



BOOK STORE

Project Report

Kumar, Aditya
Holavanahalli Raghavendra Rao, Brinda Rao
Manvi, Srujana
Shenoy, Swathi
Thangapalam, Vaishnavi

TABLE OF CONTENTS

1. PROJECT OVERVIEW	1
2. REQUIREMENT/FEATURE LIST	2
3. DOMAIN DICTIONARY	3
4. ACTOR DICTIONARY	5
5. USE CASE LIST	6
6. USE CASE DIAGRAM	7
7. ANALYSIS CLASS DIAGRAM	8
8.1 SEQUENCE DIAGRAM 1	9
8.2 SEQUENCE DIAGRAM 2	9
8.3 SEQUENCE DIAGRAM 3	10
8.4 SEQUENCE DIAGRAM 4	11
8.5 SEQUENCE DIAGRAM 5	12
9. DESIGN CLASS DIAGRAM	13
10. OBJECT RELATIONAL MODEL	14
11.1 STRATEGY DESIGN PATTERN	14
11.2 OBSERVER DESIGN PATTERN	15
11.3 STATE DESIGN PATTERN	15
11.4 ADAPTER DESIGN PATTERN	16
11.5 SCREENSHOTS	18-73
12 REFERENCES	74

PROJECT OVERVIEW

In this project we have analyzed, designed and implemented the app for a bookstore. Through the app the customer can register, search for books, place order, create wish list and read/write reviews and ratings. The admin through the app can manage the inventory, add books and categories, update orders and customer account. The implementation of the design is done using Java Swings for the user interface and MySQL as the backend database.

Through this app, we have used various design patterns to implement the functionalities that are identified. As part of developing this app, we have been through the entire Software Development Life Cycle of the app, like, we were part of the requirement gathering and analysis, design the app based on the requirements collected, once the design has been decided upon we have carried on to implement the same using Java as the main programming language along with Java Swings for User Interface and MySQL for the database part. Once the implementation was done, we have tested for all the scenario's and debugged and fixed the issues if there were any while doing the testing.

REQUIREMENT AND FEATURE LIST

The key features of the bookstore app are listed below:

User Registration:

- User is given online form to enter details like First Name, Last Name, Date of Birth, Address, Email ID, UserName, Password.
- Guest login option is also provided, where the guest user need not login to the system in order to continue with the processes of either searching books or ordering the books.

Browse/Search Books:

- User can search for the books and can apply few filters like Category, Book Title, Author etc. in order to make the search process easier and quicker.

Order:

- User can select multiple books and add to cart.
- User must specify the shipping address and billing address.
- User can track their order.
- Registered user can create a wish list.
- Total bill amount must be displayed.

Billing:

- Coupons or Discounts may be applied on the order items.
- Payment is validated and confirmed.
- Send the receipt and order details.

Rate & Review Books:

- User can rate and provide reviews of books which he/she has purchased earlier

Feedback to Book store:

- User can provide feedback about their experience of using the app.

Feedback about seller:

- User can provide their feedback about sellers.

Admin Functionality:

- Add or Remove Books to and from catalog
- Add or remove categories
- Add or Remove Discount and Offers
- Verify Reviews for illicit contents
- Update Customer Information
- Cancel order
- Update order
- Inventory Management

Customer Care:

- User can contact customer care in case of any issues like refund, return or any information needed. The support staff can raise an incident ticket to track the issues.

DOMAIN DICTIONARY:

<u>Name</u>	<u>Type</u>	<u>Description</u>
Admin	Role	Updates Customer Account, updates and cancels orders, adds or removes categories, books, discounts and offers, does inventory management and verifies reviews.
Guest User	Role	Places order, checks out the order, views book reviews, returns or replaces order.
Registered Customer	Role	Creates and updates customer account, Places order, cancel order, checks out the order, reviews books, recommends books to a friend, pays bill, creates Wishlist, views suggestions
Customer	Object	The customer's details like CustomerId, name, email ID, Phone number
Banking Agent	Role	Verifies Payment and credit card info.
Inventory Management	Process	Managing the stocks of books. Performed by the admin
Order	Object	The books ordered by the customer
Place Order	Process	Placing an order of books. Performed by Customer
Cancel Order	Process	Cancelling an order of books. Performed by Customer
Cancel Order	Process	Cancelling an order of books. Performed by Admin
Update Order	Process	Updating an order of books. Performed by Admin
Order Details	Object	The details of the order like orderId, product details, date of placing order, total bill order, shipment address etc.
Track Order	Process	Track the order once it has been shipped.
Send Order Details	Process	The details of the order are sent to the customer
User Registration	Function	The act of user getting registered in the system
Book	Object	Book details like title, author, edition, book ID, price, reviews, seller, genre etc.
Selling Books	Function	The act of providing a collection of books available for purchase.
Billing	Function	The act of calculating the total bill amount for the order
Add to cart	Process	Adding the selected books to cart
Checkout	Process	Proceeding with the order to billing and shipping
Shipping	Process	Shipping the order to the customer
Review	Object	Review of books given by customer
Feedback	Object	Feedback about the shopping experience, seller

Rating	Process	Rating the book
Rating	Object	Rating given to the book
Update Account by Customer	Process	The user's account details are updated. Performed by Customer
Update Account by Admin	Process	The user's details are updated. Performed by Admin
Payment Validation	Process	The payment is validated by checking the credit card info. Performed by Banking Agent
Search Books	Process	Browse the various categories for books
Apply Discount	Process	Apply coupons or seasonal discounts on books
Discount	Object	The discount amount applied to the bill
Return or Replace	Process	The books in the order are returned or replaced
Customer Care Service	Process	The customer is provided with any help or details as requested. Performed by the Support Technicians
Support Technician	Role	Provides customer any help or information requested
Add Category	Process	A new category of books is added. Performed by Admin
Remove Category	Process	An existing category of books is removed. Performed by Admin
Category	Object	The category of books added or removed.
Add Book	Process	A new book is added. Performed by Admin
Remove Book	Process	An existing book is removed. Performed by Admin
Book	Object	The book added or removed
Add Offers	Process	New offers are added for the customers. Performed by Admin
Remove Offers	Process	Existing offers are removed. Performed by Admin
Offer	Object	The new offer added or removed
Log out	Process	All order information is saved to DB and session is ended. Performed by Customer
Log out	Function	The act of logging out of the app
Verification of Reviews	Process	The reviews are verified for authentication and checked for illicit content.
Review	Function	The act of checking the review given by customer
Create Wishlist	Process	The books which customer likes but may be purchased in future and not in current order. Performed by the customer
Wishlist	Object	The list of books saved by customer for future purchase
Wishlist	Function	The act of creating a wish list
Display Suggestions	Process	The books which the user may like based on his current selection of books.

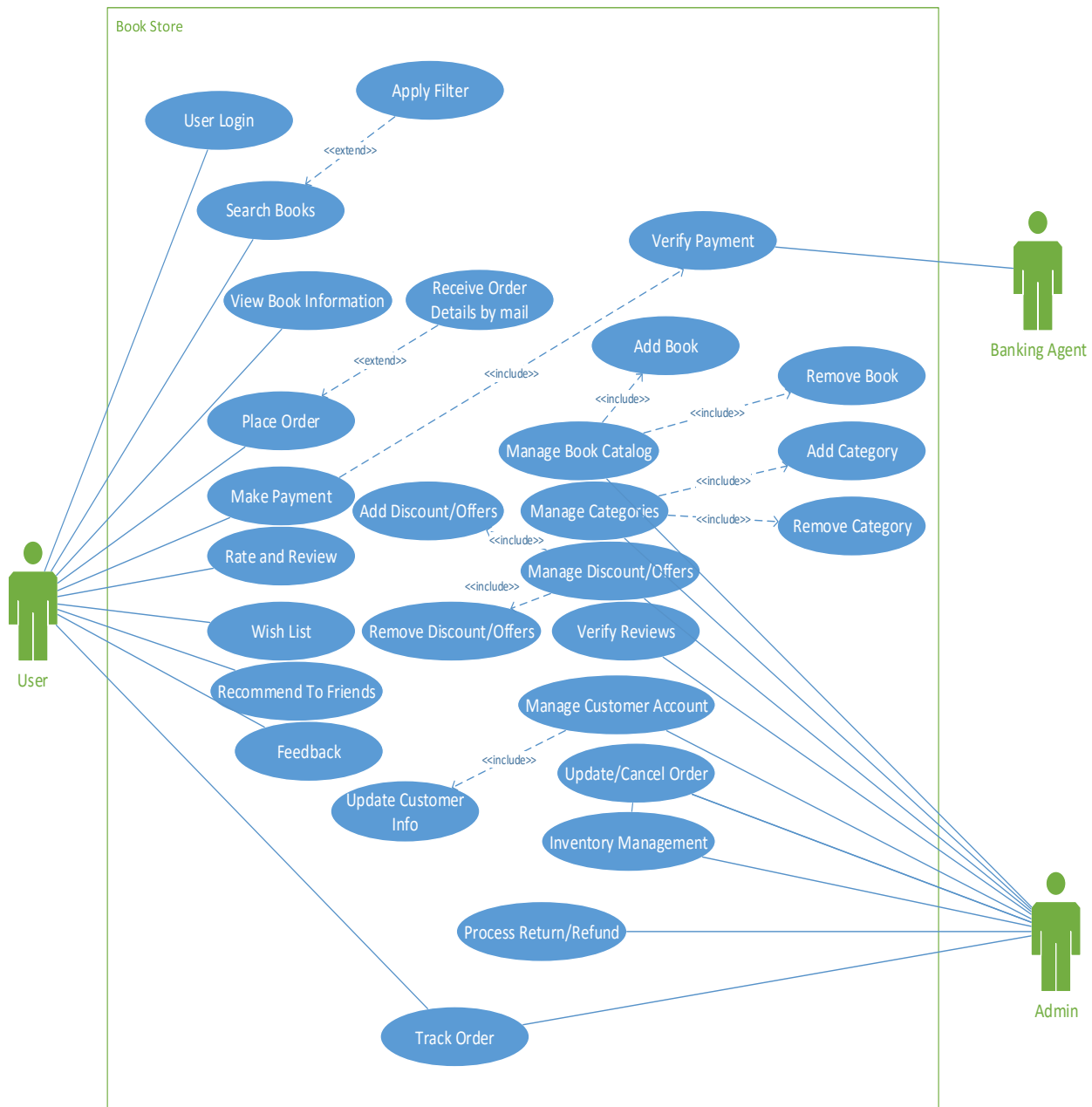
ACTOR DICTIONARY:

<u>Actor</u>	<u>Description</u>	<u>Abstract</u>	<u>Use Case</u>
Admin	Updates Customer Account, updates and cancels orders, adds or removes categories, books, discounts and offers, does inventory management and verifies reviews.		Manage Book Catalog Manage Categories Manage Discount/Offer Verify Reviews Manage Customer Account Update Order Cancel Order Inventory Management Process Return/Refund Track Order
User	The customer who places order for books	Yes	Search Books View Book Information Make Payment Track Order Place Order
Guest User	Places order, checks out the order, views book reviews, returns or replaces order.		Search Books View Book Information Place Order Make Payment Track Order
Registered Customer	Creates and updates customer account, Places order, cancel order, checks out the order, reviews books, recommends books to a friend, pays bill, creates Wishlist, views suggestions		User Login Search Books View Book Information Place Order Make Payment Rate and Review Track Order Wish List Recommend Book to Friends Feedback
Banking Agent	Verifies Payment and credit card info.		Validate Payment
Support Technician	Provides customer any help or information requested		Support Customer Log Incident Ticket

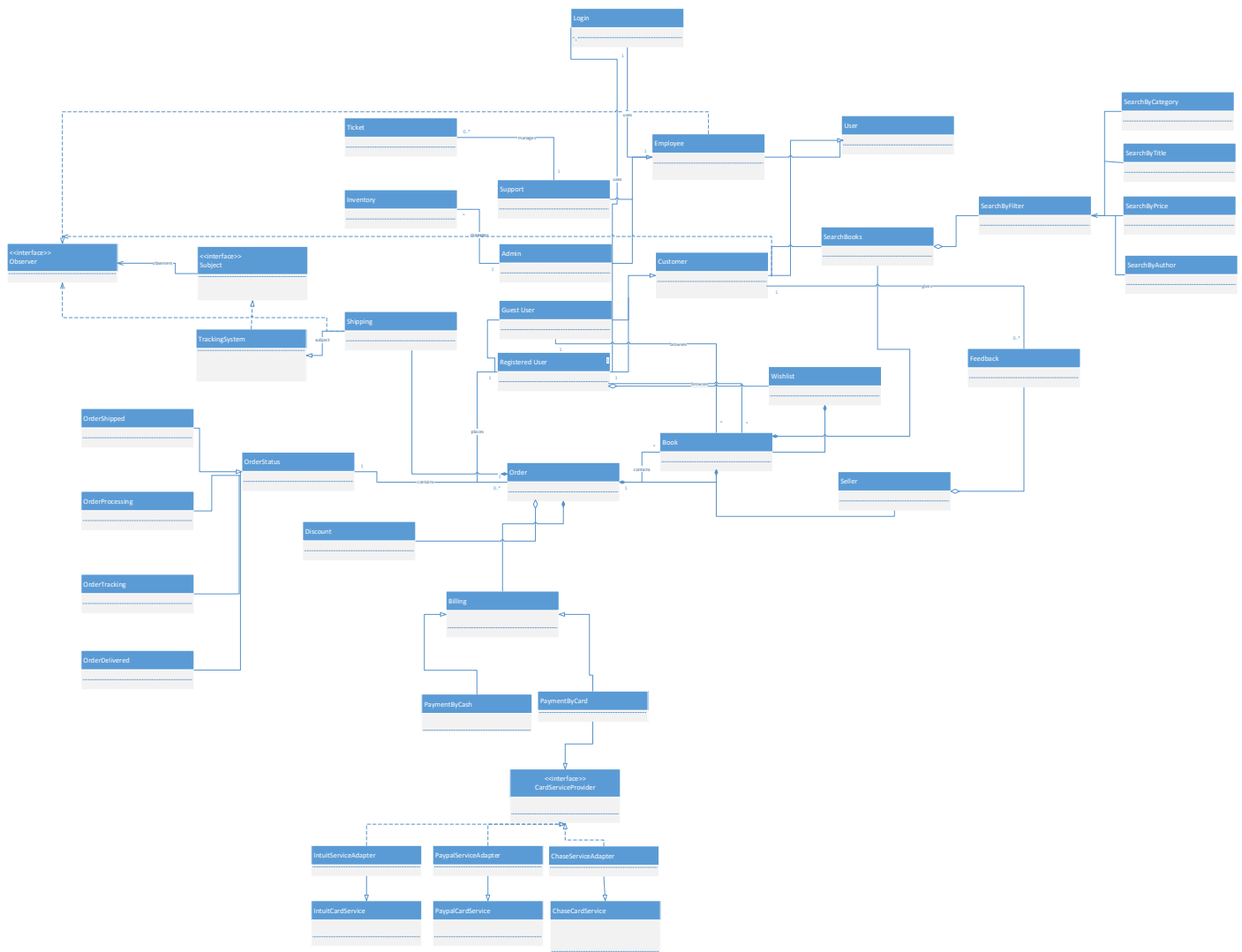
USE CASES

- User Login
- Search Books
- View Book Information
- Place Order
- Make Payment
- Rate and Review
- Create Wishlist
- Recommend to Friends
- Feedback
- Track Order
- Apply Filter
- Receive Order Details by Mail
- Add Discount or Offers
- Remove Discount or Offers
- Update Customer Information
- Verify Payment
- Add Book
- Manage Book Catalog
- Manage Categories
- Manage Discount or Offers
- Verify Reviews
- Manage Customer Account
- Update or Cancel Order
- Inventory Management
- Process Return or Refund
- Remove Book
- Add Category
- Remove Category

USE CASE DIAGRAM:

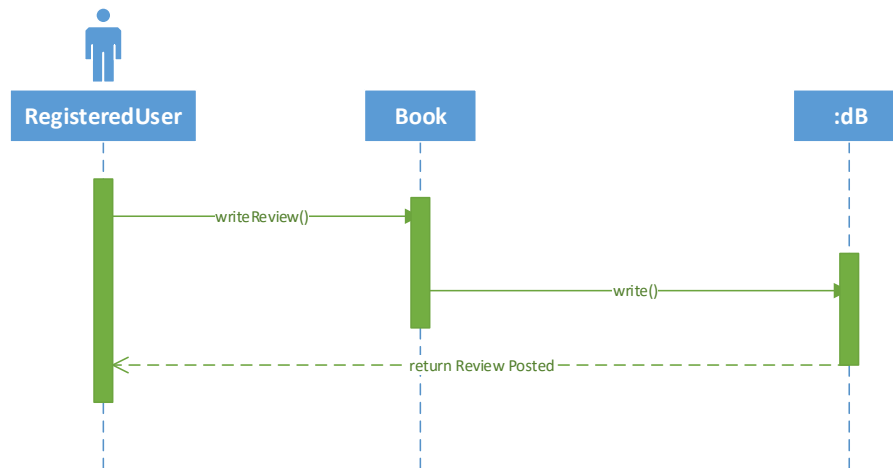


ANALYSIS DIAGRAM:

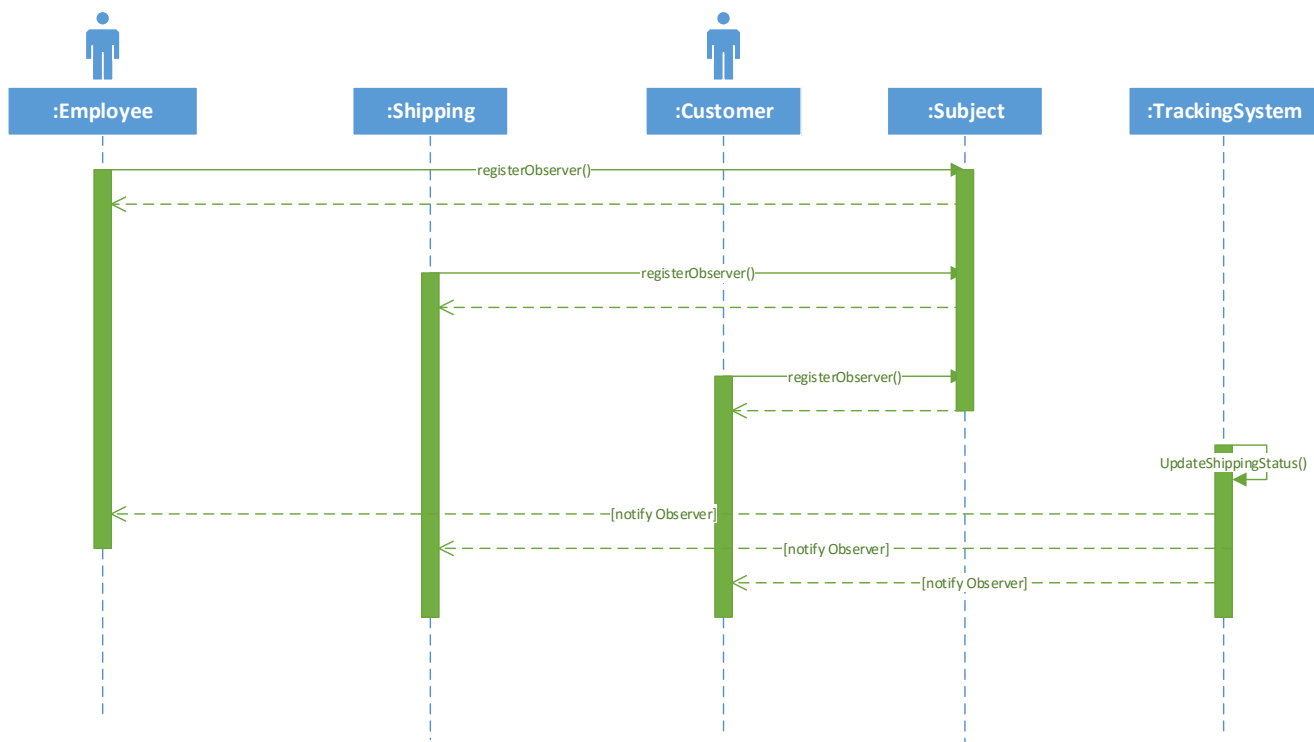


Sequence Diagrams

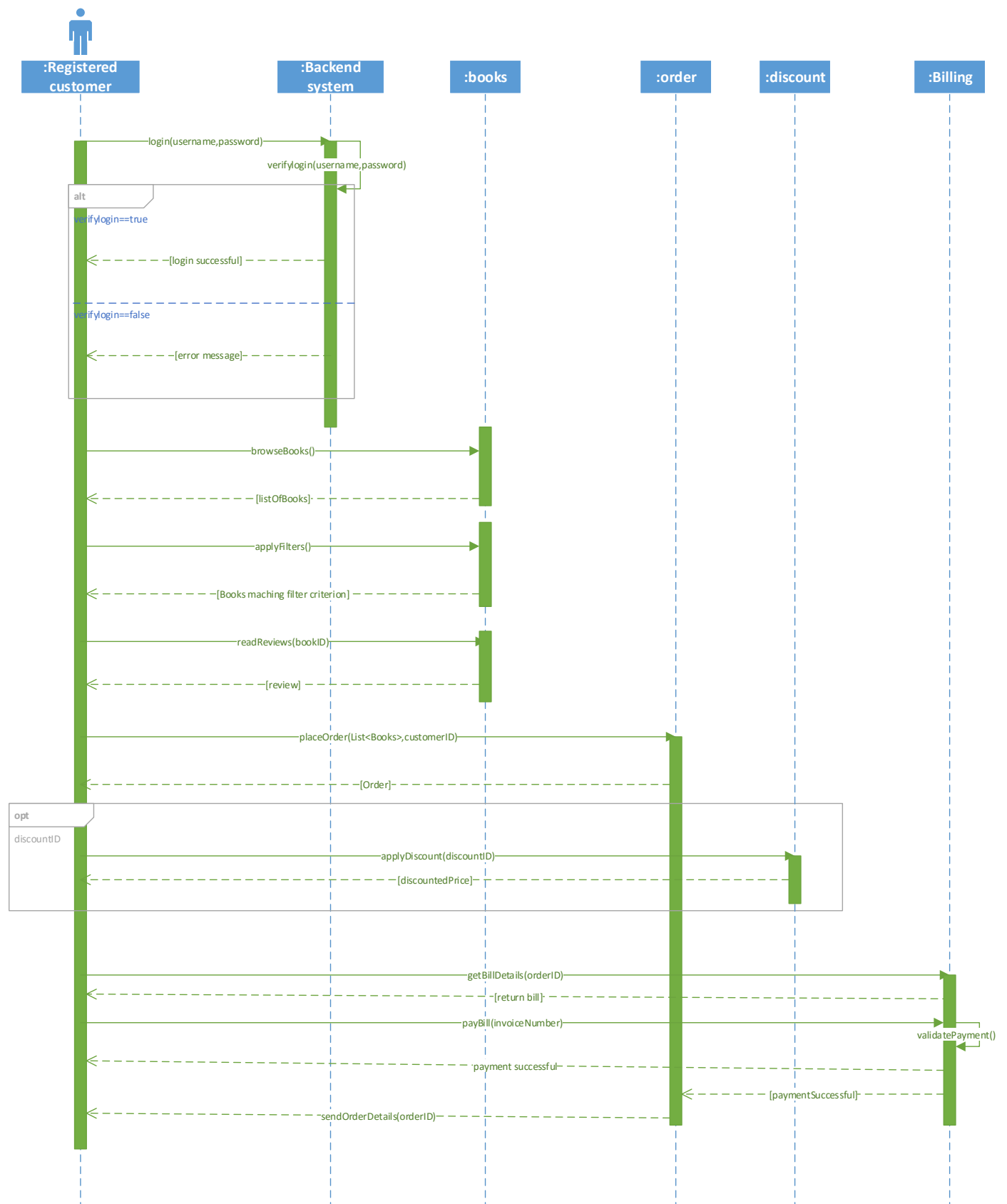
1.1 Sequence Diagram to Write Review :



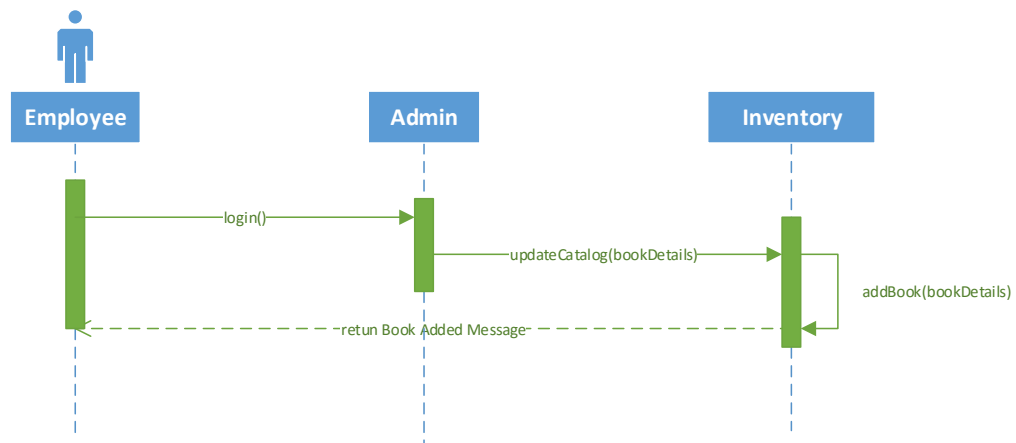
1.2 Sequence diagram to track shipping details



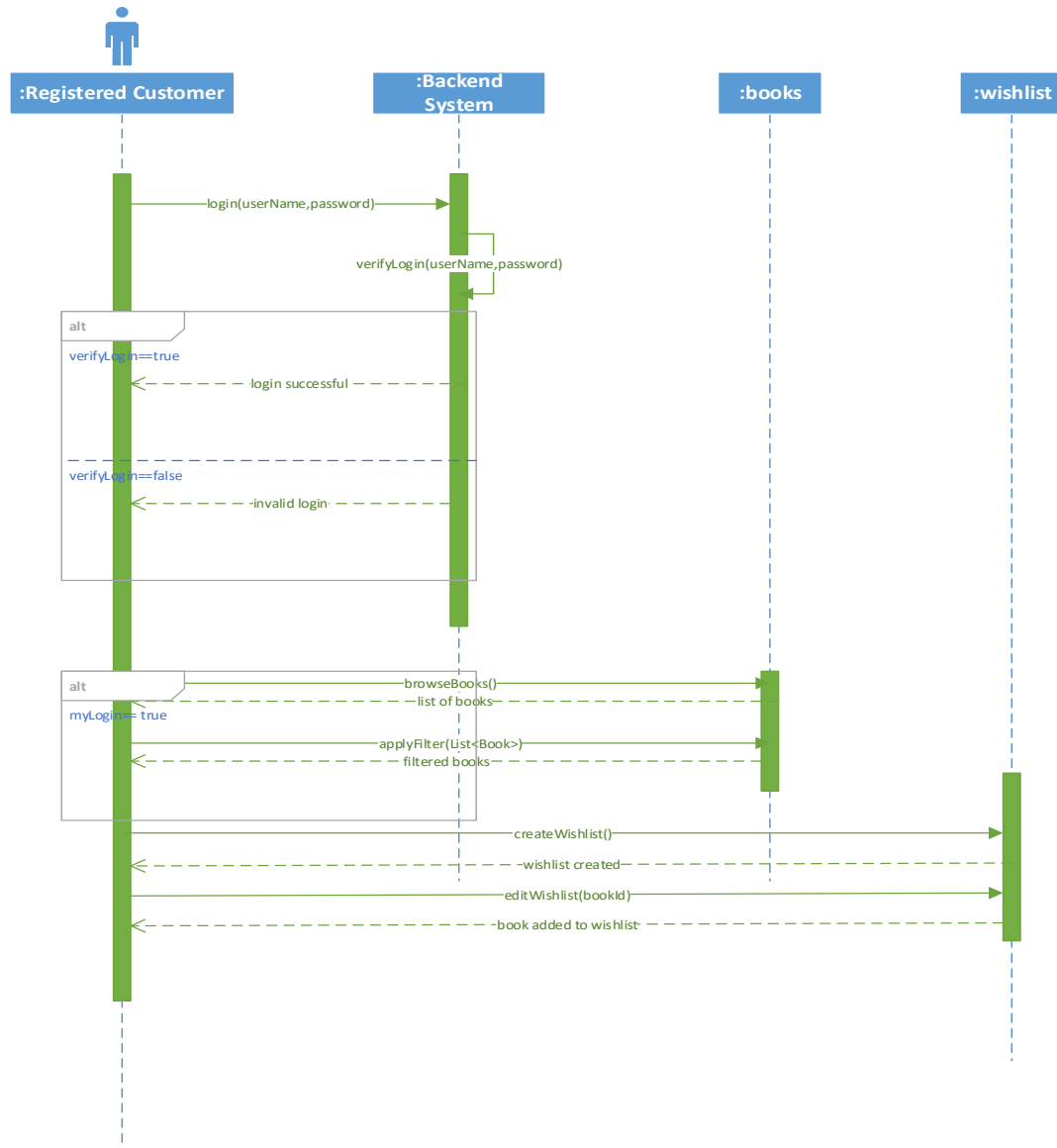
1.3 Sequence diagram to place an order



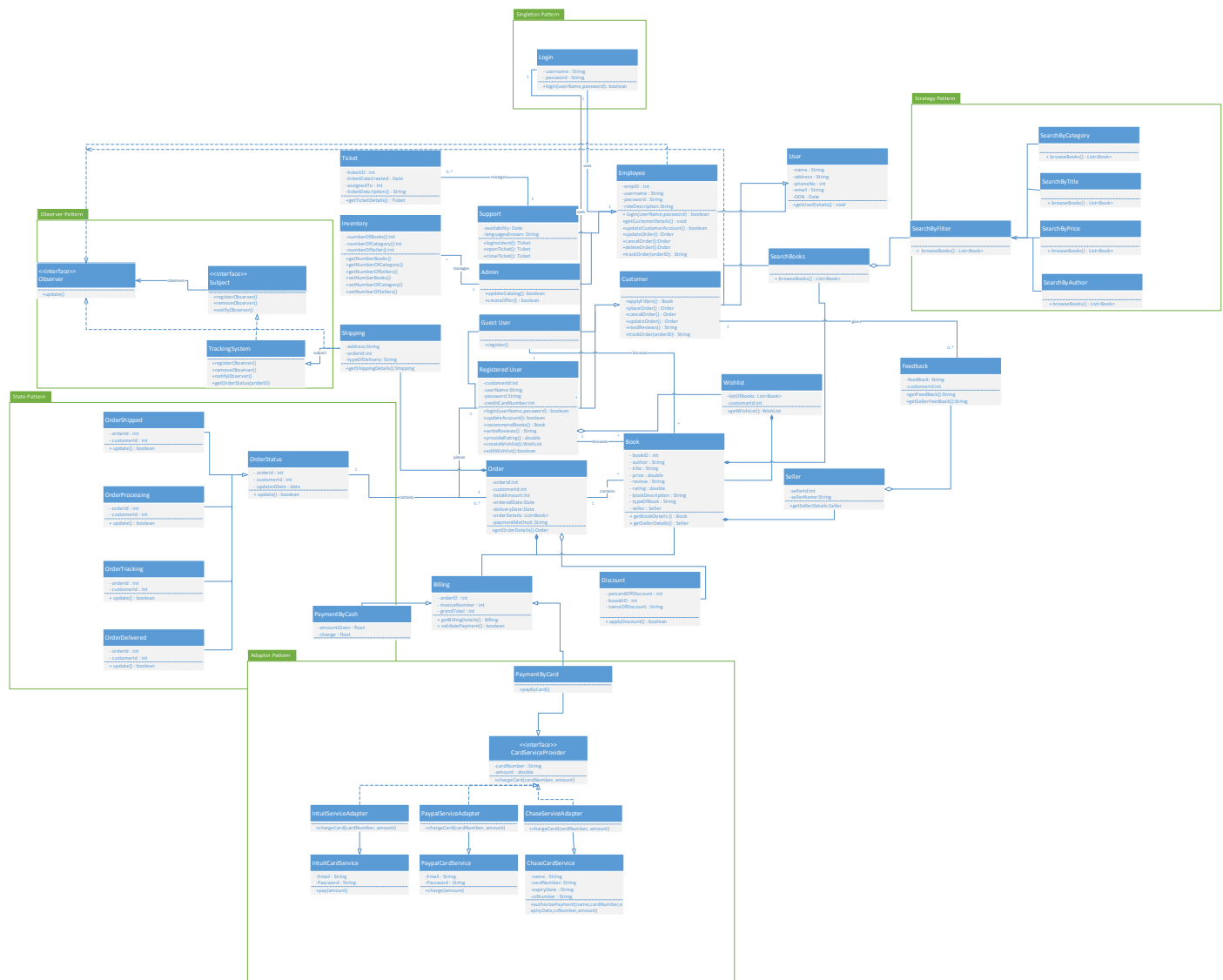
1.4 Sequence diagram to manage inventory



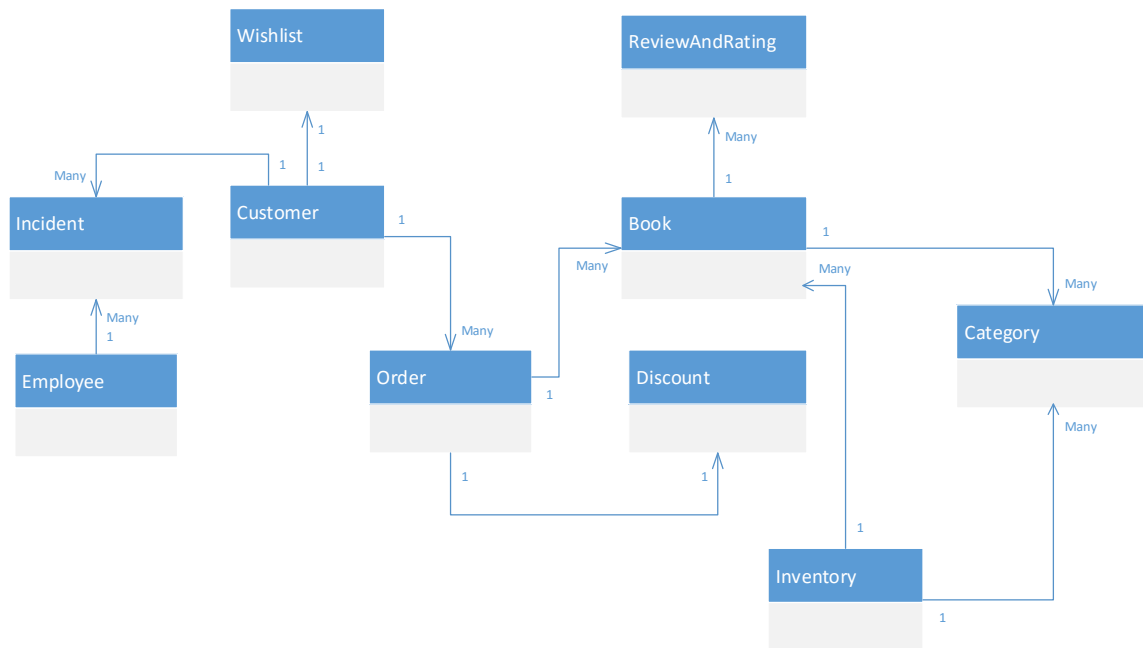
1.5 Sequence Diagram to Create Wish List



Design Class Diagram:

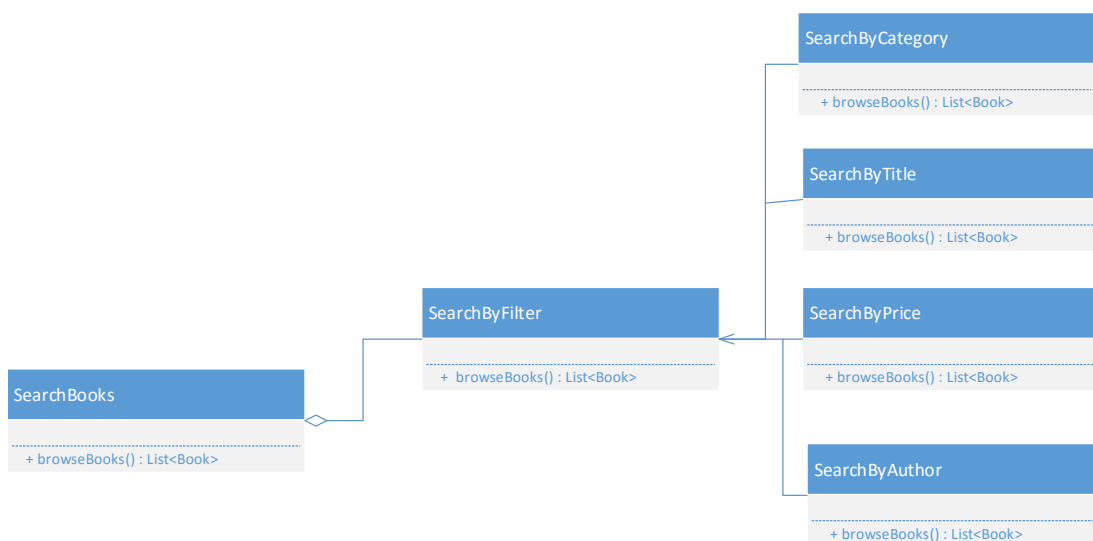


Object Relational Model:



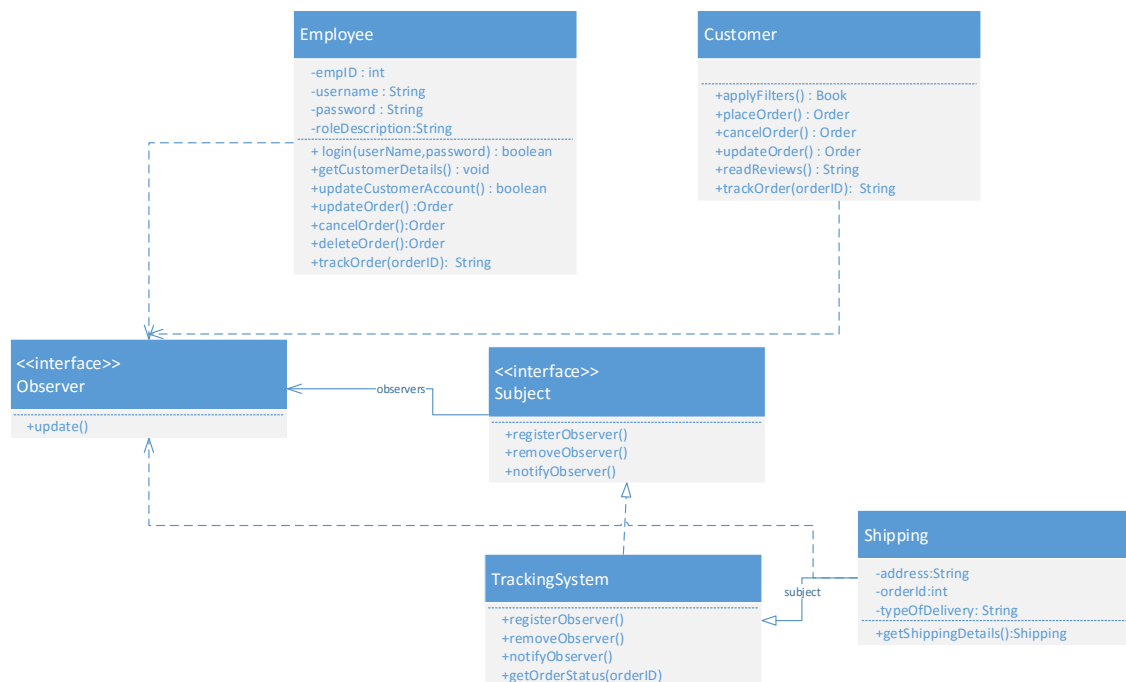
Design Pattern:

Strategy Design Pattern:



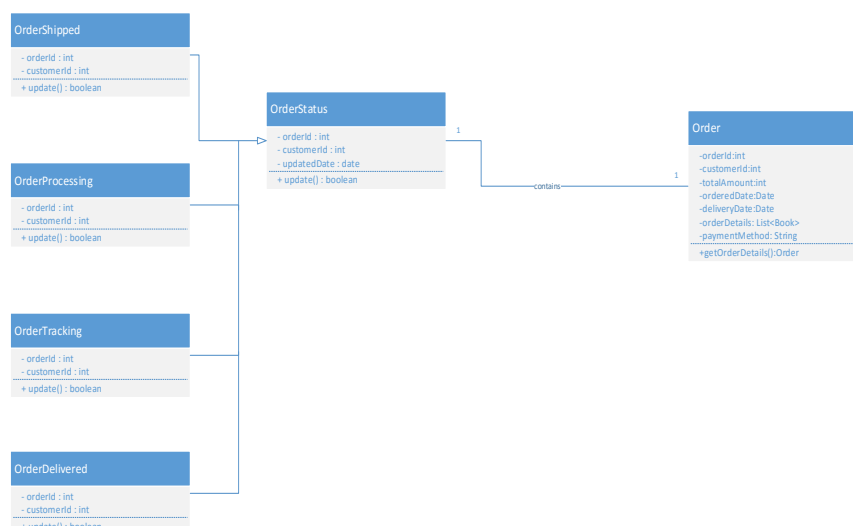
In this project, ways of applying filter to search books is a strategy. We can search books by applying filter and which indeed can be extracted based on category, title, price or author. Hence we have used strategy design pattern to demonstrate this. Here, SearchByCategory, SearchByTitle, SearchByPrice and SearchByAuthor are different ways of filtering in order to search books.

Observer Design Pattern:



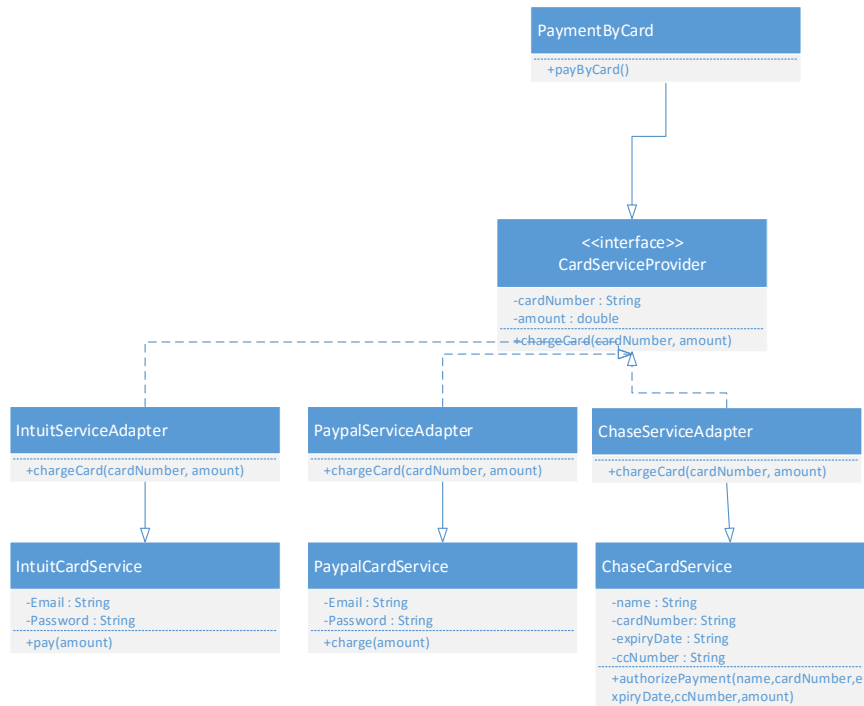
Here, when one object changes, all its dependents changes state. Objects have one-to-many dependency. When the shipping status changes, the registered observers i.e., the customer, employee and shipping are notified of the change so that they can perform required actions. Hence, we have used observer design pattern.

State Design Pattern:



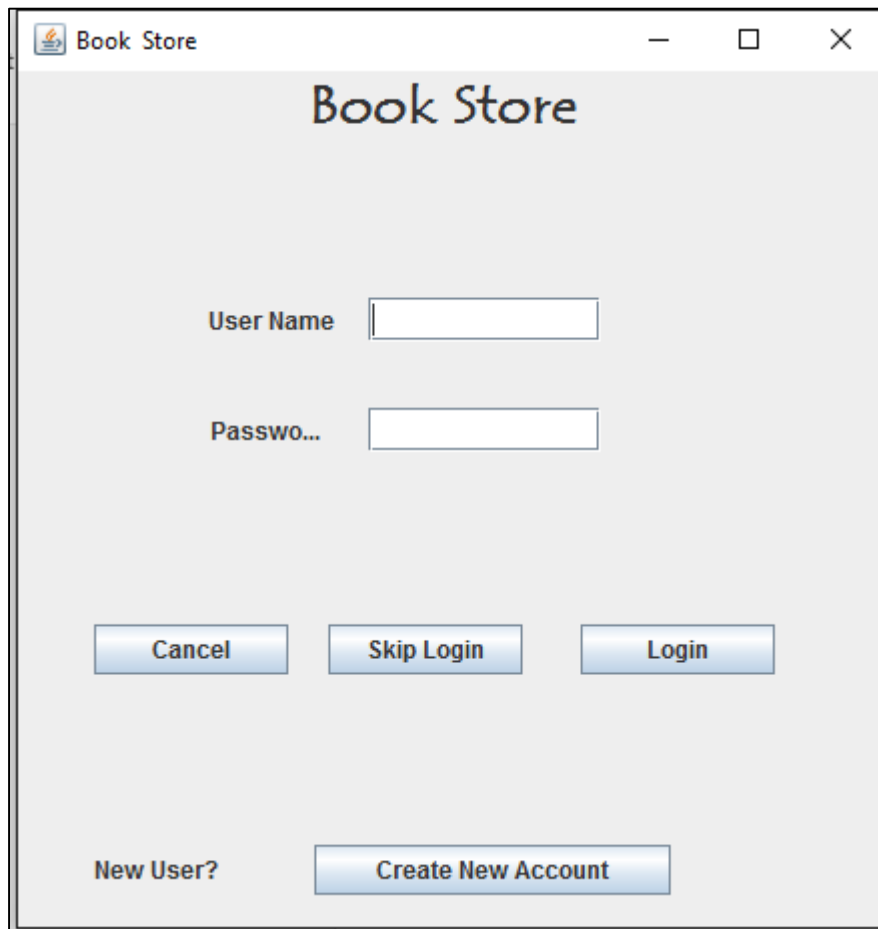
This pattern allows an object to change its behavior when its internal state changes. We are using this pattern to get the status of the order. The order status can have possible states such as processing, delivered or shipped. As the internal state of the order changes, the behavior of the object Order Status changes.

Adapter Design Pattern:



Here, **CardServiceProvider** is an interface which acts as a super type for various service providers as they have different ways of validating payment. Hence, we have used adapter design pattern.

LOGIN SCREEN



The image shows a login window titled "Book Store". It features a title bar with a small icon, the text "Book Store", and standard window controls (minimize, maximize, close). The main content area has a light gray background. At the top, the text "Book Store" is displayed in a large, dark font. Below this, there are two input fields: "User Name" and "Passwo...". Each field is preceded by its label and followed by a white rectangular input box with a thin border. Underneath the input fields, there are three buttons: "Cancel", "Skip Login", and "Login". These buttons are light blue with a slight gradient and a thin border. At the bottom left, the text "New User?" is displayed. To its right is a button labeled "Create New Account", which is also light blue with a slight gradient and a thin border.

Book Store

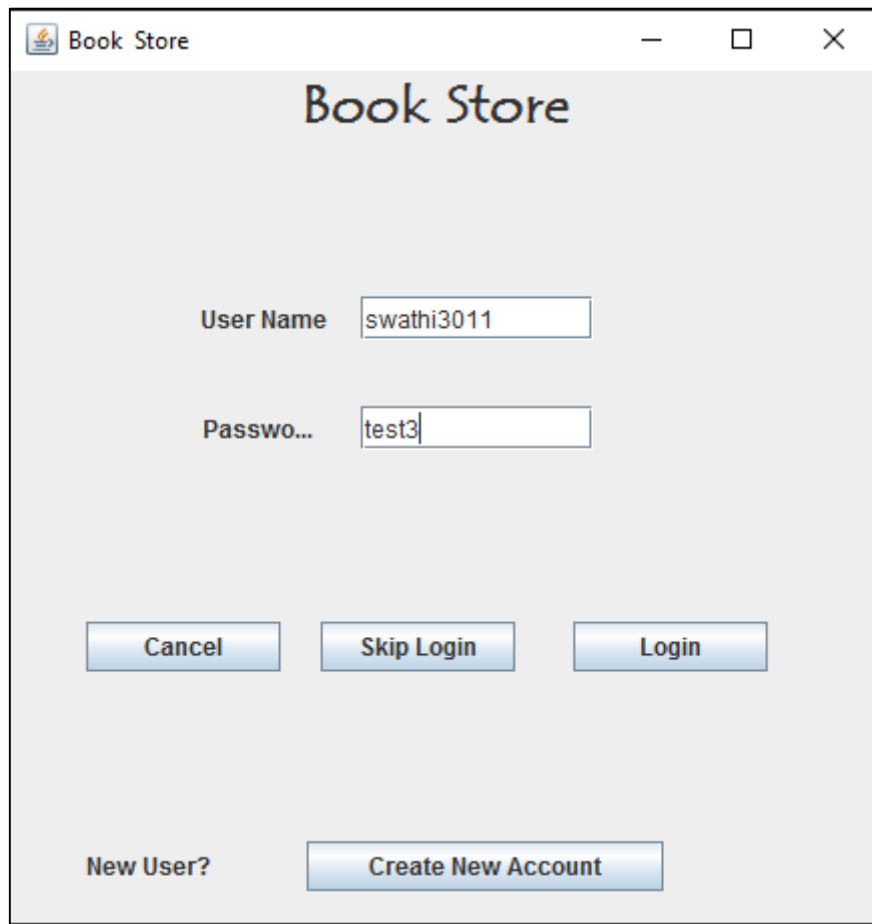
User Name

Passwo...

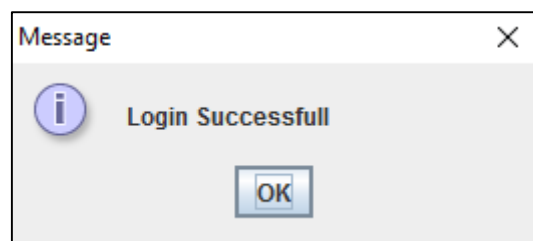
Cancel Skip Login Login

New User?

Enter the user name and password and click on Login button



The image shows a window titled "Book Store" with a standard Windows-style title bar (minimize, maximize, close buttons). The window has a light gray background. At the top, the text "Book Store" is displayed in a large, dark font. Below this, there are two input fields. The first is labeled "User Name" and contains the text "swathi3011". The second is labeled "Passwo..." (truncated) and contains the text "test3". Below the input fields, there are three buttons: "Cancel", "Skip Login", and "Login". At the bottom left, there is a label "New User?" and a button "Create New Account".



DB Screenshot:

1 • `select * from book_store.customer`

CustomerID	FirstName	LastName	DOB	EmailID	Address	PhoneNumber	UserName	Password
1	sruj	manvi	1993-12-11	srujana.manvi@gmail.com	2901 S King Dr	7911119	ssss	test1
3	Vaishnavi	Thangapalam	1994-12-12	vt@gmail.com	2988 SKD	790119	sm	test2
363	Swathi	Shenoy	1990-01-30	swathi.shenoy3011@gmail.com	2901 S King Drive	3125369054	swathi3011	test3

Click on Skip Login:

Book Store

Book Store

User Name

Passwo...

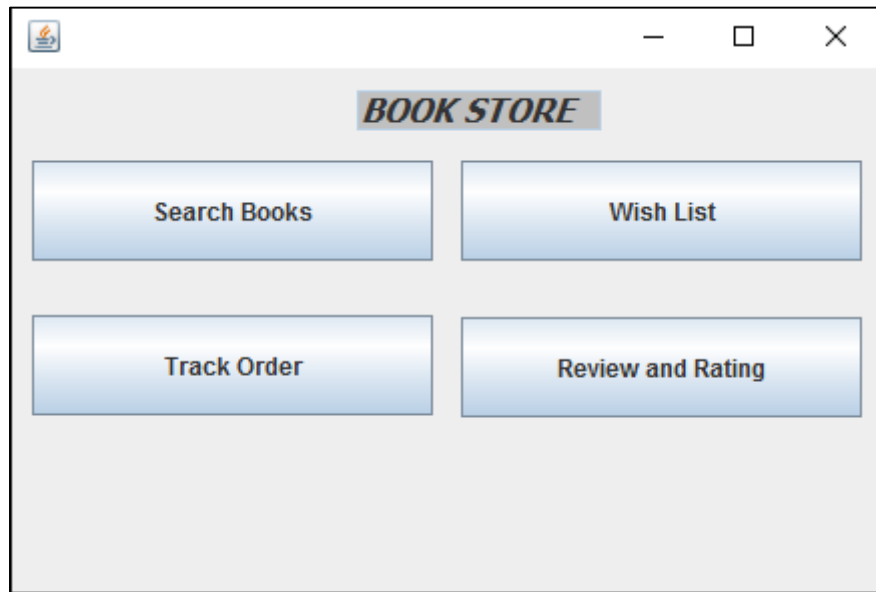
Cancel

Skip Login

Login

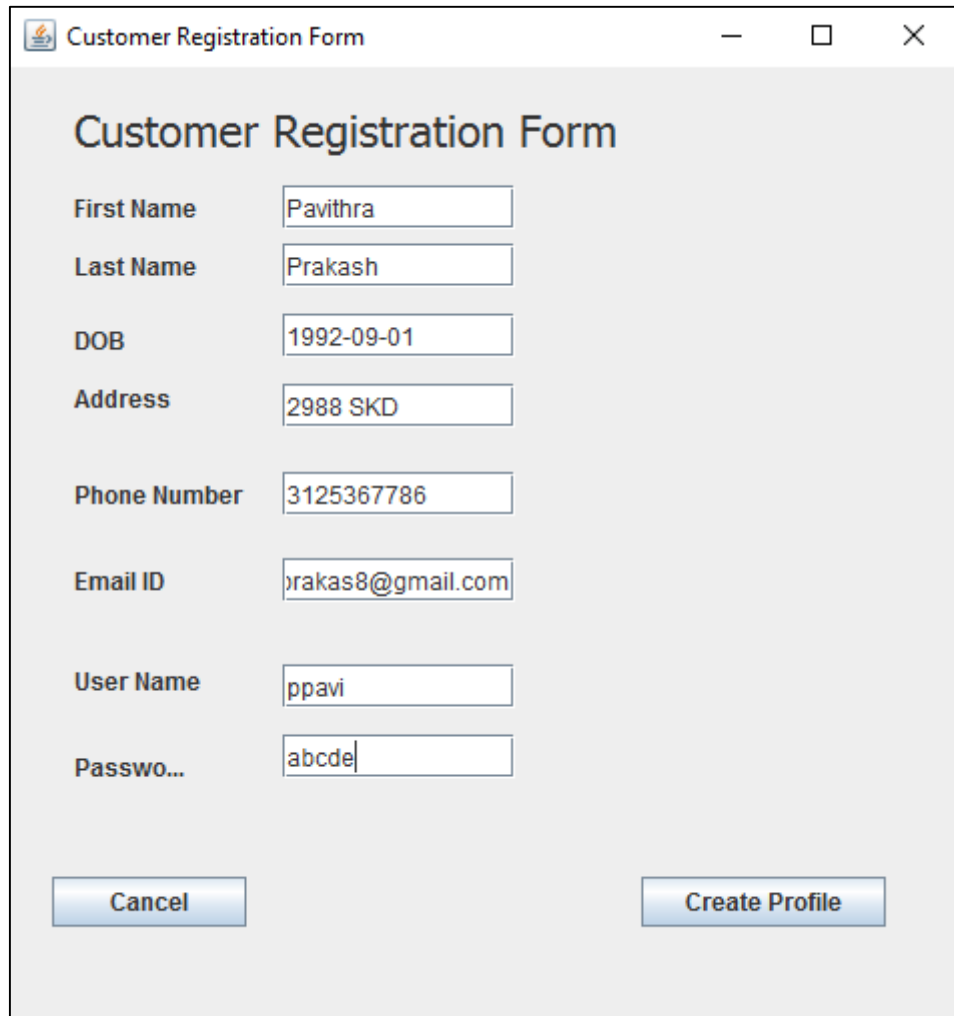
New User?

Create New Account



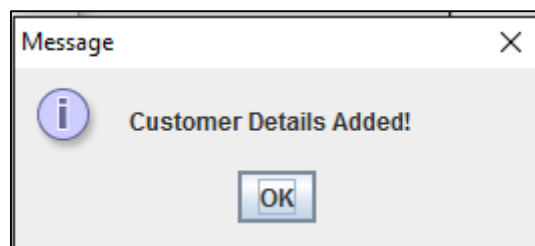
Click on Create New Account

The screenshot shows a window titled "Customer Registration Form" with a light gray background. The title "Customer Registration Form" is displayed in a bold font. Below the title, there are eight input fields with labels to their left: "First Name", "Last Name", "DOB", "Address", "Phone Number", "Email ID", "User Name", and "Passwo...". Each input field is a white rectangle with a thin border. At the bottom of the form, there are two blue buttons with white text: "Cancel" on the left and "Create Profile" on the right. The window has a standard Windows-style title bar with a minimize button, a maximize button, and a close button.



A screenshot of a Windows-style application window titled "Customer Registration Form". The window has a standard title bar with minimize, maximize, and close buttons. The form itself is light gray and contains several labeled text input fields arranged vertically. The labels are "First Name", "Last Name", "DOB", "Address", "Phone Number", "Email ID", "User Name", and "Passwo...". The corresponding input values are "Pavithra", "Prakash", "1992-09-01", "2988 SKD", "3125367786", "prakas8@gmail.com", "ppavi", and "abcde". At the bottom of the form, there are two buttons: "Cancel" on the left and "Create Profile" on the right.

Field	Value
First Name	Pavithra
Last Name	Prakash
DOB	1992-09-01
Address	2988 SKD
Phone Number	3125367786
Email ID	prakas8@gmail.com
User Name	ppavi
Passwo...	abcde



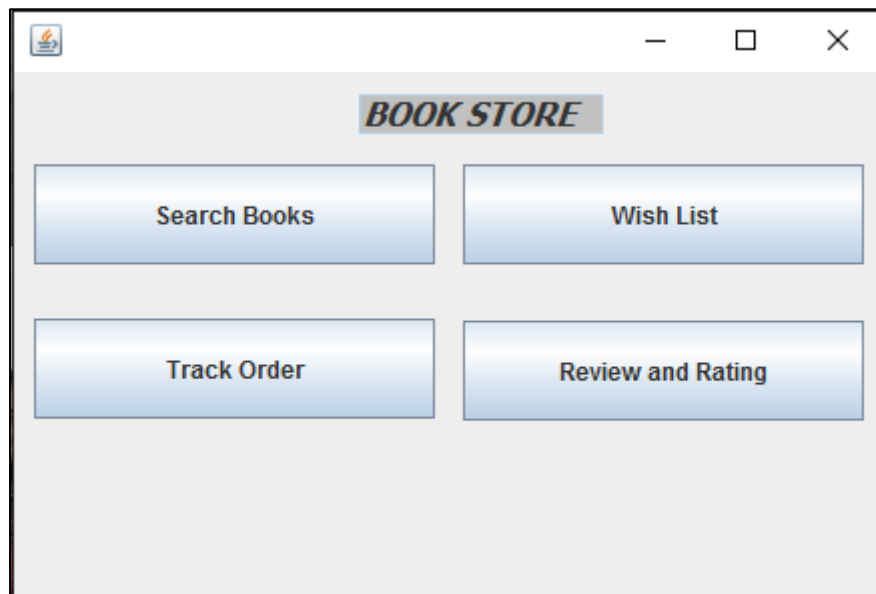
1 • `select * from book_store.customer`

<

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content: [I](#)

	CustomerID	FirstName	LastName	DOB	EmailID	Address	PhoneNumber	UserName	Password
	1	sruj	manvi	1993-12-11	srujana.manvi@gmail.com	2901 S King Dr	7911119	ssss	test1
	3	Vaishnavi	Thangapalam	1994-12-12	vt@gmail.com	2988 SKD	790119	sm	test2
	363	Swathi	Shenoy	1990-01-30	swathi.shenoy3011@gmail.com	2901 S King Drive	3125369054	swathi3011	test3
▶	9119	Pavithra	Prakash	1992-01-01	pprakas8@gmail.com	2988 SKD	3125367786	ppavi	abcde

After clicking OK in the dialog box the below screen is displayed:



CS521 – Book Store Project Report

Click on Search Books:

Search For Books

Search By Filter

Book Title

Book Price

Author

Category

Click on the search button to display all the books:

Search For Books

Search By Filter

Book Title

Book Price

Author

Category

Book ID	Book Title	Author Name	Price	Seller ID	Category
11	intro to algo	CLRS	60	101	Science
22	OSDI	Collmen	65	102	Science
33	cloud computing	raicu	70	103	Science
44	fifty shades of grey	grey	20	104	Romance
55	secret wishlist	preeti shenoy	25	105	Romance
Book1	Jungle Book	Rudyard Kipling	45	Sell1	Fiction

1
✕
select * from book_store.book

Result Grid

Filter Rows:

Edit:

Export/Import:

	BookID	BookTitle	AuthorName	Price	SellerId	Category
▶	11	intro to algo	CLRS	60	101	Science
	22	OSDI	Collmen	65	102	Science
	33	cloud computing	raicu	70	103	Science
	44	fifty shades of grey	grey	20	104	Romance
	55	secret wishlist	preeti shenoy	25	105	Romance
	Book1	Jungle Book	Rudyard Kipling	45	Sell 1	Fiction

Enter any of the search criteria to filter the books:

Search Books

Search For Books

Search By Filter

Book Title

Book Price

Author

Category

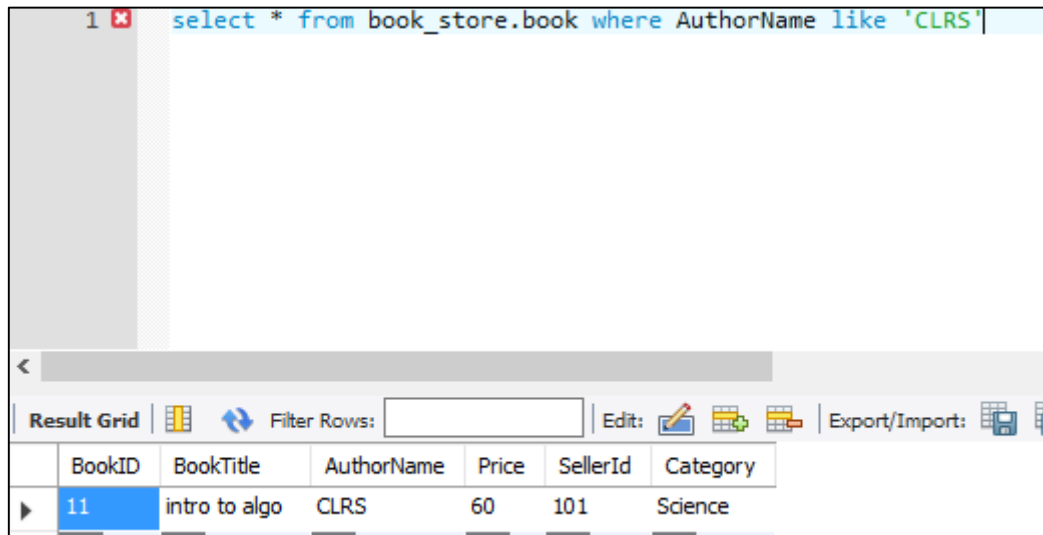
CLRS

Search

Place Order

Book ID	Book Title	Author Name	Price	Seller ID	Category
11	intro to algo	CLRS	60	101	Science

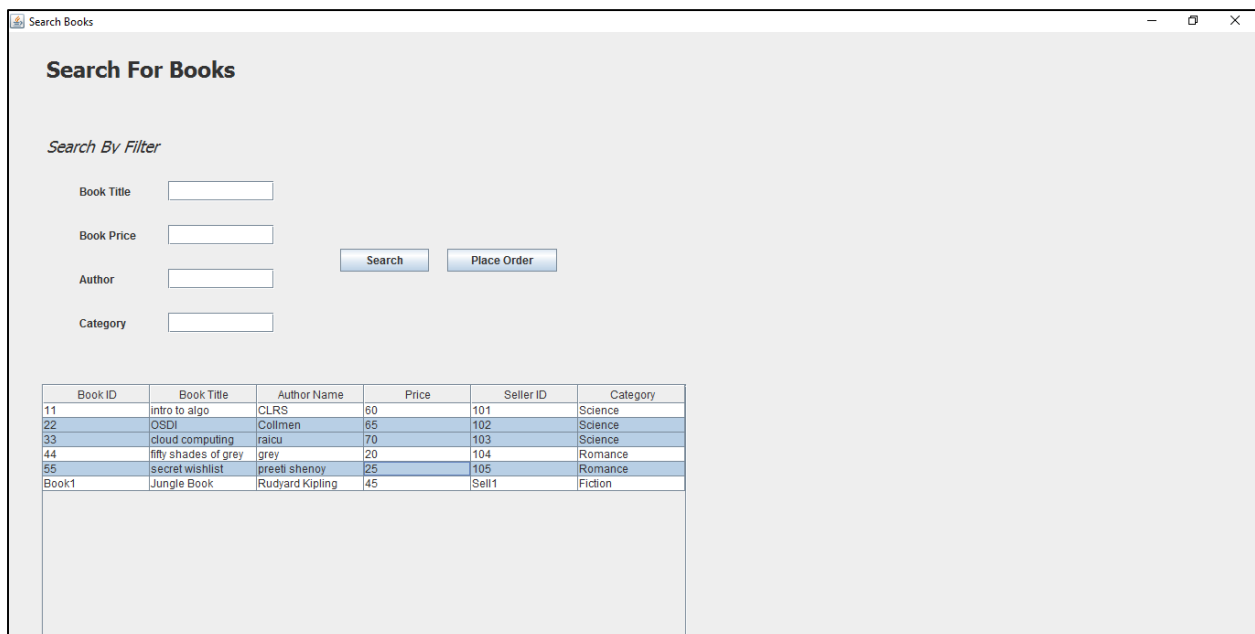
CS521 – Book Store Project Report



The screenshot shows a database query interface. At the top, a SQL query is entered: `select * from book_store.book where AuthorName like 'CLRS'`. Below the query, a "Result Grid" displays the results. The grid has columns: BookID, BookTitle, AuthorName, Price, SellerId, and Category. One row is highlighted, showing BookID 11, BookTitle "intro to algo", AuthorName "CLRS", Price 60, SellerId 101, and Category "Science".

BookID	BookTitle	AuthorName	Price	SellerId	Category
11	intro to algo	CLRS	60	101	Science

Select the books you want to purchase and click on place order to place the order:



The screenshot shows a web application titled "Search Books". It has a section "Search By Filter" with input fields for Book Title, Book Price, Author, and Category. There are "Search" and "Place Order" buttons. Below the filters is a table listing books with columns: Book ID, Book Title, Author Name, Price, Seller ID, and Category.

Book ID	Book Title	Author Name	Price	Seller ID	Category
11	intro to algo	CLRS	60	101	Science
22	OSDI	Collmen	65	102	Science
33	cloud computing	raicu	70	103	Science
44	fifty shades of grey	grey	20	104	Romance
55	secret wishlist	preeti shenoy	25	105	Romance
Book1	Jungle Book	Rudyard Kipling	45	Sell1	Fiction

After clicking Place Order. It displays all orders placed so far and the current books selected. Below is the snapshot for another user:

orderID	quantity	BookID	OrderPlac...	customerID	price	orderStatus
ORD180	1	11	2016-06-30	3	2000	Processing
ORD21	1	22	2016-06-30	3	2000	Processing
ORD221	6	55	2016-06-30	3	2000	Processed
ORD273	5	44	2016-06-30	3	2000	Processing
ORD291	4	33	2016-06-29	3	2000	Processed
ORD631	44	Book1	2016-06-29	3	2000	Processed
ORD641	4	22	2016-06-30	3	2000	Processing
ORD660	5	11	2016-06-30	3	2000	Processing
ORD671	6	22	2016-06-30	3	2000	Processing
ORD751	5	55	2016-06-30	3	2000	Processed
ORD933	1	44	2016-06-30	3	2000	Processing

Number of Books:	Enter the number of books
Update Quantity	Checkout

This is the order screen for the user who selected the books previously:

orderID	quantity	BookID	OrderPlac...	customerID	price	orderStatus
ORD31	1	22	2016-06-30	363	2000	Processing
ORD54	1	55	2016-06-30	363	2000	Processing
ORD922	1	33	2016-06-30	363	2000	Processing

Number of Books:		Enter the number of books
Update Quantity		Checkout

1 `select * from book_store.order where customerID like '363'`

Result Grid Filter Rows: Edit: Export/Import:

	orderID	quantity	BookID	OrderPlacedDate	customerID	price	orderStatus
	ORD304	1	55	2016-06-30	363	2000	Processing
	ORD31	1	22	2016-06-30	363	2000	Processing
	ORD54	1	55	2016-06-30	363	2000	Processing
	ORD842	1	33	2016-06-30	363	2000	Processing
	ORD891	1	22	2016-06-30	363	2000	Processing
	ORD922	2	33	2016-06-30	363	2000	Processing

Update the quantity as required by selecting particular row and then entering the value in the textbox. Then click on update quantity button:

orderID	quantity	BookID	OrderPlac...	customerID	price	orderStatus
ORD31	1	22	2016-06-30	363	2000	Processing
ORD54	1	55	2016-06-30	363	2000	Processing
ORD922	1	33	2016-06-30	363	2000	Processing

Number of Books:

Update Quantity **Checkout**

orderID	quantity	BookID	OrderPlac...	customerID	price	orderStatus
ORD31	1	22	2016-06-30	363	2000	Processing
ORD54	1	55	2016-06-30	363	2000	Processing
ORD922	2	33	2016-06-30	363	2000	Processing

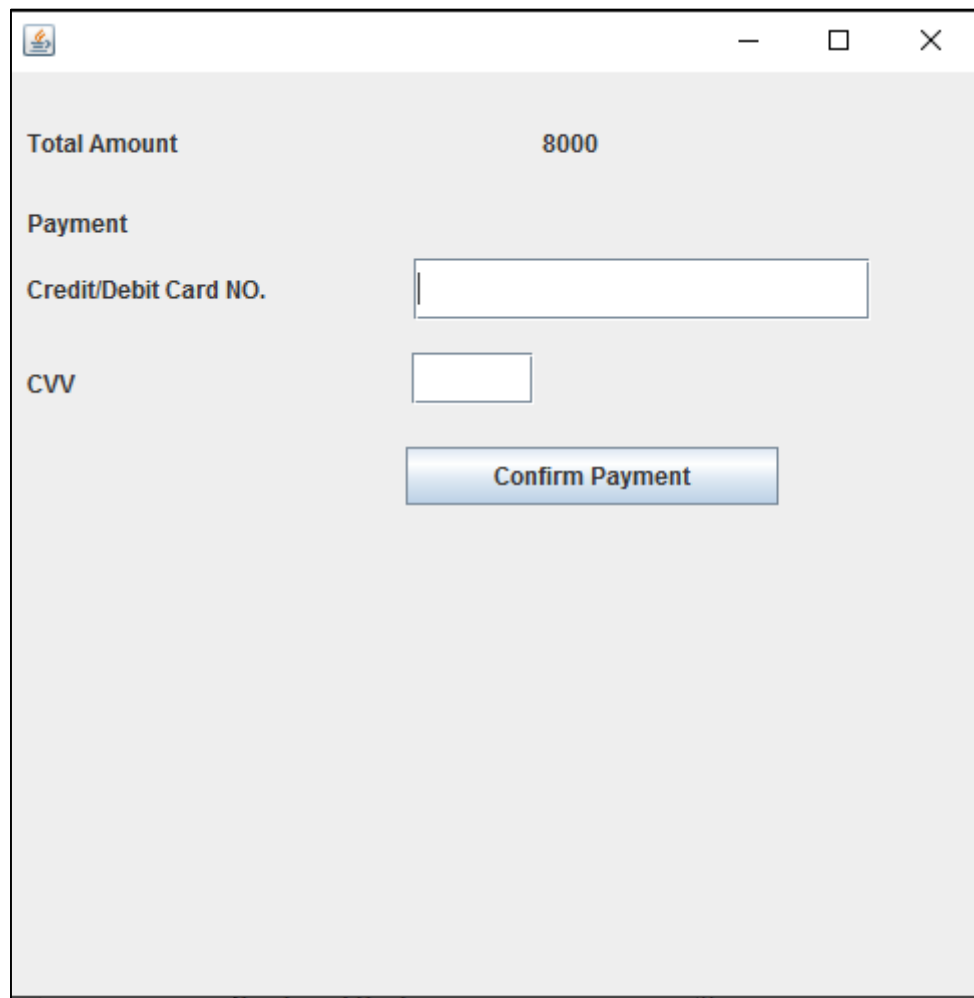
Number of Books:			2
Update Quantity		Checkout	

CS521 – Book Store Project Report

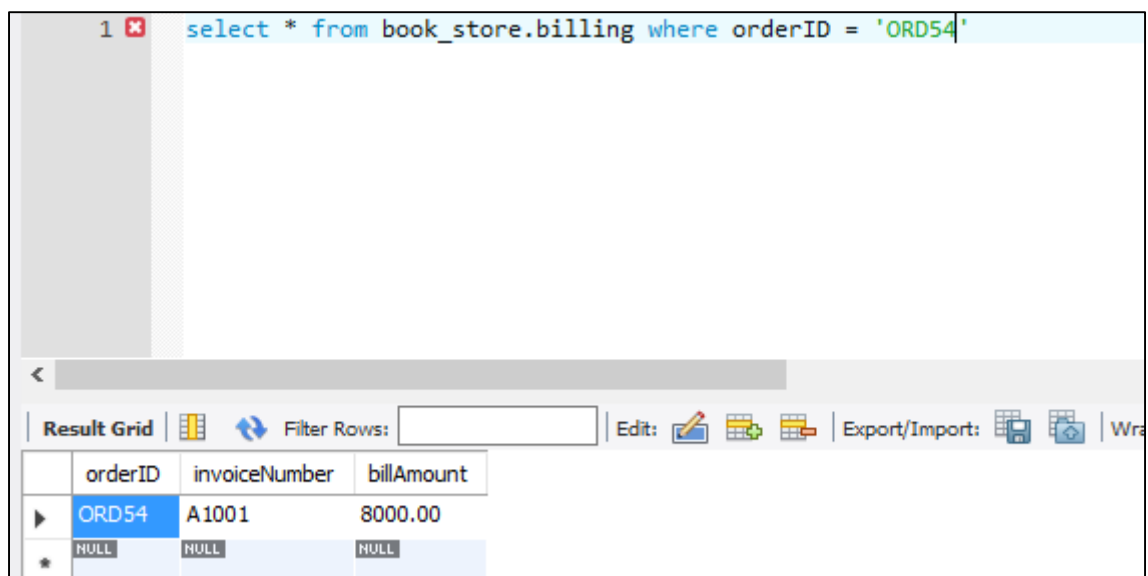
Select all the rows and click on Checkout:

orderID	quantity	BookID	OrderPlac...	customerID	price	orderStatus
ORD31	1	22	2016-06-30	363	2000	Processing
ORD54	1	55	2016-06-30	363	2000	Processing
ORD922	2	33	2016-06-30	363	2000	Processing

Number of Books:			2
Update Quantity		Checkout	



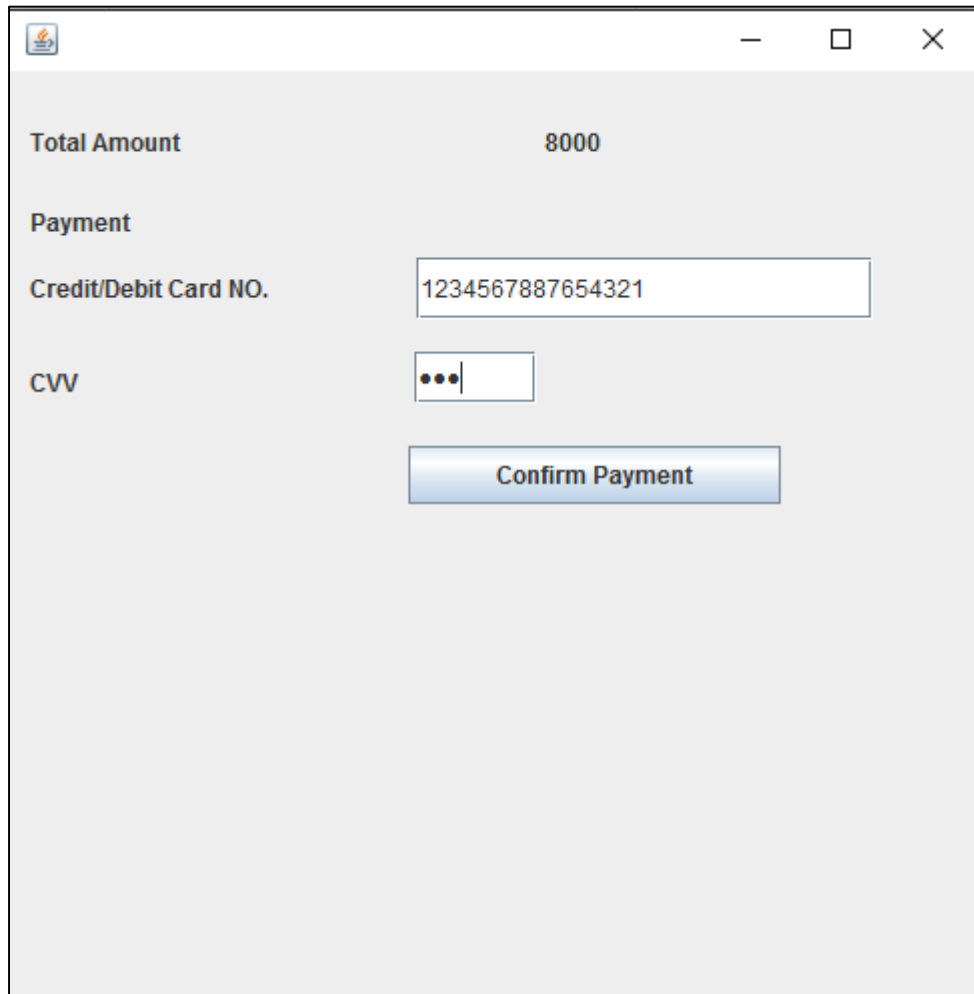
A screenshot of a payment confirmation window. The window has a title bar with a small icon and standard minimize, maximize, and close buttons. The main content area is light gray. At the top, it displays 'Total Amount' followed by '8000'. Below this, the section is titled 'Payment'. There are two input fields: 'Credit/Debit Card NO.' and 'CVV'. At the bottom, there is a blue button labeled 'Confirm Payment'.



A screenshot of a database management tool showing a SQL query and its results. The query is: `select * from book_store.billing where orderID = 'ORD54'`. Below the query, there is a 'Result Grid' section. The grid has columns for 'orderID', 'invoiceNumber', and 'billAmount'. The first row shows the results for 'ORD54'. Below the grid, there are icons for 'Filter Rows', 'Edit', and 'Export/Import'.

	orderID	invoiceNumber	billAmount
▶	ORD54	A1001	8000.00
★	NULL	NULL	NULL

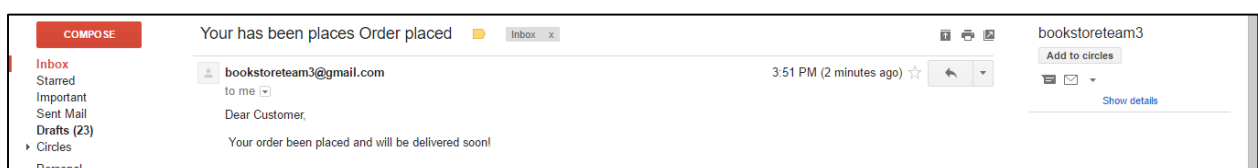
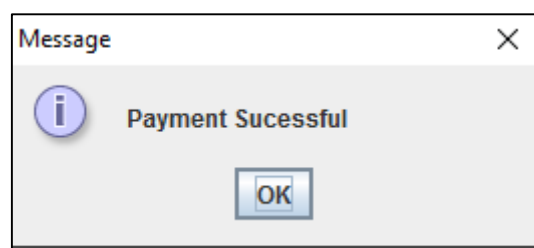
Enter 16-digit credit card number and CVV:



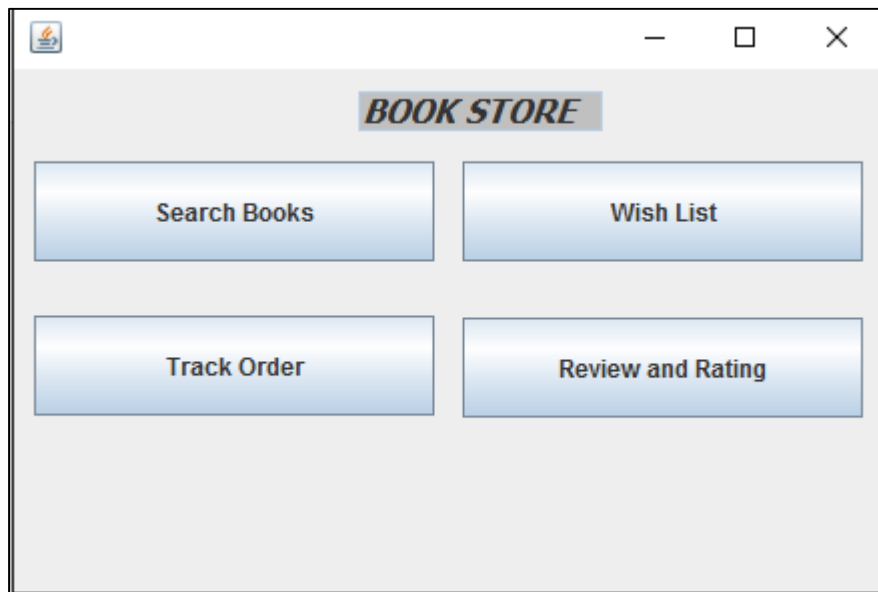
A screenshot of a payment form window. The window has a title bar with a small icon and standard minimize, maximize, and close buttons. The form content is as follows:

- Total Amount**: 8000
- Payment**: Section header
- Credit/Debit Card NO.**: A text input field containing the number 1234567887654321.
- CVV**: A text input field containing three dots and a vertical line, indicating masked digits.
- Confirm Payment**: A blue button with white text.

On Clicking Confirm Payment, a dialogue box is displayed and email is sent to customer:

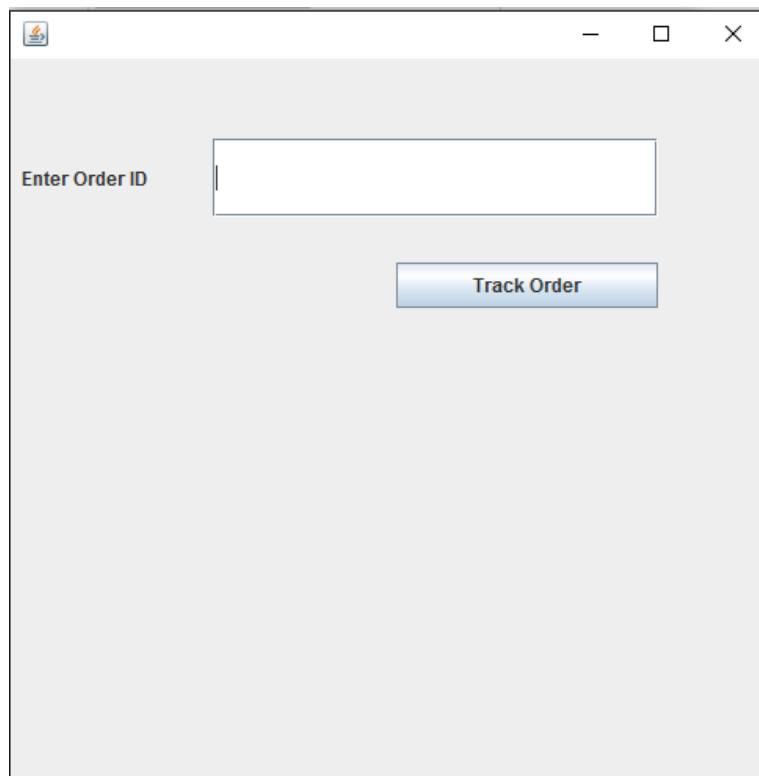


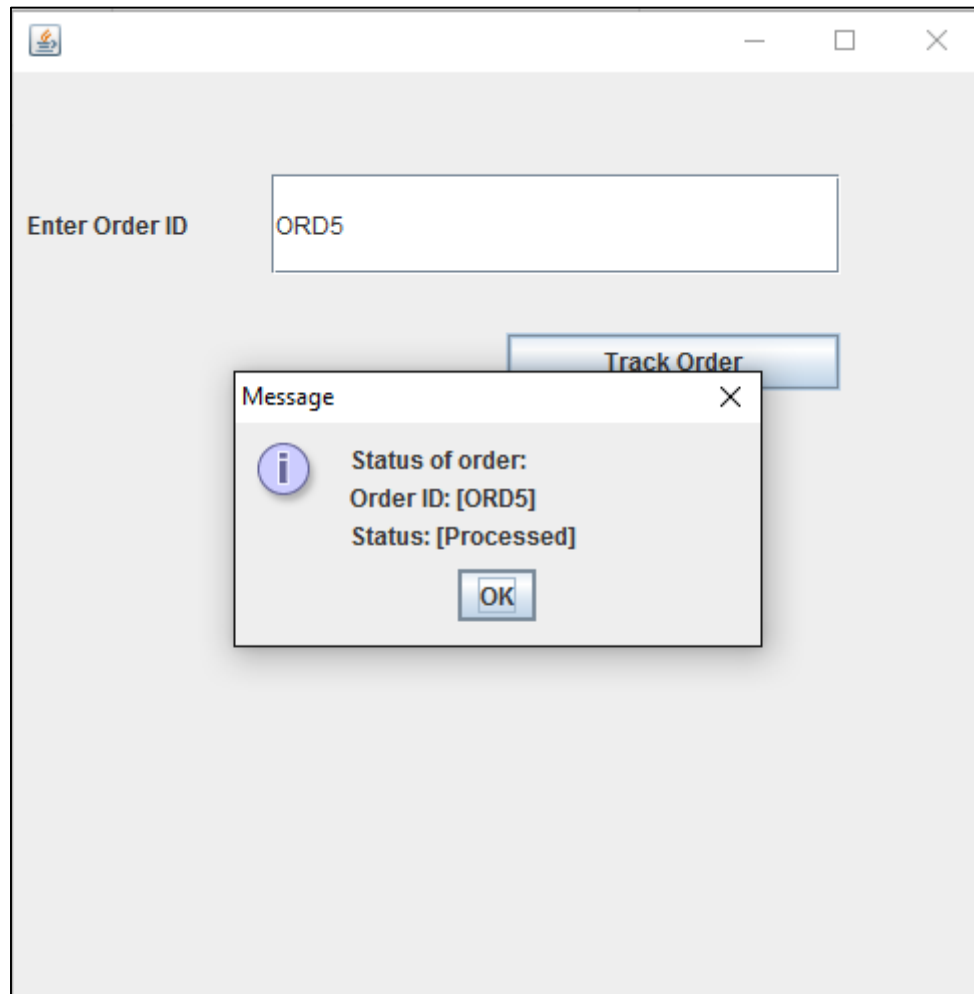
We again display the initial screen:



We can track the order by providing orderId:

Click on Track Order:

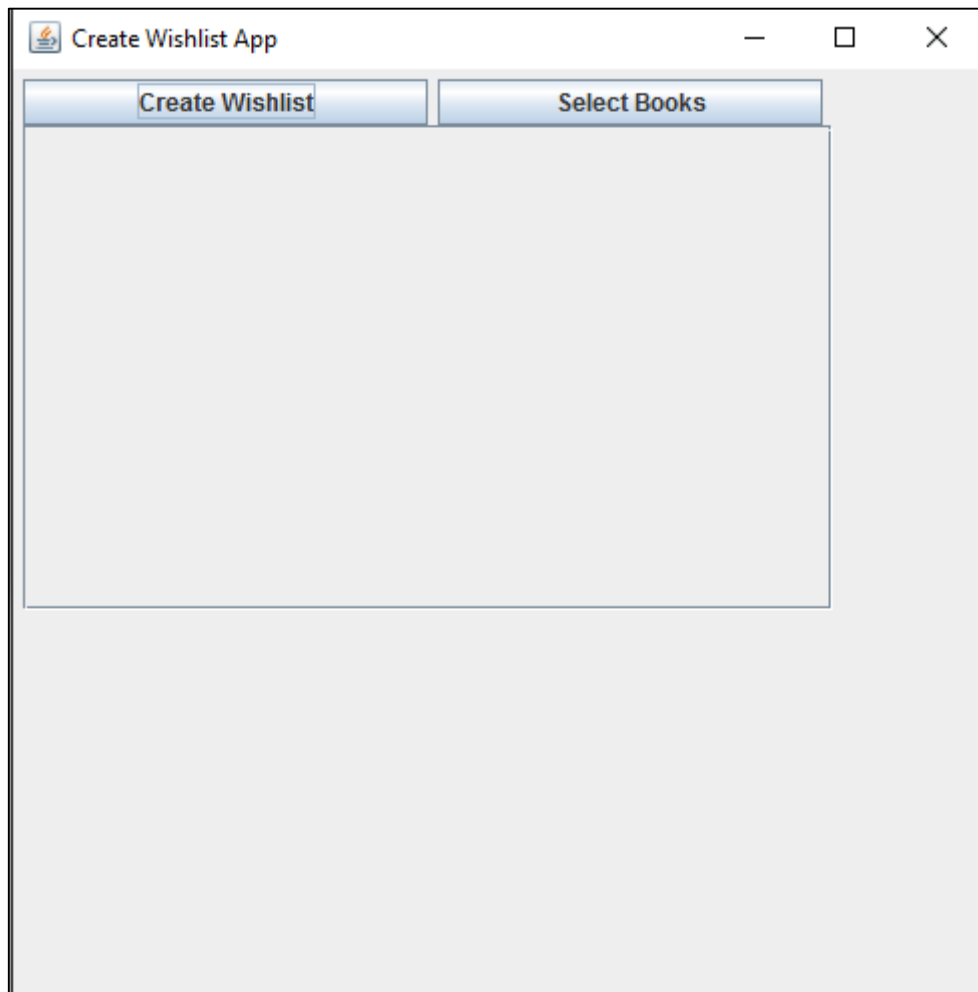




```
1 ✖ select * from book_store.order where orderID like 'ORD5'
```

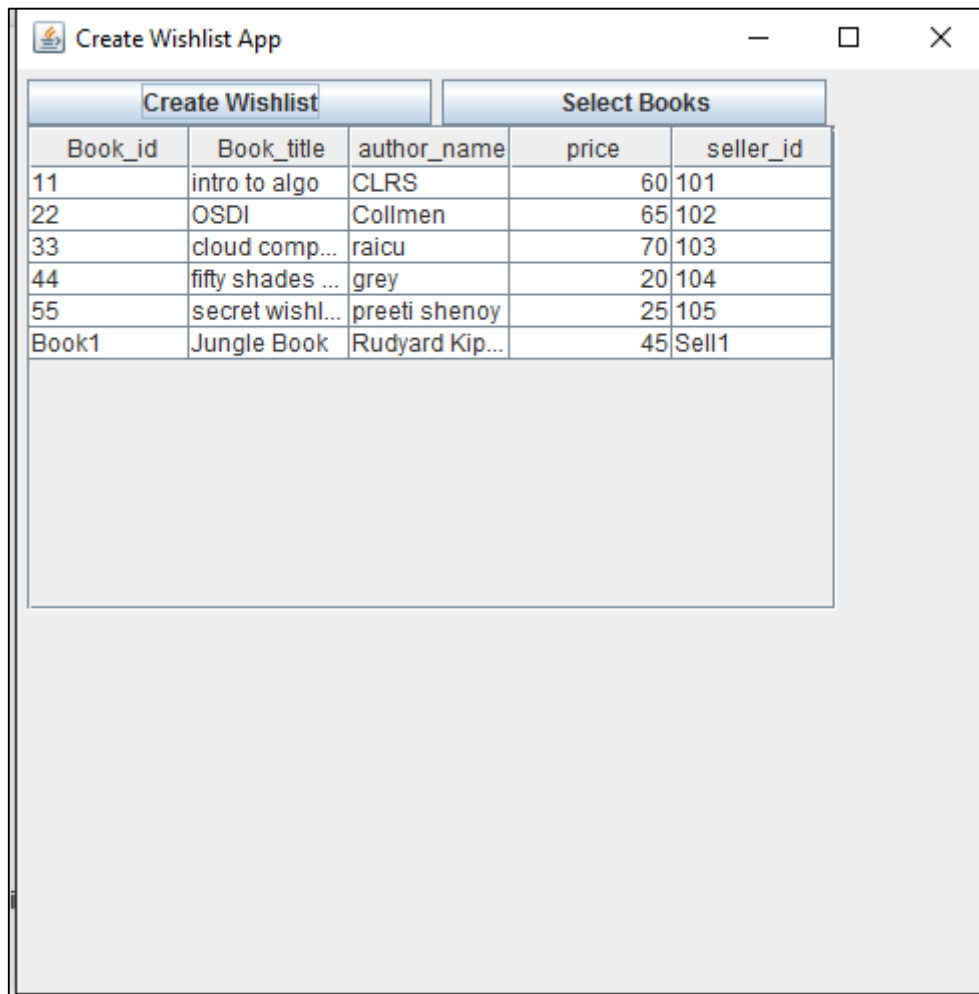
orderID	quantity	BookID	OrderPlacedDate	customerID	price	orderStatus
ORD5	1	Book1	2016-06-26	C0004	3016	Processed
NULL	NULL	NULL	NULL	NULL	NULL	NULL

Create Wishlist: Click on Wish List button:

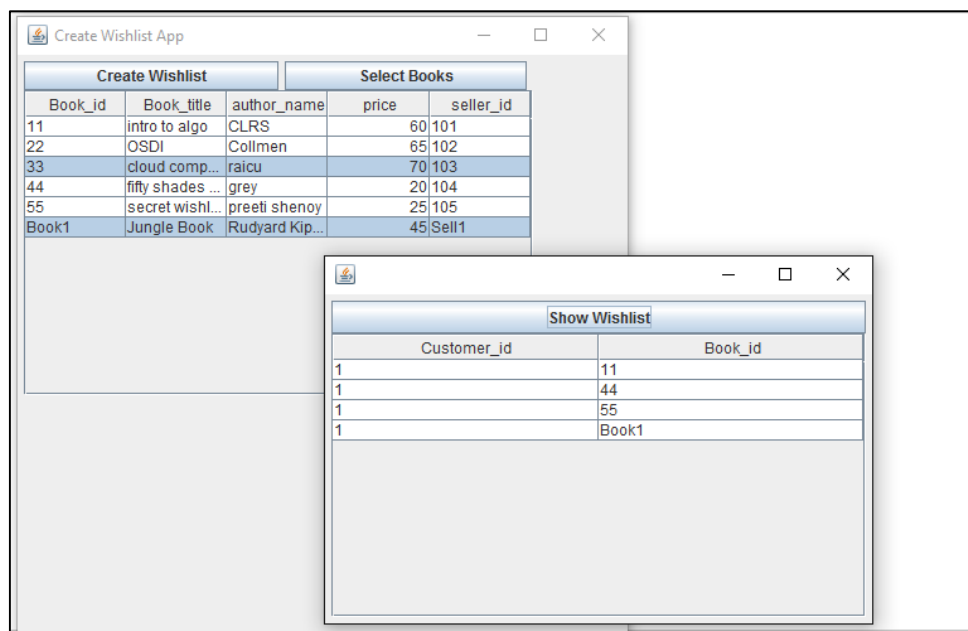


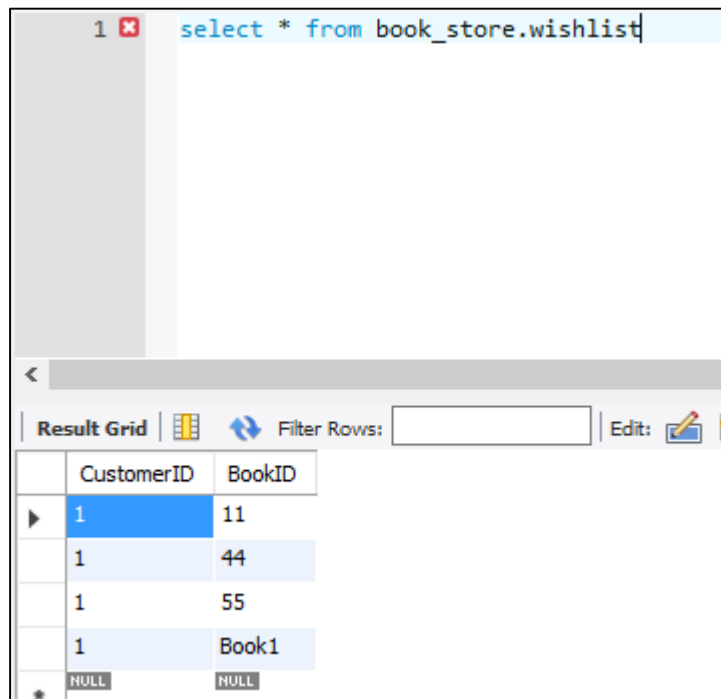
CS521 – Book Store Project Report

Click on create Wishlist button:



Select the books and click on select books:

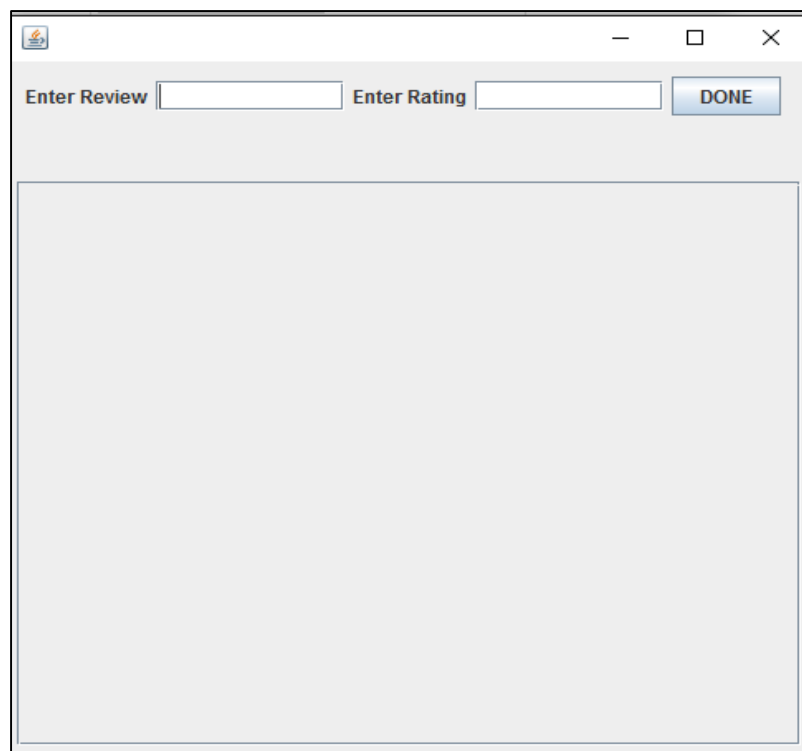




The screenshot shows a SQL query editor with the query `select * from book_store.wishlist` and a result grid below it. The result grid has two columns: CustomerID and BookID. The first four rows show CustomerID 1 with BookIDs 11, 44, 55, and Book1. The fifth row shows NULL for both fields.

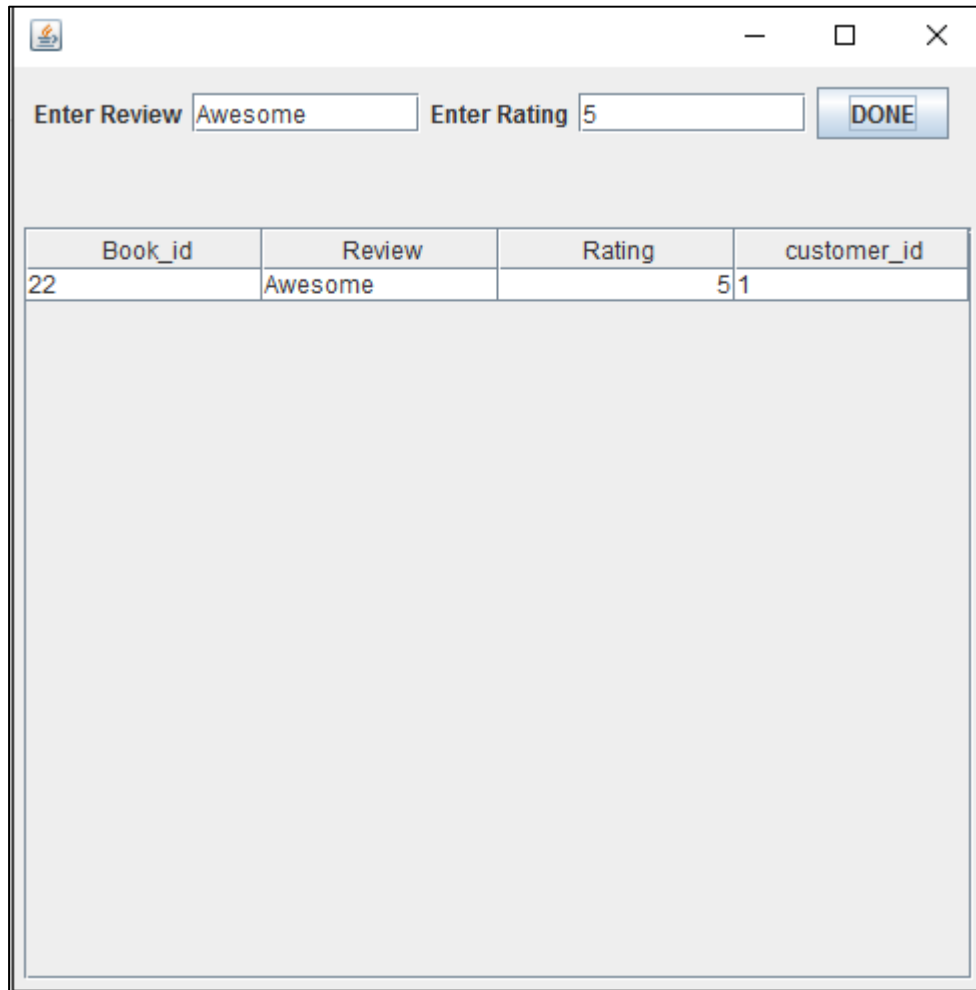
CustomerID	BookID
1	11
1	44
1	55
1	Book1
NULL	NULL

Click on Review and Rating button:



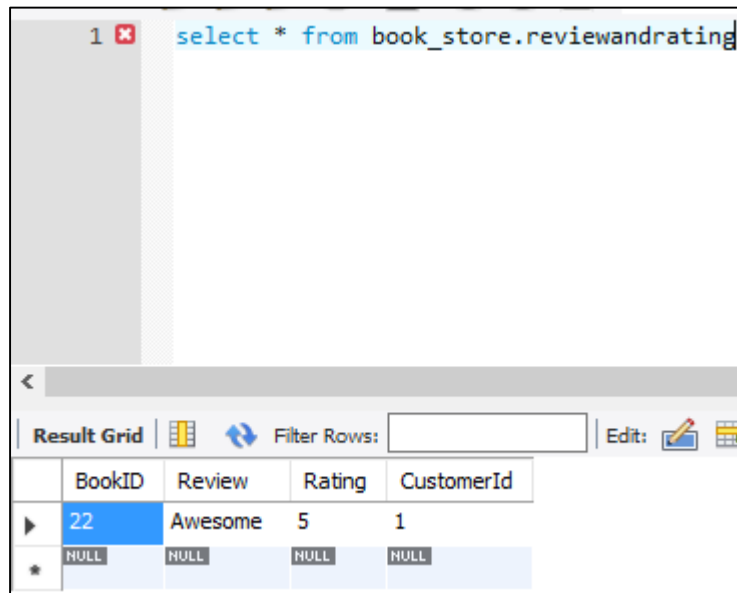
The screenshot shows a window titled "Enter Review" and "Enter Rating". It has two input fields for text and a "DONE" button. Below the input fields is a large empty text area for the review.

Enter Review Enter Rating



Enter Review Enter Rating

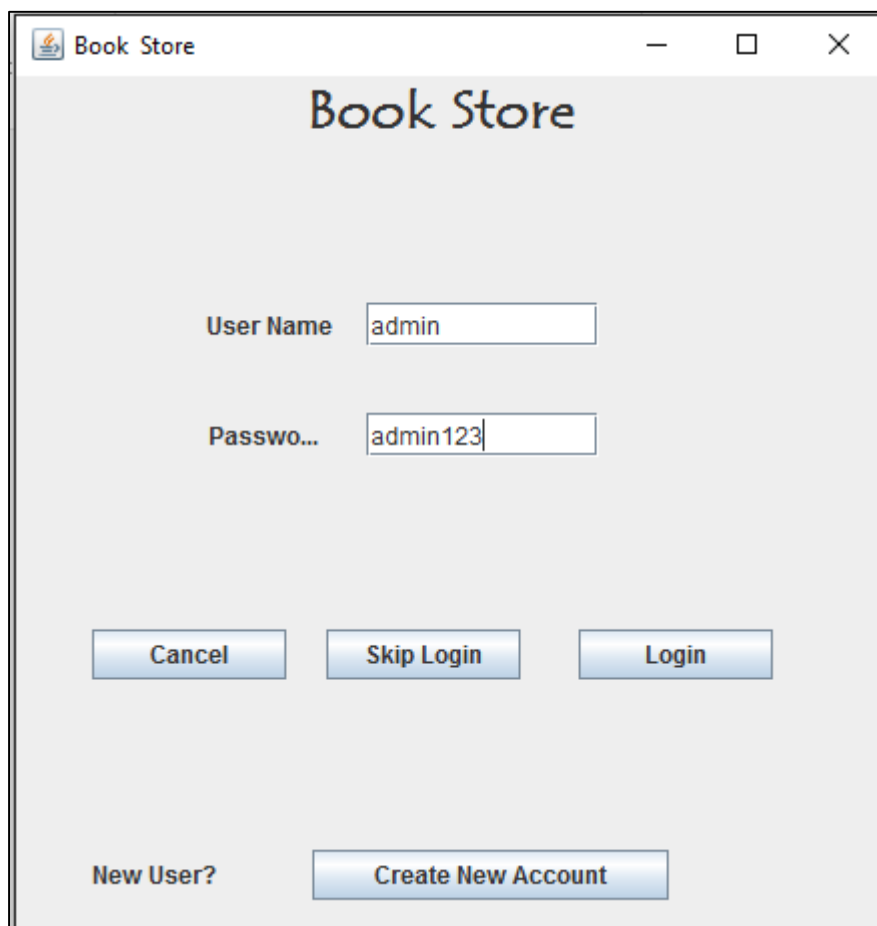
Book_id	Review	Rating	customer_id
22	Awesome	5	1



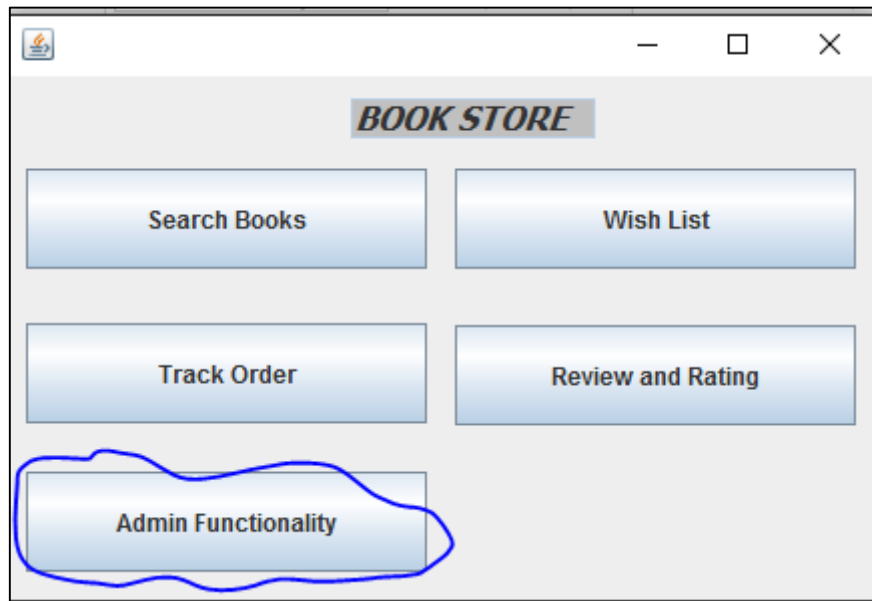
The screenshot shows a SQL query execution window. The query entered is `select * from book_store.reviewandrating`. Below the query, a "Result Grid" is displayed with the following data:

BookID	Review	Rating	CustomerId
22	Awesome	5	1
NULL	NULL	NULL	NULL

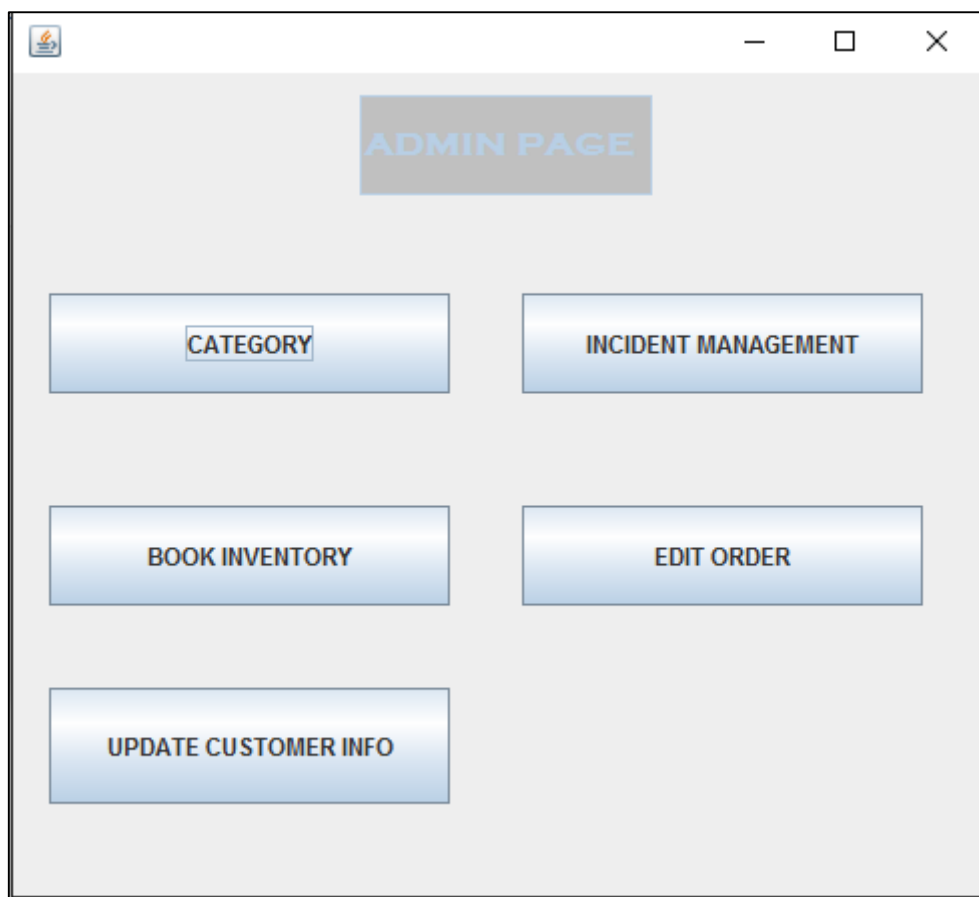
Next, login as Admin and view the admin functionalities. We have the user name for admin to be “admin” with password “admin123”



The screenshot shows the "Book Store" login window. It has a title bar with the text "Book Store" and standard window controls. The main content area has the title "Book Store" in a large font. Below the title, there are two input fields: "User Name" with the value "admin" and "Passwo..." (password) with the value "admin123". At the bottom, there are three buttons: "Cancel", "Skip Login", and "Login". Below these buttons, there is a "New User?" label and a "Create New Account" button.



This is the additional button provided only for the admin. On clicking this button the below screen is displayed:



Click on Category:

CategoryId	CategoryName
Cat1	Fiction
Cat2	Romance
Cat3	Sci-Fi
Cat4	Science

CategoryID:	Enter category Id here
Category Name:	Enter category name here
Add Category	
Delete Category	


Enter the new category details and click on Add Category:

CategoryId	CategoryName
Cat1	Fiction
Cat2	Romance
Cat3	Sci-Fi
Cat4	Science

CategoryId:	<input type="text" value="Cat5"/>
Category Name:	<input type="text" value="Educational"/>
<input type="button" value="Add Category"/>	
<input type="button" value="Delete Category"/>	

CategoryId	CategoryName
Cat1	Fiction
Cat2	Romance
Cat3	Sci-Fi
Cat4	Science

Message ✕

 Adding the following row:
 Category ID: [Cat5]
 Category Name: [Educational]




CategoryId:

Category Name:

```

1 ❌ select * from book_store.category
2

```

Result Grid |   Filter Rows: | Edit: 

	CategoryId	CategoryName
▶	Cat1	Fiction
	Cat2	Romance
	Cat3	Sci-Fi
	Cat4	Science
	Cat5	Educational
*	NULL	NULL

CategoryId	CategoryName
Cat1	Fiction
Cat2	Romance
Cat3	Sci-Fi
Cat4	Science
Cat5	Educational

CategoryId:	Cat5
Category Name:	Educational
<input type="button" value="Add Category"/>	
<input type="button" value="Delete Category"/>	

Select a particular row and click on delete category to delete it:

CategoryId	CategoryName
Cat1	Fiction
Cat2	Romance
Cat3	Sci-Fi
Cat4	Science
Cat5	Educational

CategoryId:	Cat5
Category Name:	Educational
Add Category	
Delete Category	

CategoryId	CategoryName
Cat1	Fiction
Cat2	Romance
Cat3	Sci-Fi
Cat4	Science

CategoryId:

Cat5

Category Name:

Educational

Add Category

Delete Category

1

2

select * from book_store.category

Result Grid

Filter Rows:

Edit:

CategoryId	CategoryName
Cat1	Fiction
Cat2	Romance
Cat3	Sci-Fi
Cat4	Science
NULL	NULL

Next click on Book Inventory:

BookID	NumberOfBooks	CategoryID
Book1	58	Cat1
Book2	200	Cat2

BookId:	Enter the book ID
Number of Books:	Enter the number of books
CategoryId:	Enter the category ID
Add Book	Update Book Details
Delete Book	

Enter new book details and click on Add book to add the book to the inventory:

BookID	NumberOfBooks	CategoryID
Book1	58	Cat1
Book2	200	Cat2

BookId:	<input type="text" value="22"/>
Number of Books:	<input type="text" value="455"/>
CategoryId:	<input type="text" value="Cat4"/>

Add Book	Update Book Details
Delete Book	

The screenshot shows a graphical user interface for a book store application. At the top, there is a table with three columns: BookID, NumberOfBooks, and CategoryID. The table contains two rows: Book1 with 58 books in Cat1, and Book2 with 200 books in Cat2. Below the table is a large gray rectangular area. In the center of this area is a 'Message' dialog box. The dialog box has a title bar with a close button (X) and an information icon (i). The message text inside the dialog reads: 'Adding the following row: Book Id: [22] Number of Books: [455] Category Id: [Cat4]'. There is an 'OK' button at the bottom of the dialog. Below the gray area, there are three input fields with labels: 'BookId:' with the value '22', 'Number of Books:' with the value '455', and 'CategoryId:' with the value 'Cat4'. At the bottom of the window, there are three buttons: 'Add Book', 'Update Book Details', and 'Delete Book'.

BookID	NumberOfBooks	CategoryID
Book1	58	Cat1
Book2	200	Cat2

Message

Adding the following row:
Book Id: [22]
Number of Books: [455]
Category Id: [Cat4]

OK

BookId: 22
Number of Books: 455
CategoryId: Cat4

Add Book Update Book Details
Delete Book

BookID	NumberOfBooks	CategoryID
Book1	58	Cat1
Book2	200	Cat2
22	455	Cat4

BookId:	22
Number of Books:	455
CategoryId:	Cat4

Add Book	Update Book Details
Delete Book	

```

1 select * from book_store.inventory
2

```

BookID	NumberOfBooks	CategoryID
22	455	Cat4
Book1	58	Cat1
Book2	200	Cat2
NULL	NULL	NULL

Select a row and enter the details to be edited:

BookID	NumberOfBooks	CategoryID
22	455	Cat4
Book1	58	Cat1
Book2	200	Cat2

BookId:	22
Number of Books:	555
CategoryId:	Cat4
Add Book	Update Book Details
Delete Book	

BookID	NumberOfBooks	CategoryID
22	555	Cat4
Book1	58	Cat1
Book2	200	Cat2

BookId:	22
Number of Books:	555
CategoryId:	Cat4
Add Book	Update Book Details
Delete Book	

1 ✖
2

```
select * from book_store.inventory
```

Result Grid

Filter Rows:

Edit:

	BookID	NumberOfBooks	CategoryID
▶	22	555	Cat4
	Book1	58	Cat1
	Book2	200	Cat2
*	NULL	NULL	NULL

Select the row and click Delete Book to delete it:

BookID	NumberOfBooks	CategoryID
22	555	Cat4
Book1	58	Cat1
Book2	200	Cat2

BookId:	22
Number of Books:	555
CategoryId:	Cat4

Add Book	Update Book Details
Delete Book	

BookID	NumberOfBooks	CategoryID
Book1	58	Cat1
Book2	200	Cat2

BookId:	<input type="text" value="22"/>
Number of Books:	<input type="text" value="555"/>
CategoryId:	<input type="text" value="Cat4"/>

Add Book	Update Book Details
Delete Book	

1
2

```
select * from book_store.inventory
```

Result Grid

Filter Rows:

Edit:

	BookID	NumberOfBooks	CategoryID
▶	Book1	58	Cat1
	Book2	200	Cat2
	NULL	NULL	NULL

Click on Incident Management:


IncidentID	CustomerID	EmployeeID	Description	Status
------------	------------	------------	-------------	--------

ID:	<input type="text"/>
CustId:	<input type="text"/>
EmpId	<input type="text"/>
Desc	<input type="text"/>
Status	<input type="text"/>

Add row to table	Update database
Delete Incident	

Enter the details and click on Add row to table:

IncidentID	CustomerID	EmployeeID	Description	Status
ID:	12			
CustId:	Cust1			
EmpId	Emp1			
Desc	Application is very slow			
Status	Open			
Add row to table			Update database	
Delete Incident				

IncidentID	CustomerID	EmployeeID	Description	Status
<div> <div>Message</div> <div>  <p>Adding the following row: Incident Number: [12] Status: [Open]</p> <div>OK</div> </div> </div>				
ID:	12			
CustId:	Cust1			
EmpId	Emp1			
Desc	Application is very slow			
Status	Open			
Add row to table			Update database	
Delete Incident				

IncidentID	CustomerID	EmployeeID	Description	Status
12	Cust1	Emp1	Application is v...	Open

ID:	12
CustId:	Cust1
EmpId	Emp1
Desc	Application is very slow
Status	Open

Add row to table	Update database
Delete Incident	

1 `select * from book_store.incident`

	IncidentID	CustomerID	EmployeeID	Status	Description	CreatedAt
▶	12	Cust1	Emp1	Open	Application is very slow	2016-06-30 16:28:23
★	NULL	NULL	NULL	NULL	NULL	NULL

IncidentID	CustomerID	EmployeeID	Description	Status
12	Cust1	Emp1	Application is v...	Open
13	Cust1	Emp1	Application is v...	Open

ID:	12
CustId:	
EmpId	
Desc	Application is very slow
Status	Serviced
Add row to table	
Delete Incident	
Update database	

IncidentID	CustomerID	EmployeeID	Description	Status
12	Cust1	Emp1	Application is very ...	Open
13	Cust1	Emp1	App is slow	Serviced

ID:	13
CustId:	Cust1
Empld	Emp1
Desc	App is slow
Status	Serviced


Add row to table	Update database
Delete Incident	

1 ✖ `select * from book_store.incident`

<

Result Grid | Filter Rows: | Edit: |

	IncidentID	CustomerID	EmployeeID	Status	Description
	12	Cust1	Emp1	Open	Application is very slow
✎	13	Cust1	Emp1	Servi...	App is slow
*	NULL	NULL	NULL	NULL	NULL

 Incident

IncidentID	CustomerID	EmployeeID	Description	Status
12	Cust1	Emp1	Application is very ...	Open
13	Cust1	Emp1	App is slow	Serviced

ID:
Custid:
Empld
Desc
Status

Add row to table

Update database

Delete Incident

IncidentID	CustomerID	EmployeeID	Description	Status
12	Cust1	Emp1	Application is very ...	Open

ID:	13
Custid:	Cust1
Empld	Emp1
Desc	App is slow
Status	Serviced
<div> <div>Add row to table</div> <div>Update database</div> </div>	
<div>Delete Incident</div>	

```
1 select * from book_store.incident
```

IncidentID	CustomerID	EmployeeID	Status	Description
12	Cust1	Emp1	Open	Application is very slow
NULL	NULL	NULL	NULL	NULL

Click on Edit Order:

orderID	quantity	BookID
ORD1	12	Book1
ORD11	55	Book1
ORD180	1	11
ORD2	1	Book1
ORD21	1	22
ORD221	6	55
ORD273	5	44
ORD291	4	33
ord3	600	Book1
ORD304	1	55
ORD31	1	22
ord4	5	Book1
ORD5	1	Book1
ORD54	1	55
ORD6	2	Book1
ORD61	56	22
ORD631	44	Book1
ORD641	4	22
ORD660	5	11
ORD661	45	11
ORD671	6	22
ORD714	1	55

Orderid:	Enter the order ID
Number of Books:	Enter the number of books
BookId:	Enter the book ID
Update Order	Delete Order

1 `select * from book_store.order`

<

Result Grid Filter Rows: Edit: Export/Import

	orderID	quantity	BookID	OrderPlacedDate	customerID	price	orderStatus
▶	ORD1	12	Book1	2016-06-26	C0001	4500	Pending
	ORD11	55	Book1	2016-06-29	CUS1	2000	Processed
	ORD180	1	11	2016-06-30	3	2000	Processing
	ORD2	1	Book1	2016-06-27	C0002	2001	Shipped
	ORD21	1	22	2016-06-30	3	2000	Processing
	ORD221	6	55	2016-06-30	3	2000	Processed
	ORD273	5	44	2016-06-30	3	2000	Processing
	ORD291	4	33	2016-06-29	3	2000	Processed
	ord3	600	Book1	2016-06-25	Cust1	453	Processing
	ORD304	1	55	2016-06-30	363	2000	Processing
	ORD31	1	22	2016-06-30	363	2000	Processing
	ord4	5	Book1	2016-06-24	Cust1	453	Processing
	ORD5	1	Book1	2016-06-26	C0004	3016	Processed
	ORD54	1	55	2016-06-30	363	2000	Processing
	ORD6	2	Book1	2016-06-28	C0003	2013	Shipped

Enter the details and click on Update Order to update the order:

orderID	quantity	BookID
ORD1	12	Book1
ORD11	55	Book1
ORD180	1	11
ORD2	1	Book1
ORD21	1	22
ORD221	6	55
ORD273	5	44
ORD291	4	33
ord3	600	Book1
ORD304	1	55
ORD31	1	22
ord4	5	Book1
ORD5	1	Book1
ORD54	1	55
ORD6	2	Book1
ORD61	56	22
ORD631	44	Book1
ORD641	4	22
ORD660	5	11
ORD661	45	11
ORD671	6	22
ORD714	1	55

OrderId:	ORD714
Number of Books:	66
BookId:	55
Update Order	Delete Order


orderID	quantity	BookID
ORD1	12	Book1
ORD11	55	Book1
ORD180	1	11
ORD2	1	Book1
ORD21	1	22
ORD221	6	55
ORD273	5	44
ORD291	4	33
ord3	600	Book1
ORD304	1	55
ORD31	1	22
ord4	5	Book1
ORD5	1	Book1
ORD54	1	55
ORD6	2	Book1
ORD61	56	22
ORD631		
ORD641		
ORD660		
ORD661		
ORD671		
ORD714		

Orderid:
Number of Books:
BookId:

Update Order

Delete Order

Message



Adding the following row:
Book Id: [55]
Number of Books: [66]
Order Id: [ORD714]

OK

orderID	quantity	BookID
ORD1	12	Book1
ORD11	55	Book1
ORD180	1	11
ORD2	1	Book1
ORD21	1	22
ORD221	6	55
ORD273	5	44
ORD291	4	33
ord3	600	Book1
ORD304	1	55
ORD31	1	22
ord4	5	Book1
ORD5	1	Book1
ORD54	1	55
ORD6	2	Book1
ORD61	56	22
ORD631	44	Book1
ORD641	4	22
ORD660	5	11
ORD661	45	11
ORD671	6	22
ORD714	66	55

OrderID:	ORD714
Number of Books:	66
BookId:	55
<input type="button" value="Update Order"/>	<input type="button" value="Delete Order"/>

1 `select * from book_store.order`

Result Grid Filter Rows: Edit: Export/Import:

	orderID	quantity	BookID	OrderPlacedDate	customerID	price	orderStatus
	ORD61	56	22	2016-06-29	sm	2000	Processed
	ORD631	44	Book1	2016-06-29	3	2000	Processed
	ORD641	4	22	2016-06-30	3	2000	Processing
	ORD660	5	11	2016-06-30	3	2000	Processing
	ORD661	45	11	2016-06-29	CUS1	2000	Processed
	ORD671	6	22	2016-06-30	3	2000	Processing
▶	ORD714	66	55	2016-06-30	363	2000	Processing
	ORD751	5	55	2016-06-30	3	2000	Processed
	ORD842	1	33	2016-06-30	363	2000	Processing
	ORD891	1	22	2016-06-30	363	2000	Processing
	ORD922	2	33	2016-06-30	363	2000	Processing
	ORD933	1	44	2016-06-30	3	2000	Processing
	ORD952	3	33	2016-06-30	1	2000	Processing
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Select the row and click on Delete Order to delete the order:

orderID	quantity	BookID
ORD1	12	Book1
ORD11	55	Book1
ORD180	1	11
ORD2	1	Book1
ORD21	1	22
ORD221	6	55
ORD273	5	44
ORD291	4	33
ord3	600	Book1
ORD304	1	55
ORD31	1	22
ord4	5	Book1
ORD5	1	Book1
ORD54	1	55
ORD6	2	Book1
ORD61	56	22
ORD631	44	Book1
ORD641	4	22
ORD660	5	11
ORD661	45	11
ORD671	6	22
ORD714	66	55

OrderID:	ORD714
Number of Books:	66
BookId:	55
Update Order	Delete Order

orderID	quantity	BookID
ORD21	1	22
ORD221	6	55
ORD273	5	44
ORD291	4	33
ord3	600	Book1
ORD304	1	55
ORD31	1	22
ord4	5	Book1
ORD5	1	Book1
ORD54	1	55
ORD6	2	Book1
ORD61	56	22
ORD631	44	Book1
ORD641	4	22
ORD660	5	11
ORD661	45	11
ORD671	6	22
ORD751	5	55
ORD842	1	33
ORD891	1	22
ORD922	2	33
ORD952	3	33

OrderId:	ORD714
Number of Books:	66
BookId:	55
Update Order	Delete Order

1 `select * from book_store.order`

< Filter Rows: Edit: Export/Import:

	orderID	quantity	BookID	OrderPlacedDate	customerID	price	orderStatus
	ORD6	2	Book1	2016-06-28	C0003	2013	Shipped
	ORD61	56	22	2016-06-29	sm	2000	Processed
	ORD631	44	Book1	2016-06-29	3	2000	Processed
	ORD641	4	22	2016-06-30	3	2000	Processing
	ORD660	5	11	2016-06-30	3	2000	Processing
	ORD661	45	11	2016-06-29	CUS1	2000	Processed
	ORD671	6	22	2016-06-30	3	2000	Processing
	ORD751	5	55	2016-06-30	3	2000	Processed
	ORD842	1	33	2016-06-30	363	2000	Processing
	ORD891	1	22	2016-06-30	363	2000	Processing
	ORD922	2	33	2016-06-30	363	2000	Processing
	ORD933	1	44	2016-06-30	3	2000	Processing
	ORD952	3	33	2016-06-30	1	2000	Processing

Click on Update Customer Info. The below screen is displayed:

FirstName	LastName	Address	PhoneNum...	EmailID	CustomerID
sruj	manvi	2901 S King...	7911119	srujana.ma...	1
Vaishnavi	Thangapalam	2988 SKD	790119	vt@gmail.com	3
Swathi	Shenoy	2901 S King...	3125369054	swathi.shen...	363
John	Smith	2111 SKD	22334444	johnsmith@...	Cust1

First Name:	<input type="text" value="Enter first name here"/>
Last Name:	<input type="text" value="Enter last name here"/>
Address:	<input type="text" value="Enter the address here"/>
Phone Number:	<input type="text" value="Enter the phone number here"/>
Email Id:	<input type="text" value="Enter the email ID here"/>
Customer ID:	<input type="text" value="whose details are to be updated"/>
<input type="button" value="Update Customer Details"/>	


Then Enter the details to be edited or updated:

FirstName	LastName	Address	PhoneNum...	EmailID	CustomerID
sruj	manvi	2901 S King...	7911119	srujana.ma...	1
Vaishnavi	Thangapalam	2988 SKD	790119	vt@gmail.com	3
Swathi	Shenoy	2901 S King...	3125369054	swathi.shen...	363
John	Smith	2111 SKD	22334444	johnsmith@...	Cust1

First Name:	Brinda
Last Name:	Rao
Address:	2988 SKD
Phone Number:	88776655
Email Id:	brindarao91@gmail.com
Customer ID:	Cust1
Update Customer Details	

FirstName	LastName	Address	PhoneNum...	EmailID	CustomerID
sruj	manvi	2901 S King...	7911119	srujana.ma...	1
Vaishnavi	Thangapalam	2988 SKD	790119	vt@gmail.com	3
Swathi	Shenoy	2901 S King...	3125369054	swathi.shen...	363
John	Smith	2111 SKD	22334444	johnsmith@...	Cust1

Message ✕

 Adding the following row:
 First Name: [Brinda]
 Last Name: [Rao]
 Address:[2988 SKD]
 Phone Number: [88776655]
 Email Id: [brindarao91@gmail.com]

First Name:	Brinda
Last Name:	Rao
Address:	2988 SKD
Phone Number:	88776655
Email Id:	brindarao91@gmail.com
Customer ID:	Cust1

FirstName	LastName	Address	PhoneNum...	EmailID	CustomerID
sruj	manvi	2901 S King...	7911119	srujana.ma...	1
Vaishnavi	Thangapalam	2988 SKD	790119	vt@gmail.com	3
Swathi	Shenoy	2901 S King...	3125369054	swathi.shen...	363
Brinda	Rao	2988 SKD	88776655	brindarao91...	Cust1

First Name:	Brinda
Last Name:	Rao
Address:	2988 SKD
Phone Number:	88776655
Email Id:	brindarao91@gmail.com
Customer ID:	Cust1
<input type="button" value="Update Customer Details"/>	

1 select * from book_store.customer

<

Result Grid Filter Rows: Edit: Export/Import: Wrap Cell Content:

	CustomerID	FirstName	LastName	DOB	EmailID	Address	PhoneNumber	UserName	Password
▶	1	sruj	manvi	1993-12-11	srujana.manvi@gmail.com	2901 S King Dr	7911119	ssss	test1
	3	Vaishnavi	Thangapalam	1994-12-12	vt@gmail.com	2988 SKD	790119	sm	test2
	363	Swathi	Shenoy	1990-01-30	swathi.shenoy3011@gmail.com	2901 S King Drive	3125369054	swathi3011	test3
	9119	Pavithra	Prakash	1992-01-01	pprakas8@gmail.com	2988 SKD	3125367786	ppavi	abcde
	Cust1	Brinda	Rao	0000-00-00	brindarao91@gmail.com	2988 SKD	88776655	jsmith	ggthyy
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

References

- [1]"Google", *Google.com*, 2016. [Online]. Available: <http://www.google.com>. [Accessed: 30-Jun- 2016].
- [2]"Main Page", *Wikipedia*, 2016. [Online]. Available: https://en.wikipedia.org/wiki/Main_Page. [Accessed: 30- Jun- 2016].
- [3] CS521 Tutorials