

# **“Verificate” An Encrypted IPFS And Blockchain Based Credentials Verifier**

*by*

**Tasfia Rahman (191014005)  
Sumaiya Islam Mouno (192014043)  
Arunangshu Mojumder Raatul (192014037)**

*Capstone project report (CSE 499) submitted in partial fulfillment of the  
requirements for the degree of*

**Bachelor of Science in Computer Science and Engineering**

Under the supervision of

**Dr. Nafees Mansoor, SMIEEE**



**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING  
UNIVERSITY OF LIBERAL ARTS BANGLADESH**

**SUMMER 2023**

© *Tasfia Rahman, Sumaiya Islam Mouno, and Arunangshu Mojumder Raatul*  
All rights reserved

# DECLARATION

**Project Title** “Verificate” An Encrypted IPFS And Blockchain Based Credentials  
Verifier  
**Authors** Tasfia Rahman, Sumaiya Islam Mouno, and Arunangshu Mojumder  
Raatul  
**Student IDs** 191014005, 192014043, and 192014037  
**Supervisor** Dr. Nafees Mansoor, SMIEEE

---

We declare that this capstone project report entitled “*Verificate*” An Encrypted IPFS And Blockchain Based Credentials Verifier is the result of our own work except as cited in the references. The capstone project report has not been accepted for any degree and is not concurrently submitted in candidature of any other degree.

---

**Tasfia Rahman**  
**191014005**

Department of Computer Science and Engineering  
University of Liberal Arts Bangladesh

---

**Sumaiya Islam Mouno**  
**192014043**

Department of Computer Science and Engineering  
University of Liberal Arts Bangladesh

---

**Arunangshu Mojumder Raatul**  
**192014037**

Department of Computer Science and Engineering  
University of Liberal Arts Bangladesh

**Date:** July 11 , 2023

## CERTIFICATE

This is to certify that the capstone project report entitled **“Verificate” An Encrypted IPFS And Blockchain Based Credentials Verifier**, submitted by **Tasfia Rahman** (Student ID: 191014005), **Sumaiya Islam Mouno** (Student ID: 192014043) and **Arunangshu Mojumder Raatul** (Student ID: 192014037) are undergraduate students of the **Department of Computer Science and Engineering** has been examined. Upon recommendation by the examination committee, we hereby accord our approval of it as the presented work and submitted report fulfill the requirements for its acceptance in partial fulfillment for the degree of *Bachelor of Science in Computer Science and Engineering*.

---

**Nafees Mansoor, PhD, SMIEEE, Associate Professor**  
Dept. of CSE, University of Liberal Arts Bangladesh

---

**Muhammad Golam Kibria, PhD, SMIEEE, Professor Head (Acting)**  
Dept. of CSE, University of Liberal Arts Bangladesh

---

**Md Ferdous Bin Hafiz, Lecturer**  
Dept. of CSE, University of Liberal Arts Bangladesh

---

**Suravi Akhter, Lecturer**  
Dept. of CSE, University of Liberal Arts Bangladesh

**Place:** Dhaka

**Date:** July 11 , 2023

# ACKNOWLEDGEMENTS

We would like to express our deep and sincere gratitude to our research supervisor, *Dr. Nafees Mansoor, SMIEEE*, for giving us the opportunity to conduct research and providing invaluable guidance throughout this work. His dynamism, vision, sincerity and motivation have deeply inspired us. He has taught us the methodology to carry out the work and to present the works as clearly as possible. It was a great privilege and honor to work and study under his guidance.

We are greatly indebted to our honorable teachers of the Department of Computer Science and Engineering at the University of Liberal Arts Bangladesh who taught us during the course of our study. Without any doubt, their teaching and guidance have completely transformed us to the persons that we are today.

We are extremely thankful to our parents for their unconditional love, endless prayers and caring, and immense sacrifices for educating and preparing us for our future. We would like to say thanks to our friends and relatives for their kind support and care.

Finally, we would like to thank all the people who have supported us to complete the project work directly or indirectly.

*Tasfia Rahman, Sumaiya Islam Mouno and Arunangshu Mojumder Raatul*

**University of Liberal Arts Bangladesh**

**Date:** July 11 , 2023

Dedicated to  
*My Capstone Peers "Mouno"*  
*and "Raatul"*  
– *Tasfia Rahman*

To My Beloved Partners  
*"Raatul" and "Tasfia"*  
*the source of my inspiration*  
– *Sumaiya Islam Mouno*

To My Enthusiastic Friends  
*"Mouno" and "Tasfia"*  
*a good soul.*  
– *Arunangshu Mojumder Raatul*

# ABSTRACT

In Southeast Asia, submitting fraudulent certifications is a prevalent issue that keeps qualified candidates from gaining the positions they deserve. Students must provide their academic credentials—obtained both inside and outside the classroom—as proof of their qualifications when looking for jobs. Verifying academic credentials prior to employment is essential to preventing fraud. The use of blockchain technology might be able to solve this problem. Blockchain offers a tamper-proof, irrefutable electronic certificate, making it challenging for students to falsify their academic records. This study proposes a prototype for an academic credential verification mechanism that makes use of the IPFS (Interplanetary File System) and blockchain security characteristics. Before being sent to IPFS, certificates are momentarily kept in a database where a special hash code is created using a hashing technique. The blockchain nodes record this hash code, which acts as the certificate's distinctive identifier. Companies can check a candidate's credentials by looking for the candidate and retrieving any certificates that have already been confirmed. The costs associated with directly storing large amounts of data on the blockchain are reduced by using cloud technology as a middleman storage platform. In conclusion, the suggested method will improve the cost-effectiveness, efficiency, and security of the certificate verification process. The effort and materials required to manually validate certificates would be spared.

**Keywords:** Blockchain, Cloud, Certificate, Secure, Verification

# Contents

|          |  |          |
|----------|--|----------|
| <b>1</b> | <b>Introduction</b>                                | <b>1</b> |
| 1.1      | Problem statement . . . . .                        | 2        |
| 1.2      | Aims, objectives and Motivation . . . . .          | 2        |
| 1.2.1    | Aims . . . . .                                     | 2        |
| 1.2.2    | Objective . . . . .                                | 3        |
| 1.2.3    | Motivation . . . . .                               | 3        |
| 1.3      | Project Specification . . . . .                    | 4        |
| 1.3.1    | Functional Requirements: User Side . . . . .       | 5        |
| 1.3.2    | Functional Requirements: On the platform . . . . . | 5        |
| 1.3.3    | Non Functional Requirements . . . . .              | 5        |
| 1.4      | Outcome . . . . .                                  | 6        |
| 1.5      | Report Layout . . . . .                            | 7        |
| 1.6      | Summary . . . . .                                  | 8        |
| <b>2</b> | <b>Literature Review</b>                           | <b>9</b> |
| 2.1      | Preliminaries . . . . .                            | 9        |
| 2.1.1    | Blockchain . . . . .                               | 9        |
| 2.1.2    | Cloud Technology . . . . .                         | 10       |
| 2.1.3    | Ethereum And Smart Contracts . . . . .             | 10       |
| 2.1.4    | Gas Price . . . . .                                | 11       |
| 2.1.5    | Certificates . . . . .                             | 12       |
| 2.1.6    | Database . . . . .                                 | 12       |
| 2.1.7    | Encryption . . . . .                               | 13       |
| 2.1.8    | Public Blockchain . . . . .                        | 14       |
| 2.2      | Related Works . . . . .                            | 14       |
| 2.3      | Comparative Analyses . . . . .                     | 21       |
| 2.4      | Performance Analyses . . . . .                     | 24       |



|          |  |           |
|----------|--|-----------|
| <b>3</b> | <b>Methodology</b>                             | <b>33</b> |
| 3.1      | Introduction . . . . .                         | 33        |
| 3.2      | System Development Model . . . . .             | 34        |
| 3.2.1    | Incremental . . . . .                          | 34        |
| 3.2.2    | Iterative . . . . .                            | 35        |
| 3.3      | System Analysis . . . . .                      | 35        |
| 3.3.1    | System Design . . . . .                        | 36        |
| 3.3.2    | Coding . . . . .                               | 36        |
| 3.3.3    | Testing . . . . .                              | 36        |
| 3.3.4    | Implementation . . . . .                       | 36        |
| 3.4      | Requirement Engineering . . . . .              | 37        |
| 3.5      | Requirement Engineering Process . . . . .      | 37        |
| 3.5.1    | Feasibility Study . . . . .                    | 37        |
| 3.5.2    | Requirement Gathering . . . . .                | 37        |
| 3.5.3    | Software Requirement Specification . . . . .   | 38        |
| 3.5.4    | Software Requirement Validation . . . . .      | 38        |
| 3.6      | Software Requirement for Development . . . . . | 38        |
| 3.6.1    | Visual Studio 2023 . . . . .                   | 38        |
| 3.6.2    | Microsoft Visual Studio Code . . . . .         | 39        |
| 3.6.3    | Go . . . . .                                   | 39        |
| 3.6.4    | IPFS . . . . .                                 | 39        |
| 3.6.5    | NodeJS . . . . .                               | 39        |
| 3.6.6    | Metamask . . . . .                             | 40        |
| 3.6.7    | Truffle . . . . .                              | 40        |
| 3.6.8    | Mendeley . . . . .                             | 40        |
| 3.6.9    | LaTeX . . . . .                                | 41        |
| 3.6.10   | Draw.io . . . . .                              | 41        |
| 3.6.11   | Xampp . . . . .                                | 41        |
| 3.6.12   | Lucid chart . . . . .                          | 41        |
| 3.7      | Required Programming Languages . . . . .       | 42        |
| 3.7.1    | HTML . . . . .                                 | 42        |
| 3.7.2    | CSS . . . . .                                  | 42        |
| 3.7.3    | EJS . . . . .                                  | 43        |
| 3.7.4    | Java Script . . . . .                          | 43        |
| 3.7.5    | NodeJs . . . . .                               | 43        |
| 3.7.6    | Solidity . . . . .                             | 43        |
| 3.8      | Use case Diagram . . . . .                     | 44        |
| 3.8.1    | Use Case Diagram for Applicants . . . . .      | 44        |
| 3.8.2    | Use Case Diagram for Admin . . . . .           | 45        |

|          |  |           |
|----------|--|-----------|
| 3.8.3    | Use Case Diagram for Institutions . . . . .            | 46        |
| 3.8.4    | Use Case Diagram for Company . . . . .                 | 47        |
| 3.9      | Activity Diagram . . . . .                             | 48        |
| 3.9.1    | Activity Diagram for Applicants . . . . .              | 48        |
| 3.9.2    | Activity Diagram for Admin . . . . .                   | 48        |
| 3.9.3    | Activity Diagram for Institutions . . . . .            | 49        |
| 3.9.4    | Activity Diagram for Company . . . . .                 | 49        |
| 3.10     | Sequence Diagram for Smart-Contract . . . . .          | 50        |
| 3.11     | ER Diagram . . . . .                                   | 51        |
| 3.12     | Overview . . . . .                                     | 51        |
| 3.13     | System Stakeholders . . . . .                          | 52        |
| <b>4</b> | <b>Design and Development</b>                          | <b>53</b> |
| 4.1      | Project Management and Financial Activities . . . . .  | 53        |
| 4.1.1    | Work Breakdown Structure . . . . .                     | 53        |
| 4.1.2    | Estimate Costs . . . . .                               | 55        |
| 4.1.3    | Budget Plan . . . . .                                  | 57        |
| 4.1.4    | Funding Plan: . . . . .                                | 57        |
| 4.1.4.1  | Crowdfunding: . . . . .                                | 57        |
| 4.1.4.2  | Government Grants: . . . . .                           | 58        |
| 4.1.5    | Financial Report . . . . .                             | 59        |
| 4.2      | Introduction . . . . .                                 | 59        |
| 4.2.1    | Front-end . . . . .                                    | 60        |
| 4.2.2    | Back-end . . . . .                                     | 69        |
| 4.2.3    | Smart-Contract . . . . .                               | 71        |
| 4.2.4    | Algorithm . . . . .                                    | 72        |
| 4.2.5    | Encryption Algorithm Comparative Analysis . . . . .    | 74        |
| 4.2.5.1  | Graph 1: Our System . . . . .                          | 74        |
| 4.2.5.2  | Graph 2: Research Paper (Hill Cipher Scheme) . . . . . | 74        |
| 4.2.5.3  | Comparative Analysis . . . . .                         | 75        |
| 4.3      | Complexity of Our Algorithm . . . . .                  | 75        |
| 4.4      | Comparison and Conclusion . . . . .                    | 77        |
| 4.5      | summary . . . . .                                      | 78        |
| <b>5</b> | <b>Business Model</b>                                  | <b>79</b> |
| 5.1      | Infrastructure: . . . . .                              | 79        |
| 5.2      | Product/Service Offering: . . . . .                    | 79        |
| 5.3      | Revenue Model: . . . . .                               | 79        |
| 5.3.1    | Subscription Model: . . . . .                          | 79        |

|          |   |           |
|----------|---|-----------|
| 5.3.2    | Transaction Fees: . . . . .                           | 80        |
| 5.4      | Marketing Strategy: . . . . .                         | 80        |
| 5.4.1    | Identify Target Customers: . . . . .                  | 80        |
| 5.4.2    | Develop a Brand and Messaging: . . . . .              | 80        |
| 5.4.3    | Website and Content Marketing: . . . . .              | 80        |
| 5.4.4    | Search Engine Optimization: . . . . .                 | 81        |
| 5.4.5    | Paid Advertising: . . . . .                           | 81        |
| 5.5      | Technical Support: . . . . .                          | 81        |
| 5.6      | Data Security: . . . . .                              | 81        |
| 5.7      | Operational Plan: . . . . .                           | 81        |
| 5.7.1    | Team Building: . . . . .                              | 81        |
| 5.7.2    | Infrastructure Setup: . . . . .                       | 82        |
| 5.7.3    | Platform Development: . . . . .                       | 82        |
| 5.7.4    | Testing and Quality Assurance: . . . . .              | 82        |
| 5.7.5    | Launch and Customer Acquisition: . . . . .            | 82        |
| 5.7.6    | Customer Support and Maintenance: . . . . .           | 82        |
| 5.7.7    | Expansion and Growth: . . . . .                       | 83        |
| 5.8      | Cost Analysis . . . . .                               | 83        |
| 5.8.1    | Project Scope and Objectives: . . . . .               | 83        |
| 5.8.2    | Resource Needs: . . . . .                             | 83        |
| 5.8.3    | Labor Costs: . . . . .                                | 83        |
| 5.8.4    | Blockchain Platform Costs: . . . . .                  | 84        |
| 5.8.5    | Infrastructure Costs: . . . . .                       | 84        |
| 5.8.6    | Security Costs: . . . . .                             | 84        |
| 5.8.7    | Smart Contract Costs: . . . . .                       | 84        |
| 5.8.8    | Indirect Costs: . . . . .                             | 84        |
| 5.8.9    | Contingency Costs: . . . . .                          | 84        |
| 5.8.10   | ROI Analysis: . . . . .                               | 84        |
| 5.8.11   | Risk Analysis: . . . . .                              | 84        |
| 5.9      | Protocol . . . . .                                    | 85        |
| 5.10     | Selling Prices for Certificate Verification . . . . . | 87        |
| <b>6</b> | <b>Result analysis</b>                                | <b>89</b> |
| 6.1      | Performance Analysis . . . . .                        | 89        |
| 6.2      | Comparative Analysis . . . . .                        | 93        |
| 6.2.1    | Graph 1: Our System . . . . .                         | 93        |
| 6.3      | Graph 2: Research Paper . . . . .                     | 94        |
| 6.3.1    | Comparative Points . . . . .                          | 94        |
| 6.3.2    | Comparison and Conclusion . . . . .                   | 95        |

|          |  |            |
|----------|--|------------|
| 6.4      | Discussion . . . . .                                       | 95         |
| <b>7</b> | <b>Conclusions</b>   | <b>96</b>  |
| 7.1      | Social, Legal, Ethical, and Environmental Issues . . . . . | 96         |
| 7.1.1    | Social . . . . .   | 96         |
| 7.1.2    | Ethical . . . . .  | 97         |
| 7.1.3    | Environmental . . . . .                                    | 98         |
| 7.2      | Brief Summary . . . . .                                    | 98         |
| 7.3      | Future works . . . . .                                     | 99         |
|          | <b>References</b>  | <b>101</b> |

# List of Figures

|      |  |    |
|------|--|----|
| 1.1  | Block Diagram . . . . .                      | 3  |
| 1.2  | Fake Certificate Scandal . . . . .           | 4  |
| 3.1  | Hybrid Software Development Model . . . . .  | 34 |
| 3.2  | Applicant Use Case . . . . .                 | 44 |
| 3.3  | Admin use case . . . . .                     | 45 |
| 3.4  | Institution use case . . . . .               | 46 |
| 3.5  | Company use case . . . . .                   | 47 |
| 3.6  | Applicant Activity . . . . .                 | 48 |
| 3.7  | Admin Activity . . . . .                     | 48 |
| 3.8  | Institution Activity . . . . .               | 49 |
| 3.9  | Company Activity . . . . .                   | 49 |
| 3.10 | Sequence Diagram . . . . .                   | 50 |
| 3.11 | ER Diagram . . . . .                         | 51 |
| 4.1  | Work Breakdown Structure Flowchart . . . . . | 54 |
| 4.2  | Homepage . . . . .                           | 60 |
| 4.3  | About Us . . . . .                           | 61 |
| 4.4  | Login Page . . . . .                         | 62 |
| 4.5  | Applicant Interface . . . . .                | 62 |
| 4.6  | Applicant Certificate Upload . . . . .       | 63 |
| 4.7  | Applicant View Certificate . . . . .         | 63 |
| 4.8  | Applicant Profile . . . . .                  | 64 |
| 4.9  | University Interface . . . . .               | 64 |
| 4.10 | Admin Interface . . . . .                    | 65 |
| 4.11 | Requested Certificates . . . . .             | 65 |
| 4.12 | Requested Certificates . . . . .             | 66 |
| 4.13 | Send Verification Request . . . . .          | 66 |
| 4.14 | Push To Blockchain . . . . .                 | 67 |
| 4.15 | Company Page . . . . .                       | 67 |
| 4.16 | Company Interface . . . . .                  | 68 |
| 4.17 | Company Profile . . . . .                    | 68 |

|      |  |    |
|------|--|----|
| 4.18 | IPFS . . . . .                                 | 69 |
| 4.19 | Sepolia Testnet . . . . .                      | 69 |
| 4.20 | Database . . . . .                             | 70 |
| 4.21 | Transaction Details . . . . .                  | 70 |
| 4.22 | Graph from Research Paper . . . . .            | 75 |
| 6.1  | Search Time . . . . .                          | 89 |
| 6.2  | Time Required for Each Block Number . . . . .  | 90 |
| 6.3  | Retrieval Time for Each Block Number . . . . . | 91 |
| 6.4  | Graph from Our System . . . . .                | 93 |
| 6.5  | Graph from Research Paper . . . . .            | 94 |

# List of Tables

|     |   |    |
|-----|---|----|
| 2.1 | Comparative Analysis . . . . .  | 32 |
| 4.1 | Cost Breakdown . . . . .  | 57 |
| 4.2 | Cash Flow Statement . . . . .   | 59 |
| 5.1 | Selling Prices for Certificate Verification . . . . .                     | 87 |
| 5.2 | Subscription Prices for Certificate Verification System (in Tk) . . . . . | 88 |
| 6.1 | Test Cases: Certificate Submission and Verification . . . . .             | 92 |

# Chapter 1

## Introduction

The basic structure of the mainstream education system includes primary, secondary, and tertiary [Sanka et al. \(2021\)](#). Hence, after the completion of primary and secondary education, students get enrolled in universities and can pursue further studies based on their preferences. Furthermore, students participate in a variety of extracurricular activities throughout their academic years. This means that they receive a plethora of certificates throughout each stage of their education journey. The problem regarding this situation is that these certificates are at an increased risk of being lost or damaged, and there is no regulated system in place to store all these certificates digitally or verify their authenticity.

Many countries have huge populations, and with millions of graduates applying for jobs each year, individually verifying the credentials can be extremely time-consuming and taxing. It is exceedingly challenging to manage and authenticate such a vast amount of records, leading to an unfavorable situation where falsified or replicated certificates can be created through tampering. This aspect has given rise to an increasing number of fraudulent organizations that have been engaging in the unethical practice of forging academic certificates. Unfortunately, as technology advances, it has become more difficult to distinguish between genuine and forged certificates [Vujičić et al. \(2018\)](#).

To address this issue, this proposed system utilizes Blockchain, a new emerging sophisticated technology. So, what are the benefits of using Blockchain? The immutability and tamper-proof nature of blockchain make it a very robust system to use [Yaga et al. \(2019\)](#). Even if the state of the data is compromised, it can detect the change in less than a second. In Blockchain, data or nodes are validated only when multiple parties approve them [Zheng et al. \(2018\)](#). As a result, the system would always be reliable and authenticated. It is not only secure but extremely transparent about the transactions occurring in the system and there is also a traceability aspect to it.

Blockchain technology has rapidly gained popularity in recent years as a novel and promising approach to securely store, share, and manage data. Originally developed as a distributed ledger technology to support cryptocurrencies such as Bitcoin,



blockchain has now evolved to become a versatile and robust platform for a wide range of applications beyond finance, including supply chain management, healthcare, real estate, voting systems, and more [Beck \(2018\)](#). At its core, blockchain technology offers a decentralized and immutable database that is resistant to tampering and fraud. It achieves this by employing a consensus mechanism that ensures the integrity of the ledger and eliminates the need for intermediaries or central authorities to manage the data. This makes blockchain technology an ideal solution for use cases that require high levels of trust, transparency, and security. By using IPFS, files can be stored and accessed in a secure, decentralized, and censorship-resistant way [Tasatanattakool & Techapanupreeda \(2018\)](#). The integration of IPFS with blockchain technology provides an additional layer of security and immutability to the file storage system [Pilkington \(2016\)](#). Since everything is stored digitally and all the certificates are verified before being stored in the IPFS, students do not have to worry about losing or damaging their certificates. Furthermore, it streamlines the process for companies to view these verified certificates and hire eligible applicants accordingly. As a result, this proposed system closes the gaps in the current system and provides us with an effective and tangible solution.

## **1.1 Problem statement**

Our country encompasses a huge population, and with millions of graduates applying for jobs each year, the process of individually verifying the credentials can be extremely time-consuming and taxing. Keeping track of and validating such a large number of records is extremely difficult. As a result, an unfavorable scenario, namely tampering and the production of fake or duplicate certificates, has emerged. This aspect has given rise to an increasing number of fraudulent organizations that have been engaging in the unethical practice of forging academic certificates. Unfortunately, as technology advances, it has become more difficult to distinguish between genuine and forged certificates.

## **1.2 Aims, objectives and Motivation**

### **1.2.1 Aims**

The proposed solution aims to design and implement a decentralized certificate verification system using Blockchain and IPFS technology. The system will enable secure and tamper-proof storage of digital certificates, allowing for easy and efficient verification of the authenticity and integrity of the certificates by authorized parties. The use of Blockchain technology will ensure the immutability and transparency of the data stored

on the network, while IPFS will enable decentralized and distributed storage of the certificates. The system will also have a user-friendly interface for easy access and usage.

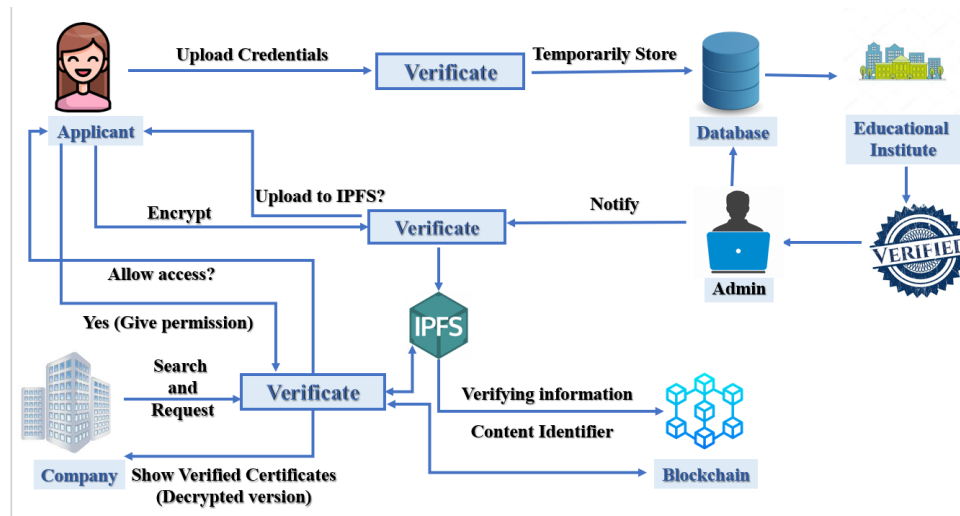


Figure 1.1: Block Diagram

## 1.2.2 Objective

- Develop a secure credential verification system using the blockchain and IPFS technology.
- Resulting in curbing the rate of certificate forgery during the job application process.
- Helps in reducing the reputational risks that the company might face if their employees are found to have submitted forged certificates to land a job.

## 1.2.3 Motivation

In 2016, The Daily Star reported that for 16 years, a man worked as a physical education instructor in a high school, purportedly with a forged Bachelor of Physical Education (BP-Ed) diploma [Star \(2023\)](#). On 28 February, 2022, the TBS news reported that Hundreds of Bangladeshi sailors have taken jobs on seagoing vessels with fake Certificates of Proficiency (COP) and Certificates of Competency (COC) [T.B.S \(2022\)](#). Furthermore, in October of 2022, after investigating nearly 25,000 educational institutions for over a decade, the government compiled a list of some 1,100 teachers with fake MPO certificates [Tribune \(2022\)](#).

On April 07, 2022, the UNB news published that a group of detectives have arrested six members of a fraud gang, including a fake vice-chancellor of a university [UNB](#)

(2023). Certificate forgery can have serious consequences for trust in the job market. Thus, this system has been proposed to eradicate the aforementioned issues.



Figure 1.2: Fake Certificate Scandal

### 1.3 Project Specification

The Prototype of a blockchain-based certificate verification system is described in this section. An candidate first uploads the system with the required credentials. The admin receives a request from the applicant to validate the certificates once the credentials have been uploaded. To ensure the validity of these certifications, the admin contacts educational institutions. The administrator notifies the users and uploads the certificates to IPFS once the credentials have been confirmed.

The certificates are divided up into smaller pieces and subjected to a hashing algorithm when students upload them to an existing protocol node in IPFS. The result of the hashing operation is a hash key, also referred to as the content identifier (CID). The CID acts as a fingerprint to identify the files specifically. For every upload, whether of fresh data or previously uploaded data, a new cryptographic hash (CID) is created. This makes every upload to the network distinct and impervious to hacking or security breaches.

The hash key is encrypted before being stored in the Blockchain nodes to increase security. The issuer must approve the generation fees in Metamask after the data has been transferred to the Blockchain via IPFS. Normally, once the hash is saved on the Blockchain, it cannot be changed. The technology notifies users right away if any data is altered. In order to apply for any job, the applicant can now email organizations the hash number. The systems can be used by the businesses to enter the hash and search for the applicants or check the validity of their certifications.

### 1.3.1 Functional Requirements: User Side

- The system must allow all the users to log into their accounts with email and password
- The system must allow its users (Applicant and Employer) to register with their university ID/ company ID, email, and password and also reset their passwords.
- The system must allow its users( Admin ) to control the access and act as the primary verifier.
- The system must allow users to maintain a user profile (creation and edition).

### 1.3.2 Functional Requirements: On the platform

- The system must allow users (Employers) to explore the available candidates.
- The system must allow its corresponding users (Applicants) to get and uploads documents.
- The system must allow generating a hash of the documents through IPFS and again encrypt the hash for security and store the encrypted hash into the Blockchain from the admin account.
- The system must allow its corresponding users ( Applicant ) to receive the request from the Employer for access control if the applicant accepts then the process will be a prepared and complete response to hassle-free hiring.

### 1.3.3 Non Functional Requirements

- **Security:** To prevent unauthorized access, tampering, or manipulation of academic credentials, the system provides a high level of security. Encryption, authentication, and access control are just a few of the strong security features offered by the IPFS and blockchain components.
- **Reliability:** The system is dependable and always open for requests for verification. It minimizes downtime and guarantees the reliability of certificates, hash codes, and stored data. To reduce possible failures, redundancy measures have been put in place.
- **Scalability:** The system is built to handle a lot of requests for verification of academic qualifications. It can handle a rising number of users and certificates without noticeably suffering from performance reduction.

- **Performance:** Verification of certificates is swift and effective thanks to the system. Low response times for verification requests guarantee a smooth process for businesses obtaining and verifying applicants' credentials.
- **Usability:** The user interface is simple and easy to use, making it simple for businesses to search for and access validated certificates. The system delivers brief, unambiguous feedback on the status of certificate verification.
- **Data Privacy:** The privacy and confidentiality of personal data and academic qualifications are given top priority by the system. It complies with applicable data protection laws and puts in place the necessary safeguards to protect sensitive data.
- **Compatibility:** The system is compatible with the infrastructure and technology currently in use by businesses and educational institutions. With well defined APIs or data exchange protocols, it integrates with other systems and supports common file formats.
- **Maintenance and Support:** The system can be updated, bugs fixed, and improvements made as needed since it is maintainable. To resolve any problems or worries made by users, adequate technical help is offered.
- **Cost-Effectiveness:** When compared to conventional manual verification methods, the solution is less expensive. It is less expensive to store huge volumes of data directly on the blockchain when using IPFS as a storage platform.
- **Compliance:** The system conforms with pertinent legal and regulatory standards pertaining to the validation of academic credentials, data security, and adoption of blockchain technology.

## 1.4 Outcome

The deployment of the model for verifying academic credentials using IPFS and blockchain has shown encouraging results in addressing the persistent problem of false certificates in Southeast Asia. Results that have been seen include:

- **Enhanced Verification Efficiency:** The effectiveness of certifying academic credentials has greatly increased with the implementation of the blockchain-based verification system. Businesses no longer need to spend time on laborious manual verification procedures since they can easily search for candidates and retrieve their validated credentials stored on the blockchain.

- **Increased Security and Fraud Prevention:** Blockchain technology's tamper-proof and non-repudiable features have successfully discouraged people from submitting false certifications. The legitimacy and integrity of academic credentials are guaranteed, lowering the danger of fraud, thanks to the use of distinctive hash codes and the decentralized storage offered by IPFS.
- **Improved Candidate Selection:** Employers may safely pick individuals based on confirmed qualifications by creating a strong verification mechanism. By ensuring that meritorious individuals with legitimate credentials are given the opportunity they deserve, this improves the workforce's overall quality.
- **Cost-Effective Solution:** The expenses involved with storing massive amounts of data directly on the blockchain have decreased thanks to the introduction of IPFS as an intermediary storage platform. With this practical method, companies of all sizes may apply the verification model more easily and sustainably.
- **Time and Resource Savings:** The automated verification procedure significantly reduces the amount of time and resources needed for manual verification. Businesses may concentrate their efforts on other important elements of the hiring process, which will simplify operations and boost efficiency.
- **Trust and Transparency:** The use of blockchain technology promotes transparency and confidence in the labor market. Verified credentials for applicants are safely saved on the blockchain, giving a trustworthy and open record of their credentials. This promotes an atmosphere that is more reputable and trustworthy for both businesses and job seekers.

Implementing the academic credential verification approach using IPFS and blockchain has, overall, had very excellent results. It has effectively addressed the widespread problem of false certifications in Southeast Asia by providing a quick, safe, and affordable solution that guarantees deserving candidates are chosen based on their true qualifications.

## 1.5 Report Layout

**Chapter 1** outlines the objectives, anticipated results, and project management for the auto parts supply chain project. It lays the framework for the remainder of the report. **Chapter 2** focuses on the reasons behind choosing this project and the motivation for its development. It also highlights the limitations and issues faced in previous comparative studies and similar projects, emphasizing the need for further research.

**Chapter 3** explores the specifics of requirements analysis, use case modeling, and implementation requirements. It specifies the actions and factors that must be taken into account for the project to be implemented properly.

**Chapter 4** analyzes the project's interface design, user experience, and front-end and back-end design specs. This chapter offers details on the system's intended aesthetic and functional layout.

**Chapter 5** covers the project's testing phase. It reviews the test findings and discusses their importance. Moreover, test results and reports are provided, emphasizing the efficiency and dependability of the adopted solution.

**Chapter 6** investigates the project's larger effects on the community and the environment. It discusses moral questions, environmental concerns, and the project's long-term viability. At the chapter's conclusion, the project's overall activities are summarized, along with future plans and prospective advancements.

Briefly stated, this report is organized to give an overview of the auto parts supply chain project, talk about the inspiration behind its creation, go into detail about the requirements and design considerations, show the outcomes of the testing and evaluation, and assess the project's effects and potential outcomes.

## **1.6 Summary**

In this chapter, We have discussed the situation of the current system, necessity and brief overview of the proposed system.

# Chapter 2

## Literature Review

In this chapter, we have covered the preliminaries along with the related works and comparative analyses.

### 2.1 Preliminaries

In this section, we have discussed the following topics related with the context of making a blockchain based credentials verification system.

#### 2.1.1 Blockchain

Blockchain technology is a distributed ledger that provides a decentralized and transparent platform for secure data management [Kumutha & Jayalakshmi \(2022\)](#). With its immutable and tamper-proof nature, blockchain offers an ideal solution for certificate verification, ensuring the integrity and authenticity of educational certificates [Castro & Au-Yong-Oliveira \(2021\)](#). A blockchain-based certificate verification system enables graduates to store their academic credentials securely on the blockchain, eliminating the need for physical storage and management of paper certificates [Castro & Au-Yong-Oliveira \(2021\)](#). The system ensures that only authorized individuals can access and verify the certificates, reducing the possibility of fraudulent activities [Gräther et al. \(2018\)](#).

Blockchain technology offers a reliable and efficient method for managing educational data, providing graduates with complete control over their information [Dongre et al. \(2020\)](#). The decentralized structure of the blockchain ensures that no central authority can manipulate the data, ensuring data security and integrity [Angelis & Da Silva \(2019\)](#). Furthermore, the system is highly resilient to any malicious attacks, providing a secure and trustworthy way of verifying academic credentials.



### 2.1.2 Cloud Technology

Cloud technology has been a game-changer in the world of data management, providing a scalable and flexible platform for organizations to store and manage their data. However, concerns over data security and privacy remain a significant challenge for cloud-based solutions, particularly when it comes to sensitive data such as educational certificates [Castro & Au-Yong-Oliveira \(2021\)](#). One potential solution to this challenge is to leverage blockchain technology in a cloud-based certificate verification system.

A blockchain-based certificate verification system hosted on the cloud offers several advantages. Firstly, the cloud infrastructure provides a highly scalable and flexible platform for managing the blockchain network, allowing for easy access and management of educational certificates. Additionally, the cloud infrastructure provides advanced security features such as access control, encryption, and authentication mechanisms, further enhancing the security of the system [Hasan et al. \(2020\)](#).

Furthermore, a cloud-based blockchain solution for certificate verification offers an efficient and cost-effective way of managing and verifying educational certificates. By eliminating the need for physical storage and management of paper certificates, the system reduces administrative costs and streamlines the verification process [Gupta & Sadoghi \(2021\)](#). Moreover, the use of blockchain technology ensures the immutability and tamper-proof nature of the data, reducing the possibility of fraudulent activities.

In conclusion, leveraging blockchain technology in a cloud-based certificate verification system offers a promising solution for ensuring the security and authenticity of educational certificates. The combination of cloud and blockchain technology provides a highly scalable and secure platform for managing and verifying certificates, enabling graduates to share their educational data with ease and confidence. With the increasing importance of data security and privacy, the adoption of cloud-based blockchain solutions in certificate verification is a step towards a more secure and efficient data management ecosystem [Kiffer et al. \(2018\)](#).

### 2.1.3 Ethereum And Smart Contracts

Ethereum is a blockchain-based platform that enables the creation and deployment of smart contracts [Rasool et al. \(2020\)](#). Smart contracts are self-executing contracts that automatically enforce the terms of the agreement, providing a transparent and efficient way of executing transactions [Khan et al. \(2021\)](#). In the context of certificate verification, Ethereum and smart contracts offer a promising solution for ensuring the authenticity and security of educational certificates.

By leveraging Ethereum and smart contracts, educational institutions can create a decentralized system for managing and verifying certificates. The use of smart contracts eliminates the need for intermediaries or central authorities, reducing the risk

of fraudulent activities [S. Wang et al. \(2019\)](#). Additionally, the immutability and tamper-proof nature of the Ethereum blockchain ensure that once a certificate is recorded on the blockchain, it cannot be altered or deleted, providing a high level of security and integrity to the system [Watanabe et al. \(2016\)](#).

Smart contracts also offer an efficient way of executing transactions, eliminating the need for manual verification and reducing administrative costs. With smart contracts, the verification process can be automated, providing instant verification of certificates. Furthermore, smart contracts can be programmed to enforce certain conditions, such as expiration dates, ensuring that the certificates remain valid and up-to-date [Shakan et al. \(2021\)](#).

Ultimately, the management and verification of educational credentials is made much easier, safer, and more transparent through the use of Ethereum and smart contracts. Graduates may fully manage their educational data, ensuring that it is safe and available, by doing away with the need for intermediaries and offering a tamper-proof system. A step toward a more secure and effective data management ecosystem, which will help both graduates and educational institutions, is being made with the implementation of Ethereum and smart contracts in certificate verification.

#### **2.1.4 Gas Price**

In the context of blockchain-based certificate verification, gas price plays a crucial role in determining the cost and efficiency of executing transactions. Gas price refers to the fee paid in ether, the native cryptocurrency of the Ethereum blockchain, for executing transactions on the blockchain [Gupta & Sadoghi \(2021\)](#).

In the case of certificate verification, gas price can impact the speed and cost of verifying certificates on the blockchain. A higher gas price results in faster transaction execution, but at a higher cost, while a lower gas price results in slower transaction execution but at a lower cost [Gourisetti et al. \(2019\)](#). Therefore, it is important to strike a balance between gas price and transaction speed and cost when designing a certificate verification system on the blockchain.

One potential solution to balance gas price and transaction speed and cost is to use off-chain solutions such as layer-two solutions or sidechains. These off-chain solutions can reduce the gas cost of executing transactions while still maintaining the security and integrity of the blockchain. Additionally, using a gas price oracle, which provides real-time gas prices, can help optimize transaction costs by setting the gas price to match the current market conditions [Kosba et al. \(2016\)](#).

The cost of gas is a major consideration for developing a blockchain-based certificate verification system that is both effective and affordable. Educational institutions may offer a dependable and secure verification procedure for graduates' certificates while

reducing the cost and time necessary for transaction execution by striking the correct balance between gas price, transaction speed, and cost.

### 2.1.5 Certificates

Certificates are an essential component of education and professional development, providing individuals with proof of their knowledge, skills, and qualifications. However, traditional paper-based certificates are often subject to fraudulent activities such as tampering and forgery, which can undermine their authenticity and reliability [Malik et al. \(2019\)](#).

Blockchain technology offers a promising solution for securely storing and verifying certificates. By leveraging a decentralized and tamper-proof database, blockchain-based certificates can provide a high level of security and transparency, ensuring that certificates cannot be forged or duplicated [Nouman et al. \(2021\)](#).

One of the key benefits of blockchain-based certificates is that they can be easily verified by any party without the need for a central authority. The certificates are stored on the blockchain in a secure and immutable manner, and their authenticity can be verified through a cryptographic hash or digital signature [Norta \(2017\)](#).

In addition, blockchain-based certificates can be easily shared and transferred between individuals, educational institutions, and employers. This eliminates the need for physical storage and management of paper certificates, reducing administrative costs and streamlining the verification process [Hellman \(2002\)](#).

Smart contracts, a key feature of blockchain technology, can further enhance the certificate verification process by automating the verification and validation of certificates. Smart contracts can be programmed to execute certain actions automatically, such as verifying the authenticity of a certificate or revoking a certificate in case of fraudulent activity [Liu & Guo \(2019\)](#).

In general, blockchain-based certificates provide a trustworthy, safe, and effective method for certificate verification. Blockchain-based certificates can offer a more reliable and trustworthy system for both individuals and organizations since they have the ability to lower fraud and expedite the verification process.

### 2.1.6 Database

The certificate verification system relies on an SQL database to efficiently store, organize, and manage data. SQL databases offer a structured approach to data storage, using tables, rows, and columns [Desai \(2018\)](#). With relationships defined through primary and foreign keys, data integrity is ensured, allowing for seamless retrieval and manipulation of information. The SQL database enables the system to execute powerful queries to fetch specific data, apply filters, and perform complex operations [Gupta & Sadoghi](#)

(2021). Moreover, SQL databases provide built-in security features, including access control mechanisms, to protect the confidentiality and integrity of the stored data. Overall, the SQL database plays a crucial role in facilitating secure and efficient data management within the certificate verification system.

### 2.1.7 Encryption

Encryption plays a vital role in ensuring the security and confidentiality of data in a blockchain-based document verification system [Kosba et al. \(2016\)](#). Encryption is the process of transforming data into a form that can only be read by authorized parties. In a blockchain-based document verification system, encryption is used to protect the digital certificates and sensitive information from unauthorized access and tampering [Zhai et al. \(2019\)](#).

In a blockchain-based document verification system, the digital certificates are encrypted using advanced cryptographic techniques, such as symmetric and asymmetric encryption [Balogh et al. \(2021\)](#). Symmetric encryption uses a single key to encrypt and decrypt data, while asymmetric encryption uses two keys, a public key and a private key, to encrypt and decrypt data [M. Wang et al. \(2018\)](#).

When a digital certificate is added to the blockchain, it is encrypted using the public key of the certificate owner. The encrypted certificate is then added to the blockchain, where it is stored and verified using the distributed consensus mechanism [Hellman \(2002\)](#).

To access the digital certificate, the authorized parties must have the private key to decrypt the certificate. This ensures that only the certificate owner and authorized parties can access and verify the certificate. Encryption also ensures that the certificates cannot be tampered with or altered without the private key [Leka & Selimi \(2021\)](#).

In addition to encrypting the digital certificates, encryption is also used to protect other sensitive information, such as personal identification information, transaction details, and user credentials. This helps to ensure the confidentiality and integrity of the data, preventing any unauthorized access or tampering [Ge et al. \(2022\)](#).

Encryption, in general, is a crucial part of a blockchain-based document verification system because it offers a safe and dependable mechanism to shield digital certificates and other sensitive data from illegal access and manipulation. A blockchain-based document verification system may guarantee the security and secrecy of the data by employing modern encryption techniques, making it a very trustworthy and dependable system for managing and confirming digital certificates.

### 2.1.8 Public Blockchain

A public blockchain is a decentralized, transparent, and immutable database that is publicly accessible and verifiable. It allows anyone to participate in the network, view and verify the transactions, and access the data stored on the blockchain. A public blockchain can be used for various applications, including document verification and authentication [Gai et al. \(2020\)](#).

In a blockchain-based document verification system, a public blockchain can be used to store and verify digital certificates. When a digital certificate is added to the blockchain, it becomes a permanent and tamper-proof record that can be verified by anyone on the network. The distributed consensus mechanism employed by a public blockchain ensures that the data stored on the blockchain is accurate and valid [Rahardja et al. \(2021\)](#).

Using a public blockchain for a document verification system offers several advantages. Firstly, it eliminates the need for a central authority to manage and authenticate the certificates. Instead, the blockchain network itself is responsible for validating and verifying the certificates [Cheng et al. \(2018\)](#). This makes the system more transparent, secure, and efficient.

Secondly, a public blockchain is highly resistant to tampering and fraud. Since the data stored on the blockchain is immutable and transparent, it is virtually impossible to alter or manipulate the data without detection. This ensures the integrity and authenticity of the certificates, making the verification process more reliable and trustworthy [Natarajan et al. \(2017\)](#).

Thirdly, a public blockchain enables seamless and secure sharing of certificates among different parties, such as employers, educational institutions, and government agencies. Since the certificates are stored on the blockchain, the certificate owners can easily share them with anyone on the network without the need for intermediaries or additional verification.

Ultimately, managing and validating digital certificates can be done securely, consistently, and effectively via a public blockchain. Organizations may save administrative expenses, get rid of fraud, promote transparency, and boost confidence in the document verification process by adopting a public blockchain.

## 2.2 Related Works

The primary objective of this literature review is to investigate the current state of academic certificate verification through blockchain technology. The review will examine various aspects such as different approaches, strategies, systems, and techniques utilized in previous research studies. The main goal is to obtain a comprehensive

understanding of this area and identify potential opportunities for further research and improvement.

A comprehensive analysis of the utilization of blockchain technology in smart contracts is conducted. The investigation delves into the fundamental principles of blockchain, such as its decentralized nature, the different consensus algorithms used, and the importance of cryptographic security. In addition, a detailed examination of smart contracts is provided, exploring their functions and possible applications. The article also takes a critical stance, evaluating the potential advantages and disadvantages of implementing blockchain technology in smart contracts. The author examines how the benefits of blockchain, such as transparency and immutability, could revolutionize the way smart contracts operate, while also recognizing the challenges and limitations of blockchain technology [Zheng et al. \(2018\)](#).

In recent years, there has been a growing need for decentralized document sharing and version control in different industries, and the education sector is no exception. This demand stems from the fact that the traditional centralized systems for document sharing and version control can be vulnerable to cyber threats and unauthorized access, leading to data breaches and loss of sensitive information. Leverage emerging technologies such as the InterPlanetary File System (IPFS) and smart contracts, the proposed framework offers a decentralized approach to document storage and version control, allowing multiple users to access and modify documents without relying on a central authority or intermediary. It also enables event triggering and transaction monitoring, ensuring that all actions taken on the documents are transparent and tamper-proof. The primary actors in this framework are developers and approvers, who have the necessary permissions to create, modify, and approve documents [Nizamuddin et al. \(2019\)](#).

The authors introduce a certificate verification and generation system that provides users with two distinct modes of certificate generation. The first option involves filling out a form to generate a single certificate, while the second option allows for bulk certificate generation by uploading a CSV file. The ability to generate certificates in bulk is particularly useful for organizations or institutions that need to issue certificates to a large number of people. With this option, users can upload a CSV file containing the relevant data, and the system will automatically generate certificates for all the entries in the file. On the other hand, the single-certificate generation option is ideal for individual users who need to generate a certificate for a one-time event or activity. With this option, users can simply fill out the necessary details in the form, and the system will generate a certificate for them [Lamkoti et al. \(2021\)](#).

A new feature has been integrated into the certificate verification and generation system, which is the ability to generate Quick Response (QR) codes. These QR codes enable a smoother verification process as users can quickly scan the code to verify the



certificate's legitimacy, making the process more efficient. Additionally, the QR codes make sharing certificate information easier and faster, as they can be easily shared through various digital platforms such as email or messaging apps. The incorporation of this new feature has added an extra layer of security and convenience to the certificate verification process, ultimately making it more accessible and user-friendly [Abdullahi et al. \(n.d.\)](#).

The UniverCert platform is designed to be a consortium blockchain that involves various stakeholders, including higher education institutions, governments, law enforcement agencies, and employers. The platform's primary objective is to provide a reliable and secure system for verifying academic and professional certificates, which can be easily accessible to all stakeholders. It operates on the Ethereum blockchain technology, which is an open-source and decentralized platform. To access the UniverCert platform's features, the RestAPI channel is used, which enables users to access the platform's functionalities in a simple and user-friendly way [Shakan et al. \(2021\)](#).

In an effort to enhance the efficiency of transaction throughput, Kafka was integrated into the message queuing process. As a result of the implementation of this technology, the processing times for transactions increased significantly, outpacing those of other blockchain-based solutions. For the study, Hyperledger Fabric, which is a well-established blockchain framework employed in the creation of enterprise-level applications, was employed. The researchers' discoveries are critical since quicker processing times have the potential to enhance overall efficiency and decrease expenses for enterprises [Liu & Guo \(2019\)](#).

Introduction of an additional accrediting body to further ensure the legitimacy of universities authorized to issue and verify certificates is an important step. This feature adds an extra layer of validation to the certificate verification process, providing increased security and trust in the system. To ensure confidentiality and data security, the AES encryption algorithm was used. The system also offers the ability to submit and verify multiple academic certificates in bulk, streamlining the verification process and increasing its efficiency [Leka & Selimi \(2021\)](#).

An in-depth examination is conducted on the methods for enhancing the security of digital documents through the use of timestamping and digital signatures. The digital signature itself is comprised of four essential components, namely the hash code, public key, private key, and timestamp. It is worth noting that the university issues both a physical copy of the educational certificate as well as the digitally signed document to the student [Ghazali & Saleh \(2018\)](#).

An in-depth evaluation of various blockchain platforms, and their findings revealed that Hyperledger Fabric proved to be the most fitting platform for their research objectives. The research work also emphasized the importance of the platform's sturdy privacy features, which include a mechanism that allows access control based on the

role assigned to a user within the system. This access control feature enables the regulation of user's access to data and transactions. Additionally, Hyperledger Fabric adopts a permissioned network, which guarantees that only authorized nodes are granted access to the network and its data. This feature contributes significantly to ensuring data security and privacy for users on the platform [Saleh et al. \(2020\)](#).

The primary objective of the paper is to create a data management system for assessing student quality using a consortium blockchain fabric based on RBFT (Redundant Byzantine Fault Tolerance). Alongside designing the system, the authors also proposed a plan to optimize its performance to ensure high availability. The optimization of performance was achieved by utilizing a data storage model that combined multiple technologies such as Redis, MySQL, IPFS, and consortium blockchain. The data storage process adopted an "on-chain and off-chain" mechanism, whereby data was stored both on the blockchain for security purposes and off-chain to minimize the size of the blockchain. This hybrid approach was utilized to maintain the security of the data while still ensuring that the blockchain's size remains manageable. Additionally, the combination of various storage technologies contributed to making the system more efficient and reliable, thereby guaranteeing better performance [Liang & Zhao \(2020\)](#).

MIT Media Lab utilize Blockcerts to provide digital certificates to student groups, giving the recipients more authority over their earned certificates. This initiative ensures that the recipients do not have to rely on any third-party intermediaries to validate, store or verify their credentials. The certification architecture developed by MIT operates by having the issuer sign a digital certificate and store its hash in the blockchain transaction. Subsequently, the output of this transaction is then assigned to the recipient. This mechanism helps to ensure that the certification process is secure and reliable, with no risk of manipulation or loss of data [Vidal et al. \(2019\)](#).

An innovative method of sharing data utilizing blockchain technology is brought to the forefront. The researchers propose a semi-decentralized approach that integrates the InterPlanetary File System (IPFS) to enable secure and efficient data sharing. The process involves the data owner uploading an encrypted file onto the IPFS platform, which is then divided into  $n$  sections, known as hash codes. These hash codes serve as secret keys that grant authorized parties access to the data. In addition, the data owner is required to create 7 access permissions for the encrypted file, thereby enhancing its security. Through this novel approach, data sharing becomes more secure and transparent, with only authorized parties being able to access the data. This method promises to revolutionize data sharing, providing a more secure and efficient way of sharing information between parties while minimizing the risk of data breaches and unauthorized access [Athanere & Thakur \(2022\)](#).

A new method for verifying educational certificates proposes the incorporation of biometric technology. To be more precise, students are asked to submit the hash of



their biometric data along with a distinctive phrase to authenticate their identity. The biometric data is employed as a primary means of verification, while the unique phrase is used as a secondary factor to further strengthen the security of the verification process. The use of biometric technology in this novel approach ensures that the certificate holder's identity is thoroughly verified, and any potential for fraud is eliminated [Dalal et al. \(2020\)](#).

The creation of a blockchain network that stores certificates is achieved through the utilization of GETH, which is the Go implementation of Ethereum. Based on the findings of the research, the GETH implementation was shown to be reliable, processing a considerable amount of 200 transactions within an 8-second period, thus demonstrating its efficiency in handling the storage of certificates. In terms of scalability testing, it was discovered that becoming a node or miner on the blockchain network would require a storage capacity of approximately 22.6 GB to accommodate a substantial amount of up to 10 million blocks [Faaroek et al. \(2022\)](#).

The study involves the exploration of several blockchain technology applications. Firstly, a thorough analysis of three of the most popular blockchain-based cryptocurrencies, Bitcoin, Litecoin, and Ethereum, was conducted. The analysis helped to identify the unique features of each cryptocurrency, along with their strengths and limitations, providing an improved understanding of the cryptocurrency landscape. Secondly, the study focused on examining the features and concerns related to Bitcoin cryptocurrency, such as scalability issues and the potential for fraud. These considerations are crucial for the adoption of this cryptocurrency in various applications. Lastly, a graphical user interface was developed to facilitate IPFS bandwidth analysis. The interface enables the storage of files on the network using Web3 JS and Smart Contracts [Nouman et al. \(2021\)](#).

The Open University UK's Knowledge Media Institute (KMI) has started implementing badges, certificates, and web reputation using blockchain technology as a reliable record-keeping system. KMI is using Ethereum to transform badges into smart contracts and has built a sample model for awarding micro-credentials on the blockchain. KMI's efforts are centered on creating blockchain solutions for UK higher education credentials and leading the way in blockchain initiatives in higher education. KMI has formed partnerships with other institutions, including the University of Ghent and the University of Texas, to collaborate on blockchain initiatives [Domingue \(2017\)](#).

The University of Nicosia (UNIC) has integrated the use of Bitcoin blockchain into various activities, including accepting Bitcoin payments for tuition fees for all degree programs and issuing academic certificates on the Bitcoin blockchain. The implementation of educational certificates on the blockchain aims to eradicate fraudulent activities and also address issues related to fraudulent payments made by international students. The primary objective is to overcome issues of tampering with student cohort numbers.

Since 2017, UNIC has been issuing all diplomas using the blockchain and provides software tools for users to verify the authenticity of the certificate. Their user-facing systems use current open-source standards, and UNIC is a member of the Blockcerts consortium. The SHA-256 hash algorithm is used to share certificates as a PDF file with other entities due to its ability to create a hash from the certificate, but the reverse is not possible. The authenticity of the certificate is ensured by searching for the certificate's SHA-256 code within the index document, and if it matches, the certificate is deemed authentic [Bond et al. \(2015\)](#).

SmartCert is a blockchain-based platform that has been specifically developed to verify the authenticity of academic credentials and address the problem of fake certificates. The platform utilizes cryptographic signing of educational certificates, which ensures transparency in the recruitment process by enabling potential employers to validate the certificates. This feature enhances the credibility and trustworthiness of the certificates, thereby minimizing the risk of fraudulent activities. The student can share the hash, which is a unique identifier assigned to each certificate, with the prospective employer for verification purposes, providing a simple and secure way to authenticate the certificate [Kanan et al. \(2019\)](#).

Records Keeper is a blockchain-based solution that offers an innovative way to verify academic certificates. Educational institutions can use Records Keeper to issue certificates to their students, and provide them with a receipt that can be shared with third parties as proof of authenticity. The receipt obtained from the student can be used by the third party to verify the certificate's authenticity in the Records Keeper ledger, which serves as a trusted and secure repository for all academic certificates. This ensures that the certificate is genuine and has not been tampered with in any way. By using Records Keeper, educational institutions can provide a reliable and efficient way for employers and other third parties to verify the credentials of their students, making it easier for them to gain employment or pursue further education opportunities [Desai \(2018\)](#).

The paper discusses the use of hash functions in blockchain systems, with a focus on the SHA-256 hash function, which is considered to have more security than the vulnerable MD5 hash function. The SHA-256 processor is described as performing 64 iterations with 512-bit message blocks and 256-bit hash values. The paper recommends the use of the SHA-256 hash for optimal data security in blockchain systems. However, the paper notes that embedding a hash into a certificate and then using the certificate as a hash again will produce a different combination of hash codes. Therefore, the paper proposes a new method for embedding blockchain technology in a certificate, which involves checking the certificate from a hash [Rahardja et al. \(2021\)](#).

The study is based on the Ethereum platform and uses the EVM (Ethereum Virtual Machine) to run the application. Three groups of users are involved in the system:

schools or certification units, graduates, and companies or employment agencies. The process of the system includes schools granting a degree certificate and entering the student's data into the system, the certificate system verifying all the data, and schools granting e-certificates containing a QR code to the graduates whose data have been successfully verified. Graduates can look up their certificate by signing into the system. When a company acquires a serial number or QR code from a job applicant, they sign into the system to verify the veracity of the associated certificate [Cheng et al. \(2018\)](#).

DistB-CVS is an online certificate verification system, which uses private blockchain, cloud database, and a dissemination mechanism to provide a secure and efficient solution for verifying certificates. The proposed system uses an on-chain approach for data storage and includes features such as timestamping, hash function, digital multi-signature, and revocation lists. The network layer includes a peer-to-peer network and an authentication mechanism for users to log in with their user ID and password. Finally, the authors present an algorithm for block validation that uses a consensus mechanism to validate the block [Hasan et al. \(2020\)](#).

The paper proposes an architecture for a blockchain-based education record tracking and verification system. The system consists of a provider node, individual nodes, and a miner, with the blockchain at the center. The education records are managed by trusted nodes and institutions, which can add records to the system and transfer the data to the blockchain system with security and privacy considerations. The miner adds blocks to the blockchain, and smart contracts are used to automate the execution of commands based on user preferences. It also allows individuals to retain control of their data without being data managers and eliminates the need for coordinating with multiple agencies for record gathering and education verification [Han et al. \(2018\)](#).

The authors propose a consortium blockchain approach, which involves collaborating universities acting as peers in a distributed ledger network. This approach allows for public verification of some academic data while maintaining anonymity for other data, such as student records. The paper describes the use of a distributed consensus protocol to validate the chronology of generated data in the blockchain. The protocol involves electing a peer among the collaborating universities to prepare and stamp a newly created block. The paper also presents algorithms for miner election and block validation [Srivastava et al. \(2018\)](#).

The paper proposes a blockchain-based solution for the Higher Education Commission (HEC) degree attestation traceability problem. The proposed solution uses two distinct types of traceability, which are used by the organization: hiring bodies to the respective university and universities to HEC. It is deployed on the fabric hyperledger, using a distributed network of hyperledger fabric-based nodes, which create the distributed network. The proposed solution uses the InterPlanetary File System (IPFS) storage as an external storage structure for HEC degree attestation traceability, which

preserves candidate degree credentials and personal information and is shared only among the stakeholders within the permissioned private chain network architecture. The paper also discusses multiple peer channels used to update and query execution on the HEDU-ledger and auto-synchronizes and executes two roles mainly, such as endorsing to committing transactions or vice versa [Ayub Khan et al. \(2021\)](#).

In conclusion, the use of blockchain and IPFS technology for certificate verification has gained significant attention in recent years. The reviewed research papers showcase different approaches and solutions to ensure data security, integrity, and easy verification in the education sector. The immutability of the blockchain and the tamper-proof storage of IPFS provide secure, efficient, and cost-effective solutions for document management and verification. However, each proposed system has its limitations and drawbacks, such as excessive bandwidth consumption or vulnerability in storing files locally. As the demand for secure document verification in education continues to grow, further research is necessary to overcome these limitations and enhance the efficacy and feasibility of blockchain and IPFS-based solutions. Overall, the integration of blockchain and IPFS in certificate verification has the potential to revolutionize the education sector's document management system and bring about a significant transformation in the industry.

## 2.3 Comparative Analyses

Performance metrics play a crucial role in evaluating the effectiveness and efficiency of a system, and hence they are an important aspect of any research paper. By measuring the system's performance, researchers can analyze the system's strengths and weaknesses, identify any bottlenecks or performance issues, and propose improvements [Padilla et al. \(2020\)](#). They are a gateway to compare the system's performance with that of other similar systems. Some elected parameters for comparative analysis of this study include:

- **Type of Framework-** A blockchain framework is a set of protocols and rules that govern the functioning and use of a blockchain network [Quasim et al. \(2020\)](#). It outlines how transactions are processed, validated, and recorded on the blockchain, and how participants interact with the network. Examples of blockchain frameworks include Ethereum, Hyperledger, and EOS [Gourisetti et al. \(2019\)](#).
- **Type of Blockchain-** there are several types of blockchain, including:
- **Public blockchains:** Public blockchains are open-source and allow anyone to participate in the network as a node or user. Examples include Bitcoin and Ethereum [Bhutta et al. \(2021\)](#).

- **Private blockchains:** Private blockchains are restricted to a select group of participants and are often used for enterprise applications [Pahlajani et al. \(2019\)](#). These blockchains are typically permissioned, meaning that participants must be granted permission to access and use the network [Bhutta et al. \(2021\)](#).
- **Consortium blockchains:** Consortium blockchains are a hybrid of public and private blockchains and are governed by a group of organizations [Dib et al. \(2018\)](#). This type of blockchain is often used in industries where a trusted network of participants is required, but where a fully private network is not necessary [Bhutta et al. \(2021\)](#).
- **Hybrid blockchains:** Hybrid blockchains combine elements of both public and private blockchains to provide the best of both worlds. These blockchains can be used to balance the benefits of public and private blockchains and to provide customized solutions for specific use cases [Ge et al. \(2022\)](#).
- **On Chain** - refers to transactions, activities or data that are stored and processed directly on a blockchain network. This means that they are directly recorded on the blockchain ledger, forming a permanent and publicly accessible record [Fekete & Kiss \(2023\)](#).
- **Off Chain** - refers to transactions or activities that are not recorded on the blockchain and are instead processed outside of the blockchain network. example: other database systems [Fekete & Kiss \(2023\)](#).
- **Cloud** - Cloud storage refers to the practice of storing data and files on remote servers accessed over the internet, rather than on a local computer or physical storage device . The data is maintained, managed, and backed up by a cloud storage service provider [Gai et al. \(2020\)](#).
- **User Centric-** A user-centric system in blockchain technology refers to a system that is designed to prioritize the needs and experiences of the end-user. This includes making the system user-friendly, accessible, and easy to understand, as well as ensuring that the user is in control of their data and assets [Augot et al. \(2017\)](#).
- **Pseudonymity** - Pseudonymity in blockchain refers to the ability for users to transact on a blockchain network without revealing their true identity [Monrat et al. \(2019\)](#). Instead of using real-world identity information, users are identified on the blockchain through a pseudonym or a string of characters, such as a public key or a blockchain address. This allows for privacy and anonymity in transactions, as the identity of the person behind the pseudonym is not publicly accessible [Monrat et al. \(2019\)](#).

- **Access Control** - Access control in blockchain refers to the management of who is allowed to access and modify the data stored on a blockchain network. It is a key aspect of security in blockchain, as it ensures that only authorized users can perform certain actions, such as adding new transactions or modifying existing ones. Access control can be achieved through the use of digital signatures, encryption, and permissions management systems [Li et al. \(2020\)](#). In a public blockchain, access control is typically based on cryptographic proof of ownership, such as private key ownership. In a consortium or private blockchain, access control is often managed through a centralized authority or a group of authorized participants. The specific access control mechanism used can vary based on the blockchain framework and the specific use case [Steichen et al. \(2018\)](#).
- **Consensus Type** - process by which the participants in a blockchain network agree on the current state of the ledger and validate new transactions [Mingxiao et al. \(2017\)](#). The consensus mechanism is a key component of blockchain technology, as it ensures the integrity and security of the network by preventing double-spending and other forms of tampering [Monrat et al. \(2019\)](#).
- **Confidentiality** - the ability to keep sensitive information hidden from unauthorized parties while still maintaining the transparency and immutability of the blockchain. Confidentiality is a key concern in many blockchain use cases, particularly in financial and healthcare applications, where personal or sensitive information must be protected [Diaconita et al. \(2023\)](#). Example methods: encryption, zero knowledge proofs, private or consortium Blockchain.
- **Accountability** - responsibility of participants in a blockchain network for the actions they take and the decisions they make [Monrat et al. \(2019\)](#). For example, the use of digital signatures and public key infrastructure can provide a strong form of accountability, as it links a transaction to a specific user. Additionally, the transparency and immutability of the blockchain can increase accountability by allowing all participants to see the actions taken by others and to hold them accountable for any malicious or fraudulent behavior [Spanò et al. \(2022\)](#).
- **Cost effective** - The ability of a blockchain solution to minimize costs while still delivering the desired outcome. The cost-effectiveness of a blockchain solution depends on various factors, including the size of the network, the complexity of the use case, and the specific requirements and constraints of the project [Fekete & Kiss \(2023\)](#).
- **Scalability** - Ability of a blockchain network to accommodate a growing number of users and transactions without reducing its performance or efficiency [Monrat et al. \(2019\)](#).



- **Autonomy** - Ability of a blockchain network to operate independently, without the need for centralized control or intervention. In a blockchain network, autonomy is achieved through the use of decentralized consensus mechanisms, smart contracts, and other self-executing code [Bhutta et al. \(2021\)](#).
- **Fault Tolerance** - The ability of a blockchain network to continue operating even when some components of the network fail or are unavailable [Bhutta et al. \(2021\)](#).

## 2.4 Performance Analyses

The table provides a comparison of various blockchain frameworks and their features as described by different authors. The authors have focused on different aspects of the frameworks, including whether they are on-chain, off-chain, or cloud-based, their pseudonymity, access control, and consensus mechanism.

One common framework discussed by many authors is Ethereum, which is used for both consortium and public blockchains. Ethereum is described as having both on-chain and off-chain features, and it offers a high degree of pseudonymity and access control. It also uses a consensus mechanism to validate transactions.

Hyperledger is another commonly discussed framework that is used for permissioned blockchains. The authors describe Hyperledger as being mostly off-chain and offering strong access control, but it is less pseudonymous than other frameworks. It also has a consensus mechanism to validate transactions.

Bitcoin is also discussed by some authors, and it is generally described as a public blockchain with strong pseudonymity and consensus mechanisms. However, it is less focused on access control than other frameworks.

Overall, the table provides a useful overview of different blockchain frameworks and their features. Now based on the research here are the systemic analysis;

The Ethereum consortium blockchain framework supports on-chain and off-chain transactions, cloud-based deployment, pseudonymity, access control, and consensus. The system provided a user-centric solution with confidentiality, accountability, and fault tolerance. The author did not address the issue of data storage and management. The paper did not discuss the potential limitations of the proposed blockchain-based system for verifying student and graduate data. It would have been useful to have a discussion on the potential drawbacks of the system, such as the possibility of errors in data entry, the risk of data manipulation, and the need for a secure and reliable blockchain infrastructure. Additionally, the authors did not address how their proposed system would handle cases where graduates may wish to keep their personal information private. Further research and discussion on these limitations could provide valuable insights for future work in this area [Shakan et al. \(2021\)](#).

A thorough examination of the Ethereum consortium blockchain framework assessment revealed its privacy and security features. They reported that the framework provides strong data privacy and security through the use of smart contracts and encryption techniques. The authors noted that the framework is highly customizable, allowing users to specify various parameters such as consensus algorithms and block sizes to suit their needs. However, the authors did not evaluate the performance of the system in terms of latency and throughput. This is an important limitation of their study because the performance of a blockchain system is critical for its adoption and scalability. Without a thorough performance evaluation, it is difficult to determine whether the Ethereum consortium blockchain framework is suitable for large-scale applications that require high throughput and low latency. Another limitation of the study is that it focused solely on the Ethereum consortium blockchain framework, without considering other blockchain frameworks or platforms. This narrow focus limits the generalizability of the findings and makes it difficult to compare the performance and features of different blockchain frameworks [Liang & Zhao \(2020\)](#).

The framework supports on-chain and off-chain transactions, which can be used to create flexible and efficient payment systems. The framework allows for cloud deployment, enabling developers to deploy and manage their dApps on cloud platforms like AWS or Microsoft Azure. Ethereum also provides pseudonymity, which is useful for privacy-sensitive applications such as financial applications or voting systems. Access control enables developers to restrict access to certain parts of their dApps, ensuring that only authorized users can access sensitive information or perform certain actions. Ethereum uses a consensus mechanism called proof-of-work (PoW), which ensures that the blockchain is secure and immutable. However, there are limitations to the Ethereum framework. The current implementation of PoW consumes a significant amount of energy, which is not environmentally sustainable. The on-chain transaction throughput of Ethereum is limited, which can result in slow and expensive transactions during periods of high network congestion. Solidity, Ethereum's smart contract language, is not always easy to use, and developers must be careful to avoid vulnerabilities that could lead to security breaches. Additionally, the pseudonymity feature of Ethereum can also be a double-edged sword, as it can be used to facilitate illegal activities on the blockchain, such as money laundering or terrorism financing [Nizamuddin et al. \(2019\)](#).

A comprehensive analysis of the Ethereum public blockchain framework and found that it supports the execution of smart contracts, which are self-executing contracts with the terms of the agreement between buyer and seller being directly written into lines of code. Smart contracts allow for the creation of decentralized applications (dApps) that can automate complex processes, reduce costs, and increase efficiency. The authors also noted that Ethereum provides transparency, as all transactions are recorded on the blockchain, which enables anyone to verify the authenticity of a



transaction or digital asset. However, the authors also identified several limitations of the Ethereum framework that should be considered when designing a blockchain-based document verification system. Firstly, the on-chain transaction throughput of Ethereum is limited, which can result in slow and expensive transactions during periods of high network congestion. Secondly, the current implementation of the PoW consensus mechanism consumes a significant amount of energy, which is not environmentally sustainable. Thirdly, Ethereum's smart contract language, Solidity, is not always easy to use, and developers must be careful to avoid vulnerabilities that could lead to security breaches. In addition to these limitations, it is also important to consider the issue of interoperability with other blockchain systems. In order for a document verification system to be effective, it must be able to interact with other blockchain systems that are used by different organizations. This requires the development of standard protocols and interfaces that can be used by different blockchain systems to communicate with each other [Lamkoti et al. \(2021\)](#).

The authors did not mention the specific framework they examined. They found that the framework supports on-chain and off-chain transactions, cloud deployment, and consensus. However, the authors did not explore the framework's pseudonymity, and access control features and evaluate the scalability of their system [Nouman et al. \(2021\)](#).

The Hyperledger permissioned blockchain framework is suitable for enterprise-level applications due to its support for on-chain and off-chain transactions, pseudonymity, access control, and consensus mechanisms. The authors also highlighted the framework's scalability and interoperability features, which make it easier to integrate with existing systems. However, the authors did not thoroughly analyze the security features of the Hyperledger framework and did not explore the limitations of the consensus mechanism. Additionally, the framework may not be as decentralized as some other blockchain frameworks, which could affect its resilience to attacks and its ability to provide a truly trustless system. In terms of limitations for a blockchain-based document verification system, it is important to note that the Hyperledger framework may not provide the same level of privacy as other blockchain frameworks, as it is designed for permissioned networks where all participants are known and trusted. Additionally, the framework may require more complex setup and maintenance compared to other blockchain frameworks, which could be a barrier for smaller organizations or individuals looking to use the system. Finally, as with any blockchain-based system, scalability remains a key challenge that needs to be addressed, especially for applications like document verification systems that may require a high throughput of transactions [Liu & Guo \(2019\)](#).

The Ethereum public blockchain framework supports on-chain transactions, pseudonymity, and access control. However, the authors did not explore the off-chain and consensus

features of the framework, which are important for scalability and security. Additionally, the authors did not evaluate the performance of their system in terms of latency and throughput, which are important for real-world applications. The study also did not address the issue of energy consumption, which is a significant concern for public blockchains that use proof-of-work consensus mechanisms. Furthermore, the authors did not provide a detailed analysis of the security vulnerabilities of their system, which is critical for a blockchain-based document verification system that requires secure storage and transfer of sensitive data [Leka & Selimi \(2021\)](#).

The Hyperledger permissioned blockchain framework provides pseudonymity and access control features, which are particularly useful for enterprise-level applications that require privacy and security. The framework allows users to transact on the blockchain without revealing their real identities and enables developers to restrict access to sensitive information or certain actions, ensuring that only authorized users can perform them. However, the authors did not explore the on-chain, off-chain, cloud deployment, and consensus features of the framework, which are crucial for building a scalable and efficient blockchain-based document verification system. Additionally, the authors did not address the issue of interoperability with other blockchain systems, which is essential for ensuring that the system can communicate and exchange data with other platforms. Therefore, while the pseudonymity and access control features of Hyperledger are promising for enterprise-level applications, further research is needed to evaluate the framework's overall suitability for building a blockchain-based document verification system [Ghazali & Saleh \(2018\)](#).

The Ethereum public blockchain framework supports on-chain and off-chain transactions, cloud deployment, pseudonymity, access control, and consensus. However, the authors did not explore the limitations of the framework and address the issue of interoperability with other blockchain systems [Athanere & Thakur \(2022\)](#).

The Bitcoin public blockchain framework supports on-chain transactions, pseudonymity, and access control. However, the authors did not explore the framework's off-chain, cloud deployment, and consensus features and address the issue of data storage and management [Vidal et al. \(2019\)](#).

The Ethereum public blockchain framework supports on-chain transactions, cloud deployment, pseudonymity, and access control. However, the authors did not explore the off-chain and consensus features of the framework and evaluate the scalability of their system [Faaroek et al. \(2022\)](#).

The Hyperledger private blockchain framework and found that it supports on-chain transactions, pseudonymity, access control, and consensus. However, the authors did not explore the cloud deployment feature of the framework and provide a thorough analysis of their system's security [Saleh et al. \(2020\)](#).

The system uses public blockchains for secure data sharing in e-commerce applications. The system uses Ethereum blockchain to implement a decentralized platform for secure transactions between buyers and sellers. The proposed system utilizes access control mechanisms and pseudonymity to ensure data privacy and security. The study highlights the advantages of using blockchain technology, such as decentralization, transparency, and immutability. However, the limitation of this study is the lack of real-world implementation and testing and addressing the issue of interoperability with other blockchain systems [Ghazali & Saleh \(2018\)](#).

The blockchain-based system focuses on a secure and decentralized storage of sensitive data. The proposed system uses Ethereum blockchain to ensure secure and efficient storage of data. The framework employs access control mechanisms and pseudonymity to ensure data privacy and security. The study demonstrates the advantages of using blockchain technology, such as data immutability, transparency, and security [Vidal et al. \(2019\)](#).

The proposed system is based on the Ethereum blockchain and utilizes smart contracts to manage access control and data sharing. The framework employs off-chain storage for storing large data files and uses on-chain transactions to record access control rules and audit trails [Athanere & Thakur \(2022\)](#).

The blockchain-based solution for secure and private data sharing in the healthcare domain is based on the Hyperledger Fabric blockchain and uses smart contracts to enforce access control and data sharing policies. The framework employs off-chain storage for storing sensitive data and utilizes on-chain transactions to record access control rules and audit trails [Dalal et al. \(2020\)](#).

The objective of the initiative by Open Univeristy is not centered on the end-user and it is not accessible with the intention of meeting the needs of third-parties, employers, and so on. The average user may find it challenging to comprehend the functioning of blockchain and may need the assistance of a technical intermediary. Although end-to-end encryption is used to safeguard the privacy of users in their model, there are potential risks due to the release of private data on a public blockchain. In such a situation, there is no mechanism in place to safeguard the recipient's privacy and ownership of the data [Domingue \(2017\)](#).

While the University of Nicosia (UNIC) has put in place several measures to maintain the confidentiality, ownership, and integrity of certificates, there is a need for further enhancements to enable the hash to be publicly validated, which is a crucial requirement for employers to be able to view the certificate; moreover, the recipient of the certificate may not have the necessary authorization to permit a potential employer to verify the authenticity of the certificate using the hash [Bond et al. \(2015\)](#).

Although hash or digitally signed certificates offer enhanced security, there are still some potential difficulties in accessing them for legitimate users due to the risk of

cyber attacks. Since computer systems are vulnerable to intrusion by unauthorized individuals, accessing and verifying these certificates may pose a challenge. It is also worth noting that while cryptography provides a strong layer of protection against data tampering, it does not ensure complete data security. As such, it is important to implement additional security measures to guard against potential threats. Despite these challenges, the use of cryptographically secured certificates in SmartCert is an effective means of reducing the risk of certificate forgery and improving the reliability of academic credential verification [Kanan et al. \(2019\)](#).

While Records Keeper presents a relatively simple blockchain-based solution for academic certificate verification, there are some limitations to consider. For example, it is important to note that parties interested in viewing a certificate stored in the Record Keeper blockchain must have ownership rights, which could result in a transfer of ownership to the third-party. This transfer could, in turn, lead to tampering with the certificate, compromising its authenticity and accuracy. As such, while this solution may be effective on a private blockchain where access to the blockchain is limited and carefully controlled, it may not be suitable for use on a public blockchain where access is more open and potentially less secure. Ultimately, it is important for educational institutions and other stakeholders to carefully consider the specific needs and requirements of their certificate verification system and to choose a blockchain-based solution that is best suited to those needs [Desai \(2018\)](#).

The research paper presents valuable insights into the application of hash functions in blockchain systems and proposes a fresh approach for integrating blockchain technology into certificates. However, there is a need for a more extensive evaluation of the proposed technique to determine its effectiveness in ensuring the security of data. The verification process lacks independent user validation, and personal information is only included in the shared paper, which raises privacy concerns because of the public nature of blockchain, which is not GDPR compliant by default. Additionally, since there is a lack of personal information and user validation, it is challenging to establish a clear association between the certificate and its owner [Rahardja et al. \(2021\)](#).

The system is accessible to three groups of users: schools or certification units, graduates, and companies or employment agencies. This enhances accessibility to certificates and verification for employment purposes. It uses access control to restrict access to the system database to schools or certification units. Graduates are only able to access their own certificate information, and companies or employment agencies can only access certificate information of job applicants. However, there is no information about the specific access control mechanisms used to restrict access to the system. It also does not provide much information about confidentiality measures, such as encryption or anonymization, that may have been implemented to protect the confidentiality of the data stored on the blockchain [Cheng et al. \(2018\)](#).

The data layer uses the on-chain approach to store data in the blockchain, with a timestamp as proof of integrity, hash function to ensure collision resistance, and digital multi-signature to collect votes from nodes. The architecture also includes an in-chain revocation list, and a dissemination mechanism to share information with network nodes and verification application users. Additionally, a peer-to-peer network and an authentication mechanism are included to permit users to perform their roles in the system. The on-chain approach used for data storage might lead to scalability issues in large systems, and the use of a private blockchain can limit interoperability. There is also no discussion of how consensus is achieved among trusted members and what happens in the event of a failure or an attack [Hasan et al. \(2020\)](#).

The proposed system assumes that there are trusted parties involved in the system already, and thus, we expect that the nodes can provide proof. However, achieving consensus among all parties involved may not always be feasible or straightforward. It also does not explicitly specify whether the proposed system is on-chain or off-chain, and this could impact the system's scalability and security. It mentions that the database is managed on servers with network connectivity, but it does not specify whether the servers are on-premises or cloud-based. This could impact the system's accessibility and security [Han et al. \(2018\)](#).

The proposal includes a private and permissioned blockchain to ensure anonymity of student records, and the subset of peer nodes selected to transact through the platform is determined using a hashing algorithm. The selection of peers may become ineffective in terms of security if it is random, and that the network up time is limited by this approach. The elected peer must be motivated to ensure that it does not attack the network, and computing power is proposed as a possible selection criterion. Another limitation is that the procedure should ensure fairness in miner node selection [Srivastava et al. \(2018\)](#).

The proposed system handles degree credentials or record data block creation and consensus verification and is implemented for robust information integrity preservation and secure interaction among stakeholders. The system is broadcast from the orderer to the committer on the blockchain hyperledger fabric for validation, and the hyperledger fabric uses the consensus algorithm to manage certificate identity verification and validation [Ayub Khan et al. \(2021\)](#).

To summarize, There are some existing solutions for educational certificate verification that were analyzed for their security themes, including authentication, authorization, confidentiality, ownership, and privacy. The authentication theme is addressed by RecordsKeeper and MIT solutions, while other solutions do not provide details on authentication. MIT and SmartCerts address authorization, but technical details are not available. Confidentiality is not ensured by any of the reviewed solutions. Ownership is managed by MIT and UNIC, while Smartcert and Recordskeeper have

shared ownership. Privacy is ensured by KMI OU UK and MIT solutions, while the other solutions do not provide much information on this theme.

Table 2.1: Comparative Analysis

| Author                    | Framework   | Blockchain Framework | On-Chain | Off-Chain | Cloud | Pseudonymity | Access Control | Consensus | User Centric | Confidentiality | Accountability | Cost Effective | Scalability | Autonomy | Fault Tolerance |
|---------------------------|-------------|----------------------|----------|-----------|-------|--------------|----------------|-----------|--------------|-----------------|----------------|----------------|-------------|----------|-----------------|
| Shakan et al. (2021)      | Ethereum    | Consortium           | ✓        | ✓         | ✓     | ✓            | ✓              | ✓         | ✓            |                 |                | ✓              |             | ✓        |                 |
| (Liang et al., 2020)      | Ethereum    | Consortium           | ✓        | ✓         | ✓     | ✓            | ✓              | ✓         | ✓            |                 |                | ✓              | ✓           | ✓        | ✓               |
| (Arshad et al., 2019)     | Ethereum    | Public               | ✓        | ✓         | ✓     | ✓            | ✓              | ✓         | ✓            |                 | ✓              | ✓              | ✓           | ✓        |                 |
| (Maji et al., 2021)       | Ethereum    | Public               | ✓        | ✓         | ✓     | ✓            | ✓              | ✓         | ✓            | ✓               |                | ✓              |             | ✓        |                 |
| (Muhammad et al., 2022)   |             |                      | ✓        | ✓         | ✓     | ✓            | ✓              | ✓         | ✓            |                 |                | ✓              |             | ✓        |                 |
| (Liu et al., 2019)        | Hyperledger | Permissioned         | ✓        | ✓         | ✓     | ✓            | ✓              | ✓         | ✓            |                 | ✓              | ✓              | ✓           | ✓        | ✓               |
| (Leka et al., 2021)       | Ethereum    | Permissioned         | ✓        | ✓         | ✓     | ✓            | ✓              | ✓         | ✓            |                 | ✓              | ✓              | ✓           | ✓        | ✓               |
| (Ghazali et al., 2018)    | Hyperledger | Permissioned         | ✓        | ✓         | ✓     | ✓            | ✓              | ✓         | ✓            |                 | ✓              | ✓              | ✓           | ✓        | ✓               |
| (Athanere et al., 2022)   |             |                      | ✓        | ✓         | ✓     | ✓            | ✓              | ✓         | ✓            |                 |                | ✓              |             | ✓        |                 |
| (Vidal et al., 2019)      | Ethereum    | Public               | ✓        | ✓         | ✓     | ✓            | ✓              | ✓         | ✓            |                 |                | ✓              | ✓           | ✓        |                 |
| (Faarooq et al., 2022)    | Hyperledger | Private              | ✓        | ✓         | ✓     | ✓            | ✓              | ✓         | ✓            |                 |                | ✓              | ✓           | ✓        |                 |
| (Saleh et al., 2020)      |             |                      | ✓        | ✓         | ✓     | ✓            | ✓              | ✓         | ✓            |                 |                | ✓              | ✓           | ✓        |                 |
| (Subathra et al., 2022)   |             |                      | ✓        | ✓         | ✓     | ✓            | ✓              | ✓         | ✓            | ✓               | ✓              | ✓              | ✓           | ✓        |                 |
| (Ghazali et al., 2022)    |             |                      | ✓        | ✓         | ✓     | ✓            | ✓              | ✓         | ✓            | ✓               | ✓              | ✓              | ✓           | ✓        | ✓               |
| (Lamkoti et al., 2021)    | Ethereum    | Public               | ✓        | ✓         | ✓     | ✓            | ✓              | ✓         | ✓            |                 |                | ✓              |             |          |                 |
| (Vidal et al., 2019)      | Ethereum    | Public               | ✓        | ✓         | ✓     | ✓            | ✓              | ✓         | ✓            |                 |                | ✓              |             |          |                 |
| (Athanere et al., 2022)   | Ethereum    | Public               | ✓        | ✓         | ✓     | ✓            | ✓              | ✓         | ✓            |                 |                | ✓              |             |          |                 |
| (Dalal et al., 2020)      |             |                      | ✓        | ✓         | ✓     | ✓            | ✓              | ✓         | ✓            |                 |                | ✓              | ✓           |          |                 |
| (Nouman et al., 2021)     | Bitcoin     | Public               | ✓        | ✓         | ✓     | ✓            | ✓              | ✓         | ✓            | ✓               |                |                |             |          |                 |
| (Rahardja et al., 2021)   |             |                      | ✓        | ✓         | ✓     | ✓            | ✓              | ✓         | ✓            |                 |                |                |             |          |                 |
| (Cheng et al., 2017)      | Ethereum    | Public               | ✓        | ✓         | ✓     | ✓            | ✓              | ✓         | ✓            |                 |                | ✓              |             | ✓        | ✓               |
| (Hasan et al., 2020)      |             |                      | ✓        | ✓         | ✓     | ✓            | ✓              | ✓         | ✓            |                 |                |                |             |          |                 |
| (Han et al., 2018)        | Ethereum    | Private              | ✓        | ✓         | ✓     | ✓            | ✓              | ✓         | ✓            |                 | ✓              |                |             | ✓        |                 |
| (Srivastava et al., 2018) |             |                      | ✓        | ✓         | ✓     | ✓            | ✓              | ✓         | ✓            |                 |                |                |             | ✓        |                 |
| (Ayub et al., 2021)       | Hyperledger | Consortium           | ✓        | ✓         | ✓     | ✓            | ✓              | ✓         | ✓            | ✓               | ✓              | ✓              |             | ✓        |                 |

# Chapter 3

## Methodology

Methodology refers to the systematic and theoretical analysis of strategies and approaches within a particular field of study. It involves examining the collection of strategies and guidelines that guide the acquisition and application of knowledge. It is important to note that a methodology does not aim to produce specific results, unlike a system. Instead, it provides the theoretical foundation for understanding which systems, styles, or practices can be employed in a given context to achieve a desired outcome.

In the context of software analysis and design, methodology plays a crucial role in facilitating the transition from requirement specification to implementation. It serves as an intermediate stage where human-readable requirements are transformed into executable code. Examples of systems analysis include modifying computer code to accomplish a task, fixing a malfunctioning air conditioning system, or analyzing daily routines to prevent errors. For the following system we are focusing on hybrid methodology consists of incremental and iterative.

### 3.1 Introduction

The methodology employed for implementing the academic credential verification system involves the development of a smart contract that governs the issuance, verification, and revocation of certificates. This smart contract is deployed on the blockchain network, allowing certificates to be stored as transactions. The certificates are linked to their respective issuers, and authorized users can verify their authenticity by leveraging the transparency and immutability of the blockchain.



## 3.2 System Development Model

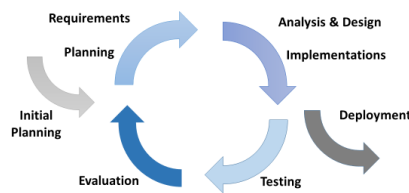


Figure 3.1: Hybrid Software Development Model

### 3.2.1 Incremental

The system's development will be split up into small, manageable chunks, or iterations, with each one concentrating on meeting a specific set of the system's criteria. For the deve

The incremental development method enables the early development and delivery of crucial system features and functionality. The fundamental features, such certificate verification, will be implemented in the earliest versions. This makes sure that the system can already be used and of benefit, even if it doesn't currently have all the features that are ideal.

The work completed in earlier iterations will be built upon in later iterations, which will gradually add additional features and improve functionality that are already there. Each iteration will have a cyclical structure and include the analysis of the requirements, the design, the implementation, the testing, and the deployment. This iterative process enables ongoing development and enhancement of the system throughout the development process.

The development team may manage complexity and reduce risks efficiently by using an incremental method. Iterative development provides for the early detection and resolution of possible problems, ensuring that the system stays on course and achieves the required goals. Additionally, it offers a chance for obtaining user input, which may then be used to inform later iterations and make the system more responsive to users' wants and requirements.

Overall, the incremental method makes it possible to construct the academic credential verification system in a flexible and adaptive manner. It enables effective risk management, stakeholder participation, early delivery of critical functionality, and ongoing system improvement throughout the development life-cycle.

### **3.2.2 Iterative**

In an iterative approach, the development is divided into smaller iterations or cycles, with each cycle concentrating on a particular subset of the system's requirements. The requirements analysis, design, implementation, testing, and assessment phases will all be completed by the development team throughout each iteration.

The earliest iterations will focus on creating and proving the system's fundamental features, namely certificate issuing and fundamental verification. These iterations act as a platform for further iterations, which add new features and improve existing ones in response to user input and changing requirements.

A constant feedback loop exists during each iteration, enabling system evaluation and improvement. The system may develop and get better over time thanks to user feedback, testing findings, and lessons learnt from each iteration.

The iterative method has a number of benefits for creating a system for verifying academic credentials. The first benefit is that it enables early and frequent delivery of functional increments, allowing stakeholders to offer comments and confirm the system's development. This cycle of repeated feedback makes sure that the system satisfies the consumers' wants and expectations.

The iterative method also encourages adaptation and flexibility. The system may change and adapt to changing demands and conditions by incorporating new requirements and insights as they become available into later versions.

The iterative technique also enables efficient risk management. The project's total risk can be decreased by segmenting the development process into smaller iterations and identifying and addressing any risks and problems early on.

Overall, the iterative process offers a structured and flexible method for creating the system for verifying academic credentials. It enables ongoing improvement, adaptability, stakeholder involvement, and risk reduction, all of which contribute to the successful deployment of a strong and user-centered system.

## **3.3 System Analysis**

System analysis is an analytical strategy used to explain to customers or stakeholders the specifics and operation of a system. It include outlining the system's goals, functionality, implementation process, and any intended changes or advancements in the future. To promote efficient communication and decision-making throughout the development and maintenance stages, system analysis seeks to give a thorough knowledge of the system's characteristics, procedures, and capabilities.

### **3.3.1 System Design**

The process of designing and organizing the execution of a system is referred to as system design. It entails responding to queries and worries that consumers could have regarding the system's look and operation. System design acts as an early step in the system development process before a system is actually implemented. It includes the preliminary work of defining the system's structure, the interactions between its many parts, and its operational principles. The total development process is significantly shaped by system design, which establishes the groundwork for a successful system implementation.

### **3.3.2 Coding**

Implementing the certificate verification system requires coding. It entails setting up and maintaining a secure database and employing query languages for data management and retrieval. The user interface is designed using HTML, CSS, EJS, and JavaScript, while Node.js is utilized for server-side functionality. The issue, verification, and revocation of certificates are controlled by smart contracts that are written in blockchain code. Overall, coding enables the system to safely and dependably check academic qualifications.

### **3.3.3 Testing**

The development of the certificate verification system includes several steps that involve software testing. Before finishing them, developers test certain components, such as building user interface elements like the home page, carousel, or navigation drawer. The creation of new entities or characteristics, as well as maintaining the smooth transfer of data between the user interface and the database, present difficulties when the database structure is changed. To assure the integrity and operation of the application at each stage, extensive testing is done on the back-end procedures.

### **3.3.4 Implementation**

The research paper's implementation phase focuses on the actualization of the blockchain-based system for verifying academic certificates. The methods, technologies, and technical specifics used to construct the system are covered in this section. It examines the important elements, including the implementation of smart contracts, blockchain integration, and user interface development, and it details the procedures used to transform the requirements into a practical solution. The system's construction methods, frameworks, and tools are highlighted throughout the implementation phase.

## **3.4 Requirement Engineering**

The requirements for the academic credential verification system are collected, analyzed, documented, and validated throughout the requirement engineering process. The procedure for compiling and comprehending the requirements and goals of stakeholders, such as educational institutions, employers, and credential holders, is described in this part. It focuses on how crucial it is to record both functional and non-functional needs, and it looks at methods for managing requirements throughout the development life cycle.

## **3.5 Requirement Engineering Process**

The section on the requirement engineering process gives a general overview of the methodical procedure used to determine, record, and validate the requirements for the system for verifying academic credentials. The phases that must occur, such as the feasibility study, requirement collection, software requirement specification, and software requirement validation, are discussed. This section illustrates the approaches used to assure the correctness and completeness of the requirements and also discusses the difficulties and factors to be taken into account for each stage.

### **3.5.1 Feasibility Study**

Assessing the potential and practicality of developing a blockchain-based system for examining academic credentials is the aim of the feasibility research stage. This subsection covers the study that was conducted to assess the technical, operational, economic, and scheduling viability. It examines the potential benefits, drawbacks, and limits associated with the system's implementation and provides judgments that have an impact on the decision to continue with the development.

### **3.5.2 Requirement Gathering**

The needs and expectations of the stakeholders for the academic credential verification system are actively obtained and recorded throughout the requirement collecting stage. The techniques used to gather and document the requirements, including as workshops, questionnaires, and interviews, are described in this section. It emphasizes the value of effective coordination and contact with the stakeholders to ensure a complete understanding of their needs.

### **3.5.3 Software Requirement Specification**

At the software requirement definition stage, the functional and non-functional requirements for the academic credential verification system are outlined. This area presents the requirements for the system's behavior, interfaces, performance, security, and other relevant characteristics. Use case diagrams, entity-relationship diagrams, and user stories are a few examples of common documentation techniques that are discussed in this article for effectively documenting requirements.

### **3.5.4 Software Requirement Validation**

Validating software requirements entails making sure that the written specifications are correct, consistent, and comprehensive. The methods and strategies used to examine, confirm, and validate the requirements are covered in this subsection. To find and address any ambiguities, inconsistencies, or holes in the requirements, it underlines the value of incorporating stakeholders and subject matter experts in the validation process.

## **3.6 Software Requirement for Development**

The following elements are part of the software requirements for developing the prototype. Firstly, the basis for storing and confirming academic credentials is a strong and secure blockchain platform, like Ethereum. The development team makes sure that smart contracts are integrated for verifying certificates and retrieve certificates. For the temporary storing and retrieval of certificates, a scalable and trustworthy IPFS implementation is used. To protect sensitive data, the program has to have encryption and access control features. Employers and educational institutions need an easy-to-use interface to search for and access certified certifications. The system also has features for resolving errors, handling verification request requests, and submitting certificates. During the development process, compliance with pertinent data protection laws and compatibility with current systems are taken into account. Regular maintenance, monitoring, and updates are incorporated to ensure the longevity and efficiency of the software.

### **3.6.1 Visual Studio 2023**

Visual Studio 2023 is a powerful IDE for developing the academic credential verification model. [Beck \(2018\)](#) It supports popular programming languages like JavaScript, and Solidity, offering intuitive code editors, debugging tools, and version control systems. With containerization support and a rich ecosystem of extensions, Visual Studio 2023

enhances productivity and enables seamless integration with additional tools and frameworks [Krizhevsky et al. \(2012\)](#) It is an ideal choice for creating a reliable and scalable solution using blockchain and IPFS

### 3.6.2 Microsoft Visual Studio Code

Microsoft Visual Studio Code (VS Code) is a lightweight yet powerful code editor ideal for developing the academic credential verification model. It supports multiple programming languages and offers customizable features and extensions that boost productivity. With a built-in terminal and Git integration, it facilitates command-line interactions and collaboration [Bhutta et al. \(2021\)](#). The extensive marketplace provides specialized blockchain development extensions, enhancing functionality for working with blockchain and IPFS. With cross-platform compatibility, VS Code ensures flexibility and accessibility. [Dongre et al. \(2020\)](#) Overall, it offers a versatile and efficient development environment for writing, debugging, and deploying code, leveraging its extensive ecosystem of tools and extensions.

### 3.6.3 Go

Go (Golang) is a cutting-edge and effective programming language that is ideal for creating the model for verifying academic credentials [Dai et al. \(2022\)](#) It features a straightforward syntax, robust concurrent programming capabilities, and extensive security and data management packages [Desai \(2018\)](#). Go's quick execution, small memory footprint, and static typing guarantee top efficiency and code dependability. Go speeds the development process by fusing simplicity, performance, and parallelism. It has a thriving community and a wealth of open-source resources [Kiffer et al. \(2018\)](#).

### 3.6.4 IPFS

IPFS (Interplanetary File System) is a decentralized file storage protocol that securely stores and retrieves data using a peer-to-peer network. It chunks and hashes files across multiple nodes, ensuring redundancy and fault tolerance [Nyalety et al. \(2019\)](#). In the academic credential verification model, IPFS acts as a storage intermediary, generating unique hash codes for certificates and reducing costs by storing data off the blockchain [Sohail et al. \(2020\)](#). It enables efficient content addressing for certificate retrieval, enhancing scalability, security, and cost-effectiveness.

### 3.6.5 NodeJS

Node.js is a versatile JavaScript runtime environment for server-side development. It offers high scalability and performance, making it suitable for handling concurrent

operations and processing large volumes of data. Node.js can be used to build the server-side logic for handling certificate verification requests, interacting with the blockchain network, and integrating with IPFS. Its vast ecosystem of libraries and frameworks provides extensive functionality for web server development, API creation, and database connectivity. With an active community and abundant resources, Node.js is a robust and efficient platform for developing the academic credential verification model, leveraging the power of JavaScript across the application stack [Gourisetti et al. \(2019\)](#).

### 3.6.6 Metamask

Metamask is a browser extension and wallet that enables users to interact with Ethereum-based decentralized applications (dApps) from their web browser. It is valuable for the development of the academic credential verification model, providing secure authentication, transaction signing, and smart contract interaction. Metamask simplifies Ethereum account management and offers a user-friendly interface. Its JavaScript library facilitates seamless integration with web applications. Metamask enhances the user experience and simplifies the certificate verification process. With its documentation and developer community, Metamask is a valuable tool for leveraging blockchain technology in the academic credential verification model [Hasan et al. \(2020\)](#).

### 3.6.7 Truffle

Truffle is a popular Ethereum development framework for building decentralized applications (dApps), including the academic credential verification model. It simplifies smart contract compilation, deployment, and testing. Truffle's standardized project structure and built-in testing framework enhance organization and security. Integration with tools like Ganache enables local blockchain simulation. The JavaScript-based API facilitates interaction with deployed smart contracts. Truffle's documentation and community support provide valuable resources for developers. Overall, Truffle accelerates dApp development, including the academic credential verification model, with its robust features and Ethereum integration [Natarajan et al. \(2017\)](#).

### 3.6.8 Mendeley

Mendeley is a reference management tool and academic social network, beneficial for the academic credential verification model. It organizes scholarly literature and references. Mendeley integrates citations and references accurately. Collaboration features enhance productivity. Its extensive library supports research and best practices. Mendeley provides plugins and APIs for seamless integration. Its user-friendly

interface is cross-platform compatible. Mendeley streamlines reference management, collaboration, and access to scholarly resources for the verification model [Foytik et al. \(2020\)](#).

### **3.6.9 LaTeX**

LaTeX is a widely-used typesetting system for academia and scientific research, beneficial for the academic credential verification model. It creates professional documents with precise formatting. LaTeX ensures industry-standard documentation, adhering to conventions. Its mathematical typesetting capabilities handle complex formulas. LaTeX offers packages for citations, references, and bibliographies. Integration with version control facilitates collaboration. Cross-referencing features create well-structured documents. LaTeX separates content from formatting, focusing on content creation. Graphics and charts can be included to represent the verification process or present data. LaTeX enhances professionalism and consistency in project documentation [Hellman \(2002\)](#).

### **3.6.10 Draw.io**

Draw.io is a versatile online diagramming tool that benefits the development of the academic credential verification model. It creates various diagrams and visual representations. Draw.io offers customizable shapes and templates. Its intuitive interface and drag-and-drop functionality enable easy diagram creation. Real-time collaboration enhances teamwork. Draw.io supports various export options. It is web-based and accessible from any device. Draw.io visualizes and documents the verification model effectively, aiding understanding and communication [Lamkoti et al. \(2021\)](#).

### **3.6.11 Xampp**

XAMPP is an open-source web server solution crucial for the academic credential verification model. It creates a local server environment for testing and development. XAMPP includes Apache for seamless web application execution and MySQL for database management. It supports PHP and Perl for server-side scripting and dynamic web applications. The user-friendly control panel simplifies server management. XAMPP is cross-platform and allows for additional component installation. It ensures compatibility and a smooth development process [Sathya et al. \(2021\)](#).

### **3.6.12 Lucid chart**

Lucidchart is a cloud-based visualization tool that benefits the development of the academic credential verification model. It creates professional diagrams and visual



representations of the system's architecture and workflows. Lucidchart offers customizable templates and intuitive drag-and-drop functionality. Real-time collaboration and integration with productivity tools enhance teamwork. Lucidchart diagrams can be embedded in documentation and shared easily. The tool's cloud-based nature enables access from any device, and various export options facilitate integration into project materials. Lucidchart simplifies diagram creation and effectively communicates the verification model's complexities [Kosba et al. \(2016\)](#).

## 3.7 Required Programming Languages

Proficiency in Solidity, JavaScript, Node.js, HTML, CSS, and database management languages is essential for developing the academic credential verification model. Solidity is needed for smart contract development on the Ethereum blockchain. JavaScript is crucial for web-based components and interacting with smart contracts. Node.js facilitates server-side applications and blockchain integration. HTML and CSS are necessary for web design and structure. Database management languages handle data storage and retrieval. Proficiency in these languages ensures seamless integration and efficient development of the verification model [Spanò et al. \(2022\)](#).

### 3.7.1 HTML

HTML is essential for designing the user interface of the academic credential verification model. It provides the structure and layout of web pages, including interactive elements like forms and buttons. HTML ensures accessibility and compatibility across devices and browsers. It also allows integration of media elements. Combined with CSS, HTML enables consistent and visually appealing designs. In summary, HTML forms the foundation of the web interface, presenting information and functionality in a clear and intuitive manner [Steichen et al. \(2018\)](#).

### 3.7.2 CSS

CSS is essential for the academic credential verification model's user interface. It defines the visual style, including layout, colors, fonts, and responsiveness. CSS ensures a cohesive and aesthetically pleasing design across different devices. It enhances usability and readability, and enables interactive elements and animations. In summary, CSS plays a vital role in creating an appealing and user-friendly interface for the verification model [Vidal et al. \(2019\)](#).

### 3.7.3 EJS

EJS (Embedded JavaScript) is a dynamic templating engine for the academic credential verification model. It enables developers to embed JavaScript code within HTML templates, allowing for dynamic content generation and customization. EJS supports conditionals, loops, and partials, promoting code organization and reusability. It seamlessly integrates with Express.js, facilitating efficient routing and rendering of EJS templates. EJS enhances the user experience and code maintainability by providing a flexible and efficient solution for rendering dynamic content [Glaser et al. \(2019\)](#).

### 3.7.4 Java Script

JavaScript (JS) is crucial for the academic credential verification model. It creates dynamic and interactive user interfaces, enhancing the user experience. JS enables form validation, real-time updates, and interactive elements. Frameworks like React, Angular, or Vue.js streamline UI development. JS supports asynchronous programming for efficient communication with backend services. Its vast ecosystem offers libraries and modules for specific features. With extensive resources and community support, JS is fundamental for developing the verification model, providing dynamic interfaces, seamless integration, and access to tools and libraries [Kosba et al. \(2016\)](#).

### 3.7.5 NodeJs

Node.js is vital for the academic credential verification model. It executes server-side JavaScript code and enables robust and scalable server applications. With its event-driven, non-blocking architecture, Node.js ensures high-performance handling of concurrent requests. It offers a wide range of libraries and frameworks like Express.js for web development and API interaction. Node.js facilitates seamless database integration and promotes code reuse and consistency. Overall, Node.js is essential for building a scalable and efficient server-side application in the verification model [Kiffer et al. \(2018\)](#).

### 3.7.6 Solidity

Solidity is vital for the academic credential verification model. It is a language for writing Ethereum smart contracts. Solidity enables secure and efficient contracts with features like inheritance and interfaces. It supports cryptographic operations for data integrity. With Solidity, developers create transparent and tamper-proof smart contracts for academic credential verification. It ensures trust and immutability in the verification process. Solidity is crucial for implementing core functionality on the Ethereum blockchain [Hasan et al. \(2020\)](#).

## 3.8 Use case Diagram

### 3.8.1 Use Case Diagram for Applicants

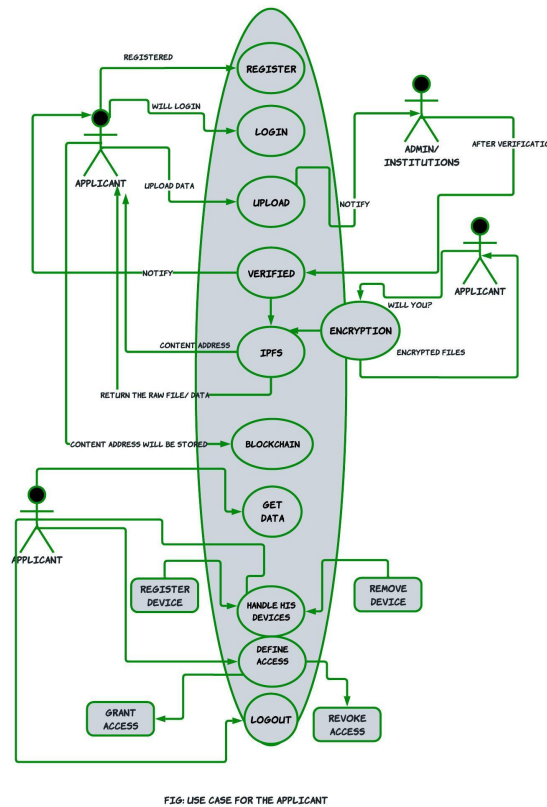


Figure 3.2: Applicant Use Case

Figure 3.2 depicts the applicant's use case. In order to utilize the system, the applicant must first register and log in. The applicant can upload their credentials in the system and request the admin for verification. While the verification process is still ongoing, the certificates will be temporarily stored in a local database. When a company requests access to view the credentials, the applicants have the option to review the access request and accept it accordingly.

### 3.8.2 Use Case Diagram for Admin

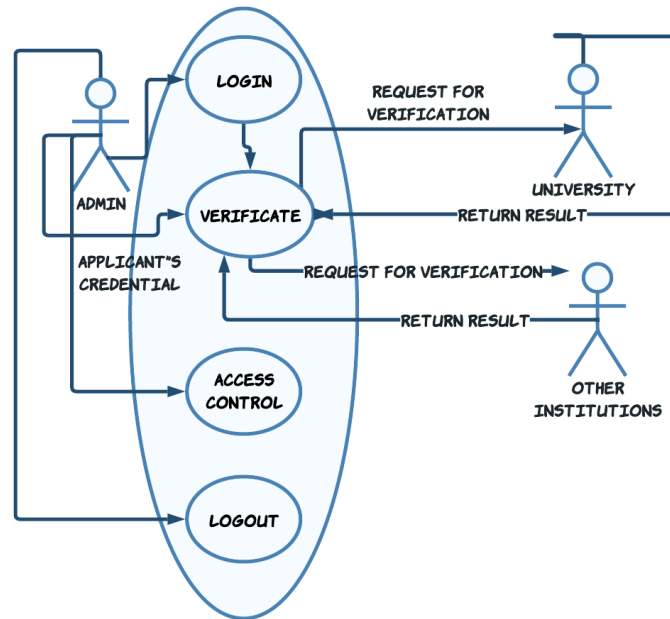


FIG: USE CASE FOR THE ADMIN

Figure 3.3: Admin use case

Figure 3.3 illustrates the administrator's use case. The administrator's job is to receive verification requests from applicants and contact certificate providers such as educational institutions to check the certificates' legitimacy. They upload the confirmed results to the IPFS and notify the applicant once they get the results.

### 3.8.3 Use Case Diagram for Institutions

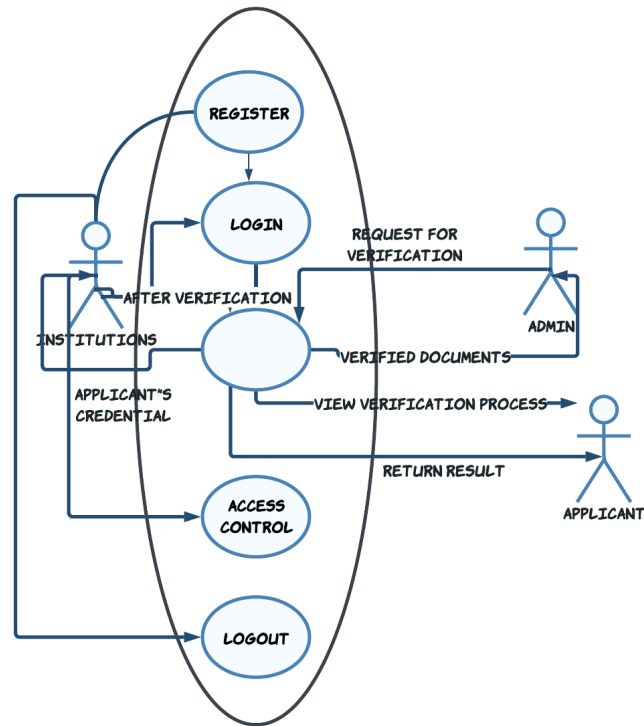


FIG: USE CASE FOR THE INSTITUTIONS

Figure 3.4: Institution use case

In Figure 3.4, we can observe the various activities of the educational institution within the proposed blockchain-based academic certificate verification system. Firstly, the institution has the ability to log in and register on the system, after which they can start receiving verification requests from applicants. These requests require the institution's involvement in verifying the authenticity and validity of the applicant's academic certificates.

### 3.8.4 Use Case Diagram for Company

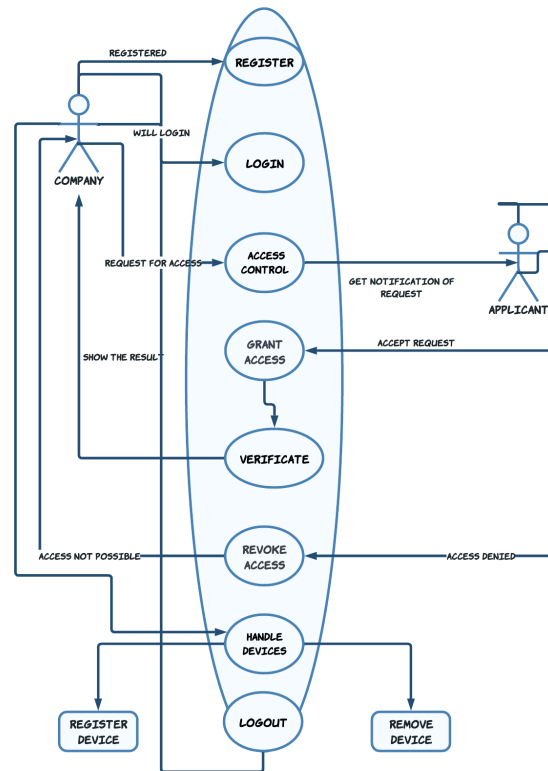


FIG: USE CASE FOR THE COMPANY

Figure 3.5: Company use case

Figure 3.5 shows the use case for the company. When the firm receives the hash key from the applicant, it uses it to search the system for the specific applicant. Because the system has access tiers for further protection, the company must first request access before seeing the credentials. The company is able to examine the certificates after the applicant approves the request from their account.

## 3.9 Activity Diagram

### 3.9.1 Activity Diagram for Applicants

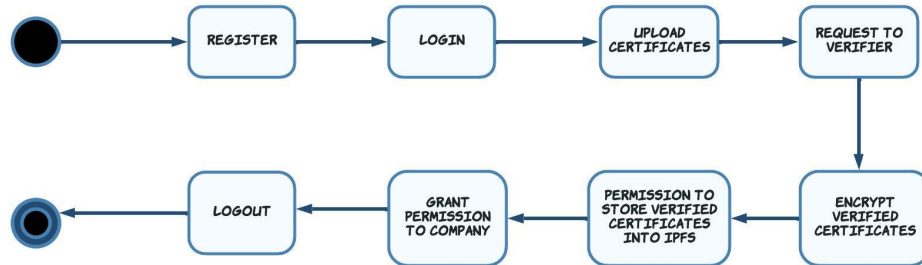


Figure 3.6: Applicant Activity

Figure 3.6 illustrates the applicant's activity in the system. To utilize the system, the applicant must register and log in. The applicant can upload their credentials to the system and request the admin to verify them. While the verification process is in progress, the certificates will be temporarily saved in a local database. When a company requests access to view the credentials, the applicants have the opportunity to review the access request and accept it as appropriate.

### 3.9.2 Activity Diagram for Admin

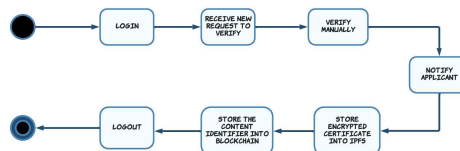


Figure 3.7: Admin Activity

Figure 3.7 illustrates the activity for the admin. The administrator is responsible for receiving verification requests from applicants and redirecting it to the relevant institution for validating the authenticity of the certificates. Once the verification is complete, the administrator uploads the confirmed results to the IPFS and blockchain.

### 3.9.3 Activity Diagram for Institutions

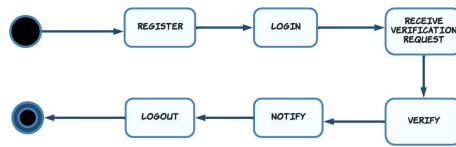


Figure 3.8: Institution Activity

In Figure 3.8, we can observe the various activities of the educational institution within the proposed blockchain-based academic certificate verification system. Firstly, the institution has the ability to log in and register on the system, after which they can start receiving verification requests from applicants. These requests require the institution's involvement in verifying the authenticity and validity of the applicant's academic certificates

### 3.9.4 Activity Diagram for Company

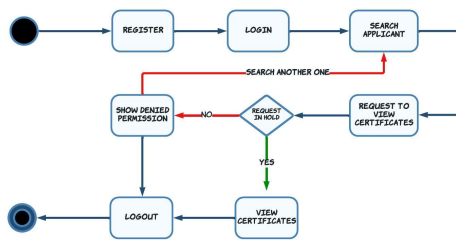


Figure 3.9: Company Activity

Figure 3.9 illustrates the activity for the company. The firm can register, login and use the system to search for any particular applicant. As the system has access tiers for added security, the company must initially request access before being able to view the credentials. Once the applicant approves the request from their account, the company can examine the certificates.



### 3.10 Sequence Diagram for Smart-Contract

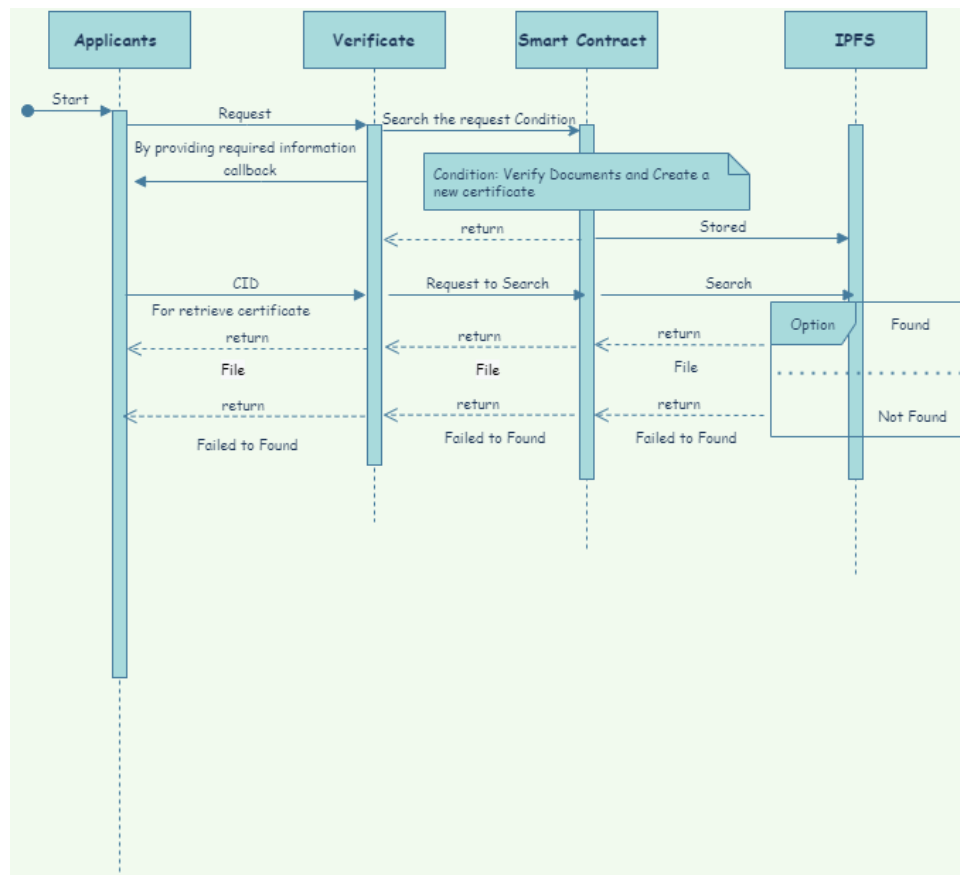


Figure 3.10: Sequence Diagram

Figure 3.10 demonstrates a sequence diagram for the system. The applicant initiates the certificate verification process by sending a verification request to the system. The system generates a digital representation of the academic certificate and uploads it to IPFS, obtaining a unique IPFS hash for the certificate. The system invokes the smart contract's verification function, passing the transaction ID and the IPFS hash of the certificate.

## 3.11 ER Diagram

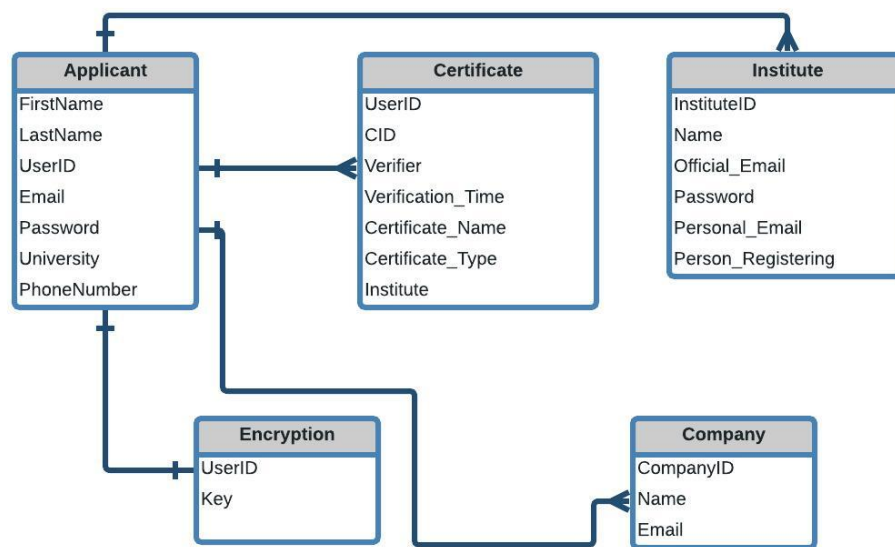


Figure 3.11: ER Diagram

Figure 3.11 shows the ER diagram of the system. The entities involved in the system are applicant, company, institute, certificate, encryption. Each entity has a unique identifying variable. The first three entities store key information about the stakeholders. The certificate entity makes sure it has information about the type of certificate, time of verification, and information about the stakeholder that verified it.

## 3.12 Overview

The first step in the certificate verification process involves an applicant uploading their certificate to the system. This initial upload places the certificate in temporary storage to ensure that it is not lost or compromised during the verification and encryption process. Once uploaded, the certificate is prepared for further processing, which includes verifying its authenticity and encrypting it for secure storage. This step ensures that only authorized parties can access the certificate, providing an added layer of security to the entire process. After the certificate is uploaded and verified, the system sends it for verification. This process involves various checks, including cross-referencing with official records and communicating with relevant authorities. Educational institutions that issue the certificate are considered stakeholders in the system, as they are responsible for verifying the authenticity of the certificate. Once verified, the user is informed of the outcome of the verification process. This step ensures that the certificates stored on the system are genuine, reliable, and trusted.

If the certificate is approved, the user can choose to encrypt it and store it in IPFS (InterPlanetary File System). The IPFS then returns a CID (Content Identifier) that is stored in the blockchain alongside other necessary information such as the time of verification, who verified it, and the type of certificate. When a company or other authorized party requests a copy of the certificate, the user is notified and asked for permission to share the certificate. If permission is granted, the system retrieves the encrypted certificate from IPFS, decrypts it, and temporarily stores it on the system's servers for the company to view. This step ensures that the privacy of the certificate is maintained while also enabling authorized parties to view it securely.

### 3.13 System Stakeholders

The stakeholders involved in a system for certificate verification are essential for the development and implementation of the technology. Educational institutions, professional organizations, and employers play a critical role in issuing and verifying certificates, which are key indicators of an individual's knowledge, skills, and competencies. The stakeholders involved in this system are as follows:

- **Applicant:** The individual who is applying has the capability to upload their pertinent qualifications to the system. Once the qualifications are verified, they are encrypted and kept securely using IPFS. In addition, applicants can grant permission for companies to request access to their validated certificates.
- **Company:** Organizations have the capacity to search for an applicant and ask for permission to view their qualifications so that they can access their credentials.
- **Admin:** The administrator gets verification requests from applicants to validate the genuineness of their certificates. The administrator forwards these verification requests to the appropriate educational establishments to confirm the accuracy of the certificates. After the verification process is finished, the encrypted certificates are uploaded to IPFS and the metadata of the database is stored on the blockchain.
- **Institution:** The institution's main responsibility is to verify the legitimacy of the certificates

# Chapter 4

## Design and Development

### 4.1 Project Management and Financial Activities

By effectively managing resources, stakeholders, and timelines, project managers can oversee the development, deployment, and maintenance of the system. Simultaneously, financial activities such as budgeting, fundraising, and cost management are vital to sustain the project's long-term viability. In this article, we delve into the intersection of project management and financial activities within the context of a blockchain-based certificate verification system, exploring their importance, challenges, and potential solutions to ensure the success of such initiatives.

#### 4.1.1 Work Breakdown Structure

The WBS allows project managers to effectively plan, organize, and control the project by identifying the necessary activities, dependencies, and resources required at each level of the structure. It provides clarity and structure to the project team, ensuring that all project aspects are accounted for and properly executed.

Figure 4.1: Work Breakdown Structure Flowchart



Figure 4.1 shows the WBS of our system. It plays a crucial role in ensuring project success by providing a roadmap for project execution and enabling effective coordination of activities.

### 4.1.2 Estimate Costs

To estimate the costs associated with each work package in the WBS, we need to consider both direct costs and indirect costs. Here is an example of estimated costs for each work package, taking into account various cost factors:

#### 1. Project Initiation

- Define project scope and objectives: No direct costs.
- Identify key stakeholders: No direct costs.
- Conduct initial feasibility study: Cost of market research reports or consulting fees.
- Prepare project charter: No direct costs.

#### 2. Requirements Gathering

- Identify user requirements: No direct costs.
- Document functional and non-functional requirements: No direct costs.
- Conduct market research on existing blockchain solutions: Cost of market research reports or consulting fees.
- Define system architecture and technical specifications: No direct costs.

#### 3. Development Phase

- Set up blockchain infrastructure: Cost of hardware, software licenses, and cloud services.
- Develop smart contract for certificate verification: Development resources' salaries or consulting fees.
- Design user interface for the verification system: Cost of design tools and UI/UX resources.
- Implement data encryption and security measures: Cost of encryption software and security consultants.
- Integrate the system with existing databases and systems: Integration resources' salaries or consulting fees.

#### 4. Testing and Quality Assurance

- Develop test cases and test scenarios: Testing resources' salaries or consulting fees.

- Conduct unit testing of smart contracts and system components: Testing resources' salaries or consulting fees.
- Perform system integration testing: Testing resources' salaries or consulting fees.
- Conduct user acceptance testing: Testing resources' salaries or consulting fees.
- Resolve issues and bugs identified during testing: Development resources' salaries or consulting fees.

## 5. Deployment and Training

- Prepare deployment plan and schedule: No direct costs.
- Install and configure the verification system on production environment: Deployment resources' salaries or consulting fees.
- Conduct end-user training and provide user documentation: Training resources' salaries or consulting fees, cost of training materials.
- Transition the system to operational support: No direct costs.

## 6. Project Management and Governance

- Project planning and scheduling: No direct costs.
- Risk management and mitigation: No direct costs.
- Stakeholder management and communication: No direct costs.
- Project progress monitoring and reporting: Cost of project management software or tools.

## 7. Maintenance and Upgrades

- Provide ongoing technical support and troubleshooting: Support resources' salaries or consulting fees.
- Perform regular system updates and maintenance tasks: Maintenance resources' salaries or consulting fees.
- Incorporate user feedback and implement system enhancements: Development resources' salaries or consulting fees.
- Plan for future upgrades and scalability: No direct costs.

### 4.1.3 Budget Plan

It is necessary to create a project budget that accounts for all expenses and materials needed to execute the project, as well as any contingency funds. It is important to keep an eye on the budget to make sure it stays on track and within the project's parameters.

Table 4.1: Cost Breakdown

| Cost Type         | Description                             | Cost (in TK.)   |
|-------------------|---|-----------------|
| Hardware Cost     | 1 Router                                | 1,400           |
|                   | 3 Computers                             | 2,60,000        |
|                   | Computer 1                              | 1,20,000        |
|                   | Computer 2                              | 90,000          |
|                   | Computer 3                              | 70,000          |
| Software Cost     | Working tools                           | 0               |
| Developer Cost    | Blockchain Developer (2)                | 2,88,000        |
|                   | Per Month                               | 45,000          |
|                   | Development Period                      | 3.2 Months      |
|                   | System Developer (1)                    | 96,000          |
|                   | Per Month                               | 30,000          |
|                   | Development Period                      | 3.2 Months      |
| Maintenance Cost  | Monthly Maintenance Cost                | 3,000           |
|                   | Monthly Internet Cost (4 months)        | 3,600           |
| Other Cost        | Operational Cost, Furniture, Rent, etc. | 80,000          |
| Field Testing     |   | 40,000          |
| Logistic Supports | Stationary<br>Research Associate        | 150,000         |
| Publication       | Scholarly Articles                      | 100,000         |
|                   | Patent                                  |                 |
| <b>Total</b>      |   | <b>9,84,000</b> |

Table 5.1 shows the estimated cost based on our analysis. The cost of making the prototype may vary depending on other factors.

### 4.1.4 Funding Plan:

Financing will be required to cover the platform's initial development costs as well as infrastructure and marketing costs. The funds will be raised through a combination of venture capital, angel investing, and crowdfunding.

#### 4.1.4.1 Crowdfunding:

Another possibility for financing a SAAS company is through crowdsourcing. On a number of websites, such as Kickstarter and Indiegogo, businesspeople may share their concepts with the public and raise money through pre-orders or donations. To launch a



successful crowdfunding campaign, the team would need to create an interesting video and presentation, set realistic financial goals, and provide enticing rewards to backers.

#### **4.1.4.2 Government Grants:**

Depending on the country where the firm is situated, there may be government grants or funding options available to assist technology enterprises. For instance, Singapore's government offers a range of incentives and initiatives to help technological companies. To qualify for government assistance, the team would need to meet specific criteria, such as having a strong business plan and a distinct path to revenue.

### 4.1.5 Financial Report

The financial report provides a comprehensive overview of the project's financial status, including actual expenses, budget variances, cash flow statements, and other relevant financial metrics.

Table 4.2: Cash Flow Statement

| <b>Cash Inflows</b>        | <b>Amount (in TK.)</b> |                |
|----------------------------|------------------------|----------------|
| Funding Sources            | 500,000                |                |
| <b>Total Cash Inflows</b>  | 500,000                |                |
| <b>Cash Outflows</b>       | <b>Year 1</b>          | <b>Year 2</b>  |
| Hardware Cost              | 40,000                 | -              |
| Software Cost              | -                      | -              |
| Developer Cost             | 200,000                | 200,000        |
| Maintenance Cost           | 10,000                 | 10,000         |
| Other Cost                 | 20,000                 | 20,000         |
| Field Testing              | 30,000                 | -              |
| Logistic Supports          | 15,000                 | 15,000         |
| Publication                | -                      | 10,000         |
| <b>Total Cash Outflows</b> | <b>315,000</b>         | <b>255,000</b> |
| <b>Net Cash Flow</b>       | <b>185,000</b>         | <b>245,000</b> |
| <b>Opening Balance</b>     | <b>0</b>               | <b>185,000</b> |
| <b>Closing Balance</b>     | <b>185,000</b>         | <b>430,000</b> |

Table 4.2 shows the cash flow statement to build the system which track the inflow and outflow of cash throughout the project's duration.

## 4.2 Introduction

We created a solid architecture through careful planning and collaboration that makes use of blockchain technology to guarantee the immutability, transparency, and integrity of certificate records. To automate the verification process and enable quick and trustworthy certificate verification, smart contracts were implemented during the development phase. An intuitive and user-friendly interface was produced as a result of our user-centric design methodology, which gave simplicity and accessibility priority. The system underwent thorough testing and optimization to guarantee its scalability, effectiveness, and reliability. The basis for an original and reliable solution that will change the verification of academic credentials was successfully laid during the design and development phase.

## 4.2.1 Front-end

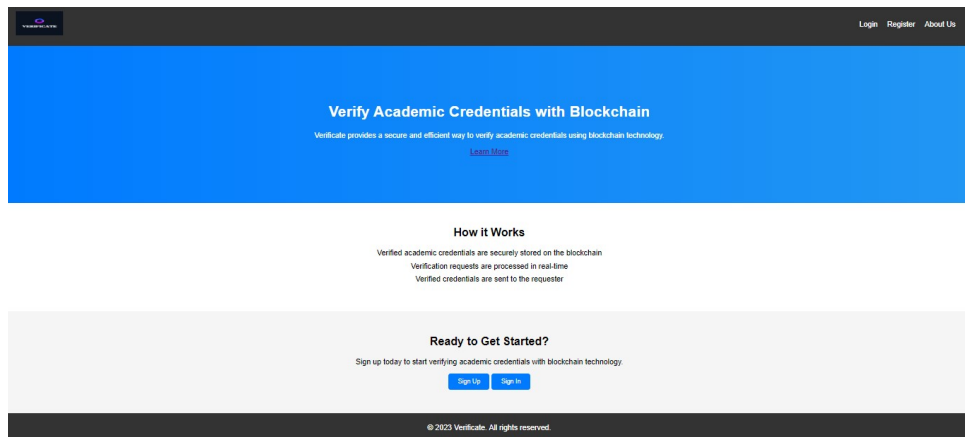


Figure 4.2: Homepage

*Figure 4.1 demonstrates the homepage for the system. This page gives a brief description to the viewer into what the system has to offer. Furthermore, it redirects the user to the sign in and sign up page.*

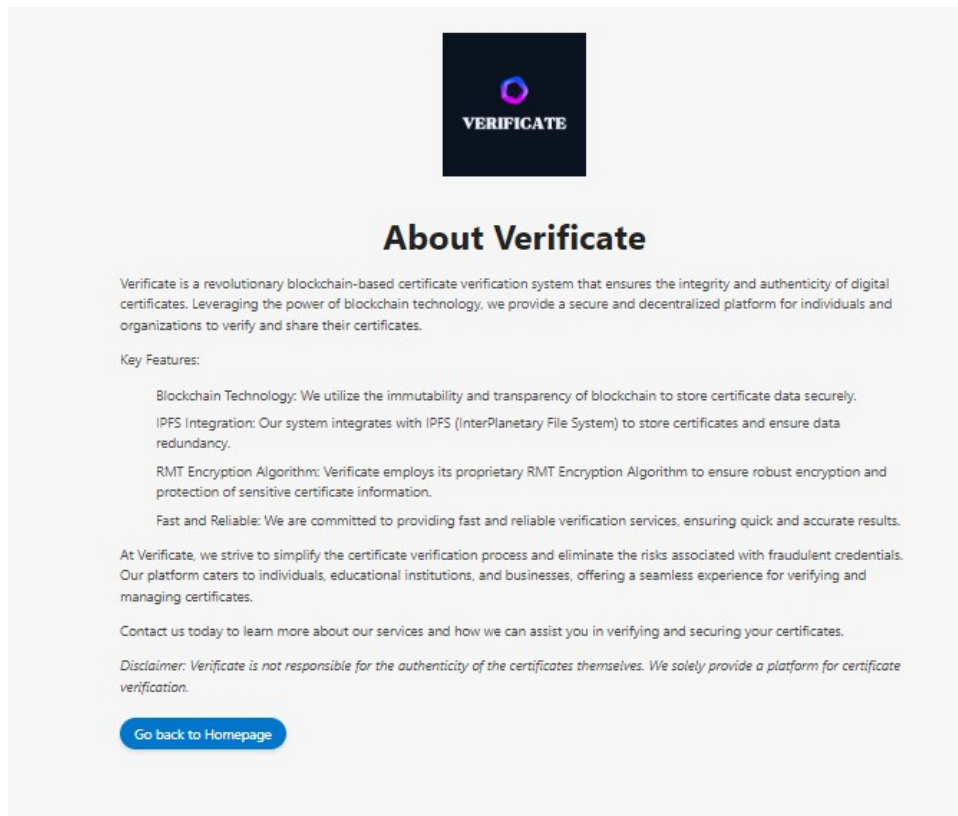
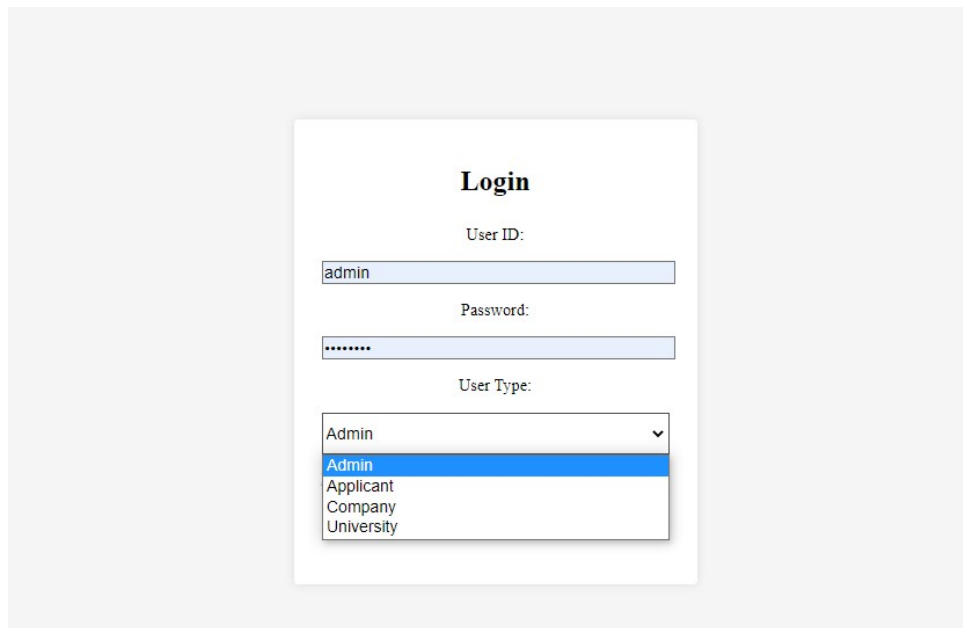


Figure 4.3: About Us

*Figure 4.2 goes into details explaining the main functionalities of the system. It also highlights the key features of the system in pair with the blockchain and IPFS technology. From this page, the user can choose to go back to the homepage.*

The image shows a login form titled "Login" centered on a light gray background. The form has three input fields: "User ID:" with the text "admin" entered, "Password:" with masked characters "\*\*\*\*\*", and "User Type:" which is a dropdown menu. The dropdown menu is open, showing four options: "Admin" (highlighted in blue), "Applicant", "Company", and "University".

**Login**

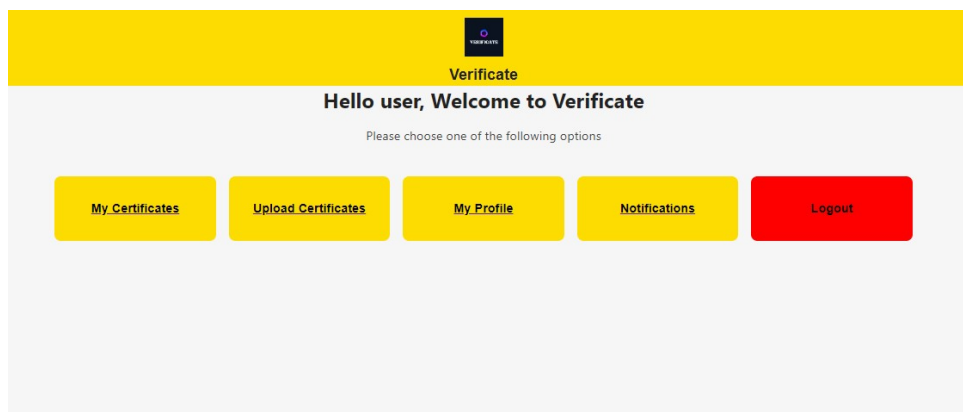
User ID:  
admin

Password:  
\*\*\*\*\*

User Type:  
Admin  
Applicant  
Company  
University

Figure 4.4: Login Page

Figure 4.3 shows the login page. The page provides options for different stakeholders to login from the same interface. it only requires choosing the mode of stakeholder along with the username and password.

The image shows the applicant interface. It has a yellow header bar with a logo and the word "Verify". Below the header, it says "Hello user, Welcome to Verify". Underneath, it says "Please choose one of the following options". There are five buttons: "My Certificates", "Upload Certificates", "My Profile", "Notifications", and "Logout". The "Logout" button is red, while the others are yellow.

**Verify**

Hello user, Welcome to Verify

Please choose one of the following options

My Certificates Upload Certificates My Profile Notifications Logout

Figure 4.5: Applicant Interface

Figure 4.4 shows the applicant page. The page provides options for the applicant to upload their certificates for verification, view their certificate, view their profile, receive notifications and logout.

[My Certificates](#) [Upload Certificates](#) [My Profile](#) [Notifications](#) [Logout](#)

## Certificate Upload Form

Full Name:

Certificate Name:

Certificate Type:

Transcript

Institute:

British Council

Possible Verifier:

Certificate File:

Choose File

No file chosen

Submit

Figure 4.6: Applicant Certificate Upload

Figure 4.5 shows the applicant certificate upload page. The page provides options for the applicant to upload their certificates for verification.

| <div> <a href="#">My Certificates</a> <a href="#">Upload Certificates</a> <a href="#">My Profile</a> <a href="#">Notifications</a> <a href="#">Logout</a> </div> |                  |                                       |                 |  |           |                                 |
|--|------------------|---------------------------------------|-----------------|--|-----------|---------------------------------|
| Certificate Information  |                  |                                       |                 |  |           |                                 |
| No   | Certificate Name | Institute                             | Verified Status | Verified Date Time   | Verifier  | Content Identifier (IPFS Hash)  |
| 1  | Transcript       | University of Liberal Arts Bangladesh | Verified        | Mon May 15 2023 00:00:00 GMT+0600 (Bangladesh Standard Time) | Registrar | athocdepe8t54fct5etlnvveoldfice |

Figure 4.7: Applicant View Certificate

Figure 4.6 shows the applicant certificate view page. The page provides options for the applicant to view their uploaded certificates.

| My Profile       |                                       |
|------------------|---------------------------------------|
| First Name:      | Arunangshu                            |
| Last Name:       | Mojumder                              |
| User-ID:         | ratul12316                            |
| Email:           | ratul12316@gmail.com                  |
| University Name: | University of Liberal Arts Bangladesh |
| Phone Number:    | 1400447008                            |

Figure 4.8: Applicant Profile

Figure 4.7 shows the applicant certificate view page. The page provides options for the applicant to view their uploaded certificates.

Verify

Hello ULAB, welcome to Verify

Please choose one of the following options

View Requests

Accepted Requests

Notifications

Logout

Figure 4.9: University Interface

Figure 4.8 shows the login page. The page provides options for different stakeholders to login from the same interface. it only requires choosing the mode of stakeholder along with the username and password.

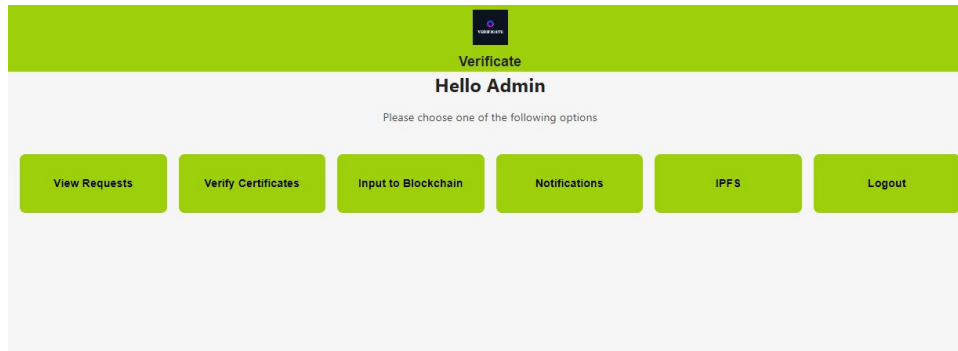


Figure 4.10: Admin Interface

Figure 4.9 shows the admin page. The page provides options for the admin to view the verification request, redirect the request to university, receive notification, push the details of the verification to IPFS and Blockchain.

| View Requests   Verify Certificates   Notifications   Push to Blockchain   IPFS   Cryptography   Logout |                            |                      |                  |                 |                   |                         |            |          |
|---|----------------------------|----------------------|------------------|-----------------|-------------------|-------------------------|------------|----------|
| Requested Certificates  |                            |                      |                  |                 |                   |                         |            |          |
| #   | Full Name                  | Certificate Name     | Certificate Type | Institute       | Possible Verifier | Certificate File        | User       | Action   |
| 1   | Arunangshu Mojumder Raatul | Probational          | Probational      | University      | Registrar         | Application-PDF.pdf     | ratul12316 | Verified |
| 2   | Arunangshu Mojumder Raatul | Transcript           | Transcript       | University      | Registrar         | Application-PDF (2).pdf | ratul12316 | Verified |
| 3   | Arunangshu Mojumder Raatul | A-Levels Certificate | A-Levels         | British Council | British Council   | CSE 417 Lecture CFG.pdf | ratul12316 | Verified |
| 4   | Arunangshu Mojumder Raatul | Studentship          | Studentship      | University      | Registrar         | 79030.pdf               | ratul12316 | Verified |
| 5   | Arunangshu Mojumder Raatul | MOI                  | MOI              | University      | Registrar         | Maftun.pdf              | ratul12316 | Verified |

Figure 4.11: Requested Certificates

Figure 4.10 shows the requested certificates page for the admin. The page has a table that shows the information about the user, certificate name, certificate type, institute name, verifier information, file, and the action of verification.



| View Requests   Verify Certificates   Notifications   Push to Blockchain   IPFS   Cryptography   Logout |                            |                      |                  |                 |                   |                         |            |                           |
|---|----------------------------|----------------------|------------------|-----------------|-------------------|-------------------------|------------|---------------------------|
| Requested Certificates  |                            |                      |                  |                 |                   |                         |            |                           |
| #   | Full Name                  | Certificate Name     | Certificate Type | Institute       | Possible Verifier | Certificate File        | User       | Action                    |
| 1   | Arunangshu Mojumder Raatul | Probational          | Probational      | University      | Registrar         | Application-PDF.pdf     | ratul12316 | <button>Verified</button> |
| 2   | Arunangshu Mojumder Raatul | Transcript           | Transcript       | University      | Registrar         | Application-PDF (2).pdf | ratul12316 | <button>Verified</button> |
| 3   | Arunangshu Mojumder Raatul | A-Levels Certificate | A-Levels         | British Council | British Council   | CSE 417 Lecture CFG.pdf | ratul12316 | <button>Verified</button> |
| 4   | Arunangshu Mojumder Raatul | Studentship          | Studentship      | University      | Registrar         | 79030.pdf               | ratul12316 | <button>Verified</button> |
| 5   | Arunangshu Mojumder Raatul | MOI                  | MOI              | University      | Registrar         | Maftun.pdf              | ratul12316 | <button>Verified</button> |

Figure 4.12: Requested Certificates

Figure 4.11 shows the requested certificates page for the admin. The page has a table that shows the information about the user, certificate name, certificate type, institute name, verifier information, file, and the action of verification.

|               |                     |               |                    |      |        |
|---------------|---------------------|---------------|--------------------|------|--------|
| View Requests | Verify Certificates | Notifications | Push to Blockchain | IPFS | Logout |
|---------------|---------------------|---------------|--------------------|------|--------|

## Send Certificate for Verification

Certificate File:  No file chosen

Recipient Email:

Message:

Figure 4.13: Send Verification Request

Figure 4.12 shows the send verification request page for the admin. The page has an option to upload certificate file, select the recipient email and send additional message.

The screenshot shows a web interface with a green navigation bar at the top containing links: 'View Requests', 'Verify Certificates', 'Notifications', 'Push to Blockchain', 'IPFS', and 'Logout'. Below the navigation bar, the main heading is 'Certificate Verifier' in green. The central part of the page features a white form with the following fields: 'Username:', 'Content Identifier:', 'Verifier:', 'Verification Time:', and 'Certificate Name:'. Each field has a corresponding light gray input box. At the bottom of the form is a green button labeled 'Verify Certificate'.

Figure 4.14: Push To Blockchain

Figure 4.13 shows the push to blockchain page for the admin. The page has an option to upload certain information to the blockchain such as username, content identifier from IPFS, verifier information, verification time, certificate name.

The screenshot shows a web interface with a blue header bar. On the right side of the header is a black button with a white 'V' icon and the text 'Verify'. Below the header, the text 'Hello Company, welcome to Verify' is displayed. Underneath this is a smaller text: 'Please choose one of the following options'. At the bottom, there are four blue buttons arranged horizontally: 'Search for User', 'Search for Certificate', 'Profile', and 'Logout'.

Figure 4.15: Company Page

Figure 4.14 shows the company page. The page provides options for the company to search for the particular applicant, search for the certificate, view their profile and logout.

[Search User](#)
[Search Certificate](#)
[Company Profile](#)
[Logout](#)

### User Search

User ID:

Search

| First Name | Last Name | Certificate Name | Institute                             | Verified Status | Date and Time  | Content Identifier                          | Verifier  | Actions                          |
|------------|-----------|------------------|---------------------------------------|-----------------|--|---|-----------|----------------------------------|
| Arunangshu | Mojumder  | Transcript       | University of Liberal Arts Bangladesh | Verified        | Mon May 15 2023 00:00:00 GMT+0600 (Bangladesh Standard Time) | Qmaqs5Ka8GVpdeNnDmMCDvc9QPyPzTMqv2a1Mn3XEiJ | Registrar | <a href="#">Show Certificate</a> |

#### Certificate Information

|                     |  |
|---------------------|--|
| Username:           | ratul12316   |
| Fullname:           | Arunangshu Mojumder Raatul                                   |
| Content Identifier: | Qmaqs5Ka8GVpdeNnDmMCDvc9QPyPzTMqv2a1Mn3XEiJ                  |
| Verifier:           | British Council  |
| Verification Time:  | Thu May 25 2023 20:31:53 GMT+0600 (Bangladesh Standard Time) |
| Certificate Name:   | A-Levels Certificate   |
| Certificate Type:   | Statement of Entry   |
| Institute Name:     | British Council  |
| Verifier Contact:   | britishcouncil@edu.bd  |

Figure 4.16: Company Interface

Figure 4.15 shows the company user search page. The page provides options for the company to search for the particular applicant.

[Search User](#)
[Search Certificate](#)
[Company Profile](#)
[Logout](#)

## About Company

|                       |                     |
|-----------------------|---------------------|
| Company Name:         | MacroSoft           |
| User-ID:              | macrosoft           |
| Email:                | macrosoft@ms.com    |
| Password:             | macrosoft123        |
| Personal Email:       | billdoors@ms.com    |
| Registered Personnel: | Bill Doors          |
| Designation:          | Recruitment Manager |
| Company ID:           | 123789456           |

Figure 4.17: Company Profile

Figure 4.16 shows the company profile page. The page provides options for the company to view the details of their profile and update information if necessary.

## 4.2.2 Back-end

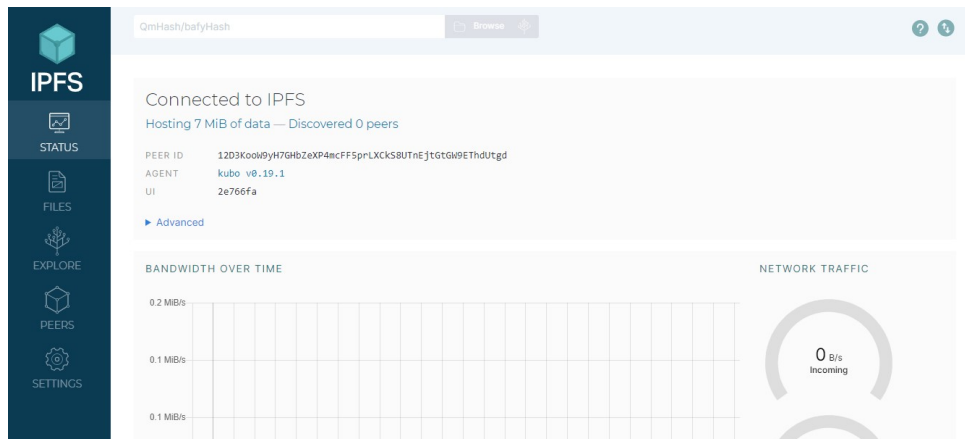


Figure 4.18: IPFS

Figure 4.17 shows the connection of IPFS with the system. The IPFS portal provides the option store the encrypted certificates into the cloud.

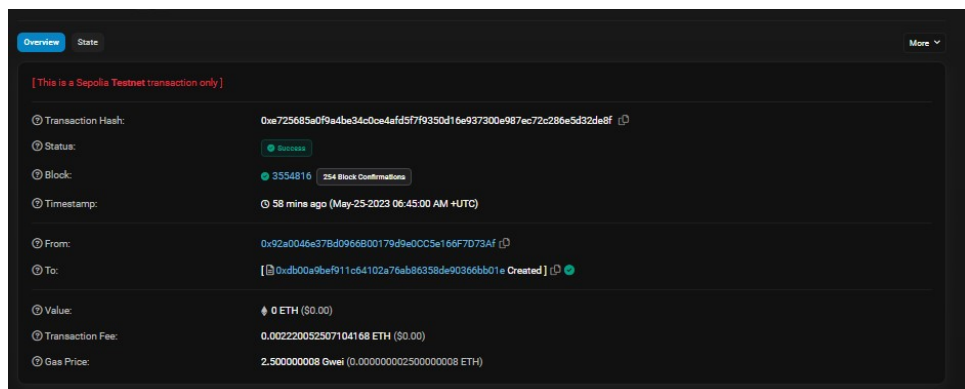



Figure 4.19: Sepolia Testnet

Figure 4.18 shows the connection with Sepolia testnet. It allow developers to experiment, test, and deploy smart contracts.

| Table   | Action                                      | Rows | Type   | Collation          | Size      | Overhead |
|---|---|------|--------|--------------------|-----------|----------|
| <input type="checkbox"/> admin                | ★ Browse Structure Search Insert Empty Drop | 1    | InnoDB | utf8mb4_general_ci | 16.0 KiB  | -        |
| <input type="checkbox"/> applicant            | ★ Browse Structure Search Insert Empty Drop | 1    | InnoDB | utf8mb4_general_ci | 16.0 KiB  | -        |
| <input type="checkbox"/> applicant_request    | ★ Browse Structure Search Insert Empty Drop | 0    | InnoDB | utf8mb4_general_ci | 16.0 KiB  | -        |
| <input type="checkbox"/> certificate_request  | ★ Browse Structure Search Insert Empty Drop | 5    | InnoDB | utf8mb4_general_ci | 48.0 KiB  | -        |
| <input type="checkbox"/> company              | ★ Browse Structure Search Insert Empty Drop | 1    | InnoDB | utf8mb4_general_ci | 16.0 KiB  | -        |
| <input type="checkbox"/> company_request      | ★ Browse Structure Search Insert Empty Drop | 0    | InnoDB | utf8mb4_general_ci | 16.0 KiB  | -        |
| <input type="checkbox"/> encryption           | ★ Browse Structure Search Insert Empty Drop | 0    | InnoDB | utf8mb4_general_ci | 16.0 KiB  | -        |
| <input type="checkbox"/> notifications        | ★ Browse Structure Search Insert Empty Drop | 12   | InnoDB | utf8mb4_general_ci | 16.0 KiB  | -        |
| <input type="checkbox"/> university           | ★ Browse Structure Search Insert Empty Drop | 1    | InnoDB | utf8mb4_general_ci | 16.0 KiB  | -        |
| <input type="checkbox"/> university_request   | ★ Browse Structure Search Insert Empty Drop | 1    | InnoDB | utf8mb4_general_ci | 48.0 KiB  | -        |
| <input type="checkbox"/> verified_certificate | ★ Browse Structure Search Insert Empty Drop | 1    | InnoDB | utf8mb4_general_ci | 64.0 KiB  | -        |
| <input type="checkbox"/> verifiers            | ★ Browse Structure Search Insert Empty Drop | 1    | InnoDB | utf8mb4_general_ci | 16.0 KiB  | -        |
| 12 tables                                     | Sum   | 24   | InnoDB | utf8mb4_general_ci | 304.0 KiB | 0 B      |

Figure 4.20: Database

Figure 4.19 shows the web based database connection using MySQL. It stores additional information about the stakeholders.


EthValidate
Transaction
Address
Token

TxHash

sepolia (infura)

TxHash: [0xe725685a0f9a4be34c0ce4afd5f7f9350d16e937300e987ec72c286e5d32de8f](#)

Status: Success

Block Height: [3554816](#) (273 block confirmations)

From: [0x92a0046e37bd0966b00179d9e0cc5e166f7d73af](#)

To: [null](#)

Value: 0 Ether

Gas Limit: 888021

Gas Used: 888021

Gas Price: 0.000000002500000008 Ether (2.500000008 Gwei)

Tx Fee: 0.002220052507104168 Ether

Figure 4.21: Transaction Details

Figure 4.20 shows the transaction details. It allows the user to see Transaction Hash, Block Number, Gas Used, Gas Price, Transaction Status, Timestamp.

### 4.2.3 Smart-Contract

The presented smart contract, named "CertificateVerifier," serves as a solution for verifying and storing certificates in a decentralized manner. It leverages the Solidity programming language and the Ethereum blockchain platform. The contract defines a Certificate structure that holds essential fields, including username, full name, content identifier, verifier, verification time, certificate name, certificate type, institute name, and verifier contact.

---

**Algorithm 1:** CertificateVerifier Smart Contract

---

**Data:** Certificate Structure

**Result:** Verified and Stored Certificates

**Input :** Certificate Information

**Output:** Verified Certificate

**1 Function**

2 | `verifyCertificate(Certificate Information)`

3 **end**

4 Create a new Certificate object with the provided details;

5 Store the Certificate object in the "certificates" mapping using the content identifier as the key;

**6 Function**

7 | `getCertificate(Content Identifier)`

8 **end**

9 Retrieve the Certificate object from the "certificates" mapping using the content identifier;

10 **return** the details of the Certificate object;

---

The smart contract's pseudocode is displayed in Algorithm 1. The contract provides two essential services. Users can validate a certificate using the first function, "verifyCertificate," by giving the necessary data as inputs. Using the supplied information to generate a new Certificate object, it stores it in a private mapping called "certificates," using the content identification as a special key. Using the content identifier and the second function, "getCertificate," it is easier to retrieve the certificate's data. Users have access to details including their username, complete name, content identification, verifier, verification time, certificate name, kind, institute name, and verifier contact by contacting this function.

A useful solution for certificate verification and storage on the blockchain is offered by the CertificateVerifier contract. It guarantees the immutability and transparency of the data that is saved, making it a desirable alternative for businesses, educational institutions, or any other body needing a tamper-proof certificate verification system. Due to the lack of a centralized authority, the use of blockchain technology improves trust and security. The contract is a user-friendly method for checking and getting certificate information on the Ethereum network due to its functionality and ease of use.

#### 4.2.4 Algorithm

One of the most important factors to take into account when it comes to certificate verification is the security element. The system may greatly improve the security of the certificate data by creating a unique encryption and decryption technique. The system can prevent unwanted access by encrypting the certificate data before putting it in IPFS. Without the decryption key, which is held by only authorized users, even if a hacker manages to access the stored data, they will not be able to read it.

---

**Algorithm 2:** Encryption Algorithm

---

**Input** :None

**Output**:None

```
1 encrypt()
2 fileInput ← document.getElementById("fileInput");
3 encryptionKey ← document.getElementById("encryptionKey").value;
4 file ← fileInput.files[0];
5 timestamp ← file.lastModifiedDate.getTime();
6 reader ← new FileReader();
7 reader.readAsArrayBuffer(file);
8 reader.onload = async function() arrayBuffer ← reader.result;
9 byteArray ← new Uint8Array(arrayBuffer);
10 keyArray ← new Uint8Array(encryptionKey.length);
11 for i ← 0 to encryptionKey.length do
12   | keyArray[i] ← encryptionKey.charCodeAt(i);
13 end
14 for i ← 0 to byteArray.length do
15   | byteArray[i] ← byteArray[i] ^ keyArray[i % keyArray.length];
16 end
17 filename ← "encrypted_" + file.name;
18 blob ← new Blob([byteArray], type: file.type);
19 blob.lastModified ← timestamp;
20 folderName ← "Encrypted_Files";
21 url ← URL.createObjectURL(blob);
22 downloadLink ← document.getElementById("downloadLink");
23 downloadLink.href ← url;
24 downloadLink.download ← folderName + "/" + filename;
25 downloadLink.innerHTML ← "Download Encrypted File";
26 encryptedFile ← new File([blob], filename, type: file.type);
27 const ipfs = IpfsHttpClient.create( host: "localhost", port: "5001", protocol:
    "http");
28 const addedFile ← await ipfs.add(encryptedFile);
29 const contentIdentifier ← addedFile.cid.toString();
30 console.log("Content Identifier:", contentIdentifier);
31 ;
32 encrypt();
```

---

The file input element and encryption key are first retrieved by the "encrypt()"

method from the HTML page. After reading the file's contents, it turns them into an array of bytes. Additionally, the encryption key is transformed into an array of bytes. The code then uses the relevant byte from the encryption key to conduct the XOR operation on each byte of the file's contents. As a result, the encrypted data is generated and then transformed back into a blob object. The user may now download the encrypted file by updating the download URL. The method takes input values from the HTML content, processes the file and encryption key, encrypts the byte arrays, generates a blob for download, sets the metadata.

---

**Algorithm 3:** Decryption Algorithm

---

**Input** :None

**Output**:None

```

1 decrypt()
2 fileInput ← document.getElementById("fileInput");
3 encryptionKey ← document.getElementById("encryptionKey").value;
4 file ← fileInput.files[0];
5 reader ← new FileReader();
6 reader.readAsArrayBuffer(file);
7 reader.onload = async function() arrayBuffer ← reader.result;
8 byteArray ← new Uint8Array(arrayBuffer);
9 keyArray ← new Uint8Array(encryptionKey.length);
10 for i ← 0 to encryptionKey.length do
11 | keyArray[i] ← encryptionKey.charCodeAt(i);
12 end
13 for i ← 0 to byteArray.length do
14 | byteArray[i] ← byteArray[i] ^ keyArray[i % keyArray.length];
15 end
16 filename ← "decrypted_" + file.name;
17 blob ← new Blob([byteArray ], type: file.type );
18 timestamp ← blob.lastModified;
19 folderName ← "Decrypted_Files";
20 url ← URL.createObjectURL(blob);
21 downloadLink ← document.getElementById("downloadLink");
22 downloadLink.href ← url;
23 downloadLink.download ← folderName + "/" + filename;
24 downloadLink.innerHTML ← "Download Decrypted File";
25 ;
26 decrypt();

```

---

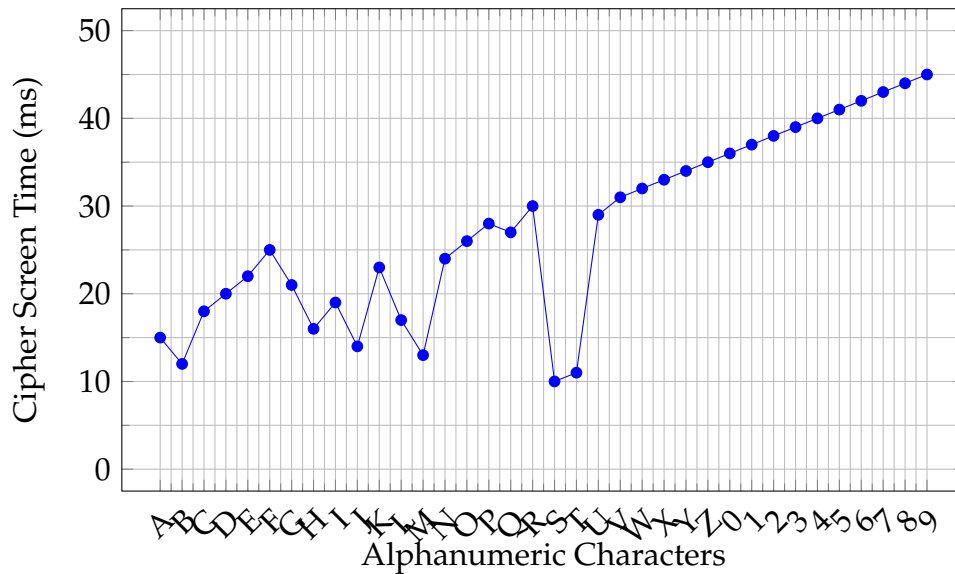
The "decrypt()" function, which carries out the "encrypt()" function's opposite operation, is illustrated in Algorithm 3. The encrypted file is retrieved, its contents are read, and it is then converted to an array of bytes. The original file contents are then generated by XORing the encrypted data with the encryption key bytes. The user may now access the decrypted file by using the updated download link once the original file contents are changed back to a blob object.



### 4.2.5 Encryption Algorithm Comparative Analysis

This analysis compares and contrasts the performance of two encryption algorithms, one of which is used in our system and the other of which is detailed in a research article. While the technique from the research article is based on the Hill cipher scheme for lightweight Internet of Things (AIoT) enabled encrypted devices in wireless sensor networks, the encryption algorithm in our system concentrates on the "texttt"encrypt() function [Sohail et al. \(2020\)](#).

#### 4.2.5.1 Graph 1: Our System



The encryption algorithm used in our system is shown by the first graph. It demonstrates the connection between file size and encryption time, illuminating the algorithm's performance in terms of effectiveness and scalability. The task of utilizing an encryption key to encrypt the file falls to the `textttencrypt()` method. The file's contents are read by the algorithm and transformed into an array of bytes. The encryption key is simultaneously turned into an array of bytes.

#### 4.2.5.2 Graph 2: Research Paper (Hill Cipher Scheme)

The performance of the Hill cipher scheme, as detailed in the study article, is shown in the second graph. The suitable key is used to encrypt the message in this system, and the inverse key is used to decode it. The suggested encryption system is presented as an upgrade to the Hill cipher scheme, which is shown to be vulnerable to known plain-text assaults. The suggested method comprises steps such as selecting a  $3 \times 3$  matrix, multiplying the message with the key, taking mod 128, converting the cipher-text into binary form, shifting row and column positions, and performing XOR operations with a random number for lightweight encrypted devices.

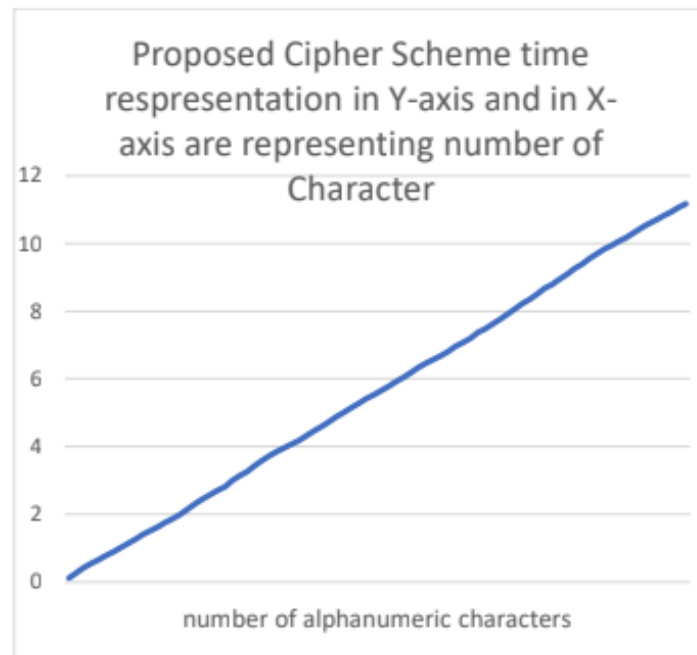


Figure 4.22: Graph from Research Paper

#### 4.2.5.3 Comparative Analysis

##### 1. Efficiency and Security:

- The performance of our system's encryption technique is displayed in Graph 1, which emphasizes the connection between file size and encryption time. It offers information about the effectiveness and scalability of the algorithm.
- The suggested cipher system is introduced in Graph 2 as a remedy for the Hill cipher scheme's flaws. It handles the well-known plain-text attack and includes extra security-enhancing steps.

##### 2. Randomness and Error-Free Encryption:

- The study paper's suggested cipher scheme asserts to provide more randomness to the cipher-text than to the plain-text. For greater encryption, this is essential.
- The encryption technique used by our system focuses on error-free encryption and decryption operations. It makes sure that the encrypted information can be reliably decoded without being lost or corrupted.

### 4.3 Complexity of Our Algorithm

The encrypt function in the provided code performs file encryption and includes various steps. Firstly, it retrieves file information such as the file input and encryption

key, which has a constant time complexity of  $O(1)$ . The next step involves reading the file using a `FileReader` and converting it into an array buffer, with a time complexity proportional to the size of the file ( $O(\text{file size})$ ). The encryption process then converts the array buffer and encryption key into `Uint8Array` objects, both with time complexities of  $O(\text{file size})$  and  $O(\text{key length})$ , respectively. The actual encryption is performed by applying a bitwise XOR operation on each byte of the file, again with a time complexity proportional to the file size ( $O(\text{file size})$ ).

Subsequently, the function creates an encrypted file by generating a filename ( $O(1)$ ) and creating a blob object from the encrypted data ( $O(\text{file size})$ ). Setting up the download link and displaying the content identifier involves manipulating DOM elements, which has a constant time complexity ( $O(1)$ ). The function then proceeds to upload the encrypted file to IPFS. This step includes setting up an IPFS client ( $O(1)$ ) and adding the encrypted file to IPFS, which again has a time complexity proportional to the file size ( $O(\text{file size})$ ).

Lastly, the function copies the content identifier to the clipboard, which involves accessing DOM elements and creating a button ( $O(1)$ ), and then copying the content identifier itself, with a constant time complexity ( $O(1)$ ).

In terms of time complexity analysis, the best case scenario assumes constant file size and encryption key length, along with fast IPFS upload, resulting in an overall time complexity of  $O(1)$ . On the other hand, the worst case scenario considers a large file size and significant encryption key length, leading to a time complexity of  $O(\text{file size} + \text{key length})$ . It is important to note that the time complexity of asynchronous operations, such as file reading and IPFS uploading, can be affected by implementation details and external factors like network speed, thus the provided analysis assumes ideal scenarios for simplicity.

The decrypt function in the provided code is responsible for decrypting a file and involves several steps. Firstly, it retrieves the necessary file information by accessing the `fileInput` and `encryptionKey` elements, both of which have a constant time complexity of  $O(1)$ . The next step involves reading the file using a `FileReader` and converting it into an array buffer, with a time complexity proportional to the size of the file ( $O(\text{file size})$ ). The decryption process then converts the array buffer and encryption key into `Uint8Array` objects, both with time complexities of  $O(\text{file size})$  and  $O(\text{key length})$ , respectively. The actual decryption is achieved by performing a bitwise XOR operation on each byte of the file, again with a time complexity proportional to the file size ( $O(\text{file size})$ ).

Following the decryption, the function proceeds to create a decrypted file. This step involves generating a filename ( $O(1)$ ) and creating a blob object from the decrypted data ( $O(\text{file size})$ ). Setting up the download link and displaying the content identifier involves manipulating DOM elements, which has a constant time complexity ( $O(1)$ ).

Next, the function uploads the decrypted file to IPFS. This step includes setting up an IPFS client ( $O(1)$ ) and adding the decrypted file to IPFS, with a time complexity proportional to the file size ( $O(\text{file size})$ ). Finally, the function copies the content identifier to the clipboard. This involves accessing DOM elements and creating a button ( $O(1)$ ), and then copying the content identifier itself, with a constant time complexity ( $O(1)$ ).

In terms of time complexity analysis, the best case scenario assumes constant file size and encryption key length, along with fast IPFS upload, resulting in an overall time complexity of  $O(1)$ . On the other hand, the worst case scenario considers a large file size and significant encryption key length, leading to a time complexity of  $O(\text{file size} + \text{key length})$ . It is important to note that the time complexity of asynchronous operations, such as file reading and IPFS uploading, can be influenced by implementation details and external factors like network speed. Therefore, the provided analysis assumes ideal scenarios for simplicity.

## 4.4 Comparison and Conclusion

The two encryption algorithms' comparative study enables an assessment of their security, effectiveness, and performance. The effectiveness of our system's encryption technique in terms of encryption time for various file sizes is shown in Graph 1 for comparison. The Hill cipher scheme's flaws are shown in Graph 2, which also presents the suggested cipher scheme as an upgrade with more robust security measures. Potential areas for development for each algorithm may be found by comparing their advantages and disadvantages.

In conclusion, this comparison of the performance and security features of two encryption algorithms offers useful information. These results can assist academics and practitioners evaluate and implement secure encryption systems for particular applications, as well as direct future improvements and optimizations to the encryption algorithm in our system.

## 4.5 summary

To guarantee transparency, security, and immutability of certificate records, the system design uses blockchain technology, more especially the Ethereum platform. A smart contract with sound architecture that manages certificate storage and verification is a crucial component. Important elements including username, full name, content identification, verifier, verification time, certificate name, certificate type, institute name, and verifier contact are included in the smart contract's structure. The technology takes advantage of the blockchain's decentralized structure to do away with the requirement for a central authority and offer a tamper-proof method of validating and retrieving certificate data.

# Chapter 5

## Business Model

### 5.1 Infrastructure:

The platform has high-speed storage and transaction processing capabilities thanks to its blockchain technology. For the platform to guarantee the security and accuracy of client data, secure connection protocols and encryption are also used.

### 5.2 Product/Service Offering:

The SAAS platform will offer a blockchain-based document verification solution for Southeast Asia. The applicant will be able to submit their papers to the site and verify their legality thanks to a tamper-proof and irreversible blockchain ledger. The platform will offer scalable verification processes that can be customized to meet the specific needs of each customer.

### 5.3 Revenue Model:

The site will earn money through a subscription-based business model. Users must purchase a monthly or annual membership in order to access the platform's document verification services. How many papers need to be verified, and at what level, will determine the price.

#### 5.3.1 Subscription Model:

One way to generate income would be to offer a subscription service where users pay a monthly or yearly fee to access the network. The quantity of verifications finished, the level of service, and the features and functionality offered might all have an impact on the cost of membership. The group could consider offering several pricing tiers to cater to different customer categories like Applicant, Business, and Institution.

### **5.3.2 Transaction Fees:**

Charge a transaction charge as an extra revenue stream for each platform verification. As a transaction fee, the verifier or the requester may be charged a set price or a percentage of the verification value. This strategy may be interesting to businesses and governmental entities who do multiple verifications and are willing to pay for a reliable and secure platform.

## **5.4 Marketing Strategy:**

The marketing strategy will be focused on establishing partnerships with businesses and governmental bodies in Southeast Asia to increase platform usage. The platform will also leverage digital marketing tools like content marketing, search engine optimization, and social networking to reach potential customers.

### **5.4.1 Identify Target Customers:**

Finding the platform's target clients would be the first stage in creating a marketing plan. The team can think about focusing on Southeast Asian companies, governments, and educational institutions that need safe and dependable document verification solutions.

### **5.4.2 Develop a Brand and Messaging:**

After identifying the target market, the team would then need to develop a brand and messaging that appeals to them. The platform's branding should differentiate it from competitors and convey its advantages. The value proposition, benefits, and unique qualities of the platform should be convincingly and clearly communicated in the messaging.

### **5.4.3 Website and Content Marketing:**

The team may create a website with information on the platform's features, benefits, and prices as well as a form for potential users to sign up for a free trial or subscription. The organization may also launch a blog that provides intelligent articles on document verification, blockchain, and related topics and promote it via email marketing, social media, and other channels.

#### **5.4.4 Search Engine Optimization:**

To increase the platform's visibility in search engines, the team may optimize the website for relevant keywords and phrases related to document verification and blockchain technology. This might comprise creating high-quality content, building high-quality backlinks, and enhancing the website's metadata and structure.

#### **5.4.5 Paid Advertising:**

The group can also consider using sponsored advertising to target potential customers in Southeast Asia. Using online advertising tools like Google Ads, Facebook Ads, and LinkedIn Ads to target certain customer demographics and boost website traffic can help with this. The team would need to carefully assess the effectiveness of the ads and adjust the language and targeting as needed in order to make them as effective as feasible.

### **5.5 Technical Support:**

Customers may access technical support from the platform via a number of methods, including email, phone, and live chat. The technical support team will take care of any platform-specific technical issues.

### **5.6 Data Security:**

To guarantee the security and confidentiality of client data, the platform will apply stringent data protection and privacy rules. Also, the site will abide by all Southeast Asian legal and regulatory standards.

### **5.7 Operational Plan:**

The platform will be managed by a team of seasoned blockchain technologists and industry professionals. The team will be responsible for platform upkeep, client assistance, and continuous platform feature and functionality improvements.

#### **5.7.1 Team Building:**

The first step in the operational plan would be to assemble a team with the skills required to design and maintain the platform. These might include customer service representatives, software developers, UX designers, and marketing specialists. The team should be able to work together and have a mix of technical and business skills to achieve the organization's goals.



### **5.7.2 Infrastructure Setup:**

The next phase would include building the platform's infrastructure. This would need selecting and configuring the appropriate blockchain technology, database, server, and storage options. The infrastructure needs to be flexible and scalable, able to handle enormous volumes of data and transactions, and improved in terms of speed and security.

### **5.7.3 Platform Development:**

After the infrastructure was in place, the group would begin constructing the platform. This would comprise designing the user interface, developing the verification process, integrating with external systems, and more in addition to producing the necessary features and functionality. The platform must be user-friendly, intuitive, and able to meet the particular needs of Southeast Asian customers.

### **5.7.4 Testing and Quality Assurance:**

To make sure the platform is dependable, secure, and error-free before deploying it, the team must carry out extensive testing and quality assurance. This would entail performing numerous tests, including functional, performance, and security tests, and resolving any issues that are found. The team must do comprehensive testing and quality assurance before deploying the platform to ensure that it is dependable, secure, and error-free. To do this, a variety of tests would need to be run, including functional, performance, and security tests, and any flaws that were discovered would need to be fixed.

### **5.7.5 Launch and Customer Acquisition:**

Once the platform was ready, the team would launch it and begin acquiring Southeast Asian clients. This would require developing a marketing strategy, partnering with businesses and governmental agencies, and reaching out to potential customers through a number of media. Feedback from customers is necessary so that the team can constantly adjusting the platform to suit their needs and preferences.

### **5.7.6 Customer Support and Maintenance:**

After the platform is up and running, the group would have to continue providing client support and maintenance. This would comprise addressing any technical issues, helping users set up and utilize the platform, and continually improving the platform based on user feedback. The team should also monitor the platform's usability, security, and scalability and take preventative measures to assure its continued success.

### **5.7.7 Expansion and Growth:**

As the platform gains traction and has a sizable clientele in Southeast Asia, the team might consider expanding to other regions and markets. This may mean growing the platform to service a larger customer base, establishing connections with institutions and governments abroad, and developing new features and functionalities.

Overall, this SAAS approach for a Southeast Asian blockchain-based document verification system provides a scalable and creative answer to a developing issue in the area. This SAAS strategy has the potential to develop into a lucrative business with the appropriate team and finance.

## **5.8 Cost Analysis**

For a blockchain-based DApp, hardware resources such as servers, storage, and networking devices are required. It also requires user interfaces, blockchain infrastructure software, and software for building smart contracts. Employing testers, UI/UX designers, and blockchain engineers could be essential. In order to determine the required resource levels, we must estimate the quantities of hardware and software resources, as well as the quantity of labor required to design, create, and maintain the DApp.

### **5.8.1 Project Scope and Objectives:**

The project's objectives and scope may include developing a secure method for organizations to verify documents' validity using blockchain technology. Interaction with external systems, document posting, verification, and retrieval are all potential aspects. Performance requirements such as scalability, reliability, and security are possible.

### **5.8.2 Resource Needs:**

Hardware resources like servers, storage, and networking gear, software resources like programs for creating smart contracts and blockchain infrastructure, and people resources like blockchain developers and UI/UX designers may all be required for the project. It is important to pay close attention to the unique needs of blockchain technology, such as the requirement of scattered nodes to maintain decentralization.

### **5.8.3 Labor Costs:**

Salaries, benefits, and overhead for UI/UX designers and blockchain engineers with the unique skill sets needed for blockchain development may be included in the project's labor expenses.

#### **5.8.4 Blockchain Platform Costs:**

Licensing costs, upkeep, and upgrades for the blockchain platform utilized for development may be included.

#### **5.8.5 Infrastructure Costs:**

Servers, storage, and networking costs may be incurred as part of the infrastructure needed to develop and operate the blockchain network, with specific attention paid to the decentralized nature of blockchain technology.

#### **5.8.6 Security Costs:**

Cybersecurity safeguards and auditing might add to the cost of keeping the blockchain network secure.

#### **5.8.7 Smart Contract Costs:**

The expense of creating, testing, and deploying the smart contract may be included in the costs related to designing and implementing them for document verification.

#### **5.8.8 Indirect Costs:**

Marketing and sales expenses are examples of indirect project costs.

#### **5.8.9 Contingency Costs:**

To provide for unforeseen expenses or project overruns, a contingency fund should be established.

#### **5.8.10 ROI Analysis:**

Based on income from document verification services and cost savings from doing away with manual document verification procedures, the project's anticipated return on investment may be calculated.

#### **5.8.11 Risk Analysis:**

Technical risks like scalability and dependability, operational risks like adoption rates and regulatory compliance, and financial concerns like cost overruns are all possible project hazards. To reduce these risks, a risk management strategy should be created.

## 5.9 Protocol

Initially, it determines the verification time and then takes the necessary steps: if the time is greater than three units, the request is submitted again to the institution, and the user receives a partial refund; if the time is zero, a complete refund is issued. Second, if the institution doesn't reply, communication is made via alternative channels like phone or email. Increased processing times, the addition of resources, and the usage of automated email alerts for users are just a few of the steps taken to manage the large amount of verification requests. Clear instructions, instructional videos, frequently asked questions, and channels for chat or email help all improve user support. Automated fraud detection technologies, user reporting processes, and activity monitoring are used to stop platform abuse. Users are alerted when a certificate is fraudulent or invalid, and they have the option to challenge the decision by providing further information or filing an appeal. Also, based on network speed, session timeout durations are modified, file size restrictions are imposed for certificate uploads, and file format compatibility checks are made. Additional safety precautions include only allowing company emails to register, uploading one certificate at a time, assigning registrar office verifiers, and routinely verifying their validity through HR. Institutions are notified of verification requests through email and both systems, and confirmed certificates are stamped with a watermark. The issuing institution is contacted, proof is obtained, and a challenge request is made with a justification and supporting documentation if a certificate is shown to be invalid. If the challenge is successful, the certificate is changed to reflect validity; if it is unsuccessful, there may be further alternatives available, such as making an appeal to higher authorities or hiring legal representation. Valid certificates don't need to be renewed.

---

**Algorithm 4: System Protocol**

---

```
1 ForFordoEndFor
   Input: verification_time
2 If verification_time > 3 then
3   | Resend the request to the institution and provide a portion of the refund
   | back to the user.
4 ElseIf verification_time = -1 then
5   | Provide a full refund to the applicant.
   Input: institution_response
6 If institution_response = "Not responding" then
7   | Contact the institution through phone or email.
8 Switch High volume of verification requests do
9   Case true do
10  | Implement measures to increase processing times;
    |
    | • Add additional staff or resources to handle requests.
    |
    | • Implement an automated email notification system to update users on
    |   verification status.
    |
    | • Communicate transparently with users about processing times and delays.
11 Switch Users having difficulty using the platform or navigating verification process do
12   Case true do
13   | Provide clear and accessible documentation and user support;
    |
    | • Create tutorial videos.
    |
    | • Provide FAQ section.
    |
    | • Offer chat or email support.
    |
    | • Regularly collect user feedback and use it to improve user experience.
14 Switch Platform not being used as intended or being abused do
15   Case true do
16   | Implement measures to detect and prevent misuse;
    |
    | • Implement automated fraud detection tools.
    |
    | • Create reporting mechanism for users to report abuse.
    |
    | • Regularly monitor and analyze user activity to identify patterns of misuse.
17 Switch Certificate found to be fraudulent or invalid do
18   Case true do
19   | Flag the certificate and notify the user;
    |
    | • Provide clear process for challenging the decision.
    |
    | • Allow user to provide additional evidence or appeal the decision.
20 Switch Session Timeout do
21   Case true do
```

## 5.10 Selling Prices for Certificate Verification

Table 5.1: Selling Prices for Certificate Verification

| Profit Margin (%) | Operational Expenses (tk) | Gas Fee (tk) | Total Cost (tk) | Selling Price (tk) |
|-------------------|---------------------------|--------------|-----------------|--------------------|
| 10                | 80                        | 120          | 200             | 220                |
| 15                | 80                        | 120          | 200             | 230                |
| 20                | 80                        | 120          | 200             | 240                |
| 10                | 100                       | 130          | 230             | 253                |
| 15                | 100                       | 130          | 230             | 263                |
| 20                | 100                       | 130          | 230             | 273                |
| 10                | 120                       | 140          | 260             | 286                |
| 15                | 120                       | 140          | 260             | 296                |
| 20                | 120                       | 140          | 260             | 306                |
| 10                | 140                       | 150          | 290             | 319                |
| 15                | 140                       | 150          | 290             | 329                |
| 20                | 140                       | 150          | 290             | 339                |
| 10                | 160                       | 160          | 320             | 352                |
| 15                | 160                       | 160          | 320             | 362                |
| 20                | 160                       | 160          | 320             | 372                |

The following table 5.2 displays certificate verification selling prices in a professional setting. It provides a number of scenarios with varied operating costs, profit margins, and gas prices. Although operational expenditures show the costs involved in maintaining the certificate verification system, profit margin reflects the intended percentage of profit for each certificate verification. The transaction cost on the blockchain is referred to as the gas fee. The table shows how various combinations of operating costs, gas surcharges, and profit margins affect the overall cost and ultimately decide the selling price.

Table 5.2: Subscription Prices for Certificate Verification System (in Tk)

| <b>Subscription Tier</b> | <b>Stakeholders Included</b>   | <b>Features Included</b>                        | <b>Monthly Price (Tk)</b> |
|--------------------------|--|---|---------------------------|
| Basic                    | Individuals, Small businesses  | Certificate verification for up to 5 documents  | 199 Tk                    |
| Standard                 | Individuals, Small businesses  | Certificate verification for up to 10 documents | 299 Tk                    |
| Advanced                 | Individuals, Small businesses  | Certificate verification for up to 15 documents | 399 Tk                    |
| Professional             | Individuals, Small businesses, Companies                             | Certificate verification for up to 20 documents | 499 Tk                    |
| Enterprise               | Customizable plan for institutes, companies, and large organizations | Customizable features and document limits       | Contact for pricing       |

Table 5.3 shows the estimated price for the system stakeholder based on the subscription package. There are some differentiations from basic to enterprise level and the monthly fees are different as well.

# Chapter 6

## Result analysis

### 6.1 Performance Analysis

A prototype of the certificate verification system was created utilizing the Ethereum blockchain technology in order to undertake the performance analysis. Moreover, it required the use of a test network with several nodes running on cloud-based virtual computers. A collection of synthetic data depicting academic credentials from a significant educational institution was used in the studies.

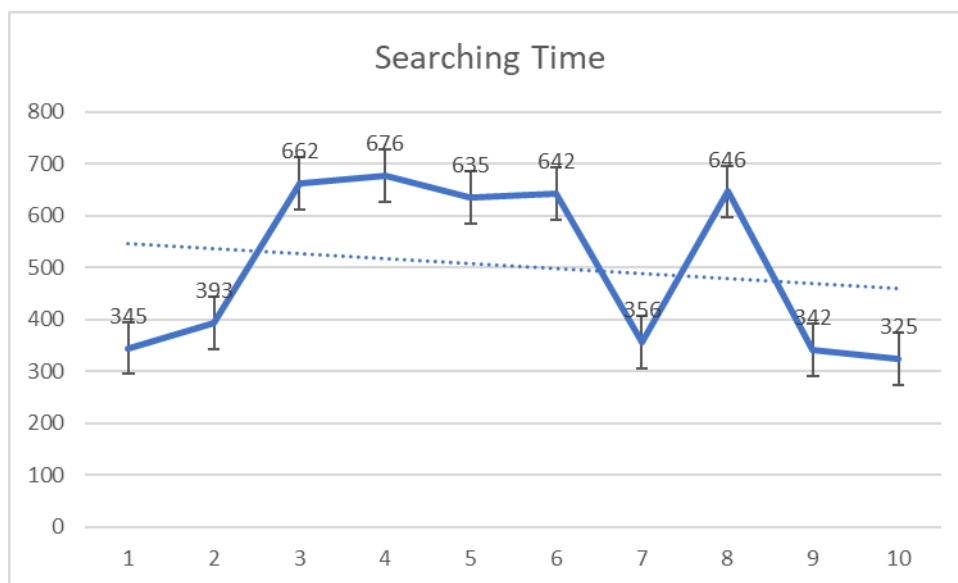


Figure 6.1: Search Time

Figure 6.1 compares the time spent searching for certificates against the number of successful retrievals. The data indicates that the system may not be consistently efficient for all certificate retrievals because there is some variance in the search time values throughout the data set. There may be substantial discrepancies in the amount of time needed to get certificates, according to the range of search times, which range from 325 to 676 milliseconds.



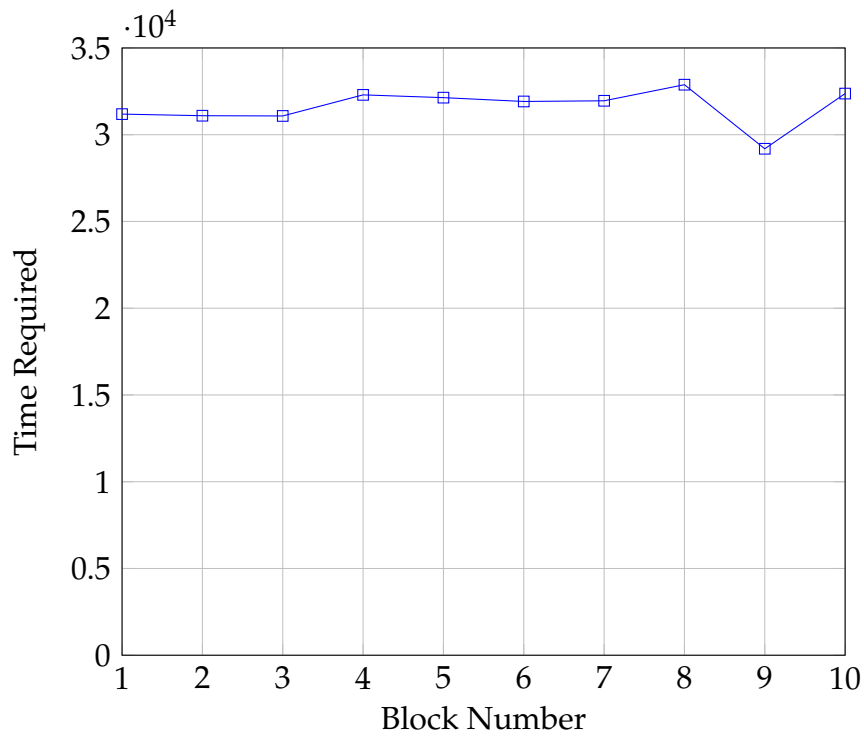


Figure 6.2: Time Required for Each Block Number

Figure 6.2 represents the time required for different block numbers. Each block number is associated with a specific time value, indicating the amount of time it takes to process or complete that particular block. The time values in the data range from 29190 to 32882, showing a spread of durations. The data does not follow a consistent pattern or exhibit a clear trend, with some blocks requiring more time than others. This suggests that the time required for each block is independent of the block number itself, and other factors likely influence the processing time. Further analysis and contextual information would be needed to understand the specific factors affecting the time required for each block.

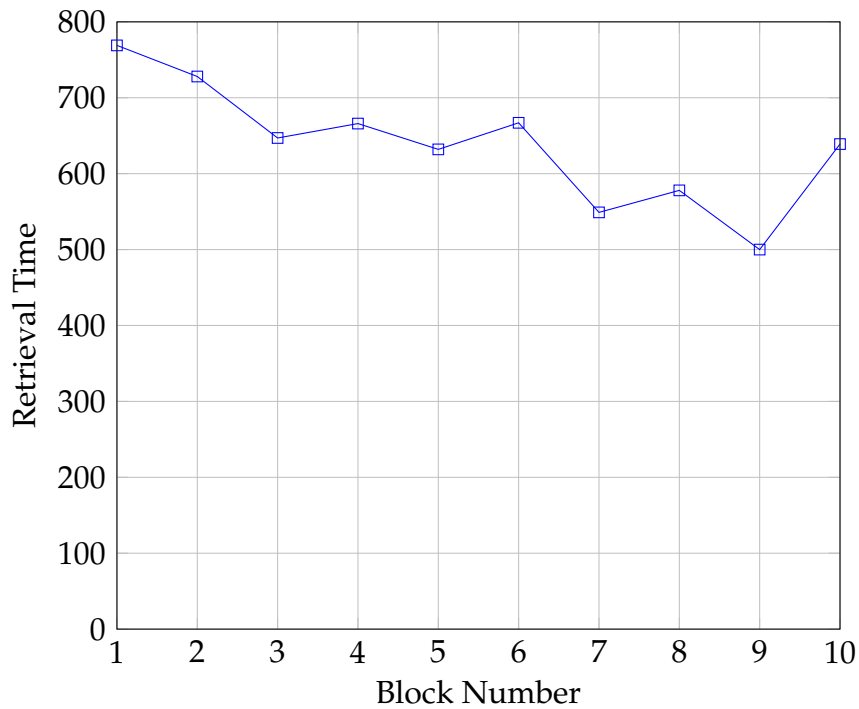


Figure 6.3: Retrieval Time for Each Block Number

Figure 6.3 represents the relationship between block numbers and retrieval time. Each block number is associated with a specific retrieval time value, indicating the time taken to retrieve the corresponding block. The data points show some variations in retrieval time across the different block numbers, but there is no clear linear or exponential trend observed. The pattern suggests that the retrieval time is influenced by various factors, such as the characteristics of the blocks or the retrieval system itself. Possible reasons for the variations in retrieval time could include differences in block sizes, data access patterns, caching mechanisms, or the efficiency of the retrieval algorithm.

Table 6.1: Test Cases: Certificate Submission and Verification

| <b>Test Case: Certificate Submission and Verification</b>          | <b>Expected Result</b>   | <b>Actual Result</b>   |
|--|--|--|
| <b>Test Scenario: Submitting and verifying a valid certificate</b> | The system should successfully store the certificate in IPFS, generate a unique hash code, store the hash code in the blockchain, and retrieve the verified certificate when searched for by the applicant's information.                        | The certificate submission process was smooth, and the system stored the certificate in IPFS. The hash code was generated and successfully stored in the blockchain. When verifying the certificate using the applicant's information, the system retrieved the correct certificate and displayed the verified details.  |
| <b>Test Case: Verification of Multiple Certificates</b>            | The system should successfully store and associate multiple certificates with the applicant's information. When verifying the certificates using the applicant's details, the system should retrieve and display all the verified certificates.  | The system successfully stored and associated multiple certificates with the applicant's information. When verifying the certificates using the applicant's details, the system accurately retrieved and displayed all the verified certificates, ensuring that none were missed or incorrectly linked to the applicant. |
| <b>Test Scenario: Performance and scalability testing</b>          | The system should handle the increased load without significant degradation in performance. Response time should remain within acceptable limits, and resource usage should scale efficiently with the growing number of users and certificates. | During the performance and scalability testing, the system maintained acceptable response times even with a large number of concurrent users and certificates. Resource usage increased proportionately, indicating good scalability.  |

## 6.2 Comparative Analysis

The aim of this comparative analysis is to evaluate and compare the performance of parallelization and bulk certificate addition in two different certificate verification systems. Graphs from our own system and a research paper will be analyzed to assess the efficiency and scalability of these approaches.

### 6.2.1 Graph 1: Our System

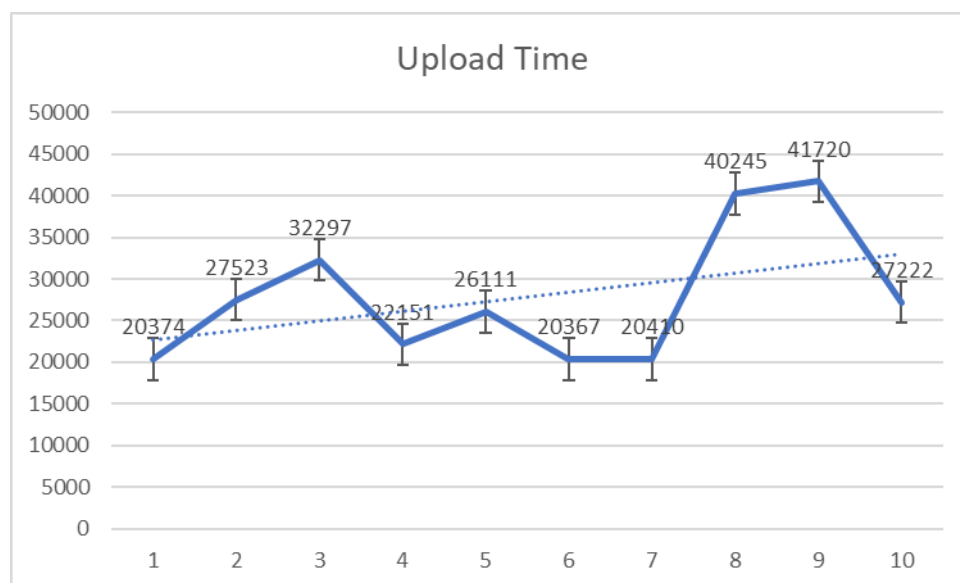


Figure 6.4: Graph from Our System

Figure 7.3: The first graph illustrates the performance of our certificate verification system in terms of parallelization and bulk certificate addition. It showcases the relationship between the number of parallel requests or bulk certificate additions and the corresponding response time or processing speed.

## 6.3 Graph 2: Research Paper



Figure 6.5: Graph from Research Paper

Figure 7.4: The second graph, obtained from a research paper, represents the performance of an alternative certificate verification system in parallelizing requests and adding certificates in bulk. It presents similar metrics as Graph 1 to assess the system's efficiency.

### 6.3.1 Comparative Points

#### 1. Parallelization Capability:

- Graph 1 demonstrates the parallelization capability of our system by showcasing how the response time decreases as the number of parallel requests increases. This indicates the system's ability to handle multiple verification requests concurrently.
- Graph 2 from the research paper also exhibits the parallelization capability of the alternative system, providing insights into how its response time improves with an increasing number of parallel requests.

#### 2. Bulk Certificate Addition:

- In Graph 1, the performance of our system is depicted concerning bulk certificate addition. It illustrates how the processing speed increases as more certificates are added in bulk, demonstrating the system's ability to efficiently handle bulk operations.
- Graph 2 presents a similar analysis for the alternative system, showcasing its ability to handle bulk certificate addition and its impact on processing speed.

### **6.3.2 Comparison and Conclusion**

The importance of parallelization and bulk certificate adding in enhancing the effectiveness and scalability of certificate verification systems is highlighted in both Graphs 1 and 2. By comparing the two graphs, it is possible to assess how well our system performs in terms of parallelization and bulk certificate adding when compared to the system that is described in the study report. We can determine the strengths, shortcomings, and possible areas for development for both systems by looking at the trends, patterns, and performance indicators in the graphs.

The two certificate verification systems' performance, effectiveness, and scalability have been compared in this comparative examination of their parallelization and bulk certificate adding capabilities. These results can help us improve and optimize our system further so that it can successfully handle requests for blockchain-based certificate verification.

## **6.4 Discussion**

Examining the precision and dependability of blockchain technology in the verification of academic credentials is one of the testing's key goals. The outcomes showed that blockchain-based verification offers a very safe and impenetrable way to verify credentials. We were able to do away with the requirement for middlemen and ensure the integrity and authenticity of certificates by utilizing the decentralized nature of blockchain.

The use of a blockchain-based system for verifying academic credentials also has tremendous promise for reducing administrative procedures. The solution enabled effective and real-time verification, saving the time and effort needed for human verification through the use of smart contracts and automated verification algorithms. This has significant ramifications for organizations, businesses, and people since it makes the validation of academic credentials quicker and easier.

# Chapter 7

## Conclusions

### 7.1 Social, Legal, Ethical, and Environmental Issues

The method intends to address issues with traditional verification techniques related to social, ethical, and environmental problems. It aids in addressing problems like phony certificates, where people may exaggerate their credentials and harm the prospects and justice for others. The solution provides transparency, immutability, and traceability of certificate data by utilizing blockchain technology, lowering the possibility of fraud and boosting confidence in the verification process. Furthermore, by offering a trustworthy and impervious means to check credentials, it encourages fair access to educational and career possibilities. By decreasing the need for paper-based paperwork and easing administrative duties, the technology also aids in environmental sustainability.

#### 7.1.1 Social

A blockchain and IPFS based certificate verification system can have several social problems, including:

- **Trust issues:** Users must have faith in the parties engaged in the verification process for a certificate verification system to function. If the parties are not transparent or are thought to be prejudiced, this trust may be damaged.
- **Privacy concerns:** As sensitive and personal data may be accessed by anybody, storing it on a public blockchain or IPFS network can cause users to worry about their privacy.
- **Inadequate security:** The accuracy of the data contained in the system might be compromised if the system is not properly secured against hacking.
- **Unequal access:** A blockchain-based and IPFS-based certificate verification system could not be available to everyone, especially in places with restricted access to technology. Inequalities in society and the economy may result from this.

- **Technical challenges:** Technological obstacles including scalability, interoperability, and compatibility might hinder the system's adoption and efficacy.
- **Misuse of information:** If certificates and personal data are compromised, they might be utilized for malicious purposes. This may result in fraud, identity theft, or other nefarious behaviors.

### 7.1.2 Ethical

A blockchain and IPFS based certificate verification system can raise several ethical issues, including:

- **Privacy:** Because sensitive data may be accessed by anyone, using a public blockchain and IPFS network to store sensitive information might cause privacy problems.
- **Centralization:** If a small number of companies control the bulk of the network's nodes or processing capacity, blockchains may become centralized in practice, resulting in uneven representation and decision-making.
- **Discrimination:** The system may encourage prejudice based on racial, gender, religious, or other distinctions, or it might establish new biases.
- **Accessibility:** Because not everyone can use the technology, there may be moral issues with injustice and prejudice.
- **Ownership:** In especially when it comes to personal data, the issue of who owns the data kept on the blockchain and IPFS network might be a complicated ethical one.
- **Responsibility:** There is a chance that the technology may be abused, either for censorship or monitoring reasons, and it is not obvious whose responsibility it is to handle these problems.
- **Tampering:** Data manipulation is possible if the system is not adequately protected, which might have major ethical repercussions.

Designing and implementing a blockchain and IPFS based certificate verification system must take into account these ethical concerns in order to avoid possible harm and maximize the technology's advantages.



### 7.1.3 Environmental

A certificate verification system built on the blockchain and IPFS may contribute to a number of environmental problems, such as:

- **High energy consumption:** The “mining” process of a blockchain network, which entails confirming transactions, uses a lot of processing power and energy, resulting in high energy consumption and potentially harmful environmental effects.
- **Electronic waste:** The creation and destruction of electronic equipment utilized in the IPFS network and blockchain might result in electronic waste and environmental degradation.
- **Physical infrastructure:** Data centers, which are part of the infrastructure needed to operate IPFS and the blockchain, can use a lot of resources and pollute the environment.
- **Carbon footprint:** The blockchain and IPFS network’s energy use can have a sizable carbon footprint, which fuels global warming.

## 7.2 Brief Summary

In conclusion, the process of verifying academic qualifications is essential and necessitates a safe and dependable system. The development of blockchain technology in recent years has provided a potentially effective remedy for the problems related to the verification process. The reliability and trustworthiness of academic credentials can be improved by blockchain technology by offering a decentralized and tamper-proof alternative.

Moreover, the integration of IPFS with blockchain technology provides a practical way to store and validate academic credentials. The establishment of a strong and secure storage system that can guarantee the legitimacy of academic credentials is made possible by the immutability of the blockchain and the decentralized nature of IPFS. Blockchain and IPFS offer a trustworthy and tamper-proof solution that can reduce the danger of fraudulent activity by offering a permanent and visible record of the credentials.

Blockchain and IPFS have a lot going for them when it comes to academic degree verification, according to an examination of the literature and case studies that have already been published. These technologies can save expenses and time-consuming bureaucratic procedures while improving the efficiency and reliability of the verification process. The acceptance, scalability, and interoperability of blockchain and IPFS in academic institutions, however, may present some difficulties.

Besides these difficulties, IPFS and blockchain are considered to be viable future solutions because of the advantages they provide for academic degree verification. The verification process is probably going to get faster, more dependable, and more secure as more institutions use these technologies.

### **7.3 Future works**

Blockchain technology has been more popular recently in academic credential verification systems thanks to its advantages including immutability, tamper-proof certifications, and increased confidence. It deals with problems including certificate fraud, forgery, and slow verification. Online credentialing platforms, immigration procedures, deep learning for predictive modeling, trade schools and apprenticeships, healthcare sector verification, government applications, and professional licensing can all benefit from blockchain-based verification systems. However, further investigation is required to examine topics like biometric identification, access control systems, and customized suggestions utilizing machine learning algorithms. These developments might speed up the verification procedure and increase the effectiveness of academic credential verification systems as a whole.

Online credentialing systems, which provide certification programs for a range of industries including technology, business, healthcare, and more, are growing in popularity. These platforms provide people an accessible and practical means to acquire the knowledge and skills required to do their jobs, sometimes at a cheaper cost than traditional education programs. The verification of the certifications, however, is one of the key issues with online credentialing platforms. Employers want a dependable technique to confirm the validity of these credentials because the problem of false certificates has increased with the growth of online education.

Blockchain and machine intelligence are both capable of significantly enhancing academic certificate verification methods. Large datasets of academic credentials may be analyzed using machine learning techniques in order to spot trends and abnormalities that can point to fraud or manipulation. In order to make it simpler to spot false certificates that could have been created using a template, machine learning algorithms can, for instance, be used to find trends in the typefaces, layouts, and formatting of certificates.

The verification of educational qualifications for foreigners looking to study or work abroad might be revolutionized by blockchain technology. This is due to the fact that conventional means of authenticating educational qualifications are sometimes time-consuming, difficult, and expensive third parties. Immigration officials may swiftly and efficiently confirm the legitimacy of a person's academic credentials without the

need for human verification or dependence on third-party organizations by storing educational credentials on a decentralized and tamper-proof blockchain network.

Professional licensing is a highly regulated industry that calls for people to receive licenses or certificates in order to perform in specific professions. This covers, among others, occupations like those of a doctor, an attorney, an engineer, and an accountant. The process of earning a professional license often entails extensive training and study, followed by an exam that assesses a person's subject-matter expertise. The information may be securely and decentralizedly kept on a blockchain. Included in this would be details like the institution's name, the kind of license or certification, the date of issuance, and the expiration date.

# References

- Abdullahi, M. U., Aimufua, G., & Aminu, A. (n.d.). Certificate generation and verification system using blockchain technology and quick response code.
- Angelis, J., & Da Silva, E. R. (2019). Blockchain adoption: A value driver perspective. *Business Horizons*, 62(3), 307–314.
- Athanere, S., & Thakur, R. (2022). Blockchain based hierarchical semi-decentralized approach using ipfs for secure and efficient data sharing. *Journal of King Saud University-Computer and Information Sciences*, 34(4), 1523–1534.
- Augot, D., Chabanne, H., Chenevier, T., George, W., & Lambert, L. (2017). A user-centric system for verified identities on the bitcoin blockchain. In *Data privacy management, cryptocurrencies and blockchain technology: Esorics 2017 international workshops, dpm 2017 and cbt 2017, oslo, norway, september 14-15, 2017, proceedings* (pp. 390–407).
- Ayub Khan, A., Laghari, A. A., Shaikh, A. A., Bourouis, S., Mamlouk, A. M., & Alshazly, H. (2021). Educational blockchain: A secure degree attestation and verification traceability architecture for higher education commission. *Applied Sciences*, 11(22), 10917.
- Balogh, S., Gallo, O., Ploszek, R., Špaček, P., & Zajac, P. (2021). Iot security challenges: Cloud and blockchain, postquantum cryptography, and evolutionary techniques. *Electronics*, 10(21), 2647.
- Beck, R. (2018). Beyond bitcoin: The rise of blockchain world. *Computer*, 51(2), 54–58.
- Bhutta, M. N. M., Khwaja, A. A., Nadeem, A., Ahmad, H. F., Khan, M. K., Hanif, M. A., ... Cao, Y. (2021). A survey on blockchain technology: Evolution, architecture and security. *Ieee Access*, 9, 61048–61073.
- Bond, F., Amati, F., & Blousson, G. (2015). Blockchain, academic verification use case. *Buenos Aires*.
- Castro, R. Q., & Au-Yong-Oliveira, M. (2021). Blockchain and higher education diplomas. *European Journal of Investigation in Health, Psychology and Education*, 11(1), 154–167.
- Cheng, J.-C., Lee, N.-Y., Chi, C., & Chen, Y.-H. (2018). Blockchain and smart contract for digital certificate. In *2018 ieee international conference on applied system invention (icasi)* (pp. 1046–1051).
- Dai, Q.-y., Zhang, B., & Dong, S.-q. (2022). A ddos-attack detection method oriented to the blockchain network layer. *Security and Communication Networks*, 2022.

- Dalal, J., Chaturvedi, M., Gandre, H., & Thombare, S. (2020). Verification of identity and educational certificates of students using biometric and blockchain. In *Proceedings of the 3rd international conference on advances in science & technology (icast)*.
- Desai, S. (2018). New recordkeeping on the block: An assessment of 2 blockchain-based recordkeeping systems.
- Diaconita, V., Belciu, A., & Stoica, M. G. (2023). Trustful blockchain-based framework for privacy enabling voting in a university. *Journal of Theoretical and Applied Electronic Commerce Research*, 18(1), 150–169.
- Dib, O., Brousmiche, K.-L., Durand, A., Thea, E., & Hamida, E. B. (2018). Consortium blockchains: Overview, applications and challenges. *Int. J. Adv. Telecommun*, 11(1), 51–64.
- Domingue, J. (2017). *Blockchains as a component of the next generation internet*.
- Dongre, J. G., Tikam, S. M., & Gharat, V. B. (2020). Education degree fraud detection and student certificate verification using blockchain. *Int. J. Eng. Res. Technol*, 9, 300–303.
- Faaroeek, S. A., Panjaitan, A. S., Fauziah, Z., & Septiani, N. (2022). Design and build academic website with digital certificate storage using blockchain technology. *IAIC Transactions on Sustainable Digital Innovation (ITSDI)*, 3(2), 175–184.
- Fekete, D. L., & Kiss, A. (2023). Toward building smart contract-based higher education systems using zero-knowledge ethereum virtual machine. *Electronics*, 12(3), 664.
- Foytik, P., Shetty, S., Gochhayat, S. P., Herath, E., Tosh, D., & Njilla, L. (2020). A blockchain simulator for evaluating consensus algorithms in diverse networking environments. In *2020 spring simulation conference (springsim)* (pp. 1–12).
- Gai, K., Guo, J., Zhu, L., & Yu, S. (2020). Blockchain meets cloud computing: A survey. *IEEE Communications Surveys & Tutorials*, 22(3), 2009–2030.
- Ge, Z., Loghin, D., Ooi, B. C., Ruan, P., & Wang, T. (2022). Hybrid blockchain database systems: design and performance. *Proceedings of the VLDB Endowment*, 15(5), 1092–1104.
- Ghazali, O., & Saleh, O. S. (2018). A graduation certificate verification model via utilization of the blockchain technology. *Journal of Telecommunication, Electronic and Computer Engineering (JTEC)*, 10(3-2), 29–34.
- Glaser, F., Hawlitschek, F., & Notheisen, B. (2019). Blockchain as a platform. *Business Transformation through Blockchain: Volume I*, 121–143.
- Gourisetti, S. N. G., Mylrea, M., & Patangia, H. (2019). Evaluation and demonstration of blockchain applicability framework. *IEEE Transactions on Engineering Management*, 67(4), 1142–1156.
- Gräther, W., Kolvenbach, S., Ruland, R., Schütte, J., Torres, C., & Wendland, F. (2018). Blockchain for education: lifelong learning passport. In *Proceedings of 1st ercim blockchain workshop 2018*.

- Gupta, S., & Sadoghi, M. (2021). Blockchain transaction processing. *arXiv preprint arXiv:2107.11592*.
- Han, M., Li, Z., He, J., Wu, D., Xie, Y., & Baba, A. (2018). A novel blockchain-based education records verification solution. In *Proceedings of the 19th annual sig conference on information technology education* (pp. 178–183).
- Hasan, M., Rahman, A., & Islam, M. J. (2020). Distb-cvs: A distributed secure blockchain based online certificate verification system from bangladesh perspective. In *2020 2nd international conference on advanced information and communication technology (icaict)* (pp. 460–465).
- Hellman, M. E. (2002). An overview of public key cryptography. *IEEE Communications Magazine*, 40(5), 42–49.
- Kanan, T., Obaidat, A. T., & Al-Lahham, M. (2019). Smartcert blockchain imperative for educational certificates. In *2019 ieee jordan international joint conference on electrical engineering and information technology (jeeit)* (pp. 629–633).
- Khan, S. N., Loukil, F., Ghedira-Guegan, C., Benkhelifa, E., & Bani-Hani, A. (2021). Blockchain smart contracts: Applications, challenges, and future trends. *Peer-to-peer Networking and Applications*, 14, 2901–2925.
- Kiffer, L., Rajaraman, R., & Shelat, A. (2018). A better method to analyze blockchain consistency. In *Proceedings of the 2018 acm sigsac conference on computer and communications security* (pp. 729–744).
- Kosba, A., Miller, A., Shi, E., Wen, Z., & Papamanthou, C. (2016). Hawk: The blockchain model of cryptography and privacy-preserving smart contracts. In *2016 ieee symposium on security and privacy (sp)* (pp. 839–858).
- Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25, 1097–1105.
- Kumutha, K., & Jayalakshmi, S. (2022). Blockchain technology and academic certificate authenticity—a review. *Expert Clouds and Applications: Proceedings of ICOECA 2021*, 321–334.
- Lamkoti, R. S., Maji, D., Gondhalekar, A. B., & Shetty, H. (2021). Certificate verification using blockchain and generation of transcript. *Int. J. Eng. Res. Technol*, 10(3).
- Leka, E., & Selimi, B. (2021). Development and evaluation of blockchain based secure application for verification and validation of academic certificates. *Annals of Emerging Technologies in Computing (AETiC)*, 5(2), 22–36.
- Li, X., Jiang, P., Chen, T., Luo, X., & Wen, Q. (2020). A survey on the security of blockchain systems. *Future generation computer systems*, 107, 841–853.
- Liang, X., & Zhao, Q. (2020). On the design of a blockchain-based student quality assessment system. In *2020 international conference on high performance big data and intelligent systems (hpbdbis)* (pp. 1–7).

- Liu, D., & Guo, X. (2019). Blockchain based storage and verification scheme of credible degree certificate. In *2019 2nd international conference on safety produce informatization (iicspi)* (pp. 350–352).
- Malik, G., Parasrampur, K., Reddy, S. P., & Shah, S. (2019). Blockchain based identity verification model. In *2019 international conference on vision towards emerging trends in communication and networking (vitecon)* (pp. 1–6).
- Mingxiao, D., Xiaofeng, M., Zhe, Z., Xiangwei, W., & Qijun, C. (2017). A review on consensus algorithm of blockchain. In *2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC)* (pp. 2567–2572).
- Monrat, A. A., Schelén, O., & Andersson, K. (2019). A survey of blockchain from the perspectives of applications, challenges, and opportunities. *IEEE Access*, 7, 117134–117151.
- Natarajan, H., Krause, S., & Gradstein, H. (2017). Distributed ledger technology and blockchain.
- Nizamuddin, N., Salah, K., Azad, M. A., Arshad, J., & Rehman, M. (2019). Decentralized document version control using ethereum blockchain and ipfs. *Computers & Electrical Engineering*, 76, 183–197.
- Norta, A. (2017). Designing a smart-contract application layer for transacting decentralized autonomous organizations. In *Advances in computing and data sciences: First international conference, icacds 2016, ghaziabad, india, november 11-12, 2016, revised selected papers 1* (pp. 595–604).
- Nouman, M., Ullah, K., & Azam, M. (2021). Secure digital transactions in the education sector using blockchain. *EAI Endorsed Transactions on Scalable Information Systems*, 9(35).
- Nyalety, E., Parizi, R. M., Zhang, Q., & Choo, K.-K. R. (2019). Blockipfs-blockchain-enabled interplanetary file system for forensic and trusted data traceability. In *2019 IEEE International Conference on Blockchain (Blockchain)* (pp. 18–25).
- Padilla, R., Netto, S. L., & Da Silva, E. A. (2020). A survey on performance metrics for object-detection algorithms. In *2020 international conference on systems, signals and image processing (IWSSIP)* (pp. 237–242).
- Pahlajani, S., Kshirsagar, A., & Pachghare, V. (2019). Survey on private blockchain consensus algorithms. In *2019 1st international conference on innovations in information and communication technology (ICIICT)* (pp. 1–6).
- Pilkington, M. (2016). Blockchain technology: principles and applications. In *Research handbook on digital transformations* (pp. 225–253). Edward Elgar Publishing.
- Quasim, M. T., Khan, M. A., Algarni, F., Alharthy, A., & Alshmrani, G. M. M. (2020). Blockchain frameworks. *Decentralised Internet of Things: A Blockchain Perspective*, 75–89.

- Rahardja, U., Hidayanto, A. N., Putra, P. O. H., & Hardini, M. (2021). Immutable ubiquitous digital certificate authentication using blockchain protocol. *Journal of applied research and technology*, 19(4), 308–321.
- Rasool, S., Saleem, A., Iqbal, M., Dagiuklas, T., Mumtaz, S., & ul Qayyum, Z. (2020). Docschain: Blockchain-based iot solution for verification of degree documents. *IEEE Transactions on Computational Social Systems*, 7(3), 827–837.
- Saleh, O. S., Ghazali, O., & Rana, M. E. (2020). Blockchain based framework for educational certificates verification. *Journal of critical reviews*, 7(3), 79–84.
- Sanka, A. I., Irfan, M., Huang, I., & Cheung, R. C. (2021). A survey of breakthrough in blockchain technology: Adoptions, applications, challenges and future research. *Computer Communications*, 169, 179–201.
- Sathya, A., Panda, S. K., & Hanumanthakari, S. (2021). Enabling smart education system using blockchain technology. In *Blockchain technology: Applications and challenges* (pp. 169–177). Springer.
- Shakan, Y., Kumalakov, B., Mutanov, G., Mamykova, Z., & Kistaubayev, Y. (2021). Verification of university student and graduate data using blockchain technology. *INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL*, 16(5).
- Sohail, M., Khan, M. A., Ahmad, I., & Sohail, O. (2020). Intelligent data encryption scheme for light weighted aiot enabled devices. *Journal of Information Assurance & Security*, 15(1).
- Spanò, R., Massaro, M., Ferri, L., Dumay, J., & Schmitz, J. (2022). Blockchain in accounting, accountability and assurance: an overview. *Accounting, Auditing & Accountability Journal*(ahead-of-print).
- Srivastava, A., Bhattacharya, P., Singh, A., Mathur, A., Prakash, O., & Pradhan, R. (2018). A distributed credit transfer educational framework based on blockchain. In *2018 second international conference on advances in computing, control and communication technology (iac3t)* (p. 54-59). doi: 10.1109/IAC3T.2018.8674023
- Star, T. D. (2023). *Schoolteacher on job for 16 years with 'fake certificate*. Retrieved 2023-01-28, from <https://www.thedailystar.net/country/schoolteacher-job-16-years-fake-certificate-1230901>
- Steichen, M., Fiz, B., Norvill, R., Shbair, W., & State, R. (2018). Blockchain-based, decentralized access control for ipfs. 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)* (pp. 1499–1506).
- Tasatanattakool, P., & Techapanupreeda, C. (2018). Blockchain: Challenges and applications. In *2018 international conference on information networking (icoi)* (pp. 473–475).
- T.B.S. (2022). *Hundreds of sailors get jobs with fake certificates*. Retrieved 2023-01-28, from <https://www.tbsnews.net/bangladesh/hundreds-sailors-get-jobs-fake-certificates-377929>



- Tribune, D. (2022). *Mpo irregularities: Probe finds 1,100 teachers with fake certificates*. Retrieved 2023-01-28, from <https://www.dhakatribune.com/bangladesh/2022/10/07/mpo-irregularities-probe-finds-1100-teachers-with-fake-certificates>
- UNB. (2023). *6 fraud gang members involved in making 144 types of fake certificates held*. Retrieved 2023-01-28, from <https://unb.com.bd/category/Bangladesh/6-fraud-gang-members-involved-in-making-144-types-of-fake-certificates-held/90732>
- Vidal, F., Gouveia, F., & Soares, C. (2019). Analysis of blockchain technology for higher education. In *2019 international conference on cyber-enabled distributed computing and knowledge discovery (cyberc)* (pp. 28–33).
- Vujičić, D., Jagodić, D., & Ranić, S. (2018). Blockchain technology, bitcoin, and ethereum: A brief overview. In *2018 17th international symposium infoteh-jahorina (infoteh)* (pp. 1–6).
- Wang, M., Duan, M., & Zhu, J. (2018). Research on the security criteria of hash functions in the blockchain. In *Proceedings of the 2nd acm workshop on blockchains, cryptocurrencies, and contracts* (pp. 47–55).
- Wang, S., Ouyang, L., Yuan, Y., Ni, X., Han, X., & Wang, F.-Y. (2019). Blockchain-enabled smart contracts: architecture, applications, and future trends. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 49(11), 2266–2277.
- Watanabe, H., Fujimura, S., Nakadaira, A., Miyazaki, Y., Akutsu, A., & Kishigami, J. (2016). Blockchain contract: Securing a blockchain applied to smart contracts. In *2016 IEEE international conference on consumer electronics (icce)* (pp. 467–468).
- Yaga, D., Mell, P., Roby, N., & Scarfone, K. (2019). Blockchain technology overview. *arXiv preprint arXiv:1906.11078*.
- Zhai, S., Yang, Y., Li, J., Qiu, C., & Zhao, J. (2019). Research on the application of cryptography on the blockchain. In *Journal of physics: Conference series* (Vol. 1168, p. 032077).
- Zheng, Z., Xie, S., Dai, H.-N., Chen, X., & Wang, H. (2018). Blockchain challenges and opportunities: A survey. *International journal of web and grid services*, 14(4), 352–375.