

電影購票系統

第二組

9B117011 殷弘騰

9B117010 陳瑞源

壹、簡介

電影首頁

電影首頁為簡單首頁畫面，使用網路上「Free Website Templates」下載下來的 CSS 進行更改，配合 flask 的框架做了一點小修正。首頁上面有簡單的電影圖片，點選圖片或是電影名稱，頁面會跳到「電影資訊」的頁面。

電影資訊

電影資訊頁面，在網頁板模 CSS 方式使用了 Grid Container，使畫面上可以整齊的排列，並且將電影、劇情介紹及電影分級顯示在同一個區塊，提升用戶閱讀的方便性。

資料庫部分使用了 movies 資料表，建立了 title、director、rating 欄位。

票價資訊

票價部分則簡單的表示出全票、優待票、愛心票、敬老票，並各分為 300 元、250 元、150、150 元，並以表格的方式呈現，

快速訂票

快速訂票頁面會先檢查是否為登入狀態，程式碼一開始會檢查 session 是否有手機號碼，如果無責由路由導向至登入頁面進行登入動作。

如是登入狀態，則會顯示電影列表，可以選擇自己喜歡的電影，點選選擇場次按鈕會跳出一個表單，可以選擇場次時間及座位號碼。在場次部分與資料表 movies 及 showtimes 做連結，當你選擇電影時會判斷是哪一個的場次，並以迴圈方式，將該場次的座位數顯示出來(seats_count)。

票種部份使用了下拉式選單，各票種不得超過 4 張，一個會原最多可以預訂 16 張票，然當座位已被預訂走或是未選擇票種張數時，座位會呈現灰色圖

案為無法選取樣態，當選擇票種張數時會將剩餘座位做呈現(紅色)，則可以進行點選，點選後會顯示為藍色代表已選取，並會在下方顯示選取的座位號碼。

在選擇座位時使用了 javascript 做一個互動的介面，較能模仿一般電影院選擇座位的樣子。

會員資訊

會員登入頁面為一個會員資訊的表單及訂票紀錄的表單，主要用來呈現會員資訊及一個修改密碼的功能、購票紀錄，其保留可擴充性。

登入/註冊

登入 / 註冊頁面，為一個動態的表單，與資料庫做一個互動。表單一開始會要求會員輸入手機號碼，當所輸入的手機號碼在資料表(cinema.db/users)查無時，代表該會員尚未註冊帳號，則按下一步後會跳出註冊頁面表單，依序要求輸入電子郵件、密碼、確認密碼及驗證碼，此時表單的手機號碼為 read-only 狀態，以免在註冊頁面修改導致 bug 出現。創建完成後會將該會員資料寫入資料表中，並且直接登入跳至首頁畫面，登入畫面則為顯示輸入密碼及驗證碼的表單。在登入的同時上方的導覽列會顯示「會員資料」及「登出」按鈕。

在登入 / 註冊使用了正規表示式，包含的驗證手機號碼格式('^09\d{8}\$')是否正確、mail 格式('^([a-zA-Z0-9_+])+@[a-zA-Z0-9]+\.[a-zA-Z0-9-]+\.\$')是否正確、密碼格式('^([0-9a-zA-Z]{8,})\$')是否正確，當輸入錯誤時會顯示錯誤訊息並且需要重新輸入。

在驗證碼的部分，以 0-9 隨機產生四碼的方式，將該四碼存入 session，會與會員在頁面輸入的驗證碼做比對，如果錯誤則顯示錯誤訊息，直到輸入正確為止。當錯誤時會重整頁面及驗證碼。

管理員登入

此次在設計時將會員與管理員做分開。在管理員登入的部分，於一開始會先判斷 session 是否有值，如 session 中的值是手機號碼，則為會員，會將路由導向首頁。如果為空則可以進行管理員帳號密碼登入，此為防範登入會員帳號後又記得網址列，造成多重登入的問題。

管理員介面

管理員介面的呈現與電影介紹類似，只是多了些可以做後台修正的功能，例如新增電影、刪除電影、增修場次資訊導功能，同時保留了可擴充性。

在新增及刪除電影時，會與資料表 `movies` 做連結，針對所選取的電影做刪除或是在資料表中依照欄位輸入電影資料。

增修場次部分使用的是一個表單，分別有時間、日期及座位，是與 `showtimes` 資料表做連結，並且在選擇日期、時間後會在後端做格式的修正，使其為 `YYYY-MM-DD HH:MM` 的格式，以符合場次的要求。

在座位數量，設定了最多為 50 的限制，如輸入大於 50 座位則會顯示錯誤訊息。

貳、開發環境

開發工具

1. Visual Studio Code
2. Windows 11

開發語言

1. Python flask
2. CSS
3. HTML
4. Javascript

參、說明

本次開發的電影管理系統，主要是透過學習前端與後端的互動性，透過前端的頁面模仿一般的使用者所能瀏覽的資訊，加上後臺管理模仿一般的管理員可管理的功能，結合了資料庫的運作。

為了有效的在較短的時間內熟悉相關語言的運用，選擇了一個需要較多功能的系統，從中可以學習到如何建立資料庫、網頁語法的撰寫、網頁路由的導向方式及配合 UI/UX，使功能較為簡便但不失學習性。

肆、實作方法及分組心得

前置作業：一週

開發前先擬定了流程圖，因為是購票系統所以需要有一個會員帳號，所以在設計方面有規劃了使用者的資料庫及使用者登入介面。另外為了需要維護資料庫做處理，也另外規劃了管理員的介面及資料庫，可不需直接開啟資料庫，即可從網頁上直接修訂資料庫資料。其餘的頁面初步設計則是基本的首頁瀏覽，電影簡介瀏覽、票價資訊及訂票功能。

開發階段：四週

初期開發階段均依照上課模式行走，加上有購買的 flask 書籍有相關介紹，例如在登入方面介紹的是使用 flask-login，但因為第三次小考時有用到 session 後發現功能可以更為簡便，且開發初期製作出來驗證碼功能也是使用 session 來儲存，所以就在登入時改用 session 來保存登入號碼。

其餘較需技術面的部份在於訂票處理以及管理者模式的功能，訂票處理因為想達成比較具有互動性的頁面，所以使用了 javascript 來製作，有參考了網路上類似製作訂位系統的介紹。

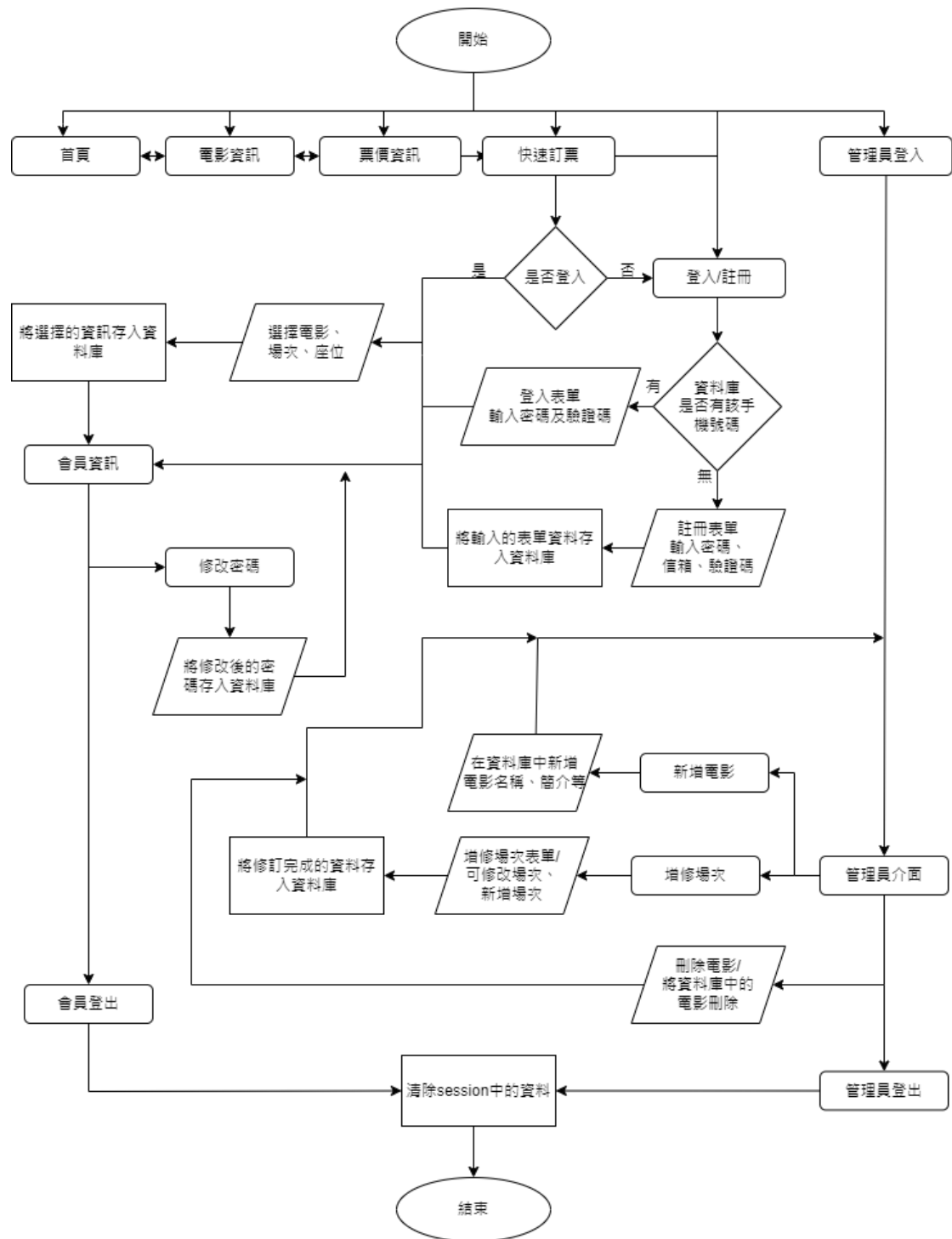
而在管理者介面，修改了多次頁面，主要是最終版本時考量到製作與電影資訊相同頁面可以節省較多時間，加上可以在部分空為上添加管理者的功能、CRUD、可以減少一些路由及網頁製作，且在修訂後看到的介面內容就是使用者可以看到的內容，相當方便。

收尾階段：二週

在製作完成後，我們進行了多次了測試來尋找 BUG，例如在選單部分設定了使用者防呆、輸入介面的限制與相對應的錯誤訊息，測試了兩週，稍微修改了部分的程式碼及版型，使介面呈現比較具有一致性。

而在找 BUG 中，因為組員對於程式碼不太熟悉，所以請他當一個使用者的角度來做測試，以致於可以從使用者體驗上來進行修改，或是提供哪裡需要修改的部分。

伍、相關圖表



圖一、程式流程圖

資料表 html	admin	bookings	buy	movies	seats	showtimes	users
addmovies				✓			
admin_login	✓						
booking							
buy			✓	✓	✓	✓	
chpwd							✓
editshowtime						✓	
index							
login							✓
member			✓				✓
movies				✓			
movielist				✓			
ticket							

表一、資料庫使用情況

組員 工作內容	殷弘騰	陳瑞源
程式架構、流程圖	100%	0%
程式碼	100%	0%
資料查詢	100%	0%
文件	60%	40%
簡報	60%	40%
測試除錯	50%	50%

表二、工作分配

陸、參考文獻

1. https://www.w3schools.com/css/css_grid_container.asp
2. <https://ithelp.ithome.com.tw/articles/10318320?sc=rss.iron>
3. https://blog.csdn.net/qq_43248263/article/details/121558331
4. https://hackmd.io/@chrislinlin/chooseSeats_cinema
5. chatgpt、copilot
6. <https://freewebsitetemplates.com/>
7. Flask 網頁開發第二版 ISBN:978-986-476-868-4
8. https://hackmd.io/@shaoeChen/ryvr_ly8f?type=view