

# 中華大學資訊工程學系專題期末報告

## 結合社群網路之個人化數位看板

### - 以景點資訊系統為範例

指導教授：張欽智

組長：溫思語 (B09802013)

組員：梁威祥 (B09802001)

王凱彥 (B09802007)

蔡明運 (B09802025)

專題題號：**PRJ2012-CSIE-10102**

執行期間:101 年 2 月 至 102 年 1 月

# 目錄

1、摘要.....	3
2、研究動機與研究問題.....	3
3、研究目的.....	4
4、開發相關軟體.....	4
5、工作分配與進度表.....	5
6、系統設計與架構.....	6
7、資料庫設計.....	7
8、系統實作.....	9
8.1 電子看版首頁設計.....	9
8.2 景點內頁設計.....	13
8.3 卡號與RFID的連接與註冊機制.....	15
8.4 景點推薦機制.....	18
8.5 與社群網站的連接.....	19
9、行動端程式設計.....	23
10、結論與未來展望.....	25
結語與銘謝.....	26
銘謝.....	26
參考文獻.....	26

## 1. 摘要

以Android作業系統及Facebook社群網站為主的時代來臨，其快速的成長明顯的改變了人們的生活方式。基於Android open source及Facebook API的整合，因此更容易開發應用程式及掌握網路社群資訊的交流，而在數位電子看板(Digital Signage)應用領域中，多媒體系統將個人在社群網站資訊進行整合，並由系統所查詢到使用者的興趣主動推薦內容，讓使用者能夠快速接收較符合個人喜好的資訊，減少不必要的資訊閱覽。

此專題的目地主要提出一套以Android作業系統為主之個人化數位電子看板，並以旅遊景點為例，結合使用者所參與Facebook社群網站資訊以及無線射頻辨識(RFID)技術，讓使用者藉由系統所登錄的「悠遊卡」識別資料，推薦具個人化的旅遊資訊。推薦機制採用評價可信度及個人喜好推薦演算法，然而在社群網路資料的分析中，由於使用者的群體行為對於使用者個別的行為有著深厚影響力；因此，系統亦收錄這些關鍵因素，例如：評分、喜好及興趣，讓系統提供更準確及合適的景點資訊。

## 2. 研究動機與研究問題

由於旅遊人數與日俱增，對於想要深入了解地方文化特色，並依據個人喜好參訪的旅者，必須在行前充分收集地方景點資訊，雖然景點資訊提供的來源很多，但是資料量眾多導致資訊混亂，由於無眾多人的推薦，讓大多數人選擇由業者安排制式的行程，造就重點景點觀光熱潮，具地方特色的景點乏人問津。設想，如果有一套資訊服務系統安置在各地公共場所，使用者只須刷卡識別機後，系統會主動推薦出合適的景點，不僅減少資料查閱量，亦讓遊客快速的找到喜愛的景點。

目前數位電子看板運作方式大多朝垂直整合，因此，資訊內容的呈現方式大多已制訂，且無法隨著使用者的差異給予適當的回應，但隨著許多不同社群網路的產生，多元性質的社群網站大量吸引各式使用者加入，許多個人化的推薦技術也開始結合社群中的群體決策當作有效推薦。總合上述，數位電子看板的資訊內容如果能夠結合社群網路的網路活動，對於景點資訊的分享及回應，那能夠預期使用者在利用景點資訊時所得到的推薦將更加充分及可信，在研究中以數位看板結合景點資訊內容為研究題材，透過社群上個人設定檔(User Profile)的興趣設定以及對景點的評比，結合個人化的推薦技術回饋合適的景點資訊給使用者。

### 3.研究目的

本篇專題的主要研究目的是希望數位電子看板能夠整合社群網路的群聚效應，利用群體中每位使用者的評比記錄去預測使用者的需求，最後將合適的資訊內容呈現給使用者，讓使用者無須浪費時間在於資訊查閱的動作。

在專題中，我們建立兩套以景點為主的資訊系統，一套是運行在 Android AVD (Android Virtual Device) 模擬器上的資訊系統，另外是內嵌於 Facebook 網站的景點資訊網站。在實驗中為讓系統能夠使用刷卡方式來辨別使用者，我們架設一套 RFID 設備於模擬環境上，當使用者進行刷卡動作後，裝置會透過所識別資料進行確認，並依據使用者在社群網站登錄資料與後端資料庫的歷程資料進行個人化推薦演算，最後，將處理過的景點資訊回饋至使用者畫面上。

### 4. 開發相關軟體

網頁端 — AppServ

AppServ 2.5.10 內含：

- \* Apache 2.2.28
- \* PHP 5.2.6
- \* MySQL 5.0.51b
- \* phpMyAdmin-2.10.3

**Apache** 是一款免費又強大的網站伺服器軟體，能跨平台使用，比 Windows 的 IIS(Internet Information Server)還要更強大

**PHP** 是 Hypertext Preprocessor 的簡稱，是伺服器端的程式語言，嵌入 HTML 語法中，讓伺服器能與使用者互動。

**MySQL** 是 SQL 資料庫伺服器，網站的資料都儲存在這裡。

**phpMyAdmin** 是一種網頁程式，能讓使用者方便管理 MySQL。

## 行動端 — Eclipse

Eclipse是一款由Java所編寫成的整合開發環境，最早是由Object Technologies International所開發，於1996年被IBM收購後便改成Common Public License的授權方式，使Eclipse成為開放原始碼的軟體。雖然最初主要是用於Java語言的開發，但透過活躍的社群和許多的外掛、插件的支持下，使Eclipse成為一個多功能、高品質且支援多種程式語言的軟體開發環境，同時也是Google官方所推薦的Android開發平台。

### Android SDK

Android的軟體開發工具包。其中包括Android作業系統、Android模擬器、Android開發工具、函式庫、應用程式API與Android範例程式。它提供了Windows/Linux/Mac各種平台的開發套件，使我們能夠透過各種平台來開發Android軟體，所有的Android應用程式都是使用Java語言來撰寫的，再利用Android模擬器來做測試與除錯。

## 5. 工作分配與進度表

表1 成員工作分配

溫思語	系統整合、景點資訊系統建置、社群網路連接與整合、行動端頁面與程式設計、報告撰寫
梁威祥	個人系統數位看板建置、系統測試、功能構思
王凱彥	資料收集與統整、資料庫建置、識別功能(RFID)設計
蔡明運	Google API 整合、系統測試、海報製作、美術監製

表2 專題進度表

工作內容	3月	4月	5月	6月	7月	8月	9月	10月	11月	12月	1月
前置作業與討論											
個人數位看板系統建置											
景點資訊管理系統建置											
APAS、Facebook資料庫建置											
Android平台整合											
社群網路整合											
識別功能(RFID)設計											
模擬實驗											
實驗結果分析與探討											
報告撰寫											

## 6. 系統設計與架構

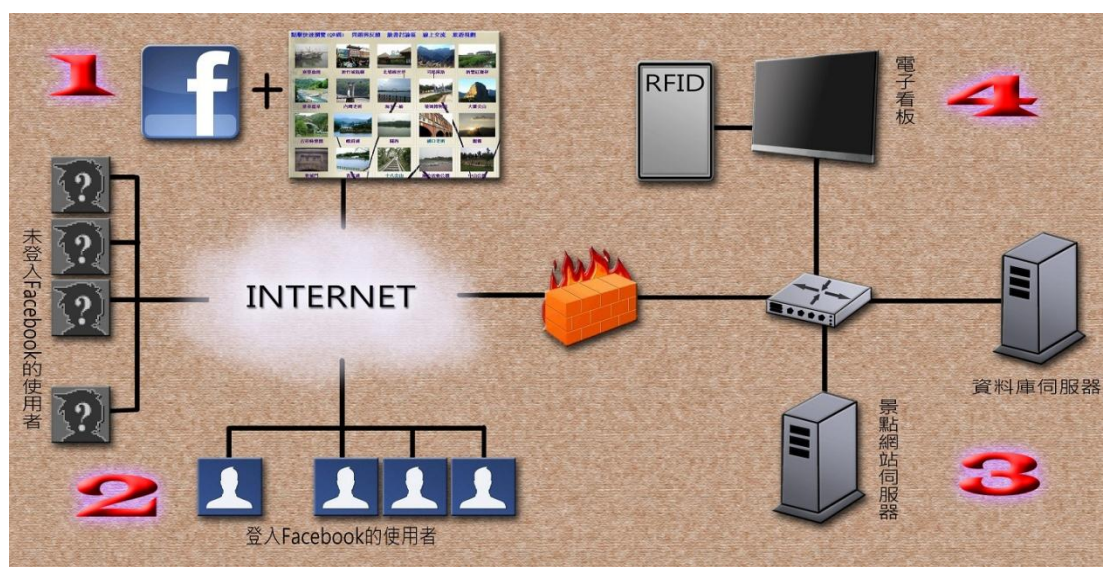


圖 1 系統運作情境的概念圖

依據圖 1 中的①~④分別說明使用情境：

- ① 內嵌於 Facebook 的旅遊景點資訊網站，並隨時提供使用者景點資訊的查詢服務。
- ② 遠端的使用者經由網路連上位於 Facebook 網站內的景點資訊頁面，並對景點進行分享、讚好、評論或評分。
- ③ 伺服器端提供景點網站資訊的發佈並由資料庫伺服器彙整使用狀態。
- ④ 位於設置景點查詢系統的使用者透過悠遊卡刷卡取得推薦的景點資訊。

在 APAS 整體的系統架構中，我們可以將系統歸納三部分[3]：

1. 個人化數位看板系統(Personalized Digital Signage System)，主要運行在 Android 作業系統上的互動式景點瀏覽介面，一般使用者可以透過觸控式介面查詢景點的相關資訊，而曾在景點網站註冊過的 Facebook 使用者，可以攜帶悠遊卡刷卡進行個人化的查詢，並由系統主動推薦景點資訊；
2. 景點資訊管理系統(Attractions Information Management System)，內嵌於 Facebook 上的景點資訊管理介面，功能包括提供景點網站以及數位電子看板的景點資訊，另外也提供使用者悠遊卡卡號的登錄作業。
3. 資料庫(Databases)，包含了自建的 APAS 資料庫與外部 Facebook 的資料庫，APAS 資料庫儲存景點與使用者的評價資訊，Facebook 的資料庫儲存使用者在 Facebook 上的個資與讚好評論相關資訊。

## 7. 資料庫設計

資料庫 ER 圖

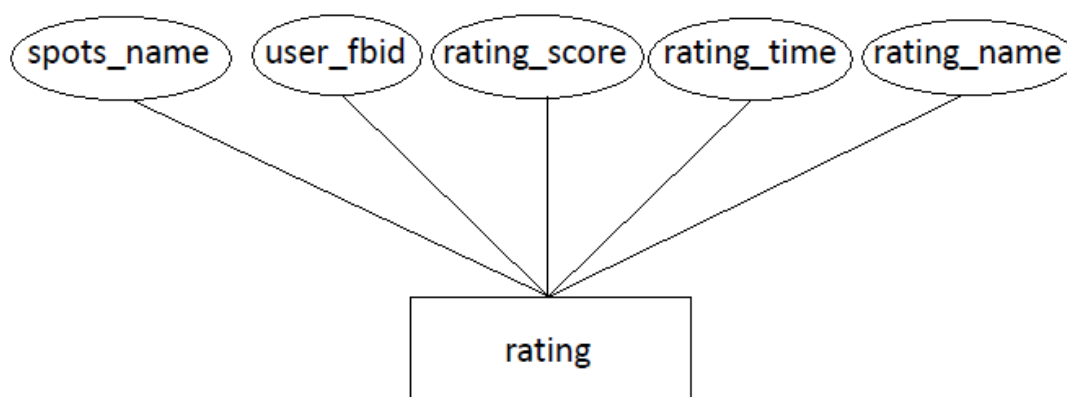


圖 3 Rating 的 ER 圖

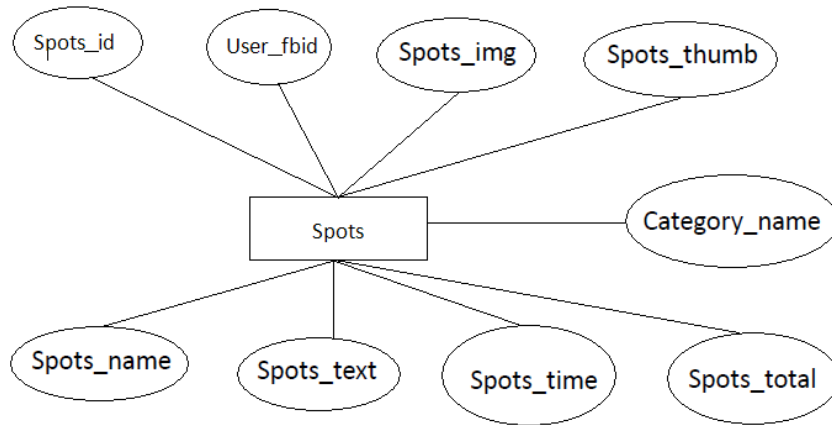


圖 2 Spots 的 ER 圖

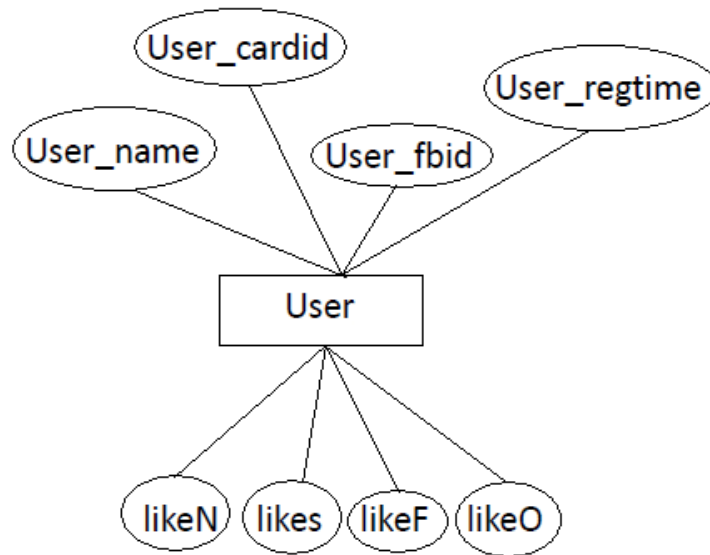


圖4 User 的 ER 圖

欄位	型態	校對	屬性	Null	預設值
spots_name	varchar(20)	utf8_general_ci		是	NULL
users_fbid	varchar(20)	utf8_general_ci		是	NULL
rating_score	int(11)			否	0
rating_time	text	utf8_general_ci		是	NULL
rating_name	text	utf8_general_ci		否	

圖 5 Rating 表格內容



欄位	型態	校對	屬性	Null	預設值
user_name	char(20)	utf8_general_ci		否	0
user_cardid	char(10)	utf8_general_ci		否	0
user_fbid	char(20)	utf8_general_ci		否	0
user_regtime	text	utf8_general_ci		否	
likeN	int(11)			是	1
likeS	int(11)			是	0
likeF	int(11)			是	0
likeO	int(11)			是	0

圖 7 User 表格內容

## 8. 系統實作

### 8.1 電子看版首頁設計

使用者由首頁登入景點網站時將有二個資訊可供瀏覽及選取，一是景點的陳列照片，二是連結至卡號登錄頁面。在景點照片的陳列方式中，我們會先從資料庫取得各景點所紀錄的數據，並依據景點被使用者分享、評論等數據由大至小排序。當使用者需要更詳細的景點內容，可以點選該景點圖示並轉往主題頁面，在主題頁面中使用者可以發表評論或分享內容，如系統判斷該使用者曾註冊過卡號，則頁面會額外顯示評分元件讓使用者評分。



圖 8 電子看版設計

上面功能列有各項附加功能可以使用，分別為 1.點擊瀏覽 QR 碼 2.問題與反饋 3.臉書討論區 4.線上交流 5.旅遊規劃，以下來作依序的說明

## 1. 點擊瀏覽 QR 碼



圖 9 QR 碼集中頁面

QR 碼的主要功能為在內部儲存文字、或連結在內，一但使用具有解析 QR 碼的 APP 掃描後即可解讀其內容，現在大多行動裝置皆可用此功能，非常的便利

## 2. 問題與反饋

### 新竹景點資訊服務系統 -錯誤回報-

如對於本網站內容有問題也可在此作即時的回報，  
我們收到的第一時間將會為您服務。

當然，別忘了填寫電子信箱好方便我們能夠在最快的時間回覆您。

※您在此填寫的回報與建議將不會公開在網路上。  
**\*必要**

**感恩您光臨我們的網站。\***

請選擇執行的項目

BUG問題回報

網站建議反饋

其他：

技術提供：Google 文件

[檢舉濫用情形](#) - [服務條款](#) - [其他條款](#)

圖 10 錯誤回報頁面

此功能主要是用來讓回報系統問題，以利管理者做改良的措施，是使用者與管理者溝通的界面，採用 GOOGLE 提供的雲端儲存空間來儲存使用者回報的問題

### 3.臉書討論區



圖 11 Facebook 討論專頁

此功能由如一般的討論區，不過是建立在 Facebook 上，配合現在人大部分都有 Facebook 帳號的情況下，只需登入即可在專頁上進行留言及討論

### 4.線上交流

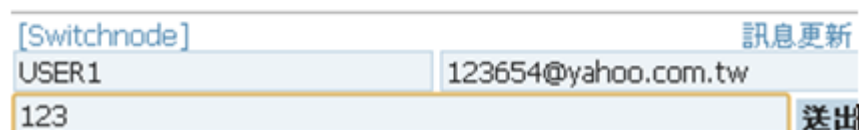
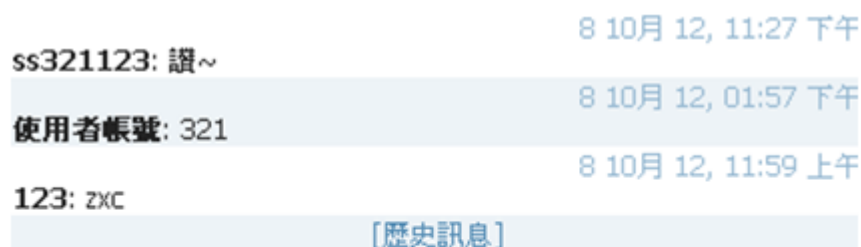


圖 12 線上交流頁面

可以把此功能看做線上聊天室來使用，只需輸入使用者姓名後即可進行各種發言，

屬於使用者與使用者溝通的界面

## 5. 旅遊規劃

此函式是用於設定 Google Map 起始狀態的控制項參數

```
function initialize2(pFrom,pEnd) {  
    var myOptions = {  
        zoom: 12,  
        center: new google.maps.LatLng(24.804032,120.96461),  
        mapTypeId: google.maps.MapTypeId.ROADMAP  
    }  
}
```

此函式是用於抓取使用者輸入的內容至 GoogleMap 旅遊模式

```
function calcRoute2(pFrom,pEnd) {  
    var start = pFrom;  
    var end = pEnd;  
    var request = {  
        origin:start,  
        destination:end,  
        travelMode: google.maps.DirectionsTravelMode.DRIVING  
    };
```

圖 13 旅遊規劃頁面

有兩種規劃的方式，一是自己選擇出發點和目的地；另外一種是由已建好的起點與終點下拉式選單直接選擇，按送出後會出現以下結果，A 為起點、B 為終點，在視窗的右邊可以看到詳細的走法與需要花費的時間和里程



圖 14 旅遊規劃結果顯示頁面

## 8.2 景點內頁設計



圖 15 景點內頁設計

在首頁選擇想要觀看的景點後進入在左側會出現使用者目前的資訊，中間則有景點像片和內容的介紹，點擊相片還會出現更多的照片，此部分是使用 javascript 寫成的秀圖 API



圖 16 圖片瀏覽畫面

在往下拉會有評分和推薦這兩個功能可以使用，使用者可以依自己的喜好來評分和決定是否推薦此景點，接著最下方的是留言功能，此部分引用 Facebook 所提供的 API 寫成。另外這裡也有提供旅遊規劃和 QR 碼這兩項功能在外加 GOOGLE MAP



圖 17 使用者操作區塊

GOOGLE MAP 與網頁的結合入圖 18 所示，可以知道想要前往景點的附近的地理位置和附近有什麼建築景觀

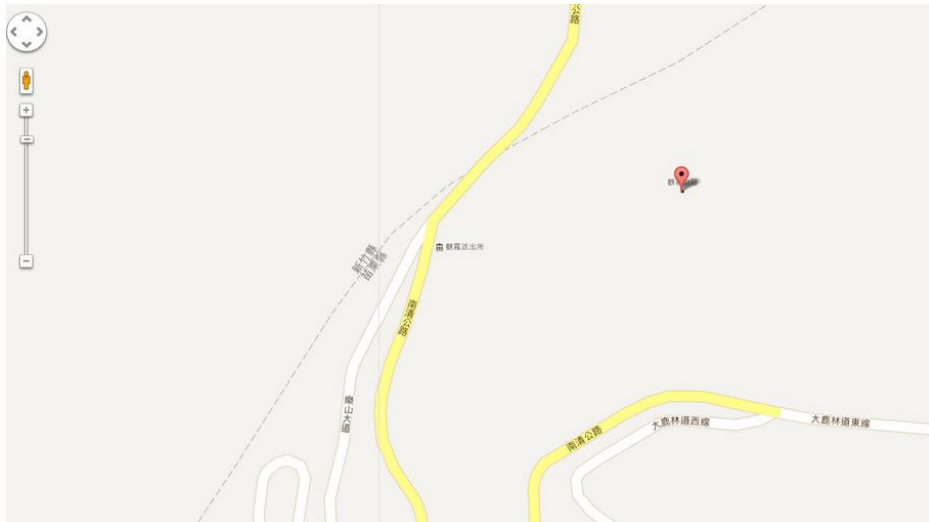


圖 18 Google Map

### 8.3 卡號與 RFID 的連接與註冊機制



圖 19 facebook 登入並註冊圖示

此部分的實作方式是以 Facebook 帳號來和卡號一起做認證註冊，因此一定要登入 Facebook 進行註冊後，之後所輸入的卡號才會有進行評分、推薦等功能

圖 20 Facebook 登入頁面

登入後須進行使用者的授權，因為會取得使用者的姓名，FBID 等資料



圖 21 應用程式授權頁



圖 22 使用者授權頁面

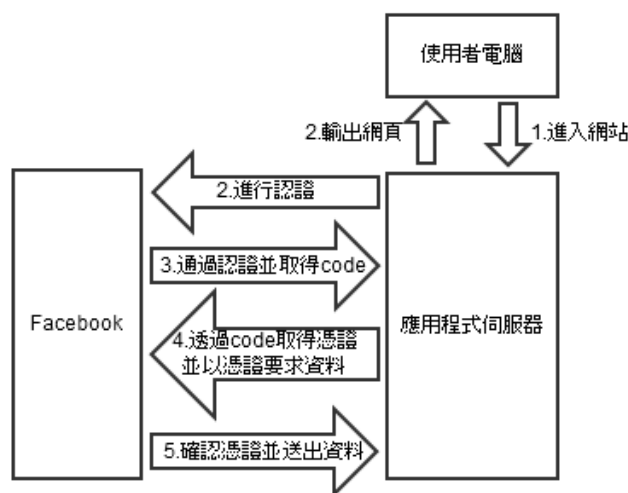


圖 23 卡號與 RFID 連接的示意圖

網頁設計是利用 PHP 語法，因此一開始必須下載官方專為 PHP 所提供的 SDK，接著套用 SDK 中的 Facebook.php 檔案及前面描述的應用程式識別資訊，讓網頁運作時即可透過 Facebook.php 檔案中的函數(Function)來存取資源，其 PHP 原始碼範例如下：



```

<?php
require './src/Facebook.php'; // 引入官方提供的 Facebook.php
$Facebook = new Facebook(array(
    'appId' => 'yourAPPID', // 應用程式 ID
    'secret' => 'your secret', // 應用程式私鑰
));
$user = $Facebook->getUser(); // 取得使用者 ID
?>

```

成功授權後即會出現刷卡頁面，此時只需要將卡片刷過 RFID READER 顯示卡號送出即完成卡號註冊，之後登入只需刷卡就可以，不需在登入使用者註冊成功後就可以回到首頁進行各項操作。



圖 24 註冊後的刷卡登入頁面



圖 25 RFID 使用示意圖

## 8.4 景點推薦機制

一般未登入使用者只會出現左邊的人氣景點推薦，登入後就會出現個人推薦，剛開始個人推薦會空值，因為還沒選擇喜好，在選擇個人喜好的景點類型後送出就會在右邊的系統推薦景點選出最適合的景點給使用者



圖 26 使用者尚未登入時的頁面



圖 27 使用者登入後的頁面

推薦機制的具體做法:

人氣景點推薦做法: 目前是採用每個景點的評分除以各個景點的使用者得到的總分進行比較選出分數前三高的景點做為人氣景點。

系統推薦景點做法: 採用各個景點的總推薦數依照不同的分類分別找出推薦數量前三高的景點, 然後再依據使用者的喜好來當作條件, 借以篩選出最適合目前使用者喜好的景點出來, 用此種方式來達到個人推薦的目的。

## 8.5 與社群網站的連接

只需在收所列輸入"新竹景點資訊站"就會找到相對應的專頁



圖 28 facebook 專頁搜索

點擊圖 29 的連結即可前往應用程式

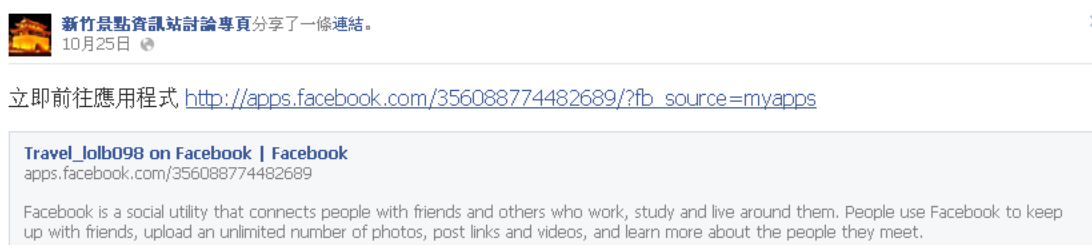


圖 29 應用程式超連結

進入後會出現如圖 30 的頁面



圖 30 進入 facebook 應用程式時的頁面

之後等待一下就會跳轉頁面到景點資訊站首頁

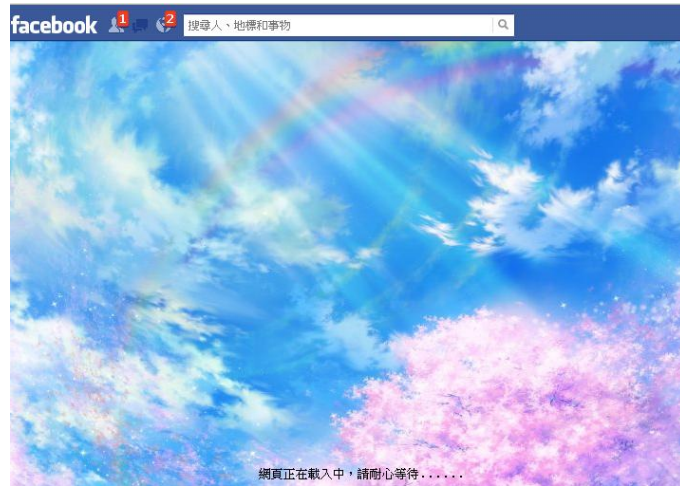


圖 31 載入程式等待畫面



圖 32 進入 facebook 上的應用程式頁面顯示

上傳程式到 Facebook 的平臺 heroku

Ruby 語言的雲端運算平台：Heroku

須使用 "GIT BASH" 這個程式才有辦法把寫好的程式在 Facebook 上運行

因此要先註冊一組帳號才可以使用

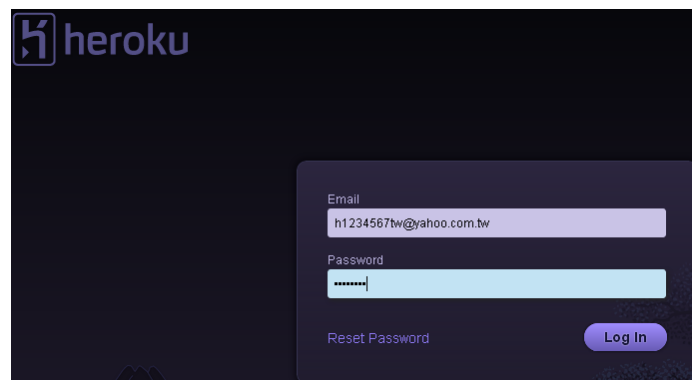


圖 33 heroku 登入頁面

註冊好之後進入 git bash 進行登入驗證

```
Enter your Heroku credentials.
Email: h1234567tw@yahoo.com.tw
Password (typing will be hidden):
Authentication successful.

vampire@KEVIN-417CAF6B2 ~
$
```

圖 34 git bash 登入驗證頁面

之後下指令把原本的目錄複製下來修改

```
vampire@KEVIN-417CAF6B2 ~
$ git clone git@heroku.com:pacific-beyond-7860.git -o heroku
```

圖 29 複製指令 clone 用以複製目錄

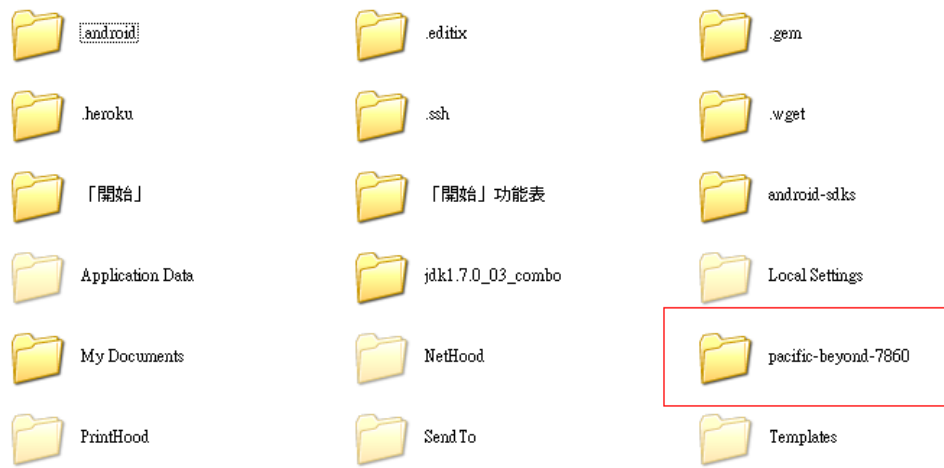


圖 35 從 heroku 複製下來的目錄

修改完後下以下指令上傳及完成開發的程式與 Facebook 的连接

```
vampire@KEVIN-417CAF6B2 ~/pacific-beyond-7860 (master)
$ git commit -am "changed greeting"
[master ba184d2] changed greeting
Committer: unknown <vampire@kevin-417caf6b2.(none)>
Your name and email address were configured automatically based
on your username and hostname. Please check that they are accurate.
You can suppress this message by setting them explicitly:

    git config --global user.name "Your Name"
    git config --global user.email you@example.com

After doing this, you may fix the identity used for this commit with:

    git commit --amend --reset-author

1 files changed, 29 insertions(+), 0 deletions(-)
create mode 100644 fb_header.php
```

圖 36 下修改指令 changed greeting

```
$ git push heroku
counting objects: 4, done.
Delta compression using up to 2 threads.
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 660 bytes, done.
Total 3 (delta 1), reused 0 (delta 0)
----> Git submodules detected, installing
Submodule 'sdk' (https://github.com/facebook/facebook-php-sdk.git) regist
ered for path 'sdk'
Initialized empty Git repository in /tmp/build_lqi442bw5jzck/sdk/.git/
Submodule path 'sdk': checked out '98f2be163c96a51166354e467b95dd38aa4b0a
19'
----> PHP app detected
----> Bundling Apache version 2.2.22
----> Bundling PHP version 5.3.10
----> Discovering process types
Procfile declares types -> (none)
Default types for PHP -> web
----> Compiled slug size: 12.1MB
----> Launching... done, v108
http://pacific-beyond-7860.herokuapp.com deployed to Heroku

To git@heroku.com:pacific-beyond-7860.git
5f30302..ba184d2 master -> master
```

圖 37 把修改完後的檔案上傳時的頁面

### 已註冊者的登入

如果卡號已註冊過，直接點擊連接卡號的部分，然後進行刷卡即可登入，因為此時的使用者不適用 Facebook 登入，而是取得資料庫內的卡號與現在刷的卡號比對選擇出的使用者，因此不想 Facebook 的綁定帳號，所以用把使用者存在 cookie 的方式來達成，並且設一段存活時間，若超時則要重新登入



圖 38 註冊後的快速登入區塊

如果未註冊過刷卡的話會出現以下畫面



圖 39 未註冊過的卡號登入時顯示的訊息

## 9. 行動端程式設計

每一個 Android 應用程式都可以視為一個獨立的專案，其完整的 Android 專案是由下列三個檔案所組合而成：

1. AndroidManifest.xml：專案描述檔，也是 Android 應用程式開發必要的檔案，其描述應用程式的資訊以及定義各種元件，其專案描述檔內容圖 40。

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="chunghua.csie.apas"
    android:versionCode="1"
    android:versionName="1.2">
    <uses-sdk android:minSdkVersion="7" />
    <uses-permission android:name="android.permission.RECEIVE_BOOT_COMPLETED">□
    <uses-permission android:name="android.permission.INTERNET"></uses-permission>

    <application android:icon="@drawable/icon" android:label="@string/app_name">
        <activity android:name=".APASActivity">□
        <!-- 委派receiver名稱類別為AutoBoot -->
        <!-- 取消開機自動執行的程序 -->
    </application>
</manifest>
```

圖 40 專案描述檔

2. Java 原始檔：其按照標準的 Java 語言撰寫方式，並可利用各種 Java 類別以及 Android 類別，其 Java 原始檔內容如下圖 41：

```
* Copyright 2011 Chung Hua university CSIE.□

package chunghua.csie.apas;

import android.content.Context;□

// Autoboot: 捕捉android.intent.action.BOOT_COMPLETED的Receiver類別
public class AutoBoot extends BroadcastReceiver {
    @Override
    public void onReceive(Context context, Intent intent) {
        if(intent.getAction().equals(Intent.ACTION_BOOT_COMPLETED))
        {
            // 當收到Receiver時，指定開啟APASActivity程式
            Intent bootActivityIntent = new Intent(context, APASActivity.class);
            bootActivityIntent.addFlags(Intent.FLAG_ACTIVITY_NEW_TASK);
            // 將Intent以startActivity傳送給作業系統
            context.startActivity(bootActivityIntent);
        }
    }
}
```

圖 41 Java 原始檔內容

3. 資源檔：包含 XML 檔案、圖片、原始資料檔等，其中又以介面佈局(Layout) 檔最為重要，下圖 42 為佈局檔格式內容：

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    >
<TextView
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:text="@string/hello"
    />
</LinearLayout>

```

圖 42 佈局檔案內的格式

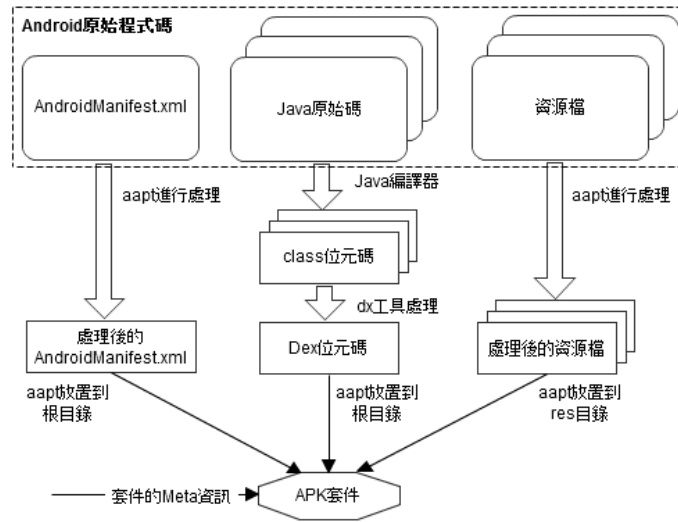


圖 43 Android 應用程式的編譯過程



評分:



名稱：新竹城隍廟

介紹：新竹城隍廟位於新竹市北區中山路上，列屬二級古蹟，其廟宇規模在當時是全台灣之最，專門掌管陰間與陽間賞罰善惡的神明為城隍，所以城隍廟的佈置猶如古代的衙門；新竹都城隍廟廟埕的市集小吃攤，成為新竹特色之一。

圖 44 行動端頁面的呈現

行動端的頁面呈現如此圖 44，上方的拉條每 30 秒會換成下一個景點，也可以直接移動拉條切換下一景點



---

[點擊前往討論區](#)  
[點擊線上交流](#)  
[點擊問題與反饋](#)  
[點擊GOOGLE MAP路徑規劃](#)  
[點擊切換為網頁版](#)

圖 45 行動端的超連結部分

行動端主要的功能為瀏覽，如果要做留言或是推薦等功能時需點擊如上圖中的超連結到網頁端去進行操作

## 10. 結論與未來展望

在本專題中提出一個以 Android 平台為基礎的個人化景點資訊系統，其最主要目的是將個人化的推薦機制整合於以 Android 平台所建置的景點數位看板上，以期協助外來遊客能在不熟悉旅遊的景點下，能隨時拿著旅遊版的悠遊卡在所有設置的系統中查詢個人喜好的景點資訊。在整個系統開發中，我們驗證以下幾項功能：

1. 建構一套以 Android 平台為主的數位看板系統。
2. 整合 Facebook 社群網路資源於 Android 平台上。
3. 社群個人資料與悠遊卡結合達到辨識功能。
4. 數位看板的個人化資訊推薦功能。
5. 系統具有實用性、低成本、及高擴充性。

在後續的發展上，我們將繼續研究個人化的推薦方式，並探討社群上各族群的關係，如朋友、家人、粉絲團等關係來加強使用者對推薦系統的信賴度，並試圖減少評比資料量來降低系統負擔；除了在旅遊資訊外，也可將範圍擴充到不同的領域上，如展場與博物館的應用、個人化的學習看板；目前 Android 看板上主要以圖片展示為主，內容可以再強化，如影片與 Flash 動畫；在 RFID 識別方面仍然無法有效整合至 Android 平台上，導致識別功能是以讀取字元方式判別，這容易造成系統的誤判及安全性問題，而 Android 目前已加強近距離通訊功能及 API，在後續也將繼續探索此功能並應用在數位看板的個人識別上。

## 結語與銘謝

這次的專題製作讓我們對軟體方面有更進一步的了解，經由學習程式語言以

及系統分析，軟體工程等等，幫我們打下良好的基礎。在製作過程中，整體團隊合作以及彼此獨立思考解決問題的能力，相信這個過程會是我們最大的收穫。由於建置整個討論平台的架構中，製作過程有些辛苦，但是利用 Facebook 和 Google 龐大的 API，使我們省去了一些時間。最後我們利用了團隊的合作精神，分工合作的方式，克服了重重的關卡，不但增進了彼此的感情，也如預期的執行並完成老師的期許與要求。

在此專題製作之中，我們遭遇許多問題，尋找很多解決方法，查詢相關書籍內容、上網查資料等，其中也包括詢問我們的專題指導老師 - 張欽智老師，若有疑問或者是我們的想法，經過與張欽智老師的討論，也會給予我們很大的受益。在此感謝此專題製作互相合作的組員，同時也非常感謝所有曾幫助過我們專題的人。

## 參考文獻

- [1] 陳婉菁、簡佩萍、陳彌彰 (2009)，數位電子看板創新應用案例探討，2009年9月
- [2] 莊清男、陳彥良 (2005)，協同過濾式群體推薦，國立中央大學資訊管理研究所碩士論文
- [3] 黃智偉 (2011)，塗鴉牆的秘密-Facebook Graph API 實戰開發手冊，佳魁資訊
- [4] 鍾文昌、梁文耀 (2009)，Android 作業系統移植之研究與實現，國立台北科技大學資訊工程系碩士論文
- [5] 韓超、梁泉 (2010)，深入淺出 Android 系統原理及開發要點，博碩文化
- [6] 羅建銘、陳銘崑 (2001)，協同過濾於網站推薦之研究，國立台北科技大學商業自動化與管理研究所碩士論文
- [7] 羅濟群、葉靜蓉、程鼎元、楊慕均、陳志華 (2010)，基於社群網路行為的決策智慧推薦分析系統-以股票投資為例，2010年台灣網際網路研討會
- [8] Wei Chen and S. Fong (2010), "Social Network Collaborative Filtering Framework and Online TrustFactors: a Case Study on Facebook", 2010 Fifth International Conference on Digital Information Management (ICDIM), pp. 266-273.