Model : Kitaab Duniya(website)

Submitted by: Ayush Tiwari (20232751)

Hardik Garg (20232771)

Submitted to : Dr. Neha

SRS(Software Requirement Specification)

1. Introduction

1.1 Purpose

The purpose of **KitaabDuniya** is to create a user-friendly online platform for buying and selling books. It aims to bridge the gap between book sellers (new or second-hand) and buyers by offering a secure, efficient, and convenient system for book discovery, order placement, and delivery.

1.2 Intended Audience

- **Developers –** To understand the system requirements for implementation.
- **Testers –** For validating the functionalities of the platform.
- **Project Managers –** To manage development timelines and team efforts.
- **End Users –** Students, readers, and book enthusiasts.

1.3 Scope

The **KitaabDuniya** application will:

- Allow users to browse, search, and filter books.
- Enable user registration/login with secure authentication.
- Let users buy and sell books with a simple interface.
- Provide order tracking and history.
- Offer an admin dashboard to manage users, inventory, and reports.

1.4 Definitions

- **User:** A person using the platform (buyer/seller/admin).
- **Book Listing:** A digital entry of a book with description, price, and seller info.
- **Order Management:** The system of tracking and managing placed orders.
- **Inventory:** Database of books available on the platform.

1.5 References

- IEEE SRS Documentation Standards
- OWASP Secure Coding Practices
- Amazon and Flipkart UI Patterns (for inspiration)

---

2. Overall Description

2.1 User Interfaces

- **Homepage:** Show featured books, categories, and offers.
- **Login/Register Page:** For secure user access.
- **Dashboard:** For users to manage listings, cart, and orders.
- **Admin Panel:** Manage books, users, categories, and reports.

2.2 System Interfaces

- Payment gateway APIs (e.g., Razorpay, Stripe)
- Email/SMS APIs for notifications
- Database for user, book, and order management

2.3 Constraints, Assumptions, Dependencies

- The platform must be responsive and mobile-friendly.
- Secure payment handling is a must.
- Database backups and secure cloud hosting are required.
- Internet connection is assumed for all operations.

2.4 User Characteristics

- **Buyers:** Students or general readers with basic internet knowledge.
- **Sellers:** Bookstore owners or individuals selling second-hand books.
- **Admin:** Technically skilled user for managing the backend.

---

3. System Features and Requirements

3.1 Functional Requirements

- User Registration/Login with authentication
- Book Listing creation/edit/delete
- Book Search with filters (genre, price, author, etc.)
- Shopping cart and order placement
- Payment integration
- Order history and tracking
- Admin control panel for system management

3.2 Use Cases

Use Case 1: Book Purchase

Actors: User, System
Steps:

1. User logs in
2. Searches and adds books to cart
3. Proceeds to checkout
4. Makes payment
5. Order confirmation is shown

Use Case 2: Book Listing by Seller

Actors: Seller, System
Steps:

1. Seller logs in
2. Fills listing form with book details
3. Submits for approval
4. Admin reviews and publishes the listing

3.3 External Interface Requirements

- RESTful APIs for frontend-backend communication
- Payment APIs for transactions
- SMTP or SMS gateways for notifications

3.4 Logical Database Requirements

- **Users Table:** User info, roles, and credentials
- **Books Table:** Book metadata and availability
- **Orders Table:** Order details and status
- **Reviews Table:** User feedback and ratings

3.5 Non-functional Requirements

- **Performance:** Page loads under 3 seconds
- **Scalability:** Able to support 10k+ users
- **Security:** Role-based access, password encryption
- **Availability:** 99.9% uptime

---

4. Project Management (Function Points Estimation)

Function Type	Count	Complexity	Function Points
User Registration/Login	1	4	4
Book Listings	2	4	8
Book Search/Filter	1	5	5
Shopping Cart	1	4	4
Payment Gateway Integration	1	5	5
Order Management	1	4	4
Admin Panel	1	5	5
Notifications (Email/SMS)	1	3	3
Total Function Points (FP)			**38 FP**

Effort Estimation:
38 FP × 2.5 (Productivity Factor) = 95 Person-Hours

---

5. Schedule Timeline Chart

Phase	Duration
Requirement Gathering	1 week
Design (UI + DB)	1-2 weeks
Development	3-4 weeks
Testing (Unit + Integration)	2 weeks
Deployment	1 week
Maintenance	Ongoing (as needed)

---

6. Risk Analysis Table

Risk Category	Probability	Impact
Team Members Unavailable	50–75%	Project delay
Payment Gateway Failure	25–50%	Failed transactions and user complaints
Server Downtime	25–50%	Poor user experience

Compatibility Issues	0–25%	Minor UI bugs
Database Failure	50–75%	Data loss or unprocessed orders

1. Architectural Design

The system architecture of "Kitaabi Duniya" will follow a **three-layer architecture**:

- **Presentation Layer (Frontend):**
  - Developed using React.js, HTML5, CSS3, and JavaScript.
  - Users interact with the website to browse, search, buy, and sell books.
  - Communication with backend through secure RESTful APIs.
- **Application Layer (Backend):**
  - Developed using Node.js (Express.js) or PHP Laravel.
  - Handles business logic: user authentication, book listing management, order processing, notifications, etc.
  - Provides APIs for frontend to perform CRUD (Create, Read, Update, Delete) operations.
- **Data Layer (Database):**
  - Uses MySQL or PostgreSQL.
  - Stores all persistent data like user information, book listings, orders, and reviews.
  - Ensures secure data storage and efficient querying.

---

2. Data Design

The data design includes key tables/entities with their important fields:

2.1 Users Table

Field Name	Data Type	Description
user_id (PK)	INT	Unique ID for each user

name	VARCHAR	Full name of the user
email	VARCHAR	User email (unique)
password	VARCHAR	Encrypted password
phone	VARCHAR	Contact number
address	TEXT	Delivery address
role	ENUM	buyer/seller/admin
created_at	TIMESTAMP	Registration date

2.2 Books Table

Field Name	Data Type	Description
book_id (PK)	INT	Unique ID for each book
title	VARCHAR	Book name
author	VARCHAR	Author of the book
genre	VARCHAR	Book category
price	DECIMAL	Selling price
description	TEXT	Book description
seller_id (FK)	INT	ID of the seller
condition	ENUM	New or Used
image_url	TEXT	Image of the book
status	ENUM	Available/Sold
listed_at	TIMESTAMP	Listing date

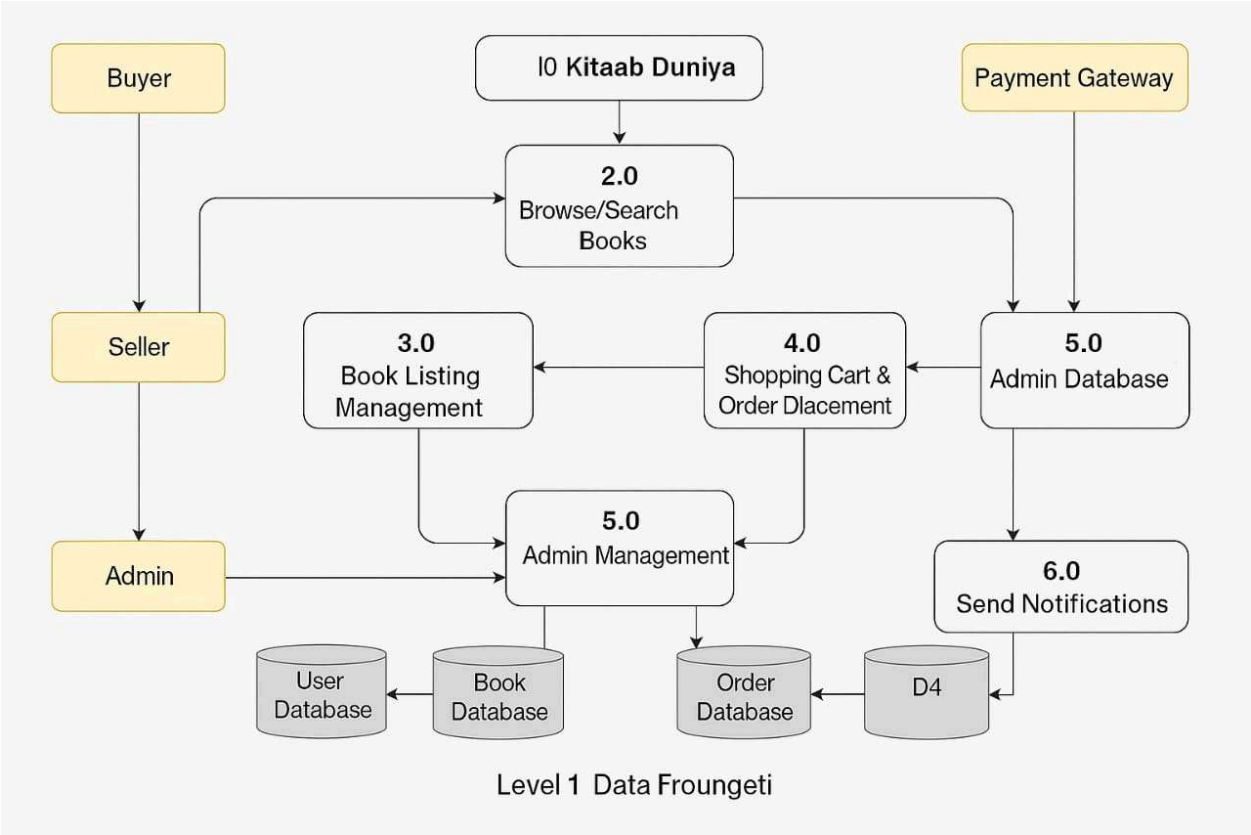
2.3 Orders Table

Field Name	Data Type	Description
order_id (PK)	INT	Unique ID for each order
buyer_id (FK)	INT	ID of the buyer
book_id (FK)	INT	ID of the purchased book
payment_status	ENUM	Paid/Pending/Failed
order_status	ENUM	Placed/Shipped/Delivered/Cancelled
delivery_address	TEXT	Address for delivery
ordered_at	TIMESTAMP	Order placed date

2.4 Reviews Table

Field Name	Data Type	Description
review_id (PK)	INT	Unique ID for each review
book_id (FK)	INT	Book being reviewed
user_id (FK)	INT	User who wrote the review
rating	INT	Rating (1 to 5)
comment	TEXT	Review text
reviewed_at	TIMESTAMP	Date of review

Data Flow Diagram

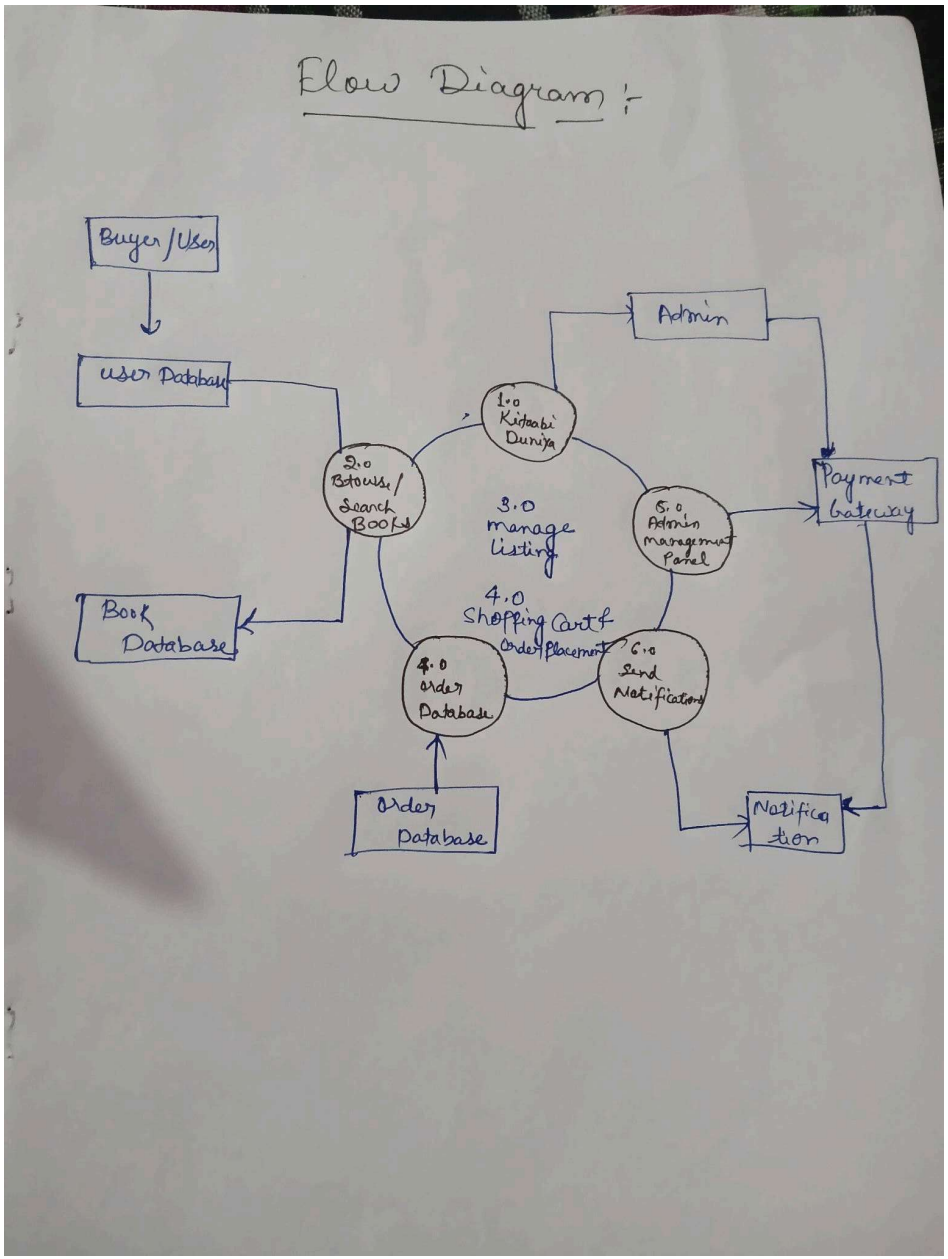


3. Component Level Design

The system will be divided into the following components/modules:

Component	Description
Authentication Module	Handles user login, registration, password encryption, session management.
Book Listing Module	Allows sellers to add, edit, delete books; users to browse and search books.

Shopping Cart Module	Enables buyers to add books to cart, manage cart items, and place orders.
Order Management Module	Processes orders, updates order status, handles payment confirmations.
Payment Integration Module	Integrates with payment gateways like Razorpay or Stripe.
Admin Dashboard Module	Admin can manage users, view reports, handle book listings, and oversee platform activities.
Notification Module	Sends email/SMS alerts for order confirmation, shipment updates, password reset etc.
Review & Rating Module	Allows buyers to review and rate purchased books.


Flow Diagram :-
Buyer / User
User Database
Admin
1.0 Kitaabi Duniya
2.0 Browse / Search Books
3.0 manage listing
5.0 Admin management Panel
Payment Gateway
Book Database
4.0 Shopping Cart & Order Placement
4.0 order Database
6.0 Send Notifications
Order Database
Notification