

國立台北大學資訊工程學系專題報告

RFID 智慧型停車場管理系統

專題組員:蔡江漁

指導老師:張仁俊老師、吳信龍老師

執行期間:100 年 2 月 至 101 年 1 月

1. 動機

在現今的社會中，人口不斷的成長，連帶著汽車的數量也快速的增加，尤其是在大都市中，許多的開車族經常面臨到停車的問題。現在的停車場無論是公營或是私營，多半都有管理系統，但僅提供車輛停放以及計時收費的簡易功能。

設想一個情境，在一個室內停車場內僅剩下兩三個空位，而一位駕駛將車輛駛入該停車場，卻因為不知道空位在哪裡，只好靠著運氣在停車場內找位置。

另一個更糟糕的情況，假如停車場沒有適當的管制進入停車場的車輛，很有可能造成在停車場內找車位的車輛比空位還要多，此時這些駕駛只好漫無目的地在停車場裡面繞圈子，看看會不會剛好碰到要離開的車子。

無論是哪一種情境，對於駕駛者而言都是一種時間上的浪費。如果能讓停車場管理系統更智慧，無論對於駕駛者或是停車場管理者來說都是一大福音！

2. 目的

我們所追求的是一個有效的管理機制與系統，該系統可以解決上述問題並且將其它在一般停車場內可能會需要的應用一併整合。最終的目的是希望駕駛者可以在最短的時間之內找到車位並停好車，管理者也可以輕鬆容易的管理停車場。

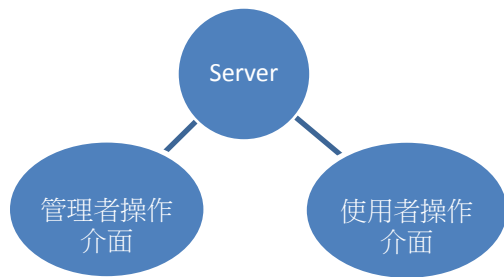
3. 專題進行方式

本專題執行時間共約十二個月。首先因為有了動機而發想到這個題目，在題目確定之後緊接著設定這個系統的主要功能：我們把使用這個系統的角色分為兩種，分別是「使用者」與「管理者」。

使用者即為駕駛者，也就是要來停車場停車的消費者，對於使用者而言所要提供的功能為停車導引，將車輛從停車場入口正確且快速的導引到空的停車位，就是本系統最大的功能。除了停車導引之外，便利的繳費方式應該也是使用者需要的功能。

對於管理者而言，我希望提供一個很簡單易懂的介面，並且希望該系統的通用性可以很高，所以本系統提供了停車場內部編輯、人員資料編輯以及值班表編輯這些功能。

心中有了這幾個主要的目標之後，我開始思考系統的主要架構，最後我決定採用主從式架構，所有的停車場資訊、使用者資料以及一些複雜的運算都在 Server 端儲存或運算，Client 端僅負責顯示介面與發送指令。如此一來除了可以減輕 Client 端的負擔之外，在擴充系統(停車場有多個出入口，或是系統要增加新功能)時也非常的方便。



確認了系統的主要功能與架構之後，便真正的開始實作程式。本系統以 C++ 撰寫，使用了 wxWidgets 2.8.12 處理使用者介面、OpenGL 處理圖像介面、OpenSC 連接與操作 RFID 讀卡機、SQLite 3.7.9 當作資料庫。本專題多數的時間就是花在撰寫程式與 Debug。

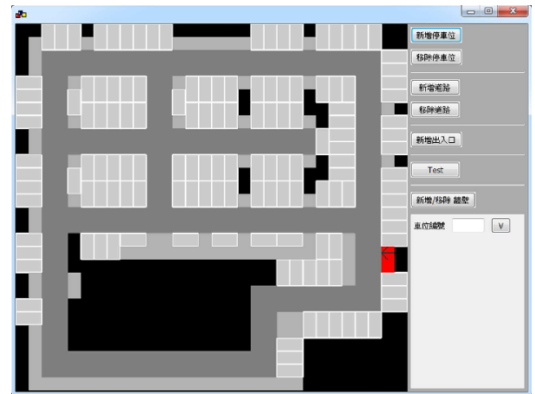
系統大致完成之後，我便找了一個真實的停車場，將這個停車場輸入到我的系統中，以模擬的方式測試系統的正确性與穩定度。

4. 專題實作方式與成果



經過了將近一年的開發與規劃，目前系統已經可以上線使用，主要的功能有：

(一)自由編輯停車場內部配置



本系統不是專門為某個停車場設計，透過編輯停車場內部配置的功能，讓系統可以適用於任何一個停車場。

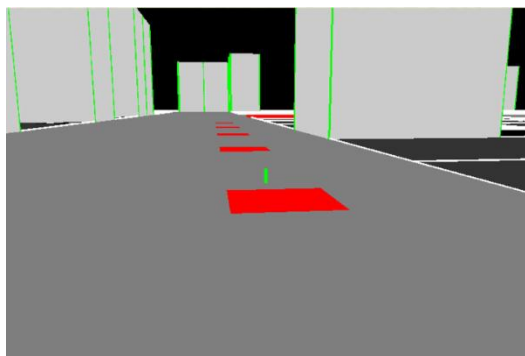
我們將停車場簡化為四個元素：

- 甲、 車道
- 乙、 車位
- 丙、 出入口
- 丁、 牆壁

我們將每個車道兩端點或是車道之間的交叉處視為一個節點，節點與節點是否連通就取決於是否有車道將接，同時也記錄下節點之間的距離(即為車道長度)，如此一來即可將停車場化為一個 Graph，在規畫路徑時即可使用最短路徑之演算法解決。

另外牆壁的部分，則是考量到室內停車場通常會有建築物的柱子穿插於停車位之間，為了盡量呈現出停車場內部的構造，所以將它納入基本元素之一。

(二)自動分配車為與停車導引



駕駛者在入口處感應 RFID 卡後，系統會自動分配車位並且導引駕駛將車輛開往該車位。

利用前一項所提到，將停車場轉化為一個 Graph 後運用最短路徑演算法規畫路徑。本來是使用 Dijkstra Shortest Path 演算法，但後來考量到駕駛者可能不依照系統規劃的路線前進，此時就需要重算最短路徑。為了避免這個現象，所以後來採用 Floyd-Warshall Shortest Path 演算法，雖然該演算法的複雜度較前者來的高，但是優點是偏離本來規畫的路徑時無須重算。

(三)管理者資料管理與權限管理

考量到一般停車場通常會請工讀生來顧停車場，所以本系統將管理者又分為有所有權限的管理人與有部分權限的值班人。管理人可以編輯停車場或排班表，值班人就只能管理顧客資料。權限的區分是利用進入系統時感應 RFID 卡，也可以設定文字與數字組合的密碼來加強安全性。

(四)值班表系統



最高權限的管理人可以編輯值班表，本系統將每天區分為三個班，早班、午班與晚班，每天的每個班都可以設定該時段的值班人是誰，事後也可以追蹤值班人上下班的情況，作為發薪水的依據。

(五)顧客資料管理



配合顧客繳費的功能，本系統可由管理者登錄顧客基本資料，除了可以更精準地掌握停車場內顧客的資訊外，另外也可以對顧客的 RFID 卡進行增值或是人工繳款。

(六)顧客使用 RFID 卡自動繳費

由於入場與出場皆使用 RFID 卡感應，所以本系統整合了自動繳費的機制，系統會於顧客出場時依據車輛在停車場內停留的時間自動計算出應繳金額，顧客僅需要事先於 RFID 卡內儲值足夠的金額，即可自動扣款。

5. 製作過程遇到的困難

(一)讀卡機沒有完整的 API



讀卡機是由老師的實驗室提供，共有兩套設備，其中一套設備需要安裝額外的驅動程式，為了避免使用者安裝系統上的麻煩，所以我採用另外一套新採購的讀卡機(SCL010、SCL3711)。該讀卡機雖然有附上 SDK，但卻沒有完整的 API。

我解決的方法是在網路上搜尋，最後找到針對智慧卡讀卡機所設計的函式庫 OpenSC，根據我們使用讀卡機的 Reference manual 針對 OpenSC 函式庫做小幅度的修改，最後終於可以正確的控制讀卡機。

(二)無法實際運行測試

因為本專題在規劃時並沒有考量到實際測試的問題，所以我們的硬體設備除了電腦以外就只有讀卡機，以至於我們沒有辦法真正安裝到一個停車場做測試。

針對這個問題，我們選擇用電腦模擬的方式去測試，透過 OpenGL 呈現立體的虛擬環境給使用者來體驗。雖然跟現實狀況仍有差距，但已經可以展示出系統絕大部分的功能。

(三)資料庫無法加密

由於我們使用的是免費(開放原始碼)版的資料庫 SQLite，該版本不提供加密的功能，對於此問題我們曾經思考過儲存在資料庫內的個人資料是否有安全性上的疑慮。

我們從系統整體的架構上去解決這個問題，因為我們採用了主從式的架構，對於顧客或是管理者，接觸到的都是 Client，只要在 Server 沒有被入侵的情況之下，任何存取資料庫的行為皆是由本系統定義並且合法的。

另外對於密碼的儲存也多了一到手續，我們使用 MD5 將密碼加密後再儲存於資料庫，所以就算是接觸到 Server 的維護人員，也無法得知用戶的密碼。

6. 結論與未來展望

在開發程式的這一段時間，還是看到了很多停車場因為沒有有效的管理方式而導致於混亂，許多駕駛者都在抱怨，停車場管理員卻也無能為力。本系統的建置成本並不高，也沒有過多的條件限制，假如多數停車場可以安裝此系統，顧客滿意度必然可以提高，並且解決了相當多的停車問題。

該系統的架構設計成可擴充性相當高，大部分的功能都可以模組的方式掛上現有的系統，對於後續的開發維護相當便利。例如監視攝影系統、車牌辨識系統，甚至可以與警方連線，協助贓車的搜尋。

專題製作期間，非常感謝張仁俊老師、吳信龍老師與實驗室同學們的批評與指教，也期盼未來若有學弟妹承接此題目，可以將更多的應用整合至本系統中。