

# 107學年度大學部專題競賽



國立清華大學資訊工程學系  
Department of Computer Science, National Tsing Hua University

## 餐廳候位line bot

專題成員：賴怡如

### 一、動機與目的

有感於經過有名的餐廳時，看見大量排隊人潮。有些餐廳即使已經出現一些網站或app可提供網路候位，但大部分卻限制在某個區域內的餐廳，或餐廳不能自行加入餐廳列。

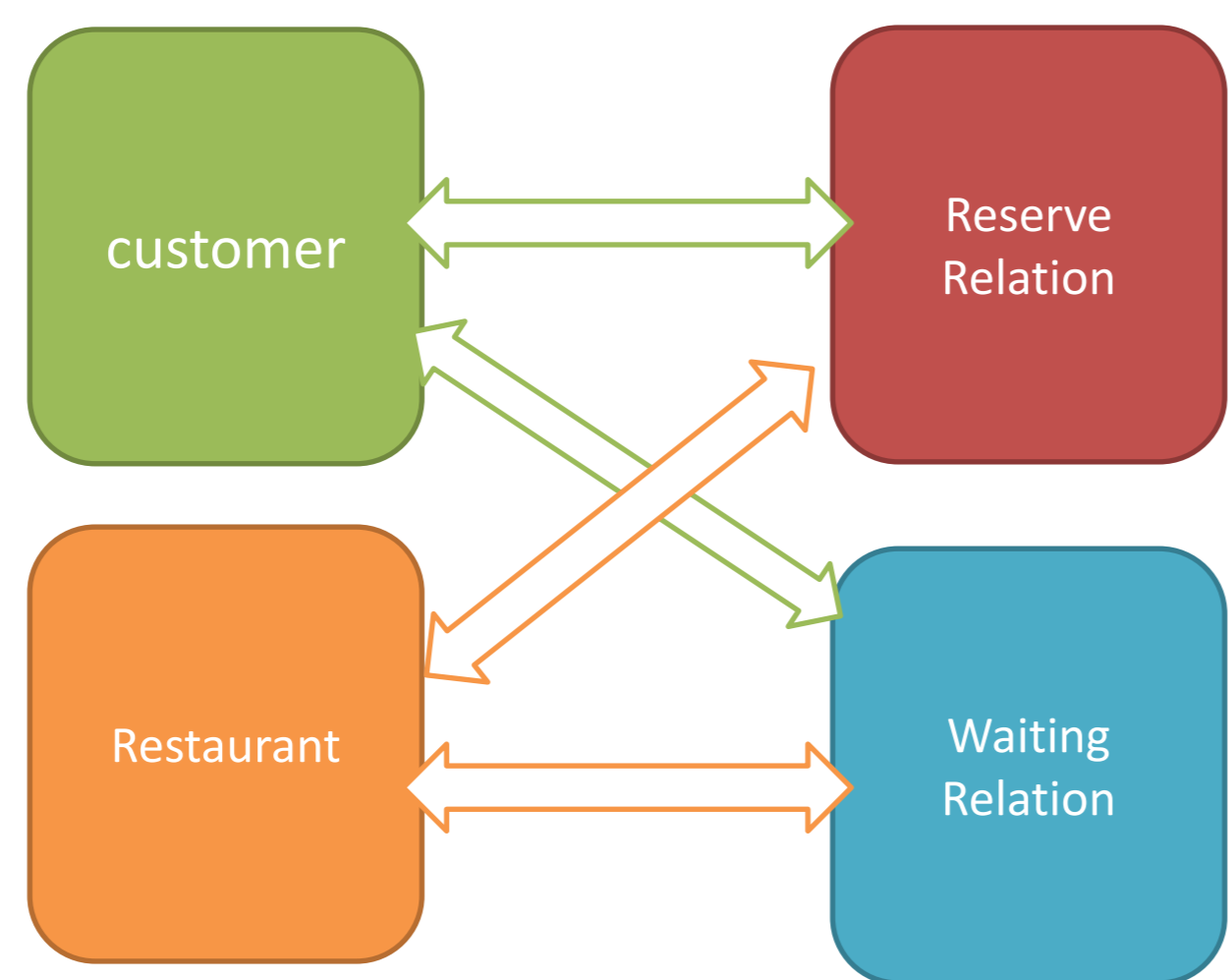
又考慮到實用層面，若實作app，使用者需要另外去app store下載才得以使用，不易讓使用者觸及。依據現代人的使用習慣，line近乎人人皆有，所以改採用line bot，以對話機器人實作，讓餐廳可以自行加入，顧客則可以在任何地點透過app排隊。

### 二、實作方法

此專題以Heroku作為後端，以python實作，並用git上傳。Database則採用Heroku Postgres，並用SQLAlchemy實作。

當使用者用line傳送訊息時，實際上是每次以一個json檔來傳送資訊，因此需要紀錄使用者的state。除了主要運用到以event.message.text來取得使用者傳送的訊息外，也用event.source.user\_id取得使用者id。另外，訊息方面主要運用line的「確認型介面」和「按鍵型介面」。

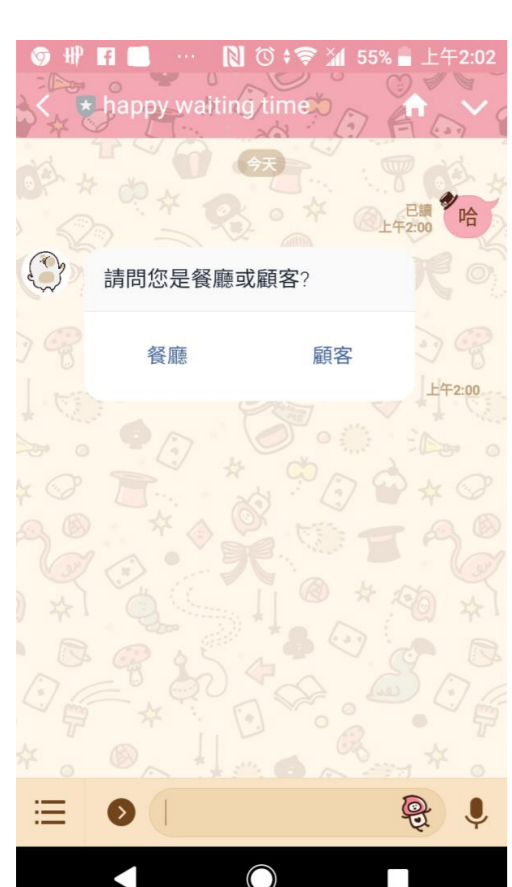
```
{
  "events": [
    {
      "replyToken": "ef3779fba3b349968c5d07db31eab56f",
      "type": "message",
      "timestamp": 1462629479859,
      "source": {
        "type": "user",
        "userId": "U4af4980629..."
      },
      "message": {
        "id": "325768",
        "type": "text",
        "text": "Hello, world"
      }
    }
  ]
}
```



Database方面，主要建立了四個class，記錄顧客資訊的Customer、各間餐廳資訊的Restaurant、雙方候位資訊的Waiting\_Relation和訂位資訊的Reserve\_Relation。

預估時間方面，每人平均等候時間=現在時間-前一組時間)/此組人數，再求出新平均等候時間=此組平均等候時間\*(1/資料數)+原過去時間\*(過去資料數-1/資料數)

### 三、成果



一開始傳送訊息至line bot，line bot將詢問使用者身分為餐廳或顧客



此為當使用者選擇身分為餐廳的主選介面



此為當使用者選擇身分為顧客的主選介面

### (一)餐廳：

#### 1. 輸入修改餐廳資料



餐廳可透過line bot自行修改餐廳名稱、餐廳介紹、餐廳圖片和餐廳網址。且當餐廳修改餐廳名稱後，顧客才可搜尋到該餐廳。

#### 2. 開始/結束候位



餐廳開始候位後，將一則則依候位順序詢問顧客是否到達，且line bot除了提醒該組，也主動提醒後五組顧客；結束候位即刪除所有未輪到的候位。

#### 3. 查詢訂位資訊



餐廳可選擇查詢指定日期的訂位或選擇直接列出所有日期的訂位。

#### 4. 確認訂位訊息



當有新訂位需求時，餐廳需確認訂位訊息，選擇拒絕或同意訂位，line bot將傳送訂位結果至顧客

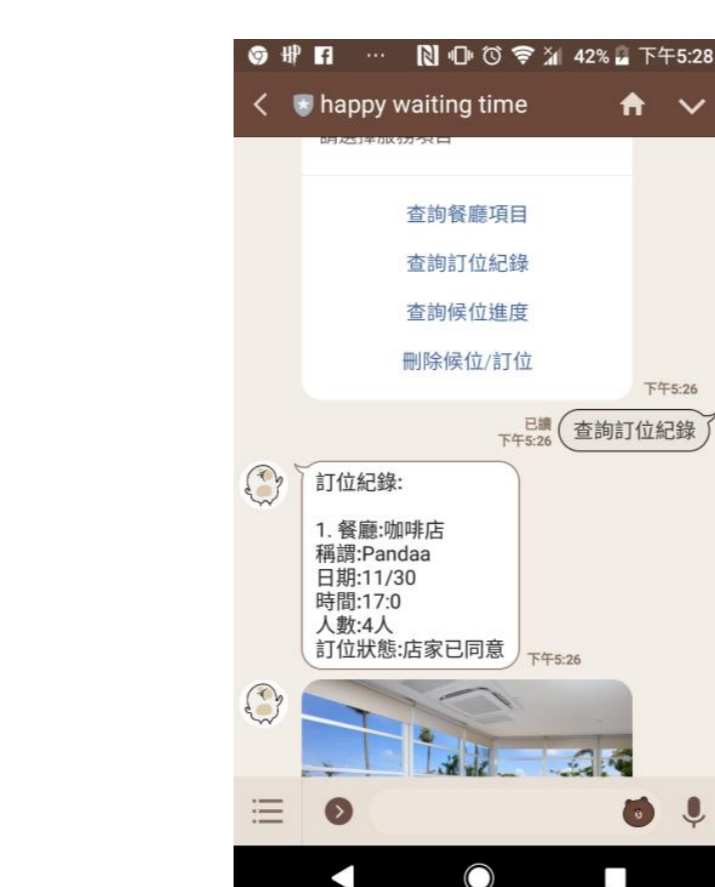
### (二)顧客

#### 1. 查詢餐廳項目



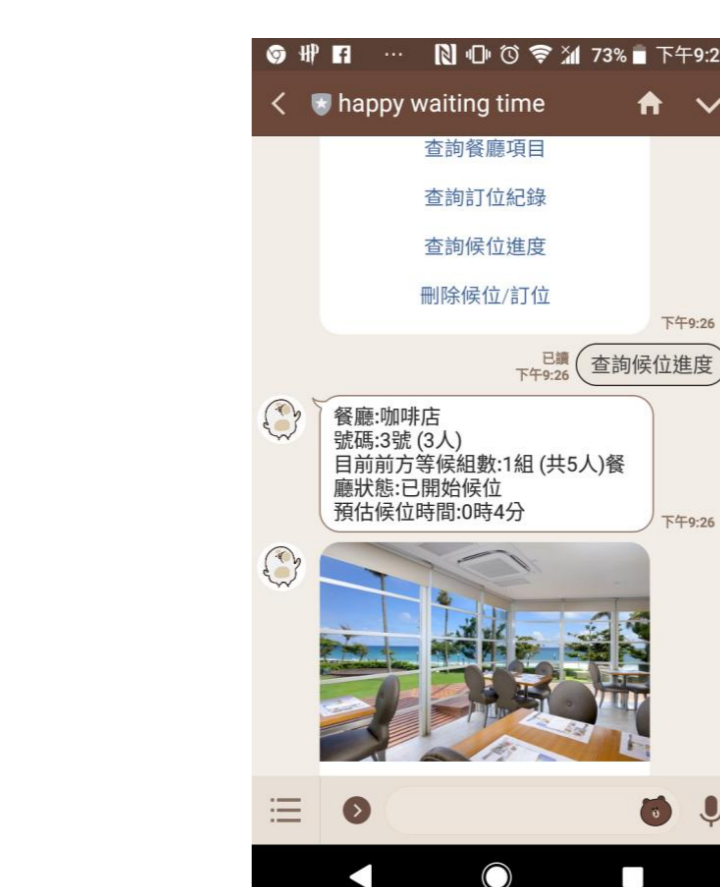
顧客查詢餐廳項目，line bot將列出所有餐廳名單，顧客輸入對應名稱，將查詢到該餐廳的主頁(主頁資訊餐廳可自行設置)。顧客若選擇候位，將取得候位號碼；顧客若選擇訂位，需輸入時間、日期等資訊，完成後餐廳將收到訂位需求通知。另外，顧客也可事先查詢候位人數。

#### 2. 查詢訂位紀錄



顧客可查詢訂位紀錄，其中除了訂位資訊，也會列出店家是否已同意訂位

#### 3. 查詢候位進度



顧客可查詢候位進度，其中包含前面候位組數、總人數和預估等候時間

#### 4. 刪除候位/訂位



顧客可選擇刪除候位或訂位，line bot皆會列出所有紀錄，顧客再輸入欲刪除的項目即可。