



# **HW#3 A Single-Page Application Made from a Spreadsheet**

## ! Copyright and Ownership Notice

關於這個 notion page 以及其包含的所有 pages ( 以下統稱「本 page」) , 其著作權與所有權是屬於台大電機系黃鐘揚教授以及其所開設的網路服務程式設計課程之所有助教 (以下統稱「作者們」) 所擁有。

基於推廣技術與服務大眾之精神，本 page 採取 MIT 授權，即任何人皆可因任何目的使用、複製、修改、發布、與再散佈本 page，包括商業用途。不需簽署協議，也不需支付授權費，惟任何的使用與散佈必須包含本著作權與所有權聲明。如有違反本著作權與所有權聲明，本 page 之作者們仍保有相關的法律追朔權利。詳細關於 MIT 授權的規範請見 [OSI 官網](#)。

本 page 之內容以「現狀」提供 (provided "as-is")，不附帶任何保證。對於本 page 之內容如有任何疑問或是建議，請來信 [eewebprogramming@googlegroups.com](mailto:eewebprogramming@googlegroups.com)。

The copyright and ownership of this Notion page and all pages it contains (hereinafter referred to as "this page") belongs to Professor Chung-Yang Huang of Department of Electrical Engineering, National Taiwan University, and all teaching assistants of this Web Programming course that he teaches.

In the spirit of promoting technology and serving the public, this page adopts the MIT License, meaning anyone can use, copy, modify, publish, and redistribute this page for any purpose, including commercial use. No agreement needs to be signed, and no licensing fee needs to be paid, provided that any distribution must include this copyright and ownership notice. In case of violation of this copyright and ownership notice, the authors of this page still retain the relevant legal right to take retroactive action. For detailed regulations regarding the MIT License, please see the [OSI official website](#).

The content of this page is provided "as-is", without any warranty. If you have any questions or suggestions regarding the content of this page, please email [eewebprogramming@googlegroups.com](mailto:eewebprogramming@googlegroups.com).



[Web Programming Home](#)

## 截止時間

- Submission Deadline : 9pm, Mon, 2025/10/13
- Review Deadline : 9pm, Mon, 2023/10/20

## 題目說明

- 本次作業是希望你練習從一個 data source (i.e. one more multiple spreadsheet(s)) 以及使用 AI coding tool (i.e. Cursor) 來打造一個純前端的網路服務。
- 題目可以自訂，但整個服務應該要有“項目瀏覽”，“選取至某個容器 (到購物車、行事曆等)”，“送出記錄”等階段。
- 在「項目瀏覽」方面，你的設計可以是分類瀏覽、關鍵字/條件搜尋... 檔，總之，請從服務設計的角度出發，設計出一個可以讓使用者輕鬆找到自己想要的項目。
  - 每個項目在展示時應該要提供足夠、有幫助、但也不要過多的資訊說明，以利使用者做決定
  - 如有需要，可以在項目上加上一些互動，以利使用者瀏覽更詳細的資訊，甚至做比較
- 在「選取至某個容器」方面，請從服務設計的角度出發，設計一個讓使用者方便理解、做決定的 UI/UX (e.g. 選課系統的課表, 電商系統的購物車, 等)
  - 與下一步“送出記錄”的區別是，在這一步像是在 git 裡頭的“staging”，也就是說，使用者還在 shop around, 這個容器應該要讓他無負擔的可以隨時增減、修改
  - 歡迎做出一些有利於使用者作出決定的設計
- 在「送出記錄」方面，表示使用者已經做好決定，正式把它送出。Again, 請從服務設計的角度思考，定義出這個步驟與上一個步驟的差別，例如：選課系統裡頭的確定選課、電商系統的送出訂單等。
  - 記錄一但送出，如果使用者如要修改，請你依照你想像中的服務邏輯，either 是不可以修改，或者是要走過一個跟上述課表/購物車不一樣的流程，已表示這個送出記錄不同的意義
  - 在一個完整的服務，這個步驟應該是要把記錄送到伺服器，存在使用者的個人資料裡頭。但我們這次還是純前端的 app, 所以你還是可以使用一個 component/state 存這個記錄就好
  - 因為是純前端，所以你也不用做到最後的金流、會員系統等。請自行設計一個合理的 ending 即可
- 這個服務不需要有“使用者 ID”或者是“登入”的設計

- 當然，你的服務的最佳設計很可能不一定符合“項目瀏覽”，“選取至某個容器 (到購物車、行事曆等)”，“送出記錄”等三個階段，所以這邊可以有彈性，但我們期待你的設計是以讓使用者覺得容易理解資訊、做出選擇為原則。
- 如果你想要做一個 **台大的選課服務 app**，我們提供了以前 (100 學年度下學期) 全校的課表供你參考，可以從這裡下載 ([NTU COOL](#))。
- 如果你想做其他的服務，例如：購物網站、dating app 等，而如果你沒有合理合法的 data source, 你可以請 AI (e.g. in Cursor) 幫你生成。無論如何，請不要直接拿上課示範的 restaurant-app 的 dish-data.csv 來使用 (i.e. 即使你還是想要做一個點餐系統，請自行產生別的 spreadsheet)。

## 基本要求

- 除了上述題目說明的要求/規定事項之外，請達成底下基本。
- 程式語言除了必要的 HTML, CSS 之外，動態網頁的部分規定使用 React.ts (i.e. TypeScript 版本的 React)，而不是 pure JavaScript or React.js。
- UI 的部分請使用現成的框架 (i.e. 不要只是使用 pure CSS)，你可以選擇課堂上有教過的 MaterialUI, Ant Design, Shadcn, TailwindCSS (with cn), 或者是任何你喜歡的框架。
- Spreadsheet 原始應為一個或者是多個 .csv 檔案，存放在 hw3/src/public/data 底下。你的服務應該要做到當使用者編輯/更改 .csv 檔案，更新到你的 repo 以後，你的服務應該會自動更新 (either hot reload, or restart app)，不需要改動其他的程式碼
- 你的服務如果需要多媒體檔案 (e.g. 圖檔、音檔、影像檔等)，除非數量與 size 很小，否則請不要上傳到 GitHub repo. Instead, 你應該要把它們放到雲端的空間，然後讓你的 app 可以在執行的時候去抓取檔案即可 (i.e. 把連結放在 code 裡面)
- 請 (指揮 AI) 適時地將你的 code refactoring. 例如，不要把產生畫面與操作資料的邏輯混在一起，應該要把 code 分成 hooks, components 等，以利閱讀與維護
- 如果有多個 components 需要 access 同樣的一些資料，請把他們放到 context 底下，以利管理
- 由於 app 越來越複雜，你的 Cursor 對話記錄很可能會變得太大 — 我們的初衷是希望同學們可以分享如何 “vibe” code 你的作業，因此，請留下你的 prompt 部分就好，不用把 Cursor 的回應細節也傳上來
  - 你可以將 Cursor chat history 的 .md 檔交給 cursor 處理，請他把 prompts 留下來就好
- 雖然不會算在檢查的項目裡頭，但請務必養成開 branches, 打 tags, 常常 add/commit 的習慣，以利你如果需要 backtrack/undo 到某個 historical point, 你的 code 還是救得回來

- 請務必建立 .gitignore 檔案，把例如 node\_modules 等不必要的檔案擋住，不要上傳到 GitHub
- 除了 Cursor 產生的 README 之外，請務必在上傳作業前仔細的看過你的 README — 是否有把如何安裝、執行你的 app 有寫清楚？如果因故造成 reviewer 有評分的困難，需要動用到 TA 的協助，將會有一定程度的扣分

## Review 評分建議

- 這次作業的評分請以整體 web service 的設計是否貼心、完整、好用為主 (80%)，而設計美觀等因牽涉個人主觀意見，請從寬認定 (20%)。
- 評分時也請協助檢查一下是否有符合/完成上述的基本要求，評分建議具體說明如下：
  - Dead (0): 跑不起來，或者是一打開就當掉、白畫面
  - Bad (1): 整體服務實在是太陽春，令人使用起來不太舒服
  - Fair (3): 服務算是堪用，但有些許地方做得不太貼心，防呆不足，很容易讓使用者踩雷
  - Good (5): 有做到上述的基本要求，且使用起來相當順暢，沒有什麼不合理的地方
  - Excellent (6): 可以感受到貼心的服務設計，或者是有令人驚豔的 UI 呈現，整體服務使用起來十分舒服
- 請發揮同學愛，不要用過於嚴苛的標準來評分。

## 繳交、評分方式

- 繳交：
  - 請將所有作業內容放在 wp1141/hw3 內，並在繳交截止時間前將你的作業 push 到你 GitHub 上 wp1141 這個 repo 上面的 main branch：

```
# in wp1141/, on the `main` branch git add hw3 git commit -m "<some comment>" git push
```

- 請確認 GitHub 上預設 branch 為 main。若不是，請參照[這則教學](#)修改。
- 我們不接受遲交哦！（遲交 = 0 分）