➢ 1)INTRODUCTION

This website namely "BOOK HOUSE" motivates the excitement of a bibliophile which serves different kinds of genres of books providing brief details of the author and description of the books which generates a passion to read the book .All the book are equipped for sale throught the website with all king of payment methods

1.1 OVERALL DESCRIPTION:

This Bookhouse is the perfect destination for bookworms with different kind of books allowing the readers to purchase with no need of visiting stores or waiting in queues instead the user can visit the site to check for the book they want for reading

The site also describes the book which enlights the key elements presented in the book and the characters of the book or novel triggering the curiosity to know what's next in the book

We also provide a brief introduction of the author about his or her life and famous bestsellers of their works are also displayed here

➢ 2)SYSTEM ANALYSIS

System Analysis is the description of a system into its component pieces to study how thecomponent pieces are study and work.

2.1 Software Requirement Specifications:

Software Requirement Specification is the starting point of software developing activity.

As system grew more complex it became evident that the goal of the entire system cannot be a silly comprehended. Hence the needs for the requirements phase are use. The software project is initiated by the client needs, the SRS is the means of translating the ideas of the minds of clients (he input) into a formal document. The purpose of the software requirement specification is to reduce the communication gap between the clients and developers. Software Requirement Specification is the medium through which the client and user needs are accurately specified. It forms the basis of software development. A good SRS should satisfy all the parties of involved in the system.

2.1.1 Purpose:

The purpose of Bookhouse is to help the book lovers and book sellers by increasing the availability of books.

2.1.2 Scope:

The scope of this system is to make books available to the users through Bookhouse This system can be accessed through mobile also. As our System supports both desktop mode and mobile mode, it is Compatible. Internet Connection is necessary to access this System.

2.1.3 Objective:

The objective of website is to provide books for the readers in a easy and efficient way with reviews and description of the book introducing the authors

2.1.4 Existing System:

This is a general way of buying a book in a store where we can examine the book properly we are supposed to buy it in a offline system

2.1.5Proposed System:

The proposed system provides a user interface to view the book categorically and buy using different payment modes it also provides admin interface to look after he functionalities of the website

2.1.6 Functional Requirements:

- Easy login process and have a Ability to login with crendentials
- The user has to Register If he/she have no account
- On successful registration the user can access the website
- The user can view the website and purchase the books arranged in various categories
- The user can also get the entire description of a book which he wants to purchase
- The user can drop his valuable Feedback about the website in the feedback box
- He can also view the details of the Creators to contact them
- Admins can avail the functionalities of the website through admin login
- The admin can administer the website
- He can add and delete the admin whenever necessary
- He also has access to view the feedbacks sent by the user
- He can regulate and monitor the orders received and confirm them after successful deliveries

2.1.7 Non Functional Requirements:

- User Interface should be compatible to load HTML, CSS, bootstrap pages in Front End.
- Software Interface should support the OS and Database of the User.
- Communication Interface can support all Web Browsers except Internet Explorer.
- Availability: The application is available to all the intended users, all the time based on the Network Availability.
- Maintainability: Issues that have been solved can be deleted from the database so as to maintain less complexity.
- Implementation: This System can be easily implemented and has scope for making future changes easily, since the system is

developed by using the feature of Modularity.

- Security: Security is provided by JSON web tokens, data is encrypted by SHA256.

2.1.8 Software Requirements:

- HTML.
- CSS.
- MY SQL.
- php
- XAMPP
- Bootstrap
- Any browser except internet explorer.

2.1.9 Hardware Requirements:

- Desktop Computers and Personal Mobile Devices
- Keyboard.
- Mouse.
- Minimum 4GB RAM.
- Pentium Processor and above.
- Minimum 256GB Hard Disk.

SYSTEM DESIGN

System design is the process of defining elements of a system like modules, architecture, components and their interfaces and data for a system based on the specified requirements. It is the process of defining, developing and designing systems which satisfies the specific needs and requirements of a business or organization.

Object Oriented Design is concerned with developing an object oriented model of a software system to implement the identified requirements. It is the process of defining the components, interfaces, objects, classes, Attributes and operations that will satisfy the requirements.

The designs can be defined in graphical or textual modelling languages.The designer's goal is how the outputs to be produced and in what format samples of output are also presented. The processing phases are handled through the program construction and testing.

The word "QUALITY" sums up the significance of software design. Design gives us access to visual examples of software that can be evaluated for quality. The only method to successfully translate a customer's needs into a final software product or system without taking any design risks is through design.

Object-oriented design (OOD) is the process of planning a system of interacting objects for the purpose of solving a software problem. From enabling the implementation of a software based on the concepts of objects and deleting the shared data areas to distributing and executing the object sequentially or in parallel, the benefits of this approach of software design are numerous. Object oriented design can yield the following benefits:

EASY TO USE:

The primary benefit of object oriented analysis and design is that it is understandable. The approach of this model builds on common paradigms that most people use to deal with complexity. Therefore, software developers and programmers are able to decompose complex problems into objects.

MAINTAINABILITY:

OOP systems are convenient and easy to operate compared to structured programs. It provides less complexity in system design, easier verification by the user. It therefore becomes easier to identify errors early in the software development process. The OOP design has become popular with software developers due to its maintainability.

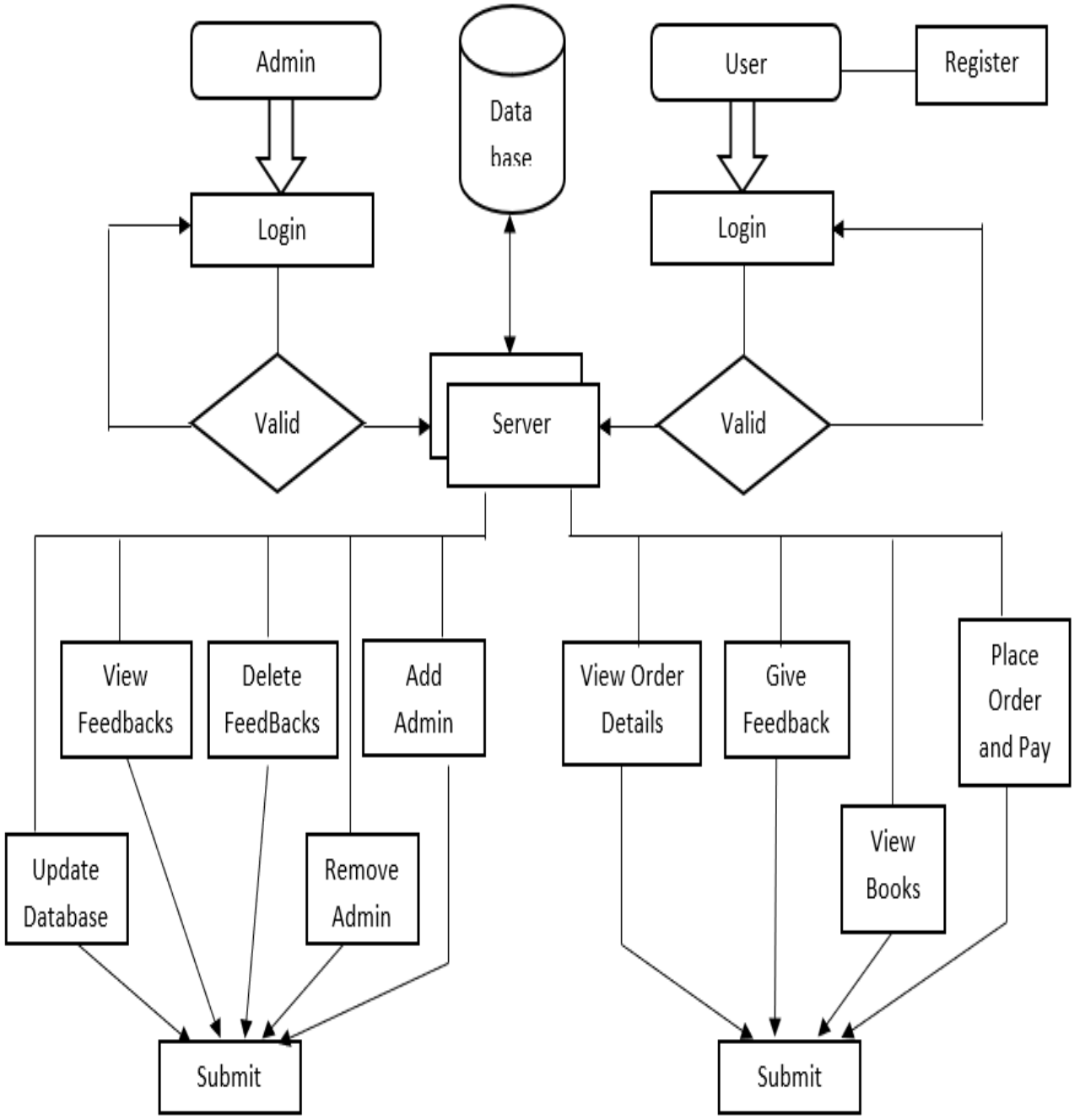
REUSABILITY:

Encapsulation binds code and enables the implementation of classes without constant changes to its constituent services (for example modifying the code without altering the old one). Additionally, OOP programs are scalable as compared to their structured application software counterparts. This is because OOP allows reusing of code in new applications and reusability of the design artifacts, which saves time and cost

PRODUCTIVITY:

Benefits from the direct mapping of Object Oriented Programming Language features.

PROBLEM ARCHITECTURE:



DATABASE DESIGN

Database :

A database is an organized collection of data, generally stored and accessed electronically from a computer system. Where databases are more complex they are often developed using formal design and modeling techniques.

The database management system(DBMS) is the software that interacts with end users, applications, and the database itself to capture and analyze the data. The DBMS software additionally encompasses the core facilities provided to administer the database. The sum total of the database, the DBMS and the associated applications can be referred to as a "database system". Often the term "database" is also used to loosely refer to any of the DBMS, the database system or an application associated with the database.

ER Diagram :

An Entity Relationship Diagram (ERD) is a visual representation of different entities within a system and how they relate to each other.ER-modeling is a data modeling technique used in software engineering to produce a conceptual data model of an information system. Diagrams created using this ER-modeling technique are called Entity-Relationship Diagrams, or ER diagrams or ERDs. So you can say that Entity Relationship Diagrams illustrate the logical structure of databases.

ERDs show entities in a database and relationships between tables within that database. It is essential to have ER-Diagrams if you want to create a good database design. The diagrams help focus on how the database actually works.

ER modeling is one of the most cited papers in the computer software field. Currently the ER model serves as the foundation of many system analysis and design methodologies, computer-aided software engineering (CASE) tools, and repository system.

Elements in ER diagram:

Entity relationship diagrams are used in software engineering during the planning stages of the software project. They help to identify different system elements and their relationships with each other. It is often used as the basis for data flow diagrams or DFD's as they are commonly known.

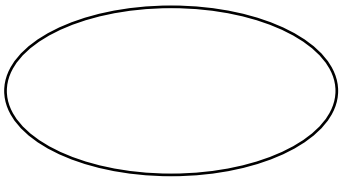
The basic elements in ER-Diagrams :

- **Entity :**

Entities are the "things" for which we want to store information. An entity is a person, place, thing or event. Entity can be represented with Rectangles.

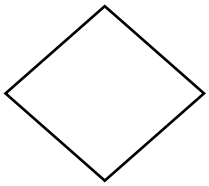
- **Attributes :**

Attributes are the data we want to collect for an entity. An attribute is a property, trait, or characteristic of an entity, relationship, or another attribute. Attributes are represented by Oval shapes.



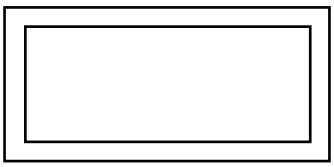
- **Relationships :**

Relationships describe the relations between the entities. ERDs show entities in a database and relationships between tables within that database. It is essential to have ER-Diagrams if you want to create a good database design. The diagrams help focus on how the database actually works.



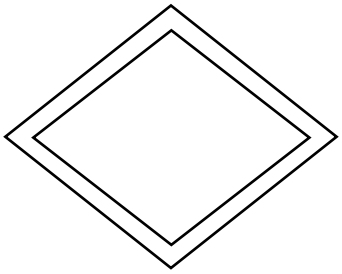
- **Weak Entity :**

A weak entity is an entity that depends on the existence of another entity . In more technical terms it can be defined as an entity that cannot be identified by its own attributes. It uses a foreign key combined with its attributed to form the primary key.



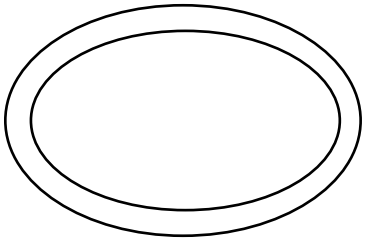
- **Identifying relationship :**

The relationship that relates the weak entity type to an owner entity type is known as identifying relationship. The weak entity type always has total participation(existence dependency) in a relationship because the weak entity type can not be identified without an owner identity.



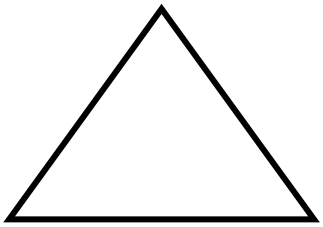
- **Multi-valued Attribute :**

If an attribute can have more than one value it is called a multi-valued attribute. It is important to note that this is different from an attribute having its own attributes.



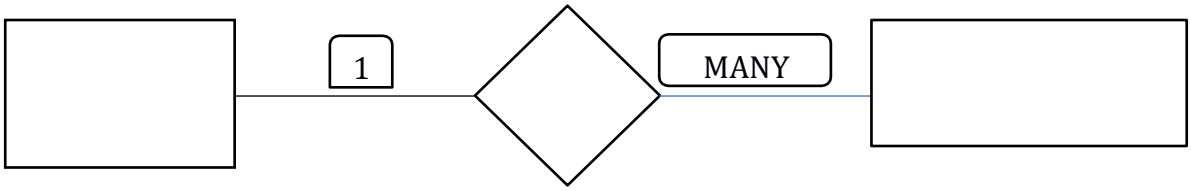
- **ISA Relationship:**

Inheritance is defined as the ability of a lower-level object to inherit, or access, the data items and behaviors associated with all classes which are above it in the class hierarchy. For example, consider a scenario where a many-to-many relationship exists between ORDER and ITEM classes. Because an order may be for many items, and an item participates in many orders, these entities for the classic many-to-many relationship



- **One-to-Many Relationship :**

When only one instance of an entity is associated with therelationship, then it is known as one to one relationship.

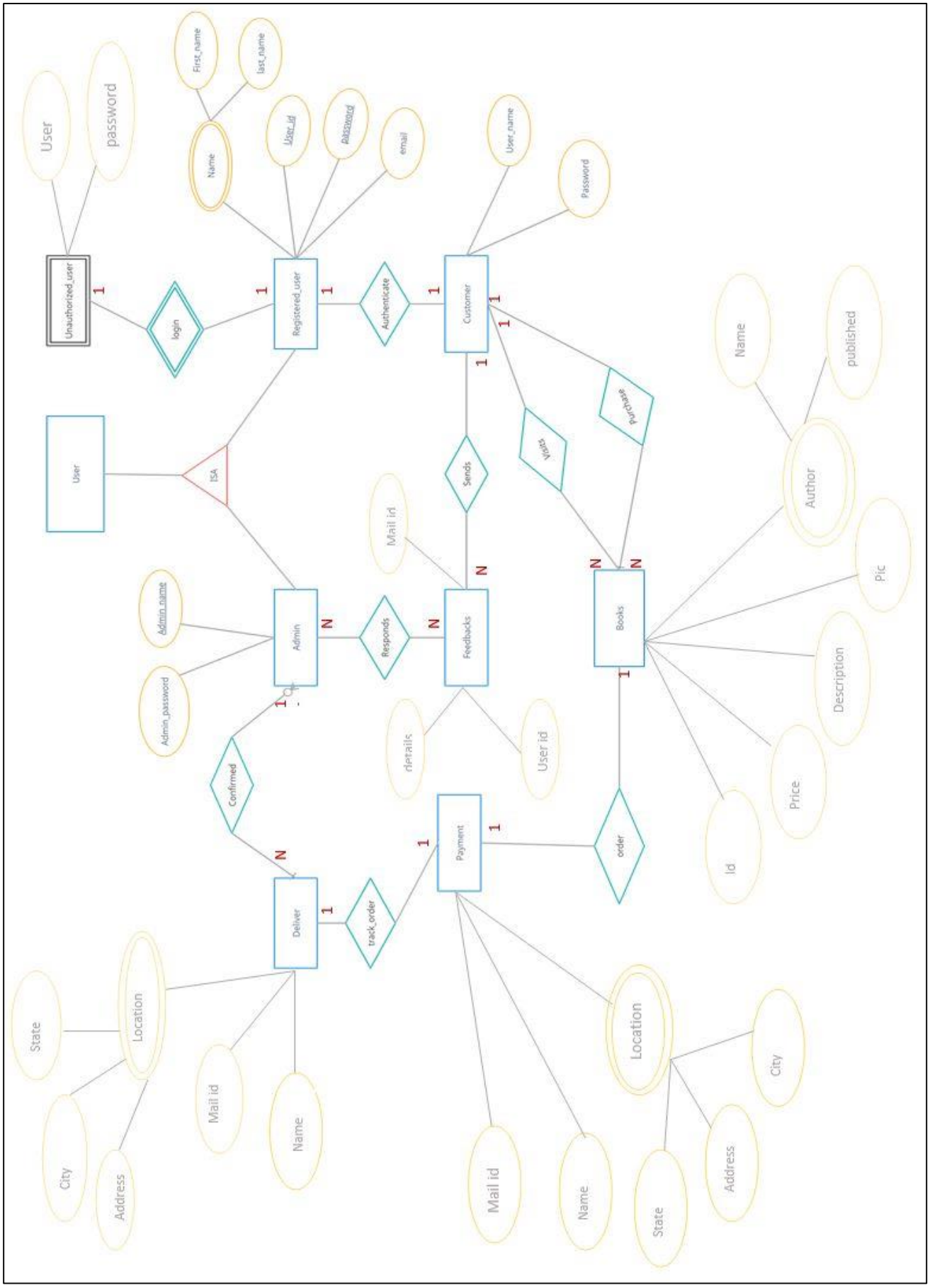


- **How to draw ER Diagrams :**

Below points show how to go about creating an ER diagram.

1. Identify all the entities in the system. An entity should appear only once in a particular diagram. Create rectangles for all entities and name them properly.
2. Identify relationships between entities. Connect them using a line and add a diamond in the middle describing the relationship.
3. Add attributes for entities. Give meaningful attribute names so they can be understood easily

- **ER Diagram for project :**



Database Tables :

List of tables :

Tables_in_book_store
admin_d
book_det
confirm_d
fed_d
login_d
pay_d
reg_details

Admin table :

Field	Type	Null	Key	Default	Extra
admin_name	varchar(225)	NO	PRI	NULL	
password	varchar(225)	NO		NULL	

Books table :

Field	Type	Null	Key	Default	Extra
pic	text	YES		NULL	
name	char(20)	NO		NULL	
author	char(20)	NO		NULL	
price	int	NO		NULL	
des	text	NO		NULL	
id	int	YES		NULL	

Confirm table :

Field	Type	Null	Key	Default	Extra
name	varchar(225)	NO		NULL	
mail_id	varchar(225)	NO		NULL	
address	varchar(225)	NO		NULL	
city	varchar(225)	NO		NULL	
state	varchar(225)	NO		NULL	

Feedback table :

Field	Type	Null	Key	Default	Extra
name	varchar(225)	NO		NULL	
mail	varchar(225)	NO		NULL	
feedback	varchar(1000)	NO		NULL	

Login table :

Field	Type	Null	Key	Default	Extra
user_name	varchar(225)	NO		NULL	
passcode	varchar(225)	NO		NULL	

Registration table :

Field	Type	Null	Key	Default	Extra
f_name	varchar(225)	NO		NULL	
l_name	varchar(225)	NO		NULL	
user_id	varchar(225)	NO	PRI	NULL	
email	varchar(225)	NO		NULL	
password	varchar(225)	NO	PRI	NULL	

Payment table :

Field	Type	Null	Key	Default	Extra
name	varchar(225)	NO		NULL	
mail_id	varchar(225)	NO		NULL	
address	varchar(225)	NO		NULL	
city	varchar(225)	NO		NULL	
state	varchar(225)	NO		NULL	

IMPLEMENTATION

Implementation is the stage where the theoretical design is turned into a working system. The most crucial stage in achieving a new successful system and in giving confidence on the system for the users that will work efficiently and effectively. The system will be implemented only after through testing and if its found to work according to the specification.

Overview of Software Used :

HTML:

Hypertext Markup Language (HTML) is the standard markup language for creating web pages and web applications. With Cascading Style Sheets (CSS) and JavaScript, it forms a triad of cornerstone technologies for the World Wide Web.

Web browsers receive HTML documents from a web server or from local storage and render the documents into multimedia web pages. HTML describes the structure of a web page semantically and originally included cues for the appearance of the document.

HTML elements are the building blocks of HTML pages. With HTML constructs, images and other objects such as interactive forms may be embedded into the rendered page. HTML provides a means to create structured documents by denoting structural semantics for text such as headings, paragraphs, lists, links, quotes and other items.

HTML elements are delineated by tags, written using angle brackets. Tags such as <img/> and <input/> directly introduce content into the page. Other tags such as <p> surround and provide information about document text and may include other tags as sub-elements. Browsers do not display the HTML tags, but use them to interpret the content of the page.

HTML can embed programs written in a scripting language such as JavaScript, which affects the behavior and content of web pages. Inclusion of CSS defines the look and layout of content.

CSS:

Stands for "Cascading Style Sheet". Cascading style sheets are used to format the layout of Web pages. They can be used to define text styles, table sizes, and other aspects of Web pages that previously could only be defined in a page's HTML.

CSS helps Web developers create a uniform look across several pages of a Web site. Instead of defining the style of each table and each block of text within a page's HTML, commonly used styles need to be 38 defined only once in a CSS document. Once the style is defined in cascading style sheet, it can be used by any page that references the CSS file. Plus, CSS makes it easy to change styles across several pages at once. For example, a Web developer may want to increase the default text size from 10pt to 12pt for fifty pages of a Web site. If the pages all reference the same style sheet, the text size only needs to be changed on the style sheet and all the pages will show the larger text.

While CSS is great for creating text styles, it is helpful for formatting other aspects of Web page layout as well. For example, CSS can be used to define the cell padding of table cells, the style, thickness, and color of a table's border, and the padding around images or other objects. CSS gives Web developers more exact control over how Web pages will look than HTML does. This is why most Web pages today incorporate cascading style sheets.

PHP :

Hypertext Preprocessor(PHP) is a server-side scripting language embedded in HTML in its simplest form. PHP allows web developers to create dynamic content and interact with databases. PHP is known for its simplicity, speed, and flexibility — features that have made it a cornerstone in the web development world.

PHP is a loosely-typed language that uses eight data types to construct variables (which store data of different types). Unlike other programming languages, PHP is quite relaxed regarding variables, evaluating and guessing the data type. All variables have a dollar sign ($) to start but can be named anything that begins with a letter.

PHP utilities that can significantly enhance your coding process. Numerous libraries and classes in PHP help with debugging, testing, profiling, and writing code, among other things.

PHP developers have access to rich frameworks, databases, and libraries to support their work, with the flexibility to set up on any Linux, Windows, or Unix OS. Most web hosting providers offer PHP, and when it comes to cost, PHP often comes out ahead in both development time and the overall cost to run and maintain. As an efficient language, PHP can deliver the high-performance times demanded by today's consumers.

- PHP is an acronym for "PHP: Hypertext Preprocessor"
- PHP is a widely-used, open source scripting language
- PHP scripts are executed on the server
- PHP is free to download and use
- PHP can generate dynamic page content
- PHP can create, open, read, write, delete, and close files on the server
- PHP can collect form data
- PHP can send and receive cookies
- PHP can add, delete, modify data in your database
- PHP can be used to control user-access
- PHP can encrypt data
- PHP files can contain text, HTML, CSS, JavaScript, and PHP code
- PHP code is executed on the server, and the result is returned to the browser as plain HTML
- PHP runs on various platforms (Windows, Linux, Unix, Mac OS X, etc.)
- PHP is compatible with almost all servers used today (Apache, IIS, etc.)
- PHP supports a wide range of databases
- PHP is easy to learn and runs efficiently on the server side

Bootstrap :

Bootstrap is the most popular CSS Framework for developing responsive and mobile-first websites. Bootstrap utilizes Sass for a modular and customizable architecture. Import only the components you need, enable global options like gradients and shadows, and write your own CSS with our variables, maps, functions, and mixins.

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

In other words, Bootstrap helps web developers build websites faster as they don't need to worry about basic commands and functions. It consists of HTML,CSS and JS-based scripts for various web design-related functions and components.

Bootstrap 5 is evolving with each release to better utilize CSS variables for global theme styles, individual components, and even utilities. We provide dozens of variables for colors, font styles, and more at a root level for use anywhere. On components and utilities, CSS variables are scoped to the relevant class and can easily be modified.

Advantages of Bootstrap:

- **Easy to use:** Anybody with just basic knowledge of HTML and CSS can start using Bootstrap
- **Responsive features:** Bootstrap's responsive CSS adjusts to phones, tablets, and desktops
- **Mobile-first approach:** In Bootstrap 3, mobile-first styles are part of the core framework
- **Browser compatibility:** Bootstrap is compatible with all modern browsers (Chrome, Firefox, Internet Explorer, Edge, Safari, and Opera) Node.js runtime on OS X, Microsoft Windows, and Linux

MYSQL:

MySQL is a fast, easy-to-use RDBMS being used for many small and big businesses. MySQL is developed, marketed and supported by MySQL AB, which is a Swedish company. MySQL is becoming so popular because of many good reasons –

- MySQL is released under an open-source license. So you have nothing to pay to use
- MySQL is a very powerful program in its own right. It handles a large subset of the functionality of the most expensive and powerful database packages.
- MySQL uses a standard form of the well-known SQL data language.

MySQL works on many operating systems and with many languages including PHP, PERL, C, C++, JAVA, etc.

- MySQL works very quickly and works well even with large data sets.
- MySQL is very friendly to PHP, the most appreciated language for web development.
- MySQL supports large databases, up to 50 million rows or more in a table. The default file size limit for a table is 4GB, but you can increase this (if your operating system can handle it) to a theoretical limit of 8 million terabytes (TB).
- MySQL is customizable. The open-source GPL license allows programmers to modify the MySQL software to fit their own specific environments.

XAMPP SERVER:

XAMPP is an abbreviation where *X stands for Cross-Platform, A stands for Apache, M stands for* MYSQL, *and the Ps stand for PHP and Perl,* respectively. It is an open-source package of web solutions that includes Apache distribution for many servers and command-line executables along with modules such as Apache server, MariaDB, PHP, and Perl.

In a nutshell, XAMPP is a local server that you can install on your laptop/desktop to mimic an actual web server. XAMPP helps a local host or server to test its website and clients via computers and laptops before releasing it to the main server. It is a platform that furnishes a suitable environment to test and

verify the working of projects based on Apache, Perl, MySQL database, and PHP through the system of the host itself. Among these technologies, Perl is a programming language used for web development, PHP is a backend scripting language, and MariaDB is the most vividly used database developed by MySQL.

- The creators of XAMPP intended it to be used as a development tool, allowing web designers and programmers to test their work on their personal computers without the need for Internet connections. Many key security elements are disabled by default to make this as simple as feasible. XAMPP is used to serve the web pages on the Internet.
- It can also use to create and manipulate databases in MariaDB and SQLite, among other databases.
- Once XAMPP is installed, an FTP client can connect to a local host and treat it as if it were a remote host. When installing a content management system like Joomla or WordPress, using a tool like FileZilla. You can also use an HTML editor to connect to a local host through FTP.

CODING

Designing a complete page using PHP :

```
<?php
$db_connect = mysqli_connect('localhost','root','root','book_store') or die('connection failed');
if(isset($_GET["delete"])){
  mysqli_query($db_connect,"DELETE FROM `admin_d` WHERE a = '$_GET[delete]' ");
  header("location: admin_del.php");
}
?>
<!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>feedbacks</title>
  <link rel="stylesheet" href="admin_style.css">
  <style>
  .navbar{
    display: flex;
    background-color:rgb(0,0,0,0.3);
    position: fixed;
    top:0;
    width: 100%;
    font-size: 20px;
  }
   .nav-right{

  text-align: right;
    float:right;
```

```
  }
  .nav-right li {
     display: inline;
     float: left;
     padding-left: 15px;
    }
    .nav-right li a {
     display: block;
     padding: 8px;
     text-decoration: none;
     color:white;
    }
    .nav-right  li ul {
     list-style-type: none;
     margin: 0;
     padding: 0;
     overflow: hidden;
     background-color:rgb(0,0,0,0.8);
    }
    .nav-right li a:hover {
     background-color: rgba(250,96,0,0.8);
    }
    .nav-right .active {
     background-color: #04AA6D;
    }
body {
background-image:url("https://wallpapers.com/images/hd/colorful-book-stack-outside-
dg76b6qqzwaavwx1.jpg");
  background-size: "fixed";
  background-repeat: "no-repeat";
}
```

```
</style>
</head>
<body>
<header class="header">
<nav class="navbar">
        <div class="nav-right">
            <li><a href="front.html">Home</a></li>
            <li><a href="feed_up.php">Feedbacks</a></li>
            <li><a href="admin_add.html">Add Admin</a></li>
            <li><a href="admin_del.php">Remove Admin</a></li>
            <li><a href="pay_status1.php">Orders pending</a></li>
         </div>
   </nav>
</div>
</header>
  <section class="orders">
  <h1 class="title"></h1>
    <h1 class="title">Admins</h1>
    <div class="box-container">
      <?php
     $all_orders = mysqli_query($db_connect, "SELECT * FROM `admin_d`");
     if (mysqli_num_rows($all_orders) > 0) {
       while ($row = mysqli_fetch_assoc($all_orders)) {
     ?>
        <div class="box">
          <p> user id : <span><?= $row['a'] ?></span> </p>
          <p> password : <span><?= $row['b'] ?></span> </p>
          <form action="" method="post">
           <a href="admin_del.php?delete=<?= $row["a"] ?>" onclick="return confirm('delete the
feedback?');" class="delete-btn">delete</a>
```

```
 </form>
        </div>
     <?php
       }
     } else {
       echo '<p class="empty">Admins are nill!</p>';
     }
     ?>
   </div>
  </section>
</body>
</html>
```

Backend PHP code :

```
<?php
$server_name="localhost";
$username="root";
$password="root";
$database_name="book_store";
$conn=mysqli_connect($server_name,$username,$password,$database_name);
//now check the connection
if(!$conn)
{
   die("Connection Failed:" . mysqli_connect_error());
}
if(isset($_POST['save1']))
{
    $first_name = $_POST['user1'];
    $last_name = $_POST['pass1'];
    $sql_query = "INSERT INTO admin_d(a,b)
    VALUES ('$first_name','$last_name')";
```

```
 if (mysqli_query($conn, $sql_query))
    {
       header("Location: http://localhost/DBMS/admin_del.php");
    }
    else
    {
       echo "invalid details";
    }
    mysqli_close($conn);
}
?>
```

Frontend CSS code :

```
body {margin:0;}
.navbar{
   display: flex;
   background-color:rgb(0,0,0,1);
   position: fixed;
   top:0;
   width: 100%;
   font-size: 20px;
 }
 .nav-right{
   text-align: right;
   float:right;
 }
 .nav-right li {
     display: inline;
     float: left;
     padding-left: 15px;
   }
   .nav-right li a {
     display: block;
     padding: 8px;
     text-decoration: none;
     color:white;
   }
   .nav-right  li ul {
     list-style-type: none;
```

```
      margin: 0;

      padding: 0;
      overflow: hidden;
      background-color:rgb(0,0,0,1);
    }
    .nav-right li a:hover {
      background-color: rgba(250,96,0,0.8);
    }
    .nav-right .active {
      background-color: #04AA6D;
    }
    .navbar .social_icon i{
      margin: 0 5px;
      font-size: 18px;
  }
.section-1-text{
    display: block;
    border: black;
}
h1{
    font-size: 100px;
    color: white;
}
.section-1 p{
    font-size: 20px;
    color: white;
}
.section-2
{
    padding: 55px;
}
.gallery div{
    display: inline !important;
    border:10px solid white;
}
.section-2  a{
    text-decoration: none;
}
.section-2 p{
    color: black;
    text-align: center;
}
div.gallery a:hover {
    border: 10px solid white;
}
```

```
div.gallery image-holder desc {
   text-align: center;
 }

img{
   height: 300px;
   width:250px;
   border:10px solid white;
}
.btn
{
   display: block;
   border-radius: 5px;
   color:white;
   background-color: rgba(250,96,0,0.8);
   font-size:20px;
   text-decoration: none;
   width:fit-content;
   height:fit-content;
   padding:5px;
}
.btn:hover{
   background-color:rgba(250,96,0,0.8);
}
li {
   display: inline;
   float: left;
   padding-left: 15px;
  }
  li a {
   display: block;
   padding: 8px;
   text-decoration: none;
   color:white;
  }
  li ul {
   list-style-type: none;
   margin: 0;
   padding: 0;
   overflow: hidden;
   background-color: #333;
  }
  li a:hover {
   background-color: rgba(250,96,0,0.8);
  }
  .active {
```

```
    background-color: #04AA6D;
  }

/* section-3 image gallery 2 */

  .section-3{
      padding:35px;
  }
  .view-card {
    float: left;
    width: 25%;
    padding: 0 10px;
}
  .card {
    box-shadow: 0 4px 8px 0 rgba(0, 0, 0, 0.2);
    padding: 16px;
    text-align: center;
    display: inline;
  }
  .card:hover {
    box-shadow: 0 8px 16px 0 rgba(0,0,0,0.2);
  }
  .container {
    padding: 2px 16px;
    align-items: center;
  }

  /* section-4 brands */

  .section-4{
    padding:35px;
  }
  .brand{
      width: 20%;
      display: inline;
  }
  .brand img{
      height: 175px;
  }
.services{
    padding:35px;
    width: 100%;
    height: auto;
    margin: 35px 0;
}
.services .services_box{
```

```
    width: 100%;
    margin: 0 auto;
    display: flex;
    align-items: center;

 justify-content: space-around;
}
.services .services_box .services_card{
    text-align: center;
    width: 200px;
    height: 200px;
    box-shadow: 0 0 8px rgba(250,96,0,0.8);
    padding: 15px 10px;
}
.services .services_box .services_card i{
    color: rgba(250,96,0,0.8);
    font-size: 45px;
    margin-bottom: 15px;
    cursor: pointer;
}
.services .services_box .services_card h3{
    margin-bottom: 10px;
}
.blog{
  padding:35px;
  width: 100%;
  height: auto;
  margin: 35px 0;}
.blog .blog_box{
  width: 95%;
  margin: 0 auto;
  display: grid;
  grid-template-columns: 1fr 1fr 1fr;
}
.blog .blog_box .blog_card{
  margin: 0 auto;
  width: 450px;
  height: auto;
  box-shadow: 0 0 8px rgba(0,0,0,0.5);
}
.blog .blog_box .blog_card .blog_img{
  width: 450px;
  height: 300px;
  overflow: hidden;
}
.blog .blog_box .blog_card .blog_img img{
```

```
  width: 100%;
  height: 100%;
  object-fit: cover;
  object-position: center;
}


.blog .blog_box .blog_card .blog_tag{
  padding: 20px;
}
.blog .blog_box .blog_card .blog_tag h2{
  color:rgba(250,96,0,0.8);
}
.blog .blog_box .blog_card .blog_tag p{
  margin-top: 10px;
  text-align: justify;
  line-height: 22px;
}
.blog .blog_box .blog_card .blog_tag .blog_icon{
  display: flex;
  align-items: center;
  justify-content: space-between;
  color: rgba(250,96,0,0.8);
  cursor: pointer;
}

footer{
  width: 100%;
  background: rgba(0,0,0,1);
}
footer .footer_main{
  width: 100%;
  display: flex;
  justify-content: space-around;
}
footer .footer_main .tag{
  margin: 10px 0;
}
footer .footer_main .tag img{
  width: 100px;
  margin-bottom: 10px;
}
footer .footer_main .tag p{
  width: 250px;
  line-height: 22px;
  text-align: justify;
```

```
}
footer .footer_main .tag h1{
  font-size: 25px;
  margin: 25px 0;
  color: #000;
}



footer .footer_main .tag a{
  display: block;
  color: black;
  text-decoration: none;
  margin: 10px 0;
}
footer .footer_main .tag i{
  margin-right: 10px;
}
footer .footer_main .tag .social_link i{
  margin: 0 5px;
}
footer .footer_main .tag .search_bar{
  width: 230px;
  height: 30px;
  background: rgba(202,202,202);
  border-radius: 25px;
}
footer .footer_main .tag .search_bar input{
  width: 200px;
  padding: 2px 0;
  position: relative;
  top: 17%;
  left: 6%;
  border: none;
  outline: none;
  font-size: 13px;
  background: none;
}
footer .footer_main .tag .search_bar button{
  padding: 7px 15px;
  background: rgba(250,96,0,0.8);
  border: none;
  margin-top: 15px;
  border-radius: 25px;
  color: #fff;
  cursor: pointer;
```

```
}
footer .end{
  display: flex;
  align-items: center;
  justify-content: center;
  padding: 15px;
  color: #000;
}



footer .end span{
  color: rgba(250,96,0,0.8);
  margin-left: 10px;
}
```

Frontend HTML code :

```
<!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">

    <script src="https://use.fontawesome.com/be1ba39dfe.js"></script>

    <link rel="stylesheet" href="./css/style.css">

                                                    <link                    rel="stylesheet"
href="https://cdn.jsdelivr.net/npm/bootstrap@4.1.3/dist/css/bootstrap.min.css"
integrity="sha384-
MCw98/SFnGE8fJT3GXwEOngsV7Zt27NXFoaoApmYm81iuXoPkFOJwJ8ERdknLPMO"
crossorigin="anonymous">

                <link          rel="stylesheet"          href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/6.1.1/css/all.min.css"                                                    integrity="sha512-
KfkfwYDsLkIlwQp6LFnl8zNdLGxu9YAA1QvwINks4PhcElQSvqcyVLLD9aMhXd13uQjoX
tEKNosOWaZqXgel0g==" crossorigin="anonymous" referrerpolicy="no-referrer" />

    <title>Book Zone</title>

</head>

<!doctype html>

<html lang="en">
```

```
<body>

    <script src="https://code.jquery.com/jquery-3.3.1.slim.min.js" integrity="sha384-q8i/X+965DzO0rT7abK41JStQIAqVgRVzpbzo5smXKp4YfRvH+8abtTE1Pi6jizo" crossorigin="anonymous"></script>

    <script src="https://cdn.jsdelivr.net/npm/popper.js@1.14.3/dist/umd/popper.min.js" integrity="sha384-ZMP7rVo3mIykV+2+9J3UJ46jBk0WLaUAdn689aCwoqbBJiSnjAK/l8WvCWPIPm49" crossorigin="anonymous"></script>

    <script src="https://cdn.jsdelivr.net/npm/bootstrap@4.1.3/dist/js/bootstrap.min.js" integrity="sha384

ChfqqxuZUCnJSK3+MXmPNIyE6ZbWh2IMqE241rYiqJxyMiZ6OW/JmZQ5stwEULTy" crossorigin="anonymous"></script>

  <nav class="navbar">

    <div class="nav-right">

      <li><a href="./admin.html"><img src="./logo.png" style="width: 60px; height:40px" ></a></li>

      <li><a href="./front.html">Home</a></li>

      <li><a href="./products.html">Products</a></li>

      <li><a href="./feedback.html">Feedback</a></li>

      <li><a href="./About.html">Team</a></li>

    </div>

    <div class="social_icon">

      <i class="fa-solid fa-magnifying-glass"></i>

      <i class="fa-solid fa-heart"></i>

    </div>

  </nav>

  <div class="section-1">

    <div id="carouselExampleIndicators" class="carousel slide" data-ride="carousel">

      <ol class="carousel-indicators">

        <li data-target="#carouselExampleIndicators" data-slide-to="0" class="active"></li>

        <li data-target="#carouselExampleIndicators" data-slide-to="1"></li>

        <li data-target="#carouselExampleIndicators" data-slide-to="2"></li>
```

```
        </ol>
                <div id="carouselExampleFade" class="carousel slide carousel-fade" data-ride="carousel">
            <div class="carousel-inner">
              <div class="carousel-item active">
                <img class="d-block w-100" src="orffer4.jpg" alt="forth slide">
              </div>
             <div class="carousel-item ">
              <img class="d-block w-100" src="offer1.jpg" alt="First slide">
             </div>
             <div class="carousel-item">
              <img class="d-block w-100" src="offer3.jpg" alt="Second slide">
             </div>
             <div class="carousel-item">
              <img class="d-block w-100" src="offer5.jpg" alt="Third slide">
             </div>
            </div>
             <a class="carousel-control-prev" href="#carouselExampleFade" 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="#carouselExampleFade" role="button" data-slide="next">
              <span class="carousel-control-next-icon" aria-hidden="true"></span>
              <span class="sr-only">Next</span>
            </a>
          </div>
    </div>
    <div class="section-2">
       <h1>Best Sellers</h1>
```

```
<div class="gallery">
    <div class="image-holder">
        <a href="#">
                                                <img src="https://storage.googleapis.com/du-prd/books/images/9781791392796.jpg" alt="img-1">
        </a>
    </div>
    <div class="image-holder">
        <a href="#">
                                                <img src="https://storage.googleapis.com/du-prd/books/images/9780735219090.jpg" alt="img-2" style="margin-left:10px">
        </a>
    </div>
    <div class="image-holder">
        <a href="#">
            <img src="https://encrypted-tbn0.gstatic.com/images?q=tbn:ANd9GcTW6tJbod-fARZeVyBaIbqCjXbR6Sshj3jqT9OwbazRTt_TKTwN3-OGpvF_14JSAKmYpBA&usqp=CAU" alt="img-3" style="margin-left:5px">
        </a>
    </div>
    <div class="image-holder">
        <a href="#">
                                                <img src="https://prodimage.images-bn.com/pimages/9781250301703_p0_v5_s192x300.jpg" alt="img-4" style="margin-left:10px">
        </a>
    </div>
    <div class="image-holder">
        <a href="#">
                                                                                <img src="https://smallbiztrends.com/ezoimgfmt/media.smallbiztrends.com/2022/11/when-mckinsey-comes-to-town-business-book-of-the-year-2022.jpg?ezimgfmt=rs:345x520/rscb12/ngcb12/notWebP" alt="img-5" style="margin-left:5px">
```

```
    </a>
  </div>
  <div class="image-holder">
    <a href="#">
      <img src="https://m.media-amazon.com/images/I/71ZKMWgafjL._AC_UL320_.jpg" alt="img-6" style="margin-left:2px;margin-top:20px">
    </a>
  </div>
  <div class="image-holder">
    <a href="#">
      <img src="https://images-na.ssl-images-amazon.com/images/S/compressed.photo.goodreads.com/books/1633097753i/40132775.jpg" alt="img-7" style="margin-right:1px ;margin-top:20px">
    </a>

</div>
  <div class="image-holder">
    <a href="#">
      <img src="https://encrypted-tbn0.gstatic.com/images?q=tbn:ANd9GcRQyhJKkL9pApJieceJP6z2_COf8-quk8h5yA&usqp=CAU" alt="img-8" style="margin-left:5px ;margin-top:20px">
    </a>
  </div>
  <div class="image-holder">
    <a href="#">
      <img src="https://storage.googleapis.com/du-prd/books/images/9781668001226.jpg" alt="img-9" style="margin-left:5px ;margin-top:20px">
    </a>
  </div>
  <div class="image-holder">
    <a href="#">
```

<img src="https://encrypted-tbn0.gstatic.com/images?q=tbn:ANd9GcT-2jWXRM_wJp7NHCV1KP8m1XrBBqYutWdAnycODnwbk6Scv6fp492Xd_S6NBePeQ7jRBU&usqp=CAU" alt="img-10" style="margin-left:5px ;margin-top:10px">

<h1 style="padding-left:10px">Featured Authors</h1>

<img src="https://m.media-amazon.com/images/S/amzn-author-media-prod/ddl6isug919ukj2r4bla21gm4q.jpg" alt="" style="margin-left:30px;margin-right:20px ;margin-top:10px">

<img src="https://www.pirates.ind.in/wp-content/uploads/2017/09/Dale-Carnegie-GQ-12May14_rex_b.jpg" alt="" style="margin-right:20px;margin-top:10px">

<img src="https://www.shethepeople.tv/wp-content/uploads/2017/09/agatha-christie-e1505461541415.jpg" alt="" style="margin-right:20px;margin-top:10px">

<img src="https://encrypted-tbn0.gstatic.com/images?q=tbn:ANd9GcSA3uBWaeoiJM83VSC70uD_gdswURbHLzbdZg&usqp=CAU" alt="" style="margin-right:20px;margin-top:10px">

<img src="https://images.squarespace-cdn.com/content/v1/51b81078e4b018d977c2fada/1371592566414-K4KZMY4W9TXCLNGQS9MW/Chitra.gif" alt="" style="margin-right:20px;margin-top:10px">

```
  </div>
  <!-- section-5 contact us  -->
  <div class="services">
     <h1  style="padding-left:10px" >Our Services</h1>
     <div class="services_box">
        <div class="services_card">
           <i class="fa-solid fa-truck-fast"  style="margin-top:30px"></i>
           <h3>Fast Delivery</h3>
        </div>
        <div class="services_card">
           <i class="fa-solid fa-headset" style="margin-top:10px"></i>
           <h3>24 x 7 Services</h3>
        </div>
        <div class="services_card" >
           <i class="fa-solid fa-tag" style="padding-top:30px"></i>
           <h3>Best Deal</h3>
        </div>
<div class="services_card" >
           <i class="fa-solid fa-lock"  style="margin-top:10px"></i>
           <h3>Secure Payment</h3>
        </div>
     </div>
  </div>
  <div class="blog">
     <h1>Books Review</h1>
     <div class="blog_box">
        <div class="blog_card">
           <div class="blog_img">
              <img src="https://m.media-amazon.com/images/I/71kLBC9eHAL.jpg">
           </div>
```

<h2>Review1</h2>

As the title of the book suggests, it promises to be an insightful look into powers beyond the scope of our daily prowess. There is a lot of information imparted in the book, which is divided into sections, and the theories grow in their complexity as you near the end. It’s easy to follow the examples stated in the book that act as vivid similes to break down complex statements.

At times, it can also be read like a self-help book for those who lack the motivation to push through their defenses and will their thoughts into actions.That makes total sense because we can be one of the biggest obstacles in our path due to personal prejudices whether we like to address them or not.

Please do explore the book

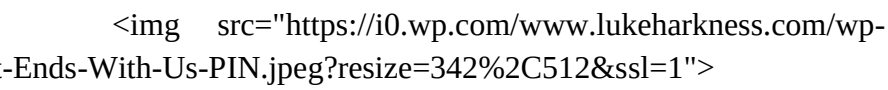
<h2>Review2</h2>

This book is based on the lessons that Jay had learned during his time as Monk in India. Most of the content is something you had already read somewhere or heard someone say it to you.

This book is a detailed read, one that is bound to take you days to read, and weeks or months to fully implement the lessons that it has to teach. it took me a good 20 days to finish this book.

The book is broadly divided into three sections – Let Go , Grow and Give, the book effectively charts out a course for living a more peaceful and content life.

When picking up this book, please remember that this book is not about growing professionally but personally and spiritually. in those pages lies life lessons that teach you to think and act like with a monk's mind.



<h2>Review3</h2>

After reading this book I was confused as to how I felt about it. On one hand, there is a clear depth to it; the book talks about abusive relationships in a raw, vulnerable way that makes you think about the mental and emotional trauma involved, as well as the complexity of such relationships.

On the other hand, there is also a lot of what I see as juvenile romance – irrational, childish expectations from relationships, overflowing emotion, tantrums, rapid switches between extreme highs and lows, and levels of drama that no rational adult would ever subject themselves to.After reading the book, my aversion to romance novels remains unchanged, but I did find myself feeling much more empathy for victims of domestic abuse

```
                <i class="fa-solid fa-heart"></i>
            </div>
        </div>
    </div>

</div>
</div>
<footer>
<div class="footer_main">
<div class="tag">
<p style="color:white">
<br>
Bookhouse!!!!!The one stop book destination!Explore!Buy!Feel the crunch of the book
<br>
<br>
Enjoy shopping
</p>
</div>
<div class="tag">
<h1 style="color:white">Quick Link</h1>
<a href="#" style="color:white">Home</a>
</div>
<div class="tag">
<h1 style="color:white" >Contact Info</h1>
<a href="#" style="color:white"><i class="fa-solid fa-phone" style="color:white"></i>+94 12 345 6789</a>
<a href="#" style="color:white"><i class="fa-solid fa-envelope" style="color:white"></i>bookstore123@gmail.com</a>
</div>
<div class="tag">
```

```
            <h1 style="color:white">Follow Us</h1>
            <div class="social_link">
                <i class="fa-brands fa-facebook-f" style="color:white"></i>
                <i class="fa-brands fa-instagram" style="color:white"></i>
                <i class="fa-brands fa-twitter" style="color:white"></i>
                <i class="fa-brands fa-linkedin-in" style="color:white"></i>
            </div>     </div>
        <div class="tag">
            <h1 style="color:white">Newsletter</h1>
            <div class="search_bar">
                <input type="text" placeholder="You email id here">
                <button type="submit">Subscribe</button>
            </div>
        </div>
    </div>
    <p class="end" style="color:white">Design By<span><i class="fa-solid fa-face-grin"></i>
Harika & team</span></p>
  </footer>
</body>
</html>
```

TESTING:

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a product. It provides a way to check the functionality of components, sub assemblies, assemblies and/or a finished product. It is the process of exercising software with the intent of ensuring that the software system meets its requirements and user expectations and does not fail in an unacceptable manner.

Software testing is an important element of the software quality assurance and represents the ultimate review of specification, design and coding. The increasing feasibility of software as a system and the cost associated with the software failures are motivated forces for well planned through testing.

Testing Objectives:

These are several rules that can save as testing objectives they are:
Testing is a process of executing program with the intent of finding an error.
A good test case is one that has a high probability of finding an undiscovered error.

Types of Testing:

In order to make sure that the system does not have errors, the different levels of testing strategies that are applied at differing phases of software development are:

Unit Testing:

Unit Testing is done on individual modules as they are completed and become executable. It is confined only to the designer's requirements. Unit testing is different from and should be preceded by other techniques, including:
Inform debugging
Code debugging
Each module can be tested using the following two strategies:

Black Box Testing:

In this strategy some test cases are generated as input conditions that fully execute all functional requirements for the program. This testing has been used to find error in the following categories:
Incorrect or missing functions.
Interface errors.
Errors in data structure or external database access.
Performance Error.
Initialization and termination errors.
In this testing only the output is checked for correctness.
The logical flow of the data is not checked.

White Box Testing:

In this the test cases are generated on the logic of each module by drawing flow graphs of that module and logical decisions are tested on all cases. It has been used to generate the test cases in the following cases:
Guarantee that all independent paths have been executed.
Execute all loops at their boundaries and within their operational bounds.
Execute internal data structures to ensure their validity.

Integration Testing

Integration testing ensures that software and subsystems work together a whole. It tests the interface of all the modules to make sure that the modules behave properly when integrated together. It is typically performed by developers, especially at lower, module-to-module level. Testers become involved at the higher levels.

System Testing

Involves in house testing of entire system before delivery to the user. The aim is to satisfy the user, the meets all the requirements of the client's specifications. It is conducted by the testing organization if a company has one. Test data may range from hand generated to production. Requires test scheduling to plan and organize:

Inclusion of changes/fixes.

Test data to use.

One common approach is graduated testing: as system testing progresses and (hopefully) fewer and fewer defects are found, the code is frozen for testing for increasingly longer time periods.

Acceptance Test

It is a pre-delivery testing in which entire system is tested at client's site on real world data to find errors.

User Acceptance Test (UAT)

"Beta Testing": Acceptance testing in the customer environment. Requirements traceability:

Match requirements to test cases.

Every requirement ha to be cleared by at least one test case.

Display in a matrix of requirements vs. test case

Test Cases

In general, a test case is a set of test data and test program and their expected results. A test case in software engineering normally consists of unique identifier, requirement references from a design specification, preconditions, events, a series of steps (also known as actions) to follow, input, output and it validates one or more system requirements and generates a pass or fail.

TEST CASES FOR PROJECT:

In general a test case is a set of test data and test programs and their expected results. A test case in software engineering normally consists of a unique identifier, requirement references from a design specification, pre conditions, events, a series of steps (also known as actions) to follow input, output and it validates one or more system requirements and generates a pass or fail.

INVALID REGISTRATION

Test plan id: Pid1

Test case id: 101

Features to be tested: Checking validity of details during registration

Pre conditions: Website must be running

Test Script:

1.Verify Registraion username,password,email

2.Accept and Update the Database if username doesnot exist

Test Data:

1.Username already exist

2.Username doesnot exist

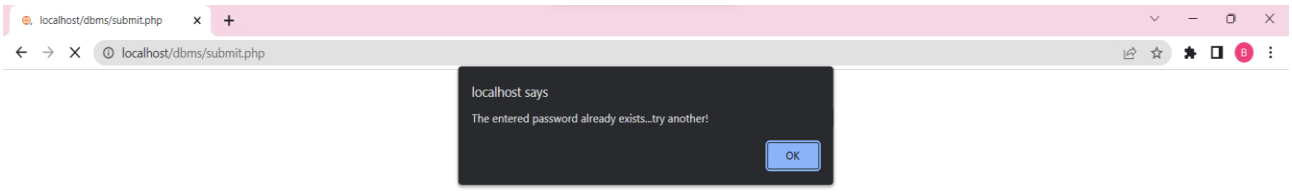
Expected Results:

1.Accept the user registration request if details are valid

2.Display the error message if details are invalid redirect to same page

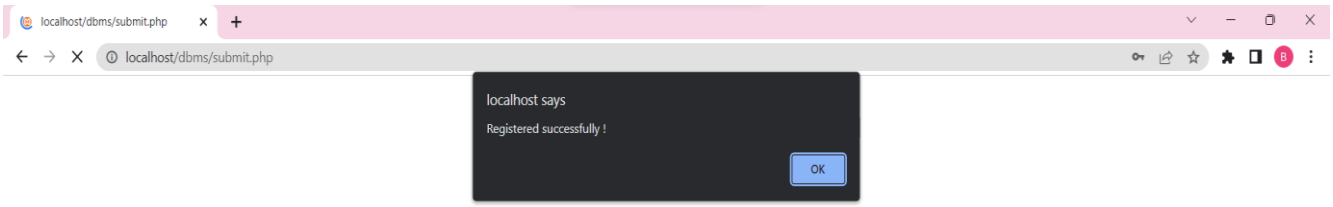
Test status: Fail

Display the error message if details are invalid redirect to same page:



Test status: Success

Accept the user registration request if details are valid:



INVALID LOGIN

Test plan id: Pid2

Test case id: 102

Features to be tested: Checking validity of login

Pre conditions:

1. Website must be running

2. Get Registration Details

Test Script:

1.Verify Login Details

2. Accept and Update the Database if the user detais are valid

Test Data:

1.Valid username and password

2.Invalid username or password

3.New username or password

Expected Results:

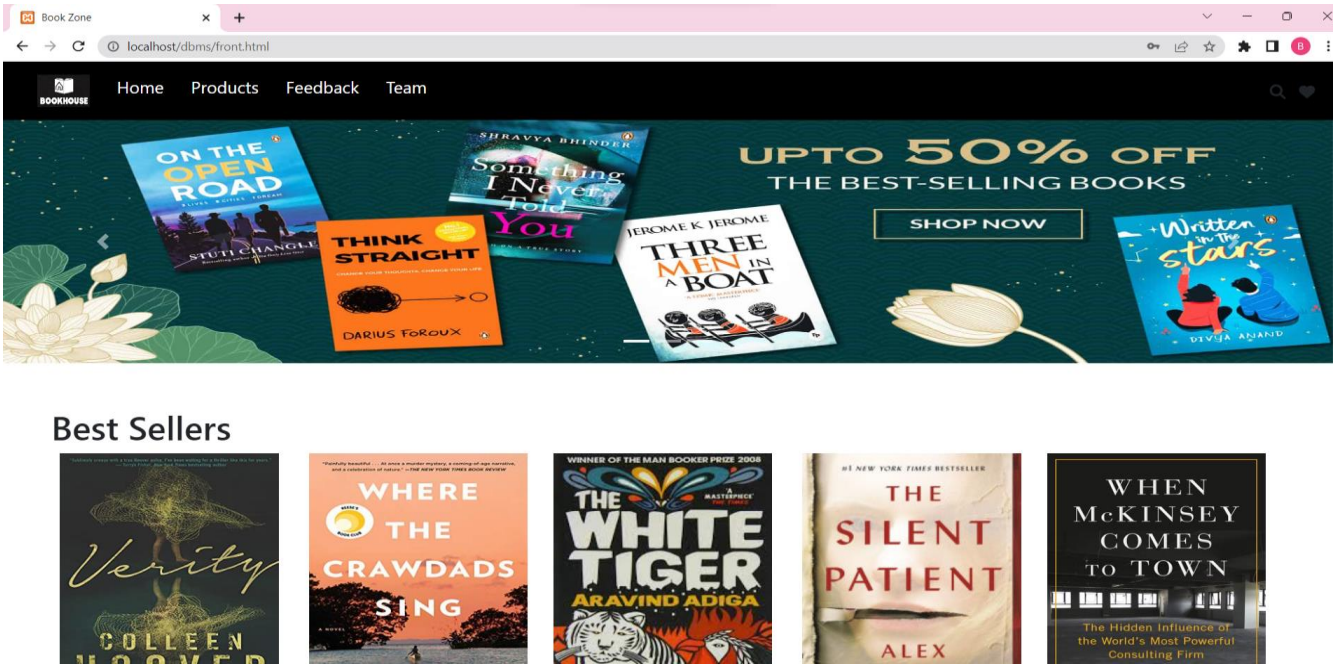
1.Display the Courses outlet page

2.Redirect to the same login page if login details are invalid

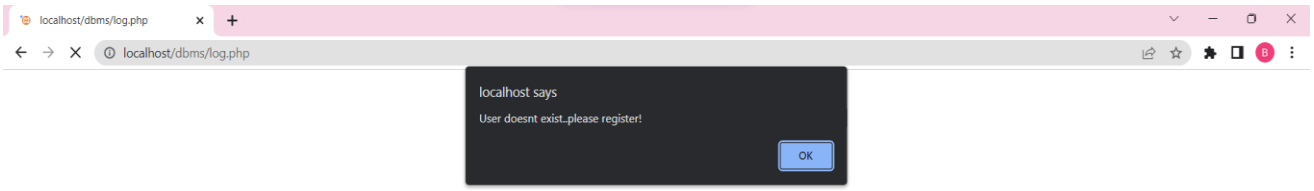
3.Redirect to the Registration page if login details are New

Test Status: Success

Display the Courses outlet page:



Redirect to the Registration page if login details are New:



INVALID ADMIN LOGIN

Test plan id: Pid3

Test case id: 103

Features to be tested: Checking validity of Admin login

Pre conditions:

1. Website must be running

2. Get admin login Details

Test Script:

1.Verify Admin Login Details

2. Accept and Update the Database if the user details are valid

Test Data:

1.Valid username and password

2.Invalid username or password

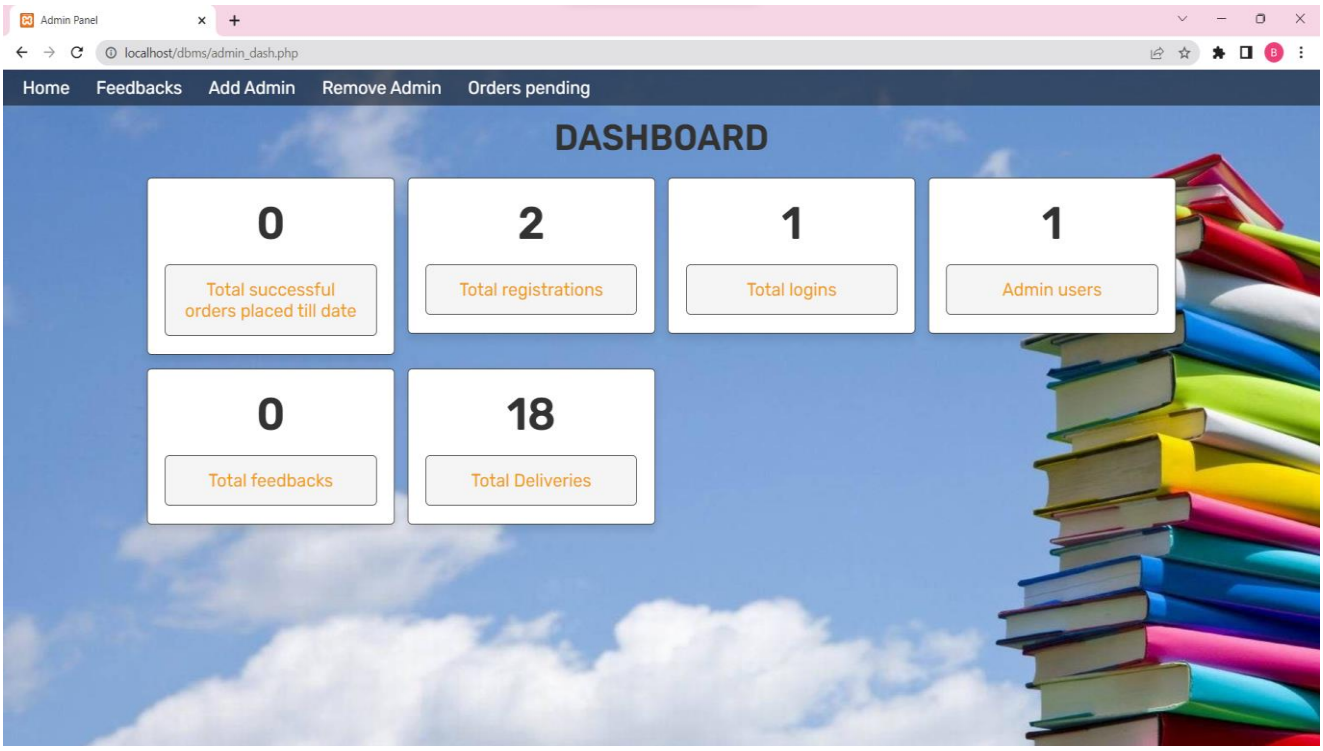
Expected Results:

1.Display the Admin Main page

2.Redirect to the same login page if login details are invalid

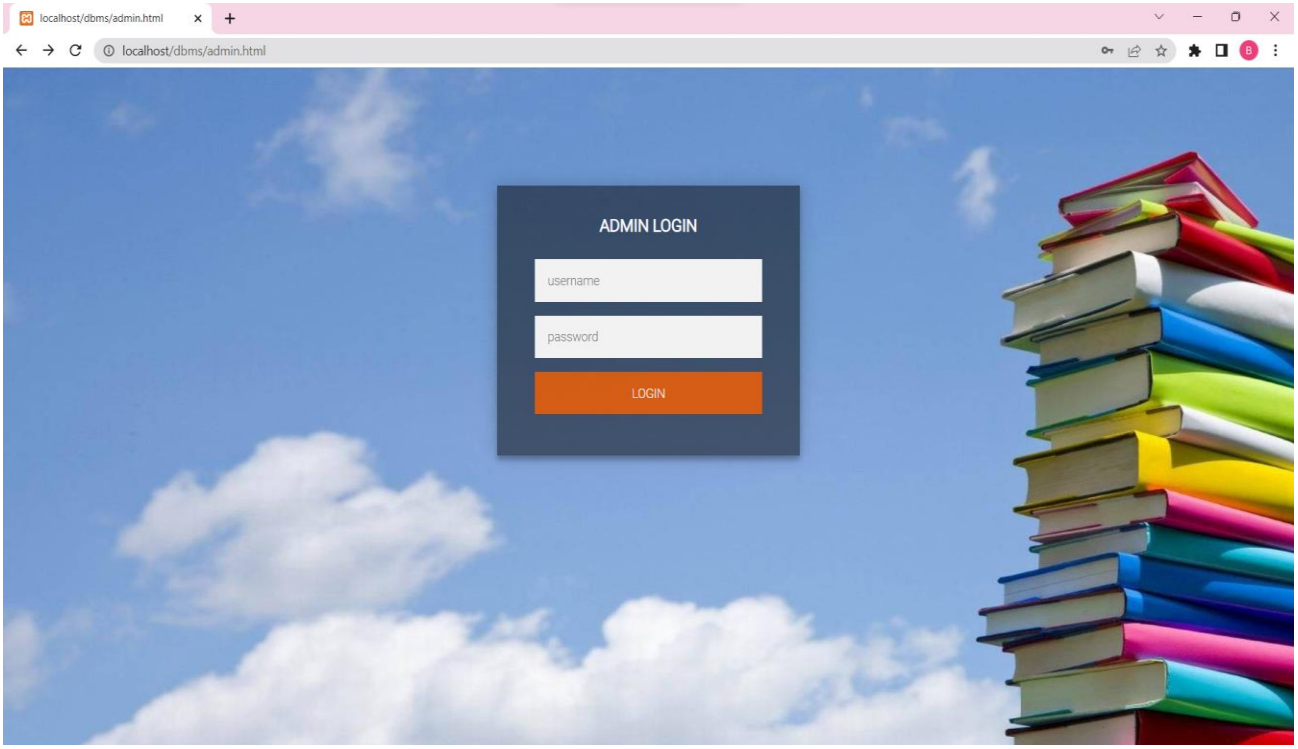
Test status: Success

Display the Admin Main page :



Test status: Fail

Redirect to the same login page if login details are invalid:



6.2.4 GIVING ADMIN ACCESS

Test plan id: Pid4

Test case id: 104

Features to be tested: Checking Whether we can give Admin access

Pre conditions:

1. Website must be running
2. Get admin login Details

Test Script:

1.Enter the new admin details

2. If admins less than 5 then update the database

Test Data:

1.Enter new username and password

Expected Results:

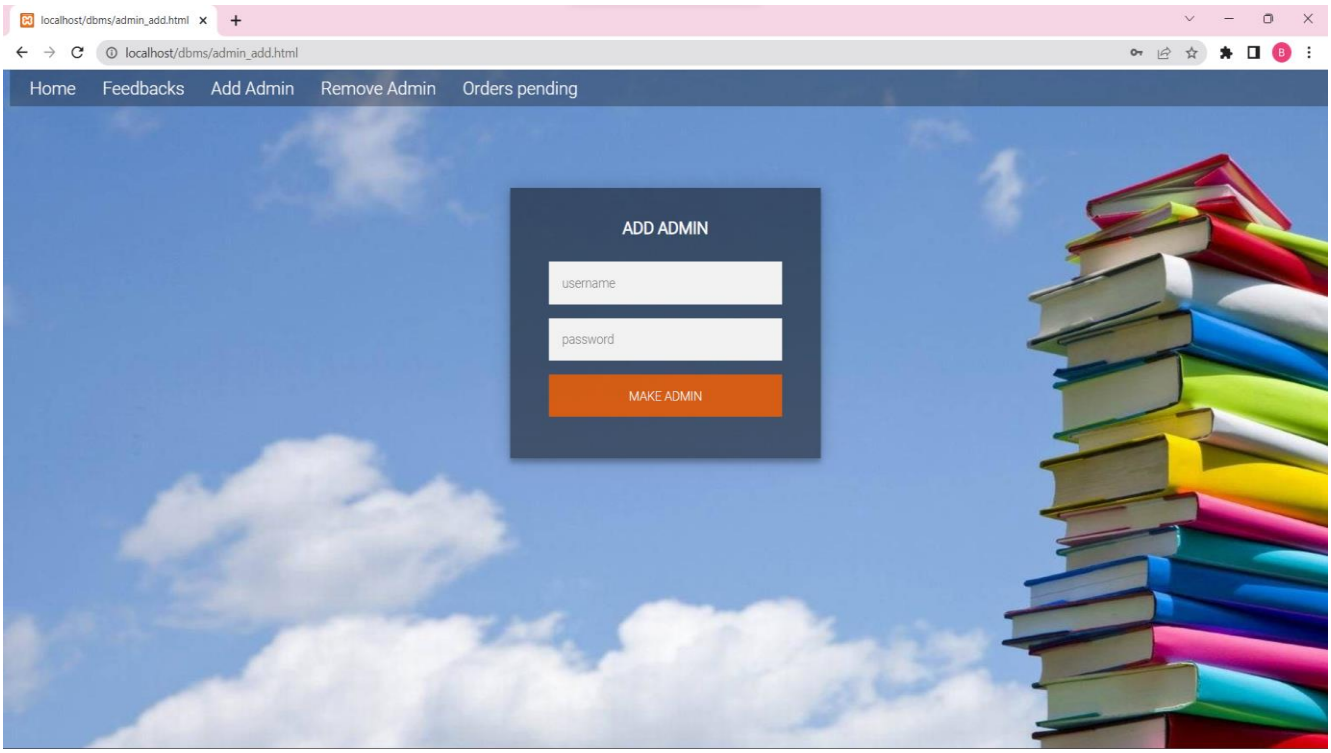
1.Display success message of adding new admin and redirect to admin access page

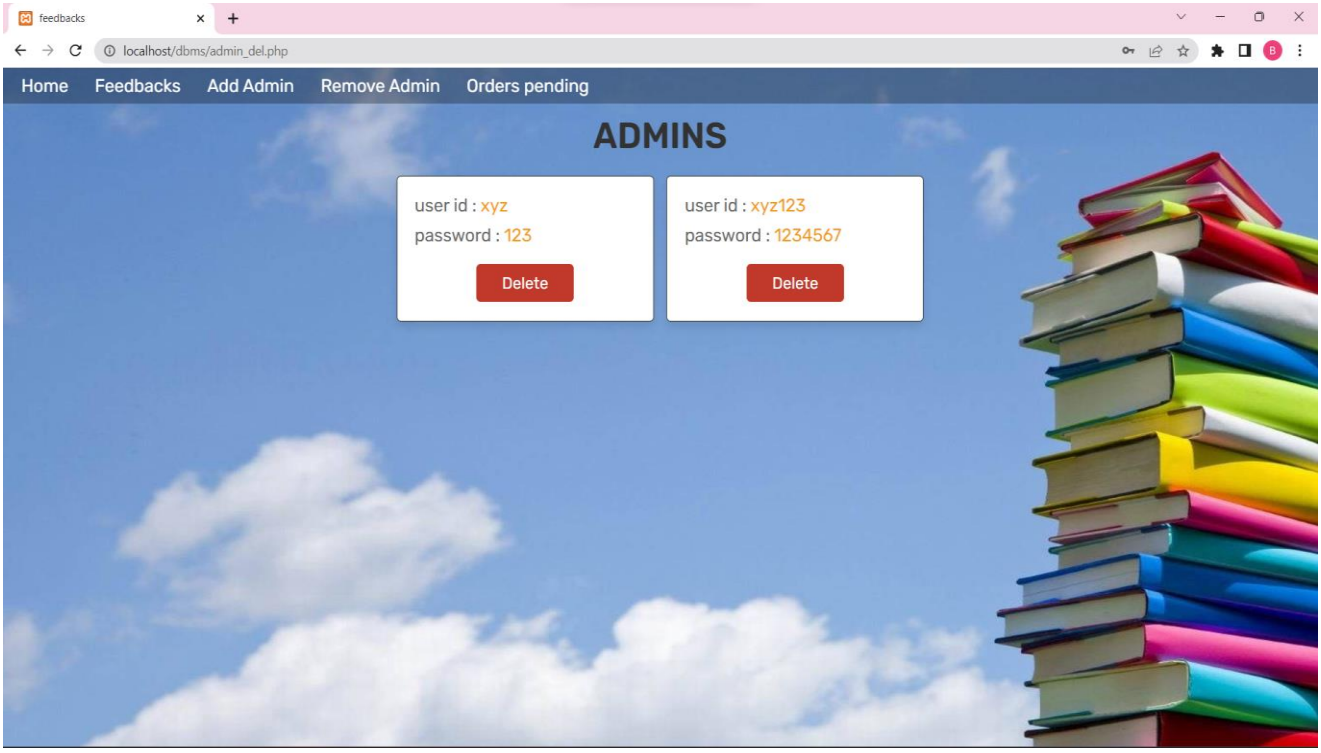
2.Redirect to the admin access page if admins are more than 5 by displaying error message

3. Redirect to the same page if admin already exists.

Test status: Success

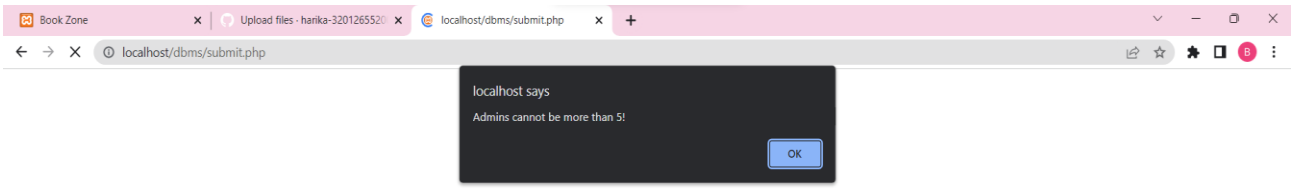
Display of adding new admin:





Test status: Fail

Redirect to the same page if admins are more than 5 by displaying error message:



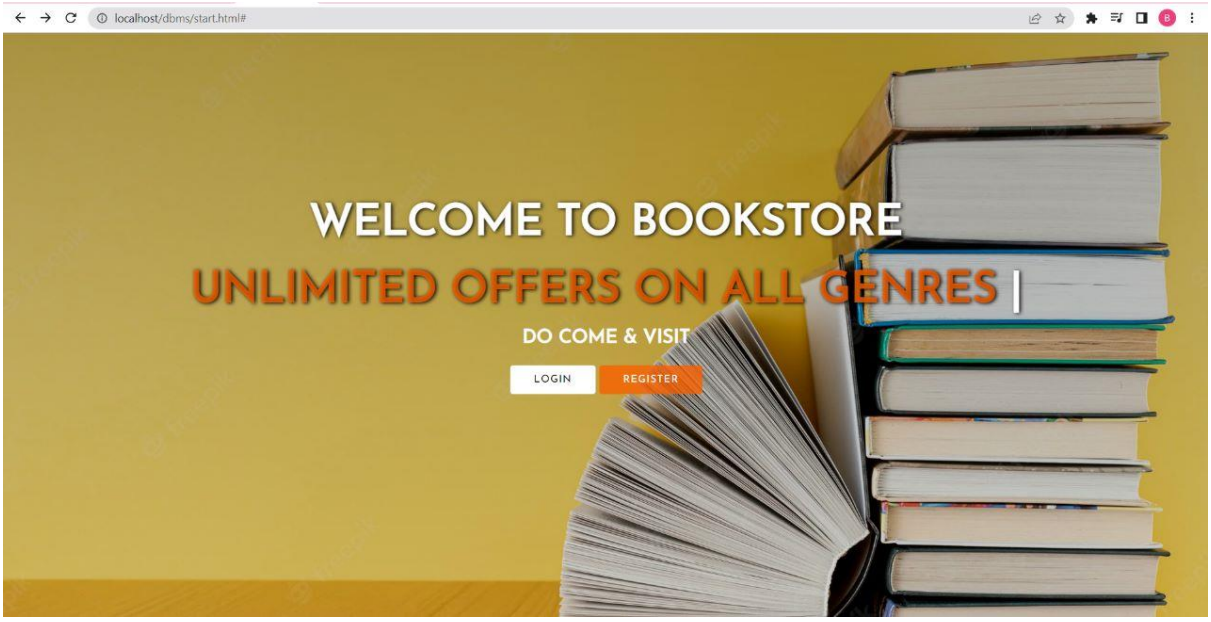
Redirect to the same page if admin already exits:

S.NO	TEST CASE CONDITION	INPUT	ERROR MESSAGE
1	Invalid EMAIL	Harikagmail.com	Please Type your mail address in a valid format
2	Invalid Password	1234	The password you provided must have At least 6 characters
3	Username/Password Duplicity	XYZ XYZ	The username/password already exists please try another

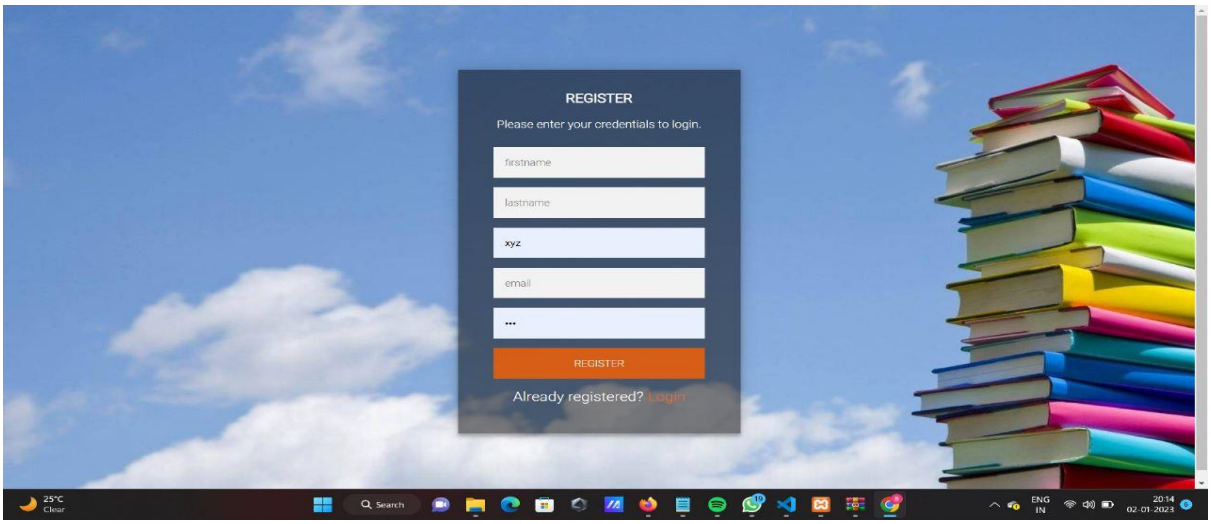
RESULTS

Input/Output Design:

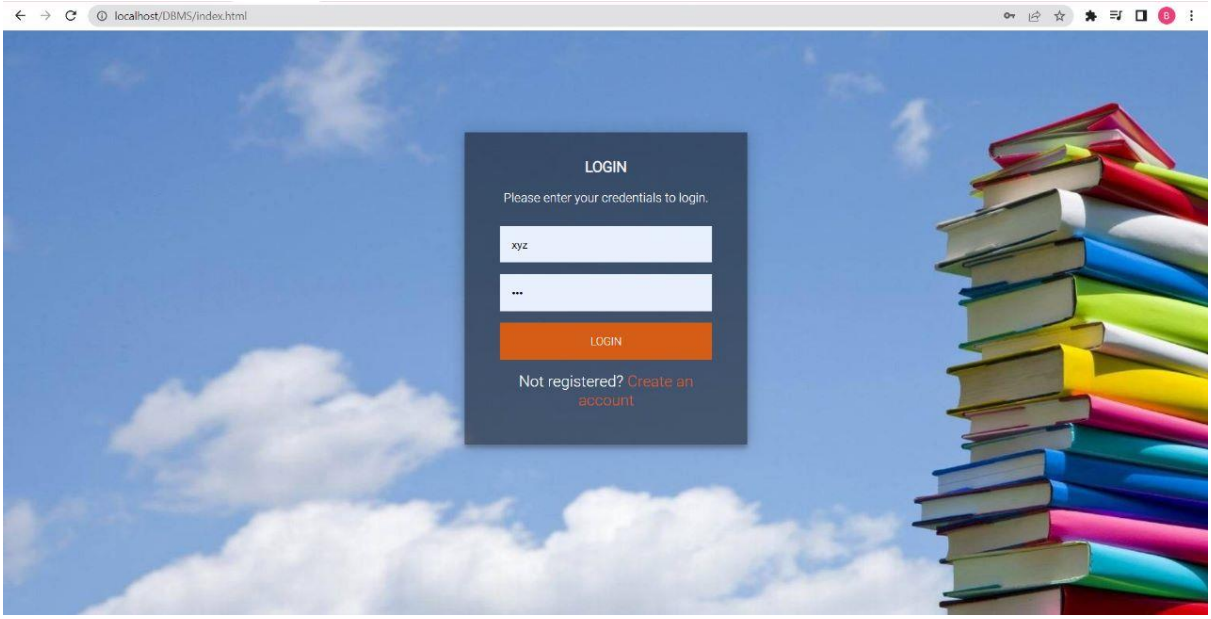
Home page(Redirecting to Login and Register):



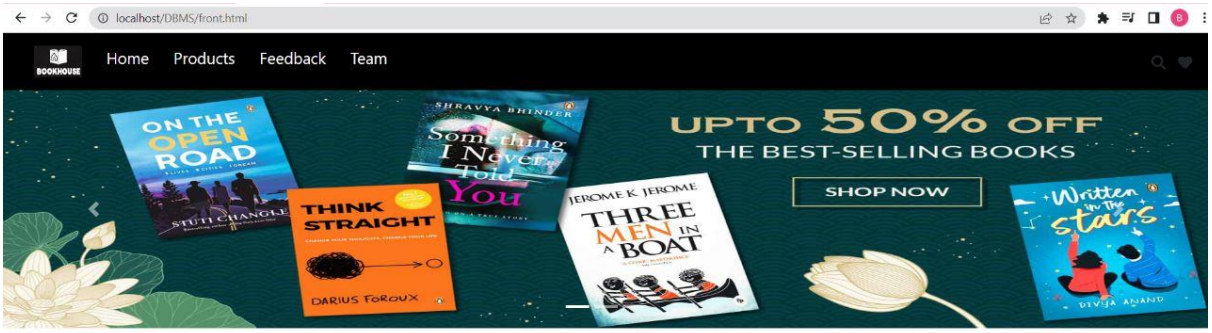
UnAuthorized user Can be Registered through this page:

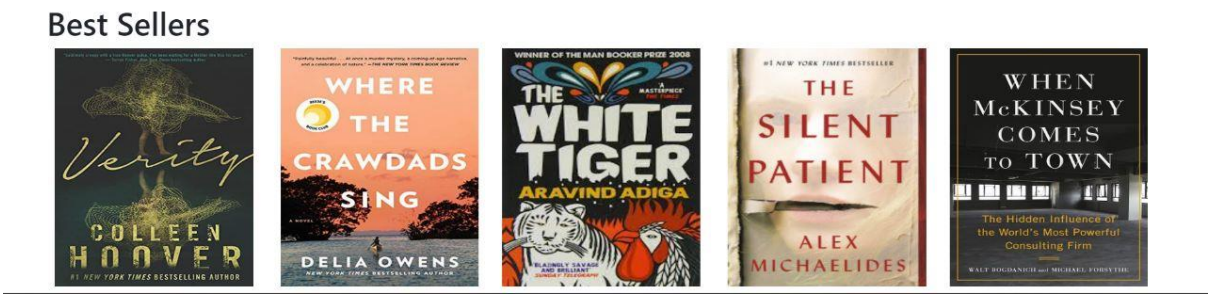


Already Registerd Users can login to the website in this page:

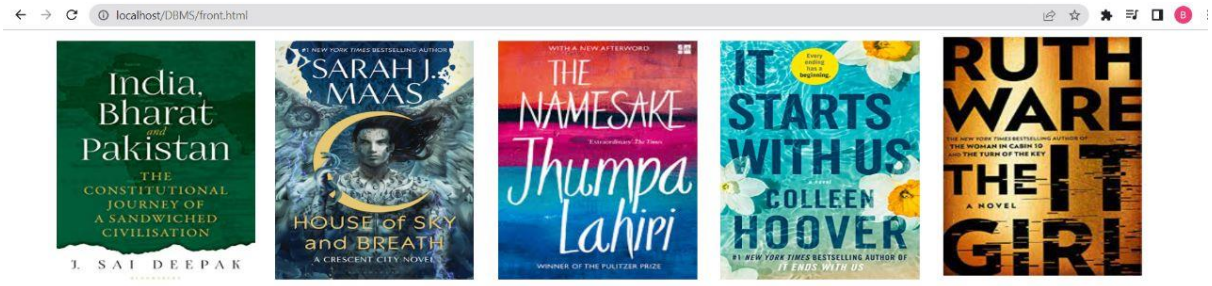


Introduction Page of the website:

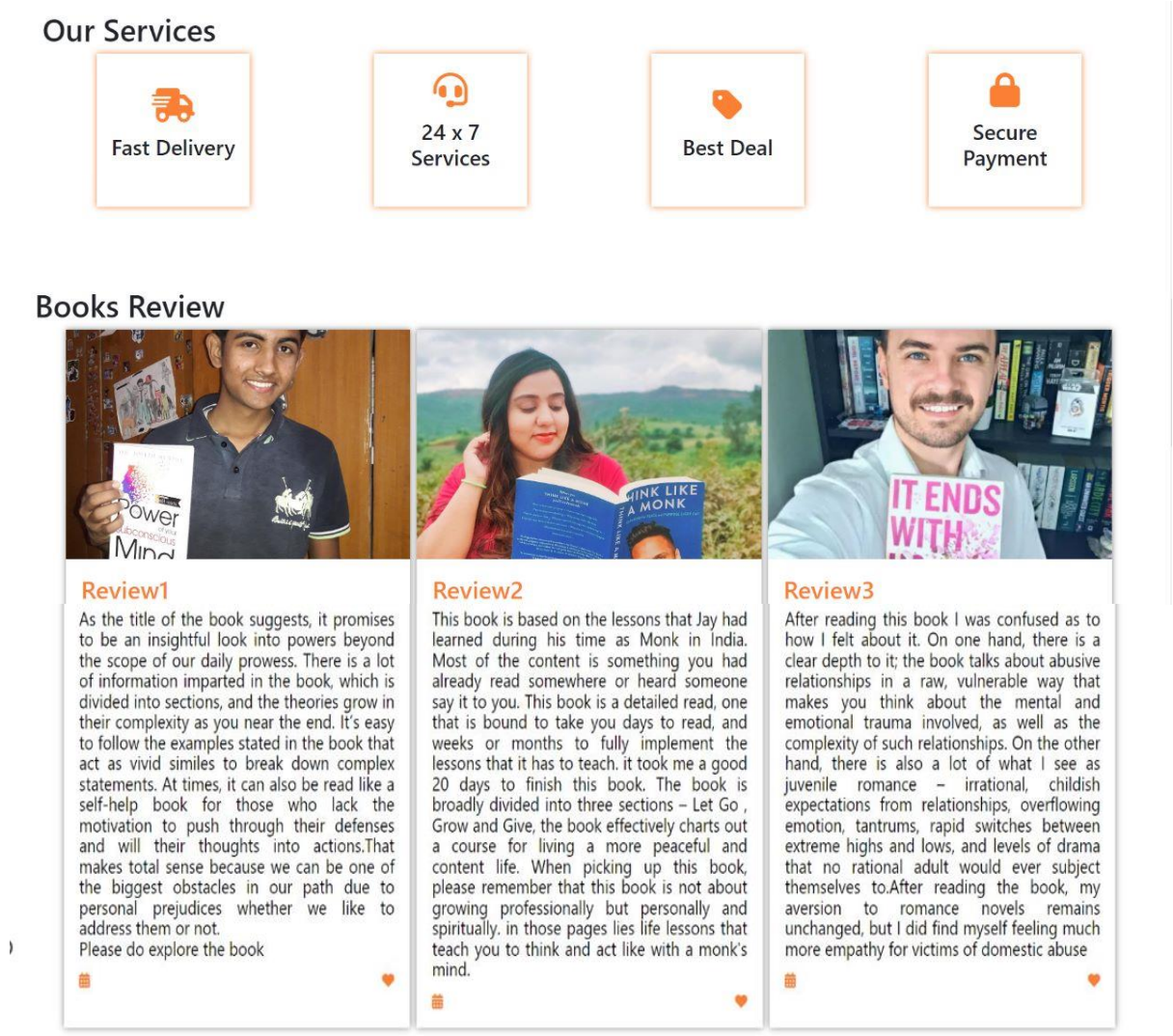




Featured Authors:

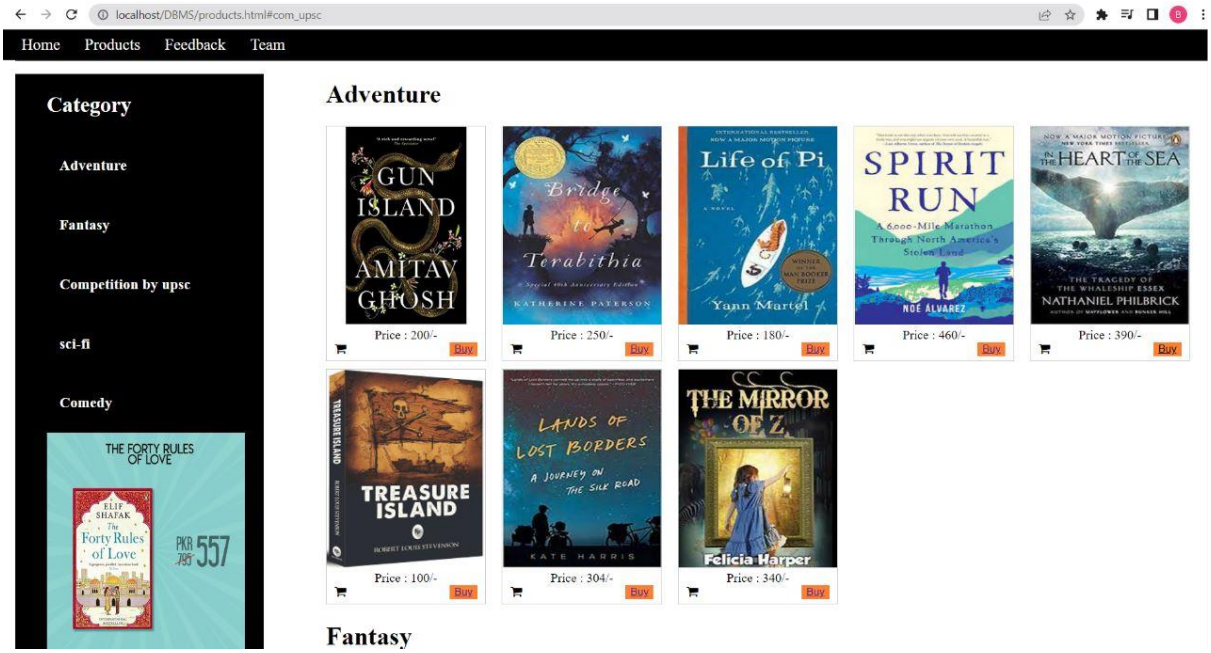


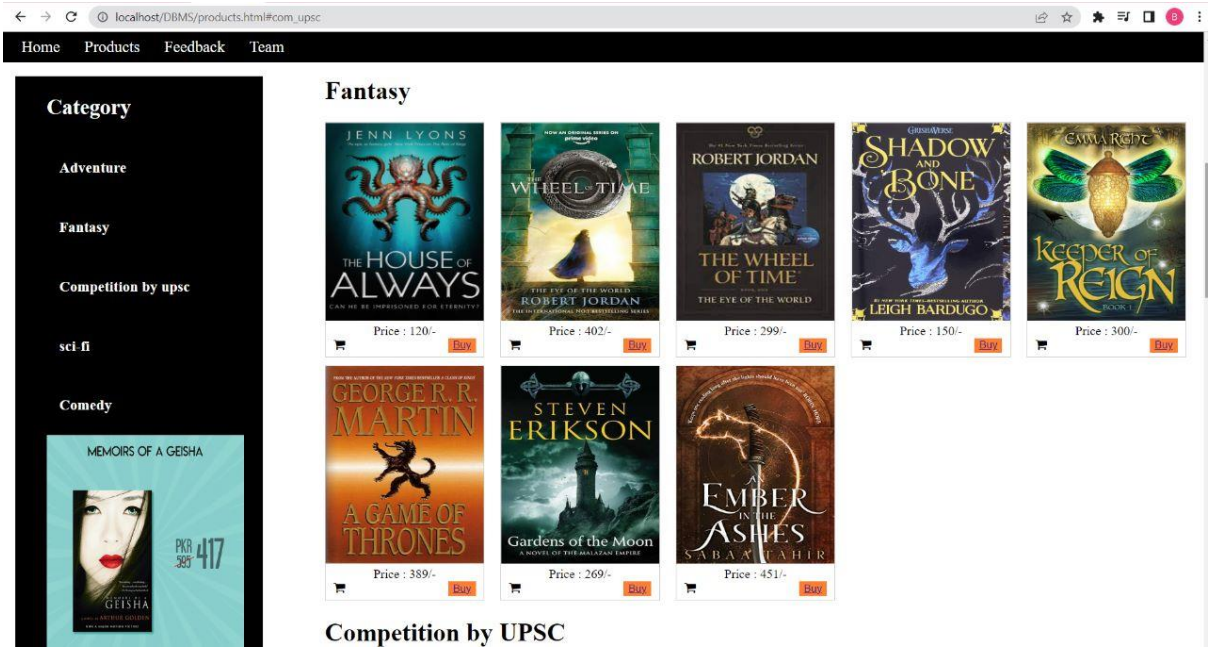
Website Services:



Footer Page:

Products Page Showing the books of various categories:


localhost/DBMS/products.html#com_upsc
Home
Products
Feedback
Team
Category
Adventure
Fantasy
Competition by upsc
sci-fi
Comedy
THE FORTY RULES OF LOVE
PKR 795 557
Adventure
GUN ISLAND AMITAV GHOSH
Price : 200/-
Buy
Bridge to Terabithia
KATHERINE PATERSON
Price : 250/-
Buy
Life of Pi
Yann Martel
Price : 180/-
Buy
SPIRIT RUN
Price : 460/-
Buy
IN THE HEART OF THE SEA
NATHANIEL PHILBRICK
Price : 390/-
Buy
TREASURE ISLAND
Price : 100/-
Buy
LANDS OF LOST BORDERS
A JOURNEY ON THE SILK ROAD
KATE HARRIS
Price : 304/-
Buy
THE MIRROR OF Z
Felicia Harper
Price : 340/-
Buy
Fantasy


localhost/DBMS/products.html#com_upsc
Home
Products
Feedback
Team
Category
Adventure
Fantasy
Competition by upsc
sci-fi
Comedy
MEMOIRS OF A GEISHA
PKR 595 417
Fantasy
JENN LYONS
THE HOUSE OF ALWAYS
Price : 120/-
Buy
THE WHEEL OF TIME
ROBERT JORDAN
Price : 402/-
Buy
ROBERT JORDAN
THE WHEEL OF TIME
THE EYE OF THE WORLD
Price : 299/-
Buy
SHADOW AND BONE
LEIGH BARDUGO
Price : 150/-
Buy
EMMA RIGHT
KEEPER OF REIGN
Price : 300/-
Buy
GEORGE R. R. MARTIN
A GAME OF THRONES
Price : 389/-
Buy
STEVEN ERIKSON
Gardens of the Moon
Price : 269/-
Buy
AN EMBER IN THE ASHES
SABAA TAHIR
Price : 451/-
Buy
Competition by UPSC

localhost/DBMS/products.html#com_upsc

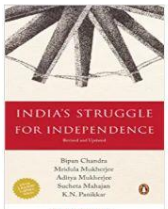
Home Products Feedback Team

Category

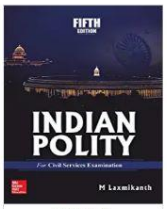
- Adventure
- Fantasy
- Competition by upsc
- sci-fi
- Comedy

Competition by UPSC

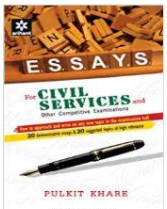
Price : 450/- Buy



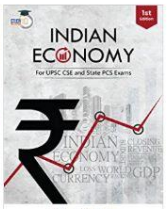
Price : 275/- Buy



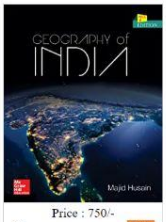
Price : 759/- Buy



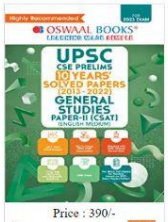
Price : 116/- Buy



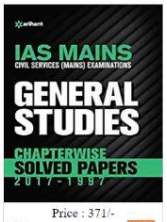
Price : 359/- Buy



Price : 750/- Buy



Price : 390/- Buy



Price : 371/- Buy

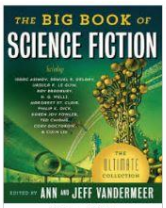
Sci-fi

Home Products Feedback Team

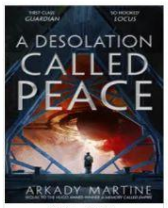
Category

- Adventure
- Fantasy
- Competition by upsc
- sci-fi
- Comedy

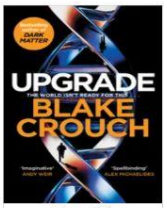
Sci-fi



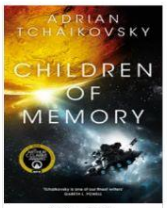
Price : 250/- Buy



Price : 300/- Buy

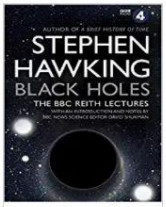


Price : 450/- Buy

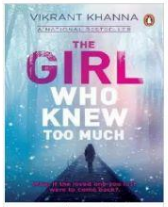


Price : 299/- Buy

Price : 105/- Buy



Price : 199/- Buy



Price : 299/- Buy

Price : 349/- Buy

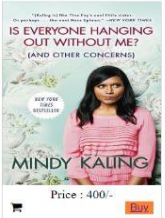
Comedy

Home Products Feedback Team

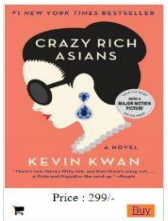
Category

- Adventure
- Fantasy
- Competition by upsc
- sci-fi
- Comedy

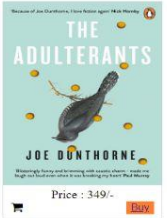
Comedy



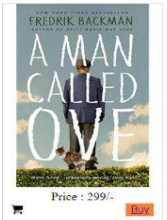
Price : 400/- Buy



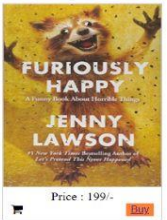
Price : 299/- Buy



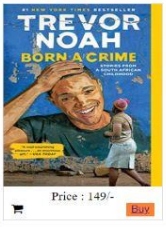
Price : 349/- Buy



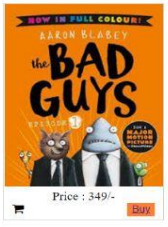
Price : 299/- Buy



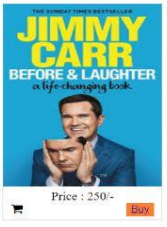
Price : 199/- Buy



Price : 149/- Buy



Price : 349/- Buy

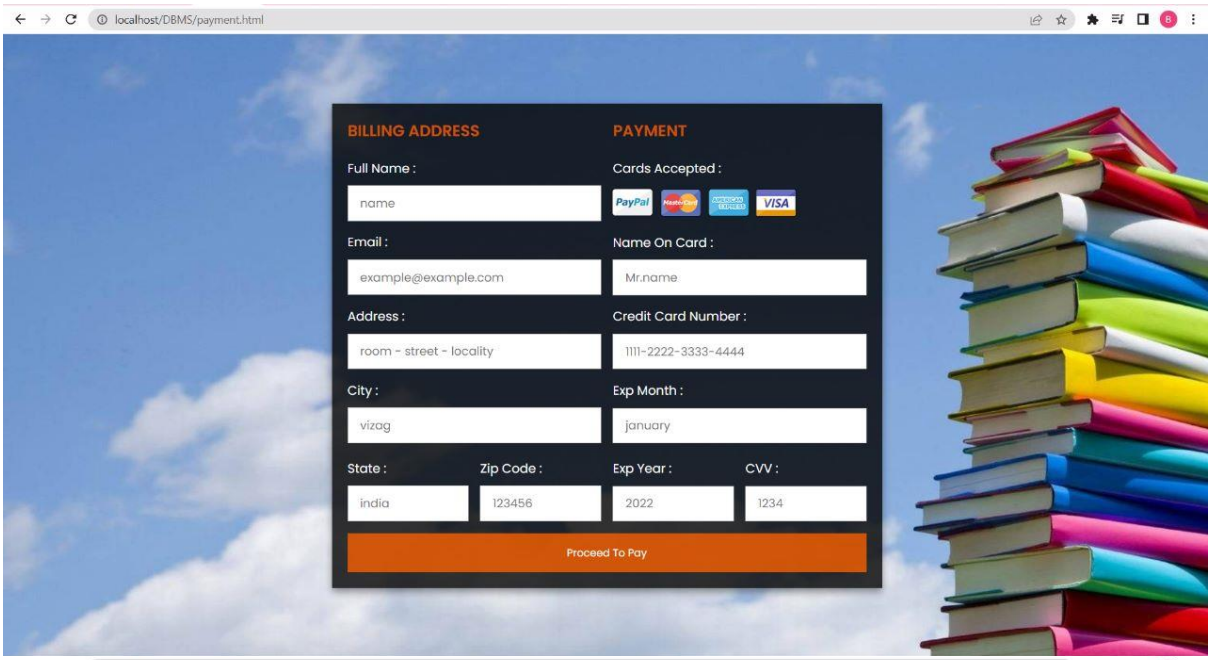


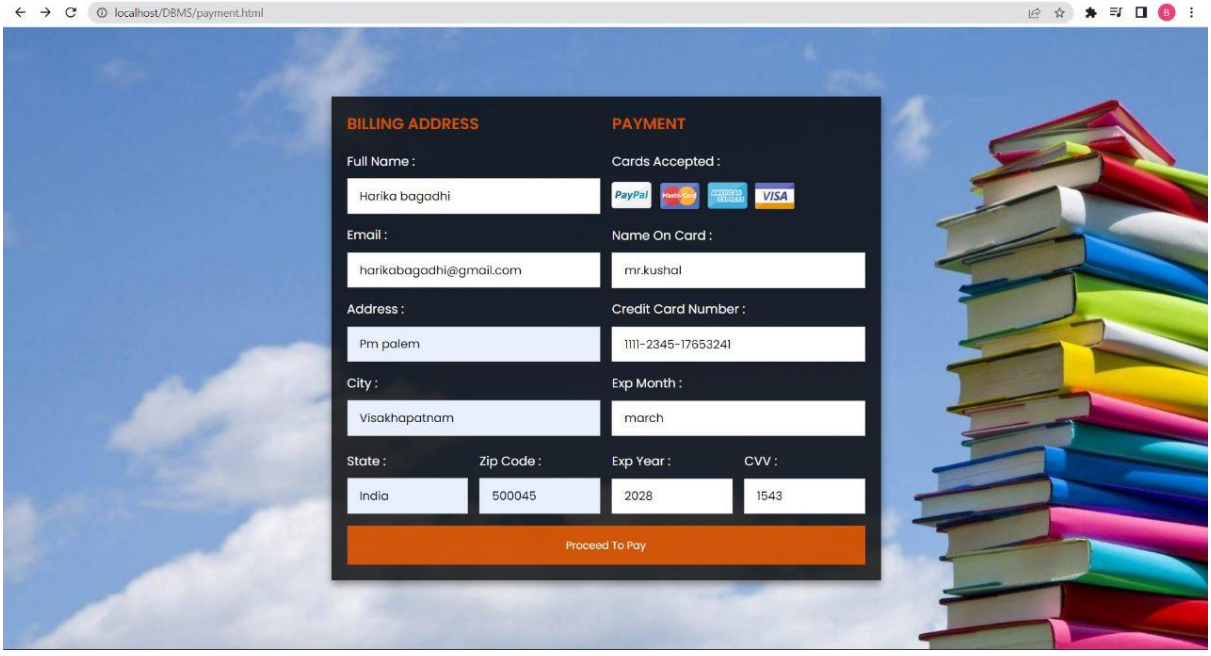
Price : 250/- Buy

Back

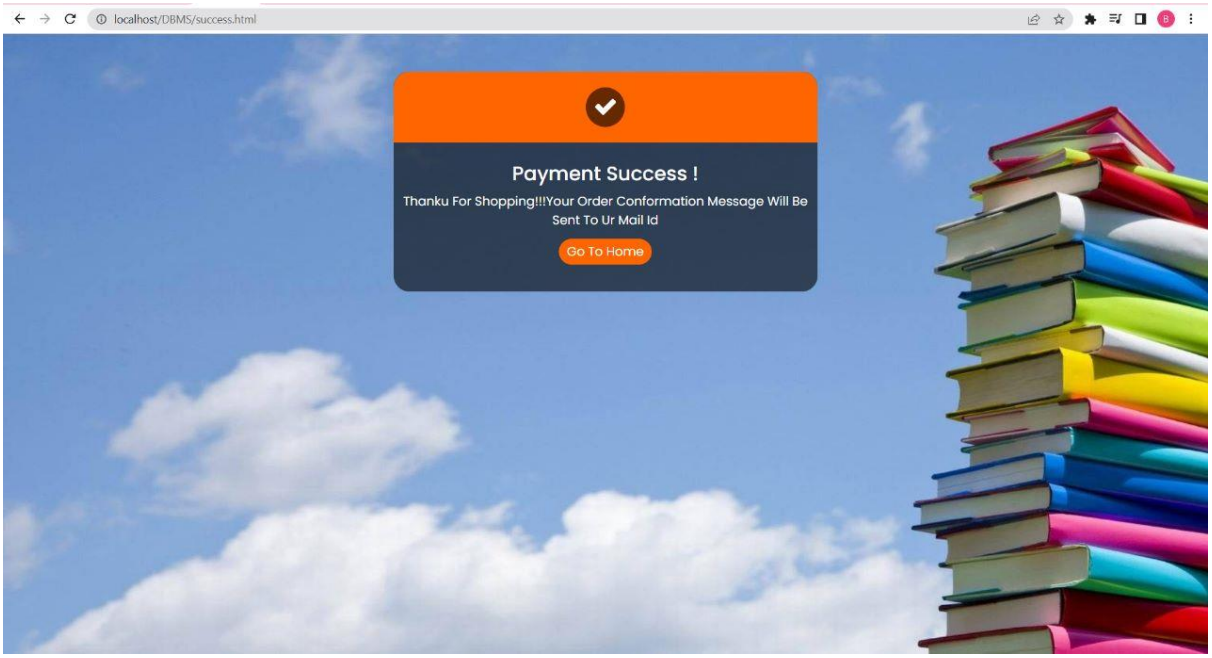
Payment Page :

Allows the user to process the payment of required book

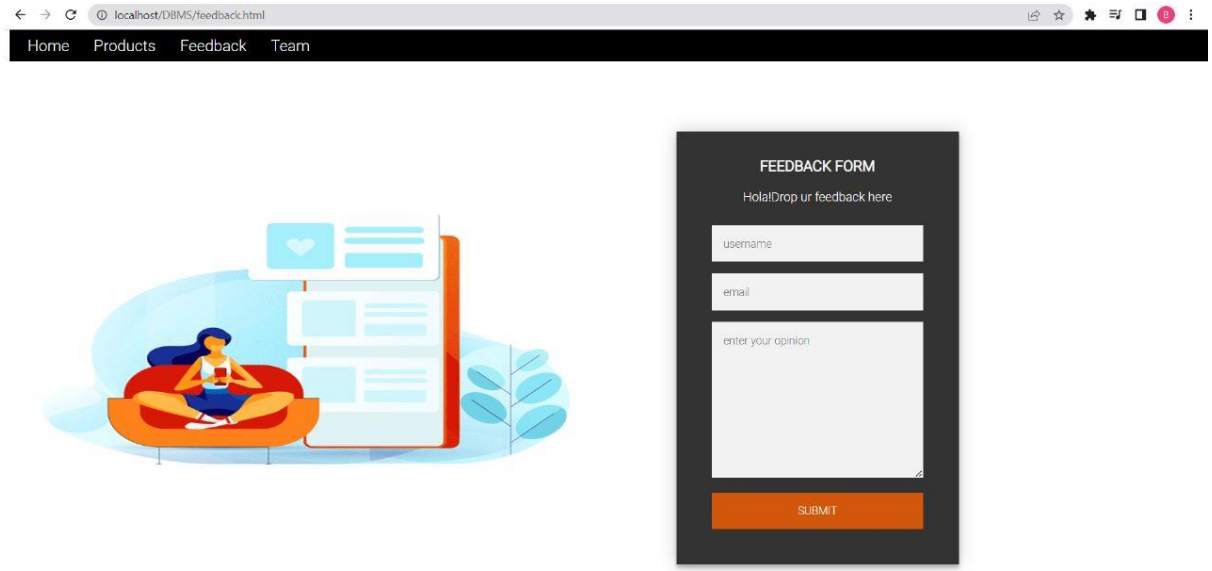




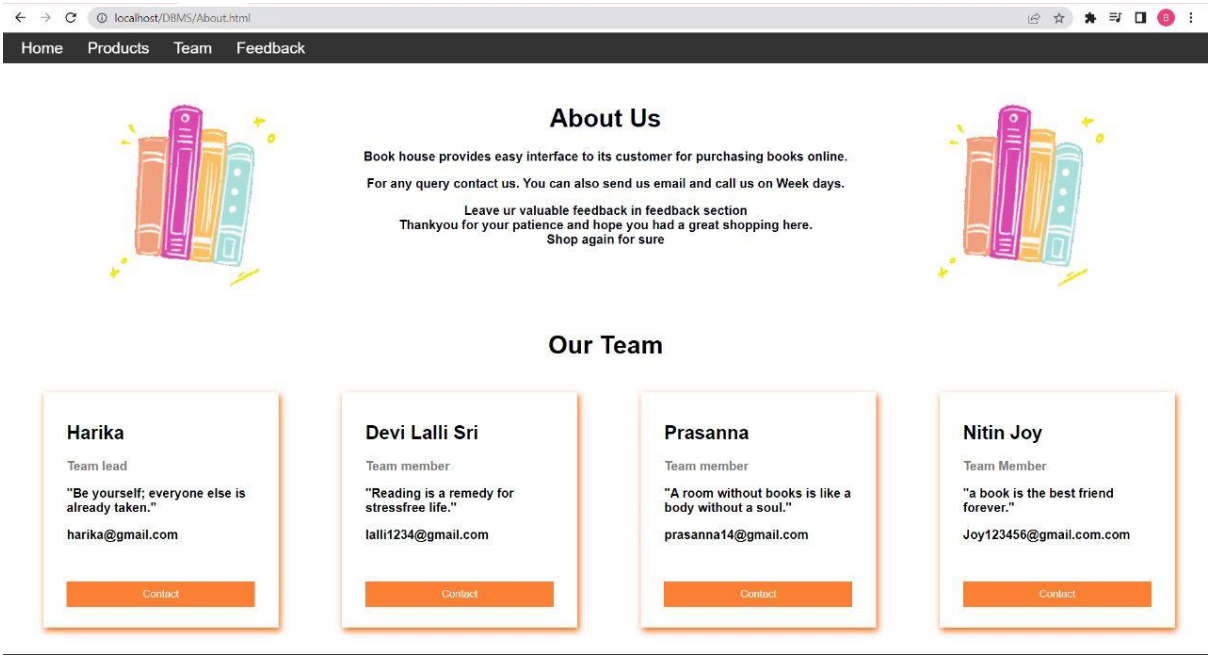
On Succesful Payment the customer gets redirected to this page:



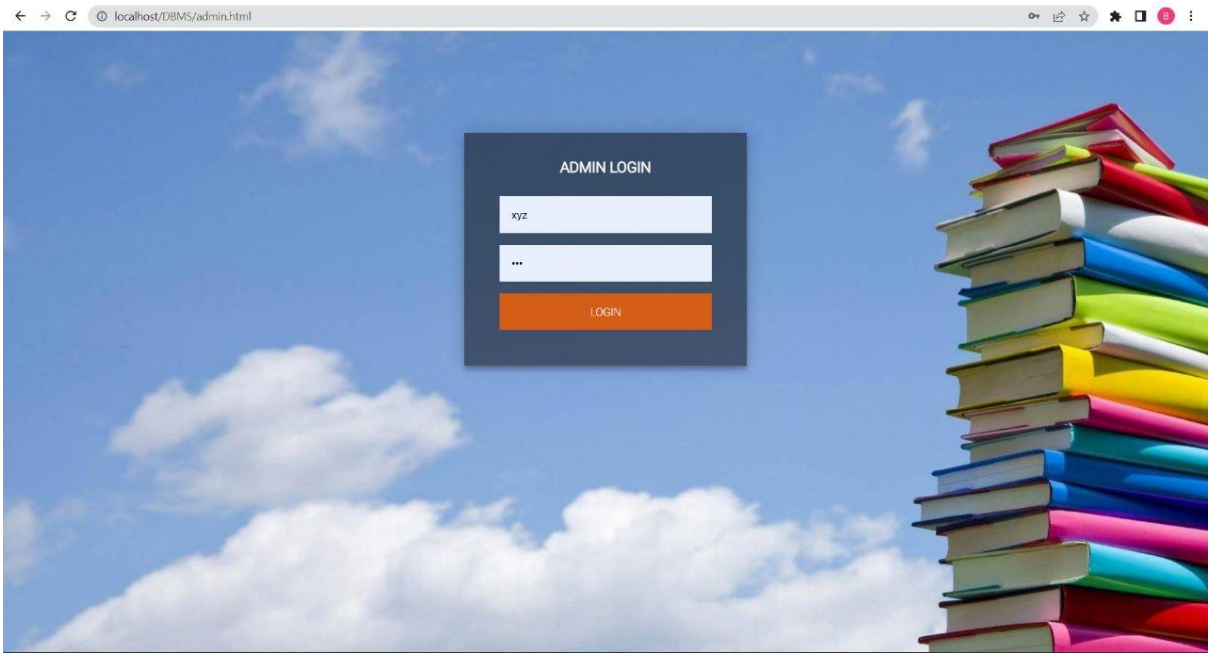
The Customer can give their valuable FeedBack in this page:



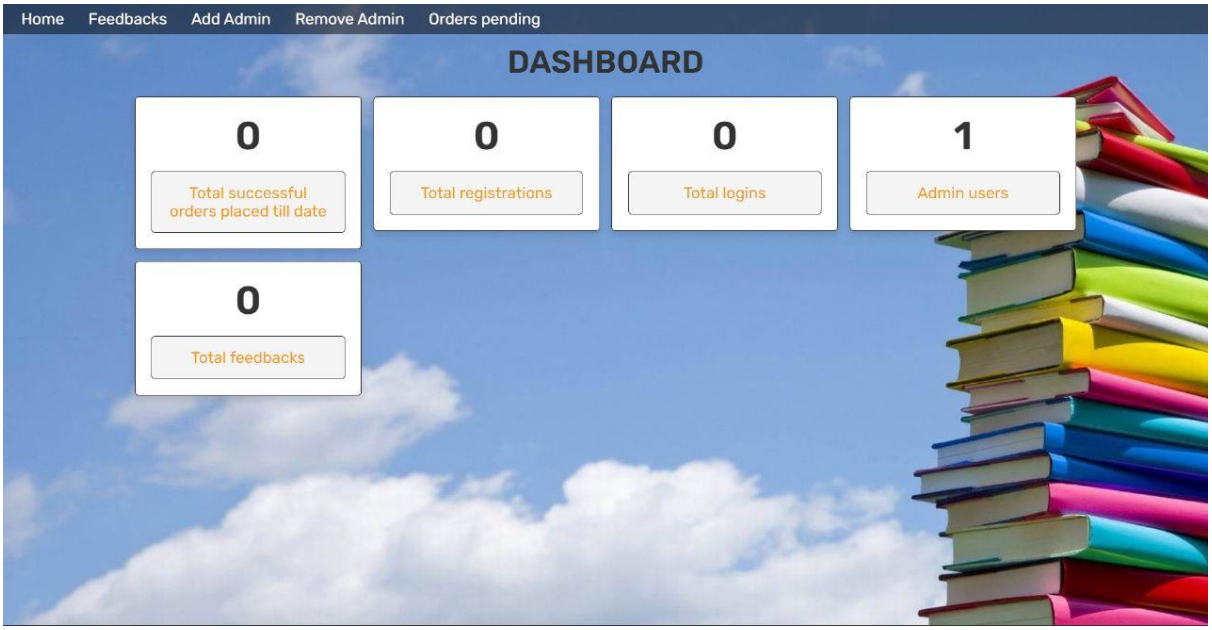
This page gives the description about the website and details of website Creators



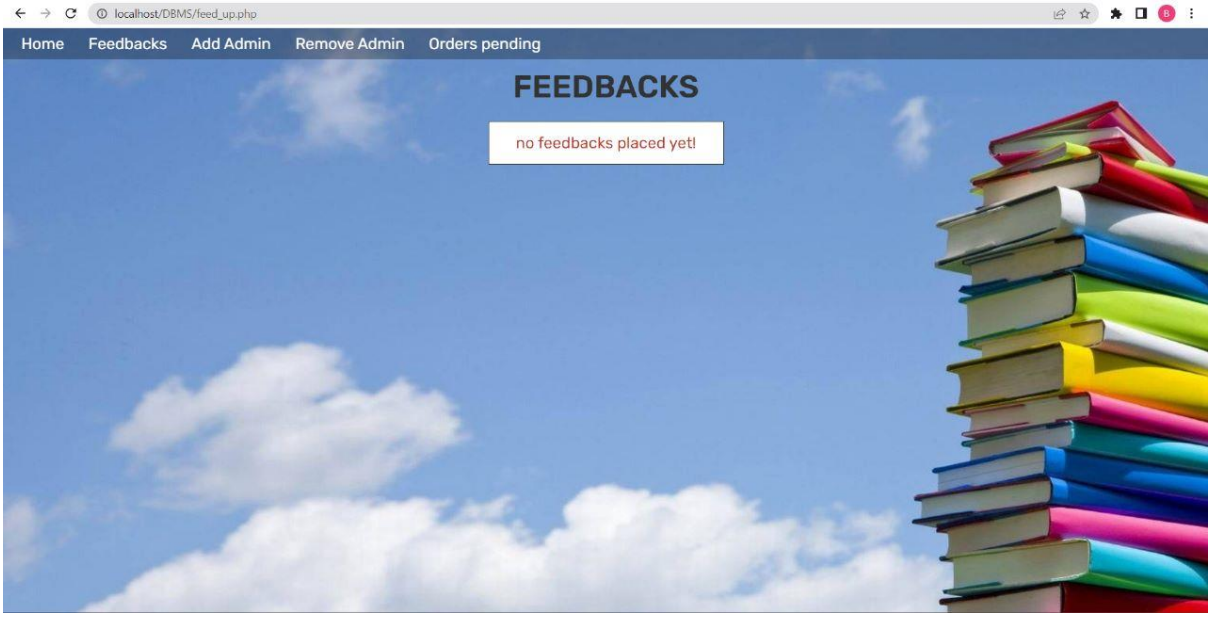
This page redirects the admin to the admin interface



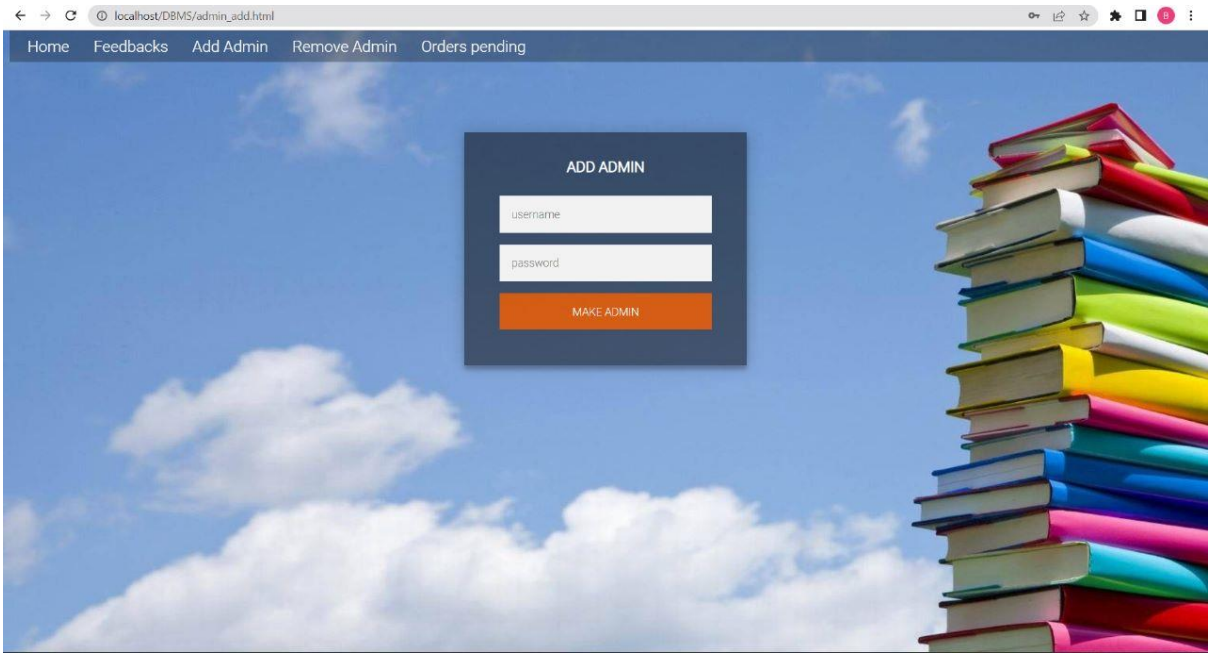
Gives the admin the insight of website functionalities



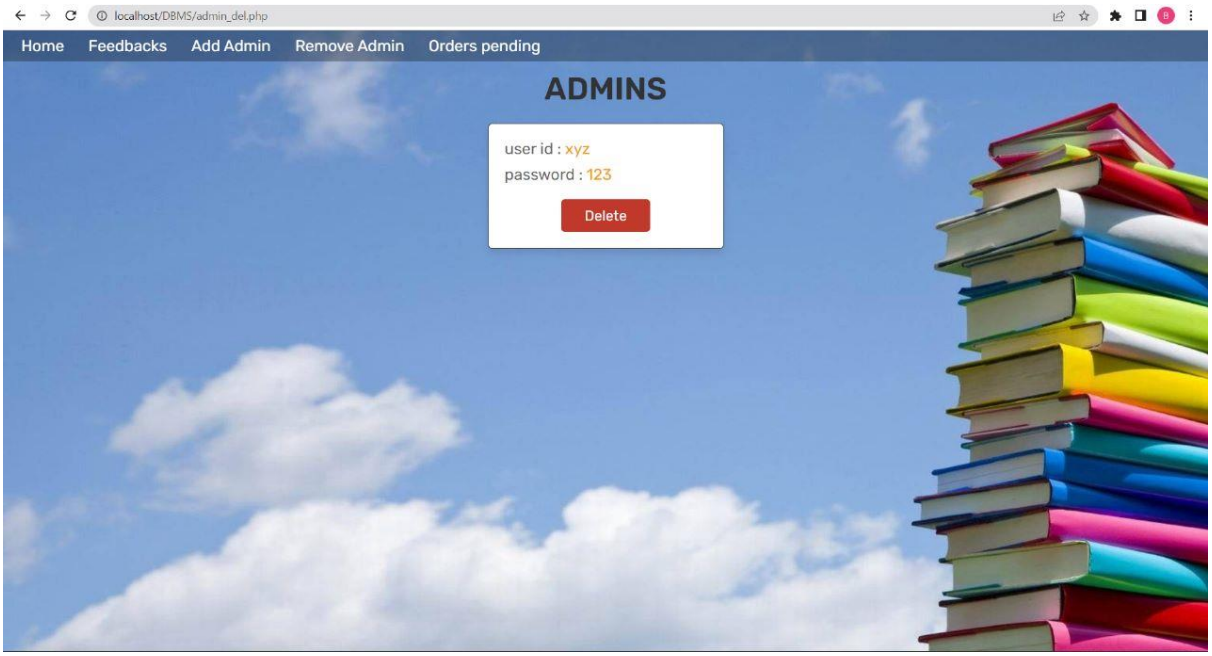
Customer Feedbacks can be viewed, deleted by the admin in this page



New Admin are given access through this page:



Admins are viewed and their access is revoked in this page:



Order details and their confirmations are ddone through this page:

CONCLUSION AND FUTURE SCOPE

CONCLUSION:

It also has features like viewing the description of the book such as book_name,author_name,publication details.it includes user feedback so that the website can be modified as per the willingness of the customer.

finally we can conclude that book house is a good alternative to the traditional way of buying books also the customer can buy the book from the home itself without visiting book store.

FUTURE SCOPE:

At present the website provides minimal functionalities such as buying books of various categories through easy way of payment for the website can be added with features like adding the favourite books to the wishlist and modifying the user interface such that the website can be designed as a one to one interfacing providing facilites like add to cart and recommending the books to the user as per their taste and choice.the admin interface can be structured such that new arrivals gets added directly to the product page of the website and viewing the monthly sales report of the orders.

REFERENCES:

Web Sites:

1) https://www.w3schools.com/TAGS/default.asp
2) https://en.wikipedia.org/wiki/CSS
3) https://getbootstrap.com/docs/4.1/getting-started/introduction/
4) https://www.tutorialspoint.com/php/php_and_mysql.htm
5) https://www.youtube.com/watch?v=qxZthRY_cN8

Textbooks:

1. Steven Holzner, “HTML Black Book: The Programmer’s Complete HTML Reference Book”

2. Robin Nixon, “Learning PHP, MySQL, and JavaScript”, 4th edition

GitHub Link:

https://github.com/harika-320126552002/DBMS_PROJECT