

INDIRA GANDHI NATIONAL OPEN UNIVERSITY

MAIDAN GARHI, NEW DELHI-110068



“eBOOK STORE-ONLINE COMPUTER BOOK STORE

Submitted by **Shrey Mishra**

Under the supervision of

Mr. Saurabh Misra (M.C.A)

Submitted to the School of Computer
and Information Science (SOCIS), IGNOU
in partial fulfillment of the requirements

for the award of the degree

Bachelor of Computer Application (B.C.A)

Content

| | |
|---|----|
| 1. Approval Proforma of Synopsis..... | 4 |
| 2. Certificate of Originality..... | 5 |
| 3. Guide's Biodata..... | 6 |
| 4. Synopsis..... | 9 |
| 5. Project Report..... | 34 |
| 5.1. Project Title..... | 35 |
| 5.2. Introduction and Objectives..... | 36 |
| 5.2.1. Introduction..... | 37 |
| 5.2.2. Objective(s)..... | 38 |
| 5.3. Technologies Survey..... | 39 |
| 5.4. System Requirement Specification | 47 |
| 5.4.1. Problem Definition..... | 47 |
| 5.4.2. Introduction to different modules..... | 48 |
| 5.4.3. Module Description..... | 49 |
| 5.4.4. Planning and Schedule..... | 53 |
| 5.4.5. Software Development Lifecycle..... | 54 |
| 5.4.6. Entity-Relationship Diagram..... | 57 |
| 5.4.7. Use-Case Diagram..... | 58 |
| 5.4.8. Data Flow Diagram..... | 59 |
| 5.4.9. Project Category..... | 62 |
| 5.4.10. Data Structure/Schemas..... | 63 |
| 5.4.11. Hardware and Software Requirements..... | 66 |

| | |
|---|-----|
| 5.5. Program Code..... | 67 |
| 5.6. Input/Output Screen..... | 116 |
| 5.6.1. Home Page view..... | 116 |
| 5.6.2. About Page view..... | 118 |
| 5.6.3. Book Page view..... | 120 |
| 5.6.4. Review Page view..... | 122 |
| 5.6.5. Contact Form Page view..... | 123 |
| 5.6.6. Privacy Policy Page view..... | 124 |
| 5.6.7. Refund Policy Page view..... | 125 |
| 5.6.8. Signup Form Page view..... | 126 |
| 5.6.9. Login Page view..... | 127 |
| 5.6.10. Forget Password Page view..... | 127 |
| 5.7. Functionality of the website..... | 128 |
| 5.7.1. Sign up..... | 128 |
| 5.7.2. Login..... | 131 |
| 5.7.3. Contact..... | 135 |
| 5.7.4. Feedback..... | 136 |
| 5.8. Testing..... | 139 |
| 5.9. Implementation of Security..... | 144 |
| 5.10. Future Scope and further enhancement..... | 147 |
| 5.11. Bibliography..... | 151 |

CERTIFICATE OF ORIGINALITY

This is to certify that the project report entitled "**eBOOK STORE- ONLINE COMPUTER BOOK**" submitted to **Indira Gandhi National Open University** in partial fulfilment of the requirement for the award of the degree of **BACHELOR OF COMPUTER APPLICATIONS (BCA)**, is an original work carried out by **Mr. Shrey Mishra** Enrolment No.: **2004267093** under the guidance of

Mr. Saurabh Mishra

The matter embodied in this project is a genuine work done by the student and has not been submitted whether to this University or to any other University / Institute for the fulfilment of the requirement of any course of study.

Shrey Mishra

14 March' 2023
Signature of the student

Saurabh Misra

14 March' 2023
Signature of the Guide

Name and address of the student:

Shrey Mishra
194-G Pardewanpur, Lal bangla
Kanpur, Uttar Pradesh - 208007

Enrollment No.: **2004267093**

Name, Designation, and address of the Guide:

Saurabh Misra
(Technical Project Leader)
Hyde park, Sector-78
Noida-201307

SYNOPSIS

1. PROJECT TITLE

“**eBOOK STORE - ONLINE COMPUTER BOOK STORE**” is title of my project. It is designed to keep objective to manage eBooks online. The title is proposed because this system help to sale purchase and manage eBooks through our system.

This system provides very innovative way to support to the customer.

User have two choices either they purchase book as a soft copy format or hard copy format.

With the advancement in technologies, demands of soft copy is increasing day by day. eBook can be easily used in any devices like smart phones, Tab, Computer System, and Laptop, iPad etc. Best part is that, carry and storage of eBook is very easy. For environment support it is very good steps to use eBooks in our daily life.

So, for environmental support as well as user support we decide to develop this system with title “**eBook – ONLINE COMPUTER BOOK STORE**”.

2. INTRODUCTION & OBJECTIVE

2.1 Introduction

This online platform provides the visitor, customer and administrator an easy and efficient way to buy and manage books in online. By “**eBOOK STORE - ONLINE COMPUTER BOOK STORE**” we try to support environment as well as provide full support to user. They can carry lots of book in their phone and study anytime. So, this system is very effective and useable in future scenario.

2.2 Objective

“**eBOOK STORE - ONLINE COMPUTER BOOK STORE**” is designed

to fulfill many objectives such as:

- Facilitates easy shopping online
- Customer can choose either soft copy or hard copy of selected book
- They also have facility to buy both type of copy or book
- Provides information about the products in categories
- Provides e-mail facility for future correspondence
- Provides Backup facility
- Provide online storage facility
- Software provides the following facilities to the software manager
- Manages the Transportation of hard copy of Book
- Look after the payment method

3. CATEGORY OF PROJECT

The Project is a web- based product, which will be used by multiple users i.e. the administrator, registered user and guest. Administrator will be provided with the interface, with the help of which administrator can manage accounts and UI.

4. TECHNOLOGIES SURVEY

“EBOOK STORE - ONLINE COMPUTER BOOK STORE” is middle level project and after seeing all the available technologies today, we chose the following:

- ★ **FRONT END INTERFACE** : HTML, CSS, JS
- ★ **BACK END INTERFACE** : MONGODB
- ★ **INTERNET TECHNOLOGY** : BOOTSTRAP, EXPRESSJS
- ★ **INTERFACE USED** : JAVASCRIPT
- ★ **WEB SERVER** : NODEJS

FRONT END - HTML,CSS and JS are the most widely used internet technology

- Easy Integration
- Short Time to Market
- Flexibility
- Availability of Resources

BACK END (mongoDB): This is middle sized application. Although MySQL, Oracle, MySQL and other database are good for using backend interface but I choose mongoDB for backend database.

WEB Server (NODEJS): We choose NODEjs as a web server because our project is middle level application and this web server properly supports this system.

OTHER Technology(HTML, JavaScript, CSS): Other technology used for client side scripting and interactive designing. HTML, JavaScript and Ejs will be used to develop this system.

5. PROBLEM DEFINITION

"EBOOK STORE - ONLINE COMPUTER BOOK STORE" will enable the customer to easily get involved in sale, purchase and management of the book in soft copy as well as hard copy format. This system provides many benefits to the society as well as our environment. But to develop this proposed we to faces many problem and solve this in effective way.

Following are some problem area which is identified to develop this system and we have to solve that, they are:

- Login id creation and management
- Book details management in both format (Soft & Hard Copy)
- Stock Management
- Customer management
- Seller Management
- Feedback Management
- Software and web management
- Transportation management in case of hard copy
- Payment management
- Copyright management
- Software awareness management

6. REQUIREMENT SPECIFICATION

After analysis and study of this system we have decided to manage module which is mentioned below to support and develop this system properly. The modules of proposed system are:

- 1) Customer Module
- 2) Seller Module
- 3) Administrator Module
- 4) Book (Product)Module
- 5) Feedback Module
- 6) Payment Module

These above modules help to solve above mentioned problem and get final objective which is decided to achieve the goal for the proposed system.

7. MODULE DESCRIPTION & PROCESS

1) Customer Module: This module is designed to manage the customer properly.

By using this module customer registered with this system and perform eBook shopping and other activities on the system.

Process Logic:

- Customer visited on website
- They register with system
- System provides a valid id and password
- System confirms and verifies the user
- After verification they are forwarded to shopping module

2) Seller Module: This module is designed for seller of this system.

The seller also registered with system and provides an interface with the help of administrator to manage sellers.

Process Logic:

- Seller registers with system
- After registration administrator provide seller id to the seller
- Sellers send and upload product details
- And help customer in providing product and support
- Solve customer query

3) Administrator Module: This module is designed for administrator only. They are responsible for managing the entire system. Manage customer as well as seller of this system and provide security to the system.

Process Logic:

- Administrator is responsible for overall execution of the project
- They create customer id and seller id
- Verify the customer and seller
- Provide forgotten password to the customer and seller if required
- Making SEO based website for “COMPUTER BOOK STORE”
- Manage the database properly
- Manage validation and connection with full security

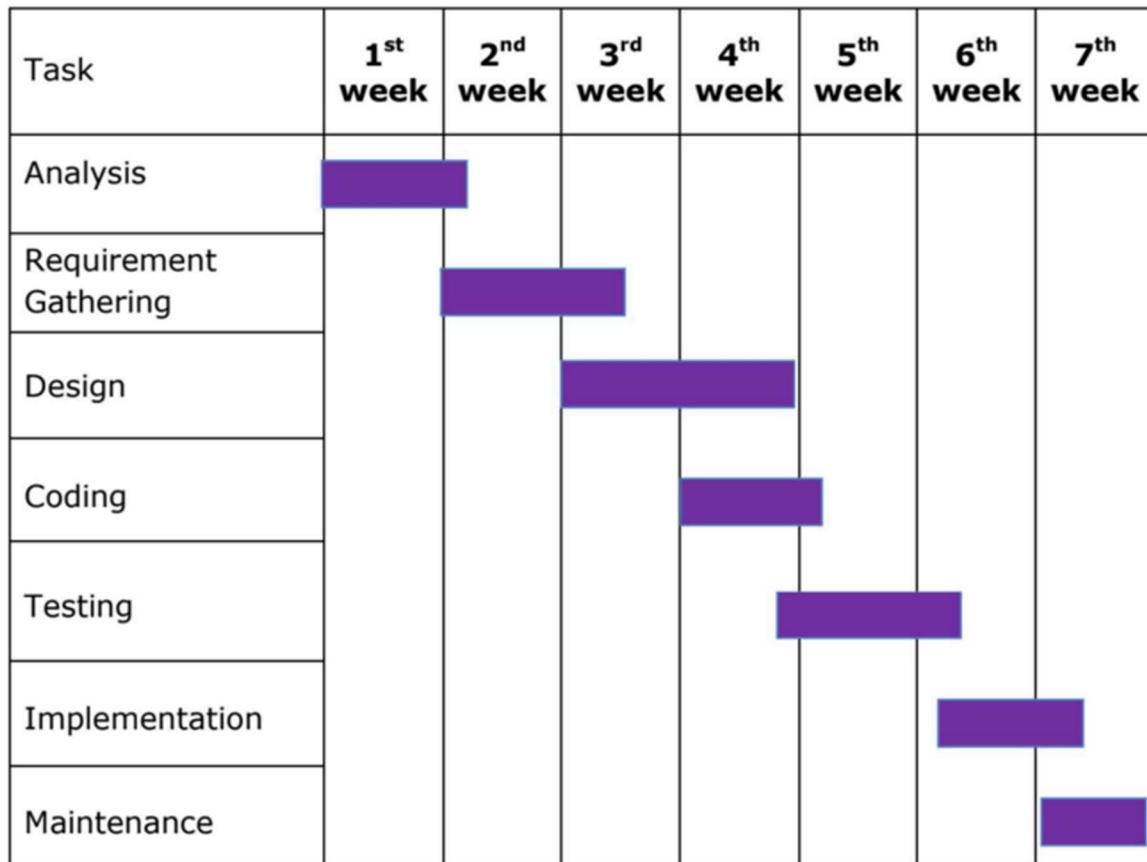
4) Book (Product) Module: This module is used for book and their stock management. Each product has complete details with system. We integrated it in future.

5) Feedback Module: This module is used for feedback purpose. With the help of this module customer and seller share their valuable feedback. On the basis of this feedback administrator manage and provide new approach to manage this system.

6) Payment Module : It is used to manage the entire process related to payment and its collection.

8. PLANNING AND SCHEDULING

A Gantt chart is a type of bar chart, developed by Henry Gantt in the 1910s, that illustrates a project schedule. Gantt charts illustrate the start and finish dates of the terminal elements and summary elements of a project.



9. SYSTEM REQUIREMENT SPECIFICATION

System Requirements Specification is most important technique to gather the software and hardware requirement. Following are hardware and software specification which required for smooth functionality of this “**eBOOK STORE-ONLINE COMPUTER BOOK STORE**”.

Hardware:

- Processor – Intel core i3
- Ram – 1GB
- Hard Disk – minimum 80Gb /250GB
- Keyboard and mouse – USB / PS2

Software:

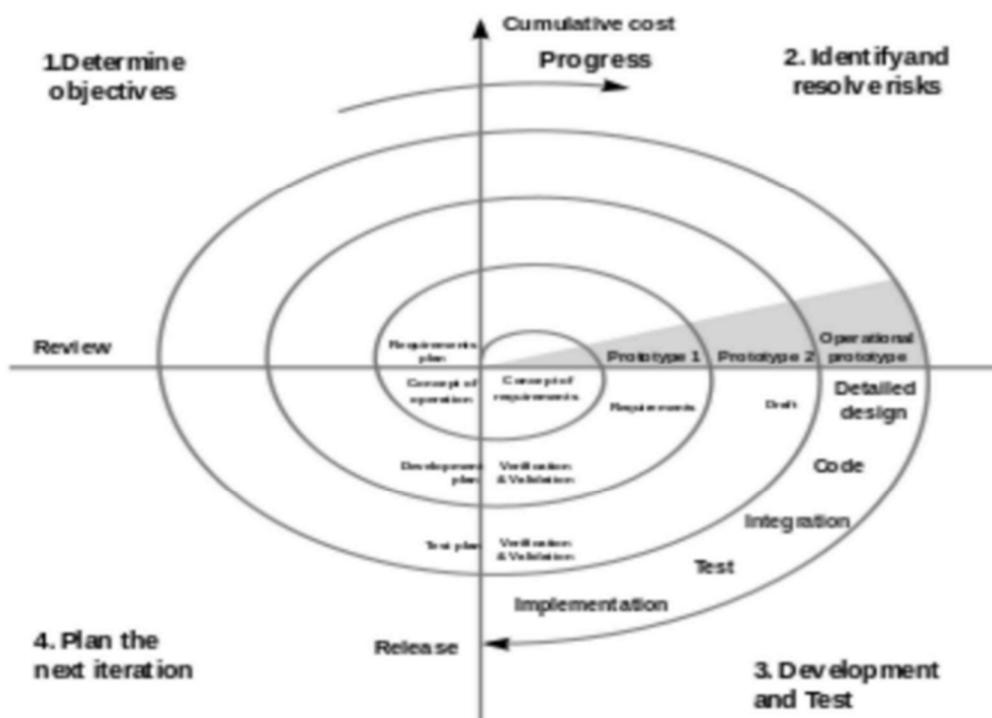
- OS – windows7/ XP/ Windows 2008
- Front End - HTML, CSS, JS
- Database - MONGODB
- Web Browser – IE/Chrome
- Scripting – Java Script
- Web Server – NODEJS

10. SYSTEM DEVELOPMENT LIFE CYCLE

To manage this, a number of system development life cycle (SDLC) models present such as

- Waterfall Model
- Build and Fix Model
- Spiral Model
- Prototype Model
- Interactive Enhancement Model
- RAD Approach

The more efficient model is **Spiral Model** so I choose Spiral Model to develop this project.



11. ENTITY RELATIONSHIP (ER) DIAGRAM

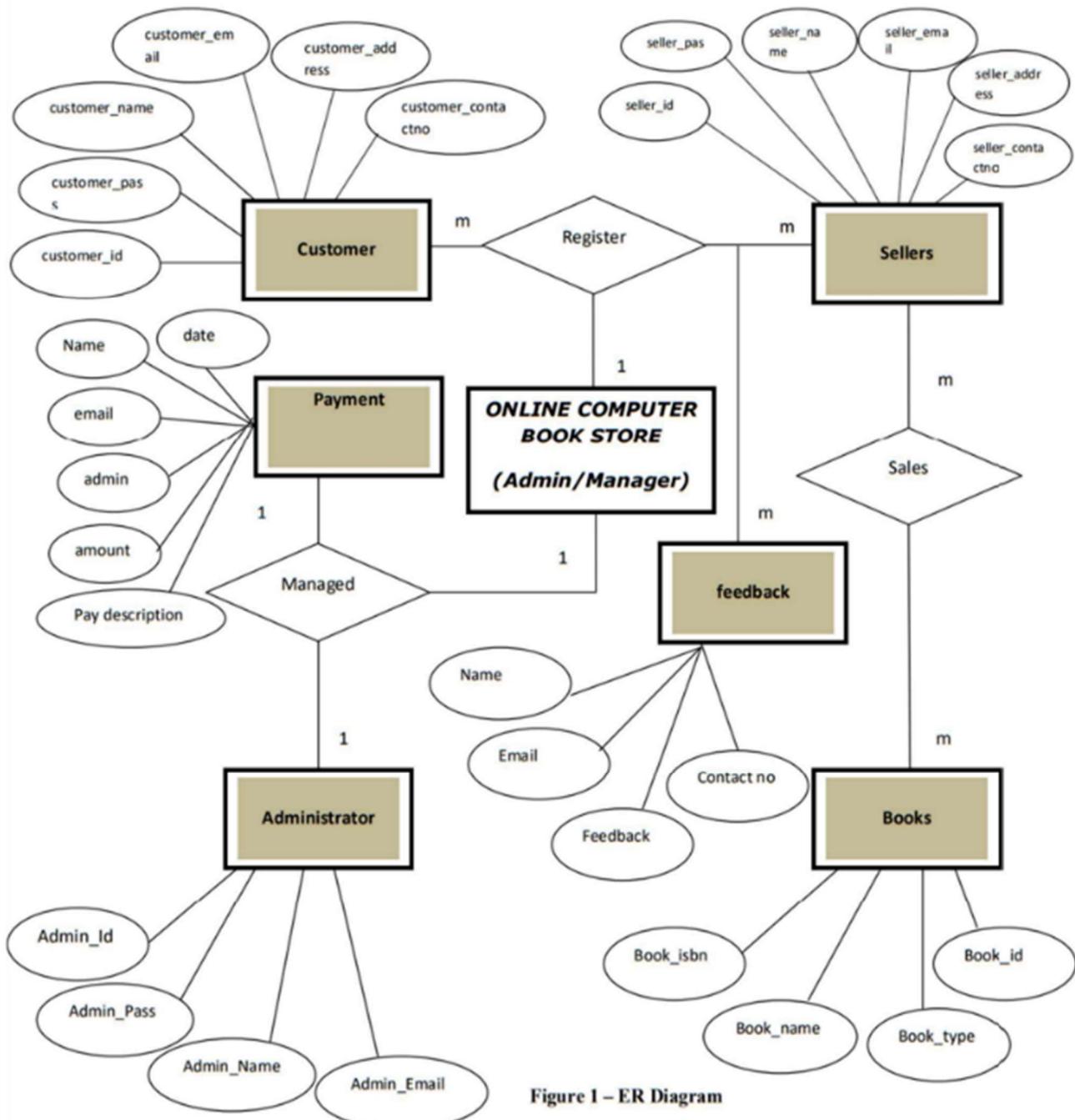


Figure 1 – ER Diagram

12. USE CASE DIAGRAM

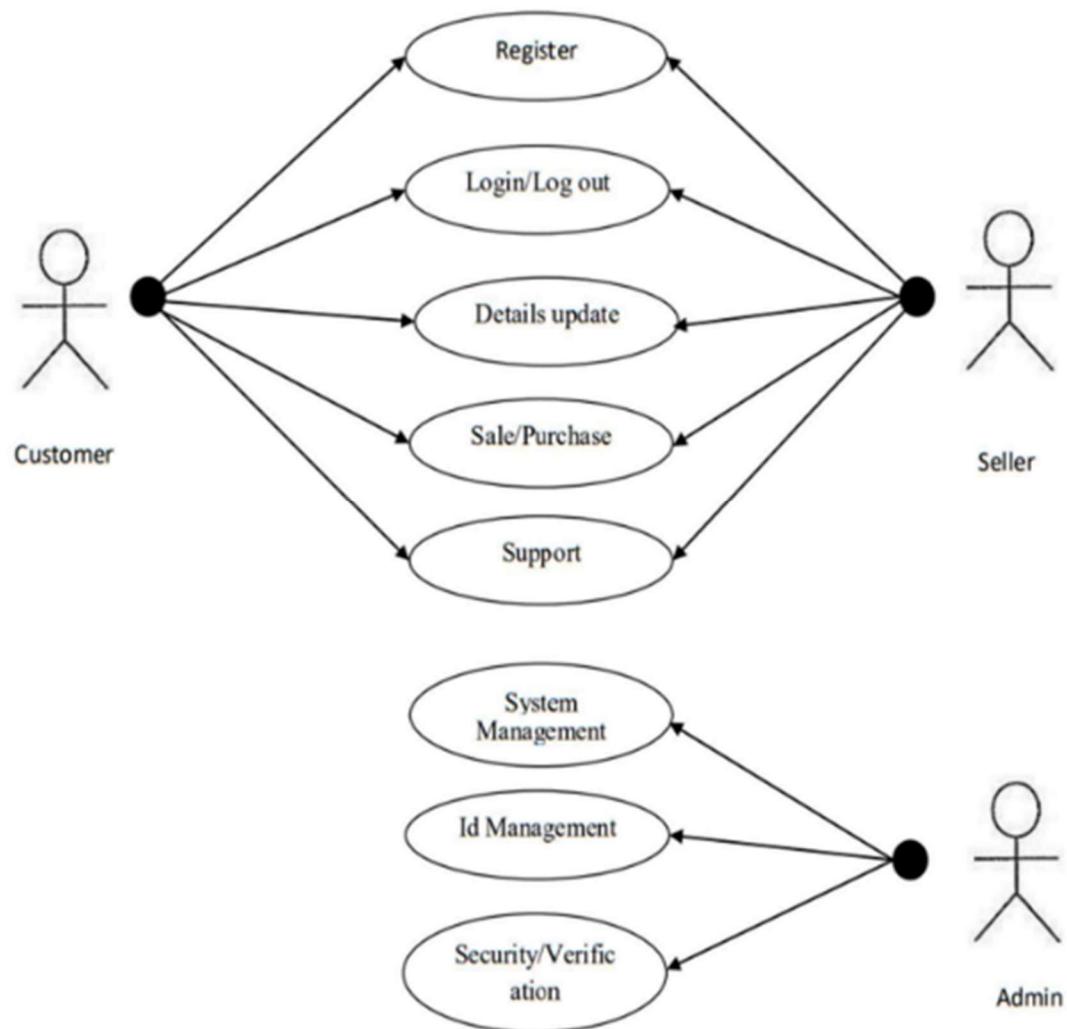


Figure 2 – Use Case Diagram

13. DATA FLOW DIAGRAM (DFD)

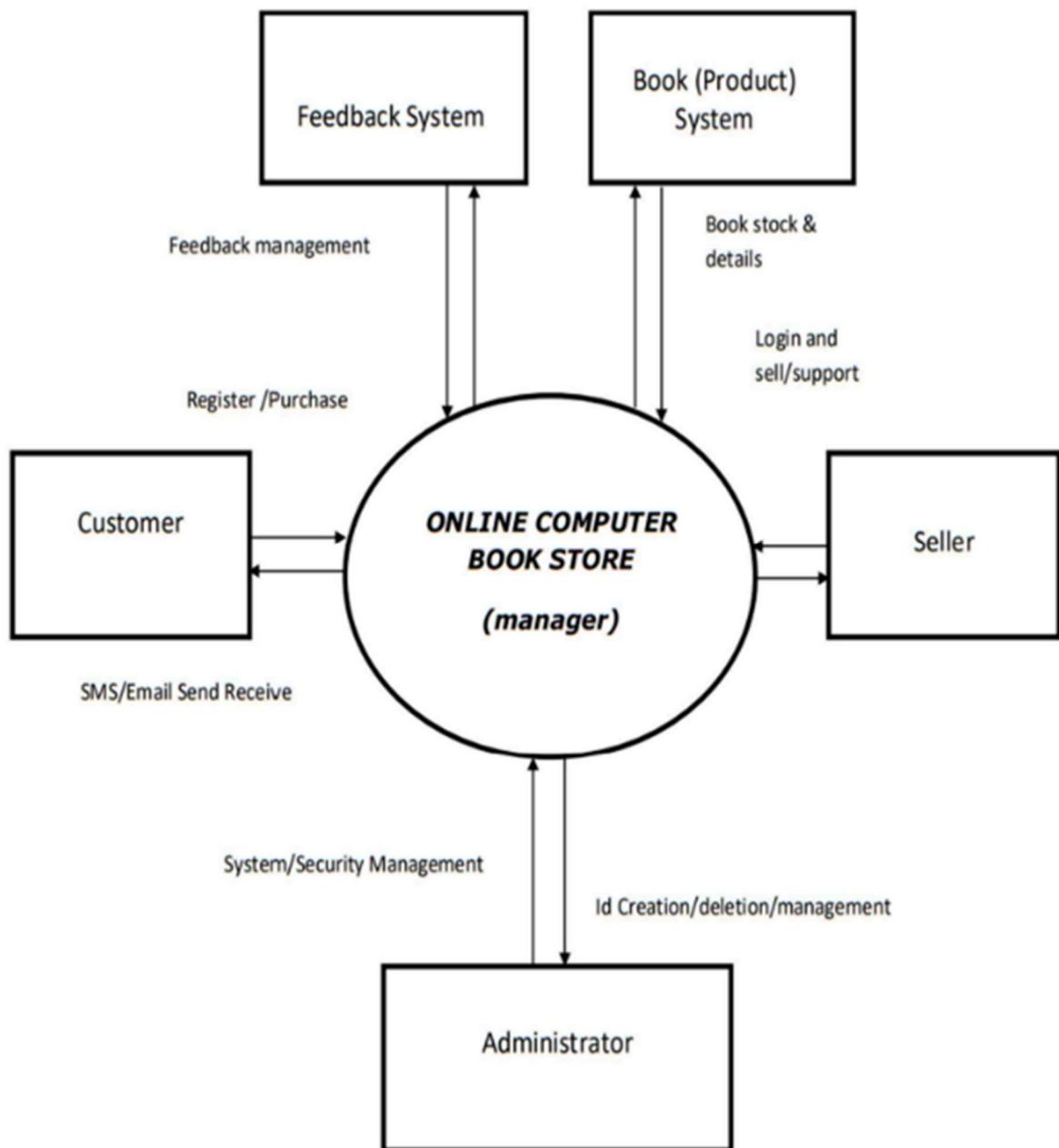


Figure3: DFD-Level 0

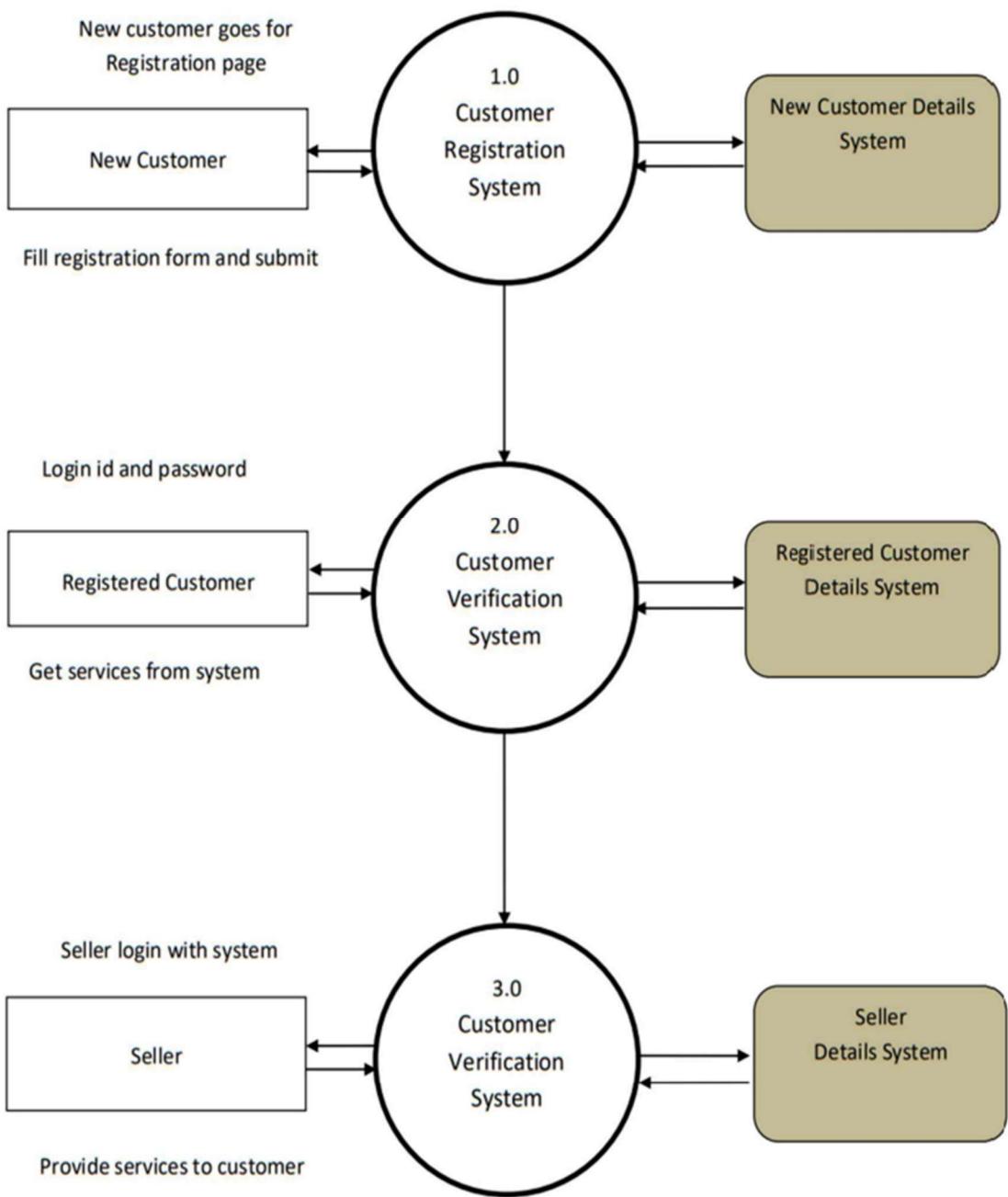


Figure4: DFD-Level 1

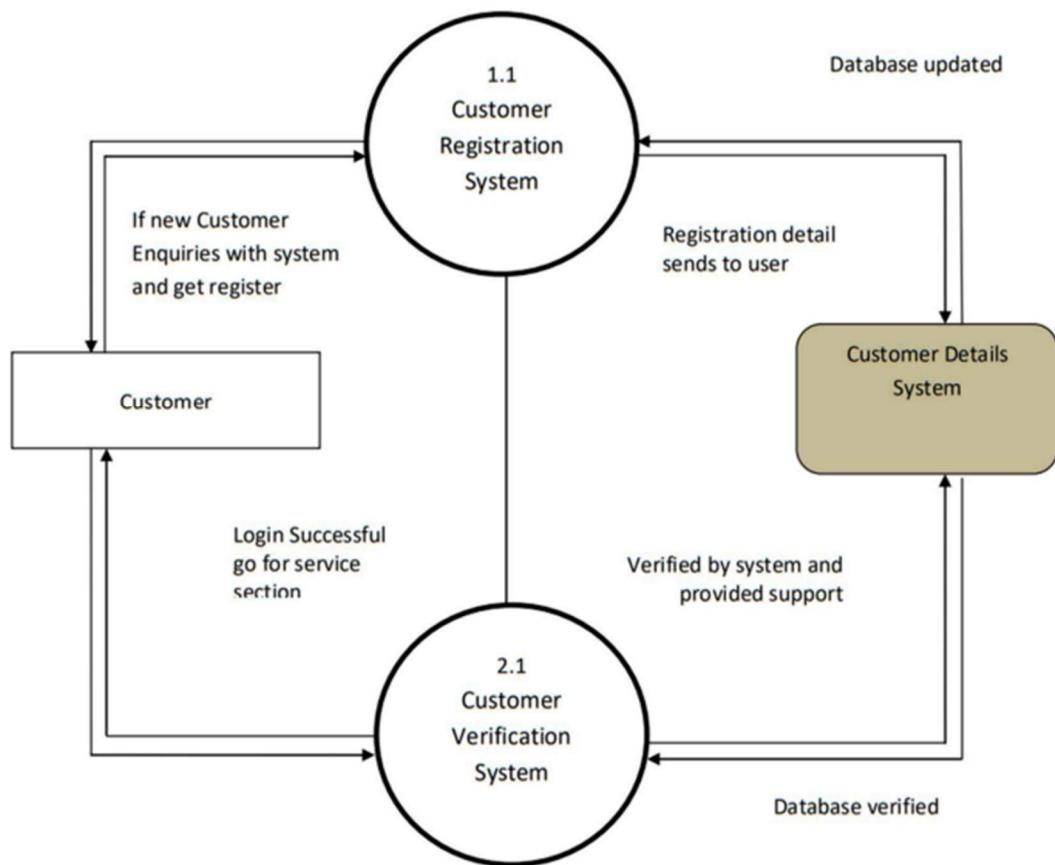


Figure5: DFD-Level 2

14. DATA STRUCTURE

Customer table

| S.No. | Field Name | Description | Data Type |
|-------|--------------------|---------------------|-------------|
| 1. | customer_id | Customer id | Varchar(10) |
| 2. | customer_pass | Customer password | Varchar(10) |
| 3. | customer_name | Customer Name | Varchar(22) |
| 4. | customer_email | Customer Email | Varchar(22) |
| 5. | customer_address | Customer Address | Varchar(50) |
| 6. | customer_contactno | Customer Contact No | Varchar(10) |

Book (Product) table

| S.No. | Field Name | Description | Data Type |
|-------|------------|---------------------------------|-------------|
| 1. | Book_id | Book unique id | Varchar(10) |
| 2. | Book_type | Book Type – Soft copy/Hard Copy | Varchar(10) |
| 3. | Book_name | Book Name | Varchar(25) |
| 4. | Book_isbn | Book ISBN No | Varchar(25) |

| S.No. | Field Name | Description | Data Type |
|--------------|-------------------|--------------------|------------------|
| 1. | seller_id | Seller id | Varchar(10) |
| 2. | seller_pass | Seller password | Varchar(10) |
| 3. | seller_name | Seller Name | Varchar(22) |
| 4. | seller_email | Seller Email | Varchar(22) |
| 5. | seller_address | Seller Address | Varchar(50) |
| 6. | seller_contactno | Seller Contact No | Varchar(10) |

Seller table

Administrator Table

| S.No. | Field Name | Description | Data Type |
|--------------|-------------------|--------------------|------------------|
| 1. | Admin_Id | Admin unique id | Varchar(10) |
| 2. | Admin_Pass | Admin password | Varchar(10) |
| 3. | Admin_Name | Admin Name | Varchar(22) |
| 4. | Admin_Email | Admin Email Id | Varchar(22) |

Feedback Table

| S.No. | Field Name | Description | Data Type |
|-------|------------|-------------|-------------|
| 1. | Name | Name | Varchar(10) |
| 2. | Email | Email | Varchar(20) |
| 3. | Contact | Contact | Varchar(22) |
| 4. | Feedback | Feedback | Varchar(22) |

Payment Table

| S.No. | Field Name | Description | Data Type |
|-------|----------------|----------------|-------------|
| 1. | Name | Name | Varchar(10) |
| 2. | Email | Email | Varchar(20) |
| 3. | Contact | Contact | Varchar(22) |
| 4. | Feedback | Feedback | Varchar(22) |
| 5. | Amount | Amount | Varchar(10) |
| 6. | Date | Date | Varchar(10) |
| 7. | Payment Status | Payment Status | Varchar(22) |

15. TESTING

Software testing can be stated as the process of validating and verifying a computer program. It is the process of evaluation a software item to detect differences between given input and expected output. Testing assesses the quality of the product and satisfies the needs of stakeholders.

Software testing, depending on the testing method employed, can be implemented at any time in the software development process. Traditionally most of the test effort occurs after the requirements have been defined and the coding process has been completed. There are various types of software testing methods which are discussed in below:

- Black box testing
- White box testing
- Functional testing
- End-to-end testing
- Acceptance testing
- Stress testing
- Load testing
- Security testing

- Compatibility testing
- System testing
- Unit testing
- Alpha testing
- Beta testing

Different kinds of tests shall be carried out at different phases of the development to reach the desired outcome, and tested result will be shared at the time of development and testing of this project.

16. INDUSTRY/CLIENT INTERFACE

Are you doing this project for any Industry/Client?

Mention Yes/No. If Yes, Mention the Name and Address of the Industry or Client.

Are you doing this project for any Industry/Client?

Yes

No

17. FUTURE SCOPE

"eBOOK STORE - ONLINE COMPUTER BOOK STORE has wide scope because everyday everyone related to study. In that situation there are minimum possibilities to carry hard book. So with the help of system customer can purchase soft copy book also and integrated with their online book. Any time they are to read this book after login. So Following are key points for future scope and further enhancement of this project:

- Project URL/platform should be secure.
- It is easy to download in slow connection also.
- Proper email and mobile integration facility need to enable.
- Need to develop mobile apps to easily use this system.

18. BIBLIOGRAPHY

- Introduction to Software Engineering (IGNOU)
- Project Guide Booklet (IGNOU)
- PHP& MySQL Black Book (DreamTech Press)
- Software Engineering (Techmax Publication)

Project Report

5.1 PROJECT TITLE

“eBOOK STORE - ONLINE COMPUTER BOOK STORE” is title of my project. It is designed with the objective to manage eBooks online. The title is proposed because this system help to sale purchase and manage eBooks through our system. This system provides very innovative way to support to the customer. User have two choice either they can purchase book as a soft copy format or hard copy format. As per technologies innovation, demands of soft copy increases day by day. eBook can be easily used in any devices like smart phones, Tab, Computer System, and Laptop, iPad etc. Best part is that carry and storage of eBook is very easy. For environment support it is very good steps to use eBooks in our daily life. So, for environmental support as well as user support we decide to develop this system with title “eBook – ONLINE COMPUTER BOOK STORE”.

5.2 INTRODUCTION AND OBJECTIVES

5.2.1 Introduction

In today's world of emerging technology, computers are playing a vital role in every walk of life. The problems due to the traditional system are overcome with the help of tasks being online. Maintenance of the data like insertion, deletion and modification is difficult with manual systems. The basic option of security is itself not provided, which is of major concern. Apart from this, storage of voluminous amounts of data is difficult. Moreover, the problems of consistency, reliability, integrity also exists. Since this is a manual system there is always a probability that there is loss of data, resulting in less durability. As these issues are of major concern, we developed an application where in all the above factors are achieved.

This online platform provides the visitor, customer and administrator an easy and efficient way to buy and manage books online. By “eBOOK STORE - ONLINE COMPUTER BOOK STORE” we try to

support the environment as well as provide full support to user. They can carry lots of books in their phone and study anytime. So, this system is very effective and useable in future scenarios.

In our case we have named our eBook facility as “Ardent” which means one who love doing something, so in our case it would be book so that summarizes the meaning of Ardent as one who loves reading book.

5.2.2 Objective(s)

“eBOOK STORE - ONLINE COMPUTER BOOK STORE” is designed to fulfill many objectives such as:

- Our system facilitates easy shopping online with just one click of buying different genres of books.
- Customers can choose either a soft copy which will be delivered online to their respective emails or hard copy of selected book which will be delivered to their address with the inclusion of nominal or no delivery charges.
- Our system provides an easy user interface (UI) with the best user

experience (UX) as we provide information about the products in categories or different genres.

- Provides e-mail facility for future correspondence.
- Our system provides backup facility as if the user lost the soft copy of the book we will provide them with the new one after the authentication of the user.
- Provide online storage facility.

5.3 TECHNOLOGIES SURVEY

“EBOOK STORE - ONLINE COMPUTER BOOK STORE” is middle level project and after seeing all the available technologies today, we chose the following:

- FRONT-END INTERFACE : HTML, CSS, JS
- BACK-END INTERFACE : NODEJS
- DATABASE : MONGODB
- INTERNET TECHNOLOGY : BOOTSTRAP, EXPRESSJS
MONGOOSE, MONGODB
COMPASS
- IDE : VS CODE EDITOR

1. **FRONT-END INTERFACE:** The front end is designed using HTML, CSS and JS. HTML or Hyper Text Markup Language is the main markup language for creating web pages and other information that can be displayed in a web browser. HTML is written in the form of HTML elements consisting of tags enclosed in angle brackets within the web page content. HTML tags most commonly come in

pairs like <h1> and </h1> although some tags represent empty elements and so are unpaired, for example s. The first tag in a pair is the start tag, and the second tag is the end tag (they are also called opening tags and closing tags). In between these tags web designers can add text, further tags, comments and other types of text-based content. The purpose of a web browser is to read HTML documents and compose them into visible or audible web pages. The browser does not display the HTML tags but uses the tags to interpret the content of the page. HTML elements form the building blocks of all websites. HTML allows images and objects to be embedded and can be used to create interactive forms. It provides a means to create structured documents by denoting structural semantics for text such as headings, paragraphs, lists, links, quotes, and other items. It can embed scripts written in languages such as JavaScript which affects the behavior of HTML web pages.

CSS or Cascading Style Sheets is a style sheet language used for describing the look and formatting of a document written in a markup language. While most often used to style web pages and interfaces written in HTML, the language can be applied to any kind of XML

document, including plain XML, SVG and XUL. CSS is a cornerstone specification of the web and almost all web pages use CSS style sheets to describe their presentation. CSS is designed primarily to enable the separation of document content from document presentation, including elements such as the layout, colors, and fonts. This separation can improve content accessibility, provide more flexibility and control in the specification of presentation characteristics, enable multiple pages to share formatting, and reduce complexity and repetition in the structural content. CSS can also allow the same markup page to be presented in different styles for different rendering methods, such as on-screen, in print, by voice (when 14 read out by a speech-based browser or screen reader) and on Braille-based, tactile devices. It can also be used to allow the web page to display differently depending on the screen size or device on which it is being viewed. While the author of a document typically links that document to a CSS file, readers can use a different style sheet, perhaps one on their own computer, to override the one the author has specified. However, if the author or the reader did not link the document to a specific style sheet the default style of the browser

will be applied. CSS specifies a priority scheme to determine which style rules apply if more than one rule matches against a particular element. In this so-called cascade, priorities or weights are calculated and assigned to rules, so that the results are predictable.

JS or JavaScript is a dynamic computer programming language. It is most used as part of web browsers, whose implementations allow client-side scripts to interact with the user, control the browser, communicate asynchronously, and alter the document content that is displayed. It is also being used in server-side programming, game development and the creation of desktop and mobile applications.

JavaScript is a prototype-based scripting language with dynamic typing and has first-class functions. Its syntax was influenced by C. JavaScript copies many names and naming conventions from Java, but the two languages are otherwise unrelated and have very different semantics. The key design principles within JavaScript are taken from the Self and Scheme programming languages. It is a multiparadigm language, supporting object-oriented, imperative, and functional programming styles. The application of JavaScript to use outside of web pages—for example, in PDF documents, site-specific

browsers, and desktop which is also significant. Newer and faster JavaScript VMs and platforms built upon them (notably Node.js) have also increased the popularity of JavaScript for server-side web applications.

2. BACK-END INTERFACE: Node.js is a cross-platform, open-source server environment that can run on Windows, Linux, Unix, macOS, and more. Node.js is a back-end JavaScript runtime environment, runs on the V8 JavaScript Engine, and executes JavaScript code outside a web browser.

Node.js lets developers use JavaScript to write command line tools and for server-side scripting. The functionality of running scripts server-side produces dynamic web page content before the page is sent to the user's web browser. Consequently, Node.js represents a "JavaScript everywhere" paradigm, unifying web-application development around a single programming language, rather than different languages for server-side and client-side scripts.

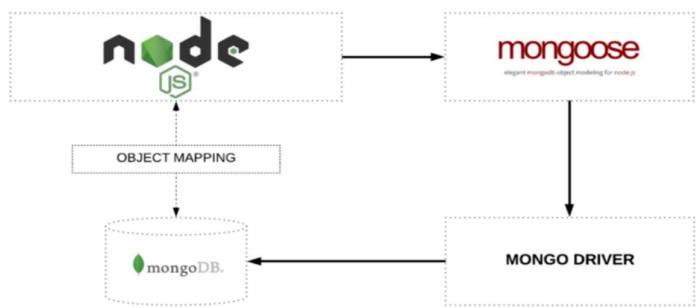
3. DATABASE: MongoDB is a source-available cross-platform document-oriented database program. Classified as a NoSQL

database program, MongoDB uses JSON-like documents with optional schemas. MongoDB's document model is simple for developers to learn and use, while still providing all the capabilities needed to meet the most complex requirements at any scale.

4. INTERNET TECHNOLOGIES: BOOTSTRAP is a free and open-source CSS framework directed at responsive, front-end web development. It is an HTML, CSS and JS library that focuses on simplifying the development of informative web. The primary purpose of adding it to a web project is to apply Bootstrap's choices of color, size, font and layout to that project. Bootstrap also comes with several JavaScript components which do not require other libraries like jQuery. They provide additional user interface elements such as dialog boxes, tooltips, progress bars, navigation dropdowns, and carousels. Each Bootstrap component consists of an HTML structure, CSS declarations, and in some cases accompanying JavaScript code. They also extend the functionality of some existing interface elements.

Express.js, or simply Express, is a back-end web application framework for building RESTful APIs with Node.js, released as free and open-source software. It is designed for building web applications and APIs. It has been called the de facto standard server framework for Node.js. Express.js is a small framework that works on top of Node.js web server functionality to simplify its APIs and add helpful new features. It makes it easier to organize your application's functionality with middleware and routing. It adds helpful utilities to Node.js HTTP objects and facilitates the rendering of dynamic HTTP objects.

MONGOOSE is an Object Data Modeling (ODM) library for MongoDB and Node.js. It manages



Object Mapping between Node and MongoDB managed via Mongoose

relationships between data, provides schema validation, and is used to translate between objects in code and the representation of those objects in MongoDB.

MongoDB Compass is a powerful GUI for querying, aggregating, and analyzing your MongoDB data in a visual environment.

Compass is free to use and source available, and can be run on macOS, Windows, and Linux.

5. INTEGRATED DEVELOPMENT ENVIRONMENT: Visual Studio Code, also commonly referred to as VS Code, is a source-code editor made by Microsoft with the Electron Framework, for Windows, Linux and macOS. Features include support for debugging, syntax highlighting, intelligent code completion, snippets, code refactoring, and embedded Git. Users can change the theme, keyboard shortcuts, preferences, and install extensions that add functionality.

5.4. SYSTEM REQUIREMENT SPECIFICATION

A software requirements specification (SRS) is a description of a software system to be developed. It provides all the details of the software that needs to be developed inclusive of problem definition, technologies to be used, costing etc.

5.4.1 PROBLEM DEFINITION

“EBOOK STORE - ONLINE COMPUTER BOOK STORE will enable the customer to easily get involved in sale, purchase and management of the book in soft copy as well as hard copy format.

This system provides many benefits to society as well as our environment. But to develop this proposed we have to face many problems and solve them in effective way.

Following are some problem areas which are identified to develop this system and we have to solve them, they are:

- Login id creation and management
- Book details management in both formats (Soft & Hard Copy)

- Stock Management
- Customer management
- Seller Management
- Feedback Management
- Software and web management
- Transportation management in case of hard copy
- Payment management
- Copyright management
- Software awareness management

5.4.2 INTRODUCTION TO DIFFERENT MODULE

After analysis and study of this system we have decided to manage below mention module to support and develop this system properly.

The modules of proposed system are:

- 1) Customer Module
- 2) Seller Module
- 3) Administrator Module
- 4) Book Module

5) Feedback Module

These above modules help to solve above mention problem and get final objective which is decided to achieve the goal for the proposed system.

5.4.3 MODULE DESCRIPTION

Following module used in this proposed system are described in below:

- 1) Customer Module
- 2) Seller Module
- 3) Administrator Module
- 4) Book Module
- 5) Feedback Module
- 6) Payment Module

- 1) **Customer Module:** This module is designed to manage the customer properly. By using this module customer registered with

this system and perform eBook shopping and other activities on the system.

Process Logic:

- Customer visited on website.
- They register with system.
- They generate a valid username same as email and the password.
- System confirms and verifies the user.
- After verification the forwarded on shopping module.
- If user comes back to the site, he/she needs to login again.
- If any chance he forgets his password, then the system automatically generates a password and send it to the email address of the user and updates the same in the data base.

2) **Seller Module:** This module is designed for seller of this system.

The seller also registered with system and provides an interface with the help of administrator to manage sellers.

Process Logic:

- Seller registers with the system.
- After registration administrator provide seller id to the seller.

- Sellers send and upload product details.
- And help customers to provide product and support.
- If seller comes back to the site, he/she needs to login again
- If any chance he forgets his password, then the system automatically generates a password and sent it to the email address of the seller and updates the same in the data base.

3) **Administrator Module:** This module is designed for administrator only. They are responsible for managing the entire system. Manage customer as well as seller of this system. And provide security to the system.

Process Logic:

- Administrator is responsible for overall execution of the project
- They create customer id and seller id
- Verify the customer and seller
- Provide forgotten password to the customer and seller if required
- Making SEO based website for “**eBOOK ONLINE**

COMPUTER BOOK STORE”

- Manage the database properly
- Manage validation and connection with full security

4) Book (Product) Module: This module is used for book and their stock management. Each product has complete details with system.

We will integrate more functionality to this model in future.

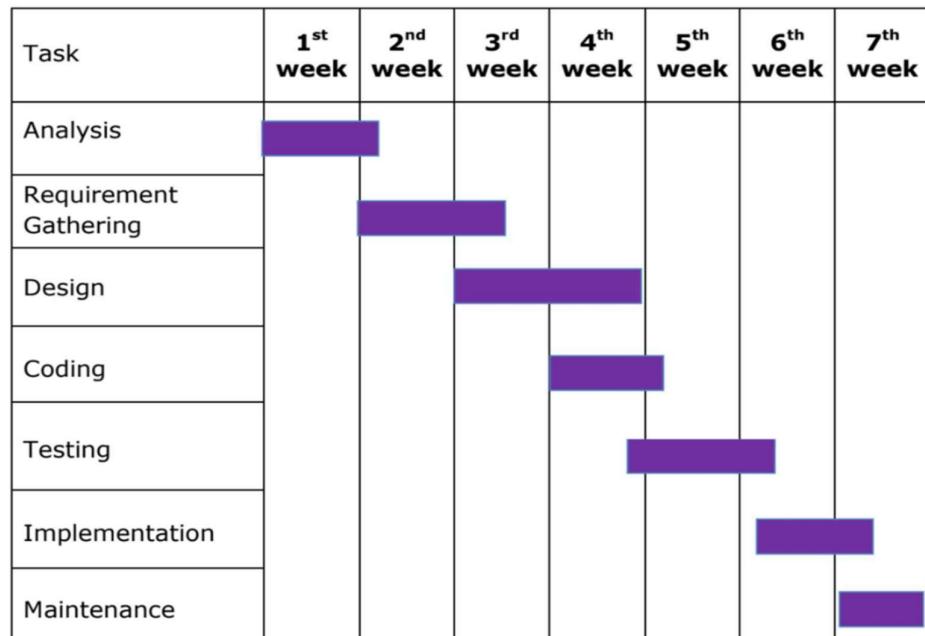
5) Feedback Module: This module is used for feedback purpose.

With the help of this module customer and seller share their

valuable feedback. Based on this feedback administrator manage and provide new approach to manage this system.

5.4.4 PLANNING AND SCHEDULING

Gantt chart is a type of bar chart, developed by Henry Gantt in the 1910s, that illustrates a project schedule. Gantt charts illustrate the start and finish dates of the terminal elements and summary elements of a project.



5.4.5 SOFTWARE DEVELOPMENT LIFECYCLE

To manage this, a number of system development life cycle (SDLC) models present such as

- Waterfall Model
- Build and Fix Model
- Spiral Mode
- Prototype Model
- Interactive Enhancement Model
- RAD Approach

The more efficient model is Spiral Model so I choose Spiral Model to develop this project.

Spiral Model - Design

The spiral model has four phases. A software project repeatedly passes through these phases in iterations called Spirals.

The four phases of the Spiral model are:

1. **Identification** : This phase starts with gathering the business requirements in the baseline spiral. In the subsequent spirals as the

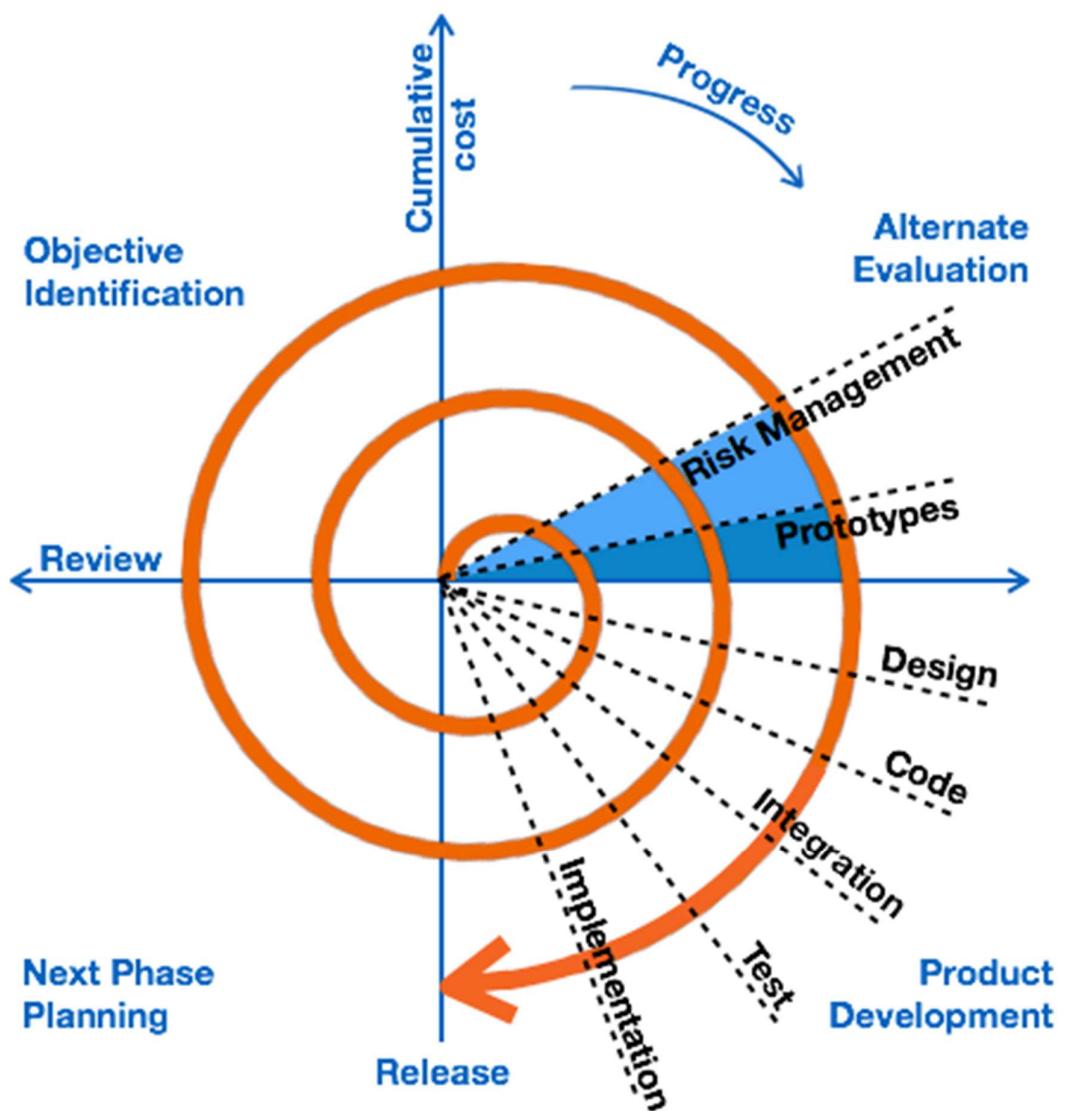
product matures, identification of system requirements, subsystem requirements and unit requirements are all done in this phase.

2. Design : The Design phase starts with the conceptual design in the baseline spiral and involves architectural design, logical design of modules, physical product design and the final design in the subsequent spirals.

3. Construct or Build : The Construct phase refers to production of the actual software product at every spiral. In the baseline spiral, when the product is just thought of and the design is being developed a POC (Proof of Concept) is developed in this phase to get customer feedback.

4. Evaluation and Risk Analysis : Risk Analysis includes identifying, estimating and monitoring the technical feasibility and management risks, such as schedule slippage and cost overrun. After testing the build, at the end of first iteration, the customer evaluates the software and provides feedback.

The following diagram is a representation of the Spiral Model, listing the activities in each phase.



5.4.6 ENTITY-RELATIONSHIP(ER) DIAGRAM

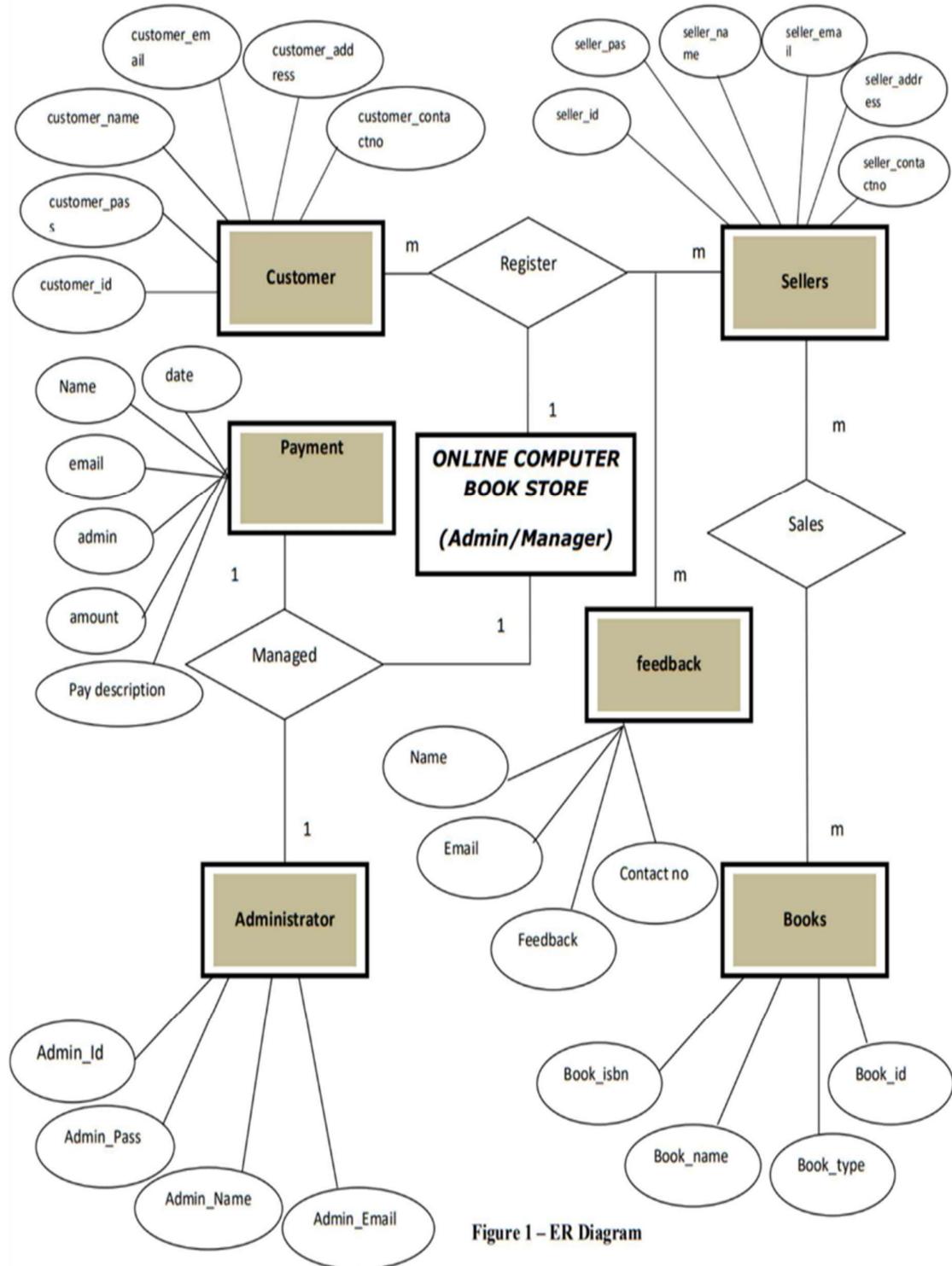


Figure 1 – ER Diagram

5.4.7 USE CASE DIAGRAM

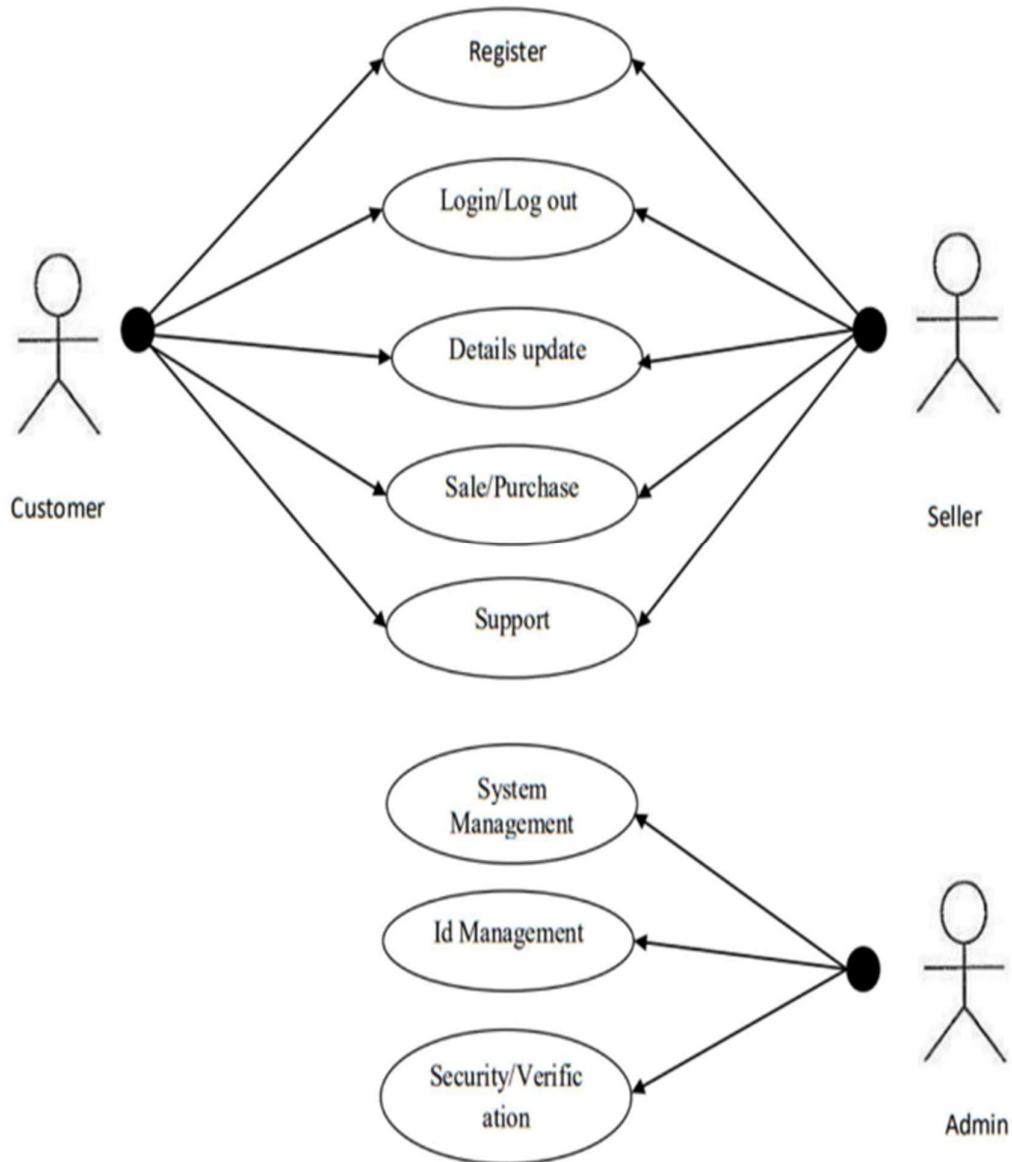


Figure 2 – Use Case Diagram

5.4.8 DATA FLOW DIAGRAM (DFDs)

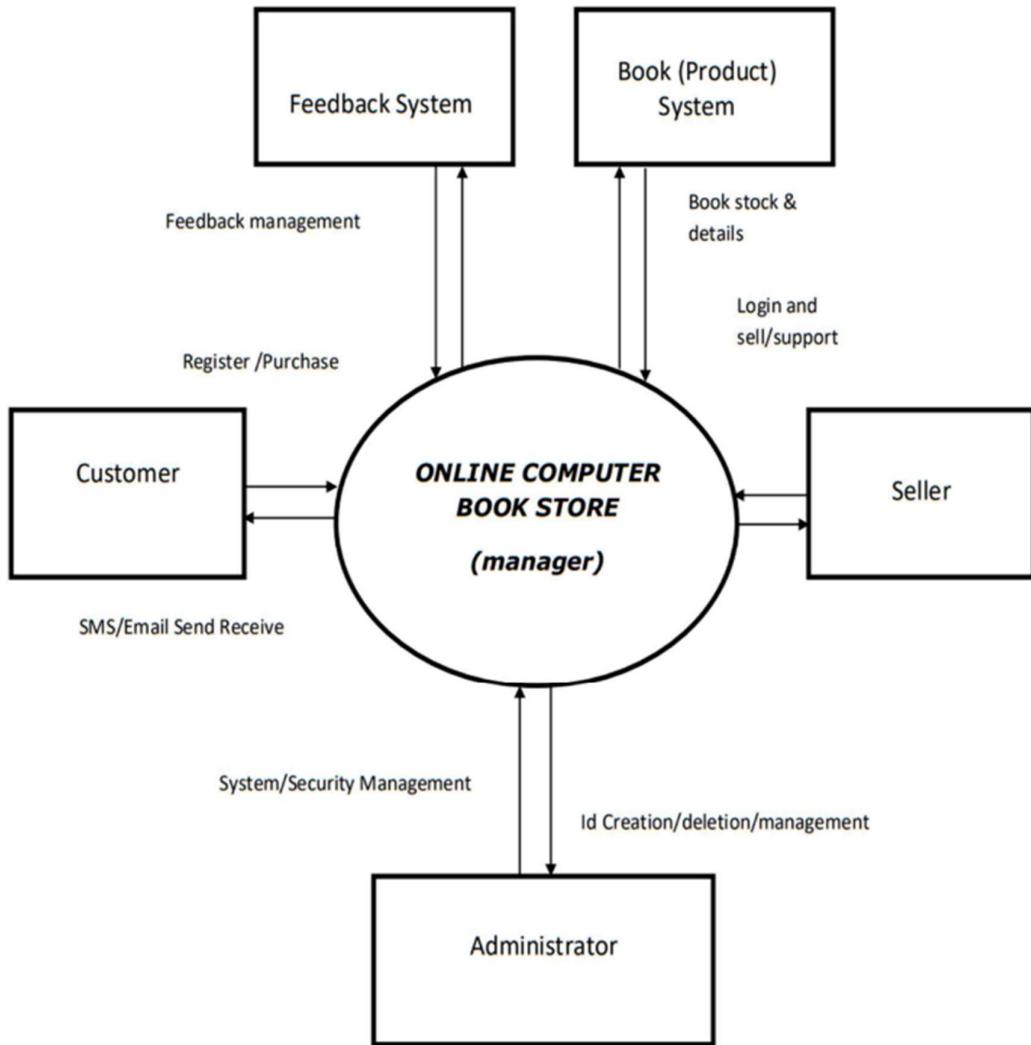


Figure3: DFD-Level 0

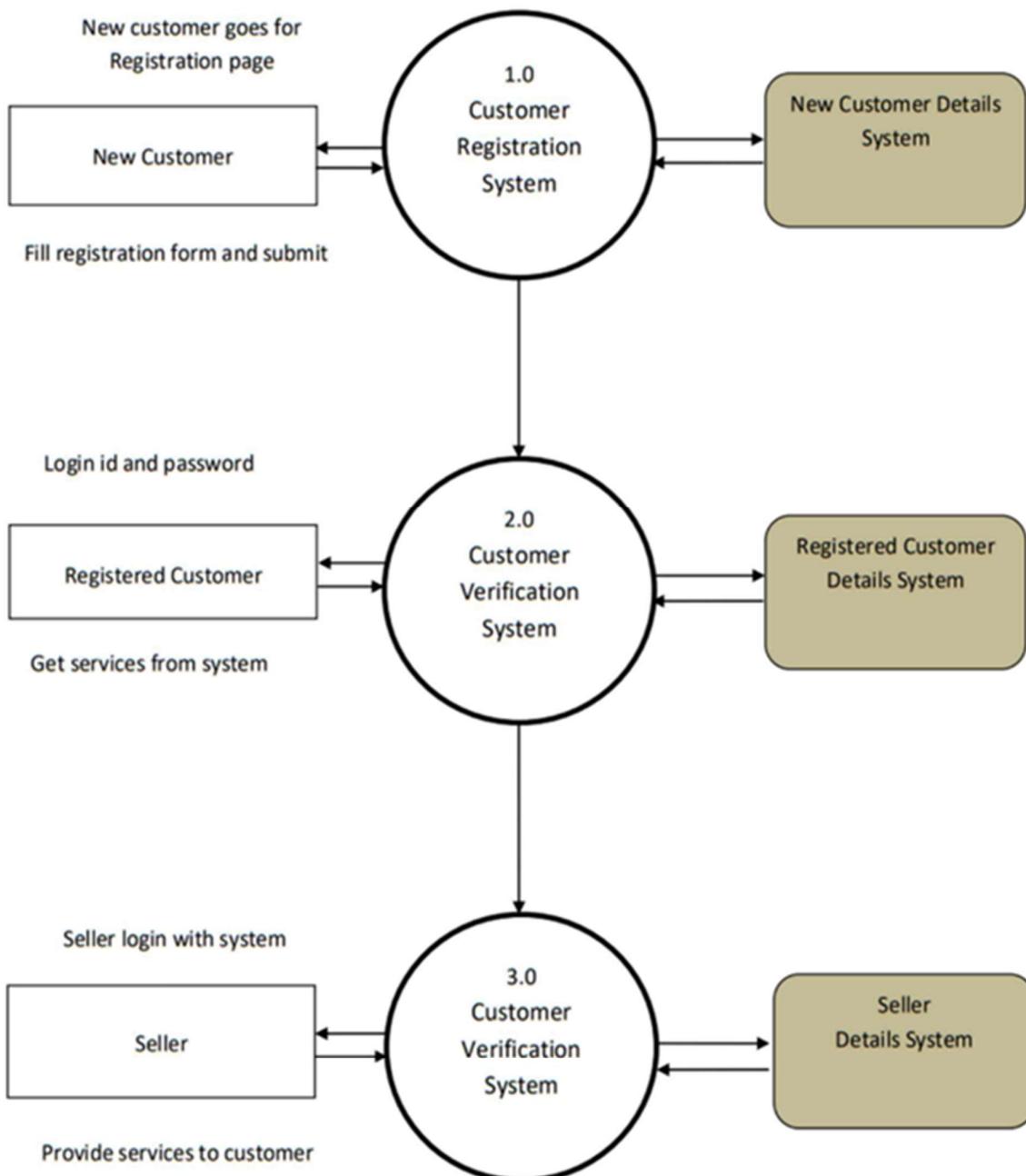


Figure4: DFD-Level 1

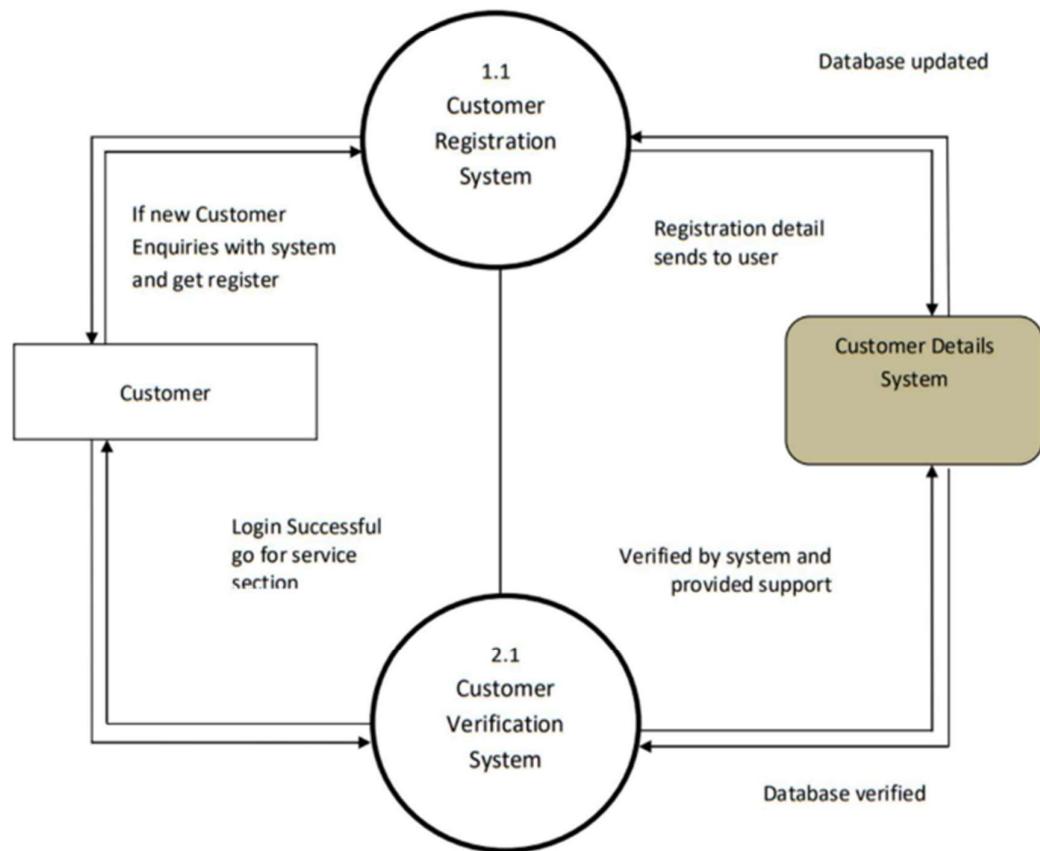
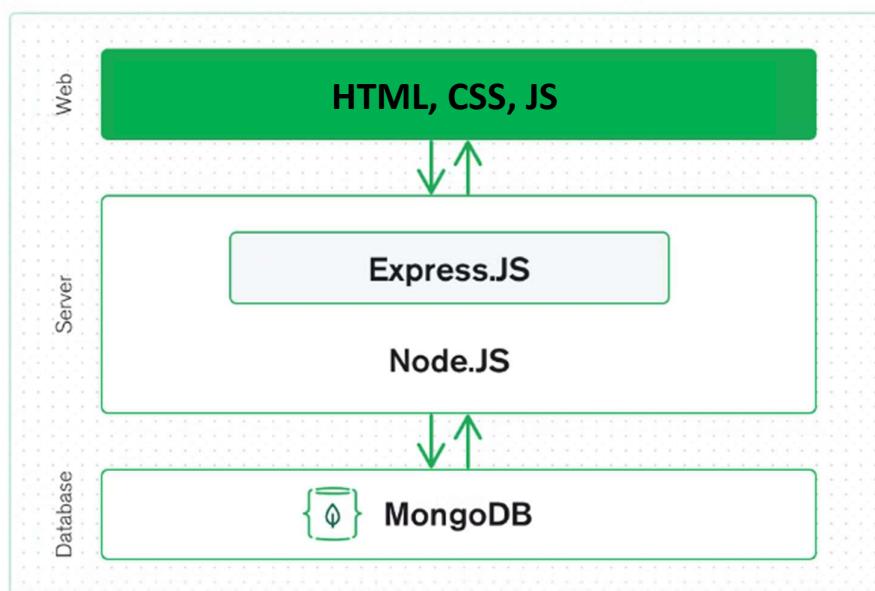
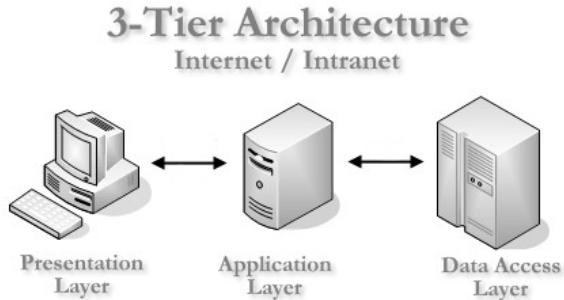


Figure5: DFD-Level 2

5.4.9 PROJECT CATEGORY

This application is based on three tier architecture where HTML, CSS and JS is used as front-end technology , NodeJs and ExpressJs is used as a middleware or application layer and MongoDB is used as back-end technology and database. The connection between the front-end and the back-end is established using ExpressJs which is the web framework of Nodejs. Some style sheets, and scripting technologies are used to enhance the dynamism of the project.



5.4.10 DATA STRUCTURES/ DATA SCHEMAS

CUSTOMER MODEL :

```
customer_name: String,  
customer_email: String,  
customer_mobileNo: Number,  
customer_password: String,  
customer_address: String,  
customer_pincode: Number
```

SELLER MODEL :

```
seller_name: String,  
seller_email: String,  
seller_mobileNo: Number,  
seller_password: String,  
seller_address: String,  
seller_pincode: Number
```

FEEDBACK MODEL

```
reviewer_name: String,  
reviewer_email: String,  
reviewer_mobileNo: Number,  
review: String
```

BOOK MODEL :

```
book_name: String,  
book_author: String,  
book_type: String,  
book_description: String,  
book_price: Number,  
book_img:  
{ data: Buffer,  
  contentType: String }
```

ADMIN MODEL :

```
admin_name: String,  
admin_mobileNo: Number,  
admin_password: String,  
admin_email: String,  
admin_address: String,  
admin_pincode: Number
```

5.4.11 HARDWARE AND SOFTWARE REQUIREMENT

Hardware:

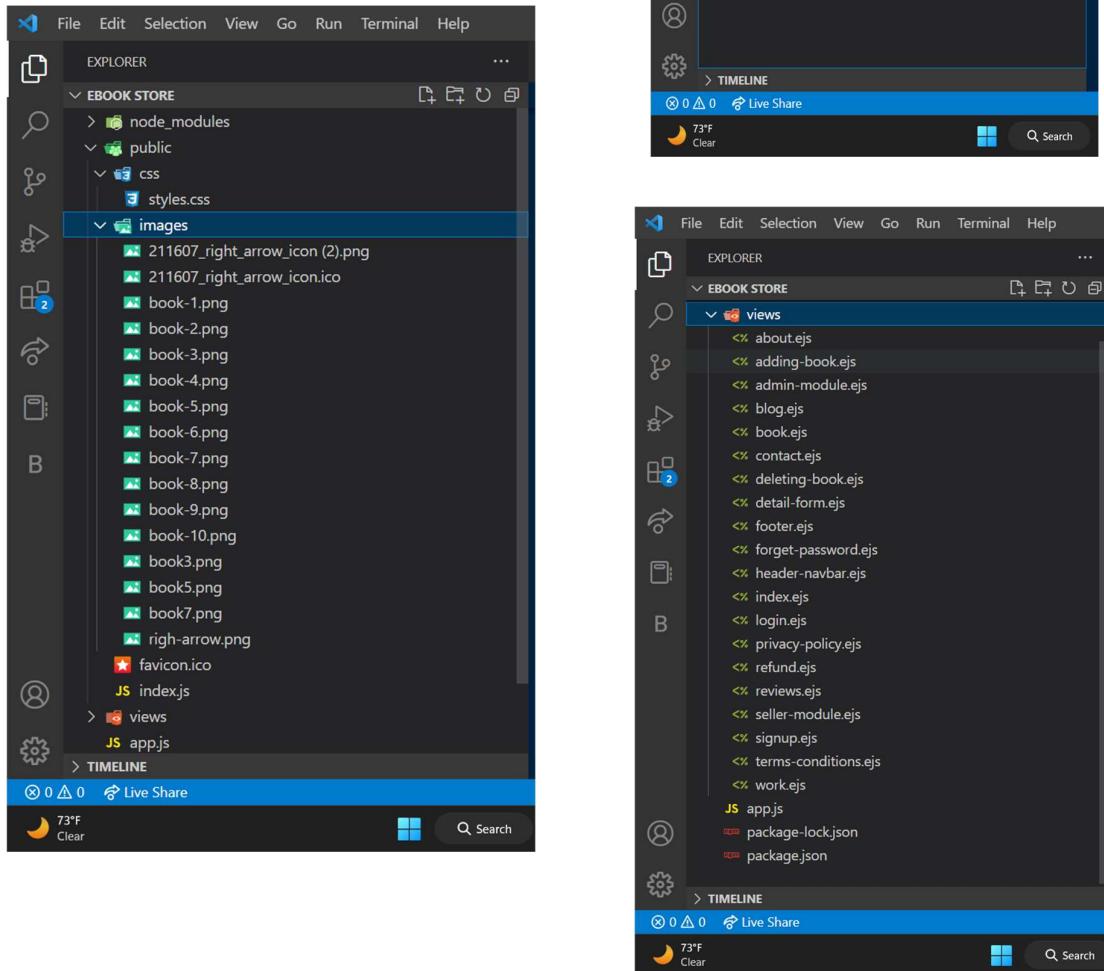
- Processor – Intel core i3
- Ram – 1GB
- Hard Disk – minimum 80Gb /250GB
- Keyboard and mouse – USB / PS2

Software:

- OS – windows7/ XP/ Windows 2008
- Front End – HTML, CSS , JS
- Database - MONGODB
- Web Browser – IE/Chrome
- Scripting – Java Script
- Web Server – NODEJS
- Other Tools – EXPRESSJS, BOOTSTRAP

5.5 PROGRAM CODE

Before writing all the codes, I would like to include the screenshot of the folder structure, so we can easily understand the implementation of code and the linking and importing of each file.



6.1 Source code for “index.ejs” file :

```
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Ardent</title>

    <!-- bootstrap -->
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css" rel="stylesheet"
        integrity="sha384-1BmE4kBq78iYhFdvKuhfTAU6auU8tT94WrHftjDbrCEXSU1oBoqyl2QvZ6jIW3"
        crossorigin="anonymous">

    <!-- CSS links -->
    <link rel="stylesheet" href="css/styles.css">

    <!-- font awesome -->
    <script src="https://kit.fontawesome.com/ed7230efcb.js"
        crossorigin="anonymous"></script>

    <!-- icon link -->
    <link rel="shortcut icon" href="/favicon.ico" type="image/x-icon">

</head>

<body>

    <!-- navigation bar      -->
    <%- include('./header-navbar'); %>

    <!-- book image carousel -->
    <div id="carouselExampleControls" class="carousel slide" data-ride="carousel"
        data-interval="3000">
        <div class="carousel-inner">
            <div class="carousel-item active">
                
                
                
            </div>
            <div class="carousel-item">
                
                
                
            </div>
            <div class="carousel-item">
                
                
                
            </div>
        </div>
    </div>


```

```

        </div>

        <a class="carousel-control-prev" href="#carouselExampleControls"
           role="button" data-slide="prev">
            <span class="carousel-control-prev-icon" aria-hidden="true"></span>
            <span class="sr-only">Previous</span>
        </a>
        <a class="carousel-control-next" href="#carouselExampleControls"
           role="button" data-slide="next">
            <span class="carousel-control-next-icon" aria-hidden="true"></span>
            <span class="sr-only">Next</span>
        </a>
    </div>

<div class="content-con">
    <h1>Grow Your Mind!!</h1>
    <h5>Today a Reader, Tomorrow a Leader</h5>
</div>

<hr>

<div class="feature-box">

    <div class="feature">
        <i class="fa-solid fa-building-lock fa-2xl">
            </i>
        <p>Secure Payments</p>
    </div>
    <div class="feature">
        <i class="fa-solid fa-rotate-left fa-2xl">
            </i>
        <p>Easy Returns</p>
    </div>
    <div class="feature">
        <i class="fa-solid fa-headset fa-2xl">
            </i>
        <p>24x7 customer support</p>
    </div>
    <div class="feature">
        <i class="fa-solid fa-truck-fast fa-2xl">
            </i>
        <p>Fast shipping</p>
    </div>

</div>

```

```

<!-- review section -->

<h3 class="title-heading">Client's Reviews</h3>

<div class="card-deck">

    <div class="card" >
        
        <div class="card-body">
            <h5 class="card-title">Roman Reign</h5>
            <p class="card-text">Lorem ipsum dolor sit, amet consectetur adipisicing
elit. Eius exercitationem obcaecati molestias?</p>
            <a href="#" class="btn btn-outline-info">Read More</a>
        </div>
    </div>

    <div class="card" >
        
        <div class="card-body">
            <h5 class="card-title">Sam Morek</h5>
            <p class="card-text">Lorem ipsum dolor sit, amet consectetur adipisicing
elit. Eius exercitationem obcaecati molestias?</p>
            <a href="#" class="btn btn-outline-info">Read More</a>
        </div>
    </div>

    <div class="card" >
        
        <div class="card-body">
            <h5 class="card-title">Arnold Satur</h5>
            <p class="card-text">Lorem ipsum dolor sit, amet consectetur adipisicing
elit. Eius exercitationem obcaecati molestias?</p>
            <a href="#" class="btn btn-outline-info">Read More</a>
        </div>
    </div>

    <div class="card" >
        
        <div class="card-body">
            <h5 class="card-title">Eric Rowan</h5>
            <p class="card-text">Lorem ipsum dolor sit, amet consectetur adipisicing
elit. Eius exercitationem obcaecati molestias?</p>
            <a href="#" class="btn btn-outline-info">Read More</a>
        </div>
    </div>

<h3 class="title-heading">Books</h3>

```

```

    <!-- book section -->
<div class="container">
  <div class="card-deck">
    <div class="row">
      <!-- including book using java language -->
      <% books.forEach(function(book){%>
        <div class="col-3">
          <div class="card" >
            
            <div class="card-body" >
              <h5 class="card-title"><%= book.book_name %></h5>
              <p class="card-text"><%= book.book_description %></p>
              <h5 class="card-title">Rs.<%= book.book_price %></h5>
              <button class="btn btn-primary">Buy Now</button>
              <button class="btn btn-outline-primary">Add to Cart</button>
            </div>
          </div>
        </div>
      <%})%>
    </div>
    </div>
  </div>
</div>
      <a href="book" > <button class="btn btn-success" >Get more
books</button></a>
<!-- footer section including using Java -->
<% include("./footer"); %>

      <script src="https://code.jquery.com/jquery-3.3.1.slim.min.js"
           integrity="sha384-
q8i/X+965Dz00rT7abK41JStQIAqVgRVzpbzo5smXKp4YfRvH+8abtTE1Pi6jizo"
           crossorigin="anonymous"></script>
      <script
src="https://cdn.jsdelivr.net/npm/popper.js@1.14.7/dist/umd/popper.min.js"
           integrity="sha384-
U02eT0CpHqdSJQ6hJty5KVphPhzWj9W01clHTMGa3JDZwrnQq4sF86dIHNDz0W1"
           crossorigin="anonymous"></script>
      <script
src="https://cdn.jsdelivr.net/npm/bootstrap@4.3.1/dist/js/bootstrap.min.js"
           integrity="sha384-
JjSmVgyd0p3pXB1rRibZUAYoIIy6OrQ6VrjIEaFF/nJGzIxFDsf4x0xIM+B07jRM"
           crossorigin="anonymous"></script>
</body>
</html>

```

6.2 Source code for “Contact.ejs” file :

```

<!DOCTYPE html>
<html lang="en">
<head>
```

```

<meta charset="UTF-8">
<meta http-equiv="X-UA-Compatible" content="IE=edge">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Contact</title>
<!-- bootstrap -->
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-1BmE4kWBq78iYhFldvKuhfTAU6auU8tT94WrHftjDbrCEXSU1oBoqyl2QvZ6jIW3" crossorigin="anonymous">
<!-- js -->
<script src="https://cdn.jsdelivr.net/npm/@popperjs/core@2.10.2/dist/umd/popper.min.js" integrity="sha384-7+zCNj/IqJ95wo16oMtfSkBz9ccEh31eOz1HGyDuCQ6wgnyJNSYdrPa03rtR1zdB" crossorigin="anonymous"></script>
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/js/bootstrap.min.js" integrity="sha384-1BmE4kWBq78iYhFldvKuhfTAU6auU8tT94WrHftjDbrCEXSU1oBoqyl2QvZ6jIW3" crossorigin="anonymous"></script>
<!-- CSS links -->
<link rel="stylesheet" href="css/styles.css">

<!-- icon link -->
<link rel="icon" href="favicon.ico">
</head>
<body>
<!-- navigation bar -->
<% include('../header-navbar'); %>
<div class="contact-form">
<p id="form">Form</p>
<form action="/contact" method="post">
<% if(isSubmitted){ %>
      <div class="alert alert-success">Your form is successfully Submitted.  
We'll revert back soon!!</div>
<% } %>
      <label for="name">NAME</label>
      <input type="text" placeholder="Enter your name" name="name">
      <label for="email">EMAIL</label>
      <input type="email" placeholder="Enter your email address" name="email">
      <label for="mobile">CONTACT</label>
      <input type="number" placeholder="Enter your mobile number" name="mobileNo">
      <label for="message">MESSAGE</label>
      <textarea name="message" cols="30" rows="10" maxlength="200" placeholder="I want to contact you because..." name="message"></textarea>
      <button class="button-dark" type="submit">Submit</button>
</form>
</div>
<!-- footer -->
<% include('../footer'); %>
</body>
</html>

```

6.3 Source code for “review.ejs”

```
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Contact</title>
    <!-- bootstrap -->
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css" rel="stylesheet"
        integrity="sha384-1BmE4kWBq78iYhFldvKuhfTAU6auU8tT94WrHftjDbrCEXSU1oBoqyl2QvZ6jIW3"
        crossorigin="anonymous">

    <!-- js -->
    <script
        src="https://cdn.jsdelivr.net/npm/@popperjs/core@2.10.2/dist/umd/popper.min.js"
        integrity="sha384-7+CNj/IqJ95wo16oMtf5KbZ9ccEh31eOz1HGyDuCQ6wgnyJNSYdrPa03rtR1zdB"
        crossorigin="anonymous"></script>
    <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/js/bootstrap.min.js"
        integrity="sha384-1BmE4kWBq78iYhFldvKuhfTAU6auU8tT94WrHftjDbrCEXSU1oBoqyl2QvZ6jIW3"
        crossorigin="anonymous"></script>
    <!-- CSS links -->
    <link rel="stylesheet" href="css/styles.css">

    <!-- icon link -->
    <link rel="icon" href="favicon.ico">
</head>

<body>
    <!-- navigation bar      -->
    <% include('./header-navbar'); %>
    <div class="contact-form">
        <p id="form">Feedback</p>
        <form action="/review" method="post">
            <% if(isSubmitted){ %>
                <div class="alert alert-success">Your feedback is valuable for us.
                Thanks for your response!!</div>
            <% } %>
                <label for="name">NAME</label>
                <input type="text" placeholder="Enter your name" name="name">
                <label for="email">EMAIL</label>
                <input type="email" placeholder="Enter your email address" name="email">
                <label for="mobile">CONTACT</label>
                <input type="number" placeholder="Enter your mobile number" name="mobileNo">
                <label for="message">FEEDBACK</label>
                <textarea name="feedback" cols="30" rows="10" maxlength="200"
                    placeholder="Type you feedback here!!" name="message"></textarea>
    </div>
</body>
```

```

                <button class="button-dark" type="submit">Submit</button>
            </form>
        </div>
        <!-- footer -->
        <%- include("./footer"); %>
</body>
</html>

```

6.4 Source Code for “book.ejs” file :

```

<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Ardent</title>
    <!-- bootstrap -->
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-1BmE4kWBq78iYhFldvKuhfTAU6auU8tT94WrHftjDbrCEXSU1oBoqyl2QvZ6jIW3" crossorigin="anonymous">
    <!-- CSS links -->
    <link rel="stylesheet" href="css/styles.css">
    <!-- font awesome -->
    <script src="https://kit.fontawesome.com/ed7230efcb.js" crossorigin="anonymous"></script>
    <link rel="shortcut icon" href="/favicon.ico" type="image/x-icon">
</head>
<body>
    <div class="navigation-bar">
        <a href="index">
            <h3>Ardent eBook</h3>
        </a>
        <div class="links">
            <a href="about">
                <h6>About</h6>
            </a>
            <a href="book">
                <h6>Books</h6>
            </a>
            <a href="reviews">
                <h6>Reviews</h6>
            </a>
            <a href="contact">
                <h6>Contact</h6>
            </a>
        </div>
        <form class="form-inline" action="/book" method="post">
            <input class="form-control mr-sm-2" type="search" placeholder="Search" aria-label="Search" name="searchedBook">
            <button class="btn btn-outline-success my-2 my-sm-0" type="submit">Search</button>
    </div>

```

```

        </form>
    </div>

    <!-- main content -->
<% if(searchedBookFound==="true"){ %>
<div class="container">
    <div class="card-deck">
        <div class="row">
            <% books.forEach(function(book){%>
                <div class="col-3" >
                    <div class="card" >

                        
                        <div class="card-body" >
                            <h5 class="card-title"><%= book.book_name %></h5>
                            <p class="card-text"><%= book.book_description %></p>
                            <h5 class="card-title">Rs.<%= book.book_price %></h5>
                            <button class="btn btn-primary">Buy Now</button>
                            <button class="btn btn-outline-primary">Add to Cart</button>
                        </div>
                    </div>
                </div>
            <%})%>
        </div>
    </div>
</div>
<div class="container">
    <div class="card-deck">
        <div class="row">
            <% books.forEach(function(book){%>
                <div class="col-3" >
                    <div class="card" >
                        
                        <div class="card-body" >
                            <h5 class="card-title"><%= book.book_name %></h5>
                            <p class="card-text"><%= book.book_description %></p>
                            <h5 class="card-title">Rs.<%= book.book_price %></h5>
                            <button class="btn btn-primary">Buy Now</button>
                            <button class="btn btn-outline-primary">Add to Cart</button>
                        </div>
                    </div>
                </div>
            <%})%>
        </div>
    </div>
</div>
<% } %>
<% if(searchedBookFound==="false"){ %>
    <div class="alert alert-danger">
        Not found
    </div>
<% } %>
    <%- include("./footer"); %>
</body>
</html>

```

Source code for “deleting-book.ejs” file:

```
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Ardent</title>
    <!-- bootstrap -->
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css" rel="stylesheet"
        integrity="sha384-1BmE4kWBq78iYhFldvKuhfTAU6auU8tT94WrHftjDbrCEXSU1oBoqyl2QvZ6jIW3"
        crossorigin="anonymous">
    <!-- font awesome -->
    <script src="https://kit.fontawesome.com/ed7230efcb.js" crossorigin="anonymous"></script>
    <link rel="shortcut icon" href="/favicon.ico" type="image/x-icon">
</head>
<body>
    <section style="background-color: #eee; height: 100vh;">
        <div class="container h-100">
            <div class="row d-flex justify-content-center align-items-center h-100">
                <div class="col-lg-12 col-xl-11">
                    <div class="card text-black" style="border-radius: 25px;">
                        <div class="card-body p-md-5">
                            <div class="row justify-content-center">
                                <div class="col-md-10 col-lg-6 col-xl-5 order-2 order-lg-1">
                                    <p class="text-center h1 fw-bold mb-5 mx-1 mx-md-4 mt-4">Delete Books</p>
                                    <% if(status=="1"){ %>
                                        <div class="alert alert-success" role="alert">
                                            <%= message %>
                                        </div>
                                    <% } %>
                                    <% if(status=="2"){ %>
                                        <div class="alert alert-danger" role="alert">
                                            <%= message %>
                                        </div>
                                    <% } %>
                                    <!-- deleting form -->
                                    <form class="mx-1 mx-md-4" action="/deleting-book" method="post" >
                                        <div class="d-flex flex-row align-items-center mb-4">
                                            <div class="form-outline flex-fill mb-0">
                                                <input type="text" class="form-control" name="bookName" />
                                                <label class="form-label" for="bookName">Book Name</label>
                                            </div>
                                        </div>
                                        <div class="d-flex flex-row align-items-center mb-4">
                                            <div class="form-outline flex-fill mb-0">
                                                <input type="text" class="form-control" name="authorName"/>
                                            </div>
                                        </div>
                                    </form>
                                </div>
                            </div>
                        </div>
                    </div>
                </div>
            </div>
        </div>
    </section>
</body>
```

Source code for “adding-book.ejs” file :

```
<!DOCTYPE html>
<html lang="en">

<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Adding Book</title>
```

```

<link rel="stylesheet"
      href="https://cdn.jsdelivr.net/npm/bootstrap@4.3.1/dist/css/bootstrap.min.css"
      integrity="sha384-gg0yR0iXCbMQv3Xipma34MD+dH/1fQ784/j6cY/iJTQUOhcWr7x9JvoRxT2MZw1T"
      crossorigin="anonymous">
</head>
<body>
    <section style="background-color: #eee;">
        <div class="container h-100">
            <div class="row d-flex justify-content-center align-items-center h-100">
                <div class="col-lg-12 col-xl-11">
                    <div class="card text-black" style="border-radius: 25px;">
                        <div class="card-body p-md-5">
                            <% if(success!=""){ %>
                                <div class="alert alert-success" role="alert">
                                    <%= success %>
                                </div>
                            <% } %>
                            <div class="row justify-content-center">
                                <div class="col-md-10 col-lg-6 col-xl-5 order-2 order-lg-1">
                                    <p class="text-center h1 fw-bold mb-5 mx-1 mx-md-4 mt-4">Add Books</p>
                                    <!-- adding form -->
                                    <form class="mx-1 mx-md-4" action="/adding-book"
method="post" enctype="multipart/form-data">
                                        <div class="d-flex flex-row align-items-center mb-4">
                                            <div class="form-outline flex-fill mb-0">
                                                <input type="text" class="form-control" name="bookName" />
                                                <label class="form-label" for="bookName">Book Name</label>
                                            </div>
                                        </div>
                                        <div class="d-flex flex-row align-items-center mb-4">
                                            <div class="form-outline flex-fill mb-0">
                                                <input type="text" class="form-control" name="authorName" />
                                                <label class="form-label" for="authorName">Author Name</label>
                                            </div>
                                        </div>
                                        <div class="d-flex flex-row align-items-center mb-4">
                                            <div class="form-outline flex-fill mb-0">
                                                <input type="number" class="form-control" name="bookPrice" />
                                                <label class="form-label" for="bookPrice">Book price</label>
                                            </div>
                                        </div>
                                        <div class="d-flex flex-row align-items-center mb-4">
                                            <div class="form-outline flex-fill mb-0">
                                                <label class="form-label" for="form3Example4c">Book Type</label>
                                                <select name="bookType" class="form-control">
                                                    <option value="action"> Action</option>
                                                    <option value="comedy">Comedy</option>
                                                    <option value="drama">Drama</option>
                                                    <option value="fantasy">Fantasy</option>
                                                    <option value="horror">Horror</option>
                                                    <option value="mystery">Mystery</option>
                                                    <option value="Romance">Romance</option>
                                                    <option value="Thriller">Thriller</option>
                                                </select>
                                            </div>
                                        </div>
                                        <div class="d-flex flex-row align-items-center mb-4">
                                            <div class="form-outline flex-fill mb-0">

```

```
<textarea class="form-control" name="bookDescription" cols="40" rows="8" placeholder="Give Description about the"></textarea>
                                <label class="form-label" for="bookDescription">Book Description</label>
                            </div>
                        </div>
                    <div class="d-flex flex-row align-items-center mb-4">
                        <div class="flex-fill mb-0">
                            <input type="file" name="bookImage" />
                            <label class="form-label" for="bookImage">Upload Image</label>
                        </div>
                    </div>
                    <div class="d-flex justify-content-center mx-4 mb-3 mb-lg-4">
                        <button type="submit" class="btn btn-primary btn-lg">submit</button>
                    </div>
                </form>
            </div>
            <div class="col-md-10 col-lg-6 col-xl-7 d-flex align-items-center order-1 order-lg-2">
                
            </div>
        </div>
    </div>
</section>
</body>
</html>
```

Source code for “footer.ejs” file :

```
<footer class="copyright">
  <div class="grid">
    <div class="row" id="element1">
      <div class="col-lg-6">
        <h5>Ardent eBooks</h5>
        <p>We are most loved eBook website on the internet!!</p>
      </div>
      <div class="col-lg-3" id="element2">
        <h6>Helpful Links</h6>
        <ul>
          <a href="privacy-policy">
            <li>Privacy Policy</li>
          </a>
          <a href="refund">
            <li>Refund Policy</li>
          </a>
```

```

        <a href="terms-conditions">
            <li>Terms and conditions</li>
        </a>
        <a href="about">
            <li>About us</li>
        </a>
    </ul>
</div>
<div class="col-lg-3">
    <h6>Get in touch</h6>
    <a href="mailto:ardentebook@gmail.com"> <p id="email-link">hello@ardentebook.in</p></a>
</div>

</div>
</div>

<div>
    <i class="fa-brands fa-facebook-f fa-2xl copyright-icon"></i>
    <i class="fa-brands fa-linkedin fa-2xl copyright-icon"></i>
    <i class="fa-brands fa-twitter fa-2xl copyright-icon"></i>
    <i class="fa-brands fa-instagram fa-2xl copyright-icon"></i>
</div>
<div id="copyright-text">Created by Shrey Mishra | All rights reserved!</div>
</footer>
```

Source code for “header-navbar.ejs” file :

```

<div class="navigation-bar">
    <a href="index">
        <h3 >Ardent eBook</h3>
    </a>
    <div class="links">
        <a href="about">
            <h6>About</h6>
        </a>
        <a href="book">
            <h6>Books</h6>
        </a>
        <a href="reviews">
            <h6>Reviews</h6>
        </a>
        <a href="contact">
            <h6>Contact</h6>
        </a>
    </div>
    <div class="registration">
        <a href="signup"><button class="button-dark">SignUp</button></a>
        <a href="login"><button class="button-light">Login</button></a>
    </div>
</div>
```

```
</div>
</div>
```

Source code for “forgetting-password.ejs” file :

```
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Ardent</title>
    <!-- bootstrap -->
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css" rel="stylesheet"
        integrity="sha384-1BmE4kWBq78iYhFldvKuhfTAU6auU8tT94WrHftjDbrCEXSU1oBoqyl2QvZ6jIW3"
        crossorigin="anonymous">
    <!-- CSS links -->
    <link rel="stylesheet" href="css/styles.css">
    <!-- font awesome -->
    <script src="https://kit.fontawesome.com/ed7230efcb.js"
        crossorigin="anonymous"></script>
    <link rel="shortcut icon" href="/favicon.ico" type="image/x-icon">
</head>

<body>
    <%- include("header-navbar"); %>
    <div class="card text-center"
        style="margin:auto; width: 40vw; height: 80vh; margin-bottom: 10vh; font-size: x-large;">
        <div class="card-header text-white bg-dark">Password Reset</div>
        <div class="card-body px-5">
            <% if(isWrongCredentials){ %>
                <div class="alert alert-danger" style="font-size:large">Wrong
                    Credentials entered</div>
            <% }%>
            <% if(isPasswordUpdated){ %>
                <div class="alert alert-success" style="font-size:large">New
                    password has successfully
                    updated and sent to your email!!<a href="login">Click
                    here to login</a></div>
            <% } %>
            <p class="card-text py-2">
                Enter your email address and we'll send you an email
                with instructions to reset your
                password.
            </p>
            <form action="/forget-password" method="post">
                <div class="form-outline">
                    <label class="form-label" for="typeEmail">Email
                    input :</label>
```

```

                <input type="email" id="typeEmail" class="form-
control my-3" name="email" />
                <div class="form-outline mb-4">
                    <label class="form-label"
for="accounttype">Choose account type:</label>
                    <select name="accountType" class="form-
control">
                        <option
value="user">User/Customer</option>
                        <option value="admin">Admin</option>
                        <option value="seller">Seller</option>
                    </select>
                </div>
            </div>
            <button class="btn btn-outline-dark w-100"
type="submit"> Reset Password</button>
        </form>
        <div class="d-flex justify-content-between mt-4 " >
            <a class="text-dark forget-password-btn"
href="login">Login</a>
            <a class="text-dark forget-password-btn"
href="signup">Register</a>
        </div>
    </div>
    <%- include("./footer"); %>
    <script src="https://code.jquery.com/jquery-3.3.1.slim.min.js"
        integrity="sha384-
q8i/X+965Dz00rT7abK41JStQIAqVgRVzbzo5smXKp4YfRvH+8abTE1Pi6jizo"
        crossorigin="anonymous"></script>
    <script
src="https://cdn.jsdelivr.net/npm/popper.js@1.14.7/dist/umd/popper.min.js"
        integrity="sha384-
U02eT0CpHqdSJQ6hJty5KVphPhzWj9W01clHTMGa3JDZwrnQq4sF86dIHNDz0W1"
        crossorigin="anonymous"></script>
    <script
src="https://cdn.jsdelivr.net/npm/bootstrap@4.3.1/dist/js/bootstrap.min.js"
        integrity="sha384-
JjSmVgyd0p3pXB1rRibZUAYoIIy60rQ6VrjIEaFf/nJGzIxFDsf4x0xIM+B07jRM"
        crossorigin="anonymous"></script>
</body>
</html>

```

Source code for “login.ejs” file :

```

<!doctype html>
<html lang="en">
<head>
    <!-- Required meta tags -->
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-
fit=no">
    <!-- Bootstrap CSS -->

```

```

        <link rel="stylesheet"
        href="https://cdn.jsdelivr.net/npm/bootstrap@4.3.1/dist/css/bootstrap.min.css"
            integrity="sha384-
gg0yR0iXCbMQv3Xipma34MD+dH/1fQ784/j6cY/iJTQUOhcWr7x9JvoRxT2MZw1T"
crossorigin="anonymous">
    <title>Login</title>
</head>
<body>
    <section class="vh-100">
        <div class="container py-5 h-100">
            <div class="row d-flex align-items-center justify-content-center h-100">
                <div class="col-md-8 col-lg-7 col-xl-6">
                    
                </div>
                <div class="col-md-7 col-lg-5 col-xl-5 offset-xl-1">
                    <form action="/login" method="post">
                        <p class="text-center h1 fw-bold mb-5 mx-1 mx-md-4 mt-4">Log
in</p>
                        <% if(status=="2" ) { %>
                            <div class="alert alert-danger">
                                <%= message %>
                            </div>
                        <% } %>
                        <% if(status=="3" ) { %>
                            <div class="alert alert-danger">
                                <%= message %>
                            </div>
                        <% } %>
                        <!-- Email input -->
                        <div class="form-outline mb-4">
                            <input type="email" id="form1Example13"
class="form-control form-control-lg"
                                name="email" />
                            <label class="form-label"
for="form1Example13">Email address</label>
                        </div>
                        <!-- Password input -->
                        <div class="form-outline mb-4">
                            <input type="password" id="form1Example23"
class="form-control form-control-lg"
name="passwords" />
                            <label class="form-label"
for="form1Example23">Password</label>
                        </div>
                        <div class="form-outline mb-4">
                            <label class="form-label"
for="accounttype">Choose account type:</label>
                            <select name="accountType" class="form-
control">
                                <option
value="user">User/Customer</option>
                                <option value="admin">Admin</option>
                                <option value="seller">Seller</option>
                            </select>
                        </div>
                        <!-- Submit button -->
                </form>
            </div>
        </div>
    </section>
</body>

```

```

                                <button type="submit" class="btn btn-dark btn-lg
btn-block">Sign in</button>
                                <button type="submit" class="btn btn-dark btn-lg
btn-block"> <a href="forget-password" style="color: white;">Forget Password</a></button>
                                <br>
                                <a href="signup">
                                    <button type="submit" class="btn btn-outline-
dark btn-lg btn-block">Sign Up</button>
                                </a>
                            </form>
                            <!-- Optional JavaScript -->
                            <!-- jQuery first, then Popper.js, then Bootstrap JS -->
                            <script src="https://code.jquery.com/jquery-3.3.1.slim.min.js"
                                   integrity="sha384-
q8i/X+965Dz00rT7abK41JStQIAqVgRVzbzo5smXKp4YfRvH+8abtTE1Pi6jizo"
                                   crossorigin="anonymous"></script>
                            <script
src="https://cdn.jsdelivr.net/npm/popper.js@1.14.7/dist/umd/popper.min.js"
                                   integrity="sha384-
U02eT0CpHqdSJQ6hJty5KVphPhzWj9W01clHTMGa3JDZwrnQq4sF86dIHNDz0W1"
                                   crossorigin="anonymous"></script>
                            <script
src="https://cdn.jsdelivr.net/npm/bootstrap@4.3.1/dist/js/bootstrap.min.js"
                                   integrity="sha384-
JjSmVgyd0p3pXB1rRibZUAYoIIy60rQ6VrjIEaFf/nJGzIxFDsf4x0xIM+B07jRM"
                                   crossorigin="anonymous"></script>
</body>
</html>

```

Source code for “review.ejs” file :

```

<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Contact</title>
    <!-- bootstrap -->
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css"
rel="stylesheet"
      integrity="sha384-
1BmE4kWBq78iYhFldvKuhfTAU6auU8tT94WrHftjDbrCEXSU1oBoqyl2QvZ6jIW3"
crossorigin="anonymous">

    <!-- js -->
    <script
src="https://cdn.jsdelivr.net/npm/@popperjs/core@2.10.2/dist/umd/popper.min.js"
      integrity="sha384-
7+zCNj/IqJ95wo16oMtfSkbZ9ccEh31e0z1HGyDuCQ6wgnyJNSYdrPa03rtR1zdB"
      crossorigin="anonymous"></script>
    <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/js/bootstrap.min.js"

```

```

        integrity="sha384-
QJHtvGhmr9XOIpI6YVutG+2QOK9T+ZnN4kzFN1RtK3zEFEIsxhlmWl5/YESvpZ13"
        crossorigin="anonymous">></script>
<!-- CSS links -->
<link rel="stylesheet" href="css/styles.css">

<!-- icon link -->
<link rel="icon" href="favicon.ico">
</head>

<body>
<!-- navigation bar -->
<%- include('./header-navbar'); %>
<div class="contact-form">
    <p id="form">Feedback</p>
    <form action="/review" method="post">
        <% if(isSubmitted){ %>
            <div class="alert alert-success">Your feedback is valuable for us.
Thanks for your response!!</div>
        <% } %>
            <label for="name">NAME</label>
            <input type="text" placeholder="Enter your name" name="name">
            <label for="email">EMAIL</label>
            <input type="email" placeholder="Enter your email address"
name="email">
            <label for="mobile">CONTACT</label>
            <input type="number" placeholder="Enter your mobile number"
name="mobileNo">
            <label for="message">FEEDBACK</label>
            <textarea name="feedback" cols="30" rows="10" maxlength="200"
placeholder="Type you feedback here!!"
name="message"></textarea>
            <button class="button-dark" type="submit">Submit</button>
        </form>
    </div>
    <!-- footer -->
    <%- include("./footer"); %>
</body>
</html>

```

Source code for “seller-module.ejs” file :

```

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Ardent</title>
    <!-- bootstrap -->
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css"
rel="stylesheet"
        integrity="sha384-1BmE4kWBq78iYhFldvKuhfTAU6auU8tT94WrHftjDbrCEXSU1oBoqyl2QvZ6jIW3"
        crossorigin="anonymous">

```

```

<!-- CSS links -->
<link rel="stylesheet" href="css/styles.css">
<!-- font awesome -->
<script src="https://kit.fontawesome.com/ed7230efcb.js"
crossorigin="anonymous"></script>
<link rel="shortcut icon" href="/favicon.ico" type="image/x-icon">
</head>
<body>
<%- include("header-navbar"); %>
<div id="seller-row">
<a href="adding-book">
<div class="func">
<i class="fa fa-book" aria-hidden="true"></i>
<div>Add Books</div>
</div>
</a>
<a href="deleting-book">
<div class="func"><i class="fa-solid fa-trash"></i>
<div>Delete Books</div>
</div>
</a>
<a href="book">
<div class="func">
<i class="fa-solid fa-book-open"></i>
<div> All Books</div>
</div>
</a>
<a href="contact">
<div class="func">
<i class="fa-solid fa-address-book"></i>
<div>Contact</div>
</div>
</a>
</div>
<%- include("footer"); %>
<!-- Optional JavaScript -->
<!-- jQuery first, then Popper.js, then Bootstrap JS -->
<script src="https://code.jquery.com/jquery-3.3.1.slim.min.js"
integrity="sha384-q8i/X+965Dz00rT7abK41JStQIAqVgRVzbzo5smXKp4YfRvH+8abtTE1Pi6jizo"
crossorigin="anonymous"></script>
<script src="https://cdn.jsdelivr.net/npm/popper.js@1.14.7/dist/umd/popper.min.js"
integrity="sha384-U02eT0CpHqdSJQ6hJty5KVphtPhzWj9W01clHTMGa3JDZwrnQq4sF86dIHNDz0W1"
crossorigin="anonymous"></script>
<script src="https://cdn.jsdelivr.net/npm/bootstrap@4.3.1/dist/js/bootstrap.min.js"
integrity="sha384-JjSmVgyd0p3pXB1rRibZUAYoIIy6OrQ6VrjIEaFf/nJGzIxFDsf4x0xIM+B07jRM"
crossorigin="anonymous"></script>
</body>
</html>

```

Source code for “signup.ejs” file :

```

<!DOCTYPE html>
<html lang="en">

<head>
    <!-- Required meta tags -->
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
    <!-- Bootstrap CSS -->
    <link rel="stylesheet"
    href="https://cdn.jsdelivr.net/npm/bootstrap@4.3.1/dist/css/bootstrap.min.css"
        integrity="sha384-gg0yR0iXCBMQv3Xipma34MD+dH/1fQ784/j6cY/iJTQUOhcWr7x9JvoRxT2MZw1T"
    crossorigin="anonymous">
    <style>
        .password-not-matched {
            color: #ff9933;
        }

        .password-matched {
            color: #33cc33;
        }
    </style>
    <title>Signup</title>
    <link rel="shortcut icon" href="/favicon.ico" type="image/x-icon">
</head>

<body>
    <section style="background-color: #eee;">
        <div class="container h-100">
            <div class="row d-flex justify-content-center align-items-center h-100">
                <div class="col-lg-12 col-xl-11">
                    <div class="card text-black" style="border-radius: 25px;">
                        <div class="card-body p-md-5">
                            <div class="row justify-content-center">
                                <div class="col-md-10 col-lg-6 col-xl-5 order-2 order-lg-1">
                                    <p class="text-center h1 fw-bold mb-5 mx-1 mx-md-4 mt-4">Sign up</p>
                                    <% if(status === "2" ) { %>
                                        <div class="alert alert-danger">
                                            <%= message %><a href="login">Click here to login</a>
                                        </div>
                                    <% } %>
                                    <% if(status === "3" ) { %>
                                        <div class="alert alert-success">
                                            <%= message %>
                                            <a href="login">Click here</a>
                                        </div>
                                    <% } %>
                                    <form class="mx-1 mx-md-4" action="/signup" method="post">
                                        <div class="d-flex flex-row align-items-center mb-4">
                                            <div class="form-outline flex-fill mb-0">
                                                <input type="text" id="form3Example1c" class="form-control" name="name" />
                                                <label class="form-label" for="form3Example1c">Your
                                                Name</label>
                                            </div>
                                        </div>
                                        <div class="d-flex flex-row align-items-center mb-4">
                                            <div class="form-outline flex-fill mb-0">
                                                <input type="email" class="form-control" name="email" />
                                            </div>
                                        </div>
                                    </form>
                                </div>
                            </div>
                        </div>
                    </div>
                </div>
            </div>
        </div>
    </section>
</body>

```

```

        <label class="form-label" for="form3Example3c">Your
Email</label>
        </div>
    </div>
    <div class="d-flex flex-row align-items-center mb-4">
        <div class="form-outline flex-fill mb-0">
            <input type="password" class="form-control"
name="password" id="password" />
            <label class="form-label"
for="form3Example4c">Password</label>
            </div>
        </div>
        <div class="d-flex flex-row align-items-center mb-4">
            <div class="form-outline flex-fill mb-0">
                <input type="password" class="form-control"
name="confirmationPassword"
id="confirmation-password" />
                <label class="form-label" for="form3Example4c">Confirm
Password</label>
            </div>
        </div>
        <p id="password-status"></p>
        <div class="d-flex flex-row align-items-center mb-4">
            <div class="form-outline flex-fill mb-0">
                <input type="text" class="form-control" name="address" />
                <label class="form-label"
for="form3Example4c">Address</label>
            </div>
        </div>
        <div class=" d-flex flex-row align-items-center mb-4">
            <div class="form-outline flex-fill mb-0">
                <input type="number" class="form-control"
name="pincode" />
                <label class="form-label"
for="form3Example4c">Pincode</label>
            </div>
        </div>
        <div class=" d-flex flex-row align-items-center mb-4">
            <div class="form-outline flex-fill mb-0">
                <input type="text" class="form-control"
name="mobileNo" />
                <label class="form-label"
for="form3Example4c">Contact</label>
            </div>
        </div>
        <div class=" d-flex flex-row align-items-center mb-4">
            <div class="form-outline flex-fill mb-0">
                <label for="accounttype">Choose account
type:</label>
                <select name="accountType" class="form-control">
                    <option value="user">User/Customer</option>
                    <option value="admin">Admin</option>
                    <option value="seller">Seller</option>
                </select>
            </div>
        </div>
        <button type="submit" class="btn btn-dark btn-block
btn-lg">Register</button>

```

```

        <br>
        <a href="login"><button type="button"
           class="btn btn-outline-dark btn-block btn-lg">Sign
In</button></a>
        </form>
    </div>
    <div class="col-md-10 col-lg-6 col-xl-7 d-flex align-items-center order-1
order-lg-2">
        
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
</section>
<!-- Optional JavaScript -->
<!-- jQuery first, then Popper.js, then Bootstrap JS -->
<script src="https://code.jquery.com/jquery-3.3.1.slim.min.js"
       integrity="sha384-q8i/X+965Dz0rT7abK41JStQIAqVgRVzbzo5smXKp4YfRvH+8abTE1Pi6jizo"
       crossorigin="anonymous"></script>
<script src="https://cdn.jsdelivr.net/npm/popper.js@1.14.7/dist/umd/popper.min.js"
       integrity="sha384-U02eT0CpHqdSJQ6hJty5KVphtPhzWj9W01clHTMGa3JDZwrnQq4sF86dIHNDz0W1"
       crossorigin="anonymous"></script>
<script src="https://cdn.jsdelivr.net/npm/bootstrap@4.3.1/dist/js/bootstrap.min.js"
       integrity="sha384-JjSmVgyd0p3pXB1rRibZUAYoIIy60rQ6VrjIEaFf/nJGzIxFDsf4x0xIM+B07jRM"
       crossorigin="anonymous"></script>

<script src="/index.js"></script>

</body>
</html>

```

Source code for “privacy-policy.ejs” file :

```

<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Ardent</title>
    <!-- bootstrap -->
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css"
          rel="stylesheet"
          integrity="sha384-1BmE4kWBq78iYhFldvKuhfTAU6auU8tT94WrHftjDbrCEXSU1oBoqyl2QvZ6jIW3"
          crossorigin="anonymous">
    <!-- CSS links -->

```

```
<link rel="stylesheet" href="css/styles.css">
<script src="https://kit.fontawesome.com/48f9ae67d0.js"
crossorigin="anonymous"></script>
<!-- font awesome -->
<script src="https://kit.fontawesome.com/ed7230efcb.js"
crossorigin="anonymous"></script>
<link rel="shortcut icon" href="/icon.png" type="image/png">
</head>

<body>
<%- include("./header-navbar"); %>
<!-- main content -->
<div id="privacy-policy">
    <h1>PRIVACY POLICY</h1>
    <p>
        This Privacy Policy discloses the privacy practices for Ardent eBooks ("We/we" or "Us/us") with regard to Your ("You/you" or "Your/your") use of the online platform www.ardentebook.in ("Website"). This Privacy Policy and Terms of Use describes our current practices regarding, including your choices in relation to how we collect, compile, store, use, share and secure your personal information across our Website. It also describes your choices regarding access, use and correction of personal information and your rights in relation to your personal information and how you can contact us or authorities in the event you have a grievance or complaint. By visiting www.ardentebook.in, you agree to be bound by the terms and conditions of this Privacy Policy. Kindly go through this Privacy Policy carefully. By using the Website, you indicate that you understand and abide by this Policy. If you are not in agreement with the terms and conditions of this Privacy Policy, kindly do not access this Website. We ensure to review this Privacy Policy from time to time so that it is updated. If you are just a visitor, then kindly note that this Policy is subject to change at any time without notice. To ensure that you are aware of any changes in the Policy, please review this Policy from time to time. By Personal Data, we mean any information that can either itself identify you as an individual.
    </p>
    <dl>
        <dt> 1. COLLECTION OF INFORMATION </dt>
        <dd>
            As a visitor, you can browse through our Website to find out more about Ardent eBooks. You are not required to provide us with any Personal Data if you are merely a visitor. When you visit the Site, we collect and store certain information in order to increase security, analyse developments and administer the Website with a view to help us in making customer experience better. We utilise this information to analyse traffic patterns on our Website so as to make it more beneficial for our visitors. We collect your Personal Data only when you register with us, when you are interested in
        </dd>
    </dl>
</div>
```

obtaining information regarding our products and services, when you participate in activities on our Website or contact us. We will only collect your Personal Information if there is a proper and valid reason for doing so, for instance : to comply with our regulatory and legal obligations; for the performance of our contract with you or to take the required steps at your request before entering into a legal contract; for our legitimate interests or where you have given consent. A legitimate interest is when we have a valid commercial or business reason to use your information. In certain cases, we may also have a legal obligation to collect personal data from you or may otherwise need the personal information to protect your interests or those of another individual (for instance, in order to prevent payment fraud or confirm your identity). Our primary objective in collecting your Personal Information is to provide you a safe, effective, seamless and customized experience. This allows us to provide online courses, study material and tutorials that most likely meet your needs and to customize our Website to make your experience easier, better and safer. The information we gather from you helps us personalize and continually improve customer experience at our Website. We do not voluntarily make this information available to any third parties, nor do we use it for any other objective, except as set out herein. In order to register on our website or complete the purchase process of one of our courses, we may collect the following information that you will voluntarily provide to us through one of our contact forms, chat or phone session, such as:

1. Information pertaining to personal details, like name, email id, photograph, contact address, contact number, IP address or geographic location
2. Information pertaining to identity, like your mark-sheets, Aadhar /Voter ID/ Passport details;
3. Information pertaining to billing, like credit/debit card number and billing address;
4. Username or password;
5. Ratings or reviews, syllabus/ course preferences, account settings; and
6. Information provided to us through interaction with customer service, surveys or promotions.

We collect data about your interactions with us and also the information regarding the device (Laptop, computer, mobile, etc) used to access our service. This information contains your activity on our Website, and search queries; your interactions details with customer service, and if you call us , your contact number and recordings of your calls; device IDs or other unique identifiers, device and software characteristics, connection information, statistics regarding page views, referral URLs, IP address, browser and web server log information, and information collected via the

use of cookies. With the aid of back end applications, we strive to provide better services and quality interaction with teachers, depending on your interest. We may also seek this information at other instances, such as when you enter contests or other marketing promotions sponsored by us and/or our partners. If you use a functionality that requires fee payment, options will appear. In case of payments via credit or debit card, we will redirect you to registered payment gateway razorpay or paytm. You may store and save your payment details like card numbers with the gateway.

We cannot access this data. All payment transactions are processed through secure payment gateway providers. When you access any of our paid products, we store information about the web pages on our servers. This helps us to track items that have been completed by you, and those that you need to see. The user's full name and his/her display picture can be publicly displayed on the Website.

Ardent eBooks can seek more information regarding billing information, address, certificates, etc depending on what profile and what services they are seeking. Ardent eBooks can use technologies like cookies, clear gifs, log files, etc to help understand how you interact with our site and services, so as to render a better experience.

<dt>2. USAGE AND RETENTION OF INFORMATION</dt>

<dd>We use the personal information we collect, including your Personal Data where it is necessary to render the services requested by you, where it is necessary to comply with legal obligations or rights or for normal business purposes of the kind set out in this Policy. We will use your personal information to provide, analyse and improve our offerings, to provide you with a customized experience on our Website, to contact you about your account and our services, to render customer service, to provide you with personalized marketing and to any fraudulent or illegal activities. In addition to this, we use your personal information to figure out your general geographic location, provide localized course content and classes, recommendations, determine your ISP, and help us quickly and efficiently respond to queries and enforcing our terms and communicate with you concerning our service (for example by email, push notifications, text messaging ,and online messaging channels), so that we can provide you details about latest features and content available on the Website, offers, and promotions, surveys, and to assist you with operational requests like password reset requests.

<dt> 3. COMMUNITY </dt>

Ardent eBooks is a community, wherein we offer a number of features that allow members to connect and

communicate in public or semi-public spaces, such as Forums and Teams. Please use your judgment before posting in these community spaces or sharing your personal information with others on Ardent eBooks. You should be aware that any personal information you submit there can be read, collected, or used by others, or could be used to send you unsolicited messages.</dd>

<dt> 4. SHARING AND DISCLOSING PERSONAL INFORMATION </dt>

<dd>We use other Service Providers to perform services on our behalf or to assist us with the provision of services to you. We engage these Service Providers to provide promotional, infrastructure and IT services, to personalize and optimize our service, to process payment transactions, to provide customer service, to analyze and enhance data, and to conduct consumer surveys. In the process of providing such services, these Service Providers may have access to your personal or other information. We do not authorize them to use or disclose your personal information except in relation to providing their services.</dd>

<dt> 5. NOTE TO OUR USERS IN EUROPE:</dt>

<dd>We transfer your personal information from the European Economic Area and Switzerland to India. By submitting your data and/or using our services, you consent to the transfer, storing, and processing of your personal information in India.</dd>

<dt> 6. SECURITY </dt>

<dd>We shall make an effort to take all precautions to shield the personal information both online and offline. We will try to protect your information using security measures to reduce the risks of loss and misuse of data. We have standard SSL certification which basically helps us create a secure connection between our server and user to render any information or action. Some of the protective measures we use are firewalls and data encryption, physical access controls to our data centres and information access authorization controls. Only user passwords are encrypted in 64-bit encryption and stored because generally users use the same password on multiple sites, to prevent any kind of theft, piracy or unauthorised access. If you think that your Ardent eBooks account has been compromised, you can contact us by writing an email to support@ardentebook.in.</dd>

<dt>7. CHANGES TO THIS POLICY</dt>

<dd>Ardent eBooks reserves the right to change this Privacy Policy as we may deem necessary from time to time or as may be required by law. Any changes will be immediately posted on the Web Site and you are deemed to have accepted the terms of the Policy on your first use of the Web Site following the alterations.</dd>

</dl>

</div>

<%- include("./footer"); %>

```
</body>  
</html>
```

Source code for stylesheet (styles.css) file :

```
@import url('https://fonts.googleapis.com/css2?family=Merienda&display=swap');  
@import url('https://fonts.googleapis.com/css2?family=Lobster&display=swap');  
@import url('https://fonts.googleapis.com/css2?family=PT+Serif&display=swap');  
@import url('https://fonts.googleapis.com/css2?family=Roboto+Slab&display=swap');
```

```
body {  
    font-family: 'PT Serif', serif;  
    line-height: 1.5;  
    text-align: center;  
    color: black;  
}  
  
a {  
    text-decoration: none;  
}  
  
.about-us-content {  
    width: 90vw;  
    text-align: left;  
    margin: auto;  
    margin-top: 40px;  
    font-family: 'PT Serif', serif;  
}  
  
.about-us {  
    color: white;  
    background-color: #301E67;  
    padding: 120px;  
    width: 95vw;  
    margin: auto;  
    font-family: 'PT Serif', serif;  
    font-size: 90px;  
}  
  
.address {  
    margin-top: 228px;  
    padding: 42px;  
    text-align: left;  
}  
  
.address a {  
    color: grey;  
}  
  
.address p {  
    margin-bottom: 40px;  
}
```

```
.button-light {
    margin: auto;
    margin-right: 60px;
    width: 142px;
    height: 49px;
    background-color: white;
    color: black;
    border: 1px solid black;
}

.button-light:hover {
    background-color: black;
    color: white;
}

#carousel-img {
    height: 5%;
}

.button-dark {
    margin: auto;
    margin-right: 60px;
    width: 142px;
    height: 49px;
    background-color: black;
    color: white;
    border: none;
}

.button-dark:hover {
    background-color: white;
    border: 1px solid black;
    color: black;
}

.links h6:hover {
    color: black;
    font-size: 110%;
}

.terms-conditions h2 {
    font-family: 'PT Serif', serif;
    font-size: 80px;
}
.detail-form-section{
    background-color: #cae1da;
    width: 80vw;
    height: 80vh;
    margin: 10vh auto;
}

.detail-form-row{
    height : 80vh;
}

.detail-form-col{
    display : flex ;
```

```
    align-items: center;
    justify-content: center;
}

#library-img{
    width: 80%;
}

.terms-conditions-content dl {
    width: 80vw;
    margin: auto;
    text-align: left;
    margin-top: 40px;
}

.terms-conditions-content dd {
    color: grey;
}

.terms-conditions-content dt {
    margin: 20px;
    margin-left: 0px;
    font-size: x-large;
}

.background-img {
    height: 600px;
    width: 1400px;
}

#privacy-policy {
    width: 80vw;
    margin: auto auto 40px;
    text-align: left;
}

#privacy-policy h1 {
    text-align: center;
    font-family: 'PT Serif', serif;
    font-size: 80px;
    margin-bottom: 8vh;
}

#privacy-policy dt {
    margin: 40px;
    margin-left: 0px;
    font-size: x-large;
}

#privacy-policy dd {
    color: grey;
}

#note {
    margin-top: 5vh;
    font-weight: bolder;
}
```

```
.blog {  
    width: 1192px;  
    margin: auto;  
    margin-top: 82px;  
}  
  
.blog h1 {  
    margin: auto;  
}  
  
.blog-items img {  
    width: 100%;  
    margin: 40px auto 40px;  
}  
  
.blog-items h2 {  
    margin-top: 20px;  
    margin-bottom: 10px;  
}  
  
.blog-items-text {  
    text-align: left;  
    width: 70%;  
    margin: auto;  
}  
  
.column {  
    text-align: left;  
    padding: 30px;  
}  
  
.contact-us-link {  
    display: flex;  
    justify-content: center;  
    align-items: center;  
}  
  
.contact-form {  
    width: 50vw;  
    border: 1px solid rgb(209, 199, 199);  
    margin: 50px auto;  
    padding: 72px;  
    text-align: left;  
}  
  
.contact-form button {  
    margin-left: 258px;  
    margin-top: 26px  
}  
  
.contact-form p {  
    margin-bottom: 30px;  
}  
  
.contact-form input {  
    margin-bottom: 30px;  
    width: 100%;
```

```

        border: 1px solid rgb(209, 199, 199);
        padding: 7px;
    }

.contact-form textarea {
    border: 1px solid rgb(209, 199, 199);
    width: 100%;
    padding: 7px;
}

.contact-team {
    display: flex;
    justify-content: space-between;
    width: 80vw;
    margin: 70px auto 70px;
}

.contact-team img {
    width: 330px;
    background-color: black;
}

.contact-team a {
    color: grey;
}

.books-navbar {
    display: flex;
    justify-content: space-evenly;
    padding: 10px 40px;
    background-color: #B6EADA;
}

.card-registration .select-input.form-control[readonly]:not([disabled]) {
    font-size: 1rem;
    line-height: 2.15;
    padding-left: .75em;
    padding-right: .75em;
}
    .card-registration .select-arrow {
    top: 13px;
}
.forget-password-btn:hover{
    font-size: 110%;

}

#seller-row{
    width:80vw;
    margin:auto;
    display: flex;
    justify-content: space-evenly;
    margin-top: 10vh;
    margin-bottom: 10vh;
    font-size: x-large;
}
.func{
    background-color: #9fafafa39;
    padding: 20px;
}

```

```

        width: 17vw;
        color: black;
    }
    .func i:hover{
        font-size: 110%;
    }

    .books-navbar h3 {
        font-family: 'Merienda', cursive;
        font-size: 40px;
    }
    .wrong-password-status{
        background-color: rgb(239, 145, 145);
        color: black;
    }

    .books-navbar form {
        display: flex;
    }
    .registration-form
    {
        width: 80vw;
        height: 80vh;
        margin: auto;
        padding: 10px;
        /* background-color: #e1dfdf; */
        margin-top: 40px;
        background-image: url(https://images.unsplash.com/photo-1481627834876-b7833e8f5570?ixlib=rb-4.0.3&ixid=MnwxMjA3fDB8MHxzZWFiY2h8Mnx8bGlicmFyeXxlbnwwfHwwfHw%3D&w=1000&q=80);
    }

    .btn-outline-success {
        background-color: white;
        margin-left: 10px;
        border: #301E67 1px solid;
        color: #301E67;
    }

    .btn-outline-success:hover {
        background-color: #301E67;
        color: white;
    }

    .form-inline {
        display: flex;
        height: 50px;
        margin: auto;
        margin-right: 20px;
    }

    #form {
        margin-left: 40%;
        font-size: 50px;
        color: black;
    }

```

```
}

.buy-now {
    margin-top: 20px;
}

.content {
    background-image: url("https://wallpapercave.com/wp/wp2853390.jpg");
    color: white;
    height: 600px;
    width: 1462px;
    margin: auto;
    display: flex;
    justify-content: center;
    align-items: center;
    font-family: 'Lobster', cursive;
}

.contents {
    display: flex;
    margin: 170px;
    text-align: left;
}

.content-img {
    height: 380px;
}

.content-text {
    margin-right: 100px;
}

.content-text h6,
p {
    color: grey;
}

.content-text-2 {
    margin-left: 100px;
}

.content-text-3 h6 {
    color: grey;
    margin-top: 40px;
}
.me-3{
    margin: 100px;
}

.content-text-4 {
    display: flex;
    margin: 50px 166px;
}

.content-con {
    text-align: center;
}
```

```
.content-con h1 {
    font-size: 70px;
}

.content-con h5 {
    color: rgb(133, 131, 131);
}

.content-con button {
    background-color: white;
    border: none;
    padding: 18px 29px;
    margin-top: 24px;
}

.contact-detail {
    display: flex;
    justify-content: space-evenly;
}

.copyright {
    padding-bottom: 2vh;
    background-color: #B6EADA;
    color: #301E67;
    font-size: larger;
}

.btn-success {
    margin-bottom: 40px;
}

.card {
    margin: 20px;
    width: 18rem;
}

.card-img-top {
    align-self: center;
    margin-top: 10px;
}

#news-letter-form {
    display: flex;
    margin-top: 30px;
}

.card-img-top-round {
    width: 100px;
    height: 100px;
    border-radius: 100%;
}

.card-deck {
    display: flex;
    justify-content: center;
    margin: 5vh 5vh;
}
```

```
.btn-outline-info {  
    color: #B6EADA;  
    border-color: #B6EADA;  
}  
  
.btn-outline-info:hover {  
    background-color: #B6EADA;  
    border-color: #301E67;  
    color: #301E67;  
}  
  
.card-group {  
    margin-top: 5vh;  
}  
  
.title-heading {  
    border: 1px solid #B6EADA;  
    display: inline;  
    padding: 10px;  
    color: cadetblue;  
    margin-bottom: 10vh;  
}  
  
.copyright-icon {  
    margin: 1vw;  
    margin-bottom: 4vh;  
}  
  
.feature-box {  
    margin-top: 1rem;  
    display: flex;  
    justify-content: center;  
}  
  
.feature {  
    margin: 5vh;  
}  
  
.feature p {  
    margin-top: 1vh;  
}  
  
hr {  
    width: 80%;  
    margin: auto;  
    margin-top: 40px;  
}  
  
.links {  
    display: flex;  
    width: 70vw;  
    justify-content: center;  
    align-items: center;  
}  
  
.links h6 {  
    padding: 35px;  
    color: grey;  
}
```

```
}

.navigation-bar {
    display: flex;
}

.navigation-bar h3 {
    margin-top: 41px;
    margin-left: 37px;
    color: black;
    font-family: 'Merienda', cursive;
}

.copyright {
    padding-top: 20px;
}

.copyright h5 {
    font-family: 'Merienda', cursive;
    font-size: 50px;
}

.copyright h6 {
    font-family: 'PT Serif', serif;
    font-size: 20px;
    color: #301E67;
}

.copyright ul li {
    color: grey;
    list-style-image: url(/images/211607_right_arrow_icon\ \(2\).png);
    list-style-position: outside;
    text-align: left;
}

.refund-text {
    height: 70vh;
    padding-top: 20vh;
    align-items: center;
}

.refund-text h1 {
    font-family: 'Roboto Slab', serif;
    font-size: 300%;
}

.refund-text h5 {
    margin-top: 5vh;
    font-size: x-large;
    width: 80vw;
    margin: auto;
}

#element2 {
    text-align: left;
}

#element2 h6 {
```

```
    padding-left: 30px;
}

#element1 {
    margin: 30px;
}

.quote {
    text-align: center;
    margin-top: 130px;
}

.quote h6 {
    color: grey;
}

.quote h1 {
    margin: auto;
    width: 960px;
}

.text-2 {
    width: 70vw;
    margin: auto;
    margin-top: 110px;
}

.text-2 p {
    margin-top: 80px;
}

.text-3 h6 {
    margin-top: 50px;
}

.text-3 h1 {
    margin-bottom: 40px;
}

.text-4 {
    display: flex;
    justify-content: space-between;
    text-align: left;
    width: 70vw;
    margin: auto;
}

.text-4 h5 {
    margin-bottom: 20px;
}

.text-6 {
    margin: 75px auto 75px;
}

.text-6 h6 {
    color: grey;
}
```

```

.text-7 img {
    width: 80vw;
}

.text-7 img:hover {
    opacity: 70%;
}

.text-7 h3 {
    margin-top: 40px;
}

.text-7 h4 {
    margin-bottom: 40px;
    color: grey;
}

.logo-img {
    width: 20%;
    margin-bottom: 30px;
}

#element2 li:hover {
    font-size: 105%;
}

#email-link:hover {
    font-size: 105%;
}

.box {
    padding-right: 77px;
    margin-top: 35px;
}

.registration {
    display: flex;
    align-items: center;
}

```

Source code for “app.js” file :

```

const mongoose = require("mongoose");
const express = require("express");
const ejs = require('ejs');
const bodyParser = require("body-parser");
const bcrypt = require("bcrypt");
const nodemailer = require("nodemailer");
const multer = require("multer");
const app = express();
var fs = require('fs');
var path = require('path');
const saltRounds = 10;

```

```

//Schema declarations
const customerSchema = new mongoose.Schema({
    customer_name: String,
    customer_email: String,
    customer_mobileNo: Number,
    customer_password: String,
    customer_address: String,
    customer_pincode: Number
});

const sellerSchema = new mongoose.Schema({
    seller_name: String,
    seller_email: String,
    seller_mobileNo: Number,
    seller_password: String,
    seller_address: String,
    seller_pincode: Number
});

const feedbackSchema = new mongoose.Schema({
    reviewer_name: String,
    reviewer_email: String,
    reviewer_mobileNo: Number,
    review: String
});

const booksSchema = new mongoose.Schema({
    book_name: String,
    book_author: String,
    book_type: String,
    book_description: String,
    book_price: Number,
    book_img:
    {
        data: Buffer,
        contentType: String
    }
});

const adminSchema = new mongoose.Schema({
    admin_name: String,
    admin_mobileNo: Number,
    admin_password: String,
    admin_email: String,
    admin_address: String,
    admin_pincode: Number
});

const paymentSchema = mongoose.Schema({
    payer_name: String,
    payer_email: String,
    payment_amount: Number,
    payment_status: Boolean,
    payment_date: Date
})

```

```

});

//Model declarations
const customerModel = new mongoose.model('Customer', customerSchema);
const sellerModel = new mongoose.model('Seller', sellerSchema);
const feedbackModel = new mongoose.model('Feedback', feedbackSchema);
const bookModel = new mongoose.model('Book', booksSchema);
const paymentModel = new mongoose.model('Payment', paymentSchema);
const adminModel = new mongoose.model('admin', adminSchema);

//setting of engines and usage
app.set('view engine', 'ejs');
app.use(express.static("public"));
app.use(bodyParser.urlencoded({
    extended: true
}));

//connecting with the database MONGODB
mongoose.set("strictQuery", true);
mongoose.connect("mongodb+srv://admin-
shrey:shrey001@cluster0.kto3pxv.mongodb.net/eBookDB", {
    useNewUrlParser: true,
    useUnifiedTopology: true
}, (err) => {
    if (err) console.log(err);
    else {
        console.log("connected to db");
    }
});

var storage = multer.diskStorage({
    destination: (req, file, cb) => {
        cb(null, 'uploads')
    },
    filename: (req, file, cb) => {
        cb(null, file.fieldname + '-' + Date.now())
    }
});

var upload = multer({ storage: storage });

//get request handlers
app.get("/", (req, res) => {
    bookModel.find((err, books) => {
        if (err)
            console.log(err);
        else {
            res.render("index", { books });
        }
    });
});

```

```

app.get("/about", (req, res) => {
    res.render("about");
});

app.get("/index", (req, res) => {
    bookModel.find((err, books) => {
        if (err)
            console.log(err);
        else {

            res.render("index", { books });
        }
    });
});

app.get("/blog", (req, res) => {
    res.render("blog");
});
app.get("/contact", (req, res) => {
    res.render("contact", { isSubmitted: false });
});

app.get("/work", (req, res) => {
    res.render("work");
});
app.get("/signup", (req, res) => {
    res.render("signup", { status: "1" });
});

app.get("/login", (req, res) => {
    res.render("login", { status: "1" });
});

app.get("/forget-password", (req, res) => {
    res.render("forget-password", { isWrongCredentials: false, isPasswordUpdated: false });
});

app.get("/privacy-policy", (req, res) => {
    res.render("privacy-policy");
});

app.get("/terms-conditions", (req, res) => {
    res.render("./terms-conditions")
});
app.get("/about", (req, res) => {
    res.render("./about");
});
app.get("/reviews", (req, res) => {
    res.render("reviews");
});

app.get("/book", (req, res) => {
    bookModel.find((err, books) => {

```

```

        if (err)
            console.log(err);
        else {
            res.render("book", { searchedBookFound: "true", books });
        }
    });
});

app.get("/review", (req, res) => {
    res.render("review", { isSubmitted: false });
});

app.get("/seller-module", (req, res) => {
    res.render("seller-module");
});

app.get("/adding-book", (req, res) => {
    res.render("adding-book", { success: "" });
});

app.get("/refund", (req, res) => {
    res.render("refund");
});

app.get("/deleting-book", (req, res) => {
    res.render("deleting-book", { status: "0" });
});

// post request handlers

app.post("/signup", (req, res) => {
    if (req.body.password === req.body.confirmationPassword) {

        const accountType = req.body.accountType;
        console.log(accountType);
        //add the username already existed to all other module.
        if (accountType === "user") {
            customerModel.findOne({ customer_email: req.body.email }, (err, foundOne) =>
{
                if (err) console.log(err);
                if (foundOne) {
                    res.render("signup", { status: "2", message: "Username already
existed!! Try with another email or " });
                }
                else {
                    bcrypt.hash(req.body.password, saltRounds, function (err, hash) {
                        if (err) console.log(err);
                        const newCustomer = new customerModel({
                            customer_name: req.body.name,
                            customer_email: req.body.email,
                            customer_password: hash,
                            customer_mobileNo: req.body.mobileNo,
                            customer_address: req.body.address,
                            customer_pincode: req.body.pincode
                        });
                        newCustomer.save((err) => {
                            if (err) console.log(err);
                        });
                    });
                }
            });
        }
    }
});

```

```

        });
    });
    res.render("signup", { status: "3", message: "Account successfully
created. You can login into the account!!" });
}
});

else if (accountType === "seller") {
    bcrypt.hash(req.body.password, saltRounds, function (err, hash) {
        const newSeller = new sellerModel({
            seller_name: req.body.name,
            seller_email: req.body.email,
            seller_password: hash,
            seller_mobileNo: req.body.mobileNo,
            seller_address: req.body.address,
            seller_pincode: req.body.pincode
        });
        newSeller.save((err) => {
            if (err) console.log(err);
        });
    });
}

res.render("signup", { status: "3", message: "Account successfully created.
You can login into the account!!" });
}
else {
    bcrypt.hash(req.body.password, saltRounds, function (err, hash) {
        const newAdmin = new adminModel({

            admin_name: req.body.name,
            admin_password: hash,
            admin_email: req.body.email,
            admin_mobileNo: req.body.mobileNo,
            admin_address: req.body.address,
            admin_pincode: req.body.pincode
        });
        newAdmin.save((err) => {
            if (err) console.log(err);
        });
    });
}

res.render("signup", { status: "3", message: "Account successfully created.
You can login into the account!!" });
}
} else {
    res.render("signup", { status: "2", message: "Password doesn't match!!" });
}
});

app.post("/review", (req, res) => {
    const feedback = new feedbackModel({
        reviewer_name: req.body.name,
        reviewer_email: req.body.email,
        reviewer_mobileNo: req.body.mobileNo,
        review: req.body.feedback
    });
    feedback.save((err) => {
        console.log(err);
    });
});

```

```

        res.render("review", { isSubmitted: true });
    });

app.post("/book", (req, res) => {
    const nameOfSearchedBook = req.body.searchedBook;
    if (nameOfSearchedBook === "") res.render("book");
    else {
        books.findOne({ book_name: nameOfSearchedBook }, (err, foundOne) => {
            if (err) console.log(err);
            if (foundOne) { console.log(true); }
            else {
                console.log("false");
            }
        });
    }
});

app.post("/adding-book", (req, res) => {

    const newBook = new bookModel({
        book_name: req.body.bookName,
        book_author: req.body.authorName,
        book_type: req.body.bookType,
        book_description: req.body.bookDescription,
        book_price: req.body.bookPrice,
    });
});

app.post("/login", (req, res) => {
    const email = req.body.email;
    const accountType = req.body.accountType;
    if (accountType === "user") {
        customerModel.findOne({ customer_email: email }, (err, foundOne) => {
            if (err) console.log(err);
            if (foundOne) {
                bcrypt.compare(req.body.passwords, foundOne.customer_password, function
                (err, result) {
                    // result == true
                    if (result) {
                        res.redirect("index");
                    }
                    else {
                        res.render("login", { status: "2", message: "Incorrect
Password!!" });
                    }
                });
            } else {
                res.render("login", { status: "3", message: "Incorrect Username/email!!"
});
            }
        });
    }
    else if (accountType === "seller") {
        sellerModel.findOne({ seller_email: email }, (err, foundOne) => {
            if (err) console.log(err);
            if (foundOne) {
                bcrypt.compare(req.body.passwords, foundOne.seller_password, function
                (err, result) {
                    // result == true

```

```

        if (result) {
            res.render("seller-module");
        }
        else {
            res.render("login", { status: "2", message: "Incorrect
Password!!" });
        }
    });

} else {
    res.render("login", { status: "3", message: "Incorrect Username/email!!"
});
}
};

}

else {
    adminModel.findOne({ admin_email: email }, (err, foundOne) => {
        if (err) console.log(err);
        if (foundOne) {
            bcrypt.compare(req.body.passwords, foundOne.admin_password, function
(err, result) {
                // result == true
                if (result) {
                    res.render("admin-module");
                }
                else {
                    res.render("login", { status: "2", message: "Incorrect
Password!!" });
                }
            });
        }
        else {
            res.render("login", { status: "3", message: "Incorrect Username/email!!"
});
        }
    });
}
};

app.post("/contact", (req, res) => {

const userFeedback = new feedbackModel({
    reviewer_name: req.body.name,
    reviewer_email: req.body.email,
    reviewer_mobileNo: req.body.mobileNo,
    review: req.body.message
});

userFeedback.save((err) => {
    console.log(err);
});

res.render("contact", { isSubmitted: true });
});

app.post('/adding-book', upload.single('bookImage'), (req, res, next) => {

const newBook = new bookModel({

```

```

        book_name: req.body.bookName,
        book_author: req.body.authorName,
        book_type: req.body.bookType,
        book_description: req.body.bookDescription,
        book_price: req.body.bookPrice,
        book_img: {
            data: fs.readFileSync(path.join(__dirname + '/uploads/' +
req.file.filename)),
            contentType: 'image/png'
        }
    });
bookModel.create(newBook, (err, item) => {
    if (err) {
        console.log(err);
    }
    else {
        item.save();
        res.render("adding-book", { success: "Voila!! Book added successfully" })
    }
});
});

app.post("/deleting-book", (req, res) => {

    const bookName = req.body.bookName;
    const authorName = req.body.authorName;
    const genre = req.body.bookType;

    bookModel.findOne({ book_name: bookName, author_name: authorName, book_type: genre },
{ book_img: 0 }, (err, foundOne) => {
        if (err) console.log(err);
        if (foundOne) {
            bookModel.deleteOne({ book_name: bookName, author_name: authorName,
book_type: genre }, (err) => {
                if (err) console.log(err);
            });
            console.log("deleted successfully");
            res.render("deleting-book", { status: "1", message: "Voila!! Book Deleted
Successfully" });
        }
        else {
            //not found
            res.render("deleting-book", { status: "2", message: "Book Not found ,
Incorrect details!!" });
        }
    });
});

app.post("/book", (req, res) => {
    const nameOfSearchedBook = req.body.searchedBook;
    if (nameOfSearchedBook === "") {
        res.redirect("book");
    }
    else {
        bookModel.find({ book_name: nameOfSearchedBook }, (err, foundOne) => {
            if (err) console.log(err);
            if (foundOne) {

                res.render("book", { books: foundOne, searchedBookFound: "true" });
            }
        });
    }
});

```

```

        console.log("508")
    }
    else {
        res.render("book", { searchedBookFound: "false" });
    }
});

//function that generates a random password when the user forgot the password
function randomPasswordGenerator() {
    var chars =
"0123456789abcdefghijklmnopqrstuvwxyz!@#$%^&*()ABCDEFGHIJKLMNOPQRSTUVWXYZ";
    var passwordLength = 12;
    var password = "";
    for (var i = 0; i <= passwordLength; i++) {
        var randomNumber = Math.floor(Math.random() * chars.length);
        password += chars.substring(randomNumber, randomNumber + 1);
    }
    return password;
}

app.post("/forget-password", (req, res) => {
    const email = req.body.email;
    const accountType = req.body.accountType;
    if (accountType === "user") {
        customerModel.findOne({ customer_email: email }, (err, foundOne) => {
            if (foundOne) {
                const pass = randomPasswordGenerator();
                let transporter = nodemailer.createTransport({
                    host: "smtp.ethereal.email",
                    port: 587,
                    secure: false, // true for 465, false for other ports
                    auth: {
                        user: 'gaetano.boehm88@ethereal.email',
                        pass: 'G2evJnE8wYA5V5AVr7' // generated ethereal password
                    },
                });

                // send mail with defined transport object
                let info = transporter.sendMail({
                    from: 'Shrey Mishra 📩 <ardentebooks@ardent.co.in>', // sender
                    address
                    to: "mishra.shrey001@gmail.com", // list of receivers
                    subject: "Random Generated Password for your account", // Subject
                    line
                    text: "The new password for your account is : " + pass, // plain text
                    body
                });

                bcrypt.hash(pass, saltRounds, function (err, hash) {
                    if (err) console.log(err);
                    customerModel.updateOne({ customer_name: foundOne.customer_name }, {
                        customer_password: hash }, (err) => {
                            if (err) console.log(err);
                        });
                    });
                    res.render("forget-password", { isWrongCredentials: false,
                    isPasswordUpdated: true });
                });
            }
        });
    }
});

```

```
        else {
            res.render("forget-password", { isWrongCredentials: true,
isPasswordUpdated: false });
        }
    })
});
app.listen(3000, (err) => {
    if (err) {
        console.log("running on server");
    }
});
```

5.6. INPUT/OUTPUT SCREENS

5.6.1 Homepage view

The screenshot shows a web browser window with multiple tabs open. The main content area displays the Ardent eBook homepage. At the top, there is a navigation bar with links for About, Books, Reviews, and Contact, along with SignUp and Login buttons. Below the navigation is a section titled "Ardent eBook" featuring three book covers: "THE ART CITY SYMBOL ICON", "YOUR TITLE GOES HERE", and "eMusic Rock". A large, bold text "Grow Your Mind!!" is centered below these images, followed by the tagline "Today a Reader, Tomorrow a Leader". The bottom of the page includes a footer with icons for Secure Payments, Easy Returns, 24x7 customer support, and Fast shipping.

This screenshot shows the same Ardent eBook homepage as above, but with different content visible. It features a section titled "Client's Reviews" containing four testimonial cards for Roman Reign, Sam Morek, Arnold Satur, and Eric Rowan, each with a small profile picture and a "Read More" button. Below this is a section titled "Books" with a single "Books" button. The footer is identical to the previous screenshot, showing a taskbar with various application icons and system status indicators.

Adding Book Data | Cloud: MongoDB Cloud node.js - MongoError: connect Ardent - Google Search Ardent

localhost:3000/index Books

Book 1: A book cover featuring a green and red design with the word "Retro" at the top. It includes placeholder text about Lorem ipsum.

Book 2: A red book cover with a white heart shape formed by hands and the word "Love" below it.

Book 3: A yellow book cover featuring a cartoon illustration of a child reading a book, with the text "BLACK HISTORY MONTH" visible.

Book 4: A blue book cover with a lighthouse illustration and the text "Sample text".

90°F Sunny

Search

3:58 PM 03-Mar-23

Adding Book Data | Cloud: MongoDB Cloud node.js - MongoError: connect Ardent - Google Search Ardent

localhost:3000/index Buy Now Add to Cart Buy Now Add to Cart Buy Now Add to Cart Buy Now Add to Cart

Book 5: A book cover with a geometric pattern and the placeholder text "PLACE YOUR TEXT HERE".

Book 6: A yellow book cover with a bar chart graphic and the placeholder text "SAMPLE TEXT".

Book 7: A book cover with a guitar illustration and the words "Music" and "Rock".

Book 8: A book cover with a Thanksgiving theme featuring the text "BE GRATEFUL AND GIVE THANKS".

Rs.1124

Rs.142

Rs.781

Rs.451

Buy Now Add to Cart

90°F Sunny

Search

3:58 PM 03-Mar-23

Book 9

Book 10

Book 11

Book 12

Rs.474

Rs.139

Rs.452

Rs.124

[Buy Now](#) [Add to Cart](#)

[Get more books](#)

Ardent eBooks

We are most loved eBook website on the internet!!

Helpful Links

- > Privacy Policy
- > Refund Policy
- > Terms and conditions
- > About us

Get in touch

hello@ardentebook.in

90°F Sunny

Created by Shrey Mishra | All rights reserved!

3:58 PM 03-Mar-23

5.6.2 About page view

Ardent eBook

About Books Reviews Contact

[SignUp](#) [Login](#)

About us

About Ardent eBooks

We sell ebooks direct to consumers via Ardent eBooks.com. We provide a range of ebook-specific technical services to book publishers. Launched in 2000, Ardent eBooks.com is a popular ebook retailer with a reputation for innovation, integrity and independence. We sell ebooks direct to millions of consumers around the world, with five local sales portals in the US, Canada, UK, Europe and Australia. You can read our ebooks online, or download them to a vast array of devices, using our Ebook Reader apps. Ardent eBooks.com is privately held and run by its founders, life-long booksellers Stephen and Trudy Cole. It is the only independent ebook retailer that sells into every country in the world, including Antarctica and the International Space Station. We have our own publisher licenses, delivery platforms, reader apps and repository of titles. Ebook Reader, our mobile app, is one of the most popular apps in the ebook market, with millions of installs. Ardent eBooks.com hosts millions of unique ebook titles and has millions of members. In addition to our popular ebook-store, we provide services to book publishers. Digital Comps is a system that enables publishers to send secure ebooks to reviewers and instructors, called "complimentary copies". Ebook Engine is a system that enables publishers to sell ebooks direct to consumers from their own website. And publishers use our Ebook Collections platform to sell collections of their ebooks direct to institutions.

90°F Sunny

4:16 PM 03-Mar-23



About Ardent eBooks

We sell ebooks direct to consumers via Ardent eBooks.com. We provide a range of ebook-specific technical services to book publishers About Us Launched in 2000, Ardent eBooks.com is a popular ebook retailer with a reputation for innovation, integrity and independence. We sell ebooks direct to millions of consumers around the world, with five local sales portals in the US, Canada, UK, Europe and Australia. You can read our ebooks online, or download them to a vast array of devices, using our Ebook Reader apps. Ardent eBooks.com is privately held and run by its founders, life-long booksellers Stephen and Trudy Cole. It is the only independent ebook retailer that sells into every country in the world, including Antarctica and the International Space Station. We have our own publisher licenses, delivery platforms, reader apps and repository of titles. Ebook Reader, our mobile app, is one of the most popular apps in the ebook market, with millions of installs. Ardent eBooks.com hosts millions of unique ebook titles and has millions of members. In addition to our popular ebook-store, we provide services to book publishers. Digital Comps is a system that enables publishers to send secure ebooks to reviewers and instructors, called "complimentary copies". Ebook Engine is a system that enables publishers to sell ebooks direct to consumers from their own website. And publishers use our Ebook Collections platform to sell collections of their ebooks direct to institutions.

Mission

To use the power of emerging technology to make books more accessible and cheaper for everyone, so that one day everyone in the world will have ready access to the world of books.

Vision

To make Ardent eBooks.com the simplest, most beautiful access point for all the world's books.

Proprietary DRM

We have our own DRM system that has secured the content delivered via our Ebook Reader apps since 2007. It is not based on Adobe ACS. Having our own DRM system improves usability for our customers, and enables us to respond quickly to new requirements and trends.

Online (Browser-Based) Reader App

Our proprietary web-based reader enables desktop users to immediately start reading, without having to download or install software. Currently undergoing an upgrade to make it suitable for mobile platforms. Suitable for retail and library use.

Email Newsletter Program

Quis autem velut unius libri per se distinguebat? Quis autem velut unius libri per se distinguebat? Quis autem velut unius libri per se distinguebat?



A Collection of Unique, High Quality Titles

Quis autem velut unius libri per se distinguebat? Quis autem velut unius libri per se distinguebat? Quis autem velut unius libri per se distinguebat?

Publishers' Management Portal

Quis autem velut unius libri per se distinguebat? Quis autem velut unius libri per se distinguebat? Quis autem velut unius libri per se distinguebat?

Ebook Processing Platform

Publishers upload 20,000 ebooks to us every month. Our production unit has been enhanced and modified steadily since March 2000, and has evolved along with the industry. It deals elegantly with all the complexities thrown up by an evolving sector, including faulty files and metadata supplied by publishers. It manages territorial copyright requirements across 195 countries and local sales tax across thousands of jurisdictions.

The Team

Deep, broad, detailed domain expertise. The company comprises a small team of long-time ebook specialists. Our newest developer was engaged in 2010. Some of our guys started in 2001. It's fair to say that our team knows pretty much everything about processing, managing and securely delivering ebooks to a global audience.



5.6.3 Book page view

The screenshot shows a web browser window with the title "Ardent - Google Search". The main content area displays a grid of eight book covers, each with a unique design and title. The books are labeled Book 1 through Book 4 in the first row and Book 5 through Book 8 in the second row. Each book cover includes a small description below it. The top navigation bar has links for About, Books, Reviews, and Contact, along with a search bar. The bottom navigation bar shows various application icons and the system status bar indicating "90°F Sunny" and the date "03-Mar-23".

| Book 1 | Book 2 | Book 3 | Book 4 |
|---|---|---|---|
| Lorem ipsum dolor sit amet consectetur adipiscing elit. Earum nostrum natus laboriosam dolor nulla quod modi quo obcaecati expedita quam delectus et adipisci, impedit quisquam beatae dolore! | Lorem ipsum dolor sit amet consectetur adipiscing elit. Earum nostrum natus laboriosam dolor nulla quod modi quo obcaecati expedita quam delectus et adipisci, impedit quisquam beatae dolore! | Lorem ipsum dolor sit amet consectetur adipiscing elit. Earum nostrum natus laboriosam dolor nulla quod modi quo obcaecati expedita quam delectus et adipisci, impedit quisquam beatae dolore! | Lorem ipsum dolor sit amet consectetur adipiscing elit. Earum nostrum natus laboriosam dolor nulla quod modi quo obcaecati expedita quam delectus et adipisci, impedit quisquam beatae dolore! |
| Rs.454 Buy Now Add to Cart | Rs.414 Buy Now Add to Cart | Rs.144 Buy Now Add to Cart | Rs.545 Buy Now Add to Cart |
| Lorem ipsum dolor sit amet consectetur adipiscing elit. Earum nostrum natus laboriosam dolor nulla quod modi quo obcaecati expedita quam delectus et adipisci, impedit quisquam beatae dolore! | Lorem ipsum dolor sit amet consectetur adipiscing elit. Earum nostrum natus laboriosam dolor nulla quod modi quo obcaecati expedita quam delectus et adipisci, impedit quisquam beatae dolore! | Lorem ipsum dolor sit amet consectetur adipiscing elit. Earum nostrum natus laboriosam dolor nulla quod modi quo obcaecati expedita quam delectus et adipisci, impedit quisquam beatae dolore! | Lorem ipsum dolor sit amet consectetur adipiscing elit. Earum nostrum natus laboriosam dolor nulla quod modi quo obcaecati expedita quam delectus et adipisci, impedit quisquam beatae dolore! |

This screenshot shows the same website layout as the previous one, but the book descriptions now include prices. The books are labeled Book 5 through Book 8. Each book cover has a price tag above it and two buttons below it: "Buy Now" and "Add to Cart". The rest of the interface, including the navigation bar and system status bar, remains consistent.

| Book 5 | Book 6 | Book 7 | Book 8 |
|---|---|---|---|
| Rs.454 Buy Now Add to Cart | Rs.414 Buy Now Add to Cart | Rs.144 Buy Now Add to Cart | Rs.545 Buy Now Add to Cart |
| Lorem ipsum dolor sit amet consectetur adipiscing elit. Earum nostrum natus laboriosam dolor nulla quod modi quo obcaecati expedita quam delectus et adipisci, impedit quisquam beatae dolore! | Lorem ipsum dolor sit amet consectetur adipiscing elit. Earum nostrum natus laboriosam dolor nulla quod modi quo obcaecati expedita quam delectus et adipisci, impedit quisquam beatae dolore! | Lorem ipsum dolor sit amet consectetur adipiscing elit. Earum nostrum natus laboriosam dolor nulla quod modi quo obcaecati expedita quam delectus et adipisci, impedit quisquam beatae dolore! | Lorem ipsum dolor sit amet consectetur adipiscing elit. Earum nostrum natus laboriosam dolor nulla quod modi quo obcaecati expedita quam delectus et adipisci, impedit quisquam beatae dolore! |

Book 9
Lorem ipsum dolor sit amet consectetur adipisicing elit. Earum nostrum natus laboriosam dolor nulla quod modi quo obcaecati expedita quam delectus et adipisci, impedit quisquam beatae dolore!

Rs.474
[Buy Now](#) [Add to Cart](#)

Book 10
Lorem ipsum dolor sit amet consectetur adipisicing elit. Earum nostrum natus laboriosam dolor nulla quod modi quo obcaecati expedita quam delectus et adipisci, impedit quisquam beatae dolore!

Rs.139
[Buy Now](#) [Add to Cart](#)

Book 11
Lorem ipsum dolor sit amet consectetur adipisicing elit. Earum nostrum natus laboriosam dolor nulla quod modi quo obcaecati expedita quam delectus et adipisci, impedit quisquam beatae dolore!

Rs.452
[Buy Now](#) [Add to Cart](#)

Book 12
Lorem ipsum dolor sit amet consectetur adipisicing elit. Earum nostrum natus laboriosam dolor nulla quod modi quo obcaecati expedita quam delectus et adipisci, impedit quisquam beatae dolore!

Rs.124
[Buy Now](#) [Add to Cart](#)



Book 9
Lorem ipsum dolor sit amet consectetur adipisicing elit. Earum nostrum natus laboriosam dolor nulla quod modi quo obcaecati expedita quam delectus et adipisci, impedit quisquam beatae dolore!

Rs.474
[Buy Now](#) [Add to Cart](#)

Book 10
Lorem ipsum dolor sit amet consectetur adipisicing elit. Earum nostrum natus laboriosam dolor nulla quod modi quo obcaecati expedita quam delectus et adipisci, impedit quisquam beatae dolore!

Rs.139
[Buy Now](#) [Add to Cart](#)

Book 11
Lorem ipsum dolor sit amet consectetur adipisicing elit. Earum nostrum natus laboriosam dolor nulla quod modi quo obcaecati expedita quam delectus et adipisci, impedit quisquam beatae dolore!

Rs.452
[Buy Now](#) [Add to Cart](#)

Book 12
Lorem ipsum dolor sit amet consectetur adipisicing elit. Earum nostrum natus laboriosam dolor nulla quod modi quo obcaecati expedita quam delectus et adipisci, impedit quisquam beatae dolore!

Rs.124
[Buy Now](#) [Add to Cart](#)

Ardent eBooks
We are most loved eBook website on the internet!!

Helpful Links
[Privacy Policy](#)
[Refund Policy](#)
[Terms and conditions](#)
[About us](#)

Get in touch
hello@ardentebook.in

[Facebook](#) [LinkedIn](#) [Twitter](#) [Instagram](#)

Created by Shrey Mishra | All rights reserved!



5.6.4 Review page view

The screenshot shows a desktop browser window displaying the Ardent eBook website. The page features a header with the logo 'Ardent eBook' and navigation links for About, Books, Reviews, and Contact. On the right side of the header are 'SignUp' and 'Login' buttons. Below the header is a grid of four review cards, each containing a profile picture, the author's name, a short bio placeholder text, and a 'Read More' button.

| Author | Description | Action |
|--------------|--|-----------|
| Roman Reign | Lorem ipsum dolor sit, amet consectetur adipisicing elit. Eius exercitationem obcaecati molestias? | Read More |
| Sam Morek | Lorem ipsum dolor sit, amet consectetur adipisicing elit. Eius exercitationem obcaecati molestias? | Read More |
| Arnold Satur | Lorem ipsum dolor sit, amet consectetur adipisicing elit. Eius exercitationem obcaecati molestias? | Read More |

At the bottom of the browser window, the taskbar shows various open applications and system status indicators like battery level and date/time.

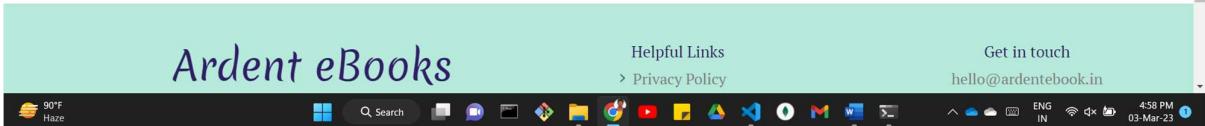
This screenshot is identical to the one above, showing the Ardent eBook website with a grid of four review cards. The layout, content, and system status indicators at the bottom of the browser window are the same.

The screenshot shows the footer section of the Ardent eBooks website. It includes the website name 'Ardent eBooks', a tagline 'We are most loved eBook website on the internet!!', a 'Helpful Links' section with links to Privacy Policy, Refund Policy, Terms and conditions, and About us, and a 'Get in touch' section with an email address 'hello@ardentebook.in'. Below this is a row of social media icons for Facebook, LinkedIn, Twitter, and Instagram. At the bottom, it says 'Created by Shrey Mishra | All rights reserved!' and shows the Windows taskbar with various application icons and system status.

5.6.5 Contact form page view

The screenshot shows a Windows desktop environment with a browser window open to the 'Contact' page at localhost:3000/contact. The browser tabs include 'Adding Book', 'Data | Cloud: MongoDB Cloud', 'node.js - MongoError: connect E', 'Ardent - Google Search', and 'Contact'. The page header features the 'Ardent eBook' logo and navigation links for 'About', 'Books', 'Reviews', and 'Contact'. A 'SignUp' button is in a black box, and a 'Login' button is in a white box. The main content area contains a form titled 'Form' with fields for 'NAME', 'EMAIL', 'CONTACT', and 'MESSAGE'. The 'NAME' field has placeholder text 'Enter your name'. The 'EMAIL' field has placeholder text 'Enter your email address'. The 'CONTACT' field has placeholder text 'Enter your mobile number'. The 'MESSAGE' field has placeholder text 'I want to contact you because...'. Below the form is a Windows taskbar with various pinned icons and system status indicators.

This screenshot shows the same contact form page as above, but with a larger 'MESSAGE' input field that is scrollable. The rest of the form and the browser interface remain identical to the first screenshot.



5.6.6 Privacy-Policy page view

The screenshot shows a Windows desktop environment with a taskbar at the bottom. The taskbar icons include a weather widget (90°F Haze), a search bar, and various application icons like File Explorer, Task View, and Microsoft Edge. The system tray shows the date (03-Mar-23) and time (4:59 PM).
The main window is a web browser displaying the 'PRIVACY POLICY' page for 'Ardent eBook'. The page has a dark header with the site logo and navigation links for About, Books, Reviews, Contact, SignUp, and Login. The main content area contains a detailed privacy policy statement. The URL in the address bar is 'localhost:3000/privacy-policy'.

This Privacy Policy discloses the privacy practices for Ardent eBooks ("We/we" or "Us/us") with regard to Your ("You/you" or "Your/your") use of the online platform www.ardentebook.in ("Website"). This Privacy Policy and Terms of Use describes our current practices regarding, including your choices in relation to how we collect, compile, store, use, share and secure your personal information across our Website. It also describes your choices regarding access, use and correction of personal information and your rights in relation to your personal information and how you can contact us or authorities in the event you have a grievance or complaint. By visiting www.ardentebook.in, you agree to be bound by the terms and conditions of this Privacy Policy. Kindly go through this Privacy Policy carefully. By using the Website, you indicate that you understand and abide by this Policy. If you are not in agreement with the terms and conditions of this Privacy Policy, kindly do not access this Website. We ensure to review this Privacy Policy from time to time so that it is updated. If you are just a visitor, then kindly note that this Policy is subject to change at any time without notice. To ensure that you are aware of any changes in the Policy, please review this Policy from time to time. By Personal Data, we mean any information that can either itself identify you as an individual.

1. COLLECTION OF INFORMATION

As a visitor, you can browse through our Website to find out more about Ardent eBooks. You are not required to provide us with any Personal Data if you are merely a visitor. When you visit the Site, we collect and store certain information in order to increase security, analyse developments and administer the Website with a view to help us in making customer experience better. We utilise this information to analyse traffic patterns on our Website so as to make it more beneficial for our visitors. We collect your Personal Data only when you register with us, when you are interested in obtaining information regarding our products and services, when you participate in activities on our Website or contact us. We will only collect your Personal Information if there is a proper and valid reason for doing so, for instance : to comply with our regulatory and legal obligations; for the performance of our contract with you or to take the required steps at your request before entering into a legal contract; for our legitimate interests

The screenshot shows a Windows desktop environment with a taskbar at the bottom. The taskbar icons include a weather widget (90°F Haze), a search bar, and various application icons like File Explorer, Task View, and Microsoft Edge. The system tray shows the date (03-Mar-23) and time (4:59 PM).
The main window is a web browser displaying the 'PRIVACY POLICY' page for 'Ardent eBook'. The page has a dark header with the site logo and navigation links for About, Books, Reviews, Contact, SignUp, and Login. The main content area contains a detailed privacy policy statement. The URL in the address bar is 'localhost:3000/privacy-policy'.

2. USAGE AND RETENTION OF INFORMATION

We use the personal information we collect, including your Personal Data where it is necessary to render the services requested by you, where it is necessary to comply with legal obligations or rights or for normal business purposes of the kind set out in this Policy. We will use your personal information to provide, analyse and improve our offerings, to provide you with a customized experience on our Website, to contact you about your account and our services, to render customer service, to provide you with personalized marketing and to any fraudulent or illegal activities. In addition to this, we use your personal information to figure out your general geographic location, provide localized course content and classes, recommendations, determine your ISP, and help us quickly and efficiently respond to queries and enforcing our terms and communicate with you concerning our service (for example by email, push notifications, text messaging, and online messaging channels), so that we can provide you details about latest features and content available on the Website, offers, and promotions, surveys, and to assist you with operational requests like password reset requests.

3. COMMUNITY

Ardent eBooks is a community, wherein we offer a number of features that allow members to connect and communicate in public or semi-public spaces, such as Forums and Teams. Please use your judgment before posting in these community spaces or sharing your personal information with others on Ardent eBooks. You should be aware that any personal information you submit there can be read, collected, or used by others, or could be used to send you unsolicited messages.

4. SHARING AND DISCLOSING PERSONAL INFORMATION

We use other Service Providers to perform services on our behalf or to assist us with the provision of services to you. We engage these Service Providers to provide promotional, infrastructure and IT services, to personalize and optimize our service, to process payment transactions, to provide customer service, to analyze and enhance data, and to conduct consumer surveys. In the process of providing such services, these Service Providers may have access to your personal or other information. We do not authorize them to use or disclose your personal information except in relation to providing their services.

5. NOTE TO OUR USERS IN EUROPE:

The screenshot shows a web browser window with multiple tabs open. The active tab is titled "localhost:3000/privacy-policy" and contains the following content:

6. SECURITY

We shall make an effort to take all precautions to shield the personal information both online and offline. We will try to protect your information using security measures to reduce the risks of loss and misuse of data. We have standard SSL certification which basically helps us create a secure connection between our server and user to render any information or action. Some of the protective measures we use are firewalls and data encryption, physical access controls to our data centres and information access authorization controls. Only user passwords are encrypted in 64-bit encryption and stored because generally users use the same password on multiple sites, to prevent any kind of theft, piracy or unauthorised access. If you think that your Ardent eBooks account has been compromised, you can contact us by writing an email to support@ardentebook.in.

7. CHANGES TO THIS POLICY

Ardent eBooks reserves the right to change this Privacy Policy as we may deem necessary from time to time or as may be required by law. Any changes will be immediately posted on the Web Site and you are deemed to have accepted the terms of the Policy on your first use of the Web Site following the alterations.

The page includes a header with "Ardent eBooks" and a sub-header "We are most loved eBook website on the internet!!". On the right, there's a "Helpful Links" sidebar with links to Privacy Policy, Refund Policy, Terms and conditions, and About us. Below the sidebar are social media icons for Facebook, LinkedIn, Twitter, and Instagram. A footer at the bottom right says "Get in touch hello@ardentebook.in". The browser status bar at the bottom shows the URL "localhost:3000/terms-conditions" and various system icons.

5.6.7 Refund policy page view

The screenshot shows a web browser window with multiple tabs open. The active tab is titled "localhost:3000/refund" and contains the following content:

REFUND & CANCELLATION POLICY

There is a strict no refund & no cancellation policy. You are entitled to a refund only in the case where you have not been delivered the eBook after payment.

The page includes a header with "Ardent eBook" and a navigation menu with links to About, Books, Reviews, and Contact. On the right, there are "SignUp" and "Login" buttons. The browser status bar at the bottom shows the URL "localhost:3000/refund" and various system icons.

5.6.8 Signup Form Page view

Screenshot of the Signup form page (localhost:3000/signup) showing fields for Name, Email, Password, Confirm Password, Address, and Pincode. A decorative illustration of a person sitting at a desk with colorful geometric shapes is visible.

Sign up

Your Name

Your Email

Password

Confirm Password

Address

Pincode



90°F Haze 5:04 PM 03-Mar-23

Screenshot of the Signup form page (localhost:3000/signup) showing fields for Password, Confirm Password, Address, Pincode, Contact, and a dropdown for Choose account type (User/Customer). Buttons for Register and Sign In are present. The contact field has been removed from this version of the form.

Password

Confirm Password

Address

Pincode

Contact

Choose account type:

User/Customer

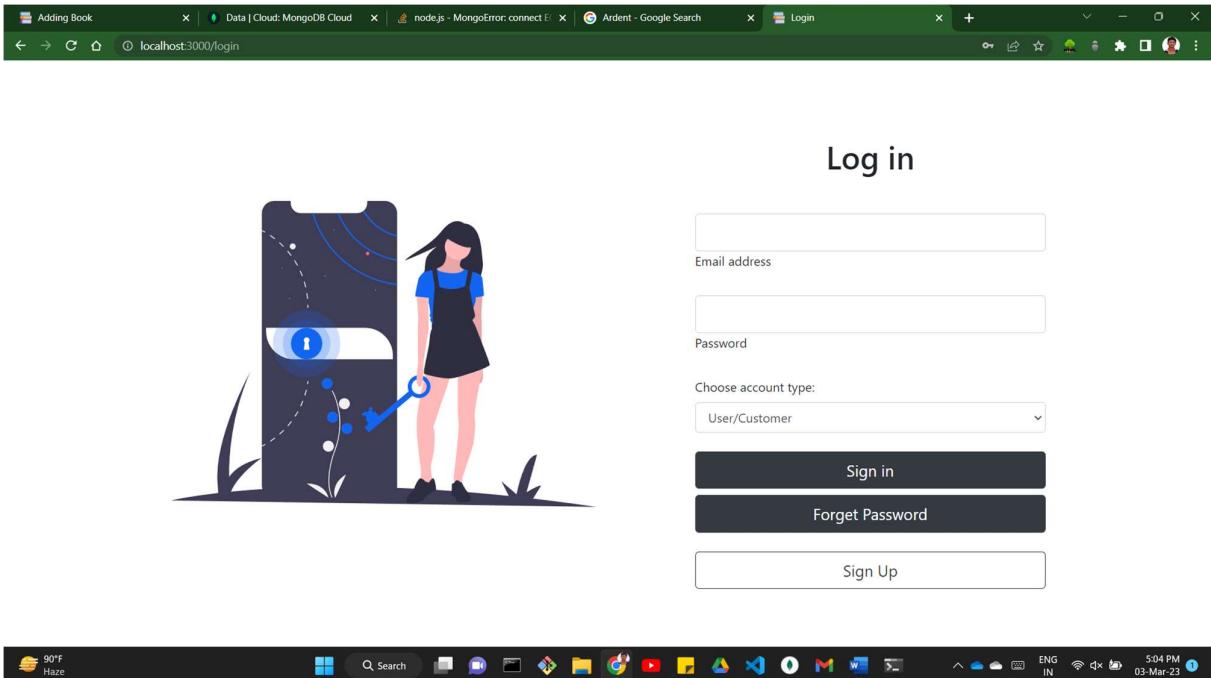
Register

Sign In

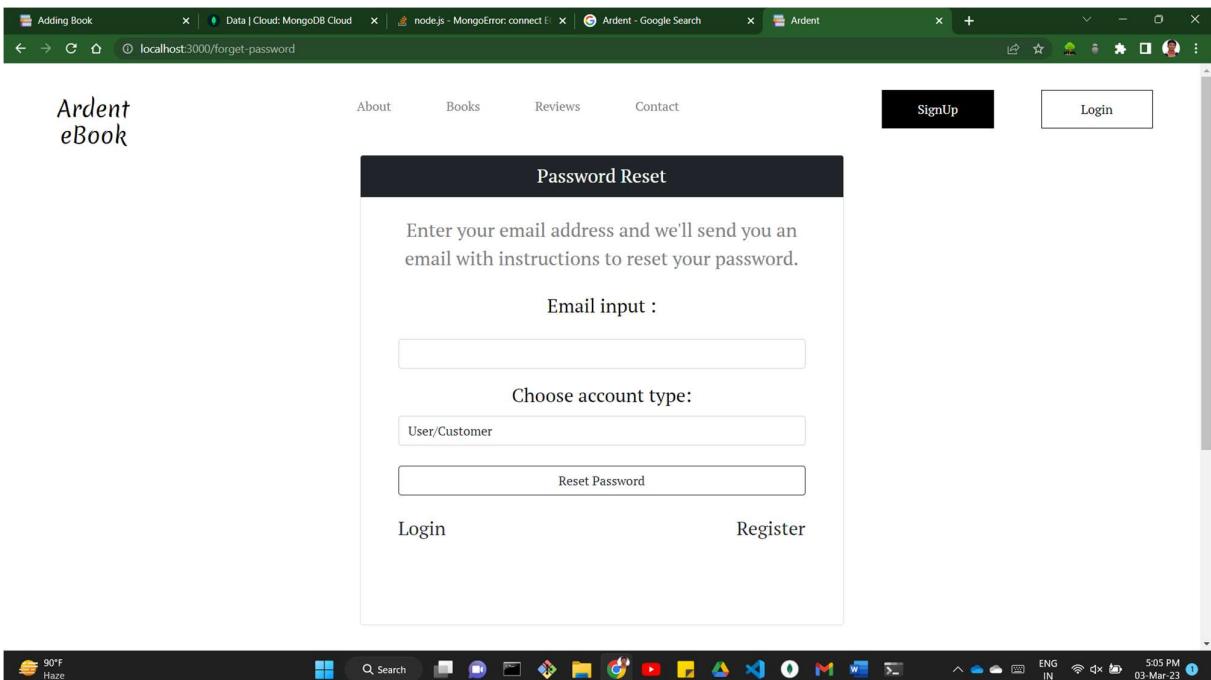


90°F Haze 5:04 PM 03-Mar-23

5.6.9 Login Page View



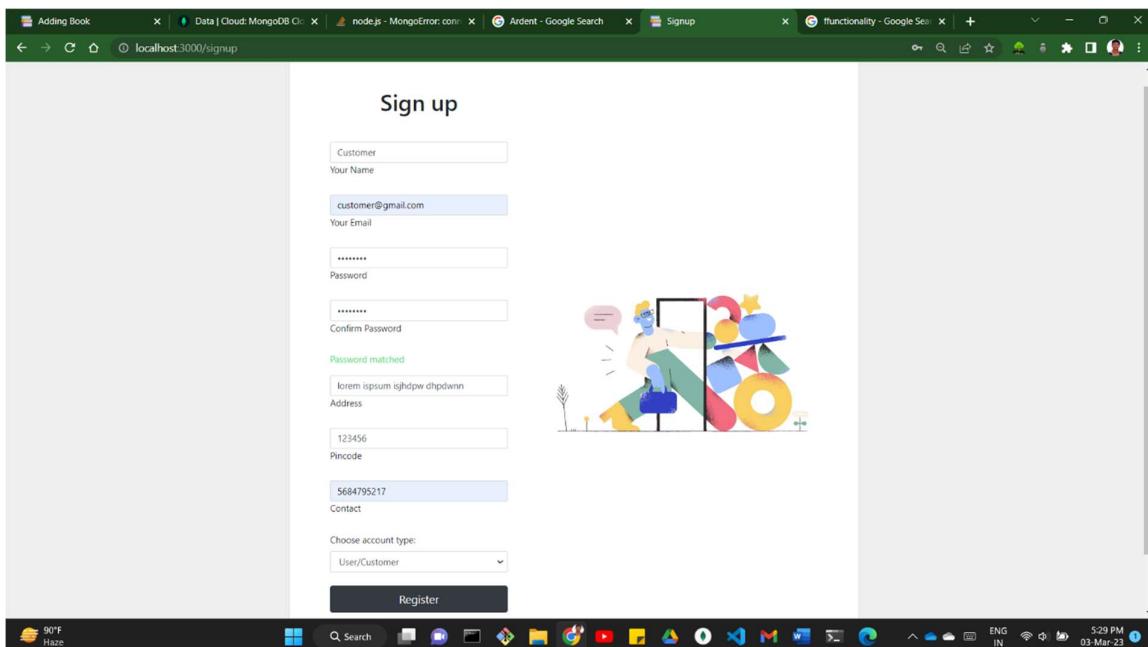
5.6.10 Forget Password Page View



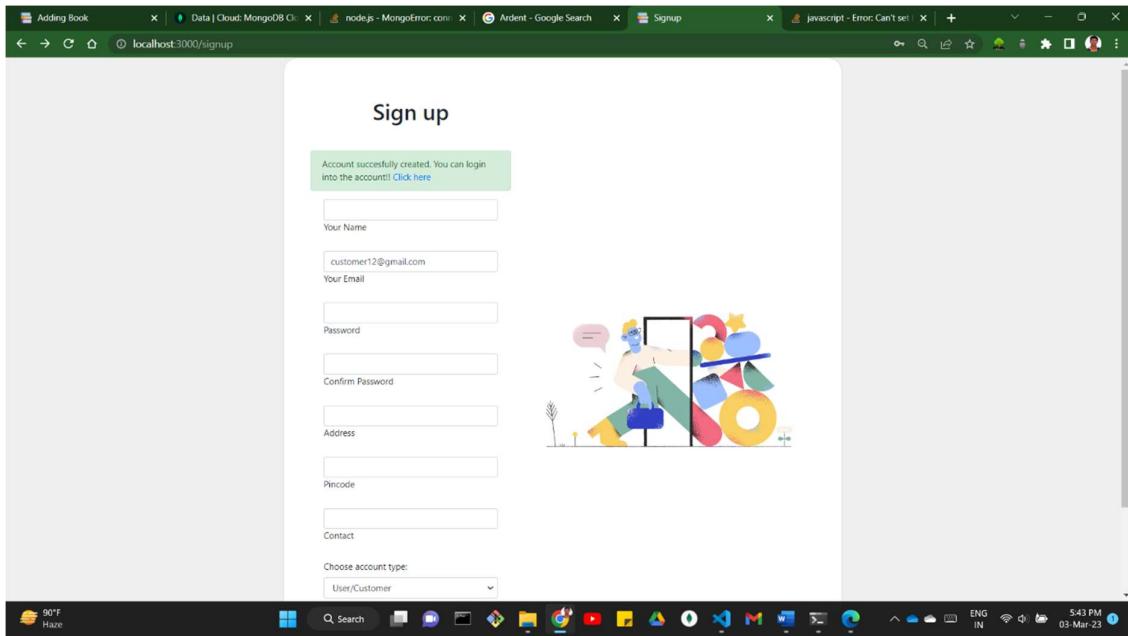
5.7 Functionality of the website

5.7.1 Sign Up

Creating an account by visiting the Signup page and filling up the details.



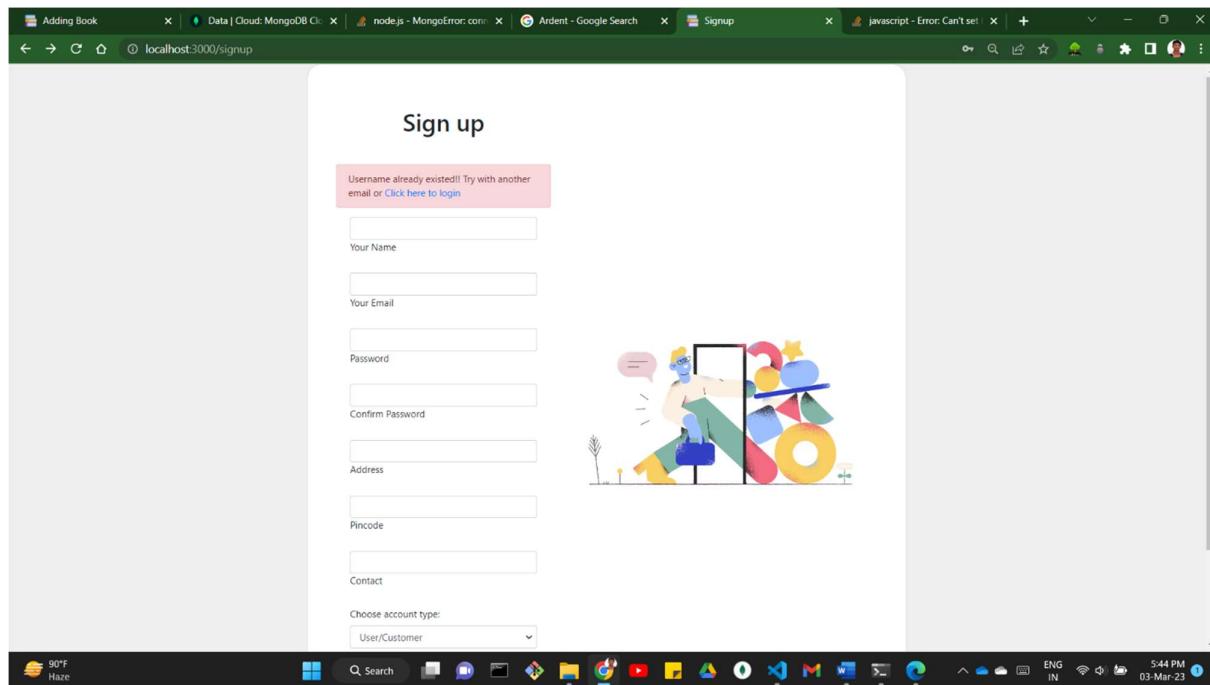
After clicking register, the account will be created and the user will be asked to login by clicking on the link “Click here”.



The data successfully uploaded to the database.

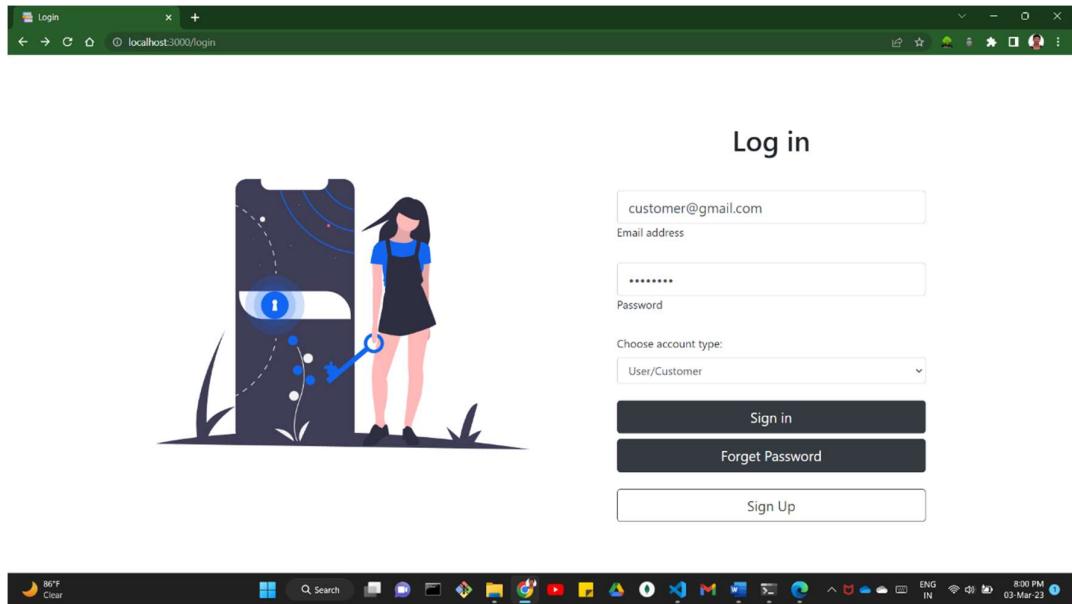
A screenshot of the MongoDB Cloud Atlas interface. On the left sidebar, under "PROJECTS", "Atlas", and "eBookDB", there are sections for "DEPLOYMENT", "SERVICES" (Triggers, Data API, Data Federation), and "SECURITY" (Quickstart, Database Access, Network Access, Advanced). The main area shows the "eBookDB.customers" collection. It displays storage details: STORAGE SIZE: 6KB, LOGICAL DATA SIZE: 2.14KB, TOTAL DOCUMENTS: 9, INDEXES TOTAL SIZE: 36KB. Below this are tabs for "Find", "Indexes", "Schema Anti-Patterns", "Aggregation", and "Search Indexes". A "FILTER" button with the query "{ field: 'value' }" is present. The "QUERY RESULTS" section shows 1-1 OF 1 document: _id: ObjectId("63ff853a2d7c87c9776c9414"), customer_name: "Customer", customer_email: "customer@gmail.com", customer_mobileNo: 8957183830, customer_password: "\$2b\$10\$WnR7ouUCfwYKFcxMw5zxeGR7VqH4jLLHAcCFVBEK1Jb.hIgqE6y", customer_address: "123 new delhi", customer_pincode: 200007. There are buttons for "INSERT DOCUMENT", "OPTIONS", "Apply", and "Reset". The browser's address bar shows "cloud.mongodb.com/v2/63f7eed060de6bbca3edf9#/metrics/replicaSet/63f7fc19210e76c1e87d1b4/explorer/eBookDB/customers/find". The taskbar at the bottom includes icons for "Contact", "Ethereal Email", "Data | Cloud: MongoDB Cloud", and "calculator". The system tray shows the date as 03-Mar-23.

If the user already existed in the DB, the page would ask to either login or try with new email.

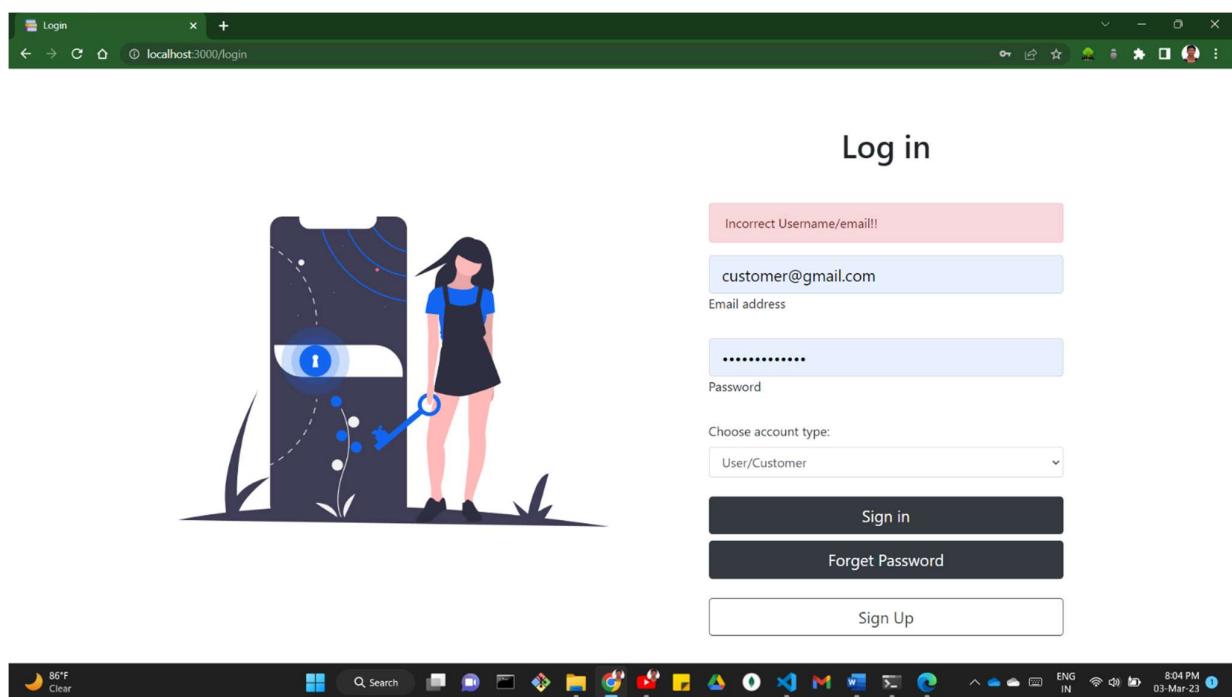


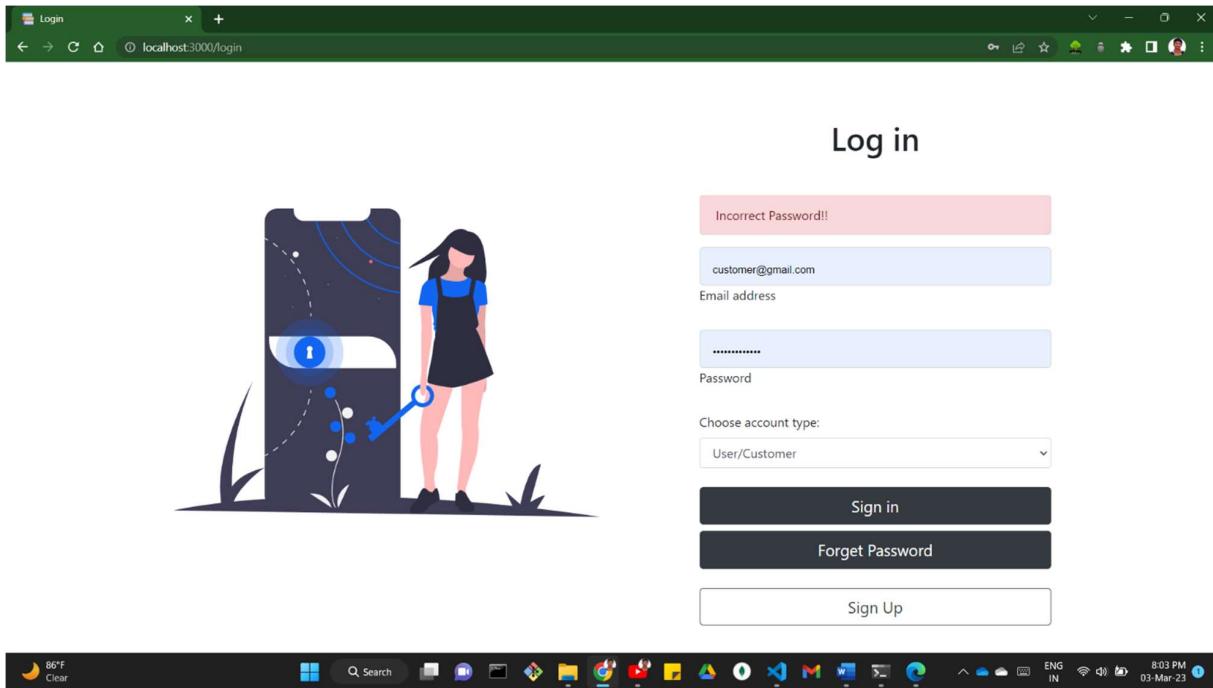
5.7.2 Login

User logging in the website. After successful login, the User will be redirected to the Book page for ordering



If user entered the wrong email or password, he will be asked to fill with correct details.

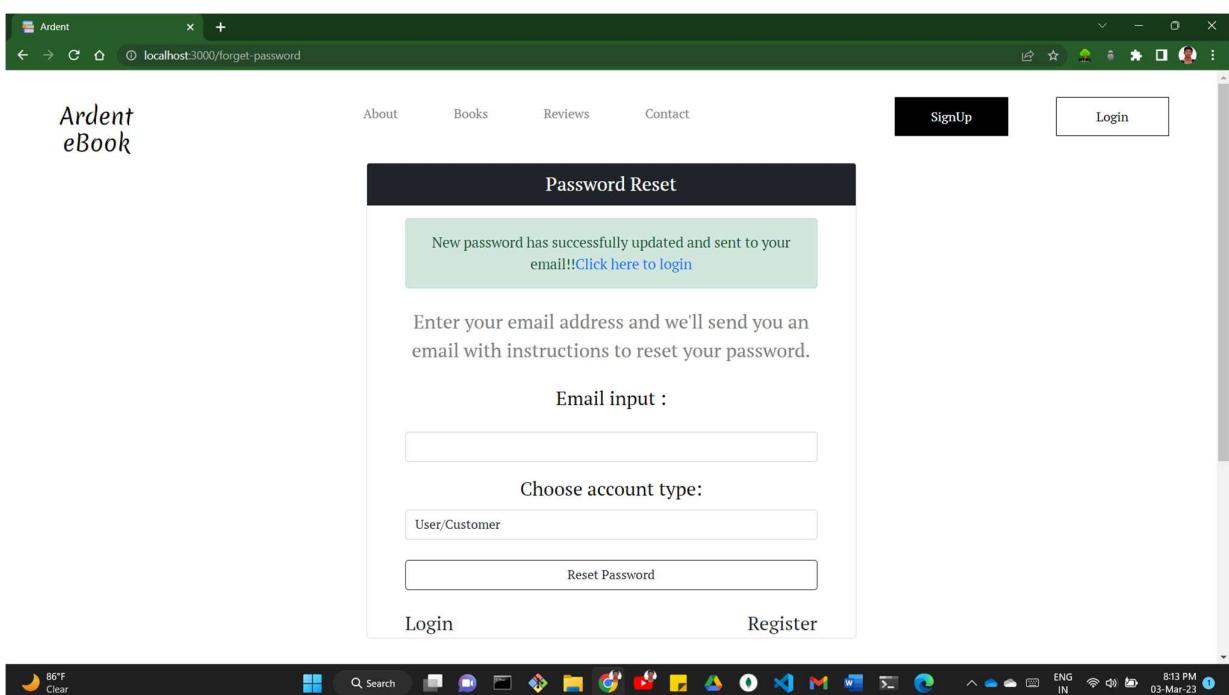




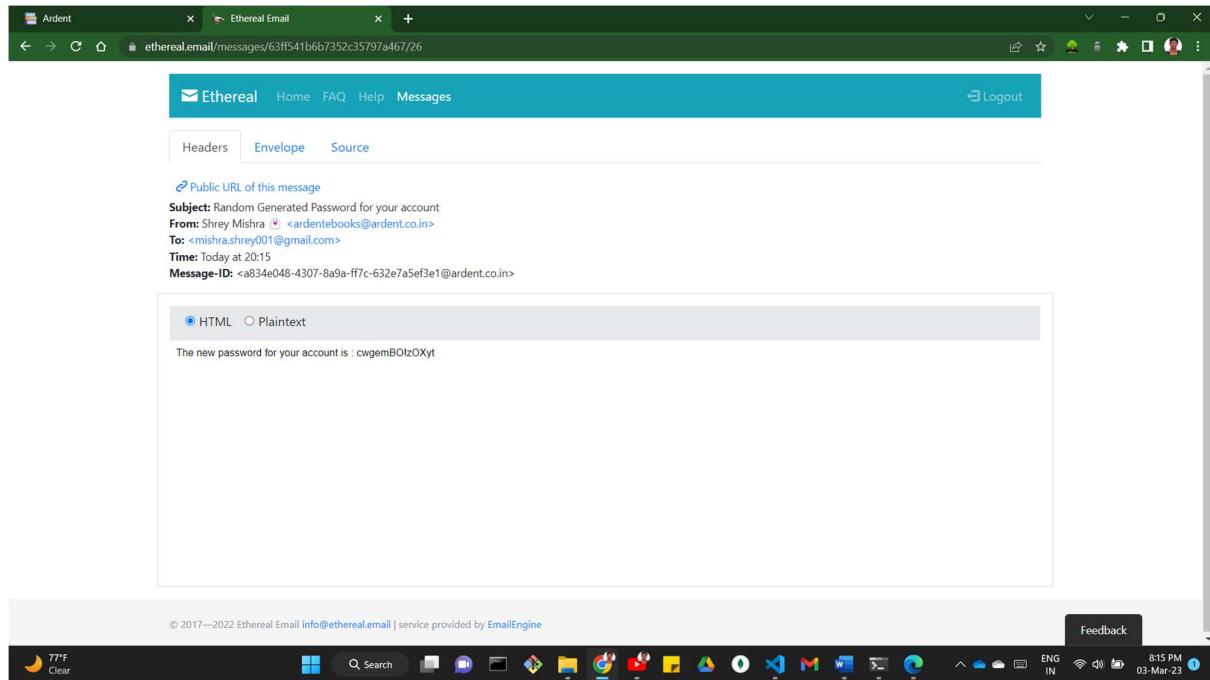
If the user is a seller, then he will be redirected to a different page

with details of adding and deleting book.

If the user/seller forget the password and clicks on the forgotten password, he will be redirected to the forgotten password page, where he will be asked to provide the email and after submission of the page, an automatically generated password is send to the concerned email and the same is updated in the database

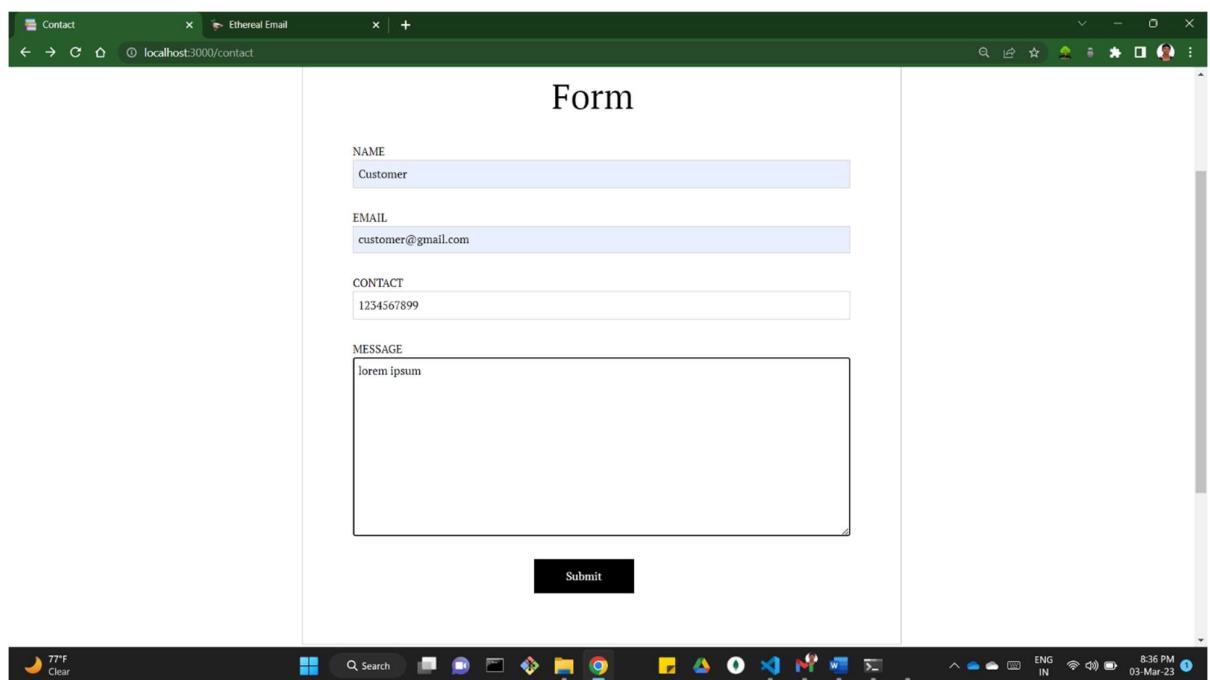


The screenshot of the password sent to the email using the free services of “Ethereal”



5.7.3 Contact

Submitting the form



After the submission of the form

The screenshot shows a web browser window with the URL localhost:3000/contact. The page title is "Contact". The header includes links for "About", "Books", "Reviews", "Contact", "SignUp" (in a dark button), and "Login". The main content area has a title "Form" and a green success message: "Your form is successfully Submitted. We'll revert back soon!!". Below this are four input fields: "NAME" (placeholder: "Enter your name"), "EMAIL" (placeholder: "Enter your email address"), "CONTACT" (placeholder: "Enter your mobile number"), and "MESSAGE" (placeholder: "I want to contact you because..."). The browser's taskbar at the bottom shows various pinned icons and the date/time: 8:36 PM 03-Mar-23.

5.7.4 Feedback

If someone wants to give the feedback , he can feedback on the contact page and submit the feedback

The screenshot shows a web browser window with the URL localhost:3000/review. The page title is "Feedback". The form has four input fields: "NAME" (value: "customer"), "EMAIL" (value: "customer@gmail.com"), "CONTACT" (value: "5684795217"), and "FEEDBACK" (value: "lorem ipsum"). A "Submit" button is located at the bottom of the form. The browser's taskbar at the bottom shows various pinned icons and the date/time: 8:54 PM 03-Mar-23.

After submission of the feedback

The screenshot shows a web browser window titled "Ethereal Email" with the URL "localhost:3000/review". The main content is a "Feedback" form. A green success message at the top says "Your feedback is valuable for us. Thanks for your response!!". The form fields are: NAME (input placeholder "Enter your name"), EMAIL (input placeholder "Enter your email address"), CONTACT (input placeholder "Enter your mobile number"), and FEEDBACK (text area placeholder "Type your feedback here!!"). Below the form is a "Submit" button. The browser's toolbar includes standard icons like back, forward, search, and refresh. The taskbar at the bottom shows various application icons and the system clock "8:54 PM 03-Mar-23".

Your feedback is valuable for us. Thanks for your response!!

NAME
Enter your name

EMAIL
Enter your email address

CONTACT
Enter your mobile number

FEEDBACK
Type your feedback here!!

Submit

5.8 Testing

Glass Box Testing:-

Testing performed by a programmer with access to and understanding of the source code is called Glass Box Testing or structural Testing.

Glass Box testing is usually performed during source code development and prior to Black Box testing which is also known as Functional Testing. The original programmers typically perform some Glass Box testing. Unlike Glass Box testing, Black Box testing is testing of the program from the user's perspective with no access to source code. Glass Box testing may include the following:

Code Review- A careful line-by-line review of the source code by another programmer familiar with the program's overall objectives. In the development teams, exchanging source code with another programmer best performs this.

Complete Coverage- A coverage monitor ensures that all lines of the source code are executed. Different types of coverage monitoring programs exist which look at executing all lines, all branches, or all logic conditions for each branch.

Drivers- Adding code that calls the module being developed with artificial data and getting a known response. This also allows for incremental or evolutionary programming, which allows the entire program to be run even when intermediate modules have not yet been developed.

Module deletion- To narrow down how crashes occur, modules are removed and stubs (non-performing modules with the same name) are substituted.

Assertion checks- Conditional statements of a premise at a certain point in the program. The result of an assertion that is incorrect should be a special error condition. Assertion checks may be conditionally compiled, so that they do not exist in the final program.

In addition, the programmer should identify the following during Glass Box Testing, if not already defined in the specification:

Input ranges – The maximum and minimum allowable values for every data entry field.

Internal boundaries – Threshold values of inputs that cause different program segments to be used.

Error handler triggers – Values of input that should cause error handling routines to be executed.

Black Box Testing:-

Black Box testing that is also known as Functional or Behavioral testing. This is a testing pre-release or release version of the program and trying various inputs looking for incorrect outputs or program crashes. It is one of the best ways to diagnose bugs before users discover them. Given the complexity of many software packages, much thought must be devoted to optimizing the testing process. Ideally only one test will be performed for each possible set of software conditions. Of all possible sets of software conditions there are groupings of conditions that if all tested would be testing the same code and reveal the same bugs (or hopefully the lack of bugs). These groups of conditions are considered an equivalent class or each is called an equivalent class partition.

No program or system design is perfect, communication between the user and the designer is not always complete or clear, and time is

usually short. So, the result is error and more errors. The number and nature of errors in a new design depend on several factors:-

- Communication between the user and the designer.
- The programmer's ability to generate a code that reflects exactly the system specification.
- The time frame for design.

Theoretically, a newly designed system should have all the pieces in working order, but in reality, each piece works independently. Now is the time to put all the pieces into one system and test it to determine whether it meets the user's requirements. This is the last chance to detect and correct errors, before the system is installed for user acceptance testing. The purpose of system testing is to consider all the likely variations to which it will be subjected and then push the system to its limit. It is a tedious but necessary step in system development.

Testing is vital to the success of the system. System testing makes a logical assumption that if all parts of the system are correct, the goal

will be successfully achieved. Inadequate testing or contesting leads to errors that may not appear until months later.

Another reason for system testing is its utility as a user-oriented vehicle before implementation. The best program is worthless if it does not meet user needs. Unfortunately, the user's demands are often compromised by efforts to facilitate program or design efficiency in terms of processing time or memory utilization. Often the computer technician and the user have communication barriers due to different backgrounds, interests, priorities, and perhaps languages. The system tester (designer, programmer, or user) who has developed some computer mastery can bridge this barrier.

Before going to test, it is very important to know some basic terms:-

- Unit Testing- It is a testing changes made in an existing or a new program.

- Sequential or Series Testing- It is checking the logic of one or more programs in the candidate system, where the output of one program will affect the processing done by another program.

- System Testing- It is executing a program to check logic changes made in it and with the intention of finding errors-making the program fail. Effective testing does not guarantee reliability. Reliability is a design consideration.

- Acceptance Testing- It is running the system with live data by the actual user.

The first test of a system is to see whether it produces the correct outputs. No other test can be more crucial. Practically it is to be kept in mind that Testing has two broad aspects or sites:-

5.9 IMPLEMENTATION OF SECURITY

Security holds an important role in a software. Most people are not aware of the significance of security in software system and tend to assume that they will be fine without security in their software systems. However, the lack of security features causes to expose all the vulnerabilities possible to the public. This provides opportunities for the attackers to perform dangerous activities to the vulnerable insecure systems. This is the reason why many organizations are reported for being victims of system security attacks. In order to achieve the security requirement, developers must take time to study so that they truly understand the consequences and importance of security.

MongoDB has various built in features for the security of the confidential data such as :

Network Isolation

MongoDB Atlas dedicated clusters are deployed in a unique Virtual Private Cloud (VPC) with dedicated firewalls. Access must be granted by an IP access list or VPC Peering.

Role-based access management

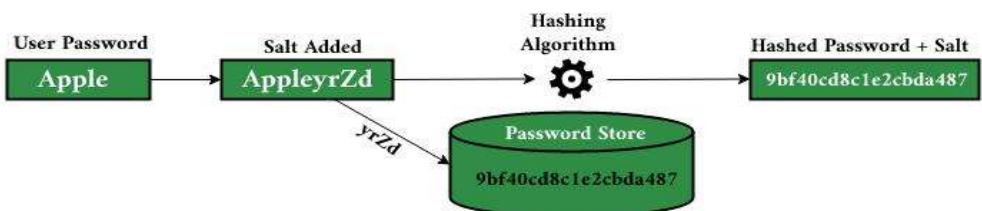
Configure sophisticated role-based access rules to control which users and teams can access, manipulate, and delete data in your databases.

End-to-end encryption

All network traffic is encrypted using Transport layer Security (TLS), with flexibility to configure the minimum TLS protocol version. Encryption for data at rest is automated using encrypted storage volumes.

Although MongoDB provides all the above security measure, but keeping in mind the breaching and hacking of data, in our project we have used the hashing technique to store the password so that the password can't be known to the hacker, even if he hacked our database. For this purpose we have used the basic hashing algorithm called “bcrypt” with salting. This hashing algorithm converts the actual data into the alphanumeric string which can't be easily understand by the hacker or to any modern computer system even with the advance computational capabilities. Salted password hashing

can be used to improve password security by adding additional layers of randomness on top of the hashing process. Salt is a cryptographically secure random string that is added to a password before it's hashed, and the salt should be stored with the hash, making it difficult for an attacker to know the original plaintext without having access to both sources. This process is often used in combination with bcrypt, another function that adds computing requirements for each and every attempt by an attacker who doesn't have access to either source. Salted hashing is a much more complex and secure process because each hash requires the use of a different and random 'salt', that acts as an additional layer of encryption. This means that every known salt (or collection) is required to perform the password hashing function.



5.10 FUTURE SCOPE AND FURTHER ENHANCEMENT

The future scope of eBook stores looks promising as technology continues to evolve and reading habits shift towards digital formats. Here are some potential areas of growth:

Expansion of the ebook market: With the increasing popularity of ebooks, there will likely be a larger customer base for ebook stores. This will result in more sales, more revenue, and more competition.

Integration with other technologies: Ebook stores will likely continue to integrate with other technologies, such as smartphones, tablets, and e-readers. This will make it easier for customers to access and purchase ebooks.

Personalization and customization: Ebook stores will likely offer more personalized and customized experiences for customers, such as recommendations based on reading habits and preferences.

More diverse content: Ebook stores will likely expand their offerings to include more diverse content, such as graphic novels, cookbooks, and academic textbooks.

Increased emphasis on self-publishing: As more authors choose to self-publish, ebook stores will likely offer more opportunities for self-publishing and marketing services.

Overall, the future of ebook stores looks bright, with continued growth and innovation in the digital reading industry.

Further Enhancements: -

There are several ways to improve our ebook website to enhance the customer experience and increase sales. Here are some suggestions:

User-friendly interface: Though we have ensure that the website is easy to navigate, with clear categories, search filters, and a simple checkout process but still with the advancement of technology and

keeping the feedback of the user in mind we can always add more to it for better user experience

Mobile responsiveness: Recent studies have clearly shown that the more user are accessing internet through the smartphones and tablet. The major drawback of this website is that it is not mobile responsive, with time and technology we can convert the website to be responsive.

Personalization: Using data analytics from surveys and researches, we can provide personalized recommendations based on a customer's reading history, preferences, and purchase behavior.

Quality content: We can also integrate a new module which ensures that the ebooks offered on the website are of high quality and relevant to the target audience.

Customer reviews and ratings: With the customer feedback, reviews and ratings, we can further improve our product.

By implementing these improvements, our eBook websites can provide a better customer experience and increase sales, leading to a more successful and profitable business.

5.11 BIBLIOGRAPHY

- INTRODUCTION TO SOFTWARE ENGINEERING (IGNOU)
- PROJECT GUIDE BOOKLET (IGNOU)
- LEARNING WEB DESIGN BY JENNIFER ROBBINGS
- WEB DEVELOPMENT BOOTCAMP BY ANGELA YU
- FULL STACK-REACT PROJECTS BY SHAMA HOQUE
- OFFICIAL DOCUMENTATION OF MONGODB AND
NODEJS
- OFFICIAL DOCUMENTATION OF EXPRESSJS