

Final Project -- Github URL

Final Project -- Vercel URL

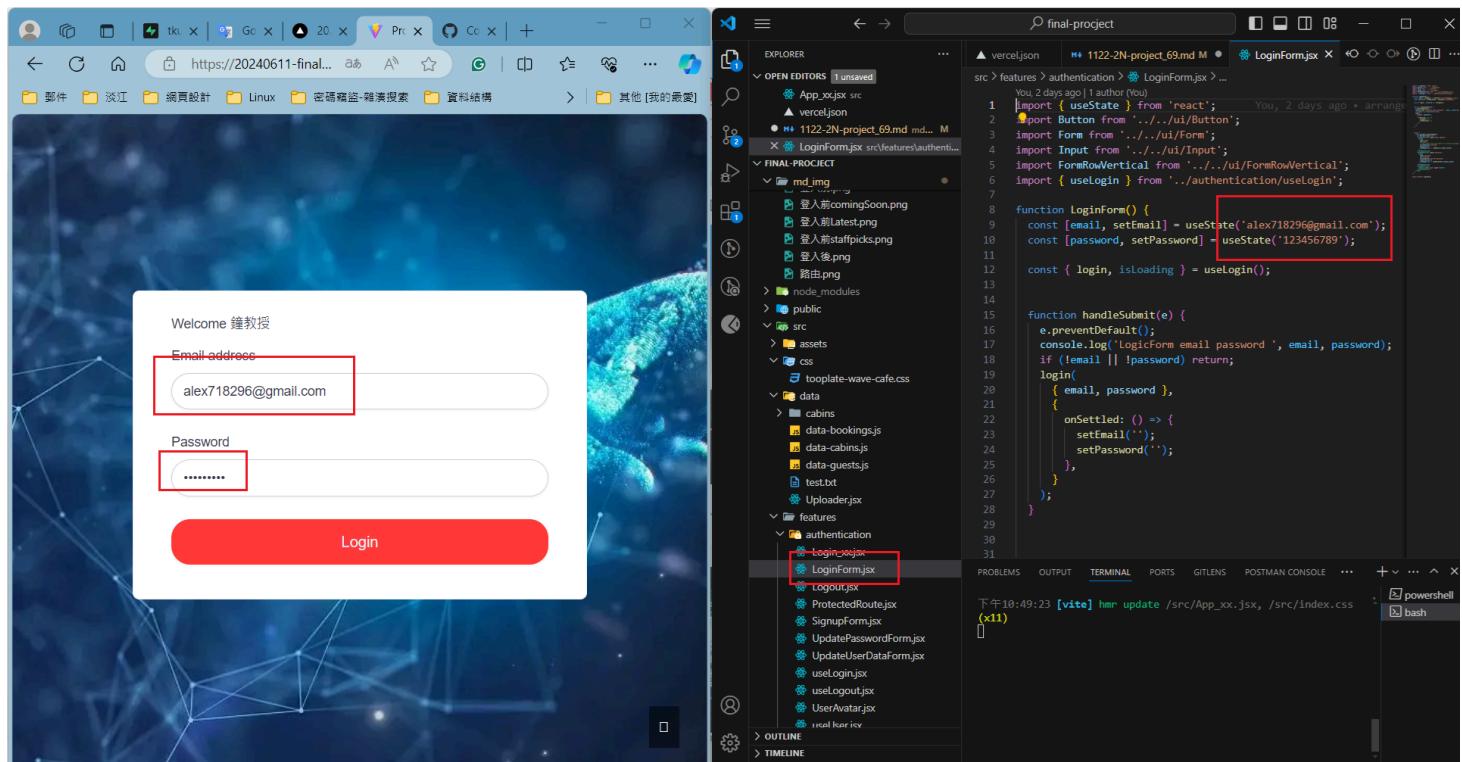
# 教授 20240612 指導須修正事項(已完成項目)

## 1. 帳號登入 LOG 紀錄(已將帳號密碼顯示於頁面中)

記錄截圖

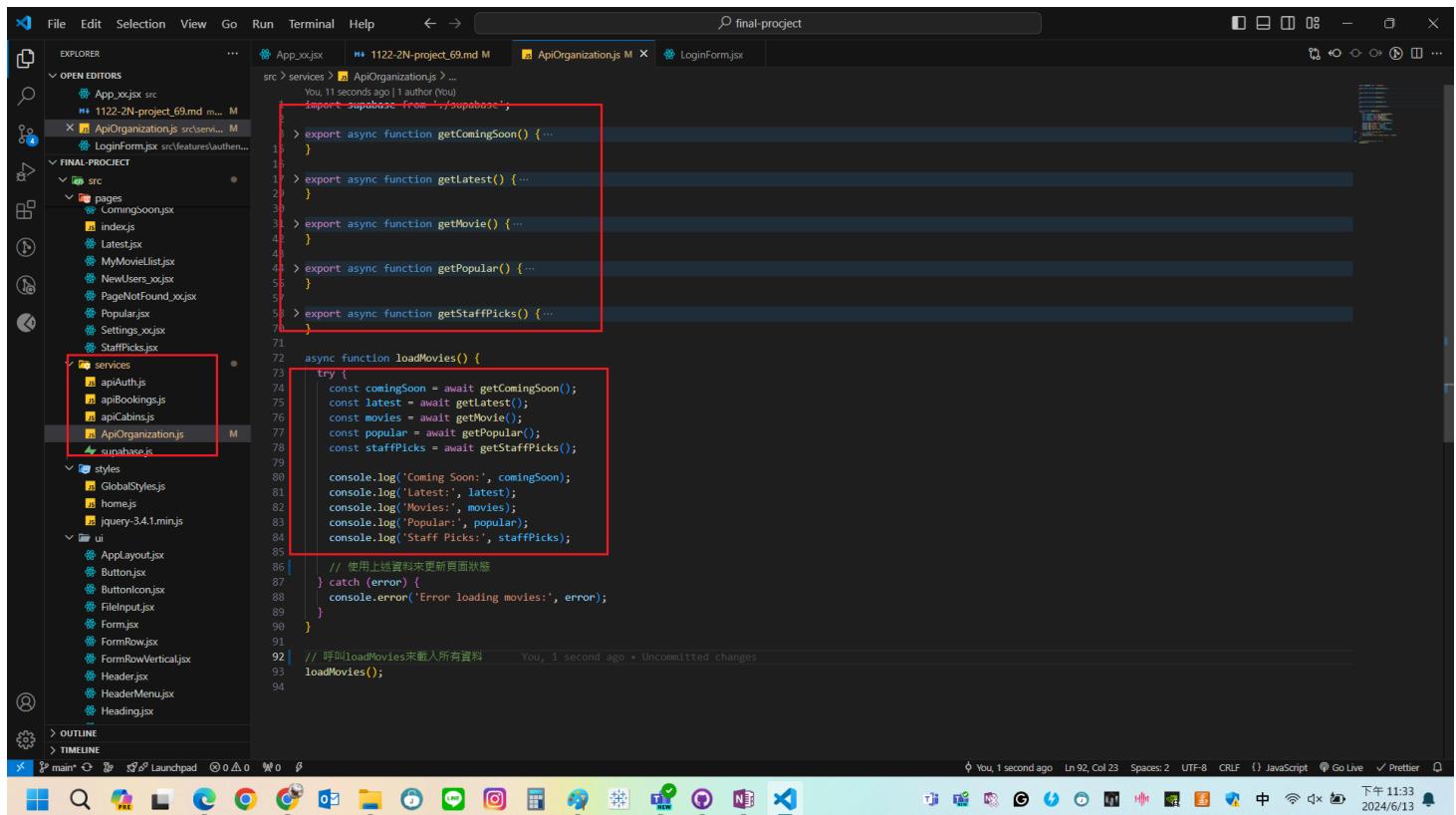
The screenshot shows a browser window displaying a movie trailer website. A red box highlights the blue refresh icon in the top right corner of the page header. Another red box highlights the 'Local storage' section in the Network tab of the developer tools, specifically the entry for 'access\_token'. The token value is partially visible as 'eyJhbGciOiJIUzI1Ni...'. A third red box highlights the detailed view of the 'access\_token' object in the 'Storage' section, showing properties like 'access\_token', 'refresh\_token', 'token\_type', and 'user' (with ID, provider, email, and role information). The bottom status bar shows the date and time as '下午 11:10 2024/6/13'.

## email & password 紀錄



## 2.API\_arrange(Add API organization, Delete 6 APIs)

將 apiPopular、apiLatest、apiComingSoon、apiStaffPicks、apiMyMovielist、movieList 整合成 API organization 單一 server



# 3.Vercel page can't open right

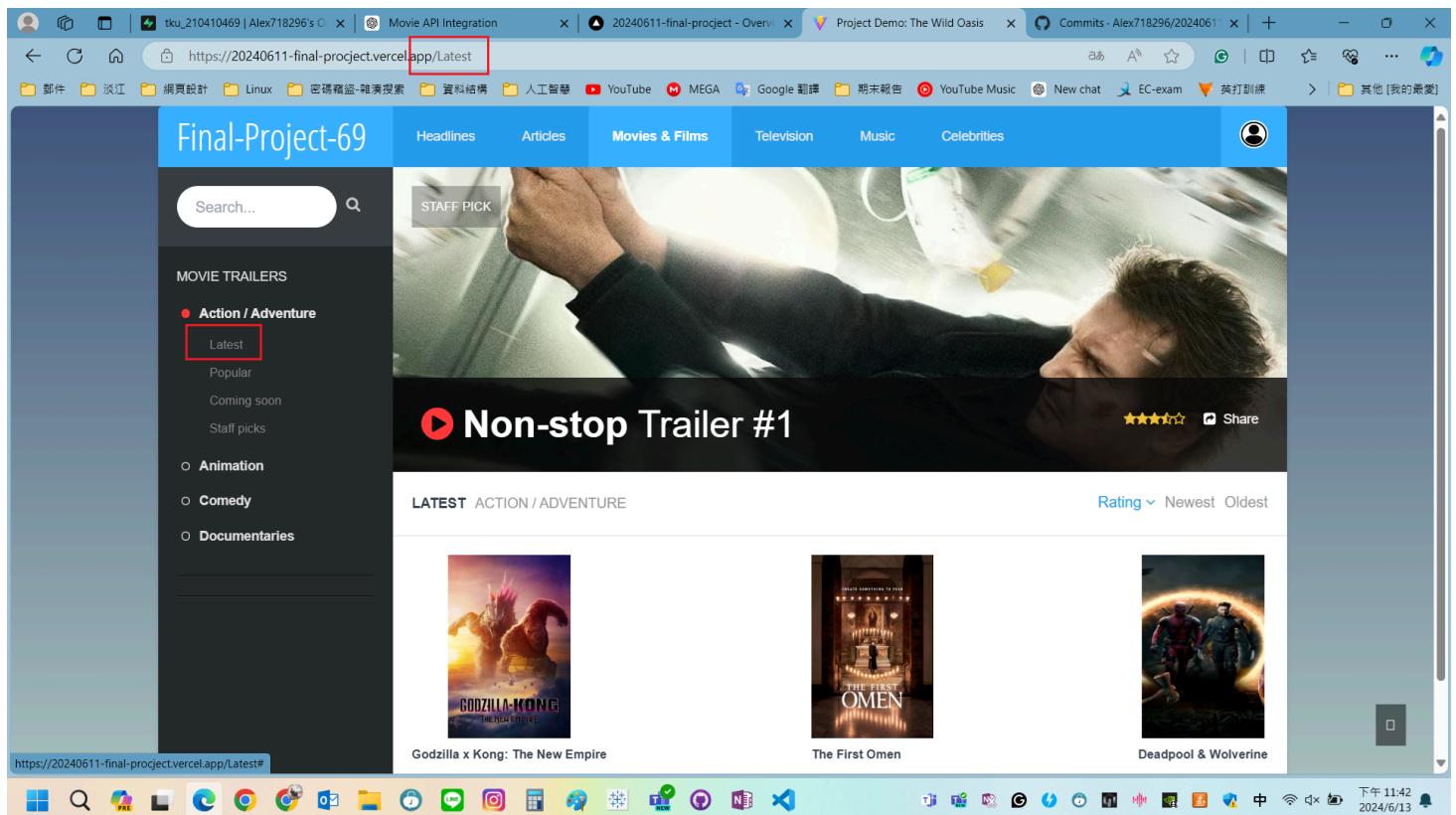
新增 `vercel.json`，將由 JavaScript 管理 router。(定義 router 邏輯)

The screenshot shows a code editor (VS Code) and a browser window. The code editor has several files open, including `vercel.json` which contains a rewrite rule:

```
rewrites: [ { "source": "/(.*)", "destination": "/" } ]
```

The browser window shows a preview of the project with the title "2.API\_arrange(Add API organization, Delete 6 APIs)". Below it, another browser window shows the "3.Vercel page can't open right" error message.

## Vercel 實際頁面執行



tku\_210410469 | Alex718296's C | Movie API Integration | 20240611-final-project - Overview | Project Demo: The Wild Oasis | Commits - Alex718296/20240611 | +

https://20240611-final-project.vercel.app/popular

# Final-Project-69

Headlines Articles Movies & Films Television Music Celebrities

Search...

MOVIE TRAILERS

Action / Adventure

- Latest
- Popular Popular
- Coming soon
- Staff picks

Animation

Comedy

Documentaries

STAFF PICK

Non-stop Trailer #1

★★★★★ Share

POPULAR TRAILERS ACTION / ADVENTURE

Rating Newest Oldest

Movie	Rating	Release Date
The Amazing Spiderman	★★★★★	2012-07-13
Sin City: A Dame To Kill For	★★★★★	2014-05-09
Transcendence	★★★★★	2014-04-18
Need For Speed	★★★★★	2014-03-14

https://20240611-final-project.vercel.app/popular#

下午 11:43  
2024/6/13

tku\_210410469 | Alex718296's C | Movie API Integration | 20240611-final-project - Overview | Project Demo: The Wild Oasis | Commits - Alex718296/20240611 | +

https://20240611-final-project.vercel.app/ComingSoon

# Final-Project-69

Headlines Articles Movies & Films Television Music Celebrities

Search...

MOVIE TRAILERS

Action / Adventure

- Latest
- Popular
- Coming soon Coming soon
- Staff picks

Animation

Comedy

Documentaries

STAFF PICK

Non-stop Trailer #1

★★★★★ Share

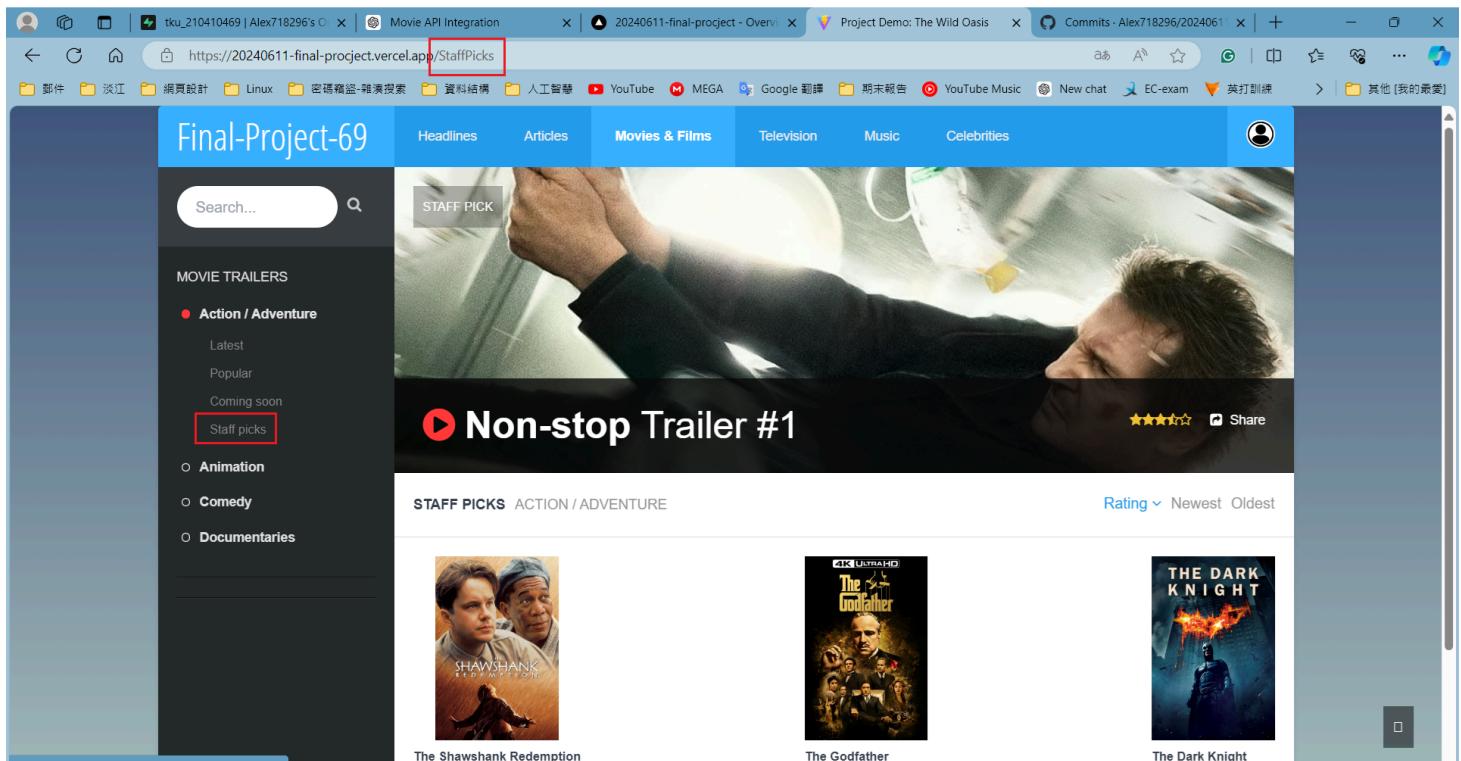
COMING SOON ACTION / ADVENTURE

Rating Newest Oldest

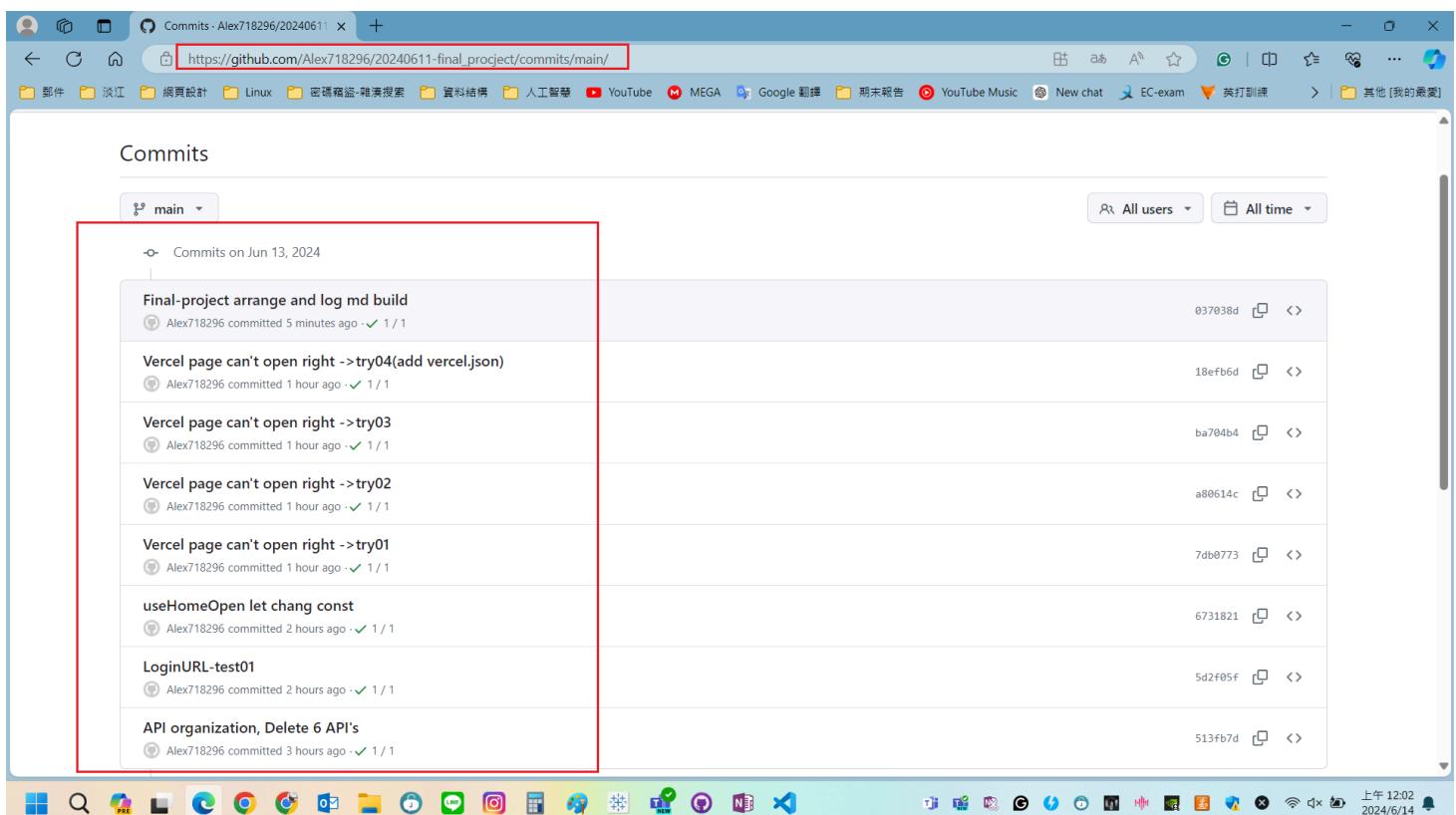
Movie	Rating	Release Date
Venom: The Last Dance	★★★★★	2024-06-14
Borderlands	★★★★★	2024-06-14
A Quiet Place: Day One	★★★★★	2024-06-14

https://20240611-final-project.vercel.app/ComingSoon#

下午 11:44  
2024/6/13



## 4. All Debug Log



---

## 評分說明

### 1. 基本分：(30%~70%)

依老師上課架構及內容，改換成其他應用，基本上可以執行者

### 2. 進階分 (30%)

依前台及後台完成度評分，自己努力愈多，完成度愈高者，分數愈高。

## Project 說明

### 重點 1 -- 單人，還是兩人，如何分工

單人 陶少煜 210410469

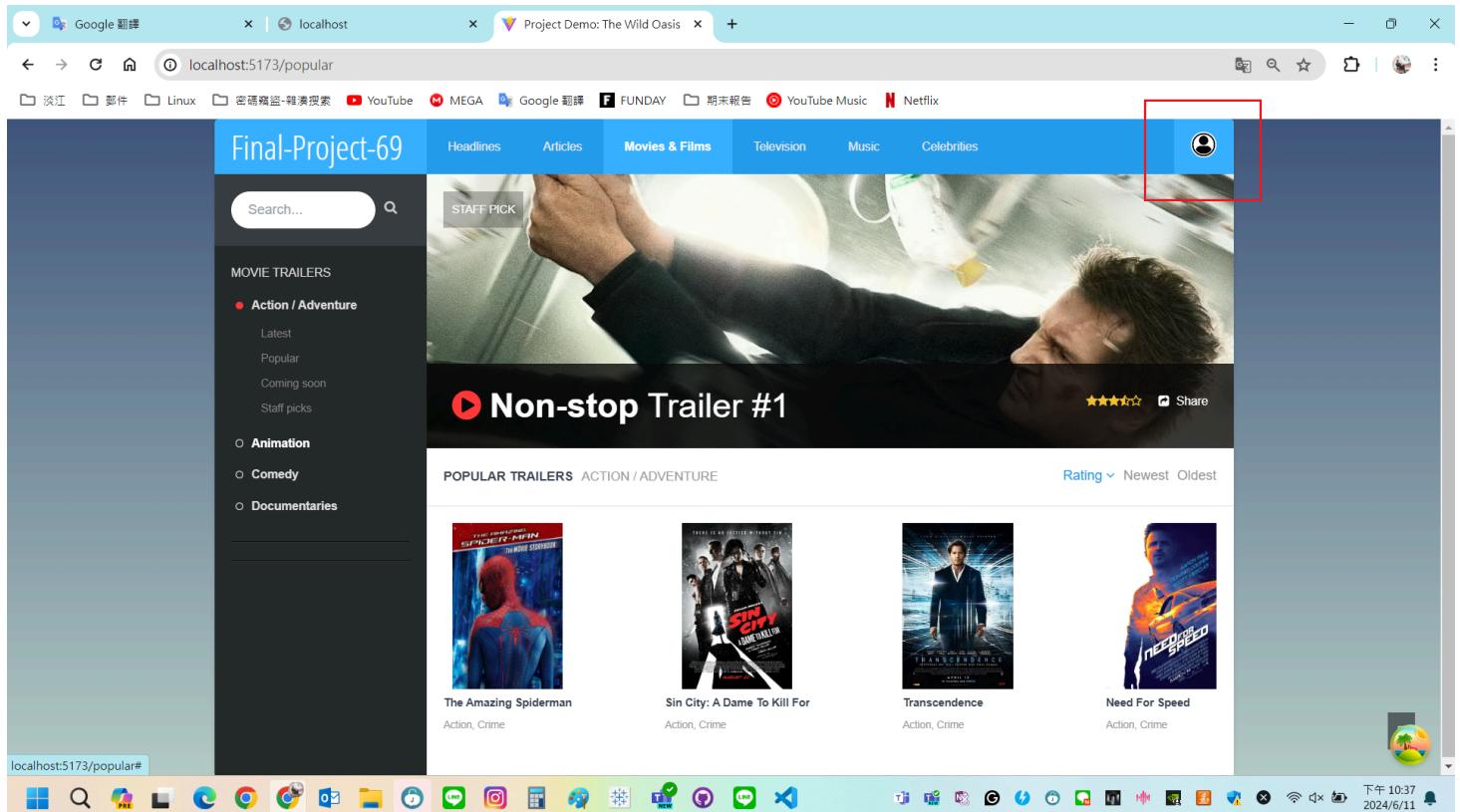
# 重點 2 -- Project 簡要介紹

## => project requirements

### 製作電影資訊清單包含

※登入前 : "/Popular",'Latest','StaffPicks','ComingSoon'

#### 1. 登入前 (Popular):



## 2. 登入前 (Latest):

The screenshot shows a web browser window with the URL `localhost:5173/Latest`. The page title is "Final-Project-69". On the left, there's a sidebar for "MOVIE TRAILERS" under "Action / Adventure", with "Latest" selected. The main content area features a "STAFF PICK" trailer for "Non-stop Trailer #1". Below it, a section titled "LATEST ACTION / ADVENTURE" displays three movie posters: "Godzilla x Kong: The New Empire", "The First Omen", and "Deadpool & Wolverine". Each poster has its title and genre below it. The Windows taskbar at the bottom shows various pinned icons.

## 3. 登入前 (StaffPicks):

The screenshot shows a web browser window with the URL `localhost:5173/StaffPicks`. The page title is "Final-Project-69". On the left, there's a sidebar for "MOVIE TRAILERS" under "Action / Adventure", with "Staff picks" selected. The main content area features a "STAFF PICK" trailer for "Non-stop Trailer #1". Below it, a section titled "STAFF PICKS ACTION / ADVENTURE" displays three movie posters: "The Shawshank Redemption", "The Godfather", and "The Dark Knight". Each poster has its title and genre below it. The Windows taskbar at the bottom shows various pinned icons.

#### 4. 登入前 (ComingSoon):

A screenshot of a web browser displaying a movie trailer website. The URL in the address bar is `localhost:5173/ComingSoon`. The page features a navigation bar with links to Headlines, Articles, Movies & Films, Television, Music, and Celebrities. On the left, there's a sidebar for 'MOVIE TRAILERS' with categories like Action / Adventure (selected), Animation, Comedy, and Documentaries. A 'STAFF PICK' section shows a trailer for 'Non-stop Trailer #1'. Below it, a 'COMING SOON ACTION / ADVENTURE' section lists three movies: 'Venom: The Last Dance' (Action, Crime), 'Borderlands' (Action, Crime), and 'A Quiet Place: Day One' (Action, Crime). The Windows taskbar at the bottom shows various pinned icons and the system tray.

#### 5. 登入前 (auth):

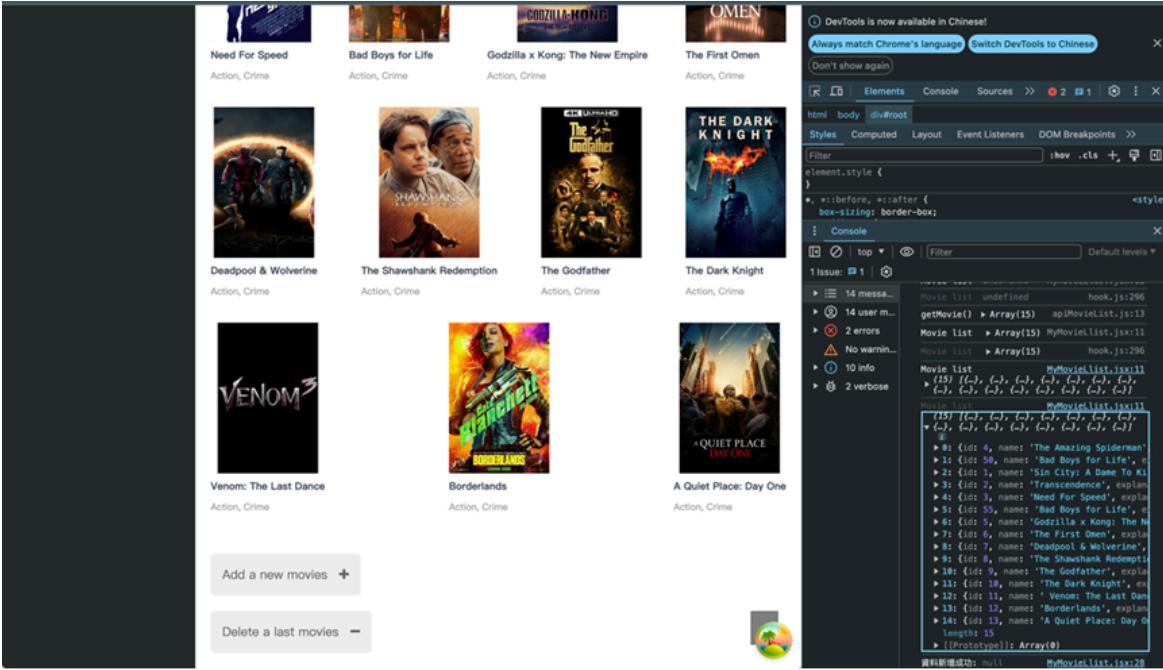
A screenshot of a web browser displaying a login page. The URL in the address bar is `localhost:5173/login`. The page has a dark blue background featuring a glowing network of nodes and lines. A central white login form is displayed, containing fields for 'Email address' (with the value `alex718296@gmail.com`) and 'Password', along with a red 'Login' button. The Windows taskbar at the bottom shows various pinned icons and the system tray.

能依標題顯示對應內容

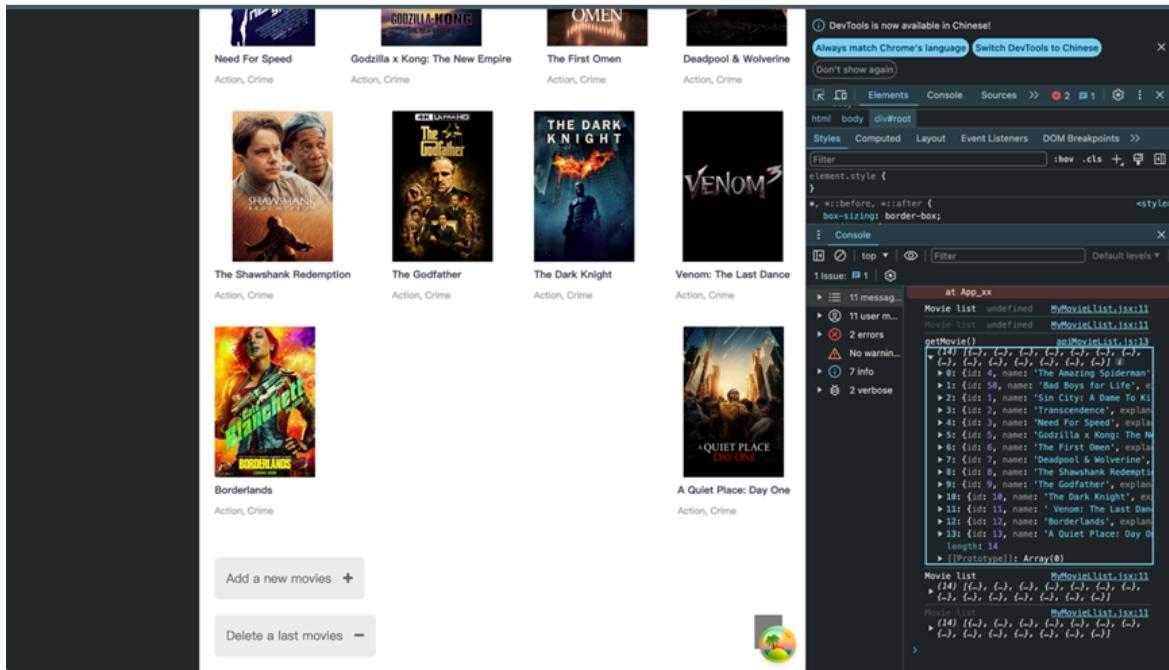
登入後："[/auth/my\\_movie\\_list](/auth/my_movie_list)"

能新增一筆電影資料，以及刪除最後一筆電影資料

**新增：**



**刪除:**



=> featured category, pages with routes

## 重點 3 -- Supabase 後台 schemas and data 支援

Table Editor

schema: public

Filtered by 1 rule

Sort

Insert

Format/Setting text

text text

img text

1 Auth policy

role postgres

Realtime off

API Docs

100 rows 4 records

Refresh Data Definition

https://supabase.com/dashboard/project/qfvxjrlpnpxdkvjlnnu/editor/30061

Table Editor

schema: public

Filtered by 1 rule

Sort

Insert

name text

explanation text

img text

formatid int8

video text

3 Auth policies

role postgres

Realtime off

API Docs

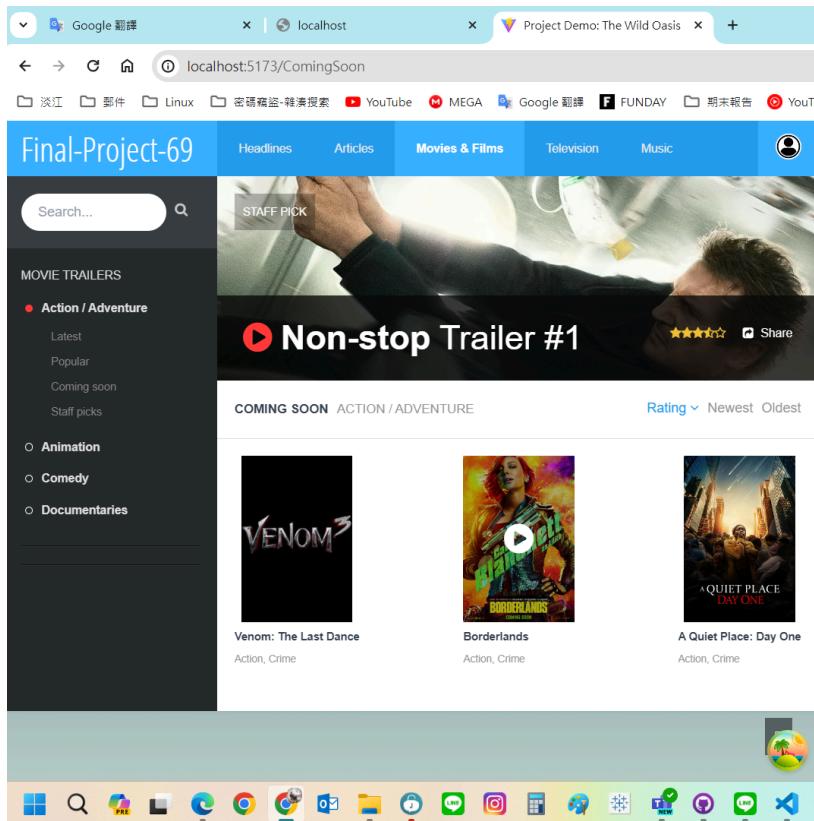
100 rows 16 records

Refresh Data Definition

https://supabase.com/dashboard/project/qfvxjrlpnpxdkvjlnnu/editor/30072

# 重點 4 -- React 前台 · code 模組設計

## 1.原始介面及資料:



The screenshot shows a movie review website with a dark header bar containing links for Headlines, Articles, Movies & Films (highlighted), Television, and Music. Below the header is a search bar and a sidebar with movie trailers categorized by genre: Action / Adventure (Latest, Popular, Coming soon, Staff picks) and Animation (Comedy, Documentaries). The main content area features a "STAFF PICK" section with a thumbnail for "Non-stop Trailer #1" and a rating of 5 stars. Below this are sections for "COMING SOON" movies, specifically ACTION / ADVENTURE. Three movies are listed: "Venom: The Last Dance" (Action, Crime), "Borderlands" (Action, Crime), and "A Quiet Place: Day One" (Action, Crime). Each movie has a thumbnail, title, and genre information.

The browser's developer tools are open, showing the "Console" tab with the following log output:

```
Popularity Data
  ▾ (3) [{}]
    ▾ 0:
      explanation: null
      formatted_id: 3
      id: 11
      img: "https://qfvxjr1pnpxdkvvl1nnu.supabase.co/storage/v1/object/public/proj"
      name: "Venom: The Last Dance"
      user_id: null
      video: null
    ▾ 1:
      explanation: null
      formatted_id: 3
      id: 12
      img: "https://qfvxjr1pnpxdkvvl1nnu.supabase.co/storage/v1/object/public/proj"
      name: "Borderlands"
      user_id: null
      video: null
    ▾ 2:
      explanation: null
      formatted_id: 3
      id: 13
      img: "https://qfvxjr1pnpxdkvvl1nnu.supabase.co/storage/v1/object/public/proj"
      name: "A Quiet Place: Day One"
      user_id: null
      video: null
  ▾ [[Prototype]]: Object
  ▾ 1:
    explanation: null
    formatted_id: 3
    id: 11
    img: "https://qfvxjr1pnpxdkvvl1nnu.supabase.co/storage/v1/object/public/proj"
    name: "Venom: The Last Dance"
    user_id: null
    video: null
  ▾ 2:
    explanation: null
    formatted_id: 3
    id: 12
    img: "https://qfvxjr1pnpxdkvvl1nnu.supabase.co/storage/v1/object/public/proj"
    name: "Borderlands"
    user_id: null
    video: null
  ▾ 3:
    explanation: null
    formatted_id: 3
    id: 13
    img: "https://qfvxjr1pnpxdkvvl1nnu.supabase.co/storage/v1/object/public/proj"
    name: "A Quiet Place: Day One"
    user_id: null
    video: null
  ▾ [[Prototype]]: Object
  ▾ 1:
    constructor: f Object()
    hasOwnProperty: f hasOwnProperty()
    isPrototypeOf: f isPrototypeOf()
    propertyIsEnumerable: f propertyIsEnumerable()
    toLocaleString: f toLocaleString()
    toString: f toString()
    valueOf: f valueOf()
  ▾ _defineGetter_: f __defineGetter__()
  ▾ _defineSetter_: f __defineSetter__()
  ▾ _lookupGetter_: f __LookupGetter__()
  ▾ _lookupSetter_: f __LookupSetter__()
  ▾ __proto__: (...)

  ▾ get __proto__: f __proto__()
  ▾ set __proto__: f __proto__()

  ▾ [[Prototype]]: Object
```

## 2.路由新增:

The screenshot shows the Visual Studio Code interface with the following details:

- File Explorer (Left):** Shows the project structure. The `src` folder contains `assets`, `css`, `data`, `features`, `pages`, `services`, `styles`, `ui`, and `utils`. The `pages` folder is expanded, showing files like `index.jsx`, `home.jsx`, `App.jsx`, `main.jsx`, `App.css`, `home.css`, `index.css`, `.env`, `.eslintrc.cjs`, `.gitignore`, and `image.png`.
- Code Editor (Top Right):** Displays the `App.jsx` file. A red box highlights the following import statements:

```
import { Popular, Latest, StaffPicks, ComingSoon, MyMovieList, Login_xx, PageNotFound_xx, Settings_xx, NewUsers_xx } from './pages';
```
- Terminal (Bottom):** Shows the output of the Vite build process:

```
> w13-wild-oasis-demo-xx@0.0.0 dev
> vite

Re-optimizing dependencies because vite config has changed

VITE v5.2.11 ready in 266 ms

→ Local: http://localhost:5173/
→ Network: use --host to expose
→ press h + enter to show help
```

### 3.supa 新增(Code):

The screenshot shows the VS Code interface with the following details:

- File Explorer (Left):** Shows the project structure. The `src` folder contains `assets`, `css`, `data`, `features`, `pages`, `index.js`, and `Latest.jsx`. The `pages` folder contains `open`, `Account_xx.jsx`, `Bookings_xx.jsx`, `Cabins_xx.jsx`, `ComingSoon.jsx`, and `MyMovieList.jsx`.
- Code Editor (Center):** Displays the `MyMovieList.jsx` file. Two sections of code are highlighted with red boxes:
  - The first section (lines 46-59) uses `useEffect` to fetch the last item from the `'movie_name'` table.

```
const MyMovieList = () => {
  useEffect(() => {
    const deleteData = async () => {
      // 獲取Supabase表格中的最後一筆資料
      const { data: lastData, error } = await supabase
        .from('movie_name')
        .select('id')
        .order('id', { ascending: false })
        .limit(1);

      if (error) {
        console.error(error);
        return;
      }
```
  - The second section (lines 61-75) performs a delete operation on the same table.

```
// 刪除最後一筆資料
const { data: deletedData, error: deleteError } = await supabase
  .from('movie_name')
  .delete()
  .match({ id: lastData[0].id });

if (deleteError) {
  console.error(deleteError);
  return;
}

console.log('資料刪除成功:', deletedData);
} catch (error) {
  console.error(error);
}
```
- Terminal (Bottom):** Shows the command line output for running the application.

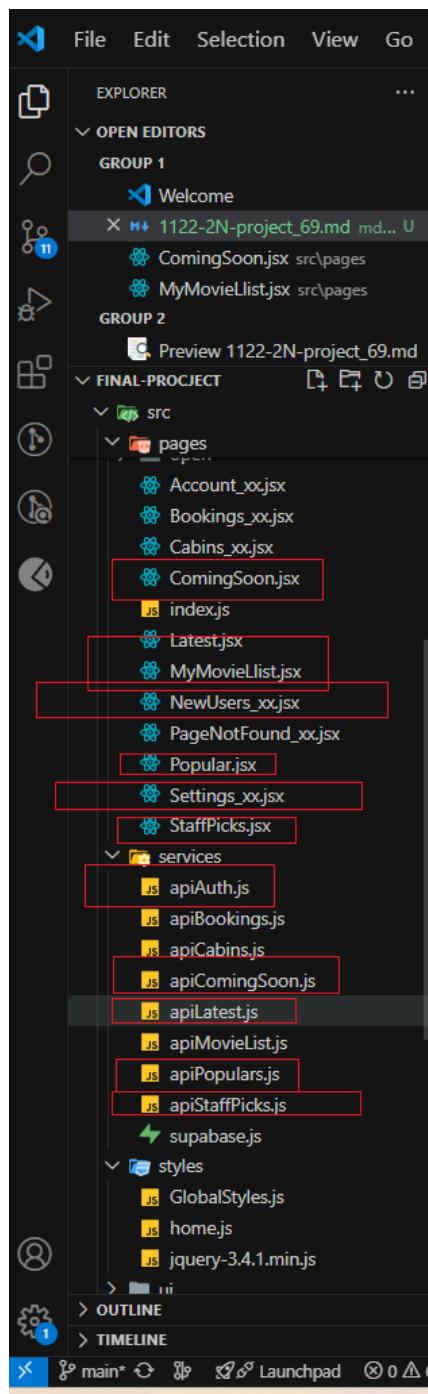
```
w13-wild-oasis-demo-xx@0.0.0 dev
> vite

Re-optimizing dependencies because vite config has changed

VITE v5.2.11 ready in 266 ms

→ Local: http://localhost:5173/
→ Network: use --host to expose
→ press h + enter to show help
```

#### 4. 頁面設計(Code):



## 5.頁面設計-Popular(Code):

The screenshot shows the Visual Studio Code interface with the following details:

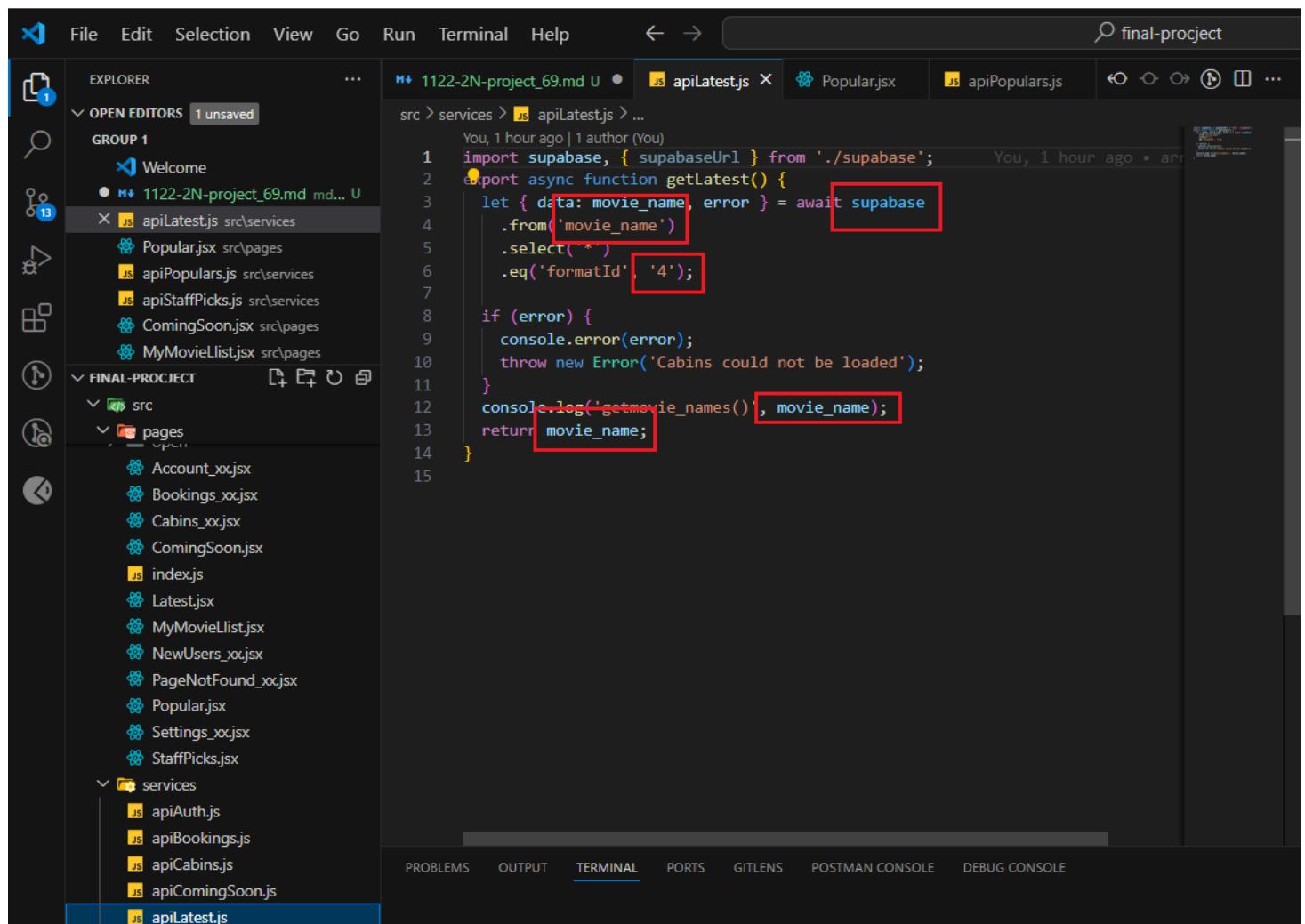
- File Explorer:** Shows the project structure under "OPEN EDITORS".
  - GROUP 1:** Welcome, 1122-2N-project\_69.md, ComingSoon.jsx, MyMovieList.jsx.
  - GROUP 2:** Preview 1122-2N-project\_69.md.
- Terminal:** The title bar shows "final-project".
- Code Editor:** The file "apiPopulars.js" is open, showing the following code:

```
import supabase, { supabaseUrl } from './supabase';
export async function getPopular() {
  let { data: movie_name, error } = await supabase
    .from('movie_name')
    .select('*')
    .eq('formatId', '1');

  if (error) {
    console.error(error);
    throw new Error('Cabins could not be loaded');
  }
  console.log('getmovie_names()', movie_name);
  return movie_name;
}
```

Specific lines of code are highlighted with red boxes:
  - Line 2: `let { data: movie_name, error } = await supabase`
  - Line 4: `.from('movie_name')`
  - Line 5: `.select('*')`
  - Line 6: `.eq('formatId', '1');`
  - Line 13: `return movie_name;`
- Bottom Navigation:** PROBLEMS, OUTPUT, TERMINAL (underlined), PORTS, GITLENS, POSTMAN CONSOLE, DEBUG CONSOLE.

## 6.頁面設計-Latest(Code):



The screenshot shows the Visual Studio Code interface with the following details:

- File Explorer:** Shows the project structure under "FINAL-PROJECT".
  - src:** Contains "pages" (Account\_xx.jsx, Bookings\_xx.jsx, Cabins\_xx.jsx, ComingSoon.jsx, index.jsx, Latest.jsx, MyMovieList.jsx, NewUsers\_xx.jsx, PageNotFound\_xx.jsx, Popular.jsx, Settings\_xx.jsx, StaffPicks.jsx).
  - services:** Contains "apiAuth.js", "apiBookings.js", "apiCabinsjs", "apiComingSoon.js", and "apiLatest.js".
- Code Editor:** The file "apiLatest.js" is open, showing the following code:

```
You, 1 hour ago | 1 author (You)
1 import supabase, { supabaseUrl } from './supabase';
2 export async function getLatest() {
3   let { data: movie_name, error } = await supabase
4     .from('movie_name')
5     .select('*')
6     .eq('formatId', '4');
7
8   if (error) {
9     console.error(error);
10    throw new Error('Cabins could not be loaded');
11  }
12  console.log('getmovie_names()', movie_name);
13  return movie_name;
14}
```
- Terminal:** The terminal tab is visible at the bottom.

#### 4. 頁面設計 - StaffPicks(Code):

The screenshot shows the Visual Studio Code interface with the following details:

- File Bar:** File, Edit, Selection, View, Go, Run, Terminal, Help.
- Search Bar:** final-project
- Explorer:** Shows the project structure:
  - OPEN EDITORS: 1 unsaved
  - 1122-2N-project\_69.md
  - apiMovieList.js
  - apiLatest.js
  - Popular.jsx
  - apiPopulars.js
  - apiStaffPicks.js (selected)
  - ComingSoon.jsx
  - MyMovieList.jsx

GROUP 2

  - FINAL-PROJECT
    - src
      - index.js
      - Latest.jsx
      - MyMovieList.jsx
      - NewUsers\_xx.jsx
      - PageNotFound\_xx.jsx
      - Popular.jsx
      - Settings\_xx.jsx
      - StaffPicks.jsx
    - pages
      - Cabins\_xx.jsx
      - ComingSoon.jsx
  - services
    - apiAuth.js
    - apiBookings.js
    - apiCabins.js
    - apiComingSoon.js
    - apiLatest.js
    - apiMovieList.js
    - apiPopulars.js

- Editor:** The code for `apiStaffPicks.js` is displayed. The code uses Supabase to get staff picks from a database table. Lines 3 through 13 are highlighted with red boxes, indicating the portion of the code being discussed or annotated.
- Bottom Bar:** PROBLEMS, OUTPUT, TERMINAL (underlined), PORTS, GITLENS, POSTMAN CONSOLE, DEBUG CONSOLE.

## **重點 5 -- 整體完成度**

**大約 90%**

## **資料及網路資源來源說明**

**網路資源來源 1**

**網路資源來源 2**

<https://supabase.com/docs/reference/javascript/delete>

<https://supabase.com/docs/reference/javascript/upsert>

# 後端 Supabase 資料庫設計

## SQL schema and data

=> table 1

The screenshot shows the Supabase Table Editor interface. On the left, the sidebar lists various tables: blog\_69, bookings, bookings\_xx, cabins, cabins\_xx, card\_69, guests, guests\_xx, mid2\_blog\_69, movie\_format, movie\_name, settings, and antispam. The 'movie\_format' table is currently selected and highlighted with a red box around its name in the 'Update table' field.

The main area displays the schema for the 'movie\_format' table. It has four columns:

Name	Type	Default Value	Primary
id	# int8	NULL	✓
Format/Setting	T text	NULL	✗
text	T text	NULL	✗
img	T text	NULL	✗

A red box highlights the column definitions in the 'Columns' section. There is also a red box around the 'Format/Setting' column header.

## => table 2

The screenshot shows the Supabase Table Editor interface. On the left, the sidebar lists various tables in the schema: public, blog\_69, bookings, bookings\_xx, cabins, cabins\_xx, card\_69, guests, guests\_xx, mid2\_blog\_69, movie\_format, movie\_name, settings, settings\_xx, and task\_69. The 'movie\_name' table is currently selected and highlighted with a red box.

The main area displays the 'movie\_name' table with 16 records. The columns are: id (int8), name (text), explanation (text). The data includes titles like 'The Amazing Spiderman', 'Bad Boys for Life', etc.

On the right, the 'Update table movie\_name' section is shown. It includes an 'ML Documentation' link, a checkbox for 'Enable Realtime' (unchecked), and a note about broadcasting changes to authorized subscribers. Below this is the 'Columns' configuration panel, also highlighted with a red box. It lists the columns with their types: id (int8), name (text), explanation (text), img (text), formatId (int8), video (text), and user\_id (uuid). The 'Primary' key status is indicated by a green checkmark next to the id column.

=> 外部鍵及 RLS 設定

=> 提供執行一次就可重新建立 schema 及 data 之 SQL 指令

# **前端程式設計說明**

## **=> 功能 1 及畫面設計，完成度說明**

如前圖所示

## **=> 功能 2 及畫面設計，完成度說明**

如前圖所示

## **=> 功能 3 及畫面設計，完成度說明**

如前圖所示

## **=> 功能 4 及畫面設計，完成度說明**

如前圖所示

## **=> 功能 5 及畫面設計，完成度說明**

如前圖所示

# 解決問題說明

## => 問題 1

畫面的整合配置，路蠻多時間

## => 問題 2

登入登出問題上網找了些資料，但是最後還是以老師上課的為主

## => 問題 3

製作分頁顯示資料，出現蠻多問題，資料庫權限還有 fk 的配置要衝新思考會比較完善

## => 問題 4

時間不足問題，能做的東西變得很有限

## => 問題 5

完成新增刪除跟助教討論過，並查看過資網路資料，產生想發及概念，並希望之後時間上能夠可以更完善客製化

# **學習甘苦談(學習歷程及感想)**

## **=> 1.創造力的發揮：**

製作網頁是一個充滿創造力的過程。選擇設計網頁的佈局、顏色、字體和圖像等元素，以創造出獨特而吸引人的網頁。

## **=> 2.技術的學習：**

製作網頁需要學習各種技術，可能對初學者或是尚不完全熟悉的人來說要有一定的學習，需要花費時間和精力來理解和應用。

## **=> 3.實現功能和互動性：**

製作網頁的過程中，可以添加各種功能和互動性。

## **=> 4.相容性：**

不同的瀏覽器可能對網頁的顯示和支持有所不同。這可能需要進行測試和調整，以確保您的網頁在各種瀏覽器上都能夠正常運行。

## **=> 5.實現成果：**

當成功創建一個令人讚嘆的網頁並將其發佈時，感到極大的成就感和喜悅。作品將能夠被訪問和欣賞，這是一種令人滿足的感覺。

---

-