VISVESVARAYA TECHNOLOGICAL UNIVERSITY

JNANASANGAMA , BELAGAVI -590018

DATABASE MANAGEMENT SYSTEM MINI PROJECT

(18CSL58)

ON

“BOOK DRIVE”

Submitted in partial fulfillment of the requirements for the 5th Semester

INFORMATION SCIENCE AND ENGINEERING

Submitted by

HITESH N (1BI19IS021)

KUSHAL A C (1BI19IS028)

Under the guidance of

Dr. Hema Jagadish
Associate Professor
Dept of ISE,BIT

M.Shilpa
Assistant Professor
Dept of ISE,BIT

2021-2022

DEPARTMENT OF INFORMATION SCIENCE AND ENGINEERING
BANGALORE INSTITUTE OF TECHNOLOGY
K. R. Road, V. V. Pura, Bengaluru-560004

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

1. INTRODUCTION

Database is a collection of data and Management System is a set of programs to store and retrieve those data.

1.1 General Overview:

A 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 loosely to refer to any of the DBMS, the database system or an application associated with the database.

1.1.1 What is the need of DBMS?

Database systems are basically developed for large amounts of data. When dealing with huge amounts of data, there are two things that require optimization: Storage of data and retrieval of data.

Storage: According to the principles of database systems, the data is stored in such a way that it acquires a lot less space as the redundant data (duplicate data) has been removed before storage. Let’s take a layman example to understand this. In a banking system, suppose a customer has two accounts, one is a savings account and another is a salary account. Let’s say bank stores saving account data at one place and salary account data at another place, in that case if the customer information such as customer name, address etc. are stored at both places then this is just a wastage of storage (redundancy/ duplication of data), to organize the data in a better way the information should be stored at one place and both the accounts should be linked to that information somehow. The same thing we achieve in DBMS.

Fast Retrieval of data: Along with storing the data in an optimized and systematic manner, it is also important to retrieve the data quickly when needed. Database systems ensure that the data is retrieved as quickly as possible.

The choice of a database product is often influenced by factors such as:

1. The computing platform (i.e., hardware, operating system)
2. The volume of data to be managed.

3. The number of transactions required per second.
4. Existing applications or interfaces that an organization may have support for heterogeneous and/or distributed computing.
5. Cost

1.1.2 Design and Modeling:

The first task of a database designer is to produce a conceptual data model that reflects the structure of the information to be held in the database. A common approach to this is to develop an entity-relationship model, often with the aid of drawing tools. Another popular approach is the Unified Modeling Language. A successful data model will accurately reflect the possible state of the external world being modeled: for example, if people can have more than one phone number, it will allow this information to be captured.

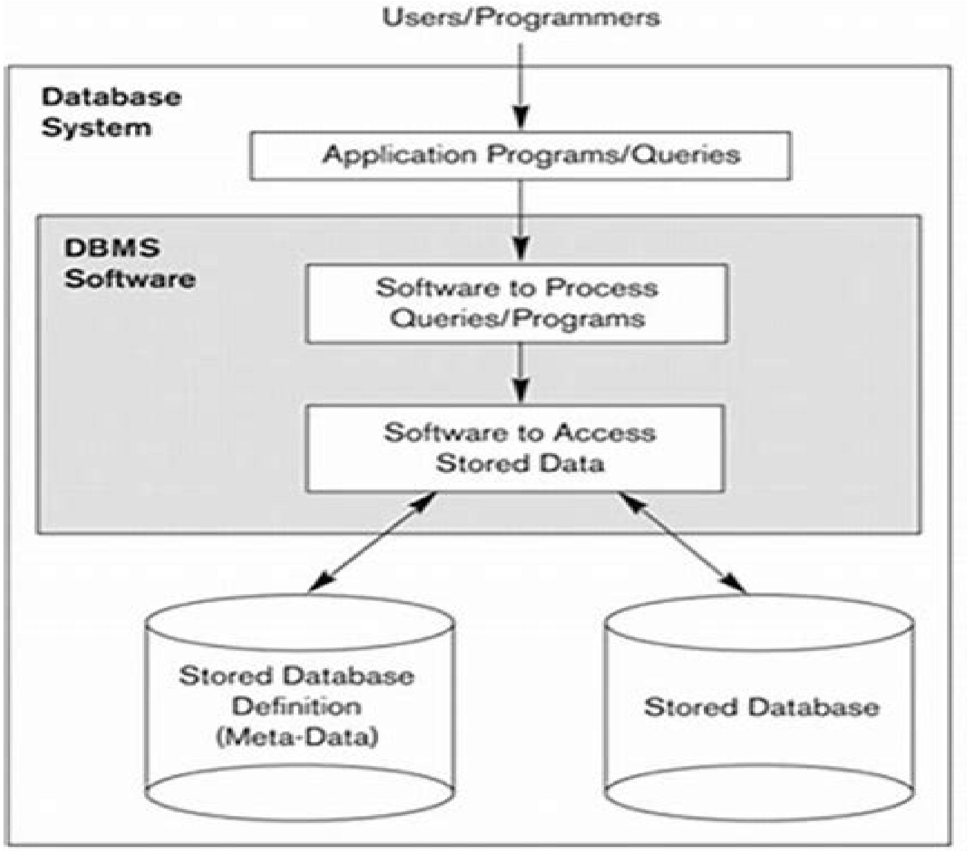

Users/Programmers
Database System
Application Programs/Queries
DBMS Software
Software to Process Queries/Programs
Software to Access Stored Data
Stored Database Definition (Meta-Data)
Stored Database

Figure 1.1 : Simplified Database System Environment

1.2 Problem Statement

- The problem occurred before having a computerized system includes:
- File lost When a computerized system is not implemented file is always lost because of the human environment.Some times due to some human error there may be a loss of records.
- File damaged When a computerized system is not there file is always lost due to some accident like spilling of water by some member on file accidentally. Besides, some natural disasters like floods or fires may also damage the files.
- Difficult to search records
- When there is no computerized system there is always a difficulty in searching records if the records are large in number .
- After the number of records becomes large the space for physical storage of file and records also increases if no computerized system is implemented.
- Cost consuming As there is no computerized system the to add each record paper will be needed which will increase the cost for the management of library.

1.3 OBJECTIVE

1. Provides the searching facilities based on various factors such as books, author , categories and description .
2. It shows the information and description of books and their authors .
3. To increase the efficiency of managing books .
4. Editing , adding and updating of records is improved which results in proper resource management of books data .
5. Bookdrive will allow various users to download or preview various books depending upon the category .
6. Bookdrive will also help in generating the report on various books uploaded .
7. The font type, size and margins can be customized.

2. SOFTWARE AND HARDWARE REQUIREMENT

The program works on Desktop PC and is executed using a Python flask which interacts with a MySQL database running on localhost.

2.1 Software Requirements:

- ➢ Operating System: Windows 7/8/10/11, LINUX
- ➢ Back End : MySQL, PHP
- ➢ Front End : HTML, CSS ,Bootstrap.
- ➢ Application required: Xampp, VisualStudio.

2.2 Hardware Requirements:

- ➢ Hard disk – 10 GB or Above
- ➢ RAM – minimum 1 GB or Above
- ➢ Processor – Intel core i4/i5/i7/i9

3. DESIGN

Entity Relationship Model (ER Modeling) is a graphical approach to database design. It is a high-level data model that defines data elements and their relationship for a specified software system.

3.1 ER Analysis: Identifying Entity Sets and Relationship Sets:

It is a high-level data model that defines data elements and their relationship for a specified software system.An ER model is used to represent real-world objects.

3.1.1 Entities and Attributes:

An **Entity** is a thing or object in the real world that is distinguishable from the surrounding environment. An Entity is an object of Entity Type and the set of all entities is called an Entity **set**. **Attributes** are the properties which define the entity type.

This section of the document explains the entities used in the project , their attributes and how they will work together. Basically, this is intended to make the design more easy and understandable for everyone.

ENTITIES

1. Admin
2. Authors
3. Book
4. Category

1. Admin

Attributes of Admin are id, full_name , email , password.

Table 3.1:Admin Attributes

Attributes	Data Type	Description
id	INT(11)	Primary key
full_name	VARCHAR(255)	Non key attribute
email	VARCHAR(255)	Non key attribute
password	TEXT	Non key attribute

2. Authors

Attributes of Authors are author_id , name

Table 3.2 : Authors Attributes

Attributes	Data Type	Description
author_id	INT(11)	Primary key
name	VARCHAR(255)	Non key attribute

3. Books

Attributes of Books are id, title, author_id, description, category_id, cover, file

Table 3.3 : Books Attributes

Attributes	Data Type	Description
id	INT(11)	Primary key
title	VARCHAR(255)	Non key attribute
author_id	INT(11)	Foreign key
description	TEXT	Non key attribute
category_id	INT(11)	Foreign key
cover	VARCHAR(255)	Non key attribute
file	VARCHAR(255)	Non key attribute

4. Category

Attributes of category are category_id ,name

Table 3.4 : Category Attributes

Attributes	Data Type	Description
category_id	INT(11)	Primary key
name	VARCHAR(255)	Non key attribute

3.1.2 Relationship Sets:

A relationship type represents the **association between entity types**. A set of relationships of the same type is known as a relationship set.

The following are the relationships between the various entities used in the project.

1. **author_id** : It is used to identify various authors and assign unique id.
2. **Id** : It identifies different books and assigns book id.
3. **Category_id** : It is used to identify various categories of books and it assigns unique id to each category.
4. **Password** : Provides authentication for each user to access books from the server .
5. **Description** : Provides necessary information on each book.

3.2 Entity Relationship Diagram:

An entity relationship model, also called an entity-relationship (ER) diagram, is a graphical representation of entities and their relationships to each other, typically used in computing in regard to the organization of data within databases or information systems

3.2.1 ER Diagram with relationships and cardinality ratio

The cardinality or fundamental principle of one data aspect with respect to another is a critical feature. The relationship of one to the other must be precise and exact between each other in order to explain how each aspect links together. In simple words Cardinality is a way to define the relationship between two entities. The following are the notations of the ER diagram:

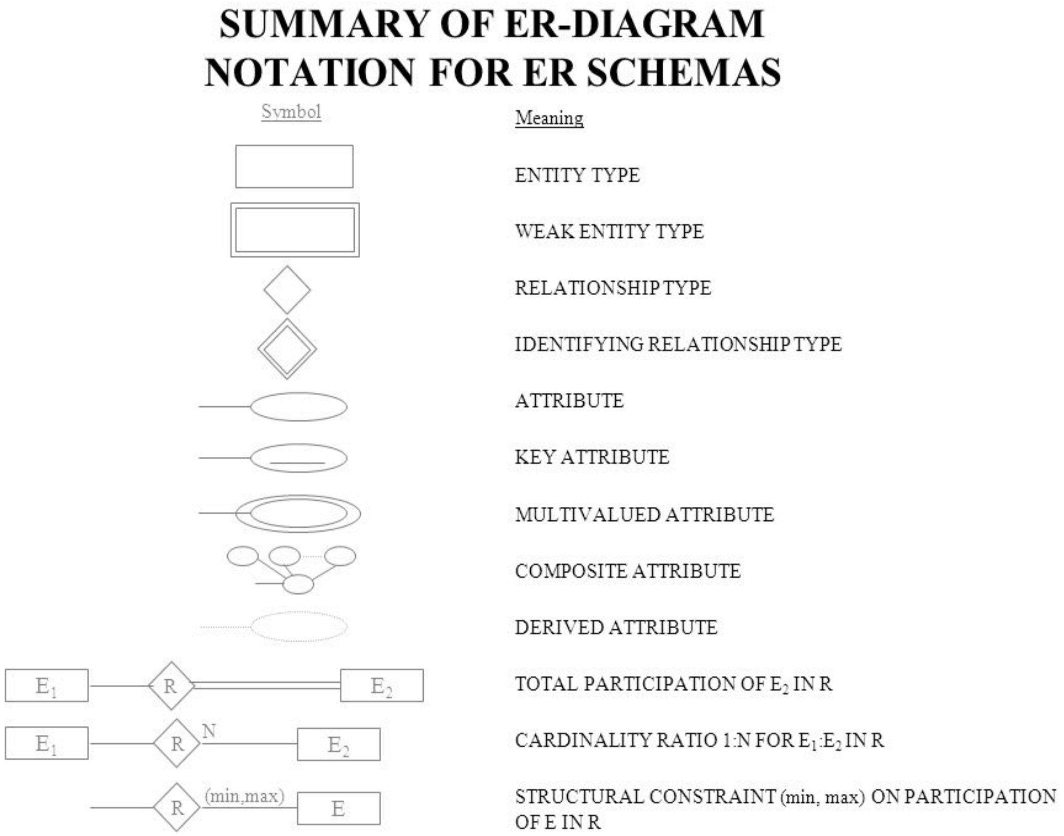


Fig 3.1 : Notations for ER Diagrams

The ER diagram below shows the relationship between the many tables that exist in the database for the functioning of **Book Drive**

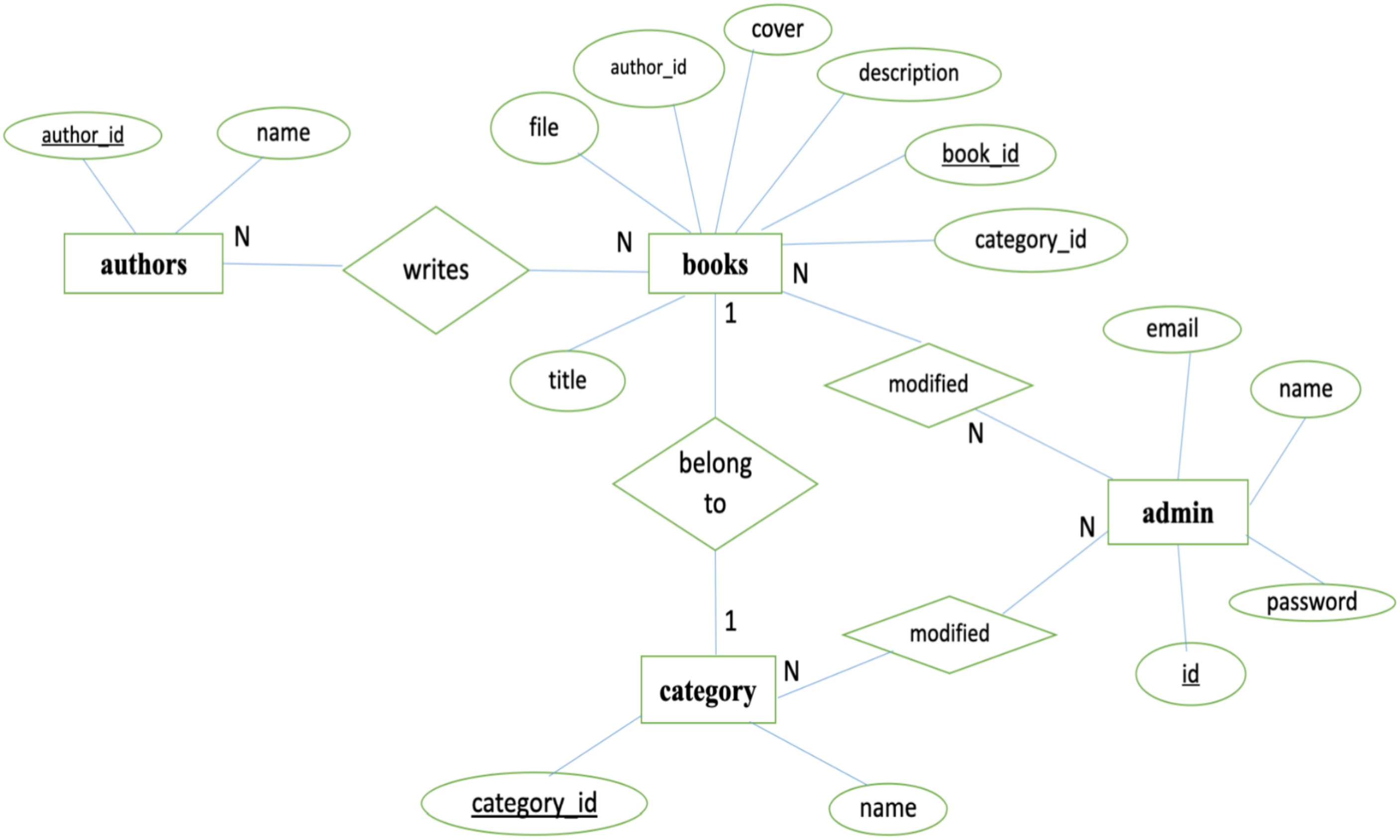


Fig 3.2 : ER Diagram of Book Drive

3.2.2 Entity Sets

In DBMS, an entity set is a set of entities of the same type. An entity set may be of two types- Strong Entity Set and Weak Entity Set. An entity refers to any object having either a physical existence or a conceptual existence.

1. Admin

Admin is the main entity of the project who has access to the database of the project .Each user has its own identity such as id , full_name, email.

Attributes : Admin consists of attributes such as id , full_name, email,password.

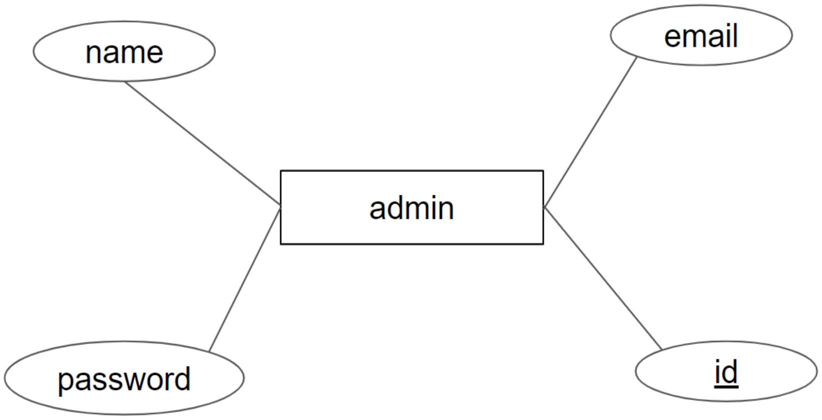


Fig 3.3: Entity Diagram for Admin

2. Authors

Author is the one who writes the books .Each author has its own identity such as author_id , name

Attributes : Authors consist of attributes such as author_id , name.

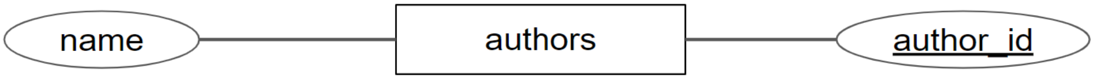


Fig 3.4: Entity Diagram for Authors

3.Books

Books maintains the information on books such as title, category, author name of books

Attributes : Books consists of Attributes such as id, title, author_id, description, category_id, cover,file

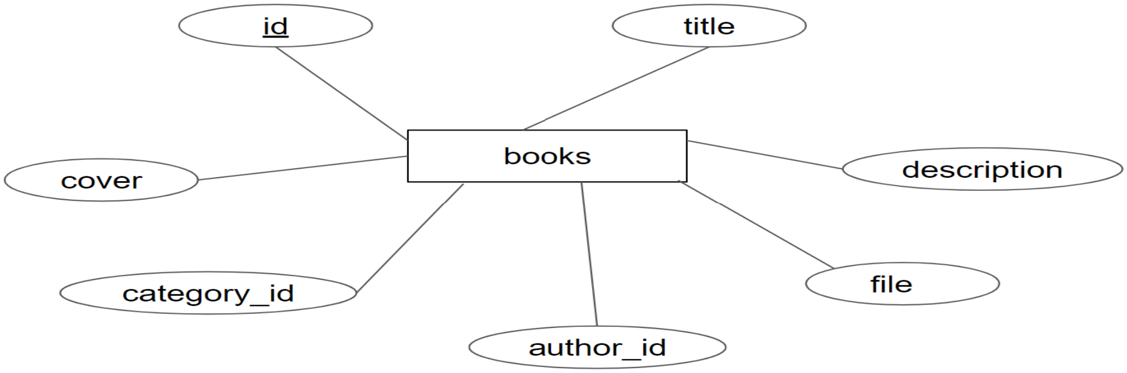


Fig 3.5: Entity Diagram for Books

4. Category

Category stores the information about various categories

Attributes : Category consists of Attributes such as category_id, name

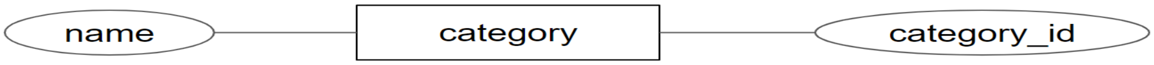


Fig 3.6: Entity Diagram for Category

3.3 Schema Diagram

In any data model it is important to distinguish between the description of the database and the database itself. The description of a database is called the database schema, which is specified during database design and is not expected to change frequently. A displayed schema is called a schema diagram. A schema diagram displays only some aspects of a schema, such as the names of record types and data items, and some types of constraints.

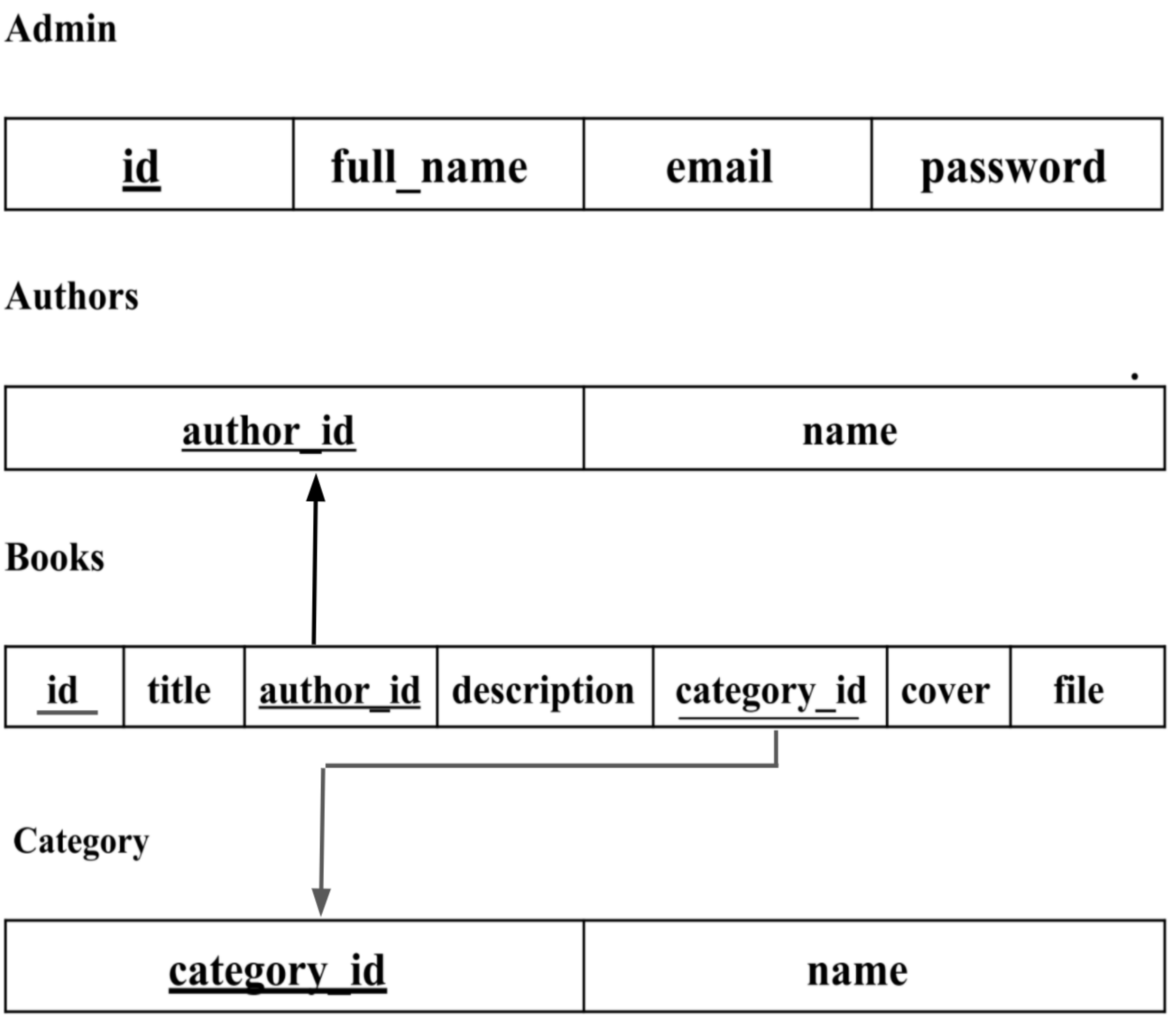


Fig 3.7 Schema Diagram for Book Drive

3.4 Tables and their Functional Dependencies

Functional Dependency (FD) is a constraint that determines the relation of one attribute to another attribute in a Database Management System (DBMS). Functional Dependency helps to maintain the quality of data in the database. It plays a vital role to find the difference between good and bad database design.

A functional dependency is denoted by an arrow "→". The functional dependency of X on Y is represented by X → Y.

1) Admin(id , full_name, email,password)
   FD={id→full_name, email,password}

2) Authors(author_id , name)
   FD={author_id →name}

3) Books(id, title, author_id, description, category_id, cover,file)
   FD={id →title, author_id, description, category_id, cover,file}
   {author_id, category_id} is foreign key constraints

4) Category(category_id, name)
   FD={category_id→name}

3.5 Normalization

Normalization is a process of organizing the data in a database to avoid data redundancy, insertion anomaly, update anomaly & deletion anomaly.
Database Normalization is a technique of organizing the data in the database. Normalization is a systematic approach of decomposing tables to eliminate data redundancy(repetition) and undesirable characteristics like Insertion, Update and Deletion Anomalies. It is a multi-step process that puts data into tabular form, removing duplicated data from the relation tables.

Normalization is used for mainly two purposes,

1. Eliminating redundant(useless) data.
2. Ensuring data dependencies make sense i.e data is logically stored.

3.5.1 First Normal form

For a table to be in the First Normal Form, it should follow the following 4 rules:

1. It should only have single(atomic) valued attributes/columns.
2. Values stored in a column should be of the same domain
3. All the columns in a table should have unique names.
4. And the order in which data is stored, does not matter.

   All of the above tables IN this project have 1NF.

3.5.2 Second Normal form

For a table to be in the Second Normal Form,

1. It should be in the First Normal form.
2. And, it should not have Partial Dependency.

All the tables which are part of this project are in 2NF as they have at most one primary key, so no partial dependency.

3.5.3 Third Normal form

A table is said to be in the Third Normal Form when,

1. It is in the Second Normal form.
2. And, it doesn't have Transitive Dependency.

All the used tables satisfy both these conditions and hence are in 3NF.

4. IMPLEMENTATION

The implementation phase is where you install DBMS on the required hardware,optimize the database to run best on that hardware and software platforms,and create the database and load the data.

4.1 Basic Steps in Implementation:

- Only the admins are eligible to append different users' email and password.
- Admins can add various books by providing necessary description of the books unders suitable categories
- Admins can also edit the description by using the “edit” option on the website.
- Every user has an account which gives authentication using password and email and he can access the various books under various categories
- Users logging-in can access all the books under the categories by either previewing or downloading.

4.2 Technologies and Frameworks:

HTML

Hypertext Markup Language is the standard markup language for documents designed to be displayed in a web browser. It can be assisted by technologies such as Cascading Style Sheets and scripting languages such as JavaScript.

CSS

Cascading Style Sheets is a style sheet language used for describing the presentation of a document written in a markup language like HTML. CSS is a cornerstone technology of the World Wide Web, alongside HTML and JavaScript. Functional Modules.

Bootstrap

Bootstrap is a free and open-source CSS framework directed at responsive, mobile-first frontend web development. It contains CSS- and JavaScript-based design templates for typography, forms, buttons, navigation, and other interface components.

PHP

PHP is an open-source, interpreted, and object-oriented scripting language that can be executed at the server-side.PHP is well suited for web development. PHP stands for Hypertext Preprocessor.

XAMPP

It is one of the widely used cross-platform web servers, which helps developers to create and test their programs on a local webserver.

MySQL

MySQL is a relational database management system based on the Structured Query Language, which is the popular language for accessing and managing the records in the database. MySQL is open-source and free software under the GNU license. It is supported by Oracle Company.

4.3 Frontend Code

index

```
<?php
session_start();

# Database Connection File
include "db_conn.php";

# Book helper function
include "php/func-book.php";
$books = get_all_books($conn);

# author helper function
include "php/func-author.php";
$authors = get_all_author($conn);

# Category helper function
include "php/func-category.php";
$categories = get_all_categories($conn);

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

```
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Book Store</title>

 <!-- bootstrap 5 CDN-->
  <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.1/dist/css/bootstrap.min.css" rel="stylesheet"
integrity="sha384-F3w7mX95PdgyTmZZMECAngseQB83DfGTowi0iMjiWaeVhAn4FJkqJByhZMI3
AhiU" crossorigin="anonymous">

  <!-- bootstrap 5 Js bundle CDN-->
  <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.1/dist/js/bootstrap.bundle.min.js"
integrity="sha384-/bQdsTh/da6pkI1MST/rWKFNjaCP5gBSY4sEBT38Q/9RBh9AH40zEOg7Hlq2TH
RZ" crossorigin="anonymous"></script>

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

</head>
<body>
  <div class="container">
    <nav class="navbar navbar-expand-lg navbar-light bg-light">
     <div class="container-fluid">
      <a class="navbar-brand" href="index.php">Online Book Store</a>
      <button class="navbar-toggler" type="button" data-bs-toggle="collapse"
data-bs-target="#navbarSupportedContent" aria-controls="navbarSupportedContent"
aria-expanded="false" aria-label="Toggle navigation">
       <span class="navbar-toggler-icon"></span>
      </button>
      <div class="collapse navbar-collapse"
         id="navbarSupportedContent">
       <ul class="navbar-nav me-auto mb-2 mb-lg-0">
        <li class="nav-item">
         <a class="nav-link active"
           aria-current="page"
           href="index.php">Store</a>
        </li>
        <li class="nav-item">
         <a class="nav-link"
           href="#">About</a>
          </li>
          <li class="nav-item">
         <a class="nav-link"
           href="research.php">Research</a>
          </li>
        <li class="nav-item">
         <?php if (isset($_SESSION['user_id'])) {?>
          <a class="nav-link"
           href="admin.php">Admin</a>
         <?php }else{ ?>
         <a class="nav-link"
           href="login.php">Login</a>
         <?php } ?>
         </li>
       </ul>
      </div>
     </div>
    </nav>
    <form action="search.php"
      method="get"
      style="width: 100%; max-width: 30 rem">
     <div class="input-group my-5">
     <input type="text"
         class="form-control"
         name="key"
         placeholder="Search Book..."
         aria-label="Search Book..."
         aria-describedby="basic-addon2">
        <button class="input-group-text
            btn btn-primary"
         id="basic-addon2">
         <img src="img/search.png"
           width="20">
   </button>
    </div>
   </form>
   <img src="img/books.jpg" width="1300" height="300">
    <div class="d-flex pt-3">
      <?php if ($books == 0){ ?>
        <div class="alert alert-warning
            text-center p-5"
         role="alert">
         <img src="img/empty.png"
           width="100">
         <br>
        There is no book in the database
       </div>
      <?php }else{ ?>
      <div class="pdf-list d-flex flex-wrap">
        <?php foreach ($books as $book) { ?>
        <div class="card m-1">
          <img src="uploads/cover/<?=$book['cover']?>"
             class="card-img-top">
          <div class="card-body">
            <h5 class="card-title">
              <?=$book['title']?>
            </h5>
            <p class="card-text">
```

```
          <i><b>By:
            <?php foreach($authors as $author){
              if ($author['id'] == $book['author_id']) {
                echo $author['name'];
                break;
              }
            ?>

            <?php } ?>
          <br></b></i>
          <?=$book['description']?>
          <br><i><b>Category:
            <?php foreach($categories as $category){
              if ($category['id'] == $book['category_id']) {
                echo $category['name'];
                break;
              }
            ?>
         <?php } ?>
          <br></b></i>
        </p>
       <a href="uploads/files/<?=$book['file']?>"
         class="btn btn-success">Open</a>

        <a href="uploads/files/<?=$book['file']?>"
         class="btn btn-primary"
         download="<?=$book['title']?>">Download</a>
      </div>
    </div>
    <?php } ?>
  </div>
<?php } ?>

<div class="category">
  <!-- List of categories -->
  <div class="list-group">
    <?php if ($categories == 0){
      // do nothing
    }else{ ?>
    <a href="#"
      class="list-group-item list-group-item-action active">Category</a>
      <?php foreach ($categories as $category ) {?>

      <a href="category.php?id=<?=$category['id']?>"
        class="list-group-item list-group-item-action">
        <?=$category['name']?></a>
    <?php } } ?>
  </div>
```

```
      <!-- List of authors -->
      <div class="list-group mt-5">
        <?php if ($authors == 0){
          // do nothing
        }else{ ?>
        <a href="#"
          class="list-group-item list-group-item-action active">Author</a>
          <?php foreach ($authors as $author ) {?>
          <a href="author.php?id=<?=$author['id']?>"
            class="list-group-item list-group-item-action">
            <?=$author['name']?></a>
        <?php } } ?>
      </div>
    </div>
    </div>
  </div>
</body>
</html>
```

Login

```
<!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>LOGIN</title>

   <!-- bootstrap 5 CDN-->
   <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.1/dist/css/bootstrap.min.css"
rel="stylesheet"
integrity="sha384-F3w7mX95PdgyTmZZMECAngseQB83DfGTowi0iMjiWaeVhAn4FJkqJByhZMI3
AhiU" crossorigin="anonymous">

  <!-- bootstrap 5 Js bundle CDN-->

    <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.1/dist/js/bootstrap.bundle.min.js"
integrity="sha384-/bQdsTh/da6pkI1MST/rWKFNjaCP5gBSY4sEBT38Q/9RBh9AH40zEOg7Hlq2TH
RZ" crossorigin="anonymous"></script>
</head>
<body>108
  <div class="d-flex justify-content-center
        align-items-center"
        style="min-height: 100vh;">
      <form class="p-5 rounded shadow"
       style="max-width: 30 rem; width: 100%"
```

```
            method="POST"
            action="php/auth.php">
            <h1 class="text-center display-4 pb-5">LOGIN</h1>
            <?php if (isset($_GET['error'])) { ?>
            <div class="alert alert-danger" role="alert">
            <?=htmlspecialchars($_GET['error']); ?>
          </div>
          <?php } ?>

            <div class="mb-3">
              <label for="exampleInputEmail1"
              class="form-label">Email address</label>
              <input type="email"
                     class="form-control"
                     name="email"
                     id="exampleInputEmail1"
                     aria-describedby="emailHelp">

           </div>
           <div class="mb-3">
              <label for="exampleInputPassword1"
              class="form-label">
                 Password</label>
              <input type="password" class="form-control"
              name="password"
               id="exampleInputPassword1">

           </div>
           <button type="submit"
                 class="btn btn-primary">
                 Login</button>
                 <a href="index.php">Store</a>
           </form>
            </div>
    </body>
    </html>
```

Admin

```
<?php
session_start();

# If the admin is logged in
if (isset($_SESSION['user_id']) &&
   isset($_SESSION['user_email'])) {

   # Database Connection File
   include "db_conn.php";
```

```
    # Book helper function
    include "php/func-book.php";
    $books = get_all_books($conn);

     # author helper function
    include "php/func-author.php";
  $authors = get_all_author($conn);

   # Category helper function
    include "php/func-category.php";
  $categories = get_all_categories($conn);

?>

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>ADMIN</title>

    <!-- bootstrap 5 CDN-->
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.1/dist/css/bootstrap.min.css" rel="stylesheet"
integrity="sha384-F3w7mX95PdgyTmZZMECAngseQB83DfGTowi0iMjiWaeVhAn4FJkqJByhZMI3
AhiU" crossorigin="anonymous">

    <!-- bootstrap 5 Js bundle CDN-->
    <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.1/dist/js/bootstrap.bundle.min.js"
integrity="sha384-/bQdsTh/da6pkI1MST/rWKFNjaCP5gBSY4sEBT38Q/9RBh9AH40zEOg7Hlq2TH
RZ" crossorigin="anonymous"></script>

</head>
<body>
    <div class="container">
        <nav class="navbar navbar-expand-lg navbar-light bg-light">
          <div class="container-fluid">
            <a class="navbar-brand" href="admin.php">Admin</a>
            <button class="navbar-toggler" type="button" data-bs-toggle="collapse"
data-bs-target="#navbarSupportedContent" aria-controls="navbarSupportedContent"
aria-expanded="false" aria-label="Toggle navigation">
              <span class="navbar-toggler-icon"></span>
            </button>
            <div class="collapse navbar-collapse"
                 id="navbarSupportedContent">
              <ul class="navbar-nav me-auto mb-2 mb-lg-0">
                <li class="nav-item">
                  <a class="nav-link"
                aria-current="page"
                href="index.php">Store</a>
            </li>
            <li class="nav-item">
              <a class="nav-link"
                href="add-book.php">Add Book</a>
            </li>
            <li class="nav-item">
              <a class="nav-link"
                href="add-category.php">Add Category</a>
            </li>
            <li class="nav-item">
              <a class="nav-link"
                href="add-author.php">Add Author</a>
            </li>
            <li class="nav-item">
              <a class="nav-link"
                href="logout.php">Logout</a>
            </li>
          </ul>
        </div>
      </div>
    </nav>
   <form action="search.php"
         method="get"
         style="width: 100%; max-width: 30 rem">

    <div class="input-group my-5">
      <input type="text"
             class="form-control"
             name="key"
             placeholder="Search Book..."
             aria-label="Search Book..."
             aria-describedby="basic-addon2">

      <button class="input-group-text
                    btn btn-primary"
              id="basic-addon2">
              <img src="img/search.png"
                   width="20">
       </button>
    </div>
   </form>
   <div class="mt-5"></div>
    <?php if (isset($_GET['error'])) { ?>
      <div class="alert alert-danger" role="alert">
          <?=htmlspecialchars($_GET['error']); ?>
      </div>
```

```
  <?php } ?>
  <?php if (isset($_GET['success'])) { ?>
   <div class="alert alert-success" role="alert">
     <?=htmlspecialchars($_GET['success']); ?>
   </div>
  <?php } ?>

<?php  if ($books == 0) { ?>
    <div class="alert alert-warning
          text-center p-5"
       role="alert">
       <img src="img/empty.jpg"
         width="100">
       <br>
     There is no book in the database
   </div>
  <?php }else {?>

 <!-- List of all books -->
  <h4>All Books</h4>
  <table class="table table-bordered shadow">
    <thead>
      <tr>
        <th>#</th>
        <th>Title</th>
        <th>Author</th>
        <th>Description</th>
        <th>Category</th>
        <th>Action</th>
      </tr>
    </thead>
    <tbody>
     <?php
     $i = 0;
     foreach ($books as $book) {
      $i++;
     ?>
     <tr>
      <td><?=$i?></td>
      <td>
        <img width="100"
           src="uploads/cover/<?=$book['cover']?>" >
        <a  class="link-dark d-block
              text-center"
          href="uploads/files/<?=$book['file']?>">
          <?=$book['title']?>
        </a>
```

```
            </td>
            <td>
              <?php if ($authors == 0) {
                echo "Undefined";}else{
                foreach ($authors as $author) {
                  if ($author['id'] == $book['author_id']) {
                    echo $author['name'];
                  }
                }
              }
              ?>

            </td>
            <td><?=$book['description']?></td>
            <td>
              <?php if ($categories == 0) {
                echo "Undefined";}else{

                foreach ($categories as $category) {
                  if ($category['id'] == $book['category_id']) {
                    echo $category['name'];
                  }
                }
              }
              ?>
            </td>
            <td>
              <a href="edit-book.php?id=<?=$book['id']?>"
                class="btn btn-warning">
                Edit</a>

              <a href="php/delete-book.php?id=<?=$book['id']?>"
                class="btn btn-danger">
                Delete</a>
            </td>
          </tr>
          <?php } ?>
        </tbody>
      </table>
      <?php }?>
    <?php  if ($categories == 0) { ?>
        <div class="alert alert-warning
              text-center p-5"
           role="alert">
           <img src="img/empty.jpg"
               width="100">
           <br>
          There is no category in the database
```

```
   </div>
<?php }else {?>
<!-- List of all categories -->
<h4 class="mt-5">All Categories</h4>
<table class="table table-bordered shadow">
   <thead>
      <tr>
         <th>#</th>
         <th>Category Name</th>
         <th>Action</th>
      </tr>
   </thead>
   <tbody>
      <?php
      $j = 0;
      foreach ($categories as $category ) {
      $j++;
      ?>
      <tr>
         <td><?=$j?></td>
         <td><?=$category['name']?></td>
         <td>
            <a href="edit-category.php?id=<?=$category['id']?>"
              class="btn btn-warning">
              Edit</a>

            <a href="php/delete-category.php?id=<?=$category['id']?>"
              class="btn btn-danger">
              Delete</a>
         </td>
      </tr>
      <?php } ?>
   </tbody>
</table>
<?php } ?>

<?php  if ($authors == 0) { ?>
   <div class="alert alert-warning
            text-center p-5"
       role="alert">
       <img src="img/empty.jpg"
           width="100">
       <br>
     There is no author in the database
   </div>
<?php }else {?>
<!-- List of all Authors -->
<h4 class="mt-5">All Authors</h4>
```

```
        <table class="table table-bordered shadow">
          <thead>
            <tr>
              <th>#</th>
              <th>Author Name</th>
              <th>Action</th>
            </tr>
          </thead>
          <tbody>
            <?php
            $k = 0;
            foreach ($authors as $author ) {
            $k++;
            ?>
            <tr>
              <td><?=$k?></td>
              <td><?=$author['name']?></td>
              <td>
                <a href="edit-author.php?id=<?=$author['id']?>"
                  class="btn btn-warning">
                  Edit</a>

                <a href="php/delete-author.php?id=<?=$author['id']?>"
                  class="btn btn-danger">
                  Delete</a>
              </td>
            </tr>
            <?php } ?>
          </tbody>
        </table>
        <?php } ?>
            </dev>
</body>
</html>

<?php }else{
  header("Location: login.php");
  exit;

} ?>
```

Logout

```
<?php

session_start();
session_unset();
session_destroy();
```

```
header("Location: login.php");
exit;
```

Author

```
<?php
session_start();

# If not author ID is set
if (!isset($_GET['id'])) {
   header("Location: index.php");
   exit;
}

# Get author ID from GET request
$id = $_GET['id'];

# Database Connection File
include "db_conn.php";

# Book helper function
include "php/func-book.php";
$books = get_books_by_author($conn, $id);

# author helper function
include "php/func-author.php";
$authors = get_all_author($conn);
$current_author = get_author($conn, $id);


# Category helper function
include "php/func-category.php";
$categories = get_all_categories($conn);
 ?>
<!DOCTYPE html>
<html lang="en">
<head>
   <meta charset="UTF-8">
   <meta name="viewport" content="width=device-width, initial-scale=1.0">
   <title><?=$current_author['name']?></title>

   <!-- bootstrap 5 CDN-->
   <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.1/dist/css/bootstrap.min.css" rel="stylesheet"
integrity="sha384-F3w7mX95PdgyTmZZMECAngseQB83DfGTowi0iMjiWaeVhAn4FJkqJByhZMI3
AhiU" crossorigin="anonymous">

   <!-- bootstrap 5 Js bundle CDN-->
```

```
  <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.1/dist/js/bootstrap.bundle.min.js"
integrity="sha384-/bQdsTh/da6pkI1MST/rWKFNjaCP5gBSY4sEBT38Q/9RBh9AH40zEOg7Hlq2TH
RZ" crossorigin="anonymous"></script>

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

</head>
<body>
  <div class="container">
    <nav class="navbar navbar-expand-lg navbar-light bg-light">
     <div class="container-fluid">
      <a class="navbar-brand" href="index.php">Online Book Store</a>
      <button class="navbar-toggler" type="button" data-bs-toggle="collapse"
data-bs-target="#navbarSupportedContent" aria-controls="navbarSupportedContent"
aria-expanded="false" aria-label="Toggle navigation">
       <span class="navbar-toggler-icon"></span>
      </button>
      <div class="collapse navbar-collapse"
         id="navbarSupportedContent">
       <ul class="navbar-nav me-auto mb-2 mb-lg-0">
        <li class="nav-item">
         <a class="nav-link active"
           aria-current="page"
           href="index.php">Store</a>
        </li>
        <li class="nav-item">
         <a class="nav-link"
           href="#">Contact</a>
        </li>
        <li class="nav-item">
         <a class="nav-link"
           href="#">About</a>
        </li>
        <li class="nav-item">
         <?php if (isset($_SESSION['user_id'])) {?>
          <a class="nav-link"
           href="admin.php">Admin</a>
         <?php }else{ ?>
         <a class="nav-link"
           href="login.php">Login</a>
         <?php } ?>
       </li>
      </ul>
     </div>
    </div>
   </nav>
   <h1 class="display-4 p-3 fs-3">
     <a href="index.php"
```

```
        class="nd">
         <img src="img/back-arrow.PNG"
              width="35">
       </a>
      <?=$current_author['name']?>
    </h1>
    <div class="d-flex pt-3">
       <?php if ($books == 0){ ?>
          <div class="alert alert-warning
                text-center p-5"
           role="alert">
           <img src="img/empty.jpg"
               width="100">
           <br>
          There is no book in the database
         </div>
       <?php }else{ ?>
       <div class="pdf-list d-flex flex-wrap">
          <?php foreach ($books as $book) { ?>
          <div class="card m-1">
             <img src="uploads/cover/<?=$book['cover']?>"
                 class="card-img-top">
             <div class="card-body">
                <h5 class="card-title">
                   <?=$book['title']?>
                </h5>
                <p class="card-text">
                   <i><b>By:
                      <?php foreach($authors as $author){
                         if ($author['id'] == $book['author_id']) {
                            echo $author['name'];
                            break;
                         }
                      ?>
                      <?php } ?>
                   <br></b></i>
                   <?=$book['description']?>
                   <br><i><b>Category:
                      <?php foreach($categories as $category){
                         if ($category['id'] == $book['category_id']) {
                            echo $category['name'];
                            break;
                         }
                      ?>
                      <?php } ?>
                   <br></b></i>
                </p>
               <a href="uploads/files/<?=$book['file']?>"
```

```
                    class="btn btn-success">Open</a>

                  <a href="uploads/files/<?=$book['file']?>"
                    class="btn btn-primary"
                    download="<?=$book['title']?>">Download</a>
              </div>
            </div>
            <?php } ?>
          </div>
        <?php } ?>

        <div class="category">
          <!-- List of categories -->
          <div class="list-group">
            <?php if ($categories == 0){
              // do nothing
            }else{ ?>
            <a href="#"
              class="list-group-item list-group-item-action active">Category</a>
              <?php foreach ($categories as $category ) {?>

              <a href="category.php?id=<?=$category['id']?>"
                class="list-group-item list-group-item-action">
                <?=$category['name']?></a>
            <?php } } ?>
          </div>

          <!-- List of authors -->
          <div class="list-group mt-5">
            <?php if ($authors == 0){
              // do nothing
            }else{ ?>
            <a href="#"
              class="list-group-item list-group-item-action active">Author</a>
              <?php foreach ($authors as $author ) {?>

              <a href="author.php?id=<?=$author['id']?>"
                class="list-group-item list-group-item-action">
                <?=$author['name']?></a>
            <?php } } ?>
          </div>
        </div>
        </div>
      </div>
    </body>
    </html>
```

Category

```
<?php
session_start();

# If not category ID is set
if (!isset($_GET['id'])) {
        header("Location: index.php");
        exit;
}
# Get category ID from GET request
$id = $_GET['id'];

# Database Connection File
include "db_conn.php";

# Book helper function
include "php/func-book.php";
$books = get_books_by_category($conn, $id);

# author helper function
include "php/func-author.php";
$authors = get_all_author($conn);

# Category helper function
include "php/func-category.php";
$categories = get_all_categories($conn);
$current_category = get_category($conn, $id);

 ?>
<!DOCTYPE html>
<html lang="en">
<head>
        <meta charset="UTF-8">
        <meta name="viewport" content="width=device-width, initial-scale=1.0">
        <title><?=$current_category['name']?></title>

  <!-- bootstrap 5 CDN-->
        <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.1/dist/css/bootstrap.min.css"
rel="stylesheet"
integrity="sha384-F3w7mX95PdgyTmZZMECAngseQB83DfGTowi0iMjiWaeVhAn4FJkqJByhZMI3
AhiU" crossorigin="anonymous">

  <!-- bootstrap 5 Js bundle CDN-->
  <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.1/dist/js/bootstrap.bundle.min.js"
integrity="sha384-/bQdsTh/da6pkI1MST/rWKFNjaCP5gBSY4sEBT38Q/9RBh9AH40zEOg7Hlq2TH
RZ" crossorigin="anonymous"></script>

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

```
</head>
<body>
        <div class="container">
                <nav class="navbar navbar-expand-lg navbar-light bg-light">
                  <div class="container-fluid">
                    <a class="navbar-brand" href="index.php">Online Book Store</a>
                    <button class="navbar-toggler" type="button" data-bs-toggle="collapse"
data-bs-target="#navbarSupportedContent" aria-controls="navbarSupportedContent"
aria-expanded="false" aria-label="Toggle navigation">
                      <span class="navbar-toggler-icon"></span>
                    </button>
                    <div class="collapse navbar-collapse"
                         id="navbarSupportedContent">
                      <ul class="navbar-nav me-auto mb-2 mb-lg-0">
                        <li class="nav-item">
                          <a class="nav-link active"
                             aria-current="page"
                             href="index.php">Store</a>
                        </li>
                        <li class="nav-item">
                          <a class="nav-link"
                             href="#">Contact</a>
                        </li>
                        <li class="nav-item">
                          <a class="nav-link"
                             href="#">About</a>
                        </li>
                        <li class="nav-item">
                          <?php if (isset($_SESSION['user_id'])) {?>
                            <a class="nav-link"
                             href="admin.php">Admin</a>
                          <?php }else{ ?>
                          <a class="nav-link"
                             href="login.php">Login</a>
                          <?php } ?>
                        </li>
                      </ul>
                    </div>
                  </div>
                </nav>
                <h1 class="display-4 p-3 fs-3">
                        <a href="index.php"
                           class="nd">
                                <img src="img/back-arrow.PNG"
                                     width="35">
                        </a>
                   <?=$current_category['name']?>
        </h1>
        <div class="d-flex pt-3">
            <?php if ($books == 0){ ?>
                <div class="alert alert-warning
        text-center p-5"
      role="alert">
      <img src="img/empty.jpg"
       width="100">
      <br>
             There is no book in the database
          </div>
            <?php }else{ ?>
            <div class="pdf-list d-flex flex-wrap">
                <?php foreach ($books as $book) { ?>
                <div class="card m-1">
                    <img src="uploads/cover/<?=$book['cover']?>"
                      class="card-img-top">
                    <div class="card-body">
                        <h5 class="card-title">
                            <?=$book['title']?>
                        </h5>
                        <p class="card-text">
                            <i><b>By:
                            <?php foreach($authors as $author){
                                if ($author['id'] == $book['author_id']) {
                                    echo $author['name'];
                                        break;
                                    }
                                ?>

                                <?php } ?>
                            <br></b></i>
                            <?=$book['description']?>
                            <br><i><b>Category:
                            <?php foreach($categories as $category){
                                if ($category['id'] == $book['category_id'])
{
                                        echo $category['name'];
                                        break;
                                    }
                                ?>

                                <?php } ?>
                            <br></b></i>
                        </p>
       <a href="uploads/files/<?=$book['file']?>"
        class="btn btn-success">Open</a>
        <a href="uploads/files/<?=$book['file']?>"
          class="btn btn-primary"
          download="<?=$book['title']?>">Download</a>
                        </div>
                </div>
                <?php } ?>
            </div>
        <?php } ?>


        <div class="category">
            <!-- List of categories -->
            <div class="list-group">
                <?php if ($categories == 0){
                        // do nothing
                }else{ ?>
                <a href="#"
                  class="list-group-item list-group-item-action active">Category</a>
                  <?php foreach ($categories as $category ) {?>

                  <a href="category.php?id=<?=$category['id']?>"
                    class="list-group-item list-group-item-action">
                    <?=$category['name']?></a>
                <?php } } ?>
            </div>

            <!-- List of authors -->
            <div class="list-group mt-5">
                <?php if ($authors == 0){
                        // do nothing
                }else{ ?>
                <a href="#"
                  class="list-group-item list-group-item-action active">Author</a>
                  <?php foreach ($authors as $author ) {?>

                  <a href="author.php?id=<?=$author['id']?>"
                    class="list-group-item list-group-item-action">
                    <?=$author['name']?></a>
                <?php } } ?>
            </div>
        </div>
        </div>
    </div>
</body>
</html>
```

Search

```
<?php
session_start();

# If search key is not set or empty
if (!isset($_GET['key']) || empty($_GET['key'])) {
    header("Location: index.php");
    exit;
}
$key = $_GET['key'];

# Database Connection File
include "db_conn.php";

# Book helper function
include "php/func-book.php";
$books = search_books($conn, $key);

# author helper function
include "php/func-author.php";
$authors = get_all_author($conn);

# Category helper function
include "php/func-category.php";
$categories = get_all_categories($conn);

 ?>
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Book Store</title>

    <!-- bootstrap 5 CDN-->
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.1/dist/css/bootstrap.min.css" rel="stylesheet"
integrity="sha384-F3w7mX95PdgyTmZZMECAngseQB83DfGTowi0iMjiWaeVhAn4FJkqJByhZMI3
AhiU" crossorigin="anonymous">

    <!-- bootstrap 5 Js bundle CDN-->
    <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.1/dist/js/bootstrap.bundle.min.js"
integrity="sha384-/bQdsTh/da6pkI1MST/rWKFNjaCP5gBSY4sEBT38Q/9RBh9AH40zEOg7Hlq2TH
RZ" crossorigin="anonymous"></script>

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

</head>
```

```
<body>
  <div class="container">
    <nav class="navbar navbar-expand-lg navbar-light bg-light">
     <div class="container-fluid">
      <a class="navbar-brand" href="index.php">Online Book Store</a>
      <button class="navbar-toggler" type="button" data-bs-toggle="collapse"
data-bs-target="#navbarSupportedContent" aria-controls="navbarSupportedContent"
aria-expanded="false" aria-label="Toggle navigation">
       <span class="navbar-toggler-icon"></span>
      </button>
      <div class="collapse navbar-collapse"
        id="navbarSupportedContent">
       <ul class="navbar-nav me-auto mb-2 mb-lg-0">
        <li class="nav-item">
         <a class="nav-link active"
           aria-current="page"
           href="index.php">Store</a>
        </li>
        <li class="nav-item">
         <a class="nav-link"
           href="#">Contact</a>
        </li>
        <li class="nav-item">
         <a class="nav-link"
           href="#">About</a>
        </li>
        <li class="nav-item">
         <?php if (isset($_SESSION['user_id'])) {?>
          <a class="nav-link"
           href="admin.php">Admin</a>
         <?php }else{ ?>
         <a class="nav-link"
           href="login.php">Login</a>
         <?php } ?>

        </li>
       </ul>
      </div>
     </div>
    </nav><br>
    Search result for <b><?=$key?></b>

    <div class="d-flex pt-3">
      <?php if ($books == 0){ ?>
        <div class="alert alert-warning
            text-center p-5 pdf-list"
         role="alert">
         <img src="img/search_no_result.png"
```

```
          width="220">
     <br>
      The key <b>"<?=$key?>"</b> didn't match to any record
       in the database
  </div>
<?php }else{ ?>
<div class="pdf-list d-flex flex-wrap">
    <?php foreach ($books as $book) { ?>
    <div class="card m-1">
        <img src="uploads/cover/<?=$book['cover']?>"
             class="card-img-top">
        <div class="card-body">
            <h5 class="card-title">
                <?=$book['title']?>
            </h5>
            <p class="card-text">
                <i><b>By:
                    <?php foreach($authors as $author){
                        if ($author['id'] == $book['author_id']) {
                            echo $author['name'];
                            break;
                        }
                    ?>

                    <?php } ?>
                <br></b></i>
                <?=$book['description']?>
                <br><i><b>Category:
                    <?php foreach($categories as $category){
                        if ($category['id'] == $book['category_id']) {
                            echo $category['name'];
                            break;
                        }
                    ?>

                    <?php } ?>
                <br></b></i>
            </p>
           <a href="uploads/files/<?=$book['file']?>"
              class="btn btn-success">Open</a>

            <a href="uploads/files/<?=$book['file']?>"
              class="btn btn-primary"
              download="<?=$book['title']?>">Download</a>
        </div>
    </div>
    <?php } ?>
</div>
```

```
        <?php } ?>
        </div>
    </div>
</body>
</html>
```

Research

```
<?php
session_start();

# Database Connection File
include "db_conn.php";

# Book helper function
include "php/func-book.php";
$books = get_all_books($conn);

# author helper function
include "php/func-author.php";
$authors = get_all_author($conn);

# Category helper function
include "php/func-category.php";
$categories = get_all_categories($conn);

 ?>
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Book Store</title>

    <!-- bootstrap 5 CDN-->
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.1/dist/css/bootstrap.min.css" rel="stylesheet"
integrity="sha384-F3w7mX95PdgyTmZZMECAngseQB83DfGTowi0iMjiWaeVhAn4FJkqJByhZMI3
AhiU" crossorigin="anonymous">

    <!-- bootstrap 5 Js bundle CDN-->
    <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.1/dist/js/bootstrap.bundle.min.js"
integrity="sha384-/bQdsTh/da6pkI1MST/rWKFNjaCP5gBSY4sEBT38Q/9RBh9AH40zEOg7Hlq2TH
RZ" crossorigin="anonymous"></script>

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

</head>
<body>
```

```
    <div class="container">
        <nav class="navbar navbar-expand-lg navbar-light bg-light">
          <div class="container-fluid">
            <a class="navbar-brand" href="index.php">Online Book Store</a>
            <button class="navbar-toggler" type="button" data-bs-toggle="collapse"
data-bs-target="#navbarSupportedContent" aria-controls="navbarSupportedContent"
aria-expanded="false" aria-label="Toggle navigation">
              <span class="navbar-toggler-icon"></span>
            </button>
            <div class="collapse navbar-collapse"
                 id="navbarSupportedContent">
              <ul class="navbar-nav me-auto mb-2 mb-lg-0">
                <li class="nav-item">
                  <a class="nav-link active"
                     aria-current="page"
                     href="index.php">Store</a>
                </li>

                <li class="nav-item">
                  <a class="nav-link"
                     href="#">About</a>
                    </li>
                    <li class="nav-item">
                  <a class="nav-link"
                     href="research.php">Research</a>
                    </li>
                <li class="nav-item">
                  <?php if (isset($_SESSION['user_id'])) {?>
                    <a class="nav-link"
                     href="admin.php">Admin</a>
                  <?php }else{ ?>
                  <a class="nav-link"
                   href="login.php">Login</a>
                  <?php } ?>

                </li>
              </ul>
            </div>
          </div>
        </nav>

        <form action="search.php"
             method="get"
             style="width: 100%; max-width: 30rem">

        <div class="input-group my-5">
          <input type="text"
```

```
         class="form-control"
         name="key"
         placeholder="Search Book..."
         aria-label="Search Book..."
         aria-describedby="basic-addon2">

   <button class="input-group-text
                btn btn-primary"
          id="basic-addon2">
          <img src="img/search.png"
               width="20">

   </button>
 </div>
</form>
<img src="img/bit.jpeg" width="1300" height="350"
 <div class="d-flex pt-3">

    <?php if ($books == 0){ ?>
        <div class="alert alert-warning
                text-center p-5"
         role="alert">
         <img src="img/empty.png"
              width="100">
         <br>
        There is no book in the database
       </div>
    <?php }else{ ?>
    <div class="pdf-list d-flex flex-wrap">
        <?php foreach ($books as $book) { ?>
            <?php
    if($book['category_id']==9){?>
        <div class="card m-1">
         <img src="uploads/cover/<?=$book['cover']?>"
             class="card-img-top">
            <div class="card-body">
                <h5 class="card-title">
                    <?=$book['title']?>
                </h5>
                <p class="card-text">
                    <i><b>By:
                        <?php foreach($authors as $author){
                            if ($author['id'] == $book['author_id']) {
                                echo $author['name'];
                                break;
                            }
                        ?>
```

```
                                <?php } ?>
                            <br></b></i>
                            <?=$book['description']?>
                            <br><i><b>Category:
                                <?php foreach($categories as $category){
                                    if ($category['id'] == $book['category_id']) {
                                        echo $category['name'];
                                        break;
                                    }
                                ?>

                                <?php } ?>
                            <br></b></i>
                        </p>
                       <a href="uploads/files/<?=$book['file']?>"
                          class="btn btn-success">Open</a>

                        <a href="uploads/files/<?=$book['file']?>"
                          class="btn btn-primary"
                          download="<?=$book['title']?>">Download</a>
                    </div>
                             <?php   }
                                else{
                                    break;
                                }?>
                </div>
                <?php } ?>
            </div>
        <?php } ?>


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

Db_conn

```
<?php

# server name
$sName = "localhost";
# user name
$uName = "root";
# password
$pass = "";

# database name
```

```
$db_name = "online_book_store_db";

/**
creating database connection
using The PHP Data Objects (PDO)
**/
try {
   $conn = new PDO("mysql:host=$sName;dbname=$db_name",
                  $uName, $pass);
   $conn->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
}catch(PDOException $e){
  echo "Connection failed : ". $e->getMessage();
}
```

Add author

```
<?php
session_start();

# If the admin is logged in
if (isset($_SESSION['user_id']) &&
   isset($_SESSION['user_email'])) {
?>

<!DOCTYPE html>
<html lang="en">
<head>
   <meta charset="UTF-8">
   <meta name="viewport" content="width=device-width, initial-scale=1.0">
   <title>Add Author</title>

   <!-- bootstrap 5 CDN-->
   <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.1/dist/css/bootstrap.min.css" rel="stylesheet"
integrity="sha384-F3w7mX95PdgyTmZZMECAngseQB83DfGTowi0iMjiWaeVhAn4FJkqJByhZMI3
AhiU" crossorigin="anonymous">
   <!-- bootstrap 5 Js bundle CDN-->
   <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.1/dist/js/bootstrap.bundle.min.js"
integrity="sha384-/bQdsTh/da6pkI1MST/rWKFNjaCP5gBSY4sEBT38Q/9RBh9AH40zEOg7Hlq2TH
RZ" crossorigin="anonymous"></script>

</head>
<body>
   <div class="container">
       <nav class="navbar navbar-expand-lg navbar-light bg-light">
         <div class="container-fluid">
           <a class="navbar-brand" href="admin.php">Admin</a>
```

```
            <button class="navbar-toggler" type="button" data-bs-toggle="collapse"
data-bs-target="#navbarSupportedContent" aria-controls="navbarSupportedContent"
aria-expanded="false" aria-label="Toggle navigation">
              <span class="navbar-toggler-icon"></span>
            </button>
            <div class="collapse navbar-collapse"
                 id="navbarSupportedContent">
              <ul class="navbar-nav me-auto mb-2 mb-lg-0">
                <li class="nav-item">
                  <a class="nav-link"
                     aria-current="page"
                     href="index.php">Store</a>
                </li>
                <li class="nav-item">
                  <a class="nav-link"
                     href="add-book.php">Add Book</a>
                </li>
                <li class="nav-item">
                  <a class="nav-link"
                     href="add-category.php">Add Category</a>
                </li>
                <li class="nav-item">
                  <a class="nav-link active"
                     href="add-author.php">Add Author</a>
                </li>
                <li class="nav-item">
                  <a class="nav-link"
                     href="logout.php">Logout</a>
                </li>
              </ul>
            </div>
          </div>
          </nav>
     <form action="php/add-author.php"
           method="post"
           class="shadow p-4 rounded mt-5"
           style="width: 90%; max-width: 50rem;"
        <h1 class="text-center pb-5 display-4 fs-3">
            Add New Author
        </h1>
        <?php if (isset($_GET['error'])) { ?>
          <div class="alert alert-danger" role="alert">
              <?=htmlspecialchars($_GET['error']); ?>
          </div>
        <?php } ?>
        <?php if (isset($_GET['success'])) { ?>
          <div class="alert alert-success" role="alert">
              <?=htmlspecialchars($_GET['success']); ?>
          </div>
        <?php } ?>
        <div class="mb-3">
            <label class="form-label">
                    Author Name
                   </label>
            <input type="text"
                   class="form-control"
                   name="author_name">
        </div>

        <button type="submit"
                class="btn btn-primary">
                Add Author</button>
     </form>
    </div>
</body>
</html>

<?php }else{
  header("Location: login.php");
  exit;
} ?>
```

Add category

```
<?php
session_start();

# If the admin is logged in
if (isset($_SESSION['user_id']) &&
   isset($_SESSION['user_email'])) {
?>

<!DOCTYPE html>
<html lang="en">
<head>
   <meta charset="UTF-8">
   <meta name="viewport" content="width=device-width, initial-scale=1.0">
   <title>Add Category</title>

   <!-- bootstrap 5 CDN-->
   <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.1/dist/css/bootstrap.min.css" rel="stylesheet"
integrity="sha384-F3w7mX95PdgyTmZZMECAngseQB83DfGTowi0iMjiWaeVhAn4FJkqJByhZMI3
AhiU" crossorigin="anonymous">

   <!-- bootstrap 5 Js bundle CDN-->
```

```
  <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.1/dist/js/bootstrap.bundle.min.js"
integrity="sha384-/bQdsTh/da6pkI1MST/rWKFNjaCP5gBSY4sEBT38Q/9RBh9AH40zEOg7Hlq2TH
RZ" crossorigin="anonymous"></script>

</head>
<body>
  <div class="container">
      <nav class="navbar navbar-expand-lg navbar-light bg-light">
        <div class="container-fluid">
          <a class="navbar-brand" href="admin.php">Admin</a>
          <button class="navbar-toggler" type="button" data-bs-toggle="collapse"
data-bs-target="#navbarSupportedContent" aria-controls="navbarSupportedContent"
aria-expanded="false" aria-label="Toggle navigation">
            <span class="navbar-toggler-icon"></span>
          </button>
          <div class="collapse navbar-collapse"
               id="navbarSupportedContent">
            <ul class="navbar-nav me-auto mb-2 mb-lg-0">
              <li class="nav-item">
                <a class="nav-link"
                   aria-current="page"
                   href="index.php">Store</a>
              </li>
              <li class="nav-item">
                <a class="nav-link"
                   href="add-book.php">Add Book</a>
              </li>
              <li class="nav-item">
                <a class="nav-link active"
                   href="add-category.php">Add Category</a>
              </li>
              <li class="nav-item">
                <a class="nav-link"
                   href="add-author.php">Add Author</a>
              </li>
              <li class="nav-item">
                <a class="nav-link"
                   href="logout.php">Logout</a>
              </li>
            </ul>
          </div>
        </div>
      </nav>
    <form action="php/add-category.php"
          method="post"
          class="shadow p-4 rounded mt-5"
          style="width: 90%; max-width: 50rem;">
        <h1 class="text-center pb-5 display-4 fs-3">
            Add New Category
        </h1>
        <?php if (isset($_GET['error'])) { ?>
          <div class="alert alert-danger" role="alert">
              <?=htmlspecialchars($_GET['error']); ?>
          </div>
        <?php } ?>
        <?php if (isset($_GET['success'])) { ?>
          <div class="alert alert-success" role="alert">
              <?=htmlspecialchars($_GET['success']); ?>
          </div>
        <?php } ?>
        <div class="mb-3">
            <label class="form-label">
                   Category Name
                   </label>
            <input type="text"
                   class="form-control"
                   name="category_name">
        </div>

        <button type="submit"
                class="btn btn-primary">
                Add Category</button>
     </form>
    </div>
</body>
</html>

<?php }else{
 header("Location: login.php");
 exit;
} ?>
```

Add-book

```
<?php
session_start();

# If the admin is logged in
if (isset($_SESSION['user_id']) &&
   isset($_SESSION['user_email'])) {

   # Database Connection File
   include "db_conn.php";<?

   # Category helper function
   include "php/func-category.php";
```

```
    $categories = get_all_categories($conn);

    # author helper function
    include "php/func-author.php";
    $authors = get_all_author($conn);

    if (isset($_GET['title'])) {
        $title = $_GET['title'];
    }else $title = '';

    if (isset($_GET['desc'])) {
        $desc = $_GET['desc'];
    }else $desc = '';

    if (isset($_GET['category_id'])) {
        $category_id = $_GET['category_id'];
    }else $category_id = 0;

    if (isset($_GET['author_id'])) {
        $author_id = $_GET['author_id'];
    }else $author_id = 0;
?>

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Add Book</title>

    <!-- bootstrap 5 CDN-->
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.1/dist/css/bootstrap.min.css" rel="stylesheet"
integrity="sha384-F3w7mX95PdgyTmZZMECAngseQB83DfGTowi0iMjiWaeVhAn4FJkqJByhZMI3
AhiU" crossorigin="anonymous">

    <!-- bootstrap 5 Js bundle CDN-->
    <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.1/dist/js/bootstrap.bundle.min.js"
integrity="sha384-/bQdsTh/da6pkI1MST/rWKFNjaCP5gBSY4sEBT38Q/9RBh9AH40zEOg7Hlq2TH
RZ" crossorigin="anonymous"></script>

</head>
<body>
    <div class="container">
        <nav class="navbar navbar-expand-lg navbar-light bg-light">
          <div class="container-fluid">
            <a class="navbar-brand" href="admin.php">Admin</a>
          <button class="navbar-toggler" type="button" data-bs-toggle="collapse"
data-bs-target="#navbarSupportedContent" aria-controls="navbarSupportedContent"
aria-expanded="false" aria-label="Toggle navigation">
            <span class="navbar-toggler-icon"></span>
          </button>
          <div class="collapse navbar-collapse"
               id="navbarSupportedContent">
            <ul class="navbar-nav me-auto mb-2 mb-lg-0">
              <li class="nav-item">
                <a class="nav-link"
                   aria-current="page"
                   href="index.php">Store</a>
              </li>
              <li class="nav-item">
                <a class="nav-link active"
                   href="add-book.php">Add Book</a>
              </li>
              <li class="nav-item">
                <a class="nav-link"
                   href="add-category.php">Add Category</a>
              </li>
              <li class="nav-item">
                <a class="nav-link"
                   href="add-author.php">Add Author</a>
              </li>
              <li class="nav-item">
                <a class="nav-link"
                   href="logout.php">Logout</a>
              </li>
            </ul>
          </div>
        </div>
      </nav>
    <form action="php/add-book.php"
          method="post"
          enctype="multipart/form-data"
          class="shadow p-4 rounded mt-5"
          style="width: 90%; max-width: 50rem;">

      <h1 class="text-center pb-5 display-4 fs-3">
         Add New Book
      </h1>
      <?php if (isset($_GET['error'])) { ?>
        <div class="alert alert-danger" role="alert">
          <?=htmlspecialchars($_GET['error']); ?>
        </div>
      <?php } ?>
      <?php if (isset($_GET['success'])) { ?>
```

```
  <div class="alert alert-success" role="alert">
      <?=htmlspecialchars($_GET['success']); ?>
  </div>
<?php } ?>
<div class="mb-3">
    <label class="form-label">
           Book Title
           </label>
    <input type="text"
           class="form-control"
           value="<?=$title?>"
           name="book_title">
</div>

<div class="mb-3">
    <label class="form-label">
           Book Description
           </label>
    <input type="text"
           class="form-control"
           value="<?=$desc?>"
           name="book_description">
</div>

<div class="mb-3">
    <label class="form-label">
           Book Author
           </label>
    <select name="book_author"
            class="form-control">
            <option value="0">
                Select author
            </option>
            <?php
            if ($authors == 0) {
                # Do nothing!
            }else{
            foreach ($authors as $author) {
                if ($author_id == $author['id']) { ?>
                <option
                  selected
                  value="<?=$author['id']?>">
                  <?=$author['name']?>
                </option>
                <?php }else{ ?>
                <option
                    value="<?=$author['id']?>">
                    <?=$author['name']?>
                    </option>
                <?php }} } ?>
            </select>
        </div>

        <div class="mb-3">
            <label class="form-label">
                Book Category
                </label>



        <div class="mb-3">
            <label class="form-label">
                File
                </label>
            <input type="file"
                class="form-control"
                name="file">
        </div>

        <button type="submit"
              class="btn btn-primary">
              Add Book</button>
      </form>
     </div>
   </body>
   </html>

   <?php }else{
    header("Location: login.php");
    exit;
   } ?>
```

Edit-Author

```
<?php
session_start();

# If the admin is logged in
if (isset($_SESSION['user_id']) &&
   isset($_SESSION['user_email'])) {

   # If author ID is not set
   if (!isset($_GET['id'])) {
       #Redirect to admin.php page
       header("Location: admin.php");
```

```
        exit;
    }

    $id = $_GET['id'];

    # Database Connection File
    include "db_conn.php";

    # author helper function
    include "php/func-author.php";
    $author = get_author($conn, $id);

    # If the ID is invalid
    if ($author == 0) {
        #Redirect to admin.php page
        header("Location: admin.php");
        exit;
    }
?>

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Edit Author</title>

    <!-- bootstrap 5 CDN-->
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.1/dist/css/bootstrap.min.css" rel="stylesheet"
integrity="sha384-F3w7mX95PdgyTmZZMECAngseQB83DfGTowi0iMjiWaeVhAn4FJkqJByhZMI3
AhiU" crossorigin="anonymous">

    <!-- bootstrap 5 Js bundle CDN-->
    <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.1/dist/js/bootstrap.bundle.min.js"
integrity="sha384-/bQdsTh/da6pkI1MST/rWKFNjaCP5gBSY4sEBT38Q/9RBh9AH40zEOg7Hlq2TH
RZ" crossorigin="anonymous"></script>

</head>
<body>
    <div class="container">
        <nav class="navbar navbar-expand-lg navbar-light bg-light">
          <div class="container-fluid">
            <a class="navbar-brand" href="admin.php">Admin</a>
            <button class="navbar-toggler" type="button" data-bs-toggle="collapse"
data-bs-target="#navbarSupportedContent" aria-controls="navbarSupportedContent"
aria-expanded="false" aria-label="Toggle navigation">
              <span class="navbar-toggler-icon"></span>
            </button>
```

```
       <div class="collapse navbar-collapse"
            id="navbarSupportedContent">
         <ul class="navbar-nav me-auto mb-2 mb-lg-0">
           <li class="nav-item">
             <a class="nav-link"
                aria-current="page"
                href="index.php">Store</a>
           </li>
           <li class="nav-item">
             <a class="nav-link"
                href="add-book.php">Add Book</a>
           </li>
           <li class="nav-item">
             <a class="nav-link"
                href="add-category.php">Add Category</a>
           </li>
           <li class="nav-item">
             <a class="nav-link"
                href="add-author.php">Add Author</a>
           </li>
           <li class="nav-item">
             <a class="nav-link"
                href="logout.php">Logout</a>
           </li>
         </ul>
       </div>
     </div>
   </nav>
<form action="php/edit-author.php"
      method="post"
      class="shadow p-4 rounded mt-5"
      style="width: 90%; max-width: 50rem;">

   <h1 class="text-center pb-5 display-4 fs-3">
       Edit Author
   </h1>
   <?php if (isset($_GET['error'])) { ?>
     <div class="alert alert-danger" role="alert">
         <?=htmlspecialchars($_GET['error']); ?>
     </div>
   <?php } ?>
   <?php if (isset($_GET['success'])) { ?>
     <div class="alert alert-success" role="alert">
         <?=htmlspecialchars($_GET['success']); ?>
     </div>
   <?php } ?>
   <div class="mb-3">
       <label class="form-label">
```

```
                Author Name
               </label>

            <input type="text"
                value="<?=$author['id'] ?>"
                hidden
                name="author_id">


           <input type="text"
               class="form-control"
               value="<?=$author['name'] ?>"
               name="author_name">
       </div>

       <button type="submit"
             class="btn btn-primary">
             Update</button>
     </form>
   </div>
</body>
</html>

<?php }else{
 header("Location: login.php");
 exit;
} ?>
```

Edit-Book

```
?php
session_start();

# If the admin is logged in
if (isset($_SESSION['user_id']) &&
   isset($_SESSION['user_email'])) {

   # If book ID is not set
   if (!isset($_GET['id'])) {
       #Redirect to admin.php page
       header("Location: admin.php");
       exit;
   }

   $id = $_GET['id'];

   # Database Connection File
```

```
  include "db_conn.php";

  # Book helper function
  include "php/func-book.php";
  $book = get_book($conn, $id);

  # If the ID is invalid
  if ($book == 0) {
     #Redirect to admin.php page
     header("Location: admin.php");
     exit;
  }

  # Category helper function
  include "php/func-category.php";
  $categories = get_all_categories($conn);

  # author helper function
  include "php/func-author.php";
  $authors = get_all_author($conn);
?>

<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Edit Book</title>

  <!-- bootstrap 5 CDN-->
  <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.1/dist/css/bootstrap.min.css" rel="stylesheet"
integrity="sha384-F3w7mX95PdgyTmZZMECAngseQB83DfGTowi0iMjiWaeVhAn4FJkqJByhZMI3
AhiU" crossorigin="anonymous">

  <!-- bootstrap 5 Js bundle CDN-->
  <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.1/dist/js/bootstrap.bundle.min.js"
integrity="sha384-/bQdsTh/da6pkI1MST/rWKFNjaCP5gBSY4sEBT38Q/9RBh9AH40zEOg7Hlq2TH
RZ" crossorigin="anonymous"></script>

</head>
<body>
  <div class="container">
     <nav class="navbar navbar-expand-lg navbar-light bg-light">
       <div class="container-fluid">
         <a class="navbar-brand" href="admin.php">Admin</a>
         <button class="navbar-toggler" type="button" data-bs-toggle="collapse"
data-bs-target="#navbarSupportedContent" aria-controls="navbarSupportedContent"
aria-expanded="false" aria-label="Toggle navigation">
```

```
      <span class="navbar-toggler-icon"></span>
     </button>
     <div class="collapse navbar-collapse"
        id="navbarSupportedContent">
      <ul class="navbar-nav me-auto mb-2 mb-lg-0">
       <li class="nav-item">
         <a class="nav-link"
           aria-current="page"
           href="index.php">Store</a>
       </li>
       <li class="nav-item">
         <a class="nav-link"
           href="add-book.php">Add Book</a>
       </li>
       <li class="nav-item">
         <a class="nav-link"
           href="add-category.php">Add Category</a>
       </li>
       <li class="nav-item">
         <a class="nav-link"
           href="add-author.php">Add Author</a>
       </li>
       <li class="nav-item">
         <a class="nav-link"
           href="logout.php">Logout</a>
       </li>
      </ul>
     </div>
   </div>
  </nav>
<form action="php/edit-book.php"
    method="post"
    enctype="multipart/form-data"
    class="shadow p-4 rounded mt-5"
    style="width: 90%; max-width: 50rem;">

  <h1 class="text-center pb-5 display-4 fs-3">
     Edit Book
  </h1>
  <?php if (isset($_GET['error'])) { ?>
   <div class="alert alert-danger" role="alert">
      <?=htmlspecialchars($_GET['error']); ?>        </div>

  <?php } ?>
  <?php if (isset($_GET['success'])) { ?>
   <div class="alert alert-success" role="alert">
      <?=htmlspecialchars($_GET['success']); ?>
   </div>
```

```
<?php } ?>
<div class="mb-3">
  <label class="form-label">
      Book Title
      </label>
  <input type="text"
      hidden
      value="<?=$book['id']?>"
      name="book_id">

  <input type="text"
      class="form-control"
      value="<?=$book['title']?>"
      name="book_title">
</div>

<div class="mb-3">
  <label class="form-label">
      Book Description
      </label>
  <input type="text"
      class="form-control"
      value="<?=$book['description']?>"
      name="book_description">
</div>

<div class="mb-3">
  <label class="form-label">
      Book Author
      </label>
  <select name="book_author"
       class="form-control">
       <option value="0">
          Select author
       </option>
       <?php
       if ($authors == 0) {
          # Do nothing!
       }else{
       foreach ($authors as $author) {
          if ($book['author_id'] == $author['id']) { ?>
          <option
            selected
            value="<?=$author['id']?>">
            <?=$author['name']?>
          </option>
          <?php }else{ ?>
          <option
```

```
               value="<?=$author['id']?>">
               <?=$author['name']?>
            </option>
        <?php }} } ?>
   </select>
</div>

   </select>
</div>

<div class="mb-3">
   <label class="form-label">
        Book Cover
        </label>
   <input type="file"
        class="form-control"
        name="book_cover">

    <input type="text"      <div class="mb-3">
   <label class="form-label">
        Book Category
        </label>
   <select name="book_category"
         class="form-control">
         <option value="0">
            Select category
         </option>
         <?php
         if ($categories == 0) {
            # Do nothing!
         }else{
         foreach ($categories as $category) {
            if ($book['category_id'] == $category['id']) { ?>
            <option
              selected
              value="<?=$category['id']?>">
              <?=$category['name']?>
            </option>
            <?php }else{ ?>
            <option
               value="<?=$category['id']?>">
               <?=$category['name']?>
            </option>
        <?php }} } ?>

        hidden
        value="<?=$book['cover']?>"
        name="current_cover">
        <a href="uploads/cover/<?=$book['cover']?>"
          class="link-dark">Current Cover</a>
     </div>

     <div class="mb-3">
        <label class="form-label">
             File
             </label>
        <input type="file"
             class="form-control"
             name="file">

         <input type="text"
             hidden
             value="<?=$book['file']?>"
             name="current_file">

        <a href="uploads/files/<?=$book['file']?>"
          class="link-dark">Current File</a>
     </div>

     <button type="submit"
           class="btn btn-primary">
           Update</button>
   </form>
  </div>
</body>
</html>

<?php }else{
 header("Location: login.php");
 exit;
} ?>
```

Edit-Category

```
<?php
session_start();

# If the admin is logged in
if (isset($_SESSION['user_id']) &&
  isset($_SESSION['user_email'])) {

  # If category ID is not set
  if (!isset($_GET['id'])) {
     #Redirect to admin.php page
     header("Location: admin.php");
```

```
        exit;
    }

    $id = $_GET['id'];

    # Database Connection File
    include "db_conn.php";

    # Category helper function
    include "php/func-category.php";
    $category = get_category($conn, $id);

    # If the ID is invalid
    if ($category == 0) {
        #Redirect to admin.php page
        header("Location: admin.php");
        exit;
    }
?>

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Edit Category</title>

    <!-- bootstrap 5 CDN-->
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.1/dist/css/bootstrap.min.css" rel="stylesheet"
integrity="sha384-F3w7mX95PdgyTmZZMECAngseQB83DfGTowi0iMjiWaeVhAn4FJkqJByhZMI3
AhiU" crossorigin="anonymous">

    <!-- bootstrap 5 Js bundle CDN-->
    <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.1/dist/js/bootstrap.bundle.min.js"
integrity="sha384-/bQdsTh/da6pkI1MST/rWKFNjaCP5gBSY4sEBT38Q/9RBh9AH40zEOg7Hlq2TH
RZ" crossorigin="anonymous"></script>

</head>
<body>
    <div class="container">
        <nav class="navbar navbar-expand-lg navbar-light bg-light">
          <div class="container-fluid">
            <a class="navbar-brand" href="admin.php">Admin</a>
            <button class="navbar-toggler" type="button" data-bs-toggle="collapse"
data-bs-target="#navbarSupportedContent" aria-controls="navbarSupportedContent"
aria-expanded="false" aria-label="Toggle navigation">
              <span class="navbar-toggler-icon"></span>
            </button>
```

```
      <div class="collapse navbar-collapse"
           id="navbarSupportedContent">
        <ul class="navbar-nav me-auto mb-2 mb-lg-0">
          <li class="nav-item">
            <a class="nav-link"
               aria-current="page"
               href="index.php">Store</a>
          </li>
          <li class="nav-item">
            <a class="nav-link"
               href="add-book.php">Add Book</a>
          </li>
          <li class="nav-item">
            <a class="nav-link"
               href="add-category.php">Add Category</a>
          </li>
          <li class="nav-item">
            <a class="nav-link"
               href="add-author.php">Add Author</a>
          </li>
          <li class="nav-item">
            <a class="nav-link"
               href="logout.php">Logout</a>
          </li>
        </ul>
      </div>
    </div>
  </nav>
<form action="php/edit-category.php"
      method="post"
      class="shadow p-4 rounded mt-5"
      style="width: 90%; max-width: 50rem;">

  <h1 class="text-center pb-5 display-4 fs-3">
      Edit Category
  </h1>
  <?php if (isset($_GET['error'])) { ?>
    <div class="alert alert-danger" role="alert">
        <?=htmlspecialchars($_GET['error']); ?>
    </div>
  <?php } ?>
  <?php if (isset($_GET['success'])) { ?>
    <div class="alert alert-success" role="alert">
        <?=htmlspecialchars($_GET['success']); ?>
    </div>
  <?php } ?>
  <div class="mb-3">
      <label class="form-label">
```

```
                 Category Name
                </label>

             <input type="text"
                 value="<?=$category['id'] ?>"
                 hidden
                 name="category_id">
            <input type="text"
                 class="form-control"
                 value="<?=$category['name'] ?>"
                 name="category_name">
        </div>
        <button type="submit"
               class="btn btn-primary">
               Update</button>
     </form>
   </div>
</body>
</html>
<?php }else{
 header("Location: login.php");
 exit;
} ?>
```

4.4 Back end Code:

A back-end database is a database that is accessed by users indirectly through an external application rather than by application programming stored within the database itself or by low level manipulation of the data. A back-end database stores data but does not include end-user application elements such as stored queries, forms, macros or reports.

4.4.1 DB Access and connectivity

Database is accessed using MySQL Connector for python .

```
mydb = mysql.connector.connect

            (

             host='localhost',

            user='root',

             password='PASSWORD',
                        database = 'DBMS_PROJECT'
                        )
```

4.4.2 Table Creation

Creating a basic table involves naming the table and defining its columns and each column's data type. The SQL **CREATE TABLE** statement is used to create a new table.

Syntax

The basic syntax of the CREATE TABLE statement is as follows −

```
 CREATE TABLE table_name(
     column1 datatype,
     column2 datatype,
     column3 datatype,
     .....
     columnN data type,
     PRIMARY KEY (one or more columns)
 );
```

CREATE TABLE is the keyword telling the database system what you want to do. In this case, you want to create a new table. The unique name or identifier for the table follows the CREATE TABLE statement.

Then in brackets comes the list defining each column in the table and what sort of data type it is.

The SQL **CREATE DATABASE** statement is used to create a new SQL database.

Syntax

The basic syntax of this CREATE DATABASE statement is as follows −

```
CREATE DATABASE DatabaseName;
```

The database name should be unique within the RDBMS.

1. Creation of new database named ONLINE_BOOK_STORE_DB

```
CREATE DATABASE ONLINE_BOOK_STORE_DB;
```

```
USE ONLINE_BOOK_STORE_DB;
```

2. Creation of table admin

```
CREATE TABLE admin (
 id  int(11) NOT NULL AUTO_INCREMENT,
 full_name varchar(255) NOT NULL,
 email varchar(255) NOT NULL,
 password text NOT NULL,
 PRIMARY KEY (id)
) ;
```

3. Creation of table authors

```
CREATE TABLE AUTHORS (
 id int(11) NOT NULL AUTO_INCREMENT,
 name varchar(255) NOT NULL,
 PRIMARY KEY (id)
);
```

4. Creation of table books

```
CREATE TABLE BOOKS (
 id int(11) NOT NULL AUTO_INCREMENT,
 title varchar(255) NOT NULL,
 author_id int(11) NOT NULL,
 description text NOT NULL,
 category_id int(11) NOT NULL,
 cover varchar(255) NOT NULL,
 file varchar(255) NOT NULL,
 PRIMARY KEY (id)
);
```

5. Creation of table categories

```
CREATE TABLE CATEGORIES (
 id int(11) NOT NULL AUTO_INCREMENT,
 name varchar(255) NOT NULL,
 PRIMARY KEY (`id`)
) ;
```

auth

```
<?php
session_start();

if (isset($_POST['email']) &&
    isset($_POST['password'])) {

         # Database Connection File
        include "../db_conn.php";


    # Validation helper function
    include "func-validation.php";

     /**
       Get data from POST request
       and store them in var
    **/

    $email = $_POST['email'];
    $password = $_POST['password'];

    # simple form validation
    $text = "Email";
    $location = "../login.php";
    $ms = "error";
    is_empty($email, $text, $location, $ms, "");

    $text = "Password";
    $location = "../login.php";
    $ms = "error";
    is_empty($password, $text, $location, $ms, "");

    # search for the email

    $sql = "SELECT * FROM admin
```

```
      WHERE email=?";
      $stmt = $conn->prepare($sql);
      $stmt->execute([$email]);

      #if the email is exist
      if($stmt->rowCount()===1){
        $user = $stmt->fetch();

        $user_id = $user['id'];
        $user_email = $user['email'];
        $user_password = $user['password'];
        if ($email === $user_email) {
          if (password_verify($password, $user_password)) {
            $_SESSION['user_id'] = $user_id;
            $_SESSION['user_email'] = $user_email;
            header("Location: ../admin.php");
          }else {
            # Error message
            $em = "Incorrect User name or password";
            header("Location: ../login.php?error=$em");
          }
      }else{
        #error message
        $em= "Incorrect username or password";
        header("Location: ../login.php?error=$em");
      }
      }else{
        #error message
        $em= "Incorrect username or password";
        header("Location: ../login.php?error=$em");
      }


      }else{
        #redirect to "../login.php"
        header("Location:../login.php");
      }
```

func-author

```
<?php
# Get all Author function
function get_all_author($con){
  $sql  = "SELECT * FROM authors";
  $stmt = $con->prepare($sql);
  $stmt->execute();
  if ($stmt->rowCount() > 0) {
       $authors = $stmt->fetchAll();
  }else {
     $authors = 0;
  }

  return $authors;
}

# Get  Author by ID function
function get_author($con, $id){
  $sql  = "SELECT * FROM authors WHERE id=?";
  $stmt = $con->prepare($sql);
  $stmt->execute([$id]);

  if ($stmt->rowCount() >0) {
       $author = $stmt->fetch();
  }else {
     $author = 0;
  }

  return $author;
}
```

<u>func-book</u>

```
<?php

# Get All books function
function get_all_books($con){
   $sql  = "SELECT * FROM books ORDER bY id DESC";
   $stmt = $con->prepare($sql);
   $stmt->execute();

   if ($stmt->rowCount() > 0) {
        $books = $stmt->fetchAll();
   }else {
      $books = 0;
   }

   return $books;
}

# Get  book by ID function
function get_book($con, $id){
   $sql  = "SELECT * FROM books WHERE id=?";
   $stmt = $con->prepare($sql);
   $stmt->execute([$id]);
   if ($stmt->rowCount() > 0) {
        $book = $stmt->fetch();
   }else {
      $book = 0;
   }

   return $book;
}

# Search books function
function search_books($con, $key){
   # creating simple search algorithm :)
   $key = "%{$key}%";

   $sql  = "SELECT * FROM books
            WHERE title LIKE ?
            OR description LIKE ?";
   $stmt = $con->prepare($sql);
   $stmt->execute([$key, $key]);

   if ($stmt->rowCount() > 0) {
        $books = $stmt->fetchAll();
   }else {
      $books = 0;
   }

   return $books;
}

# get books by category
function get_books_by_category($con, $id){
   $sql  = "SELECT * FROM books WHERE category_id=?";
   $stmt = $con->prepare($sql);
   $stmt->execute([$id]);

   if ($stmt->rowCount() > 0) {
        $books = $stmt->fetchAll();
   }else {
      $books = 0;
   }

   return $books;
}


# get books by author
function get_books_by_author($con, $id){
   $sql  = "SELECT * FROM books WHERE author_id=?";
   $stmt = $con->prepare($sql);
   $stmt->execute([$id]);

   if ($stmt->rowCount() > 0) {
        $books = $stmt->fetchAll();
   }else {
      $books = 0;
   }

   return $books;
}
```

func-category

```
<?php

# Get all Categories function
function get_all_categories($con){
  $sql  = "SELECT * FROM categories";
  $stmt = $con->prepare($sql);
  $stmt->execute();

  if ($stmt->rowCount() > 0) {
       $categories = $stmt->fetchAll();
  }else {
     $categories = 0;
  }

  return $categories;
}


# Get category by ID
function get_category($con, $id){
  $sql  = "SELECT * FROM categories WHERE id=?";
  $stmt = $con->prepare($sql);
  $stmt->execute([$id]);

  if ($stmt->rowCount() > 0) {
       $category = $stmt->fetch();
  }else {
     $category = 0;
  }

  return $category;
}
```

func-file upload

```
<?php

# File upload helper function
function upload_file($files, $allowed_exs, $path){
  # get data and store them in var
  $file_name = $files['name'];
  $tmp_name  = $files['tmp_name'];
  $error     = $files['error'];

  # if there is no error occurred while uploading
  if ($error === 0) {

     # get file extension store it in var
     $file_ex = pathinfo($file_name, PATHINFO_EXTENSION);

     /**
       convert the file extension into lower case
       and store it in var
     **/
     $file_ex_lc = strtolower($file_ex);

     /**
       check if the the file extension
       exist in $allowed_exs array
     **/
       if (in_array($file_ex_lc, $allowed_exs)) {
          /**
          renaming the  file
          with random strings
          **/
          $new_file_name = uniqid("",true).'.'.$file_ex_lc;

          # assigning upload path
          $file_upload_path = '../uploads/'.$path.'/'.$new_file_name;
          /**
            moving uploaded file to
            root directory upload/$path folder
          **/
          move_uploaded_file($tmp_name, $file_upload_path);

          /**
             Creating success message associative array
             with named keys status and data
          **/
          $sm['status'] = 'success';
          $sm['data']   = $new_file_name;
          #  Return the sm array
          return $sm;

       }else{
        /**
          Creating error message associative array
          with named keys status and data
        **/
         $em['status'] = 'error';
         $em['data']   = "You can't upload files of this type";

         #  Return the em array
         return $em;
       }
  }else {
    /**
      Creating error message associative array
     with named keys status and data
    **/
     $em['status'] = 'error';
     $em['data']   = 'Error occurred while uploading!';

     #  Return the em array
     return $em;
  }
}
```

func-validation

```
<?php
# Form validation function
function is_empty($var, $text, $location, $ms, $data){
   if (empty($var)) {
     # Error message
     $em = "The ".$text." is required";
     header("Location: $location?$ms=$em&$data");
     exit;
   }
   return 0;
}
```

add-author

```
/**
      check if author
      name is submitted
    **/
    if (isset($_POST['author_name'])) {
        /**
        Get data from POST request
        and store it in var
        **/
        $name = $_POST['author_name'];

        #simple form Validation
        if (empty($name)) {
            $em = "The author name is required";
            header("Location: ../add-author.php?error=$em");
            exit;
        }else {
            # Insert Into Database
            $sql  = "INSERT INTO authors (name)
                     VALUES (?)";
            $stmt = $conn->prepare($sql);
            $res  = $stmt->execute([$name]);

            /**
              If there is no error while
              inserting the data
            **/
             if ($res) {
                # success message
                $sm = "Successfully created!";
                header("Location: ../add-author.php?success=$sm");
                exit;
             }else{
                # Error message
                $em = "Unknown Error Occurred!";
                header("Location: ../add-author.php?error=$em");
                exit;
             }
        }
    }else {
      header("Location: ../admin.php");
      exit;
    }

}else{
  header("Location: ../login.php");
  exit;
}
```

add-book

```
<?php
session_start();

# If the admin is logged in
if (isset($_SESSION['user_id']) &&
   isset($_SESSION['user_email'])) {

   # Database Connection File
   include "../db_conn.php";

   # Validation helper function
   include "func-validation.php";

   # File Upload helper function
   include "func-file-upload.php";


   /**
     If all Input field
     are filled
   **/
   if (isset($_POST['book_title'])       &&
       isset($_POST['book_description']) &&
       isset($_POST['book_author'])      &&
       isset($_POST['book_category'])    &&
       isset($_FILES['book_cover'])      &&
       isset($_FILES['file'])) {
       /**
       Get data from POST request
       and store them in var
       **/
       $title       = $_POST['book_title'];
       $description = $_POST['book_description'];
       $author      = $_POST['book_author'];
       $category    = $_POST['book_category'];

       # making URL data format
       $user_input = 'title='.$title.'&category_id='.$category.'desc='.$description.'&author_id='.$author;

       #simple form Validation

       $text = "Book title";
       $location = "../add-book.php";
       $ms = "error";
       is_empty($title, $text, $location, $ms, $user_input);
```

```
$text = "Book description";
$location = "../add-book.php";
$ms = "error";
is_empty($description, $text, $location, $ms, $user_input);
$text = "Book author";
$location = "../add-book.php";
$ms = "error";
is_empty($author, $text, $location, $ms, $user_input);

$text = "Book category";
$location = "../add-book.php";
$ms = "error";
is_empty($category, $text, $location, $ms, $user_input);

# book cover Uploading
$allowed_image_exs = array("jpg", "jpeg", "png");
$path = "cover";
$book_cover = upload_file($_FILES['book_cover'], $allowed_image_exs, $path);

/**
If error occurred while
uploading the book cover
**/
if ($book_cover['status'] == "error") {
   $em = $book_cover['data'];

   /**
     Redirect to '../add-book.php'
     and passing error message & user_input
   **/
   header("Location: ../add-book.php?error=$em&$user_input");
   exit;
}else {
   # file Uploading
   $allowed_file_exs = array("pdf", "docx", "pptx");
   $path = "files";
   $file = upload_file($_FILES['file'], $allowed_file_exs, $path);

   /**
   If error occurred while
   uploading the file
   **/
   if ($file['status'] == "error") {
      $em = $file['data'];

      /**
        Redirect to '../add-book.php'
        and passing error message & user_input
```

```
            **/
            header("Location: ../add-book.php?error=$em&$user_input");
            exit;
          }else {
            /**
              Getting the new file name
              and book cover name
            **/
      \
            $book_cover_URL = $book_cover['data'];

            # Insert the data into database
            $sql  = "INSERT INTO books (title,
                                  author_id,
                                  description,
                                  category_id,
                                  cover,
                                  file)
                        VALUES (?,?,?,?,?,?)";
            $stmt = $conn->prepare($sql);
            $res  = $stmt->execute([$title, $author, $description, $category, $book_cover_URL,
$file_URL]);

      }else {
        header("Location: ../admin.php");
        exit;
      }

}else{
 header("Location: ../login.php");
 exit;
}
```

add-category

```
<?php
session_start();

# If the admin is logged in
if (isset($_SESSION['user_id']) &&
    isset($_SESSION['user_email'])) {

    # Database Connection File
    include "../db_conn.php";


    /**
```

```
      check if category
      name is submitted
    **/
    if (isset($_POST['category_name'])) {
        /**
        Get data from POST request
        and store it in var
        **/
        $name = $_POST['category_name'];

        #simple form Validation
        if (empty($name)) {
            $em = "The category name is required";
            header("Location: ../add-category.php?error=$em");
            exit;
        }else {
            # Insert Into Database
            $sql  = "INSERT INTO categories (name)
                     VALUES (?)";
            $stmt = $conn->prepare($sql);
            $res  = $stmt->execute([$name]);

            /**
              If there is no error while
              inserting the data
            **/
             if ($res) {
                # success message
                $sm = "Successfully created!";
                header("Location: ../add-category.php?success=$sm");
                exit;
             }else{
                # Error message
                $em = "Unknown Error Occurred!";
                header("Location: ../add-category.php?error=$em");
                exit;
             }
        }
    }else {
      header("Location: ../admin.php");
      exit;
    }

}else{
  header("Location: ../login.php");
  exit;
}
```

edit-author

```
<?php
session_start();

# If the admin is logged in
if (isset($_SESSION['user_id']) &&
    isset($_SESSION['user_email'])) {

    # Database Connection File
    include "../db_conn.php";


    /**
      check if author
      name is submitted
    **/
    if (isset($_POST['author_name']) &&
        isset($_POST['author_id'])) {
        /**
        Get data from POST request
        and store them in var
        **/
        $name = $_POST['author_name'];
        $id = $_POST['author_id'];

        #simple form Validation
        if (empty($name)) {
            $em = "The author name is required";
            header("Location: ../edit-author.php?error=$em&id=$id");
            exit;
        }else {
            # UPDATE the Database
            $sql  = "UPDATE authors
                     SET name=?
                     WHERE id=?";
            $stmt = $conn->prepare($sql);
            $res  = $stmt->execute([$name, $id]);

            /**
              If there is no error while
              inserting the data
            **/
             if ($res) {
                # success message
                $sm = "Successfully updated!";
                header("Location: ../edit-author.php?success=$sm&id=$id");
                exit;
```

```
         }else{
           # Error message
           $em = "Unknown Error Occurred!";
           header("Location: ../edit-author.php?error=$em&id=$id");
           exit;
         }
      }
   }else {
    header("Location: ../admin.php");
    exit;
   }

}else{
 header("Location: ../login.php");
 exit;
}
```

edit-book

```
<?php
session_start();

# If the admin is logged in
if (isset($_SESSION['user_id']) &&
  isset($_SESSION['user_email'])) {

  # Database Connection File
  include "../db_conn.php";

  # Validation helper function
  include "func-validation.php";

  # File Upload helper function
  include "func-file-upload.php";

  /**
   If all Input field
   are filled
  **/
  if (isset($_POST['book_id'])          &&
     isset($_POST['book_title'])       &&
     isset($_POST['book_description']) &&
     isset($_POST['book_author'])      &&
     isset($_POST['book_category'])    &&
     isset($_FILES['book_cover'])      &&
     isset($_FILES['file'])            &&
     isset($_POST['current_cover'])    &&
     isset($_POST['current_file'])) {

     /**
     Get data from POST request
     and store them in var
     **/
     $id         = $_POST['book_id'];
     $title      = $_POST['book_title'];
     $description = $_POST['book_description'];
     $author     = $_POST['book_author'];
     $category   = $_POST['book_category'];

      /**
       Get current cover & current file
       from POST request and store them in var
     **/

     $current_cover = $_POST['current_cover'];
     $current_file  = $_POST['current_file'];

     #simple form Validation
     $text = "Book title";
     $location = "../edit-book.php";
     $ms = "id=$id error";
     is_empty($title, $text, $location, $ms, "");

     $text = "Book description";
     $location = "../edit-book.php";
     $ms = "id=$id&error";
     is_empty($description, $text, $location, $ms, "");

     $text = "Book author";
     $location = "../edit-book.php";
     $ms = "id=$id&error";
     is_empty($author, $text, $location, $ms, "");

     $text = "Book category";
     $location = "../edit-book.php";
     $ms = "id=$id&error";
     is_empty($category, $text, $location, $ms, "");

     /**
       if the admin try to
       update the book cover
     **/
       if (!empty($_FILES['book_cover']['name'])) {
           /**
               if the admin try to
```

```
    update both
  **/
 if (!empty($_FILES['file']['name'])) {
   # update both here

   # book cover Uploading
   $allowed_image_exs = array("jpg", "jpeg", "png");
   $path = "cover";
   $book_cover = upload_file($_FILES['book_cover'], $allowed_image_exs, $path);

   # book cover Uploading
   $allowed_file_exs = array("pdf", "docx", "pptx");
   $path = "files";
   $file = upload_file($_FILES['file'], $allowed_file_exs, $path);

   /**
      If error occurred while
      uploading
   **/
   if ($book_cover['status'] == "error" ||
      $file['status'] == "error") {

      $em = $book_cover['data'];

      /**
        Redirect to '../edit-book.php'
        and passing error message & the id
      **/
      header("Location: ../edit-book.php?error=$em&id=$id");
      exit;
   }else {
     # current book cover path
     $c_p_book_cover = "../uploads/cover/$current_cover";

     # current file path
     $c_p_file = "../uploads/files/$current_file";

     # Delete from the server
     unlink($c_p_book_cover);
     unlink($c_p_file);

     /**
        Getting the new file name
        and the new book cover name
     **/
      $file_URL = $file['data'];
      $book_cover_URL = $book_cover['data'];
```

```
                    # update just the data
                    $sql = "UPDATE books
                            SET title=?,
                                author_id=?,
                                description=?,
                                category_id=?,
                                cover=?,
                                file=?
                            WHERE id=?";
                    $stmt = $conn->prepare($sql);
                    $res  = $stmt->execute([$title, $author, $description, $category,$book_cover_URL,
$file_URL, $id]);

                    /**
                      If there is no error while
                      updating the data
                    **/
                     if ($res) {
                        # success message
                        $sm = "Successfully updated!";
                        header("Location: ../edit-book.php?success=$sm&id=$id");
                        exit;
                     }else{
                        # Error message
                        $em = "Unknown Error Occurred!";
                        header("Location: ../edit-book.php?error=$em&id=$id");
                        exit;
                     }
                }
              }else {
                # update just the book cover

                # book cover Uploading
                $allowed_image_exs = array("jpg", "jpeg", "png");
                $path = "cover";
                $book_cover = upload_file($_FILES['book_cover'], $allowed_image_exs, $path);

                /**
                    If error occurred while
                    uploading
                **/
                if ($book_cover['status'] == "error") {

                    $em = $book_cover['data'];

                    /**
                      Redirect to '../edit-book.php'
                      and passing error message & the id
```

```
          **/
          header("Location: ../edit-book.php?error=$em&id=$id");
          exit;
        }else {
         # current book cover path
         $c_p_book_cover = "../uploads/cover/$current_cover";

         # Delete from the server
         unlink($c_p_book_cover);

         /**
            Getting the new file name
            and the new book cover name
         **/
          $book_cover_URL = $book_cover['data'];

           # update just the data
           $sql = "UPDATE books
                  SET title=?,
                      author_id=?,
                      description=?,
                      category_id=?,
                      cover=?
                  WHERE id=?";
           $stmt = $conn->prepare($sql);
           $res  = $stmt->execute([$title, $author, $description, $category,$book_cover_URL, $id]);

           /**
             If there is no error while
             updating the data
           **/
            if ($res) {
              # success message
              $sm = "Successfully updated!";
              header("Location: ../edit-book.php?success=$sm&id=$id");
              exit;
            }else{
              # Error message
              $em = "Unknown Error Occurred!";
              header("Location: ../edit-book.php?error=$em&id=$id");
              exit;
            }
        }
      }
    }
    /**
    if the admin try to
    update just the file
```

```
**/
else if(!empty($_FILES['file']['name'])){
  # update just the file

  # book cover Uploading
  $allowed_file_exs = array("pdf", "docx", "pptx");
  $path = "files";
  $file = upload_file($_FILES['file'], $allowed_file_exs, $path);

  /**
      If error occurred while
      uploading
  **/
  if ($file['status'] == "error") {

      $em = $file['data'];

      /**
        Redirect to '../edit-book.php'
        and passing error message & the id
      **/
      header("Location: ../edit-book.php?error=$em&id=$id");
      exit;
  }else {
    # current book cover path
    $c_p_file = "../uploads/files/$current_file";

    # Delete from the server
    unlink($c_p_file);

    /**
        Getting the new file name
        and the new file name
    **/
     $file_URL = $file['data'];

      # update just the data
      $sql = "UPDATE books
              SET title=?,
                  author_id=?,
                  description=?,
                  category_id=?,
                   file=?
              WHERE id=?";
      $stmt = $conn->prepare($sql);
      $res  = $stmt->execute([$title, $author, $description, $category, $file_URL, $id]);
      /**
       If there is no error while
       updating the data
      **/
       if ($res) {
         # success message
         $sm = "Successfully updated!";
         header("Location: ../edit-book.php?success=$sm&id=$id");
         exit;
       }else{
         # Error message
         $em = "Unknown Error Occurred!";
         header("Location: ../edit-book.php?error=$em&id=$id");
         exit;
       }
    }

   }else {
    # update just the data
    $sql = "UPDATE books
         SET title=?,
           author_id=?,
           description=?,
           category_id=?
         WHERE id=?";
    $stmt = $conn->prepare($sql);
    $res  = $stmt->execute([$title, $author, $description, $category, $id]);

    /**
     If there is no error while
     updating the data
    **/
     if ($res) {
       # success message
       $sm = "Successfully updated!";
       header("Location: ../edit-book.php?success=$sm&id=$id");
       exit;
     }else{
       # Error message
       $em = "Unknown Error Occurred!";
       header("Location: ../edit-book.php?error=$em&id=$id");
       exit;
     }
   }
}else {
 header("Location: ../admin.php");
 exit;
}
```

```
}else{
 header("Location: ../login.php");
 exit;
}
```

edit-category

```
<?php
session_start();

# If the admin is logged in
if (isset($_SESSION['user_id']) &&
    isset($_SESSION['user_email'])) {

    # Database Connection File
    include "../db_conn.php";
    /**
      check if category
      name is submitted
    **/
    if (isset($_POST['category_name']) &&
        isset($_POST['category_id'])) {
        /**
        Get data from POST request
        and store them in var
        **/
        $name = $_POST['category_name'];
        $id = $_POST['category_id'];

        #simple form Validation
        if (empty($name)) {
            $em = "The category name is required";
            header("Location: ../edit-category.php?error=$em&id=$id");
            exit;
        }else {
            # UPDATE the Database
            $sql  = "UPDATE categories
                     SET name=?
                     WHERE id=?";
            $stmt = $conn->prepare($sql);
            $res  = $stmt->execute([$name, $id]);

            /**
              If there is no error while
              updating the data
            **/
      if ($res) {
        # success message
        $sm = "Successfully updated!";
        header("Location: ../edit-category.php?success=$sm&id=$id");
        exit;
      }else{
        # Error message
        $em = "Unknown Error Occurred!";
        header("Location: ../edit-category.php?error=$em&id=$id");
        exit;
      }
    }
  }else {
   header("Location: ../admin.php");
   exit;
  }

}else{
 header("Location: ../login.php");
 exit;
}
```

delete-author

```
<?php
session_start();

# If the admin is logged in
if (isset($_SESSION['user_id']) &&
  isset($_SESSION['user_email'])) {

  # Database Connection File
  include "../db_conn.php";

  /**
   check if the author
   id set
  **/
  if (isset($_GET['id'])) {
    /**
    Get data from GET request
    and store it in var
    **/
    $id = $_GET['id'];

    #simple form Validation
    if (empty($id)) {
      $em = "Error Occurred!";
```

```
            header("Location: ../admin.php?error=$em");
            exit;
        }else {
            # DELETE the category from Database
            $sql  = "DELETE FROM authors
                     WHERE id=?";
            $stmt = $conn->prepare($sql);
            $res  = $stmt->execute([$id]);

            /**
              If there is no error while
              Deleting the data
            **/
             if ($res) {
                # success message
                $sm = "Successfully removed!";
                header("Location: ../admin.php?success=$sm");
                exit;
             }else {
                $em = "Error Occurred!";
                header("Location: ../admin.php?error=$em");
                exit;
             }

        }
    }else {
      header("Location: ../admin.php");
      exit;
    }

}else{
  header("Location: ../login.php");
  exit;
}
```

delete book

```
<?php
session_start();

# If the admin is logged in
if (isset($_SESSION['user_id']) &&
    isset($_SESSION['user_email'])) {

    # Database Connection File
    include "../db_conn.php";
```

```
/**
  check if the book
  id set
**/
if (isset($_GET['id'])) {
    /**
    Get data from GET request
    and store it in var
    **/
    $id = $_GET['id'];

    #simple form Validation
    if (empty($id)) {
        $em = "Error Occurred!";
        header("Location: ../admin.php?error=$em");
        exit;
    }else {
         # GET book from Database
         $sql2  = "SELECT * FROM books
                  WHERE id=?";
         $stmt2 = $conn->prepare($sql2);
         $stmt2->execute([$id]);
         $the_book = $stmt2->fetch();

         if($stmt2->rowCount() > 0){
            # DELETE the book from Database
            $sql  = "DELETE FROM books
                     WHERE id=?";
            $stmt = $conn->prepare($sql);
            $res  = $stmt->execute([$id]);

            /**
              If there is no error while
              Deleting the data
            **/
             if ($res) {
                # delete the current book_cover and the file
                $cover = $the_book['cover'];
                $file  = $the_book['file'];
                $c_b_c = "../uploads/cover/$cover";
                $c_f = "../uploads/files/$cover";

                unlink($c_b_c);
                unlink($c_f);


                # success message
```

```
          $sm = "Successfully removed!";
          header("Location: ../admin.php?success=$sm");
          exit;
        }else{
          # Error message
          $em = "Unknown Error Occurred!";
          header("Location: ../admin.php?error=$em");
          exit;
        }
      }else {
        $em = "Error Occurred!";
        header("Location: ../admin.php?error=$em");
        exit;
      }

    }
  }else {
    header("Location: ../admin.php");
    exit;
  }
}else{
  header("Location: ../login.php");
  exit;
}
```

delete-category

```
<?php
session_start();

# If the admin is logged in
if (isset($_SESSION['user_id']) &&
  isset($_SESSION['user_email'])) {

  # Database Connection File
  include "../db_conn.php";


  /**
   check if the category
   id set
  **/
  if (isset($_GET['id'])) {
    /**
    Get data from GET request
    and store it in var
    **/
    $id = $_GET['id'];
```

```
        #simple form Validation
        if (empty($id)) {
           $em = "Error Occurred!";
           header("Location: ../admin.php?error=$em");
           exit;
        }else {
           # DELETE the category from Database
           $sql  = "DELETE FROM categories
                  WHERE id=?";
           $stmt = $conn->prepare($sql);
           $res  = $stmt->execute([$id]);
           /**
             If there is no error while
             Deleting the data
           **/
            if ($res) {
              # success message
              $sm = "Successfully removed!";
              header("Location: ../admin.php?success=$sm");
              exit;
            }else {
              $em = "Error Occurred!";
              header("Location: ../admin.php?error=$em");
              exit;
            }

        }
     }else {
       header("Location: ../admin.php");
       exit;
     }
}else{
  header("Location: ../login.php");
  exit;
}
```

4.4.3 TRIGGER

1.Trigger to add authors to authors_backup table

```
CREATE TRIGGER added_author

AFTER INSERT ON authors

FOR EACH ROW
```

```
BEGIN
INSERT INTO authors_backup
values(null,NEW.name);
end
```

2.Trigger to add books to books_backup table

```
CREATE TRIGGER added_book
AFTER INSERT  ON books
FOR EACH ROW
BEGIN
INSERT into books_backup
Values(null,NEW.title,NEW.author_id,NEW.description,NEW.category_id,NEW.cover,NEW.file);
end
```

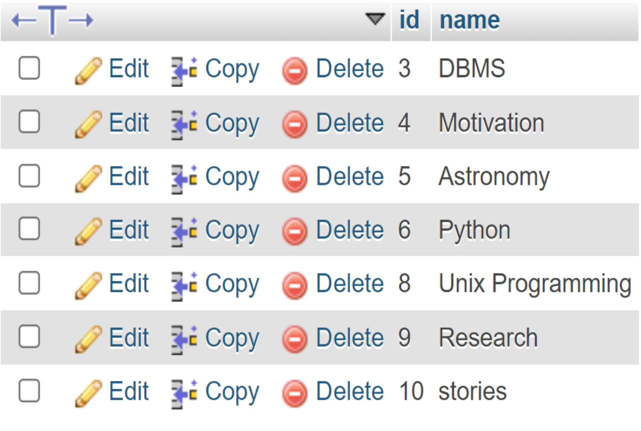
3.Trigger to add category to categories_backup table

```
CREATE TRIGGER added_category
AFTER INSERT ON categories
FOR EACH ROW
BEGIN
INSERT INTO categories_backup VALUES(null,NEW.name);
end
```

4.Trigger to add admin to admin_backup table

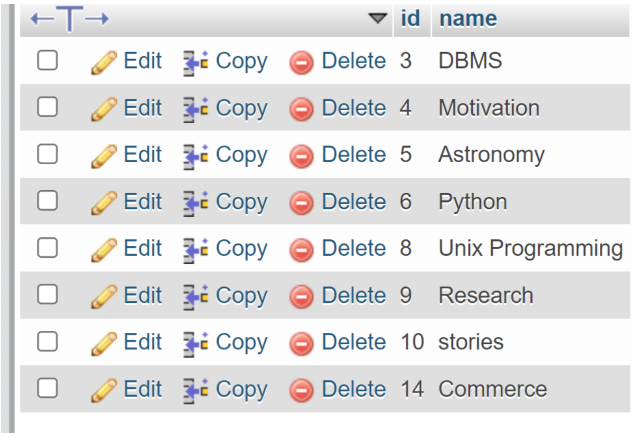
```
CREATE TRIGGER added_admin
AFTER INSERT ON admin
FOR EACH ROW
BEGIN INSERT INTO admin_backup VALUES(null,NEW.full_name,NEW.email,NEW.password);
End
```

4.4.4 SNAPSHOTS OF TRIGGERS USED



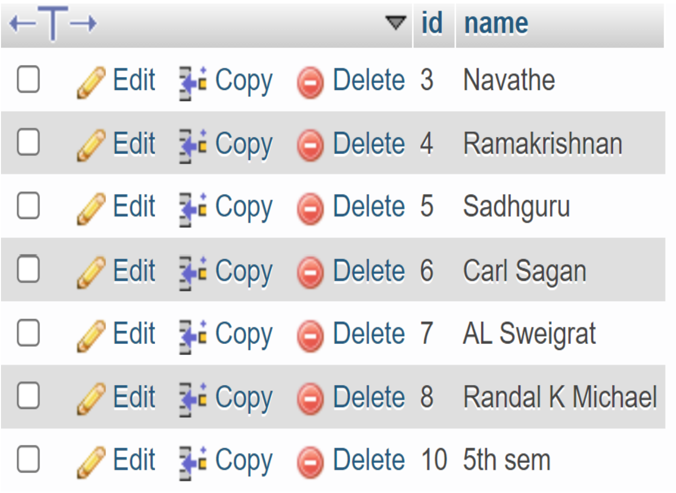
←T→			▼	id	name
□	Edit	Copy	Delete	3	DBMS
□	Edit	Copy	Delete	4	Motivation
□	Edit	Copy	Delete	5	Astronomy
□	Edit	Copy	Delete	6	Python
□	Edit	Copy	Delete	8	Unix Programming
□	Edit	Copy	Delete	9	Research
□	Edit	Copy	Delete	10	stories

Fig 4.1 Category table



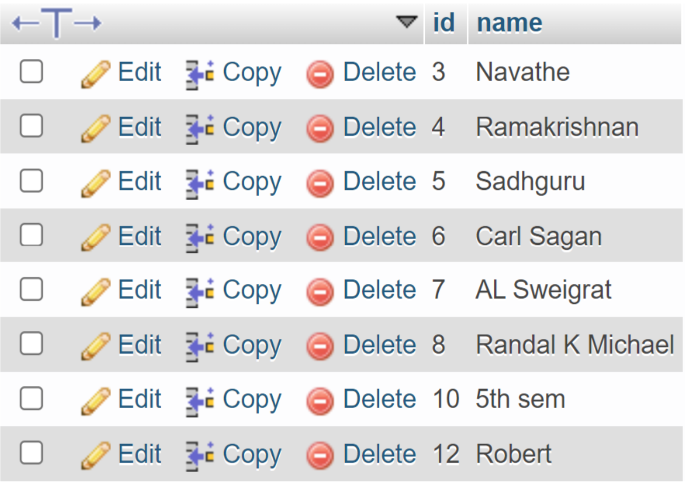
←T→			▼	id	name
□	Edit	Copy	Delete	3	DBMS
□	Edit	Copy	Delete	4	Motivation
□	Edit	Copy	Delete	5	Astronomy
□	Edit	Copy	Delete	6	Python
□	Edit	Copy	Delete	8	Unix Programming
□	Edit	Copy	Delete	9	Research
□	Edit	Copy	Delete	10	stories
□	Edit	Copy	Delete	14	Commerce

Fig 4.2 Category_backup table



←T→			▼	id	name
□	Edit	Copy	Delete	3	Navathe
□	Edit	Copy	Delete	4	Ramakrishnan
□	Edit	Copy	Delete	5	Sadhguru
□	Edit	Copy	Delete	6	Carl Sagan
□	Edit	Copy	Delete	7	AL Sweigrat
□	Edit	Copy	Delete	8	Randal K Michael
□	Edit	Copy	Delete	10	5th sem

Fig 4.3 author table



←T→			▼	id	name
□	Edit	Copy	Delete	3	Navathe
□	Edit	Copy	Delete	4	Ramakrishnan
□	Edit	Copy	Delete	5	Sadhguru
□	Edit	Copy	Delete	6	Carl Sagan
□	Edit	Copy	Delete	7	AL Sweigrat
□	Edit	Copy	Delete	8	Randal K Michael
□	Edit	Copy	Delete	10	5th sem
□	Edit	Copy	Delete	12	Robert

Fig 4.4 author_backup table

5. SOFTWARE TESTING

Software testing is the process of evaluating and verifying that a software product or application does what it is supposed to do. The benefits of testing include preventing bugs, reducing development costs and improving performance**.**

5.1 Introduction to Testing

Software Testing is a method to check whether the actual software product matches expected requirements and to ensure that the software product is Defect free. It involves execution of software/system components using manual or automated tools to evaluate one or more properties of interest. The purpose of software testing is to identify errors, gaps or missing requirements in contrast to actual requirements.

5.2 Testing Objectives:

- Testing is the process of executing a program with the intent of finding an error.
- A good test case design is one that has a probability of finding an as yet undiscovered error.
- A successful test is one that uncovers an as yet undiscovered error.
- These above objectives imply a dramatic change in view port.
- Testing cannot show the absence of defects, it can only show that software errors are present.

5.3 Testing Principles:

- All tests should be traceable to end user requirements.
- Tests should be planned long before testing begins.
- Testing should begin on a small scale and progress towards testing in large.
- Exhaustive testing is not possible.

5.4 Validation Testing

The process of evaluating software during the development process or at the end of the development process to determine whether it satisfies specified business requirements. Validation Testing ensures that the product actually meets the client's needs. It can also be defined as to demonstrate that the product fulfills its intended use when deployed in an appropriate environment.

5.5 Unit Testing:

In this test the entire system is divided into smaller entities and each entity is known as a unit and each unit is tested one at a time. The test ensures the working of each module as per the requirement in functionality point of view rather than actual module visible to the user.

5.6 Integration Testing:

In this test each unit is looped together to form a long and continuous chain of modules till it does not cover the entire system. The test checks the behavior of the interface. This test is generally performed after a unit test.

5.7 Test Cases

A Test Case is a set of actions executed to verify a particular feature or functionality of your software application. A Test Case contains test steps, test data, precondition, postcondition developed for specific test scenarios to verify any requirement.

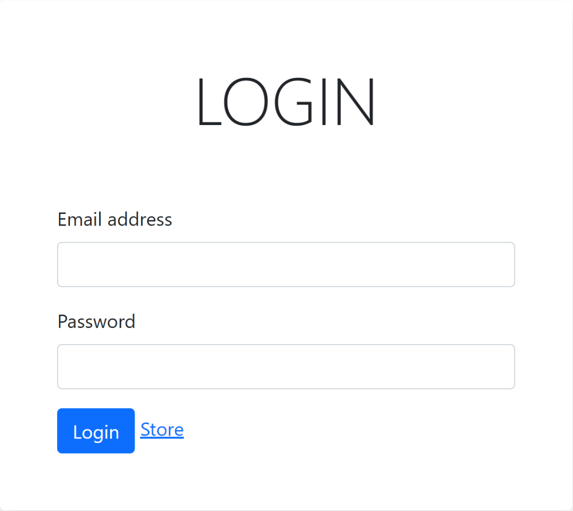
Table 5.1: Test Cases

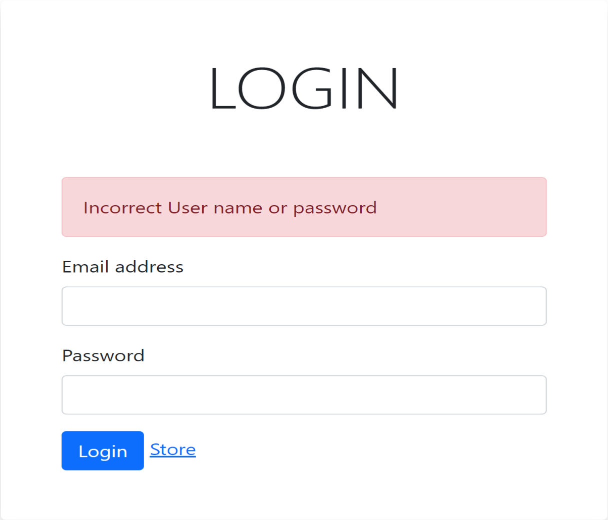
Test No.	Test Name	Actual OOutput	Expected Output	Status
1	login	Successfully logged in	Successfully logged in	pass
2	Searching	Books details must be shown	Books details must be shown	pass
3	Adding new data	New data created	New data created	pass
4	Deleting the data	Data deleted	Data deleted	pass
5	Updating the data	Data updated	Data updated	pass

5.8 Regression Testing:

This test is performed after the integrating test where the system is checked if it gives the desired output for the expected input data from the user.

1.When we enter wrong user details

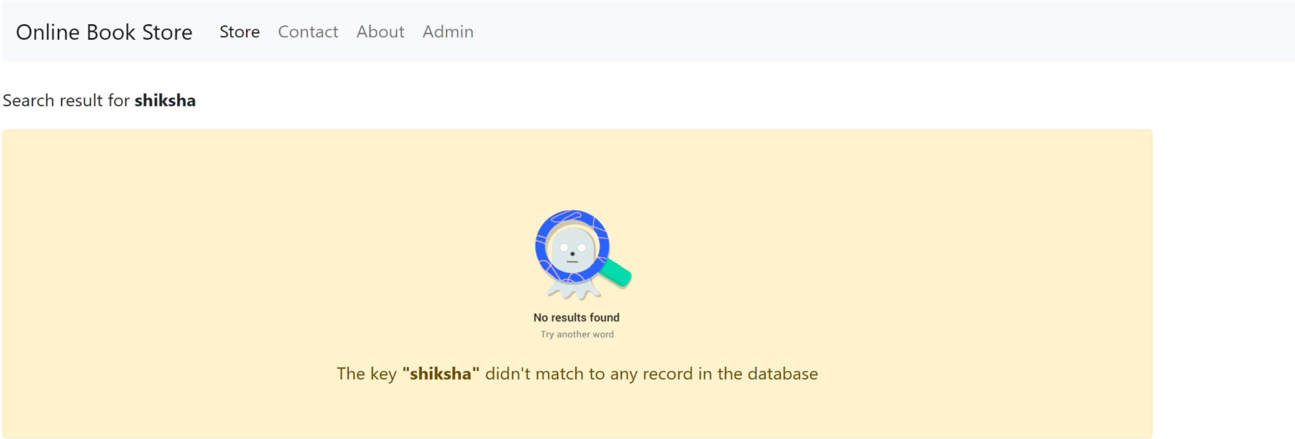




Incorrect User name or password

Fig 5.1: Login Page **Fig 5.2: Login Page Error**

2. When search fails



Search result for **shiksha**

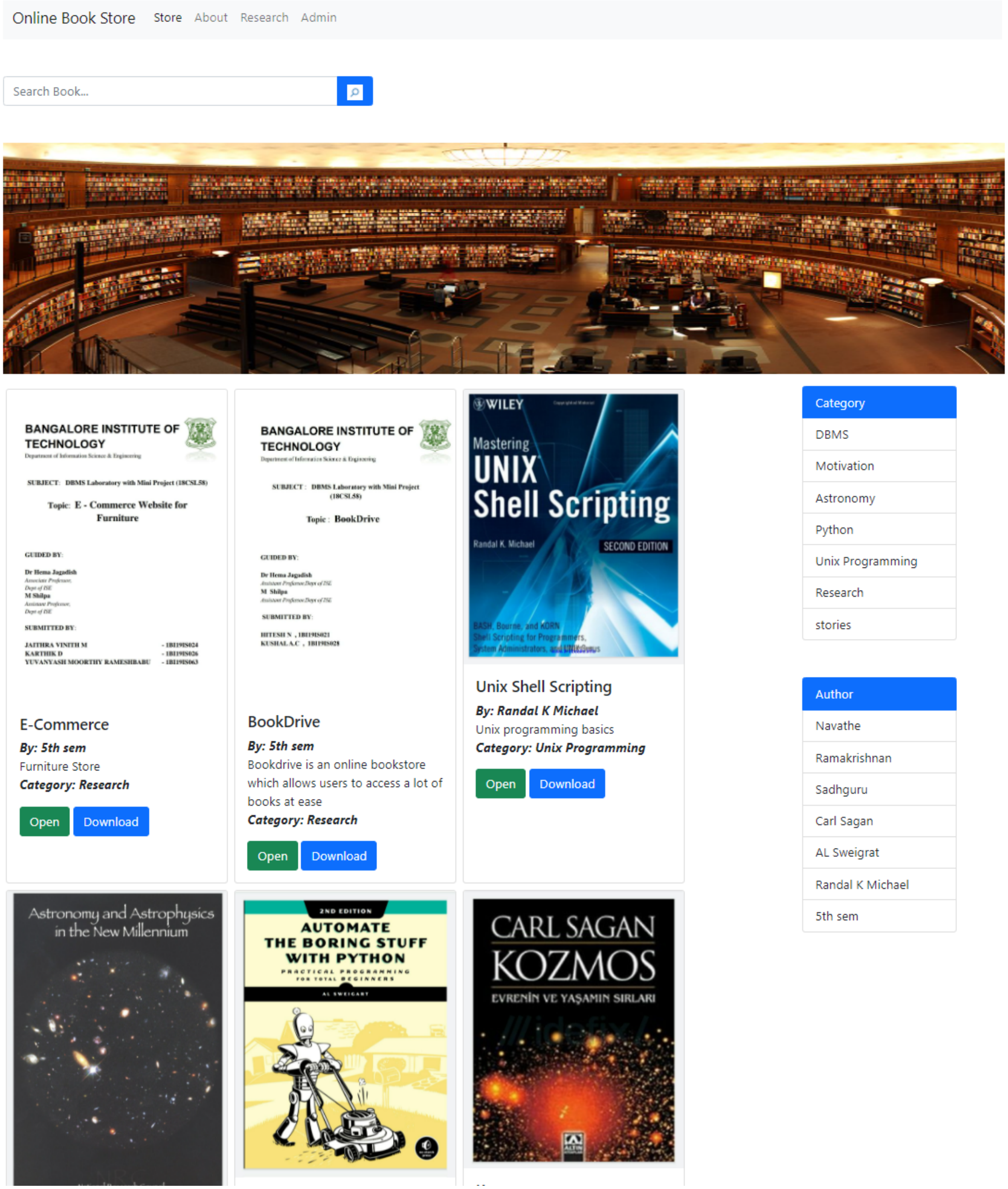
No results found

Try another word

The key **"shiksha"** didn't match to any record in the database

Fig 5.3 : Search Fails

6. SNAPSHOTS



E-Commerce

By: 5th sem

Furniture Store

Category: Research

BookDrive

By: 5th sem

Bookdrive is an online bookstore which allows users to access a lot of books at ease

Category: Research

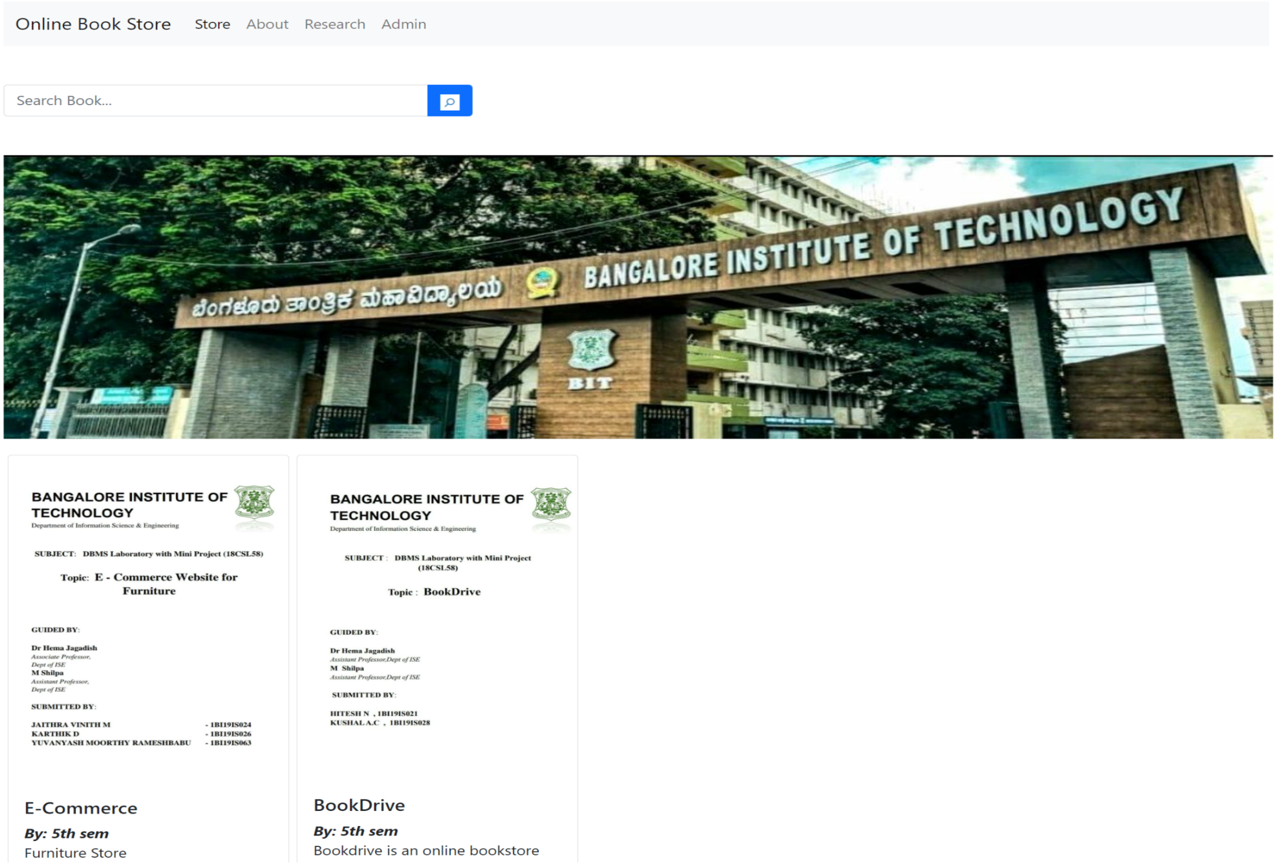
Unix Shell Scripting

By: Randal K Michael

Unix programming basics

Category: Unix Programming

Fig 6.1 Home Page


Online Book Store
Store
About
Research
Admin
Search Book...
BANGALORE INSTITUTE OF TECHNOLOGY
ಬೆಂಗಳೂರು ತಾಂತ್ರಿಕ ಮಹಾವಿದ್ಯಾಲಯ
BIT

BANGALORE INSTITUTE OF TECHNOLOGY
Department of Information Science & Engineering

SUBJECT: DBMS Laboratory with Mini Project (18CSL58)

Topic: E - Commerce Website for Furniture

GUIDED BY:

Dr Hema Jagadish
Associate Professor,
Dept of ISE
M Shilpa
Assistant Professor,
Dept of ISE

SUBMITTED BY:

JAITHRA VINITH M	- 1BI19IS024
KARTHIK D	- 1BI19IS026
YUVANYASH MOORTHY RAMESHBABU	- 1BI19IS063

E-Commerce

By: 5th sem

Furniture Store

BANGALORE INSTITUTE OF TECHNOLOGY
Department of Information Science & Engineering

SUBJECT : DBMS Laboratory with Mini Project (18CSL58)

Topic : BookDrive

GUIDED BY:

Dr Hema Jagadish
Assistant Professor,Dept of ISE
M Shilpa
Assistant Professor,Dept of ISE

SUBMITTED BY:

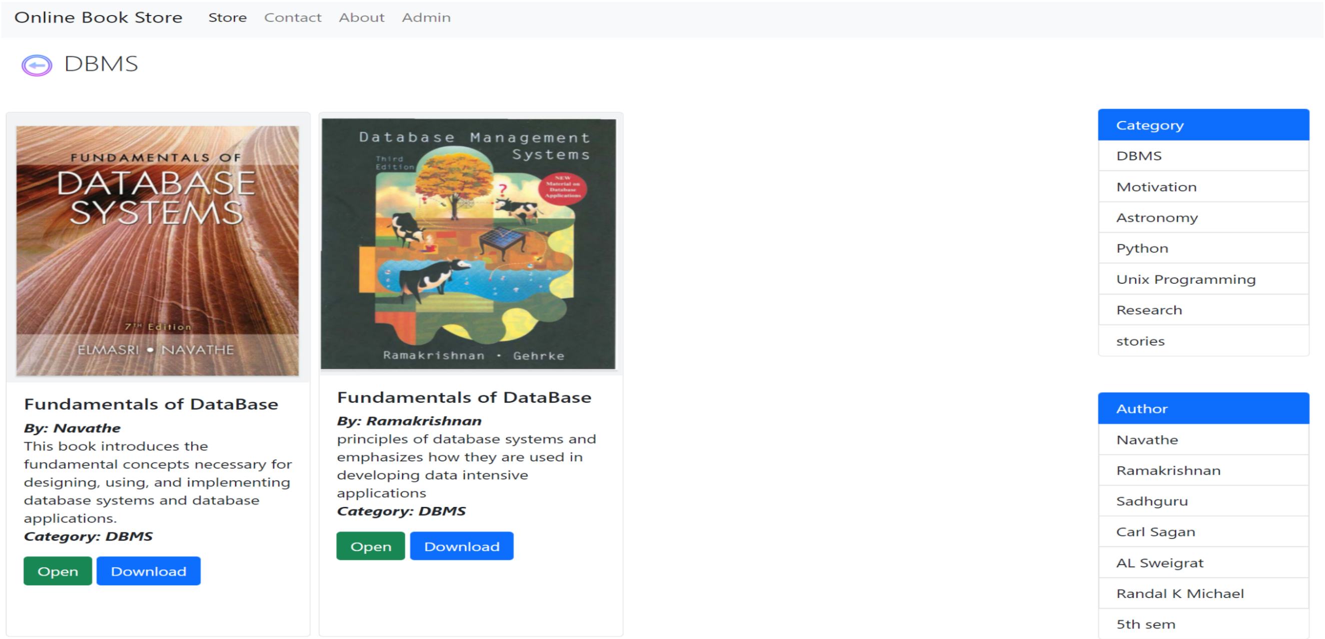
HITESH N , 1BI19IS021
KUSHAL A.C , 1BI19IS028

BookDrive

By: 5th sem

Bookdrive is an online bookstore

Fig 6.2 Research Page



Fundamentals of DataBase

By: Navathe

This book introduces the fundamental concepts necessary for designing, using, and implementing database systems and database applications.

Category: DBMS

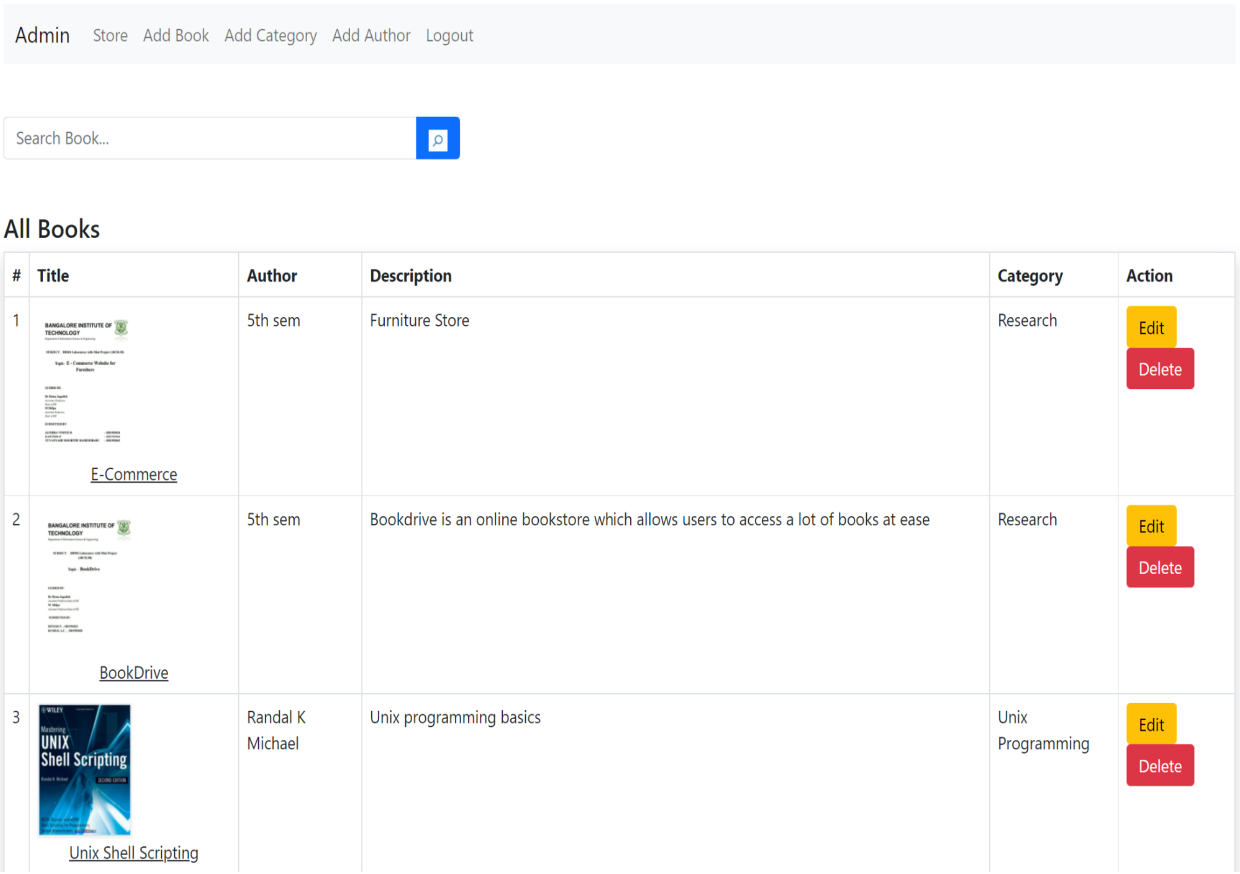
Fundamentals of DataBase

By: Ramakrishnan

principles of database systems and emphasizes how they are used in developing data intensive applications

Category: DBMS

Fig 6.3 Category Page



All Books

#	Title	Author	Description	Category	Action
1	E-Commerce	5th sem	Furniture Store	Research	Edit Delete
2	BookDrive	5th sem	Bookdrive is an online bookstore which allows users to access a lot of books at ease	Research	Edit Delete
3	Unix Shell Scripting	Randal K Michael	Unix programming basics	Unix Programming	Edit Delete

Fig 6.4 Admin Page

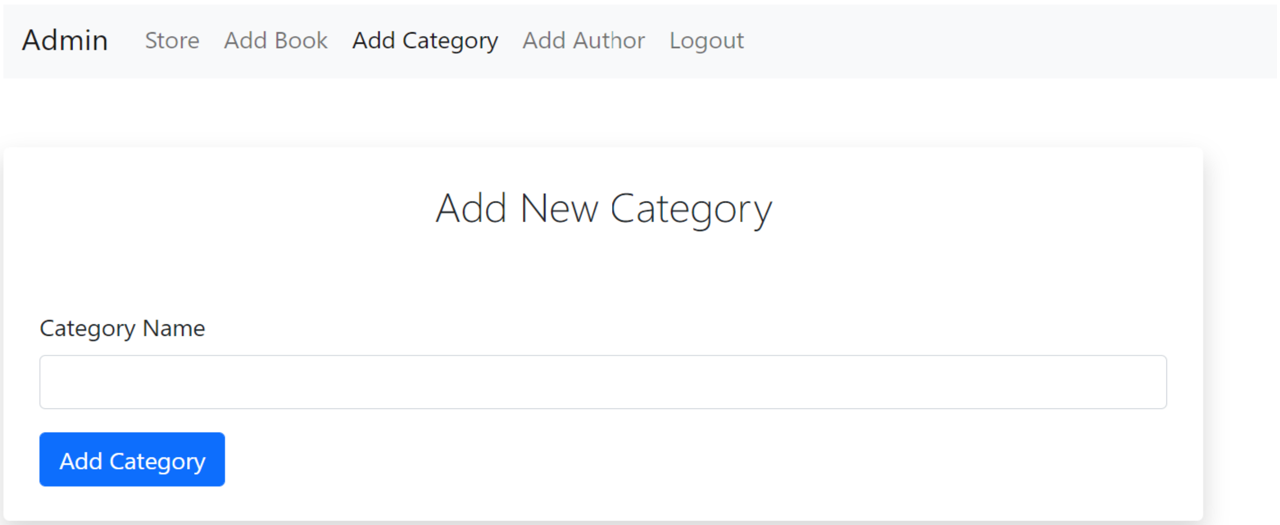
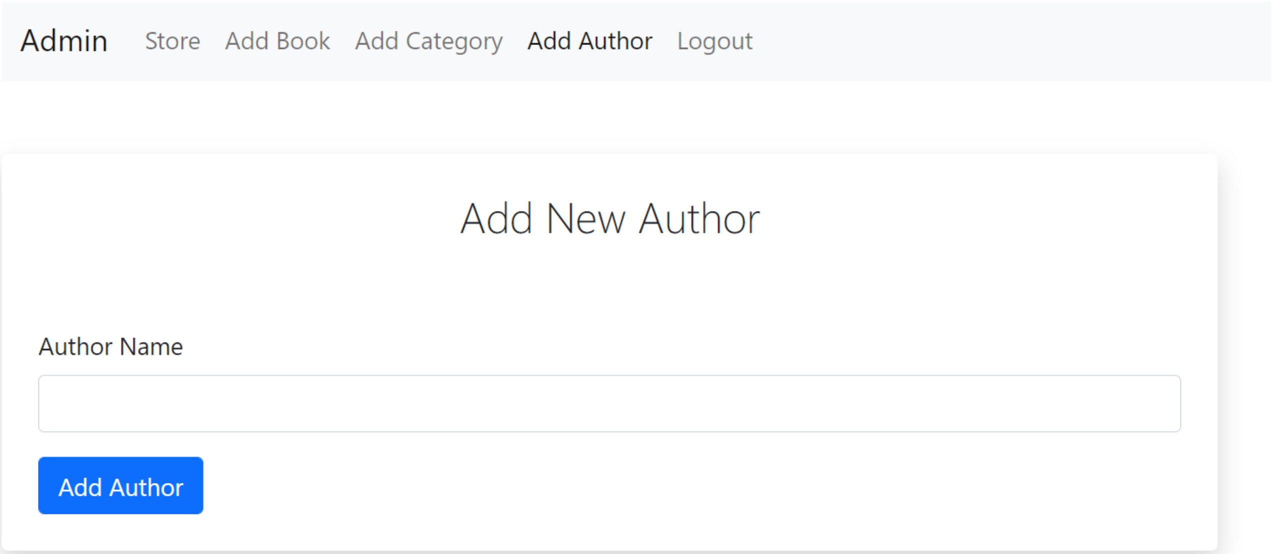


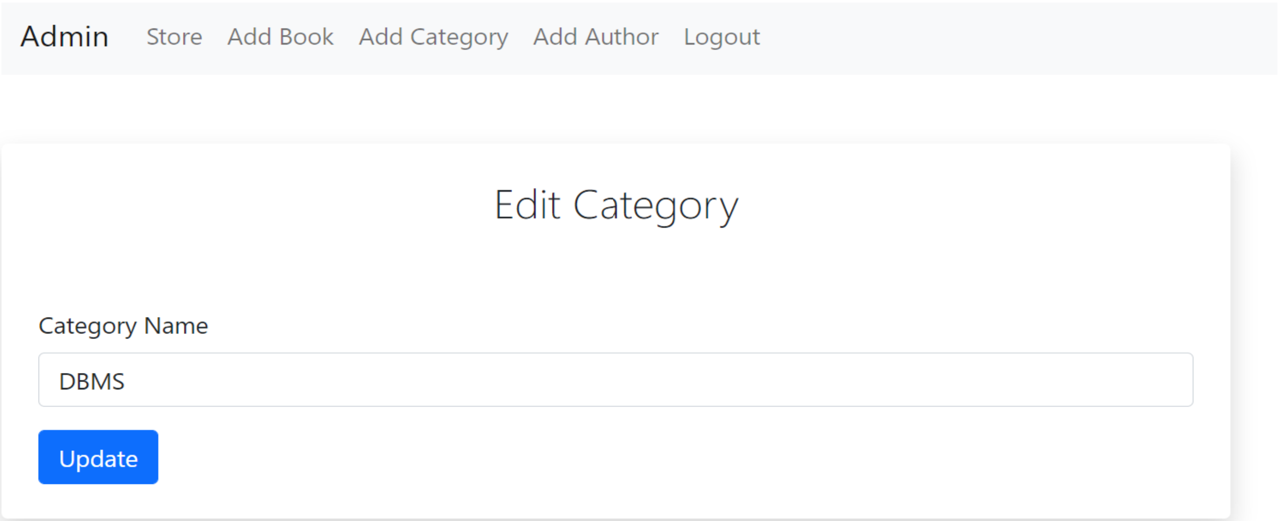
Fig 6.5 Add Category Page



Add New Author

Author Name

Fig 6.6 Add New Book Page

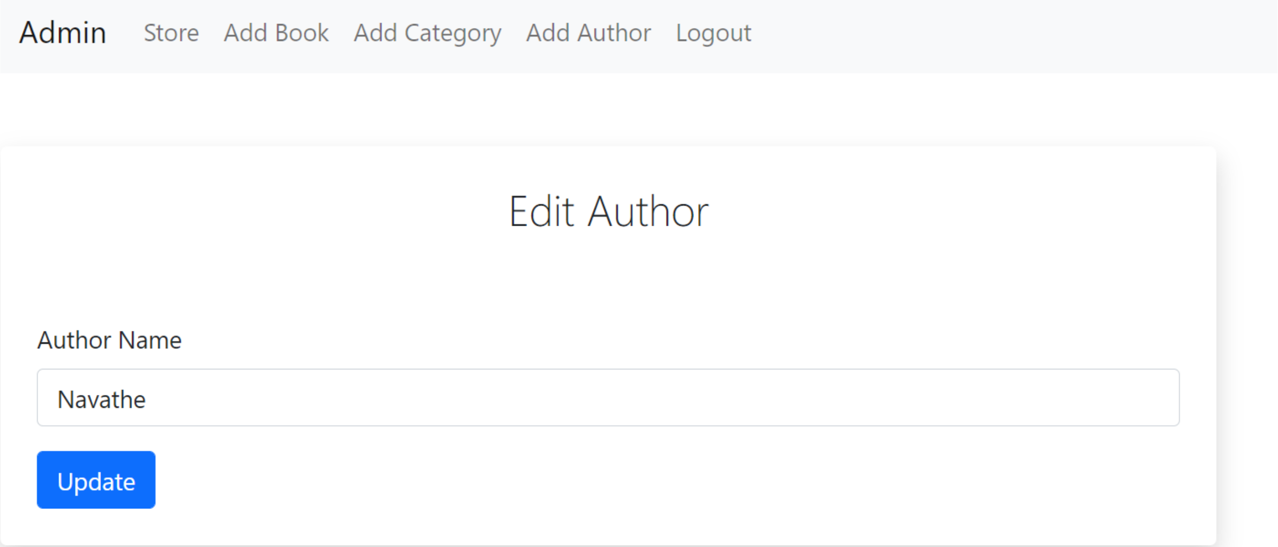


Edit Category

Category Name

DBMS

Fig 6.7 Edit Category Page

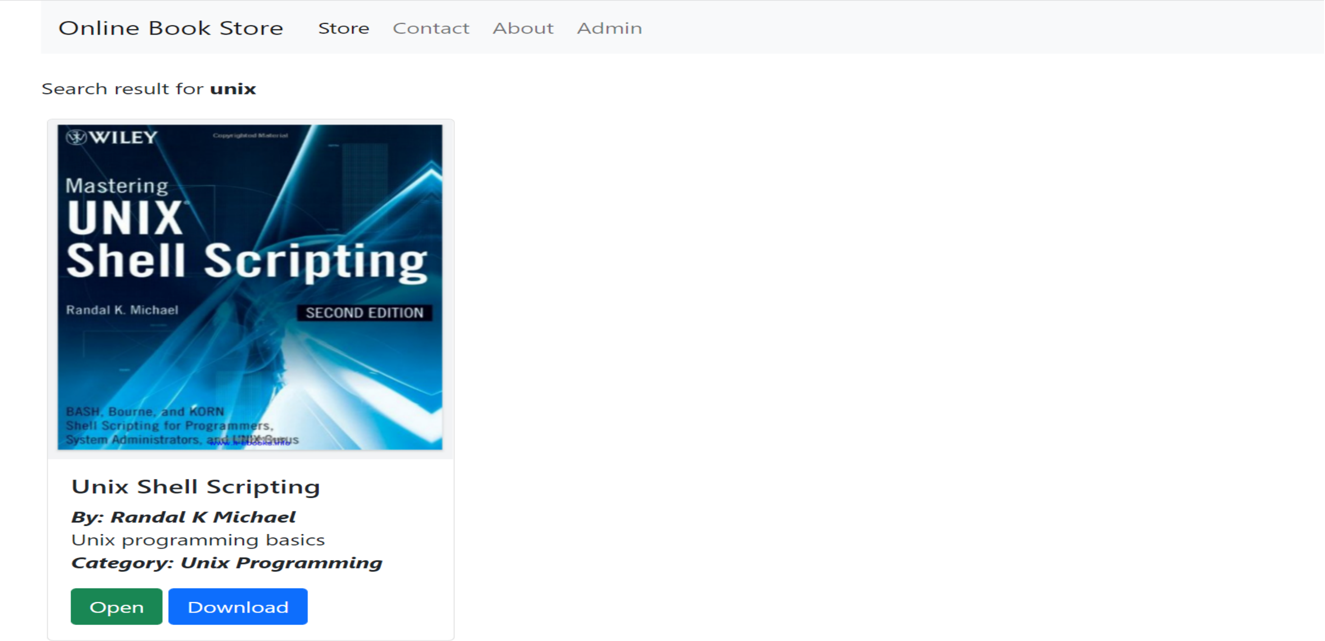


Edit Author

Author Name

Navathe

Fig 6.8 Edit Author Page



Search result for **unix**

Unix Shell Scripting

By: Randal K Michael

Unix programming basics

Category: Unix Programming

Fig 6.9 : Search Page

7. CONCLUSION AND FUTURE ENHANCEMENT

The objective of this project was to build a system which will benefit the students as well as the staff of the library. It makes the entire process online where students can search books, staff can generate reports .

7.1 Conclusion

This website provides a computerized version of the library management system which will benefit the students as well as the staff of the library. It makes the entire process online where students can search books, staff can generate reports and do book transactions. It also has a facility for student login where students can login and can see status of books issued as well as request for books or give some suggestions. It has a facility of teacher's login where teachers can add lectures notes and also give necessary suggestions to the library and also add info about workshops or events happening in our college or nearby college in the online notice board.

7.2 Future Enhancement

There is a future scope of this facility that many more features such as online lectures video tutorials can be added by teachers as well as online assignments submission facility , a feature Of group chat where students can discuss various issues of engineering can be added to this project thus making it more interactive more user friendly and project which fulfills each users need in the best way possible

8. REFERENCES

https://www.w3schools.com/
https://www.w3schools.com/html/html_intro.asp

List of reference for php:

The Joy of PHP Programming: A Beginner's Guide – by Alan Forbes

PHP & MySQL Novice to Ninja – by Kevin Yank

Head First PHP & MySQL – by Lynn Beighley & Michael Morrison

PHP: A Beginner's Guide – by Vikram Vaswani

Murach's PHP & MySQL – by Joel Murach & Ray Harris

PHP & MySQL Web Development – by Luke Welling & Laura Thompson

PHP & MySQL: The Missing Manual – by Brett McLaughlin

Learn PHP & MySQL – Zero to Hero Programming Crash Course – by Paul Madoff

PHP mother site @ http://php.net/.

PHP Manual @ http://php.net/manual/en/.

PHP Language Reference @ http://php.net/manual/en/langref.php.

List of references for Sql:

MySQL Mother Site @ www.mysql.com.

MySQL 5.7 "Reference Manual" @ http://dev.mysql.com/doc/.

MySQL 5.7 "SQL Statement Syntax" @ http://dev.mysql.com/doc/refman/5.7/en/sql-syntax.html.

"ISO/IEC 9075 Information Technology - Database Languages SQL": 1989 (SQL-89), 1992 (SQL-92 or SQL2), 1999 (SQL-99 or SQL3), 2003 (SQL-2003), 2006 (SQL-2006) and 2011 (SQL-2011).

Codd E. F., "A Relational Model of Data for Large Shared Data Banks", Communications of the ACM, vol. 13, issue 6, pp. 377–387, June 1970.

(For Java Programmers) "JDBC Basics", Java Online Tutorial @ http://download.oracle.com/javase/tutorial/jdbc/basics/index.html.

Paul DuBois, "MySQL Developer's Library", 4th ed, 2009 (5th ed is probably available).

Russell Dyer, "MySQL in a Nutshell", 2nd ed, O'reilly, 2008.