**“BookTown”**

**MAJOR PROJECT REPORT**

***Submitted by:-***

0206CS191119

0206CS191100

0206CS191090

0206CS191123

0206CS191062

**Piyush Raj Kumar**

**Manul Shrivastava**

**Kshitij Chauhan**

**Prateek Pathak**

**Deepak Manglani**

***in partial fulfillment for the award of the degree***

***of***

**BACHELOR OF TECHNOLOGY**

***in***

**COMPUTER SCIENCE AND ENGINEERING**



**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

GYAN GANGA INSTITUTE OF TECHNOLOGY & SCIENCES

JABALPUR (M.P.)

RAJIV GANDHI PRODYOGIKI VISHWAVIDYALAYA,

BHOPAL (M.P.)

**June- 2023**

#### *CERTIFICATE*

This is to certify that the Major Project Report entitled “**BookTown**” submitted by **Piyush Raj Kumar, Manul Shrivastava, Prateek Pathak, Kshitij Chauhan, Deepak Manglani** has been carried out under my guidance & supervision. The project report is approved for submission towards partial fulfillment of the requirement for the award of degree of **BACHELOR OF ENGINEERING** in **COMPUTER SCIENCE AND ENGINEERING** from **RAJIV GANDHI PROUDYOGIKI VISHWA-VIDYALAYA, BHOPAL (M.P).**

|  |  |
| --- | --- |
| Dr. Ashish Mishra  Guide  **Dept. of Computer Science and Engineering** | Dr. Ashok Verma  HoD  **Dept. of Computer Science and Engineering** |

#### *CERTIFICATE*

This is to certify that the Major Project Report entitled “**BookTown**” is submitted by **Piyush Raj Kumar, Manul Shrivastava, Prateek Pathak, Kshitij Chauhan, Deepak Manglani** for the partial fulfillment of the requirement for the award of degree of **BACHELOR OF ENGINEERING** in **COMPUTER SCIENCE AND ENGINEERING** from **RAJIV GANDHI PROUDYOGIKI VISHWAVIDYALAYA, BHOPAL (M.P).**

Internal Examiner External Examiner

Date: Date:

# 

# 

#### *DECLARATION*

We hereby declare that the project entitled **“BookTown”** which is being submitted in partial fulfillment of the requirement for award of the Degree of Bachelor of Engineering in Computer Science to **“RAJIV GANDHI PROUDYOGIKI VISHWAVIDYALAYA, BHOPAL (M.P.)”** is an authentic record of our own work done under the guidance of **Dr. Ashish Mishra, Department of Computer Science and Engineering,** **GYAN GANGA INSTITUTE OF TECHNOLOGY & SCIENCES, JABALPUR**.

The matter reported in this Project has not been submitted earlier for the award of any other degree.

**Date:**

**Place: JABALPUR**

#### *ACKNOWLEDGEMENT*

We sincerely express indebtedness to esteemed and revered guide **Dr. Ashish Mishra of Department of Computer Science** for his invaluable guidance, supervision and encouragement throughout the work. Without his kind patronage and guidance the project would not have taken shape.

We take this opportunity to express deep sense of gratitude to Dr**. Ashok Verma, Head of Department of Computer Science** for his encouragement and kind approval. Also we thank him in providing the computer lab facility. We would like to express our sincere regards to him for advice and counseling from time to time.

We owe sincere thanks to all the faculties in Department of Computer Science and Engineering for their advice and counseling time to time.

**Piyush Raj Kumar**

**Manul Shrivastava**

**Prateek Pathak**

**Kshitij Chauhan**

**Deepak Manglani**

**Date :**

##### **Place : JABALPUR**

**TABLE OF CONTENTS**

|  |  |  |
| --- | --- | --- |
| **Serial No.** | **Title** | **Page No.** |
| **1.** | **INTRODUCTION** | 9 |
|  | 1.1 Purpose of Project | 9 |
|  | 1.2 Project and Product Overview | 9 |
|  | 1.3 Intended Audience | 9 |
|  | 1.4 Team Architecture | 9 |

|  |  |  |
| --- | --- | --- |
|  | 1.5 Overall Description | 10 |
|  | 1.6 Product Perspective | 10 |

|  |  |  |
| --- | --- | --- |
| **2.** | **PROBLEM STATEMENT** | 11 |
|  | 2.1 Business Requirements | 11 |
|  | 2.1.1 Entry Point | 11 |
|  | 2.1.2 Selection of Product | 11 |
|  | 2.1.3 Reports | 11 |
|  | 2.2 System Requirement | 12 |
|  | 2.2.1 Usability | 12 |
| **3.** | **PROJECT UNDERSTANDING DOCUMENT** | 13 |
|  | 3.1 Purpose of Project | 13 |
|  | 3.2 Objective | 13 |
|  | 3.3 MIS Reports | 13 |

|  |  |  |
| --- | --- | --- |
| **4.** | **DURATION** | 14 |
|  | 4.1 Timeline | 14 |

|  |  |  |
| --- | --- | --- |
| **5.** | **REQUIREMENTS** | 15 |
|  | 5.1 Specific Requirements | 15 |
|  | 5.1.1 External Interface Requirements | 15 |
|  | 5.1.2 Hardware Interface | 15 |
|  | 5.1.3 Software Interface | 15 |
|  | 5.2 Non functional Requirements | 15 |
|  | 5.3 Software System Attributes | 16 |

|  |  |  |
| --- | --- | --- |
| **6.** | **DESIGN TECHNIQUES** | 17 |
| **7.** | **TIER ARCHITECTURE** | 19 |

|  |  |  |
| --- | --- | --- |
| **8.** | **SOFTWARE PROCESS MODEL** | 21 |
|  | 8.1 Why not Evolutionary models? | 21 |
|  | 8.2 Why not Waterfall model? | 21 |
|  | 8.3 Why Incremental RAD model? | 21 |
|  | 8.4 Observation | 21 |
|  | 8.5 Determining project feasibility | 22 |
| **9.** | **DESIGN** | 23 |
|  | 9.1 BPMN | 23 |
|  | 9.2 DFD | 24 |
|  | 9.3 USE CASE DIAGRAM | 27 |
|  | 9.4 CLASS DIAGRAM | 28 |
|  | 9.5 OBJECT DIAGRAM | 29 |

|  |  |  |
| --- | --- | --- |
|  | 9.6 SEQUENCE DIAGRAM | 30 |
| **10.** | **DATABASE** | 31 |
| **11.** | **SCREENSHOTS** | 32 |
| **12.** | **TEST CASES** | 36 |
| **13.** | **CONCLUSION** | 38 |

|  |  |  |
| --- | --- | --- |
| **14.** | **BIBLIOGRAPHY** | 39 |

**LIST OF FIGURES**

|  |  |  |
| --- | --- | --- |
| S.No. | Figure name | Page No. |
| 9.1 | BPMN | 23 |
| 9.2 | DFD (0 LEVEL) | 24 |
| 9.2 | DFD (1 LEVEL) | 25 |
| 9.2 | DFD (2 LEVEL) | 26 |
| 9.3 | USE CASE DIAGRAM | 27 |
| 9.4 | CLASS DIAGRAM | 28 |
| 9.5 | ACTIVITY DIAGRAM | 29 |
| 9.6 | SEQUENCE DIAGRAM | 30 |

**1. INTRODUCTION:**

Welcome to the introduction of our online book purchasing website project. This project is designed to provide a user-friendly and convenient platform for book lovers to purchase their desired books online without any hassle. In this introduction, we will provide an overview of the project's purpose, product, intended audience, team architecture, overall description, and product perspective.

* 1. **Purpose of Project:**

The purpose of this project is to create an online book purchasing website that simplifies the process of buying books online. With the advent of the internet, more and more people are turning to online shopping for their convenience and time-saving benefits. However, purchasing books online can be a hassle, especially when searching for specific titles or authors. Therefore, the aim of this project is to create a user-friendly platform where customers can easily search for books, browse different categories, and make purchases securely.

**1.2 Project and Product Overview:**

The project will involve the creation of a user-friendly website that will allow customers to purchase books online. The website will feature various book categories such as fiction, non-fiction, biography, children's books, etc. Customers will be able to search for books using various filters such as author, title, genre, and publisher. The website will also offer a secure payment gateway for customers to make their purchases.

The product overview of our website will include the following features:

* User registration and login
* Book search functionality
* Book categories and filters
* Product pages with book details, prices, and reviews
* Shopping cart and checkout process
* Secure payment gateway
* Order tracking and history

**1.3 Intended Audience**:

The intended audience for this online book purchasing website includes book lovers, students, researchers, and anyone who wishes to purchase books online conveniently. The website will be designed to cater to all age groups and provide a diverse selection of books. Customers who prefer to shop from the comfort of their own homes or who do not have access to physical bookstores will benefit greatly from this website.

**1.4 Team Architecture:**

The project will require a team of developers, designers, and content creators to work together to create a seamless shopping experience for customers. The team will consist of front-end and back-end developers, UI/UX designers, content writers, and project managers. The front-end developers will be responsible for creating the user interface and designing the website's layout. The back-end developers will be responsible for creating the website's functionality, such as the search and filter options, shopping cart, and checkout process. The UI/UX designers will ensure that the website is user-friendly and easy to navigate, while the content writers will create compelling descriptions for each book. Finally, the project managers will oversee the project's progress and ensure that the website is delivered on time and within budget.

**1.5 Overall Description:**

The overall description of the online book purchasing website is to provide customers with a convenient and secure platform to purchase books online. The website will be designed to cater to all age groups and provide a diverse selection of books. Customers will be able to search for books using various filters, add items to their shopping cart, and proceed to the checkout process, where they can make payments securely. The website will also offer an order tracking and history feature, allowing customers to keep track of their purchases.

**1.6 Product Perspective:**

The product perspective of the online book purchasing website is to provide an easy and convenient platform for customers to purchase books online. The website will be designed to cater to all age groups and provide a diverse selection of books. The website will be easy to use and offer a secure payment gateway, making it a reliable platform for customers to purchase books online. The website's intuitive design and functionality will make it a go-to platform for book lovers looking for a hassle-free online shopping experience.

**2. PROBLEM STATEMENT:**

The current process of purchasing books online can be cumbersome and time-consuming, with limited options for filtering and searching. Additionally, customers may face issues with the security of online payments and the reliability of delivery. Therefore, the problem that this online book purchasing website aims to address is to simplify the process of buying books online, provide a diverse selection of books, and offer secure and reliable payment and delivery options.

**2.1 Business Requirements:**

**2.1.1 Entry Point:**

The website must have an easy and accessible entry point for customers to begin their shopping experience. The homepage should provide clear navigation options and a search bar for customers to find the books they want quickly.

The system is required to have two entry points:

1) Vendor Direct: A “Vendor Direct” entry point is where the vendor can manage the inventory and service the customer requests.

2) Customer Linking: A “Customer Linking” entry point is where a prospective customer can view the details of products and place their order.

**2.1.2 Selection of Product:**

The website will provide customers with a wide selection of books across various categories, including fiction, non-fiction, children's books, etc. Customers should be able to filter their search results by author, title, genre, and publisher to find the books they want quickly.

**2.1.3 Reports:**

The website should provide reports on customer purchases and order history. Customers should be able to view their purchase history, track their orders, and receive notifications about the status of their orders and can generate the following reports: -

* Annual reports
* Monthly reports
* Sales reports
* Customer trends

**2.2 System Requirements:**

Performance and Scalability: The system is required to scale to support transaction volume. Webpages should be light and render fast.

**2.2.1 Usability:**

The website will have a user-friendly interface that is easy to navigate and use. It should provide clear and concise information about each book, including its title, author, publisher, price, and reviews. The website should also provide secure and reliable payment options, including credit/debit card payments and cash on delivery, to ensure customer satisfaction.

Overall, the online book purchasing website should provide a hassle-free and enjoyable shopping experience for customers, with easy navigation, diverse book selections, secure payment options, and reliable delivery.

**3. PROJECT UNDERSTANDING DOCUMENT:**

**3.1 Purpose of Project:**

The purpose of this project is to develop an online book purchasing website that provides customers with a simple and user-friendly interface to browse and purchase books online. The website should offer a wide selection of books across various categories, provide secure and reliable payment options, and ensure timely delivery of orders.

**3.2 Objectives:**

The objectives of the project are as follows:

* To provide customers with a user-friendly interface for browsing and purchasing books online.
* To offer a wide selection of books across various categories, including fiction, non-fiction, children's books, etc.
* To provide secure and reliable payment options, including credit/debit card payments and cash on delivery.
* To ensure timely delivery of orders, with regular updates and notifications about the status of orders.
* To offer MIS reports on customer purchases and order history, allowing customers to view their purchase history and track their orders.

**3.3 MIS Reports:**

The website should provide MIS reports on customer purchases and order history. These reports should include:

* Order history: A report that shows a customer's order history, including details such as the date of purchase, book titles, quantity, price, and delivery status.
* Sales report: A report that provides an overview of the website's sales, including the total number of orders, the total revenue generated, and the most popular book categories.
* Inventory report: A report that shows the current inventory levels for each book title, including the number of copies available and the number of copies sold.
* Customer feedback report: A report that summarizes customer feedback on the website, including ratings and reviews of books, delivery experience, and customer service.
* Overall, the online book purchasing website should provide customers with a hassle-free and enjoyable shopping experience, while also providing valuable MIS reports to help the website's management make informed decisions about the website's performance and direction.

**4. DURATION**

4.1 Timeline

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Sr. No** | **Project Phase** | **Phase Period (Working Days)**  **(1)** | **No. of Consultants (2)** | **Category** | **Total person days**  **(1) \* (2)** |
|  | Business Requirement Study | 7 | 3 | Business Consultant | 21 |
|  | Low Level/Application Design | 20 | 4 | Technical Associates and Design Specialists | 80 |
|  | Development and Unit Testing | 40 | 5 | Database Administrator /Technical Associates and Project manager | 200 |
|  | Test & Bug Fixing | 13 | 3 | Tester & Technical Associates | 39 |
|  | Documentation | 30 | 3 | Technical Associates | 90 |
|  | Deployment User Acceptance Testing | 5 | 3 | Project Leaders & Technical Associates | 15 |
|  | User Training/handover | 10 | 4 | Project Leaders | 40 |
|  | Project Management | 10 | 2 | Project Leader/Manager | 20 |
|  | Total |  |  |  | 505 |

**5. REQUIREMENTS**

**5.1 SPECIFIC REQUIREMENTS**

**5.1.1 EXTERNAL INTERFACE REQUIREMENTS**

The external system is to assume full responsibility for storage functions as well as warehouse management and warehouse control for an entire warehouse. The interfaces in this section are specified by documenting: the name and description of each scheme, source or input, destination or output, ranges, accuracy and tolerances, units of measure, timing, display formats and organization, and data formats. The interfaces in this section are specified by documenting: the name and description of each scheme, source or input, destination or output, ranges, accuracy and tolerances, units of measure, timing, display formats and organization, and data formats.

The user interface required to be developed for the system should be user friendly and attractive. The interface between the user and the system will be WIMP (Windows, Icons, Menu, Pointers) keeping in mind that the system is to be run through web browser. All operations will be of point and click nature with all navigations performed through windows of the system specifically buttons and menus:

Buttons: The button is activated when the user will click with the left click of the mouse within the bounds of the button. And thus, the action associated with it will be carried out.

Menu: All the operations will be arranged.

**5.1.2 HARDWARE INTERFACE**

Here's What You Need to Use the inventory management system:

• 20 GB HDD

• 256 MB RAM

• Pentium IV Processor

• Input Devices: Keyboard, Mouse

• Output Devices: Monitor, Printer

**5.1.3 SOFTWARE INTERFACE**

Supporting Tools: Blue print foundry 4.0, Microsoft Word 2007, Microsoft PowerPoint 2007.

**5.2 NON-FUNCTIONAL REQUIREMENTS**

**•** Performance Criteria: Time. The elapsed time between the submission of documents process between customer to vendor that between vendor to company should be as minimum as possible. Similarly, there is being a minimal gap between all information about products.

• User friendly: Our Inventory Management system should be more users friendly. The user interface should be kept simple and uncluttered. Since different type of people will interact in this process so our project should be very easy to them to understand.

• Flexibility: Our project should be so flexible that whenever we want to make changes in it very easily it can be done on

• Extensibility:

It should be able to accommodate the variations like:

> Different schemes should be handled easily.

> Client interaction after sending his/her details.

> It should be able for direct money transfer from one place to another.

• Portable: Our project should be portable on any platform and available on websites easily and at a faster speed than others.

• Reusable: All the client web pages that are being used for client information should be easily get processed so that many clients can interact with us very easily and very fast without any information destroy.

**5.3 SOFTWARE SYSTEM ATTRIBUTES**

• Reliability: The health insurance process on the project should be easy and without any mistakes so that clients should take information about all the policies and their interest rate and update by company should be very easy and safe.

• Availability: The project should be available 24 hours a day, 7 days a week. The availability can be measured in terms of MTTR (Mean Time to Repair) and MTBF (Mean Time Between failures). The system will be available to the user whenever the user needs it.

• Maintainability: Our project should be easy to maintain by administrators or by our company. After certain of time system should be added a new policy and our user interactive schemes so that we can deal with our users according to market and time.

**6. DESIGN TECHNIQUES**

Design of the site has been done using the following technologies: -

* HTML
* CSS
* JavaScript
* Bootstrap
* SQL YOG

**HTML: HYPER TEXT MARKUP LANGUAGE**

In computing, Hypertext Markup Language (HTML) is a markup language designed for creation of web pages with hypertext and other information to be displayed in a web browser. HTML is used to structure information denoting certain text as headings, paragraphs, lists and so on and can be used to describe, to some degree, the appearance and semantics of a document. HTML’s grammar structure is the HTML DTD that was created using SGML syntax.

The HTML document format is used on the Web. Web pages are built with HTML tags (codes) embedded in the text. HTML defines the page layout, fonts and graphic element as well as the hypertext links to other documents on the web. Each link contains the URL, or address, of a Web page residing on the same server or any server worldwide, hence “World Wide Web”.

HTML is a markup language (the ML in HTML) that uses a fixed set of markup tags. A markup language can also be thought of as a “Presentation Language”, but it is not a programming language. You cannot “if this-do that” like you can in Java, JavaScript or C++. However, in order to make pages interactive, programming code can be embedded in an HTML page.

**CSS: CASCADING STYLE SHEETS**

CSS stands for Cascading Style Sheets. It is a style sheet language which is used to describe the look and formatting of a document written in markup language. It provides an additional feature to HTML. It is generally used with HTML to change the style of web pages and user interfaces. It can also be used with any kind of XML documents including plain XML, SVG and XUL.

CSS is used along with HTML and JavaScript in most websites to create user interfaces for web applications and user interfaces for many mobile applications.

**JAVASCRIPT:**

JavaScript (js) is a light-weight object-oriented programming language which is used by several websites for scripting the webpages. It is an interpreted, full-fledged programming language that enables dynamic interactivity on websites when applied to an HTML document. It was introduced in the year 1995 for adding programs to the webpages in the Netscape Navigator browser. Since then, it has been adopted by all other graphical web browsers. With JavaScript, users can build modern web applications to interact directly without reloading the page every time. The traditional website uses js to provide several forms of interactivity and simplicity.

**BOOTSTRAP:**

Bootstrap is a free and open-source web development framework. It’s designed to ease the web development process of responsive, mobile-first websites by providing a collection of syntax for template designs.

**MySQL**

Modern day web sites seem to be relying more and more on complex database systems. These systems store all of their critical data, and allow for easy maintenance in some cases.

The Structured Query Language (SQL) is a very popular database language, and its standardization makes it quite easy to store, update and access data. One of the most powerful SQL servers out there is called MySQL and surprisingly enough, its free.

Some of the features of MySQL Include: Handles large databases, in the area of 50,000,000+ records. No memory leaks. Tested with a commercial memory leakage detector (purify). A privilege and password system which is very flexible and secure, and which allows host-based verification. Passwords are secure since all password traffic when connecting to a server is encrypted.

**7. TIER ARCHITECTURE**

The various classes as obtained from the business class diagram is categorized as follows-

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Form of the project** |  | **Class** |  | **Class** |
| **Application or Presentation Layer** |  | **Business Layer or Logical Layer** |  | **Data Layer or Data Access Layer** |

The 3-tier architecture consists of three layers:

**Presentation Layer** - The web site or windows forms application is called the presentation layer. The presentation layer is the most important layer simply because it’s the one that everyone sees and uses. Even with a well-structured business and data layer, if the presentation layer is designed poorly, this gives the users a poor view of the system. Presentation layer is the form where we design using the controls like textbox, labels, command buttons etc.

**Business Layer** - Though a web site could talk to the data access layer directly, it usually goes through another layer called the business layer.

This layer is a class which we use to write the function which works as a mediator to transfer the data from Application or presentation layer data layer. In the three-tier architecture we never let the data access layer to interact with the presentation layer.

This layer is also a class where we declare the variable corresponding to the fields of the database which can be required for the application and make the properties so that we can get or set the data using these properties into the variables. These properties are public so that we can access its values.

One of the best reasons for reusing logic is that applications that start off small usually grow in functionality. For instance, a company begins to develop a web site, and as they realize their business needs, they later decide to add a smart client application and windows service to supplement the web site. The business layer helps move logic to a central layer for “maximum reusability.”

Business layer have been presented having two roles

* client application
* server component

**EXAMPLE OF BUSINESS LAYER-**

The Business layer has functions of which takes the parameters from the example given in the presentation layer .As the user inputs the data values, corresponding functions are called in the business layer which are further passed on through the data layer where corresponding procedures are called and the data is been updated.

Business layer is the class where we write the functions which get the data from the application layer and passes through the data access layer.

**Data layer** - The key component to most applications is the data. The data has to be served to the presentation layer somehow. The data layer is a separate component whose sole purpose is to serve up the data from the database and return it to the caller. This layer is also a class which we use to get or set the data to the database back and forth. This layer only interacts with the database. We write the database queries or use stored procedures to access the data from the database or to perform any operation to the database.

**ADVANTAGE OF 3 TIER ARCHITECTURE**

* Client-Server architecture is 2-Tier architecture because the client does not distinguish between Presentation layer and business layer.
* The increasing demands on GUI controls caused difficulty to manage the mixture of source code from GUI and Business Logic.
* Further, Client Server Architecture does not support enough the Change Management. Let suppose that the government increases the Entertainment tax rate from 4% to 8 %, then in the Client-Server case, we have to send an update to each client and they must update synchronously on a specific time otherwise we may store invalid or wrong information.
* The Client-Server Architecture is also a burden to network traffic and resources. Let us assume that about five hundred clients are working on a data server then we will have five hundred ODBC connections and several ruffian record sets, which must be transported from the server to the clients.
* This categorization of the application makes the function more reusable easily and it becomes too easy to find the functions which have been written previously. If programmer wants to make further update in the application, then he easily can understand the previous written code and can update easily.

**DISADVANTAGES**

* Increase complexity /effort
* More difficult to build 3 tier architecture rather than a 2 tier.
* Points of communication are doubled
* Maintenance tools are currently inadequate for maintaining server libraries.

**8. SOFTWARE PROCESS MODEL**

**8.1** **Why not Evolutionary models?**

These models are best suited where requirements are fuzzy. These models are best suited for the systems where requirements keep on changing. But for our system requirements are crystal clear so it is not feasible to adopt any of the evolutionary models.

**8.2** **Why not Waterfall model?**

Waterfall model can be adopted because in our case because requirements are known in advance but there are some limitations of waterfall model due to which it is not feasible to adopt:

* No parallelism of work.
* Time consuming

**8.3 Why Incremental RAD model?**

Incremental model is advisable where requirements are clear and the development time is less. The striking feature of incremental model is that each module can be completed and released as and when requirement arises because of lack of time.

As in our system many of the modules are not inter-related so can be released in isolation. The user can thus get a feel of these modules and give his feedback which can be utilized for making the software more user friendly and in line with the user requirements.

Not only that the deadline set for this project is 3 months and we need a high adaptation model and again will be concentrating on parallelism because our team will be working on different module on same time. Moreover, we will be using latest tools such as Visio, Project Manager as a result of which we can work much faster. So, looking into all these requirements we find Incremental RAD model is best suited for our system because it enables the development team to create a fully functional system within very short period of time.

**8.4 Observation**

We have observed that our system that is Inventory Management and Costing would be of immense help to the client as currently everything is done manually, which results in a lot of time consumption, is error prone and also increases economic burden in the form of salaries paid to the workers. Moreover, such a manual system of managing inventory is quite unstructured. Our system would be efficient, accurate and easy to use.

**8.5 Determining Project Feasibility**

The feasibility study is not a full-blown systems study. Rather, the feasibility study is used to gather broad data to make a decision on whether to proceed with system study. System project feasibility is assessed in three principal ways:

* Economically
* Technically
* Operationally
* Economic Feasibility

The organization has evaluated cost of software and hardware required for the system including the storage of data. The benefits expected from the system are studied to assess the reduced cost due to the new system.

**Technical Feasibility:**

Organization has shown willingness to purchase all hardware and software tools which we recommend to successfully implement the system. Hence technically there are no limitations for the development of the system. Thus, the project is technically feasible.

**Operational Feasibility:**

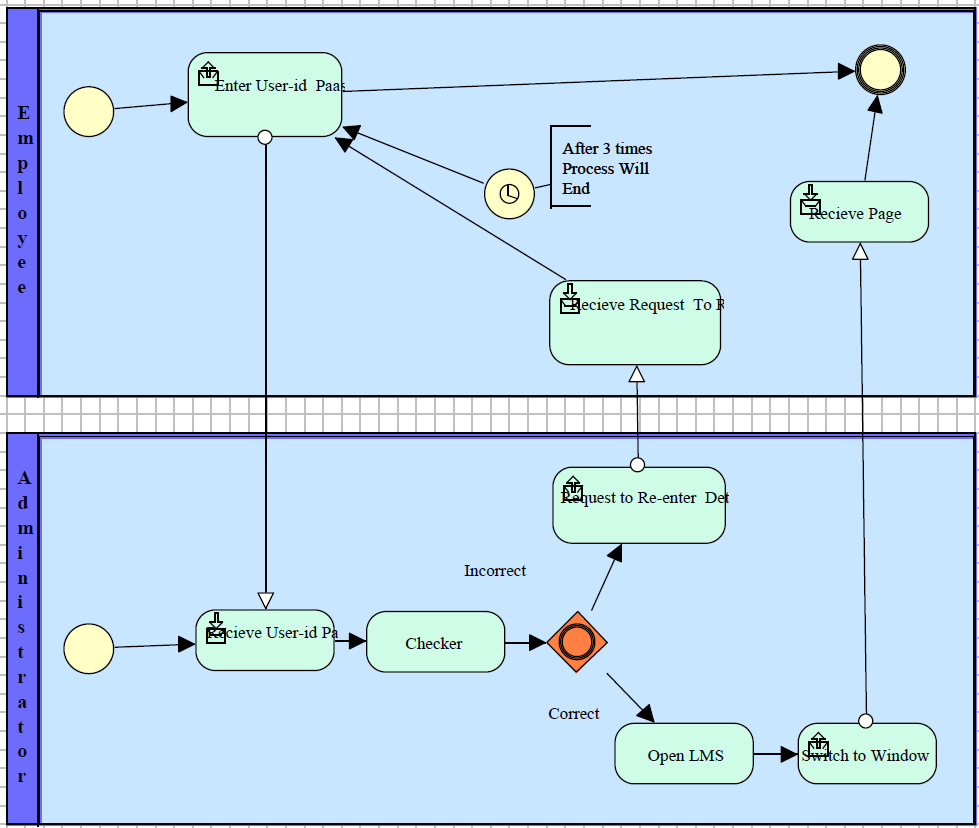
Operational feasibility is dependent on the humans who will be using the software once it’s ready and installed for use. The software will have a user-friendly interface which will be much convenient as compared to the current manual procedure. Thus, the project is operationally feasible.

1. **DESIGN**

**9.1 BPMN**

The Business Process Modeling Notation (BPMN) specification provides a graphical notation for specifying business processes in a Business Process Diagram (BPD).

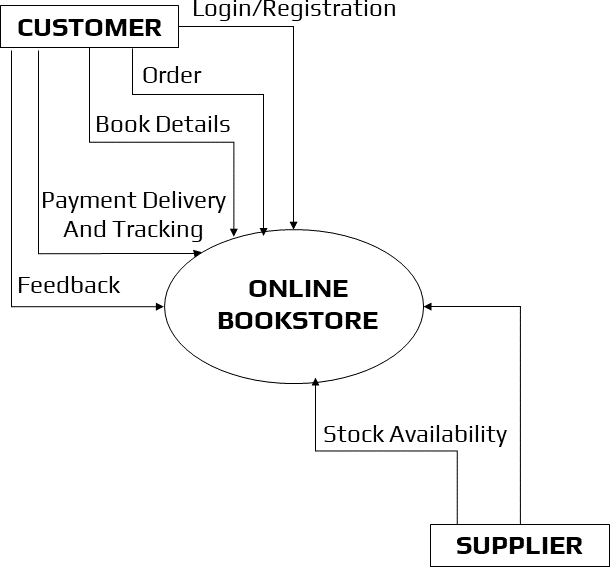
**Login**

****

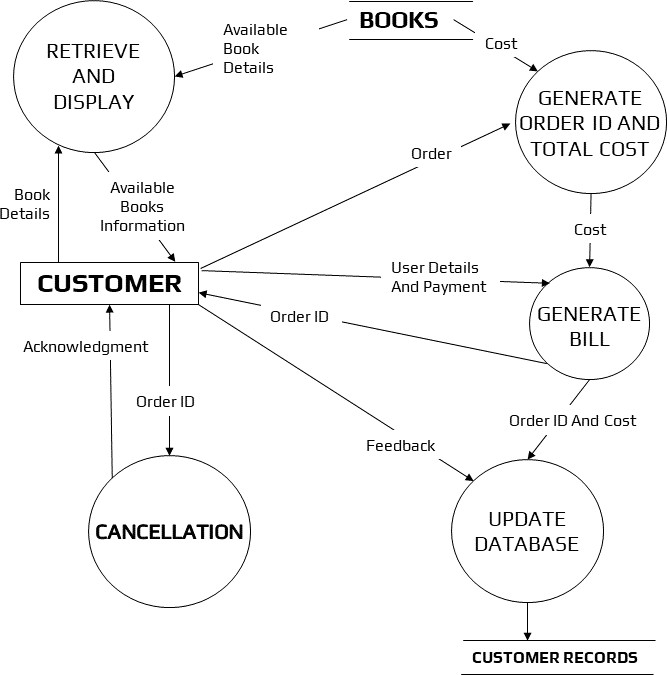
**9.2 DATA FLOW DIAGRAM (DFD)**

The data flow diagram is a graphical representation of the flow of data in an information system. It is capable of depicting incoming data flow, outgoing data flow and stored data. The DFD does not mention anything about how data flows through the system. There is a prominent difference between DFD and Flowchart. The flowchart depicts a flow of control in program modules. DFDs depict the flow of data in the system at various levels. DFD does not contain any control or branch elements.

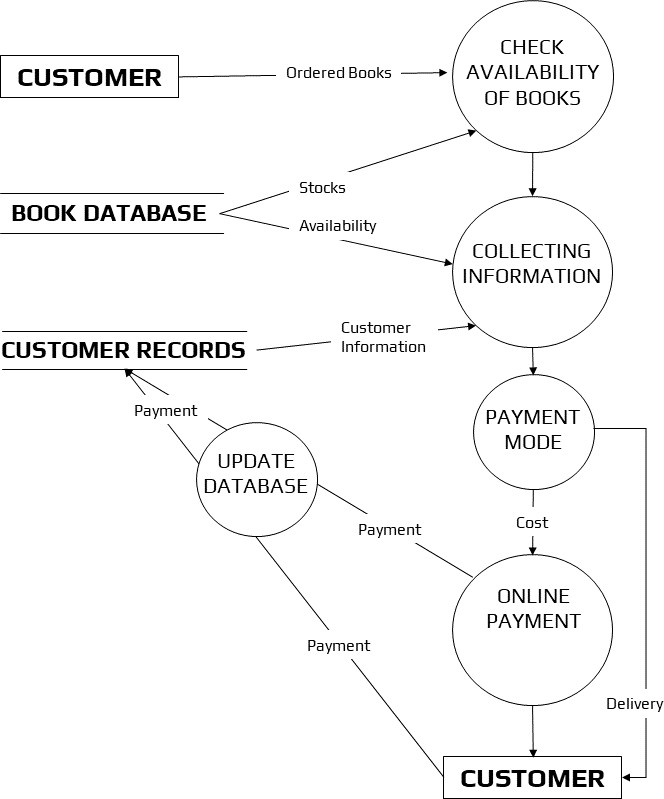
# Zero Level Data Flow Diagram (0 Level DFD):



# First Level Data Flow Diagram (1 Level DFD):



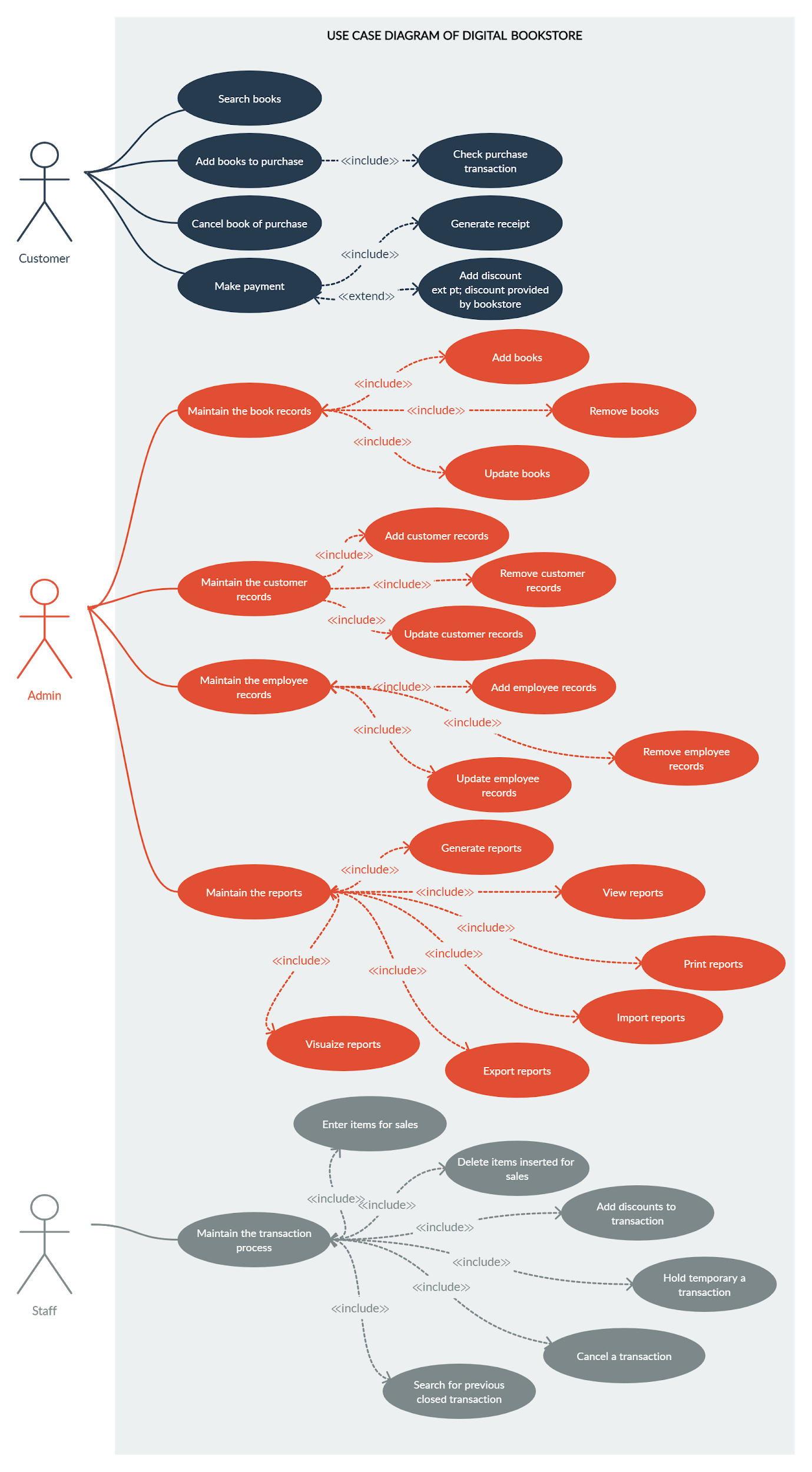
# Second Level Data Flow Diagram (2 Level DFD):



**9.3 USE CASE DIAGRAM**

A UML use case diagram is the principal form of system/software specifications for an undeveloped computer program. Use cases specify the expected behavior (what), and not the exact method of making it happen (how). Use cases once specified can be denoted by both textual and visual representation (i.e. use case diagram). A key concept of use case modelling is that it helps us design a system from the end user's perspective. It is an effective technique for communicating system behavior in the user's terms by specifying all externally visible system behavior.

A use case diagram is usually simple. It does not show the detail of the use cases:

1. It only outlines several of the connections between use cases, actors, and systems.
2. It does not show the order in which steps are performed to achieve the goals of each use case.

**9.4 CLASS DIAGRAM**

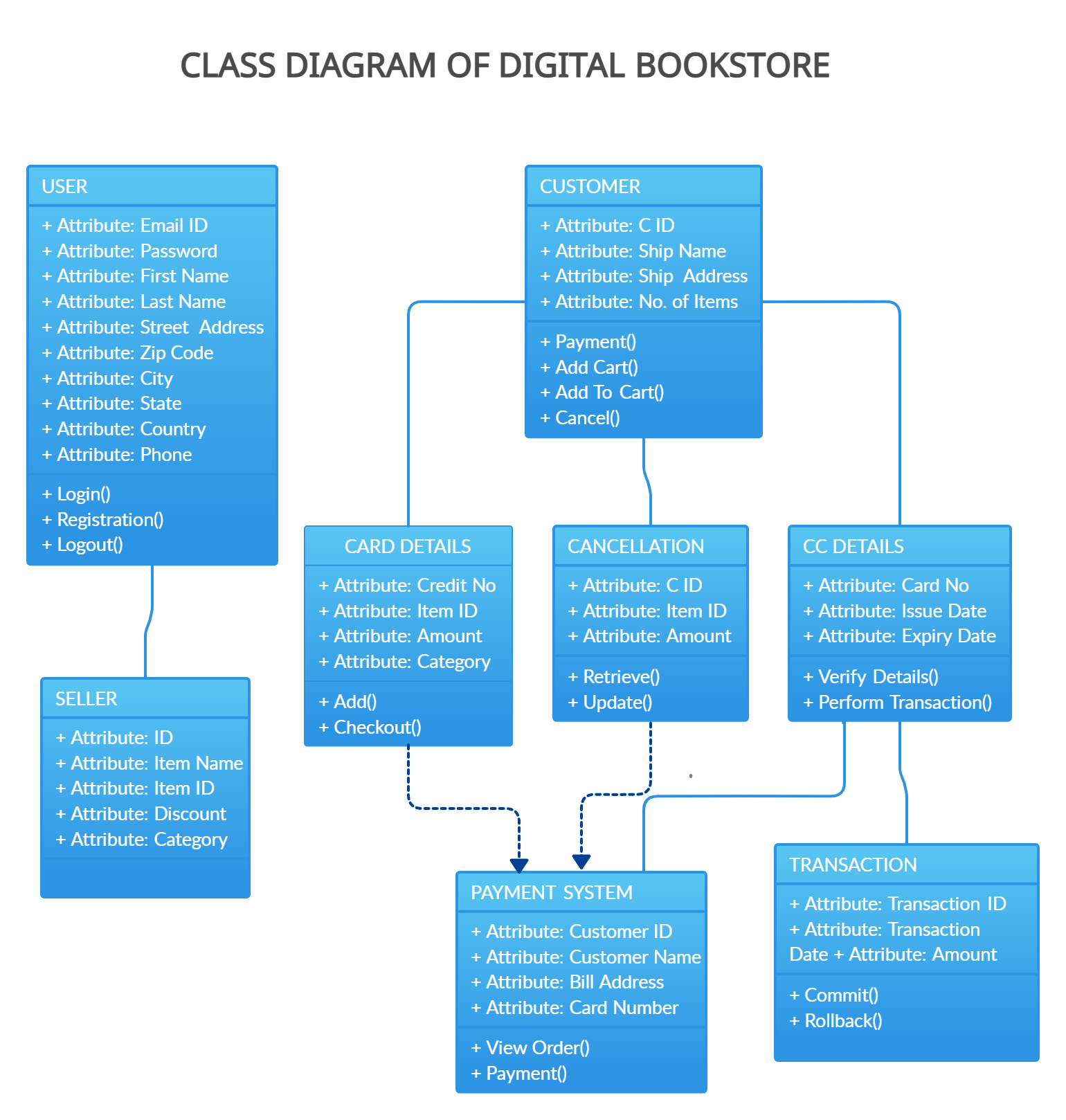
The class diagram is a static diagram. It represents the static view of an application. The class diagram is not only used for visualizing, describing, and documenting different aspects of a system but also for constructing executable code of the software application. The class diagram describes the attributes and operations of a class and also the constraints imposed on the system. Since they're the only UML diagrams that can be translated directly to object-oriented languages, class diagrams are frequently utilized in the designing of object-oriented systems. The class diagram shows a collection of classes, interfaces, associations, collaborations, and constraints.

**Purpose of Class Diagram**

The class diagram is used to represent the basic perspective of a system. Class diagrams are the only designs that can be highly associated with object-oriented languages and are thus generally applied throughout development. UML diagrams like activity diagrams, sequence diagrams can only give the sequence flow of the application, however, the class diagram is a bit different. That's the most widely used UML diagram in the computing world.

The class diagram's aim may be described as:

1. Design and development of a software's static view.
2. Describe the responsibilities of a system.
3. The base for component and deployment diagrams.
4. Forward and reverse engineering.

UML class diagrams are useful when modelling business data. By accurately modelling attributes and associations of class entities, we can easily map these class diagram specifications to entity beans with CMP. Class attributes map to abstract access methods for persistent fields, and association roles map to abstract access methods for relationship fields. Navigability determines whether relationship access methods appear in both related entity beans or just one. Furthermore, multiplicity notation determines the correct type for relationship fields, life cycle issues, and cascading delete characteristics.

**9.5 ACTIVITY DIAGRAM**

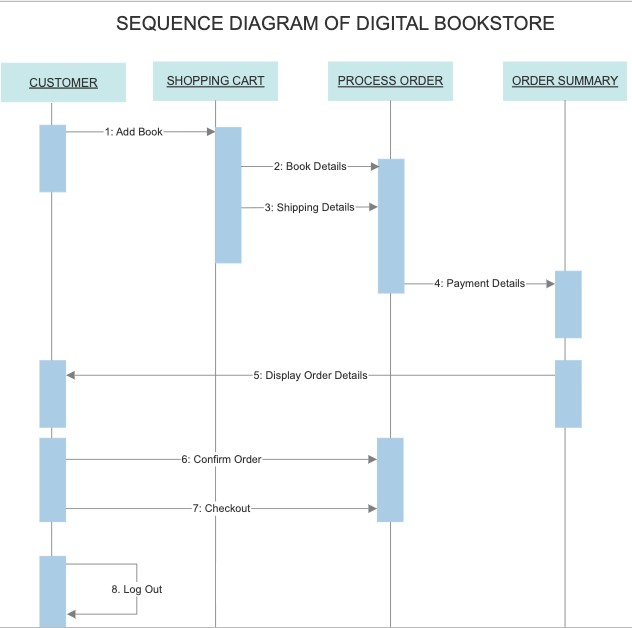
Another essential diagram in UML for describing the dynamic features of the system is the activity diagram. It is essentially a flowchart that represents the transition from one activity to another. The action can be defined as a system operation. The control flow is directed from one activity to the next. This flow might be linear, branching, or parallel in nature.

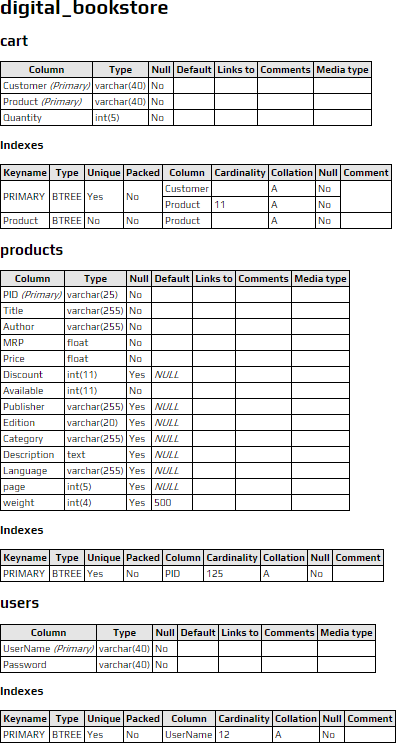


**9.6 SEQUENCE DIAGRAM**

The sequence diagram represents the flow of messages in the system and is also termed an event diagram. It helps in envisioning several dynamic scenarios. It portrays the communication between any two lifelines as a time-ordered sequence of events, such that these lifelines took part at the run time. In UML, the lifeline is represented by a vertical bar, whereas the message flow is represented by a vertical dotted line that extends across the bottom of the page. It incorporates the iterations as well as branching.

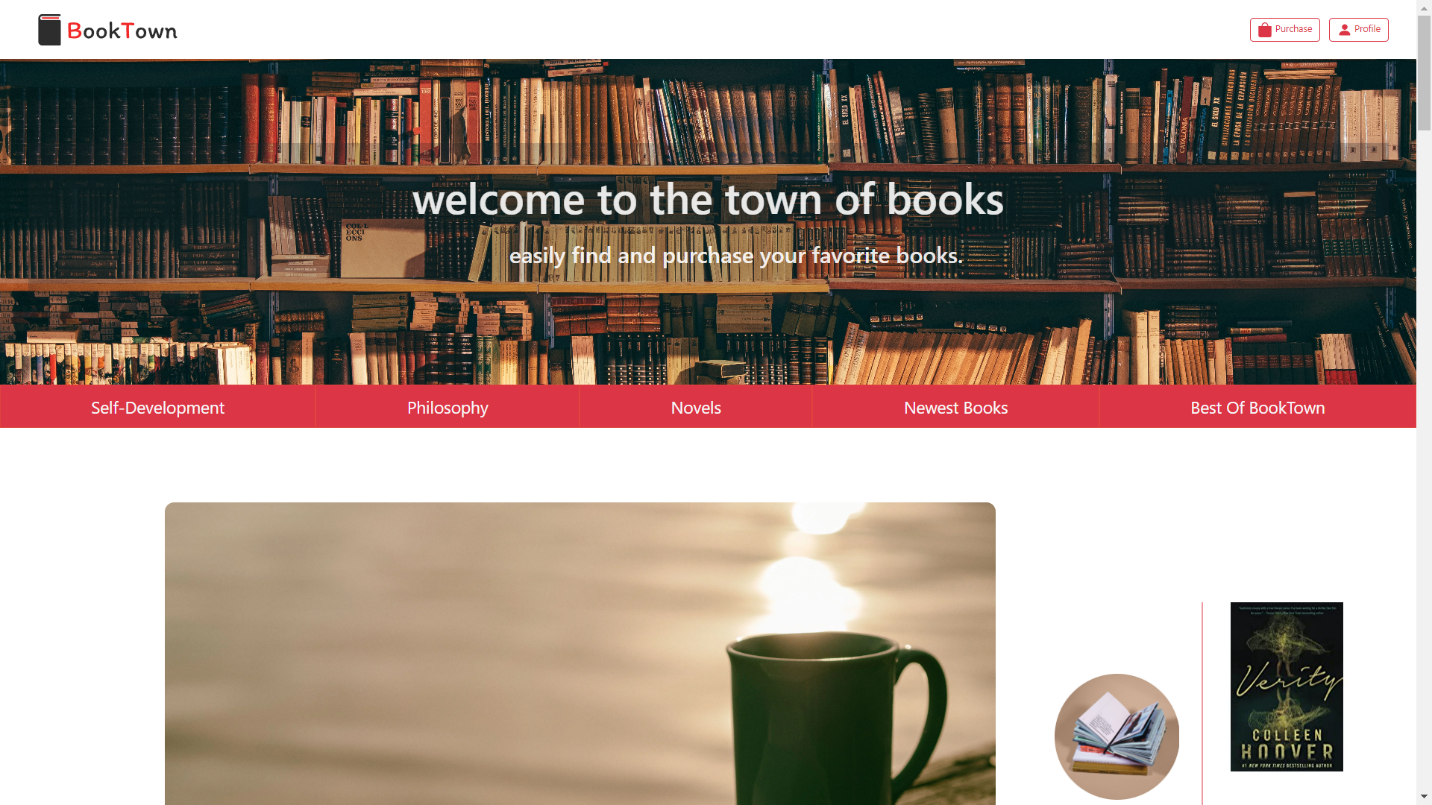
Purpose of Sequence Diagram

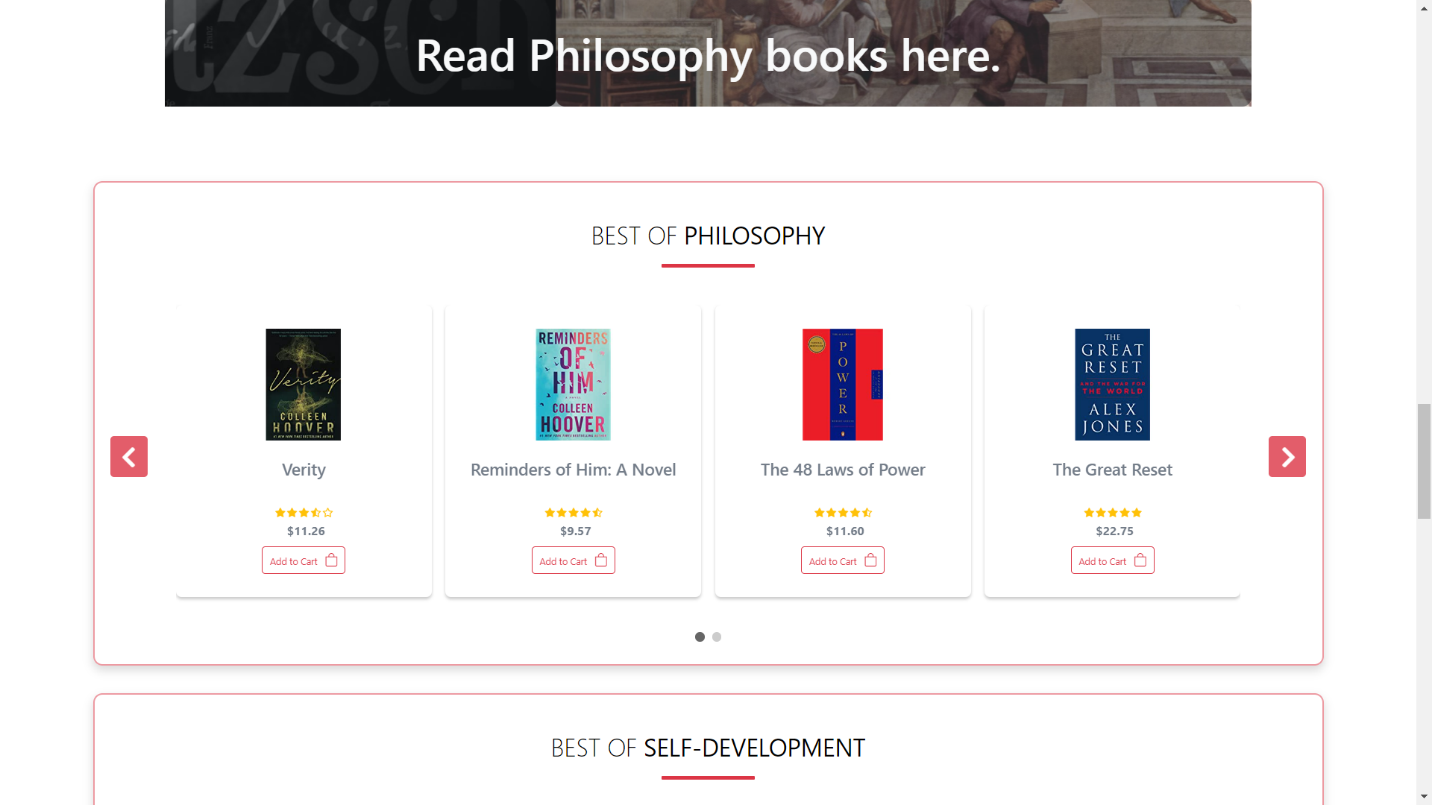
1. To model high-level interaction among active objects within a system.
2. To model interaction among objects inside a collaboration realizing a use case.
3. It either models’ generic interactions or some certain instances of interaction.

**10. DATABASE**

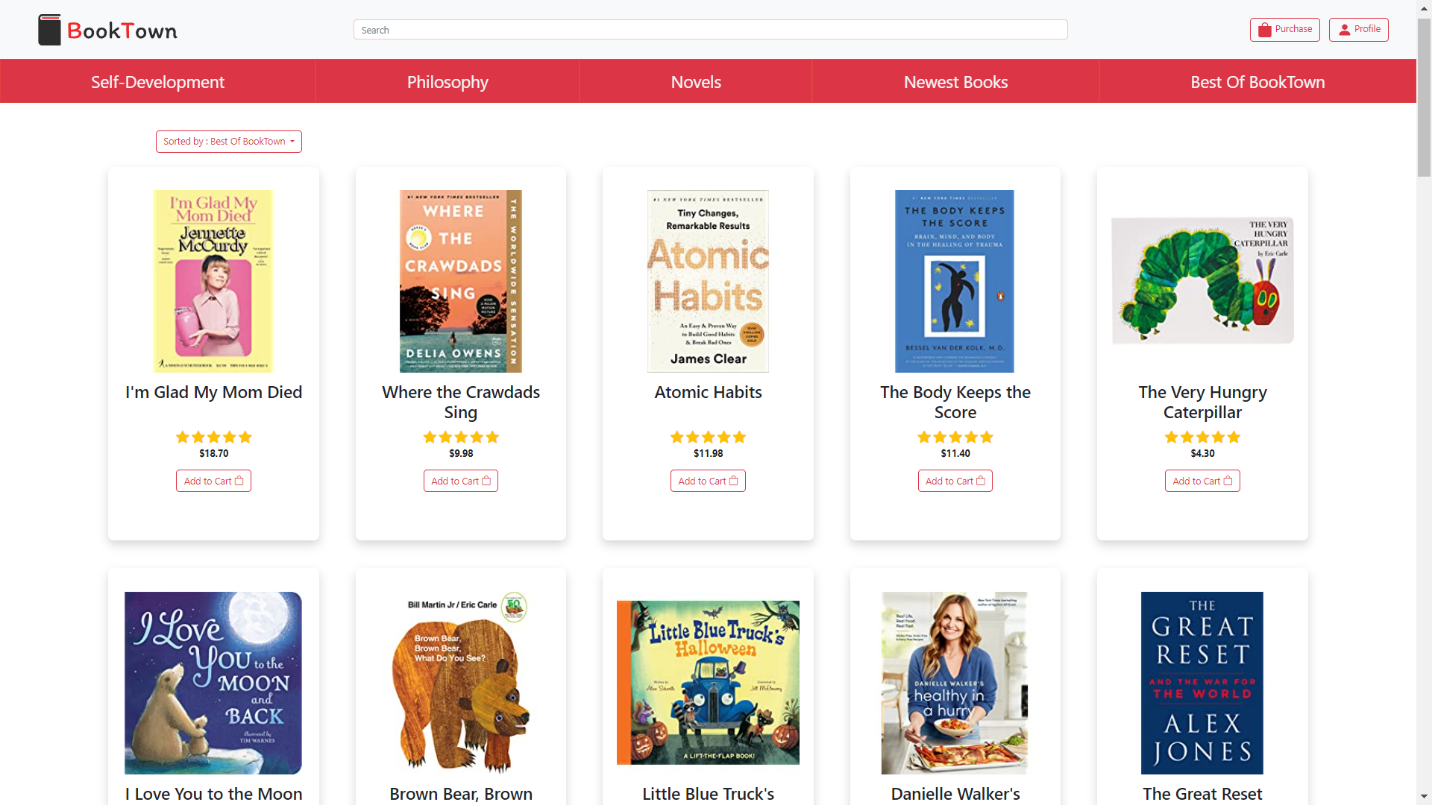
**11. SCREENSHOTS**

**Home Page**

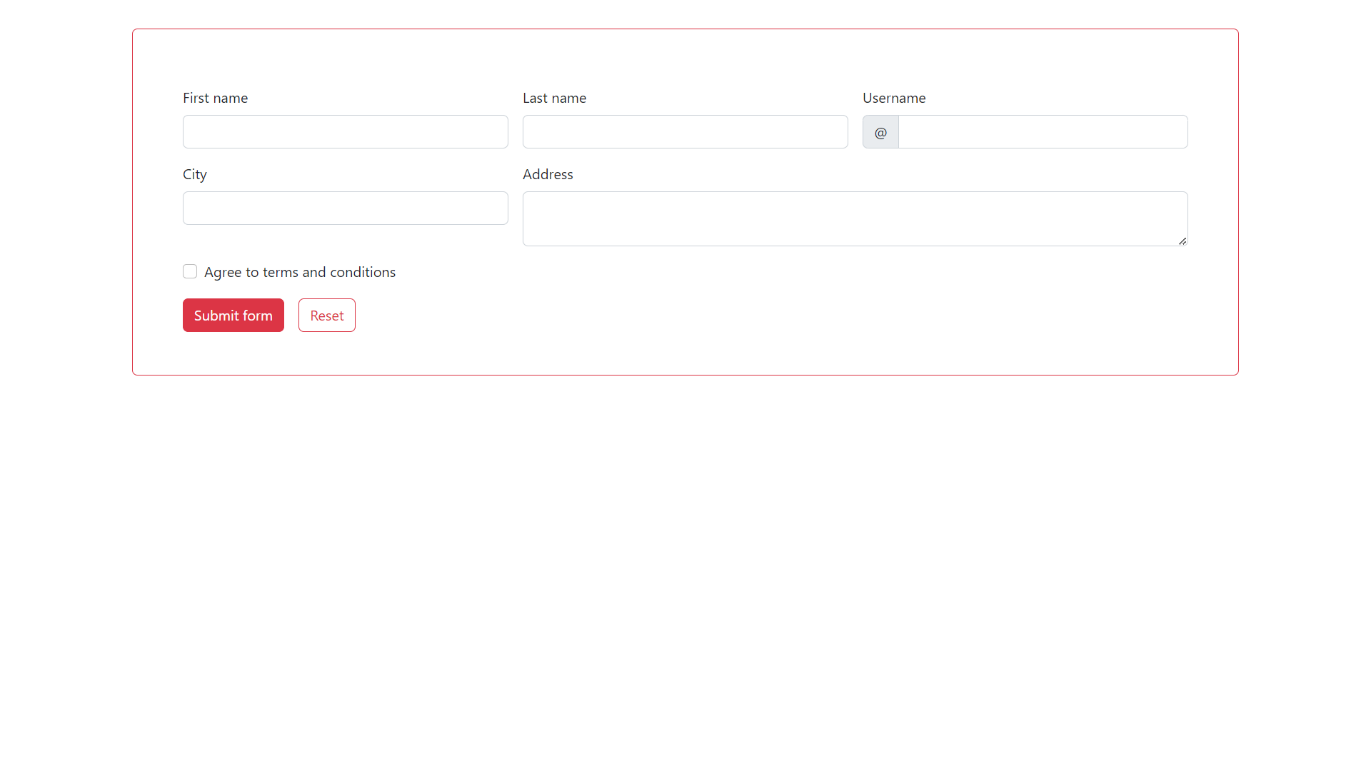




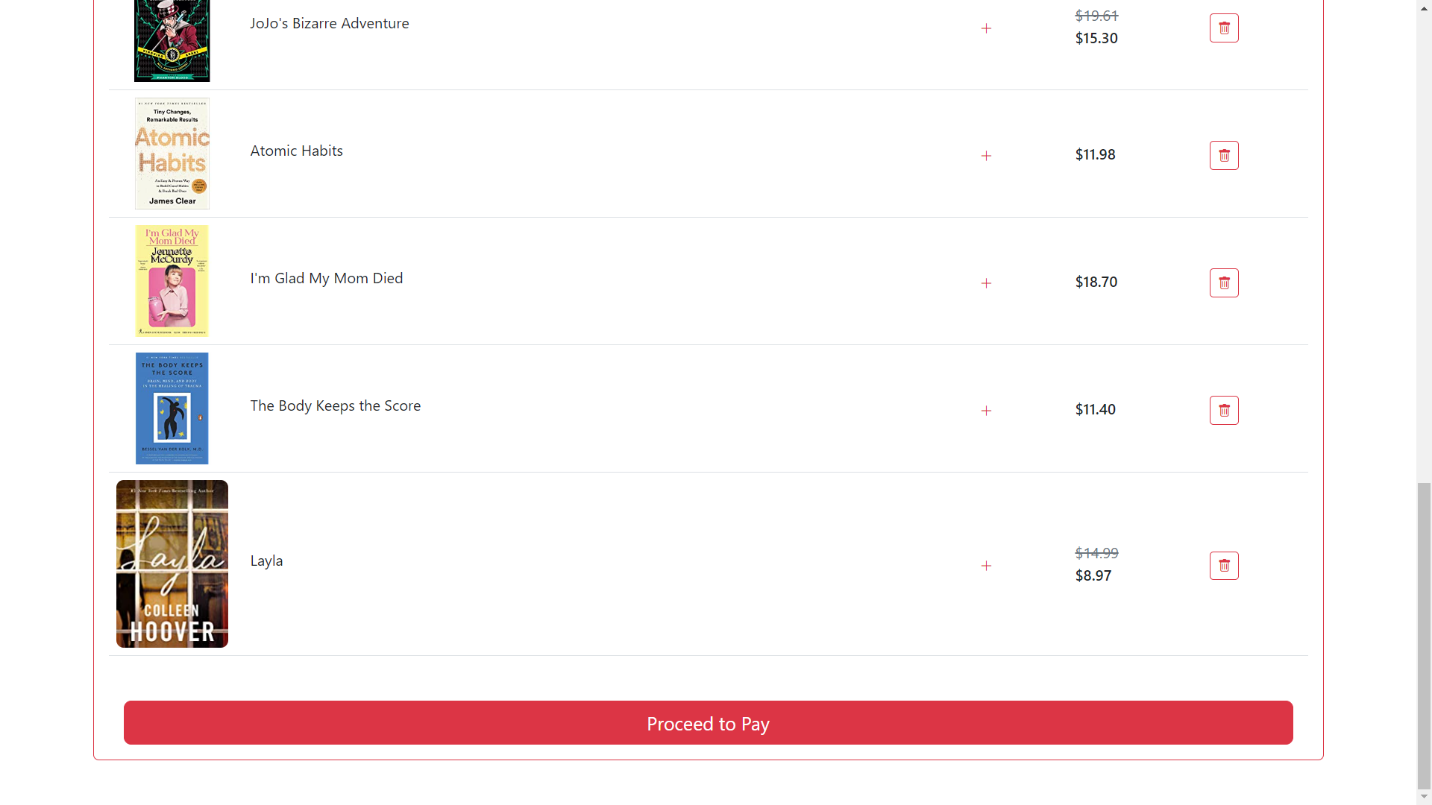
**Book Category Section**

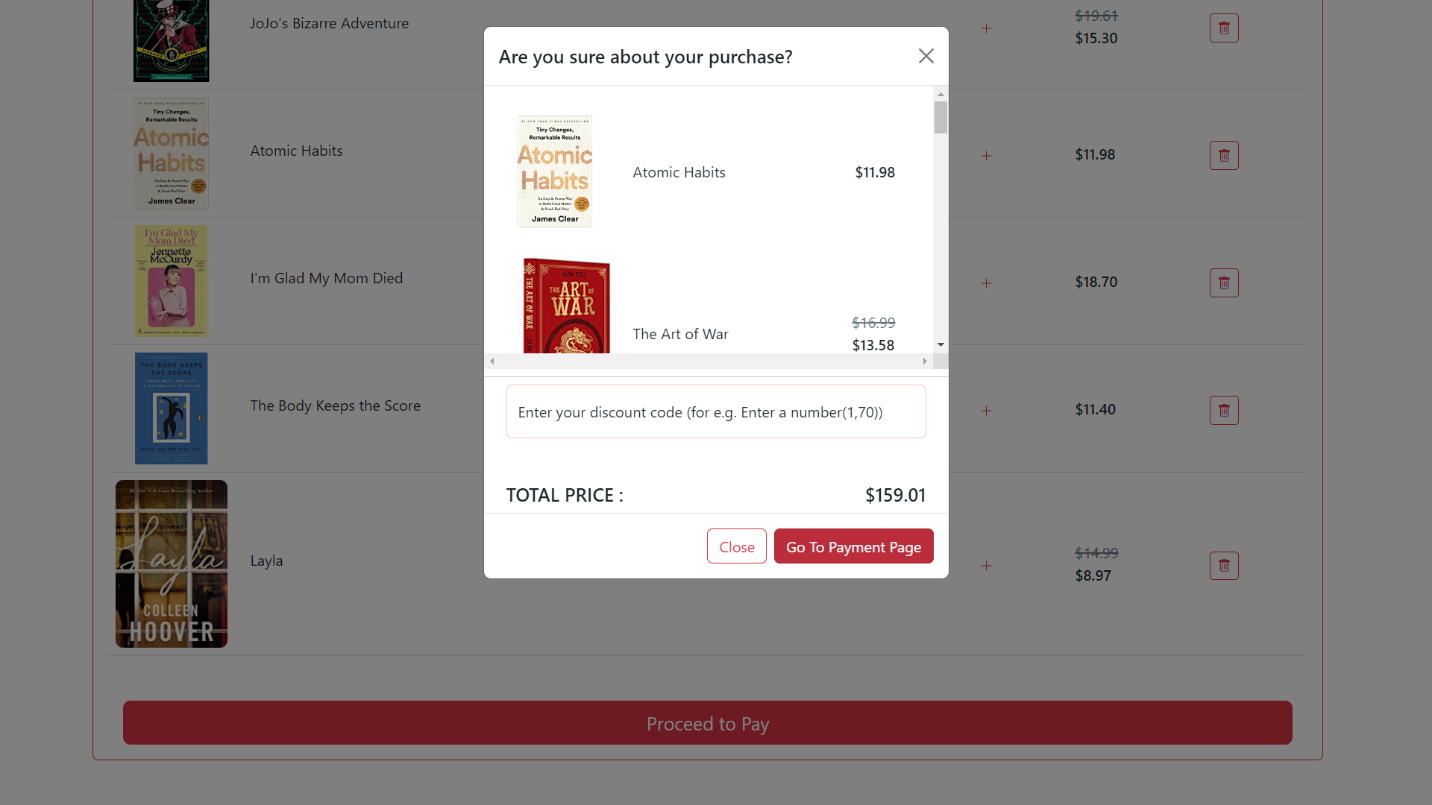


**Registration page**

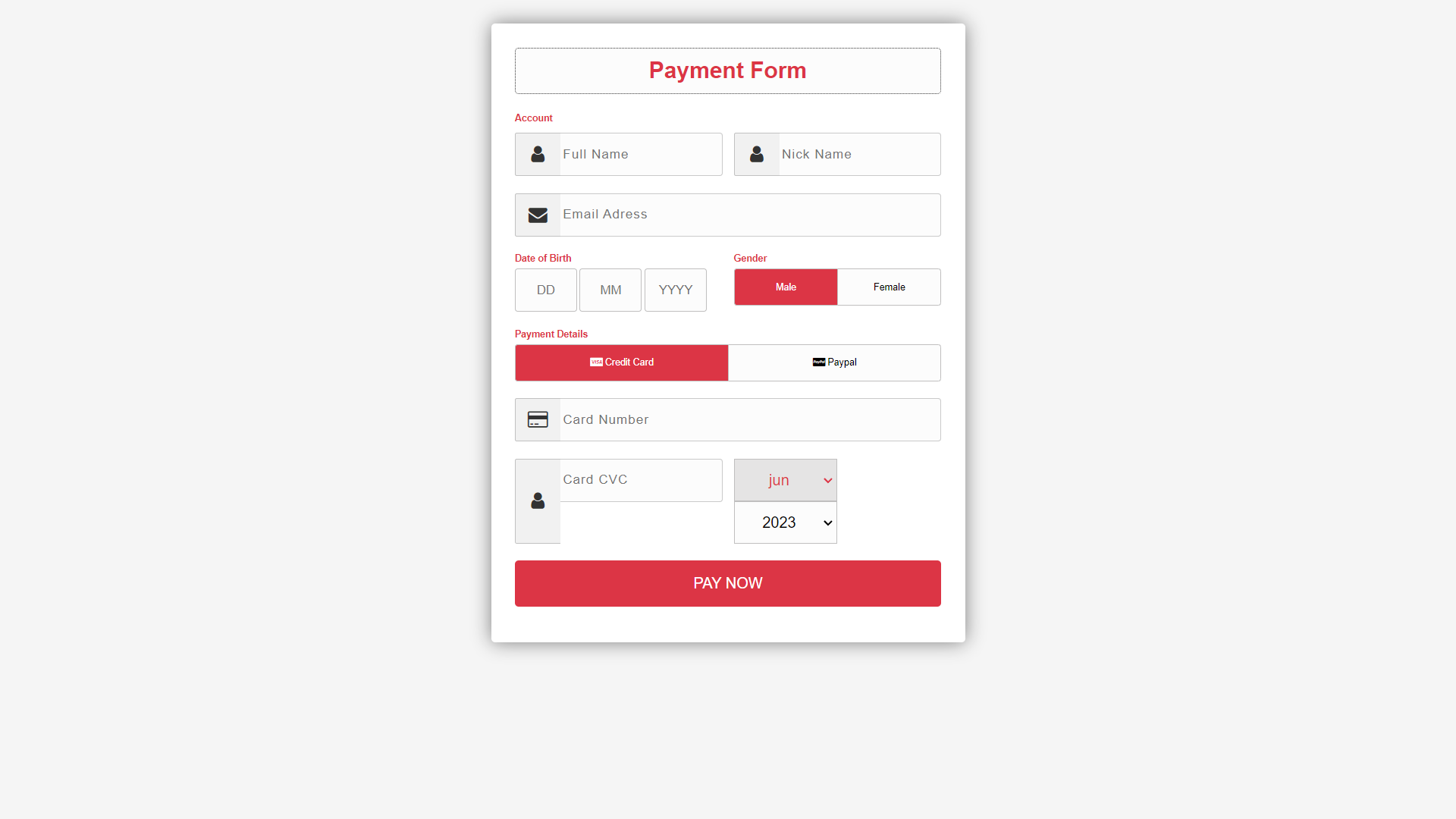


**Payment Section**





**Payment Form**



**12. TEST CASES**

**Purpose:**

The purpose of this document is to outline the testing strategy and approach for the online book purchasing website. This document will cover the different types of testing that will be conducted, the tools and resources required for testing, and the expected outcomes and deliverables of the testing process.

**Scope:**

The testing process will cover the entire website, including all front-end and back-end functionalities. The testing process will include manual and automated testing, as well as user acceptance testing (UAT) to ensure that the website meets the requirements and expectations of the end-users.

**Testing Types:**

The following types of testing will be conducted:

* Unit testing: Individual components of the website will be tested to ensure that they function correctly and meet the specifications.
* Integration testing: The integration of different components and modules will be tested to ensure that they work together seamlessly.
* System testing: The entire system will be tested to ensure that all functionalities and features work as intended.
* Acceptance testing: Users will be invited to test the website and provide feedback on its usability, functionality, and overall user experience.

**Testing Approach:**

The following approach will be taken for testing:

* Test cases will be created for each requirement, with expected outcomes and acceptance criteria defined.
* Automated testing tools will be used for regression testing and to speed up the testing process.
* The website will be tested on different browsers and devices to ensure cross-browser and cross-device compatibility.
* Load testing will be conducted to ensure that the website can handle a large number of concurrent users and orders.
* UAT will be conducted to gather feedback from end-users and ensure that the website meets their expectations.

**Testing Deliverables:**

The following deliverables will be produced as part of the testing process:

* Test cases: A comprehensive set of test cases for each requirement, with expected outcomes and acceptance criteria defined.
* Test results: Detailed reports on the results of each testing phase, including any issues or bugs identified and their severity.
* User feedback: Feedback gathered from end-users during the UAT process, including suggestions for improvements and areas of concern.
* Final testing report: A final report summarizing the testing process, outcomes, and any remaining issues or bugs.
* Overall, the testing process will be thorough and comprehensive, with a focus on ensuring that the website meets the requirements and expectations of the end-users.

**13. CONCLUSION**

The transition from buying written books in bookshops to ordering them online or even simply digital versions has had a significant impact on the industry, including retailers and libraries, as well as the general public throughout the globe. We present an application developed using software engineering methodologies. Digital Bookstore allows the users to buy as well as review books online. Consumers can login and search for their books, whether it is available or out of stock. Users can also give feedback. We have implemented and tested the web application to satisfy the user specifications.

**14. BIBLIOGRAPHY**

During the development of E-learning project, we have gone through the following websites:-

* Blueprint Foundry. (n.d.). BPF 3.0. Retrieved from https://blueprintfoundry.com/bpf-3-0/
* BPMN. (n.d.). Business Process Model and Notation. Retrieved from https://www.bpmn.org/
* Object Management Group. (2017). Unified Modeling Language (UML). Retrieved from https://www.omg.org/spec/UML/2.5.1/PDF
* Bootstrap. (n.d.). Bootstrap - The world’s most popular front-end open source toolkit. Retrieved from https://getbootstrap.com/
* JavaScript. (n.d.). Retrieved from https://www.javascript.com/
* JavaTpoint. (n.d.). Retrieved from https://www.javatpoint.com/
* Wikipedia. (n.d.). Retrieved from https://en.wikipedia.org/
* W3Schools. (n.d.). Retrieved from https://www.w3schools.com/
* Google. (n.d.). Retrieved from https://www.google.com/
* BPF 3.0 (Blue Print Foundry)

It is open-source software which is helpful for designing the modules of software. With the help of BPF we were able to design

* BPMN Diagrams
* UML Diagrams (use case, class, sequence, activity diagrams)