**therightbookstoreindia.com**

****

**FUNCTIONAL REQUIREMENTS**

**Product Name**: Online Bookstore Application

**Introduction**

An Online Bookstore application is a digital platform where users can browse, purchase, and sometimes even read books online.. The application is built using Java and Spring Boot for the backend, and React for the frontend.

Typically, these applications offer a user-friendly interface that allows customers to search for books by title, author, genre, or topic. They often include features such as personalized recommendations based on past purchases or browsing history, customer reviews and ratings, and the ability to preview excerpts of books before making a purchase.

**Bookstore Application Frontend**

This Document is intended to be used as an introduction to the bookstore default

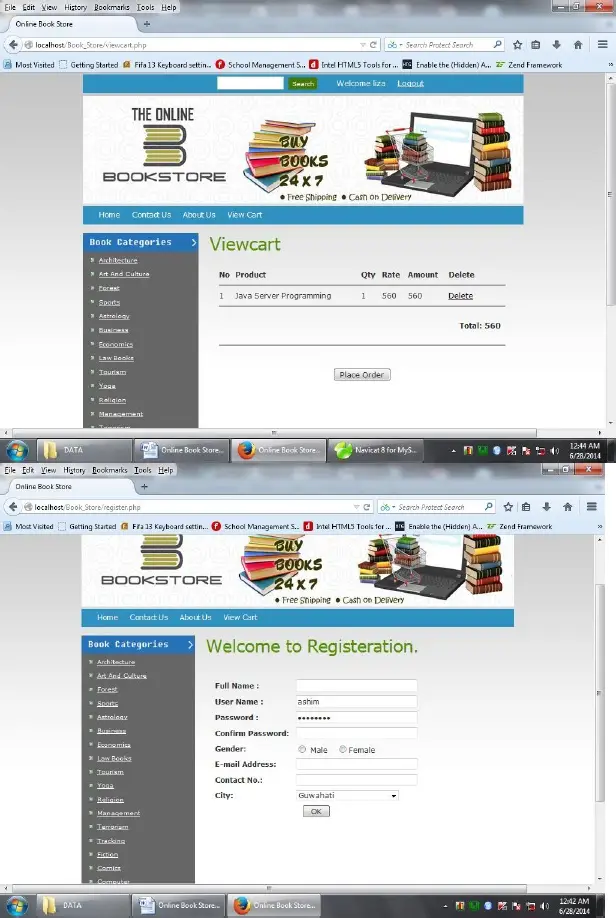
store front. The store front reveals how the customer views and interacts with

the store.

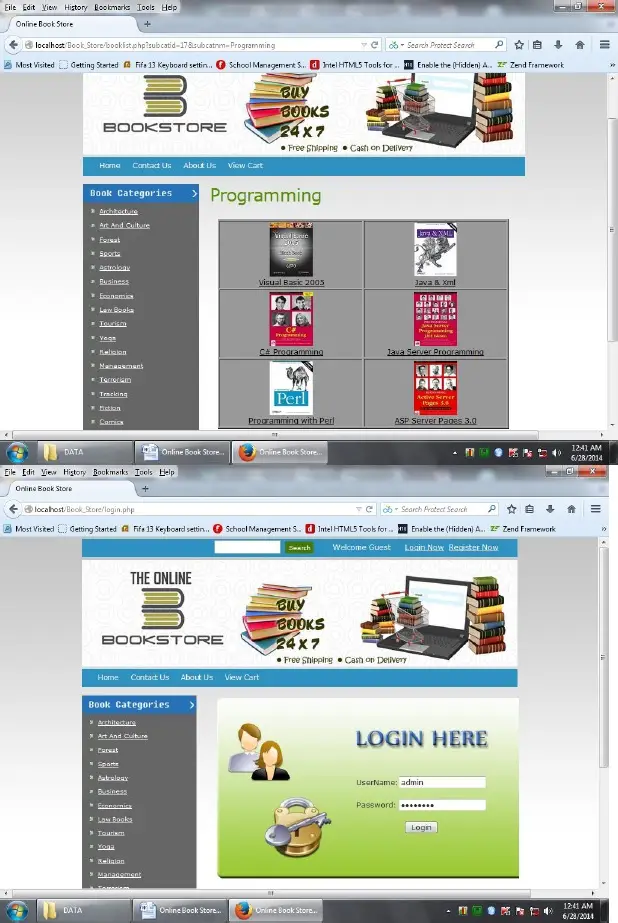
**Index page**

****

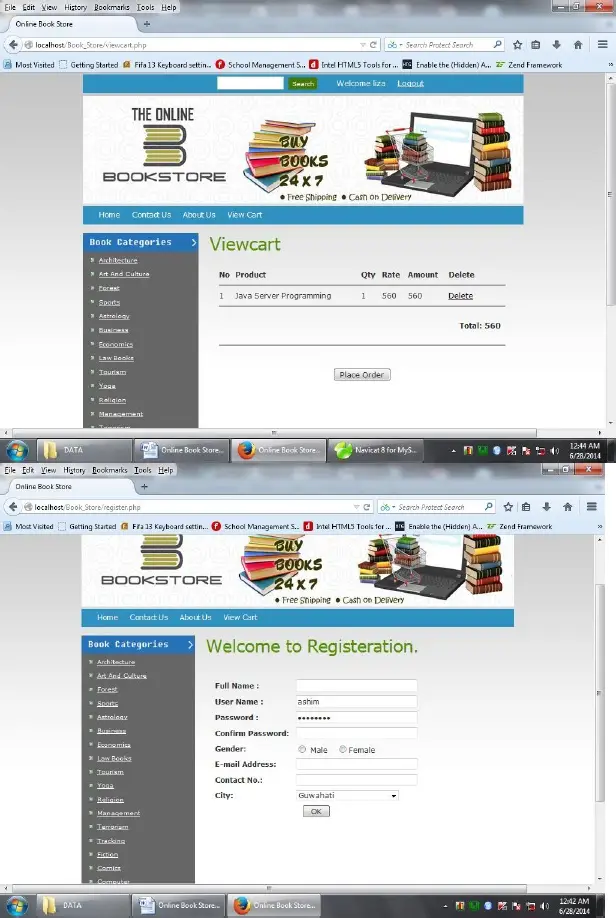
**Registration Page**

****

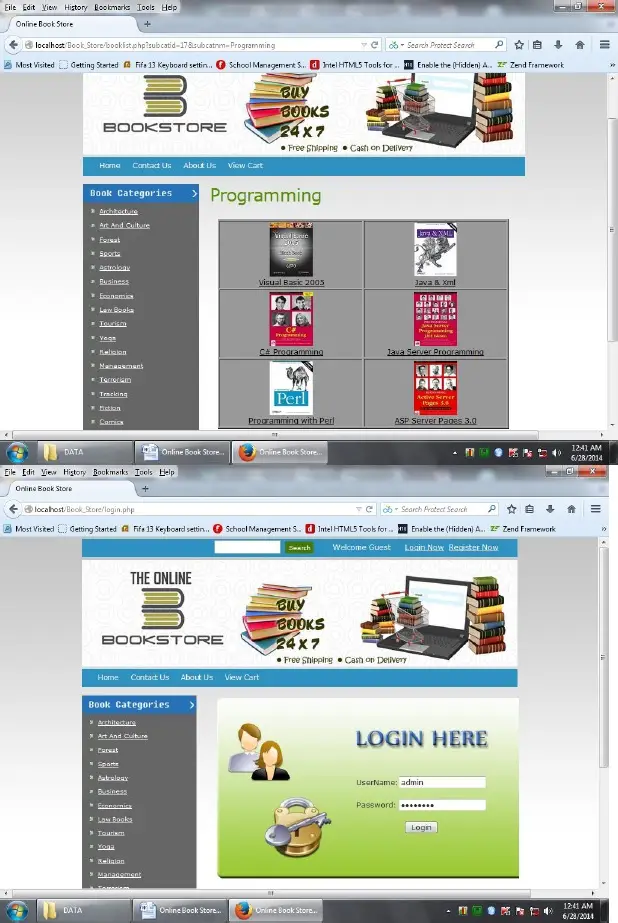
**Login Page**

****

**View Cart Page**

****

**Book List Page**

****

**Application overview**

**The Online Bookstore application has the following features:**

* **User Authentication:**
  + Users can create accounts and log in.
  + Passwords should be securely stored.
  + Users should be able to reset their passwords.
* **Book Catalog:**
  + Users can browse a catalog of available books.
  + Each book should have details like title, author, price, and availability.
  + Users can search for books by title or author.
* **Shopping Cart:**
  + Users can add books to their shopping cart.
  + The cart should display the selected items and the total price.
  + Users can remove items from the cart.
* **Checkout:**
  + Users can proceed to checkout.
  + During checkout, users need to provide shipping information.
  + The system should calculate the total order amount.
* **Order Confirmation:**
  + Users receive an order confirmation email after a successful purchase.
  + Orders are stored in the database for reference.
* **User Interface:**
  + Ensure that the UI is intuitive and responsive.
  + Verify that error messages are displayed appropriately.
  + Test for compatibility across different browsers.