|  |  |
| --- | --- |
|  | **國立聯合大學資訊工程學系**  **資工二甲**  **資料庫管理系統實務**  **期末專題報告** |

**圖書資訊系統之架設建置**

指導教授：黃豐隆 老師

學 生: 劉祐伸

學 號: U1224051

中 華 民 國 114 年 6 月 11 日

目錄

[第一章、前言 3](#_Toc200497991)

[1.1 圖書館資訊系統的重要性 3](#_Toc200497992)

[1.2 架設建置的目的 3](#_Toc200497993)

[第二章、系統架設建置 4](#_Toc200497994)

[2.1 系統架構 4](#_Toc200497995)

[第三章、資料庫之實體關聯圖與資料模型 5](#_Toc200497996)

[3.1 正規化的目的 5](#_Toc200497997)

[3.2 實體關聯圖 5](#_Toc200497998)

[3.3 邏輯資料模型 5](#_Toc200497999)

[3.4 實體資料模型 6](#_Toc200498000)

[4.1 表列完成的各項功能 6](#_Toc200498001)

[4.2 操作畫面展示與說明 7](#_Toc200498002)

[第五章 結語 13](#_Toc200498003)

[5.1 總結與新得 13](#_Toc200498004)

[第六章 程式碼 13](#_Toc200498005)

第一章、前言

1.1 圖書館資訊系統的重要性

圖書資訊系統就像是圖書館的智慧大腦。它的重要性不只是在於能借書還書，而是在於能把一整座書海變得井然有序、好查、好用，像個知識GPS。來幾個重點：

1. 提升管理效率

不用再靠人工翻卡片找書啦～系統自動處理書籍編目、館藏管理、借還書紀錄，大幅節省人力和時間。

1. 優化讀者體驗

讀者可以線上查詢書籍、預約、續借，甚至查自己的借閱紀錄，不用走到圖書館也能輕鬆找書！

1. 資料整合與決策支援

系統可以分析熱門借閱書籍、館藏使用率，幫助圖書館做出更好的採購與服務規劃。

1. 保存與傳承知識

數位化的系統不只讓現有知識被有效保存，還能讓未來的人繼續使用、擴展、連結這些知識。

1.2 架設建置的目的

它的終極目標不只是「讓書好找」，而是打造一個「資訊流通、知識共享」的橋樑。具體來說：

1. 建立標準化的作業流程

書籍的分類、編碼、借還流程等透過系統統一規範，不再混亂。

1. 提供穩定的服務平台

讓圖書館可以提供24/7不間斷的服務——即使圖書館關門，線上功能還在跑。

1. 促進資源共享與合作

不同圖書館之間可以互借、共享資源（像大學圖書館聯盟），靠的就是資訊系統的串接能力。

1. 因應數位化趨勢

數位書籍、影音資料、開放教育資源越來越多，沒有系統根本無法有效管理。

第二章、系統架設建置

2.1 系統架構

一張含有 文字, 圖表, 字型, 數字 的圖片

AI 產生的內容可能不正確。

一張含有 文字, 圖表, 螢幕擷取畫面, 方案 的圖片

AI 產生的內容可能不正確。

第三章、資料庫之實體關聯圖與資料模型

3.1 正規化的目的

資料正規化的目的是消除資料重複、確保資料一致性和強化資料關聯性與彈性。

3.2 實體關聯圖

一張含有 文字, 圖表, 行 的圖片

AI 產生的內容可能不正確。

3.3 邏輯資料模型

一張含有 文字, 螢幕擷取畫面, 數字, 字型 的圖片

AI 產生的內容可能不正確。

3.4 實體資料模型

一張含有 文字, 數字, 字型, 平行 的圖片

AI 產生的內容可能不正確。

**第四章 完成的功能說明**

4.1 表列完成的各項功能

|  |  |
| --- | --- |
| **一般功能** | **說明** |
| 查詢 | 查詢書本，可選擇 ISBN、書名、類別、作者、出版年份或是出版社來查詢。 |
| 輸入相關資料 | 在註冊、登入、查詢、借書、還書、新增書本、刪除書本、查看借閱記錄功能都會與資料庫存取。 |
| 新增讀者 | 使用者需輸入使用者名稱、Email、密碼與確認密碼這些資料才可以註冊帳號，註冊成功即登入且跳轉到入口畫面。 |
| 讀者登入 | 使用者需輸入使用者名稱與密碼才可以登入，且跳轉到入口畫面。  使用者可以直接登出。 |
| 借書 | 限制已登入的使用者使用，利用書本查詢結果的 ISBN 與選擇借書日期去借閱書本。  如有逾期未還的書，使用者無法使用此功能。 |
| 還書 | 限制已登入的使用者使用，點擊還書按鈕可以查看使用者有多少書本需要歸還。  再次點擊該還書資料的還書按鈕即可還書，並且顯示是否有逾期罰款。 |
| 修改圖書 | 限制管理者使用，依照輸入的 ISBN 去修改書名、類型、出版年份、作者或是出版社 ID 欄位。 |
| 刪除圖書 | 限制管理者使用，輸入該書本的 ISBN 即可刪除。 |
| 額外功能 | 說明 |
| 身份組 | 此系統新增了兩個身份組，分別為：一般使用者與管理者。 |
| 登入鎖定 | 當使用者短時間內登入多次會導致登入暫時鎖定，需等待一段時間後才可以再次登入。 |
| 新增圖書 | 限制管理者使用，輸入 ISBN、書名、類型、出版年份、作者與出版社 ID 來新增書本資料。 |
| 查看借閱記錄 | 限制管理者使用，可選擇性的輸入使用者 ID 與書本 ISBN 來查詢其使用者借閱的記錄。 |

4.2 操作畫面展示與說明

主畫面

一張含有 文字, 螢幕擷取畫面, 軟體, 字型 的圖片

AI 產生的內容可能不正確。

選擇欄位搜尋

一張含有 文字, 螢幕擷取畫面, 陳列, 數字 的圖片

AI 產生的內容可能不正確。 一張含有 文字, 螢幕擷取畫面, 字型, 數字 的圖片

AI 產生的內容可能不正確。

查詢結果

一張含有 文字, 螢幕擷取畫面, 數字, 字型 的圖片

AI 產生的內容可能不正確。

註冊

一張含有 文字, 螢幕擷取畫面, 字型, 數字 的圖片

AI 產生的內容可能不正確。 一張含有 文字, 螢幕擷取畫面, 字型, 數字 的圖片

AI 產生的內容可能不正確。

登入、登入失敗與登入鎖定

一張含有 文字, 螢幕擷取畫面, 字型, 數字 的圖片

AI 產生的內容可能不正確。 一張含有 文字, 螢幕擷取畫面, 字型, 數字 的圖片

AI 產生的內容可能不正確。 一張含有 文字, 螢幕擷取畫面, 字型, 數字 的圖片

AI 產生的內容可能不正確。

使用者登入介面

一張含有 文字, 螢幕擷取畫面, 軟體, 字型 的圖片

AI 產生的內容可能不正確。

借書

一張含有 文字, 螢幕擷取畫面, 字型, 數字 的圖片

AI 產生的內容可能不正確。 一張含有 文字, 螢幕擷取畫面, 字型, 數字 的圖片

AI 產生的內容可能不正確。

一張含有 文字, 螢幕擷取畫面, 字型, 數字 的圖片

AI 產生的內容可能不正確。 一張含有 文字, 螢幕擷取畫面, 字型, 數字 的圖片

AI 產生的內容可能不正確。

還書

一張含有 文字, 螢幕擷取畫面, 字型, 數字 的圖片

AI 產生的內容可能不正確。一張含有 文字, 螢幕擷取畫面, 字型, 數字 的圖片

AI 產生的內容可能不正確。一張含有 文字, 螢幕擷取畫面, 字型, 標誌 的圖片

AI 產生的內容可能不正確。

管理員介面

一張含有 文字, 螢幕擷取畫面, 軟體, 字型 的圖片

AI 產生的內容可能不正確。

管理員操作項目

一張含有 文字, 螢幕擷取畫面, 字型, 數字 的圖片

AI 產生的內容可能不正確。 一張含有 文字, 螢幕擷取畫面, 字型, 數字 的圖片

AI 產生的內容可能不正確。 一張含有 文字, 螢幕擷取畫面, 字型, 數字 的圖片

AI 產生的內容可能不正確。一張含有 文字, 螢幕擷取畫面, 字型, 數字 的圖片

AI 產生的內容可能不正確。 一張含有 文字, 螢幕擷取畫面, 字型, 數字 的圖片

AI 產生的內容可能不正確。

查詢借閱記錄

一張含有 文字, 螢幕擷取畫面, 數字, 字型 的圖片

AI 產生的內容可能不正確。

第五章 結語

5.1 總結與新得

整體製作過程有些複雜，從一開始要評估設計需要哪些實體、屬性，以及檢查資料表有沒有符合正規化，還需畫出資料表之間的實體關聯圖、邏輯資料模型與實體資料模型，設計完這些圖之後要輸入到資料庫裡面並新增有關的資料，再來要讓使用者使用這個網頁，要設計全端的系統架構，最麻煩的部分是後端的部分，頁面跳轉、路由的控制與資料庫的存取，都需要好好的思考該怎麼交互。

整個製作下來，我覺得之後可以多新增一些管理者的功能，例如，更改使用者的權限或操作更多資料表等。

第六章 程式碼

# index

<?php

if (session\_status() == PHP\_SESSION\_NONE) {

session\_start();

}

?>

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Library Management System</title>

<link rel="stylesheet" href="/assets/css/theme.css">

</head>

<body>

<?php include "./assets/pages/header.php"; ?>

<main>

<div class="main-content">

<?php include "./assets/pages/book\_search.php"; ?>

<?php

if (isset($\_SESSION['user\_id'])) {

include "./assets/pages/user.php";

if ($\_SESSION['role'] === 'admin') {

include "./assets/pages/admin.php";

}

}

?>

</div>

</main>

<?php include "./assets/pages/footer.php"; ?>

</body>

</html>

# header.php

<?php

if (session\_status() == PHP\_SESSION\_NONE) {

session\_start();

}

?>

<header class="header-container">

<div class="header-content">

<a href="../../index.php" class="logo"><h1>Library Management System</h1></a>

<div class="login-register">

<?php if (isset($\_SESSION['user\_id'])): ?>

<p class="username">Welcome, <?php echo htmlspecialchars($\_SESSION['username']); ?>!</p>

<form action="/assets/components/signOut.php" method="post">

<button type="submit">Logout</button>

</form>

<?php else: ?>

<form action="/assets/pages/auth.php" method="post">

<input type="hidden" name="enter" value="login">

<button type="submit">Login</button>

</form>

<form action="/assets/pages/auth.php" method="post">

<input type="hidden" name="enter" value="register">

<button type="submit">Register</button>

</form>

<?php endif; ?>

</div>

</div>

</header>

# footer

<footer>

<p>Copyright &copy; 2025 Library Management System by 劉祐伸</p>

</footer>

# db.php

<?php

function connect(){

$servername = "localhost";

$dbname = "library";

$username = "root";

$password = "";

$conn = null;

try{

$conn = new PDO("mysql:host=$servername;dbname=$dbname", $username, $password);

$conn->setAttribute(PDO::ATTR\_ERRMODE, PDO::ERRMODE\_EXCEPTION);

return $conn;

}catch (PDOException $e){

return false;

}

}

?>

# signOut.php

<?php

session\_start();

session\_unset();

session\_destroy();

header("Location: ../../index.php");

exit;

?>

# theme.css

\* {

margin: 0;

padding: 0;

box-sizing: border-box;

user-select: none;

}

body {

font-family: Arial, sans-serif;

background-color: #f4f4f4;

color: #333;

}

header {

background-color: #4caf50;

color: white;

padding: 1rem;

}

.header-container .header-content {

display: flex;

justify-content: space-between;

align-items: center;

}

.header-container .header-content .logo {

color: white;

text-decoration: none;

}

.header-container .header-content .login-register {

display: flex;

align-items: center;

gap: 10px;

}

.header-container .header-content .login-register .username {

font-size: 18px;

color: white;

font-weight: bold;

}

main {

padding: 20px;

}

main .main-content {

display: flex;

align-items: center;

justify-content: center;

gap: 5rem;

}

main .main-content div {

display: flex;

flex-direction: column;

gap: 10px;

align-items: center;

justify-content: center;

}

main .main-content .form-container {

display: flex;

flex-direction: column;

align-items: center;

gap: 10px;

margin-bottom: 20px;

width: 100%;

max-width: 600px;

padding: 20px;

background-color: #fff;

border-radius: 10px;

box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);

margin-top: 20px;

margin-bottom: 20px;

margin-left: 20px;

margin-right: 20px;

}

main .main-content .form-container label {

font-size: 16px;

font-weight: bold;

margin-bottom: 5px;

}

main .main-content .form-container .button-container {

width: 100%;

display: flex;

justify-content: center;

align-items: center;

gap: 20px;

}

table {

width: 100%;

border-collapse: collapse;

background: white;

margin-top: 20px;

}

th,

td {

padding: 12px;

border: 1px solid #aaa;

text-align: left;

}

th {

background-color: #4caf50;

color: white;

}

footer {

background-color: #4caf50;

color: white;

padding: 10px;

text-align: center;

position: fixed;

bottom: 0;

width: 100%;

}

button {

width: 100%;

padding: 10px;

border-radius: 5px;

border: 1px solid #ccc;

font-size: 20px;

font-weight: bold;

background-color: #4caf50;

color: white;

cursor: pointer;

transition: background-color 0.3s ease;

&:hover {

background-color: #45a049;

}

}

input {

width: 100%;

padding: 10px;

border-radius: 5px;

border: 1px solid #ccc;

font-size: 16px;

letter-spacing: 1px;

transition: background-color 0.3s ease;

}

select {

width: 100%;

padding: 10px;

border-radius: 5px;

border: 1px solid #ccc;

font-size: 16px;

font-weight: bold;

letter-spacing: 1px;

transition: background-color 0.3s ease;

}

.btn-back {

padding: 10px;

border-radius: 5px;

border: 1px solid #ccc;

font-size: 20px;

font-weight: bold;

background-color: white;

color: black;

cursor: pointer;

transition: background-color 0.3s ease;

&:hover {

background-color: #ccc;

}

}

.btn-function {

padding: 10px;

border-radius: 5px;

border: 1px solid #ccc;

font-size: 20px;

font-weight: bold;

background-color: #2d87ad;

color: white;

cursor: pointer;

transition: background-color 0.3s ease;

&:hover {

background-color: #1f5f7a;

}

}

# 404.html

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8" />

<title>404 Not Found</title>

<style>

body {

background-color: #f8f9fa;

color: #343a40;

font-family: "Segoe UI", Tahoma, Geneva, Verdana, sans-serif;

text-align: center;

padding: 100px;

}

h1 {

font-size: 6em;

margin-bottom: 0;

}

p {

font-size: 1.5em;

margin-top: 0;

}

a {

display: inline-block;

margin-top: 30px;

text-decoration: none;

color: #fff;

background-color: #4caf50;

padding: 10px 20px;

border-radius: 8px;

font-size: 1em;

transition: background-color 0.3s ease;

}

a:hover {

background-color: #45a049;

}

</style>

</head>

<body>

<h1>404</h1>

<p>Oops! 找不到你要的頁面。</p>

<a href="../../index.php">返回首頁</a>

</body>

</html>

# auth.php

<?php

$enter = $\_POST['enter'] ?? $\_GET['enter'] ?? '';

?>

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Auth</title>

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

</head>

<body>

<?php include 'header.php'; ?>

<main>

<div class="main-content">

<div class="auth-container">

<?php

switch ($enter) {

case 'login':

include 'login.php';

break;

case 'register':

include 'register.php';

break;

default:

header("Location: 404.html");

exit;

break;

}

?>

<?php if (isset($\_GET['error'])): ?>

<?php

switch ($\_GET['error']) {

case 'locked':

$msg = "❌ 以嘗試多次，請稍後登入。";

break;

case 'login':

$msg = "❌ 登入失敗：帳號或密碼錯誤。";

break;

case 'register':

$msg = "❌ 註冊失敗：請檢查您的輸入。";

break;

case 'exists':

$msg = "⚠️ 此使用者名稱已被註冊。";

break;

default:

$msg = "";

}

?>

<?php if (!empty($msg)): ?>

<div class="alert alert-danger" role="alert">

<?= $msg ?>

</div>

<?php endif; ?>

<?php endif; ?>

</div>

</div>

</main>

<?php include 'footer.php'; ?>

</body>

</html>

# register.php

<h2>Register</h2>

<form action="dashboard.php" method="post" class="form-container">

<input type="hidden" name="action" value="register">

<label>Username:</label>

<input type="text" name="username" required>

<label>Email:</label>

<input type="email" name="email" required>

<label>Password:</label>

<input type="password" name="password" required>

<label>Confirm Password:</label>

<input type="password" name="confirm\_password" required>

<div class="button-container">

<button type="submit">Register</button>

<button type="button" onclick="window.location.href='../../index.php'" class="btn-back">返回</button>

</div>

</form>

# login.php

<h2>Login</h2>

<form action="dashboard.php" method="post" class="form-container">

<input type="hidden" name="action" value="login">

<label>Username:</label>

<input type="text" name="username" required>

<label>Password:</label>

<input type="password" name="password" required>

<div class="button-container">

<button type="submit">Login</button>

<button type="button" onclick="window.location.href='../../index.php'" class="btn-back">返回</button>

</div>

</form>

# dashboard.php

<?php

session\_start();

date\_default\_timezone\_set("Asia/Taipei");

$action = $\_POST['action'] ?? '';

require "../components/db.php";

$conn = connect();

if (!$conn) {

echo "<h2>Database connection failed.</h2>";

exit;

}

if ($action === 'login') {

$username = $\_POST['username'] ?? '';

$password = $\_POST['password'] ?? '';

$sql = "SELECT u.UID, u.name, u.role, a.password, a.failed\_attempts, a.is\_locked, a.login\_time

FROM user as u

JOIN account as a ON u.UID = a.AID

WHERE u.name = :username";

$stmt = $conn->prepare($sql);

$stmt->bindValue(':username', $username);

$stmt->execute();

if ($row = $stmt->fetch(PDO::FETCH\_ASSOC)) {

if ($row['is\_locked']) {

$locked\_time = new DateTime($row['login\_time']);

$now = new DateTime();

$interval = $now->getTimestamp() - $locked\_time->getTimestamp();

if ($interval >= 5) {

$unlock = $conn->prepare("UPDATE account SET is\_locked = 0, failed\_attempts = 0 WHERE AID = :aid");

$unlock->bindValue(':aid', $row['UID']);

$unlock->execute();

$stmt->execute();

$row = $stmt->fetch(PDO::FETCH\_ASSOC);

} else {

header("Location: auth.php?enter=login&error=locked");

exit;

}

}

if (password\_verify($password, $row['password'])) {

$update = $conn->prepare("UPDATE account SET failed\_attempts = 0, is\_locked = 0, login\_time = NOW() WHERE AID = :aid");

$update->bindValue(':aid', $row['UID']);

$update->execute();

$\_SESSION['user\_id'] = $row['UID'];

$\_SESSION['username'] = $row['name'];

$\_SESSION['role'] = $row['role'];

header("Location: ../../index.php");

exit;

} else {

$fail = $conn->prepare("UPDATE account SET failed\_attempts = failed\_attempts + 1 WHERE AID = :aid");

$fail->bindValue(':aid', $row['UID']);

$fail->execute();

$checkFail = $conn->prepare("SELECT failed\_attempts FROM account WHERE AID = :aid");

$checkFail->bindValue(':aid', $row['UID']);

$checkFail->execute();

$attempt = $checkFail->fetchColumn();

if ($attempt >= 3) {

$lock = $conn->prepare("UPDATE account SET is\_locked = 1, login\_time = NOW() WHERE AID = :aid");

$lock->bindValue(':aid', $row['UID']);

$lock->execute();

}

header("Location: auth.php?enter=login&error=login");

exit;

}

}

header("Location: auth.php?enter=login&error=login");

exit;

} elseif ($action === 'register') {

$username = $\_POST['username'] ?? '';

$email = $\_POST['email'] ?? '';

$password = $\_POST['password'] ?? '';

$confirm\_password = $\_POST['confirm\_password'] ?? '';

if (empty($username) || empty($password) || $password !== $confirm\_password) {

header("Location: auth.php?enter=register&error=register");

exit;

}

// 檢查帳號是否已存在

$stmt = $conn->prepare("SELECT COUNT(\*) FROM user WHERE name = :username");

$stmt->bindValue(':username', $username);

$stmt->execute();

if ($stmt->fetchColumn() > 0) {

header("Location: auth.php?enter=register&error=exists");

exit;

}

// 開始註冊

$hashed = password\_hash($password, PASSWORD\_DEFAULT);

$now = date("Y-m-d H:i:s");

try {

$conn->beginTransaction();

// 建立 user（UID 自動遞增）

$stmt1 = $conn->prepare("INSERT INTO user (name, email, join\_date) VALUES (:name, :email, :date)");

$stmt1->execute([

':name' => $username,

':email' => $email,

':date' => $now

]);

$uid = $conn->lastInsertId(); // 拿自動產生的 UID

// 建立 account（用相同 UID）

$stmt2 = $conn->prepare("INSERT INTO account (AID, password, login\_time, failed\_attempts, is\_locked) VALUES (:aid, :pw, :time, 0, 0)");

$stmt2->execute([

':aid' => $uid,

':pw' => $hashed,

':time' => $now

]);

$conn->commit();

// 自動登入

$\_SESSION['user\_id'] = $uid;

$\_SESSION['username'] = $username;

$role\_stmt = $conn->prepare("SELECT role FROM user WHERE UID = :uid");

$role\_stmt->bindValue(':uid', $uid);

$role\_stmt->execute();

$\_SESSION['role'] = $role\_stmt->fetchColumn();

header("Location: ../../index.php");

exit;

} catch (Exception $e) {

$conn->rollBack();

echo "<h2>❌ 註冊失敗：" . htmlspecialchars($e->getMessage()) . "</h2>";

exit;

}

} else {

header("Location: 404.html");

exit;

}

?>

# book\_search.php

<div id="book-search">

<h2>書本查詢</h2>

<form action="/assets/pages/book\_query.php" method="post" class="form-container">

<select name="type" required>

<option value="">--請選擇--</option>

<option value="ISBN">ISBN</option>

<option value="title">書名</option>

<option value="genre">類別</option>

<option value="author">作者</option>

<option value="year">出版年份</option>

<option value="publisher">出版社</option>

</select>

<input type="text" name="search" placeholder="請輸入檢索詞" required>

<button type="submit">查詢</button>

</form>

</div>

# book\_query.php

<?php

$type = $\_POST['type'] ?? '';

$search = $\_POST['search'] ?? '';

require "../components/db.php";

$conn = connect();

if (!$conn) {

echo "資料庫連線失敗";

exit;

}

$validFields = [

'ISBN' => 'b.ISBN',

'title' => 'b.title',

'genre' => 'b.genre',

'year' => 'b.year',

'author' => 'b.author',

'publisher' => 'p.name'

];

if (!isset($validFields[$type])) {

echo "不合法的查詢欄位";

exit;

}

$field = $validFields[$type];

$useLike = in\_array($type, ['title', 'genre', 'author', 'publisher']);

$sql = "SELECT b.ISBN, b.title, b.genre, b.year, b.author, p.name AS publisher

FROM book AS b

JOIN publisher AS p ON b.PID = p.PID

WHERE $field " . ($useLike ? "LIKE :search" : "= :search");

$stmt = $conn->prepare($sql);

$stmt->bindValue(':search', $useLike ? "%$search%" : $search);

$stmt->execute();

$results = $stmt->fetchAll(PDO::FETCH\_ASSOC);

?>

<!DOCTYPE html>

<html lang="zh-Hant">

<head>

<meta charset="UTF-8">

<title>查詢結果</title>

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

</head>

<body>

<?php include "./header.php"; ?>

<main>

<div class="main-content">

<div class="book-query">

<h2>查詢結果</h2>

<?php if ($results): ?>

<table>

<thead>

<tr>

<th>ISBN</th>

<th>書名</th>

<th>類型</th>

<th>年份</th>

<th>作者</th>

<th>出版社</th>

</tr>

</thead>

<tbody>

<?php foreach ($results as $row): ?>

<tr>

<td><?= htmlspecialchars($row['ISBN']) ?></td>

<td><?= htmlspecialchars($row['title']) ?></td>

<td><?= htmlspecialchars($row['genre']) ?></td>

<td><?= htmlspecialchars($row['year']) ?></td>

<td><?= htmlspecialchars($row['author']) ?></td>

<td><?= htmlspecialchars($row['publisher']) ?></td>

</tr>

<?php endforeach; ?>

</tbody>

</table>

<?php else: ?>

<p>查無資料，請重新查詢。</p>

<?php endif; ?>

<button type="button" onclick="window.location.href='../../index.php'" class="btn-back">返回</button>

</div>

</div>

</main>

<?php include "./footer.php"; ?>

</body>

</html>

# user.php

<div id="user-section">

<h2>借書</h2>

<form action="/assets/pages/borrow.php" method="POST" class="form-container">

<label for="ISBN">ISBN:</label>

<input type="text" id="ISBN" name="ISBN" required>

<label for="borrow\_date">借書日期:</label>

<input type="date" id="borrow\_date" name="borrow\_date" required>

<div class="button-container">

<button type="submit">借書</button>

<button type="button" onclick="window.location.href='/assets/pages/return.php'" class="btn-function">還書</button>

</div>

<?php

if (isset($\_GET['section']) && $\_GET['section'] === 'user' && isset($\_GET['borrow'])) {

switch ($\_GET['borrow']) {

case 'success':

echo "<div class='alert alert-success'>✅ 借書成功！</div>";

break;

case 'overdue':

echo "<div class='alert alert-danger'>🚫 您有逾期未還的書，請先歸還。</div>";

break;

case 'invalid':

echo "<div class='alert alert-warning'>⚠️ 資料輸入不完整或錯誤。</div>";

break;

case 'notfound':

echo "<div class='alert alert-danger'>❌ 該書籍不存在，請檢查 ISBN。</div>";

break;

case 'error':

echo "<div class='alert alert-danger'>❌ 借書失敗，請檢查您的輸入。</div>";

break;

case 'unavailable':

echo "<div class='alert alert-warning'>⚠️ 該書籍目前無法借閱，請稍後再試。</div>";

break;

default:

echo "<div class='alert alert-danger'>❌ 借書失敗，請再試一次。</div>";

break;

}

}

?>

</form>

</div>

# return.php

<?php

session\_start();

require '../components/db.php';

$conn = connect();

$uid = $\_SESSION['user\_id'];

$message = "";

if ($\_SERVER['REQUEST\_METHOD'] === 'POST' && isset($\_POST['lid'])) {

$lid = $\_POST['lid'];

$return\_date = date("Y-m-d H:i:s");

$stmt = $conn->prepare("SELECT due\_date FROM borrowlog WHERE LID = :lid AND UID = :uid");

$stmt->execute([':lid' => $lid, ':uid' => $uid]);

$row = $stmt->fetch(PDO::FETCH\_ASSOC);

if ($row) {

$due\_date = $row['due\_date'];

$fine = 0;

if (strtotime($return\_date) > strtotime($due\_date)) {

$days\_late = ceil((strtotime($return\_date) - strtotime($due\_date)) / 86400);

$fine = $days\_late \* 10;

}

$update = $conn->prepare("UPDATE borrowlog SET is\_returned = 1, return\_date = :ret, fine = :fine WHERE LID = :lid AND UID = :uid");

$update->execute([

':ret' => $return\_date,

':fine' => $fine,

':lid' => $lid,

':uid' => $uid

]);

$message = ($fine > 0)

? "<div class='alert alert-warning'>⚠️ 還書成功，但您逾期 {$days\_late} 天，罰鍰 {$fine} 元。</div>"

: "<div class='alert alert-success'>✅ 還書成功，感謝您準時歸還！</div>";

} else {

$message = "<div class='alert alert-danger'>❌ 找不到對應的借書紀錄。</div>";

}

}

$stmt = $conn->prepare("SELECT \* FROM borrowlog WHERE UID = :uid AND is\_returned = 0 ORDER BY borrow\_date DESC");

$stmt->execute([':uid' => $uid]);

$logs = $stmt->fetchAll(PDO::FETCH\_ASSOC);

?>

<!DOCTYPE html>

<html lang="zh-Hant">

<head>

<meta charset="UTF-8" />

<title>還書記錄</title>

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

</head>

<body>

<?php require 'header.php'; ?>

<main>

<div class="main-content">

<div id="return-section">

<h2>還書紀錄</h2>

<?= $message ?>

<?php if (count($logs) >

0): ?>

<table border="1" cellpadding="8">

<tr>

<th>書籍 ISBN</th>

<th>借書日期</th>

<th>到期日期</th>

<th>操作</th>

</tr>

<?php foreach ($logs as $log): ?>

<tr>

<td><?= htmlspecialchars($log['ISBN']) ?></td>

<td><?= $log['borrow\_date'] ?></td>

<td><?= $log['due\_date'] ?></td>

<td>

<form method="POST" action="return.php">

<input type="hidden" name="lid" value="<?= $log['LID'] ?>" />

<button type="submit">還書</button>

</form>

</td>

</tr>

<?php endforeach; ?>

</table>

<?php else: ?>

<p>✅ 您目前沒有未歸還的書籍。</p>

<?php endif; ?>

<button type="button" onclick="window.location.href='../../index.php'" class="btn-back">返回</button>

</div>

</div>

</main>

<?php require 'footer.php'; ?>

</body>

</html>

# borrow.php

<?php

if (session\_status() == PHP\_SESSION\_NONE) {

session\_start();

}

date\_default\_timezone\_set("Asia/Taipei");

require '../components/db.php';

$conn = connect();

$uid = $\_SESSION['user\_id'];

$isbn = $\_POST['ISBN'] ?? '';

$borrow = $\_POST['borrow\_date'] ?? '';

if (empty($isbn)) {

header("Location: /index.php?section=user&borrow=error");

exit;

}

// 檢查是否有逾期未還

$check = $conn->prepare("

SELECT COUNT(\*) FROM borrowlog

WHERE UID = :uid AND is\_returned = 0 AND due\_date < NOW()

");

$check->bindValue(':uid', $uid);

$check->execute();

if ($check->fetchColumn() > 0) {

header("Location: /index.php?section=user&borrow=overdue");

exit;

}

// 檢查該書籍是否存在

$book\_check = $conn->prepare("SELECT COUNT(\*) FROM book WHERE ISBN = :isbn");

$book\_check->bindValue(':isbn', $isbn);

$book\_check->execute();

if ($book\_check->fetchColumn() == 0) {

header("Location: /index.php?section=user&borrow=notfound");

exit;

}

// 檢查書籍是否已有借閱未歸還

$check\_book = $conn->prepare("

SELECT COUNT(\*) FROM borrowlog

WHERE ISBN = :isbn AND is\_returned = 0

");

$check\_book->bindValue(':isbn', $isbn);

$check\_book->execute();

if ($check\_book->fetchColumn() > 0) {

header("Location: /index.php?section=user&borrow=unavailable");

exit;

}

// 寫入借書紀錄

$due = date("Y-m-d H:i:s", strtotime($borrow . " +14 days"));

$stmt = $conn->prepare("

INSERT INTO borrowlog (UID, ISBN, borrow\_date, due\_date, is\_returned, fine)

VALUES (:uid, :isbn, :borrow\_date, :due\_date, 0, 0)

");

$stmt->execute([

':uid' => $uid,

':isbn' => $isbn,

':borrow\_date' => $borrow,

':due\_date' => $due

]);

header("Location: /index.php?section=user&borrow=success");

exit;

?>

# admin.php

<div id="admin-section">

<h2>管理員操作</h2>

<form action="/assets/pages/admin\_action.php" method="POST" class="form-container">

<label for="action">選擇操作:</label>

<select id="action" name="action" required>

<option value="">-- 請選擇 --</option>

<option value="add">新增書籍</option>

<option value="update">修改書籍</option>

<option value="delete">刪除書籍</option>

<option value="view\_logs">查看借閱紀錄</option>

</select>

<div class="button-container">

<button type="submit">執行操作</button>

</div>

</form>

</div>

# admin\_action.php

<?php

session\_start();

if (!isset($\_SESSION['user\_id']) || $\_SESSION['role'] !== 'admin') {

echo "<h3>🚫 您沒有權限存取此頁面。</h3>";

exit;

}

$action = $\_POST['action'] ?? $\_GET['action'] ?? '';

?>

<!DOCTYPE html>

<html lang="zh-Hant">

<head>

<meta charset="UTF-8">

<title>管理員操作</title>

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

</head>

<body>

<?php require "header.php"; ?>

<main>

<div class="main-content">

<div class="admin-action">

<h2>📚 管理員操作 - <?= htmlspecialchars($action) ?></h2>

<?php if ($action === 'add'): ?>

<form action="add\_book.php" method="POST" class="form-container">

<label>ISBN:</label>

<input type="text" name="isbn" required>

<label>書名:</label>

<input type="text" name="title" required>

<label>類型:</label>

<input type="text" name="genre" required>

<label>出版年份:</label>

<input type="number" name="year" required>

<label>作者:</label>

<input type="text" name="author" required>

<label>出版社 ID:</label>

<input type="number" name="pid" required>

<div class="button-container">

<button type="submit">新增書籍</button>

<button type="button" onclick="window.location.href='../../index.php'" class="btn-back">返回</button>

</div>

<?php if (isset($\_GET['result']) && $\_GET['result'] === 'success'): ?>

<div class="alert alert-success">✅ 書籍新增成功！</div>

<?php elseif (isset($\_GET['result']) && $\_GET['result'] === 'exists'): ?>

<div class="alert alert-warning">⚠️ 該書籍已存在，請勿重複新增。</div>

<?php elseif (isset($\_GET['result']) && $\_GET['result'] === 'error'): ?>

<div class="alert alert-danger">❌ 書籍新增失敗，請檢查欄位。</div>

<?php endif; ?>

</form>

<?php elseif ($action === 'delete'): ?>

<form action="delete\_book.php" method="POST" class="form-container">

<label>ISBN:</label>

<input type="text" name="isbn" required>

<div class="button-container">

<button type="submit">刪除書籍</button>

<button type="button" onclick="window.location.href='../../index.php'" class="btn-back">返回</button>

</div>

<?php if (isset($\_GET['result']) && $\_GET['result'] === 'success'): ?>

<div class="alert alert-success">✅ 書籍刪除成功！</div>

<?php elseif (isset($\_GET['result']) && $\_GET['result'] === 'error'): ?>

<div class="alert alert-danger">❌ 書籍刪除失敗，請檢查欄位。</div>

<?php endif; ?>

</form>

<?php elseif ($action === 'update'): ?>

<form action="update\_book.php" method="POST" class="form-container">

<label>ISBN (要修改的):</label>

<input type="text" name="isbn" required>

<label>書名:</label>

<input type="text" name="title">

<label>類型:</label>

<input type="text" name="genre">

<label>出版年份:</label>

<input type="number" name="year">

<label>作者:</label>

<input type="text" name="author">

<label>出版社 ID:</label>

<input type="number" name="pid">

<div class="button-container">

<button type="submit">修改圖書</button>

<button type="button" onclick="window.location.href='../../index.php'" class="btn-back">返回</button>

</div>

<?php if (isset($\_GET['result']) && $\_GET['result'] === 'success'): ?>

<div class="alert alert-success">✅ 書籍修改成功！</div>

<?php elseif (isset($\_GET['result']) && $\_GET['result'] === 'error'): ?>

<div class="alert alert-danger">❌ 修改失敗，請確認 ISBN。</div>

<?php endif; ?>

</form>

<?php elseif ($action === 'view\_logs'): ?>

<form action="borrowlog.php" method="GET" class="form-container">

<label>使用者 ID（可選）:</label>

<input type="text" name="uid">

<label>ISBN（可選）:</label>

<input type="text" name="isbn">

<div class="button-container">

<button type="submit">查詢借閱紀錄</button>

<button type="button" onclick="window.location.href='../../index.php'" class="btn-back">返回</button>

</div>

</form>

<?php else: ?>

<p>⚠️ 請選擇有效的操作。</p>

<button type="button" onclick="window.location.href='../../index.php'" class="btn-back">返回</button>

<?php endif; ?>

</div>

</div>

</main>

<?php require "footer.php"; ?>

</body>

</html>

# add\_book.php

<?php

require '../components/db.php';

$conn = connect();

$isbn = $\_POST['isbn'] ?? '';

$title = $\_POST['title'] ?? '';

$genre = $\_POST['genre'] ?? '';

$year = $\_POST['year'] ?? '';

$author = $\_POST['author'] ?? '';

$pid = $\_POST['pid'] ?? '';

$check = $conn->prepare("SELECT COUNT(\*) FROM book WHERE ISBN = :isbn");

$check->execute([':isbn' => $isbn]);

if ($check->fetchColumn() > 0) {

header("Location: admin\_action.php?action=add&result=exists#add");

exit;

}

if ($isbn && $title && $genre && $year && $author && $pid) {

$stmt = $conn->prepare("INSERT INTO book (ISBN, title, genre, year, author, PID)

VALUES (:isbn, :title, :genre, :year, :author, :pid)");

$stmt->execute([

':isbn' => $isbn,

':title' => $title,

':genre' => $genre,

':year' => $year,

':author' => $author,

':pid' => $pid

]);

header("Location: admin\_action.php?action=add&result=success#add");

} else {

header("Location: admin\_action.php?action=add&result=error#add");

}

exit;

?>

# delete\_book.php

<?php

require '../components/db.php';

$conn = connect();

$isbn = $\_POST['isbn'] ?? '';

if ($isbn) {

$stmt = $conn->prepare("DELETE FROM book WHERE ISBN = :isbn");

$stmt->execute([':isbn' => $isbn]);

header("Location: admin\_action.php?action=delete&result=success#delete");

} else {

header("Location: admin\_action.php?action=delete&result=error#delete");

}

exit;

?>

# update\_book.php

<?php

require '../components/db.php';

$conn = connect();

$isbn = $\_POST['isbn'] ?? '';

$title = $\_POST['title'] ?? '';

$genre = $\_POST['genre'] ?? '';

$year = $\_POST['year'] ?? '';

$author = $\_POST['author'] ?? '';

$pid = $\_POST['pid'] ?? '';

if ($isbn) {

$stmt = $conn->prepare("UPDATE book SET title = :title, genre = :genre, year = :year, author = :author, PID = :pid WHERE ISBN = :isbn");

$stmt->execute([

':isbn' => $isbn,

':title' => $title,

':genre' => $genre,

':year' => $year,

':author' => $author,

':pid' => $pid

]);

header("Location: admin\_action.php?action=update&result=success#update");

} else {

header("Location: admin\_action.php?action=update&result=error#update");

}

exit;

?>

# borrowlog.php

<?php

require '../components/db.php';

$conn = connect();

$uid = $\_POST['uid'] ?? '';

$isbn = $\_POST['isbn'] ?? '';

$sql = "SELECT \* FROM borrowlog WHERE 1=1";

$params = [];

if (!empty($uid)) {

$sql .= " AND UID = :uid";

$params[':uid'] = $uid;

}

if (!empty($isbn)) {

$sql .= " AND ISBN = :isbn";

$params[':isbn'] = $isbn;

}

$stmt = $conn->prepare($sql);

$stmt->execute($params);

$records = $stmt->fetchAll(PDO::FETCH\_ASSOC);

?>

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>借閱記錄</title>

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

</head>

<body>

<?php require "header.php"; ?>

<main>

<div class="main-content">

<div class="log-query">

<h2>📖 借閱紀錄</h2>

<table border="1" cellpadding="8">

<tr>

<th>LID</th>

<th>UID</th>

<th>ISBN</th>

<th>借書日</th>

<th>到期日</th>

<th>還書日</th>

<th>已還</th>

<th>罰鍰</th>

</tr>

<?php foreach ($records as $r): ?>

<tr>

<td><?= $r['LID'] ?></td>

<td><?= $r['UID'] ?></td>

<td><?= $r['ISBN'] ?></td>

<td><?= $r['borrow\_date'] ?></td>

<td><?= $r['due\_date'] ?></td>

<td><?= $r['return\_date'] ?></td>

<td><?= $r['is\_returned'] ? '是' : '否' ?></td>

<td><?= $r['fine'] ?></td>

</tr>

<?php endforeach; ?>

</table>

<button type="button" onclick="window.location.href='../../index.php'" class="btn-back">返回</button>

</div>

</div>

</main>

<?php require "footer.php"; ?>

</body>

</html>