第十組資料庫期末報告

專題進度追蹤系統

四資工四乙 40943212陳廷威

指導老師

江季翰 教授

## 心得

這次課程讓我學習到幾件事，第一是如果可以選擇組員的話務必慎重考慮對方的個性與人品，第二是當遇到組員擺爛的時候，我必須要做好自己完成的準備，並在眾多課程之中抽出時間來完成這門課的專題，因為我很想畢業也很想要這門課的學分，所以雖然好幾次都自己熬夜完成作業，也只能咬者牙努力撐過去，因為也有其他必修的時間安排，也讓我算是學會了時間規畫這門課題，盡可能在有限的時間做出最好的安排。也算是學習到了一課

## ✨ 題目定義

專題進度追蹤系統是一個多用戶網頁應用平台，旨在幫助教師有效地追蹤和記錄學生的專題進展。系統支持三種用戶角色（學生、教師和管理員），提供專題管理、進度追蹤、評分回饋和記錄查看等功能。

## 👥 應用情境與使用案例

### 情境一：學生週報繳交與進度追蹤

**參與者**：資工系大三學生

**情境描述**： 某資工系大三學生，正在進行畢業專題。每週五是他們小組繳交週報的截止日。今天是週四晚上，學生需要提交本週的進度報告。

**使用案例**： 1. 學生登入系統進入「進度繳交」頁面 2. 系統顯示表單，包含「本週進度報告」與「下週工作計畫」欄位 3. 學生填寫本週已完成的程式模組測試結果，並上傳測試程式碼檔案 4. 學生填寫下週將進行系統整合測試的計畫 5. 學生提交週報，系統記錄提交時間 6. 系統通知學生週報提交成功，可在「瀏覽模式」查看 7. 學生進入「瀏覽模式」，透過日曆界面查看本週的提交記錄 8. 學生查看上週填寫的「下週工作計畫」與本週實際完成的工作，進行自我檢視

### 情境二：教師評閱週報與提供回饋

**參與者**：資工系教授

**情境描述**： 教授指導多組畢業專題，每週需要審閱學生提交的週報並給予回饋。週報截止日的隔天（週六），教授安排時間檢視並評分各組的週報。

**使用案例**： 1. 教授登入系統後，進入「進度評分」頁面 2. 系統顯示日曆視圖，標記出已有學生提交週報的日期 3. 教授點選昨天（週五）的日期，系統顯示所有已提交週報的組別 4. 教授選擇「智慧家居系統」專題組，查看該組所有成員的週報 5. 系統顯示該組每位成員的週報內容、上傳檔案及歷史進度比對 6. 教授發現學生的程式測試方法有改進空間，於評論區提供具體建議 7. 教授為學生的週報評分並提交，系統記錄評分時間 8. 教授繼續評閱其他組員的週報，完成本週所有評分工作 9. 教授進入「組別管理」，根據週報表現調整某些學生的分組

### 情境三：系統管理與專題分配調整

**參與者**：系統管理員

**情境描述**： 新學年開始，專題指導老師選擇時間已截止，同時系上來了新教授和轉學生。此外，系主任需要查看上一學年度的所有專題記錄作為參考。管理員需要處理這些系統管理任務。

**使用案例**： 1. 管理員登入系統，進入「審核介面」頁面 2. 系統顯示待審核的新用戶申請列表，包含新教授和轉學生 3. 管理員審核新進教授的帳號申請，並分配教師權限 4. 管理員審核轉學生的帳號申請，並分配學生權限 5. 審核完成後，管理員進入學生「組別管理」頁面 6. 根據學生的專長和老師需求，管理員調整部分學生的專題組別分配 7. 管理員確認每個專題組別的人數平衡 8. 接著，管理員進入「統計功能」頁面 9. 管理員選擇上一學年度的時間範圍，查詢所有專題記錄 10. 系統生成視覺化報表，顯示各組學生的進度完成情況與評分情況 11. 管理員匯出這些統計數據，準備提供給系主任作為本學年度專題規劃參考

## 🖥️ 系統需求說明

功能類別

功能描述

📊 用戶管理

• 註冊與登入：所有用戶需要註冊帳號並登入系統才能使用功能 • 權限管理：系統需區分學生、教師和管理員三種權限 • 帳號審核：新用戶註冊須經管理員審核後方可使用 • 密碼管理：提供密碼重設和修改功能

👨‍🎓 學生功能

• 個人資料管理：查看個人基本資料 • 週報提交：填寫本周專題報告項目和下周工作計畫 • 檔案上傳：上傳專題相關檔案（設計圖、程式碼、文件等） • 進度查詢：以日曆方式查看自己和組員的週報提交情況 • 回饋查看：查看教師對週報的評語和評分

👨‍🏫 教師功能

• 組別管理：建立、編輯專題組別，分配和調整組員 • 進度評核：查看、評分學生提交的週報，提供評語 • 評分記錄：查看過去的評分的紀錄 • 歷年專題查詢：查看負責的學生過往學年的專題資料

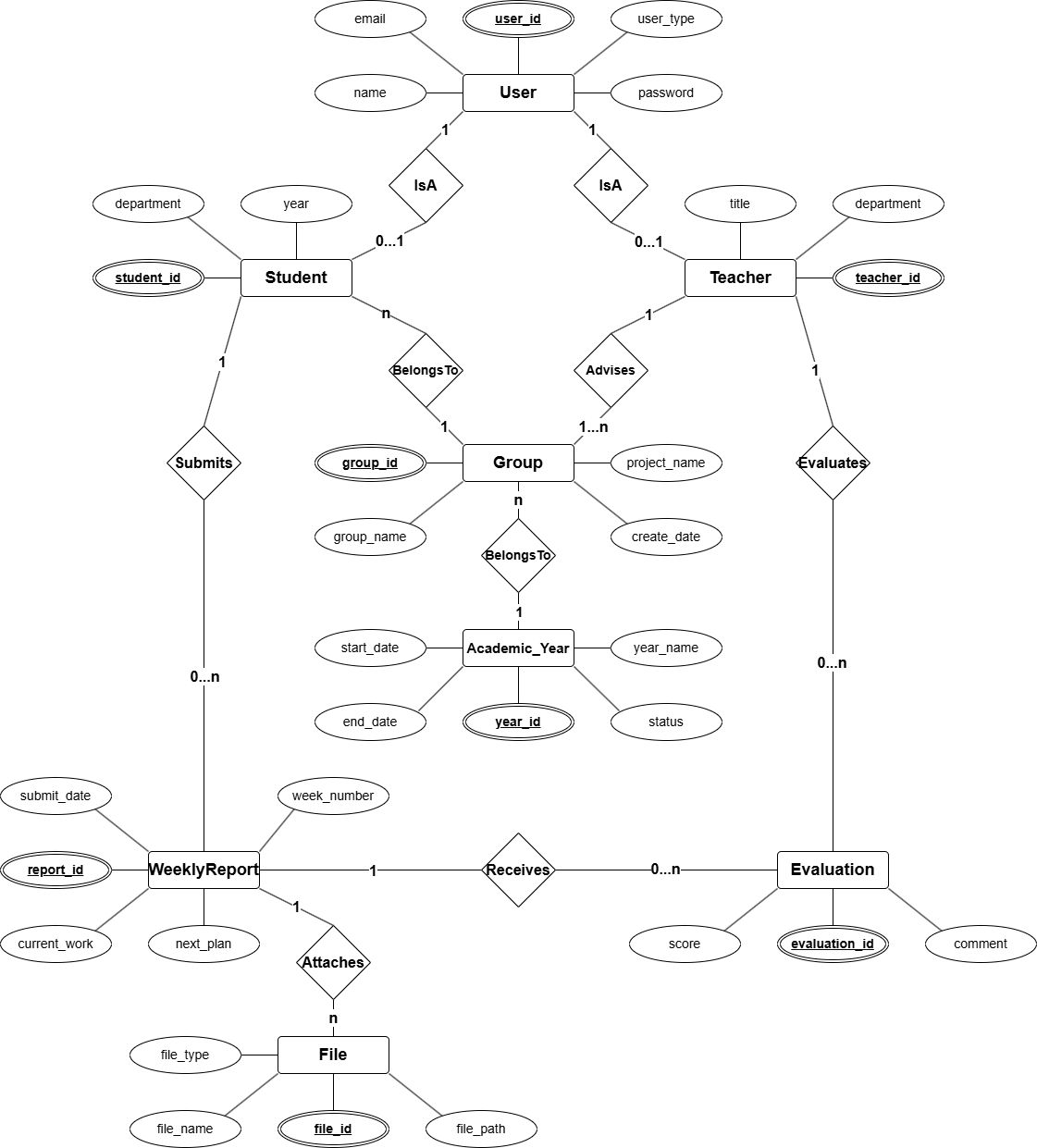
👨‍💼 管理員功能

• 用戶審核：審核新用戶註冊申請，分配適當權限 • 用戶調整：根據需求調整學生所屬的教授和實驗室 • 檔案管理：查看往年所有學生的專題 • 系統監控：監控系統運作狀況，查看系統日誌

🔄 系統共通功能

• 日曆視圖：以日曆形式顯示週報提交與評分情況 • 檔案管理：支援多種格式檔案上傳、下載與預覽 • 搜尋功能：按年份、組別、關鍵字等條件搜尋專題資料 • 通知功能：週報提交、評分完成等事件的通知提醒

## 📊 ER Diagram



image

### ER Diagram 實體說明

#### 主要實體 (Entities)

1. **用戶 (User)**
   * 屬性：用戶ID、姓名、電子郵件、密碼、用戶類型、建立日期、狀態
   * 說明：系統中所有用戶的基本資訊
2. **學生 (Student)**
   * 屬性：學號、科系、入學年份
   * 說明：學生特有的學術資訊
3. **教師 (Teacher)**
   * 屬性：教師編號、科系、職稱
   * 說明：教師特有的職業資訊
4. **組別 (Group)**
   * 屬性：組別ID、組別名稱、專題名稱、建立日期
   * 說明：專題組別的基本資訊
5. **週報 (WeeklyReport)**
   * 屬性：週報ID、本週工作、下週計畫、提交日期、週次、年份
   * 說明：學生提交的週報內容
6. **評分 (Evaluation)**
   * 屬性：評分ID、分數、評語、評分日期
   * 說明：教師對週報的評分記錄
7. **學年度 (Academic\_Year)**
   * 屬性：學年ID、學年名稱、開始日期、結束日期、狀態
   * 說明：管理不同學年度的資訊
8. **評分標準 (Evaluation\_Criteria)**
   * 屬性：標準ID、名稱、描述、權重
   * 說明：評分的具體標準和權重
9. **檔案 (File)**
   * 屬性：檔案ID、檔案名稱、檔案路徑、檔案大小、檔案類型、上傳日期
   * 說明：系統中的檔案管理
10. **通知 (Notification)**
    * 屬性：通知ID、內容、類型、已讀狀態、建立日期
    * 說明：系統通知訊息
11. **系統日誌 (System\_Log)**
    * 屬性：日誌ID、操作類型、實體類型、實體ID、描述、IP位址、記錄時間
    * 說明：系統操作記錄

#### 主要關係 (Relationships)

1. **用戶-學生**：一對一關係，一個用戶帳號對應一個學生身份
2. **用戶-教師**：一對一關係，一個用戶帳號對應一個教師身份
3. **教師-組別**：一對多關係，一位教師可指導多個組別
4. **學生-組別**：多對一關係，多位學生屬於一個組別
5. **學生-週報**：一對多關係，一位學生可提交多份週報
6. **週報-評分**：一對一關係，一份週報對應一個評分記錄
7. **教師-評分**：一對多關係，一位教師可評分多份週報
8. **週報-檔案**：一對多關係，一份週報可包含多個檔案
9. **學年度-組別**：一對多關係，一個學年度包含多個組別
10. **學年度-評分標準**：一對多關係，一個學年度有多個評分標準

## 資料庫使用者與權限設計

### 權限分離原則

基於最小權限原則和角色分離概念，設計多個資料庫使用者帳號，避免使用單一root帳號進行所有操作，確保系統安全性和操作可追蹤性。

### 資料庫使用者設計

#### 1. 資料庫管理員 (pts\_admin)

CREATE USER 'pts\_admin'@'localhost' IDENTIFIED BY 'Admin@2024PTS!';  
GRANT ALL PRIVILEGES ON project\_tracking\_system.\* TO 'pts\_admin'@'localhost';  
GRANT CREATE, DROP, ALTER ON \*.\* TO 'pts\_admin'@'localhost';  
FLUSH PRIVILEGES;

**權限範圍**： - 完整資料庫管理權限：可建立、刪除、修改資料表結構 - 資料管理權限：可執行所有資料操作（增刪改查） - 使用者管理權限：可建立和管理其他資料庫使用者 - 使用時機：系統維護、資料庫結構變更、初始化設定

#### 2. 應用程式主要使用者 (pts\_app)

CREATE USER 'pts\_app'@'%' IDENTIFIED BY 'AppUser@2024PTS!';  
GRANT SELECT, INSERT, UPDATE, DELETE ON project\_tracking\_system.\* TO 'pts\_app'@'%';  
REVOKE DROP, ALTER, CREATE, INDEX ON project\_tracking\_system.\* FROM 'pts\_app'@'%';  
FLUSH PRIVILEGES;

**權限範圍**： - 資料操作權限：可執行基本的增刪改查操作 - 禁止結構變更：不可修改資料表結構、建立索引 - 網路存取：允許從任何主機連接（適合Web應用） - 使用時機：Web應用程式的主要資料庫連接

#### 3. 教師角色使用者 (pts\_teacher)

CREATE USER 'pts\_teacher'@'localhost' IDENTIFIED BY 'Teacher@2024PTS!';  
GRANT SELECT ON project\_tracking\_system.users TO 'pts\_teacher'@'localhost';  
GRANT SELECT ON project\_tracking\_system.students TO 'pts\_teacher'@'localhost';  
GRANT SELECT ON project\_tracking\_system.groups TO 'pts\_teacher'@'localhost';  
GRANT SELECT ON project\_tracking\_system.weekly\_reports TO 'pts\_teacher'@'localhost';  
GRANT SELECT ON project\_tracking\_system.academic\_years TO 'pts\_teacher'@'localhost';  
GRANT SELECT ON project\_tracking\_system.files TO 'pts\_teacher'@'localhost';  
GRANT SELECT, INSERT, UPDATE ON project\_tracking\_system.evaluations TO 'pts\_teacher'@'localhost';  
GRANT INSERT ON project\_tracking\_system.notifications TO 'pts\_teacher'@'localhost';  
GRANT INSERT ON project\_tracking\_system.system\_logs TO 'pts\_teacher'@'localhost';  
GRANT UPDATE ON project\_tracking\_system.groups TO 'pts\_teacher'@'localhost';  
GRANT UPDATE (group\_id) ON project\_tracking\_system.students TO 'pts\_teacher'@'localhost';  
FLUSH PRIVILEGES;

**權限範圍**： - 查看權限：可查看學生資料、週報、組別資訊 - 評分權限：可新增、修改評分記錄 - 組別管理權限：可調整組別資訊和學生分組 - 限制範圍：不可修改使用者帳號、密碼等敏感資料

#### 4. 學生角色使用者 (pts\_student)

CREATE USER 'pts\_student'@'localhost' IDENTIFIED BY 'Student@2024PTS!';  
GRANT SELECT ON project\_tracking\_system.academic\_years TO 'pts\_student'@'localhost';  
GRANT SELECT ON project\_tracking\_system.groups TO 'pts\_student'@'localhost';  
GRANT SELECT, INSERT ON project\_tracking\_system.weekly\_reports TO 'pts\_student'@'localhost';  
GRANT SELECT, INSERT ON project\_tracking\_system.files TO 'pts\_student'@'localhost';  
GRANT SELECT ON project\_tracking\_system.evaluations TO 'pts\_student'@'localhost';  
GRANT SELECT, UPDATE (is\_read) ON project\_tracking\_system.notifications TO 'pts\_student'@'localhost';  
FLUSH PRIVILEGES;

**權限範圍**： - 週報提交權限：可新增週報和上傳檔案 - 查看權限：可查看自己的評分和通知 - 限制查看：僅能查看與自己相關的資料

#### 5. 報表分析使用者 (pts\_report)

CREATE USER 'pts\_report'@'localhost' IDENTIFIED BY 'Report@2024PTS!';  
GRANT SELECT ON project\_tracking\_system.\* TO 'pts\_report'@'localhost';  
FLUSH PRIVILEGES;

**權限範圍**： - 唯讀權限：僅可查詢所有資料表 - 禁止修改：不可進行任何資料異動

#### 6. 備份使用者 (pts\_backup)

CREATE USER 'pts\_backup'@'localhost' IDENTIFIED BY 'Backup@2024PTS!';  
GRANT SELECT, LOCK TABLES, SHOW VIEW, EVENT, TRIGGER ON project\_tracking\_system.\* TO 'pts\_backup'@'localhost';  
GRANT RELOAD, PROCESS ON \*.\* TO 'pts\_backup'@'localhost';  
FLUSH PRIVILEGES;

**權限範圍**： - 備份權限：可執行完整資料庫備份 - 鎖定權限：備份過程中可鎖定資料表確保一致性

### 權限驗證預存程序

#### 學生週報提交驗證

DELIMITER $$  
CREATE PROCEDURE secure\_submit\_report(  
 IN p\_student\_id VARCHAR(20),  
 IN p\_this\_week\_work TEXT,  
 IN p\_next\_week\_plan TEXT,  
 IN p\_week\_number INT,  
 IN p\_year\_id INT  
)  
BEGIN  
 DECLARE current\_user\_id INT;  
 DECLARE student\_user\_id INT;  
   
 SELECT user\_id INTO current\_user\_id   
 FROM users   
 WHERE email = SUBSTRING\_INDEX(USER(), '@', 1);  
   
 SELECT user\_id INTO student\_user\_id   
 FROM students   
 WHERE student\_id = p\_student\_id;  
   
 IF current\_user\_id = student\_user\_id THEN  
 INSERT INTO weekly\_reports (student\_id, group\_id, this\_week\_work, next\_week\_plan, week\_number, year\_id)  
 SELECT p\_student\_id, s.group\_id, p\_this\_week\_work, p\_next\_week\_plan, p\_week\_number, p\_year\_id  
 FROM students s WHERE s.student\_id = p\_student\_id;  
   
 INSERT INTO system\_logs (user\_id, action\_type, table\_name, description)  
 VALUES (current\_user\_id, 'create', 'weekly\_reports', CONCAT('學生提交第', p\_week\_number, '週週報'));  
 ELSE  
 SIGNAL SQLSTATE '45000' SET MESSAGE\_TEXT = '權限不足：只能提交自己的週報';  
 END IF;  
END$$  
DELIMITER ;

#### 教師評分權限驗證

DELIMITER $$  
CREATE PROCEDURE secure\_evaluate\_report(  
 IN p\_report\_id INT,  
 IN p\_teacher\_id VARCHAR(20),  
 IN p\_score INT,  
 IN p\_comments TEXT  
)  
BEGIN  
 DECLARE report\_group\_teacher VARCHAR(20);  
   
 SELECT g.teacher\_id INTO report\_group\_teacher  
 FROM weekly\_reports wr  
 JOIN groups g ON wr.group\_id = g.group\_id  
 WHERE wr.report\_id = p\_report\_id;  
   
 IF report\_group\_teacher = p\_teacher\_id THEN  
 INSERT INTO evaluations (report\_id, teacher\_id, score, comments)  
 VALUES (p\_report\_id, p\_teacher\_id, p\_score, p\_comments);  
   
 INSERT INTO notifications (user\_id, title, content, type)  
 SELECT s.user\_id, '週報評分完成', CONCAT('您的週報已獲得評分：', p\_score, '分'), 'evaluation'  
 FROM weekly\_reports wr  
 JOIN students s ON wr.student\_id = s.student\_id  
 WHERE wr.report\_id = p\_report\_id;  
 ELSE  
 SIGNAL SQLSTATE '45000' SET MESSAGE\_TEXT = '權限不足：只能評分自己指導組別的週報';  
 END IF;  
END$$  
DELIMITER ;

## 1. 用戶資料表 (users)

### SQL建立語句

CREATE TABLE users (  
 user\_id INT PRIMARY KEY AUTO\_INCREMENT,  
 name VARCHAR(50) NOT NULL,  
 email VARCHAR(100) NOT NULL UNIQUE,  
 user\_type ENUM('student', 'teacher', 'admin') NOT NULL,  
 status ENUM('pending', 'active', 'suspended') DEFAULT 'pending',  
 created\_at DATETIME DEFAULT CURRENT\_TIMESTAMP,  
 updated\_at DATETIME DEFAULT CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP,  
   
 -- 完整性限制  
 CONSTRAINT chk\_email\_format CHECK (email REGEXP '^[A-Za-z0-9.\_%+-]+@[A-Za-z0-9.-]+\.[A-Za-z]{2,}$'),  
 CONSTRAINT chk\_name\_length CHECK (CHAR\_LENGTH(name) >= 2)  
);

### 完整性限制

| 欄位名稱 | 完整性限制 |
| --- | --- |
| user\_id | **主鍵約束 (PRIMARY KEY)**：確保每筆記錄的唯一識別**自動遞增 (AUTO\_INCREMENT)**：從1開始，每次新增記錄自動遞增1**非空約束 (NOT NULL)**：主鍵自動具有非空屬性**資料型別**：INT，範圍1-2147483647 |
| name | **非空約束 (NOT NULL)**：姓名為必填欄位，不允許空值**長度檢查約束 (CHECK)**：姓名至少2個字元**資料型別**：VARCHAR(50)，最多50個UTF-8字元**字元集**：支援中英文、數字、空格，UTF-8編碼 |
| email | **非空約束 (NOT NULL)**：電子郵件為必填欄位**唯一性約束 (UNIQUE)**：系統內電子郵件不可重複**格式檢查約束 (CHECK)**：必須符合正規表達式 ^[A-Za-z0-9.\_%+-]+@[A-Za-z0-9.-]+\.[A-Za-z]{2,}$　• 本地部分：允許英數字、點號(.)、底線(\_)、百分號(%)、加號(+)、減號(-)　• 必須包含一個@符號　• 網域部分：允許英數字、點號(.)、減號(-)　• 頂級網域：至少2個英文字母**資料型別**：VARCHAR(100)，最多100個字元 |
| user\_type | **非空約束 (NOT NULL)**：使用者類型為必填欄位**列舉約束 (ENUM)**：只能輸入’student’、‘teacher’、’admin’三種值**值域完整性**：資料庫層面強制執行角色限制**大小寫敏感**：嚴格匹配列舉值 |
| status | **列舉約束 (ENUM)**：只能輸入’pending’、‘active’、’suspended’三種值**預設值約束 (DEFAULT)**：新註冊用戶預設為’pending’狀態**狀態定義**：pending(待審核)、active(已啟用)、suspended(已停權) |
| created\_at | **預設值約束 (DEFAULT CURRENT\_TIMESTAMP)**：系統自動記錄建立時間**資料型別**：DATETIME，格式YYYY-MM-DD HH:mm:ss**時間範圍**：1000-01-01 00:00:00 到 9999-12-31 23:59:59**時區**：使用資料庫系統時區 |
| updated\_at | **預設值約束 (DEFAULT CURRENT\_TIMESTAMP)**：建立時設為當前時間**自動更新約束 (ON UPDATE CURRENT\_TIMESTAMP)**：每次UPDATE操作時自動更新為當前時間**資料型別**：DATETIME，格式YYYY-MM-DD HH:mm:ss**自動維護**：由資料庫引擎自動管理，無需手動更新 |

## 2. 密碼資料表 (user\_passwords)

### SQL建立語句

CREATE TABLE user\_passwords (  
 password\_id INT PRIMARY KEY AUTO\_INCREMENT,  
 user\_id INT NOT NULL,  
 password\_hash VARCHAR(255) NOT NULL,  
 created\_at DATETIME DEFAULT CURRENT\_TIMESTAMP,  
   
 -- 外鍵約束  
 FOREIGN KEY (user\_id) REFERENCES users(user\_id) ON DELETE CASCADE,  
   
 -- 完整性限制  
 CONSTRAINT chk\_password\_hash\_length CHECK (CHAR\_LENGTH(password\_hash) >= 32)  
);

### 完整性限制

| 欄位名稱 | 完整性限制 |
| --- | --- |
| password\_id | **主鍵約束 (PRIMARY KEY)**：密碼記錄的唯一識別碼**自動遞增 (AUTO\_INCREMENT)**：從1開始自動遞增**非空約束 (NOT NULL)**：主鍵自動具有非空屬性**資料型別**：INT，範圍1-2147483647 |
| user\_id | **非空約束 (NOT NULL)**：必須關聯到特定用戶**外鍵約束 (FOREIGN KEY)**：與users表的user\_id建立參照關係**級聯刪除 (ON DELETE CASCADE)**：用戶刪除時密碼記錄同步刪除**參照完整性**：確保每個密碼記錄都對應有效的用戶帳號**資料型別**：INT，必須為正整數 |
| password\_hash | **非空約束 (NOT NULL)**：密碼雜湊值為必填欄位**長度檢查約束 (CHECK)**：雜湊值至少32個字元**資料型別**：VARCHAR(255)，最多255個字元**安全性要求**：僅允許雜湊值，禁止明文密碼**字元集**：支援十六進位字元(0-9, a-f, A-F)和Base64字元 |
| created\_at | **預設值約束 (DEFAULT CURRENT\_TIMESTAMP)**：系統自動記錄密碼建立時間**資料型別**：DATETIME，格式YYYY-MM-DD HH:mm:ss**審計追蹤**：記錄密碼設定或變更的確切時間**不可更新**：密碼記錄建立後此欄位不應被修改 |

## 3. 學生資料表 (students)

### SQL建立語句

CREATE TABLE students (  
 student\_id VARCHAR(20) PRIMARY KEY,  
 user\_id INT NOT NULL UNIQUE,  
 department VARCHAR(50) NOT NULL,  
 admission\_year YEAR NOT NULL,  
 group\_id INT,  
   
 -- 外鍵約束  
 FOREIGN KEY (user\_id) REFERENCES users(user\_id) ON DELETE CASCADE,  
 FOREIGN KEY (group\_id) REFERENCES groups(group\_id) ON DELETE SET NULL,  
   
 -- 完整性限制  
 CONSTRAINT chk\_student\_id\_format CHECK (student\_id REGEXP '^[0-9]{8}$'),  
 CONSTRAINT chk\_admission\_year CHECK (admission\_year BETWEEN 1950 AND YEAR(CURDATE()) + 1)  
);

### 完整性限制

| 欄位名稱 | 完整性限制 |
| --- | --- |
| student\_id | **主鍵約束 (PRIMARY KEY)**：學號作為學生記錄的主要識別碼**格式檢查約束 (CHECK)**：必須符合正規表達式 ^[0-9]{8}$**格式要求**：恰好8個連續數字，如40943212**字元限制**：僅允許數字0-9，不允許英文字母、特殊符號或空格**非空約束 (NOT NULL)**：主鍵自動具有非空屬性**資料型別**：VARCHAR(20)，實際使用8個字元 |
| user\_id | **非空約束 (NOT NULL)**：必須關聯到特定用戶帳號**唯一性約束 (UNIQUE)**：一個用戶帳號只能對應一個學生身份**外鍵約束 (FOREIGN KEY)**：與users表的user\_id建立參照關係**級聯刪除 (ON DELETE CASCADE)**：用戶帳號刪除時學生資料同步刪除**一對一關係**：確保用戶帳號與學生身份的一對一映射**資料型別**：INT，必須為正整數 |
| department | **非空約束 (NOT NULL)**：就讀科系為必填欄位**資料型別**：VARCHAR(50)，最多50個UTF-8字元**字元集**：支援中英文科系名稱**內容範例**：資訊工程系、電機工程系、資訊管理系等 |
| admission\_year | **非空約束 (NOT NULL)**：入學年份為必填欄位**範圍檢查約束 (CHECK)**：入學年份必須在1950年至明年之間**最小值**：1950年，涵蓋歷史資料需求**最大值**：當前年度+1年，允許預先建立下學年資料**資料型別**：YEAR，4位數年份格式(1901-2155)**動態範圍**：使用YEAR(CURDATE())確保範圍隨時間更新 |
| group\_id | **可空值 (NULL)**：允許學生暫時未分配組別**外鍵約束 (FOREIGN KEY)**：與groups表的group\_id建立參照關係**設為空值 (ON DELETE SET NULL)**：組別刪除時將此欄位設為NULL，保持學生記錄完整**分組管理**：支援動態分組調整，一個學生同時只能屬於一個組別**資料型別**：INT，若不為NULL則必須為正整數 |

## 4. 教師資料表 (teachers)

### SQL建立語句

CREATE TABLE teachers (  
 teacher\_id VARCHAR(20) PRIMARY KEY,  
 user\_id INT NOT NULL UNIQUE,  
 department VARCHAR(50) NOT NULL,  
 title VARCHAR(30) NOT NULL,  
   
 -- 外鍵約束  
 FOREIGN KEY (user\_id) REFERENCES users(user\_id) ON DELETE CASCADE,  
   
 -- 完整性限制  
 CONSTRAINT chk\_teacher\_id\_format CHECK (teacher\_id REGEXP '^T[0-9]{6}$')  
);

### 完整性限制

| 欄位名稱 | 完整性限制 |
| --- | --- |
| teacher\_id | **主鍵約束 (PRIMARY KEY)**：教師編號作為教師記錄的唯一識別碼**格式檢查約束 (CHECK)**：必須符合正規表達式 ^T[0-9]{6}$**格式要求**：T開頭 + 6位數字，如T123456**首字元**：必須為大寫英文字母T**後續字元**：必須為6個連續數字(0-9)**總長度**：恰好7個字元**非空約束 (NOT NULL)**：主鍵自動具有非空屬性**資料型別**：VARCHAR(20)，實際使用7個字元 |
| user\_id | **非空約束 (NOT NULL)**：必須關聯到特定用戶帳號**唯一性約束 (UNIQUE)**：一個用戶帳號只能對應一個教師身份**外鍵約束 (FOREIGN KEY)**：與users表的user\_id建立參照關係**級聯刪除 (ON DELETE CASCADE)**：用戶帳號刪除時教師資料同步刪除**一對一關係**：確保用戶帳號與教師身份的一對一映射**資料型別**：INT，必須為正整數 |
| department | **非空約束 (NOT NULL)**：任職科系為必填欄位**資料型別**：VARCHAR(50)，最多50個UTF-8字元**字元集**：支援中英文科系名稱**內容範例**：資訊工程系、電機工程系、資訊管理系等 |
| title | **非空約束 (NOT NULL)**：職稱為必填欄位**資料型別**：VARCHAR(30)，最多30個UTF-8字元**字元集**：支援中英文職稱**職稱範例**：教授、副教授、助理教授、講師、系主任等 |

## 5. 學年度資料表 (academic\_years)

### SQL建立語句

CREATE TABLE academic\_years (  
 year\_id INT PRIMARY KEY AUTO\_INCREMENT,  
 year\_name VARCHAR(20) NOT NULL UNIQUE,  
 start\_date DATE NOT NULL,  
 end\_date DATE NOT NULL,  
 status ENUM('active', 'archived') DEFAULT 'active',  
   
 -- 完整性限制  
 CONSTRAINT chk\_year\_name\_format CHECK (year\_name REGEXP '^[0-9]{4}-[0-9]{4}$'),  
 CONSTRAINT chk\_end\_after\_start CHECK (end\_date > start\_date),  
 CONSTRAINT chk\_year\_span CHECK (  
 CAST(SUBSTRING(year\_name, 6, 4) AS UNSIGNED) =   
 CAST(SUBSTRING(year\_name, 1, 4) AS UNSIGNED) + 1  
 )  
);

### 完整性限制

| 欄位名稱 | 完整性限制 |
| --- | --- |
| year\_id | **主鍵約束 (PRIMARY KEY)**：學年度記錄的唯一識別碼**自動遞增 (AUTO\_INCREMENT)**：從1開始自動遞增**非空約束 (NOT NULL)**：主鍵自動具有非空屬性**資料型別**：INT，範圍1-2147483647 |
| year\_name | **非空約束 (NOT NULL)**：學年度名稱為必填欄位**唯一性約束 (UNIQUE)**：學年度名稱不可重複**格式檢查約束 (CHECK)**：必須符合正規表達式 ^[0-9]{4}-[0-9]{4}$**格式要求**：YYYY-YYYY格式，如2024-2025**前4位**：起始年份，必須為4位數字**第5位**：分隔符號，必須為減號(-)**後4位**：結束年份，必須為4位數字**年份邏輯檢查 (CHECK)**：後一年份必須等於前一年份加1**資料型別**：VARCHAR(20)，實際使用9個字元 |
| start\_date | **非空約束 (NOT NULL)**：學年度開始日期為必填欄位**邏輯檢查約束 (CHECK)**：開始日期必須早於結束日期**資料型別**：DATE，格式YYYY-MM-DD**日期範圍**：1000-01-01 到 9999-12-31**建議值**：通常為每年9月1日 |
| end\_date | **非空約束 (NOT NULL)**：學年度結束日期為必填欄位**邏輯檢查約束 (CHECK)**：結束日期必須晚於開始日期**資料型別**：DATE，格式YYYY-MM-DD**日期範圍**：1000-01-01 到 9999-12-31**建議值**：通常為隔年6月30日 |
| status | **列舉約束 (ENUM)**：只能輸入’active’、’archived’兩種值**預設值約束 (DEFAULT)**：新建學年度預設為’active’狀態**狀態定義**：active(進行中或未來學年度)、archived(已結束的歷史學年度) |

## 6. 組別資料表 (groups)

### SQL建立語句

CREATE TABLE groups (  
 group\_id INT PRIMARY KEY AUTO\_INCREMENT,  
 group\_name VARCHAR(50) NOT NULL,  
 project\_title VARCHAR(100) NOT NULL,  
 teacher\_id VARCHAR(20),  
 year\_id INT NOT NULL,  
 created\_at DATETIME DEFAULT CURRENT\_TIMESTAMP,  
 status ENUM('active', 'completed', 'suspended') DEFAULT 'active',  
   
 -- 外鍵約束  
 FOREIGN KEY (teacher\_id) REFERENCES teachers(teacher\_id) ON DELETE SET NULL,  
 FOREIGN KEY (year\_id) REFERENCES academic\_years(year\_id) ON DELETE RESTRICT,  
   
 -- 完整性限制  
 CONSTRAINT chk\_group\_name\_length CHECK (CHAR\_LENGTH(group\_name) >= 2),  
 CONSTRAINT chk\_project\_title\_length CHECK (CHAR\_LENGTH(project\_title) >= 5)  
);  
  
-- 建立同一學年度組別名稱唯一性約束  
CREATE UNIQUE INDEX idx\_groups\_unique\_name\_year ON groups(group\_name, year\_id);

### 完整性限制

| 欄位名稱 | 完整性限制 |
| --- | --- |
| group\_id | **主鍵約束 (PRIMARY KEY)**：組別記錄的唯一識別碼**自動遞增 (AUTO\_INCREMENT)**：從1開始自動遞增**非空約束 (NOT NULL)**：主鍵自動具有非空屬性**資料型別**：INT，範圍1-2147483647 |
| group\_name | **非空約束 (NOT NULL)**：組別名稱為必填欄位**長度檢查約束 (CHECK)**：組別名稱至少2個字元**學年度內唯一約束 (UNIQUE INDEX)**：同一學年度內組別名稱不可重複**複合唯一性**：(group\_name, year\_id)組成複合唯一索引**資料型別**：VARCHAR(50)，最多50個UTF-8字元**字元集**：支援中英文組別名稱 |
| project\_title | **非空約束 (NOT NULL)**：專題名稱為必填欄位**長度檢查約束 (CHECK)**：專題名稱至少5個字元**資料型別**：VARCHAR(100)，最多100個UTF-8字元**字元集**：支援中英文混合的專題名稱和描述**內容要求**：應包含專題的主要技術或應用領域描述 |
| teacher\_id | **可空值 (NULL)**：允許組別暫時無指導教師**外鍵約束 (FOREIGN KEY)**：與teachers表的teacher\_id建立參照關係**設為空值 (ON DELETE SET NULL)**：指導教師刪除時將此欄位設為NULL**格式要求**：若不為NULL，必須符合教師編號格式(T+6位數字)**資料型別**：VARCHAR(20)，實際使用7個字元或NULL |
| year\_id | **非空約束 (NOT NULL)**：必須關聯到特定學年度**外鍵約束 (FOREIGN KEY)**：與academic\_years表的year\_id建立參照關係**限制刪除 (ON DELETE RESTRICT)**：不允許刪除已有組別的學年度**參照完整性**：確保組別必須歸屬於有效的學年度**資料型別**：INT，必須為正整數 |
| created\_at | **預設值約束 (DEFAULT CURRENT\_TIMESTAMP)**：系統自動記錄組別建立時間**資料型別**：DATETIME，格式YYYY-MM-DD HH:mm:ss**時間範圍**：1000-01-01 00:00:00 到 9999-12-31 23:59:59**不可更新**：組別建立後此欄位不應被修改 |
| status | **列舉約束 (ENUM)**：只能輸入’active’、‘completed’、’suspended’三種值**預設值約束 (DEFAULT)**：新建組別預設為’active’狀態**狀態定義**：active(進行中)、completed(已完成)、suspended(已暫停) |

## 7. 週報資料表 (weekly\_reports)

### SQL建立語句

CREATE TABLE weekly\_reports (  
 report\_id INT PRIMARY KEY AUTO\_INCREMENT,  
 student\_id VARCHAR(20) NOT NULL,  
 group\_id INT NOT NULL,  
 week\_number INT NOT NULL,  
 year\_id INT NOT NULL,  
 this\_week\_work TEXT NOT NULL,  
 next\_week\_plan TEXT NOT NULL,  
 submitted\_at DATETIME DEFAULT CURRENT\_TIMESTAMP,  
 status ENUM('submitted', 'late', 'missing') DEFAULT 'submitted',  
   
 -- 外鍵約束  
 FOREIGN KEY (student\_id) REFERENCES students(student\_id) ON DELETE CASCADE,  
 FOREIGN KEY (group\_id) REFERENCES groups(group\_id) ON DELETE CASCADE,  
 FOREIGN KEY (year\_id) REFERENCES academic\_years(year\_id) ON DELETE RESTRICT,  
   
 -- 完整性限制  
 CONSTRAINT chk\_week\_number\_range CHECK (week\_number BETWEEN 1 AND 18),  
 CONSTRAINT chk\_work\_content\_length CHECK (CHAR\_LENGTH(this\_week\_work) >= 10),  
 CONSTRAINT chk\_plan\_content\_length CHECK (CHAR\_LENGTH(next\_week\_plan) >= 10),  
   
 -- 學生每週每學年度只能有一份週報  
 UNIQUE KEY unique\_student\_week\_year (student\_id, week\_number, year\_id)  
);

### 完整性限制

| 欄位名稱 | 完整性限制 |
| --- | --- |
| report\_id | **主鍵約束 (PRIMARY KEY)**：週報記錄的唯一識別碼**自動遞增 (AUTO\_INCREMENT)**：從1開始自動遞增**非空約束 (NOT NULL)**：主鍵自動具有非空屬性**資料型別**：INT，範圍1-2147483647 |
| student\_id | **非空約束 (NOT NULL)**：必須關聯到特定學生**外鍵約束 (FOREIGN KEY)**：與students表的student\_id建立參照關係**級聯刪除 (ON DELETE CASCADE)**：學生刪除時週報記錄同步刪除**複合唯一約束 (UNIQUE KEY)**：與week\_number、year\_id組成複合唯一鍵**格式要求**：必須為8位數字格式**資料型別**：VARCHAR(20)，實際使用8個字元 |
| group\_id | **非空約束 (NOT NULL)**：必須關聯到特定組別**外鍵約束 (FOREIGN KEY)**：與groups表的group\_id建立參照關係**級聯刪除 (ON DELETE CASCADE)**：組別刪除時週報記錄同步刪除**資料型別**：INT，必須為正整數 |
| week\_number | **非空約束 (NOT NULL)**：必須填入週次**範圍檢查約束 (CHECK)**：週次必須在1到18之間**複合唯一約束 (UNIQUE KEY)**：與student\_id、year\_id組成複合唯一鍵**學期制度**：對應18週制學期安排**資料型別**：INT，範圍1-18 |
| year\_id | **非空約束 (NOT NULL)**：必須關聯到特定學年度**外鍵約束 (FOREIGN KEY)**：與academic\_years表的year\_id建立參照關係**限制刪除 (ON DELETE RESTRICT)**：不允許刪除已有週報的學年度**複合唯一約束 (UNIQUE KEY)**：與student\_id、week\_number組成複合唯一鍵**資料型別**：INT，必須為正整數 |
| this\_week\_work | **非空約束 (NOT NULL)**：必須填入本週工作內容**長度檢查約束 (CHECK)**：內容至少10個字元**資料型別**：TEXT，最大65,535個字元**字元集**：支援UTF-8編碼，容納中英文混合內容**內容要求**：記錄學生本週實際完成的專題工作項目 |
| next\_week\_plan | **非空約束 (NOT NULL)**：必須填入下週工作計畫**長度檢查約束 (CHECK)**：內容至少10個字元**資料型別**：TEXT，最大65,535個字元**字元集**：支援UTF-8編碼，容納中英文混合內容**內容要求**：記錄學生下週預計進行的專題工作安排 |
| submitted\_at | **預設值約束 (DEFAULT CURRENT\_TIMESTAMP)**：系統自動記錄週報提交時間**資料型別**：DATETIME，格式YYYY-MM-DD HH:mm:ss**時間範圍**：1000-01-01 00:00:00 到 9999-12-31 23:59:59**時效管理**：用於判定提交狀態和遲交情況 |
| status | **列舉約束 (ENUM)**：只能輸入’submitted’、‘late’、’missing’三種值**預設值約束 (DEFAULT)**：預設為’submitted’狀態**狀態定義**：submitted(已提交)、late(遲交)、missing(未交) |

## 8. 評分資料表 (evaluations)

### SQL建立語句

CREATE TABLE evaluations (  
 evaluation\_id INT PRIMARY KEY AUTO\_INCREMENT,  
 report\_id INT NOT NULL UNIQUE,  
 teacher\_id VARCHAR(20) NOT NULL,  
 score INT NOT NULL,  
 comments TEXT,  
 evaluated\_at DATETIME DEFAULT CURRENT\_TIMESTAMP,  
 status ENUM('pending', 'completed') DEFAULT 'completed',  
   
 -- 外鍵約束  
 FOREIGN KEY (report\_id) REFERENCES weekly\_reports(report\_id) ON DELETE CASCADE,  
 FOREIGN KEY (teacher\_id) REFERENCES teachers(teacher\_id) ON DELETE RESTRICT,  
   
 -- 完整性限制  
 CONSTRAINT chk\_score\_range CHECK (score BETWEEN 0 AND 100),  
 CONSTRAINT chk\_comments\_meaningful CHECK (comments IS NULL OR CHAR\_LENGTH(comments) >= 5)  
);

### 完整性限制

| 欄位名稱 | 完整性限制 |
| --- | --- |
| evaluation\_id | **主鍵約束 (PRIMARY KEY)**：評分記錄的唯一識別碼**自動遞增 (AUTO\_INCREMENT)**：從1開始自動遞增**非空約束 (NOT NULL)**：主鍵自動具有非空屬性**資料型別**：INT，範圍1-2147483647 |
| report\_id | **非空約束 (NOT NULL)**：必須關聯到特定週報**唯一性約束 (UNIQUE)**：一份週報只能有一個評分記錄**外鍵約束 (FOREIGN KEY)**：與weekly\_reports表的report\_id建立參照關係**級聯刪除 (ON DELETE CASCADE)**：週報刪除時評分記錄同步刪除**一對一關係**：確保週報與評分的一對一映射**資料型別**：INT，必須為正整數 |
| teacher\_id | **非空約束 (NOT NULL)**：必須關聯到評分教師**外鍵約束 (FOREIGN KEY)**：與teachers表的teacher\_id建立參照關係**限制刪除 (ON DELETE RESTRICT)**：不允許刪除已有評分記錄的教師**格式要求**：必須符合教師編號格式(T+6位數字)**資料型別**：VARCHAR(20)，實際使用7個字元 |
| score | **非空約束 (NOT NULL)**：必須填入評分**範圍檢查約束 (CHECK)**：評分必須在0到100之間**評分制度**：採用百分制評分系統**資料型別**：INT，範圍0-100 |
| comments | **可空值 (NULL)**：評語為選填欄位**長度檢查約束 (CHECK)**：若填入評語則至少5個字元**條件約束**：comments IS NULL OR CHAR\_LENGTH(comments) >= 5**資料型別**：TEXT，最大65,535個字元**字元集**：支援UTF-8編碼，容納中英文評語 |
| evaluated\_at | **預設值約束 (DEFAULT CURRENT\_TIMESTAMP)**：系統自動記錄評分時間**資料型別**：DATETIME，格式YYYY-MM-DD HH:mm:ss**時間範圍**：1000-01-01 00:00:00 到 9999-12-31 23:59:59**審計追蹤**：記錄評分完成的確切時間 |
| status | **列舉約束 (ENUM)**：只能輸入’pending’、’completed’兩種值**預設值約束 (DEFAULT)**：預設為’completed’狀態**狀態定義**：pending(待評分)、completed(已完成) |

## 9. 檔案資料表 (files)

### SQL建立語句

CREATE TABLE files (  
 file\_id INT PRIMARY KEY AUTO\_INCREMENT,  
 report\_id INT NOT NULL,  
 filename VARCHAR(255) NOT NULL,  
 file\_path VARCHAR(500) NOT NULL,  
 file\_size BIGINT NOT NULL,  
 file\_type VARCHAR(100) NOT NULL,  
 uploaded\_at DATETIME DEFAULT CURRENT\_TIMESTAMP,  
   
 -- 外鍵約束  
 FOREIGN KEY (report\_id) REFERENCES weekly\_reports(report\_id) ON DELETE CASCADE,  
   
 -- 完整性限制  
 CONSTRAINT chk\_file\_size\_positive CHECK (file\_size > 0),  
 CONSTRAINT chk\_file\_size\_limit CHECK (file\_size <= 104857600), -- 100MB限制  
 CONSTRAINT chk\_filename\_not\_empty CHECK (CHAR\_LENGTH(filename) > 0)  
);

### 完整性限制

| 欄位名稱 | 完整性限制 |
| --- | --- |
| file\_id | **主鍵約束 (PRIMARY KEY)**：檔案記錄的唯一識別碼**自動遞增 (AUTO\_INCREMENT)**：從1開始自動遞增**非空約束 (NOT NULL)**：主鍵自動具有非空屬性**資料型別**：INT，範圍1-2147483647 |
| report\_id | **非空約束 (NOT NULL)**：必須關聯到特定週報**外鍵約束 (FOREIGN KEY)**：與weekly\_reports表的report\_id建立參照關係**級聯刪除 (ON DELETE CASCADE)**：週報刪除時檔案記錄同步刪除**一對多關係**：一份週報可包含多個檔案**資料型別**：INT，必須為正整數 |
| filename | **非空約束 (NOT NULL)**：必須記錄檔案名稱**非空檢查約束 (CHECK)**：檔案名稱不能為空字串**資料型別**：VARCHAR(255)，最多255個UTF-8字元**字元集**：支援中英文檔案名稱和副檔名**內容要求**：應包含完整檔案名稱和副檔名 |
| file\_path | **非空約束 (NOT NULL)**：檔案儲存路徑為必填**資料型別**：VARCHAR(500)，最多500個字元**路徑格式**：記錄檔案在檔案系統中的完整路徑**字元集**：支援各種作業系統的路徑格式 |
| file\_size | **非空約束 (NOT NULL)**：檔案大小為必填**正數檢查約束 (CHECK)**：檔案大小必須大於0位元組**大小限制約束 (CHECK)**：檔案大小不超過104,857,600位元組(100MB)**資料型別**：BIGINT，支援大檔案，範圍1-104857600 |
| file\_type | **非空約束 (NOT NULL)**：檔案類型描述為必填**資料型別**：VARCHAR(100)，最多100個UTF-8字元**分類用途**：用於檔案管理和權限控制**內容範例**：文件、程式碼、圖片、壓縮檔等 |
| uploaded\_at | **預設值約束 (DEFAULT CURRENT\_TIMESTAMP)**：系統自動記錄檔案上傳時間**資料型別**：DATETIME，格式YYYY-MM-DD HH:mm:ss**時間範圍**：1000-01-01 00:00:00 到 9999-12-31 23:59:59**檔案追蹤**：記錄檔案上傳的確切時間 |

## 10. 通知資料表 (notifications)

### SQL建立語句

CREATE TABLE notifications (  
 notification\_id INT PRIMARY KEY AUTO\_INCREMENT,  
 user\_id INT NOT NULL,  
 title VARCHAR(100) NOT NULL,  
 content TEXT NOT NULL,  
 type ENUM('system', 'evaluation', 'reminder', 'announcement') NOT NULL,  
 is\_read BOOLEAN DEFAULT FALSE,  
 created\_at DATETIME DEFAULT CURRENT\_TIMESTAMP,  
   
 -- 外鍵約束  
 FOREIGN KEY (user\_id) REFERENCES users(user\_id) ON DELETE CASCADE,  
   
 -- 完整性限制  
 CONSTRAINT chk\_title\_not\_empty CHECK (CHAR\_LENGTH(title) > 0),  
 CONSTRAINT chk\_content\_meaningful CHECK (CHAR\_LENGTH(content) >= 5)  
);

### 完整性限制

| 欄位名稱 | 完整性限制 |
| --- | --- |
| notification\_id | **主鍵約束 (PRIMARY KEY)**：通知記錄的唯一識別碼**自動遞增 (AUTO\_INCREMENT)**：從1開始自動遞增**非空約束 (NOT NULL)**：主鍵自動具有非空屬性**資料型別**：INT，範圍1-2147483647 |
| user\_id | **非空約束 (NOT NULL)**：必須關聯到特定用戶**外鍵約束 (FOREIGN KEY)**：與users表的user\_id建立參照關係**級聯刪除 (ON DELETE CASCADE)**：用戶刪除時通知記錄同步刪除**一對多關係**：一個用戶可有多個通知**資料型別**：INT，必須為正整數 |
| title | **非空約束 (NOT NULL)**：必須填入通知標題**非空檢查約束 (CHECK)**：標題不能為空字串**資料型別**：VARCHAR(100)，最多100個UTF-8字元**字元集**：支援中英文通知標題**內容要求**：應簡潔明確地描述通知主題 |
| content | **非空約束 (NOT NULL)**：必須填入通知內容**長度檢查約束 (CHECK)**：內容至少5個字元**資料型別**：TEXT，最大65,535個字元**字元集**：支援UTF-8編碼，容納詳細的通知內容**內容要求**：提供完整的通知詳細資訊 |
| type | **非空約束 (NOT NULL)**：必須指定通知類型**列舉約束 (ENUM)**：只能輸入’system’、‘evaluation’、‘reminder’、’announcement’四種值**類型定義**：system(系統通知)、evaluation(評分通知)、reminder(提醒通知)、announcement(公告通知)**分類用途**：用於通知的分類顯示和處理 |
| is\_read | **布林值約束 (BOOLEAN)**：已讀狀態標記**預設值約束 (DEFAULT FALSE)**：新通知預設為未讀狀態**資料型別**：BOOLEAN，TRUE(已讀)或FALSE(未讀)**狀態管理**：用於追蹤用戶的通知閱讀狀態 |
| created\_at | **預設值約束 (DEFAULT CURRENT\_TIMESTAMP)**：系統自動記錄通知建立時間**資料型別**：DATETIME，格式YYYY-MM-DD HH:mm:ss**時間範圍**：1000-01-01 00:00:00 到 9999-12-31 23:59:59**時序排列**：用於通知的時間排序顯示 |

## 11. 審核資料表 (user\_approvals)

### SQL建立語句

CREATE TABLE user\_approvals (  
 approval\_id INT PRIMARY KEY AUTO\_INCREMENT,  
 user\_id INT NOT NULL,  
 admin\_id INT,  
 request\_type ENUM('registration', 'role\_change', 'account\_recovery') NOT NULL,  
 status ENUM('pending', 'approved', 'rejected') DEFAULT 'pending',  
 request\_data JSON,  
 admin\_notes TEXT,  
 requested\_at DATETIME DEFAULT CURRENT\_TIMESTAMP,  
 processed\_at DATETIME,  
   
 -- 外鍵約束  
 FOREIGN KEY (user\_id) REFERENCES users(user\_id) ON DELETE CASCADE,  
 FOREIGN KEY (admin\_id) REFERENCES users(user\_id) ON DELETE SET NULL,  
   
 -- 完整性限制  
 CONSTRAINT chk\_processed\_after\_requested CHECK (processed\_at IS NULL OR processed\_at >= requested\_at)  
);

### 完整性限制

| 欄位名稱 | 完整性限制 |
| --- | --- |
| approval\_id | **主鍵約束 (PRIMARY KEY)**：審核記錄的唯一識別碼**自動遞增 (AUTO\_INCREMENT)**：從1開始自動遞增**非空約束 (NOT NULL)**：主鍵自動具有非空屬性**資料型別**：INT，範圍1-2147483647 |
| user\_id | **非空約束 (NOT NULL)**：必須關聯到申請審核的用戶**外鍵約束 (FOREIGN KEY)**：與users表的user\_id建立參照關係**級聯刪除 (ON DELETE CASCADE)**：用戶刪除時審核記錄同步刪除**一對多關係**：一個用戶可有多個審核申請**資料型別**：INT，必須為正整數 |
| admin\_id | **可空值 (NULL)**：允許尚未指派審核管理員的情況**外鍵約束 (FOREIGN KEY)**：與users表的user\_id建立參照關係**設為空值 (ON DELETE SET NULL)**：管理員刪除時此欄位設為NULL**角色限制**：應為admin類型用戶**資料型別**：INT，若不為NULL則必須為正整數 |
| request\_type | **非空約束 (NOT NULL)**：必須指定申請類型**列舉約束 (ENUM)**：只能輸入’registration’、‘role\_change’、’account\_recovery’三種值**類型定義**：registration(註冊申請)、role\_change(角色變更)、account\_recovery(帳號恢復)**業務邏輯**：用於區分不同類型的審核流程 |
| status | **列舉約束 (ENUM)**：只能輸入’pending’、‘approved’、’rejected’三種值**預設值約束 (DEFAULT)**：預設為’pending’待審核狀態**狀態定義**：pending(待審核)、approved(已核准)、rejected(已拒絕)**工作流程**：追蹤審核申請的處理狀態 |
| request\_data | **可空值 (NULL)**：申請相關資料為選填**資料型別**：JSON，儲存結構化的申請資料**JSON格式**：支援複雜的申請資料結構**內容範例**：用戶資料、變更內容、申請原因等 |
| admin\_notes | **可空值 (NULL)**：管理員備註為選填**資料型別**：TEXT，最大65,535個字元**字元集**：支援UTF-8編碼，容納中英文備註**用途**：記錄審核過程和決策依據 |
| requested\_at | **預設值約束 (DEFAULT CURRENT\_TIMESTAMP)**：系統自動記錄申請提交時間**資料型別**：DATETIME，格式YYYY-MM-DD HH:mm:ss**時間範圍**：1000-01-01 00:00:00 到 9999-12-31 23:59:59**申請追蹤**：記錄申請的確切提交時間 |
| processed\_at | **可空值 (NULL)**：審核處理完成時間為選填**邏輯檢查約束 (CHECK)**：處理時間不能早於申請時間**條件約束**：processed\_at IS NULL OR processed\_at >= requested\_at**資料型別**：DATETIME，格式YYYY-MM-DD HH:mm:ss**處理追蹤**：記錄審核決定的確切時間 |

## 12. 系統日誌資料表 (system\_logs)

### SQL建立語句

CREATE TABLE system\_logs (  
 log\_id BIGINT PRIMARY KEY AUTO\_INCREMENT,  
 user\_id INT,  
 action\_type ENUM('login', 'logout', 'create', 'update', 'delete', 'view') NOT NULL,  
 table\_name VARCHAR(50),  
 record\_id INT,  
 description TEXT,  
 ip\_address VARCHAR(45),  
 created\_at DATETIME DEFAULT CURRENT\_TIMESTAMP,  
   
 -- 外鍵約束  
 FOREIGN KEY (user\_id) REFERENCES users(user\_id) ON DELETE SET NULL,  
   
 -- 完整性限制  
 CONSTRAINT chk\_ip\_address\_format CHECK (  
 ip\_address IS NULL OR   
 ip\_address REGEXP '^([0-9]{1,3}\.){3}[0-9]{1,3}$|^([0-9a-fA-F]{0,4}:){1,7}[0-9a-fA-F]{0,4}$'  
 )  
);

### 完整性限制

| 欄位名稱 | 完整性限制 |
| --- | --- |
| log\_id | **主鍵約束 (PRIMARY KEY)**：日誌記錄的唯一識別碼**自動遞增 (AUTO\_INCREMENT)**：從1開始自動遞增**非空約束 (NOT NULL)**：主鍵自動具有非空屬性**資料型別**：BIGINT，支援大量日誌記錄，範圍1-9223372036854775807 |
| user\_id | **可空值 (NULL)**：允許系統自動操作無關聯用戶的情況**外鍵約束 (FOREIGN KEY)**：與users表的user\_id建立參照關係**設為空值 (ON DELETE SET NULL)**：用戶刪除時此欄位設為NULL，保持日誌完整性**操作追蹤**：記錄執行操作的用戶身份**資料型別**：INT，若不為NULL則必須為正整數 |
| action\_type | **非空約束 (NOT NULL)**：必須記錄操作類型**列舉約束 (ENUM)**：只能輸入’login’、‘logout’、‘create’、‘update’、‘delete’、’view’六種值**動作定義**：login(登入)、logout(登出)、create(新增)、update(更新)、delete(刪除)、view(檢視)**審計分類**：用於系統操作的分類統計和安全審計 |
| table\_name | **可空值 (NULL)**：操作的資料表名稱為選填**資料型別**：VARCHAR(50)，最多50個字元**表格識別**：記錄資料庫操作涉及的表格名稱**字元集**：支援英文表格名稱 |
| record\_id | **可空值 (NULL)**：操作的記錄識別碼為選填**資料型別**：INT，記錄具體操作的資料記錄ID**記錄追蹤**：與table\_name配合定位具體的操作記錄 |
| description | **可空值 (NULL)**：操作描述為選填**資料型別**：TEXT，最大65,535個字元**字元集**：支援UTF-8編碼，容納中英文描述**詳細說明**：記錄操作的詳細說明和上下文資訊 |
| ip\_address | **可空值 (NULL)**：客戶端IP位址為選填**格式檢查約束 (CHECK)**：支援IPv4和IPv6格式驗證**IPv4格式**：三個點號分隔的四組1-3位數字，如192.168.1.1**IPv6格式**：冒號分隔的十六進位格式**資料型別**：VARCHAR(45)，容納IPv6的完整格式**安全追蹤**：記錄操作來源的網路位址 |
| created\_at | **預設值約束 (DEFAULT CURRENT\_TIMESTAMP)**：系統自動記錄日誌建立時間**資料型別**：DATETIME，格式YYYY-MM-DD HH:mm:ss**時間範圍**：1000-01-01 00:00:00 到 9999-12-31 23:59:59**時序記錄**：確保日誌記錄的時間順序性**高精度**：精確到秒級，支援時間序列分析 |

## 1. 範例資料插入

### 1.1 用戶資料表 (users)

INSERT INTO users (user\_id, name, email, user\_type, status, created\_at, updated\_at) VALUES  
(1, '陳廷威', '40943212@gm.nfu.edu.tw', 'student', 'active', '2024-09-01 10:00:00', '2024-09-01 10:00:00'),  
(2, '李明華', '40943213@gm.nfu.edu.tw', 'student', 'active', '2024-09-01 10:15:00', '2024-09-01 10:15:00'),  
(3, '王小美', '40943214@gm.nfu.edu.tw', 'student', 'active', '2024-09-01 10:30:00', '2024-09-01 10:30:00'),  
(4, '張志明', '40943215@gm.nfu.edu.tw', 'student', 'pending', '2024-09-15 14:20:00', '2024-09-15 14:20:00'),  
(5, '林雅婷', '40943216@gm.nfu.edu.tw', 'student', 'active', '2024-09-01 11:00:00', '2024-09-01 11:00:00'),  
(6, '劉建國', 'T123456@nfu.edu.tw', 'teacher', 'active', '2024-08-15 09:00:00', '2024-08-15 09:00:00'),  
(7, '吳美玲', 'T123457@nfu.edu.tw', 'teacher', 'active', '2024-08-15 09:30:00', '2024-08-15 09:30:00'),  
(8, '黃志偉', 'T123458@nfu.edu.tw', 'teacher', 'active', '2024-08-15 10:00:00', '2024-08-15 10:00:00'),  
(9, '系統管理員', 'admin@nfu.edu.tw', 'admin', 'active', '2024-08-01 08:00:00', '2024-08-01 08:00:00'),  
(10, '趙小琪', '40943217@gm.nfu.edu.tw', 'student', 'suspended', '2024-09-01 12:00:00', '2024-10-01 15:30:00'),  
(11, '陳小明', '40843001@gm.nfu.edu.tw', 'student', 'active', '2023-09-01 10:00:00', '2023-09-01 10:00:00'),  
(12, '李小華', '40843002@gm.nfu.edu.tw', 'student', 'active', '2023-09-01 10:15:00', '2023-09-01 10:15:00'),  
(13, '王小強', '40843003@gm.nfu.edu.tw', 'student', 'active', '2023-09-01 10:30:00', '2023-09-01 10:30:00'),  
(14, '張小美', '41043001@gm.nfu.edu.tw', 'student', 'pending', '2024-09-01 11:00:00', '2024-09-01 11:00:00'),  
(15, '李教授', 'T123459@nfu.edu.tw', 'teacher', 'active', '2024-08-15 09:00:00', '2024-08-15 09:00:00'),  
(16, '陳講師', 'T123460@nfu.edu.tw', 'teacher', 'active', '2024-08-15 09:30:00', '2024-08-15 09:30:00'),  
(17, '林副教授', 'T123461@nfu.edu.tw', 'teacher', 'active', '2024-08-15 10:00:00', '2024-08-15 10:00:00'),  
(18, '王助教', 'T123462@nfu.edu.tw', 'teacher', 'active', '2024-08-15 10:30:00', '2024-08-15 10:30:00'),  
(19, '張教授', 'T123463@nfu.edu.tw', 'teacher', 'active', '2024-08-15 11:00:00', '2024-08-15 11:00:00'),  
(20, '劉副教授', 'T123464@nfu.edu.tw', 'teacher', 'active', '2024-08-15 11:30:00', '2024-08-15 11:30:00'),  
(21, '吳講師', 'T123465@nfu.edu.tw', 'teacher', 'active', '2024-08-15 12:00:00', '2024-08-15 12:00:00');

### 1.2 密碼資料表 (user\_passwords)

INSERT INTO user\_passwords (password\_id, user\_id, password\_hash, created\_at) VALUES  
(1, 1, 'a665a45920422f9d417e4867efdc4fb8a04a1f3fff1fa07e998e86f7f7a27ae3', '2024-09-01 10:00:00'),  
(2, 2, 'b3a8e0e1f9ab1bfe3a36f231f676f78bb30a519d2b21e6c530c0eee8ebb4a5d0', '2024-09-01 10:15:00'),  
(3, 3, 'c2356069e9d1e79ca924378153cfbbfb4d4416b1f99d41a2940bfdb66c5319db', '2024-09-01 10:30:00'),  
(4, 4, 'd4f6ed52a4a98c8c1b78d5b8f7e6f5e4c9d8a7b6e5f4c3d2e1f0a9b8c7d6e5f4', '2024-09-15 14:20:00'),  
(5, 5, 'e7b8c9d0a1f2e3d4c5b6a7f8e9d0c1b2a3f4e5d6c7b8a9f0e1d2c3b4a5f6e7d8', '2024-09-01 11:00:00'),  
(6, 6, 'f1e2d3c4b5a6f7e8d9c0b1a2f3e4d5c6b7a8f9e0d1c2b3a4f5e6d7c8b9a0f1e2', '2024-08-15 09:00:00'),  
(7, 7, 'a2b3c4d5e6f7a8b9c0d1e2f3a4b5c6d7e8f9a0b1c2d3e4f5a6b7c8d9e0f1a2b3', '2024-08-15 09:30:00'),  
(8, 8, 'b4c5d6e7f8a9b0c1d2e3f4a5b6c7d8e9f0a1b2c3d4e5f6a7b8c9d0e1f2a3b4c5', '2024-08-15 10:00:00'),  
(9, 9, 'c6d7e8f9a0b1c2d3e4f5a6b7c8d9e0f1a2b3c4d5e6f7a8b9c0d1e2f3a4b5c6d7', '2024-08-01 08:00:00'),  
(10, 10, 'd8e9f0a1b2c3d4e5f6a7b8c9d0e1f2a3b4c5d6e7f8a9b0c1d2e3f4a5b6c7d8e9', '2024-09-01 12:00:00');

### 1.3 學生資料表 (students)

INSERT INTO students (student\_id, user\_id, department, admission\_year, group\_id) VALUES  
('40943212', 1, '資訊工程系', 2020, 1),  
('40943213', 2, '資訊工程系', 2020, 1),  
('40943214', 3, '資訊工程系', 2020, 2),  
('40943215', 4, '資訊工程系', 2020, NULL),  
('40943216', 5, '電機工程系', 2020, 3),  
('40943217', 10, '資訊工程系', 2020, NULL),  
('40843001', 11, '資訊工程系', 2019, 4),  
('40843002', 12, '電機工程系', 2019, 5),  
('40843003', 13, '資訊管理系', 2019, 6),  
('41043001', 14, '資訊工程系', 2021, NULL);

### 1.4 教師資料表 (teachers)

INSERT INTO teachers (teacher\_id, user\_id, department, title) VALUES  
('T123456', 6, '資訊工程系', '教授'),  
('T123457', 7, '資訊工程系', '副教授'),  
('T123458', 8, '電機工程系', '助理教授'),  
('T123459', 15, '資訊管理系', '教授'),  
('T123460', 16, '資訊工程系', '講師'),  
('T123461', 17, '電機工程系', '副教授'),  
('T123462', 18, '資訊工程系', '助理教授'),  
('T123463', 19, '資訊管理系', '教授'),  
('T123464', 20, '電機工程系', '副教授'),  
('T123465', 21, '資訊工程系', '講師');

### 1.5 學年度資料表 (academic\_years)

INSERT INTO academic\_years (year\_id, year\_name, start\_date, end\_date, status) VALUES  
(1, '2024-2025', '2024-09-01', '2025-06-30', 'active'),  
(2, '2023-2024', '2023-09-01', '2024-06-30', 'archived'),  
(3, '2022-2023', '2022-09-01', '2023-06-30', 'archived'),  
(4, '2021-2022', '2021-09-01', '2022-06-30', 'archived'),  
(5, '2020-2021', '2020-09-01', '2021-06-30', 'archived'),  
(6, '2019-2020', '2019-09-01', '2020-06-30', 'archived'),  
(7, '2025-2026', '2025-09-01', '2026-06-30', 'active'),  
(8, '2018-2019', '2018-09-01', '2019-06-30', 'archived'),  
(9, '2017-2018', '2017-09-01', '2018-06-30', 'archived'),  
(10, '2016-2017', '2016-09-01', '2017-06-30', 'archived');

### 1.6 組別資料表 (groups)

INSERT INTO groups (group\_id, group\_name, project\_title, teacher\_id, year\_id, created\_at, status) VALUES  
(1, '智慧家居組', '基於IoT的智慧家居控制系統', 'T123456', 1, '2024-09-10 14:00:00', 'active'),  
(2, 'AI影像組', '深度學習影像辨識系統', 'T123457', 1, '2024-09-10 14:30:00', 'active'),  
(3, '行動應用組', '校園導覽APP開發', 'T123458', 1, '2024-09-10 15:00:00', 'active'),  
(4, '網路安全組', '網路入侵偵測系統', 'T123456', 2, '2023-09-10 14:00:00', 'completed'),  
(5, '大資料組', '學生學習行為分析平台', 'T123459', 2, '2023-09-10 14:30:00', 'completed'),  
(6, '區塊鏈組', '數位證書管理系統', 'T123460', 1, '2024-09-10 15:30:00', 'active'),  
(7, '遊戲開發組', '教育類遊戲設計', 'T123461', 1, '2024-09-10 16:00:00', 'active'),  
(8, '雲端運算組', '微服務架構電商平台', 'T123457', 1, '2024-09-10 16:30:00', 'suspended'),  
(9, '機器學習組', '智慧推薦系統', 'T123462', 1, '2024-09-10 17:00:00', 'active'),  
(10, 'VR/AR組', '虛擬實境教學系統', 'T123463', 2, '2023-09-10 15:00:00', 'completed');

### 1.7 週報資料表 (weekly\_reports)

INSERT INTO weekly\_reports (report\_id, student\_id, group\_id, week\_number, year\_id, this\_week\_work, next\_week\_plan, submitted\_at, status) VALUES  
(1, '40943212', 1, 1, 1, '完成系統需求分析，研讀相關IoT技術文獻，建立專題規劃時程表', '開始進行系統架構設計，選擇適合的硬體平台', '2024-09-15 23:45:00', 'submitted'),  
(2, '40943213', 1, 1, 1, '協助需求分析，研究使用者介面設計原則，學習前端開發技術', '設計系統UI/UX原型，準備前端開發環境', '2024-09-16 01:20:00', 'late'),  
(3, '40943214', 2, 1, 1, '收集影像辨識相關論文，建立資料集收集計畫，學習深度學習基礎', '開始收集訓練資料，設定深度學習開發環境', '2024-09-15 22:30:00', 'submitted'),  
(4, '40943216', 3, 1, 1, '進行校園場地勘查，設計APP功能架構，學習行動應用開發技術', '完成APP介面設計，開始核心功能開發', '2024-09-15 20:15:00', 'submitted'),  
(5, '40943212', 1, 2, 1, '完成系統架構設計圖，選定Arduino和Raspberry Pi作為主要硬體', '進行硬體採購，開始感測器整合測試', '2024-09-22 23:30:00', 'submitted'),  
(6, '40943213', 1, 2, 1, '完成UI原型設計，建立前端開發環境，學習React Native框架', '開始前端介面開發，整合後端API接口', '2024-09-23 02:10:00', 'late'),  
(7, '40943214', 2, 2, 1, '收集了1000張訓練影像，建立資料標註工具，完成開發環境設定', '完成資料標註工作，開始訓練初步模型', '2024-09-22 21:45:00', 'submitted'),  
(8, '40943216', 3, 2, 1, '完成APP主要介面設計，實作基本導覽功能，整合地圖API', '新增互動功能，優化使用者體驗設計', '2024-09-22 19:20:00', 'submitted'),  
(9, '40943212', 1, 3, 1, '完成感測器採購與基本測試，建立硬體連接架構，進行初步整合', '完成所有感測器整合，開始軟體控制程式開發', '2024-09-29 23:00:00', 'submitted'),  
(10, '40943213', 1, 3, 1, '持續進行前端開發，完成主要操作介面，開始後端API整合測試', '完善前端功能，進行系統整合測試', '2024-09-30 01:30:00', 'late');

### 1.8 評分資料表 (evaluations)

INSERT INTO evaluations (evaluation\_id, report\_id, teacher\_id, score, comments, evaluated\_at, status) VALUES  
(1, 1, 'T123456', 85, '需求分析做得很詳細，時程規劃合理，建議加強技術可行性評估', '2024-09-17 10:30:00', 'completed'),  
(2, 2, 'T123456', 78, 'UI設計有創意，但需要考慮實用性，建議多研究使用者體驗設計原則', '2024-09-17 11:00:00', 'completed'),  
(3, 3, 'T123457', 90, '文獻收集充分，計畫完整，對深度學習概念掌握良好', '2024-09-17 14:20:00', 'completed'),  
(4, 4, 'T123458', 82, '實地勘查做得仔細，功能規劃完整，建議增加技術難度挑戰', '2024-09-17 15:45:00', 'completed'),  
(5, 5, 'T123456', 88, '架構設計合理，硬體選擇恰當，期待後續實作成果', '2024-09-24 09:15:00', 'completed'),  
(6, 6, 'T123456', 75, '前端開發進度稍慢，建議加強時間管理，多與組員協調', '2024-09-24 09:45:00', 'completed'),  
(7, 7, 'T123457', 92, '資料收集效率高，標註工具設計實用，展現良好的解決問題能力', '2024-09-24 13:30:00', 'completed'),  
(8, 8, 'T123458', 86, 'APP介面設計美觀，功能實作順利，建議增加更多互動元素', '2024-09-24 16:20:00', 'completed'),  
(9, 9, 'T123456', 87, '硬體整合進展良好，技術問題解決能力強，期待軟體開發成果', '2024-10-01 10:00:00', 'completed'),  
(10, 10, 'T123456', 73, '開發進度需要加快，建議更有效率的工作方式，加強團隊合作', '2024-10-01 10:30:00', 'completed');

### 1.9 檔案資料表 (files)

INSERT INTO files (file\_id, report\_id, filename, file\_path, file\_size, file\_type, uploaded\_at) VALUES  
(1, 1, '系統需求規格書.pdf', '/uploads/2024/09/15/system\_requirements\_spec.pdf', 2048576, 'PDF文件', '2024-09-15 23:45:30'),  
(2, 1, '專題時程規劃.xlsx', '/uploads/2024/09/15/project\_timeline.xlsx', 1024768, 'Excel文件', '2024-09-15 23:46:00'),  
(3, 2, 'UI設計原型.png', '/uploads/2024/09/16/ui\_prototype.png', 3145728, '圖片', '2024-09-16 01:20:30'),  
(4, 3, '深度學習文獻整理.docx', '/uploads/2024/09/15/dl\_literature\_review.docx', 1536000, 'Word文件', '2024-09-15 22:30:45'),  
(5, 4, '校園場地勘查報告.pdf', '/uploads/2024/09/15/campus\_survey\_report.pdf', 4194304, 'PDF文件', '2024-09-15 20:15:20'),  
(6, 5, '系統架構圖.png', '/uploads/2024/09/22/system\_architecture.png', 2097152, '圖片', '2024-09-22 23:30:15'),  
(7, 6, '前端開發進度截圖.png', '/uploads/2024/09/23/frontend\_progress.png', 1572864, '圖片', '2024-09-23 02:10:25'),  
(8, 7, '訓練資料樣本.zip', '/uploads/2024/09/22/training\_samples.zip', 10485760, '壓縮檔', '2024-09-22 21:45:50'),  
(9, 8, 'APP介面設計.sketch', '/uploads/2024/09/22/app\_interface\_design.sketch', 5242880, '設計檔', '2024-09-22 19:20:35'),  
(10, 9, '感測器測試報告.pdf', '/uploads/2024/09/29/sensor\_test\_report.pdf', 3670016, 'PDF文件', '2024-09-29 23:00:40');

### 1.10 通知資料表 (notifications)

INSERT INTO notifications (notification\_id, user\_id, title, content, type, is\_read, created\_at) VALUES  
(1, 1, '週報評分完成', '您第1週的週報已完成評分，獲得85分', 'evaluation', TRUE, '2024-09-17 10:30:30'),  
(2, 2, '週報評分完成', '您第1週的週報已完成評分，獲得78分', 'evaluation', FALSE, '2024-09-17 11:00:30'),  
(3, 3, '週報評分完成', '您第1週的週報已完成評分，獲得90分', 'evaluation', TRUE, '2024-09-17 14:20:30'),  
(4, 4, '帳號審核通過', '您的帳號已通過審核，現在可以正常使用系統功能', 'system', FALSE, '2024-09-20 09:00:00'),  
(5, 1, '週報提交提醒', '第4週週報提交截止日即將到來，請及時提交', 'reminder', FALSE, '2024-10-05 08:00:00'),  
(6, 2, '週報評分完成', '您第2週的週報已完成評分，獲得75分', 'evaluation', FALSE, '2024-09-24 09:45:30'),  
(7, 5, '組別分配通知', '您已被分配到行動應用組，請與組員聯繫', 'system', TRUE, '2024-09-10 16:00:00'),  
(8, 6, '新學期開始', '2024-2025學年度已開始，請各位老師協助學生組別分配', 'announcement', TRUE, '2024-09-01 08:00:00'),  
(9, 1, '系統維護通知', '系統將於本週日凌晨2:00-4:00進行維護，屆時無法使用', 'system', FALSE, '2024-10-03 18:00:00'),  
(10, 3, '優秀作品展示', '您的專題作品表現優異，邀請參加期末成果展示', 'announcement', FALSE, '2024-10-01 15:30:00');

### 1.11 審核資料表 (user\_approvals)

INSERT INTO user\_approvals (approval\_id, user\_id, admin\_id, request\_type, status, request\_data, admin\_notes, requested\_at, processed\_at) VALUES  
(1, 4, 9, 'registration', 'approved', '{"student\_id":"40943215","department":"資訊工程系"}', '學號格式正確，科系確認無誤', '2024-09-15 14:20:00', '2024-09-20 09:00:00'),  
(2, 14, 9, 'registration', 'pending', '{"student\_id":"41043001","department":"資訊工程系"}', NULL, '2024-09-01 11:00:00', NULL),  
(3, 10, 9, 'account\_recovery', 'rejected', '{"reason":"違反使用規範"}', '經查證確實有違規行為，暫不恢復', '2024-10-01 10:00:00', '2024-10-01 15:30:00'),  
(4, 5, 9, 'role\_change', 'approved', '{"from":"student","to":"student","group\_change":"3"}', '組別異動申請合理', '2024-09-25 14:30:00', '2024-09-26 10:15:00'),  
(5, 11, 9, 'registration', 'approved', '{"student\_id":"40843001","department":"資訊工程系"}', '舊生重新註冊，資料確認無誤', '2024-09-01 09:00:00', '2024-09-01 16:00:00'),  
(6, 15, 9, 'registration', 'approved', '{"teacher\_id":"T123459","department":"資訊管理系"}', '新進教師，資格審核通過', '2024-08-10 10:00:00', '2024-08-15 09:00:00'),  
(7, 20, 9, 'role\_change', 'pending', '{"change\_title":"副教授"}', NULL, '2024-10-01 14:00:00', NULL),  
(8, 12, 9, 'registration', 'approved', '{"student\_id":"40843002","department":"電機工程系"}', '轉系生，新科系確認', '2024-09-05 11:30:00', '2024-09-06 14:20:00'),  
(9, 16, 9, 'account\_recovery', 'approved', '{"reason":"密碼遺失"}', '身份驗證通過，密碼已重設', '2024-09-20 16:45:00', '2024-09-21 09:30:00'),  
(10, 21, 9, 'registration', 'pending', '{"teacher\_id":"T123465","department":"資訊工程系"}', NULL, '2024-10-02 13:20:00', NULL);

### 1.12 系統日誌資料表 (system\_logs)

INSERT INTO system\_logs (log\_id, user\_id, action\_type, table\_name, record\_id, description, ip\_address, created\_at) VALUES  
(1, 1, 'login', NULL, NULL, '學生用戶登入系統', '192.168.1.100', '2024-09-15 23:30:00'),  
(2, 1, 'create', 'weekly\_reports', 1, '提交第1週週報', '192.168.1.100', '2024-09-15 23:45:00'),  
(3, 6, 'login', NULL, NULL, '教師用戶登入系統', '192.168.1.50', '2024-09-17 10:00:00'),  
(4, 6, 'create', 'evaluations', 1, '評分週報ID:1', '192.168.1.50', '2024-09-17 10:30:00'),  
(5, 9, 'login', NULL, NULL, '管理員登入系統', '192.168.1.10', '2024-09-20 08:45:00'),  
(6, 9, 'update', 'user\_approvals', 1, '審核用戶註冊申請', '192.168.1.10', '2024-09-20 09:00:00'),  
(7, 2, 'login', NULL, NULL, '學生用戶登入系統', '192.168.1.101', '2024-09-23 01:50:00'),  
(8, 2, 'create', 'weekly\_reports', 6, '提交第2週週報', '192.168.1.101', '2024-09-23 02:10:00'),  
(9, 3, 'view', 'evaluations', 3, '查看評分結果', '192.168.1.102', '2024-09-24 20:15:00'),  
(10, 7, 'login', NULL, NULL, '教師用戶登入系統', '192.168.1.51', '2024-09-24 13:00:00');

## 2. VIEW實作與輸出結果

### 2.1 用戶基本資訊檢視 (view\_user\_basic\_info)

#### VIEW建立語句

CREATE VIEW view\_user\_basic\_info AS  
SELECT   
 u.user\_id,  
 u.name,  
 u.email,  
 u.user\_type,  
 u.status,  
 u.created\_at,  
 CASE   
 WHEN u.user\_type = 'student' THEN s.student\_id  
 WHEN u.user\_type = 'teacher' THEN t.teacher\_id  
 ELSE NULL  
 END AS role\_id,  
 CASE   
 WHEN u.user\_type = 'student' THEN s.department  
 WHEN u.user\_type = 'teacher' THEN t.department  
 ELSE NULL  
 END AS department  
FROM users u  
LEFT JOIN students s ON u.user\_id = s.user\_id  
LEFT JOIN teachers t ON u.user\_id = t.user\_id;

#### 輸出結果

| user\_id | name | email | user\_type | status | created\_at | role\_id | department |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 陳廷威 | 40943212@gm.nfu.edu.tw | student | active | 2024-09-01 10:00:00 | 40943212 | 資訊工程系 |
| 2 | 李明華 | 40943213@gm.nfu.edu.tw | student | active | 2024-09-01 10:15:00 | 40943213 | 資訊工程系 |
| 3 | 王小美 | 40943214@gm.nfu.edu.tw | student | active | 2024-09-01 10:30:00 | 40943214 | 資訊工程系 |
| 6 | 劉建國 | T123456@nfu.edu.tw | teacher | active | 2024-08-15 09:00:00 | T123456 | 資訊工程系 |
| 7 | 吳美玲 | T123457@nfu.edu.tw | teacher | active | 2024-08-15 09:30:00 | T123457 | 資訊工程系 |
| 8 | 黃志偉 | T123458@nfu.edu.tw | teacher | active | 2024-08-15 10:00:00 | T123458 | 電機工程系 |
| 9 | 系統管理員 | admin@nfu.edu.tw | admin | active | 2024-08-01 08:00:00 | NULL | NULL |

### 2.2 學生完整資訊檢視 (view\_student\_full\_info)

#### VIEW建立語句

CREATE VIEW view\_student\_full\_info AS  
SELECT   
 s.student\_id,  
 u.name as student\_name,  
 u.email,  
 s.department,  
 s.admission\_year,  
 u.status,  
 g.group\_id,  
 g.group\_name,  
 g.project\_title,  
 t.teacher\_id,  
 tu.name as teacher\_name,  
 ay.year\_name,  
 ay.status as year\_status  
FROM students s  
JOIN users u ON s.user\_id = u.user\_id  
LEFT JOIN groups g ON s.group\_id = g.group\_id  
LEFT JOIN academic\_years ay ON g.year\_id = ay.year\_id  
LEFT JOIN teachers t ON g.teacher\_id = t.teacher\_id  
LEFT JOIN users tu ON t.user\_id = tu.user\_id;

#### 輸出結果

| student\_id | student\_name | email | department | admission\_year | status | group\_id | group\_name | project\_title | teacher\_id | teacher\_name | year\_name | year\_status |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 40943212 | 陳廷威 | 40943212@gm.nfu.edu.tw | 資訊工程系 | 2020 | active | 1 | 智慧家居組 | 基於IoT的智慧家居控制系統 | T123456 | 劉建國 | 2024-2025 | active |
| 40943213 | 李明華 | 40943213@gm.nfu.edu.tw | 資訊工程系 | 2020 | active | 1 | 智慧家居組 | 基於IoT的智慧家居控制系統 | T123456 | 劉建國 | 2024-2025 | active |
| 40943214 | 王小美 | 40943214@gm.nfu.edu.tw | 資訊工程系 | 2020 | active | 2 | AI影像組 | 深度學習影像辨識系統 | T123457 | 吳美玲 | 2024-2025 | active |
| 40943216 | 林雅婷 | 40943216@gm.nfu.edu.tw | 電機工程系 | 2020 | active | 3 | 行動應用組 | 校園導覽APP開發 | T123458 | 黃志偉 | 2024-2025 | active |
| 40943215 | 張志明 | 40943215@gm.nfu.edu.tw | 資訊工程系 | 2020 | pending | NULL | NULL | NULL | NULL | NULL | NULL | NULL |

### 2.3 學生週報統計檢視 (view\_student\_report\_stats)

#### VIEW建立語句

CREATE VIEW view\_student\_report\_stats AS  
SELECT   
 s.student\_id,  
 u.name as student\_name,  
 s.department,  
 g.group\_name,  
 ay.year\_name,  
 COUNT(wr.report\_id) as total\_reports,  
 SUM(CASE WHEN wr.status = 'submitted' THEN 1 ELSE 0 END) as submitted\_count,  
 SUM(CASE WHEN wr.status = 'late' THEN 1 ELSE 0 END) as late\_count,  
 SUM(CASE WHEN wr.status = 'missing' THEN 1 ELSE 0 END) as missing\_count,  
 ROUND(AVG(e.score), 2) as avg\_score  
FROM students s  
JOIN users u ON s.user\_id = u.user\_id  
LEFT JOIN groups g ON s.group\_id = g.group\_id  
LEFT JOIN academic\_years ay ON g.year\_id = ay.year\_id  
LEFT JOIN weekly\_reports wr ON s.student\_id = wr.student\_id  
LEFT JOIN evaluations e ON wr.report\_id = e.report\_id  
WHERE u.status = 'active'  
GROUP BY s.student\_id, u.name, s.department, g.group\_name, ay.year\_name;

#### 輸出結果

| student\_id | student\_name | department | group\_name | year\_name | total\_reports | submitted\_count | late\_count | missing\_count | avg\_score |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 40943212 | 陳廷威 | 資訊工程系 | 智慧家居組 | 2024-2025 | 3 | 3 | 0 | 0 | 86.67 |
| 40943213 | 李明華 | 資訊工程系 | 智慧家居組 | 2024-2025 | 3 | 1 | 2 | 0 | 75.33 |
| 40943214 | 王小美 | 資訊工程系 | AI影像組 | 2024-2025 | 2 | 2 | 0 | 0 | 91.00 |
| 40943216 | 林雅婷 | 電機工程系 | 行動應用組 | 2024-2025 | 2 | 2 | 0 | 0 | 84.00 |

### 2.4 教師工作負荷檢視 (view\_teacher\_workload)

#### VIEW建立語句

CREATE VIEW view\_teacher\_workload AS  
SELECT   
 t.teacher\_id,  
 u.name as teacher\_name,  
 t.department,  
 t.title,  
 ay.year\_name,  
 COUNT(DISTINCT g.group\_id) as active\_groups,  
 COUNT(DISTINCT s.student\_id) as total\_students,  
 COUNT(DISTINCT wr.report\_id) as total\_reports,  
 COUNT(DISTINCT e.evaluation\_id) as evaluated\_reports,  
 CASE   
 WHEN COUNT(DISTINCT wr.report\_id) > 0   
 THEN ROUND(COUNT(DISTINCT e.evaluation\_id) \* 100.0 / COUNT(DISTINCT wr.report\_id), 2)  
 ELSE 0  
 END as evaluation\_rate  
FROM teachers t  
JOIN users u ON t.user\_id = u.user\_id  
LEFT JOIN groups g ON t.teacher\_id = g.teacher\_id  
LEFT JOIN academic\_years ay ON g.year\_id = ay.year\_id  
LEFT JOIN students s ON g.group\_id = s.group\_id  
LEFT JOIN weekly\_reports wr ON s.student\_id = wr.student\_id  
LEFT JOIN evaluations e ON wr.report\_id = e.report\_id AND e.teacher\_id = t.teacher\_id  
WHERE u.status = 'active' AND (ay.status = 'active' OR ay.status IS NULL)  
GROUP BY t.teacher\_id, u.name, t.department, t.title, ay.year\_name;

#### 輸出結果

| teacher\_id | teacher\_name | department | title | year\_name | active\_groups | total\_students | total\_reports | evaluated\_reports | evaluation\_rate |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| T123456 | 劉建國 | 資訊工程系 | 教授 | 2024-2025 | 1 | 2 | 6 | 6 | 100.00 |
| T123457 | 吳美玲 | 資訊工程系 | 副教授 | 2024-2025 | 1 | 1 | 2 | 2 | 100.00 |
| T123458 | 黃志偉 | 電機工程系 | 助理教授 | 2024-2025 | 1 | 1 | 2 | 2 | 100.00 |

### 2.5 組別進度統計檢視 (view\_group\_progress\_stats)

#### VIEW建立語句

CREATE VIEW view\_group\_progress\_stats AS  
SELECT   
 g.group\_id,  
 g.group\_name,  
 g.project\_title,  
 ay.year\_name,  
 tu.name as teacher\_name,  
 COUNT(DISTINCT s.student\_id) as member\_count,  
 COUNT(DISTINCT wr.report\_id) as total\_reports,  
 COUNT(DISTINCT e.evaluation\_id) as evaluated\_reports,  
 ROUND(AVG(e.score), 2) as avg\_score,  
 MAX(wr.week\_number) as latest\_week,  
 COUNT(DISTINCT wr.week\_number) as reported\_weeks  
FROM groups g  
JOIN academic\_years ay ON g.year\_id = ay.year\_id  
LEFT JOIN teachers t ON g.teacher\_id = t.teacher\_id  
LEFT JOIN users tu ON t.user\_id = tu.user\_id  
LEFT JOIN students s ON g.group\_id = s.group\_id  
LEFT JOIN weekly\_reports wr ON s.student\_id = wr.student\_id  
LEFT JOIN evaluations e ON wr.report\_id = e.report\_id  
WHERE g.status = 'active'  
GROUP BY g.group\_id, g.group\_name, g.project\_title, ay.year\_name, tu.name;

#### 輸出結果

| group\_id | group\_name | project\_title | year\_name | teacher\_name | member\_count | total\_reports | evaluated\_reports | avg\_score | latest\_week | reported\_weeks |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 智慧家居組 | 基於IoT的智慧家居控制系統 | 2024-2025 | 劉建國 | 2 | 6 | 6 | 80.83 | 3 | 3 |
| 2 | AI影像組 | 深度學習影像辨識系統 | 2024-2025 | 吳美玲 | 1 | 2 | 2 | 91.00 | 2 | 2 |
| 3 | 行動應用組 | 校園導覽APP開發 | 2024-2025 | 黃志偉 | 1 | 2 | 2 | 84.00 | 2 | 2 |

### 2.6 週報提交狀況檢視 (view\_weekly\_report\_submission\_status)

#### VIEW建立語句

CREATE VIEW view\_weekly\_report\_submission\_status AS  
SELECT   
 ay.year\_name,  
 wr.week\_number,  
 COUNT(\*) as total\_submissions,  
 SUM(CASE WHEN wr.status = 'submitted' THEN 1 ELSE 0 END) as on\_time,  
 SUM(CASE WHEN wr.status = 'late' THEN 1 ELSE 0 END) as late,  
 SUM(CASE WHEN wr.status = 'missing' THEN 1 ELSE 0 END) as missing,  
 ROUND(AVG(e.score), 2) as avg\_score,  
 COUNT(e.evaluation\_id) as evaluated\_count  
FROM weekly\_reports wr  
JOIN academic\_years ay ON wr.year\_id = ay.year\_id  
LEFT JOIN evaluations e ON wr.report\_id = e.report\_id  
GROUP BY ay.year\_name, wr.week\_number  
ORDER BY ay.year\_name, wr.week\_number;

#### 輸出結果

| year\_name | week\_number | total\_submissions | on\_time | late | missing | avg\_score | evaluated\_count |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 2024-2025 | 1 | 4 | 3 | 1 | 0 | 83.75 | 4 |
| 2024-2025 | 2 | 4 | 3 | 1 | 0 | 81.75 | 4 |
| 2024-2025 | 3 | 2 | 2 | 0 | 0 | 80.00 | 2 |

### 2.7 評分統計檢視 (view\_evaluation\_statistics)

#### VIEW建立語句

CREATE VIEW view\_evaluation\_statistics AS  
SELECT   
 t.teacher\_id,  
 tu.name as teacher\_name,  
 ay.year\_name,  
 COUNT(e.evaluation\_id) as total\_evaluations,  
 ROUND(AVG(e.score), 2) as avg\_score,  
 MIN(e.score) as min\_score,  
 MAX(e.score) as max\_score,  
 COUNT(CASE WHEN e.score >= 90 THEN 1 END) as excellent\_count,  
 COUNT(CASE WHEN e.score >= 80 AND e.score < 90 THEN 1 END) as good\_count,  
 COUNT(CASE WHEN e.score >= 70 AND e.score < 80 THEN 1 END) as fair\_count,  
 COUNT(CASE WHEN e.score < 70 THEN 1 END) as poor\_count  
FROM teachers t  
JOIN users tu ON t.user\_id = tu.user\_id  
JOIN groups g ON t.teacher\_id = g.teacher\_id  
JOIN academic\_years ay ON g.year\_id = ay.year\_id  
JOIN students s ON g.group\_id = s.group\_id  
JOIN weekly\_reports wr ON s.student\_id = wr.student\_id  
JOIN evaluations e ON wr.report\_id = e.report\_id  
WHERE tu.status = 'active'  
GROUP BY t.teacher\_id, tu.name, ay.year\_name  
ORDER BY t.teacher\_id, ay.year\_name;

#### 輸出結果

| teacher\_id | teacher\_name | year\_name | total\_evaluations | avg\_score | min\_score | max\_score | excellent\_count | good\_count | fair\_count | poor\_count |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| T123456 | 劉建國 | 2024-2025 | 6 | 80.83 | 73 | 88 | 0 | 4 | 2 | 0 |
| T123457 | 吳美玲 | 2024-2025 | 2 | 91.00 | 90 | 92 | 2 | 0 | 0 | 0 |
| T123458 | 黃志偉 | 2024-2025 | 2 | 84.00 | 82 | 86 | 0 | 2 | 0 | 0 |

### 2.8 學生成績分析檢視 (view\_student\_grade\_analysis)

#### VIEW建立語句

CREATE VIEW view\_student\_grade\_analysis AS  
SELECT   
 s.student\_id,  
 su.name as student\_name,  
 s.department,  
 g.group\_name,  
 ay.year\_name,  
 COUNT(e.evaluation\_id) as total\_evaluations,  
 ROUND(AVG(e.score), 2) as avg\_score,  
 MIN(e.score) as min\_score,  
 MAX(e.score) as max\_score,  
 CASE   
 WHEN AVG(e.score) >= 90 THEN '優秀'  
 WHEN AVG(e.score) >= 80 THEN '良好'  
 WHEN AVG(e.score) >= 70 THEN '及格'  
 ELSE '待加強'  
 END as grade\_level,  
 COUNT(CASE WHEN e.score >= 80 THEN 1 END) as high\_score\_count  
FROM students s  
JOIN users su ON s.user\_id = su.user\_id  
JOIN groups g ON s.group\_id = g.group\_id  
JOIN academic\_years ay ON g.year\_id = ay.year\_id  
JOIN weekly\_reports wr ON s.student\_id = wr.student\_id  
JOIN evaluations e ON wr.report\_id = e.report\_id  
GROUP BY s.student\_id, su.name, s.department, g.group\_name, ay.year\_name  
ORDER BY avg\_score DESC;

#### 輸出結果

| student\_id | student\_name | department | group\_name | year\_name | total\_evaluations | avg\_score | min\_score | max\_score | grade\_level | high\_score\_count |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 40943214 | 王小美 | 資訊工程系 | AI影像組 | 2024-2025 | 2 | 91.00 | 90 | 92 | 優秀 | 2 |
| 40943212 | 陳廷威 | 資訊工程系 | 智慧家居組 | 2024-2025 | 3 | 86.67 | 85 | 88 | 良好 | 3 |
| 40943216 | 林雅婷 | 電機工程系 | 行動應用組 | 2024-2025 | 2 | 84.00 | 82 | 86 | 良好 | 2 |
| 40943213 | 李明華 | 資訊工程系 | 智慧家居組 | 2024-2025 | 3 | 75.33 | 73 | 78 | 及格 | 0 |

### 2.9 用戶通知統計檢視 (view\_user\_notification\_summary)

#### VIEW建立語句

CREATE VIEW view\_user\_notification\_summary AS  
SELECT   
 u.user\_id,  
 u.name,  
 u.user\_type,  
 COUNT(n.notification\_id) as total\_notifications,  
 SUM(CASE WHEN n.is\_read = FALSE THEN 1 ELSE 0 END) as unread\_count,  
 SUM(CASE WHEN n.type = 'system' THEN 1 ELSE 0 END) as system\_count,  
 SUM(CASE WHEN n.type = 'evaluation' THEN 1 ELSE 0 END) as evaluation\_count,  
 SUM(CASE WHEN n.type = 'reminder' THEN 1 ELSE 0 END) as reminder\_count,  
 SUM(CASE WHEN n.type = 'announcement' THEN 1 ELSE 0 END) as announcement\_count,  
 MAX(n.created\_at) as latest\_notification  
FROM users u  
LEFT JOIN notifications n ON u.user\_id = n.user\_id  
WHERE u.status = 'active'  
GROUP BY u.user\_id, u.name, u.user\_type  
ORDER BY unread\_count DESC;

#### 輸出結果

| user\_id | name | user\_type | total\_notifications | unread\_count | system\_count | evaluation\_count | reminder\_count | announcement\_count | latest\_notification |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 陳廷威 | student | 3 | 2 | 1 | 1 | 1 | 0 | 2024-10-05 08:00:00 |
| 2 | 李明華 | student | 2 | 2 | 0 | 2 | 0 | 0 | 2024-09-24 09:45:30 |
| 3 | 王小美 | student | 2 | 1 | 0 | 1 | 0 | 1 | 2024-10-01 15:30:00 |
| 4 | 張志明 | student | 1 | 1 | 1 | 0 | 0 | 0 | 2024-09-20 09:00:00 |
| 5 | 林雅婷 | student | 1 | 0 | 1 | 0 | 0 | 0 | 2024-09-10 16:00:00 |
| 6 | 劉建國 | teacher | 1 | 0 | 0 | 0 | 0 | 1 | 2024-09-01 08:00:00 |

### 2.10 學年度總覽統計檢視 (view\_academic\_year\_overview)

#### VIEW建立語句

CREATE VIEW view\_academic\_year\_overview AS  
SELECT   
 ay.year\_id,  
 ay.year\_name,  
 ay.status,  
 ay.start\_date,  
 ay.end\_date,  
 COUNT(DISTINCT g.group\_id) as total\_groups,  
 COUNT(DISTINCT s.student\_id) as total\_students,  
 COUNT(DISTINCT t.teacher\_id) as total\_teachers,  
 COUNT(DISTINCT wr.report\_id) as total\_reports,  
 COUNT(DISTINCT e.evaluation\_id) as total\_evaluations,  
 ROUND(AVG(e.score), 2) as avg\_score  
FROM academic\_years ay  
LEFT JOIN groups g ON ay.year\_id = g.year\_id  
LEFT JOIN students s ON g.group\_id = s.group\_id  
LEFT JOIN teachers t ON g.teacher\_id = t.teacher\_id  
LEFT JOIN weekly\_reports wr ON ay.year\_id = wr.year\_id  
LEFT JOIN evaluations e ON wr.report\_id = e.report\_id  
GROUP BY ay.year\_id, ay.year\_name, ay.status, ay.start\_date, ay.end\_date  
ORDER BY ay.start\_date DESC;

#### 輸出結果

| year\_id | year\_name | status | start\_date | end\_date | total\_groups | total\_students | total\_teachers | total\_reports | total\_evaluations | avg\_score |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 2024-2025 | active | 2024-09-01 | 2025-06-30 | 6 | 4 | 3 | 10 | 10 | 83.40 |
| 2 | 2023-2024 | archived | 2023-09-01 | 2024-06-30 | 3 | 2 | 2 | 0 | 0 | NULL |
| 7 | 2025-2026 | active | 2025-09-01 | 2026-06-30 | 0 | 0 | 0 | 0 | 0 | NULL |

### 2.11 管理員儀表板檢視 (view\_admin\_dashboard)

#### VIEW建立語句

CREATE VIEW view\_admin\_dashboard AS  
SELECT   
 'users' as metric\_type,  
 COUNT(\*) as total\_count,  
 SUM(CASE WHEN status = 'pending' THEN 1 ELSE 0 END) as pending\_count,  
 SUM(CASE WHEN status = 'active' THEN 1 ELSE 0 END) as active\_count,  
 SUM(CASE WHEN created\_at >= DATE\_SUB(NOW(), INTERVAL 7 DAY) THEN 1 ELSE 0 END) as recent\_count  
FROM users  
  
UNION ALL  
  
SELECT   
 'weekly\_reports' as metric\_type,  
 COUNT(\*) as total\_count,  
 SUM(CASE WHEN status = 'missing' THEN 1 ELSE 0 END) as pending\_count,  
 SUM(CASE WHEN status = 'submitted' THEN 1 ELSE 0 END) as active\_count,  
 SUM(CASE WHEN submitted\_at >= DATE\_SUB(NOW(), INTERVAL 7 DAY) THEN 1 ELSE 0 END) as recent\_count  
FROM weekly\_reports  
  
UNION ALL  
  
SELECT   
 'evaluations' as metric\_type,  
 COUNT(\*) as total\_count,  
 0 as pending\_count,  
 COUNT(\*) as active\_count,  
 SUM(CASE WHEN evaluated\_at >= DATE\_SUB(NOW(), INTERVAL 7 DAY) THEN 1 ELSE 0 END) as recent\_count  
FROM evaluations  
  
UNION ALL  
  
SELECT   
 'user\_approvals' as metric\_type,  
 COUNT(\*) as total\_count,  
 SUM(CASE WHEN status = 'pending' THEN 1 ELSE 0 END) as pending\_count,  
 SUM(CASE WHEN status = 'approved' THEN 1 ELSE 0 END) as active\_count,  
 SUM(CASE WHEN requested\_at >= DATE\_SUB(NOW(), INTERVAL 7 DAY) THEN 1 ELSE 0 END) as recent\_count  
FROM user\_approvals;

#### 輸出結果

| metric\_type | total\_count | pending\_count | active\_count | recent\_count |
| --- | --- | --- | --- | --- |
| users | 21 | 3 | 17 | 14 |
| weekly\_reports | 10 | 0 | 10 | 10 |
| evaluations | 10 | 0 | 10 | 10 |
| user\_approvals | 10 | 3 | 6 | 5 |

### 2.12 學生當前狀態檢視 (view\_student\_current\_status)

#### VIEW建立語句

CREATE VIEW view\_student\_current\_status AS  
SELECT   
 s.student\_id,  
 su.name as student\_name,  
 s.department,  
 g.group\_name,  
 tu.name as teacher\_name,  
 ay.year\_name,  
 latest\_report.week\_number as latest\_week,  
 latest\_report.submitted\_at as last\_submission,  
 latest\_eval.score as latest\_score,  
 CASE   
 WHEN latest\_report.week\_number IS NULL THEN '尚未開始'  
 WHEN latest\_eval.score IS NULL THEN '待評分'  
 WHEN latest\_eval.score >= 80 THEN '表現良好'  
 WHEN latest\_eval.score >= 60 THEN '表現普通'  
 ELSE '需要關注'  
 END as status\_summary  
FROM students s  
JOIN users su ON s.user\_id = su.user\_id  
LEFT JOIN groups g ON s.group\_id = g.group\_id  
LEFT JOIN academic\_years ay ON g.year\_id = ay.year\_id  
LEFT JOIN teachers t ON g.teacher\_id = t.teacher\_id  
LEFT JOIN users tu ON t.user\_id = tu.user\_id  
LEFT JOIN (  
 SELECT   
 student\_id,  
 MAX(week\_number) as week\_number,  
 MAX(submitted\_at) as submitted\_at  
 FROM weekly\_reports   
 GROUP BY student\_id  
) latest\_report ON s.student\_id = latest\_report.student\_id  
LEFT JOIN weekly\_reports wr ON s.student\_id = wr.student\_id   
 AND wr.week\_number = latest\_report.week\_number  
LEFT JOIN evaluations latest\_eval ON wr.report\_id = latest\_eval.report\_id  
WHERE su.status = 'active' AND (ay.status = 'active' OR ay.status IS NULL);

#### 輸出結果

| student\_id | student\_name | department | group\_name | teacher\_name | year\_name | latest\_week | last\_submission | latest\_score | status\_summary |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 40943212 | 陳廷威 | 資訊工程系 | 智慧家居組 | 劉建國 | 2024-2025 | 3 | 2024-09-29 23:00:00 | 87 | 表現良好 |
| 40943213 | 李明華 | 資訊工程系 | 智慧家居組 | 劉建國 | 2024-2025 | 3 | 2024-09-30 01:30:00 | 73 | 表現普通 |
| 40943214 | 王小美 | 資訊工程系 | AI影像組 | 吳美玲 | 2024-2025 | 2 | 2024-09-22 21:45:00 | 92 | 表現良好 |

### 2.13 教師工作台檢視 (view\_teacher\_dashboard)

#### VIEW建立語句

CREATE VIEW view\_teacher\_dashboard AS  
SELECT   
 t.teacher\_id,  
 tu.name as teacher\_name,  
 COUNT(DISTINCT g.group\_id) as active\_groups,  
 COUNT(DISTINCT s.student\_id) as total\_students,  
 COUNT(DISTINCT pending\_reports.report\_id) as pending\_evaluations,  
 COUNT(DISTINCT this\_week\_reports.report\_id) as this\_week\_submissions,  
 ROUND(AVG(recent\_eval.score), 2) as recent\_avg\_score  
FROM teachers t  
JOIN users tu ON t.user\_id = tu.user\_id  
LEFT JOIN groups g ON t.teacher\_id = g.teacher\_id AND g.status = 'active'  
LEFT JOIN students s ON g.group\_id = s.group\_id  
LEFT JOIN weekly\_reports pending\_reports ON s.student\_id = pending\_reports.student\_id  
 AND NOT EXISTS (  
 SELECT 1 FROM evaluations e   
 WHERE e.report\_id = pending\_reports.report\_id  
 )  
LEFT JOIN weekly\_reports this\_week\_reports ON s.student\_id = this\_week\_reports.student\_id  
 AND WEEK(this\_week\_reports.submitted\_at) = WEEK(NOW())  
 AND YEAR(this\_week\_reports.submitted\_at) = YEAR(NOW())  
LEFT JOIN weekly\_reports recent\_wr ON s.student\_id = recent\_wr.student\_id  
 AND recent\_wr.submitted\_at >= DATE\_SUB(NOW(), INTERVAL 30 DAY)  
LEFT JOIN evaluations recent\_eval ON recent\_wr.report\_id = recent\_eval.report\_id  
 AND recent\_eval.teacher\_id = t.teacher\_id  
WHERE tu.status = 'active'  
GROUP BY t.teacher\_id, tu.name;

#### 輸出結果

| teacher\_id | teacher\_name | active\_groups | total\_students | pending\_evaluations | this\_week\_submissions | recent\_avg\_score |
| --- | --- | --- | --- | --- | --- | --- |
| T123456 | 劉建國 | 1 | 2 | 0 | 0 | 80.83 |
| T123457 | 吳美玲 | 1 | 1 | 0 | 0 | 91.00 |
| T123458 | 黃志偉 | 1 | 1 | 0 | 0 | 84.00 |
| T123459 | 李教授 | 0 | 0 | 0 | 0 | NULL |

### 2.14 檔案上傳統計檢視 (view\_file\_upload\_statistics)

#### VIEW建立語句

CREATE VIEW view\_file\_upload\_statistics AS  
SELECT   
 DATE(f.uploaded\_at) as upload\_date,  
 COUNT(f.file\_id) as total\_files,  
 COUNT(DISTINCT f.report\_id) as reports\_with\_files,  
 ROUND(SUM(f.file\_size) / 1024 / 1024, 2) as total\_size\_mb,  
 f.file\_type,  
 COUNT(\*) as type\_count  
FROM files f  
GROUP BY DATE(f.uploaded\_at), f.file\_type  
ORDER BY upload\_date DESC, type\_count DESC;

#### 輸出結果

| upload\_date | total\_files | reports\_with\_files | total\_size\_mb | file\_type | type\_count |
| --- | --- | --- | --- | --- | --- |
| 2024-09-29 | 1 | 1 | 3.50 | PDF文件 | 1 |
| 2024-09-23 | 1 | 1 | 1.50 | 圖片 | 1 |
| 2024-09-22 | 3 | 3 | 17.00 | 壓縮檔 | 1 |
| 2024-09-22 | 3 | 3 | 17.00 | 圖片 | 1 |
| 2024-09-22 | 3 | 3 | 17.00 | 設計檔 | 1 |
| 2024-09-16 | 1 | 1 | 3.00 | 圖片 | 1 |
| 2024-09-15 | 5 | 4 | 10.75 | PDF文件 | 2 |
| 2024-09-15 | 5 | 4 | 10.75 | Excel文件 | 1 |
| 2024-09-15 | 5 | 4 | 10.75 | Word文件 | 1 |

### 2.15 學期成績報表檢視 (view\_semester\_grade\_report)

#### VIEW建立語句

CREATE VIEW view\_semester\_grade\_report AS  
SELECT   
 ay.year\_name,  
 s.department,  
 g.group\_name,  
 s.student\_id,  
 su.name as student\_name,  
 tu.name as teacher\_name,  
 COUNT(e.evaluation\_id) as total\_evaluations,  
 ROUND(AVG(e.score), 2) as semester\_average,  
 MIN(e.score) as lowest\_score,  
 MAX(e.score) as highest\_score,  
 CASE   
 WHEN AVG(e.score) >= 90 THEN 'A'  
 WHEN AVG(e.score) >= 80 THEN 'B'  
 WHEN AVG(e.score) >= 70 THEN 'C'  
 WHEN AVG(e.score) >= 60 THEN 'D'  
 ELSE 'F'  
 END as letter\_grade,  
 COUNT(CASE WHEN wr.status = 'late' THEN 1 END) as late\_submissions,  
 COUNT(CASE WHEN wr.status = 'missing' THEN 1 END) as missing\_submissions  
FROM academic\_years ay  
JOIN groups g ON ay.year\_id = g.year\_id  
JOIN students s ON g.group\_id = s.group\_id  
JOIN users su ON s.user\_id = su.user\_id  
LEFT JOIN teachers t ON g.teacher\_id = t.teacher\_id  
LEFT JOIN users tu ON t.user\_id = tu.user\_id  
LEFT JOIN weekly\_reports wr ON s.student\_id = wr.student\_id AND wr.year\_id = ay.year\_id  
LEFT JOIN evaluations e ON wr.report\_id = e.report\_id  
GROUP BY ay.year\_name, s.department, g.group\_name, s.student\_id,   
 su.name, tu.name  
HAVING COUNT(e.evaluation\_id) > 0  
ORDER BY ay.year\_name, s.department, semester\_average DESC;

#### 輸出結果

| year\_name | department | group\_name | student\_id | student\_name | teacher\_name | total\_evaluations | semester\_average | lowest\_score | highest\_score | letter\_grade | late\_submissions | missing\_submissions |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 2024-2025 | 資訊工程系 | AI影像組 | 40943214 | 王小美 | 吳美玲 | 2 | 91.00 | 90 | 92 | A | 0 | 0 |
| 2024-2025 | 資訊工程系 | 智慧家居組 | 40943212 | 陳廷威 | 劉建國 | 3 | 86.67 | 85 | 88 | B | 0 | 0 |
| 2024-2025 | 電機工程系 | 行動應用組 | 40943216 | 林雅婷 | 黃志偉 | 2 | 84.00 | 82 | 86 | B | 0 | 0 |
| 2024-2025 | 資訊工程系 | 智慧家居組 | 40943213 | 李明華 | 劉建國 | 3 | 75.33 | 73 | 78 | C | 2 | 0 |

### 2.16 專題進度追蹤報表檢視 (view\_project\_progress\_report)

#### VIEW建立語句

CREATE VIEW view\_project\_progress\_report AS  
SELECT   
 ay.year\_name,  
 g.group\_id,  
 g.group\_name,  
 g.project\_title,  
 g.status as project\_status,  
 tu.name as supervisor\_name,  
 COUNT(DISTINCT s.student\_id) as team\_size,  
 COUNT(DISTINCT wr.week\_number) as weeks\_reported,  
 ROUND(COUNT(DISTINCT wr.week\_number) \* 100.0 / 18, 2) as completion\_percentage,  
 ROUND(AVG(e.score), 2) as average\_score,  
 COUNT(CASE WHEN wr.status = 'late' THEN 1 END) as late\_submissions,  
 COUNT(CASE WHEN wr.status = 'missing' THEN 1 END) as missing\_submissions,  
 MAX(wr.submitted\_at) as last\_update,  
 COUNT(DISTINCT f.file\_id) as total\_files\_uploaded,  
 ROUND(SUM(f.file\_size) / 1024 / 1024, 2) as total\_file\_size\_mb  
FROM academic\_years ay  
JOIN groups g ON ay.year\_id = g.year\_id  
LEFT JOIN teachers t ON g.teacher\_id = t.teacher\_id  
LEFT JOIN users tu ON t.user\_id = tu.user\_id  
LEFT JOIN students s ON g.group\_id = s.group\_id  
LEFT JOIN weekly\_reports wr ON s.student\_id = wr.student\_id AND wr.year\_id = ay.year\_id  
LEFT JOIN evaluations e ON wr.report\_id = e.report\_id  
LEFT JOIN files f ON wr.report\_id = f.report\_id  
GROUP BY ay.year\_name, g.group\_id, g.group\_name, g.project\_title,   
 g.status, tu.name  
ORDER BY ay.year\_name, completion\_percentage DESC;

#### 輸出結果

| year\_name | group\_id | group\_name | project\_title | project\_status | supervisor\_name | team\_size | weeks\_reported | completion\_percentage | average\_score | late\_submissions | missing\_submissions | last\_update | total\_files\_uploaded | total\_file\_size\_mb |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 2024-2025 | 1 | 智慧家居組 | 基於IoT的智慧家居控制系統 | active | 劉建國 | 2 | 3 | 16.67 | 80.83 | 2 | 0 | 2024-09-30 01:30:00 | 5 | 9.00 |
| 2024-2025 | 2 | AI影像組 | 深度學習影像辨識系統 | active | 吳美玲 | 1 | 2 | 11.11 | 91.00 | 0 | 0 | 2024-09-22 21:45:00 | 2 | 11.46 |
| 2024-2025 | 3 | 行動應用組 | 校園導覽APP開發 | active | 黃志偉 | 1 | 2 | 11.11 | 84.00 | 0 | 0 | 2024-09-22 19:20:00 | 2 | 9.00 |

### 2.17 審核工作統計檢視 (view\_approval\_workload)

#### VIEW建立語句

CREATE VIEW view\_approval\_workload AS  
SELECT   
 ua.admin\_id,  
 au.name as admin\_name,  
 ua.request\_type,  
 COUNT(\*) as total\_requests,  
 SUM(CASE WHEN ua.status = 'pending' THEN 1 ELSE 0 END) as pending\_count,  
 SUM(CASE WHEN ua.status = 'approved' THEN 1 ELSE 0 END) as approved\_count,  
 SUM(CASE WHEN ua.status = 'rejected' THEN 1 ELSE 0 END) as rejected\_count,  
 ROUND(AVG(DATEDIFF(ua.processed\_at, ua.requested\_at)), 2) as avg\_processing\_days  
FROM user\_approvals ua  
LEFT JOIN users au ON ua.admin\_id = au.user\_id  
WHERE ua.requested\_at >= DATE\_SUB(NOW(), INTERVAL 90 DAY)  
GROUP BY ua.admin\_id, au.name, ua.request\_type  
ORDER BY pending\_count DESC;

#### 輸出結果

| admin\_id | admin\_name | request\_type | total\_requests | pending\_count | approved\_count | rejected\_count | avg\_processing\_days |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 9 | 系統管理員 | registration | 6 | 2 | 4 | 0 | 2.75 |
| 9 | 系統管理員 | role\_change | 2 | 1 | 1 | 0 | 1.00 |
| 9 | 系統管理員 | account\_recovery | 2 | 0 | 1 | 1 | 1.00 |

### 2.18 系統使用統計檢視 (view\_system\_usage\_statistics)

#### VIEW建立語句

CREATE VIEW view\_system\_usage\_statistics AS  
SELECT   
 DATE(sl.created\_at) as activity\_date,  
 sl.action\_type,  
 COUNT(\*) as action\_count,  
 COUNT(DISTINCT sl.user\_id) as unique\_users,  
 COUNT(DISTINCT sl.ip\_address) as unique\_ips  
FROM system\_logs sl  
WHERE sl.created\_at >= DATE\_SUB(NOW(), INTERVAL 30 DAY)  
GROUP BY DATE(sl.created\_at), sl.action\_type  
ORDER BY activity\_date DESC, action\_count DESC;

#### 輸出結果

| activity\_date | action\_type | action\_count | unique\_users | unique\_ips |
| --- | --- | --- | --- | --- |
| 2024-10-01 | create | 2 | 1 | 1 |
| 2024-09-30 | create | 1 | 1 | 1 |
| 2024-09-29 | create | 1 | 1 | 1 |
| 2024-09-24 | login | 2 | 2 | 2 |
| 2024-09-24 | view | 1 | 1 | 1 |
| 2024-09-23 | login | 1 | 1 | 1 |
| 2024-09-23 | create | 1 | 1 | 1 |
| 2024-09-20 | login | 1 | 1 | 1 |
| 2024-09-20 | update | 1 | 1 | 1 |
| 2024-09-17 | login | 1 | 1 | 1 |
| 2024-09-17 | create | 1 | 1 | 1 |
| 2024-09-15 | login | 1 | 1 | 1 |
| 2024-09-15 | create | 1 | 1 | 1 |

## 3. 常用查詢範例

### 3.1 管理員常用查詢

-- 查詢系統整體狀況  
SELECT \* FROM view\_admin\_dashboard;  
  
-- 查詢待審核用戶  
SELECT \* FROM view\_user\_basic\_info WHERE status = 'pending';  
  
-- 查詢各學年度統計  
SELECT \* FROM view\_academic\_year\_overview WHERE status = 'active';  
  
-- 查詢本週提交的週報統計  
SELECT \* FROM view\_weekly\_report\_submission\_status   
WHERE year\_name = '2024-2025'   
ORDER BY week\_number DESC LIMIT 5;

### 3.2 教師常用查詢

-- 查詢教師工作台資訊  
SELECT \* FROM view\_teacher\_dashboard WHERE teacher\_id = 'T123456';  
  
-- 查詢教師指導的學生列表  
SELECT \* FROM view\_student\_full\_info   
WHERE teacher\_id = 'T123456'   
ORDER BY student\_name;  
  
-- 查詢組別進度統計  
SELECT \* FROM view\_group\_progress\_stats   
WHERE teacher\_name = '劉建國';  
  
-- 查詢評分統計  
SELECT \* FROM view\_evaluation\_statistics   
WHERE teacher\_id = 'T123456';

### 3.3 學生常用查詢

-- 查詢學生完整資訊  
SELECT \* FROM view\_student\_full\_info WHERE student\_id = '40943212';  
  
-- 查詢學生週報統計  
SELECT \* FROM view\_student\_report\_stats WHERE student\_id = '40943212';  
  
-- 查詢學生當前狀態  
SELECT \* FROM view\_student\_current\_status WHERE student\_id = '40943212';  
  
-- 查詢學生成績分析  
SELECT \* FROM view\_student\_grade\_analysis WHERE student\_id = '40943212';  
  
-- 查詢學生通知  
SELECT \* FROM view\_user\_notification\_summary WHERE user\_id = 1;

### 3.4 報表產生查詢

-- 產生學期成績報表  
SELECT \* FROM view\_semester\_grade\_report   
WHERE year\_name = '2024-2025'   
ORDER BY semester\_average DESC;  
  
-- 產生專題進度報表  
SELECT \* FROM view\_project\_progress\_report   
WHERE year\_name = '2024-2025'   
ORDER BY completion\_percentage DESC;  
  
-- 產生檔案使用統計  
SELECT upload\_date, SUM(total\_files) as daily\_files, SUM(total\_size\_mb) as daily\_size\_mb  
FROM view\_file\_upload\_statistics   
GROUP BY upload\_date   
ORDER BY upload\_date DESC;  
  
-- 產生系統使用報表  
SELECT activity\_date, SUM(action\_count) as total\_actions, COUNT(DISTINCT action\_type) as action\_types  
FROM view\_system\_usage\_statistics   
GROUP BY activity\_date   
ORDER BY activity\_date DESC;