長庚大學資訊管理系畢業專題報告

指導老師同意書

資訊管理系大學部

B0644212	吳韋霆
B0644217	陳俞臻
B0644225	謝昀蓉
B0644235	余　晨
B0644246	黃如郁

君等所提之畢業專題報告

lunch, dinner or Mate？一起趣吃飯

業經本人審閱完畢，同意提付系辦存查。

此致

系主任

指導老師_____(簽名)

_____年____月____日

中文摘要

近期，交友軟體市場蓬勃發展，然而多數人使用交友軟體後，不見得會出門與網友見面，因而減少了人與人真實互動的機會。因此，為改善使用交友軟體，卻缺乏實際互動之現況，本研究希望藉由每日之例行公事——享用一頓飯的時間，結合拓展交際圈，讓兩件事同時進行。

有別於常見的交友軟體，使用 FoodMate 交友，不再需要在網路世界裡，建立基礎的交流與認識，直接出門見面，若見面後不喜歡對方，大不了就安安靜靜吃一頓飯，日後待有緣再相見。

本研究開發之 APP 有三大功能，分別是餐點抽卡、餐廳搜尋以及飯友邀請，所有的功能均以個人化推薦的內容為優先。個人化推薦的內容，將藉由 content-based 推薦演算法，根據食物類型、餐廳的預算、評分、環境、天氣狀況、使用者的年齡、性別等資料，進行餘弦相似度計算，以建立一款基於一日三餐之飲食偏好的交友軟體。

目錄

圖表目錄

第一章　研究動機與背景

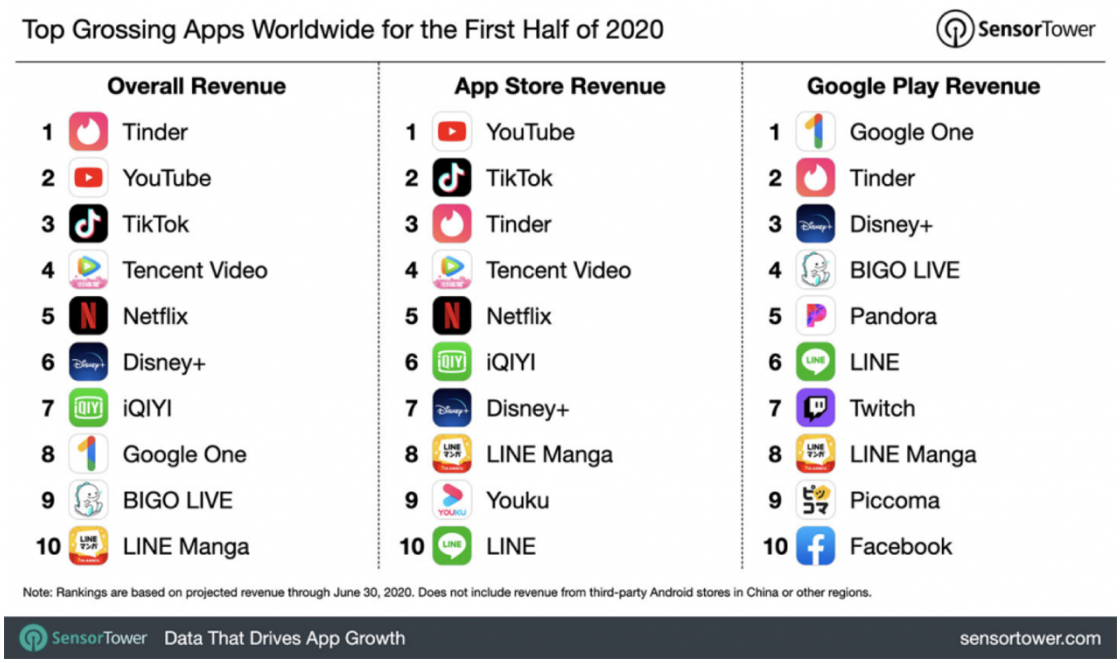

圖 1、Sensor Tower 調查

受疫情之影響，人與人之間的交際距離被迫退到 1.5 公尺之外，網路交友的市場因而快速擴大，根據數據調查公司 Sensor Tower 的一份調查（圖 1），交友軟體 Tinder，成為 2020 上半年全球行動裝置的獲利最高者。

有鑑於此，本研究開發了一款藉由飲食來交友的 APP，相較於普遍的交友軟體，本研究設計之 APP，是運用搜集各個用戶之飲食偏好，以相近的飲食習慣、交友之偏好等依據，進行飯友推薦，而有了相近的飲食習慣，更能成為用戶創造共同話題的契機。

使用本 APP 之用戶，只需運用一日三餐的時間，就能多認識一位朋友，不必為了見上一面，而特地排開原有的工作或行程。由於不論情侶、朋友，甚至家人，都有可能為了每日三餐的飲食選擇，產生煩惱，而每個人不同的飲食習慣，甚至可能成為平時常見的小紛爭，此些情境就成為了本研究想要開發 FoodMate 的動機。

第二章　系統使用對象與環境

2.1　系統使用對象

主要目標為 18 歲以上的群眾，並具備以下特點：

1. 對於日常生活中的三餐，有需要花時間思考或選擇上的困擾。

2. 喜歡交朋友，即使只有短短的吃飯時間，也想把握機會認識新朋友。

3. 想要找夥伴一起到餐廳用餐，填滿餐桌空位。

2.2　系統使用環境

1. 使用 iOS 作業系統的智慧型手機

2. 有無線網路的環境供軟體資料即時更新及程式 API 呼叫。

第三章　系統開發工具與環境

我們的開發環境分為後端資料處理以及前端系統與介面兩個部分。前端系統與介面這部分，我們將系統建設在 Xcode 平台上，開發語言則是以 Swift 撰寫，也利用 Xcode 自身元件來建立使用者介面。後端資料部分，我們依據功能，分別將資料儲存在 PostgreSQL 和 Firebase 中，並使用 Python 和 AWS 服務做資料處理。

3.1　前端系統與介面

（一） Xcode

【用途】開發系統的平台

【選擇 iOS 系統的原因】

1. Xcode 有強大的整合開發環境，內建 iPhone 模擬器。
2. 以作業系統來看，iOS 系統比安卓系統較為穩定、安全。

（二） Swift

【用途】撰寫系統的程式語言

【選擇原因】

1. iOS 官方支援的開發語言。
2. 比起同為開發 iOS 的 Objective-C，使用較為方便、安全。

3.2 後端資料處理

（一） Python

【用途】

1. 建立餐點、餐廳、飯友的個人化推薦演算法。
2. 蒐集 Google Maps 餐廳詳細資訊。

【選擇原因】

1. 支援多種作業系統，可跨平台編輯程式碼。
2. 提供豐富完善的函式庫，能實現複雜的資料處理流程。

（二） PostgreSQL

【用途】

1. 存放使用者的個人資料、APP 使用紀錄。
2. 存放餐廳的詳細資料。

【選擇原因】

1. 開源的關聯式資料庫管理系統，資料表之間能透過索引鍵建立關聯，後續使用 SQL 即能完整存取所需的資料。
2. 支援各種程式語言，能夠使用 Python 操作資料庫。

（三） Amazon RDS

【用途】架設 PostgreSQL 資料庫

【選擇原因】

Amazon RDS 提供在雲端建置及管理關聯式資料庫的服務，支援 PostgreSQL，因此能用此服務架設資料庫，而使用這項服務好處是定期會更新資料庫至最新版本，也能進行資料的備份，有別於一般建立在主機上的做法，操作更為便利、安全。

（四） AWS Lambda

【用途】執行個人化推薦演算法的程式碼

【選擇原因】

將利用 Python 撰寫的程式碼上傳至 AWS Lambda，就能與其他 AWS 資源進行整合，觸發事件來自動執行程式。

(五) AWS API Gateway

【用途】建立 RESTful API 讓 APP 取得個人化推薦資料

【選擇原因】

AWS API Gateway 可以在雲端建立無伺服器的 RESTful API，產生可用於 iOS 用戶端的 SDK，存取後端演算法執行後得到的資料。

（六） Firebase

【用途】

1. 進行帳戶管理。

2. 存放使用者頭貼以及餐點圖片。

3. 在邀請飯友，進行即時的訊息傳輸。

【選擇原因】

1. Xcode 的開發環境，有套件可以直接連接 Firebase。

2. 能夠解決 PostgreSQL 資料庫無法存放圖片的問題。

3. 透過 Firestore 服務，能夠達成即時傳送邀請的功能。

第四章　系統設計與介紹

4.1　系統特色

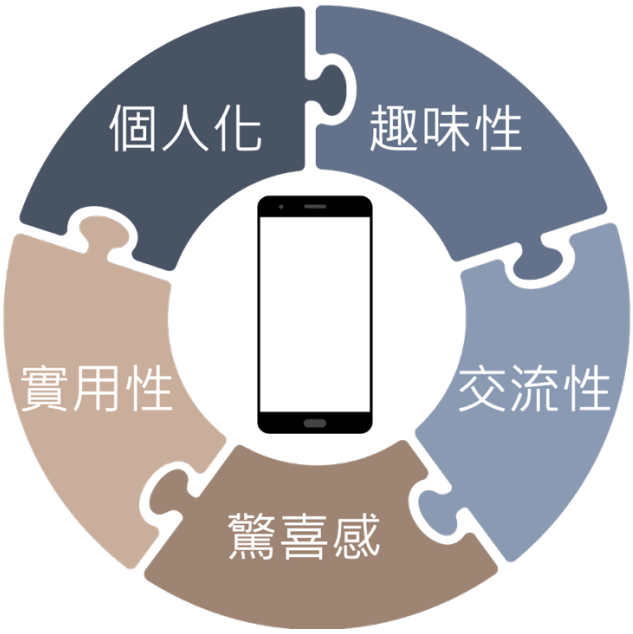

圖 2、系統五大特色

1. **個人化—**

第一次使用，除了註冊帳號，也會先讓使用者篩選不吃的食物，作為日後推薦餐點時的依據，讓隨機模式更貼近使用者的飲食習慣。

2. **趣味性—**

選擇餐點利用滑動卡片的方式，與使用者增進互動性。

3. **交流性—**

受科技影響，現代的人際關係裡常常缺乏實際互動，交友不能只侷限在手機上，所以此 APP 的設計，沒有聊天的功能，不讓使用者侷限在手機上的交流，而是真實的互動。

4. **實用性—**

民以食為天，一日三餐。吃東西是每個人每天都要做的事，無論如何，每天至少會有 3 次、每週 21 次的機會可以使用此 APP。

5. **驚喜感—**

系統會隨著使用者所在城市的天氣，不定時地在使用者打開此 APP 時，推播通知詢問使用者要不要 依據天氣狀況推薦餐點，例如：當氣溫低於攝氏 20 度，可能會變成推薦火鍋、薑母鴨、羊肉爐等等。

4.2 系統介面與流程

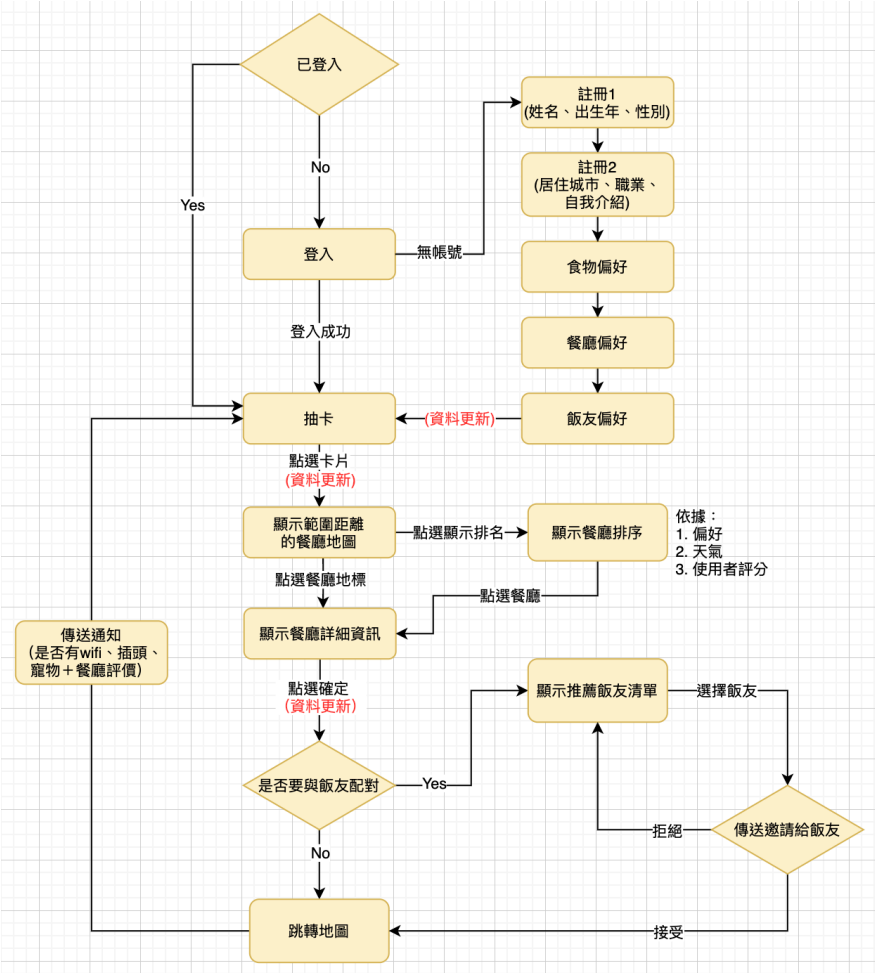

圖 3、系統流程圖

（圖 3）為我們 APP 的系統流程圖，我們主功能將分成五個部分逐一介紹，分別為偏好設定、餐點抽卡、餐廳地圖、飯友配對邀請以及飯後的餐廳回饋通知。

4.2.1 偏好設定

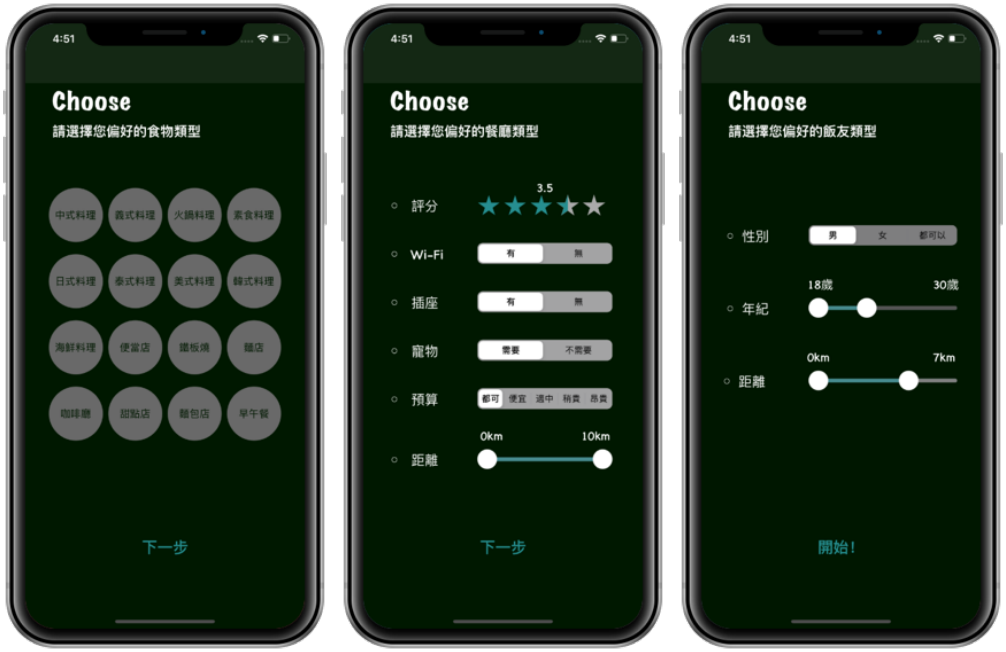

圖 4、偏好設定介面

在註冊過程中，需要設定的三種偏好，分別是食物偏好、餐廳偏好以及飯友偏好（圖 4），為初次的推薦提供基礎。

首先是食物類型偏好，總共分 16 大類，可以選擇自己喜歡的食物類型；餐廳偏好，設有評分、wifi、插座、寵物、預算、距離，依照每個使用者的用餐偏好去做設定；飯友偏好，則是可以選擇想要配對飯友的條件，像是年齡、距離、性別等，這些偏好日後都能進行更改，也會進行推薦演算法的資料更新。

4.2.2 餐點抽卡

圖 5、餐點抽卡介面

完成登入動作或註冊完畢後，就會進到餐點推薦畫面（圖 5），利用抽卡的方式來作為互動，卡片可以往上、下、左、右四個方向滑動，四個方向代表著對於餐點的喜好程度，上下左右分別為：最愛、黑名單、不喜歡以及喜歡，若是忘記也可以點選右下角資訊鈕查看抽卡規則，所有滑動的紀錄皆會寫入資料庫，作為日後推薦餐點的依據。

4.2.3 餐廳地圖

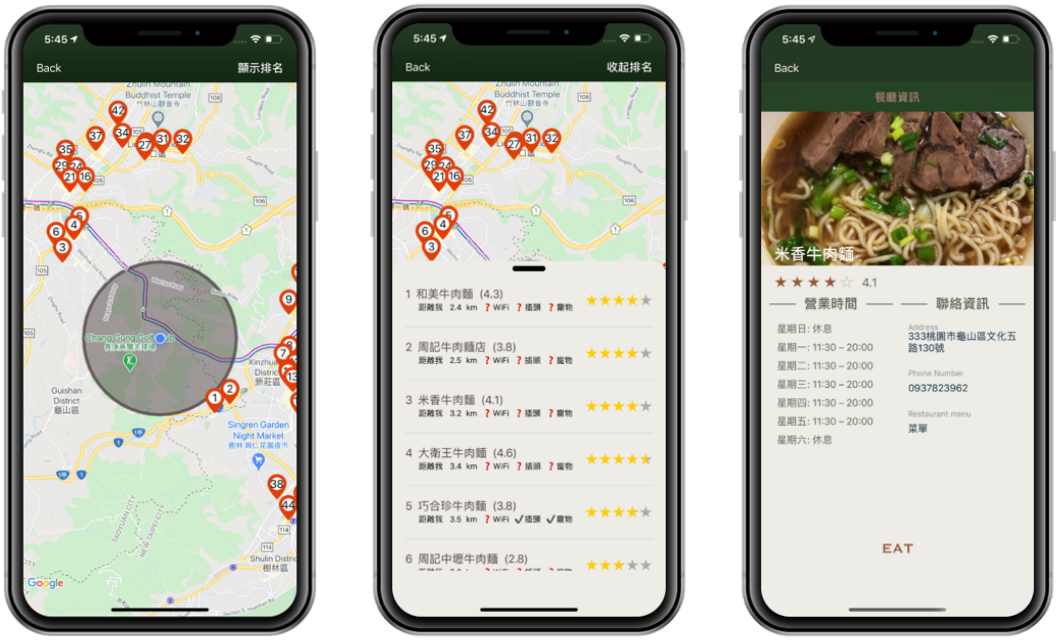

圖 6、餐廳地圖與餐廳詳細資訊介面

點選餐點後則會顯示餐廳地圖（圖 6－左），會依據使用者設定的偏好、天氣來提供排名，排名清單會標示店名、環境設備的有無、評分和距離，並且依序的標示在地圖上（圖 6－中），接下來點選圖標或是餐廳即可查看餐廳的詳細資訊（圖 6－右），詳細資訊包含了營業時間、聯絡資訊和菜單，可供使用者參考。

4.2.4　飯友配對邀請

圖 7、飯友配對邀請介面

選定餐廳後，會讓使用者自行選擇是否想要與飯友共進餐點（圖 7 – 左），若選擇找飯友一起吃，系統會依據使用者所設定的飯友偏好，推薦人選，點選即可看到飯友的資訊並發送邀請（圖 7 – 中），發送邀請需輸入用餐的時間以及付款方式，等待對方回覆，而接收者會跳出訊息框，標示邀請者、地點、時間、以及地點（圖 7 – 右），若選擇拒絕，傳送邀請者則會跳回飯友資訊的頁面，若是接受，雙方都會跳轉至地圖進行導航。另外，若是不想接收到別人的邀請通知，可至設定將通知關閉。

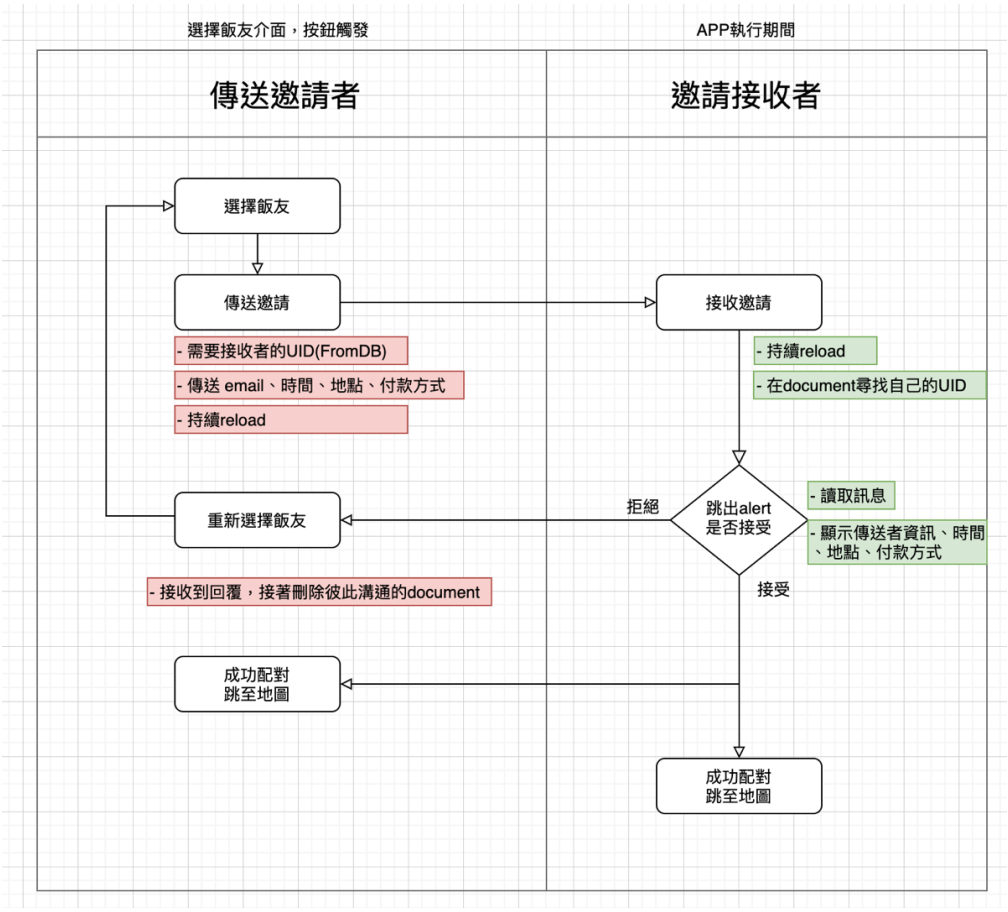

圖 8、傳送/接收邀請流程圖

（圖 8）為傳送/接收邀請的流程圖，以傳送邀請者與邀請接受者來做區分。我們使用 Firebase 的 FireStore 服務，傳接送都會透過此服務存放資料來，達到即時傳輸訊息的功能。

接下來概述身為傳送方與接收方的流程：

1. 傳送方傳送的資訊會寫進 Firestore 裡，持續搜尋接收方是否有回應，等待接收方拒絕或接受。

2. 接受方則是持續搜尋 Firestore 是否有傳送方傳送訊息給自己，收到訊息時則會跳出訊息框，作出回應後，再將回應訊息寫入 Firestore。

4.2.5 餐廳回饋通知

圖 9、餐廳回饋通知介面

在確定使用者會前往該餐廳吃飯後，在一小時內系統會傳送一則通知（圖 9），詢問該餐廳是否有 wifi、插頭、是否能攜帶寵物，以及餐廳評價，這些資料都會回傳到我們的資料庫，有了使用者們的評價資料，往後我們所推薦的餐廳資訊將更為準確。

第五章　資料存取與處理流程

5.1　使用者個人資料

在這個 APP 中會記錄每位使用者的基本資料、目前位置、卡片使用紀錄、最愛餐點與黑名單、餐廳歷史紀錄、朋友列表，以及使用者的食物、餐廳、飯友偏好。

其中，基本資料包含使用者的帳號、密碼、名稱、性別、生日、E-mail、職業、所在城市、自我介紹、大頭照；卡片使用紀錄包含餐點名稱、卡片停留時間、日期、時間、當下天氣、溫度，以及是否有點選、喜歡、不喜歡、最愛、黑名單。

5.2　餐點種類

在這個 APP 上的餐點，主要以台灣常見的食物為主，總共提供 338 種餐點做選擇，這些餐點分為 16 個類別，包括：

- 中式料理
- 日式料理
- 韓式料理
- 美式料理
- 義式料理
- 泰式料理
- 火鍋料理
- 海鮮料理
- 素食料理
- 便當店
- 麵店
- 鐵板燒
- 甜點店
- 早午餐
- 麵包店
- 咖啡廳

5.3 餐廳詳細資訊

餐廳詳細資訊的取得，我們是使用 Google Place API，將 338 種餐點作為關鍵字，尋找在長庚大學方圓五公里內，有賣這些餐點的餐廳，將這些餐廳的名稱、電話、地址、經緯度、價位、評分、營業時間、販售的餐點，以及 Google Maps 網址，儲存至資料庫中，目前總共蒐集了 7154 間餐廳。

另外，餐廳的封面照和菜單，我們使用 Selenium 爬蟲工具，透過餐廳的 Google Maps 網址，連結到餐廳在 Google Maps 上的頁面，動態擷取餐廳的封面照和菜單照片的連結，儲存至資料庫。

圖 10、餐廳的封面照及菜單

5.4 資料更新流程

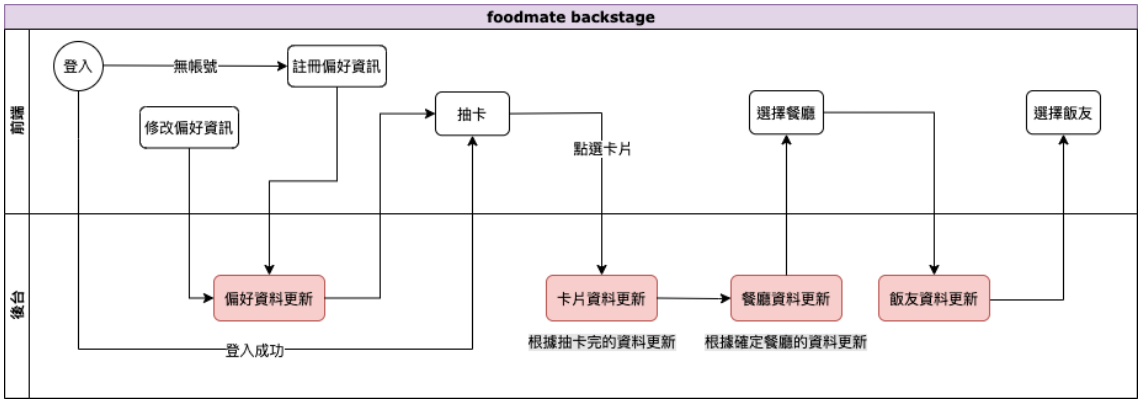

圖 11、資料更新流程圖

（圖 11）為前端與後端銜接後的流程圖。可以看出使用者在操作 APP 的過程中，有四個更新資料庫的時機，以下將詳細說明。

流程的最後，可以根據使用者每次操作後的結果進行資料更新，並提供個人化的推薦。

5.4.1 偏好資料更新

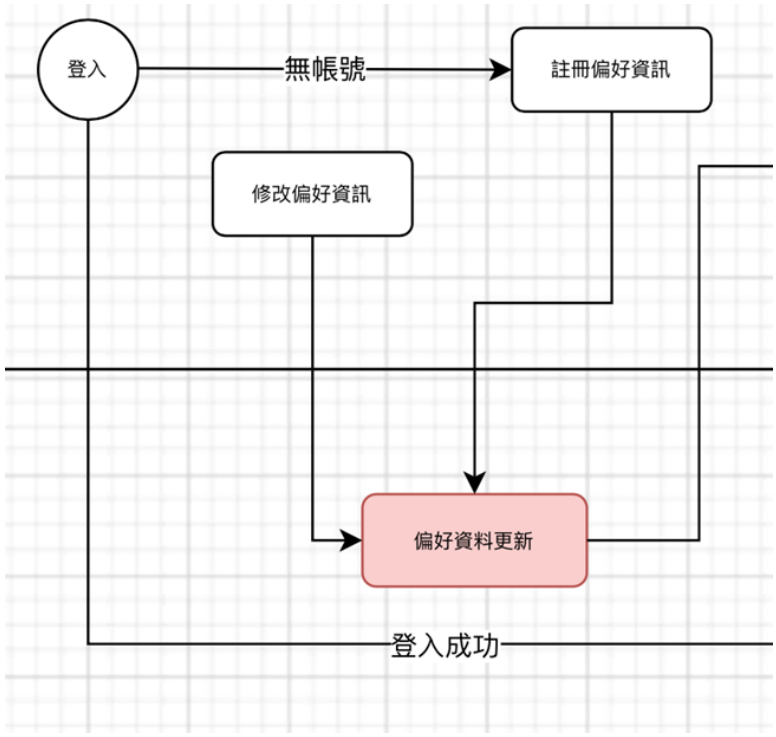

圖 12、偏好資料更新流程圖

有兩種可能會讓偏好資料更新，第一為使用者尚無帳號時，進行註冊偏好資訊；第二種可能為使用者登入後修改偏好資訊。

例如：使用者最近對日式特別偏愛，就可以到修改頁面進行修改，而資料庫即會更新，將結果更貼近使用者的需求。

5.4.2 餐點卡片資料更新

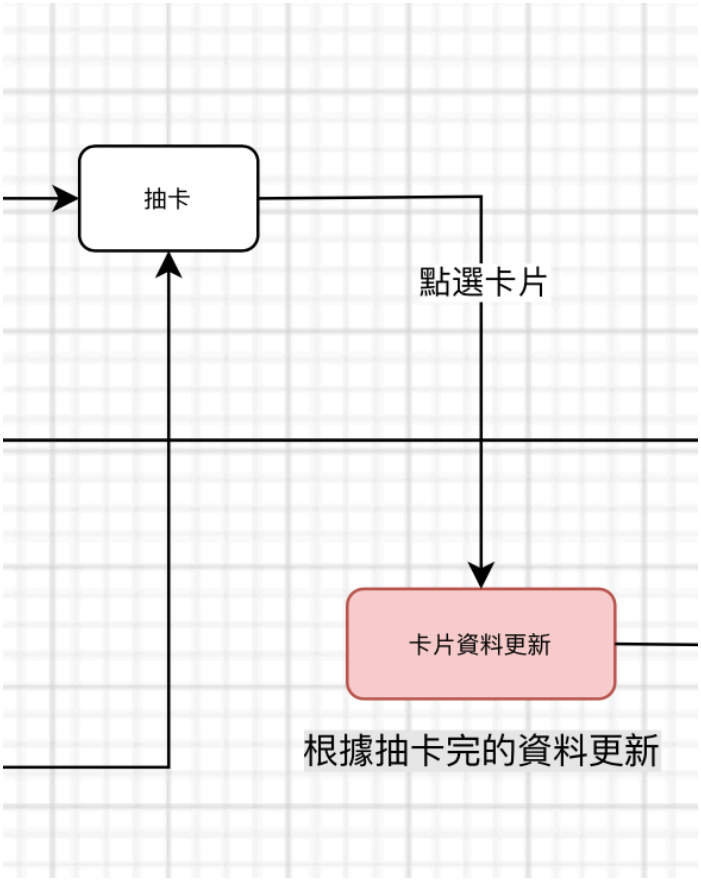

圖 13、餐點卡片資料更新

再來，使用者會進行餐點抽卡，而在確定點選餐點卡片後，卡片的資料就會進行更新。

例如：user1 這次選擇牛肉麵，並且不喜愛義大利麵。也就是說，演算法會根據這個結果將 user1 判定為比較喜愛中式料理。

5.4.3 餐廳資料更新

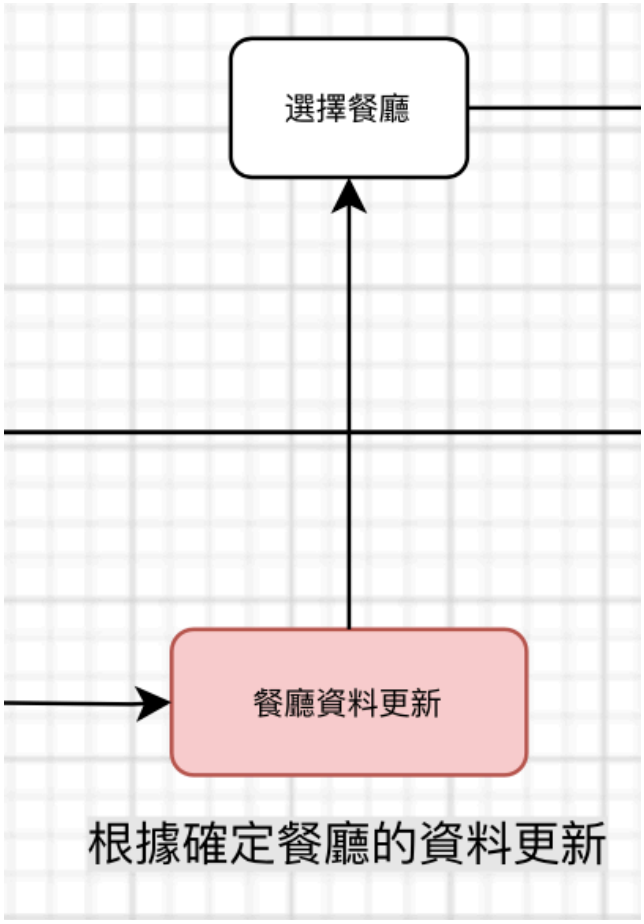

圖 14、餐廳資料更新流程圖

在選擇餐廳之前，我們會根據使用者所在位置的天氣及選擇的餐點卡片進行餐廳資料更新，提供符合使用者需求的餐廳選項。

5.4.4　飯友資料更新

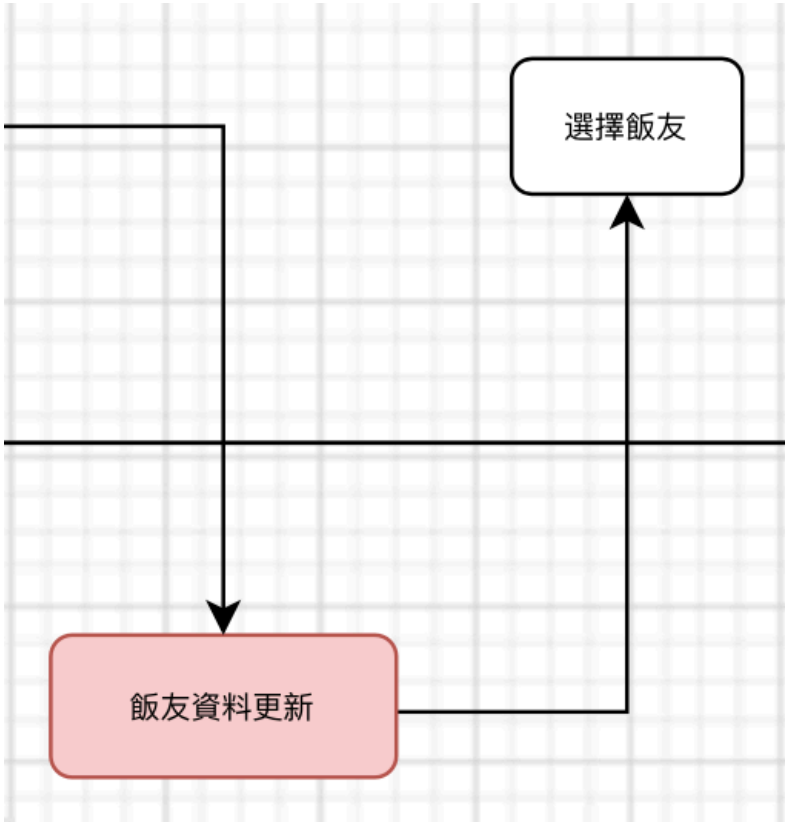

圖 15、飯友資料更新流程圖

選擇完餐廳後，就進行到飯友資料更新，我們根據使用者選擇餐廳後更新所在位置有無符合條件的飯友，並且優先推薦相似度高的人給使用者。

5.5 API 設計與操作流程

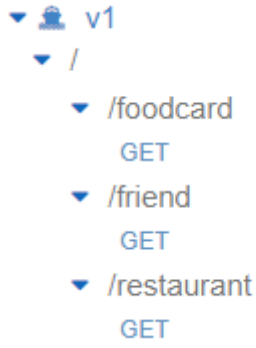

圖 16、API 架構

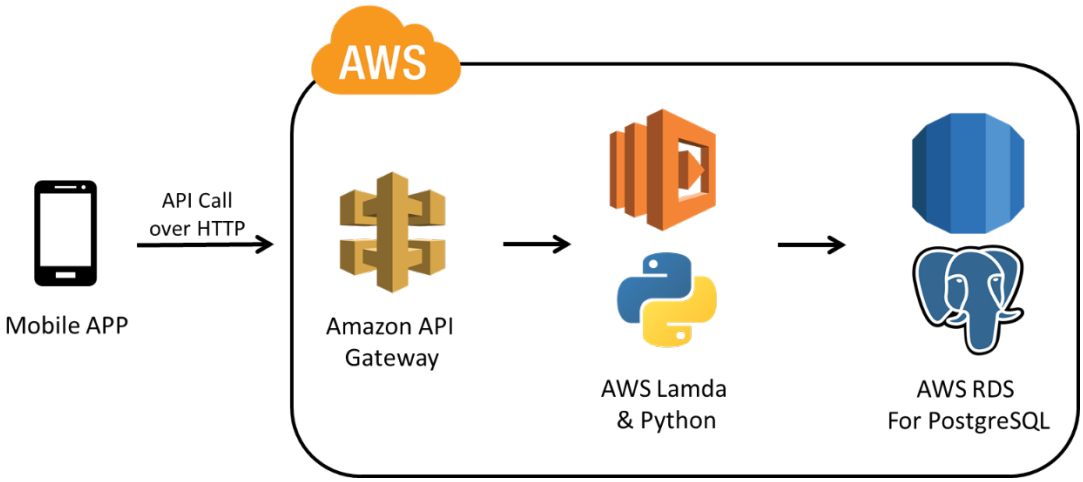

圖 17、APP 呼叫 API 流程

為了讓 APP 存取演算法計算出的個人化推薦內容，我們使用 Amazon API Gateway 建立 Lambda 代理整合的 RESTful API。當 APP 呼叫 API 後，API Gateway 會觸發存放在 AWS lambda 的程式碼，傳入參數執行演算法，最後以 json 格式將推薦內容回傳給 APP（圖 17）。

依照推薦內容我們將 API 細分出三個子層（圖 16）：

1. /foodcard
   - 用於取得餐點的個人化推薦內容
   - 參數：user_id
   - 範例：

     /foodcard?user_id=abc@gmail.com

2. /restaurant
   - 用於取得餐廳的個人化推薦內容
   - 參數：user_id、food_name、lat、lng、weather
   - 範例：

     /restaurant?user_id=abc@gmail.com&food_name=水餃&lat=25.033467&lng=121.387467&weather=Rain

3. /friend
   - 用於取得飯友的個人化推薦內容
   - 參數：user_id、lat、lng
   - 範例：

     /friend?user_id=abc@gmail.com&lat=25.033467&lng=121.387467

5.6 個人化推薦演算法

5.6.1 Content-based 推薦演算法

我們使用 Content-based 推薦演算法，提供餐點、餐廳和飯友的個人化推薦內容，演算法分為三個步驟。

1. Item Representation

   為每個 item 抽取一些特徵屬性，進行內容分析。

2. Profile Learning

   利用使用者對於過去喜歡或不喜歡的 item 特徵資料，來學習該使用者的喜好特徵。

3. Recommendation Generation

   將特徵學習得到的使用者特徵進行比較，計算餘弦相似度 (cosine similarity)，推薦和使用者的喜好特徵的相似度高的 item 給使用者。

5.6.2 餐點推薦演算法

1. Item Representation

   第一步，將食物類別作為餐點的特徵屬性，區分出這項餐點為中式、西式、甜點類等等。

2. Profile Learning

   接著，根據使用者的食物偏好和抽選卡片的歷史紀錄，了解使用者偏好口味。我們以食物偏好為預設值，有選擇為 1 分，沒有選擇為 0，再根據使用者的抽卡紀錄調整對餐點的喜好程度，當使用者點選餐點，會將餐點分數+1，依此類推，最愛+2，喜歡+1，不喜歡-1，討厭-2，最後進行平均，作為使用者的餐點偏好參考值。

3. Recommendation Generation

   了解使用者偏愛的餐點條件後，就能計算餐點與使用者偏好口味的相似度，依照相似度的分數進行排序，分數越高則排序越前面，例如喜歡中式料理的使用者，牛肉麵、水餃等中式料理的順序就會在比較前面。最後，我們會取出前 100 名的餐點做隨機排序，隨著氣溫變化，還會放入不同的餐點，例如天氣冷會加入火鍋、羊肉爐等熱食，增加抽到這些餐點的機會。

5.6.3 餐廳推薦演算法

1. Item Representation

   第一步，將餐廳販售的餐點、評分、價位、是否提供 Wi-Fi、插座或寵物友善的用餐環境，作為餐廳的特徵屬性。

2. Profile Learning

   再來，根據使用者偏好的餐廳評分、預算、用餐環境等，了解使用者想要的餐廳需求。

3. Recommendation Generation

   了解使用者偏愛的餐廳條件後，就能找出使用者設定的範圍內有賣該餐點的餐廳，並計算與使用者偏好的用餐環境的相似度，依照相似度的分數進行排序，分數越高則排序越前面，例如：使用者喜歡評分達到 4 星以上、中等價位、有 Wi-Fi 的餐廳，則越符合以上條件的餐廳，就會排在越前面。而若當下氣候氣候不佳下雨，則會直接以餐廳和使用者的距離排序，讓使用者能就近用餐。

5.6.4 飯友推薦演算法

1. Item Representation

   第一步，將使用者的年齡、性別作為特徵屬性。

2. Profile Learning

   再來，根據使用者偏好設定的年齡、性別，了解使用者想要的飯友需求。

3. Recommendation Generation

   了解使用者想要飯友條件後，就能找出使用者設定的範圍內，其他使用者與這些條件的相似度，依照相似度的分數進行排序，分數越高則排序越前面，例如：使用者偏好和 20 到 30 歲的女性用餐，則越符合條件的使用者就會排序在前面。

第六章　結論與未來展望

6.1　結論

FoodMate 提供使用者一個輕鬆有趣的使用環境，能透過有趣的餐點抽卡模式，挑選想吃的餐點並決定想去的餐廳，輕鬆愜意的坐下來與飯友用餐，更貼近使用者的生活。我們的 APP 能做到：

1. 拓展交友圈

讓我們有離開舒適圈的機會，透過飯友功能認識新朋友。

2. 解決餐點選擇的困擾

對於在選擇上有困擾的人，更快速地知道自己想吃什麼。

3. 個人化服務

跟一般的吃飯 App 相比，我們提供了個人化的服務，依據個人喜好去設定食物偏好、餐廳偏好以及飯友偏好，進而去推薦符合使用者需求的餐點、餐廳和飯友。

4. 為選擇障礙增添樂趣

透過系統有趣的抽卡模式，來解決選擇障礙的問題。

5. 更加了解自身的飲食偏好

平常不知道要吃什麼，透過系統個人化推薦，來提供更貼近使用者喜好的餐點。

6.2 未來展望

1. 持續蒐集其他地區的餐廳，期望未來位於台灣各個角落的使用者都能使用我們 APP。
2. 串接金流，讓使用者和飯友用餐完畢後，可以連結到電子支付平台（如 Line Pay、街口支付…）分攤付款金額。
3. 規劃商業營利模式，例如：廣告插入到卡片中，讓使用者左右滑動，表示喜歡或不喜歡，提供個人化的廣告體驗。
4. 可以新增卡路里功能，在食物卡片上標示卡路里，多一項使用者可能會考慮的因素，更貼近使用者心理，也可以連結 iPhone 的健康 app，輸入飲食紀錄做整合。

參考文獻

1. Chaudhry Talha（2019 年 10 月）。Simple Text Chat App using Firebase in Swift 5。檢自 https://ibjects.medium.com/simple-text-chat-app-using-firebase-in-swift-5-b9fa91730b6c
2. Amazon Web Services （2020 年）。AWS Documentation。檢自 https://docs.aws.amazon.com/index.html
3. Google Maps Platform（2020 年）。Getting Started。檢自 https://developers.google.com/maps/documentation/ios-sdk/start
4. Google Developers （2020 年）。Places API。檢自 https://developers.google.com/places/web-service/overview
5. 台灣 PostgreSQL 社群(2020 年)。PostgerSQL 正體中文使用手冊。檢自 https://docs.postgresql.tw/
6. 一起學 Python 112 :使用 numpy 庫計算經緯度間的距離（2018 年 7 月）。檢自 http://wyj-learning.blogspot.com/2018/07/python-112-numpy.html
7. Mattia Gasparini （2020 年 4 月）。Scraping Google Maps reviews in Python。檢自 https://towardsdatascience.com/scraping-google-maps-reviews-in-python-2b153c655fc2

8. Stefan French （2020 年 4 月）。Python packages in AWS Lambda made easy。檢自 https://towardsdatascience.com/python-packages-in-aws-lambda-made-easy-8fbc78520e30

附錄

使 用 者 介 面

圖 18、使用者登入介面

圖 19、使用者註冊帳號介面

圖 20、註冊個人資料（1）


4:50
More
請填寫更多關於您的資訊
居住城市
請選擇
職業類型
請選擇
簡短的描述一下自己吧!
註冊
選擇居住城市
選擇職業類型
自我介紹
註冊

圖 21、註冊個人資料（2）

圖 22、偏好設定－食物類型偏好

圖 23、偏好設定－餐廳類型偏好

圖 24、偏好設定－飯友類型偏好


往上下左右
滑動卡片
Menu
1. 設定
2. 飯友列表
3. 用餐紀錄
顯示抽卡規則
5:44
16.94°C
牛肉麵
2:56
最愛
不喜歡
喜歡
黑名單
點擊螢幕即可關閉

圖 25、餐點推薦介面

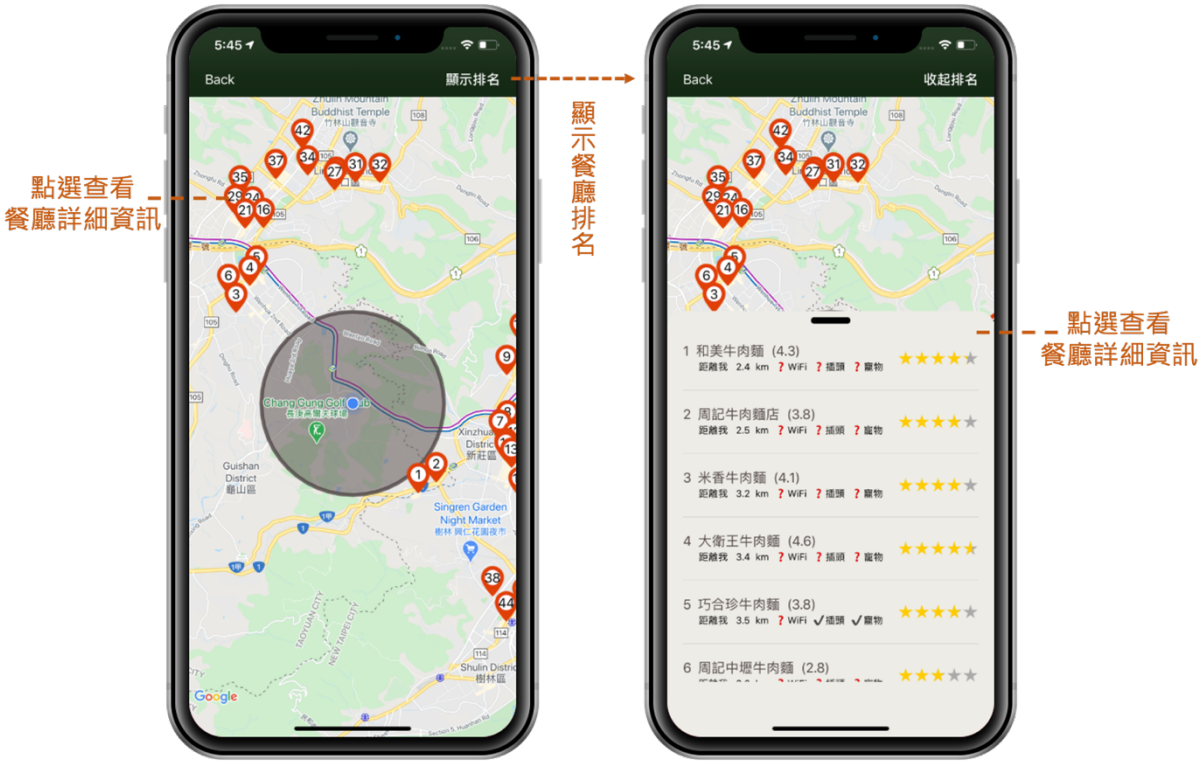
點選查看
餐廳詳細資訊
5:45
Back
顯示排名
顯示餐廳排名
收起排名
點選查看
餐廳詳細資訊
1 和美牛肉麵 (4.3)
2 周記牛肉麵店 (3.8)
3 米香牛肉麵 (4.1)
4 大衛王牛肉麵 (4.6)
5 巧合珍牛肉麵 (3.8)
6 周記中壢牛肉麵 (2.8)

圖 26、餐聽地圖介面


5:45
Back
餐廳資訊
米香牛肉麵
4.1
營業時間
聯絡資訊
星期日: 休息
星期一: 11:30 – 20:00
星期二: 11:30 – 20:00
星期三: 11:30 – 20:00
星期四: 11:30 – 20:00
星期五: 11:30 – 20:00
星期六: 休息
Address
333桃園市龜山區文化五路130號
Phone Number
0937823962
Restaurant menu
菜單
EAT
餐廳照片與餐廳名
餐廳評分
營業時間
跳至地圖
撥打電話
查看菜單
選定餐廳

圖 27、餐廳詳細資訊


跳至地圖
進行導航
顯示飯友清單
點選查看
飯友資訊
5:45
Back
餐廳資訊
米香牛肉麵
要不要來一場邂逅呢?
選擇一位飯友一起趣吃飯
今天只想一個人
星期三: 11:30 – 20:00
星期四: 11:30 – 20:00
星期五: 11:30 – 20:00
星期六: 休息
0937823962
Restaurant menu
菜單
EAT
J
Jenny
T
Tim
波
波波

圖 28、推薦飯友介面

圖 29、飯友資訊介面與傳送邀請選項


5:46
16.94°C
畢專飯友想與你共進佳餚
地點: 米香牛肉麵
時間： 6:45 PM
付款方式： AA制
拒絕
接受
日式炸豬排
關閉訊息框
跳至地圖導航
5:47
FoodMate
No. 130 Wenhua 5th Road
13 min
16 min
13 min
59°
To No. 130 Wenhua 5th Road
From Chang Gung University
13 min
2.8 mi · Fuxing 3rd Road
Fastest route
GO
Drive
Walk
Transit
Ride

圖 30、接收邀請介面


餐廳名
詢問是否有wifi
詢問是否
可攜帶寵物
FOODMATE
米香牛肉麵
餐廳評價：
3.5
送出
滑動對
餐廳評價
1. 餐廳是否有wifi
是
否
不知道
2. 餐廳是否有插座
是
否
不知道
詢問是否有插座
3. 餐廳是否可攜帶寵物
是
否
不知道
FOODMATE
one meal ; a relationship started
餐廳資訊意見調查
請協助我們更進一步了解您
請回答以上問題

圖 31、餐廳回饋通知


6:14
設定
使用者
個人檔案
邀請通知
偏好
我的最愛
黑名單
食物類型偏好
餐廳類型偏好
飯友類型偏好
幫助
回報問題
登出
登出
關閉飯友
邀請通知
偏好修改
查看個人檔案
查看我的最愛
與黑名單
登出帳號

圖 32、設定介面


6:15
設定
PROFILE
點選即可
修改資料
Hellooooo
姓名
畢專
性別
男
女
出生年
1999
居住城市
台北市
職業類型
資訊/科技業
顯示
個人資訊

圖 33、個人資料介面

圖 34、食物偏好－我的最愛


6:16
設定
Edit
黑名單
火鍋料理 - 味噌鍋
中式料理 - 油條
中式料理 - 燻肉飯
中式料理 - 牛肉麵
日式料理 - 章魚燒
中式料理 - 羹
日式料理 - 鍋貼
日式料理 - 鐵板燒
中式料理 - 雞排
中式料理 - 雞肉
日式料理 - 飯捲
火鍋料理 - 養生火鍋
甜點店 - 馬卡龍
中式料理 - 魚
點選即可
修改資料
向左滑動即可
移除黑名單

圖 35、食物偏好－黑名單

圖 36、食物類型偏好修改

圖 37、餐廳類型偏好修改

圖 38、飯友類型偏好修改


6:16
Edit
朋友列表
最愛飯友
安迪 ♂
飯友列表
hey ♀
Zoe ♀
點選即可
修改資料
向左滑動即可
移除最愛飯友
向左滑動即可
新增至最愛

圖 39、朋友列表


6:16
歷史紀錄
2020/12/25 6:45 PM
米香牛肉麵
2020/12/25 6:40 PM
韓爐韓式餐廳
2020/12/25 5:35 PM
陳記鹹酥雞 - 龜山店
2020/12/25 5:31 PM
奈良極品涮涮鍋鐵板
2020/12/25 5:28 PM
千葉火鍋林口尊爵館
2020/12/25 5:22 PM
八方雲集長庚科大店
2020/12/11 4:16 PM
鼎級prime鐵板燒
對此餐廳
進行評分
顯示
用餐時間、餐廳

圖 40、用餐歷史紀錄