

國立臺灣大學電資工程學院資訊工程所

碩士論文

Department of Computer Science and Information Technology

College of Engineering

National Taiwan University

Master Thesis

基於 OurChain 的自主身分系統設計與實作

Design and Implementation of Autonomous Identity  
System Based on OurChain

林俊佑

Jun You, Lin

指導教授：薛智文 博士

Advisor: Chih-Wen (Steven) Hsueh Ph.D.

中華民國 113 年 7 月

July, 2024

國立臺灣大學碩士學位論文

口試委員會審定書



基於 OurChain 的自主身分系統設計與實作

Design and Implementation of Autonomous Identity  
System Based on OurChain

本論文係林俊佑君（R11922114）在國立臺灣大學資訊工程所完成之碩士學位論文，於民國 113 年 7 月 25 日承下列考試委員審查通過及口試及格，特此證明

口試委員：\_\_\_\_\_

（指導教授）

_____	_____
_____	_____
_____	_____
_____	_____

所 長：\_\_\_\_\_



[illegible]





## 摘要

當前的數位身分管理系統正面臨危機。大型科技公司和政府機構對個人數據的壟斷，不僅侵蝕了使用者的數位自由，更導致了嚴重的隱私問題。使用者被迫在獲取服務與保護個人數據間做出抉擇，而他們的數位足跡不斷被收集、分析和利用，往往超出同意的範圍。這種失衡的權力結構不僅威脅個人自由，更動搖了整個社會的道德基礎。

本研究直面這些挑戰，提出了一種革命性的解決方案——自主身分（AID）系統。AID 系統希望從根本上重塑數位身分管理的權力結構，將主權從壟斷者轉移到個體使用者手中。通過創新的架構和先進的區塊鏈技術，AID 系統不僅為使用者提供了身份的自由，還引入了基於共識的動態道德準則，通過建立公平的遊戲規則來抑制強者的壟斷行為。

AID 系統作為一項顛覆性技術，其廣泛採用仍面臨諸多挑戰，包括來自既得利益者的阻力以及社會認知方面的障礙。儘管如此，本研究為解決數位時代最緊迫的身份管理問題開闢了新的方向。

**關鍵字：**自主身分、管理、認證、隱私、區塊鏈





# Abstract

Abstract Abstract Abstract Abstract Abstract Abstract Abstract Abstract Abstract Ab-  
stract Abstract Abstract Abstract Abstract Abstract Abstract Abstract Abstract Abstract  
Abstract Abstract Abstract Abstract Abstract Abstract Abstract Abstract Abstract Abstract  
Abstract Abstract Abstract Abstract Abstract Abstract Abstract Abstract Abstract Abstract  
Abstract Abstract Abstract Abstract Abstract Abstract Abstract Abstract Abstract Abstract  
Abstract Abstract Abstract Abstract Abstract Abstract Abstract Abstract Abstract Abstract  
Abstract Abstract Abstract Abstract

**Keywords:** Autonomous Identity, Management, Authentication, Privacy, Blockchain

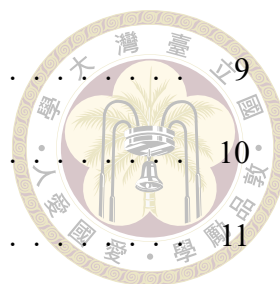






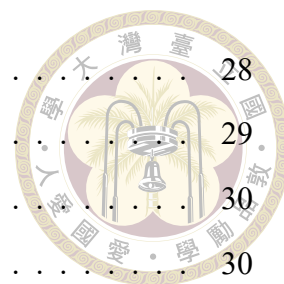
# 目錄

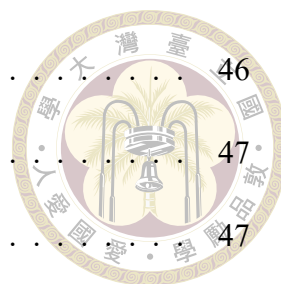
	Page
口試委員審定書	i
致謝	iii
摘要	v
Abstract	vii
目錄	ix
圖目錄	xiii
表目錄	xv
符號列表	xvii
第一章 緒論	1
1.1 研究背景 . . . . .	1
1.2 研究目的與目標 . . . . .	2
1.3 論文架構 . . . . .	3
第二章 文獻探討	5
2.1 身分系統的想像 . . . . .	5
2.2 身分系統的迭代 . . . . .	7
2.2.1 中心化身分 . . . . .	7
2.2.2 聯合身分 . . . . .	8



2.2.3	使用者中心的身分 . . . . .	9
2.2.4	自治身分 . . . . .	10
2.2.5	未來展望 . . . . .	11
2.3	身分系統的困境 . . . . .	11
2.3.1	使用者體驗 . . . . .	11
2.3.2	使用者認知 . . . . .	13
2.3.3	隱私保護 . . . . .	14
2.3.4	平等信任 . . . . .	15
2.4	身分系統的評估 . . . . .	16
2.5	總結 . . . . .	17
<b>第三章</b>	<b>系統設計</b>	<b>19</b>
3.1	系統核心機制 . . . . .	19
3.1.1	關鍵概念 . . . . .	19
3.1.1.1	道德標準的建立 . . . . .	19
3.1.1.2	自由的管理自己 . . . . .	21
3.1.2	技術機制 . . . . .	22
3.1.2.1	自主證照 . . . . .	22
3.1.2.2	數據證照 . . . . .	24
3.2	系統架構設計 . . . . .	25
3.2.1	系統結構概覽 . . . . .	26
3.2.2	層次與角色對應 . . . . .	26
3.2.2.1	共識層與共識核心 . . . . .	26
3.2.2.2	服務層與服務提供者 . . . . .	27
3.2.2.3	數據層與終端使用者 . . . . .	28

3.2.3	共識層	28
3.2.3.1	評價機制	29
3.2.3.2	區塊鏈的維運	30
3.2.3.3	跨服務的狀態共識	30
3.2.4	服務層	32
3.2.4.1	身分管理	32
3.2.4.2	證照管理	33
3.2.4.3	數據管理	34
3.2.5	數據層	35
3.2.5.1	身分管理	36
3.2.5.2	數據管理	36
3.2.5.3	證照管理	36
3.2.5.4	數據存儲	37
3.3	資安與隱私議題	38
3.3.1	身分識別問題	38
3.3.1.1	基於使用者時空的分析方法	39
3.3.1.2	基於危險程度的驗證機制	40
3.3.2	密碼救援問題	41
3.3.2.1	極致多因素認證	41
3.3.3	被遺忘權	42
3.3.4	積極數據授權	43
3.4	資料結構	43
3.4.1	共識核心	43
3.4.2	AID Server	44
3.4.3	Wallet	45
3.5	本章總結	45





3.5.1	創新突破	46
3.5.2	法規遵循	47
3.5.3	公認原則	47
<b>第四章</b>	<b>系統實作</b>	<b>49</b>
4.1	系統架構	49
4.2	實現細節	50
4.3	流程分析	52
4.3.1	產生新的 AID 與自主證照	52
4.3.2	進入支付服務獲取收據	53
4.3.3	使用 AI 服務對話	54
4.4	本章小結	55
<b>第五章</b>	<b>結論與未來展望</b>	<b>57</b>
5.1	研究結論	57
5.2	未來展望	58
5.3	總結	58
	<b>參考文獻</b>	<b>59</b>
	<b>附錄 A — 模型參數表</b>	<b>65</b>
A.1	模型一	65
A.2	模型二	65
	<b>附錄 B — BraTS 2021 分割結果圖</b>	<b>67</b>
B.1	編號 0001 0050	67
B.2	編號 0051 0100	67



## 圖目錄

2.1	中心化身分 . . . . .	7
2.2	聯合身分 . . . . .	8
2.3	使用者中心身分 . . . . .	9
2.4	自治身分 . . . . .	10
3.1	自主身分 . . . . .	21
3.2	自主證照流程 . . . . .	23
3.3	數據證照流程 . . . . .	25
3.4	自主身分系統結構圖 . . . . .	27
3.5	跨服務共識流程 . . . . .	31
4.1	AID 概念驗證架構簡圖 . . . . .	49
4.2	產生新的 AID 與自主證照 . . . . .	52
4.3	進入支付服務獲取收據 . . . . .	53
4.4	產生新的 AID 與自主證照 . . . . .	54





## 表目錄

3.1	自主身分系統層次與角色對應 . . . . .	26
3.2	各代身分管理系統比較 . . . . .	46
4.1	系統分層結構及介面 . . . . .	50







## 符號列表

$\mathfrak{a}$	$\mathfrak{a}$ 符號解釋
$\int$	$\int$ 符號解釋
$v$	$v$ 符號解釋





# 第一章 緒論

在當今數位時代，個人身分管理已成為一個日益重要且複雜的議題。隨著網絡技術的快速發展和社會對隱私保護的日益重視，傳統的身分管理系統面臨著前所未有的挑戰。本研究旨在探討一種創新的解決方案——自主身分（Autonomous Identity, AID）系統，該系統融合了新的思想與先進技術，為數位身分管理帶來根本性的變革。通過重新審視「自主」的概念並將其應用於身分管理領域，我們希望能夠為構建一個更自由、更公平的數位社會貢獻一份力量。

## 1.1 研究背景

自主（Autonomy）概念源於 18 世紀啟蒙運動，標誌著人類開始質疑對王權與神權的依賴，轉而追求通過科學與理性實現思想獨立與自由。啟蒙思想家伊曼努爾·康德（Immanuel Kant）對自主提出了開創性定義：「按照自己認同的道德標準自由行事」。這一定義至今仍廣泛被應用於政治、法律和教育等多個領域。接著，19 世紀的重要哲學家弗里德里希·尼采（Friedrich Nietzsche）對道德提出了更深入的探討。他對道德的本質進行了批判性分析，認為道德源於弱者對強者的反抗，是一種透過集結群體的共識來約束個體行為的機制。

這兩位哲學家的思想不僅塑造了現代社會對自主的理解，也為我們重新思考數位時代的身分管理提供了重要的理論基礎。隨著網絡技術的飛速發展，個人身

分資訊的管理與保護已成為當今社會的重要議題。在這一背景下，我們將自主的概念創新性地應用於數位身分管理領域。



本研究提出的「自主身分」(Autonomous Identity, AID)概念與目前主流探討的「自治身分」(Self-sovereign identity, SSI)系統有著本質的區別。SSI允許使用者參與身分管理系統的經營，賦予使用者對其數位身分的一定控制權。然而，AID則更進一步，使每位使用者在具備道德標準的身分系統中自由的管理自己。這種創新模式不僅賦予使用者更大的自主權，還在系統設計中納入了道德考量，以確保個人自主不會損害社會整體利益。

## 1.2 研究目的與目標

本研究旨在探討自主身分 (Autonomous Identity, AID) 作為解決當前身分管理問題的創新方案，並將「自主」的哲學思想融入數位身分管理的實踐中。具體而言，本研究將致力於實現以下目標：

1. **分析現有數位身分系統的局限性：**深入探討當前身分管理系統的主要挑戰和固有限制。
2. **構建 AID 系統的理論框架：**提出自主身分系統的核心理念，並闡述其如何解決現有系統的問題。
3. **設計 AID 系統的技術架構：**提出一種能夠實現使用者完全自主管理的技術方案，包括去中心化存儲、智能合約等關鍵技術的應用。
4. **評估 AID 系統的優勢與挑戰：**全面比較 AID 系統與傳統身分管理系統在多方面的差異，並分析 AID 系統在實際應用中可能面臨的挑戰。
5. **提出 AID 系統的未來展望：**探討 AID 系統在不同領域（如金融、醫療、政務等）的潛在應用場景，並提出相應的實施策略和路線圖。

最終，我們希望這項研究能夠推動數位身分管理領域的典範轉移，為構建更加安全、自由、便捷和公平的數位社會奠定基礎。



### 1.3 論文架構

為了基於自主的理念設計出一個完整的身分驗證系統，我們需要先了解現有的身分驗證技術，並且對於這些技術進行分析，找出其缺點（第二章）。接著提出我們的系統設計，並且說明其架構、資料結構與威脅模型（第三章）。最後，我們會透過實作來驗證我們的系統設計（第四章），最終我們會提出結論與對方案的未來展望（第五章）。





## 第二章 文獻探討

身分系統的發展經歷了從中心化到去中心化的演變，現代身分系統設計面臨著使用者體驗、認知負擔、隱私保護和平等信任等多方面的挑戰，需要同時滿足技術創新、法規遵循和行業最佳實踐，以創造出既能滿足多樣化需求又具備足夠靈活性的解決方案，本章將探討相關文獻以深入理解這些挑戰及可能的解決途徑。


### 2.1 身分系統的想像

網際網路（Internet）發源於 20 世紀 70 年代，逐漸發展成為現代網際網路的基礎架構。最初，Internet 的設計源自美國國防部的 ARPANET 計畫，旨在滿足軍事需求下的網路可用性和穩定性。因此，其初期設計基於以下假設 [29]：

- 最終使用者至少在最低程度上相互信任。
- 網路由於潛在的物理攻擊而本質上不可靠。

這些假設在當時的網路環境中是合理的。然而，隨著網路的普及和應用範圍的擴大，這些假設已不再適用 [9]。隨著 Internet 從一個研究驅動的项目演變為主流社會的重要組成部分，新的需求不斷湧現，不僅挑戰了原有的設計原則，還促使我們重新審視一些既有的假設。





在現代網路環境中，使用者之間的信任關係變得日益複雜。人們需要可靠的方式來識別自己和他人，以便在網路上進行交流和交易。因此，各種身分系統應運而生，以滿足網路環境中的身分識別需求。儘管人們希望找到一種統一而理想的身分管理系統，但正如 Cameron[7] 所指出的，這樣的系統實際上並不存在。身分系統涉及的範疇廣泛而複雜，個人與組織之間存在多樣化且往往相互衝突的需求，試圖通過單一標準來限制或規範這些需求是不切實際的。

例如，終端使用者可能希望自由地訪問和分享信息，而內容提供商和知識產權持有者則希望保護其知識產權。政府可能希望監管某些網絡活動以維護社會秩序，而使用者和隱私倡導者則強調個人隱私的重要性。這種複雜的利益衝突導致了諸如網絡中立性、數據隱私、內容審查等一系列熱點問題的出現 [41]。

面對這種複雜的網絡環境，Blumenthal[5] 認為確保設計上的一般性、彈性與開放性至關重要。他設想的未來充滿衝突：企業的管理者和被管理者爭論分紅，垃圾郵件的發送者和接收者爭論各自的困難。在這個網路世界中，不同身分的參與者沒有絕對的贏家，也沒有天生的失敗者。理想的系統應該通過所有使用者不斷的爭論和互動逐漸形成。

這種新的設計思路不僅需要考慮技術因素，還需要權衡經濟、社會、法律等多方面的因素。正如 Lessig[23] 所言，”程式碼就是法則”（”Code is Law”）。軟體設計本身就在表達某種價值觀，人們需要縝密的思考，結合對未來世界的想像，才能創造出足以改變網路環境既有困境的身分系統。

總的來說，現代網路環境的變遷對身分系統提出了新的挑戰，這些挑戰不僅來自技術層面，還涉及社會、文化和法律等多個方面。為了應對這些挑戰，我們需要重新思考身分系統的設計理念，尋找一種既能適應當前多樣化需求，又能为未來發展預留足夠靈活性的解決方案。

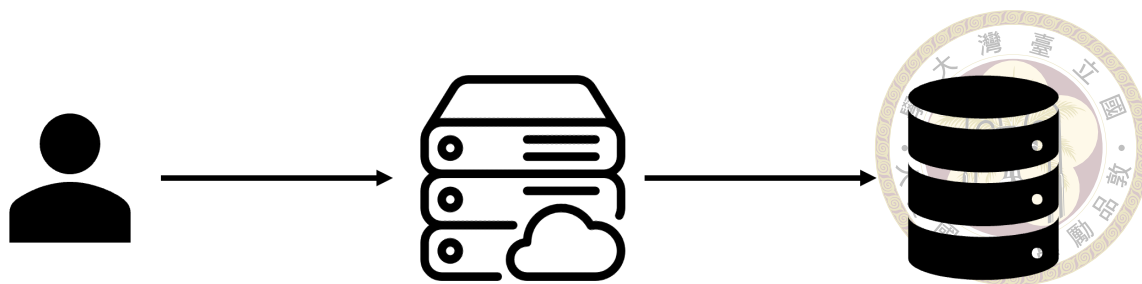


Figure 2.1: 中心化身分

## 2.2 身分系統的迭代

身分系統的設計經歷了多個階段的演變，每個階段都試圖解決特定的問題，同時也帶來了新的挑戰。本節將介紹不同世代的身分系統設計，以說明彼此衝突的需求和技術限制，並為後續討論提供背景。

### 2.2.1 中心化身分

中心化身分系統如圖2.1是最早期的身管理方案，由同一位管理者操作和存儲所有使用者資訊，在企業和政府機構中廣泛應用。典型例子包括活動目錄（Windows Active Directory, AD）和輕量級目錄訪問協議（Lightweight Directory Access Protocol, LDAP）[26, 36]。這類系統的主要優勢在於其集中管理的特性，便於系統管理員進行使用者管理和權限控制，同時確保組織內部身分信息的一致性和及時更新。然而，中心化身分系統也面臨著諸多挑戰，如單點故障風險、隱私保護問題，以及跨組織可遷移性差等。使用者通常需要為每個服務創建單獨帳戶，這不僅增加了認知負擔[19]，還導致使用者身分被服務提供商完全控制，缺乏自主權。

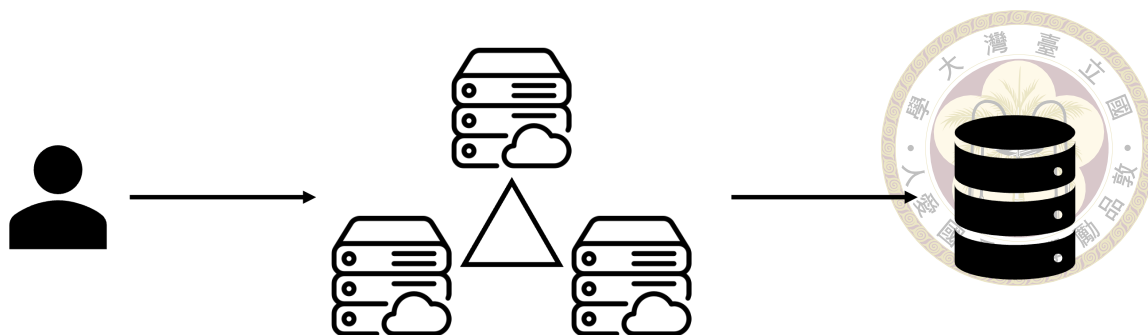


Figure 2.2: 聯合身分

### 2.2.2 聯合身分

主要為了解決同一個使用者擁有太多身分的認知負擔，聯合身分系統如圖2.2允許不同組織間共享身分信息，代表性技術包括安全斷言標記語言（Security Assertion Markup Language, SAML）和 WS-Federation [14, 31]。這種模式的出現大大改善了使用者體驗，實現了單點登錄（Single sign-on, SSO），減少了密碼疲勞問題。聯合身分系統促進了組織間的協作和資源共享，同時也降低了重複身分管理的運營成本。然而，這種模式也帶來了隱私方面的挑戰 [2]，如使用者信息在多個服務提供商間共享可能違反《通用資料保護規則》（General Data Protection Regulation, GDPR） [11] 等隱私法規。此外，實施和維護聯合身分系統的技術複雜度較高，參與組織之間需要建立並維護信任關係。



Figure 2.3: 使用者中心身分

### 2.2.3 使用者中心的身分

為了解決隱私方面的問題，使用者中心的身分系統如圖2.3所示逐漸興起。這種系統允許使用者透過單一的身分供應者登入多個獨立的服務，同時每個服務各自掌握使用者在其內部的資料。隨著這種需求的增長，新的技術標準應運而生。OpenID 和 OAuth 等協議的出現 [17, 34]，標誌著身分管理向使用者賦權的重要轉變。這種模式增強了使用者對個人身分信息的控制權，提供了更大的靈活性，允許使用者選擇不同的身分提供者。

使用者中心的身分系統實現了以服務供應商為單位的資訊範圍控制，在一定程度上改善了隱私保護。然而，這種方式也面臨著一些挑戰。首先是身分碎片化的問題，多個身分提供者的存在可能導致使用者體驗的不一致。其次，安全風險如釣魚攻擊和身分提供者數據洩露等仍然存在 [39]。此外，儘管使用者獲得了更多控制權，但他們仍然在某種程度上依賴中心化的身分提供者，因此使用者的自主性還擁有很大的提升空間 [4]。



Figure 2.4: 自治身分

## 2.2.4 自治身分

自治身分如圖2.4所示，是身分管理系統的最新發展 [32]。這種方式透過區塊鏈技術取代中心化的身分供應商，其代表性例子包括基於以太坊的 uPort 和微軟的自治身分覆蓋網絡 [24, 25]。自治身分系統賦予使用者對自身身分的控制權與對身分管理的治理權，同時提高了身分的可攜性和一致性。此外，它還增強了使用者與服務提供商之間的平等性，並提供了抗審查的特性。

然而，作為一種新興技術，自治身分系統也面臨著諸多挑戰 [35]：

- **法律框架衝突：**它可能與現有法律框架存在潛在衝突，例如與 GDPR 中的被遺忘權不相容 [12]。
- **技術複雜性：**自治身分系統的技術複雜性可能影響普通使用者的使用體驗，降低其易用性 [20]。
- **其他問題：**還有諸如隱私保護、系統互操作性等問題需要解決。



## 2.2.5 未來展望

綜上所述，身分系統的發展經歷了多個階段，每個階段都試圖解決特定的問題。從中心化到使用者自治，身分系統的設計逐漸向使用者賦權，提高了使用者對自身身分的控制權。然而，即使到了使用者自治階段，我們依舊不認為找到了理想的解決方案。如同 Schardong 等人 [35, 38] 所說，當今的自治身分系統仍面臨著許多挑戰，包括安全性、隱私保護、易用性、信任建立等問題。因此，我們認為身分系統的設計仍有很大的改進空間，需要更多的研究和實踐來不斷優化。

## 2.3 身分系統的困境

為了設計出一個理想的身分系統，在本節中，我們會從不同維度的多個方向來探討身分系統的困境。這些討論旨在幫助我們理解身分系統的核心特徵，並為未來的設計提供參考。

### 2.3.1 使用者體驗

使用者體驗在身分系統設計中扮演著關鍵角色，直接影響系統的可用性和採納率。然而，Hamme 等人 [16] 的研究闡明了使用者體驗、安全性和隱私保護之間的複雜關係。該研究指出了一個普遍存在的現象：使用者傾向於選擇最簡單的方式來設置和使用身分系統，這種傾向可能導致系統安全性和隱私保護程度的降低。

這種情況產生了一個兩難困境，為了提高安全性而強制使用者採用複雜的身分驗證方式可能會適得其反。例如 Zhang 等人 [43] 的研究表明，要求使用者定期更改密碼往往導致使用者僅修改特定字元，反而造成更大的安全隱患。同樣地，



為了增強隱私保護而要求使用者完成詳細的隱私設置也可能降低使用者體驗。Acquisti 等人 [1] 的研究發現，複雜的隱私設置過程往往讓使用者感到困惑和沮喪，甚至導致他們放棄設置而選擇默認選項，從而降低了隱私保護水平。

為解決這一困境，研究者提出了「無摩擦驗證」(Zero Friction Authentication) 的概念，旨在最小化使用者在設置和使用過程中遇到的困難，同時維持適當的安全性和隱私保護水平。Hamme 等人 [16] 強調，無摩擦驗證的核心目標是在保護使用者安全和隱私的同時，顯著降低使用者的操作負擔。這種平衡對於現代身分系統的設計至關重要，因為它直接影響系統的使用率和效能。

為實現無摩擦驗證，近年來安全領域出現了多種新技術。如 Ghorbani 等人 [13] 研究了無密碼登入（如 FIDO2）的可用性，發現這種方法能透過硬體密鑰在手機上跨裝置完成高安全性的驗證，且使用者普遍認為方便並願意持續使用。Wiefling 等人 [40] 探討了基於風險的驗證（Risk-Based Authentication, RBA），該方法透過追蹤身分與系統互動的歷史數據，在每次服務請求時動態判斷危險性，並在危險時採用更安全的多因素驗證（Multi-Factor Authentication, MFA）[6]。Alaca 等人 [3] 關於裝置指紋的研究顯示，透過在每次使用者對服務發出請求時記錄並比對裝置指紋，可以有效辨識部分惡意行為，且不會增加使用者的操作負擔。

另外，為解決複雜隱私設置帶來的問題，Acquisti 等人 [1] 提出了「隱私設計」(Privacy by Design) 的概念。這種方法將複雜的設置過程分解為多個簡單步驟，並在使用者使用系統的不同階段逐步引導使用者完成設置。研究表明，這種方法不僅能提高使用者的隱私保護水平，還能顯著改善使用者體驗。

這些新技術的應用表明，在不影響使用者體驗的前提下提供更高的安全性和更好的隱私保護是可能的。然而，如何在自主身分系統中實現真正的無摩擦驗證，以及如何有效地平衡使用者體驗、安全性和隱私保護的需求，仍然是一個值得深

入研究的課題。



### 2.3.2 使用者認知

使用者認知在數位身分管理中扮演著至關重要的角色，直接影響到系統的安全性和有效性。LastPass[21] 的研究揭示了使用者認知與實際情況之間存在顯著差距：使用者平均估計自己擁有 20 個線上帳號，而實際上平均擁有 37 個以上的帳號。這種認知偏差背後反映了使用者使用上的不便，例如經常因為忘記密碼而無法登入帳號，或者因為各個帳號資料不互通而需要耗費大量時間來管理。

Dhamija 等人 [10] 的研究進一步指出，使用者對身分管理系統的認知和理解程度直接影響其安全行為。隨著需要管理的使用者名和密碼數量增加，使用者往往感到困惑，進而採取不安全的行為，如使用弱密碼或在多個平台使用相同密碼等。此外，使用者對身分管理系統的認知不足也會導致他們無法有效應對釣魚攻擊、社交工程等安全威脅。

為了解決這個問題，未來的身分管理解決方案應該朝多個方向發展。首要任務是簡化多層次、多維度的使用者身分管理，允許單一的身分管理多樣的別名，以適用於不同的場景。例如，使用者可以用唯一的帳戶創建三個別名，分別對應自己的三種社會身分：在家中是家長，在工作中是員工，在社交場合是朋友。甚至針對單一的服務，使用者也可以擁有多種別名，如在論壇中既可以以專家身分發表權威言論，也可以作為普通使用者表達個人觀點。這樣的設計可以幫助使用者在盡可能不增加認知負擔的情況下有效管理自己的身分。

然而，真正簡化使用者認知並非易事。即使是宣稱已解決這個問題的使用者中心身分系統，實際上也未能完全做到。以 Google 的組織管理文件 [15] 為例，為了確保不同組織擁有不同的安全限制與規定，系統仍然要求使用者在不同組織間



創建不同的身分。這表明，同時簡化使用者認知與滿足組織需求，仍然是一個有待解決的挑戰。



### 2.3.3 隱私保護

在數位時代，隱私保護已成為身分系統設計的核心考量之一。歐盟制定的《通用數據保護條例》(GDPR) [11] 代表了目前全球最嚴格的隱私保護標準。本研究認為，一個理想的身分系統應當能夠全面符合 GDPR 的要求，從而確保使用者隱私得到最大程度的保護。然而，近年來的 GDPR 違規案例表明，即便是大型企業也面臨著遵守某些 GDPR 規定的挑戰。

基於 Schardong 等人的研究 [35]，我們發現當前身分系統中存在兩個尤為突出的關鍵問題。首先是使用者積極授權的實現困難。Saemann[33] 的研究強調，在當前的身分系統框架下，企業難以實現使用者對數據使用的明確授權。具體而言，企業難以證明其對數據或權限的使用行為已獲得使用者授權，而使用者也缺乏有效途徑證明自己的數據或權限被不當使用。這種情況不僅增加了企業的法律風險，也削弱了使用者對系統的信任。

第二個問題是被遺忘權的實現困難。Smirnova[37] 指出，滿足使用者的被遺忘權在當前身分系統中存在著一定的挑戰。使用者數據在系統中往往呈分散狀態，即使刪除核心使用者的資料，仍可能保留使用者的系統日誌或與其他使用者的互動數據。這種情況使得完全實現被遺忘權變得複雜而困難，可能導致使用者隱私無法得到全面保護。

基於以上分析，我們提出符合現代隱私保護要求的身分系統應該具備兩個關鍵特點。首先，系統應提供合理的機制，讓使用者或企業能夠證明其數據使用行為是否符合授權，這將有助於提高系統的透明度和可信度。其次，系統應提供有

效的方法讓使用者行使被遺忘權，確保使用者不會被難以刪除的數據綁架。這意味著系統需要設計更精細的數據管理和刪除機制。在後續研究中，我們將詳細探討如何在自主身分系統中實現這些特性，並提出相應的技術解決方案。



#### 2.3.4 平等信任

身分系統中的平等信任問題是一個複雜的多方利益平衡問題，涉及系統的公平性和可信度。研究 [32] 強調了身分系統中各方利益的衝突，主要表現在使用者之間的權益差異、不同系統間的互操作性問題，以及使用者與系統供應者之間的利益衝突。例如，身分系統供應者可能希望獲取更多使用者個人資料以獲取利益，而使用者則希望保護自己的隱私。這種利益衝突如果處理不當，可能導致系統環境惡化、使用者權益受到侵犯，以及市場壟斷和不公平競爭。Zuboff[43] 的研究進一步指出，這種數據收集和利用的不平等可能導致所謂的「監視資本主義」，對個人自由和社會公平造成深遠影響。

在當前的身分系統中，建立健全的信任模型仍然是一個重大挑戰。傳統的二元邏輯驗證模式（即完全信任或完全不信任）已不能滿足現代身分系統的需求。研究 [35] 指出，現實世界的信任往往是模糊而不確定的，人們很難用簡單的真假邏輯分辨使用者驗證的成功與否。例如，可能存在多組憑證同時存在，部分驗證成功，部分驗證失敗的情況，或者不同照照的可信度和重要性各不相同。Josang 等人 [18] 提出的主觀邏輯數學框架為處理這類多組不確定性證照的問題提供了一個系統性的解決方案。該框架將憑證的可信度表示為一個區間，從而更好地模擬了現實世界的身分驗證過程。

此外，在去中心化系統中，解決身分驗證問題尤其困難。正如 Dhamija[10] 所強調的，使用者需要向系統證明自己的身分，同時系統也需要向使用者證明自己

的合法性和可信度。Tze-Nan[42] 提出的自主證照機制為解決這一問題提供了一個新的思路。他改良了傳統的證照簽署（Certificate Authority）技術，使證照不僅可以由使用者自主操作，還能被所有經手者評分。這樣一來，使用者和系統之間就可以在自主的前提下互相驗證並評分，從而建立起一個平等的信任關係。

然而，長期來看，一個合理的評分機制，甚至是一個長期的治理機制也是必要的。在這方面，Chohan 等人 [8] 關於 DAO（去中心化自治組織，Decentralized Autonomous Organization）治理的研究提供了核心概念，可以為自主身分系統的制度建設提供參考。

在後續的研究中，我們將探討如何在自主身分系統中實現技術上的互相驗證與制度上的互相信任，最終構建一個能夠平衡各方利益、促進公平競爭而不易壟斷的自主身分生態系統。這種系統將能夠在保護使用者權益的同時，也為系統供應者提供合理的發展空間，從而實現真正的平等信任。

## 2.4 身分系統的評估

鑒於當前身分系統在各個方面面臨的特定挑戰，我們提出以下評估標準，用以衡量自主身分系統的設計是否符合下一代身分系統的要求：

- **創新突破：**自主身分系統的設計應解決上節所述困境，才能實現真正的技術創新。以下陳列：
  - 使用者體驗：提供無摩擦驗證，簡化操作流程，提高系統易用性。
  - 使用者認知：提供簡單的身分管理機制，幫助使用者有效管理身分，減少認知負擔。
  - 隱私保護：實施有效的隱私保護機制，保護使用者個人數據，確保隱私得到充分保障。



- 平等信任：建立互相驗證和互相信任的機制，促進所有個體間平等的信任關係。
- 法規遵循：自主身分系統應符合相關法律法規，特別是在數據保護和隱私保護方面。例如：
  - GDPR：遵守一般資料保護規範 [11] 的要求，包括使用者同意、數據可控性和被遺忘權等規定。
  - NIST：符合美國國家標準 [28]，涵蓋多因素驗證、風險評估、身分驗證和授權等要求。
- 公認原則：遵循身分識別領域的公認原則，包括但不限於：
  - 身分法則：符合 Kim Cameron 提出的身分管理基本法則 [7]，包括使用者控制、最小化披露、合法性和互操作性等原則。
  - 避免常見缺陷：克服 Dhamija 等人 [10] 指出的身分管理七大缺陷，如密碼管理、安全提示和密碼重置等問題。
  - 自治身分：符合 Allen[4] 所說，自治身分應該遵循的 10 個原則。

這些評估標準綜合考慮了技術創新、法律合規性和業界最佳實踐，為評估和設計下一代自主身分系統提供了全面的合理的目標。

## 2.5 總結

本章從身分系統的迭代、設計原則和技術評估三個方面探討了現代身分系統的設計問題。我們發現，現代身分系統的設計仍面臨著諸多挑戰，包括使用者體驗、使用者認知、隱私保護、平等信任等方面。為了解決這些問題，我們提出了一系列設計原則，包括無摩擦驗證、隱私設計、使用者認知簡化、..... 等。在未來的研究中，我們將進一步探討如何在自主身分系統中達到這些評估標準，並提

出相應的技術解決方案。





## 第三章 系統設計

自主身分 (AID) 系統是一種創新的身分管理方案，其核心理念在於讓每位使用者在具備道德標準的身分系統中**自由的管理自己**，從而解決現有身分管理系統的問題。本章節將深入探討如何通過核心機制的設計將這一理念融入系統中，同時闡述系統的完整架構設計、資安與隱私議題和資料結構，為 AID 系統的實際應用奠定理論基礎。

### 3.1 系統核心機制

為了實現 AID 系統的核心理念，我們需要深入理解**道德標準與自由的管理自己**這兩個關鍵概念，並將其轉化為具體的技術機制。

#### 3.1.1 關鍵概念

以下將分別探討道德標準和自由的管理自己這兩個關鍵概念。

##### 3.1.1.1 道德標準的建立

就像 Nietzsche 所說的道德是弱者對強者的反抗，我們認為道德標準的建立並不是簡單的制定某些規範，而是創建出一種機制能讓多數普通使用者能夠集結力

量，形成共識，從而有效地制衡可能濫用權力的少數強勢參與者。因此，AID 系統引入了一種基於區塊鏈的去中心化評價機制。在這個機制中：



- 每個系統參與者（包括使用者和服務提供商）都有權對其他參與者進行評價。
- 這些評價被記錄在區塊鏈上，確保其不可篡改和公開透明。
- 系統參與者可以按照自己的需求分析其他參與者的評價，從而形成對其道德標準的判斷。

通過這種機制，AID 系統實現了一種基於共識的道德標準，使得每位使用者都能夠參與到對系統的建設和監督中，從而確保系統道德標準的建立。



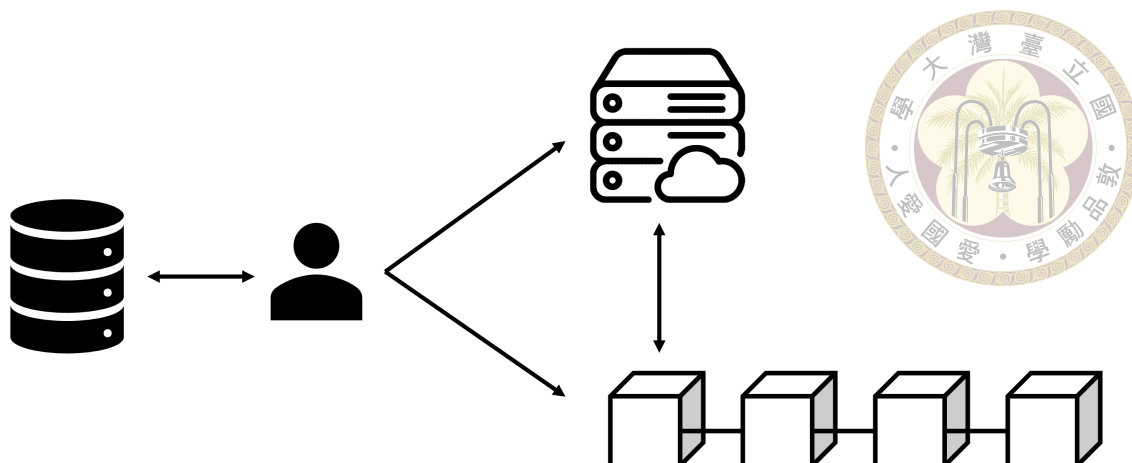


Figure 3.1: 自主身分

### 3.1.1.2 自由的管理自己

為了實現使用者對自己身分的自主管理，我們需要深入了解自主身分系統中可管理的元素。在 AID 系統架構中，使用者可管理的內容包括兩個方面：

1. **數據管理**：指使用者在系統交互過程中產生的全部數據。
2. **功能管理**：指只能由使用者自己決定要行使哪些功能。

實現數據的自由管理是一項具有挑戰性的任務。在傳統身分管理系統中，使用者數據通常由身分供應商（Identity Provider）或服務提供商（Service Provider）集中控制，這嚴重限制了使用者在個人數據上的自由。為了克服這一限制，AID 系統採用了創新的「數據層反轉」策略如圖3.1：將使用者數據完全遷移至使用者端設備，從而實現使用者對個人數據的直接控制。

與此同時，在功能管理方面，AID 系統摒棄了管理者或任何形式的中心化控制方法。這樣的設計進一步增強了使用者管理自身身分的自由度，再次彰顯了自主身分的核心理念。





### 3.1.2 技術機制

基於上述關鍵概念，我們要強調幾項與之相關的重要技術機制。

#### 3.1.2.1 自主證照

為了實現使用者能夠自由加入服務而無需在服務中存儲個人數據，同時允許其他系統參與者對使用者進行評價，我們參考了 Tze-Nan[42] 提出的自主證照概念，設計了一個基於區塊鏈技術的自主身分管理流程。

自主證照是由使用者自行生成的一種證照，使用者可以根據個人需求自定義證照的內容和權限，並且擁有隨時撤銷證照的權力。這種機制讓使用者能夠靈活地管理自己的身分信息，同時確保了系統的安全性和可靠性。本研究提出的身分管理流程包含以下關鍵步驟如圖3.2：

1. **生成 AID**：透過 UUID[22] 機制產生唯一編號，使用者可以根據需求設置 AID 內含的詮釋資料 (metadata)。
2. **產生證照**：使用者可以傳送相同的 AID 到不同的簽章者處建立不同目的的證照，AID 內的 metadata 可以視情況調整。
3. **證照上鏈**：AID 的雜湊 (hash) 作為證照，會被簽章者記錄在區塊鏈上。
4. **自主加入**：使用者在加入服務時，主動向服務提供者提交自己的 AID，表明自己的身分。
5. **區塊鏈驗證**：服務提供者可以在區塊鏈上取得證照得知 AID 的真實性。
6. **彈性驗證**：使用者在登入服務時，可以根據 AID 自主選擇多因素驗證 (MFA) 的方式，來取得服務的信任。
7. **使用者評價**：區塊鏈上的證照是 AID 的雜湊被放在智能合約中，相關參與者可以對證照進行評價，從而形成該使用者的信譽。

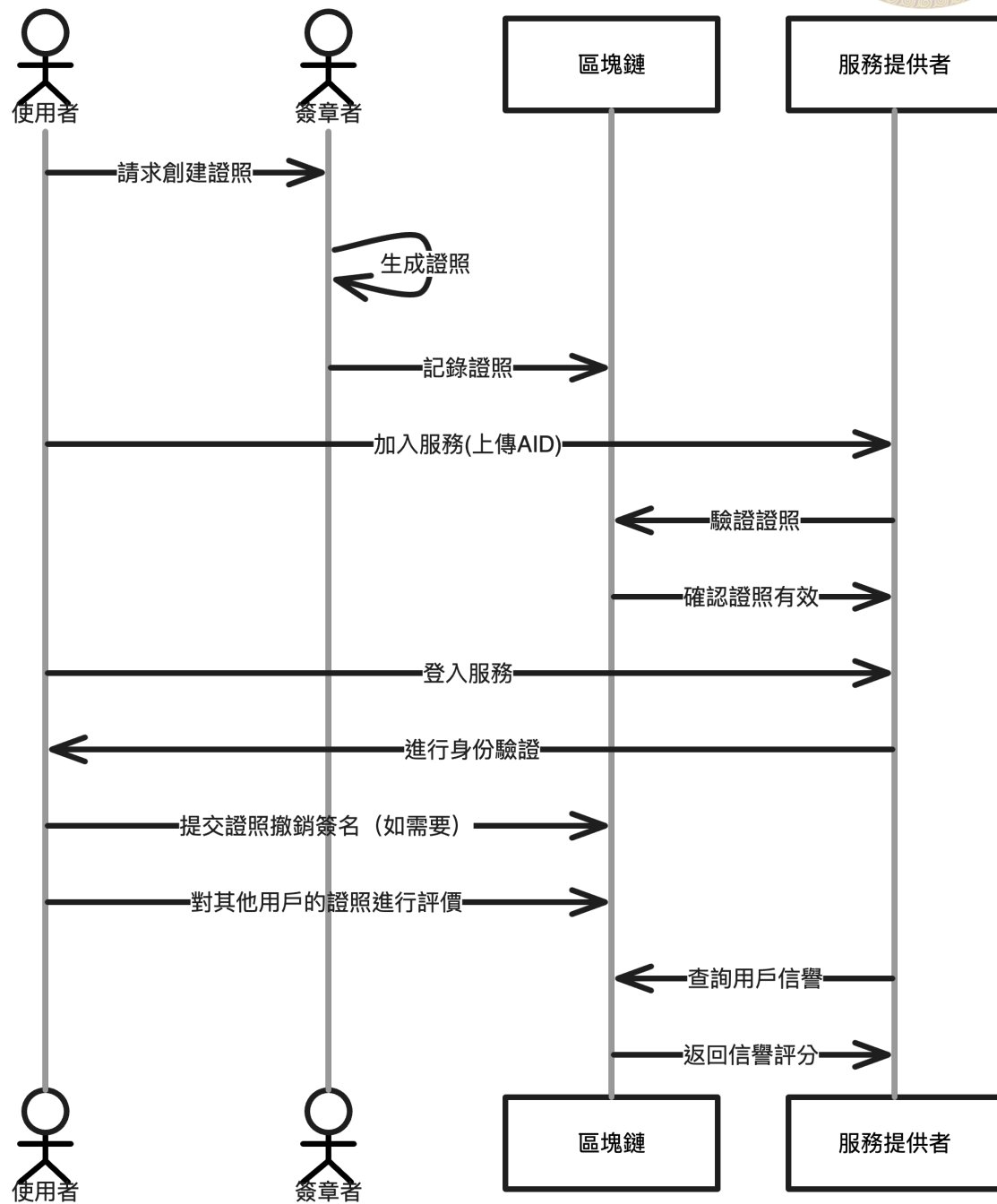


Figure 3.2: 自主證照流程

8. **撤銷證照**：使用者可以隨時撤銷證照，並在區塊鏈上自行操作。這種機制可以應用於 AID 外流等情況。



此外，本系統的 AID 除了必須的唯一識別號，支持多樣化的 metadata，以下舉例說明：

- 自定義多因素認證選項：滿足使用者登入的驗證所需。
- 選擇性資訊揭露：依照服務場景，使用者可以自行選擇要揭露的資訊。
- 設置證照有效期限：使用者可以設定證照的有效期限，以保護自己的隱私。
- 指定特定的驗證條件 (如特定設備，地點或時間)：增強使用者對證照的控制。
- 指定特定的驗證規則 (設備和網路使用限制)：增強使用者對證照的控制。
- 資源存取權限控制 (如檔案存取權限)：增強使用者對證照的控制。

總結來說，通過這種機制提升了系統對使用者隱私和安全的保護程度，讓使用者能夠更自主地管理其身分驗證流程，保持對個人身分驗證的控制權。

### 3.1.2.2 數據證照

自主身分系統中，使用者在個人設備上自行管理數據的模式引發了數據共識問題。與傳統身分系統中服務可直接調用其他服務獲取數據不同，自主身分系統在處理跨服務數據共享需求時，需要一種機制來確保數據的一致性和可信度。為此，我們擴展了自主證照的概念，提出了數據證照機制。數據證照機制的運作方式如圖3.3：

1. 當使用者需要向特定服務提交某個數據時，首先向能夠證明該數據真實性的其他服務提供者申請數據證照。
2. 接受申請的服務提供者在區塊鏈上提交該使用者數據的校驗雜湊。

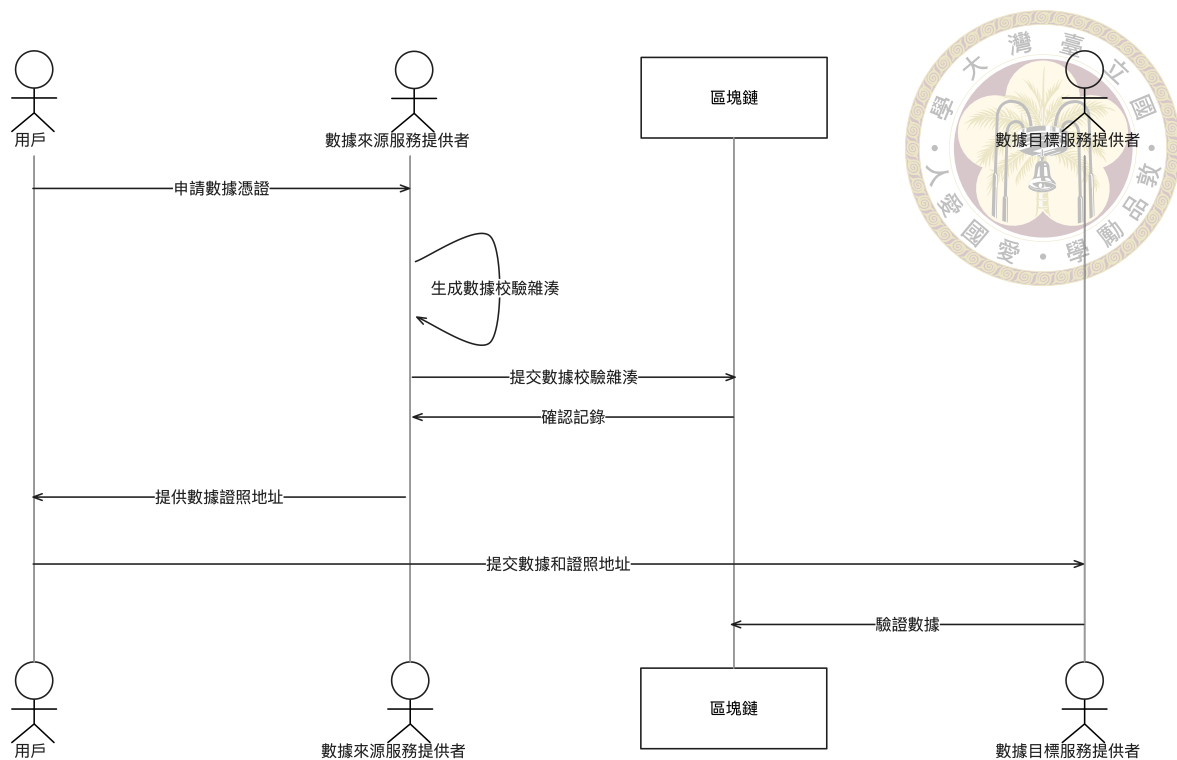


Figure 3.3: 數據證照流程

3. 使用者在向特定服務提交數據。
4. 接收數據的服務通過區塊鏈上的證照完成數據的校驗，從而確認數據的真實性。

這種設計讓使用者能夠在不同的自主身分服務中安全地共享數據，同時向其他使用者保證數據的一致性和可信度。

此外，數據證照的雜湊被放置在智能合約中，允許相關使用者對該筆數據進行評價。這一做法將道德標準的概念從身分層面擴展到數據層面，進一步提高了數據的可信度和安全性。

## 3.2 系統架構設計

基於上述核心機制，我們提出了一個完整的自主身分系統架構。這個架構融合了系統的技術層次和參與者角色，形成了一個統一且高效的生態系統。本節將

詳細介紹這個架構的結構和設計理念。



### 3.2.1 系統結構概覽

層次	角色	主要功能
共識層	共識核心	提供可信賴的數據讀寫機制
服務層	服務提供者	提供特定服務, 不直接儲存使用者資料
數據層	終端使用者	管理個人數據, 使用服務

Table 3.1: 自主身分系統層次與角色對應

我們的自主身分系統架構包含三個層次，由上到下分別是共識層、服務層、數據層，每個層次對應一種關鍵角色分別是共識核心、服務提供者、終端使用者。這種對應既反映了系統的技術架構，也體現了各參與者在系統中的功能和職責。圖3.4展示了這種層次-角色對應關係，表3.1則展示了各自的存在目的。

### 3.2.2 層次與角色對應

自主身分系統是個龐大的系統，每個層次內可以包含多個可以平行擴展的角色實作，因此為了確保各個層次能得到模組化、易抽換、可擴展、靈活、安全和互操作等優勢，我們明確訂出了各層的功能和職責。

#### 3.2.2.1 共識層與共識核心

共識層在自主身分系統中扮演著基礎設施的角色，其主要參與者是共識核心。共識核心通過提供可信賴的數據讀寫機制，確保了「數據證照」與「自主證照」等機制在達成共識方面的可能性。其設計目的是確保數據的一致性和可信度，為整個系統提供堅實的信任基礎。共識層的實現可以靈活選擇，既可以採用區塊鏈技術，也可以使用其他形式的共識機制，以滿足不同場景的需求。

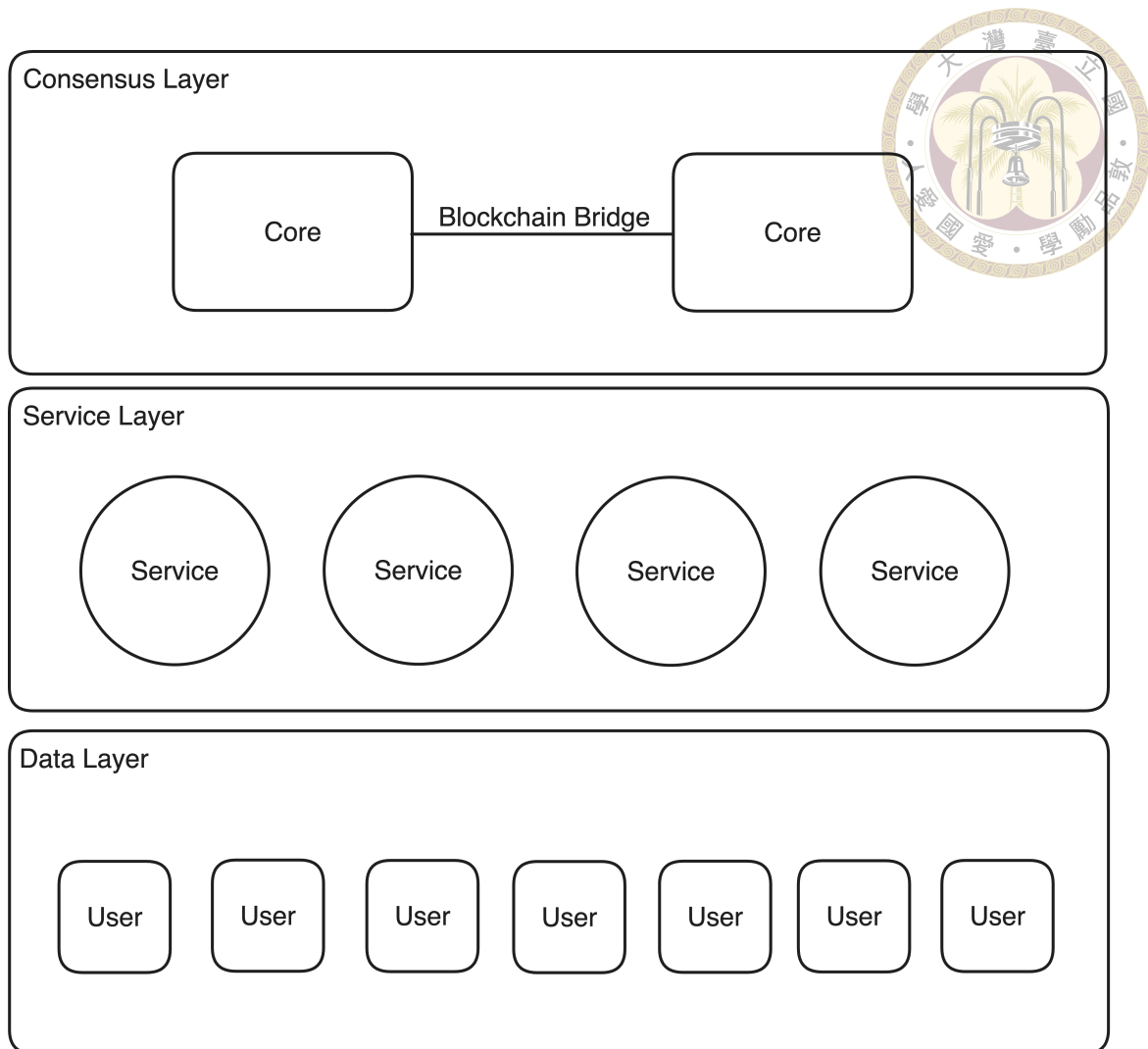


Figure 3.4: 自主身分系統結構圖

### 3.2.2.2 服務層與服務提供者

服務層是系統的中間層，對應的參與者是各個服務提供者。每個服務提供者都提供特定的服務，並且可以通過符合 AID 系統標準的介面加入系統。服務提供者不直接儲存使用者資料，而是作為使用者聚集的節點，使用者能夠自由地利用 AID 系統使用所需的服務。這種設計既保護了使用者數據隱私，又提供了靈活的服務串接機制。



### 3.2.2.3 數據層與終端使用者

數據層是系統的應用層，直接面向終端使用者。在這一層中，每個終端使用者擁有自己的數據存儲空間，並且系統提供了一個統一的數據管理介面。使用者可以自由地管理自己的數據與身分，並且通過「數據證照」與「自主證照」機制來獲得他人對個人身分或數據的信任。

### 3.2.3 共識層

共識層是自主身分系統的基礎建設，主要由無數個共識核心組成。共識核心內包含無數個證照，不管是「數據證照」還是「自主證照」。當共識核心由區塊鏈實作時，會使用智能合約來實現證照的寫入、讀取、更新和評價等功能，分別介紹：

- **寫入**：用於合約初始化，將數據對應的雜湊與公開資訊寫入合約狀態。
- **讀取**：讀取合約狀態中的雜湊或公開資訊，用於驗證或查詢。
- **更新**：由證照擁有者執行，用於即時變更證照狀態，如撤銷或停用。
- **評價**：允許使用者根據智能合約規則，以特定格式讀取或寫入評價內容。

为了更好地理解共識層的運作，讓我們以學歷驗證為例。在這個場景中，使用者向學校申請學歷證明，學校將證明的數位雜湊寫入區塊鏈。任何需要驗證該學歷的服務都可以通過讀取區塊鏈上的雜湊來確認其真實性。若使用者的學歷狀態發生變化，學校可以即時更新區塊鏈上的雜湊狀態，確保驗證方獲得最新資訊。此外，如果某企業對學歷證明的可信度存疑，可以在區塊鏈上留下評價，供其他驗證方參考。

然而，這樣的設計也面臨著諸多挑戰。首要問題是如何有效評價鏈上雜湊，



並將其轉化為影響使用者信譽的機制。其次，作為系統信任核心的區塊鏈，其長期穩定運營的可行性也是一個極需解決的問題。最後，在跨服務的使用者數據變化過程中，如何建立有效的共識機制同樣是一個關鍵議題。這些新架構下衍生的問題都需要深入探討和研究，以確保系統的可靠性和可持續性。

### 3.2.3.1 評價機制

為了解決評價問題，我們把整個自主身分系統視為一個去中心化自治組織（DAO）。基於自主身分的核心理念，我們希望每個系統參與者（包含服務供應者和終端使用者）都擁有自己的標準來基於評價判斷信譽，同時每個參與者也可以決定自己應該如何被評價。為了實現這一目標，我們提出了完整的流程來描述評價與信譽轉換：

1. **產生證照：**系統參與者創建「數據證照」或「自主證照」後，可以把對應雜湊放在自己決定的智能合約中上鏈，這個智能合約可以自訂許多規則，如是否允許別人對證照進行評價，又或是評價的格式或條件等。
2. **產生評價：**取得明文證照的人可以在區塊鏈上找到對應的智能合約，根據合約的規則對證照進行評價。評價的內容必須遵循智能合約的規則，否則無法上鏈。
3. **信譽轉換：**用有明文證照的人找到對應智能合約後可以讀取評價列表，之後根據自己認同的計算標準或演算法將評價轉換為信譽值。這個信譽值可以用於後續的信任判斷。

進一步來說，一些使用者可能希望根據評價者的信譽值來調整每條評價的權重，以更有效地防範惡意評價。然而，設計這樣的系統並非易事。它可能需要引入評價者作為系統中的新角色，並考慮如何綜合多個評價者的評分結果[18]。此外，



還需要考慮如何通過智能合約來實現這些複雜的功能。



### 3.2.3.2 區塊鏈的維運

作為系統信任核心的區塊鏈如何運營，這涉及到區塊鏈的經濟模型。我們提議在區塊鏈中發行一種逐漸增加的自主身分統治代幣。這種代幣可被抵押，用於創建使用者的自主身分，藉此防止惡意使用者大規模創建惡意帳戶。每個自主身分（相當於其背後的代幣）可參與定期投票，討論評價機制的調整、新的智能合約協議等議題。為確保區塊鏈的長期運作，我們建議對失去信任的使用者實施懲罰，同時獎勵基礎設施的運作。因此，用於創建自主身分的代幣抵押後不可贖回，而是被鎖定在區塊鏈上。這樣可以確保使用者不會輕率創建自主身分，並激勵使用者維護自身自主身分的信譽。此外，我們建議對使用者在鏈上的每項操作採取使用者付費模式，使區塊鏈的維護者能獲得報酬，從而保證區塊鏈的持續運作。

### 3.2.3.3 跨服務的狀態共識

最後，我們需要設計一個機制來實現跨服務的使用者數據變化共識。這是一個常見的網路服務需求，如微服務間的調用即為典型案例。在我們的設計中，資料的控制權從服務提供者轉移到使用者身上，這將大幅改變當前網路後端應用的設計。基於共識層的功能，我們提議讓使用者成為多個服務之間的橋樑。具體而言，使用者在區塊鏈外傳遞特定格式的資訊，而服務提供者則在區塊鏈上驗證這些資訊，從而達成共識。舉例來說，實作一個使用第三方支付服務的購票系統時，我們的方法與現行的網路後端設計有所不同。現有設計中，購票系統全權負責與支付體系的串接，使用者僅需與購票系統溝通。而在我們提出的架構中如圖3.5，流程如下：

1. 購票系統要求使用者到銀行服務完成支付並取得收據。

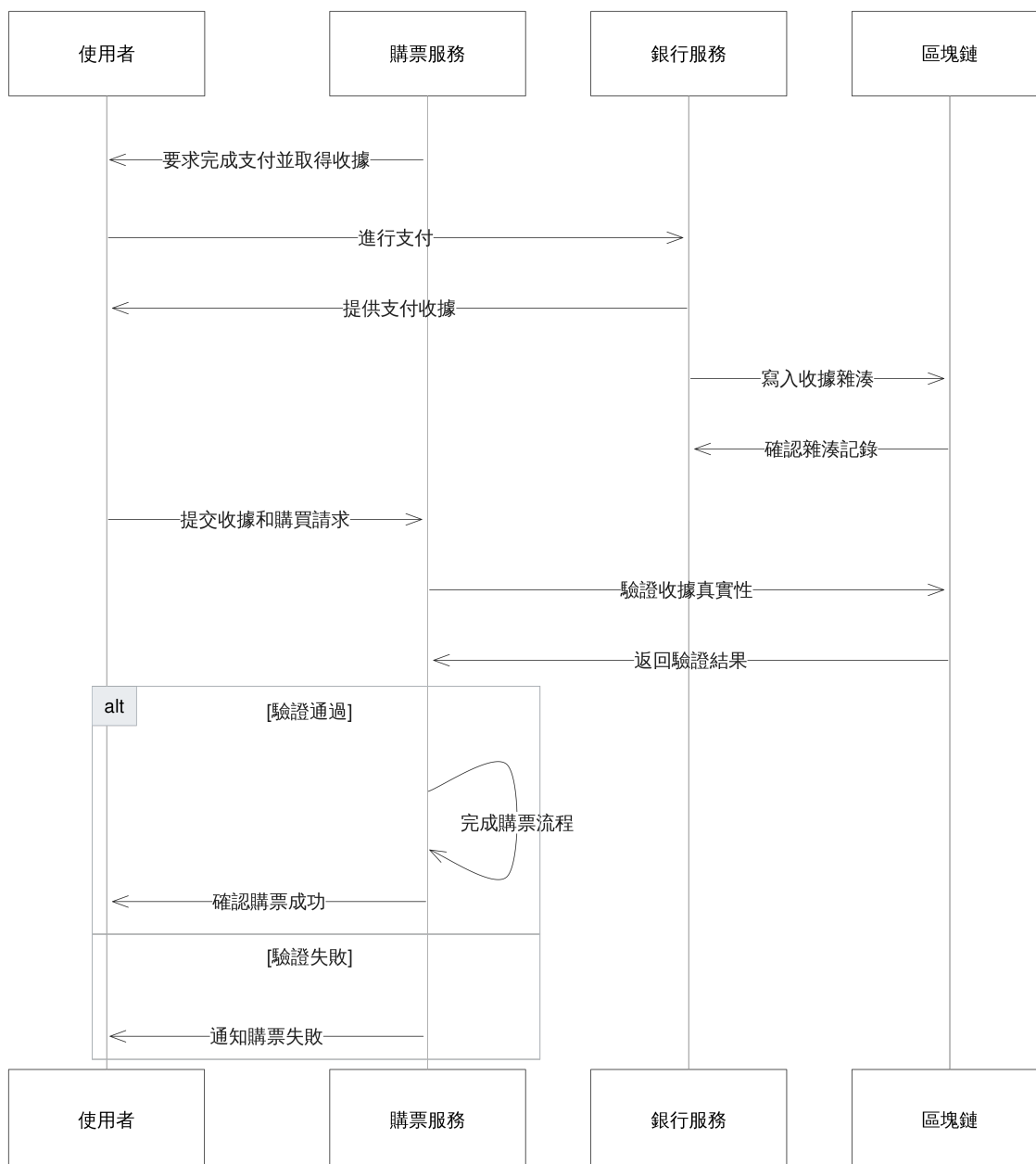


Figure 3.5: 跨服務共識流程



2. 銀行服務將收據的雜湊寫入區塊鏈。
3. 使用者向購票系統提交收據和購買請求。
4. 購票系統在區塊鏈上驗證收據的真實性。
5. 驗證通過後，完成購票流程。

這種方式不僅確保了數據的可信度，還賦予了使用者更多對自身數據的控制權，體現了自主身分系統的核心理念。

### 3.2.4 服務層


服務層是自主身分系統的應用層，負責提供各種服務。自主身分系統不包含具體服務的實作，只是提供名為 AID Server 的後端 SDK 規格，讓服務開發者能夠輕鬆地把自己的應用串接自主身分系統。我們設計了 AID Server 的幾個關鍵功能：

- **身分管理**：提供服務加入、登入、登出等基本身分管理功能。
- **證照管理**：提供證照的創建、更新、讀取、評價等功能。
- **數據管理**：提供使用者數據的導入、導出等功能。

以下將分別介紹這幾個功能的設計細節。

#### 3.2.4.1 身分管理

對服務提供者而言，新的自主身分加入必須先由使用者上傳「自主證照」。此時，服務提供者可選擇是否接受該新自主身分。多數公開服務可能接受任何自主身分，但私人服務或有更高要求，如僅接受特定機構簽章的自主證照，或基於評價機制被充分信任的自主身分。一旦服務提供者接受新的自主身分，該身分即可開始使用服務。



為了在確保安全的同時提供便利，自主身分系統提供了兩種登入方式：簡易登入和多因素驗證登入。簡易登入是指使用者僅需提供少量資訊即可登入，如使用者別名和密碼。多因素驗證登入則要求使用者提供更多資訊，如電子郵件驗證碼、手機簡訊驗證碼或私鑰等。服務提供者可根據自身需求選擇是否啟用簡易登入功能。

另外，儘管自主身分（AID）的生成是透過 UUID 機制 [22] 產生唯一識別號，但 AID 系統允許使用者在服務中設定自己偏好的別名，而非如傳統身分服務限制使用唯一的電子郵件地址作為使用者名稱。在日常操作中，系統優先讓使用者使用別名作為帳號完成簡易登入，只有當別名難以被識別時，才會要求使用者使用與 UUID 關聯的多因素驗證登入機制進行識別。這樣的設計我們認為更能表現出使用者對自己身分的自由控制權。

最後，服務提供者應當實作完善的登出功能，讓使用者能夠有效管理自身的在線狀態。這一功能不僅允許使用者標示自己的當前狀態，更應包含請求服務端遺忘所有數據和終止會話（session）等選項。因為在自主身分中，使用者擁有自己的數據，服務提供者僅提供服務，因此每個使用週期結束後，服務提供者應當刪除所有與該使用者相關的數據。通過提供這種全面的登出機制，服務提供者能夠進一步強化使用者對其身分資訊的掌控，同時滿足使用者在資訊安全和隱私保護方面的需求。

#### 3.2.4.2 證照管理

服務層中的證照管理包含兩個部分：分別是「自主證照」和「數據證照」。在「自主證照」方面，大多數都是由使用者直接上傳，服務可以據此找到區塊鏈上使用者身分證照的狀態與雜湊來完成身分驗證。此外，其他實體（如服務提供者或

其他使用者)可基於自身經歷，在使用者證照對應的鏈上合約中留下評價。

在「數據證照」方面，當使用者申請共享特定數據時，服務提供者可按照預定的智能合約協議創建證照，並將證照發布到區塊鏈上。如果數據發生變動，服務提供者可即時更新證照狀態，確保使用者數據的即時性和真實性。當另一個服務收到使用者在區塊鏈外提交的完整數據後，除了可以通過區塊鏈上的證照驗證數據的真實性外，還可以在區塊鏈上留下評價，為其餘服務提供者提供參考。

### 3.2.4.3 數據管理

因為自主身分系統的核心在於賦予使用者控制自身數據的權利，數據管理在服務層相對簡單。理論上，服務提供者的最低要求是確保使用者能夠：

1. 將當次操作所必需的數據導入服務中
2. 在操作結束後將數據導回使用者端

然而，這種簡單設計可能大幅降低使用者體驗。例如：

- 若使用者每次操作都需要導入和導出數據，許多優化使用者體驗的功能將變得不可行，因系統無法追蹤使用者歷史數據。
- 頻繁的大量數據導入導出會顯著增加操作延遲和網路成本。

因此，「混合數據管理」模式成為必然選擇，即部分數據由使用者保管，部分由服務提供者保管。這種設計在保護使用者隱私的同時，也確保了操作的便利性。然而，「混合數據管理」模式仍面臨諸多需要服務提供者與使用者達成共識的問題，包括：

- 哪些數據勢必要導入服務端才能完成操作？
- 哪些數據可以持續保留在使用者端？



- 服務提供者管理的數據保留時間？
- 數據的使用權限？

這些問題涉及對隱私和資安的權衡。為確保使用者自主權，我們建議：

1. 服務層在索取數據時應提供清晰的說明，讓使用者瞭解資料上傳的必要性。
2. 採用逐步詢問的方式協助使用者設定所有簡化的資安措施或降低的隱私保護措施。
3. 設置應從高到低、從嚴格到寬鬆平滑過渡，使使用者能根據自身需求調整安全性和隱私保護程度。

這種方法能在保護使用者權益的同時，為服務提供者提供必要的數據支持，實現雙方利益的平衡。透過這種「混合數據管理」模式，我們可以在自主身分系統中實現數據的高效管理，同時保障使用者的數據主權。

### 3.2.5 數據層

數據層是自主身分系統的存儲層，負責存儲使用者個人數據。我們設計了名為 Wallet 的前端 SDK 規格，讓數據層應用開發者能夠輕鬆地把自己的應用接入自主身分系統。Wallet 的幾個關鍵功能如下：

- **身管理**：提供自主身分的創建功能。
- **數據管理**：提供使用者數據的上傳、下載等功能。
- **證照管理**：對共識雜湊的更新、讀取、評價等功能。
- **數據存儲**：儲存使用者的數據。

以下將分別介紹這幾個功能的設計細節。





### 3.2.5.1 身分管理

Wallet 的身分管理主要提供一個核心功能：創建自主身分。這裡的創建並非指使用者加入某個服務，而是包含兩個主要功能：

1. 使用者直接在本地裝置透過隨機產生的 UUID 創建自己的自主身分
2. 透過「自主證照」機制創建新的證照

在我們的設計中，使用者可依據需求使用個人裝置創建不同的 AID，並為不同需求生成不同的身分證照，再利用這些證照證明 AID 的真實性。當使用者需要向服務證明自己是 AID 的真實持有者時，僅需透過 Wallet 完成 AID 上標示的多因素驗證方案。

### 3.2.5.2 數據管理

Wallet 的數據管理功能主要在服務需要時，將特定數據自主上傳至服務中，並在服務結束後自主下載回本地裝置。這裡的「自主」指使用者可自由選擇是否上傳或下載數據，並可自由選擇上傳或下載的數據內容。這樣的設計不僅更徹底地保護了使用者的安全與隱私，更因為由使用者自行攜帶數據在服務間移動，而徹底解決了數據孤島的問題。

### 3.2.5.3 證照管理

Wallet 的證照管理功能主要對共識層內證照進行更新、讀取、評價等操作。基於「數據證照」與「自主證照」機制，使用者與服務需要在共識層中的區塊鏈上基於智慧合約留下 AID 雜湊，讓人們可在區塊鏈上驗證 AID 的真實性。此外，所有使用者（包含服務提供者）都可在區塊鏈上對智能合約進行操作，以留下評

價，藉此形成共識達成信任。

在 Wallet 中實作的證照管理功能，使用者可輕易從 Wallet 中直接讀取共識層內的信任關係，並直接在共識層中更新自己身分證照的狀態，以及對他人或服務的證照評價。這樣的設計不僅提高了使用者的自主權，更讓使用者能更直接地參與共識層的運作。

#### 3.2.5.4 數據存儲

Wallet 的數據存儲功能主要用於儲存使用者的各類數據，包括證照、公私鑰、個人資料以及與各服務的交互紀錄等。這種設計使 Wallet 成為使用者的個人化數據中心，實現統一管理。然而，將個人移動設備轉化為數據中心需解決備份、遷移和雲端儲存等問題。雖然我們鼓勵使用者自主選擇解決方案，但仍提供以下建議實作：

每個採用自主身分系統的應用都應包含 Wallet 模塊，並利用設備的嵌入式資料庫存放數據。使用者可設置各 Wallet 的同步策略，包括自動同步（按需獲取數據）、完全同步（複製全部使用者數據）和手動同步（使用者指定數據複製）。這種設計便於實現備份、遷移和雲端儲存等功能，提升使用者數據管理的便利性。具體實現概念如下：

- **遷移**：新 Wallet 可通過手動輸入舊 Wallet 的公開地址或直接連接同一設備上的已啟動 Wallet 來建立連結。遷移後，新 Wallet 可設置對舊 Wallet 的同步策略。
- **雲端儲存**：考慮到在單一移動設備上存儲全部個人數據的安全風險，以及終端使用者難以維護家庭點對點（Point-to-Point,P2P）集群的現實，專業雲端服務供應商可提供執行完整 Wallet 的服務。使用者支付費用後，可讓其他



Wallet 連接到此雲端 Wallet。

- **備份：**當使用者在多個設備上維護多個 Wallet，並重複存儲每項數據時，自然形成了數據備份機制。

這種多元化的數據管理策略不僅提高了數據安全性，也增強了系統的靈活性和使用者體驗。

### 3.3 資安與隱私議題

自主身分系統不僅僅要提供卓越的功能，我們還希望能在確保使用者體驗的同時，保護使用者的隱私和資訊安全。為此，我們提出了一系列的資安與隱私議題，並提出了相應的解決方案。

#### 3.3.1 身分識別問題

以自治身分為例，為了確保身分系統的去中心化，大幅度使用了區塊鏈相關技術，卻因此導致使用者體驗大幅度偏離一般使用者的需求。我們為了解決這個問題，在自主身分系統的共識層中定義了兩種登入方法：簡易登入和多因素驗證登入。期望可以根據不同情境，讓使用者選擇登入方式，以同時滿足安全性和便利性的需求。

但是，系統該如何確保使用者能使用簡易登入方式呢？畢竟簡易登入可能因提供資訊不足而被攻擊者冒充或無法成功辨識身分。為此，我們提出了「基於使用者時空的分析方法」，通過追蹤使用者每次操作時夾帶或產生的資訊來辨識使用者身分，並配合「基於危險程度的驗證機制」找出應要求多因素驗證的時機。

接著我們會詳細介紹「基於使用者時空的分析方法」與「基於危險程度的驗



證機制」這兩種特殊解決方案。



### 3.3.1.1 基於使用者時空的分析方法

每個身分本質上可視為一個隨時間變化的動態向量空間，其維度可謂無窮。每次對使用者的身分驗證都可視為取得特定時間僅含部分維度的向量。這方法的核心是讓使用者自主決定提供哪些維度，並在每次與系統互動時攜帶這些資訊。基於此概念，即讓使用者使用簡易登入，只要使用者自動攜帶裝置指紋、位置等資訊，系統即可通過比對這些資訊來進一步確認使用者身分。

1. 使用者  $i$  在時間  $t$  的身分向量：

$$\mathbf{i}(t) = (i_1(t), i_2(t), \dots, i_n(t), \dots), \quad n \rightarrow \infty \quad (3.1)$$

2. 使用者  $i$  在時間  $k$  登入時傳入的向量：

$$\mathbf{v}_k = (v_{k1}, v_{k2}, \dots, v_{km}), \quad m \ll n, \quad v_{kj} = i_j(t_k) \text{ 對某些 } j \quad (3.2)$$

3. 暫存的向量集合（假設  $\dim V = m$ ）：

$$\mathbf{v}_k = (v_{k1}, v_{k2}, \dots, v_{km}) \in V, \quad \dim V = m \ll n \quad (3.3)$$

4. 假設相似度函數為  $s$ ：

$$s : \mathbb{R}^m \times \mathbb{R}^m \rightarrow [0, 1] \quad (3.4)$$

5. 身分推測方法：

$$\text{Identity} = \arg \max_{\mathbf{v}_i \in V} s(\mathbf{v}_{\text{new}}, \mathbf{v}_i) \quad (3.5)$$

舉例來說：使用者在某次登入時提供了自己的別名、性別、所在地區等資訊，而

在下次登入時僅提供了性別、所在地區等資訊，系統可以通過比對這兩次向量來推測使用者的別名。這種方法使使用者在不提供完整資訊的情況下，仍能通過部分資訊完成身分驗證。



然而，這種方法也帶來了一些問題。例如，使用者提供的維度多寡會影響系統的準確性，甚至使用者提供的維度是否包含可變資訊會影響系統的安全性。因此，我們提出維度的選擇甚至各個維度的權重應由使用者自主決定，而系統僅提供推薦機制。這樣可讓使用者在不同情境下使用不同方法，以滿足其需求。

### 3.3.1.2 基於危險程度的驗證機制

我們提出「基於危險程度的驗證機制」，並結合「基於使用者時空的分析方法」，以適應不同情境下使用者對身分敏感度的變化。此機制在比較身分與歷史紀錄時，根據情境選擇適當的比較方法和嚴謹程度。高嚴謹度要求可能導致識別困難，因為需要更多維度符合和更近時間點的紀錄；反之，低嚴謹度雖然便於識別，但可能因標準過於寬鬆而威脅系統安全。

這個機制的核心概念是將三個要素視為相互影響的因素：行為的危險程度、符合的維度數量，以及時間點的接近程度。這三者的綜合考量決定了驗證的嚴謹程度。具體而言，使用者行為的危險程度與所需的驗證嚴謹度呈正相關，而驗證的嚴謹程度又與符合的維度數量和時間點的接近程度成正相關。換言之，越危險的行為需要越嚴格的驗證，而越嚴格的驗證則要求更多的維度符合和更接近的時間點。

值得注意的是，在我們的系統中，使用者行為的危險程度是由使用者自主決定的，系統僅提供推薦機制。這種設計旨在滿足使用者在各種情境下的需求。舉例來說，對於個人銀行帳戶，使用者可能將所有行為都視為最高危險，因此需要

最嚴格的驗證。相對地，對於個人社群帳戶，讀取文章可能被視為低風險行為，而發布文章則可能被認為較為危險，因此需要不同程度的驗證嚴謹度。



我們建議使用者將危險程度的概念定義為對數據變動的敏感程度：越敏感的數據越危險，越危險的數據則需要越嚴謹的驗證機制。通過這種方式，我們希望能夠為各種使用情境提供靈活且安全的身分驗證解決方案。

### 3.3.2 密碼救援問題

密碼救援問題是自主身分系統中的一個重要問題。在傳統身分系統中，密碼救援通常是通過中心化的機制來實現的，其本質的概念就如同系統內存在著一位管理者，共同管理著使用者的身分。但這明顯違背了自主身分追求的價值觀，因此我們提出了「極致多因素認證」的概念。

允許使用者在創建自主證照時設置多種驗證方案，包括難以遺失或忘記的方法，如生物特徵識別（指紋或臉部辨識）。如此，即便使用者忘記密碼，亦可通過這些驗證方式重新取回身分。進一步擴展此方案，我們還可以將「極致多因素認證」概念應用於「基於危險程度的驗證機制」，根據危險程度要求使用者使用多種方法完成多因素驗證，進一步提高安全性，讓使用者即使已經被攻擊者奪取身分，也能通過補充更多項驗證的方式來取回身分。

#### 3.3.2.1 極致多因素認證

傳統身分驗證系統中，多因素認證主要被視為登入時的固定驗證方案。然而，在自主身分系統的設計中，為了實現使用者功能的自由管理，我們需要一種更靈活、更強大的驗證機制，以取代傳統系統中系統管理員的角色。為此，我們提出了「極致多因素認證」的概念。這一新概念的核心在於：不再將特定的多因素驗

證方法視為使用者登入的唯一途徑。相反，從服務器的角度來看，每個驗證因素都應被視為使用者向服務器自主證明身分的一種方法。這種方法的靈活性體現在：

- 在高風險情況下，系統可能要求連續多種因素的驗證以確保安全。
- 在低風險環境中，可能僅需一種因素即可完成驗證。
- 即便使用者遺失了某個重要的驗證因素，系統也不應將其視為無法登入，而是要求提供更多其他因素的驗證來補償。

通過這種動態和適應性的方法，極致多因素認證不僅提高了系統的安全性，還增強了使用者的身分管理自主權，為自主身分系統提供了一個更加靈活和強大的驗證框架。

### 3.3.3 被遺忘權

雖然在自主身分系統中，由使用者的個人裝置管理所有數據，但這並不意味著解決了被遺忘權的問題。實際上，不管是使用者與他人互動後其他人存下的數據亦或是單純為某個行為的日誌，都不太可能被完全遺忘。為此，我們提出當使用者希望遺忘數據時，除了直接清除個人保留的數據，還能讓其他人無法再利用這些數據指向使用者本身。因此不管是「數據證照」還是「自主證照」都確保了區塊鏈中僅保留雜湊，而不保留數據本身。理論上，服務內部不會存儲使用者數據；即使確實存儲了，在自主身分中唯一能證明數據擁有的是使用者本身，因此相當於使用者與數據無關。這個概念類似於 Cameron[7] 所描述的單向身分：使用者可以通過個人證明指向自己的數據，但僅有數據無法指向使用者。



### 3.3.4 積極數據授權

從近年來的 GDPR 研究中就可以發現，積極數據授權是最多大型企業所面臨的難題。在自主身分系統中，各個服務一樣也會經手使用者的數據，因此我們也提出一個機制來解決這個問題。這個機制的核心在於：使用者與服務提供者之間的數據授權應該是明確的，公平的。

透過數據層的 API 與服務互動時，使用者與服務應該使用「數據證照」來明確定義數據的內容與授權。使用者與服務都應該要留存該證照，並在區塊鏈上留下雜湊。這樣，當數據被濫用時，使用者可以通過公開證照證明數據被濫用，反之，服務提供者也可以通過公開證照證明數據被合法使用。這樣的設計讓使用者與服務提供者之間的數據授權變得更加明確且公平。

## 3.4 資料結構

我們會分別介紹自主身分系統中各角色重要物件的資料結構，以此讓讀者更清晰的了解系統的設計。

### 3.4.1 共識核心

共識層由不同的智能合約組成，每個智能合約都有自己的資料結構。這裡我們將介紹「自主證照」和「數據證照」機制所需智能合約的參考資料結構。

- **自主身分唯一編號**：透過 UUID 生成，用於識別使用者，此欄位在重視隱私的情況下可不填。
- **證照操作公鑰**：對應私鑰被註記在明文證照中，因此智能合約可以確保操作



者真的有得到實際的證照。

- **證照雜湊**：擁有明文證照的系統使用者可以藉此完成證照真實性的驗證。
- **證照狀態**：用於記錄證照的當前狀態，可能包括「正常」、「失效」、「遺忘」等。
- **證照評價**：用於記錄證照的評價，可能包括「好」、「壞」、「中立」等，並且同時可以紀錄評價者的 AID。



### 3.4.2 AID Server

服務層中一般會暫存使用者的明文證照與關聯服務商業邏輯的資料結構，我們這裡僅介紹明文證照的參考資料結構。

- **自主身分唯一編號**：透過 UUID 生成，用於識別使用者。
- **證照操作私鑰**：對應公鑰被註記在智能合約中，因此智能合約可以確保操作者真的有得到實際的證照。
- **簡易識別方案**：列出使用者想要使用的簡易登入方案與對應辨識內容，如「帳號密碼」、「IP 綁定」等。
- **多因素驗證方案**：列出使用者想要使用的多因素驗證方案與對應辨識內容，如「簡訊驗證」、「硬體金鑰」等。
- **個人資訊**：用於揭露使用者願意提供的個人資訊，如「姓名」、「性別」等。
- **服務設定**：用於記錄使用者希望自主設定的內容，如證照的權限、有效期等。

拿簡易識別方案舉例，明文證照就會包含「帳號」與「密碼」兩個欄位，而「IP 綁定」則會包含「IP 位址」欄位。多因素驗證方案的資料結構也類似，如「簡訊驗證」會包含「手機號碼」欄位。



另外，為了確保簡易登入的安全性，我們也建議服務提供者暫存使用者每次識別空間與對應時間，以便透過「基於使用者時空的分析方法」來識別使用者。接著提供我們建議的暫存資料結構。

- 使用者身分：自主證照明文。
- 使用者時間：用於記錄使用者每次識別的時間。
- 使用者空間：用於記錄使用者每次識別所拿到的所有數據，包含裝置指紋、IP 位址等。

### 3.4.3 Wallet

數據層中一般會存儲使用者的所有數據，我們將介紹所有數據的參考資料結構。

- 自主身分列表同一個 Wallet 可以管理多個 AID。
  - 自主身分唯一編號：透過 UUID 生成，用於識別使用者。
  - 證照列表：用於記錄 AID 的所有證照明文。
  - 使用者個資：用於記錄 AID 的所有個人資訊。
  - 使用者密鑰：用於記錄 AID 的所有公私鑰。
  - 使用者數據：用於記錄 AID 在所有服務中的數據，以列表的形式存在，可以用服務的 AID 來索引。

## 3.5 本章總結

為了全面評估新型自主身分系統相較於傳統身分系統的優勢，我們基於前文提出的評估標準製作了以下比較表（表3.2）。這個表格總結了本章討論的主要方



面，還突出了系統和目前身分系統的差異。自主身分（AID）系統通過創新設計



比較項目	中心化身分	聯合身分	使用者中心身分	自治身分	自主身分
身分控制	組織控制	多組織共同控制	身分提供者控制	使用者控制	使用者完全控制
使用者體驗	簡單但缺乏靈活性	單點登錄	單點登錄	複雜的區塊鏈操作	無摩擦驗證
隱私保護	低	中等	較高	高	非常高
數據存儲	集中存儲	分散在多個組織	身分提供者與服務	區塊鏈與服務	使用者端設備
互操作性	低	中等	較高	高	高
單點故障風險	高	中等	較低	低	極低
可擴展性	低	中等	較高	高	高
法規遵循	困難	較困難	較容易	容易	容易
信任模型	中心化信任	聯盟信任	身分提供者信任	去中心化信任	自主互信
身分證明	中心化驗證	聯盟驗證	OAuth 等協議	區塊鏈驗證	自主證照機制
數據共享	直接共享	組織間共享	API 授權共享	有限共享	數據證照機制

Table 3.2: 各代身分管理系統比較

成功解決了現有身分管理系統面臨的諸多挑戰，同時從以下幾個方面達到嚴格的評估標準：

### 3.5.1 創新突破

- **使用者體驗：**系統採用簡易登入和多因素驗證相結合的方式，並引入基於使用者時空的分析方法，大幅提升了驗證的便利性。
- **使用者認知：**自主身分可以自由地參與服務，不需要額外創建大量身分，減輕了認知負擔。
- **隱私保護：**採用「數據層反轉」策略，將使用者數據完全遷移至使用者端設備，極大程度地保護了使用者隱私。
- **平等信任：**引入去中心化評價機制，建立了互相驗證和互相信任的機制。



### 3.5.2 法規遵循

- 系統的設計充分考慮了 GDPR 等法規要求，特別是在使用者同意、數據可控性和被遺忘權等方面。
- 多因素驗證和風險評估等功能符合 NIST 等標準的要求。

### 3.5.3 公認原則

- 身分法則：系統設計符合 Kim Cameron 提出的身分管理基本法則，特別是在使用者控制、最小化披露等方面。
- 避免常見缺陷：例如通過創新的密碼救援機制（極致多因素認證）等設計，克服了傳統身分管理系統的多項缺陷。
- 自治身分：整個系統的設計理念與 Allen 提出的自治身分 10 大原則高度一致，真正實現了使用者對自身身分的完全控制。

總的來說，這個自主身分系統通過創新的技術設計和機制，成功解決了現有身分管理系統的主要困境，同時在使用者體驗、隱私保護、法規遵循和公認原則等方面都達到了嚴格的評估標準，為下一代身分管理系統提供了一個全面且可行的解決方案。





## 第四章 系統實作

本研究已成功實現了系統的核心功能，並在受控的測試環境中進行了全面的概念驗證（Proof of Concept, PoC）。系統的驗證對象包含兩個主要服務，一是遵循微軟 AI 聊天系統規格 [27] 的典型 AI 聊天服務；二是一個假想的第三方支付服務。使用者可以使用自主身分，在支付費用的前提下使用 AI 聊天服務。本章將詳細介紹系統的架構和實現細節，並通過流程分析來進一步展現自主身分系統的應用價值。

### 4.1 系統架構

整個自主身分系統的概念驗證分成多個模塊如圖4.1，以下按照分層架構來介紹：



Figure 4.1: AID 概念驗證架構簡圖

- **共識層**: 一條區塊鏈，用於存放「自主證照」和「數據證照」的智能合約。
- **服務層**: 實際與使用者產生互動的服務，包括：
  - 一個完成證照所需的簽章服務，用於生成「自主證照」。
  - 一個 AI 聊天服務，用於提供 AI 聊天功能。
  - 一個支付服務，用於提供 AI 聊天所需的支付功能。
- **使用者層**: 使用者的手機應用程式，一對一對應到服務層的服務。



## 4.2 實現細節

Table 4.1: 系統分層結構及介面

層次	實現的介面
共識層	<ul style="list-style-type: none"> <li>- 初始化與獲取 AID 資訊</li> <li>- 設置與獲取證照評論</li> </ul>
服務層	<ul style="list-style-type: none"> <li>- 簽名證照並上鏈</li> <li>- 驗證證照與管理使用者數據</li> <li>- 點數驗證與證明生成</li> <li>- 登出時數據處理</li> <li>- 身分驗證與登入</li> <li>- 生成與回傳交易證照</li> </ul>
使用者層	<ul style="list-style-type: none"> <li>- GUI 開發 (Flutter)</li> <li>- 本地數據存儲 (Hive)</li> <li>- 跨應用數據共享</li> </ul>

如4.1所示，我們實現了系統的各個層次，並提供了相應的介面。

在共識層中，我們挑選了 OurChain[30] 作為區塊鏈的實作，並且在其上實現了智能合約。智能合約的介面如下：

- **initAID**: 設置合約的基本資訊，本實作包含擁有者 AID 和證照簽名。
- **getAIDInfo**: 獲取合約的擁有者 AID 和證照簽名。
- **setComment**: 設置針對證照的評論，本實作提供純文字紀錄。
- **getComment**: 獲取針對證照的評論。

服務層中，我們首先利用 Golang 調用 OurChain 的 RPC 接口來實現智能合約的操作，完成了簽章服務。簽章服務的介面如下：



- **signAID**: 使用者傳入透過「自主證照」生成的明文證照，服務端將其簽名後上鏈，並返回簽名後的證照與合約地址。

雖然 AI 聊天服務和支付服務的具體實現與本研究無直接關係，但我們主要探討如何在這些服務中嵌入自主身分系統。

在 AI 服務中，使用者必須使用自主身分登入，並擁有足夠的點數才能使用服務。因此我們實作了以下機制：

- 驗證並暫存使用者的「自主證照」，實現簡易登入。
- 暫存並管理使用者數據（點數、歷史紀錄）。
- 要求使用者上傳「數據證照」以驗證點數，並生成新的點數證明。
- 登出時回傳使用者數據並清除服務端資料。

在支付系統中，我們同樣加入了自主身分系統。其包含以下機制：

- 基於「自主證照」進行身分驗證和複雜登入。
- 支付後生成「數據證照」作為交易證照。
- 不存儲使用者資訊，僅回傳「數據證照」。

最後，關於數據層的實現，我們採取了以下措施：

- 基於 Flutter 框架開發跨平台網絡前端應用，為每個服務提供直觀且功能豐富的圖形使用者界面（GUI）
- 使用 Google 開發的 Hive 套件，以嵌入式資料庫的形式將所有使用者數據存儲在使用者的手機上，確保數據安全
- 利用作業系統的剪貼簿共享功能，實現使用者在不同前端應用中輕易共享數

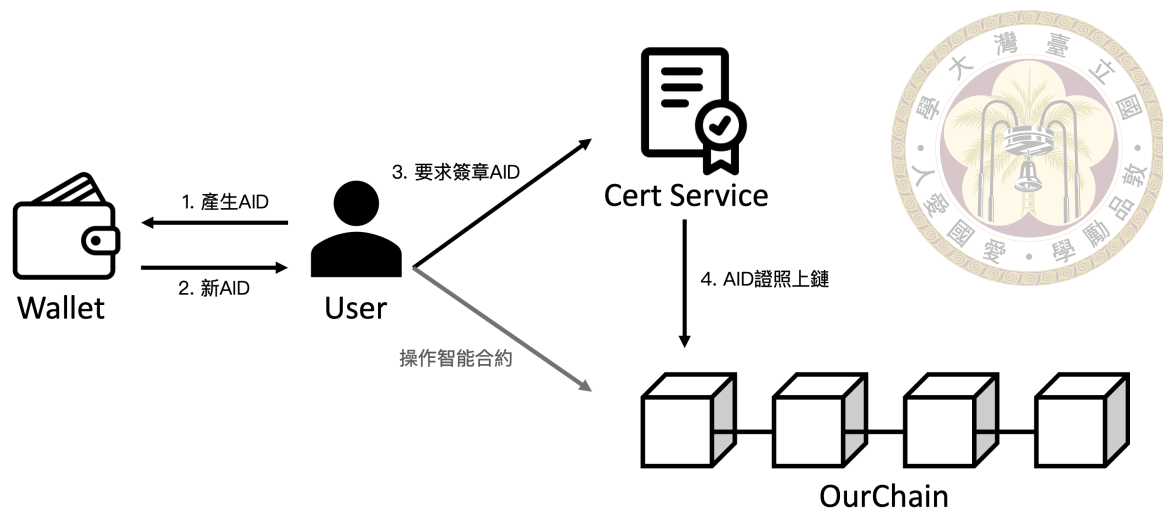


Figure 4.2: 產生新的 AID 與自主證照

據

雖然這並不是一個完整的實作，但我們已經成功證明了自主身分系統的可行性，接下來我們將通過流程分析來展示自主身分系統的應用價值。

## 4.3 流程分析

我們將找出三個典型的用例，配合簡圖與細節流程來展示自主身分系統的應用價值。這三個用例分別是：

- 產生新的 AID 與自主證照
- 進入支付服務獲取收據
- 使用 AI 服務對話

### 4.3.1 產生新的 AID 與自主證照

如圖4.2，以下是產生新的 AID 與自主證照的流程：

1. 使用者在任意支援自主身分的前端應用中，點擊「產生新的 AID」

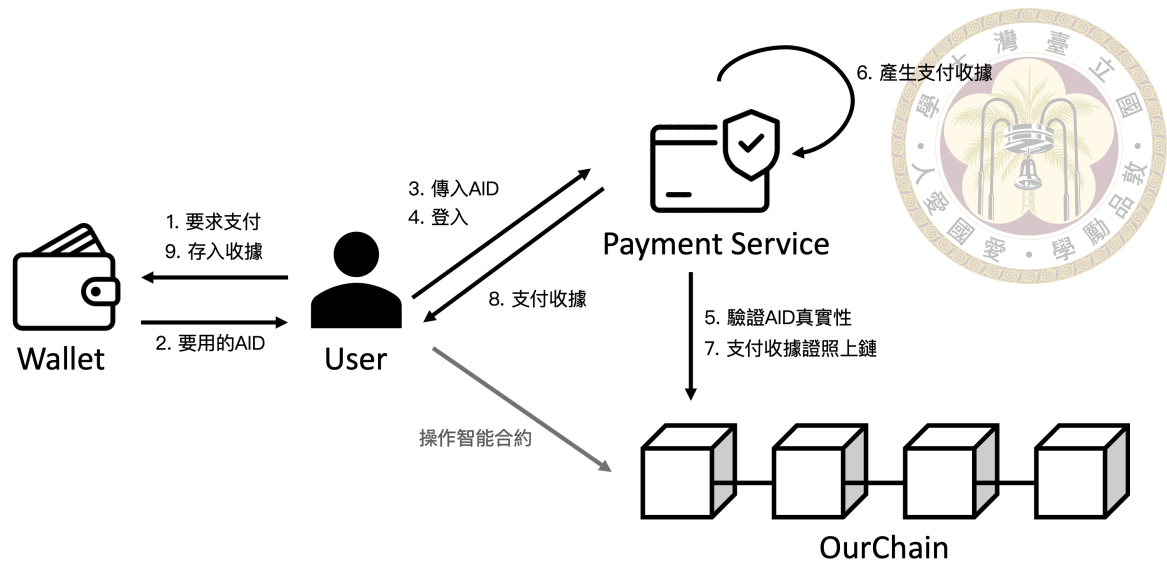


Figure 4.3: 進入支付服務獲取收據

- (a) 在裝置內部基於 UUID 機制產生唯一識別號後存儲
2. 使用者在簽章服務的前端介面中，選擇 AID 並且點擊「簽章」，產生表單讓使用者填寫資料
3. 使用者填寫個人資訊，包含別名與 pin 碼，並且點擊「確認」
  - (a) 透過使用者填入的資訊與 AID 中的識別號，生成未簽名「自主證照」
  - (b) 把自主證照傳入服務層產生簽名並上鏈
  - (c) 服務端回傳簽名後的「自主證照」與合約地址，使用者存儲在個人設備

通過這樣的流程，使用者自主的生成了自己的 AID 和自主證照，整個使用者識別不依賴單一服務商，而是由使用者自己控制。

### 4.3.2 進入支付服務獲取收據

如圖4.3，以下是進入支付服務獲取收據的流程：

1. 使用者在支付服務的前端介面中，選擇 AID 並且點擊「支付」
  - (a) 自動取用設備上存放的「自主證照」，上傳至服務端，觸發複雜登入
  - (b) 數據層的前端介面自動取用設備上的私鑰產生簽章完成登入





Figure 4.4: 產生新的 AID 與自主證照

- (c) 服務端驗證簽章後，完成支付，於是生成「數據證照」並上鏈
- (d) 服務端回傳「數據證照」給使用者存儲在個人設備
- 2. 使用者在支付服務的前端介面中，點擊「查看收據」

通過這樣的流程，使用者可以在支付後獲取到一份「數據證照」，作為交易的證照，使用者可以在任何時候查看這份證照，而不需要依賴支付服務商。

### 4.3.3 使用 AI 服務對話

如圖4.4，以下是使用 AI 服務對話的流程：

- 1. 登入:
  - (a) 使用者進入 AI 服務的前端介面
  - (b) 傳入選定 AID 的「自主證照」，觸發簡易登入
  - (c) 使用者使用證照上的別名與密碼登入
  - (d) 登入後即開始 AI 服務中的一次對話
- 2. 對話前:
  - (a) AI 服務接受數據層中上傳的使用者點數與歷史數據

- (b) 要求使用者上傳點數的「數據證照」，以確保點數真實性

### 3. 對話中:

- (a) 使用者與 AI 進行對話
- (b) 每次對話結束後，AI 服務更新暫存的使用者點數與歷史數據
- (c) 使用者即便離開對話，也只需使用別名與密碼簡易登入即可繼續，無需重新上傳數據

### 4. 登出:

- (a) 使用者選擇離開系統
- (b) AI 服務將使用者點數、歷史數據與相關「數據證照」回傳至個人裝置
- (c) 系統刪除所有使用者數據

通過這樣的流程，我們實現了數據與服務分離的目標，讓使用者能夠更加方便地管理自己的數據。

## 4.4 本章小結

儘管本實現並非完整的商業系統，但已成功證明了自主身分系統的可行性和潛在價值。透過在 AI 聊天服務、支付系統等實際場景中的應用，展現了該系統在提升使用者隱私、資料安全和身分管理方面的巨大潛力。未來研究可進一步探討該系統在更廣泛領域的應用，以及在大規模商業環境中的實施策略。







## 第五章 結論與未來展望

本研究深入探討了自主身分（AID）系統，這是一個旨在解決現有序分管理系統挑戰的創新解決方案。通過將「自主」的概念融入數位身分管理，AID 系統致力於在保障道德標準的同時，賦予使用者對其身分信息和數據的完全控制權。

### 5.1 研究結論

儘管 AID 系統通過結合無數創新的機制與系統架構，已展現出巨大的潛力，但作為一個新興的研究領域，仍有許多方面值得未來深入探討：

1. **艱鉅的推廣任務：** AID 系統可以說是一個全新的身分管理系統，需要克服使用者習慣、技術標準、法律法規等多方面的障礙。未來需要進一步研究如何推廣 AID 系統，提高使用者接受度和市場競爭力。
2. **更高的開發難度：** AID 系統不但使用了區塊鏈等新技術，還引入了反轉數據層等新概念，這將對開發人員的技術水平提出更高的要求。未來需要進一步研究如何降低開發難度。
3. **大規模應用與部署：** 目前，AID 系統的概念驗證主要集中在 AI 聊天服務和支付系統等有限的場景。未來需要進一步研究如何將 AID 系統應用於更廣泛的領域，例如金融、醫療、政務等，並探討大規模部署的可行性和挑戰。



## 5.2 未來展望

我們將從以下幾個領域展望 AID 系統的未來發展：

- 金融
- 醫療
- 政務

這些領域都是當前身分管理系統存在問題的重要應用場景，AID 系統的應用將為這些領域帶來深遠的影響。

## 5.3 總結

總而言之，自主身分系統代表了數字身分管理的未來方向，它為構建一個更加安全、自由、便捷和公平的數位社會提供了巨大的潛力。相信通過持續的研究和創新，AID 系統將在未來展現出更廣闊的應用前景，並為人類社會帶來深遠的影響。



## 參考文獻

- [1] A. Acquisti, I. Adjerid, R. Balebako, L. Brandimarte, L. F. Cranor, S. Komanduri, P. G. Leon, N. Sadeh, F. Schaub, M. Sleeper, Y. Wang, and S. Wilson. Nudges for privacy and security: Understanding and assisting users' choices online. ACM Comput. Surv., 50(3), aug 2017. 隱私設置的研究.
- [2] G.-J. Ahn and M. Ko. User-centric privacy management for federated identity management. In 2007 International Conference on Collaborative Computing: Networking, Applications and Worksharing (CollaborateCom 2007), pages 187–195, 2007.
- [3] F. Alaca and P. C. van Oorschot. Device fingerprinting for augmenting web authentication: classification and analysis of methods. In Proceedings of the 32nd Annual Conference on Computer Security Applications, ACSAC '16, pages 289–301, New York, NY, USA, 2016. Association for Computing Machinery.
- [4] C. Allen. The path to self-sovereign identity, 4 2016.
- [5] M. S. Blumenthal and D. D. Clark. Rethinking the design of the internet: the end-to-end arguments vs. the brave new world. ACM Trans. Internet Technol., 1(1), aug 2001.
- [6] J. Bonneau, C. Herley, P. C. v. Oorschot, and F. Stajano. The quest to replace pass-

words: A framework for comparative evaluation of web authentication schemes. In 2012 IEEE Symposium on Security and Privacy, pages 553–567, 2012.




- [7] K. Cameron. The laws of identity. White paper, Microsoft Corporation, 2005.
- [8] U. W. Chohan. Decentralized autonomous organizations (daos): Their present and future. March 2024. Available at SSRN: <https://ssrn.com/abstract=3082055>.
- [9] D. D. Clark, J. Wroclawski, K. R. Sollins, and R. Braden. Tussle in cyberspace: defining tomorrow's internet. SIGCOMM Comput. Commun. Rev., 32(4), aug 2002.
- [10] R. Dhamija and L. Dusseault. The seven flaws of identity management: Usability and security challenges. IEEE Security and Privacy, 6(2):24–29, 2008.
- [11] European Parliament and Council of the European Union. General Data Protection Regulation (GDPR), 2016. Regulation (EU) 2016/679.
- [12] M. Finck. Blockchains and data protection in the european union. Eur. Data Prot. L. Rev., 4:17, 2018.
- [13] S. Ghorbani Lyastani, M. Schilling, M. Neumayr, M. Backes, and S. Bugiel. Is fido2 the kingslayer of user authentication? a comparative usability study of fido2 passwordless authentication. In 2020 IEEE Symposium on Security and Privacy (SP), pages 268–285, 2020.
- [14] M. Goodner and A. Nadalin. Web services federation language (ws-federation) version 1.2. OASIS Standard, 2009.
- [15] Google Cloud. Best practices for planning your identity architecture. <https://cloud.google.com/architecture/identity/>

[best-practices-for-planning#combine\\_cloud\\_identity\\_and\\_g\\_suite\\_in\\_a\\_single\\_account](#), 2024. Accessed: 2024-07-08.



- [16] T. Hamme, V. Rimmer, D. Preuveneers, W. Joosen, M. A. Mustafa, A. Abidin, and E. Argones Rúa. Frictionless authentication systems: Emerging trends, research challenges and opportunities. 09 2017.
- [17] Hard. The oauth 2.0 authorization framework. IETF RFC 6749, 2012.
- [18] A. Jøsang, S. Marsh, and S. Pope. Exploring different types of trust propagation. In K. Stølen, W. H. Winsborough, F. Martinelli, and F. Massacci, editors, Trust Management, pages 179–192, Berlin, Heidelberg, 2006. Springer Berlin Heidelberg. 信任傳播.
- [19] A. Jøsang, M. A. Zomai, and S. Suriadi. Usability and privacy in identity management architectures. In Proceedings of the Fifth Australasian Symposium on ACSW Frontiers - Volume 68, ACSW '07, pages 143–152, AUS, 2007. Australian Computer Society, Inc. 使用這認知過度的文章.
- [20] M. Kubach, C. H. Schunck, R. Sellung, and H. Roßnagel. Self-sovereign and decentralized identity as the future of identity management? In H. Roßnagel, C. H. Schunck, S. Mödersheim, and D. Hühnlein, editors, Open Identity Summit 2020, Lecture Notes in Informatics (LNI), pages 35–47, Bonn, 2020. Gesellschaft für Informatik. DID UX.
- [21] LastPass. Psychology of passwords: The online behavior that’s putting you at risk, 2020.
- [22] P. Leach, M. Mealling, and R. Salz. A universally unique identifier (uuid) urn namespace. 01 2005.



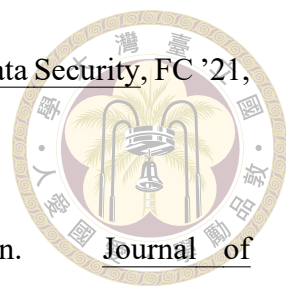
- 
- [23] L. Lessig. Code is law: On liberty in cyberspace. Harvard Magazine, 2000.
- [24] C. Lundkvist, R. Heck, J. Torstensson, Z. Mitton, and M. Sena. uport: A platform for self-sovereign identity. [https://publications.aston.ac.uk/id/eprint/42147/1/uPort\\_SSI\\_DrNitinNaik.pdf](https://publications.aston.ac.uk/id/eprint/42147/1/uPort_SSI_DrNitinNaik.pdf), 2017.
- [25] Microsoft. Ion - we have liftoff! Microsoft Azure Blog, 2020.
- [26] Microsoft. Active directory domain services overview. <https://docs.microsoft.com/en-us/windows-server/identity/ad-ds/get-started/virtual-dc/active-directory-domain-services-overview>, 2021.
- [27] Microsoft. Ai chat protocol. <https://github.com/microsoft/ai-chat-protocol/tree/main/spec>, 2024. Accessed: 2024-07-17.
- [28] National Institute of Standards and Technology. Digital identity guidelines. Special Publication 800-63-3, National Institute of Standards and Technology, Gaithersburg, MD, June 2017.
- [29] P. Nikander, A. Gurtov, and T. R. Henderson. Host identity protocol (hip): Connectivity, mobility, multi-homing, security, and privacy over ipv4 and ipv6 networks. IEEE Communications Surveys & Tutorials, 12(2):186–204, 2010.
- [30] NTU CSIE Lab408. Ourchain. <https://github.com/OurLab408/OurChain>, 2024. Accessed: 2024-07-17.
- [31] OASIS. Security assertion markup language (saml) v2.0 technical overview. Technical report, OASIS Committee Draft, 2005.
- [32] A. Preukschat and D. Reed. Self-Sovereign Identity: Decentralized Digital Identity

and Verifiable Credentials. Manning Publications, May 2021. Available translations: Korean, Simplified Chinese.



- [33] M. Saemann, D. Theis, T. Urban, and M. Degeling. Investigating gdpr fines in the light of data flows. Proceedings on Privacy Enhancing Technologies, 2022.
- [34] N. Sakimura. Openid connect core 1.0. OpenID Foundation.
- [35] F. Schardong and R. Custódio. Self-sovereign identity: A systematic review, mapping and taxonomy. Sensors, 22(15), 2022.
- [36] J. Sermersheim. Lightweight directory access protocol (ldap): The protocol. IETF RFC 4511, 2006. 輕量化 AD.
- [37] Y. Smirnova and V. Travieso-Morales. Understanding challenges of gdpr implementation in business enterprises: a systematic literature review. International Journal of Law and Management, 66:326–344, 01 2024.
- [38] R. Soltani, U. T. Nguyen, and A. An. A survey of self-sovereign identity ecosystem. Security and Communication Networks, 2021(1):8873429, 2021.
- [39] S.-T. Sun and K. Beznosov. The devil is in the (implementation) details: an empirical analysis of oauth sso systems. In Proceedings of the 2012 ACM Conference on Computer and Communications Security, CCS '12, pages 378–390, New York, NY, USA, 2012. Association for Computing Machinery. OAuth 2.0 實作之漏洞, 建議更改.
- [40] S. Wiefeling, M. Dürmuth, and L. Lo Iacono. What's in Score for Website Users: A Data-Driven Long-Term Study on Risk-Based Authentication Characteristics. In

25th International Conference on Financial Cryptography and Data Security, FC '21,  
pages 361–381. Springer, Mar. 2021.

- 
- [41] T. Wu. Network neutrality, broadband discrimination. Journal of Telecommunications and High Technology Law, 2:141, 2003.
- [42] T.-N. Wu. The design and implementation of general autonomous certification on blockchain. Master's thesis, National Taiwan University, Taiwan, 2021. Department of Computer Science and Information Engineering.
- [43] Y. Zhang, F. Monrose, and M. K. Reiter. The security of modern password expiration: an algorithmic framework and empirical analysis. In Proceedings of the 17th ACM Conference on Computer and Communications Security, CCS '10, pages 176–186, New York, NY, USA, 2010. Association for Computing Machinery.



## 附錄 A — 模型參數表

### A.1 模型一

### A.2 模型二





## 附錄 B — BraTS 2021 分割結果圖

**B.1 編號 0001 0050**

**B.2 編號 0051 0100**