慈濟大學醫學資訊系

醫學資訊專題

專題報告書

基於區塊鏈技術實現輕量化電子健康紀錄去中心化之安全傳輸

指導教授：李添福 教授

專題參與人員：何采榛、鄭品吟、張銘傑

中 華 民 國 113 年 12 月 06 日

摘要

隨著遠距醫療的普及，患者健康數據與電子病歷在共享與傳輸過程中的資訊安全問題日益受到關注。區塊鏈技術以其去中心化、不可竄改性及共識機制，成為解決醫療數據安全挑戰的重要工具。然而，傳統區塊鏈運算成本高，對醫療應用的推廣造成了限制。

本專題結合物理不可複製函數（PUF）技術與區塊鏈，提出一套輕量化的安全解決方案，提升數據管理效率與安全性。系統利用 PUF 技術生成唯一的挑戰-反應金鑰對（Challenge/Response），確保具備不可偽冒性與不可否認性，並對病歷數據進行摘要處理，將結果存入區塊鏈，利用其不可竄改性保障病歷完整性。同時，用戶可以透過區塊鏈交易記錄，查詢和驗證電子病歷是否遭到篡改。

該系統提供了一般用戶與醫療人員的專屬功能操作，涵蓋電子病歷的註冊、簽章、驗證、查詢及新增健康檢查紀錄等功能。系統架構由前端（Dart）、後端（C#）、區塊鏈伺服器與 SQL 資料庫構成，並在資料庫中設計了包含用戶登入資訊、患者基本資料及健康檢查記錄的數據表。所有數據在存入資料庫前均進行加密處理，進一步提升隱私保護能力。

整體而言，本專題透過 PUF 技術的隨機性與唯一性，以及區塊鏈的去中心化特性，實現了高效且安全的電子病歷管理，不僅解決了數據完整性與隱私保護問題，也為醫療保健系統的安全性提升提供了一種創新途徑。

關鍵字：區塊鏈、物理不可複製函數、不可偽冒性、不可否認性

目錄

摘要 2
目錄 3
圖目錄 5
表目錄 6
第一章 緒論 7
1.1 背景 7
1.2 目的 8
1.3 專題貢獻 8
1.4 報告架構 8
第二章 先備知識 9
2.1 區塊鏈介紹 9
2.2 PUF 介紹 10
第三章 系統架構與研究方法 11
3.1 系統架構圖 11
3.2 區塊鏈上鏈流程 12
3.3 區塊鏈實作 15
3.2.1 區塊鏈交易實作 17
3.4 功能架構圖 18
3.5 系統流程圖 18
3.6 實作網頁介面 21
3.7 資料庫 27
3.8 開發工具及模式 29
第四章 安全分析與效能分析 30
4.1 安全分析 30
4.2 效能分析 31
第五章 總結 33

5.1 結論 33
5.2 未來展望 33
參考文獻 34

圖目錄

圖 1、使用群體與功能劃分 11
圖 2、系統架構圖 12
圖 3、區塊鏈上鏈流程 14
圖 4、資料驗證流程 14
圖 5、創世區塊設定檔 15
圖 6、啟動私有鏈 16
圖 7、建立帳號 16
圖 8、挖礦 16
圖 9、交易紀錄 17
圖 10、區塊鏈交易紀錄 17
圖 11、功能架構圖 18
圖 12、一般用戶的系統流程 19
圖 13、醫療人員的系統流程 20
圖 14、首頁 21
圖 15、一般用戶登入端 21
圖 16、醫療人員登入端 22
圖 17、一般用戶端登入後畫面 22
圖 18、一般用戶登入提醒 23
圖 19、醫護人員登入後選單介面 23
圖 20、醫護人員查詢病患資料介面 24
圖 21、查無病患資料或者尚未檢查提示 24
圖 22、正常查詢結果 25
圖 23、新增電子病歷的輸入介面 25
圖 24、成功新增提示 26
圖 25、資料庫關聯圖 28
圖 26、SCHEMA 28

表目錄

表格 1、各學者使用方法與本系統的效能比較.....31

第一章 緒論

1.1 背景

網路的發展，遠距醫療為我們的生活帶來了改變，已成為日常生活中重要的一環。遠距醫療數據資料在共享、傳輸過程中的資訊安全也逐漸受到關注。近期惡意的攻擊事件層出不窮，提升患者個人資料的安全及隱私，防範患者病歷資料或健康數據洩漏，進而危害到患者生命安全是被大家所重視的。

近年來，許多醫療保健系統相關的研究應用區塊鏈技術來提升其安全性，確保資料完整性、正確性，並利用複雜的公開金鑰密碼系統與計算沉重的模指數運算及橢圓曲線點乘運算來達到相關規範的安全特性，包含資訊安全與個人隱私。當資料儲存到一定的數量後，需要耗費大量的資源在搜尋、驗證區塊正確性，會使整體效率降低。因此患者健康數據與電子病歷整體資料，從行動裝置、醫療中心伺服器、PHI 資料庫到醫療保健服務提供者，應該嚴謹地受到保護，以提升患者健康紀錄與電子病歷整體資料的安全性，讓醫護人員更加準確地了解患者的身體狀況。

發展安全、有效率、基於區塊鏈技術輕量化運算的醫療保健資訊保護機制，並符合安全/隱私規範，是現今重要的研究與實務議題。

1.2 目的

本專題以物理不可複製函數 (Physically Unclonable Functions, PUF) 技術與區塊鏈技術為核心，希望利用上述這兩個技術去實作一電子健康紀錄系統，讓此系統可以做到降低對電子病歷資料庫系統的可信任度，藉此達到系統應具備之使用者認證、不可偽冒性與不可否認性，提升運算效率，同時達成去中心化的特性。

1.3 專題貢獻

本專題利用物理不可複製函數與區塊鏈達成以下幾點：

一、以物理不可複製函數的特性達成使用者認證、不可偽造性與不可否認性

二、藉由物理不可複製函數降低運算成本，維持輕量化運算

三、以區塊鏈的特性達成去中心化，輔助提升使用者對中央資料庫的可信任度

1.4 報告架構

第二章為先備知識，主要在介紹區塊鏈與物理不可複製函數；第三章為本專題的系統架構與研究方法；第四章為安全分析與效能分析；第五章為本專題的總結。

第二章 先備知識

2.1 區塊鏈介紹

1. 區塊鏈技術是一種進階資料庫機制，允許在業務網路中分享透明的資訊。區塊鏈資料庫會將資料存放在連結於同一鏈的區塊中。資料在時間順序上具有一致性，因為在無網路共識的情況下，您不能刪除或修改此鏈。因此，您可以使用區塊鏈技術建立不可更改或不可變的總帳，以追蹤訂單、付款、帳戶以及其他交易。[1]

2. 區塊鏈分為三種不同的模式[2]

   i. 公有鏈：顧名思義，公有鏈就是大眾共同擁有的區塊鏈，任何人想要加入或離開區塊鏈都不需要經過審查。

   ii. 私有鏈：私有鏈是指權限由某一個組織或機構掌握的區塊鏈，只有經過授權的節點才能參與該條區塊鏈。

   iii. 聯盟鏈：聯盟鏈的使用僅限於聯盟內成員，聯盟可以由組織、機構等共同來建立，並在建立時協商他們聯盟的規範。聯盟鏈的開放程度介於公有鏈和私有鏈之間，不像公有鏈一樣的開放，但也不像私有鏈一樣只允許擁有該條區塊鏈的組織或機構使用。

3. 區塊鏈特性[2]

   i. 去中心化：區塊鏈中的去中心化是指將控制和決策從集中式實體 (個人、組織或團體) 轉移到分散式網路。去中心化區塊鏈網路使用透明度來降低參與者之間對信任的需求。這些網路還會阻止參與者以降低網路功能的方式相互施加權威或控制。

   ii. 匿名性：讓區塊鏈系統中的節點得以沒有寫上名字參與其中，節點使用一組「英文＋數字」的代碼作為名稱，只要不跟別人透露，就沒有人能夠知道節點背後的人是誰。

   iii. 不可竄改性：不可篡改性意味著某些東西無法變更或更改。一旦有人將交易記錄到共享總帳中，任何參與者都不能竄改交易。如果交易記錄包含錯誤，您必須新增新的交易以修正錯誤，並且這兩個交易對網路都是可見的。

iv. 講求共識：區塊鏈是最鮮明與最民主的機制。每一種區塊鏈系統都會採用某一種共識演算法。

2.2 PUF 介紹

1. 物理不可複製函數（Physically Unclonable Functions, PUF）是一種硬體安全技術，透過半導體製程引入的隨機變數，以及挑戰/反應數據庫（Challenge / Response）的建立，使 PUF 達到隨機性、唯一性及不可複製性。[5]
2. PUF 的特性：
   i. Randomness（隨機性）
   ii. Uniqueness（唯一性）
   iii. Reliability（可靠性）
   iv. Correctness（正確性）

基於 PUF 的唯一性與不可複製性，可以達到不可否認、不可偽冒的特性，做到身分驗證。

第三章 系統架構與研究方法

本專題的實作部分主要可分成兩大使用群與四個功能小區塊。

其中，兩大使用群指的是一般用戶與醫療體系內部人員；而四個小區塊則是註冊、驗證、查詢與新增健康檢查紀錄。

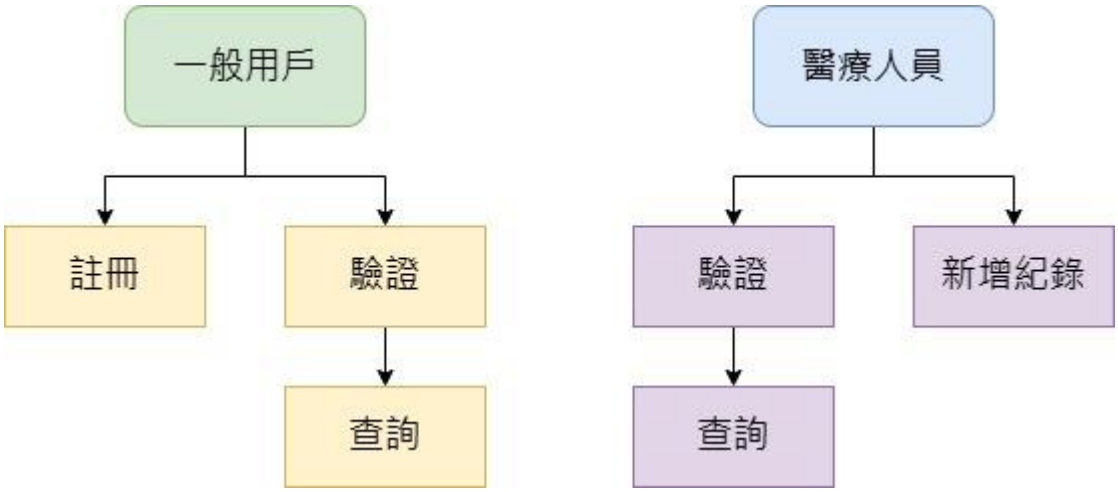


圖 1、使用群體與功能劃分

3.1 系統架構圖

圖 2 為本專題所使用之系統架構圖，由醫護人員與一般使用者為前端、C# 為後台、Blockchain Server 與 SQL Server 構成。

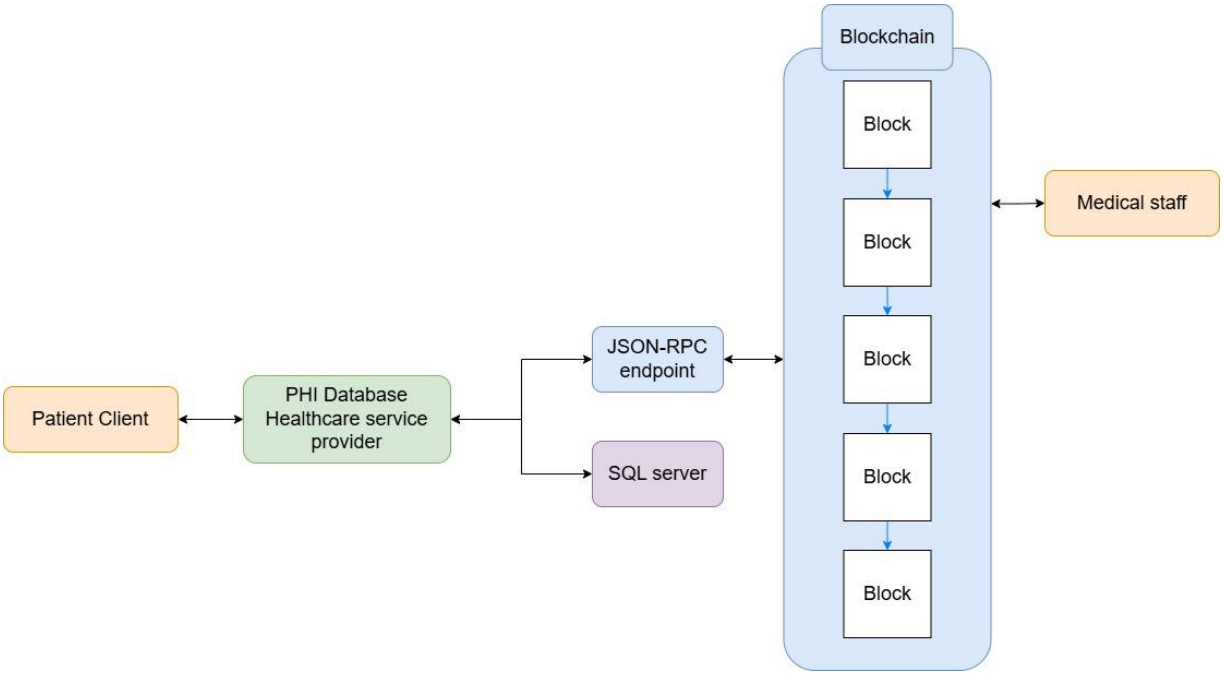


圖 2、系統架構圖

3.2 區塊鏈上鏈流程

本專題系統利用基於 PUF 的唯一性與不可複製性，達到不可否認、不可偽冒的特性，做到身分驗證。本專題系統包含：註冊階段、病歷摘要上傳階段、資料驗證，詳細描述如下：(如圖 9 與圖 10)

- 註冊階段

  Step 1：使用者計算 K_1=PUF(Random,ID)並向 RC 註冊 ID,h(ID, K_1)。RC 幫使用者上傳 ID,h(ID, K_1)至 Blockchain。

  Step 2：將 Random 上傳至 Block-Chain。

- 病歷摘要上傳階段

  Step 1：取用 Block-Chain 上的 Key。

  Step 2： [初次上傳]將病歷和 K_1 結合。

  [非初次上傳]對 K_{n-1} 做 PUF 運算產生 K_n，並將病歷和 K_n 結合。

Step 3：對結合後的資訊做 SHA256 形成摘要。

Step 4：將摘要再次做 SHA256 並上傳至 Block-Chain。

Step 5：將 K_n 公開上傳至 Block-Chain。

- 資料驗證

Step 1：用 ID 先找到電子病歷(c)以及他在區塊鏈上存放的 hash 地址（hc）。

Step 2：用 hc 在區塊鏈上找到 kc。

Step 3：對電子病歷與 ID 做 SHA256 來取得 k'。

Step 4：將 k'與 kc 作對比，若兩者相同則代表資料未被更動；若兩者不同則代表資料已被 Super User 更改過。

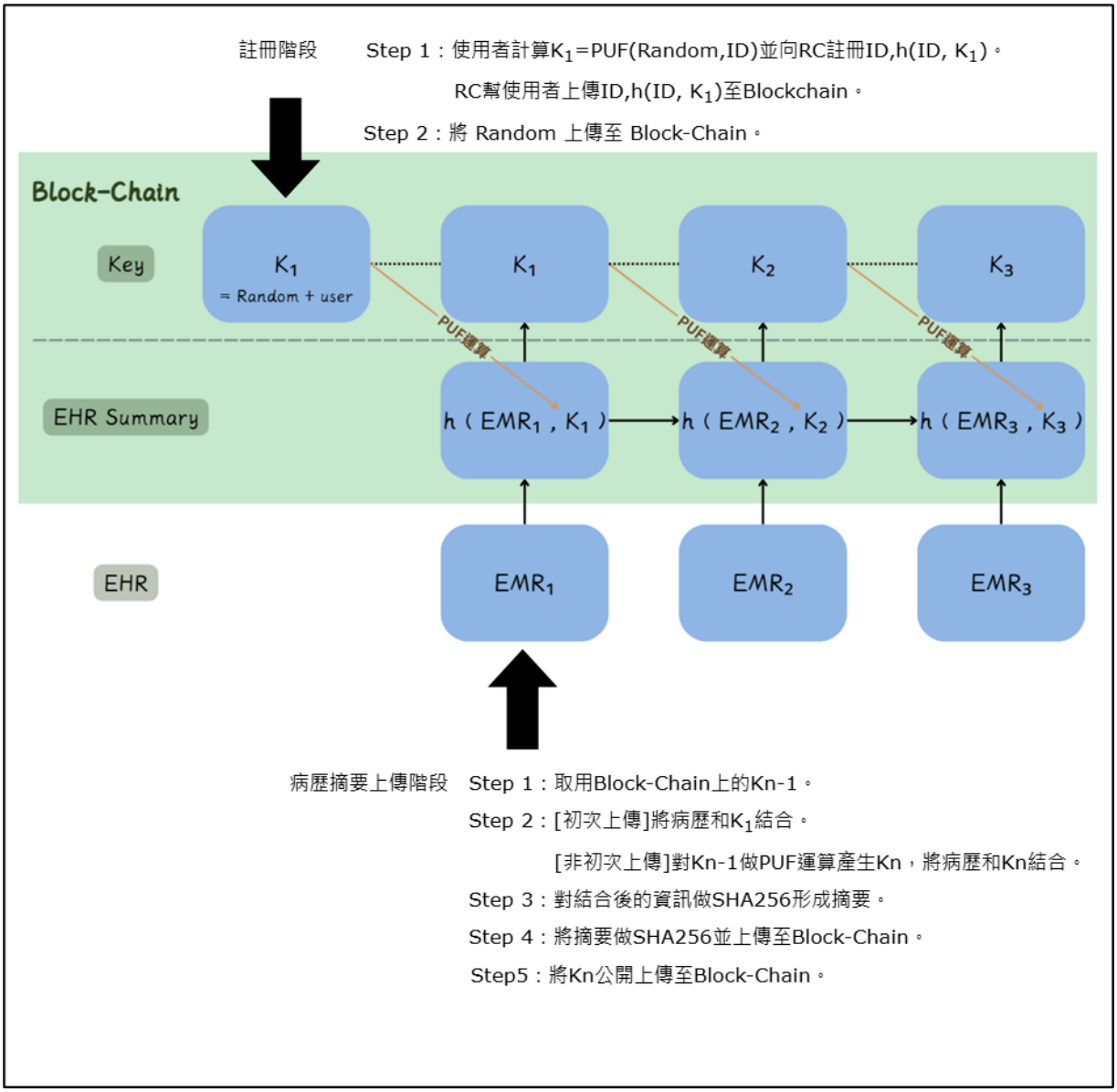


圖 3、區塊鏈上鏈流程

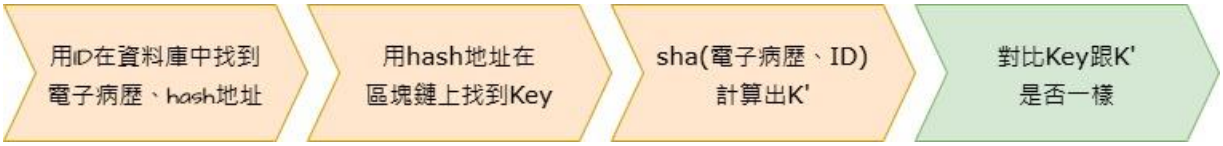


圖 4、資料驗證流程

3.3 區塊鏈實作

本專題是在本地端 Linux 電腦架設以太坊私有鏈，在以太坊私有鏈中設有兩個帳號，分別為發送交易帳號和挖礦的帳號，另外一個是接收交易的帳號，以太坊私有鏈在運作時，挖礦的帳號會執行挖礦，把之前發送的交易進行驗證，並打包到鏈上，並且礦工成功挖出一個新區塊時，它會將獎勵推送到配置的帳戶中。[3][4]

在一開始建置前我們先設定了一個創世區塊（圖 5），目的是為了在沒有任何區塊被產生時當作第一個區塊，裡面包含參數 chainID:區塊鏈的識別 ID，不能與公有鏈產生衝突所以避免使用（1~10），homesteadBlock：以太坊版本，值等於 0 代表正在使用、eip155Block,eip158Block：區塊鏈分叉提議，設為 0 表示不會干涉私有鏈、difficulty：挖礦的難度，coinbase：預設第一個帳號為礦工、timestamp：創立區塊時間，gasLimit：交易所使用到的 gas 限制、alloc：配置帳戶預設的以太幣。

圖 5、創世區塊設定檔

再來我們在設備使用啟動指令私有鏈（圖 6），指令 geth：啟動 Geth 客戶端、--datadir：指定區塊儲存的資料夾、--networkid：設定以太坊的網路，欲連接一個私有鏈必須設定相同，--rpc：啟動 rpc 通訊協定 HTTP-RPC，進行智能合約的部屬跟呼叫、--rpcaddr "0.0.0.0"：將 RPC 伺服器綁訂到所有可用的網路接口，0.0.0.0 代表允許任何設備連接、--nodiscover：關閉自動同步其他節點、--rpcapi：可連接的客戶端

應用、--allow-insecure-unlock：允許通過 HTTP-RPC 解所帳號、console：啟動交互式 JavaScript 控制台，可於節點進行值交互。

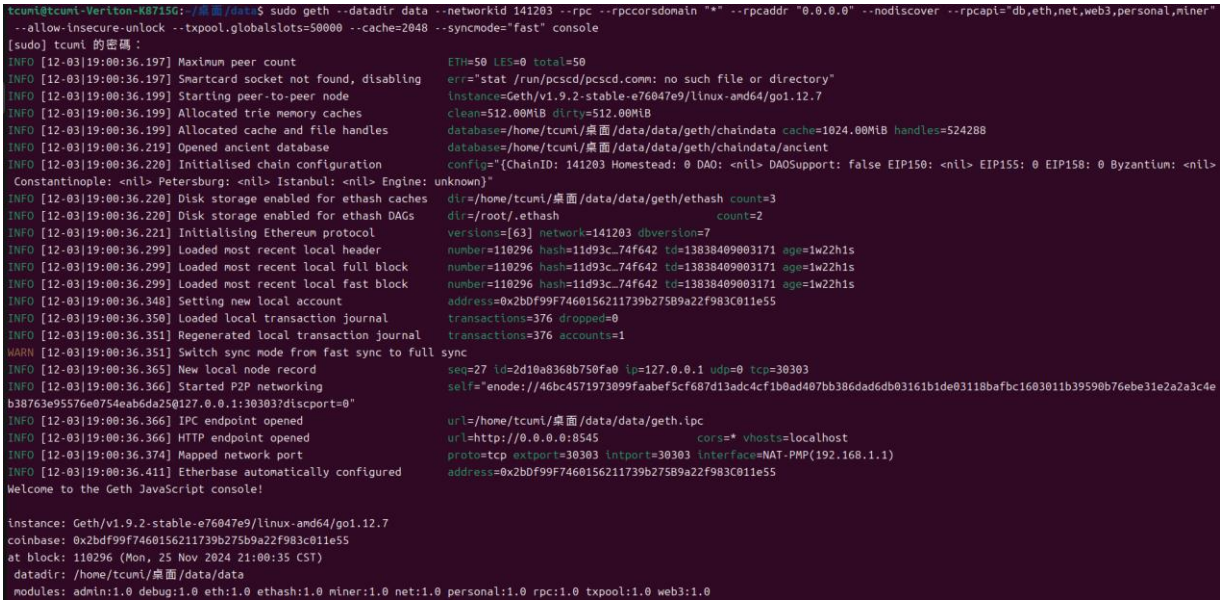


圖 6、啟動私有鏈

建立礦工帳號，在建立帳號時可以把密碼預先輸入在參數，也可以執行之後再把密碼輸入。

圖 7、建立帳號

輸入指令後挖礦帳號會進行挖礦，當挖到區塊時返還的獎勵會推送獎勵至配置的帳戶，其中括號裡面的是挖礦時所用的核心數。

圖 8、挖礦

在交易紀錄裡面我們發現在交易的格式裡面有一欄是關於交易備註的，這時我們就想說是不是可以把電子病歷摘要跟 PUF_KEY 放在備註欄裡。

圖 9、交易紀錄

3.2.1 區塊鏈交易實作

此系統是使用 Web3.py 的函式庫去實現跟以太坊節點互動，透過 HTTP-RPC 協定去跟以太坊私有鏈做通信。

摘要上傳區塊鏈

將電子健康紀錄（EHR）跟 PUF_Key 數據經過 SHA-256 轉換成哈希值，並把轉換後的資料放進交易格式裡的備註欄（input）再發送交易，發送交易後會回傳交易的哈希值（圖 10）。

圖 10、區塊鏈交易紀錄

交易查詢

從資料庫拿到該病人的交易哈希值，利用哈希值查詢區塊鏈上交易時所夾帶的資料。

3.4 功能架構圖

圖 11 為系統功能架構圖，其功能包含：一般使用者註冊、電子病歷簽章、電子病歷驗證、電子病歷查詢、新增電子病歷等功能。

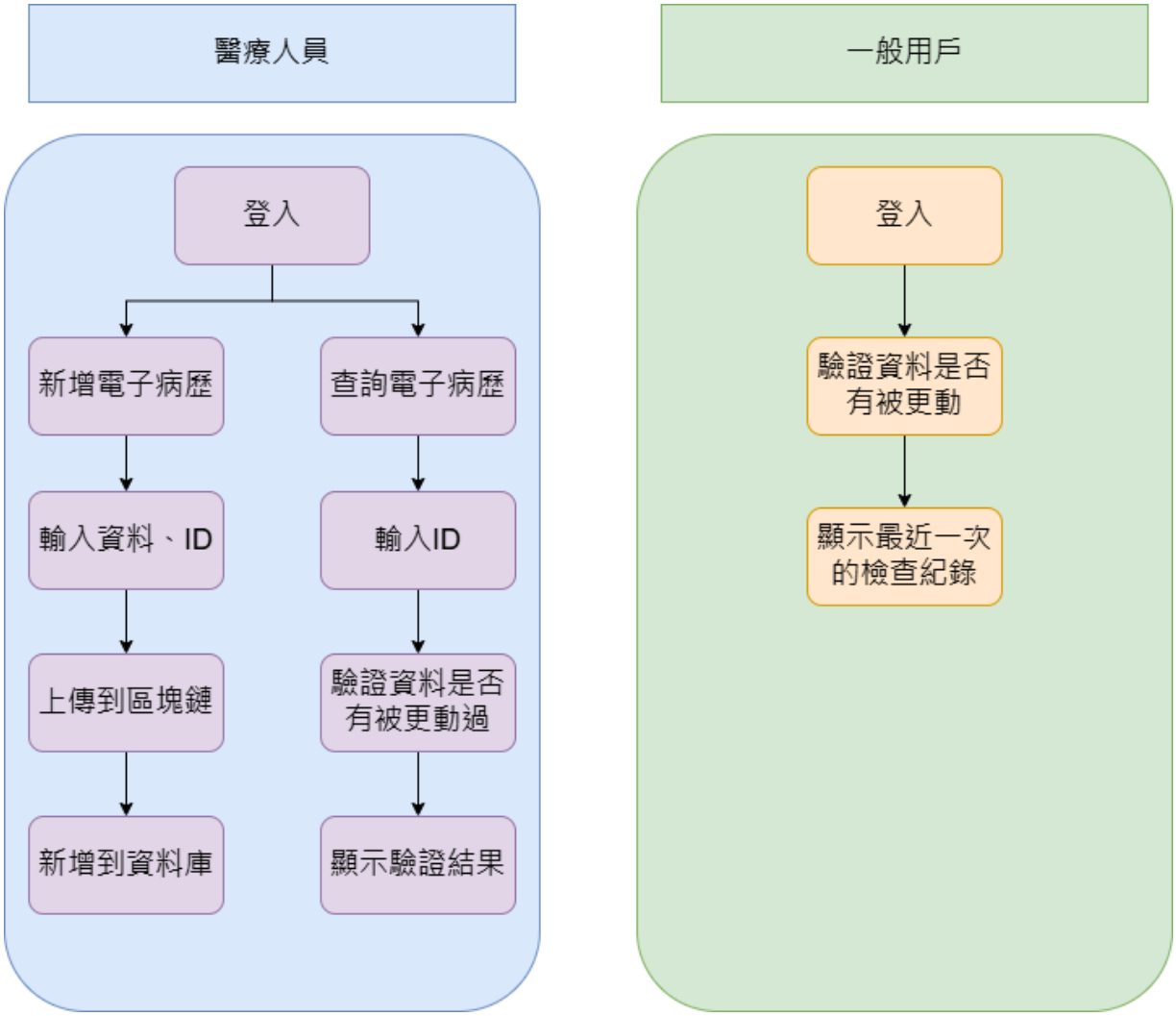


圖 11、功能架構圖

3.5 系統流程圖

圖 12 為一般用戶的系統流程圖，圖 13 為醫療人員的系統流程圖。

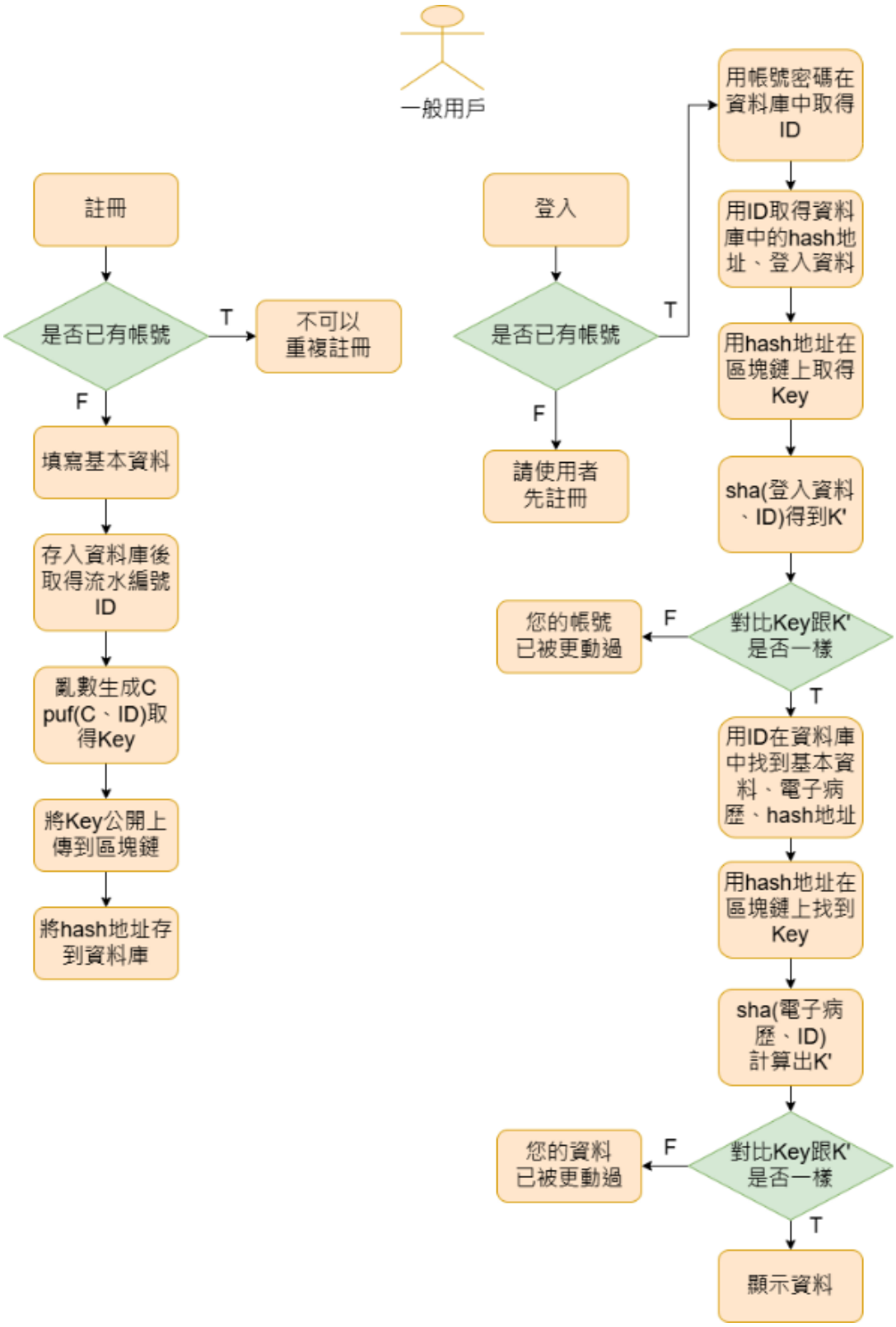


圖 12、一般用戶的系統流程

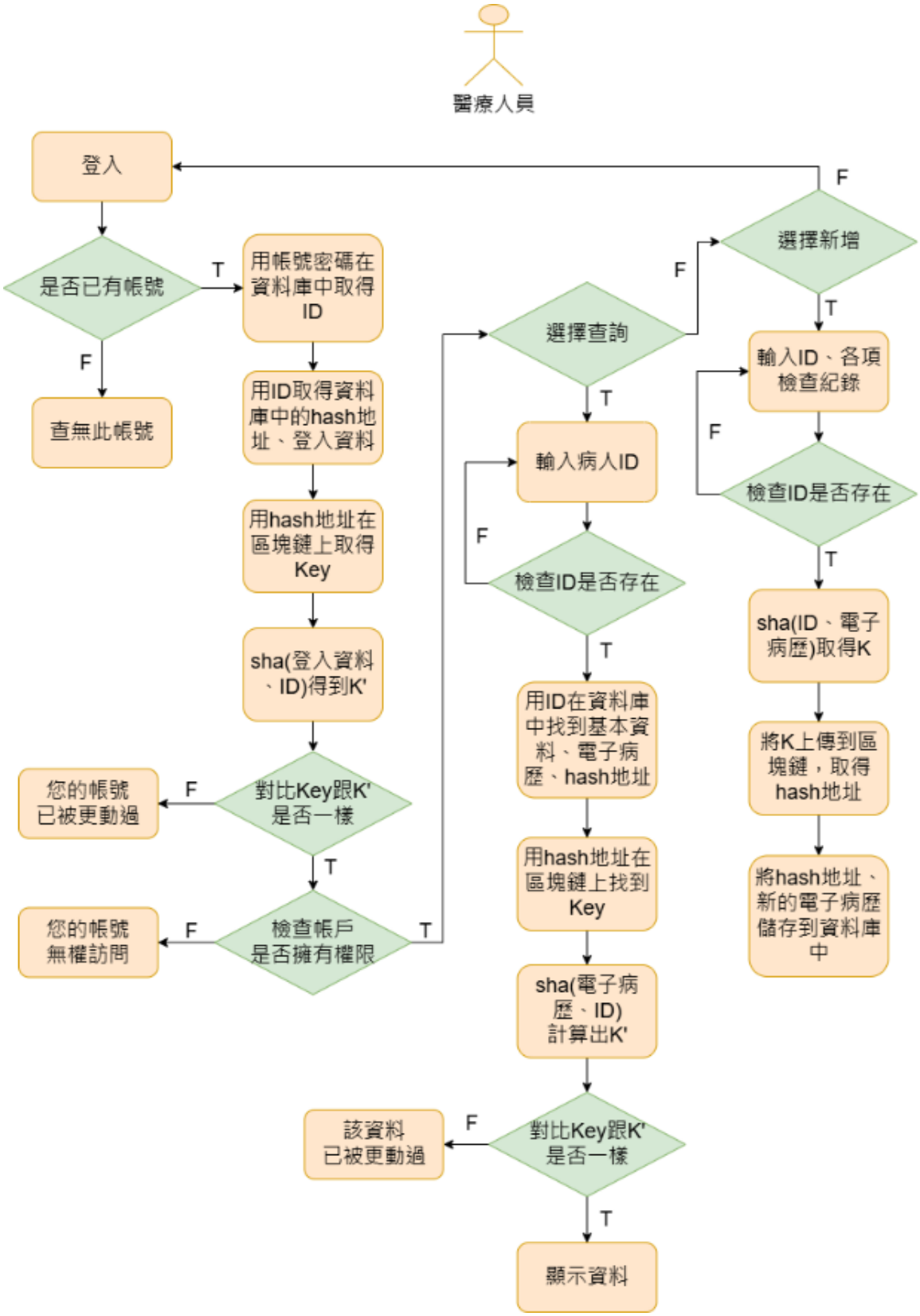


圖 13、醫療人員的系統流程

3.6 實作網頁介面

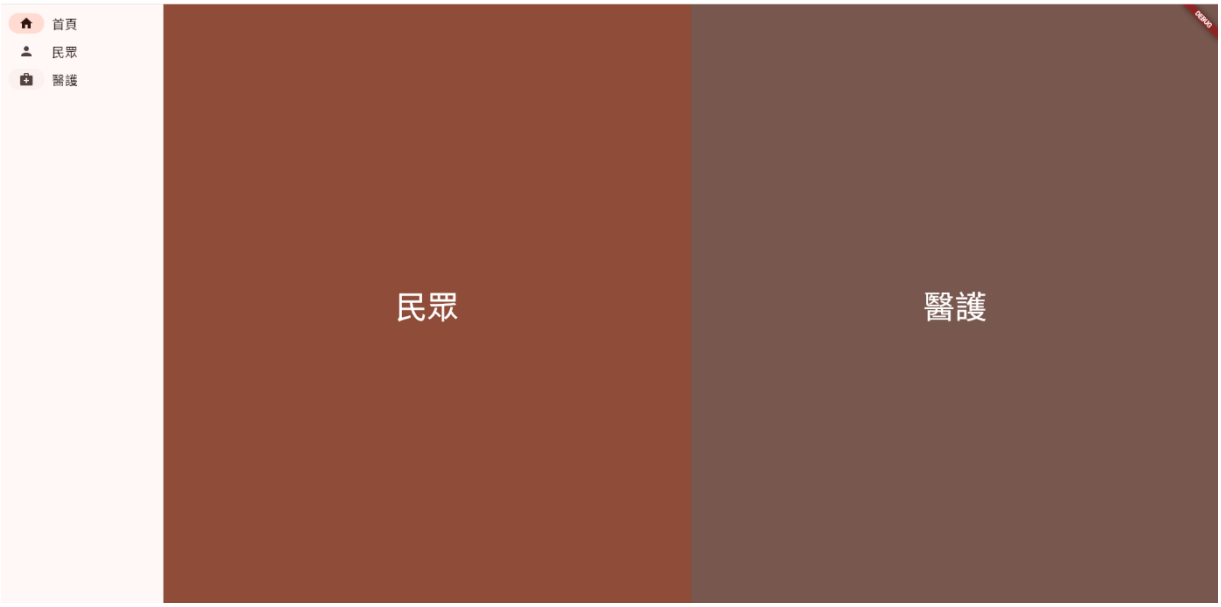


圖 14、首頁

圖 15、一般用戶登入端

圖 16、醫療人員登入端

一般用戶登入後，若是目前還沒有檢查紀錄，系統以彈窗形式提醒；若是資料已被 SA 更動，則以彈窗形式提醒使用者在資料庫的資料已遭異常手段更改。

圖 17、一般用戶端登入後畫面

圖 18、一般用戶登入提醒

圖 19、醫護人員登入後選單介面

圖 20、醫護人員查詢病患資料介面

若醫療人員輸入無效 ID 或者該病患當前並無健檢紀錄，則顯示查詢失敗。

圖 21、查無病患資料或者尚未檢查提示

圖 22、正常查詢結果

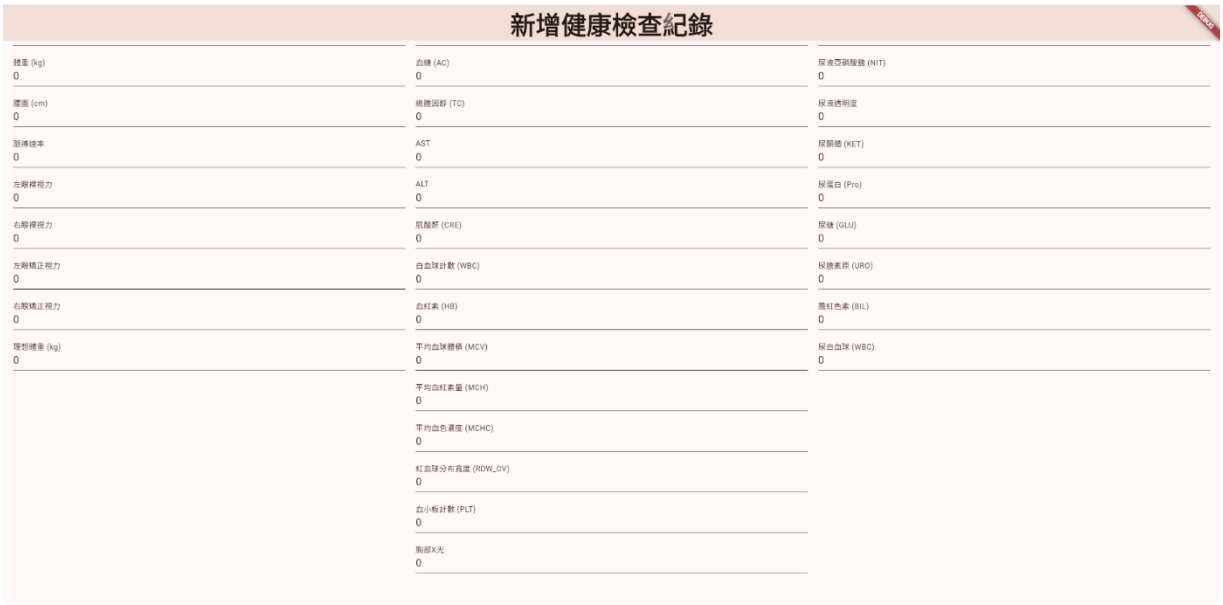


圖 23、新增電子病歷的輸入介面

新增時，會在醫療人員點下新增紀錄後檢查是否有欄位空缺，若有欄位未填寫，則會提醒醫療人員記錄未填寫完整。

圖 24、成功新增提示

3.7 資料庫

在本專題中，我們使用了三個資料表來儲存欄位，分別是 login、patient、his_data，且因為後臺在將資料存入前都會進行加密（除了 ID）以保障資料隱私，故使用的欄位都是 varchar 型態。

- patient：主要用來儲存個人基本資料，其中包含 Patient_id、P_name、Phone、Address、Personal_H、Family_H、H_address。
  - Patient_id：儲存病人 ID 的欄位，為 patient 表的 primary key。
  - H_address：儲存區塊鏈回傳之地址的欄位。
- login：主要用來儲存登入資料，其中包含 patient_id、account、password、privacy、K_address、H_address。
  - patient_id：儲存病人 ID 的欄位，為 login 表的 primary key，且同時為 foreign key，他將會參照到 Patient 表中的 Patient_id 欄位。
  - privacy：儲存權限的欄位。
  - K_address：儲存區塊鏈上公開的 key 的地址。
  - H_address：儲存區塊鏈上計算後的 response。
- his_data：主要用來儲存健康檢查紀錄。
  - patient_id：儲存病人 ID 的欄位，為 his_data 表的 primary key，且同時為 foreign key，他將會參照到 patient 表中的 Patient_id 欄位。
  - K_address：儲存區塊鏈上公開的 key 的地址。
  - H_address：儲存區塊鏈上計算後的 response。

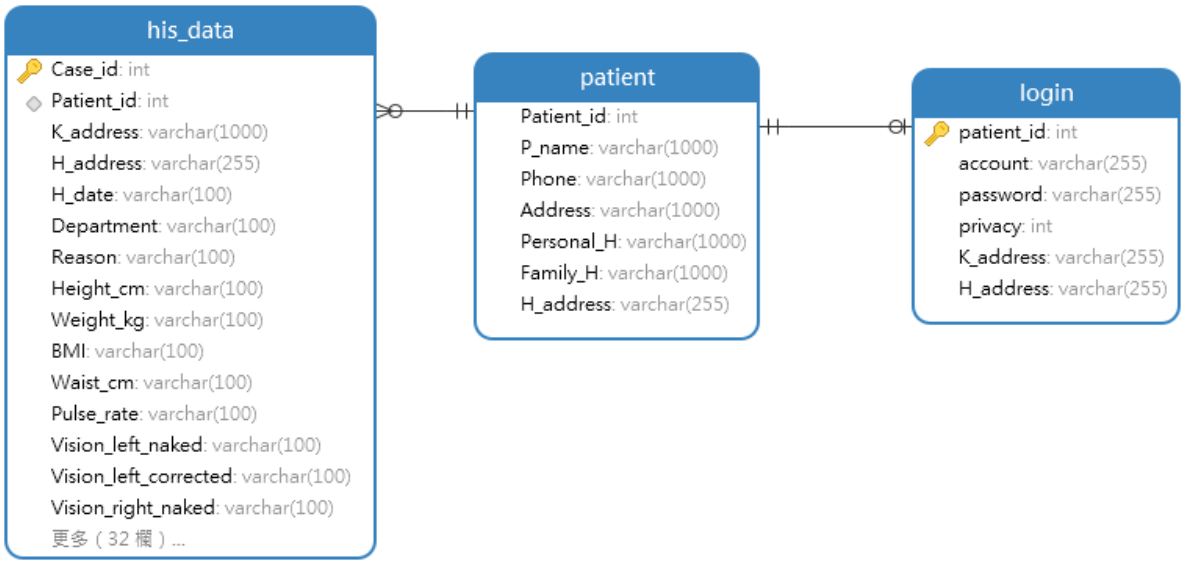
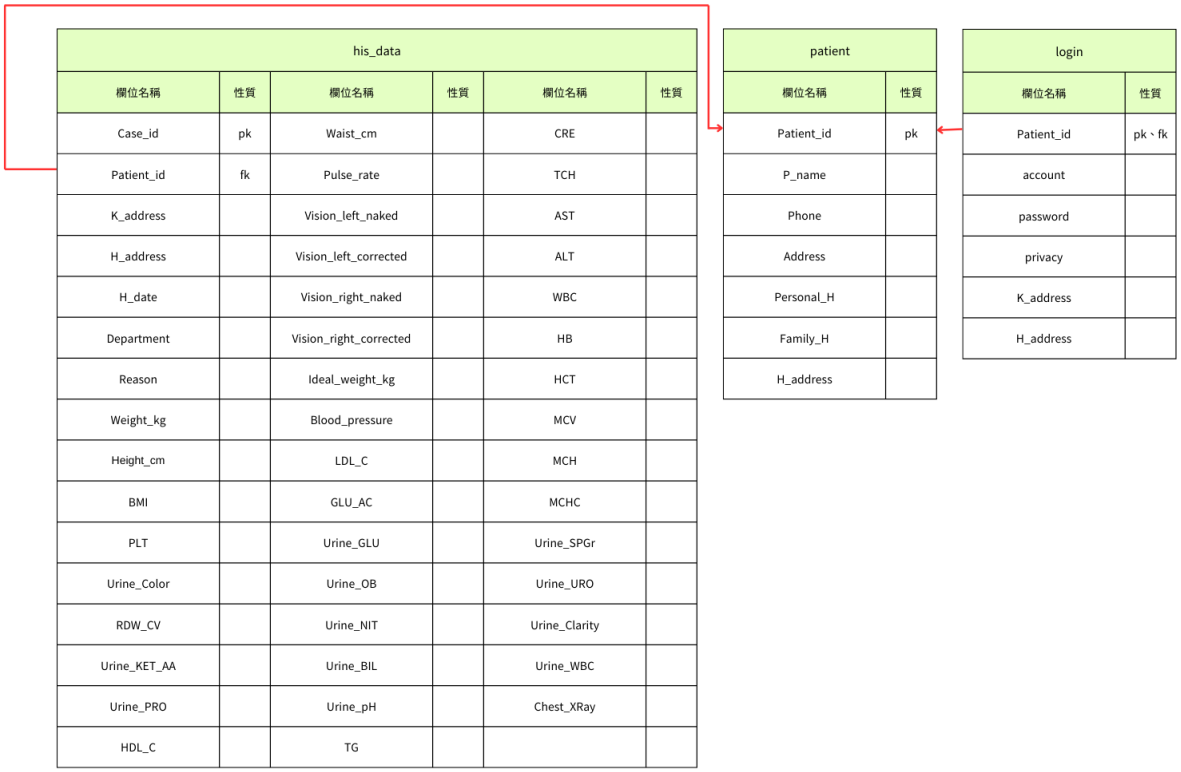


圖 25、資料庫關聯圖



his_data					
欄位名稱	性質	欄位名稱	性質	欄位名稱	性質
Case_id	pk	Waist_cm		CRE	
Patient_id	fk	Pulse_rate		TCH	
K_address		Vision_left_naked		AST	
H_address		Vision_left_corrected		ALT	
H_date		Vision_right_naked		WBC	
Department		Vision_right_corrected		HB	
Reason		Ideal_weight_kg		HCT	
Weight_kg		Blood_pressure		MCV	
Height_cm		LDL_C		MCH	
BMI		GLU_AC		MCHC	
PLT		Urine_GLU		Urine_SPGr	
Urine_Color		Urine_OB		Urine_URO	
RDW_CV		Urine_NIT		Urine_Clarity	
Urine_KET_AA		Urine_BIL		Urine_WBC	
Urine_PRO		Urine_pH		Chest_XRay	
HDL_C		TG			

patient	
欄位名稱	性質
Patient_id	pk
P_name	
Phone	
Address	
Personal_H	
Family_H	
H_address	

login	
欄位名稱	性質
Patient_id	pk、fk
account	
password	
privacy	
K_address	
H_address	

圖 26、schema

3.8 開發工具及模式

- 區塊鏈
  - 作業系統 Ubuntu 24.04 LTS
  - 語言 GO 1.12.7
  - Geth v1.9.2
  - Python 3.11.9
- PUF
  - Python 3.11.9
- 前端網頁
  - 框架 Flutter 3.24.5
  - 語言 Dart
- 後端伺服器
  - 框架 ASP.NET Core
  - 語言 C#
- 開發環境
  - Visual Studio、Visual Studio Code
- 資料傳輸、資料庫儲存使用的加解密配置
  - Mode : AES CBC
  - Padding : PKCS7

第四章 安全分析與效能分析

4.1 安全分析

i. 使用者認證 User Authentication：使用者 Ui 向 RC 註冊其 PUF 之 Challenge-Response Pair (CPR)中之 C_0 , $h(R_0)$。且 RC 將資訊 $h(ID, K_1)$ 上傳至 Block-Chain，只有使用者 Ui 可計算出電子病歷之簽章金鑰 $R_0 = K_1 = PufUi(C_0)$。因此，Ui 公布出 K_1 可做為使用者 Ui 之身分認證。

ii. 不可偽造性 Unforgeability：所提機制利用 Block-chain 不可偽造與不可竄改的特性，結合 PUF 所算出來的變數放上區塊鏈，讓其更無法偽造。

iii. 不可否認性 Non-repudiation：所提機制利用 PUF 與 Block-chain 技術產生不可偽造與不可竄改的特性，因此具備其不可否認特性。

iv. 傳輸安全 Transport security：在前端網頁傳輸資料給後臺前、後臺與資料庫交互時，分別用了兩套不同的 IV、金鑰進行 AES 的加密傳輸，以保障資料傳輸的安全性。

v. 機密性 Confidentiality：本系統採用 AES 加密技術來確保電子病歷在傳輸過程中的機密性。

vi. 去中心化 Decentralization：透過區塊鏈去中心化的特性，本專題輔助提升了使用者對中央資料庫的可信任度。病歷摘要與密鑰記錄於區塊鏈交易中，所有交易需經過共識機制進行驗證，避免單一節點或系統管理者對數據的操控與修改。

4.2 效能分析

表格 1、各學者使用方法與本專題系統的效能比較

Schemes/Efficiency	PSN area/運算成本	反應時間(ms)	去中心化
Zhang 等學者 2016 [7]	3 T_h + 2 T_e	0.05591	No
Li 等學者 2018 [8]	1 T_h +2 T_s	0.02461	No
Lee 等學者 2020 [9]	1T_h + 1T_s + 1T_e	0.03381	No
本系統	1T_h +1 T_p+2 T_s	0.02864	Yes

T_h: 一次單向雜湊函數計算所需時間，0.00645 毫秒
T_p: 一次 PUF 計算所需的時間，1/1.6 * T_h = 0.00404 毫秒 [10-13]
T_s: 一次對稱式加解密計算所需時間， 0.00908 毫秒
T_e: 一橢圓曲線點乘計算所需時間，0.01828 毫秒[14]

表格 1 列出各學者使用方法與本專題系統的效能比較。本專題在運算成本、反應時間與去中心化特性上具有以下優勢與特點：

1. 運算成本與反應時間

Zhang 等學者[7]與 Lee 等學者[9] 需要耗時的橢圓曲線點乘運算 T_e；相較於 Li 等學者 [8]與本專題系統，利用等輕量化運算 T_h 、1 T_p、2 T_s 發展，因此 Zhang 等學者[7]與 Lee 等學者[9]需要更多的反應時間。

本專題系統反應時間僅稍高於 Li 等學者的 0.02461 毫秒。
反應時間的提升主要得益於：

- PUF 的高效性 (T_p)。
- 使用對稱加密 (T_s) 代替較高成本的橢圓曲線運算 (T_e)。

2. 去中心化特性

與其他三種方法相比，本系統是唯一實現去中心化特性的方案：

- 數據透過區塊鏈存儲與驗證，避免了對中央伺服器的依賴。
- 去中心化架構增強了系統的透明性與數據安全性。

3. 整體結論

本系統在保持去中心化特性的同時，實現了高效的運算性能與較短的反應時間：

- 適合應用於輕量級數據傳輸與驗證場景，如電子病歷管理。
- 與傳統方法相比，具有更高的安全性與運算效率平衡。

第五章 總結

5.1 結論

本專題透過區塊鏈及 PUF 的特性，輔助提升使用者對中央資料庫系統的可信任度，達到使用者認證、不可偽冒、不可否認、資料完整性、去中心化的特性，並提升使用者之電子健康紀錄的傳輸安全性。

5.2 未來展望

在未來希望可以定期把以太坊私有鏈其中的區塊上傳到公有鏈，這樣更可以保證去中心化，在之後去計算哈希值時也可以跟主鏈上的哈希值做對比。同時藉由定期放上主鏈這個動作，不僅大幅減少了花費，也使區塊鏈的安全性更增加了幾分。

本系統已做到病歷個資資料被攻擊者異動竄改可被偵測出，包含具特殊權限的系統管理者 SA，現階段系統仍未能做到系統管理者 SA 之瀏覽權限的限制。未來將導入存取控制機制來限制特殊權限使用者的權限，做到完整的權限控管與限制。

參考文獻

[1] 什麼是區塊鏈技術？: https://aws.amazon.com/tw/what-is/blockchain/?aws-products-all.sort-by=item.additionalFields.productNameLowercase&aws-products-all.sort-order=asc

[2] 你不可不知區塊鏈的五大特色: https://medium.com/cobinhood-中文報/你不可不知-區塊鏈的五大特色-fef29c1b90b6

[3] 用 Geth 架設私有鏈: https://medium.com/samumu-clan/用-geth-架設私有鏈-41a2baa0efd8

[4] BlockChain - 私有鏈建立 (Geth & Node.js): https://dotblogs.com.tw/explooosion

[5] 無法複製的晶片指紋——PUF 技術 科技大觀園: https://scitechvista.nat.gov.tw/Article/C000003/detail?ID=6f18c7f5-80e6-4183-b928-28c77263f7a7

[6] 郭咨均，Blockchain-based Keyed-hash Digital Signature Scheme，國立成功大學，碩士論文 2021。

[7] Zhang,J.,Xue, N., Huang, X.: A Secure system for pervasive social network-based healthcare. IEEE Accesss 4, 9239–9250 (2016)

[8] Li, H., Zhu, L., Shen, M., Gao, F., Tao, X., Liu, S.: Blockchainbased data preservation system for medical data. J. Med. Syst. 42(8), 141 (2018)

[9] Lee,T.F.,Li H.-Z., Hsieh, Y.-P., A Blockchain-based Medical Data Preservation Scheme for Telecare Medical Information Systems,

International Journal of Information Security, Published online, Aug. 2020. DOI 10.1007/s10207-020-00521-8

[10] Gope P, Das AK, Kumar N, Cheng Y. Lightweight and physically secure anonymous mutual authentication protocol for real-time data access in industrial wireless sensor networks. IEEE transactions on industrial informatics 2019;15(9):4957–68

[11] Lee, T.F., Chen, W.Y., "Lightweight Fog Computing-based Authentication Protocols Using Physically Unclonable Functions for Internet of Medical Things," Journal of Information Security and Applications, Vol. 59, June 2021, 102817.

[12] P. Gope, LAAP: Lightweight anonymous authentication protocol for D2D-Aided fog computing paradigm. computers & security, 86, 223-237 (2019).

[13] W. Bian, P. Gope, Y. Cheng, Q. Li, Bio-AKA: An efficient fingerprint based two factor user authentication and key agreement scheme, Future Generation Computer Systems, 109, 45-55 (2020)

[14] Handbook of Applied Cryptography (Menezes, van Oorschot, and Vanstone)