

---

# 中華大學

## 資訊工程學系

---

### 100 學年度專題製作 期末報告

## 餐廳點餐系統

Ruby on Rails 環境開發

---

指導教授：歐陽雯 老師

學生：成瑞文(B09702079)

彭韓毅(B09702070)

---

中華民國 101 年 8 月 27 日

# 目錄

1. 摘要.....	2
2. 研究背景與目的.....	3
2.1. 背景.....	3
2.2. 目的.....	4
3. 開發環境.....	5
4. 專題分工.....	6
5. 專題進度表.....	7
6. ER-MODEL.....	8
7. 系統設計與架構.....	9
7.1 使用者架構.....	9
7.2 主要功能.....	9
7.3 管理者架構.....	9
7.4 主要功能.....	10
8. 專題內容.....	11
8.1 首頁.....	11
8.2 點餐.....	12
8.3 購物車.....	13
8.4 管理者頁面.....	14
8.5 MENU 管理.....	15
8.6 訂單顯示.....	16
8.7 訂單刪除.....	17
9. 未來展望.....	18
10. 結語.....	18
銘謝.....	19
資料參考.....	19

# 1. 摘要

Ruby on Rails 是用 Ruby 寫的網頁應用程式開發框架(web application framework)，一般簡稱” Rails” 或是” RoR” 。它的作者是 David Heinemeier Hansson (簡稱 DHH) ，一個來自丹麥的年輕程式師。RoR 是他為了 Basecamp 專案而開發的架構，之後在 2005 年釋出的 open source project，目標是要加快以資料庫驅動的網站的開發速度與簡化過程。要達成這個目標，有兩個重大原則：一個是絕不做重覆的事，(DRY, Don’ t Repeat Yourself)，程式設計師要寫的程式碼愈少愈好，因為程式碼少的話，出錯的機率也會降低，最後維護起來也容易；另一個原則是用慣例(通則)取代設定(Convention over Configuration)。

Ruby on Rails 的觀念在於利用慣例優於組態(convention over configuration)，軟體精簡(less software)與越快越有生產力(programmer happiness ultimately leads to better productivity)。Ruby on Rails 是完全的 Model-View-Control 架構，資料庫、程式邏輯與呈現完全清楚的分離。在檔案名稱與目錄結構方面，也都定義的很清楚。資料庫這部份(Active Record)是 Ruby on Rails 最強大的功能，只要設定好資料庫連結，之後建立、調整 Schema 或是其他存取都是直接由 Ruby on Rails 去控制，不用由我們親手去動資料庫，資料庫裡的欄位也能和 Ruby on Rails 裡的變數直接對應，不用再像用 php 的時候，和 SQL 語法一大堆冒號、引號奮戰。另一個可以加快速度的特點在於自動產生使用者界面的程式碼(scaffold 功能)，這代表我們只要把定義好資料庫和要作的動作，Ruby on Rails 就會為我們產生使用者界面，我們只要專注在真正要解決的問題上，而不用花太多時間在建立使用者界面上。是一種結合 Ruby 語言與 Rails 平台的一種網頁程式語言，Ruby 語言以自然、簡潔、快速著稱，全面支援物件導向程式設計，而 Rails 則是 Ruby 廣泛應用方式之一，在 Rails 平台上設計出一套獨特的 MVC 開發架構，採取模型 (Model)、外觀 (View)、控制器 (Controller) 分離的開發方式，不但減少了開發中的問題，更簡化了許多繁複的動作。

## 2. 研究背景與目的

### 2.1. 背景

Ruby 的作者是個名叫松本行弘 (Yukihiro Matsumoto, matz) 的日本人，他在用了多種不符他需求的程式語言後，決定開發一個比 Perl 功能更強大，比 Python 更物件導向的程式設計語言。Ruby 這個用 C 語言寫成的純物件導向 Script Language 因此誕生了。松本的設計理念是給程式設計師最少的驚奇 (principle of least surprise)，他站在程式設計者的角度設想，希望程式設計者使用 Ruby 能得心應手，用得快樂，迅速開發。Ruby 第一個 alpha 測試版是在 1994 年 12 月釋出，雖然在 1995 年在日本正式釋出時，日本人並不陌生，但英文版的 Ruby 則是到 1998 年年底才出現。

於 2004 年 7 月，Rails 的創始人大衛·海納梅爾·韓森從 [37signals](#) 公司的項目管理工具 Basecamp 裡面分離出 Ruby on Rails，並且以開源方式發佈。

Rails 在發佈以後的短短的時間內就迅速獲得很多開發人員歡迎，大衛認為這歸功於 Rails 設計為 opinionated software。Rails 當中有很多規矩從一開始就按照 David 的意見制定好了，所以在 Rails 上開發應用程式時，開發人員可以專注於應用程式自身的設計，省卻那些花在瞭解及配置基礎框架上面的時間。這也正是 Rails 很重要的精神「約定優於配置」，開發人員遵照 Rails 本身的慣例便可以省卻配置組態檔的時間；此外，Rails 雖然強調慣例的重要及便利性，但針對不同的需求，Rails 也提供修改的空間讓開發人員可以進行自訂的組態。

## 2.2. 目的

我們看到國外很流行使用 Ruby on Rails 的語言來撰寫購物網站，但國內卻很少人使用這種語言，因此想藉由這次的機會來好好的學習 Ruby on Rails，所以我們研究了 Ruby on Rails 開發環境，並試著撰寫一個點餐的系統。

藉由 Ruby on Rails 的開發環境，設計了一個點餐系統，在網站裡可以快速的知道有哪些餐點，並進行訂購，這樣就可以在事前進行點餐的動作，就不用到店家在進行點餐，而減少點餐時間跟等待時間，可以馬上使用餐點或是帶走餐點。

### 3. 開發環境

**Ubuntu 12.04LTS**：是一個以桌面應用為主的 GNU/Linux 作業系統，Ubuntu 12.04LTS 為最新的 Ubuntu，桌面版和伺服器版均可獲得為期五年的技術支援。Ubuntu 的目標在於為一般用戶提供一個最新的、同時又相當穩定的主要由自由軟體構建而成的作業系統。Ubuntu 具有龐大的社群力量，用戶可以方便地從社群獲得幫助。

**Ruby v 1.8.7**：Ruby，一種為簡單快捷物件導向編程而創的指令碼語言，在 20 世紀 90 年代由日本人松本行弘開發，遵守 GPL 協議和 Ruby License。它的靈感與特性來自於 Perl、Smalltalk、Eiffel、Ada 以及 Lisp 語言。由 Ruby 語言本身還發展出了 JRuby（Java 平台）、IronRuby（.NET 平台）等其他平台的 Ruby 語言替代品。

**Rails 3.0.11**：是一個使用 Ruby 語言寫的開源 Web 應用框架，它是嚴格按照 MVC 結構開發的。它努力使自身保持簡單，來使實際的應用開發時的代碼更少，使用最少的配置。Rails 的設計原則包括「不做重複的事」（Don't Repeat Yourself）和「慣例優於設定」（Convention Over Configuration）

## 4. 專題分工

本組成員，共有二人，分工如下

表 1：專題分工表

	程式撰寫	美工設計	蒐集資料	文件撰寫
成瑞文		★	★	★
彭韓毅	★	★	★	

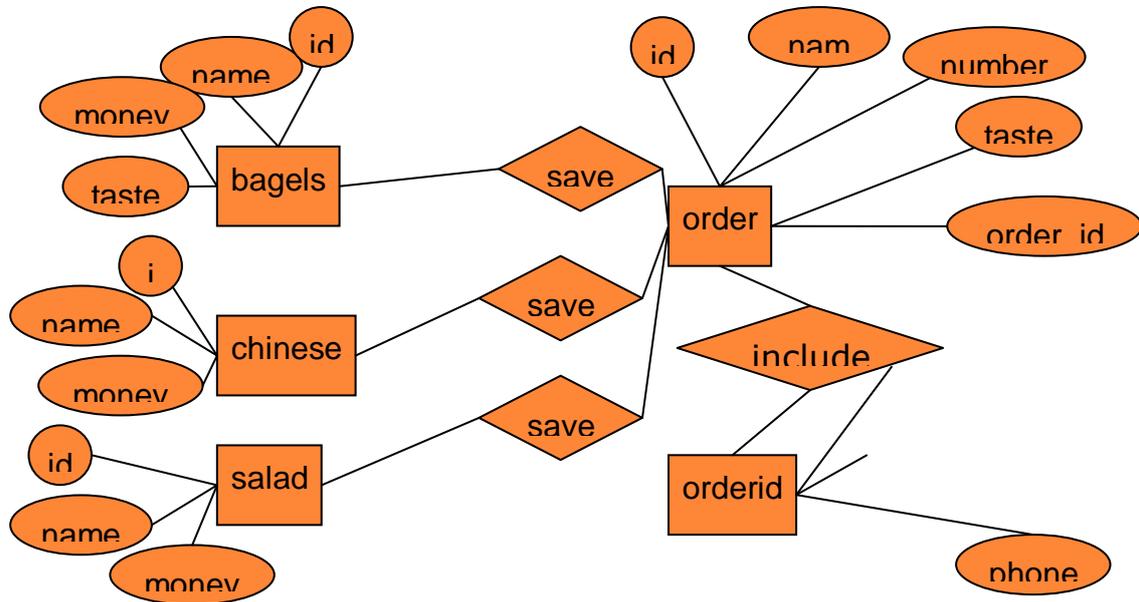
## 5. 專題進度表

本專題由七月開始，隔年的六月結束，進度如表 2。

表 2：進度表

Mouth work	Sep	Oct	Nov	Dec	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug
資料收集	★	★	★									
學習相關 軟體		★	★	★	★	★						
程式開發					★	★	★	★				
實作與整 合							★	★	★	★		
測試與偵 錯									★	★	★	
報告與發 表成 果											★	★

# 6. ER-Model



## 7. 系統設計與架構

### 7.1 使用者架構

在設計一套系統時，要先預想可能有哪些功能，提供給使用者，所以我們先設計了一張流程圖(圖 1)。

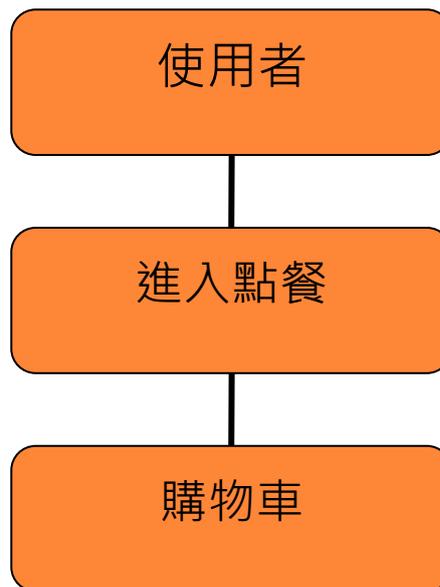


圖 1

### 7.2 主要功能

#### 1. 進入點餐：

使用者進入介面時，會有一個選項是點餐，進入以後會有所有的餐點可以進行勾選。

#### 2. 購物車：

選好餐點以後進行下一步會來到購物車，會看到使用者上一頁所勾選的餐點，留下電話就可以完成訂餐。

### 7.3 管理者架構

我們在管理者這方也需要一個介面供管理者使用，所以我們也設計一張流程圖(圖 2)。

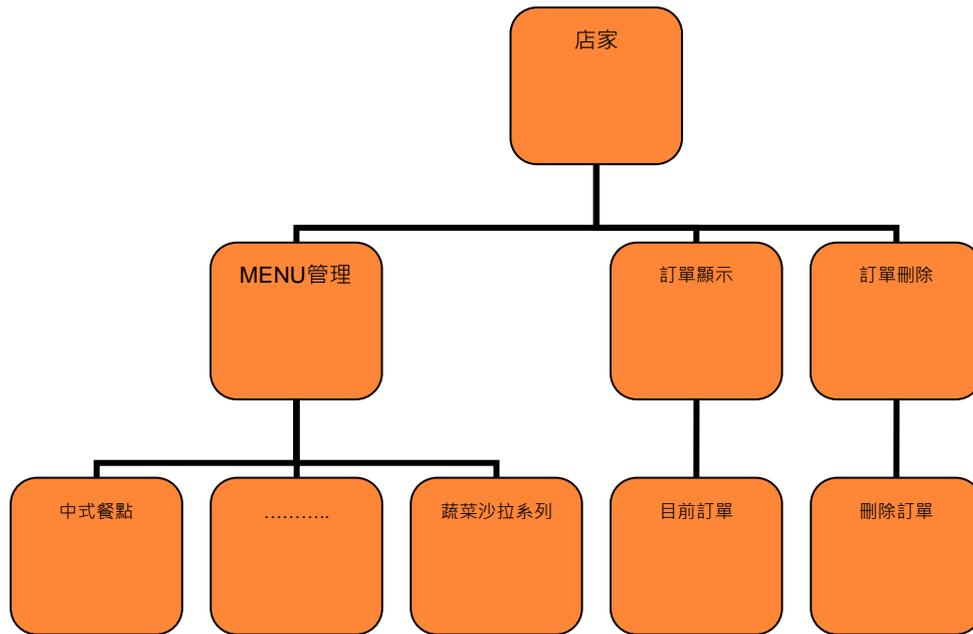


圖 2

## 7.4 主要功能

### 1.MENU 管理：

進入 MENU 管理之後，會有各個系列餐點的選項，可以選擇管理哪種系列餐點。

### 2.訂單顯示：

可以查詢使用者那方的訂單，並隨時更新。

### 3.訂單刪除：

當使用者下錯單時，可以立即的幫她做刪除。

## 8. 專題內容

### 8.1 首頁

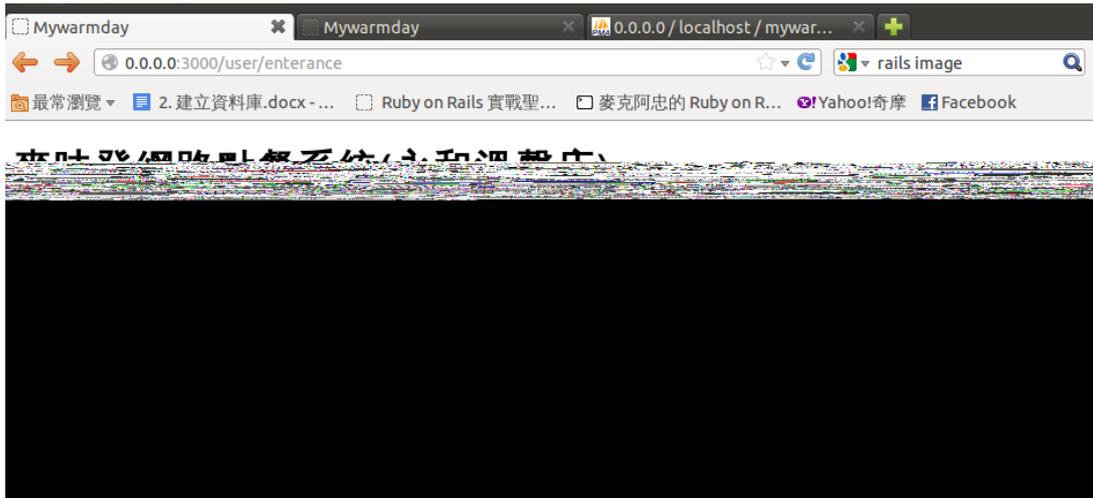


圖 3：首頁

首頁的配置(如圖 3)，第一行是店家的名稱，第二行是點餐，第三行是地址，再來是電話。

## 8.2 點餐

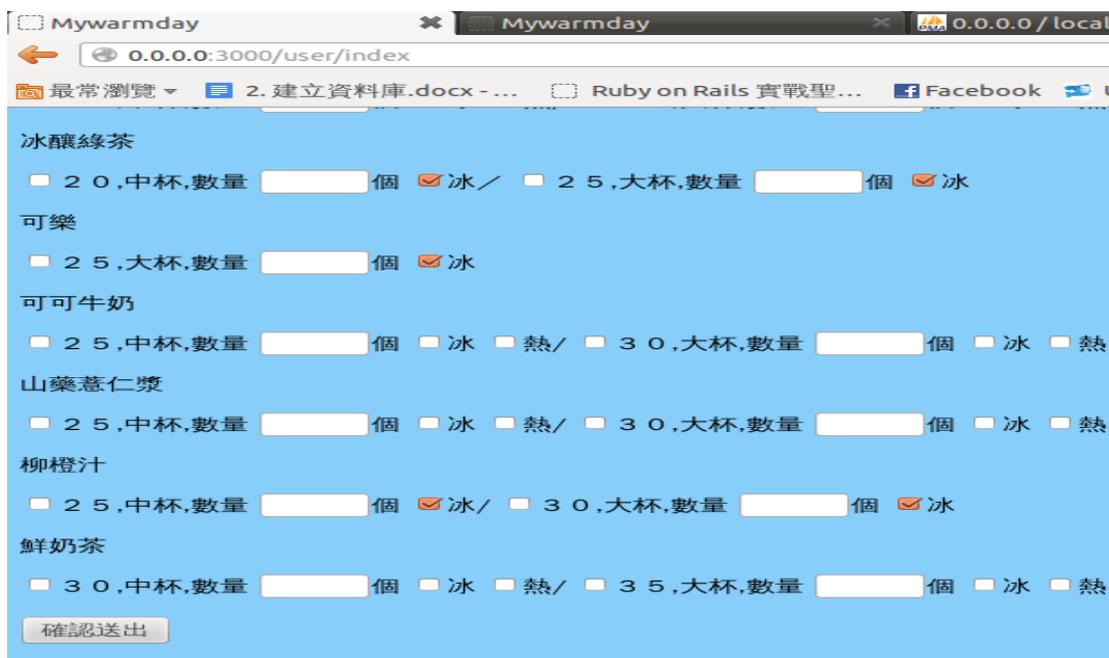
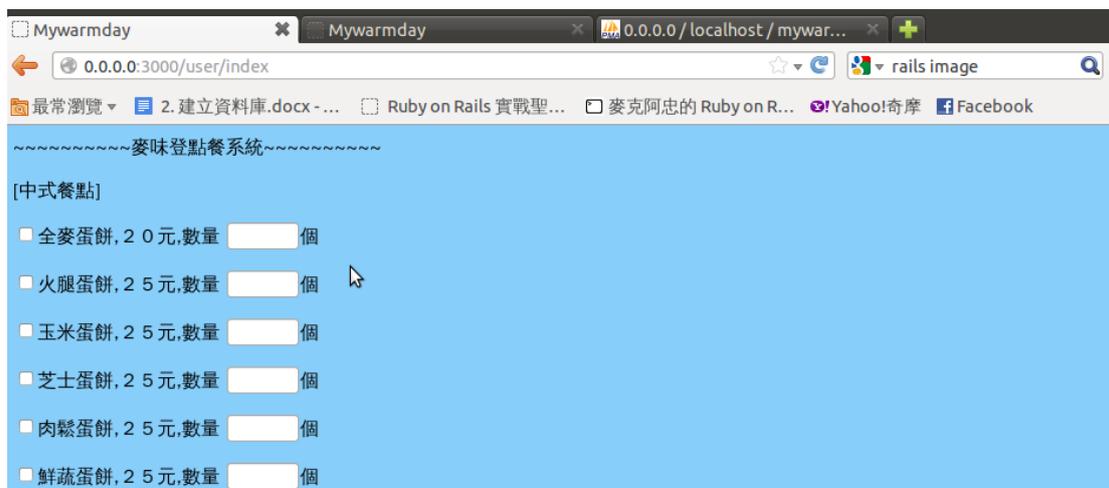


圖 4&圖 5：點餐

這個畫面(如圖 4)，我們在餐點前面設計了一個框框可以勾選，勾選完以後可以在後面填的數量，最後有確認送出的選項，當選好餐點時需點選這個選項才可送出。

## 8.3 購物車



圖 6：購物車

這些是使用者選餐點送出以後會跳到這個畫面確認自己點的餐點有無錯誤，有錯誤的可以點選回上頁的選項，做重新選取的動作，如果沒有可以確認價錢並填寫電話送出，就可以完成訂餐的動作了。

## 8.4 管理者頁面



圖 7：管理者首頁

會有三個選項，一個是餐點的管理，一個是查詢訂單，最後一個是訂單修改。

## 8.5 MENU 管理

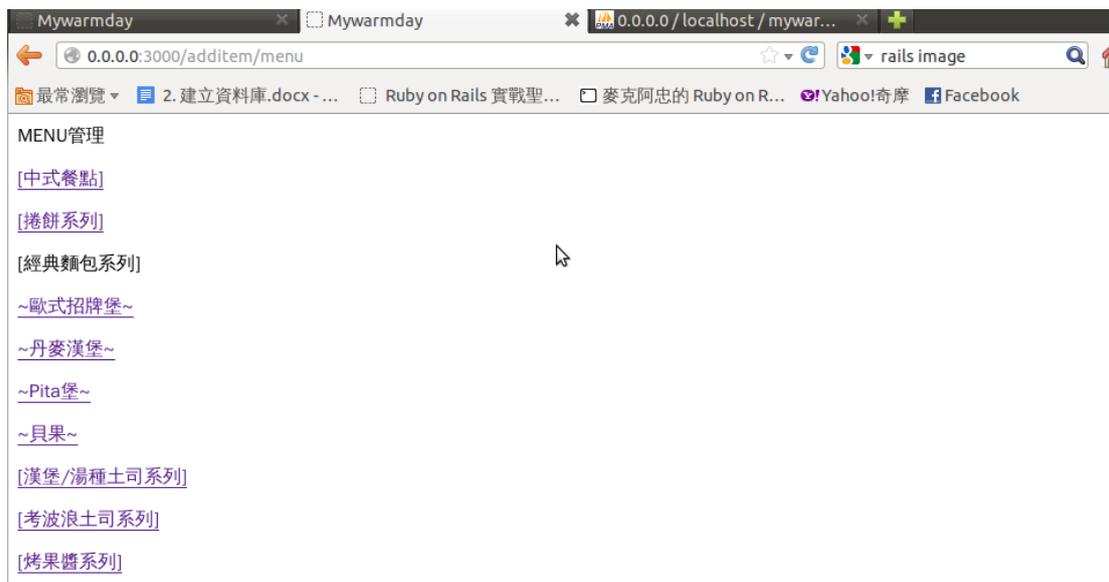


圖 8：MENU 管理

(如圖 8)，會有各個系列餐點，可以做選取，例如點選中式餐點(如圖 9)。

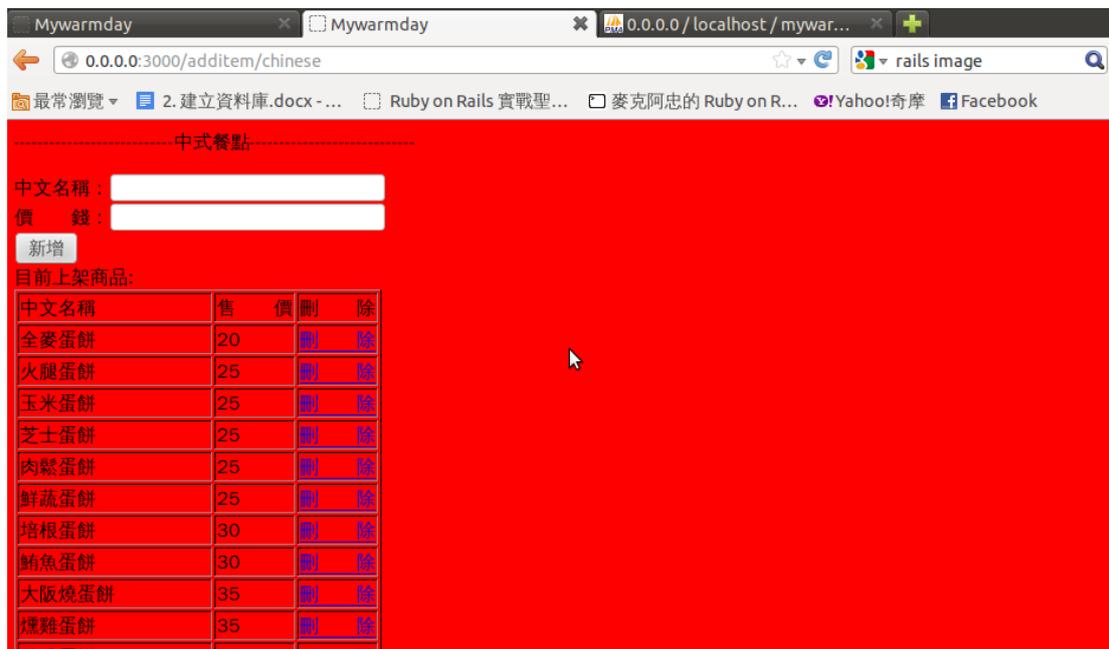


圖 9：中式餐點

就可以進行中式餐點的管理，可以做新增跟刪除。

## 8.6 訂單顯示



目前訂單

編號	總價	電話
10	150	0912345678

訂單項目

訂單編號	名稱	售價	數量	種類
10	全麥蛋餅	20	2	
10	火腿蛋餅	25	2	
10	中杯豆漿	15	2	冰
10	中杯紅茶	15	2	冰

圖 10：訂單顯示

可以查詢目前訂單的狀況，並一直刷新檢查是否有新訂單。

## 8.7 訂單刪除



圖 11



圖 12

如圖(圖 11&圖 12)可以將錯誤的訂單整筆刪掉

## 9. 未來展望

藉由這次專題學習了 Ruby on Rails 這全新的語言，學一套新的語言很花時間，安裝環境也花了不少時間，因此我們只寫出了比較簡單的介面，希望之後能縮短這方面的時間，把時間都花在最重要的排版上，並把這套語言變成自己的能力，在未來職場上也能加以使用。

## 10. 結語

這次的專題製作讓我們對 Ruby on Rails 的軟體開發有更進一步的了解。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。建置整個網站的過程中，雖然製作過程有些辛苦，但是我們學會了，怎麼發揮一個團隊的力量，合作的精神，分工合作的方式，克服了重重的問題，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。

# 銘謝

在此專題製作之中，我們遭遇許多問題，尋找很多解決方法，上網查資料等等，不斷的嘗試各種方式來解決我們所遭遇到了問題，所幸我們遭遇到的問題一一的解決，專題雖然有所耽誤，但是我們還是大致上完成了。其中也包括詢問我們的專題指導老師-歐陽雯 老師，經過與老師的討論，也會給予我們很大的受益。此外非常感謝一有問題就被我們纏著不放的學長，經由他的指導讓我們問題的能順利的解決，也讓我們學習很多解決問題的方法。

# 資料參考

- [1] Ruby on Rails 實戰聖經網站 <http://ihower.tw/rails3/index.html>
- [2] 麥克阿忠的Ruby on Rails 特快車  
<http://www.openfoundry.org/tw/tech-column/8440-ruby-on-rails->
- [3] Rails Fun!! Ruby & Rails 正體中文論壇 , Everything so Fun!! <http://railsfun.tw/>