**逢甲大學 資訊工程學系**

**專題研究(一)**

**期末書面報告**

**虛擬展會平台**

**組員：資訊三甲 D1150136 陳可軒**

**資訊三丙 D1150166 曾予菲**

**資訊三甲 D1150504 徐承毅**

**指導老師：陳錫民 教授**

**指導老師簽名：**

**目錄**

[壹、 專題名稱 3](#_Toc199939609)

[貳、 專題內容概述 3](#_Toc199939610)

[**一、 研究動機與目的 3**](#_Toc199939611)

[**二、 作品構想特色 4**](#_Toc199939612)

[(一) 沉浸式展會介面 4](#_Toc199939613)

[(二) 即時互動交流系統 5](#_Toc199939614)

[(三) 多角色功能設計 5](#_Toc199939615)

[(四) 社群式介面體驗 6](#_Toc199939616)

[(五) AI的結合與應用 6](#_Toc199939617)

[(六) 突破實體限制 7](#_Toc199939618)

[(七) 數據與成效分析 7](#_Toc199939619)

[**三、 與市場現有**App**比較 7**](#_Toc199939620)

[(一) 內容介紹 7](#_Toc199939621)

[(二) 功能比較 8](#_Toc199939622)

[**四、 主要功能介紹 9**](#_Toc199939623)

[參、 本學期專題進度介紹 10](#_Toc199939624)

[**一、 系統架構與技術 10**](#_Toc199939625)

[(一) 前端系統（Flutter & Unity） 10](#_Toc199939626)

[(二) 後端系統（Spring Boot） 11](#_Toc199939627)

[(三) AI 模型服務（Ollama） 11](#_Toc199939628)

[(四) 資料庫系統（MySQL） 11](#_Toc199939629)

[**二、 資料庫設計 12**](#_Toc199939630)

[**三、 使用者介面設計 12**](#_Toc199939631)

[**四、 前端開發進度 12**](#_Toc199939632)

[(一) 使用者認證系統 13](#_Toc199939633)

[(二) 主應用程式框架與導覽 15](#_Toc199939634)

[(三) 內容瀏覽互動 16](#_Toc199939635)

[(四) 個人化功能 17](#_Toc199939636)

[(五) 展會與攤位 18](#_Toc199939637)

[**五、 後端開發進度 19**](#_Toc199939638)

[(一) 實作功能項目 19](#_Toc199939639)

[(二) 內部實作方式 19](#_Toc199939640)

[**六、** AI**功能開發進度 21**](#_Toc199939641)

[肆、 下學期預計進度 21](#_Toc199939642)

[伍、 專題分工說明 22](#_Toc199939643)

[陸、 資料來源 23](#_Toc199939644)

1. **專題名稱：虛擬展會平台**
2. **專題內容概述**
3. **研究動機與目的**

**什麼是展會？**

展會，又稱為展覽或展覽會，是指在特定時間內舉辦、聚集各類團體、組織或企業，供大眾參觀的一種集會活動。

展會通常具有明確的主題（如商品、藝術、服務、科技等），主要目的為宣傳或展示特定內容。其主要特色為，參展單位可於展會現場以擺設攤位、張貼海報及發放宣傳單等方式進行推廣與交流。又依舉辦形式的不同可分為：實體展會、巡迴展會、網路展會等類型。

**對於參觀者而言**

根據我們以往參加實體展會（如就業博覽會）的經驗，展會現場經常出現人潮擁擠、參觀動線混亂等問題，導致行走不便、甚至降低參觀意願。此外，展場工作人員亦時常因人力不足，無法及時協助並妥善服務所有參觀者，進一步影響整體參與體驗與滿意度。前往展場的交通亦常面臨不便或需耗費大量時間的情形，進一步降低參與的便利性與積極性。參觀者往往在投入大量時間、精力與金錢後，卻未必能獲得預期的資訊與成果，整體參展體驗因而受到影響。

**對於策展單位與參展單位而言**

從場地租金、人力調度、攤位裝潢，到文宣品設計與各項行政支出，皆需投入可觀的成本。更不用說，實體展會常受限於場地與時程，不僅在展前需要花費大量時間尋找合適的地點、規劃場地設計與動線，展會期間亦需動員大量人力進行現場管理、宣傳與解說，整體負擔因此提升。

此外，若策展方希望吸引更多海外或跨區的參展單位或參觀者，亦需額外投入交通補助、住宿協助等資源，進一步增加展會執行難度與預算負擔。若參展單位無法在展會期間獲得符合期望的曝光或收穫，將可能降低其未來再次參與設展的意願，對展會的長期經營造成不利的影響。

最重要的是，實體展會在籌辦的過程中時常面臨許多突發狀況，如設備故障、疫情等，相較於線上展會更難以即時應對與處理。此外，在展會後續追蹤與成效分析方面，實體展會往往缺乏完整且可量化的數據，使得成效評估困難且效果有限，進一步限制了展會價值的延伸與提升。

因此，綜合我們過往參與展會的經驗，並考量到實體展會在執行成本、管理難度、風險管控、成效追蹤等層面的諸多限制，我們計畫開發一款虛擬展會平台App。透過線上互動參展機制，重塑傳統以實體為主的展會運作模式，不僅能突破場地與地理的限制，更可大幅提升展會的整體曝光、參與體驗與營運效率。

1. **作品構想特色**

除了基本的會員系統、搜尋與瀏覽功能外，我們預計採用類似社群軟體的操作概念進行設計，並以遊戲化的互動模式呈現主要展場畫面。整體介面將結合即時互動元素與AI技術，打造一個數位化、互動性高且具沉浸式體驗的虛擬展會空間，讓參觀者即使透過線上方式，也能擁有不亞於實體展會的參與體驗。主要特色如下：

1. **沉浸式展會介面**

主畫面採類遊戲化介面設計，參觀者可於展場中自由操控角色移動，瀏覽感興趣的攤位內容，亦可透過點擊與攤位或其他人進行互動，提升參與感與趣味性，進一步模擬實體展會的參展體驗。

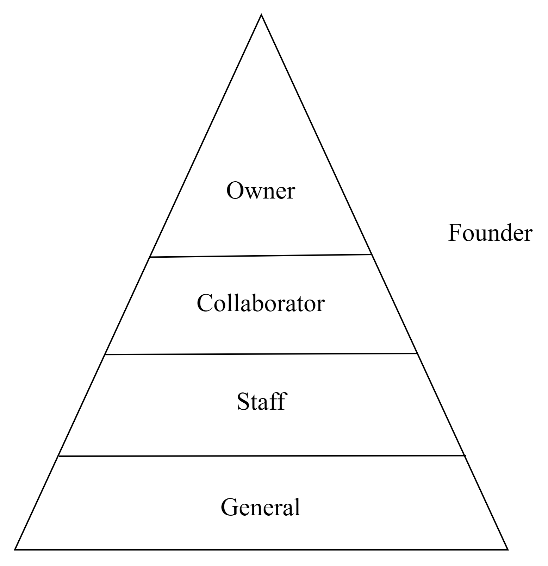
1. **即時互動交流系統**

內建文字及語音聊天功能，提供個人私訊聊天室與展會共通聊天室，參觀者可隨時向其他使用者或攤位工作人員發起對話，打破虛擬平台常見的孤立體驗，讓資訊不再停留於單向傳遞，並促進即時資訊交流與互動。

1. **多角色功能設計**

依照使用者的參與身分與實際需求，設計具彈性且層級明確的多角色系統，提供不同權限與操作介面，強化管理與使用者體驗：

圖 1、角色層級金字塔圖



1. **雙角色身分切換：**

每位使用者具備以下兩種角色，可依需求於系統中自由切換，對應不同使用功能與首頁內容，(見圖 1)：

* 1. **General(觀展人)：**對應參觀者角色。首頁顯示可參觀之展會清單，使用者可自由瀏覽、進行互動與探索。
  2. **Founder(策展人)：**對應管理者角色。首頁顯示其所擁有或參與策劃之展會與攤位。於展會中具特殊身分標示。

1. **策展人角色階級：**

若使用者切換為策展人(Founder)身分，又可據其權限的不同，進一步區分為三種策展角色：

* 1. **Owner(擁有者)：**對應展會／攤位的創建者。擁有完整的管理與編輯權限，並可指派使用者合作者(Collaborator)或工作人員(Staff)。
  2. **Collaborator(合作者)：**由Owner指派，擁有與Owner幾乎相同的編輯與管理權限，唯無刪除展會／攤位之權限。
  3. **Staff(工作人員)：**無管理或編輯權限，僅於展會中具身分標示。

1. **社群式介面體驗**

結合社群軟體常見操作方式與介面風格，提供更直覺且親切的使用畫面，降低學習門檻，讓參觀者能輕鬆無負擔的使用此App，提升整體使用意願。

1. **AI的結合與應用**

本專案的另一項核心特色在於廣泛導入人工智慧技術，不僅順應數位科技發展趨勢，不僅提升資訊傳遞的效率，更強化互動性、智能化應用與個人化體驗，提供更加沉浸的數位參展體驗。具體應用包括：

1. **AI導覽員功能**

每個展會與攤位皆配置專屬AI導覽員，能自動讀取介紹內容，並即時回應參觀者常見問題、導覽展出內容，甚至推薦其他相關展品或攤位，提供不中斷、個人化且高效率的參觀體驗。

1. **展會數據分析**

每個展會與攤位皆內建結合AI技術的數據分析頁面，能讀取系統所紀錄的使用者行為數據，進一步生成分析報告與優化建議，協助策展單位與參展單位制定未來優化策略與發展方向。

1. **個人化推薦頁面**

系統將根據使用者的參觀行為（如瀏覽紀錄、停留時間、互動紀錄等），透過AI即時生成個人化推薦頁面，向使用者推薦可能感興趣的展會或攤位內容，提升使用者體驗、探索效率與參展深度。

1. **突破實體限制**

不再受限於時間與空間，使用者可隨時隨地舉辦或參與展會，有效降低實體場地、人力調度等方面的成本，大幅提升營運效率與資源利用。

1. **數據與成效分析**

系統將自動記錄展會與攤位中的使用者行為數據，即時統計參觀人次、平均停留時間、互動次數等關鍵數據，並以圖表與報表的形式視覺化呈現，讓成效一目了然，同時結合AI協助進行數據分析。此功能有助於追蹤展會互動成效，彌補實體展會無法量化成果的限制。

1. **與市場現有App比較**
   1. **內容介紹**

|  |  |  |  |
| --- | --- | --- | --- |
| **Logo** | Virtual EXPO 2025 JAPAN - Apps on Google Play | 一張含有 字型, 圖形, 設計 的圖片  AI 產生的內容可能不正確。 | 一張含有 文字, 螢幕擷取畫面, 電子藍, 標誌 的圖片  AI 產生的內容可能不正確。 |
| **App名稱** | **VIRTUAL EXPO 2025** | **Accelevents** | **Xbitural** |
| **用途介紹** | 日本關西2025大阪萬國會TECH WORLD館架設的虛擬展館 | 可隨時發起、參與各項活動會議的行動應用程式 | 可隨時發起、參與各項展會的行動應用程式 |
| **主畫面呈現** | 3D虛擬建模場景 | 視訊與文字 | 2D類遊戲化界面 |
| **即時互動** | 大廳提供文字及語音交流 | 提供留言板、個人聊天室等功能 | 提供個人私訊及展會共通聊天室 |
| **AI功能** | 無 | 無 | 提供AI導覽員與數據分析等功能 |
| **數據分析** | 無，因為沒有提供創建展會功能 | 網頁中提供數據及圖表，並可供下載 | App中提供數據及圖表，並結合AI給予優化建議 |
| **適用裝置** | * iOS App * Android App | * iOS App * Android App * WEB | * iOS App * Android App |
| **操作設計** | 整體為遊戲風格，提示充足 | 功能繁多且跨平台：App可簡單設定及參與活動；WEB則是可詳細設定活動及查看數據分析，不能參與活動 | 以社群介面風格與類遊戲化界面為主，展會創建步驟單純 |
| **特色功能** | * 可自行創造虛擬角色 * 具備AR功能 * 沉浸式體驗 | * 亦可結合實體活動使用 * 提供詳細的數據分析 | * 導入AI應用 * 任何人皆可創建、參與展會且具沉浸感 |

* 1. **功能比較**

|  |  |  |  |
| --- | --- | --- | --- |
| App名稱 | **VIRTUAL EXPO 2025** | **Accelevents** | **👑Xbitural** |
| **應用介面** | | | |
| **2D虛擬場景** |  |  | ✔ |
| **3D虛擬場景** | ✔ |  |  |
| **虛擬角色** | ✔ |  |  |
| **AR功能** | ✔ |  |  |
| **即時互動功能** | | | |
| **文字交流** | ✔ | ✔ | ✔ |
| **語音交流** | ✔ | ✔ | ✔ |
| **視訊交流** |  | ✔ |  |
| **個人聊天室** |  | ✔ | ✔ |
| **展會共通聊天室** | ✔ | ✔ | ✔ |
| **AI應用** | | | |
| **AI導覽員** |  |  | ✔ |
| **AI分析建議** |  |  | ✔ |
| **個人化推薦主頁** |  |  | ✔ |
| **展會相關功能** | | | |
| **自由創建展會** |  | ✔ | ✔ |
| **結合實體活動** |  | ✔ |  |
| **數據分析報表** |  | ✔ | ✔ |
| **操作難易度** | | | |
| 學習門檻 | 低 | 高 | 中 |

1. **主要功能介紹**

一張含有 螢幕擷取畫面, 字型, 文字, Rectangle 的圖片

AI 產生的內容可能不正確。

圖 2、主要功能流程圖

1. **本學期專題進度介紹**

這學期的開發重點在於建構應用的核心使用者體驗流程，從使用者帳號的建立與管理，到內容的瀏覽、互動，以及個人化設定，同時也完成了應用的大部分 UI/UX 設計。

1. **系統架構與技術**

一張含有 文字, 螢幕擷取畫面, 字型, 標誌 的圖片

AI 產生的內容可能不正確。

圖 3、技術使用架構圖

我們採用前後端分離的架構設計，整合行動應用開發、後端伺服器、資料庫，以及AI模型服務，打造一個智慧化、互動性強的展覽平台。各項元件的角色與互動關係如下，(見圖 3)：

* 1. **前端系統（Flutter & Unity）**

1. **Flutter**

用於開發主應用介面與使用流程，提供跨平台的行動裝置使用體驗，並負責應用邏輯與畫面控制。

1. **Unity**

專注於展覽互動場景的呈現，運用2D類遊戲化方式設計展區，提升沉浸感與互動性。Unity可透過嵌入或模組整合與 Flutter 溝通，實現前後無縫切換。

* 1. **後端系統（Spring Boot）**

後端採用Spring Boot作為主要框架，提供穩定且高效的RESTful API，處理：

* + 1. 前端請求的接收與業務邏輯的處理
    2. 資料存取與轉換
    3. 與AI模型及資料庫之間的整合作業

透過模組化設計，系統具備良好的擴充性與維護彈性。

* 1. **AI 模型服務（Ollama）**

系統整合了本地部署的Ollama AI模型平台，支援多項自然語言處理任務，包括：

* + 1. 語意理解與分析
    2. 使用者問答回應
    3. 展覽推薦與內容個人化

AI模組以服務形式提供，前後端皆可透過API呼叫，提升整體系統智慧化程度。

* 1. **資料庫系統（MySQL）**

資料儲存方面，採用 MySQL 資料庫，管理以下核心資訊：

* + 1. 使用者帳號與操作紀錄
    2. 展覽與攤位內容資料
    3. AI 分析結果等結構化數據

後端透過Spring Boot ORM框架與資料庫互動，確保資料一致性、安全性與查詢效率。

1. **資料庫設計**

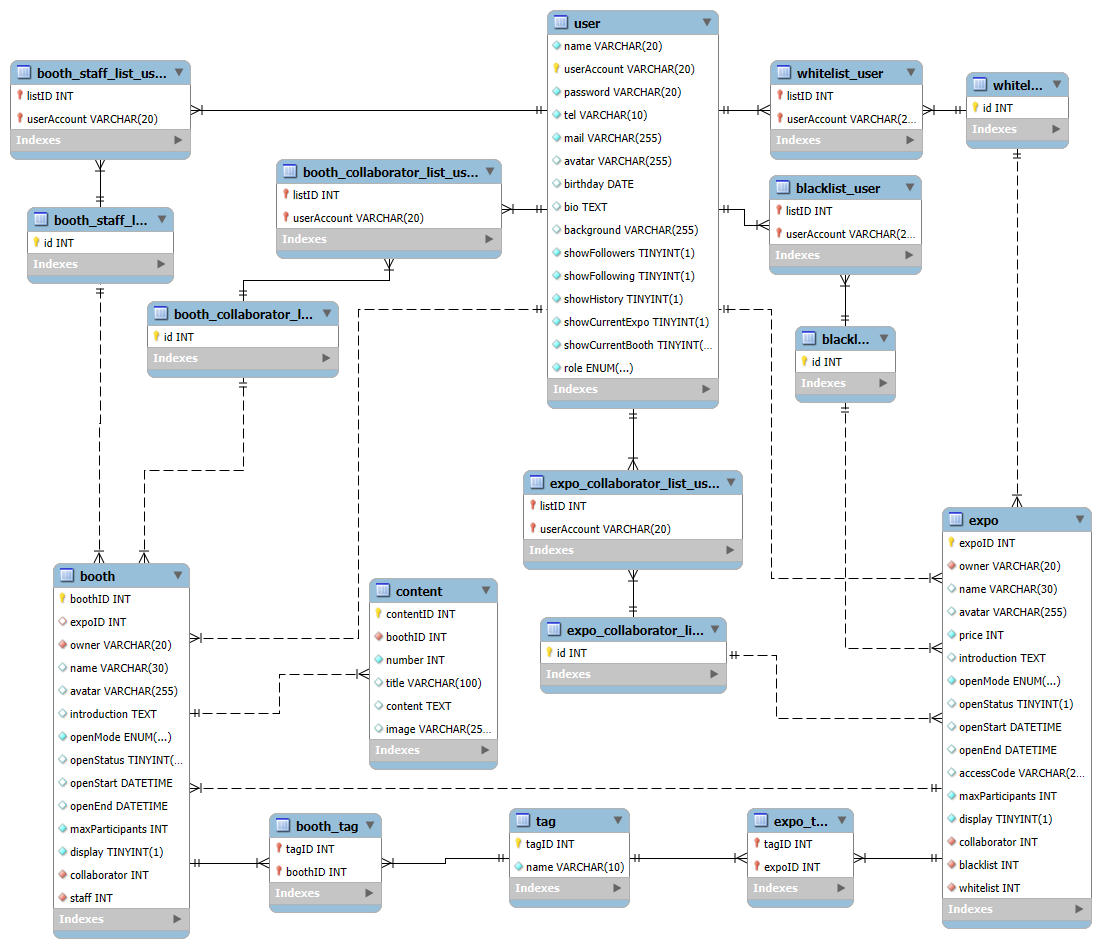
****

圖 4、ERD (不含歷史紀錄)

1. **使用者介面設計**

UI介面設計檔案連結：[Figma Link](https://www.figma.com/proto/vcxmU1vUuFK1YpW8oaO7YC/App--S21-ultra-?node-id=8-623&t=tgSOBuoehSPzt2eG-0&scaling=scale-down&content-scaling=fixed&page-id=0%3A1&starting-point-node-id=8%3A623&show-proto-sidebar=1)

1. **前端開發進度**

本學期已完成應用程式的整體 UI/UX 設計。整題著重視覺觀感盡量一致，設計目標是能用簡單易懂的主程式為主題，並在Flutter中實作，而展覽以及攤位設計實現會在Unity中完成。已完成多個核心功能，如下：

* 1. **使用者認證系統**
     1. **註冊 (Signup Page)**

已完成API連接。

1. 使用者名稱設定 (Username Setting Page)：包含長度、有效字元等驗證規則，（見圖 5）。
2. 電子郵件設定與驗證 (Email Page)：使用者需輸入Email並透過驗證碼進行確認。驗證碼部分僅完成UI，尚未實作驗證流程，（見圖 6）。
3. 密碼設定 (Password Setting Page)：包含密碼長度、大小寫字母及數字組合的驗證規則。前端有先執行驗證，避免浪費過多時間，(見圖 7)。

|  |  |  |
| --- | --- | --- |
| 一張含有 文字, 螢幕擷取畫面, 設計 的圖片  AI 產生的內容可能不正確。  圖 5、註冊流程1-  輸入使用者帳號 | 一張含有 文字, 螢幕擷取畫面, 軟體, 設計 的圖片  AI 產生的內容可能不正確。  圖 6、註冊流程2-  輸入密碼及驗證密碼 | 一張含有 文字, 螢幕擷取畫面 的圖片  AI 產生的內容可能不正確。  圖 7、註冊流程3-  輸入郵箱及驗證郵箱 |

1. 電話號碼設定 (Phone Number Page)：用於帳號恢復，(見圖 8)。
2. 頭像選擇 (Profile Picture Page)：提供預設頭像選擇或及上傳功能。上傳功能尚未實作，等待連接資料庫後實現，(見圖 9)。
3. 註冊成功提示 (Registration Successful Page) (見圖 10)

|  |  |  |
| --- | --- | --- |
| 圖 8、註冊流程4-  輸入電話 | 圖 9、註冊流程5-  選擇頭像 | 圖 10、註冊流程6-  註冊成功提示 |

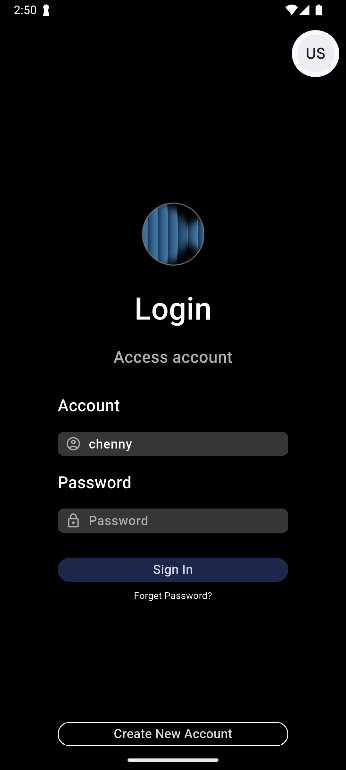
* + 1. **登入 (Login Page)**

圖 11、登入頁面

尚未完成API連接。

使用者可透過帳號（Email/使用者名稱）和密碼登入。登入成功後，使用者資訊將由 Current User Notifier 管理，(見圖 11)。

* + 1. **密碼重設 (Recover Account)**

尚未完成API連接。

* + - 1. 帳號驗證 (Verification Page)：使用者輸入註冊的識別資訊以接收驗證碼，(見圖 12)。
      2. 重設密碼 (Password Reset Page)：設定新的密碼，(見圖 13)。
      3. 成功提示 (Password Successfully Changed Page) (見圖 14)

|  |  |  |
| --- | --- | --- |
| 圖 12、密碼重設流程1-  帳號驗證 | 圖 13、密碼重設流程2-  重設密碼 | 圖 14、密碼重設流程3-  密碼重設成功提示 |

* 1. **主應用程式框架與導覽**
     1. **頂部應用欄 (Custom App Bar)：**

包含可點擊進入個人資料頁的頭像、搜尋框及訊息icon。

* + 1. **底部導覽列 (Custom Bottom Bar)：**

提供在「首頁」、「發燒」、「追蹤」和「個人資料」四個主要頁面間的切換。其中已完成個人資料頁面。

* + 1. **切換邏輯 (Navigation Helper)：**

處理底部導覽列的頁面跳轉邏輯。

* 1. **內容瀏覽互動**
     1. **首頁 (Home Page)**

尚未完成API連接，(見圖 15)。

展示不同展覽分類的標籤 (Category Tag)，根據選定分類或預設顯示推薦展覽，使用 Exhibition Card 滑動軌呈現。進行中的展覽，使用 Exhibition Card Large 滑動軌呈現。

* + 1. **展覽詳情 (Exhibition Entry Page)：**

尚未完成API連接，(見圖 16)。

點擊卡片觸發，顯示展覽細資訊，提供「Follow」策展人的功能，若展覽在展期內，則有「Enter」按鈕。

* + 1. **發燒 (Trending Page)：**

尚未完成API連接，(見圖 17)。

頂部輪播展示熱門展覽，下方瀑布流(Masonry Grid View)展示其他熱門攤位(Booth Data)的圖片。

* + 1. **追蹤頁面 (Following Page)：**

尚未完成。

|  |  |  |
| --- | --- | --- |
| 圖 15、首頁 | 圖 16、展覽詳情 | 圖 17、發燒 |

* 1. **個人化功能**
     1. **個人資料 (Profile Page)：**

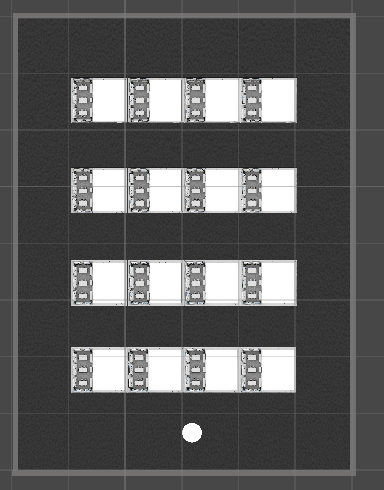
部分完成API連接，(見圖 18)。

* + - 1. 展示使用者的頭像、背景圖、名稱：已連接API。
      2. 追蹤者、使用時長、追蹤中的數量統計：未完成API連接。
      3. 個人簡介：已連接API。
      4. 按鈕列（點讚歷史、設定、發文、身分切換）：未完成API連接。
      5. 「最近參與」展覽列：未完成API連接。
      6. 關於我們、聯繫我們：未完成API連接。
      7. 登出：已連接API。
    1. **設定 (Settings Page)：**

皆尚未完成修改邏輯，(見圖 19)。

* + - 1. 基本資訊（姓名、生日、電話、電子郵件）
      2. 隱私設定（顯示追蹤者、追蹤對象、參與的展覽、歷史參與的展覽）
      3. 安全設定（更改密碼、停用帳號）

|  |  |
| --- | --- |
| 圖 18、個人資料頁面 | 圖 19、設定頁面 |

* 1. **展會與攤位**
     1. **展會設計：**

將會提供幾種展會的模板給使用者使用，每個模板之間可能會有不同的展會數量、攤位排列、展場樣式等。使用者可以在展會內走動，查看並進入有興趣的攤位。策展者則可以選擇展會的圖示、挑選展會模板等。(見圖 20)

圖 20、展會設計樣式

* + 1. **攤位設計：**

使用者可於攤位內展示作品，如圖片、影片。呈現方式畫展一樣，作品將展示於牆壁上，參展者可左右移動，瀏覽自己有興趣的作品以及作品介紹，並且可以與AI機器人交談，獲得更多作品相關的資訊。(見圖 21)

一張含有 油畫, 藝術, 現代藝術, 美術展覽 的圖片

AI 產生的內容可能不正確。

圖 21、攤位設計樣式

1. **後端開發進度**

本學期已利用Java的Spring Boot框架，完成本專題主要API之建置，並已Swagger UI工具進行測試與呈現功能結果(見圖 22、圖 23、圖 24)。完成以下功能：

* 1. **實作功能項目**

1. 前述資料庫設計中所有資料表的基本CRUD功能(Create, Read, Update, Delete)
2. 整合多數資料表間多對多與多對一的映射關係，如展會與攤位的一對多關係
3. 支援簡單查詢功能，如依標籤獲取展會資料
   1. **內部實作方式**
4. 採用MVC分層架構設計，分為Controller(接收及回傳請求)、Service(處理商業邏輯)、Repository(資料庫存取)三層實作後端API
5. 以RESTful API標準定義各功能對應的URL路徑與請求方式，並統一使用JSON格式作為資料回傳格式
6. 建立全域錯誤處理機制（Global Exception Handler），統一處理例外並回傳一致格式之錯誤訊息
7. 使用@Valid、@NotNull等註解進行資料驗證，並導入Lombok套件的@Data註解以簡化getter、setter等樣板程式碼
8. 引入DTO（Data Transfer Object）封裝資料內容，實現資料庫實體與對外傳輸資料的分離，降低耦合度、提升維護性
9. 使用Spring Boot JPA與資料庫溝通，透過 Entity 類別對應資料表，並由 JPA 自動生成查詢邏輯，簡化資料存取程式碼

一張含有 文字, 螢幕擷取畫面, 數字, 字型 的圖片

AI 產生的內容可能不正確。

圖 22、Swagger UI-主畫面

一張含有 文字, 螢幕擷取畫面, 數字, 字型 的圖片

AI 產生的內容可能不正確。

圖 23、Swagger UI-單一使用者相關API方法

**一張含有 文字, 螢幕擷取畫面, 軟體, 數字 的圖片

AI 產生的內容可能不正確。**

圖 24、Swagger UI-新增使用者功能測試畫面

1. **AI功能開發進度**

展場及攤位都設有AI聊天機器人，使用者在和機器人交談後，機器人會根據作品的介紹回答，讓使用者了解攤位、展示作品的相關資訊。面對策展者，則有專屬的AI分析，可以讓策展者更了解自己攤位的統計數據，例如參訪人數、平均攤位停留時間等資訊，AI將會根據這些統計數據，生成分析報告給使策展者。

1. **下學期預計進度**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | **六月** | **七月** | **八月** | **九月** | **十月** | **十一月** | **十二月** |
| **前端** | | | | | | | |
| 展覽資料的讀取與瀏覽 |  |  |  |  |  |  |  |
| 展覽搜尋功能 |  |  |  |  |  |  |  |
| 攤位資訊的顯示與整合 |  |  |  |  |  |  |  |
| 使用者資料的讀取與編輯 |  |  |  |  |  |  |  |
| 追蹤與互動功能 |  |  |  |  |  |  |  |
| 展覽與攤位的串接與整合 |  |  |  |  |  |  |  |
| 優化展會與攤位 |  |  |  |  |  |  |  |
| **後端** | | | | | | | |
| 歷史紀錄 |  |  |  |  |  |  |  |
| 聊天室相關功能 |  |  |  |  |  |  |  |
| 身分驗證 |  |  |  |  |  |  |  |
| 修改資料庫 |  |  |  |  |  |  |  |
| AI | | | | | | | |
| 優化AI機器人 |  |  |  |  |  |  |  |
| 數據分析建議功能 |  |  |  |  |  |  |  |
| 建立個人化主頁 |  |  |  |  |  |  |  |
| **綜合項目** | | | | | | | |
| 前後端串接 |  |  |  |  |  |  |  |
| 整合後端與AI |  |  |  |  |  |  |  |
| 系統測試與優化 |  |  |  |  |  |  |  |

1. **專題分工說明**

|  |  |  |  |
| --- | --- | --- | --- |
| 姓名 | 陳可軒 | 曾予菲 | 徐承毅 |
| 系級 | 資訊三甲 | 資訊三丙 | 資訊三甲 |
| 學號 | D1150136 | D1150166 | D1150504 |
| 分工 | * UI介面設計 * 前端開發 * AI功能開發 | * UI介面設計 * 資料庫設計 * 後端API開發 * 書面資料撰寫 | * UI介面設計 * 前端開發 * 前後端串接 |

1. **資料來源**

* [Wikipedia-展覽會](https://zh.wikipedia.org/zh-tw/%E5%B1%95%E8%A6%BD%E6%9C%83)
* [EcomPlus-品牌參加展覽有用嗎？美國參加展會優點、成本全攻略](https://www.ecomplusco.com/blog-posts/advantages-and-practical-suggestions-for-exhibiting-in-the-united-states)
* [BlueSign-虛擬展覽與真實展覽之間的比較：優缺點一覽](https://blog2.ibie.org.tw/blog/2023/10/27/%E8%99%9B%E6%93%AC%E5%B1%95%E8%A6%BD%E8%88%87%E7%9C%9F%E5%AF%A6%E5%B1%95%E8%A6%BD%E4%B9%8B%E9%96%93%E7%9A%84%E6%AF%94%E8%BC%83%EF%BC%9A%E5%84%AA%E7%BC%BA%E9%BB%9E%E4%B8%80%E8%A6%BD/)
* [Esther H-【盤點】7 款虛擬展間製作平台分享，從數位收藏到博覽展會都適用](https://zine.artogo.tw/%E7%9B%A4%E9%BB%9E-7-%E6%AC%BE%E8%99%9B%E6%93%AC%E5%B1%95%E9%96%93%E8%A3%BD%E4%BD%9C%E5%B9%B3%E5%8F%B0%E5%88%86%E4%BA%AB-%E5%BE%9E%E6%95%B8%E4%BD%8D%E6%94%B6%E8%97%8F%E5%88%B0%E5%8D%9A%E8%A6%BD%E5%B1%95%E6%9C%83%E9%83%BD%E9%81%A9%E7%94%A8-59fdeded4bf4)
* [vFAIRS-10 Best Virtual Exhibition Platforms [2025]](https://www.vfairs.com/blog/best-virtual-exhibition-platforms/)
* [TECH WORLD官網-VIRTUAL EXPO 2025](https://www.expo2025-techworld.com/TechWorld2025/wSite/ct?xItem=162&ctNode=472)
* [TECH WORLD-FB專頁](https://www.facebook.com/expo2025techworld/?locale=zh_TW)
* [Accelevents官網-About Us](https://www.accelevents.com/about-us)
* VIRTUAL EXPO 2025應用程式實際體驗
* Accelevents應用程式實際體驗
* [Virsody 官網](https://virsody.io/)
* [古古- Spring Boot 零基礎入門 系列](https://ithelp.ithome.com.tw/users/20151036/ironman/6130?page=2)
* [ian\_liu- Spring Boot+Android 30天 實戰開發 系列](https://ithelp.ithome.com.tw/users/20150395/ironman/6028)
* [clairehsy -從卡關到通關的Spring Boot 腦內風暴 系列](https://ithelp.ithome.com.tw/users/20163341/ironman/7989)
* [Spring Boot官網-Exception Handling in Spring MVC](https://spring.io/blog/2013/11/01/exception-handling-in-spring-mvc)
* [Wayne (偉恩)-【WebAPI】分層設計模式 - 三層式架構](https://wayne-blog.com/2023-02-24/webapi-3-tier-introduction/)
* [praveen sharma- Handle Exceptions in Spring Boot: A Guide to Clean Code Principles](https://medium.com/@sharmapraveen91/handle-exceptions-in-spring-boot-a-guide-to-clean-code-principles-e8a9d56cafe8)
* [HK- Spring Boot Many To Many Relationship](https://medium.com/@hk09/spring-boot-many-to-many-relationship-ac0ffd9b545d)
* [Dulanjaya Sandaruwan- Uploading Images in a Spring Boot Project: A Step-by-Step Guide](https://medium.com/@dulanjayasandaruwan1998/uploading-images-in-a-spring-boot-project-a-step-by-step-guide-8a55248ea520)
* [Ana Peterlić -@Operation vs @ApiResponse in Swagger](https://www.baeldung.com/swagger-operation-vs-apiresponse)
* [Jayden Lin- Java ♨️ 客製化 Lombok @Builder 的方法 (一)：傳入 Null 採用預設值](https://medium.com/%E7%A8%8B%E5%BC%8F%E7%8C%BF%E5%90%83%E9%A6%99%E8%95%89/java-%EF%B8%8F-%E5%AE%A2%E8%A3%BD%E5%8C%96-lombok-builder-%E7%9A%84%E6%96%B9%E6%B3%95-%E4%B8%80-%E5%82%B3%E5%85%A5-null-%E6%8E%A1%E7%94%A8%E9%A0%90%E8%A8%AD%E5%80%BC-d9cf76b5b2b)