

致理技術學院

資訊網路技術系 實務專題報告

致理技術學院- 電子化畢業資格審查系統

指導老師：劉爵至 老師

學生：張均瑋 69434106

林姿儀 69434115

黃經樺 69434144

中華民國 97 年 11 月

致理技術學院

資訊網路技術系 實務專題報告

致理技術學院- 電子化畢業資格審查系統

指導老師：劉爵至 老師

學生：張均瑋 69434106

林姿儀 69434115

黃經樺 69434144

本成果報告書經審查及口試合格特此證明。

指導教師：_____

中華民國 97 年 11 月

專題研究授權書

本授權書所授權之專題研究為致理技術學院-電子化畢業資格審查系統

共3人，在致理技術學院資訊網路技術系 97 學年度第1 學期完成資網實務專題。

專題名稱：致理技術學院-電子化畢業資格審查系統

同意 不同意

本組同學共3人，皆同意著作財產權之論文全文資料，授予教育部指定送繳之圖書館及本人畢業學校圖書館，為學術研究之目的以各種方法重製，或為上述目的再授權他人以各種方法重製，不限地域與時間，惟每人以一份為限。

上述授權內容均無須訂立讓與及授權契約書。依本授權之發行權為非專屬性發行權利。依本授權所為之收錄、重製、發行及學術研發利用均為無償。上述同意與不同意之欄位若未勾選，該組同學皆同意視同授權。

指導教師姓名：

專題學生簽名：

學號：

中華民國 97 年 11 月 28 日

誌 謝

轉眼之間，在致理的時光就要邁入最後的里程了。當初選校時毫不猶豫的選擇了致理，時至如今果然是滿載而歸。完成這個專題的製作，要感謝的人非常的多，感謝系上每位細心教學的老師們，感謝同組一起奮鬥的組員們，感謝有緣一起度過這四年時光的同學們。

謝謝林正平老師，林老師帶領了我們進入程式的世界；謝謝王勝石老師，王老師除了課堂上的指導之外，也利用閒暇之餘指導我們，怎樣才算是一個好報告，讓我們獲益良多；謝謝陳文雄老師，教導我們如何學習使用有架構的系統；謝謝高楊達主任的教導我們學習資料庫系統，以及網頁製作、資料電子化的相關基礎知識，並跟我們一起度過兩年的快樂導生相處時光。

最後特別感謝我們的指導老師劉爵至老師，在整個畢業專題寫作期間始終如一的指導。打從劉老師不嫌棄地點頭願意擔任指導老師開始，擬訂專題題目、建立系統架構、網頁技術指導、乃至最後專題的格式編排與寫作等等，老師雖然總是抱著頭口中總是說著：「好煩好煩我要爆炸了」，但仍然付出無比的耐心對我們專題細心的加以審閱與指導。老師不但利用暑假以及私人時間為我們上課，讓我們從一知半解，進而能完成這個系統，言語已不足讓我們表達對老師的感謝了！還有一起奮鬥至今的組員，這一年來專題的製作過程雖然有風有雨，也度過無數個熬夜的夜晚，但是我們完成了這份專題，這份學習克服難關的精神與經驗，將成為我們四年來在

致理最棒的句點。

摘 要

畢業資格審查作業，為每所大專院校或技職體系學校必須處理的例行性業務，人工的審核作業通常耗時、繁瑣又吃重，常造成業務人員不少的困擾。隨著資訊科技與網際網路的進步，行政流程電子化的概念也漸漸受到大家的重視，電子化是運用資訊通訊科技及網際網路的技術來提高行政效率，達到省時、便利、節省支出與即時互動的效果。倘若能結合電子化的概念，來協助畢業資格審查作業，必定能提高學校行政效率。

本研究最主要的目的是希望將繁瑣的畢業資格審查作業，透過電子化的方式，進行線上審核作業，以簡化整個作業流程，有效地縮短審核時間，並提高審核資料的正確性，以增進行政作業效率與服務品質。本研究是以「致理技術學院」為主要研究範圍，利用 Microsoft .NET 的技術，結合 SQL Server 2005 資料庫，建置出「電子化畢業資格審查系統」。

「電子化畢業資格審查系統」將原本人工運作的審查流程加以簡化，整合課務組、註冊組、各系所(科)的資料彙整與回報，並提供學生線上查詢審核結果，不但可以減少人力資源，在未來也可與其他校務行政系統結合，縮短整個行政作業時間，讓校園資源可作更有效的分配與利用，進而實現電子化的便利性與互動性。

關鍵字：電子化、畢業資格審查、ASP.NET、Visual Studio 2005、SQL Server 2005

目 錄

專題研究授權書	III
誌謝	IV
摘要	VI
目錄	VII
圖目錄	X
表目錄	XV
第一章 緒論	1
第一節 研究背景	1
第二節 研究動機與目的	5
第三節 研究範圍與報告書架構	7
第二章 相關研究與文獻探討	10
第一節 .NET技術概論	10
(一) Microsoft.NET技術	11
(二) ASP.NET技術	15
第二節 資料庫系統	24
(一) 資料庫系統簡介	24

(二) SQL Server 2005	32
(三) 小結	39
第三節 VISUAL STUDIO 2005	41
(一) Visual Studio 2005 功能	41
(二) Visual Studio 2005 系統環境介紹	42
(三) 小結	47
第三章 電子化畢業資格審查系統	51
第一節 系統架構	51
第二節 系統功能簡介	57
(一) 註冊組	57
(二) 課務組	58
(三) 系所(科)	58
(四) 學生	59
第三節 系統操作流程	60
(一) 安裝 Visual Studio 2005 套件	60
(二) 安裝 SQL Server 2005	68
第四節 系統特色	84
第四章 系統呈現	85
第一節 預期效能	85
(一) 註冊組	86

(二) 課務組	86
(三) 系所(科)	87
(四) 學生	87
第二節 實際效能	88
(一) 註冊組	88
(二) 課務組	89
(三) 系所(科)	90
(四) 學生	91
第三節 系統畫面	92
第五章 結論	116
第一節 結論	116
第二節 未來展望	117
參考文獻	119

圖 目 錄

圖 1-1、畢業資格審查流程圖.....	2
圖 1-2、專題研究流程圖.....	8
圖 2-1、.NET的基本架構.....	10
圖 2-2、.NET架構與應用程式、作業系統之間的關係.....	13
圖 2-3、.NET 架構圖.....	14
圖 2-4、ASP.NET 的執行流程.....	19
圖 2-5、ASP.NET安全性架構圖.....	22
圖 2-6、簡單的資料庫系統環境.....	26
圖 2-7、大型主機/終端機架構.....	29
圖 2-8、主從式架構.....	30
圖 2-9、分散式架構.....	31
圖 2-10、SQL 主從架構.....	33
圖 2-11、SQL SERVER 2005 的架構.....	34
圖 2-12、VISUAL STUDIO TOOLS FOR OFFICE的環境.....	42
圖 2-13、工具列.....	43
圖 2-14、工作清單.....	44
圖 2-15、類別物件瀏覽器.....	45
圖 2-16：遠端伺服器呼叫.....	46

圖 3-1、電子化畢業資格審查系統架構圖.....	51
圖 3-2、畢業資格審查系統流程圖.....	54
圖 3-3、安裝VISUAL STUDIO 2005 畫面	60
圖 3-4、VISUAL STUDIO 2005 安裝程式載入安裝元件.....	61
圖 3-5、完成VISUAL STUDIO 2005 安裝程式載入安裝元件.....	61
圖 3-6、使用者同意書以及輸入序號、使用者名稱	62
圖 3-7、選擇安裝的功能選項.....	62
圖 3-8、選擇安裝功能的自訂選項，安裝所需要的程式內容	63
圖 3-9、所需選項打勾.....	63
圖 3-10、安裝程式開始安裝.....	64
圖 3-11、安裝DOCUMENT EXPLORER 2005	64
圖 3-12、安裝VISUAL STUDIO 2005	65
圖 3-14、安裝MICROSOFT VISUAL J#可轉發套件	66
圖 3-15、安裝SQL SERVER 2005 MOBILE EDITION	66
圖 3-16、安裝MICROSOFT DEVICE EMULATOR 1.0 版	67
圖 3-17、VISUAL STUDIO安裝完成	67
圖 3-18、打開SQL SEVRVER 2005 組態管理員	68
圖 3-19、選擇SQL SEVRVER服務	69
圖 3-20、啟用NAMES PIPES服務	69
圖 3-21、系統提示.....	70

圖 3-22、啟用TCP/IP服務.....	70
圖 3-23、電腦服務更動提示.....	70
圖 3-24、重新啟動服務.....	71
圖 3-25、開始安裝SQL SERVER 2005	71
圖 3-26、SQL SERVER 2005 授權合約	72
圖 3-27、輸入註冊資訊.....	72
圖 3-28、安裝功能選擇.....	73
圖 3-29、安裝準備動作完成，開始安裝.....	73
圖 3-30、安裝程序中.....	74
圖 3-31、安裝完成.....	74
圖 3-32、啟動SQL SERVER 2005	75
圖 3-33、選擇連接SQL SERVER 2005 伺服器名稱.....	75
圖 3-34、開啟資料庫.....	76
圖 3-35、點右鍵開啟資料庫內容屬性.....	76
圖 3-36、進入伺服器屬性頁面，設定安全性	77
圖 3-37、伺服器驗證選擇SERVER 以及WINDOWS。	78
圖 3-38、設定完成後，必須重新啟動SQL SERVER才能生效。	78
圖 3-39、點選SQL SERVER 2005 組態工具後進入SQL SERVER 組態管理 員	79
圖 3-40、按下重新啟動服務.....	79

圖 3-41、系統提示停止服務.....	80
圖 3-42、系統提示重新啟動服務.....	80
圖 4-1、登入畫面.....	92
圖 4-2、註冊組登入.....	93
圖 4-3、課務組登入.....	93
圖 4-4、系所(科)登入.....	94
圖 4-5、學生登入.....	94
圖 4-6、建立系所(科)畢業門檻.....	95
圖 4-7、輸入系所(科)畢業門檻內容.....	96
圖 4-8、建立各學制修業年限資料.....	97
圖 4-9、建立應修科目表.....	98
圖 4-10、初修改之應修科目表.....	99
圖 4-11、選擇加入應修科目表之課程.....	100
圖 4-12、選擇修課項目代碼.....	101
圖 4-13、應修科目表選填完畢.....	102
圖 4-14、選擇未審核應修科目表.....	103
圖 4-15、課務組審核應修科目表(上方畫面).....	104
圖 4-16、課務組審核應修科目表(上方畫面).....	104
圖 4-17、學期審核.....	105
圖 4-18、學期審核與畢業班審核.....	106

圖 4-19、學期審核開始.....	107
圖 4-20、畢業班審核開始.....	108
圖 4-21、學生查詢成績（上方畫面）.....	110
圖 4-22、學生查詢成績（下方畫面）.....	111
圖 4-23、未通過審核之名單.....	112
圖 4-24、查詢應修科目表.....	113
圖 4-25、學號查詢.....	114
圖 4-26、自訂查詢.....	115

表 目 錄

表 2-1、各種開發系統功能比較表.....	47
表 3-1、使用者功能列表.....	53
表 4-1、管理者介面系統成效列表（註冊組）	88
表 4-2、管理者介面系統成效列表（課務組）	89
表 4-3、管理者介面系統成效列表（系所（科））	90
表 4-4、管理者介面系統成效列表.....	91

第一章 緒論

第一節 研究背景

近年來臺灣經濟實力與成長傲視全球，資訊產值獨占鰲頭，無論工商業、行政、國防、醫療以及教育等，已全面性的使用電腦處理資料，資訊的提供與運用，已成為行政效率的代名詞，顯示我國已全面邁向資訊時代。值此階段，傳統的學校行政作業方式亦將有所變革，「校務行政電腦化」是必經之路，藉由校務行政電腦化之推展，以提高行政效率，並帶動資訊科技水準之提昇。

學校校務行政工作的內容是週而復始、重覆性高，且變化性不大，但由於資料量大，如採人工作業方式，曠時廢事且易生錯誤，而改採電腦化作業，將簡化一些處理過程，有效管理龐大的資料，更能即時提供老師及管理者所需之資訊，提高學校教與學的服務品質。另一方面，學校組織運作均著重整體性，而非各處室各自為政，校務行政電腦化後，將有助於各類資訊的迅速獲取及整體的運作。

目前致理技術學院在校務行政系統上，主要包括「教務行政系統」以及「學務行政系統」兩大主要系統，其中「教務行政系統」所包含的項目有新生入學管理系統、註冊繳費、學籍資料、成績管理、課程資料、開課選課以及教學評量等；而「學務行政系統」所包含的項目有考勤資料、獎懲資料、學生輔導、軍訓資料等。在目前學校各種電腦化的系統中，發現

在畢業資格審查方面，目前尚未完備，且採取的是人工畢業資格審查的方式，下圖 1-1 為致理技術學院畢業資格審查的流程。

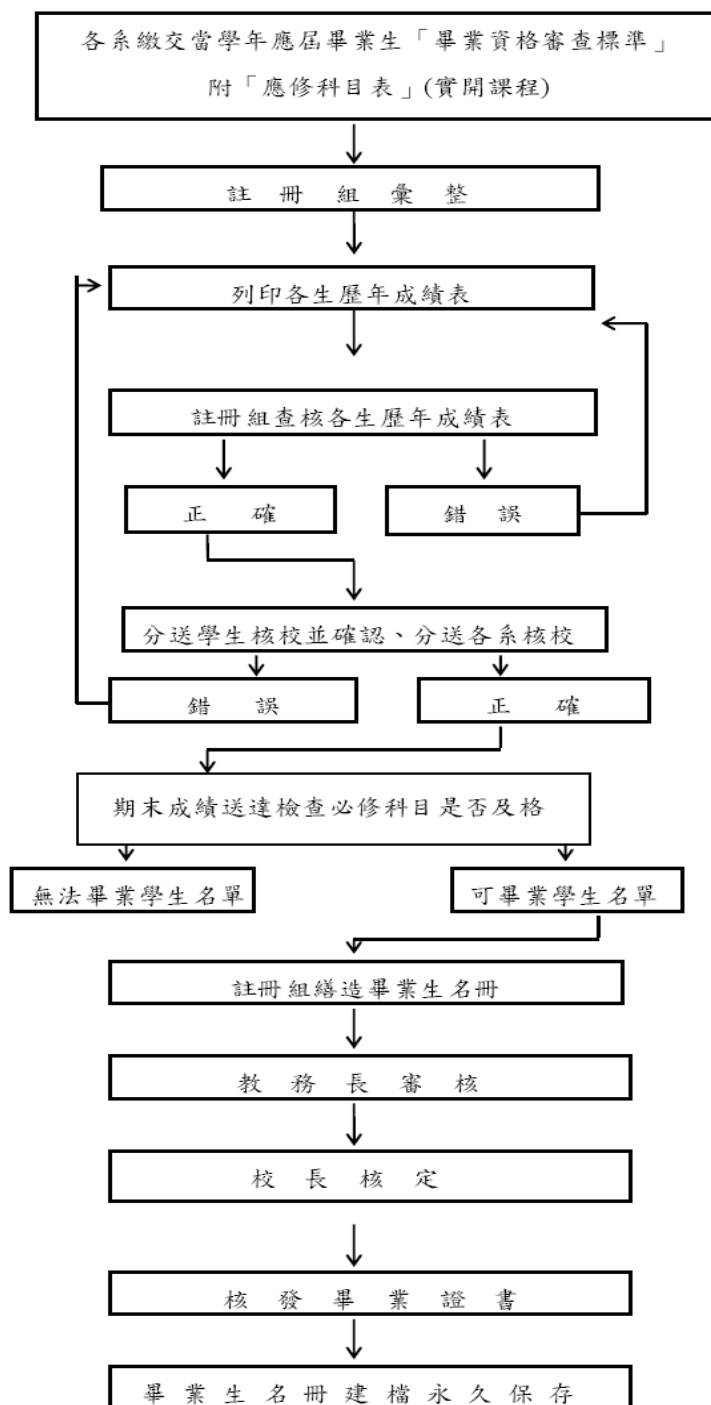


圖 1-1、畢業資格審查流程图

致理技術學院畢業資格審查的流程，說明如下：

- 各系繳交當學年度應屆畢業生之「畢業資格審查標準」以及「應修科目表」：致理技術學院各系科於畢業前的學期初，會將過去的開課紀錄整合並提交應修科目表與畢業資格審查標準，送交至註冊組。
- 註冊組彙集整理各系所(科)繳交之資料。
- 註冊組列印各生歷年成績表：註冊組收到各系送交的資料後，列印出各系應屆畢業學生歷年成績表。
- 註冊組查核各生歷年成績表：註冊組將印好之歷年成績表，用人工的方式，檢查是否有無錯誤。如有錯誤，重新整理該學生之歷年成績表並印出重新核校。
- 分送學生核校並確認、分送各系核校：註冊組查核通過的歷年成績表，將分送至各應屆畢業生及各系進行核校。如有錯誤，轉交回註冊組，且需重新校對歷年成績表是否有誤，再進行二次核對，經修正無誤後，歷年成績表再經由註冊組、各系以及學生重新核校。
- 期末成績送達檢查必修科目是否及格：待學期末成績送達至註冊組後，再以人工核校的方式核對必修科目與各系科訂定之畢業資格審核標準。如果通過審核，則列入「可畢業學生名單」；未通過審核者，列入「無法畢業學生名單」。

- 註冊組繕造畢業生名冊：最後註冊組再將「可畢業學生名單」繕造成冊。
- 教務長審核：教務長審核「畢業生名冊」。
- 校長核定：校長核定「畢業生名冊」。
- 核發畢業證書：核定無誤後，頒發畢業證書給應屆畢業生。
- 畢業生名冊建檔永久保存：最後再將「畢業生名冊」建檔並永久保存。

整個畢業資格審查的過程，是以人工的方式進行，容易發生許多盲點與錯誤。對於註冊組來說，審查作業是相當耗時與費力的。對於各系而言，各學年度入學的學生，應修科目表以及畢業資格審核標準皆不一樣，審查作業也不易進行。對於學生來說，沒有一個可以查詢目前所修習之學分數，根據應修科目表以及畢業資格審核標準，來顯示學生已完成了多少學分、還差多少學分、通識課程修了多少學分、是否需通過特定檢定或取得專業證照…等，對於學生自行審核畢業資格以及在選課的規劃上，都相當不易。

第二節 研究動機與目的

由於目前企業紛紛已將人工行政轉為行政電腦化的風潮之下，人工行政系統的畢業資格審查，又較容易造成許多困擾。例如學生在三年級下學期時，於開學選課時遇到了一個問題，就是學生所選的他系必修或他系選修之學分，學生並不知道是否為本系所(科)核可之學分，因為在選課當時，並沒有辦法從學校的選課系統中，來判斷自己所選的他系課程的學分，是否為本系所(科)核可之學分，而學分又是影響學生是否能夠順利畢業之關鍵，這個問題讓我們於學校的系統興起了加以完善的念頭，若是有一個系統，能夠讓學生知道自己目前所修習之學分，必修有多少學分、選修有多少學分、還差多少學分可畢業，即時記錄各項學生修課紀錄與各系科畢業資格條件，學生於選課時，就可以有更多的規劃，各系科與註冊組於畢業資格審查時，也可更方便、正確、快速。

基於以上原因，才會有「致理技術學院-電子化畢業資格審查系統」的發想動機。在經過數次的討論之後，我們將研究目標注重於畢業資格審查流程的數位化與自動化方面，透過前台的使用者介面跟後台的程式與資料庫的整合，讓電腦進行自動審核學生的畢業資格，並且附加給予學生或老師查詢或其他服務的功能與權限，例如哪些學分哪些系所(科)承認，並且整合為個人化的資料方便查詢。

畢業資格審查的數位化與自動化結果，不但可以節省大量的人力與時間，達到有效的管理以及增加方便性，倘若能與其它數位化的系統做整合，如：選課系統、學生查詢系統與教師查詢系統等，方能為學生、教師與學校帶來效率與方便。

第三節 研究範圍與報告書架構

本專題研究的範圍以致理技術學院現行的畢業資格審查流程為主，並參考各大專院校的畢業資格審查流程，規劃出一個新的畢業資格審查流程，並以資訊系統方式加以實作。

因為資料庫系統目前有許多的版本，我們透過資料的收集與研究分析，來評估各種資料庫系統的適用性與開發環境，經過整合與研究之後，我們以 Microsoft SQL Server 2005 來架設資料庫，使用 Visual Studio 2005 作為我們開發的工具，結合 ASP.NET 應用程式來設計使用者介面，讓系統的實用性與方便性都能互相兼顧。

所以我們針對專題研究範圍，作了進一步討論與規劃，下圖 1-2 為專題研究流程圖。

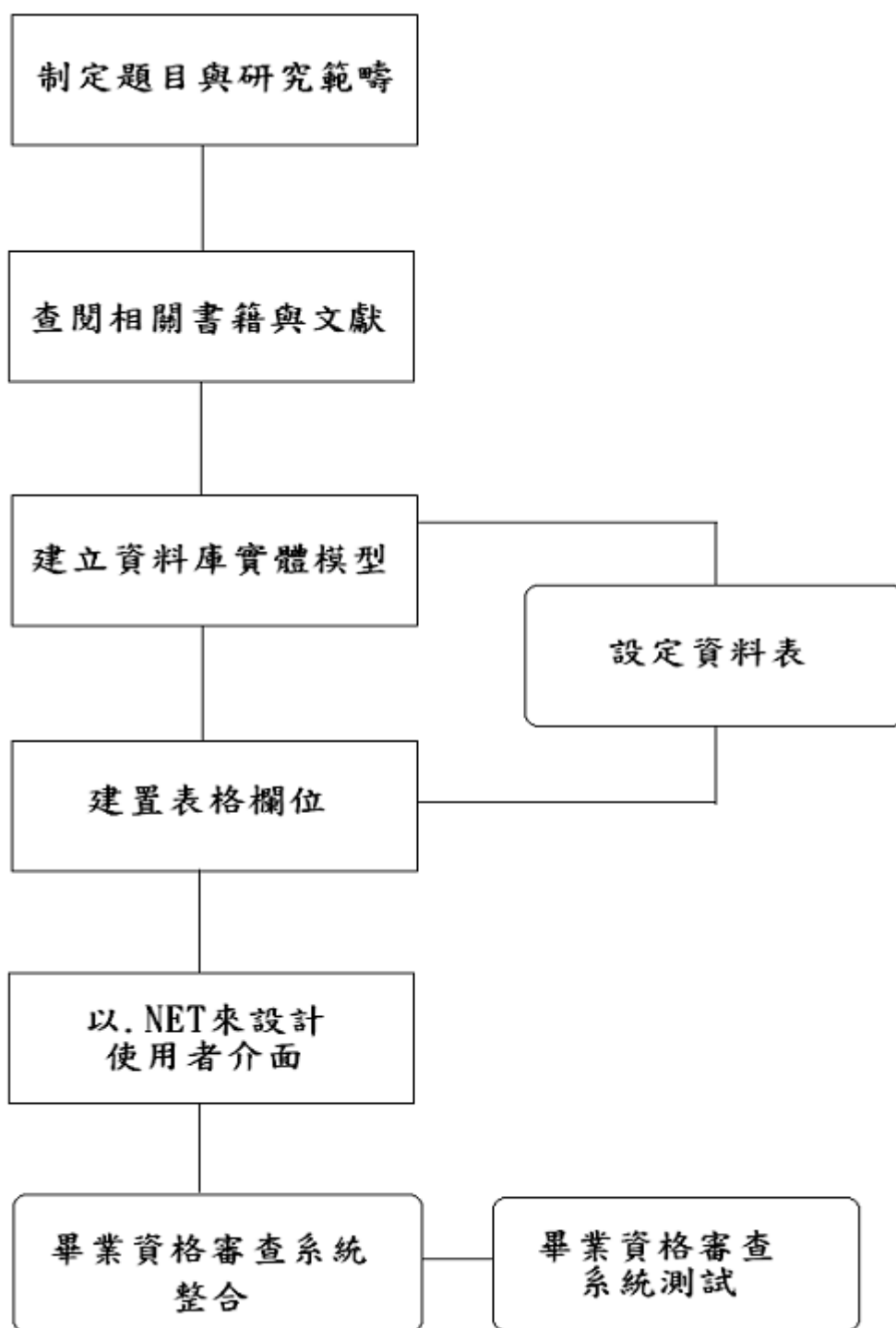


圖 1-2、專題研究流程圖

本報告書的架構，第一章主要敘述資訊數位化的實用性與方便性、研究動機與目的、研究範圍以及專題研究的流程。第二章是針對開發系統所使用的系統與開發環境作探討研究，包括.NET 技術概論、ASP.NET 技術、資料庫系統的簡介、SQL Server 2005 與 Visual Studio 2005 的介紹。第三章著重在畢業資格審查系統整體架構的說明以及系統功能的簡介與操作流程、系統的特色與技術。而第四章則是對系統預期的效能、系統實際能達到的效能以及系統的相關畫面做說明。最後第五章說明本報告書之結論以及未來可以深入研究的方向。

第二章 相關研究與文獻探討

第一節 .NET 技術概論

.NET 是由許多個程式庫集合而成，利用這個程式庫集合，使用者可以使用透過網際網路的方式來進行應用程式之設計，並且在開發環境上予以簡化。.NET Framework 是一個新的運算平臺，簡化了在網際網路高度分散式的環境中，開發應用程式的困難度。Microsoft .NET Framework 類別庫是作業系統層級的物件導向類別庫，與 Common Language Runtime 緊密整合，可供程式語言呼叫。所有支援.NET 的程式語言都可以使用.NET Framework 類別庫，減少學習新語言的時間 [1]。

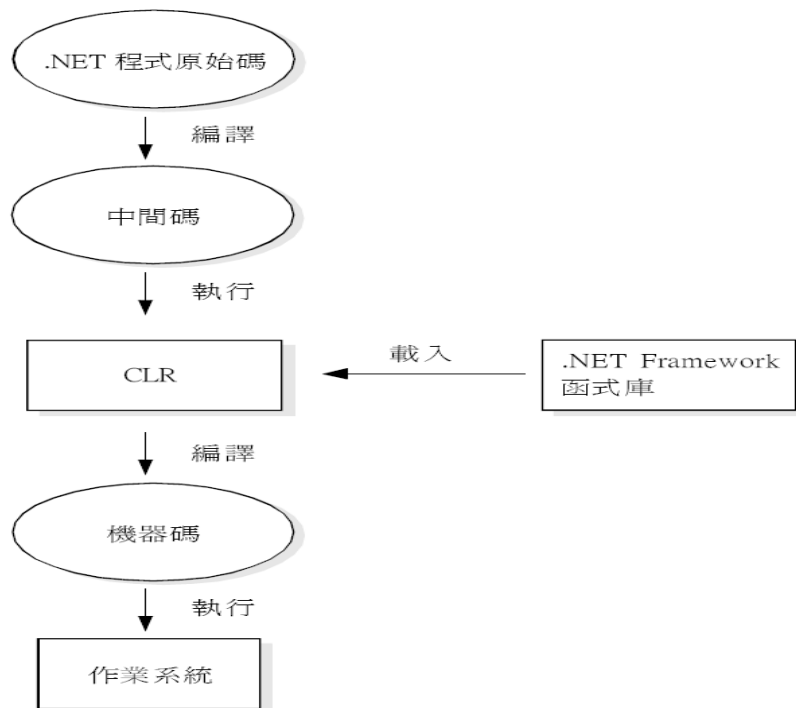


圖 2-1、.NET 的基本架構

(一) Microsoft.NET 技術

網際網路的崛起，徹底改變了資訊流通的方式，也大大的改變了人們使用電腦的方式及概念，透過網路資訊的傳輸，將世界的距離拉的更小，有了電腦網際網路，電腦的功能又多了許許多多的可能性，但是這之間其實更多的問題隨之而來，例如隨著網路資料量的倍數成長，以及發展出各種不同的平台系統之間的資料交換問題…等等。在各種軟體架構的演進之中，各大軟體廠商無不使出渾身解數想要主導軟體的走向與使用者的使用習慣，而 Microsoft.NET 則將整個網際網路視為一個作業系統，讓使用者在網際網路上開發應用程式，如同是使用自己的電腦一樣的习惯與方便。

Microsoft.NET 平台架構與簡介

Microsoft.NET 平台，簡單的說是一套能支援多種程式語言的程式執行架構，任何語言只要符合 .NET 平台所制定的規格，皆可以在上面執行。因此 Microsoft.NET 提供的服務對於軟體設計使用者而言是非常方便的。

受到分散式系統和網際網路快速發展的影響，考慮 XML 對於網際網路資料傳遞及交換之影響，以及傳統 C/C++ 語言的缺點和 Java 語言的成功，使 Microsoft.NET 實現了物件導向的設計以及兼顧程式跨平台、資源回收機制、型別安全檢查、結構化例外處理機制，再加上不同語言函式庫之整合 (C/C++、VB…) 等等功能，成為一套完整且方便使用的系統。Microsoft.NET 的技術為了要結合上述各種功能，所使用的基礎架構技術稱為 .NET Framework。

1. .NET Framework 概述

.NET Framework 有兩個主要元件，也就是共通語言執行階段 (Common Language Runtime) 以及 .NET Framework 類別庫。共通語言執行階段 (Common Language Runtime) 是 .NET Framework 的基礎。可以視為程式執行期間使用來管理程式碼的代理程式，也提供記憶體的管理、執行緒的執行、程式碼的執行、程式碼安全驗證以及編譯等等的服務。

.NET Framework 可由 Unmanaged 元件所裝載，Unmanaged 元件將 Common Language Runtime 載入它們的處理緒 (Process) 並啟始 Managed 程式碼的執行，藉此建立可同時利用 Managed 和 Unmanaged 功能的軟體環境。 .NET Framework 不僅提供數個執行階段主應用程式，也支援協力廠商執行階段主應用程式的開發。

例如 ASP.NET 裝載執行階段以提供可擴充、伺服器端的 Managed 程式碼環境。 ASP.NET 直接利用執行階段啟用 ASP.NET 應用程式和 XML Web Service [1][2]。

接著介紹 .NET Framework 二大主要元件：

- 共通語言執行 (Common Language Runtime, 簡稱 CLR) : CLR 是一種以物件導向為核心的程式執行環境，只要符合 CLR 的程式語言都可以在此環境下執行。
- 基礎類別函式庫 (Base Class Library) : 這是一套物件導向的類別程式庫，不管是 VB、C++，甚至 C# 等等，都可以使用這個類別

程式庫，因此可以縮短程式的開發時間。

從圖 2-2 可以看出作業系統與.NET Framework 的關係：

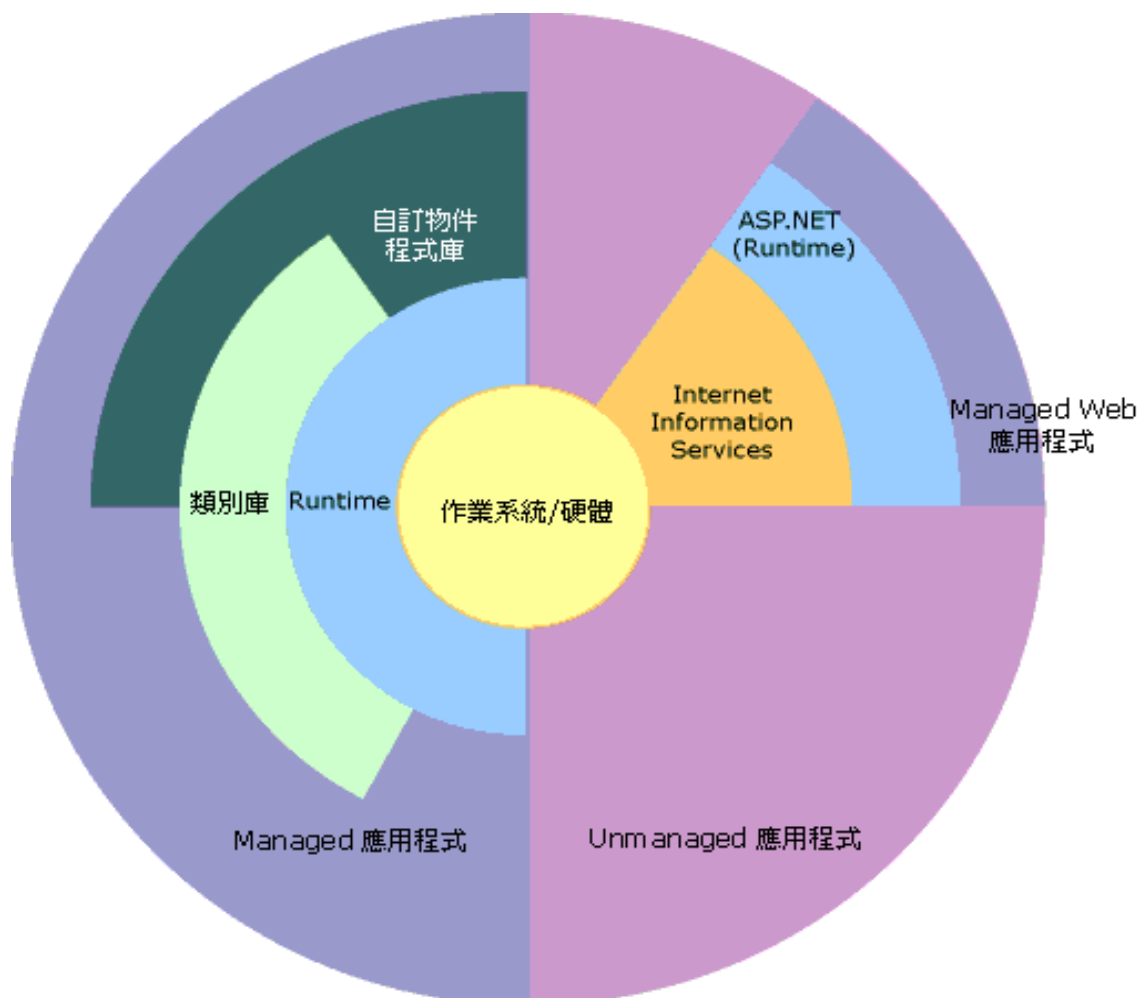


圖 2-2、.NET 架構與應用程式、作業系統之間的關係

傳統程式分為直譯式與編譯式的執行方式兩種，直譯式雖然方便，但是每次執行時都必須重新轉譯一次，導致執行效率較差；而編譯式會將程式轉譯為二進位的機器碼，執行效率較高，但由於不同電腦的機器語言有所差異，若將相同程式拿到其他平台執行時，還是必須經過重新編譯後，

才能執行。

在這個方面，.NET 採用了混合式的作法，.NET 原始程式會編譯為二進位的中間碼（MSIL, Microsoft Intermediate Language），然後透過 CLR 執行，執行時會先從 .NET Framework 類別庫中載入必要的元件，最後中間碼與元件結合後，再編譯為機器碼執行[1][2]。

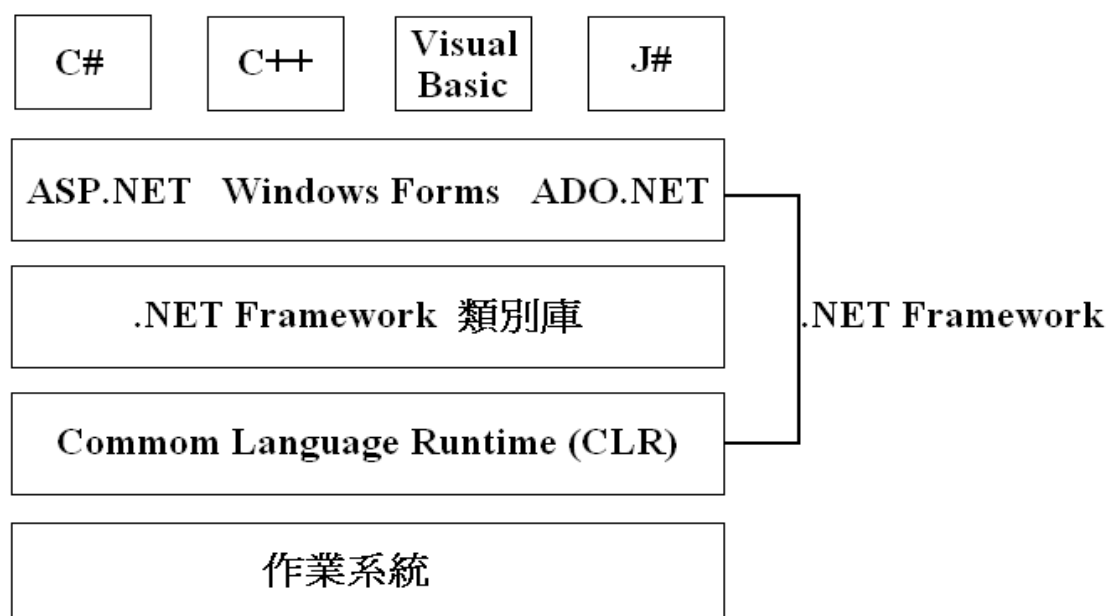


圖 2-3、.NET 架構圖

圖 2-3 為 .NET 的架構圖，在第一層可以看到 .NET 可支援 C++、C#、J#、Visual Basic 等語言，不論以哪一種語言開發程式，皆可以使用 .NET 內的元件與機制，因為各語言都使用相同的資料型別，以彼此之間可以輕易的互相呼叫與溝通。第二層則是各種開發機制，ASP.NET 是專門用來開發網頁應用程式，Windows Forms 主要針對桌面應用程式，而 ADO.NET

則是各種資料庫的存取機制。第三層為 .NET Framework 類別庫，包括了程式執行時所需的元件。第四層 CLR (Common Language Runtime，共通語言執行環境)，提供了 .NET 程式執行的環境。第二到第四層統稱為 .NET Framework，也就是 .NET 所有元件與開發機制的總集合[1]。

(二) ASP.NET 技術

ASP.NET 提供了統一的 Web 開發模型，其中包括建置企業級 Web 應用程式所需要的服務。另外 ASP.NET 也提供新的程式設計模型和基礎結構，以開發延展性更大、穩定性更佳、安全性更強的應用程式。

1. ASP.NET 架構與簡介

ASP.NET 為編譯的 .NET 架構環境，軟體設計開發可以使用任何 .NET 相容語言 (包括 Visual Basic .NET、C# 和 JScript .NET) 來撰寫應用程式。此外，整個 .NET Framework 都可供任何 ASP.NET 應用程式使用。ASP.NET 具備的程式特色，包括 Managed Common Language Runtime 環境、型別安全、繼承等等。

ASP.NET 可以將 WYSIWYG HTML 編輯器以及其他程式設計工具 (包括 Microsoft Visual Studio .NET) 一起搭配使用。這項功能使得在 Web 開發方面更為容易，也提供許多有用的功能，包括設計開發人員可用來置放伺服器控制項至 Web 網頁上的 GUI 以及完全整合的偵錯支援等[3]。

2. ASP.NET 的特性與優點

建立 ASP.NET 應用程式時，開發人員可以使用 Web Form 或 XML

Web Service，或是以任何認為適合的方式加以結合。各個功能都受相同的基礎結構所支援，允許設計者使用驗證配置、快取常用資料或自訂應用程式的組態。

Web Form 允許設計者建置功能強大的表單架構 Web 網頁。XML Web Service 提供遠端存取伺服器功能的方法。如果使用 XML Web Service，使用者可以將程式設計介面公開 (Expose) 他們的資料或商務邏輯，後者又可以由用戶端和伺服器應用程式取得和操作。XML Web Service 會使用像 HTTP 和 XML 訊息一樣的標準，將資料移動穿越防火牆 (Firewall)，在用戶端對伺服器或伺服器對伺服器案例中進行資料的交換。XML Web Service 並不受特定元件技術或物件呼叫慣例 (Calling Convention) 的限制。因此，使用任何語言撰寫、採用任何元件模型，以及在任何作業系統上執行的程式，都可以存取 XML Web Service [4]。

3. ASP 與 ASP.NET 的差異性

ASP.NET 優異於 ASP 之處，不是在於 ASP 版本的更新，而是一個全新的統一 Web 開發平臺。ASP 的直譯式語言架構，每次讀取網頁皆需逐行編譯程式碼，容易影響執行效率，而 ASP.NET 則是編譯式語言，執行編譯過的原生機器碼能夠使得效能大幅提升。ASP.NET 和 ASP 一樣可以在記事本撰寫程式，更重要的是，ASP.NET 不限於 VBScript 或 JScript 語言，可搭配 Visual Studio .NET 工具，使用任何 .NET 相容語言開發 ASP.NET，包括 Visual Basic .NET、Visual C#及 JScript .NET。

ASP 的副檔名為「.asp」，ASP.NET 的副檔名為「.aspx」，具備 ASP 開發技術的人，會發現 ASP.NET 的撰寫模式並不陌生，但是，ASP.NET 物件模型作了大幅修改，更加結構化和物件導向，並非完全回溯相容 (Backward Compatible)，大部分的 ASP 網頁必須稍加修改才能在 ASP.NET 下執行。

安裝 ASP.NET 並不會毀損 ASP 應用程式，開發人員不用擔心必須一次把所有 ASP 網頁轉換成 ASP.NET 網頁，IIS Web 伺服器容許 ASP 和 ASP.NET 網頁並存，不會互相干擾，但要特別注意 ASP 和 ASP.NET 不能共用 SESSION 和應用程式狀態。

程式設計開發人員可於 ASP.NET 的應用程式檔 Global.asax，撰寫應用程式層級的程式碼，回應 ASP.NET 或 HTTP 模組引發的應用程式層級的事件。Global.asax 存放於 ASP.NET 應用程式的根目錄，並設定拒絕任何個別網頁呼叫使用，外部使用者也無法下載或檢視內容。ASP.NET 的 Global.asax 與 ASP 的 Global.asa 可以共同存在，且是選擇性的檔案，沒有應用程式層級的判斷和 SESSION 事件處理程式，則不需要定義。

建立 ASP.NET 應用程式可視企業需求選擇開發 Web Form 或 XML Web Services，或者組合應用，因為兩者都建立在 Common Language Runtime 之上，所以都能享有相關技術的優點：安全性、繼承性、效能的提昇、執行環境的管理等。Web Form 就是 ASP.NET 網頁，它與所有的瀏覽器及用戶端裝置例如 WAP 手機、PDA 等相容，所以 aspx 程式可以在

任何用戶端裝置中執行，並自動轉譯成任何可使用 HTTP 的語言，包括 HTML、XML、WML 和 ECMAScript，簡單的說 ASP.NET 應用程式的兩種類型，能夠使效能大幅的提升。

而 Web Form 網頁的使用者介面包含兩個部分：視覺化元件和程式邏輯。視覺化元件就是 Web Form 網頁，包含靜態 HTML 和 ASP.NET 伺服器控制項。程式邏輯就是與表單互動的程式碼，可以選擇直接撰寫在 ASP.NET 網頁裡面，若用 Visual Studio .NET 開發 ASP.NET 網頁，會自動將程式碼建置在不同的檔案中，稱作「程式碼後置」(Code-Behind) 檔，可使用 Visual Basic .NET 或 Visual C# 撰寫程式碼後置檔，副檔名分別會是「.aspx.vb」和「.aspx.cs」。

Web 專案中所有的程式碼後置類別檔，會被編譯成一個動態連結資料庫.dll。ASPX 檔也會被編譯，但略有不同，使用者首次瀏覽網頁時，ASP.NET 會自動把它編譯成繼承自程式碼後置檔的.dll 檔，當使用者連結網頁時，.dll 檔會在伺服器執行，並動態產生網頁輸出。圖 2-4 為整個 ASP.NET 執行的流程[5]。

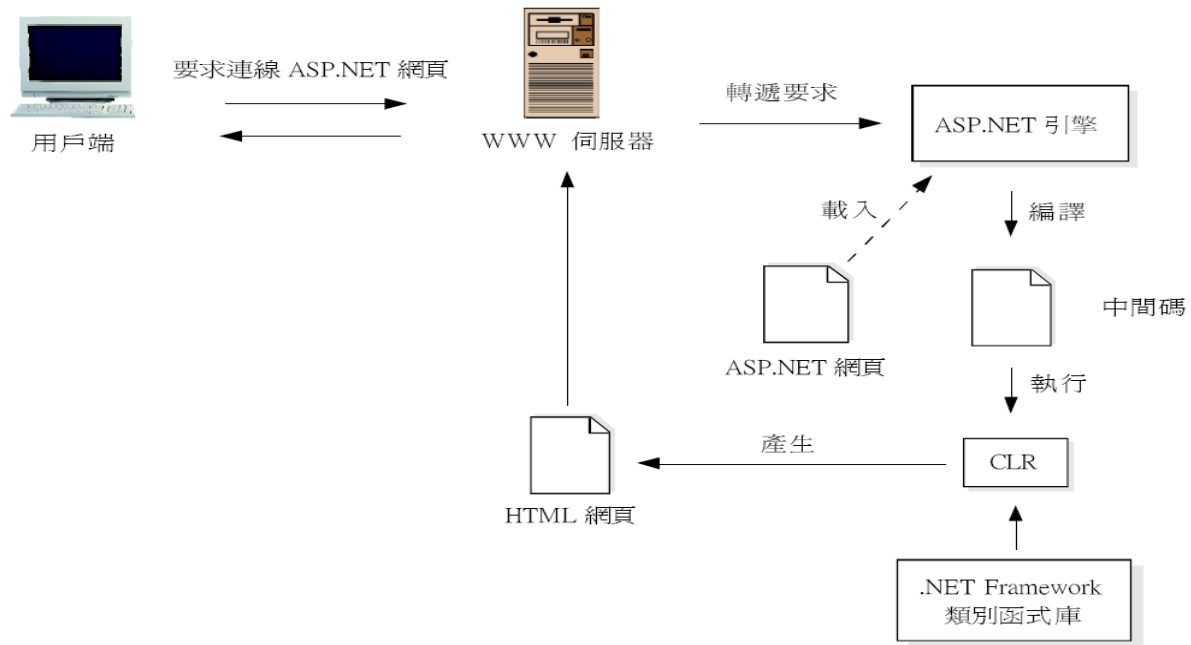


圖 2-4、ASP.NET 的執行流程

XML Web Services 是提供遠端伺服器服務的方法，主要服務的對象不是人，而是網路上的應用程式，透過 XML 文字格式和 SOAP 傳輸協定，資料可以穿越防火牆，進行企業對企業或者企業對個人的資料交換。

ASP.NET 是編譯式語言，所以執行效率大幅提升，編譯方法是使用 JIT (Just-in-Time Compilation；即時編譯)，當使用者首次存取某 ASPX 網頁時，JIT Compiler 便會即時將程式碼編譯成原生機器碼 (Native Code)，並以機器碼執行並暫存於快取，若有人存取同一個網頁時，系統便直接執行機器碼，不需再耗費系統資源和編譯時間，ASP.NET 的執行流程，如圖 2-4 所示。

快取功能可將指定的網頁和物件暫存起來，避免重複建立要求過的資訊，以節省時間和資源。ASP.NET 提供兩種快取模式：輸出快取和應用程

式資料快取。輸出快取使用 @OutputCache，將經常存取的整份或部份網頁暫存一段時間，以提升 Web 伺服器的效能。應用程式資料快取則是以程式設計的方式將物件（如：資料集），暫存在伺服器的記憶體中一段時間，節省重複建立物件的時間與資源。

ASP.NET 是可以輕易拆解的，開發人員可移除與開發的應用程式無關的模組。另外 ASP.NET 支援兩個效能計數器群組：系統和應用程式。使用者可在 Windows 系統的效能計數器記錄檔中，新增有關 ASP.NET 的記錄檔，加入 ASP.NET 系統及 ASP.NET 應用程式的效能物件，開發人員及系統管理人員便能監視應用程式的效能，也能夠依程式需求選擇狀態管理方法，利用擴充式且可繼承的組態設定，可啟動追蹤功能偵錯。

HTTP 是屬於無狀態（stateless）的通訊協定，HTTP 不會自動識別各個要求是否來自相同的使用者，也就是說無法知道使用者目前是否仍在檢視網頁。每一次要求的存取，網頁都會重新建立，所以除非設定快取，否則 Web 伺服器並不會保留網頁的變數值和控制項，只有藉著釋放伺服器的資源，Web 應用程式才能同時支援成千上萬的使用者連線使用。為了克服這個 Web 程式設計上的限制，ASP.NET 有內含檢視狀態的功能，會儲存網頁和控制項的屬性，並提供狀態管理機制，可儲存變數和特定應用程式或 session 的資訊，也可偵測第一次要求及張貼表單的時間。事實上應用程式的狀態可儲存全域性的資料，而在存取應用程式狀態最常用的方法是透過 Page 物件的 Application 屬性。開發人員應考量使用全域性變數的

影響，儲存為應用程式狀態的資料，在被移除或取代之之前，所佔用的記憶體不會被釋放，若是需要保留大型暫時性資料區塊，則可以使用 ASP.NET 快取功能[5][6][7]。

接著說明 ASP.NET 工作階段 (session) 狀態管理功能，ASP.NET 能自動識別單一用戶端的要求，在伺服器上儲存 session 資料，可以提供跨網頁的要求使用。當使用者在設定時間內未再度使用應用程式，便自動釋放資料。ASP.NET 支援數個 ASP 尚未提供的功能：

- 能自應用程式毀損中復原
- 跨多個背景工作處理分割應用程式
- 大幅改善應用程式的可用性和延展性
- 跨多個 Web Form 電腦分割應用程式

在網頁之間傳遞訊息是經常需要的，可使用 Server 物件的 Transfer 方法傳遞資料，若無安全性考量，使用 cookie 或隱藏欄位等方法，將資料儲存在用戶端，可享有較快速的伺服器效能。ASP.NET 組態是擴充式基礎架構，可在首次部署應用程式時定義組態設定，也可隨時加入或修改設定，ASP.NET 會自動偵測並套用新的設定，不需要重新啟動 IIS 或重新開機，使運作中的 Web 應用程式及伺服器的衝擊降到最低。

組態資訊儲存在 XML 架構的文字檔，可使用任何文字編輯器編輯組態檔，多個組態檔的檔名都命名為 Web.config，存放在應用程式伺服器的多個目錄下，子目錄的組態檔會繼承父目錄的所有組態設定。\\WINNT\

Microsoft.NET\Framework\version\CONFIG\Machine.config 是整個 Web 伺服器適用的組態設定。

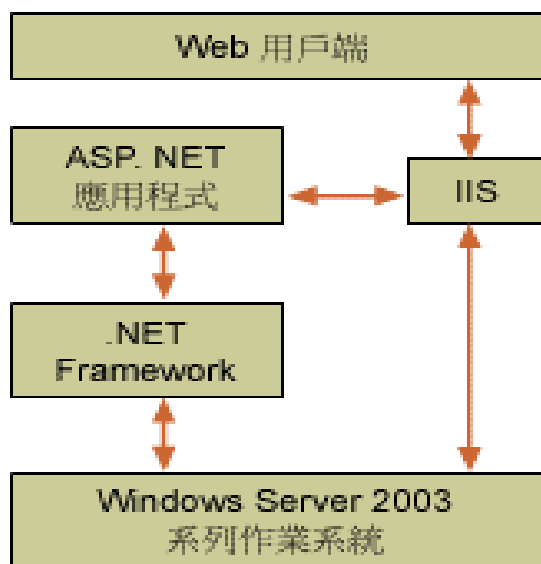


圖 2-5、ASP.NET 安全性架構圖

如圖 2-5 所示，所有 Web 用戶端都是透過 Microsoft Internet Information Services (IIS) 與 ASP.NET 應用程式通訊。IIS 會視需要驗證要求，然後找出所要求的資源（例如 ASP.NET 應用程式）。如果用戶端已獲得授權，即可使用資源。當 ASP.NET 應用程式在執行時，可以使用內建的 ASP.NET 安全性功能。此外，ASP.NET 應用程式可以使用 .NET Framework 的安全性功能[4][7]。

多數的網站資源是開放給所有使用者存取，但是像信用卡資料、股票交易資料、付費資訊等不能公開的資料，則需要安全的防護機制，才能確

保資料的安全性和使用者權益。ASP.NET 結合 Microsoft .NET Framework 和 IIS 提供的 Web 應用程式驗證、授權及模擬三種主要的安全性機制。驗證可透過帳號密碼、數位簽章、智慧卡等憑證資料，確認使用者身分。當登入者通過驗證，便可依據使用者權限，授權能否進行讀取、寫入、新增、修改、刪除等作業。模擬主要的目的是判斷使用者的要求，在使用者的帳戶之下執行，或在本機系統帳戶之下執行。模擬功能預設是停用的，開發者必須在組態檔加入模擬設定。當要求從 IIS 傳送到應用程式時，ASP.NET 會檢查組態檔，是否啟用使用者帳號模擬，或透過匿名帳號存取本機資料。

相對地部署 ASP 元件十分麻煩，因為 IIS 會鎖定 (lock) 線上服務的元件，所以更新元件必須先停掉 IIS 服務，再執行 regsvr32.exe 註冊新的元件，最後再啟動 IIS 服務才能更新。

ASP.NET 在這方面有了很大的進步，IIS 不再鎖定 dll 檔，所以可在服務不中斷的情況之下，將新版的元件複製到指定的目錄下。相較於 ASP，ASP.NET 在安全性、效能、相容性、延展性各方面都大幅改善許多，雖然 ASP 和 ASP.NET 可以並存，但是以 ASP.NET 的優異性來看，未來趨勢將逐步把 ASP 程式移轉到 ASP.NET 上[4][8]。

第二節 資料庫系統

(一) 資料庫系統簡介

資料庫系統 (Database System) 是一個電腦化的資料儲存系統，一般來說我們常提到的「資料庫」(Database)，僅是「資料庫系統」的一部分而已。「資料庫系統」是由「資料庫」和「資料庫管理系統」(Database Management System ; DBMS) 所組合而成。

對於資料的保存與處理，從早期的人工處理方式一直發展到現在利用資料庫系統這個強大的助手來協助整理，僅僅數十年的光陰的發展。在 70 年代的大型 Mainframe 時代，資料庫需要大型主機在進行運算與資料的儲存；到了 80 年代的個人電腦的盛行，資料庫系統也跟著應用在個人電腦、PDA 的身上。而當物件導向技術逐漸成熟時，資料庫系統也跟著這股物件導向的熱潮，將資料庫系統導入物件導向的設計。90 年代則是一個屬於全民進入網際網路的時代，各式各樣的軟體應用都搬到了虛擬網路的世界，資料庫也不落人後地加入了這股無法抵擋的趨勢，電腦可以透過網際網路來進行交換資訊，而一直進步的電腦硬體設備將會提供足夠的運算能力，來解決許多目前仍無法順利進行的大量運算問題。

資料庫 (Database) 是相關資料的集合。藉由資料 (Data)，使用者能得知所紀錄下來的事情以及它所隱涵的意義。舉例來說，跟自己所有的同學、老師、同事、朋友拿到姓名、電話、住址、E-mail，你可能將這些資料記錄在手機或者有索引的通訊錄裡面，或者存在電腦裡面使用 Excel 建

檔，而這些資料可以視為是一個隱含意義的相關資料集合，因此是一個資料庫。當然也可以把組成一頁的所有單字集合起來，作為相關資料，也能構成一個資料庫。因此前述二例對於資料庫的定義較為籠統，通常資料庫這個術語的條件限制是比較多的，一個資料庫會有以下幾種特性：

- 資料庫代表真實世界裡面的某一方面，有時會將此稱為迷你世界（Mini-World）或論域（Universe Of Discourse，簡稱 UOD），這個迷你世界的改變會反應在資料庫裡面。
- 理論上來說，資料庫與資料的集合，在原來的意義上是相連貫的，隨機抽取任何一類資料，並不能稱為一個資料庫。
- 資料庫是爲了特定目的而設計、建立並存入資料，它有一群預定的 User，以及針對這些使用者設計的應用程式。

換句話說，一個資料庫的資料來源是從衍生的資料得來的，並且在真實世界裡，它與事件以及使用者會產生某種程度的互動。資料庫可以是任意的大小以及多樣的複雜程度，舉前例來說，姓名地址電話 E-mail 的通訊錄，可能只有幾百筆紀錄（Record），每筆紀錄都有簡單的結構，記載著索引與姓名、電話、住址、E-mail 的資料；但是圖書館可能就有五十萬張書卡放在不同的目錄，主要以作者的姓氏、主題、書名來區分每個目錄，相較之下則比通訊錄要龐大複雜上許多[9]。

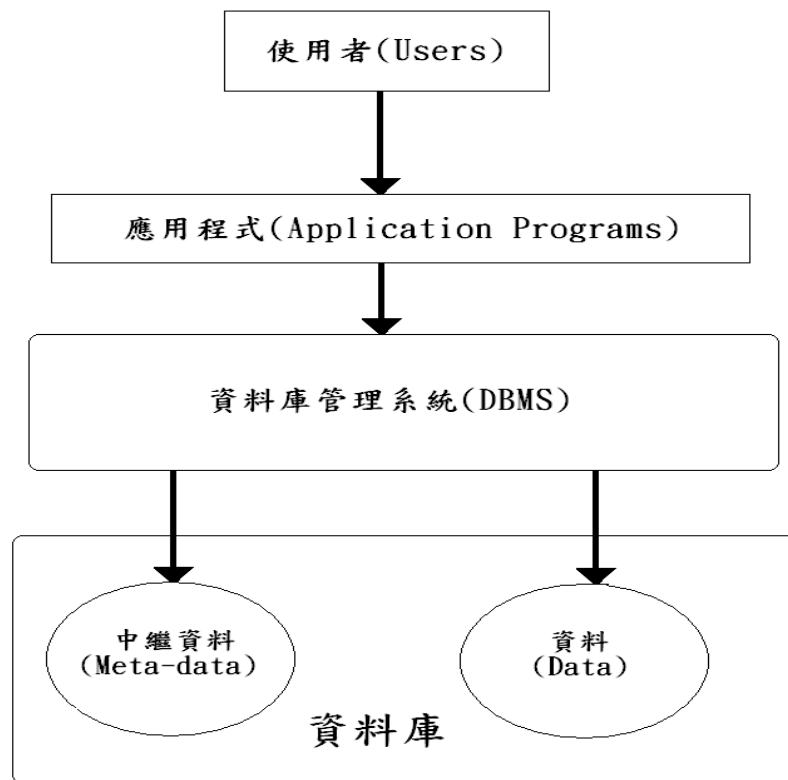


圖 2-6、簡單的資料庫系統環境

如圖 2-6 所示，一個資料庫的系統環境，資料庫除了建立與使用之外，這些龐大的資料也必須要加以組織與管理，如此使用者才能在需要時作搜尋或者檢索與更新資料。在資料庫的技術之中，資料庫的儲存方式只需要被定義一次，就可讓不同的使用者來存取資料，相較於傳統的檔案處理方法，資料庫的技術有下列幾種主要的特徵：

- 資料庫系統的自我描述特性。
- 程式與資料間的獨立性，及資料抽象化。
- 支援資料多重景觀。
- 支援共享與多使用者的交易處理。

1. 資料庫的類型

資料庫的類型可以依照資料模型來區分如下：

- (1) 階層式資料庫 (Hierarchical data model)：將資料組成類似樹狀的階層模式，簡單的說就是把資料看成是一組樹狀結構的型態。
- (2) 網路式資料庫 (Network data model)：一個資料庫，其子元素可置一個或多個母元素，簡單的說就是把資料看成一組網狀結構的型態 (最早研發出來)。
- (3) 關聯式資料庫 (Relational data model)：簡單的說就是把資料看成一些彼此有相關的表格，目前以此類的資料庫最為成功 (大概是因為觀念簡單好實作)。
- (4) 物件導向式資料模式 (Object-Oriented)：資料以物件的方式處理，包含類別、繼承等。

2. 關聯式資料庫與內部結構

關聯性資料庫 (Relation Database) 的觀念在 1970 年正式由 IBM 提出，從此改變了資料庫的管理與應用。過去所有資料是以單純的儲存方式進行堆積，電腦無法進行有效的資料管理。而關聯性資料庫的設計，是把所有資料都以表格的形式儲存，經由表格正交化 (Normalize)，依照有效率的資料表設計流程，降低資料的重複率，並將稽核記錄有效地儲存於資料庫中，達到資料庫結構最佳化。

關聯性資料庫能使個人或企業能夠以更快速、更簡易的方式來建立、取得或管理大量的資料，大幅提昇工作效率。

關聯式資料庫組成元素，如下所示：

- (1) 資料結構：(Date Structures) 資料的組成方式，以關聯性資料庫的模型來說，就是欄和列組成表格的關聯表 (Relations)。
- (2) 資料操作或運算 (Date Manipulation 或 Operational)：資料的相關操作是關聯式代數 (Relation Algebra) 和關聯式計算 (Relation Caluclus)。
- (3) 完整性限制條件 (Integritl Constraints)：維護資料完整性的條件，其目的是確保儲存的資料是合法的資料。

3. 資料庫系統的網路架構

網路架構指的是資料庫系統要如何佈署的問題。通常，我們會依組織的大小與規模、資料量的多寡、使用的人數、軟 / 硬體設備等條件來作為考量的條件，一般而言常見的有下列 4 種網路架構：

- (1) 單機架構：單機架構是由同一部電腦包辦所有資料庫系統的工作，包括保存資料、處理資料、管理及使用資料庫系統 ... 等等。適合使用者少、資料量也不多的資料庫系統使用，如小公司或個人使用者所建立的資料庫系統。通常使用 Access、FoxPro 所設計的資料庫系統多採用這種架構。

(2) 大型主機/終端機架構：大型主機/終端機架構是由一部大型主機負責儲存及處理龐大的資料，使用者則透過終端機與大型主機連線，以存取資料庫的內容。這種架構的缺點在於當多人同時使用時，由於所有的工作都要由大型主機來處理，因此會非常忙碌，容易造成回應緩慢的問題。目前除了一些大型機構外，已比較少使用這一類的架構了，而且此類的大型主機價格都相當昂貴，一般中小企業可能負擔不起。

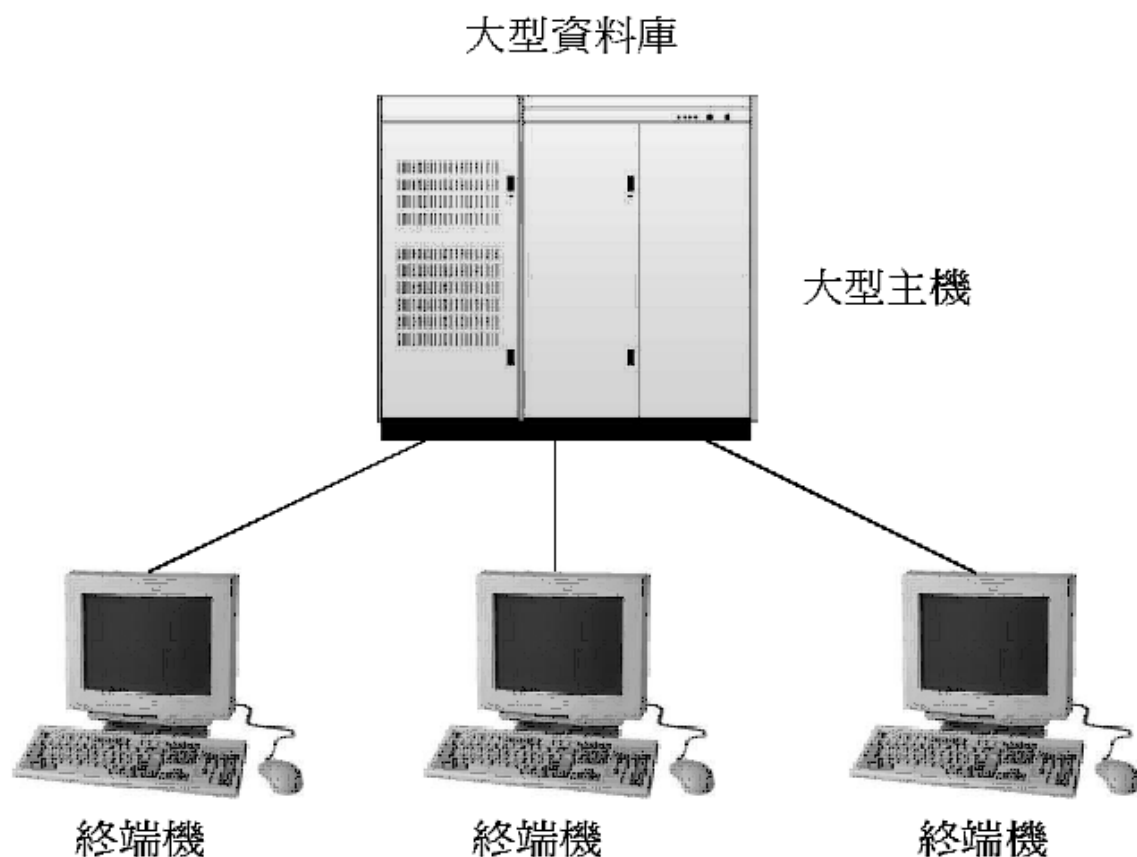


圖 2-7、大型主機/終端機架構

(3) 主從式架構：由於個人電腦的價格低廉，運算速度也不錯，利用網路互相連接之後，作為用戶端（Client）的各台電腦只要連結到做為資料庫伺服器端（Server）的電腦，就可以存取資料庫，而且部份的工作可由用戶端電腦來處理，分散資料庫伺服器的負荷，這就是主從式架構的佈署方式，同時也是目前一般公司中最普遍採用的方式。若採用主從式架構，通常還會另外撰寫用戶端程式，以提供使用者易學易用的操作介面。

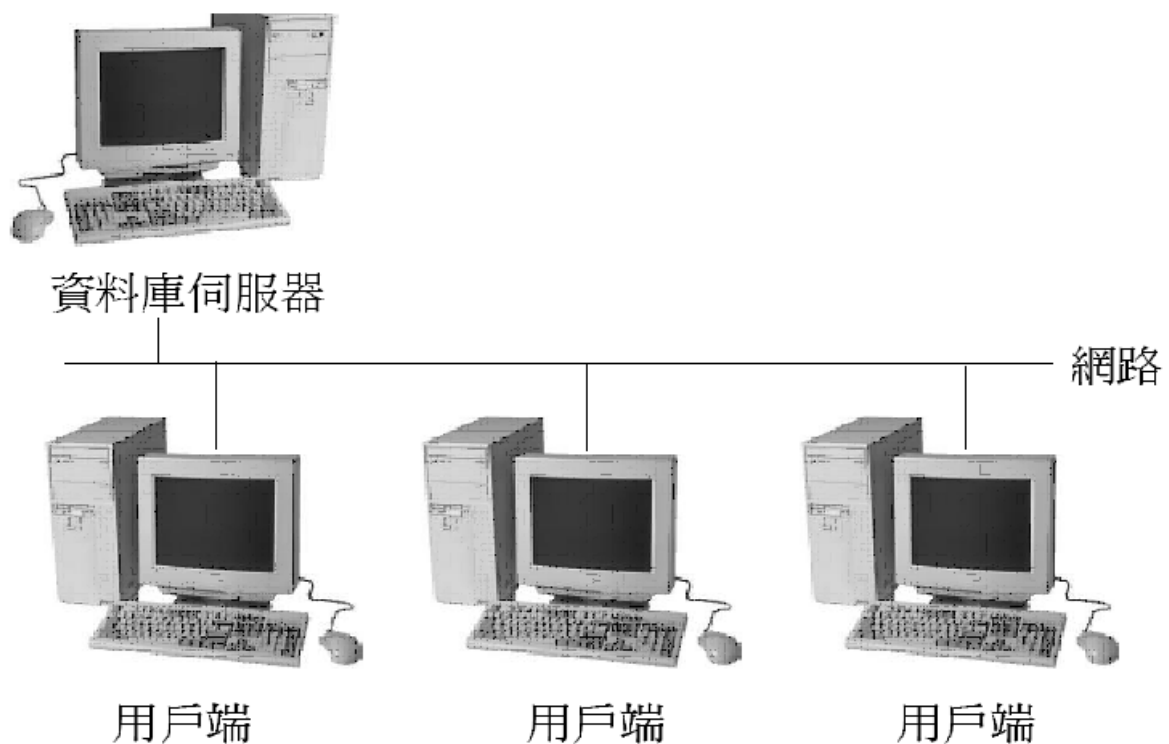


圖 2-8、主從式架構

(4) 分散式架構：分散式架構是由數台資料庫伺服器所組成，使用者在存取資料時，資料可以來自於不同的伺服器中，如此在存取的效率上會比較好。分散式架構的資料存取方式和主從式架構類似，只不過是多了幾台資料庫伺服器而已。



圖 2-9、分散式架構

4. 資料庫管理系統的基本功能

資料庫管理系統其實就是管理資料庫的軟體系統，它們要負責整個資料庫的建立、資料存取、權限設定、資料備份、操作監督與記錄等等的工作。接著就再進一步詳述資料庫管理系統（DataBase Management System, DBMS）所應具備的基本功能。

- (1) 資料定義：DBMS 必須能夠充份定義並管理各種類型的資料項目，例如關聯式資料庫管理系統必須具備建立資料庫、資料表、定義各欄位的資料型別以及資料表之間的關聯等等的能力才行。
- (2) 資料處理：DBMS 必須提供使用者對資料庫的存取能力，包括新增、修改、查詢與刪除等基本功能。有時 DBMS 提供的功能雖然完善，但是並不是很適合一般的使用者操作，這時就需要程式設計師另外再撰寫用戶端的應用程式，以供一般使用者操作。
- (3) 資料安全：DBMS 應該具備設定使用者帳戶、密碼、及權限的功能，讓每一個使用者只能存取授權範圍內的資料，以防止機密資料外洩，或資料庫遭受任何有意或無意的破壞。
- (4) 資料備份：DBMS 必須提供方便的資料備份功能，如此在資料庫不幸意外毀損時，還可以還原到備份資料時的狀況，以減少損失。

此外，維護資料庫的效率也是非常重要，尤其是在資料量很大或使用者很多的時候，資料庫若因效率不佳而導致存取速度變慢，亦會嚴重影響到操作人員的工作效率[10][11][12][13]。

(二) SQL Server 2005

簡單來說，SQL Server 2005 是個採主從架構的關聯式資料庫系統。不過這個簡單的敘述，幾乎可用來形容市面上所有的主流資料庫系統產品，實在無法由此看出 SQL Server 的特點。對於管理者而言，必須對 SQL Server 的整體架構有基本的認識，才能在進行管理工作時，做出正確的決

策，執行有效的管理工作。以下對 SQL Server 2005 的架構，做一簡要的說明[9][14]。

1. 伺服器架構

以 SQL Server 的主從架構而言，就是由 SQL Server 扮演存放資料和提供資料給用戶端的角色，至於使用者要取用資料時，則可由各種不同的用戶端應用程式，透過 SQL Server 所支援的介面，向 SQL Server 提出要求，然後取得資料庫中的資料並顯示給使用者。

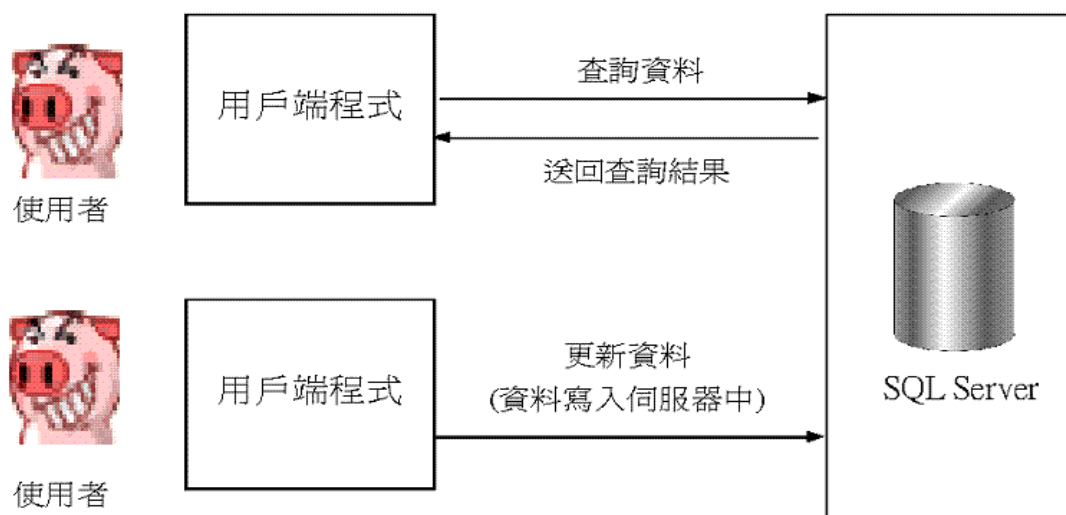


圖 2-10、SQL 主從架構

不過以上這種方式，是屬於較傳統的兩層式主從式架構。近年來，三層式 (3-tier) 或多層式 (N-tier) 主從架構的應用也逐漸廣泛，SQL Server 當然也支援在此環境下的運作，例如再搭配微軟 IIS 伺服器、ASP.NET 技術或其它伺服器產品，即可建立多層的主從運算模式。

2. 功能與特色

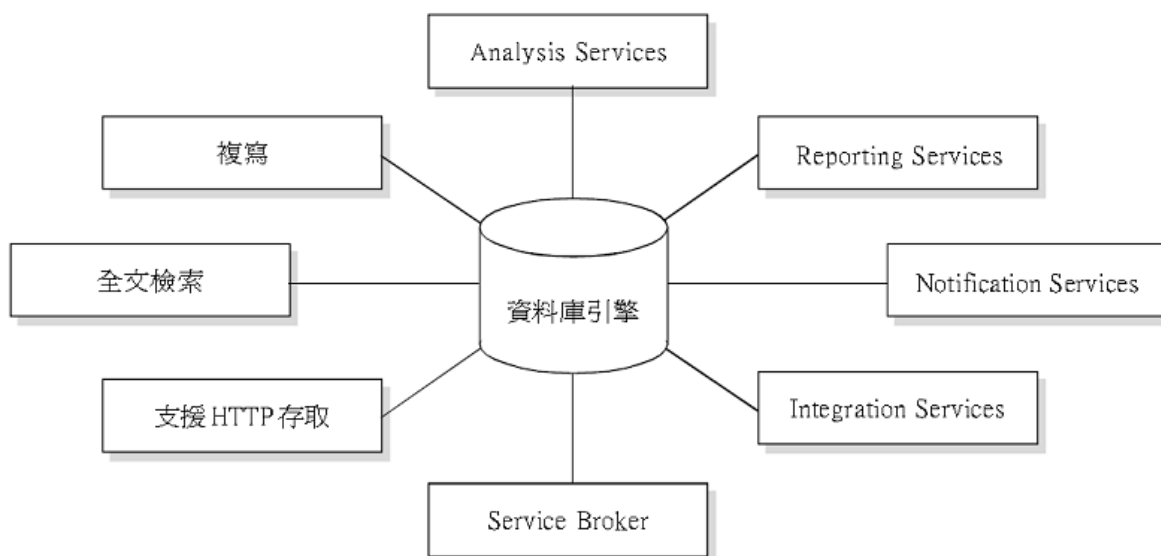


圖 2-11、SQL Server 2005 的架構

SQL Server 2005 各個組成的功能與特色：

- 資料庫引擎：這是 SQL Server 最核心的元件，提供資料庫的寫入與讀出等存取功能。SQL Server 2005 已經整合了 .NET CLR，所以除了使用 T-SQL 語言存取資料庫外，現在也可以直接用 .NET 撰寫資料庫的預存程序、自訂函數...等。
- Analysis Services：這是微軟的商業智慧（BI, Business Intelligence）解決方案之一，此服務能從各種維度取得資料，徹底達成決策支援的功用。
- 全文檢索：全文檢索可以讓資料庫從長度較長的欄位中，快速搜尋到需要的資料。SQL Server 2005 已經內建全文檢索的搜尋引

擎，所以不需要額外加裝 Microsoft Search 服務。

- 複寫：複寫功能可以讓資料庫由某處自動複製到一個或多個目的地，提供資料庫的分散式應用。
- 支援 HTTP 存取：SQL Server 2005 直接支援 HTTP 存取的功能，不需要額外加裝 IIS，就可以讓使用者透過 SOAP 呼叫取資料庫中的資料。
- Service Broker：此功能用來取代過去的 message queue，提供儲存訊息佇列的機能，配合新的 T-SQL 語法時，還可以從佇列傳送和接收訊息，所以可以用來開發非同步或分散式資料庫應用程式。
- Reporting Services：這個功能必須配合 IIS 才能使用，其主要用途在於讓使用者可以從資料庫的資料建立各式報表。
- Notification Services：此功能可以讓管理者使用 XML 定義各種事件（Event），之後事件發生時，SQL Server 便會立刻通知訂閱者並進行相關處理工作。
- Integration Services：此功能可以讓管理者將大量資料從 Excel、Access 或 Oracle 等其他資料庫轉入或轉出 SQL Server。

3. 安全機制

要使用 SQL Server 的服務，必須先「登入」SQL Server，所謂登入就是以系統認得的使用者帳戶和密碼通過伺服器的驗證（authentication），然後才能進入系統並使用其中的各項服務。但是系統為了簡化 Windows

2000/2003 使用者的登入過程，SQL Server 除了支援以 SQL Server 本身帳戶做登入外，也特別支援直接以 Windows 2000/2003 的使用者帳戶登入 SQL Server 的方式。因為 Windows 2000/2003 使用者在進入 Windows 網路環境時，就需先以 Windows 2000/2003 的使用者帳戶來進行登入。因此到了準備使用 SQL Server 服務時，使用者應該已經通過安全驗證。

若是使用者將 SQL Server 設定為可接受 Windows 2000/2003 的使用者帳戶，管理者就可直接將 2000/2003 中的使用者或群組帳戶設定成 SQL Server 的登入帳戶，並可設定是允許存取 SQL Server 或禁止存取 SQL Server 的帳戶。

而使用者在登入 SQL Server 時，也不需再以專屬於 SQL Server 的登入帳戶來登入，SQL Server，而是會直接根據使用者目前所用的 Windows 2000/2003 的帳戶來查核此帳戶是否允許登入。這對管理者和使用者都是相當方便的的機制。

4. 平行處理能力

接下來要介紹 SQL Server 平行處理能力。Windows 2000/2003 是個具有平行處理能力（支援多 CPU）的作業系統，Windows 2000/2003 所採用的是對稱式平行處理（SMP），也就是說運算的工作會分配到各處理器上來分工完成，因此可大幅提升系統的效能。但並非任何的應用程式放在有多個處理器的 Windows 2000/2003 電腦上，就能發揮平行處理的效用，應用程式必須是針對平行處理的架構來設計才行。例如多執行緒的應

用程式，就能讓各執行緒分別在不同的處理器上來處理，加速完成其工作。而 SQL Server 正是個多執行緒的應用程式，因此在配備多重處理器的平台上，就能發揮其平行處理的能力，使伺服器的效能有所提升。

5. SQL Server 版本在平行處理方面的不同支援情形

(1) 以服務的方式執行

在 Windows 2000/2003 中，SQL Server 是以服務 (Service) 的方式被執行。所謂的服務是一種較特殊的應用程式執行方式，它可以由作業系統自己來啟動程式。以服務啟動的程式，會固定在背景執行，以便隨時向使用者提供「服務」。

服務可設定成在系統啟動時就自動被啟動 (SQL Server 預設就是會自動啟動)，如此一來，只要電腦一開機，該項服務就已準備好回應使用者對它的要求了。使用 Windows 98/ME 的使用者，可能會想：「只要自己設一個啟動電腦時就會自動執行的應用程式，如此豈不也和服務一樣是一開機就執行」。其實這兩種方式仍是不一樣的！在 Windows 98/ME 的開始/程式集/ 啟動資料夾下所放的應用程式捷徑，必須等到使用者「登入」作業系統之後才會被執行。也就是說若在開機後，不做登入而讓登入交談窗一直停留在畫面上，則這些捷徑並不會被執行。

但服務就不同了，在 Windows 2000/2003 中將服務設成了自動啟動後，則只要作業系統已啟動完畢，就算沒有人去登入該電腦，服務也已經是在執行中的狀態了，此時網路上的使用者，就能向電腦中已啟動的服務

提出要求，例如向 SQL Server 提出查詢的動作等等。

(2) 整合於 Windows 2000/2003 的功能

Windows 2000/2003 本身有很多系統管理的功能，而 SQL Server 也特別加以利用以達到與作業系統結合的目的。例如在使用層面上最被重視的項目之一就是效能，而 SQL Server 就支援將其執行時的各項效能數據顯現在 Windows 2000/2003 的效能監視器中因此系統管理者能以他原本熟悉的工具來監視 SQL Server 的效能表現。

此外 SQL Server 除了會將各類錯誤或重要事件記錄到自己的錯誤記錄檔中，也會將各項事件記錄到 Windows 2000/2003 的事件記錄中。因此有狀況發生時，管理者就能用 Windows 2000/2003 的事件檢視器來查看與 SQL Server 相關的記錄，來找出問題的所在。

(3) 善用加強的磁碟 I/O 功能

資料庫伺服器為了提供眾多使用者的查詢服務，必須有強大的運算能力來處理各種查詢需求；另一方面也要能快速存取磁碟上的資料，以便讀出資料供使用者使用，或是將新的資料寫入。因此 SQL Server 提供了 Windows 2000/2003 兩項特殊的磁碟 I/O 功能以提升其資料處理效能。

- a. 非同步 (Asynchronous) I/O：Windows 2000/2003 提供了一種稱為非同步 (Asynchronous) I/O 的處理方式。當程式送出 I/O 的要求給作業系統時，Windows 2000/2003 會立即將控制權返回給應用程式，讓程式可繼續進行其它的處理工作，稍後再來檢查看看

資料是否已讀出或寫入。而傳統的同步式 I/O 則是在送出 I/O 需求給作業系統後，需等作業系統確實完成 I/O 的動作，才會將控制權交還應用程式，因此在 I/O 較耗時的情況下，就會使應用程式的效能大幅下滑。

- b. Scatter-gather I/O：Windows 從 NT 4.0 Service Pack 2 之後使用了一種新的 I/O 模式，稱為 Scatter-gather I/O。因為早期在 Windows NT 中，每次讀出或寫入磁碟的資料，都必須放在連續的記憶體空間才行，在這種情況下，若 SQL Server 要將分散在數個不連續空間中的資料寫入磁碟，必須先另行將它們複製到一塊連續的記憶體空間才行，SQL Server 就可省下此一步驟，Windows 2000/2003 自己會把這些分散的部分放在緩衝區來做處理[9][14]。

(三) 小結

從以上資料庫系統的介紹，資料庫管理者的工作可以歸納為以下幾個主要項目：

- 安裝與更新 SQL Server
- 建立與管理資料庫
- 管理安全性
- 備份與回存資料庫
- 維護伺服器運作
- 協調資料庫

資料庫管理者負責的工作內容相當的多，因此要做好資料庫管理者的工作必須具備能有效完成各項工作的知識如下列：

- 作業系統
- SQL Server
- 關聯式資料庫
- T-SQL
- 組織規則與需求

雖然我們能從書籍、論文以及期刊取得大多數的資訊，不過為尋找各種相關問題的解決方案，資料庫管理者也需要不時上網取得最新知識，並且藉著對系統的了解與知識來打造一個量身訂做的系統。

第三節 Visual Studio 2005

Visual Studio 是用來建置 ASP.NET Web 應用程式、XML Web Services、桌面應用程式及行動應用程式的一套完整開發工具。可以使用 Visual Basic、Visual C++、Visual C# 和 Visual J# 等不同程式語言，於相同的整合式開發環境（IDE）中開發，讓它們能共用工具和協助混合語言方案的建立。此外，這些語言可利用 .NET Framework 強大的功能，簡化 ASP Web 應用程式與 XML Web Services 的開發工作。這個小節中，將介紹 Visual Studio 2005 的功能與開發環境。

（一）Visual Studio 2005 功能

Visual Studio 2005 是一個整合式並且可擴充的軟體開發週期工具平台，可以協助軟體小組改善整個軟體開發程序的溝通與共同作業能力，它包含了下列部分：

- Team Foundation 是可擴充開發者共同作業伺服器，可提供工作項目追蹤、原始檔控制、報告和程序導引等功能。
- Team Edition for Architects 是一組整合式的應用程式設計工具，可用於以服務為導向的開發工作。
- Team Edition for Developers 提供程式碼品質工具與效能工具，可讓開發者用來建置可靠、重要的服務和應用程式。
- Team Edition for Testers 提供進階負載測試工具，可讓開發者在部署之前先驗證應用程式的效能[15][16]。

(二) Visual Studio 2005 系統環境介紹

1. Visual Studio Tools for Office：可以協助使用者使用 Visual Basic 和 Visual C# 擴充 Word 2003 文件和 Excel 2003 活頁簿以建立方案。Visual Studio Tools for Office 包含 Visual Studio 的新專案，可在 Word 文件、Word 範本、Excel 活頁簿以及 Excel 範本的背後建立程式碼，如圖 2-12 所示。

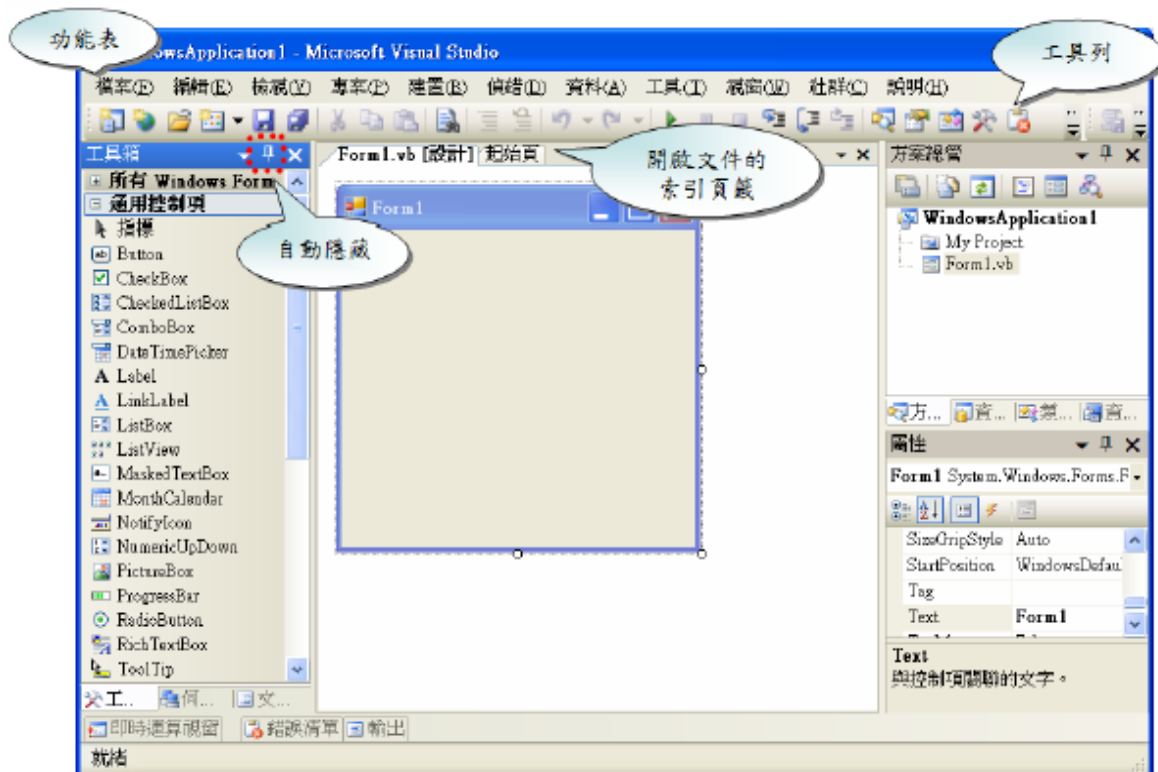


圖 2-12、Visual Studio Tools for Office 的環境

2. Visual Web Developer：Visual Studio 提供一個名為 Visual Web Developer 的新網頁設計工具，其中包含許多建立和編輯 ASP.NET 網頁和 HTML 網頁的加強功能。它提供一個比 Visual Studio .NET 2003 更簡單且更快速的方法來建立 Web Form 頁面。Visual Web Developer 提供網站開發所有層面的改善。可以在網際網路資訊服務 (IIS)、FTP 或 SharePoint 伺服器上以本機資料夾來建立和維護網站。Visual Web Developer 設計工具支援所有的 ASP.NET 加強功能，包含簡化許多 Web 開發工作的 24 個新控制項。



圖 2-13、工具列

3. 智慧型裝置應用程式：Visual Studio 整合式環境包含的工具，目標對象是 PDA 和 Smartphone 等裝置。對於 Visual C++ 工具、原生裝置執行階段以及 Managed 設計工具的改善，包含提供平台特定 WYSIWYG 的改善、多重表單因素支援、新模擬器、與桌面相似的資料處理工具以及排除以手動編輯 inf 檔案的使用者開發專案。

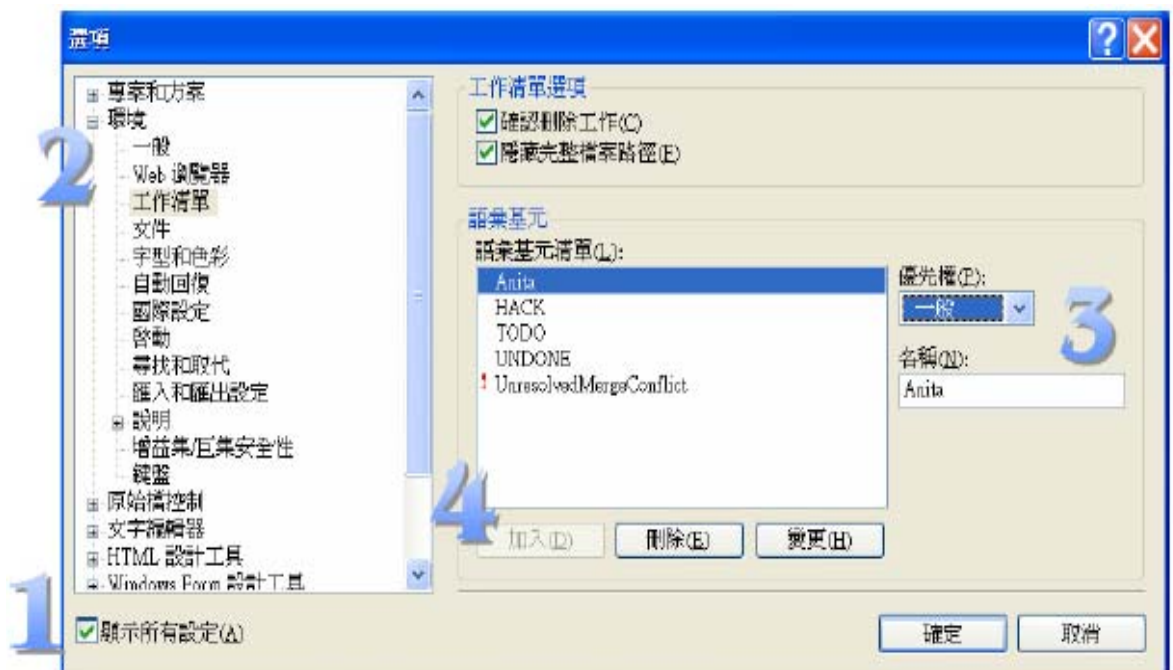


圖 2-14、工作清單

4. Web Form：Web Form 是用來建立可程式化網頁的 ASP.NET 技術。Web Form 本身會轉譯為瀏覽器相容的 HTML 和指令碼，可以讓任何平台上的任何瀏覽器檢視網頁。使用 Web

Form 時，只要將控制項拖放至設計工具，然後加入程式碼，就可以建立 Web 網頁，類似建立 Visual Basic 表單的方式。

5. Windows Form：Windows Form 是用來在 .NET Framework 上建立 Microsoft Windows 應用程式。這個架構提供清楚的物件導向可擴充類別集，可以開發多樣的 Windows 應用程式。除此之外，Windows Form 還可做為多層分散式方案（Multi-Tier Distributed Solution）中的本機使用者介面。

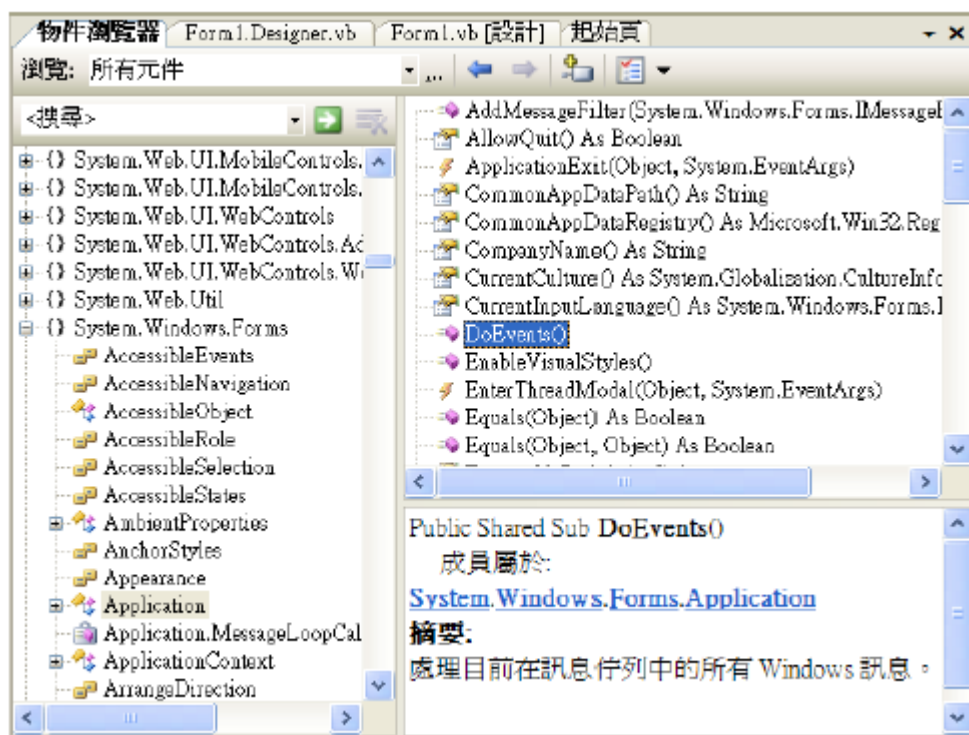


圖 2-15、類別物件瀏覽器

6. XML Web Service：使用 XML 透過 HTTP 接收要求和資料的應用程式。XML Web Services 不會與特定元件技術或物件呼叫慣例繫結，因此可由語言、元件模型或作業系統存取。在 Visual Studio 中，可以使用 Visual Basic、Visual C#、JScript 或 ATL Server，快速建立及包含 XML Web Services[16]。



圖 2-16：遠端伺服器呼叫

(三) 小結

表 2-1 為各種開發系統功能的比較，從資料表中我們可以看出，在各種功能之中 Visual Studio 2005 的功能與環境都優於其他系統，並且 Express Edition 這個版本目前為免費，功能也足以開發一個有規模的系統。

表 2-1、各種開發系統功能比較表

特色	表示產品	Visual Studio 標準版	Visual Studio 專業版	辦公室的 Visual Studio 工具	Visual studio 小組系統
IntelliSense	Yes	Yes	Yes	Yes	Yes
代碼編輯器	Yes	Yes	Yes	Yes	Yes
代碼片斷	Yes	Yes	Yes	Yes	Yes
程式設計語言適用	VB, VC#, VC++, and VJ# are single language. Visual Web Developer includes VC# and VB	All	All	VB and VC#	All
辦公室發展支援	No	No	No	Support for Excel 2003, Word 2003, InfoPath 2003 and Outlook 2003	Support for Excel 2003, Word 2003, InfoPath 2003 and Outlook 2003
用戶經驗	Streamlined	Streamlined	Full	Full	Full

Windows 形成設計者	VB, VC#, VC++, VJ#	Yes	Yes	Yes	Yes
Web 形成設計者	Visual Web Developer	Yes	Yes	Yes	Yes
移動設備支援	No	Yes	Yes	No	Yes
數據庫設計工具 “數據連接”節點	Local	All	All	All	All
數據存取	VB, VC#, VC++, VJ#:local, Visual Web Developer: local and remote	All	All	All	All
文檔編製	10mb "Getting Started"; Starter Kits targeted at first-time programmers; 200mb optional MSDN Express	MSDN	MSDN	MSDN	MSDN
種類設計者/	No	Yes	Yes	Yes	Yes

對象測試台					
XML 支援	XML only	XML/XSLT	XML/XSLT	XML/XSLT	XML/XSLT
部署工具	No	Click Once	Yes	Yes	Yes
可伸展性	Add external tools to the menu only. Use 3rd party controls.	Consume extensions	All	All	All
報告	SQL Server Reporting Services	SQL Server Reporting Services	SQL Server Reporting Services / Crystal Reports	SQL Server Reporting Services / Crystal Reports	SQL Server Reporting Services / Crystal Reports
來源代碼控制	No	MSSCCI-compatible (Visual SourceSafe sold separately)	MSSCCI-compatible (Visual SourceSafe sold separately)	MSSCCI-compatible (Visual SourceSafe sold separately)	MSSCCI-compatible (includes Visual SourceSafe, Visual Studio Team Foundation Server sold separately)
除錯	Local	Local	Local / remote	Local	Local / remote
64 位元的編譯器支援	No	No	Yes	No	Yes
服務器資源管理器 “服務器”節點	No	No	All	All	All
SQL 2005 伺服器整合	No	No	Yes	Yes	Yes
代碼素描	No	No	No	No	Yes
靜態分析	No	No	No	No	Yes

單元測試	No	No	No	No	Yes
代碼覆蓋率	No	No	No	No	Yes
項目管理	No	No	No	No	Yes
測試個案管理	No	No	No	No	Yes
尺寸	80mb (Express + SQL Express + .NET Framework Redist)	Single CD	Multiple CDs	Multiple CDs	Multiple CDs
附加的工具包括	None	None	SQL Server 2005 Developer Edition	Windows 2003 Server Developer Edition; SQL Server 2005 Developer Edition; Microsoft Office Access 2003 Developer Extensions; Access 2003 Runtime license	SQL Server 2005 Developer Edition (included with client products only)
特徵	表示產品	Visual Studio 標準版	Visual Studio 專業版	辦公室的 Visual Studio 工具	Visual studio 小組系統

第三章 電子化畢業資格審查系統

第一節 系統架構

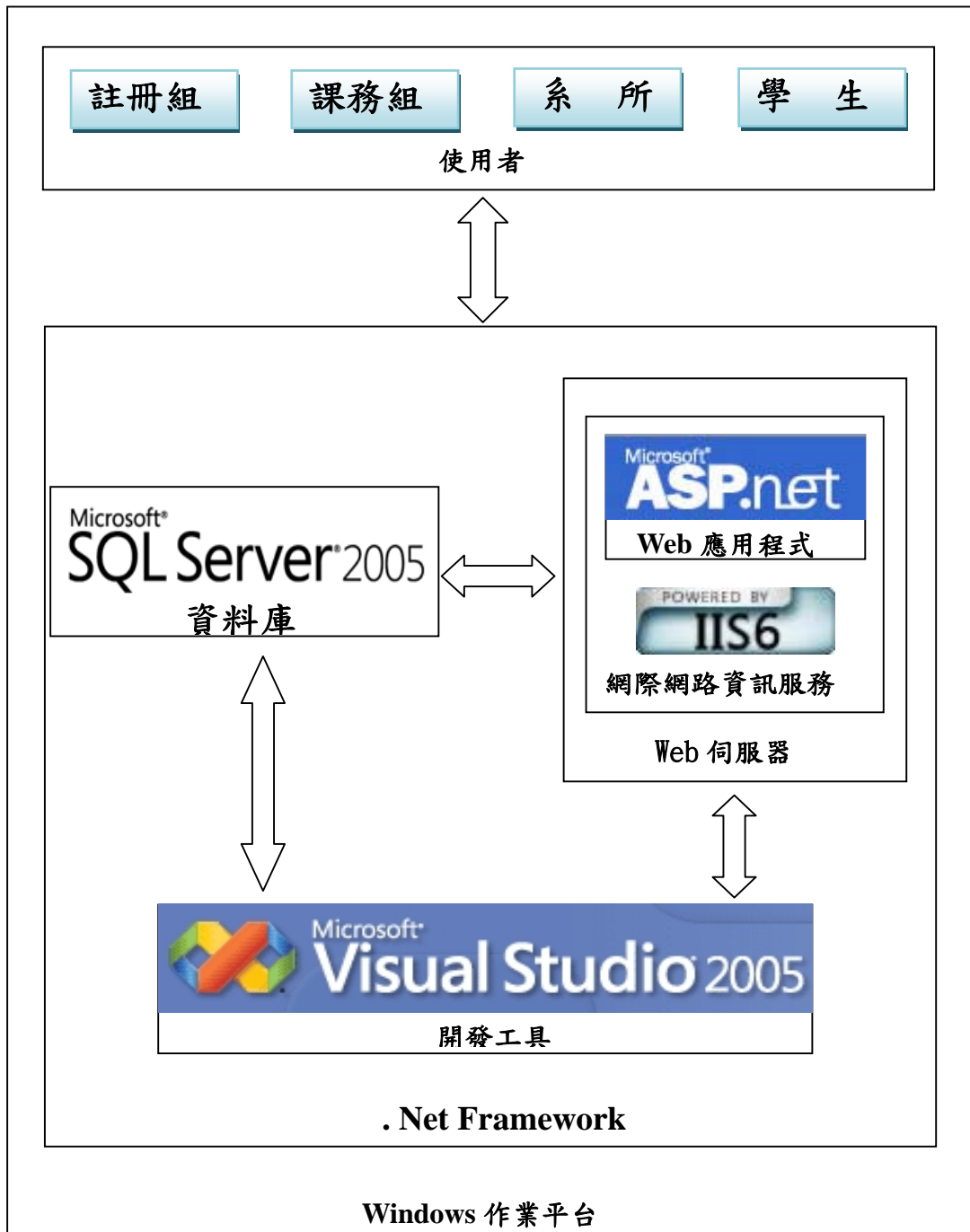


圖 3-1、電子化畢業資格審查系統架構圖

本系統是架設在 Windows 作業平台上(Windows Server 2003)，於作業系統上安裝 .NET Framework，將環境升級成 .NET 平台，整個系統執行於 .NET 平台上。開發工具為 Visual Studio 2005，使用 ASP.NET 網站應用程式，利用 Visual Basic 2005 語法進行開發，Web 伺服器使用 IIS 6.0 (Internet Information Services 6.0)，所使用的資料庫為 SQL Server 2005 資料庫，詳見圖 3-1 所示。

「.NET 平台」在經過幾年的發展後，目前已經是業界的主流平台，於財星(Fortune)前 100 大的企業中，已超過 96% 企業採用 Microsoft.Net 技術，故本專題使用「Microsoft ASP.NET」開發技術來建置電子化畢業資格審查系統。「Microsoft ASP.NET」開發技術，由微軟公司所開發，不但功能彈性大、且可以完全走網路平台，對於系統的使用年限，也可以長達許久。

Visual Studio 2005 便是微軟公司為 .NET 程式設計者所提供的 IDE(Integrated Development Environment) 軟體，支援多種程式語言、具備統一的開發介面，不論開發桌面應用程式、ASP.NET 網頁程式、XML Web Service 或是行動裝置應用程式，都可以在相同的環境與介面下完成，系統提供的功能，詳見下表：

表 3-1、使用者功能列表

註冊組	畢業資格審查、畢業資格審查狀態查詢、應修科目表查詢、學生歷年修課狀態查詢、學生成績查詢
課務組	應修科目表審核、應修科目表維護、應修科目表查詢、學生歷年修課狀態查詢、學生成績查詢
系所(科)	畢業門檻建立、應修科目表建立、應修科目表查詢、畢業資格審查、畢業資格審查狀態查詢、學生歷年修課狀態查詢、學生成績查詢
學生	畢業資格審查狀態查詢、畢業資格審查狀態確認、學生歷年修課狀態查詢、應修科目表查詢、學生成績查詢

關於整個系統使用流程，詳見圖 3-2，說明如下所示：

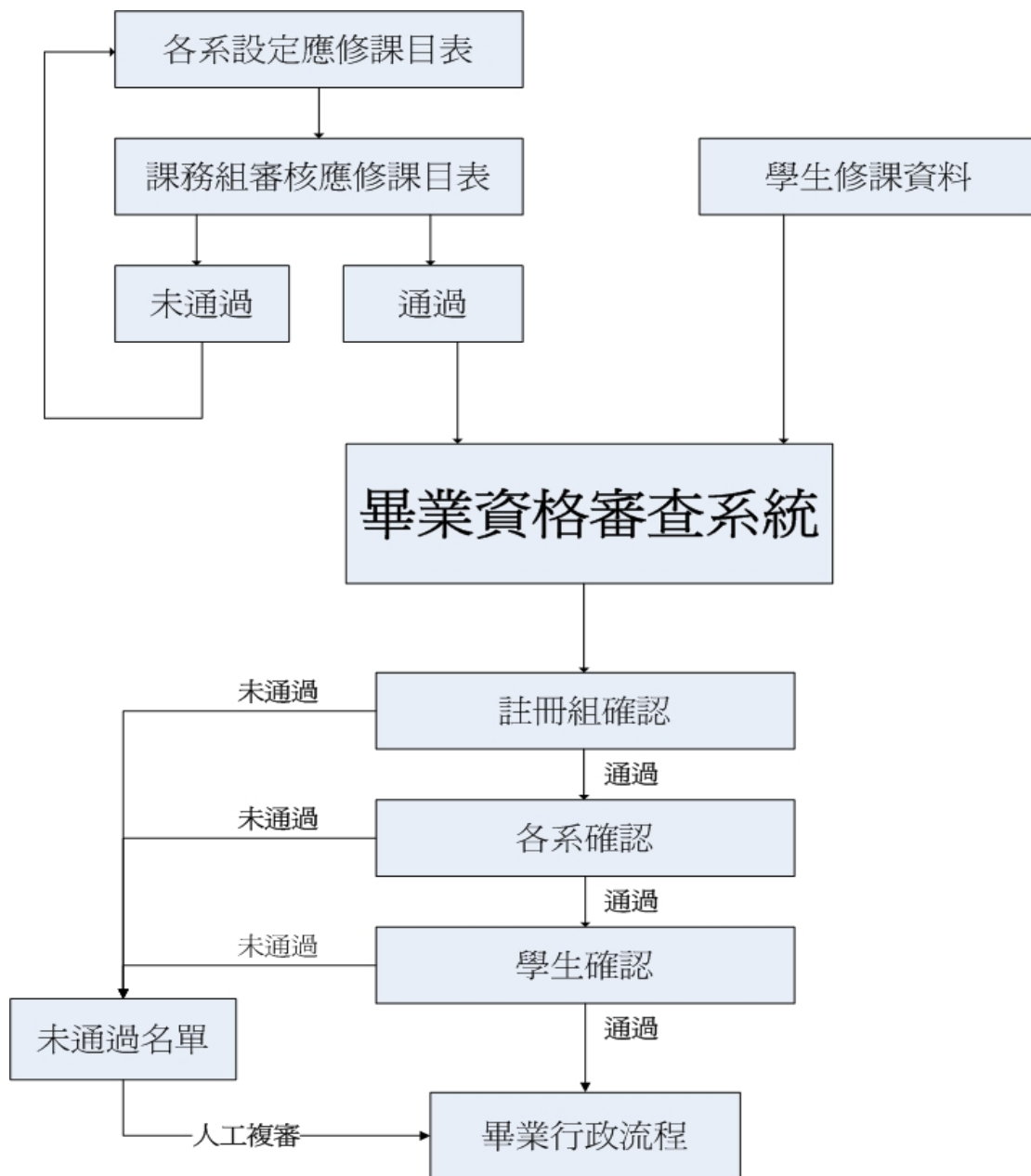


圖 3-2、畢業資格審查系統流程圖

各系設定應修課目表：

於新生入學前，各系科上網登入系統，填入應修課目表，由系統送交課務組審查。

課務組審核應修科目表：

課務組收到各系科繳交之應修科目表後，進行審核：

- 通過：進行建檔，以便未來依照應修科目表進行開課。
- 未通過：通知各系科，修改應修科目表後再次審核。

學生修課資料：

學生從入學至畢業所有修過的必選修科目資料。

畢業資格審查系統：

- 本系統於每學期結束時都會結算當期學生的畢業資格。將已通過審核並建檔的應修科目表與學生修課資料進行比對，即時性的結算出學生當下所擁有之學分數與未修得之學分數，若有必修科目未通過或是本學期學分未達標準（二一或三二），皆會反應予學生及各系科，除了學生可以於學校的信箱收到通知，也會另由系科通知學生。
- 當最後一個學期的審核結束後，不論學生有無通過畢業資格審查，皆會發佈通知給予學生、各系科及註冊組。

學生確認：

除了一般時期學生皆可以上網去查詢自己的畢業資格，當最後一學期的審核結束後，學生會收到最終確認的資料，自行比對後如果無誤，須於網路上回報正確；若是有問題，則回報錯誤，正確與錯誤訊息皆會送回各系科予以統計整合。

各系確認：

各系科擁有查閱本系科所有學生之畢業資格的權限。各系科於最後一學期的審核結束後，會收到系科所有學生的畢業資格表，各系科必須督促學生上網去自行核對審核資料是否正確或錯誤並回報，等各系科全部學生回報後，一併經由各系科將結果送交註冊組。

註冊組確認：

註冊組擁有查閱與修改全校學生之畢業資格的權限。當學生回報完畢後，由各系各將回報結果送回註冊組，正確的學生則進入後續畢業流程，回報錯誤之學生，將由註冊組進行人工二次審核。

畢業行政流程：

審核通過的學生，再經過教務長審核、校長核定，即可核發畢業證書，最後再將畢業生名冊建檔並永久保存。

第二節 系統功能簡介

電子化畢業資格審查系統，主要的使用者有「註冊組」、「課務組」、「系所(科)」以及「學生」，各使用者所擁有的功能說明如下：

(一) 註冊組

1. 畢業資格審查：針對全校應屆畢業生做畢業資格審核的動作，分為「校訂必修科目審核」及「校訂選修科目審核」審核。
 - 校訂必修科目審核：針對校訂必修科目，如國文、英文…等做審核之動作。
 - 校訂選修科目審核：針對通識課程及核心通識課程做審核之動作。
2. 畢業資格審查狀態查詢：查詢全校應屆畢業生畢業資格審核情形，包括未通過及已通過審核名單。
3. 應修科目表查詢：查詢各學制系所(科)之應修科目表。
4. 學生歷年修課狀態查詢：查詢全校應屆畢業生目前修課狀態，例如：必選修學分數、距畢業資格所欠缺條件等。
5. 學生成績查詢：查詢全校學生歷年修課成績。

(二) 課務組

1. 應修科目表審核：審核各學制系所(科)所建立之應修科目表之正確性。
2. 應修科目表維護：維護各學制系所(科)之應修科目表，可進行應修科目表之新增、刪除、修改等動作。
3. 應修科目表查詢：查詢各學制系所(科)之應修科目表。
4. 學生歷年修課狀態查詢：查詢全校應屆畢業生目前修課狀態，例如：必選修學分數、距畢業資格所欠缺條件等。
5. 學生成績查詢：查詢全校學生歷年修課成績。

(三) 系所(科)

1. 畢業門檻建立：建立系所(科)畢業所需之條件與門檻。
2. 應修科目表建立：建立系所(科)之應修科目表。
3. 應修科目表查詢：查詢系所(科)歷年應修科目表。
4. 畢業資格審查：針對系所(科)應屆畢業生做畢業資格審核的動作，主要審核系訂必選修科目以及系所(科)畢業門檻。
5. 畢業資格審查狀態查詢：查詢系所(科)應屆畢業生畢業資格審核情形，包括未通過及已通過審核名單。
6. 學生歷年修課狀態查詢：查詢系所(科)應屆畢業生目前修課狀態，例如：必選修學分數、距畢業資格所欠缺條件等。
7. 學生成績查詢：查詢系所(科)學生歷年修課成績。

(四) 學生

1. 畢業資格審查狀態查詢：查詢學生本人畢業資格審核情形。
2. 畢業資格審查狀態確認：確認學生本人畢業資格審查結果無誤。
3. 學生歷年修課狀態查詢：查詢學生本人目前修課狀態，例如：必選修學分數、距畢業資格所欠缺條件等。
4. 應修科目表查詢：查詢學生所屬系所(科)之歷年應修科目表。
5. 學生成績查詢：查詢學生本人歷年修課成績。

第三節 系統操作流程

(一) 安裝 Visual Studio 2005 套件

本節介紹安裝 Visual Studio 2005 套件的實做流程：



圖 3-3、安裝 Visual Studio 2005 畫面



圖 3-4、Visual Stidio 2005 安裝程式載入安裝元件



圖 3-5、完成 Visual Stidio 2005 安裝程式載入安裝元件



圖 3-6、使用者同意書以及輸入序號、使用者名稱

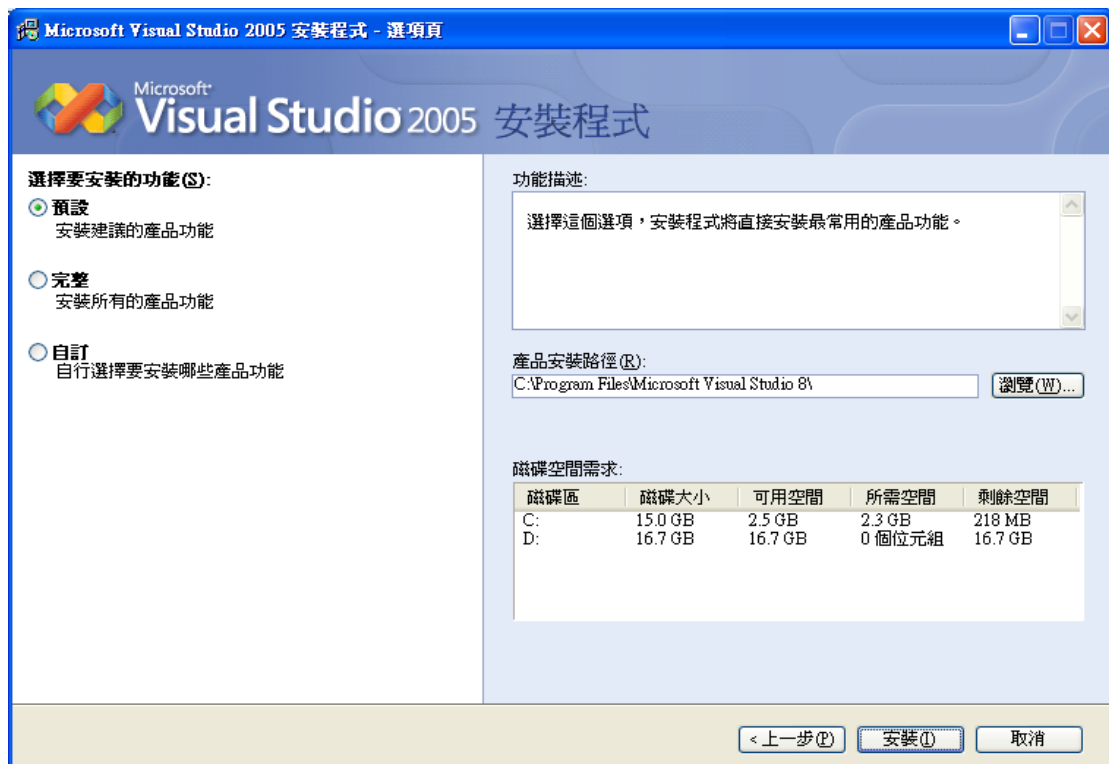


圖 3-7、選擇安裝的功能選項



圖 3-8、選擇安裝功能的自訂選項，安裝所需要的程式內容



圖 3-9、所需選項打勾

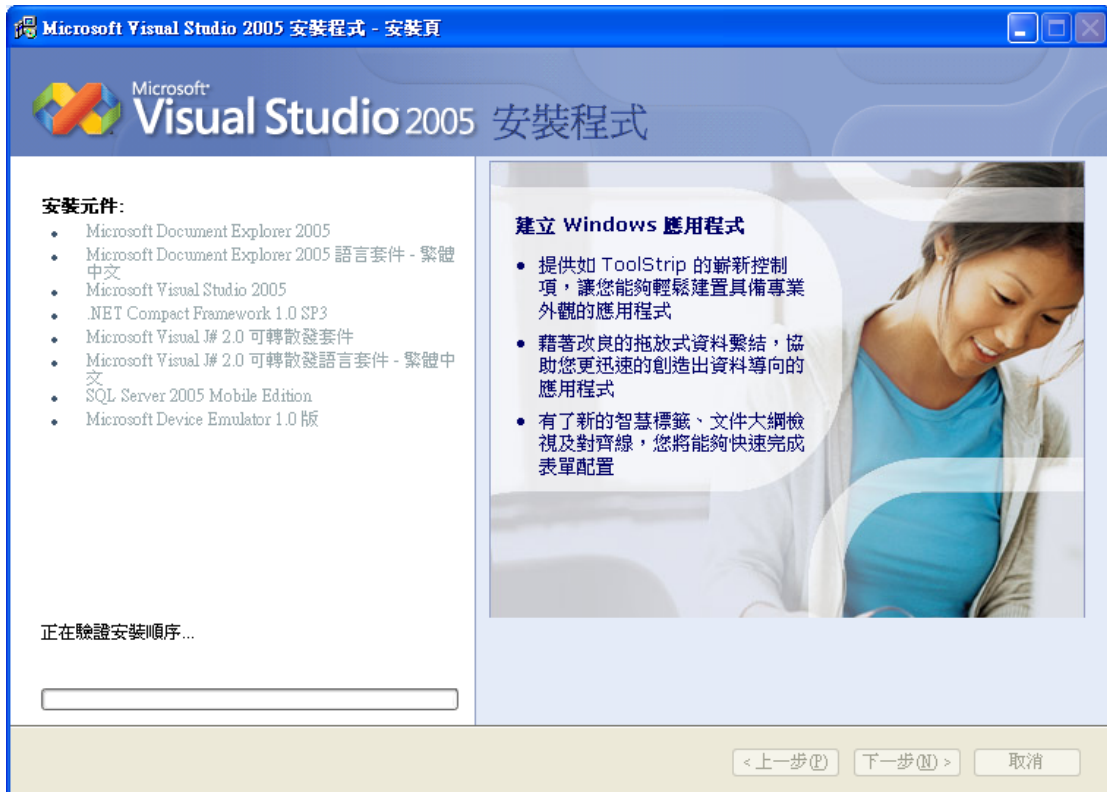


圖 3-10、安裝程式開始安裝



圖 3-11、安裝 Document Explorer 2005



圖 3-12、安裝 Visual Studio 2005



圖 3-13、安裝 .NET Compact Framework 1.2 sp3



圖 3-14、安裝 Microsoft Visual J#可轉發套件



圖 3-15、安裝 SQL Server 2005 Mobile Edition

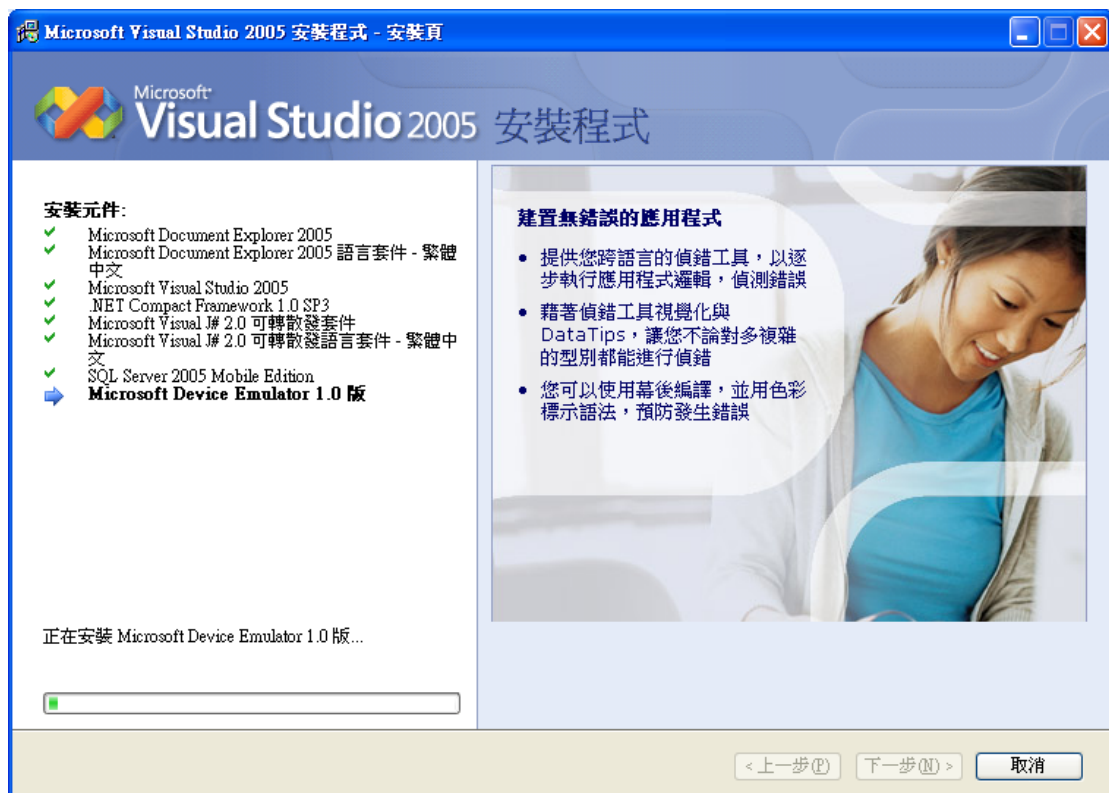


圖 3-16、安裝 Microsoft Device Emulator 1.0 版



圖 3-17、Visual Studio 安裝完成

(二) 安裝 SQL Server 2005

接著介紹安裝 SQL Server 2005 的實作流程：



圖 3-18、打開 SQL Sevrver 2005 組態管理員

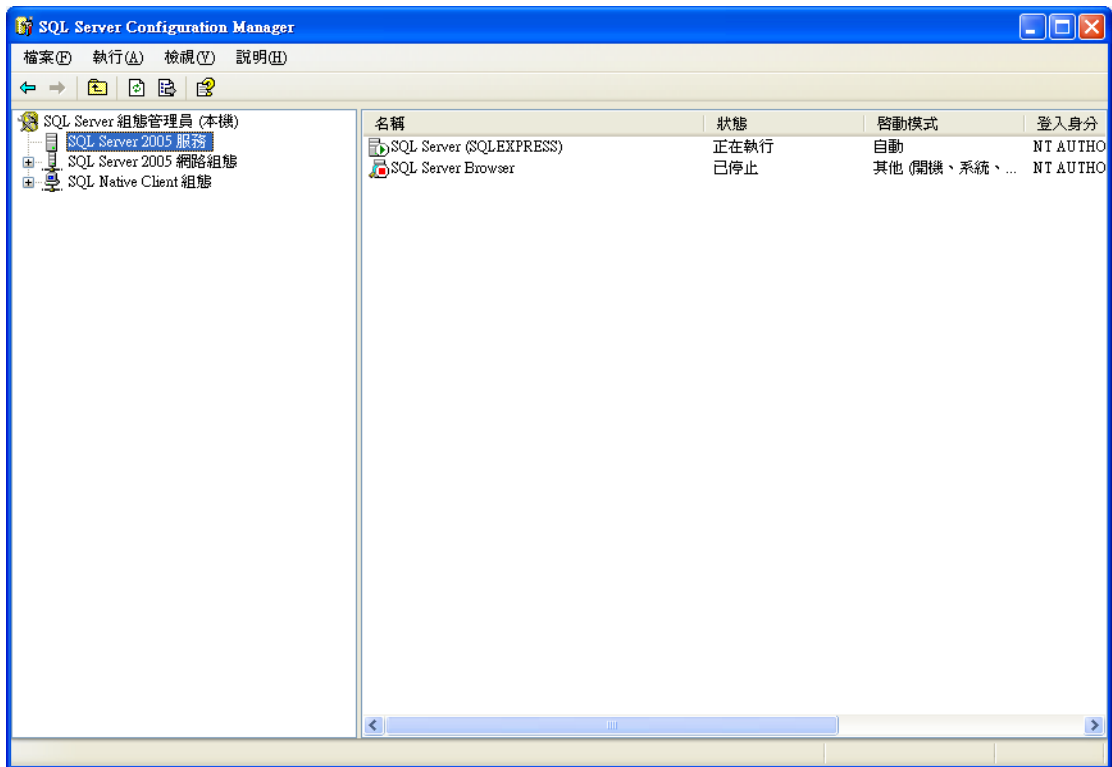


圖 3-19、選擇 SQL Sevrver 服務

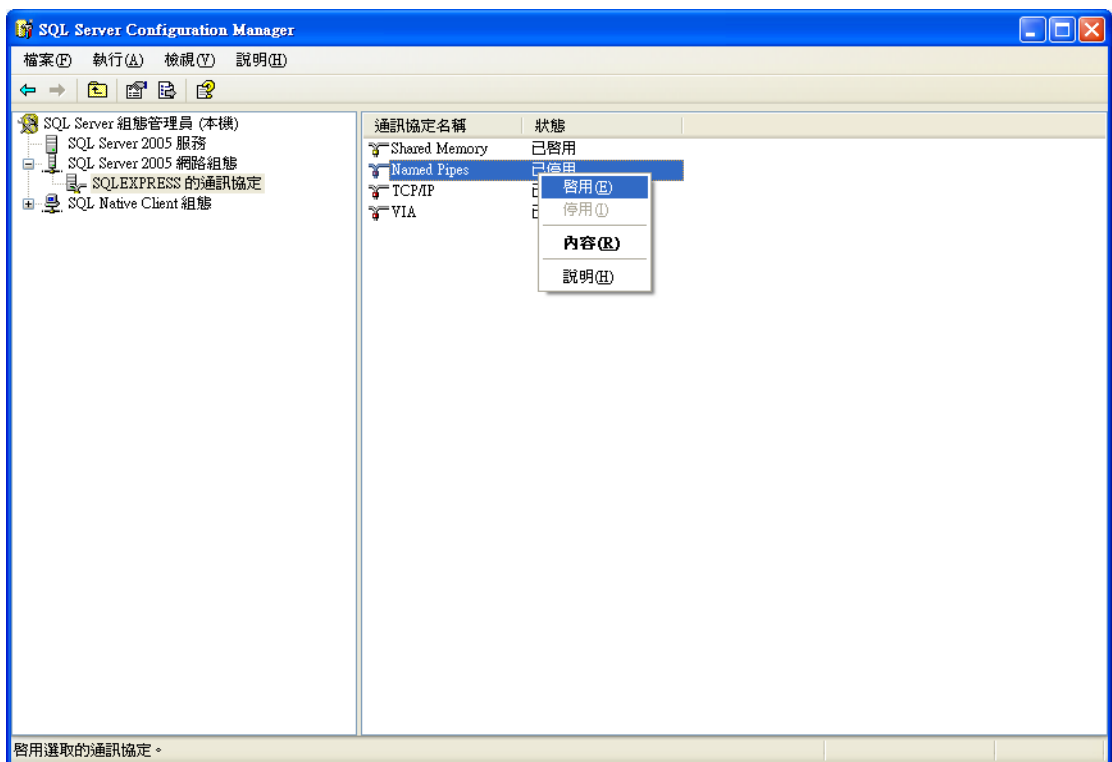


圖 3-20、啟用 Names Pipes 服務

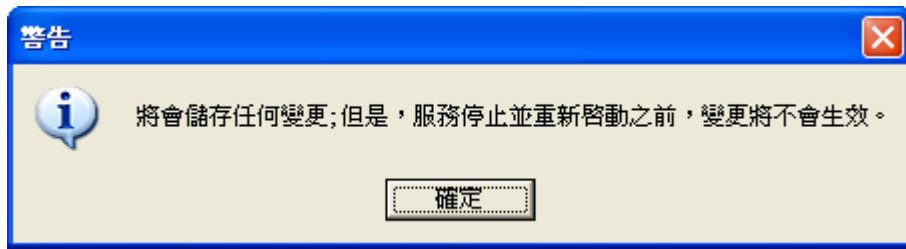


圖 3-21、系統提示

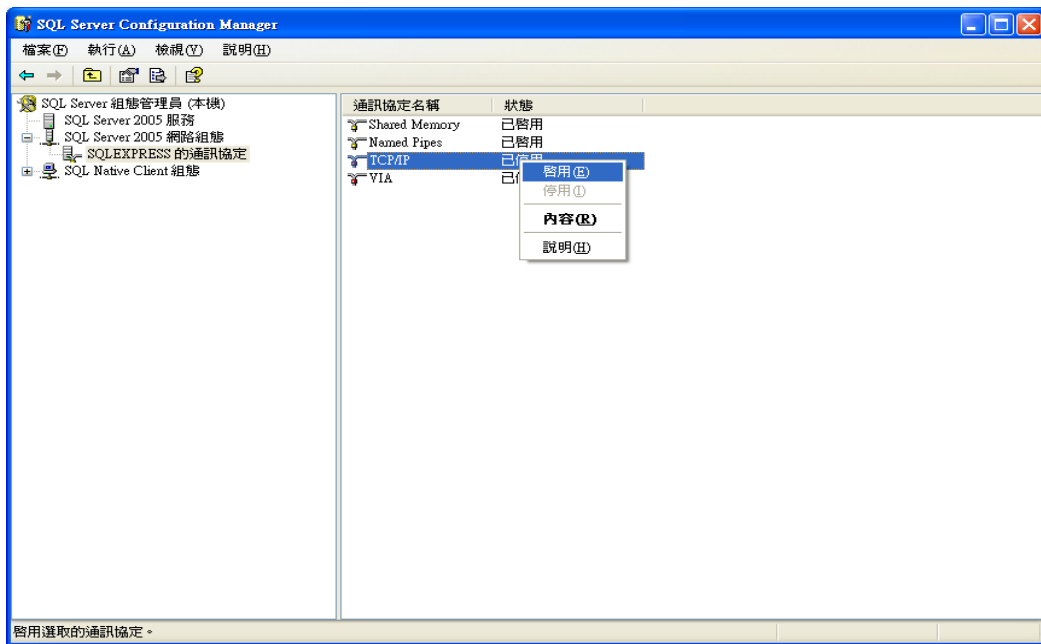


圖 3-22、啟用 TCP/IP 服務

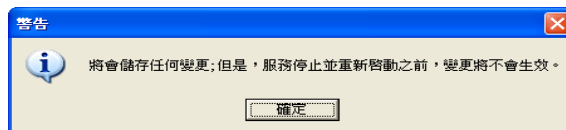


圖 3-23、電腦服務更動提示

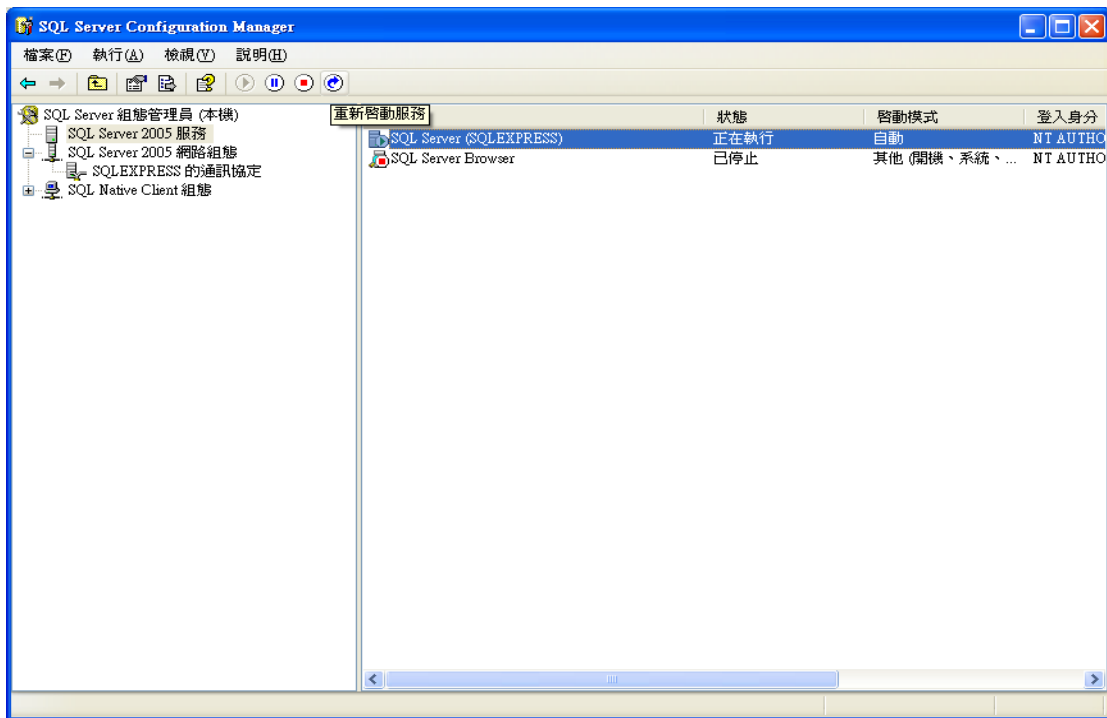


圖 3-24、重新啟動服務

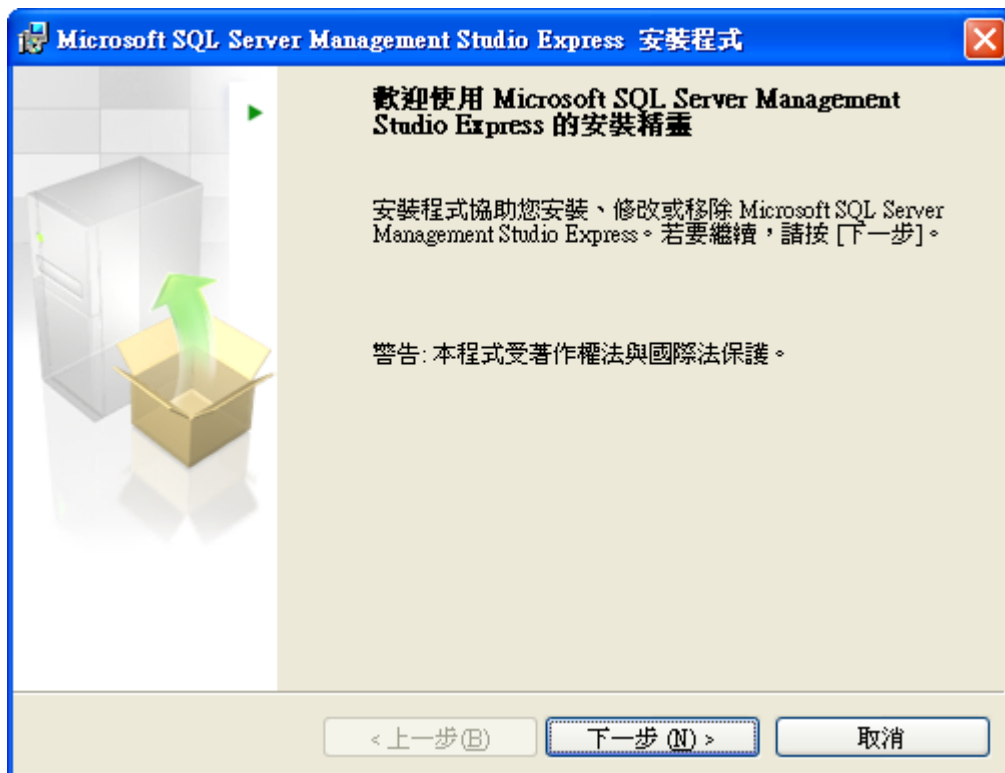


圖 3-25、開始安裝 SQL Server 2005



圖 3-26、SQL Server 2005 授權合約

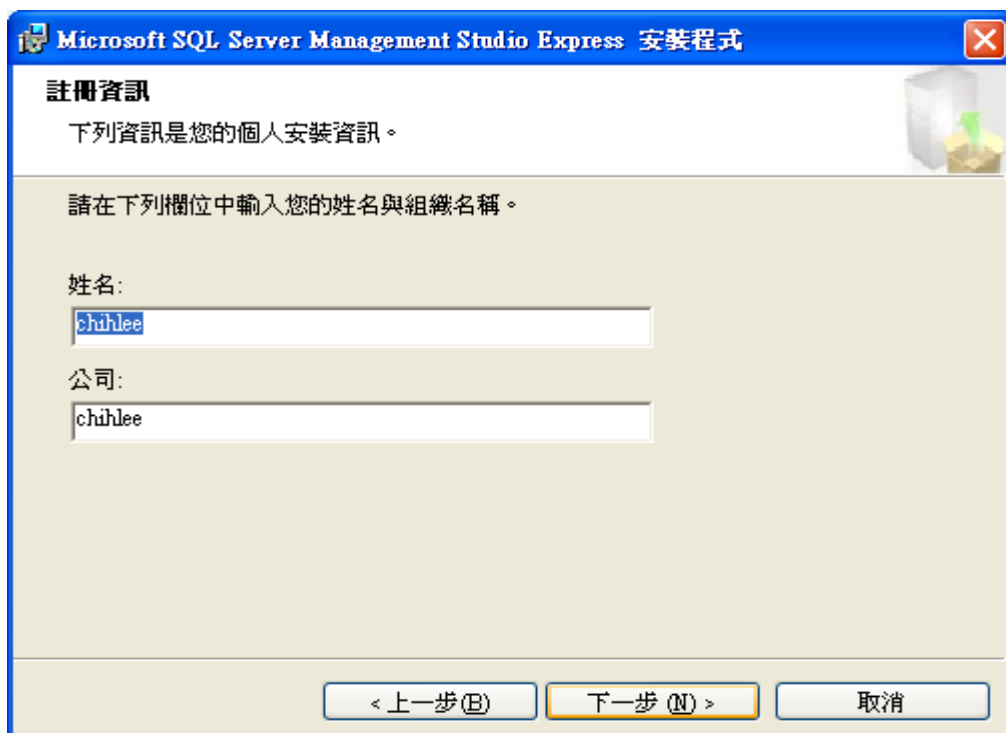


圖 3-27、輸入註冊資訊

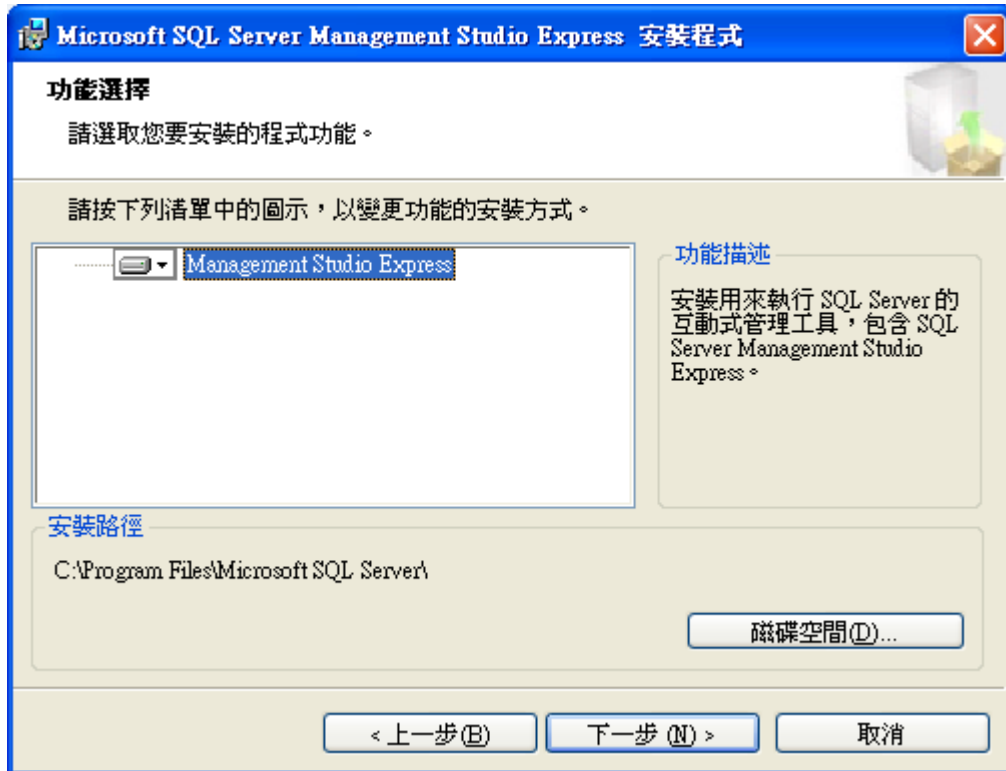


圖 3-28、安裝功能選擇

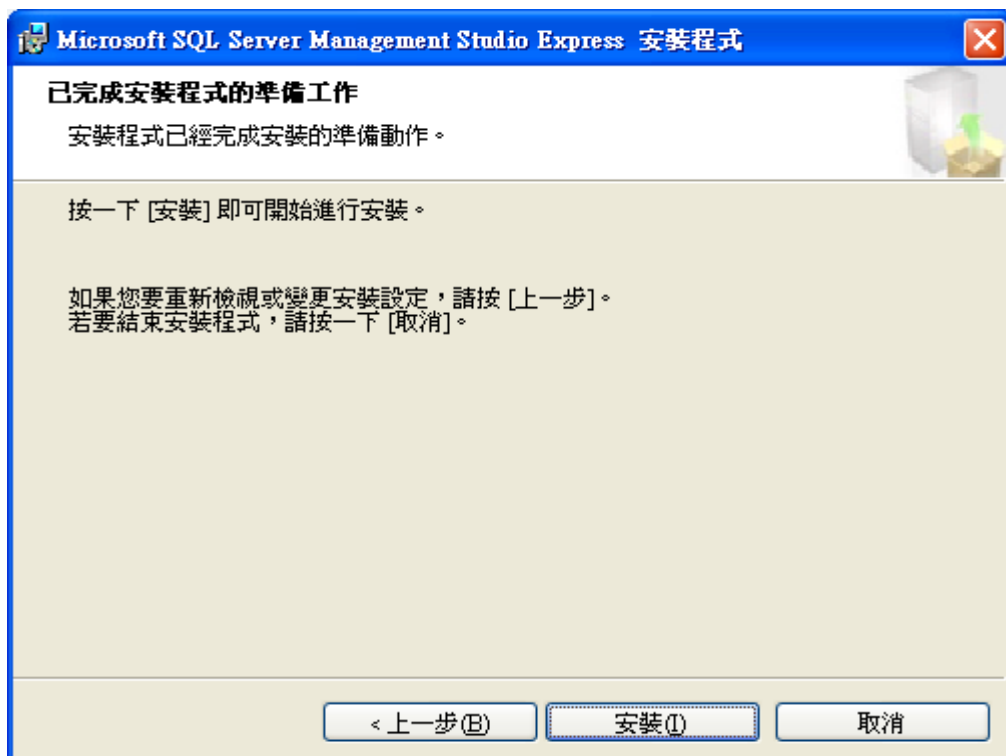


圖 3-29、安裝準備動作完成，開始安裝

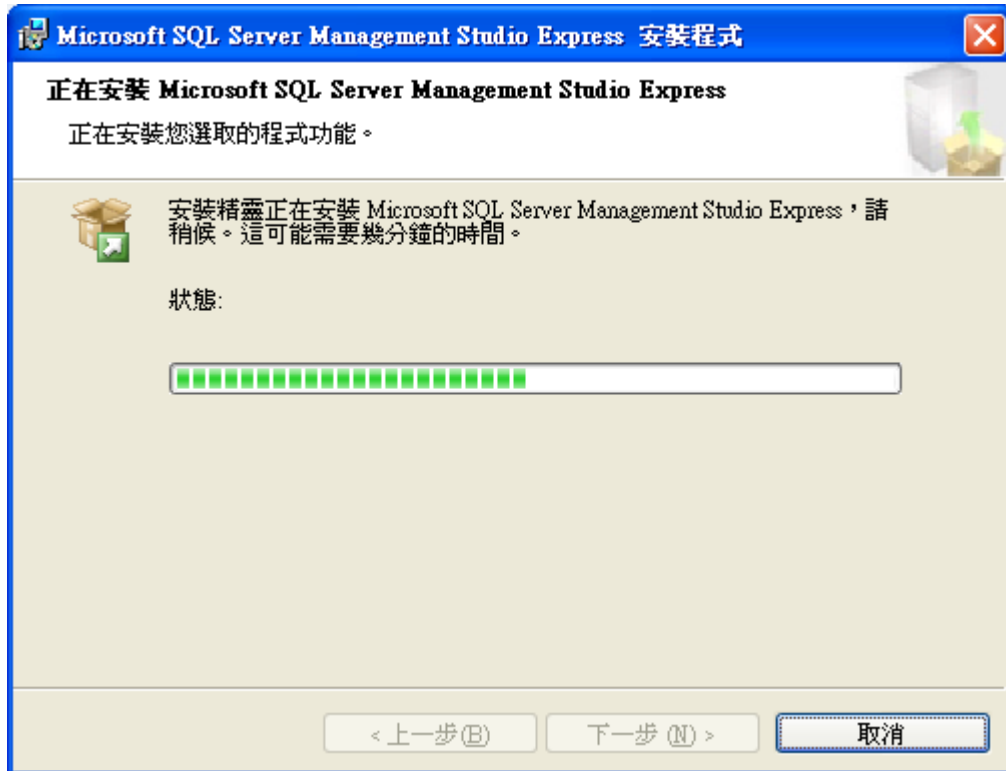


圖 3-30、安裝程序中



圖 3-31、安裝完成

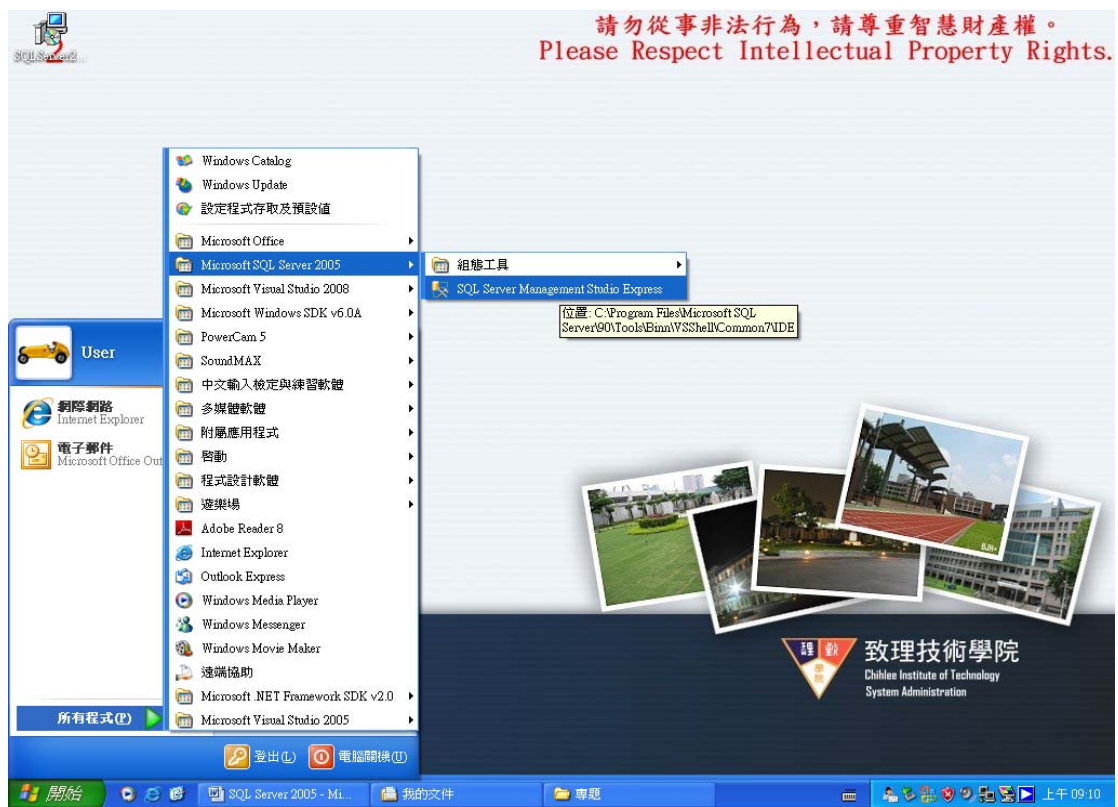


圖 3-32、啟動 SQL Server 2005



圖 3-33、選擇連接 SQL Server 2005 伺服器名稱

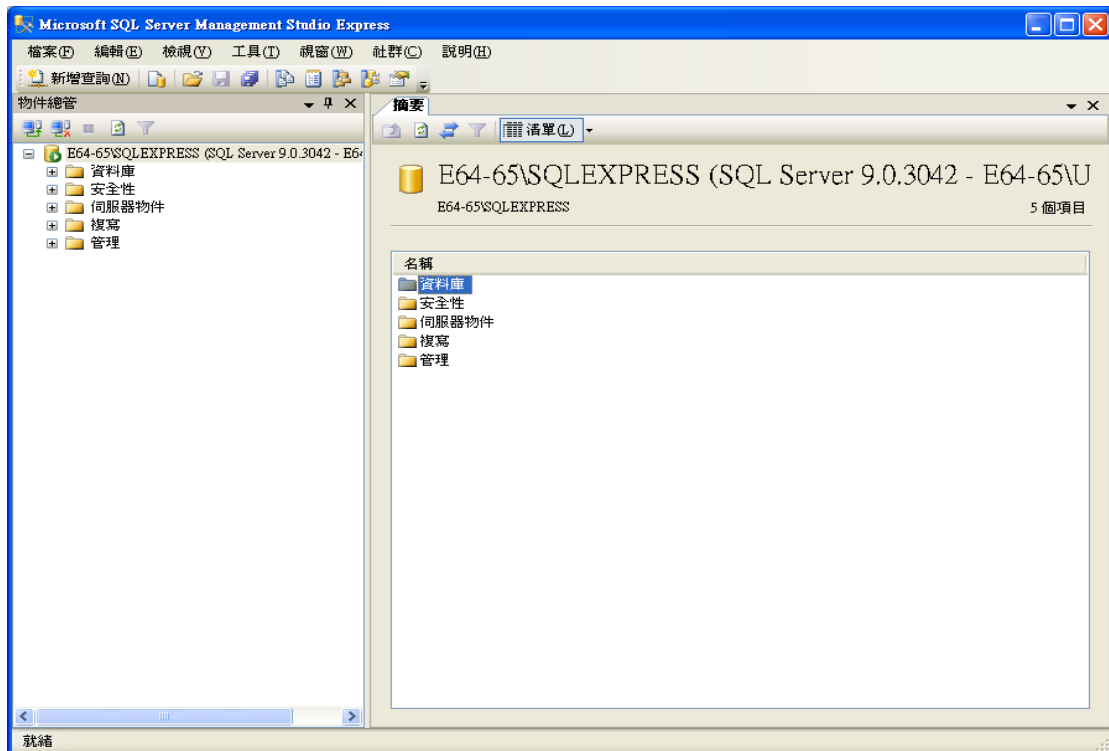


圖 3-34、開啟資料庫

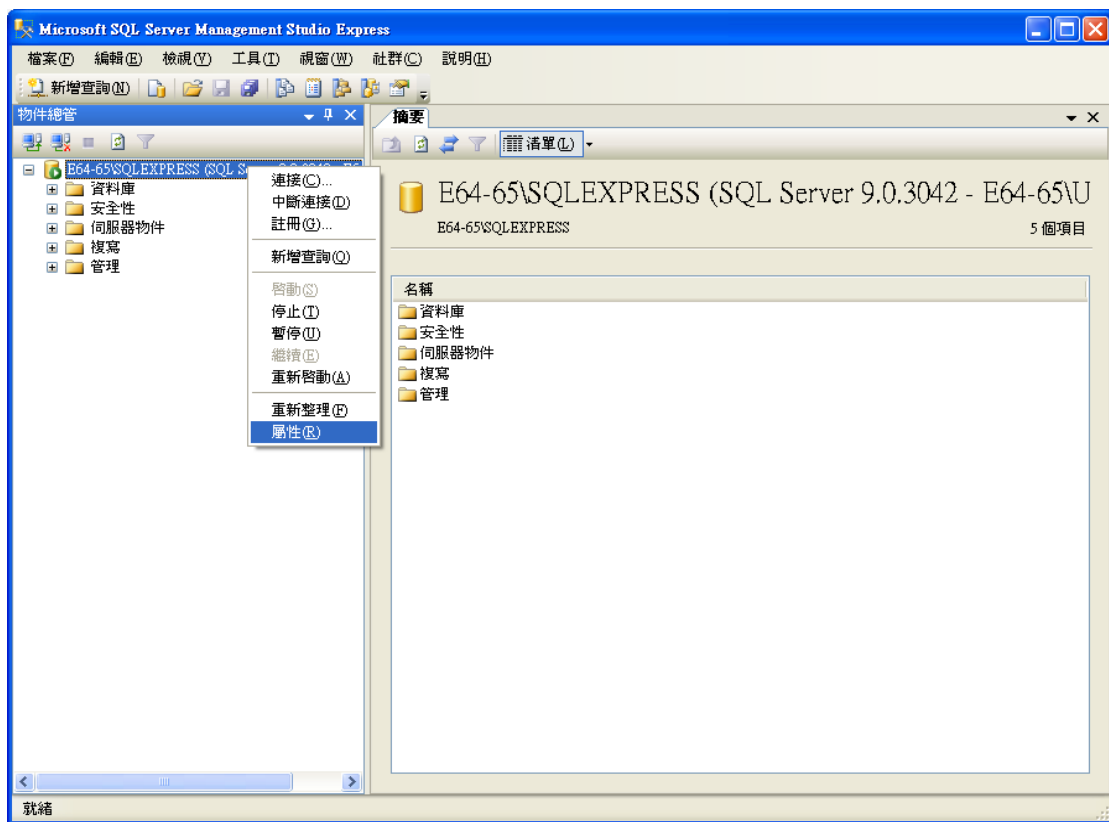


圖 3-35、點右鍵開啟資料庫內容屬性

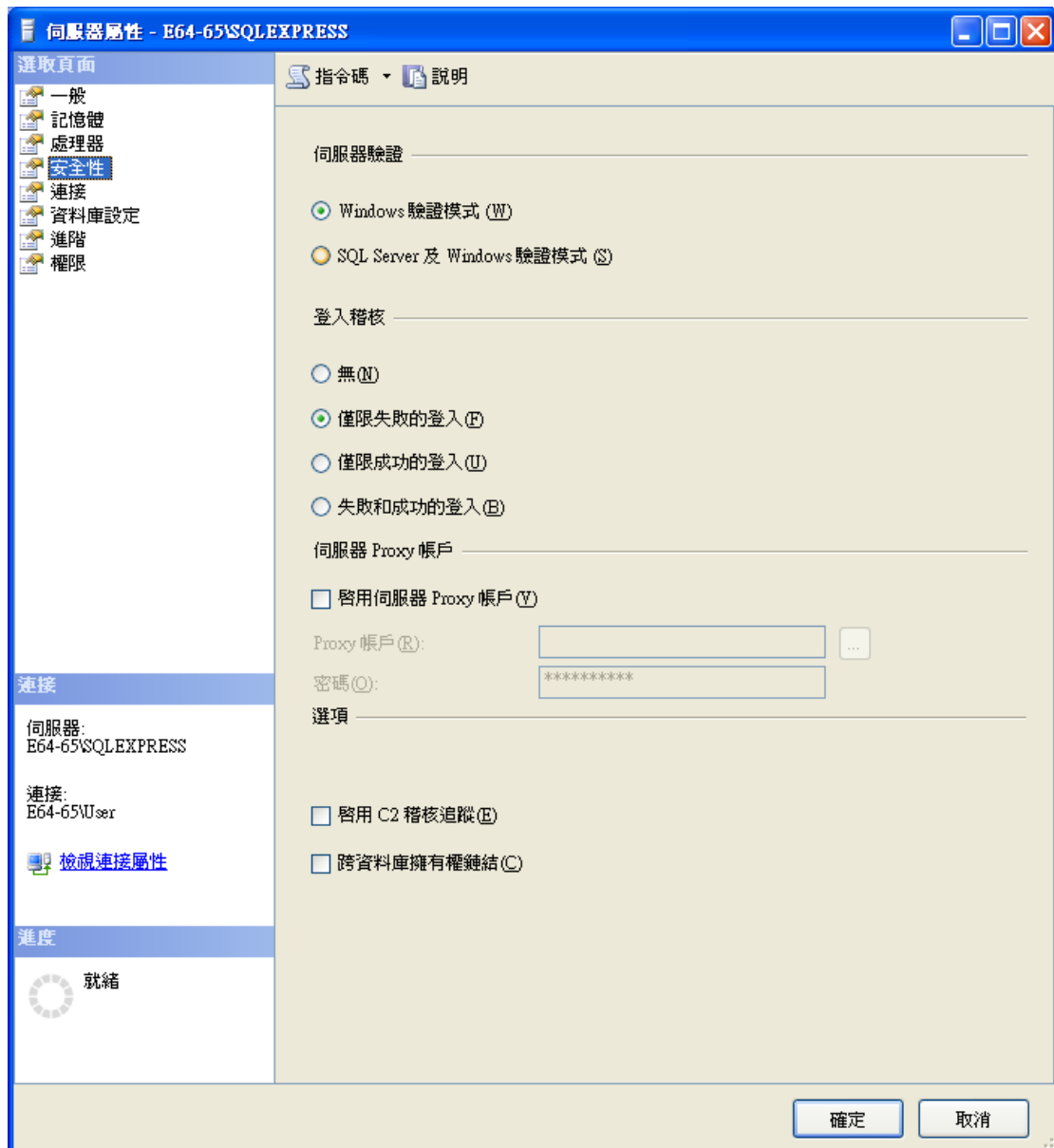


圖 3-36、進入伺服器屬性頁面，設定安全性

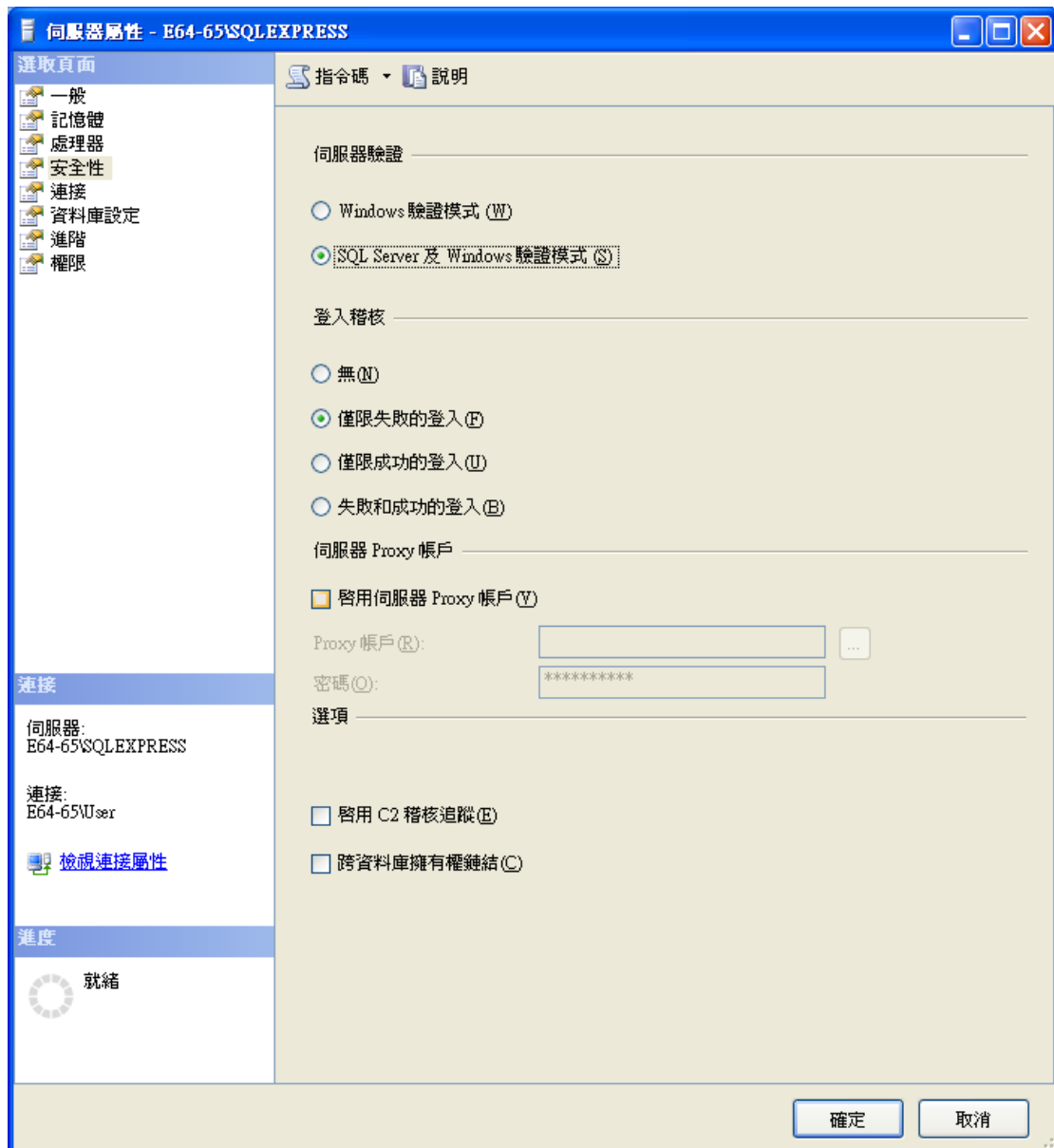


圖 3-37、伺服器驗證選擇 Server 以及 Windows。



圖 3-38、設定完成後，必須重新啟動 SQL Server 才能生效。



圖 3-39、點選 SQL Server 2005 組態工具後進入 SQL Server 組態管理員

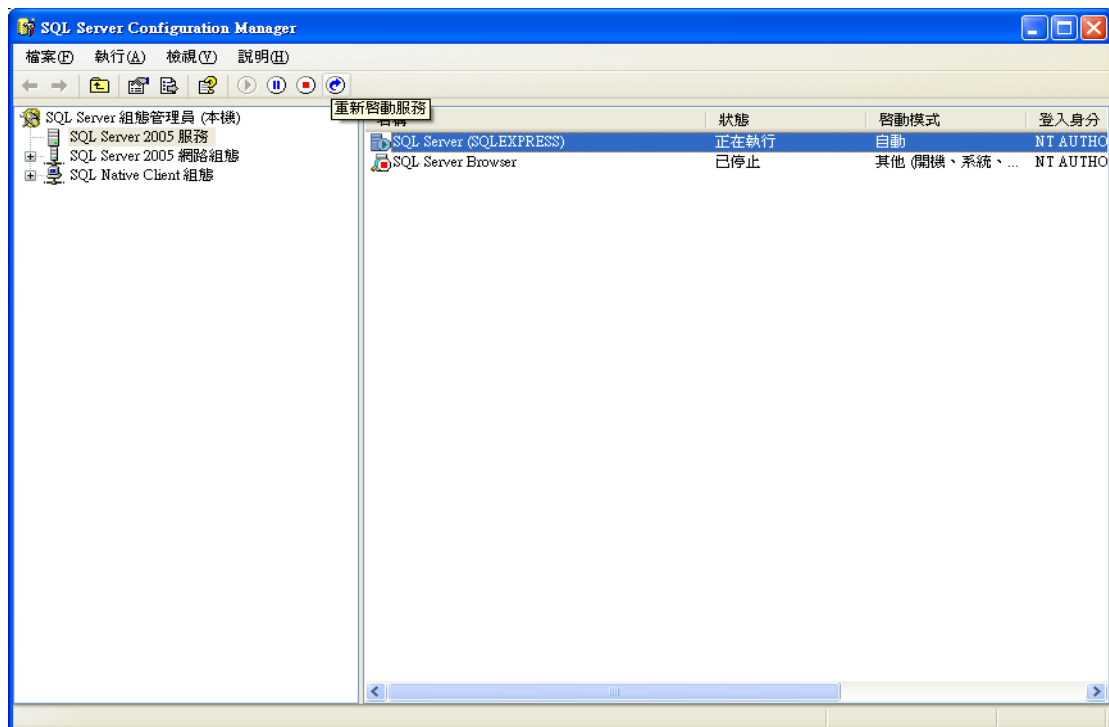


圖 3-40、按下重新啟動服務

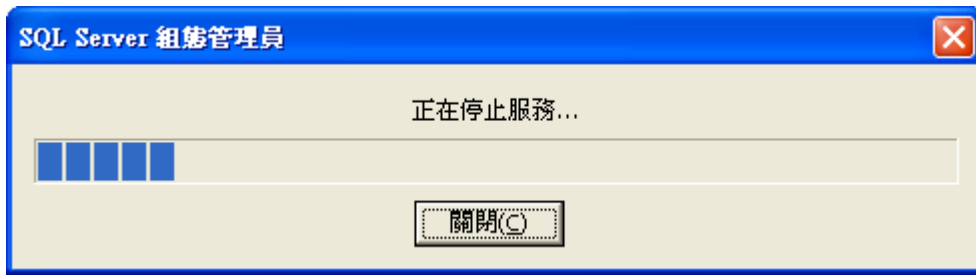


圖 3-41、系統提示停止服務

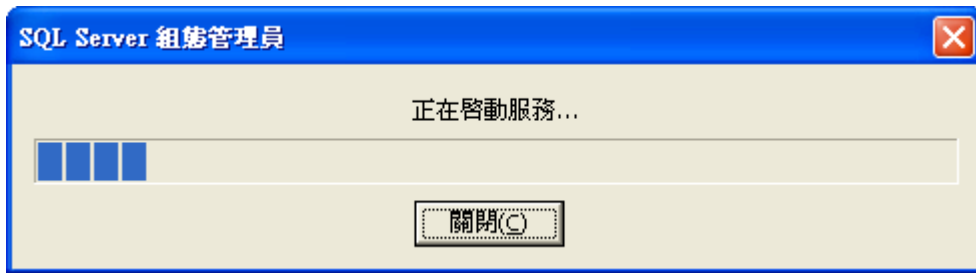


圖 3-42、系統提示重新啟動服務

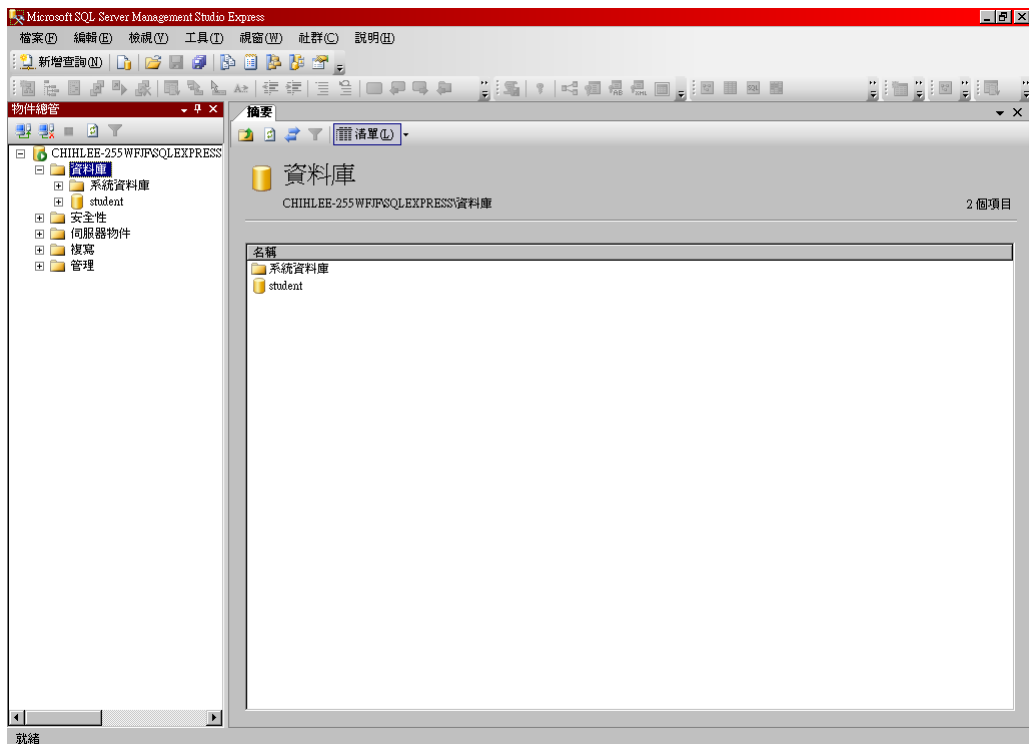


圖 3-43、開啟 SQL SERVER 2005

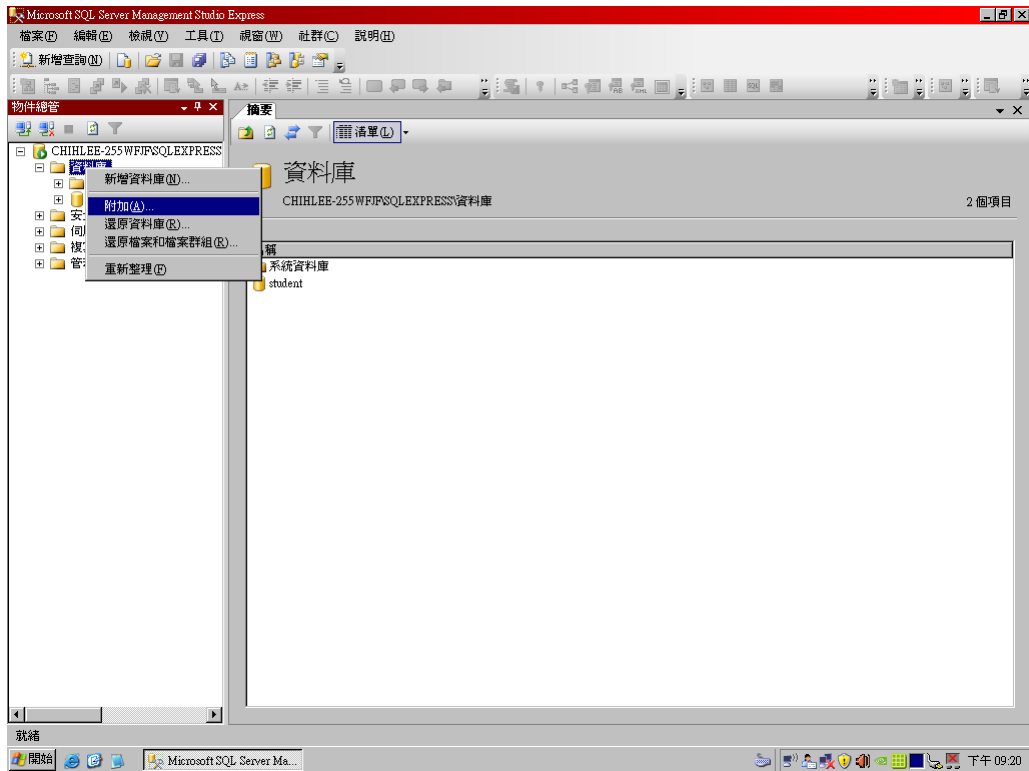


圖 3-44、於資料庫上點選右鍵，選擇附加

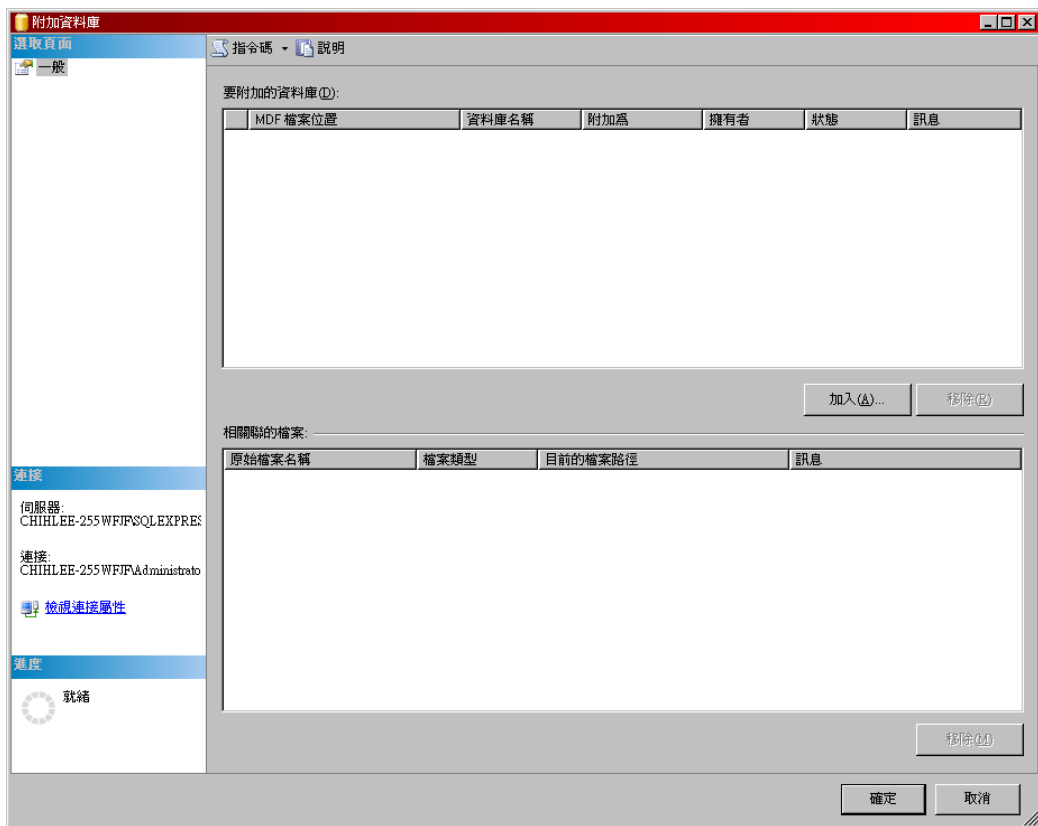


圖 3-45、點選加入

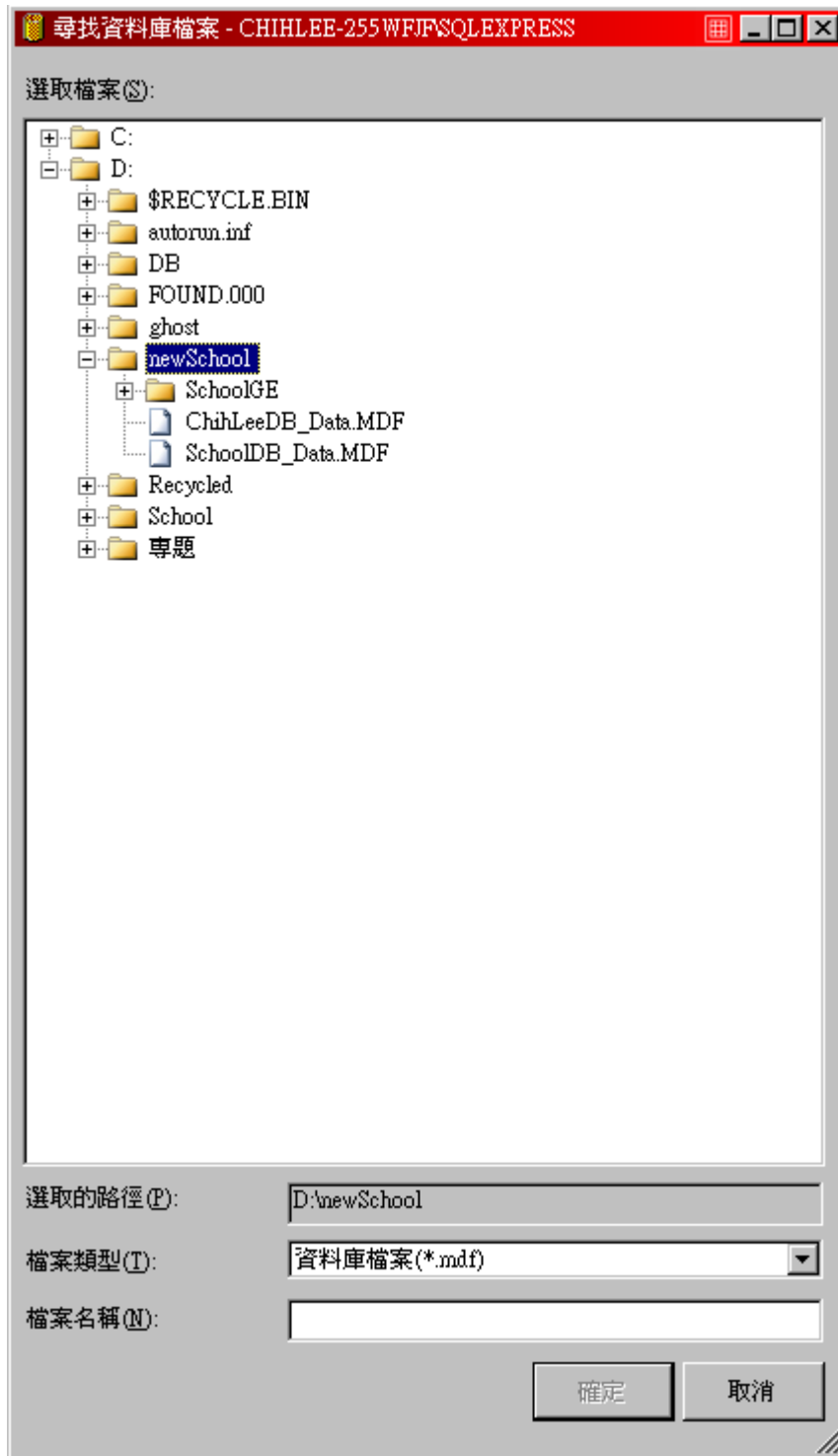


圖 3-46、選擇要附加之資料庫檔案

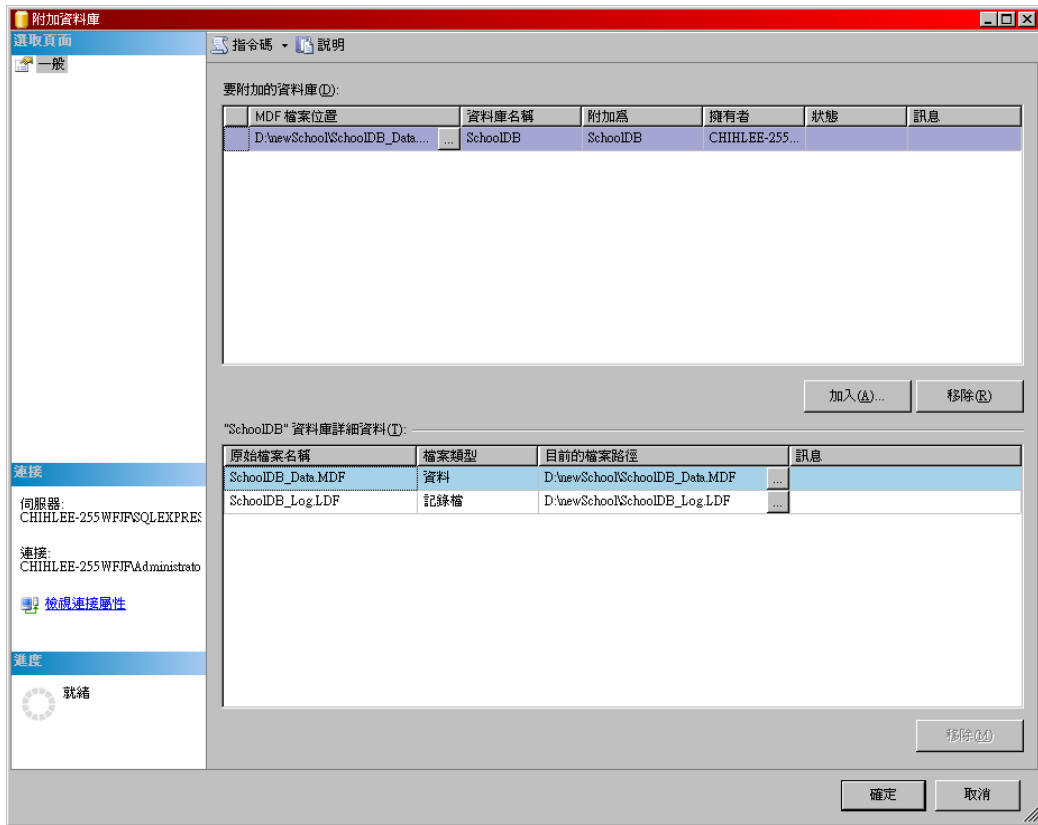


圖 3-47、附加完畢後，點選確定

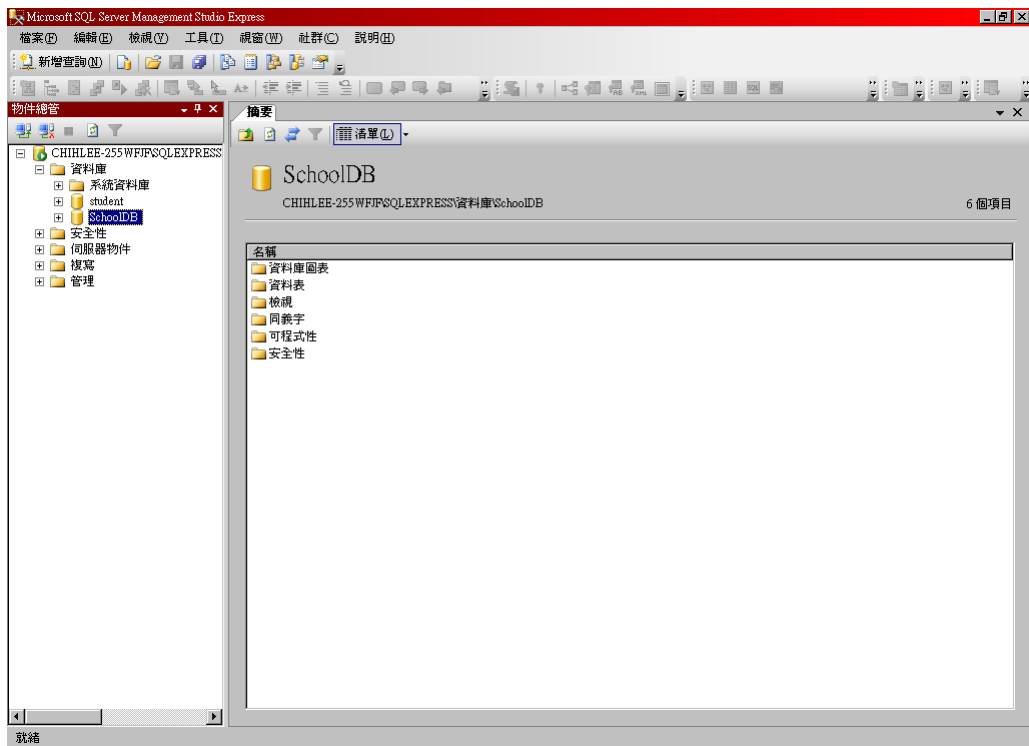


圖 3-48、確認附加資料庫成功，完成附加

第四節 系統特色

電子化畢業資格審查系統，最主要就是將繁瑣的畢業資格審查作業，透過電子化的方式，進行線上審核作業，相關的特色說明如下：

1. 電子化作業：使用網際網路進行線上一貫電子化作業，可大量減少人工審核作業所造成之耗時，降低人力耗用，藉由電子化審核作業，可提高資料正確性。
2. 線上即時審核：註冊組以及系所(科)可以透過電子化畢業資格審查系統，線上即時了解應屆畢業生之畢業資格審核狀態，並做適時地追蹤。學生也可以隨時上線了解自己畢業資格審核狀態，可以做為往後課程規劃的依據。
3. 符合節能需求：一般人工畢業資格審查，必須列印所有應屆畢業生歷年成績單以及各系所(科)之應修科目表和畢業門檻，再做人工審核動作，期間的處理過程須使用很多紙張文件往來行政單位之間，使用線上一貫電子化可以減少使用紙本文件，不但可節省實際成本更是符合環保。

第四章 系統呈現

第一節 預期效能

本專題研究之目的，是希望建構一個電子化畢業資格審核的系統，以協助學校畢業資格審核作業，簡化整個作業流程，加速審核動作，進而提升行政作業效率，增進服務品質。

畢業資格審核，一直都是各個學校每學年必須要進行的作業，作業之正確性，則會影響學生畢業與否；作業的時間，也關係著畢業證書發放的時間點。所以一個好的畢業資格審核流程與制度，是相當重要的，本專題是以「致理技術學院」為例，進行電子化畢業資格審核系統之建置。

現行致理技術學院之畢業資格審核，是透過人工的方式進行，註冊組於畢業學年度學期，必須向各系所(科)索取應屆畢業生之「畢業資格審查標準」及「應修科目表」，再透過校務系統將應屆畢業生之歷年成績單列印出來，最後再進行人工比對的動作，比對完後再分送至各系所(科)及學生做確認，整個審核時間冗長，且正確性也時有疑慮。故希望透過建置「電子化畢業資格審查系統」，來加速畢業資格審核動作及提升畢業資格審核之正確性，學生也可透過此系統查詢目前課程修習狀況，具體來說，可分成「註冊組」、「課務組」、「系所(科)」以及「學生」四方面來做說明。

(一) 註冊組

註冊組進行畢業資格審查，主要是針對「校訂必選修科目」及「通識課程」進行審核的動作，註冊組是「電子化畢業資格審查系統」建置之最大受益者，系統建置完成後，註冊組無須再利用人工審核的方式，檢視所有應屆畢業生之畢業資格，可透過系統進行自動化審核，可大大縮短畢業資格審核時間，提升行政效率。針對審核未通過的學生，也可透過系統持續追蹤。

(二) 課務組

畢業資格審核作業，主要是根據各系所(科)所訂定的「應修課目表」及「畢業資格門檻」做審核之動作，各系所(科)於每學年度新生入學前，必須訂定「應修課目表」及「畢業資格門檻」，課務組主要扮演審核「應修課目表」及「畢業資格門檻」的角色。課務組可透過電子化畢業資格審核系統，審核各系所(科)之「應修課目表」及「畢業資格門檻」，一旦審核通過後，各系所(科)將無法修改相關內容，若欲修改，必須再次通過課務組審核，以課務組來做統籌管理與維護，以維護畢業資格審核之正確性。課務組也可於系統上查詢歷年各系所(科)之「應修課目表」及「畢業資格門檻」。

(三) 系所(科)

各系所(科)於每學年度新生入學前，可透過電子化畢業資格審核系統進行「應修課目表」及「畢業資格門檻」建置，建置時也提供帶入上一屆新生之「應修課目表」及「畢業資格門檻」的功能，系所(科)可透過修改上一屆新生的相關資料，加速建置「應修課目表」及「畢業資格門檻」的時間及提升正確性。於各系所(科)於畢業資格審核作業，主要審核該系所(科)應屆畢業生之系訂必修科目及系訂畢業資格門檻。透過電子化畢業資格審查系統，各系所(科)不必再利用人工方式審核，系統會自動進行應修科目表及系所(科)畢業門檻的比對，也大大節省系所(科)畢業資格審查的時間。各系所(科)也可透過系統隨時追蹤系上學生修課情形，協助學生順利畢業。

(四) 學生

學生可透過電子化畢業資格審查系統，了解自己修課狀態，例如：可以知道自己必修學分修了多少、選修學分修了多少、通識課程是否修習完畢、還差多少必修或通識學分才可畢業、是否需要考取證照等。透過此系統所呈現的資訊，學生也可以有個依據，針對自己未來修課計畫做進一步的規劃。

第二節 實際效能

本專題架設在.NET 平台上，利用 Visual Studio 2005 為開發工具，撰寫 ASP.NET 網站應用程式，連結 SQL Server 2005 資料庫，建構出電子化的畢業資格審查系統，而本章節將介紹系統整體的實作成效。

系統實作的成效，分成「註冊組」、「課務組」、「系所(科)」及「學生」四方面來做分析與探討。

(一) 註冊組

表 4-1、管理者介面系統成效列表（註冊組）

	功能	系統實作成效
註 冊 組	畢業資格審查	針對校訂必修科目、校訂選修科目(通識科目、核心通識)、各系所(科)畢業門檻做初審及總審。
	畢業資格審查 狀態查詢	查詢全校應屆畢業生畢業資格審核情形，包括未通過及已通過審核名單。
	應修科目表查詢	查詢各學制系所(科)之應修科目表。
	學生歷年修課 狀態查詢	查詢全校應屆畢業生目前修課狀態，例如：必選修學分數、距畢業資格所欠缺條件等。
	學生成績查詢	查詢全校學生歷年修課成績。

(二) 課務組

表 4-2、管理者介面系統成效列表 (課務組)

	功能	系統實作成效
課 務 組	應修科目表審核	審核各學制系所(科)所建立之應修科目表之正確性。
	應修科目表維護	維護各學制系所(科)之應修科目表，可進行應修科目表之新增、刪除、修改等動作。
	應修科目表查詢	查詢各學制系所(科)之應修科目表。
	學生歷年修課 狀態查詢	查詢全校應屆畢業生目前修課狀態，例如：必選修學分數、距畢業資格所欠缺條件等。
	學生成績查詢	查詢全校學生歷年修課成績。

(三) 系所(科)

表 4-3、管理者介面系統成效列表 (系所 (科))

	功能	系統實作成效
系 所 (科)	畢業門檻建立	建立系所(科)畢業所需之條件與門檻。
	應修科目表建立	建立系所(科)之應修科目表。
	應修科目表查詢	查詢系所(科)歷年應修科目表。
	畢業資格審查	針對系所(科)應屆畢業生做畢業資格審核的動作，主要審核系訂必選修科目以及系所(科)畢業門檻。
	畢業資格審查 狀態查詢	查詢系所(科)應屆畢業生畢業資格審核情形，包括未通過及已通過審核名單。
	學生歷年修課 狀態查詢	查詢系所(科)應屆畢業生目前修課狀態，例如：必選修學分數、距畢業資格所欠缺條件等。
	學生成績查詢	查詢系所(科)學生歷年修課成績。

(四) 學生

表 4-4、管理者介面系統成效列表

	功能	系統實作成效
學生	畢業資格審查 狀態查詢	查詢學生本人畢業資格審核情形。
	畢業資格審查 狀態確認	確認學生本人畢業資格審查結果無誤。
	學生歷年修課 狀態查詢	查詢學生本人目前修課狀態，例如：必選 修學分數、距畢業資格所欠缺條件等。
	應修科目表查詢	查詢學生所屬系所(科)之歷年應修科目表。
	學生成績查詢	查詢學生本人歷年修課成績。

第三節 系統畫面

使用者於「致理技術學院畢業資格審查入口」登入畢業資格審查系統，如圖 4-1。



圖 4-1、登入畫面

進入畢業資格審查系統後，系統會依照使用者權限的不同，於系統畫面左方的樹狀功能選單顯示不同的功能項目，如圖 4-2、4-3、4-4、4-5，系統畫面中央為最新消息，在這會公佈畢業審查或系統相關之最新消息與內容。

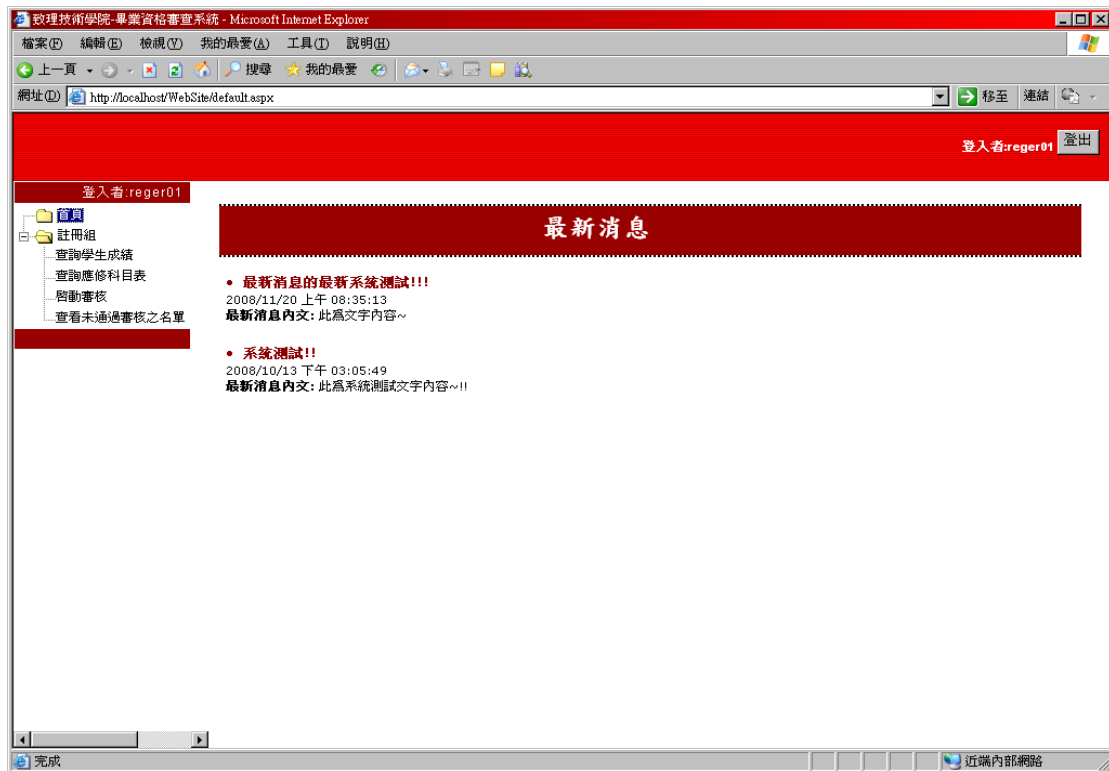


圖 4-2、註冊組登入



圖 4-3、課務組登入



圖 4-4、系所(科)登入



圖 4-5、學生登入

以下將根據畢業資格審查系統的流程，進行系統畫面的呈現與說明：

1. 建立系所(科)畢業門檻

各系所(科)於每學年度第二學期末時，必須於系統新增下一學年度入學新生之「系所(科)畢業門檻」，可點選系統左方樹狀選單裡的「系所畢業門檻」功能，進行系所(科)畢業門檻的建立，如圖 4-6 所示。

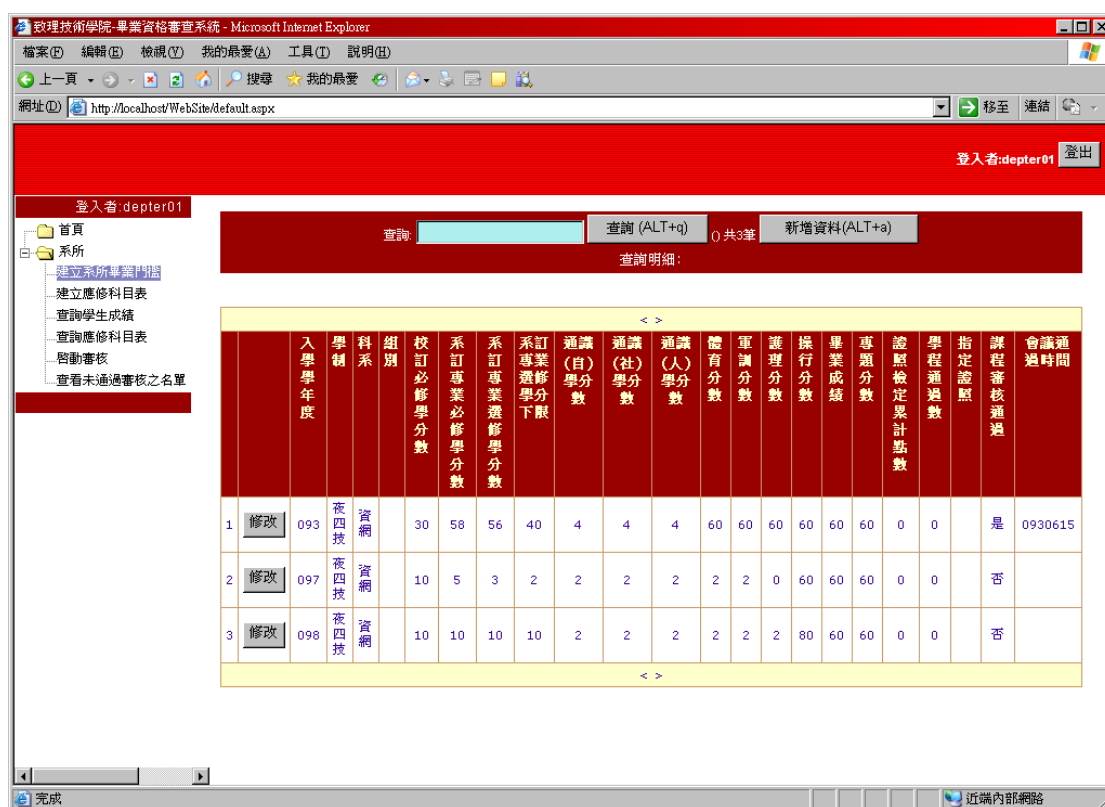


圖 4-6、建立系所(科)畢業門檻

點選「新增資料」，可切換至新增資料內容網頁，依照網頁內容輸入下一學年度入學之新生的系所(科)畢業門檻，如圖 4-7 所示。

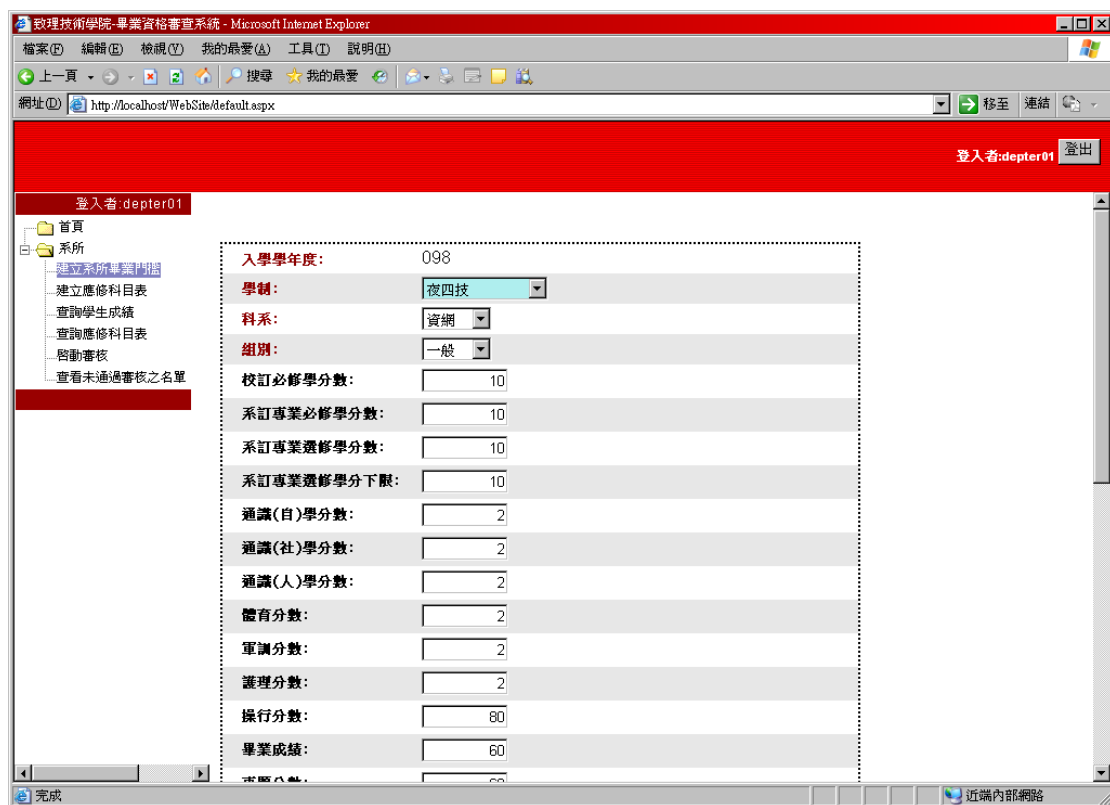


圖 4-7、輸入系所(科)畢業門檻內容

輸入資料時，可選取目前使用者需要的輸入之學制及組別（預留將來升科技大學時使用），並根據相關畢業資格條件做設定。設定完成後，必須建立應修科目表，應修科目表是以學年度學期為單位（即規定每學年度學期必須修習那些課程），所以各科系（所）必須先針對不同學制設定對應的科目表以及對應適用的學年度及學期，於網頁下方可做此設定，如圖 4-8 所示。



圖 4-8、建立各學制修業年限資料

完成系所(科)畢業門檻設定且產生應修課目表後，系所(科)的使用者可點選左方樹狀選單的「管理應修科目表」功能網頁。此網頁所進行之動作為修改下一學年度預先訂定之應修科目表，如圖 4-9 所示。

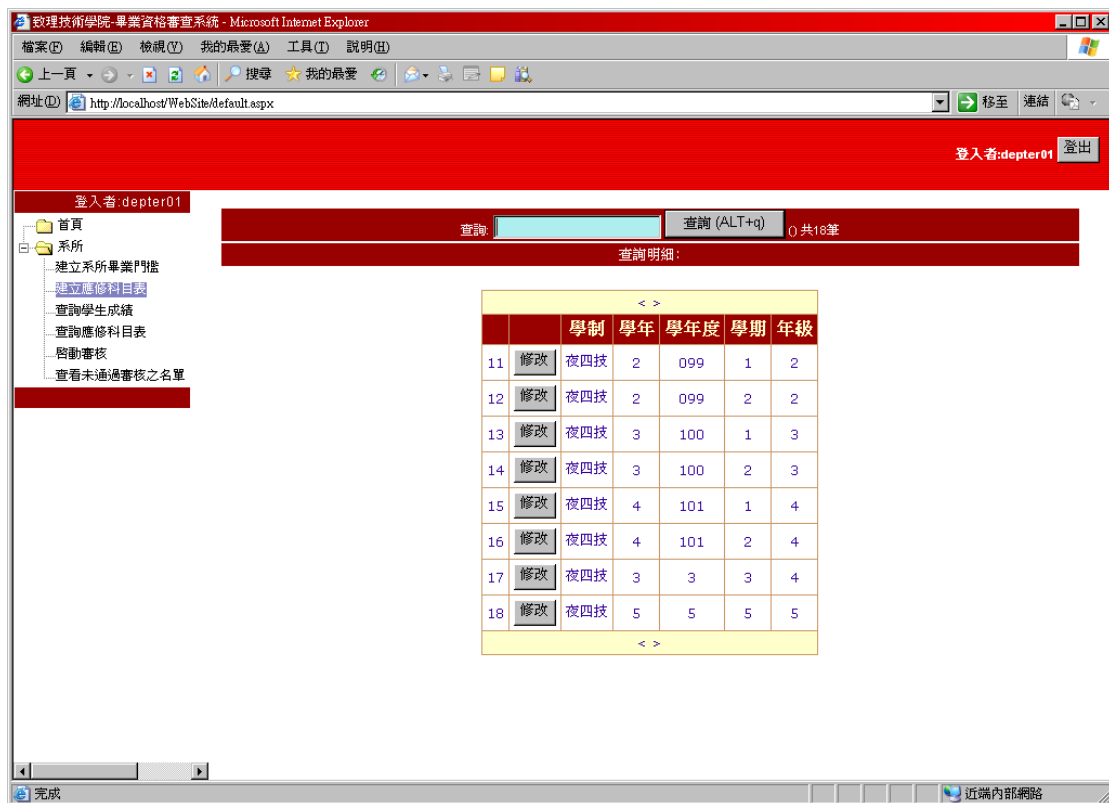


圖 4-9、建立應修科目表

使用者按下畫面中的「修改」按鈕之後，進入應修科目表修改畫面。此畫面之應修科目表為系所(科)畢業門檻建立後，由各系所(科)所自行建立符合各系所(科)學分要求的獨立表格，因此第一次進行修改的應修課目表，下方皆為空白之輸入框，如圖下圖 4-10 所示。



圖 4-10、初修改之應修科目表

在課程代碼的表格下，按下「速查」功能按鈕後，跳出小視窗。視窗內所顯示之課程基本資料可讓各系所（科）的使用者查閱後選擇需要加入應修科目表之課程，如下圖 4-11 所示，按下「選定」功能按鈕後，小視窗自動關閉，且將使用者所選定之課程寫入應修科目表內。

致理技術學院課程基本資料查詢 - Microsoft Internet Explorer

課程基本資料查詢： 查詢 () 共51筆

	課程代碼	課程名稱	選課別	學分	是否為共同選修	是否為學年選修	是否為重點課程	及格分數	是否計入平均	是否計入累計學分	上課時數	實習時數	上課使用特別教室	實習使用特別教室	課程別	是否為通識	總時數	通識類別代碼
<input type="button" value="選定"/>	634G2002	體育(二下)	必修	0	否	否	否	60	是	是	2	0	否	否	體育	否	2	
<input type="button" value="選定"/>	634H2001	生涯規劃與發展	必修	2	否	否	否	60	是	是	2	0	否	否	一般	是	2	核心通識
<input type="button" value="選定"/>	634H2002	人際關係與溝通	必修	2	否	否	否	60	是	是	2	0	否	否	一般	是	2	核心通識
<input type="button" value="選定"/>	634R2001	系統程式	必修	2	否	否	否	60	是	是	2	0	否	否	一般	否	2	
<input type="button" value="選定"/>	634R2002	電信網路工程	必修	2	否	否	否	60	是	是	2	0	否	否	一般	否	2	
<input type="button" value="選定"/>	634R2003	光纖通訊系統	必修	2	否	否	否	60	是	是	2	0	否	否	一般	否	2	
<input type="button" value="選定"/>	634R2004	作業系統	必修	2	否	否	否	60	是	是	2	0	否	否	一般	否	2	
<input type="button" value="選定"/>	634R2005	寬頻網路	必修	2	否	否	否	60	是	是	2	0	否	否	一般	否	2	
<input type="button" value="選定"/>	634R2006	電子電路學	必修	2	否	否	否	60	是	是	2	0	否	否	一般	否	2	
<input type="button" value="選定"/>	634R2007	計算機組裝	必修	2	否	否	否	60	是	是	2	0	否	否	一般	否	2	

關閉視窗(ALT+c)

圖 4-11、選擇加入應修科目表之課程

各系所(科)的使用者選完應修科目表內的課程代碼之後，點選位於系統畫面左方修課項目表格裡第一欄的「速查」按鈕，會跳出小視窗，視窗內所顯示之修課項目資料表格，可讓使用者查閱後選擇需要加入應修科目表之修課選別代碼名稱，如下圖 4-12 所示，按下「選定」功能按鈕後，小視窗自動關閉，且將使用者所選定之修課項目代碼寫入應修科目表內。



圖 4-12、選擇修課項目代碼

重複上述之步驟，將此學期之應修科目表裡所需要的課程資料逐一填入，如下圖 4-13 所示，結束後按下「儲存修改」功能按鈕，將此應修課目表存檔。按下「返回查詢畫面」，返回「管理應修科目表」功能網頁，繼續完成不同學年與學期所需之應修科目表。

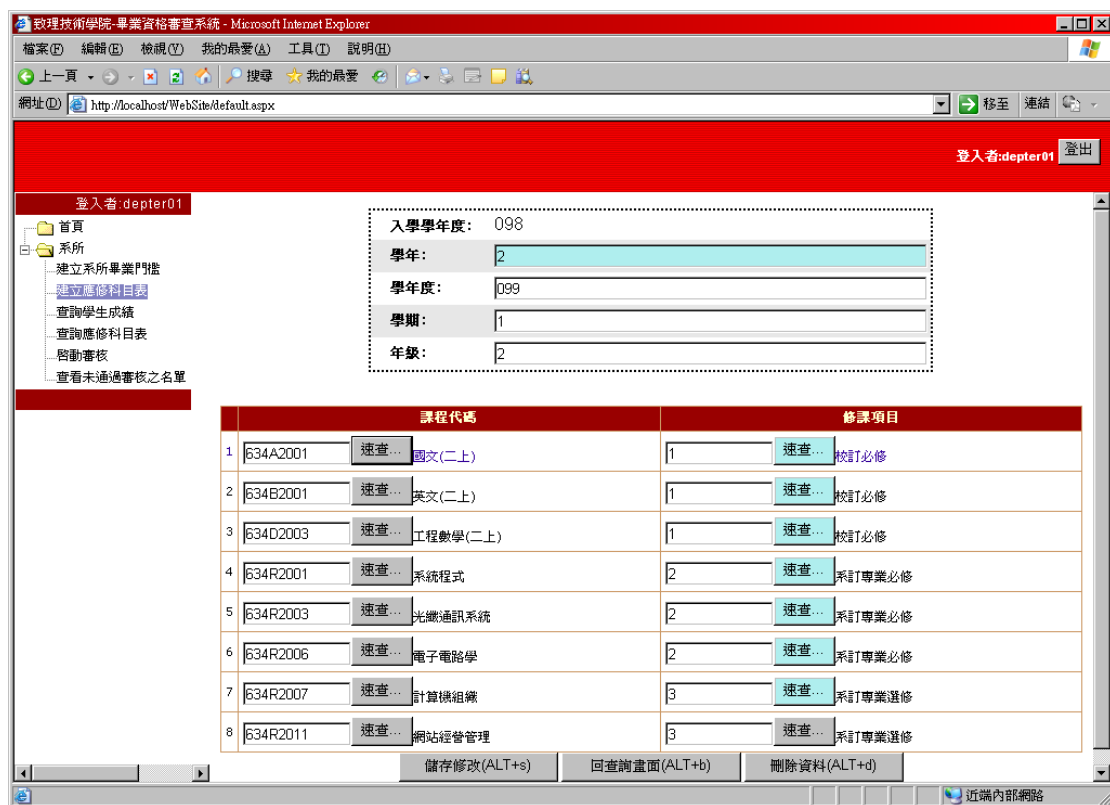


圖 4-13、應修科目表選填完畢

系所(科)使用者訂定完應修科目表後，下一程序為課務組審核系所(科)畢業門檻。當課務組登入畢業資格審查系統後，點選位於畫面左邊樹狀選單中的「審核應修科目表」功能按鈕，進入審核畫面。

畫面中央會出現各學制的各系所(科)使用者所訂定之下一學年度應修科目表；按下需要審核的該筆資料前方「審核」功能按鈕，即可進入審核畫面，如下圖 4-14 所示。

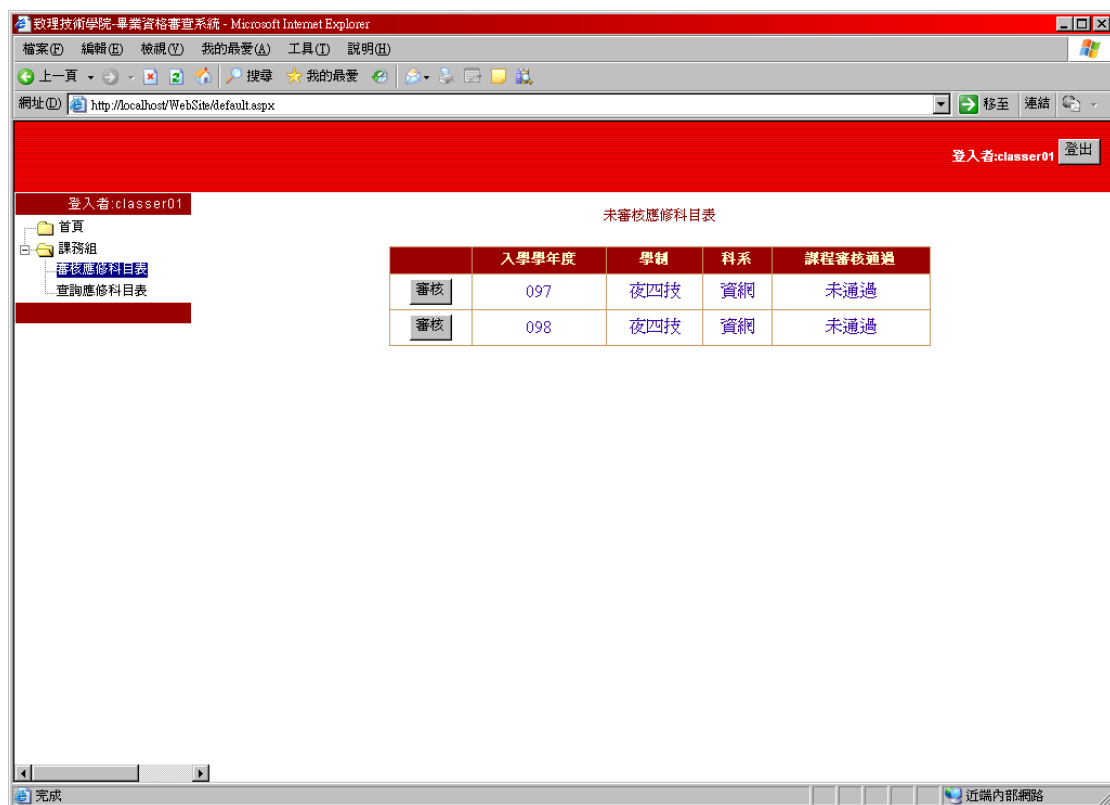


圖 4-14、選擇未審核應修科目表

審核畫面上方呈現該指定學年度應屆畢業學生所需之畢業門檻，下方則將應修科目表按照學年與學期分類的方式呈現，如下圖 4-15、圖 4-16 所示。

如審核中有發現問題，審核人員可於圖 4-16 系統畫面中最下方的備註欄輸入備註；若是課務組確認審核無誤後，按下位於畫面下方的「通過審核」功能按鈕，該筆資料將會從「審核應修科目表」（圖 4-14）的功能網頁中移除，且各系所（科）所擁有的「管理應修課目表」（圖 4-11）功能網頁內，該年度應修科目表則被移除。

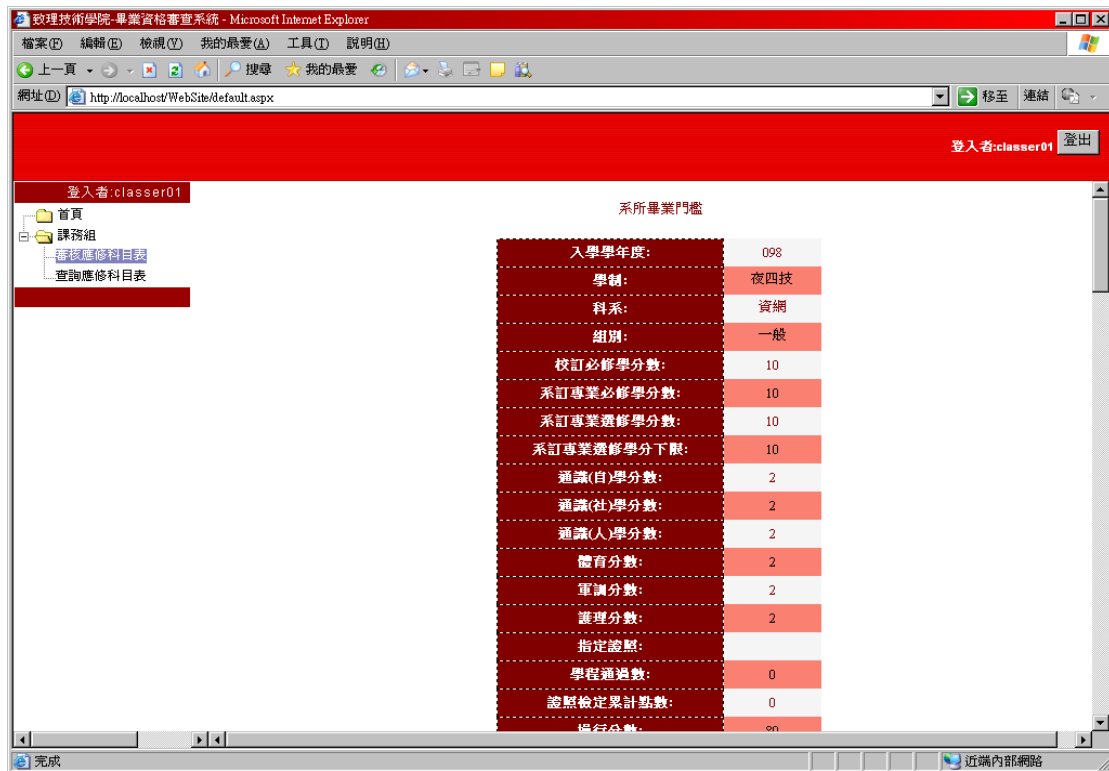


圖 4-15、課務組審核應修科目表(上方畫面)



圖 4-16、課務組審核應修科目表(上方畫面)

當下一學年度入學生所對應之應修課目表完成後，依照系統流程，則會來到下一學年度第一學期末；當註冊組登入畢業資格審查系統後，點選位於系統畫面左邊樹狀選單裡的「啟動審核」功能網頁，於畫面中央區域出現審查按鈕，審查分為「學期審查」與「畢業班審查」兩項目，若是第一學期進入「啟動審核」功能網頁時，則不會顯示「畢業班審查」之選項，如圖 4-17 所示；反之若為第二學期進入「啟動審核」功能網頁時，則顯示「畢業班審查之選項」，如圖 4-18 所示。

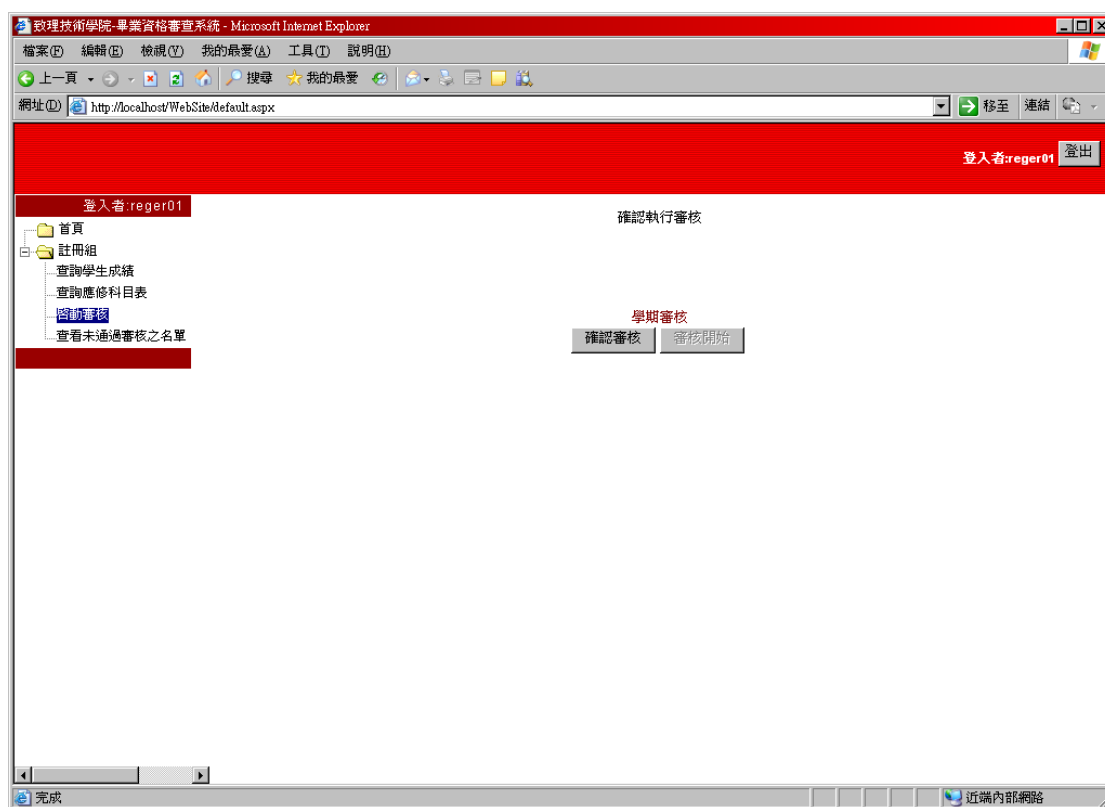


圖 4-17、學期審核

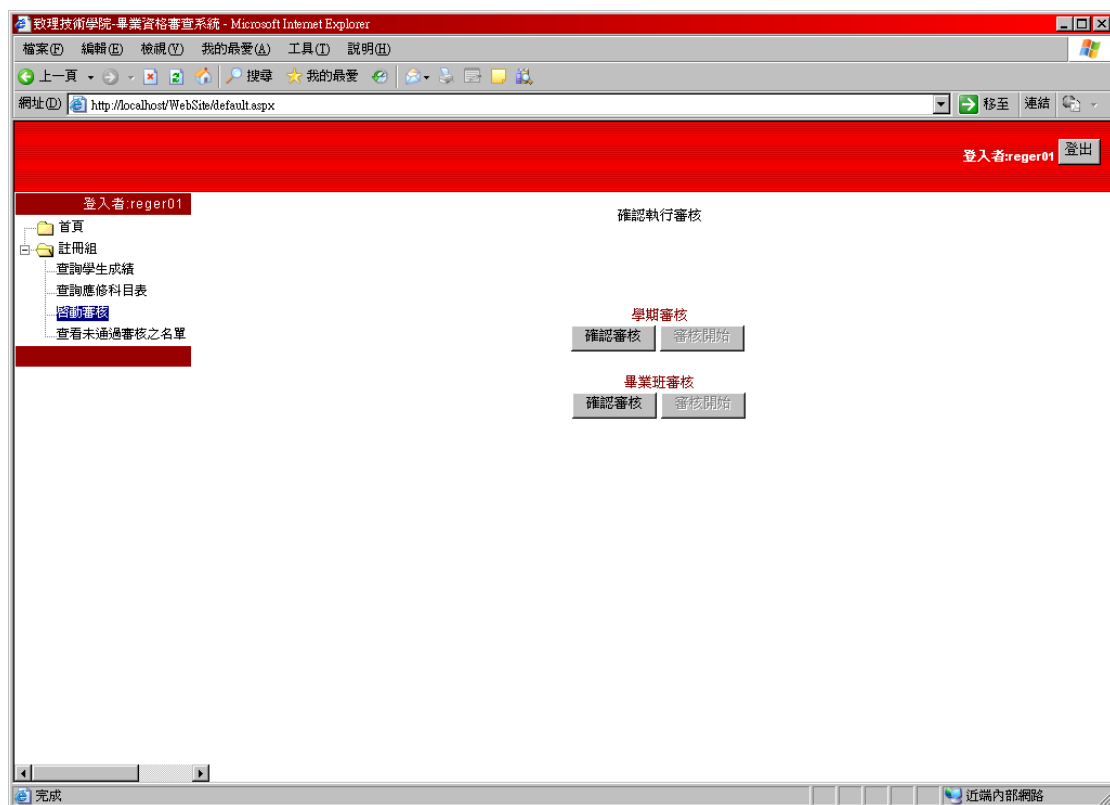


圖 4-18、學期審核與畢業班審核

在一開始進入畫面時，「審核開始」這項功能按鈕會是呈現灰色並且無法使用的狀態，使用者必須先按下「確認審核」按鈕之後才能夠使用「審核開始」的功能按鈕。

按下「確認審核」功能按鈕後，系統將會比對該學期所有科目的授課教師是否都以送出成績，如遇有授課教師尚未送出期末學期成績，則無法進行審核；待所有授課教師皆送出期末學期成績後，再次按下「確認審核」功能按鈕，即可確認開始執行審核作業時，此時「確認審核」按鈕將會以灰色無法使用的狀態呈現，而右邊的「審核開始」按鈕變成可使用之按鈕，如圖 4-19 所示。

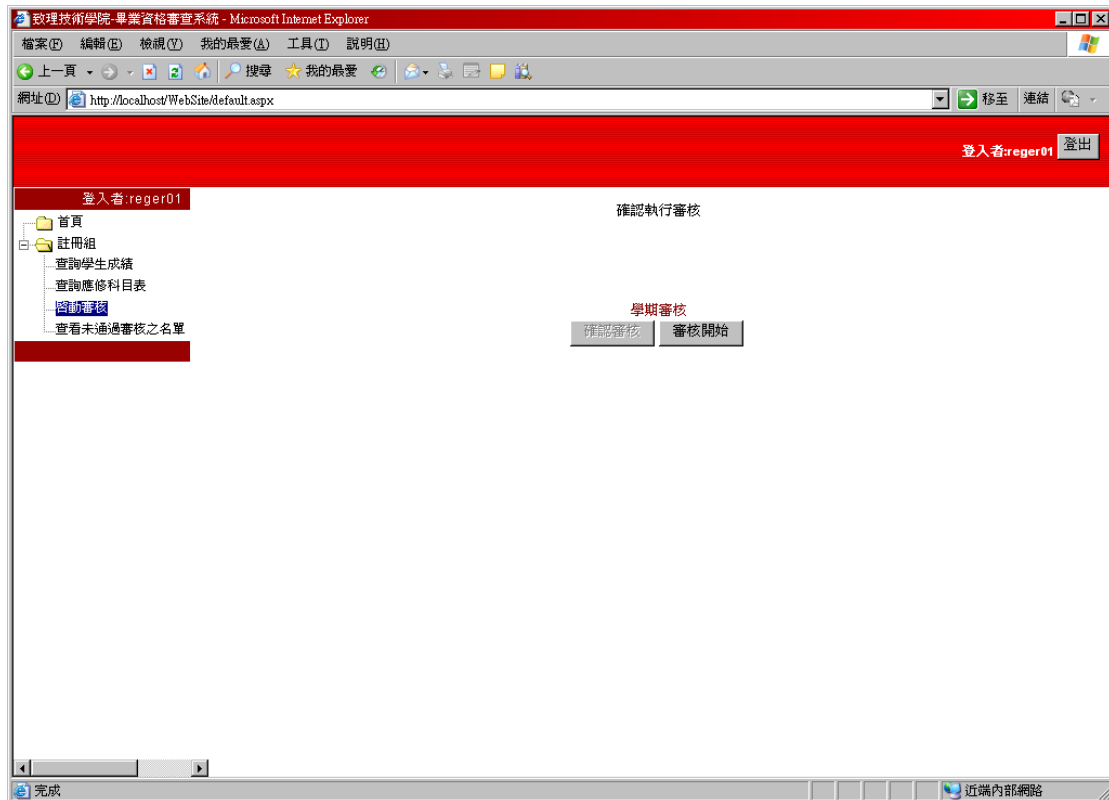


圖 4-19、學期審核開始

由於畢業班須比其他班級提早開始進行畢業資格審核，所以若是於第二學期使用畢業班審核的「確認審核」功能按鈕，比對的則是畢業班的授課教師是否都已送出成績，如所有畢業班授課教師皆送出期末學期成績後，按下「確認審核」功能按鈕，同上，確認可開始執行審核作業時，「確認審核」該按鈕將會以灰色無法使用的狀態呈現，而右邊的「審核開始」按鈕變成可使用之按鈕，如圖 4-20 所示。

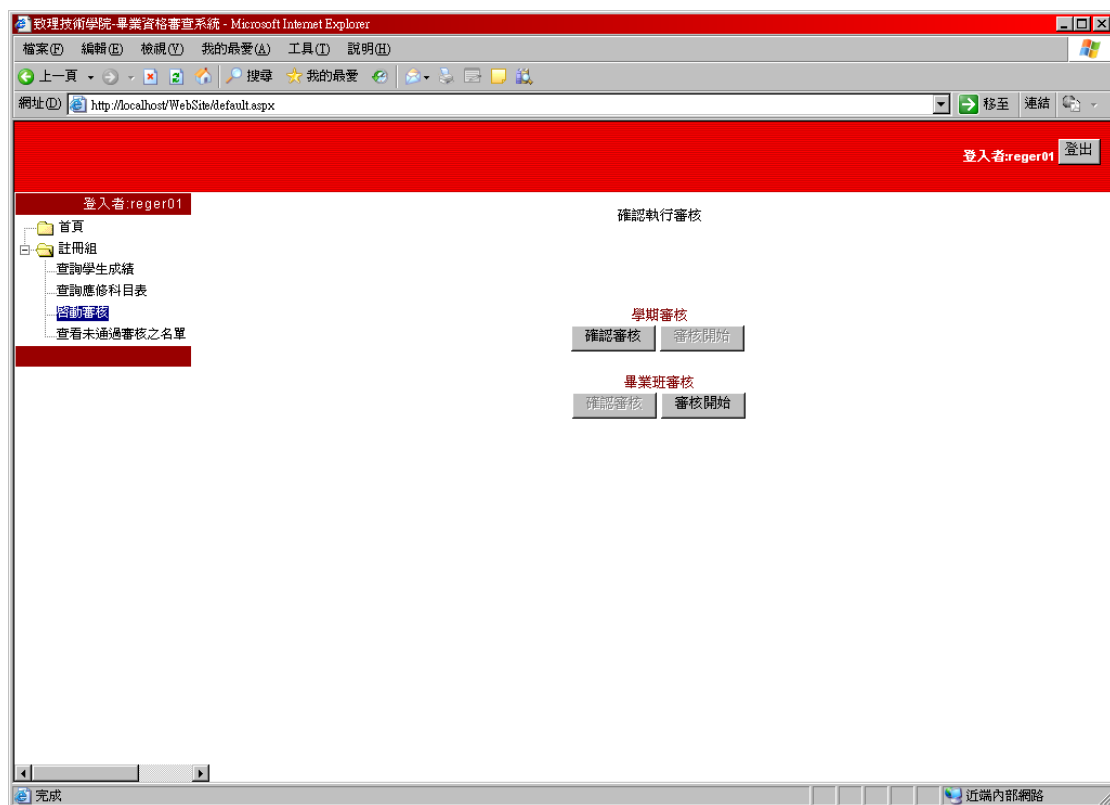


圖 4-20、畢業班審核開始

當註冊組按下「審核開始」按鈕後，系統會開始對學生的學期成績判斷是否達到及格分數，但只會判斷校定的必選修科目，其他由系定的必選修科目，則由各系所(科)執行。

註冊組審核部分結束，系所(科)使用者登入畢業資格審查系統，點選畫面左邊的樹狀選單裡的「啟動審核」功能網頁，和註冊組所呈現的網頁一樣，分為「學期審查」與「畢業班審查」兩項目，與註冊組方面相同，若是第一學期進入「啟動審核」功能網頁時，則不會顯示「畢業班審查」之選項；反之若為第二學期進入「啟動審核」功能網頁時，則顯示「畢業班審查之選項」。

系所(科)與註冊組雙方的功能使用差異性在於「確認審核」功能按鈕的判斷不相同，當註冊組按下「確認審核」按鈕時，所執行的是比對該學期所有科目的授課教師是否都以送出成績；而若是系所(科)使用者按下「確認審核」的功能按鈕後，所執行的比對則是判斷註冊組是否已經按下「確認審核」的按鈕，當確定註冊組已經判斷且可以進行審核，系所(科)才可按下「確認審核」的按鈕。按下「確認審核」功能按鈕，確認可開始執行審核作業時，「確認審核」該按鈕將會以灰色無法使用的狀態呈現，而右邊的「審核開始」按鈕變成可使用之按鈕。

由於畢業班須比其他班級提早開始進行畢業資格審核，所以若是於第二學期使用畢業班審核的「確認審核」按鈕，同樣也是先判斷註冊組是否已經執行且通過畢業班審核的「確認審核」，系所(科)方面才可以執行畢業班方面的「審核開始」按鈕。

當系所(科)按下「審核開始」按鈕後，系統會開始對學生的學期成績判斷是否達到及格分數，而判斷的部份就是註冊組未判斷的必選修科目，非本系的學程科目也一並於此判斷。

於每學年度的下學期，註冊組與系所(科)都已完成畢業班審查結束後，畢業班的學生登入畢業資格審查系統，在畫面左邊樹狀功能網頁點選「查詢審核」網頁，畫面中央會出現該學生每一學期的修課成績與審核結果，如圖 4-21 所示。



圖 4-21、學生查詢成績（上方畫面）

按下右方「查看系所(科)畢業門檻」功能按鈕，則跳出小視窗顯示畢業門檻詳細內容，拉至畫面下方，可以看到學生的學分統計與所需學分數，更下方則有二個按鈕，「審核確認無誤」與「錯誤回報」，如圖 4-22 所示。學生登入看完後若確認系統資料無誤，則按下「審核確認無誤」按鈕，完成畢業資格審查系統流程；若學生發現有誤，或有其他問題，則按下「錯誤回報」按鈕，會跳出文字輸入窗格，輸入原因後送出，該學生則被系統歸類於為審核未通過（錯誤回報）之狀態。

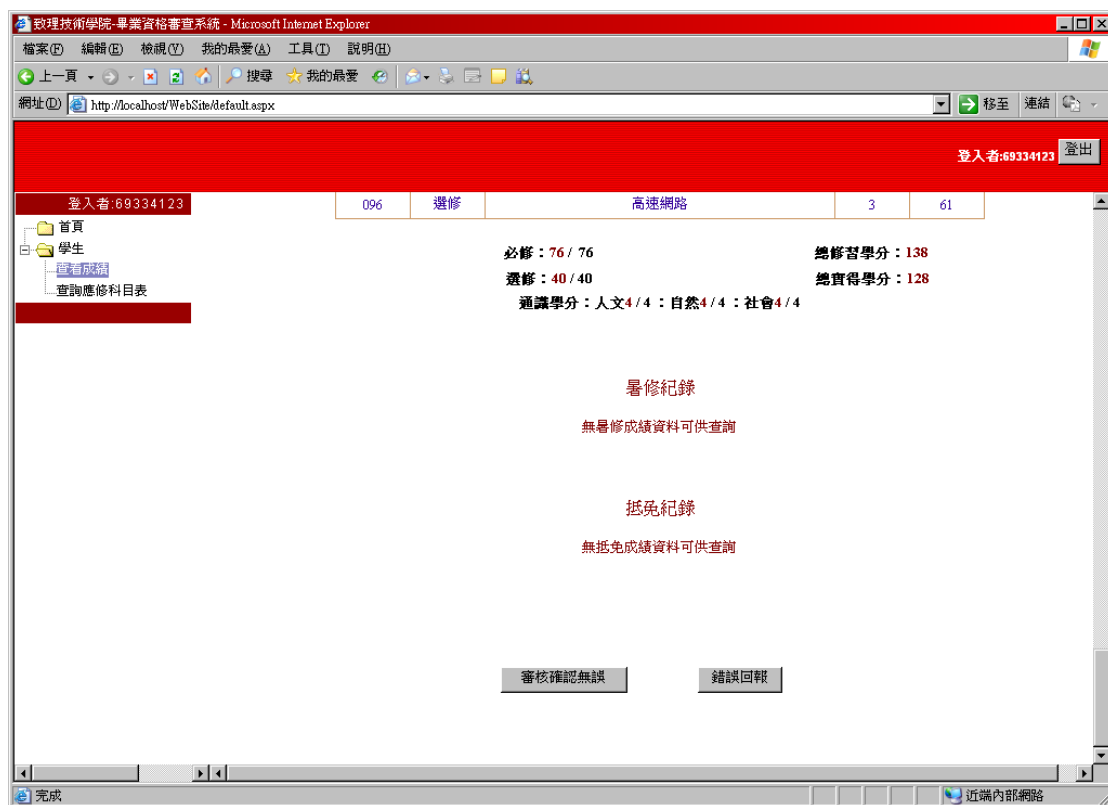


圖 4-22、學生查詢成績（下方畫面）

在註冊組與系所(科)登入後網頁的左方樹狀功能表裡有「未通過審核名單」，此名單內之資料皆為審核未通過之學生資料，包含學分不足、必修未過、未達到系所(科)畢業門檻、學生確認錯誤回報等等；只要狀態是審核未通過之學生資料皆於此匯整，註冊組與系所(科)擁有查看之權力，假如該筆資料是學生自行回報錯誤，若學生有於回報時在備註欄位內輸入內容，系統將再此一併顯示，如圖 4-23 所示。未通過審核名單此區資料一律由註冊組或系所(科)雙方以人工方式進行複審，待複審完畢後，可於畫面下方點選「審查通過」，會跳出兩次的確認選項，以防誤按。確認完成後，該筆學生資料成為審查通過狀態，並從「未通過審核名單」中移除。



圖 4-23、未通過審核之名單

各單位（包含註冊組、課務組、系所(科)、學生）皆可利用畢業資格審查系統查詢歷屆每一學年度之應修科目表與畢業門檻。登入系統後，點選位於左方樹狀功能表上的「查詢應修科目表」功能網頁，畫面中央會出現可以選擇查詢相關內容之下拉式選單，選擇完學制、科系、入學學年度三樣條件後，按下「執行查詢」按鈕，會出現對應之應修科目表，點選資料前方「查看」按鈕可查看詳細內容，如圖 4-24 所示；但尚未通過審核之下一學年度的應修科目表不在此查詢範圍內。



圖 4-24、查詢應修科目表

各單位（包含註冊組、課務組、系所(科)）皆可利用畢業資格審查系統查詢學生成績。註冊組與課務組皆有查看全校學生成績之權限；而各系查詢功能只有自訂查詢部分，且僅能查詢各自科系學生之成績；學生則是只能觀看自己成績，且學生介面僅能顯示個人學分資料功能。

在課務組與註冊組的學生查詢頁面下，可選擇依「學號查詢」或「自訂查詢」。選擇使用「學號查詢」方式後，按下「輸入查詢」按鈕，下方出現可輸入學號欄位，輸入學號後，按下「執行查詢」，於下方顯示查詢結果。如下圖 4-25 所示；點選資料後方「查看」按鈕則可查看該筆資料詳細內容。



圖 4-25、學號查詢

選擇「自訂查詢」，則依照順序選擇查詢項目，按下「執行查詢」，於下方顯示查詢結果，如圖 4-26 所示。點選資料後方「查看」按鈕可察看相關學生的詳細內容。



圖 4-26、自訂查詢

第五章 結論

第一節 結論

目前各大專院校的校務系統漸漸都走向電子化，行政流程電子化已經成為現代必然的趨勢；而目前以致理技術學院所使用的畢業資格審查方式來說，必須先請註冊組列印所有應屆畢業生之歷年成績單，並請各系所（科）一一列印應修科目表及應屆畢業資格門檻，經過各部門資料彙整後，再統一交由註冊組利用人工的方式來進行畢業資料比對，不但耗時且浪費紙本資源，另外以人工方式核對每位畢業生之畢業資格，也較容易出現錯誤之疑慮。

本專題利用 ASP.NET、Visual Studio 2005 以及 SQL Server 2005 等相關技術，建置出一個「電子化畢業資格審查系統」，實現改造現有的人工畢業資格審查方式，將畢業資格審查數位化與自動化，簡化了畢業資格審查流程，節省了大量的人力與時間，達到有效的管理以及增加方便性。

資料電子化是企業電子化的基礎流程改造必要步驟之一，除了可以改善原本的工作程序之外，在未來更可以與其他校務系統結合，進行整體校務系統電子化，增加校務整體運作效率。

第二節 未來展望

「電子化畢業資格審查系統」已初步達到將畢業資格審查電子化的目的，簡化了人工畢業資格審查的流程、增加畢業資格審查的正確性以及節省人力資源等目的，但仍有不足之處，為了能使「電子化畢業資格審查系統」達到最大效益，對於系統未來的展望與期許，描述如下：

- (1) 結合校內資訊系統，達到校園 e 化：本系統是以「致理技術學院」為研究範圍所建置出來的一套電子化畢業資格審核系統，為學校目前所沒有的資訊系統，若是能有效地結合校內各個資訊系統，例如：單一簽入入口網站、學生選課系統、導生資源 e 化系統以及即將要開發的學生歷程檔案系統等，必定能為致理技術學院達到全面校園 e 化之路，盡上一份心力，讓致理成為「坐擁資訊，掌控全球」的資訊校園。
- (2) 導生畢業資格審核查詢：目前電子化畢業資格審查系統，提供註冊組、各系所(科)以及學生可於系統查詢畢業資格審核狀況，若是能開放導師查詢自己班上學生修習課程的狀況以及畢業資格審核的情形，對於輔導學生畢業或是規劃修習課程，都會有很大的幫助。
- (3) 特殊狀況審核機制：電子化畢業資格審查系統，已初步對校訂必修、通識課程、系訂必選修以及系所(科)畢業門檻做初步的審核，已大大地縮短畢業資格審查的時間，並且增加畢業資格審查的正確性。但是對於一些較特殊的狀況，例如：轉學或轉系抵免、跨校或跨系

修課以及修習雙主修、輔系、學程等狀況，並無審核的機制。若是能於未來加入這些審核機制，電子化畢業資格審查系統必定能更趨完整且達到更大效益。

- (4) 更友善的使用者介面：所謂「一圖勝過千言萬語」，目前系統大多以文字或是數值的方式呈現資訊，若是能用更視覺化的方式呈現，例如利用圖表方式呈現訊息，不管是對於使用者或是決策者，都會有更直覺的感受，也可達到事半功倍之效果。

參考文獻

- [1]. .NET Framework 概觀 < [http://msdn.microsoft.com/zh-tw/library/zw4w595w\(vs.80\).aspx](http://msdn.microsoft.com/zh-tw/library/zw4w595w(vs.80).aspx) > 。
- [2]. 布拉特 (PlattDavid) 著/潘德龍譯，2004，Microsoft .NET 徹底研究，第一版，文魁資訊。
- [3]. 山田祥寬，2006，學會 ASP.NET 2.0 的關鍵課程/汪牧君譯，旗標出版社。
- [4]. 文件庫 ASP.NET 技術 < [http://msdn.microsoft.com/zh-tw/library/aa720210\(VS.71\).aspx](http://msdn.microsoft.com/zh-tw/library/aa720210(VS.71).aspx) > Microsoft 。
- [5]. 郭瑞軍、郭磬君著，2007，ASP.NET 2.0 網站開發實例演練，電腦人文化。
- [6]. 陳會安，2006，ASP.NET 2.0 網頁設計範例教本，初版，學貫出版社。
- [7]. 陳宗興，2003，ASP.NET 網際網路系統程式開發手冊，初版，全華出版社。
- [8]. 王國榮，ASP.NET 網頁製作教本-從基本語法學起 < <http://web.ezcare.tw/hkc/file/F6461/F6461-ch01.ppt> >
- [9]. 陳會安，2005，SQL Server 2005 資料庫安全系統設計與開發實務，初版，學貫行銷股份有限公司。
- [10]. 顏春煌，2007，資料庫系統原理與應用，初版，金禾資訊。

- [11]. 魯瓦卡巴 (RuvalcabaZak), 2003, ASP. Net 與資料庫完全架站攻略, 上奇科技。
- [12]. 譚普勒曼·歐爾生 (TemplemanJulian·OlsenAndy) 著/陳佳新譯, 2002, Microsoft Visual C++.NET 程式設計之鑰, 文魁資訊。
- [13]. 艾爾瑪斯利、納瓦茲 (ElmasriRamez NavatheSham) 著/藍中賢 & 吳郁瑩譯, 2005, 資料庫系統原理, 基峰資訊。
- [14]. Microsoft SQL Server 2005 功能介紹 <<http://www.microsoft.com/taiwan/sql/prodinfo/overview/whats-new-in-sqlserver2005.mspx>>。
- [15]. 黃三凱, 2007, 同質性網路資料庫系統之效能評估與探討, 私立立德管理學院應用資訊研究所碩士論文。
- [16]. Microsoft Visual Studio 簡介, <[http://msdn.microsoft.com/zh-tw/library/fx6bk1f4\(VS.80\).aspx](http://msdn.microsoft.com/zh-tw/library/fx6bk1f4(VS.80).aspx)>。