

國立臺北商業大學

資訊管理系

110 資訊系統專案設計

系統手冊



組別：第 110504 組

題目：My Day 我的天

指導老師：林宏仁老師

組長：10656004 張恩瑤

組員：10656003 吳東陽 10656009 林哲卉

10656025 林欣儀 10656035 吳乙萱

中華民國 111 年 1 月 5 日

目錄

第 1 章 背景與動機	1
1-1 簡介	1
1-2 問題與機會	2
1-3 相關系統探討	3
第 2 章 系統目標與預期成果	4
2-1 系統目標	4
2-2 預期成果	4
第 3 章 系統規格	5
3-1 系統架構	5
3-2 系統軟、硬體需求與技術平台	5
3-3 開發標準與使用工具	6
第 4 章 專案時程與組織分工	8
4-1 專案時程	8
4-2 專案組織與分工	8
第 5 章 需求模型	11
5-1 功能分解圖 (Functional decomposition diagram)	11
5-2 需求清單	11
第 6 章 程序模型	13
6-1 資料流程圖(Data flow diagram)	13
6-2 程序規格書(Process specification)	18

第 7 章 資料模型	32
7-1 實體關聯圖(Entity relationship diagram)	32
7-2 資料字典(Data dictionary)	32
第 8 章 資料庫設計	35
8-1 資料庫關聯圖	35
8-2 表格及其 Meta data	35
第 9 章 程式規格	48
9-1 軟體架構與程式清單	48
9-2 程式規格描述	60
第 10 章 測試模型	84
10-1 測試計畫	84
10-2 測試個案與測試結果	89
第 11 章 操作手冊	127
第 12 章 使用手冊	128
第 13 章 感想	137
第 14 章 參考資料	140
附錄	141
附錄一---一評評審建議事項之修正情形	141
附錄二---二評評審建議事項之修正情形	144

圖目錄

圖 3-1-1、系統架構.....	5
圖 4-1-1、專案時程.....	8
圖 5-1-1、功能分解圖(FDD).....	11
圖 6-1-1、系統環境圖.....	13
圖 6-1-2、圖 0.....	13
圖 6-1-3、圖 1.....	14
圖 6-1-4、圖 2.....	14
圖 6-1-5、圖 3.....	15
圖 6-1-6、圖 4.....	15
圖 6-1-7、圖 5.....	16
圖 6-1-8、圖 6.....	16
圖 6-1-9、圖 7.....	17
圖 6-1-10、圖 8.....	17
圖 6-1-11、圖 9.....	18
圖 7-1-1、實體關聯圖.....	32
圖 8-1-1、資料庫關聯圖.....	35

表目錄

表 1-2-1、SWOT 分析	2
表 1-3-1、相關系統比較.....	3
表 3-2-1、軟硬體需求.....	5
表 3-3-1、開發環境.....	6
表 4-2-1、分工表.....	8
表 5-2-1、功能性需求.....	11
表 6-2-1、程序 1.....	18
表 6-2-2、程序 2.....	20
表 6-2-3、程序 3.....	22
表 6-2-4、程序 4.....	23
表 6-2-5、程序 5.....	24
表 6-2-6、程序 6.....	25
表 6-2-7、程序 7.....	27
表 6-2-8、程序 8.....	29
表 6-2-9、程序 9.....	30
表 7-2-1 字典編號.....	32
表 8-2-1、T1 Meta data	35
表 8-2-2、T2 Meta data	36
表 8-2-3、T3 Meta data	36
表 8-2-4、T4 Meta data	36

表 8-2-5、T5 Meta data	37
表 8-2-6、T6 Meta data	37
表 8-2-7、T7 Meta data	38
表 8-2-8、T8 Meta data	38
表 8-2-9、T9 Meta data	39
表 8-2-10、T10 Meta data	39
表 8-2-11、T11 Meta data	39
表 8-2-12、T12 Meta data	40
表 8-2-13、T13 Meta data	40
表 8-2-14、T14 Meta data	41
表 8-2-15、T15 Meta Data	41
表 8-2-16、T16 Meta Data	42
表 8-2-17、T17 Meta Data	42
表 8-2-18、T18 Meta Data	42
表 8-2-19、T19 Meta Data	43
表 8-2-20、T20 Meta Data	43
表 8-2-21、T21 Meta Data	43
表 8-2-22、T22 Meta Data	44
表 8-2-23、T23 Meta Data	44
表 8-2-24、T24 Meta Data	44
表 8-2-25、T25 Meta Data	45

表 8-2-26、T26 Meta Data	45
表 8-2-27、T27 Meta Data	45
表 8-2-28、T28 Meta Data	46
表 8-2-29、T29 Meta Data	46
表 8-2-30、T30 Meta Data	47
表 8-2-31、T31 Meta Data	47
表 9-1-1、前端程式清單.....	48
表 9-1-2、API 程式及後端程式清單	59
表 9-2-1、Dart 部分程式碼描述	60
表 9-2-2、Python 部分程式碼描述	73
表 10-2-A1、測試功能「註冊」	89
表 10-2-A2、測試功能「登入/登出」	89
表 10-2-A3、測試功能「忘記密碼」	89
表 10-2-A4、測試功能「更改密碼」	90
表 10-2-B1、測試功能「建立課表」	90
表 10-2-B2、測試功能「編輯課表」	91
表 10-2-B3、測試功能「刪除課表」	91
表 10-2-B4、測試功能「取得課表列表」	92
表 10-2-B5、測試功能「取得課表」	92
表 10-2-B6、測試功能「取得主課表列表」	93
表 10-2-B7、測試功能「編輯課表資訊」	93

表 10-2-B8、測試功能「分享課表」	94
表 10-2-B9、測試功能「刪除接收課表」	94
表 10-2-B10、測試功能「取得接收課表列表內容」	94
表 10-2-B11、測試功能「接收課表」	95
表 10-2-B12、測試功能「取得分享碼」	95
表 10-2-B13、測試功能「分享碼接收課表」	96
表 10-2-B14、測試功能「取得節次時間」	96
表 10-2-B15、測試功能「取得接收課表列表」	96
表 10-2-C1、測試功能「建立行程」	97
表 10-2-C2、測試功能「編輯行程」	97
表 10-2-C3、測試功能「刪除行程」	98
表 10-2-C4、測試功能「取得當日行程列表」	98
表 10-2-C5、測試功能「取得行程」	98
表 10-2-C6、測試功能「建立共同行程」	99
表 10-2-C7、測試功能「取得共同行程列表」	99
表 10-2-C8、測試功能「取得共同行程」	100
表 10-2-C9、測試功能「取得倒數列表」	100
表 10-2-D1、測試功能「建立讀書計畫」	101
表 10-2-D2、測試功能「編輯讀書計畫」	101
表 10-2-D3、測試功能「刪除讀書計畫」	102
表 10-2-D4、測試功能「共享讀書計畫」	102

表 10-2-D5、測試功能「取消共享讀書計畫」	103
表 10-2-D6、測試功能「取得讀書計畫列表」	103
表 10-2-D7、測試功能「取得讀書計畫」	103
表 10-2-D8、測試功能「取得群組讀書計畫列表」	104
表 10-2-E1、測試功能「主題設定」	104
表 10-2-E2、測試功能「隱私設定」	105
表 10-2-E3、測試功能「個人資料列表」	105
表 10-2-E4、測試功能「編輯個人資料」	105
表 10-2-F1、測試功能「邀請成為好友」	106
表 10-2-F2、測試功能「好友邀請回覆」	106
表 10-2-F3、測試功能「刪除好友」	107
表 10-2-F4、測試功能「新增摯友」	107
表 10-2-F5、測試功能「刪除摯友」	108
表 10-2-F6、測試功能「取得好友列表」	108
表 10-2-F7、測試功能「取得好友行程及課表」	108
表 10-2-F8、測試功能「取得好友邀請列表」	109
表 10-2-F9、測試功能「取得摯友列表」	109
表 10-2-G1、測試功能「建立群組」	110
表 10-2-G2、測試功能「取得群組/群組邀請列表」	110
表 10-2-G3、測試功能「同意/拒絕加入群組」	111
表 10-2-G4、測試功能「取得群組」	111

表 10-2-G5、測試功能「編輯群組」	111
表 10-2-G6、測試功能「群組邀請好友」	112
表 10-2-G7、測試功能「設定管理者」	112
表 10-2-G8、測試功能「退出群組」	113
表 10-2-G9、測試功能「自行加入群組」	113
表 10-2-G10、測試功能「取得成員列表」	114
表 10-2-H1、測試功能「建立臨時群組」	114
表 10-2-H2、測試功能「取得臨時群組/臨時群組邀請列表」	115
表 10-2-H3、測試功能「取得臨時群組邀請資訊」	115
表 10-2-H4、測試功能「同意/拒絕加入臨時群組」	115
表 10-2-H5、測試功能「取得臨時群組」	116
表 10-2-H6、測試功能「編輯臨時群組」	116
表 10-2-H7、測試功能「臨時群組邀請好友」	117
表 10-2-H8、測試功能「設定管理者」	117
表 10-2-H9、測試功能「退出臨時群組」	118
表 10-2-H10、測試功能「取得成員列表」	118
表 10-2-I1、測試功能「建立投票」	119
表 10-2-I2、測試功能「編輯投票」	119
表 10-2-I3、測試功能「刪除投票」	120
表 10-2-I4、測試功能「取得投票列表」	120
表 10-2-I5、測試功能「取得投票」	121

表 10-2-I6、測試功能「投票」	121
表 10-2-I7、測試功能「加入投票選項」	122
表 10-2-J1、測試功能「建立筆記」	122
表 10-2-J2、測試功能「編輯筆記」	123
表 10-2-J3、測試功能「刪除筆記」	123
表 10-2-J4、測試功能「取得筆記列表」	124
表 10-2-J5、測試功能「取得筆記」	124
表 10-2-J6、測試功能「取得群組筆記列表」	125
表 10-2-J7、測試功能「共享筆記」	125
表 10-2-J8、測試功能「取消共享筆記」	126
表 11-1-1、操作手冊.....	127
表 12-1-1、註冊及登入.....	128
表 12-1-2、新增行程.....	129
表 12-1-3、群組功能.....	130
表 12-1-4、玩聚.....	133
表 12-1-5、讀書計畫.....	135

第1章 背景與動機

1-1 簡介

My Day 是一款專為學生設計的 App，無論是學習、課程、娛樂、社團、交友都可以透過我們的 App 一手包辦，我們最主要的功能著重於時間規劃，相較市面上的行事曆，My Day 是透過課表作為時間軸，因此更能貼近學生的生活；我們致力於讓學生族群的生活能夠更方便，可以將更多的時間及精力花在學習上，因此我們改善了現有的行事曆使其能夠融入生活中，在完全不需思考的狀況下以更直覺的方式來操作所需要的功能，例如：老師說第 16 週要交報告，但是多數的同學無法及時計算出要將行程記錄在哪一個時間點，透過 My Day 我們只要輸入第 16 週的星期二第三節課，My Day 會自動將行程存入正確的時間點。

透過 My Day 可以更輕鬆得組成讀書會，並且可與大家共享筆記；對學生來說，倒數功能也是很重要的，將考試日期紀錄下來，透過倒數的功能可以清楚看見距離考試還剩幾天；建立群組來記錄社團活動的重要事項，讓成員們都不會遺漏任何活動，透過群組中的投票功能決定社團中的大小事，不用擔心過於充實的生活導致重要的事情被遺忘；在「玩聚」的功能中發起行程邀約，使得大家能夠共襄盛舉；而且能夠在想要安排行程製造驚喜時，透過瀏覽對方行程的功能來確認對方有空，如此以來便不會遇到精心準備卻沒能傳達的尷尬窘境。當然的我們也相當注重隱私，可以針對特定的對象來關閉此功能，也可以將所有人的權限關閉，如此一來就可以無所顧忌地將比較私人的內容存在 My Day 不會因為想記錄秘密而沒有享受方便生活的權利。

My Day 適合身為學生的你，不管你是常常忘東忘西、是生活充實總能將行事曆排滿，又或是多彩多姿的生活想與朋友們分享，My Day 都將會是最適合的一款 App。

1-2 問題與機會

在市場上的行事曆軟體都是比較商業取向的，沒有專門為學生設計的行事曆，將行事曆結合課表是可以嘗試的新想法，因為學生市場很大，而且每年都會有新的學生，如何規劃好時間一直都是學生時代很重要的課題，面對這樣無可避免的問題，如果有一個便利的應用程式來輔助，是不是更能事半功倍呢？

▼表 1-2-1、SWOT 分析

Strengths 優勢	Weaknesses 劣勢
<ul style="list-style-type: none">1. 課程時間表，方便學生使用。2. 可在 iOS、Android 手機裝置使用。3. 促進人際關係。4. 共享行程方便做時間管理。5. 提升學習效率。	<ul style="list-style-type: none">1. 使用對象僅限於大專生。2. 功能較複雜，需要時間學習。
Opportunities 機會	Threats 威脅
<ul style="list-style-type: none">1. 未來與學校合作，提升使用率。2. 每年都有新的大專生。3. 年輕族群喜歡嘗試新的事物。	<ul style="list-style-type: none">1. 手機有內建行事曆，且市面上的行事曆種類眾多。2. 少子化，學生數量減少。

1-3 相關系統探討

▼表 1-3-1、相關系統比較

	My Day	Google 日曆	Time Tree	iOS 行事曆
時間軸	課表	一般時間軸	一般時間軸	一般時間軸
主題顏色	提供彩色主題	提供黑白主題	無	提供黑白主題
與他人共享行程	僅需邀請一次	需邀請多次	僅需邀請一次	需邀請多次
讀書計畫行程	有	無	無	無
查看他人行程	可	不可	不可	不可
支援作業系統	Android、iOS	Android、iOS	Android、iOS	iOS
使用對象	大專生	所有人	所有人	所有人

My Day 跟其他行事曆最不同的地方在於，我們的功能皆是為學生打造，主畫面就是課表結合行事曆，可讓學生們一目了然。

還有讀書計畫行程功能，可增加學習力及做好時間分配，並有群組功能可一次共享多個行程、讀書計畫、課表等，達到節省時間的效果。

第2章 系統目標與預期成果

2-1 系統目標

1. 邀請大家一起規劃校園生活，從此之後不必再獨自一人。
2. 與朋友建立讀書計畫，更有效率地利用時間。
3. 隨時發起行程邀約，來場說走就走的旅行。

2-2 預期成果

1. 一目了然的行程：

透過課表與行程的結合以及行程顏色分類，一眼即能得知行程規劃。

2. 共學：

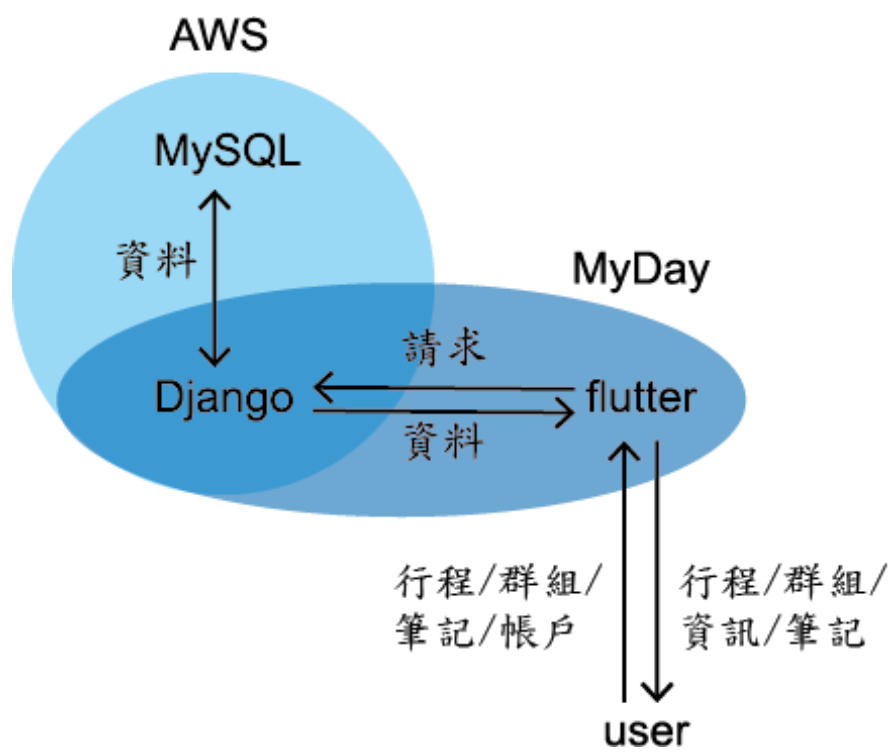
提供共同讀書計畫及共享筆記，與朋友共同學習，一起進步。

3. 共享行程：

透過群組可以新增行程，並且和成員們共享。

第3章 系統規格

3-1 系統架構



▲圖 3-1-1、系統架構

3-2 系統軟、硬體需求與技術平台

▼表 3-2-1、軟硬體需求

手機硬體需求		
作業系統	 	Android/iOS
網路需求		
網路	   	Wifi/乙太網路/3G/4G/5G

3-3 開發標準與使用工具

▼表 3-3-1、開發環境

系統開發環境		
作業系統		Windows/MacOS
撰寫工具		Visual Studio Code、Xcode、Android Studio
程式語言		Dart、Python
程式開發工具		
前端		Flutter
後端		Django
資料庫		MySQL Workbench
推播		Firebase
介面及插圖繪製工具		
介面		Adobe XD
插圖		Adobe Illustrator
文件及美化工具		
文件		Microsoft Word
圖表		Draw.io、Xmind
資料庫關聯		dbdiagram.io
簡報		Microsoft PowerPoint

專案管理及版本控制工具		
應用程式		fork
版本控制		git

第4章 專案時程與組織分工

4-1 專案時程



▲圖 4-1-1、專案時程

4-2 專案組織與分工

▼表 4-2-1、分工表

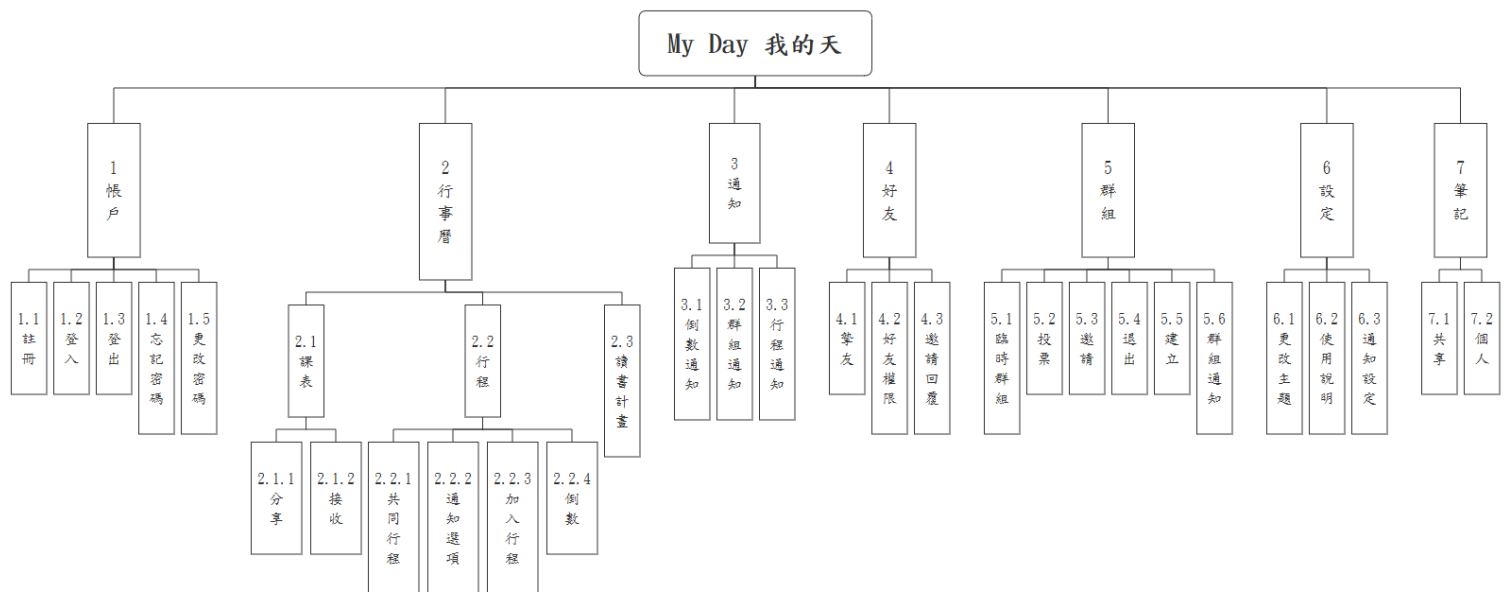
組員		10656004	10656003	10656009	10656025	10656035
工作項目		張恩瑤	吳東陽	林哲卉	林欣儀	吳乙萱
規劃	主題制定	●	●	●	●	●
	工作分配	●				
	進度制定	●				
	功能設計	●	●	●	●	●
	API 規劃	●	●	●	●	●

資料庫		資料庫 關聯設計	●	◎		◎	
		資料庫 建置	●	◎		◎	
程式 撰寫	前端	首頁	●				
		帳號					●
		設定					●
		行程	●				
		讀書計畫			●		●
		筆記					●
		玩聚			●		
		群組			●		
		課表	◎	●	◎		
程式 撰寫	後端	帳號		●			
		好友				●	
		群組				●	
		筆記	●				
		行程	●				
		讀書計畫				●	
		玩聚				●	

		課表	◎	●			
		個人資料		●			
		投票	●				
美工		LOGO 設計	●		●		
		介面設計			●		●
		簡報製作	●	●	●	●	●
		名牌製作			●		
		海報製作	●		●		
影片	簡介	編劇	●	◎	◎		
	系統	剪輯	●				
	專題	編劇	●	◎	◎	◎	◎
	心得	剪輯	●				
文件撰寫		編輯		●		●	
		校正	●	●	●	●	●
		整合				●	
		API 規格 書製作	●	●			

第5章 需求模型

5-1 功能分解圖（Functional decomposition diagram）



▲圖 5-1-1、功能分解圖(FDD)

5-2 需求清單

▼表 5-2-1、功能性需求

功能名稱	功能敘述
1.帳戶	1. 使用者使用信箱，密碼進行註冊登入。 2. 密碼忘記時會發送驗證碼到使用者信箱，也可更改密碼及登出。
2.行事曆	1. 接收他人的課表或分享自己的課表。 2. 增加共同行程，行程是否通知或重複通知，是否倒數行程結束日期。 3. 建立個人或群組讀書計畫，確認是否完成計畫。

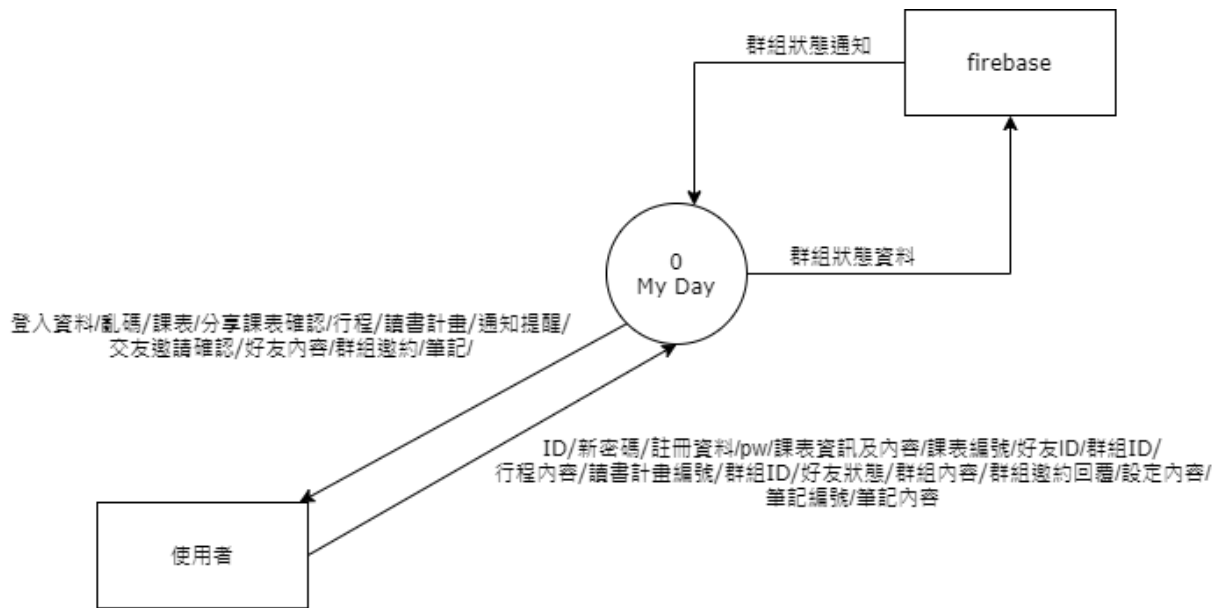
3.通知	1. 通知使用者倒數時間、群組事項、行程提醒。
4.好友	1. 設定好友權限、邀請好友、設置摯友。
5.群組	1. 分成群組和玩聚，內有投票功能。 2. 可邀請好友進入，也可自己退出群組，不能強制成員退出。 3. 設定通知是否開啟。
6.設定	1. 更改主題顏色。 2. 開啟或關閉倒數、群組、行程通知或玩聚邀請。
7.筆記	1. 建立個人筆記或共享給其他人使用。

非功能性需求

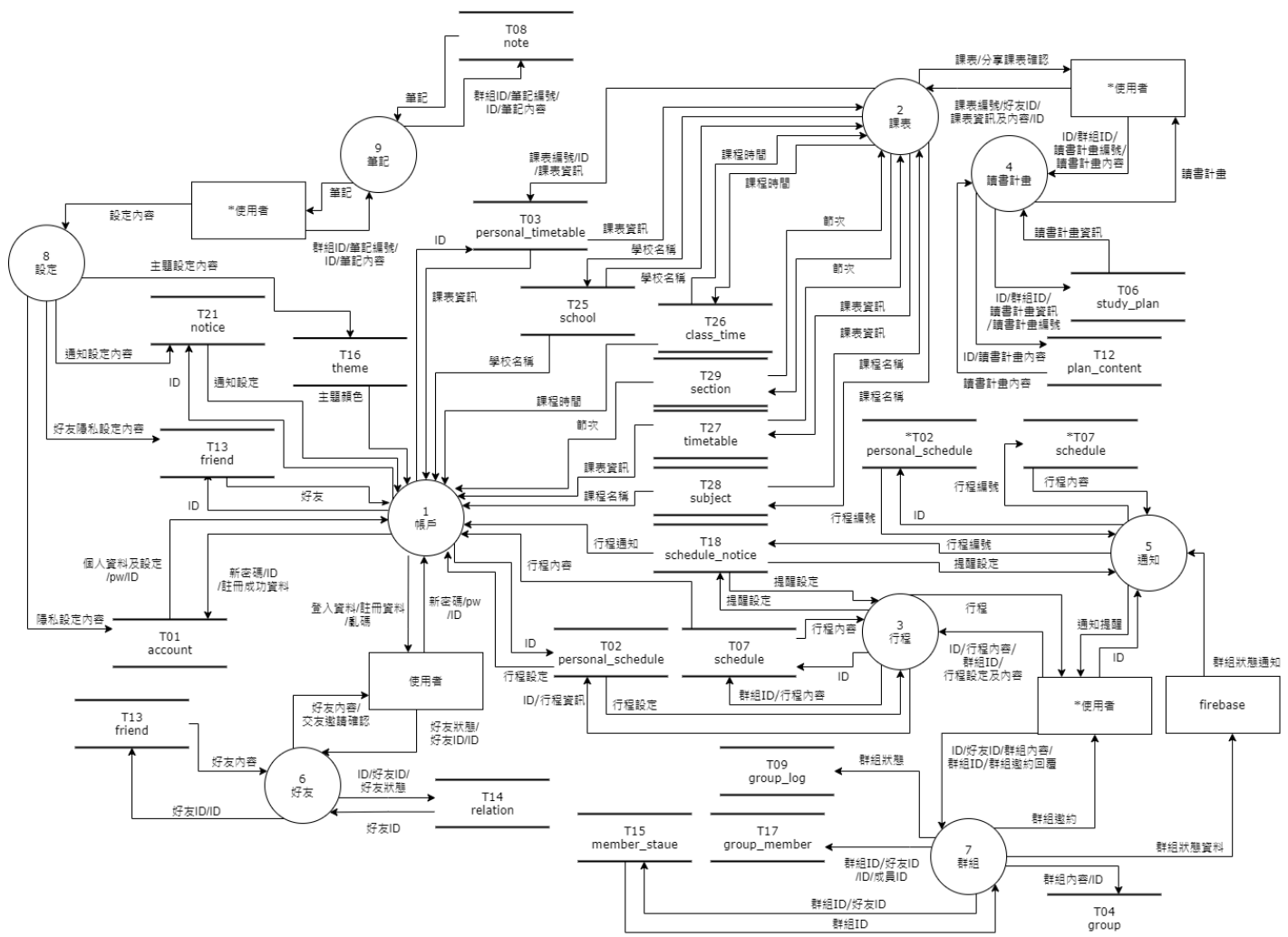
1. 使用者需登入才能使用系統所有功能

第6章 程序模型

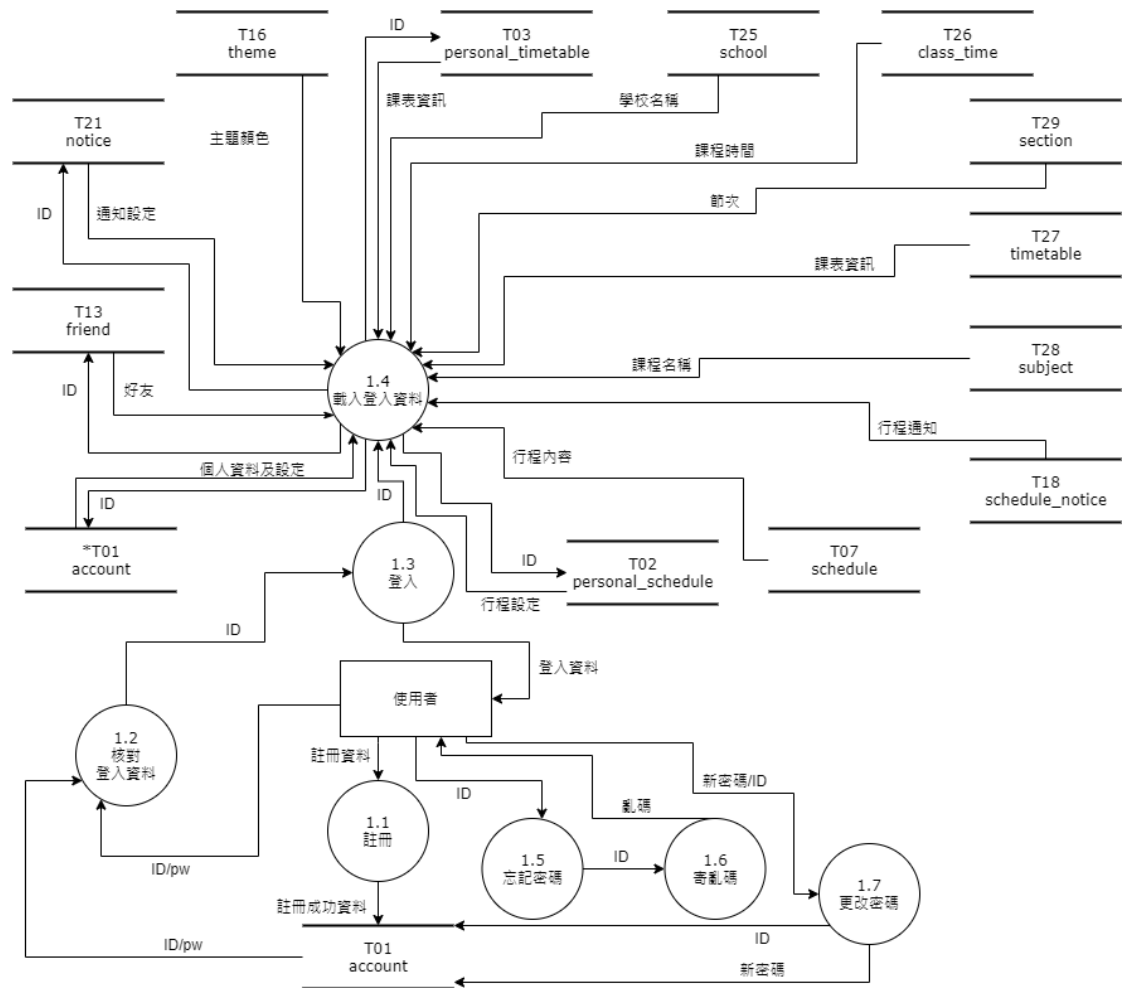
6-1 資料流程圖(Data flow diagram)



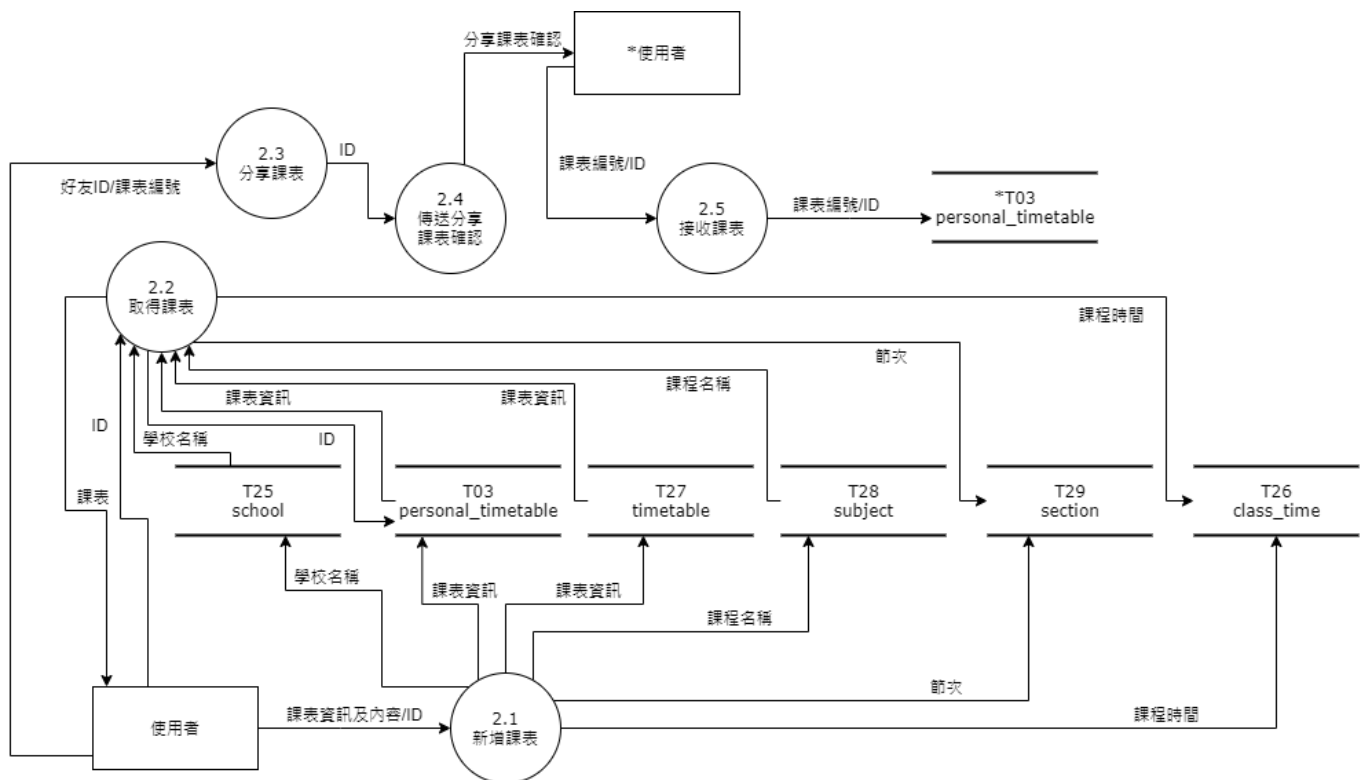
▲圖 6-1-1、系統環境圖



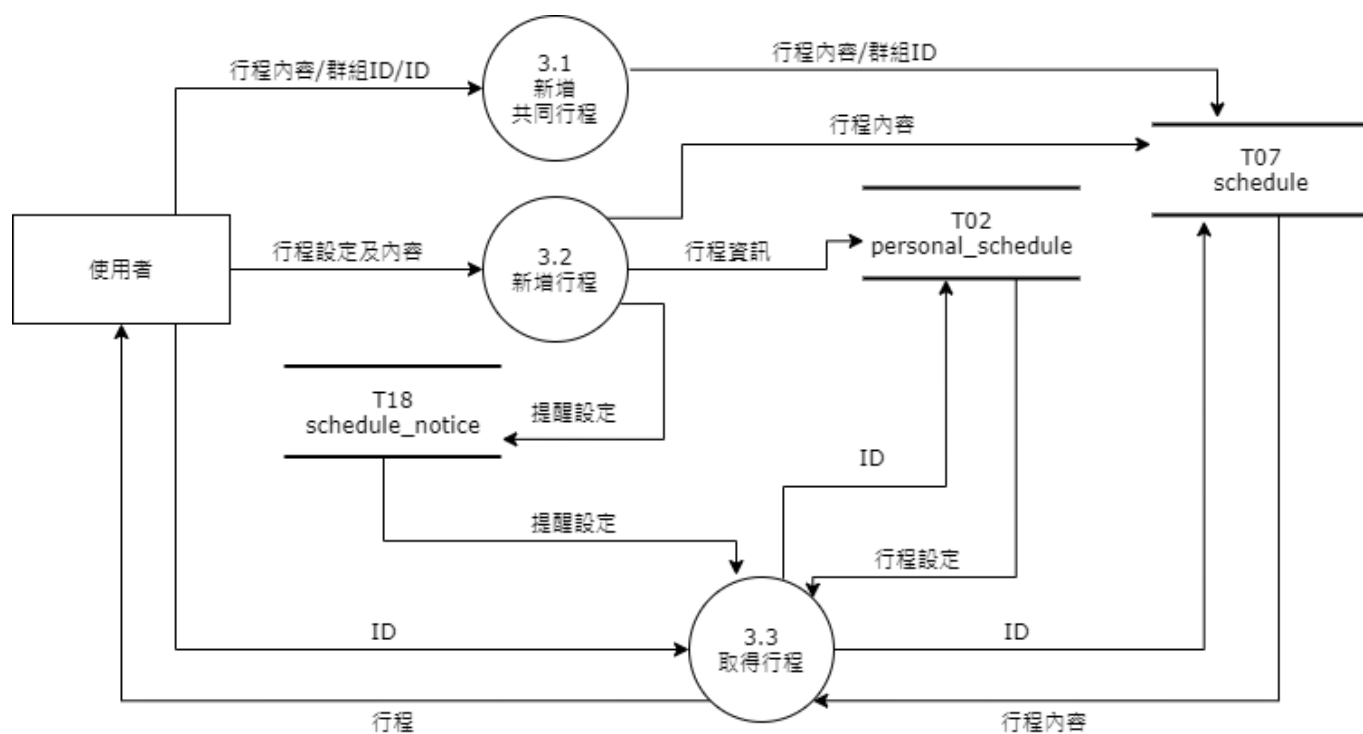
▲圖 6-1-2、圖 0



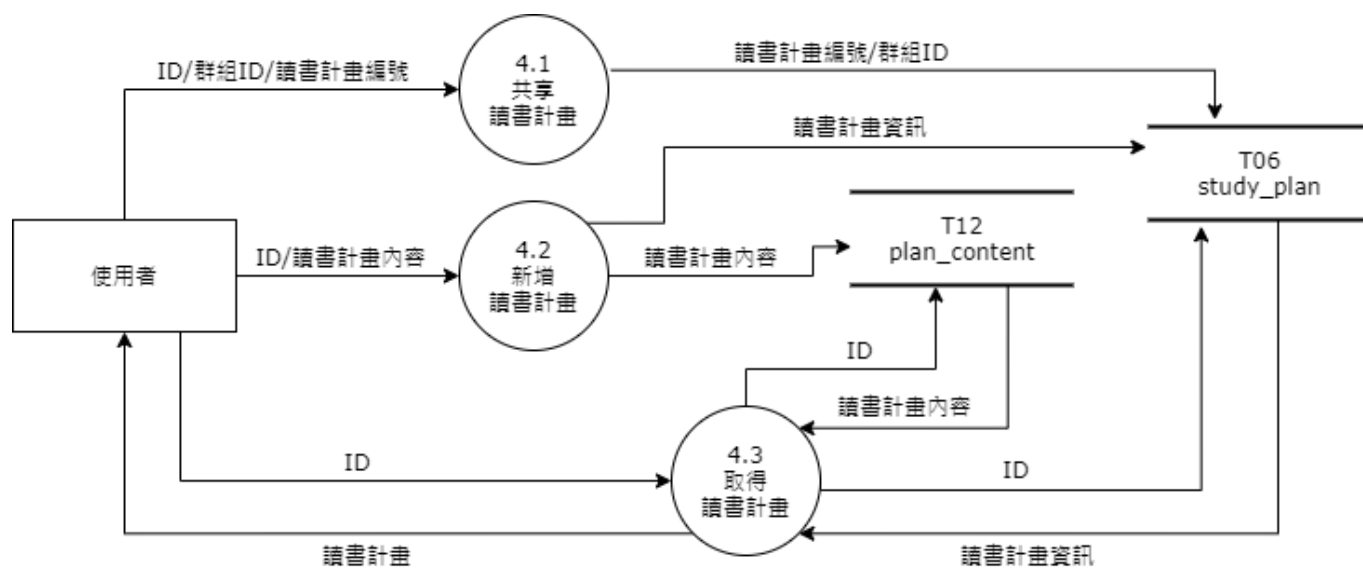
▲圖 6-1-3、圖 1



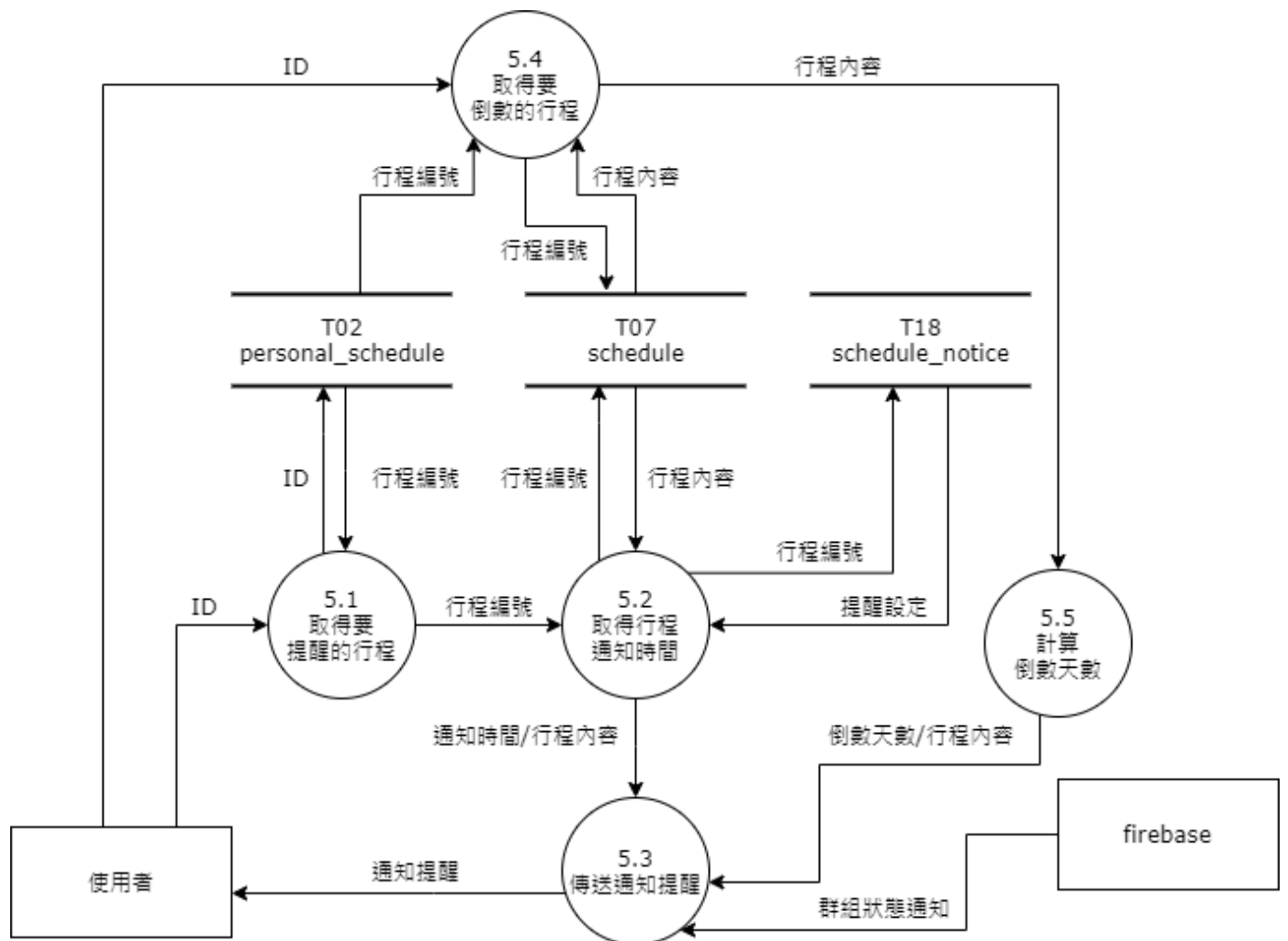
▲圖 6-1-4、圖 2



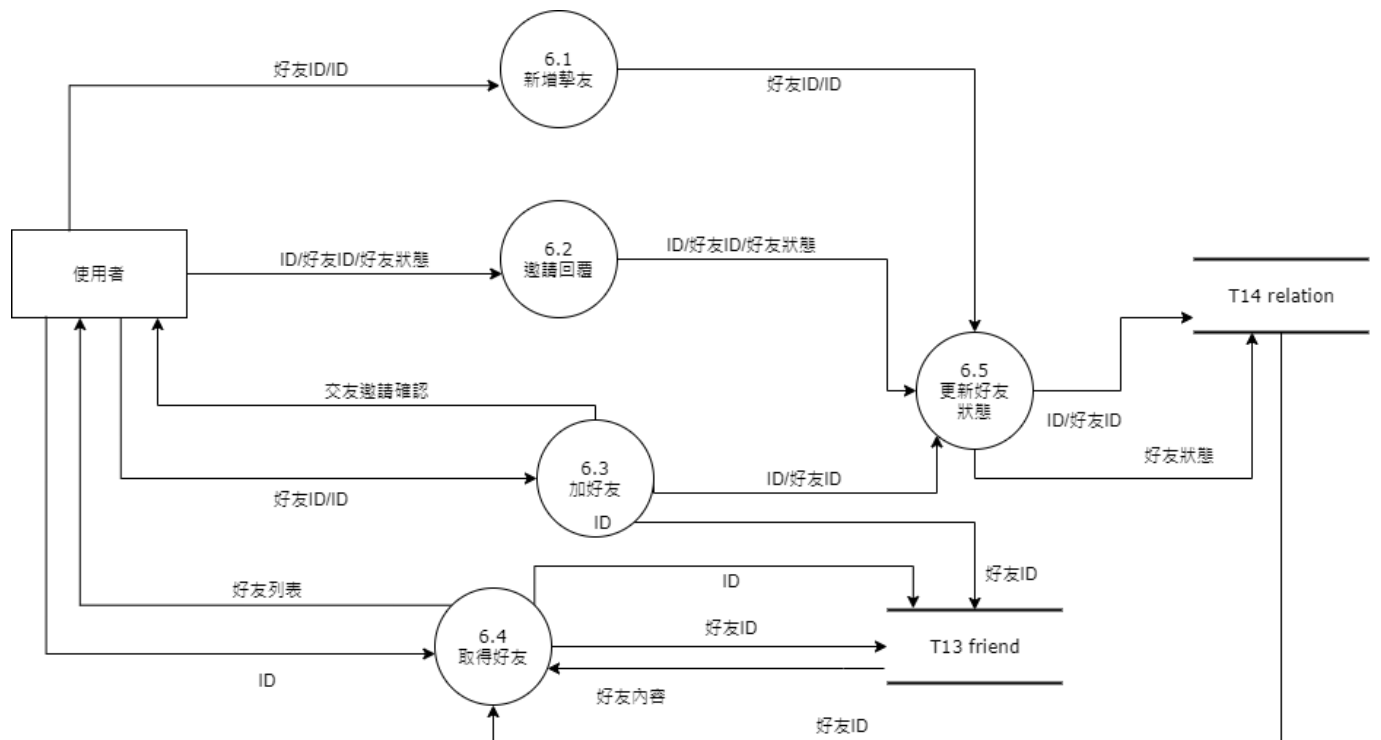
▲圖 6-1-5、圖 3



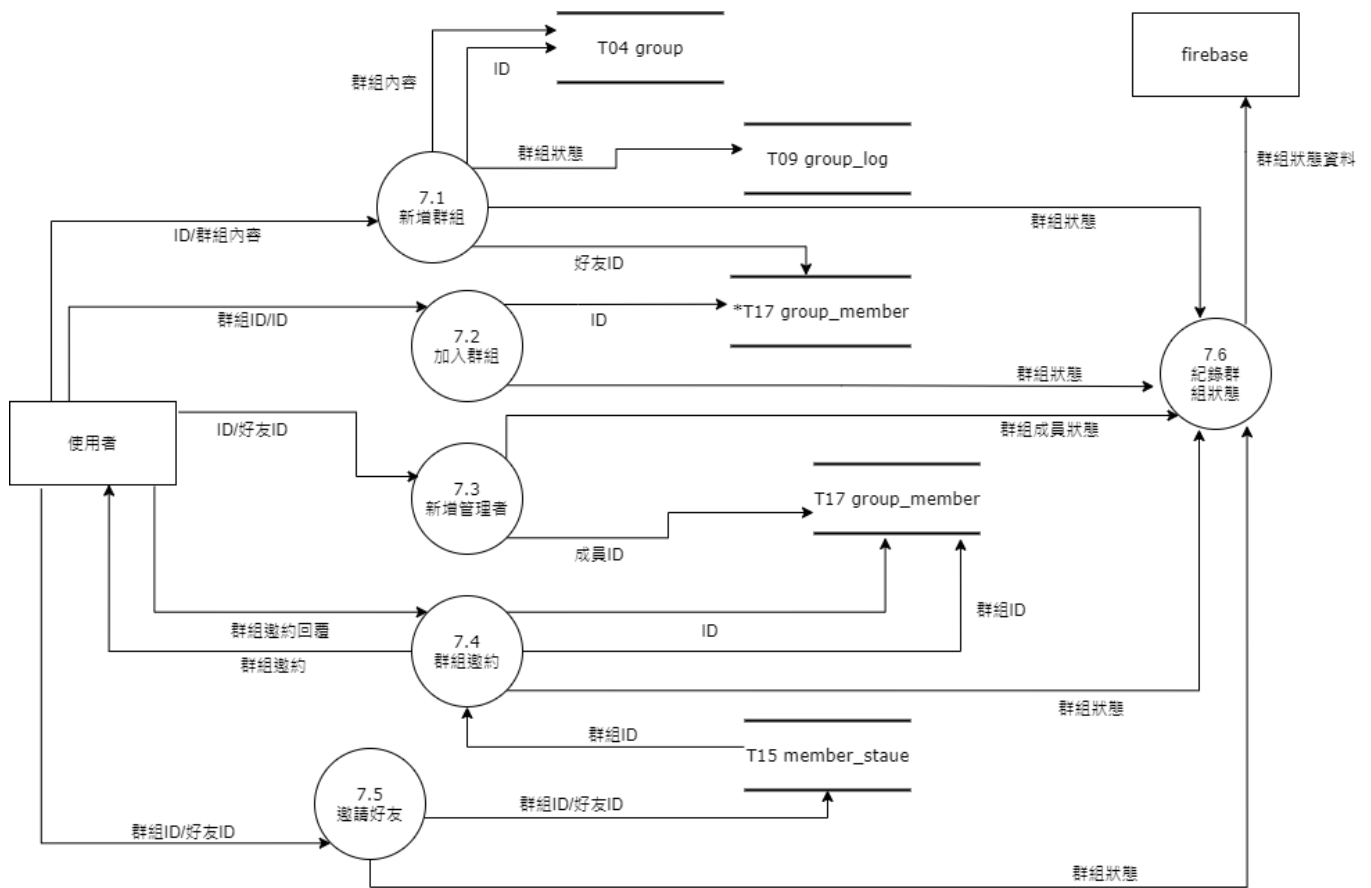
▲圖 6-1-6、圖 4



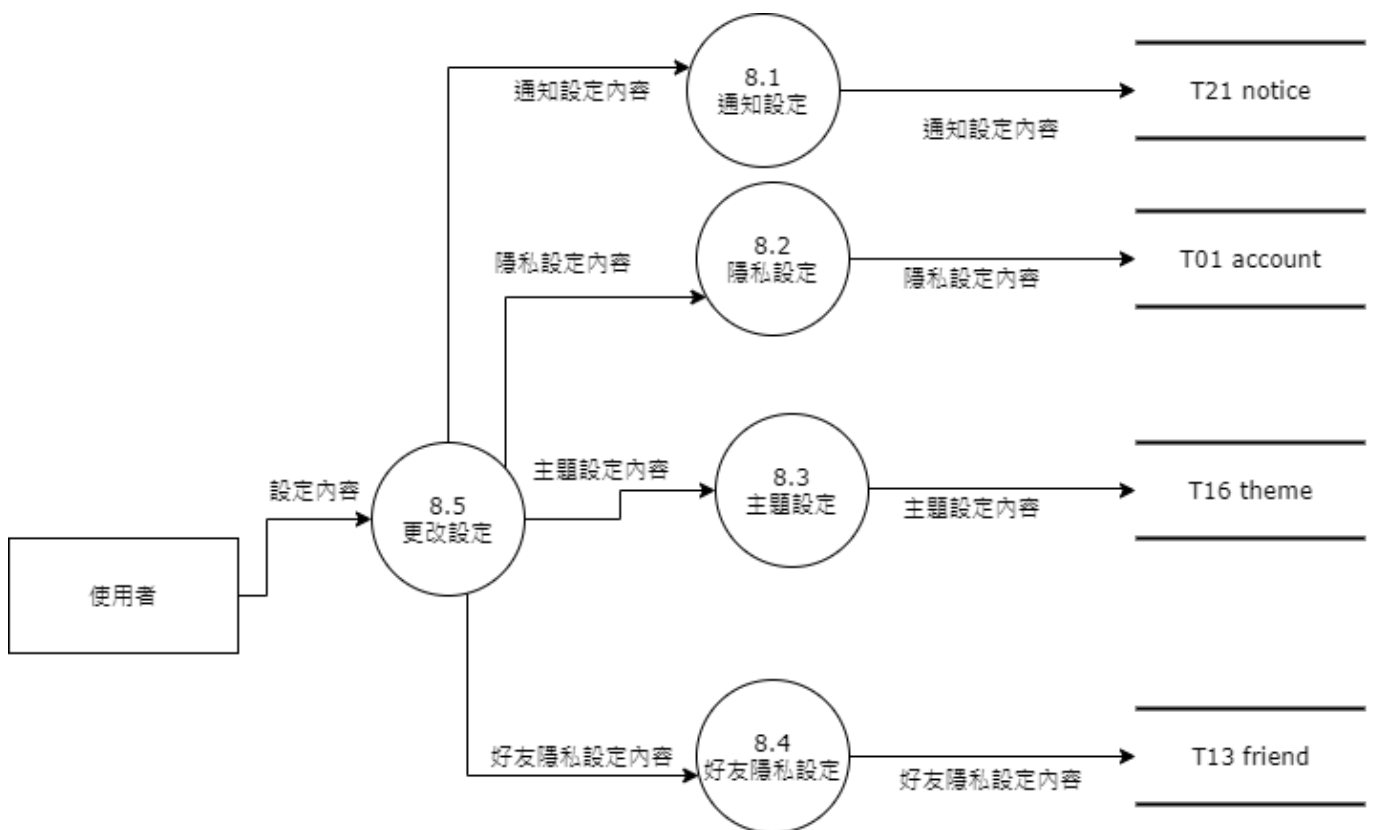
▲圖 6-1-7、圖 5



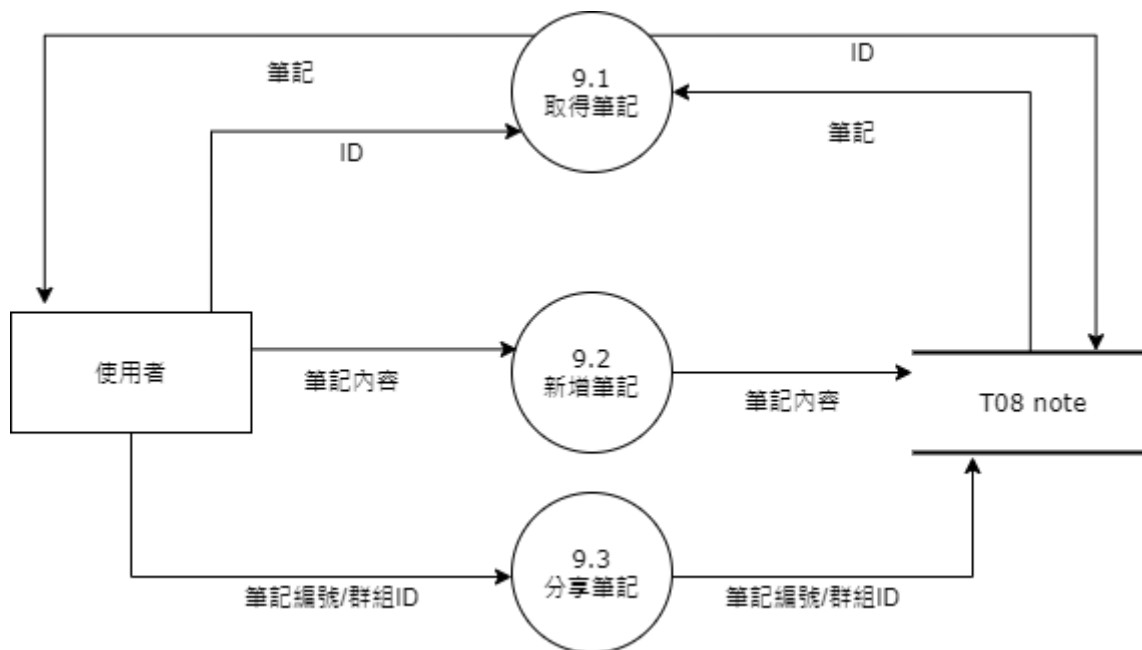
▲圖 6-1-8、圖 6



▲圖 6-1-9、圖 7



▲圖 6-1-10、圖 8



▲圖 6-1-11、圖 9

6-2 程序規格書(Process specification)

▼表 6-2-1、程序 1

編號	1	功能名稱	帳戶
操作說明	使用者管理個人帳號		
輸入值	帳號、密碼、新密碼		
輸出值	登入資料、註冊資料、亂碼		
編號	1.1	功能名稱	註冊
操作說明	使用者輸入基本的註冊資料		
輸入值	註冊資料		
輸出值	註冊成功資料		

編號	1.2	功能名稱	核對登入資料
操作說明	核對資料庫的帳號密碼是否正確		
輸入值	帳號、密碼		
輸出值	帳號		
編號	1.3	功能名稱	登入
操作說明	正確的話，進去載入登入資料；不正確，回傳帳密錯誤		
輸入值	帳號		
輸出值	帳號、登入資料		
編號	1.4	功能名稱	載入登入資料
操作說明	從資料庫拉資料給使用者		
輸入值	帳號、課表資訊、學校名稱、課程時間、節次、課表資訊、課程名稱、行程通知、行程內容、個人資料及設定、好友、通知設定、主題顏色		
輸出值	帳號、登入資料		
編號	1.5	功能名稱	忘記密碼
操作說明	使用者忘記密碼確認是否有這個帳號		
輸入值	帳號		

輸出值	帳號		
編號	1.6	功能名稱	寄亂碼
操作說明	將使用者密碼改為亂碼傳到使用者信箱		
輸入值	帳號		
輸出值	亂碼		
編號	1.7	功能名稱	更改密碼
操作說明	使用者輸入新密碼兩次，即可更改密碼		
輸入值	帳號、新密碼		
輸出值	帳號、新密碼		

▼表 6-2-2、程序 2

編號	2	功能名稱	課表
操作說明	使用者管理有關課表的項目		
輸入值	課表編號、帳號、好友帳號、課表資訊及內容		
輸出值	課表、分享課表確認		
編號	2.1	功能名稱	新增課表
操作說明	輸入課表資料，建立該學期課表		
輸入值	帳號、課表資訊及內容		

輸出值	學校名稱、課表資訊、課程名稱、節次		
編號	2.2	功能名稱	取得課表
操作說明	使用者點選該學年學期，即可顯示課表		
輸入值	帳號		
輸出值	課表		
編號	2.3	功能名稱	分享課表
操作說明	使用者將某課表分享的功能		
輸入值	好友帳號、課表編號		
輸出值	帳號		
編號	2.4	功能名稱	傳送分享課表確認
操作說明	使用者確認分享該課表		
輸入值	帳號		
輸出值	分享課表確認		
編號	2.5	功能名稱	接收課表
操作說明	好友接受使用者傳的課表		
輸入值	帳號、課表編號		

輸出值	帳號、課表編號
-----	---------

▼表 6-2-3、程序 3

編號	3	功能名稱	行程
操作說明	使用者管理有關行程的項目		
輸入值	帳號、行程內容、群組編號、行程設定及內容		
輸出值	行程		
編號	3.1	功能名稱	新增共同行程
操作說明	使用者在群組輸入標題、開始時間、結束時間、提醒、倒數、地點、備註來新增共同行程		
輸入值	帳號、群組編號、行程內容		
輸出值	群組編號、行程內容		
編號	3.2	功能名稱	新增行程
操作說明	使用者輸入標題、開始時間、結束時間、提醒、倒數、地點、備註來新稱行程		
輸入值	帳號、行程設定及內容		
輸出值	行程資訊、行程內容、提醒設定		
編號	3.3	功能名稱	取得行程

操作說明	使用者點選該行程
輸入值	帳號
輸出值	行程資料

▼表 6-2-4、程序 4

編號	4	功能名稱	讀書計畫
操作說明	使用者管理有關讀書計畫的項目		
輸入值	帳號、群組編號、讀書計畫編號、讀書計畫內容		
輸出值	讀書計畫		
編號	4.1	功能名稱	共享讀書計畫
操作說明	使用者分享讀書計畫到某群組		
輸入值	帳號、群組編號、讀書計畫編號		
輸出值	群組編號、讀書計畫編號		
編號	4.2	功能名稱	新增讀書計畫
操作說明	使用者輸入日期、開始時間、結束時間、科目來新增讀書計畫		
輸入值	帳號、讀書計畫內容		
輸出值	讀書計畫資訊、讀書計畫內容		

編號	4.3	功能名稱	取得讀書計畫
操作說明	使用者點選該讀書計畫，取得內容		
輸入值	帳號		
輸出值	讀書計畫資料		

▼表 6-2-5、程序 5

編號	5	功能名稱	通知
操作說明	使用者管理有關通知的項目		
輸入值	帳號		
輸出值	通知提醒		
編號	5.1	功能名稱	取得要提醒的行程
操作說明	使用者點選該行程提醒功能來設定		
輸入值	帳號		
輸出值	行程編號		
編號	5.2	功能名稱	取得行程通知時間
操作說明	使用者點選該行程通知時間功能來設定		
輸入值	行程編號、行程內容、提醒設定		
輸出值	行程內容、通知時間		

編號	5.3	功能名稱	傳送通知提醒
操作說明	使用者輸入要通知的時間及倒數設定回傳資料庫		
輸入值	行程內容、通知時間、倒數天數、群組狀態通知		
輸出值	通知提醒		
編號	5.4	功能名稱	取得要倒數的行程
操作說明	使用者點選該行程倒數功能來設定		
輸入值	帳號、行程編號、行程內容		
輸出值	行程內容		
編號	5.5	功能名稱	計算倒數天數
操作說明	使用者設定該行程要不要倒數		
輸入值	行程內容		
輸出值	行程內容、倒數天數		

▼表 6-2-6、程序 6

編號	6	功能名稱	好友
操作說明	使用者管理有關好友的項目		
輸入值	帳號、好友帳號、好友狀態		
輸出值	好友內容、交友邀請確認		

編號	6.1	功能名稱	新增摯友
操作說明	使用者點選該好友成為摯友		
輸入值	帳號、好友帳號		
輸出值	帳號、好友帳號		
編號	6.2	功能名稱	邀請回覆
操作說明	好友回覆使用者是否成為好友		
輸入值	帳號、好友帳號、好友狀態		
輸出值	帳號、好友帳號、好友狀態		
編號	6.3	功能名稱	加好友
操作說明	使用者輸入好友帳號來加好友		
輸入值	帳號、好友帳號、交友邀請確認		
輸出值	帳號、好友帳號		
編號	6.4	功能名稱	取得好友
操作說明	使用者點選好友功能取得所有好友		
輸入值	帳號、好友列表		
輸出值	帳號、好友帳戶		

編號	6.5	功能名稱	更新好友狀態
操作說明	使用者加好友、好友邀請回覆這兩個動作，都會更新好友狀態		
輸入值	帳號、好友帳號、好友狀態		
輸出值	帳號、好友帳號、好友狀態		

▼表 6-2-7、程序 7

編號	7	功能名稱	群組
操作說明	使用者管理有關群組的項目		
輸入值	帳號、好友帳號、群組內容、群組編號、群組邀約回覆		
輸出值	群組邀約		
編號	7.1	功能名稱	新增群組
操作說明	使用者輸入群組名稱、好友帳號來新增群組		
輸入值	帳號、群組內容		
輸出值	帳號、群組內容、群組狀態		
編號	7.2	功能名稱	加入群組
操作說明	被邀請的使用者回覆加入該群組		
輸入值	帳號、群組編號		
輸出值	帳號、群組狀態		

編號	7.3	功能名稱	新增管理者
操作說明	群組管理員，新增某群組成員也成為管理者		
輸入值	帳號、成員帳號		
輸出值	群組成員狀態		
編號	7.4	功能名稱	群組邀約
操作說明	使用者		
輸入值	群組邀約回覆、群組編號		
輸出值	帳號、群組編號、群組邀約		
編號	7.5	功能名稱	邀請好友
操作說明	使用者邀請某好友進入該群組		
輸入值	帳號、好友帳號		
輸出值	好友帳號、群組狀態		
編號	7.6	功能名稱	紀錄群組狀態
操作說明	群組的行程、投票功能都被記錄起來		
輸入值	群組狀態		
輸出值	群組狀態資料		

▼表 6-2-8、程序 8

編號	8	功能名稱	設定
操作說明	使用者管理有關設定的項目		
輸入值	設定內容		
輸出值	通知設定內容、隱私設定內容、主題設定內容、好友隱私設定內容		
編號	8.1	功能名稱	通知設定
操作說明	使用者設定所有的通知		
輸入值	通知設定內容		
輸出值	通知設定內容		
編號	8.2	功能名稱	隱私設定
操作說明	使用者設定定位、公開課表功能		
輸入值	隱私設定內容		
輸出值	隱私設定內容		
編號	8.3	功能名稱	主題設定
操作說明	使用者設定想要的主題顏色		
輸入值	主題設定內容		

輸出值	主題設定內容		
編號	8.4	功能名稱	好友隱私設定
操作說明	使用者設定公開課表、臨時群組邀請功能		
輸入值	好友隱私設定內容		
輸出值	好友隱私設定內容		
編號	8.5	功能名稱	更改設定
操作說明	使用者進入設定來更改設定		
輸入值	設定內容		
輸出值	通知設定內容、隱私設定內容、主題設定內容、好友隱私設定內容		

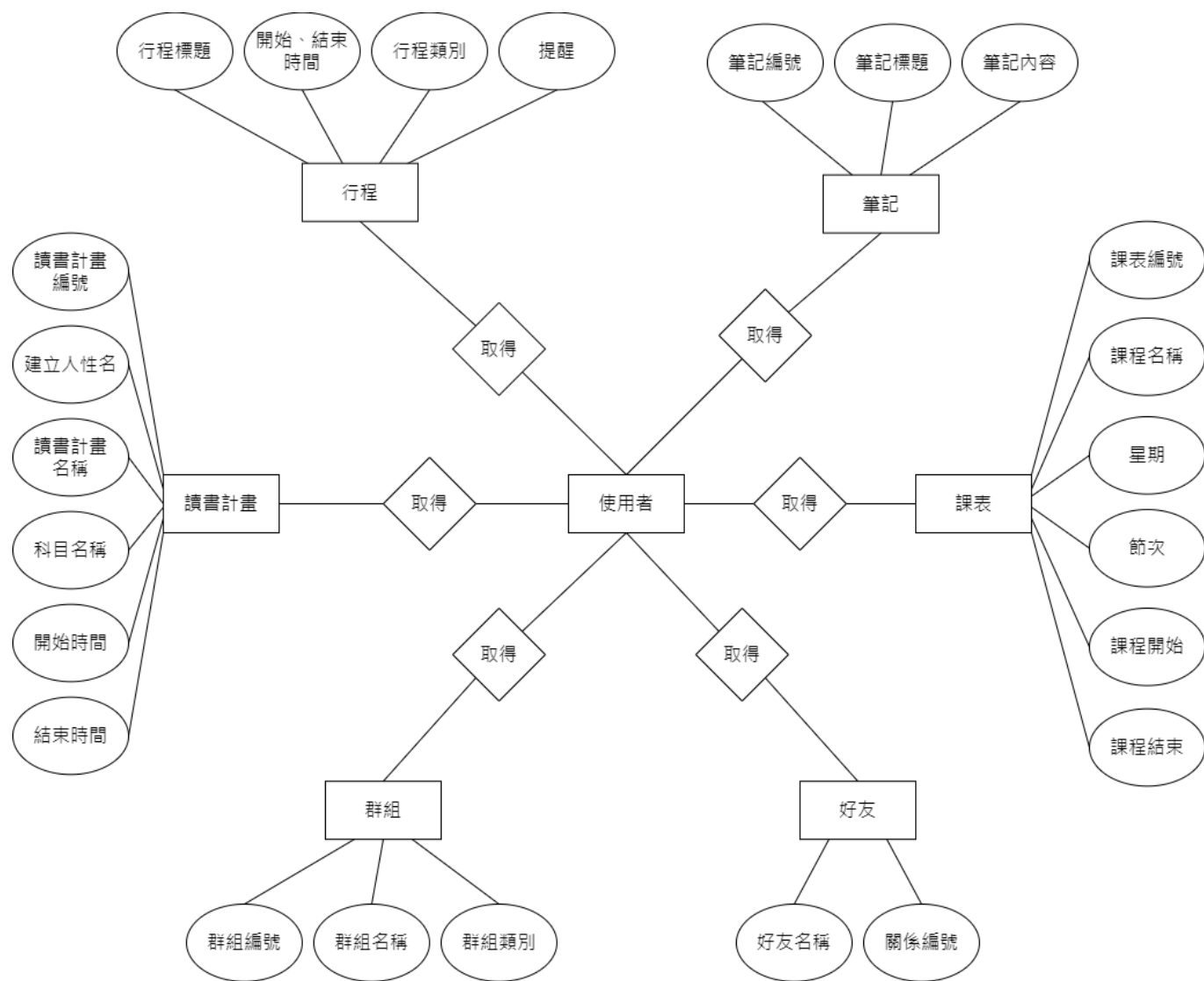
▼表 6-2-9、程序 9

編號	9	功能名稱	筆記
操作說明	使用者管理有關筆記的項目		
輸入值	帳號、筆記編號、群組編號、筆記內容		
輸出值	筆記		
編號	9.1	功能名稱	取得筆記
操作說明	使用者點選該筆記，取得筆記內容		

輸入值	帳號、筆記編號		
輸出值	筆記		
編號	9.2	功能名稱	新增筆記
操作說明	使用者輸入分類、標題、內容來新增筆記		
輸入值	帳號、筆記內容		
輸出值	筆記		
編號	9.3	功能名稱	分享筆記
操作說明	使用者將某筆記分享到群組給群組成員看		
輸入值	帳號、筆記編號		
輸出值	群組編號、筆記編號		

第7章 資料模型

7-1 實體關聯圖(Entity relationship diagram)



▲圖 7-1-1、實體關聯圖

7-2 資料字典(Data dictionary)

▼表 7-2-1、字典編號

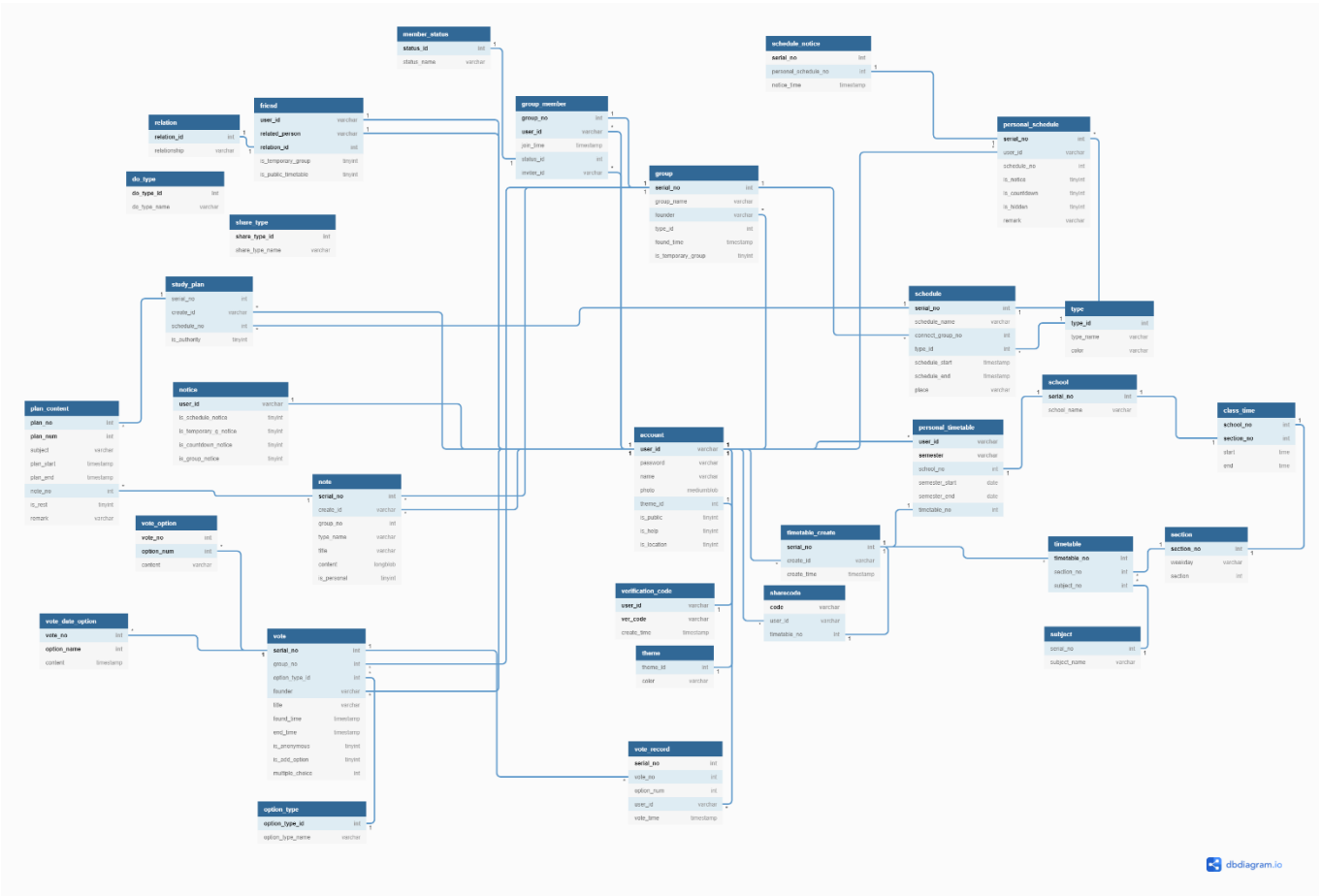
資料表編號	資料表名稱	資料表中文名稱
T01	account	帳號
T02	class_time	上課時間

T03	do_type	行為分類
T04	friend	好友
T05	group	群組
T06	group_member	群組成員
T07	member_status	成員身分
T08	note	筆記
T09	notice	通知
T10	option_type	項目類別
T11	personal_schedule	個人行程
T12	personal_timetable	個人課表
T13	plan_content	讀書計畫內容
T14	relation	關係
T15	schedule	行程
T16	schedule_notice	行程通知
T17	school	學校
T18	section	節次
T19	share_type	分享類別
T20	sharecode	分享碼
T21	study_plan	讀書計畫
T22	subject	科目

T23	theme	主題
T24	timetable	課表
T25	timetable_create	建立課表
T26	type	類別
T27	verification_code	驗證碼
T28	vote	投票
T29	vote_date_option	投票選項時間
T30	vote_option	投票選項
T31	vote_record	投票紀錄

第8章 資料庫設計

8-1 資料庫關聯圖



▲圖 8-1-1、資料庫關聯圖

8-2 表格及其 Meta data

▼表 8-2-1、T1 Meta data

T1 account 行程				
欄位名稱	欄位中文名稱	資料型態	資料長度	主索引
user_id	使用者編號	VARCHAR	255	V
password	密碼	VARCHAR	255	
name	姓名	VARCHAR	100	

photo	照片	MEDIUMBLOB	3	
theme_id	主題編號	INT	4	
is_public	公開	TINYINT	1	
is_help	小幫手	TINYINT	1	
is_location	定位	TINYINT	1	

▼表 8-2-2、T2 Meta data

T2 class_time 上課時間				
欄位名稱	欄位中文名稱	資料型態	資料長度	主索引
school_no	學校編號	INT	4	V
section_no	節次編號	INT	4	V
start	開始	TIME	3	
end	結束	TIME	3	

▼表 8-2-3、T3 Meta data

T3 do_type 行為分類				
欄位名稱	欄位中文名稱	資料型態	資料長度	主索引
do_type_id	行為編號	INT	4	V
do_type_name	行為名稱	VARCHAR	10	

▼表 8-2-4、T4 Meta data

T4 friend 好友				
--------------	--	--	--	--

欄位名稱	欄位中文名稱	資料型態	資料長度	主索引
user_id	使用者編號	VARCHAR	255	V
related_person	關係人	VARCHAR	255	V
relation_id	關係編號	INT	4	
is_temporary_group	臨時群組邀請	TINYINT	1	
is_public_timetable	是否公開課表	TINYINT	1	

▼ 表 8-2-5、T5 Meta data

T5 group 群組				
欄位名稱	欄位中文名稱	資料型態	資料長度	主索引
serial_no	流水號	INT	4	V
group_name	群組名稱	VARCHAR	255	
founder	邀請人	VARCHAR	255	
type_id	狀態	INT	4	
found_time	邀請時間	TIMESTAMP	4	
is_temporary_group	是否為 臨時群組	TINYINT	1	

▼ 表 8-2-6、T6 Meta data

T6 group_member 群組成員				
欄位名稱	欄位中文名稱	資料型態	資料長度	主索引

group_no	流水號	INT	4	V
user_id	使用者編號	VARCHAR	255	V
join_time	加入時間	TIMESTAMP	4	
status_id	身分	INT	4	
inviter_id	邀請人	VARCHAR	255	

▼表 8-2-7、T7 Meta data

T7 member_status 成員身分				
欄位名稱	欄位中文名稱	資料型態	資料長度	主索引
status_id	流水號	INT	4	V
status_name	身分名稱	VARCHAR	10	

▼表 8-2-8、T8 Meta data

T8 note 筆記				
欄位名稱	欄位中文名稱	資料型態	資料長度	主索引
serial_no	流水號	INT	4	V
create_id	建立者編號	VARCHAR	255	
group_no	群組編號	INT	4	
type_name	類別名稱	VARCHAR	255	
title	標題	VARCHAR	255	
content	內容	LONGBLOB	1	

is_personal	是否為個人	TINYINT	1	
-------------	-------	---------	---	--

▼ 表 8-2-9、T9 Meta data

T9 notice 通知				
欄位名稱	欄位中文名稱	資料型態	資料長度	主索引
user_id	使用者編號	VARCHAR	255	V
is_schedule_notice	是否行程通知	TINYINT	1	
is_temporary_group_notice	是否 臨時群組通知	TINYINT	1	
is_countdown_notice	是否倒數通知	TINYINT	1	
is_group_notice	是否群組通知	TINYINT	1	

▼ 表 8-2-10、T10 Meta data

T10 option_type 行為類別				
欄位名稱	欄位中文名稱	資料型態	資料長度	主索引
option_type_id	行為類別編號	INT	4	V
option_type_name	行為類別名稱	VARCHAR	10	

▼ 表 8-2-11、T11 Meta data

T11 personal_schedule 個人行程				
欄位名稱	欄位中文名稱	資料型態	資料長度	主索引
serial_no	流水號	INT	4	V

user_id	行為類別名稱	VARCHAR	255	
schedule_no	行程編號	INT	4	
is_notice	是否提醒	TINYINT	1	
is_countdown	是否倒數	TINYINT	1	
is_hidden	是否隱藏	TINYINT	1	
remark	備註	VARCHAR	255	

▼ 表 8-2-12、T12 Meta data

T12 personal_timetable 個人課表				
欄位名稱	欄位中文名稱	資料型態	資料長度	主索引
user_id	使用者編號	VARCHAR	255	V
semester	學期	VARCHAR	6	V
school_no	學校編號	INT	4	
semester_start	學期開始日期	DATE	3	
semester_end	學期結束日期	DATE	3	
timetable_no	課表編號	INT	4	

▼ 表 8-2-13、T13 Meta data

T13 plan_content 讀書計畫內容				
欄位名稱	欄位中文名稱	資料型態	資料長度	主索引
plan_no	讀書計畫	INT	4	V

	編號			
plan_num	個別編號	INT	4	V
subject	科目	VARCHAR	20	
plan_start	讀書計畫 開始時間	TIMESTAMP	4	
plan_end	讀書計畫 結束時間	TIMESTAMP	4	
note_no	筆記編號	INT	4	
is_rest	是否休息	TINYINT	1	
remark	備註	VARCHAR	255	

▼表 8-2-14、T14 Meta data

T14 relation 關係				
欄位名稱	欄位中文名稱	資料型態	資料長度	主索引
relation_id	關係編號	INT	4	V
relationship	關係	VARCHAR	10	

▼表 8-2-15、T15 Meta Data

T15 schedule 行程				
欄位名稱	欄位中文名稱	資料型態	資料長度	主索引
serial_no	流水號	INT	4	V

schedule_name	行程名稱	VARCHAR	255	
connect_group_no	群組編號	INT	4	
type_id	行程類型	INT	4	
schedule_start	開始時間	TIMESTAMP	4	
schedule_end	結束時間	TIMESTAMP	4	
place	地點	VARCHAR	255	

▼表 8-2-16、T16 Meta Data

T16 schedule_notice 行程通知				
欄位名稱	欄位中文名稱	資料型態	資料長度	主索引
serial_no	流水號	INT	4	V
personal_schedule_no	個人行程編號	INT	4	
notice_time	通知時間	TIMESTAMP	4	

▼表 8-2-17、T17 Meta Data

T17 school 學校				
欄位名稱	欄位中文名稱	資料型態	資料長度	主索引
serial_no	流水號	INT	4	V
school_name	學校名稱	VARCHAR	50	

▼表 8-2-18、T18 Meta Data

T18 section 節次				
----------------	--	--	--	--

欄位名稱	欄位中文名稱	資料型態	資料長度	主索引
section_no	節次編號	INT	4	V
weekday	星期	VARCHAR	255	
section	節次	INT	4	

▼表 8-2-19、T19 Meta Data

T19 share_type 分享類別				
欄位名稱	欄位中文名稱	資料型態	資料長度	主索引
share_type_id	分享類別編號	INT	4	V
share_type_name	類型名稱	VARCHAR	10	

▼表 8-2-20、T20 Meta Data

T20 sharecode 分享碼				
欄位名稱	欄位中文名稱	資料型態	資料長度	主索引
code	分享碼	VARCHAR	8	V
user_id	使用者編號	VARCHAR	255	
timetable_no	課表編號	INT	4	

▼表 8-2-21、T21 Meta Data

T21 study_plan 讀書計畫				
欄位名稱	欄位中文名稱	資料型態	資料長度	主索引
serial_no	流水號	INT	4	V

create_id	建立者編號	VARCHAR	255	
schedule_no	行程編號	INT	4	
is_authority	權限開啟	TINYINT	1	

▼表 8-2-22、T22 Meta Data

T22 subject 科目				
欄位名稱	欄位中文名稱	資料型態	資料長度	主索引
serial_no	流水號	INT	4	V
subject_name	科目名稱	VARCHAR	20	

▼表 8-2-23、T23 Meta Data

T23 theme 主題				
欄位名稱	欄位中文名稱	資料型態	資料長度	主索引
theme_id	主題編號	INT	4	V
color	顏色	VARCHAR	6	

▼表 8-2-24、T24 Meta Data

T24 timetable 課表				
欄位名稱	欄位中文名稱	資料型態	資料長度	主索引
timetable_no	課表編號	INT	4	V
section_no	節次編號	INT	255	V
subject_no	課程編號	INT	4	

▼表 8-2-25、T25 Meta Data

T25 timetable_create 建立課表				
欄位名稱	欄位中文名稱	資料型態	資料長度	主索引
serial_no	流水號	INT	4	V
create_id	使用者編號	VARCHAR	255	
create_time	建立時間	TIMESTAMP	4	

▼表 8-2-26、T26 Meta Data

T026 type 類別				
欄位名稱	欄位中文名稱	資料型態	資料長度	主索引
type_id	類別編號	INT	4	V
type_name	類別名稱	VARCHAR	10	
color	顏色	VARCHAR	7	

▼表 8-2-27、T27 Meta Data

T27 verification_code 驗證碼				
欄位名稱	欄位中文名稱	資料型態	資料長度	主索引
user_id	使用者編號	VARCHAR	255	V
ver_code	驗證碼	VARCHAR	6	V
create_time	建立時間	TIMESTAMP	4	

▼表 8-2-28、T28 Meta Data

T28 vote 投票				
欄位名稱	欄位中文名稱	資料型態	資料長度	主索引
serial_no	流水號	INT	4	V
group_no	行程名稱	INT	4	
option_type_id	連接的群組編號	INT	4	
founder	行程類型	VARCHAR	255	
title	開始時間	VARCHAR	255	
found_time	結束時間	TIMESTAMP	4	
end_time	地點	TIMESTAMP	4	
is_anonymous	是否匿名	TINYINT	1	
is_add_option	是否新增選項	TINYINT	1	
multiple_choice	選項數量	INT	4	

▼表 8-2-29、T29 Meta Data

T29 vote_date_option 投票選項時間				
欄位名稱	欄位中文名稱	資料型態	資料長度	主索引
vote_no	投票編號	INT	4	V
option_num	選項編號	INT	4	V
content	時間內容	TIMESTAMP	4	

▼表 8-2-30、T30 Meta Data

T30 vote_option 投票選項				
欄位名稱	欄位中文名稱	資料型態	資料長度	主索引
vote_no	投票編號	INT	4	V
option_num	選項編號	INT	4	V
content	內容	VARCHAR	255	

▼表 8-2-31、T31 Meta Data

T31 vote_record 投票紀錄				
欄位名稱	欄位中文名稱	資料型態	資料長度	主索引
serial_no	流水號	INT	4	V
vote_no	投票編號	INT	4	
option_num	選項編號	INT	255	
user_id	使用者編號	VARCHAR	255	
vote_time	投票時間	TIMESTAMP	4	

第9章 程式規格

9-1 軟體架構與程式清單

▼表 9-1-1、前端程式清單

編號	檔案名稱	功能名稱
1	account/login.dart	登入
2	account/login_fail.dart	登入失敗
3	account/register.dart	註冊
4	account/register_fail.dart	註冊
5	account/change_password.dart	更改密碼
6	account/register_forget_password.dart	忘記密碼
7	common_note/common_note_detail_page.dart	共同筆記內容
8	common_note /common_note_list_page.dart	共同筆記列表
9	common_note /share_note_page.dart	分享筆記
10	common_schedule /common_schedule_form.dart	共同行程表單
11	common_schedule /common_schedule_list.dart	共同行程列表
12	common_schedule /edit_common_schedule_page.dart	編輯共同行程
13	common_studyplan/common_studyplan_list_page.dart	共同讀書計畫列表
14	common_studyplan/customer_check_box_studyplan.dart	讀書計畫客製化核取 方塊
15	common_studyplan/select_note_page.dart	選擇筆記

16	common_studyplan/share_studyplan_page.dart	分享讀書計畫
17	friend/bestfriend_add.dart	增加摯友
18	friend/bestfriend.dart	摯友
19	friend/friend_fail.dart	加好友錯誤
20	friend/friends_add.dart	新增好友
21	friend/friends_invitation.dart	好友邀請
22	friend/friends_privacy_settings.dart	好友隱私設定
23	friend/friends.dart	好友
24	group/customer_check_box.dart	群組客製化核取方塊
25	group/group_create_page.dart	建立群組
26	group/group_detail_page.dart	群組細節
27	group/group_information_page.dart	群組資訊
28	group/group_invite_page.dart	群組邀請
29	group/group_join_page.dart	加入群組
30	group/group_list_page.dart	群組列表
31	group/group_manager_page.dart	群組管理者
32	group/group_member_page.dart	群組成員
33	group/group_setting_page.dart	群組設定
34	home/home_schedule/schedule_button.dart	行程按鈕
35	home/home_schedule/schedule_table_function.dart	行程表方法

36	home/home_schedule/schedule_table.dart	行程表
37	home/home_page_functions.dart	主頁方法
38	home/home_page.dart	主頁
39	home/home_popup_menu.dart	主頁選單彈出框
40	home/home_schedule.dart	主頁行程
41	home/home_Update.dart	主頁更新
42	schedule/create_schedule.dart	建立行程
43	schedule/edit_schedule.dart	編輯行程
44	schedule/remind_item.dart	提醒項目
45	schedule/schedule_dialog.dart	行程對話框
46	schedule/schedule_form.dart	行程表單
47	setting/change_password_personal.dart	個人更改密碼
48	setting/notice.dart	通知
49	setting/open_class_schedule.dart	公開行程
50	setting/personal_information.dart	個人資訊
51	setting/play_together_invite.dart	玩聚邀請
52	setting/privacy.dart	隱私
53	setting/settings.dart	設定
54	setting/theme.dart	主題
55	study/edit_studyplan_page.dart	編輯讀書計畫

56	study/note_choose.dart	選擇筆記
57	study/note_detail_page.dart	筆記細項
58	study/note_fail.dart	筆記錯誤
59	study/notes_add.dart	建立筆記
60	study/notes_edit.dart	編輯筆記
61	study/notes.dart	筆記
62	study/studyplan_detail_page.dart	讀書計畫細項
63	study/studyplan_form.dart	讀書計畫表單
64	study/studyplan_list_page.dart	讀書計畫列表
65	temporary_group/temporary_group_create_page.dart	建立臨時群組
66	temporary_group/temporary_group_invite_page.dart	臨時群組邀請
67	temporary_group/temporary_group_list_page.dart	臨時群組列表
68	timetable/template/timetable_template.dart	課表模板
69	timetable/timetable_action_list.dart	課表動作列表
70	timetable/timetable_choose_past.dart	選擇歷年課表
71	timetable/timetable_choose_share.dart	選擇分享課表
72	timetable/timetable_create_cancel.dart	取消建立課表
73	timetable/timetable_create_popup_menu.dart	建立課表選單彈出框
74	timetable/timetable_create.dart	建立課表
75	timetable/timetable_edit_form.dart	編輯課表表單

76	timetable/timetable_edit_popup.dart	編輯課表選單彈出框
77	timetable/timetable_edit.dart	編輯課表
78	timetable/timetable_form.dart	課表表單
79	timetable/timetable_import_past.dart	匯入歷年課表
80	timetable/timetable_import_share.dart	匯入分享課表
81	timetable/timetable_improt_choose.dart	匯入選擇課表
82	timetable/timetable_information.dart	課表資訊
83	timetable/timetable_receive_preview.dart	預覽接收課表
84	timetable/timetable_receive.dart	接收課表
85	timetable/timetable_share_friend.dart	好友分享課表
86	timetable/timetable_share_group.dart	群組分享課表
87	timetable/timetable_share.dart	分享課表
88	vote/vote_create_page.dart	建立投票
89	vote/vote_edit_page.dart	編輯投票
90	vote/vote_end_detail_page.dart	投票結果
91	vote/vote_list_page.dart	投票列表
92	vote/vote_page.dart	投票
93	vote/vote_setting_page.dart	投票設定
94	home.dart	首頁
95	my_day_icon.dart	圖示

96	models/friend/best_friend_list_model.dart	摯友列表
97	models/friend/friend_list_model.dart	好友列表
98	models/friend/make-friend-invite-list_model.dart	好友邀請列表
99	models/group/common_schedule_list_model.dart	共同行程列表
100	models/group/get_common_schedule_model.dart	取得共同行程
101	models/group/get_group_model.dart	取得群組
102	models/group/group_invite_friend_list_model.dart	群組邀請好友列表
103	models/group/group_invite_list_model.dart	群組邀請列表
104	models/group/group_list_model.dart	群組列表
105	models/group/group_log_model.dart	群組紀錄
106	models/group/group_member_list_model.dart	群組成員列表
107	models/note/get_note_model.dart	取得筆記
108	models/note/note_list_model.dart	筆記列表
109	models/note/share_note_list_model.dart	分享筆記列表
110	models/schedule/group_studyplan_list_model.dart	群組讀書計畫列表
111	models/schedule/schedule_list_model.dart	行程列表
112	models/schedule/schedule_model.dart	行程
113	models/setting/theme_model.dart	主題
114	models/studyplan/common_studyplan_list_model.dart	共同讀書計畫列表
115	models/studyplan/personal_share_studyplan_model.dart	個人分享

116	models/studyplan/studyplan_list_model.dart	讀書計畫列表
117	models/studyplan/studyplan_model.dart	讀書計畫
118	models/temporary_group/ get_temporary_group_invitet_model.dart	取得臨時群組邀請
119	models/temporary_group/ temporary_group_list_model.dart	臨時群組列表
120	models/timetable/main_timetable_list_model.dart	主要課表列表
121	models/timetable/section_time_model.dart	節次
122	models/timetable/sharecode_model.dart	分享碼
123	models/timetable/timetable_list_model.dart	課表列表
124	models/vote/get_vote_model.dart	取得投票
125	models/vote/vote_end_list_model.dart	投票結果
126	models/vote/vote_list_model.dart	投票列表
127	public/account_request/change_pw.dart	更改密碼
128	public/account_request/forget_pw.dart	忘記密碼
129	public/account_request/login.dart	登入
130	public/account_request/register.dart	註冊
131	public/account_request/send_code.dart	驗證碼
132	public/friend_request/add_best.dart	加摯友
133	public/friend_request/add-friend-reply.dart	加好友回覆
134	public/friend_request/add.dart	加好友
135	public/friend_request/best_friend_list.dart	摯友列表

136	public/friend_request/delete_best.dart	刪除摯友
137	public/friend_request/delete.dart	刪除好友
138	public/friend_request/friend_list.dart	好友列表
139	public/friend_request/make-friend-invite-list.dart	好友邀請列表
140	public/group_request/create_group.dart	建立群組
141	public/group_request/edit_group.dart	編輯群組
142	public/group_request/get_log.dart	取得群組紀錄
143	public/group_request/get.dart	取得群組
144	public/group_request/group_list.dart	群組列表
145	public/group_request/invite_friend_list.dart	邀請朋友列表
146	public/group_request/invite_friend.dart	邀請朋友
147	public/group_request/invite_list.dart	邀請列表
148	public/group_request/member_list.dart	群組成員列表
149	public/group_request/member_status.dart	群組身分
150	public/group_request/quit_group.dart	離開群組
151	public/group_request/setting_manager.dart	設定管理者
152	public/note_request/cancel_share.dart	取消分享筆記
153	public/note_request/create_note.dart	建立筆記
154	public/note_request/delete.dart	刪除筆記
155	public/note_request/edit.dart	編輯筆記

156	public/note_request/get_group_list.dart	取得群組列表
157	public/note_request/get_list.dart	取得筆記列表
158	public/note_request/get.dart	取得筆記
159	public/note_request/share.dart	分享筆記
160	public/schedule_request/common_list.dart	共同行程列表
161	public/schedule_request/create_common.dart	建立共同行程
162	public/schedule_request/create_new.dart	建立行程
163	public/schedule_request/delete.dart	刪除行程
164	public/schedule_request/edit.dart	編輯行程
165	public/schedule_request/get_common.dart	取得共同行程
166	public/schedule_request/get_list.dart	取得行程列表
167	public/schedule_request/get.dart	取得行程
168	public/setting/friend_privacy.dart	好友隱私
169	public/setting/get_theme.dart	取得主題
170	public/setting/notice_temporary.dart	臨時群組通知
171	public/setting/notice.dart	通知
172	public/setting/privacy_location.dart	定位隱私
173	public/setting/privacy_timetable.dart	課表隱私
174	public/setting/theme.dart	主題
175	public/studyplan_request/cancel_sharing.dart	取消分享讀書計畫

176	public/studyplan_request/create_studyplan.dart	建立讀書計畫
177	public/studyplan_request/delete.dart	刪除讀書計畫
178	public/studyplan_request/edit_studyplan.dart	編輯讀書計畫
179	public/studyplan_request/get.dart	取得讀書計畫
180	public/studyplan_request/group_list.dart	群組讀書計畫列表
181	public/studyplan_request/one_group_list.dart	單一群組列表
182	public/studyplan_request/personal_list.dart	個人讀書計畫列表
183	public/studyplan_request/personal_share_list.dart	分享個人讀書計畫列表
184	public/studyplan_request/sharing.dart	分享讀書計畫
185	public/temporary_group_request/create_group.dart	建立臨時群組
186	public/temporary_group_request/get_invite.dart	取得臨時群組邀請
187	public/temporary_group_request/invite_list.dart	臨時群組邀請列表
188	public/temporary_group_request/temporary_list.dart	臨時群組列表
189	public/timetable_request/get_section_time.dart	取得節次
190	public/timetable_request/get_sharecode.dart	取得分享碼
191	public/timetable_request/get_timetable_list.dart	取得課表列表
192	public/timetable_request/main_timetable_list.dart	課表列表
193	public/vote_request/add_items.dart	新增投票項目
194	public/vote_request/create_new.dart	新增投票

195	public/vote_request/delete.dart	刪除投票
196	public/vote_request/edit.dart	編輯投票
197	public/vote_request/get_end_list.dart	取得投票結果列表
198	public/vote_request/get_list.dart	取得投票列表
199	public/vote_request/get.dart	取得投票
200	public/vote_request/vote.dart	投票
201	public/alert.dart	警示
202	public/convert.dart	轉換
203	public/request.dart	請求
204	public/sizing.dart	調整畫面大小
205	public/time_range.dart	計算時間區間
206	public/toast.dart	彈出提示
207	public type_color.dart	顏色類別

▼表 9-1-2、API 程式及後端程式清單

編號	檔案名稱	功能名稱
1	accounts/views.py	帳號相關
2	friend/views.py	好友相關
3	group/views.py	群組相關
4	group/functions.py	群組與臨時群組共用方法
5	group/serializers.py	群組序列化
6	note/views.py	筆記相關
7	schedule/views.py	行程相關
8	setting/views.py	使用者設定相關
9	studyplan/views.py	讀書計畫相關
10	studyplan/serializers.py	讀書計畫序列化
11	temporary_group/views.py	臨時群組相關
12	temporary_group/ serializers.py	臨時群組序列化
13	timetable/views.py	課表相關
14	user_profile/views.py	使用者檔案相關
15	vote/views.py	投票相關
16	core/urls.py	API 導向的 URL
17	api/log_func.py	使用者使用狀況
18	api/response.py	系統回覆訊息

19	api/urls.py	查看使用情況的 URL
20	api/views.py	查看使用情況

9-2 程式規格描述

▼表 9-2-1、Dart 部分程式碼描述

程式名稱	schedule_button.dart
目的	行程按鈕
<pre>// dart // flutter import 'package:flutter/material.dart'; // therd // my day import 'package:My_Day_app/home/home_page_functions.dart'; import 'package:My_Day_app/home/home_schedule.dart'; import 'package:My_Day_app/home/home_schedule/schedule_table_function.dart'; import 'package:My_Day_app/models/schedule/schedule_list_model.dart'; import 'package:My_Day_app/public/convert.dart'; import 'package:My_Day_app/public/sizing.dart'; import 'package:My_Day_app/public/type_color.dart'; import 'package:My_Day_app/public/time_range.dart'; class ScheduleButton { Color _weekDayColor = Color(0xFF6F6F6F); double _allDayRowHeight; BuildContext context; ScheduleGetList scheduleList; DateTime monday; List timeLineList; Sizing _sizing; double _fullHeight; List<Widget> _allDay = []; List<Widget> _schedules = [];</pre>	

```

double _position;

void _scheduleArrange() {
    DateTime _today;
    List<List> _scheduleCount = [
        for (var i = 0; i < timeLineList.length; i++)
            [for (var j = 0; j < 7; j++) 0]
    ]; // 計算一個時段的行程數量 // 時間區段數量*7 的陣列
    // [[0,0,0,0,0,0,0],[0,0,0,0,0,0,0],[0,0,0,0,0,0,0],...]

    for (int i = 0; i < 7; i++) {
        // 從本週一開始每天跑一次

        _today = monday.add(Duration(days: i));

        int _max = 0; // 檢查經過時段的行程數量

        List<Widget> _column = [];

        for (var s in scheduleList.schedule) {
            int _scheduleNumber = s.scheduleNum;
            String _scheduleTitle = s.title;
            DateTime _startTime = s.startTime;
            DateTime _endTime = s.endTime;
            int _scheduleType = s.typeId;
            _position = 0;

            TimeRange _startTimeRange = TimeRange(_startTime);
            TimeRange _endTimeRange = TimeRange(_endTime);

            if (_startTimeRange.notAfter(_today) &&
                _endTimeRange
                    .notBefore(_today.add(Duration(hours: 23, minutes: 59)))) {

                // 開始和結束都不在今天 表示是整天行程

                _column.add(Container(
                    height: _sizing.height(1),
                    child: TextButton(

```



```

        child: Container(),
        style: ButtonStyle(
            backgroundColor: MaterialStateProperty.all<Color>(
                typeColor(_scheduleType))),
        onPressed: () {},
    ),
));
} else {

    // 整天的行程不執行

    DateTime _thisStart;
    DateTime _thisEnd;
    double _scheduleHeight = 0;

    for (int j = 0; j < timeLineList.length; j++) {

        // 表格的每一個時間段檢查一遍

        bool isStart = false;

        String timeLineStart = timeLineList[j]['start'];
        String timeLineEnd = timeLineList[j]['end'];

        Duration _heightStart;
        Duration _heightEnd;

        // 用 _heightStart 和 _heightEnd 計算在這格中行程的長度

        // => 將每隔中行程的長度相加為行程總長

        if (timeLineStart == " || timeLineEnd == ") {
            timeLineStart = timeLineList[j - 1]['end'];
            timeLineEnd = timeLineList[j + 1]['start'];
        } // 如果「開始」和「結束」為空上一段的結束作為「開始」下一
        段的開始作為「結束」

        _thisStart = _today.add(ConvertString.toDuration(timeLineStart));
        _thisEnd = _today.add(ConvertString.toDuration(timeLineEnd));

        if (_thisStart.isBefore(_today

```

```

        .add(ConvertString.toDuration(timeLineList[0]['start']))) {
// 如果此段時間在今天表格開始之前 => 加一天表示明天的這個
時段

// 例如課表是 8:00 開始 如果出現 7:00 表示是明天的時間（在
表格最下方）=

    _thisStart = _thisStart.add(Duration(days: 1));
    _thisEnd = _thisEnd.add(Duration(days: 1));
}

if (_startTimeRange.inTime(_thisStart, _thisEnd)) {
// 在這格開始 => 開始時間=行程開始

isStart = true;
_heightStart = timeOfDateTime(_startTime);

_position += timeHeight(
    timeLineStart,
    ConvertDuration.toShortTime(timeOfDateTime(_startTime)),
    _fullHeight); // 如果在這格開始 => 上方的空間增加「這格
開始～行程開始」

} else if (_startTimeRange.notAfter(_thisStart)) {
// 在這格之前開始(不在這格之後) => 開始時間=這格開始

isStart = true;
_heightStart = timeOfDateTime(_thisStart);
} else if (_startTimeRange.notBefore(_thisStart)) {
// 在這格之後開始(不在這格之前) => 沒有開始時間

_heightStart = null;

if (!isStart) {
    _position += timeHeight(timeLineStart, timeLineEnd,
        _fullHeight); // 如果行程還沒開始 => 上方的空間增加

```

```

    }
}

if (_endTimeRange.inTime(_thisStart, _thisEnd)) {
    // 在這格結束 => 結束時間=行程結束

    _heightEnd = timeOfDateTime(_endTime);
} else if (_endTimeRange.notBefore(_thisEnd)) {
    // 在這格之後結束(不在這格之後) => 結束時間=這格結束

    _heightEnd = timeOfDateTime(_thisEnd);
} else if (_endTimeRange.notAfter(_thisEnd)) {
    // 在這格之前結束(不再這格之後) => 沒有開始時間

    _heightEnd = null;
}

if (_heightStart != null && _heightEnd != null) {
    _scheduleCount[j][i]++;

    _scheduleHeight += timeHeight(
        ConvertDuration.toShortTime(_heightStart),
        ConvertDuration.toShortTime(_heightEnd),
        _fullHeight);

    // 此行程在這天的總長累積
}

// 如果開始時間在結束時間之後
if (_scheduleCount[j][i] > _max) {
    _max = _scheduleCount[j][i];
}
}

double _left = (_sizing.width(100) / 8) * (i + 1);

if (_max <= 4) {
    _left += (_max - 1) * _sizing.width(3);

    _schedules.add(homeSchedule(context,
        top: _position,

```

```

        left: _left,
        height: _scheduleHeight,
        scheduleNum: _scheduleNumber,
        typeId: _scheduleType,
        count: _max));
    }
}
}

_allDay.add(Container(
    height: _allDayRowHeight,
    width: _sizing.width(100) / 8,
    child: ListView(children: _column),
));
}
}

ScheduleButton(
    {this.context, this.scheduleList, this.monday, this.timeLineList}) {
    _sizing = Sizing(context);
    _fullHeight = _sizing.height(100);
    _allDayRowHeight = _sizing.height(3);
    _allDay = [
        Container(
            color: _weekDayColor,
            height: _sizing.height(3),
            alignment: Alignment.center,
            child: Text('整日'),
        )
    ];
    _scheduleArrange();
}

getAllDay() => _allDay; // 整天行程按钮列表

getSchedules() => _schedules; // 行程按钮列表
}

```

```

// 1      2      3      4      5      6
//
//      ||
//      ||      ||
//      ||      ||
// -----
//      ||      ||
// ||      ||      ||
// ||      ||      ||
// -----
//      ||      ||
//      ||      ||      ||
//      ||      ||
//
// 在這格開始

// |_____ 在這格結束      1
// |_____ 在這格之後結束    2

// 在這格之前開始

// |_____ 在這格結束      3
// |_____ 在這格之前結束    4
// |_____ 在這格之後結束    5

// 在這格之後開始

// |_____ 在這格之後結束    6
//
// 對於這格而言只需判斷: 1235

```

程式名稱	schedule_dialog.dart
目的	行程對話框

```

import 'package:flutter/material.dart';

import 'package:animations/animations.dart';
import 'package:shared_preferences/shared_preferences.dart';

import 'package:My_Day_app/models/schedule/schedule_list_model.dart';
import 'package:My_Day_app/public/alert.dart';
import 'package:My_Day_app/public/convert.dart';
import 'package:My_Day_app/public/sizing.dart';
import 'package:My_Day_app/public/toast.dart';
import 'package:My_Day_app/public/type_color.dart';
import 'package:My_Day_app/public/schedule_request/delete.dart';
import 'package:My_Day_app/schedule/create_schedule.dart';
import 'package:My_Day_app/schedule/edit_schedule.dart';

Future<void> scheduleDialog(
  BuildContext context, DateTime date, ScheduleGetList scheduleList) {
  Sizing _sizing = Sizing(context);
  Color _color = Theme.of(context).primaryColor;

  double _fabDimension = 56.0;
  return showDialog(
    context: context,
    builder: (BuildContext context) {
      return Dialog(
        shape: RoundedRectangleBorder(
          borderRadius: BorderRadius.circular(_sizing.height(3))),
        child: Container(
          alignment: Alignment.centerLeft,
          height: _sizing.height(50),
          width: _sizing.width(70),
          child: Stack(children: [
            Column(
              children: [
                Padding(
                  padding: EdgeInsets.only(
                    top: _sizing.height(1.5), left: _sizing.width(8)),
                  child: Container(
                    alignment: Alignment.topLeft,
                    child: Column(
                      children: [

```

```

        Text(
          '${date.day} 日',
          style: TextStyle(
            color: _color, fontSize: _sizing.height(4)),
          textAlign: TextAlign.left,
        ),
        Text(
          ConvertInt.toWeekDay(date.weekday),
          style: TextStyle(
            color: _color,
            fontSize: _sizing.height(2),
            height: _sizing.height(0.1)),
        ),
      ],
    ),
  ),
  Divider(
    height: _sizing.height(2),
    indent: _sizing.width(5),
    endIndent: _sizing.width(5),
    color: Colors.grey,
    thickness: 0.5,
  ),
  Padding(
    padding: EdgeInsets.only(
      left: _sizing.width(5), right: _sizing.width(5)),
    child: Container(
      height: _sizing.height(35),
      child: ListView(
        children: listItems(context, date, scheduleList,
          _sizing.height(100), _sizing.width(100))),
      )
    ),
  ),
  Padding(
    padding: EdgeInsetsDirectional.only(end: 20.0, bottom: 16.0),
    child: Align(
      alignment: Alignment.bottomRight,
      child: OpenContainer(

```

```

        transitionType: ContainerTransitionType.fadeThrough,
        openBuilder: (BuildContext context, VoidCallback _) {
            return CreateSchedule();
        },
        closedElevation: 6.0,
        closedShape: RoundedRectangleBorder(
            borderRadius:
                BorderRadius.all(Radius.circular(_fabDimension / 2)),
        ),
        closedColor: _color,
        closedBuilder:
            (BuildContext context, VoidCallback openContainer) {
                return SizedBox(
                    height: _fabDimension,
                    width: _fabDimension,
                    child: Center(
                        child: Icon(
                            Icons.add,
                            color: Colors.white,
                        ),
                    ),
                );
            },
    ),
),
],
),
);
});
}

```

```

List<Widget> listItems(BuildContext context, DateTime date,
    ScheduleGetList scheduleList, double height, double width) {
    Sizing _sizing = Sizing(context);
    String _uid;
    _uidLoad() async {
        SharedPreferences prefs = await SharedPreferences.getInstance();
        _uid = prefs.getString('uid');
    }
}

```



```

_uidLoad();
List<Widget> itemList = [];
DateTime dayStart = DateTime.utc(date.year, date.month, date.day, 0, 0, 0);
DateTime dayEnd = dayStart.add(Duration(hours: 24));
for (var s in scheduleList.schedule) {
    DateTime start = s.startTime;
    DateTime end = s.endTime;

    String showStart;
    String showEnd;
    if (!start.isBefore(dayStart) &&
        !start.isAfter(dayEnd) &&
        !end.isBefore(dayStart) &&
        !end.isAfter(dayEnd)) {
        // 在今天結束之前開始 在今天開始之後結束

        showStart =
            '${start.hour.toString().padLeft(2, '0')}:${start.minute.toString().padLeft(2,
'0')}';
        showEnd =
            '${end.hour.toString().padLeft(2, '0')}:${end.minute.toString().padLeft(2,
'0')}';
    } else if (start.isBefore(dayStart) && !end.isBefore(dayStart)) {
        // 在今天之前開始 在今天開始之後結束

        showStart = '00:00';
        if (end.isAfter(dayEnd)) {
            // 在今天結束之後結束

            showEnd = '24:00';
        } else if (!end.isAfter(dayEnd)) {
            showEnd =
                '${end.hour.toString().padLeft(2, '0')}:${end.minute.toString().padLeft(2,
'0')}';
        }
    } else if (!start.isBefore(dayStart) &&
        !start.isAfter(dayEnd) &&
        end.isAfter(dayEnd)) {
        // 在今天開始之後開始 在今天之後結束

        showStart =

```

```

        '${start.hour.toString().padLeft(2, '0')}:${start.minute.toString().padLeft(2,
'0')}}';
        showEnd = '24:00';
    }

    if (showStart != null && showEnd != null) {
        itemList.add(Container(
            width: _sizing.width(6),
            child: TextButton(
                child: Row(
                    children: [
                        Container(
                            height: _sizing.height(2.5),
                            child: CircleAvatar(
                                radius: _sizing.height(2.5),
                                backgroundColor: typeColor(s.typeId),
                            ),
                        ),
                        Column(crossAxisAlignment: CrossAxisAlignment.start, children: [
                            Container(
                                width: _sizing.width(40),
                                child: Text(
                                    s.title,
                                    style: TextStyle(
                                        color: Colors.black,
                                        fontSize: _sizing.height(2.5),
                                        fontWeight: FontWeight.normal,
                                        fontStyle: FontStyle.normal),
                                    textAlign: TextAlign.left,
                                ),
                            ),
                            Text(
                                '$showStart-$showEnd',
                                style: TextStyle(
                                    color: Colors.grey,
                                    fontWeight: FontWeight.normal,
                                    fontStyle: FontStyle.normal,
                                ),
                            )
                        ])
                    )
                )
            )
        ],

```

```

    ),
    onPressed: () => Navigator.push(
      context,
      MaterialPageRoute(
        builder: (context) => EditSchedule(
          uid: _uid,
          scheduleNum: s.scheduleNum,
        )),
    ),
    onLongPress: () => msgBox(context, '警告', '確定要刪除行程嗎?', ()
async {
    Delete delete = Delete(
      context: context, uid: _uid, scheduleNum: s.scheduleNum);
    bool isError = await delete.getIsError();
    if (isError) {
      toast(context, '刪除行程錯誤');
    }
  })),
  ));
}
}

return itemList;
}

```

▼表 9-2-2、Python 部分程式碼描述

程式名稱	studyplan/views.py
以下部分程式碼皆來自於 studyplan/views.py	
方法名稱	create_studyplan
目的	建立讀書計畫
<pre> @action(detail=False, methods=['POST']) def create_studyplan(self, request): data = request.data uid = data.get('uid') try: Account.objects.get(pk=uid) except ObjectDoesNotExist: return not_found(Msg.NotFound.account, request) serializer = CreateStudyPlanSerializer(data=data) serializer.is_valid(raise_exception=True) subjects = serializer.validated_data.pop('subjects') if len(subjects) == 0: return err(Msg.Err.StudyPlan.at_least_one_subject, 'ST-A-001', request) study_plan = None schedule_num = data['scheduleNum'] if not schedule_num: serializer.validated_data.update({'type_id': 1}) schedule_no = serializer.save() study_plan = StudyPlan.objects.create(create_id=uid, schedule_no=schedule_no) elif isinstance(schedule_num, int): if not Schedule.objects.filter(serial_no=schedule_num).exists(): return not_found(Msg.NotFound.schedule, request) if not PersonalSchedule.objects.filter(schedule_no=schedule_num, user_id=uid).exists(): </pre>	

```

        return no_authority('行程建立讀書計畫', request)

    if StudyPlan.objects.filter(schedule_no=schedule_num).exists():
        return err(Msg.Err.StudyPlan.already_existed, 'ST-A-002', request)

    schedule = Schedule.objects.get(pk=schedule_num)
    if schedule.type_id == 1:
        schedule.schedule_start = serializer.validated_data['schedule_start']
        schedule.schedule_end = serializer.validated_data['schedule_end']
        schedule.save()
        study_plan = StudyPlan.objects.create(create_id=uid,
schedule_no=schedule)
    else:
        return err(Msg.Err.StudyPlan.not_study_plan_schedule, 'ST-A-003',
request)

    all_subjects = []
    num = 0
    for subject in subjects:
        num += 1
        note_no = None
        if subject['note_no'] and Note.objects.filter(serial_no=subject['note_no'],
create_id=uid).exists():
            note_no = Note.objects.get(pk=subject['note_no'])
        plan_content = PlanContent(plan_no=study_plan, plan_num=num,
subject=subject['subject'],
                                plan_start=subject['plan_start'],
plan_end=subject['plan_end'],
                                note_no=note_no, is_rest=subject['is_rest'],
remark=subject['remark'])
        all_subjects.append(plan_content)
    PlanContent.objects.bulk_create(all_subjects)
    return success(request=request)

```

方法名稱	edit_studyplan
目的	編輯讀書計畫

```

@action(detail=False, methods=['PUT'])
def edit_studyplan(self, request):
    data = request.data
    uid = data.get('uid')
    study_plan_num = data.get('studyplanNum')

    try:
        study_plan = StudyPlan.objects.get(pk=study_plan_num)
    except ObjectDoesNotExist:
        return not_found(Msg.NotFound.study_plan, request)

    creator = study_plan.create_id
    connect_no = study_plan.schedule_no.connect_group_no
    user_in_group = GroupMember.objects.filter(group_no=connect_no, user_id=uid,
status_id__in=[1, 4])

    if uid != creator and \
        (not study_plan.is_authority or not user_in_group.exists()):
        return no_authority('編輯讀書計畫', request)

    serializer = CreateStudyPlanSerializer(data=data)
    serializer.is_valid(raise_exception=True)
    subjects = serializer.validated_data.pop('subjects')

    if len(subjects) == 0:
        return err(Msg.Err.StudyPlan.at_least_one_subject, 'ST-B-001', request)

    all_subjects = []
    num = 0
    creator_note = Note.objects.filter(create_id=creator).values_list('serial_no',
flat=True)

    for subject in subjects:
        num += 1
        note_no = subject['note_no']
        if note_no:
            group_note = Note.objects.filter(serial_no=note_no,
group_no=connect_no).exists()
            if not group_note and note_no not in creator_note:
                return not_found(Msg.NotFound.note, request)

```

```

        note_no = Note.objects.get(pk=note_no)
        subject_item = PlanContent(plan_no=study_plan, plan_num=num,
subject=subject['subject'],
                                plan_start=subject['plan_start'],
plan_end=subject['plan_end'],
                                note_no=note_no, is_rest=subject['is_rest'],
remark=subject['remark'])
        all_subjects.append(subject_item)

plan_content = PlanContent.objects.filter(plan_no=study_plan)
plan_content.delete()
PlanContent.objects.bulk_create(all_subjects)

if uid == creator:
    study_plan.is_authority = data['is_authority']
    study_plan.save()

schedule = study_plan.schedule_no
old_schedule_name = schedule.schedule_name
schedule.schedule_name = serializer.validated_data['schedule_name']
schedule.schedule_start = serializer.validated_data['schedule_start']
schedule.schedule_end = serializer.validated_data['schedule_end']
schedule.save()

if schedule.connect_group_no:
    if old_schedule_name != schedule.schedule_name:
        GroupLog.objects.create(do_time=datetime.now(),
group_no=schedule.connect_group_no, user_id=uid,
                                trigger_type='U', do_type_id=4,
                                old=old_schedule_name,
new=schedule.schedule_name)
    else:
        GroupLog.objects.create(do_time=datetime.now(),
group_no=schedule.connect_group_no, user_id=uid,
                                trigger_type='U', do_type_id=4,
new=schedule.schedule_name)
    return success(request=request)

```

方法名稱	sharing
目的	共享讀書計畫

```

@action(detail=False, methods=['PATCH'])
def sharing(self, request):
    data = request.data

    uid = data.get('uid')
    study_plan_num = data.get('studyplanNum')
    group_num = data.get('groupNum')

    user_study_plan = StudyPlan.objects.filter(create_id=uid,
serial_no=study_plan_num)
    user_in_group = GroupMember.objects.filter(group_no=group_num, user_id=uid,
status_id__in=[1, 4])

    if user_study_plan.exists() and user_in_group.exists():
        group_num = Group.objects.get(pk=group_num)
        schedule = StudyPlan.objects.get(pk=study_plan_num).schedule_no

        if not schedule.connect_group_no:
            schedule.connect_group_no = group_num
            schedule.save()
            GroupLog.objects.create(do_time=datetime.now(),
group_no=group_num, user_id=uid,
                                trigger_type='I', do_type_id=4,
new=schedule.schedule_name)
            return success(request=request)
        elif schedule.connect_group_no != group_num:
            return err(Msg.Err.StudyPlan.only_share_with_one_group, 'ST-C-001',
request)
        else:
            return err(Msg.Err.StudyPlan.has_been_shared, 'ST-C-002', request)
    elif not user_in_group.exists():
        return not_found(Msg.NotFound.not_in_group, request)
    else:
        return no_authority('分享讀書計畫', request)

```

方法名稱	cancel_sharing
目的	取消共享讀書計畫


```

@action(detail=False, methods=['PATCH'])
def cancel_sharing(self, request):
    data = request.data

    uid = data.get('uid')
    study_plan_num = data.get('studyplanNum')

    user_study_plan = StudyPlan.objects.filter(create_id=uid,
serial_no=study_plan_num)
    if user_study_plan.exists():
        schedule = StudyPlan.objects.get(pk=study_plan_num).schedule_no

        if schedule.connect_group_no:
            plan_content = PlanContent.objects.filter(plan_no=study_plan_num,
note_no__isnull=False)
            for content in plan_content:
                if content.note_no.create_id != uid:

plan_content.filter(plan_num=content.plan_num).update(note_no=None)

            group_no = schedule.connect_group_no
            schedule.connect_group_no = None
            schedule.save()
            GroupLog.objects.create(do_time=datetime.now(), group_no=group_no,
user_id=uid,

                                trigger_type='D', do_type_id=4,
old=schedule.schedule_name)
            return success(request=request)
        else:
            return err(Msg.Err.StudyPlan.has_been_canceled, 'ST-D-001', request)
    else:
        return no_authority('取消分享讀書計畫', request)

```

方法名稱	delete
目的	刪除讀書計畫

```

@action(detail=False, methods=['DELETE'])
def delete(self, request):
    data = request.data

    uid = data.get('uid')
    study_plan_num = data.get('studyplanNum')

    try:
        study_plan = StudyPlan.objects.get(pk=study_plan_num)
    except ObjectDoesNotExist:
        return not_found(Msg.NotFound.study_plan, request)

    user_study_plan = StudyPlan.objects.filter(create_id=uid,
serial_no=study_plan_num)
    if user_study_plan.exists():
        schedule = study_plan.schedule_no

        if schedule.connect_group_no:
            return err(Msg.Err.StudyPlan.disconnect_group_first, 'ST-E-001', request)

        plan_content = PlanContent.objects.filter(plan_no=study_plan_num)
        plan_content.delete()
        study_plan.delete()

        personal_schedule = PersonalSchedule.objects.filter(schedule_no=schedule)
        if not personal_schedule.exists():
            schedule.delete()
        return success(request=request)
    else:
        return no_authority('刪除讀書計畫', request)

```

方法名稱	get
目的	取得讀書計畫

```

@action(detail=False)
def get(self, request):
    data = request.query_params

    uid = data.get('uid')
    study_plan_num = data.get('studyplanNum')

```

```

try:
    study_plan = StudyPlan.objects.get(pk=study_plan_num)
except ObjectDoesNotExist:
    return not_found(Msg.NotFound.study_plan, request)
schedule = study_plan.schedule_no

user_is_member = GroupMember.objects.filter(user_id=uid,
group_no=schedule.connect_group_no,
status_id__in=[1, 4])

if uid == study_plan.create_id or user_is_member.exists():
    plan_content = PlanContent.objects.filter(plan_no=study_plan_num)
    return success({
        'creatorId': study_plan.create_id,
        'creator': study_plan.create.name,
        'isAuthority': bool(study_plan.is_authority),
        'title': schedule.schedule_name,
        'date': schedule.schedule_start.date(),
        'startTime': schedule.schedule_start,
        'endTime': schedule.schedule_end,
        'subject': [
            {
                'subjectName': p.subject,
                'subjectStart': p.plan_start,
                'subjectEnd': p.plan_end,
                'remark': p.remark,
                'noteNum': p.note_no.pk if p.note_no else None,
                'rest': bool(p.is_rest),
            }
            for p in plan_content
        ]
    }, request)
else:
    return not_found(Msg.NotFound.study_plan, request)

```

方法名稱	personal_list
目的	取得個人過去和未來讀書計畫列表
@action(detail=False) def personal_list(self, request): data = request.query_params	

```

uid = data.get('uid')
study_plan = StudyplanList.objects.filter(create_id=uid)

plan = [
    {
        'studyplanNum': s.studyplan_num,
        'title': s.schedule_name,
        'date': s.schedule_start.date(),
        'startTime': s.schedule_start,
        'endTime': s.schedule_end,
    }
    for s in study_plan
]

return success({
    'pastStudyplan': [p for p in plan if p['endTime'] < datetime.now()],
    'futureStudyplan': [p for p in plan if p['endTime'] >= datetime.now()]
}, request)

```

方法名稱	personal_share_list
目的	取得個人共享或未共享讀書計畫列表

```

@action(detail=False)
def personal_share_list(self, request):
    data = request.query_params

    uid = data.get('uid')
    share_status = int(data.get('shareStatus'))
    study_plan =
StudyplanList.objects.filter(create_id=uid).exclude(connect_group_no__isnull=share_st
atus)

    return success({
        'studyplan': [
            {
                'studyplanNum': s.studyplan_num,
                'title': s.schedule_name,
                'date': s.schedule_start.date(),
                'startTime': s.schedule_start,
                'endTime': s.schedule_end,
            }

```

<pre> } for s in study_plan] }, request) </pre>	
方法名稱	group_list
目的	取得各個群組的過去和未來讀書計畫列表
<pre> @action(detail=False) def group_list(self, request): data = request.query_params uid = data.get('uid') study_plan = GroupStudyplanList.objects.filter(user_id=uid, status_id__in=[1, 4]) past_future = [study_plan.filter(schedule_end__lt=datetime.now()), study_plan.filter(schedule_end__gte=datetime.now())] group_plan = [[{ 'groupNum': group_no, 'groupName': Group.objects.get(pk=group_no).group_name, 'studyplanCount': pf.filter(group_no=group_no).count(), 'studyplanContent': [{ 'studyplanNum': p.studyplan_num, 'title': p.schedule_name, 'date': p.schedule_start.date(), 'startTime': p.schedule_start, 'endTime': p.schedule_end, } for p in pf.filter(group_no=group_no)] } for group_no in pf.values_list('group_no', flat=True).distinct()] for pf in past_future] </pre>	

```

return success({
    'pastStudyplan': group_plan[0],
    'futureStudyplan': group_plan[1]
}, request)

```

方法名稱

one_group_list

目的

取得單一群組過去和未來讀書計畫列表

```

@action(detail=False)
def one_group_list(self, request):
    data = request.query_params

    uid = data.get('uid')
    group_num = data.get('groupNum')
    study_plan = GroupStudyplanList.objects.filter(user_id=uid,
group_no=group_num, status_id__in=[1, 4])

    plan = [
        {
            'studyplanNum': s.studyplan_num,
            'creatorId': s.create_id,
            'creator': s.creator,
            'title': s.schedule_name,
            'date': s.schedule_start.date(),
            'startTime': s.schedule_start,
            'endTime': s.schedule_end
        }
        for s in study_plan
    ]

    return success({
        'pastStudyplan': [p for p in plan if p['endTime'] < datetime.now()],
        'futureStudyplan': [p for p in plan if p['endTime'] >= datetime.now()]
    }, request)

```

第10章 測試模型

10-1 測試計畫

以使用者角度使用 My Day，測試以下功能：

A. 帳號

1. 註冊：是否註冊成功
2. 登入/登出：是否登入/登出成功
3. 忘記密碼：是否取得驗證碼及登入成功
4. 更改密碼：是否更改成功

B. 課表

1. 建立課表：是否建立成功
2. 編輯課表：是否編輯成功
3. 刪除課表：是否刪除成功
4. 取得課表列表：是否取得成功
5. 取得課表：是否取得成功
6. 取得主課表列表：是否成功取得
7. 編輯課表資訊：是否編輯成功
8. 分享課表：是否分享成功
9. 刪除接收課表：是否刪除成功
10. 取得接收課表內容列表：是否取得成功
11. 接收課表：是否接收成功
12. 取得分享碼：是否取得成功

13. 分享碼接收課表：是否取得成功
14. 取得節次時間：是否取得成功
15. 取得接收課表列表：是否取得成功

C. 行程

1. 建立行程：是否建立成功
2. 編輯行程：是否編輯成功
3. 刪除行程：是否刪除成功
4. 取得當日行程列表：是否取得成功
5. 取得行程：是否取得成功
6. 建立共同行程：是否建立成功
7. 取得共同行程列表：是否取得成功
8. 取得共同行程：是否取得成功
9. 取得倒數列表：是否取得成功

D. 讀書計畫

1. 建立讀書計畫：是否建立成功
2. 編輯讀書計畫：是否編輯成功
3. 刪除讀書計畫：是否刪除成功
4. 共享讀書計畫：是否共享成功
5. 取消共享讀書計畫：是否取消成功
6. 取得讀書計畫列表：是否取得成功
7. 取得讀書計畫：是否取得成功

8. 取得群組讀書計畫列表：是否取得成功

E. 設定

1. 主題設定：是否設定成功

2. 隱私設定：是否設定成功

3. 個人資料列表：是否取得成功

4. 編輯個人資料：是否編輯成功

F. 好友

1. 邀請成為好友：是否邀請成功

2. 好友邀請回覆：是否回覆成功

3. 刪除好友：是否刪除成功

4. 新增摯友：是否設定成功

5. 刪除摯友：是否刪除成功

6. 取得好友列表：是否取得成功

7. 取得好友：是否取得成功

8. 取得好友邀請列表：是否取得成功

9. 取得摯友列表：是否取得成功

G. 群組

1. 建立群組：是否建立成功

2. 取得群組/群組邀請列表：是否取得成功

3. 同意/拒絕加入群組：是否加入/拒絕成功

4. 取得群組：是否取得成功

5. 編輯群組：是否編輯成功
6. 群組邀請好友：是否邀請成功
7. 設定管理員：是否設定成功
8. 退出群組：是否退出成功
9. 自行加入現有群組：是否加入成功
10. 取得成員列表：是否取得成功

H. 臨時群組

1. 建立臨時群組：是否建立成功
2. 取得臨時群組/臨時群組邀請列表：是否取得成功
3. 取得臨時群組邀請資訊：是否取得成功
4. 同意加入/拒絕臨時群組：是否加入成功
5. 取得臨時群組：是否取得成功
6. 編輯臨時群組：是否編輯成功
7. 臨時群組邀請好友：是否邀請成功
8. 設定管理員：是否設定成功
9. 退出群組：是否退出成功
10. 取得成員列表：是否取得成功

I. 投票

1. 建立投票：是否建立成功
2. 編輯投票：是否編輯成功
3. 刪除投票：是否刪除成功

4. 取得投票列表：是否取得成功
5. 取得投票：是否取得成功
6. 投票：是否投票成功
7. 加入投票選項：是否加入成功

J. 筆記

1. 建立筆記：是否建立成功
2. 編輯筆記：是否編輯成功
3. 刪除筆記：是否刪除成功
4. 取得筆記列表：是否取得成功
5. 取得筆記：是否取得成功
6. 取得群組筆記列表：是否取得成功
7. 共享筆記：是否共享成功
8. 取消共享筆記：是否取消成功

10-2 測試個案與測試結果

測試結果：以使用者角度使用 My Day

▼表 10-2-A1、測試功能「註冊」

功能編號	A1	功能名稱	註冊
測試流程	1. 進入 My Day 2. 輸入資料 3. 註冊		
預期成果	成功註冊		
執行結果	註冊成功		

▼表 10-2-A2、測試功能「登入/登出」

功能編號	A2	功能名稱	登入/登出
測試流程	1. 進入 My Day 2. 輸入帳號密碼 3. 登入		
預期成果	成功登入/登出		
執行結果	登入/登出成功		

▼表 10-2-A3、測試功能「忘記密碼」

功能編號	A3	功能名稱	忘記密碼
測試流程	1. 進入 My Day 2. 進入忘記密碼		

	3. 輸入帳號(Email) 4. 點「發送驗證碼」按鈕 5. 在 Email 收到驗證碼 6. 輸入驗證碼後，再輸入新密碼即可登入
預期成果	成功發送驗證碼並且登入
執行結果	發送驗證碼並且登入成功

▼表 10-2-A4、測試功能「更改密碼」

功能編號	A4	功能名稱	更改密碼
測試流程	1. 登入 My Day 2. 進入設定畫面 3. 進入個人資料畫面 4. 進入更改密碼畫面 5. 輸入新密碼兩次 6. 更改		
預期成果	成功更改密碼		
執行結果	更改密碼成功		

▼表 10-2-B1、測試功能「建立課表」

功能編號	B1	功能名稱	建立課表
測試流程	1. 登入 My Day 2. 進入管理課表		

	3. 進入新增課表 4. 輸入課表資訊 5. 輸入課表 6. 確認
預期成果	成功建立課表
執行結果	建立課表成功

▼表 10-2-B2、測試功能「編輯課表」

功能編號	B2	功能名稱	編輯課表
測試流程	1. 登入 My Day 2. 進入管理課表 3. 進入編輯課表 4. 點選該課表 5. 輸入要更改的課表 6. 確認		
預期成果	成功編輯課表		
執行結果	編輯課表成功		

▼表 10-2-B3、測試功能「刪除課表」

功能編號	B3	功能名稱	刪除課表
測試流程	1. 登入 My Day 2. 進入管理課表		

	3. 進入編輯課表 4. 點選該課表 5. 點選右上角更多 6. 刪除課表
預期成果	成功刪除課表
執行結果	刪除課表成功

▼表 10-2-B4、測試功能「取得課表列表」

功能編號	B4	功能名稱	取得課表列表
測試流程	1. 登入 My Day 2. 進入管理課表 3. 進入編輯課表		
預期成果	成功取得課表列表		
執行結果	取得課表列表成功		

▼表 10-2-B5、測試功能「取得課表」

功能編號	B5	功能名稱	取得課表
測試流程	1. 登入 My Day 2. 進入管理課表 3. 進入編輯課表 4. 點選該課表		
預期成果	成功取得課表		

執行結果	取得課表成功
------	--------

▼表 10-2-B6、測試功能「取得主課表列表」

功能編號	B6	功能名稱	取得主課表列表
測試流程	1. 進入 My Day 2. 輸入帳號密碼 3. 登入 4. 進入 My Day		
預期成果	成功取得主課表列表		
執行結果	取得主課表列表成功		

▼表 10-2-B7、測試功能「編輯課表資訊」

功能編號	B7	功能名稱	編輯課表資訊
測試流程	1. 登入 My Day 2. 進入管理課表 3. 進入編輯課表 4. 點選該課表 5. 點選右上角更多 6. 進入課表資訊 7. 輸入課表資訊 8. 確認		
預期成果	成功編輯課表資訊		

執行結果	編輯課表資訊成功
------	----------

▼表 10-2-B8、測試功能「分享課表」

功能編號	B8	功能名稱	分享課表
測試流程	1. 登入 My Day 2. 進入分享課表 3. 點選要分享的課表 4. 選擇分享給誰 5. 確認		
預期成果	成功分享課表		
執行結果	分享課表成功		

▼表 10-2-B9、測試功能「刪除接收課表」

功能編號	B9	功能名稱	刪除接收課表
測試流程	1. 登入 My Day 2. 進入接收課表 3. 點選要接收的課表 4. 刪除		
預期成果	成功刪除接收課表		
執行結果	刪除接收課表成功		

▼表 10-2-B10、測試功能「取得接收課表列表內容」

功能編號	B10	功能名稱	取得接收課表列表
------	-----	------	----------

			內容
測試流程	1. 登入 My Day 2. 進入接收課表 3. 點選要接收的課表		
預期成果	成功取得接收課表列表內容		
執行結果	取得接收課表列表內容成功		

▼表 10-2-B11、測試功能「接收課表」

功能編號	B11	功能名稱	接收課表
測試流程	1. 登入 My Day 2. 進入接收課表 3. 點選要接收的課表 4. 點選預覽 5. 確認		
預期成果	成功接收課表		
執行結果	接收課表成功		

▼表 10-2-B12、測試功能「取得分享碼」

功能編號	B12	功能名稱	取得分享碼
測試流程	1. 登入 My Day 2. 進入分享課表 3. 點選要分享的課表		

預期成果	成功取得分享碼
執行結果	取得分享碼成功

▼表 10-2-B13、測試功能「分享碼接收課表」

功能編號	B13	功能名稱	分享碼接收課表
測試流程	4. 登入 My Day 5. 進入分享課表 6. 輸入分享碼 7. 送出 8. 確認		
預期成果	成功分享碼接收課表		
執行結果	分享碼接收課表成功		

▼表 10-2-B14、測試功能「取得節次時間」

功能編號	B14	功能名稱	取得節次時間
測試流程	1. 進入 My Day 2. 輸入帳號密碼 3. 登入 4. 進入 My Day		
預期成果	成功取得節次時間		
執行結果	取得節次時間成功		

▼表 10-2-B15、測試功能「取得接收課表列表」

功能編號	B15	功能名稱	取得接收課表列表
測試流程	1. 登入 My Day 2. 進入接收課表 3. 點選要接收的學期		
預期成果	成功取得接收課表列表		
執行結果	取得接收課表列表成功		

▼表 10-2-C1、測試功能「建立行程」

功能編號	C1	功能名稱	建立行程
測試流程	1. 登入 My Day 2. 點擊「+」 3. 輸入行程內容 4. 確定		
預期成果	成功建立行程		
執行結果	建立行程成功		

▼表 10-2-C2、測試功能「編輯行程」

功能編號	C2	功能名稱	編輯行程
測試流程	1. 登入 My Day 2. 點擊想編輯行程當日隨便一處的格子 3. 點選想編輯的行程 4. 編輯行程內容		

	5. 確定
預期成果	成功編輯行程
執行結果	編輯行程成功

▼表 10-2-C3、測試功能「刪除行程」

功能編號	C3	功能名稱	刪除行程
測試流程	1. 登入 My Day 2. 點擊想編輯行程當日隨便一處的格子 3. 長按想刪除的行程 4. 刪除		
預期成果	成功刪除行程		
執行結果	刪除行程成功		

▼表 10-2-C4、測試功能「取得當日行程列表」

功能編號	C4	功能名稱	取得當日行程列表
測試流程	1. 登入 My Day 2. 點擊想編輯行程當日隨便一處的格子		
預期成果	成功取得當日行程列表		
執行結果	取得當日行程列表成功		

▼表 10-2-C5、測試功能「取得行程」

功能編號	C5	功能名稱	取得行程
測試流程	1. 登入 My Day		

	2. 點擊想編輯行程當日隨便一處的格子 3. 選擇行程 4. 確定
預期成果	成功取得行程
執行結果	取得行程成功

▼表 10-2-C6、測試功能「建立共同行程」

功能編號	C6	功能名稱	建立共同行程
測試流程	1. 登入 My Day 2. 進入群組/臨時群組畫面 3. 選擇群組/臨時群組 4. 點擊共同行程 5. 點擊「+」 6. 輸入內容 7. 確定		
預期成果	成功建立共同行程		
執行結果	建立共同行程成功		

▼表 10-2-C7、測試功能「取得共同行程列表」

功能編號	C7	功能名稱	取得共同行程列表
測試流程	1. 登入 My Day 2. 進入群組/臨時群組畫面		

	3. 選擇群組/臨時群組 4. 進入共同行程畫面 5. 取得列表
預期成果	成功取得共同行程列表
執行結果	取得共同行程列表成功

▼表 10-2-C8、測試功能「取得共同行程」

功能編號	C8	功能名稱	取得共同行程
測試流程	1. 登入 My Day 2. 進入群組/臨時群組畫面 3. 選擇群組/臨時群組 4. 進入共同行程畫面 5. 點擊行程 6. 取得共同行程		
預期成果	成功取得共同行程		
執行結果	取得共同行程成功		

▼表 10-2-C9、測試功能「取得倒數列表」

功能編號	C9	功能名稱	取得倒數列表
測試流程	1. 登入 My Day 2. 點選倒數		
預期成果	成功取得倒數列表		

執行結果	取得倒數列表成功
------	----------

▼表 10-2-D1、測試功能「建立讀書計畫」

功能編號	D1	功能名稱	建立讀書計畫
測試流程	1. 登入 My Day 2. 進入學習畫面 3. 進入讀書計畫畫面 4. 點擊「+」 5. 輸入內容 6. 確定		
預期成果	成功建立讀書計畫		
執行結果	建立讀書計畫成功		

▼表 10-2-D2、測試功能「編輯讀書計畫」

功能編號	D2	功能名稱	編輯讀書計畫
測試流程	1. 登入 My Day 2. 進入學習畫面 3. 進入讀書計畫畫面 4. 選擇讀書計畫 5. 選擇編輯 6. 輸入內容 7. 確定		

預期成果	成功編輯讀書計畫
執行結果	編輯讀書計畫成功

▼表 10-2-D3、測試功能「刪除讀書計畫」

功能編號	D3	功能名稱	刪除讀書計畫
測試流程	1. 登入 My Day 2. 進入學習畫面 3. 進入讀書計畫畫面 4. 選擇讀書計畫 5. 點擊刪除		
預期成果	成功刪除讀書計畫		
執行結果	刪除讀書計畫成功		

▼表 10-2-D4、測試功能「共享讀書計畫」

功能編號	D4	功能名稱	共享讀書計畫
測試流程	1. 登入 My Day 2. 進入群組畫面 3. 選擇群組 4. 點擊共同讀書計畫 5. 點擊「+」 6. 選擇讀書計畫 7. 確定		

預期成果	成功共享讀書計畫
執行結果	共享讀書計畫成功

▼表 10-2-D5、測試功能「取消共享讀書計畫」

功能編號	D5	功能名稱	取消共享讀書計畫
測試流程	1. 登入 My Day 2. 進入學習畫面 3. 進入讀書計畫畫面 4. 進入已共享畫面 5. 選擇讀書計畫 6. 點擊取消分享		
預期成果	成功取消分享讀書計畫		
執行結果	取消分享讀書計畫成功		

▼表 10-2-D6、測試功能「取得讀書計畫列表」

功能編號	D6	功能名稱	取得讀書計畫列表
測試流程	1. 登入 My Day 2. 進入學習畫面 3. 進入讀書計畫畫面 4. 取得列表		
預期成果	成功取得讀書計畫列表		
執行結果	取得讀書計畫列表成功		

▼表 10-2-D7、測試功能「取得讀書計畫」

功能編號	D7	功能名稱	取得讀書計畫
測試流程	1. 登入 My Day 2. 進入學習畫面 3. 進入讀書計畫畫面 4. 選擇讀書計畫		
預期成果	成功取得讀書計畫		
執行結果	取得讀書計畫成功		

▼表 10-2-D8、測試功能「取得群組讀書計畫列表」

功能編號	D8	功能名稱	取得群組讀書計畫列表
測試流程	1. 登入 My Day 2. 進入群組畫面 3. 進入某一群組畫面 4. 進入共同讀書計畫畫面 5. 取得列表		
預期成果	成功取得群組讀書計畫列表		
執行結果	取得群組讀書計畫列表成功		

▼表 10-2-E1、測試功能「主題設定」

功能編號	E1	功能名稱	主題設定
測試流程	1. 登入 My Day		

	2. 進入設定畫面 3. 點擊主題 4. 設定主題
預期成果	成功設定主題
執行結果	尚未完全設定主題

▼表 10-2-E2、測試功能「隱私設定」

功能編號	E2	功能名稱	隱私設定
測試流程	1. 登入 My Day 2. 進入設定畫面 3. 點擊隱私 4. 隱私設定		
預期成果	成功設定隱私		
執行結果	隱私設定成功		

▼表 10-2-E3、測試功能「個人資料列表」

功能編號	E3	功能名稱	取得個人資料列表
測試流程	1. 登入 My Day 2. 進入設定畫面 3. 點擊個人資料		
預期成果	成功取得個人資料列表		
執行結果	個人資料列表成功		

▼表 10-2-E4、測試功能「編輯個人資料」

功能編號	E4	功能名稱	編輯個人資料
測試流程	1. 登入 My Day 2. 進入設定畫面 3. 點擊個人資料 4. 編輯個人資料		
預期成果	成功編輯個人資料		
執行結果	編輯個人資料成功		

▼表 10-2-F1、測試功能「邀請成為好友」

功能編號	F1	功能名稱	邀請成為好友
測試流程	1. 登入 My Day 2. 進入設定畫面 3. 點擊好友 4. 點擊「+」 5. 輸入帳號		
預期成果	成功邀請成為好友		
執行結果	邀請成為好友成功		

▼表 10-2-F2、測試功能「好友邀請回覆」

功能編號	F2	功能名稱	好友邀請回覆
測試流程	1. 登入 My Day 2. 進入設定畫面		

	3. 點擊好友 4. 進入交友邀請畫面 5. 回覆邀請
預期成果	成功好友邀請回覆
執行結果	好友邀請回覆成功

▼表 10-2-F3、測試功能「刪除好友」

功能編號	F3	功能名稱	刪除好友
測試流程	1. 登入 My Day 2. 進入設定畫面 3. 進入好友列表畫面 4. 點擊移除		
預期成果	成功刪除好友		
執行結果	刪除好友成功		

▼表 10-2-F4、測試功能「新增摯友」

功能編號	F4	功能名稱	新增摯友
測試流程	1. 登入 My Day 2. 進入設定畫面 3. 進入好友列表畫面 4. 進入摯友列表畫面 5. 輸入摯友帳號		

預期成果	成功新增摯友
執行結果	新增摯友成功

▼表 10-2-F5、測試功能「刪除摯友」

功能編號	F5	功能名稱	刪除摯友
測試流程	1. 登入 My Day 2. 進入設定畫面 3. 進入好友列表畫面 4. 進入摯友列表畫面 5. 點擊移除		
預期成果	成功刪除摯友		
執行結果	刪除摯友成功		

▼表 10-2-F6、測試功能「取得好友列表」

功能編號	F6	功能名稱	取得好友列表
測試流程	1. 登入 My Day 2. 進入設定畫面 3. 進入好友列表畫面		
預期成果	成功取得好友列表		
執行結果	取得好友列表成功		

▼表 10-2-F7、測試功能「取得好友行程及課表」

功能編號	F7	功能名稱	取得好友行程
------	----	------	--------

			及課表
測試流程	1. 登入 My Day 2. 進入設定畫面 3. 進入好友列表畫面 4. 點選好友頭貼 5. 取得好友行程及課表(如果對方有開權限)		
預期成果	成功取得好友行程及課表		
執行結果	取得好友行程及課表成功		

▼表 10-2-F8、測試功能「取得好友邀請列表」

功能編號	F8	功能名稱	取得好友邀請列表
測試流程	1. 登入 My Day 2. 進入設定畫面 3. 進入好友列表畫面 4. 進入交友邀請畫面		
預期成果	成功取得好友邀請列表		
執行結果	取得好友邀請列表成功		

▼表 10-2-F9、測試功能「取得摯友列表」

功能編號	F9	功能名稱	取得摯友列表
測試流程	1. 登入 MyDay 2. 進入設定畫面		

	3. 進入好友列表畫面 4. 進入摯友列表畫面
預期成果	成功取得摯友列表
執行結果	取得摯友列表成功

▼表 10-2-G1、測試功能「建立群組」

功能編號	G1	功能名稱	建立群組
測試流程	1. 登入 My Day 2. 進入群組畫面 3. 點擊「+」 4. 輸入內容 5. 建立		
預期成果	成功建立群組		
執行結果	建立群組成功		

▼表 10-2-G2、測試功能「取得群組/群組邀請列表」

功能編號	G2	功能名稱	取得群組/群組邀請列表
測試流程	1. 登入 My Day 2. 進入群組畫面 3. 取得列表		
預期成果	成功取得群組/群組邀請列表		

執行結果	取得群組/群組邀請列表成功
------	---------------

▼表 10-2-G3、測試功能「同意/拒絕加入群組」

功能編號	G3	功能名稱	同意/拒絕加入群組
測試流程	1. 登入 My Day 2. 進入群組畫面 3. 在邀約選擇加入/拒絕(有邀約才會顯示)		
預期成果	成功同意/拒絕加入群組		
執行結果	同意/拒絕加入群組成功		

▼表 10-2-G4、測試功能「取得群組」

功能編號	G4	功能名稱	取得群組
測試流程	1. 登入 My Day 2. 進入群組畫面 3. 進入某一群組畫面 4. 取得群組		
預期成果	成功取得群組		
執行結果	取得群組成功		

▼表 10-2-G5、測試功能「編輯群組」

功能編號	G5	功能名稱	編輯群組
測試流程	1. 登入 My Day 2. 進入群組畫面		

	3. 進入某一群組畫面 4. 進入群組設定畫面 5. 進入群組資訊畫面 6. 輸入內容 7. 編輯
預期成果	成功編輯群組
執行結果	編輯群組成功

▼表 10-2-G6、測試功能「群組邀請好友」

功能編號	G6	功能名稱	群組邀請好友
測試流程	1. 登入 My Day 2. 進入群組畫面 3. 進入某一群組畫面 4. 進入邀請好友畫面 5. 選擇好友 6. 邀請		
預期成果	成功群組邀請好友		
執行結果	群組邀請好友成功		

▼表 10-2-G7、測試功能「設定管理者」

功能編號	G7	功能名稱	設定管理者
測試流程	1. 登入 My Day		

	2. 進入群組畫面 3. 進入某一群組畫面 4. 進入群組設定畫面 5. 進入管理者成員畫面 6. 設定管理者
預期成果	成功設定管理者
執行結果	設定管理者成功

▼表 10-2-G8、測試功能「退出群組」

功能編號	G8	功能名稱	退出群組
測試流程	1. 登入 My Day 2. 進入群組畫面 3. 進入某一群組畫面 4. 退出		
預期成果	成功退出群組		
執行結果	退出群組成功		

▼表 10-2-G9、測試功能「自行加入群組」

功能編號	G9	功能名稱	自行加入群組
測試流程	1. 登入 My Day 2. 進入群組畫面 3. 點擊「+」		

	4. 點擊加入群組 5. 輸入群組 ID 6. 加入
預期成果	成功自行加入群組
執行結果	自行加入群組成功

▼表 10-2-G10、測試功能「取得成員列表」

功能編號	G10	功能名稱	取得成員列表
測試流程	1. 登入 My Day 2. 進入群組畫面 3. 進入某一群組畫面 4. 點擊成員按鈕 5. 取得成員列表		
預期成果	成功取得成員列表		
執行結果	取得成員列表成功		

▼表 10-2-H1、測試功能「建立臨時群組」

功能編號	H1	功能名稱	建立臨時群組
測試流程	1. 登入 My Day 2. 進入玩聚(臨時群組)畫面 3. 點擊「+」 4. 輸入內容		

	5. 建立
預期成果	成功建立臨時群組
執行結果	建立臨時群組成功

▼表 10-2-H2、測試功能「取得臨時群組/臨時群組邀請列表」

功能編號	H2	功能名稱	取得臨時群組/ 臨時群組邀請列表
測試流程	1. 登入 My Day 2. 進入玩聚(臨時群組)畫面 3. 取得列表(邀請列表僅在有被邀請才顯示)		
預期成果	成功取得臨時群組/臨時群組邀請列表		
執行結果	取得臨時群組/臨時群組邀請列表成功		

▼表 10-2-H3、測試功能「取得臨時群組邀請資訊」

功能編號	H3	功能名稱	取得臨時群組邀請 資訊
測試流程	1. 登入 My Day 2. 進入玩聚(臨時群組)畫面 3. 點擊被邀請的群組 4. 取得臨時群組邀請資訊		
預期成果	成功取得臨時群組邀請資訊		
執行結果	取得臨時群組邀請資訊成功		

▼表 10-2-H4、測試功能「同意/拒絕加入臨時群組」

功能編號	H4	功能名稱	同意/拒絕加入臨時群組
測試流程	1. 登入 My Day 2. 進入玩聚(臨時群組)畫面 3. 在邀約選擇加入/拒絕(有邀約才會顯示)		
預期成果	成功同意/拒絕加入臨時群組		
執行結果	同意/拒絕加入臨時群組成功		

▼表 10-2-H5、測試功能「取得臨時群組」

功能編號	H5	功能名稱	取得臨時群組
測試流程	1. 登入 My Day 2. 進入玩聚(臨時群組)畫面 3. 進入某一玩聚畫面 4. 取得臨時群組		
預期成果	成功取得臨時群組		
執行結果	取得臨時群組成功		

▼表 10-2-H6、測試功能「編輯臨時群組」

功能編號	H6	功能名稱	編輯臨時群組
測試流程	1. 登入 My Day 2. 進入玩聚(臨時群組)畫面 3. 進入某一玩聚畫面		

	4. 進入群組設定畫面 5. 進入群組資訊畫面 6. 輸入內容 7. 編輯
預期成果	成功編輯臨時群組
執行結果	編輯臨時群組成功

▼表 10-2-H7、測試功能「臨時群組邀請好友」

功能編號	H7	功能名稱	臨時群組邀請好友
測試流程	1. 登入 My Day 2. 進入玩聚(臨時群組)畫面 3. 進入某一玩聚畫面 4. 進入邀請好友畫面 5. 選擇好友 6. 邀請		
預期成果	成功臨時群組邀請好友		
執行結果	臨時群組邀請好友成功		

▼表 10-2-H8、測試功能「設定管理者」

功能編號	H8	功能名稱	設定管理者
測試流程	1. 登入 My Day 2. 進入玩聚(臨時群組)畫面		

	3. 進入某一玩聚畫面 4. 進入群組設定畫面 5. 進入管理者成員畫面 6. 設定管理者
預期成果	成功設定管理者
執行結果	設定管理者成功

▼表 10-2-H9、測試功能「退出臨時群組」

功能編號	H9	功能名稱	退出臨時群組
測試流程	1. 登入 My Day 2. 進入玩聚(臨時群組)畫面 3. 進入某一玩聚畫面 4. 退出		
預期成果	成功退出臨時群組		
執行結果	退出臨時群組成功		

▼表 10-2-H10、測試功能「取得成員列表」

功能編號	H10	功能名稱	取得成員列表
測試流程	1. 登入 My Day 2. 進入玩聚(臨時群組)畫面 3. 進入某一玩聚畫面 4. 點擊成員按鈕		

	5. 取得成員列表
預期成果	成功取得成員列表
執行結果	取得成員列表成功

▼表 10-2-I1、測試功能「建立投票」

功能編號	I1	功能名稱	建立投票
測試流程	1. 登入 My Day 2. 進入群組/臨時群組畫面 3. 進入某一群組/臨時群組畫面 4. 進入投票畫面 5. 輸入內容 6. 建立投票		
預期成果	成功建立投票		
執行結果	建立投票成功		

▼表 10-2-I2、測試功能「編輯投票」

功能編號	I2	功能名稱	編輯投票
測試流程	1. 登入 My Day 2. 進入群組/臨時群組畫面 3. 進入某一群組/臨時群組畫面 4. 進入投票畫面 5. 選擇某一投票		

	6. 點擊編輯 7. 輸入內容 8. 編輯
預期成果	成功編輯投票
執行結果	編輯投票成功

▼表 10-2-I3、測試功能「刪除投票」

功能編號	I3	功能名稱	刪除投票
測試流程	1. 登入 My Day 2. 進入群組/臨時群組畫面 3. 進入某一群組/臨時群組畫面 4. 進入投票畫面 5. 選擇某一投票 6. 刪除		
預期成果	成功刪除投票		
執行結果	刪除投票成功		

▼表 10-2-I4、測試功能「取得投票列表」

功能編號	I4	功能名稱	取得投票列表
測試流程	1. 登入 My Day 2. 進入群組/臨時群組畫面 3. 進入某一群組/臨時群組畫面		

	4. 進入投票畫面 5. 取得投票列表
預期成果	成功取得投票列表
執行結果	取得投票列表成功

▼表 10-2-I5、測試功能「取得投票」

功能編號	I5	功能名稱	取得投票
測試流程	1. 登入 My Day 2. 進入群組/臨時群組畫面 3. 進入某一群組/臨時群組畫面 4. 進入投票畫面 5. 選擇投票 6. 取得投票		
預期成果	成功取得投票		
執行結果	取得投票成功		

▼表 10-2-I6、測試功能「投票」

功能編號	I6	功能名稱	投票
測試流程	1. 登入 My Day 2. 進入群組/臨時群組畫面 3. 進入某一群組/臨時群組畫面 4. 進入投票畫面		

	5. 選擇某一投票 6. 進行投票
預期成果	成功投票
執行結果	投票成功

▼表 10-2-I7、測試功能「加入投票選項」

功能編號	I7	功能名稱	加入投票選項
測試流程	1. 登入 My Day 2. 進入群組/臨時群組畫面 3. 進入某一群組/臨時群組畫面 4. 進入投票畫面 5. 選擇某一投票 6. 加入投票選項		
預期成果	成功加入投票選項		
執行結果	加入投票選項成功		

▼表 10-2-J1、測試功能「建立筆記」

功能編號	J1	功能名稱	建立筆記
測試流程	1. 登入 My Day 2. 進入學習畫面 3. 進入筆記畫面 4. 點擊「+」		

	5. 輸入內容 6. 建立
預期成果	成功建立筆記
執行結果	建立筆記成功

▼表 10-2-J2、測試功能「編輯筆記」

功能編號	J2	功能名稱	編輯筆記
測試流程	1. 登入 My Day 2. 進入學習畫面 3. 進入筆記畫面 4. 選擇筆記 5. 點擊編輯 6. 輸入內容 7. 編輯		
預期成果	成功編輯筆記		
執行結果	編輯筆記成功		

▼表 10-2-J3、測試功能「刪除筆記」

功能編號	J3	功能名稱	刪除筆記
測試流程	1. 登入 My Day 2. 進入學習畫面 3. 進入筆記畫面		

	4. 選擇筆記 5. 點擊刪除
預期成果	成功刪除筆記
執行結果	刪除筆記成功

▼表 10-2-J4、測試功能「取得筆記列表」

功能編號	J4	功能名稱	取得筆記列表
測試流程	1. 登入 My Day 2. 進入學習畫面 3. 進入筆記畫面 4. 取得列表		
預期成果	成功取得筆記列表		
執行結果	取得筆記列表成功		

▼表 10-2-J5、測試功能「取得筆記」

功能編號	J5	功能名稱	取得筆記
測試流程	1. 登入 My Day 2. 進入學習畫面 3. 進入筆記畫面 4. 選擇筆記 5. 取得筆記		
預期成果	成功取得筆記		

執行結果	取得筆記成功
------	--------

▼表 10-2-J6、測試功能「取得群組筆記列表」

功能編號	J6	功能名稱	取得群組筆記列表
測試流程	1. 登入 My Day 2. 進入群組畫面 3. 進入某一群組畫面 4. 進入共同筆記畫面 5. 取得列表		
預期成果	成功取得群組筆記列表		
執行結果	取得群組筆記列表成功		

▼表 10-2-J7、測試功能「共享筆記」

功能編號	J7	功能名稱	共享筆記
測試流程	1. 登入 My Day 2. 進入群組畫面 3. 進入某一群組畫面 4. 進入共享筆記畫面 5. 點擊「+」 6. 選擇筆記 7. 確認		
預期成果	成功共享筆記		

執行結果	共享筆記成功
------	--------

▼表 10-2-J8、測試功能「取消共享筆記」

功能編號	J8	功能名稱	取消共享筆記
測試流程	1. 登入 My Day 2. 進入群組畫面 3. 進入某一群組畫面 4. 進入共享筆記畫面 5. 選擇筆記		
預期成果	成功共享筆記		
執行結果	共享筆記成功		

第11章 操作手冊


▼表 11-1-1、操作手冊

系統安裝元件資訊	
元件名稱	My Day
版本	V1.0
軟體類別	行事曆
支援語言	繁體中文
價格	免費
低版本需求	iOS 11 以上 Android 9 以上
內容分級	普遍級
需要權限	隱私 ●相簿存取 裝置存取權 ●完整網路存取權 ●接收網路資料

第12章 使用手冊

1. 進入 My Day 中先註冊資料再登入

▼表 12-1-1、註冊及登入

步驟一	
	

2. 抵達主頁後新增行程，按右下角加號，填寫要新增的內容，填寫完按圈圈確認完成

▼表 12-1-2、新增行程

步驟一



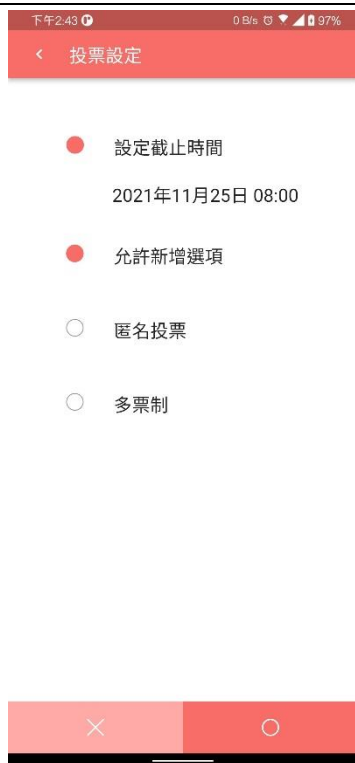
- 按下方列表從左邊數來第二個群組圖示，進入後按右上角加號建立群組，填寫完群組資訊按圈圈確認。進入看電影群組後，按投票建立投票，輸入要投票的主題

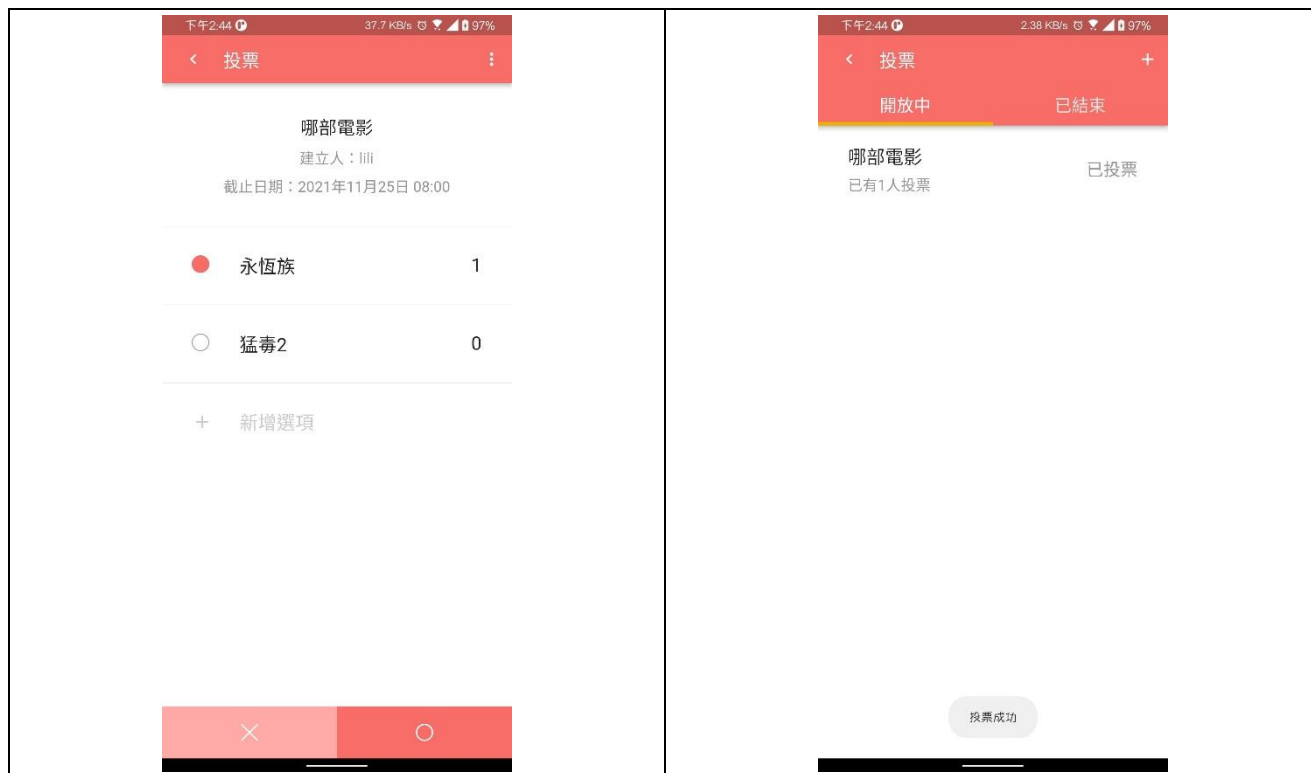
及內容，確認後選擇投票設定。點選剛剛建立的投票，點選要投的選項，再回到投票列表查看。點選共同行程，輸入行程內容。

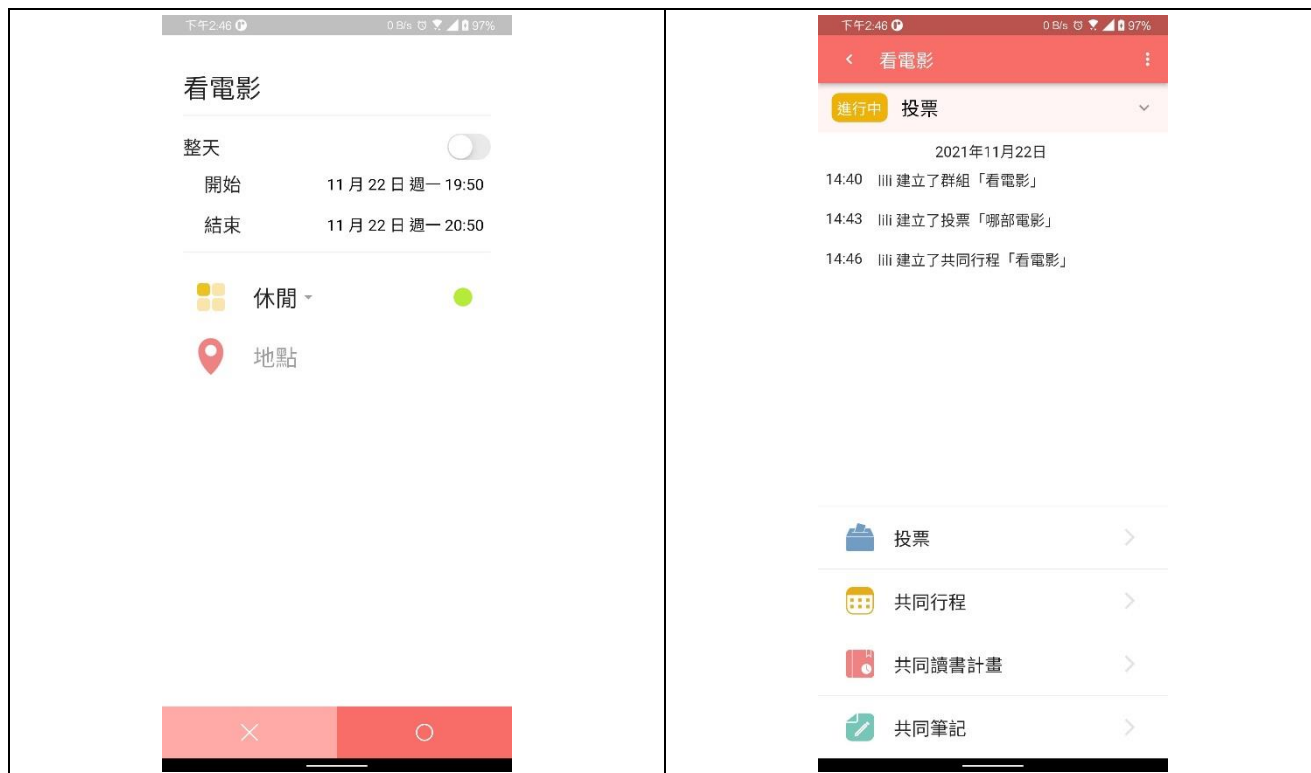
▼表 12-1-3、群組功能

步驟一(建立群組)	
	
	

步驟二(投票)







4. 按下方列表從左邊數來第三個玩聚圖示，建立玩聚輸入群組資訊以及邀請好友。

▼表 12-1-4、玩聚





5. 按下方列表最右邊讀書圖示，進入讀書計畫，按右上角建立讀書計畫，輸入標題、時間以及看的科目，確認是否中間需要休息，按圈圈建立完成

▼表 12-1-5、讀書計畫

步驟一	
	
	

6. 按下方列表最右邊讀書圖示，進入筆記，點選右上角加號，輸入標題、分類、上傳圖片，按圈圈後完成。

▼表 12-1-6 筆記

步驟一	
	
	

第13章 感想

10656003 吳東陽

這是我第一次做專題，不同於之前每學期的小組報告，這次的規模顯然大了很多，在這一年半裡的相處能磨練自己人與人的相處，還學到程式以外的技能，像是如何找出問題並且找出答案的能力，並不是像以前遇到問題就舉手向老師提問，因為在專題裡老師不會常常在身邊，所以就必須自己上網找資料，在找資料的過程中發現其網路上的大家都會遇到差不多的狀況，下面就有人已經幫你解決了，就算第一個網友沒辦法解決下面還是有其他網友提供方法可以去試看看。最後我想要感謝組員，我們討論事情都很快，就算有意見衝突也可以很快解決，有問題的話也會互相幫忙，還有謝謝我們的指導老師，林宏仁老師，在四年級下的時候找四技的學長教我們 Django，雖然那時候我有課沒跟到全部，但還是謝謝老師幫我們找學長教我們。

10656004 張恩瑀

在專題中我覺得學到最多的莫過於時間管理，因為身為組長必須安排全組的進度，因此原本對於時間相當沒有觀念的我是非常大的挑戰，在專題的期間對於時間的安排從原本的生疏到現在已經漸漸的熟練，希望在未來可以更加進步；除此之外在專題中我還有學到許多跟團隊合作相關的事，在團隊合作中最重要的是溝通，我們從專題開始幾乎每週都有例行的會議，在一次又一次的開會中，我們學習到如何溝通才能讓進度進行得更順利，也因為要讓大家都對同一件事有同樣的理解，因此溝通也是絕對必要的；另外技術在專題中的重要性也是不可忽視的，研究、學習各種新的技術，包含各種框架像是 Django、Flutter 等，而且為了不延遲進度，也要在其他同學不會

的時候提供協助，在協助他人的同時還可以精進自己的實力，因此除了組內的同學之外我也很樂意協助他組的同學，當然在我自己有疑問的時候也會向他人尋求協助，在這樣的交流中大家可以不但可以對技術有更深的了解，而且還可以發現自己的不足，在教導別人以及提問的時候還能夠練習更好的表達能力，像這樣的交流簡直就是一舉數得；當然在這一年多的時間內還有非常多的事情，很多都是我們不曾遇過的問題，在這之中我們要學習如何面對和解決各種障礙和挫折，我想這些解決問題的方式對於未來的工作是非常有用的經驗，整體來說專題對我而言可以說是有很大的成長從在各方面都由組內的成員們完成，在開始做專題之前完全無法想像我們能夠完成一個系統，從規劃、設計到問卷調查甚至是設計海報、拍攝影片等都由我們一手包辦實在讓人感到不可思議而且很有成就感。

10656009 林哲卉

在專題開始做程式撰寫的時候，我就遇到了之前完全沒有接觸的程式語言 flutter，因為沒學過，第一次寫出來的程式寫得很爛，很多都寫不出來也不太會用，感到非常挫折，也因為我們前端比較晚開始寫，實際的進度比較慢一些，在二評時評審的評論不怎麼好，很難過很想就此放棄，之後組長跟我說了很多鼓勵的話，還有其他組員的安慰，我才有動力繼續研究下去，暑假這段期間大家互相幫忙交流技術，我也把 flutter 基本用法學會了，大部分的程式撰寫工作也完成了，把之前緩慢的進度趕回來，我在專題中收穫滿滿，學會了新技術以及團隊合作的能力還有溝通技巧等等，很感謝有可靠的組長跟組員們，還有宏仁老師適時的指導，才能完成這次專題。

10656025 林欣儀

做了一年半的專題，讓我更了解到一個專案的流程，並且從中學習技術，從剛開始的確定主題、設計、寫程式、整合後的修正，再到後來的完成，這些都是我在專題才能完完整整知道的過程，更知道了一個團隊合作的重要性，並且要按部就班，這樣才能把一件事情做好。感謝組員互相幫忙以及老師的協助，我們才能順利完成這個專題。

10656035 吳乙萱

在這為期一年半的專題中，在這段時間裡，讓我成長了很多。過程中遇到任何問題都不能像以前一樣什麼事都問老師，需要學習自己找答案自己嘗試，從失敗中吸取經驗，還學習到了如何團隊合作，增強溝通技巧和表達能力，規劃自己的時間，順利完成專題，這些都是寶貴的經驗。而我也要感謝我的隊友，幫助我解決問題，加快我自學速度，一起奮鬥，一同完成這次的專題，也感謝指導老師林宏仁老師的指導，給予我們的鼓勵，還有在這次專題中幫助我們的老師，謝謝老師。

第14章 參考資料

- Django 官方文件

<https://docs.djangoproject.com/en/3.2/>

- Django rest framework 官方網站

<https://www.django-rest-framework.org/>

- Flutter 官方文件

<https://flutter.dev/docs>

- Flutter---Google 推出的跨平台框架，Android、iOS 一起搞定

<https://ithelp.ithome.com.tw/users/20119550/ironman/2221>

- 從零開始學 Flutter 程式設計

<https://tw-hkt.blogspot.com/2019/07/2019-flutter-714.html>

- Flutter 參考文件

<https://api.flutter.dev/index.html>

附錄

附錄一---一評評審建議事項之修正情形

評審建議事項	修正情形
1. 課表資料介接需確認可行性。	目前沒有介接學校系統的規劃，因為我們的目標客群是所有學生，而非校內學生，若未來有機會可考慮與多數的學校介接。
2. 若課表為特色功能，可獨立出來。	我們 App 的特色並非「課表」，而是「以課表為時間軸的行事曆」。
3. 除課表、社交活動的規劃。	功能以行程為主，已將多餘的功能簡化。
4. Flutter 的介面設計。	介面已完成。
5. 目標族群、教師、職員為何會使用？	我們在經過討論之後將目標族群範圍縮小至大專生，教職員可以輔助的角色做使用。
6. 如何增加使用者的接受度？使用前要登入，會降底接受度。	多數的 app 皆需登入為了配合不同的使用者提供更好的服務，因此維持原本登入的決定。
7. 開發能量的評估，建議以單一平台為主，進行 POC。大部份的功能應用，以行動裝置為主。	經過評估我們已將網頁版的功能移除。

8. 功能應該要聚焦，目前感覺很雜亂； 且功能太理所當然，為什麼要全部放在一起？	我們的功能是以「行程」為主軸做擴展的，已經將多餘的功能刪除。															
9. 題目重覆性很高，和其他同校系學生撞題目。	根據我們的研究，目前市面上並沒有結合課表的行事曆。															
10. Dcard 就是一個留言功能為何客戶還是願意花整天時間在上面。	Dcard 與我們的系統完全不同，Dcard 為社群軟體，My Day 為時間管理軟體。															
11. 收費制不太可能實現，商業模式要想清楚。	收費制從來不是我們的考量，無論是一評的報告或是二評的報告中，我們都並未提到收費。															
12. 四技組專題「一訂行」完成度就很高，且與你們要做的都已達成。	<p>一訂行主要的功能是將學校的行事曆結合個人行事曆，與我們的系統完全不同，以下提供一些不同之處作為參考：</p> <p>My Day 與 一訂行 比較</p> <table><tr><td>差異</td><td>My Day</td><td>一訂行</td></tr><tr><td>平台</td><td>行動裝置</td><td>網頁</td></tr><tr><td>結合內容</td><td>個人課表</td><td>學校行事曆</td></tr><tr><td>主要客群</td><td>所有大專生</td><td>本校學生</td></tr><tr><td>主要功能</td><td>記錄行程、 規劃時間</td><td>整合</td></tr></table>	差異	My Day	一訂行	平台	行動裝置	網頁	結合內容	個人課表	學校行事曆	主要客群	所有大專生	本校學生	主要功能	記錄行程、 規劃時間	整合
差異	My Day	一訂行														
平台	行動裝置	網頁														
結合內容	個人課表	學校行事曆														
主要客群	所有大專生	本校學生														
主要功能	記錄行程、 規劃時間	整合														

	群組功能	有	無
	投票功能	有	無
	提醒功能	有	無

附錄二---二評評審建議事項之修正情形

評審建議事項	修正情形
1. 文件：應有附錄，附上次評分老師建議。	此次有附上相關內容。
2. 專案進行要適當規劃進度，此次無看到任何功能是前、後端已連接。	已完成功能。
3. 「共同學習」是否可進行好友之間的協同合作？類似 github、google document，例如一份筆記內容，幾個好友一起閱讀並註記，如何使用本系統進行筆記版本控制？	因為我們是行事曆會著重於行程功能，在筆記上沒有過多的功能避免失焦。
4. 因為系統功能較多，可使用中段過程 USER 測試，及早了解 USER 對於系統功能面、系統使用介面等問題。	有根據評審意見對使用者進行調查。
5. 但在系統界面與呈現上宜活潑有趣好玩，才能吸引使用者使用。	已為介面加上更豐富的 icon
6. 依據目前展示，系統宜考量： a. 留意使用者入群隱私(為什麼他或她都來這裡找我，很困擾) b. 較活潑與美觀的介面	a. 此項目在文件內皆有說明，在隱私的部分使用者都能自行設定。 b. 同第 5 點回覆 c. 我們透過對使用者的調查來改進

c. 強化使用者持續實用的設計	
7. 與你們更相似的一個課表 App 叫 Colorgy 可以研究看看	根據調查此產品已下架。
8. Prototype 看起來不是很易用	同第 5 點及第 6 點 c 回覆。