

# 逢甲大學

資訊工程學系專題報告

逢甲大學電腦機房  
借用及管理系統

學生：劉士億（四丁）

指導教授：顧偉倫 老師

中華民國九十二年十二月

## 目錄

### 第一章 研究動機與目的

1-1 研究動機.....P 4

1-2 研究目的.....P 5

### 第二章 系統分析

2-1 系統簡介.....P 6

2-2 可行性分析.....P 7

2-3 開發工具簡介

2-3.1 IIS Server.....P 11

2-3.2 Microsoft SQL Server 2000.....P 15

2-3.3 ASP.....P 17

2-4 甘特圖.....P 18

### 第三章 系統設計

3-1 系統架構.....P 19

3-2 DFD 流程圖.....P 21

3-3 Task Analysis 圖.....P 24

3-4 SFD 流程圖.....P 31

3-5 STD 流程圖.....P 34

3-6 Data Element Table.....P 35

3-7 User Interface Control.....P 39

3-8 資料庫結構.....P 41

### 第四章 成品展示

### 第五章 心得

### 附錄

### 參考資料

## 表目錄

1. ASP 與 PHP 之比較	09
2. 圖 2-4 甘特圖	18
3. 表 3-6.1 課表紀錄 table	35
4. 表 3-6.2 固定排課 table	35
5. 表 3-6.3 LOG 紀錄檔 table	36
6. 表 3-6.4 教職員申請 table	37
7. 表 3-6.5 電腦機房資訊 table	37
8. 表 3-6.6 上課時段 table	37
9. 表 3-6.7 教職員所屬單位 table	37
10. 表 3-6.8 管理者帳號設定 table	38
11. 表 3-6.9 假日設定 table	38



## 圖目錄

1. ASP 作業示意圖	17
2. 甘特圖	18
2. 圖 3-1 系統架構圖	19
3. 圖 3-2 管理者登入架構圖	20
4. 圖 3-3.1 DFD 圖	21
5. 圖 3-3.2 教職員部分 DFD 圖	22
6. 圖 3-3.3 教職員部分 DFD 圖展開	22
7. 圖 3-3.4 管理者部分 DFD 圖	23
8. 圖 3-3.5 管理者部分 DFD 圖展開	23
9. 圖 3-4.1 Task Analysis 教職員部分全圖	24
10. 圖 3-4.2 Task Analysis 頂層圖	25
11. 圖 3-4.3 Task Analysis 程序 2 展開圖	26
12. 圖 3-4.4 Task Analysis 程序 3 展開圖	27
13. 圖 3-4.5 Task Analysis 管理者登入展開圖	28
14. 圖 3-4.6 Task Analysis 程序 4.2 展開圖	29
15. 圖 3-4.7 Task Analysis 程序 4.3 展開圖	30
16. 圖 3-4.1 SFD 頂層圖	31
17. 圖 3-4.2 SFD 電腦教室查詢部分圖	32
18. 圖 3-4.3 SFD 教職員申請借用部分圖	32
19. 圖 3-4.4 SFD 管理者登入圖	33
20. 圖 3-5.1 STD 流程圖 教職員借用機房部分	34
21. 圖 3-5.2 STD 流程圖 管理者管理機房部分	34
22. 圖 3-7.1 使用者查詢畫面	40
23. 圖 3-7.2 教職員借用畫面	40
24. 圖 4-1 系統主畫面	44
25. 圖 4-2 電腦教室課表查詢畫面	45
26. 圖 4-3 教職員借用電腦教室畫面	46
27. 圖 4-4 管理者登入畫面	47

# 第一章 研究動機與目的

## 1-1 研究動機

因在學校的資源管理中心工讀，並於工讀期間獲得辦公室主管及督導肯定，而被指派為工讀生幹部，藉此了解許多學校電腦機房相關的行政作業。恰巧，本學期初本校的線上借用電腦機房系統故障，工讀生及督導們亦覺得原本的系統太過老舊，有很多資料必須做更新，故心中有了想要製作新的電腦機房線上借用管理系統的想法，經過和顧偉倫老師討論之後，認為此系統將有助於日後教授們借用電腦機房時更加方便，專題題目就此確定。

早年的教師要預借電腦機房時，大多都是手工作業，以現在的眼光及角度看來是十分費時費力，又容易產生錯誤。現在電腦與網路非常普及，尤其是現在政府事事都提倡e化，本校亦在這方面下了不少苦工。但若是系統介面設計不良，將會造成使用者操作時的疑惑和困擾，讓原本加速處理過程、降低文書成本、提高對使用者服務等等美意失去其應有的意義了。

## 1-2 研究目的

因為學校使用的軟體必須注意到安全性、維修方便、以及實用程度，故我採用了本校也在使用的 Microsoft SQL Server 2000 來開發資料庫系統，外部介面的部分則採用了以 IIS 為基礎的 ASP 語法寫作，因為 PHP 雖然具有較佳的跨平台能力，但是程式語法複雜，維修十分不易，故不採用。

希望能藉由這套系統，能帶給教授們借用電腦機房時更多便利與資訊，以及更方便督導們精確掌握電腦機房各項配備的汰舊及更新，透過表格、圖形、文字等表達方式，讓使用者能更方便、更人性化來操作此「逢甲大學電腦機房借用及管理系統」。



## 第二章 系統分析

### 2-1 系統簡介

本系統是以 ASP 搭配 Microsoft SQL Server 2000 與 IIS WEB Server 完成的，主要是改進舊系統單調的畫面，並且修改了使用者操作介面上的缺點，將部份功能進行整合，讓使用者更佳方便。

舊系統的缺點條列如下：

1. 若教授要借用電腦機房超過一個時段，必須重複輸入數次教職員證號及密碼。
2. 教授若要查詢電腦機房的課表和已經借用的時段，要分開兩次查詢。
3. 目前替代系統無法查詢借用狀況。
4. 介面操作不方便，很多地方必須手動輸入。

## 2-2 可行性分析

實做這個系統之前的評估過程，就是要在有限的經濟與能力下，在眾多選擇方案中，去選擇出一個最佳的解決方案，做出最有效率以及符合使用者需求的系統，是非常重要的。

從加權評估法中，挑選幾個選擇標準作為我們評估的主要依據，其選擇標準如下：

### 2-2.1 經濟可行性

經濟的可行性係指所需的費用以及所獲得的效益，通常是會進行成本效益的分析，此乃可行性分析中最重要的一環，因此以降低成本來追求高效益將是實做的努力方向。

在實做程式的花費上，由於是採用 Microsoft SQL Server 2000+ASP+IIS 的網路伺服器環境，所以部分成本是屬於免費或內建於作業系統中，因此在軟體取得方面，除了 Windows 的作業系統之外，所花費的可說是相當少。

至於參考書籍的花費，除了可以到學校圖書館、省立圖書館、西屯區圖書館借閱書籍之外，顧偉倫老師和資源管理中心的黃東炳先生亦提供了相當多的參考資料，只額外買了兩本參考書，就總花費而言，是相當精簡的。

### 2-2.2 時間可行性

由於這次所開發的網站主要是以本校資源管理中心為主要開發對象，因此在開發時間上，必須盡量去考慮到學校行政作業需要的急迫性，所幸目前有暫時的備用系統，時間考量上不成問題，但因備用系統的功能實在太簡陋，所以仍要盡早完成。



### 2-2.3 政策可行性

在政策的評估上，最主要要考量到不僅是網站的使用者、督導，尚須考量到以後負責控管維護的人員，因此所需考量的接受程度，也必須做最大多數人的考量情形。

因此再設計本系統時，盡量以使用者的角度來設計網站，也盡量以直覺化的設計來設計本系統，亦即『所見即所得』的直覺化設計，也試著融入多一些人性化的介面設計，以按鈕和選單取代輸入選項，以圖形介面取代文字介面…等更方便使用者的設計。

在指導老師方面，由於顧老師曾多次使用舊的電腦機房預借系統，指出了很多舊系統使用不方便與設計不合理的地方，因此在政策的可行性上，著重於使用者接受程度上。

### 2-2.4 維護永續性

就系統維護而言，包含了軟體硬體和維護人員等方面。

就軟體和硬體方面，所需要的硬體設備和系統軟體，在資源管理中心都有充足的資源，資源管理中心的黃東炳先生亦提供了一部電腦當作日後本系統的網路伺服器，而黃先生本人則負責維護本系統，未來維護不構成問題。

### 2-2.5 作業可行性

資源管理中心的督導們，對於本系統抱有相當大的期許態度，也因此大力支援本系統作業時所需的軟硬體設備，在運作上面沒有太大的困難，可行性是沒有問題的。

### 2-2.6 開發工具選擇

由於學校現行的系統大多都有和微軟簽約，加上目前現行的系統皆使用微軟的 SQL Server 2000，應資源管理中心方面要求，爲了方

便資料轉移，故資料庫部分採用 SQL Server 2000，接下來是要決定和資料庫配合的語言。

目前最多使用者的當屬 PHP 和 ASP 兩大系統，下面列出這兩者的比較。

項目	ASP	PHP
作業平台	Windows 98 Windows NT 或更新版之 OS	除麥金塔之外，可以說跨平台。
Web Server	Microsoft IIS/Personal Web Server	Apache、fhttpd、Netscape、IIS、PWS、Omni
物件導向	可以使用物件	可以使用物件
語法	可以結合 VBScript、JavaScript、或是 Perl Script 語法 like Visual Basic	有自己的語法，相當複雜。
資料庫	透過 ODBC、ADO 存取	直接 compile 成資料庫 Client 端
成本	OS(Windows 系列) Server(IIS 內附於 Windows) ASP(free) Database(SQL Server 2000)	OS(最低 free) Server(最低 free) PHP(free) Database(MySQL free)
Open Source	No	Yes

表 2-2 ASP 與 PHP 之比較圖

就上圖而言，PHP 似乎在成本上大佔優勢。不過由於學校方面早就已經有 Microsoft 相關軟體，ASP 弱勢的成本部份頓時變成零成本，加上攥寫程式和維修的難易程度考量，決定採用 ASP 為本系統開發語言。



## 2-3 開發工具簡介

### 2-3.1 IIS Server

#### 2-3.1.1 Internet Information Server 簡介

微軟的 Internet Information Server (IIS) 與 Windows NT Server 作業系統緊密地整合在一起。他 (IIS) 是為了確保使用團體在建置 Internet 或是 Intranet 的高度安全性而設計。另外也提供 Server Sockets Layer 3.0 (SSL) 良好的執行效益，以及 X.509 RSA Public Key Clipher 的安全認證，保障使用者在網路上溝通的權益。由於嚴格的安全管制，使得管理 IIS 相當容易，因此無論是使用 Internet 或是 Intranet 都可以在不同的平台上迅速地建置網站 (Web Site)。這份白皮書就是提供讀者 IIS 使用的安全性模組概要。

微軟的 Internet Information Server (IIS) 與 Windows NT Server 作業系統緊密地整合在一起。目的是提供使用組織最有利的 Web Site 以獲得 Internet 或是 Intranet 所帶來的優勢，以及確保應用程式與資訊的高度安全性。

整合意味著 IIS 提供與 Windows NT 相同健全的安全性，微軟的 Windows NT 符合美國政府的 C2 安全規定 (U.S. Government's C2 Security Evaluation)，因此 IIS 的系統管理者不但可充分利用作業系統及的安全機制，並可在單一整合的安全控管模式下，作簡單而有效的管理。

### 2-3.1.2 整合的安全性

Windows NT Server 所使用的安全傳輸架構適用於安裝在其下的系統元件，透過確認身份的方式可以控制使用者存取所有的系統資源。IIS 整合了 Windows NT 安全模組以及作業系統的服務，像檔案系統與目錄。因為 IIS 可以利用 Windows NT Server 的使用者資料庫，所以管理者不需要在每一個網站上建立額外的使用者帳號，並且建置在 Intranet 中的使用者只要登錄網路一次，便可以合法地存取系統資源。IIS 會自動地根據已經存在檔案伺服器、列印伺服器以及應用程式中使用者的權限，正確地讓每一位使用者存取系統資源。

有些網路伺服器將安全性建置在作業系統的最上層，因為缺乏整合性與一致性，容易產生額外虛有的帳號，致使安全性暴露在外。Windows NT Server 設計的安全性是絕對可以信賴的，任何系統資源只允許有正式合法權限的使用者可以存取，而所有使用者的帳號都是透過公用系統給予唯一的識別證。當刪除部分帳號之後，該使用者所有的存取權限，包括在群組中的權限，都一併刪除。即使下一個新進人員使用曾經被刪除的帳號，該帳號的使用者並不會繼承上一個使用者的權限。

### 2-3.1.3 簡易的管理方式

對於使用者存取檔案與目錄的權限，可以按照實際需要設定，因為 IIS 就跟其他 Windows Service（例如檔案共享或是 Microsoft SQL Server TM）一樣，使用相同的 Windows NT Server 中的存取控制清單（Access Control Lists, ACLs）。網路伺服器的存取權限並不會針對其他的檔案服務而有不同的設定，相同檔案即使在不同的通訊協定（例如 FTP、CIFS/SMB 或是 NFS 等協定）中傳送也不會出問題，因此管理者不需要重複設定。

Internet 提供數以百計的服務均可經由單一的公用系統工具統籌管理，因此，管理者真的不需要去維護數個內容重複的使用者資料庫。只有 IIS 與 Windows NT 4.0 可以提供最簡易的管理程序，管理者只需動一動滑鼠，就可以賦予使用者存取網路資源（例如網頁、共享的檔案、列表機、資料庫以及繼承所有伺服器上的應用程式）的權限。

IIS 會產生網路伺服器的存取紀錄，與 Windows NT Server 整合意味著 IIS 可以利用系統審查之便，追蹤誰使用了哪些系統資源。舉例來說，若想嘗試存取某個檔案，即使失敗了，在 Windows NT 登錄事件（Event Log）中必定會留下這次的存取紀錄，並且利用管理現存伺服器相同的工具審核每一筆記錄。

#### 2-3.1.4 包羅萬象的解決方案

IIS 利用與 Microsoft Proxy Server、Certificate Server、Site Server、BackOffice 以及其他應用程式緊密結合之便，提供一套內建整合性、極度安全的作業平台。

不論是對外公開 Internet 或是內部 Intranet 網站伺服器，只有 IIS 與 Windows NT Server 才能提供綜合性的作業平台，作為建置快速、健全、完善的網路伺服器之用。建置網路服務是 Windows NT 作業系統的重要核心功能之一，Microsoft IIS 的特色就是高度的效益、簡易的管理以及更好的安全性，就是這些優勢讓 IIS 成為建置網站的最佳平台，如此一來，不但整合了現有的解決方案，同時也產生新一代的網站應用程式。

## 2-3.2 Microsoft SQL Server 2000

主從架構關連式資料庫管理系統，SQL Server 是特別針對主從運算所設計的高伸縮性高效率的資料庫管理系統。他內建的資料複製功能，強有力的管理工具，以及開放的架構，賦予高階平台一個經濟又實惠的資訊解決方案。在企業致力追求有效率的流程與分散決策風險時，他們顯然越來越依賴那些能夠使用者與資訊整合的技術，其結果是企業組織逐漸已分散式運算作為資料與決策制訂的重要依據。

事實上，SQL Server 原本即是針對分散是主從運算所設計之關連式資料庫管理系統，在成功的發展基礎上，SQL Server 更針對企業在傳統上最需要的高效率，可靠度與伸縮彈性做了極大程度的擴充，以應快速變遷下的種種挑戰。

能夠讓您集中管理所有的資料庫伺服器，才是理想的分散是管理架構，藉助新進的是窗管理工具與功能強悍的伺服器作業排程系統，您能夠以視覺化的方式輕鬆控制多部伺服器，並而自動啟動分散是環境中的遠端作業。內建的資料複製功能，提供一種強大且可告的方式將所有資訊正確的發佈至企業的每一個角落。重要的是此一功能並未僅僅侷限於 SQL Server 的資料庫，它可以涵蓋其他多種資料庫。當設計 SQL Server 時，我們仍然心繫 Internet，因此他能以及高的速度存取您公司 Web 的頁面，而其新提供的 Web Assistant 更可以透過多種方式將 SQL 資料發佈至你的 Web 伺服器，以及將公司資料分散至內部的 Intranet 或 Web 的全球資訊網上。



SQL Server 的另外一項新功能就是分散是異動交易協調員，他能自動化橫跨多部伺服器的異動交易管理，這不僅使分散式應用程式更易建立，也讓程式設計師大大節省過去為了達到此項功能，而必須個別替前端程式撰寫特定程式碼所耗費的時間，SQL Server 同時也滿足了您對可靠度與安全性的嚴苛要求，舉凡是資料參考完整性與安全性等，都可藉著 ANIS. FIPS. NIST 等工業標準來達成。

與 Windows NT 緊密的整合，乃是達成高效率與高可靠度的主要原因，除此之外。身為 Back Office 家族的成員之一，SQL Server 當然可以與其他 Back Office 伺服器應用程式整合以成就全方為的主從架構系統。

不管是在前端或伺服器端 SQL Server 都擁有非常開放的介面，您不僅可以從他所提供的各項工具獲益，更能夠快速且輕易地將 SQL Server 與您既有得系統結合在一起。

### 2-3.3 ASP

- 是一種伺服器端指令執行環境，可以讓您用來建立動態網頁或建置功能強大的 Web 應用程式。
- ASP 網頁是包含 HTML 標記、文字和指令的檔案。
- ASP 網頁可以呼叫 ActiveX 元件來執行工作，例如連接到資料庫或執行商業計算。
- 無瀏覽器相容的問題。
- 程式碼隱藏，Client 端僅能看到由 ASP 所輸出的動態 HTML 文件 有了 ASP 之後，您可以新增互動式內容到網頁，或者建立完全以 HTML 網頁當成與客戶之間的介面的 Web 應用程式。

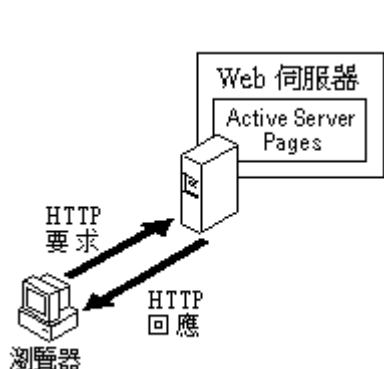


圖 2-3 ASP 作業示意圖

當客戶端瀏覽器對 Web 伺服器發出一個 .asp 檔案的要求時，Web 伺服器會呼叫這個 ASP，然後從頭到尾讀取要求的檔案、執行任何指令，最後傳送標準的 HTML 網頁給客戶端的瀏覽器。

因為指令是在 Server 上執行，而不是在用戶端，也就是由 Server 來負責網頁的產生，經過 Server 的處理後，所產生的網頁乃是以標準 HTML 的格式送至 Client 端，所以網頁設計者不需要擔心 Client 端的瀏覽器是否能處理網頁的問題，而使用者在觀看網頁時也無法查看網頁是如何建構而成的。

## 2-4 甘特圖

工作名稱	開始日期	結束日期	工作天數
詢問使用者需求	2003/10/06	2003/10/9	4
繪製流程圖	2003/10/10	2003/10/15	6
程式實作	2003/10/16	2003/11/01	16
書面報告	2003/11/02	2003/11/16	15

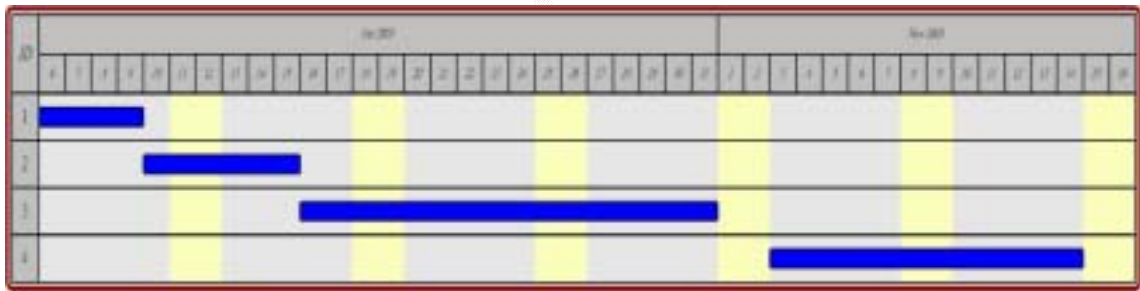
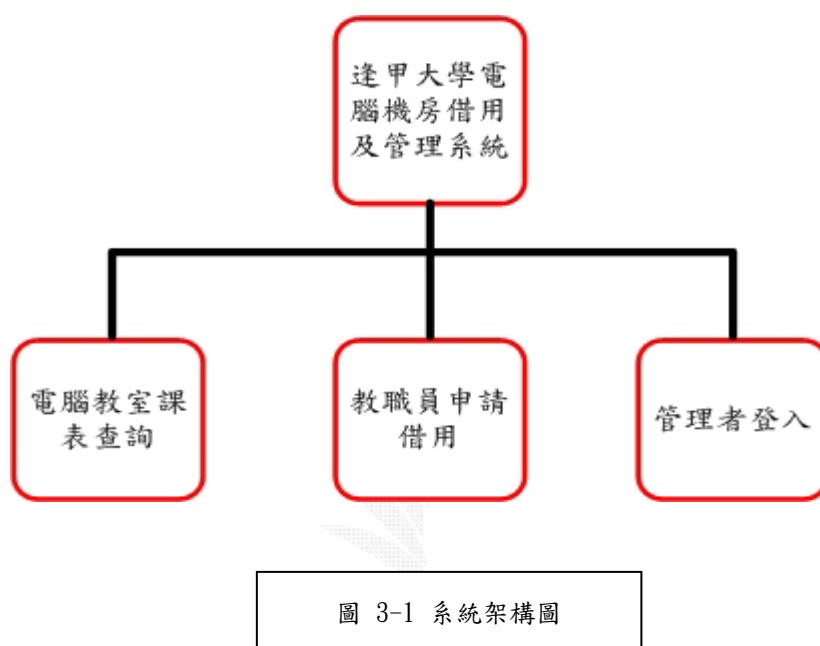


圖 2-4 甘特圖

## 第三章 系統設計

### 3-1 系統架構

以功能來區分，本系統分為三大部分，如圖 3-1。



#### 1. 電腦教室課表查詢部分：

提供使用者查詢所有電腦機房的使用情況，系統會列出使用者所選擇日期的該週所有當間機房的課表，使用者可以點選「上週」和「下週」按鈕查詢所需要的資訊。

#### 2. 教職員申請借用部分：

提供教職員線上提出借用電腦教室申請，只需要簡單的點選所屬單位，欲借用的電腦機房，借用事由，以及預借人的姓名、分機、Email 位址等等，即可完成借用手續，等候審核。

### 3. 管理者登入：

提供資源管理中心的督導使用，以審查借用人所提出的申請。見圖 3-2。

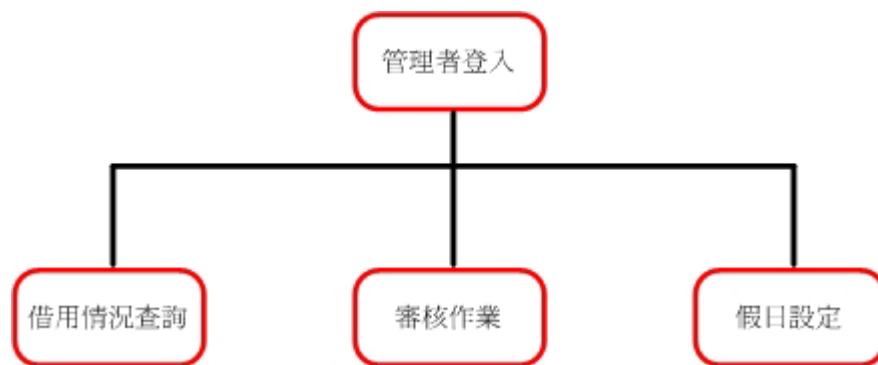


圖 3-2 管理者登入架構圖

### 3-2 DFD 流程圖

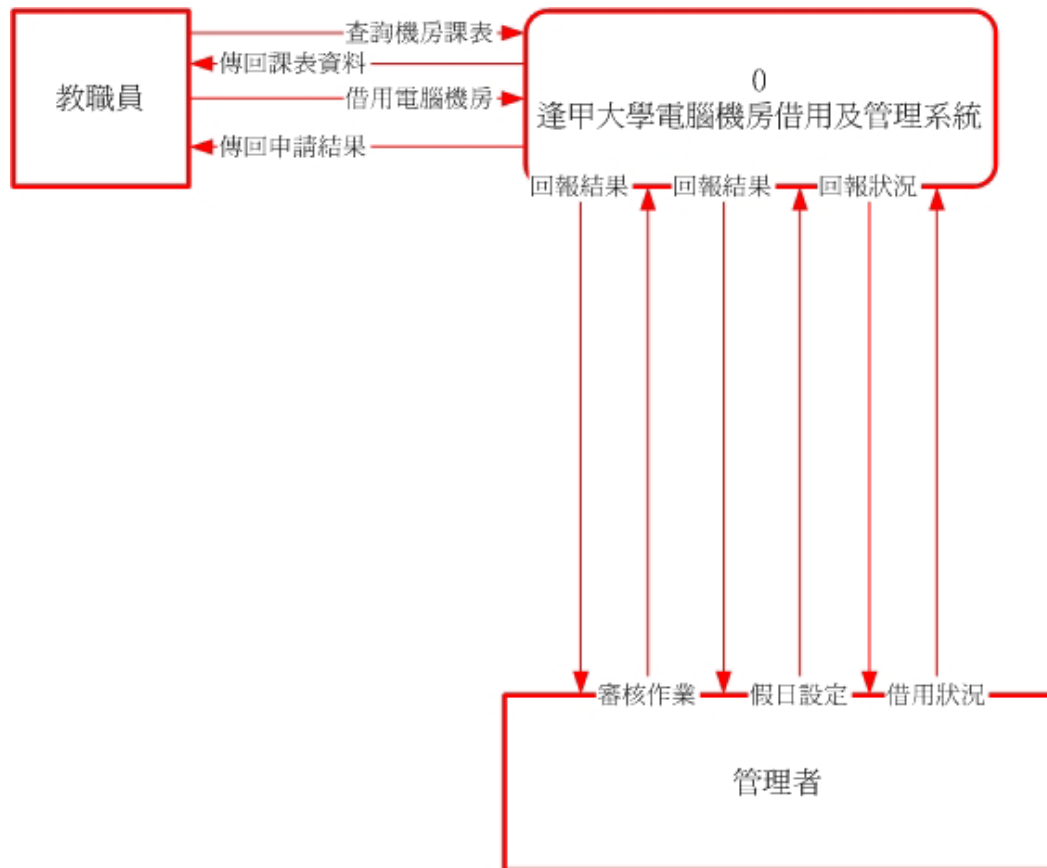


圖 3-3.1 DFD 圖

單就使用者而言，其 DFD 圖如下：



圖 3-3.2 教職員部分 DFD 圖

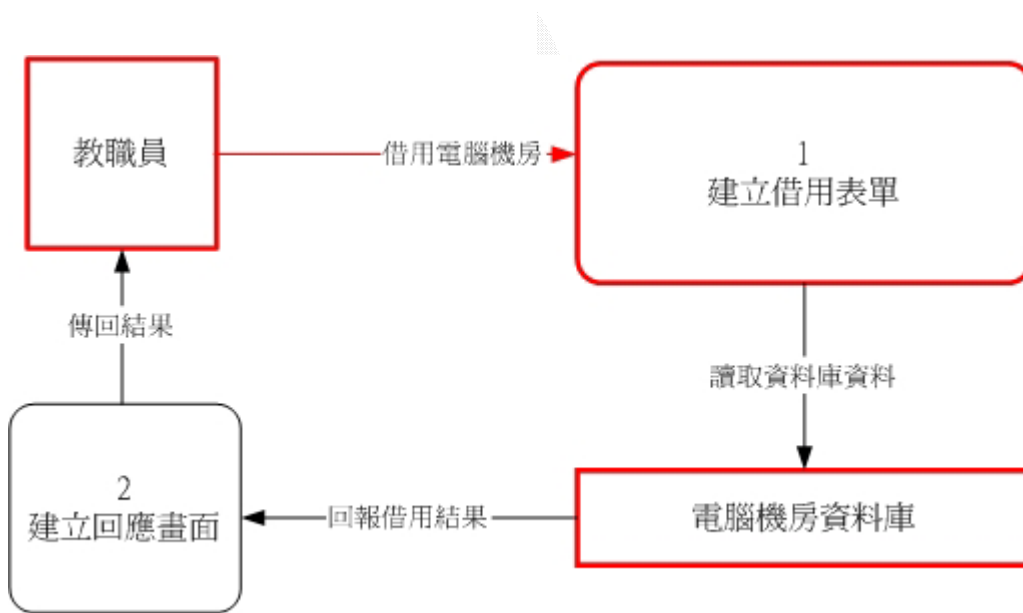


圖 3-3.3 教職員部分 DFD 圖展開

對管理者而言，其 DFD 圖如下：

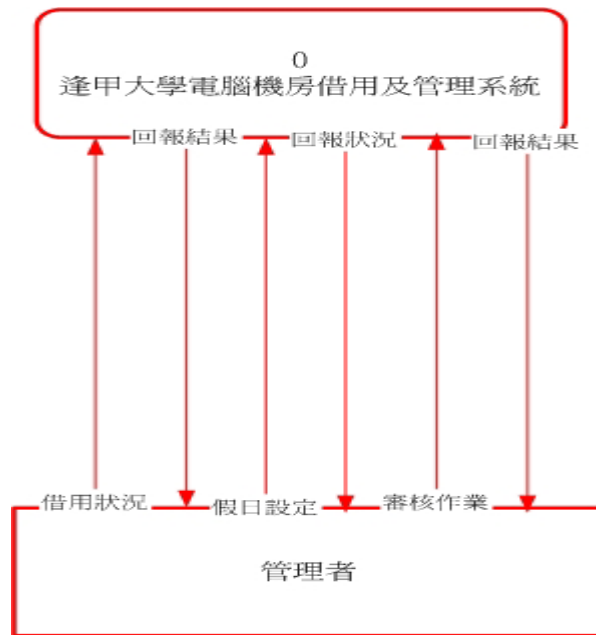


圖 3-3.4 管理者部分 DFD 圖

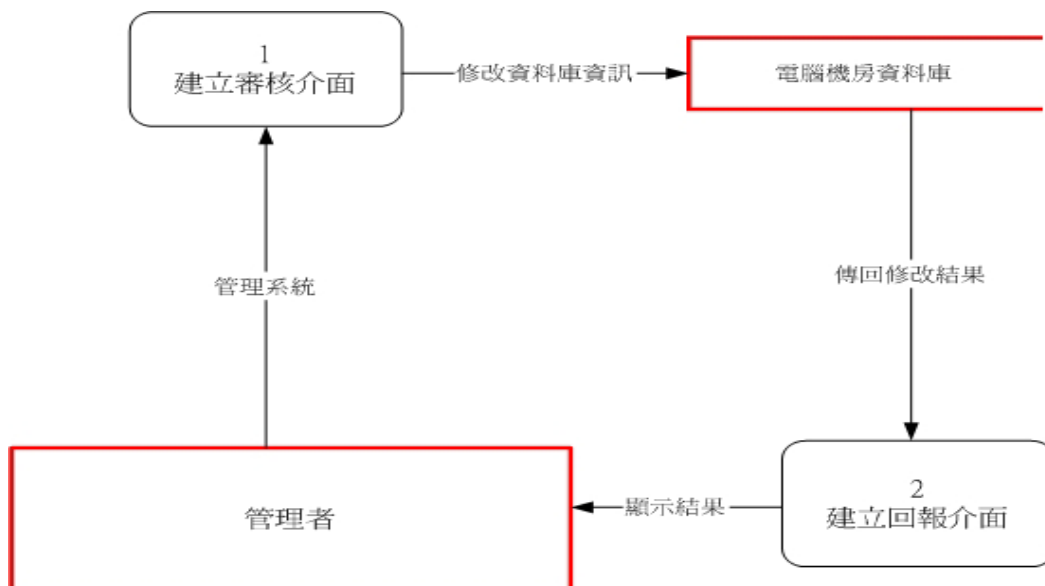




圖 3-3.5 管理者部分 DFD 圖展開

### 3-3 Task Analysis 圖

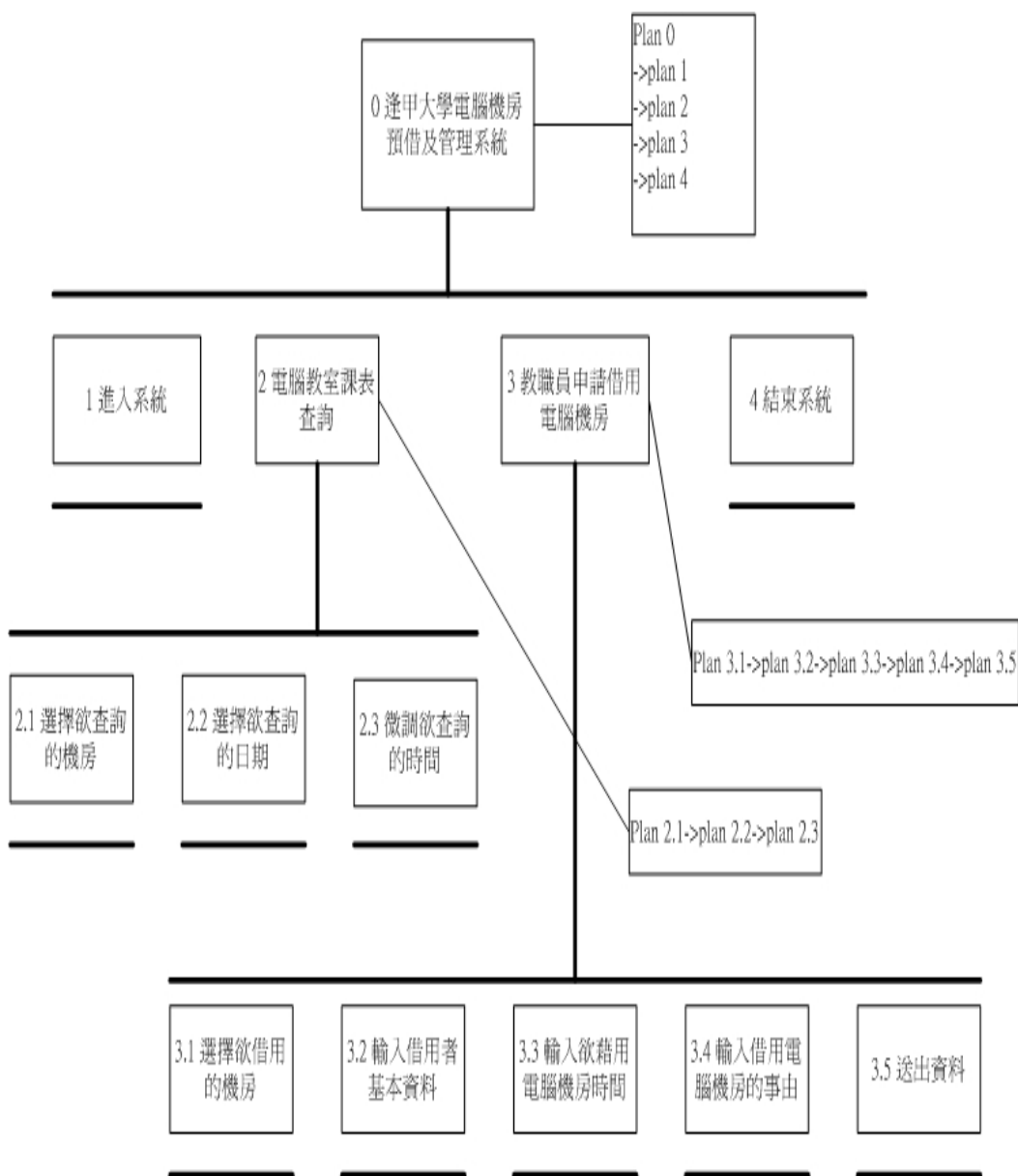


圖 3-4.1 Task Analysis 教職員部分全圖

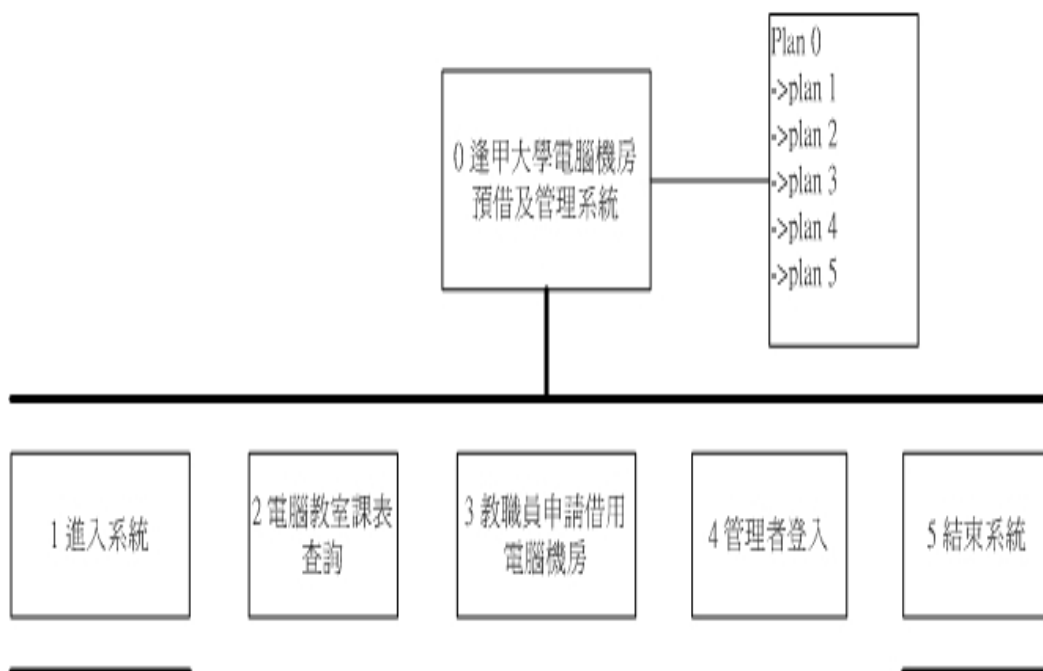


圖 3-4.2 Task Analysis 頂層圖

使用者操作流程：

- 一般使用系統：進入系統 → 查詢電腦教室課表 → 教職員申請借用點電腦機房 → 結束系統
- 直接借用系統：進入系統 → 教職員申請借用點電腦機房 → 結束系統
- 管理者進行管理作業：進入系統 → 管理者登入 → 結束系統

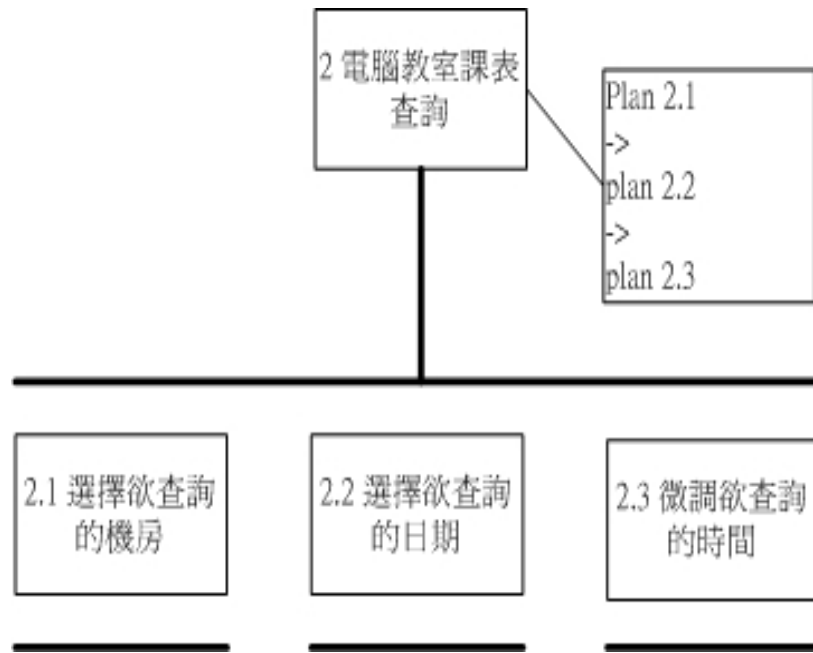


圖 3-4.3 Task Analysis 程序 2 展開圖

使用者操作流程：

- 電腦教室課表查詢：進入系統 → 查詢電腦教室課表 → 選擇欲查詢的機房 → 選擇欲查詢的日期 → 選擇欲查詢的時間

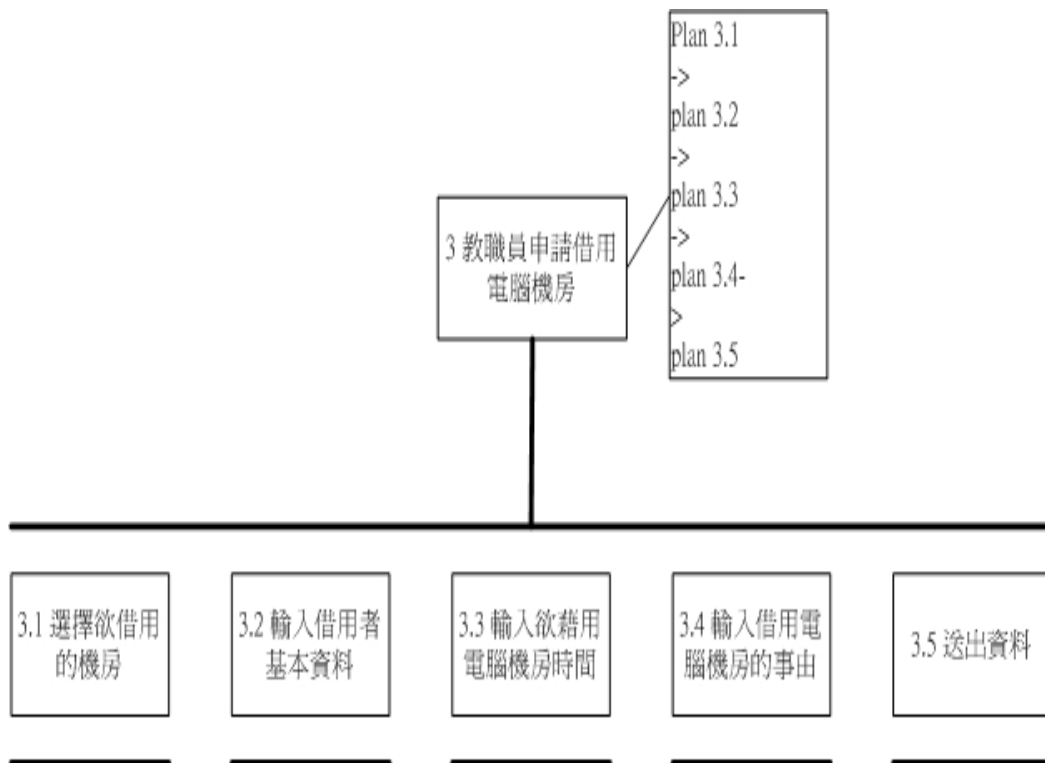


圖 3-4.4 Task Analysis 程序 3 展開圖

使用者操作流程：

- 教職員借用電腦機房：進入系統 → 教職員借用電腦機房 → 選擇欲查詢的機房 → 輸入借用者基本資料 → 輸入欲借用電腦機房的時間 → 輸入借用電腦機房的事由 → 送出資料

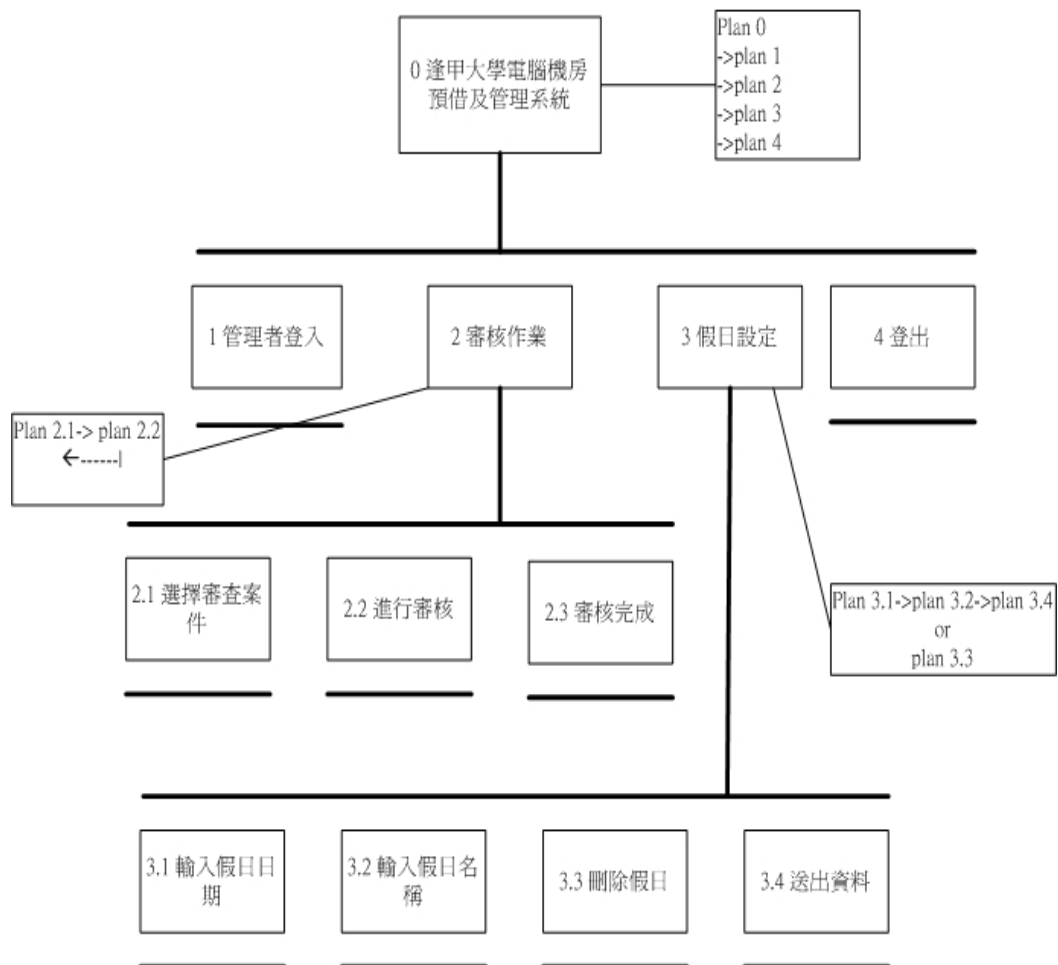


圖 3-4.5 Task Analysis 管理者登入展開圖

使用者操作流程：

- 管理者審核作業： 管理者登入 → 審核作業 → 登出
- 管理者假日機房不開放設定： 管理者登入 → 假日設定 → 登出

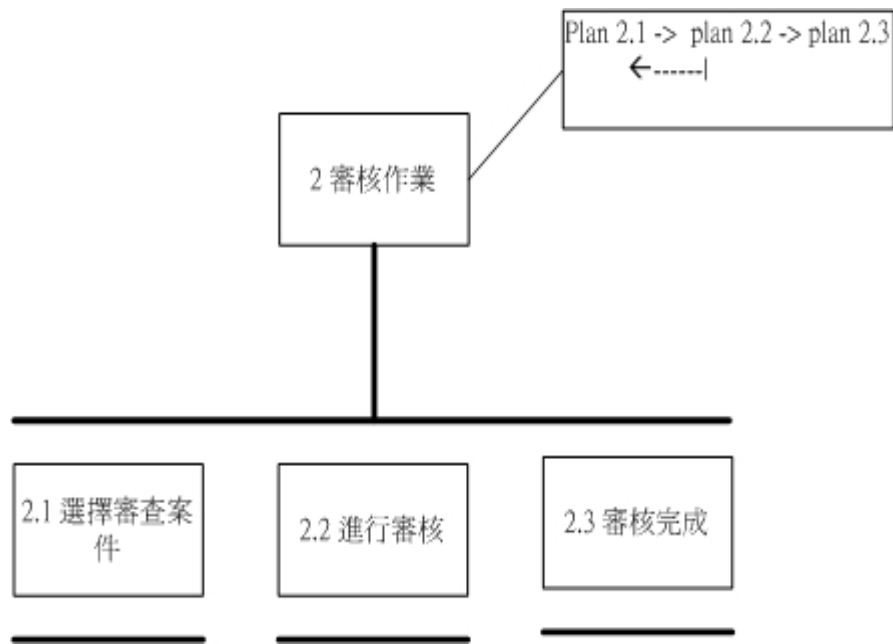


圖 3-4.6 Task Analysis 程序 4.2 展開圖

使用者操作流程：

- 管理者審核預借資料： 管理者登入 → 審核作業 → 選擇審查案件 → 進行審核 → 審核完成 → 登出  
←

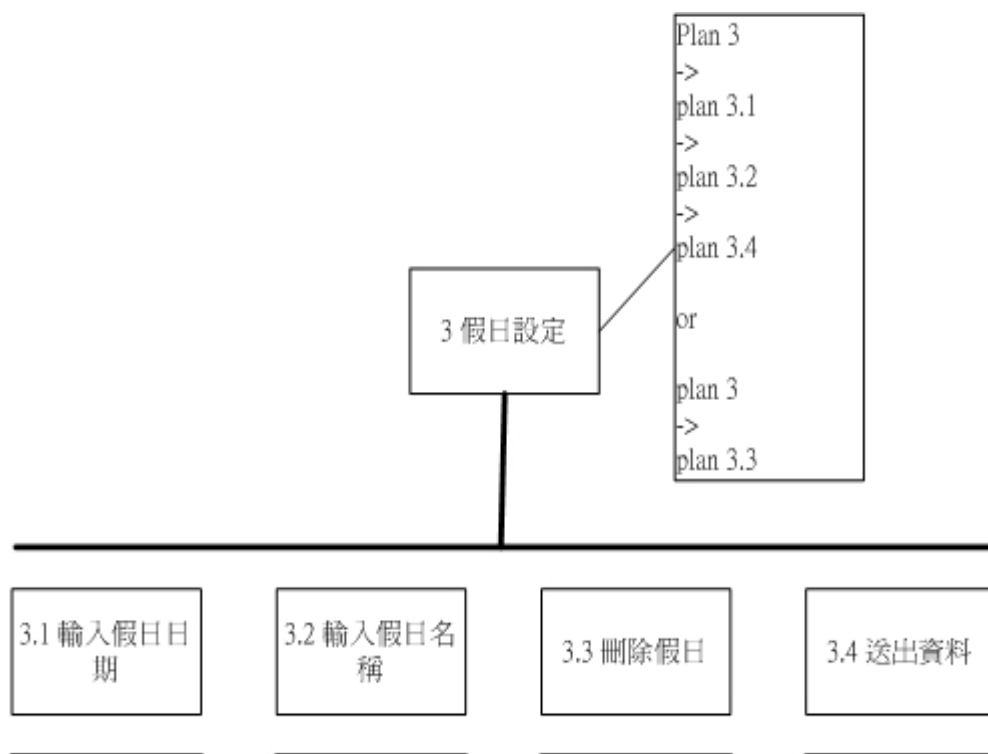


圖 3-4.7 Task Analysis 程序 4.3 展開圖

### 使用者操作流程：

- 管理者假日電腦機房不開放設定： 管理者登入 → 假日設定 → 輸入假日日期 → 輸入假日名稱 → 送出資料 → 登出
- 管理者刪除假日電腦機房不開放設定： 管理者登入 → 假日設定 → 刪除假日 → 送出資料 → 登出

### 3-4 SFD 流程圖

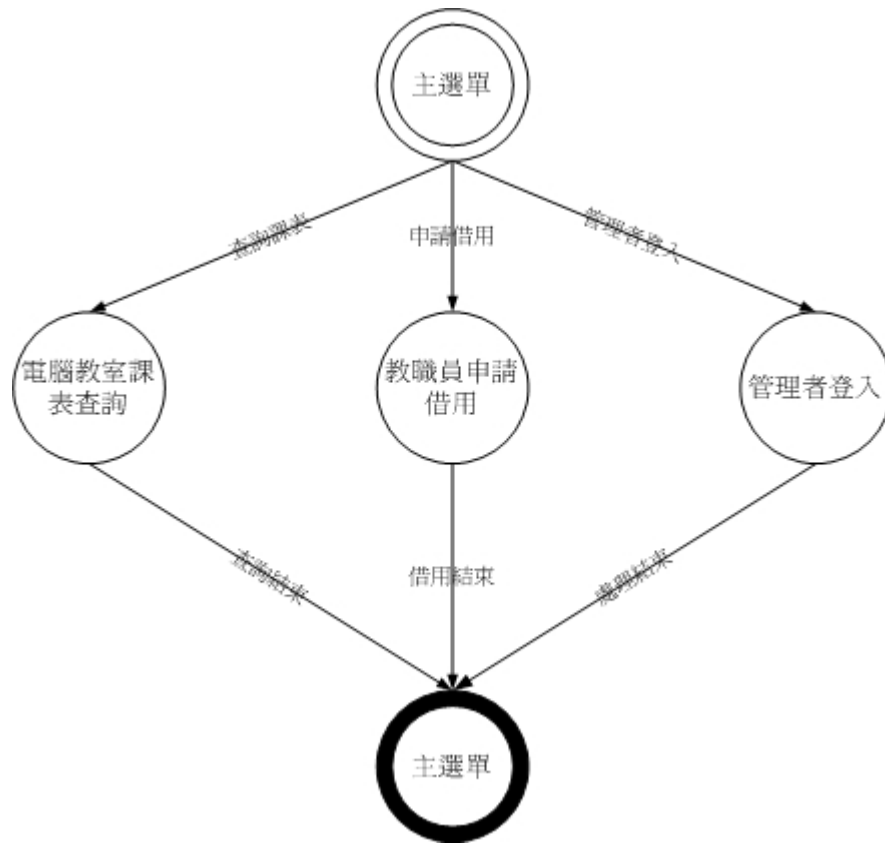


圖 3-4.1 SFD 頂層圖



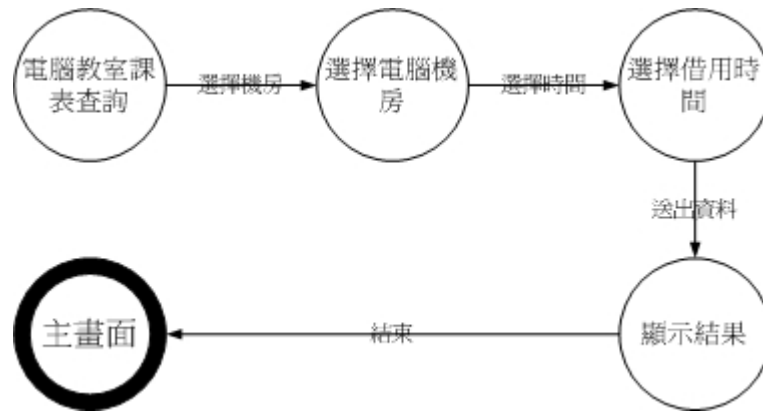


圖 3-4.2 SFD 電腦教室查詢部分圖

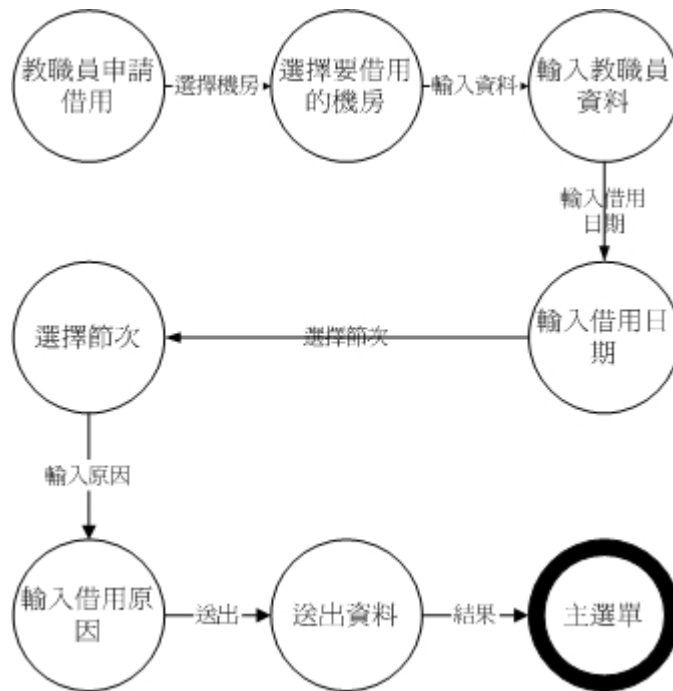


圖 3-4.3 SFD 教職員申請借用部分圖

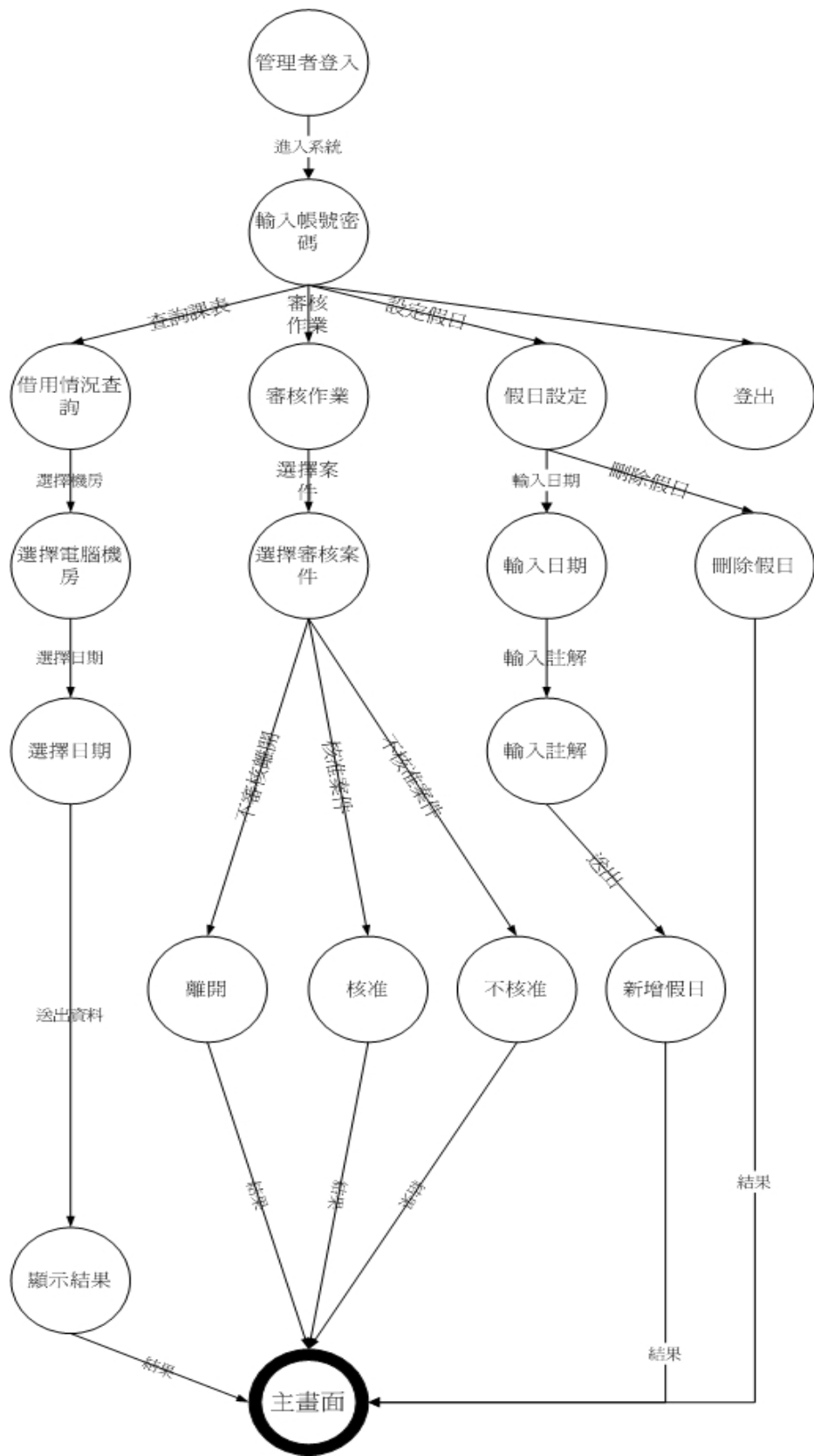


圖 3-4.4 SFD 管理者登入圖

### 3-5 STD 流程圖

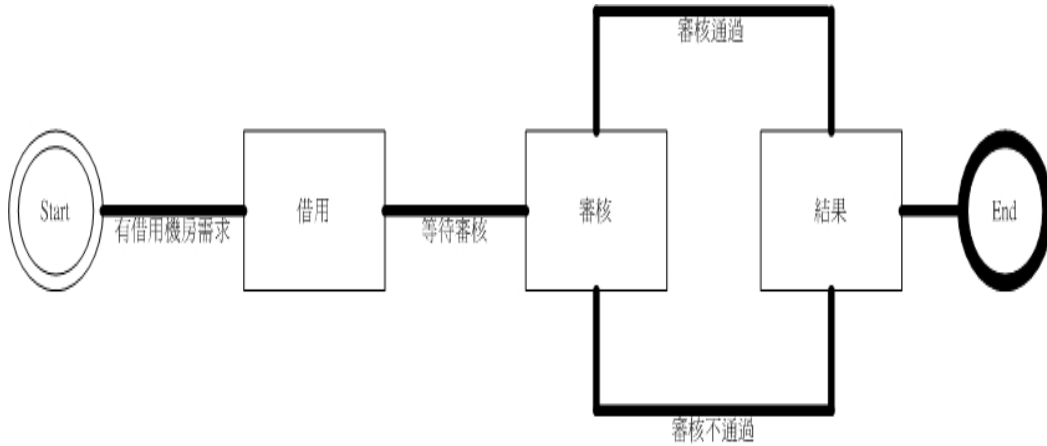


圖 3-5.1 STD 流程圖 教職員借用機房部分

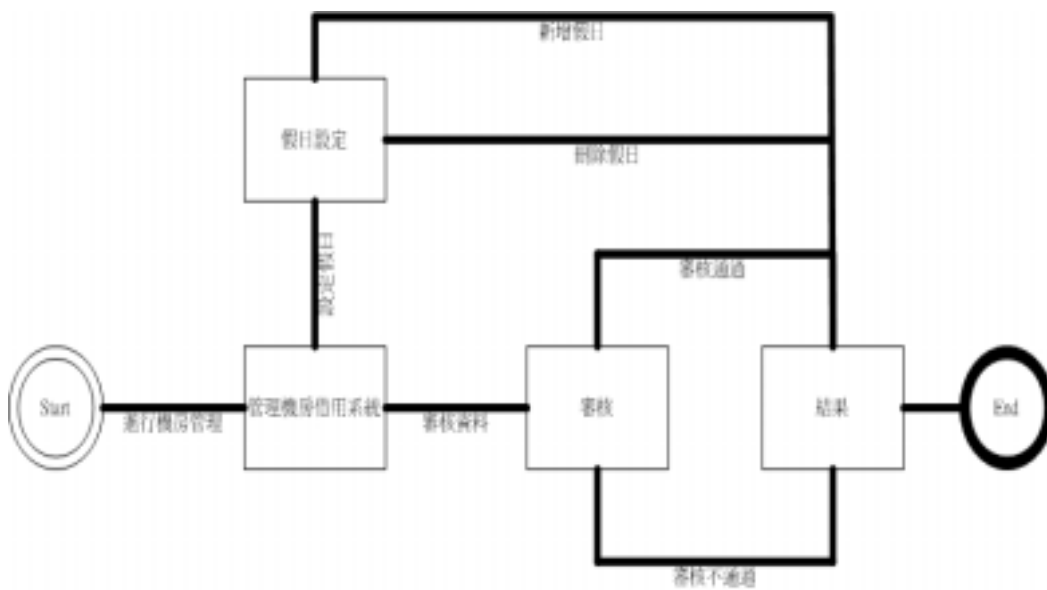


圖 3-5.2 STD 流程圖管理者管理機房部分

## 3-6 Data Element Table

## [tblDetail]

欄位	簡介	資料格式	大小	NULL
SerialNo	借用序號	varchar	14	×
BorrowDate	借用日期	varchar	20	○
BorrowSession	借用節次	varchar	2	○

表 3-6.1 課表紀錄 table

## [tblFixClass]

欄位	簡介	資料格式	大小	NULL
SerialNo	序號	varchar	14	×
BorrowObj	教室代號	varchar	50	○
BorrowUnit	借用單位	varchar	10	○
BorrowDateB	開始日期	varchar	20	○
BorrowDateE	結束日期	varchar	20	○
BorrowSessionB	開始節次	varchar	2	○
BorrowSessionE	結束節次	varchar	2	○
byWho	設定者	varchar	20	○
byIP	IP	varchar	40	○
byTime	時間	datetime	datetime	○
Stus	內容	varchar	40	○

表 3-6.2 固定排課 table

[tblLog]

欄位	簡介	資料格式	大小	NULL
SerialNo	序號	varchar	10	×
DeptNo	單位	varchar	10	○
Role	職稱	varchar	20	○
FullName	設定者名稱	varchar	20	○
Behalf	狀態	varchar	20	○
BehalfTime	設定時間	datetime	datetime	○
IP	IP 紀錄	varchar	20	○

表 3-6.3 LOG 紀錄檔 table

[tblMaster]

欄位	簡介	資料格式	大小	NULL
SerialNo	序號	varchar	10	×
Status	審核狀態	char	1	○
Type	預留用	char	1	○
BorrowObj	借用機房	varchar	10	○
ApplyUnit	借用單位	varchar	10	○
ApplyMan	借用者	varchar	50	○
Tel	電話分機	varchar	15	○
Email	E-mail	varchar	40	○
TeachProf	課程名稱	varchar	20	○
BorrowDate	借用日期	varchar	20	○
BorrowSessionB	借用開始節次	varchar	2	○
BorrowSessionE	借用結束節次	varchar	2	○

	次			
BorrowReason	借用原因	varchar	50	○
ApplyTime	送出時間	datetime	datetime	○
Stus	審核狀態	varchar	10	○
IP	送出者 IP	varchar	40	○
DocNo	其他備註	varchar	20	○

表 3-6.4 教職員申請 table

[tblObj]

欄位	簡介	資料格式	大小	NULL
ObjNo	機房代稱	varchar	10	×
ObjName	機房名稱	varchar	20	○
type	預留	char	1	○

表 3-6.5 電腦機房資訊 table

[tblSession]

欄位	簡介	資料格式	大小	NULL
SessionNo	序號	smallint	NOT	○
SessionName	時段	nvarchar	20	○
SessionMk	時段敘述	nvarchar	40	○

表 3-6.6 上課時段 table

[tblUnit]

欄位	簡介	資料格式	大小	NULL
UnitNo	單位代稱	varchar	20	×
UnitName	單位名稱	nvarchar	40	○
type	預留用	char	1	○

表 3-6.7 教職員所屬單位 table

[tblUsers]

欄位	簡介	資料格式	大小	NULL
loginid	Login 時的帳號	varchar	20	×
Pwd	密碼	varchar	10	○
fullname	管理者全名	nvarchar	10	○
rank	權限	char	1	○
lastlogin	最後一次登入時間	datetime	datetime	○

表 3-6.8 管理者帳號設定 table

[tblfiesta]

欄位	簡介	資料格式	大小	NULL
SerialNo	序號	int	IDENTITY (1, 1)	×
OutDate	假日日期	varchar	10	○
MK	假日名稱	nvarchar	10	○

表 3-6.9 假日設定 table

### 3-7 User Interface Control

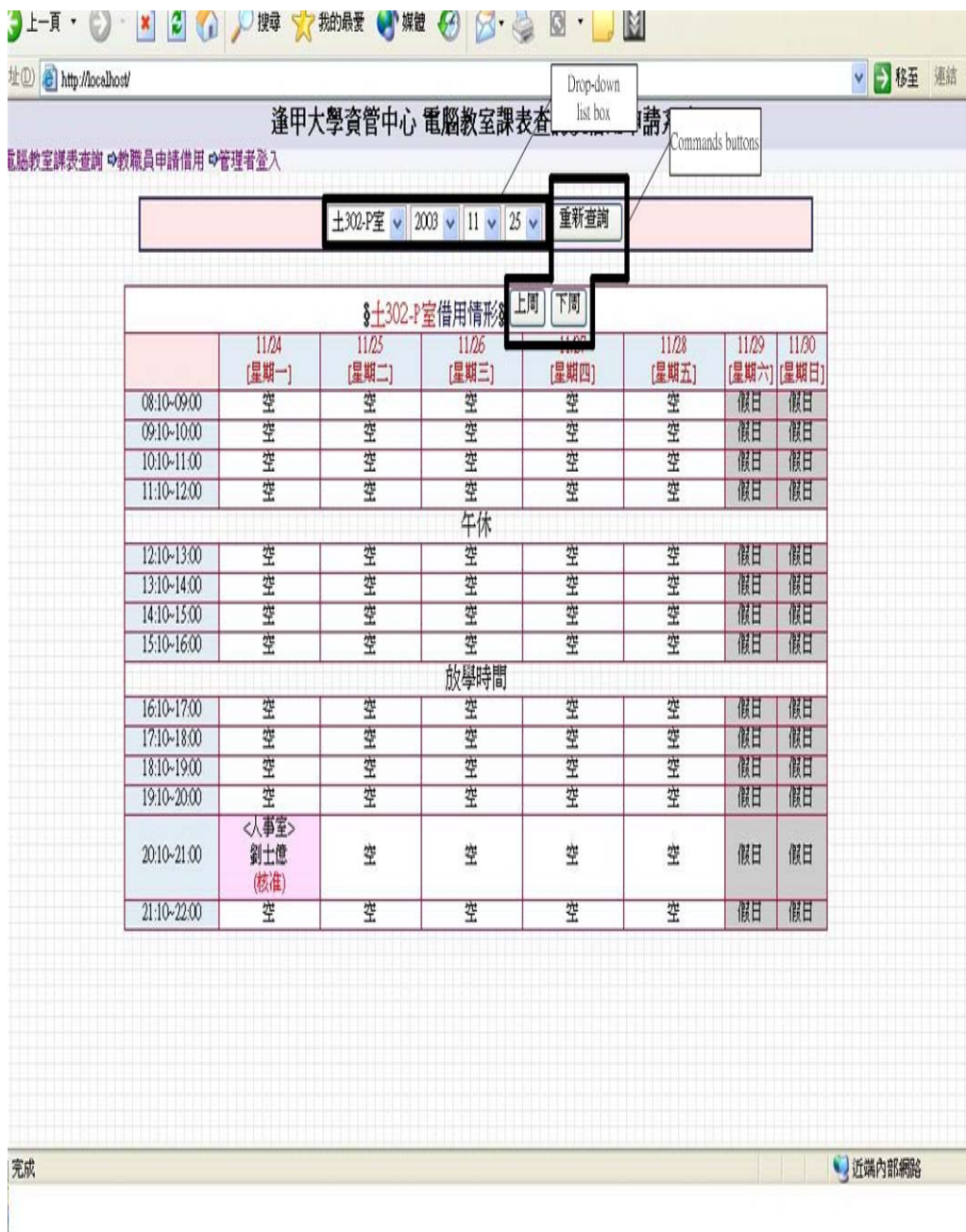


圖 3-7.1 使用者查詢畫面



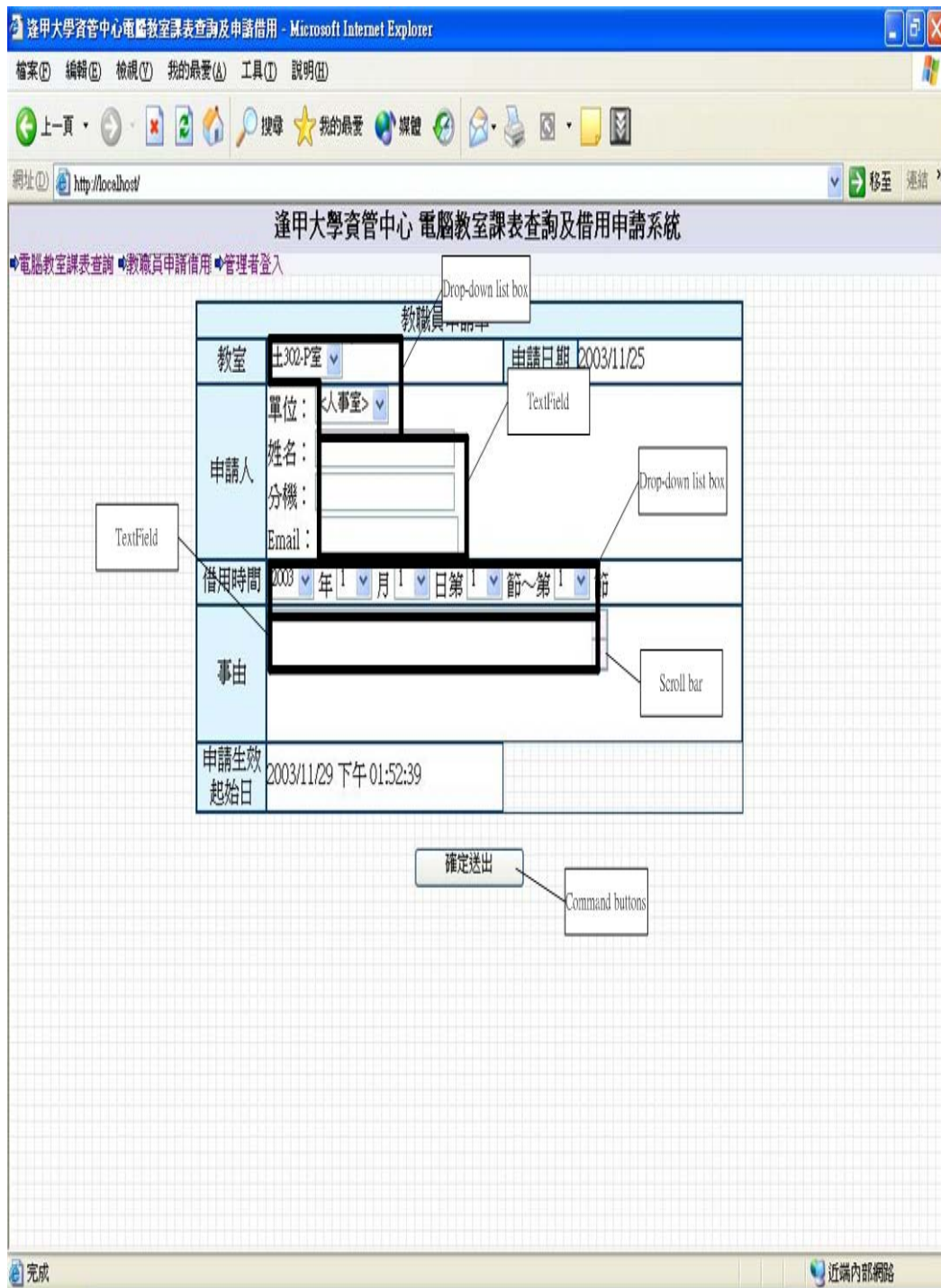


圖 3-7.2 教職員借用畫面

### 3-8 資料庫結構

[tblDetail] 課表紀錄

```
(  
    [SerialNo] [varchar] (14) NULL ,  
    [BorrowDate] [varchar] (20) NULL ,  
    [BorrowSession] [varchar] (2) NULL  
)
```

[tblFixClass] 固定排課

```
(  
    [SerialNo] [varchar] (14) NULL ,  
    [BorrowObj] [varchar] (50) NULL ,  
    [BorrowUnit] [varchar] (10) NULL ,  
    [BorrowDateB] [varchar] (20) NULL ,  
    [BorrowDateE] [varchar] (20) NULL ,  
    [BorrowSessionB] [varchar] (2) NULL ,  
    [BorrowSessionE] [varchar] (2) NULL ,  
    [byWho] [varchar] (20) NULL ,  
    [byIP] [varchar] (40) NULL ,  
    [byTime] [datetime] NULL ,  
    [Stus] [varchar] (40) NULL  
)
```

[tblLog] LOG 紀錄檔

```
(  
    [SerialNo] [varchar] (10) NULL ,  
    [DeptNo] [varchar] (10) NULL ,  
    [Role] [varchar] (20) NULL ,  
    [FullName] [varchar] (20) NULL ,
```

```
[Behalf] [varchar] (20) NULL ,  
[BehalfTime] [datetime] NULL ,  
[IP] [varchar] (20) NULL  
)
```

[tblMaster] 教職員申請

```
(  
[SerialNo] [varchar] (10) NOT NULL ,  
[Status] [char] (1) NULL ,  
[Type] [char] (1) NULL ,  
[BorrowObj] [varchar] (10) NULL ,  
[ApplyUnit] [varchar] (10) NULL ,  
[ApplyMan] [nvarchar] (50) NULL ,  
[Tel] [varchar] (15) NULL ,  
[Email] [varchar] (40) NULL ,  
[TeachProf] [nvarchar] (20) NULL ,//課程名稱  
[BorrowDate] [varchar] (20) NULL ,  
[BorrowSessionB] [varchar] (2) NULL ,  
[BorrowSessionE] [varchar] (2) NULL ,  
[BorrowReason] [nvarchar] (50) NULL ,  
[ApplyTime] [datetime] NULL ,  
[Stus] [varchar] (10) NULL ,  
[IP] [varchar] (40) NULL ,  
[DocNo] [varchar] (20) NULL  
)
```

[tblObj] 電腦機房資訊

```
(  
[ObjNo] [varchar] (10) NULL ,  
[ObjName] [nvarchar] (20) NULL ,
```

```
    [type] [char] (1) NULL
)
[tblSession] 上課時段
(
    [SessionNo] [smallint] NOT NULL ,
    [SessionName] [nvarchar] (20) NULL ,
    [SessionMk] [nvarchar] (40) NULL
)
```

```
[tblUnit] 教職員所屬單位
(
    [UnitNo] [varchar] (20) NOT NULL ,
    [UnitName] [nvarchar] (40) NULL ,
    [type] [char] (1) NULL
)
```

```
[tblUsers] 管理者帳號設定
(
    [loginid] [varchar] (20) NOT NULL ,
    [Pwd] [varchar] (10) NULL ,
    [fullname] [nvarchar] (10) NULL ,
    [rank] [char] (1) NULL ,
    [lastlogin] [datetime] NULL
)
```

```
[tblfiesta] 假日設定
(
    [SerialNo] [int] IDENTITY (1, 1) NOT NULL ,
    [OutDate] [varchar] (10) NULL ,
    [MK] [nvarchar] (10) NULL
)
```

)

## 第四章 成果展示

### ● 教師借用流程

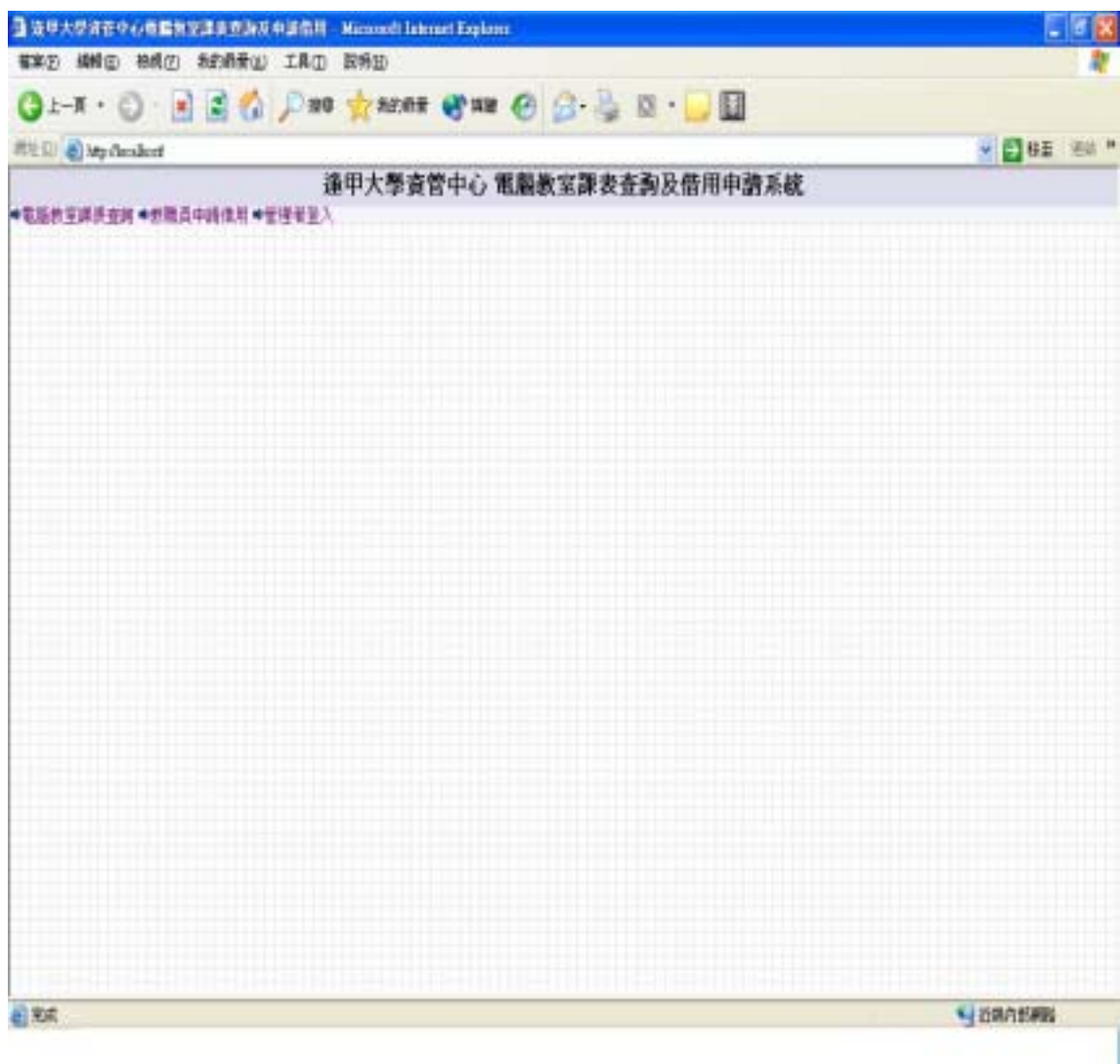


圖 4-1 系統主畫面

- 如果不確定借用時間，哪間機房有空，可以利用本畫面查詢。



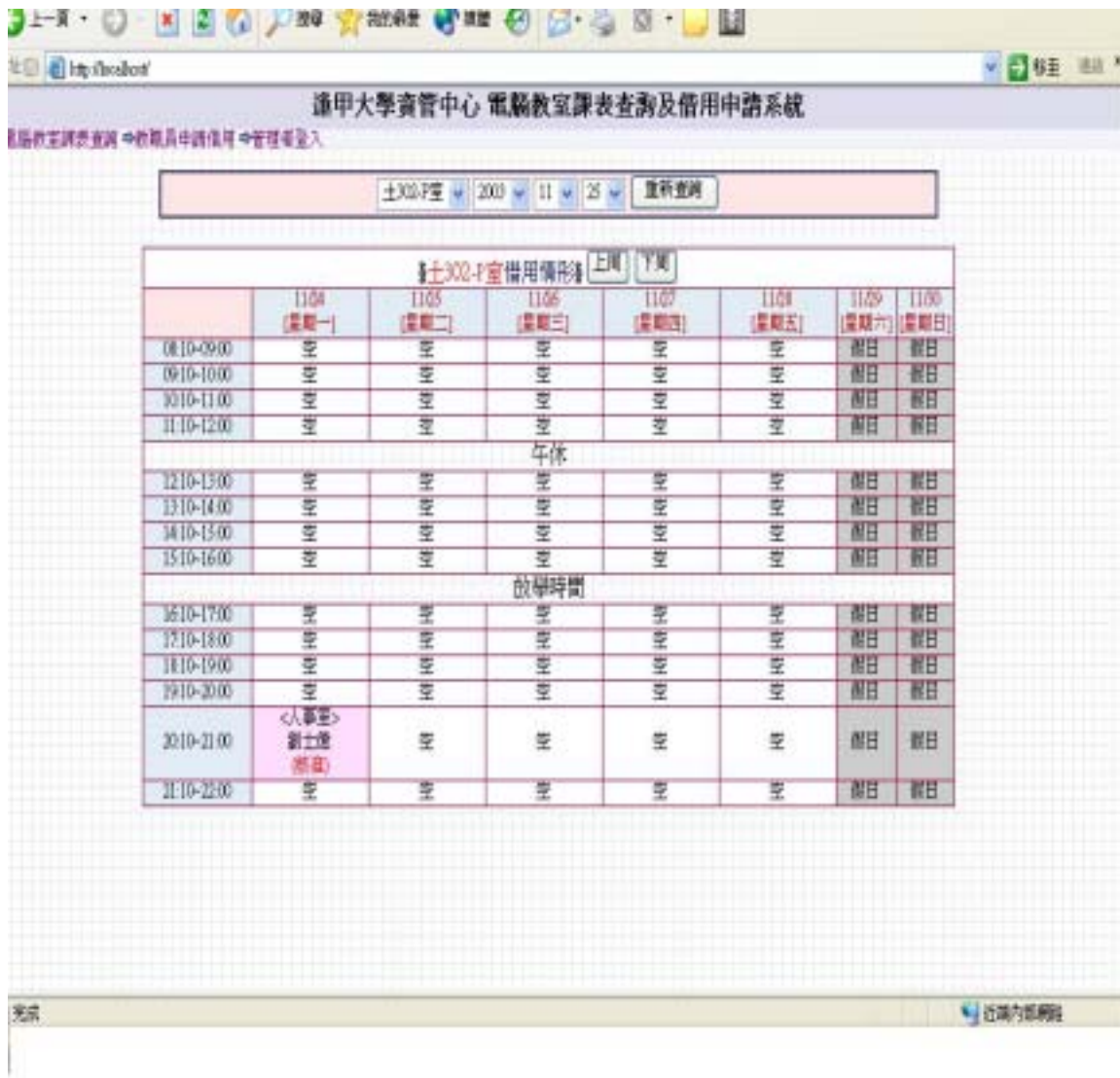


圖 4-2 電腦教室課表查詢畫面

- 提出借用主畫面，依照欄位輸入資訊，最後點選「確定送出」即可提出申請。

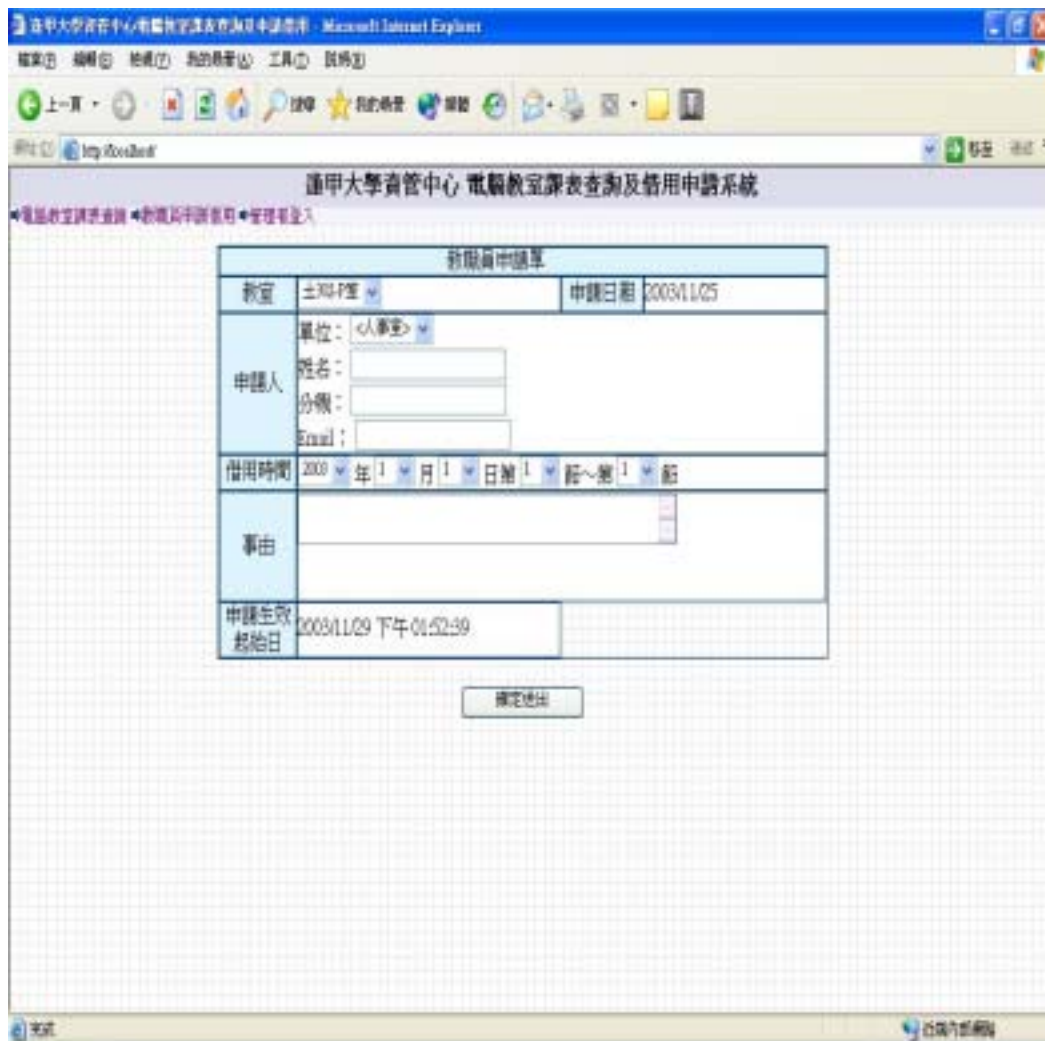


圖 4-3 教職員借用電腦教室畫面

- 此為管理者登入之畫面。



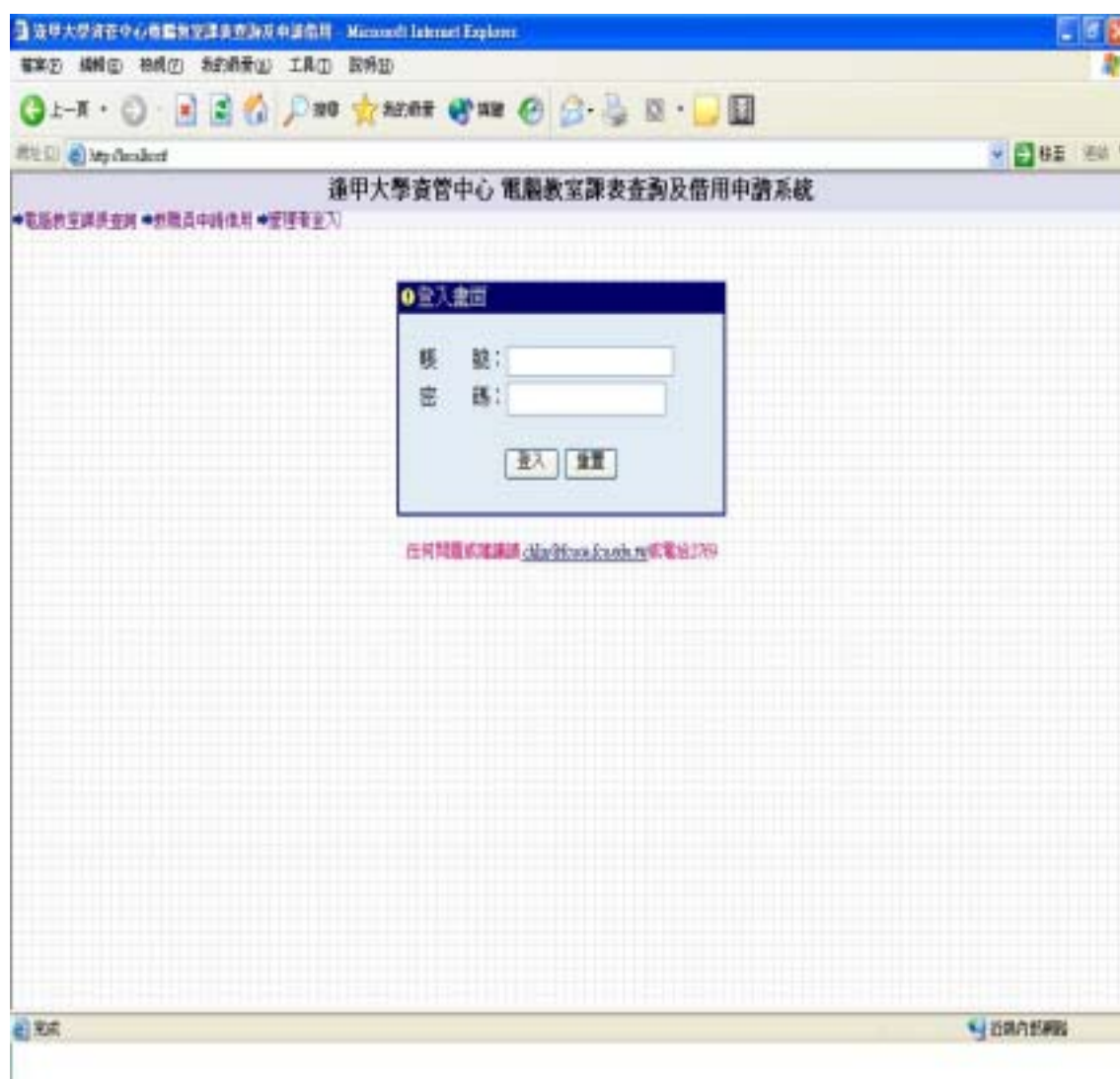


圖 4-4 管理者登入畫面

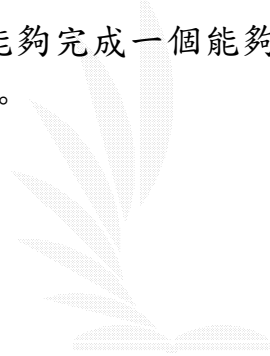
## 第五章 心得

這次的專題，由於只有我一個人做，加上是幫資源管理中心寫給學校的程式，所以我盡量挑學校現有的系統來實做。加上因為時間緊迫，很多方面督導那邊給我的答案也不明確，所以有部分的介面可能會因為資源管理中心方面的要求而在加以改變。

在資料庫方面，資源管理中心的黃東炳先生要求我盡量做簡單，加上資料量非常小，所以我沒有用到關連性資料庫，只是純粹用獨立的 table 完成。

剛開始因為我從未接觸過 ASP 語言，花了很多時間在摸索階段，所以程式到最後幾乎是每晚趕工，幸虧黃東炳先生提供了相當多的技術支援，並且借我不少相關資料，顧老師也給我相當多在畫流程圖方面的建議，才能勉強完成本專題。

我的最終目標是，能夠完成一個能夠滿足學校資源管理中心督導們要求的機房借用系統。



## 附錄

### SQL Server 特性

特性	說明
<p>和微軟 windows 家族緊密結合</p>	<p>SQL Server 可以充分和 Windows 及 Windows 環境的應用程式緊密結合，降低成本及安裝的複雜度，目前已有許多的協力廠商開發的 Windows 應用程式及工具能夠支援 SQL Server</p> <p>管理人員可以從單一的場所，透過 Windows 95 及 Windows NT 的作業系統上管理網站上不同的 SQL Server 伺服器，而 Cursor 的使用允許開發者輕鬆的建立內容豐富，具有互動能力的應用程式，SQL Server 的連結環境允許 Windows 應用程式存取大型電腦上的 DB2. Oracle. Ingres 等其他資料庫的資料，透過 SQL Server 開放式的介面，開發者可以利用各種 Windows 的開發工具像是 Visual Basic Visual C++ 來進行開發的工作。</p>
<p>和 Microsoft Back office 充分的結合</p>	<p>Microsoft SQL Server 是 BackOffice 家族的成員之一，整個家族還包括了 Microsoft Exchange、Microsoft Internet Information Server、Microsoft Systems Management Server、Microsoft SNA Server，以及 Windows NT Server。SQL Server 和 BackOffice 的其他成員充分整合，提供架構應用程式完整的伺服器環境。</p> <p>SQL Server 同時也善用 Windows NT Server 的相關服務，像是保全環境、網路登入、事件</p>

	<p>登錄以及效能監控等等。它同時也和一些低階的服務配合，包括多執行緒、記憶體管理和非同步輸出入處理。SQL Server 可以搭配 Microsoft Exchange 或是其他符合 MAPI 規格的電子訊息系統傳送及接送電子郵件。而 Systems Management Server 則使用 SQL Server 做為儲存管理資料的場所。而 Internet Information Server 可以和 SQL Server 結合提供 Web 站台存取資料庫的能力。至於 SNA Server 則提供 SQL Server 和大型主機連結的能力。</p>
<p>網際網路整合</p>	<p>SQL Server 是和 Web 站台相結合最理想的資料庫產品。由於它能夠和 Internet Information Server 緊密配合，使用者可以透過瀏覽程式取得 SQL Server 所提供的資料。而 SQL Server 在 ODBC 上使用的驅動程式也可以和 Internet Database Connector 及 Active Server Page 等 Web 資料庫存取技術結合，提供高效能的資料處理環境。另外 SQL Server 所提供的 Web Assistant 可以讓系統自動更新 HTML 頁面，定時提供使用者正確的資料。</p>
<p>大量的開發工具、應用程式及傳輸開道</p>	<p>SQL Server 可以搭配許多市場上重要的主從式方案開發工具，另外在微軟公司的網站上也提供了可搭配的應用程式的相關資料供客戶查詢。</p>
<p>內建資料複製功能</p>	<p>SQL Server 內建的資料複製功能可以輕易的將資料就近提供給使用者。而整個設定的動作經由 SQL Server 圖形化管理工具的協助，將變得更為簡單。</p> <p>SQL Server 提供了異質複製的能力，可以出版資料給 Oracle、IBM DB2、Sybase、Microsoft</p>

	<p>Access 及其他 ODBC 資料庫。另一個新功能則是提供 TEXT 及 IMAGE 資料類型的複製能力。</p>
<p>主從式架構</p>	<p>Microsoft SQL Server 能夠同時提供傳統大型主機環境所具備的集中保密環境、資料整合性，同時又具有今日個人電腦簡易使用、多樣化使用者介面以及大量配合工具等優點。</p> <p>它能夠結合不同形式的前端環境提供共用的資料，因此你可以自由選取適合的工具來進行工作。</p> <p>SQL Server 可以有效的運用網路環境，因於所有的資料庫查詢動作是在伺服器上集中進行，所有網路的流量可以有效降低。</p>
<p>功能完整的管理工具</p>	<p>SQL Enterprise Manager 是一個圖形化的管理工具，您可以透過它在單一的場所進行所有 SQL Server 的管理工作。</p> <p>SQL Enterprise Manager 簡化了複雜的管理工作，這些工作包括儲存空間的安排、權限的管理等等。透過這個 Windows 環境下的管理介面，所有的物件都可以經由視覺化的方式進行操作，包括資料表的定義、stored procedure 及 trigger 的設計等等。其他圖形化的工具還包括了保全環境的管理工具、磁帶備份工具及前端工作站的設定工具。</p> <p>SQL Server 的管理架構在以開放的物件架構為基礎，透過這個稱做 SQL Distributed Management Objects (SQL-DMO) 的環境，我們可以輕易使用支援 COM 規格的工具進行開發工作，並提供管理的能力。</p> <p>SQL Server 同時也內建自動維護資料庫的功</p>

	<p>能，您可以透過 SQL Server Database Maintenance Wizard 的管理精靈，只需輸入貴公司資料量成長等相關資料，便可自動產生維護計劃，並可設定系統排程定期自動執行，進行維護任務，讓管理者可以將例行性的維護工作自動化。</p>
Open Data Services (ODS)	<p>透過 ODS，我們可以整合 SQL Server 及其他各式各樣的應用程式及資料來源。Microsoft Open Data Services 是微軟公司多執行緒架構的開發程式庫，透過它可以開發出頗具威力的開道程式，存取自訂的資料來源。它同時也提供事件及警訊的處理功能，以利重要工作的處置動作。</p>
網路獨立性	<p>某些資料庫伺服器必須搭配專屬的作業系統或是僅支援專屬的網路協定。而 SQL Server 則不受限於特定網路環境。</p> <p>由於 SQL Server 完全支援開放業界標準，它可以在多數受歡迎的網路環境上使用，包括了 IBM LAN Server、Microsoft LAN Manager、Novell NetWare、Banyan VINES、DEC? PATHWORKS、AppleTalk 以及 Windows NT Server 等網路。</p>
加強的 SQL 語言	<p>SQL Server 所採用 Transact-SQL 是一個經過改良的資料庫語言，可以和業界的標準資料存取語言相容。Transact-SQL 包括了重要的延伸功能—例如 stored procedure 及 trigger，以支援企業內任務導向的應用。</p>
支援 scrollable cursor	<p>SQL Server 所支援的 cursor 簡化了資料瀏覽的設計工作，設計者可以充分運用前後捲動的功能來檢視資料，並對特定的記錄進行更新、刪</p>



	<p>除等動作。這對 Windows 系統上的應用程式幫助最大，因為在使用者介面的設計上，cursor 能提供無比的彈性，並且強化資料的瀏覽能力。</p>
<p>支援業界標準</p>	<p>SQL Server 支援 ANSI SQL-92、FIPS 127-2、SNMP 以及 XA 等業界標準。此外也支援一些開放的程式介面，像是 ODBC、MAPI 以及 COM。</p>
<p>交易處理</p>	<p>一致性和資料復原能力可以確保您的資料庫在多人使用的環境下，即使是在進行複雜更新動作當時碰到動作失敗的情況，依舊有機會還原資料，避免遺失。</p> <p>SQL Server 內部在處理資料時會把每一筆交易視做單一的工作單位。所謂的交易是指一個確定完成的動作，並且會在資料庫中做出反應；或者說是被取消，而對資料庫沒有任何影響，就像不曾發生過任何動作一樣。</p> <p>交易處理能力對於任務導向的應用程式而言是一個不可獲缺的功能，這些應用的範圍可能包括會計、資產管理和線上訂購處理等等。透過兩階段確認的協定方式 (2-phase commit protocol*)，SQL Server 可以支援數台伺服器的同步交易，並且確保網路上的所有伺服器資料是維持在同樣的狀況下。* 兩階段確認的協定 (2-phase commit protocol)：是定義嚴謹的資料庫系統所擁有的必要功能之一。它可以確保在處理兩個表格或分散在各地不同資料庫中的資料時，不會因交易失敗，而導致資料不一致的情況發生。所以，在交易進行的過程中，因某一機器突然當機或網路斷線，SQL Server 可以使得交易中所有相關資料回覆到交易前的狀況。</p>

<p>內建平行處理 控制</p>	<p>SQL Server 的交易處理在設計上另一項優點是平行存取的控制。SQL Server 透過動態鎖定的架構讓同時存取的使用者在查詢和更新時不會互相干擾。預設值是採取資料頁鎖定 (page-level locking) 的方式。</p> <p>所有 SQL Server 的鎖定動作都是隱含進行，程式設計人員不需要擔心鎖定處理的動作。由於所有鎖定的資訊都是存放在資料表當中，有處理鎖定的動作相當快。而 SQL Server 也支援多重鎖定，系統會自行選擇較不嚴格的鎖定方式以利資料的操作處理。</p>
<p>伺服器強制處理資料整合性</p>	<p>配合上伺服器強制處理資料整合性的能力，您可以在資料庫中安排複雜的商業原則以及資料整合的要求，透過集中管制的環境來要求所有應用程式遵循。</p> <p>SQL Server 提供 ANSI 標準所制定的參照整合要求及欄位限制設定，同時也提供許多先進的資料整合特性，像是 rule、default、stored procedure、trigger 以及使用者定義資料型態，來強迫達成資料的整合性。</p>
<p>Stored Procedure</p>	<p>SQL Server 將事先編譯好的 SQL 命令(stored procedure)存放在資料庫中，在某些情況下，可以減少查詢的處理時間達 80%。</p> <p>Stored procedure 也可以用來做為集中管理程式邏輯的環境，讓您在伺服器上制定執行步驟，以進行複雜的查詢作業以及更新動作。由於 stored procedure 可以接受參數的傳遞，所以執行方式充滿彈性。Stored procedure 可以同時由多個程式共用，簡化開發的過程。</p> <p>SQL Server 同時也支援擴充的 stored</p>



	<p>procedure 設計能力，透過 C 語言設計的程式館來提供更大的彈性，甚至引進 OLE Automation stored procedure，讓您的 SQL Server 跨進 OLE 的程式設計環境。</p>
<p>Trigger</p>	<p>Trigger 可以避免任何未經授權或是不一致的資料修改動作。Trigger 事實上是一種特殊形式的 stored procedure，可以在任何資料修改動作發生時由 SQL Server 自動執行。</p> <p>Trigger 的威力驚人。它們允許您在集中的環境下強制要求資料的整合性，並遵循商業處理原則，而不需要在個別的程式中安排這樣的設計。例如：</p> <p>當某個客戶仍有貨款未清時，不允許客戶以賒欠的方式訂貨。</p> <p>每當某個產品的零件成本更動時，自動更新該產品的成本。</p> <p>不允許增加的信用額度超過 25%。</p> <p>Trigger 大幅簡化了在 SQL Server 上要求資料正確性的設計方式，提供威力強大的控制能力。</p>
<p>規則制定及預設值處理</p>	<p>Rule 是另一個 SQL Server 用來集中強制維護資料整合性的工具，而且可以針對欄位進行處理。Rule 可以自動要求某個值局限於特定的範圍內，像是特定的字串型式、某個數字範圍或是資料清單當中。例如，我們可以要求一個資深員工的服務年資必須是在 35 到 70 年之間。Default 則允許您安排欄位的標準值，當使用者未提供時就直接採用這個預設值。例如，當使</p>

	<p>用者未輸入購買數量時，自動指定 1 到數量欄位中。</p> <p>Rule 以及 default 可以安排在特定欄位、某些欄位、或是所有和某個使用者資料型態結合的欄位。</p>
<p>使用者定義資料類型</p>	<p>SQL Server 可以讓您建立自己的資料型態，以補強 SQL Server 所提供的資料型態。</p> <p>使用者自訂型態的好處在於您可以結合 rule 或是 default 到某個使用者型態上，而這個型態可以安排在不同的資料表欄位上。因此我們只要定義一次連結的關係，就可以應用在不同的環境。</p>
<p>多執行緒架構</p>	<p>Microsoft SQL Server 提供了高效率的資料處理環境，充分展現主從式資料庫的威力。</p> <p>傳統的資料庫管理系統是以兩種方式處理同時存取的使用者：一是為每個使用者產生獨立的程序(這樣會增加系統的負擔及記憶體的使用量)；或者是利用單一的程序服務所有的使用者，以循序的方式處理所有的查詢，而不是以平行的方式處理。</p> <p>SQL Server 則充分和 Windows NT 的環境結合，並且採用了最具效率的處理方式，在單一的程序中利用不同的執行緒處理所有的查詢，一來可以透過執行緒平行處理不同的查詢，同時又在一個程序中共用記憶體，降低系統資源的耗用。</p> <p>這樣的架構同時也減少了記憶體的使用量，每一個額外的使用者只耗掉 SQL Server 約 40K 的記憶體，而不像其他的資料庫系統可能要消耗 400K 或是更多的記憶體(如 Oracle)。</p>

	<p>SQL Server 多執行緒的設計最主要的優勢在於傳輸量，因為 SQL Server 不會因為使用者的增加而減緩處理的速度。</p>
<p>高可用性 (動態備份及資料恢復)</p>	<p>SQL Server 在處理例行性的維護工作時，不會造成停頓的情形，增加維護的成本。因為網路資源的停滯，勢必會構成企業運作的成本浪費。SQL Server 動態備份的能力可以讓您在備份資料庫的同時，依舊提供資料存取的能力，讓使用者依舊可以更新或讀取資料庫的內容，以應付任務導向程式的要求。</p> <p>一旦系統失效(作業系統當機、電源中斷等等,) SQL Server 的自動回復機能會在系統回復後自動恢復所有的資料，並確認資料的一致性。您的應用程式可以馬上恢復進行處理。</p> <p>SQL Server 高可用性的設計能讓您在系統忙線時繼續進行資料庫設計的變更，或是進行診斷的動作。</p> <p>SQL Server 中新引進的功能還包括個別資料表的備份及復原、簡易的災難復原能力以及特定時間點回復的能力。</p>
<p>容錯能力</p>	<p>SQL Server 提供了儲存區域鏡射(mirror)的服務，可以提供周詳的容錯能力，以吻合任務導向程式的需求。SQL Server 的資料庫設備可以快速的進行鏡射、解除鏡射或是重新鏡射的動作，並且提供不停頓的運作能力。我們也可以單純針對 transaction log 進行鏡射，以提供更大的彈性。</p> <p>Microsoft SQL Server 同時也充分支援 Windows NT Server RAID 5 或是硬體的鏡射功能，以提供容錯的服務。</p>

<p>高度保全能力</p>	<p>SQL Server 提供了最嚴密的方式，來保障高機密性的資料。SQL Server 能提供依使用者所設定的權限，並且可以按照資料表、view、stored procedure 以及 SQL 命令決定權限。它同時也支援欄位的保全能力。另外權限的設定也可以針對使用者群組進行，因此保全設定的設定及管理工作會很容易進行。由於保全是 SQL Server 集中管理，您不需要在您的前端程式中考慮使用者認證方面的設計，減輕開發的負擔。SQL Sever 同時也 and Windows NT 作業系統的保全環境相結合，讓使用者只需登入一次，就可以取用網路及資料庫的資源。同時在帳戶的管理上還提供密碼期限和帳戶失效的控制能力，同時還能透過最新的加密服務，對經由公眾網路進行存取的資料加密，以達到最高的保全要求。</p>
<p>開放的前端介面</p>	<p>SQL Server 提供了開放的程式設計介面，同時又能結合 DB-Library 及 ODBC 等標準環境，使得任何一個軟體開發人員都可以建立搭配 SQL Server 的應用程式。市場上已經有數以百計的應用程式、工具及重要商業解決方案都支援 SQL Server。Microsoft 也承諾持續的加強這些介面，以保障客戶的投資。</p>
<p>支援多媒體資料儲存</p>	<p>SQL Server 支援大型的資料型態，例如 TEXT 及 IMAGE，每一個欄位的限制是 2GB。因此它很適合用於文件及影像管理的系統，這些欄位也很適合管理 Web 站台的內容。</p>

<p>分散式資料管理</p>	<p>SQL Server 的分散式資料管理環境允許工作站在同一時間內存取數個 SQL Servers 上的資料，進而允許開發人員設計分散式應用程式。SQL Server 允許您透過單一的分散式交易處理，對數個不同的伺服器資料進行更新，而依舊能保持資料的整合性及一致性。如果該工作站或是任何一台伺服器有失效的情況，所有伺服器上所進行的相關交易都會自動取消，回復到交易進行前的狀況。</p> <p>SQL Server 針對上述狀況的處理方式相當聰明，而不需要應用程式處理細部的程序。它是透過 Distributed Transaction Coordinator (DTC) 這個元件進行處理的動作，DTC 目前支援市場上流行的程式介面，包括 ODBC、DB-Library、Transact-SQL、XA 以及 OLE transactions。</p>
----------------	--

## 參考資料

編號，作者，名稱，書(期刊)名，期別，書局名稱，出版日期。

[1]Shelly/Cashman/Rosenblatt System Analysis and  
Design ,Course Technology ,Fourth Edition

[2]東海林 誠/窪田寬之/坂本篤/橋本大輔 學UML的第一本書,博  
碩文化,2002/4

[3]陳峰琪 ,ASP & SQL 網頁資料庫程式設計 ,知城數位科技股份  
有限公司,2001/8

