

系統分析與設計

指導教授：陳仲儼

系統名稱：Event Registration System

成員：

110408003 楊筑安

110408532 梁志宇

110408530 王宥歲

110408539 黃宥棠

111403202 徐上哲

日期：2024/12/20

壹、系統介紹

活動註冊系統旨在提供用戶與活動主辦方一個便利的平台，促進活動的創建與參與。系統透過直觀的界面和清晰的功能設計，讓用戶可以輕鬆瀏覽活動、報名參加，並接收系統通知。同時，活動主辦方可以高效地管理活動的創建、修改與通知發送。

貳、功能介紹

一、用戶功能

1. 註冊與管理帳號

使用者可以在系統上進行帳號註冊，透過填寫電子郵件以完成註冊流程。註冊成功後，系統將為使用者創建專屬帳號。隨後，使用者即可使用註冊的帳號登入系統，進一步探索和操作系統提供的功能，例如參與活動、管理個人資訊。

2. 瀏覽活動

使用者可瀏覽系統上所陳列的活動，點選進入該活動網頁後可觀看其詳細資訊，例如活動名稱、舉辦時間、地點以及其他相關內容。此外，為了提升使用者體驗，系統還提供篩選和搜尋功能，使用者能根據自己的興趣或需求，快速找到符合條件的活動，例如按時間排序、依地點篩選或直接搜尋特定關鍵字，從而輕鬆找到適合參與的活動並進行下一步操作。

（上述功能在系統並未實作篩選及搜尋功能）

3. 活動報名

使用者可以從活動列表中選擇有興趣的活動，填寫個人資料後並完成報名。在填寫個人資料之過程中，系統會要求使用者填寫規定的個人資料格式，確保個人資料的正確性。報名成功後，系統將發送報名確認通知之電子郵件，確保使用者能更加確認已報名的活動。

4. 取消報名活動

使用者可以取消已報名的活動，並透過系統按下取消案按鈕。在取消過程中，系統將確認使用者的選擇，例如「確定要取消報名？」取消成功後，系統會同時更新其活動狀態，讓使用者能有效管理個人參與的活動紀錄。

5. 探索所有活動

此功能允許使用者或主辦方查看系統內的所有活動，包括活動的基本資訊（如名稱、日期、地點、描述等）。無論用戶是否登入，都可以瀏覽活動列表；若用戶已登入。

二．活動主辦方功能(系統中並未實作)

1. 創建活動

主辦方可以通過系統建立新的活動，設定活動資訊，包括日期、地點、活動名稱、活動介紹等，為使用者提供清晰的活動內容。系統還支援多種自訂功能，幫助主辦方管理活動，例如設置參與人數的上限、報名截止日期或其他特定條件，確保活動規劃更加完善。這些功能除了能協助主辦方有效率地管理活動，還能提升使用者體驗。

2. 更新及刪除活動

主辦方可以根據實際需求更新或刪除已建立的活動。例如，主辦方可以修改活動的日期、地點或活動介紹，以確保資訊的準確性。同時，系統允許主辦方向已報名參加活動的使用者發送通知，提醒活動資訊已被更新。例如活動的時間變更、場地調整或其他相關公告，確保使用者能及時獲取最新資訊，提升活動的順利進行與使用者的滿意度。

3. 探索所有活動

與第一點之「探索所有活動」功能相同。

參、系統分析之 UML 圖

一、使用案例圖(Use Case Diagram)

1. 功能描述

(1) 使用者的功能

- a. 使用者註冊
- b. 瀏覽活動
- c. 報名活動
- d. 取消報名活動

(2) 主辦方的功能

- a. 建立活動
- b. 更新活動資訊
- c. 取消建立活動

2. 使用案例之間的關聯

(1) 登入系統與報名活動之間為「包含」關係

使用者報名活動之前，必須要先登入系統。

(2) 登入系統與取消報名之間為「包含」關係

使用者取消已報名的活動，必須要先登入系統。

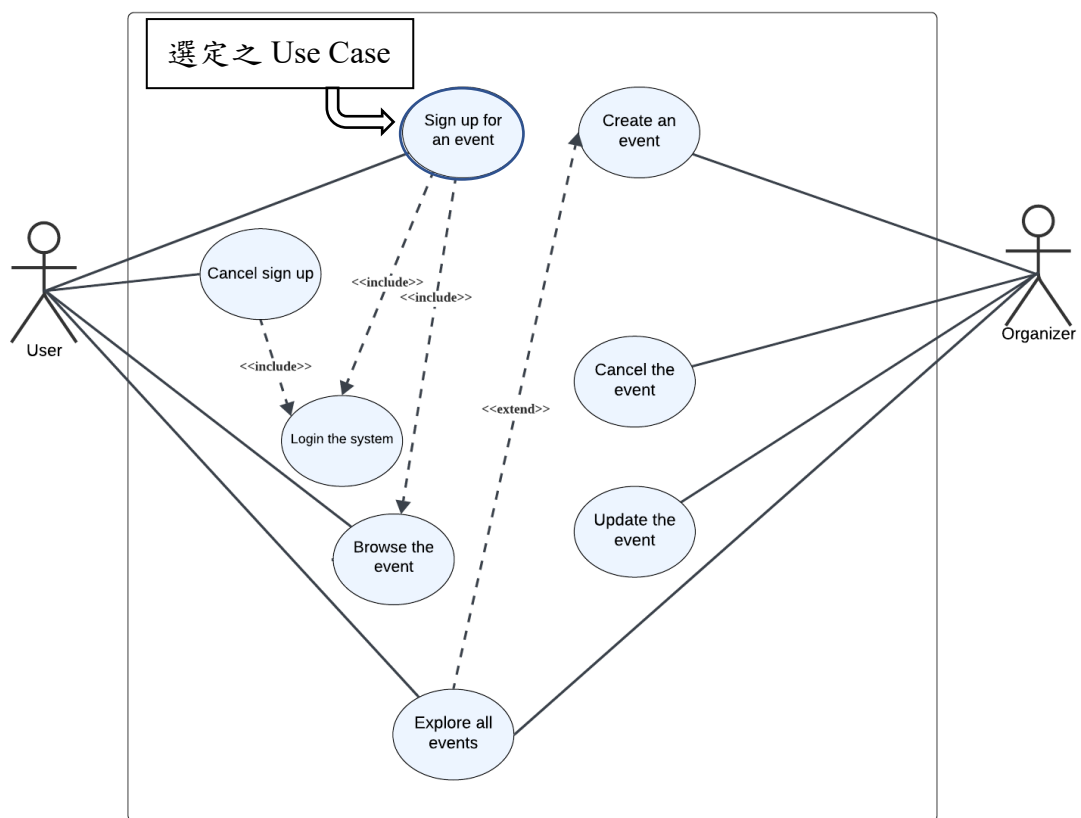
(3) 瀏覽活動與報名活動之間為「包含」關係

使用者報名對其有興趣的活動，必會先瀏覽該活動之詳細資訊。

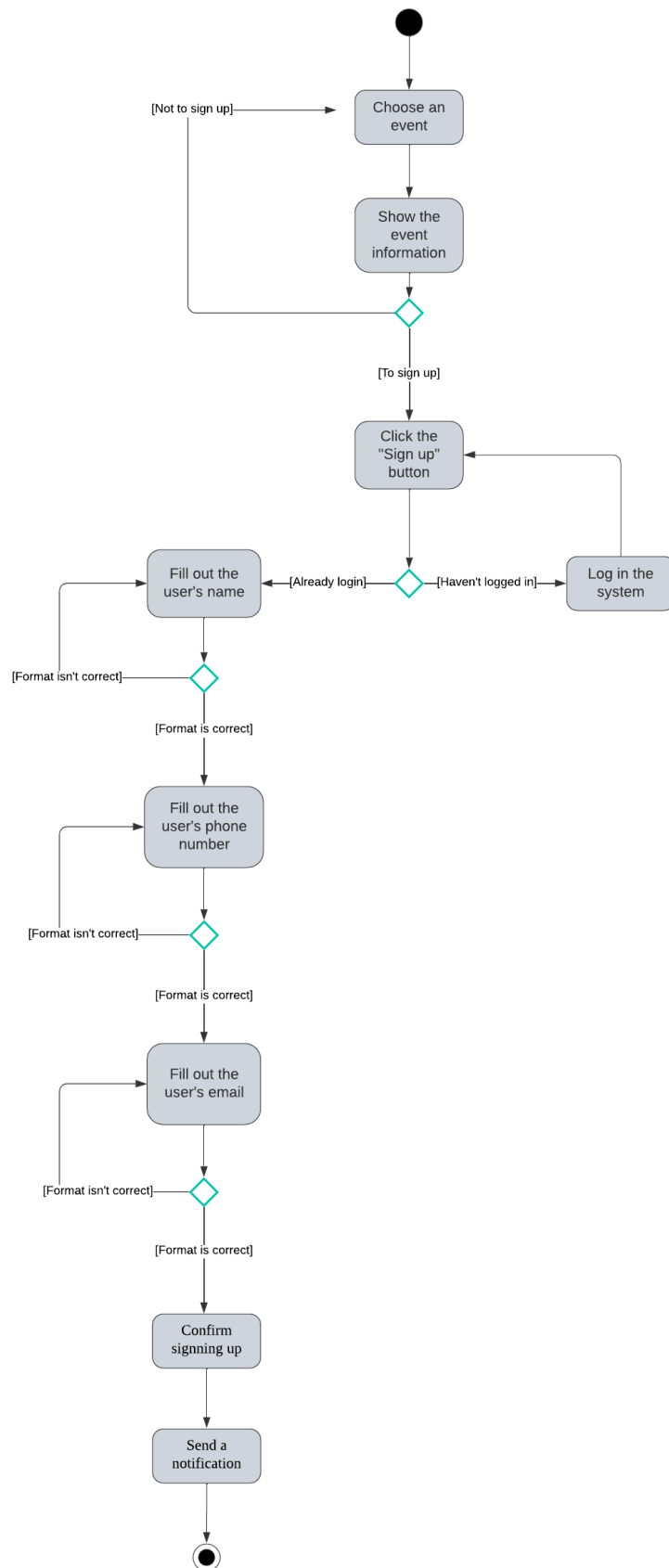
3. 探索所有活動與創建活動之間為「展延」關係

主辦方創建活動前，可探索系統上是否有類似的活動被創建。

3. 使用案例圖展示



二、活動圖(Activity Diagram)

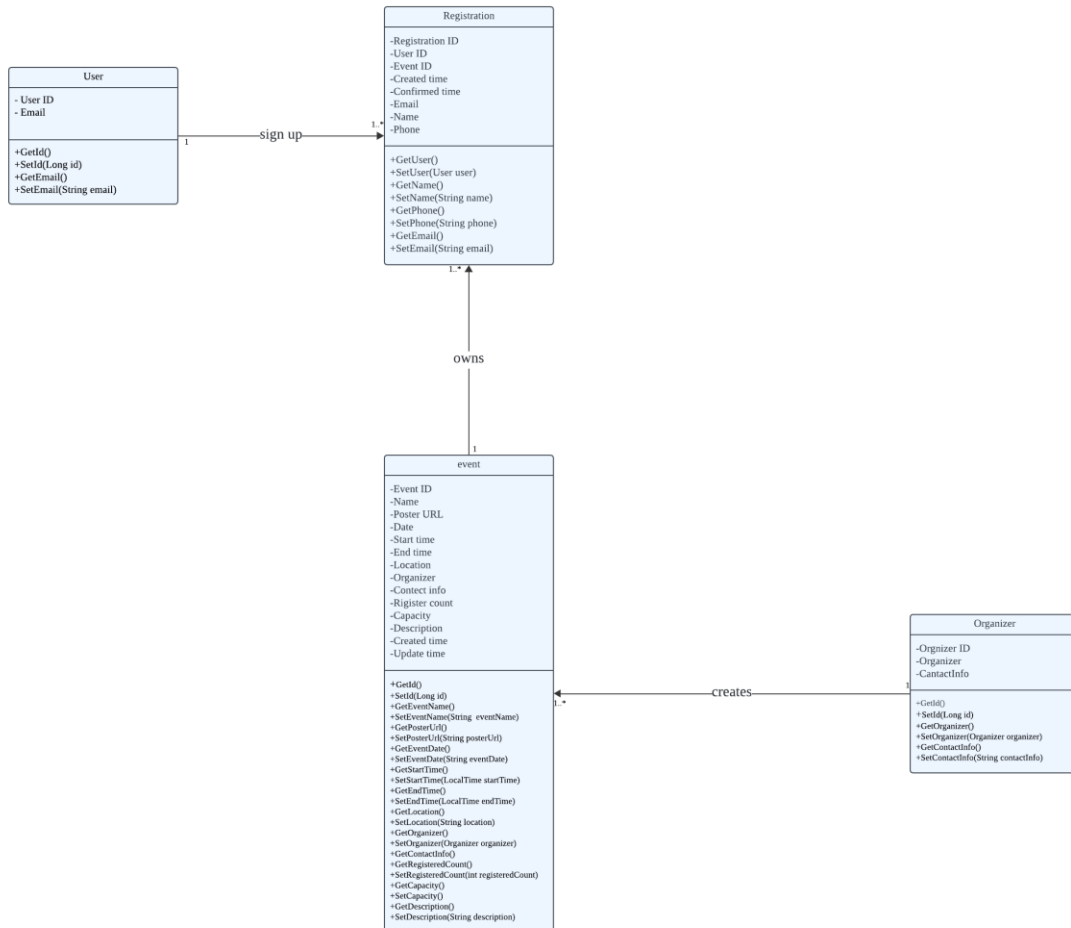


三、使用案例描述(UCD)

Use Case Name: Sign up for Activity	ID: 1	Importance Level: High
Primary Actor: User	Use-Case Type: Detail	
Stakeholders and Interests: User - wants to sign up for events. Organizer - wants to attract people to join events.		
Brief Description: This use case describes how users sign up for an activity.		
Trigger:	The user logs into the system and signs up for events.	
Type:	External	
Relationships: Association: User Include: Log in the System, Browse an event		
Normal Flow of Events: 1. The user selects an available event. 2. The system shows the event details. 3. If the user has no interest in registering the activity Returns to events selection. 4. The user clicks the "Sign up" button to request signing up for an event. 5. If the user hasn't logged in the system Execute the Login System use case. 6. Fill out the user's name 7. If the format is incorrect Fill out the user's name again. 8. Fill out the user's phone number 9. If the format is incorrect Fill out the user's phone number again. 10. Fill out the user's email 11. If the format is incorrect Fill out the user's email again. 12. The user confirms signing up for the event. 13. The user gets a notification about signing up for the event.		
SubFlows:		
Alternate/Exceptional Flows: 1. If the event is full, the user cannot sign up.		

四、類別圖(Class Diagram)

這張圖表展示了「使用者」、「主辦方」和「活動」類別之間的互動關係，並加入「報名流程」，展現了完整的活動管理系統的設計。



五、循序圖(Sequence Diagram)

1. 相關之類別

(1)使用者(User)

使用者觸發活動報名之流程

(2)PostMapping

處理用戶請求的入口，對應於程式碼中的 `@PostMapping` 註解。專門負責處理使用者發起的「報名」要求，將使用者的報名資料（例如選定活動之活動 ID、姓名、電話、電子郵件等）傳遞給其他實體進行處理。作為前端與後端之間的橋樑，確保報名資料正確接收和操作。

(3)HomeController

處理應用程式中的業務邏輯，為使用者界面與實體類別之間的橋樑。協調管理報名資料的存取與操作。在此案例中，對應於程式碼中的 registerForEvent 方法。該方法負責處理使用者的報名要求並進行驗證（例如活動是否額滿等），與相關實體類別（Registration）進行互動。

(4)實體類別：存放使用者及活動的相關的資訊，例如姓名、電話、電子郵件、活動時間等

- a. Registration
- b. User
- c. Event

2. 流程說明

(1)使用者與系統的互動

圖中左側代表使用者，透過點擊「報名」按鈕向系統發出報名請求。若使用者尚未登入，系統會顯示「請先登入系統」的訊息。

(2)系統處理流程

用戶的操作經由 PostMapping 傳送到 HomeController 中，在控制層中，調用了 RegisterForEvent 方法，該方法傳遞了 eventId、session、name、phone 和 email 等參數。

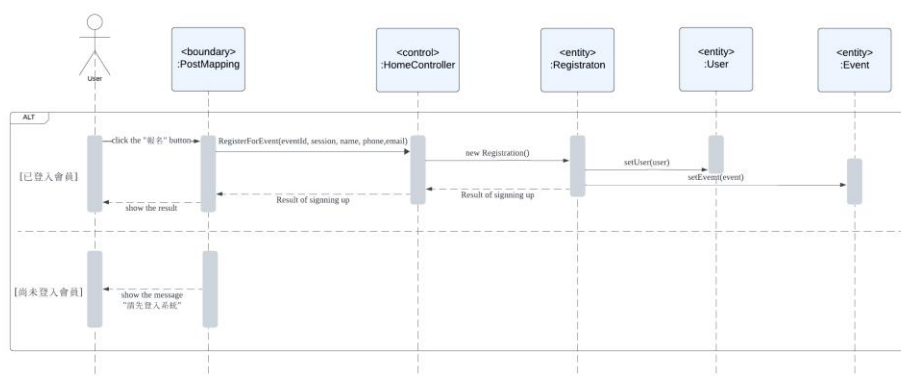
(3)實體對象的互動

HomeController 創建了一個新的 Registration 實體。接著，將用戶資料（setUser(user)）及活動資料（setEvent(event)）設置到 Registration 實體。

(4)返回結果

如果使用者已登入，系統將報名結果返回給使用者。如果用戶尚未登入，系統會提示「請先登入系統」的消息。

3. 循序圖展示



六、狀態圖(Behavior State Machine)

1. 選擇 Registration 為核心物件之原因

(1)狀態的明確性

Registration 具有清晰的狀態表示，反映了報名的不同階段，包括 "Created"（已創建）、"Confirmed"（已確認）和 "Canceled"（已取消）。每個狀態都對應特定的行為，確保系統能準確管理報名流程。

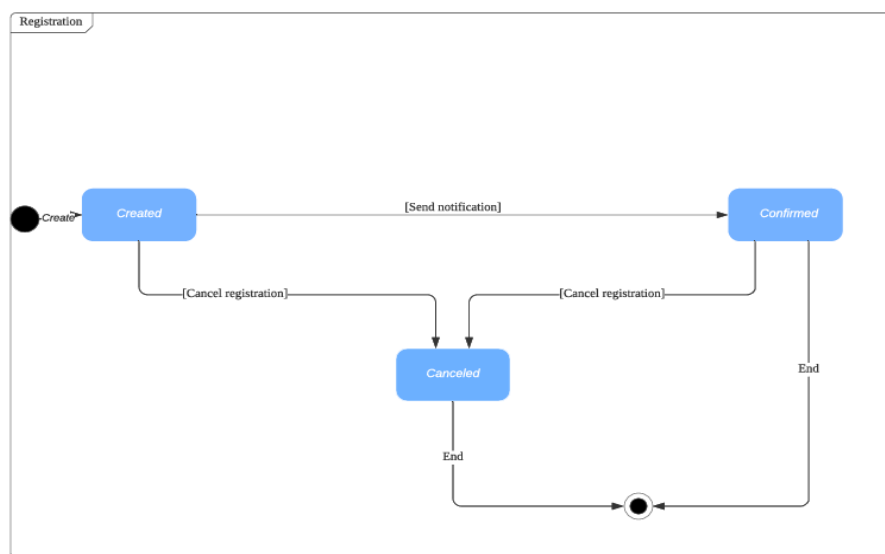
(2)狀態的可變性

Registration 狀態會根據特定事件或操作而變化，例如「使用者提交報名」會將狀態設為 Created，「確認報名」會轉為 Confirmed，而「取消報名」則將狀態更新為 Canceled，充分展示了狀態的動態變化。

(3)與其他物件的關聯性

Registration 與 User 和 Event 緊密關聯，代表使用者針對特定活動進行的報名操作。

2. 狀態圖展示



肆、組員貢獻度

楊筑安 20% 內容:程式碼編寫&設計

梁志宇 20% 內容:程式碼編寫&設計

王宥崴 20% 內容:圖形繪製&書面編輯

黃宥棠 20% 內容:圖形繪製&書面編輯

徐上哲 20% 內容:圖形繪製&書面編輯

伍、附件

第 21 組 活動報名系統 安裝說明書

1. 下載安裝 IntelliJ IDE (設定版本：Java 21)

<https://www.jetbrains.com/idea/download/>

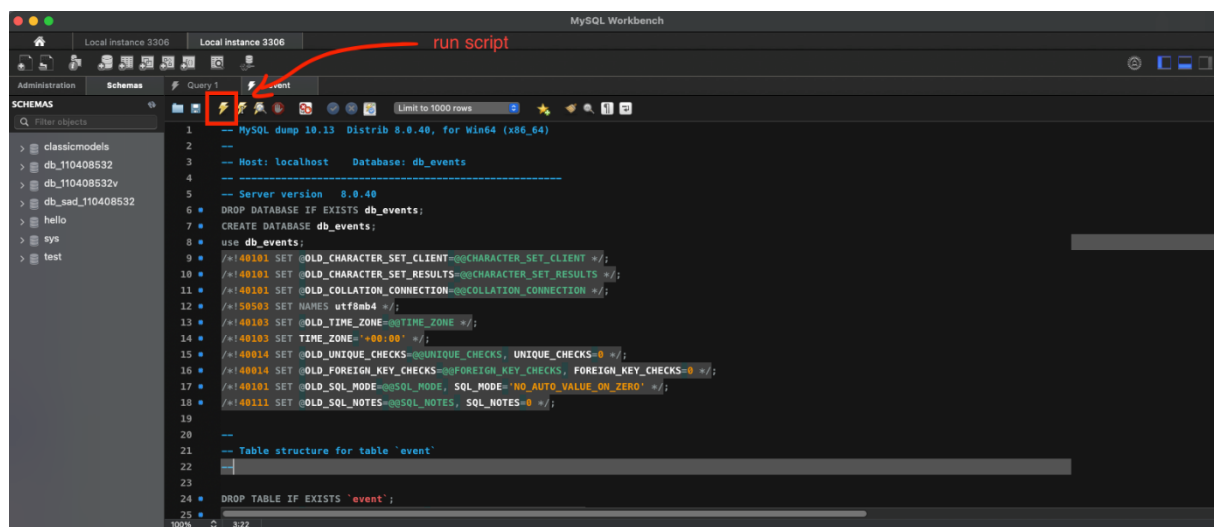
2. 下載安裝並設定資料庫(版本：MySQL 8.0.40)

MySQL: <https://www.mysql.com/downloads/>

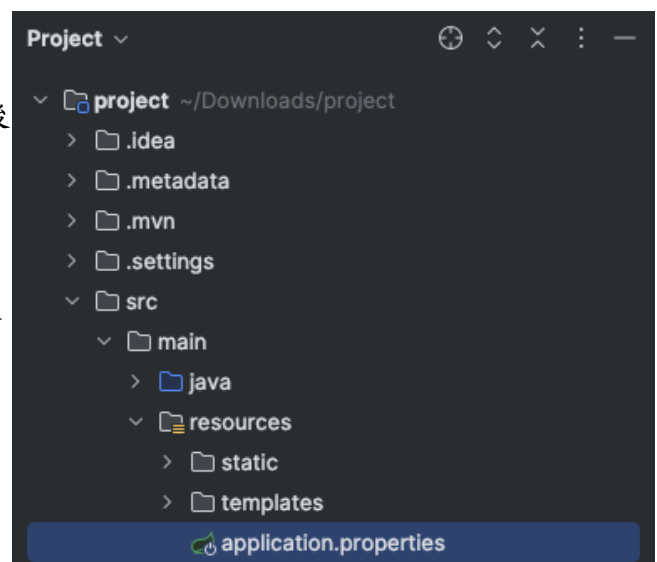
Workbench: <https://dev.mysql.com/downloads/workbench/>

3. 啟動本地端的 MYSQL server

4. 以 MySQL Workbench 打開 db_events.sql 後 run script (建立資料庫 + 匯入假資料)



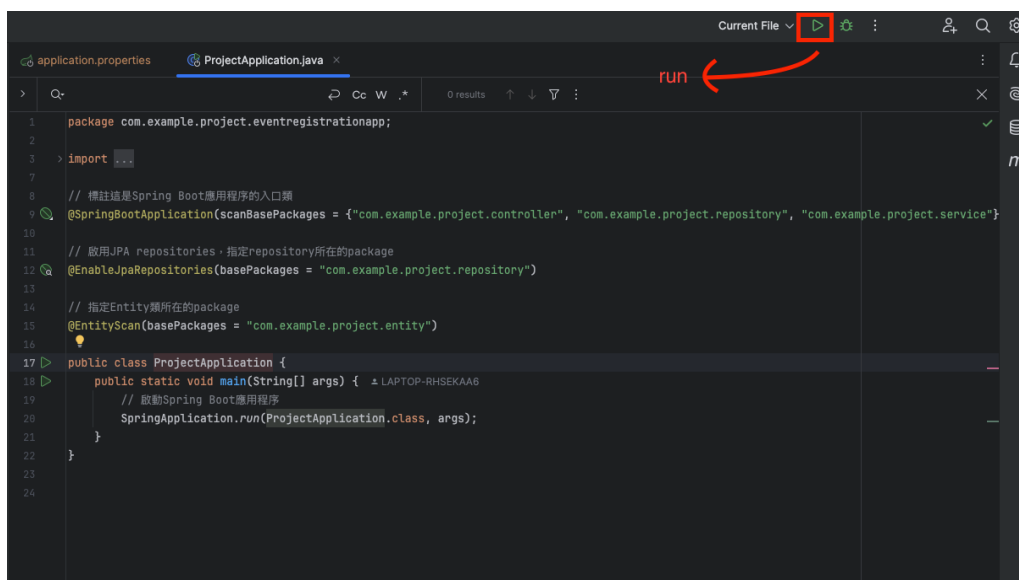
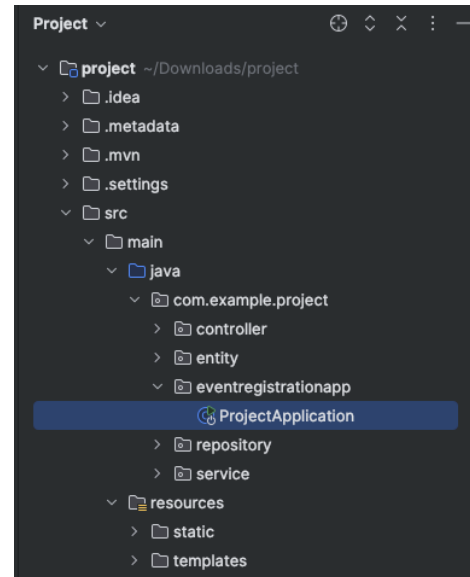
5. 以 IntelliJ 打開 project 資料夾後
打開 src 資料夾
打開 main 資料夾
打開 resources 資料夾
打開 com.example.project 資料夾
打開 application.properties



```
# spring setup database
```

```
spring.datasource.url=jdbc:mysql://localhost:3306/db_events
spring.datasource.driver-class-name=com.mysql.cj.jdbc.Driver
spring.datasource.username=您的使用者名稱
spring.datasource.password=您的密碼
```

7. 以 IntelliJ 打開 project 資料夾後
打開 src 資料夾
打開 main 資料夾
打開 java 資料夾
打開 com.example.project 資料夾
打開 eventregistrationapp 資料夾
點選 ProjectApplication 並執行(run)



8. 打開瀏覽器到 <http://localhost:8080/> 即可使用系統

