Online Book Store

APROJECTREPORT

Submitted by

SUJITHA G (2303811724322113)

Inpartial fulfillment of requirements for the award of the course

CGB1201–JAVAP ROGRAMMING

in

ARTIFICIAL INTELLIGENCE AND DATA SCIENCE

K.RAMAKRISHNAN COLLEGE OF TECHNOLOGY

(An Autonomous Institution, affiliated to Anna University Chennai and Approved by AICTE, New Delhi)

SAMAYAPURAM–621112

DECEMBER,2024

K.RAMAKRISHNAN COLLEGE OF TECHNOLOGY(AUTONOMOUS)

SAMAYAPURAM–621112

BONAFIDECERTIFICATE

Certified that this project report on **" ONLINE BOOK STORE"** is the bonafide work of **SUJITHA G (ADB23113)** who carried out the project work during the academic year 2024 - 2025 under my supervision.

Signature

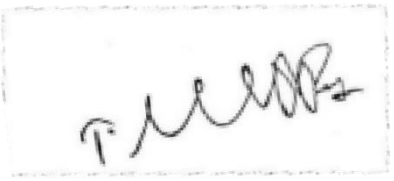
Mrs.S.GEETHA M.E,

SUPERVISOR,

Department of Artificial Intelligence,

K.Ramakrishnan College of Technology,

Samayapuram,Trichy-621112.



Signature

Dr.T.AVUDAIAPPANM.E.,Ph.D.,

HEAD OF THE DEPARTMENT,

Department of Artificial Intelligence,

K.Ramakrishnan College of Technology,

Samayapuram,Trichy-621112.

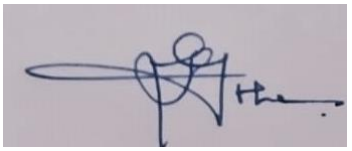
Submitted for the viva-voce examination held on3.12.24

INTERNALEXAMINER

EXTERNALEXAMINER

DECLARATION

I declare that the project report on "**ONLINE BOOK STORE** " is the result of original work done by us and best of our knowledge, similar work has not been submitted to "**ANNA UNIVERSITY CHENNAI**" for the requirement of Degree of **BACHELOROF ENGINEERING**. This project report is submitted on the partial fulfillment of the requirement of the award of the **CGB1201 – JAVA PROGRAMMING.**



Signature

SUJITHA G

Place: Samayapuram

Date: 3/12/2024

ACKNOWLEDGEMENT

It is with great pride that I express our gratitude and indebtedness to our institution, **"K. Ramakrishnan College of Technology (Autonomous)",** for providing us with the opportunity to do this project.

I extend our sincere acknowledgment and appreciation to the esteemed and honorable Chairman, **Dr.K.RAMAKRISHNAN,B.E.,** for having provided the facilities during the course of our study in college.

I would like to express our sincere thanks to our beloved Executive Director, **Dr. S. KUPPUSAMY, MBA, Ph.D.,** for forwarding our project and offering an adequate duration to complete it.

I would like to thank **Dr.N.VASUDEVAN,M.TECH., Ph.D.,**Principal, who gave the opportunity to frame the project to full satisfaction.

I thank **Dr.T.AVUDAIAPPAN, M.E.,Ph.D**., Head of the Department of **ARTIFICIAL INTELLIGENCE AND DATA SCIENCE**, for providing her Encouragement in pursuing this project.

I wish to convey our profound and heartfelt gratitude to our esteemed project guide **Mrs.S.GEETHA M.E**., Department of **ARTIFICIAL INTELLIGENCE AND DATA SCIENCE**, for her incalculable suggestions, creativity, assistance and patience, which motivated us to carry out this project.

I render our sincere thanks to the Course Coordinator and other staff members for providing valuable information during the course.

I wish to express our special thanks to the officials and Lab Technicians of our departments who rendered their help during the period of the work progress.

VISION OF THE INSTITUTION

To serve the society by offering top-notch technical education on par with global standards.

MISSION OF THE INSTITUTION

- Be a centre of excellence for technical education in emerging technologies by exceeding the needs of industry and society.
- Be an institute with world class research facilities.
- Be an institute nurturing talent and enhancing competency of students to transform them as all- round personalities respecting moral and ethical values.

VISION AND MISSION OF THE DEPARTMENT

To excel in education, innovation and research in Artificial Intelligence and Data Science to fulfill industrial demands and societal expectations.

Mission 1: To educate future engineers with solid fundamentals, continually improving teaching methods using modern tools.

Mission2:To collaborate with industry and offer top-notch facilities in a conductive learning environment.

Mission 3: To foster skilled engineers and ethical innovation in AI and Data Science for global recognition and impactful research.

Mission4:To tackle the societal challenge of producing capable professionals by instilling employability skills and human values.

PROGRAM EDUCATIONAL OBJECTIVES(PEOS)

PEO 1: Compete on a global scale for a professional career in Artificial Intelligence and Data Science.

PEO 2: Provide industry-specific solutions for the society with effective communication and ethics.

PEO3: Hone their professional skills through research and lifelong learning initiatives.

PROGRAM OUTCOMES

Engineering students will be able to:

1. **Engineering knowledge:** Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
2. **Problem analysis:** Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.
3. **Design/development of solutions:** Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.
4. **Conduct investigations of complex problems:** Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
5. **Modern tool usage:** Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.
6. **The engineer and society:** Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.
7. **Environment and sustainability:** Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.
8. **Ethics:** Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

9. **Individual and team work:** Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.
10. **Communication:** Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.
11. **Project management and finance:** Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.
12. **Life-long learning:** Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

PROGRAMSPECIFICOUTCOMES(PSOs)

- **PSO 1:** Capable of working on data-related methodologies and providing industry-focussed solutions.
- **PSO2:** Capable of analyzing and providing a solution to a given real-world problem by designing an effective program.

ABSTRACT

This project aims to develop an online bookstore that provides a seamless shopping experience for customers while enabling efficient management of book sales and inventory.

The system allows users to search for and purchase books based on various criteria such as title, author, and category.

Key features include a user-friendly interface, personalized book recommendations using machine learning algorithms, real-time inventory management, and secure payment processing. The project also includes an admin module for managing inventory, user accounts, and generating sales reports.

By leveraging modern technologies, this online bookstore enhances convenience for customers and operational efficiency for store owners, addressing the limitations of traditional brick-and-mortar bookstores.

TABLEOF CONTENTS

CHAPTER 1

INTRODUCTION

INTRODUCTION

The Online Bookstore Management System is a Java-based platform designed to simplify book purchasing and management. It offers users features like book searches, a shopping cart, and secure payment options. Administrators can efficiently manage inventory, user accounts, and orders. This system ensures a seamless and user-friendly experience for both customers and administrators.

OBJECTIVE

The main objective of this project is to develop an online book store that:

Allows users to search for and purchase books based on title, author, and category.

Provides a seamless shopping experience with features like a shopping cart and secure payment options.

Enables administrators to manage book inventory, user accounts, and orders efficiently.

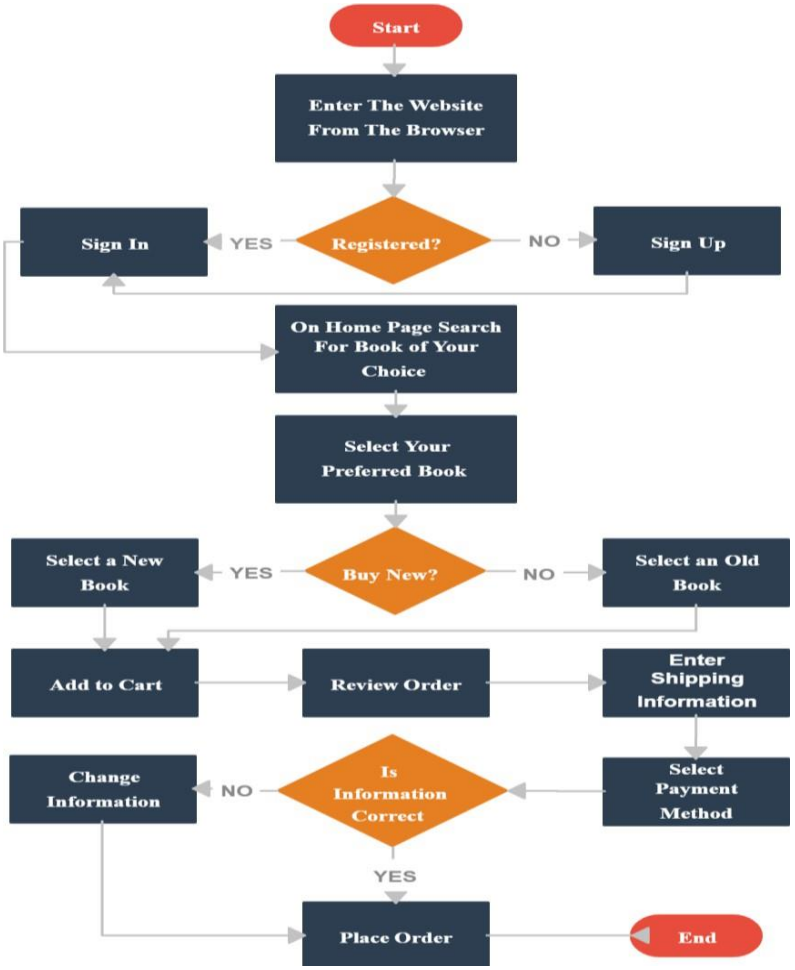
CHAPTER 2

PROJECTMETHODOLOGY

PROPOSEDWORK

The proposed **Online Book Store Management System** is a Java-based platform designed to streamline book purchasing and management. It includes modules for user registration, book browsing, shopping cart functionality, order processing, and admin management. The system aims to provide a user-friendly interface with efficient workflows and robust database integration to enhance the overall experience for users and administrators.

BLOCKDIAGRAM



CHAPTER3

JAVAPROGRAMMINGCONCEPTS

Core Java Concepts

- **Object-Oriented Programming (OOP)**:The system uses OOP principles like encapsulation, abstraction, and modular design to create reusable and maintainable code.
- **Inheritance and Polymorphism**: These concepts allow for flexibility and code reuse, enabling the development of scalable classes and dynamic method behavior.
- **Exception Handling**:Ensures smooth operation by gracefully managing errors and exceptions that may arise during user interactions or systemoperations.

Advanced Java Concepts

- **Java Database Connectivity (JDBC)**: Facilitates secure and efficient data management by connecting the system to the backend database for user, book, and order details.
- **Collections Framework**: Manages dynamic data structures like lists andmaps, enabling efficient handling of book catalogs, shopping carts, and user data.
- **File I/O**: Supports reading and writing data to files for backup and report generation.
- **Multithreading**: Improves system responsiveness by handling multiple processes, such as user requests and order processing, concurrently.

CHAPTER 4

MODULEDESCRIPTION

Module1

User Module

Manages user registration, login, and authentication to ensure secure access to the platform. Users can create new accounts, log in, and manage their profiles. This module also ensures secure authentication for accessing other features of the bookstore.

Module2

Book Management Module

Enables users to search for books using keywords or categories and view details such as title, author, price, and condition(new or used). It handles book selection and provides an intuitive interface for exploring available books, ensuring an efficient search experience.

Module3

Shopping Cart Module

Allows users to add selected books to their cart, view and edit cart contents, and calculate the total price, including applicable discounts or offers. It provides options to update quantities or remove items, ensuring flexibility before proceeding to checkout.

Module4

Order Management Module

Handles shipping information, payment processing, and order validation to ensure accurate transactions. Users can choose payment methods and confirm shipping details. It securely processes orders and maintains records for future reference, ensuring smooth transaction management.

Module5

Admin Module

Enables administrators to manage the book inventory by adding, updating, or removing books from the catalog. It also allows monitoring of user orders, ensuring timely processing and delivery. Additionally, this module generates reports for business analytics, helping in decision-making and operational improvements.

CHAPTER 5

CONCLUSION

The **OnlinE Book Store Management System** is a comprehensive Java-based application that streamlines the process of purchasing books online.

It integrates key modules such as **User Management**, **Book Management**, **Shopping Cart**, **Order Management**, and **Admin Management** to ensure a seamless and efficient user experience.

Users can register, search for books, add items to their cart, and place orders with ease, while administrators can manage inventory and monitor orders. By leveraging Java's object-oriented principles and integrating database operations, the system provides a robust, scalable, and user-friendly platform that meets the demands of modern e-commerce.

This project not only automates book purchasing but also serves as a learning model for building modular and efficient software systems.

REFERENCES:

1) **GitHub - shashirajraja/online book store**: This project includes functionalities for managing, buying, adding, removing, and selling books, with login and logout security for both users and admins. GitHub Repository

2) **GitHub – Code By Aidan/online-book store-system**: An online bookstore system implemented in Java, allowing users to purchase books, leave reviews, control inventory, and process payments. GitHub Repository

3) **Code With C - Online Book Store Java Project**: This site provides the full source code and necessary documentation for an online book store projectin Java. Code With C

APPENDICES

APPENDIXA–SOURCECODE

PACKAGE NAME:online book store

MAIN CLASS

```
Package online book store ;

import javax.swing.*;

public class Main {

    Public static void main(String[]args){
        //Run the Online Book store App in the Swing event dispatch thread
        Swing Utilities.invokeLater(() -> {
            Online Book store App book store App=new Online Book store
            App();
            bookstoreApp.set Visible(true);
            // Display the app window
        });
    }
}
```

Online Book store App Class

```
Package online book store;
import javax.swing.*;
import java.awt.*;
import  java.awt.event.*;
import java.util.Array List;
import java.util.List;
public class Online Book store App extends J Frame{
    private List<String> books;
    private Array List<String> cart;
    private Card Layout card Layout;
    private J Panel card Panel; private
    J List<String> bookList;
    private Default List Model<String>list Model;
    private J Text Area cart Area;
    private J Text Field search Field;
    private J Text Field user name Field;
    private J Password Field password Field; private
    J Text Field payment Card Number Field;
    private J Text Field payment Expiry Date Field;
    private J Text Field payment CVV Field;
    public Online Book store App(){
    books = new Array List<>();
    cart = new Array List<>();
        books.add("The Great Gats by-$10");
        books.add("Moby Dick - $12");
        books.add("Clean Code-$25");
        books.add("Introduction to Algorithms - $50");
```

```
books.add("Effective  Java - $35");
books.add("The Pragmatic Programmer-$30");
books.add("Design Patterns:Elements of Object-Oriented Software-$45");
books.add("Python Crash Course - $40");
books.add("Head First Java-$35");
books.add("Java Script:The Good Parts -$28");
books.add("You Don't Know JS - $22");

books.add("To Kill a Mockingbird - $15");
books.add("1984 - $18");
books.add("Pride and Prejudice-$14");
books.add("The White Tiger - $15");
books.add("Chetan Bhagat-Five Point Someone-$10");
books.add("The God of Small Things - $18");
books.add("Midnight's Children - $22");
books.add("The Namesake - $17");
books.add("A Fine Balance - $19");
books.add("Shanta ram - $20");
books.add("The Inheritance of Loss-$14");
books.add("The Elephant Vanishes - $16");
books.add("Train to Pakistan - $18");
books.add("The Discovery of India - $13");
books.add("The Guide - $12");
books.add("Gitanjali - $14");
books.add("A Passage to India - $15"); // E.M. Forster
books.add("The Great Indian Novel - $17"); // Shashi Tharoor
books.add("The Immortals of Meluha-$16");//AmishTripathi
books.add("The Secret of the Nagas - $18"); // Amish Tripathi
books.add("Sacred Games - $19"); // Vikram Chandra
```

```
    set Title("Online Bookstore");
    set Size(600, 450);
    set Default Close Operation(J Frame.EXIT_ON_CLOSE);
    set Location Relative To(null);
    card Layout = new Card Layout();
     card Panel=new J Panel(card Layout);
    card Panel.add(create Login Page(), "Login");
    card Panel.add(create Book store
    Page(),"Bookstore");
    card Panel.add(create Cart Page(), "Cart"); card
    Panel.add(create Checkout Page(), "Checkout");
    add(card Panel);
}
```

//1.LoginPage

```
Private J Panel create Login Page(){
    J Panel login Panel=new J Panel(new Grid Layout(3,2,10,10));
    Login Panel.set Background(new Color(255,255,255));
    Font label Font = new Font("Arial", Font.BOLD, 14);
    username Field = new J Text Field(20);
    Password Field = new J Password Field(20); J
    Button login Button=new J Button("Login");
    Login Button.set Background(new Color(0,122,204));
    login Button.set  Foreground(Color.WHITE);
    login Button.add Action Listener(e -> login());

    J Label username Label = new J Label("Username:");
    username Label.set Font(labelFont);
    username Label.set Foreground(newColor(0,122,204));
```

```
J Label password Label = new J Label("Password:");
password Label.set Font(label Font);
password Label.set Foreground(new Color(0,122,204));
login Panel.add(username Label);
login Panel.add(username Field);
login Panel.add(password Label);
login Panel.add(password Field);
login Panel.add(login Button);
login Panel.set Layout(new Grid Bag Layout());
Grid Bag Constraints gbc=new Grid BagConstraints();
gbc.insets = new Insets(5, 5, 5, 5);
gbc.gridx=0;
gbc.gridy = 0;
login Panel.add(username Label, gbc);
gbc.gridx = 1;
login Panel.add(username Field, gbc);
gbc.gridx = 0;
gbc.gridy = 1;
login Panel.add(password Label, gbc);
gbc.gridx = 1;
login Panel.add(password Field, gbc);
gbc.gridx = 0;
gbc.gridy=2;
gbc.gridwidth=2;
gbc.fill=Grid Bag Constraints.HORIZONTAL;
login Panel.add(login Button, gbc);
return login Panel;
}
```

```
Private void login() {
    String user name=user name Field.get Text();
    String password=new String(password Field.get Password());

    if(username.equals("AkashS")&&password.equals("am23004")){
     card Layout.show(card Panel, "Bookstore");
    }else{
       J Option Pane.showMessage Dialog(this, "Invalid credentials. Please try again.");
    }
}
```

//2.BookstorePage(Search&DisplayBooks)

```
Private  J Panel create Book store Page() {
    J Panel book store Panel=new J Panel();
    Book store Panel.set Layout(new Border Layout());
    bookstorePanel.setBackground(newColor(240,248,255));//Lightbackground

    searchField = new J Text Field(20);
    searchField.set Font(new Font("Arial",Font.PLAIN,14));
    J Button searchButton = new J Button("Search");
    searchButton.set Background(new Color(0, 122, 204));
    searchButton.set Foreground(Color.WHITE);

    List Model=new Default List Model<>();
    for (String book : books) {
       List Model.add Element(book);
    }
```

```
    BookList = new J List<>(list Model);
    bookList.set Selection Mode(List Selection
    Model.SINGLE_SELECTION);
    J Scroll Pane list Scroll Pane = new J Scroll Pane(book List);
    searchButton.add Action Listener(e -> search Books());
    J Panel search Panel = new J Panel();
    search Panel.set Layout(new Border Layout());
    search Panel.add(new J Label("Search for a book:"),Border Layout.WEST);
    search Panel.add(searchField, BorderLayout.CENTER);
    search Panel.add(searchButton, BorderLayout.EAST);
    book store Panel.add(searchPanel, BorderLayout.NORTH);
    book store Panel.add(list Scroll Pane, BorderLayout.CENTER);
    J Button add To Cart Button = new JButton("Add to Cart");
    addToCart Button.set Background(new Color(0, 122, 204));
    addToCartButton.set Foreground(Color.WHITE);
    addToCartButton.add Action Listener(e -> addToCart());
    bookstore Panel.add(add To CartButton, BorderLayout.SOUTH);
    return bookstorePanel;
}
Private void search Books(){
    Stringquery=search Field.getText().to LowerCase();
    listModel.clear();
    for(Stringbook:books){
        if(book.toLowerCase().contains(query)){

listModel.add Element(book);
        }
    }
}
```

```
Private void add To Cart(){
    Strings electedBook=bookList.get Selected Value();
    if (selectedBook != null) {
        cart.add(selectedBook);
        J Option Pane.showMessageDialog(this,"Added to cart!");
        show Payment Details Dialog();
    }else{
        J Option Pane.showMessageDialog(this,"Please select a book to add to the
cart.");
    }
}
Private void show Payment Details Dialog(){
    J Panel paymentPanel=new J Panel(new Grid Layout(4,2,10,10));
    Payment Panel.set Background(new Color(255, 255, 255));
    Font label Font = new Font("Arial", Font.BOLD, 14);
    payment Card Number Field = new J Text Field(20);
    payment Expiry Date Field = new J Text Field(10);
    payment CVVField = new JTextField(4);
    paymentPanel.add(new JLabel("Card Number:"));
     payment Panel.add(paymentCardNumberField);
     payment Panel.add(newJLabel("ExpiryDate(MM/YY):"));
    payment Panel.add(paymentExpiryDateField);
     payment Panel.add(new JLabel("CVV:"));
     payment Panel.add(paymentCVVField);
    int option = J Option Pane.show Confirm Dialog(this, payment Panel,
    "Enter     Payment Details", J Option Pane.OK_CANCEL_OPTION,
     J Option Pane.PLAIN_MESSAGE);
    if(option==J Option Pane.OK_OPTION){
```

```
        Stringcard Number=payment Card Number Field.get Text();
        String expiry Date = payment Expiry Date Field.get Text();
        String cvv = payment CVV Field.get Text();

        if (cardNumber.isEmpty() || expiryDate.isEmpty() || cvv.isEmpty()) {
         J Option Pane.show Message Dialog(this,"Please enter all payment details.");
        }else{
           J Option Pane.show Message Dialog(this,"Payment Successful.Proceeding to checkout.");
           Card Layout.show(cardPanel,"Cart");//Showcartafterpayment
        }
      }
   }
```

//3.CartPage

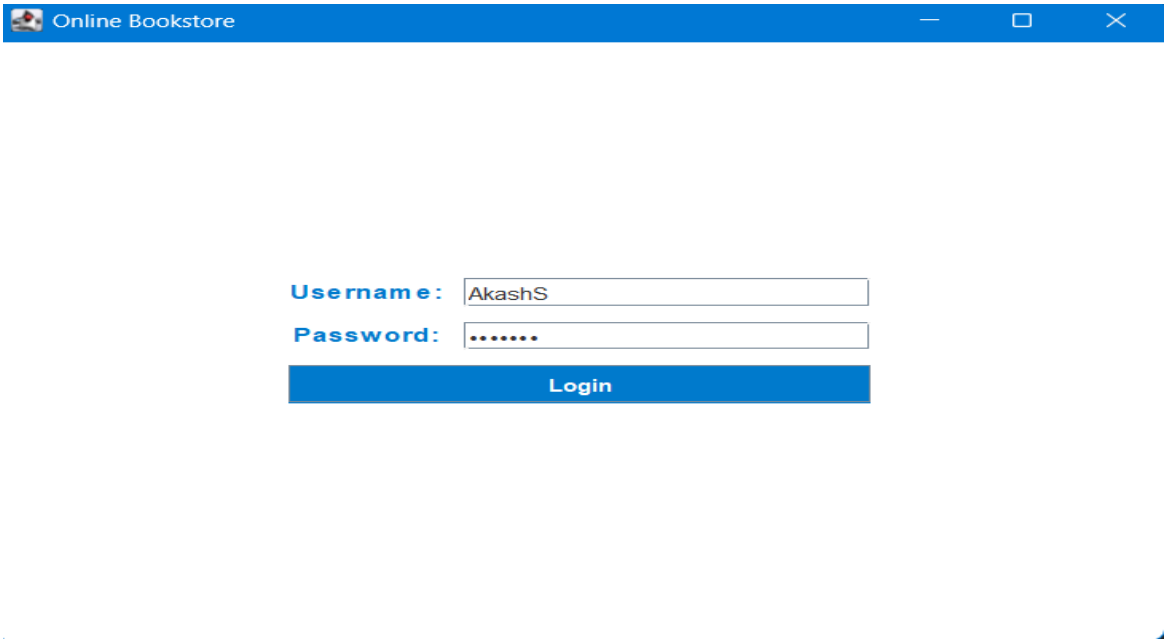
```
   Private J Panelcreate Cart Page(){
      J PanelcartPanel=new J Panel(newBorderLayout());
      cartPanel.set Background(newColor(240,248,255));

      cartArea=newJTextArea(10,30);
      cartArea.setEditable(false);
      JScrollPanecartScrollPane=newJScrollPane(cartArea);

      JButton checkoutButton = new JButton("Proceed to Checkout");
      checkoutButton.setBackground(new Color(0, 122, 204));
      checkoutButton.setForeground(Color.WHITE);
      checkoutButton.addActionListener(e->cardLayout.show(cardPanel, "Checkout"));
```

```
        cartPanel.add(newJLabel("YourCart:"),BorderLayout.NORTH);
        cartPanel.add(cartScrollPane, BorderLayout.CENTER);
        cartPanel.add(checkoutButton, BorderLayout.SOUTH);
        returncartPanel;
    }

    //UpdateCartDisplay
    privatevoidupdateCartDisplay(){
        cartArea.setText("");
        for (String item : cart) {
            cartArea.append(item+"\n");
        }
    }
    //4.CheckoutPage(PaymentProcess)
    privateJPanelcreateCheckoutPage(){
        JPanelcheckoutPanel=newJPanel(newBorderLayout());
        checkoutPanel.setBackground(newColor(240,248,255));
        JTextArea orderSummary = new JTextArea(10, 30);
        orderSummary.setEditable(false);
        JScrollPane orderScrollPane = new JScrollPane(orderSummary);
        JButton placeOrderButton = new JButton("Place Order");
        placeOrderButton.setBackground(new Color(0, 122, 204));
        placeOrderButton.setForeground(Color.WHITE);
        placeOrderButton.addActionListener(e->placeOrder(orderSummary));
        checkoutPanel.add(new JLabel("Checkout - Order Summary:"),
BorderLayout.NORTH);
        checkoutPanel.add(orderScrollPane, BorderLayout.CENTER);
        checkoutPanel.add(placeOrderButton, BorderLayout.SOUTH);
```

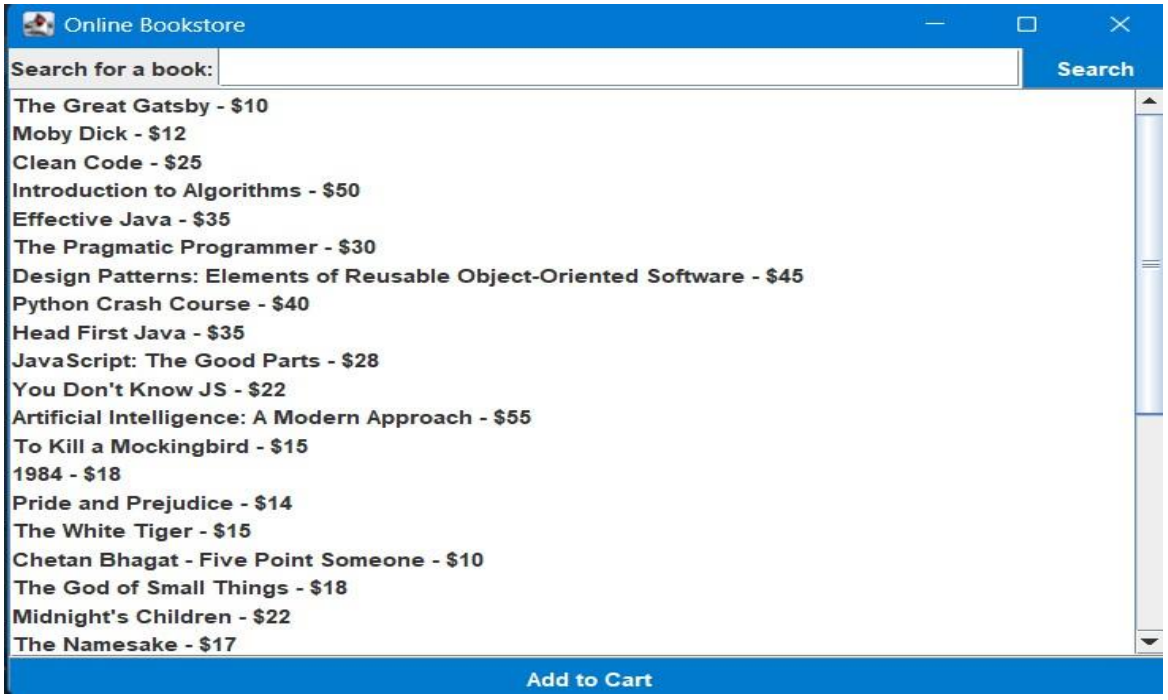
```
        returncheckoutPanel;
    }

    privatevoidplaceOrder(JTextAreaorderSummary){
        StringBuilderreceipt=newStringBuilder("OrderSummary:\n\n"); double
        total = 0;
        for (String item : cart) {
            receipt.append(item).append("\n");
            total += extractPrice(item);
        }
        receipt.append("\nTotal: $").append(total);
        orderSummary.setText(receipt.toString());

        intoption =JOptionPane.showConfirmDialog(this,"Proceedwith payment?",
"Payment", JOptionPane.YES_NO_OPTION);
        if(option==JOptionPane.YES_OPTION){
            JOptionPane.showMessageDialog(this, "Payment Successful. Thank you for
your purchase!");
            cart.clear();
            updateCartDisplay();
            cardLayout.show(cardPanel,"Login");
        }
    }
    privatedoubleextractPrice(StringbookDetails){
        String[] parts = bookDetails.split(" - \\$");
        return Double.parseDouble(parts[1]);
    }
}
```

APPENDIXB-SCREENSHOTS

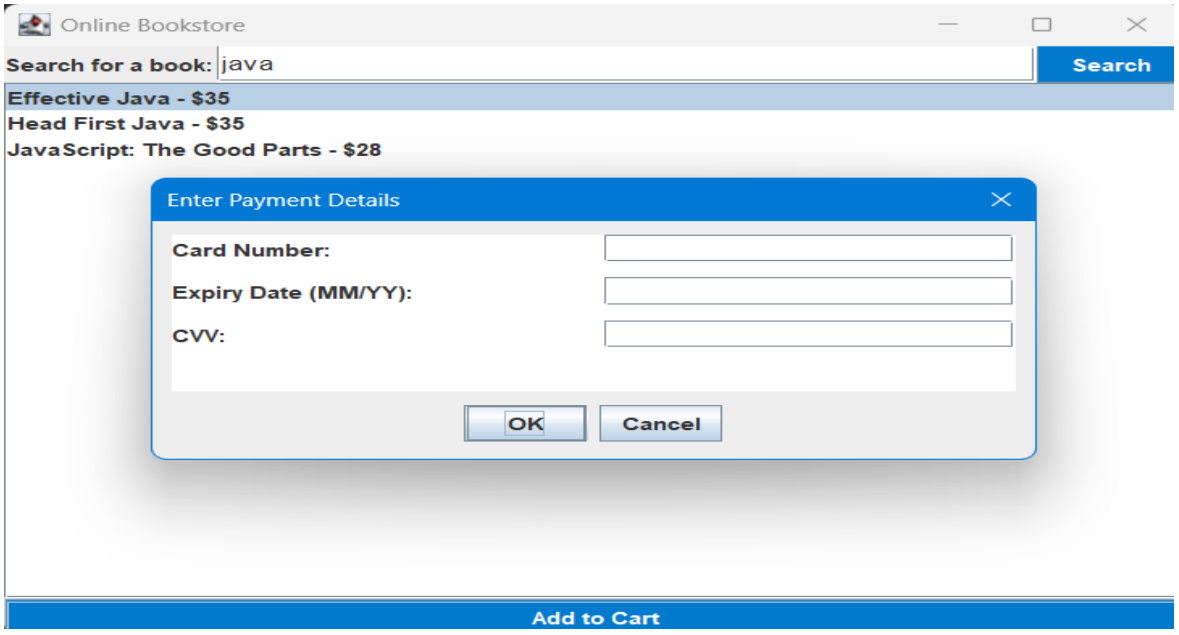
1 LOGINPAGE:



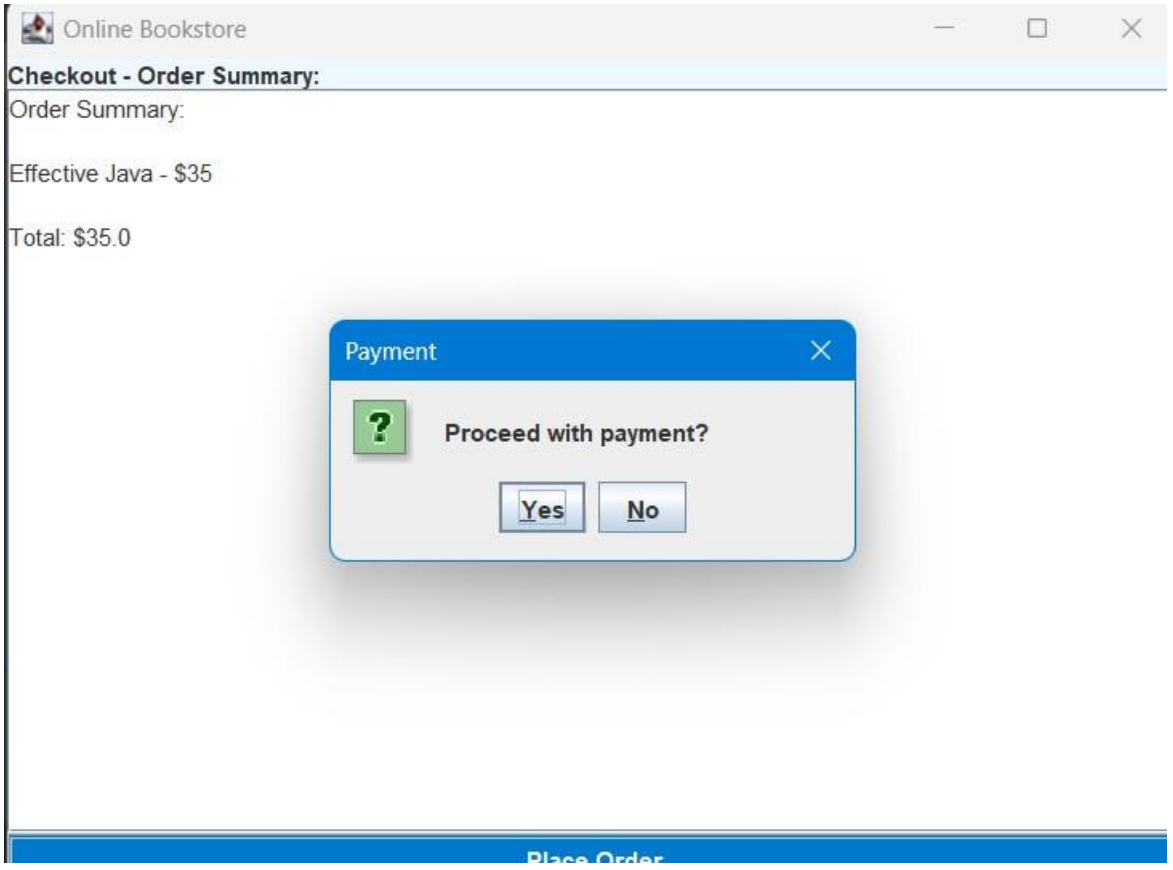
2 SEARCHPAGE



3 PAYMENTPROCESS



4 SUMMARYCHECK



5 FINISH&LOGOUT