

中國文化大學

資訊工程學系

資訊系統專題

多功能生活應用食品管理系統

學 生：蔡 佩 蓉

杜 宜 家

陳 凱 恩

詹 孟 陵

指導教授：徐 盛 軒

中 華 民 國 105 年 5 月

多功能生活應用食品管理系統

專題學生：陳凱恩、詹孟陵、蔡佩蓉、杜宜家

指導教授：徐盛軒 博士

中國文化大學 資訊工程學系

摘要

現代科技日趨進步，衣住行育樂都獲得很大的滿足，但在“食”方面卻沒有，從食品的生產、購買、飲食再到食品保存逐漸被重視，對此產生較大的問題是，民眾無法親自把關每一個環節，所以不容易掌握食品的資訊，更別說要利用這些資訊有效率的管理自家現有的食品。

於是我們進行研究，期望開發一個食品管理系統，讓民眾可以快速掌握食品的即時資訊，進一步結合線上資訊，讓應用程式更貼近日常生活，並且活化生活。

這是一個對自家食品進行管理的系統，管理項目有食品安全、庫存狀態以及料理，目的是讓民眾可以吃的安心也吃的開心，即便是剩餘的食材，也能透過應用程式找到合適的食譜；利用應用程式的購物清單，準確的購買必要的食品；簡單操作的查詢系統，使民眾快速取得食品的生產履歷；購物後，即時更新自家食品的狀況。

相較於以往，我們希望應用程式不只有提供訊息讓使用者自行判斷，而是能將訊息與現況結合，提供便利實際的生活應用。

關鍵詞：食品管理系統、食品履歷、智慧食譜搜尋、自動化購物清單

指導教授_____ (簽名)

目錄

摘要.....	I
表目錄.....	IV
圖目錄.....	V
第 1 章 研究動機與目的	1
1.1 研究動機	1
1.2 研究目的	2
第 2 章 文獻探討	3
2.1 相關應用程式	3
2.1.1 IQC 應用程式	3
2.2 開發環境	5
2.2.1 Python 3.0	5
2.2.2 C# Visual Studio 2015	6
2.2.3 PhoneGap.....	7
2.2.4 MySQL	8
2.2.6 Node.js	9
第 3 章 研究內容與方向	10
3.1 軟硬體需求	10
3.1.1 開發階段硬體需求	10
3.1.2 開發階段軟體需求	10
3.1.3 執行階段硬體需求	10
3.2 食品管理系統	12
3.2.1 架構圖	12

3.2.2 系統伺服器	13
3.2.3 管理員	15
3.2.4 使用者應用程式	16
第 4 章 人力配置	25
第 5 章 執行速度	26
第 6 章 現階段成果	27
6.1 資料庫	27
6.2 應用程式	30
第 7 章 預期成果	33

表目錄

表 4.1 人力分配表	25
-------------------	----

圖目錄

圖 2.1 IQC 應用程式運作模式	4
圖 2.2 Python logo	5
圖 2.3 C# logo	6
圖 2.4 HTML5 提供聚合性質	7
圖 2.5 MySQL logo	8
圖 2.6 Node.js logo	9
圖 3.1 系統架構圖	12
圖 3.2 伺服器跟資料庫供需圖	13
圖 3.3 伺服器跟管理員供需圖	13
圖 3.4 伺服器跟行動裝置供需圖	14
圖 3.5 管理員維護架構圖	15
圖 3.6 系統功能圖	16
圖 3.7 註冊會員流程圖	17
圖 3.8 食品查詢流程圖	18
圖 3.9 食品管理流程圖	19
圖 3.10 食品管理資料流	20
圖 3.11 食譜搜尋流程圖	21
圖 3.12 食譜搜尋資料流	22
圖 3.13 購物清單流程圖	23
圖 3.14 購物清單資料流	24
圖 5.1 專題進度甘特圖	26
圖 6.1 食譜資料庫關聯圖	27

圖 6.2 查詢 SQL 語法	28
圖 6.3 食譜查詢結果	29
圖 6.4 應用程式圖示	30
圖 6.5 應用程式初步介面	31
圖 6.6 二維掃描器	32

第 1 章 研究動機與目的

1.1 研究動機

現在到賣場買東西，雖然買到過期食品的機率偏低，但每當要查看食品的詳細資料都會感到不方便，每種商品標示詳細資料的位置都不同，更別說有些標示的字會很小；一般來說不會在出門前，徹底翻遍冰箱確認家裡的食物還剩甚麼，僅憑印象購買食品，在沒有清單的情況下就有可能買到重複的食品；有時會想換換菜餚風味，但苦於沒有豐富的做菜經驗，沒能嘗到不同的滋味。

在這些情況下，其實家庭主夫/婦在採購食品與下廚的時候都會有一些問題產生，像是購買過多食品以致過期、煮太多菜吃不完、一成不便的家常菜或是找不到食品的詳細資料等，如果有個東西可以提醒家裡食物的狀況、解決做菜的困擾、減少食物的浪費、快速明確的食品標示，也許能給生活帶來很大的幫助。

因此與老師討論出，將人們的需求自動化，可以提醒食品現況、提供食譜與步驟、購買時有一個簡便的清單、快速查詢食品履歷，希望能解決普遍會遇到的困擾，也能達到食品環保減少浪費。

1.2 研究目的

我們希望開發出一個可以管理食品的系統，藉由此系統，能協助人們快速取得資訊，簡單管理食品，輕鬆購買食材。

透過討論與研究，我們將應用程式分為三個部分：

(1) 管理員

管理員維護的對象是系統資料庫與系統伺服器，可以新增各項資訊，對會員資料行管理，或是當系統發生運作失常時可以進行維護。

(2) 系統伺服器

這部分要處理的是，出貨者輸入食品履歷後就要產生二維條碼，再來是與賣場資訊做連結，每個商品的售價與進貨的商品做對應。最後結帳時，要能產生結帳清單，自動與使用者家裡的食物狀況做彙整。

(3) 使用者應用程式

可以解決使用者從購物到下廚會遇到困難，首先，食品管理功能會提醒食物的狀況，如：快過期的、剛買的等等，給使用者建議的食譜，或者使用者可以自行選擇食譜，選定食譜後，提示使用者缺少的食材與現有的食材，在賣場只要簡單操作應用程式就可以看到食品履歷，結帳後不需手動輸入，就能即時更新食物狀況。下一次要購物時，使用者還可以預先在 APP 上做欲購買的記號，有購物清單的提醒，就可輕鬆又環保的購買食材。

第 2 章 文獻探討

2.1 相關應用程式

我們在設計系統前，尋找其他相關的應用程式，對此了解與研究人們的使用習性，並從中發現我們可以創新與提升的地方。

2.1.1 IQC 應用程式

現在智慧型手機便利且價格並非天價，一般民眾也可以擁有，使用上日趨普遍，許多功能、APP 產生後，讓應用性質更加多元，搭車隨時可以查看時刻表、線上訂購商品、通過文字或語音的聊天等，手機的功用不再只有打電話、傳訊息，多元的應用使得資訊網快速擴張，民眾可以即時獲得更多資訊，當發生社會問題時，這些資訊不再被遺漏或隱藏，除了快速取得資訊外，還能加以應用在生活上。

食安問題常常是人們所顧慮的，像是食品來源、中間的過程等，民眾無法自己去監督每種食品的處理流程，所以只能檢查食品上的標示，就算食品安全，在食品的管理上也有很多問題產生，忙碌的現代人無法一一管理食品的狀況，所以我們透過研究食譜 APP 與食品管理 APP，設計新系統重組功能達到我們的目的。

圖 2.3 為 IQC APP 商品安全資訊網的運作模式，建置一個食品安全平台，讓民眾可以上網查詢食品的資訊，如：此商品有沒有檢驗合格等，再利用專屬的手機 APP 掃描商品上的條碼，就能辨識這件商品的檢驗結果。

如果遇到政府食品的法規變更、商品期限不符或過期、未通過檢驗的商品等狀況，IQC 利用雲端即時同步資訊，更正商品資訊，並將不合格的商品拿掉。IQC 的目的是為了讓食品安全透明化，不只要求

廠商做好食品把關、提供完整的食品資訊，讓消費者擁有完整資訊可以買到合格的商品，得到消費的保障，也藉此淘汰黑心商品。



圖 2.1 IQC 應用程式運作模式 [1]

IQC APP 具有掃描條碼的功能，民眾在掃描後可以取得商品的檢驗結果，若沒有檢驗結果顯示，有兩種狀況，一是廠商沒有提供相關資訊，民眾可使用 APP 上的聯署功能，共同勉勵廠商提供完整資訊；二是此商品還沒有建檔，回報系統後，會將此商品做紀錄，作為建檔的先後順序。此 APP 也有提供關鍵字查詢，可依據品名、廠商、品牌等等字眼去搜尋商品，只要有建檔的項目都可以在平台上找到資訊。

2.2 開發環境

設計系統時發現我們使用的工具有很多種，於下列舉說明。

2.2.1 Python 3.0

Python 是一種以物件導向、直譯式逐句編寫的程式語言，且提供了更完善的標準函式庫，讓開發人員能輕鬆地完成常使用的任務。它與其他程式設計語言較為不同的是使用“縮排”代替“{ }”來定義語句的範圍。

能夠有效的管理記憶體，刪除不必用到的記憶體回收資源。常用來編寫網路程式、管理系統，和各種高階任務。在大多數的作業系統中皆可以使用 Python。可透過 Py2exe、PyPy 與 PyInstaller 等程式將 Python 原始碼脫離 Python 直譯器獨立執行。

Python 基本的函示提供數字、字串、串列、字典、檔案等使用，而完善的函示庫提供系統管理、網路通訊、文字處理、資料庫介面、圖形系統、XML 等額外功能。



圖 2.2 Python logo [2]

Python 的套件 BeautifulSoup，回復時間快速，如網頁抓取。特點有提供簡單的方法、語法、搜索和修改解析樹，不需要太多程式碼；自動將輸入的文件轉成 Unicode 輸出的轉成 UTF-8，除非文件沒有指定編碼且 BeautifulSoup 無法檢測；BeautifulSoup 的 lxml 和 html5lib 在 Python 中常被使用於不同分析策略或交易速度。 [3]

2.2.2 C#

C#是一種程式語言，設計用來建置可在 .NET Framework 上執行的多種應用程式。C#是簡單、強大、類型安全且以物件為導向的語言。C# 中使用了許多創新的做法，能夠提供快速的應用程式開發，同時保留了 C 樣式語言的表達能力與雅緻。

Visual C# 是 Microsoft 的 C# 語言實作。Visual Studio 支援 Visual C#，包含功能完整的程式碼編輯器、編譯器、專案範本、設計工具、程式碼精靈、功能強大且易於使用的偵錯工具，以及其他各種工具。.NET Framework 類別庫對於廣泛的作業系統服務，以及其他有用、設計良好、可顯著加速開發週期的類別，提供存取的支援。



圖 2.3 C# logo [4]

- (1) 使用 ZXing.Net 函式庫製作 QRcode。ZXing.Net 是一個 Apache License 2.0 的開放函式庫。
- (2) 其中引用 Zxing 的 DLL 使用其條碼產生跟讀取的功能。

2.2.3 PhoneGap

這個世代手機普遍到人手一支，但不是每個應用程式都能在不同的平台上運作，目前以 Android 和 iOS 兩大平台為主流，如果這個應用程式只能在某一平台上運作，對於它使用的廣泛度就會大大降低，為了解決應用程式跨平台的問題，我們選擇用 PhoneGap 開發程式。

現在的程式運作以「一次編寫，到處執行」的趨勢前進，對於這樣的新興模式，HTML5 適於作為開發工具，如下圖 A，HTML5 在各領域中是共同元素，它所提供的聚合(convergence)性質，可以跨裝置、作業系統、瀏覽器與應用程式，換句話說，開發人員只需撰寫一次應用程式，就能部署到多個平台上，不只可以有效減少開發作業，還能讓應程式更加普遍的使用。



圖 2.4 HTML5 提供聚合性質 [5]

HTML5 也有一些限制，尤其欠缺 API，由於一般的網頁應用程式基於安全性考量，無法直接或完全獲得裝置的裝置的存取權限，但很多應用程式會需要用到智慧型手機的特殊裝置，如：GPS、平衡感

測、照相等；另一個限制是瀏覽器的支援，每個瀏覽器能提供的支援程度不同，因此，在各行動裝置的瀏覽器上所呈現的功能與執行效果也不盡相同，所以選擇撰寫混和應用程式的方式解決這些問題。

我們所選用的 PhoneGap 是一個開放原始碼，以 HTML5 為開發基礎，使用 HTML 及 JavaScript 語法就可以開發應用程式，之後再將應用程式封包，讓行動裝置存取。由於，不同的平台會需要各種原生 API 和應用程式的開發方式，若要為每個裝置都開發一個專屬的應用程式，不只耗時又耗力，若在 PhoneGap 的框架下開發，且符合網頁標準建置應用程式，就能解決跨平台的問題，將應用程式部署到多種平台，即是不同的行動裝置也能運作。

2.2.4 MySQL

適合個人或中小型企業，因為他規模較小、功能有限，但好處則是開放原始碼。多使用由 Linux、Apache、MySQL、PHP 組成的架構建置網頁，也因都為開放原始碼，所有人都可以使用和編輯，降低了不少成本。

MySQL 資料庫是支援不同的後端的客戶端與伺服器系統，有廣泛的 API 用於不同的客戶端程式、程式庫、管理工具。在實作中我們將使用 node.js 作為網頁伺服器與 MySQL 資料庫，由於 node.js 沒有內建 MySQL API，所以執行 `npm install mysql` 來安裝模組。



圖 2.5 MySQL logo [6]

SQL 是高階語言，允許在高層資料結構下執行。資料並不指定儲存方式，在操作查詢，新增，修改，刪除等功能時可使用巢狀式語句，使指定修改資料更為便利與實用。大多在語句末以分號做為結束；在處理完上一個命令後會回到“mysql”等待下一個命令；查詢以表格方式顯示。資料來自於多個 tables，而非從一個巨大的 table，資料庫組成用來加快運作速度的檔案。

2.2.6 Node.js

Ryan Dahl 為解決伺服器效能問題，在 2009 年發表 Javascript 環境，並解設計了使用容易的 API。無阻塞與異步架構需要花費長時間來完成活動，但靈活實用。

其特性有：跨平台(OS X、Windows、Linux)、開放原始碼、伺服器端、網路應用執行環境、JavaScript、無阻塞、事件驅動、異步架構。



圖 2.6 Node.js logo [7]

Node.js 不同於其他語言需要透過其他伺服器才能使用，可作為伺服器供給使用。 [8]載入 HTTP 模組：`var http = require('http');`

並且需要一個函數來回應 HTTP 請求

```
var handleRequest = function(request, response){  
  response.end("HelloWorld ! "); }  

```


第 3 章 研究內容與方向

在開方此專案的軟硬體需求及其食品管理系統方向和系統伺服器都會在以下詳細解說。

3.1 軟硬體需求

在應用程式的開發與執行階段，我們使用了不同的程式開發環境及裝置，並用繪圖軟體加以表達。

3.1.1 開發階段硬體需求

應用程式需要跨平台運作，所以需要多種系統：Windows 系統的電腦、OS X 系統的電腦、伺服器、MySQL 資料庫、Android 系統的手機、iOS 系統的手機裝載軟體開發此系統。

3.1.2 開發階段軟體需求

應用程式需要跨平台運作、優化使用介面，我們使用以下軟體：

- (1) 使用 Cordova 撰寫 iOS 系統與 Android 系統的應用程式，使用 Html、JavaScript、JQuery 等語言撰寫前後端程式
- (2) 使用 Bootstrap、CSS 優化網頁之介面
- (3) 使用 Python- BeautifulSoup 撰寫，收集資料
- (4) Android SDK

3.1.3 執行階段硬體需求

我們使用 Android 與 iOS 行動裝置以下裝置測試應用程式，伺服

器、條碼掃描器、發票機配合應用程式的功能。

3.2 食品管理系統

此應用程式包含不同的運作區塊，維護區塊、管理區塊、使用區塊等。

3.2.1 架構圖

圖 3.1 是整個系統的架構圖，以伺服器為系統中樞，連結三個部分為資料庫、管理員以及行動裝置，伺服器可以到系統資料庫取得相關資料；管理員操作電腦，當系統運作失常時可以進入系統修復；使用者藉由行動裝置操作食品管理應用程式，點擊程式上的功能，向伺服器提出要求，系統經過運算後回覆正確的結果，且可從外部的實體，如：購物清單、二維條碼等，通過行動裝置功能取得相關資訊。

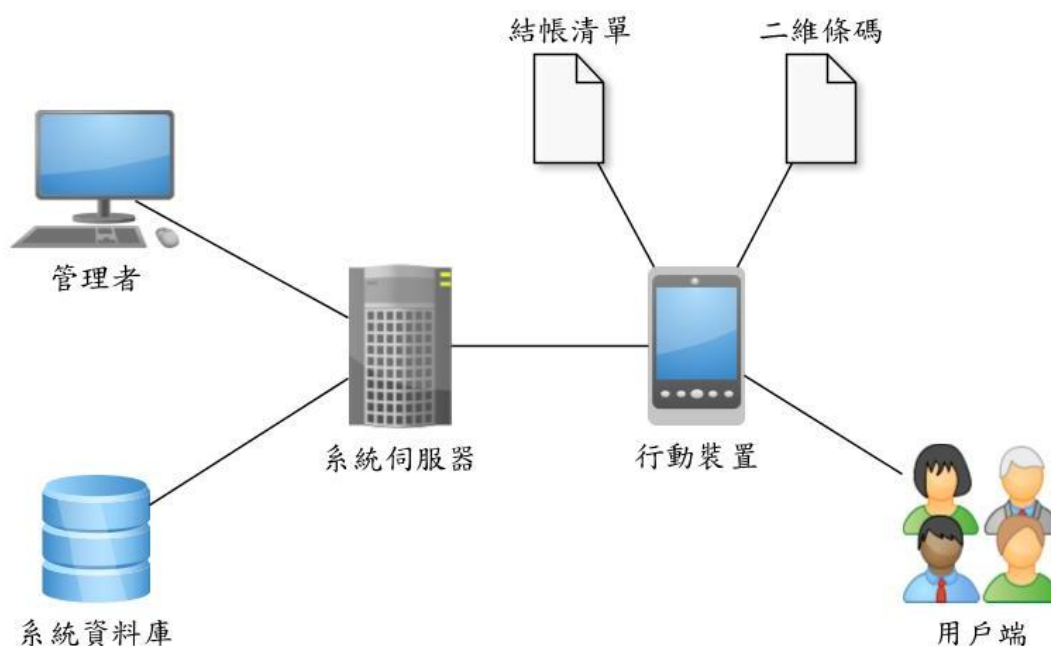


圖 3.1 系統架構圖

3.2.2 系統伺服器

系統伺服器是在資料庫、管理員以及行動裝置間運作，以下分三個部分做說明。

(1) 伺服器與資料庫

圖 3.2 是伺服器與資料庫的供需圖。資料庫存放各類資料，如：使用者註冊時產生的會員資料、食品與食譜的資料等，當裝置上提出要求時，會透過伺服器從資料庫取得相關的資料。

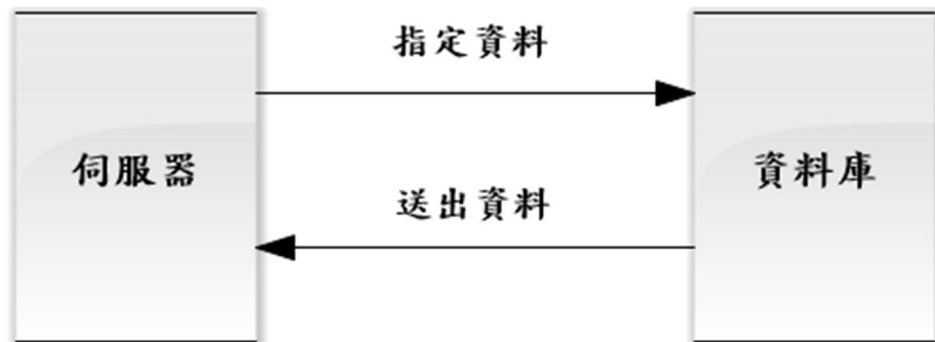


圖 3.2 伺服器跟資料庫供需圖

(2) 伺服器與管理員

圖 3.3 是伺服器與管理員的供需圖。管理員會將新資訊匯入系統，或是當系統發生運作問題時，管理員會進行維護。



圖 3.3 伺服器跟管理員供需圖

(3) 伺服器與行動裝置

圖 3.4 是伺服器與行動裝置的供需圖。使用者操作行動裝置時，應用程式會依使用者的操作項目向伺服器提出要求，如：使用者註冊後系統會存取會員資料、應用程式提出需求後，伺服器會回覆正確結果等。



圖 3.4 伺服器跟行動裝置供需圖

3.2.3 管理員

管理員在維護作業上有兩個部分，分別為系統資料庫及系統伺服器。

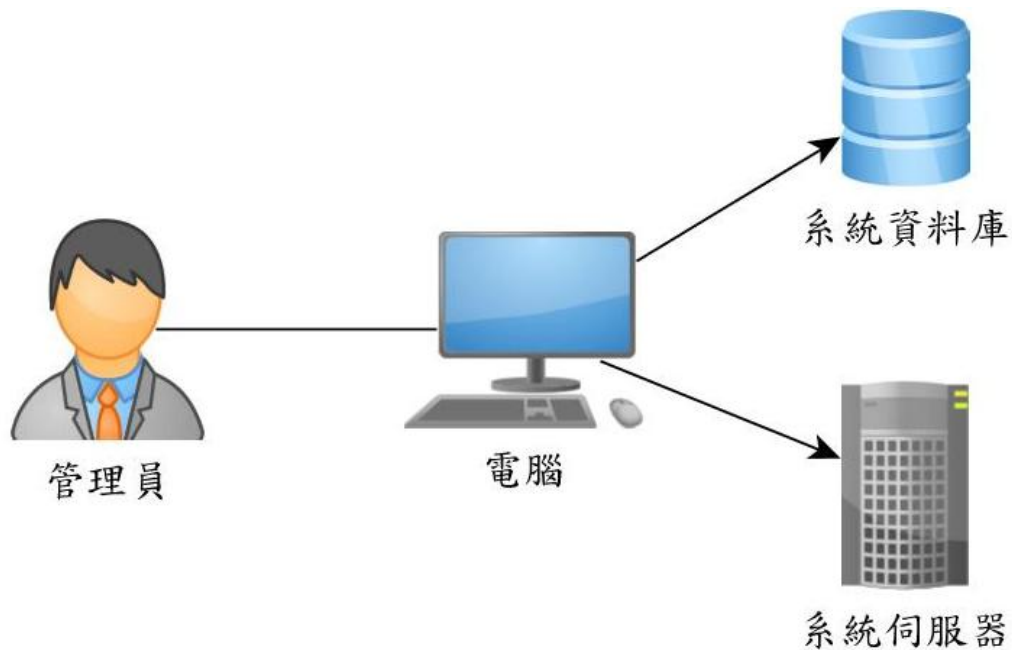
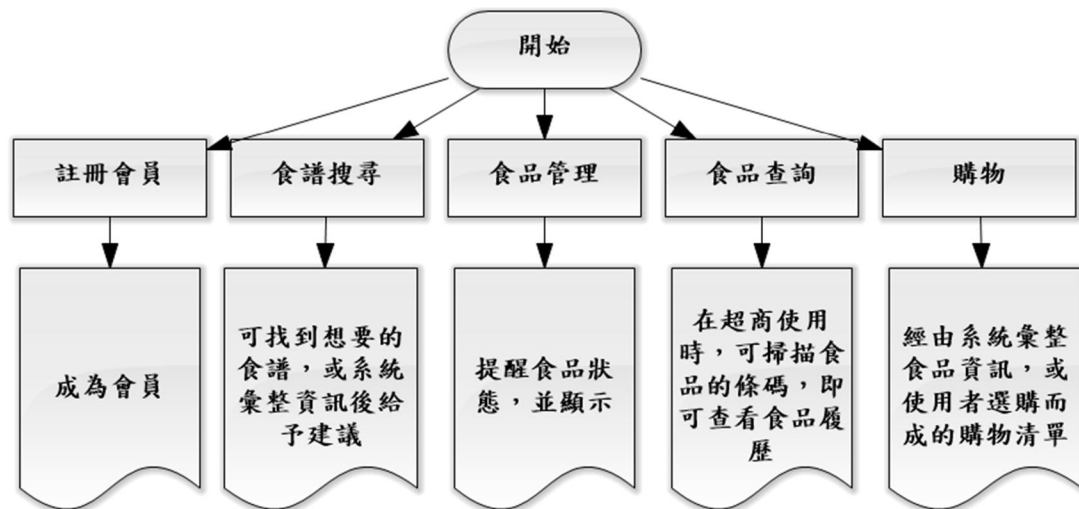


圖 3.5 管理員維護架構圖

由圖 3.5 知道管理員可以匯入、建置並管理系統的相關資料，系統資料庫發生遺失或損毀的情況，管理員進入資料庫進行維護，重新建置。

3.2.4 使用者應用程式

圖 3.6 是應用程式的系統功能圖，主要分為五個功能：註冊會員、食品查詢、食譜搜尋、食品管理與購物清單。



(1) 註冊會員

圖 3.7 是註冊會員流程圖，含有兩支流：註冊流程與登入流程。註冊流程是在使用者第一次使用應用程式的情況下，使用者必須先進行註冊，輸入完整資訊，確認無誤後，系統會建置新會員，之後使用者可以直接進入系統；登入流程則是使用者已經是會員的情況下，進入應用程式時，輸入帳號密碼進行身分驗證，驗證成功即可進入系統，驗證不成功則須再驗證一次，若忘記密碼，系統會寄送驗證碼到使用者的信箱，使用者輸入驗證碼後，先更新密碼，並再次登入才能進入系統。

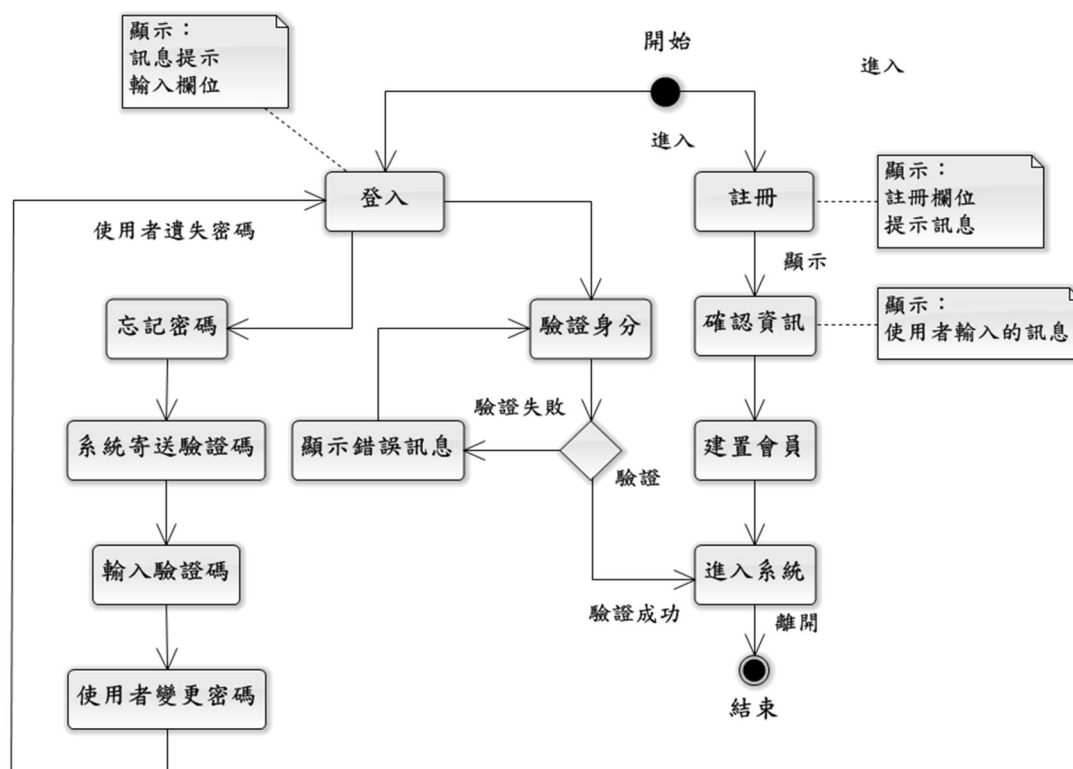


圖 3.7 註冊會員流程圖

(2) 食品查詢

在購買商品前，總會先查看食品的生產日期、有效期限、營養標示等資訊，有時會遇到標示不明，或是翻來翻去就是找不到有效期限，所以我們設置掃描條碼的功能，讓使用者可以輕鬆取得需要的訊息。

圖 3.8 是食品查詢流程圖，使用者進入系統即可看到掃描按鈕，點擊後會出現對焦視窗，對準掃描目標等待大概兩秒，即會顯示此食品的生產履歷。

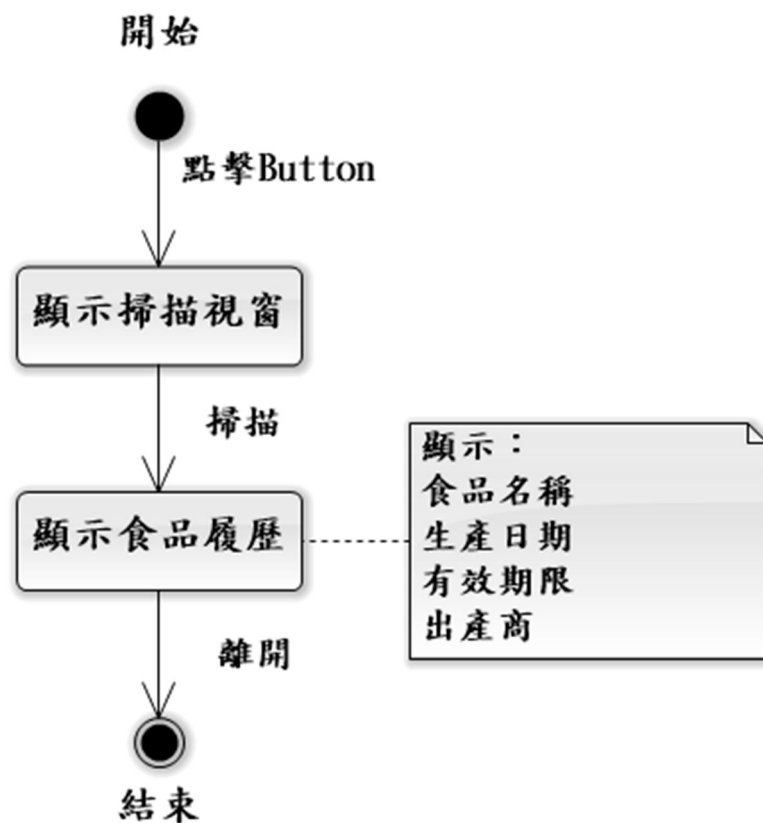


圖 3.8 食品查詢流程圖

(3) 食品管理

雖然冰箱和儲食櫃常使用，卻無法即時又精確的掌握食品的保鮮狀況，如：購買日期、有效期限、食品數量等，為了解決這些問題，我們設計食品管理系統不只隨時彙整新舊資訊，還可自動提醒食品的狀況。

圖 3.9 是食品管理系統流程圖，在使用者到賣場購物前，食品管理系統會先提醒已存有的食品狀況，如：快過期、已過期、現存的食物等，使用者也可以自行查看所有食品的狀況，此時還沒有新購入的食品項目；在購物後，掃描發票上的二維條碼即可將購買的項目新增至食品管理系統。

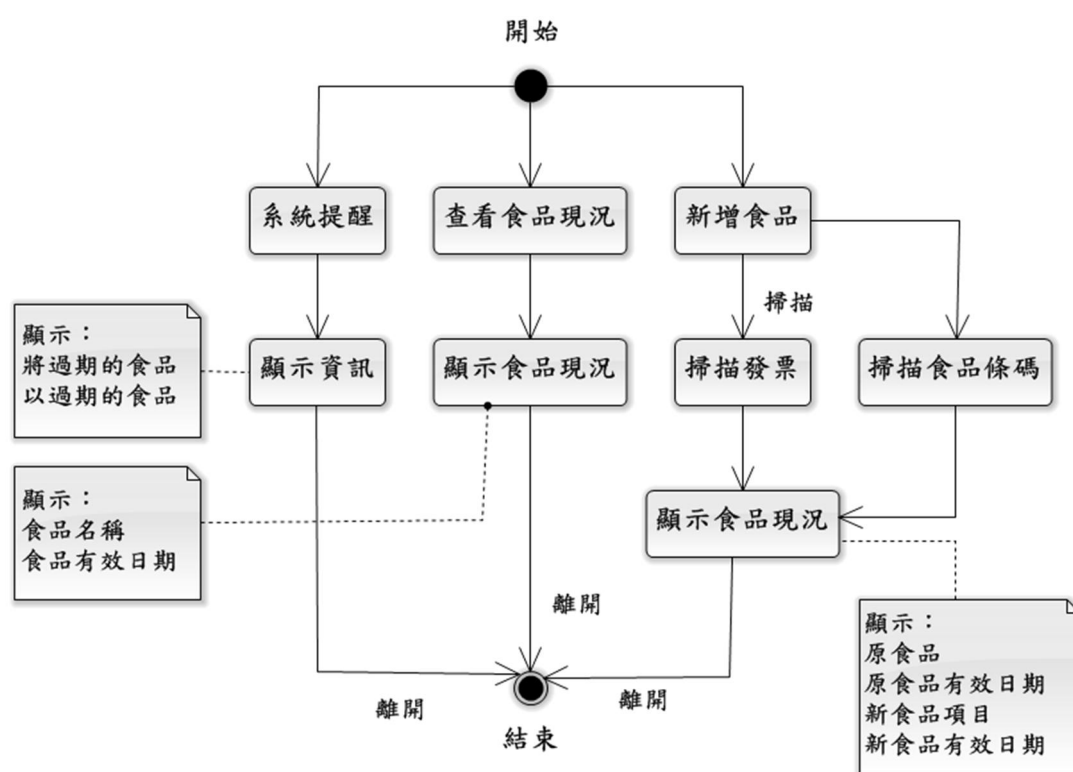


圖 3.9 食品管理流程圖

圖 3.10 是食品管理系統的資料流示意圖。食品管理系統會有兩個資料來源，一是家裡存有的食品，二是使用者結帳後所產生的購買清單，在系統內進行彙整產生完整的訊息。

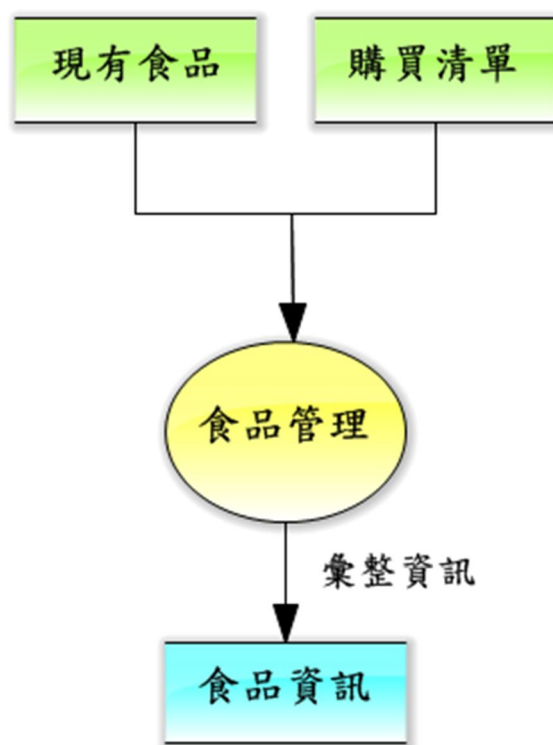


圖 3.10 食品管理資料流

(4) 食譜搜尋

在沒有豐富做菜經驗或沒有人指導的情況下，菜色通常簡單普遍，若參考食譜的書籍，往往要買家裡沒有的食品，用剩的食材也不一定會處理，於是我們設計“食譜搜尋系統”來解決這個問題。

圖 3.11 是食譜搜尋流程圖。使用者可以依喜好輸入食品或食譜的關鍵字搜尋食譜，或是讓系統彙整食品現況自動搜尋，提供合適的食譜，並顯示食譜；使用者可以進入食譜查看詳細資料，還能標記整個食譜或各項食品項目放入購物清單。

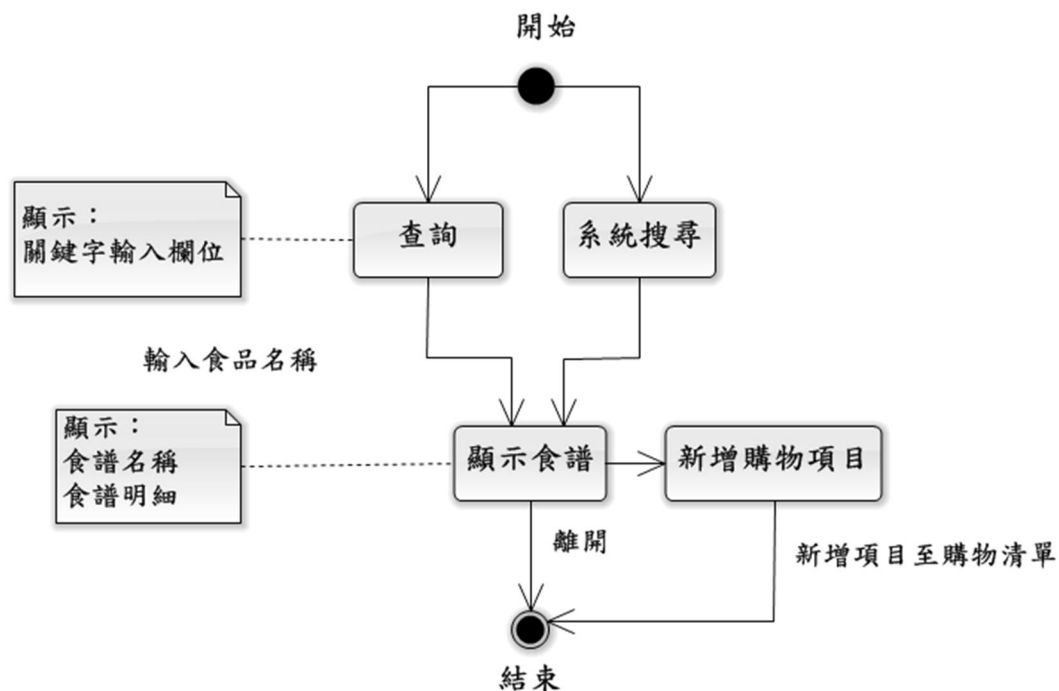


圖 3.11 食譜搜尋流程圖

圖 3.12 是食譜搜尋資料流示意圖。系統運作時資料來源有二，一是使用者輸入的關鍵字，會找出符合關鍵字的相關食譜；二是取自“食品管理系統”的資訊，篩選出裡面有較多項目符合現有食品的食譜，將符合率從高至低依序顯示，目的在於利用現有食品減少過多的食品消費。

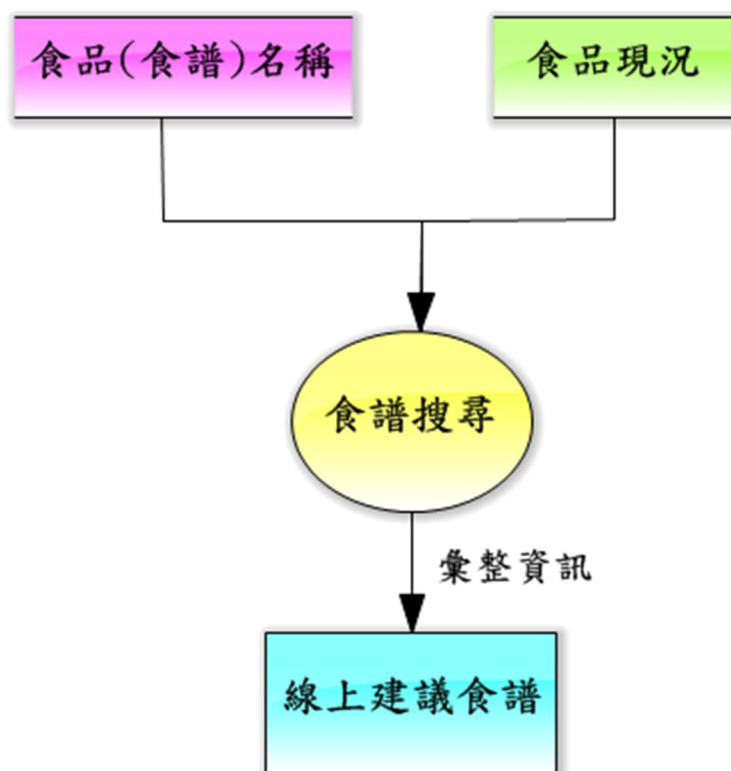


圖 3.12 食譜搜尋資料流

(5) 購物清單

有時候在出門前都想好了要買的東西，可是一到賣場就忘東忘西，甚至還會買同樣的食品，每每出門前都要翻一次冰箱做清單不只花時間又麻煩，所以應用程式提供購物清單服務，便於使用者統整購買物品。

圖 3.13 是購物清單流程圖，使用者可以隨時查看購物清單，若臨時有想加的食品，也可以直接輸入食品名稱進行新增。

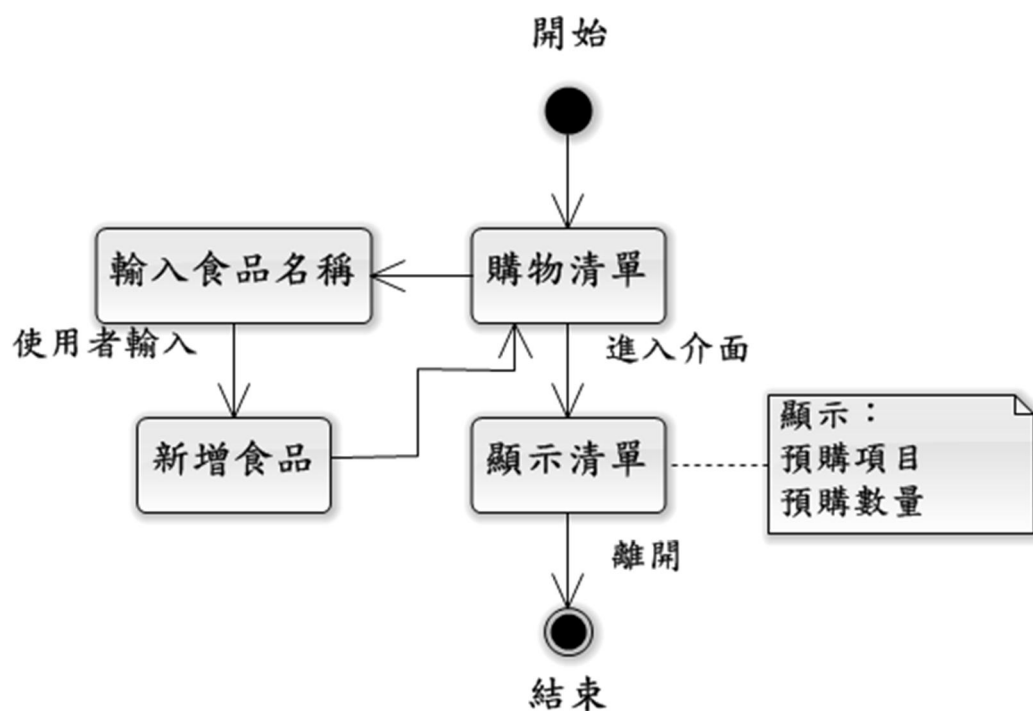


圖 3.13 購物清單流程圖

圖 3.14 是購物清單系統的資料流程圖，資料有三個來源，一是使用者輸入食品名稱後新增的項目，二是食品管理系統的食品現況資訊，與食譜搜尋系統的合適菜單結合，彙整出缺少的食品項目，經由這三個資料來源就可產生一張完整的購物清單。

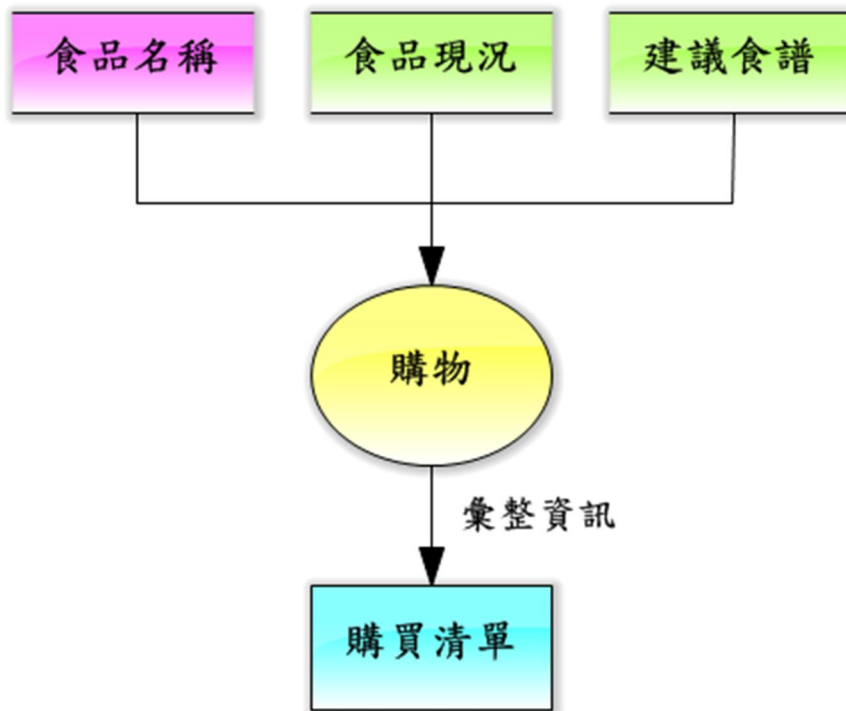


圖 3.14 購物清單資料流

第 4 章 人力配置

在此專題中包含四位組員，於製作專題期間內分工合作，其分工如下表 4.1。

表 4.1 人力分配表

姓名	工作內容	
蔡佩蓉	1. QR code 分析、設計 2. APP 應用程式介面設計	1. 資料蒐集 2. 資料統整 3. 報告書撰寫 4. 系統設計與分析
詹孟陵	1. APP 應用程式功能開發 2. APP 應用程式介面設計	
杜宜家	1. 資料蒐集 2. 資料庫設計、分析 3. 伺服器	
陳凱恩	1. APP 應用程式功能開發 2. APP 應用程式介面設計	

第 5 章 執行速度

圖 5.1 為此專題的進度規劃。以條狀圖表示進度與時間的關係，有益於規畫研究進度，與時間控管，明確表示任務項目、工時。

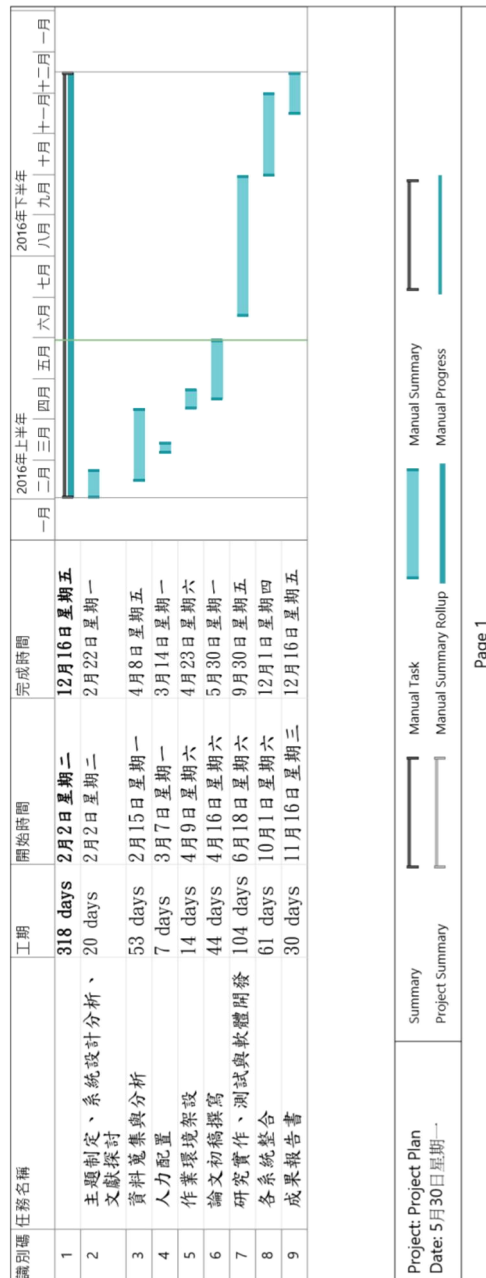


圖 5.1 專題進度甘特圖

第 6 章 現階段成果

我們對各個部分功能做了一些測試，以下為初步成果。

6.1 資料庫

應用程式中會有許多資料，我們將建立資料庫方便做管理和存取。
下面是我們的食譜資料庫。

圖 6.1 為食譜資料庫關聯圖是表示 table 之間的關係。將資料庫正規化，以降低資料的重複性與相依性，減少空間的浪費。

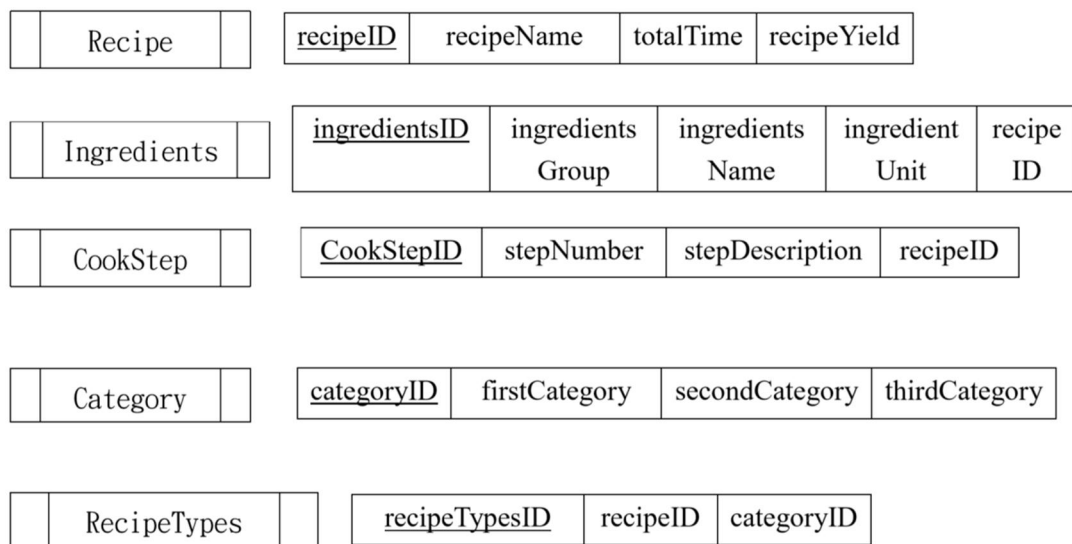


圖 6.1 食譜資料庫關聯圖

圖 6.2 為查詢食譜的 SQL 語法，我們利用選定的食譜 ID 依照資料庫的關聯性把我們想要看到的食譜資訊顯示出來。

```
SELECT
    test.recipe.recipeName,
    test.category.firstCategory,
    test.category.secondCategory,
    test.category.thirdCategory,
    test.ingredients.ingredientsGroup,
    test.ingredients.ingredientsName,
    test.ingredients.ingredientUnit
FROM
    test.category,
    test.ingredients,
    test.recipe,
    test.recipeTypes
WHERE
    test.recipe.recipeID = 20000
    AND test.ingredients.recipeID = 20000
    AND test.recipeTypes.recipeID = 20000
    AND test.category.categoryID = test.recipeTypes.categoryID;
```

圖 6.2 查詢 SQL 語法

圖 6.3 是執行查詢結果。由於各欄數目不盡相同，若要將查詢結果放於同一表格，則會出現資料重複的狀況。

recipeName	firstCategory	secondCategory	thirdCategory	ingredientsGroup	ingredientsName	ingredientUnit
青椒炒豆 干	主要食材	瓜果料理	青椒	食材	青椒	4 顆
青椒炒豆 干	主要食材	瓜果料理	青椒	食材	五香豆干	5 片
青椒炒豆 干	主要食材	瓜果料理	青椒	食材	紅蘿蔔絲	適量
青椒炒豆 干	主要食材	瓜果料理	青椒	食材	蒜頭	2 粒
青椒炒豆 干	主要食材	瓜果料理	青椒	調味料	濕豆豉	2 大匙
青椒炒豆 干	主要食材	瓜果料理	青椒	調味料	醬油	適量
青椒炒豆 干	主要食材	瓜果料理	青椒	調味料	糖	少許

圖 6.3 食譜查詢結果

6.2 應用程式

對於應用程式的介面，我們使用 Bootstrap 來美化介面。APP 功能中我們有利用二維掃描器去取得資訊；因此在專案中先加入 Cordova 的套件之後，再撰寫關於掃描器的程式。

圖 6.4 是本次專題研究的應用程式，目前已將程式的部分框架完成，可將程式下載後安裝到行動裝置上。

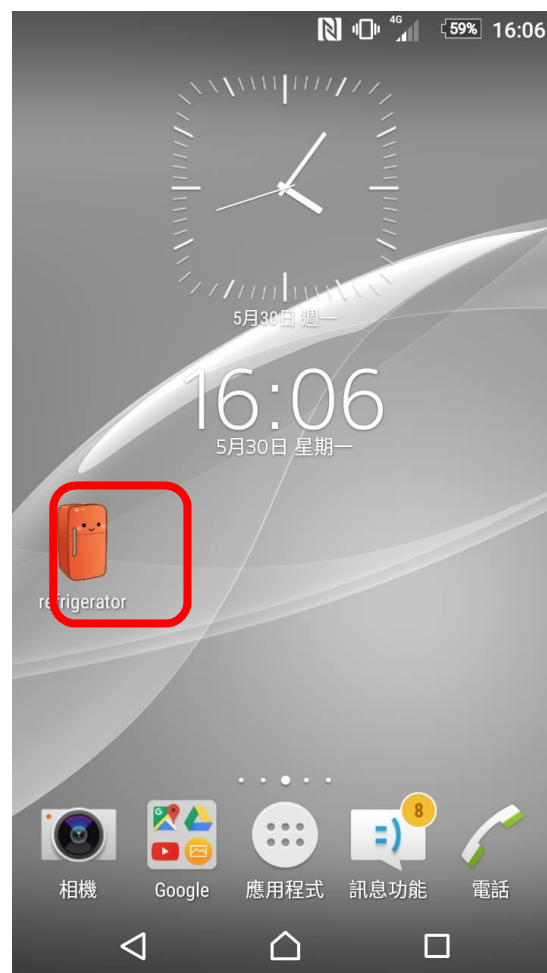


圖 6.4 應用程式圖示

圖 6.5 為點擊應用程式後進入的介面，此應用程式連結 Bootstrap 美化介面，目前僅建置掃描功能，進入應用程式後點即按鈕觸發掃描介面。



圖 6.5 應用程式初步介面

圖 6.6 為掃描功能的介面，目前可以掃描一般發票的二維條碼。



圖 6.6 二維掃描器

第 7 章 預期成果

在行動裝置普及的時代，除了攜帶方便，手機應用程式的功能為了滿足民眾的需求而快速提升、創新多變，但與融入生活有點差距，因此，我們研究此專題的目的是讓應用程式融入民眾的生活。

這次研究的專題以民眾的生活圈為主向外拓展。逐步查詢、過濾資訊再彙整最後結果這些步驟逐漸被程式所取代，人力資源不再花費時間在這部分，研究其他應用程式後發現三個步驟總無法一氣呵成，尤其彙整最後結果的部分，雖然應用程式提供了精確的資訊，但民眾還是得將得到的資訊與自身狀況再做一次彙整，才能得到真正有利於自己的資訊，可是忙碌的人們根本沒有時間把自己的作息一筆一筆輸入到應用程式裡，以此為目標，在民眾不需手動的情況下將生活中的資訊放進應用程式裡，意即在民眾動作完成後，就能在應用程式得到需要的資訊。

本次專題研究主題是生活中的飲食，現代的生活步調快速，現代人若要追求健康美味的佳餚不是用錢就是用時間換，不管用哪一種換都不容易，花大錢只為了一餐、學一身廚意只為了一餐，從購買食品到下廚或許不如我們想的困難，只是沒有豐富的料理經驗而無從下手罷了。此應用程式不只提供食品食譜的相關訊息，還能結合民眾的生活資訊，食安資訊、家中庫存的食物、缺少的食物、過期的食物、新購買的食物等，進入應用程式彙整後給與合適的結果，民眾不必再翻找冰箱檢查食物的狀態；不一定只有家常菜，利用庫存的食物也能做出意想不到的料理等。

期望此應用程式可以符合更多年齡層的民眾的生活所需，人人都能當上家庭主廚，輕鬆管理食物，將耗時的過程變得精簡，讓使用者從採購、食安疑慮到成功出菜不會再如臨大敵。

參考文獻

- [1] 科健商品驗證有限公司. (2010). 2016 年 4 月 25 日 擷取自 <http://monkey2010.pixnet.net/blog/post/50318234-%3C%E6%89%>
- [2] python. (無 日 期). 2016 年 4 月 擷 取 自 <http://makeawebsitehub.com/wp-content/uploads/2016/02/python-logo-master.png>
- [3] FoundationPython SoftwareThe. (2016 年 1 月 21 日). 12.6. sqlite3 — DB-API 2.0 interface for SQLite databases. 2016 年 5 月 擷取自 <https://docs.python.org/3.4/library/sqlite3.html>
- [4] C#. (無 日 期). 2016 年 4 月 擷 取 自 <http://romacode.com/media/1004/csharp-logo.png>
- [5] 使用 HTML5 和 PhoneGap* 的行動應用程式在 Intel® 架構平台之發展. (無 日 期). 2016 年 4 月 擷 取 自 <https://software.intel.com/sites/landingpage/tw/the-development-of-mobile-applications-using-html5-and-phonegap-on-intel-architecture-based-platforms.php>
- [6] MySQL. (無 日 期). 2016 年 4 月 擷 取 自 <https://upload.wikimedia.org/wikipedia/zh/thumb/6/62/MySQL.svg/1280px-MySQL.svg.png>
- [7] node.js. (無 日 期). 2016 年 4 月 擷 取 自 <https://brainhub.eu/img/technology/nodejs-brainhub-software-development.png>
- [8] FoundationNode.js. (2016). Node.js v6.2.0 Documentation. 2016 年 4 月 擷取自 <https://nodejs.org/api/synopsis.html>

[9] 林睦芸，賴宇庭，簡好健. (2014). 門禁管理系統.