

資料庫系統期末報告

診所掛號系統

第15組

組員:41143202 林詩宸

41143203 張祐慈

41143204 陳郁庭

41143206 盧奕岑

目錄

一、摘要.....	3
二、研究動機.....	3
三、研究目的.....	3
四、應用情境.....	3
五、使用案例.....	4
5.1 使用案例.....	4
5.2 使用案例圖.....	4
六、系統功能需求.....	5
6.1 功能性需求.....	5
6.2 非功能性需求.....	5
6.3 完整性限制.....	6
七、ER Diagram.....	6
八、DB Schema.....	7
九、資料庫設計.....	7
9.1 sick_basic病患基本資料表.....	8
9.2 sick_register掛號資料表.....	10
9.3 doc_schedule排班資料表.....	12
9.4 clinic_room_basic診間基本資料表.....	13
9.5 doctor_basic基本資料表.....	14
十、VIEW表建立.....	15
10.1使用者VIEW表.....	15
10.1.1 personal_register_history_view.....	15
10.1.2 available_doctor_schedule_view.....	16
10.1.3 user_basic_info_view.....	17
10.2管理員VIEW表.....	17
10.2.1 sick_register_today.....	18
10.2.2 today_doctor_schedule_view.....	19
10.2.3 today_sick_basic_info_view.....	20
10.2.4 today_doctor_pending_patients_view.....	21
10.2.5 duplicate_register_check_view.....	22
十一、心得.....	22

一、摘要

本專案為一套為診所設計的門診掛號系統，可為病患提供線上掛號等功能，並可支援醫師門診排班、設定門診時間、管理及查詢病患基本資料等功能。不僅提升病患掛號的便利性，也減少現場等候與人員負擔。透過後端資料庫的整合與管理，診所能有效掌握每日門診狀況、病患掛號紀錄與醫師排班安排。

二、研究動機

隨著現代人對醫療服務效率與便利性的要求提升，傳統診所使用紙本或電話方式掛號的方式，已漸漸無法滿足病患的需求。不僅容易出現資料遺漏或重複掛號的情況，也常造成病患在現場等候過久，影響整體看診體驗。另一方面，診所人員也需花費大量時間處理登記、查詢與安排門診等行政工作，導致作業效率下降。因此，我們希望透過開發一套診所門診掛號系統，協助病患更輕鬆地完成掛號流程，同時也讓診所能夠簡化作業流程、減少人力負擔，提升整體服務品質與管理效率。

三、研究目的

本專案旨在提升診所掛號流程的效率，改善病患的掛號體驗，減少人工登記的錯誤以及等待時間，讓病患能夠更方便、快速地完成掛號。系統將整合病患基本資料、醫師門診時段與掛號紀錄，使病患能輕鬆查詢與預約門診。同時，也協助診所管理人員更有效率地安排醫師班表、掌握現場掛號情況，進而提升整體醫療服務品質與診所營運效率。

四、應用情境

本系統設計以小型診所日常作業為核心，模擬真實掛號與看診流程，提供病患、醫護人員與櫃台管理員便捷的資訊操作介面。

在實際情境中，病患透過系統登入後，可查詢目前可掛號的醫師與其出診時段，並依據需求，選擇合適的時段與醫師進行掛號。系統會即時判斷該時段是否仍有可用名額，若額滿則自動顯示其他可掛號選項。掛號完成後，病患可於就診日前再次登入系統，確認掛號資訊，並在需要時進行取消與更改。

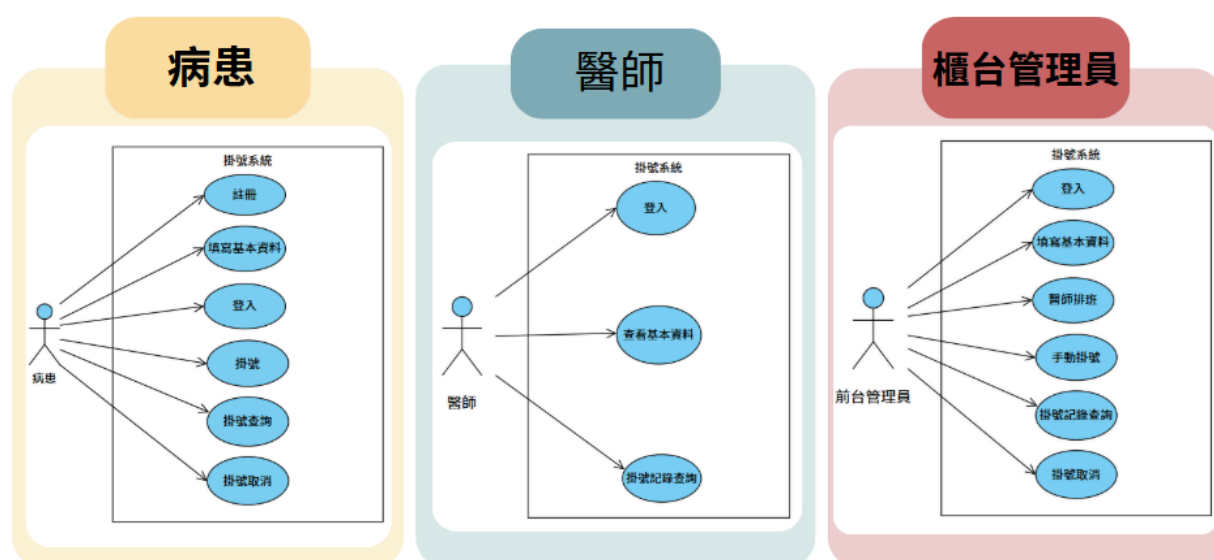
在診所端，櫃台管理員每日可透過後台管理介面，設定各醫師的門診排班資訊，包含出診時間、診間與可掛號人數等。管理員能協助未使用網路掛號的病患進行手動掛號，並修改病患資料或查看報到名單。當病患前來就診時，醫護人員可直接於系統查詢當日掛號名單，並勾選完成報到。

五、使用案例

5.1 使用案例

- 病患掛號：塵塵因為人不舒服，想透過線上掛號避免現場排隊，確保有無掛號名額。利用身分證號註冊登入後，查找林某某醫師的就診時段，選擇5/1的14:00到17:30，按下確認掛號後，會跳出掛號成功，若該醫師掛號名額已到，會跳出已額滿。
- 醫師排班管理：櫃台盧某某要幫每位醫師排班，使用統一的帳號密碼登入後，設定林某某醫師5/12的08:30到12:00及18:00到21:30要上班並顯示可掛號人數。
- 掛號查詢/取消：停停掛號成功後，查詢了掛號紀錄，確保他有無掛到號，若要取消該掛號，點選取消鍵。
- 報到與診療紀錄管理：又又線上掛號到現場報到，因為她第一次去填寫了基本資料，櫃台管理員將她的資料打進系統，方便醫師為她整治。

5.2 使用案例圖



圖一、使用案例圖

六、系統功能需求

6.1 功能性需求

依照不同使用者進行區分，主要包含病患、醫護人員與櫃台管理員三大類。

對於病患端，系統提供帳號註冊與登入功能，使使用者能以個人身份進行操作。登入後，病患可透過系統查詢可掛號的醫師與時段，並選擇心儀的醫師與時間完成掛號預約。此外，系統亦支援查詢與取消掛號紀錄，以及檢視與修改個人基本資料，使病患可即時掌握自身就診資訊並進行彈性調整。

對於醫護人員，能夠查看每日掛號名單，掌握預計看診的病患清單。醫護人員可在完成看診後，直接於系統中建立診療紀錄。系統亦支援查詢歷史掛號與診療資料，以便於追蹤病患病況發展與後續追蹤照護。同時，醫護人員也可查詢自己的排班資訊，掌握出診時間與診間安排。

針對櫃台管理員，系統提供登入與權限控管機制，以保護病患與醫護資訊。管理員可設定每位醫師的排班時段與可掛號人數，並進行診間的安排與管理。對於未使用線上掛號的病患，櫃台亦可進行現場的手動掛號服務。此外，管理員擁有查詢與修改病患基本資料的權限，能協助病患建立完整資料檔案，提升整體掛號與診療效率。

6.2 非功能性需求

除了基本的功能性需求外，本系統也重視非功能性層面的設計，以確保系統能穩定、安全且具未來擴充性。首先，系統必須具備高度的資料安全性與隱私保護機制，由於病患個資與醫療紀錄皆屬於敏感資訊，因此系統在設計上必須透過使用者驗證、權限分級、資料加密與異常操作監控等方式，降低資料外洩風險。

其次，系統需具備高可用性與備援能力，避免因系統故障造成診所掛號服務中斷。透過資料庫定期備份與容錯設計，確保即便在意外發生時也能快速恢復資料與系統運作。此外，系統介面需具備良好的使用者體驗設計，不論年長者或初次使用者皆能直覺操作。採用清晰的選單結構與友善的畫面設計，提升病患掛號與查詢的便利性。

最後，考量到未來診所業務可能擴大或多診所共用情境，本系統也必須具備良好的可擴充性與模組化架構，以支援更多使用者、醫師與診間的資料整合與管理，進一步提升整體診所資訊化管理的效能與彈性。

6.3 完整性限制

為了確保資料正確性與邏輯一致性，在資料庫設計中納入多項完整性限制。首先在主鍵與唯一性方面，系統中的關鍵資料表，如：患者資料、醫師資料、掛號資料與排班資料，皆設有唯一的主鍵，以防止資料重複，確保每筆資料能被唯一識別。

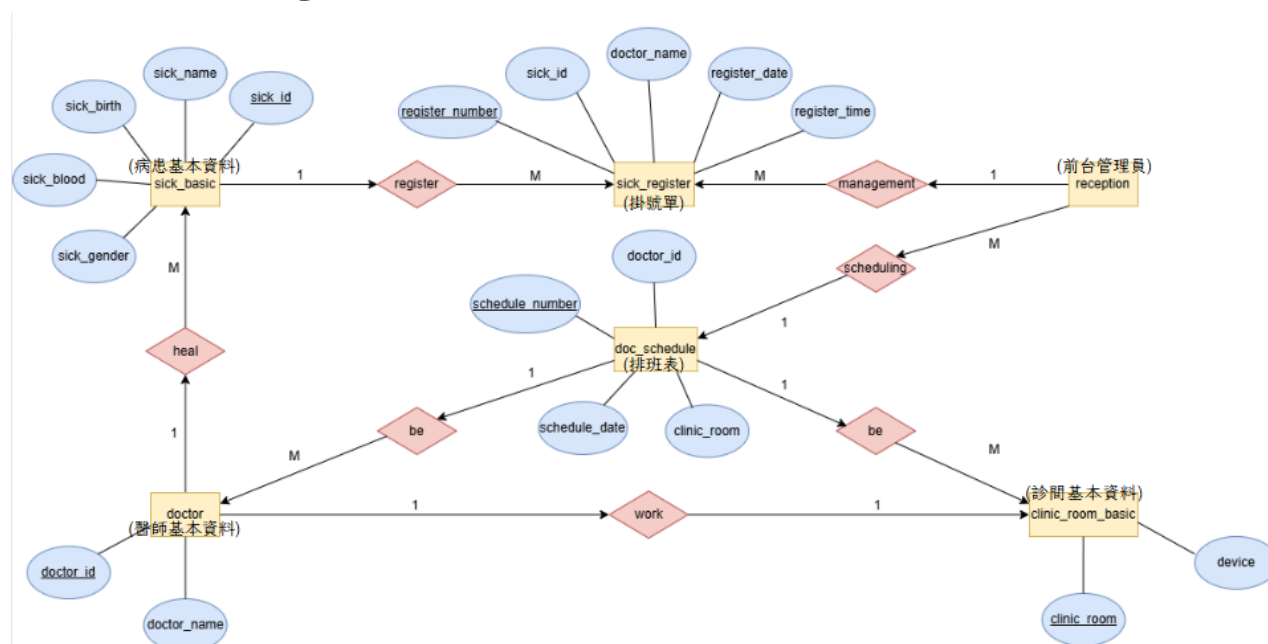
在外鍵關聯上，掛號紀錄必須具備對應的病患與醫師資訊，因此掛號資料表中，設有指向病患與醫師資料的外鍵約束，避免出現無效或不存在的病患與醫師紀錄。同樣地，每筆掛號紀錄也必須對應到有效的排班時段，避免患者掛號到非出診時段。

系統強制規定掛號日期不得早於當前日期，以防使用者選擇過期時段。

報到時間僅能設定在掛號當天，並且報到紀錄必須依附於一筆有效掛號資料，且每筆掛號僅能報到一次，避免重複診療或異常紀錄。為防止濫用掛號資源，系統也設定「每位病患每日對同一位醫師僅能掛號一次」的限制，有效避免重複掛號或搶掛。

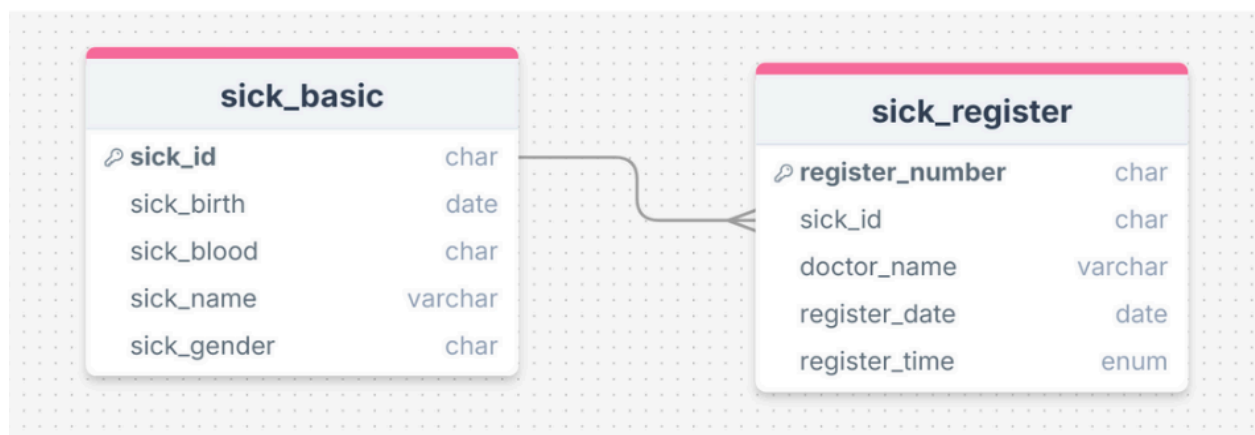
透過上述這些完整性限制的設計，能有效提升系統資料的可靠性與安全性，並確保後續診療、查詢、統計等功能的準確性與一致性。

七、ER Diagram



圖二、ER Diagram

八、DB Schema



前台管理員查看掛號單可以透過病患的身分證字號找到病患的基本資料

前台管理員使用排班表可以透過診間編號找到診間基本資料



前台管理員使用排班表可以透過醫師編號找到醫師基本資料

圖三、DB Schema

九、資料庫設計

以下建立四個關鍵資料表:sick_basic、sick_register、doc_schedule及clinic_room_basic, 共同支撐病患就診與掛號管理的核心流程。

sick_basic 負責儲存病患的基本資料, 包括姓名、身分證號、出生年月日及血型, 作為所有就診與掛號資訊的基礎來源。

sick_register 紀錄每位病患的掛號資訊, 例如就診日期、時段與指定醫師, 並透過病患 ID 與 sick_basic 關聯, 方便查詢歷史就診紀錄。

doc_schedule 表用來定義各位醫師的姓名、排班時段與診間編號，確保排班資料清楚無衝突，並作為掛號流程的時段篩選依據。

clinic_room_basic 則管理診間資訊，如診間編號與對應診間是否能使用，有助於診間資源的合理分配與調度。

9.1 sick_basic病患基本資料表

表 1:sick_basic資料表

欄位名稱	資料型別	限制條件	說明
sick_id	CHAR(10)	PRIMARY KEY , NOT NULL	病患身分證字號
sick_birth	DATE	NOT NULL	病患生日
sick_blood	CHAR(2)	NOT NULL	病患血型(A,B,O,AB)
sick_name	STRING	NOT NULL	病患姓名
sick_gender	CHAR(1)	只可填寫M或F	病患性別(男M女F)

表 2:sick_basic完整性限制

欄位名稱	完整性限制	說明
sick_id	第一個字元只可為大寫字母，第二個字元則是1或2，總長度10	病患身分證字號
sick_birth	日期格式為 yyyy-mm-dd，不可填寫未來日期	病患生日
sick_blood	只可輸入'A', 'B', 'AB', 'O'，不能含有數字、特殊符號、除這四種英文字組外的英文出現	病患血型
sick_name	長度不可超過50字元，不包含特殊符號	病患姓名
sick_gender	只可填寫 'M' 或 'F'，不能含有數字、特殊符號、除這兩種英文字外的英文出現	病患性別

sick_id 是用來辨識每一位病患的主要欄位，當病患資料建立時，這個欄位會儲存病患的身分證字號，作為每筆紀錄的唯一識別碼。系統後續在查詢病患資料(例如掛號、診

斷紀錄等)時, 會透過這個 ID 作為關聯依據, 確保只查詢到該名病患的資訊。為了避免重複或資料錯亂, 這個欄位設為主鍵, 並且限制為長度10、首碼只能是1或2, 以符合實際身分格式。

sick_birth 是紀錄病患出生日期的欄位。這個欄位為必填, 系統會禁止填入未來的日期, 以確保資料真實有效。出生日期可協助醫護人員掌握病患年齡層, 進而作為醫療診斷或健康管理的參考依據。

sick_blood 用來儲存病患的血型資料, 僅能填入 'A'、'B'、'AB' 或 'O' 四種格式, 是不可為空的欄位。血型資訊在緊急醫療處置如輸血時非常關鍵, 因此系統會強制使用者填寫正確的血型資料。

sick_name 是病患的姓名欄位, 讓醫護人員或系統能夠清楚顯示與識別該病患。這個欄位必須輸入, 且長度不能超過 50 個字元, 確保資料不會異常過長而導致顯示或儲存上的問題。

sick_gender 是表示病患性別的欄位, 僅可輸入 'M'(男性)或 'F'(女性)。系統會根據性別進行分類統計或醫療判斷, 因此這個欄位設計為必要欄位, 並限制輸入的格式, 以維持資料一致性。

表 3:sick_basic資料建置

sick_id	sick_birth	sick_blood	sick_name	sick_gender
A123456789	1990-06-15	A	王小明	M
B287654321	1985-11-23	B	李美麗	F
C135792468	2000-01-30	O	張大華	M
D146813579	1992-08-05	AB	陳志強	M
E212233445	1998-12-12	A	林佳慧	F
F156677889	1987-04-08	O	黃子豪	M
G298877665	1995-03-22	B	周宛如	F
H234455667	1993-07-17	AB	許文婷	F
I176655443	1989-09-10	A	邱志明	M
J223789456	1997-05-03	O	洪佩君	F

9.2 sick_register掛號資料表

表 4:sick_register資料表

欄位名稱	資料型別	限制條件	說明
register_number	CHAR(10)	PRIMARY KEY	掛號編號
sick_id	CHAR(10)	NOT NULL , FOREIGN KEY-> sick_register(sick_id)	病患身分證字號
doctor_name	STRING	NOT NULL	醫生姓名
register_data	DATE	NOT NULL	就診日期
register_time	CHAR(1)	NOT NULL	就診時段

表 5:sick_register完整性限制

欄位名稱	完整性限制	說明
register_number	格式為前六碼為日期, 後兩碼為大於0小於100的雙數編號	掛號編號
sick_id	第一個字元只可為大寫字母, 第二個字元則是1或2, 總長度10	病患身分證字號
doctor_name	長度不可超過50字元, 不包含特殊符號	醫生姓名
register_data	格式為 yyyy-mm-dd, 不可填寫過去日期	就診日期
register_time	只可填寫'早', '中', '晚', 不能含有數字、特殊符號、除這三種文字外的字出現	就診時段

register_number 是每筆掛號紀錄的唯一編號, 並設為主鍵, 確保每筆掛號資料都能唯一識別, 可以讓系統快速查找與排序掛號資料, 同時避免重複編號造成的錯誤。

sick_id 是病患身分證字號, 是用來與病患基本資料進行關聯的外鍵欄位。當病患掛號時, 系統會根據此欄位連接至該病患的詳細基本資料, 確保掛號紀錄的正確性與一致性。此欄位同樣限制第一個字元只能是 1 或 2、總長度為 10, 並禁止為空。

doctor_name 是掛號時指定的醫師姓名，用來表示該次就診由哪位醫師負責。此欄位為必填欄位，並限制長度在 50 字元以內，讓病患與醫師間的就診紀錄能夠正確對應。

register_data 是病患預約就診的日期，為必要欄位。系統會檢查輸入的日期不可早於當天，避免掛號到過去的時段，並確保掛號作業符合實際流程。

register_time 表示病患掛號的時段，資料型別為 ENUM，僅可填入「早」、「中」、「晚」的時段，有助於醫師排班與病患流量管理，並可避免時段輸入錯誤。

表 6:sick_register資料建置

register_number	sick_id	doctor_name	register_data	register_time
2025060602	A123456789	陳建安	2025-05-01	早
2025060604	B287654321	林彥廷	2025-05-01	中
2025060606	C135792468	陳建安	2025-05-01	晚
2025060702	D146813579	陳建安	2025-05-02	早
2025060704	E212233445	林彥廷	2025-05-02	中
2025060706	F156677889	黃怡君	2025-05-02	晚
2025060802	G298877665	黃怡君	2025-05-03	早
2025060804	H234455667	陳建安	2025-05-03	中
2025060806	I176655443	林彥廷	2025-05-03	晚
2025060902	J223789456	黃怡君	2025-05-04	早

9.3 doc_schedule排班資料表

表 7: doc_schedule資料表

欄位名稱	資料型別	限制條件	說明
schedule_number	INTEGER	NOT NULL , AUTO_INCREMENT , PRIMARY KEY	排班編號
doctor_id	CHAR(8)	NOT NULL	醫生id
schedule_data	DATE	NOT NULL	排班日期
clinic_room	INTEGER	NOT NULL , FOREIGN KEY-> clinicroom_basic(clinic_ room)	診間編號

表 8: doc_schedule完整性限制

欄位名稱	完整性限制	說明
schedule_number	系統依照順序編號編號，從1開始的整數	排班編號
doctor_id	只可為八碼的數字	醫生id
schedule_data	格式為 yyyy-mm-dd，不可填寫過去日期	排班日期
clinic_room	只能填寫'101', '102', '103'，不能含有數字、特殊符號、除這三種數組外的數組出現，系統若查詢到診間設備為'N'則不可填寫	診間編號

schedule_number 是每筆排班資料的唯一編號，系統會自動產生連號，並設為主鍵，避免出現重複的排班編號，方便日後查詢、修改與追蹤醫師的排班紀錄。

doctor_id 是用來對應醫師身分的欄位，填寫的是醫師的身分識別碼。

schedule_data 是指醫師排班的日期，系統會檢查不得輸入過去的日期，確保所有排班都是有效的未來行程。這個欄位是必要填寫的，避免產生空白或錯誤的時間資訊。

clinic_room 是安排醫師值班的診間代碼, 只能從預先定義好的診間(如 '101'、'102'、'103')中選擇。會與 clinicroom_basic 資料表建立關聯, 若該診間設備標註為不可使用, 則系統不允許填入。

表 9: doc_schedule 資料建置

schedule_number	doctor_id	schedule_data	clinic_room
1	23456789	2025-05-14	101
2	87654321	2025-05-14	102
3	12233445	2025-05-14	103
4	23456789	2025-05-15	101
5	87654321	2025-05-15	102
6	12233445	2025-05-15	103
7	23456789	2025-05-16	101
8	87654321	2025-05-16	102
9	12233445	2025-05-16	103
10	23456789	2025-05-17	101

9.4 clinic_room_basic 診間基本資料表

表 10: clinic_room_basic 資料表

欄位名稱	資料型別	限制條件	說明
clinic_room	INTEGER	NOT NULL ,PRIMARY KEY	診間編號
device	CHAR(1)	NOT NULL	設備好壞

表 11: clinic_room_basic 完整性限制

欄位名稱	完整性限制	說明
clinic_room	只能填寫'101', '102', '103'	診間編號

device	只能填寫 'Y','N', 不能含有數字、特殊符號、除這兩種英文外的字母出現	設備好壞
--------	--	------

clinic_room是診間的識別代碼，用來區分不同的診療空間。系統只允許填寫 '101'、'102'、'103'三種編號，並設為主鍵，確保每個診間在資料表中都是唯一且不重複的。

device則是用來標示該診間是否具備可用設備的欄位。只能填寫 'Y' 或 'N'，系統會根據這個欄位判斷該診間是否可供安排使用。

表 12:clinic_room_basic資料建置

clinic_room	device
101	Y
102	N
103	Y

9.5 doctor_basic基本資料表

表 13:doctor_basic資料表

欄位名稱	資料型別	限制條件	說明
doctor_id	CHAR(8)	NOT NULL ,PRIMARY KEY	醫師編號
doctor_name	STRING	NOT NULL	醫生姓名

表 14:doctor_basic完整性限制

欄位名稱	完整性限制	說明
doctor_id	只可為八碼的數字	醫師編號
doctor_name	長度不可超過50字元， 不包含特殊字元	醫生姓名

doctor_id 是用來對應醫師身分的欄位，填寫的是醫師的身分識別碼。

doctor_name是醫師的名字，用來清楚標示是哪一位醫師值班。儘管系統會以 doctor_id 作為關聯，但加上醫師姓名可以提升介面清晰度與使用者操作便利性。為避免欄位過長，系統限制最多只能輸入 50 個字元。

表 15: clinic_room_basic資料建置

doctor_id	doctor_name
92025001	陳建安
62025002	林彥廷
22025003	黃怡君

十、VIEW表建立

10.1使用者VIEW表

表16: 使用者VIEW表資訊

名稱	選擇的屬性	說明
查詢個人掛號記錄	sick_register(register_number, doctor_name, register_data, register_time)	病患可以查詢自己的掛號記錄，包含掛號號碼、醫師姓名、掛號日期及時間
查詢可預約的醫師與時段	sick_register(doctor_name, register_data, register_time, remaining_slots)	病患可以查詢目前可預約的醫師與時段，包含排班號碼、醫師姓名、日期及診間
查詢個人基本資料	sick_basic(sick_id, sick_name, sick_gender, sick_birth, sick_blood)	病患可以查看自己的基本資料，包含姓名、性別、生日及血型

10.1.1 personal_register_history_view

該檢視表透過病患身份證字號，可顯示其完整的掛號紀錄，包含每筆掛號的時間、看診醫師，以及病患的基本個資如姓名、性別、生日與血型等。結合了掛號紀錄表與病患基本資料表，讓查詢結果更加完整且具關聯性，是醫師在看診時常用來回顧病患就診歷史的重要依據。

```
CREATE VIEW personal_register_history_view AS
SELECT
  r.register_number,
  r.sick_id,
  b.sick_name,
  b.sick_gender,
  b.sick_birth,
  b.sick_blood,
  r.doctor_name,
  r.register_data,
  r.register_time
FROM sick_register r
JOIN sick_basic b ON r.sick_id = b.sick_id;
SELECT *
FROM personal_register_history_view
WHERE sick_id = 'A123456789';
```

10.1.2 available_doctor_schedule_view

此檢視表將醫師排班資料表與病患掛號表進行連接比對，並計算每位醫師在特定日期與時段中尚未被掛號的名額（每時段最多 33 位病患）。系統會從每筆排班資料中減去實際已掛號人數，產生 available_slots 欄位，用以顯示剩餘掛號名額，若為 0 則不會顯示在結果中。

```
CREATE VIEW available_doctor_schedule_view AS
SELECT
  ds.schedule_number,
  ds.doctor_id,
```



```

ds.doctor_name,
ds.schedule_date,
ds.schedule_time,
ds.clinic_room,
33 - COUNT(sr.register_number) AS available_slots
FROM doctor_schedule ds
LEFT JOIN sick_register sr
ON ds.doctor_id = sr.doctor_id
AND ds.schedule_date = sr.register_date
AND ds.schedule_time = sr.register_time
GROUP BY
ds.schedule_number,
ds.doctor_id,
ds.doctor_name,
ds.schedule_date,
ds.schedule_time,
ds.clinic_room
HAVING available_slots > 0;

```

10.1.3 user_basic_info_view

此檢視表是用來查詢病患的基本資料，包含病患的身份證字號、姓名、性別、出生年月日以及血型等資訊。該檢視表是直接從病患基本資料表，擷取欄位內容所建立，方便系統中其他模組快速查詢病患基本身分。

```

CREATE VIEW user_basic_info_view AS
SELECT
sick_id,
sick_name,
sick_gender,
sick_birth,
sick_blood
FROM sick_basic;
SELECT * FROM user_basic_info_view

```

WHERE sick_id = 'A123456789';

10.2 管理員VIEW表

表17: 管理員VIEW表資訊

名稱	選擇的屬性	說明
系統內今日掛號單	sick_register(sick_id, register_data), sick_basic(sick_id, sick_name, sick_gender, sick_birth, sick_blood)	管理者可以查詢到當日掛號單資訊, 包含掛號號碼、病患ID、醫師姓名、掛號日期及時間
今日醫生排班表	sick_register(schedule_number, doctor_id, doctor_name, schedule_data, clinic_room)	查詢今日醫生排班表, 包含排班號碼、醫師ID、醫師姓名、排班日期及診間
管理者透過今日掛號單查詢患者個人資料	sick_basic(sick_id, sick_name, sick_gender, sick_birth, sick_blood)	管理者可透過掛號單以病患身分字號查詢個人資料, 包含姓名、性別、生日及血型
統計每位醫師今日看診數	sick_register(doctor_name, register_data), COUNT(*) AS pending_patient_count	統計每位醫師今日看診人數
查出同一病患在同一天是否重複掛號	sick_register(sick_id, register_data), COUNT(*) AS count_per_day	查出同一病患在同一天是否重複掛號, 若次數大於1則顯示

10.2.1 sick_register_today

管理員與醫師可以透過今日掛號單, 查詢今日的掛號紀錄。內容包含掛號編號、病患身份證號碼、負責醫師姓名、掛號日期以及掛號時段。可快速了解當日有哪些病患掛號, 以及這些病患分別安排在哪位醫師的看診時段, 進而進行診間安排、看診提醒等後續作業。

CREATE VIEW sick_register_today AS

```

SELECT
    register_number,
    sick_id,
    doctor_name,
    register_data,
    register_time
FROM sick_register
WHERE register_data = CURDATE();

```

register_number	sick_id	doctor_name	register_data	register_time
101	S001234567	Dr. Wang	2025-06-04	早
102	S004567890	Dr. Chen	2025-06-04	中
103	S009876543	Dr. Lin	2025-06-04	晚

3 rows in set (0.001 sec)

圖四、sick_register_today建置內容

10.2.2 today_doctor_schedule_view

用來查詢今日所有醫師的排班資訊，包含排班編號、醫師代號、醫師姓名、排班日期以及對應的診間。透過這個檢視表，管理員能夠即時了解今天有哪些醫師排班、分別在哪間診間看診，作為診間安排、醫師查勤或報表統計的依據。

```

CREATE VIEW today_doctor_schedule_view AS
SELECT
    schedule_number,
    doctor_id,
    doctor_name,
    schedule_data,
    clinic_room
FROM sick_register
WHERE schedule_data = CURDATE();

```

```
MariaDB [clinic_system]> SELECT * FROM today_doctor_schedule_view;
```

schedule_number	doctor_id	doctor_name	schedule_data	clinic_room
1	D001	王小明	2025-06-04	101
2	D002	張大偉	2025-06-04	102

```
2 rows in set (0.001 sec)
```

圖五、today_doctor_schedule_view建置內容

10.2.3 today_sick_basic_info_view

主要提供當日掛號病患的基本資料，包含身份證字號、姓名、性別、出生日期與血型等欄位。透過與 sick_basic 病患基本資料表進行連接，可以讓系統快速篩選出今日有掛號紀錄的病患基本資訊，常用於掛號確認、診間叫號、病歷開立前的身份查驗等場景。

```
CREATE VIEW today_sick_basic_info_view AS
SELECT
    b.sick_id,
    b.sick_name,
    b.sick_gender,
    b.sick_birth,
    b.sick_blood
FROM sick_register r
JOIN sick_basic b ON r.sick_id = b.sick_id
WHERE r.register_data = CURDATE();
```

```
MariaDB [clinic_system]> SELECT * FROM today_sick_basic_info_view
-> WHERE sick_id = 'A123456789';
```

sick_id	sick_name	sick_gender	sick_birth	sick_blood
A123456789	林小芳	F	1995-05-15	0

```
1 row in set (0.001 sec)
```

圖六、today_sick_basic_info_view建置內容

10.2.4 today_doctor_pending_patients_view

統計了每位醫師今天的病患人數，有助於了解目前候診狀況與病患分布情形。系統透過計算 sick_register 表中對應日期的紀錄數量，來統整各醫師目前排隊中的人數，便於診所進行資源安排與叫號調度。

```
CREATE VIEW today_doctor_pending_patients_view AS
SELECT
  doctor_name,
  COUNT(*) AS pending_patient_count
FROM sick_register
WHERE register_data = CURDATE()
GROUP BY doctor_name;
```

```
MariaDB [clinic_system]> SELECT * FROM today_doctor_pending_patients_view;
```

doctor_name	pending_patient_count
王小明	2
陳美娟	2

```
2 rows in set (0.001 sec)
```

圖七、today_doctor_pending_patients_view建置內容

10.2.5 duplicate_register_check_view

偵測病患是否於同一天重複掛號的檢視表。系統會根據病患的身份證字號與掛號日期進行統計，當發現同一位病患在某天掛號超過一次時，便會列出來供管理員或醫師查核，有助於防止誤掛號或重複收費等問題。

```
CREATE VIEW duplicate_register_check_view AS
SELECT
    sick_id,
    register_data,
    COUNT(*) AS count_per_day
FROM sick_register
GROUP BY sick_id, register_data
HAVING COUNT(*) > 1;
```

```
MariaDB [clinic_system]> SELECT * FROM duplicate_register_check_view;
+-----+-----+-----+
| sick_id | register_data | count_per_day |
+-----+-----+-----+
| A123456789 | 2025-06-04 | 3 |
| C333333333 | 2025-06-03 | 2 |
+-----+-----+-----+
2 rows in set (0.001 sec)
```

圖八、duplicate_register_check_view建置內容

十一、心得

林詩宸：經過這次的期末報告，讓我學會了ER Diagram與DB Schema在資料庫規劃中的重要性和差異。在進行ER Diagram與DB Schema的設計，我按照老師教得下去繪畫，畫完後讓幾位同學觀看並建議有哪裡需要修改，會比較完善、清楚，讓我更理解整個系統的邏輯架構以及它們之間的轉換關係，ER Diagram就像是在規劃整個系統的地圖，幫助我把資料和它們的關係先想清楚；而 DB Schema 則是把這些規劃真正變成資料表，讓整個系統能運作起來，使我收穫頗多。

張祐慈:透過這次的報告,我學會了如何根據實際需求設計資料表, 用SQL 建立多個 VIEW 來簡化常用查詢,例如病患掛號紀錄查詢、可預約醫師與時段查詢等。這不僅提升了查詢效率,也讓前端使用上更清楚直觀,是這次學習中很有收穫的一部分。

陳郁庭:在這次的期末報告中,讓我了解如何去規劃資料庫內的資料表,也更了解SQL語法,並根據功能需求,將資料表與資料表相連,也意識到了完整性限制的重要性,像是填寫生日不可填寫未來日期,這一點都不符合邏輯,資料庫得規劃可以很簡單,也可以很複雜,取決於你的系統要怎麼做。

盧奕岑:透過這次的期末報告,我更深入理解了資料如何從分析、設計到實作的完整流程。特別是在設計VIEW表時,讓我學到如何利用SQL語法來達成資料整合與驗證的目的,提升系統的資料正確性與實用性。此外,經過觀察掛號流程與使用者需求,讓我體會到系統設計不能只著重功能實現,更要顧及使用情境與資料完整性。