

# 致理技術學院

## 資訊網路技術系 實務專題報告

### 遊戲買賣交換平台

指導教師：蕭勝華

學生：陳韋竹(69434101)

林宗儒(69434135)

何文惠(69434137)

李薇薇(69434172)

中華民國 97 年 12 月

# 致理技術學院

## 資訊網路技術系 實務專題報告

### 遊戲買賣交換平台

學生：陳韋竹(69434101)

林宗儒(69434135)

何文惠(69434137)

李薇薇(69434172)

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 12 月

# 專題研究授權書

本授權書所授權之專題研究為\_\_\_\_\_

共\_\_4\_\_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：\_\_\_\_\_

同意       不同意

本組同學共\_\_4\_\_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：

(親筆正楷)

學號：

(務必填寫)

中華民國      年      月      日

## 誌 謝

四年的大專夜校生活隨著論文的完成下拉下了落幕，但是也為人生的另一個舞台揭開了序幕。回憶如幻燈片般浮現在我心頭，有歡笑有淚水交織著。專題研究的過程是孤寂且漫長的，非一己之力所能完成，其間遭遇到不少的挫折與挫敗，幸而在艱難的時刻有一群的優秀的同學們互相鼓勵著，與老師不厭其煩的指導，讓我們能如期的完成本專題，絕對要萬分感謝我們的指導老師蕭勝華蕭老師，老師從開始決定指導我們這組的專題研究之後，除了給我們很大的空間讓我們能夠在自己的專長上選擇發揮，也不侷限我們的研究方向，老師不論在任何的領域或是技術上都能給予我們跟指導，在這段期間深深的領會到老師的學識淵博，冠世之才，尤其最令我們感動的是老師在這些日子，即使校務繁身也抽空幫我們督促進度，就算是假日也從不倦怠，這段日子從未缺席，讓偶有幾次請假的我們真是汗顏，而老師也從未給我們壓力或是語氣稍重的責備，總是用信任跟引導的方式，也讓我們從中學習到自動自發，對自己負責的態度，使我們在此次的研究中，不論學識或是個人的品德上都有所成長，這一切都要感謝我們遇到了位這樣才德兼備的老師。每當在老師的提點當下有

所頓悟時，使我們皆有從學甚晚的懊悔，老師若能夠早點來我們系上，一定能會讓更多學生感同身受。

另外當然也要感謝我們這小組所有同學的齊心合作，大家從頭至今堅持相互鼓勵的精神，讓我們深深感受到團隊合作的重要性，以及團結的重要，我們一起走過這段大家一起熬夜的日子，大家一起討論、一起分享、一同煩惱，一起歡笑、其中感謝我們的小組長，陳韋竹同學的指揮跟分配，並且讓我們隨時都有團隊的向心力，專擅美文的文惠同學，讓我們有漂亮的版面，測試除錯並且又兼報告的薇薇同學，更是可以讓人信任的夥伴，我們在這些日子一同成長，感謝老師也感謝系上有這樣的專題研究，讓我們在心靈跟學識上，都有極大的收穫跟助益，希望這資訊網路技術系的優良傳統，能在我們(現任)的系主任領導下，日益精進，打破傳統，成為芊芊學子們心中的第一志願。

# 摘 要

隨著網路發展日益加速，引領時代的商機與潮流，如何從這變遷的資訊進化中找到與眾不同的商機，在各種網站林立，網路業者百家爭鳴的戰場上，最終出名獲勝得以生存的只有少數，我們要如何掌握消費者的心，獨創一格，尤其是在現在全球瘋遊戲的熱潮中，找出不同的方向，這是我們專題研究中最重要地方，如何把握商機，強調特色，讓我們在這市場中占有一席之地，我們將從消費型態上做不同的改變，由普遍的金錢交易中，來嘗試更多交易型態上的服務，由此我們決定設立一個與眾不同的網站，不論是在特色與安全方面加強探討，或是創新及服務上的應用，藉本次專題來印證我們期盼的方向，『交換』以及『拍賣』。

**關鍵詞：**拍賣網站、交易、交換、安全性、使用者、溝通，便利性。

# 目 錄

遊戲買賣交換平台 .....	i
遊戲買賣交換平台 .....	ii
專題研究授權書 .....	iii
目 錄 .....	vii
圖 目 錄 .....	x
表 目 錄 .....	xiii
第一章 緒論 .....	1
第一節 研究背景 .....	1
1-1.1 網路交易買賣平台使用與發展演進 .....	1
1-1.2 網路交易拍賣平台的環境趨勢 .....	2
1-1.3 網路交易平台的現狀 .....	5
第二節 研究動機與目的 .....	6
1-2.1 網路以物易物的故事 .....	6
1-2.2 網路交易拍賣平台所衍生的問題 .....	7
1-2.3 交易平台的溝通改進與應用 .....	8
第三節 研究範圍與報告書架構 .....	9
1-3.1 研究範圍 .....	9
1-3.2 報告書架構 .....	9
第二章 系統介紹 .....	11
第一節 C# .....	11
2-1.1 C#介紹 .....	11
2-1.2 C#特色 .....	12

2-1.3 C#之於其它程式語言 .....	15
第二節 SQL Server 2005 .....	17
2-2.1 SQL Server 2005 簡介 .....	17
2-2.2 SQL Server 2005 伺服器架構.....	18
2-2.3 SQL Server 2005 組成架構.....	20
第三節 ASP.NET .....	23
2-3.1 ASP.NET 與 ASP 的差異 .....	23
2-3.2 ASP.NET 應用程式分為兩種類型 .....	25
2-3.3 ASP.NET 網頁管理方式 .....	27
2-3.4 ASP.NET 授權、驗證和模擬網站安全機制.....	29
2-3.5 ASP.NET 3.5.....	31
第三章 拍賣網站平台及服務 .....	33
第一節 網站系統分析 .....	33
3-1.1 網站的類型與架構.....	33
3-1.2 網站系統分析.....	35
第二節 拍賣網站.....	37
3-2.1 拍賣網站功能介紹 .....	37
3-2.3 註冊及會員中心.....	55
第四章 網站的附掛功能及安全性 .....	66
第一節 網站的安全性 .....	66
4-1.1 SSL 的安全性.....	66
4-1.2 關於個人資料的安全性 .....	68
4-2.1 防止鍵盤側錄個人資料外洩.....	77
4-2.2 SQL Injection (資料隱碼).....	78
第五章 結論與探討 .....	83
第一節 網站的討論與改進 .....	83
5-1.1 使用者的意見及滿意度 .....	83
5-1.2 結論 .....	87
5-1.3 改進與遠景.....	90



」 .....	91
參考文獻.....	92

# 圖目錄

圖 2-1、SQL Server 2005 元件圖	18
圖 2-2、SQL 路徑圖	19
圖 2-3、SQL 組成架構圖	20
圖 3-1、專題網站製作流程圖	34
圖 3-2、專題網站系統架構圖	35
圖 3-3、網站首頁圖	37
圖 3-4、家用主機 PS3 最新消息頁面	38
圖 3-5、家用主機 Wii 最新消息頁面	39
圖 3-6、家用主機 XBOX 最新消息頁面	39
圖 3-7、家用主機 NDSL 最新消息頁面	40
圖 3-8、掌上型主機 PSP 最新消息頁面	40
圖 3-9、PS3 之熱門商品頁面	41
圖 3-10、Wii 之熱門商品頁面	41
圖 3-11、XBOX 360 之熱門商品頁面	42
圖 3-12、NDSL 之熱門商品頁面	42
圖 3-13、PSP 之熱門商品頁面	43
圖 3-14、熱門遊戲介紹之連結畫面	44
圖 3-15、商品類別選擇之畫面	45

圖 3-16、商品圖片上傳之畫面	46
圖 3-17、商品圖片預覽之畫面	47
圖 3-18、填寫商品內容描述之畫面	48
圖 3-19、商品上架成功之畫面	48
圖 3-20、賣家關於我之編輯畫面	50
圖 3-21、賣家關於我編輯完成後之呈現畫面	51
圖 3-22、賣家聯絡我之編輯畫面	51
圖 3-23、賣家聯絡我編輯完成後之呈現畫面	52
圖 3-24、買家得標信之編輯畫面	52
圖 3-25、上架商品交易方式為交換之畫面	53
圖 3-26、上架商品交易方式為有直購價拍賣之畫面	54
圖 3-27、上架商品交易方式為無直購價拍賣之畫面	54
圖 3-28、會員申請畫面	55
圖 3-29、新申請帳號無人使用之畫面	56
圖 3-30、新申請帳號有他人已使用之畫面	57
圖 3-31、密碼強度偵測之畫面	57
圖 3-32、使用者完成註冊之畫面	58
圖 3-33、註冊成功會員通知信之畫面	59
圖 3-34、會員修改個人資料之畫面	60

圖 3-35、會員修改密碼之畫面.....	61
圖 3-36、會員登入密碼不正確之畫面.....	61
圖 3-37、會員輸入帳號及註冊聯絡信箱之畫面.....	62
圖 3-38、系統寄出索取新密碼之驗證信件.....	63
圖 3-39、會員設定新密碼之畫面.....	63
圖 3-40、會員成功索取新密碼之畫面.....	64
圖 3-41、認證失敗之畫面.....	65
圖 4-1、會員資料表及公鑰資料表.....	69
圖 4-2、ForgotPwd.aspx 頁面.....	70
圖 4-3、防止鍵盤側錄小鍵盤.....	78

# 表 目 錄

表 5-1、網站測試滿意度.....	85
--------------------	----

# 第一章 緒論

## 第一節 研究背景

### 1-1.1 網路交易買賣平台使用與發展演進

網路拍賣這項服務，在國內的歷史只有 7 年而已。雅虎奇摩是在 2001 年才開闢出專屬的拍賣，第一個月上架的拍賣商品只有 2000 件左右，不過到現在雅虎奇摩已經號稱每個月擁有三百萬件刊登物件，其間的變化很大。早期的網路拍賣賣家，不少是以做副業賺外快的方式經營，小額批些商品來賣，也有人直接將家中多餘的東西出清，這些都算是簡單型的拍賣模式。不過，在趨勢所至下，越來越多人把網路拍賣當成一種發展自己事業的跳板。因為網路拍賣除了可以增加收入外，更能以較低的成本來完成創業的夢想，所以逐漸發展成目前嶄新的經營型態。以前大多數賣家都是小小的個體戶，採買商品、文案、拍照、寄貨等全部自己來，現在越來越多是幾個人合夥，流程分工並互相支援。

雅虎奇摩曾針對成交金額居前 20% 的賣家，進行過一次抽樣問卷調查(抽樣 800 人、有效回收問卷數 560 份)。結果發現，有高達 13% 的人是以網路拍賣為其全職工作，至於一般上班族拿來當副業的佔 24%，家庭主婦佔 13%，其他

如自營商則約佔 20%。以前想要在網路拍賣上獲利，只要懂點絕竅加上努力不懈，便不難達成。但是現在要面對無數的競爭對手，並面對商品絕對不會只有你有的難題，利潤空間越來越被壓縮，以前的暴利或超額利潤，現在已不復見了。要在網路拍賣獲利，可能就是賺平穩的基本利潤。賣家和賣家的競爭，從價格戰擴大到增值服務戰，也就是說一個真正體驗商業經濟整體服務的時代來臨了，比價格之外還要比服務。在拍賣平台上良好的社群關係一旦建立，客戶回訪的機率變會大大增加，因此行銷的挑戰一天比一天多，顧客要求的附加服務也越來越多[1]。

要讓客戶有興趣競標，比的可能是你的增值服務和創新行銷，當然也別忘了要有誘人的促銷活動，這些都能讓你的網路拍賣更有競爭力。有鑒於網路的服務跟技術越來越發達，各項服務的內涵越來越競爭，也許今天提供一種跨平台的策略結盟便會有創新的發展空間，例如：開放線上信用卡刷卡購物服務，或是使用信用卡紅利回饋網路購物等等…。

### 1-1.2 網路交易拍賣平台的環境趨勢

台灣上網人口突破 1,300 萬人，幾乎每 2 人就有 1 人使用網路。曾使用線上購物或競標拍賣的人口，目前高達 210

萬人，較前年成長近 3 倍。歷經 2000 年崩盤前的瘋狂階段、2000~2004 年的沈潛階段，網際網路已經進入第三波的復甦階段，線上開店、競標拍賣正高速起飛！這一波新投入者，在網路泡沫後調整經營心態，找到新利基與商業模式。無論個人或企業，網路開店已經走向國際！有的「個體戶」把台灣出品的外國歌手相關物件賣到國外，能賺一倍以上價差；有的企業上網賣公仔，每年營收 9,000 萬，連歐洲廠商也在網路下單[2]。

網路開店與競標拍賣的熱潮，更引爆 3 個新現象：「跨國交易」、「清庫存、賺價差」，以及「延伸性服務」的興起。然而，網路開店有賺錢機會，也有尚未釐清的模糊地帶，買賣爭議在所難免；消費者「貪小便宜、一窩蜂」的心態，容易落入網路詐騙的圈套。個人與企業上網開店怎麼做？買家與賣家交易流程怎麼走？跨國交易如何靠網路全球賺透透？..... 看來，曾被視為已泡沫化的 B2C（企業對終端消費者）電子商務，又再現熱潮。上網人口遽增，消費者也逐步改變習慣，開始進行上網購物，電子商務的市場確實成形[2]。

數位時代的規模經濟，是取決於使用者的人口。愈多消費者調整消費習慣，增加網路上的消費行為，就有愈多的銷



貨空間產生。歷經網路泡沫化，新投入者在心態上有很大的調整，不再有做大眾市場的妄想。這時候進場的，多是考慮在利基市場進行拚戰。這樣的質變，也帶來更多新的商業模式。網路開店與競標拍賣的熱潮之下，現在已經出現3個新現象：「跨國交易」、「清庫存賺價差」，和「延伸性服務」。這些利基型的商業模式，是這一波網路商機的新趨勢，網路開店與競標拍賣的熱潮下，一些周邊服務也應運而生。例如代購或代標商品、代登物件、甚至是交易歷史資料分析的服務，都紛紛出現。由於語文隔閡，台灣網站就出現代為標購日本拍賣網站商品的服務。在網路拍賣市場已經相當成熟的美國，有更多延伸性的服務。美國 eBay 甚至已出現為拍賣品價格做歷史資料分析的特殊服務；買、賣家只要支付適當費用，就能以資料分析，決定買入或賣出價格。此外，有些 eBay 賣家由於經常進行交易，需要大量的拍賣品包裝材料，自己靈機一動，反而變成專門販賣拍賣品包裝材料的賣家；買家只要說明拍賣物品的大小，他就能提供適當的包裝材料。在台灣，相信隨著網路開店與競標拍賣的市場更趨成熟，這些延伸性的服務也會更盛行。然而當平台後端的延伸性服務不停的更新，提供更便利的服務之時，我們發現…唯一沒有改變的…使用者的服務平台，始終停留在最初的買賣

討論區，單方提供訊息及使用者等待新訊息的單一網頁，在資訊的獲得與便利性卻是非常的缺乏[2]。

### 1-1.3 網路交易平台的現狀

現時的拍賣網站功能十分強大，如：yahoo 拍賣、露天拍賣，交易都十分活絡。美中不足的是使用者對使用者的聯繫非常緩慢，有時甚至耗費一整天時間等待，非常沒有效率。而在各式各樣的拍賣平台中，並沒有一個非常完善的遊戲買賣交換平台出現，大家還是只能依賴一些遊戲討論版，像是巴哈姆特討論版或是遊戲基地討論版，或是上述之拍賣平台，來完成遊戲買賣交換。而這類的討論版十分具有影響力，吸引成千上萬電玩迷流連，內容包羅萬象，從討論區、商城、遊戲介紹、遊戲評論、二手討論區應有盡有。想要買賣遊戲的話只要到二手討論區新增文章即可，然後等待，終究還是脫離不了上述的缺點，無法即時聯繫。然而在這已經發展到某一階段成熟的環境下，如何從本身的網站、商品、服務中，去研究出更有利於其他人不同的經營方式，我們可從當中去加以改進。

## 第二節 研究動機與目的

### 1-2.1 網路以物易物的故事

在網路間相傳的網路新聞，有一個外國的男生用一個迴紋針慢慢的交易...到了最後換到一間房子。我們從網路的使用的行為看來，這是非常誘人的例子，也是很好的網路行銷廣告。這則真實的故事，看來雖然誘人，但是背後卻是花了2年的時間，還有不斷的投注心力在網路上分享需求以及認識其他交易者。且這個實例只用部落格來表達其需求，雖然交易的物品不一定是等值的商品，但是卻是一個供需需求表達的最好實例。從普通不起眼的迴紋針，到一棟價值約5萬加幣的房子。據本人說法，若是有更好的交易或交換平台，可能不用到2年才能達到這個目標，可能只需要8個月左右。因其受訪之時的語出驚人，所以才出現今日許多的交換網站。究竟以物易物有何魅力，為何可以創造每日龐大的瀏覽數及會員數，其中價值以及所創造的商機，更是讓人吸引的地方，我們也將由交換的機制中，來吸引更多商機與話題。

## 1-2.2 網路交易拍賣平台所衍生的問題

網路拍賣發展雖然蓬勃不過尚有些漏洞。基本上，買家賣家因為見不到面，不是當面銀貨兩訖，所以雙方的交易是建立在對彼此的信任上。買家把錢匯給賣家了，賣家就要依照約定把貨品寄出給買家。除了交易的方式與以往傳統的市場買賣方式不同，當買賣雙方無法現場面對面交易所衍生的問題，我們期盼能夠從雙方的溝通方面來解決。若是能夠改善這當中的溝通方式，除了能夠免去雙方買賣條件的不利，也不用擔心個人資料的洩露，從溝通的衍生性服務做到最完整。如何從既有的使用者平台中去研究出最好的改進方式，以及使用的便利性，操作的方便性，傳輸的安全性，這些是我們此篇論文的研究重點。如何改進在買賣雙方交易的過程中，達到訊息傳遞的重點，讓雙方能夠在最短的時間內達成共識，改善溝通上的盲點，既能節省時間與網路資源，又能快速成交達到快速服務資訊透明的效果，是我們希望能夠改進跟探討的重點。

### 1-2.3 交易平台的溝通改進與應用

SSL (Secure Socket Layer) 是網路線上交易普遍採用、使用上較方便的加解密安全控管機制，可確保機密資料傳輸上的安全性，例如密碼、帳號、交易資料等。當客戶使用 SSL 進入網路銀行查詢或交易時，只要輸入 ID、密碼及使用者代號便可以執行相關作業，操作簡便。SSL 是以提供一個安全的傳輸路徑為主，只要協定中的安全機制不被壞，基本上在此路徑上傳送資料都是安全的，雖然 SSL 大多用在所謂線上交易，但其實任何需要傳輸的線上操作方式，皆可以使用 SSL 的安全性機制。SSL 安控機制因交易面之系統安全設計不具備「無法否認傳遞訊息」、「無法否認接收訊息」、「訊息來源辨識」的功能，為了確保客戶權益及降低交易或是個人資料外露的風險。我們將於本專題的研究中，嘗試加入 SSL 的安全機制做處理，或是從其他的方向加以研究如何改善網路交易的安全機制，並實際操作模擬。

## 第三節 研究範圍與報告書架構

### 1-3.1 研究範圍

本篇報告除了第二章描述相關程式之應用外，研究範圍平台首先要參考 yahoo 拍賣為主，各國其他交易交換平台為輔。從當中的經營方式，網站內容，服務模式參考其優缺點，並以現有之網路服務平台其他安全性功能測試，包括本身設立之測試相關網站。還有主要以通訊功能的技術跟服務發展方面擷取，主要範圍如下：

1. 相關網站:交易拍賣網站的平台研究。
2. SSL 安全機制相關程式及資料收集與判斷。
3. 交易買賣行為模式的探討:各大交易平台的資料收集與評估。

### 1-3.2 報告書架構

本篇報告第一章為探討動機以及序論，包括研究網路交易的原由、模式，以及買賣行為探討。

第二章為系統的介紹，以及所應用的程式相關的說明，還有本篇報告所使用之技術。

第三章為針對本篇報告所架設之相關類似交易拍賣平台的網站，我們將提供基本的功能於程式的測試及操作。

第四章將我們所建置的拍賣平台做安全性的實際應用，除了性能及功能的測試，並將實際的功能及成效做評估，以判斷網路交易的完整的安全性。

第五章我們提供實際使用測試的數據，以及判斷當中的優缺點來評估程式的可用度、實用性，也針對所衍生的問題加以探討。另外還有改進的方式，以及未來的應用還有服務的加值化探討。

## 第二章 系統介紹

### 第一節 C#

#### 2-1.1 C#介紹

C#是個由 Microsoft 所創造的語言，並且交由 ECMA ( European Computer Manufacturer' s Association，歐洲電腦製造商協會 ) 達成標準化。這個新語言是 Microsoft 一個由 Anders Hejlsberg 所領導的團隊所創造的。有趣的是，Hejlsberg 是 Microsoft 的傑出工程師，他還創造了其他的產品與語言，包括：Borland Turbo C++和 Borland Delphi。在設計 C#時，他著眼於兼採各家現存語言之長，並且力圖改進以求做得更好。

C#是個具有彈性的程式語言。如同所有程式語言一般，C#可以用來產生各種應用程式。在使用 C#時只會侷限於想像力，但這個語言並沒有對所能做的事做侷限。C#已經用於多種專案，諸如：動態 Web、開發工具，甚至還有編譯器。C#被設計為一個物件導向程式 ( object-oriented programming ( OOP ) ) 語言。其它程式語言也有包括物件導向的特徵，但卻少有全物件導向的程式語言。



## 2-1.2 C#特色

### (1) 簡單、現代化

C#刪除了語言（例如：Java 和 C++）中的某些複雜和容易犯錯之處，包括移除巨集（macro）、模版（template）、多重繼承（multiple inheritance）和虛擬基底類別（virtual base class）。這是所有會對 C++ 開發者造成困擾或可能發生問題的部分。因為是以 C 和 C++ 為基礎，所以 C# 是簡單的。如果熟悉 C 和 C++ 或是 Java 會發現 C# 在許多方面都非常地類似。敘述、表示、運算子（operator）和許多方面都直接沿襲自 C 和 C++，但做了些改善使得使用這個語言更為簡單。部分改善包含去除冗餘之處，其它改善的地方包括了額外的語法變更。

為什麼可以構成現代化的語言？譬如例外處理（exception handling）、垃圾收集（garbage collection）、可擴充資料型態（extensible data type）和程式碼安全性（code security）等特徵，一般被認為是現代化語言所應具備之特徵。C# 具備以上所有特徵。

### (2) 物件導向

物件導向語言的精隨再於封裝（encapsulation）、繼承（inheritance）和多型（polymorphism），C# 則以上

兼備。封裝是將功能置入單一套件（ package ）中。繼承是以結構化的方式，將現有的程式碼與功能擴展為新的程式與套件。多型則為視所需完成的事件提供功能之能力。

### (3) 一步到位的程式設計

在 C# 語言中，與宣告相關的內容被侷限在設定宣告區域，而不會分散到不同的原始檔案中，或散落在一個原始檔中的好幾個地方。至於型別（ Type ）也不需要標頭（ Header ）檔或介面定義語言（ Interface Definition Language，簡稱 IDL ）檔中做額外的宣告；內容涵式（ property ）所提供的 get / set 存取方法，邏輯上會被聚集在一起，而文件說明則直接內嵌在宣告之中。此外，由於宣告的順序與型別無關，因此並不需要在另一個「型別」中做額外的「虛設宣告」（ stub declaration ）。

### (4) 版本控制

C# 語言所提供的許多特色，譬如：明確的介面實作，隱藏所繼承的成員以及唯獨的修飾字，有助於新版元件跟與其相關的舊版元件之間共同運作。

### (5) 型別安全與一致化型別系統（ Type safety and a unified type system ）

C# 是個型別安全（ type-safe ）的語言，也就是說它保

證一個變動只能經由其被指定的型別來加以存取。採取這個封裝方式，目的在鼓勵優良的程式設計以及降低潛在的程式錯誤或安全漏洞（如果一個變數可以意外地或故意地複寫另一個變數的內容，很容易導致安全上的違規與破壞）。C#語言的所有型別（包括最基本的型別）接生字單一的基礎型別，並且提供了「一致的型別系統」（unified type system）。這表示所有的型別 structs、interfaces、delegates、enums 以及 arrays 共享相同的基本功能，就像這些能力：轉換成字串、序列化（serialized）或儲存在「集合」（collection）中。

(6) 自動與人工的記憶體管理（Automatic and memory management）

C#語言的記憶體管理會在「執行階段」自動進行。這可避免程式設計師任意配置物件，因此可排除諸如「懸空指標」（dangling pointer）、「記憶體漏洞」（memory leak）以及「循環參照」（circular reference）等問題。然而，C#語言並未因此便捨棄「指標」不用，它只是巧妙的讓「指標」在大部分的程式設計中變得不重要而已。針對「強調效能導向」（performance critical hotspot）以及程式的「互通性」（interoperability）等應用，還是會應用到

指標的功能，但這只允許在一個需要高度安全現才能執行的 unsafe 程式區塊中使用。

#### (7) CLR 的運用

C# 語言相較一般程式語言（特別是傳統的編譯式語言，譬如 C++）的一大優勢，在於它能夠與 .NET 的 CLR 緊密的結合。C# 在許多方面與 CLR 是一至的，特別是指它的型別系統、記憶體管理模式，以及例外狀況的處理機制。

#### (8) 模組化

C# 程式碼能夠（且應該）以被稱為類別（class）的區塊來撰寫，類別中包含了被稱為成員方法（member method）的常式（routine）。這些類別和方法可以在其它的應用程式或是程式中重複使用。藉著將某些資訊放到類別和方法中，可以產生再利用的程式碼。

### 2-1.3 C# 之於其它程式語言

Microsoft 聲稱 C# 具備像 C++ 一樣的強大能力，卻又像 Visual Basic 一樣簡單。C# 確實擁有強大的能力，但真的像 Visual Basic 那樣簡單嗎？它也許不像 Visual Basic 6 那樣簡單，但和從頭重新撰寫過的 Visual Basic .NET（第 7 版）一樣簡單。所以，最後的結論是 Visual Basic 實際上

並沒有比寫 C# 程式要來得簡單。事實上，確實可以使用 C#，以較少的程式碼來撰寫任何程式。

儘管 C# 刪除了 C++ 造成程式設計師困擾的某些特徵。但實際上並沒有損失任何的能力或功能。在 C++ 中很容易產生某些程式設計錯誤，而這些錯誤在 C# 中則完全被免除掉。另一個備受矚目的語言就是 Java。Java，如同 C++ 和 C# 一般，是建構於 C 之上，如果決定遲些要去學 Java，會發現很多在 C# 中學到的知識會派上用場[1]。

## 第二節 SQL Server 2005

### 2-2.1 SQL Server 2005 簡介

SQL Server 2005 是一個功能完備的資料庫平台，利用整合式商業智慧工具，提供企業級資料管理功能。SQL Server 2005 資料庫引擎提供更安全、可靠的儲存環境給關聯式和結構式資料，讓您能夠建置並管理用於企業的高可用性、高效能資料應用程式。SQL Server 2005 資料引擎是企業資料管理解決方案的核​​心。SQL Server 2005 還結合了最佳的分析、報表、整合和通知功能，以便企業建置及部署高成本效益的解決方案。透過計分卡、儀表板、Web 服務和行動裝置，幫助團隊將資料帶到企業每一個角落。與 Microsoft Visual Studio、Microsoft Office System 和一套新開發工具（包括 Business Intelligence Development Studio）的密切整合，讓 SQL Server 2005 與眾不同。不論您是開發人員、資料庫管理員、資訊工作者或決策者，SQL Server 2005 均提供創新的解決方案，幫助您從資料中獲得更多價值。下圖 2-1 說明 SQL Server 2005 的核心元件。

# SQL Server

The comprehensive, integrated data platform



圖 2-1、SQL Server 2005 元件圖

## 2-2.2 SQL Server 2005 伺服器架構

SQL Server 2005 是個採主從架構的關聯式資料庫系統。隨著 1980 年代個人電腦的興起，以及個人電腦的運算能力快速的成長，使得主從式軟體架構也漸為大眾接受並採行之，而逐漸取代過去的主機-終端機式或 PC 網路上檔案伺服器(單純分享檔案資源)的運作方式。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。

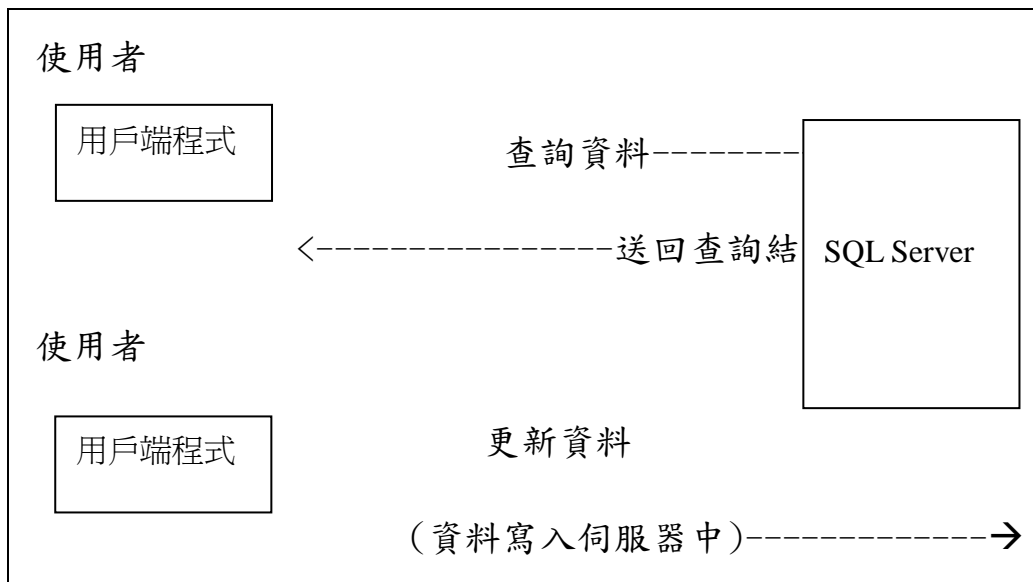


圖 2-2、SQL 路徑圖

不過以上這種方式，只是比較傳統的兩層式主從式架構，近年來三層式或多層式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(Internet Information Services)伺服器、ASP.NET 技術或其他伺服器產品，即可建立多層的主從運算模式。

為了讓用戶端(應用程式)能順利查詢／異動資料庫中的資料，SQL Server 提供了數種不同的介面，讓用戶端可視其需求，透過這些介面來存取資料庫中的資料，這些介面包括：

(1) OLE DB：

OLE DB 是微軟公司所發展的一組用來存取各種資料的 COM(Component Object Model)元件。使用 OLE DB 作為資料存取介面，應用程式就能透過一致的方式從各



種不同資料來源取得所需的資料。

(2)ODBC：

ODBC(Open Data Base Connectivity)是根據 SQL 語言所發展出來的資料庫存取介面,如其名稱中的 Open(開放)所指,其目的是希望應用程式能利用簡單的 SQL 查詢,即能從各種不同的資料庫系統取得資料。

### 2-2.3 SQL Server 2005 組成架構

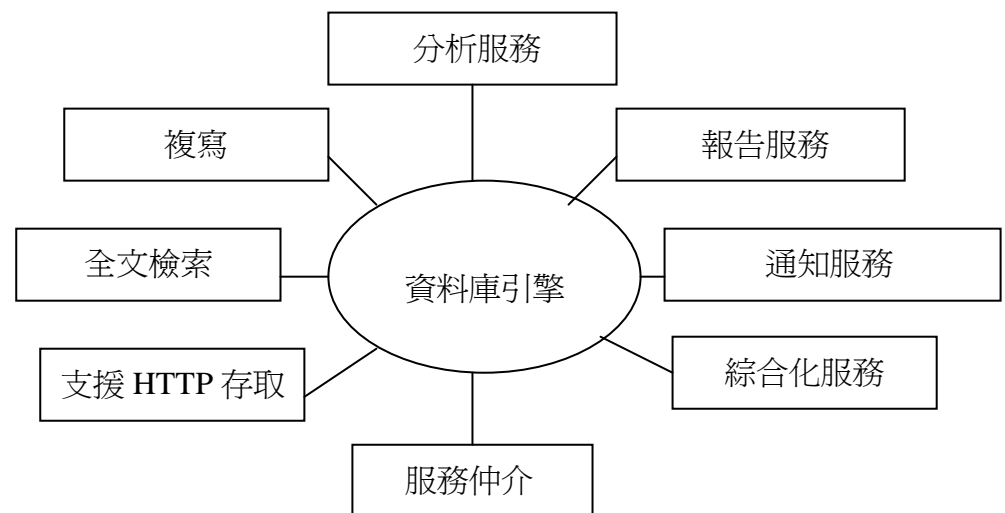


圖 2-3、SQL 組成架構圖

- 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。
- 分析服務：這是微軟的商業智慧解決方案之一，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。

- 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(Internet Information Services)，就可以讓使用者透過 SOAP(Simple Object Access Protocol)呼叫取資料庫中的資料。

- 服務仲介：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。

- 報告服務：這個服務必須配合 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。

- 通知服務：此服務可以讓管理者使用 XML(Extensible Markup Language)定義各種事件(Event)，之後事件發生時，SQL Server 便會立即通知訂閱者並進行相關處理工作。

- 綜合化服務：此服務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[2]。

## 第三節 ASP.NET

### 2-3.1 ASP.NET 與 ASP 的差異

ASP.NET 優於 ASP 之處，不是 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，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使效能大幅提升。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。

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。

.NET Framework 是新的運算平臺，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。.NET 不是全新的作業系統，開發端只要安裝 .NET Framework Component Update，即可將 Windows 升級成支援 .NET Framework 的工作平臺，伺服器則必須是 Windows 2000 Server 或 Windows XP Professional。.NET Framework 有兩個主要的元件：Common Language Runtime 和 .NET Framework 類別庫。Common Language Runtime 則是 .NET Framework 的基礎，可視為程式執行期間管理程式碼的代理程式，提供記憶體的管理、執行緒的執行、程式碼的執行、程式碼安全驗證及編譯等服務。

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。

## 2-3.2 ASP.NET 應用程式分為兩種類型

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(Personal Digital Assistant)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(HyperText Markup Language)、XML(eXtensible Markup Language)、WML(Wireless Markup Language)和 ECMAScript(European Computer Manufacturer's Association，歐洲電腦製造商協會)。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 網頁裏面，若用 Visual Studio .NET 開發 ASP.NET 網頁，會自動將程式碼建在不同的檔案中，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。

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 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，再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。

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，將經常存取的整分或部分網頁暫存一段時間，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。此外，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。

### 2-3.3 ASP.NET 網頁管理方式

HTTP 是無狀態（stateless）的通訊協定，它不會自動識別各個要求是否來自相同的使用者，也無法知道使用者目前是否仍在檢視網頁。每一次要求的往返，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。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 Web 程式設計的限制，ASP.NET 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。應用程式的狀態可儲存全域性的資料，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，若要保留大型暫時性資料區塊，使用 ASP.NET 快取是較佳的方案。由於 Common Language Runtime 所設計的自訂物件沒有執行緒的限制，如果沒有撰寫同步處理程式，可能造成死結、競爭和存取違規的問題。此外，存取全域變數可能會影響延展性及安全性，而且如果系統當機或處理序重新啟動，資料將會遺失，所以若要避免重要資料遺失，應儲存在資料庫或其他持久性儲存設備。

工作階段 (session) 狀態管理功能，能自動識別單一用戶端的要求，在伺服器上儲存 session 資料，可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，例如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

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。開發人員常需要在程式中撰寫變數或判斷提示 (Assert) 來偵錯，程式部署時又必須移除偵錯陳述式，避免程式執行時，顯示奇怪的訊息。ASP.NET 引進新功能，允許在網頁或應用程式中啟用追蹤功能—@Page 指示詞的 Trace 屬性，可直接在程式碼中撰寫偵錯陳述式，程式完成時不用移除他們，只要把 Trace 屬性設定改為 False，即停用網頁追蹤功能。

#### 2-3.4 ASP.NET 授權、驗證和模擬網站安全機制

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，在他們自己的帳戶之下執行，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。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。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，逐步將 ASP 程式移轉到 ASP.NET 才是長遠的做法。

## 2-3.5 ASP.NET 3.5

ASP.NET 上實現了 MVC Patter(Model-View-Controller) 後，開發者也可輕易地在 Visual Studio 中建構 MVC 架構的網站。

- ASP.NET Dynamic Data：Dynamic Data 則是提供了快速建構鷹架 (Scaffolding) 的功能，讓你迅速建立可存取資料庫的網頁。

- ASP.NET AJAX：ASP.NET AJAX 的 Controls 很早就出現外掛了，不過在這次的 Extensions 中，支援控制瀏覽歷史 (Browser History) 的功能，也就是瀏覽器的上一頁、下一頁將可在 ASP.NET AJAX 正常運作，不影響使用體驗。

- ADO.NET Entity Framework：這是在 ADO.NET 上實現 ORM (Object-relationsla mapping) 的技術，ORM 可將資料表轉為物件的類別，使得開發人員在對資料庫進行存取的時候只需要使用物件導向的語法、而無須撰寫 SQL 語法，維持開發過程中物件觀念的一致性。

- ADO.NET Data Services：這就是 ADO.NET 實現 REST (Representational State Transfer) 的機制，未來我們也可以透過它在 ASP.NET 中開發 RESTful 的網站。

Silverlight Controls for ASP.NET：很顯然地，ASP.NET

MVC & Dynamic Data 與 Rails 之類的 Web Framework 很像，一樣都是實現 MVC pattern、Scaffolding、ORM、REST 等概念，相較於其他已經發展許久的 Framework，微軟在這塊是比較晚進場的，對很多早在 ASP.NET 打滾多年的開發人員而言，若這些套件並不是真的這麼好用，或許短期內並不會有太多的迴響。

從表面上看來，此延伸套件可有效地提昇 ASP.NET 網站開發人員的生產力，ASP.NET MVC 可以讓開發團隊進行更有效率地分工；ASP.NET Dynamic Data 可以讓開發者快速建構網頁的雛型；ASP.NET AJAX 可以快速建立各種 AJAX 的瀏覽方式；ADO.NET Entity Framework 則是讓開發人員可以使用物件語法存取資料庫，使得開發過程大部分的時間無須撰寫 SQL 語法[3]。

## 第三章 拍賣網站平台及服務

### 第一節 網站系統分析

#### 3-1.1 網站的類型與架構

由於我們所要架設的網站為提供交換或是買賣行為之網站，首先針對使用者需求以及所需之服務判斷系統架構。網站主要以交換功能為主，交易拍賣功能為輔，故以建立足夠的安全會員機制的資料庫，以及完整的搜尋與交易問答機制。

對於我們將要建置的網站，我們在參考其他相似類型之網站之後決定由系統中增加「交換功能機制」的說明，以加強我們的主題，並且設計個人化的風格頁面，以更能親近使用者。尤以能夠預留日後增加增值化功能，或是其他交易服務、例:線上小額交易... 付款等的機制。故在架設之初我們先由主要功能建立完整需求為主!

專題網站系統製作流程大致如圖 3-1 所示：

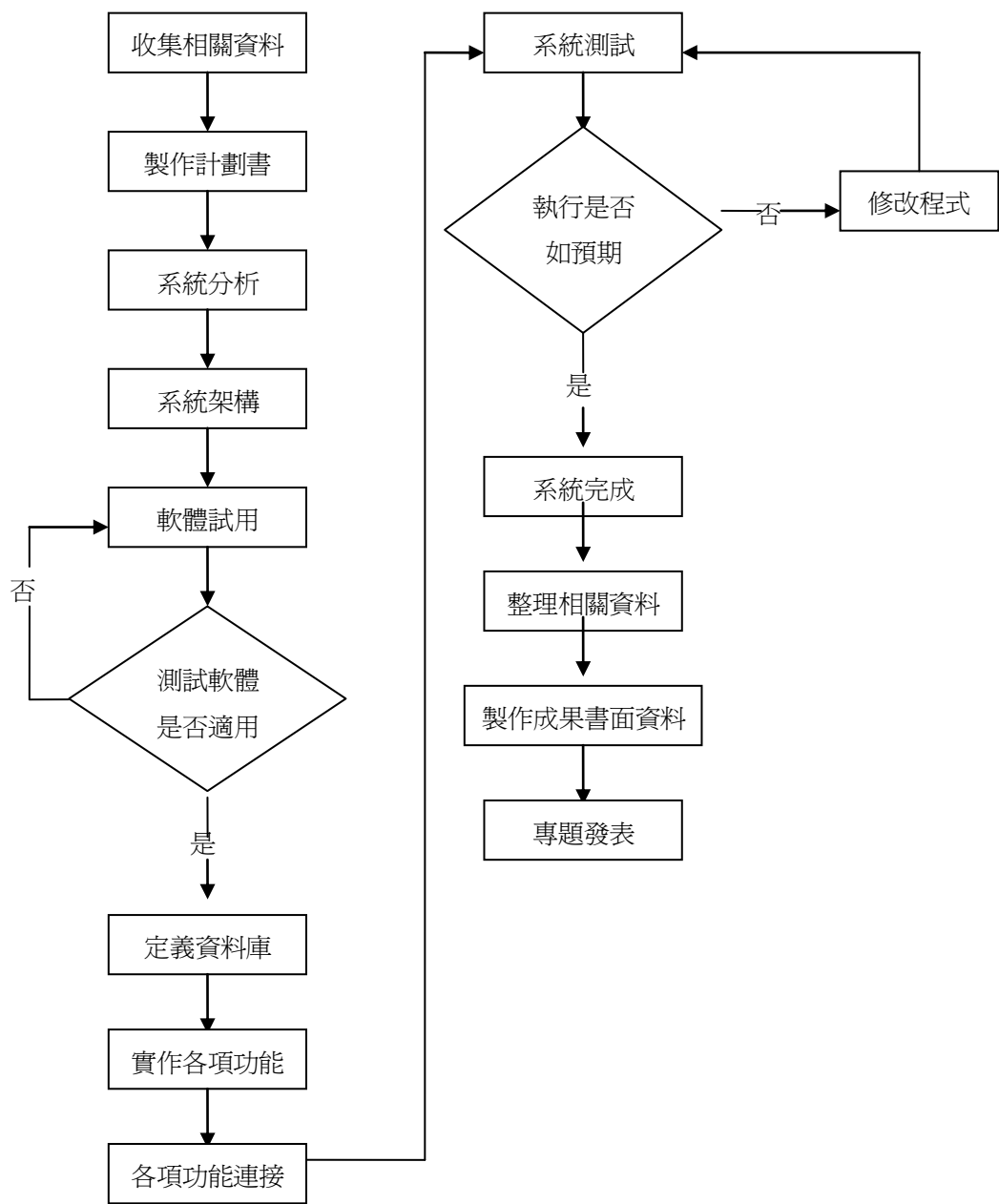


圖 3-1、專題網站製作流程圖

### 3-1.2 網站系統分析

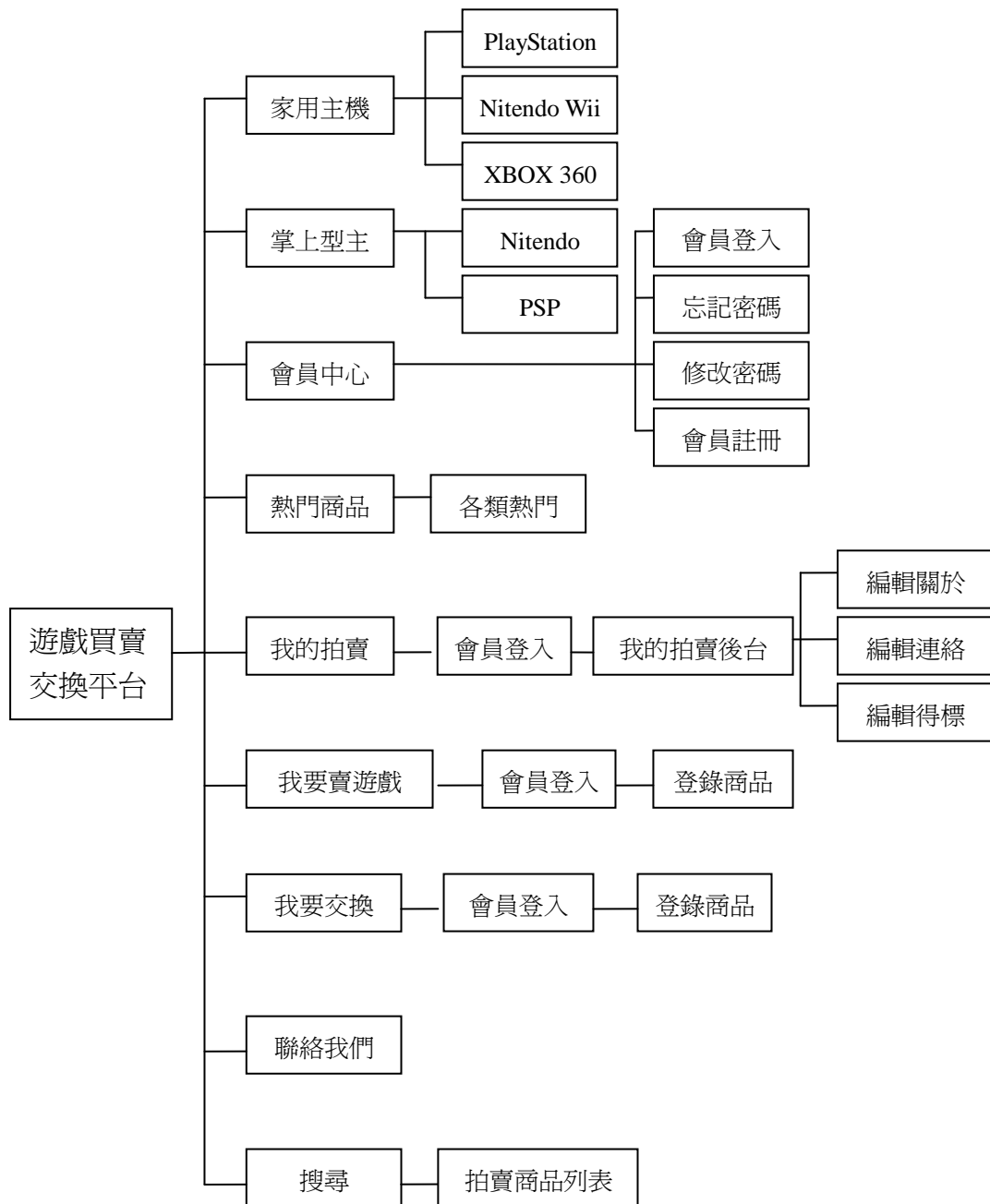


圖 3-2、專題網站系統架構圖

如圖 3-2 所示，我們將網站的初步架構圖形成後，接下來就依架構的方向依序設計我們將要建立的服務跟功能！在



網站功能選項部份加上搜尋功能，大致上分為九大項。

- 家用主機：主要有目前市面上熱門的 PlayStation3、Wii、XBOX 360 等三款家用遊戲主機官網之最新消息。
- 掌上型主機：主要有目前市面上熱門的 NDS 與 PSP 等兩款掌上型主機官網之最新消息。
- 會員中心：主要提供網站會員進行個人資料與密碼的修改，若會員忘記帳號之密碼也可在此進行索取新密碼之動作，另外新使用使也可在此進行加入會員之註冊。
- 熱門商品：此處有家用主機及掌上型主機之熱門遊戲介紹，每種主機類型之熱門遊戲各有獨立頁面。
- 我的拍賣：會員登入後可在此處連結至會員拍賣之後台，可進而編輯賣家關於我、聯絡我、及系統自動寄發給買家之得標通知信。
- 我要賣遊戲：會員登入後可在此進行登錄欲拍賣之商品。
- 我要交換：會員登入後可在此進行登錄欲交換之商品。
- 聯絡我們：本網站製作小組之聯絡資訊。
- 搜尋：，使用者可在此搜尋使用者欲競標購買或交換遊戲。

## 第二節 拍賣網站

### 3-2.1 拍賣網站功能介紹

在網站已經架設好之後，我們也參考其他的網站平台，為了符合主題及設計特色化介面，我們參考了 TV 遊戲專用的手把，設計了以下的畫面如圖 3-3：



圖 3-3、網站首頁圖

網站的製作技術主要是使用 Dreamweaver、Flash、Swift 3D 等，網站設計概念只要是想讓使用者一進入網站就能充分感受到遊戲的輕鬆感。上方 LOGO 圖層的外觀主要是以任天

堂 NDSL 主機之圖樣來包裹網站的 LOGO，下方選項則是使用下拉式選單。網站的圖層由上至下分別為最上方 LOGO 圖層、網站選項圖層、內容圖層再分為左邊功能區圖層，裡面包括搜尋、會員登入、最新遊戲消息、各遊戲公司官網之連結等功能，右方則為各家用型主機之最新遊戲連結等。

首先網站有每種遊戲主機最新消息之頁面，使用者可透過這些頁面來了解各遊戲主機之最新消息。如圖 3-4 至圖 3-8 所示：

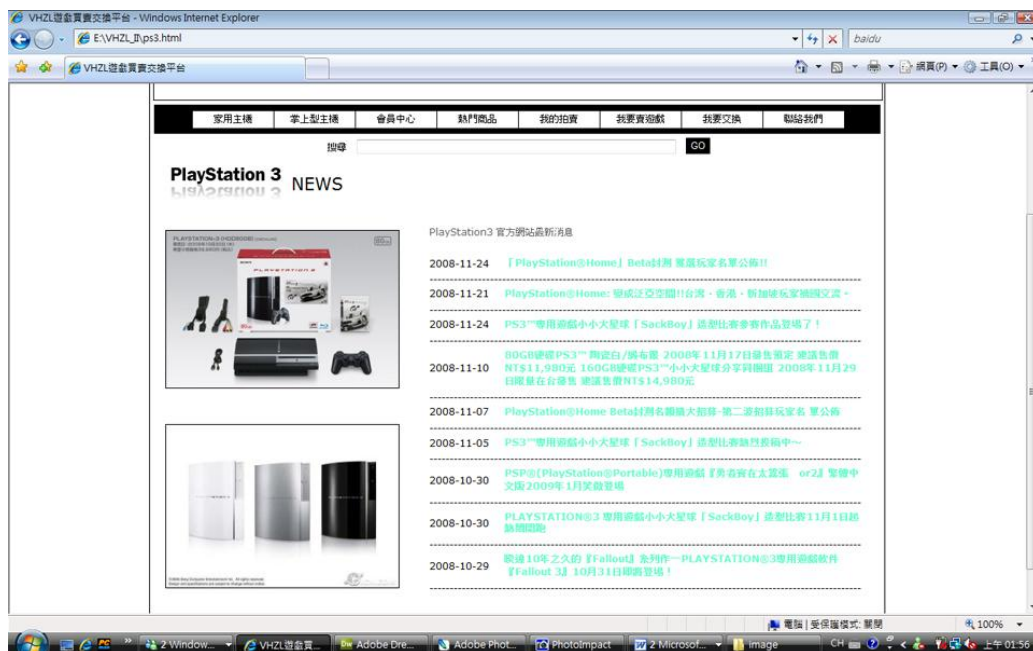


圖 3-4、家用主機 PS3 最新消息頁面



圖 3-5、家用主機 Wii 最新消息頁面

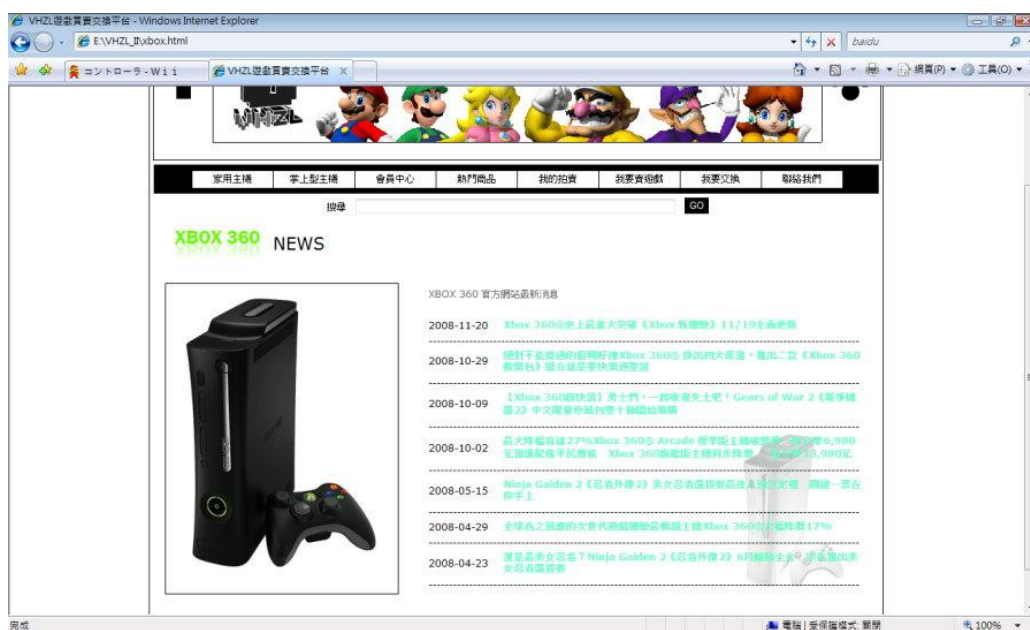


圖 3-6、家用主機 XBOX 最新消息頁面



圖 3-7、家用主機 NDSL 最新消息頁面

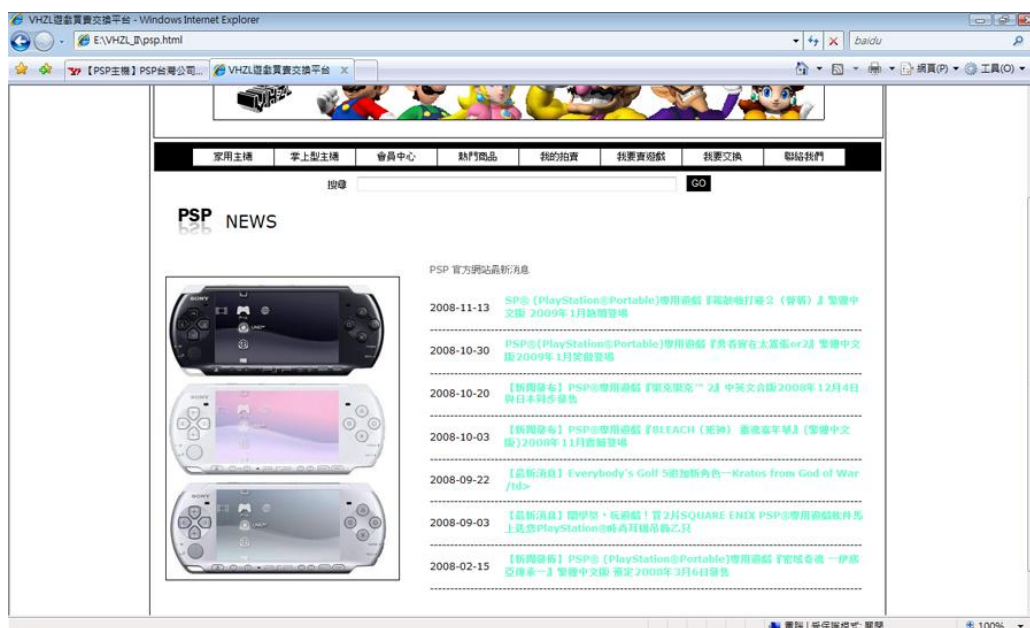


圖 3-8、掌上型主機 PSP 最新消息頁面

另外網站還有各遊戲主機之熱門商品，使用者可透過這些頁面來了解各遊戲主機之熱門商品。如圖 3-9、圖 3-10、

圖 3-11、圖 3-12、圖 3-13 所示：

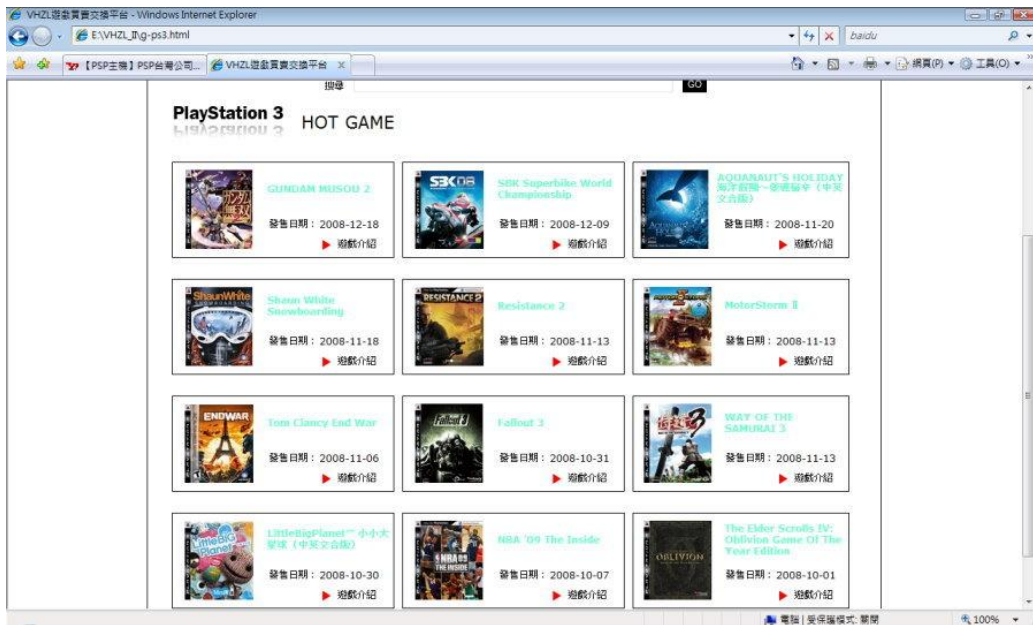


圖 3-9、PS3 之熱門商品頁面

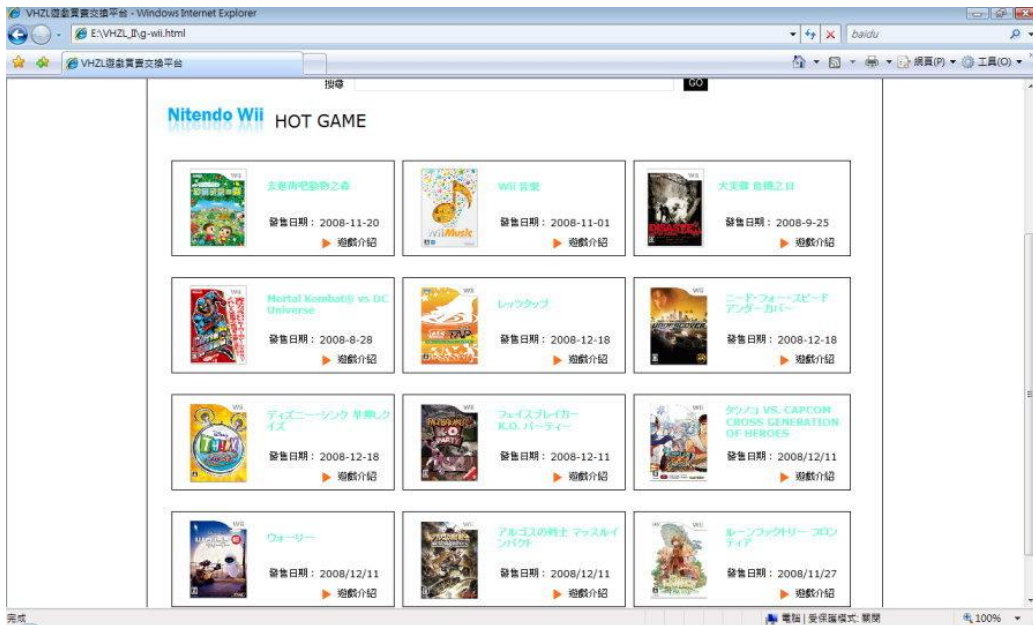


圖 3-10、Wii 之熱門商品頁面

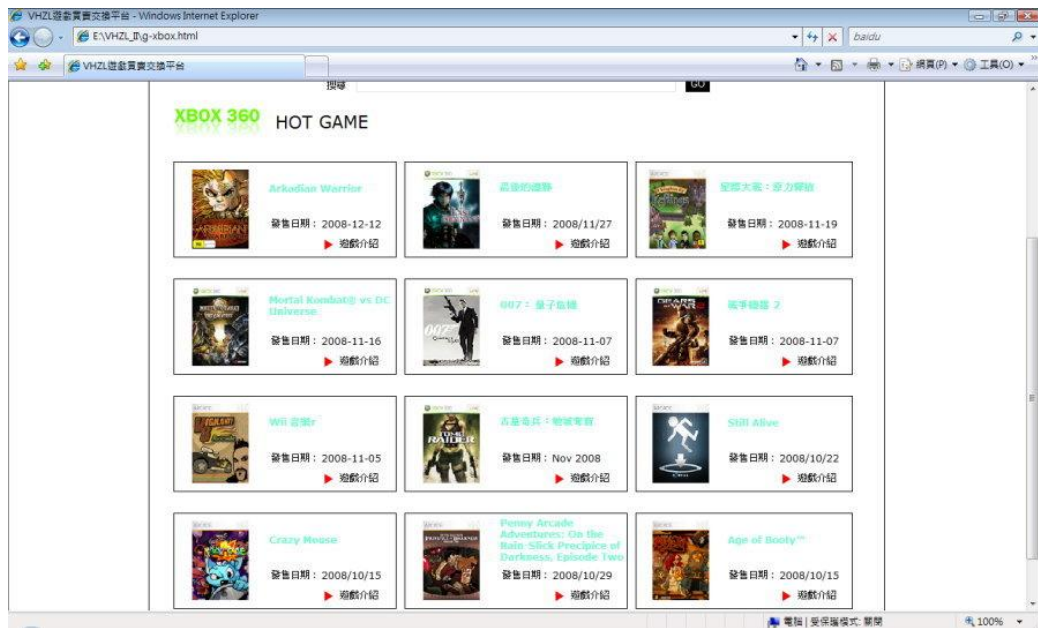


圖 3-11、XBOX 360 之熱門商品頁面

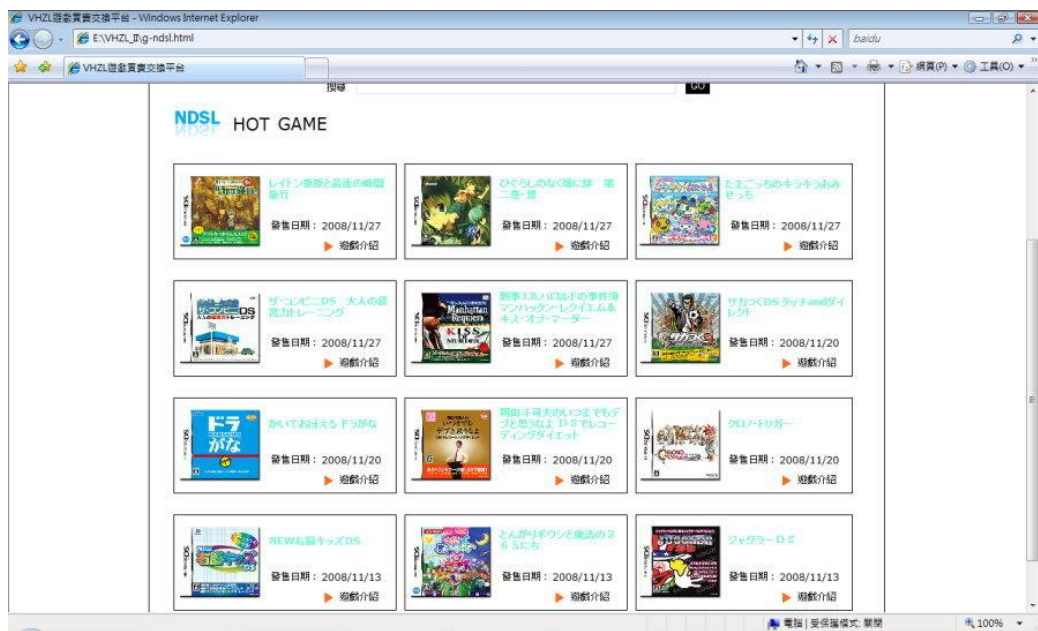


圖 3-12、NDSL 之熱門商品頁面

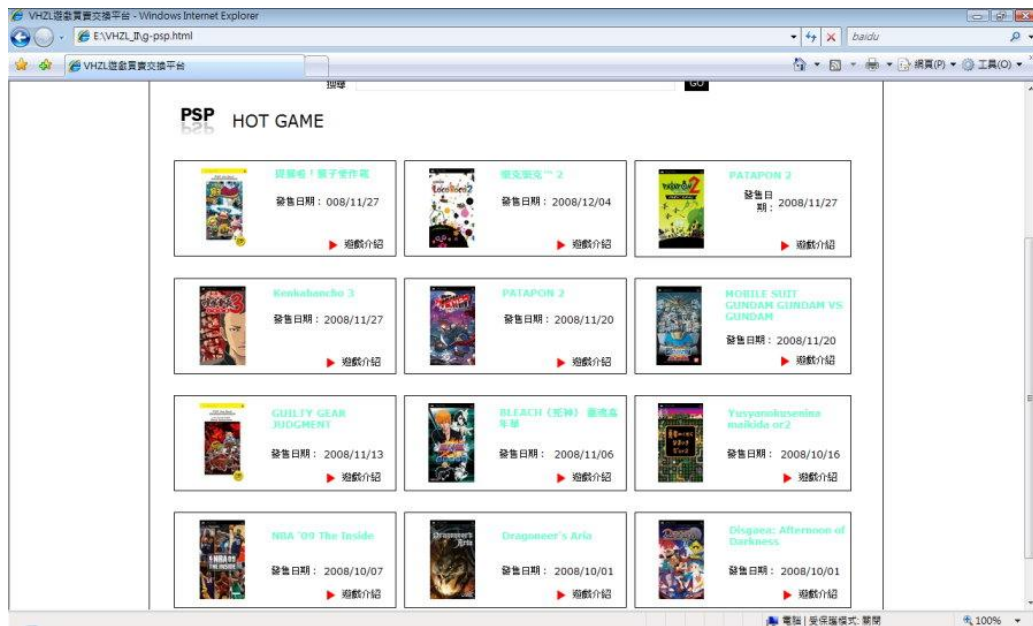


圖 3-13、PSP 之熱門商品頁面

熱門遊戲頁面裡除了有遊戲名稱、圖片與發行日期之外，我們還特別增加該熱門商品之遊戲介紹連結，使用者只要點選欲了解之遊戲介紹，會開啟一個新網頁視窗至該遊戲網站。如下圖 3-14 所示：





圖 3-14、熱門遊戲介紹之連結畫面

由於線上遊戲大多屬於非實體的商品，故商品的交易或是交換，我們將由使用者自行判斷其虛擬道具的價值，將可在商品的詢問跟交易中做溝通。也因為線上遊戲種類繁雜，故關於虛擬的商品我們將不多做分類。

在主分類中最主要的就是屬於實體的遊戲商品，例如當紅的 WII、X-BOX、NDSL...等，類似這些屬於 TV GAME 或掌上型電玩。由於屬可認定的交易商品，所以我們區分的類別較細，又分為遊戲主機及週邊配件。甚至每種遊戲的卡帶或光碟，不論虛擬道具或是實體商品，我們皆要求有要實物的的照片。若是有其他非價值性認定之商品，我們將由使用者自行溝通協調，所以我們的交換功能服務，大大的提升可完成

交易的機率跟可能。也提供買賣雙方更多變通的空間！

接下來是使用者將商品上架之過程，首先使用者必須先選擇欲拍賣或交換之商品所在的類別。如圖 3-15 所示：



圖 3-15、商品類別選擇之畫面

再來使用者可上傳商品之圖片。如圖 3-16 所示：



圖 3-16、商品圖片上傳之畫面

為了使用者瀏覽網站的方便，我們限制照片為 500KB 以下，設此限制為增加系統的效能，且方便傳輸，檔案的類型只能為 JPG 與 GIF 格式，在相片上傳後可先在上方預設的小視窗預覽。如圖 3-17 所示：



圖 3-17、商品圖片預覽之畫面

使用者上傳完圖片之後，需填寫商品內容，如交換、競標、拍賣的條件、時間、商品概況等，如圖 3-18 所示：



圖 3-18、填寫商品內容描述之畫面

完成以上之商品上架程序後，會出現商品上架成功之畫面。如圖 3-19 所示：



圖 3-19、商品上架成功之畫面

在確認資料無誤後上傳至網頁，會出現商品完整的圖案。並可在下方查詢有關於商品之問答，或是由買方提出問題與賣方的回答。我們在需要鍵入文字的畫面皆設有文字編輯器；刊登的天數，還有關於買賣雙方運費的負擔也會一併列出。最重要的我們在此有別於其他拍賣網站的設計是：我們的功能上有多增加『物品交換』的功能。除了既定的設定之外，賣方可以設定自己想要交換的物品，更可以增加文字的說明，使買方或瀏覽的使用者有更多的資訊，在此提供買賣雙方基本的交易交流。

除了商品圖片及商品內容描述外，另外我們也提供了賣家關於我、聯絡我及買家得標信的編輯。若賣家未編輯買家得標信，當買家得標時，系統所寄發出之買家得標信則會是空白的。編輯畫面如圖 3-20、圖 3-21、圖 3-22、圖 3-23、圖 3-24 所示：



圖 3-20、賣家關於我之編輯畫面



圖 3-21、賣家關於我編輯完成後之呈現畫面



圖 3-22、賣家聯絡我之編輯畫面





圖 3-23、賣家聯絡我編輯完成後之呈現畫面



圖 3-24、買家得標信之編輯畫面

接下來為賣家商品上架完成後呈現在買方的畫面。下方賣家可以輸入關於商品的資訊，直接購買價，定價，商品的交易方式，現金付款，ATM 轉帳，都有詳細的說明跟選擇，甚至付費方式，有別於其他拍賣網站尚須連結至其他網頁才有詳盡的資料，由於我們網站單純化，故所有資料皆清楚地呈現在畫面的頁面。而交易方式分別有拍賣及交換兩種，拍賣又分為有直接購買價及無直接購買價。如圖 3-25、圖 3-26、圖 3-27 所示：



圖 3-25、上架商品交易方式為交換之畫面



圖 3-26、上架商品交易方式為有直購價拍賣之畫面



圖 3-27、上架商品交易方式為無直購價拍賣之畫面

當進入商品畫面所顯示的畫面，我們能看到商品的照片、拍賣的出價跟出價次數、買賣的數量、競標的時間、商品的其他資料，以及賣方的基本資料。有別於其他網站不同

的地方，我們提供了供買賣雙方提出交換的項次，賣方可以提出可以交換等值的物品。買方可以表達欲交換的籌碼，買賣雙方可以有金錢之外的交易談判空間。有需求互相供應交流，也許不一定等值的商品，卻能夠透過交換及供應需求之共識而達成協議。

### 3-2.3 註冊及會員中心

接下來為一新使用者註冊申請會員之過程，首先需先註冊為本網站之會員。如圖 3-28 所示：



圖 3-28、會員申請畫面

主要註冊內容分別有最重要的帳號、密碼、密碼確認等，帳號密碼的字元長度限制為最多 16 個英文或數字，個人資料部份分別有真實姓名、性別、身分證字號、生日、郵遞區號、連絡信箱、地址、聯絡電話、行動電話等，另外我們也有提供外籍人士註冊使用。

在帳號部分須先通過帳號是否與他人帳號重複之檢測。

如圖 3-29、圖 3-30 所示：



圖 3-29、新申請帳號無人使用之畫面



圖 3-30、新申請帳號有他人已使用之畫面

在使用者設定帳號密碼時，我們也寫了來偵測該密碼強度之程式。如圖 3-31 所示：

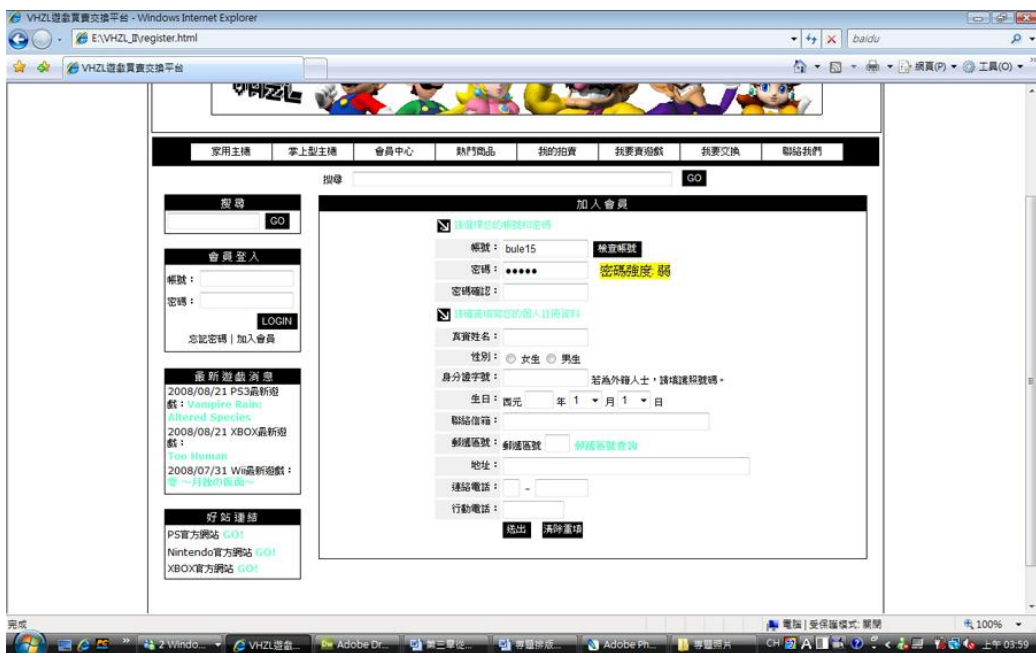


圖 3-31、密碼強度偵測之畫面

如使用者完成所有註冊須填寫之欄位後，完成註冊則會出現使用者完成註冊之畫面，內容會把使用者申請帳號、密碼及信箱顯示出來。如圖 3-32 所示：



圖 3-32、使用者完成註冊之畫面

在使用者完成註冊程序，身分證字號經由產生器確認無誤，系統會寄發註冊確認信。如圖 3-33 所示：

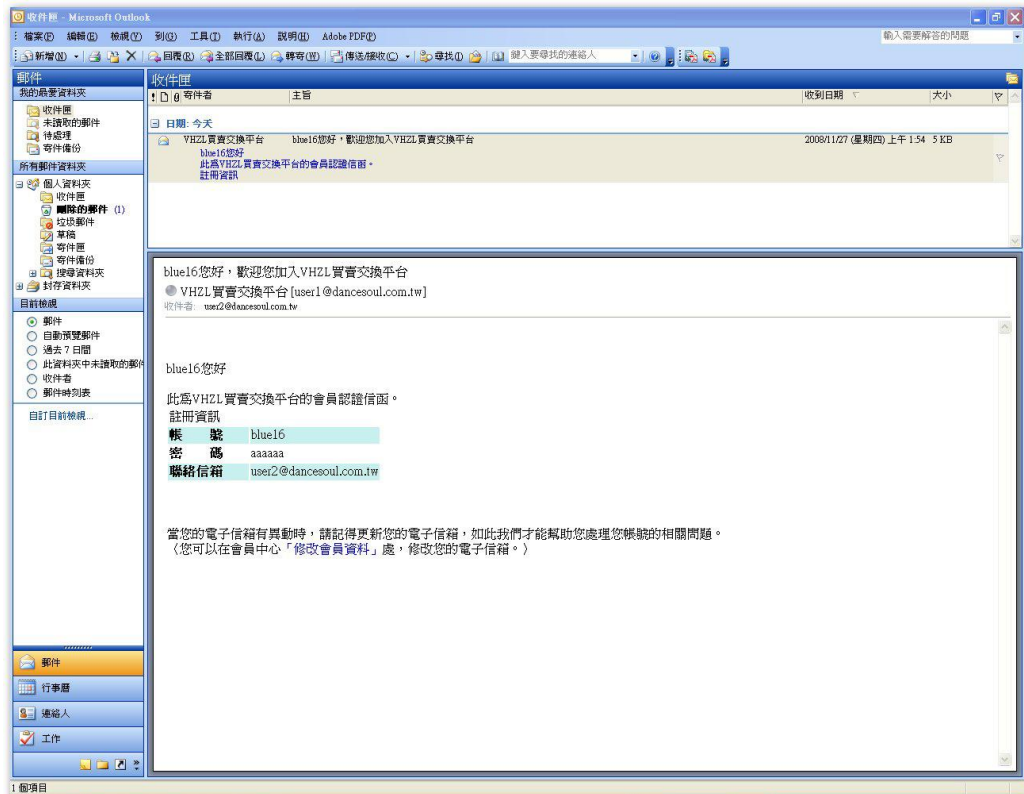


圖 3-33、註冊成功會員通知信之畫面

若有忘記使用者帳號或是密碼，可點選忘記密碼進入確認畫面，在輸入帳號與信箱之後，系統會判斷補寄密碼給使用者再做登入，忘記密碼確認畫面。

使用者成為本平台會員後，如需要修改個人資料，則可至會員中心來進行修改之動作。如圖 3-34：





圖 3-34、會員修改個人資料之畫面

若要做帳號密碼的更改，也可由會員中心選入更改。我們只提供更改密碼的服務，為避免有使用者惡意重複申請會員帳號影響交易公平，我們將由系統針對 IP 做限制。故使用者申請帳號之後，我們不提供使用者更改帳號，凡是系統中需輸入密碼的部分，我們皆會在使用者欲輸入時，自動顯示小鍵盤，如此可防側錄或是釣魚網站等之風險性，更改密碼的顯示如圖 3-35 所示：

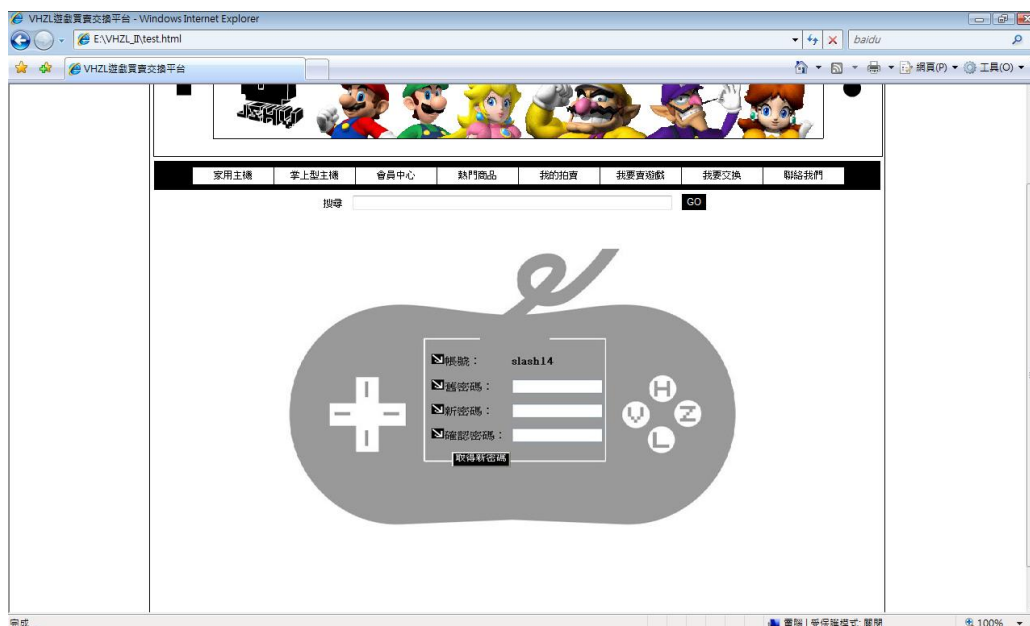


圖 3-35、會員修改密碼之畫面

另外若使用者忘記帳號之密碼，輸入多次帳號及密碼後仍然無法成功登入，如圖 3-36 所示，我們也提供會員索取新密碼之功能。



圖 3-36、會員登入密碼不正確之畫面

接下來為使用者取得新密碼之過程，首先須先輸入會員帳號及註冊時之聯絡信箱。如圖 3-37 所示：



圖 3-37、會員輸入帳號及註冊聯絡信箱之畫面

系統收到會員要求索取新密碼之訊息後，系統會寄出一封索取新密碼之驗證信件至會員信箱。如圖 3-38 所示：

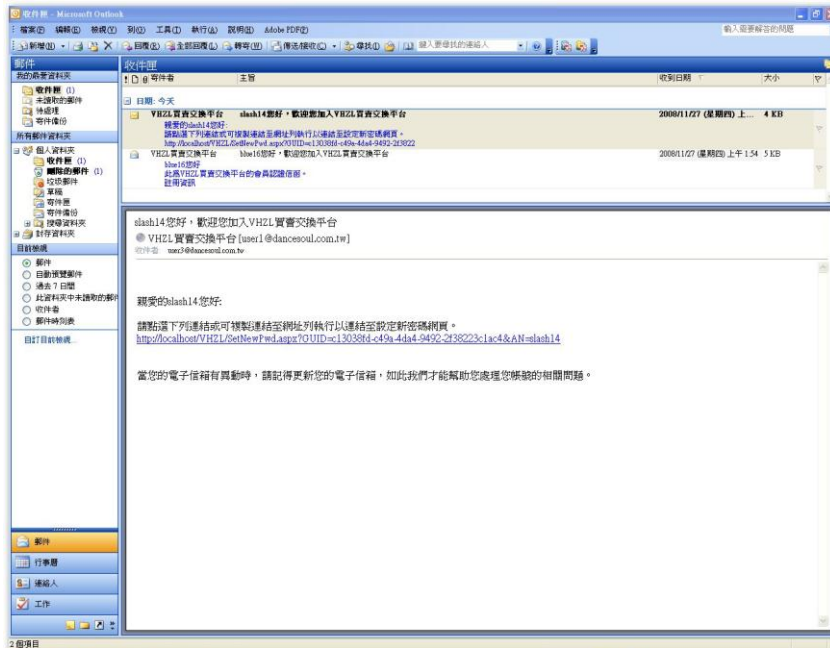


圖 3-38、系統寄出索取新密碼之驗證信件

會員驗證成功後還需設定新密碼。如圖 3-39 所示：



圖 3-39、會員設定新密碼之畫面

在會員新密碼設定步驟成功後，則會出現修改密碼成功之畫面。如圖 3-40 所示：

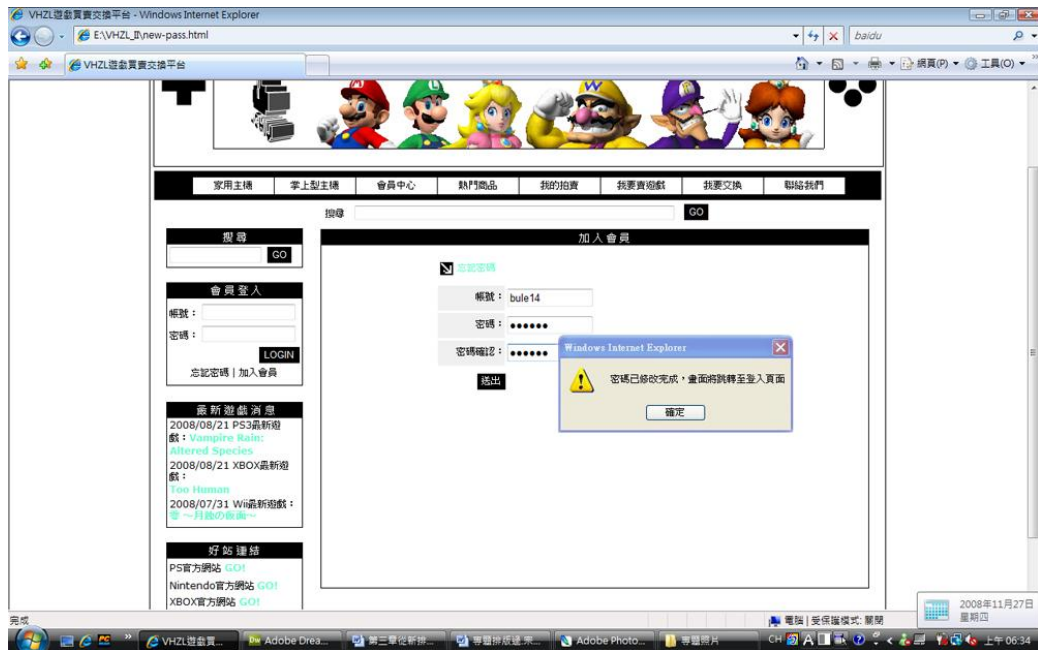


圖 3-40、會員成功索取新密碼之畫面

若會員在索取新密碼之驗證時限超過系統所設定而未通過驗證時，則會出現認證失敗之畫面，並返回首頁。如圖 3-41 所示：

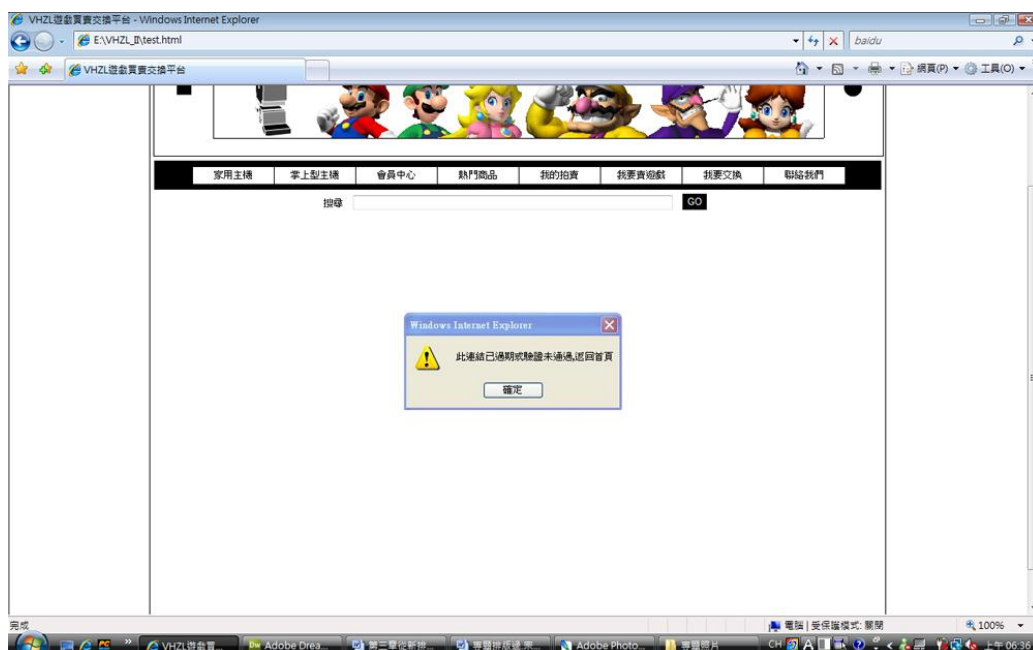


圖 3-41、認證失敗之畫面

以上為我們網站所執行的畫面，所有的功能與介面皆經過測試確認網站的運作沒有問題！並且進行其他的功能研究，詢問使用者使用方面的觀感以做改進與提供更好的服務。

## 第四章 網站的附掛功能及安全性

### 第一節 網站的安全性

#### 4-1.1 SSL 的安全性

網景 (Netscape) 開發的 SSL (Secure Socket Layer)，是一種網際網路上最普遍使用的安全通訊協定，保障網站伺服器及瀏覽器之間的數據資料傳輸的安全性。透過使用這個協定，網路上的數據傳輸會按照認證的種類(40 位元、128 位元) 進行不同程度的加密，更會檢查資料的完整性。除此以外，透過所謂『金鑰匙』的加密技術及嚴謹的 SSL 認證註冊的程序，SSL 可以驗證伺服器的身分而達到網站瀏覽者向網站身分作出檢查的目的。網站瀏覽者當看到瀏覽器右下角出現『金鑰匙』，瀏覽者可以點選查看伺服器的位置及身分，確認網站是否真實可靠。

SSL 提供客戶端與商家 Web Site 間的加密，保障資料傳送過程中的安全。基本上，SSL 是一種利用私密鑰匙 (Secret-key) 來進行密碼加密的方式，正如以公開鑰匙 (Public-key) 加密，都必須經過授權之後才能獲得認證。

SSL 網路安全協定有幾項特色：

1. 私密性(Privacy)：透過加密而能確保資訊的私密性。即

使訊息仍然可能會被第三者攔截，但是他們無法閱讀這些資訊，因為他們沒有鑰匙可以開啟。

2. 完整性(Integrity)：藉由加密來確保訊息的完整性，如果在傳輸過程資料遭到如果在傳輸過程資料遭到竄改，接收者會因為無法解密而得知訊息遭到破壞。

3. 認證(Authentication)：經由數位授權，提供資料的真實性[6]。

Windows 網路作業系統內置的 IIS 是大家最常用的 Web 伺服器。但在系統默認配置下，IIS 使用的是“HTTP 協議”以明文形式傳輸數據，沒有採用任何加密手段，傳輸的重要數據很容易被竊取。這對於一些安全性要求高的網站來說，是遠遠不夠的。為了保證重要數據的萬無一失，IIS 也提供了 SSL 安全加密機制。

IIS 伺服器的數據通信過程如下：首先客戶端與 IIS 伺服器建立通信連接，接著 IIS 把數字證書與公用密鑰發給客戶端。然後使用這個公共密鑰對客戶端的會話密鑰進行加密後，傳遞給 IIS 伺服器，伺服器端接收後用私人密鑰進行解密，這時就在客戶端和 IIS 伺服器間創建了一條安全數據通道，只有被 IIS 伺服器允許的客戶才能與它進行通信。

SSL (Security Socket Layer) 的中文全稱是“加密套



接字協議層” ，它位於 HTTP 協議層和 TCP 協議層之間，能夠對信用卡和個人資訊提供較強的保護。SSL 在客戶和伺服器之間建立一條加密通道，確保所傳輸的數據不被非法竊取，SSL 安全加密機制功能是依靠使用數字證書來實現的，應用了 SSL 加密機制後，我們便能在安全性的功能上，能夠有完善的防範[6] ！

#### 4-1.2 關於個人資料的安全性

由於個人帳號隱私的保密性，我們將會於網站首頁建議使用者自行定期更改密碼，以增加防護的功能！

會員密碼重新設定增強安全性如下圖 4-1:

UserInfo	
PK	UserInfoID
	Member_ID
	Account_Num
	Password
	MemberName
	Birthday
	Sex
	Zip
	ID
	Email
	Address
	M_Phone
	Tel_Phone
	Verify
	Role
	RegIDateTime
	IPAdress
	Private_Guid
Public_Guid	
PK	Public_Guid
	Account_Num
	URL_Info

圖 4-1、會員資料表及公鑰資料表

以上包括會員資料表以及公鑰資料表，會員資料表裏有個Private\_Guid欄位用來存放會員私鑰，公鑰資料表主要存放公鑰、會員帳號以及連結到設定密碼頁面的網址。

## 相關程式碼

ForgotPwd.aspx 頁面按鈕事件觸發部份程式碼



圖 4-2、ForgotPwd.aspx 頁面

//向頁面註冊用戶端指令碼

```
Page.RegisterClientScriptBlock("Msg",  
"
```

//使用GUID產生一組公鑰

```
Guid Public_Guid = Guid.NewGuid();
```

//產生使用者連結網址

```
String UrlInfo = Application["Root"].ToString() +  
"SetNewPwd.aspx?GUID=" + Public_Guid + "&AN=" +  
AccountNum.Text;
```

//sql query將公鑰、會員帳號、網址寫入公鑰資料表

```
String comm = "insert into Public_Guid
```

```

(Public_Guid, Account_Num, URL_Info) values(' " +
Public_Guid + "', ' " + AccountNum.Text.Trim() + "', ' "
+ UrlInfo + "' )";

//Public名稱空間下自定義方法doExecutLog()執行sql
query
Public.doExecutLog(comm);

//自定義方法SendMail()寄出會員信
SendMail(Public_Guid);

    // 存放使用者私鑰

    //使用GUID產生一組私鑰

        Guid Private_Guid = Guid.NewGuid();

    //sql query更新使用者資料表的IP Address、私鑰

        String cmd = "update UserInfo set
IPAdress=' " + Request.UserHostAddress + "',
        Private_Guid = ' " + Private_Guid + "' where MemberID
= ' " + AccountNum.Text + "' ";

//Public名稱空間下自定義方法doExecutLog()執行sql query

        Public.doExecutLog(cmd);

    //將私鑰存放在cookie

        HttpCookie aCookie;

```

```

        if (Request.Cookies["PG"] == null)
        {
aCookie = new HttpCookie("PG");

            aCookie.Value = Private_Guid.ToString();
aCookie.Expires = DateTime.Now.AddDays(7);

            Response.Cookies.Add(aCookie);}

else

        {    aCookie = Request.Cookies["PG"];

            aCookie.Value = Private_Guid.ToString();

                aCookie.Expires =

DateTime.Now.AddDays(7);

            Response.Cookies.Add(aCookie);

```

### SetNewPwd.aspx 頁面程式碼

```

//事件載入事件

protected void Page_Load(object sender, EventArgs e)

    //檢查公鑰與帳號是否為空

        if (Request.QueryString["AN"] == null ||

Request.QueryString["GUID"] == null)

            { //向頁面註冊用戶端指令碼

ClientScript.RegisterClientScriptBlock(this.GetType

```

```

(), "Msg", "<script>alert(' 此連結已過期或驗證未通過,
返回首頁
');location.href='../Default.aspx';</script>");
else //公鑰與帳號不為空
Try //將傳來的公鑰與帳號存到變數裡

        String AN =
Request.QueryString["AN"].ToString();

        Guid public_guid = new
Guid( Request.QueryString["GUID"].ToString());

        //將帳號顯示在畫面上
        lblAN.Text = AN;

        //從會員資料表查詢出Email

        String commend = "select Email from
UserInfo where MemberID = '" + AN + "'";

        mail = Public.doExecuteScalar(commend);

        //從會員資料表查詢出IPAddress,Private_Guid的值

        String comm = "select
IPAddress,Private_Guid from UserInfo where MemberID in
(select MemberID from Public_guid where Public_Guid =
'" + public_guid + "')";

```

```

cn.ConnectionString =
System.Configuration.ConfigurationManager.ConnectionStrings["ShopConStr"].ConnectionString;

        SqlCommand.CommandText = comm;

        SqlCommand.Connection = cn;

        cn.Open();

        //讀取DataReader

        SqlDataReader reader = SqlCommand.ExecuteReader();

        if (reader.HasRows)

            while (reader.Read())

                if (reader.GetValue(0) != DBNull.Value)

                    { //將IPAddress存到UserIP變數裡

UserIP =

IPAddress.Parse(reader.GetString(0));

                if (reader.GetValue(1) != DBNull.Value)

                    { //將私鑰存到UserGd變數裡

                        UserGd =(Guid)reader.GetValue(1):

                            if (Request.Cookies["PG"] != null)

```

```

{

    HttpCookie bCookie =
Request.Cookies["PG"];

    String TmpPg =
bCookie.Value.ToString();

//將cookie的私鑰取出指定給CookiePG變數

    if (TmpPg != "")

        CookiePG = new Guid(TmpPg);

//比較IP跟私鑰

//假如使用者IP Address和cookie私鑰
與使用者資料表的不符合就視為驗證未通過

if
(UserIP.Equals(IPAddress.Parse(Request.UserHostAddr
ess)) != true || UserGd.Equals(CookiePG) != true)
{

ClientScript.RegisterClientScriptBlock(this.GetType
(), "Msg", "<script>alert('此連結已過期或驗證未通過,
返回首頁
');location.href='../Default.aspx';</script>");

```



```

        }
    }

    catch (SqlException exc)
    {
        throw exc;
    }

    finally
    {
        cn.Close();

        SqlCommand.Dispose();
    }
}

```

1. 超鏈結中帶的參數是一組 GUID1(公鑰)

例如：

<http://www.google.com.tw/abc.aspx?GID=405CACB3-F658-4DD5-82E5-D1DA7DDFDB7E>

2. 驗證該【使用者】存於 Cookie 中的【GUID2 私鑰】以及限定的 IP，讓只有【該電腦的 IP】並且擁有【GUID2 私鑰】的人才可以開啟，否則將拒絕進入。

3. 需要【GUID1(公鑰)+使用者 IP 驗證+GUID2(私鑰)】三個條件缺一不可，才能夠進入設定密碼頁面。

要實現以上的方式，需有三個步驟

1. 【產生私鑰】
2. 產生 Email 中超鏈結時，【產生公鑰】
3. 鏈結回來設定新密碼頁面後【驗證 3 要件】

以上的步驟，為使用者欲更改密碼時所做的安全性的呈現，我們透過公私鑰驗證機制，確保使用者能在安全的頁面做個人資料的修改，如此可以防止他人盜用 Email 超鏈結而獲取密碼，此兼顧了安全性與 user friendly。

#### 4-2.1 防止鍵盤側錄個人資料外洩

在我們的網站中，除了會員個人安全性的機制之外，我們所建置的一個特點，就是在網站中任何一個使用者輸入帳號密碼的位置，皆設置防止鍵盤側錄程式專用的小鍵盤。由於使用者直接從電腦輸入其帳號個人隱私之資料，有極高的風險會在輸入的同時洩漏個人帳號密碼。因此我們設置一個強迫使用者輸入的小鍵盤程式，藉此以保護使用者個人帳號資料的安全性!輸入資料畫面如下圖 4-3 所示。

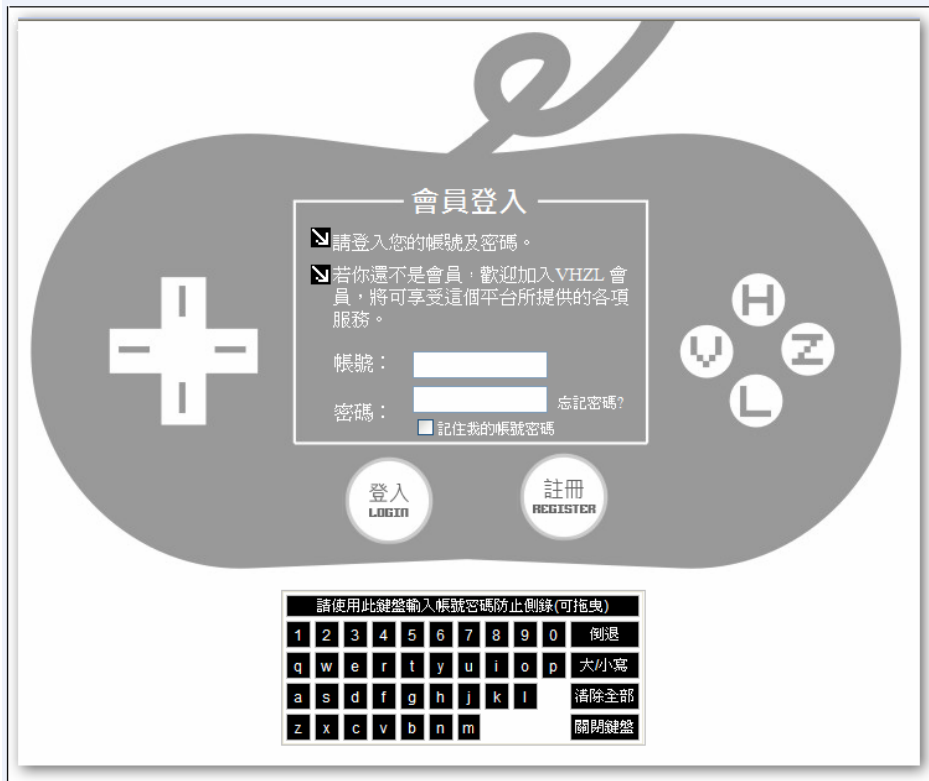


圖 4-3、防止鍵盤側錄小鍵盤

除了輸入時保密的設置外，我們會於會員申請帳號後三個月寄發系統通知信，請會員自行更改帳號密碼。此措施為通知之義務行為，並不會如其他網站，若不進行更改將無法繼續使用，以增加使用者之意願。

#### 4-2.2 SQL Injection (資料隱碼)

在現今的應用程式架構中，大部分都含有資料庫，以容納各式各樣的資料。而在各類型的資料庫中，又以結構化查詢語言(SQL Structure Query Language)為基礎的關聯式資

料庫管理系統 (RDBMS Relational Database Management System) 最為流行。

一般的程式設計師在存取資料庫時，往往是利用 Visual Basic 等第三代語言來組織 SQL 語言，然後再傳遞給關聯式資料庫系統執行，以建立或刪除資料結構，賦予或移除使用權限，乃至於新增、修改、刪除或查詢資料。因為關聯式資料庫所有的執行動作皆是遵循 SQL 命令，所以透過此種方式可以很方便地完成各種資料維護工作。但也正因為 SQL 語言無所不能，所以稍有漏洞就會讓駭客有機可乘。我們將就針對這個主題做一個深入的探討。

網站的資料存取一般來說是比較危險的，因為網際網路是一個開放的環境，而不像一般公司內部網路，除了有電腦本身的安全設計，還可以過濾篩檢員工的身分背景。網際網路上龍蛇雜處，大部分的使用者都循規導矩，但少數圖謀不軌的人卻處心積慮地要侵入我們的系統，竊取有價值的資料。但一般的網管人員及網頁設計師，可能在安全設定上有著重重防範，如架設防火牆，設計非軍事區 (DMZ)，限制網站登入者的身分等等。但由於缺乏對 SQL 語言及資料庫管理系統的認知，而大開系統的後門[5]。

以下為本網站所設置的 SQL Injection 防範機制相關程式碼

```
//Public 名稱空間下自定義方法 CheckString() 用來檢查字  
串有無特殊字元有則傳回 err 無則傳回空字串  
  
    //預防輸入 ' 等會造成資料庫錯誤之特殊字元  
  
    if (Public.CheckString(InputA_N.Text) != "" ||  
Public.CheckString(InputPWD.Text) != "")  
  
ClientScript.RegisterClientScriptBlock(this.GetType  
(  
), "Msg", "<script>alert(' 不正確的帳號或密碼!');</script>");  
  
    else  
  
        //取得mail  
  
        String commend = "select Email from UserInfo where  
Account_Num=" + InputA_N.Text;  
  
        if (Public.doExecuteScalarLog(commend) !=  
null || (String)Public.doExecuteScalarLog(commend) !=  
"  
  
            mail =  
  
(String)Public.doExecuteScalarLog(commend);
```

```

//取得memberuid

String getuid = "select Member_ID from
UserInfo where Account_Num=' " + InputA_N.Text + "' ";

String uid =
Convert.ToString(Public.ExecuteScalar2(getuid));

SqlConnection cn = new SqlConnection();

SqlCommand SqlCmd = new SqlCommand();

//取得存放在web.conf的連接字串

cn.ConnectionString =

System.Configuration.ConfigurationManager.Connection
nStrings["ShopConStr"].ConnectionString;

SqlCmd.CommandText = "Loginchk";

SqlCmd.Connection = cn;

//使用預存程序預防sql injection

SqlCmd.CommandType =

CommandType.StoredProcedure;

SqlCmd.Parameters.AddWithValue("@Account_Num",
InputA_N.Text.Trim());

//使用Public名稱空間下自定義方法Encrypt()將
字串使用MD5單向雜湊演算法加密儲存到資料庫

```

```
SqlCommand.Parameters.AddWithValue("@Password",  
  
Public.Encrypt(InputPWD.Text));
```

資訊安全一直是 Web Form 程式設計中很容易被輕忽的問題，其中以 SQL Injection 攻擊最常見破壞性也最為嚴重。因此我們使用了上述幾種方式來避免 SQL Injection 攻擊，例如，防止輸入特殊字元、使用預存程序。其中最主要的方法便是防止使用者輸入特殊字元，此舉不僅防止了 SQL Injection 攻擊也同時防範了 XSS 攻擊，可說是一舉兩得。

# 第五章 結論與探討

## 第一節 網站的討論與改進

### 5-1.1 使用者的意見及滿意度

為提升服務品質，了解現階段使用者對於網站之使用情況，針對網站內容與服務檢討並改善，以達到使用者期許之目標規劃方向的導出：

(1) 方法：問卷滿意度調查、使用者深度訪談(有固定使用者)、網站內各細部功能 log(使用者登錄紀錄)分析及流量統計、會員資料統計及與使用狀況統計

(2) 以上的方法可導引出一些改善案：

1. 使用者上此網站目的，及網站設置目的是否相符？及未來要如何調整？使用者上此網站的目的，與滿意度調查的差距，可診斷出網站的問題。

2. 使用者 對於各細部功能使用率是否平均？(可看出各細部功能對使用者的幫助程度)，(開放式、及封閉式網站的分析方法不太一樣)進一步了解原因並進行「弱勢的功能補強」及「優勢功能加強服務」的方向



3. 使用者對於使用便利性要求:含速度、介面容易操作、「美工及畫面清晰度方面」(外觀、顏色、圖像)…方面的改善案。

4. 如果是有會員資料及交易資料(電子商務網站)就可進行，明定目前消費者等級及經營策略、商品貢獻度檢視、熱門商品、成本與效益..等方面的改善依據。

由於我們設置之網站只鎖定部分遊戲玩家族群，針對遊戲方面，需要客觀的評論及意見。由於時間及進度之關係，故我們沒有實際上線供網路上開放的公共空間供網友鄉民們測試，此也經過多方的考慮，包括開放空間之安全性，或是關於數據的掌握度及真實性。在眾多風險考量之下經由小組成員決議後，我們初步只開放班上有資深遊戲經驗之同學做測試，以增加問卷的準確性，及可參考數據的真實度!

我們開放 15 位同學做線上的測試，初步測試結果如下：

表 5-1.1、網站測試滿意度

	非常好	滿意	很普通	不方便	非常糟
實用性	8	3	2	1	1
外觀	4	3	5	2	1
安全性	6	4	4	1	1
便利性	9	2	1	2	0
新鮮感	7	4	2	1	1

經由上述問卷結果，我們更進一步經由同學們的意見可以發現幾個建議跟改進方向：

1. 實用性：有近 11 位同學對於可以將舊有的遊戲光碟片拿出來與其他同學分享交換。這點是非常實用的，並且在無等值商品可以做交換之時，也可以利用拍賣的功能一邊兜售，這點是我們未來可以考慮參考的優點之一。

2. 外觀：由於網頁版面設計部分，留白的部分比較單調，大部分同學傾向能夠與其他相關網頁連結，或是有相關的廣告畫面，較能吸引人的活動之類的方式呈現，由於時間的關係這部分尚未

多做其他連結，且某些網站廣告部分尚需與其互助合作，這也將列入我們未來改進的範圍之一。

3. 安全性:在經由同學們測試之後，由我們所設置關於會員資料部分的安全性、公鑰的驗證、更改密碼的方式、以及在每個登入畫面的輔助小鍵盤，這點在初步構成的網站中，算是有規劃的設計，故對此也有 10 位的同學表示肯定。

4. 便利性:由於網站本身並沒有太多衍生其他不相關的商品，故不論操作或是交易進行的流程中，我們提供簡單安全的環境。這點在便利的程度上來說，如同交易市場的直接跟單純化，因為商品固定在遊戲方面，與其他的拍賣交換網站眾多商品議價估值的繁雜性來說，鎖定單一特定類別商品，是同學們感到方便的地方。

5. 新鮮感:在此項問卷調查發現，我們網站的創新感，似乎針對屬於遊戲玩家的同學們較有吸引力。此問卷調查的最不滿意的選項『非常糟』的選項，是一位不常接觸遊戲的同學，此一結果也讓我們對於網站未來的市場有非常大的助益，網站的效益是否也要與商品類別一樣鎖定單一族群，或是走向讓眾多消費大眾可

以接受的消費類別，或是『拍賣』與『交換』的效益跟方向。若勢趨向拍賣，勢必將往營利方向以及商品市場多樣化結構方向衍生；若為交換，也必須考慮如何在單純的以物易物的交換網站以長久經營為目的的服務，更有效的系統整合。我們將參考以上各位同學的意見，來讓我們的網站從中得到改進的方向。

## 5-1.2 結論

本專題研究讓我們發現拍賣網站的獨特性以及提升拍賣網站服務品質的重要性。對於拍賣網站服務品質，還有網站未來的發展面，我們做出以下針對服務品質發展構面作為提升策略的建議：

### (一)、效率性與系統穩定性

我們在網站上商品的分類與搜尋應更加人性化，貼近使用者，讓無論是買家或是賣家都能輕鬆的連結到想要的頁面或商品，並且不會有錯誤連結或是網站當機的情況，導致網站使用者蒙受不必要的損失。

### (二)、娛樂性

將來可加入即時通訊與視訊功能，能讓買賣雙方立即線上磋商與討論更能有效率的提升交易達成率，享受虛擬網路中逛街與殺價的快感，進而達到愉悅與沉浸的感覺。亦或者是增加網站上的娛樂性功能，以增加瀏覽人數，增加更多的交易機會。

### (三)、隱私、安全性

拍賣網站上之交易，買賣雙方最關心且令他們有所猶豫的是網路交易是否安全，個人資料是否會外洩。我們為了取信於買家，充分揭露賣家的資訊，但是卻讓賣家的個人資料有外洩之疑慮，這是經營拍賣網站自相矛盾的地方。建議網站在個人資料審核評鑑上應該更加重視，一但買賣糾紛紀錄太多，網站應立即停止其使用者或是該 IP 位置，甚至相關個人資料再度申請帳號。例如：同樣的手機號碼，或是地址身分證字號等。網站經營者應更加重視網站申請帳號者資格，嚴格掌握正確的個人資料，妥善管理保密，並且在發生買賣交易糾紛時，能適時的給予受害者幫助與服務。而要完全防堵多重申請帳號便是有賴管理者能對個人資料能嚴格的審查，並妥善管理避免外洩。

### (四)、個人化

最終也是最吸引人的，可能不是網站的功能有多強大，或是網站的人氣有多高，而是一個使用上的習慣。若能輔助個人化信箱、即時通或是結合網路視訊與電話，會使拍賣網站不只是拍賣網站，能讓賣家進行客戶關係管理的工作，會使更多使用者願意申請帳號投身拍賣之中，形成一個個人化的團體群聚。使賣家有足夠的市場經營商店，也充分的讓其享受在個人化的世界。因為網路的方便，最終在於個人的習慣。若能讓使用者使用上手方便

並且習慣專屬於自我的介面，也讓買家有足夠的選擇挑選商品，因此我們覺得強調個人化色彩以及不失網站獨特性，能夠吸引更多的使用者族群。

#### (五)、獨創性的特色

我們自始自終所強調的特色，最主要就是以交換為主，交易買賣為輔。設置一個針對遊戲使用者最具強大功能性的畫面，使用者玩膩了想換口味，或是想嘗試其他不同的體驗，都可在我們專屬遊戲的網站上找到新鮮貨，這也是我們想放大的特色跟焦點。

依據上述使用者的意見調查跟分類，我們可以清楚的發現，針對遊戲使用者以及普遍的玩家來說，急需一個遊戲的整合平台。伴隨著網路的發達，遊戲玩家越來越多，一個完整且又方便的平台是符合眾多遊戲愛好者的心聲。當業者只重視著金流部分卻忽略了最基本的 TV 電玩類型的愛好者，目前市場的熱度跟重心都放在線上遊戲的市場，遊戲主機市場的產值卻往往為大家所忽略。待集合與改進使用者們的建議及意見後，我們希望藉此單純又獨創的網站服務，能夠由此網站可以帶給更多的使用者一個方便，良好的使用者平台，往後的遠景將往交換服務擴大跟衍生，願此交換機制的服務能造福更多有需要之使用者。

### 5-1.3 改進與遠景

就網站功能部分的改進方面，我們將除了在安全性方面再增加會員資料的保密功能之外，也嘗試不論在使用者或是後台部分，包括傳輸時的安全性方面繼續添加，也將上線測試採取更多使用者的意見並加以改進，所以安全性的部份，將有最大的改進空間。

還有上述提及為了能吸引更多的使用者喜歡並且願意使用我們的網站，所以個人化的部份，我們如何將網站的特色以及個人化的網頁風格有所區別，又不影響整個網站風格，我們也將設計更多版面供使用者選擇。

在買賣雙方相互溝通的使用上，我們希望將來能透過線上的傳訊系統來整合使用者間的通訊服務。不論是線上傳輸通訊程式，或是未來有可能衍生到手機即時傳訊服務，都是將來可以針對溝通方面所做的改進模式。因為原本所存在的討論區，或是拍賣的問與答，在交易的時效上，真的不符合使用者快速簡便的需求。故買賣雙方溝通的便利性，將大大提升我們與其他拍賣網站的不同，這是我們極力追求的目標。另外商品的規劃方面，將從特立的交換機制方面去衍生跟設定，除了讓買賣雙方的溝通更方便之外，也讓交易或是交換的進行更為順利快速。若是雙方無法達成共識，第一次協調商品若無法交換，後來可將其列入拍賣或

是線上交易，以及從溝通方面去改進。總結在缺點的改進上，我們將從安全性以及個人化的區別，還有便利性的方向去改變，以求讓使用者有最完善的服務及有效的運用每個人的資源，從重複使用的習慣上，讓我們的網站成為使用者必備的『習慣』。

」



## 參考文獻

[1] 台灣網路拍賣的趨勢 雅虎奇摩知識家

<http://tw.knowledge.yahoo.com/question/question?qid=1306051304997>

[2] 大家對網拍拍賣的意見?

<http://tw.knowledge.yahoo.com/question/question?qid=1306041715639>

[3] C#自學手冊 作者：洪維恩 出版社：博碩 出版日期：  
2003-05-05

[4] Microsoft SQL Server 2005 管理實務 作者：施威銘研究室著 出版社：旗標 出版日期：2006 年 07 月 14 日

[5] ASP.NET 首頁 Microsoft 網站技術支援

<http://www.microsoft.com/taiwan/msdn/asp.net/default.aspx>

[6] IIS 網站 S S L 安全性機制技術網站

<http://www.pcdog.com/edu/security-tech/2006/08/v135695.html>

[7] (資料隱碼) - 駭客的 SQL 填空遊戲 駭客論壇

[http://www.microsoft.com/taiwan/sql/SQL\\_Injection\\_G1.htm](http://www.microsoft.com/taiwan/sql/SQL_Injection_G1.htm)

[8]網路銀行的 SSL 安全保護安全嗎? 雅虎奇摩知識家

<http://tw.knowledge.yahoo.com/question/question?qid=1106111912482>

[9]感謝資網系同學陳佑宗.何勝露...等 15 位同學參與問卷!