

台大電資校友牆網站規劃

系統規格

VERSION 0.1.0

2023.06.09

Contents

CONTENTS	1
TABLES	3
FIGURES	3
REVISION HISTORY	4
1 簡介	5
1.1 需求規劃	5
1.1.1 電視牆呈現規劃	5
1.1.2 後台資料建立規劃	5
1.2 採用系統相關技術	6
1.2.1 資料庫規劃	6
1.2.2 前端技術建議	6
1.2.3 後端技術建議	6
2 系統架構	8
2.1 硬體架構	8
2.1.1 資料維護後台主機	8
2.1.2 電視牆前端主機	9
2.1.3 電視牆	9
2.2 軟體規格	10
2.2.1 資料維護後台主機	10
2.2.2 電視牆前端主機	10
2.2.3 開發使用相關軟體	10
2.2.3.1 電視牆前端呈現相關軟體	10
2.2.3.2 電視牆後端資料維護相關軟體	10
3 功能需求說明	11
3.1 電視牆前端畫面呈現	11
3.1.1 傑出校友	11
3.1.2 各級畢業校友	11
3.1.3 研究亮點	11
3.2 資料維護後台主機	12
3.2.1 資料維護後台主機	12
3.3 系統安裝程序	12
3.4 系統維護資料	12
3.5 系統備份	12



3.6	資料安全與保護	12
4	資料庫規劃	13
4.1	USER SCHEMA	13
4.2	CONTENT SCHEMA	14
4.3	LOG SCHEMA	16
5	流程及畫面示意	18
5.1	SITEMAP	18
6	專案時程	20

Tables

Table 1 : User Schema.....	13
Table 2 : Content Schema	15
Table 3 : Log Schema.....	16

Figures

Figure 1: 硬體架構示意圖	8
Figure 2: 電視牆設計示意圖	9
Figure 3: 電視牆實景	9



Revision History

Version	Date	Description
V 0.1.0	2023.06.09	Creation

1 簡介

本專案電視牆展示的主要目的：在於加強校友與學校的連結，建立更持續長久的關係，希望校友校友在畢業後的經歷，可以帶入新的資源進入學校與學院，也可以提供與在校生分享畢業後的經歷與發展，提供在校生生涯發展的參考和典範。

本文件說明台大電資校友牆網站相關系統規劃，包含內容需求、系統架構、資料建立流程、畫面、後台權限控管及系統資料架構。

1.1 需求規劃

本系統流程規劃依據需求面主要分成二大部份，第一部份為前端電視牆呈現規劃，第二部份為後台資料建立系統規劃。

1.1.1 電視牆呈現規劃

電視牆展示主要針對我們學校的優秀校友，以及他們在校期間的學習歷程，以及之後在各自領域的發展與成就。本專案的內容規劃包含三大項目，包括「傑出校友」、「各級畢業校友」和「研究亮點」。

在「傑出校友」部分，我們將展示歷年來傑出校友，他們在學習歷程中表現卓越，並在各自的領域中取得了重要的成就。這些校友的故事將激勵現在的學生，鼓舞他們努力學習，追求自己的夢想。

「各級畢業校友」部分將展示歷年每一級的畢業生在學校生活中的成長點滴和努力，這部分將突顯他們在學術、社交和領導各方面的成就，並呈現他們在學校的經歷及回憶；藉由校友們的建樹，展示我們學校的教育品質和培養學生全面發展的能力。

最後，「研究亮點」部分將學校在資電相關各個領域的研究成果和創新項目，例如：學校教職員和學生們在專業領域的重要突破和貢獻，研究亮點將展現學校作為一個知識創新的中心的地位及專業能力和影響力。

1.1.2 後台資料建立規劃

為了實現前端資料介面，以便校友們能夠貢獻他們珍藏的相關資料和影音，後台資料建立需要進行以下規劃：

- 資料分類：確定資料的類型和分類方式，例如根據年度、校友身份、資料類型等。
- 介面設計：設計一個直觀且易於使用的上稿介面，使校友能夠方便地上傳和貢獻他們的資料和影音。這可能包括上傳照片、影音檔案、文章及填寫說明文字等。

- 資料驗證與審核：建立一個驗證機制，確保上傳的資料符合特定要求或品質標準。由相關人員或管理員進行審核，以確保資料的正確性和可信度。
- 資料儲存和管理：建立一個有效的資料儲存和管理系統，包括建立適當的資料庫架構和文件結構，以便將上傳的資料組織和儲存起來。
- 隱私和安全考量：確保校友上傳的資料受到適當的隱私和安全保護。這可以包括使用安全的儲存和傳輸協議、設置適當的使用者權限和隱私選項等。
- 資料更新和維護：確保資料庫中的資料得到定期的更新和維護，包括移除過時的資料、修復損壞的連結等。

1.2 採用系統相關技術

1.2.1 資料庫規劃

本系統採用非關聯式資料庫架構，於前期作業時，規劃相關的需求資料欄位，非關聯式資料庫可針對未來特殊需求提供部份彈性，增加特殊需要的資料，採用 JSON 格式，存放於系統檔案中。

非關聯式資料庫，評估採用 MongoDB，是一個開源的文件導向型(document-oriented)資料庫管理系統。它使用 JSON 風格的文件來儲存資料，並且支援動態的架構，可以輕鬆地擴展和修改資料結構。

1.2.2 前端技術建議

本系統主要以展示校友及研究成果為主，且置於戶外展示平台，人機介面的操控宜以便利方便為主，與使用者的互動介面，擬採簡潔清楚方式，少了傳統的鍵盤滑鼠互動方式，因此採用的前端技術宜為富有彈性、獨立性及專門性的前端 UI 技術。

評估採用目前前端廣泛被採用的 React 技術，React 是一個開源的 JavaScript 函式庫，用於建立使用者介面。它由 Facebook 開發，並且已經成為現代前端開發中非常受歡迎的選擇之一。

React 提供了一種稱為「組件化」的方式來建構使用者介面。開發者可以將整個應用程式拆分為小型、可重用的組件，每個組件都有自己的狀態和生命週期方法。透過這種組件化的架構，可以更容易地管理應用程式的複雜性，並且提高開發效率。

1.2.3 後端技術建議

配合前端的使用技術，後端的資料上傳維護，亦將簡化傳統複雜程序，擬採用 node.js，一樣配合 Javascript，統一與前端開發技術的一致性，降低工程人員的學習門檻。

Node.js 是一個基於 JavaScript 引擎的開源、跨平台的運行環境。它允許你使用 JavaScript 開發伺服器端應用程式，並且具有非阻塞 I/O 和事件驅動的特性，使得它能夠處理高效率的並發請求。

Node.js 快速和輕量級特性，底層使用了 Chrome V8 引擎，它是一個高效的 JavaScript 引擎。這使得 Node.js 應用程式運行速度快且佔用系統資源少。

2 系統架構

2.1 硬體架構

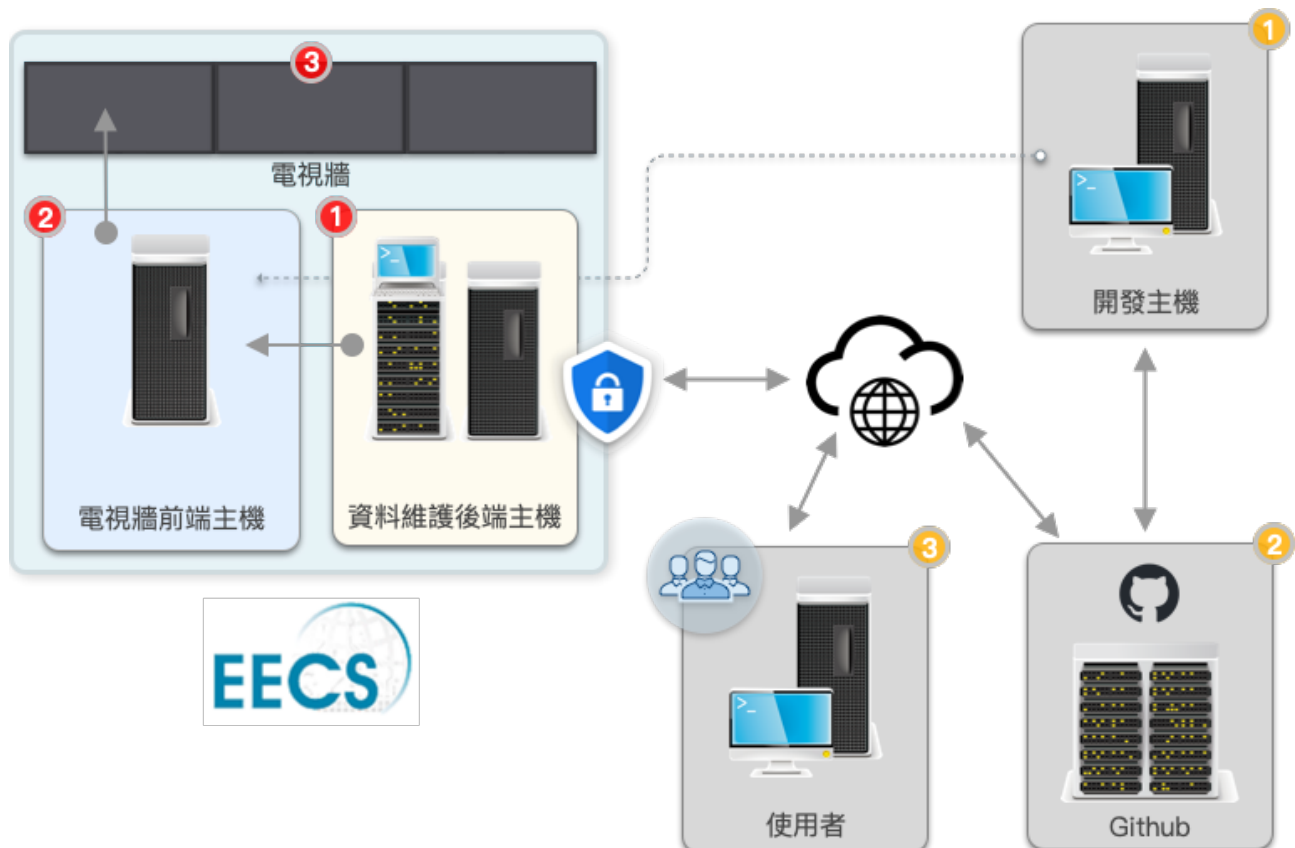


Figure 1: 硬體架構示意圖

本系統硬體主要架構分成二大部份，一部份實際運作時需要的電資學院中架設的主機，第二部份則是程式開發平台(黃 1)及用以版本控管及未來佈建營運系統使用的第三方 Github 主機(黃 2)，實際運作時，則包含校友們上傳資料的個人主機(黃 3)。

第一部份分成三個項目，如圖 Figure 1: 硬體架構示意圖，紅 1：資料維護後端主機，紅 2 電視牆前端主機，紅 3 本案呈現主要畫面的電視牆。

本系統僅有資料維護主機(紅 1)有對外網路，可供外部使用者連線更新上傳校友資料，前端主機採用區域內網形式，不對外展示，展示內容只提供給學院中的電視牆。

2.1.1 資料維護後台主機

本系統採用 Linux 主機，硬體需求如下：

2.1.2 電視牆前端主機

本系統採用 Linux 主機，硬體需求如下：

2.1.3 電視牆

電視牆座落在學新館(校友捐贈的館舍)最顯眼與核心的空間，初期展示傑出校友資料，在台大求學或是重要活動的數位影音資料。

電視牆由三片 65 吋的螢幕組成，其中只有中間的一片可以提供多點觸控，其餘左右二片僅供展示如下圖：



Figure 2: 電視牆設計示意圖



Figure 3: 電視牆實景

電視牆規格如下：

SAMSUNG QM65B/4K 商用顯示器多點觸控大屏 65 吋

- 解析度：3840 x 2160
- 水平掃描頻率：30~81 kHz
- 最大像素頻率：594 MHz
- 垂直掃描頻率：48~75 Hz
- HDMI 輸入：3
- DP 輸入 X 1，版本 1.2

- HDMI 版本 X 2
- HDCP 版本 2.2
- USB X 2
- RS232：輸入/輸出
- 通訊介面：RJ45 及 WiFi
- 電源
 - 電源供應：AC100-240V 50/60Hz
 - 耗電量 (工作模式)：182 W
 - 耗電量 (睡眠模式)：0.5 W
- 尺寸
 - 機身尺寸 (寬 x 高 x 深)1453.9 x 831 x 46.3 mm
 - 包裝尺寸 (寬 x 高 x 深)1611 x 973 x 185 mm
- 重量
 - 機身重量 24.9 kg
 - 包裝重量 33.3 kg

2.2 軟體規格

2.2.1 資料維護後台主機

本主機採用的 OS 版本是

2.2.2 電視牆前端主機

本主機採用的系統是 Linux，OS 版本
SSL 版本

2.2.3 開發使用相關軟體

2.2.3.1 電視牆前端呈現相關軟體

- React
- MongoDB

2.2.3.2 電視牆後端資料維護相關軟體

- Node.js
- MongoDB

3 功能需求說明

本章節說明本規劃案中需要的各項功能需求，並說明需求的相關技術要求。

3.1 電視牆前端畫面呈現

電視牆展示內容包含三大項目，包括「傑出校友」、「各級畢業校友」和「研究亮點」。內容以影片、照片、音檔及文章為主。

3.1.1 傑出校友

歷屆傑出校友，將採條列式的列表方式，呈現校友的個人資料、學術成就及在社會上的傑出表現，目前可採用的資料包括：

- 使用台大為傑出校友拍攝的影音資料，計 16 隻影片(已經取得校友中心 資料與授權)。
- 李琳山教授數位典藏資料 (規劃八月上線)。

3.1.2 各級畢業校友

各級畢業校友可由 1943 年起算迄今，因年份過多，為利使用者操作查詢便利性，擬採十年分區方式顯示，以十年為分頁索引。

資料的呈現，可包含影片、照片、音檔及文章，特別針對在校的生活點滴及學術發展表示，展示校友在學成果，用以激勵後進學弟妹們學習方向。

資料呈現部份可分成：

- 十年為一分頁索引
- 每一個十年內，再分該十年中各個年級
- 各年級可依內容的屬性不同，保留不同排列方式的彈性
- 素材的來源有二大方向
 - A. 學院中典藏畢業紀念冊及相關照片文檔
 - B. 由歷屆校友上傳公開相關資料

3.1.3 研究亮點

TBD

3.2 資料維護後台主機

3.2.1 資料維護後台主機

3.3 系統安裝程序

3.4 系統維護資料

3.5 系統備份

3.6 資料安全與保護

4 資料庫規劃

本資料規劃採用非關聯式 MongoDB。

4.1 User Schema

■ JSON 格式：

```
const userSchema = new Schema({
  username: { type: String, required: true },
  password: { type: String, required: true },
  name: { type: String, required: true },
  email: { type: String, required: true },
  phone: String,
  permission: { type: Number, required: true }, //0: admin, 1: reviewer, 2: editor...
});
```

編號	欄位名稱	欄位說明	型態	主鍵	必要	備註
1	username	使用者帳號	String	Y	Y	
2	password	密碼	String		Y	長度 10 字元以上，
3	name	使用者名稱	String		Y	
4	email	電子郵件	String	Y	Y	
5	phone	電話號碼	String			
6	permission	權限	Number		Y	0: admin, 1: reviewer, 2: editor

Table 1 : User Schema

- A. 本資料，系統由 admin 使用者建立。
- B. 密碼部份需經過加密。
- C. 使用者帳號不得重覆。

4.2 Content Schema

■ JSON 格式：

```
const contentSchema = new Schema({
  author: { type: String, required: true },
  postId: { type: Number, required: true },
  title: { type: String, required: true },
  description: String,
  media: [{
    id: { type: Number, required: true },
    url: { type: String, required: true },
    filename: { type: String, required: true },
    extension: { type: String, required: true }, //png, jpg, gif, mp4....
    order: { type: Number, required: false },
  }], //array of media
  category: [String],
    //year of graduation, industry, gender, and other tags to sort posts
  contentType: { type: Number, required: true },
    //0: text only, 1: image only, 2: video only, 3: text and image, etc
  layout: { type: Number, required: true }, //layout id for different formats
  created: { type: Date, required: true },
  updated: { type: Date, default: Date.now, required: true },
  approved: { type: Boolean, required: true },
});
```

編號	欄位名稱	欄位說明	型態	主鍵	必要	備註
1	author/name	作者名稱	String		Y	與登入使用者姓名相同
2	postId	文章編號	Number	P	Y	
3	Title	文章標題	String		Y	
4	description	內容描述	String		Y	
4.1	media	上傳媒體				
4.1.1	Id	媒體編號	Number		Y	
4.1.2	url	媒體 URL	String		Y	
4.1.3	filename	媒體檔案名稱	String		Y	下載後的檔案名稱
4.1.4	extension	媒體格式	String		Y	可以為影片、照片、音章或是聲音檔

						png, jpg, gif, mp4, pdf, text
4.1.5	order	媒體排列順序	Number			
5	category	類別	String		Y	文章分類，例如 2000 年
6	contentType	文章格式	Number		Y	0: text only, 1: image only, 2: video only, 3: text and image, etc
7	layout	排版格式	Number		Y	
8	created	建檔時間日期	Date		Y	
9	updated	更新時間日期	Date			
10	approved	審查通過狀態	Boolean Number		Y	0: wait for approved, initial value 1: approved 2: has issues and need owner to modify

Table 2 : Content Schema

- A. arthur 欄位必須與登入的使用者名稱相同。
- B. postid 必須唯一序號，不能重覆。
- C. 上傳的媒體資料 media，必須要有描述欄位，上傳的媒體數量可以為複數；例如：同一個時期的活動對映多個媒體資料。
- D. media 中的 url，因展示系統不對外連線，對映網址內容需要事先下載至本機端，以檔案(filename)方式存於資料格式中。
- E. media 中的 order(此格式為選項)，用以調整媒體內容的排序方式，管理員/審查員可調整權重。
- F. category 內容格式需統一，例如：以年代分，需明確規範為民國年或是西元年份，或是以畢業的年屆分，就需統一年屆的數字格式。
- G. contentType 是否和 media 中的 extension 重覆????
- H. layout 排版格式，配合前端的呈現方式樣版排版使用。
- I. created 建檔日期，為該筆資料第一次新增日期時間。
- J. updated 更新日期，為該筆資料任何被更動的日期時間。
- K. approved 審查結果狀態，例如：0：檔案建立等待審核，1:審核通過，2:有問題檔案，等待建檔的人員更新。
- L. 是否需要列出媒體版權出處?

4.3 Log Schema

本資料格式，用以記錄登入的使用者，在登入後的操作使用動作及規跡，任何資料異動的作業，均需記錄存檔，該資料僅供檢閱，不允許更動或是刪除。

記錄的內容包含：

- 任何使用者的登入時間、地點(IP)
- 管理員新增修改刪除使用者
- 管理員新增修改刪除文章
- 使用者(審查員，校友)異動文章內容
- 使用者(審查員，校友)異動個人資料
- 管理員系統備份還原

資料格式如下：

- JSON 格式：

```
const logSchema = new Schema({  
  username: { type: String, required: true },  
  postId: { type: Number, required: true },  
  actionid: { type: Number, required: true }, //0: edit, 1: delete, 2: archive....  
  comments: String,  
  time: { type: Date, default: Date.now, required: true },  
});
```

編號	欄位名稱	欄位說明	型態	主鍵	必要	備註
1	username	使用者帳號	String		Y	
2	postId	文章編號/	String			
3	actionid	執行動作	Number		Y	動作編號 0: login 1: create 2: update 3: delete 4:
4	comments	執行動作內容	String		Y	
5	time	紀錄時間	Date		Y	

Table 3 : Log Schmea

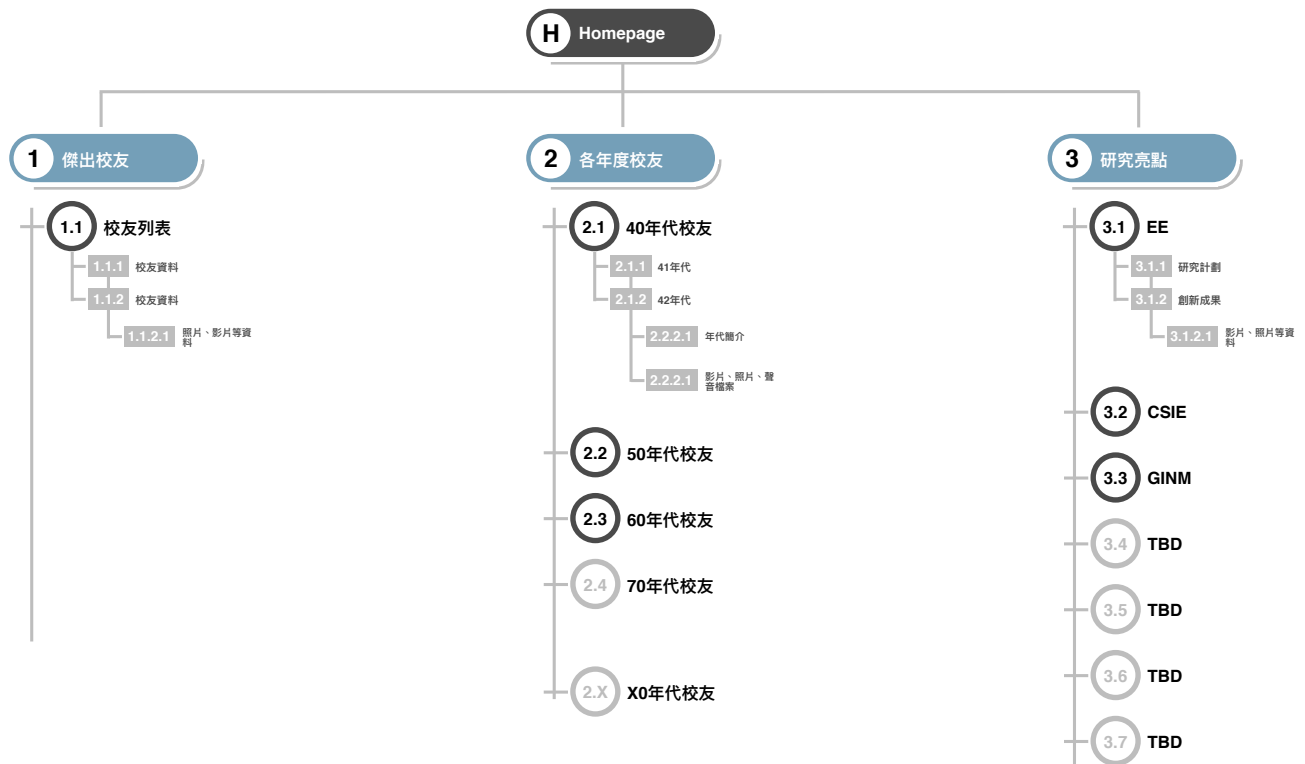
A. username，記錄當下登入的使用者，執行的動作。

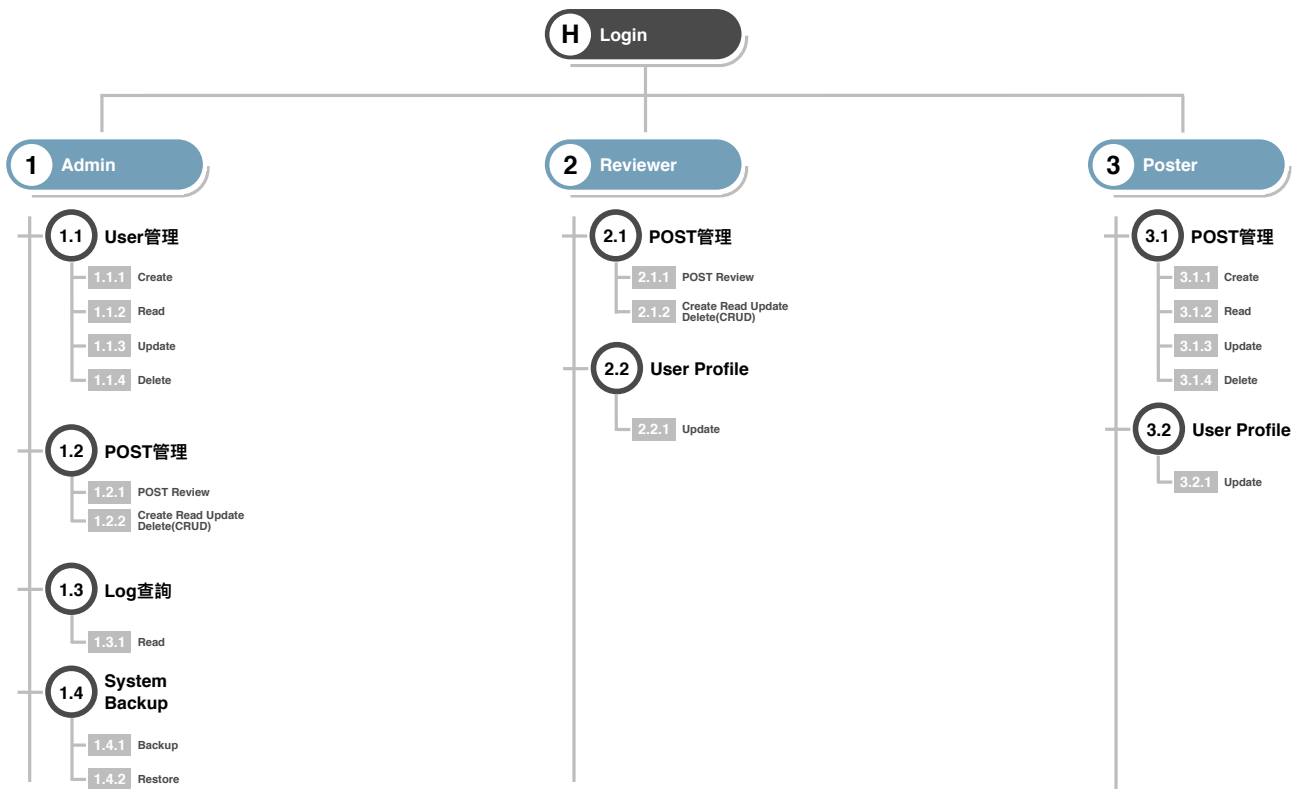
B. postId 記錄異動的文章編號，如果為其他模式，例如登入，此時 postId 可以設為"-1"。



5 流程及畫面示意

5.1 SiteMap







6 專案時程