

Intelligent Approaches for Robust Blockchain-based Identity Management

by Mekhled Alharbi

Thesis submitted in fulfilment of the requirements for
the degree of

Doctor of Philosophy

under the supervision of Professor Farookh Hussain

University of Technology Sydney
Faculty of Engineering and Information Technology

February 2023

Certificate of Original Authorship

I, Mekhled Alharbi, declare that this thesis is submitted in fulfilment of the requirements for the award of Doctor of Philosophy, in the Faculty of Engineering and Information Technology at the University of Technology Sydney.

This thesis is wholly my own work unless otherwise reference or acknowledged. In addition, I certify that all information sources and literature used are indicated in the thesis.

This document has not been submitted for qualifications at any other academic institution.

This research is supported by the Australian Government Research Training Program.

Production Note:

Signature: Signature removed prior to publication.

Date: February 2023

Acknowledgements

I would like to acknowledge the blessings of Almighty Allah, the most gracious and beneficent, for giving me this opportunity to write this thesis.

I would like to express my deep and sincere gratitude to my supervisor, Professor Farookh Hussain, to whom I am indebted for his continuous support, encouragement, and guidance throughout my study. This PhD thesis would not have been possible without his motivation and patience.

This study has been a long journey and one of the most challenging endeavours for my family, wife and children.

I would like to express my warmest thanks to my mother for her endless care, inspiration and encouragement for without her support and prayers, this PhD would not have been accomplished.

I would also like to express offer deepest appreciation to my brothers and sisters for their unending support and encouragement.

My deepest thanks and gratitude go to my wife for standing by my side, for her understanding, and for caring for our children when I was away from home throughout my PhD journey.

A special thanks to my lovely children, Malek, Hossam, and Mazen for their love, patience and support throughout these years and for understanding that their dad had to study and sometimes had to be far from them. You spent years of your childhood outside your home country and learned a second language, to support my ambition. I hope that one day these words will be an inspiration to you in your life and will motivate you to achieve your dreams.

ABSTRACT

Smart contracts, which are maintained on blockchain, are self-executing protocols designed to monitor and confirm the fulfilment of a contract's terms. The trustworthiness of these contracts is guaranteed by these protocols, which also excludes any intermediaries from the transactions. Blockchain is a modern technology with rapidly expanding significance that is used in many applications, such as financial transactions, smart cities, and share trading. Currently, users' identities are stored and managed by service providers using their centralized system. Identity information management is usually undertaken by the providers which raises concerns about user privacy and trustworthiness. Blockchain technology has the potential to enhance the identity management domain by eliminating the need for a trusted intermediary. However, the advent of blockchain technology has led to new identity management concepts to tackle trustworthiness and privacy challenges, granting users control over their information. Blockchain is suitable for situations requiring both trust and transparency due to its inherent characteristics. Therefore, there is a critical need to develop intelligent approaches to manage user identity information in a reliable manner. Thus, we tackle this issue by providing a solution that combines the mechanism of identity management with smart contracts based on blockchain and the use of artificial intelligence.

We performed a systematic literature review to deepen our understanding of the issues and solutions employed in addressing these challenges to identify the drawbacks of the existing methods in the field of identity management. In the existing literature, no solution has been proposed to manage user identities in a way that guarantees data privacy and trustworthiness through the use of blockchain-based smart contracts and artificial intelligence techniques. The use of blockchain based on smart contracts has the potential to play a significant part in identity management by improving transparency and privacy.

In this thesis, we develop intelligent approaches to solve the aforementioned research issue. We integrate blockchain-based smart contracts with identity management to detect duplicate user identities while maintaining the privacy of the data of these identities, thus multiple machine learning approaches are proposed to detect duplicate users' identities on top of blockchain. We also develop an early warning system to generate alerts for users whose identities are nearing expiration. Furthermore, we propose an algorithm to intelligently compute the trustworthiness score of a user's identity based on the identity documents provided by the user, which are stored safely, hence boosting confidence in the users' trustworthiness score. Finally, a software prototype is selected to validate the performance of the methods proposed in this thesis.

List of Publications

The following is a list of my research papers during my PhD study.

Conference Papers

- C-1. Alharbi, M. and Hussain, F.K., 2021, October. Blockchain-based identity management for personal data: a survey. In International Conference on Broadband and Wireless Computing, Communication and Applications (pp. 167-178). Springer, Cham.
- C-2. Alharbi, M. and Hussain, F.K., 2022. A Systematic Literature Review of Blockchain Technology for Identity Management. In International Conference on Advanced Information Networking and Applications (pp. 345-359). Springer, Cham.

Journal Papers

- J-1. Alharbi, M. and Hussain, F.K. and Hussain, O.K., 2023. A Comprehensive Identity Management Framework based on Integrating Blockchain and Machine Learning, In International Journal of Web and Grid Services. (under review).

Table of Contents

Certificate	ii
Acknowledgments	iii
Abstract	iv
List of Publications	vi
List of Figures	xiii
List of Tables	xvi
1 Introduction	1
1.1 Introduction	1
1.2 Problem Statement	4
1.3 Research Challenges	5
1.4 Objectives of the thesis	7
1.5 Significance of the Thesis	7
1.5.1 Scientific Contributions	8
1.5.2 Social Contribution	8
1.6 Structure of the Thesis	8
1.7 Conclusion	10
2 Literature Review	12
2.1 Introduction	12
2.2 Preliminaries	15

2.2.1	Identity management	15
2.2.2	Blockchain technology	16
2.2.2.1	Overview of blockchain	16
2.2.2.2	Smart contract	18
2.3	Systematic Literature Review (SLR) Methodology	18
2.3.1	Data source selection and search strategies	19
2.3.2	Inclusion and exclusion criteria	20
2.3.3	Citation and inclusion decision management	21
2.3.4	Final selection and quality assessment	22
2.3.5	Data extraction and synthesis	22
2.4	Results and Discussion of Existing Work	23
2.4.1	Authentication	24
2.4.2	Privacy	27
2.4.3	Trust	29
2.5	Comparative analysis of the existing research	30
2.6	Shortcomings of the existing literature on identity management based on Blockchain	30
2.7	Conclusion	32
3	Problem Definition	36
3.1	Introduction	36
3.2	Gaps in the Literature	36
3.3	Definitions of Key Concepts and Terms	36
3.3.1	Blockchain	37
3.3.2	Smart Contracts	37

3.3.3	Identity Management	37
3.3.4	Ethereum	37
3.3.5	Trustworthiness of User Identity	38
3.3.6	Service Provider	38
3.3.7	Duplicate Detection	38
3.3.8	IPFS	38
3.3.9	MetaMask	38
3.3.10	Early Warning System	39
3.3.11	Machine Learning	39
3.4	Research Questions	39
3.5	Research Objectives	41
3.6	Conclusion	42
4	Research Methodology and Solution Overview	43
4.1	Introduction	43
4.2	Solution Overview	43
4.2.1	Solution Overview for RQ1	43
4.2.2	Solution Overview for RQ2	45
4.2.3	Solution Overview for RQ3	46
4.2.4	Solution Overview for RQ4	47
4.3	Research Methodology	49
4.4	Conclusion	51
5	Identity Management Model based on the integration of Blockchain and Machine Learning	52
5.1	Introduction	52

5.2	The proposed model	54
5.2.1	Preprocessing	56
5.2.2	Indexing	56
5.2.3	Feature extraction	57
5.2.4	Classification	58
5.2.5	Storing the data in blockchain	59
5.3	Experiments	60
5.3.1	Implementation	60
5.3.2	Datasets	61
5.3.3	Evaluation Metrics	62
5.3.3.1	Accuracy	62
5.3.3.2	Recall	63
5.3.3.3	Precision	63
5.3.3.4	F-measure	63
5.4	Results and Discussion	63
5.4.1	Comparison with the state-of-the-art models	71
5.5	Conclusion	73

6 Blockchain-based method for generating notifications for user identity expiration 74

6.1	Introduction	74
6.2	Generating notifications about the expiration of user identity	76
6.2.1	Solution Workflow	76
6.2.2	Generating Alerts for Users	78
6.3	Model Implementation	78

6.3.1	Steps for generating blockchain-based warnings	79
6.4	Conclusion	80
7	Blockchain-based Model for Computing the Trustworthiness of a User's Identity	82
7.1	Introduction	82
7.2	Determination of the Trustworthiness Values of User Identities	84
7.2.1	Solution Workflow	84
7.2.2	Calculation Trustworthiness Score of User's Identity	85
7.3	Prototype Implementation	88
7.4	Prototype Evaluation and Discussion	88
7.5	Conclusion	89
8	Evaluation and Prototype Implementation	90
8.1	Introduction	90
8.2	Evaluation of the Performance of Machine Learning Techniques and the Prototype for Managing Users' Identities	91
8.2.1	Evaluation of machine learning performance	91
8.2.2	Prototype for Managing User Identities	91
8.2.3	Blockchain Configuration	92
8.3	Prototype for Generating Warnings for Users	99
8.4	Prototype for Computing the Trustworthiness of the Users' Identities .	105
8.5	Conclusion	113
9	Conclusion and Future Work	114
9.1	Introduction	114

9.2	Problems Addressed in this Thesis	114
9.3	Contributions of this thesis to the existing literature	115
9.3.1	Contribution 1: Systematic Literature Review	115
9.3.2	Contribution 2: A framework for the integration of blockchain and machine learning methods for identity management	116
9.3.3	Contribution 3: Intelligent model for generating warnings about the expiration of a user's identity	116
9.3.4	Contribution 4: Intelligent model for determining the current trustworthiness score of a user based on the user identities stored on blockchain	117
9.3.5	Contribution 5: Implementation and evaluation of the proposed solutions	117
9.4	Conclusion and Future Work	118

List of Figures

1.1	Structure of the Thesis	11
2.1	Literature Review Process	21
2.2	Number of publications by category and year	25
4.1	Overview of the proposed method for duplicate identity detection . .	44
4.2	Early Warning Model	46
4.3	The proposed model for user trustworthiness	48
4.4	Design science research methodology (Peffer et al., 2007)	50
5.1	Overview of the proposed method for duplicate identity detection . .	55
5.2	The performance of the classifiers on the Scholar-DBLP dataset . . .	69
5.3	The performance of the classifiers on the ACM-DBLP dataset	70
5.4	Sequence diagram for identity storage	71
6.1	Early Warning Model to alert users	77
6.2	The steps involved in generating warnings	79
6.3	Blockchain DApp for the proposed EWS model	80
7.1	Intelligent Method for Trustworthiness Calculation	86

7.2	An Overview of the Steps involved in the User's Trustworthiness Score Calculation	87
8.1	Remix IDE interface	92
8.2	The interface of Ganache for objective 1	93
8.3	Running of Ganache blockchain	94
8.4	Compiling the smart contract for objective 1	94
8.5	Smart contract deployment for objective 1	95
8.6	Connecting Ethereum account to Ganache blockchain	96
8.7	The DApp interface for objective 1	96
8.8	Settings of the DApp GUI	97
8.9	The file stored on IPFS	99
8.10	The proposed model for generating warnings	100
8.11	The main interface for DApp for objective 2	100
8.12	Ganache main interface for objective 2	101
8.13	The new account created through MetaMask	101
8.14	The endpoint settings	102
8.15	Deployment of smart contract on Remix for objective 2	102
8.16	Smart contract confirmation of deployment on Ganach	103
8.17	The algorithm used to check expiry dates	103
8.18	The Mailgun account details	104
8.19	The API settings	104
8.20	Adding users' information through DApp for objective 2	105
8.21	Confirmation request of the transaction for objective 2	106
8.22	The wallet balance after transaction confirmation	107

8.23	Notification advising that the email was received successfully	107
8.24	Connecting to the MetaMask account	108
8.25	MetaMask Ether Faucet	108
8.26	Main page for DApp for objective 3	109
8.27	The process of uploading documents to DApp	110
8.28	The weight value for the documents	110
8.29	Request to confirm the transaction for objective 3	111
8.30	Algorithm for computing the trustworthiness score	112
8.31	The calculated score on the user's page for objective 3	112

List of Tables

2.1	Scientific Assessment Process	22
2.2	Quality Criteria	23
2.3	Mapping of categories to respective publications	24
2.4	Comparative analysis of the existing research on blockchain-based identity management in the literature	31
2.4	Comparative analysis of the existing research on blockchain-based identity management in the literature (continued)	32
2.5	Existing research studies on identity management based on blockchain	33
2.5	Existing research studies on identity management based on blockchain (continued)	34
5.1	Overview of evaluation benchmark datasets	62
5.2	F-measure results of the classifiers	64
5.3	The runtime of the models in seconds	66
5.4	Comparison of the proposed model with state-of-the-art models . . .	72
7.1	Weighted User Identity Documents	83
7.2	Computed scores of users	89

Chapter 1

Introduction

1.1 Introduction

Technological advancements have paved the way for communication between users over the years. Such advancements facilitate the continuous success of interactions between multiple entities. Typically, online services take place between service providers and users, many of whom have not dealt with each other before. Thus, trust concerns often emerge when users take the risk of dealing with service providers before the service is even provided because the user is unable to verify that the other party to a transaction will provide the desired services. Trust is crucial for users in these situations.

When it comes to emerging technologies and how their data is handled and stored, individuals have high expectations, particularly in terms of privacy and trust, which are critical concerns. Concerns originate because sensitive and personal data is managed by personalized systems and stored in centralized systems. In such systems, the data is being managed in a central fashion, where a central authority controls all the transactions in this system in a centralized manner. Users will be unwilling to engage in new services if challenges about data management are not addressed in an appropriate manner. The users' data could be tampered with without their consent when it is stored in centralized systems. Therefore, privacy and trust issues have become increasingly important to users when confronted with new technologies (Crompton and McKenzie, 2010).

In today's digital society which offers a wide range of online services, users' identities are the cornerstone of any interactions that take place, hence managing these identities properly is a crucial concern. Therefore, a system is needed to manage user data and to identify users who wish to access multiple services. The need to ensure the privacy of users' data while also satisfying their requirements for trust is a significant obstacle brought on by the increasing demand for user data management. Furthermore, data privacy and users' trust are at stake as a result of the rapid growth of the internet. Users often have the impression that they have no control over their information, according to surveys carried out by the European Commission (El Haddouti and El Kettani, 2019). User identity systems are the core of online service communication, which can span numerous sectors that may share the same trust level. The process of establishing and managing user information to gain access to services offered by service providers is known as identity management (Warschofsky et al., 2011). Users' data is managed by a central authority in centralized systems; thus, the central authority has the ability to access and alter user identity information. Thus, the problem with these systems is that they are not able to provide adequate data privacy and trust. In addition, the information could potentially be misused as a result of breaches in these systems. Massive amounts of sensitive information have been exposed through various leaks, and there have been allegations that information has been hacked and stolen. An increasing number of sensitive and personal data have been compromised recently, as documented by the Identity Theft Resource Center (Othman and Callahan, 2020). In April 2021, it was discovered that personal data of over 533 million Facebook users from 106 countries was exposed online. The data included phone numbers, email addresses, and other details (Verge, 2021). It was reported that third-party suppliers exposed the personal information of a large number people (Rana et al., 2019). According to the 2021 Identity Fraud Study by Javelin Strategy and Research

(Buzzard and Kitten, 2021), identity fraud incidents in the United States increased by 15% in 2020, reaching 1.4 million victims, and the estimated losses from identity fraud amounted to \$56 billion in the same year. A data breach occurred in Malaysia involving the identity of more than 220 thousands organ donors in 2018 (Lim et al., 2018). A recent announcement made by Medibank in 2022 stated that personal data relating to Medibank customers' memberships had been breached. The information that was compromised included the users' names, passport numbers, genders, dates of birth, and other personal data (Medibank, 2022). Hence, it is imperative to develop systems that are able to overcome these shortcomings.

Blockchain technology has gained significant attention due to its disruptive characteristics, such as decentralization, distribution, and immutability. Thus, blockchain represents the optimal solution to overcome identity management concerns and to ensure systems are robust by granting users control over their identity (Lee, 2017). Blockchain has the potential to mitigate these issues since the data are stored in immutable, decentralized, and transparent records (El Haddouti and El Kettani, 2019). Hence, blockchain technology is an optimal solution to overcome identity management concerns. Blockchain is an immutable ledger of transactions that is shared between parties where mutual trust is not a prerequisite. In numerous applications including identity management, it is essential that irreversible data records are developed, which is a key feature of blockchain networks due to its features of transparency and decentralisation (Jacobovitz, 2016). Ethereum smart contracts offer a higher level of robustness since they are executed once they meet certain criteria. Stakeholders have the ability to implement self-executing contracts due to the use of smart contracts, which removes the need for an intermediary to be involved. Blockchain facilitates the establishment of trust between users since there is no centralized authority that controls the data. Furthermore, the decentralized nature of blockchain ensures data privacy by eliminating centralized data storage.

In addition, data immutability is guaranteed in blockchain because once the data has been recorded, it cannot be tampered with except by practically impossible majority consent. There are significant benefits to be gained by integrating blockchain technology with identity management to overcome many of the challenges associated with user identity management.

1.2 Problem Statement

In light of the aforementioned explanation, it can be seen that identity management plays a crucial role in our lives. In addition, the previous section provides an overview of the problems that are associated with identity management. Various approaches have been used in the past to advance identity management, but most of these approaches are based on a centralised architecture. Centralized identity management allows users to store their personal information on a single site. Systems relying on this architecture to manage users' identities are vulnerable to many issues such as the violation of users' privacy, denial-of-service threats, and data misuse. Furthermore, users' identities are managed by a central authority, thus users have no control over their data and privacy. Hence, data owners must trust service providers not to misuse their information. Furthermore, federated approaches share information amongst several providers. However, the fact that the data is managed by the service provider implies that users still do not have any control over their own data, posing concerns about data privacy due to a lack of transparency.

Several concerns have emerged related to various aspects of identity management, including trust and the privacy of personal and sensitive data. Identity management concerns arise about individuals' identities which contain sensitive data that should be maintained and managed in appropriate manner. Additionally, unauthorized third parties should not be able to access users' personal information based on the current regulations (Hansen et al., 2004). Consequently, identity management sys-

tems should be capable of providing users with reliable approaches to manage their identities. Multiple studies have demonstrated that blockchain technology can potentially enhance identity management. It is uncertain how service providers use users' data and what privacy and trust levels are used to manage users' information. The immutability of blockchain guarantees that data cannot be altered once it is stored. Blockchain's decentralized architecture ensures that there is no central authority to control the data by eliminating the need for a third party and it removes the potential for a single point of failure and its traceability ensures that users are able to track any modifications to their data. Therefore, the trust between users and service providers is strengthened by implementing blockchain technology.

This thesis focuses on finding solutions to issues that are associated with trust, the privacy of data, and identity management. It details the first endeavour to use smart contracts and blockchain to automatically generate alerts to users when their identities are about to expire and to compute the trustworthiness values of a user's identity based on the identity documents submitted by the user. Additionally, it is the first work of its kind to suggest an intelligent method that is built on top of blockchain using smart contracts to detect duplicate user identities and ensure the privacy of these identities.

1.3 Research Challenges

Identity management approaches have recently been developed through several academic and business research endeavours that rely on centralised systems. However, centralized systems are vulnerable to misuse and thus do not meet the users' requirements for their identities to be secure in a trustworthy environment. Despite the fact that blockchain technology and its various applications have attracted a lot of attention in a wide range of fields, to date little progress has been made in the domain of identity management. Blockchain technology has been shown to have the

ability to build robust methods to overcome the drawbacks of centralized approaches due to its decentralized nature, where no central authority can control the entire system. These features make it an ideal solution to manage user identities. Furthermore, users can manage their identity information in blockchain without having to worry about illicit use, which enhances the robustness of user identity systems through the combination of blockchain and identity management. The following key challenges and gaps in the existing research on identity management were revealed by the literature review:

1. There is a need for an intelligent model that integrates identity management and blockchain-based smart contracts to identify duplicate user identities while maintaining the privacy of these identities in a trustworthy manner.
2. There is a need for an intelligent method that can be used to develop a warning system to alert users to impending user identity expiration.
3. There is a need for a method for computing the trustworthiness value of a user's identity based on the submitted identity documents which can be stored in an immutable fashion.

In this thesis, we propose, build and implement robust blockchain-based identity management methods that overcome the aforementioned shortcomings in the current literature. These methods integrate blockchain, identity management, and machine learning techniques. This research also develops an intelligent method to generate notifications to users when their identities are about to expire and computes the trustworthiness values of users' identities. The scope of this research includes an evaluation of several models that can be implemented to maintain the privacy and trust of users.

1.4 Objectives of the thesis

The primary aim of this thesis is to build and assess intelligent methods using blockchain technology for identity management. These solutions rely on artificial intelligence since it is essential to comprehend the complexity of user identity data to ensure it is managed in a reliable environment and to empower the existing identity management techniques with intelligent approaches to handle user identity data in a decentralized manner. The thesis aims to achieve the following objectives:

- To develop an intelligent method that combines blockchain-based smart contracts and identity management to detect duplicate user identities and store them in a trustworthy manner to guarantee privacy.
- To develop an early warning system which is an intelligent approach to produce user-specific notifications based on blockchain.
- To develop an approach to calculate the trustworthiness score of a user's identity based on the submitted identity documents on blockchain.
- To evaluate the proposed methods using a prototype.

1.5 Significance of the Thesis

Managing user identities on blockchain will result in enhanced user privacy, increased trust between users and identity providers and will ensure the immutability of the user identity data. Hence, the significance of the thesis is to tackle the obstacles in the current research by presenting intelligent approaches for blockchain-based identity management using machine learning and smart contracts. The significance of the thesis can be divided into two categories, its scientific contribution and its social contribution.

1.5.1 Scientific Contributions

1. This is the first research to propose an intelligent approach to address the problem of detecting duplicate user identities on the top of the blockchain while ensuring the privacy and immutability of users' identities by combining blockchain-based smart contracts and identity management.
2. This is the first research to propose the use of the personalized date-based early warning system (EWS) to generate alerts for users based on the expiration date of their identity in blockchain.
3. This is the first research to develop a method for computing the trustworthiness score of a user's identity based on single or multiple documents as a service using smart contracts on Ethereum blockchain.

1.5.2 Social Contribution

1. Applying such intelligent mechanisms to manage data on a distributed platform will increase and enhance the user's trust in obtaining services provided by service providers. Additionally, it will contribute to paving the way for the development of trustworthy and robust identity management environments.
2. This research will assist service providers explain their benefits more precisely and effectively. In addition, service providers can concentrate on other activities, which will ultimately lead to an improvement in productivity.

1.6 Structure of the Thesis

We built several methods that enable the implementation of smart contracts for identity management on blockchain. These involve intelligent approaches for handling user identity data in a reliable manner. The thesis comprises nine chapters

as illustrated in Figure 1.1 to achieve all the research objectives. Each chapter describes the methods proposed to achieve the objectives. This section provides a brief overview of each chapter:

Chapter 1 presents a concise introduction to the field of research. The objectives of this thesis are elucidated followed by a concise outline of the challenges that must be overcome to achieve these objectives. We also present a summary of the significance of the research in both the scientific and social realms.

Chapter 2 presents a systematic literature review of the current literature on applying blockchain in the field of identity management. This chapter highlights the existing problems we plan to tackle and solve.

Chapter 3 describes the research problems that will be addressed. The research questions and objectives are based on these problems. Furthermore, this chapter provides definitions of the terminologies that are employed to describe the issues discussed throughout the thesis.

Chapter 4 gives an outline of the suggested solutions to achieve the research objectives. It also discusses the methodology approach which was employed to address the gaps and loopholes that were revealed in the literature review. In particular, the design science research methodology was chosen as the model employed in this endeavour.

Chapter 5 details the model developed to address Research Objective 1 which incorporates identity management, smart contracts, blockchain, and machine learning techniques. This chapter explains how these components function and it also provides details on the experiments that were conducted.

Chapter 6 presents the EWS model that was built to alert users when their identities are about to expire. This chapter provides a detailed explanation of the intelligent system, along with modelling the algorithm that will be used to determine the expiration date of users' identities based on the data stored on the smart contract.

Chapter 7 presents the model that was developed to compute the trustworthiness value of the user based on the provided identities, which are stored in the smart contract. This model can be used to determine whether the user is trustworthy or untrustworthy based on single or multiple documents. Furthermore, blockchain ensures the immutability of the recorded values, which cannot be modified once it has been recorded.

Chapter 8 explains how the prototypes that were constructed to address the research questions posed in this research function. This explanation is conducted using screenshots and is accompanied by appropriate and sufficient demonstrations.

Chapter 9 concludes the thesis by providing an overview of what has been accomplished and what further steps might be taken in the future to broaden the research scope.

1.7 Conclusion

Blockchain technology is an emerging technology that provides a variety of features, including building trust between users and service providers and ensuring privacy for users. Users' identities should be managed in a trustworthy manner by leveraging the decentralization nature of blockchain technology. The objective of the thesis is to build solutions that address numerous significant weaknesses in the current literature. In this section, we presented the issue that is investigated in this thesis. The intelligent models developed for identity management aim to ensure users' trust and the privacy of their identities using blockchain technology and smart contracts. We stated the research problem and we discussed the challenges associated with the present identity management processes. The chapter also discussed the scientific and social contributions that will be made as a result of this research. An overview of the following chapters is also provided.

We conducted a systematic literature review of the relevant studies in the following

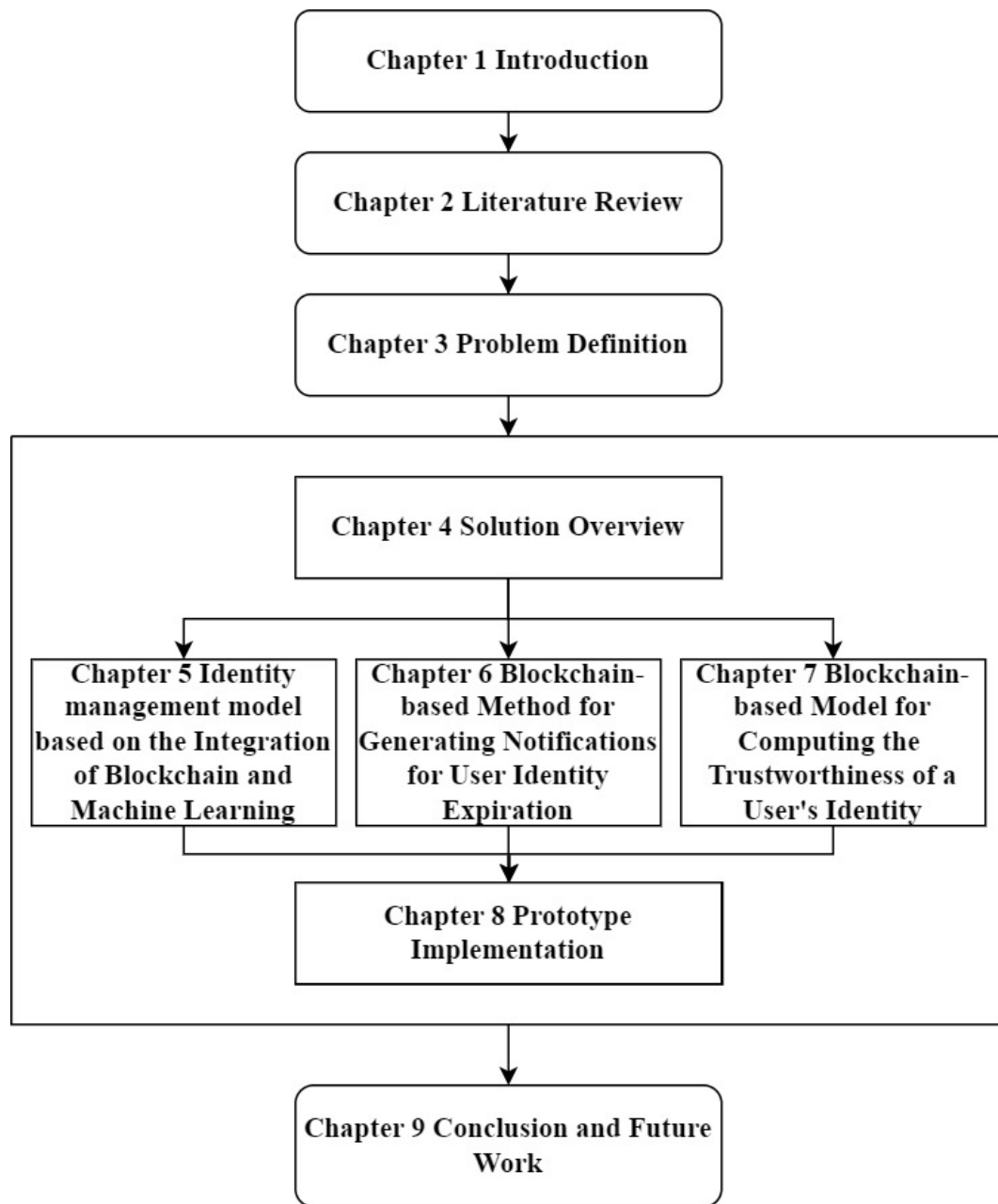


Figure 1.1 : Structure of the Thesis

chapter. The literature review confirms that the problem this research investigates has not been addressed in prior research.

Chapter 2

Literature Review

This chapter presents a systematic literature review of the existing research on the use of blockchain technology in the realm of identity management.

This chapter comprises sections that have previously been published in (Alharbi and Hussain, 2022).

2.1 Introduction

Due to the widespread use of the Internet, most traditional services such as e-commerce and e-banking have moved to an online platform. Users must verify their identity to access online application services using documents such as a passport, driver's license, birth certificate, etc. The term identity management refers to the process of identifying, authenticating, and authorising an entity to access resources (Zhu and Badr, 2018). The online applications store the users' personal information in a centralized fashion when users share their information to access services. This information is controlled by central authorities who access user data without the need for the user's consent. Thus, users have no control over this information. Centralised identity management systems which deprive users of the ownership of their identity is a major concern (Alharbi and Hussain, 2021). Hence, a central authority may exploit users' trust, posing significant privacy and security concerns. This raises concerns about identity theft that necessitates robust identity management solutions. Thus, it is imperative to address these issues of identity. Efforts are underway to decentralize identity management to alleviate the aforementioned concerns.

The advent of blockchain technology is paving the way for new opportunities for resolving critical data privacy, security, and integrity challenges in identity management (Ren et al., 2019). Over the last few years, blockchain technology has garnered substantial attention from both industry and academia. The recent introduction of blockchain and smart contracts as extensions of distributed ledger technology are redefining business models and management in different use-cases including health-care, the Internet of Things (IoT), and smart cities. The technology is known for being a tamper-resistant and transparent ledger (Lu, 2019). Thus, it can be utilized to link users' claims to their identities, thereby preventing identity fraud in identity management. Furthermore, blockchain is characterized by attractive features such as immutability, decentralized nature, and traceability, making it ideal for use in the identity management field. Blockchain technology has enormous potential in the realm of identity management due to its advantages such as decentralization, tamper-resistance, and transparency (He et al., 2015).

The main purpose of blockchain technology is to remove reliance on a third party, leading to direct communication between users and stakeholders; therefore, blockchain technology has emerged to establish trust between parties (Toth and Anderson-Priddy, 2019). The intrinsic features of blockchain make it possible for both parties to communicate with each other in a trustworthy and secure way, without the need to disclose sensitive information. Blockchain technology can ensure secure and trustworthy data exchange between users and stakeholders based on its immutability and anonymity features (Casino et al., 2019). Blockchain can bring great value to identity management by giving identity ownership to users. It has become necessary to integrate identity management systems into one single system for all stakeholders to achieve transparency, security, and immutability.

There are multiple studies in the literature which discuss the use of blockchain technology in various fields, such as financial markets, the Internet of Things (IoT),

smart homes, and healthcare (Polyviou et al., 2019), (Andoni et al., 2019), (Moniruzaman et al., 2020), (Hölbl et al., 2018). Several review articles have been published on blockchain applications in the identity management domain (Kuperberg, 2019), (Lim et al., 2018), (Ahmed et al., 2022), (Hariharasudan and Quraishi, 2022). However, no systematic literature review of recent research on blockchain-based identity management applications has been conducted and no papers that address blockchain technology in identity management applications in a systematic manner have been published. We aim to overcome this shortcoming by providing a technical background in blockchain-based identity management applications which highlight recent developments in the field. This study examines the recent research on blockchain-based identity management and examines its strengths and shortcomings. The study also highlights the existing identity management research challenges.

This chapter provides a systematic review of the state-of-the-art in the realm of identity management using blockchain technology and systematically categorizes blockchain-related research publications. The purpose of this study is to demonstrate the potential use of blockchain in identity management and highlight the obstacles and prospective directions of blockchain technology. We conducted a systematic literature review (SLR) on blockchain to provide valuable insights.

The contributions of this study are as follows:

- It provides a brief review and analysis of identity management and blockchain technology.
- It discusses the benefits and drawbacks of the currently used blockchain-based identity management solutions.
- It investigates several identity management solutions based on blockchain technology.
- It analyses identity management solutions using blockchain technology based

on a variety of factors.

- It examines the primary challenges associated with identity management solutions in the context of blockchain technology.
- It analyses the SLR's findings and makes recommendations for future research.

The rest of this chapter is organized as follows: Section 2.2 briefly reviews identity management and provides an overview of blockchain technology. Section 2.3 outlines the research methodology followed in this thesis. The study's results and a discussion of the existing work are given in section 2.4. A comparative analysis and the shortcomings of the existing research are presented in sections 2.5 and 2.6. Finally, section 2.7 concludes the chapter.

2.2 Preliminaries

This section provides the essential background for understanding the remainder of the chapter, including identity management and blockchain.

2.2.1 Identity management

Identity management is a mechanism by which participants are validated, recognized, and authorized to access sensitive data (Domingo and Enríquez, 2018). In the literature, identity management is often referred to as identity and access management. Identity management systems comprise three main components: a user, a service provider, and an identity provider. These three parties are interconnected entities: the user requests a service from the service provider, and the identity provider is tasked with validating the user's identity via the authentication protocol. The traditional identity management approaches are effective for service providers but ineffective for users, as they must remember numerous passwords to access various

websites. In the literature (Ahn and Ko, 2007), (Birrell and Schneider, 2013), (Satybaldy et al., 2019), identity management can be classified into four main groups: the isolated model, federated model, user-centric model, and centralized model.

Nonetheless, the centralized approach has been the conventional approach to keep personal data. The key difficulty lies in guaranteeing that legitimate users retain control over their identity details online when using a centralized approach for identity management (Ferdous and Poet, 2012), (Kumar and Bhardwaj, 2018). Such an architecture enables attackers to penetrate these systems and access user information. Furthermore, a third party may violate the user's trust which elevates privacy issues. To overcome the concerns of centralized databases pertinent to privacy issues, a decentralized identity management approach has been developed to ensure the system is robust. The emergence of the new technology, known as blockchain, helps users to use the internet without relying on a trusted third party (El Haddouti and El Kettani, 2019). The major advantage of blockchain is its decentralized structure since all the network nodes are retained in the entire database.

2.2.2 Blockchain technology

2.2.2.1 Overview of blockchain

Blockchain was created by Nakamoto in 2008 (Nakamoto, 2008) and is a collection of interconnected blocks that store all transaction records. Blockchain works by storing data in distributed ledgers that are disseminated across all computing devices in a decentralized fashion. The blockchain structure is composed of a sequence of blocks. The block comprises two major sections, the block header and the block body. The block header contains a block version, Merkle tree, timestamp, and parent block hash. The block body consists of a transaction counter and transactions. Each block contains the prior block hash in the block header, thus it can be linked

to one parent block only. This creates a connection between the blocks, resulting in the formation of a chain of blocks. The series of hash operations forms an immutable chain that can be traced back to the first block produced. The genesis block is the first block on a blockchain and it does not have a parent block. The majority of the network's participants must reach a consensus and confirm each transaction before it can be recorded in the public ledger. Data cannot be altered or deleted once it has been entered.

Blockchain has particular features that make it attractive as a decentralized technology due to the fact that the ledger is not controlled by a central authority. The following are some of these features:

- **Decentralization:** This is the essence of blockchain technology, as each node maintains a record of all transactions, thereby eliminating the need for a central authority. A central trusted organization should validate transactions causing performance bottlenecks at the central servers. Unlike centralized systems, blockchain eliminates reliance on a third party (Zheng et al., 2018).
- **Transparency:** Records are shared among all the participants in the blockchain. Each participant in the network has the same obligations and permissions to access permitted information (Atlam and Wills, 2019).
- **Traceability:** The blockchain employs timestamps to identify and record each transaction, thereby reinforcing the data's time dimension. This enables the participant to maintain transaction order and to make the data traceable. Thus, every transaction can be traced back to a certain time, making it easier for participants to identify the parties involved (Omar and Basir, 2018).
- **Trust:** Data exchange between participants in the network does not require mutual trust between participants because blockchain is deployed in a decen-

tralized manner (Christidis and Devetsikiotis, 2016). Therefore, trust shifts from a third party to the technology itself.

- **Immutability:** This feature ensures that any confirmed transaction cannot be tampered with. Hence, the data is unaltered after being stored on the blockchain.
- **Anonymity:** Every user on the blockchain has the ability to interact with an established address. The system will not reveal the user's true information; nonetheless, participants will be able to access the encrypted transaction information.

2.2.2.2 Smart contract

The notion of a smart contract was first introduced by Szabo (Szabo, 1997). A smart contract is a computer program that is not executed until the relevant data or action is received (Lu, 2018). A smart contract is a form of electronic agreement of a legal contract between parties to the transaction. The goal of a smart contract is to eliminate the need for a trusted intermediary. A smart contract includes execution rules and execution logic. When the rule is satisfied, the execution logic is performed automatically. Data is only released by a smart contract when certain rules are satisfied. The availability of a smart contract in blockchain builds trust among participants and automatically removes the need for a trusted third party. Ethereum is the most popular blockchain platform for smart contracts.

2.3 Systematic Literature Review (SLR) Methodology

The primary objective of this study is to examine the existing literature in the field of identity management within the framework of blockchain technology and to identify critical research gaps that require further investigation in future studies. We

survey the existing literature to identify the relevant issues, challenges, and solutions in relation to blockchain-based identity management. We conducted an SLR to accomplish this goal using the procedure outlined in (Kitchenham et al., 2010). The SLR is an organized and systematic approach to defining, synthesizing, and selecting recent literature related to the research objectives. This research comprises citation and evaluation procedures to complement the basic SLR approach to ensure the quality of the literature review.

The systematic approach involves the following steps (Kitchenham et al., 2010):

1. Data source selection and search strategies.
2. Inclusion and exclusion criteria.
3. Citation and inclusion decision management.
4. Final selection and quality assessment.
5. Data extraction and synthesis.

2.3.1 Data source selection and search strategies

Many sources have been explored to obtain an unbiased and comprehensive perspective, including the main online databases. The following popular scientific databases were used as source for this literature review:

1. IEEE Xplore (<https://www.ieeexplore.ieee.org>)
2. Elsevier ScienceDirect (<https://www.sciencedirect.com>)
3. SpringerLink (<https://link.springer.com>)
4. ACM Digital Library (<https://dl.acm.org>)
5. Google Scholar (<https://scholar.google.com>)

These well-known scientific databases were chosen because they cover the related literature. The papers reviewed were chosen from industry papers, qualitative and quantitative studies, and scientific academic studies. Figure 2.1 shows the review process at each stage and the number of papers identified. We used the Boolean operator "AND" to search for relevant research using various combinations of items from all of the search terms. The "OR" operator is used to connect similar terms to ensure maximum coverage. The search statement is split into two major sections. The first sub-section is composed of a collection of blockchain-related phrases. The second sub-section contains a collection of phrases related to identity management. As a result, the following search string is produced:

("blockchain" OR "distributed ledger technology" OR "smart contract") AND ("identity" OR "identity management").

2.3.2 Inclusion and exclusion criteria

We include certain studies that are pertinent to blockchain-based identity management and its applications that meet a certain criterion. The following factors were taken into consideration when deciding whether to include or exclude a study:

1. The paper must be relevant to the topic of blockchain and identity management.
2. The study was conducted between 2017 and 2022.
3. The paper is written in English and the full content is available.
4. The article must have undergone a peer review process.
5. The paper must include empirical evidence relating to the use of blockchain technology for identity management.

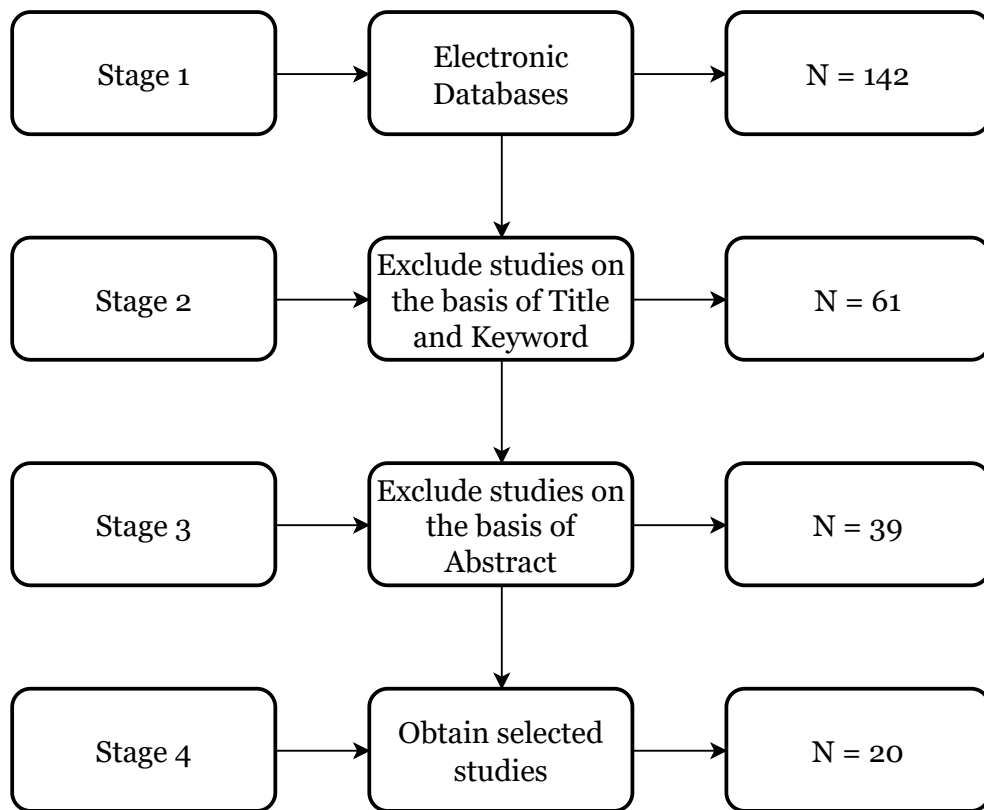


Figure 2.1 : Literature Review Process

Therefore, the study excludes papers that either do not focus on blockchain and identity management or meet the following exclusion criteria:

1. Duplicate studies.
2. Studies are not written in English.

2.3.3 Citation and inclusion decision management

At this stage, all 142 papers were exported to and stored in EndNote, where we reviewed them using the search terms in either the title or the keywords. A paper was selected if it contained at least two search terms; one from each section, in either the title or the list of keywords; otherwise, the paper was not selected for

the next filtration stage. The total number of selected papers was reduced to 61 by conducting this filtering process.

2.3.4 Final selection and quality assessment

All abstracts were thoroughly examined to ensure their relevance before the paper was included in the final stage. The articles with pertinent abstracts were selected to go through to the next filtration stage; otherwise, the paper was excluded. The total number of selected articles was reduced to 39 by carrying out this filtration process. Table 2.1 describes the scientific assessment of the filtration process.

Filtration stage	Method	Assessment criteria
First Filtration	Search keywords from scientific databases	Search terms
Second Filtration	Exclude studies on the basis of titles	Title = search term Include else exclude
Third filtration	Exclude studies on the basis of abstracts	Abstract = relevant Include else exclude
Final Filtration	Obtain selected papers and critically appraise studies	Discusses Data relevant Yes = accepted No = rejected

Table 2.1 : Scientific Assessment Process

2.3.5 Data extraction and synthesis

At this stage, the 39 papers were analyzed and their quality was ensured according to the quality criteria suggested by (Dybå and Dingsøy, 2008) as listed in Table

2.2. The most relevant papers were selected after careful consideration.

An additional 19 articles were excluded at this stage by applying the criteria in Table 2.2, leaving 20 articles for the final data review and synthesis to address the objectives of the research. The final 20 selected articles were evaluated on the basis of the quality criteria listed in Table 2.2.

Quality Criteria	
1	Is the paper research based?
2	Are the aims of the research clearly stated?
3	Is the context adequately described?
4	Is the design framework appropriate to address the aims of the research?
5	Is the data analysis sufficiently rigorous?
6	Is there clear evidence for the findings?
7	Is the study validated or implemented?

Table 2.2 : Quality Criteria

2.4 Results and Discussion of Existing Work

The literature review reveals various technical challenges in the current blockchain-based identity management systems. We grouped the papers according to the challenge each study is attempting to overcome. Thus, the papers are classified into three subcategories for better presentation and to identify the natural affinity between them. This will enable us to organize the literature by grouping the papers with related themes. The classification is based on the remaining reviewed articles after the filtering processes. We divided the studies into three categories: (1) authentication, (2) privacy, and (3) trust. The taxonomy of the technical issues that blockchain encounters in the identity management sector is summarized in Table

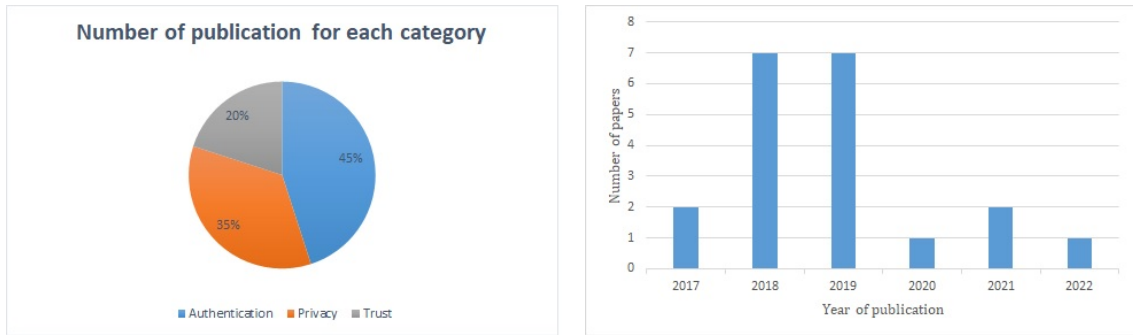
2.3. Figure 2.2a shows the percentage of the total number of publications in each category. We found that most of the publications addressed the authentication issue. The number of studies on blockchain-based identity management published per year between 2017 and 2022 is shown in Figure 2.2b. According to our findings, more pertinent articles were published after the year 2017, emphasizing the novelty of the topic at hand. We observe that most of the papers were published in the years 2018 and 2019. 20% of the articles selected for review were published in 2020. Of these challenges, the following stand out:

Category	Papers
Authentication	(Juan et al., 2018), (Liu et al., 2019), (Zhou et al., 2019), (Odelu, 2019), (Othman and Callahan, 2018), (Xu et al., 2020), (Chen et al., 2021), (Lee, 2017), (Jamal et al., 2019)
Privacy	(Chalaemwongwan and Kurutach, 2018), (Rathee and Singh, 2022), (Mudliar et al., 2018), (Saldamli et al., 2019), (Faber et al., 2019), (Alsayed Kassem et al., 2019), (Rathee and Singh, 2021)
Trust	(Stokkink and Pouwelse, 2018), (Buccafurri et al., 2018), (Hammudoglu et al., 2017), (Takemiya and Vanieiev, 2018)

Table 2.3 : Mapping of categories to respective publications

2.4.1 Authentication

As a decentralised distributed ledger, blockchain technology can act as a trustworthy decentralised authentication infrastructure. A number of research studies have been conducted on blockchain-based identity management for authentication. (Juan et al., 2018) presented an authentication model for a national electronic identity document based on blockchain. They discussed ways to address the security



(a) Number of publications by category

(b) Number of publications by year

Figure 2.2 : Number of publications by category and year

issues that are encountered in Colombia's current national identity document, such as the protection of citizens' information and the prevention of fraudulent transactions. Such issues can be addressed by integrating blockchain with biometric authentication technology using smart cards and leveraging the benefits of established authentication methods such as biometrics and physical security to mitigate the security concerns associated with identification documents. Hence, this helps the government to verify a document and identify counterfeit documents.

(Liu et al., 2019) presented a model to preserve privacy to manage identity by integrating biometrics and blockchain. A government-specified body gathers and stores the user's identity in the interplanetary file system (IPFS). A smart contract on Ethereum governs the system's access control. The system's primary objective is to enhance identity management while providing data security using smart contracts. However, user registration at entry points is required to safeguard the system against any data breaches.

(Zhou et al., 2019) presented a self-sovereign digital identity management framework (EverSSDI) based on IPFS and smart contracts to develop a framework for decentralized identity management. Users encrypt and maintain their personal data in the IPFS system utilizing data hash fingerprints which are verified through a smart

contract. Therefore, the user becomes the real and dominant owner of the identity instead of merely proving their digital identity. However, in the authorization procedure, users must supply identity attributes to the service providers.

(Odelu, 2019) presented a novel biometrics-based authentication approach in which the user's identity is managed via a blockchain. The author conducted a thorough security study of the protocol, proving it is resistant to known attacks. However, the technique does not guarantee user anonymity or untraceability.

(Othman and Callahan, 2018) proposed the Horcrux protocol, a secure decentralized authentication method which allows the end-users of a self-sovereign identity to have control over accessing their identities through a biometric authentication that is capable of ensuring the privacy of the user. The protocol relies on decentralized identifiers and it is based on the concept of self-sovereign identity. They implemented a decentralized biometric credential storage mechanism using a blockchain to store decentralized identifiers.

(Xu et al., 2020) developed a blockchain-based identity management and authentication mechanism based on the redactable blockchain for mobile networks, where users retain ownership over their identifying information. The blockchain stores legitimate users' self-sovereign identities and public keys, and the chameleon hash is utilized to remove unlawful users' data while leaving the block head unaltered.

(Chen et al., 2021) presented a decentralized identity management system and a cross-domain authentication method based on blockchain with the objective of eliminating the authentication center's single point of failure and increasing the cross-domain authentication performance. The uniqueness of an identifier is determined by the consensus mechanism of the consortium blockchain and anyone can request identifiers. The system uses a one-way accumulator to ensure the validity of the entity identity.

(Lee, 2017) introduced a blockchain-based solution for managing identity and au-

thentication for mobile users and IoT devices. Their proposed approach is to generate and maintain blockchain identities as a service, without regard for interactions or messages via the blockchain. The blockchain-based identities are only intended to be used for decentralized authentication in this scenario. Thus, authentication can be accomplished without having any preregistered users' information.

(Jamal et al., 2019) presented a blockchain-based identity system for storing personal information. This solution makes use of blockchain features to ensure that users are aware of who has access to their personal data. The system enables third parties to access personal records while maintaining their immutability.

2.4.2 Privacy

Privacy is a major concern that is still being researched. Several privacy-preserving techniques have been discussed in the literature. For example, (Chalaemwongwan and Kurutach, 2018) developed a national digital ID framework based on blockchain (NIDBC) to assist in enhancing a digital identity to a single sign-on for government services. Moreover, they affirmed that privacy is preserved by allowing users to control their data by granting permission for services to access their personal information. Furthermore, due to the inherent nature of blockchain, the system is secure since the data is distributed, which makes it difficult for attackers to attack data. However, the service provider is still able to abuse users' information.

(Rathee and Singh, 2022) proposed a blockchain-based self-sovereign identity management system. IPFS maintains users' data whereas blockchain maintains the content address of their data and the public key. Smart contracts that operate on the blockchain perform the verification process. Third parties are not permitted to access the data directly, which is exclusively accessible to the user. Data privacy is thus ensured in this manner.

(Mudliar et al., 2018) presented a model that utilizes blockchain technology to enable people to carry their national identity on their phones. Government employees can verify a citizen's national identity by scanning a barcode or QR code generated automatically through the government site. The major benefit of this approach is that communication between the government and the citizens is transparent.

(Saldamli et al., 2019) designed a system that uses Ethereum-based blockchain to store an identity hash on blockchain. A data hash is created by IPFS and the corresponding hash is stored on Ethereum blockchain. Documents are approved for verification of the identity by the user once the third party requests it. The user has the right to share the document or reject it. Thus, personal data can be completely controlled by the user with this system.

(Faber et al., 2019) proposed a conceptual design for a blockchain-based personal data and identity management system that is human-centric and General Data Protection Regulation (GDPR) compliant. They presented a framework that is transparent and gives data owners complete control over how their data is used. However, this study is still conceptual and does not provide any technical specifics or performance evaluation.

(Alsayed Kassem et al., 2019) presented blockchain-based identity management as a means of securing personal data sharing across networks, and they emphasized the importance of the blockchain and decentralized self-sovereign identities. Moreover, the system allows users to retain their identities linked to specific attributes that can be used by service providers to authenticate the user and provide their services based on verified attributes. They attempted to leverage blockchain and its characteristics as the backbone of identity management across all the realms. The result of the security analysis demonstrated that it is possible to develop a secure and resilient identity management system that can overcome the drawbacks associated with centralized identity management systems.

(Rathee and Singh, 2021) developed a blockchain identity management technique based on the Merkle hash digests algorithm (MHDA) that allows data sharing without being compromised by anonymous users. MHDA-based BIdM systems are efficient in terms of allowing users to maintain control over their identities and credentials. The blockchain's feature of intervening conflict ensures the integrity of the user's data.

2.4.3 Trust

Trust is critical when developing identity management systems. Several techniques have been proposed to provide identity in the context of mutual distrust. For example, (Stokkink and Pouwelse, 2018) introduced a digital identity model based on blockchain which builds on a generic provable claim model using zero-knowledge proofs and the collection of third-party attestations of trust are required. The work focused on a self-sovereign identity for the Netherlands and was part of an undertaking by the government that provides identity within the context of a mutual solution. They assert that their systems are suitable for general use, however, it is not shown how the work integrates with existing IT applications.

(Buccafurri et al., 2018) suggested an architecture to integrate blockchain technology with identity management via identity-based encryption to achieve the trust level between users. They created a non-anonymous blockchain by binding a digital identity with a public key, which can be used to define the author of the transaction.

(Hammudoglu et al., 2017) developed a biometric-based authentication mechanism and blockchain storage. This enables the user to maintain personal information securely, which can be accessed upon successful biometric authentication. It combines a permissionless blockchain with identity and key attestation capabilities for use with mobile phones. However, a fully accessible blockchain is used to store unen-

encrypted fingerprints, which compromises both security and privacy.

(Takemiya and Vanieiev, 2018) proposed a mobile application-based identity system that leverages blockchain technology to establish a secure protocol for storing encrypted personal data and sharing verified claims about personal data. The hash values of a user's personal information that has been encrypted using a cryptographic key are broadcast on the blockchain in this system. It is developed on top of the permissioned hyperledger "Iroha blockchain". The Sora mobile apps enable users to produce a pair of encryption keys, insert and encrypt their data, and propagate salted hashes to the blockchain. After this, users have the option of voluntarily providing sensitive information to third parties such as institutions. The drawback is that the system cannot achieve complete decentralization if the keys are stored centrally.

2.5 Comparative analysis of the existing research

Based on the above comprehensive systematic literature review, we identified three gaps and challenges in the work on managing user identity, based on blockchain using artificial intelligence. The comparative analysis of the studies that touched on the identified issues and the decision to work in these fields is shown in Table 2.4 and Table 2.5

2.6 Shortcomings of the existing literature on identity management based on Blockchain

The literature review identified various significant challenges in using blockchain to improve user identity management. In light of the comparative analysis of the existing literature review on blockchain-based identity management shown in Tables 2.4 and 2.5, we highlight the most significant shortcomings as follows:

Core papers included in the literature	Is this paper relevant to blockchain and identity management?	Is this relevant to the analysis of problems and solutions in related fields?	Has the proposed framework been developed/validated?
(Juan et al., 2018)	Yes	Yes	No
(Liu et al., 2019)	Yes	Yes	Yes
(Zhou et al., 2019)	Yes	Yes	Yes
(Odelu, 2019)	Yes	Yes	Yes
(Othman and Callahan, 2018)	Yes	Yes	Yes
(Xu et al., 2020)	Yes	Yes	Yes
(Chen et al., 2021)	Yes	Yes	Yes
(Lee, 2017)	Yes	Yes	No
(Jamal et al., 2019)	Yes	Yes	Yes
(Chalaemwongwan and Kurutach, 2018)	Yes	Yes	Yes
(Rathee and Singh, 2022)	Yes	Yes	Yes
(Mudliar et al., 2018)	Yes	Yes	Yes
(Saldamli et al., 2019)	Yes	Yes	Yes
(Faber et al., 2019)	Yes	Yes	No
(Alsayed Kassem et al., 2019)	Yes	Yes	Yes
(Rathee and Singh, 2021)	Yes	Yes	Yes

Table 2.4 : Comparative analysis of the existing research on blockchain-based identity management in the literature

(Stokkink and Pouwelse, 2018)	Yes	Yes	Yes
(Buccafurri et al., 2018)	Yes	Yes	Yes
(Hammudoglu et al., 2017)	Yes	Yes	Yes
(Takemiya and Vanieiev, 2018)	Yes	Yes	Yes

Table 2.4 : Comparative analysis of the existing research on blockchain-based identity management in the literature (continued)

- None of the existing literature has integrated blockchain and identity management using artificial intelligence techniques to detect duplicate user identities on top of blockchain.
- None of the existing literature has developed a personalized early warning system to detect user identities that are about to expire to remind users to renew them and obtain a benefit from the desired services.
- None of the existing literature takes into account how to compute the trustworthiness of user's identity based on a single or multiple documents.

To address the aforementioned gaps in the research literature, in this thesis, we develop comprehensive artificial intelligence-driven solutions on top of blockchain.

2.7 Conclusion

Blockchain is a developing technology that has the potential to revolutionize the world of information technology, as an immutable ledger can be used in a wide variety of applications. In this study, a systematic literature review was carried out

Core papers included in the literature	Approaches for detecting duplicate identity management	Approaches for using alert systems for nearly expired identities	Approaches for computing the reliability score of an identity
(Juan et al., 2018)	No	No	No
(Liu et al., 2019)	No	No	No
(Zhou et al., 2019)	No	No	No
(Odelu, 2019)	No	No	No
(Othman and Callahan, 2018)	No	No	No
(Xu et al., 2020)	No	No	No
(Chen et al., 2021)	No	No	No
(Lee, 2017)	No	No	No
(Jamal et al., 2019)	No	No	No
(Chalaemwongwan and Kurutach, 2018)	No	No	No
(Rathee and Singh, 2022)	No	No	No

Table 2.5 : Existing research studies on identity management based on blockchain

(Mudliar et al., 2018)	No	No	No
(Saldamli et al., 2019)	No	No	No
(Faber et al., 2019)	No	No	No
(Alsayed Kassem et al., 2019)	No	No	No
(Rathee and Singh, 2021)	No	No	No
(Stokkink and Pouwelse, 2018)	No	No	No
(Buccafurri et al., 2018)	No	No	No
(Hammudoglu et al., 2017)	No	No	No
(Takemiya and Vanieiev, 2018)	No	No	No

Table 2.5 : Existing research studies on identity management based on blockchain (continued)

to examine the use of blockchain technology in the identity management domain, its challenges and future work. The study demonstrates that utilizing blockchain in identity management has the potential to overcome the limitations of traditional identity management systems. Blockchain research trends in identity management indicate that it is mostly utilized for authentication, data sharing and data ownership, but it is rarely employed for other purposes such as supply chain management. According to our findings, the effort to apply blockchain technology to identity management is accelerating. In addition, the use of blockchain ensures that identity ownership is controlled by legitimate users. However, certain challenges remain unresolved and need more investigation. Future research directions in identity man-

agement have been identified based on the findings, with an emphasis on resolving concerns about the use of blockchain technology in areas such as identity modification, key management, and the cost of blockchain technology.

Chapter 3

Problem Definition

3.1 Introduction

This chapter outlines the research questions based on the systematic literature review that was reported in the previous chapter. These questions contribute to defining the research objectives, which are also outlined in this chapter.

3.2 Gaps in the Literature

We found the following gaps based on the systematic review of the existing literature, as detailed in Chapter 2:

1. None of the existing literature integrated blockchain and identity management using artificial intelligence techniques to classify data to detect duplicate user identities and to maintain privacy of these identities on top of blockchain.
2. None of the existing literature has developed a personalized early warning system to detect user identities that are about to expire to remind users to renew them and obtain a benefit from the desired services.
3. None of the existing literature takes into account how to compute the trustworthiness of a user's identity based on a single or multiple documents.

3.3 Definitions of Key Concepts and Terms

Definitions of the concepts and terms that are utilized throughout this thesis are presented in this section.

3.3.1 Blockchain

Blockchain does not rely on an intermediary to validate of transactions, thus a blockchain is a decentralised ledger. A blockchain comprises of a series of blocks and each block consists of a hash of the previous block, forming a chain of blocks (Casino et al., 2019).

3.3.2 Smart Contracts

Smart contracts are a protocol that are designed to make the negotiation and execution of a contract more convenient for all parties. They enable the tracking and execution of complicated agreements between parties without the need for human intervention (Toth and Anderson-Priddy, 2019).

3.3.3 Identity Management

Identity management involves the measures used to verify a user to manage access to services offered by various industries, including banking, finance, healthcare, government, and online commerce (Dorri et al., 2017).

3.3.4 Ethereum

Ethereum is a distributed and open source platform. It provides a blockchain solution to construct a distributed application on top of blockchain (Peter and Moser, 2017). The feature that distinguishes Ethereum is that it connects smart contracts and blockchain. Users can create their own code on top of the Ethereum platform, allowing for the creation of customised applications. Ether is a cryptocurrency employed by Ethereum to make payments of blockchain transactions. Each Ethereum user is individually recognised by an Ethereum address.

3.3.5 Trustworthiness of User Identity

The trustworthiness of a given user's identity is defined as a quantitative score that expresses the reliability of the user's identity. In other words, it reflects the extent or degree to which the identity credentials claimed by the user are genuine.

3.3.6 Service Provider

The service provider is defined as the organization that supplies the service customer with certain services.

3.3.7 Duplicate Detection

Duplicate detection is the task of finding instances that indicate similar entities in the real-world from many sources. Data Matching, Entity Resolution, and Record linkage are other terms for duplicate detection (Andoni et al., 2019)..

3.3.8 IPFS

The InterPlanetary File System (IPFS) is a decentralized file storage mechanism that operates on a peer-to-peer system (Banerjee et al., 2018). It employs a distributed hash table (DHT) to locate a file network. A content-addressed hash of the files is produced locally by the IPFS to ensure that the hash is available in the network as needed. A unique hash is assigned to each file stored in IPFS.

3.3.9 MetaMask

MetaMask is a browser extension that enables users to connect to the distributed web. Rather than deploying the entire Ethereum node, it allows users to execute Ethereum decentralised applications in their web browsers.

3.3.10 Early Warning System

An early warning system (EWS) is a mechanism for producing and disseminating useful alerts in a timely manner to help users take the necessary actions (Chaves and De Cola, 2017). It can be used to warn users when their identities are about to expire.

3.3.11 Machine Learning

Machine learning is an area of artificial intelligence which focuses on data application and algorithms to simulate the way in which people acquire knowledge (Mahesh, 2020). Complicated tasks can now be accomplished with the help of machine learning, which mimics the way people approach problem-solving.

3.4 Research Questions

The systematic literature review and the shortcomings reported in Chapter 2 reveal that the existing literature on employing blockchain in identity management suffers from several gaps. To address these gaps, the main research question in this thesis is as follows:

**How can user identities be managed intelligently and efficiently
on blockchain?**

The main research question can be broken into four research questions:

Research Question 1:

How to develop an intelligent and efficient method to detect duplicate user identities on top of blockchain?

To achieve this objective, a model is built that combines smart contract Ethereum blockchain with identity management. Blockchain and identity management are integrated to detect duplicate user identities and to store and maintain the sensitive information of users. To identify duplicate user identities, a number of different

machine learning methods are applied. Then, we manage the information of these identities by employing both on-chain and off-chain approaches to maintain the privacy of users' data. The data is encrypted before being uploaded to IPFS. Once the data hash is generated by the IPFS, the hash is encrypted before being recorded on blockchain to ensure data privacy. Data privacy is a paramount concern when it comes to personal data. Hence, sensitive information should be stored in a manner that maintains its privacy. Since the data on blockchain is publicly available, users are concerned about their personal and sensitive information. Consequently, there is a need to develop a method to preserve a user's private data. To address this concern, we integrate blockchain, IPFS, and encryption approaches to preserve data privacy which allows us to manage a user's information in a reliable manner.

Research Question 2:

How to develop an intelligent approach to generate personalised alerts regarding user identities?

The user's identity is stored on blockchain along with essential information including the expiration date. We develop a personalized early warning approach to alert users when their identities are about to expire, based on the data stored on blockchain. We developed a method whereby the user will receive reminders by email to take action. The method is based on pre-determining a threshold value which allows users to receive reminders within a specified amount of time in advance. This is achieved through the development of a date-based early warning model utilizing blockchain and smart contracts.

Research Question 3:

How to compute the trustworthiness score of a user's identity based on partial identity documents?

This objective is addressed by developing a method to compute the user's trustworthiness score even if the user only provides partial documents. The user is not

required to provide all the required documents, thus this partial identity is helpful in many cases. If the user does not have all the required documents, this partial trustworthiness score enables users to obtain an identity without full trustworthiness. Currently, there is no notion of a trustworthiness score based on partial documents. The overall trustworthiness score will be stored on the Ethereum blockchain since it is a reliable network, providing service providers with sufficient confidence in the trustworthiness score of the users. A prominent characteristic of blockchain technology is that once the trustworthiness score of a user has been broadcast, it cannot be altered. Based on these scores, different service providers can use them as a basis for providing their services based on the level of trustworthiness of each user.

Research Question 4:

How to validate the proposed methods?

To achieve this objective, we develop a prototype by employing a suitable programming language for each research question to test and validate the methods that have been proposed. Furthermore, we utilize well-known evaluation metrics to validate the outcomes of objective 1 and develop a prototype for managing the results. In addition, the Ethereum Rospsten network will be used to test the smart contracts.

3.5 Research Objectives

The following are the research objectives of this thesis to address the aforementioned research questions:

Research Objective 1:

Develop an intelligent and efficient method to detect duplicate user identities on top of blockchain.

Research Objective 2:

Develop an intelligent approach to generate personalised alerts regarding user iden-

tities based on the data stored in the blockchain.

Research Objective 3:

Develop an approach to compute the trustworthiness score of a user's identity based on partial identity documents to help service providers provide services to users based on the trustworthiness score.

Research Objective 4:

Validate the developed approaches by implementing them as a prototype or proof of concept.

3.6 Conclusion

This chapter presented the research gaps related to blockchain-based identity management and the important concepts that are utilized in this thesis were discussed. It also presented the research questions that this thesis addresses and the research objectives that are pursued and fulfilled through a systematic research approach.

The following chapter discusses the research methodology and outlines of the proposed solutions. The methodology section demonstrates the steps needed to achieve the objectives.

Chapter 4

Research Methodology and Solution Overview

4.1 Introduction

This chapter discusses the research methodology that is utilized to address the gaps identified by the literature review. It also outlines the proposed solutions and describes how the research questions are addressed.

4.2 Solution Overview

This section overviews the methods proposed to integrate identity management and blockchain.

4.2.1 Solution Overview for RQ1

The proposed method integrates blockchain with identity management to identify duplicate identities and then preserve the privacy of these identities using blockchain-based smart contracts. The proposed method utilizes machine learning algorithms to detect identical identities in an intelligent manner. In addition, we employ blockchain technology to maintain the privacy of a user's information. The method comprises the following four phases as shown in Figure 4.1:

First phase: After importing the two datasets, both of which have problematic issues with the data, we undertake the data preprocessing step to ensure the machine learning algorithms become more accurate and faster, which contributes to successfully building a machine learning model. We apply several techniques to address these issues in an appropriate manner, such as handling the missing values,

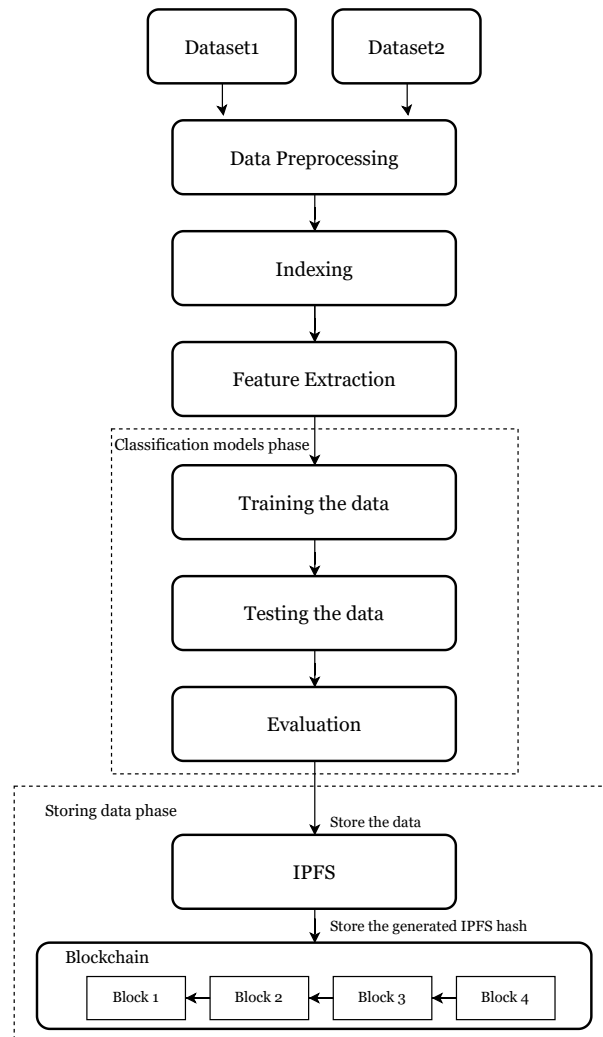


Figure 4.1 : Overview of the proposed method for duplicate identity detection

removing unwanted characters, tokenization, lemmatization, and feature scaling.

Second phase: We apply the indexing approach to divide the datasets into smaller blocks. To calculate the similarities between two instances, we need to compare every instance from one dataset with every instance in the other dataset. This could result in an extremely large number of instances. Therefore, to reduce the execution time, we apply the indexing technique to reduce the number of comparisons.

Third phase: We apply the classification method which includes training the data and testing the classifiers. We apply several machine learning algorithms for classifi-

cation to identify duplicate records, namely XGBoost, support vector machine with four different kernels (linear, RBF, Sigmoid, and polynomial), decision tree, random forest, deep neural networks, and K-nearest neighbors.

Fourth phase: We develop a data storage mechanism using off-chain storage IPFS and on-chain storage blockchain Ethereum to provide an efficient mechanism for storing the data. IPFS stores the data to avoid keeping a huge amount of information in the blockchain and then the blockchain keeps the generated IPFS hash to maintain the privacy of the user's data. The data is encrypted before being stored in IPFS and also before being stored in blockchain to provide an additional level of privacy for the data.

4.2.2 Solution Overview for RQ2

An early warning system (EWS) is designed to generate alerts and to make suggestions before action is needed (Berg et al., 2005). A warning system is utilized to provide users with alerts that inform them when the expiry date of the user's identity is imminent. This model is called the Early Warning Model (EWM) and it is designed to alert users when their identities have almost expired, based on a proactive approach and proactive action to renew their identities. The users receive alerts to take action. The system detects all the user identities stored in the blockchain which has certain information attached, such as the expiry date of the user's identity and the notifications to be sent to the user when necessary. This process is undertaken through the extraction of user identity information from the blockchain layer; thus the identity information is checked to send alerts to users if these identities expire based on the predefined value entered within a specific duration. The users set their preferred time frame for the alerts to be generated. These values produce notifications that differ from user to user to notify users in a sufficient time when their identity will expire to ensure that their identities are renewed

at the appropriate time.

The EWM informs the user of the impending expiry of their identity when a threshold value has been reached if the expiry time is less than the predetermined threshold value. The EWM compares the expiry date with the current date. The user sets the predefined threshold value and receives an alert by email.

Figure 4.2 depicts the three stages of the proposed EWM :

Stage 1: The EWM is triggered by the user or service provider and specifies a predetermined value.

Stage 2: Blockchain uses the EWM algorithm to find identities using the predetermined value.

Stage 3: The EWM generates alerts and sends them to the user.

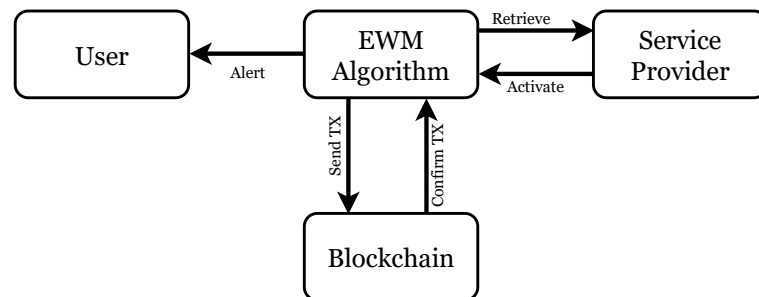


Figure 4.2 : Early Warning Model

4.2.3 Solution Overview for RQ3

The basic relationship between service providers and users is trust-based relationships. The providers need to believe that the user with whom they are communicating is who they think they are (Mayadunna and Rupasinghe, 2018). We selected several identity documents, such as passports, birth certificates, etc. to be used in the development of the model. Each identity document has a specific weight

following the personal identification system adopted by the Australian Government (Australian Government, 2018). The user's score is a numerical score on a scale of 0 - 100, where 100 denotes the highest possible trustworthiness and 0 denotes the lowest possible trustworthiness. The service provider can use the trustworthiness score to determine whether to proceed with the provision of a service. The trustworthiness score also allows service providers to make individual confidence decisions. Different service providers may assign different trustworthiness scores to a user's identity documents.

The method is based on assigning a certain weight for each user's identity document. Figure 4.3 illustrates the proposed model to compute the trustworthiness scores for users. The overall scores are according to the number of user identities provided by the user. If the user provides one identity, the user will obtain a specific score based on the defined weights. Moreover, in the case of a user providing more than one identity, the user can earn higher scores. The figure 4.3 shows that both the user and the service provider have a role in providing identity information. The user may submit their identity documents into the system. In this context, the smart contract calculates scores based on the information provided by the user or the service provider. The scores are computed by applying the predefined rules and calculations programmed into the smart contract. After the scores are calculated, they are stored on the blockchain. The blockchain is a distributed ledger technology that ensures secure and immutable storage of data. Storing the scores on the blockchain provides transparency and immutability, as the information cannot be easily altered or tampered with.

4.2.4 Solution Overview for RQ4

The validation of objective 1: We inspect several supervised machine learning algorithms to detect duplicates of user identities. Accordingly, a precise and

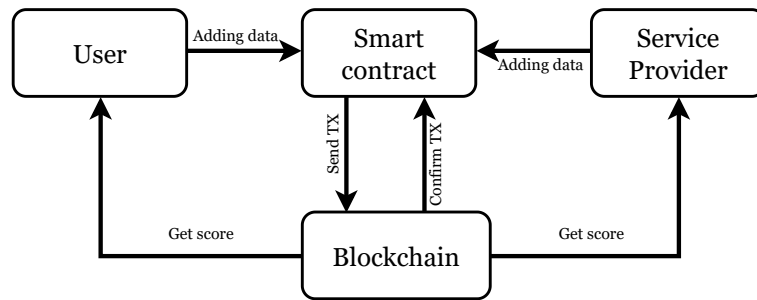


Figure 4.3 : The proposed model for user trustworthiness

appropriate model is selected.

Different quality measures (Christen, 2012) are adopted to evaluate the proposed model, namely precision equation (4.1), recall equation (4.2), and F-measure equation (4.3). In addition, we assess the average time taken to detect duplicates. The following equations are used to calculate these measures:

$$Precision = \frac{TP}{TP + FP} \quad (4.1)$$

$$Recall = \frac{TP}{TP + FN} \quad (4.2)$$

$$F - measure = \frac{2 \times Precision \times Recall}{Precision + Recall} \quad (4.3)$$

Furthermore, we build a prototype to store the data on blockchain. We use blockchain, smart contracts, IPFS, AES encryption, and React to develop the proposed method. Then, we evaluate the model's performance using Ganache blockchain as a personal blockchain (Suite, 2016).

The validation of objective 2: React and smart contracts are utilized to build the model and evaluate the model's performance for validation purposes. Then, the best performing model is selected using the following steps:

1. Developing the model using React and smart contracts.
2. Inserting the data.
3. Measuring the model's performance.
4. Choosing the best performing model.

The validation of objective 3: The model is built by combining React with smart contracts. The validation procedure is undertaken using the following steps:

1. Uploading the documents and specifying the document type.
2. Assigning the weighting score for each document depending on its type. This pertains to the document type that was previously specified.
3. Calculating the overall trustworthiness score of the user's identity based on the aggregate of these weighting values.

4.3 Research Methodology

In this study, we adopted the design science research methodology (DSRM) (Peffer et al., 2007) to fulfill the research objectives. We develop a prototype using DSRM, which is then tested to evaluate whether it achieves the study objectives. The process of research and development is repeated until the objectives are achieved. DSRM provides researchers with guidelines to conduct research on the basis of principles, practices, and procedures needed in design science. Figure 4.4 provides an overview of the steps of DSRM.

DSRM is utilized to divide the proposed method into the following six phases:

Phase 1: Identify the problem: we identify the research gaps in the existing literature in relation to managing the user's identity in blockchain using artificial intelligence.

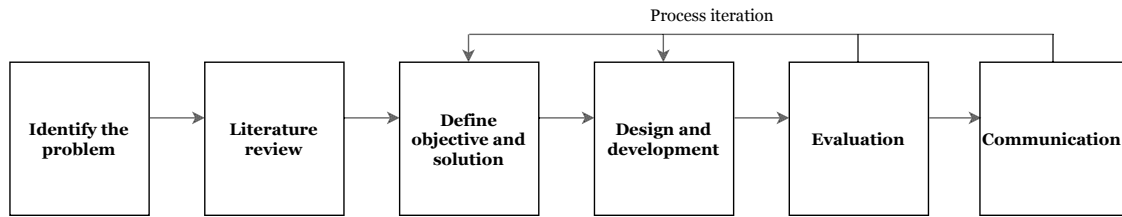


Figure 4.4 : Design science research methodology (Peffer et al., 2007)

Phase 2: Conduct the literature review: we identify the gaps in the existing state-of-the-art by conducting a critical review of these studies with respect to the research problem and moving the research forward.

Phase 3: Define the research objective and propose a solution: The main objective of this research is to develop a blockchain approach to manage user identity based on artificial intelligence. To achieve this aim, we propose a blockchain-based approach for identity management using machine learning methods to classify identities to address identity issues intelligently. The thrust of this research study is to develop intelligent and efficient methods of managing user identities on blockchain. Furthermore, we develop intelligent approaches to detect duplicate identities on top of blockchain.

Phase 4: Model design and development: In this stage, we develop the artificial intelligence models using both blockchain and machine learning techniques that correspond to the solution to research questions. The models developed are part of the overall user identity management methodology.

Phase 5: Evaluation and testing: We assess the performance of the developed models using several measures to answer research question 4.

Phase 6: Communication: the outcomes are submitted for publication in high-

ranked conferences and international peer-reviewed journals.

4.4 Conclusion

In this chapter, we presented the methodological approach adopted in this thesis to address the research objectives. The design science research technique was chosen as the model to be applied. We also provided a general overview of the proposed blockchain-based identity management methods and an overview of the solution for each of the research objectives.

The next chapter describes the blockchain-based model for detecting duplicate user identities on top of blockchain.

Chapter 5

Identity Management Model based on the integration of Blockchain and Machine Learning

5.1 Introduction

In this chapter, we discuss the process of developing a model which integrates blockchain and identity management. Machine learning is used to detect user identities, while blockchain-based smart contracts are employed to guarantee the privacy of identities. As a result of employing this approach, we are able to manage user identities in a reliable manner. The model development process is divided into four phases and each phase is discussed in detail in this chapter. Additionally, this chapter details the implementation of the proposed solution to research question one and the validation outcomes. We developed an identity management method based on integrating blockchain-based smart contracts and machine learning approaches.

Most existing identity management systems are centralized, hence preserving the privacy of users and their identity information is one of the most challenging aspects for organisations to ensure privacy (Ghaffari et al., 2022), (Alharbi and Hussain, 2021). Central authorities govern the management of identities in the current identity management systems, hence the user has no control over the privacy of their data. To overcome the concerns relating to the privacy of centralized systems, a decentralized identity management approach is needed to ensure the system is robust. The advent of blockchain enables users to use the Internet without having to rely on a central authority (Alharbi and Hussain, 2022), (El Haddouti and El Kettani, 2019). Blockchain removes reliance on a third party, as the system

is entirely decentralized. Blockchain records are immutable and irreversible, providing users with transparency. The unique characteristics of this technology such as immutability, decentralization, and traceability, make it an attractive solution for identity management. According to the evaluation framework proposed by (Lo et al., 2017), utilising blockchain for identity management is more suitable than employing traditional databases due to the inherent characteristics of this technology. When different data sources are combined, duplicate detection is one of the most crucial tasks. Duplicate detection is the procedure of determining whether two records represent the same entity in the physical world. The most recent findings for duplicate detection are obtained using supervised machine learning methods (Dong and Rekatsinas, 2018). In the existing literature, there is no comprehensive model that takes into account the problem of detecting duplicate identities and managing these identities once they have been identified in a reliable manner. Therefore, the model's primary focus is to employ machine learning to detect identity duplication, and then to utilise a blockchain-based smart contract to manage the resulting data. We conducted extensive experiments using several classification models and blockchain-based smart contracts.

Off-chain and on-chain mechanisms can have different impacts on machine learning outcomes depending on the context. Off-chain mechanisms refer to processes that occur outside the blockchain network. They typically involve pre-processing steps, data cleaning, feature engineering, model training, and evaluation. These off-chain mechanisms play a crucial role in shaping the machine learning outcome. On the other hand, on-chain mechanisms refer to executing machine learning processes directly on the blockchain network. This approach leverages the decentralized and transparent nature of blockchain technology to enhance machine learning outcomes in several ways, such as data privacy, transparency, and trust enhancement. However, on-chain mechanisms can also be slower and more expensive than off-chain

mechanisms, as they require all of the data and computations to be stored on the blockchain. The choice of whether to use off-chain or on-chain mechanisms for machine learning depends on the specific needs of the application. If performance is the most important factor, then off-chain mechanisms are the best choice.

In this chapter, we propose an intelligent approach to address the problem of detecting of duplicate identities on the top of blockchain. The work provides a comprehensive identity management blockchain-based model that combines machine learning methods with blockchain-based smart contracts to detect duplicate identities and manage user identities effectively. Furthermore, this work compares the effectiveness of various machine learning algorithms to examine the relationship between dataset size and the performance of various supervised machine learning algorithms. Moreover, incorporating blockchain and IPFS enables data to be managed and kept in a distributed fashion while maintaining the privacy of data. This is achieved by employing off-chain and on-chain storage methods that provide an immutable storage mechanism and guarantee data privacy by encrypting the data.

The rest of the chapter is organized as follows. Section 5.2 describes the proposed model and section 5.3 describes the experimental design of the evaluation. Section 5.4 describes the results of the individual experiments and discusses the main insights. Finally, Section 5.5 presents the conclusion.

This chapter is being reviewed by the International Journal of Web and Grid Services for publication (Alharbi et al., 2023).

5.2 The proposed model

This section discusses the model that was employed in this study for detecting and managing duplicate identities by utilizing machine learning and blockchain-based smart contracts. The main objective of the model is to detect identities that refer to the same-world identity and thus classify the records into matching or non-

matching, and to additionally manage the identities in such a manner to maintain the privacy of the personal data. The first step involves preprocessing the data sources to address any issues that may arise in the data. The second step is to implement the indexing technique to lessen the quadratic complexity of the data comparison process by avoiding the need to compare all records from the two data sources. The third step is to implement the feature extraction method to convert the text data into numerical data. The fourth step trains and tests the models and then evaluates these models. The final step is to store and manage the data on top of the blockchain. The proposed model is presented in Figure 5.1:

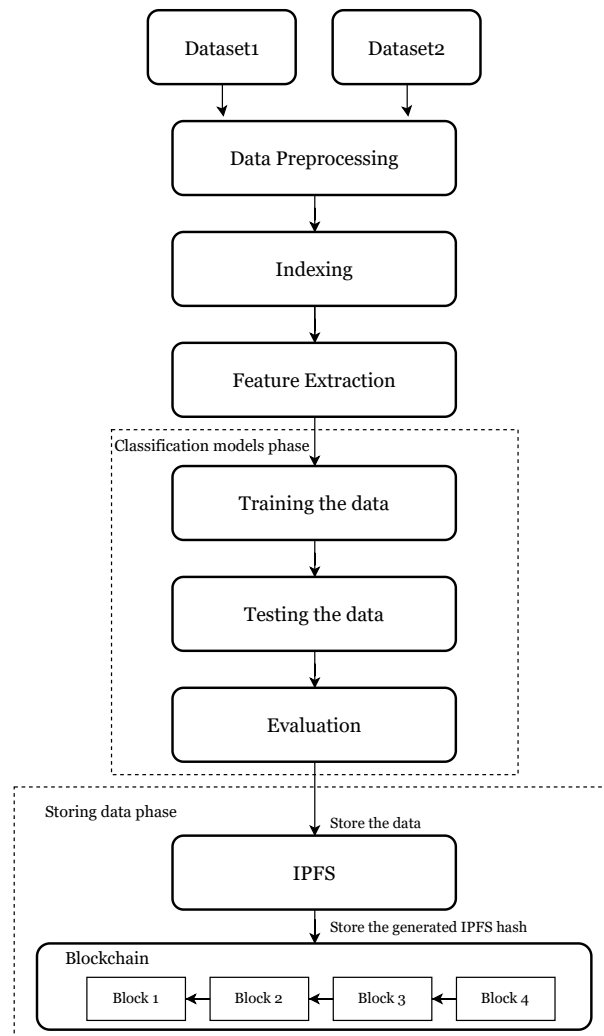


Figure 5.1 : Overview of the proposed method for duplicate identity detection

5.2.1 Preprocessing

Preprocessing entails cleaning the data of any unwanted noise to improve data quality, which is one of the most crucial phases in machine learning. The primary objective of this phase is to prepare the data in such a way that it is appropriate to construct the machine learning models. In most cases, there are numerous errors in the data. Hence, we apply the following steps:

- Removing punctuation: Removing set of symbols and additional special characters.
- Removing stop words: Stop words are the most frequently used words in a language which are meaningless such as a, an, the, etc., thus these words hold no significance in terms of differentiating between two documents.
- Lower casing: Shifting all words to lower case so input text is treated the same way.
- Lemmatization: This step breaks down all the tokens into their base form (lemma).

5.2.2 Indexing

It is challenging to compare records in a large dataset because of the huge number of records. Furthermore, comparing every record from one source with every record from another source requires complex computation which is the most expensive stage in the duplicate detection process. As a result, the indexing approach is applied to reduce the quadratic complexity of the matching process by avoiding comprehensive comparisons of all records. The indexing mechanism is more effective in the comparison process. In this stage, the data is split into blocks where each block contains records that are deemed to be a potential match. The split is according

to the blocking key which could be one attribute or a set of attributes. Records with the same value of the blocking key are grouped together in the same block. Thus, the comparisons will be executed between the record pairs that fall into the same block only, which improves the computational efficiency. Different indexing methods have been developed and the selection of the indexing technique depends on the data properties (Christen, 2011). In this study, we use a blocking technique to create the index blocks, where the keys from each record are placed into individual blocks. Thus, this technique ensures that only similar records are compared.

5.2.3 Feature extraction

In this phase, it is necessary to extract some meaningful values from the data so that machine learning techniques can use them as inputs. As a result, we apply the feature extraction technique to generate a feature vector from the cleaned dataset. The words in the text are represented by the features. Machine learning algorithm models operate on numerical data instead of textual data. Thus, the Term Frequency-Inverse Document Frequency (TF-IDF) technique is utilized to transform text into a numerical vector.

Tokenization, vocabulary creation, and encoding are the three steps that contribute to achieving the objective of this stage. Each sentence is divided into tokens during the tokenization process. During this process, the text is tokenized, which means that each word is treated as a separate piece of data and transformed into a separate token. The distinct tokens of each sentence are gathered and sorted alphabetically before being added to the vocabulary in the vocabulary creation phase. The produced vocabulary is called a feature vector and each feature is represented by a token in the vocabulary. After a vocabulary has been constructed from the whole text, the number of times each word occurs in each sentence is computed. Every sentence is given a unique numerical code in the encoding process that indicates

the frequency with which each feature appears in that sentence. Thus, the feature vector is obtained when performing these steps. As a result, the vocabulary in each sentence is displayed in the feature vector, together with the frequency of each token from the vector which appears in that sentence.

The TF-IDF technique counts the overall occurrence of a word in a document, which is computed using equation (5.1) (Schütze et al., 2008). The text is converted into numerical form by employing the TF-IDF approach.

$$tf - idf_{(w,d)} = tf_{(w,d)} \times idf_{(w,d)} \quad (5.1)$$

where $tf_{(w,d)}$ represents the number of times the word w appears in d documents, while $idf_{(w,d)}$ represents the number of times the word w appears across all documents. The TF-IDF value for word w that appears in document d is computed using equations (5.2) and (5.3), whereas n_d denotes the overall number of documents used for the training. The TF-IDF approach enables us to identify the document's most significant features.

$$idf_{(w,d)} = \log \frac{1 + n_d}{1 + df_{(d,w)}} \quad (5.2)$$

$$tf - idf_{(w,d)} = tf \log \left(\frac{1 + n_d}{1 + df_{(d,w)}} \right) + 1 \quad (5.3)$$

5.2.4 Classification

This stage involves training machine learning techniques utilizing the features produced in the last step. The task of classifying record pairs based on their values in the feature vectors into matching or non-matching is a binary classification task. These features are employed to train machine learning algorithms. Every machine learning model must have its parameter values adjusted to identify which parameter is the best performing while training the model. Then, we test the model using

the test data. We applied several supervised algorithms such as SVM with four different kernels (linear, sigmoid, RBF and polynomial), XGBoost, KNN, deep neural networks, random forest, decision tree, and GBM. We trained the models using scikit-learn (Pedregosa et al., 2011). The model's performance is evaluated utilizing the evaluation metrics.

5.2.5 Storing the data in blockchain

It is crucial to note that blockchain is not suitable for storing sensitive data as it is copied across numerous nodes, leading to redundancy, and its immutability contradicts with the GDPR's (Regulation, 2016) right to be forgotten because data saved on blockchain cannot be removed. Therefore, to overcome this issue, we propose storing users' personal data off-chain and storing a hash data pointer to that data on the blockchain.

In this phase, we use blockchain-based distributed off-chain storage for identity information using IPFS. The storage approach is immutable and content addressable. We require a distributed storage to maintain the massive amount of user information. IPFS stores the data in a manner that provides a unique hash for the data. The generated IPFS hash is more efficient and requires less storage space than the original data. The blockchain stores the generated hash which consumes less storage space. The data quantity on the blockchain can be diminished because only the hash values are recorded in the blockchain. Furthermore, data can be exchanged in an anonymous manner because the hash does not contain any information that reveals the user's identity. Therefore, the off-chain storage approach uses content addressable hashes to store the user's information, while immutability is achieved through the on-chain storage approach. The AES encryption algorithm is used to encrypt the data before storing it on IPFS. Thus, we stored the encrypted data in IPFS and then the generated IPFS hash is encrypted and stored in the blockchain. Privacy

could be ensured through combination of off-chain and on-chain mechanisms. Sensitive data is stored off-chain in a secure location to protect users privacy by using IPFS mechanism. When transferring data for off-chain processing, encryption algorithm is employed to ensure secure transmission and prevent unauthorized access. On-chain, we utilize encryption algorithm to secure data stored on the blockchain, ensuring that only authorized parties holding the corresponding decryption keys can access and decrypt the data. Consequently, sensitive data could be stored off-chain and encrypted, while an encrypted hash of the data can be stored on-chain. This approach enables authorized parties to verify the authenticity of the data without accessing the actual data. We can maintain the privacy of the user’s identity by applying this approach.

5.3 Experiments

The experiment setup for the model training and testing to detect duplicate records and provide details of its performance is described in this section. We describe the benchmark datasets that we employed in the experiment. Furthermore, we provide more details of the evaluation metrics. In addition, the experiment of storing data on blockchain is described.

5.3.1 Implementation

We perform the preprocessing task for the datasets which consists of the steps described in section 5.2.1. Moreover, we implement the classifiers using the *sklearn* library (Pedregosa et al., 2011). Additionally, we use Keras (Schanzenbach et al., 2019) to train and test the deep neural network model. The negative instances are generated by selecting one tuple from the positive instances and then randomly selecting one tuple from the relation that is not a match for the positive example. In this study, we use the Python programming language to build the classification

model. To provide the most accurate predictions, we train the model using the training test based on the target labels. We split the data into training and testing sets; by specifying the test size as 33%, and the rest of the data as the training set. We apply four popular evaluation metrics in the duplicate detection task which are accuracy, recall, precision, and F-measure. The solidity programming language is used to write the smart contract and Ganache personal blockchain to simulate the blockchain implementation. Remix IDE was used to implement and test the smart contract. IPFS provides the data storage layer for data storage and the data hash is kept on the blockchain network. Metamask is used to interact with the blockchain network. The AES algorithm is used to encrypt the data.

5.3.2 Datasets

We evaluate our proposed model using two popular benchmark datasets, namely Scholar-DBLP and ACM-DBLP (Köpcke et al., 2010) as shown in Table 5.1 which summarizes the datasets statistics. The datasets are publicly available and each dataset consists of four attributes (title, authors, venue, and year) which are about bibliographic domain and each dataset consists of two tuples to be compared. These two tuples contain duplicate records. These datasets have several quality issues, such as misspellings, missing values, etc. The dataset Scholar-DBLP contains 2,616 and 64,263 bibliographic records, respectively from DBLP and Google Scholar. The dataset ACM-DBLP contains 2,616 and 2,294 bibliographic records, respectively from DBLP and the ACM digital library. The objective is to identify whether two records belong to the same publication based on title, author, venue, and year attributes. We used an experimental approach where we selected data from multiple datasets of various sizes. We randomly selected three small subsets from each dataset. Small datasets are not explicitly defined in the literature (Dris et al., 2019). Therefore, we extracted the subsets from the large datasets by generating 20%, 50%,

and 80% of each dataset and then testing the models on these subsets as well as the full datasets. The aim of this approach is to investigate the impact of various dataset sizes on the performance of the classifiers.

	DBLP	Scholar	DBLP	ACM
Records	2,616	64,263	2,616	2,294
Ground truth	5,347		2,224	

Table 5.1 : Overview of evaluation benchmark datasets

5.3.3 Evaluation Metrics

We evaluate the classification models' performance in the domain of duplicate detection using four key metrics, accuracy, precision, recall, and F-measure. Several machine learning techniques were evaluated against two large datasets. We carry out an evaluation of the proposed duplicate detection method. To ensure that our proposed model is performing efficiently and effectively, we provide details about the evaluation metrics applied throughout the training and testing stages. We consider match quality in the evaluation. We quantify the quality of our method in terms of the perfect match result using the commonly used measures accuracy, precision, recall, and F-measure. Table 5.2 presents the F-measures obtained by all the approaches at hand. It is evident that the proposed method outperforms the other approaches on these datasets. We assess the classification models' performance using four commonly used metrics: accuracy, precision, recall, and F-measure. F-measure is used as the primary measure.

5.3.3.1 Accuracy

Accuracy is the most frequently used metric to measure the classifier's performance (Manning, 2008). We need to calculate the percentage of instances classified

correctly to estimate the classifier accuracy. Accuracy is calculated as (5.4):

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (5.4)$$

5.3.3.2 Recall

Recall is defined as the classifier’s ability to identify all positive instances (Manning, 2008). Recall is the fraction of correct matches predicted as matches. Recall is calculated using (5.5):

$$Recall = \frac{TP}{TP + FN} \quad (5.5)$$

5.3.3.3 Precision

Precision is a well-known evaluation metric in classification tasks. Precision is the fraction of match predictions that are correct (Manning, 2008). Precision is calculated using (5.6):

$$Precision = \frac{TP}{TP + FP} \quad (5.6)$$

5.3.3.4 F-measure

F-measure (also known as F-score) is the harmonic mean of precision and recall (Christen, 2012) which is calculated as in (5.7). We utilize F-measure to assess the performance of the classifiers. A higher F-measure value indicates the higher quality of the classification.

$$F - measure = \frac{2 \times Precision \times Recall}{Precision + Recall} \quad (5.7)$$

5.4 Results and Discussion

The following sections present the experiment results for the classification models developed using both datasets and their respective subsets, as well as storing

Classifier	Scholar - DBLP				ACM - DBLP			
	20%	50%	80%	100%	20%	50%	80%	100%
Linear SVM	96.43	99.18	98.24	98.86	99.65	99.20	99.17	99.07
Sigmoid SVM	81.07	69.54	82.29	75.68	67.06	87.77	80.84	78.02
RBF SVM	92.80	96.98	96.81	97.33	99.41	98.75	99.08	98.68
Polynomial SVM	82.67	95.02	94.42	97.11	95.96	98.26	97.15	97.26
XGBoost	91.75	97.60	97.16	97.75	99.71	99.23	99.16	99.22
KNN	79.99	97.02	96.24	96.91	98.25	98.50	96.05	96.97
Deep Neural Networks	93.67	98.47	97.80	97.88	99.28	99.05	98.58	98.82
Random Forest	97.14	98.90	98.61	99.17	99.14	98.78	99.08	98.91
Decision Tree	90.68	98.82	97.27	98.21	99.68	99.25	99.34	99.26
GBM	82.83	97.50	97.02	97.58	96.43	98.53	99.18	98.83

Table 5.2 : F-measure results of the classifiers

and managing data on blockchain. The experiments were conducted on a Jupyter Notebook on a Windows 10 personal computer with CPU 1.90 GHz, Core i7 processor and 16.0 GB memory (RAM). The performance of the classification algorithms is demonstrated in Figures 5.2 and 5.3 in terms of accuracy, precision, recall, and F-measure. The x-axis in the figures indicates the dataset's size. 100% indicates that the full size of the dataset is used for training and testing the datasets, as shown in Table 5.2, which presents the F-measure average values obtained from ten iterations in all classifiers.

Each line chart consists of three segments, each of which represents the result in three different scenarios for datasets of various sizes. The first segment in the line charts ranges from 20 to 50. The change in the classifier's performance occurs when the dataset size increases from 20 to a larger dataset of 50. The line charts range from 50 to 80 in the second segment, showing how the classifier's performance

changes when the dataset size increases from 50 to a much larger dataset of 80. In the third segment, the range is from 80 to 100. This represents a key conclusion in the chart, as it illustrates the differing results when the classifier was trained on a full dataset 100 as opposed to training on a smaller dataset 80.

Numerous conclusions can be drawn from these figures. The majority of classifiers exhibit a relatively similar performance when the training set size increases across all four performance metrics. This can be observed by comparing the performance of a single classifier against various performance metrics. In addition, the metrics improve relatively on the full-sized Scholar-DBLP dataset, but the classifiers vary in their performance with datasets of different sizes. In contrast, the performance of all the classifiers for the ACM-DBLP dataset decreases across all metrics when the dataset size increases. The most notable finding is that whenever the ACM-DBLP dataset size increases, the performance of the GBM model improves. Furthermore, the classifiers that perform the best across the datasets vary. When comparing different classification algorithms, the results show that random forest and linear SVM classifiers performed better than all the other classifiers on the Scholar-DBLP dataset (Figure 5.2), while decision tree, XGBoost, and linear SVM outperform all the other classifiers on the ACM-DBLP dataset (Figure 5.3). However, sigmoid SVM is the worst performing classifier on both datasets for the majority of the performance metrics. We evaluated the supervised SVM using four kernel methods (linear, polynomial, sigmoid and RBF) and, as shown in Table 5.2, the linear SVM kernel performed better than SVM classification and the other kernels.

A number of observations can be made regarding the classifiers' runtime, as shown in Table 5.3. On both datasets, decision tree and random forest are the fastest performance classifiers regardless of the dataset size. In contrast, the deep neural network classifier requires a far longer training time compared to other classifiers on both datasets, making it the slowest classifier. It is noteworthy that the random

forest and decision tree classifiers outperform the other classifiers on the Scholar-DBLP and ACM-DBLP datasets, respectively, and both are the fastest classifiers. Since random forest and decision tree are the most effective and efficient classifiers on both the Scholar-DBLP and ACM-DBLP datasets, respectively, we can conclude that dataset size and runtime will not have an impact on their performance.

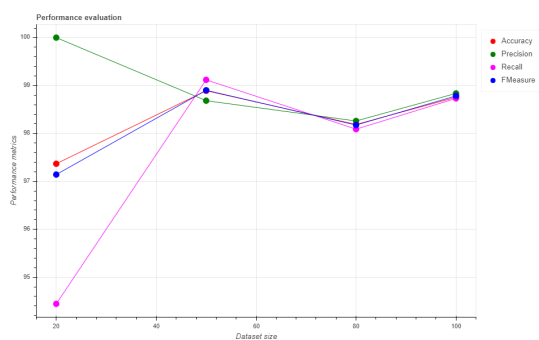
Classifier	Scholar - DBLP				ACM - DBLP			
	20%	50%	80%	100%	20%	50%	80%	100%
Linear SVM	1.02	20.53	113.1	289.1	0.10	1.36	6.46	17.93
Sigmoid SVM	1.50	45.70	202.2	668.5	0.10	2.33	15.43	46.03
RBF SVM	1.26	31.97	189.7	436.2	0.10	2.20	15.27	36.23
Polynomial SVM	1.16	31.33	208.1	542.3	0.10	1.96	14.37	34.40
XGBoost	13.30	84.17	277.8	554.5	1.53	6.66	23.47	44.33
KNN	0.93	14.57	68.6	139.4	0.10	1.43	6.00	12.50
Deep Neural Networks	34.6	260.9	1056	1555	17.47	90.83	208.20	348.10
Random Forest	1.33	12.40	47.53	89.73	0.23	0.96	4.53	9.53
Decision Tree	0.43	1.26	4.06	11.17	0.03	0.13	0.46	0.83
GBM	33.67	115.5	200.1	270.5	1.40	3.46	6.00	7.93

Table 5.3 : The runtime of the models in seconds

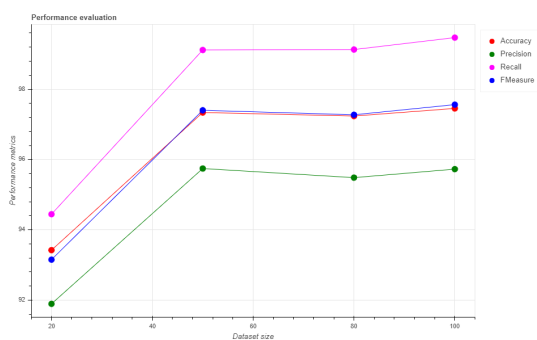
The results of the study shed light on valuable information on the classifiers' performance with datasets of different sizes. The classifiers' overall performance is determined by how closely the dataset resembles the original distribution, not by its size. Our experiments demonstrate that the most robust classifier for various dataset sizes is random forest and linear SVM, followed by decision tree, deep neural networks, and XGBoost, while SVM with the sigmoid kernel is the least robust classifier. A noteworthy observation is that a robust classification model for datasets

of different sizes does not always mean that it performs optimally in comparison to other models. This is demonstrated by the fact that while the random forest and linear SVM models achieved the best performance regardless of dataset size on the Scholar-DBLP dataset, they performed slightly worse on the ACM-DBLP dataset. Overall, the percentage of duplicates identified is an important measure of the effectiveness of a duplicate detection system. A high percentage of duplicates identified means that the system is good at finding duplicate records, which can lead to a number of benefits, such as improved data quality, reduced storage costs, improved data analysis, and improved data security.

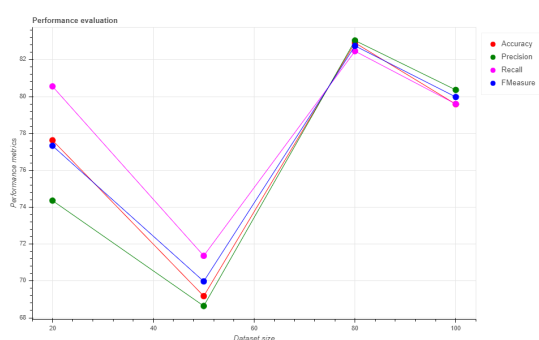
To avoid storing a huge amount of data on the blockchain, we used the off-chain and on-chain mechanism to store the data. The identity provider is responsible for uploading the file to the IPFS. The identity provider encrypts the file using the AES symmetric algorithm before transmitting the file to the IPFS. IPFS stores the encrypted file and generates the content-addressed hash and then sends it back to the identity provider. The file can be retrieved from IPFS by utilizing the generated hash. Subsequently, the file's content-addressed hash is encrypted using AES and is then kept in the blockchain by the identity provider who has the right to store the data. Even if a user gains access to the stored data in blockchain, the user will be unable to access the hash since the hash value itself is encrypted. Consequently, encrypting the hash adds an additional degree of protection to the data. The action is verified by the smart contract to ensure that it is being carried out by the owner's approved public address. The smart contract is designed to grant access to only the contract owner. Thus, access control is restricted to the contract's owner to store the data in the blockchain. Data privacy can be ensured by applying access control. Moreover, the data privacy is preserved using the AES symmetric encryption since only the identity provider can decrypt the data and also only the data hash is kept on blockchain. Furthermore, immutability is guaranteed since once a file is



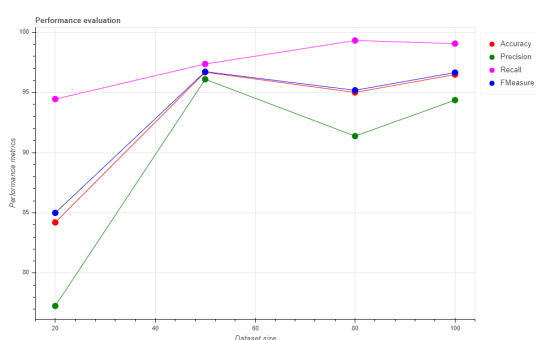
(a) Linear SVM



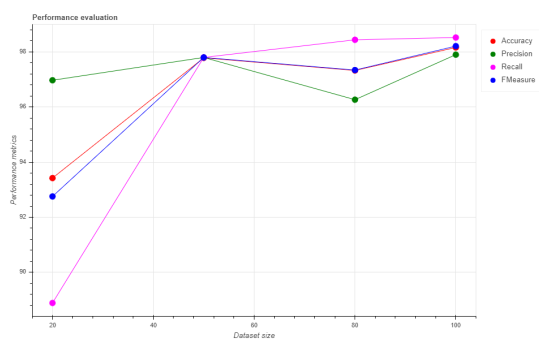
(b) rbf SVM



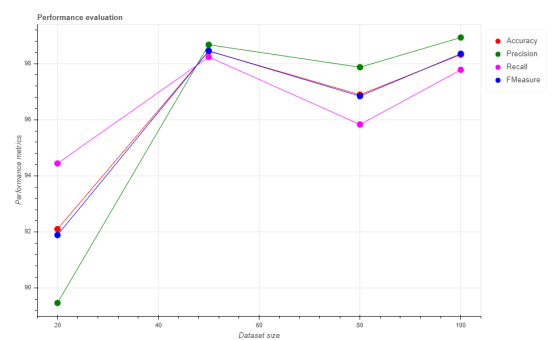
(c) Sigmoid SVM



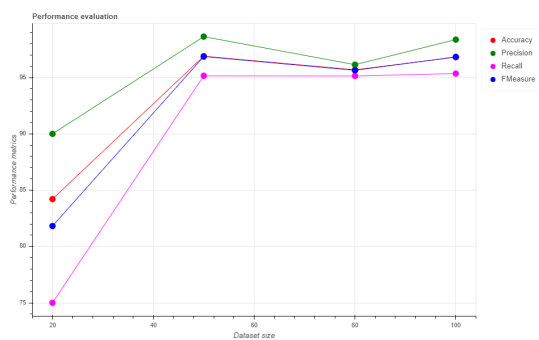
(d) Polynomial SVM



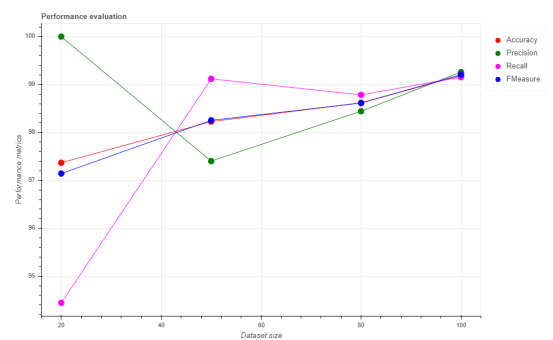
(e) XGBoost



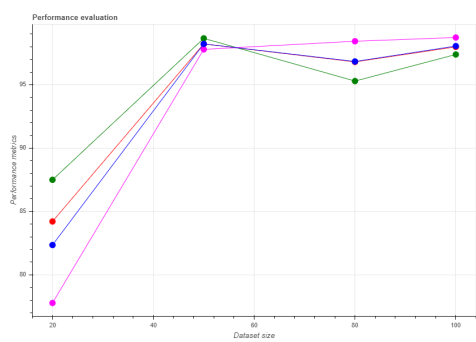
(f) Decision Tree



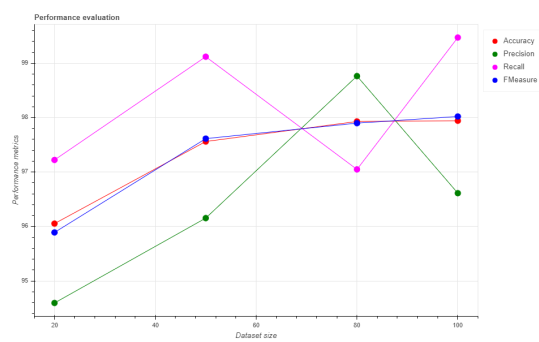
(g) KNN



(h) Random Forest

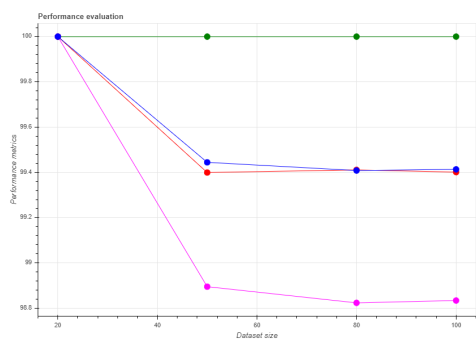


(i) GBM

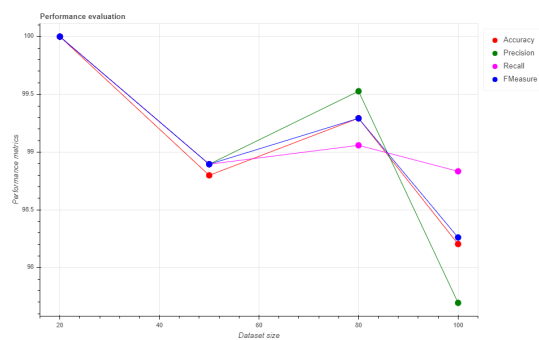


(j) Deep Neural Networks

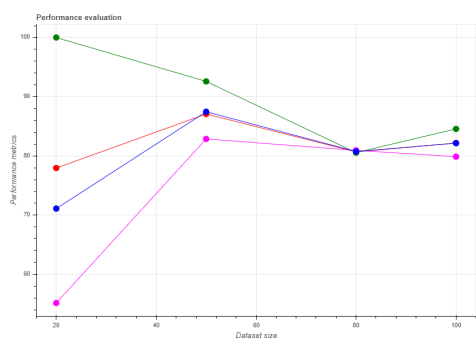
Figure 5.2 : The performance of the classifiers on the Scholar-DBLP dataset



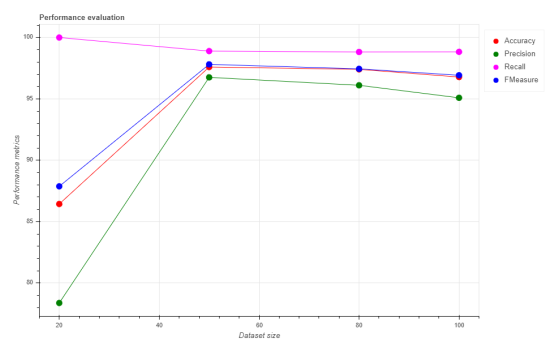
(a) Linear SVM



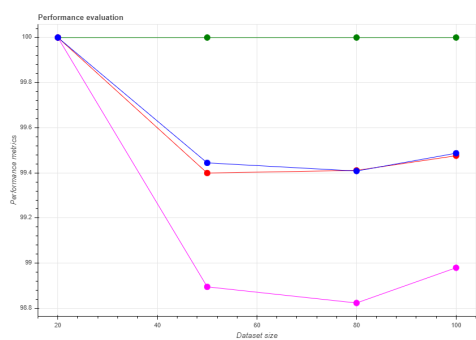
(b) rbf SVM



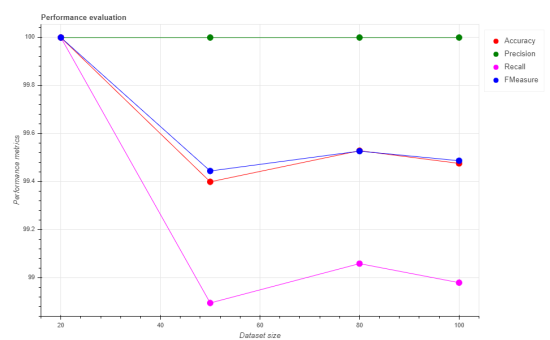
(c) Sigmoid SVM



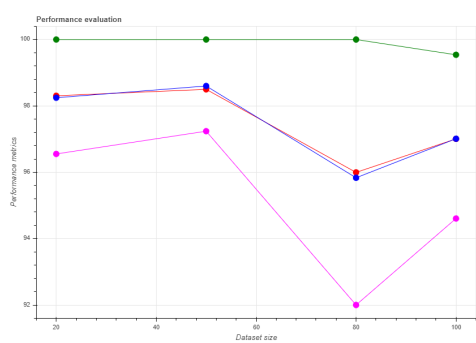
(d) Polynomial SVM



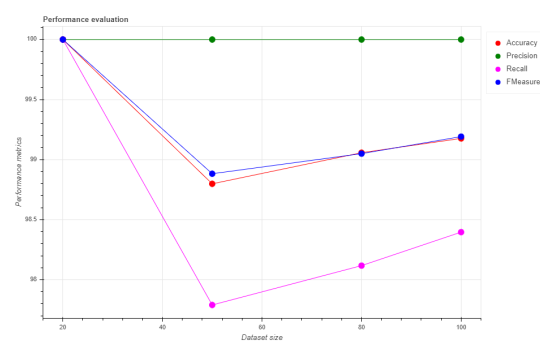
(e) XGBoost



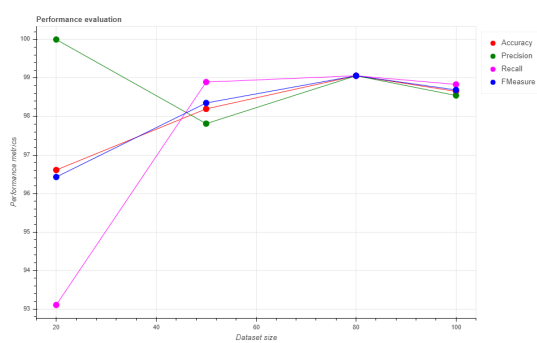
(f) Decision Tree



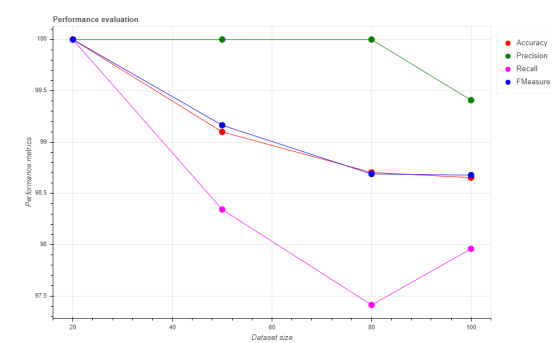
(g) KNN



(h) Random Forest



(i) GBM



(j) Deep Neural Networks

Figure 5.3 : The performance of the classifiers on the ACM-DBLP dataset

stored on the blockchain, it cannot be modified. The service provider is required to provide the data hash that is recorded in the blockchain to fetch the data. Figure 5.4 illustrates the process of identity storage in our model. We successfully tested the implementation of the smart contract using the Ganache personal blockchain network.

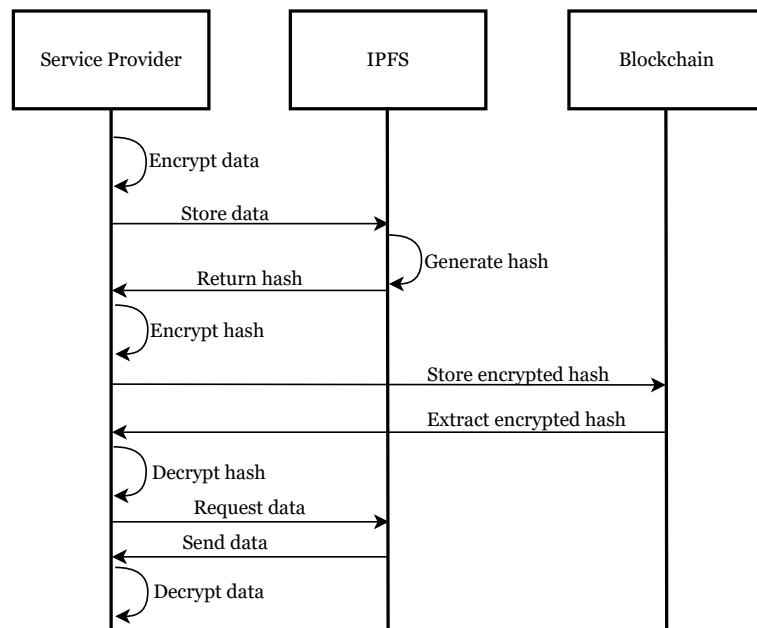


Figure 5.4 : Sequence diagram for identity storage

5.4.1 Comparison with the state-of-the-art models

The effectiveness of the proposed method is compared with the state-of-the-art methods once the results are obtained. The outcomes demonstrate that the proposed model outperforms the state-of-the-art models on both datasets. Table 5.4 presents the F-measure findings of the proposed model along with the state-of-the-art methods. The best result obtained by one of the methods is highlighted in bold. The outcomes demonstrate that our method outperforms the previous methods in terms of F-measure performance. We compare our model against four methods. The four methods are as follows:

1- Ditto (Li et al., 2020) is a supervised DNN model for duplicate detection which is based on pre-trained matching entities based on language models. It enables the injection of domain knowledge by highlighting relevant input information that may aid in the labeling decision-making process.

2- DeepER (Ebraheem et al., 2018) is a DL-based model for duplicate detection which aggregates data entries into vector representations and performs binary classification using a feedforward neural network based on the similarity of the two vectors.

3- Gradient-based matching (Reyes-Galaviz et al., 2017) is a supervised model that is capable of adjusting its structure and parameters according to similarity scores derived from the similarity functions on various attributes.

4- Seq2SeqMatcher (Nie et al., 2019) is a deep learning-based model that aims to effectively address the heterogeneous and dirty issues by modeling duplicate detection as a token-level sequence-to-sequence matching task.

The results of our model along with the other four methods are shown in Table 5.4. For each dataset, the best result is displayed in bold.

Datasets	DITTO	DeepER	A supervised gradient	Seq2Seq Matcher	Proposed model
Scholar DBLP	95.60	97.67	98.60	95.30	99.17
ACM DBLP	98.99	98.60	98.10	98.90	99.26

Table 5.4 : Comparison of the proposed model with state-of-the-art models

5.5 Conclusion

In this chapter, we focus on integrating blockchain with identity management. This research presents an intelligent method to resolve the issue of detecting duplicate identities on top of blockchain. The solution is based on a combination of the machine learning approach and blockchain-based smart contracts. Furthermore, this work combines machine learning techniques with blockchain technology to detect duplicate identities and to manage and store identities in an immutable manner to guarantee privacy. In addition, this work examines the relationship between dataset size and the performance of machine learning algorithms. We conducted extensive experiments using several classification models on two real-world datasets. Our findings demonstrate that it outperforms the existing duplicate detection approaches on the two benchmark datasets. Additionally, we conducted experiments to manage identities on top of blockchain by encrypting the data using AES and uploaded it to IPFS which generates the data hash. The generated hash is encrypted and stored on the blockchain through the use of smart contracts. The experiments' findings indicate that the proposed model has the capability to identify duplicate identities while maintaining the privacy of the users' identities.

We describe the early warning system that is used to generate alerts for users in the following chapter.

Chapter 6

Blockchain-based method for generating notifications for user identity expiration

6.1 Introduction

In this chapter, we discuss the process involved in developing a model to warn users when their identities are about to expire. A user identity contains a large amount of information, including its expiration date. Upon expiration, a user identity is considered invalid, and the user will be unable to use it to access or request services. There are three phases in building this model, namely the user phase, server-side phase and blockchain phase. In addition, we propose an approach to notify users when their identities are about to expire based on the user information stored on blockchain, which is a comprehensive solution to research objective 2. In this chapter, we provide the outcomes of the validation and implementation of the proposed method to answer the second research question. We develop an alert model using blockchain and test the system using the following technologies to answer this research question:

1. Blockchain: A blockchain, as explained in Chapter 2, is a distributed, replicated ledger where each transaction carried out in the blockchain is copied in every node of the blockchain, ensuring the irreversible modification of records. It is nearly impossible to alter transactions that have been recorded on blockchain since transactions are distributed across multiple nodes. In this research, blockchain is used for storing and retrieving users' identity information. In addition, it retains the user identity information, the transaction

actions and the most recent trustworthiness scores of the users' identities and provides feedback to the user interface.

2. Metamask: This is an extension for the web browser that interacts with the Ethereum network to run DApps in the browser rather than running the full Ethereum node. It stores data related to the Ethereum wallet such as the public address and private key. In this research, we connected React with blockchain using the Metamask extension for Chrome.
3. React: we use React to develop the front-end or the user interface.
4. Python: is used to develop the REST-API.
5. Ganache: is used as a private Ethereum blockchain environment which enables the blockchain functionality to be locally emulated and to test the smart contracts that have been published.
6. Mailgun: Users are able to send and receive emails using the Mailgun service. Additionally, Mailgun facilitates the process of incorporating email into current applications.
7. Solidity: Smart contracts can be built on Ethereum's blockchain utilizing the Solidity programming language.
8. Smart contracts: A smart contract refers to a computer program that can be performed on the blockchain to implement an agreement. The transaction actions and reliability computations are performed automatically. After this, the transaction actions and reliability score are broadcast to the blockchain.

6.2 Generating notifications about the expiration of user identity

This section discusses modelling and generating notifications about the expiration of a user's identity and delivering it to users to achieve objective 2 of this thesis.

6.2.1 Solution Workflow

The proposed EWS model is implemented utilising smart contracts and Ganache to develop a local virtual Ethereum blockchain. We utilized the Ganache Ethereum network to achieve research objective 2 in this study. By using Ganache, we are able to simulate the blockchain on a local machine. The information that is stored in the network comprises the identity holder's name, identity number, expiration date, and email address.

This system is intelligent because it automatically generates notifications for users and employs an algorithm to detect users' identities have expired based on identity information. Service providers are authorised to add user information and subsequently trigger the system to send notifications to users to remind them to take a certain action, so the information that has been entered is reliable. Blockchain is used to keep all of the information that is associated with an individual's identity. The immutability of the blockchain's records ensures that the user identity information can never be modified except with the consent of the majority. Figure 6.1 illustrates the proposed Intelligent Method for Generating Notifications.

Algorithm 1 is used to identify which users' identities have expired and generate alerts:

All the information related to users' identities is stored in the blockchain. Each

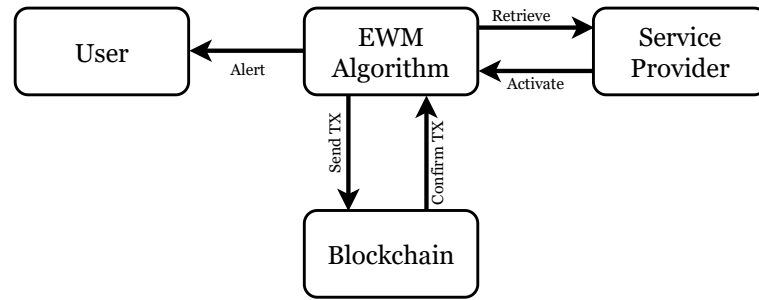


Figure 6.1 : Early Warning Model to alert users

Algorithm 1 Pseudo-code for the early warning algorithm

- 1: Start: P ▷ P: Preferred Value
 - 2: R for every user identity ▷ R: Repeat
 - 3: Compare P with the current date
 - 4: IF (P < Current date value)
 - 5: Activate the system and generate N ▷ N: Notification
 - 6: Else
 - 7: Repeat the process
 - 8: End
-

service provider is allowed to activate the algorithm to send alerts to users. The user identity expiration date is compared to the current date using the proposed algorithm. If the difference between the two dates is less than a predetermined threshold value, the user is notified that their identity will expire within a certain time. The threshold, which could be any number of days, is determined by the user, for instance, five days in advance, which the proposed algorithm then employs. Then, an email is sent to the user as a reminder.

6.2.2 Generating Alerts for Users

We propose the following workflow to notify users about imminent expiration of their identity information. Figure 6.2 depicts the workflow process. The following steps are required to store new information and generate an alert:

1. Inserting new information: First, the service provider inserts new information on the blockchain in relation to the identity of users, including the expiry date and their email address.
2. Determining the preferred value: The preferred value, which can be any number of days, is determined by either the users or service providers.
3. Detecting an expired user identity: The proposed algorithm is applied to identify the user identities that are approaching expiration by comparing the expiration date with the current date.
4. Generating notifications: Alerts are generated and delivered to the users via email to notify them that their identity is about to expire.
5. Receiving emails: The users receive an email reminding them to renew their identities or to take action.

6.3 Model Implementation

The primary objective for the prototype's development is to emulate the model's operation which is based on the implementation of Ethereum smart contracts to manage user identities. The objective is to assess the effectiveness of the model described in section 6.2 in generating alerts for user identities that are about to expire. According to the existing literature on identity management, one of the significant issues is that service providers supply services to users when they present

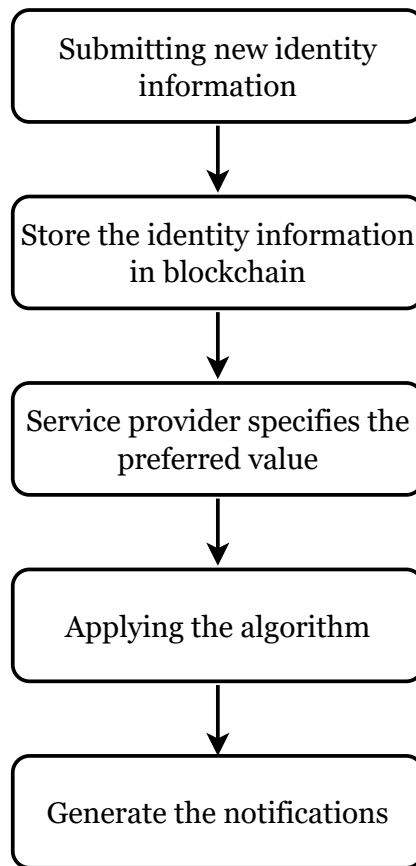


Figure 6.2 : The steps involved in generating warnings

a legal identity; otherwise, users are prevented from accessing facilities that demand a legal identity (World-Bank, 2018). Obtaining a new identity could take some time, as the process of granting an identity requires a thorough review of personal information. Therefore, it poses an issue if the user identity expires before the user can reissue an identity. For this reason, the model for warning users of impending user identity expiration is proposed.

6.3.1 Steps for generating blockchain-based warnings

Figure 6.3 depicts the development of decentralized application (DApp) to generate notifications to users.

1. A smart contract, known as Remix IDE, is built in the development environ-

ment utilizing the solidity programming language for interaction between a smart contract and blockchain.

2. The smart contract is published for processing to Ganache.
3. The user interface of the prototype is designed using React to communicate with the blockchain.
4. The information is added to React through the developed interface.
5. The information is stored on the Ethereum blockchain utilizing Metamask.
6. A server-side for tracking and checking the expiration date is set up.
7. The email delivery service Mailgun is integrated with the server-side.

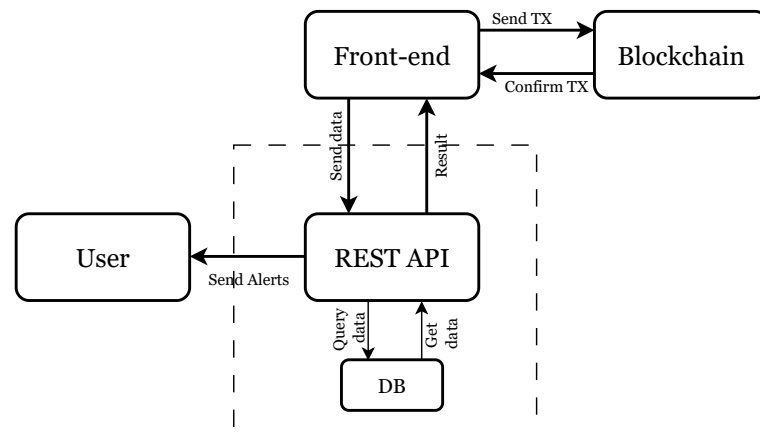


Figure 6.3 : Blockchain DApp for the proposed EWS model

6.4 Conclusion

In this chapter, we discussed the steps for generating notifications to users. In particular, we discussed in a stepwise manner the processes involved in the blockchain execution phase and in generating alerts to the users phase. In addition, we presented an approach to generate alerts to users depending on the provided

information using a smart contract. This proposed smart contract-based model for generating notifications and reminding users about the imminent expiry of their identities addresses the second research objective of this thesis. In this chapter, we also proposed a simulation framework to validate the solution to the research objective 2. The validation and implementation details related to this chapter can be found in Chapter 8.

The next chapter discusses the steps involved in developing a model to compute the trustworthiness score of users.

Chapter 7

Blockchain-based Model for Computing the Trustworthiness of a User's Identity

7.1 Introduction

We design a solution that utilizes the identities provided by individuals to compute the trustworthiness value based on the type of user identities. User identity components combine to form a distinct entity. These components are inherited at an individual's birth. Establishing trust between users and service providers entails undertaking certain computations to determine trustworthiness.

Mutual confidence is the cornerstone of the relationship between service providers and their customers. We specified the user identity documents and assigned a specific value for each. The documents selected and the values assigned to each document are inspired by the personal identification system that is used by the Australian Government (Australian Government, 2018). These values are converted to a percentage because we adopted a percentage scale in this model. The scale comprises of numerical values ranging from "0" to "100". If the user provides all the identity documents, they receive a 100 percent trustworthiness score, which is the highest level of trustworthiness that can be achieved. If a user does not provide any identity documents, they receive the lowest possible trustworthiness score which is 0 percent. Otherwise, the user is given a trustworthiness value depending on the number of user identity documents that are provided and the weight assigned to each. The weights of the user identity documents that are used in this model are listed in Table 7.1. Consequently, the user is not required to provide all identity documents

to acquire a trustworthiness score as users can be given a partially verified identity if they provide only some of the identity documents. A partially verified identity allows users to access services even if they are unable to present all of the necessary identification documents of identification. Users' trustworthiness values are taken into account by the service provider to assess whether or not to provide services to the respective user. In some cases, service providers may set a specific level of trustworthiness and offer their services in accordance with the trustworthiness level of each user. Service providers may have a varying level of trustworthiness depending on the type of service that is offered to the user. Furthermore, each service provider may adopt a different level of trustworthiness compared to other service providers. As a result, service providers can utilize trustworthiness values to make individual decisions about their customers depending on their trustworthiness level.

User Identity Documents	User Identity Weight
Passport	70
Birth Certificate	70
Citizenship Certificate	70
Driver Licence	40
Photo Identification Card	40
Student ID	40
Government Employee ID	40
Medicare Card	40
Credit Card	25
Marriage Certificate	25
Total Weight	460

Table 7.1 : Weighted User Identity Documents

The trustworthiness value is calculated using Equations 7.1 and 7.2. All the trustworthiness scores are stored on blockchain which has immutable records. Therefore, once the trustworthiness values have been recorded, they cannot be changed. When calculating a user's trustworthiness value, a threshold is used to determine whether the user is trustworthy or untrustworthy. If the user's score is above a predefined threshold, the user is deemed a *trustworthy* user otherwise, the user is deemed an *untrustworthy* user. Alternatively, each service provider has the ability to individually define their own threshold in a manner that is different from other service providers, allowing for greater flexibility. The overall score is determined based on the number of different types of identities that have been provided by the user. When a user provides only one identification document, the user is given a low score depending on the defined weights that have been assigned to that identity document. However, if a user provides several identity documents, the user will be given a higher trustworthiness score.

7.2 Determination of the Trustworthiness Values of User Identities

This section explains how we model and specify the current trustworthiness values for the user's identity to address objective 3.

7.2.1 Solution Workflow

In this thesis, we simulated blockchain implementation utilizing the Ropsten Ethereum network to achieve objective 3. The Ethereum platform is used and its programming language is Solidity. We also use React to design the interactive user interface.

The overall trustworthiness score is computed by aggregating the weights assigned to each user identity document which has been submitted, and the system collects these

weights automatically utilizing an algorithm which makes it an intelligent system. Blockchain stores the trustworthiness score of the users, while the decentralized application stores all the weights assigned to the user identity documents. The trustworthiness score can never be changed except by majority consensus because blockchain records are immutable. Figure 7.1 depicts the workflow of the intelligent method for trustworthiness calculation.

The proposed algorithm for computing the user's total trustworthiness score is:

$$\text{Trustworthiness Score} = \frac{(W_1 + W_2 + W_3 + \dots + W_i)}{(n_1 + n_2 + n_3 + \dots + n_i)} \times 100 \quad (7.1)$$

$$\text{Trustworthiness Score} = \frac{\sum W}{\sum n} \times 100 \quad (7.2)$$

where W denotes the weight value allocated to each user identity and n denotes the sum of all users' identity' weights.

DApp is used to obtain all the weights. The trustworthiness calculation in equations 7.1 and 7.2 use previously stored weights. The trustworthiness value is linked to each user in the system so the service provider is able to verify the overall score for each user. The blockchain maintains the user identity with its associated information, which include user identity type and identity weight.

7.2.2 Calculation Trustworthiness Score of User's Identity

We propose the following workflow method to compute the trustworthiness score after a new user identity has been submitted. Figure 7.2 illustrates the process for computing the trustworthiness.

The following steps outline the process for calculating and storing the trustworthiness value for a specific user:

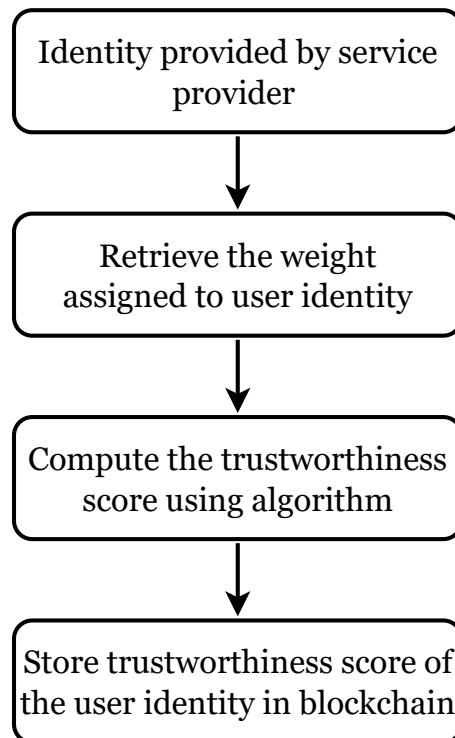


Figure 7.1 : Intelligent Method for Trustworthiness Calculation

- (a) Submitting a new user identity document: The new user identity document is inserted into the user's list of documents.
- (b) Determining the type of the newly added identity document to assign the appropriate weight.
- (c) Retrieving the weight that was allocated to the user identity document based on its type.
- (d) Calculating the overall trustworthiness of a user: The overall trustworthiness score of a user is calculated using the proposed algorithm taking into account all of the users' identity documents that have been provided.
- (e) Providing the service provider with the overall trustworthiness score of the user which is then stored on blockchain.

- (f) Obtain blockchain confirmation of the submission and publish the user's total trustworthiness score on blockchain.

Figure 7.2 depicts the flow of the preceding steps.

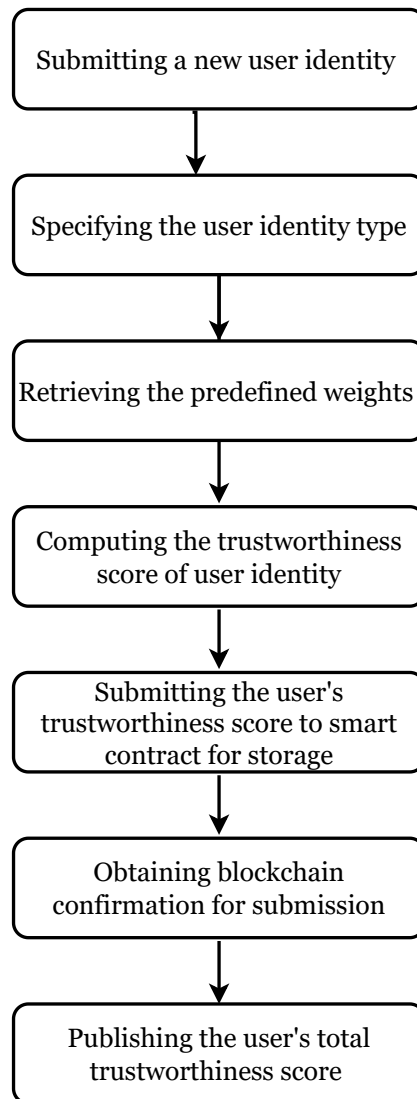


Figure 7.2 : An Overview of the Steps involved in the User's Trustworthiness Score Calculation

7.3 Prototype Implementation

The primary objective in developing the prototype system is to simulate the proposed model function, an intelligent model that manages user identity using blockchain technology. The effectiveness of the method described in Section 7.2 is evaluated by using the built prototype to calculate the trustworthiness score of the user's identity.

As discussed in Chapter 2, establishing trust between service providers and their customers is challenging and is associated with identity management as discussed in the current literature (Gefen, 2002). Service providers are facing growing challenges to differentiate between users who are trustworthy and those who are acting maliciously as a result of the increased prevalence of user identity fraud around the world (Dellarocas, 2001). The integrity of users' trustworthiness values can now be appropriately preserved using our intelligent solution that incorporates blockchain-based smart contracts and identity management. Furthermore, our approach addresses the core of the issue by presenting an efficient method for calculating the trustworthiness score of a user's identity.

7.4 Prototype Evaluation and Discussion

To achieve objective 3, we model and calculate the users' trustworthiness value using the proposed algorithm. Table 7.2 presents the trustworthiness scores of five users based on the provided identity documents. The trustworthiness scores vary, depending on the identity documents provided, indicating that the proposed method successfully calculates the users' trustworthiness scores. Using the blockchain-based method to calculate the trustworthiness score has several benefits, including accuracy, trustworthiness, and security.

Users	Trustworthiness Score
User A	92%
User B	96%
User X	77%
User Y	58%
User Z	82%

Table 7.2 : Computed scores of users

7.5 Conclusion

This chapter presents a trustworthiness calculation method for a user's identity based on blockchain. The trustworthiness score that is calculated by the proposed algorithm is used to identify the trustworthiness value of users. Users are classified as trustworthy or untrustworthy depending on their trustworthiness score or by employing a predefined threshold as specified by the service providers. The proposed method contributes to establishing greater confidence between providers and users regarding user identities. In addition, we developed a simulation model as a method of verifying the approach to achieve research objective 3. The findings of the validation and implementation process indicate that our approach is capable of computing the total trustworthiness value of the users' identity.

The following chapter will detail the functionality of the prototypes that we developed to achieve all the objectives outlined in this thesis.

Chapter 8

Evaluation and Prototype Implementation

8.1 Introduction

In the previous chapter, we presented and discussed an overview of the proposed solution for identity management based on blockchain technology to calculate the trustworthiness score for a user's identity. Based on the research solutions described in chapters 5, 6, and 7, this chapter demonstrates the functioning of the three different prototypes developed for each objective. In addition, we evaluate the effectiveness of the machine learning models using popular measurement metrics. We also illustrate the process of setting up the prototype in a step-by-step fashion, which involves the setting up decentralized applications and the blockchain using screenshots and figures. For all the prototypes, we choose Ethereum blockchain due to its open source nature to construct decentralized applications with support for smart contracts. Furthermore, it is supported by large communities including public Ethereum test networks that are available for experimental purposes. We also use the prototypes that we constructed to illustrate how the proposed models function for storing users' identities (section 8.2), for generating notifications (section 8.3) and computing trustworthiness score (section 8.4).

8.2 Evaluation of the Performance of Machine Learning Techniques and the Prototype for Managing Users' Identities

This section details the implementation of the identity management method which employs machine learning techniques to detect duplicate user identities and manages these identities using blockchain technology.

8.2.1 Evaluation of machine learning performance

The performance of the classifiers is assessed using four common measures, namely recall, precision, accuracy, and F-measure. The results of these metrics are presented in sections 5.3 and 5.4.

8.2.2 Prototype for Managing User Identities

A decentralized application (DApp) is developed to manage users' identities on top of blockchain. Using Dapp, the file is uploaded to IPFS, and then a hash of the uploaded file is recorded on the Ethereum blockchain. The purpose of DApp is to utilize IPFS to upload files, after which Ethereum blockchain is used to keep the hash of IPFS. A confirmation of the transaction is issued to the user from the Ethereum blockchain after the IPFS hash has been broadcasted. The frontend is developed utilizing React. Users who have the MetaMask added to the browser can utilize this Dapp.

We use Ganache to run the private blockchain. Ganache generates private keys that can be used to connect it with the MetaMask wallet. Then, Ganache is used to deploy the smart contract to obtain its address. We design the user interface using React as it is an interactive environment. Users interact with React by opening the Chrome browser and entering a localhost to open the home page. The DApp requires several settings to be input to be connected with IPFS and Ganache. Infura

is used as the gateway to the IPFS node which is responsible for storing the data. In this step, the Infura endpoint should be specified which is where we can connect to upload or download files from IPFS. The AES encryption algorithm is used to encrypt and decrypt the files.

8.2.3 Blockchain Configuration

The programming language known as Solidity was utilised in the creation of the smart contract, and Remix IDE is a web-based editor used to write and compile smart contracts. Figure 8.1 depicts the browser-based Remix IDE interface. There are several libraries included with the Remix IDE development environment that accelerates the development process. Smart contracts can be tested, debugged, and deployed using Remix IDE.

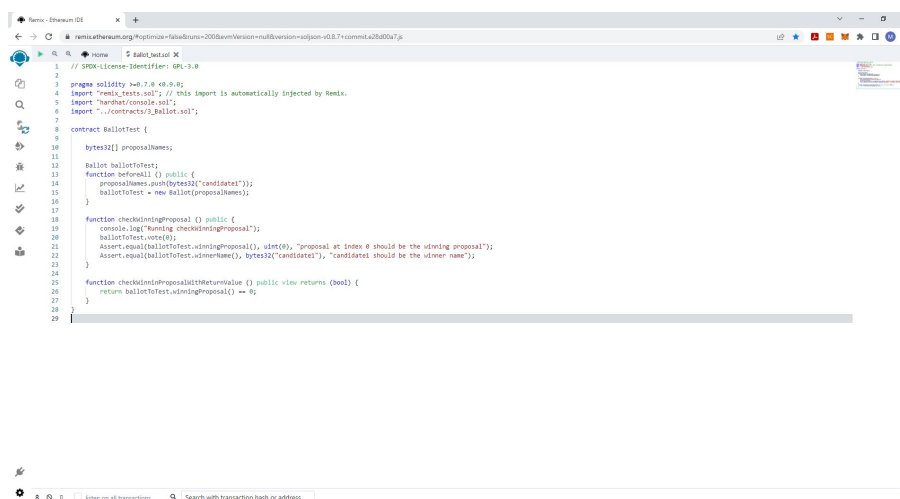


Figure 8.1 : Remix IDE interface

The next step is to set up Ganache as a local Ethereum blockchain. Ganache is used for operating a personal local blockchain so that smart contracts are tested, compiled, and launched on a local blockchain simulator. Ganache enables the secure

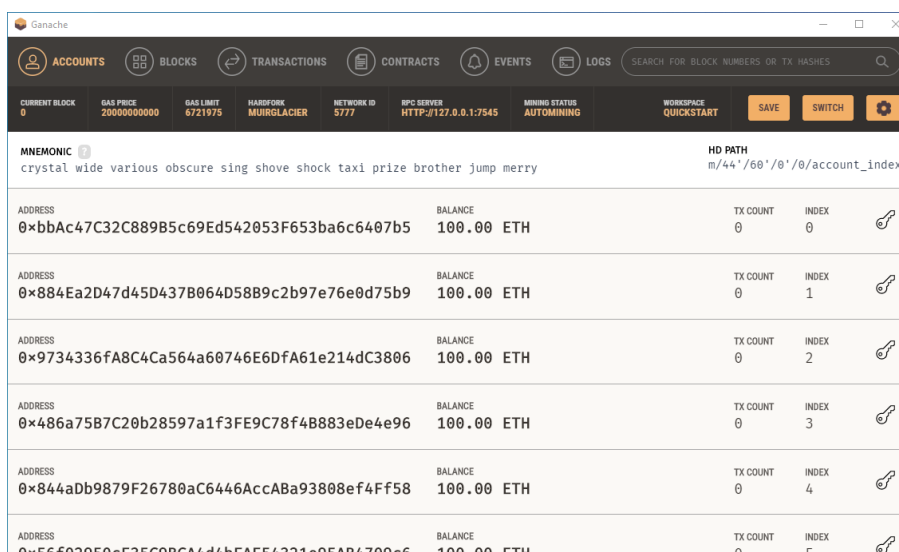


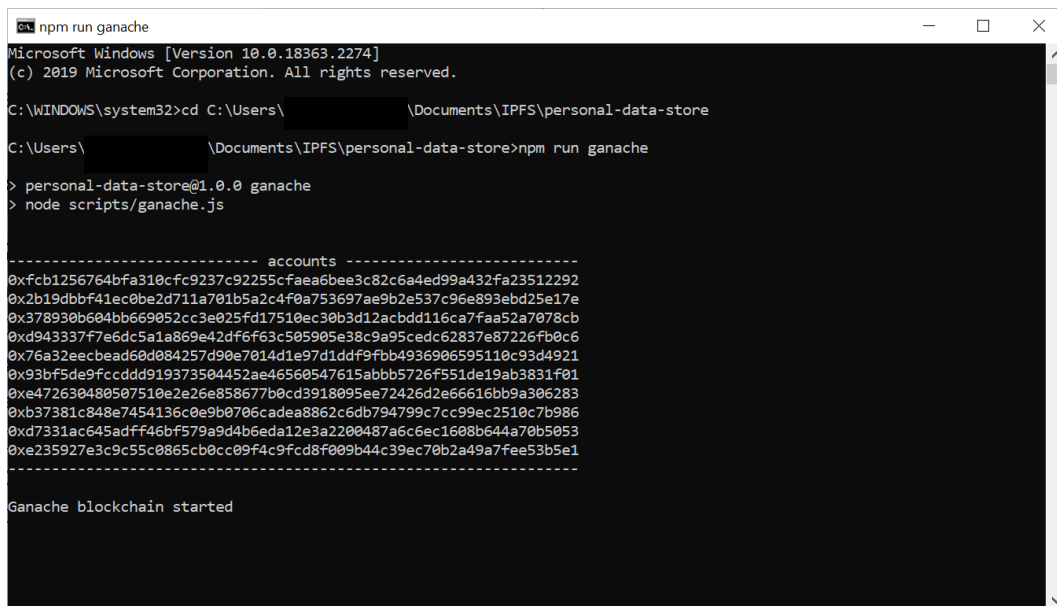
Figure 8.2 : The interface of Ganache for objective 1

testing and deployment of smart contracts in a secure environment. Thus, smart contracts can be tested and deployed faster with Ganache. Figure 8.2 shows the interface of Ganache with user accounts with 100 ETH as the balance available.

Ganache is executed during the implementation using Ganache-CLI, which is a tool for the command line. We run Ganache using the Node Package Manager (NPM) to start Ganache. The execution of Ganache through command-line with user accounts is demonstrated in Figure 8.3.

In Figure 8.4, the smart contract that has been compiled is launched on Ganache. The smart contract's address is generated after the contract has been deployed. The deployment on Ganache is demonstrated in Figure 8.5, which provides the details of our deployment. In addition, it presents details such as the transaction hash, the address of the smart contract, and the timestamp of the transaction that was published on the blockchain.

The MetaMask extension for Google Chrome was added. MetaMask safely maintains the private key and the Ethereum address. MetaMask enables us to interact



```
npm run ganache
Microsoft Windows [Version 10.0.18363.2274]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\WINDOWS\system32>cd C:\Users\          \Documents\IPFS\personal-data-store

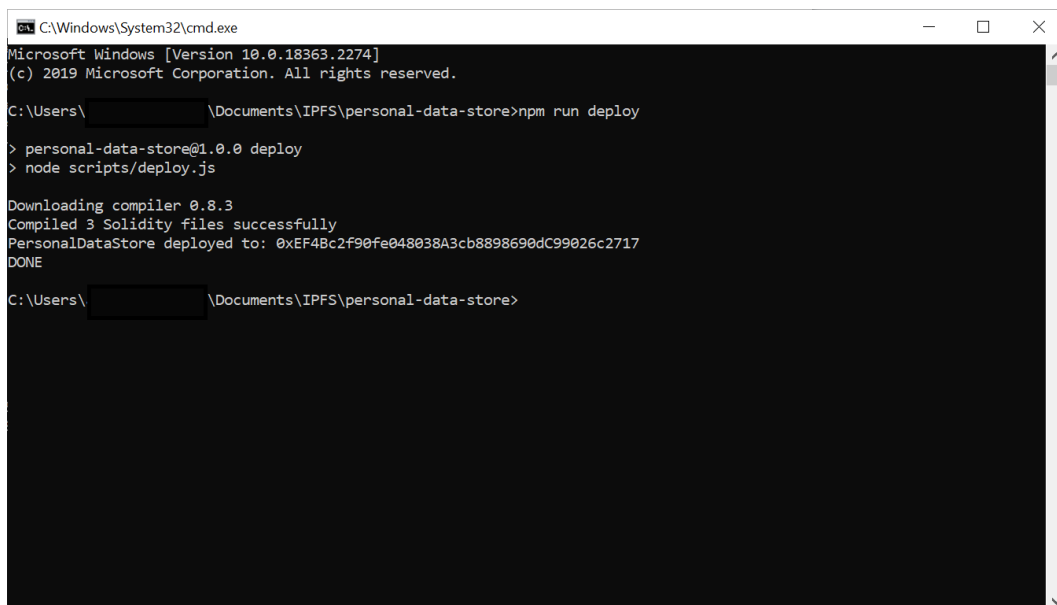
C:\Users\          \Documents\IPFS\personal-data-store>npm run ganache

> personal-data-store@1.0.0 ganache
> node scripts/ganache.js

----- accounts -----
0xfc1256764bfa310cfc9237c92255cfaea6bee3c82c6a4ed99a432fa23512292
0x2b19d9bbf41ec0be2d711a701b5a2c4f0a753697ae9b2e537c96e893ebd25e17e
0x378930b604bb669052cc3e025fd17510ec30b3d12acbdd116ca7faa52a7078cb
0xd943337f7e6dc5a1a869e42df6f63c505905e38c9a95cdc62837e87226fb0c6
0x76a32eecbead60d084257d90e7014d1e97d1ddf9fbb4936906595110c93d4921
0x93bf5de9fcddd919373504452ae46560547615abbb5726f551de19ab3831f01
0xe472630480507510e2e26e858677b0cd3918095ee72426d2e66616bb9a306283
0xb37381c848e7454136c0e9b0706cadea8862c6db794799c7cc99ec2510c7b986
0xd7331ac645adff46bf579a9d4b6eda12e3a2200487a6c6ec1608b644a70b5053
0xe235927e3c9c55c0865cb0cc09f4c9fcd8f009b44c39ec70b2a49a7fee53b5e1
-----

Ganache blockchain started
```

Figure 8.3 : Running of Ganache blockchain



```
C:\Windows\System32\cmd.exe
Microsoft Windows [Version 10.0.18363.2274]
(c) 2019 Microsoft Corporation. All rights reserved.

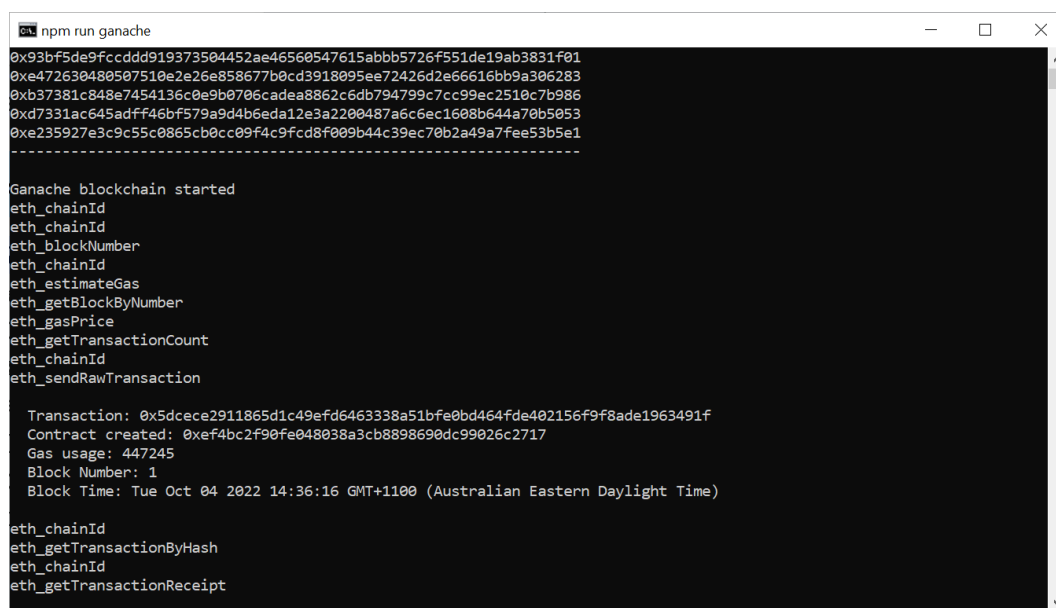
C:\Users\          \Documents\IPFS\personal-data-store>npm run deploy

> personal-data-store@1.0.0 deploy
> node scripts/deploy.js

Downloading compiler 0.8.3
Compiled 3 Solidity files successfully
PersonalDataStore deployed to: 0xEF48c2f90fe048038A3cb8898690dc99026c2717
DONE

C:\Users\          \Documents\IPFS\personal-data-store>
```

Figure 8.4 : Compiling the smart contract for objective 1



```

npm run ganache
0x93bf5da9Fccddd919373504452ae46560547615abb5726f551de19ab3831f01
0xe472630480507510e2e26e858677b0cd3918095ee72426d2e66616bb9a306283
0xb37381c848e7454136c0e9b0706cadea8862c6db794799c7cc99ec2510c7b986
0xd7331ac645adff46bf579a9d4b6eda12e3a2200487a6c6ec1608b644a70b5053
0xe235927e3c9c55c0865cb0cc09f4c9fcd8f009b44c39ec70b2a49a7fee53b5e1
-----
Ganache blockchain started
eth_chainId
eth_chainId
eth_blockNumber
eth_chainId
eth_estimateGas
eth_getBlockByNumber
eth_gasPrice
eth_getTransactionCount
eth_chainId
eth_sendRawTransaction

Transaction: 0x5dcece2911865d1c49efd6463338a51bfe0bd464fde402156f9f8ade1963491f
Contract created: 0xef4bc2f90fe048038a3cb8898690dc99026c2717
Gas usage: 447245
Block Number: 1
Block Time: Tue Oct 04 2022 14:36:16 GMT+1100 (Australian Eastern Daylight Time)

eth_chainId
eth_getTransactionByHash
eth_chainId
eth_getTransactionReceipt

```

Figure 8.5 : Smart contract deployment for objective 1

with blockchain via the browser and facilitate the signing of a blockchain transaction in a secure manner. DApp can communicate with the Ethereum blockchain through the use of MetaMask, which functions as a bridge.

As shown in Figure 8.6, the Ethereum account was connected successfully to the Ganache blockchain.

DApp was developed as a web page utilising React to interact with the smart contract. React offers an interactive user interface, enabling users to communicate with the blockchain. We run DApp using NPM which opens the browser window connected to localhost:3000. The React web page opens on the browser connected to *localhost* or *127.0.0.1*. Figure 8.7 depicts the graphical user interface (GUI) of the prototype.

We set up the DApp settings using the Infura framework. Settings are required before taking any action. Infura is used as a gateway that allows access to IPFS, which is where the file are kept. The Infura endpoint is specified during the settings process. Infura endpoint is where we can connect to IPFS to upload and download

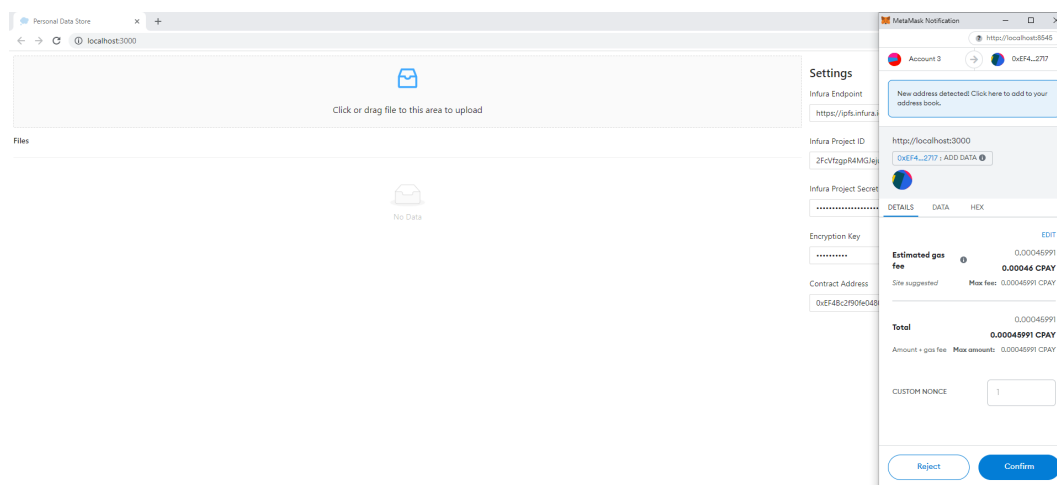


Figure 8.6 : Connecting Ethereum account to Ganache blockchain

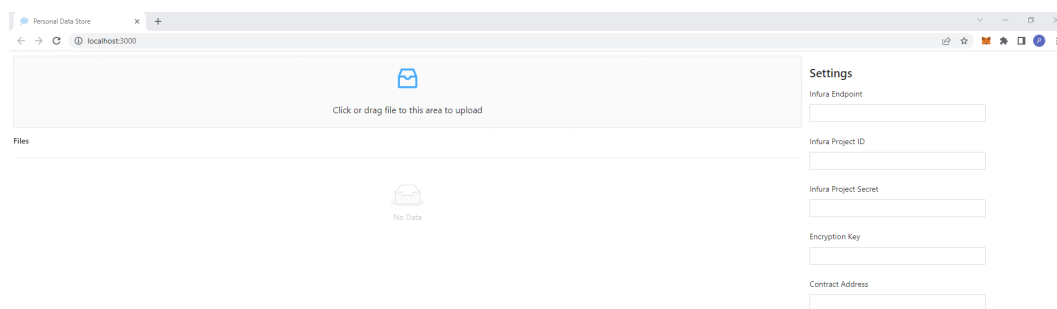


Figure 8.7 : The DApp interface for objective 1

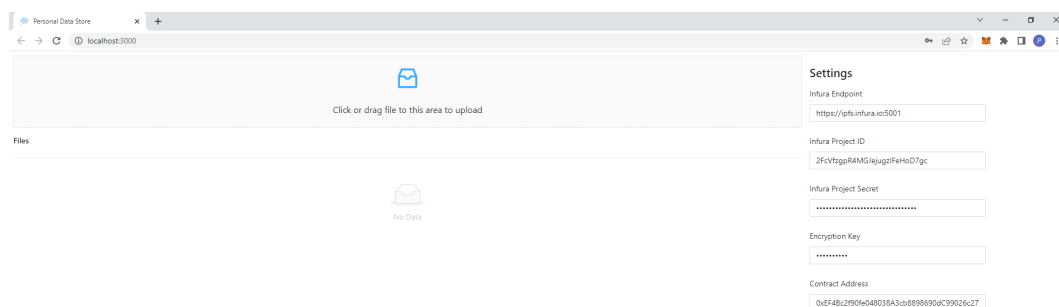


Figure 8.8 : Settings of the DApp GUI

files. IPFS is used as distributed storage to store the uploaded files to minimize the quantity of information that is kept on blockchain. The encryption key is utilized for the encryption and decryption files that are uploaded and downloaded from IPFS and the data that is extracted from the deployed contracts.

The following are the values that we defined for the settings, as shown in Figure 8.8:

- ◇ Infura Endpoint: `https://ipfs.infura.io:5001`
- ◇ Infura Project ID: The project id from Infura.
- ◇ Infura Project Secret: The project secret from Infura.
- ◇ Encryption Key: Chosen by the user.
- ◇ Contract Address: `0xEF4Bc2f9fe048038A3cb8898690dC99026c2717`

Once the DApp settings have been configured, we begin uploading files to the DApp.

A smart contract is designed to allow only the contract owner to interact with the contract and upload files by establishing access restrictions based on the public address. All the files and blockchain-stored information are encrypted using the AES algorithm. Before storing the files on IPFS, DApp performs the encryption process. After selecting a file for uploading, the encryption key is used to encrypt it. Then, a request is made to store the encrypted content on IPFS to Infura. After the file has been stored, IPFS provides a content identifier (CID) that can be used to access and retrieve the file. The resulting encrypted string includes CID, file name, and timestamp by invoking the "addData" function to make a transaction request. Once the transaction has been successfully confirmed, a request to update the file list is made. To extract the existing uploaded data from the smart contract, we obtain the total number of files uploaded by calling the function "filesCount". To retrieve the encrypted string, we use the hash by calling "dataFilesIndex". After the encrypted string has been retrieved, the string is decrypted using the encryption key. Then, we can obtain the link to download the data from IPFS. By clicking on the link of the uploaded file, a request is submitted to IPFS to download the stored data. Then, the data can be decoded utilizing the encryption key and the file can be saved locally after the decryption process.

AES encryption is employed to encrypt the files due to its functionality. IPFS generates the hash of the file that is being recorded, and then the file can be accessed by that hash. Then, the hash is encrypted utilizing the AES technique before being stored on the blockchain. The AES encryption produces a hash value, which is then recorded on the blockchain. Figure 8.9 depicts the file that was stored on the IPFS, showing details about the recorded file, such as the file name, the timestamp, and the transaction hash.

The data can be extracted from blockchain using the transaction hash to initiate

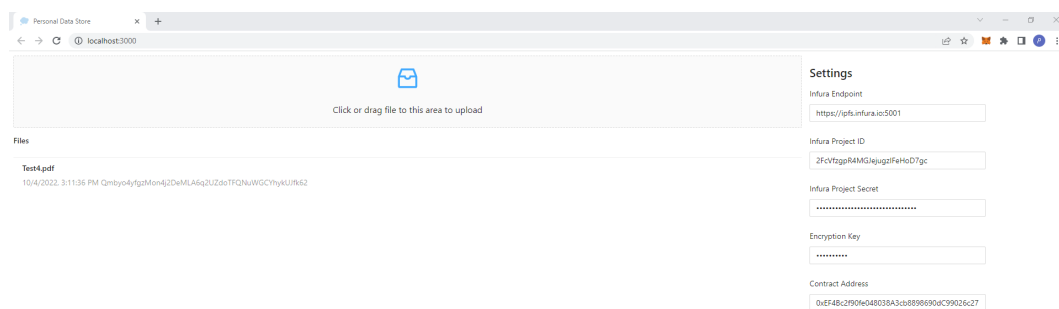


Figure 8.9 : The file stored on IPFS

the download. Then, the data is decrypted using AES to obtain the IPFS hash. Subsequently, the file is retrieved from the IPFS using the hash that was decrypted. The downloaded file is then decrypted using AES by the encryption key which is owned by the user who encrypted the file.

In this section, we described the working of the DApp to manage identities over the blockchain. In the next section, we explain the working of the DApp in relation to generating warnings as a reminder for users of their impending identity expiration.

8.3 Prototype for Generating Warnings for Users

This section presents screenshots to demonstrate the working of the DApp for generating alerts when a user's identity is about to expire. Figure 8.10 illustrates how the components of DApp interact with each other. We use Remix IDE to write and compile the smart contract. React is used to develop the DApp for this work. The screenshot in Figure 8.11 shows the DApp interface which allows service

providers to enter a new user's identity information. The service provider inserts a new user's identity through the frontend which includes information such as name, identity no, expiry date, and email.

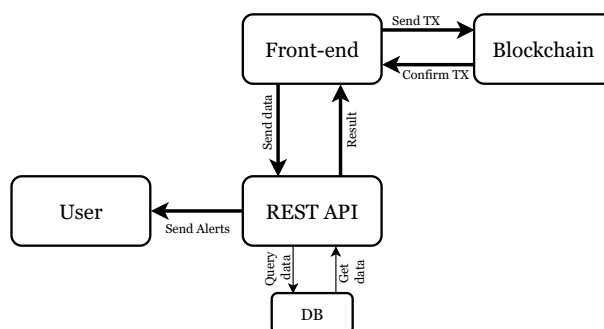


Figure 8.10 : The proposed model for generating warnings

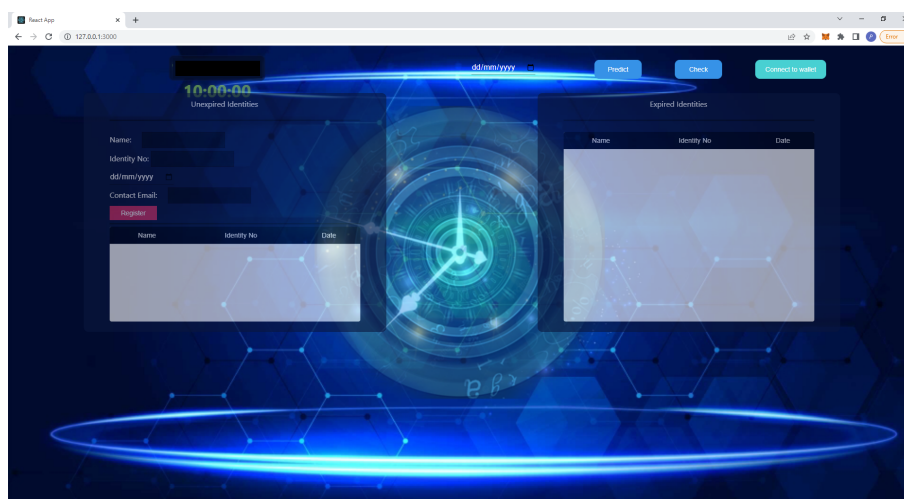
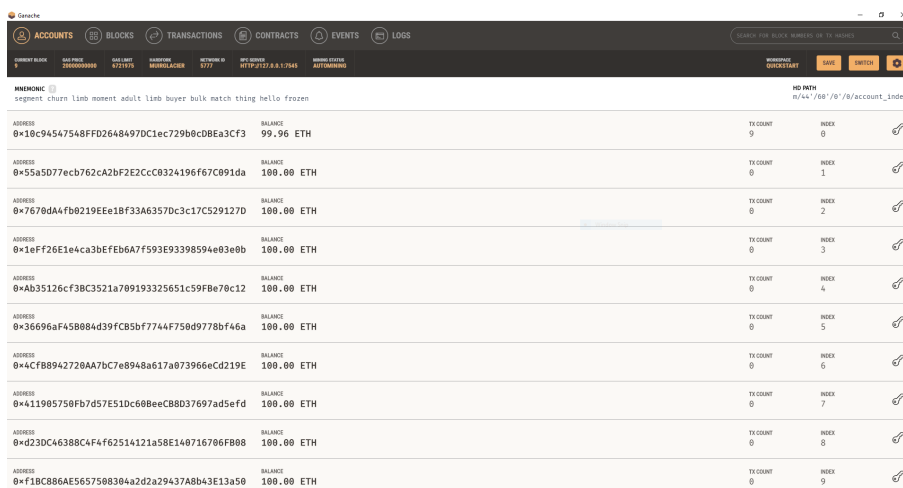


Figure 8.11 : The main interface for DApp for objective 2

Figure 8.12 depicts the main interface of Ganache which display 10 accounts with 100 ETH for testing purposes. We need to create a new account through MetaMask by importing an account from Ganache to MetaMask using the private key of the Ganache account. Figure 8.13 shows the account that has been successfully created in MetaMask with the balance available. Then, we need to create a network using the following settings:



ADDRESS	BALANCE	TX COUNT	INDEX
0x18c94547548FFD2648497DC1ec729b0cD8Ea3Cf3	99.96 ETH	0	0
0x55a5d77ecb762cA2bF2E2Cc0324196f67C091da	100.00 ETH	0	1
0x7679dA4fb0219Ee18f33A6357Dc3c17C529127D	100.00 ETH	0	2
0x1eFf26E1e4ca3bEfeb6A7f593E93398594e03e0b	100.00 ETH	0	3
0xAb35126cF3BC3521a789193325651c59F8e78c12	100.00 ETH	0	4
0x38696aF458884d39fCB5f7744f758d9778bf46a	100.00 ETH	0	5
0x4cFB8942726AA7bC7e8948a617a073966cD219E	100.00 ETH	0	6
0x411985750Fb7d57E51Dc680eCB8037697ad5efd	100.00 ETH	0	7
0xd23DC46388C4F4f6251421a58E140716706F808	100.00 ETH	0	8
0xf1BC886AE5657588304a2d2a29437A8b43E13a50	100.00 ETH	0	9

Figure 8.12 : Ganache main interface for objective 2

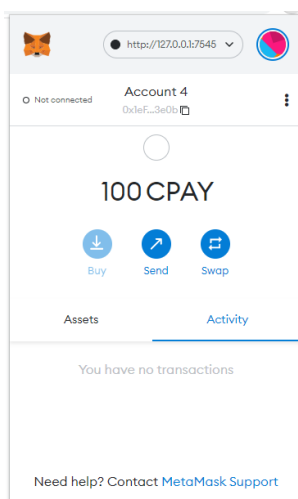


Figure 8.13 : The new account created through MetaMask

- ◇ New RPC URL : `http://127.0.0.1:7545` Or `localhost:7545`
- ◇ Chain ID: 1337
- ◇ Currency symbol: CPAY

The next step is to write the smart contract on Remix IDE and then, compile the smart contract using *Web3 Provider* as the environment. Once the smart contract is compiled, the *Web3 Provider Endpoint* is set to `http://127.0.0.1:7545` as depicted in

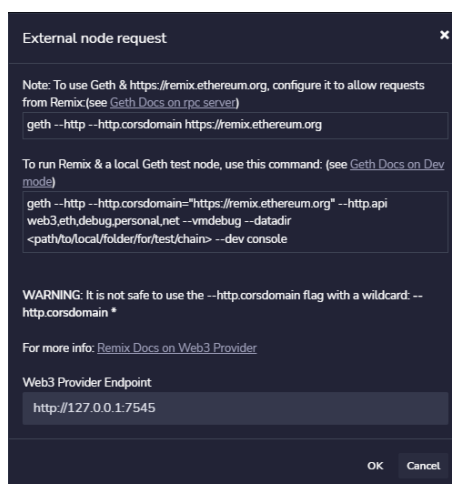


Figure 8.14 : The endpoint settings

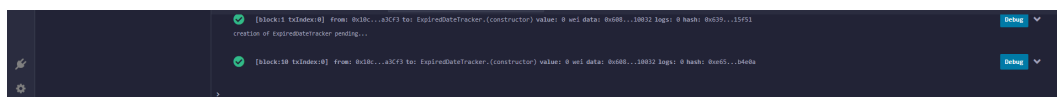


Figure 8.15 : Deployment of smart contract on Remix for objective 2

Figure 8.14. Then, the deployment of the smart contract is carried out on Remix to obtain the contract address, after which the smart contract is successfully deployed, as shown in Figure 8.15. After deploying the smart contract in Remix, it can be seen in Ganache, as depicted in Figure 8.16. The transaction consumes a small amount of Ether as a transaction fee which is offered by the Ganache test account. Figures 8.15 and 8.16 show that the transaction hash for the generation of the smart contract is identical, indicating that the deployment of the smart contract was successful.

The REST-API is designed using the Flask framework and based on the Python programming language which saves data in the local database. There are multiple frameworks that can be used to build a REST-API, however for this project, we chose the Flask framework due to its popularity and robustness to build the API server. We develop the REST-API using Flask framework in Python to ensure real-time data streaming. The REST-API is used to set up a local API endpoint to collect

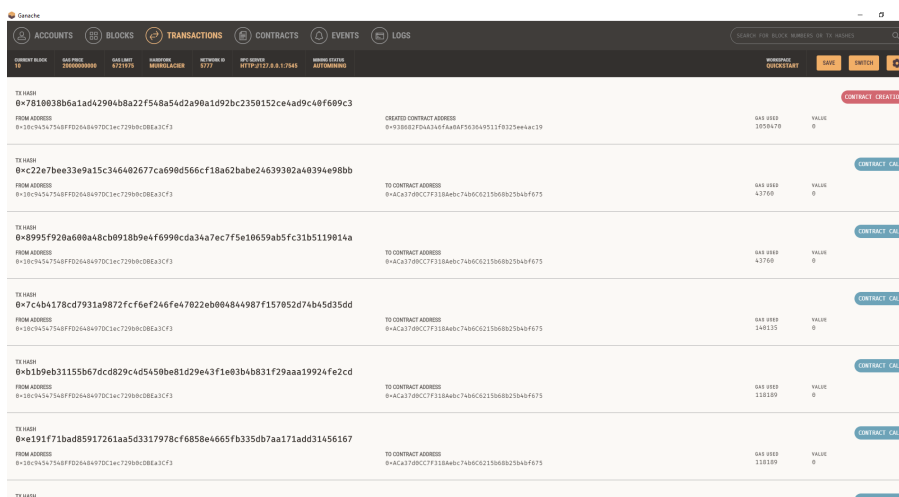


Figure 8.16 : Smart contract confirmation of deployment on Ganach

```
def transaction_check():
    result = db_query('SELECT * FROM transactions WHERE email_sent=0')
    # print(result)
    for item in result:
        current_time = datetime.now()
        record_time = datetime.fromtimestamp(item['date'])
        current_id = item['id']

        email = item['email']
        name = item['name']
        identity_no = item['identity_no']

        if record_time < current_time + timedelta(days=EMAIL_NOTICE_DATE):
            record_time_formatted = record_time.strftime("%Y-%m-%d")
            res = send_email(email, name, identity_no, record_time_formatted)

            if res.status_code == 200:
                sql = 'UPDATE transactions SET email_sent=? WHERE id=?'
                val = (1, current_id)
                result = db_query_with_val(sql, val, commit=True)
                print(datetime.now(), 'email sent!')
            else:
                # print(datetime.now(), 'error when sending email.')
                # print(res.content)
                pass

    if not result:
        print(datetime.now(), 'no pending records.')
```

Figure 8.17 : The algorithm used to check expiry dates

the user's input from the front-end. All newly created records are sent via API to this server. In addition, it monitors and checks if any existing records in the database are about to expire and if so, sends email alerts via Mailgun API. The database is used to store information such as the expiration date and email address, which are used for operational use only and are necessary for processing email notifications. The key information is maintained on blockchain which is verified. Figure 8.17 presents the algorithm that is used to check if the expiry date is imminent.

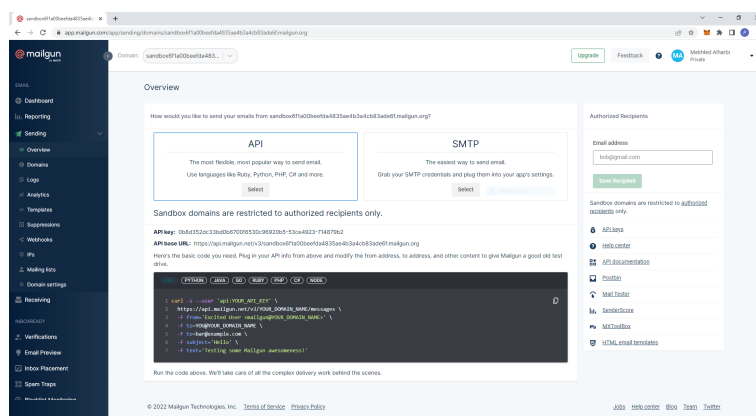


Figure 8.18 : The Mailgun account details

```

11 def send_email(email_name, identity_no, date):
12     return requests.post(
13         "https://api.mailgun.net/v3/sandboxf1a0bbe6d4833ae6b3a4c833d6f5.mailgun.org/messages",
14         auth=("api", "0bbe6d4833ae6b3a4c833d6f5:530e4923-714879b2"),

```

Figure 8.19 : The API settings

Users are able to communicate by sending and receiving emails using Mailgun. After creating an account on Mailgun, we connect Mailgun with the REST API using the domain name and the private API key of the Mailgun account. Figures 8.18 and 8.19 illustrate the connection between the Mailgun service account and REST API.

Then, REST API is run to enter the user identity information such as name, identity No, expiry date, and email. After adding all the details, we click on "Register" button to invoke the MetaMask interface. Then, the information goes through the confirmation process to register the transaction. Figure 8.20 presents the process of adding the user's details. Subsequently, the information is recorded on the blockchain after confirming the transaction. Figure 8.21 shows the transaction confirmation process and Figure 8.22 depicts the wallet balance after confirming the transaction. Once the transaction is confirmed, a notification is sent to the user and is received by the registered email, as shown in Figure 8.23.

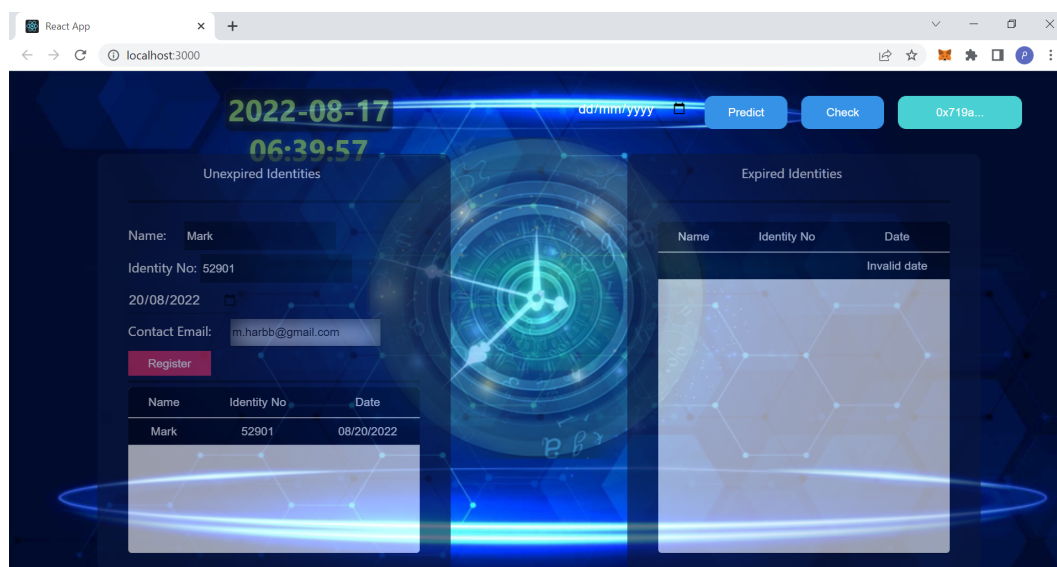


Figure 8.20 : Adding users' information through DApp for objective 2

8.4 Prototype for Computing the Trustworthiness of the Users' Identities

This section details how the DApp functions for computing the trustworthiness score of users' identities. In this work, we use React Bootstrap, web3, Solidity and the MetaMask Ethereum wallet. We use Remix IDE to write and compile the smart contract using the Solidity language. Then, we deployed the smart contract on Ropsten test net. We then created an Ethereum account on MetaMask to interact with the Ropsten test network. The MetaMask wallet allows us to manage personal accounts as well as the Ether funds that we need to pay for transactions. Figure 8.24 presents the account that has been created and the process of connecting the account to the Ropsten network.

As shown in the Figure 8.24, the Ether is available in the account wallet which is necessary to execute the transactions through blockchain. We got Ether through the MetaMask Ether Faucet page to request Ether, as shown in Figure 8.25. As shown in Figure 8.25, we requested many Ethers and successfully added them to the wallet.

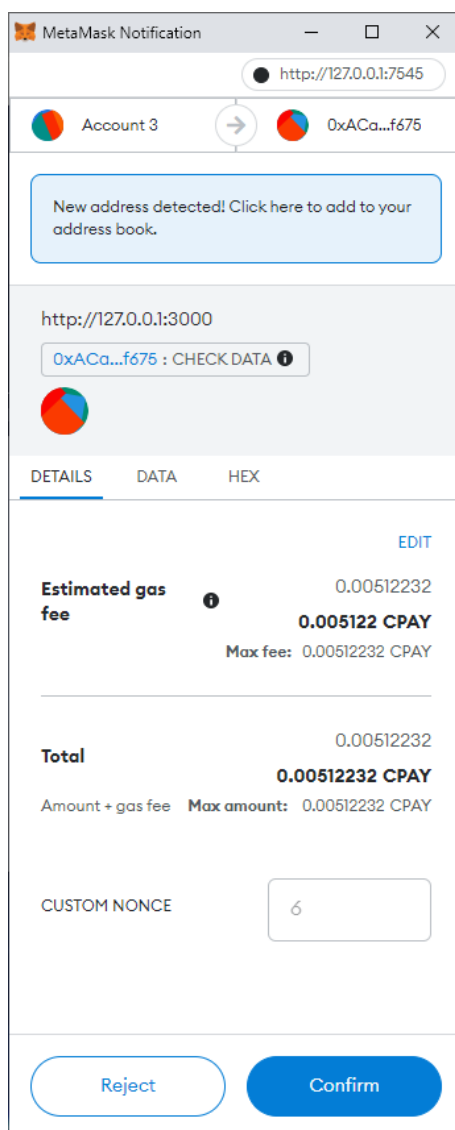


Figure 8.21 : Confirmation request of the transaction for objective 2

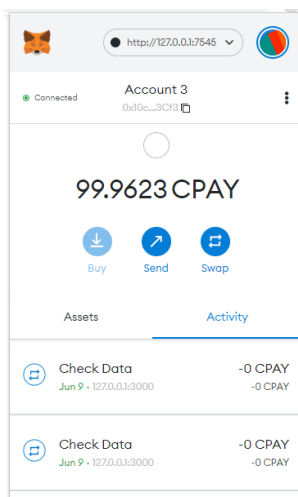


Figure 8.22 : The wallet balance after transaction confirmation

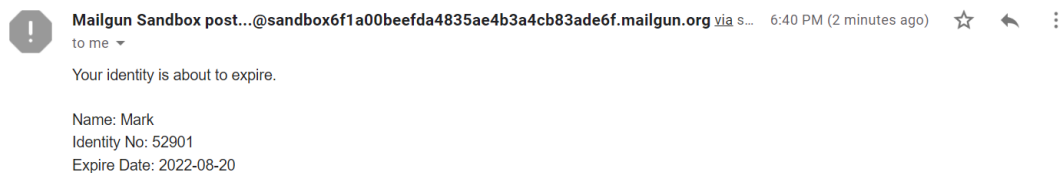


Figure 8.23 : Notification advising that the email was received successfully

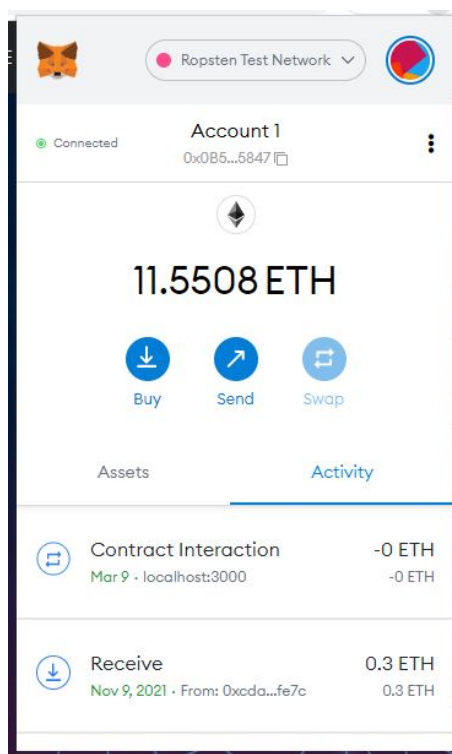


Figure 8.24 : Connecting to the MetaMask account

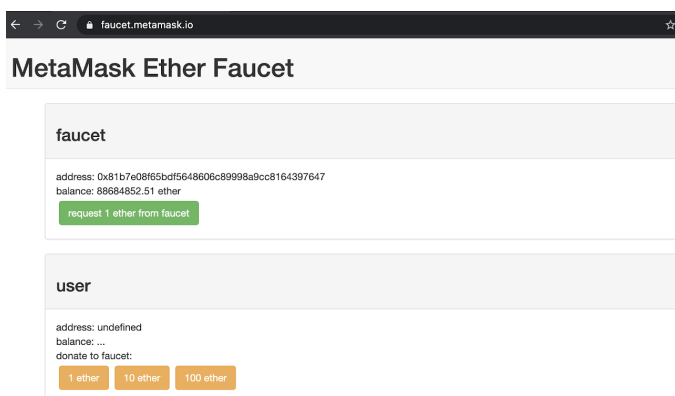


Figure 8.25 : MetaMask Ether Faucet

React is used to build the interactive user interface to calculate the trustworthiness score of the user.

Next, we installed the Node Package Manager (NPM) which comes with Node.js. After we successfully connected the Ethereum account, we ran the web server using

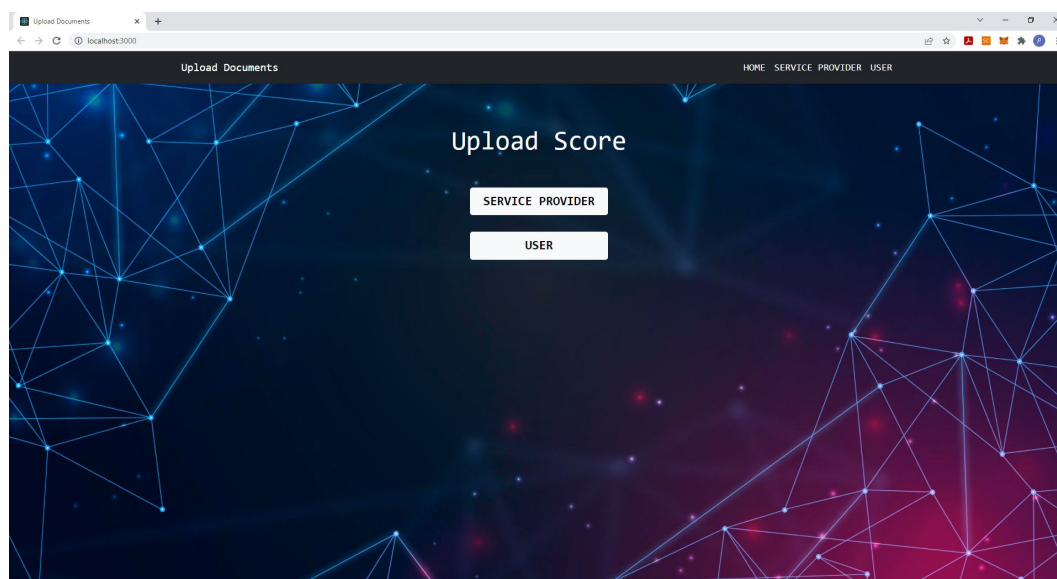


Figure 8.26 : Main page for DApp for objective 3

NPM to launch the DApp. The Chrome browser opens automatically on either localhost or 127.0.0.1 to open the home page of DApp. The main page of the prototype contains two options, Service Provider and User, as shown in Figure 8.26.

Service provider is responsible for uploading the user's documents and submitting these to blockchain while the user can view the obtained score based on the provided documents. Figure 8.27 illustrates the process of uploading the documents which includes a drop down list of different documents as suggested in section 7.1. During the uploading process, the service provider selects the document type corresponding to the uploaded document. Based on the document type, the weight value for the document is retrieved. The weight for each document is assigned and is stored in DApp, as shown in Figure 8.28.

After uploading the user's documents that have been provided, we click on the "submit" option. When we click "submit", the "upload" function is called from the smart contract to calculate the overall trustworthiness score. Then, we go through the confirmation process to register the transaction through the Ropsten network.

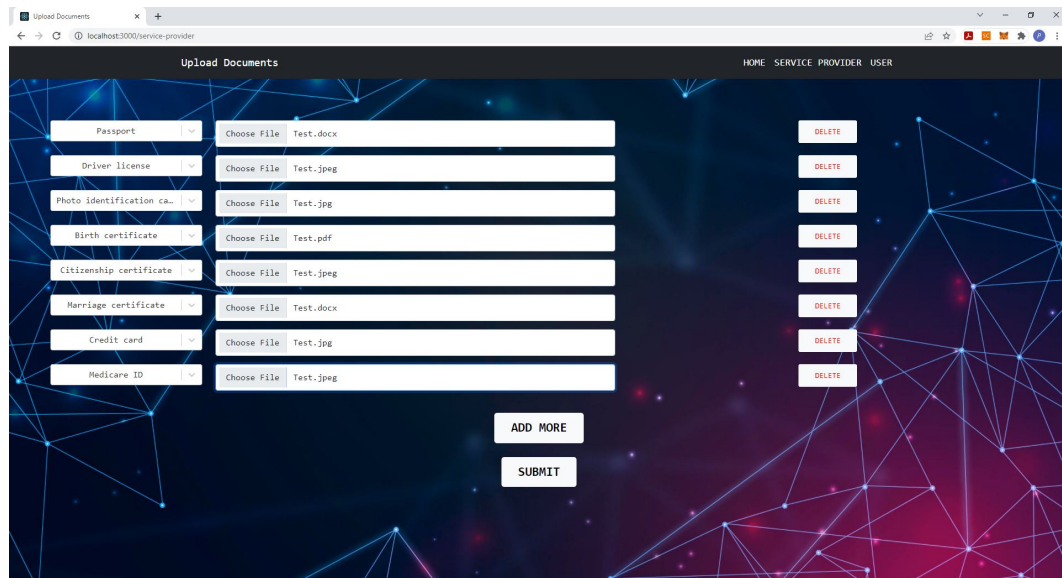


Figure 8.27 : The process of uploading documents to DApp

```

1  const Action_Type = {UPLOAD:'upload',}
2  const Status_Type = {PENDING:'pending',SUCCESS:'success', ERROR: 'error'}
3  const Docs_Score_Map = {"doc1":70,"doc2":70,
4                          "doc3":70,"doc4":40,
5                          "doc5":40,"doc6":40,
6                          "doc7":40,"doc8":40,
7                          "doc9":25,"doc10":25
8                          }

```

Figure 8.28 : The weight value for the documents

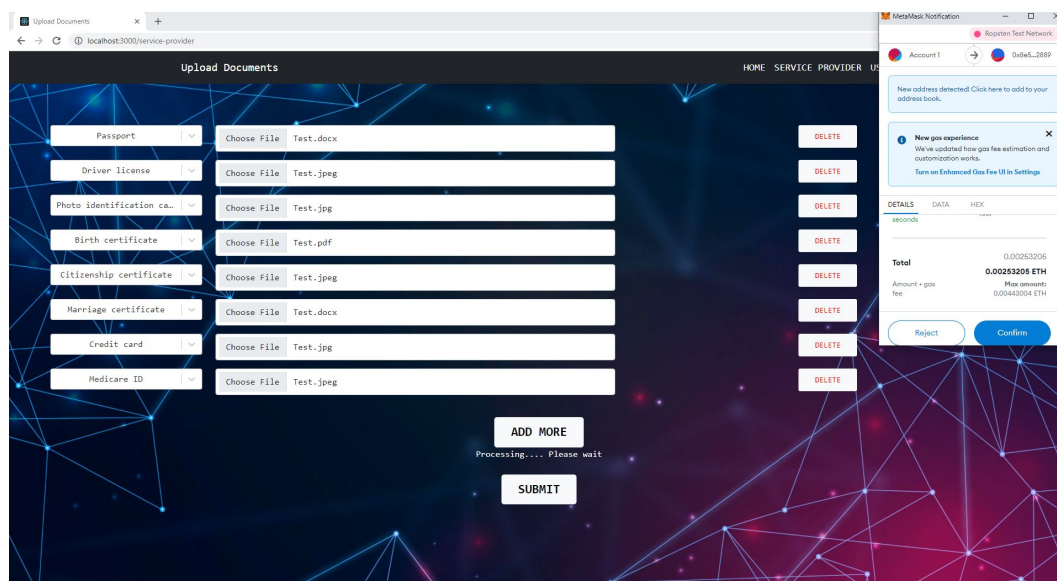


Figure 8.29 : Request to confirm the transaction for objective 3

After confirming the transaction, it is recorded on the blockchain, as shown in Figure 8.29.

Once the transaction is confirmed, the algorithm is executed to calculate the overall trustworthiness score. Figure 8.30 presents the algorithm for calculating the trustworthiness score. Then, the user is redirected to the "user" page where the calculated score is called by the "getScore" function from the smart contract. The overall trustworthiness score of the user is computed and reflected on the user's page. The user obtains the trustworthiness score based on the provided documents. Then, the user's score is published and stored in the blockchain. Figure 8.31 shows the user's trustworthiness score.

Only the user who built the smart contract is able to upload the documents so the user can be recognized by their smart contract address. We run the prototype using NPM to start the server and open the prototype on 3000 port on localhost. Recording the trustworthiness score of user's identity on blockchain ensures that the trustworthiness score can never be modified due to the immutability of blockchain

```

37 function upload(DocInfo[] memory inputDocs) public {
38     uint256 usrscore = 0;
39     uint256 score = 0;
40     bool isUpdate = false;
41     for(uint i = 0; i < inputDocs.length; i++){
42         score = score + inputDocs[i].score;
43     }
44     usrscore = score * 100 / totalScore;
45     for(uint256 i = 0; i < userScores.length; i++){
46         if(userScores[i].id == msg.sender){
47             userScores[i].score = usrscore;
48             isUpdate = true;
49         }
50     }
51     if(!isUpdate){
52         userScores.push(ScoreStruct(msg.sender, usrscore));
53     }
54 }
55

```

Figure 8.30 : Algorithm for computing the trustworthiness score

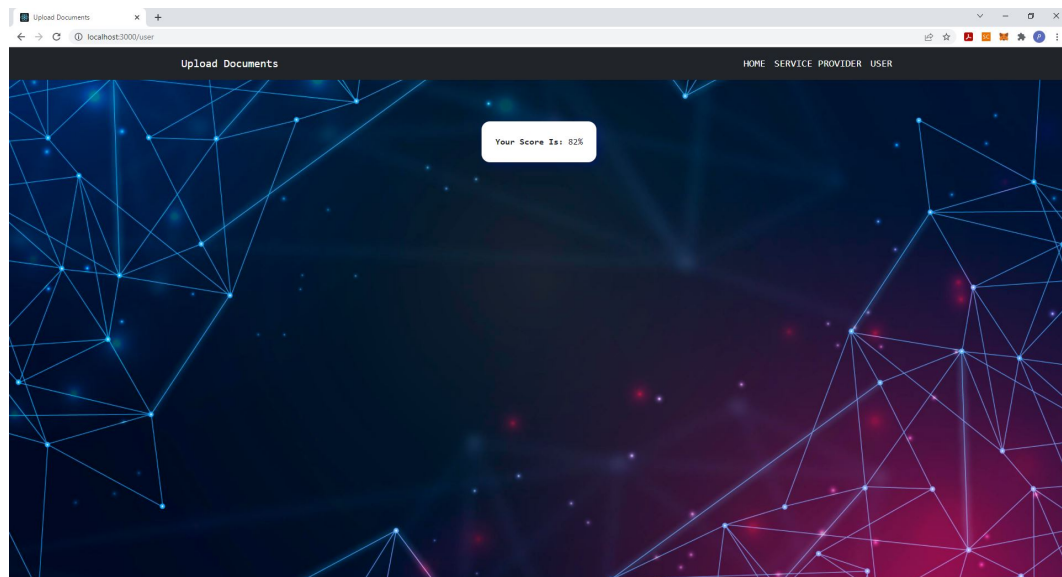


Figure 8.31 : The calculated score on the user's page for objective 3

records, except by majority consensus, hence, enhancing the service providers' trust in the trustworthiness score. The scores are reliable because the service provider is only allowed to submit user identity documents. Therefore, storing the trustworthiness score on blockchain strengthens trust between users and service providers and establishes a trustworthy relationship between them.

8.5 Conclusion

This chapter details the functionality of the prototypes that were built to achieve the objectives of this thesis. We explained the methodology and functioning of these prototypes and we provided a thorough explanation of the prototype configuration, which encompassed both the DApp and blockchain setups.

The next chapter concludes the thesis and provides suggestions for future research work.

Chapter 9

Conclusion and Future Work

9.1 Introduction

This chapter concludes the thesis by presenting a synopsis of the research results and making several recommendations for future research directions. This thesis constitutes a pioneering effort in leveraging smart contracts for identity management on blockchain. This is made clear in Chapter 2, which presents the results of the systematic literature review and a comprehensive study of prior work. Research gaps were identified as a result of the literature review, and solutions were developed to fill the gaps. This thesis proposes intelligent identity management methods that can enhance user's identity development.

9.2 Problems Addressed in this Thesis

The primary aim of this thesis is to address significant gaps in the existing literature concerning the intelligent management of user identity using blockchain-based smart contracts. The following research issues were determined in light of the literature review conducted in Chapter 2 and were then addressed in the thesis:

1. None of the existing literature has integrated blockchain-based smart contracts and identity management using artificial intelligence techniques to detect duplicate user identities on top of blockchain and then ensure privacy for these identities.
2. None of the existing literature has set a personalized early warning system to

identify user identities that are about to expire in order to renew them and obtain the benefit of the desired services.

3. None of the existing literature takes into account how to compute the trustworthiness of a user's identity based on a single or multiple documents.

9.3 Contributions of this thesis to the existing literature

This thesis makes a significant contribution to the existing body of literature in regard to the research issues that have been highlighted by proposing intelligent approaches for identity management based on blockchain. The following is a condensed summary of the research contributions of this thesis to address the gaps in the current literature:

9.3.1 Contribution 1: Systematic Literature Review

In this thesis, a comprehensive and methodical survey of the existing body of literature in the realm of blockchain technology, smart contracts, and identity management was carried out, which can be found in Chapter 2. Particular search terms for the SLR were queried using five databases, namely Elsevier ScienceDirect, IEEE Xplore, SpringerLink, ACM Digital Library, and Google Scholar. The findings obtained from the search were evaluated for their relevance and also in terms of whether they satisfied certain inclusion and exclusion criteria. A total of 20 publications that were pertinent to the study were identified and evaluated. The findings of the SLR revealed that the current literature lacks methods that enable the intelligent management of a user's identity for blockchain. To the best of the researcher's knowledge, no systematic review of the existing literature has been conducted in this area. The current literature review was classified into the following groups according to the technical issue that they are attempting to address: (1) Authentication. (2) Privacy. (3) Trust. As a result of the systematic literature review, research gaps and

questions were identified and formulated. A paper that summarises the findings of the systematic literature review has been published.

9.3.2 Contribution 2: A framework for the integration of blockchain and machine learning methods for identity management

As discussed in the previous chapters, the current literature suggests methods to manage users' identities based on blockchain, however in the current literature, no solution has been proposed after a duplication has been detected. Consequently, this thesis advocates the necessity for a comprehensive framework for the integration of machine learning and blockchain for identity management to achieve both effective performance and an immutable storage mechanism. Integrating blockchain and IPFS allows for distributed data management and storage without compromising a user's privacy. In light of this, Chapter 5 presents a comprehensive framework to detect duplicate user identities and preserve data privacy by applying machine learning techniques and blockchain technology.

To the best of the researcher's knowledge, the need for such a comprehensive framework has not been explored in the literature. The outcomes of this work have been submitted to a peer-reviewed journal.

9.3.3 Contribution 3: Intelligent model for generating warnings about the expiration of a user's identity

An EWS date-based system was developed to identify the imminent expiration of user identities. The information pertaining to a user's identity, such as user name, identity number, expiration date, and email address, is stored in blockchain. Since the user identity information that has been recorded on the blockchain cannot be modified once it has been recorded, this system is trustworthy. Furthermore, the service provider is only permitted to enter the users' details and activate the system. The system is intelligent because it generates notifications automatically.

The proposed algorithm is used to determine the impending expiration date of an identity. In addition, the system uses the REST-API framework to identify the expired user identities on the fly. Ganache personal blockchain is utilized to achieve this research objective.

9.3.4 Contribution 4: Intelligent model for determining the current trustworthiness score of a user based on the user identities stored on blockchain

The Ropsten Ethereum testnet is utilised to achieve the research objective. The proposed algorithm is utilised to compute the overall user identity trustworthiness score of the user based on the provided identities. The smart contract is designed to calculate the overall value and then publish it on the blockchain. This model is intelligent because it compiles the trustworthiness of a user and employs an algorithm to calculate the trustworthiness value of that user by utilizing the weights that are allocated to each user identity. Since users have provided their identities and only the service provider is allowed to add the identities of users to the blockchain, this ensures the trustworthiness value is reliable. The blockchain stores the trustworthiness score of a user and is characterized by immutable records, which implies that the trustworthiness values can never be changed once they have been recorded.

9.3.5 Contribution 5: Implementation and evaluation of the proposed solutions

This thesis employed software prototypes as a means of evaluating the effectiveness of the proposed models. The model described in Chapter 5 was evaluated and tested using the evaluation metrics to measure its performance, and a prototype to maintain information on user identities was developed. Chapter 8 demonstrates the functionality of the prototypes for maintaining user identity information, producing warning notifications and computing the trustworthiness of a user's identity.

In addition, the functioning of the developed prototype that corresponds to each objective is illustrated in Chapters 5 through 7.

9.4 Conclusion and Future Work

In this chapter, we conclude the thesis and provide recommendations for future research. This thesis examines various potential aspects in which user identity applications built on blockchain could become a reliable identity management platform. Additionally, the research aims to develop approaches that incorporate all of the necessary characteristics in a holistic manner. Several papers have been published in peer-reviewed journals and international conferences proceedings as a result of this research. A significant amount of research has been reported in this thesis on leveraging machine learning and smart contracts for blockchain-based identity management, however there are still further challenges that need to be investigated in the future. The following are some of the areas we plan to explore further in our future work:

1. We intend to extend the work by developing a method for automating the process of adding the resultant data to blockchain directly through the use of REST-API. Furthermore, to enhance the suggested model's overall performance, we plan to investigate other deep learning techniques such as GAN and LSTM models and to evaluate the developed model on many datasets. The process of duplicate detection can be enhanced by examining other factors.
2. Developing an EWS model that alerts service providers when a user's identity has expired, hence preventing that user from accessing the desired services. We developed the proposed EWS to be future-proof by including the REST-API, and we can extend its capabilities so that it may be integrated with various email delivery systems.

3. The trustworthiness value of this research study has been determined by only using the user identity documents proposed by the Australian Government as the basis for determining trustworthiness. However, the model has the potential to be improved by including additional factors that influence trustworthiness, such as user behaviours. Subsequently, this study constitutes a foundation for future work in the field, enabling the development of more advanced user trustworthiness models. Additionally, we plan to develop a model using machine learning that can predict the accepted threshold value of the service providers based on the services they are providing, which can then be utilised to provide services to users. Moreover, we will develop a model that enables users to approve or decline access by service providers to their trustworthiness scores which will enhance the level of privacy for the user's data, since the user will be able to identify the entity requesting their information.
4. In the future, we can integrate all the proposed models into one platform and then a commercial system can be constructed using these models. In addition, we will apply the suggested methods in real-world sectors such as real estate, banking, etc.

Bibliography

- Ahmed, M. R., Islam, A. M., Shatabda, S., and Islam, S. (2022). Blockchain-based identity management system and self-sovereign identity ecosystem: A comprehensive survey. *IEEE Access*, 10:113436–113481.
- Ahn, G.-J. and Ko, M. (2007). User-centric privacy management for federated identity management. In *2007 International Conference on Collaborative Computing: Networking, Applications and Worksharing (CollaborateCom 2007)*, pages 187–195. IEEE.
- Alharbi, M. and Hussain, F. K. (2021). Blockchain-based identity management for personal data: A survey. In *International Conference on Broadband and Wireless Computing, Communication and Applications*, pages 167–178. Springer.
- Alharbi, M. and Hussain, F. K. (2022). A systematic literature review of blockchain technology for identity management. In *International Conference on Advanced Information Networking and Applications*, pages 345–359. Springer.
- Alharbi, M., Hussain, F. K., and Hussain, O. K. (2023). A comprehensive identity management framework based on the integration of blockchain and machine learning. In *International Journal of Web and Grid Services*, volume 19, pages 345–359.
- Alsayed Kassem, J., Sayeed, S., Marco-Gisbert, H., Pervez, Z., and Dahal, K.

- (2019). Dns-idm: A blockchain identity management system to secure personal data sharing in a network. *Applied Sciences*, 9(15):2953.
- Andoni, M., Robu, V., Flynn, D., Abram, S., Geach, D., Jenkins, D., McCallum, P., and Peacock, A. (2019). Blockchain technology in the energy sector: A systematic review of challenges and opportunities. *Renewable and Sustainable Energy Reviews*, 100:143–174.
- Atlam, H. F. and Wills, G. B. (2019). Technical aspects of blockchain and IoT. In *Advances in computers*, volume 115, pages 1–39. Elsevier.
- Australian Government (2018). Individual aviation reference numbers. <https://www.casa.gov.au/licences-and-certificates/aviation-reference-numbers/individual-aviation-reference-numbers>. Accessed: 2020-03-09.
- Banerjee, M., Lee, J., and Choo, K.-K. R. (2018). A blockchain future for internet of things security: a position paper. *Digital Communications and Networks*, 4(3):149–160.
- Berg, A., Borensztein, E., and Pattillo, C. (2005). Assessing early warning systems: how have they worked in practice? *IMF staff papers*, 52(3):462–502.
- Birrell, E. and Schneider, F. B. (2013). Federated identity management systems: A privacy-based characterization. *IEEE security & privacy*, 11(5):36–48.
- Buccafurri, F., Lax, G., Russo, A., and Zunino, G. (2018). Integrating digital identity and blockchain. In *OTM Confederated International Conferences” On the Move to Meaningful Internet Systems”*, pages 568–585. Springer.
- Buzzard, J. and Kitten, T. (2021). Identity fraud study: Shifting angles. <https://javelinstrategy.com/research/2021-identity-fraud-study-shifting-angles/>.

- Casino, F., Dasaklis, T. K., and Patsakis, C. (2019). A systematic literature review of blockchain-based applications: Current status, classification and open issues. *Telematics and informatics*, 36:55–81.
- Chalaemwongwan, N. and Kurutach, W. (2018). A practical national digital id framework on blockchain (nidbc). In *2018 15th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON)*, pages 497–500. IEEE.
- Chaves, J. M. and De Cola, T. (2017). Public warning applications: Requirements and examples. In *Wireless Public Safety Networks 3*, pages 1–18. Elsevier.
- Chen, R., Shu, F., Huang, S., Huang, L., Liu, H., Liu, J., and Lei, K. (2021). Bidm: A blockchain-enabled cross-domain identity management system. *Journal of Communications and Information Networks*, 6(1):44–58.
- Christen, P. (2011). A survey of indexing techniques for scalable record linkage and deduplication. *IEEE transactions on knowledge and data engineering*, 24(9):1537–1555.
- Christen, P. (2012). *Data matching: concepts and techniques for record linkage, entity resolution, and duplicate detection*. Springer: Data-centric systems and applications.
- Christidis, K. and Devetsikiotis, M. (2016). Blockchains and smart contracts for the internet of things. *Ieee Access*, 4:2292–2303.
- Crompton, M. and McKenzie, R. (2010). Current issues and solutions in identity management. *Information Integrity Solutions*, pages 2010–10.
- Dellarocas, C. (2001). Building trust on-line: the design of reliable reputation reporting building trust on-line: the design or reliable reputation reporting.

- Domingo, A. I. S. and Enríquez, Á. M. (2018). Digital identity: the current state of affairs. *BBVA Research*, pages 1–46.
- Dong, X. L. and Rekatsinas, T. (2018). Data integration and machine learning: A natural synergy. In *Proceedings of the 2018 international conference on management of data*, pages 1645–1650.
- Dorri, A., Kanhere, S. S., Jurdak, R., and Gauravaram, P. (2017). Blockchain for iot security and privacy: The case study of a smart home. In *2017 IEEE international conference on pervasive computing and communications workshops (PerCom workshops)*, pages 618–623. IEEE.
- Dris, A. B., Alzakari, N., and Kurdi, H. (2019). A systematic approach to identify an appropriate classifier for limited-sized data sets. In *2019 International Symposium on Networks, Computers and Communications (ISNCC)*, pages 1–6. IEEE.
- Dybå, T. and Dingsøy, T. (2008). Empirical studies of agile software development: A systematic review. *Information and software technology*, 50(9-10):833–859.
- Ebraheem, M., Thirumuruganathan, S., Joty, S., Ouzzani, M., and Tang, N. (2018). Distributed representations of tuples for entity resolution. *Proceedings of the VLDB Endowment*, 11(11):1454–1467.
- El Haddouti, S. and El Kettani, M. D. E.-C. (2019). Analysis of identity management systems using blockchain technology. In *2019 International Conference on Advanced Communication Technologies and Networking (CommNet)*, pages 1–7. IEEE.
- Faber, B., Michelet, G. C., Weidmann, N., Mukkamala, R. R., and Vatrappu, R. (2019). Bpdims: A blockchain-based personal data and identity management system.

- Ferdous, M. S. and Poet, R. (2012). A comparative analysis of identity management systems. In *2012 International Conference on High Performance Computing & Simulation (HPCS)*, pages 454–461. IEEE.
- Gefen, D. (2002). Reflections on the dimensions of trust and trustworthiness among online consumers. *ACM SIGMIS Database: the DATABASE for Advances in Information Systems*, 33(3):38–53.
- Ghaffari, F., Gilani, K., Bertin, E., and Crespi, N. (2022). Identity and access management using distributed ledger technology: A survey. *International Journal of Network Management*, 32(2):e2180.
- Hammudoglu, J., Sparreboom, J., Rauhamaa, J., Faber, J., Guerchi, L., Samiotis, I. P., Rao, S., and Pouwelse, J. A. (2017). Portable trust: biometric-based authentication and blockchain storage for self-sovereign identity systems. *arXiv preprint arXiv:1706.03744*.
- Hansen, M., Berlich, P., Camenisch, J., Clauß, S., Pfitzmann, A., and Waidner, M. (2004). Privacy-enhancing identity management. *Information security technical report*, 9(1):35–44.
- Hariharasudan, V. and Quraishi, S. J. (2022). A review on blockchain based identity management system. In *2022 3rd International Conference on Intelligent Engineering and Management (ICIEM)*, pages 735–740. IEEE.
- He, Z., Xiaofeng, L., Likui, Z., and Zhong-Cheng, W. (2015). Data integrity protection method for microorganism sampling robots based on blockchain technology. *Journal of Huazhong University of Science and Technology (Natural Science Edition)*, 43(Z1):216–219.
- Hölbl, M., Kompara, M., Kamišalić, A., and Nemeč Zlatolas, L. (2018). A

- systematic review of the use of blockchain in healthcare. *Symmetry*, 10(10):470.
- Jacobovitz, O. (2016). Blockchain for identity management. *The Lynne and William Frankel Center for Computer Science Department of Computer Science. Ben-Gurion University, Beer Sheva*, 1:9.
- Jamal, A., Helmi, R. A. A., Syahirah, A. S. N., and Fatima, M.-A. (2019). Blockchain-based identity verification system. In *2019 IEEE 9th International Conference on System Engineering and Technology (ICSET)*, pages 253–257. IEEE.
- Juan, M. D., Andrés, R. P., Rafael, P. M., Gustavo, R. E., and Manuel, P. C. (2018). A model for national electronic identity document and authentication mechanism based on blockchain. *Int. J. Model. Optim*, 8(3):160–165.
- Kitchenham, B., Pretorius, R., Budgen, D., Brereton, O. P., Turner, M., Niazi, M., and Linkman, S. (2010). Systematic literature reviews in software engineering—a tertiary study. *Information and software technology*, 52(8):792–805.
- Köpcke, H., Thor, A., and Rahm, E. (2010). Evaluation of entity resolution approaches on real-world match problems. *Proceedings of the VLDB Endowment*, 3(1-2):484–493.
- Kumar, V. and Bhardwaj, A. (2018). Identity management systems: a comparative analysis. *International Journal of Strategic Decision Sciences (IJSDS)*, 9(1):63–78.
- Kuperberg, M. (2019). Blockchain-based identity management: A survey from the enterprise and ecosystem perspective. *IEEE Transactions on Engineering Management*, 67(4):1008–1027.

- Lee, J.-H. (2017). Bidaas: Blockchain based id as a service. *IEEE Access*, 6:2274–2278.
- Li, Y., Li, J., Suhara, Y., Doan, A., and Tan, W.-C. (2020). Deep entity matching with pre-trained language models. *arXiv preprint arXiv:2004.00584*.
- Lim, S. Y., Fotsing, P. T., Almasri, A., Musa, O., Kiah, M. L. M., Ang, T. F., and Ismail, R. (2018). Blockchain technology the identity management and authentication service disruptor: a survey. *International Journal on Advanced Science, Engineering and Information Technology*, 8(4-2):1735–1745.
- Liu, Y., Sun, G., and Schuckers, S. (2019). Enabling secure and privacy preserving identity management via smart contract. In *2019 IEEE conference on communications and network security (CNS)*, pages 1–8. IEEE.
- Lo, S. K., Xu, X., Chiam, Y. K., and Lu, Q. (2017). Evaluating suitability of applying blockchain. In *2017 22nd International Conference on Engineering of Complex Computer Systems (ICECCS)*, pages 158–161. IEEE.
- Lu, Y. (2018). Blockchain: A survey on functions, applications and open issues. *Journal of Industrial Integration and Management*, 3(04):1850015.
- Lu, Y. (2019). The blockchain: State-of-the-art and research challenges. *Journal of Industrial Information Integration*, 15:80–90.
- Mahesh, B. (2020). Machine learning algorithms-a review. *International Journal of Science and Research (IJSR).[Internet]*, 9:381–386.
- Manning, C. D. (2008). *Introduction to information retrieval*. Syngress Publishing,.
- Mayadunna, H. and Rupasinghe, L. (2018). A trust evaluation model for online social networks. In *2018 National Information Technology Conference (NITC)*, pages 1–6. IEEE.

- Medibank (2022). Customer notice. <https://www.medibank.com.au/health-insurance/info/cyber-security/customer-notice/>. Accessed: 2022-12-15.
- Moniruzzaman, M., Khezr, S., Yassine, A., and Benlamri, R. (2020). Blockchain for smart homes: Review of current trends and research challenges. *Computers & Electrical Engineering*, 83:106585.
- Mudliar, K., Parekh, H., and Bhavathankar, P. (2018). A comprehensive integration of national identity with blockchain technology. In *2018 International Conference on Communication information and Computing Technology (ICCICT)*, pages 1–6. IEEE.
- Nakamoto, S. (2008). Bitcoin: A peer-to-peer electronic cash system. *Decentralized Business Review*, page 21260.
- Nie, H., Han, X., He, B., Sun, L., Chen, B., Zhang, W., Wu, S., and Kong, H. (2019). Deep sequence-to-sequence entity matching for heterogeneous entity resolution. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, pages 629–638.
- Odelu, V. (2019). Imbua: identity management on blockchain for biometrics-based user authentication. In *International Congress on Blockchain and Applications*, pages 1–10. Springer.
- Omar, A. S. and Basir, O. (2018). Identity management in IoT networks using blockchain and smart contracts. In *2018 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData)*, pages 994–1000. IEEE.

- Othman, A. and Callahan, J. (2018). The horcrux protocol: a method for decentralized biometric-based self-sovereign identity. In *2018 international joint conference on neural networks (IJCNN)*, pages 1–7. IEEE.
- Othman, A. and Callahan, J. (2020). A protocol for decentralized biometric-based self-sovereign identity ecosystem. In *Securing Social Identity in Mobile Platforms*, pages 217–234. Springer.
- Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., et al. (2011). Scikit-learn: Machine learning in python. *the Journal of machine Learning research*, 12:2825–2830.
- Peppers, K., Tuunanen, T., Rothenberger, M. A., and Chatterjee, S. (2007). A design science research methodology for information systems research. *Journal of management information systems*, 24(3):45–77.
- Peter, H. and Moser, A. (2017). Blockchain-applications in banking & payment transactions: Results of a survey. *European financial systems*, 141:141.
- Polyviou, A., Velanas, P., and Soldatos, J. (2019). Blockchain technology: financial sector applications beyond cryptocurrencies. *Decentralized 2019*, page 7.
- Rana, R., Zaeem, R. N., and Barber, K. S. (2019). An assessment of blockchain identity solutions: Minimizing risk and liability of authentication. In *2019 IEEE/WIC/ACM International Conference on Web Intelligence (WI)*, pages 26–33. IEEE.
- Rathee, T. and Singh, P. (2021). Secure data sharing using merkle hash digest based blockchain identity management. *Peer-to-Peer Networking and Applications*, 14(6):3851–3864.

- Rathee, T. and Singh, P. (2022). A self-sovereign identity management system using blockchain. In *Cyber Security and Digital Forensics*, pages 371–379. Springer.
- Regulation, P. (2016). Regulation (eu) 2016/679 of the european parliament and of the council. *Regulation (eu)*, 679:2016.
- Ren, Y., Zhu, F., Qi, J., Wang, J., and Sangaiah, A. K. (2019). Identity management and access control based on blockchain under edge computing for the industrial internet of things. *Applied Sciences*, 9(10):2058.
- Reyes-Galaviz, O. F., Pedrycz, W., He, Z., and Pizzi, N. J. (2017). A supervised gradient-based learning algorithm for optimized entity resolution. *Data & Knowledge Engineering*, 112:106–129.
- Saldamli, G., Mehta, S. S., Raje, P. S., Kumar, M. S., and Deshpande, S. S. (2019). Identity management via blockchain. In *Proceedings of the International Conference on Security and Management (SAM)*, pages 63–68. The Steering Committee of The World Congress in Computer Science, Computer
- Satybaldy, A., Nowostawski, M., and Ellingsen, J. (2019). Self-sovereign identity systems. In *IFIP International Summer School on Privacy and Identity Management*, pages 447–461. Springer.
- Schanzenbach, M., Kilian, T., Schütte, J., and Banse, C. (2019). Zkclaims: Privacy-preserving attribute-based credentials using non-interactive zero-knowledge techniques. *arXiv preprint arXiv:1907.09579*.
- Schütze, H., Manning, C. D., and Raghavan, P. (2008). *Introduction to information retrieval*, volume 39. Cambridge University Press Cambridge.
- Stokkink, Q. and Pouwelse, J. (2018). Deployment of a blockchain-based self-sovereign identity. In *2018 IEEE international conference on Internet of*

Things (iThings) and IEEE green computing and communications (GreenCom) and IEEE cyber, physical and social computing (CPSCom) and IEEE smart data (SmartData), pages 1336–1342. IEEE.

Suite, T. (2016). Ganache ethereum.

Szabo, N. (1997). Formalizing and securing relationships on public networks. *First monday*.

Takemiya, M. and Vanieiev, B. (2018). Sora identity: Secure, digital identity on the blockchain. In *2018 IEEE 42nd annual computer software and applications conference (compsac)*, volume 2, pages 582–587. IEEE.

Toth, K. C. and Anderson-Priddy, A. (2019). Self-sovereign digital identity: A paradigm shift for identity. *IEEE Security & Privacy*, 17(3):17–27.

Verge, T. (2021). Personal data of 533 million facebook users leaks online.
<https://www.theverge.com/2021/4/4/22366822/facebook-personal-data-533-million-leaks-online-email-phone-numbers/>.

Warschofsky, R., Menzel, M., and Meinel, C. (2011). Automated security service orchestration for the identity management in web service based systems. In *2011 IEEE International Conference on Web Services*, pages 596–603. IEEE.

World-Bank (2018). Principles on identification for sustainable development: toward the digital age.

Xu, J., Xue, K., Tian, H., Hong, J., Wei, D. S., and Hong, P. (2020). An identity management and authentication scheme based on redactable blockchain for mobile networks. *IEEE Transactions on Vehicular Technology*, 69(6):6688–6698.

- Zheng, Z., Xie, S., Dai, H.-N., Chen, X., and Wang, H. (2018). Blockchain challenges and opportunities: A survey. *International Journal of Web and Grid Services*, 14(4):352–375.
- Zhou, T., Li, X., and Zhao, H. (2019). Everssdi: blockchain-based framework for verification, authorisation and recovery of self-sovereign identity using smart contracts. *International Journal of Computer Applications in Technology*, 60(3):281–295.
- Zhu, X. and Badr, Y. (2018). Identity management systems for the internet of things: a survey towards blockchain solutions. *Sensors*, 18(12):4215.