

選課資訊整合系統 – 書面報告

Course Integrated Information System – Formal Report

DBMS Group 14

政治二。109202007。余則緯

經濟二。109207344。王行遠

國貿二。109301095。林竺徵

國貿一。110301213。沈沛恩

風管二。109308028。蔡念澄

資科二。109703005。童昱維

目錄

更改專案方向前的版本：介紹（Introduction）	3
更改專案方向前的版本：先備知識（Requirements）	5
更改專案方向前的版本：需求分析（Need Analysis）	5
更改專案方向的契機：瓶頸（Bottleneck）	8
更改專案方向後的版本：介紹（introduction）	8
更改專案方向後的版本：實體關聯圖（ERD diagram）	9
更改專案方向後的版本：轉換成為關聯式資料庫（ERD-to-relation）	10
關聯式資料庫	10
Candidate key 說明	11
Referential Integrity Constraints 宣告	11
功能相依	12
更改專案方向後的版本：需證明每一張資料表均已達 BCNF（Normalization）	13
User	13
Order_Like	14
Comment_Like	15
Orders	16
Comment	17
Course	18
Teacher	19
Department	20
更改專案方向後的版本：系統實作 DFD（Data Flow Diagram）	21

更改專案方向前的版本：介紹（Introduction）

Our team project has been imagined from a simple issue for students: the influence of the selection system on the courses student's choice. Indeed, as a student we know how difficult it can be to be selected for courses you really want in your schedule time for the semester. Moreover, it can be difficult to assess the level of the demand for one course. Is it a very popular course? Are other students presenting quality academic records for this course compared to me? Do they take this course as a part of their major?

All of these questions remain unanswered if you do not have the opportunity or time to interview a large number of current students based on their profiles.

From this observation, we thought about how we could improve this situation of uncertainty for students. We thought that one way to save time and get out the questioning in their mind could be to have some indication of the likelihood of being enrolled in a specific course before applying. In addition, that we found it useful since every course has its own rule for students.

Therefore, our project will consist of a system where **students could see their probabilities of being selected for a given course**. These probabilities will be given for courses from the university thanks to a calculation taking into account several criteria of the student. For instance, students who double major a subject may have higher probabilities to be selected on a required subject than minor major students. Finally, to increase the accuracy of this strategic data for students, we will ask students to rank courses they plan to choose based on their preferences.

As a result of this and after having answered some questions, the student will be able to determine his/her level compared to other students and therefore his/her chances of being enrolled in a class. This new tool would help students make better choices (for example: which course should be put first or second...) for a given list of courses.

Our system will be called the 'recommendation system' and

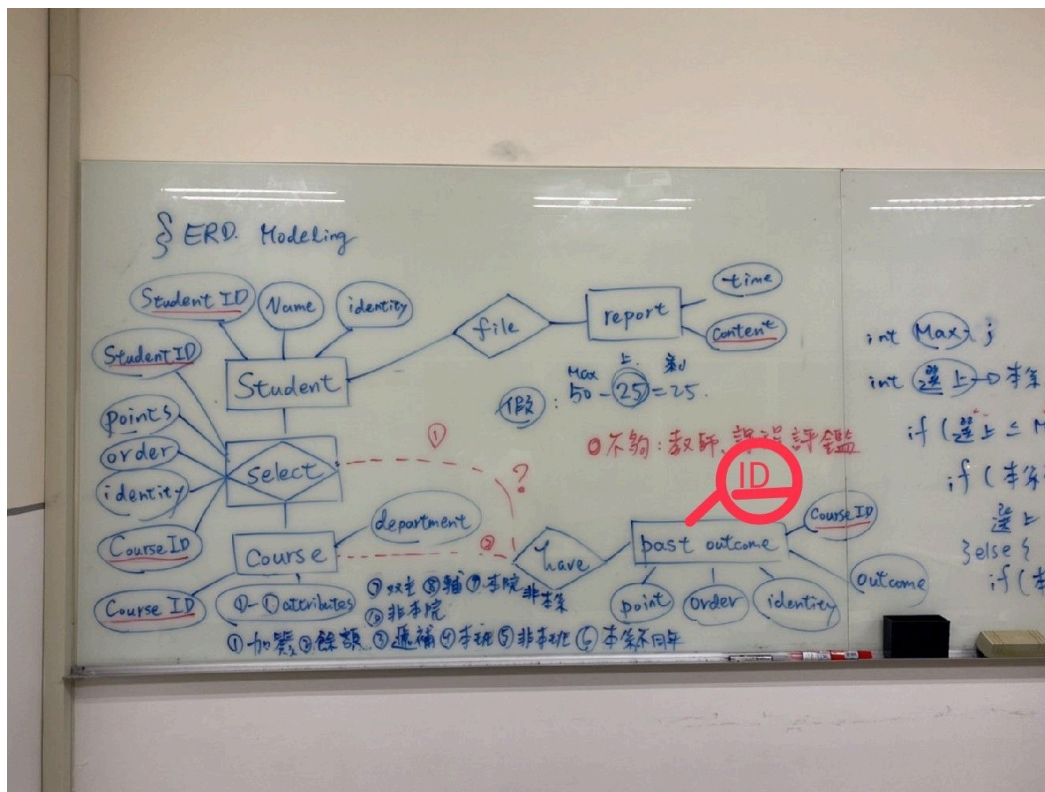
(1) the users provide his/her html file to the course selection system. or

(2) when the users come into the courses selection website, our system will automatically crawl the html file of the users' selection results and orders.

Finally, to enable the creation of this project, we will need a database that will store course and student information such as selecting requirements of the course, student grades, student ID, major / minor subjects, etc. We'll also have a list to record the result of course selection in the past few semesters, in order to provide a reference for system users.

- a database that stores course information

- draw an ERD diagram (It may be adjusted...)



更改專案方向前的版本：先備知識 (Requirements)

- **acquire needed data from school course selecting system**
 - **may use a web crawler**
 - **Java / Python as main programming language**

- **a calculator of the probabilities**

更改專案方向前的版本：需求分析 (Need Analysis)

- **We interviewed two NCCU students, and asked them the questions below:**
 - **In your opinion, is it time-consuming to decide your rank of courses?**

Student A: Absolutely Yes. Every time before the formal selecting courses, I always spend about a half of month to a month to search every course's syllabus, restricting, commenting, and also, I need to put it in the schedule in advance to prevent time conflicts between two courses. What's more, I must rank each one. Ahhh, that is such a problem.

- **Do you approve of the extra extension tool to help you select courses from the courses selecting system?**

Student B: Of course. I would like to use it. I just want to select the courses I want to select successfully, and I totally don't care about the mechanism behind the system. Hence, a recommendation tool is really useful for me.

- **Is it convenient and accurate to search information about courses selection from Dcard?**

Student A: In my opinion, it is convenient to get course selection information directly on Dcard. However, the information can be inaccurate sometimes, since everyone differs in their major subject, course selecting points and other conditions, which may vary the result among students.

- **How long did you prepare for selecting courses? (including searching, ranking etc.)**

Student B: About 20 hours actually.

- **How much information would you provide us to improve the system?**

Student A: I think all the information in the course selecting system are not confidential, so if you won't leak that information, I'd like to give you all the information in the system.

- **How much would you be willing to pay for our service?**

Student B: I think \$30 would be a fair price. But considering that you guys are a start-up, I will invest more to help you develop the system.

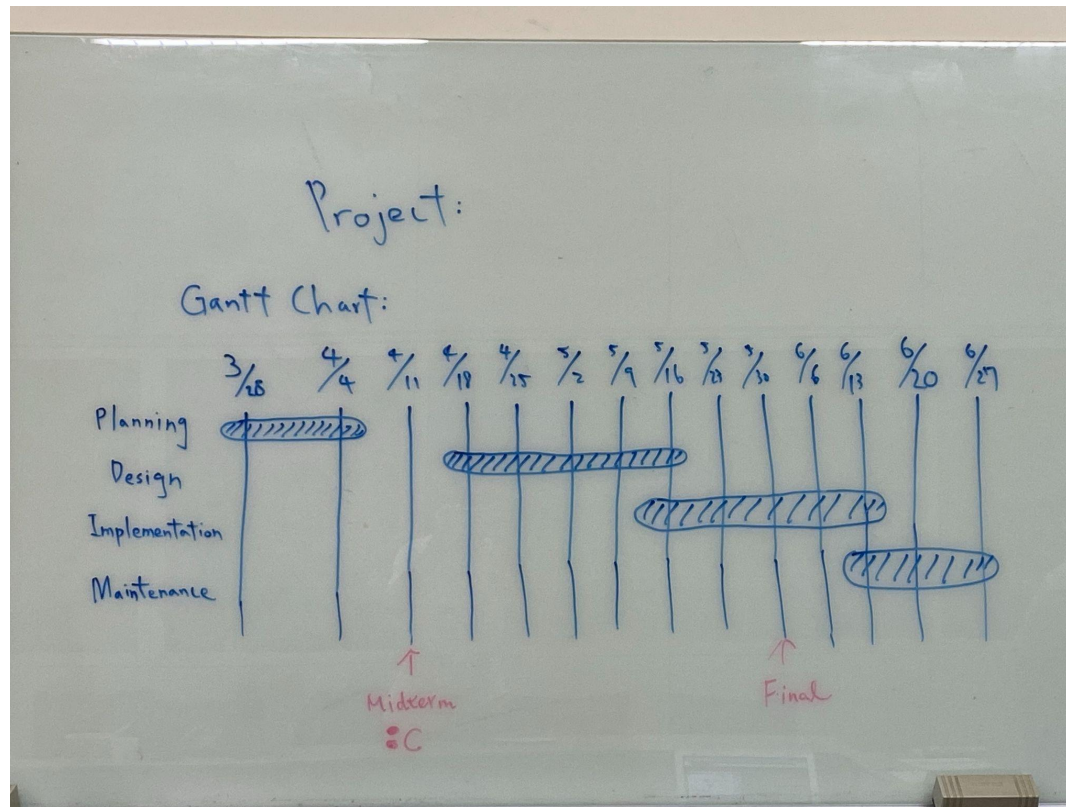
- **Economic feasibility analysis**

- The file has been attached in the same compressed folder.



- **Preliminary schedule**

- (Using Gantt chart / PERT chart)



更改專案方向的契機：瓶頸（Bottleneck）

在與助教們進行第二次討論過後，基於原本專案的想法：選課推薦系統，我們必須使用到機率和深度學習相關的概念才能完成專案，當時助教推薦給我們的方法是使用隨機森林（random forest）來預測多個變因下事件的預期結果。然而，使用這個方法的前提是需要大量過去的選課歷史資料，這個預測模型才有辦法實作。因此，我們在第二次助教討論時間時，我們已經有準備無法獲取大量過去歷史資料時的備案。

在知道我們的需求後，我們馬上向學校詢問。在與電算中心、校務研究辦公室、教務處交涉過後，因為我們並非研究生論文撰寫，亦非教師研究計畫，故必須向學校提出專案申請。專案申請需歷時數個月和繳交五千元的行政審查費用，可能在學期結束後，審查結果仍不知下落，當時即發現此方法不可行。再次與助教進行討論後，我們決定施行我們當初在第二次助教討論時間所提出的備案，即為選課資訊整合系統。在這個系統中，我們會保持著當初做選課推薦系統的初衷，為學生提供更便利的方式去選課來設計系統。

更改專案方向後的版本：介紹（introduction）

我們的系統是從一個簡單的問題為發想的：如何讓選課的過程更為輕鬆。事實上，作為一名學生，我們知道在學期的安排時間內選擇你真正想要的課程是多麼困難。此外，很難評估一門課程的水平。這是一門非常受歡迎的課程嗎？我的能力能負荷這堂課嗎？我該如何排序課的順位已得到最好的選課結果？對於這些問題，大多數學生的解決辦法是透過 Dcard、PTT 等社群網站上面的分享。然而，這些網站上的資訊非常雜亂，且分散在各個網站，使得資料蒐集非常累人，進而讓選課變成一件既麻煩又消耗時間的工作。

因此，我們決定做一個“選課資訊整理系統”，讓學生們可以有一個統一的地方回報及查詢這些過往的排課修課經驗。使用者會按照我們的格式分享經驗，在排課次序方面，使用者需要回報他對於這堂課的身分別及他在選課時的排序；在修課經驗方面，使用者會回報甜度和涼度這兩個指標性的評分以及留言。除了過往的經驗，我們也將一堂課的所有資訊(像是學生人數, 可否加退選)一同整理起來，再藉由系所、教師等去將課程分類，透過資料庫的編排及 GUI 的操作，讓選課的學生能夠輕易地找到感興趣的課及其相關的資訊。

更改專案方向後的版本：轉換成為關聯式資料庫（ERD-to-relation）

關聯式資料庫

註：Primary key 以底線標示、Foreign key 以紅色標示

- User (User_ID, Password, Name, Authority, Gender, Department_ID, Total_Violate(違規次數), StopUse(停權), ImageType(大頭貼))
- Order_Like ((User_ID, Order_ID))
- Comment_Like ((User_ID, Comment_ID))
- Orders (Order_ID, Identity, Is_Enrolled, Num, Is_Deleted, Violate(違規次數), Course_ID, User_ID)
- Comment (Comment_ID, Violate(違規次數), Content, Cold(涼度), Sweet(甜度), Is_Deleted, Course_ID, User_ID)
- Course (Course_ID, Name, 可遞補, 先修科目, 可加選, 可退選, 低年級選修, 必選修, 本系本班, 本系同年級非本班, 本系不同年級非本班, 全系, 雙主修, 輔系, 本院非本系, 非本院, Time(開課時間), 全校總人數, Teacher_ID, Department_ID)
- Teacher (Teacher_ID, Department_ID)
- Department (Department_ID, Name)

Candidate key 說明

各表格之 candidate key 整理如下表，如只有單一候選鍵(無任何 Secondary key)，則不於下表列示：

表格	Candidate key
Department	Name, Department_ID

Referential Integrity Constraints 宣告

- $\text{Order_Like (User_ID)} \subseteq \text{User (User_ID)}$
- $\text{Comment_Like (User_ID)} \subseteq \text{User (User_ID)}$
- $\text{Order_Like (Order_ID)} \subseteq \text{Orders (Order_ID)}$
- $\text{Comment_Like (Comment_ID)} \subseteq \text{Comment (Comment_ID)}$
- $\text{Orders (User_ID)} \subseteq \text{User (User_ID)}$
- $\text{Orders (Course_ID)} \subseteq \text{Course (Course_ID)}$
- $\text{Comment (User_ID)} \subseteq \text{User (User_ID)}$
- $\text{Comment (Course_ID)} \subseteq \text{Course (Course_ID)}$
- $\text{Course (Teacher_ID)} \subseteq \text{Teacher (Teacher_ID)}$
- $\text{Course (Department_ID)} \subseteq \text{Department (Department_ID)}$
- $\text{Teacher (Department_ID)} \subseteq \text{Department (Department_ID)}$

功能相依

各表格所須遵循之功能相依說明如下表：

表格	功能相依
User	User_ID→{Password, Name, Authority, Gender, Department_ID, Total_Violate(違規次數), StopUse(停權), ImageType(大頭貼)}
Order_Like	
Comment_Like	
Orders	Order_ID→{Identity, Is_Enrolled, Num, Is_Deleted, Violate(違規次數), Course_ID, User_ID}
Comment	Comment_ID→{Violate(違規次數), Content, Cold(涼度), Sweet(甜度), Is_Deleted, Course_ID, User_ID}
Course	Course_ID→{Name, 可遞補, 先修科目, 可加選, 可退選, 低年級選修, 必選修, 本系本班, 本系同年級非本班, 本系不同年級非本班, 全系, 雙主修, 輔系, 本院非本系, 非本院, Time(開課時間), 全校總人數, Teacher_ID, Department_ID}
Teacher	Teacher_ID→{Department_ID}
Department	Department_ID→{Name} Name→{Department_ID}

更改專案方向後的版本：需證明每一張資料表均已達 BCNF (Normalization)

User

1NF

1. 每個欄位都是單一值，沒有多值
2. 沒有任何兩筆以上的資料是重複
3. 沒有複合屬性
4. 沒有巢狀關係

→符合 1NF

2NF

Closure : $\{User_ID\}^+ = \{User_ID, Password, Name, Authority, Gender, Department_ID, Total_Violate(違規次數), StopUse(停權), ImageType(大頭貼)\} = \mathbf{User}$

Candidate key : User_ID

Prime attribute : User_ID

Non-prime attribute : Password, Name, Authority, Gender, Department_ID, Total_Violate(違規次數), StopUse(停權), ImageType(大頭貼)

並無任一功能相依有「key 的一部分可單獨決定其他屬性」的情況，故不存在部分功能相依

→符合 2NF

3NF

∵ 沒有 non-prime attribute 可決定其他 non-prime attribute，故不存在遞移相依

→符合 3NF

BCNF

∵ User_ID 為 Super key

∴ 不違反主屬性必須依賴於不包含它的候選鍵

→符合 BCNF

Order_Like

1NF

1. 每個欄位都是單一值，沒有多值
2. 沒有任何兩筆以上的資料是重複
3. 沒有複合屬性
4. 沒有巢狀關係

→符合 1NF

2NF

Closure : $\{(User_ID, Order_ID)\}^+ = \{(User_ID, Order_ID)\} = \mathbf{Order_Like}$

Candidate key : (User_ID, Order_ID)

Prime attribute : (User_ID, Order_ID)

Non-prime attribute : 無

並無任一功能相依有「key 的一部分可單獨決定其他屬性」的情況，故不存在部分功能相依

→符合 2NF

3NF

∵ 沒有 non-prime attribute 可決定其他 non-prime attribute，故不存在遞移相依

→符合 3NF

BCNF

∵ (User_ID, Order_ID) 為 Super key

∴ 不違反主屬性必須依賴於不包含它的候選鍵

→符合 BCNF

Comment_Like

1NF

1. 每個欄位都是單一值，沒有多值
2. 沒有任何兩筆以上的資料是重複
3. 沒有複合屬性
4. 沒有巢狀關係

→符合 1NF

2NF

Closure : $\{(User_ID, Comment_ID)\}^+ = \{(User_ID, Comment_ID)\} = \mathbf{Comment_Like}$

Candidate key : (User_ID, Comment_ID)

Prime attribute : (User_ID, Comment_ID)

Non-prime attribute : 無

並無任一功能相依有「key 的一部分可單獨決定其他屬性」的情況，故不存在部分功能相依

→符合 2NF

3NF

∵ 沒有 non-prime attribute 可決定其他 non-prime attribute，故不存在遞移相依

→符合 3NF

BCNF

∵ (User_ID, Comment_ID) 為 Super key

∴ 不違反主屬性必須依賴於不包含它的候選鍵

→符合 BCNF

Orders

1NF

1. 每個欄位都是單一值，沒有多值
2. 沒有任何兩筆以上的資料是重複
3. 沒有複合屬性
4. 沒有巢狀關係

→符合 1NF

2NF

Closure : $\{Order_ID\}^+ = \{Order_ID, Identity, Is_Enrolled, Num, Is_Deleted, Violate(\text{違規次數}), Course_ID, User_ID\} = \mathbf{Orders}$

Candidate key : Order_ID

Prime attribute : Order_ID

Non-prime attribute : Identity, Is_Enrolled, Num, Is_Deleted, Violate(違規次數), Course_ID, User_ID

並無任一功能相依有「key 的一部分可單獨決定其他屬性」的情況，故不存在部分功能相依

→符合 2NF

3NF

∵ 沒有 non-prime attribute 可決定其他 non-prime attribute，故不存在遞移相依

→符合 3NF

BCNF

∵ Order_ID 為 Super key

∴ 不違反主屬性必須依賴於不包含它的候選鍵

→符合 BCNF

Comment

1NF

1. 每個欄位都是單一值，沒有多值
2. 沒有任何兩筆以上的資料是重複
3. 沒有複合屬性
4. 沒有巢狀關係

→符合 1NF

2NF

Closure : $\{Comment_ID\}^+ = \{Comment_ID, Violate(違規次數), Content, Cold(涼度), Sweet(甜度), Is_Deleted, Course_ID, User_ID\} = \mathbf{Comment}$

Candidate key : Comment_ID

Prime attribute : Comment_ID

Non-prime attribute : Violate(違規次數), Content, Cold(涼度), Sweet(甜度), Is_Deleted, Course_ID, User_ID

並無任一功能相依有「key 的一部分可單獨決定其他屬性」的情況，故不存在部分功能相依

→符合 2NF

3NF

∵ 沒有 non-prime attribute 可決定其他 non-prime attribute，故不存在遞移相依

→符合 3NF

BCNF

∵ Comment_ID 為 Super key

∴ 不違反主屬性必須依賴於不包含它的候選鍵

→符合 BCNF

Course

1NF

1. 每個欄位都是單一值，沒有多值
2. 沒有任何兩筆以上的資料是重複
3. 沒有複合屬性
4. 沒有巢狀關係

→符合 1NF

2NF

Closure : $\{Course_ID\}^+ = \{Course_ID, Name, \text{可遞補}, \text{先修科目}, \text{可加選}, \text{可退選}, \text{低年級選修}, \text{必修}, \text{本系本班}, \text{本系同年級非本班}, \text{本系不同年級非本班}, \text{全系}, \text{雙主修}, \text{輔系}, \text{本院非本系}, \text{非本院}, \text{Time(開課時間)}, \text{全校總人數}, \text{Teacher_ID}, \text{Department_ID}\} = \mathbf{Course}$

Candidate key : Course_ID, Prime attribute : Course_ID

Non-prime attribute : Name, 可遞補, 先修科目, 可加選, 可退選, 低年級選修, 必修, 本系本班, 本系同年級非本班, 本系不同年級非本班, 全系, 雙主修, 輔系, 本院非本系, 非本院, Time(開課時間), 全校總人數, Teacher_ID, Department_ID

並無任一功能相依有「key 的一部分可單獨決定其他屬性」的情況，故不存在部分功能相依

→符合 2NF

3NF

∵ 沒有 non-prime attribute 可決定其他 non-prime attribute，故不存在遞移相依

→符合 3NF

BCNF

∵ Course_ID 均為 Super key

∴ 不違反主屬性必須依賴於不包含它的候選鍵

→符合 BCNF

Teacher

1NF

1. 每個欄位都是單一值，沒有多值
2. 沒有任何兩筆以上的資料是重複
3. 沒有複合屬性
4. 沒有巢狀關係

→符合 1NF

2NF

Closure : $\{\text{Teacher_ID}\}^+ = \{\text{Teacher_ID}, \text{Department_ID}\} = \mathbf{Teacher}$

Candidate key : Teacher_ID

Prime attribute : Teacher_ID

Non-prime attribute : Department_ID

並無任一功能相依有「key 的一部分可單獨決定其他屬性」的情況，故不存在部分功能相依

→符合 2NF

3NF

∵ 沒有 non-prime attribute 可決定其他 non-prime attribute，故不存在遞移相依

→符合 3NF

BCNF

∵ Teacher_ID 為 Super key

∴ 不違反主屬性必須依賴於不包含它的候選鍵

→符合 BCNF

Department

1NF

1. 每個欄位都是單一值，沒有多值
2. 沒有任何兩筆以上的資料是重複
3. 沒有複合屬性
4. 沒有巢狀關係

→符合 1NF

2NF

Closure : $\{\text{Department_ID}\}^+ = \{\text{Department_ID}, \text{Name}\} = \mathbf{\text{Department}}$

$\{\text{Name}\}^+ = \{\text{Name}, \text{Department_ID}\} = \mathbf{\text{Department}}$

Candidate key : Department_ID, Name

Prime attribute : Department_ID, Name

Non-prime attribute : 無

並無任一功能相依有「key 的一部分可單獨決定其他屬性」的情況，故不存在部分功能相依

→符合 2NF

3NF

∵ 沒有 non-prime attribute 可決定其他 non-prime attribute，故不存在遞移相依

→符合 3NF

BCNF

∵ Department_ID, Name, {Department_ID, Name} 為 Super key

∴ 不違反主屬性必須依賴於不包含它的候選鍵

→符合 BCNF

更改專案方向後的版本：系統實作 DFD (Data Flow Diagram)

