國立臺北商業大學

資 訊 管 理 系

113’資訊系統專案設計

**系統手冊**

****

**組 別：第113412組**

**題 目：美妝小幫手**

**指導老師：蔡文隆老師**

**組 長：11046078張明蓉**

**組 員：11046068陳力榆 11046074 柯恩璇**

**11046075郭岳鋆**

**中華民國113年10月16日**

**目錄**

[第1章 前言 5](#_Toc179844026)

[1-1 背景介紹 5](#_Toc179844027)

[1-2 動機 5](#_Toc179844028)

[1-3 系統目的與目標 5](#_Toc179844029)

[1-4 預計成果 5](#_Toc179844030)

[第2章 營運計畫 6](#_Toc179844031)

[2-1 可行性分析 6](#_Toc179844032)

[2-2 商業模式Business model 7](#_Toc179844033)

[2-3 市場分析-STP 9](#_Toc179844034)

[2-4 競爭力分析 SWOT-TOWS 9](#_Toc179844035)

[第3章 系統規格 10](#_Toc179844036)

[3-1 系統架構 10](#_Toc179844037)

[3-2 系統軟、硬體需求與技術平台 11](#_Toc179844038)

[3-3 使用標準工具 11](#_Toc179844039)

[第4章 專案時程與組織分工 12](#_Toc179844040)

[4-1 專案時程 12](#_Toc179844041)

[4-2 專案組織與分工 13](#_Toc179844042)

[第5章 需求模型 15](#_Toc179844043)

[5-1 使用者需求 15](#_Toc179844044)

[5-2 使用個案圖 16](#_Toc179844045)

[5-3 使用個案描述 17](#_Toc179844046)

[5-4 分析類別圖（Analysis class diagram） 24](#_Toc179844047)

[第6章 設計模型 25](#_Toc179844048)

[6-1 循序圖 25](#_Toc179844049)

[6-2 設計類別圖 26](#_Toc179844050)

[第7章 實作模型 27](#_Toc179844051)

[7-1 佈署圖(Deployment diagram) 27](#_Toc179844052)

[7-2 套件圖(Package diagram) 27](#_Toc179844053)

[7-3 元件圖(Component diagram) 27](#_Toc179844054)

[7-4 狀態機(State machine) 28](#_Toc179844055)

[第8章 資料庫設計 31](#_Toc179844056)

[8-1 資料庫關聯表 31](#_Toc179844057)

[8-2 表格及其Meta data 32](#_Toc179844058)

[第9章 程式 35](#_Toc179844059)

[9-1 表格說明 35](#_Toc179844060)

[第10章 測試模型 40](#_Toc179844061)

[10-1 測試說明 40](#_Toc179844062)

[第11章 操作手冊 41](#_Toc179844063)

[11-1 表格 41](#_Toc179844064)

[第12章 使用手冊 47](#_Toc179844065)

[12-1 入口網站 47](#_Toc179844066)

[12-2 產品分類 48](#_Toc179844067)

[12-3 會員頁面 49](#_Toc179844068)

[12-4 篩選條件 53](#_Toc179844069)

[第13章 感想 55](#_Toc179844070)

[第14章 參考資料 57](#_Toc179844071)

[附錄 58](#_Toc179844072)

**圖目錄**

[圖 2-1‑1系統架構圖 6](#_Toc179845050)

[圖 2-2‑1商業九宮格圖 7](#_Toc179845051)

[圖 3-1‑1後端架構圖 10](#_Toc179845052)

[圖 3-1‑2前端架構圖 10](#_Toc179845053)

[圖 4-1‑1甘特圖 12](#_Toc179845054)

[圖 4-1‑2 Github資料上傳記錄 12](#_Toc179845055)

[圖 5-2‑1使用個案圖-User 16](#_Toc179845056)

[圖 5-2‑2使用個案圖-管理者 16](#_Toc179845057)

[圖 5-3‑1肌膚檢測之活動圖 17](#_Toc179845058)

[圖 5-3‑2篩選條件之活動圖 18](#_Toc179845059)

[圖 5-3‑3篩選結果之活動圖 19](#_Toc179845060)

[圖 5-3‑4搜尋之活動圖 20](#_Toc179845061)

[圖 5-3‑5產品資訊之活動圖 21](#_Toc179845062)

[圖 5-3‑6新增評論之活動圖 22](#_Toc179845063)

[圖 5-3‑7管理者之活動圖 23](#_Toc179845064)

[圖 5-4‑1分析類別圖 24](#_Toc179845065)

[圖 6-1‑1循序圖 25](#_Toc179845066)

[圖 6-1‑2首頁產品資訊循序圖 25](#_Toc179845067)

[圖 6-2‑1設計類別圖 26](#_Toc179845068)

[圖 7-1‑1佈署圖 27](#_Toc179845069)

[圖 7-2‑1套件圖 27](#_Toc179845070)

[圖 7-3‑1元件圖 27](#_Toc179845071)

[圖 7-4‑1肌膚檢測狀態圖 28](#_Toc179845072)

[圖 7-4‑2搜尋商品狀態圖 28](#_Toc179845073)

[圖 7-4‑3查詢價格狀態圖 29](#_Toc179845074)

[圖 7-4‑4查詢歷史記錄狀態圖 30](#_Toc179845075)

[圖 7-4‑5新增評論狀態圖 30](#_Toc179845076)

[圖 8-1‑1資料庫關聯圖 31](#_Toc179845077)

[圖 12-1‑1入口網站 47](#_Toc179845078)

[圖 12-1‑2產品分類及前三評分產品 47](#_Toc179845079)

[圖 12-2‑1產品分類頁面 48](#_Toc179845080)

[圖 12-2‑2產品詳情以及可收藏產品 48](#_Toc179845081)

[圖 12-3‑1會員註冊 49](#_Toc179845082)

[圖 12-3‑2會員登入 49](#_Toc179845083)

[圖 12-3‑3肌膚檢測 49](#_Toc179845084)

[圖 12-3‑4肌膚檢測完成並依照膚質推薦匹配度最高產品畫面 50](#_Toc179845085)

[圖 12-3‑5會員頁面之肌膚檢測記錄 50](#_Toc179845086)

[圖 12-3‑6會員頁面之關注收藏 51](#_Toc179845087)

[圖 12-3‑7會員頁面之關於我 51](#_Toc179845088)

[圖 12-3‑8會員頁面之新增評論 52](#_Toc179845089)

[圖 12-3‑9會員頁面之我的評論 52](#_Toc179845090)

[圖 12-4‑1篩選條件(登入狀態) 53](#_Toc179845091)

[圖 12-4‑2篩選結果 53](#_Toc179845092)

[圖 12-4‑3篩選條件(未登入狀態) 54](#_Toc179845093)

**表目錄**

[表 2-3‑1市場分析-STP表 9](#_Toc179845025)

[表 2-4‑1競爭力分析 SWOT-TOWS表 9](#_Toc179845026)

[表 3-2‑1系統軟、硬體需求與技術平台表 11](#_Toc179845027)

[表 3-3‑1使用標準工具表 11](#_Toc179845028)

[表 4-2‑1專案組織與分工表 13](#_Toc179845029)

[表 4-2‑2專題成果工作內容與貢獻度表 14](#_Toc179845030)

[表 5-1‑1使用者需求表 15](#_Toc179845031)

[表 8-2‑1 Products–產品表 32](#_Toc179845032)

[表 8-2‑2 Users–使用者表 32](#_Toc179845033)

[表 8-2‑3 Reviews–評論表 33](#_Toc179845034)

[表 8-2‑4 NewReviewSummary–新評論摘要表 33](#_Toc179845035)

[表 8-2‑5 Favorites–收藏表 34](#_Toc179845036)

[表 8-2‑6 SkinTests–肌膚檢測結果表 34](#_Toc179845037)

[表 9-1‑1元件清單及其規格描述 35](#_Toc179845038)

[表 9-1‑2其他附屬之各種元件 35](#_Toc179845039)

[表 9-2‑1 html(首頁) 36](#_Toc179845040)

[表 9-2‑2 html(會員專區) 37](#_Toc179845041)

[表 9-2‑3 Flask(連接前後端) 38](#_Toc179845042)

[表 9-2‑4 AI簡化資料 39](#_Toc179845043)

[表 10-1‑1說明採用之測試方法及其進行方式 40](#_Toc179845044)

[表 10-1‑2測試個案與測試結果資料 40](#_Toc179845045)

[表 11-1‑1 操作手冊表 41](#_Toc179845046)

[表 11-1‑2系統元件介紹表 43](#_Toc179845047)

[表 11-1‑3安裝流程表 43](#_Toc179845048)

[表 11-1‑4系統管理表 46](#_Toc179845049)

# 前言

## 背景介紹

近年來，隨著科技的進步和社交媒體的盛行，美妝推薦系統已成為消費者選擇美妝產品的重要工具。這些系統利用機器學習和大數據分析技術，根據消費者的特徵和偏好提供個性化的推薦，解決消費者在眾多選擇中的困惑和不確定性。這一新興技術使購物更加便利，也為美妝產業帶來了發展的機遇。品牌和零售商能夠借助這些系統更好地了解市場需求、調整策略以提高銷售效率。

## 動機

建立美妝小幫手的動機來自對消費者體驗和美妝產業發展的關注。消費者在選購美妝產品時往往缺乏專業知識，易受到虛假評論和廣告的影響，導致購買不適合自己的產品，不僅影響了消費者的購物體驗和滿意度，也可能導致皮膚問題。因此，我們希望建立一個智能的美妝推薦系統，利用技術為消費者提供準確、個性化的產品推薦，從而提升其購物效率和滿意度。

同時，建立美妝推薦系統可促進美妝產業的創新和發展。隨著科技的進步，人工智慧、大數據分析等技術已廣泛應用於各行各業，美妝產業也不例外。透過這些技術建立的推薦系統能更好地了解消費者需求和偏好，有助於產品研發和市場推廣的精準化，從而提高產業的競爭力和創新能力。

## 系統目的與目標

美妝小幫手的目的在於為消費者提供個性化、精準的美妝產品推薦，幫助使用者更快速、有效地找到適合自己的產品。透過分析消費者的個人特徵和偏好，系統能夠提供相應的推薦，從而提升消費者的購物體驗，減少購買過程中的挑戰和不確定性。

提供個性化推薦：目標是提供個性化的美妝產品推薦，以滿足消費者的需求。透過分析消費者的皮膚類型、喜好等資訊，系統能夠準確地匹配產品，為消費者提供精準的推薦。

提高購物效率：希望能透過美妝推薦系統幫助消費者更快速地找到適合自己的產品，從而提高購物效率。無需花費大量時間搜索和比較產品，只需通過系統的推薦即可快速找到適合自己的產品。

提升消費者滿意度：通過提供準確的美妝產品推薦，提升消費者的滿意度。消費者可以更加信任系統的推薦，並更滿意其購物體驗，進而增強品牌忠誠度。

## 預計成果

提高消費者滿意度：系統提供個性化的美妝產品推薦，提升購物體驗，增強對品牌的信任和忠誠度。

增加品牌銷售量：推薦更貼合消費者需求的產品，提高銷售量和市場佔有率，促進銷售業績增長。

提升消費者忠誠度：提供個性化服務，增強消費者對品牌的信任感，建立良好的消費者忠誠度。

降低產品退貨率：提供準確的產品推薦，減少退貨情況，降低了產品的退貨率，提升品牌的經濟效益。

# 營運計畫

## 可行性分析

1. 技術可行性：

開發美妝推薦分析模型，我們會搭配內部一致信度的Cronbach’s α配合Datsbase、網站建構的程式技術，搭配機器學習和數據挖掘技術（Data Mining）來撰寫模型，並透過收集大量的商品數據來分析和優化推薦算法。確保能維持系統可靠性、可擴展、可維護的特性，此外也會選擇穩定安全的工具、平台、應用，確保系統能持續運作。

一張含有 文字, 螢幕擷取畫面, 字型, 圖表 的圖片

自動產生的描述

圖 2-1‑1系統架構圖

1. 時間可行性：

由於美妝用品眾多，客觀地來說無法全部分析完畢，因此本專題採用挑選3個品牌全部類別的產品進行美妝推薦模型分析比較。期望未來能有更多的時間、資料充實此模型。

1. 經濟可行性：

此平台系統是非營利網站，不排除未來有更多的擴展性發展，本系統開發成本的包含人力、技術、數據收集、處理數據的時間成本。同時也須評估系統所創造的經濟效益，是否真的有滿足使用者需求。

1. 法律可行性：

為了保護用戶隱私權益，本系統不收集使用者隱私資料，使用系統推薦產品時需重新檢測、提出需求。

在消費者權益方面，本推薦系統的建議應基於客觀的標準，不會受到品牌偏好的影響。

1. 市場潛在性：

如今市場上並沒有一個專門的個性化推薦模型，大多僅提供附加功能，主要是根據用戶的歷史瀏覽和購買記錄等數據，分析後提供給用戶做為參考推薦，另外也有專業團隊量身訂製服務，缺點則是價格高，隨者季節、時間等外在因素改變，導致影響到需求本身以及時代性差異產生認知不一致的問題。而我們的產品可以有效的根據使用者目前的個人特徵、偏好、皮膚類型等資訊去做推薦。

1. 操作可行性：

由於我們的客群是使用美妝產品的用戶，因此本系統的目標是讓所有年齡層的人都可以輕易上手，介面設計應當簡單直覺讓使用者能簡易操作，像是透過標籤分類、題目設計應明確、直接了當，使用戶能清楚理解並做出有效回應。

## 商業模式Business model

目前，我們的系統以非營利性質為主，專注於提供用戶高效、個性化的美妝產品推薦服務，目的是幫助使用者解決購物時的困惑，找到最適合自己的產品。我們希望藉由技術的創新與數據分析，建立一個讓用戶信任的平台，同時提高對品牌合作夥伴的價值貢獻。

與 Dcard 的商業模式相比，Dcard 從初期的小團隊經營開始，成功將其定位為一個匿名社群平台，並隨著用戶規模的增長，逐步轉型為盈利模式，收入主要來自廣告、聯名產品銷售以及其他商業合作。我們的系統雖然目前尚未涉及盈利，但在未來發展方向上，我們也將參考 Dcard 的成功經驗，逐步探索品牌合作、廣告收入以及聯盟行銷等潛在收益模式，從而實現平台的可持續發展。

這樣的比較，不僅讓我們清楚當前的定位，也能為未來的發展奠定堅實的基礎。

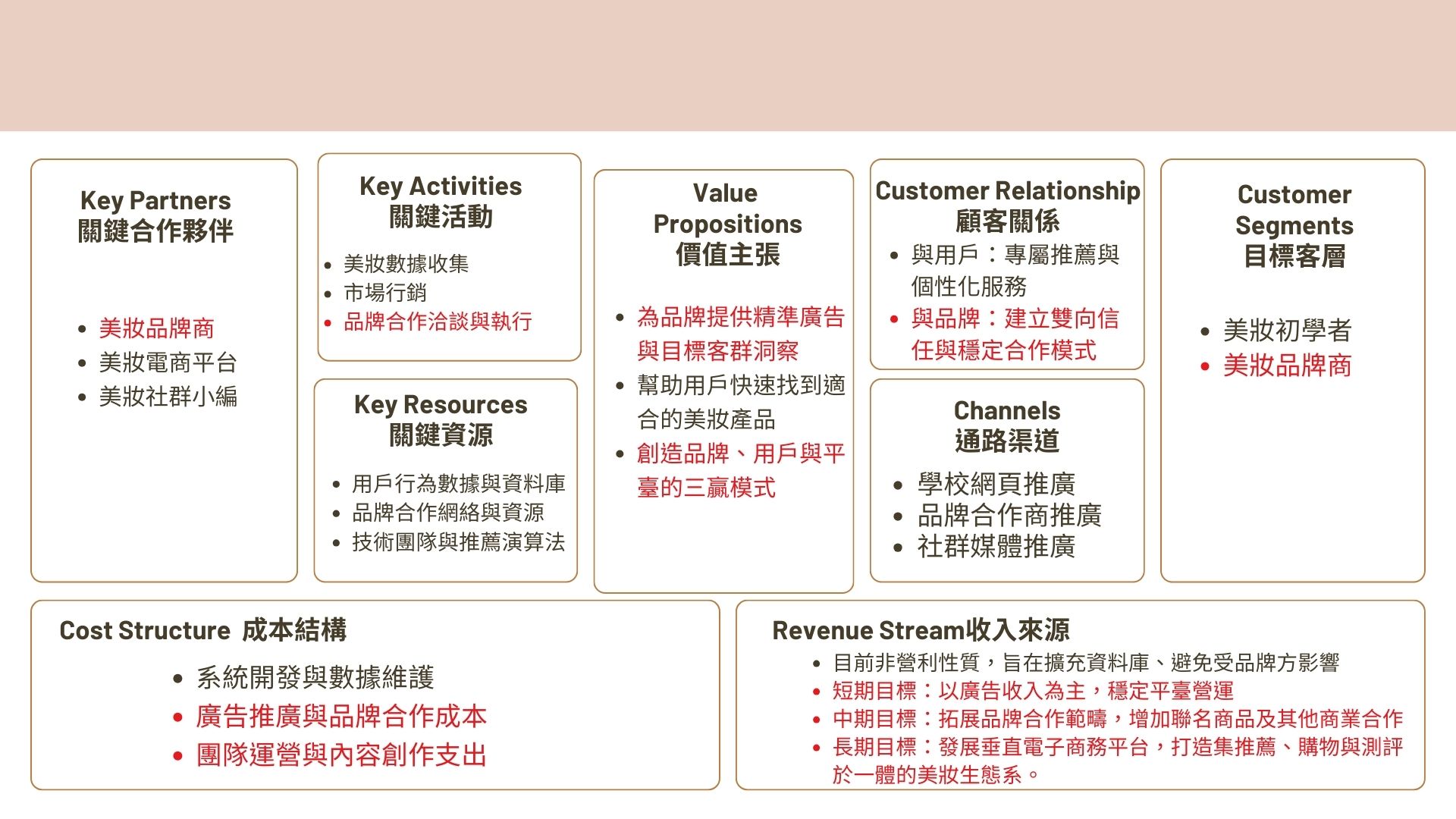


圖 2-2‑1商業九宮格圖

1. 目標客層（Customer Segments）：

系統的目標客群為主要以美妝初學者為主，希望能透過美妝小幫手找到適合自己的產品，解決購物時的困惑。

1. 價值主張（Value Propositions）：

系統主張為消費者提供個性化、精準的美妝產品推薦，幫助他們更快速、更有效地找到適合自己的產品。

1. 通路（Channels）：

系統的推廣管道由學校網頁推廣，原因為學校女生佔於多數、品牌合作商推廣另外還有我們人員社群媒體推廣。

1. 顧客關係（Customer Relationships）：

系統須對目標客群有誠實和給予使用者信任，使用者是藉由我們的推薦系統去選擇產品系統與目標客層建立互相信任的關係必然是重要的。

1. 收益流（Revenue Streams）：

目前，系統作為一個非營利平台，旨在為用戶提供方便的產品查詢服務。而未來我們計劃通過以下幾種方式實現營利：

1. 品牌合作：與美妝品牌合作，通過推薦品牌產品的方式收取佣金或合作費用。
2. 廣告收入：在平台上投放美妝相關的廣告，收取廣告費用。
3. 聯盟行銷（Affiliate Marketing）：與美妝產品銷售網站或平台合作，通過用戶從平台上點擊購買的產品鏈接獲取佣金。
4. 產品試用與評論：與品牌合作提供產品試用服務，並收集用戶的評論和反饋。品牌可以支付試用活動的費用。
5. 關鍵資源（Key Resources）：

系統關鍵資源包括資料庫管理技術、爬蟲、機器學習和數據挖掘技術（Data Mining）等。

1. 關鍵活動（Key Activities）：

系統關鍵活動為各美妝數據的收集、市場行銷、選擇納入的品牌。

1. 關鍵合作夥伴（Key Partnership）：

系統的合作夥伴為美妝電商平台、美妝社群小編，藉由互助的方式達到系統的宣傳的和美妝品的推廣。

1. 成本結構（Cost Structure）：

系統的成本為系統開發、系統維護、系統改善等成本。

## 市場分析-STP

表 2-3‑1市場分析-STP表

|  |  |
| --- | --- |
| 市場區隔(Segmenting) | 我們客群為對底妝有需求的使用者，包含美妝網站中被忽視或不充分滿足的特定客戶群體。 |
| 目標市場(Targeting) | 由於目標客群為對底妝有需求的使用者，因此我們會先專注於先了解客戶的膚質，在判斷消費者的偏好(價格、品牌…)和尋找特定風格或主題的客戶等，以個人化行銷的方式進入。 |
| 產品定位(Positioning) | 我們系統定位為能根據使用者的個人特質和需求，提供量身訂製的美妝產品建議。透過分析消費者的皮膚狀況、偏好以及產品需求等資訊，讓系統以機器學習的方式能夠準確地配對產品，以滿足消費者的需求。 |

## 競爭力分析 SWOT-TOWS

表 2-4‑1競爭力分析 SWOT-TOWS表

|  |  |
| --- | --- |
| SO  1.搜尋指定商品並分析匹配度  2.拓展新興市場、發展額外應用 | OW  1.加強保護措施  2.訓練模型提高精準度  3.利用社交媒體行銷提升知名度 |
| ST  1.透過專業分析提升差異化  2.使用會員專區記錄資料，省去來回重複測驗的時間 | WT  1.加強市場定位  2.定期更新維護資料數據庫 |

# 系統規格

## 系統架構

1.後端架構圖

一張含有 圖表, 圖畫, 行, 方案 的圖片

自動產生的描述

圖 3-1‑1後端架構圖

2.前端架構圖

一張含有 圖表, 方案, 工程製圖 的圖片

自動產生的描述

圖 3-1‑2前端架構圖

## 系統軟、硬體需求與技術平台

表 3-2‑1系統軟、硬體需求與技術平台表

|  |  |  |
| --- | --- | --- |
| 作業系統版本 | 最低系統需求 | 建議系統需求 |
| Windows10以上版本 | Windows10、11以上版本 |
| 處理器磁碟  可用空間 | 雙核心以上 | 四核心以上 |
| 1GB以上可用空間 | 2GB以上可用空間 |
| RAM網路 | 1GB以上可用記憶體 | 1GB以上可用記憶體 |
| Wi-Fi網路 | |

## 使用標準工具

表 3-3‑1使用標準工具表

|  |  |
| --- | --- |
| 系統開發輔助工具 | |
| 作業系統 | Windows10、11 |
| 開發系統 | Windows10、11、macOS14.4.1 |
| 資料庫 | MySQL |
| 網頁伺服器管理 | Nginx |
| 動態DNS | No-IP |
| 程式編輯器 | Visual Studio Code |
| 版本控制 | GitHub |
| 程式開發技術 | |
| 前端 | HTML、JavaScript、CSS |
| 後端 | Python、Colaboratory |
| 程式編輯器 | Visual Studio Code |
| 文件美工程式 | |
| 設計圖樣 | Adobe Photoshop 2023、Procreat |
| 流程設計 | Microsoft Office Visio |
| 文件 | Microsoft Office Word |
| 簡報 | Canva、Microsoft Office Power Point |

# 專案時程與組織分工

## 專案時程

一張含有 螢幕擷取畫面, 鮮豔, 文字, 繪圖 的圖片

自動產生的描述

圖 4-1‑1甘特圖

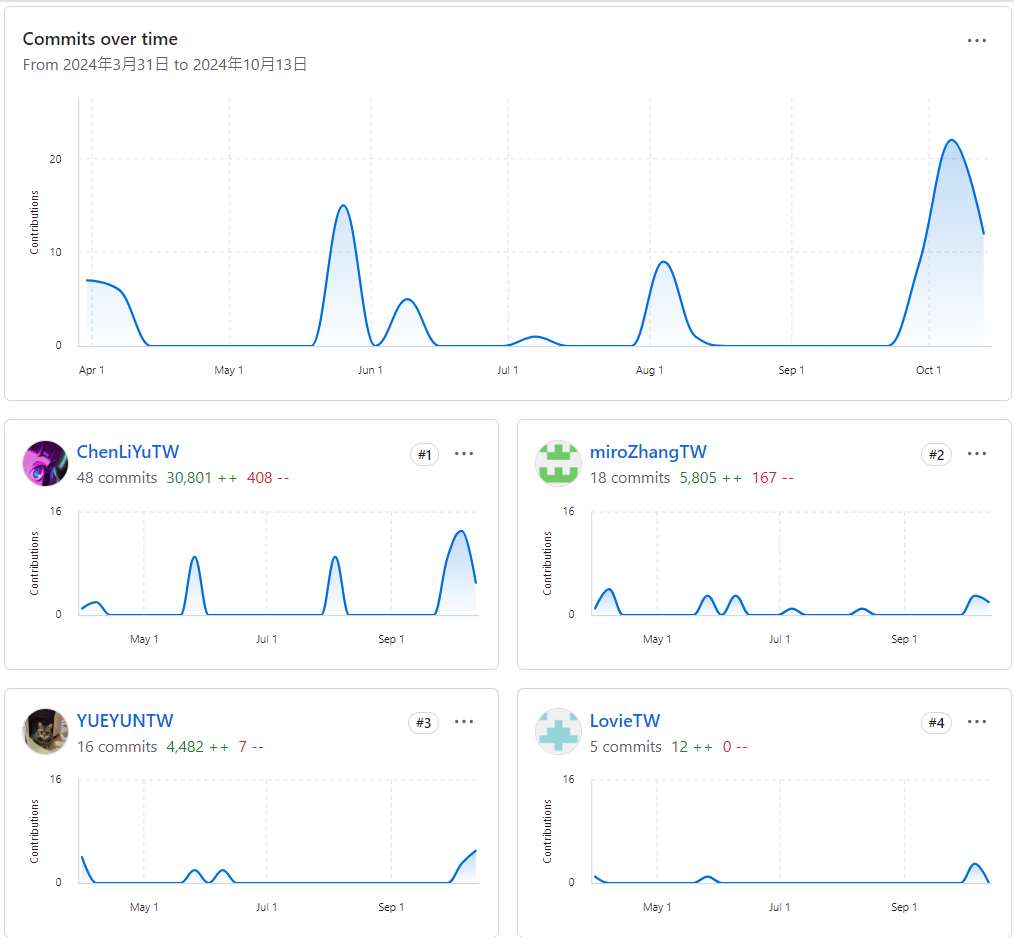


圖 4-1‑2 Github資料上傳記錄

## 專案組織與分工

●主要負責人 〇次要負責人 (每一項只能有1位主要負責人，次要負責人最多2位)

表 4-2‑1專案組織與分工表

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 項目/組員 | | | | 11046078/張明蓉 | 11046068/陳力榆 | 11046074/柯恩璇 | 11046075/郭岳鋆 |
| 後端開發 | 資料庫 | | 建置 | ● |  | 〇 | 〇 |
| 關聯圖 |  |  |  | ● |
| 前端資料建表 |  | ● |  |  |
| 匹配度建表 | ● |  |  |  |
| 伺服器架設 | | | 〇 |  | ● |  |
| 機器學習 | | | 〇 |  |  | ● |
| 爬蟲 | | |  | ● |  | 〇 |
| Flask-爬蟲匯到excel | | | ● |  |  |  |
| Flask-後端程式 | | |  | ● |  |  |
| 前端開發 | 首頁 | | |  | ● |  |  |
| 肌膚檢測 | | | ● |  | 〇 |  |
| 篩選-關鍵字 | | | 〇 |  |  | ● |
| 篩選-價格 | | |  | 〇 | ● |  |
| 產品 | | |  | ● |  | 〇 |
| 前端網頁 | | |  | ● |  |  |
| 整合 | 前後端連線 | | | 〇 |  |  | ● |
| Nginx後端匯入/  前端抓取資料 | | |  | ● |  |  |
| 美術設計 | UI/ UX | | |  | 〇 |  | ● |
| Web介面設計 | | | ● | 〇 |  |  |
| 色彩設計 | | |  |  | 〇 | ● |
| Logo設計 | | |  | ● |  |  |
| 素材設計 | | |  |  | ● | 〇 |
| 文件撰寫 | 統整 | | | 〇 |  | ● |  |
| 修改 | 內容 | | 〇 | 〇 | ● |  |
| 圖表 | |  |  | 〇 | ● |
| 第1章 前言 | | |  |  | ● | 〇 |
| 第2章 營運計畫 | | | 〇 | ● |  |  |
| 第3章 系統規格 | | | ● | 〇 |  |  |
| 第4章 專題時程與組織分工 | | |  | ● | 〇 | 〇 |
| 第5章 需求模型 | | |  |  | 〇 | ● |
| 第6章 設計模型 | | | ● | 〇 |  | 〇 |
| 第7章 實作模型 | | | 〇 | ● |  |  |
| 第8章 資料庫設計 | | | ● |  | 〇 | 〇 |
| 第9章 程式 | | | ● | 〇 |  |  |
| 第10章 測試模型 | | | ● |  | 〇 |  |
| 第11章 操作手冊 | | |  | 〇 |  | ● |
| 第12章 使用手冊 | | |  | 〇 | ● |  |
| 第13章 感想 | | |  |  | ● |  |
| 第14章 參考資料 | | |  |  |  | ● |
| 附錄 | | |  |  | ● |  |
| 報告 | 簡報製作 | | |  |  | ● |  |

表 4-2‑2專題成果工作內容與貢獻度表

|  |  |  |  |
| --- | --- | --- | --- |
| 序號 | 姓名 | 工作內容<各限100字以內> | 貢獻度 |
| 1 | 組長  張明蓉 | 機器學習、資料庫建置、匹配度建表、Flask-爬蟲匯到excel、肌膚檢測表單、Web介面設計，系統規格、設計模型、資料庫設計、程式及測試模型文件撰寫。 | 27.4% |
| 2 | 組員  陳力榆 | 資料庫前端資料建表、爬蟲、Flask-後端程式、首頁設計、產品頁面設計、前端網頁撰寫、Nginx後端匯入/前端抓取資料、Logo設計，營運計畫、專題時程與組織分工、實作模型文件撰寫。 | 28.5 % |
| 3 | 組員  柯恩璇 | 伺服器架設、篩選-價格功能、素材設計，文件統整、內容修改、前言文件、使用手冊、附錄及感想撰寫、製作簡報（初評及複評）。 | 21.5 % |
| 4 | 組員  郭岳鋆 | 資料庫維護、關聯圖、前後端連線、篩選關鍵字功能、色彩設計、圖表修改、操作手冊、參考資  料、需求模型文件撰寫、收集產品資料。 | 22.6% |
|  | | | 總計:100% |

# 需求模型

## 使用者需求

本系統分為六個部分，分別為：會員專區、新增評論、肌膚檢測功能、搜尋功能、篩選功能、查看詳細資料功能。其中，使用者可先電子郵件註冊會員，接者利用肌膚檢測功能，透過篩選功能輸入欲查詢之品牌、產品，亦可選擇產品本身理想價格，可查看產品詳細資料得到產品之有效簡化評論及相關產品資訊，肌膚檢測之結果、收藏喜愛產品、個人資料以及新增過的評論會記錄在會員專區，方便後續回查。

表 5-1‑1使用者需求表

|  |  |
| --- | --- |
| 功能性需求 | |
| 使用者 | |
| 功能項目 | 說明 |
| 會員專區 | 使用者可用電子郵件註冊會員，使用肌膚檢測功能後，記錄於會員專區，查看收藏記錄、個人資料 |
| 新增評論功能 | 使用者可在會員專區新增產品評論，再傳入後端進行簡化 |
| 肌膚檢測功能 | 透過測驗快速了解顧客的膚質狀態，並加以篩選適合的產品 |
| 皮膚偵測功能 | 利用face++技術，透過消費者上傳的圖片，進行皮膚狀況的偵測，用於推薦產品 |
| 搜尋功能 | 使用者可將欲查詢之品牌、產品等關鍵字輸入至搜尋欄位，查看商品 |
| 篩選功能 | 使用者可依自身需求篩選品牌、產品、價格欄位 |
| 查看詳細資料功能 | 使用者可查看所選產品之關產品資訊以及其他使用者簡化之評論 |

## 使用個案圖

本組將案例圖分為使用者及管理者兩類，如下：

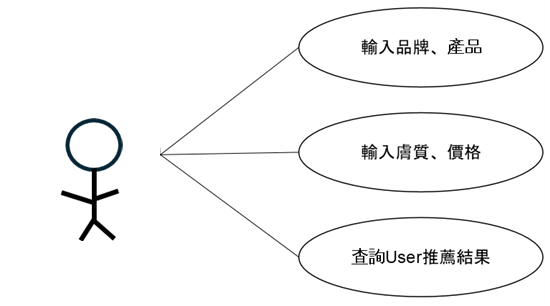


圖 5-2‑1使用個案圖-User

由上圖可得知，使用者除了可以輸入產品或品牌外，也可輸入本身膚質狀況或者理想價格，透過本系統推薦合適產品給使用者。

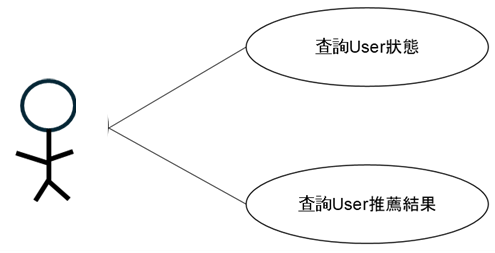


圖 5-2‑2使用個案圖-管理者

由上圖可得知，管理者主要可以查詢使用者的狀態以及使用者之推薦結果。

## 使用個案描述

本組將個案描述分為使用者端及管理者端兩類。

1. 使用者端

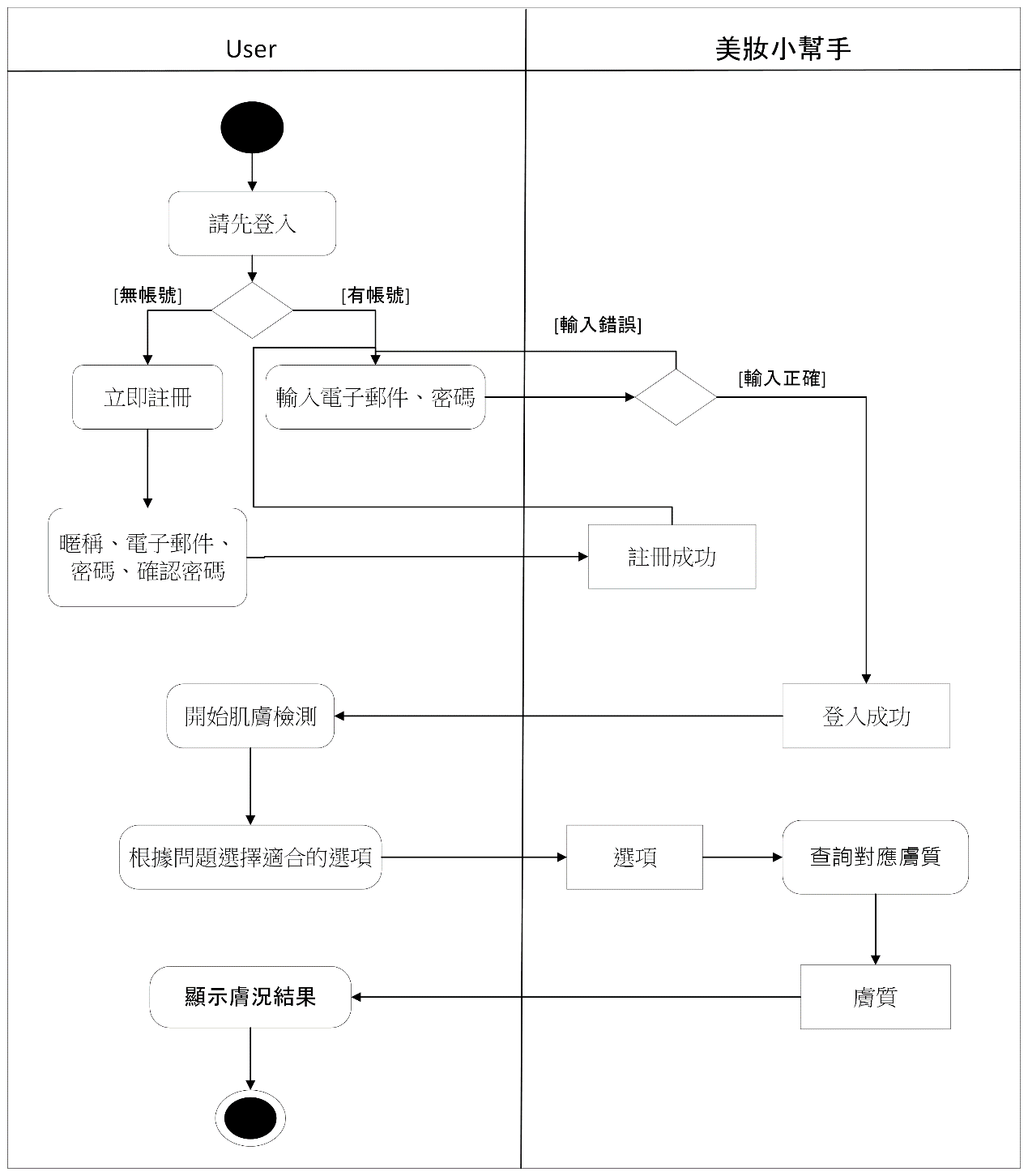


圖 5-3‑1肌膚檢測之活動圖

由圖5-3-1得知，User進入網頁先登入會員再進行肌膚檢測，根據問題選擇適合的選項，則會顯示您的膚質，其檢測記錄將會記錄在會員專區。

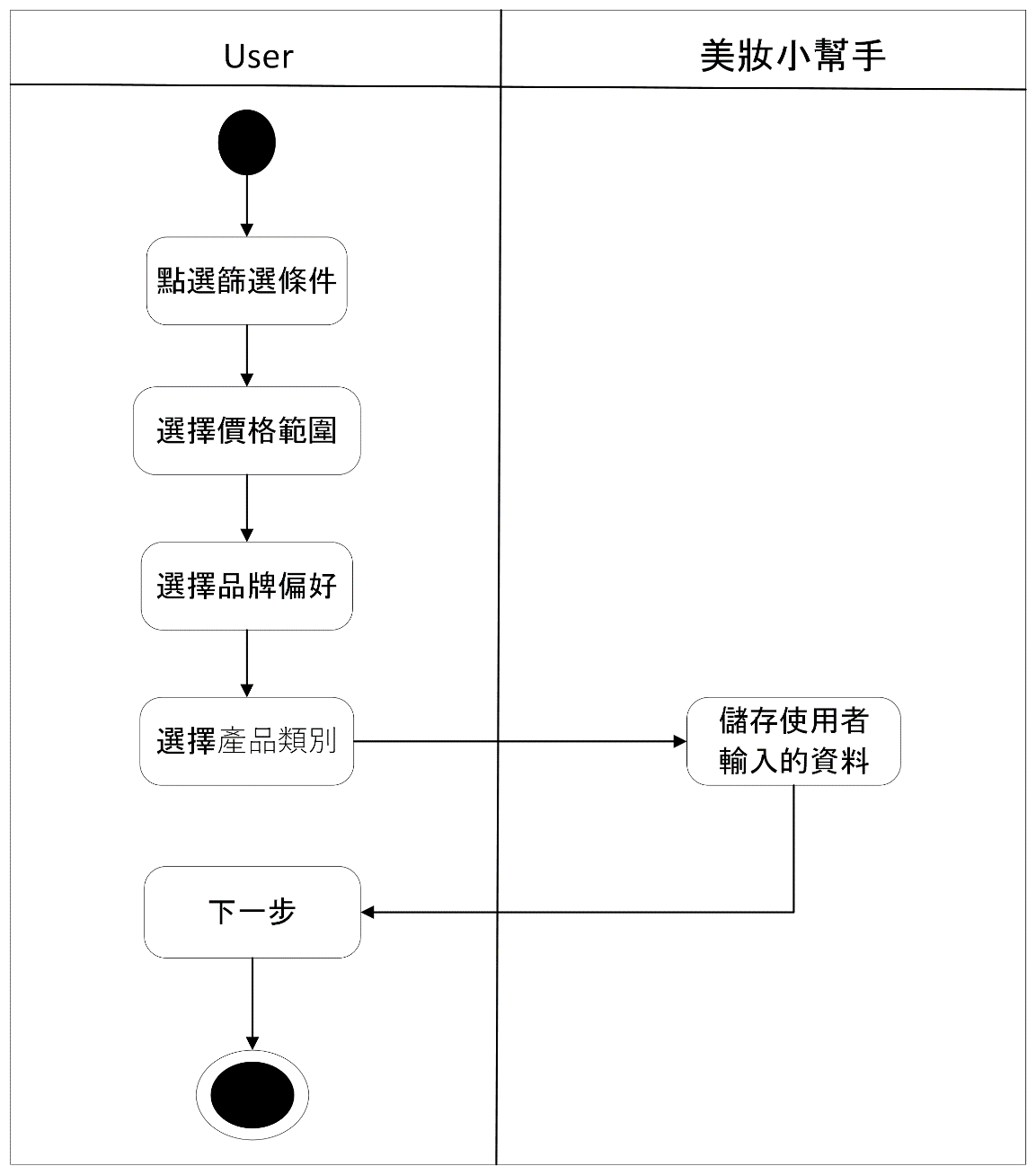


圖 5-3‑2篩選條件之活動圖

由圖5-3-2得知，User點選篩選功能，選擇您欲查詢的價格範圍，以及想尋找的品牌及產品類別。

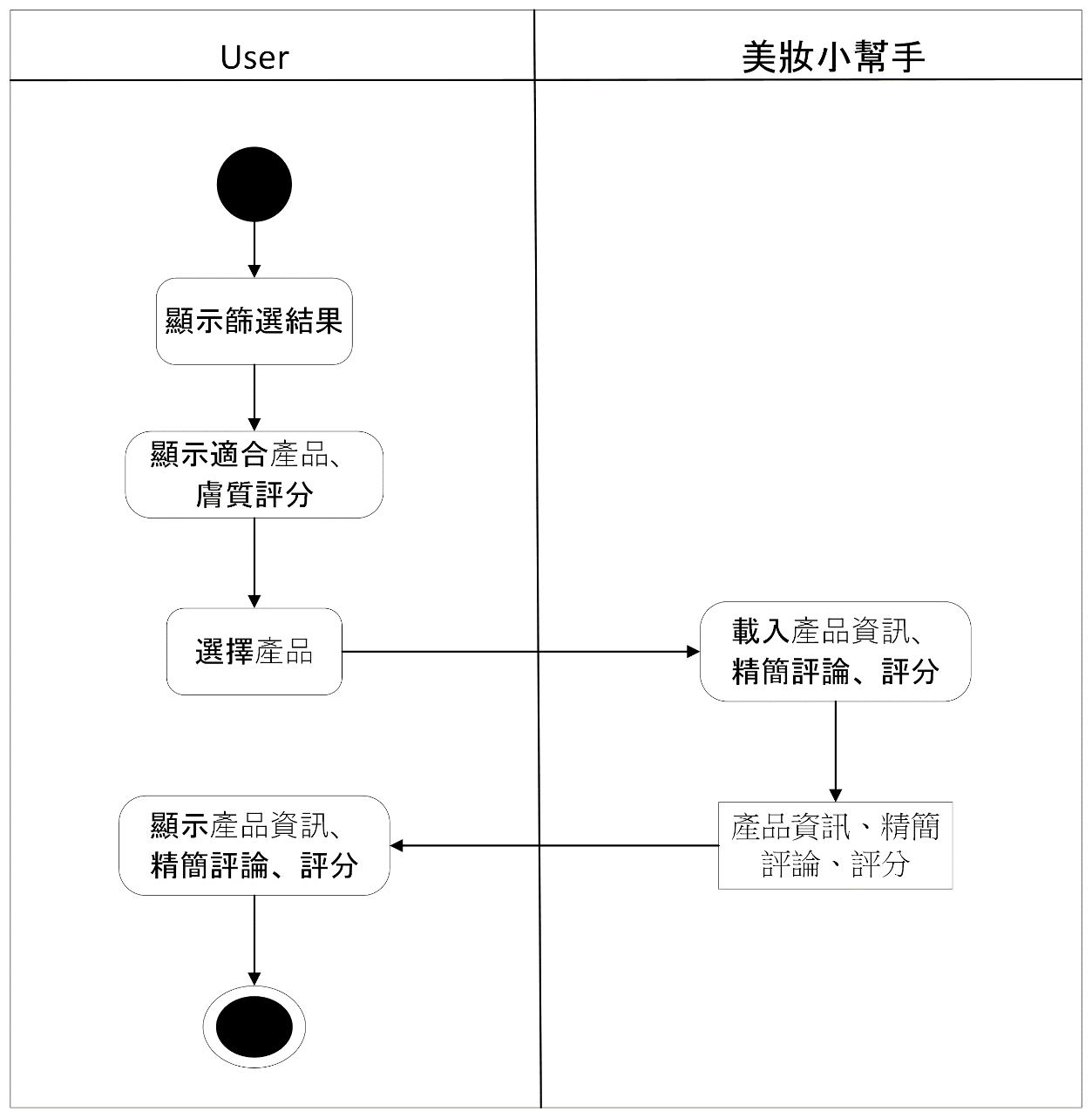


圖 5-3‑3篩選結果之活動圖

由圖5-3-3得知，載入篩選出的產品資訊，點選欲匹配的產品，則顯示產品精簡評論、AI評分及肌膚評分。

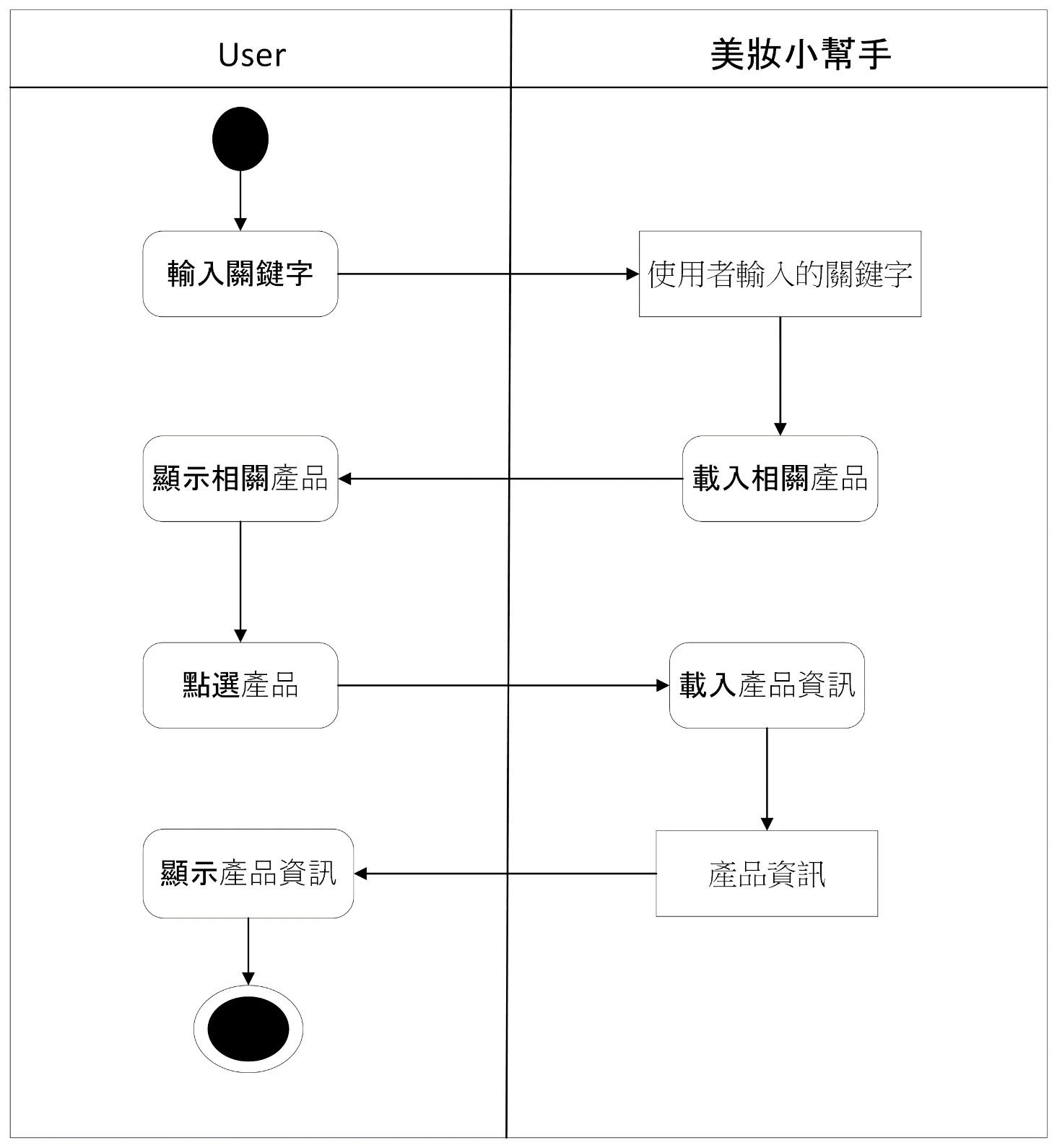


圖 5-3‑4搜尋之活動圖

由圖5-3-4得知，User可以透過輸入關鍵字查詢相關產品，也可進一步查看產品資訊。

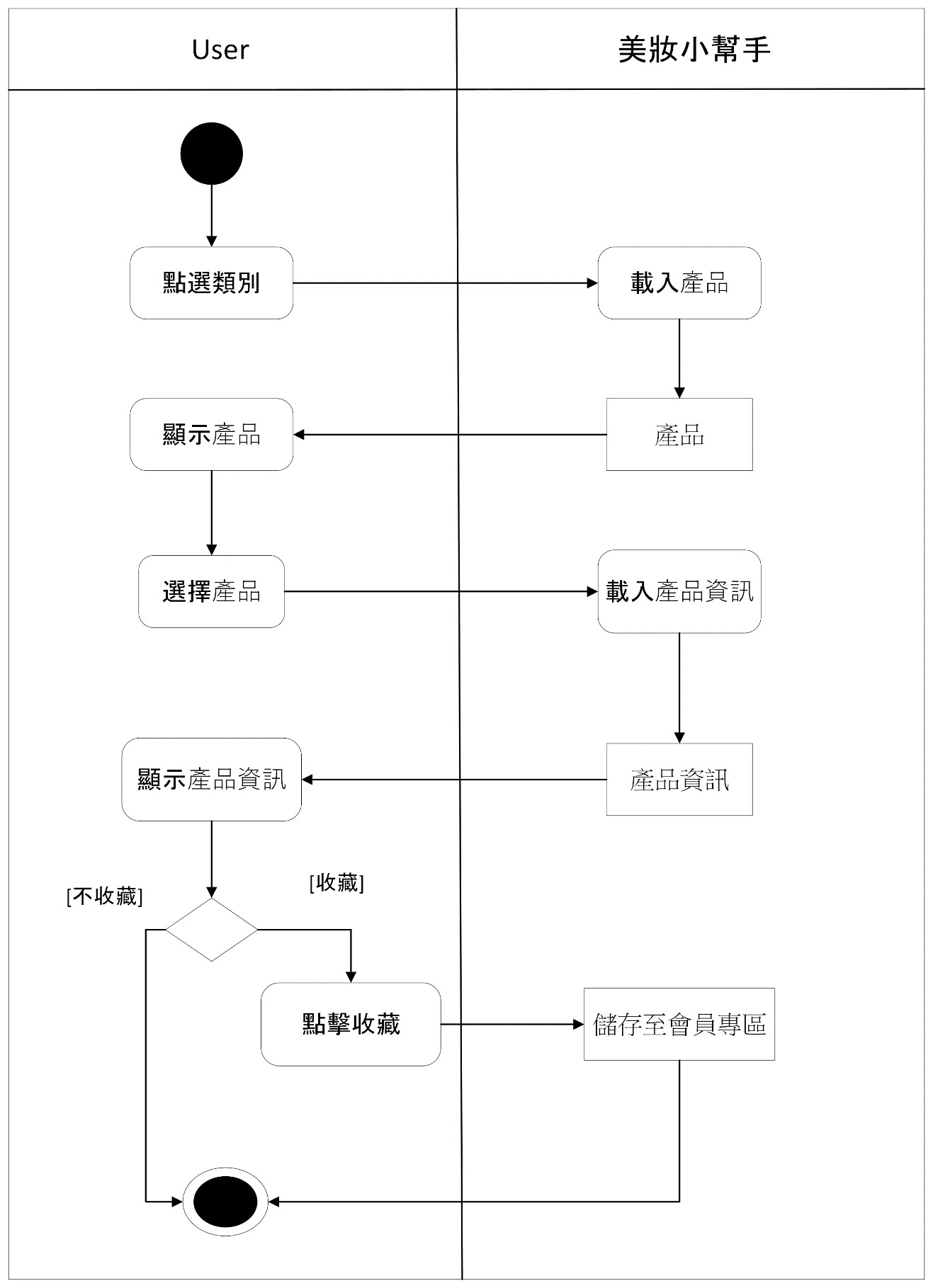


圖 5-3‑5產品資訊之活動圖

由圖5-3-5得知，User可以透過首頁類別來尋找產品，並進一步查看產品資訊，也可選擇收藏此產品。

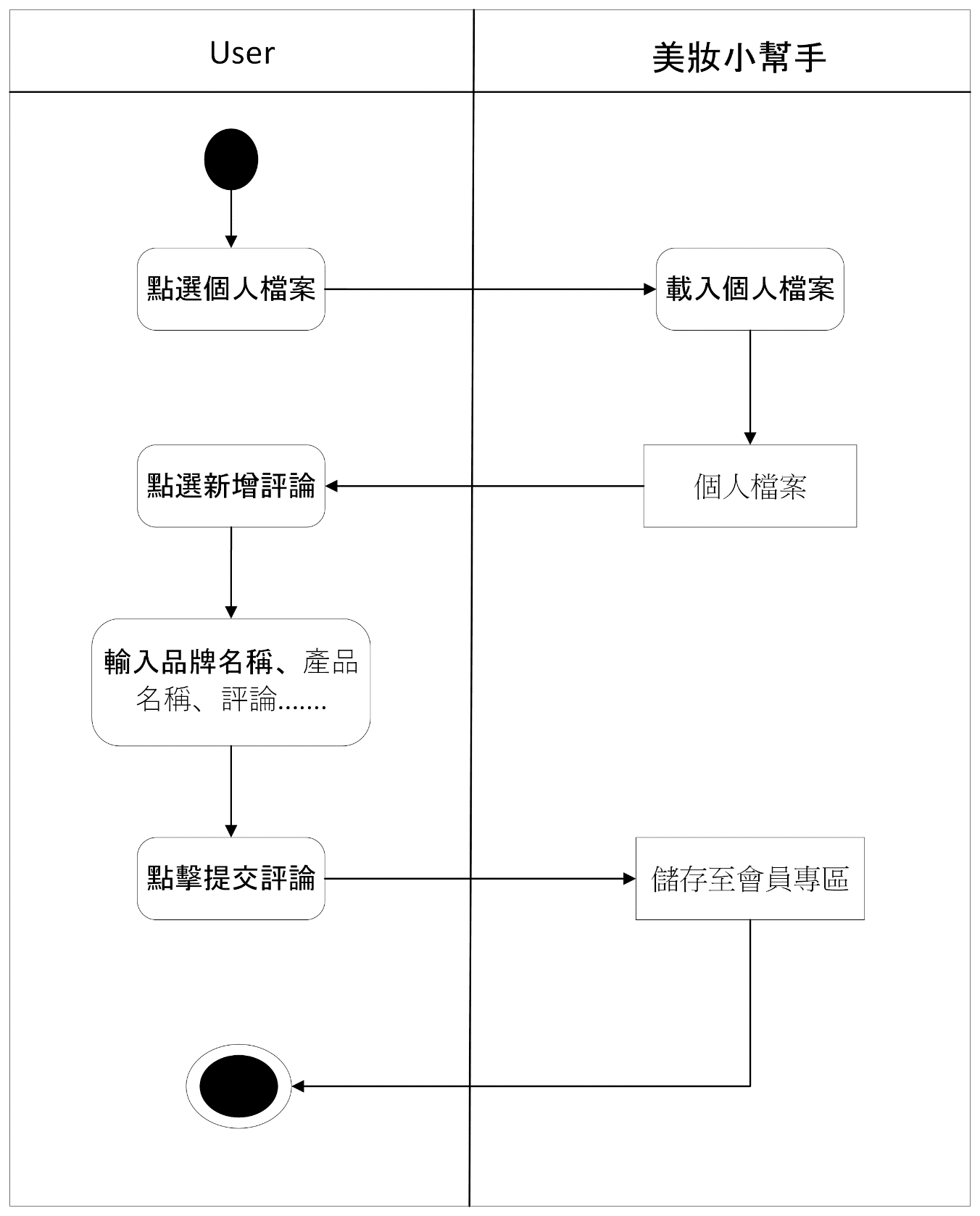


圖 5-3‑6新增評論之活動圖

由圖5-3-6得知，User可以透過會員專區的新增評論區來新增評論，輸入想要評論的品牌名稱、產品名稱等……，點擊提交評論，您新增的評論則會在會員專區裡我的評論保有記錄。

2.管理者端

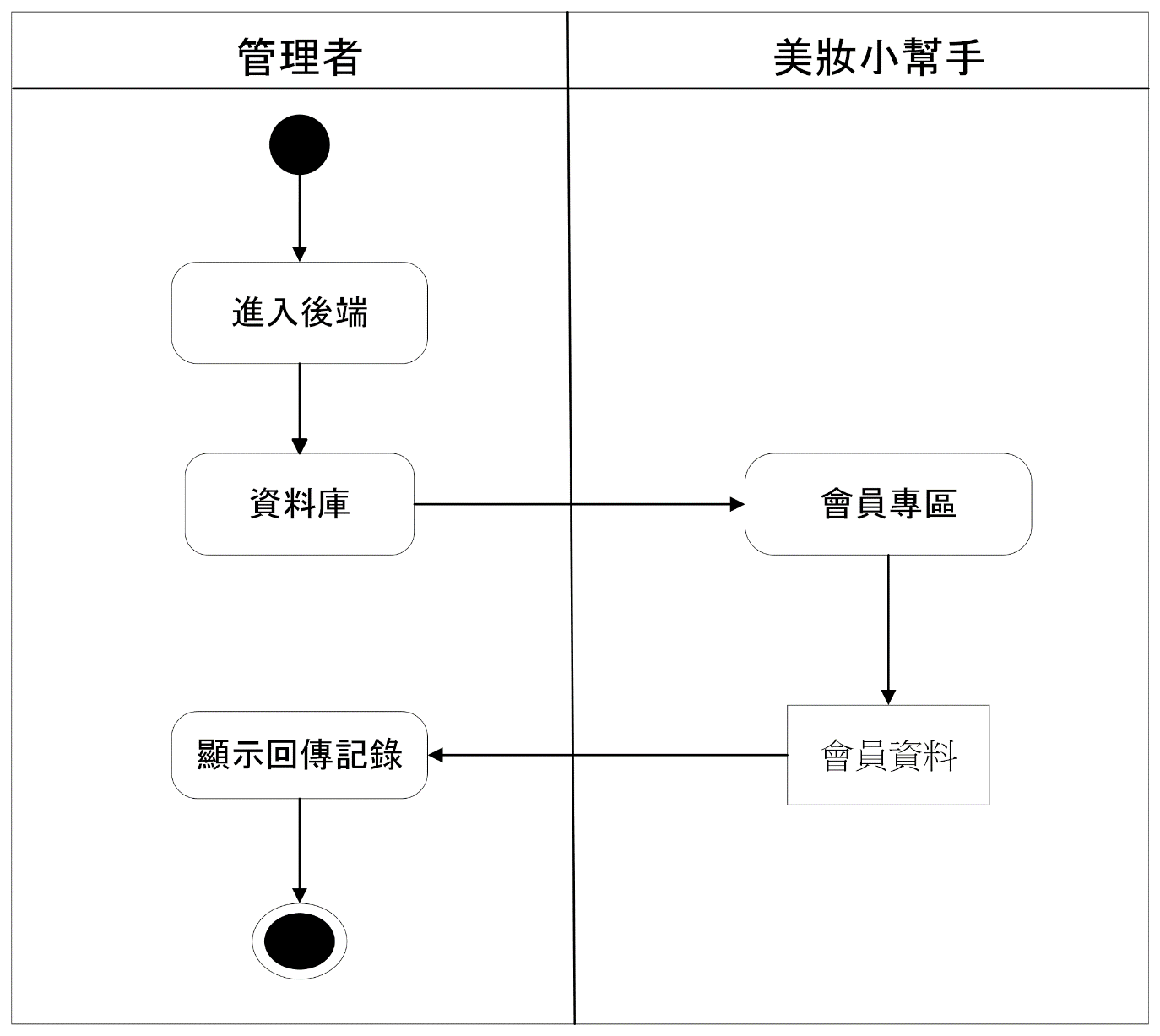


圖 5-3‑7管理者之活動圖

由圖5-3-7得知，管理者可以透過登入平台後端，即可查詢User回傳記錄。

## 分析類別圖（Analysis class diagram）

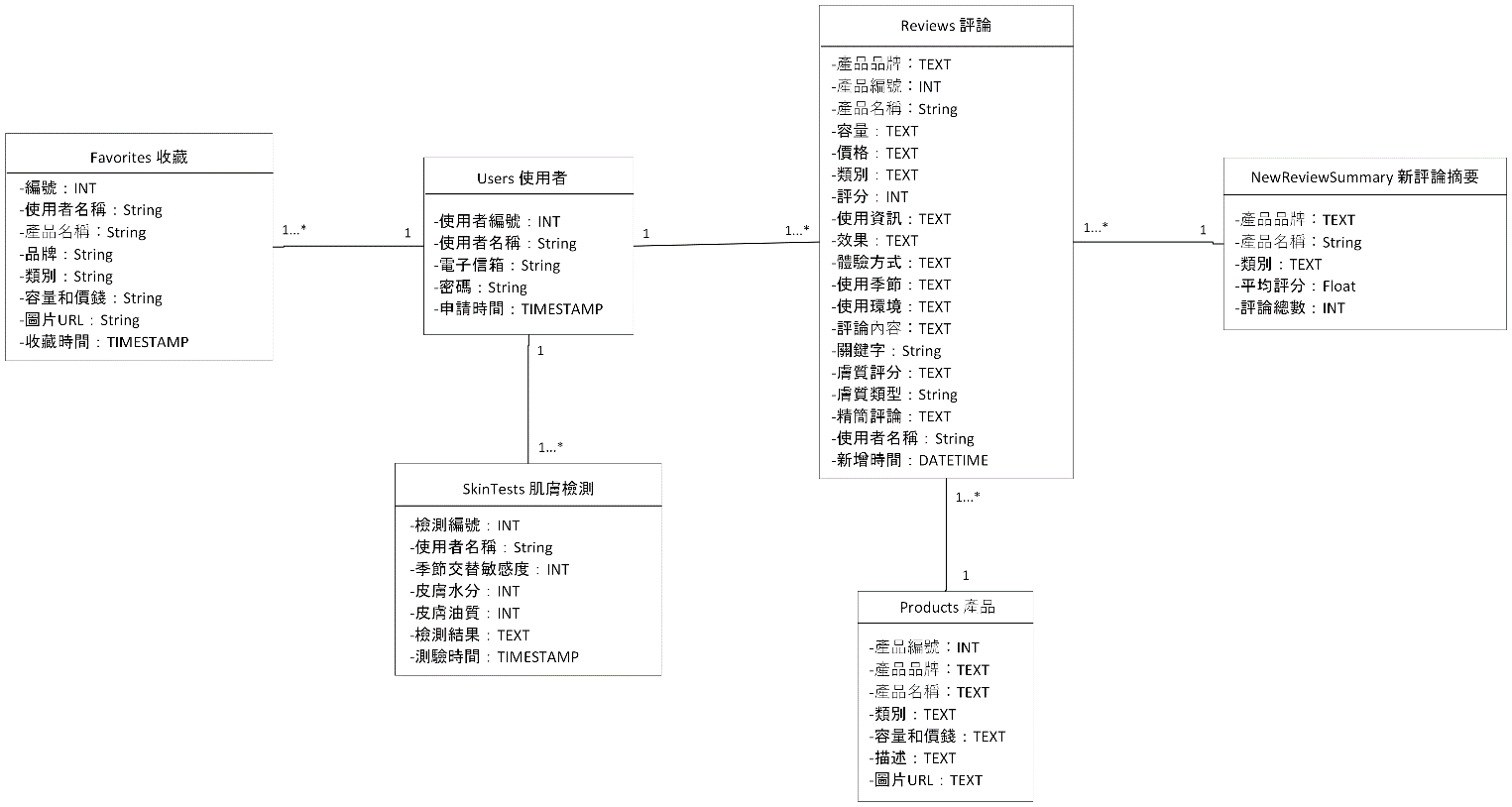


圖 5-4‑1分析類別圖

# 設計模型

## 循序圖

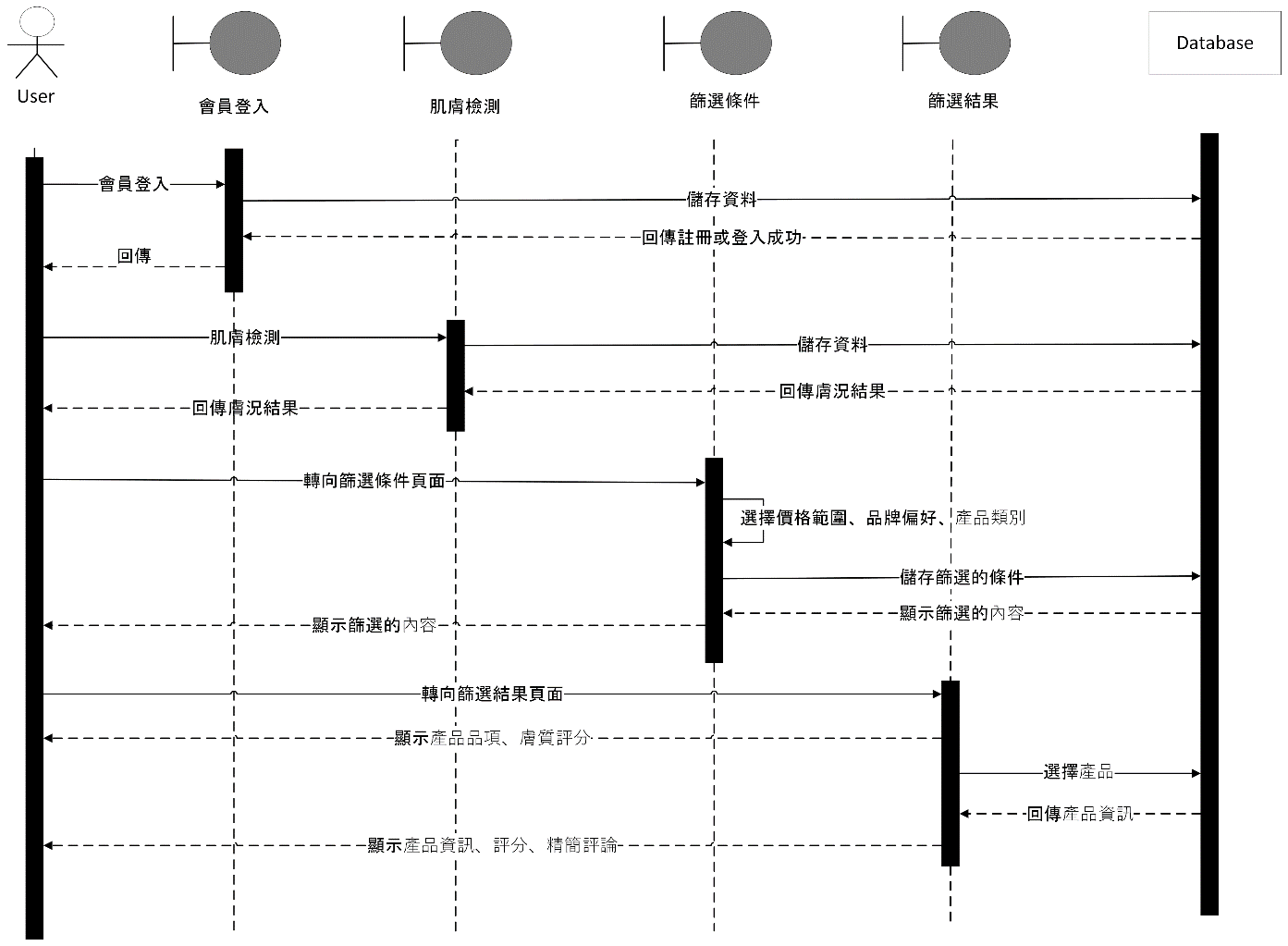


圖 6-1‑1循序圖

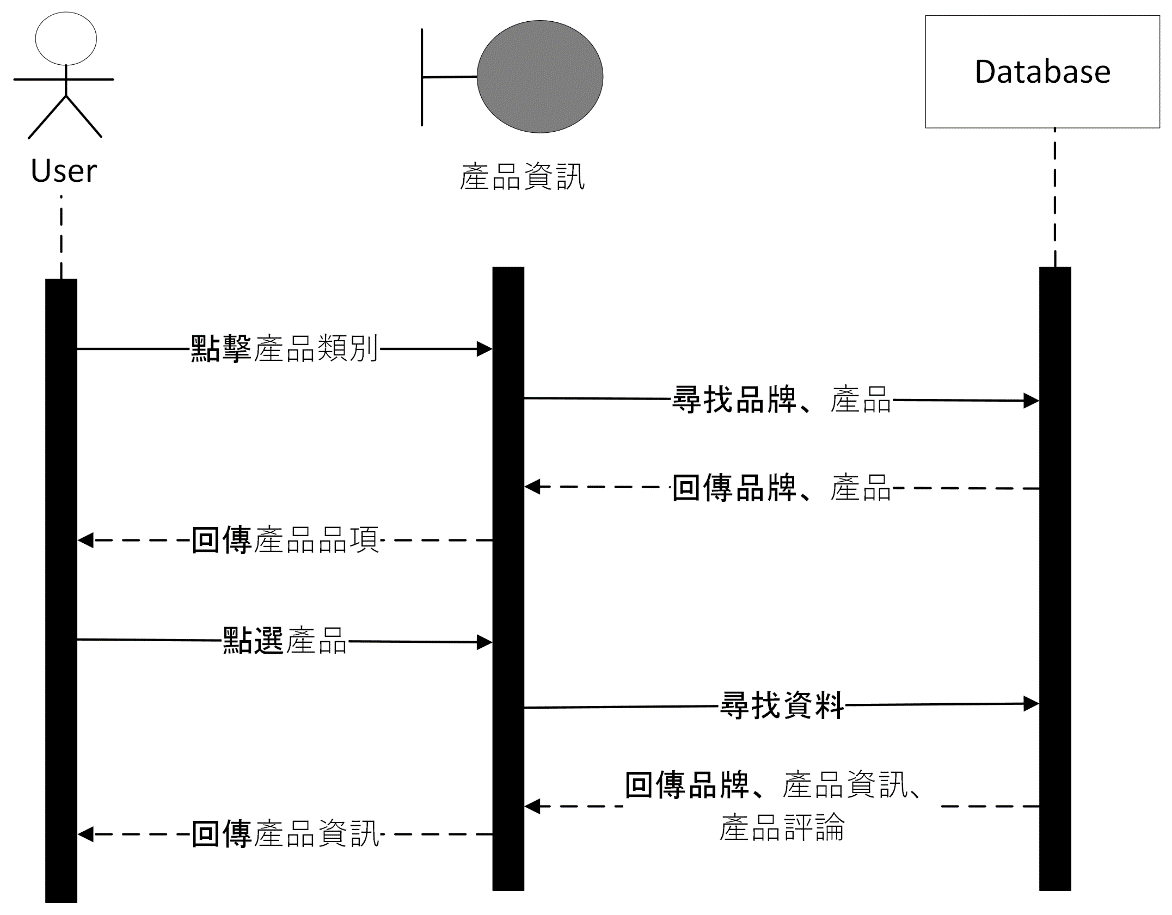


圖 6-1‑2首頁產品資訊循序圖

## 設計類別圖

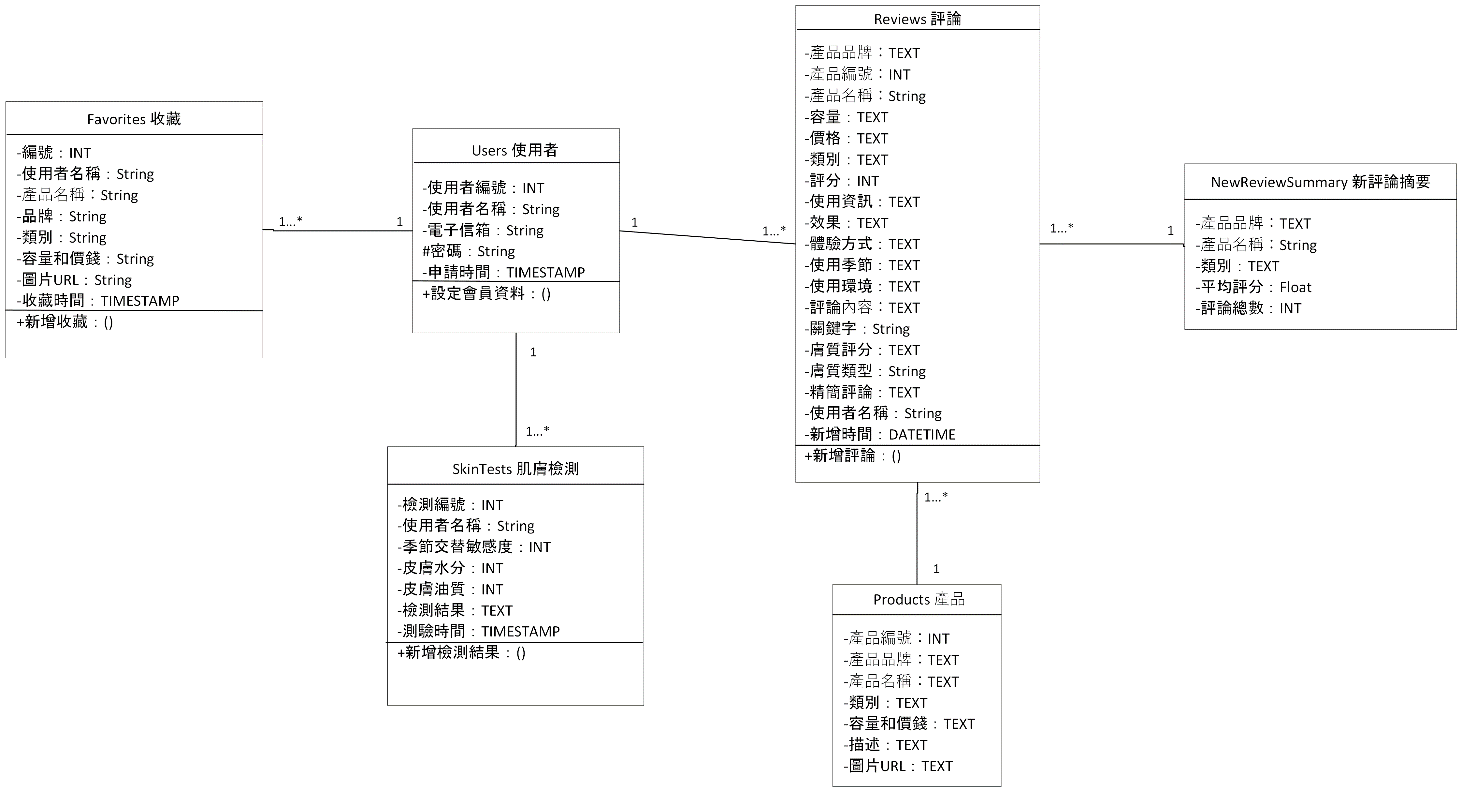


圖 6-2‑1設計類別圖

# 實作模型

## 佈署圖(Deployment diagram)

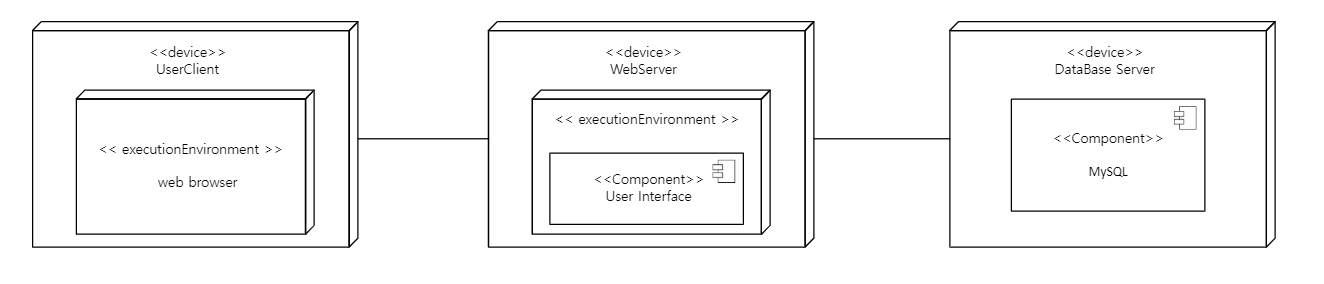


圖 7-1‑1佈署圖

## 套件圖(Package diagram)

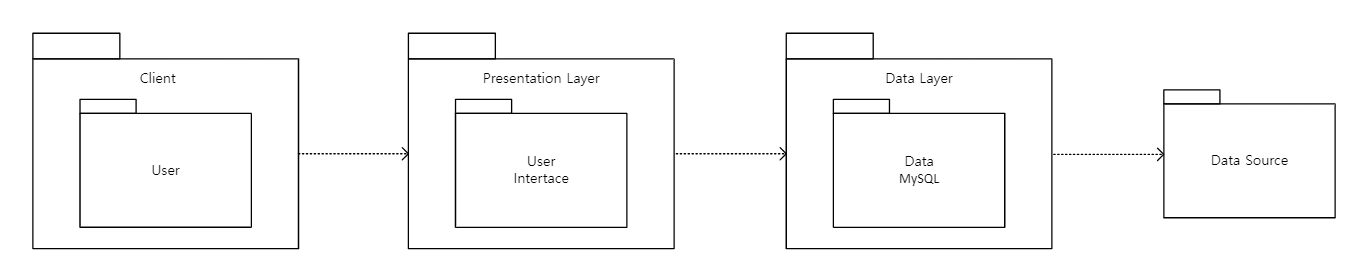


圖 7-2‑1套件圖

## 元件圖(Component diagram)

一張含有 黑暗, 黑色, 螢幕擷取畫面, space 的圖片

自動產生的描述

圖 7-3‑1元件圖

## 狀態機(State machine)

一張含有 文字, 圓形, 螢幕擷取畫面, 圖表 的圖片

自動產生的描述

圖 7-4‑1肌膚檢測狀態圖

一張含有 文字, 螢幕擷取畫面, 圓形, 圖表 的圖片

自動產生的描述

圖 7-4‑2搜尋商品狀態圖

一張含有 文字, 螢幕擷取畫面, 圓形, 寫生 的圖片

自動產生的描述

圖 7-4‑3查詢價格狀態圖

一張含有 文字, 螢幕擷取畫面, 圓形, 圖表 的圖片

自動產生的描述

圖 7-4‑4查詢歷史記錄狀態圖

一張含有 文字, 螢幕擷取畫面, 字型, 圓形 的圖片

自動產生的描述

圖 7-4‑5新增評論狀態圖

# 資料庫設計

## 資料庫關聯表

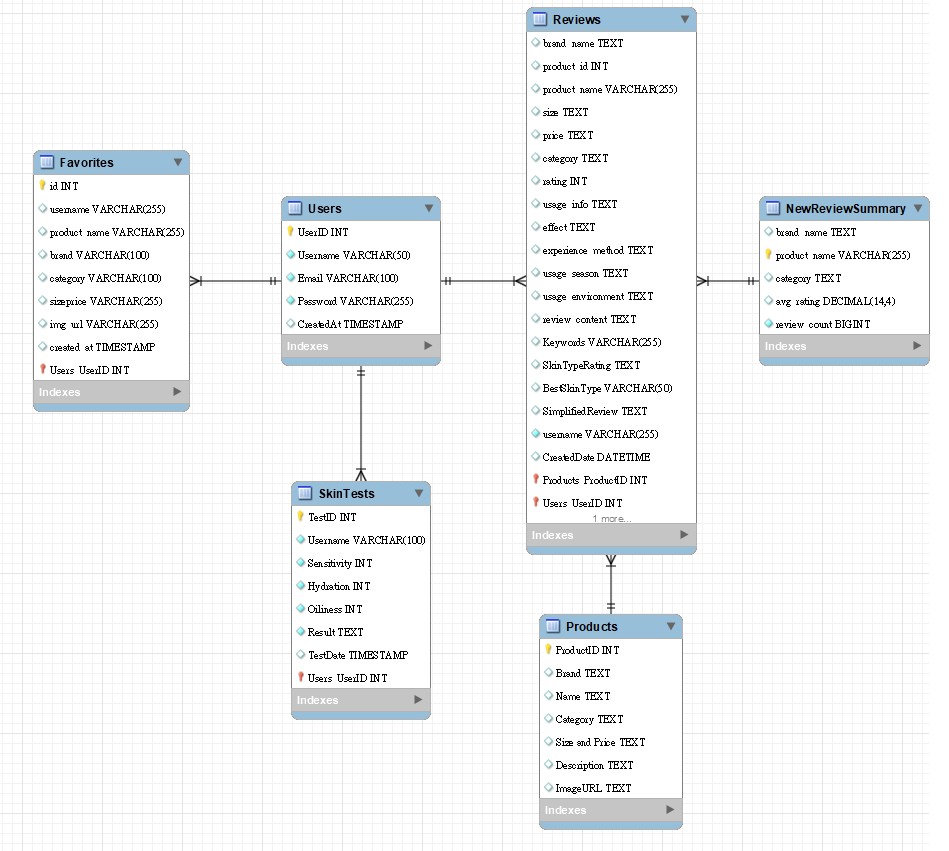


圖 8-1‑1資料庫關聯圖

## 表格及其Meta data

表 8-2‑1 Products–產品表

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 資料表 | | Products | | | | | |
| 資料表描述 | | 產品表 | | | | | |
| 項目 | 欄位名稱 | 欄位說明 | 資料型態 | 主鍵 | 可為空 | 規則 | 預設值 |
| 1 | ProductID | 產品編號 | INT | ✓ |  | AI |  |
| 2 | Brand | 產品品牌 | TEXT |  |  |  | NULL |
| 3 | Name | 產品名稱 | TEXT |  |  |  | NULL |
| 4 | Category | 類別 | TEXT |  |  |  | NULL |
| 5 | Size and Price | 容量和價格 | TEXT |  |  |  | NULL |
| 6 | Description | 描述 | TEXT |  |  |  | NULL |
| 7 | ImageURL | 圖片URL | TEXT |  |  |  | NULL |

表 8-2‑2 Users–使用者表

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 資料表 | | Users | | | | | |
| 資料表描述 | | 使用者表 | | | | | |
| 項目 | 欄位名稱 | 欄位說明 | 資料型態 | 主鍵 | 可為空 | 規則 | 預設值 |
| 1 | UserID | 使用者  編號 | INT | ✓ |  | AI |  |
| 2 | Username | 使用者  名稱 | VARCHAR |  |  | UQ |  |
| 3 | Email | 電子信箱 | VARCHAR |  |  | UQ |  |
| 4 | Password | 密碼 | VARCHAR |  |  |  |  |
| 5 | CreatedAt | 申請時間 | TIMESTAMP |  |  |  | DEFAULT CURRENT\_TIMESTAMP |

表 8-2‑3 Reviews–評論表

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 資料表 | | Reviews | | | | | |
| 資料表描述 | | 評論表 | | | | | |
| 項目 | 欄位名稱 | 欄位說明 | 資料型態 | 主鍵 | 可為空 | 規則 | 預設值 |
| 1 | brand\_name | 品牌名稱 | TEXT |  |  |  | Null |
| 2 | product\_id | 產品編號 | INT |  |  |  | Null |
| 3 | product\_name | 產品名稱 | TEXT |  |  |  | Null |
| 4 | size | 容量 | TEXT |  |  |  | Null |
| 5 | price | 價格 | TEXT |  |  |  | Null |
| 6 | category | 類別 | TEXT |  |  |  | Null |
| 7 | rating | 評分 | INT |  |  |  | Null |
| 8 | usage\_info | 使用資訊 | TEXT |  |  |  | Null |
| 9 | effect | 效果 | TEXT |  |  |  | Null |
| 10 | experience\_method | 體驗方式 | TEXT |  |  |  | Null |
| 11 | usage\_season | 使用季節 | TEXT |  |  |  | Null |
| 12 | usage\_environment | 使用環境 | TEXT |  |  |  | Null |
| 13 | review\_content | 評論內容 | TEXT |  |  |  | Null |
| 14 | Keywords | 關鍵字 | VARCHAR |  |  |  | Null |
| 15 | SkinTypeRating | 膚質評分 | TEXT |  |  |  | Null |
| 16 | BestSkinType | 膚質類型 | VARCHAR |  |  |  | Null |
| 17 | SimplifiedReview | 精簡評論 | TEXT |  |  |  | Null |
| 18 | username | 使用者  名稱 | VARCHAR |  |  |  |  |
| 19 | CreatedDate | 新增時間 | DATETIME |  |  |  | DEFAULT CURRENT\_TIMESTAMP |

表 8-2‑4 NewReviewSummary–新評論摘要表

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 資料表 | | NewReviewSummary | | | | | |
| 資料表描述 | | 新評論摘要表 | | | | | |
| 項目 | 欄位名稱 | 欄位說明 | 資料型態 | 主鍵 | 可為空 | 規則 | 預設值 |
| 1 | brand\_name | 產品品牌 | TEXT |  |  |  | NULL |
| 2 | product\_name | 產品名稱 | VARCHAR | ✓ |  |  |  |
| 3 | category | 類別 | TEXT |  |  |  | NULL |
| 4 | avg\_rating | 平均評分 | Float |  |  |  | NULL |
| 5 | review\_count | 評論總數 | BIGINT |  |  |  | ‘0’ |

表 8-2‑5 Favorites–收藏表

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 資料表 | | Favorites | | | | | |
| 資料表描述 | | 收藏表 | | | | | |
| 項目 | 欄位名稱 | 欄位說明 | 資料型態 | 主鍵 | 可為空 | 規則 | 預設值 |
| 1 | id | 編號 | INT | ✓ |  | AI |  |
| 2 | username | 使用者  名稱 | VARCHAR |  |  |  | NULL |
| 3 | product\_name | 產品名稱 | VARCHAR |  |  |  | NULL |
| 4 | brand | 品牌 | VARCHAR |  |  |  | NULL |
| 5 | category | 類別 | VARCHAR |  |  |  | NULL |
| 6 | sizeprice | 容量和  價錢 | VARCHAR |  |  |  | NULL |
| 7 | img\_url | 圖片URL | VARCHAR |  |  |  | NULL |
| 8 | created\_at | 收藏時間 | TIMESTAMP |  |  |  | DEFAULT CURRENT\_TIMESTAMP |

表 8-2‑6 SkinTests–肌膚檢測結果表

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 資料表 | | SkinTests | | | | | |
| 資料表描述 | | 肌膚檢測結果表 | | | | | |
| 項目 | 欄位名稱 | 欄位說明 | 資料型態 | 主鍵 | 可為空 | 規則 | 預設值 |
| 1 | TestID | 檢測編號 | INT | ✓ |  | AI |  |
| 2 | Username | 使用者  名稱 | VARCHAR |  |  |  |  |
| 3 | Sensitivity | 季節交替敏感度 | INT |  |  |  |  |
| 4 | Hydration | 皮膚水分 | INT |  |  |  |  |
| 5 | Oiliness | 皮膚油質 | INT |  |  |  |  |
| 6 | Result | 檢測結果 | TEXT |  |  |  |  |
| 7 | TestDate | 檢測日期 | TIMESTAMP |  |  |  | DEFAULT CURRENT\_TIMESTAMP |

# 程式

## 表格說明

表 9-1‑1元件清單及其規格描述

|  |  |  |
| --- | --- | --- |
| 元件名稱 | 功能概述 | 規格描述 |
| 前端(HTML/CSS/JavaScript) | 提供用戶互動界面，讓消費者輸入皮膚類型、喜好等訊息 | HTML5,CSS3,JavaScript(ES6+),Bootstrap |
| 後端(Node.js + Express) | 接收前端數據，利用自然語言和演算法對數據進行處理並呈現 | Node.js v16+, Express v4.17, RESTful API |
| 資料庫 (MySQL) | 儲存用戶資訊、產品數據、評論和評分等資料 | MySQL 8.0，關聯式數據庫，支援SQL查詢語句 |
| 機器學習模型 (Python + Scikit-learn) | 使用自然語言處理技術進行數據分析，提供精確的推薦結果 | Python + Regular Expressions, TextBlob, Pandas, Transformers, Datasets |
| 爬蟲工具 (Python + Scrapy) | 自動收集產品評論和評分數據，用於自然語言處理模型的訓練 | Scrapy,BeautifulSoup,Requests, 預處理數據 |

表 9-1‑2其他附屬之各種元件

|  |  |  |
| --- | --- | --- |
| 元件名稱 | 功能 | 描述 |
| 產品比對功能模組 | 支援用戶搜索和比對多個產品 | 根據用戶輸入的關鍵詞或類別搜索產品，並提供比對產生結果 |
| 評論和評分模組 | 提供產品的真實用戶評價，輔助決策 | 從多個平台收集評論資料和評分，並顯示於產品詳情 |
| 即時更新模組 | 根據市場和用戶反饋及時調整推薦策略 | 從資料庫和爬蟲獲取即時數據，並動態更新推薦算法 |

## 程式碼

表 9-2‑1 html(首頁)

|  |  |
| --- | --- |
| html(首頁) | home.html |
| <!DOCTYPE html>  <html lang="zh-Hant">  <head>      <script src="https://cdnjs.cloudflare.com/ajax/libs/headroom/0.12.0/headroom.min.js"></script>      <script src="https://cdnjs.cloudflare.com/ajax/libs/headroom/0.12.0/jQuery.headroom.min.js"></script>      <meta charset="UTF-8">      <meta name="viewport" content="width=device-width, initial-scale=1.0">      <title>Beauty Buddy</title>      <link rel="stylesheet" type="text/css" href="{{ url\_for('static', filename='css/home.css') }}">      <script src="{{ url\_for('static', filename='js/home.js') }}"></script>  </head>  <body>      <div class="header">          <div style="display: flex; align-items: center;">              <img src="https://imgur.com/Akgc4pd.jpeg" alt="Beauty Buddy Logo" id="logo">              <h1>Beauty Buddy</h1>          </div>          <div class="nav">              <ul>                  <li><a href="http://beautybuddy.ddns.net/">首頁</a></li>                  <li><a href="http://beautybuddy.ddns.net/filter">篩選產品</a></li>                  {% if username %}                      <div class="user-menu">                          <a href="#" class="user-link">哈囉 {{ username }} &#128151</a>                          <div class="dropdown-content">                              <a href="/profile">個人檔案</a>                              <a href="/logout">登出</a>                          </div>                      </div>                  {% else %}                      <li><a href="http://beautybuddy.ddns.net/login">登入/註冊</a></li> | |

表 9-2‑2 html(會員專區)

|  |  |
| --- | --- |
| html(會員專區) | Profile.html |
| <!DOCTYPE html>  <html lang="zh-Hant">  <head>      <meta charset="UTF-8">      <meta name="viewport" content="width=device-width, initial-scale=1.0">      <title>個人檔案</title>      <link rel="stylesheet" type="text/css" href="{{ url\_for('static', filename='css/profile.css') }}">  </head>  <body>      <header class="header">          <div style="display: flex; align-items: center;">              <img src="https://imgur.com/Akgc4pd.jpeg" alt="Beauty Buddy Logo" id="logo">              <h1>Beauty Buddy</h1>          </div>          <nav class="nav">              <ul>                  <li><a href="http://beautybuddy.ddns.net/">首頁</a></li>                  <li><a href="http://beautybuddy.ddns.net/filter">篩選條件</a></li>                  {% if username %}                      <div class="user-menu">                          <a href="#" class="user-link">哈囉 {{ username }}&#128512</a>                          <div class="dropdown-content">                              <a href="/profile">個人檔案</a>                              <a href="/logout">登出</a>                          </div>                      </div>                  {% else %}                      <li><a href="http://beautybuddy.ddns.net/login">登入/註冊</a></li>                  {% endif %}              </ul>              <form action="/products" method="GET" style="display: inline;">                  <input type="text" name="keyword" id="searchInput" placeholder="搜索..." required>                  <button type="submit">搜索</button>              </form>          </nav>      </header> | |

表 9-2‑3 Flask(連接前後端)

|  |  |
| --- | --- |
| Flask(連接前後端) | backend.py |
| from flask import Flask, request, render\_template, redirect, url\_for, session, flash  from flask\_cors import CORS  import mysql.connector  from werkzeug.security import generate\_password\_hash, check\_password\_hash  from functools import wraps  from flask import jsonify  from utils import simplify\_review  app = Flask(\_\_name\_\_)  app.secret\_key = 'your\_secret\_key'  class Database:      def \_\_init\_\_(self, host, port, dbname, username, pwd):          self.host = host          self.port = port          self.database = dbname          self.user = username          self.password = pwd          self.\_\_server = None      def setServer(self):          if self.\_\_server is None or not self.\_\_server.is\_connected():              self.\_\_server = mysql.connector.connect(                  host=self.host,                  port=self.port,                  database=self.database,                  user=self.user,                  password=self.password              )      def getSqlData(self, sql, data=None):          self.setServer()          cursor = self.\_\_server.cursor()          try:              cursor.execute(sql, data)              return cursor.fetchall()          finally:              cursor.close()              self.close()      def executeSql(self, sql, data=None):          self.setServer() | |

表 9-2‑4 AI簡化資料

|  |  |
| --- | --- |
| AI簡化資料 | Utils.py |
| import mysql.connector  import re  from textblob import TextBlob  from transformers import pipeline, Trainer, TrainingArguments  from datasets import load\_dataset  import pandas as pd  class Database:      def \_\_init\_\_(self, host, port, dbname, username, pwd):          self.host = host          self.port = port          self.database = dbname          self.user = username          self.password = pwd          self.\_\_server = None      @property      def Server(self):          return self.\_\_server        @Server.setter      def Server(self, value):          self.\_\_server = value      def setServer(self):          if self.Server is None or not self.Server.is\_connected():              self.Server = mysql.connector.connect(                  host=self.host,                  port=self.port,                  database=self.database,                  user=self.user,                  password=self.password              )      def test(self):          self.setServer()          return self.Server.is\_connected()      def runSql(self, sql, data=None): | |

# 測試模型

## 測試說明

表 10-1‑1說明採用之測試方法及其進行方式

|  |  |  |
| --- | --- | --- |
| 測試項目 | 測試方法 | 進行方式 |
| 功能測試 | 核心能力測試、集成測試、系統測試 | 針對每個功能模組進行測試，檢測是否能互相連動 |
| 介面測試 | 使用者測試、UI測試 | 讓多名用戶使用系統，並收集反饋，觀察可用性和友好度 |

集成測試：測試不同系統功能之間的交互，確保它們能夠無縫合作。

例如，前端輸入是否正確傳遞給後端，後端推薦結果是否正確返回到前端。

表 10-1‑2測試個案與測試結果資料

|  |  |  |  |
| --- | --- | --- | --- |
| 測試個案編號 | 測試項目 | 測試目標 | 測試結果 |
| TC001 | 功能測試-產品推薦 | 確認系統能根據用戶輸入進行準確推薦 | 成功，系統準確推薦出3款適合的產品 |
| TC002 | 介面測試-搜索功能 | 確認產品搜索結果是否準確 | 成功，搜索關鍵詞「保濕」顯示相關產品 |
| TC003 | 兼容性測試 | 測試系統在不同設備、瀏覽器和操作系統上能正常運行。 | 成功，web功能測試正常 |

# 操作手冊

## 表格

表 11-1‑1 操作手冊表

|  |
| --- |
| 網頁 |
| 1.開啟任一瀏覽器 |
| 2.輸入網址：http://beautybuddy.ddns.net/ |
| 3.會員登入或註冊 |
| 4.即可進行檢測並記錄結果 |

表 11-1‑2系統元件介紹表

|  |
| --- |
| 1.元件簡介  伺服器端：使用 Nginx 作為 Web 伺服器，並運行在 Windows 作業系統上。  資料庫：使用 MySQL 資料庫來儲存網站的數據。  應用層：網站前端使用 HTML5、CSS3 和 JavaScript 實作，後端則採用 Node.js 進行伺服器端邏輯處理。  **No-IP：**作為動態 DNS（DDNS）服務，用於將動態 IP 與固定域名綁定，實現穩定的遠端訪問 |
| 2.簡易架構圖  使用者🡪瀏覽器🡪 No-IP (動態域名解析) 🡪Web 伺服器 (nginx)🡪應用程式 (Node.js) 🡪資料庫 (MySQL) |

表 11-1‑3安裝流程表

|  |  |
| --- | --- |
| 前置條件 | 作業系統：Windows 10 或更高版本  nginx：作為 Web 伺服器  Node.js：版本 14.x 或更高  MySQL：8.0 版本 |
| 安裝步驟 | 1.安裝nginx  前往 nginx 官方網站 下載適用於 Windows 的安裝包。  解壓縮下載的檔案到某個目錄，例如：C:\nginx\。 |
| 2.安裝Node.js  前往 Node.js 官方網站 下載 Windows 版的 Node.js 安裝包。  下載並執行安裝程式，按照預設選項安裝。  安裝完成後，在 CMD 中檢查 Node.js 和 npm 是否正確安裝：  node -v  npm –v |
| 3.安裝MySQL Workbench 8.0 CE  前往 MySQL 官方網站的下載MySQL Workbench 8.0 CE。  下載並執行安裝程式，按照預設選項安裝。  連接MySQL資料庫 |
| 4.在No-IP設定網域 |
| 5.設定 nginx 代理伺服器  進入 nginx 的安裝目錄，找到nginx.conf 文件並打開。  設定添加至伺服器區塊中，將請求代理到 Node.js 應用程式，運行在 localhost:5000 上，並設定域名。 |
| 6.執行backend.py程式碼，使後端與前端連結 |

表 11-1‑4系統管理表

|  |
| --- |
| 1.動態 IP 解析與 No-IP 設定  No-IP必須保持運行，以便動態解析伺服器的 IP 地址。如果 IP 地址變更，No-IP 將自動更新其與域名的綁定。  如果 No-IP無法正常運行，可以嘗試重新設定。 |
| 2.nginx 服務管理  檢查 nginx 是否正常運行：  tasklist /FI "IMAGENAME eq nginx.exe"  如果 nginx 停止運行，重新啟動：  start nginx |
| 3.資料庫備份  定期備份 MySQL 資料庫：  mysqldump -u root -p 113-113412 > C:/path/to/backup/backup.sql |

# 使用手冊

## 入口網站

一張含有 文字, 螢幕擷取畫面, 網站, 網路廣告 的圖片

自動產生的描述

圖 12-1‑1入口網站

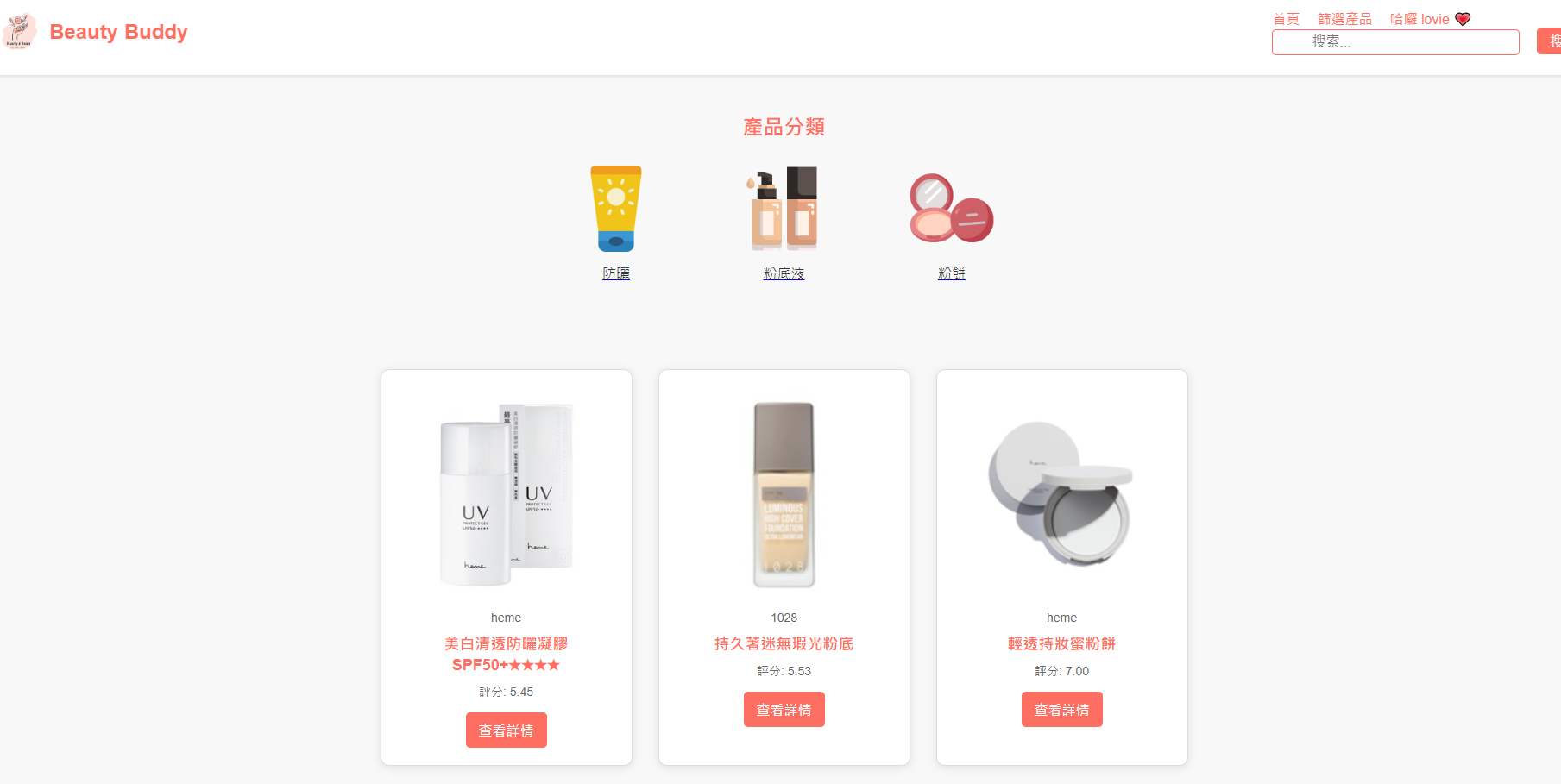


圖 12-1‑2產品分類及前三評分產品

## 產品分類

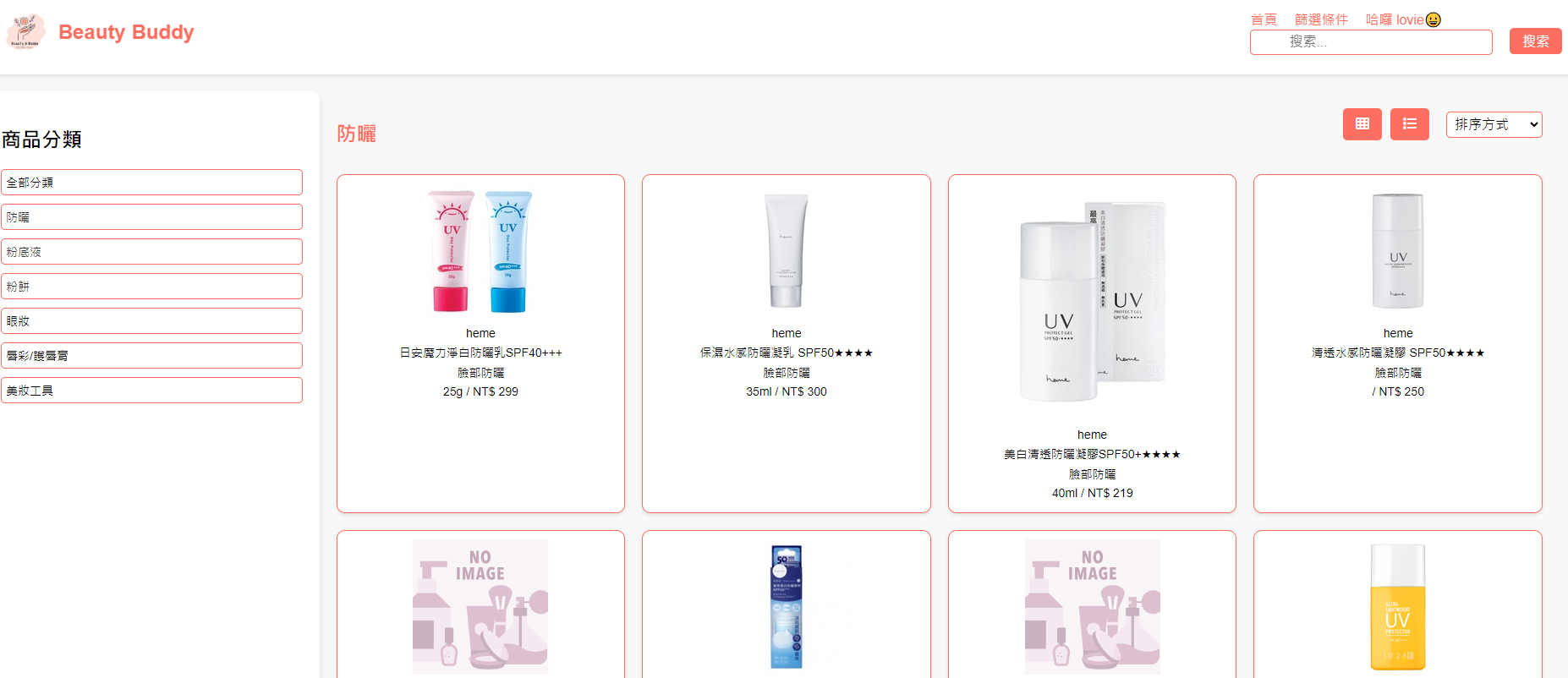


圖 12-2‑1產品分類頁面



圖 12-2‑2產品詳情以及可收藏產品

## 會員頁面



圖 12-3‑1會員註冊



圖 12-3‑2會員登入

登入後，會自動跳轉至入口網站，即可下滑開始肌膚檢測。



圖 12-3‑3肌膚檢測

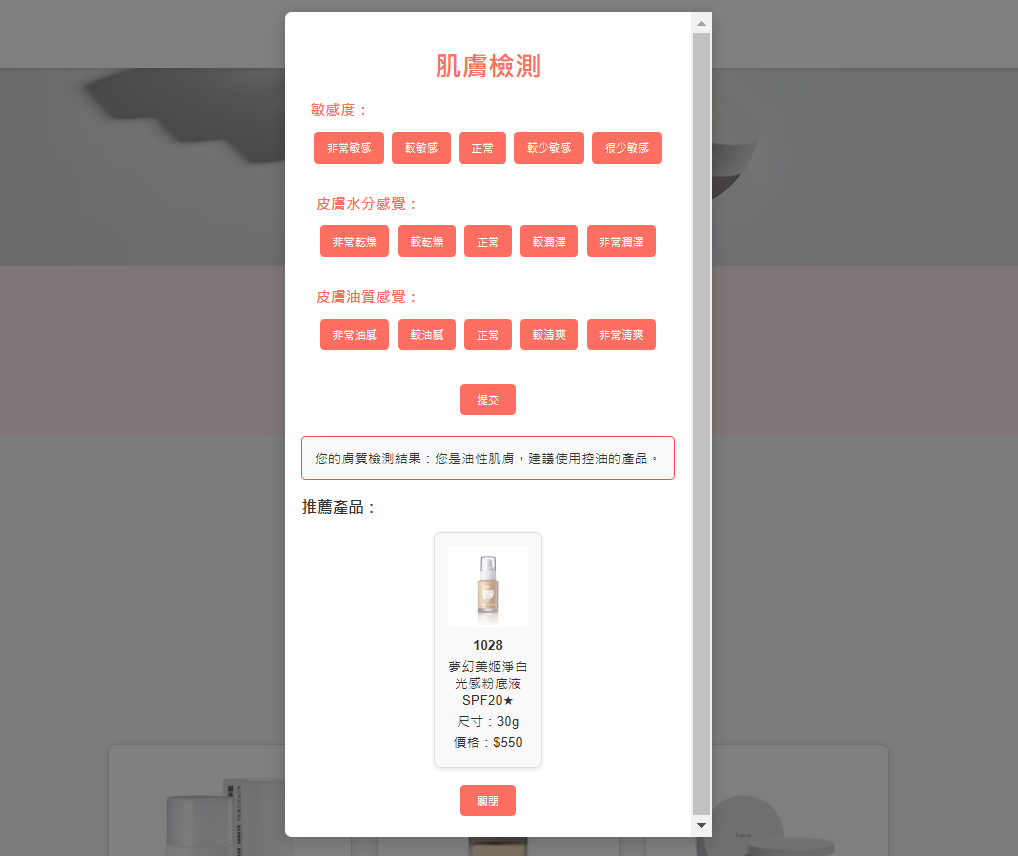


圖 12-3‑4肌膚檢測完成並依照膚質推薦匹配度最高產品畫面



圖 12-3‑5會員頁面之肌膚檢測記錄

此處將會記載每次測驗之選擇、結果以及檢測日期。

一張含有 文字, 螢幕擷取畫面, 軟體 的圖片

自動產生的描述

圖 12-3‑6會員頁面之關注收藏

未來將會新增「關注收藏」區域，可以將喜歡的產品收藏至此，即可快速查詢。

一張含有 文字, 螢幕擷取畫面, 字型, 設計 的圖片

自動產生的描述

圖 12-3‑7會員頁面之關於我

一張含有 文字, 螢幕擷取畫面, 數字, 字型 的圖片

自動產生的描述

圖 12-3‑8會員頁面之新增評論

於此處，您可以新增美妝產品之相關評論，評分選項分為7個等級可供選擇。



圖 12-3‑9會員頁面之我的評論

## 篩選條件



圖 12-4‑1篩選條件(登入狀態)

於此處，您可以選擇理想之價格範圍、品牌偏好以及產品類別。選擇完成後，將依照您的肌膚檢測狀況進行產品推薦，如圖12-4-2篩選結果所示。

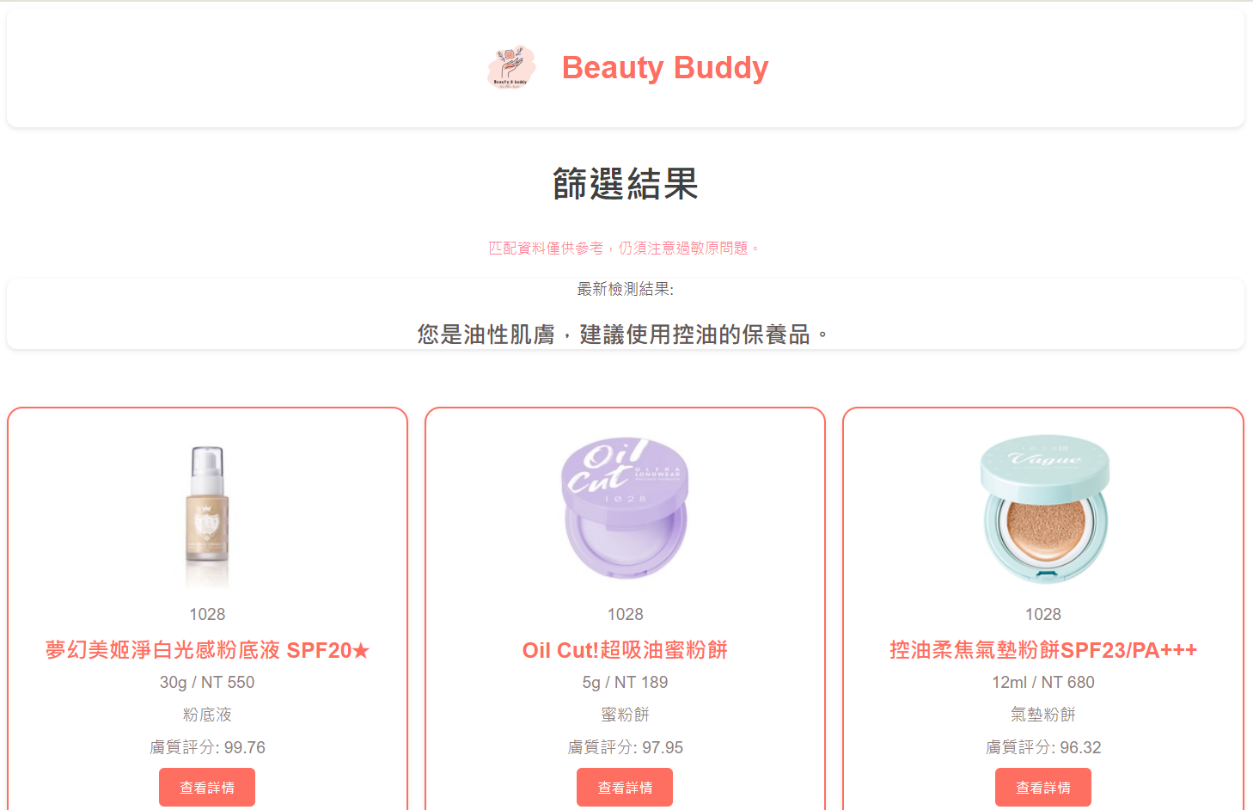


圖 12-4‑2篩選結果

在此處，您可以直覺地看到簡易產品說明、價格及膚質評分，點選下方「查看詳情」按鈕，將會跳轉至圖 12-2-2產品詳情畫面。



圖 12-4‑3篩選條件(未登入狀態)

未登入狀態在按鈕上方會顯示「你還沒登入哦～沒有帳號趕快註冊！來測試膚質查看您與產品的匹配度吧」等提示。

# 感想

11046068 陳力榆

在畢業專題製作過程我學習如何完成前端與後端的完整開發，因為課堂上學的是個別如何開發，沒有完整開發過系統的經驗，整個過程中遇到了許多挑戰。雖然我選擇的是行銷科技模組，課程上較少接觸程式撰寫，但這次專題給了我一個很好的機會去實踐並加強技術能力。在專題開發的過程中，無論是前端的設計還是後端的邏輯實現，都讓我感受到技術上的複雜性和挑戰。尤其是在實現一些功能時，我遇到了不少瓶頸，感覺自己的技術能力需要大幅提升。幸運的是，我能夠向技術比較熟悉的同學請教。透過他們的指導，我不僅解決了眼前的技術問題，還學到了許多程式設計中的原則和最佳實踐方法，這讓我在理解和應用技術時變得更加得心應手。這次專題讓我深刻體會到合作與求助的重要性，當自己能力不足時，向他人尋求幫助是一種有效的學習方式。最終也成功完成了專題整個架構，雖然整體成品或許還不夠完美，細節上也有許多改進空間。從初期的構思到最終的實現，我不僅提升了程式技術，也增強了自我解決問題的能力。未來，我會持續精進這些技能，期待在技術層面有更多突破，並應用在實際工作中。

11046075 郭岳鋆

在這次的專題製作過程中，我經歷了許多挑戰，無論是技術層面的困難，還是思路卡關的時刻，都讓我一度感到挫折。但正是這些困難促使我不斷反思與改進。每當遇到問題時，我學會了如何冷靜分析問題，尋找解決方案，並且嘗試不同的方式來克服困難。儘管過程並不總是順利，有時甚至會有進度停滯的情況，但我沒有放棄，一步一步堅持走下去。在這個過程中，我不僅提升了專業技能，還學到了如何更好地管理時間、協調資源，以及在壓力下保持冷靜的態度。最終，當專題順利完成時，我感到無比的成就感與滿足感。這段經歷讓我深刻體會到，困難是成長的催化劑，而每一個小小的進步都是努力的累積。無論結果如何，這次專題的製作過程讓我對未來充滿了信心，並堅信自己能夠迎接更多的挑戰。

11046074柯恩璇

製作本次專題面對的最大挑戰就是對程式設計近乎完全陌生，卻要開發網站來完成專題。身為行銷科技模組的學生，僅憑Python的基本邏輯，要完成這麼多系統架構，感覺像是一個不可能的任務。

在實際操作中，發現自己能幫上的忙有限。即便運用工具、查閱資料，由於缺乏實際製作的經驗，對於撰寫程式的過程真的非常困難。隨著專題進行，系統的複雜度不斷增加，網站的完善與調整也變得愈加困難，讓我感到十分無力。

不過十分謝謝組員同力協力使專題順利完成，過程中也學習到許多新知識，雖然程式設計能力仍需提升，但這次經驗也將成為未來成長的養分。但仍希望未來系上在面對非數位科技模組的學生能更加彈性，或在課程設計上接觸更多相關內容，為專題做好更充分的準備。

11046078 張明蓉

這次製作專題的經驗這對我來說是一個全方位的學習體驗，涉及到了許多我之前沒有接觸過的技術。透過查詢資料並自學，學到了很多且了解得更深入，像是如何整合後端與前端、如何使用機器學習來進行分析和預測等。

從最初的概念構想到一步步實際開發，過程中也遇到了許多挑戰，但一一解決難題後，也為了成果感到開心。無論是在設計資料庫架構，還是在開發肌膚檢測功能時，都遇到過技術困難與瓶頸，透過不斷嘗試，才逐漸克服了這些問題。製作專題讓我對如何從零開始構建一個完整的專案有了更深刻的了解，也讓我更加地意識到開發專案的不容易。

總而言之，這次專題製作對我而言是一個非常寶貴且具有挑戰的經驗。它不僅提升了我的技術能力，也教會了我如何在團隊合作中有效地溝通與解決問題，很感謝這一路上老師、同學給予我們的幫助，和這段時間大家的包容與努力，才讓我們有所收穫。

# 參考資料

1.SPSSAU(2021年12月30日)。〈信度係數的計算公式你了解嗎？〉。取自

<https://zhuanlan.zhihu.com/p/451618880>

2.Charlie Lee(2021年3月22日)。〈System Design: 系統架構基礎 - 可靠、可擴展、可維護〉。取自

<https://medium.com/bucketing/system-design-%E7%B3%BB%E7%B5%B1%E6%9E%B6%E6%A7%8B%E5%9F%BA%E7%A4%8E-%E5%8F%AF%E9%9D%A0-%E5%8F%AF%E6%93%B4%E5%B1%95-%E5%8F%AF%E7%B6%AD%E8%AD%B7-77903e2ce16a>

3.arbin(2023年9月11日)。〈[Day 1] 什麼是Flask?〉。取自

<https://ithelp.ithome.com.tw/articles/10316731>

4.Derek Liao(2020年3月22日)。〈NLP的基本執行步驟(I) — 語料的預處理(Preprocessing)〉。取自

<https://medium.com/@derekliao_62575/nlp%E7%9A%84%E5%9F%BA%E6%9C%AC%E5%9F%B7%E8%A1%8C%E6%AD%A5%E9%A9%9F-i-%E8%AA%9E%E6%96%99%E7%9A%84%E9%A0%90%E8%99%95%E7%90%86-preprocessing-8538f0b763d6>

5.Heidi\_Liu(2023)。〈Day16 - Node.js 是什麼？為什麼大家都在使用？〉。取自

<https://ithelp.ithome.com.tw/m/articles/10331821>

6.marasimc(2022年12月17日)。〈【论文精度】生成式预训练模型——BART（Bidirectional and Auto-Regressive Transformers）〉。取自

<https://blog.csdn.net/weixin_45828972/article/details/128349718>

# 附錄

|  |  |
| --- | --- |
| 評審建議事項 | 1.商業模式設計不足，未考慮網站營運問題  2.網站資料來源可能會有版權問題，且缺乏會員管理機制  3.不要有大陸用語(例如信息)  4.思考是否有會員制(個資可最小化)  5.營利方式  6.專注某一客群  7.匿名留言機制問題(黑函) |
| 修正情形 | 1.網站未來預計將與美妝品牌合作(例:廣告合作、聯名活動) 圖2-2-1商業九宮格  2.(1)資料來源有詢問機構是否能申請他們的資料庫進行學術上的分析與合作(簡報有補充、標明出處)  (2)關於品牌方的圖片，未來也預計與品牌方以推廣互惠的方式合作(如:購買連結)  3.已修正用語(系統手冊)  4.已新增會員制，記錄資料(網站功能)  5.已修正 2-2-1商業九宮格圖  6.修改成只針對底妝產品有需求的使用者  7.簡化評論功能會檢測情緒化字眼(如:去死)、字數過低等等問題並顯示原因 |