API that allows users to browse books, add books to a cart, and place orders. The system should support user authentication and authorization.

Objective

Requirements

1. **User Authentication and Authorization:**
 - Users should be able to register and log in.
 - Only authenticated users can add books to their cart and place orders.
 2. **Books:**
 - List all books.
 - View book details.
 3. **Cart:**
 - Add books to the cart.
 - Remove books from the cart.
 - View cart details.
 4. **Orders:**
 - Place an order from the cart.
 - View order history.
 -
- **Books:**
 - GET `/api/books` - List all books.
 - GET `/api/books/:id` - Get book details.
 - **Cart:**
 - GET `/api/cart` - Get cart details for the authenticated user.
 - POST `/api/cart` - Add a book to the cart.
 - DELETE `/api/cart/:itemId` - Remove a book from the cart.
 - **Orders:**
 - POST `/api/orders` - Place an order.
- API Endpoints:**
- **Authentication:**
 - POST `/api/auth/register` - Register a new user.
 - POST `/api/auth/login` - Log in a user.
 - GET `/api/orders` - View order history for the authenticated user.
 - GET `/api/orders/:id` - Get order details.

Deliverables

1. **Database Design (in image format):**
 - Database Design tables for Users, Books, Cart, Cart_Items, Orders, Order_Items, etc.
2. **Database .sql File:**
 - SQL script to create the necessary database schema.
3. **Code with Installation Guide (in GitHub repository):**
 - ExpressJS code with detailed README for setup and installation.
 - Proper folder structure and code organization.
 - Use environment variables for sensitive data.
4. **Postman Collection:**
 - Postman collection for all the API endpoints.

Notes

- Ensure proper validation and error handling for all endpoints.
- Use middleware for authentication and authorization.
- Follow best practices for code structure and organization.
- Include comments and documentation within the code for clarity.
- Use modern JavaScript (ES6+) features and syntax.