

project-description-official

專案名稱: FreeScript

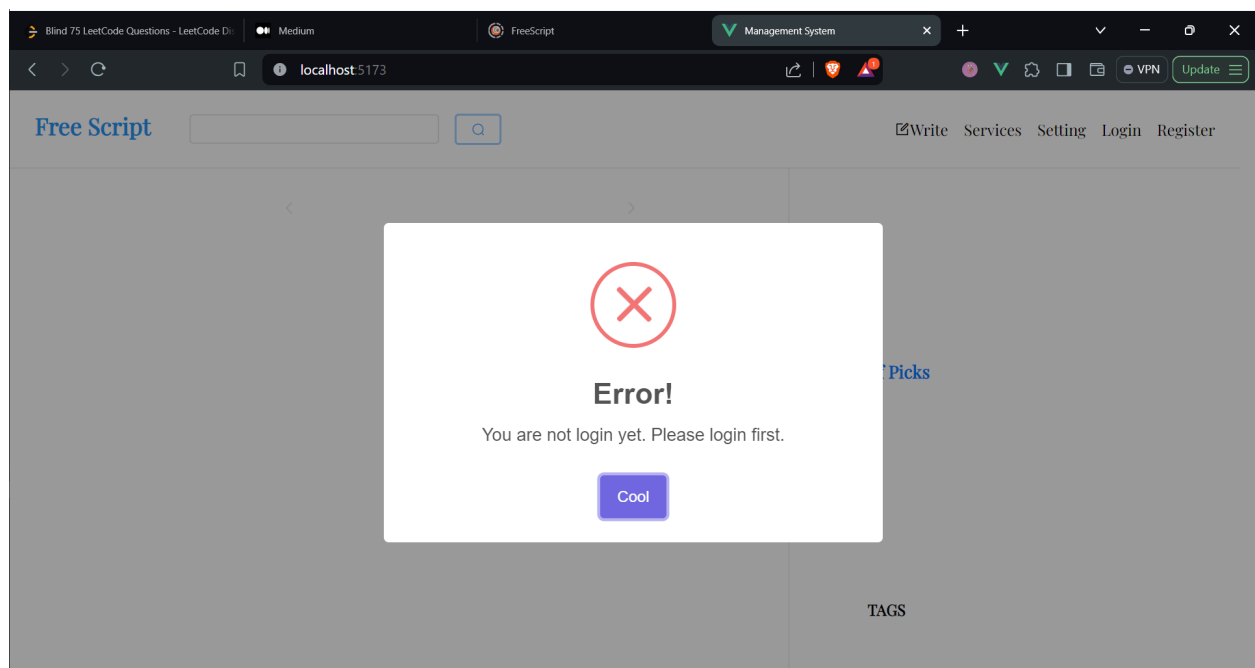
專案簡介: 以medium網站為base, 自行design的side project

技術內容:

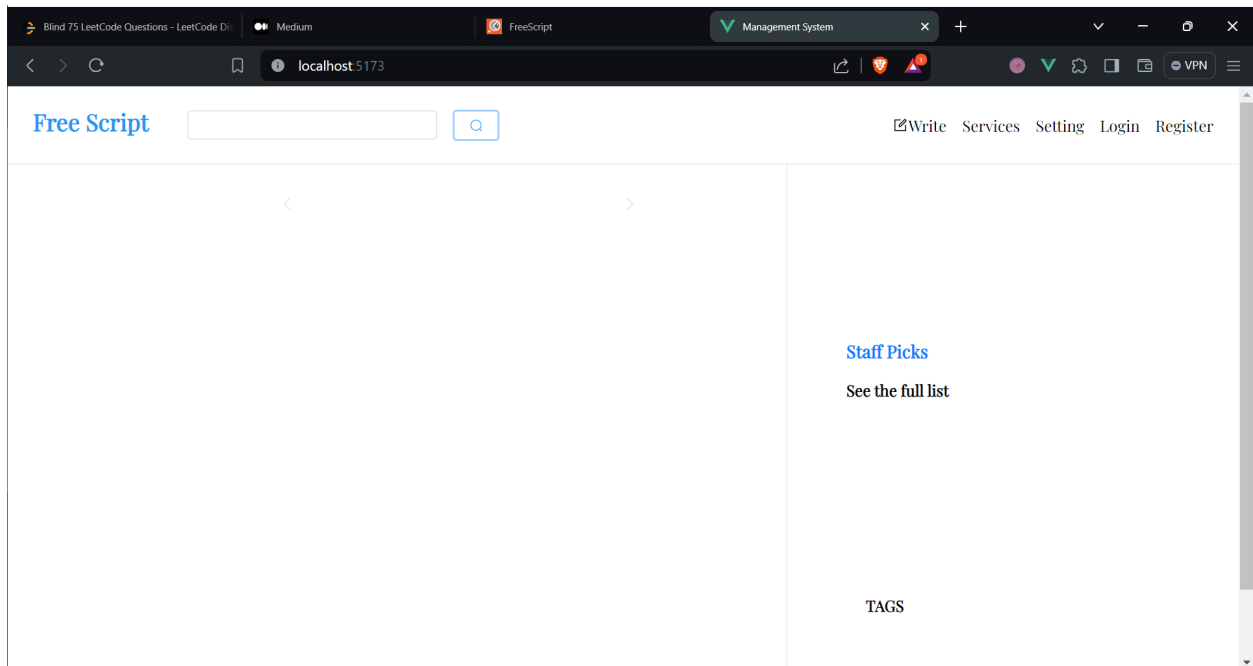
Front-end: 主要使用Vue3 with pinia store and ts, UI相關使用 tailwind and vue-element-plus

Back-end: 主要使用express.js with ts, 並使用類似MVC and OOP model進行設計

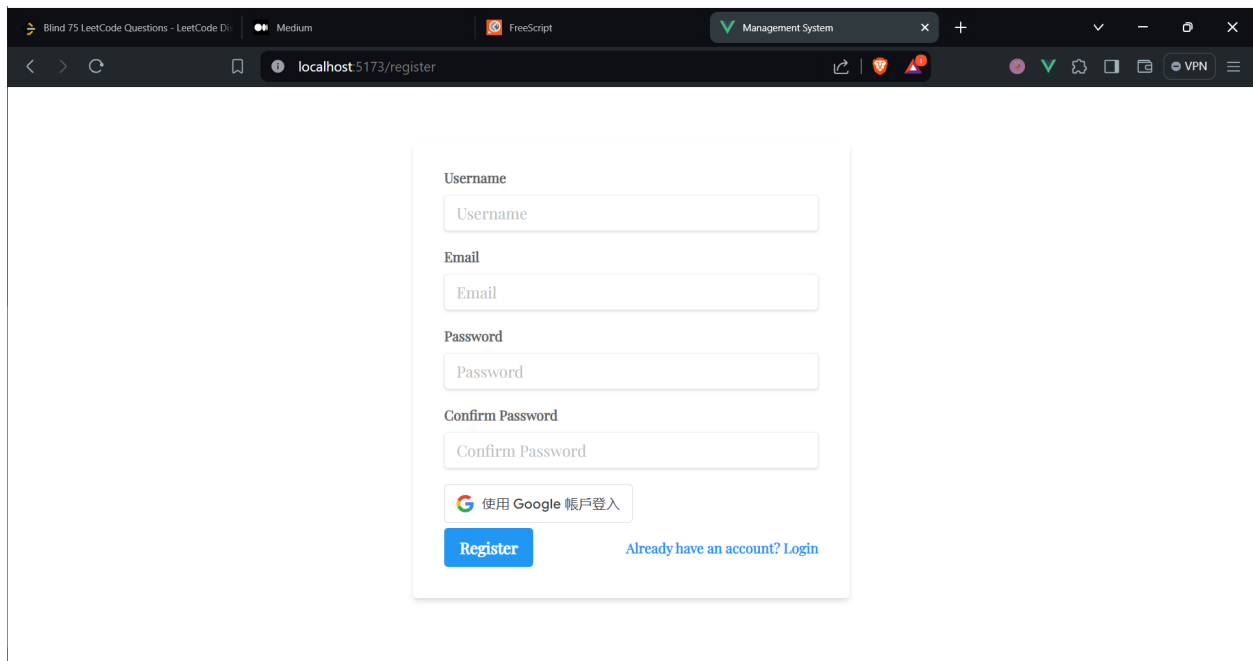
接下來我會以頁面的方式直接進行介紹



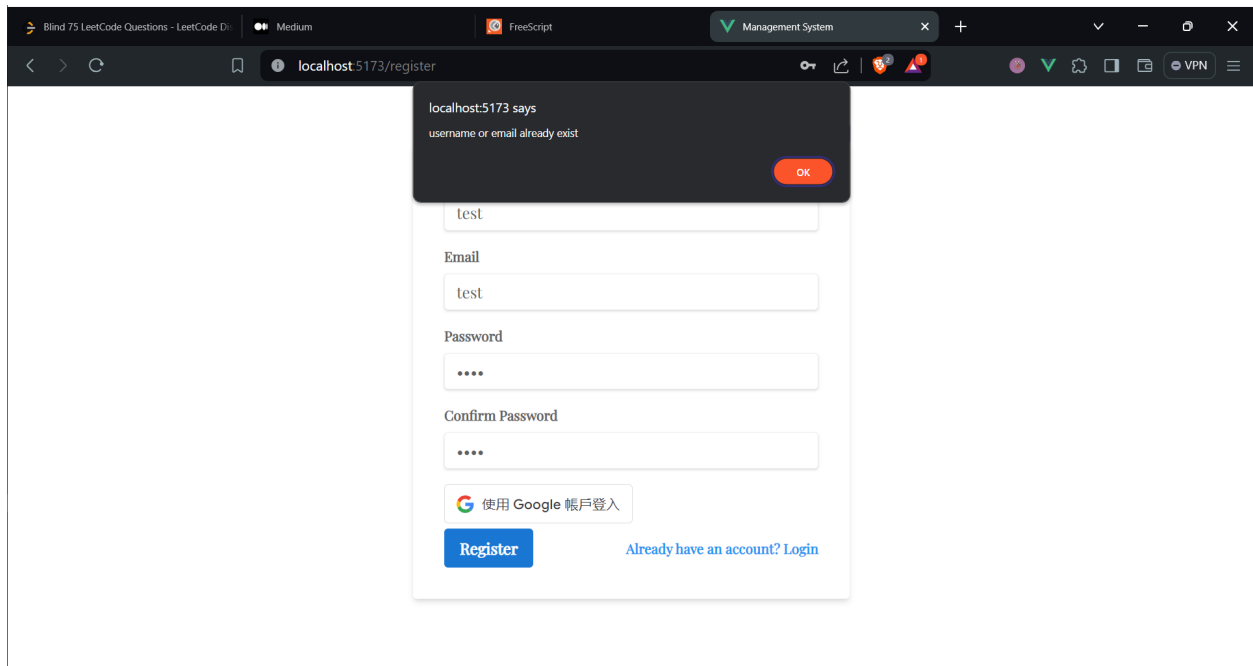
這是剛開起的首頁，會偵測login-token 而跳出 error alert, 然後request data 都會失敗，因為有back-end api有設置cookie檢測



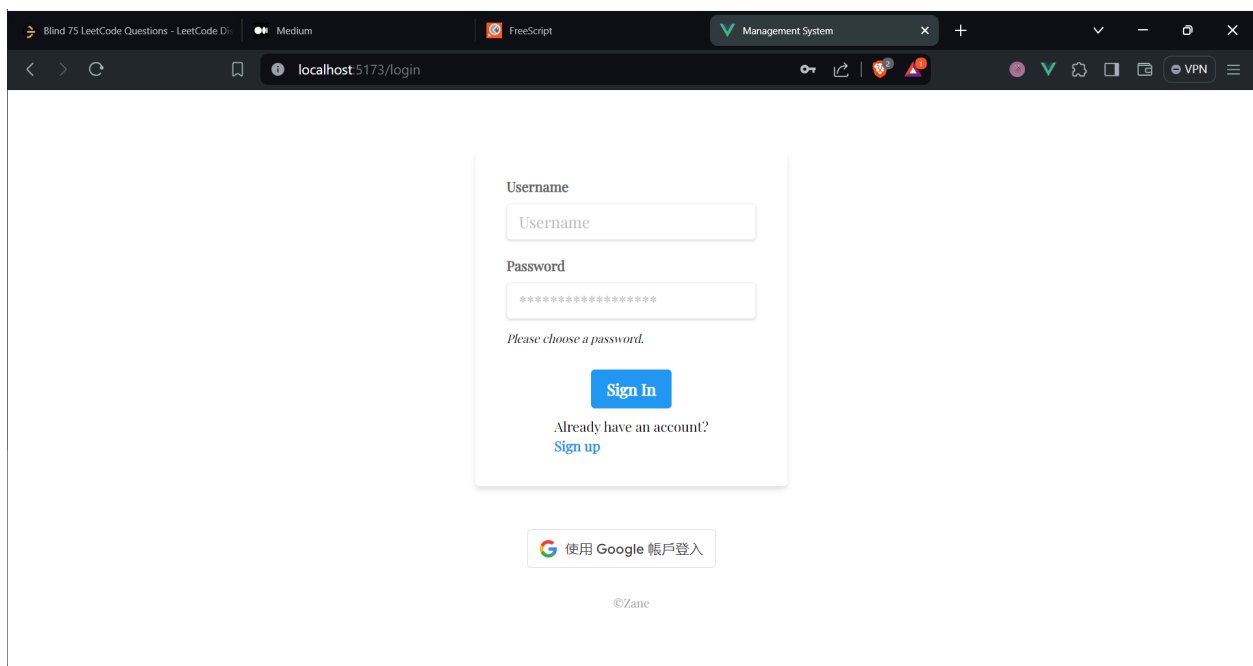
back-end api cookie 檢查機制，所以沒有任何request data



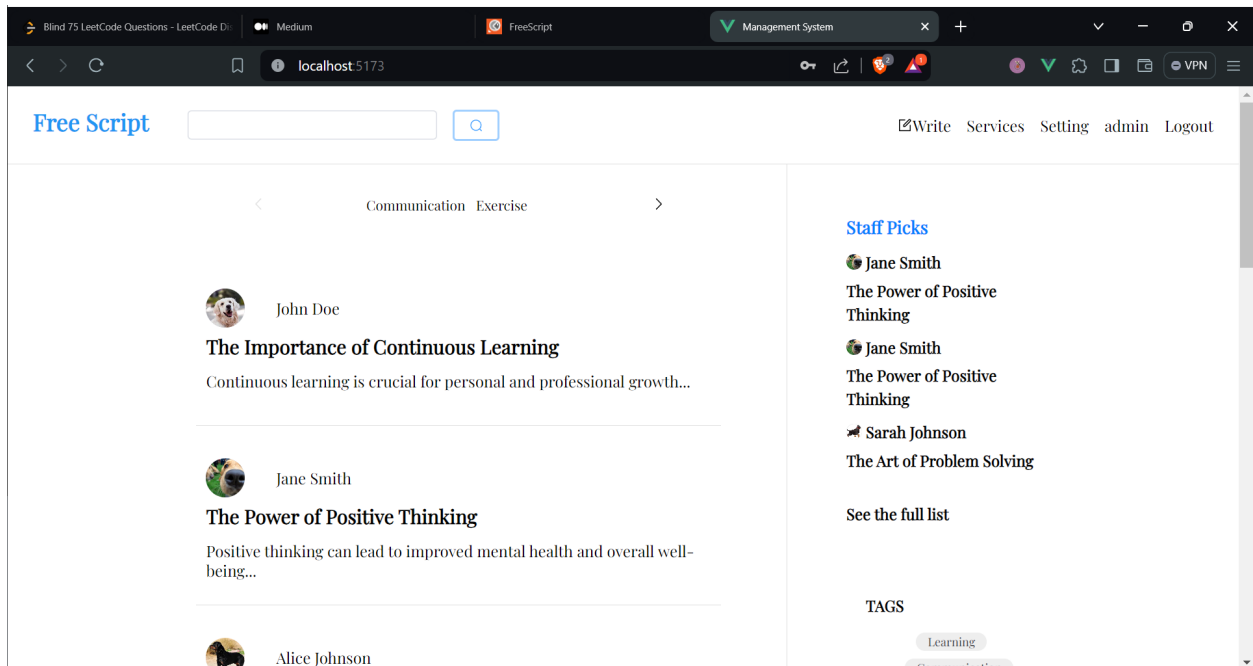
註冊頁面 可以選擇直接輸入value or directly using google api 進行登入



如果已經有註冊過的帳號，會顯示alert 說明帳號已經存在
透過api response status 來判斷錯誤原因與內容



註冊完成會直接到login頁面 一樣可以直接透過google 登入 or 自行輸入value



登入完成後就會顯示正常頁面

▼ Navigation :

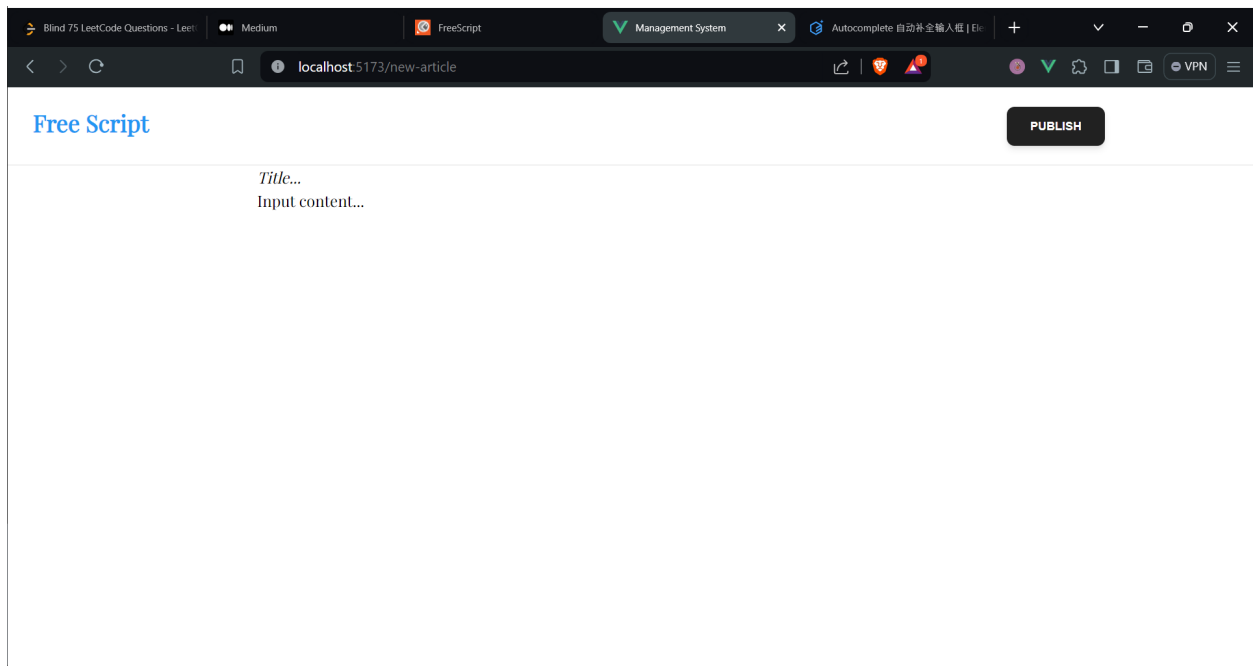
採用element-plus 的navigation, 並使用該component的slot功能, 也就是模組化的方式進行展現

1. 搜尋欄: 會根據value 做request to select db value, 可以找autor name and article title, 找到後 可以進行頁面跳轉
2. write: 撰寫文章的頁面
3. services: 目前提供兩個功能, 1. 刪除login-token cookie, 為了測試而設計的 ; 第二個則是mysql圖表展示
4. setting: 做一個簡單update user 的功能 主要展示設置有權限限制, 具備資格的使用者才可以看到user table
5. admin: 這邊是登入後會顯示username
6. logout: 會刪除所有的登入資訊, 包含login-token cookie

▼ 主頁面:

1. 左上推薦清單: 點按之後會只顯示符合該tag 的article list
2. 左側主要的article list: 有做分頁查詢以及無限捲動
3. 右上staff picks: 目前設計是先隨選三篇article views>500 的data 進行顯示
4. 右下tags: 點按之後會跳轉頁面, 該頁面只顯示該tag 的article

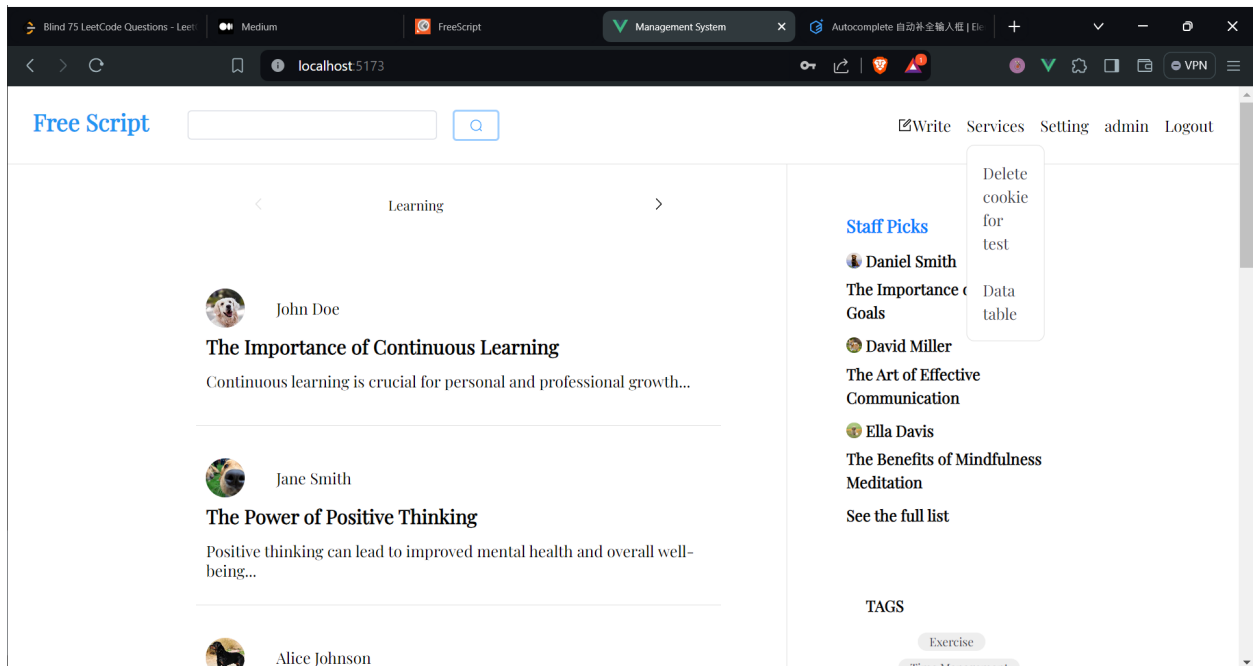
接下來對navigation 的功能做實際的展示



我使用一個第三方編輯器tiptap

輸入的內容會直接變成html format 然後request 存進db

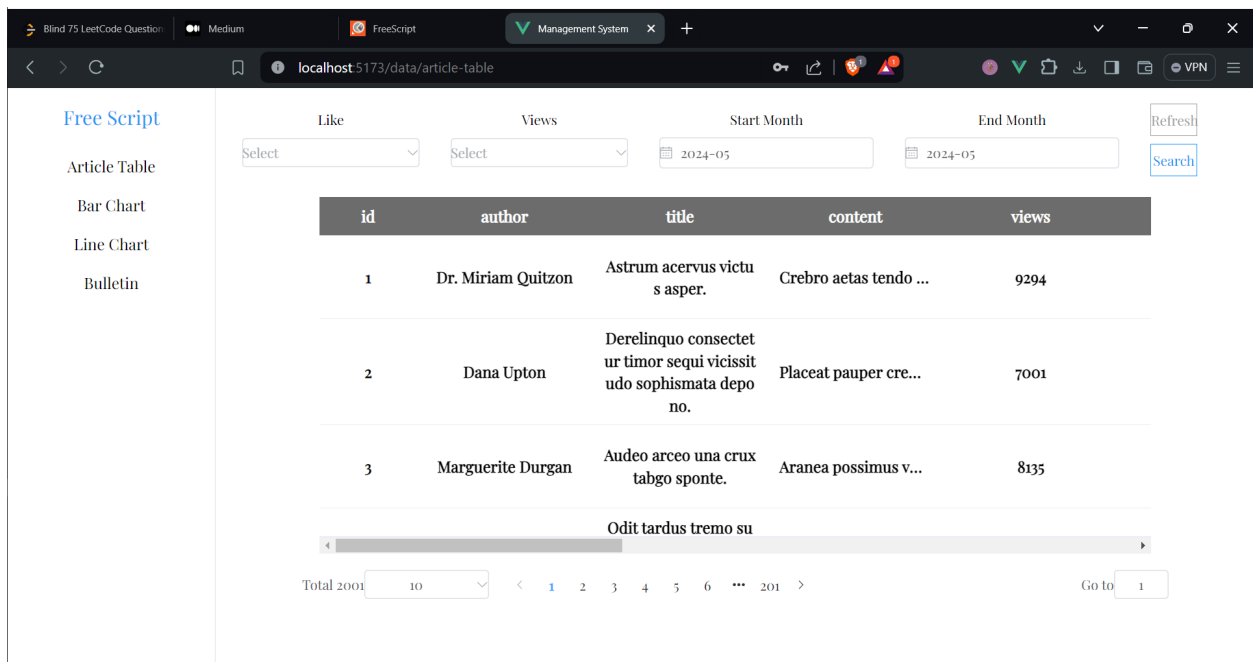
右上publish button 進行發送



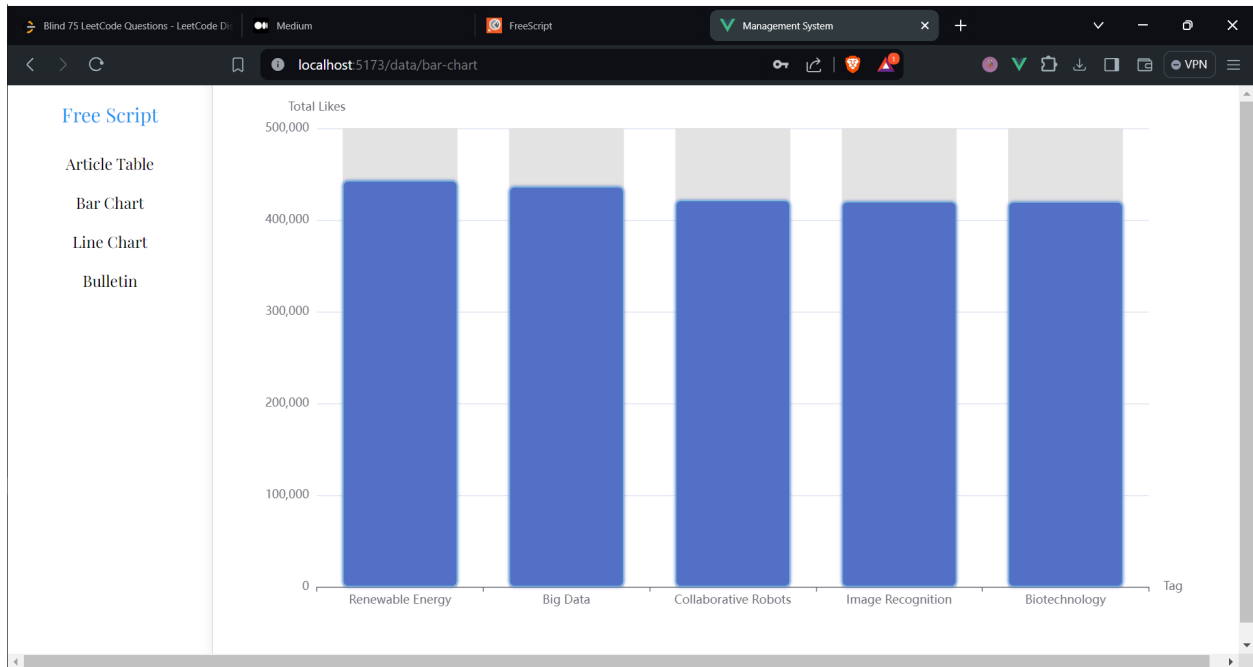
service的部份兩個

第一個只是為了測試而創建的刪除login-token cookie 為了setting部分的登入者權限的展示

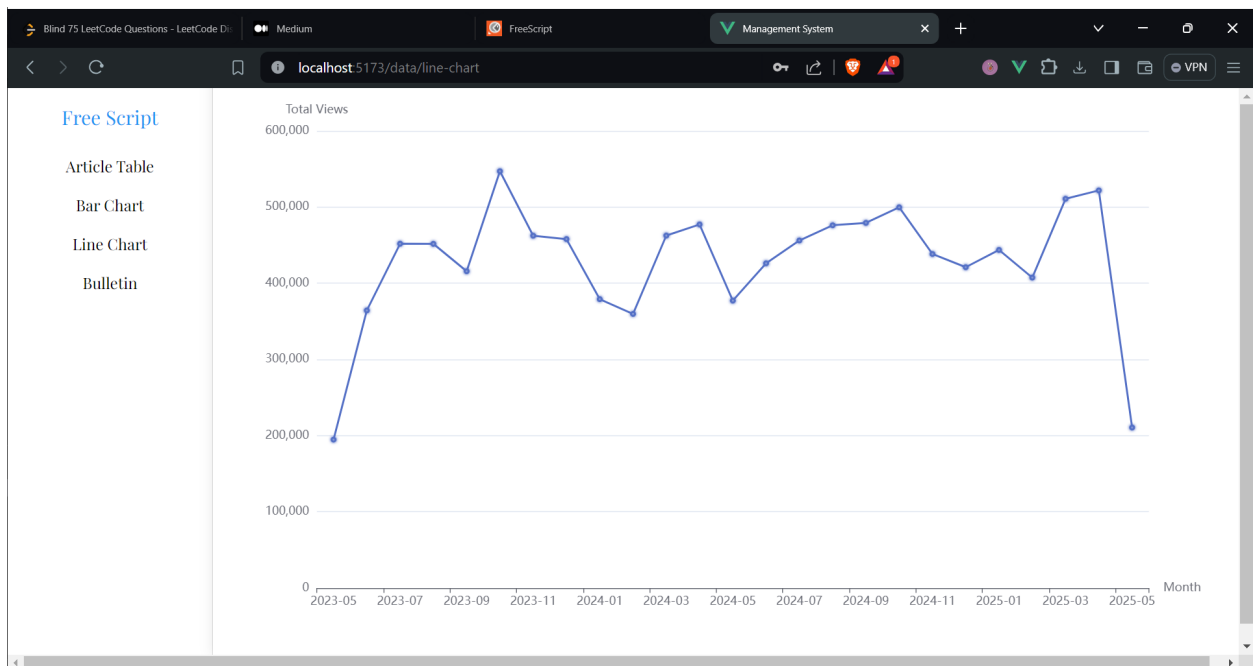
第二個則是mysql table 的展示



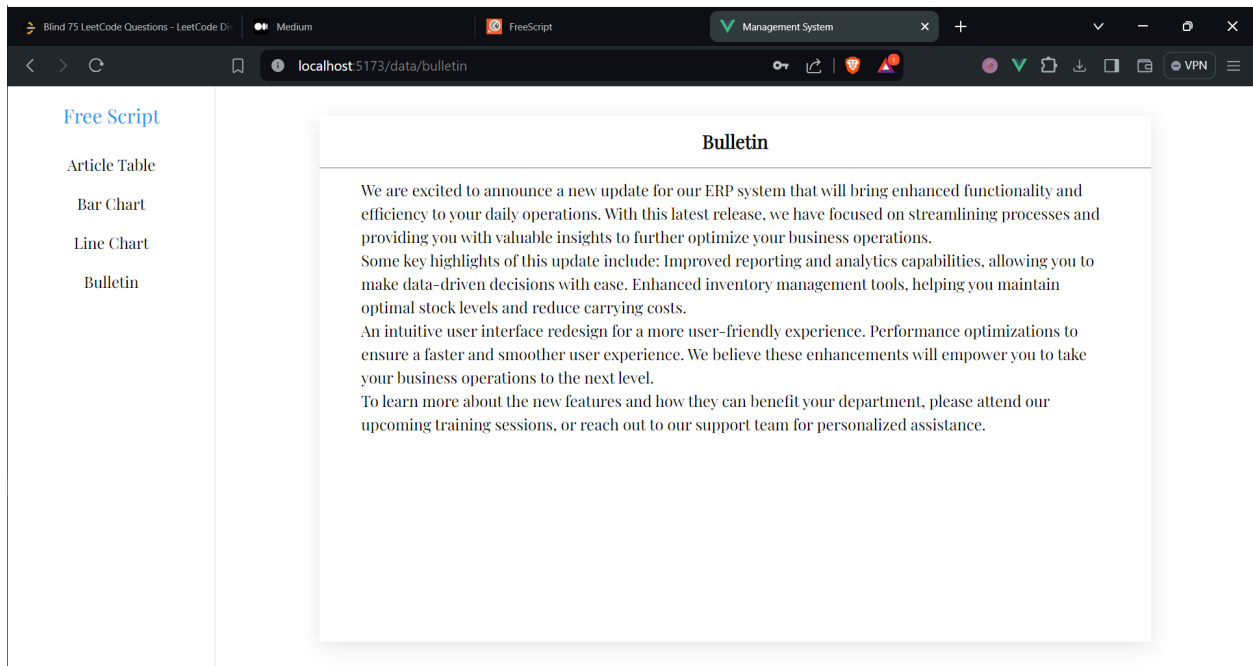
article table 展示



使用apache echart 做的圖表展示



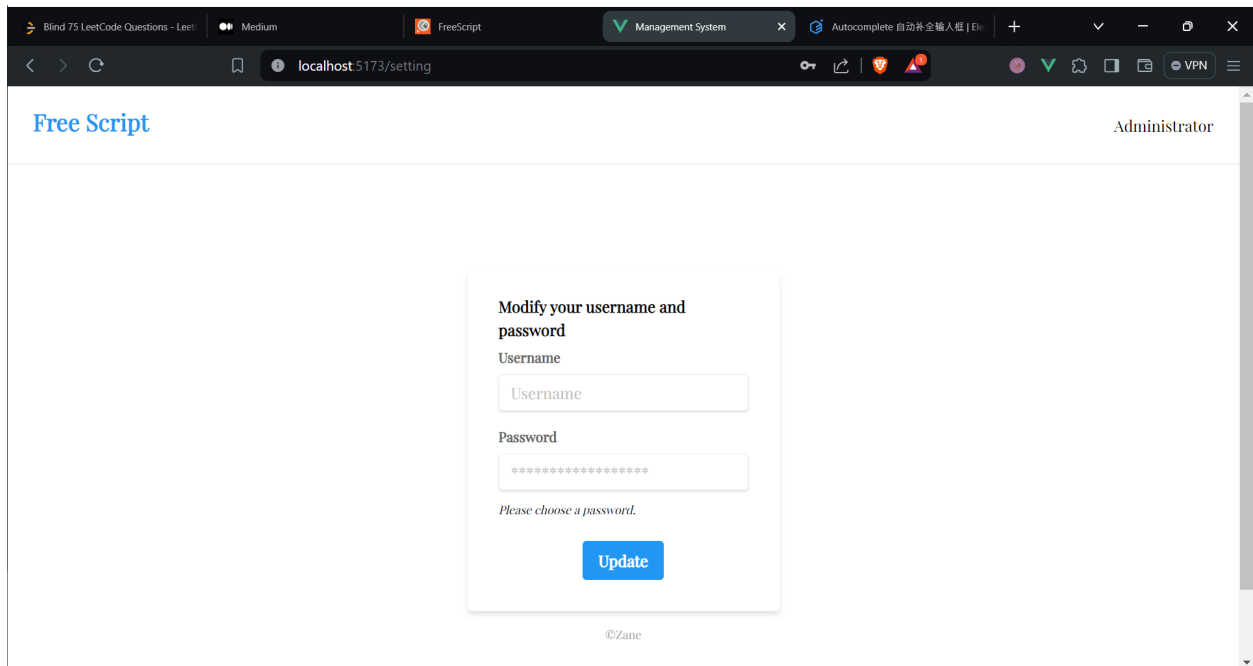
第二種類型的圖表



單純的文字內容

這一頁要展示的是SSR的功能 我是直接透過express.js with ejs response html 進行展示

雖然不是直接連線back-end server 的SSR，但做到了由back-end server 產生的部分SSR

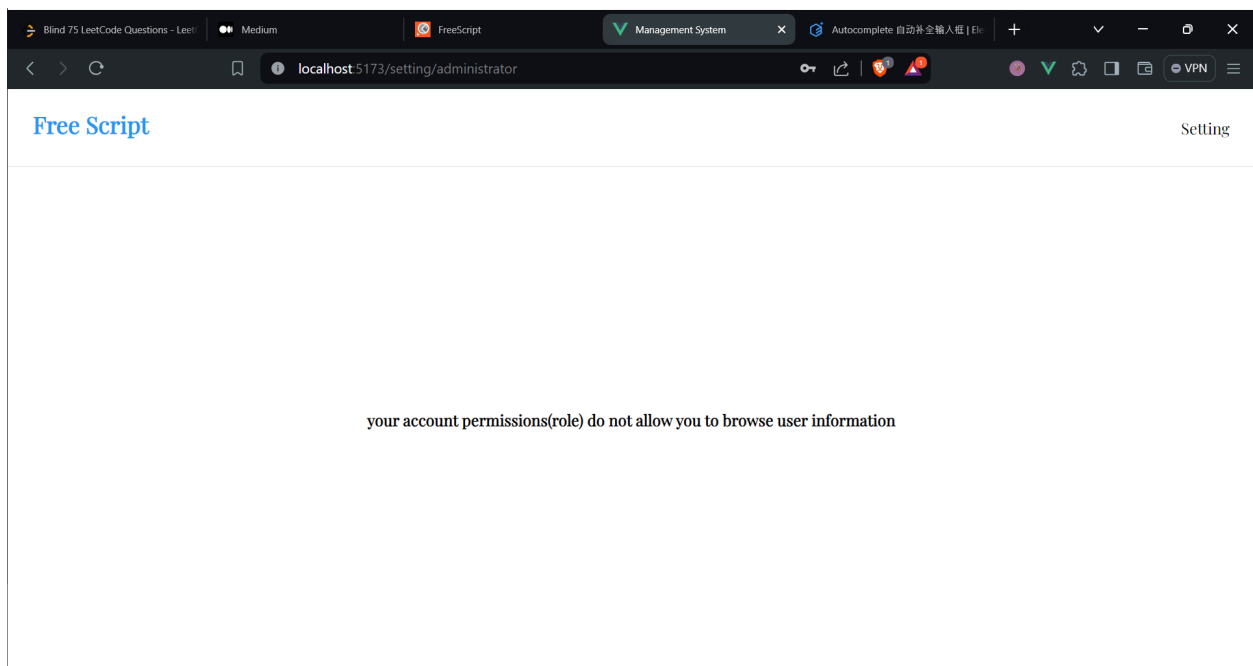


這是setting部分

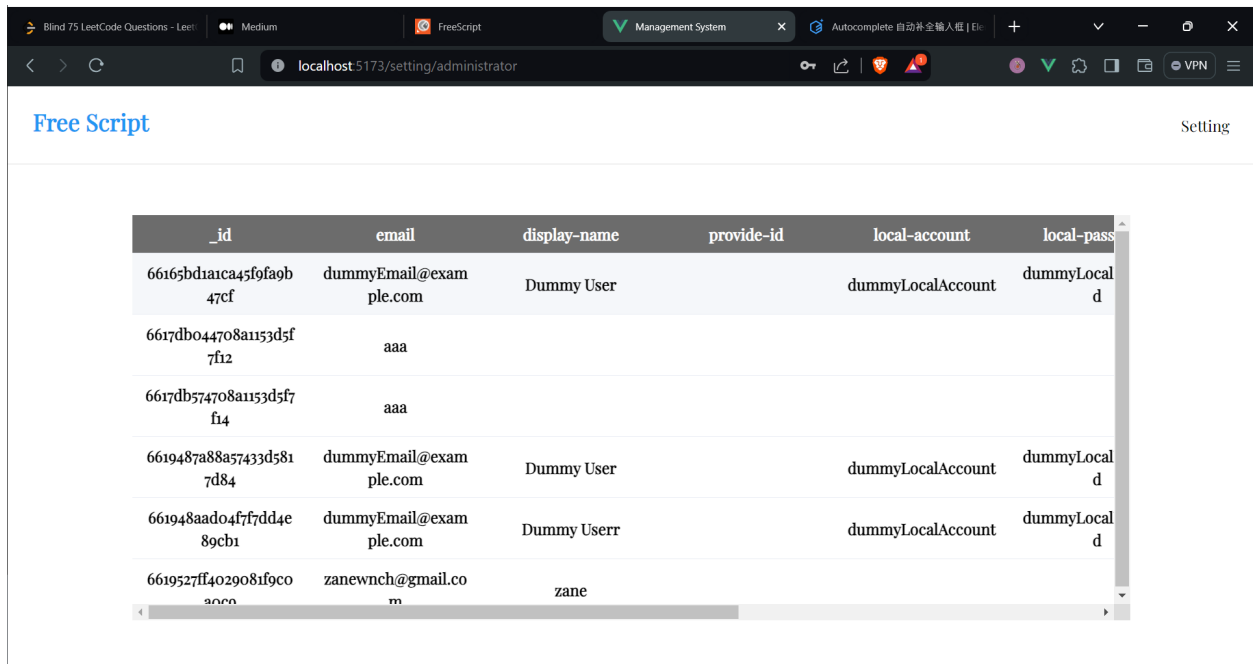
一般使用者只能修改user data

如果點按administrator

就會出現



而如果login 的account 是administrator
則會顯示 user table




_id	email	display-name	provide-id	local-account	local-pass
66165bd1a1ca45f9a9b47cf	dummyEmail@example.com	Dummy User		dummyLocalAccount	dummyLocal d
6617db044708a1153d5f7f12	aaa				
6617db574708a1153d5f7f14	aaa				
6619487a88a57433d5817d84	dummyEmail@example.com	Dummy User		dummyLocalAccount	dummyLocal d
661948aado4f7fdd4e89cb1	dummyEmail@example.com	Dummy User		dummyLocalAccount	dummyLocal d
6619527ff4029081f9c0a0c0	zanewnch@gmail.com	zane			

最後我有用postman替我的back-end api 寫api document
以下是連結:

FreeScript

The Postman Documenter generates and maintains beautiful, live documentation for your collections. Never worry about maintaining API documentation again.

 <https://documenter.getpostman.com/view/25236334/2sA3Bq5XKE>

