

誰敢跟我桌隊

桌球賽主辦系統

資科四 I09207440 陳楷璿 (隊長)

法律四 I10601014 黃和謙

地政四 I10207404 林佳瑤

資碩一 I13753124 呂沂瑾

財管四 I10307009 林可婷

地政碩一 I13257002 彭宗淳

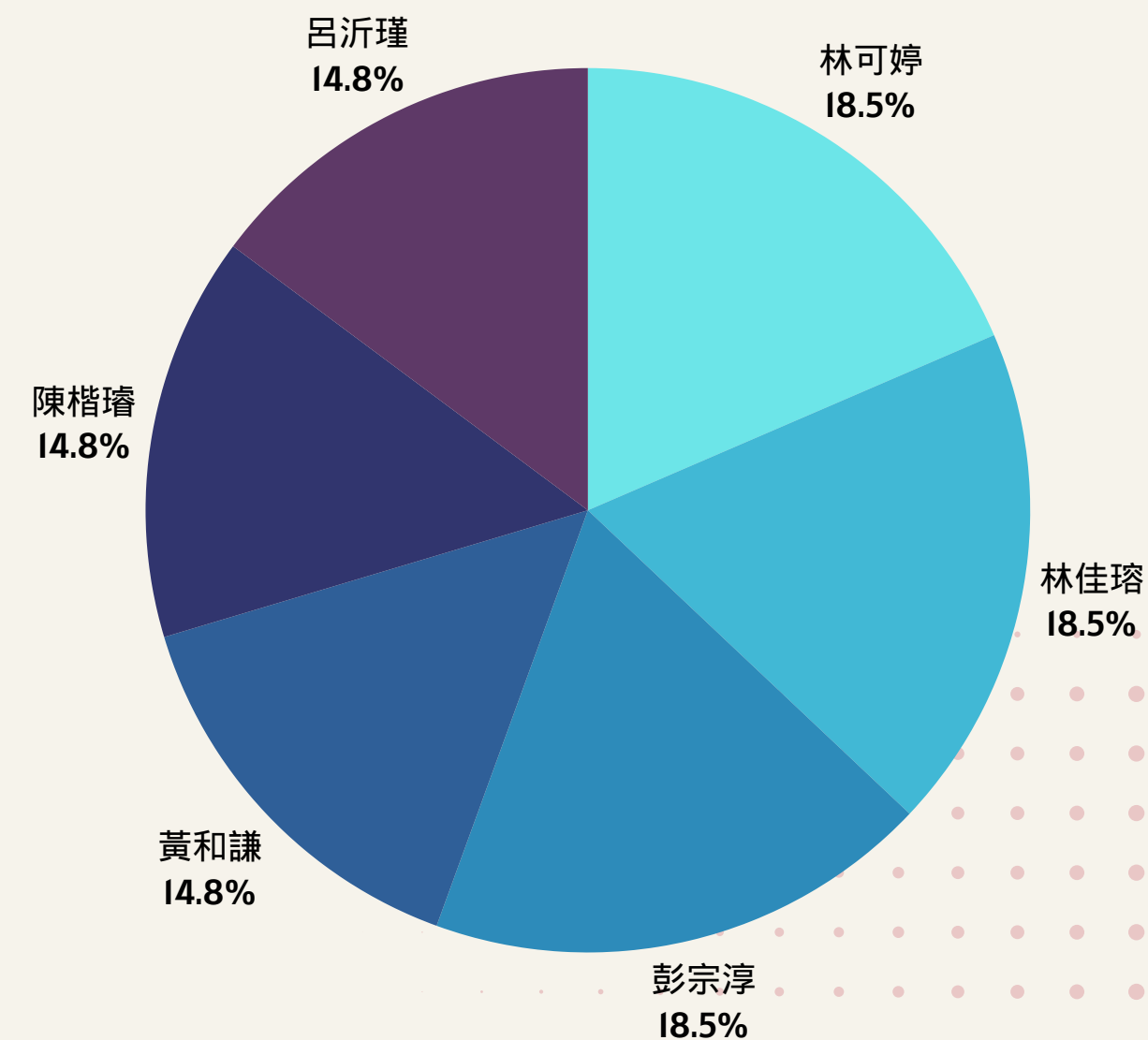
任務分工、貢獻百分比

- 前端：

- 林可婷：前後期Relational Model正規化，完成刪除報名功能和裁判報名頁面以及API串接
- 林佳瑤：前端頁面撰寫
- 彭宗淳：前端頁面撰寫

- 後端：

- 黃和謙：一般使用者、主辦方身分相關邏輯撰寫與API串接
- 陳楷璿：專案發想
- 呂沂瑾：前期ER-Model設計，ORM（Sequelize）技術引入
- 林佳瑤：帳號註冊、登入API串接
- 彭宗淳：系統報名API串接



1.需求分析

INTRODUCTION

題目發想：

過去舉辦比賽時，需要花費大量時間與人力招募裁判、安排賽程等，希望透過架設系統降低舉辦比賽時的時間與人力成本。



痛點

主辦-比賽報名

主辦方在選手、隊伍報名後，需花費時間手動抽籤安排賽程以及製作賽程表，此外亦須留意報名表單，以免超出報名上限。

選手-比賽資訊

原先的秩序冊中，要查找自己的賽程需花費大量時間瀏覽pdf檔，且因為賽程圖通常是圖片，因此無法使用文字搜尋的功能。

主辦-裁判招募

原先裁判報名後，還需進行手動安排班表，並統計用餐人數、餐食習慣。需耗費大量時間在安排班表以及處理更動，且場務組在負責統計便當時，容易因班表異動造成錯誤。

產品設計

主辦-比賽報名

透過系統完成報名程序，並在報名結束後，主辦方可透過系統自動化安排賽程，未來希望加入自動化生成賽程表的功能。

選手-比賽資訊

將安排的賽程儲存至資料庫中，並單獨使用頁面供選手查詢，降低選手花費在找尋自己賽程的時間。

主辦-裁判招募

未來將加入裁判的部分，將資訊透過資料庫維持一致性以及視覺化的方式呈現，減低工作人員在準備賽事上的不變。

2.系統功能

系統功能

1

針對比賽常見的不同角色進行功能客製化:

主辦方- 新增比賽、安排及管理賽程

使用者- 報名各種比賽、查詢賽程

2

作為管理員可以更動人別身分，隨時增、減主辦方用戶

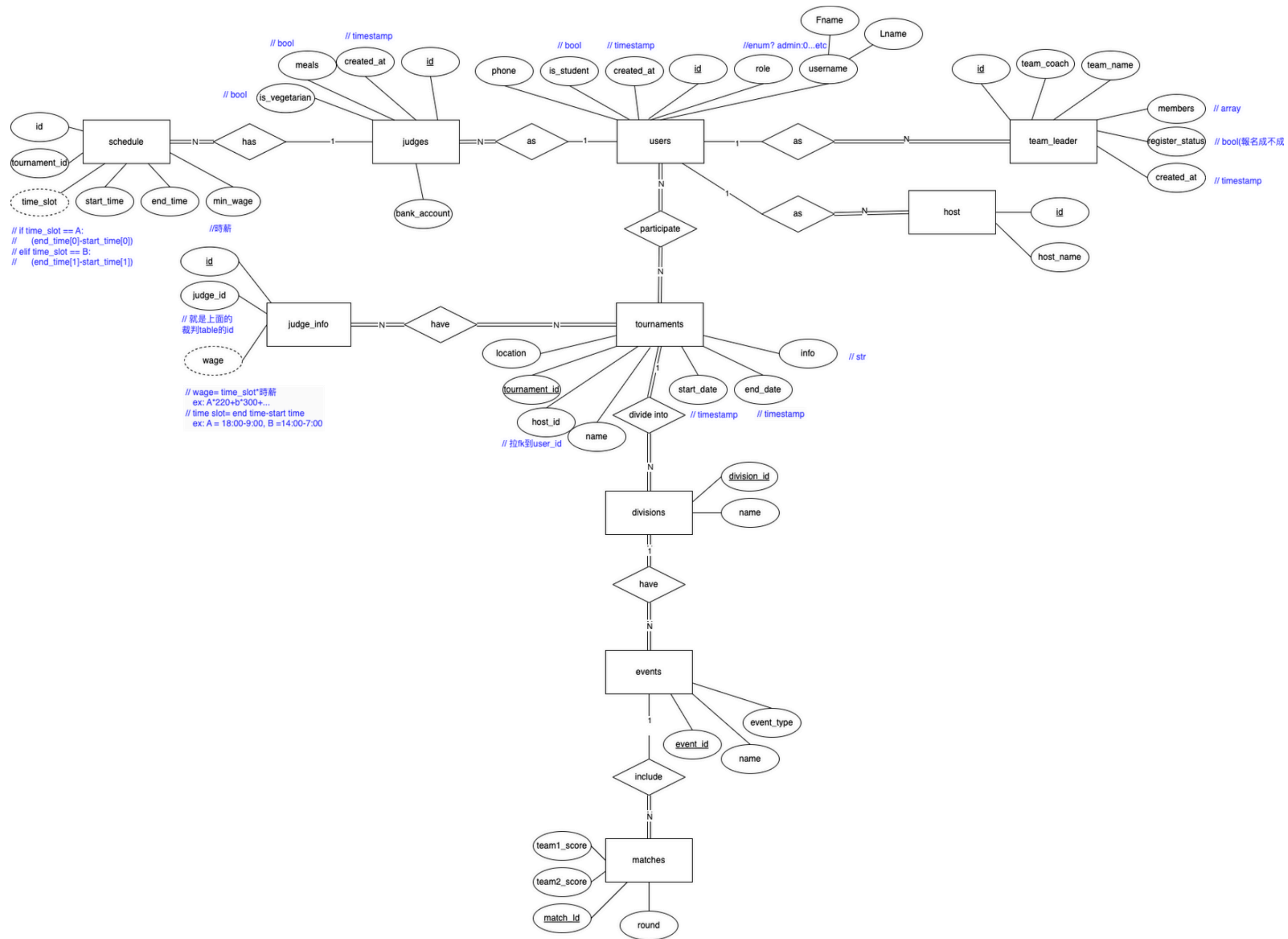
同時，想要成為主辦方的人也可以透過系統申請辦理比賽

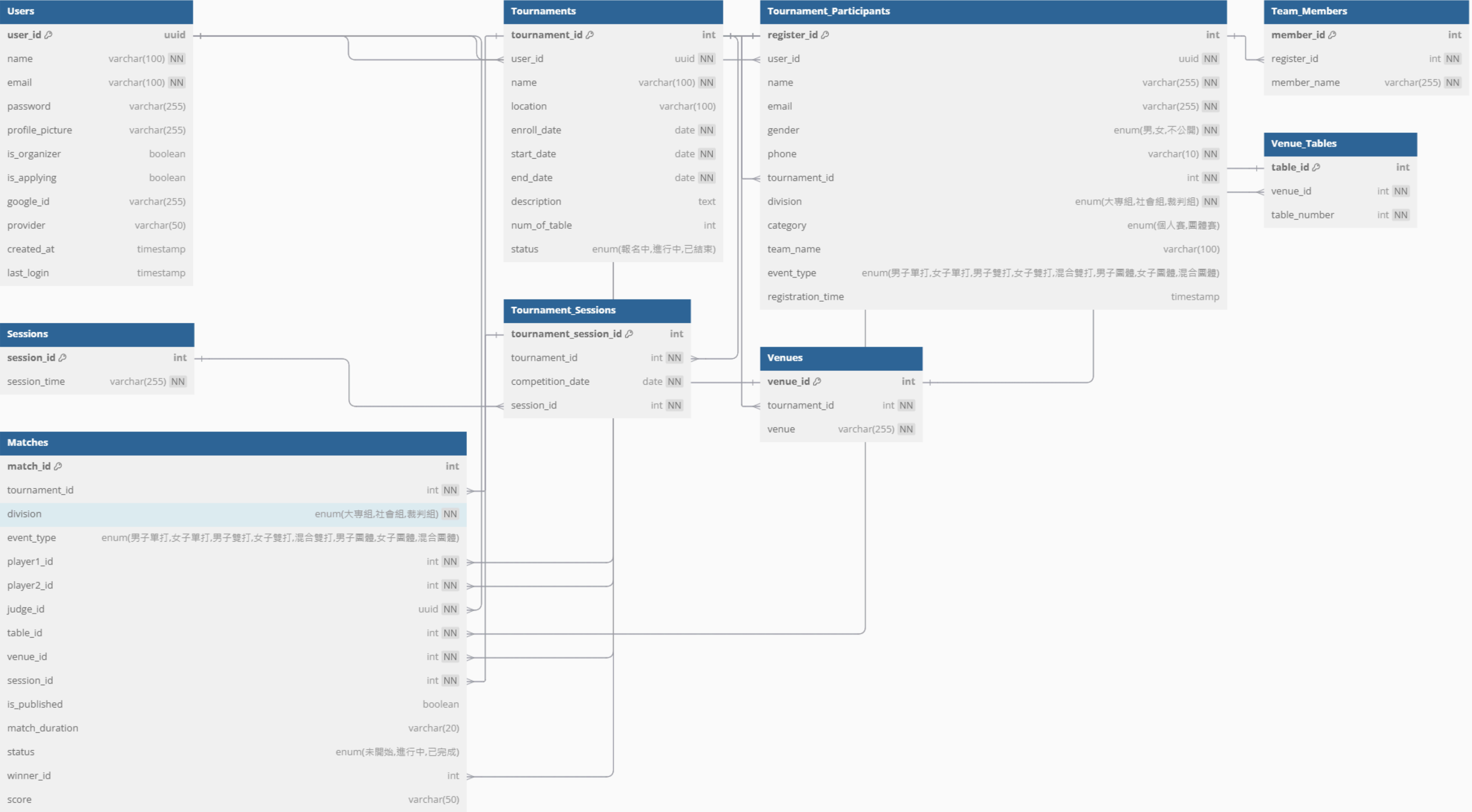
3

比賽的日期、地點、桌數、賽程表都可以客製化定義

也可以因應個人賽及團體賽進行調整

3.ER Model





4.Relational Schema

schedule						
<u>schedule_id</u>	start_time	end_time	judges_id	tournament_id		
timeslot						
<u>end_time</u>	start_time	time_slot				
minwage						
<u>tournament_id</u>	min_wage					
judges						
<u>judges_id</u>	meals	judges_created_at	is_vegetarian	bank_account	user_id	
users						
<u>users_id</u>	is_student	users_created_at	phone	role	Fname	Lname
host						
<u>host_id</u>	host_name	user_id				
team_leader						
<u>team_leader_id</u>	team_coach	team_name	members	register_status	team_leader_created_at	user_id
tournament						
<u>tournament_id</u>	location	info	tournament_name	start_date	end_date	host_id
divisions						
<u>division_id</u>	divisions_name	tournament_id				
events						
<u>event_id</u>	events_name	event_type	division_id			
matches						
<u>match_id</u>	team1_score	team2_score	round			
judges_info						
<u>judges_info_id</u>	judge_id	wage				
judges_info_tournaments						
<u>judge_info_id</u>	<u>tournament_id</u>					
events_matches						
<u>event_id</u>	<u>match_id</u>					
users_tournaments						
<u>users_id</u>	<u>tournament_id</u>					

資料庫TABLE呈現

Users

user_id (PK), name, email, password, profile_picture, is_organizer, is_applying, google_id, provider, created_at, last_login

Tournaments

tournament_id (PK), user_id (FK), name, location, enroll_date, start_date, end_date, description, num_of_table, status

Tournament_Participants

register_id (PK), user_id (FK), tournament_id (FK), name, email, gender, phone, division, category, team_name, event_type, registration_time

Team_Members

member_id (PK), register_id (FK), member_name

資料庫TABLE呈現

Sessions

session_id (PK), session_time

Tournament_Sessions

tournament_session_id (PK), tournament_id (FK), session_id (FK), competition_date

Venues

venue_id (PK), tournament_id (FK), venue

Venue_Tables

table_id (PK), venue_id (FK), table_number

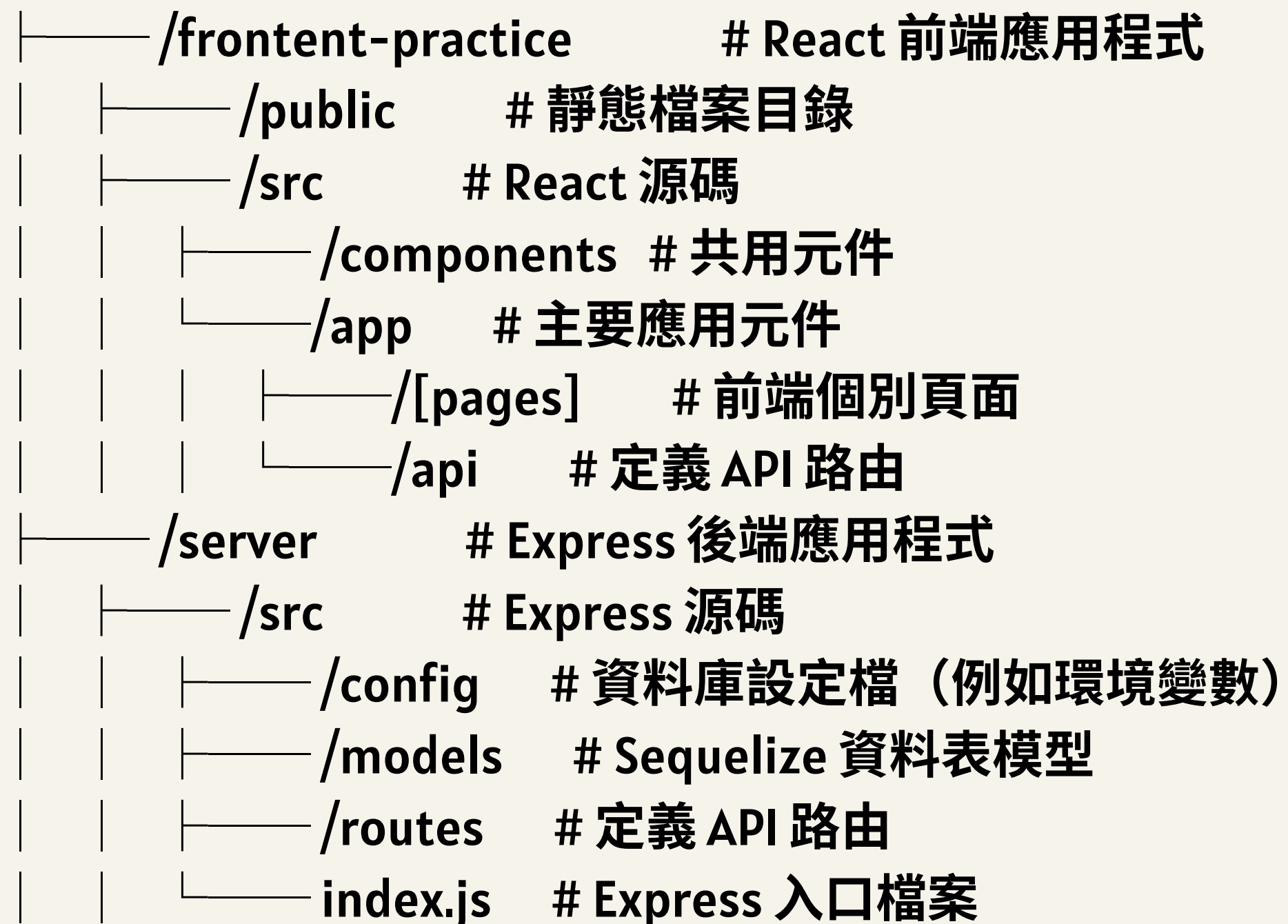
Matches

match_id (PK), tournament_id (FK), player1_id (FK), player2_id (FK), judge_id (FK), table_id (FK), venue_id (FK), session_id (FK), division, event_type, is_published, match_duration, status, winner_id (FK), score

5.系統架構

系統架構

/NCCUTT



6.DEMO

7.心得、收穫與建議

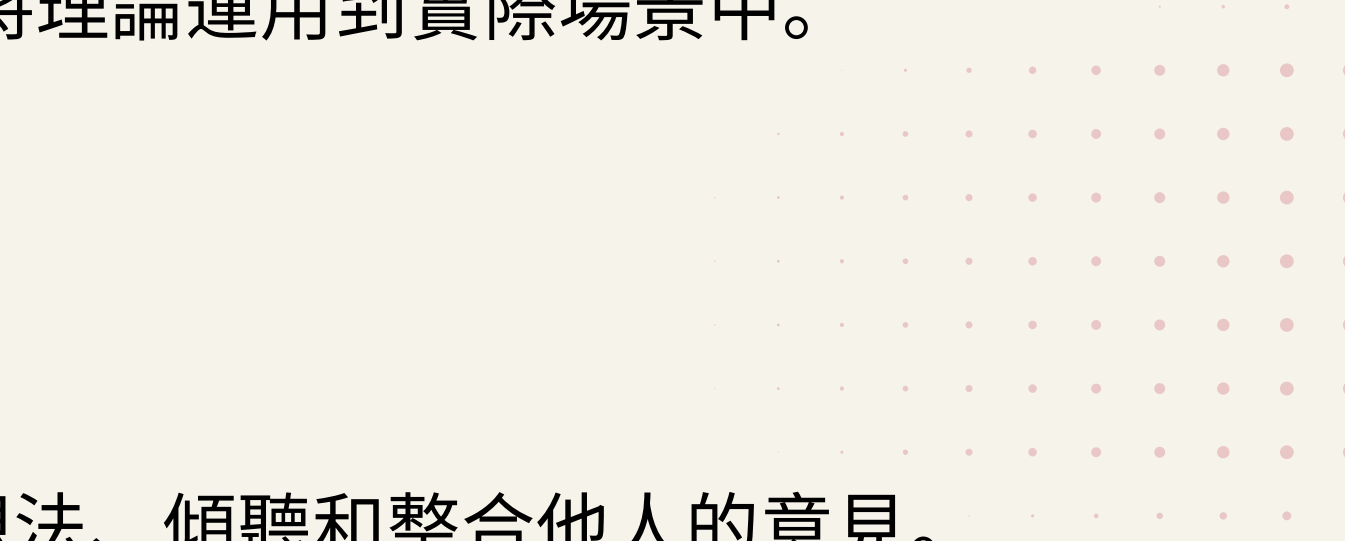


心得收穫

這次專案讓我們深刻體會到，前期的討論和準備真的非常重要。大家在一開始就需要充分了解專案的目的和實作方向，才能確保後續的開發過程更加順利，也能避免因資訊不對等導致的反覆修改。在設計頁面與資料庫時，我們也學到了前端和後端的溝通合作是不可或缺的環節。透過雙方的充分討論，我們得以確定流程與需求，讓整體開發工作更有條理且高效。

這次實作不僅讓我們對系統設計有了更清楚的概念，也讓我們有機會將理論運用到實際場景中。

幾個主要的收穫包括：

1. 前端技術：學會了架設前端頁面，並了解 **API** 的撰寫與測試流程。
 2. 後端實作：對資料庫的結構設計與資料管理有了更深入的認識。
 3. 團隊合作與溝通：透過持續的討論與協作，學習有效表達自己的想法、傾聽和整合他人的意見。
- 

總結來說，這次專案不僅讓我們在技術上更加熟練，也讓我們理解到溝通與協作對專案成功的重要性，這些經驗將會是我們未來工作的寶貴基礎。

The background features three vertical stripes on the left: a wide pink stripe, a medium blue stripe, and a narrow beige stripe. The right side of the image is a light beige background with two rectangular areas of small, light pink dots in the top right and bottom right corners.

THANK YOU