Online Bookstore — Requirement Specification Document

**Project name:** Online Bookstore Application

**Purpose:** Build a simple, production-like online bookstore where users can search for books, view details, add items to a cart, and checkout. The project is structured into a 10‑day sprint plan covering requirements, design, implementation, testing, and deployment.

**Primary goals:**

* Provide fast, accurate search and book browsing.
* Allow authenticated users to add books to cart and complete purchases.
* Maintain order records and basic recommendations.

**In scope:**

* User registration & login (basic session- or token-based auth).
* Book catalog stored in MySQL (primary), recommendations stored in DynamoDB (secondary).
* Search by title/author, view book details, add/remove cart items, checkout (payment stub or integration), order persistence.
* Browsing history (recently viewed) and simple recommendations retrieval from DynamoDB.
* Unit & BDD tests for search and checkout flows.
* CI/CD pipeline to build, test, package and deploy JAR to a Linux VM.

**Functional Requirements**

* User can search books by title and/or author
* User can view book details (title, author, price, ISBN, description)
* User can add/remove books to/from cart
* User can checkout; order record inserted in DB
* Admin or seed process can CRUD books
* Browsing history (recently viewed) displayed
* Recommendations fetched from DynamoDB (top-5)
* BDD tests for search scenario
* CI pipeline builds + tests + deploys JAR to Linux VM

**Design**

**Key Classes (OOAD)**

* Book {id, title, author, isbn, price, description, stock}
* User {id, username, email, passwordHash, createdAt}
* Cart {userId, Map<BookId, quantity>} (in-memory session or persisted)
* Order {id, userId, List, total, status, createdAt}
* OrderItem {bookId, quantity, price}

**ER Diagram (high-level tables)**

* **books**(id PK, title, author, isbn, price, description, stock, created\_at)
* **users**(id PK, username, email, password\_hash, created\_at)
* **orders**(id PK, user\_id FK, total\_amount, status, created\_at)
* **order\_items**(id PK, order\_id FK, book\_id FK, quantity, unit\_price)
* **cart\_items**(id PK, user\_id FK, book\_id FK, quantity) — optional if persisting cart