IACSD

INSTITUTE FOR ADVANCED COMPUTING AND
SOFTWARE DEVELOPMENT AKURDI, PUNE

Documentation On

"E-BookStore"- Online Books Store

PG-DAC SEPT 2022

Submitted By:

Group No: 9

Roll No.	Name:
229011	**Anuja Gupta**
229018	**Bhagyashri Badhe**

Mr. Rohit Puranik   **Mrs. Sonali Mogal**

Centre Coordinator   **Project Guide**

ACKNOWLEDGEMENT

This project was a great learning experience for us and we are submitting this work to Institute for Advanced Computing & Software Development (IACSD) Akurdi , Pune.

We are very glad to mention the name of Ms. Sonali Mogal for her valuable guidance to work on this project.

Our heartfelt thanks go to Ms. Rohit Puranik (Course Coordinator, PG_DAC) who gave us all the required support and kind coordination to provide all the necessities to complete the project and throughout the course up to the last day of the course.

We would like to express our sincere gratitude towards Mrs. Madhura Anturkar, our faculty for J2SE and J2EE, who was always there for us. Her guidance and support helped us overcome various obstacles and intricacies during the course of our project work. Without her tremendous support, guidance, and efforts, this project would not have been possible.

From:

Anuja Gupta   220941220020
Bhagyashri Badhe   220941220032

TABLE OF CONTENTS

5. TESTING

7. CONCLUSION

8. FUTURE SCOPE

9. REFERENCES

ABSTRACT

Nowadays, the network plays an important role in people's life. In the process of the improvement of the people's living standard, people's demands of the life's quality and efficiency is more higher, the traditional bookstore's inconvenience gradually emerge, and the online bookstore has gradually be used in public. The online bookstore is a revolution of book industry. The traditional bookstores' operation time, address and space is limited, so the types of books and books to find received a degree of restriction. But the online bookstore broke the management mode of traditional bookstore, as long as you have a computer, you can buy the book anywhere, saving time and effort, shortening the time of book selection link effectively. The online bookstore system based on the principle of provides convenience and service to people

1. INTRODUCTION

Whenever we need something that we don't have, we go to a store and purchase it. But sometimes, even in the store those items are not available. So we go to a mall to get everything we need, all at once. It's a one stop destination. But what if we can bring the whole mall at your own house? One can do all the shopping they need to, and can check thousands of products without even stepping outside their house. That is the idea which gave birth to the concept of online shopping.

Not only you can get exactly what you want when you shop online, you can also try to get a better deal by browsing across some other online stores.

This project "E- BookStore" is an effort to cater the people by providing them with books of various authors. From your favorite author, publication to your favorite Genre, all in one place.

From a customer's point of view, they can browse through a catalogue of books, search for books, add books to their cart and place orders. Customers can even create an account and add addresses. They can easily browse books via category feature. register their account login to their account, update their account, delete their account

From the admin's point of view, the admin can view seller details, customer details, order details and can also delete customer and seller accounts if needed. register their account login to their account, update their account, delete their account

From the seller's point of view, they can add a books, update the books, new sellers can register their account, login to their account, update their account details and delete their account.

2. PRODUCT OVERVIEW AND SUMMARY

2.1. PURPOSE

Our project, "E-BookStore", is a web-based online e-commerce website which aims to provide users with an easy to navigate and visually appealing medium to browse through a catalogue of products and shop for the products they desire.

2.2. SCOPE

"E-BookStore" aims to deliver a web-based application that hosts a wide collection of books that users can browse through. Users can add books to cart and place orders for each book. They can update their profile, add delivery address and add payment details.

They can view their current order as well. seller can manage various book details like price, adding new books, updating existing books etc.

This project does not support the actual logistics and delivery of product and actual payment logic. We are assuming that the organization that implements it will be using third-party payment API which can easily be integrated in our application if needed.

E-BookStore is only an interface for both customers (for browsing and shopping for books) and sellers (for managing books).

2.3. OVERVIEW

A. TECHNOLOGIES USED
  - ➢ FRONT END
    - a. Bootstrap
    - b. React

  - ➢ BACK END
    - a. Spring Boot

  - ➢ DATABASE MANAGEMENT SYSTEM
    - a. MySQL

B. FEATURES PROVIDED

  - ➢ FOR CUSTOMERS
    - a. Browse – Customers can browse the homepage to explore the entire collection of books available, as well as view details for individual books.
    - b. View– When a customer checks book details, they can also view the details.
    - c. Register, Login & Logout – New customers can register on the site. Existing customers can then login to access their account information and logout when the account is not in use.
    - d. View & Update Profile – When logged in, customers can view their profile and update their details.
    - e. Add to Cart & Place Orders – If customers find books that they like, they can save the books in the cart until they decide to purchase it. When they wish to purchase it, they can place orders for those books by selecting a delivery address.

f. View Order Details – Every customer can view their order history in order to get an idea about their past spendings and how much they saved on each order.

> ➤ FOR SELLER

  a. Register, Login & Logout – Similar to customers, sellers can register their account, login & logout to access their account.

  b. Delete book –The seller can delete a book if they need to for any purpose.

  c. Add New book –seller can add new book with all the necessary details like book name, author name, genre, price, etc.

  d. Update book Details – Often, the details of a book such as the price, may change over time. In such cases, the seller can update those book details to reflect the change on the application.

> ➤ FOR ADMIN

  a. Login & Logout – Similarly admin can login & logout to access their account.

  b. Delete customer and seller – The admin can delete a customer or seller if they need to for any purpose.

  c. Admin can also view order history.

  d. Admin can also update details like password

2.4. FEASIBILITY STUDY

Feasibility is the determination of whether a project is worth undertaking or not. Before actually recommending the new system, it is important to investigate if it is feasible to develop it. Before developing and implementing a system, we have to make sure that the system is feasible in the following ways:

A. TECHNICAL FEASIBILITY

In this type of feasibility study, the system analyst has to check whether it is possible or not to develop the requested system with the available manpower, software, hardware, etc. This project makes use of cross-platform software and solutions like Java, and hence can run on any operating system. React, used in front-end, is swift and light weight framework when it comes to delivering the requested page as it doesn't reload the entire page for every HTTP request. It only re-renders the components that need to fetch new data. Also, as React is modular in nature, it is easy to develop new components and scale up existing components in order to add new features to the system. The combination of Spring Boot for backend makes for a fast, easy to set-up and reliable system to interact with the database, as they are secure and transactional in nature. Since the sensitive data of customers and admins need to be stored in a robust and secure database, MySQL database management system was chosen as it is an industry standard.

B. OPERATIONAL FEASIBILITY

In this type of feasibility study, the operation of the system is considered. An analysis is performed on whether it is feasible for the user department to use the application. Thus, the proposed system is said to be operationally feasible only if clients are able to understand the system clearly and correctly, and can use it with ease.

In the design of this project, we always kept user experience in mind. We made an effort to have a good user interface with consistent theme and alluring design to keep the users interested and engaged. In our project, the use of universally known icons and instructions that are easy to understand makes sure that the user will not need any special technical know-how to use the application. We made sure that the information available throughout the application is arranged in a logically coherent and consistent manner, guaranteeing that the users will have a smooth and effortless experience and even enjoy using the application.

C. ECONOMIC FEASIBILITY

In this type of feasibility study, the benefits of the system to the organization are considered by taking into consideration the cost-benefit analysis. All the software and technologies used in our project free, open-source, and widely available, with each of the technologies having an extensive community support. This makes "E-BookStore" an economically feasible solution to the organizations that wish to implement it.

3.REQUIREMENTS FULFILLED

3.1 FUNCTIONAL REQUIREMENTS

Following are the functional requirements fulfilled by our project:

➢ Customers can browse through all available books.

➢ Customers can place orders for books and view their order history.

➢ Sellers can manage various book details like book price, adding new books, etc.

➢ Sellers can update the book details.

➢ Sellers can update their account details and delete their account if needed.

➢ Admins can delete customers and sellers if the need arises.

➢ Admins can view all the orders placed by the customers.

3.2 NON-FUNCTIONAL REQUIREMENTS

Following are the non-functional requirements fulfilled by our project:

➢ Since the application uses lightweight and established software components that are also cross-platform, it is remarkably performant and has good support for every operating system.

➢ The use of React for front end and Spring Boot for back end delivers quick response times to admins and customers alike.

➢ Card-style UI and well-known icons and symbols used throughout the application provides a consistent theme and user-friendly interface that anyone can grasp easily, even without a technical background.

IACSD

4. PROJECT DESIGN

4.1.DATA MODEL

The following tables depict the database design used for "E-Store" application:

❖ Tables Related to User Details

```
mysql> desc addresses_table;
+------------+--------------+------+-----+---------+----------------+
| Field      | Type         | Null | Key | Default | Extra          |
+------------+--------------+------+-----+---------+----------------+
| address_id | int          | NO   | PRI | NULL    | auto_increment |
| city       | varchar(255) | YES  |     | NULL    |                |
| country    | varchar(255) | YES  |     | NULL    |                |
| state      | varchar(255) | YES  |     | NULL    |                |
| zip        | varchar(255) | YES  |     | NULL    |                |
+------------+--------------+------+-----+---------+----------------+
5 rows in set (0.30 sec)
```

4.1.1 Addresses Table

```
mysql> desc admins_table;
+----------------+--------------+------+-----+---------+-------+
| Field          | Type         | Null | Key | Default | Extra |
+----------------+--------------+------+-----+---------+-------+
| admin_id       | varchar(255) | NO   | PRI | NULL    |       |
| admin_password | varchar(255) | YES  |     | NULL    |       |
| admin_name     | varchar(255) | YES  |     | NULL    |       |
+----------------+--------------+------+-----+---------+-------+
3 rows in set (0.11 sec)
```

4.1.2 Admin Table

12

```
mysql> desc customers_table;
+------------------+--------------+------+-----+---------+----------------+
| Field            | Type         | Null | Key | Default | Extra          |
+------------------+--------------+------+-----+---------+----------------+
| customer_id      | int          | NO   | PRI | NULL    | auto_increment |
| city             | varchar(255) | YES  |     | NULL    |                |
| created_timestamp| date         | YES  |     | NULL    |                |
| gender           | varchar(255) | YES  |     | NULL    |                |
| customer_name    | varchar(255) | YES  |     | NULL    |                |
| customer_phoneNo | varchar(255) | YES  |     | NULL    |                |
| customer_password| varchar(255) | YES  |     | NULL    |                |
| customer_emailID | varchar(255) | YES  |     | NULL    |                |
+------------------+--------------+------+-----+---------+----------------+
8 rows in set (0.01 sec)
```

4.1.3 Customers Table

```
mysql> desc orderdetails;
+---------+------+------+-----+---------+----------------+
| Field   | Type | Null | Key | Default | Extra          |
+---------+------+------+-----+---------+----------------+
| id      | int  | NO   | PRI | NULL    | auto_increment |
| qty     | int  | NO   |     | NULL    |                |
| bookId  | int  | YES  | MUL | NULL    |                |
| orderId | int  | YES  | MUL | NULL    |                |
+---------+------+------+-----+---------+----------------+
4 rows in set (0.00 sec)
```

4.1.4 Order Details Table

```
mysql> desc orders_table;
+-------------+------+------+-----+---------+----------------+
| Field       | Type | Null | Key | Default | Extra          |
+-------------+------+------+-----+---------+----------------+
| order_id    | int  | NO   | PRI | NULL    | auto_increment |
| order_date  | date | YES  |     | NULL    |                |
| address_id  | int  | YES  | MUL | NULL    |                |
| customer_id | int  | YES  | MUL | NULL    |                |
| payment_id  | int  | YES  | MUL | NULL    |                |
+-------------+------+------+-----+---------+----------------+
5 rows in set (0.12 sec)
```

4.1.5 Orders Table

```
mysql> desc payments_table;
+--------------+--------------+------+-----+---------+----------------+
| Field        | Type         | Null | Key | Default | Extra          |
+--------------+--------------+------+-----+---------+----------------+
| payment_id   | int          | NO   | PRI | NULL    | auto_increment |
| amount       | int          | YES  |     | NULL    |                |
| card_no      | varchar(255) | YES  |     | NULL    |                |
| name_on_card | varchar(255) | YES  |     | NULL    |                |
| payment_date | datetime(6)  | YES  |     | NULL    |                |
+--------------+--------------+------+-----+---------+----------------+
5 rows in set (0.10 sec)
```

4.1.6 Payments Table

```
mysql> desc book;
+-------------------+--------------+------+-----+---------+----------------+
| Field             | Type         | Null | Key | Default | Extra          |
+-------------------+--------------+------+-----+---------+----------------+
| id                | int          | NO   | PRI | NULL    | auto_increment |
| authorName        | varchar(255) | YES  |     | NULL    |                |
| bcat              | varchar(255) | YES  |     | NULL    |                |
| bookName          | varchar(255) | YES  |     | NULL    |                |
| created_timestamp | date         | YES  |     | NULL    |                |
| photo             | varchar(255) | YES  |     | NULL    |                |
| price             | int          | NO   |     | NULL    |                |
| subcat            | varchar(255) | YES  |     | NULL    |                |
| sellerId          | int          | YES  | MUL | NULL    |                |
+-------------------+--------------+------+-----+---------+----------------+
9 rows in set (0.02 sec)
```
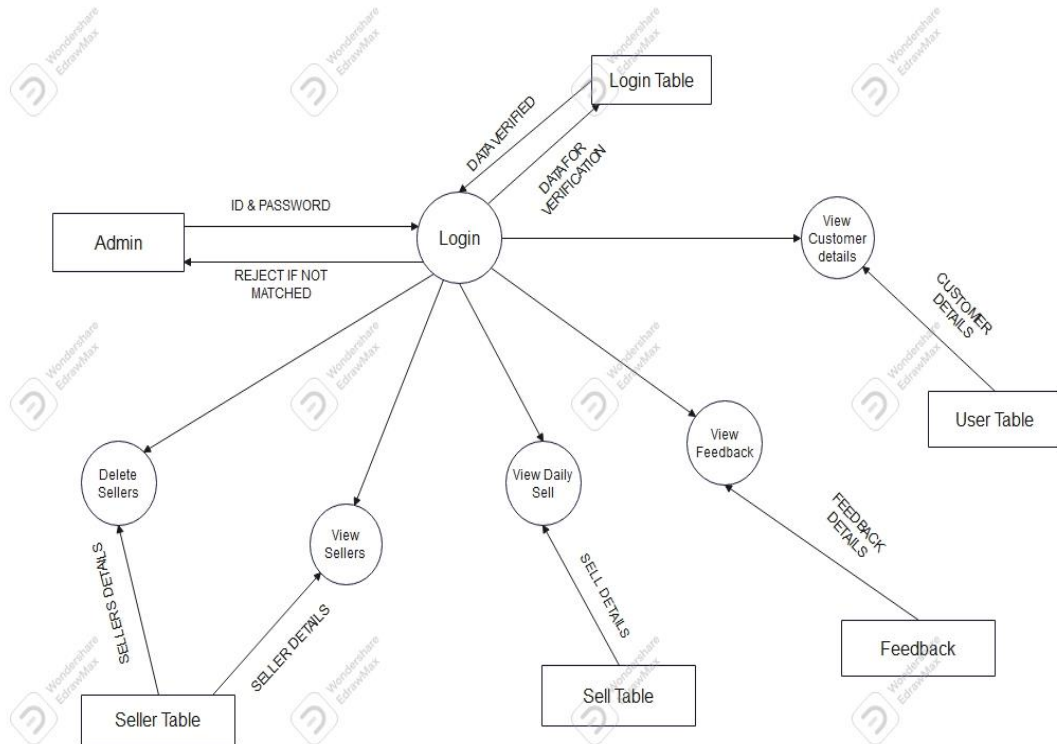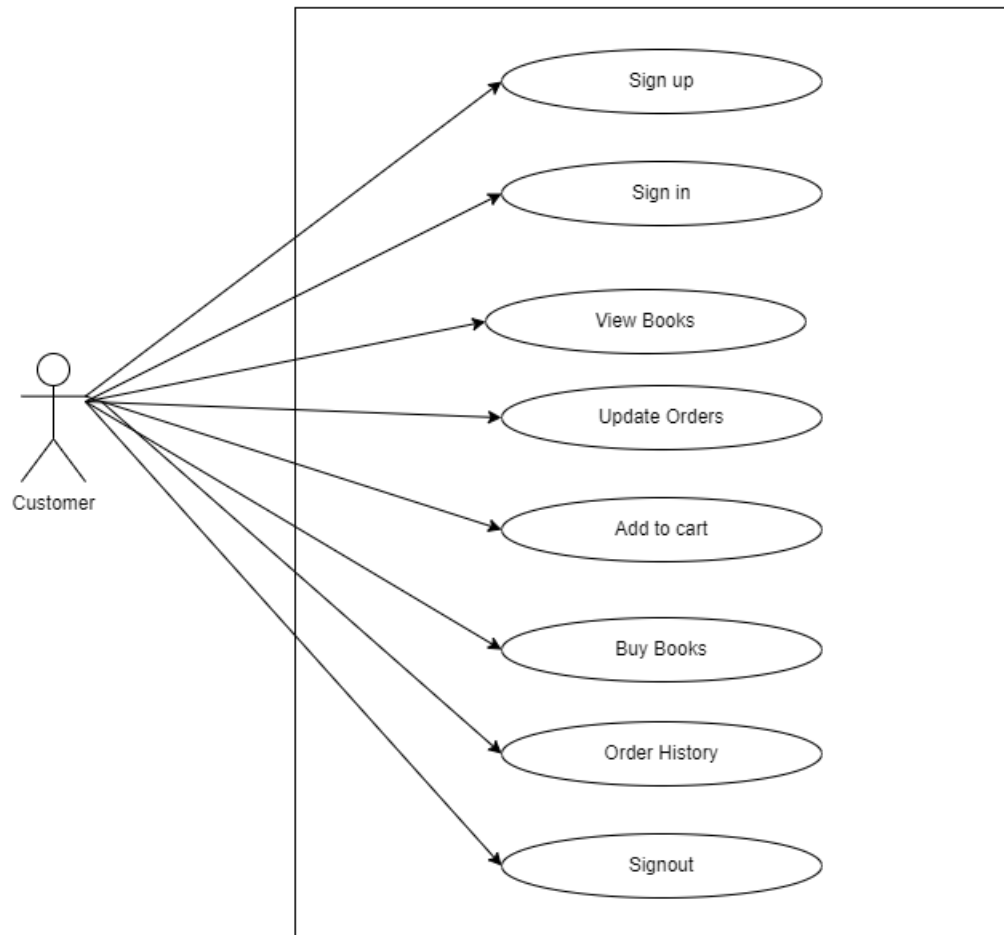
4.1.7 Books Table

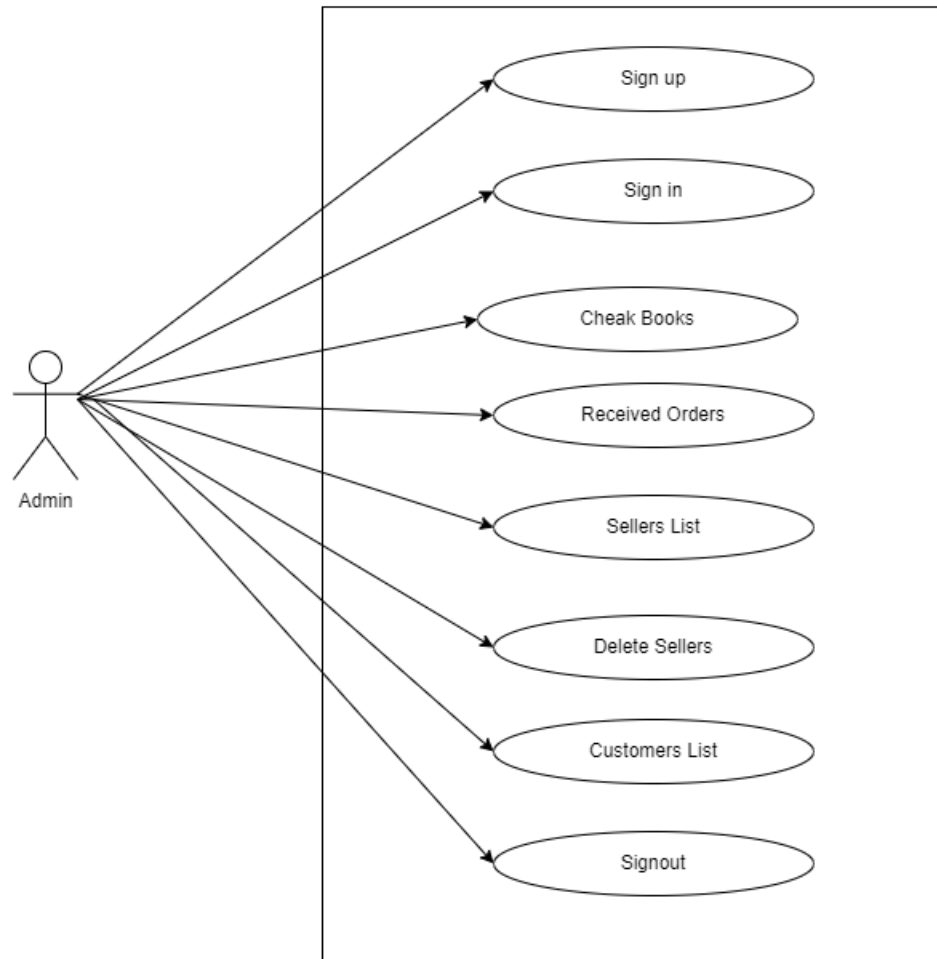# 4.2.DATA FLOW DIAGRAM

ZERO LEVEL DFD
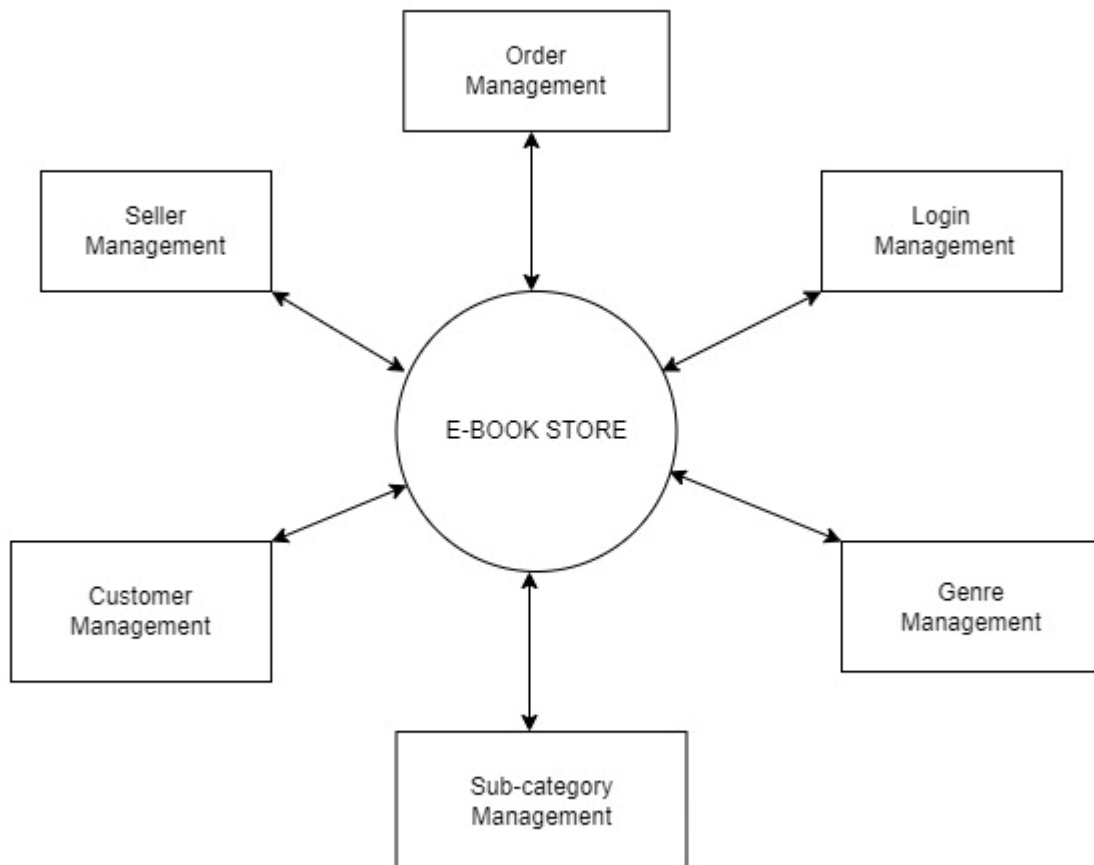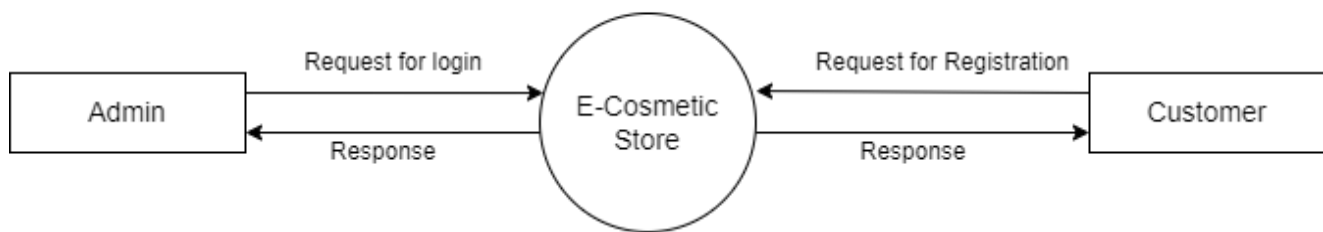
4.3.USE CASE DIAGRAM
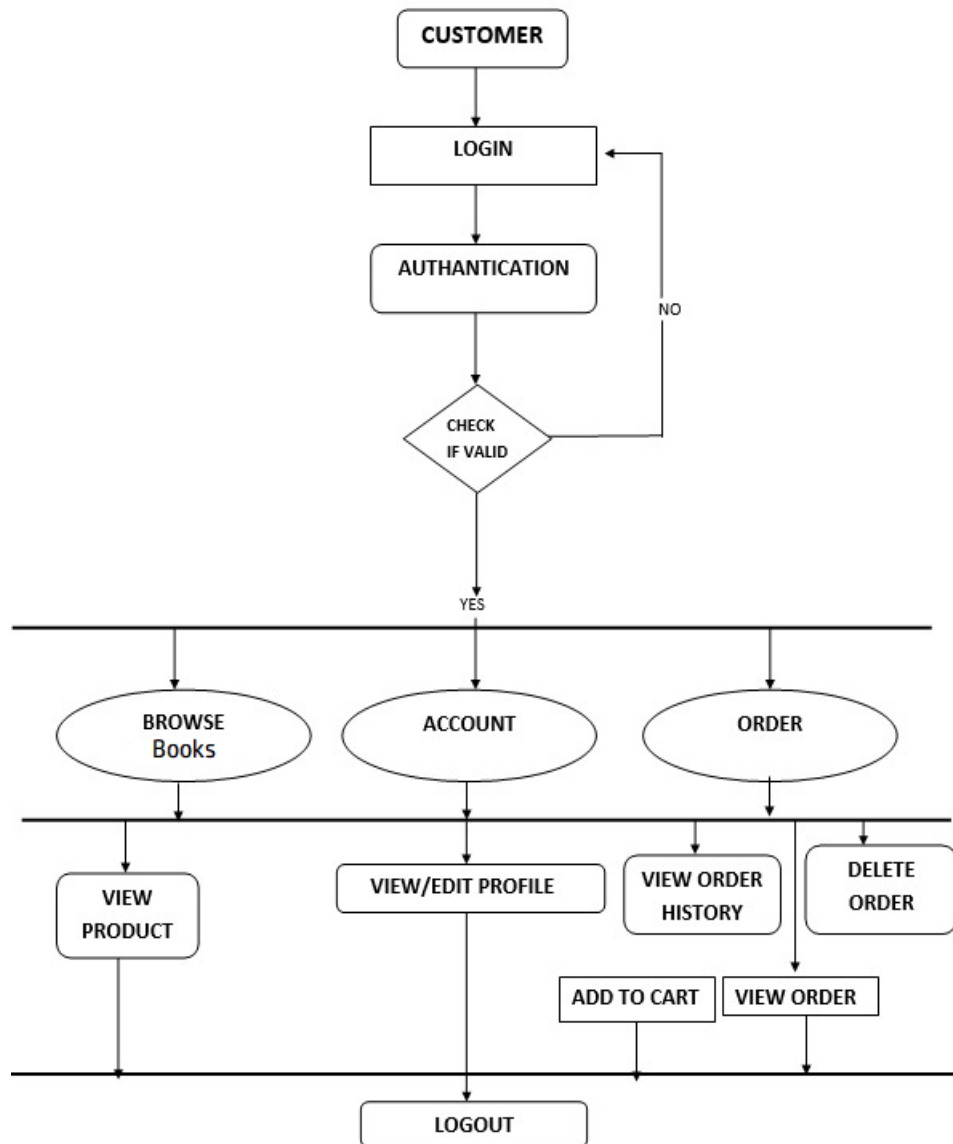## A. Customer

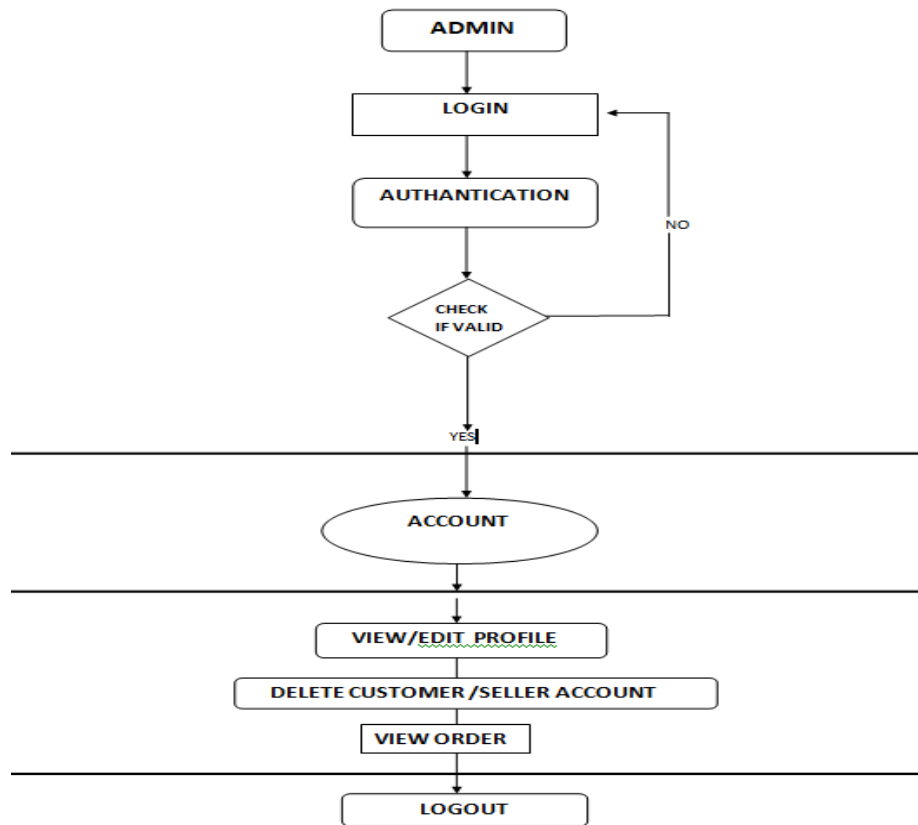B.Cstomer

B. Admin

A.**Admin**

C.SELLER

B.Seller
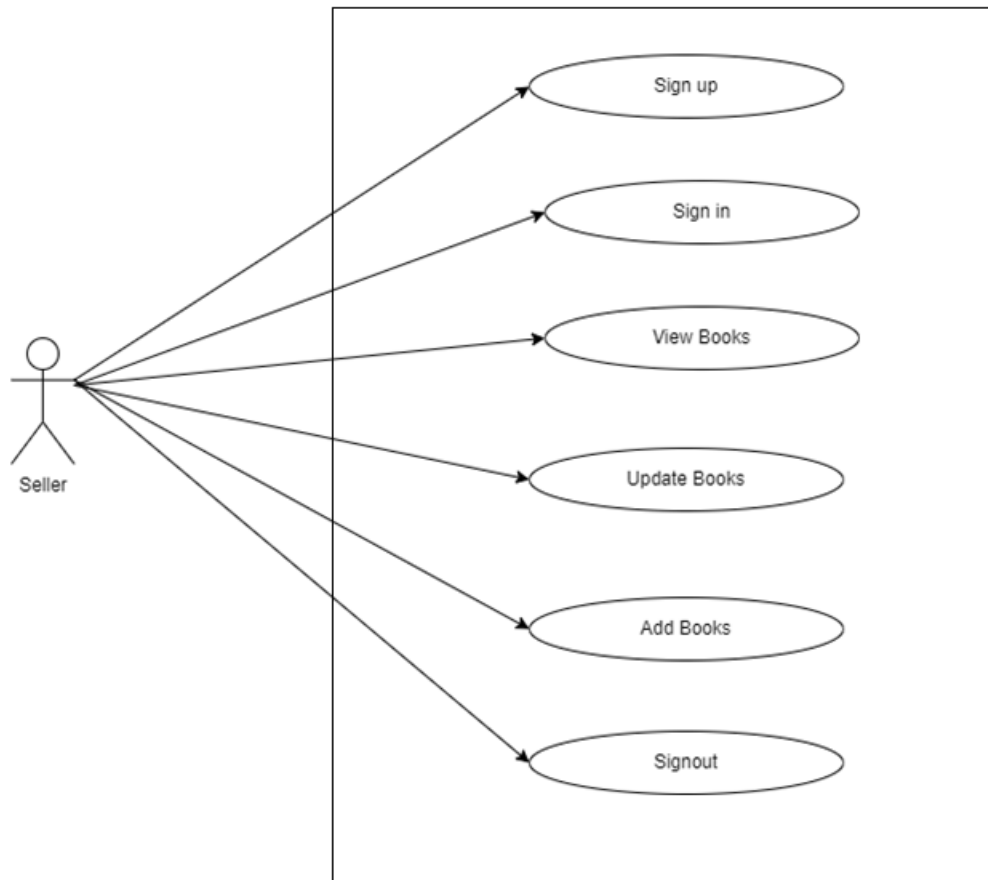
C.SELLER

4.4.ACTIVITY DIAGRAM

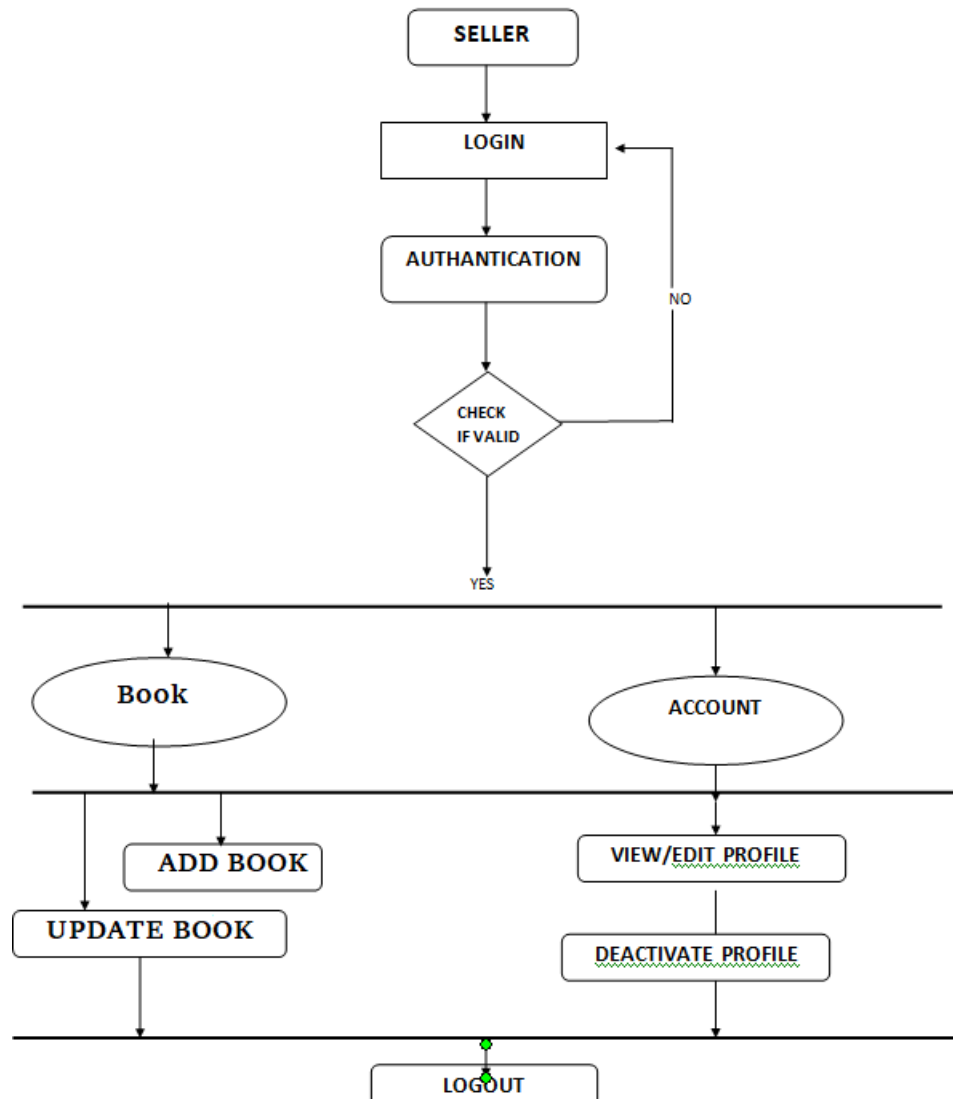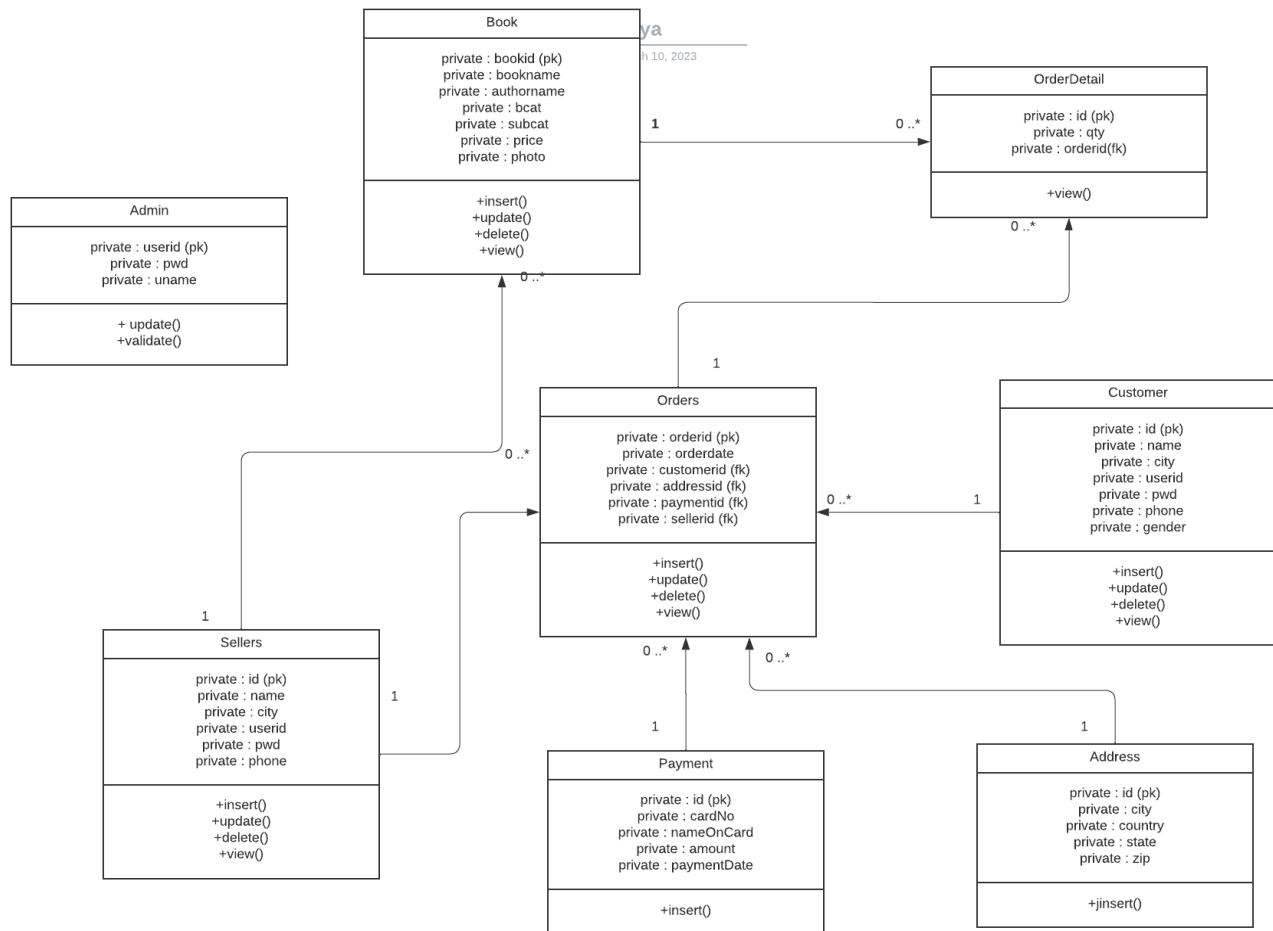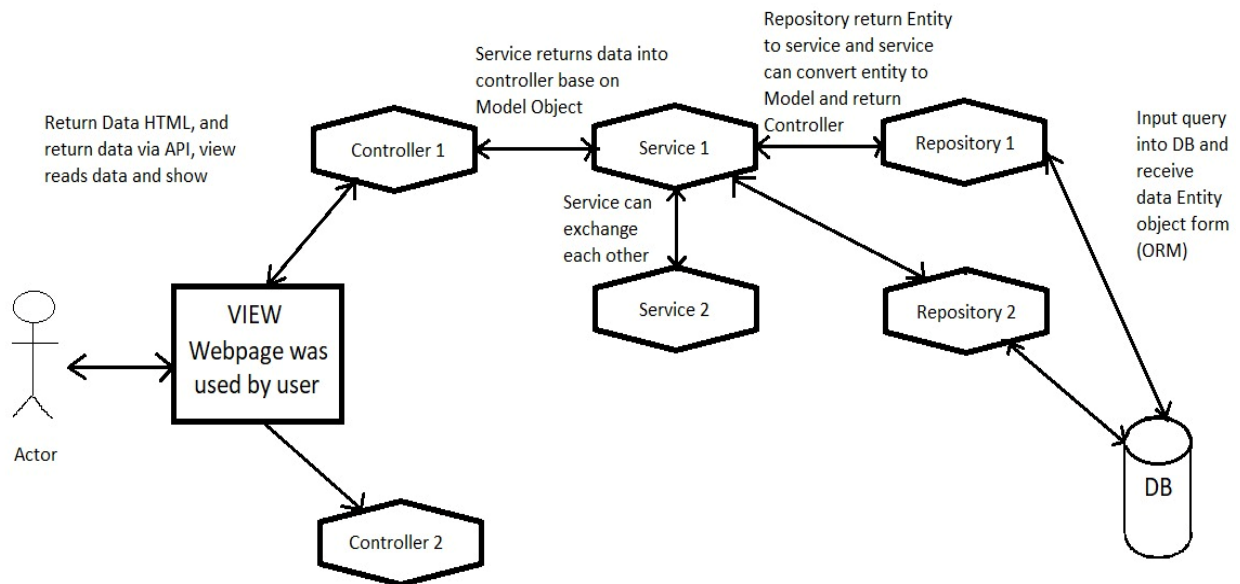## A. Customer

B. Admin

C.SELLER

4.5.CLASS DIAGRAM


```
Book
─────────────────────────
private : bookid (pk)
private : bookname
private : authorname
private : bcat
private : subcat
private : price
private : photo
─────────────────────────
+insert()
+update()
+delete()
+view()
```

```
OrderDetail
─────────────────────────
private : id (pk)
private : qty
private : orderid(fk)
─────────────────────────
+view()
```

```
Admin
─────────────────────────
private : userid (pk)
private : pwd
private : uname
─────────────────────────
+ update()
+validate()
```

```
Orders
─────────────────────────
private : orderid (pk)
private : orderdate
private : customerid (fk)
private : addressid (fk)
private : paymentid (fk)
private : sellerid (fk)
─────────────────────────
+insert()
+update()
+delete()
+view()
```

```
Customer
─────────────────────────
private : id (pk)
private : name
private : city
private : userid
private : pwd
private : phone
private : gender
─────────────────────────
+insert()
+update()
+delete()
+view()
```

```
Sellers
─────────────────────────
private : id (pk)
private : name
private : city
private : userid
private : pwd
private : phone
─────────────────────────
+insert()
+update()
+delete()
+view()
```

```
Payment
─────────────────────────
private : id (pk)
private : cardNo
private : nameOnCard
private : amount
private : paymentDate
─────────────────────────
+insert()
```

```
Address
─────────────────────────
private : id (pk)
private : city
private : country
private : state
private : zip
─────────────────────────
+jinsert()
```
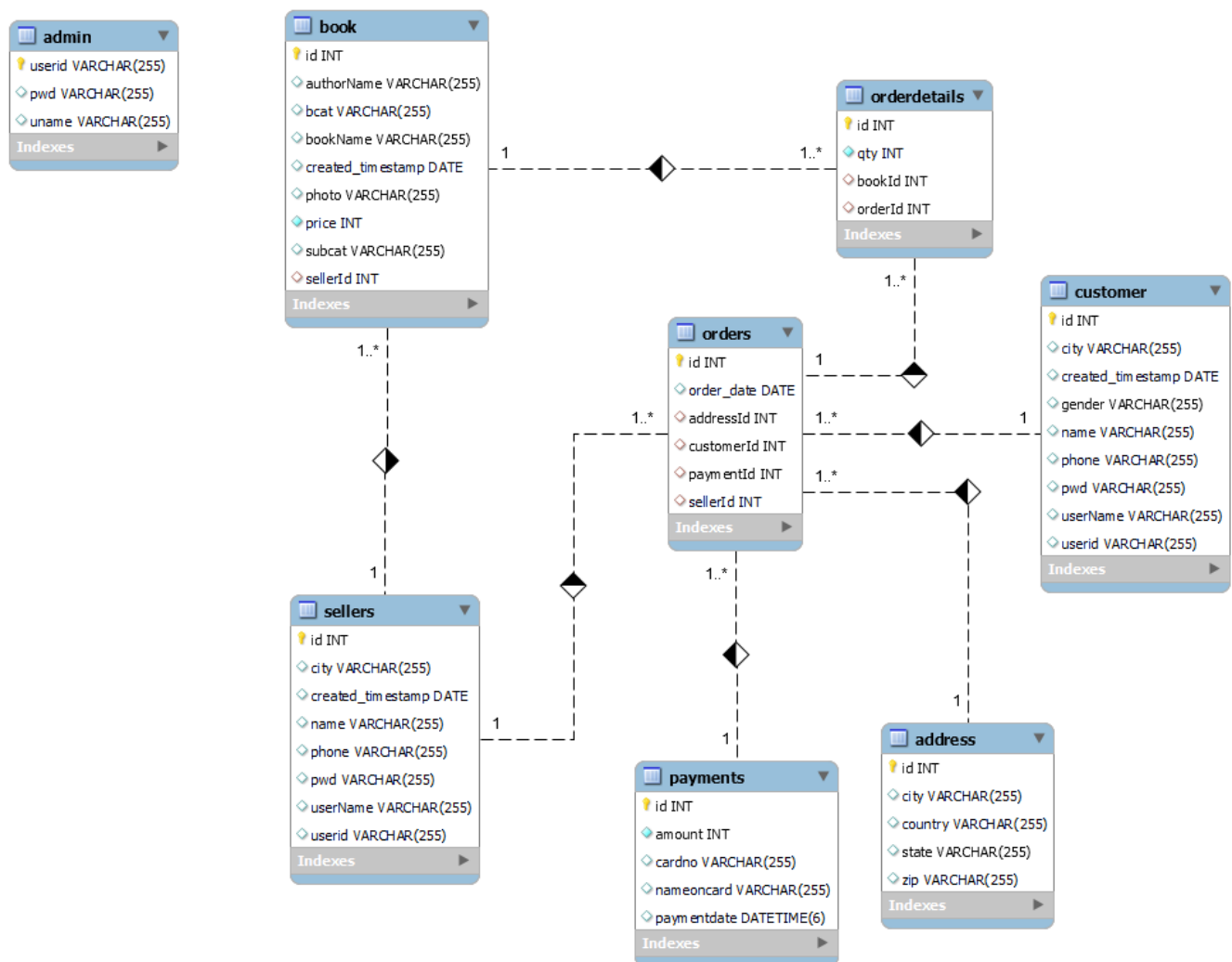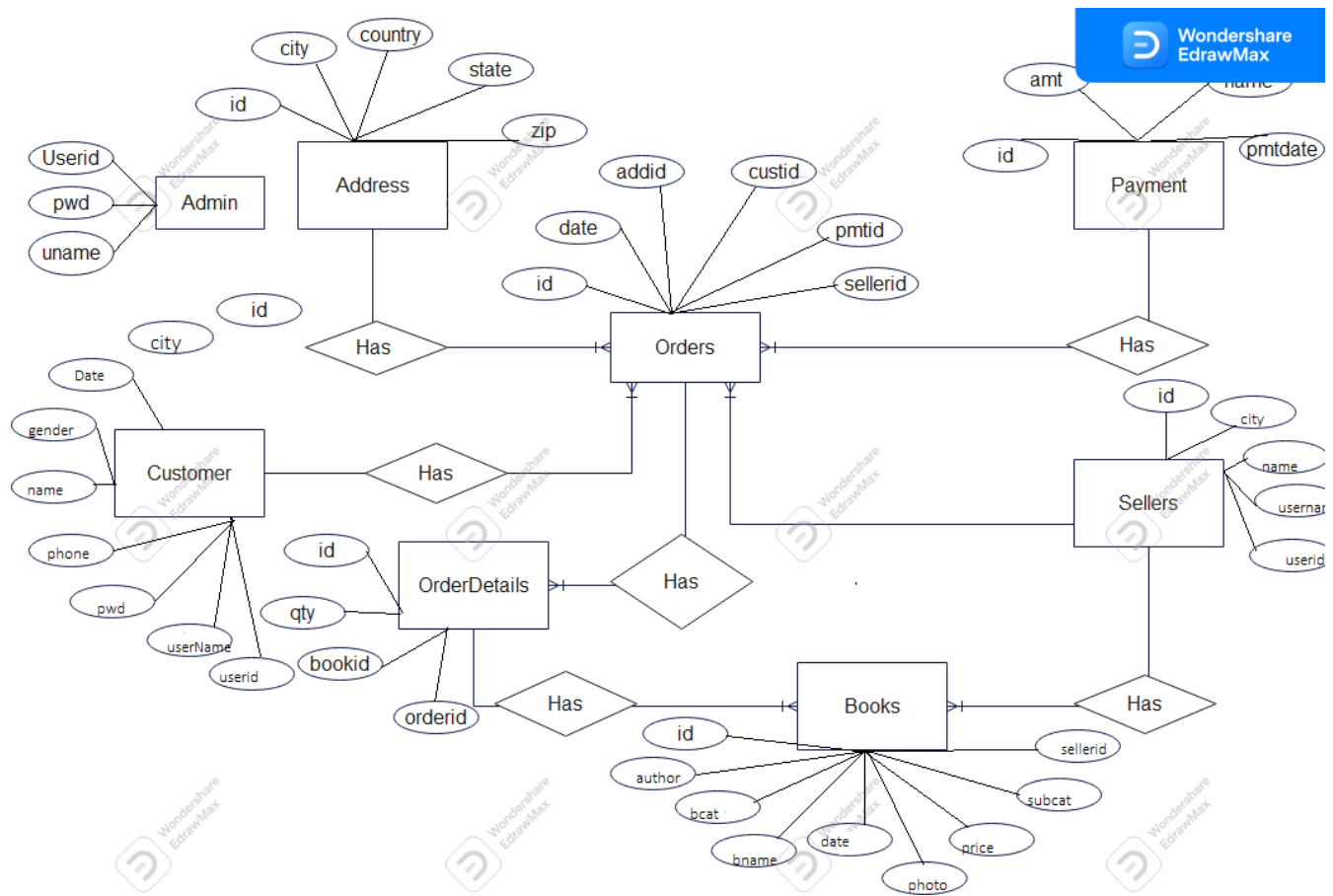
4.6.PROJECT ARCHITECTURE

4.7.ER DIAGRAM

5. TESTING

One of the main purposes of testing is to validate and verify that the system works as intended. No program or system design is perfect. However, if we implement the system without proper testing, then it may cause problems and lead to a bad user experience.

Testing and checking outcomes of each test gives us the best chance to detect and correct errors before the system is implemented in a production environment.

In the course of our project, we made an effort to manually test each component. In all cases, we obtained the desired results as demonstrated below.

A. CUSTOMER FEATURES TEST

Description Outcome Result

1. Register as Customer

1	Register new customer	The details were saved in database.	Passed

2. Login as Customer

1	Login	Successfully fetched from database and authenticated	Passed

3. Browse products

1	Browse books	Successfully fetched the books from database	Passed

4. Add product to Cart

1	Add book to cart	The book was successfully added to the cart	Passed

5. View Cart

1	View Cart	The cart details were fetched from the database	Passed

6. Place order

1	Place order	The order was successfully placed.	Passed

7.View Order History

1	View Order	The orders were fetched from the database.	Passed

8. Delete order

1	Delete order	Order was successfully removed from the database	Passed

9. Update customer profile

1	Update profile	Customer details were successfully updated	Passed

10. Logout

1	Logout	The session was cleared.	Passed

B. SELLER FEATURES TEST

Description Outcome Result

1. Register as Seller

| 1 | Register new seller | The details were saved in database. | Passed |

2. Login as Seller

| 1 | Login | Successfully fetched from database and authenticated | Passed |

3. Add product

| 1 | Add book | The book was successfully added to the database | Passed |

4. Update product details

| 1 | Update book | Book details were successfully updated | Passed |

5. Logout

| 1 | Logout | The session was cleared. | Passed |

6. Deactivate Seller account

| 1 | Delete account | The account was deleted from the database. | Passed |

C. ADMIN FEATURES TEST

1. Login as Admin

| 1 | Login | Successfully fetched from database and authenticated | Passed |

2. Deactivate Seller and Customer account

| 1 | Delete account | The account was deleted from the database. | Passed |

3. View Order History

1	View Order	The orders were fetched from the database.	Passed

4. Update admin profile

1	Update profile	Admin details were successfully updated	Passed

5. Logout

1	Logout	The session was cleared.	Passed

6. CONCLUSION

"E-BookStore", is an online shopping application, developed by our project team to simplify the online sale and purchase of books. We tried using the latest technologies that are cross-platform and robust.

Each and every software we used was open-source in nature, which keeps the cost of production at a minimum.

We were also meticulous about the user experience aspect of our application so that navigating our website is an easy and seamless experience.

In conclusion, "E-BookStore" as an application would definitely be a good choice for any book trading business that wishes to enter the online market. We are confident that the numerous features and visually appealing look of the application will certainly give a big boost to the business.

.

7. FUTURE SCOPE

Using whatever we have learnt over the duration of this course, we tried to make our project as user-friendly and gave it as many features as possible in the limited time allotted for the project work. That said, there are certainly more features that can be added to our application. Some of those are mentioned below:

1. Book details can be viewed by hovering on the book.

2. Wish list can be implemented.

3. Customers can add reviews and feedbacks.

4. Additional payment means can be added other than cards.

5. After a confirmed purchase, an email with the invoice of the orders can be sent to the customer.

6. In case the user forgets the password, a "reset password" functionality can be added.

7. An order conformation notification/mail will be sent to the seller.

8. REFERENCES

Following is the list of websites we referred during the course of our project

1. https://getbootstrap.com/docs/5.1/getting-started/introduction/

2. https://reactjs.org/docs/getting-started.html

3. https://reactjs.org/docs/getting-started.html

4. https://www.baeldung.com/

5. https://www.w3schools.com/

6. https://docs.spring.io/springdata/jpa/docs/current/reference/html/#reference

7. https://javadoc.io/doc/org.springframework.data/spring-datajpa/latest/index.html

8. https://developer.mozilla.org/en-US/

9. https://reactstrap.github.io/?path=/story/home-installation--page

10. https://mui.com/material-ui/getting-started/overview/