
一、Scenario Description

生活中常常會有許多待做事項，比如什麼時候要考試、什麼時候要交作業或是有什麼活動等等，然而我們有時候常常會不知道要如何安排這些事項的優先權級，除去每件事項的重要程度外，依照時間去安排會是一個不錯的選擇，然而目前市面上很少有單純提供計算倒數幾天的 app 與服務，即使有相應的倒數幾天 app 通常也會與日曆綁在一起作為輔助功能，然而我們可能本身就有固定使用某個日曆的習慣，因此希望能夠過 lineBot 去簡單提供這項服務，讓使用者能藉由簡短的服務快速知道剩下的日子，以安排自己的代做事項，同時也不會影響使用者本身的習慣。

二、How to design

在 app.py 中會去透過設定所連接的 IoTalk 的網址與 device name 去做連接，再透過 line 所提供的 API 的一些資訊去做連接，讓 app.py 從 lineBot 那邊取的資訊後可以再將資料轉傳給 IoTalk，等 IoTalk 回應後再將結果回傳給 lineBot 並呈現給使用者，

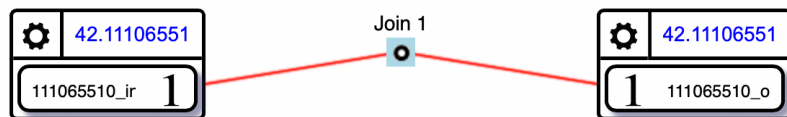
在設計上我透過 lineBot 本身提供的自動回覆功能，跟使用者介紹有提供什麼功能，並提示使用者可以輸入內容或是格式以得到結果。

在 IoTalk 上我設計在讀取使用者輸入的日期資料後，會先去檢查是否格式有符合或是內容有符合，若是沒有符合就會統一回傳提醒使用者

要輸入正確格式內容才可以運行的提醒資訊。而若內容符合就會透過
`datetime` 去讀取今天的日期，並將使用者輸入的字串轉為 `datetime` 並進行相減，若是使用者輸入的時間在未來，會告知使用者還剩下多少天，而若使用者輸入的是過去的時間，也會告知使用者已過去多少天。

三、ScreenShots

1. IoTtalk GUI connection



```
[vivian@Wei-TzudeMacBook-Pro Mini Project 4 Package % python app.py
IoTtalk Server = http://140.114.77.93:9999/
This device has successfully registered.
Device name = 42.111065510Model
Create control threading
* Serving Flask app "app" (lazy loading)
* Environment: production
  WARNING: This is a development server. Do not use it in a production deployment.
  Use a production WSGI server instead.
* Debug mode: off
* Running on http://0.0.0.0:5002/ (Press CTRL+C to quit)
```

2. DM/DF

DM 名稱 : 111065510Model

DF :

IDF: 111065510_input

ODF: 111065510_output



| 111065510Model (IDF) | | | | | Save |
|----------------------|--------|-----|-----|----------|------|
| 111065510_input | Type | Min | Max | Function | |
| x1 | sample | 0 | 0 | disabled | |



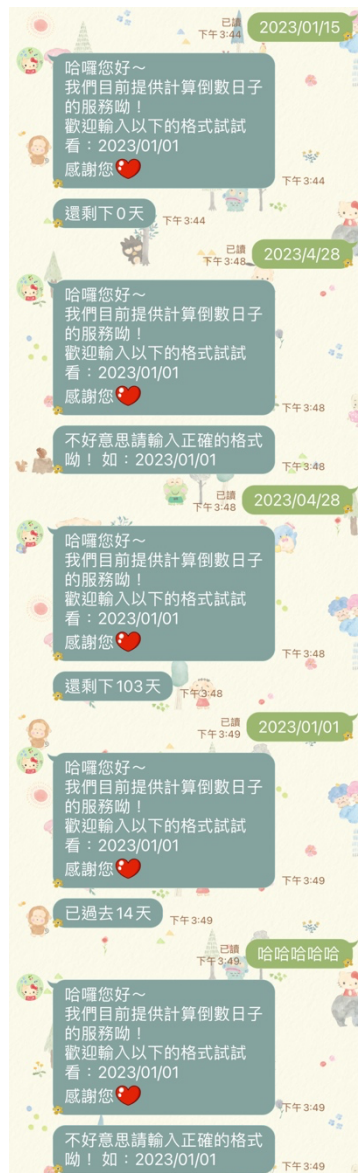
| 111065510Model (ODF) | | | | Save |
|----------------------|----------|-----|-----|------|
| 111065510_output | Function | Min | Max | |
| y1 | disabled | 0 | 0 | |

3. IDF/ODF Monitor

| IDF Monitor | |
|-------------|---|
| Sub-stage: | Input Continue Next Table 1 111065510_input ↓ |
| Timestamp | x_1 |
| 15:44:49 | 2023/01/15 |
| 15:48:36 | 2023/4/28 |
| 15:48:45 | 2023/04/28 |
| 15:49:07 | 2023/01/01 |
| 15:49:18 | 哈哈哈哈哈 |

| ODF Monitor | |
|-------------|---|
| Sub-stage: | Function Table 1 111065510_output ↓ |
| Timestamp | $y_{1,F}$ |
| 15:44:49 | 還剩下0天 |
| 15:48:36 | 不好意思請輸入正確的格式呦！如：2023/01/01 |
| 15:48:45 | 還剩下103天 |
| 15:49:07 | 已過去14天 |
| 15:49:18 | 不好意思請輸入正確的格式呦！如：2023/01/01 |

4. LineBot result



四、What I learn

這次很高興能夠有機會可以實作 lineBot，並且透過簡單的方式製作 server 並回傳，十分有趣！包括了解到 lineBot 的製作流程、如何串聯、如何運行、及 IoTtalk 的使用，都讓我十分驚奇，尤其是因為當初為了要了解 IoTtalk，因此有特別去查閱他的相關手冊，也了解了如果今天使用其他裝置，比如手機或是燈泡也能夠做出其他不一樣的應用，覺得十分有趣能有機會體會、學習到這些～

最後謝謝助教以及老師能夠設計這次的 Mini_Project，不僅有所收穫同時也十分的有趣！