

國立中央大學  
資 訊 管 理 學 系  
系 統 分 析 與 設 計  
系統軟體需求規格書

**“BrainLib”**

第22組

資工二B	111403525	陳昱翔
資管二B	111403538	黃鉅燊
資管三A	110403601	林俊仰
企管三	110401520	王長楨
中文四	109101503	何雅智

指導教授：許智誠 教授

## 第2章 資料庫設計

設計階段之資料庫，根據分析文件之實體關係圖 (Entity-Relation Diagram)，進行確認並依據其規劃資料庫之資料表，共計包含2個強實體 (Strong Entity)、4個弱實體 (Weak Entity)、13個關係 (Relationship)、4個關聯實體 (Associate Entity)，下圖 (圖 1) 為設計階段之ER圖，亦可使用資料庫綱要圖 (Schema Diagram) 進行取代：

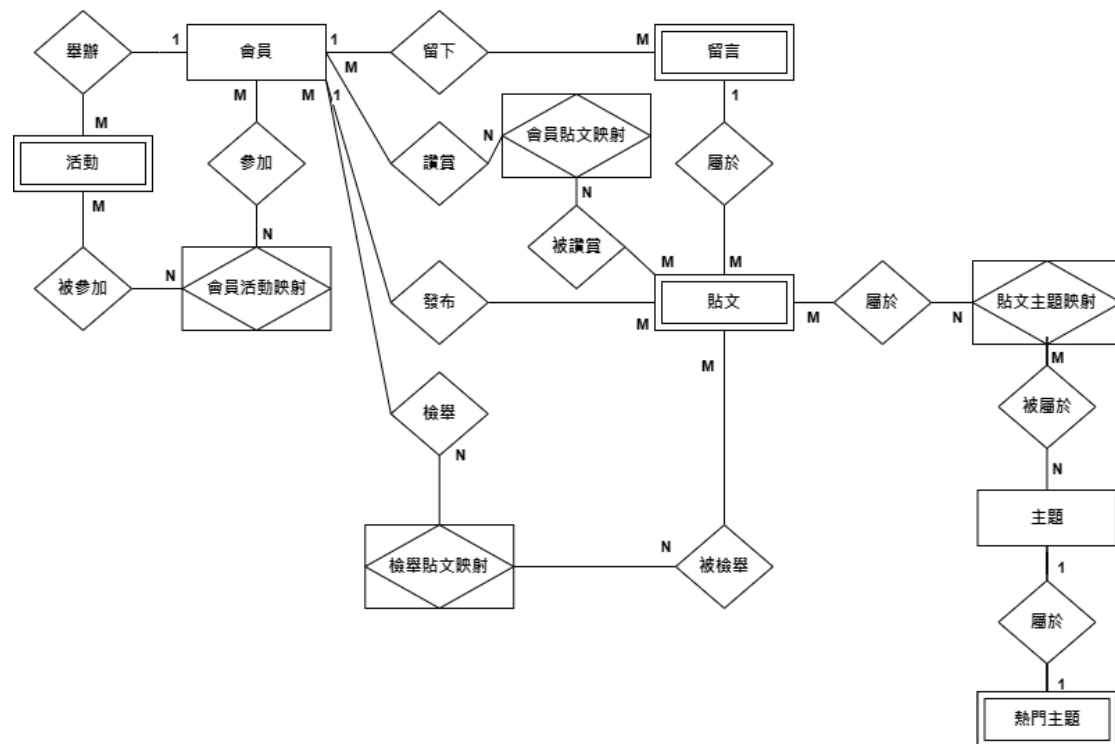


圖 1：設計階段之實體關係圖

根據上圖進行資料表之設計，以下將逐一說明資料庫每張資料表之欄位：

## 1. 會員資料表 (users)：

表 1：會員資料表 (users) 之資料結構

Key	名稱	類型	預設值	空值	自動增加	編碼
P.K	id	Int(11)	無	否	V	
	name	Text	無	否		utf8mb4_0900_ai_ci
	email	Text	無	否		utf8mb4_0900_ai_ci
	permission	Int(1)	無	否		
	profile	Text	無	是		utf8mb4_0900_ai_ci
	image	Text	無	否		utf8mb4_0900_ai_ci

- ✓ **id**: 為自動增加作為會員編號，不可更動由資料庫系統自動產生。
- ✓ **name**: 用於紀錄會員名稱。
- ✓ **email**: 用於紀錄會員電子郵件。
- ✓ **permission**: 用於代表使用者身分，內容為 0, 1, 2，分別代表會員、版主、管理員。
- ✓ **profile**: 用於記錄會員個人介紹。
- ✓ **image**: 用於紀錄會員大頭貼圖像連結。

## 2. 貼文資料表 (posts)：

表 2：貼文資料表 (posts) 之資料結構

Key	名稱	類型	預設值	空值	自動增加	編碼
P.K	id	Int(11)	無	否	V	
F.K	user_id	Int(11)	無	否		
	title	Text	無	否		utf8mb4_0900_ai_ci
	content	Text	無	否		
	image	Text	無	否		utf8mb4_0900_ai_ci
	thumb_up	Int(11)	無	否		
	visible	Boolean	false	否		
	is_suspend	Boolean	false	否		

- ✓ **id**: 為自動增加作為貼文編號，不可更動由資料庫系統自動產生。
- ✓ **user\_id**: 用於紀錄貼文發布者的ID。
- ✓ **title**: 用於記錄貼文的標題。
- ✓ **content**: 用於紀錄貼文內容。

- ✓ **image**: 用於紀錄貼文的封面圖。
- ✓ **thumb\_up**: 用於紀錄貼文的讚賞數。
- ✓ **visible**: 為 **true** 或 **false**, 用於紀錄使用者是否公開貼文。
- ✓ **is\_suspend**: 為 **true** 或 **false**, 用於紀錄貼文是否被檢舉。

### 3. 檢舉貼文資料表 (**susposts**) :

表 3：貼文資料表 (**posts**) 之資料結構

Key	名稱	類型	預設值	空值	自動增加	編碼
P.K	id	Int(11)	無	否	V	
F.K	post_id	Int(11)	無	否		
F.K	user_id	Int(11)	無	否		
	is_reviewed	Boolean	false	否		
	reason	Text	無	否		utf8mb4_0900_ai_ci

- ✓ **id**: 為自動增加作為檢舉編號, 不可更動由資料庫系統自動產生。
- ✓ **post\_id**: 用於紀錄貼文的ID。
- ✓ **user\_id**: 用於紀錄貼文發布者的 ID。
- ✓ **is\_reviewed**: 為 **true** 或 **false**, 用於紀錄貼文是否被審核。

### 4. 留言資料表 (**comments**) :

表 4：留言資料表 (**comments**) 之資料結構

Key	名稱	類型	預設值	空值	自動增加	編碼
P.K	id	Int(11)	無	否	V	
F.K	post_id	Int(11)	無	否		
F.K	user_id	Int(11)	無	否		
	content	Text	無	否		
	created_at	Text	無	否		utf8mb4_0900_ai_ci

- ✓ **id**: 為自動增加作為留言編號, 不可更動由資料庫系統自動產生。
- ✓ **post\_id**: 用於紀錄貼文的ID。
- ✓ **user\_id**: 用於紀錄留言發布者的 ID。

- ✓ **content:** 用於紀錄留言內容。
- ✓ **created\_at:** 用於紀錄留言時間

#### 5. 活動資料表 (activities) :

表 5：活動資料表 (activities) 之資料結構

Key	名稱	類型	預設值	空值	自動增加	編碼
P.K	id	Int(11)	無	否	V	
F.K	user_id	Int(11)	無	否		
	title	Text	無	否		utf8mb4_0900_ai_ci
	content	Text	無	否		utf8mb4_0900_ai_ci
	time	Datetime	無	否		
	location	Text	無	否		utf8mb4_0900_ai_ci
	visible	Boolean	false	否		

- ✓ **id:** 為自動增加作為活動編號，不可更動由資料庫系統自動產生。
- ✓ **user\_id:** 用於紀錄活動舉辦者的 ID。
- ✓ **title:** 用於紀錄活動標題。
- ✓ **content:** 用於紀錄活動內容敘述。
- ✓ **time:** 用於紀錄活動舉辦時間。
- ✓ **location:** 用於紀錄活動舉辦地點。
- ✓ **visible:** 為 **true** 或 **false**，用於紀錄貼文是否為草稿狀態。

#### 6. 會員活動映射資料表 (Attender) :

表 6：會員活動映射資料表 (Attender) 之資料結構

Key	名稱	類型	預設值	空值	自動增加	編碼
P.K	id	Int(11)	無	否	V	
F.K	activity_id	Int(11)	無	否		
F.K	user_id	Int(11)	無	否		

- ✓ **id:** 為自動增加作為映射編號，不可更動由資料庫系統自動產生。
- ✓ **activity\_id:** 用於紀錄活動的ID。
- ✓ **user\_id:** 用於紀錄活動參與者的 ID。

#### 7. 主題資料表 (tags) :

表 9：主題資料表 (tags) 之資料結構

Key	名稱	類型	預設值	空值	自動增加	編碼
P.K	id	Int(11)	無	否	V	
	name	Text	無	否		utf8mb4_0900_ai_ci

✓ **id**：為自動增加作為主題編號，不可更動由資料庫系統自動產生。

✓ **name**：用於紀錄的主題的名稱。

#### 8. 貼文主題映射資料表 (PostTag)：

表 10：貼文主題映射資料表 (PostTag) 之資料結構

Key	名稱	類型	預設值	空值	自動增加	編碼
P.K	id	Int(11)	無	否	V	
F.K	post_id	Int(11)	無	否		
F.K	tag_id	Int(11)	無	否		

✓ **id**：為自動增加作為主題編號，不可更動由資料庫系統自動產生。

✓ **post\_id**：用於紀錄的貼文的ID。

✓ **tag\_id**：用於紀錄主題的ID。

#### 9. 熱門主題資料表 (trendTags)：

表 11：熱門主題資料表 (trendTags) 之資料結構

Key	名稱	類型	預設值	空值	自動增加	編碼
P.K	id	Int(11)	無	否	V	
F.K	tag_id	Int(11)	無	否		

✓ **id**：為自動增加作為熱門主題編號，不可更動由資料庫系統自動產生。

✓ **tag\_id**：用於紀錄主題的ID

#### 10. 會員貼文映射資料表 (Appreciator)：

表 12：會員貼文映射資料表 (Appreciator) 之資料結構

Key	名稱	類型	預設值	空值	自動增加	編碼
P.K	id	Int(11)	無	否	V	
F.K	user_id	Int(11)	無	否		
F.K	post_id	Int(11)	無	否		

- ✓ **id**：為自動增加作為熱門主題編號，不可更動由資料庫系統自動產生。
- ✓ **user\_id**：用於紀錄讚賞者的ID
- ✓ **post\_id**：用於紀錄貼文的ID

## 第3章 類別圖

下圖（圖 2）係依據BrainLib的分析模型和建立的互動圖，以及實體關係圖（Entity-Relation Diagram）所繪製之設計階段之類別圖（Class Diagram），用於描述系統的類別集合，包含其中之屬性，與類別之間的關係。

本階段之類別圖屬於細部（detail）之設計圖，與上一份文件分析階段之類別圖需要有詳細之變數型態、所擁有之方法，依據這些設計原則，本類別圖之說明如下所列：類別圖除包含與資料庫相對應之物件外，亦包含相關之控制物件（controller）、與各功能相對應資料庫操作類別（例如：UserService）。

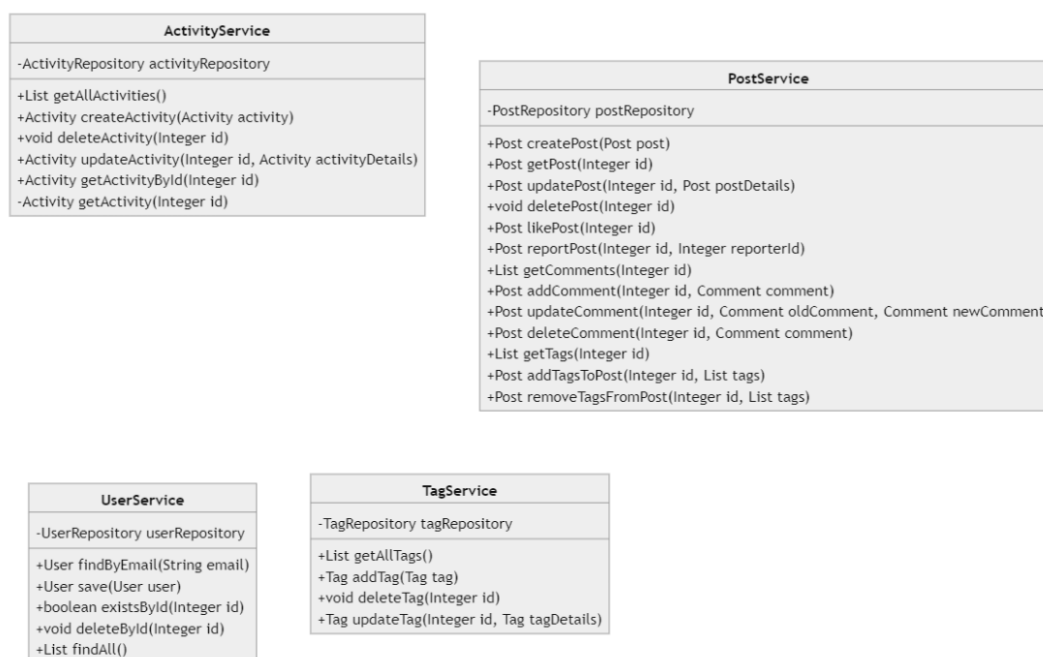


圖 2：類別圖 (1/3)

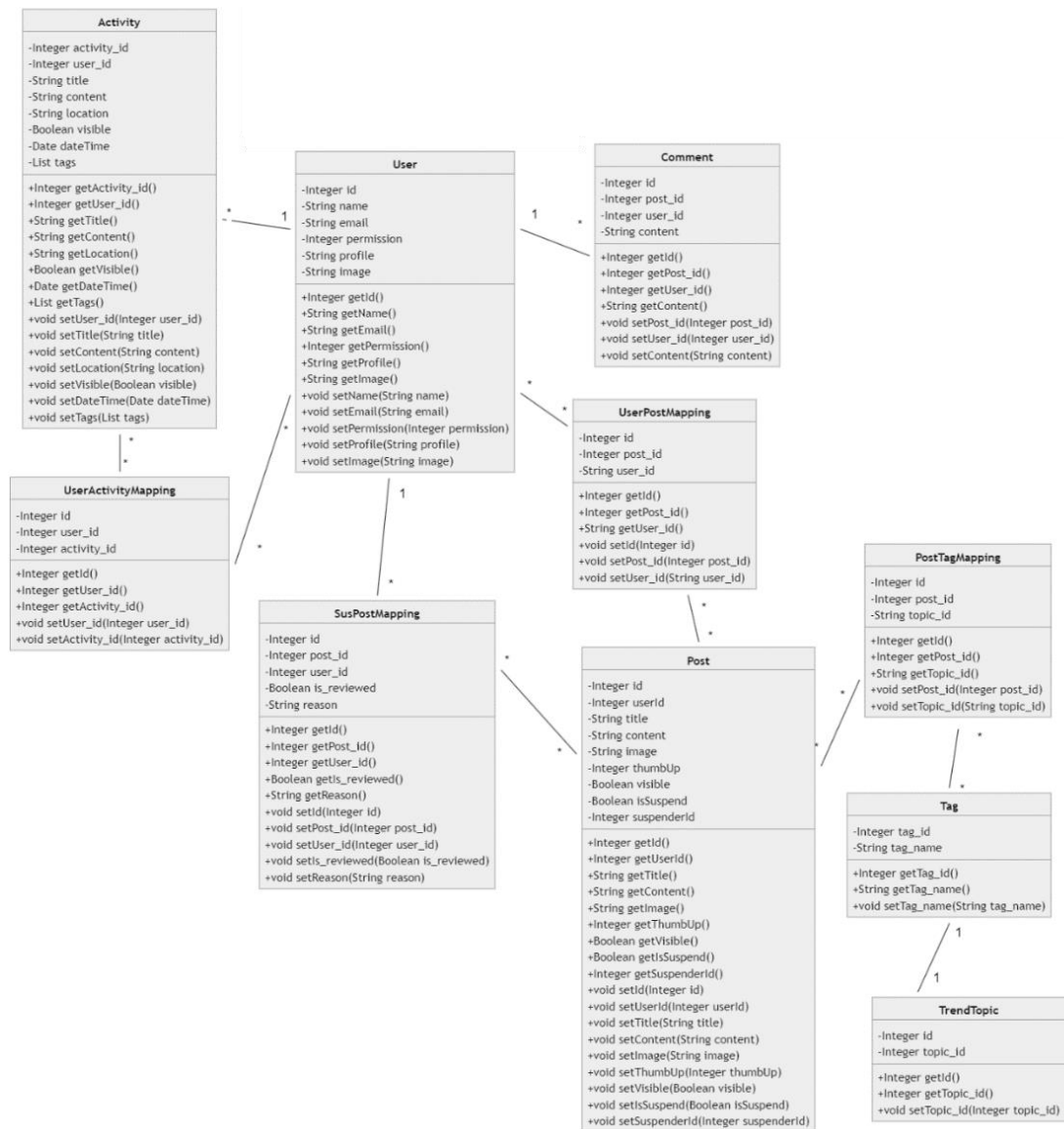


圖 3：類別圖 (2/3)



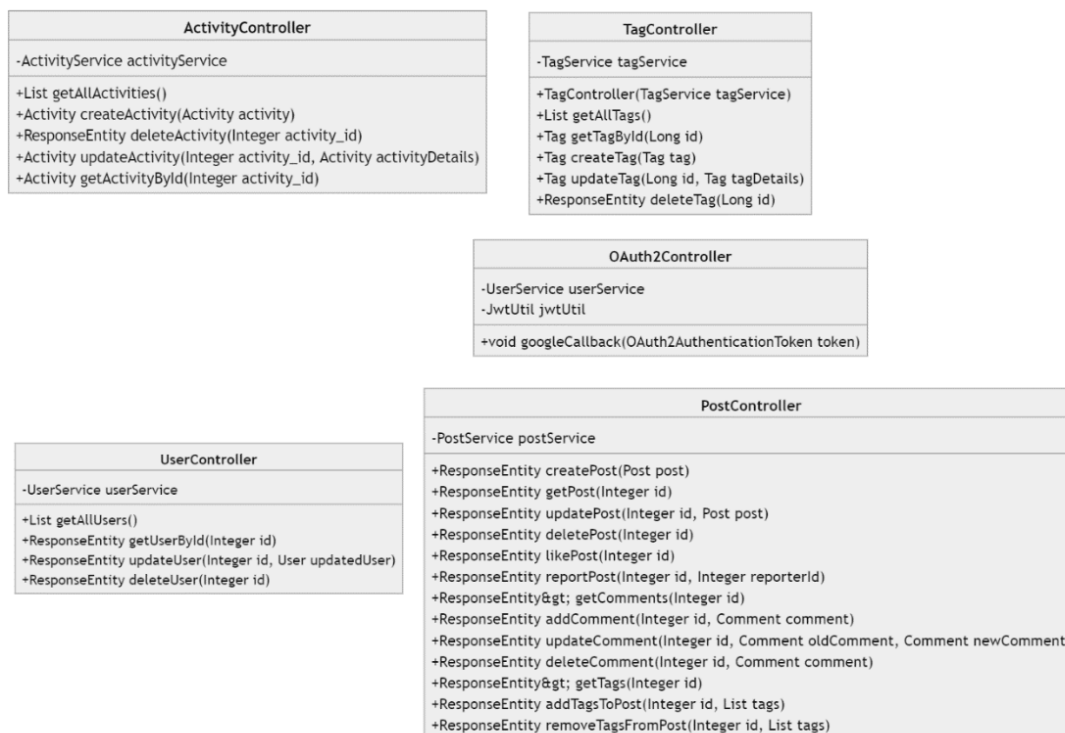


圖 4：類別圖 (3/3)

## 第4章 系統循序圖

本章節主要依照第一份文件需求所產生之使用案例圖 (use case) 與第二份文件分析之邏輯階段活動圖與強韌圖為基礎，進行設計階段之循序圖設計，將每個使用案例進行闡述。於此階段，需要有明確之類別 (class) 名稱與呼叫之方法 (method) 與傳入之變數名稱與型態等細部設計之內容。

### 4.1 使用案例圖

依據第一份文件針對專案之需求進行確定，BrainLib預計共有4位動作者與39個使用案例，並依照不同之模組區分成不同子系統共計七個子系統，其中包含以下：會員子系統、貼文子系統、貼文管理子系統、留言子系統、主題子系統、管理員子系統、活動子系統，如下圖 (圖五) 所示：

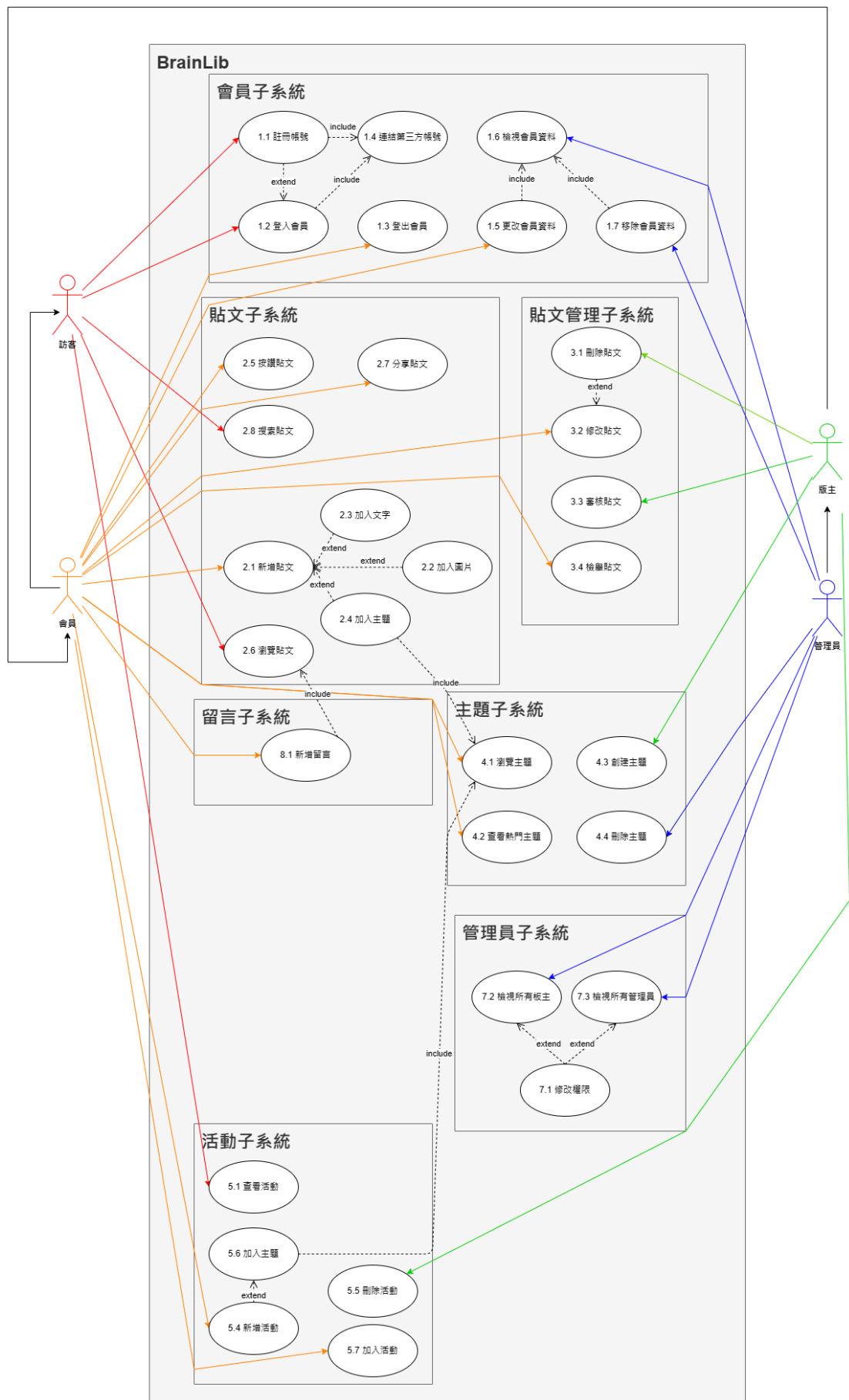


圖 3：BrainLib使用案例圖

## 4.2 Use Case實做之循序圖

### 4.2.1 商業流程編號1.0：會員模組

在Use Case 1.0中所使用到的功能，以下僅針對檢視會員資料及移除會員資料做進行詳細說明：

#### 1. Use Case 1.6：檢視會員資料

此使用案例用於檢視資料庫內存有之所有會員，本使用案例僅限擁有管理員身分權限之用戶才得以執行。

#### 2. Use Case 1.7：移除會員資料

此使用案例用於刪除資料庫內存有之會員，完成使用案例1.6檢視會員資料後，點擊欲異動之會員，即可進行刪除，本使用案例僅限擁有管理員身分權限之用戶才得以執行。

與該模組相關之頁面於下表（表 13）進行說明：

表 13：會員模組關聯頁面

HTML	關聯Use Case	說明
userManager.vue	Use Case 1.6 Use Case 1.7	會員資料管理頁面

#### 4.2.1.1 Sequence Diagram—Use Case 1.6檢視會員資料（管理者）

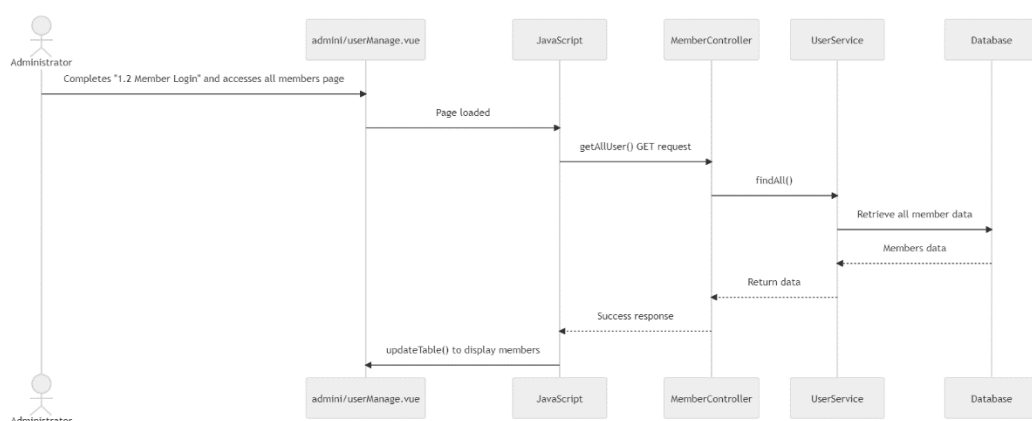


圖 4：商業流程編號1.6檢視會員資料（管理者）循序圖

1. 管理員完成商業流程編號「1.2登入會員」後，點擊進入檢視所有會員頁面

- (admini/userManage.vue)。
- 針對所欲刪除之會員點擊刪除之按鈕，JavaScript之deleteUser()發送DELETE之請求。
  - 後端以UserController之deleteUser()進行處理，透過userService物件的deleteById()方法將資料庫之該名會員進行刪除。
  - 若刪除成功則透過JavaScript之getAllUser()再次執行商業流程編號「1.6檢視會員資料」之循序圖過程。

#### 4.2.1.2 Sequence Diagram—Use Case 1.7移除會員資料（管理者）

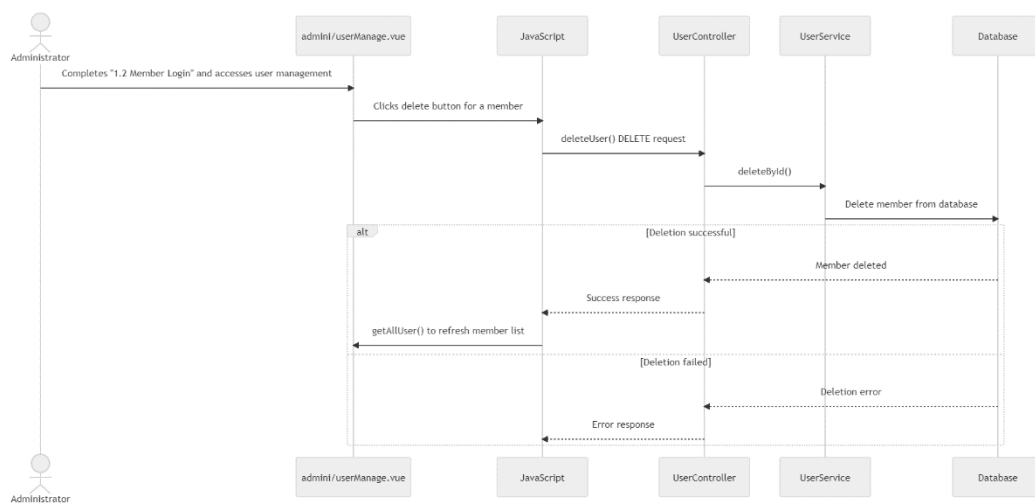


圖 7：商業流程編號1.7移除會員資料（管理者）循序圖

- 管理員完成商業流程編號「1.2 會員登入」後，點擊進入檢視所有會員頁面 (admini/userManage.vue)。
- 進入頁面後，透過JavaScript之getAllUser()送出GET請求。
- 後端以MemberController之getAllUsers()進行處理，使用userService物件的findAll()方法直接取得所有會員之資料回傳給前端。(ORM)
- 查詢成功則透過JavaScript之updateTable()更新會員列表之表格。