

逢甲大學學生報告 ePaper

煮不煮道

Cook No Cook Know

作者：黃奕慈、楊舒晴、洪毓琳、劉佳宜、黃靜儀

系級：中文二

學號：D1362607、D1325271、D1325309、D1362624、D1390236

開課老師：曾智義老師

課程名稱：人本 AI 設計思維

開課系所：人文社會學院

開課學年：114 學年度 第 1 學期

FCU ePaper



中文摘要

大學生離家就學後，要面臨生活開銷與飲食健康問題。由於外食費用高昂，且營養成分難以掌握，若能培養自煮習慣，不僅能有效降低支出，也有助於建立健康的飲食型態。然而，多數大學生因缺乏烹飪經驗、飲食知識與菜單規劃能力，而難以長期實行自煮。因此，本專題嘗試結合人工智慧工具，設計一套能依使用者身體素質與現有食材，自動產生合適食譜的系統，以降低自煮門檻並提升實用性。

在研究過程中，我們運用 ChatGPT 與 Gemini 兩種生成式 AI 協助撰寫程式碼，並以 Google Colab 作為主要開發與執行平台。使用者可輸入身高、體重等身體數據，以及手邊現有的食材資訊，系統便能進行分析，產生符合需求的圖文食譜。此外，我們也加入圖片辨識功能，讓使用者上傳冰箱內食材照片，即可自動辨識蔬菜名稱，並提供相對應的料理方式與示意圖像，同時搭配清楚的步驟說明，引導烹飪新手完成料理。

結果，本系統成功整合飲食分析、食譜生成與圖片辨識等功能，能有效協助大學生進行自煮規劃。此成果不僅有助於提升飲食健康、減少剩食浪費，也能讓使用者在自煮過程中獲得生活成就感。

關鍵字：

ChatGPT、健康飲食、Google colab、生成式人工智慧

Abstract

After leaving home to attend university, college students have to face issues related to living expenses and healthy eating. Since eating out is costly and the nutritional content is hard to control, developing the habit of cooking at home can not only effectively reduce expenses but also help establish a healthy eating pattern. However, most college students find it difficult to maintain home cooking due to a lack of cooking experience, dietary knowledge, and menu planning skills. Therefore, this project attempts to integrate artificial intelligence tools to design a system that can automatically generate suitable recipes based on users' physical condition and available ingredients, aiming to lower the barrier to home cooking and enhance its practicality.

During the research process, we used the generative AI tools ChatGPT and Gemini to assist in writing code, and Google Colab served as the main development and execution platform. Users can input physical data such as height and weight, along with information on ingredients they have on hand, and the system will analyze the input and generate recipes that meet their needs. Additionally, we incorporated image recognition functionality, allowing users to upload photos of the ingredients in their refrigerator. The system can automatically identify the vegetable names, provide corresponding cooking methods and illustrative images, and offer clear step-by-step instructions to guide cooking beginners in completing the dishes.

As a result, this system successfully integrates dietary analysis, recipe generation, and image recognition functions, effectively assisting college students in planning their meals. This achievement not only helps improve diet health and reduce food waste but also allows users to gain a sense of accomplishment through the cooking process.

Keyword :

ChatGPT, healthy eating, Google Colab, generative artificial intelligence

目 次

壹、議題規劃.....	3
貳、人本 AI 設計思考五階段	4
參、對程式碼的理解.....	17
肆、影片成果.....	20
伍、心得反思.....	21
陸、參考文獻.....	23



壹、議題規劃

- 上了大學搬離家鄉，在外有許多生活花費開銷，倘若在飲食上能自煮，或許能省下不少外食的費用，也能吃得健康，於是身為大學生的我們想利用工具幫忙產生能根據使用者的身體素質，分析並規劃適合的食譜，並詳細列出步驟引導烹飪新手，並加入圖片辨識功能讓使用者上傳冰箱中的食材照片即可獲得蔬菜名稱、料理方式與生成圖像。除了吃得健康、減少剩食，也能在自煮過程獲得生活上的成就感。
- 規劃使用工具：Chat GPT、Gemini、Google Colab

貳、人本 AI 設計思考五階段

一、同理心 (Empathize)

- 對象：缺乏烹飪經驗的大學生
- 目的：
 - 1.省下外食費用
 - 2.自己挑選食材，可以選不同的產地，減少食物碳足跡。
 - 3.可以多吃原型食物，吃的健康。
 - 4.能根據自己的需求、愛好，調味及控制餐點份量。
 - 5.可以知道廚房的整潔，衛生環境。
 - 6.不用排隊，跟其他人一起吃飯
 - 7.社恐友善
 - 8.可以獲得成就感
- 痛點：

- 1、大學生外食花費較高。
- 2、外食容易高油高鹽、營養不均衡。
- 3、外食的衛生環境無法掌握，可能導致腸胃不舒服。
- 4、外食的調味會隨著地區或店家而不同，可能會有口味上的不合。 ex：皮蛋豆腐是鹹的、熱的。
- 5.去市場不會辨別蔬菜但不敢詢問。
- 6.資料外洩

二、定義問題 (Define)

- 問題：

社恐的烹飪新手大學生到菜市場採購蔬菜時，可能不認識菜又不敢問；以及使用者該如何根據自身的身體素質，在家用合適的食譜自行料理健康的三餐。

- 需求：

- 1.簡單易上手的健康食譜。
- 2.食材需要少油少糖並且營養均衡。
- 3.份量與熱量控制。
- 4.適用於烹飪新手。

- 5.根據使用者的身體素質搭配適合的食譜。
- 6.協助使用者辨別蔬菜
- 7.更詳細的拆解烹飪步驟
- 8.完成料理後能給予鼓勵或回饋

三、構思 (Ideate)

1. 規劃菜單與採買、備料食材，節省時間。
- 2.開發一款可以讓使用者自行輸入身體需求，與要辨識的目標照片，自動生成符合營養需求的食譜，並附上詳細烹調步驟及成品圖片。新增 API key 有助於確保服務的安全性，並管理每個用戶的存取權限和用量。

四、原型 (Prototype)

提示語：根據以下內容產生一段簡單的 Python 程式碼

啟動後請求使用者輸入 Google Gemini API Key，讓使用者選擇上傳冰箱食材照片，或手動輸入食材文字。分析使用者輸入的身體資訊（性別、年齡、身高、體重、活動量），並計算 BMI 和 TDEE。根據資訊生成「一日三餐」的圖文菜單。

```
1: !pip install -U -q google-generativeai
2:
3: import google.generativeai as genai
4: from google.colab import files
5: from IPython.display import display, Image, Markdown,
HTML 6: import PIL.Image
7: import io
8: import json
9: import re
10: import urllib.parse
```

煮不煮道

```
11:
12: # =====
13: # 💡💡 自動修復：模型選擇器
14: # =====
15: def get_working_model():
16: """自動尋找可用的模型名稱，避免 404 錯誤"""
17: print("💡💡 正在尋找可用的 AI 模型...")
18: try:
19: # 列出所有支援 'generateContent' 的模型
20: available_models = [m.name for m in genai.list_models() if
'generateContent' in m.supported_generation_methods]
21:
22: # 優先順序：Flash > Pro > 其他
23: preferred_order = [
24: 'models/gemini-1.5-flash',
25: 'models/gemini-1.5-flash-latest',
26: 'models/gemini-1.5-flash-001',
27: 'models/gemini-1.5-pro',
28: 'models/gemini-pro'
29: ]
30:
31: # 挑選第一個匹配的
32: for preferred in preferred_order:
33: if preferred in available_models:
34: print(f"✅ 已自動選用模型：{preferred}") 35: return
genai.GenerativeModel(preferred) 36:
37: # 如果都沒有，就選清單裡的第一個
38: if available_models:
39: print(f"⚠️ 預設模型不可用，改用：{available_models[0]}") 40:
return genai.GenerativeModel(available_models[0]) 41:
42: except Exception as e:
43: print(f"⚠️ 無法自動列出模型（可能 API Key 有誤），將嘗試強制使用舊版
```

煮不煮道

```
'gemini-pro'。")
44:
45: return genai.GenerativeModel('gemini-pro')
46:
47: # =====
48: # 工具：JSON 清理函數
49: # =====
50: def clean_and_parse_json(text):
51: text = text.replace("```json", "").replace("```", "").strip()
52: try:
53: return json.loads(text)
54: except:
55: try:
56: if '[' in text and ']' in text:
57: match = re.search(r'\[.*\]', text, re.DOTALL) 58: if
match: return json.loads(match.group()) 59: if '{' in text
and '}' in text:
60: match = re.search(r'\{.*\}', text, re.DOTALL) 61: if
match: return json.loads(match.group()) 62: except:
63: pass
64: return None
65:
66: # =====
67: # 主程式
68: # =====
69: try:
70: # 1. 設定 API Key

71: api_key = input("請輸入你的 Google API Key (39 碼) :

").strip() 72: if not api_key: raise ValueError("✘ API Key 不能
為空！")
73:
74: genai.configure(api_key=api_key)
75:
```

```
76: # 呼叫自動修復函數來取得模型

77: model = get_working_model()
78:
79: # 2. 蔬菜圖片處理

80: print("\n--- 步驟 1/3: 冰箱食材 ---")

81: use_image = input("是否上傳圖片? (y/n) : ").strip().lower()
82: vegetable_info = "無特定食材"
83:
84: if use_image == "y":
85:     print("請選擇圖片上傳...")
86:     uploaded = files.upload()
87:     if uploaded:
88:         fn = next(iter(uploaded))
89:         img = PIL.Image.open(io.BytesIO(uploaded[fn])) 90:
91:         print("... AI 正在看圖片 ...")
92:         res = model.generate_content(["請列出圖片中的食材名稱，用逗號分
93:         隔。", img])
94:         vegetable_info = res.text
95:         print(f"識別結果: {vegetable_info}")
96:     else:
97:         print("未上傳，跳過。")
98: else:
99:     manual = input("請手動輸入食材 (或按 Enter 跳過) : ") 98:
100:     if manual: vegetable_info = manual
101:
102: # 3. 身體資訊處理

103: print("\n--- 步驟 2/3: 身體資訊 ---")

104: user_input = input("請輸入一句話描述身體狀況 (例: 男 30 歲 175/70 久
105: 坐) : ")
106: if not user_input: user_input = "男 30 歲 170 65 久坐"
107: " 104:
```

煮不煮道

```
105: print("... AI 正在分析你的身體數據 ...")
106:
107: profile_prompt = f"""
108: 任務：從字串 "{user_input}" 提取資訊。
109: 回傳格式：純 JSON。
110: 格式範例：
111: [{"gender": "男", "age": 25, "height": 170, "weight": 65, "activity": "久坐",
112: "allergies": "無", "dislikes": "無", "diseases": "無"}]
113: """
114: profile_res = model.generate_content(profile_prompt)
115: profile = clean_and_parse_json(profile_res.text)
116:
117: if not profile:
118: # 如果失敗，給予預設值以防程式崩潰
119: print("⚠ 無法解析身體資訊，使用預設值。")
120: profile = {"gender": "男", "age": 30, "height": 170, "weight": 65,
121: "activity": "久坐"}
122: h = float(profile.get('height', 170))
123: w = float(profile.get('weight', 65))
124: age = float(profile.get('age', 30))
125: bmi = w / ((h/100)**2)
126: bmr = 10*w + 6.25*h - 5*age + 5
127: tdee = bmr * 1.3
128:
129: print(f"✅ 分析完成：BMI={bmi:.1f}，TDEE 約
130: {tdee:.0f}")
131: # 4. 生成圖文菜單
132: print("\n--- ❓❓ 步驟 3/3：生成圖文食譜 ---")
```

煮不煮道

```
133:
134: menu_prompt = f"""
135: 角色：營養師。
136: 任務：根據 TDEE {tdee:.0f} kcal 和食材 "{vegetable_info}" 設計一日三餐。
137: 回傳格式：純 JSON Array。
138: 格式範例：
139: [
140:  {"meal": "早餐", "name": "燕麥粥", "en_name": "Oatmeal",
    "cal": "400kcal", "step": "煮熟即可"}},
141:  {"meal": "午餐", "name": "雞胸沙拉", "en_name": "Chicken Salad",
    "cal": "600kcal", "step": "切片拌勻"}
142: ]
143: 重要：en_name 必須翻譯成英文以生成圖片。
144: """
145:
146: menu_res = model.generate_content(menu_prompt)
147: menu_data = clean_and_parse_json(menu_res.text)
148:
149: if not menu_data:
150: print("✘ 菜單生成失敗，顯示原始文字：", menu_res.text)
151: else:
152: for item in menu_data:
153: kw = urllib.parse.quote(item.get('en_name', 'food') + ",
    realistic food photography, 4k")
154: img_url =
f"https://image.pollinations.ai/prompt/{kw}?width=600&height=300&nolog
o=true"
155:
156: display(HTML(f"""
157: <div style="background:#f9f9f9; padding:15px; border
    radius:10px; margin-bottom:15px; border-left: 5px solid
    #2ecc71;"> 158: <h3 style="margin:0;">{item.get('meal')} :
    {item.get('name')}</h3>
```

煮不煮道

```
159: 
```

```
160: <b>❖❖ 熱量 : </b>{item.get('cal')}<br> 161:
```

```
<b>❖❖ 做法 : </b>{item.get('step')}
```

```
162: </div>
```

```
163: """))
```

```
164:
```

```
165: except Exception as e:
```

```
166: print(f"\n❖❖ 錯誤 : {e}")
```

五、測試 (Test):

- 是否能產生合理餐點? 是
- 是否符合健康需求? 是
- 迭代調整或新增功能: 新增更詳細的菜單、蔬菜圖片識別、根據提供的蔬菜製作客製菜單、食譜完成圖片、API Key、菜單自動生成

第一次測試 請輸入你的 Google API Key (39 碼):

```
AIzaSyBaqJ4cN3TNJ71YJ1awKhxedRJtY0vr3JU
```

```
❖❖ 正在尋找可用的 AI 模型...
```

⚠ 預設模型不可用, 改用: models/gemini-2.5-flash

```
--- ❖❖ 步驟 1/3: 冰箱食材 ---
```

```
是否上傳圖片? (y/n) : y
```

```
請選擇圖片上傳...
```

Upload widget is only available when the cell has been executed in the current browser session. Please rerun this cell to enable.

Saving images (2).jpeg to images (2) (1).jpeg

```
... AI 正在看圖片 ...
```

識別結果：番茄，小番茄，紅椒，紅蘋果，西瓜，茄子，紫洋蔥，
紫甘藍，甜菜根，鳳梨，葡萄，羽衣甘藍，酪梨，小黃瓜，菠菜，
生菜，芹菜，青蘋果，青檸

--- ❖❖ 步驟 2/3：身體資訊 ---

請輸入一句話描述身體狀況（例：男 30 歲 175/70 久坐）：女
30 歲 165/70 久坐

... AI 正在分析你的身體數據 ...

分析完成：BMI=25.7，TDEE 約 2062

--- ❖❖ 步驟 3/3：生成圖文食譜 ---

早餐：活力綠色蔬果昔碗



❖❖ 熱量：550kcal

❖❖ 做法：將羽衣甘藍、菠菜、半顆酪梨、半顆青蘋果、青檸汁與
少量水 或植物奶放入果汁機打成果昔。倒入碗中，撒上幾片小番茄
和切碎的紫洋蔥，可依喜好加入燕麥片或奇亞籽增加飽足感。

午餐：地中海彩虹烤雞沙拉



◆◆ 熱量：760kcal

◆◆做法：雞胸肉（約 150g）煎熟後切片。將生菜、菠菜、紫甘藍、小黃瓜、芹菜、番茄、紅椒、烤茄子塊和甜菜根切塊。鳳梨丁、葡萄和紅蘋果片與上述食材混合。淋上以青檸汁、橄欖油、鹽和黑胡椒調製的油醋醬。

晚餐：香煎鮭魚佐繽紛烤時蔬



◆◆ 熱量：760kcal

◆◆做法：將茄子、紅椒、紫洋蔥、甜菜根切塊，與少量橄欖油、

鹽、黑胡椒混合後，以 200°C 烤至金黃（約 20-25 分鐘）。鮭魚片（約 150g）用少許油香煎。菠菜和羽衣甘藍與小番茄稍微拌炒。將鮭魚、烤時蔬和炒青菜擺盤，佐四分之一顆酪梨片，可搭配一小份藜麥或糙米飯。

第二次測試	<p>請輸入你的 Google API Key (39 碼) :</p> <p>AIzaSyBaqJ4cN3TNJ71YJ1awKhxedRJtY0vr3JU</p> <p>❖❖ 正在尋找可用的 AI 模型...</p> <p>⚠ 預設模型不可用，改用：models/gemini-2.5-flash</p> <p>--- ❖❖ 步驟 1/3: 冰箱食材 ---</p> <p>是否上傳圖片？(y/n) : n</p> <p>請手動輸入食材（或按 Enter 跳過）：牛排, 鮭魚</p> <p>--- ❖❖ 步驟 2/3: 身體資訊 ---</p> <p>請輸入一句話描述身體狀況（例：男 30 歲 175/70 久坐）：男 52 歲 168/70 久坐 痔瘡</p> <p>... AI 正在分析你的身體數據 ...</p> <p><input checked="" type="checkbox"/> 分析完成：BMI=24.8, TDEE 約 1944</p> <p>--- ❖❖ 步驟 3/3: 生成圖文食譜 ---</p>
-------	---

早餐：燕麥粥佐水果與堅果



◆◆ 熱量：400kcal

◆◆ 做法：將燕麥片加水或牛奶煮熟，放入您喜愛的水果切塊（如藍莓、香蕉）與少量堅果（如杏仁、核桃）即可。

午餐：嫩烤鮭魚佐藜麥飯與烤時蔬



◆◆ 熱量：680kcal

◆◆ 做法：鮭魚以鹽、黑胡椒調味後，烤箱或氣炸鍋烤至熟透。藜麥飯依照包裝指示煮熟。時蔬（如花椰菜、甜椒、蘆筍）淋上橄欖油，以鹽、胡椒調味後烤熟。

晚餐：香煎牛排佐地瓜泥與蒸綠花椰



◆◆ 熱量：720kcal

◆◆ 做法：牛排以少許橄欖油香煎至個人喜歡的熟度，靜置數分鐘後切片。地瓜蒸熟後搗成泥。綠花椰菜洗淨後蒸熟。

參、對程式碼的理解

架構說明

第 001~011 行 | 套件載入與 Colab 環境設定

- 功能：安裝與載入 Google Gemini API、資料上傳、圖片處理及環境工具。
- 流程：
 - 安裝必要套件 (google-generativeai)。
 - 匯入 genai 用於 AI 運算。
 - 匯入 Colab files 處理檔案上傳。
 - 匯入 PIL.Image、io 進行影像解碼與處理。
 - 匯入 display,HTML 用於筆記本介面渲染。

第 012~046 行 | 自動修復與模型選擇器

- 功能：確保程式碼的健壯性，自動挑選當前帳號可用的 Gemini 模型。
- 流程：
 - 列出 API Key 支援的所有模型。
- 按優先順序 (Flash -> Pro -> 其他) 自動匹配可用模型。
- 攔截 404 模型錯誤，提供回退機制 (Fallback)。

第 047~065 行 | 工具：JSON 清理與解析函數

- 功能：處理 AI 輸出的原始文字，確保後續資料處理不會報錯。
- 流程：
 - 移除 AI 回傳時夾帶的 Markdown 標籤 (如 ``json)``。
 - 使用正則表達式 (Regex) 提取 [或 { 之間的 JSON 內容。
 - 將清理後的文字轉換為 Python 字典/列表。

第 069~099 行 | API 設定與食材辨識層

- 功能：配置金鑰並從影像或文字中獲取食材資訊。
- 流程：
 - 讀取使用者輸入的 API Key 並配置。
 - 判斷是否上傳圖片：上傳後由 Gemini Vision 辨識食材清單；未上傳則手動輸入。
 - 儲存食材名稱於變數 vegetable_info。

第 100~130 行 | 身體資訊結構化與熱量計算

- 功能：將非結構化文字數位化，並計算個人營養代謝指數。
- 流程：
 - 輸入個人狀況描述。
 - 透過 AI 提取「年齡、身高、體重、性別」等數值並存為 JSON。
 - 根據數值套用科學公式計算 BMI、BMR 及 TDEE (每日總能消耗)。

第 131~166 行 | 圖文菜單生成與 HTML 渲染

- 功能：根據計算結果產出最終食譜，並呈現互動式視覺介面。
 - 流程：
 - 將食材與 TDEE 熱量需求丟給 AI 營養師。
 - AI 生成包含中英文菜名及熱量的 JSON 陣列。
 - 將英文菜名拼湊至 `pollinations.ai` 生成動態圖片網址。
 - 以 `display(HTML)` 將食譜與圖片封裝在 CSS 區塊中顯示。
-
- 關鍵邏輯：
 - 1. 使用者可以用「圖片」或「文字」產生蔬菜清單，直接影響菜單生成內容。
 - 2. 生理計算 (BMI/BMR/TDEE) 決定每日熱量需求與每餐熱量分配。
 - 3. 所有菜單的邏輯都依賴 `prompt` 語意規範，例如：每餐需包含至少一種蔬菜，需提供詳細料理步驟，過敏或疾病需調整。

參數與輸入

- 這段程式碼其實就是把三個不同的 AI 功能串接起來，像一條生產線：
 - **【第一步：看照片】** AI 幫你辨識冰箱裡有哪些菜，或是直接告訴它你有什麼菜。
 - **【第二步：算熱量】** 電腦讀取身高體重，一句話講完就好（例如：我是 35 歲女生，160 公分，很胖，不愛動），用數學公式算出一天可以吃多少。
- **【第三步：出食譜】** AI 當主廚，根據你有的菜和你的熱量限制，寫出三餐食譜，並現場「畫」出照片給你看。

限制與風險

- 使用者輸入的不可靠性：身高、體重若非數字將無法運行；活動量輸入非 1-5 會自動變 1.2 會不準。
- 圖片辨識結果可能誤判：模糊圖片或光線差，可能讓 AI 誤判蔬菜種類，菜單會基於錯誤蔬菜生成。
 - 生理計算公式有誤差：TDEE 僅估算，不適合醫療用途。

改進方向

1. 強化使用者輸入驗證

- 檢查 height、weight 是否為數字
- 活動量只能輸入 1-5
- 疾病需對應標準清單

2. 增加 AI 回傳格式驗證

- 檢查是否包含早/午/晚/點心
- 至少一種蔬菜出現在每餐
- 若不符合 → 自動讓 AI 重新生成一次

3. 強化圖片分析能力

肆、影片成果

使用 Gemini 生成影片，並用剪輯將其製作成影片。

影片連結：

https://youtu.be/8xpijdFtkYs?si=B_qJKX7j4Gg_Ya3p

伍、心得反思

劉佳宜：

藉由此次期末的專題報告，我們學會了如何讓這個程式設計更加完整，即使根本不懂程式語言的編碼，透過向 AI 工具下達合適的提示語，我們也能製作出一個作品。在我們的專題裡，這項工具不僅能幫助人類生活更好，更能給予情感上的價值，在現代生活許多人的煩惱就是三餐無法決定，加上長期外食的不健康，使用合適的 AI，不只解決了這個煩惱，過程也十分有趣。

黃澗儀：

在這次的專題中，我們透過結合 AI 生成程式碼，來為煩惱食物如何煮的大學生解決問題，畢竟住在外邊飲食上都需要自己解決，有時候根本想不到所剩的食材可以怎麼處理，所以藉 AI 所生成的程式碼，使用者僅需填入相關內容，就能為其打造專屬的食譜，而在過程中我們也不斷為提示語等等做修正，使得到的結果可以更接近我們所想要的，同時也讓我知道原來 AI 也能為我們解決這樣的問題。

黃奕慈：

這次的專題報告讓我學習到怎麼寫程式和更一步的運用 ai，第一次知道原來 ai 也可以幫忙寫程式，原本以為會失敗，但意外的成功了。使用 ai 編寫程式的過程中，我們一直有新的想法，所以一直請它加入新功能，有時候會失敗，但請它再確認程式碼一次後就成功了。做完的成果蠻實用的，我平時在家也會煮飯，有一次心血來潮，想著做都做了，那就不用一下吧，就使用了寫出來的程式碼，結果蠻滿意的，雖然沒辦法做出跟菜譜圖片一樣看起來美味的菜，但不難吃，而且都有用到我提到的菜，也有根據身體數值安排。

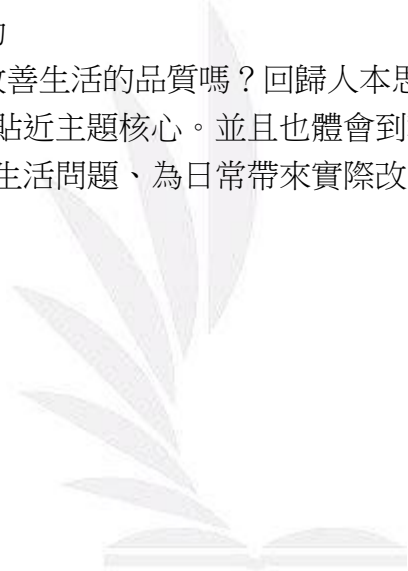
洪毓琳：

我覺得我們的主題非常實用與貼近生活，想要自煮但廚藝不精是許多人會遇到的困擾，而且很多網路食譜步驟都過於精簡，新手可能會不知所措，透過與 AI 協作可以煮出讓自己很有成就感的料理，在不知道要煮什麼的時候我曾經參考過生成的食譜，煮出來真的蠻好吃的。

楊舒晴：

這次的專題從前期的議題發想，到期末的專題影片製作與剪輯，讓我重新思考這個程式持續深化與實際應用的可能性。從影片的架構、台詞、影像呈現，嘗試站在使用者與客戶的角度思考：如果是我會希望這個程式任何幫助

我？這個程式能改善生活的品質嗎？回歸人本思維，理解人們真正的需求，使成果能更貼近主題核心。並且也體會到科技與設計不只將想像實現，更該回應生活問題、為日常帶來實際改變。



陸、參考文獻

曾智義（2025）。**Ch4：AI 如何推論答案：從設計思維到 Vibe Coding 原型**[授課講義]。逢甲大學 iLearn。

https://ilearn.fcu.edu.tw/pluginfile.php/3196301/mod_resource/content/3/1141_Ch4_AI%20%E5%A6%82%E4%BD%95%E6%8E%A8%E8%AB%96_%E5%B%E%9E%E8%A8%AD%E8%A8%88%E6%80%9D%E7%B6%AD%E5%88%B0%20Vibe%20Coding%20%E5%8E%9F%E5%9E%8B.pdf

曾智義（2025）。**Ch6：情感識別與慰藉 AI 設計**[授課講義]。逢甲大學 iLearn。

https://ilearn.fcu.edu.tw/pluginfile.php/3209403/mod_resource/content/2/1141_Ch6_%E6%83%85%E6%84%9F%E8%AD%98%E5%88%A5%E8%88%87%E6%85%B0%E8%97%89AI%E8%A8%AD%E8%A8%88.pdf