

美食助理

智聯專班第五組

組員：姜可言、馬壽聰、王鵬嘉

指導老師：簡志聰

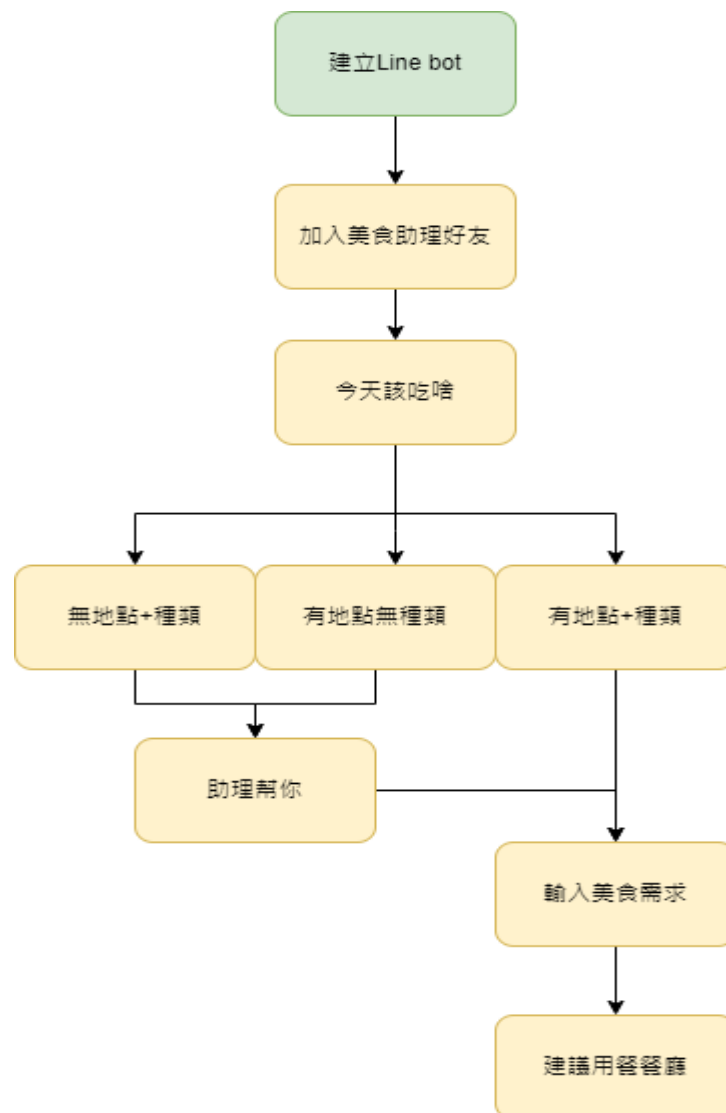
目錄

一、動機	2
二、架構設計	3
三、前言	4
四、程式執行流程圖	5
五、接收用戶訊息	6
六、爬蟲程式	7
七、FlexSendMessage 物件	8
八、line bot 的操作使用說明	10
九、未來展望	13
十、問題的發現與解決	14
參考資料	15

一、動機

身在資訊如此多元的時代，出門在外的選擇總是五花八門，對於行程的安排，時常有著令人難以抉擇的狀況產生，為了避免對於選擇上太多的困難，影響出遊的心情，我們希望由機器人來幫我們做選擇，因此有了美食助理的發想。

二、架構設計

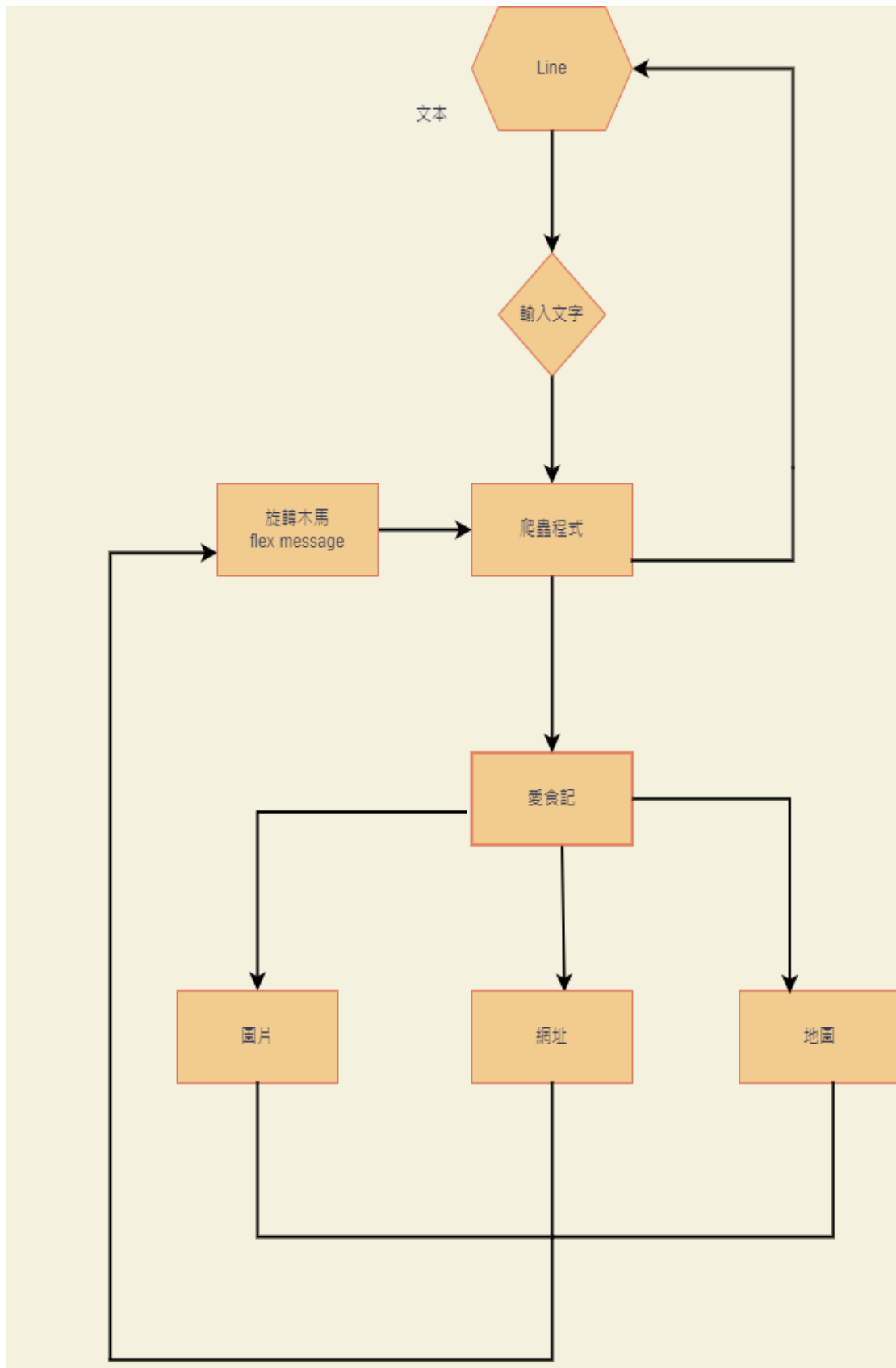


三、前言

本系統以 `python Line bot` 為主要設計架構,以 `heroku` 做為後端程式撰寫與連線資料庫，前端的程式我們選用 `linebot` 作為前端介面，提供使用者變得輸入管道，提升用戶體驗,我們採用 `Git` 作為版本取數的媒介，`heroku` 作為伺服器存取爬蟲下來的檔案供 `linebot` 抓取，並回傳爬蟲的結果到 `linebot` 上，以回應使用者的需求。爬蟲的網頁我們選用愛食記，因為比較多人使用，在評論的公信力上比較有參考價值。

由於本次皆為遠端開發，因此我們的討論都是在 `jitsi` 上討論，檔案的交流是用 `Line App` 和雲端硬碟上完成共享，流程圖的製作則是使用 `diagram.drawio`，呈現程式的執行流程。

四、程式執行流程圖



五、接收用戶訊息

```
@handler.add(MessageEvent, message=TextMessage)
def handle_message(event):
    city = event.message.text.split(' ')[0]
    food = event.message.text.split(' ')[1]
    message = eat(city, food)
    # print(type(message))
    line_bot_api.reply_message(event.reply_token, message)

import os
if __name__ == "__main__":
    app.run(host='0.0.0.0', port=os.environ.get('PORT', 5000))
```

用戶輸入關鍵字後，用空白隔開城市名稱和食物，使這兩個參數輸入到 eat()function 中，再進行爬蟲的執行，並對回傳值進行接收。

下面使用 OS 設定本機測試的路徑。

六、爬蟲程式

```
#爬蟲
def eat(x,y):
    import requests
    from bs4 import BeautifulSoup
    store_list = []
    star_list = []
    address_list = []
    img_url_list = []
    time_list = []
    url_list = []
    map_list = []
    keys1 = x
    keys2 = y

    my_headers = {'user-agent': 'Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/98.0.4758.16'}
    r = requests.get("https://ifoodie.tw/explore/" + keys1 + "/list/" + keys2, headers = my_headers)
    if r.status_code == 200:
        # print(r.text)
        soup = BeautifulSoup(r.text, 'html.parser')
        title = soup.find_all('a', class_="jsx-558691085 title-text")
        star = soup.find_all('div', class_="jsx-1207467136 text")
        address = soup.find_all('div', class_="jsx-558691085 address-row")
        img = soup.select("div.jsx-3081451459.item-list img")
        time = soup.select("div.jsx-558691085.info")
        url = soup.find_all('a', class_="jsx-558691085 click-tracker")
    #抓出來存取到list
    for t,l,a,h,u in zip(title,star,address,time,url):
        store_list.append(t.text)
        star_list.append(l.text + "☆☆")
        address_list.append(a.text)
        time_list.append(h.text)
        url_list.append("https://ifoodie.tw"+u.get('href'))
        map_list.append("https://www.google.com.tw/maps/dir/" + quote(a.text))

#圖檔判別
for i in img:
    if 'src' in i.attrs:
        if i['src'].startswith('data'):
            i_url=i.get('data-src')
            if i_url.endswith('360'):
                img_url_list.append(i_url)
        else:
            i_url=i.get('src')
            if i_url.endswith('360'):
                img_url_list.append(i_url)

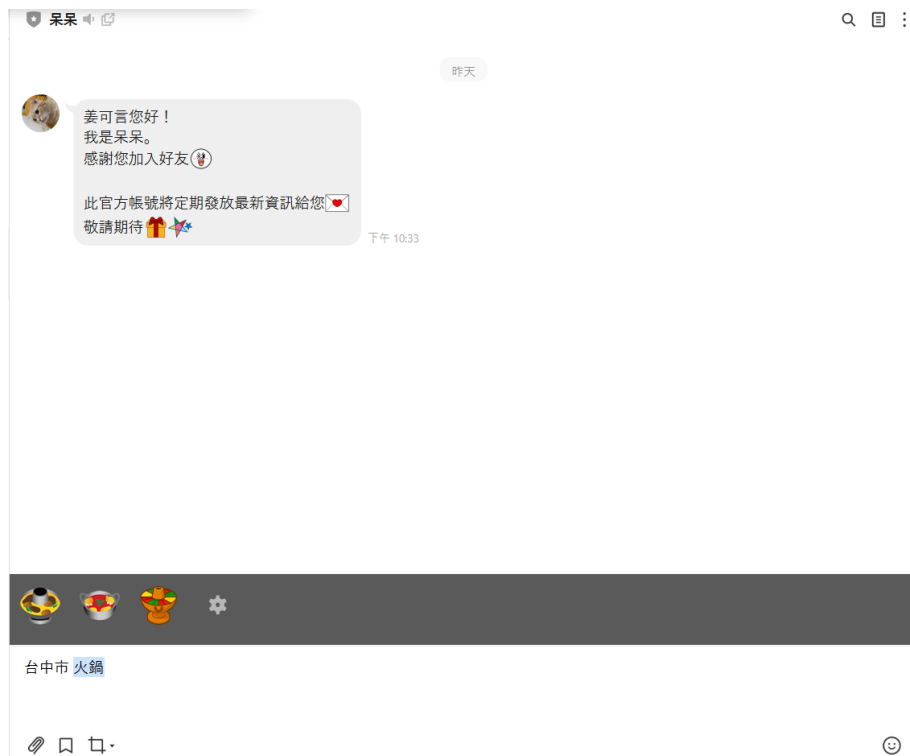
message = image_carousel(store_list,star_list,address_list,img_url_list,time_list,url_list,map_list,"收尋美食結果")
return message
```

我們是用城市名稱(keys1)和食物種類(keys2)作為關鍵字，通過 beautifulsoup 爬蟲程式去愛食記網站中爬取資訊，並將圖片 url、營業時間、店家地址、網站 url 存取為 List。

七、FlexSendMessage 物件

```
def image_carousel(store_list,star_list,address_list,img_url_list,time_list,url_list,map_list,alt_text):
    contents = dict()
    contents['type'] = 'carousel'
    contents['contents'] = []
    i=0
    for store, star, address, img, time ,url1,mapurl in zip(store_list, star_list, address_list, img_url_list, time_list, url_list, map_list):
        if i<10:
            bubble = {
                "type": "bubble",
                "hero": {
                    "type": "image",
                    "url": img,
                    "size": "full",
                    "aspectRatio": "20:13",
                    "aspectMode": "cover",
                },
                "body": {
                    "type": "box",
                    "layout": "vertical",
                    "contents": [
                        {
                            "type": "text",
                            "text": store[:30]if len(store)<30 else store[:30] + '...',
                            "weight": "bold",
                            "size": "xl"
                        },
                        {
                            "type": "box",
                            "layout": "baseline",
                            "margin": "md",
                            "contents": [
                                {
                                    "type": "icon",
                                    "size": "sm",
                                    "url": "https://scdn.line-apps.com/n/channel_devcenter/img/fox/review_gold_star_28.png"
                                },
                                {
                                    "type": "text",
                                    "text": star,
                                    "size": "sm",
                                    "color": "#999999",
                                    "margin": "md",
                                    "flex": 0
                                }
                            ]
                        },
                        {
                            "type": "box",
                            "layout": "vertical",
                            "margin": "lg",
                            "spacing": "sm",
                            "contents": [
                                {
                                    "type": "box",
                                    "layout": "baseline",
                                    "spacing": "sm",
                                    "contents": [
                                        {
                                            "type": "text",
                                            "color": "#aaaaaa",
                                            "size": "sm",
                                            "flex": 1,
                                            "text": "地址"
                                        },
                                        {
                                            "type": "text",
                                            "text": address,
                                            "wrap": True,
                                            "color": "#666666",
                                            "size": "sm",
                                            "flex": 5
                                        }
                                    ]
                                }
                            ]
                        }
                    ]
                }
            }
            i+=1
```


八、line bot 的操作使用說明



輸入“城市名稱”+ 空白 + “食物名稱”後，按下發送。



會跳出 10 個 FlexSendMessage 物件，使用者可以點選“地圖”、“網站”和“電話”按鈕。按下電話按鈕後會跳出商家電話，以供使用者撥

打。



呆呆

4.4★

地點 臺中市新社區協中街287號

營業... 今日營業: 11:00-19:00

地圖

網站

電話

4

地點

營業.

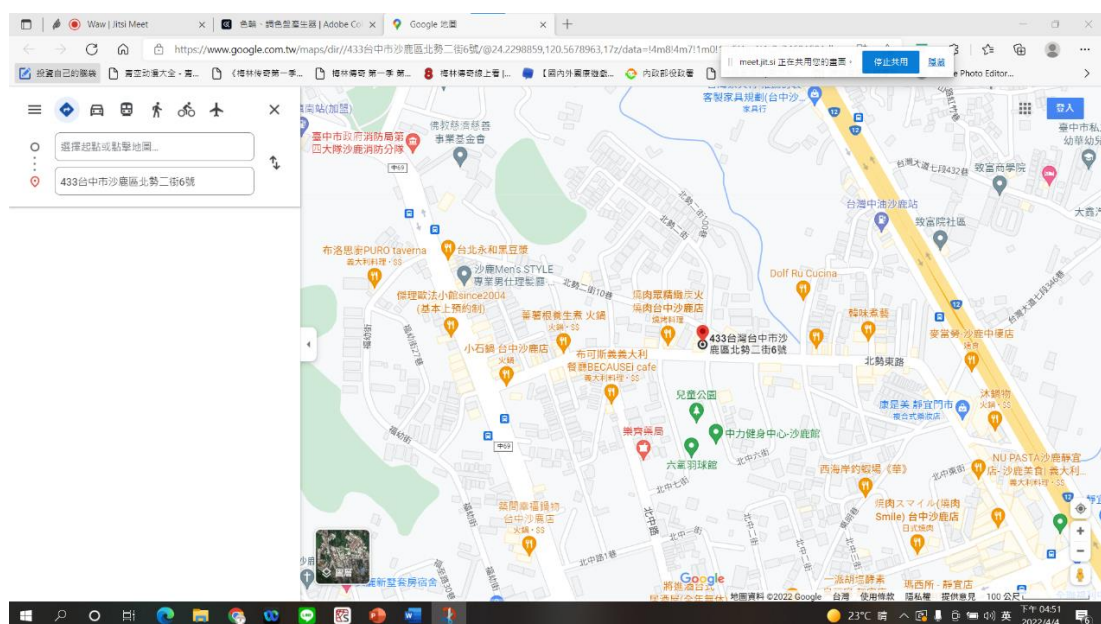
上午 11:49

已讀 上午 11:52 菇神電話: 0425822585

輸入訊息



點擊“地圖”按鈕後會連上商家地址的 google map 頁面供使用者搜尋。



點擊”網站”按鈕後會連上商家地址的網頁供使用者搜尋。



The screenshot displays the '愛食記' (Love Food) website interface. The header includes the site name, navigation links for '美食分類、餐廳' (Food Categories, Restaurants), and user options like '登入' (Login), '發表食記' (Post Food Review), '下載APP' (Download App), '專欄文章' (Column Articles), '店家專區' (Merchant Special Area), and '排行榜' (Ranking). A secondary navigation bar shows '城市' (City), '美食' (Food), and '美食專欄' (Food Column).

The main content area features the profile of '驛庭鍋物' (Yi Ting Hot Pot), which includes a 4.3-star rating, 12 reviews, and an average price of \$550. The address is listed as '台中市沙鹿區北勢二街6號'. A map shows the location near '台灣中油沙鹿站'. The page also displays '附近餐廳排名' (Nearby Restaurant Ranking) for '台中市 火鍋' (Taichung City Hot Pot), listing '無老鍋 公益店' (4.2 stars, 25 reviews) and '鼎王麻辣鍋' (4.1 stars, 0 reviews).

Additional details include a '今日營業: 17:30-22:00' (Today's Business Hours), a '評論: 520+' (Reviews: 520+) button, and a section for '驛庭鍋物的外帶/外送服務' (Takeout/Delivery Service for Yi Ting Hot Pot).

九、未來展望

這次礙於開發時間和頁面美觀的考量，部分功能未能實裝，往後開發團隊會讓機器人的功能朝向更多元的面向發展。以下是我們想到能更加優化的部分。

分享按鍵設定

設定分享按鍵可以讓使用者將自己喜歡的餐廳分享給朋友知道。

外送點餐

使用者點選按鍵，頁面將跳到 foodpanda app 頁面上，供使用者下單。

觀看評論按鍵

點擊按鍵後會自動跳到評論區的頁面，供使用者觀看/輸入評論。

用 selenium 點擊網頁爬蟲

如果未來學會多工，在 selenium 爬蟲上更有效率的話，會加入更多功能。

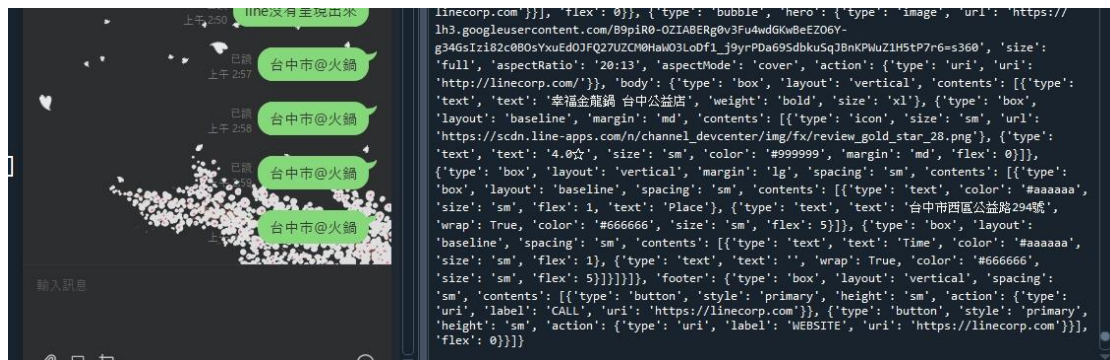
十、問題的發現與解決

1. 地圖的 ASCII 編碼轉換

```
time_list.append(n.text)
url_list.append("https://ifoodie.tw"+u.get('href'))
map_list.append("https://www.google.com.tw/maps/dir/"+quote(a.text))
```

在 google map 網址抓取時，因為發現無法用中文編入網址格式中，查找資料之後發現是中文地址需要轉換成 ASCII 編碼，才能符合 url 格式。

2. 回傳 FlexSendMessage 物件時無法顯示



```
@handler.add(MessageEvent, message=TextMessage)
def handle_message(event):
    city = event.message.text.split('@')[0]
    food = event.message.text.split('@')[1]
    content = eat(city, food)
    message
    line_bot_api.reply_message(
        event.reply_token, message
        TextSendMessage(text=content))
```

當初回傳 FlexSendMessage 物件時，不知道為甚麼只回傳了文字檔，後來去網路上查詢之後發現是回傳的檔案格式設定有問題，只要把 TextSendMessage(text=content) 代碼改成要回傳的指定 FlexSendMessage 物件就可以了。

參考資料

愛食記:

<https://ifoodie.tw/>

maso 萬事屋:

[https://github.com/maso0310/linebot selenium](https://github.com/maso0310/linebot_selenium)

<https://www.youtube.com/watch?v=JNPBLnibmsM&t=844s>

[Django 教學 3] Django Migration(資料遷移)的重要觀念:

<https://www.learncodewithmike.com/2020/03/django-model-migration.html>