**Bookstore Project Documentation**

**1. Project Overview**

**1.1 Purpose**

The Bookstore Project is designed to manage the sales of books, track inventory, and provide a smooth shopping experience for customers. It includes features for browsing books, adding items to the cart, making purchases, and managing stock levels.

**1.2 Key Features**

* **User Authentication**: Register, login, and manage user profiles.
* **Book Catalog**: Browse books by category, author, or price.
* **Shopping Cart**: Add, update, and remove books from the cart.
* **Order Management**: Place orders, track order history, and view invoices.
* **Admin Features**: Add, edit, or remove books from the catalog, manage user roles, and view sales reports.

**2. System Requirements**

**2.1 Software Requirements**

* **Operating System**: Windows/Linux/MacOS
* **Languages**: Python, JavaScript (for backend and frontend)
* **Database**: MySQL/PostgreSQL/SQLite
* **Frameworks/Tools**:
  + Backend: Django/Flask (Python)
  + Frontend: React.js/Angular/Vue.js
  + Payment Gateway: Stripe/PayPal

**2.2 Hardware Requirements**

* Processor: Intel i3 or higher
* RAM: 4GB or higher
* Disk Space: 500MB minimum

**3. Project Setup**

**3.1 Clone the Repository**

git clone https://github.com/Vetribala2/book-store.git

**3.2 Installation**

**Backend (Django Example)**

1. Install dependencies:
2. pip install -r requirements.txt
3. Setup the database:
4. python manage.py migrate

**Frontend (React Example)**

1. Navigate to the frontend directory:
2. cd frontend
3. Install dependencies:
4. npm install
5. Start the frontend server:
6. npm start

**Running the Application**

After setting up the backend and frontend, you can access the application at:

* Backend: http://localhost:8000
* Frontend: http://localhost:3000

**4. Database Design**

**4.1 Tables**

* **Users**
  + id (PK)
  + username
  + email
  + password\_hash
  + role (admin/user)
* **Books**
  + id (PK)
  + title
  + author
  + category
  + price
  + stock\_quantity
  + description
* **Orders**
  + id (PK)
  + user\_id (FK)
  + total\_price
  + order\_status (pending/complete)
  + created\_at
  + updated\_at
* **Order\_Items**
  + id (PK)
  + order\_id (FK)
  + book\_id (FK)
  + quantity
  + price

**5. API Documentation**

**5.1 Authentication API**

* **POST /api/auth/register**: Register a new user.
  + Request Body: {"username": "john\_doe", "email": "john@example.com", "password": "password123"}
  + Response: {"message": "User registered successfully"}
* **POST /api/auth/login**: Login to the system.
  + Request Body: {"email": "john@example.com", "password": "password123"}
  + Response: {"token": "jwt\_token\_here"}

**5.2 Book Catalog API**

* **GET /api/books**: Fetch all books.
  + Response:
* [
* {
* "id": 1,
* "title": "Book Title",
* "author": "Author Name",
* "price": 19.99,
* "stock\_quantity": 10
* }
* ]
* **POST /api/books**: Add a new book (Admin only).
  + Request Body: {"title": "New Book", "author": "Author", "price": 25.99, "category": "Fiction", "stock\_quantity": 100}
  + Response: {"message": "Book added successfully"}

**6. Admin Features**

**6.1 Managing Books**

Admins can add, update, or remove books from the catalog.

**6.2 Managing Users**

Admins can view all users, update roles, and manage their information.

**7. Testing**

**7.1 Unit Tests**

* Backend tests can be run using:
* python manage.py test

**7.2 End-to-End Tests**

* For frontend, use tools like Jest or Cypress to test user interaction and workflows.

**8. Deployment**

**8.1 Backend Deployment**

* Deploy using platforms like Heroku, AWS, or DigitalOcean.

**8.2 Frontend Deployment**

* Deploy the frontend on services like Netlify or Vercel.

**8.3 Continuous Integration**

* Use tools like GitHub Actions or CircleCI for automated testing and deployment pipelines.

This is a high-level structure for the documentation of a typical bookstore project. Depending on your specific project, you may need to adjust it to suit your needs, particularly regarding the tech stack or feature specifics.

Team members

B.VETRI (B.TECH(IT))

N.VIGNESH(B.TECH(IT))

P.PREMLATHA(B.TECH(IT))

R.SAJITHA(B.TECH(IT))