* DORIN LELEA *
DORIN.LELEA@UPT.RO
LOCAL 02.04.2026 16:36
20.01.2025 20:28 - 20.01.2028 20:28
UNIVERSITATEA POLITEHNICA DIN TIMISOARA
Angajat UPT
20110420940401bd89332de2924a2697

Transparent and secure management of activities carried out by teachers using blockchain technology

A Thesis Submitted for obtaining
the Scientific Title of PhD in Engineering
from
Politehnica University Timisoara
In the Field of Computers and Information Technology
by

Diana Laura SILAGHI

PhD Supervisor: Prof. univ. dr. ing. Popescu Daniela Elena

BUPT

The PhD thesis series of UPT are:

1. Automation
2. Chemistry
3. Energetics
4. Chemical Engineering
5. Civil Engineering
6. Electrical Engineering
7. Electronic Engineering and Telecommunications
8. Industrial Engineering
9. Mechanical Engineering
10. Computer Science
11. Science and Material Engineering
12. Systems Engineering
13. Energy Engineering
14. Computers and Information Technology
15. Materials Engineering
16. Engineering and Management
17. Architecture
18. Civil Engineering and Installations
19. Electronics, Telecommunications and Information Technologies

Politehnica University Timisoara, Romania, initiated the above series to disseminate the expertise, knowledge, and results of the research carried out within the doctoral school of the university. According to the Decision of the Executive Office of the University Senate No. 14/14.07.2006, the series includes the doctoral theses defended in the university since October 1, 2006.

Copyright © Editura Politehnica – Timisoara, Romania, 2025

This publication is subject to copyright law. The multiplication of this publication, in whole or in part, the translation, printing, reuse of illustrations, exhibit, broadcasting, reproduction on microfilm or any other form is allowed only in compliance with the provisions of the Romanian Copyright Law in force and permission for use obtained in writing from the Politehnica University Timișoara, Romania. The violations of these rights are under the penalties of the Romanian Copyright Law.

Romania, 300223 Timisoara, Bd. Vasile Parvan 2B

Tel./fax 0256 404677

e-mail: editura@upt.ro

BUPT

Recipients of dedication.

SILAGHI Diana Laura

Transparent and secure management of activities carried out by teachers using blockchain technology

PhD theses of UPT, Series X, No. YY, Editura Politehnica, 2025, 145 pages, 61 figures, 18 tables.
ISSN:
ISBN:

Keywords: blockchain, attestation, certificate, diploma, digital badge, educational resources, teachers' evaluation

Summary:

This thesis proposes a blockchain-based solution for the issuance and verification of professional documents of teachers in pre-university education, with the aim of creating a secure digital portfolio that can prove the authenticity of the documents in order to combat inefficiency, document loss and fraud in teacher evaluation processes.
Our solution is based on two smart contracts developed with integrated security measures and gas optimization techniques, one to manage educational resources created by teachers and another to manage credentials such as diplomas, certificates and attestations. Our solution is based on zk-Rollup integration to significantly reduce costs so that it is suitable for the education system.
The research demonstrates how blockchain can contribute to a more efficient and credible evaluation system, supporting the digital transformation of educational administration.

BUPT

Contents

LIST OF FIGURES 6

LIST OF TABLES 8

LIST OF ABBREVIATIONS 9

1. INTRODUCTION 11

1.1. Thesis context 11

1.2. Motivation 12

1.3. Thesis objective 13

1.4. The structure and content of the thesis 14

2. STATE OF THE ART 16

2.1. Blockchain technology 16
2.1.1. Blockchain technology overview 16
2.1.2. Consensus mechanisms 18
2.1.3. Blockchain's layers 19

2.2. Applications of blockchain 21
2.2.1. Blockchain platforms 21
2.2.2. The state of research in blockchain application areas 24

2.3. Summary 31

3. THE STATE OF RESEARCH IN THE FIELD OF CREDENTIAL MANAGEMENT 32

3.1. Overview 32

3.2. Traditional methods for issuing and verifying academic credentials 33

3.3. Digital signature integration in authentication of academic credentials 37

3.4. QR code integration in authentication of academic credentials 38

3.5. Blockchain-based solutions for issuing and verifying credentials 38

3.6. Summary 44

BUPT

4. BLOCKCHAIN-BASED SOLUTION FOR MANAGING TEACHER CREDENTIALS AND ACCESS TO EDUCATIONAL RESOURCES 45

4.1. Description of the problem **45**

4.2. Proposed solution **51**
4.2.1. Mapping document types and issuing institutions within the proposed solution 51
4.2.2. Rationale for blockchain in the proposed solution as an alternative to traditional and advanced databases 52
4.2.3. Our cost-effective blockchain solution for managing credentials and educational resources 53

4.3. Choosing the tools and technologies used to develop the solution **57**
4.3.1. Hardhat, an Ethereum development environment 57
4.3.2. Sepolia, an Ethereum test network 58
4.3.3. Polygon zkEVM Cardona Testnet, a Layer 2 zk-Rollup testing environment 59
4.3.4. Metamask wallet 59
4.3.5. Alchemy, as a blockchain node provider 61
4.3.6. Remix IDE, development environment for Ethereum smart contract 62
4.3.7. Solidity, the language for Ethereum smart contracts 62
4.3.8. Pinata, as the IPFS-based storage 63
4.3.9. Visual Studio Code, code editor 64
4.3.10. HTML and CSS, technologies for front-end development 65
4.3.11. Python, technology for back-end development 65
4.3.12. Firebase, authentication system 65

4.4. Implementation of the prototype **66**
4.4.1. User interface 66
4.4.2. Writing and deploying the smart contracts 74
4.4.3. Modules interaction 78

4.5. Summary **80**

5. EXPERIMENTAL RESULTS 81

5.1. System performance **81**
5.1.1. Gas consumption evaluation 81
5.1.2. Cost evaluation 89
5.1.3. Latency analysis 91

5.2. System efficiency in the teacher evaluation process **95**

5.3. Summary **96**

6. SECURITY ANALYSIS OVER THE SMART CONTRACTS 98

6.1. Overview **98**

6.2. Security analysis over of the smart contract for educational resources **101**
6.2.1. Publishing and verifying the smart contract for educational resources 101

BUPT

6.2.2. Security analysis of the smart contract using SolidityScan tool to identify and assess any potential vulnerabilities and threats 108
6.2.3. Comparative analysis of security results 113

6.3. Security analysis over of the smart contract for credentials management **115**
6.3.1. Publishing and verifying the smart contract for credentials management 115
6.3.2. Security analysis of the attestation smart contract using SolidityScan tool to identify and assess any potential vulnerabilities and threats 119
6.3.3. Comparative analysis of security results 121

6.4. Summary **122**

7. CONCLUSION AND FUTURE WORK 123

7.1 Obtained results **123**

7.2. Original contributions **124**

7.3 Limitations **125**

7.4. Future work **126**

7.5. Publications **126**

APPENDIX A: THE SMART CONTRACT FOR EDUCATIONAL RESOURCES MANAGEMENT 127

APPENDIX B: THE SMART CONTRACT FOR CREDENTIALS MANAGEMENT 130

BIBLIOGRAPHY 133

BUPT

List of figures

Figure 1: Blockchain as a data structure 16
Figure 2: Blockchain's layers. Source: own illustration. 20
Figure 3: The sequential life cycle that governs a smart contract from creation to deployment and execution. Source: own illustration. 22
Figure 4: DeFi domains. Source: own illustration based on the results from [59]. 25
Figure 5: Areas of blockchain research in healthcare. Source: own illustration based on the results from [74]. 27
Figure 6: Areas of blockchain research in IoT. Source: own illustration based on the results from [82] 28
Figure 7: Areas of blockchain research in public administration. Source: own illustration based on the results from [88] 29
Figure 8: Areas of blockchain research in education. Source: own illustration based on the results from [92, 27] 30
Figure 9: Blockcerts workflow. Source: own illustration 39
Figure 10: The University of Nicosia blockchain diploma management workflow. Source: own illustration. 41
Figure 11: Representation of the weight of document types for obtaining meritorious recognition. Source: Own illustration, based on the results from [142] 49
Figure 12: Illustrations of the significant disparity between the teacher's self-evaluation and the assessment conducted by the evaluation committee in the process of obtaining meritorious recognition. Source: Own illustration, based on the results from [143] 50
Figure 13: Detailed illustration of the files and their constituent document types. 51
Figure 14: Entities participating in the development of the documents. 52
Figure 15: Transaction flow in Polygon zkEVM. Source: own illustration. 54
Figure 16: Overview of the proposed architecture of the issuing attestations or credentials. Source: own illustration. 55
Figure 17: Local blockchain accounts. Source: screenshot from our project. 58
Figure 18: Estimation of gas price by MetaMask wallet 60
Figure 19: Pinata's interface. Source: screenshot from our Pinata workspace 64
Figure 20: User interface of the developed dApp. Source: application screenshot. 66
Figure 21: Issuer institution workflow diagram. Source: own illustration. 67
Figure 22: Owner authentication interface. Source: application screenshot. 67
Figure 23: Role assignment interface of the application. Source: application screenshot. 68
Figure 24: Credential issuance module. Source: application screenshot. 69
Figure 25: Uploading educational resources. Source: application screenshot. 70
Figure 26: Credential uploaded to Pinata for secure, decentralized, and verifiable storage. Source: Pinata screenshot. 71
Figure 27: Teachers workflow diagram. Source: own illustration. 72
Figure 28: Teacher's credentials and educational resources within the platform. Source: application screenshot. 73
Figure 29: Verifier workflow diagram. Source: own illustration. 73

BUPT

Figure 30: Compiling the smart contract for educational resources. Source: screenshot from our Remix IDE workspace 76
Figure 31: Educational resources smart contract deployment on Sepolia testnet. 76
Figure 32: Educational resources smart contract deployment on Polygon zkEVM Cardona testnet.... 76
Figure 33: Compiling the smart contract for credentials management. Source: screenshot from our Remix IDE workspace. 78
Figure 34: Credentials smart contract deployment on Sepolia testnet. 78
Figure 35: Credentials smart contract deployment on Polygon zkEVM Cardona testnet..... 78
Figure 36: Gas consumed with different educational resources contract operations 82
Figure 37: Gas consumed with different contract operations within 9 transactions. 83
Figure 38: Gas consumed with adding educational resources within 1000 transactions. 83
Figure 39: Gas used compared to the average value in educational resource contract. 84
Figure 40: Gas consumed with different credential contract operations 85
Figure 41: Gas consumption for credential contract functions over nine transactions 85
Figure 42: Gas consumed with different credential contract operations within 9 transactions. 86
Figure 43: Gas consumed with generating credential within 1000 transactions. 86
Figure 44: Gas used compared to the average value in educational resource contract 87
Figure 45: Comparative gas consumption for credential generation between initial full on-chain contract and optimized contract with IPFS off-chain storage 88
Figure 46:Comparative gas consumption for certificate issuance across BlockMEDC, ZKBAR-V, and the proposed solution 88
Figure 47:Cost deviation across transactions on the Sepolia test network compared to the average value. 91
Figure 48:Latency deviation across transactions on the Sepolia test network compared to the average value. 92
Figure 49:Distribution of transaction latency on Sepolia test netwok across time intervals expressed in seconds. 93
Figure 50: Transaction latency across transactions on the Polygon zkEVM Cardona test network compared to the average value. 93
Figure 51: Timestamped display of credentials and educational resources within selected evaluation intervals. Source: screenshot from our application. 96
Figure 52: Smart contract without any access control 103
Figure 53: Publishing and verifying the smart contract for educational resources on a blockchain explorer. 105
Figure 54: Educational resources smart contract scanning analysis. 106
Figure 55: Severity ranking of vulnerabilities in the smart contract scans conducted with SolidityScan 109
Figure 56: Vulnerabilities identified by SolidityScan for educational resources smart contract 109
Figure 57: Security score of the smart contract for educational resources 113
Figure 58: Publishing and verifying the smart contract for attestations on a blockchain explorer 116
Figure 59: Credential smart contract scanning analysis. 117
Figure 60: Vulnerabilities identified by SolidityScan for attestation smart contract 119
Figure 61: Security score of the smart contract for attestation 121

BUPT

List of tables

Table 1: Comparative analysis of the main types of blockchain architectures 18
Table 2: Types of consensus mechanisms 19
Table 3: Consensus mechanisms' characteristics 19
Table 4: Architectural comparison of main blockchains. 31
Table 5: Characteristics of credentials 33
Table 6: Comparative overview of credential lifecycle stages 37
Table 7: Comparison of Hardhat, Sepolia, and Polygon zkEVM Cardona test environments. 59
Table 8: Characteristics of MetaMask. 61
Table 9: Comparison of blockchain access methods: MetaMask only vs. Alchemy integration 62
Table 10: Characteristics of Solidity 63
Table 11: Comparison between centralized servers and IPFS 64
Table 12: Key components and functionalities of the smart contract for educational resources management. 76
Table 13: Key components and functionalities of the smart contract for attestations management. 78
Table 14: Transaction fee for deploying the smart contracts on Polygon zkEVM Cardona testnet. 90
Table 15: Smart contract deployment cost reduction. 91
Table 16: Comparative blockchain-level performance of selected credentialing solutions 94
Table 17: RBAC implementations in smart contracts 99
Table 18: Gas saved using gas optimization techniques 101

BUPT

List of abbreviations

MIT	Massachusetts Institute of Technology
dApp	Decentralized application
BTC	Bitcoin
CPU	Processor
GPU	Graphics processing unit
ASIC	Application-specific integrated circuit
FPGA	Field-programmable gate array
zk	zero-knowledge
zk-SNARK	zero-knowledge Succinct Non-Interactive Argument of Knowledge
IoT	Internet of Things
IoMT	Internet of Medical Things
DeFi	Decentralized finance
NFT	Non-fungible token
USD	American dollar
DEX	Decentralized Exchanges
VPP	Virtual power plant
iOS	iPhone Operating System
GPS	Global Positioning System
AI	Artificial intelligence
EBSI	European Blockchain Services Infrastructure
DAAD	German Academic Exchange Service
IPFS	InterPlanetary File System
DLCC	Digital Learning Chain Structure
QR	Quick response
DSA	Digital Signature Algorithm
SHA-256	Secure Hash Algorithm-256
ROCA	Return of the Coppersmith Attack
PDF	Portable Document Format
JSON	JavaScript Object Notation
LTS	Long-Term Support
npm	Node Package Manager
EVM	Ethereum Virtual Machine
CID	Content identifier
RPC	Remote Procedure Call

BUPT

API	Application Programming Interface
IDE	Integrated Development Environment
HTML	HyperText Markup Language
CSS	Cascading Style Sheets
RBAC	Role-based access control
RBAC-SC	Role-based access control – Smart Contract
SLOAD	Storage Load
MLOAD	Memory Load
SSTORE	Storage store
MSTORE	Memory store
ABI	Application Binary Interface
UI	User interface

BUPT

1. Introduction

1.1. Thesis context

The advancement of society is closely linked to its capacity for continuous development, a concept that is particularly relevant in the context of education, where ongoing improvements are necessary to respond to the shifting demands of learners and societal transformation. To keep pace with the evolving needs of today's world, it is essential to develop school curricula that reflect current realities [1], continuously professional upskilling of teachers with updated knowledge and teaching strategies [2], and integrate modern educational resources that increase students' acquisition and understanding of knowledge [3].

Consequently, an essential process both for ensuring a quality educational act and for teacher's professional development, the teaching career requires continuous improvement and constant evolution of the teacher. In Romania, regardless of whether this training is determined by the annual evaluation [4], by the legal obligation to accumulate minimum 90 transferable professional credits [5] over five years or is the result of a personal initiative that could be acknowledged through a merit-based distinction [6], each teacher participates in professional training courses, conducted online or face-to-face, is actively involved in methodological committees or school projects, participates in symposia and conferences, as well as writes and publishes books or study articles, or creates other educational resources. All these activities reflect the teacher's professional evolution over time. Usually, these professional development activities are formally recognized through diplomas, certificates, or attestations that confirm a teacher's participation and contributions, serving not only as proof of involvement but also as valuable components of a professional portfolio [7]. This portfolio becomes a reflection of the teacher's lifelong learning journey, showcasing their commitment to continuous improvement, adaptability to new educational trends, and active engagement in the academic community [8].

Therefore, the numerous diplomas, certificates, or attestations obtained by teachers throughout their careers give rise to a significant challenge: the need for efficient indexing and organization. As these documents accumulate over time, managing them in a coherent, accessible, and verifiable manner becomes increasingly complex. Without a standardized system for cataloging and storing these credentials, it can be difficult for both teachers and educational institutions to track professional progress, validate qualifications, or present a clear overview of a teacher's development. This issue highlights the necessity for digital solutions or centralized platforms that can securely store and systematically index such records, ensuring transparency, ease of access, and long-term preservation. Blockchain technology presents a promising solution for systematically storing and indexing records such as educational diplomas and certificates [9], ensuring their authenticity and protecting against fraud. This innovative approach was pioneered at the Massachusetts Institute of Technology (MIT), where Philipp Schmidt, director of learning innovation at the MIT Media Lab, recognized the growing need to reliably track and manage a wide range of credentials, from formal education to certificates that recognize the skills, achievements, and experiences accumulated throughout a person's life [9].

BUPT

In addition to the continuous professional development of teachers, achieving quality education also requires the creation of effective educational resources that help students understand new knowledge more easily [10]. These educational resources are predominantly developed by teachers themselves, whose expertise and practical experience are invaluable in ensuring that the materials are both pedagogically relevant and practically effective, and therefore, it is entirely appropriate that high-quality, innovative, or widely adopted resources be recognized and included in the teacher's portfolio as evidence of merit when applying for a merit-based distinction [6]. As a significant proportion of educational resources are disseminated through diverse online platforms [3], the increasing digital migration heightens concerns over maintaining their quality and integrity. The convenience of online sharing often comes with risks such as unauthorized alterations, inaccurate content dissemination, or misuse that undermines the original intent and educational value. Therefore, it is important to establish protections for the intellectual property of teachers and the integrity of their work. A blockchain-based system not only protects the rights of educators but also guarantees that students and fellow teachers can rely on the accuracy and credibility of the resources they use [10]. A thoughtful balance of accessibility, quality control, and copyright protection will be essential in fostering a reliable ecosystem of educational materials that support effective teaching and learning in the digital age.

1.2. Motivation

Research on the application of blockchain in education has been conducted in universities worldwide with a particular focus on issuing, managing, and verifying academic degrees and transcripts of graduation [11, 12, 13, 14, 15] to prevent fraud and to ensure the authenticity of academic credentials. Beyond simply preventing diplomas fraud, in some studies [16, 17], blockchain is seen as a transformative tool for ensuring the integrity of the entire certification process, from student admissions and examinations to the final awarding of diploma. Moreover, its decentralized structure enables efficient management of transferable credits [18] and simplifies the verification of academic achievements between universities and across countries, greatly facilitating student mobility [19, 20].

But currently, there is a lack of in-depth research on the implementation of blockchain technology in pre-university education, particularly in Romania, and especially in relation to the potential benefits it could offer to teachers' evaluation, where we can find some persistance issues.

One major issue in teacher evaluation systems is the heavy reliance on physical documents, which teachers must submit repeatedly for various assessment stages based on specific criteria [6, 21]. Each evaluation requires its own copy of the documents, often reviewed by different committees, making the verification process complex and labor-intensive. These committees face the demanding task of thoroughly checking numerous diplomas, certificates, and attestations to confirm their authenticity and ensure that evaluations are fair and accurate. A significant volume of documentation in teacher evaluations arises not only from formal diplomas or certificates earned through university programs, professional training, or micro-credentials, but also from numerous attestations issued at the school or inspectorate level for a wide range of extracurricular activities, such as coordinating projects, mentoring students,

BUPT

organizing competitions, or working with specific student groups. Given that merit-based recognitions frequently come with significant financial rewards, this verification process demands exceptional rigor and precision to uphold the integrity of the evaluation system.

Another significant concern arises when a teacher does not obtain a diploma, certificate, or attestation at the time of the activity or later misplaces the document and requires a duplicate, especially in situations where the issuing institution no longer exists, the teacher has moved to a different school, or the original school has been closed, which is an increasingly common occurrence in rural areas affected by declining student populations and school consolidations. In such cases, the teacher loses access to official proof of their contribution, despite having carried out the work. This administrative gap can negatively impact their future evaluations. A long-term, digital solution is needed, that can securely store and validate such documents so that they remain accessible and verifiable years later, regardless of changes in the teacher's place of employment or the status of the issuing institution.

An important observation that can be made from the evaluation procedures is that each stage of assessment is tied to a specific time frame for the activities being evaluated. More precisely, evaluations are based on one year for annual performance reviews, and five years when considering professional development or merit-based recognition of teachers. Since these evaluations currently rely heavily on printed documents, there is a significant need for heightened attention to the authenticity of certificates and the exact period in which they were obtained. Any uncertainty or error regarding the issuance date of a diploma, certificate, or attestation can affect the fairness and accuracy of the evaluation process, making it essential to find more secure and verifiable methods of storing and accessing such data. One of the key benefits of blockchain technology lies in its ability to secure and timestamp uploaded data, ensuring both its integrity and the precise moment it was recorded. Once a document is registered on the blockchain, its data and timestamp cannot be altered or deleted [10], providing a permanent, transparent record of its origin. Furthermore, if a transaction is entered incorrectly, it cannot be simply edited, instead, a new transaction must be recorded to override the previous one [22], maintaining a complete history.

Another challenge lies in the fact that some teachers, driven by the pursuit of professional advantages or financial incentives, may attempt to obtain falsified attestations for activities they did not actually complete, undermining the fairness and credibility of the evaluation process.

Considering the numerous evaluations that a teacher undergoes, given that all these evaluations directly impact a teacher's career advancement and remuneration, and acknowledging that a well-trained teacher is fundamental to a high-quality educational system, we considered it necessary, in this doctoral research, to raise awareness of using a blockchain-based system that would bring together all the locally issued or training documents arising from a teacher’s professional activities and facilitate the digitalization of their evaluation process.

1.3. Thesis objective

The research activity that is the subject of this doctoral thesis has as its main objective the establishment of a management framework, using blockchain technology, of all documents

BUPT

of teachers in pre-university education obtained following professional development or involvement in various activities throughout their career, with the aim of contributing to the improvement of the evaluation process of teachers' activities.

Given the main objective mentioned above, several specific objectives have been formulated, as follows:

- The objective O1 is to analyze blockchain technology, both from a theoretical point of view and from the point of view of research conducted in this field.
- The objective O2 is to examine the impact and practical implications of blockchain technology across various fields.
- The objective O3 is to examine current research on managing and verifying academic credentials, including traditional methods, digital signatures, quick response (QR) codes, and blockchain-based solutions.
- The objective O4 aims to identify and analyze the current challenges and inefficiencies in the management and verification of diplomas, certificates, and attestations awarded to pre-university teachers throughout their professional trajectories, with the aim of contributing to a more reliable evaluation framework.
- The objective O5 is to design a blockchain-based framework tailored to the specific requirements of the educational system, ensuring secure, tamper-proof, and time-stamped recording of all documents related to teachers' professional development and school-based activities.
- The objective O6 is to evaluate the feasibility, scalability, and practical implementation of the proposed blockchain model through simulations and case studies based on real-life scenarios from the Romanian pre-university education system.
- The objective of O7 is to analyze the potential impact of the blockchain-based system on the efficiency of the teacher evaluation process compared to the current evaluation method.
- The objective O8 is to evaluate security mechanisms within the proposed blockchain-based framework, ensuring data integrity, access control, and resistance to unauthorized modifications.

1.4. The structure and content of the thesis

To achieve the objectives mentioned above, the thesis is structured in 8 chapters briefly presented below.

Chapter 1 introduces the thesis by presenting the research context and motivation, outlining the objectives, and providing a detailed overview of the thesis structure.

Chapter 2 focuses on Objective O1, presenting a comprehensive analysis of the components and security aspects of blockchain technology by outlining its fundamental principles, tracing the development of distributed ledger systems, and evaluating decentralized solutions alongside relevant security protocols. Chapter 2 also addresses Objective O2 by

BUPT

presenting a synthesis of existing research highlighting the impact and applications of blockchain across various fields, with a special emphasis on its emerging role in education.

Chapter 3 addresses Objective O3 by examining current research on the management and verification of academic credentials. The chapter provides an overview of traditional methods for issuing and authenticating credentials, explores the integration of digital signatures and QR codes for verification purposes, and reviews emerging blockchain-based solutions. This analysis highlights existing gaps and identifies opportunities for enhancing the reliability, security, and efficiency of credential management systems.

Chapter 4, Section 4.1., corresponds to Objective O4 and is dedicated to identifying and analyzing the practical challenges associated with teacher evaluation through various portfolios which requires rigorous verification of diplomas, certificates, and attestations obtained as a result of the advanced training courses followed or the activities in which they were involved. This chapter highlights issues such as document loss, fraudulent certification, institutional discontinuity, and inefficiencies in manual verification processes.

Chapter 4, Sections 4.2.-4.4., are aligned with Objective O5 and introduces the proposed blockchain-based framework. It outlines the system architecture, key functionalities, and how the solution is tailored to support the secure, tamper-proof, and time-stamped registration of teachers' documents related to their professional development and school-based activities.

Chapter 5, Section 5.1., addresses Objective O6 by presenting simulations that demonstrate the feasibility and scalability of the proposed blockchain solution.

Chapter 5, Section 5.2. is dedicated to Objective O7 and provides an in-depth discussion on the anticipated impact of implementing the blockchain framework in teacher evaluation processes. The chapter examines improvements in transparency, accuracy, and efficiency, and considers the long-term benefits of ensuring trust and integrity in educational credentialing and professional recognition.

Chapter 6 addresses Objective O8 by presenting an analysis of the security mechanisms embedded in the smart contracts that form the foundational layer of our blockchain-based system. The chapter details the strategies employed to ensure data integrity, access control, and resistance to common vulnerabilities.

Chapter 7 summarizes the conclusions of the research, outlines the original contributions of the thesis, discusses the dissemination of results, and proposes directions for future research aimed at further integrating blockchain technology into the educational landscape.

BUPT

2. State of the art

This chapter provides a theoretical foundation for the design and implementation of blockchain technology in education. It introduces key concepts and terminology, outlines the necessary conditions for integrating blockchain into educational systems, and explains its roles and benefits.

2.1. Blockchain technology

2.1.1. Blockchain technology overview

Blockchain is a distributed database, with a "peer–to–peer" architecture, implemented as a simply linked list, in which the nodes of the list, called blocks, are linked to each other by hash [23].

Figure 1 illustrates the blockchain as a continuous sequence of interconnected blocks. Blocks are files in which data related to transactions in a network is permanently recorded. The first block in the list is called the genesis block . Other blocks in the network are added to this block. Each block must contain transaction data, the hash of the previous block, the hash of the current block, a timestamp, and a nonce [24, 25].

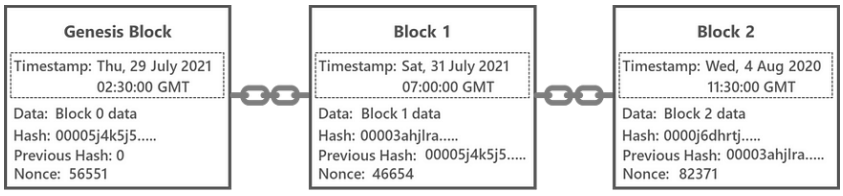


Figure 1: Blockchain as a data structure

Being a technology with multiple benefits for developing applications, a closer examination allows us to outline the following advantages:

- Being a distributed database, data in the form of transactions is not stored on a central server that can be accessed by all users, but copies of them are stored locally, at the same time, on several computers of users around the world, computers that support the functions of the blockchain and that have installed specialized software.
- Blockchain ensures data consistency and transparency across the network, as every time a block is added, all copies of the chain on network nodes are updated. Each node maintains an identical transaction history, ensuring that all participants share a synchronized and validated record [23].
- Blockchain's peer-to-peer architecture ensures that all participants maintain data integrity collaboratively [26], eliminating central control and enhancing security, resilience, and trust across the network.

Blockchain's adoption in solution development necessitates a balanced perspective, as the technology encompasses not only strengths but also limitations, the most important of which are discussed below:

BUPT

- High energy consumption: Blockchain networks, especially those using proof-of-work consensus, require substantial computational power to maintain and validate the ledger, leading to increased operational costs and environmental impact.
- Scalability challenges: As the number of users and transactions increases, blockchain networks may experience reduced performance, with slower transaction processing times and higher resource requirements, making large-scale adoption more complex [27].
- Limited interoperability: Differences in protocols and architectures across blockchain networks hinder seamless communication and the efficient exchange of data between systems [27].

As many solutions are developed on blockchain, it is important to carefully consider the type of blockchain employed. The technology has evolved and diversified, resulting four principal categories, which are distinguished by block creation permissions and data access rights:

- A public blockchain is an open, decentralized network that allows any computer worldwide, once equipped with the necessary software, to view transaction records and to participate at the consensus procedure [28]. Popular examples include Bitcoin and Ethereum [28], which are the pioneers of public blockchains.
- A private blockchain is a controlled network in which block creation and access are restricted to a select group and is typically governed by a single organization. Hyperledger is one of the most well-known examples of a private blockchain, designed for use in business environments where privacy and permissioned access to data are essential [29].
- A hybrid blockchain combines features of public and private blockchains [30], allowing transaction data to be publicly visible while restricting block creation to authorized participants. One example of a hybrid blockchain is Dragonchain [31].
- A consortium blockchain is governed collectively by a group of organizations, allowing multiple parties to jointly manage the network [32]. One example of a consortium blockchain is Corda, but Ethereum is frequently used in consortium blockchain applications [33].

Table 1 presents a comparative overview of the main types of blockchain, public, consortium, private, and hybrid, highlighting their characteristics in terms of consensus mechanisms, data accessibility, control, efficiency, and level of centralization.

Characteristic	***Public Blockchain***	***Consortium Blockchain***	***Private Blockchain***	***Hybrid Blockchain***
Consensus determination	All participants	Group of nodes	A single organization	Combination of public and private participants

BUPT

Consensus process	Permissionless	Permissioned	Permissioned	Mix of permissioned and permissionless
Read Permission	Public	Public or restricted	Restricted	Public or restricted
Data control and integrity	Highly secure and resistant to tampering	Moderately secure, possible internal alterations	Controlled by one entity, data can be altered	Varies depending on architecture
Efficiency	Low	High	High	Moderate
Centralization	No	Partial	Yes	Partial

Table 1: Comparative analysis of the main types of blockchain architectures

2.1.2. Consensus mechanisms

One of the most important part of a blockchain is the consensus mechanism which is a protocol that enables participants in a decentralized blockchain network to agree on the validity of transactions without relying on a central authority [34]. Consensus mechanisms solve the problem of trust and disagreement between participants in a distributed network. They ensure that no single entity or group of entities can control the system, preventing fraudulent transactions or double-spending. These protocols govern how transactions are validated, how blocks are added to the blockchain, and how disagreements are resolved among nodes. Each consensus mechanism has its own strengths and weaknesses, and the choice of mechanism depends on the specific needs of the blockchain system it supports, such as speed, security, and scalability.

Blockchain technology has given rise to hundreds of consensus mechanisms, each designed to address specific challenges in decentralized networks. The most important consensus mechanisms [35, 36] are presented in Table 2.

Consensus Name	***Description***	***Examples of blockchains that use it***
Proof-of-work	In proof-of-work, miners compete to solve complex mathematical puzzles, and the first to succeed earns the right to add a new block to the blockchain, receiving cryptocurrency as a reward.	Bitcoin
Proof-of-stake	In proof-of-stake, participants commit some of their cryptocurrency as a stake, and the more they stake, the higher their chances of being chosen to create the next block and earn rewards.	Ethereum
Pure-proof-of-stake	Pure proof-of-stake represents a refinement of the traditional proof-of-stake mechanism, wherein the probability of generating a new block is determined exclusively by the quantity of stake held by a participant, thereby removing the reliance on mining activities.	Algorand
Secure-proof-of-stake	Secure-proof-of-stake is an optimized version of proof-of-stake designed to strengthen security by mitigating potential attacks while also improving scalability.	MultiversX
Proof-of-history	Proof-of-history is a consensus mechanism that generates a verifiable timeline of events, demonstrating that a specific	Solana

BUPT

	action occurred at an exact point in time, thereby reducing the computational burden of achieving consensus.	
Practical Byzantine Fault Tolerance	Practical Byzantine Fault Tolerance guarantees the validity of a transaction when a supermajority of nodes reach agreement, maintaining reliability even in the presence of faulty or malicious participants.	Hyperledger Fabric
Ouroboros	Ouroboros is a proof-of-stake consensus protocol designed to secure the blockchain by randomly selecting validators in proportion to their stake.	Cardano

Table 2: Types of consensus mechanisms

Consensus protocols display notable differences in efficiency, each demonstrating distinct performance characteristics [35], as shown in Table 3.

Consensus Name	***Examples of blockchains that use it***	***Transactions per second***	***Block creation speed***
Proof-of-work	Bitcoin	approx. 7	approx. 10 minutes
Proof-of-stake	Ethereum	>30	approx. 12 seconds
Pure-proof-of-stake	Algorand	>1000	<5 seconds
Secure-proof-of-stake	MultiversX	>15000	approx. 6 seconds
Proof-of-history	Solana	2000–65000	approx. 400 milliseconds
Practical Byzantine Fault Tolerance	Hyperledger Fabric	approx. 1000	1–2 seconds
Ouroboros	Cardano	>250	approx. 20 seconds

Table 3: Consensus mechanisms' characteristics

2.1.3. Blockchain's layers

After analyzing consensus mechanisms, it is also essential to examine the layers of blockchain, as this provides insight into how inherent limitations such as scalability and interoperability can be addressed. By mapping out the roles of each layer, we can better justify the architectural choices made to ensure that our system for managing documents attesting teachers' activities remains efficient and capable of handling high volumes of simultaneous transactions and interactions.

Blockchain architecture is structured into 4 distinct layers [37] as follows:

- *Layer -1: Hardware layer*

The foundational layer of blockchain comprises the hardware and infrastructure necessary for operation, including servers, storage, networking, power, and cooling systems. Resource requirements depend on the consensus mechanism, ranging from standard processors (CPUs) and graphics processing units (GPUs) [38] in energy-efficient protocols to specialized application-specific integrated circuits (ASICs) in proof-of-work networks [39], ensuring secure and reliable performance.

- *Layer 0: Network layer*

BUPT

Positioned above the physical infrastructure, Layer 0 consists of the peer-to-peer communication protocols that facilitate decentralized data exchange among participating nodes. This layer manages node discovery, data propagation, synchronization, and message routing, thereby ensuring network coherence without the need for centralized coordination. Layer 0 plays a critical role in maintaining decentralization and resilience, as it enables the dynamic interaction of distributed nodes. Additionally, interoperability protocols designed to link disparate blockchain systems often begin at this layer, laying the groundwork for cross-chain communication and data transfer.

- *Layer 1: Blockchain layer*

Layer 1 forms the core of the blockchain, where consensus algorithms ensure agreement on valid transactions, block validation rules confirm correctness and protocol compliance, and smart contracts execute deterministically based on validated blocks, maintaining network integrity and functionality.

- *Layer 2: Scalability layer*

Layer 2 enhances blockchain scalability by processing transactions off-chain or in parallel with the main chain. Key solutions include state channels for private multi-interactions, sidechains running parallel and periodically anchored to Layer 1, rollups that batch transactions with compressed proofs, and payment channels for rapid, low-cost transfers [37]. Although these Layer 2 solutions operate externally from Layer 1, they maintain strong cryptographic links and periodic commitments to the main blockchain, preserving data authenticity and transaction finality. This interconnected operation allows Layer 2 to effectively increase network capacity, reduce latency, and lower costs, making it essential for supporting high-volume, real-time blockchain applications.

Figure 2 provides a visual overview of the multi-layered blockchain architecture, illustrating the hierarchical structure and functional relationships between layers.

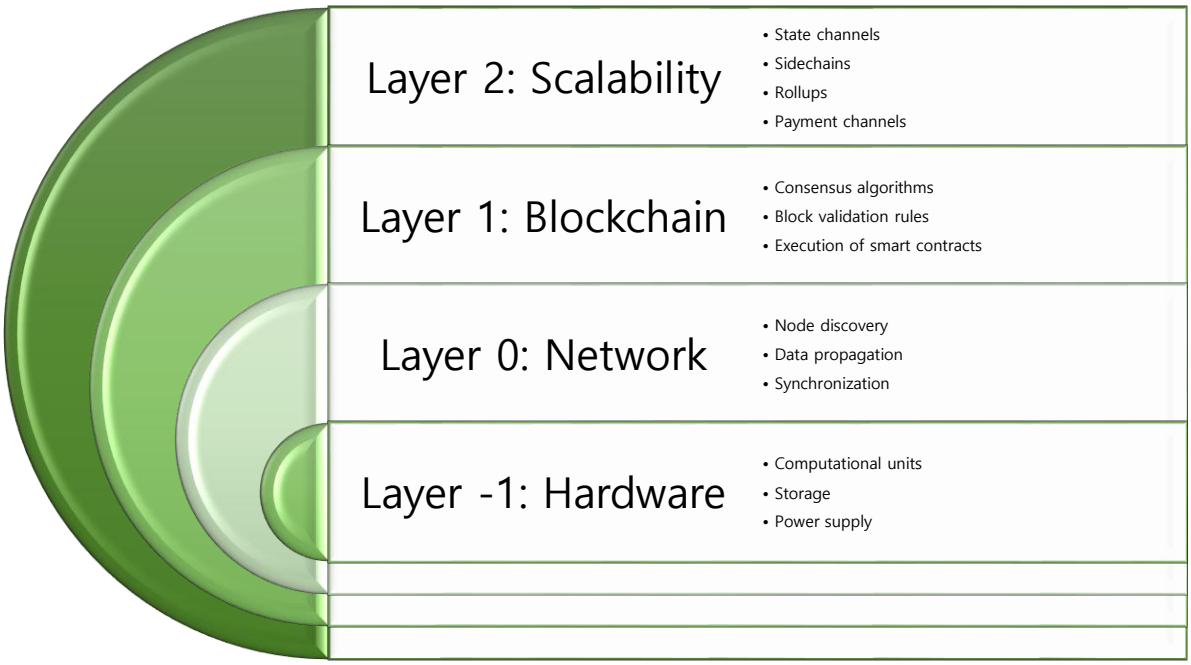


Figure 2: Blockchain's layers. Source: own illustration.

BUPT

2.2. Applications of blockchain

2.2.1. Blockchain platforms

When developing a solution, it is essential to evaluate blockchain platforms in relation to the specific needs of the application. For some adopters, the longevity and maturity of a platform serve as indicators of stability and trust, making them prefer technologies with a proven track record. Others prioritize scalability, choosing platforms capable of processing large volumes of transactions while maintaining efficiency. A different perspective emphasizes cost-effectiveness, focusing on predictable and low transaction fees that ensure economic sustainability. These diverse considerations highlight the need to balance technical and economic factors when selecting a platform. The following section examines the most widely recognized blockchain platforms that have demonstrated relevance and applicability across diverse contexts.

2.2.1.1. Bitcoin

Bitcoin is the first practical blockchain, offering a decentralized and trustless system where transactions are secured by cryptography, validated through consensus, and recorded immutably [40]. Miners maintain blockchain integrity by selecting and packaging transactions into blocks [41], which are added approximately every ten minutes after solving computationally intensive proof-of-work puzzles [42], ensuring a consistent and tamper-proof ledger. Miners receive rewards in the form of newly minted bitcoins and transaction fees [43].

The primary advantages of Bitcoin's blockchain include its decentralized validation, immutability, and resistance to fraud, which collectively create a robust foundation for digital exchange. However, these benefits come with significant limitations: proof-of-work mining demands substantial computational power, specialized hardware, and vast electricity consumption [44], making the network resource-intensive and raising concerns about scalability and environmental impact.

Bitcoin is most effective for securing high-value transactions, yet it proves less suitable for applications that require the rapid and continuous processing of large transaction volumes.

2.2.1.2. Ethereum

Ethereum is a blockchain and a development platform that allows developers to build and deploy decentralized applications (dApps) and smart contracts [45, 46]. Ethereum is widely known for its smart contracts, which can automate the transfer and verification of educational resources.

The Ethereum blockchain at its inception in 2015 was secured by a proof-of-work consensus mechanism. Ethereum switched from proof-of-work to proof-of-stake with the Ethereum 2.0 upgrade, which was completed in September 2022 [47]. In Ethereum's proof-of-stake mechanism, validators replace miners to propose and validate blocks [47]. Validators are selected to propose new blocks based on the size of their stake and a degree of randomness. To participate as a validator, a user must stake at least 32 Ether. The recent EIP-7251 upgrade, part

BUPT

of the Pectra update, raises the maximum validator stake to 2048 Ether [48]. This allows high-stake participants to consolidate validation, reducing the number of validators required while improving efficiency and maintaining security.

An important advantage of Ethereum is that its proof-of-stake mechanism eliminates the need for energy-intensive mining, making the network far more energy-efficient.

One of Ethereum's most significant capabilities lies in its support for smart contracts. In contrast to Bitcoin, which is primarily designed for peer-to-peer value transfers, Ethereum provides a platform for the development of self-executing contracts. A smart contract is a program stored on a blockchain that automatically executes the rules and terms of an agreement, ensuring trust and transparency without relying on intermediaries [49]. Ethereum's smart contracts are developed using the Solidity programming language, which features a syntax and structure closely resembling JavaScript. The Ethereum Virtual Machine (EVM) facilitates the execution of smart contracts by providing a runtime environment in which all network nodes can independently process contract operations without depending on the underlying protocol layer.

Figure 3 illustrates the structured life cycle of a smart contract, encompassing three core stages: creation, deployment, and execution [50].

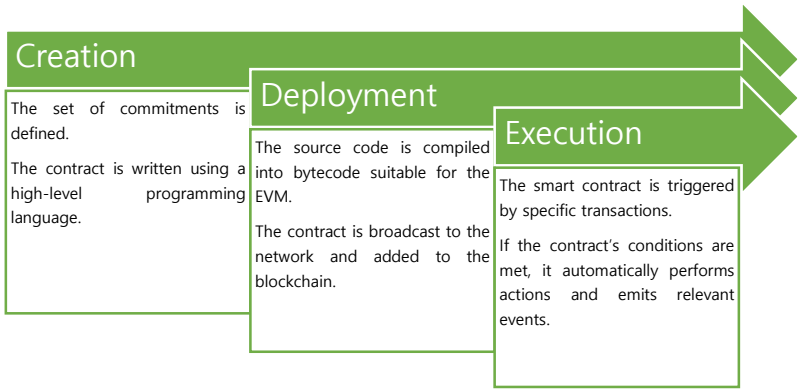


Figure 3: The sequential life cycle that governs a smart contract from creation to deployment and execution. Source: own illustration.

One major advantage of Ethereum is its ongoing development, which specifically targets limitations in scalability and interoperability. This focus has led to the implementation of Layer 2 solutions and the development of complementary blockchains, significantly improving the platform's efficiency and ability to support a growing ecosystem of applications. Ethereum proposes both Layer 1 and Layer 2 solutions to improve scalability, reduce congestion, and lower transaction fees:

- *Layer 1 solution*:
  - *Sharding*: Sharding is a scalability enhancement that divides the Ethereum network into smaller, independent units called shards [46]. Each shard operates as a distinct blockchain with its own validators, transactions, and smart contracts. This enables parallel transaction processing, increasing network throughput.

BUPT

- *Layer 2 solutions*: Layer 2 solutions reduce the load on Ethereum's main blockchain by handling transactions off-chain or on a secondary layer while storing summary data on Layer 1. Key Layer 2 approaches include:
    - *State Channels* establish off-chain channels that allow multiple sequential interactions between participants, with only the final settlement recorded on Ethereum [35].
    - *Sidechains* are independent Ethereum-compatible blockchains with their own consensus mechanisms [35].
    - *Plasma* introduces child chains anchored to Ethereum that process large volumes of transactions off-chain [35].
    - *Rollups* execute transactions off-chain while posting concise summaries or proofs to Ethereum.
        - *Optimistic Rollups* operate under the assumption that transactions are valid unless challenged [51].
        - *Zero-Knowledge (ZK) Rollups* use cryptographic proofs to validate off-chain transactions on-chain [52].

A major limitation of Ethereum is the unpredictability of transaction fees, which can affect both cost management and operational planning. Fees rise and fall with network demand, making it challenging for users to anticipate expenses. High congestion periods can lead to delays and increased costs, creating obstacles for applications that require consistent, timely, and cost-effective transaction processing.

2.2.1.3. Hyperledger

When developing systems that require stringent data privacy, Hyperledger is often the preferred framework. As a permissioned blockchain, Hyperledger operates within a controlled environment where only authorized participants can access and interact with the network [53]. It has become a leading solution for applications requiring secure, private, and auditable processes, offering multiple frameworks tailored to specific use cases. These include [54, 55]:

- Iroha, designed for lightweight mobile and embedded applications.
- Sawtooth, focused on scalability and modular architecture.
- Fabric, widely used for business process management due to its flexibility and support for chaincode, Hyperledger's implementation of programmable logic similar to smart contracts.
- Indy, dedicated to identity management and decentralized identity systems.
- Burrow, a blockchain client optimized for executing Ethereum-compatible smart contracts.

An advantage of Hyperledger is its ability to provide a highly secure and permissioned environment, allowing organizations to control access to sensitive data while maintaining transparency and auditability within the network.

A disadvantage of Hyperledger is that its reliance on a permissioned architecture can reduce decentralization, potentially limiting trust assumptions compared to fully public blockchains.

BUPT

2.2.2. The state of research in blockchain application areas

Blockchain technology continues to attract attention across numerous sectors [56, 57]. Although many projects are still in pilot or experimental stages, there is a clear trend of organizations exploring how blockchain can address real-world challenges. It is important to assess whether blockchain is truly necessary for a given problem, as some goals can sometimes be achieved more efficiently with conventional methods. Nonetheless, certain areas benefit uniquely from blockchain's features, highlighting the value of examining current research and practical implementations to understand its potential and limitations. This section presents the foremost software solutions that utilize blockchain technology for the design and development of decentralized applications, with a particular focus on its transformative role in key sectors such as finance, healthcare, Internet of Things (IoT), and public administration.

2.2.2.1. Finance

Bitcoin's launch in 2009 marked a turning point in the digital landscape, introducing blockchain technology as a secure way to record transactions without the need for central oversight. While Bitcoin proved the value of decentralized currency, it was Ethereum that truly unlocked the broader potential of blockchain. By introducing smart contracts, Ethereum paved the way for a new generation of decentralized applications. When these applications converged with the world of finance, a transformative movement emerged, decentralized finance (DeFi) [58]. This fusion reshaped the foundations of financial interaction, redefining how people interact with money by removing intermediaries and giving users direct control over financial services such as lending, borrowing, and trading [59]. Alamsyah et al. [60] and Schueffel [61] outline the following areas of interest for DeFi:

- *Payments*: DeFi has revolutionized digital payments by enabling decentralized, programmable, and real-time financial interactions. While Bitcoin marked the first major use case of decentralized payments, the advent of Ethereum and smart contracts expanded possibilities, giving rise to innovative applications like Sablier [62], a protocol for continuous, real-time payments used in payroll and micro-consulting, or Connext [63], a Layer 2 network that facilitates instant, low-cost transactions ideal for streaming payments and micropayments.
- *Lending and borrowing*: DeFi lending allows users to lend or borrow crypto assets through smart contracts, removing intermediaries and promoting open access to financial services [64]. Users earn passive income by providing liquidity to lending pools or directly engaging in peer-to-peer loans. Platforms like Aave, Maker, and Compound automate collateralized loans and interest generation [65], making borrowing and lending more efficient and inclusive compared to traditional systems.
- *Asset management*: DeFi asset management uses smart contracts to pool and allocate funds across dApps, reducing reliance on centralized custodians [66]. It includes both fungible and non-fungible tokens.
- *Insurance*: DeFi insurance replicates traditional models by providing automated coverage against risks like hacks, smart contract failures, and market events.

BUPT

Smart contracts handle policy issuance and payouts, improving speed, affordability, and transparency.

- *Stablecoins:* Stablecoins are digital assets pegged to stable values, typically USD [67], minimizing volatility and enabling reliable transactions in DeFi. These crypto assets maintain price stability, enabling reliable trading, lending, and payments in DeFi, and reducing exposure to market volatility through fiat- or crypto-collateralized mechanisms.
- *Staking:* Staking allows users to earn passive income by locking tokens into smart contracts to support proof-of-stake networks. It enhances blockchain security while offering returns without the energy-intensive demands of mining. Platforms like AQRU and Cardano offer diverse staking options. AQRU offers a seamless trading experience with features like flexible order types, access to margin trading, and low, competitive fees [68]. On the other hand, Cardano introduces a user-friendly staking model through delegation, allowing ADA holders to support network operations by assigning their tokens to staking pools while earning passive rewards [69].
- *Yield farming*: Yield farming involves locking tokens into DeFi protocols to generate high returns through interest, fees, or token incentives [70]. Strategies include lending assets or providing liquidity to decentralized exchanges. Smart contracts automate this process, offering users potential gains far exceeding those from traditional finance.
- *Decentralized exchanges* (DEXs): DEXs use smart contracts and automated market makers to enable peer-to-peer token trading [65, 71, 72] with enhanced privacy, user control, and liquidity, without centralized intermediaries.

DeFi has been successfully applied across a wide spectrum of financial domains as illustrated in Figure 4, serving as a transparent, trustless framework that eliminates intermediaries, enhances accessibility, and enables programmable financial services through smart contracts.

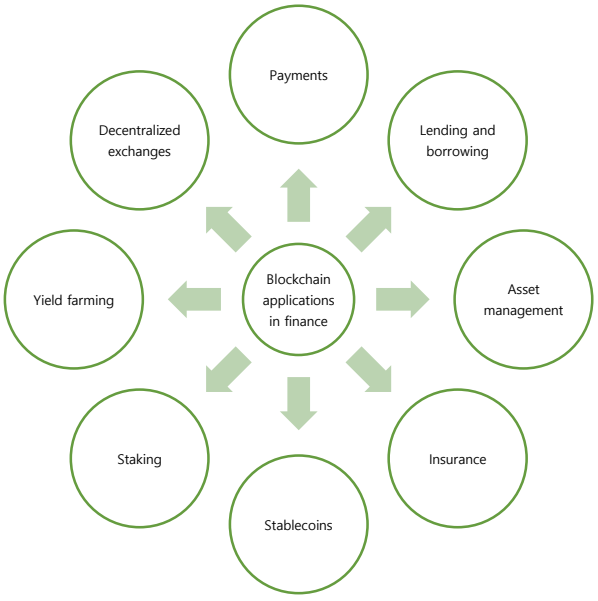


Figure 4: DeFi domains. Source: own illustration based on the results from [60].

BUPT

2.2.2.2. Healthcare

After finance, a suitable field for blockchain has proven to be healthcare, where the need for patient information confidentiality can benefit from the advantages of blockchain as a decentralized, secure, and transparent system for managing sensitive medical data. Shaikh et al. [73] identify the following areas where blockchain technology is applied in healthcare, with each addressing distinct challenges or improving essential processes:

- *Medical records*: Blockchain enables secure, patient-controlled management of electronic, medical, and personal health records by reducing data fragmentation and preventing unauthorized access while ensuring data integrity and interoperability through controlled access [74, 75].
- *Patient data sharing*: Blockchain facilitates dynamic, consent-driven sharing of patient data using smart contracts and encryption. It empowers individuals to grant specific access to providers or researchers, enhancing trust and privacy. This approach replaces inefficient, siloed data exchange systems.
- *Pharmaceutical or drug supply chains*: Blockchain ensures transparent, tamper-proof tracking of drugs from production to distribution, combating counterfeit medications. It allows stakeholders to verify drug authenticity in real time. Compliance with regulatory standards is improved through immutable audit trails.
- *Medical imaging*: Blockchain supports secure, decentralized storage of large imaging files through integration with off-chain systems like IPFS. It ensures that only authorized users can access sensitive imaging data. This reduces storage costs and enhances patient data privacy.
- *Health information exchange*: Blockchain improves interoperability across healthcare systems by standardizing and securing data exchange. It reduces duplication and enhances system scalability while giving patients control over who accesses their information. Smart contracts automate consent and access management.
- *Genomic data management*: Systems utilizing NFTs for genomic data sharing enable secure and privacy-focused collaboration in the study of genetics and disease mutations. Blockchain-based auditing mechanisms create permanent, tamper-proof records that allow researchers to access and analyze genomic information confidently, ensuring adherence to data ownership and privacy standards.
- *Internet of Medical Things (IoMT) data privacy*: IoMT devices and remote networks, widely adopted for real-time patient monitoring, produce substantial volumes of health data, primarily used by providers for remote care, raising concerns regarding data privacy and security. Blockchain-based IoMT enabled electronic health record systems have demonstrated effective real-time data protection, achieving low latency and high throughput, positioning blockchain as a promising solution for data-intensive healthcare environments.

BUPT

- *Clinical trials research*: Blockchain enhances clinical trials by providing immutable, transparent records of patient consent and trial data [76].

Figure 5 highlights the main domains in healthcare where blockchain technology is driving significant transformation.

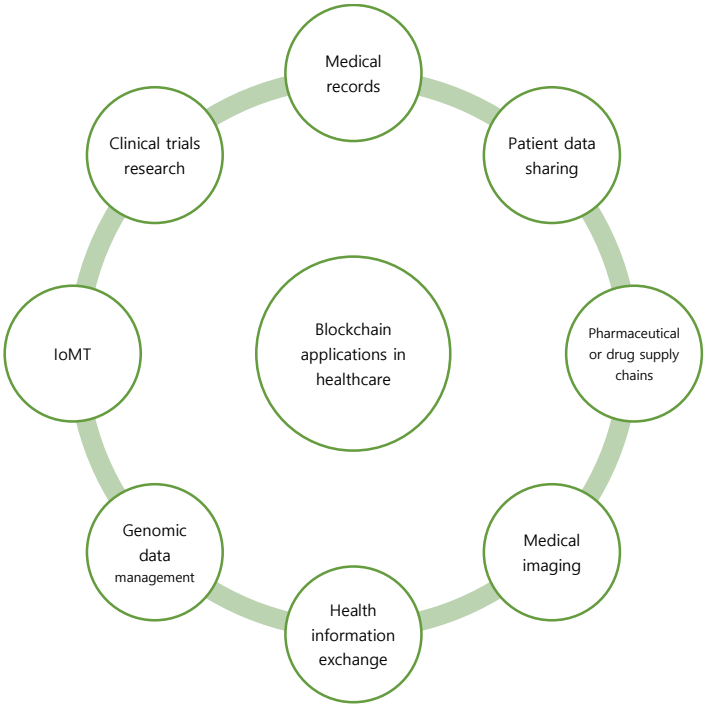


Figure 5: Areas of blockchain research in healthcare. Source: own illustration based on the results from [73].

2.2.2.3. Internet of things

As IoT devices proliferate, the security risks of centralized data storage have prompted extensive research on the benefits of using the decentralized blockchain ledger to protect data integrity and privacy.

In [77, 78, 79] is presented an in-depth exploration of how blockchain technology can be effectively integrated with various IoT domains to address persistent challenges related to data security, privacy, and trust in decentralized environments. With the rapid proliferation of connected devices, IoT systems are increasingly vulnerable to cyberattacks and data breaches [80]. Blockchain's decentralized architecture, immutability, and support for smart contracts enable secure, transparent, and autonomous data exchange across distributed networks. The following domains illustrate how blockchain enhances trust and operational integrity within IoT-enabled infrastructures:

- *Smart Healthcare*: Blockchain ensures the secure and immutable recording of patient health data captured by IoT devices, such as wearables and remote monitoring systems. Through smart contracts, it automates access control, enabling real-time yet privacy-preserving data sharing across telemedicine platforms and hospital information systems.
- *Smart Manufacturing*: By embedding blockchain within industrial IoT, production machines and robots gain a trusted ledger for operational logs, predictive maintenance data, and process parameters [81].

BUPT

- *Smart Cities*: Blockchain strengthens data integrity and security across smart city infrastructures, supporting decentralized governance models, tamper-proof logging of citizen interactions, and efficient management of urban resources.
- *Smart Vehicles*: In IoT-enabled vehicular networks, blockchain provides trusted vehicle-to-vehicle and vehicle-to-infrastructure communication, enabling secure data sharing for traffic management, accident response, and autonomous driving.
- *Smart Grids*: Blockchain facilitates decentralized energy transactions in smart grid systems [82], offering transparency, resilience against cyber threats, and automated contract enforcement in peer-to-peer energy trading environments.
- *Smart Agriculture*: Blockchain enables secure traceability and monitoring of agricultural data, improving transparency in supply chains and safeguarding IoT sensor data related to crop health, pesticide use, and environmental conditions.

To conclude, Figure 6 demonstrates how blockchain has found widespread application in diverse IoT sectors, serving as a secure, decentralized backbone for data integrity, trust management, and autonomous coordination across interconnected devices.

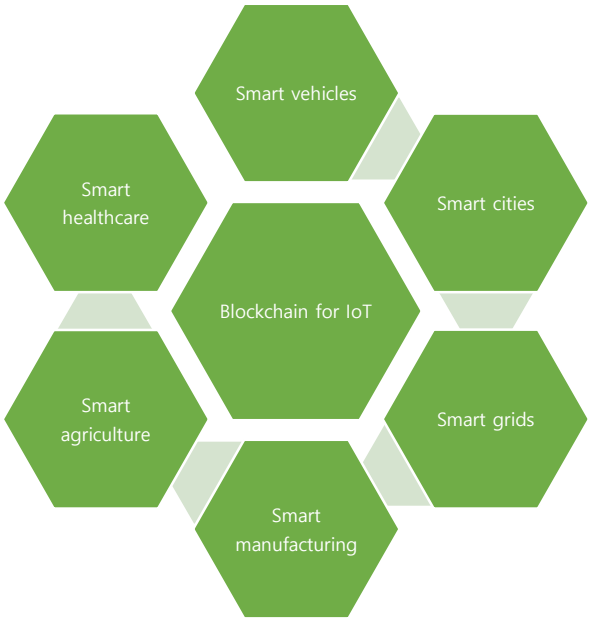


Figure 6: Areas of blockchain research in IoT. Source: own illustration based on the results from [77]

2.2.2.4. Public administration

In [83, 84] are explored a wide array of public administration domains where blockchain technology is being piloted or proposed to enhance transparency, efficiency, and trust. Each domain presents unique challenges that blockchain's core features can potentially address:

- *E-Government and voting*: Blockchain can modernize e-government services by enhancing process transparency, improving public trust, and reducing bureaucratic inefficiencies. In voting systems, it provides tamper-proof vote storage, voter authentication, and integrity of results.

BUPT

- *Social welfare and protection*: Blockchain can streamline and secure government-to-citizen transfers, particularly in welfare, healthcare, and unemployment services, by reducing intermediaries and ensuring the traceability and legitimacy of benefits.
- *Education and training*: Academic credentials, certifications, and training records can be stored securely on blockchain platforms, allowing instant verification and cross-border recognition, with smart contracts and Metaverse technologies enabling immersive, scalable education solutions.
- *Digital identity management*: Decentralized digital identity systems built on blockchain can empower citizens to control their personal data, improve service access, and reduce the cost and complexity of identity verification across public services.
- *Public procurement*: Blockchain can ensure fairness, auditability, and cost-efficiency in procurement by automating bid evaluation, tracking contract compliance, and minimizing corruption risks through transparent, immutable ledgers.
- *Public records management*: Civil and asset registries can benefit from blockchain [85] by consolidating fragmented databases, preventing unauthorized changes, and providing accessible, tamper-evident documentation for births, deaths, land ownership, and more [86].
- *Certificate issuance*: Birth certificates, notarial records, and other official documents can be issued and stored securely using blockchain and decentralized file systems, ensuring authenticity, reducing fraud, and simplifying verification.
- *Tax collection*: Blockchain can support transparent, interoperable tax collection and fraud detection systems by facilitating secure inter-agency data sharing, real-time auditing, and more reliable taxpayer identity verification.

Figure 7 illustrates this key areas where blockchain technology is being explored and applied within public administration.

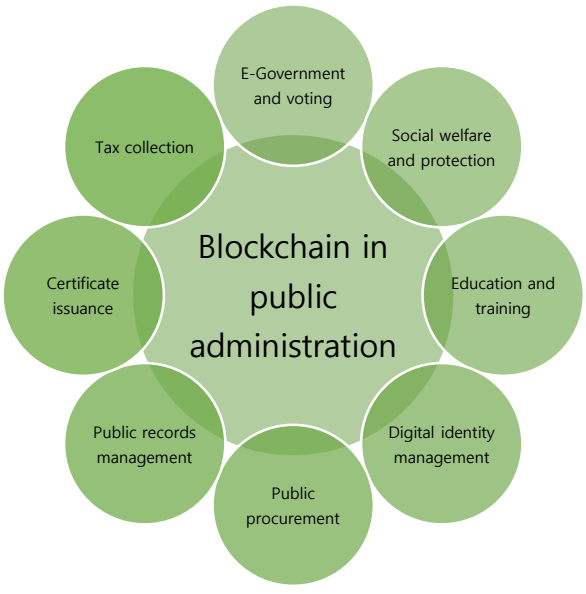


Figure 7: Areas of blockchain research in public administration. Source: own illustration based on the results from [83] .

BUPT

2.2.2.5. Education

Blockchain technology aligns seamlessly with the needs of the education domain especially considering the vast number of diplomas, certificates, and academic records that are often at risk of fraud. Blockchain provides the authenticity and security these documents need. In [87, 27] are highlighted the following domains within education where blockchain technology is currently being implemented or explored:

- *Issuing, managing and verifying academic certificates* is the central focus for articles who studies the topic of blockchain in education [87, 27].
- *Lifelong academic and professional achievement management*: As continuous learning becomes essential in response to technological evolution and labor market demands, traditional academic pathways are supplemented by micro-credentials and professional training [14]. Blockchain-based educational systems must address the challenge of reliably recording and verifying such decentralized learning achievements.
- *Management of the processes involved in the academic journey*: While a diploma marks the end of formal studies, it is the result of many key academic steps. Blockchain technology enables secure, transparent tracking and validation of the entire educational journey [16], offering verifiable records that go beyond the diploma itself.
- *Educational resources exchange*: Blockchain-based platforms can protect digital copyrights [88], and fostering a global community for trusted educational resource exchange [3].
- *Tokens transfer*: dApps can be used for payments of scholarships [89], tuition fees, and donations [90], allowing for secure and direct transactions.
- *Students evaluation*: dApps can automate and secure the entire student assessment process, providing real-time, trustworthy results.

Figure 8 showcases the integration of blockchain technology within diverse areas of education.

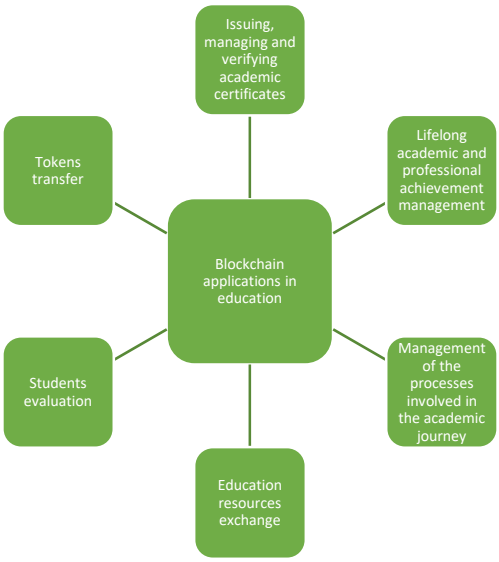


Figure 8: Areas of blockchain research in education. Source: own illustration based on the results from [87, 27]

BUPT

2.3. Summary

This chapter serves as an essential foundation for our research, providing a comprehensive exploration of the core concepts of blockchain technology and its various implementations. By analyzing different types of blockchains and examining the consensus mechanisms that underlie them, we understand how these elements shape the balance between security, decentralization, and operational efficiency. The layered architecture of blockchain systems is carefully detailed, revealing how each layer, from the physical infrastructure to advanced scalability solutions, plays a critical role in the overall performance and resilience of the system.

Following the presentation of the main blockchain platforms, it is clear that each platform embodies a distinct approach to blockchain implementation, shaped by its architecture, consensus mechanism, and scalability model. To synthesize these insights and facilitate a clear comparison, Table 4 outlines their core characteristics, offering a concise overview of the technical and operational differences most relevant to the context of this work.

Characteristic	***Bitcoin***	***Ethereum***	***Hyperledger Fabric***
Layer	Layer 1	Layer 1	Permissioned, modular framework
Transaction processing	On-chain	On-chain	On-chain
Consensus mechanism	Proof-of-work	Proof-of-stake	Pluggable (e.g.: RAFT, PBFT)
Transaction speed	Slow	Moderate	High
Transaction cost	Variable, can be high during congestion	Higher gas fees, variable with network demand	Does not have transaction costs
Scalability	Limited by block size and proof-of-work constraints	Improved post-Merge but still limited on base layer	High

Table 4: Architectural comparison of main blockchains.

The exploration of blockchain technology has revealed its capacity to disrupt and transform numerous sectors through its decentralized, transparent, and secure design. As demonstrated throughout this chapter, blockchain is no longer confined to theoretical research or niche use cases, it is now forming the backbone of practical applications that challenge centralized models. In particular, the implementation of blockchain in finance, healthcare, IoT, and public administration highlights its versatility and growing relevance across both technical and societal dimensions.

Looking ahead, the integration of blockchain with machine learning and artificial intelligence (AI) is expected to intensify, enabling more intelligent, secure, and autonomous systems across a wide range of applications. As the technology continues to mature, further interdisciplinary efforts will be essential to address technical limitations, regulatory concerns, and ethical considerations.

Related to the topic of this chapter, we have published a review article [35] focusing on Layer 2 scalability solutions in blockchain networks, and the papers [27], an analysis of blockchain-based initiatives in education, and [3], which proposes a blockchain-based model for the exchange of educational resources between teachers.

BUPT

3. The state of research in the field of credential management

This chapter presents the state of the art of studies and working methods related to the academic credential management. We have proposed a taxonomy of methods for a better understanding and classification of the different approaches to the problem.

3.1. Overview

The traditional model of a university providing a single, verifiable credential no longer fits the way people actually learn today [9]. They take a series of online courses, in-house training, and refresher courses held by educational institutions or organizations, but there are no systems to manage them. As a result, a learner's credentials are shaped by a combination of diplomas, academic certificates, qualifications in skills and competences, open recognition badges, and micro-credentials [91], together reflecting their educational background, skills, and achievements. All this credentials are presented as follows:

- *Degree*: is a formal qualification awarded by a higher education institution to recognize that an individual has completed a structured program of study and demonstrated a defined level of knowledge and skill in a particular field [92]. Academic degrees include associate, bachelor's, master's, doctoral, and professional degrees [93].
- *Diploma*: is a formal recognition awarded for the successful completion of a single subject or specialized program, typically involving both theoretical and practical training. It is commonly offered in technical fields and serves as a pathway for individuals seeking focused education without pursuing a full college degree.
- *Certificate*: is a formal document issued by an educational institution that attests to an individual's successful completion of a specific course or program [94].
- *Micro-credential*: certifies specific competencies and skills acquired through the completion of a course, seminar, or series of modules, usually delivered online and asynchronously, by educational institutions, companies, or professional and licensing organizations [95]. Micro-credentials provide a pathway for professionals to re-skill, upskill, and stay competitive in their industry [95].
- *Digital badge*: is a visual credential that represents a verified skill, achievement, or competency [96]. It includes metadata detailing what it recognizes, who issued it, and how it was earned, and can be shared online as portable proof of learning.
- *Attestation*: is a formal declaration issued to confirm that someone has successfully fulfilled the requirements of an educational task or program and demonstrated the related skills or knowledge [97]. It serves as official proof provided by an institution, certificate, or a portfolio entry.

BUPT

Table 5 provides an overview of the key characteristics of credentials.

Type of credential	*Description*	*Typical completion time*
Degree	Formal qualification for completing a full academic program	2-7 years
Diploma	Recognition for specialized or technical training	1–2 years
Certificate	Proof of completing a specific course or program	Weeks to 1 year
Micro-credential	Certification of specific skills or competencies	Hours to months
Digital Badge	Visual, shareable proof of a skill or achievement	Hours to weeks
Attestation	Formal declaration confirming completion of an educational activity or skill acquisition	Days to weeks

Table 5: Characteristics of credentials

The issue of fake credentials poses a significant threat to the integrity of both the education and employment sectors, stemming from various forms of document fraud. Saleh et al. [11] classify these fraudulent practices into five main categories. "Degree Mills" are entities that issue fake credentials without any academic coursework. "Fabricated Documents" involve the creation of entirely fictitious academic records. "Modified Documents" refer to legitimate credentials that have been altered to falsify information such as grades, graduation dates, or course details. Another concerning source is "In-House Produced" credentials, which are forged within genuine institutions by insiders using authentic materials such as official seals and signatures. Lastly, fraudulent "Translations" of academic documents involve deliberate misrepresentation during translation processes to manipulate grades or qualifications to meet foreign standards or requirements.

To address these challenges, increasing attention is being given to the adoption of digital verification systems designed to ensure the authenticity of all types of credentials, including academic, professional, vocational, and training credentials. These systems provide a secure and reliable way for employers, institutions, and other stakeholders to access and verify official records directly from the issuing bodies, thereby minimizing the risk of fraud across various sectors.

3.2. Traditional methods for issuing and verifying academic credentials

The lifecycle of academic and professional credentials involves a series of distinct stages, beginning with the identification of learning goals and concluding with the practical use of the awarded credential. Whether referring to degrees, certificates, micro-credentials, digital badges, or attestations, each type of credential follows a structured process that ensures the recognition of knowledge, skills, and competencies. Each step of the process presents potential risks related to fraud, verification, and data integrity, especially in traditional format. Table 6 outlines these key stages across different types of credentials, highlighting the specific activities involved, methods of issuance and verification, and associated risks or considerations for each category.

BUPT

Activity	*Degrees*	*Certificates, micro-credentials, digital badges*	*Attestations*
Selecting a learning goal	Engaging in a structured program of study is undertaken not merely as a requirement, but as a conscious investment in long-term academic and professional development.	The process begins by identifying a specific skill, area of knowledge, or professional competency aligned with career aspirations or industry needs.	Participation starts with selecting a workplace activity that resonates with the individual's current responsibilities, reflects their professional interests, and supports their personal passions, ensuring the experience is both relevant and motivating.
Completing an educational program or activity	Degree candidates complete a curriculum consisting of courses, assignments, research, and often a capstone or thesis project. These are designed to ensure depth and breadth in the field of study. Despite the structured nature of this stage, fraud can occur if a student is credited with completing parts of the curriculum, such as a course or thesis, without actually fulfilling the requirements. This may result from administrative errors, falsified records, or deliberate manipulation.	Once a pathway is selected, participants engage in a structured learning experience, such as a course, training session, or project module, designed to develop the targeted competencies. This type of training typically involves a financial cost, often requiring participants to pay for enrollment, materials, and assessment-related fees.	Individuals complete the designated tasks, participate in the relevant activity or initiative, and reflect on outcomes or lessons learned, often through feedback or a summary report.
Verification of academic requirements, final assessment and administrative audits	Once a student completes his program, the faculty's administrative office conducts a comprehensive review of the academic record to ensure all program requirements have been met. This includes verifying successful completion of all courses, satisfactory grades, and approval of any final projects or theses. Often, students must also pass a culminating exam or defend their thesis in front of a committee, which serves as a critical step in confirming their readiness to graduate. Additionally, a final check is made to confirm all	The awarding institution conducts a verification process to ensure all required activities have been completed, learning outcomes achieved, and assessments, where applicable, successfully passed. In many cases, automated systems monitor learner progress and initiate the issuance once all established criteria are met.	Once the activity is completed, the institution confirms that the individual has met the requirements and achieved the intended outcomes. However, when verification is weak or informal, there is a risk that attestations may be granted to individuals who did not genuinely take part. When used as evidence in performance appraisals, career advancement decisions, or professional development evaluations, attestations carry significant weight. If obtained dishonestly, they can misrepresent an individual's

BUPT

	institutional obligations are cleared, including any financial or administrative matters, ensuring the student is fully eligible to receive their degree.		qualifications, leading to unjust advantages and compromising the objectivity and fairness of internal evaluation systems.
Issuance of the degree, certificates, micro-credentials, digital badges, attestation and additional documents	Once eligibility is confirmed, the diploma is formally issued. This includes the preparation of the diploma certificate, which usually contains the university's name, the graduate's name, the degree title, the completion date, and the signature of the university rector and dean. After completing the information contained in a degree, it is printed on high-quality paper, to enhance its official appearance. At this stage, there may be risks of a counterfeit degree as it may be issued even if the student has not completed all the subjects correctly, with passing grades, or has not obtained a passing grade in the final examination or thesis defense. Upon graduation, the faculty in which the student is graduating issues a degree supplement detailing the courses and their grades.	Once all requirements are fulfilled, the credential, whether a certificate, micro-credential, or digital badge, is formally issued. It is typically delivered in a digital format that can be stored, shared, and displayed on professional platforms. However, this final stage is also where fraud risk emerges. If credentials are not protected through secure technologies, such as metadata, tamper-proof formats, or blockchain, they may be forged, altered, or misused. Fake certificates or badges can circulate online, especially on job platforms, undermining trust in the credentialing system. In order for these achievements to retain their credibility and value, secure issuance and robust verification methods are essential.	Following this verification, the organization provides an attestation, a formal statement recognizing the individual's participation and the competencies or insights gained through the activity. If verification is weak or informal, there is a risk that attestations may be granted to individuals who did not genuinely take part.
Distribution	Degrees are not usually handed out during the graduation ceremony itself. Instead, graduates typically receive a simplified certificate at the event. The official degree and transcript must be collected later through an administrative process. This involves first picking up the diploma supplement from the faculty office, then presenting it, along with two recent photographs and an identity card [98], to the university	Credentials are distributed via email or secure digital platforms. They can be added to LinkedIn.	Attestations are typically distributed as formal letters or official documents, and are incorporated into each individual's professional portfolio.

BUPT

	registrar to receive the diploma. While there is no fee for on-time collection, storage or reissue fees may apply if the degree is not picked up within a certain timeframe [99].		
Archiving	Paper degrees are specific to traditional education. This system involves issuing physical degree to graduates, which serve as tangible proof of academic achievement, and simultaneously, a corresponding copy of each degree is maintained by the issuing institution, either in a physical register or within a local digital database. The degree is archived indefinitely. These databases risk being infected by viruses or malware, which can compromise data integrity, lead to unauthorized alterations, or even result in data loss.	Micro-credentials and certificates are logged into digital learning platforms.	Attestations are recorded in internal human resources systems or professional development files. This method of archiving can lead to issues such as deterioration, misplacement, or limited accessibility over time.
Use of the diploma	The degree is used to enroll in another form of education, to get a job, to get another qualification or to get wage benefits. Centralized solutions require trust from document owners. At this stage, the risk of a counterfeit degree can arise, because people who have not studied at a university can get hold of a fake diploma modeled after those issued by the university and use it to obtain benefits. In order for an employer to verify the authenticity of the diploma, he must contact the issuing university [91], which consumes a lot of time for both parties considering that dozens or even hundreds of people are applying for the	These are often used to showcase specific skills or competencies, apply for niche roles, gain continuing education credits, or demonstrate ongoing professional development. This stage is vulnerable to fraud, including the use of fake certificates or badges created with design software, credentials from unaccredited institutions or diploma mills, altered documents with manipulated names or achievements, certificates earned by someone else due to weak identity checks, or system errors which may even generate badges for users who haven't	Attestations are used internally to support promotions, justify participation in professional development, or demonstrate engagement in organizational initiatives. When used as evidence in performance appraisals, career advancement decisions, or professional development evaluations, attestations carry significant weight. If obtained dishonestly, they can misrepresent an individual's qualifications, leading to unjust advantages and compromising the objectivity and fairness of internal evaluation systems.

BUPT

	same job. The university can issue a certificate of authenticity of a degree. Within the University of Oradea, the fee for issuing the certificate of authenticity of the diploma is 30 euros [99].	completed the required learning.	
Revoking	One of the most significant limitations is the inability to revoke or modify a paper degree once it has been issued unless the recipient physically returns it [100], a scenario that is rarely enforceable.	If the training took place online, it can be removed from the owner's account, but it could have been downloaded from the platform and used in paper format.	The recipient must physically return the attestation.

Table 6: Comparative overview of credential lifecycle stages

A thorough examination of current credential management systems highlights the need for improvements and strengthens the case for a more secure and efficient solution.

3.3. Digital signature integration in authentication of academic credentials

A digital signature is a cryptographic method that consists of encrypting a document with a private key and decrypting it with the corresponding public key [101], ensuring the authenticity of the document.

Electronic documents offer convenience and efficiency, but this digital advantage comes with significant security risks. As these documents are accessed and transmitted through public channels, they are vulnerable to various cyber-attacks that can compromise their confidentiality, integrity, and authenticity. Such attacks include [102]:

- replay attacks, where legitimate data is delayed or repeated maliciously;
- man-in-the-middle attacks, where an attacker intercepts and alters communication between two users;
- impersonation attacks, where attackers pose as legitimate senders to trick recipients;
- compromise attacks, which involve unauthorized access to or alteration of the original data;
- masquerade attacks, in which an attacker uses a fake identity to gain unauthorized access to the document.

Penubadi et al. [102] addresses the security challenges associated with protecting electronic documents, particularly those containing sensitive information, in an open web environment. It highlights the limitations of traditional encryption methods and the need for more efficient, integrated solutions. Specifically, it proposes a comprehensive security framework combining encryption, digital signatures, and watermarking algorithms to safeguard electronic documents. Specifically, it uses the secure hash algorithm-256 (SHA-256) and digital signature algorithm (DSA) to create a secure digital signature for each document,

BUPT

ensuring its authenticity and preventing unauthorized alterations. Following this, the document undergoes a two-stage encryption process. In the first stage, the document is secured using asymmetric encryption with public and private keys, while the second stage strengthens security through symmetric encryption, applying multiple transformations to the data to make it more resistant to attacks. Once encrypted, the document is watermarked using a Dynamic Wavelet Transform-based technique, embedding a watermark that further protects the document and helps in identifying any tampering. When a recipient requests the document, the system verifies the integrity and authenticity of the document through the digital signature, decrypts it using the recipient's keys, and presents the original content. This multi-layered approach ensures confidentiality, integrity, and authenticity, addressing the gaps left by traditional security methods.

3.4. QR code integration in authentication of academic credentials

Building upon established credentialing practices, this section explores how printed diplomas can incorporate quick response (QR) code technology, embedding a digital layer that strengthens security and streamlines verification. This practice is already confirmed by some universities or countries that already apply QR codes on their diplomas [103].

A practical example of how to use QR code for enhancing the security and authenticity of academic credentials is presented in [104]. The process begins with educational institutions generating a unique cryptographic key pair, comprising a public and a private key. Upon issuing a certificate or transcript, the institution encrypts the credential's details using its private key, creating a ciphertext. This encrypted data is then embedded into a QR code, which is printed on the physical document. To verify the authenticity of the credential, a third-party verifier scans the QR code using a secure application provided by the issuing institution. The application retrieves the encrypted data and decrypts it using the institution's public key, displaying the original credential details for comparison. This method ensures that any alteration to the physical document would result in a mismatch during the verification process, effectively preventing forgery and fabrication.

Ahmed and Jang [105] presents a system ensures the document’s authenticity by incorporating a university’s digital signature into a QR code that contains key certificate details, such as the graduate’s name, field of study, and academic performance. This QR code is printed directly onto the certificate, allowing verifiers to instantly confirm its validity using a dedicated mobile application, without the need to contact the issuing institution. This method not only strengthens security but also simplifies the verification process.

3.5. Blockchain-based solutions for issuing and verifying credentials

In recent years, educational research has shown strong interest in the use of blockchain to increase the security and credibility of academic credentials [106, 107, 108, 109, 110, 111, 112, 113, 114]. A significant portion of this work has focused on the registration of formal degrees, with Ethereum emerging as the preferred platform due to its programmable infrastructure. These systems typically encode essential credential details into a digital format, secure them with a cryptographic signature from the issuing institution, and store the resulting

BUPT

hash on the blockchain to ensure authenticity and tamper-resistance [115]. The core idea is to replace traditional, centralized verification methods with decentralized systems that offer greater transparency and resistance to tampering. By anchoring credential information on a public blockchain, institutions aim to provide a more reliable and efficient means of verifying qualifications, particularly in combating fraud and streamlining recognition across institutions and borders.

The increasing relevance of blockchain in the academic domain has facilitated the rise of several credential management platforms, as documented in [11, 116, 117, 118, 119, 22], which are now considered leaders in the field. Among these, Blockcerts is recognized as the pioneering platform, setting a foundational standard for secure and verifiable digital credentialing. Blockcerts is the digital wallet designed by the MIT Media Lab to manage certificates and diplomas, with ongoing development initially carried out on the Bitcoin blockchain by Learning Machine, a software development company, and is now advanced by Hyland Credentials [116]. In 2017, 111 graduates, 85 masters in finance and 26 masters in arts and media sciences, became the first to have the option of receiving their degrees on smartphones via Blockcerts, in addition to the traditional format [120].

References [11, 121, 27] detail the procedure for issuing and verifying degrees or certificate through the Blockcerts wallet, illustarted in Figure 9. The first step is to generate a diploma in pdf format, immediately after graduation or after completing a course, which is sent to the graduate via email that contains, in addition to the diploma in digital format, a link from which the graduate can download the Blockcerts wallet. The graduate downloads the Blockcerts wallet on their smartphone, and at this point the algorithms automatically generate the public-private cryptographic key pair. The private key appears in the form of a mnemonic phrase, which is not recoverable. The graduate uploads the diploma to the Blockcerts and sends the public key to the issuing institution, which uses it to create a hash of the diploma, which attests the authenticity of its contents. The third step is for MIT to create a Bitcoin transaction with the hash of the diploma. Finally, MIT emails the diploma as a JSON file with the student's public key on it. To verify the authenticity of a diploma, the student can prove that they are the holder of the diploma using the private key in their Blockcerts wallet. The authenticity of the diploma can also be proven by accessing the MIT verification portal, credentials.mit.edu/. After pasting a link or uploading the JSON file of the student's diploma, the portal searches the Bitcoin blockchain for the transaction with the diploma hash and verifies the keys, immediately displaying confirmation of the diploma's authenticity.

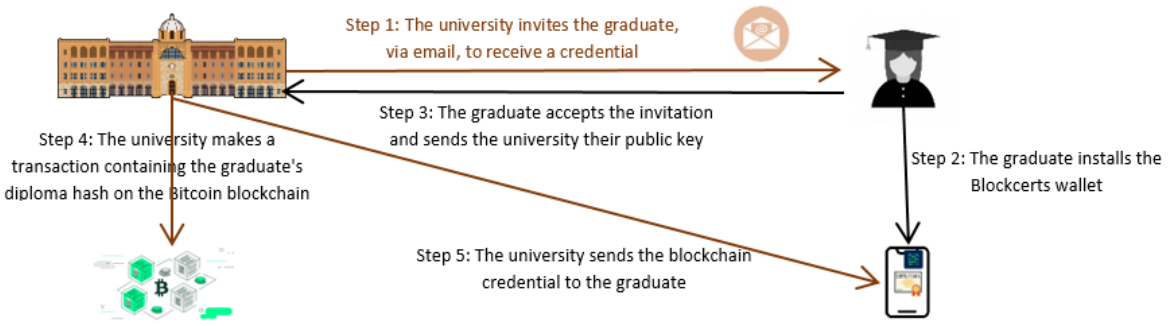


Figure 9: Blockcerts workflow. Source: own illustration

BUPT

An important feature of Blockcerts is its blockchain-agnostic design, initially integrated with Bitcoin in 2017, then extended to Ethereum, and tested with Hyperledger in 2018 [122], though there is no information on testing with other blockchains. Additionally, Blockcerts is adaptable to work with any issuer. After creating Blockcerts, MIT, which is not actively involved in the ongoing development [123], and Machine Learning looked for partners to implement a pilot in the use of the Blockcerts platform in their universities. Among the first partners, since 2017, were Malta College for Arts Science and Technology [124], University of Melbourne [119], to which were added in the following years other universities such as ECPI University in Virginia, USA, Maryville University, USA [125] and many others.

Another application that is globally recognized as a best practice is Block.co. Block.co was developed by the University of Nicosia with the aim of validating on the Bitcoin blockchain all its undergraduate degrees, namely Bachelor, Master and PhD [11, 116, 117, 118]. At the beginning, in 2015, University of Nicosia handed diplomas, using blockchain, to students who completed the "Introduction to Digital Currencies" course, but since 2017 every student who graduates at this university receives diplomas on blockchain [126].

The process by which the diplomas are issued, validated and verified at the University of Nicosia is described as follows. The first step is to generate a diploma in pdf format, a diploma that contains the name of the issuer, the name of the diploma, the year of graduation, the name of the student, the courses taken and the results of the evaluation, the date of issuance of the diploma. This diploma is sent to the graduate's email address [127]. The second step consists of digitally signing the diploma, using the university's private key. Then a hash of the diploma is created using the SHA-256 algorithm and is passed into a pdf file, along with all the other hashes of the graduates' diplomas in that series. This pdf file is sent to the email address of each graduate in that series, along with the diploma in pdf format. Then a hash is created, using the SHA-256 algorithm, of the PDF document containing the hashes of the graduates' diplomas, [127]. The third step consists of the University of Nicosia creating a Bitcoin transaction with the hash of the PDF document containing the hashes of the graduates' diplomas, in order to add it to the Bitcoin blockchain.

To verify the authenticity of a diploma, the hash of the diploma must be calculated, using the SHA-256 algorithm. This hash must be found in the pdf file sent by the university, along with the diploma. Even if the hash of the diploma is found in the list, it must be proven that the list is authentic. To verify that this list of certificates is authentic, calculate the hash of the PDF document containing the hashes of the diplomas of the graduates on the list and search for this hash in the Bitcoin blockchain using an application such as Bitcoin Core, as presented in Figure 10.

BUPT

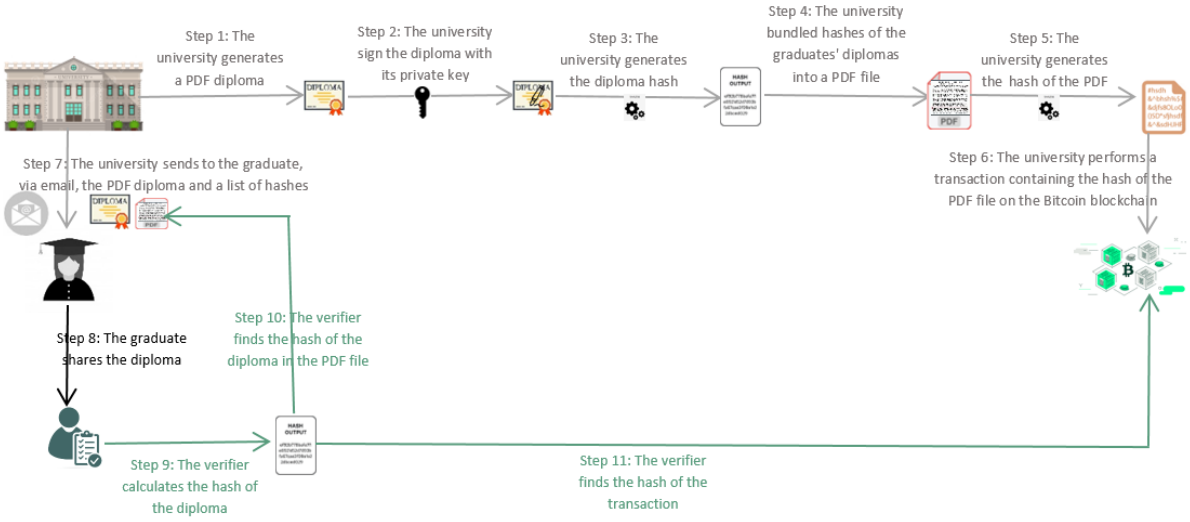


Figure 10: The University of Nicosia blockchain diploma management workflow. Source: own illustration.

A significant part of the literature on blockchain in education centers around the processes of issuing and verifying academic credentials [87, 27]. Recent articles addressing the issue, management and verification of certificates and transcripts propose distinct solutions each set apart by blockchain platform chosen and approach. Tahlil et al. [128] developed a prototype on the Algorand blockchain that uses smart contracts and NFTs to issue and authenticate academic certificates, significantly reducing the risk of forgery. Ahmed et al. [12] introduced CredChain, a platform focused on certificate verification using Ethereum. CredChain provides secure access to academic records for students, universities, and authorized third parties, with its performance thoroughly analyzed in their study. Two articles explore the practical implementation of the European Blockchain Services Infrastructure (EBSI) to streamline cross-border educational mobility through a secure and transparent credential management workflow. Tan et al. [19] illustrate a pilot case involving the issuance, storage, and verification of a student's identity and academic transcript between KU Leuven in Belgium and the University of Bologna in Italy. This example carefully maps the step-by-step process enabling seamless exchange of academic records across national boundaries. Complementing this, Holotescu et al. [20] present a comprehensive scenario following a student's academic journey from a bachelor's degree at Politehnica University of Timisoara, through a master's degree at the University of Athens, to a PhD application at the University of Lille. Their study highlights how blockchain facilitates an integrated workflow for issuing and validating credentials across multiple institutions and countries, demonstrating the technology's potential to enhance trust and mobility within the European higher education landscape. Future developments seek to create an application that can issue and verify every diploma an individual holds, regardless of its level or field of specialization. As examples, the following initiatives present complementary approaches to lifelong academic and professional records, addressing common challenges. Litoussi et al. [15] propose a decentralized Ethereum-based model for Morocco that integrates all university and workplace certificates for an individual. Kistaubayev et al. [14] emphasize the need for a unified digital registry that supports diverse international educational paths, including Erasmus and DAAD programs, and securely stores diplomas, transcripts, micro-credentials, digital badges, and developed a dApp on Ethereum consortium to address this.

BUPT

A significant limitation of many blockchain-based credential systems is their exclusive focus on the final certificate, ignoring the integrity of the entire academic process. Nzaro et al. [16] propose a blockchain-based model that secures the entire academic certification process, not just the final certificate, through the use of smart contracts. The model addresses three key phases: admission, examination, and certificate issuance. In the admission phase, smart contracts verify the authenticity of applicants' previous credentials to prevent fraudulent entries. During the examination phase, they enforce academic standards and ensure fair and transparent assessment procedures. Finally, in the issuance phase, smart contracts confirm that all academic requirements have been met before issuing a verifiable certificate, with all data immutably recorded on the blockchain.

Whereas the majority of research has concentrated on recording university degrees on the blockchain, Choi et al. [129] showed that digital Open Badges can likewise be registered, offering a secure and verifiable way to recognize smaller-scale learning achievements. Open Badges, launched by the Mozilla Foundation, is a format that aims to standardize the documentation of achievements and skills [130]. Only digital badges that adhere to the Open Badges standard, containing verifiable information about the recipient, achievement criteria, badge representation, issuing organization, and demonstrated competencies, are recognized as digital Open Badges [131]. The system utilizes blockchain technology to record each digital badge issuance as an immutable entry on a decentralized ledger. When a badge is awarded according to the Open Badges standard, it contains metadata about the achievement, issuer, and recipient. This information is linked to a blockchain transaction, ensuring that the badge's authenticity can be independently verified without relying on a centralized authority. Learners can collect and manage their badges in a personal digital "backpack," which serves as a secure repository. By scanning or accessing these badges, employers or institutions can confidently validate the learner's credentials, benefiting from the transparency, security, and permanence that blockchain provides.

It is possible to connect a paper diploma to blockchain by embedding a QR code that links to a decentralized application, allowing instant verification of the certificate's authenticity, an approach addressed by limited existing research and demonstrated [132, 110]. Building on this concept, Tariq et al. [110] created Cerberus, an Ethereum-based system that combines blockchain technology with QR codes to streamline the issuance, verification, and revocation of certificates and transcripts. In Cerberus, graduate records are stored in JSON format, hashed with SHA-256, and recorded on the blockchain via signed transactions. Employers can then verify certificates by scanning QR codes on physical certificates, instantly accessing the secure blockchain data. Gangwar and Chaurasia [132] propose a system that enables educational institutions to digitally issue certificates through the dApp, which generates a unique QR code linked to the certificate's secure record on the blockchain. When an employer or recruiter scans the QR code on the certificate or inputs the certificate number into the dApp, they can instantly verify its authenticity by accessing the immutable certificate data stored on the blockchain. This eliminates the need for physical verification and reduces the risk of forgery by leveraging blockchain's transparency and security alongside the convenience of QR code scanning.

BUPT

The vast majority of studies propose solutions based on a Layer 1 blockchain, with Ethereum being the most in demand. Only a limited number of studies have explored academic credential management solutions utilizing Layer 2 blockchain technology. Among them, Tan-Vo et al. [51] present an advanced framework that combines blockchain with machine learning to address challenges of scalability, security, and fraud prevention. Central to the framework is the implementation of optimistic rollups via Arbitrum, a Layer 2 scaling solution, which aggregates certificate transactions off-chain before periodically committing them to the Ethereum mainnet as the Layer 1 blockchain. This approach reduces network congestion and transaction fees while enabling efficient handling of certificate lifecycle events, including revocation. The revocation process is managed transparently on the blockchain, ensuring that revoked certificates are promptly and verifiably marked as invalid within the system. Complementing this scalable blockchain infrastructure, the framework integrates machine learning algorithms designed to detect and prevent fraudulent certificates by analyzing transaction patterns and certificate metadata. This approach is particularly innovative in the educational domain, where Layer 2 solutions are rarely explored.

Another publication that chose a Layer 2 solution is [133]. Fartitchou et al. [133] presents BlockMEDC, a blockchain-based system developed to improve the security and efficiency of issuing and verifying digital academic credentials in Morocco's higher education sector. Aligned with national digital transformation goals, BlockMEDC uses smart contracts deployed on Ethereum Layer 2 with zk-Rollups to securely and efficiently manage academic records. The use of zk-Rollups, which is a scalability solution that batches multiple transactions off-chain and submits a single proof to the main Ethereum network, allows the system to reduce costs and energy consumption while maintaining high security and transparency. In addition, academic documents such as diplomas, transcripts, and registration certificates are stored in a decentralized way using IPFS. By combining blockchain and decentralized storage, BlockMEDC ensures the authenticity, integrity, and easy verification of credentials, addressing common issues like fraud, manual verification delays, and institutional interoperability.

BUPT

3.6. Summary

The landscape of learner credentials is undergoing a significant transformation in response to evolving educational practices, technological advancements, and the growing need for flexible, lifelong learning pathways. This chapter has provided a comprehensive overview of how learners' achievements are now represented through a variety of formats, ranging from traditional diplomas and academic certificates to qualifications in specific skills and competences, open recognition badges, and micro-credentials.

Historically, academic credentials have been issued and verified through paper-based, centralized systems managed by institutions and government bodies. While these methods have been in place for decades, they are increasingly criticized for their inefficiencies, vulnerability to fraud, and inability to keep pace with the digital demands of modern educational and employment ecosystems. Manual verification processes are often time-consuming, institution-dependent, and difficult to scale, particularly in international or cross-sector contexts. As education becomes more global and skills more transferable, the limitations of these traditional systems become more apparent. To address these issues, the chapter explored several emerging technological solutions that aim to improve the security, authenticity, and accessibility of academic credentials.

One of the foundational technologies in this area is the use of digital signatures. By applying cryptographic techniques, digital signatures ensure that a credential is uniquely tied to the issuing authority and has not been altered or forged.

Another important innovation discussed is the integration of QR codes in credential authentication. QR codes act as digital gateways, enabling fast and user-friendly access to credential information. When embedded in certificates or digital documents, QR codes can link to verified data stored in secure environments, facilitating instant validation by employers, educational institutions, or other stakeholders. Their ease of use and compatibility with smartphones and other devices make them a practical tool for bridging physical and digital credential formats.

The most transformative approach examined in this chapter is the application of blockchain technology in the issuing and verification of academic credentials. Blockchain offers a decentralized, tamper-resistant, and transparent infrastructure for managing educational records. Credentials stored on the blockchain can be verified independently by any third party, fostering trust and interoperability across institutions, industries, and borders. Platforms such as Ethereum, with its support for smart contracts, have been particularly influential in enabling programmable, automated credential systems that support lifelong learning and open recognition frameworks.

As these technologies mature and gain wider adoption, they hold the potential to create more equitable and efficient systems for verifying learning achievements, supporting not only academic institutions but also learners, employers, and policymakers in building trustworthy and interoperable educational ecosystems.

BUPT

4. Blockchain-based solution for managing teacher credentials and access to educational resources

4.1. Description of the problem

Teachers working in the pre-university education system in Romania are required to compile detailed portfolios containing printed copies of various documents in several key situations, presented below.

Evaluating teacher performance at the end of each academic year

At the conclusion of each academic year, every teacher undergoes a formal evaluation procedure aimed at assessing their professional performance in accordance with the criteria outlined in [4]. The scoring scale for awarding annual performance ratings, in accordance with the current methodology, is as follows:

- from 100 to 85 points the rating is „Very good”;
- from 84.99 to 71 points the rating is „Good”;
- from 70.99 to 61 points the rating is „Satisfactory”;
- below 60.99 points the rating is „Unsatisfactory”.

For this assessment, teachers must complete a self-evaluation form [134] and prepare supporting documents in physical format, demonstrating the following:

- *Teaching activities*, demonstrated through effective lesson planning, delivery, and the use of varied methodologies to support learning.
- *Students' academic progress and performance*, shown in consistent improvement, examination results, and development of critical skills.
- *Involvement in school initiatives and projects*, through active participation in institutional programs and collaborative efforts.
- *Participation in symposia and conferences*, ensuring professional visibility and the integration of new pedagogical insights.
- *Engagement in continuing education*, marked by training, courses, and certifications that enhance expertise and practice.
- *Overall professional conduct*, reflected in ethical behavior, teamwork, and a commitment to fostering a respectful learning environment.All of this requires organizing multiple copies on paper.

Following the self-evaluation phase, each teacher's professional activity is assessed by two distinct committees. The Evaluation Committee, composed of teaching staff, reviews the scores recorded in the self-evaluation sheet, draws on additional insights from the accompanying explanatory report, and, where necessary, examines the supporting documentation submitted by the teacher. The final stage of the evaluation is carried out by the Board of Directors, which reviews and formally validates the teacher's performance by assigning the final score and corresponding rating.

BUPT

This multi-layered evaluative process has several disadvantages, as follows:

- requires significant time to be devoted to reviewing these sheets and reports and verifying the authenticity and relevance of supporting materials, where applicable, which imposes a considerable administrative burden on evaluators.
- the exclusive reliance on paper-based documentation poses additional challenges related to data security and integrity, including the potential for document loss, unauthorized modifications, as well as the possibility of submitting forged documents, making it difficult to ensure the integrity of the evaluation process.

Evaluating teachers' professional development

Every five years, teachers must also demonstrate that they have completed professional development courses totaling at least 90 transferable credits. Upon completing each course, they are required to submit a physical copy of the certificate to the staff member responsible for tracking professional development, which raises many disadvantages:

- *Accumulation of documents*, as repeated submissions over a five-year span generate large volumes of printed records.
- *Vulnerability of storage*, since physical files remain exposed to risks of loss, damage, or misplacement.
- *Risk of misrepresentation*, where duplicate or fraudulent certificates may be submitted without reliable safeguards.
- *Uncertainty of authenticity*, particularly when certificates are issued by providers that no longer operate or cannot be contacted.
- *Lack of efficient tracking*, with no digital infrastructure to ensure quick verification or provide a comprehensive record of professional development.
- *Inconsistency in format*, arising from the absence of standardized practices across course providers.
- *Administrative inefficiency*, requiring staff to manually process and verify extensive paperwork, which increases the likelihood of human error.

This weakens the integrity of the entire professional development oversight process. Institutions cannot be certain that their staff has truly completed the required training.

Assessment of teacher mobility

In the stages of teacher mobility, several distinct situations require careful administrative handling, as they directly affect access to available teaching positions:

- *Transfer or only moving for a specific period from one school to another*, which may occur due to professional development opportunities, personal circumstances, or institutional restructuring.
- *Reduction of teaching hours in a school*, most often caused by a decline in student enrollment, particularly in rural areas, a process that can generate simultaneous classes or even lead to the closure of schools.

BUPT

- *Completion of the teaching norm*, necessary when a school reduces the number of classes at certain levels or when legislative changes, such as the recent increase in the minimum required hours of activity with students from 18 to 20 [135], come into effect.

In each of these cases, teachers must submit to the school inspectorate a detailed portfolio containing documents that attest to their professional activity [136]. A significant component of this portfolio is reflected in the teacher's methodological and scientific activity, rigorously documented through official records and supporting materials. These contributions demonstrate sustained professional growth and are structured across two complementary timeframes:

- In the last 2 years, the teacher has shown:
  - Preparation and presentation of scientific or pedagogical papers, with participation in recent conferences, symposia, and workshops.
  - Design of innovative didactic resources and methodological guides that directly support contemporary classroom practice.
  - Active engagement in professional associations, evidenced by contributions to current projects, working groups, and specialized networks.
- Over the last 5 years, the teacher's broader activity includes:
  - Consistent involvement in research initiatives and institutional projects, contributing to experimental teaching programs or studies aimed at school development.
  - Publication of scientific articles, pedagogical studies, and curricular support materials, demonstrating a sustained contribution to the educational field.
  - Recognition of professional achievements, through awards, certificates of merit, or distinctions received at local, national, or international levels.

This portfolio holds particular importance, as it forms the basis for calculating a score that prioritizes access to available positions. For this reason, a rigorous and transparent verification process is essential to ensure fairness in the redistribution of teaching roles and to uphold equity within the education system.

The awarding of a merit-based distinction to teachers in recognition of exceptional professional performance

In order for a teacher to obtain a merit-based distinction, they must compile a portfolio containing printed documents, following the requirements set by the Ministry of Education. For every relevant activity, teachers typically receive an official certificate issued by the school, confirming their involvement. These certificates are then added to the printed portfolio submitted for evaluation.

BUPT

A significant component of this portfolio consists of the teacher's multifaceted contributions across several domains of educational practice. These include [137]:

- *Advancement of student achievement*, demonstrated through sustained learning progress, consistent success in examinations, and thorough preparation for olympiads or specialized competitions.
- *Commitment to inclusion and support*, expressed in guiding students at risk, addressing the needs of those with special educational requirements, and fostering participation among learners from disadvantaged backgrounds.
- *Promotion of innovative teaching*, by integrating contemporary pedagogical approaches, digital resources, and emerging educational technologies into classroom practice.
- *Authorship of educational materials.*
- *Professional contribution to the field*, through the active service as evaluator, trainer, or advisor within institutional and national educational structures.
- *Active extracurricular engagement*, visible in the coordination of projects, clubs, cultural or community initiatives, and volunteer programs that extend learning beyond the classroom.
- *Support for institutional development*, achieved by attracting resources or establishing partnerships.

The merit-based distinction process involves the submission of hundreds of portfolios, each containing hundreds of documents. These portfolios are reviewed by county commissions established for each specialty. Given the sheer volume and variety of materials, the evaluation process is difficult to manage, as it is often challenging to analyze and to determine the authenticity of each individual document. The commissions are responsible for verifying the authenticity of the submitted documents, assessing their suitability for the designated category, and ensuring that they fall within the required timeframe.

In most of these evaluations, the portfolios include extensive evidence of a teacher's professional activity within the school. Consequently, a substantial proportion of the supporting documents are issued directly by the school principal, who is responsible for certifying the teacher's involvement in various educational, administrative, and extracurricular activities. When considering the process of obtaining merit-based distinction, based on the official evaluation form [137], it becomes evident that many criteria rely on supporting documents, such as attestations issued by the school principal. As illustrated in Figure 11, these locally issued attestations play an important role in demonstrating a teacher's eligibility for merit-based recognition.

BUPT

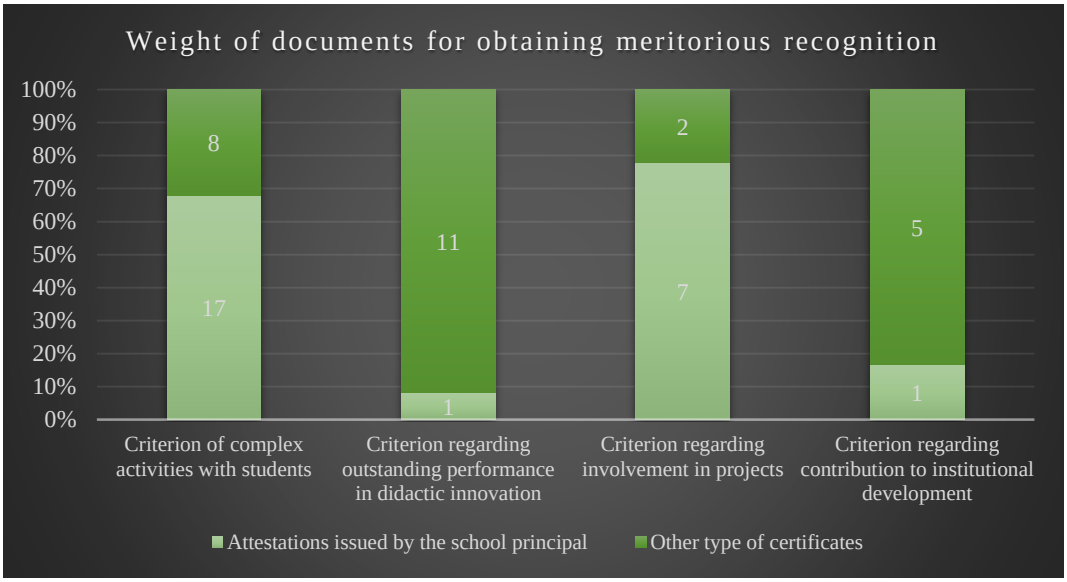


Figure 11: Representation of the weight of document types for obtaining meritorious recognition. Source: Own illustration, based on the results from [137]

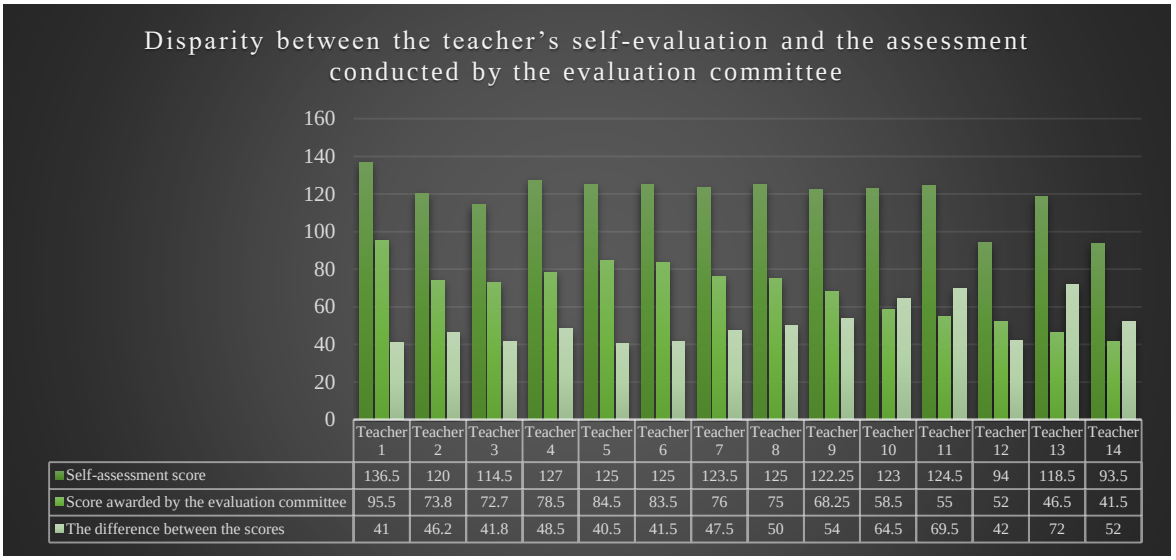
Despite their importance, attestations issued by the school principal are often vulnerable to a range of issues, including physical loss, manipulation, inconsistent recording or even fraudulent reproduction.

Another significant source of documentation for these evaluations comes from professional development courses attended by teachers. These courses are specifically designed to address the needs of educators, focusing on pedagogical strategies, subject-matter expertise, and emerging educational technologies. As a result, the certificates and attestations obtained from such programs form an important part of the evaluation portfolio, serving as evidence of a teacher's commitment to continuous professional growth. However, the wide variety of providers and formats for these courses also raises challenges in terms of standardization and in verifying the quality and authenticity of the credentials submitted.

Teachers are also the creators of many of the educational resources they employ in their instructional practice. The design and development of these materials, particularly those demonstrating innovation, rigor, and pedagogical quality, constitute an essential component of the evaluation portfolio.

Without a centralized, tamper-proof method of verification, the integrity of a teacher's professional portfolio can be compromised, resulting in unfair assessments or missed opportunities. Figure 12 illustrates the significant disparity between the teacher's self-evaluation and the assessment conducted by the evaluation committee in the process of obtaining meritorious recognition [138], emphasizing the most substantial point differences recorded. Therefore, great effort is required from the committee members to evaluate the printed documents from each portfolio and verify their validity, and the need for a more efficient solution is evident.

BUPT



	Teacher 1	Teacher 2	Teacher 3	Teacher 4	Teacher 5	Teacher 6	Teacher 7	Teacher 8	Teacher 9	Teacher 10	Teacher 11	Teacher 12	Teacher 13	Teacher 14
Self-assessment score	136.5	120	114.5	127	125	125	123.5	125	122.25	123	124.5	94	118.5	93.5
Score awarded by the evaluation committee	95.5	73.8	72.7	78.5	84.5	83.5	76	75	68.25	58.5	55	52	46.5	41.5
The difference between the scores	41	46.2	41.8	48.5	40.5	41.5	47.5	50	54	64.5	69.5	42	72	52

Figure 12: Illustrations of the significant disparity between the teacher's self-evaluation and the assessment conducted by the evaluation committee in the process of obtaining meritorious recognition. Source: Own illustration, based on the results from [138]

According to this section, certain activities appear repeatedly within the evaluation framework. For clarity and systematic analysis, we grouped these activities into three main categories:

- *Curricular and extracurricular activities*, reflecting direct engagement with students and the broader school community;
- *Creation and development of educational resources*, encompassing lesson plans, digital content, and other teaching materials;
- *Engagement in professional development programs*, which documents ongoing efforts to enhance pedagogical skills and maintain up-to-date knowledge in the field.

A blockchain-based platform could address current evaluation challenges by providing a secure, immutable, and unified record of all teacher activities. It would automatically log curricular and extracurricular engagement, educational resource development, and professional development participation, ensuring accuracy, preventing duplication, and enabling instant verification. By structuring these records within the platform, teacher contributions can be assessed consistently and transparently, reducing administrative burden and supporting fairer evaluation.

BUPT

4.2. Proposed solution

4.2.1. Mapping document types and issuing institutions within the proposed solution

The solution we propose for managing teachers' documents related to their evaluation and access to educational resources builds on the considerations outlined in section 4.1. As a starting point, it is important to distinguish several categories of documents that form the foundation of a teacher's evaluation portfolio:

- *Attestations*: When teachers take the initiative to organize or join an activity, their involvement is usually acknowledged with a certificate.
- *Educational resources*: At a time when students are accustomed to technology from an early age, the textbook, notebook, and blackboard no longer define the core of the lesson, but instead, modern teaching increasingly relies on technology-enhanced educational resources to capture and sustain students' attention.
- *Academic credentials*, including certificates, diplomas, or digital badges, are earned through advanced training courses and serve to formally recognize teachers' expertise, professional growth, and commitment to improving their educational practice.

Figure 13 provides a detailed overview of these files, illustrating the specific types of documents they contain.

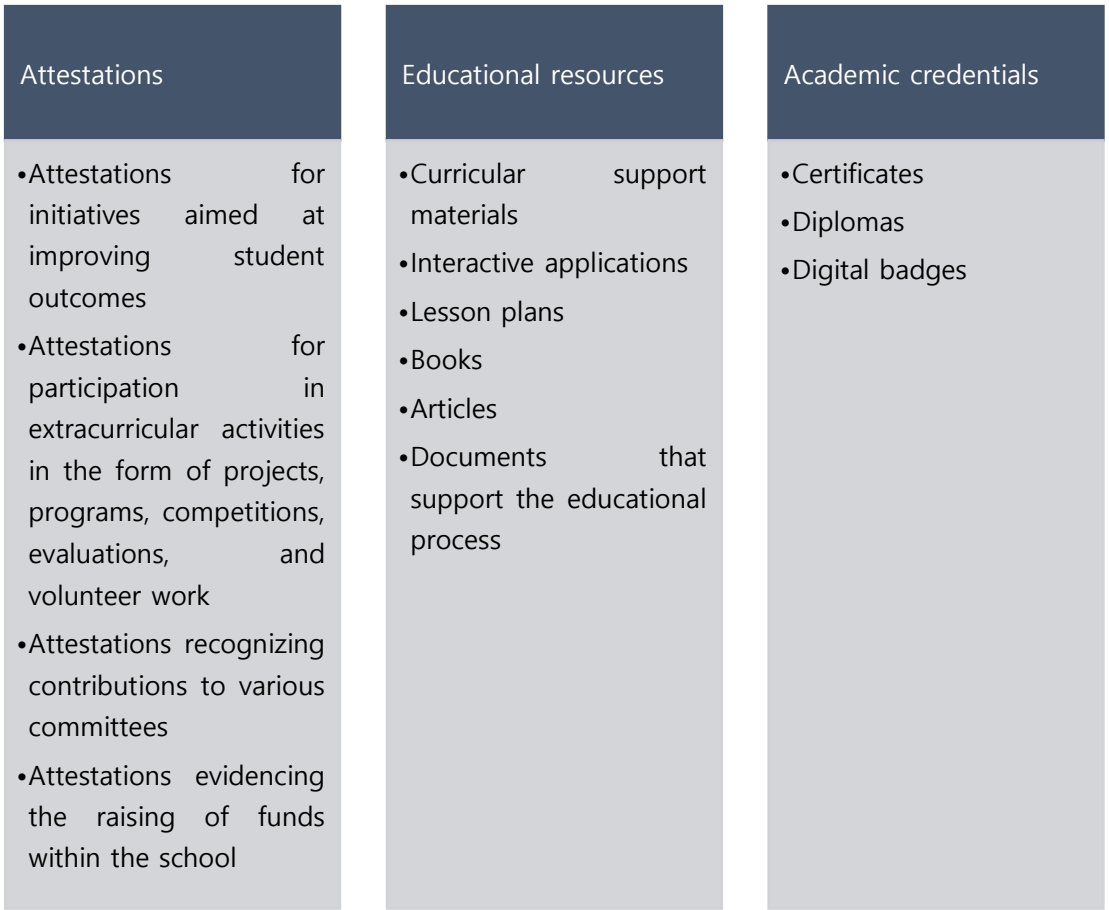


Figure 13: Detailed illustration of the files and their constituent document types.

BUPT

Section 4.1 also provided us with the bodies involved in the creation of these documents, illustrated in Figure 14:

- *School director* emits attestations, since the activity is connected to the life of the institution because it involves the school's own students and takes place within its facilities.
- *Teachers* create a variety of educational resources, including lesson plans, teaching materials, and digital content, books, and manuals for classroom use.
- *A professional training organization or certifying body* issues certificates, diplomas or digital badges for specific professional development programs.

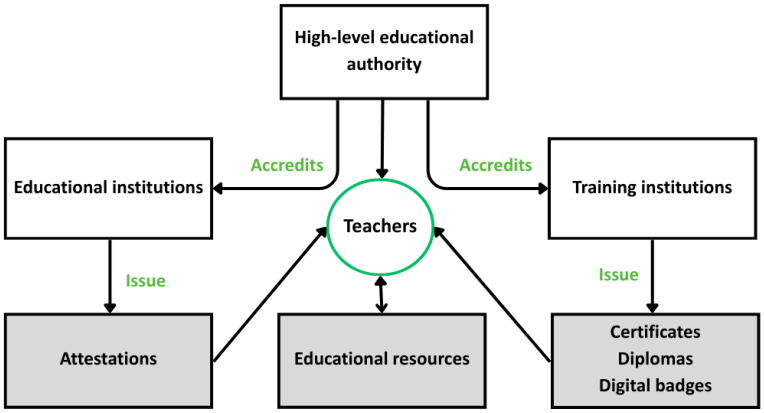


Figure 14: Entities participating in the development of the documents.

4.2.2. Rationale for blockchain in the proposed solution as an alternative to traditional and advanced databases

Considering the significance of these documents in the evaluation of teachers, we propose a blockchain-based system for recording attestations, certificates, diplomas, digital badges, and even educational resources, providing a structured and permanent framework for documenting professional achievements, tracking the accumulation of qualifications over time, and ensuring that all relevant contributions to teaching and learning are systematically preserved and easily referenced for evaluation purposes.

Adopting blockchain for managing teachers' attestations, certificates, and diplomas provides a rigorous and systematic response to a central challenge in educational governance: ensuring the integrity, authenticity, and permanence of academic records. Even the most advanced database architectures, whether relational, replicated, or encrypted, remain dependent on centralized authorities or trusted administrators, making them susceptible to insider threats, systemic vulnerabilities, or undetected breaches [139]. Blockchain establishes a decentralized and tamper-evident ledger in which each record is cryptographically hashed, linked to prior entries, and independently verifiable across all network participants [140]. This structural shift from centralized trust to distributed validation guarantees that credentials cannot be altered without detection, thereby reinforcing institutional credibility, reducing opportunities for fraudulent manipulation, and providing transparent and enduring verification essential for equitable and reliable evaluation processes.

BUPT

Blockchain's append-only and consensus-driven architecture further enhances the security and accountability of academic records by embedding auditability and verifiable provenance directly into the system [140]. Unlike traditional databases, which may deter casual tampering through access controls, blockchain ensures that no single compromised actor or cryptographic flaw can undermine the integrity of the ledger. Each transaction is permanently recorded and verifiable across the network, creating a comprehensive and immutable historical record of all actions. This feature not only strengthens accountability but also reduces the reliance on procedural safeguards and institutional oversight, providing a level of trustworthiness unattainable in conventional database systems.

The practical value of blockchain is exemplified by Estonia's experience with digital identity systems [141]. Prior to adopting blockchain-based technologies, Estonia relied on highly secure cryptographic databases yet still suffered from severe vulnerabilities, such as the 2017 ROCA(Return of the Coppersmith Attack) flaw that compromised hundreds of thousands of ID cards [141]. This incident revealed that even sophisticated centralized or replicated systems remain exposed to systemic risks if their foundational trust anchors are flawed. In response, Estonia deployed blockchain mechanisms such as the Keyless Signature Infrastructure [142] to secure records in important sectors, including healthcare, justice, and education. In the educational domain, blockchain has been employed to safeguard academic credentials, providing tamper-proof certification and verifiable records over time.

This example illustrates that the integration of decentralization, cryptographic integrity, and network-wide consensus makes blockchain a technically robust and resilient solution for managing teacher evaluation portfolios. By providing an immutable and verifiable historical record of all actions, blockchain increases accountability, ensures transparency, and delivers a level of security and trustworthiness that conventional database systems cannot achieve. Its architecture not only preserves the integrity of credentials over time but also enables reliable verification across institutions, establishing a durable foundation for fair and transparent evaluation processes.

4.2.3. Our cost-effective blockchain solution for managing credentials and educational resources

As discussed in section 3.5, blockchain applications in education have largely focused on university degrees due to the high transaction costs on Ethereum and similar EVM networks.

We propose an efficient approach to managing costs through the use of zk-Rollups, a Layer 2 scaling solution built on Ethereum that allows multiple transactions to be aggregated into a single cryptographic proof.

Polygon zkEVM is employed as a zk-rollup solution to manage the secure transfer and verification of educational data. Operating as a Layer 2 blockchain compatible with Ethereum, Polygon zkEVM executes transactions off-chain while anchoring their security on-chain through zero-knowledge proofs [143]. Unlike Ethereum, where transactions are executed and recorded individually, Polygon zkEVM processes transactions in its own environment and periodically communicates the results to Ethereum [144]. In this system, educational resources are uploaded to IPFS, and only the relevant information that forms attestations or credentials,

BUPT

along with the corresponding IPFS content identifiers, is included in blockchain transactions as metadata. The workflow is illustrated in Figure 15 and proceeds as follows:

- An administrator submits a form through a dApp containing the relevant information for creating attestations or credentials and uploads associated educational resources to IPFS. The returned content identifiers, together with key metadata from the form, form the transaction content.
- Polygon zkEVM processes the transaction metadata and IPFS references through its sequencer, updating the off-chain ledger in an Ethereum-compatible environment [144].
- Multiple transactions are aggregated into batches to optimize throughput and reduce the number of submissions required on Ethereum [143].
- For each batch, a zk-SNARK (zero-knowledge succinct non-interactive argument of knowledge) proof is generated, cryptographically validating that all included transactions comply with Ethereum consensus rules [144].
- The batch's validity proof, along with minimal transaction calldata, is submitted to a smart contract on Ethereum, enabling verification without posting all raw transaction data.
- Once Ethereum verifies the proof, the batch is finalized with Layer 1 security guarantees, even though execution occurred off-chain.

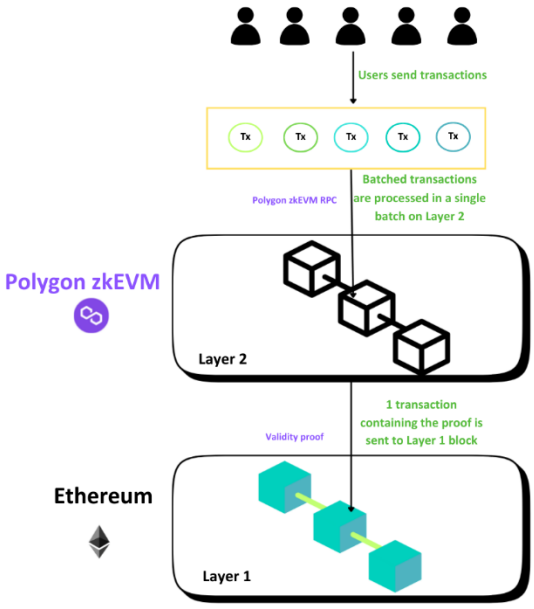


Figure 15: Transaction flow in Polygon zkEVM. Source: own illustration.

Figure 16 illustrates the functional architecture of the issuing attestations or credentials system, highlighting the interactions among teachers, academic or training institutions, and verifiers. Teachers register their participation in an activity or training course. Based on the activity or course completed, academic or training institutions generate the corresponding attestations, diplomas, or certificates. The credential information is stored on IPFS, which returns content identifiers. These content identifiers, together with relevant metadata, are recorded as transactions on Polygon zkEVM, where multiple transactions are batched and zk-

BUPT

SNARK proofs are generated. The proofs are then submitted to Ethereum, ensuring the security and immutability of the recorded data. Verifiers can consult the blockchain metadata and retrieve the corresponding attestations, diplomas, or certificates from IPFS to confirm their authenticity, while the system maintains transparency, efficiency, and scalability.

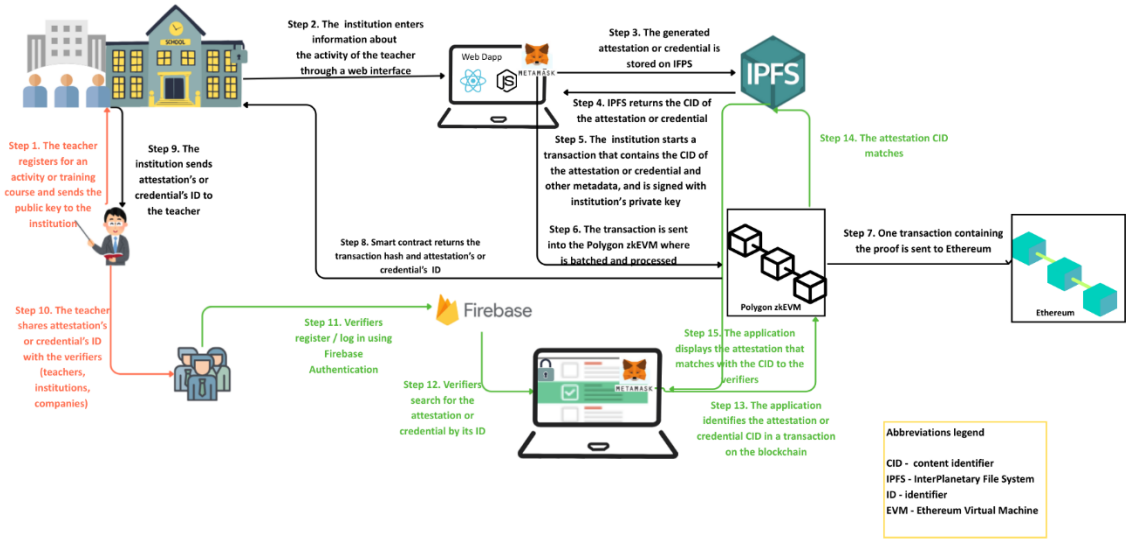


Figure 16: Overview of the proposed architecture of the issuing attestations or credentials. Source: own illustration.

To improve the process of managing teachers' documents and, implicitly, their evaluation, our proposed solution defines clear responsibilities for each stakeholder:

- *High-level educational authority*
  - Serves as the central governing entity, ensuring trust, quality, and oversight within the platform.
  - Registers educational institution as administrators through the smart contract, thereby delegating authority to manage attestations while preserving a hierarchical framework.
  - Oversees the registration of educational resources on the blockchain. After verifying the quality of a resource, the authority records it directly in the name of the teacher, embedding the teacher's public key in the transaction so that authorship and academic merit are transparently attributed.
  - Accredits institutions authorized to provide advanced training by adding them as administrators capable of issuing professional development credentials.
  - Grants teachers access rights to download and consult resources, while preventing unrestricted uploads, thereby safeguarding the repository from being diluted with low-quality materials submitted solely for evaluation points.
- *Educational institutions*
  - Act as administrators once registered by the educational authority.
  - Issue attestations for teachers' participation in activities.
  - When registering for an activity, teachers provide their public key to the educational institution, which uses it to issue the attestation. As a result,

BUPT

all attestations are permanently linked to the teacher's blockchain identity and remain in their profile even if they later change schools.
    - Retain exclusive rights to revoke only those attestations they themselves have issued, thereby ensuring accountability and preventing unauthorized interference.
- *Training institutions*
    - Operate as administrators following registration by the high-level educational authority.
    - Are responsible for issuing advanced training credentials, such as diplomas, certificates, or digital badges upon teachers' completion of professional development programs.
    - During registration, teachers submit their public key to the institution, which then uses it to issue the credential. This mechanism ensures that all diplomas from diverse training providers are consolidated into the teacher's unified blockchain profile, creating a transparent and portable record of lifelong learning.
- *Teachers*
    - Represent the primary beneficiaries of the system, with all achievements and contributions tied directly to their blockchain identity through the use of a public key.
    - Provide their public key when registering for activities at school, ensuring that attestations are issued in their name and recorded permanently in their profile.
    - Submit their public key when enrolling in professional development programs, allowing training institutions to issue diplomas and certificates that are automatically integrated into their portfolio.
    - Have their educational resources registered under their public key by the high-level authority, guaranteeing full attribution of authorship and merit.
    - In this way, teachers accumulate attestations, training credentials, and contributions to educational resources in a single, coherent, tamper-resistant, and portable portfolio, which supports fair evaluation, facilitates career mobility, and ensures lasting recognition of professional growth.
- *Verifiers*
    - Authenticate themselves by logging into our application with a username and password.
    - They check the validity of attestations or credentials using a unique identifier.
    - If they are members of an evaluation commission, they can certify the teacher's activity.

BUPT

4.3. Choosing the tools and technologies used to develop the solution

As blockchain technology continues to become more widespread in various domains, the complexity of building and managing applications on such architectures has highlighted the importance of development tool support. The following section presents the set of tools chosen to support the design and implementation of the proposed solution.

4.3.1. Hardhat, an Ethereum development environment

The initial phase of our proposed solution was developed within a local blockchain environment, allowing us to iteratively refine and optimize the smart contracts without incurring any transaction costs. To configure a temporary backend and ensure a secure environment for testing purposes, we employed Hardhat as the development framework. Hardhat is a widely used development environment within the Ethereum ecosystem, designed to simplify the process of building, testing, and deploying smart contracts and dApps [145]. It offers a comprehensive suite of tools that support the full development lifecycle, including Solidity compilation, transaction debugging, and local blockchain simulation. These capabilities make Hardhat a powerful tool for managing the complexities of blockchain development while ensuring safety and efficiency during the implementation process.

Before using Hardhat, a working installation of Node.js is necessary. We downloaded and installed the Long-Term Support (LTS) version from the official Node.js website at https://nodejs.org [146]. This process installs both Node.js and npm, providing the core tools needed to begin development. After that, Hardhat was installed using Node Package Manager (npm) with the command `npm install --save-dev hardhat`.

Once installed, we initialized a new Hardhat project by executing `npx hardhat init` in the terminal, which prompts the creation of a project workspace with a predefined structure. The initialized workspace includes the following key directories and files:

- contracts/ – directory containing Solidity smart contract source files;
- artifacts/ – folder for compiled outputs of the contracts;
- test/ – contains JavaScript test scripts utilizing Ethereum testing libraries;
- hardhat.config.js – the main configuration file that defines project settings and compiler options.

To simulate a local Ethereum network for development and testing, the command `npx hardhat node` is used. This launches a simulated blockchain environment on the local machine, preloaded with 20 test accounts funded with fake Ether, as illustrated in Figure 12. We used the first account to provide administrative control over contract implementation and interaction. This environment provides a flexible and secure space for development, enabling thorough testing and refinement before any deployment to a public blockchain [147].

BUPT

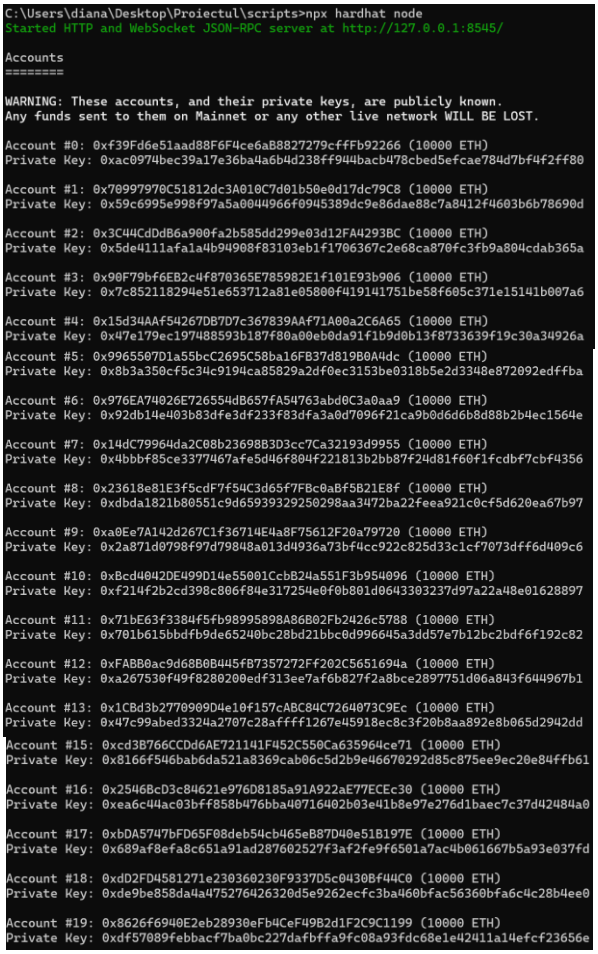


Figure 17: Local blockchain accounts. Source: screenshot from our project.

4.3.2. Sepolia, an Ethereum test network

To achieve a more accurate and realistic implementation, we moved beyond the local setup and, after thorough refinement, deployed our solution to a dedicated blockchain test network for further validation and testing. One of our objectives was to compare the behavior of smart contracts and transactions on Ethereum with their operation in a Layer 2 zk-Rollup environment using Polygon zkEVM. To facilitate this comparison, we initially employed Sepolia as a test network, which provided a controlled environment for evaluating costs, performance, and scalability before extending the analysis to the zkEVM infrastructure. Launched in 2021, Sepolia was introduced to eventually replace older testnets like Rinkeby, Ropsten, and Kovan, which have since been deprecated. In addition, Holešky, launched in September 2023, is now emerging as another principal testing environment, specifically designed to support staking, infrastructure, and protocol-level experimentation [148].

Sepolia offers a stable environment for testing smart contracts without the need to use real cryptocurrency [149]. It operates using test ether, a valueless token distributed freely through faucets, which makes it suitable for iterative testing and debugging. Sepolia supports the same EVM and core functionalities as the Mainnet [150], including compatibility with Solidity smart contracts, gas estimations, and block finality conditions. As a result, it offers a realistic and reliable setting for evaluating the correctness, performance, and security of smart contract implementations.

BUPT

4.3.3. Polygon zkEVM Cardona Testnet, a Layer 2 zk-Rollup testing environment

After conducting initial tests on Sepolia to simulate Ethereum transactions, we extended our evaluation to the Polygon zkEVM Cardona testnet. The Polygon zkEVM Cardona testnet serves as a public testing environment designed to replicate the functionality of Polygon zkEVM. It provides an Ethereum-compatible setting where smart contracts can be deployed and transactions executed under zk-Rollup conditions, without financial cost or risk. The Polygon zkEVM Cardona testnet integrates zero-knowledge proofs to ensure that batches of transactions processed off-chain can be validated and finalized on Ethereum.

Table 7 outlines key characteristics of Hardhat, the Sepolia testnet, and the Polygon zkEVM Cardona testnet, highlighting their roles, execution models, and costs features.

Characteristic	***Hardhat***	***Sepolia testnet***	***Polygon zkEVM Cardona testnet***
Role	Local testing and development	Simulates Ethereum Layer 1 environment	Simulates Ethereum Layer 2 zk-Rollup environment
Execution	Simulated blockchain on local machine	Direct execution and storage on-chain	Off-chain execution with zk-SNARK proofs anchored on Ethereum
Cost	None	Higher (Ethereum gas model with test ETH)	Lower (zk-Rollup efficiency)

Table 7: Comparison of Hardhat, Sepolia, and Polygon zkEVM Cardona test environments.

4.3.4. Metamask wallet

Interaction with the blockchain is made possible through the use of digital wallets, which serve as interfaces for managing accounts, authorizing transactions, and accessing dApps. MetaMask was selected as the primary wallet solution due to its extensive adoption within the Ethereum ecosystem [151]. MetaMask provides a user-friendly environment for creating and managing Ethereum accounts and offers robust security mechanisms. Rather than relying on cloud-based storage, MetaMask keeps account credentials under the user's direct control, significantly reducing exposure to centralized security breaches. Users can generate new accounts, import existing ones via seed phrases or private keys, and manage access across multiple devices through key export and import functionality.

MetaMask offers a suite of features that allow for the management and interaction with blockchain-based assets. These capabilities include:

- *Ether transfer* - MetaMask allows users to send, receive, and store Ether, the native cryptocurrency of the Ethereum blockchain [151]. Ether is used primarily to pay for transaction fee and serves as the medium of exchange within the Ethereum ecosystem.
- *ERC-20 token support* - The wallet supports ERC-20 tokens, which are standardized fungible tokens built on Ethereum [151]. These tokens are widely used in DeFi, governance mechanisms, and utility applications. MetaMask enables users to manage and transfer these tokens just like any other digital asset.

BUPT

- *NFT handling* - MetaMask is compatible with non-fungible tokens, including those conforming to ERC-721 and ERC-1155 standards. These tokens represent unique digital assets such as collectibles, artwork, and game items, and can be viewed and transferred within the wallet interface.
- *Transaction history monitoring* - Users can access a detailed log of all blockchain activities, including token transfers and smart contract executions, which helps ensure accountability and traceability.
- *Integration with blockchain explorers* - By linking with services like Etherscan, MetaMask provides transparent access to real-time blockchain data, including transaction status, wallet balances, and contract activity.

One of MetaMask's notable capabilities is its support for custom network configuration [152], which is especially relevant during the development and testing of smart contracts. In our work, this feature was used to connect to both the Sepolia testnet and the Polygon zkEVM Cardona testnet, allowing us to deploy contracts and execute transactions across both environments efficiently.

MetaMask features a dynamic fee estimation system that adjusts to real-time Ethereum network conditions, enabling users to optimize both transaction speed and cost [153]. It offers three predefined fee options that are Low, Market, and Aggressive, each balancing confirmation time against expense, as illustrated in Figure 18. Additionally, MetaMask allows advanced users to manually set gas and priority fees, providing greater control over transaction processing, especially during times of network congestion.

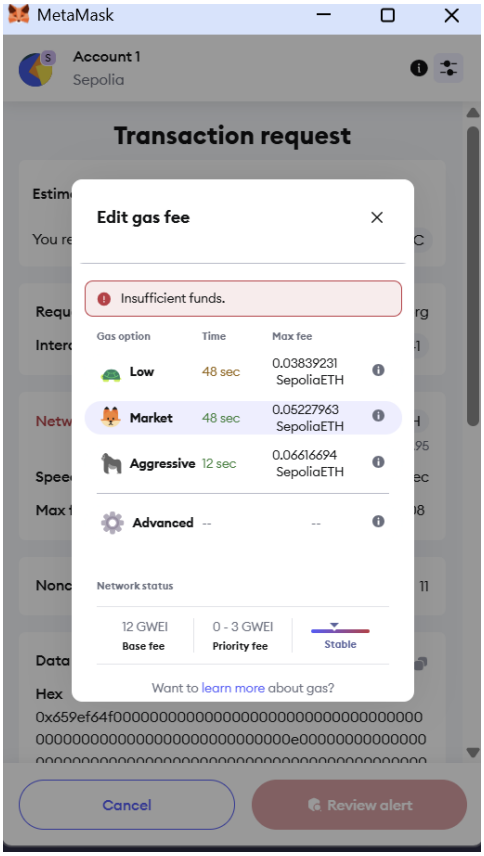


Figure 18: Estimation of gas price by MetaMask wallet

BUPT

MetaMask serves as a bridge between dApps and the Ethereum blockchain, offering seamless integration with widely used JavaScript libraries designed for web3 development. These libraries allow front-end interfaces to communicate directly with smart contracts, enabling real-time, secure blockchain interactions within web environments. Among the most notable libraries supported by MetaMask are:

- *web3.js* - A widely adopted JavaScript library that allows developers to interact with Ethereum nodes and smart contracts directly from the browser [154].
- *Ethers.js* - A lightweight and modular library offering secure and efficient tools for wallet management and smart contract communication [154].
- *web3.py* - A Python-based alternative to web3.js, providing an interface for Ethereum interaction in Python environments [154].

The features summarized in Table 8 demonstrate the rationale behind selecting MetaMask as the digital wallet integrated into our system architecture.

Characteristic	***Description***
Wallet Type	Browser extension Mobile application
Key Storage	Private keys and account credentials securely managed locally by the user
Account Management	Creation of new accounts Import/export using seed phrases or private keys
Network Compatibility	Ethereum Mainnet Test networks (e.g., Sepolia) Layer 2 solutions
Supported Assets	Ether ERC-20 tokens NFTs
Transaction History	Detailed logs of transactions Smart contract interaction records
Blockchain Explorer Integration	Integration with explorers such as Etherscan for real-time tracking and verification
Web3 Integration	Supports libraries including web3.js, ethers.js, and web3.py for dApp interaction

Table 8: Characteristics of MetaMask.

4.3.5. Alchemy, as a blockchain node provider

Although it is possible to interact with the Ethereum blockchain using only a dApp and MetaMask wallet, relying on the wallet's built-in provider to communicate directly with the test networks, this setup can be limiting. Without a dedicated infrastructure layer, transaction processing, debugging, and scalability are constrained, making the approach suitable only for basic testing or individual use. Relying solely on MetaMask's RPC provider also restricts access to advanced monitoring, logging, and performance tools that are often necessary for robust and efficient dApp development.

To ensure a more stable development environment, we integrated Alchemy as a blockchain node provider within our dApp architecture. Alchemy offers reliable and scalable access to the Ethereum network through dedicated RPC endpoints authenticated via secure API

BUPT

keys [155]. This integration enabled seamless interaction with the Sepolia testnet and Polygon zkEVM Cardona testnet and eliminated the need to manage and maintain our own nodes.

The comparison presented in Table 9 highlights the practical advantages of integrating Alchemy into our development architecture, illustrating how it improves reliability and scalability, compared to relying solely on MetaMask for blockchain interaction.

Characteristic	***Interacting with the Ethereum blockchain using MetaMask***	***Interacting with the Ethereum blockchain with Alchemy integration***
Connection method	Uses MetaMask's injected provider	Connects via dedicated RPC endpoints provided by Alchemy
Infrastructure control	No control over node behavior	Full access to a stable, scalable node infrastructure maintained by Alchemy
Transaction insight	Minimal feedback, lacks built-in logging or analytics	Provides real-time analytics, detailed logs, and debugging tools
Scalability and performance	Suitable for small-scale testing	Supports high-volume interactions with reliable throughput and uptime

Table 9: Comparison of blockchain access methods: MetaMask only vs. Alchemy integration

4.3.6. Remix IDE, development environment for Ethereum smart contract

For the deployment of our smart contracts on the Sepolia testnet and on the Polygon zkEVM Cardona testnet, we selected Remix IDE, a widely recognized web-based integrated development environment tailored for Ethereum smart contract development [156]. Remix offers a suite of tools that supports every stage of the smart contract lifecycle, from writing and compiling to debugging and deploying.

The deployment process in Remix was executed through the "Injected Provider - MetaMask" environment. This setting enabled direct interaction between Remix and the test networks through the MetaMask wallet. After compiling the smart contract within Remix, we connected our MetaMask wallet, previously configured with the Sepolia network and funded with test Ether, to the integrated development environment (IDE). Selecting the “Injected Provider” linked the wallet to the Remix interface, allowing us to deploy the contract directly onto the blockchain with a single transaction. Once confirmed through MetaMask, the deployment transaction was broadcast to the Sepolia network, and the contract address was returned, marking the successful launch of our smart contract in a decentralized test environment. We repeated this procedure for both smart contracts across the two test networks.

4.3.7. Solidity, the language for Ethereum smart contracts

As the native language of the EVM, Solidity is the primary tool for writing and deploying smart contracts on the Ethereum blockchain. Specifically designed for this purpose, Solidity is a statically typed, contract-oriented programming language that provides developers with the ability to define complex decentralized logic [157]. Its syntax draws inspiration from C++, Python, and JavaScript [157].

BUPT

One of Solidity's strengths lies in its feature set adapted for modular and secure contract design. These include inheritance for extending existing contract logic, the use of libraries to encapsulate shared functionality, and event logging to facilitate transparent interaction between contracts and user interfaces [156]. Despite its advantages, Solidity demands careful development practices due to known security challenges such as reentrancy and integer overflows [156]. As a result, rigorous testing and auditing are essential throughout the development cycle to ensure robustness and prevent vulnerabilities.

Table 10 presents a concise overview of Solidity's fundamental characteristics and its application within Ethereum smart contract development.

Characteristic	***Description***
Use Case	Primary language for writing smart contracts on Ethereum
Type of language	High-level Contract-oriented Statically typed
Compilation	Converts to EVM bytecode
Advantages	Supports modular design Code reuse via libraries Large developer community
Disadvantages	Security vulnerabilities require rigorous testing and auditing

Table 10: Characteristics of Solidity

4.3.8. Pinata, as the IPFS-based storage

Storing attestations, diplomas, certificates, digital badges created within our application, directly on a public blockchain is costly and inefficient. Instead, to optimize performance and cost-efficiency, these documents are uploaded to the IPFS, a decentralized, peer-to-peer storage protocol [158]. Unlike traditional centralized servers that retain data in a single location, thus increasing vulnerability to hardware failures, unauthorized tampering, or censorship, IPFS offers a resilient and distributed alternative. It reduces dependency on any single point of failure by fragmenting and distributing file components across a global network of nodes. Table 11 compares key characteristics of centralized servers and the IPFS.

Characteristic	***Centralized Servers***	***IPFS***
Architecture	Centralized (client-server model)	Decentralized (peer-to-peer network)
Data storage location	Stored in a single, fixed location	Distributed across multiple nodes in the network
Data retrieval method	Location-based, throuh URL or IP address	Content-based, using CID
Fault tolerance	Low, single point of failure can disrupt access	High, redundancy across nodes improves availability
Security risks	Vulnerable to centralized attacks and unauthorized access	More resistant to attacks due to decentralized nature
Scalability	Limited by server capacity and infrastructure	Scales organically as more nodes join the network
Data integrity	Must rely on server controls and checks	Ensured through cryptographic hashing of content

BUPT

Cost efficiency	Often requires costly infrastructure and maintenance	Reduces infrastructure costs through distributed resource usage
Alignment with blockchain	Less compatible with decentralized systems	Naturally complements decentralized blockchain architecture

Table 11: Comparison between centralized servers and IPFS

When a file is uploaded to IPFS, it is broken into smaller data blocks, each of which is individually hashed using cryptographic functions [159]. These hashes are then aggregated to create a unique CID, a permanent and immutable reference that is derived from the file's contents rather than its location [160]. This means that even the slightest modification to the file would result in a completely different CID, ensuring content integrity and resistance to tampering. In the context of blockchain-based attestations management, the CID, alongside relevant metadata, is recorded on the blockchain. This hybrid model guarantees both authenticity and traceability: the CID ensures that the original file can always be retrieved from IPFS, while the blockchain immutably logs its issuance and associated verification data, forming a robust foundation for secure and verifiable digital attestations.

In this project, Pinata was used as the IPFS-based storage solution to handle off-chain data efficiently [161]. To integrate Pinata into our system, we generated API keys, which provided secure, authenticated access to Pinata's services [161]. These keys allowed us to programmatically upload attestations, retrieve content using CIDs, and manage stored files without exposing sensitive credentials or relying on manual interaction, as illustrated in Figure 19.

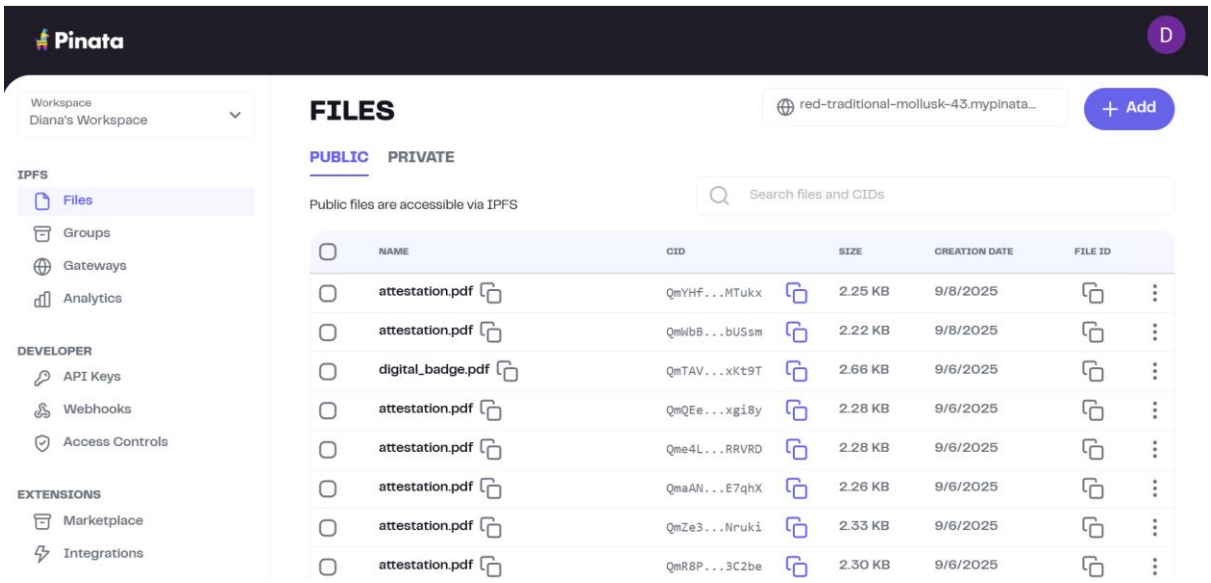


Figure 19: Pinata's interface. Source: screenshot from our Pinata workspace

4.3.9. Visual Studio Code, code editor

To develop our dApp, we used Visual Studio Code, a widely adopted, open-source code editor known for its performance, flexibility, and development tools [156]. Its intuitive interface, combined with support for extensions such as Solidity language tools, Hardhat integration, and Ethereum utilities, allowed us to efficiently write, test, and manage both smart contracts and frontend components.

BUPT

4.3.10. HTML and CSS, technologies for front-end development

As foundational technologies for front-end development, HTML and CSS serve as essential tools for constructing the visual and structural layers of our dApp. HTML provides the semantic framework that organizes content and interface elements, defining the layout, hierarchy, and flow of user interactions. CSS, in turn, governs the aesthetic presentation of these elements, controlling attributes such as color schemes, typography, spacing, and responsiveness. These technologies were carefully employed to craft a visually appealing and intuitive front-end experience, allowing users to interact with the application seamlessly. Special attention was given to layout structure, responsive design, and aesthetic coherence, ensuring that the interface not only met usability standards but also reflected the application's underlying purpose and identity.

4.3.11. Python, technology for back-end development

Python was selected to handle the application's back-end processes, including data manipulation, decision-making workflows, and integration with external systems. Its clear syntax and modular structure allowed for rapid development and efficient maintenance, making it particularly well-suited for prototyping and iterating on complex application features. Additionally, Python's compatibility with various web and blockchain libraries facilitated seamless interaction between the user interface and decentralized components. One of Python's key strengths lies in its adaptability and the availability of mature frameworks that support scalable and secure application design. Tools such as Flask and Django enable structured backend architecture [162], while libraries like web3.py provide an interface for interacting with blockchain networks [163].

4.3.12. Firebase, authentication system

Firebase, a cloud-based platform developed by Google, was integrated into our dApp to handle user authentication and identity management. As a Backend-as-a-Service (BaaS), Firebase provides a secure, scalable, and developer-friendly solution for managing user sign-up, login, and session control [164]. It supports various authentication methods, with email and password authentication being the method implemented in our system.

When a new verifier signs up, their credentials are securely transmitted to Firebase, which handles account creation and stores the necessary authentication data. Upon successful registration or login, Firebase generates a session token, which the front-end uses to verify the user's identity across subsequent interactions [164]. This token-based authentication system ensures that only authenticated verifiers can access restricted functionalities within the dApp.

BUPT

4.4. Implementation of the prototype

This section is primarily informed by our research presented in the article "Optimizing Teacher Portfolio Integrity with a Cost-Effective Smart Contract for School-Issued Teacher Documents," published in the peer-reviewed journal Computers [165].

4.4.1. User interface

In the blockchain-based credentialing system, there are three primary types of users: issuing institutions, teachers, and verifiers. All user interactions with the blockchain are facilitated through a dApp, illustrated in Figure 20, which provides an intuitive interface built using HTML and CSS for the front-end design, while Python manages the back-end logic.

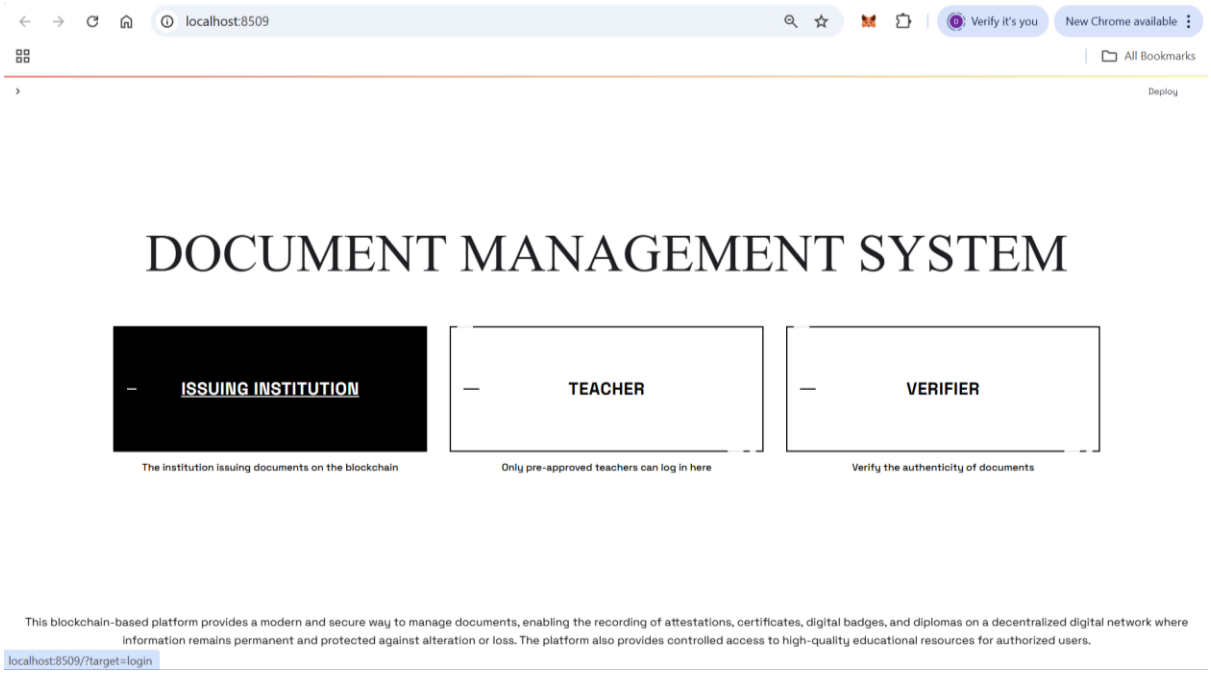


Figure 20: User interface of the developed dApp. Source: application screenshot.

4.4.1.1. Issuing institutions' dApp interactions

In the proposed dApp, the issuing process is organized under a clear hierarchical structure to ensure both regulatory oversight and operational efficiency. At the top of this structure stands the „Owner", represented by a higher authority such as the school inspectorate at the county level or, when scaled nationally, the Ministry of Education. This entity holds ultimate control over the system, guaranteeing standardization, security, and adherence to legal and institutional requirements. Complementing this role are the ‚Admins", namely schools or other authorized educational institutions, which act under the supervision of the higher authority. Their primary responsibility is to issue and, when necessary, revoke official documents granted to teachers, thereby maintaining the integrity and reliability of the credentialing process.

Figure 21 illustrates the procedural workflow used by Owner and Admins.

BUPT

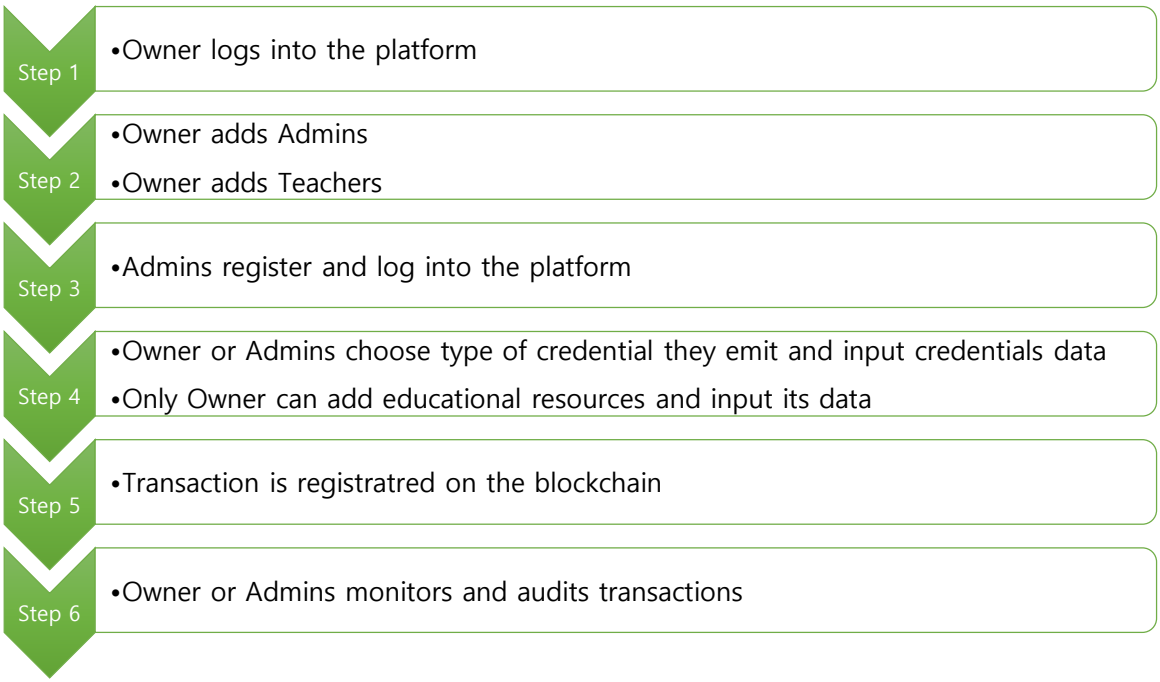


Figure 21: Issuer institution workflow diagram. Source: own illustration.

The process flow is presented to provide a clear understanding of the issuer's activities and their function within the dApp.

Step 1: Owner logs into the platform

To access the platform, the Owner must provide a username, password, and the public key of the digital wallet, as illustrated in Figure 22. Access is granted only if these credentials correspond to the information stored within the application files, otherwise, the connection is denied. The inclusion of the public key in the authentication process is essential, as it links the Owner's account to a verifiable cryptographic identity on the blockchain.

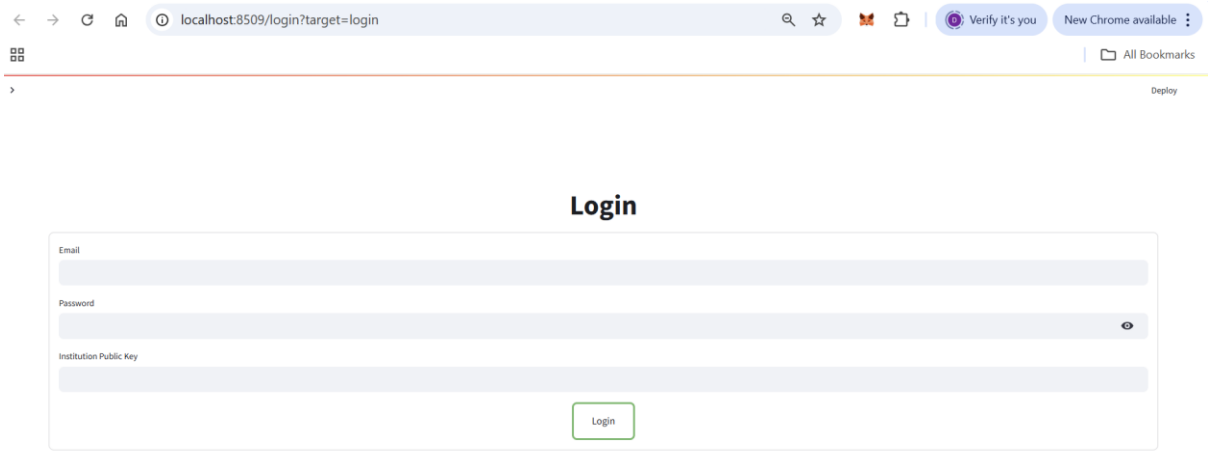


Figure 22: Owner authentication interface. Source: application screenshot.

Step 2: Owner adds Admins

The process begins with the Owner completing a form that specifies the name and public address of an institution to be designated as an Admin. Upon submission, a transaction is executed to record this information on the blockchain, as illustrated in Figure 23.

BUPT

Simultaneously, the platform logs the institution's name and address in the admins.csv file, which maintains a registry of all Admins. This registry also functions as a safeguard against duplication, as it is systematically checked before any transaction is executed to ensure that an institution is not added more than once. A similar procedure is applied for the registration of teachers on the platform, whereby each teacher's name and corresponding public key are recorded in the teachers.csv file.

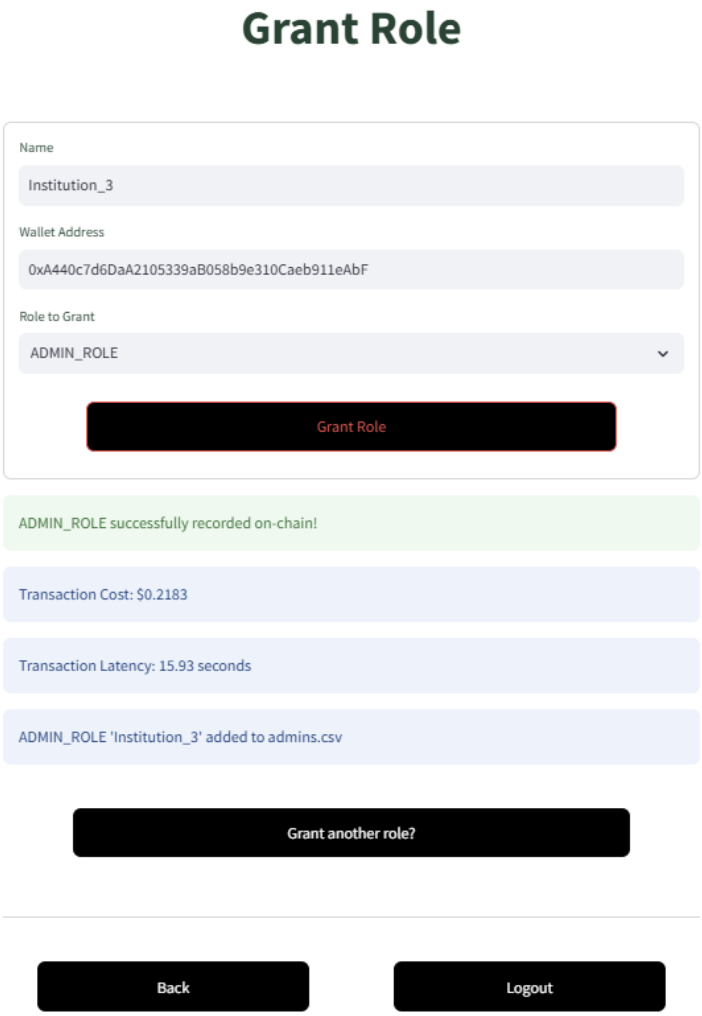


Figure 23: Role assignment interface of the application. Source: application screenshot.

Step 3: Admins register and log into the platform

Admins first register on the platform by providing the required credentials and verifying their identity. Once registered, they log into the system using secure authentication mechanisms, gaining access to the administrative interface. This step ensures that only authorized personnel can issue, monitor, or revoke credentials, maintaining the security and integrity of the credential management process.

Step 4: Owner or Admins choose type of credential they emit and input credentials data, respectively Owner can also upload educational resources

BUPT

Authorized personnel, such as Owner or Admins, initiate the credential issuance process by selecting the type of credential and completing a structured form with the recipient's information, as illustrated in Figure 24. This interface ensures that all necessary data fields are captured accurately and consistently, providing a standardized input mechanism for credential details.

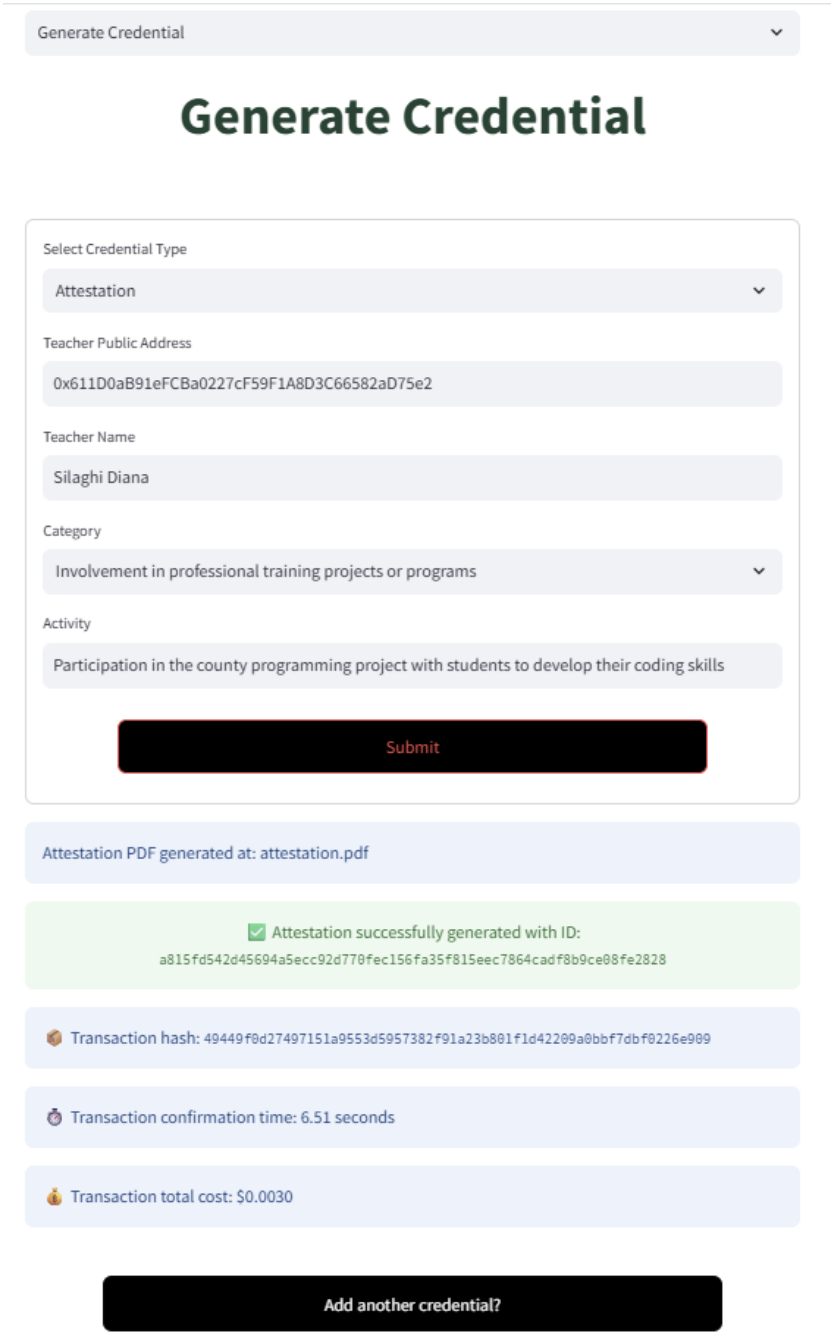


Figure 24: Credential issuance module. Source: application screenshot.

Educational resources can be added exclusively by the Owner, who uses a dedicated interface to upload the materials and provide relevant metadata, as illustrated in Figure 25.

BUPT

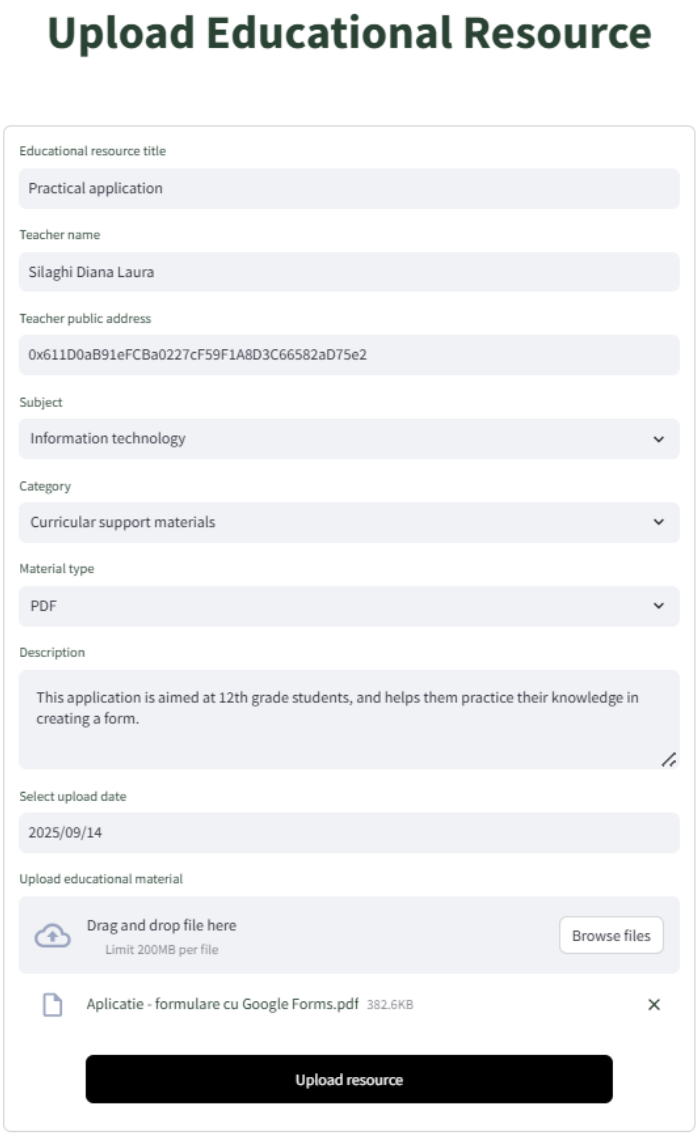


Figure 25: Uploading educational resources. Source: application screenshot.

Step 5: Transaction is registratred on the blockchain

Once the credential data is finalized, the platform transmits it to the smart contract through a REST API POST request. The API validates the submitted data, formats it into a blockchain transaction, and securely forwards it to the smart contract. The smart contract then records the credential issuance immutably on the blockchain, ensuring transparency, verifiability, and a reliable audit trail. The API also returns a transaction receipt or status confirmation to the platform, allowing administrators to verify successful registration of the credential.

Simultaneously, the credential file and its metadata are uploaded to decentralized storage via Pinata, as illustrated in Figure 26. Pinata provides persistent, tamper-resistant hosting for the credential, generating a CID that can be referenced on the blockchain. This integration ensures that the credential is not only verifiable through the smart contract but also accessible in a decentralized manner, preventing reliance on a centralized server.

BUPT

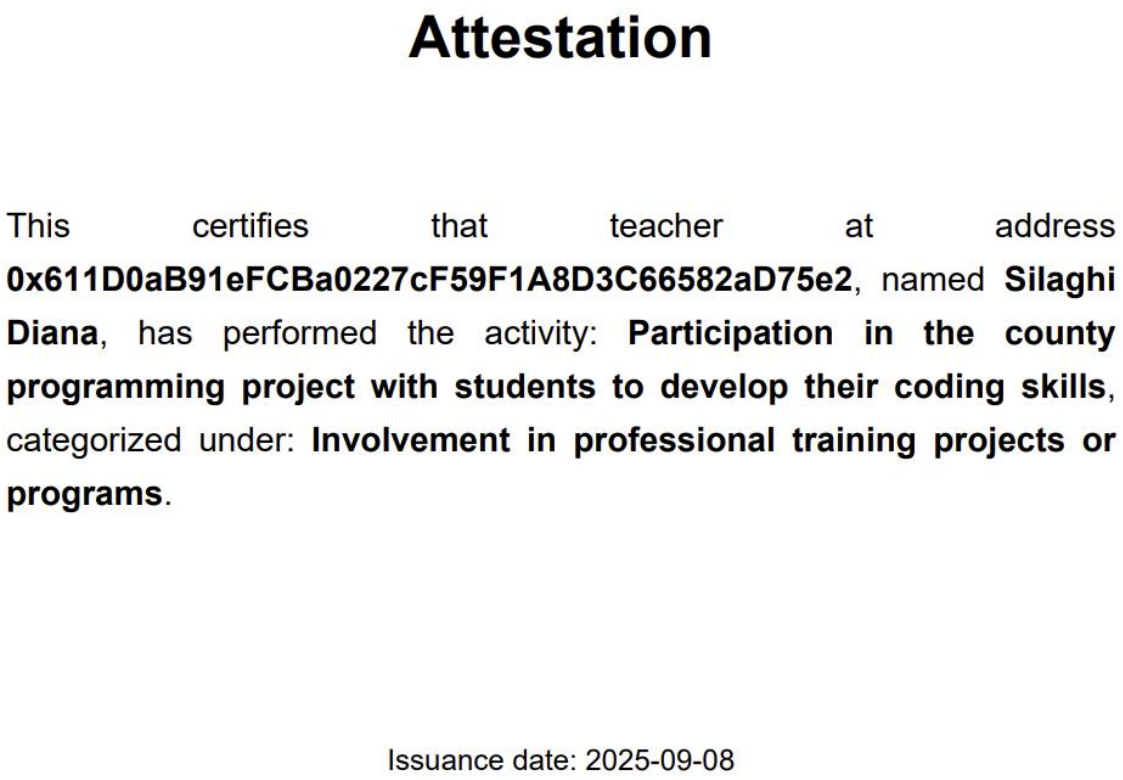


Figure 26: Credential uploaded to Pinata for secure, decentralized, and verifiable storage. Source: Pinata screenshot.

All these stages are also followed in the case of educational resources.

Step 6: Owner or Admins monitors and audits credentials

Once a credential is registered on the blockchain, owners or administrators can monitor and audit it through multiple channels. They can verify the credential's authenticity directly on the platform, ensuring the recorded information matches the intended issuance. Additionally, blockchain explorers such as Etherscan allow administrators to inspect the underlying transaction, providing full transparency and traceability of the credential's registration. If a credential needs to be revoked, owners or administrators can initiate a revocation transaction on the smart contract using the credential ID, effectively marking it as invalid on-chain while preserving the audit trail. Educational resources can be revoked in the same manner, but only by the Owner.

4.4.1.2. Teachers' dApp interaction

Teachers interact with the platform through a unified dashboard that provides visibility into all credentials issued by any institution, categorized by type. This enables them to examine credential details, monitor issuance history, and verify authenticity with ease. In parallel, the platform offers a curated repository of educational resources, allowing teachers to explore and select materials that align with their instructional objectives.

BUPT

Figure 27 illustrates the procedural workflow used by Teachers.

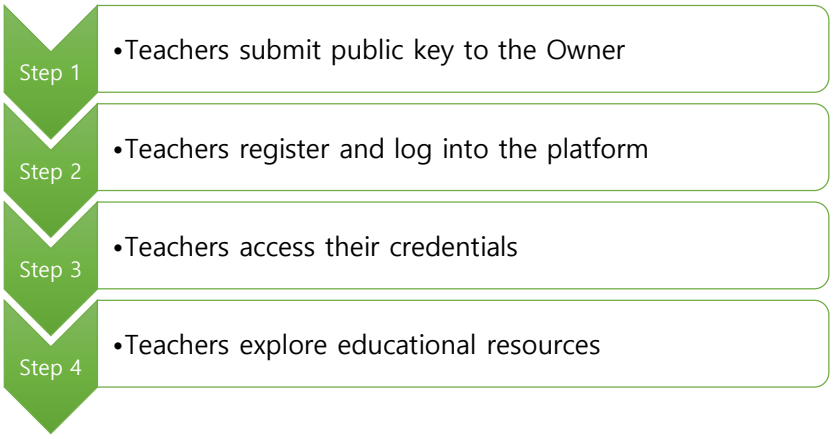


Figure 27: Teachers workflow diagram. Source: own illustration.

The process flow is presented to provide a clear understanding of the teachers' activities and their function within the dApp.

Step 1: Teachers submit public key to the Owner

Prior to account creation, teachers provide their public key to the platform Owner for registration. This ensures that each teacher can be uniquely identified and verified within the system before gaining access to platform functionalities.

Step 2: Teachers log into the platform

During login, the system checks the public key against the teachers.csv registry. This verification ensures that only authorized educators can access the platform and securely interact with features such as credential management and educational resources.

Step 3: Teachers access their credentials

After authentication, teachers access a comprehensive dashboard via the platform's API, as ilustrated in Figure 28. The dashboard displays all credentials issued by participating institutions, categorized by type. Smart contract verification confirms the authenticity of each credential, while associated metadata and files are retrieved from decentralized storage, such as Pinata. This provides teachers with a tamper-resistant, verifiable view of credentials for tracking and analysis.

BUPT

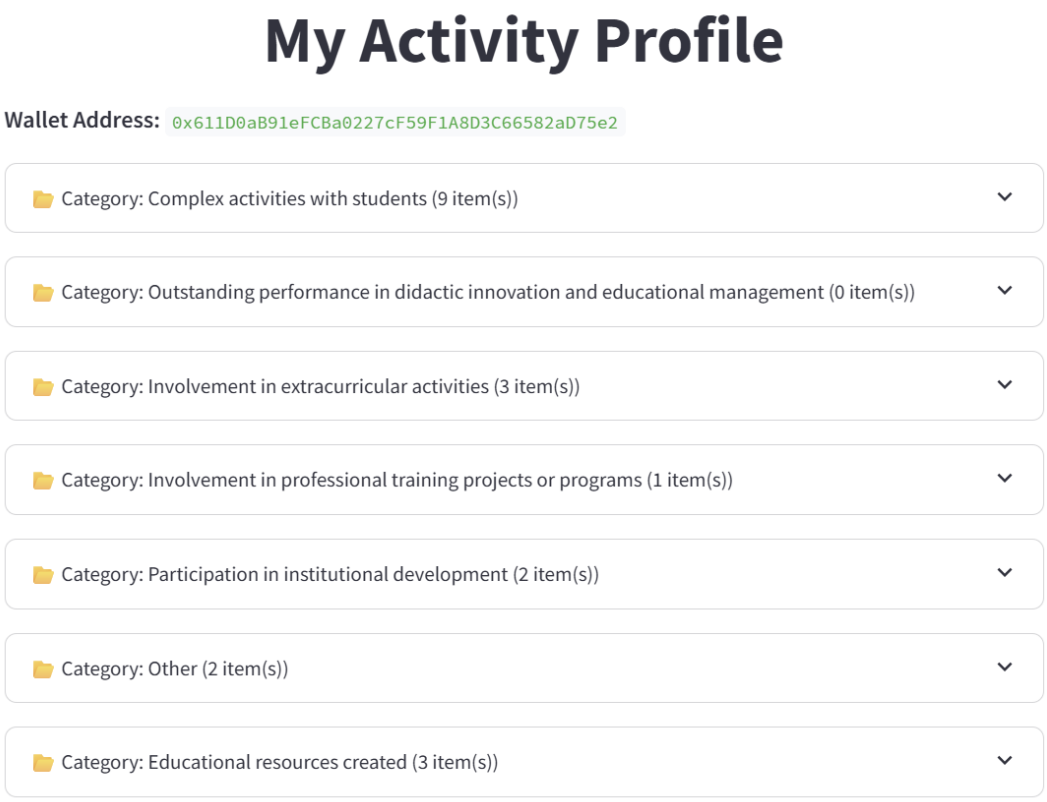


Figure 28: Teacher's credentials and educational resources within the platform. Source: application screenshot.

Step 4: Teachers explore educational resources

Teachers can browse a curated repository of educational resources maintained on decentralized storage systems. Using API-driven tools, they can filter, preview, and select materials aligned with instructional objectives.

4.4.1.3. Verifiers' dApp interaction

Verifiers are authorized users, such as teachers, companies, or educational institutions, who access the platform to confirm the authenticity of credentials and retrieve the associated documents from decentralized storage via Pinata. Figure 29 illustrates the procedural workflow followed by the verifier for accessing credentials records through the application interface and retrieving the associated documents from the IPFS network.

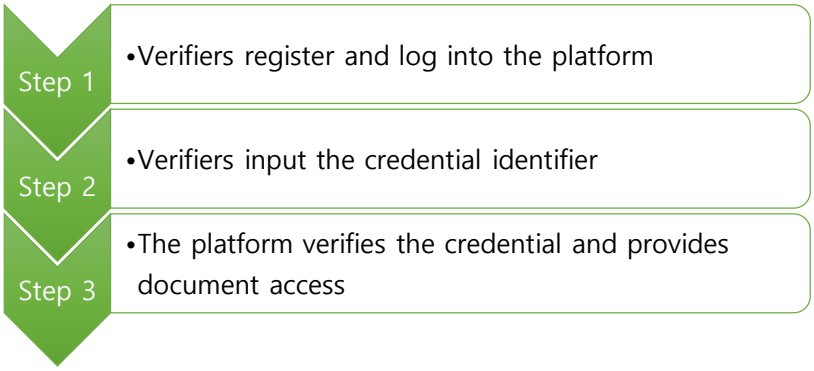


Figure 29: Verifier workflow diagram. Source: own illustration.

BUPT

The process flow is presented to provide a clear understanding of the verifiers' activities and their function within the dApp.

Step 1: Verifiers register and log into the platform

Verifiers, including teachers, companies, or educational institutions, create an account using a username and password and log in to access the verification interface securely.

Step 2: Verifiers input the credential identifier

Once authenticated, verifiers enter the unique identifier of the credential they wish to verify. The platform queries the blockchain and retrieves the credential metadata stored on decentralized storage via Pinata, ensuring tamper-resistant and verifiable access.

Step 3: The platform verifies the credential and provides document access

The platform displays whether the credential is valid or revoked. If verified, the associated credential documents are retrieved from Pinata, providing secure, decentralized access to the original records.

4.4.2. Writing and deploying the smart contracts

First, we set up our test networks so that we could deploy smart contracts. To facilitate interaction with the Sepolia test network, we configured MetaMask with the following parameters [166]:

- Network name: Sepolia testnet
- RPC URL: https://sepolia.org
- Chain ID: 11155111
- Currency Symbol: ETH
- Block Explorer URL: https://sepolia.etherscan.io

We then acquired test Ether, which is required to pay for gas fees during contract execution, by using faucets such as Google Cloud Web3. It is worth noting that some faucets, like those provided by Alchemy, have implemented anti-bot mechanisms and require a minimum balance of 0.001 ETH on the Ethereum mainnet to grant access to Sepolia test funds.

To approximate the conditions of the Ethereum Mainnet and simultaneously assess the implications of zk-Rollup scalability, we deployed and tested our contract on the Polygon zkEVM Cardona testnet. To facilitate interaction with the Polygon zkEVM Cardona network, we configured MetaMask with the following parameters:

- Network name: Polygon zkEVM Cardona testnet
- RPC URL: https://rpc.cardona.zkevm-rpc.com
- Chain ID: 2442
- Currency Symbol: ETH
- Block Explorer URL: https://cardona-zkevm.polygonscan.com

Subsequently, we acquired test Ether required to cover transaction fees for contract deployment and execution via the Triangle faucet [167]. Unlike certain faucets that impose restrictive measures, such as anti-bot verification protocols or minimum mainnet balance

BUPT

requirements, the Triangle faucet enabled direct and seamless acquisition of the necessary tokens.

4.4.2.1. Educational resources smart contract

A smart contract is used to manage the verification and recording of educational resources, eliminating reliance on trusted third parties. The following section presents the smart contract responsible for executing the verification logic and storing verifiable evidence directly on the blockchain. Appendix A contains the complete Solidity code of the contract, offering an exhaustive overview of its logic, structure, and the mechanisms employed to enable decentralized management of educational content.

Table 12 provides a structured overview of the smart contract designed for managing educational resources. It summarizes the contract's key components, including its primary functions, data structures, and access controls, as well as the specific operations used to record and verify educational materials on the blockchain.

Type	***Element***	***Description***
Data structure	EducationalResource struct	Encapsulates metadata for each educational resource, comprising: • resourceCreator: address of the teacher associated with the resource. • ipfsHash: the reference to the off-chain content stored on IPFS. • revoked: boolean flag marking whether the resource was revoked.
Storage	_resourcesById mapping	Maps a unique resource identifier to its corresponding EducationalResource struct.
	_resourcesByTeacher mapping	Maps each teacher address to an array of resource IDs they created, allowing grouped lookups per teacher.
	_existingIpfsHashes mapping	Maps the hash of an IPFS reference to a boolean value, enabling quick checks to prevent duplicate storage of the same educational resource.
Access Control	DEFAULT_ADMIN_ROLE (Owner)	Held by the contract owner, grants exclusive rights to add or revoke teachers, register new resources, and revoke existing resources.
	TEACHER_ROLE (Teacher)	Teachers' access is limited to resource viewing.
Function	addTeacher()	Grants the TEACHER_ROLE to a verified educator, enabling participation in the system.
	revokeTeacher()	Removes TEACHER_ROLE from a teacher, revoking their rights.
	addEducationalResource()	Registers a new educational resource with IPFS hash and teacher address.
	getEducationalResource()	Retrieves metadata of a resource by ID.
	revokeEducationalResource()	Allows an authorized administrator to revoke a previously issued educational resource, marking it as invalid while preserving its record on the blockchain.
	getResourcesByTeacher()	Returns all resource IDs associated with a given teacher.

BUPT

Event	EducationalResourceAdded()	Logs creation of a new resource.
	EducationalResourceRevoked()	Logs the revocation of a resource.
	TeacherAdded()	Signals addition of a teacher.
	TeacherRevoked()	Signals removal of a teacher.

Table 12: Key components and functionalities of the smart contract for educational resources management.

The smart contract was compiled using Remix, with the Solidity compiler configured to version 0.8.30 [168]. The compilation process successfully processed all contract files and confirmed their compatibility with the Shanghai EVM version, as illustrated in Figure 30. The process generated the deployable bytecode and the corresponding Application Binary Interface (ABI), providing the essential components for deploying and interacting with the contract on the test networks.

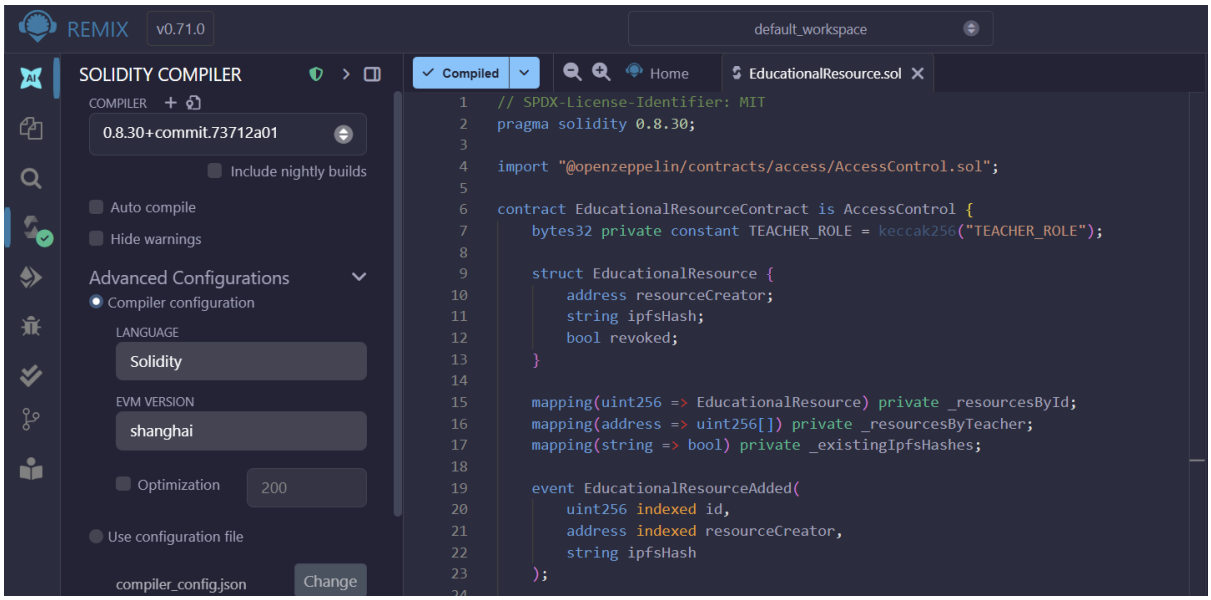


Figure 30: Compiling the smart contract for educational resources. Source: screenshot from our Remix IDE workspace

Before any interaction can occur, the Educational Resource smart contract must be deployed. Using Remix IDE with MetaMask connected first to Sepolia testnet, and after to the Polygon zkEVM Cardona testnet, the contract was deployed from our account.

Upon deployment, it received a unique blockchain address, which serves as its permanent location on Sepolia testnet [169], as illustrated in Figure 31, respectively on Polygon zkEVM Cardona testnet, as illustrated in Figure 32. All operations, including assigning and revoking teacher roles and managing educational resources, are executed through this address, ensuring that the contract's state and data are securely stored and accessible on the testnet.

```
Deploying contract with: 0x611D0aB91eFCBa0227cF59F1A8D3C66582aD75e2
Contract deployed to: 0x451cd3651d851c1af1b83019a91a914a01387e88
```

Figure 31: Educational resources smart contract deployment on Sepolia testnet.

```
Deploying contract with: 0x611D0aB91eFCBa0227cF59F1A8D3C66582aD75e2
Contract deployed to: 0xa5bf6dab83743d6f89e9b7e09ea43ac14b6a7f56
```

Figure 32: Educational resources smart contract deployment on Polygon zkEVM Cardona testnet.

BUPT

4.4.2.2. Credentials smart contract

Attestations are used to recognize activities carried out in schools and credentials are those that are awarded after completing a training course, and in our approach, their management is handled through a smart contract. Under this model, a higher authority, such as school inspectorates or the Ministry of Education, would assign administrative privileges to institutions, enabling them to issue or revoke credentials. Appendix B provides the complete Solidity code of the credential contract, outlining its logic, structure, and mechanisms for decentralized credential management.

The specific functions implemented within the smart contract, along with their roles in managing and verifying credentials, are analyzed in Table 13.

Type	*Element*	*Description*
Data structure	Credential struct	Represents a credential issued by an institution. Contains: • issuer_institution: address of the institution that created the record. • receiver: address of the individual receiving the credential. • ipfsHash: the reference to the off-chain content stored on IPFS. • credentialDate: timestamp of issuance. • revoked: boolean flag marking whether the credential was revoked.
Storage	credentials mapping	Maps a unique sequential ID to its corresponding credential. Ensures each certificate is uniquely indexed.
	_credentialsByReceiver mapping	Associates each receiver with a list of their credential IDs, enabling quick reconstruction of a user's complete history.
	_existingIpfsHashes mapping	Maps the hash of an IPFS reference to a boolean value, enabling quick checks to prevent duplicate storage of the same credential.
Access Control	DEFAULT_ADMIN_ROLE (Owner)	Assigned to the contract owner, granting exclusive authority to delegate or revoke administrative rights, as well as to issue and revoke credentials.
	_ADMIN_ROLE (Admin)	Institutions to issue and revoke credentials.
	General users (no role)	Non-privileged accounts limited to viewing and verifying credentials.
Function	addAdmin()	Assigns administrative privileges to a new address
	revokeAdmin()	Revokes administrative rights from an existing address
	generateCredential()	Creates a validated credential for a receiver with metadata stored on IPFS.
	revokeCredential()	Enables an authorized administrator to revoke a previously issued credential, ensuring it is marked as invalid while its record remains preserved on the blockchain.
	getCredential()	Returns credential details by identifier
	getReceiverCredentials()	Retrieves all credentials linked to a receiver.
	isVerified()	Verifies existence of a credential ID
Event	CredentialGenerated()	Logs creation of a new credential and metadata

BUPT

CredentialRevoked()	Logs invalidation of an credential
AdminAdded()	Logs assignment of administrative privileges
AdminRevoked()	Logs removal of administrative privileges

Table 13: Key components and functionalities of the smart contract for attestations management.

As we did in the educational resources smart contract, the credential smart contract was compiled in Remix using Solidity 0.8.30, as illustrated in Figure 33. The process confirmed compatibility with the Shanghai EVM and produced the bytecode and ABI, enabling deployment and interaction on the test networks.

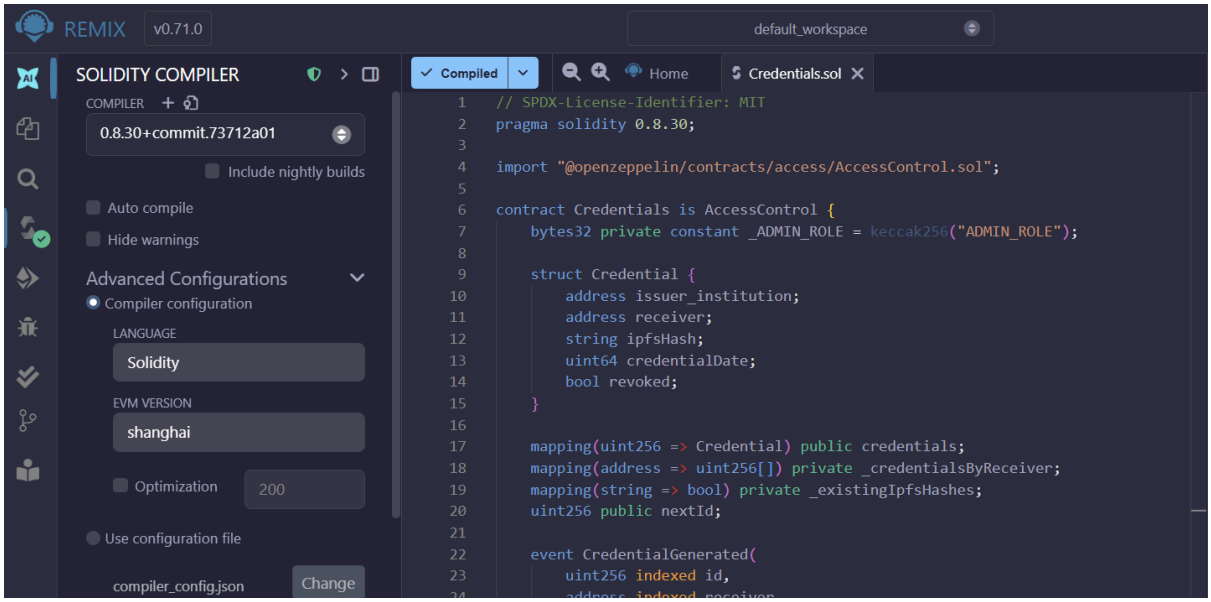


Figure 33: Compiling the smart contract for credentials management. Source: screenshot from our Remix IDE workspace.

Using Remix IDE with MetaMask connected first to Sepolia testnet, and after to the Polygon zkEVM Cardona testnet, the contract was deployed from our account, receiving a unique blockchain address as its permanent location on the testnet, as shown in Figure 34 for the contract deployed on Sepolia testnet, respectively as shown in Figure 35 for the contract deployed on Polygon zkEVM Cardona testnet.

```
Deploying contract with: 0x611D0aB91eFCBa0227cF59F1A8D3C66582aD75e2
Contract deployed to: 0xea693681fac5370d36a5c769e4ab1f4387d190d8
```

Figure 34: Credentials smart contract deployment on Sepolia testnet.

```
Deploying contract with: 0x611D0aB91eFCBa0227cF59F1A8D3C66582aD75e2
Contract deployed to: 0x822a4fe9a23575bb9120c9069effdd7698be76a6
```

Figure 35: Credentials smart contract deployment on Polygon zkEVM Cardona testnet.

4.4.3. Modules interaction

This section outlines the end-to-end workflow of the proposed platform, detailing how users interact with the system, how data is processed and stored, and how verifiable credentials or educational resources are managed on a decentralized network. The workflow integrates modern frontend frameworks, decentralized storage solutions, and blockchain technologies to ensure secure, efficient, and transparent operations. Beginning with user interaction through

BUPT

React.js and Next.js [170], the process progresses through metadata preparation, IPFS storage via Pinata, blockchain transaction construction and submission using Web3.js [171], and concludes with real-time event handling, verification, and document retrieval. Each step highlights the interplay between frontend responsiveness, backend security, decentralized data integrity, and blockchain verification, demonstrating the platform's ability to provide tamper-resistant, verifiable educational credentials and resources.

Step 1: User interaction through React.js and Next.js

Owner or admins begin the process by interacting with the platform via React.js components. Using the Next.js framework, the platform provides secure, responsive forms for creating credentials or uploading educational resources. Next.js manages page routing and server-side processing [172], ensuring that sensitive operations, such as data preparation for blockchain storage, are handled securely and efficiently while maintaining a seamless user experience.

Step 2: Metadata preparation and upload to Pinata

After completing the input forms, the platform prepares the credential or resource metadata for decentralized storage. Through Next.js API routes, this metadata and associated files are uploaded to Pinata [173], which hosts them on the IPFS network. Pinata generates a permanent, tamper-resistant reference to the stored content, ensuring decentralized accessibility and data integrity.

Step 3: Web3.js transaction construction

With the IPFS reference available, the frontend calls the Web3.js module, importing the smart contract ABI from the JSON artifact [174]. Web3.js constructs a transaction containing the credential or resource information, including the IPFS hash, issuer address, and relevant metadata. This transaction is prepared to interact with the smart contract functions, such as generateCredential or addEducationalResource.

Step 4: Transaction submission via Alchemy node

At this stage, all prepared transactions, such as attestations issued to groups of teachers or batches of academic credentials, are aggregated off-chain using zk-Rollups. A zk-Rollup is is a Layer-2 scaling solution on Ethereum. Transactions are initially collected and stored temporarily within the zk-Rollup layer, which operates off the Ethereum Mainnet. Within this layer, a cryptographic zero-knowledge proof is generated, mathematically verifying that all included transactions are valid, properly formatted, and comply with the rules defined in the smart contract [175], without requiring each individual transaction to be recorded directly on Ethereum. Once the zero-knowledge proof is generated, it is submitted to Ethereum via an Alchemy node. The smart contract on Ethereum then verifies the proof, ensuring the integrity of all aggregated transactions. Notably, Ethereum stores only the essential metadata and the IPFS references for each credential or resource, while the full transaction data remains securely off-chain within the zk-Rollup [176]. This mechanism guarantees that all records are tamper-resistant, significantly reduces gas fees, and prevents congestion on the Ethereum blockchain, while maintaining full verifiability and security of the system.

BUPT

Step 5: Smart contract event emission

Upon successful execution, the smart contract emits events such as CredentialGenerated or EducationalResourceAdded. These events provide verifiable confirmation that the transaction has been executed correctly, allowing the frontend to remain synchronized with the blockchain state.

Step 6: Event capture and user interface (UI) update

The frontend, through Web3.js event listeners, captures these smart contract events in real-time. The platform updates dashboards, displays confirmation messages, and reflects the latest credential or resource status, providing transparency and immediate feedback to users while ensuring that the UI accurately represents the blockchain state.

Step 7: Verification and document retrieval

Finally, verifiers or teachers access credentials or resources by querying the smart contract through Web3.js and the Alchemy node. The smart contract returns the verification status that can be valid, revoked, or nonexistent, and the IPFS reference. The platform retrieves the full document from Pinata and displays it securely through the React.js interface, providing tamper-resistant, decentralized, and verifiable access to credential or resource files.

4.5. Summary

In this chapter, we presented a comprehensive blockchain-based solution designed to manage teacher attestations, certificates, diplomas, digital badges, and educational resources.

We described the application's architecture, emphasizing user interactions across different roles: teachers register their participation in activities or training courses, academic or training institutions generate the corresponding credentials, and verifiers authenticate and, when authorized, certify teacher activities.

The implementation section detailed the deployment and execution of smart contracts on test networks, including Sepolia for Ethereum Layer 1 simulations and Polygon zkEVM Cardona for Layer 2 zk-Rollup operations. Credential metadata and educational resources, as well as document identifiers are stored on IPFS to optimize cost and performance, while transactions containing these metadata are batched on Polygon zkEVM and secured with zk-SNARK proofs before anchoring on Ethereum, ensuring data integrity and tamper resistance.

We also described the module interactions, demonstrating how the system coordinates between user interfaces, smart contracts, and storage layers to maintain a seamless and secure workflow. By combining off-chain storage, Layer 2 scalability solutions, and on-chain verification, our proposed system offers an efficient, reliable, and verifiable framework for managing credentials and educational resources, supporting both operational practicality and security.

BUPT

5. Experimental results

5.1. System performance

5.1.1. Gas consumption evaluation

Within Ethereum and other blockchains based on the EVM, the execution of any transaction or contract call requires the expenditure of gas, a unit that quantifies computational work [13]. The total gas consumed reflects the algorithmic complexity and resource demands of the executed operation. Among the various categories of operations, persistent data storage constitutes one of the most resource-intensive. In particular, committing a single 32-byte word to the blockchain's state incurs a cost on the order of 20000 gas units, making storage a dominant factor in transaction fees [177]. By contrast, event logging are less costly than permanent state changes, as the data is recorded in transaction receipts rather than the contract's state, requiring roughly 375 gas units plus an additional cost per byte of data recorded.

This pricing asymmetry illustrates the deliberate economic design of Ethereum's gas model: operations that permanently alter the blockchain's state are strongly disincentivized through higher costs, while transient computations and event emissions are encouraged as more efficient alternatives. Consequently, developers are incentivized to minimize on-chain storage and instead employ strategies such as event emissions or off-chain indexing to reduce transaction fees. This consideration was central to the architectural choices underpinning our contract. In designing the EducationalResource structure, we deliberately limited persistent storage to only the most essential attributes: the creator's address, a hashed representation of the IPFS content identifier, and a Boolean flag for revocation status. Each of these fields is lightweight in terms of gas usage, with the hash itself stored as a fixed-size bytes32 rather than a variable-length string, thereby avoiding the high cost of dynamic storage. More descriptive metadata, such as titles, categories, or full content, was intentionally excluded from the blockchain's state and instead delegated to off-chain storage systems like IPFS, retrievable via the recorded hash. In this way, the design reflects a conscious trade-off between immutability and efficiency, ensuring verifiability on-chain while maintaining cost-effectiveness in practice.

Gas consumed by educational resources smart contract

We evaluated the contract's efficiency using a custom Python script, which use Web3.py together with the EthereumTester backend to simulate transactions and capture the exact gas used values. This tool allows developers to obtain a precise breakdown of the computational costs associated with different contract operations in a controlled environment, without incurring real costs on a real network. The results, illustrated in Figure 36, reveal a clear differentiation across functions: adding a teacher consumed 52973 gas, revoking a teacher only 31397 gas, while adding an educational resource required 188777 gas, and revoking it 51014 gas.

BUPT

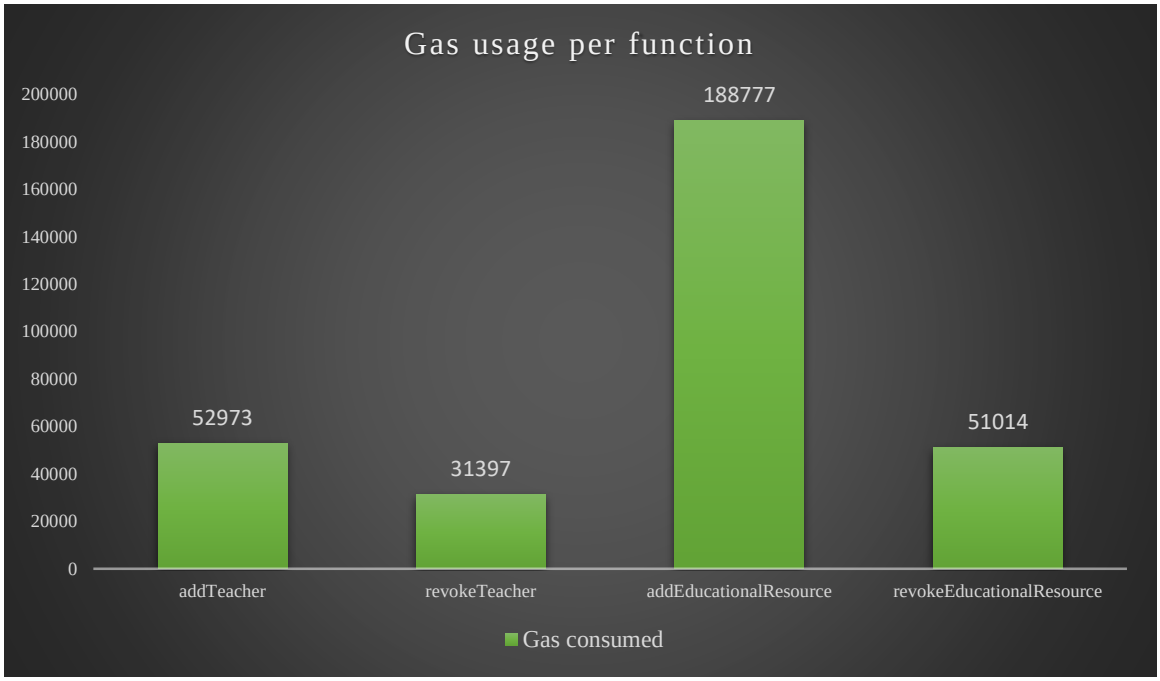


Figure 36: Gas consumed with different educational resources contract operations

Following an initial assessment of the gas consumption for a single transaction, we conducted a batch of ten simultaneous transactions to evaluate the consistency of gas usage and to assess the scalability of our solution. The gas consumption data for adding and revoking ten teachers and educational resources exhibits high consistency as illustrated in Figure 37, with only minor deviations observed. Key observations include:

- addTeacher: Gas consumption is mostly 52973, with a slightly lower value of 52961 for the second transaction.
- revokeTeacher: Gas usage is generally 31397, but the second transaction dips to 31385. The minor reduction could be due to optimized storage access after the initial mapping update.
- addEducationalResource: Gas usage stabilizes at 188765, with a slightly lower 188753 for the second transaction. The consistency indicates deterministic cost for creating a new struct and updating mappings, while small variations again suggest cold versus warm storage effects.
- revokeEducationalResource: Gas usage is mostly 51014, with the first and last transactions at 51002. This indicates predictable costs for flipping the revoked boolean, with slight deviations likely caused by storage slot caching and gas refund mechanics.

BUPT

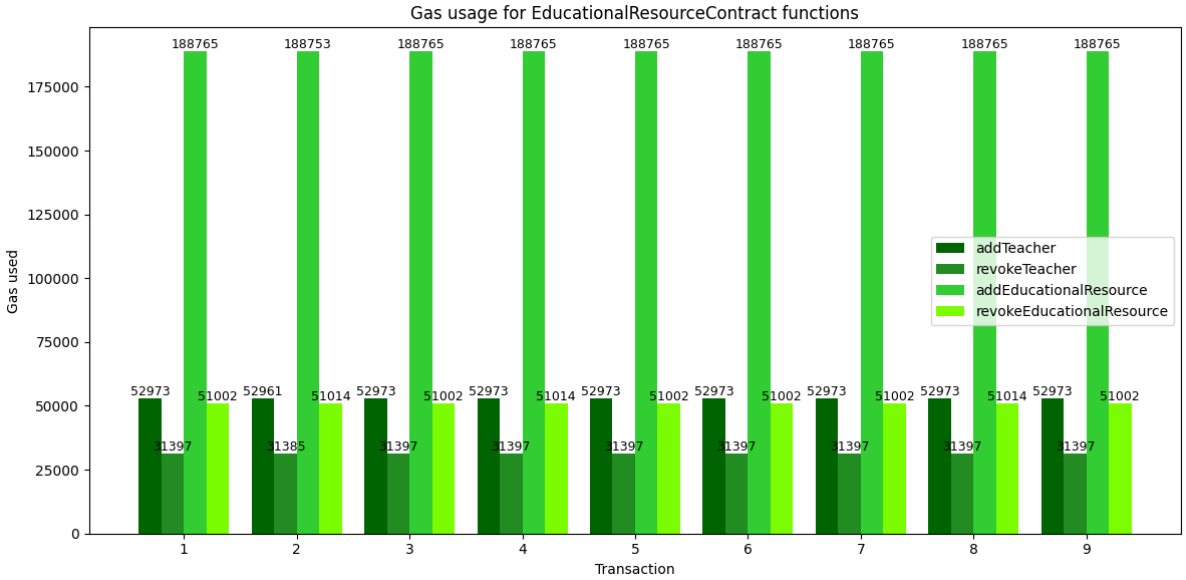


Figure 37: Gas consumed with different contract operations within 9 transactions.

After the evaluation of the gas consumption for a single transaction and for a batch of ten transactions involving the addition and revocation of teachers and educational, to further examine scalability, we extended the experiment by executing 1000 consecutive addEducationalResource() transactions. This larger test confirmed that, after a slightly higher cost during the initial insertions, gas usage stabilized and remained consistent across all subsequent operations, demonstrating predictable performance even at scale, as shown in Figure 38.

```
Gas usage statistics for 1000 addEducationalResource() transactions
   Average gas: 171670.1
   Min gas: 171653
   Max gas: 188753
   Average gas (resources 1-100): 171824
   Average gas (resources 101-200): 171653
   Average gas (resources 201-300): 171653
   Average gas (resources 301-400): 171653
   Average gas (resources 401-500): 171653
   Average gas (resources 501-600): 171653
   Average gas (resources 601-700): 171653
   Average gas (resources 701-800): 171653
   Average gas (resources 801-900): 171653
   Average gas (resources 901-1000): 171653
```

Figure 38: Gas consumed with adding educational resources within 1000 transactions.

The gas analysis for adding 1000 educational resources shows that transaction costs are highly consistent, with only a small deviation observed in the first 100 insertions. The overall average gas per resource is 171670, while the minimum and maximum values range narrowly between 171653 and 188753. After the initial batch (resources 1 - 100), where the average was slightly higher at 171824 due to initial storage and state updates in the contract, the gas usage stabilizes at a constant 171653 for all subsequent intervals. This indicates that the contract reaches a steady operational cost once the first resource insertions establish baseline mappings and storage patterns. The minimal variation suggests strong scalability, as the marginal cost of adding new resources remains predictable and uniform even at higher volumes.

BUPT

To better illustrate the distribution of gas usage across 1000 addEducationalResource() transactions, the data were represented as chart in Figure 39. Each transaction's gas consumption is shown alongside the overall average value of 171670, providing a clear visual comparison and highlighting the minor deviations, particularly during the initial insertions.

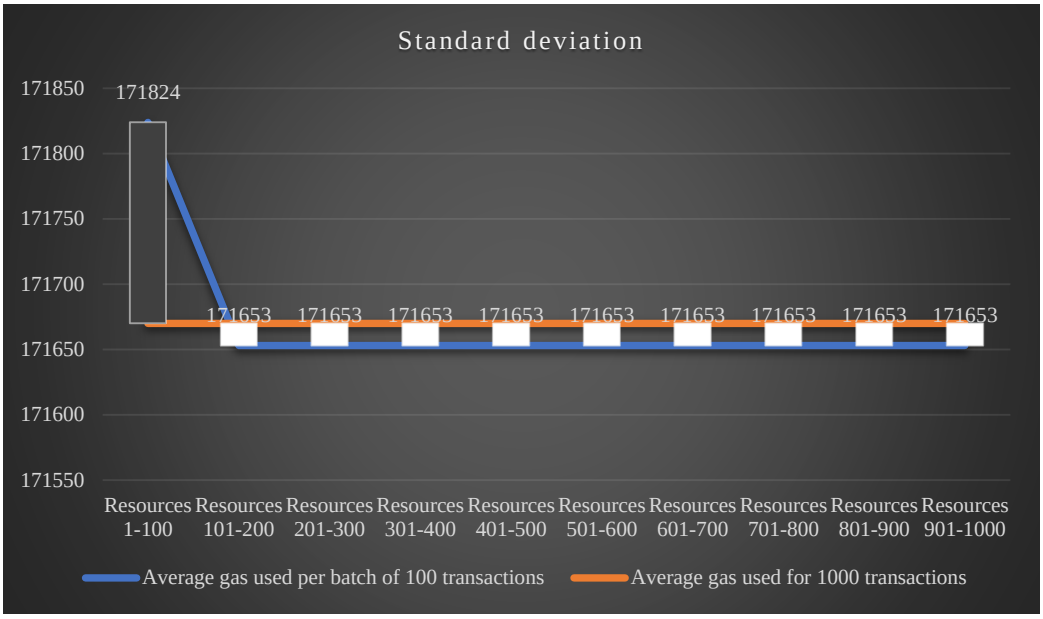


Figure 39: Gas used compared to the average value in educational resource contract.

Gas consumed by credential smart contract

After we completed the gas evaluation for the educational resource contract, we applied the same methodology to the credential smart contract to obtain comparable metrics for all key functions.

We began the evaluation of the credential smart contract by executing a single transaction for each key function to measure gas consumption. Administrative operations proved relatively inexpensive: adding an admin consumed 50223 gas, while revoking an admin used 28747 gas, reflecting the lower complexity and minimal storage changes involved in role management. Generating a credential was considerably more costly at 253401 gas, as this function creates a new struct, updates mappings, and emits an event, all of which require significant computational and storage resources. Revoking a credential required 33027 gas, a moderate cost since it only updates a boolean flag and emits an event. To visually highlight these differences we created a plot, Figure 40, showing that operations creating new on-chain data dominate gas usage, whereas administrative and revocation tasks are comparatively lightweight.

BUPT

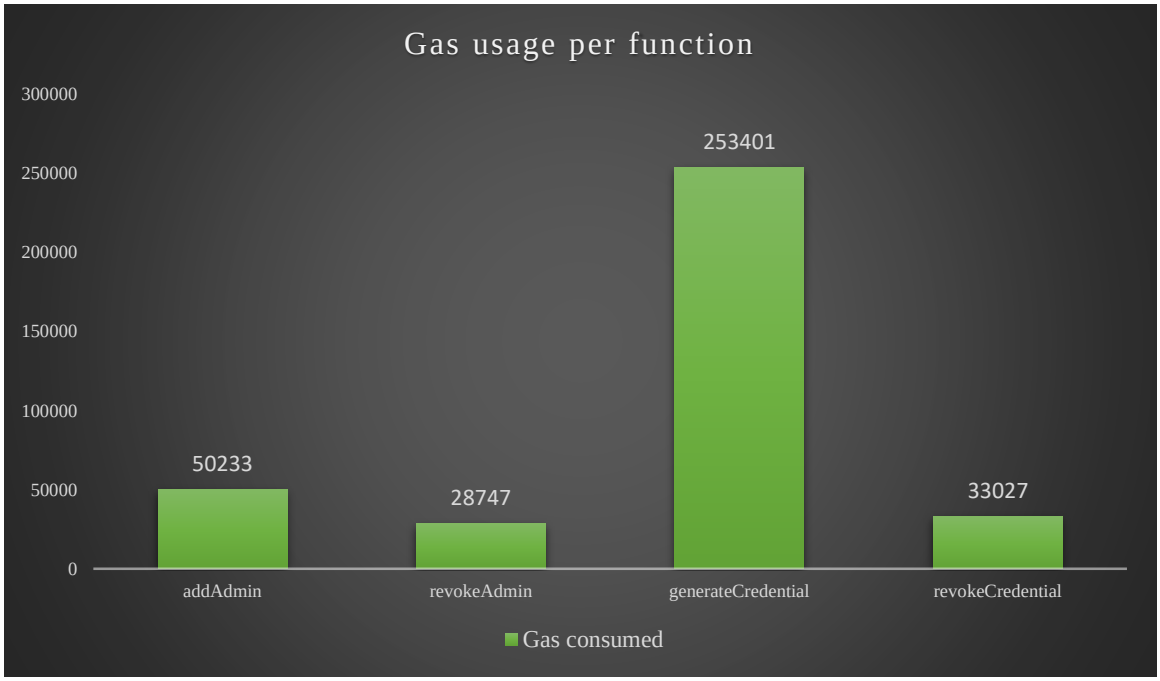


Figure 40: Gas consumed with different credential contract operations

We began by measuring gas usage for a single transaction and then ran nine consecutive transactions to evaluate performance consistency. The results, presented in Figure 41, show that adding and removing admins, as well as generating and revoking credentials, requires largely stable gas, with only minor fluctuations. Specifically:

- addAdmin: Typically consumes 50233 gas, with one slightly lower measurement at 50221.
- revokeAdmin: Mostly 28747 gas, with a small deviation at 28735, reflecting predictable updates to contract state.
- generateCredential: Generally uses 236289 gas, with the first and second transactions at 253389 and 236277, suggesting minor variations when writing new data to the blockchain.
- revokeCredential: Consistently requires 33027 gas for all transactions, indicating highly stable operation.

U#	AddAdmin	RevokeAdmin	GenerateCred	RevokeCred
1	50233	28747	253389	33027
2	50221	28735	236277	33027
3	50233	28747	236289	33027
4	50233	28747	236289	33027
5	50233	28747	236289	33027
6	50233	28747	236289	33027
7	50233	28747	236289	33027
8	50233	28747	236289	33027
9	50233	28747	236289	33027

Figure 41: Gas consumption for credential contract functions over nine transactions

Overall, these results demonstrate that the functions behave predictably and efficiently, making gas costs reliable even with multiple consecutive operations.

BUPT

Figure 42 presents the gas usage for each function in the credential contract over nine transactions, showing typical values and any minor variations.

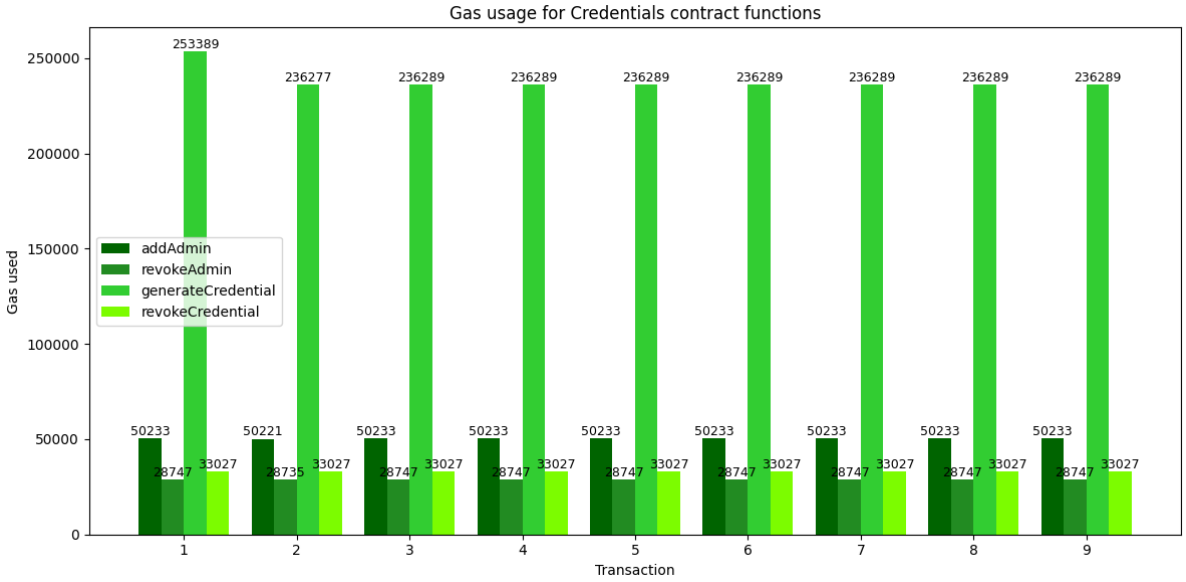


Figure 42: Gas consumed with different credential contract operations within 9 transactions.

Following the approach used in testing the Educational Resources contract, we simulate 1000 transactions to analyze performance under high usage. Figure 43 illustrates the gas usage for 1000 generateCredential() transactions, demonstrating consistent and stable performance. The average gas per transaction is 219271, with a minimum of 219237 and a maximum of 252437. After the first 100 transactions, which slightly exceeded the overall average, the gas usage stabilizes at 219237 per transaction, indicating efficient and predictable scalability for repeated credential generation.

```
Gas usage statistics for 1000 generateCredential() transactions:
   Average gas: 219271.2
   Min gas: 219237
   Max gas: 253437
   Average gas (credentials 1-100): 219579
   Average gas (credentials 101-200): 219237
   Average gas (credentials 201-300): 219237
   Average gas (credentials 301-400): 219237
   Average gas (credentials 401-500): 219237
   Average gas (credentials 501-600): 219237
   Average gas (credentials 601-700): 219237
   Average gas (credentials 701-800): 219237
   Average gas (credentials 801-900): 219237
   Average gas (credentials 901-1000): 219237
```

Figure 43: Gas consumed with generating credential within 1000 transactions.

Figure 44 depicts the gas consumption for 1000 generateCredential() transactions. The plot illustrates individual transaction costs relative to the mean value of 219271, highlighting deviations from the average, particularly during the initial credential generations, thereby providing a quantitative perspective on the variability of resource utilization.

BUPT

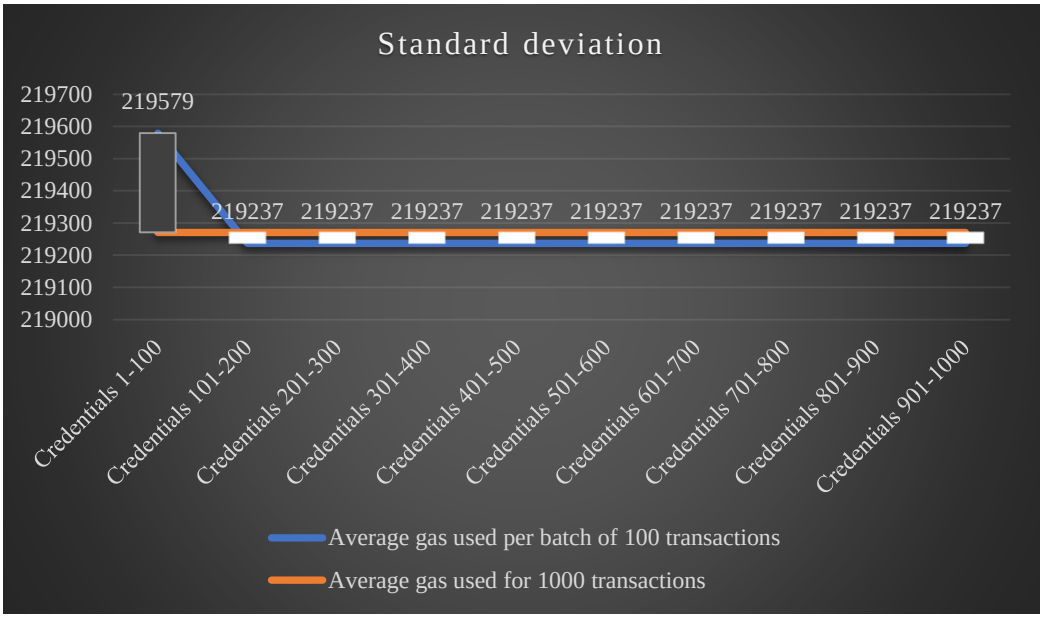


Figure 44: Gas used compared to the average value in educational resource contract

To highlight the effectiveness of gas consumption, comparative experiments were conducted between an initial contract with a detailed data structure and the optimized contract adopted in the proposed solution. In the full version, the credential stored on-chain included attributes such as the teacher's public key, name, category, activity description, IPFS hash, issuance date, and issuer institution. This extensive data structure led to considerably higher gas usage per transaction, as string fields such as teacher name, category, and activity were retained directly on-chain. This presented a major limitation, particularly considering the intended implementation within educational and training institutions, where attestations and credentials are issued in large volumes and financial resources are constrained. By contrast, the optimized contract was restructured to exclude these string fields and retain only essential elements, namely the teacher's public key, the IPFS hash of the credential, the issuance date, and the issuer institution, resulting in a substantial decrease in gas consumption.

The comparative experimental results further emphasize this improvement and are illustrated in Figure 45. In the initial contract, gas consumption for generating credentials ranged between 441463 and 470092 units, with variations caused by the different lengths of the strings stored on-chain. In the optimized version, gas usage stabilized around 236289 units, independent of the number of credentials issued. These results demonstrate the efficiency gained by eliminating string-heavy fields and relying on IPFS for storing the complete descriptive data. By recording only lightweight identifiers on the blockchain and maintaining the full information off-chain, the optimized design ensures cost-effectiveness, scalability, and reliable verification of credential details, while avoiding unnecessary overhead in blockchain operations.

BUPT

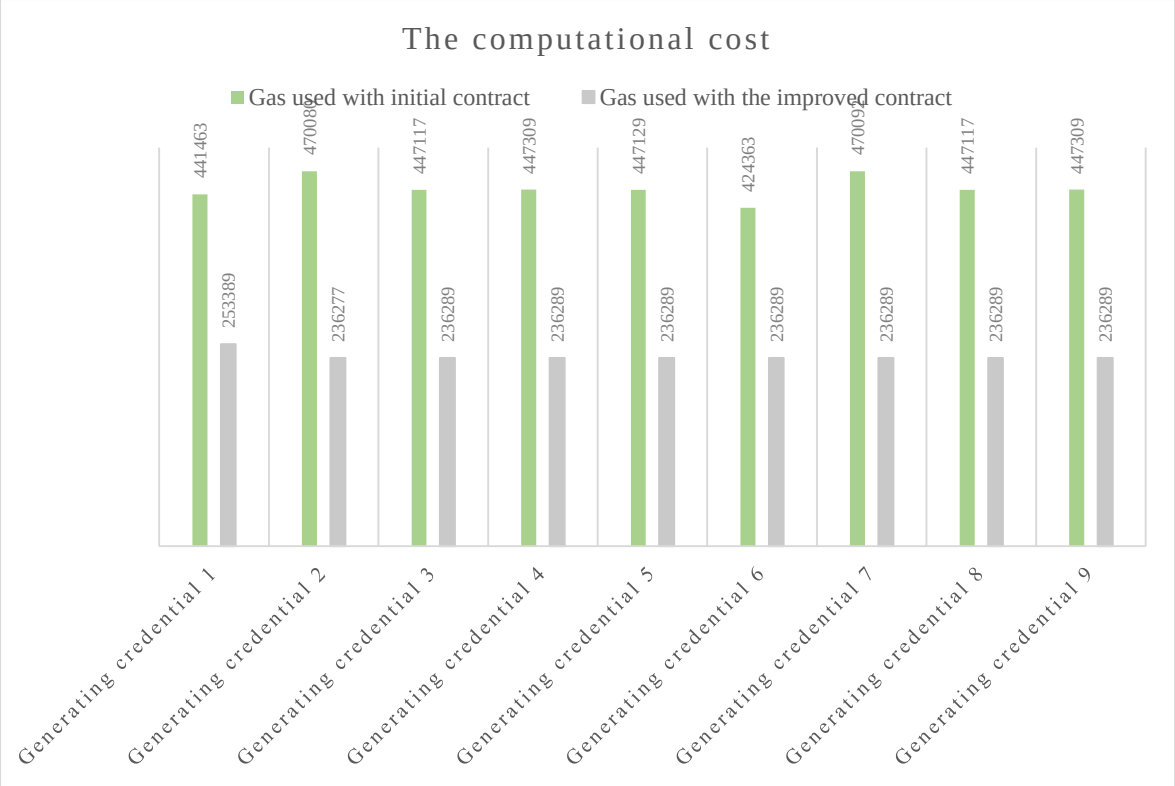


Figure 45: Comparative gas consumption for credential generation between initial full on-chain contract and optimized contract with IPFS off-chain storage

There are very few solutions that utilize zkRollups or other Layer 2 approaches for issuing certificates, and the available research on such implementations is limited, with notable studies emerging as recently as 2025. In this context, our solution demonstrates competitive performance, as shown in Figure 46, consuming 253401 gas per transaction, compared to 282132 for BlockMEDC [133] and 226858 for ZKBAR-V [143]. When tested at scale, our system maintained an average gas usage of 219271 over 1000 transactions, indicating its efficiency and positioning it alongside other recent Layer 2 implementations in certificate issuance.

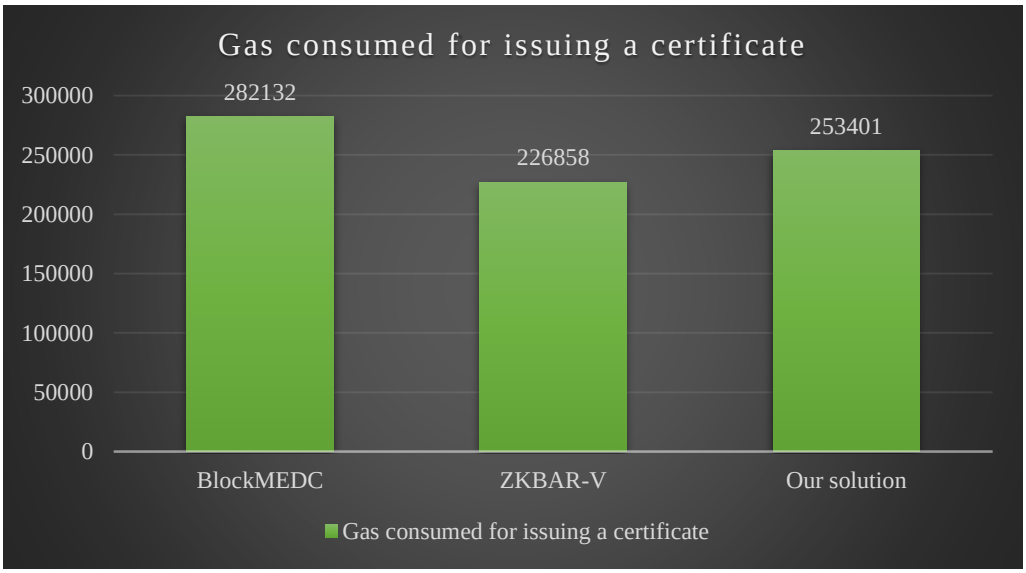


Figure 46:Comparative gas consumption for certificate issuance across BlockMEDC, ZKBAR-V, and the proposed solution

BUPT

5.1.2. Cost evaluation

To obtain a reliable estimation of transaction costs we conducted our experiments on the Sepolia testnet and on the Polygon zkEVM Cardona testnet. This approach enabled a systematic comparison of transaction costs between the Sepolia testnet and the Polygon zkEVM Cardona testnet. By monitoring fluctuations in gas fees, we were able to assess the extent to which network congestion impacts overall cost. Additionally, the analysis allowed us to evaluate whether the cost advantages of the Polygon testnet are sufficient to accommodate the substantial volume of credentials and documents that must be recorded for teacher evaluation, thereby providing practical insights into the scalability and economic feasibility of the proposed solution.

Ethereum's fee mechanism ensures that validators are compensated for the computational resources necessary to process and validate transactions, thereby upholding both security and resistance to abuse within the network [178]. In August 2021, a major upgrade was introduced through EIP-1559, which sought to address inefficiencies in fee predictability by replacing the fixed gas price model with a dual-structured approach [179]. Under this system, fees are composed of two elements: a base fee, dynamically adjusted according to network demand and permanently removed from circulation via burning, and a priority fee, or tip, which directly incentivizes validators for prompt transaction inclusion [179].

The resulting transaction cost is determined by the following formula [180]:

$$\text{Gas used} \times \min(\text{Base fee} + \text{Priority fee}, \text{Max fee per gas})$$

In this expression:

- gas used reflects the computational intensity of the transaction, with complex operations consuming more gas units.
- base fee ensures fee stability and mitigates volatility through an adaptive mechanism, while its burning process contributes to ether's deflationary pressure.
- priority fee is user-defined, enabling faster confirmations.
- max fee per gas represents the maximum amount a user is willing to pay per unit of gas, thereby constraining transaction costs even under volatile network conditions.

Gas prices are typically expressed in Gwei (1 ether = 10^9 Gwei), a smaller denomination of ether [181].

When compared to Bitcoin's transaction fee model, which operates on a simple first-price auction system where users compete by bidding higher fees, Ethereum's EIP-1559 introduces a more predictable and user-friendly mechanism. While Bitcoin relies solely on competitive bidding to prioritize transactions [182], leading to significant fee volatility during periods of high demand, Ethereum's dual-component model stabilizes fees through the dynamically adjusted base fee, while still preserving incentives for validators via the priority fee.

BUPT

The total transaction cost on Polygon zkEVM, which implements zk-Rollups, is determined by the execution of the transaction on Layer 2 and the on-chain submission of the zk-SNARK validity proof [143]:

Gas on Layer 2 × Gas Price on Layer 2 + Gas for zk-SNARK Proof × Gas Price on Layer 1

In this expression:

- Gas on Layer 2 reflects the computational effort of executing the transaction within the zkEVM environment, including contract logic and state updates.
- Gas Price on Layer 2 is the cost per unit of gas on Polygon zkEVM.
- Gas for zk-SNARK Proof represents the gas required to submit the validity proof and minimal calldata to Ethereum, ensuring the correctness of the batch.
- Gas Price on Layer 1 is the fee per unit of gas on Ethereum, which secures the zk-Rollup batch.

Polygon zkEVM transactions do not include a priority fee, as transaction ordering is handled by the sequencer, and this prevents large deviations from the average cost.

First, to quantify the resource utilization associated with our implementation, we recorded the gas consumption and estimated deployment costs for the two smart contracts on the Polygon zkEVM Cardona testnet, respectively on Sepolia testnet. The results are presented in Table 14.

Test network	***Smart contract***	***Gas used***	***Transaction fee in ETH***	***Transaction fee in USD***
Polygon zkEVM Cardona	Educational resource	1460353	0.00001460353	0.056
	Credential	1735019	0.00001735019	0.066
Sepolia	Educational resource	1460805	0.00219446055	8.40
	Credential	1735549	0.00263520363	10.09

Table 14: Transaction fee for deploying the smart contracts on Polygon zkEVM Cardona testnet.

Transaction fees on Polygon zkEVM Cardona are substantially lower than on Sepolia, with the Educational Resource contract costing approximately 0.056 USD and the Credential contract 0.066 USD, compared to 8.40 USD and 10.09 USD on Sepolia. This significant reduction in cost highlights the efficiency of Layer 2 solutions, making Polygon zkEVM particularly suitable for handling a high volume of transactions while maintaining affordability.

Moya et al. [143] conducted a similar comparison between deploying smart contracts on Ethereum Mainnet and Polygon zkEVM. Table 15 provides a comparative analysis of their results alongside those obtained in the present study, with emphasis on the cost reductions achieved when deploying smart contracts on a Layer 1 blockchain compared to a zk-Rollup Layer 2 solution.

Solution	***Layer 1 blockchain***	***Layer 2 blockchain***	***Cost reduction***
[143]	Ethereum Mainnet	Polygon zkEVM	94%
Our solution	Sepolia testnet	Polygon zkEVM Cardona testnet	99%

BUPT

Table 15: Smart contract deployment cost reduction.

The deviation of transaction costs is evident from the analysis of our transactions on the Sepolia test network, as shown in Figure 47. On the Ethereum mainnet, transaction costs can exhibit significant deviation due to network congestion. If the priority fee is not adequately adjusted, transactions may remain pending for prolonged periods, potentially increasing costs and amplifying the deviation from the average. This highlights the sensitivity of gas fees to network activity and the variability that can be expected under high-demand conditions. During testing on the Sepolia network, we implemented a mechanism to incrementally adjust the priority fee to avoid excessive transaction delays, while simultaneously imposing an upper limit of 1 USD per transaction. This approach accounts for the observed variation in transaction costs, with some transactions incurring slightly higher fees due to fluctuations in network conditions.

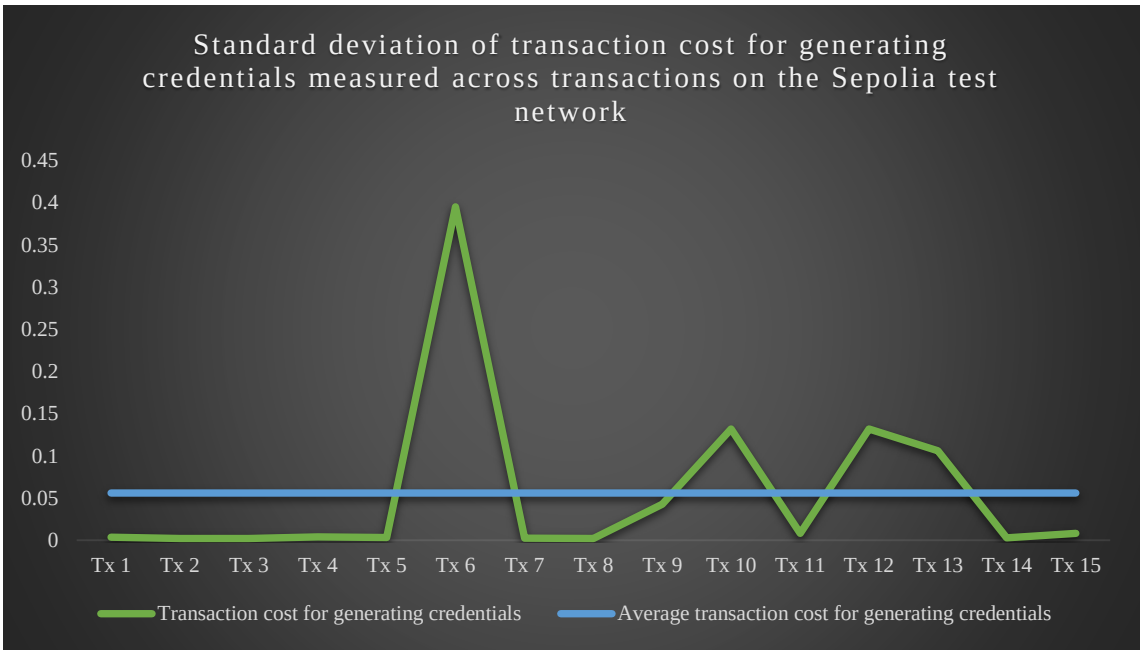


Figure 47:Cost deviation across transactions on the Sepolia test network compared to the average value.

Because Layer 2 transactions do not include tips, it is important that gas consumption remains consistent, as this directly affects the overall cost of executing transactions. On the Polygon zkEVM Cardona network, the on-chain registration of a credential required 219201 gas, reflecting the computational effort associated with verifying access permissions, ensuring the uniqueness of IPFS hashes, updating the relevant mappings, and emitting the corresponding events. Registering an educational resource consumed 171682 gas, a lower cost attributable to the comparatively simpler operations involved.

5.1.3. Latency analysis

To assess system latency, the smart contracts were first deployed using Remix, after which timing logic was incorporated into the application. This implementation facilitated the measurement of transaction duration for key operations by recording the elapsed time between the initiation of a transaction and its confirmation on the blockchain. By embedding this

BUPT

measurement mechanism within the application workflow, it was possible to systematically monitor and report latency across a range of blockchain interactions.

We present the measured latency of transactions executed on the Sepolia test network, illustrating the time required for confirmation across different operations. To provide a clearer view of how transaction latency fluctuates relative to the overall trend, the recorded data were plotted as a line chart in Figure 48. Each individual transaction latency is displayed in relation to the constant average value of 21.73 seconds, offering an immediate visual comparison that highlights both deviations and consistency across transactions.

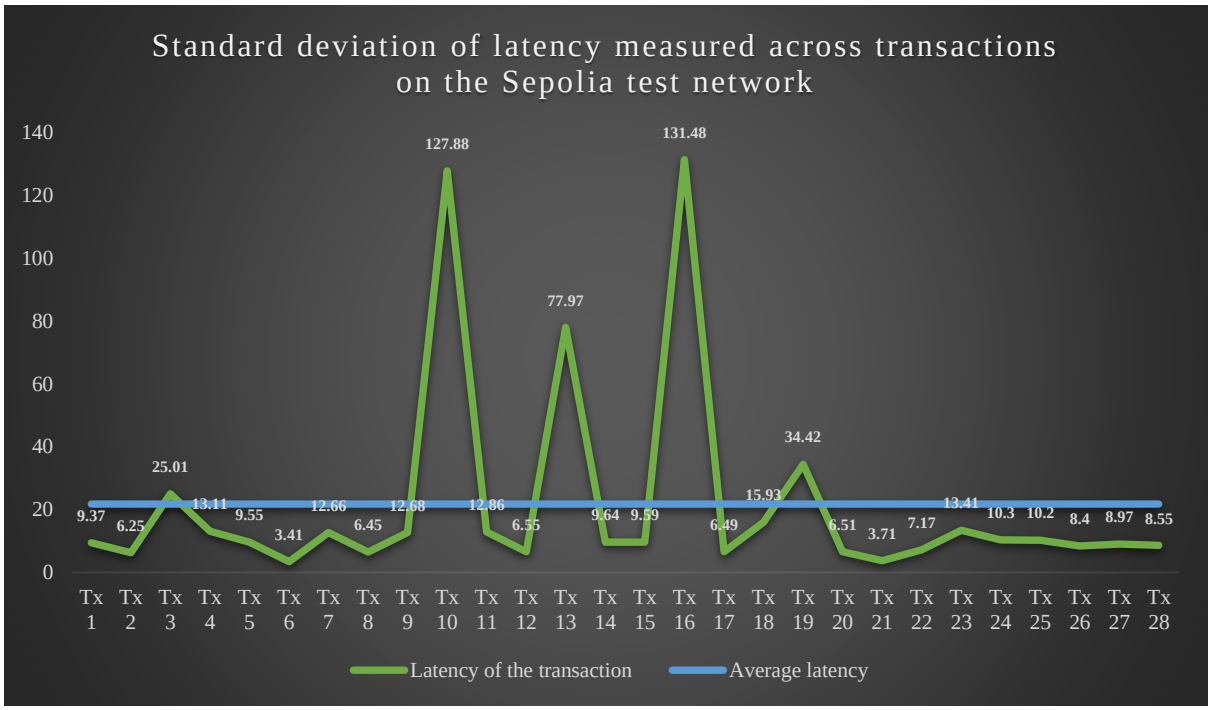


Figure 48:Latency deviation across transactions on the Sepolia test network compared to the average value.

The distribution of transaction latency is summarized across defined time intervals and illustrated in Figure 49. The results indicate that the majority of transactions are confirmed within relatively short periods, while a limited number experience substantial delays. Specifically, the largest share, 17 transactions (61%), were confirmed in the 0–10 second interval. An additional 6 transactions (21%) fell within the 10–20 second range. Beyond this, latency becomes less frequent: 2 transactions (7%) required 20–50 seconds, while only 1 transaction (4%) extended to 50–100 seconds. Finally, 2 transactions (7%) exhibited notably high latency in the 100–150 second range.

BUPT

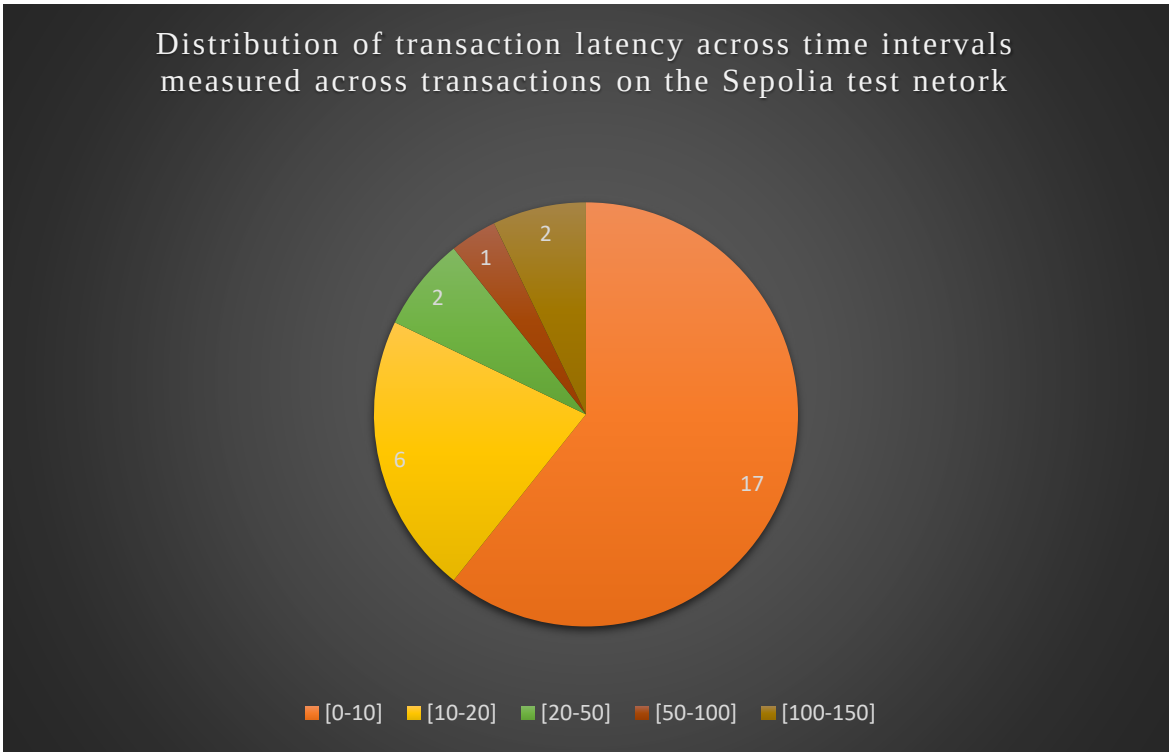


Figure 49:Distribution of transaction latency on Sepolia test netwok across time intervals expressed in seconds.

This distribution highlights that over three-quarters of all transactions complete within 20 seconds, demonstrating efficient confirmation under normal conditions, though sporadic cases of elevated latency suggest potential network congestion or variability in block production times.

We present in Figure 50 the measured latency of transactions executed on the Polygon zkEVM Cardona test network, illustrating the time required for confirmation across different operations.

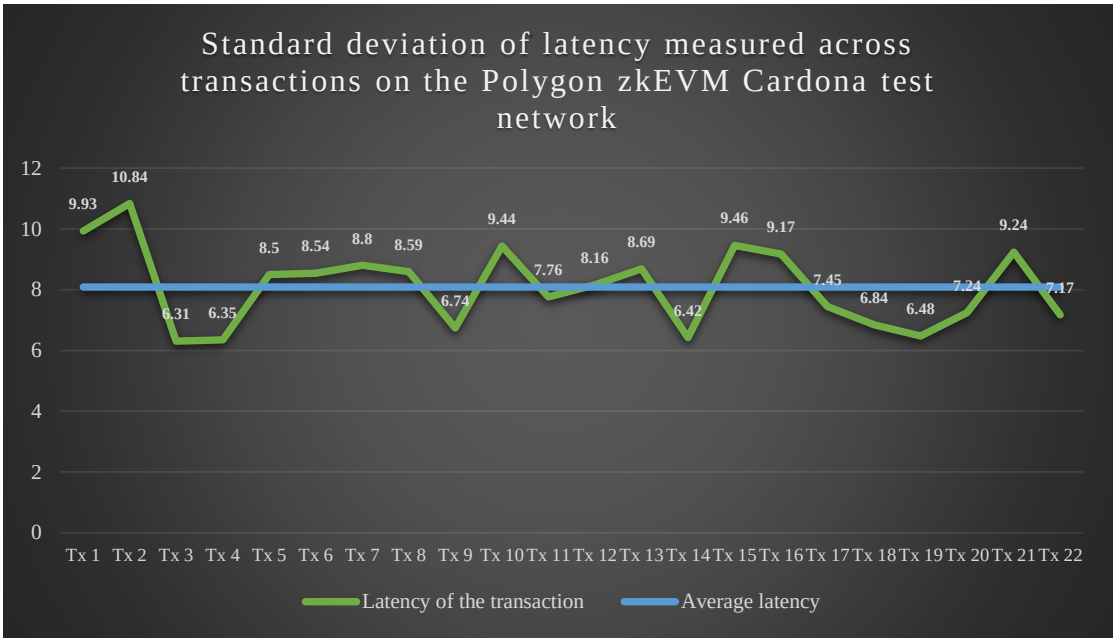


Figure 50: Transaction latency across transactions on the Polygon zkEVM Cardona test network compared to the average value.

BUPT

On the Polygon zkEVM Cardona test network, the observed transaction latency demonstrates a relatively stable performance, with most values clustering in the range of 6 to 10 seconds. Across the 22 recorded transactions, latency varied between 6.31 and 10.84 seconds, resulting in an average confirmation time of 8.09 seconds.

It is important to emphasize that latency on a Layer 1 blockchain can fluctuate far more dramatically than what is typically observed on Layer 2 networks. This variability stems from the intense competition among transactions to be included in a block, where limited block space and fluctuating network demand play a decisive role. As a result, confirmation times on Layer 1 may range from just a few seconds to several minutes, depending on factors such as network congestion, gas fee prioritization, and block production intervals. In contrast, Layer 2 solutions, designed specifically to alleviate these constraints, generally provide faster and more consistent transaction finalization, as they batch and process operations off-chain before anchoring them to the Layer 1 chain.

To analyze comparative results at the blockchain level, we considered several solutions developed in the domain of digital credentials, each relying on different infrastructures with varying trade-offs between cost, speed, and scalability. Our solution demonstrates excellent scalability for managing teacher attestations and credentials, making it suitable for implementation at a national level, as shown in Table 16. Compared to earlier blockchain-based approaches, which were limited either by high transaction costs or insufficient throughput, our design achieves a balance of affordability, efficiency, and performance. With a low average cost per transaction and transaction speeds that can scale into the hundreds per second on modern Layer 2 infrastructures, the system can reliably support the high volume of credential issuance required by educational and training institutions across an entire country.

Solution	***Stage of development***	***Blockchain***	***Average cost, in USD, per transaction over a one-month period (August 10, 2025 – September 10, 2025)***	***Average transaction speed in TPS (transactions per second)***	***Scalability***
Blockcerts	Commercial solutions	Bitcoin	0.957	7	Low
Block.co	Commercial solutions				
DIAR	Prototype	Ethereum	0.511	15-30	Moderate
CertEdu	Prototype				
BCSC-Dapp	Prototype				
Cerberus	Prototype				
Bcert	Prototype				
BlockMEDC	Prototype	Polygon zkEVM	0.01	>2000	High
ZKBAR-V	Prototype				
Our solution	Prototype				

Table 16: Comparative blockchain-level performance of selected credentialing solutions

BUPT

5.2. System efficiency in the teacher evaluation process

Digitalization within the Romanian education system is progressively advancing, with significant implications for evaluation processes. In particular, the scanning of candidates' work during examinations has become increasingly important to ensure accuracy in grading, minimize human errors, and reduce opportunities for fraud. Despite these advances in student assessment, the evaluation of teachers' professional activities largely remains a manual process, relying on the analysis of portfolios containing potentially hundreds of documents. This traditional approach is prone to inefficiencies, including the risk of document loss, misplacement, or fraudulent manipulation, which can undermine the integrity of the evaluation process.

Our first observation concerning the evaluation of teachers' professional activity revealed a notable lack of digitization in the existing processes. Traditionally, the issuance, collection, and verification of professional documents such as attestations, diplomas, and certificates have relied heavily on paper-based methods, which are both inefficient and prone to errors or loss. The system we propose addresses these challenges by providing comprehensive digitization solutions to all stakeholders involved. For issuing institutions, it eliminates the need to physically hand over documents to teachers, allowing credentials to be generated and stored digitally from the outset. For teachers, the system automatically consolidates all issued documents into a secure digital portfolio, enabling ready access and streamlined management of their professional records. Furthermore, evaluators, including members of the assessment committee, benefit from improved verification capabilities, as the authenticity of documents can be confirmed directly through the system, mitigating potential disputes and ensuring a more reliable and efficient evaluation process.

A central consideration in the design of our blockchain-based solution was the integration of timestamping as a fundamental feature. The evaluation of teachers' professional activity is carried out periodically, typically at intervals of one, two, or five years, which makes the precise date of credential issuance an important element in the assessment process. Blockchain technology, by its very nature, functions as an immutable ledger, ensuring that the timestamp associated with each transaction cannot be altered or tampered with. This characteristic provides strong guarantees of temporal authenticity, thereby strengthening the reliability of teacher evaluation records.

Beyond this inherent property of blockchain, our platform further extends functionality by incorporating a filtering option that enables teachers to display only the credentials obtained within a specific time interval. This same mechanism applies to educational resources created and uploaded to the platform, allowing teachers to present a curated view of their professional contributions. Such functionality not only facilitates compliance with evaluation requirements but also enhances usability, offering teachers a practical and transparent means to manage and showcase their achievements, as illustrated in Figure 51.

BUPT

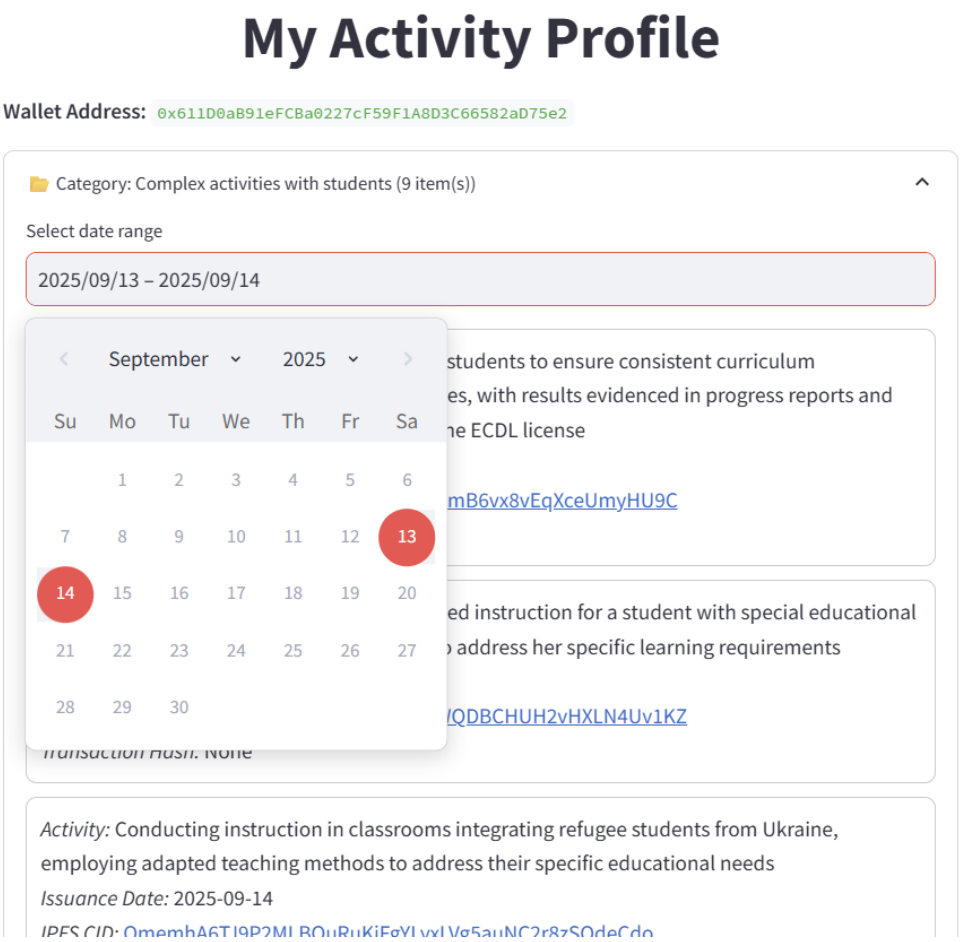


Figure 51: Timestamped display of credentials and educational resources within selected evaluation intervals. Source: screenshot from our application.

An essential dimension in which the proposed system brings significant improvements is the prevention and mitigation of fraud. In the current context, the authenticity of educational documents can often be questioned, particularly when they are transmitted or stored in physical form, where risks of forgery, alteration, or unauthorized duplication remain high. By contrast, our blockchain-based system ensures that documents are issued directly into the teacher's account, thereby eliminating intermediary steps that could expose them to tampering. Moreover, only institutions formally accredited and designated as administrators by a higher authority are authorized to issue such documents within the system. This hierarchical control mechanism, combined with the inherent immutability of blockchain records, establishes a robust safeguard against fraud, ensuring that every credential or resource is both verifiable and securely anchored to a trusted source.

5.3. Summary

The experimental results highlight two key dimensions of our proposed solution: system performance and system efficiency. In terms of performance, we evaluated the deployed smart contracts with respect to gas consumption, transaction costs, and latency across different blockchain environments. The comparative assessment between Sepolia and Polygon zkEVM Cardona illustrated the substantial cost reductions achievable through Layer 2

BUPT

solutions that use zk-Rollups, while also ensuring stable performance even under varying levels of network congestion. Latency measurements confirmed that transactions were processed within acceptable ranges, validating the system's technical feasibility for large-scale deployment.

System efficiency was examined with a focus on the practical benefits of integrating blockchain into the teacher evaluation process. Our system streamlines the traditionally paper-based and fragmented evaluation model by digitizing the issuance, storage, and verification of credentials and educational resources. Documents are securely issued by accredited institutions, automatically recorded in the teacher's digital portfolio, and easily accessible to evaluation committees, thereby reducing administrative overhead, eliminating the risk of document loss or fraud, and providing transparent and verifiable evidence of professional activity. In this way, the experimental results confirm that the proposed system not only performs effectively from a technical standpoint but also delivers meaningful improvements in efficiency and integrity for teacher activity evaluation.

BUPT

6. Security analysis over the smart contracts

6.1. Overview

This chapter introduces the systematic evaluation of the proposed smart contracts, with an emphasis on the key factors that determine their security and operational soundness. The analysis is structured around three central dimensions that are important for any blockchain-based application:

- *Access control:* represents the mechanisms by which roles and permissions are defined and enforced, ensuring that only authorized entities can perform sensitive operations. By ensuring that only privileged entities may issue or revoke records, the contract reduces the risk of denial-of-service or distributed denial-of-service attacks in which malicious actors could otherwise initiate mass revocations or uncontrolled insertions. This alignment with the principle of least privilege strengthens governance resilience and preserves system availability.
- *Constraint enforcement:* consists in the implementation of safeguards that guarantee the correctness of input data and the coherence of contract state. This involves requirements such as the use of non-zero addresses, the provision of substantive content identifiers, and explicit markers of resource existence. These safeguards suppress spam and storage bloat by rejecting malformed or valueless entries, while simultaneously preventing the integrity erosion that would otherwise manifest through inaccessible or phantom records. Robust constraint enforcement is therefore integral to both correctness and long-term maintainability.
- *Gas optimization:* refers to the efficiency with which contract functions execute, reflected in the cost of transaction processing. Efficiency at the computational layer not only reduces transaction costs but also serves as a security-relevant property. The contract must minimize redundant storage operations, applies carefully bounded numeric types where safety permits, and reduces unnecessary read operations. These measures lower the cost per transaction and thereby strengthen economic sustainability. Beyond mere efficiency, gas-aware design enhances scalability and resilience under high-volume or adversarial conditions, ensuring that the system remains operable without succumbing to cost-induced vulnerabilities.

Regarding access control, Cruz et al. [183] emphasize the need to use Ethereum smart contract that enforce secure, decentralized role-based access control (RBAC) across organizational boundaries. By recording user-role assignments immutably on-chain and incorporating a challenge–response protocol to verify user identities, the system ensures only authorized individuals can access protected resources. Within the blockchain context, three representative implementations have emerged [183], each with distinct characteristics:

BUPT

- *RBAC-SC*: a dedicated smart contract design that encodes role hierarchies and enforces access rules directly on-chain [184], offering fine-grained control but at the cost of increased complexity and gas consumption.
- *Smart policies*: policy-driven access systems where rules are expressed in a higher-level policy language and translated into executable smart contracts, providing flexibility but raising concerns regarding policy verification and enforcement consistency.
- *OpenZeppelin contracts*: a library in the Ethereum development ecosystem that provides a standardized and flexible way to implement role-based access control in smart contracts [185]. Introduced as part of the OpenZeppelin Contracts suite, AccessControl allows developers to define distinct roles with specific permissions and assign them to different addresses. Unlike simple ownership patterns that rely on a single owner address, AccessControl supports multiple roles with varying levels of authority, enabling more granular and decentralized permission management. This framework improves security and maintainability by making it easier to manage who can call certain functions within the contract, and it has become a standard best practice since its introduction in 2019 with the rise of complex dApps requiring rigorous access control.

Table 17 summarizes the main characteristics of the three RBAC implementations considered in this thesis, highlighting their defining features and corresponding limitations.

Implementation	*Core Mechanism*	*Advantages*	*Limitations*
RBAC-SC	Direct on-chain role logic	Fine-grained control. Transparent enforcement.	Higher complexity. Costly in gas consumption.
Smart Policies	Flexible policy definition	High flexibility. Adaptable to changing governance needs.	Risk of policy–contract mismatch; Verification overhead.
OpenZeppelin Contracts	Standardized RBAC modules	Widely audited. Secure and easy to integrate.	Limited customization. Relies on predefined patterns

Table 17: RBAC implementations in smart contracts

Regarding rigorous validation of input data, which serves as a guarantee in the design of smart contracts, ensuring that only well-formed and meaningful data is admitted into the system, our smart contracts comply with:

- *Enforcing strict validation of input data*: ensures that all required fields are properly completed and non-empty, thereby preventing incomplete, inconsistent, or malformed submissions from entering the system.
- *Validating the authenticity and format of provided addresses or identifiers*: guarantees that only legitimate and recognized participants can interact with the system, reducing the risk of malicious, fraudulent, or arbitrary entries.

BUPT

Regarding gas optimization, in [186, 187, 188] are presented techniques that reduce costs, which we have also integrated into our smart contracts. This techniques often involve trade-offs between efficiency, readability, and maintainability and include:

- *Storage load (SLOAD) versus memory load (MLOAD)*: Because storage loads are substantially more costly than memory loads, frequently accessed values should be cached in memory after the initial retrieval to minimize gas consumption.
- *Bitmaps*: Libraries such as OpenZeppelin's bitmap utility allow efficient mapping of uint256 to boolean, leveraging bit-level operations to reduce gas costs, particularly when writing non-zero values.
- *Variable packing*: Aligning smaller data types within the same 256-bit storage slot minimizes wasted space and improves efficiency, though improper packing can increase costs due to conversions or unnecessary slot operations.
- *Function visibility*: Declaring functions with the most restrictive visibility (private, internal, external, then public) saves gas by preventing unnecessary data handling.
- *Immutable variables*: Variables that are assigned only once during the constructor and never modified thereafter should be declared as immutable, thereby eliminating costly storage writes and reducing gas consumption.
- *Combine address mappings*: Consolidating multiple mappings into one mapping to a struct reduces storage use and lowers gas costs for reads and writes.
- *Payability design*: Declaring functions or constructors as payable can be more gas-efficient, since non-payable functions introduce additional value-checking operations.
- *Prefer private over public for constants*: Declaring constants as private reduces deployment costs, since the compiler does not generate redundant getter functions or expand the method ID table. The values remain accessible through the verified contract source code.
- *Use a recent Solidity version*: Adopting the latest stable release of Solidity is recommended, as newer versions typically include compiler-level optimizations that reduce gas consumption.
- *Syntactic preferences*: Using $<$ and $>$ instead of $\leq$ and $\geq$, can yield minor efficiency gains because Solidity compiles $\leq$ and $\geq$ by combining less than/greather than with an extra ISZERO check, which makes them less efficient than simple $<$ or $>$.
- *Minimal proxy contracts*: The ERC-1167 minimal proxy standard enables the deployment of lightweight delegate proxies, lowering gas consumption for multiple contracts that share similar functionality.

BUPT

Table 18 presents the gas optimization techniques applied in our contracts too, with estimation gas saved made by Valverde [187]:

Gas optimization technique	*Estimated gas savings*	*Illustrative examples*
Storage vs. Memory	High	Each SLOAD costs around 100 gas after the first load, compared to only 3 gas per MLOAD; caching avoids repeated storage reads.
Bitmaps	High	Writing non-zero→non-zero costs 5,000 gas, while zero→non-zero costs 22,100 gas. Using bitmaps helps avoid costly zero→non-zero writes.
Variable packing	Moderate	Combining two uint128 into one slot avoids an additional over 20,000 gas write.
Function visibility	Minor	external vs. public can save around 100–200 gas per call.
Immutable variables	Moderate	Avoids expensive storage writes of about 20,000 gas per SSTORE, since immutable variables are embedded directly in the bytecode and accessed cheaply.
Combine address mappings	Moderate	Reduces repeated lookups, saving around 200 gas per read and around 20,000 gas per write.
Payability design	Minor	payable avoids additional value-check opcodes, saving 20–40 gas.
Private constants	Minor	Prevents autogenerated getters, reducing runtime costs by 200–400 gas.
Recent Solidity version	Minor	Solidity 0.8.x optimizes arithmetic checks, saving hundreds of gas per operation compared to older versions.
Syntactic preferences	Minor	< and > avoid extra ISZERO test, saving around 3 gas per comparison.
Minimal proxy contracts	High	Deployment of a minimal proxy costs around 2,400 gas vs. over 30,000 gas for a full contract deployment.

Table 18: Gas saved using gas optimization techniques

6.2. Security analysis over of the smart contract for educational resources

6.2.1. Publishing and verifying the smart contract for educational resources

To ensure that our smart contract is both secure and efficient, we adopted an incremental development approach. Starting with a minimal and functional base version, we progressively integrated additional features while carefully evaluating their implications. At each stage, we refined the design, improved access control mechanisms, and reinforced data integrity checks. This step-by-step process allowed us to identify potential vulnerabilities early, apply security best practices consistently, and validate the contract's behavior under different scenarios. The final version thus represents a feature-complete implementation, achieving a high security score and demonstrating the advantages of gradual, methodical construction over a one-step deployment.

Publishing and verifying a smart contract on a blockchain explorer such as Etherscan for the Sepolia test network, commonly referred to as Sepolia Contract Verifier, serves critical

BUPT

roles in transparency, trust, and usability. By verifying the contract, the source code is matched with the deployed bytecode, allowing anyone to inspect and understand the logic behind the contract. This verification ensures that what was written by the developer is exactly what is running on the blockchain, preventing hidden logic or malicious behavior. Additionally, verification enables users, developers, and auditors to interact with the contract directly via the explorer's user interface without needing external tools.

Before proceeding with the publishing and verification of our smart contract, we prioritized key security considerations during development. In particular, we emphasized restricting access to the most important functions exclusively to the school entity, designated as the contract owner.

In the context of Ethereum smart contracts, access control plays a fundamental role in regulating who can execute certain functions, thereby safeguarding the contract's integrity and data. This distinction can be summarized as follows:

- *Without access control*: Any user on the Ethereum network, possessing a valid Ethereum address, can invoke any public or external function of the smart contract without restriction. This unrestricted access allows potentially malicious actors to add, modify, or revoke data, thereby exposing the contract to severe security risks and unauthorized manipulations.
- *With access control*: The smart contract enforces permissioned access by restricting function execution to specific authorized roles or addresses, such as the contract owner or designated teachers. Although any Ethereum user can send transaction requests to the contract, only those with verified permissions can successfully execute sensitive functions, significantly enhancing the security and integrity of the contract.

In a smart contract without access control, as shown in Figure 52, the absence of restrictions significantly weakens the contract's integrity, reliability, and trustworthiness, making it unsuitable for applications that require secure and controlled resource management.

BUPT

```
// SPDX-License-Identifier: MIT
pragma solidity ^0.8.13;

contract EducationalResourceContract {
    struct EducationalResource {
        string title;
        string subject;
        string teacherName;
        address teacherCreator;
        string ipfsHash;
        uint256 uploadDate;
        bool revoked;
    }

    mapping(uint256 => EducationalResource)
public educationalResources;
    mapping(address => uint256[]) private
teacherResources;
    uint256 public nextResourceId;

    event EducationalResourceAdded(
        uint256 indexed id,
        string title,
        string subject,
        string teacherName,
        address indexed teacherCreator,
        string ipfsHash,
        uint256 uploadDate
    );

    event EducationalResourceRevoked(
        uint256 indexed id,
        address revokedBy
    );

    function addEducationalResource(
        string calldata _title,
        string calldata _subject,
        string calldata _teacherName,
        string calldata _ipfsHash
    ) external {
        EducationalResource            memory
newResource = EducationalResource({
            title: _title,
            subject: _subject,
            teacherName: _teacherName,
            teacherCreator: msg.sender,
            ipfsHash: _ipfsHash,
            uploadDate: block.timestamp,
            revoked: false
        });

        educationalResources[nextResourceId]
= newResource;

teacherResources[msg.sender].push(nextResour
ceId);

        emit EducationalResourceAdded(
            nextResourceId,
            _title,
            _subject,
            _teacherName,
            msg.sender,
            _ipfsHash,
            block.timestamp
        );

        nextResourceId++;
    }

    function getEducationalResource(
        uint256 _id
    ) external view returns (
        string memory title,
        string memory subject,
        string memory teacherName,
        address teacherCreator,
        string memory ipfsHash,
        uint256 uploadDate,
        bool revoked
    ) {
        EducationalResource storage resource
= educationalResources[_id];

require(bytes(resource.ipfsHash).length > 0,
"Resource does not exist");
        return (
            resource.title,
            resource.subject,
            resource.teacherName,
            resource.teacherCreator,
            resource.ipfsHash,
            resource.uploadDate,
            resource.revoked
        );
    }

    function
revokeEducationalResource(uint256        _id)
external {
        EducationalResource storage resource
= educationalResources[_id];

require(bytes(resource.ipfsHash).length > 0,
"Resource does not exist");
        require(!resource.revoked, "Resource
already revoked");

        resource.revoked = true;

        emit EducationalResourceRevoked(_id,
msg.sender);
    }

    function
getTeacherEducationalResources(address
_teacher) external view returns (uint256[]
memory) {
        return teacherResources[_teacher];
    }
}
```

Figure 52: Smart contract without any access control

Recognizing the need for improved security, we implement several key security mechanisms in EducationalResourceContract, primarily focused on access control, input validation, and operational integrity, as follows:

BUPT

- *Immutable owner declaration*

```
address public immutable owner;
```

This declaration ensures that the contract owner is assigned only once during deployment and cannot be changed thereafter. The immutable keyword protects the ownership variable from unauthorized reassignment, thereby reducing the risk of privilege escalation.

- *Access control modifier*

```
modifier onlyOwner() {
    require(msg.sender == owner, "Caller is not the owner");
    _;
}
```

This modifier enforces that only the account which deployed the contract can invoke functions tagged with onlyOwner. This mechanism restricts unauthorized access to sensitive functions such as adding educational resources.

- *Constructor-based ownership initialization*

```
constructor() {
    owner = msg.sender;
}
```

During contract deployment, the msg.sender is assigned as the owner. This provides a trusted context for defining who holds administrative privileges from the outset, ensuring consistent enforcement of the onlyOwner modifier.

- *Controlled resource addition via onlyOwner*

```
function addEducationalResource(
    string calldata _title,
    string calldata _subject,
    string calldata _teacherName,
    string calldata _ipfsHash
) external onlyOwner {
    ...
}
```

This function is explicitly protected by the onlyOwner modifier. This restricts the ability to add new educational resources solely to the contract owner, preventing unauthorized or malicious actors from inserting unverified or harmful data.

- *Event emission for auditable actions*

```
emit EducationalResourceAdded(
    nextResourceId,
    _title,
    _subject,
    _teacherName,
    msg.sender,
    _ipfsHash,
    block.timestamp
);
```


BUPT

Emitting events during state-changing actions serves as a logging mechanism, facilitating off-chain auditing, forensic analysis, and transparent monitoring of resource submissions.

Once our smart contract for managing educational resources was fully developed and tested, it was compiled into bytecode and deployed to the blockchain using the Remix development environment. After deployment, the follow-up step involved making the contract's source code publicly accessible and verifiable. This was achieved by submitting the original code to a blockchain explorer such as Etherscan, linking it directly to the deployed contract address, as illustrated in Figure 53.

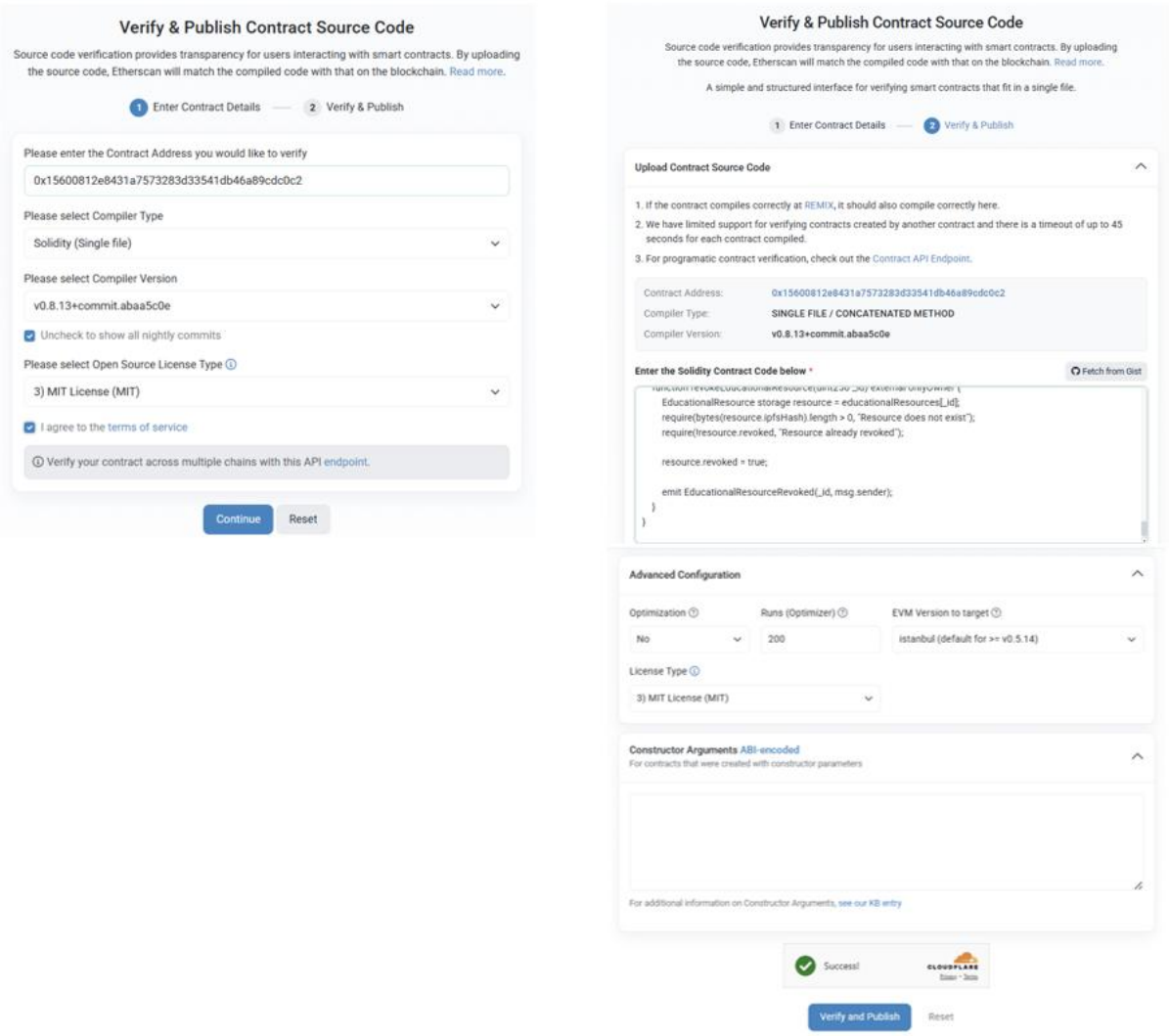


Figure 53: Publishing and verifying the smart contract for educational resources on a blockchain explorer.

After successfully publishing and verifying our smart contract on Etherscan, a list of potential security concerns and optimizations was generated, as illustrated in Figure 54.

BUPT

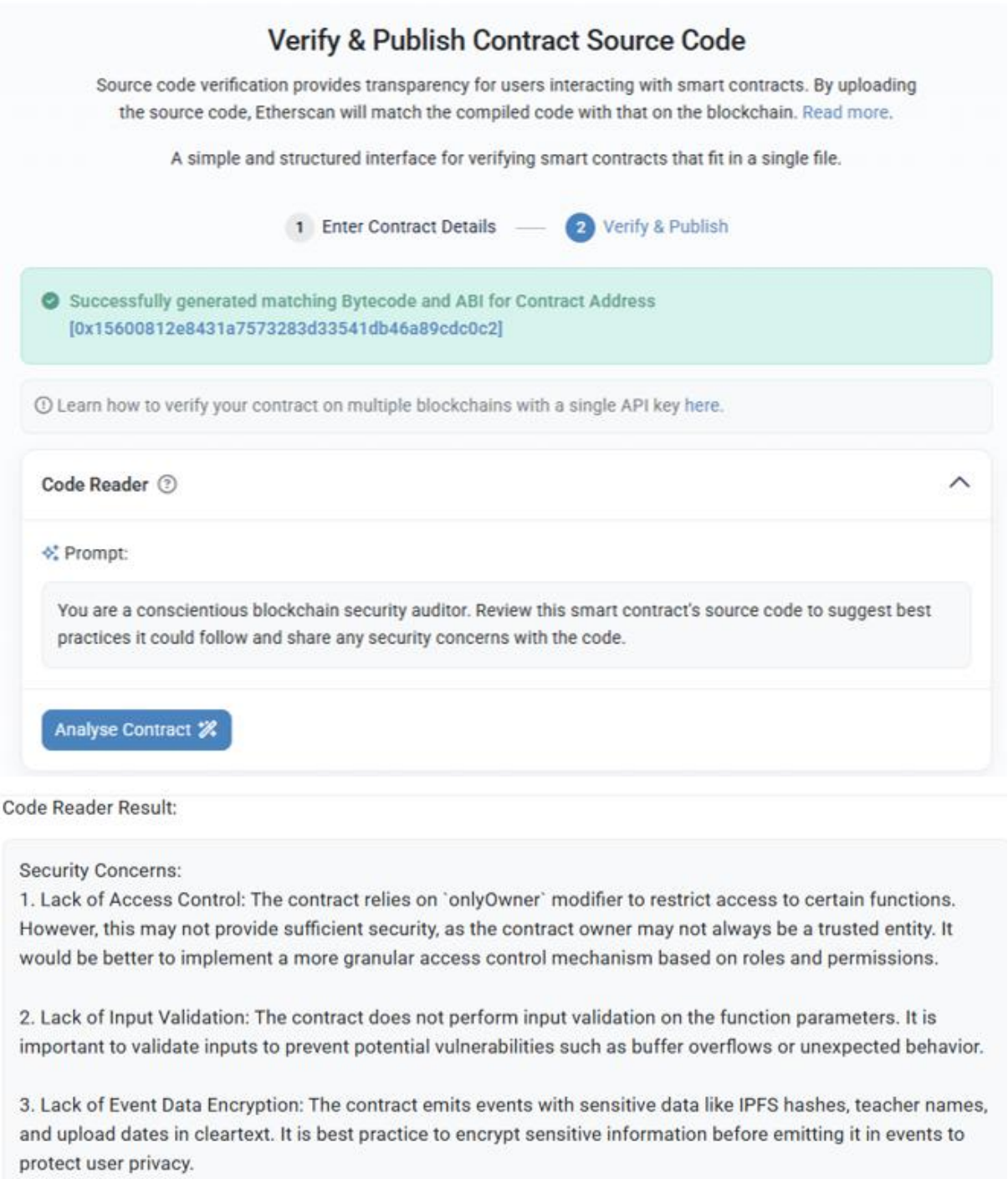


Figure 54: Educational resources smart contract scanning analysis.

These findings provided valuable insights into areas where the contract could be further strengthened. By analyzing these and addressing the recommendations, we were able to improve its security on the blockchain. We address this security concerns, as follows:

- *Granular access control with role-based permissions*

We implemented OpenZeppelin's AccessControl to introduce distinct roles:

  - DEFAULT_ADMIN_ROLE, which is assigned to the Owner who represents high-level authority institution. The owner, has exclusive

BUPT

rights to add and revoke teachers, as well as to add and revoke educational resources.

```
    constructor(address owner) {
        _grantRole(DEFAULT_ADMIN_ROLE, owner);
    }

    function addTeacher(address teacher) external
onlyRole(DEFAULT_ADMIN_ROLE) {
        require(teacher != address(0), "Invalid teacher address");
        grantRole(TEACHER_ROLE, teacher);
        emit TeacherAdded(teacher);
    }

    function addEducationalResource(
        string calldata _title,
        string calldata _subject,
        string calldata _teacherName,
        string calldata _ipfsHash,
        address _teacherCreator
    ) external onlyRole(DEFAULT_ADMIN_ROLE) {

        require(hasRole(TEACHER_ROLE, _teacherCreator), "Address is
not a registered teacher");

    }
```

- TEACHER_ROLE, which is assigned to teachers. Teachers can access educational resources. Teachers are restricted from adding educational resources themselves. This design choice ensures that every resource uploaded to the platform undergoes careful review and approval by the Owner. By centralizing the addition of resources to a trusted Owner, the system prevents a flood of unregulated or potentially low-quality content, maintaining a higher standard of educational material for all users.

- *Rigorous input validation to prevent invalid data*

We improved our smart contract to enforce strict validation of input data, requiring that all fields, including title, subject, teacher name, and IPFS hash, are non-empty. We have also ensured that the address provided for the educational resource creator is valid and belongs to a registered teacher. These safeguards help maintain the accuracy and reliability of educational resources and reduce the risk of unauthorized or incorrect submissions.

```
    require(bytes(_title).length > 0, "Title required");
    require(bytes(_subject).length > 0, "Subject required");
    require(bytes(_teacherName).length > 0, "Teacher name
required");
    require(bytes(_ipfsHash).length > 0, "IPFS hash required");
    require(_teacherCreator != address(0), "Invalid teacher
address");
    require(hasRole(TEACHER_ROLE, _teacherCreator), "Address is not
a registered teacher");
```


BUPT

- *Event emission transparency with privacy considerations*

Although the current implementation emits data clearly for traceability, implementations that have privacy in mind can be addressed off-chain.

```
event EducationalResourceAdded(
    uint256 indexed id,
    string title,
    string subject,
    string teacherName,
    address indexed teacherCreator,
    string ipfsHash,
    uint256 uploadDate
);

emit EducationalResourceAdded(
    nextResourceId,
    _title,
    _subject,
    _teacherName,
    _teacherCreator,
    _ipfsHash,
    block.timestamp
);
```

6.2.2. Security analysis of the smart contract using SolidityScan tool to identify and assess any potential vulnerabilities and threats

After successfully publishing and verifying our smart contract on Etherscan with the Contract Verifier and applying its suggested improvements, we carried out a thorough security analysis over this improved smart contract using SolidityScan, a more advanced tool for detecting and evaluating potential vulnerabilities and threats [189].

SolidityScan, launched in March 2022 by CredShields, is a comprehensive cloud-based vulnerability scanner tailored for Solidity smart contracts. It enables developers, auditors, and organizations to perform rapid and automated security audits across deployed contracts, GitHub repositories, or uploaded .sol files, generating detailed reports and security scores to help gauge and improve a contract's safety posture [190]. Additionally, SolidityScan has been integrated into blockchain explorers like Blockscout and Soneium Explorer, allowing users to instantly see security risk scores when viewing on-chain contracts [191].

After submitting our EducationalResource smart contract to SolidityScan, the platform performed an automated security analysis to assess its reliability. The tool systematically reviewed the contract's code for common vulnerabilities such as improper access control, reentrancy, unvalidated inputs, and potential logic flaws. The scanning process delivered a categorized report outlining identified issues by severity level, namely informational, low, medium, high, or critical, as illustrated in Figure 55, along with clear explanations and remediation recommendations. This streamlined workflow allowed us to quickly understand the contract's risk profile and take necessary actions to improve its overall security.

BUPT

Critical

The issue affects the contract in such a way that funds may be lost, allocated incorrectly, or otherwise result in a significant loss.

High

High-severity vulnerabilities pose a significant risk to both the Smart Contract and the organization. They can lead to user fund losses, may have conditional requirements, and are challenging to exploit.

Medium

The issue affects the ability of the contract to operate in a way that doesn't significantly hinder its behavior.

Low

The issue has minimal impact on the contract's ability to operate.

Informational

The issue does not affect the contract's operational capability but is considered good practice to address.

Gas

This category deals with optimizing code and refactoring to conserve gas.

Figure 55: Severity ranking of vulnerabilities in the smart contract scans conducted with SolidityScan

After downloading the SolidityScan security report, we analyzed the vulnerabilities shown in Figure 56 and initiated the process of systematically resolving each issue, following the recommendations made by the platform itself.

Severity	Bug Type	Instances
Low	USE OF FLOATING PRAGMA	1
Low	MISSING ZERO ADDRESS VALIDATION	2
Low	OUTDATED COMPILER VERSION	1
Informational	ADDING A RETURN STATEMENT WHEN THE FUNCTION DEFINES A NAMED RETURN VARIABLE IS REDUNDANT	1
Informational	BLOCK VALUES AS A PROXY FOR TIME	1
Informational	NAME MAPPING PARAMETERS	1
Gas	ASSIGNING TO STRUCTS CAN BE MORE EFFICIENT	1
Gas	AVOID RE-STORING VALUES	1
Gas	CHEAPER CONDITIONAL OPERATORS	4
Gas	DEFINE CONSTRUCTOR AS PAYABLE	1
Gas	FUNCTION SHOULD RETURN STRUCT	1
Gas	LONG REQUIRE/REVERT STRINGS	2
Gas	PUBLIC CONSTANTS CAN BE PRIVATE	1
Gas	STORAGE VARIABLE CACHING IN MEMORY	1
Gas	SUPERFLUOUS EVENT FIELDS	1

Figure 56: Vulnerabilities identified by SolidityScan for educational resources smart contract

BUPT

We address security concerns of „Low” level, as follows:

- *Use of floating pragma*

We addressed the use of a floating pragma by replacing „pragma solidity ^0.8.20”; with the fixed version directive „pragma solidity 0.8.20;”, thereby ensuring consistent compiler behavior and enhancing contract reliability by preventing unintended compatibility issues associated with version fluctuations.

```
pragma solidity 0.8.20;
```

- *Missing zero address validation*

We address this vulnerability by introducing explicit zero address validation throughout the smart contract to prevent the assignment of critical roles to invalid addresses. By ensuring that the contract owner, teachers, and teacher creators cannot be set as the zero address, we safeguard against unintended privilege delegation and logical inconsistencies. These checks, added in the constructor and functions like addTeacher, revokeTeacher, and addEducationalResource, reinforce secure role management and align the contract with best practices in Solidity for robust and secure deployment.

```
require(owner != address(0), "Owner address cannot be zero");

require(teacher != address(0), "Invalid address");
require(_teacherCreator != address(0), "Invalid teacher address");
```

- *Outdated compiler version*

We have upgraded the smart contract to the latest version, Solidity 0.8.30 [168], to use recent compiler improvements, optimizations, and security enhancements.

```
pragma solidity 0.8.30;
```

We address security concerns of „Informational” type, as follows:

- *Adding a return statement when the function defines a named return variable is redundant*

We enhance code clarity by removing the redundant explicit return statement in the getEducationalResource function, which defines named return variables. By directly assigning values to these named variables and relying on Solidity’s implicit return behavior, the contract improves readability and maintainability without altering functionality.

```
function getEducationalResource(
    uint256 _id
) external view returns (
    string memory title,
    string memory subject,
    string memory teacherName,
    address teacherCreator,
```


BUPT

```
            string memory ipfsHash,
            uint256 uploadDate,
            bool revoked
        ) {
            require(
                hasRole(DEFAULT_ADMIN_ROLE,        msg.sender)        ||
hasRole(TEACHER_ROLE, msg.sender),
                "Access denied: Only teacher or owner can view"
            );

            EducationalResource        storage        resource        =
educationalResources[_id];
            require(resource.exists, "Resource does not exist");

            title = resource.title;
            subject = resource.subject;
            teacherName = resource.teacherName;
            teacherCreator = resource.teacherCreator;
            ipfsHash = resource.ipfsHash;
            uploadDate = resource.uploadDate;
            revoked = resource.revoked;
        }
```

- *Block values as a proxy for time*

In this contract, the management of time is handled using the block.timestamp function. Specifically, when a new educational resource is added using the addEducationalResource function, the uploadDate is set to block.timestamp. This is the timestamp of when the transaction is mined and helps to track the exact moment the resource was uploaded.

By using block.timestamp, the contract ensures that the time at which an educational resource was created is accurately recorded and immutable. However, the contract does not use any proxies for time management, as block.timestamp is directly integrated into the contract to retrieve the current block's timestamp.

- *Name mapping parameters*

We improve code clarity and maintainability by renaming ambiguous names like teacherCreator to resourceCreator.

We address gas optimization concerns through the following measures:

- *Cheaper conditional operators*

To improve gas efficiency in Solidity smart contracts, conditional checks involving unsigned integers were refactored by replacing comparisons of the form x > 0 with the more cost-effective x != 0 construct. This substitution use the lower computational overhead of

BUPT

inequality checks over relational comparisons, thereby reducing the overall gas consumption during contract execution without compromising logical correctness.

```
require(bytes(_title).length != 0, "Title required");
require(bytes(_subject).length != 0, "Subject required");
require(bytes(_teacherName).length != 0, "Teacher name required");
require(bytes(_ipfsHash).length != 0, "IPFS hash required");
```

- *Define constructor as payable*

Given that the current smart contract does not include any Ether management or payable functions, we maintain the constructor as non-payable. This approach minimizes unnecessary complexity and potential security risks by preventing unintended Ether transfers during contract deployment, thereby promoting a simpler and more secure contract design.

- *Function should return struct*

Given the current limitations in Solidity's ABI encoding, our contract avoids returning structs directly from external functions. Instead, it employs external getter functions that return individual struct fields to ensure compatibility and straightforward interaction.

- *Long require/revert strings*

To optimize gas consumption in Solidity smart contracts, it is advisable to minimize the length of error messages used in require and revert statements. Shorter revert strings reduce the computational overhead associated with memory operations during contract execution, thereby enhancing efficiency without compromising the clarity of failure diagnostics. This practice is particularly relevant given the high cost of on-chain storage and execution, and contributes to more cost-effective and scalable decentralized applications.

We shortened the following error messages:

"Access denied: Only teacher or owner can view" → "Access denied".

"Address is not a registered teacher" → "Not teacher".

- Superfluous event fields

To reduce unnecessary gas consumption, the contract omits explicit inclusion of block.timestamp in emitted events. Since block.timestamp is inherently part of the transaction receipt metadata, manually including them duplicates information and increases storage costs without adding informational value.

```
emit EducationalResourceAdded(
    nextResourceId,
    _title,
    _subject,
    _teacherName,
    _resourceCreator,
    _ipfsHash
```


BUPT

```
);
```

To minimize gas consumption, while acknowledging that transaction costs remain relatively high, we chose to store only the most essential elements within the smart contract:

- resourceCreator, to provide verifiable evidence, through the teacher's public key, of who created each educational resource.
- ipfsHash, to maintain the reference link to the resource stored on IPFS.
- revoked, to indicate whether a resource has been invalidated or remains active.

Following a stepwise refinement methodology, the smart contract for managing educational resources was incrementally developed, with each iteration integrating additional security mechanisms and optimizations. Particular attention was given to gas efficiency, ensuring that only the most essential data elements were maintained on-chain to balance functionality with cost-effectiveness. The resulting final contract, which is deployed and utilized by our platform, is provided in Appendix A.

6.2.3. Comparative analysis of security results

We examine three successive versions of the smart contract through SolidityScan in order to illustrate how varying design choices impact the security posture of dApps and to underscore the importance of robust security factors in smart contract development. The security results score is represented in Figure 57.



Figure 57: Security score of the smart contract for educational resources

The first iteration received a security score of 61.71 after the scan made with SolidityScan, reflecting its fundamental lack of access control and incomplete validation. This absence of access control generates multiple systemic risks:

- *Denial-of-service attacks*: Because both addition and revocation of resources are unconstrained by role or provenance, any participant can indiscriminately

BUPT

inject arbitrary resources or disable legitimate ones. From a security-governance perspective, this contravenes the principle of least privilege and exposes the system to denial-of-service attacks, in which malicious actors overwhelm the contract with spurious entries while simultaneously revoking genuine content, thereby rendering the platform unreliable and effectively unavailable.

- *Spam and storage bloat:* The absence of constraints on resource submission allows adversaries to add arbitrary data at scale, inflating on-chain storage without limit. Such behavior introduces economic inefficiencies, degrades performance, and reduces the overall reliability of the application.
- *Integrity erosion and phantom entries:* addEducationalResource function does not enforce a non-empty IPFS hash. This permits the creation of "phantom" entries that consume identifiers but cannot be retrieved or revoked. The result is an erosion of dataset coherence, as auditability and provenance become compromised.

The second version of our smart contract attains a security score of 95.70. It eliminates the critical vulnerabilities of the first iteration by introducing owner-based authority, whereby only the contract owner is empowered to add or revoke resources. This improves security by reducing the attack surface, but it introduces new risks of a structural and governance nature:

- *Centralization of control:* By concentrating all decision-making authority in a single owner account, the contract establishes a single point of failure. If the owner is compromised, absent, or malicious, the entire system becomes vulnerable to unilateral manipulation or paralysis.
- *Governance rigidity:* The reliance on a singular authority figure reflects a rigid governance model that resists adaptation. While the security score is high in formal analysis, the model is brittle: system continuity is jeopardized if the owner loses keys or disengages, and no fallback or distributed mechanism is available.

The third version of our smart contract achieves a score of 98.19. It adopts role-based access control through OpenZeppelin's AccessControl, thereby providing a more nuanced and flexible governance framework. It assigns two distinct roles, DEFAULT_ADMIN_ROLE, which governs the addition or revocation of teachers and resources, and TEACHER_ROLE, which allows teachers to access and interact with educational material. This structure addresses the most severe vulnerabilities of version 1 and the centralization concerns of version 2, yet this risk remains:

- *Role mismanagement*: If administrative privileges are inadvertently revoked from all addresses or assigned improperly, the contract can enter a locked state where no further governance actions are possible. This risk, though less likely, reflects the importance of procedural discipline in managing roles.

BUPT

6.3. Security analysis over of the smart contract for credentials management

6.3.1. Publishing and verifying the smart contract for credentials management

Initially, as highlighted in our article, „Optimizing Teacher Portfolio Integrity with a Cost-Effective Smart Contract for School-Issued Teacher Documents", the smart contract was designed solely for issuing school attestations. However, since attestations represent only one component of a teacher's professional portfolio, we extended its functionality to also cover certificates, diplomas, and open badges. This expansion naturally required the introduction of role-based permissions, which we implemented using Access Control.

In initial smart contract, it allows the issuance and revocation of attestation by the director of an educational institution.

Recognizing the importance of security and integrity, the Attestations contract incorporates several mechanisms centered on access control, data validation, and auditability:

- *Owner declaration*

```
address public owner;
```

A single administrative authority is defined for the contract. The owner variable establishes who holds exclusive rights to manage attestations.

- *Constructor-based ownership initialization*

```
constructor() {
    owner = msg.sender;
}
```

The deploying account is assigned as the contract owner at deployment. This ensures a trusted origin of administrative control.

- *Access control modifier*

```
modifier onlyOwner() {
    require(msg.sender == owner, "Caller is not the owner");
    _;
}
```

Functions marked with onlyOwner can only be executed by the contract owner, preventing unauthorized parties from issuing or revoking attestations.

- *Controlled attestation generation*

```
function generateAttestation(...) external onlyOwner { ... }
```

Only the Owner can issue new attestations. This restriction ensures that all attestations are created by a verified authority, preventing the inclusion of false or malicious records.

- *Event emission for transparency*

```
emit AttestationGenerated(...);
emit AttestationRevoked(...);
```

Every issuance and revocation is logged via events, creating an immutable audit trail. This enables transparent off-chain monitoring and supports external auditing processes.

BUPT

We compiled the attestation smart contract with the Solidity v0.8.13 compiler, setting the compiler's version to "istanbul", and later deployed it on the Sepolia testnet for validation and experimentation. Once deployed, the contract was made publicly verifiable by uploading its source code to Etherscan blockchain explorer. This process ensured transparency by allowing anyone to inspect the original implementation directly at the contract's deployment address, as shown in Figure 58.

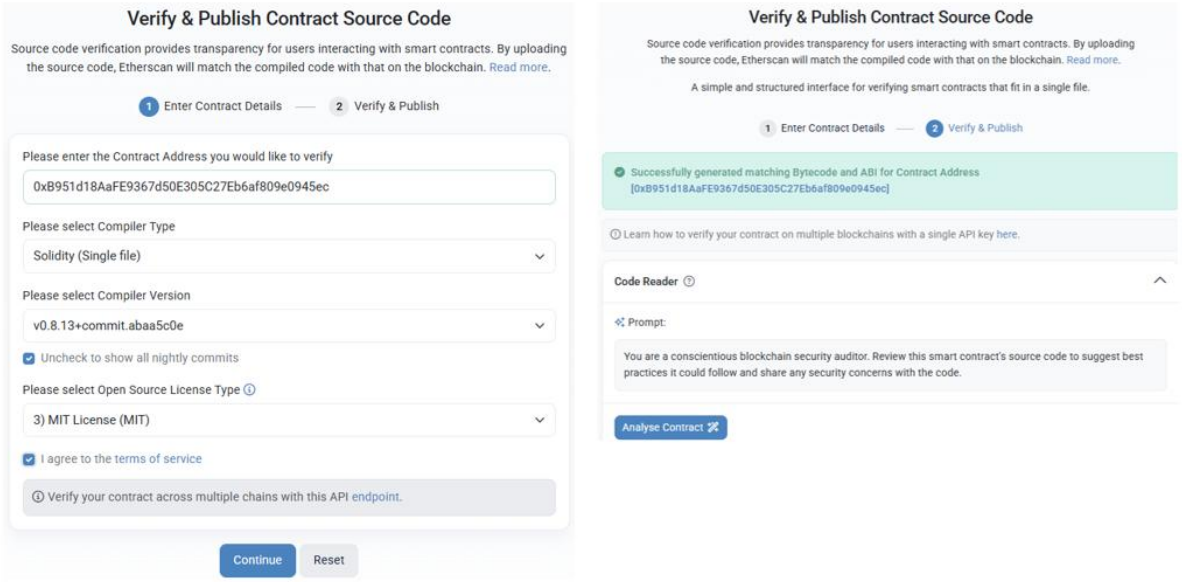


Figure 58: Publishing and verifying the smart contract for attestations on a blockchain explorer

Once the smart contract was published and verified on Etherscan, the platform automatically generated a report highlighting potential security issues and optimization opportunities, as shown in Figure 59.

BUPT

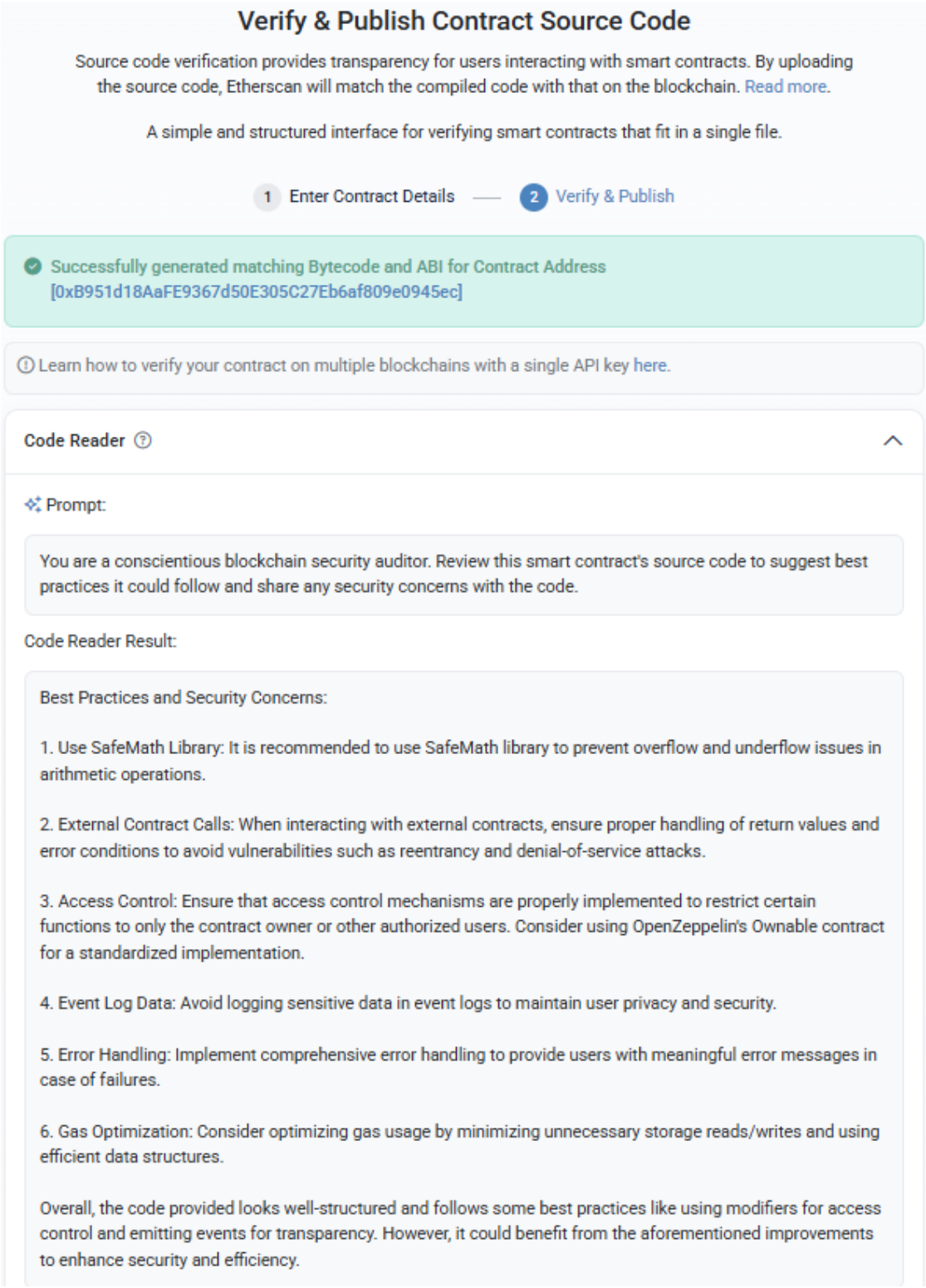


Figure 59: Credential smart contract scanning analysis.

The assessment highlighted key areas where the contract's security could be improved. Incorporating these recommendations enabled us to strengthen its security guarantees on the blockchain. The addressed concerns are detailed below:

- *Use SafeMath library*

BUPT

The recommendation to use the SafeMath library is primarily aimed at preventing issues related to integer overflow and underflow in arithmetic operations. However, starting from Solidity version 0.8.0 and above, built-in overflow and underflow checking has been introduced, making the explicit use of SafeMath unnecessary in most cases [192]. The Solidity compiler will automatically revert transactions if an overflow or underflow occurs. Following this observation, we decided to update the smart contract to use the latest version of the Solidity compiler, 0.8.30 [168].

- *External contract calls*

In our smart contract, external contract calls are not directly used.

- *Access control*

We evolved the initial contract, originally designed for a single school where the director issued attestations to teachers, into a scalable model in which the Owner role, where high-level authority institution can delegate administrative rights to educational and training institution, enabling them to issue attestations and academic credentials to teachers. We use OpenZeppelin AccessControl to define explicit roles. The roles can be described as follows:

  - *Owner*: Holds the DEFAULT_ADMIN_ROLE, which confers the highest level of authority within the contract. The owner has the ability to grant and revoke roles, thereby configuring the set of entities authorized to perform sensitive operations.
  - *Admin*: Granted the ADMIN_ROLE by the owner, these accounts are permitted to execute privileged functions such as generating and revoking attestations and academic credentials. This delegation enables a controlled distribution of operational responsibilities while maintaining owner-level oversight.
  - *Other users*: Accounts without an assigned role are restricted to non-privileged interactions, such as querying attestations and academic credentials. They retain the ability to view information but cannot alter the state of the contract.

- *Event Log Data*

In our smart contract, we need to implement event logs for actions like credential generation, revocation, and admin role changes. Additionally, we ensure that sensitive information, such as private details of teachers, is not logged in event logs to protect user privacy and security.

- *Error handling*

In our smart contract, we prioritize error handling by implementing clear and meaningful error messages for users, ensuring that failures are properly communicated.

- *Gas optimization*

To optimize gas consumption and maintain essential functionality, the smart contract was changed to store only the most important information on-chain. The key elements included are:

BUPT

- issuer_institution: the address of the issuing academic or training institution, enabling verification of the entity responsible for the document.
- receiver: the address of the individual receiving the document, linking the credential to the recipient's blockchain identity.
- ipfsHash: the IPFS link pointing to the off-chain stored document.
- credentialDate: the registration date of the document, providing a timestamp for record-keeping.
- revoked: a flag indicating whether the document has been revoked.

6.3.2. Security analysis of the attestation smart contract using SolidityScan tool to identify and assess any potential vulnerabilities and threats

After applying the recommendations received from Etherscan Contract Verifier, we moved on to the next stage of improving the security of our contract and optimizing the gas consumed by each action in the contract, performing a security analysis over the improved smart contract with SolidityScan and applying the in-depth recommendations received from the contract scan, presented in Figure 60.

Severity	Bug Type	Instances
Informational	BLOCK VALUES AS A PROXY FOR TIME	1
Informational	MISSING UNDERSCORE IN NAMING VARIABLES	1
Informational	NAME MAPPING PARAMETERS	2
Gas	AVOID ZERO-TO-ONE STORAGE WRITES	1
Gas	DEFINE CONSTRUCTOR AS PAYABLE	1
Gas	NAMED RETURN OF LOCAL VARIABLE SAVES GAS AS COMPARED TO RETURN STATEMENT	1
Gas	SPLITTING REQUIRE STATEMENTS	1
Gas	STORAGE VARIABLE CACHING IN MEMORY	3

Figure 60: Vulnerabilities identified by SolidityScan for attestation smart contract

We address security concerns of „Informational" type, as follows:

- *Missing underscore in naming variables*

BUPT

In our smart contract, we follow consistent naming conventions, ensuring that private variables are prefixed with an underscore to distinguish them from public variables.

- *Name Mapping Parameters*

We consider that the parameter names used in functions such as addAdmin(address school) and revokeAdmin(address school) did not align directly with the assigned role ADMIN_ROLE. To address this, we refined the naming of mapping parameters and function arguments to ensure they clearly reflect their underlying roles and purposes.

```
            function addAdmin(address admin) external
onlyRole(DEFAULT_ADMIN_ROLE);
            function revokeAdmin(address admin) external
onlyRole(DEFAULT_ADMIN_ROLE);
```

We address gas optimization concerns through the following measures:

- *Avoid zero-to-one storage writes*

We set nextId variable to start from 1, preventing the contract from writing a 0 ID into the storage when a new attestation is created. This prevents a potential issue where the contract could accidentally write a zero value into storage during the first attestation creation, which might be undesirable depending on the logic of the system.

```
            uint256 public nextId = 1;  // Initialize nextId to 1 to avoid
zero-to-one write
```

Additionally, the code uses:

```
            unchecked { nextId = id + 1; }
```

This increment mechanism ensures that the nextId is incremented by 1 after each attestation creation, which reduces the possibility of unnecessary writes to the blockchain, optimizing gas costs by only incrementing and writing to storage once.

- *Define constructor as payable*

Given that the current smart contract does not include any Ether management or payable functions, we maintain the constructor as non-payable to reduce unnecessary complexity.

- *Named return of local variable saves gas as compared to return statement*

We set the getCredential function to return an Credential struct and ensures that the credential exists.

```
            function getCredential(uint256 id) external view returns
(Attestation memory c) {
                c = credential[id];
                require(c.exists, "Credential does not exist");
            }
```

- *Storage variable caching in memory*

BUPT

In the initial version of the contract, several storage variables were accessed multiple times within the same function. Each SLOAD incurs a high gas cost of about 100 gas after the first read, whereas MLOAD and MSTORE are significantly cheaper with around 3 gas.

To optimize gas consumption and improve efficiency, we applied storage caching by introducing local variables for frequently reused state variables. This ensures that values are read once from storage and then reused from memory, minimizing redundant SLOAD operations. Optimization applied is over credentials[id] being cached as a storage reference in getCredential and isVerified.

The credential smart contract was designed through an incremental development process, with each stage introducing enhanced security features and structural optimizations. Emphasis was placed on ensuring gas efficiency by retaining only the most essential elements on-chain. The final version of this contract is provided in Appendix B.

6.3.3. Comparative analysis of security results

We examined two versions of the smart contract, one only for issuing attestations, respectively the other for issuing attestations and credentials, and found that while the first version works well for a single institution, it is not designed for scalability. The second version, however, introduces a role-based access control mechanism, making it far more suitable for broader implementation. This version allows the high-level authority institution to assign administrator roles to educational and training institution, empowering each one to securely issue attestations and credentials independently. This flexibility makes the second version not only scalable within a single county but also adaptable for national-level deployment.

We analyzed both versions of the smart contract using SolidityScan, and the resulting security scores are shown in Figure 61.



Figure 61: Security score of the smart contract for attestation

BUPT

The final version of the smart contract has achieved an impressive 98.29/100 security score, demonstrating a high level of protection against potential vulnerabilities. It utilizes a dual-level access control system, with the Owner role having full control over contract functions, such as adding or revoking admins, while the Admin role is restricted to managing specific aspects like generating and revoking attestations and credentials. This clear division of control enhances security by minimizing the risk of unauthorized actions. Additionally, the contract is designed with scalability in mind, ensuring that it can handle a growing number of documents and admins without compromising performance.

6.4. Summary

The security of the proposed system is ensured primarily through the design of the two smart contracts: the Educational Resource Contract and the Credential Contract. Both were developed incrementally, with each iteration strengthening their robustness against unauthorized access and manipulation. Role-based access control, implemented using the OpenZeppelin AccessControl library, was introduced to clearly separate permissions between administrators, institutions, and teachers, thereby preventing privilege escalation. Revocation mechanisms were also integrated into both contracts, allowing for the invalidation of resources or credentials in cases of error or misconduct, while preserving historical records for transparency.

In addition to access control and revocation mechanisms, special emphasis was placed on gas optimization to ensure the long-term sustainability of the contracts. Both the Educational Resource and Credential smart contracts were carefully structured to retain only the most essential data on-chain, thereby minimizing storage costs, which represent the primary driver of gas consumption. This design significantly reduces transaction costs while preserving verifiability and auditability.

BUPT

7. Conclusion and future work

7.1 Obtained results

The management and verification of teachers' professional documents in pre-university education remain fragmented and vulnerable to inefficiencies, loss, and fraud. Existing digital initiatives address aspects related to evaluation, but that related to the professional trajectories of teachers is not included. This gap motivated the development of a blockchain-based solution designed to connect institutions, teachers, and evaluation committees, while optimizing locally emitted attestations and credential issuance and verification.

This doctoral thesis established a set of specific objectives that guided the design, implementation, and evaluation of a blockchain-based framework for teacher evaluation and document management.

The findings of this research demonstrate the feasibility, efficiency, and scalability of a blockchain-based system for managing teacher attestations, credentials, and educational resources. By employing Polygon zkEVM as a Layer 2 zk-Rollup solution, the system effectively addresses one of the key challenges associated with previous blockchain implementations in education namely, the high transaction costs and limited throughput of Ethereum and comparable EVM networks. Through the use of zk-Rollups, multiple transactions are cryptographically aggregated into single proofs, significantly reducing the operational cost of recording and verifying credentials while maintaining security guarantees anchored to Ethereum. This approach ensures that credentialing processes can be executed at a national scale without imposing prohibitive costs on educational authorities or institutions.

In addition to cost reduction, the system's architecture incorporates decentralized storage through IPFS, which allows large educational resources to be managed off-chain while storing only essential metadata and content identifiers on the blockchain. This design choice balances the need for data immutability and transparency with the necessity of minimizing on-chain resource consumption. The combination of off-chain storage and on-chain verification ensures that every credential remains securely linked to the teacher's unique blockchain identity, preserving the authenticity and traceability of professional achievements.

Furthermore, the interactions between teachers, educational institutions, and verifiers are orchestrated through smart contracts, which define clear responsibilities for each stakeholder. This mechanism guarantees accountability, reduces opportunities for unauthorized manipulation, and ensures a transparent and auditable record of all issued credentials and attestations.

The experimental evaluation of the system highlights its capacity to handle high transaction volumes efficiently. By aggregating multiple credentialing operations into zk-Rollup batches, the system achieves transaction throughput in the thousands per second, a performance level far exceeding that of conventional blockchain-based credentialing approaches. This substantial increase in throughput demonstrates that the solution is technically capable of supporting the needs of a nationwide teacher evaluation framework, providing timely recording, verification, and retrieval of credentials even under heavy workloads.

BUPT

Simultaneously, the cost per transaction remains minimal, validating the solution's potential for widespread adoption without introducing financial barriers.

Another outcome of the research concerns the creation of a unified, portable portfolio for each teacher. By linking all attestations, diplomas, certificates, and contributions to educational resources to the teacher's blockchain identity, the system provides a coherent and tamper-resistant record of professional development. Teachers can readily access and present their achievements, while institutions maintain accountability through clearly defined issuance and revocation rights. Verifiers, including members of evaluation committees, benefit from streamlined access to accurate and trustworthy records, facilitating transparent and efficient evaluation processes.

The system also ensures temporal accuracy through the integration of blockchain-based timestamping. Each credential and attestation is recorded with an immutable timestamp, which allows precise verification of the issuance date and ensures that professional evaluations are based on accurate, verifiable data. This feature is particularly valuable in education systems where teacher assessments occur periodically and the timing of credential issuance directly impacts evaluation outcomes. Moreover, the platform provides filtering mechanisms that enable teachers and evaluators to select credentials and educational contributions within specific time intervals, further supporting compliance with institutional evaluation frameworks and professional development requirements.

In conclusion, the obtained results confirm that the proposed blockchain-based solution successfully overcomes the limitations of existing credentialing approaches, offering a combination of cost-effectiveness, scalability, security, and transparency. By integrating zk-Rollups, IPFS storage, and well-defined governance through smart contracts, the system provides a robust infrastructure capable of national-scale deployment. These results demonstrate that blockchain technology can serve as a practical and transformative tool for the digital management of teacher attestations and professional development records.

7.2. Original contributions

This research makes several original contributions to the field of educational technology, blockchain application, and national-level credential management. Three key contributions are identified:

- *Cost-effective credential and educational resources management*: A primary contribution is the demonstration of a blockchain solution that achieves substantial cost reduction while maintaining security and scalability. By using Polygon zkEVM and zk-Rollups, multiple credential issuance transactions are aggregated into single proofs, minimizing transaction fees and operational costs. This low-cost design enables sustainable adoption across public educational systems, where budgetary constraints often limit the implementation of new technologies. Unlike conventional Ethereum-based solutions, this approach allows credential management to scale nationwide without compromising economic feasibility.

BUPT

- *National-scale scalability and performance*: Another contribution lies in demonstrating the system's scalability for nationwide deployment. The solution achieves high throughput and rapid confirmation of credential issuance, supporting simultaneous management of teacher portfolios, professional development records, and institutional attestations across thousands of schools. The architecture ensures that large-scale adoption is operationally feasible without introducing delays or congestion, overcoming a major barrier commonly faced by blockchain-based educational solutions. This scalability is reinforced by batching transactions and cryptographic proof generation, which optimize both processing speed and on-chain storage.
- *Prototype development*: The prototype development integrated functionalities for issuing institutions, teachers and verifiers, providing a solution for digitizing the teacher evaluation process that is currently carried out by verifying paper-based documents. At the heart of these specifications are 2 smart contracts that function as the backbone of the system, governing the issuance, storage and secure verification of certificates, diplomas and other professional credentials. By leveraging the capabilities of the blockchain, the prototype guarantees both the accuracy and reliability of the registration and verification of credentials, thus covering the entire life cycle of professional documents within the educational ecosystem.
- *Alignment with European standards*: The third contribution addresses regulatory alignment. This alignment provides consistency with European initiatives for digital identity and credentialing, enhancing teacher mobility and recognition of professional development. Furthermore, the solution establishes a governance model where high-level authorities, institutions, and training providers operate within defined roles, ensuring accountability and verifiable issuance. This structured approach lays the foundation for standardization and integration with national and international educational policies.

7.3 Limitations

Despite its advantages, the proposed solution faces several limitations:

- *Adoption barriers*: Rapid adoption may be hindered by the fact that many teachers are unfamiliar with blockchain technology and do not currently own or use digital wallets. Training programs will be necessary to ensure successful onboarding.
- *System integration*: Integrating the platform with existing educational information systems, such as teacher databases, is challenging.
- *Regulatory framework*: Full adoption requires a legislative framework that formally recognizes blockchain-issued credentials. Without official legal recognition, schools may hesitate to adopt the system, limiting its impact on national teacher evaluation practices.

BUPT

7.4. Future work

Future research could build on the results of this thesis by extending the proposed blockchain-based framework to encompass the validation and management of professional development opportunities offered by accredited training institutions. An important avenue for exploration lies in the integration of artificial intelligence techniques with the blockchain framework. By analyzing teachers' professional trajectories, documented achievements, and areas of interest, AI systems could provide personalized recommendations for specialization courses, thereby supporting teachers in their continuous professional development. Such a development would not only expand the applicability of the system but also create a synergistic model in which blockchain ensures trust and immutability, while AI provides guidance for a more efficient and teacher-centered process of continuous professional development.

7.5. Publications

Diana Laura Silaghi; Daniela Elena Popescu, „A Systematic Review of Blockchain-Based Initiatives in Comparison to Best Practices Used in Higher Education Institutions," Computers, vol. 14, nr. 4, p. 141, 2025.

Diana Laura Silaghi, „Blockchain-based Solution for Protecting Teachers' Copyrights on Educational Resources," în International Conference on Engineering of Modern Electric Systems (ICEMES), Oradea, Romania, 2025.

Andrada Cristina Artenie; Diana Laura Silaghi; Daniela Elena Popescu, „Exploring the Synergy Between Ethereum Layer 2 Solutions and Machine Learning to Improve Blockchain Scalability," Computers, vol. 14, nr. 9, p. 359, 2025.

D. L. Silaghi, A. C. Artenie și D. E. Popescu, „Optimizing Teacher Portfolio Integrity with a Cost-Effective Smart Contract for School-Issued Teacher Documents," Computers, vol. 14, nr. 9, p. 395, 2025.

BUPT

Appendix A: The smart contract for educational resources management

```
// SPDX-License-Identifier: MIT
pragma solidity 0.8.30;

import "@openzeppelin/contracts/access/AccessControl.sol";

contract EducationalResourceContract is AccessControl {
    bytes32 private constant TEACHER_ROLE = keccak256("TEACHER_ROLE");

    struct EducationalResource {
        address resourceCreator;
        string ipfsHash;
        bool revoked;
    }

    mapping(uint256 => EducationalResource) private _resourcesById;
    mapping(address => uint256[]) private _resourcesByTeacher;
    mapping(string => bool) private _existingIpfsHashes;

    event EducationalResourceAdded(
        uint256 indexed id,
        address indexed resourceCreator,
        string ipfsHash
    );

    event EducationalResourceRevoked(
        uint256 indexed id,
        address indexed revokedBy,
        address indexed resourceCreator
    );

    event TeacherAdded(address indexed teacher);
    event TeacherRevoked(address indexed teacher);

    constructor(address _owner) {
        require(_owner != address(0), "Owner cannot be zero");
        _grantRole(DEFAULT_ADMIN_ROLE, _owner);
    }

    function addTeacher(address _teacher) external
onlyRole(DEFAULT_ADMIN_ROLE) {
        require(_teacher != address(0), "Invalid address");
        grantRole(TEACHER_ROLE, _teacher);
        emit TeacherAdded(_teacher);
    }

    function revokeTeacher(address _teacher) external
onlyRole(DEFAULT_ADMIN_ROLE) {
        require(_teacher != address(0), "Invalid address");
        require(hasRole(TEACHER_ROLE, _teacher), "Not teacher");
```


BUPT

```
        revokeRole(TEACHER_ROLE, _teacher);
        emit TeacherRevoked(_teacher);
    }

    function addEducationalResource(string calldata _ipfsHash, address
_resourceCreator) external onlyRole(DEFAULT_ADMIN_ROLE) {
        require(_resourceCreator != address(0), "Invalid address");
        require(!_existingIpfsHashes[_ipfsHash], "IPFS hash exists");

        _existingIpfsHashes[_ipfsHash] = true;

        uint256 resourceId = uint256(keccak256(abi.encode(block.timestamp,
msg.sender, _ipfsHash)));
        _resourcesById[resourceId] = EducationalResource({
            resourceCreator: _resourceCreator,
            ipfsHash: _ipfsHash,
            revoked: false
        });

        _resourcesByTeacher[_resourceCreator].push(resourceId);
        emit EducationalResourceAdded(resourceId, _resourceCreator,
_ipfsHash);
    }

    function getEducationalResource(uint256 _id) external view returns (
        address resourceCreator,
        string memory ipfsHash,
        bool revoked
    ) {
        require(
            hasRole(DEFAULT_ADMIN_ROLE, msg.sender) ||
hasRole(TEACHER_ROLE, msg.sender),
            "Access denied"
        );
        EducationalResource storage resource = _resourcesById[_id];
        resourceCreator = resource.resourceCreator;
        ipfsHash = resource.ipfsHash;
        revoked = resource.revoked;
    }

    function revokeEducationalResource(uint256 _id) external
onlyRole(DEFAULT_ADMIN_ROLE) {
        EducationalResource storage resource = _resourcesById[_id];
        require(!resource.revoked, "Already revoked");
        resource.revoked = true;
        emit EducationalResourceRevoked(_id, msg.sender,
resource.resourceCreator);
    }

    function getResourcesByTeacher(address _teacher, uint256 _start,
uint256 _count) external view returns (uint256[] memory) {
        require(
```


BUPT

```
            hasRole(DEFAULT_ADMIN_ROLE, msg.sender) || msg.sender ==
_teacher,
            "Access denied"
        );
        uint256 total = _resourcesByTeacher[_teacher].length;
        if (_start >= total) return new uint256 ;

        uint256 end = _start + _count;
        if (end > total) end = total;

        uint256[] memory result = new uint256[](end - _start);
        for (uint256 i = _start; i < end; i++) {
            result[i - _start] = _resourcesByTeacher[_teacher][i];
        }
        return result;
    }
}
```


BUPT

Appendix B: The smart contract for credentials management

```
// SPDX-License-Identifier: MIT
pragma solidity 0.8.30;

import "@openzeppelin/contracts/access/AccessControl.sol";

contract Credentials is AccessControl {
    bytes32 private constant _ADMIN_ROLE = keccak256("ADMIN_ROLE");

    struct Credential {
        address issuer_institution;
        address receiver;
        string ipfsHash;
        uint64 credentialDate;
        bool revoked;
    }

    mapping(uint256 => Credential) public credentials;
    mapping(address => uint256[]) private _credentialsByReceiver;
    mapping(string => bool) private _existingIpfsHashes;
    uint256 public nextId;

    event CredentialGenerated(
        uint256 indexed id,
        address indexed receiver,
        string ipfsHash,
        uint64 credentialDate,
        address indexed issuer_institution
    );

    event CredentialRevoked(
        uint256 indexed id,
        address indexed issuer_institution
    );

    event AdminAdded(address indexed admin);
    event AdminRevoked(address indexed admin);

    constructor(address owner) {
        require(owner != address(0), "Owner cannot be zero");
        _grantRole(DEFAULT_ADMIN_ROLE, owner);
    }

    modifier onlyAdmin() {
        require(
            hasRole(DEFAULT_ADMIN_ROLE, msg.sender) || hasRole(_ADMIN_ROLE,
msg.sender),
            "Access denied: only admin or owner can perform this action"
        );
        _;
    }
```


BUPT

```
    function addAdmin(address admin) external onlyRole(DEFAULT_ADMIN_ROLE)
{
        require(admin != address(0), "Invalid admin address");
        _grantRole(_ADMIN_ROLE, admin);
        emit AdminAdded(admin);
    }

    function revokeAdmin(address admin) external
onlyRole(DEFAULT_ADMIN_ROLE) {
        require(admin != address(0), "Invalid admin address");
        require(hasRole(_ADMIN_ROLE, admin), "Not an admin");
        _revokeRole(_ADMIN_ROLE, admin);
        emit AdminRevoked(admin);
    }

    function generateCredential(address receiver, string calldata ipfsHash)
external onlyAdmin {
        require(receiver != address(0), "Invalid receiver");
        require(!_existingIpfsHashes[ipfsHash], "IPFS hash already
exists");
        _existingIpfsHashes[ipfsHash] = true;

        uint256 id = ++nextId;
        credentials[id] = Credential({
            issuer_institution: msg.sender,
            receiver: receiver,
            ipfsHash: ipfsHash,
            credentialDate: uint64(block.timestamp),
            revoked: false
        });

        _credentialsByReceiver[receiver].push(id);
        emit CredentialGenerated(id, receiver, ipfsHash,
uint64(block.timestamp), msg.sender);
    }

    function revokeCredential(uint256 id) external {
        Credential storage c = credentials[id];
        require(bytes(c.ipfsHash).length != 0, "Credential does not
exist");
        require(!c.revoked, "Already revoked");

        require(
            msg.sender == c.issuer_institution ||
hasRole(DEFAULT_ADMIN_ROLE, msg.sender),
            "Access denied: only issuer or owner can revoke"
        );

        c.revoked = true;
        emit CredentialRevoked(id, msg.sender);
    }
```


BUPT

```
    function getCredential(uint256 id) external view returns (Credential
memory c) {
        c = credentials[id];
        require(bytes(c.ipfsHash).length != 0, "Credential does not
exist");
    }

    function getReceiverCredentials(address receiver) external view returns
(uint256[] memory) {
        uint256[] storage ids = _credentialsByReceiver[receiver];
        require(ids.length != 0, "No credentials");
        return ids;
    }

    function isVerified(uint256 id) external view returns (bool) {
        Credential storage c = credentials[id];
        return bytes(c.ipfsHash).length != 0 && !c.revoked;
    }
}
```


BUPT

Bibliography

[1] R. Bucea-Manea-Țoniș, D. G. Păun, O. M. D. Martins and A. J. Santos, "Education for Sustainable Development: What Matters?," *Sustainability*, vol. 16, no. 21, p. 9493, 2024.

[2] L. Gan and R. Lam, "Unveiling Classroom Assessment Literacy: Does Teachers' Self-Directed Development Play Out?," *Education Sciences*, vol. 14, no. 9, p. 961, 2024.

[3] D. L. Silaghi, "Blockchain-based Solution for Protecting Teachers' Copyrights on Educational Resources," in *International Conference on Engineering of Modern Electric Systems (ICEMES)*, Oradea, Romania, 2025.

[4] Ministry of Education and Research, "Activity and performance evaluation," [Online]. Available: https://www.edu.ro/evaluare-activitate. [Accessed 18 May 2025].

[5] Ministry of Education and Research, "Specific procedure regarding the operationalization of methodological regulations regarding the accumulation by teachers of 90 transferable professional credits," [Online]. Available: https://www.edu.ro/sites/default/files/_fi%C8%99iere/Minister/2022/inv.%20preuniversitar/note_proceduri/formare/Procedura%20specifica%20acumulare%2090%20CPT.pdf. [Accessed 18 May 2025].

[6] "The methodology and criteria for awarding the merit-based distinction to teaching staff in public pre-university education in the 2025 session," [Online]. Available: https://www.edu.ro/sites/default/files/_fi%C8%99iere/Legislatie/2025/OMEC_3745_2025_Gradatie_merit.pdf. [Accessed 30 June 2025].

[7] "Teacher's portofolio," [Online]. Available: https://lege5.ro/Gratuit/geztkmbqgu/portofoliul-profesorului-metodologie?dp=gqztanzxgq4ta. [Accessed 30 June 2025].

[8] P. Colás-Bravo, P. Magnoler and J. Conde-Jiménez, "Identification of Levels of Sustainable Consciousness of Teachers in Training through an E-Portfolio," *Sustainability*, vol. 10, no. 10, p. 3700, 2018.

[9] E. Durant and A. Trachy, "Digital Diploma debuts at MIT," 17 October 2017. [Online]. Available: https://news.mit.edu/2017/mit-debuts-secure-digital-diploma-using-bitcoin-blockchain-technology-1017. [Accessed 26 May 2024].

[10] R. Bucea-Manea-Țoniș, O. M. D. Martins, R. Bucea-Manea-Țoniș, C. Gheorghiță, V. Kuleto, M. P. Ilić and V.-E. Simion, "Blockchain Technology Enhances Sustainable Higher Education," *Sustainability*, vol. 13, no. 22, p. 12347, 2021.

[11] O. S. Saleh, O. Ghazali and M. E. Rana, "Blockchain based framework for educational certificates verification," *Journal of Critical Reviews*, vol. 7, no. 3, 2020.

[12] S. A. Ahmed, R. A. M. Rudro, A. J. Prity, S. Saha, N. Mansoor and K. Nur, "CredChain: Academic and Professional Certificate Verification System using Blockchain," in *International Conference on Advances in Computing, Communication, Electrical, and Smart Systems*, Dhaka, Bangladesh, 2024.

[13] M. A. Alkhawi and A. S. Alshameri, "Cost-Effective and Efficient Blockchain Framework for Verifying Certificate in Yemeni Universities," *Turkish Journal of Computer and Mathematics Education*, vol. 15, no. 2, pp. 206-216, 2024.

[14] Y. Kistaubayev, G. Mutanov, M. Mansurova, Z. Saxenbayeva and Y. Shakan, "Ethereum-Based Information System for Digital Higher Education Registry and Verification of Student Achievement Documents," *Future Internet*, vol. 15, no. 1, p. 3, 2023.

BUPT

[15] M. Litoussi, M. Fartitchoub, K. E. Makkaouib, A. Ezzatia and Z. E. Allalib, "Digital Certifications in Moroccan Universities: Concepts, Challenges and Solutions," in *Procedia Computer Science*, Porto, 2022.

[16] E. K. Nzaro, K. Ondimu and F. Mwakondo, "A Blockchain-Based Conceptual Model for Curbing Institutional Academic Certificate Fraud," *International Journal of Computer Applications Technology and Research,* vol. 11, no. 7, pp. 255-263, 2022.

[17] C. Kontzinos, E. Karakolis, P. Kokkinakos, S. Skalidakis and D. A. J. Psarras, "Application and Evaluation of a Blockchain-Centric Platform forSmart Badge Accreditation in Higher Education Institutions," *Appl. Sci.,* vol. 14, no. 12, p. 5191, 2024.

[18] M. Turkanović, M. Hölbl, K. Košič, M. Heričko and A. Kamišalić, "EduCTX: A Blockchain-Based Higher Education Credit Platform," *IEEE Access ,* vol. 6, pp. 5112 - 5127, 2018.

[19] E. Tan, E. Lerouge, J. D. Caju and D. D. Seuil, "Verification of Education Credentials on European Blockchain Services Infrastructure (EBSI): Action Research in a Cross-Border Use Case between Belgium and Italy," *Big Data and Cognitive Computing,* vol. 7, no. 2, p. 79, 2023.

[20] V. Holotescu, R. Vasiu, A. Ternauciuc, C. Holotescu and D. Andone, "Building a Decentralized Education Ecosystem: Politehnica University of Timisoara's Pioneering Blockchain Initiatives," in *EDEN 2024 Research Workshop & PhD Masterclass*, Timisoara, Romania, 2024.

[21] "Modele de cereri și anexe pentru mișcarea personalului didactic în anul școlar 2024 - 2025," [Online]. Available: https://www.isjbihor.ro/resurse/modele-de-cereri/2306- modele-de-cereri-si-anexe-pentru-miscarea-personalului-didactic-in-anul-scolar-2024-2025. [Accessed 23 July 2025].

[22] F. R. Vidal, F. Gouveia and C. Soares, "Revocation Mechanisms for Academic Certificates Stored on a Blockchain," in *15th Iberian Conference on Information Systems and Technologies (CISTI)*, Seville, Spain, 2020.

[23] "Blockchain," [Online]. Available: https://ro.wikipedia.org/wiki/Blockchain. [Accessed 2023 December 30].

[24] Techskill Brew, "What is a block in the blockchain?," [Online]. Available: https://medium.com/techskill-brew/what-is-a-block-in-the-blockchain-part-2-blockchain-basics-53ad20c766cc. [Accessed 10 February 2024].

[25] D. Khan, L. T. Jung and M. A. Hashmani, "Systematic Literature Review of Challenges in Blockchain Scalability," *Applied Sciences,* vol. 11, no. 20, p. 9372, 2021.

[26] "What is blockchain technology and how does it work?," [Online]. Available: https://kriptomat.io/ro/blockchain/ce-este-tehnologia-blockchain/. [Accessed 2023 December 30].

[27] D. L. Silaghi and D. E. Popescu, "A Systematic Review of Blockchain-Based Initiatives in Comparison to Best Practices Used in Higher Education Institutions," *Computers*, vol. 14, no. 4, p. 141, 2025.

[28] C. Delgado-von-Eitzen, L. Anido-Rifón and M. J. Fernández-Iglesias, "Blockchain Applications in Education: A Systematic Literature Review," *Applied Sciences,* vol. 11, no. 24, p. 11811, 2021.

[29] A. S. Alammary, "Building a Sustainable Digital Infrastructure for Higher Education: A Blockchain-Based Solution for Cross-Institutional Enrollment," *Sustainability,* vol. 17, no. 1, p. 194, 2025.

BUPT

[30] E. Guustaaf, U. Rahardja, Q. Aini, H. W. Maharani and N. A. Santoso, "Blockchain-based Education Project," *Aptisi Transactions on Management (ATM)*, vol. 5, no. 1, pp. 46-61, 2021.

[31] "About Dragonchain," [Online]. Available: https://dragonchain.com/about/. [Accessed 25 January 2025].

[32] X. Chen, S. He, L. Sun, Y. Zheng and C. Q. Wu, "A Survey of Consortium Blockchain and Its Applications," *Cryptography*, vol. 8, no. 2, p. 12, 2024.

[33] "Consortium Blockchain: What You Need to Know," [Online]. Available: https://www.kaleido.io/blockchain-blog/consortium-blockchain. [Accessed 5 April 2025].

[34] B. Lashkari and P. Musilek, "A Comprehensive Review of Blockchain Consensus Mechanisms," *IEEE Access*, vol. 9, pp. 43620 - 43652, 2021.

[35] A. C. Artenie, D. L. Silaghi and D. E. Popescu, "Exploring the Synergy Between Ethereum Layer 2 Solutions and Machine Learning to Improve Blockchain Scalability," *Computers*, vol. 14, no. 9, p. 359, 2025.

[36] "List of 10 Types of Consensus Mechanism with Examples," Blockchain Lab, [Online]. Available: https://medium.com/@theblockchains/list-of-10-types-of-consensus-mechanism-with-examples-bf65bd752967. [Accessed 17 May 2025].

[37] A. Gangwal, H. R. Gangavalli and A. Thirupathi, "A Survey of Layer-Two Blockchain Protocols," *Journal of Network and Computer Applications*, vol. 209, no. 11, p. 103539, 2022.

[38] ExplainCKBot, "Crypto Mining Hardware: Exploring CPU's, GPU's, and ASIC's," [Online]. Available: https://www.nervos.org/knowledge-base/crypto_mining_hardware_%28explainCKBot%29. [Accessed 18 February 2025].

[39] Cryptopedia, "Crypto Mining Rigs Explained: From CPUs to GPUs to ASICs," [Online]. Available: https://www.gemini.com/cryptopedia/crypto-mining-rig-bitcoin-mining-calculator-asic-miner. [Accessed 18 February 2025].

[40] "How do Bitcoin transactions work?," [Online]. Available: https://www.bitcoin.com/get-started/how-bitcoin-transactions-work/. [Accessed 13 April 2025].

[41] Bitcoin.com, "How do Bitcoin transactions work?," [Online]. Available: https://www.bitcoin.com/get-started/how-bitcoin-transactions-work/. [Accessed 21 February 2025].

[42] Bitcoin.com, "How do Bitcoin transactions work?," [Online]. Available: https://www.bitcoin.com/get-started/how-bitcoin-transactions-work/. [Accessed 21 April 2025].

[43] "When Is the Next Bitcoin Halving?," [Online]. Available: https://www.binance.com/en/events/bitcoin-halving. [Accessed 10 July 2025].

[44] "Mining," [Online]. Available: https://bitcoin.org/en/faq#mining. [Accessed 11 February 2025].

[45] "What is Ethereum?," [Online]. Available: https://www.coinbase.com/learn/crypto-basics/what-is-ethereum. [Accessed 10 May 2025].

[46] S. Kayikci and T. M. Khoshgoftaar, "Blockchain meets machine learning: a survey," *Journal of Big Data volume*, 2024.

[47] E. Kapengut and B. Mizrach, "An Event Study of the Ethereum Transition to Proof-of-Stake," *Commodities*, vol. 2, no. 2, pp. 96-110, 2023.

[48] "Ethereum Pectra Upgrade Enters Final Testing Phase with Sepolia Activation," [Online]. Available: https://cryptorank.io/news/feed/7b721-ethereum-pectra-upgrade-enters-final-testing-phase-with-sepolia-activation. [Accessed 10 August 2025].

BUPT

[49] S. S. Kushwaha, S. Joshi, D. Singh, M. Kaur and H.-N. Lee, "Ethereum Smart Contract Analysis Tools: A Systematic Review," *IEEE Access* , vol. 10, pp. 57037 - 57062, 2022.

[50] C. Wu, J. Xiong, H. Xiong, Y. Zhao and W. Yi, "A Review on Recent Progress of Smart Contract in Blockchain," *IEEE Access*, vol. 10, pp. 50839 - 50863, 2022.

[51] K. Tan-Vo, K. Pham, P. Huynh, M.-T. N. Thi, T.-T. Ta, T. Nguyen, T.-A. Nguyen-Hoang, N.-T. Dinh and H.-T. Nguyen, "Optimizing Academic Certificate Management With Blockchain and Machine Learning: A Novel Approach Using Optimistic Rollups and Fraud Detection," *IEEE Access*, vol. 13, p. 168135, 2024.

[52] A. M. Dyade, M. A. Dyade and S. M. Shinde, "Cost Optimization in Layer 2 Rollups via EIP-4844: A GasEfficiency and Economic Analysis," *IET Blockchain*, vol. 5, no. 1, 2025.

[53] A. Mohammad and S. Vargas, "Challenges of Using Blockchain in the Education Sector: A Literature Review," *Applied Sciences*, vol. 12, no. 13, p. 6380, 2022.

[54] A. A. Khan, A. A. Laghari, A. A. Shaikh, S. Bourouis, A. M. Mamlouk and H. Alshazly, "Educational Blockchain: A Secure Degree Attestation and Verification Traceability Architecture for Higher Education Commission," *Applied Sciences*, vol. 11, no. 22, p. 10917, 2021.

[55] P. Dias, H. Gonçalves, F. Silva, J. Duque, J. Martins and A. Godinho, "Blockchain Technologies: A scrutiny into Hyperledger Fabric for Higher Educational Institutions," *Procedia Computer Science*, vol. 237, pp. 213-220, 2024.

[56] D. Mourtzis, J. Angelopoulos and N. Panopoulos, "Blockchain Integration in the Era of Industrial Metaverse," *Applied Sciences*, vol. 13, no. 3, p. 1353, 2023.

[57] M. Thanasi-Boçe and J. Hoxha, "Blockchain for Sustainable Development: A Systematic Review," *Sustainability*, vol. 17, no. 11, p. 4848, 2025.

[58] W. Li, J. Bu, X. Li, H. Peng, Y. Niu and Y. Zhang, "A survey of DeFi security: Challenges and opportunities," *Journal of King Saud University - Computer and Information Sciences*, vol. 34, no. 10, pp. 10378-10404, 2022.

[59] S. S. M. Abdul, A. Shrestha and J. Yong, "CrossDeFi: A Novel Cross-Chain Communication Protocol," *Future Internet*, vol. 16, no. 9, p. 314, 2024.

[60] A. Alamsyah, G. N. W. Kusuma and D. P. Ramadhani, "A Review on Decentralized Finance Ecosystems," *Future Internet*, vol. 16, no. 3, p. 76, 2024.

[61] P. Schueffel, "DeFi: Decentralized Finance - An Introductionand Overview," *Journal of Innovation Management*, vol. 9, no. 3, pp. 1-11, 2021.

[62] "What Is Sablier?," [Online]. Available: https://docs.sablier.com/concepts/what-is-sablier. [Accessed 9 August 2025].

[63] "What is Connext?," [Online]. Available: https://docs.connext.network/. [Accessed 9 August 2025].

[64] C. Munoz-Ausecha, J. E. G. Gómez, J. Ruiz-Rosero and G. Ramirez-Gonzalez, "Asset Ownership Transfer and Inventory Using RFID UHF TAGS and Ethereum Blockchain NFTs," *Electronics*, vol. 12, no. 6, p. 1497, 2023.

[65] V. Mohan, "Automated market makers and decentralized exchanges: a DeFi primer," *Financial Innovation*, vol. 8, no. 1, p. 20, 2022.

[66] J. Zheng, Q. Chen, C. Su and H. Huang, "BrokerFi: A DeFi dApp Built upon Broker-based Blockchain," in *IEEE 29th International Conference on Parallel and Distributed Systems (ICPADS)*, Ocean Flower Island, China, 2023.

BUPT

[67] Q. Lu, A. B. Tran, I. Weber, H. O'Connor, P. Rimba, X. Xu, M. Staples, L. Zhu and R. Jeffery, "Integrated model-driven engineering of blockchain applications for business processes and asset management," *Journal of Software: Practice and Experience*, vol. 51, no. 5, pp. 1059-1079, 2021.

[68] "Top 12 DeFi Platforms To Look Out For: Comparison Guide," [Online]. Available: https://www.rapidinnovation.io/post/top-12-defi-platforms-to-look-out-for-comparison-guide. [Accessed 8 August 2025].

[69] "Delegate Your Stake," [Online]. Available: https://cardano.org/stake-pool-delegation/. [Accessed 8 August 2025].

[70] J. Xu and Y. Feng, "Reap the Harvest on Blockchain: A Survey of Yield Farming Protocols," *IEEE Transactions on Network and Service Management*, vol. 20, no. 1, pp. 858 - 869, 2023.

[71] S. Seven, Y. Yoldas, A. Soran, G. Y. Alkan, J. Jung, T. S. Ustun and A. Onen, "Energy Trading on a Peer-to-Peer Basis between Virtual Power Plants Using Decentralized Finance Instruments," *Sustainability*, vol. 14, no. 20, p. 13286, 2022.

[72] "Pangolin," [Online]. Available: https://build.avax.network/integrations/pangolin. [Accessed 10 August 2025].

[73] M. Shaikh, S. A. Memon, A. Ebrahimi and U. K. Wiil, "A Systematic Literature Review for Blockchain-Based Healthcare Implementations," *Healthcare*, vol. 13, no. 9, p. 1087, 2025.

[74] †. R. 1. J. H. by Noor Ul Ain Tahir 1, N. Ahmad, Y. Cao, M. A. Alshara and Y. Javed, "Blockchain-Based Healthcare Records Management Framework: Enhancing Security, Privacy, and Interoperability," *Technologies*, vol. 12, no. 9, p. 168, 2024.

[75] K. Abbas, M. Afaq, T. A. Khan and W.-C. Song, "A Blockchain and Machine Learning-Based Drug Supply Chain Management and Recommendation System for Smart Pharmaceutical Industry," *Electronics*, vol. 9, no. 5, p. 852, 2020.

[76] M. Johnson, M. Jones, M. Shervey, J. T. Dudley and N. Zimmerman, "Building a Secure Biomedical Data Sharing Decentralized App (DApp): Tutorial," *Journal of Medical Internet Research*, vol. 21, no. 10, 2019.

[77] T. Rathod, N. K. Jadav, S. Tanwar, R. Sharma, A. Tolba, M. S. Raboaca, V. Marina and W. Said, "Blockchain-Driven Intelligent Scheme for IoT-Based Public Safety System beyond 5G Networks," *Sensors*, vol. 23, no. 2, p. 969, 2023.

[78] A. Enaya, X. Fernando and R. Kashef, "Survey of Blockchain-Based Applications for IoT," *Applied Sciences*, vol. 15, no. 8, p. 4562, 2025.

[79] M. A. Obaidat, M. Rawashdeh, M. Alja'afreh, M. Abouali, K. Thakur and A. Karime, "Exploring IoT and Blockchain: A Comprehensive Survey on Security, Integration Strategies, Applications and Future Research Directions," *Big data and cognitive computing*, vol. 8, no. 12, p. 174, 2024.

[80] A. Djenna, S. Harous and D. E. Saidouni, "Internet of Things Meet Internet of Threats: New Concern Cyber Security Issues of Critical Cyber Infrastructure," *Applied Sciences*, vol. 11, no. 10, p. 4580, 2021.

[81] Y. Teng, L. Li, L. Song, F. R. Yu and V. C. M. Leung, "Profit Maximizing Smart Manufacturing Over AI-Enabled Configurable Blockchains," *IEEE Internet of Things Journal*, vol. 9, pp. 346 - 358, 2021.

[82] Y. Wang, Z. Su, N. Zhang, J. Chen, X. Sun and Z. Ye, "SPDS: A Secure and Auditable Private Data Sharing Scheme for Smart Grid Based on Blockchain," *IEEE Transactions on Industrial Informatics*, vol. 17, pp. 7688 - 7699, 2021.

BUPT

[83] G. Piccardo, L. Conti and A. Martino, "Blockchain Technology and Its Potential to Benefit Public Services Provision: A Short Survey," *Future Internet*, vol. 16, no. 8, p. 290, 2024.

[84] D. Cagigas, J. Clifton, D. Diaz-Fuentes and M. Fernandez-Gutierrez, "Blockchain for Public Services: A Systematic Literature Review," *IEEE Access*, vol. 9, pp. 13904-13921, 2021.

[85] I. R. Abdelhamid, I. T. A. Halim, I. A. Ibrahim and A. E.-M. A. Ali, "Redefining Governmental Services Through Blockchain and Smart Contracts," *Mathematical Modelling of Engineering Problems*, vol. 10, no. 5, pp. 1515-1528, 2023.

[86] G. Sladić, B. Milosavljević, S. Nikolić, D. Sladić and A. Radulović, "A Blockchain Solution for Securing Real Property Transactions: A Case Study for Serbia," *ISPRS International Journal of Geo-Information*, vol. 10, no. 1, p. 35, 2021.

[87] C. Delgado-von-Eitzen, L. Anido-Rifón and M. J. Fernández-Iglesias, "Blockchain Applications in Education: A Systematic Literature Review," *Applied Sciences*, vol. 11, no. 24, p. 11811, 2021.

[88] G. Zhao, H. He, B. Di and Q. Guo, "BC-DERCP: Blockchain-based copyright protection mechanism for digital educational resources," *Education and Information Technologies* , vol. 29, p. 19679–19709, 2024.

[89] K. Bálint, D. Cvetković, M. Takács, Ildikó and A. Tóth, "Connecting Bitcoin Blockchain with Digital Learning Chain Structure in Education," *Acta Polytechnica Hungarica*, vol. 16, no. 1, p. 77, 2019.

[90] A. Heredia, M.-J. Barros and G. Barros-Gavilanes, "Decentralizing Certificates Issuance Through Blockchain," in *International Conference on Electrical, Computer and Energy Technologies*, Cape Town, South Africa, 2022.

[91] P. Boulet, P. d. Coëtlogon, É. Thébault and P. v. d. Velde, "Digital transformation of certificates issued by universities for European competitiveness," 6 March 2025. [Online]. Available: https://hal.science/hal-04979478/. [Accessed 12 August 2025].

[92] "Academic degree," [Online]. Available: https://en.wikipedia.org/wiki/Academic_degree. [Accessed 12 August 2025].

[93] "5 Types of Academic Degrees: Which Is Right for You?," [Online]. Available: https://www.indeed.com/career-advice/career-development/types-of-academic-degrees. [Accessed 12 August 2025].

[94] "Learn The Difference Between Certificate, Credential & Degree," [Online]. Available: https://diplomasafe.com/credential-vs-certificate/. [Accessed 12 August 2025].

[95] "Badges and Microcredentials – What's the Difference?," [Online]. Available: https://post.edu/blog/badges-and-microcredentials-whats-the-difference/. [Accessed 12 August 2025].

[96] "Guide to Using Digital Badges for Recognition of Skills and Achievements," [Online]. Available: https://certifier.io/blog/using-digital-badges-for-recognition-of-skills-and-achievements. [Accessed 12 August 2025].

[97] "Attestation et Portfolio de L'Engagement (Engagement Portfolio)," [Online]. Available: https://pjp-eu.coe.int/documents/42128013/47262121/PortfolioEngagment.pdf/169a646f-3998-488b-b000-b97b8a74153d?t=1427197994000. [Accessed 12 August 2025].

[98] "https://www.uoradea.ro," [Online]. Available: https://www.uoradea.ro/Birou+eliberare+diplome. [Accessed 30 October 2024].

BUPT

[99] "https://www.uoradea.ro/," [Online]. Available: https://www.uoradea.ro/display32352. [Accessed 30 October 2024].

[100] M. Caramihai and I. Severin, "A Blockchain-Based Solution for Diploma Management in Universities," *Sustainability*, vol. 15, no. 20, p. 15169, 2023.

[101] "The digital signature standard," [Online]. Available: https://dl.acm.org/doi/pdf/10.1145/129902.129904. [Accessed 16 August 2025].

[102] H. R. Penubadi, P. Shah, R. Sekhar, M. N. Alrasheedy, Y. Niu, A. D. Radhi, M. Tharwat, J. F. Tawfeq, H. M. Gheni and Azmi, "Sustainable electronic document security: a comprehensive framework integrating encryption, digital signature and watermarking algorithms," *Heritage and Sustainable Development*, vol. 5, no. 2, pp. 391-404, 2023.

[103] "Order no. 4634 of June 10, 2024," [Online]. Available: https://www.edu.ro/sites/default/files/_fi%C8%99iere/Legislatie/2024/Ordin_4634_2024.pdf. [Accessed 15 August 2025].

[104] C. Shaik, "Preventing Forged and Fabricated Academic Credentials using Cryptography and QR Codes," *International Journal of Computer Science, Engineering and Applications*, vol. 11, no. 1, p. 11, 2021.

[105] H. A. Ahmed and J. W. Jang, "Document Certificate Authentication System Using Digitally Signed QR Code Tag," in *Proceedings of the 12th International Conference on Ubiquitous Information Management and Communication*, Langkawi, Malaysia, 2018.

[106] D. H. Nguyen, D. N. Nguyen-Duc, N. Huynh-Tuong and H. A. Pham, "CVSS: A Blockchainized Certificate Verifying Support System," in *SoICT '18: Proceedings of the 9th International Symposium on Information and Communication Technology*, Da Nang City, Vietnam, 2018.

[107] J. Gresch, B. Rodrigues, E. Scheid, S. S. Kanhere and B. Stiller, "The Proposal of a Blockchain-based Architecture for Transparent Certificate Handling," *Proc. Int. Workshops Bus. Inf. Syst. Workshops (BIS)*, vol. 339, pp. 185-196, 2018.

[108] W. Gräther, S. Kolvenbach, R. Ruland, J. Schütte, C. F. Torres and F. Wendland, "Blockchain for Education: Lifelong Learning Passport," in *European Society for Socially Embedded Technologies (EUSSET)*, Amsterdam, Netherlands, 2018.

[109] F. Schär and F. Mösli, "Blockchain Diplomas: Using Smart Contracts to Secure Academic Credentials," *Beiträge zur Hochschulforschung*, vol. 41, no. 3, pp. 48-58, 2019.

[110] A. Tariq, H. B. Haq and S. T. Ali, "Cerberus: A Blockchain-Based Accreditation and Degree Verification System," *IEEE Transactions on Computational Social Systems*, vol. 10, no. 4, pp. 1503 - 1514, 2023.

[111] T. Kanan, A. T. Obaidat and M. Al-Lahham, "SmartCert BlockChain Imperative for Educational Certificates," in *Jordan International Joint Conference on Electrical Engineering and Information Technology (JEEIT)*, Amman, Jordan, 2019.

[112] C. Li, J. Guo, G. Zhang, Y. Wang, Y. Sun and R. Bie, "A Blockchain System for E-Learning Assessment and Certification," in *IEEE International Conference on Smart Internet of Things (SmartIoT)*, Tianjin, China, 2019.

[113] H. Li and D. Han, "EduRSS: A Blockchain-Based Educational Records Secure Storage and Sharing Scheme," *IEEE Access*, vol. 7, pp. 179273 - 179289, 2019.

[114] R. J. Maestre, J. B. Higuera, N. G. Gómez, J. R. B. Higuera, J. A. S. Montalvo and L. O. Palma, "The application of blockchain algorithms to the management of education certificates," *Evol. Intel.*, vol. 16, p. 1967–1984, 2023.

BUPT

[115] "Blockchain for digital certificates: How it's used to ensure credential security and verification," [Online]. Available: https://diplomasafe.com/blockchain-certification-process/. [Accessed 24 April 2025].

[116] R. Q. Castro and M. Au-Yong-Oliveira, "Blockchain and Higher Education Diplomas," *Eur. J. Investig. Health Psychol. Educ,* vol. 11, no. 1, pp. 154-167, 2021.

[117] E. P. Fedorova and E. I. Skobleva, "Application of Blockchain Technology in Higher Education," *European Journal of Contemporary Education,* vol. 9, no. 3, pp. 552-571, 2020.

[118] G. Caldarelli and J. Ellul, "Trusted Academic Transcripts on the Blockchain: A Systematic Literature Review," *Appl. Sci.*, vol. 11, no. 4, p. 1842, 2021.

[119] M. Elkhodr, K. Wangsa, Ergun and S. Karim, "A Systematic Review and Multifaceted Analysis of the Integration of Artificial Intelligence and Blockchain: Shaping the Future of Australian Higher Education," *Future Internet,* vol. 16, no. 10, p. 378, 2024.

[120] A. T. Elizabeth Durant, "Digital Diploma debuts at MIT," 17 Octombrie 2017. [Online]. Available: https://news.mit.edu/2017/mit-debuts-secure-digital-diploma-using-bitcoin-blockchain-technology-1017. [Accessed 26 Ianuarie 2024].

[121] G. Capece, N. L. Ghiron and F. Pasquale, "Blockchain Technology: Redefining Trust for Digital Certificates," *Sustainability,* vol. 12, no. 21, p. 8952, 2020.

[122] F. R. Vidal, F. Gouveia and C. Soares, "Blockchain Application in Higher Education Diploma Management and Results Analysis," *Advances in Science, Technology and Engineering Systems,* vol. 5, no. 6, pp. 871-882, 2020.

[123] "About Blockcerts," [Online]. Available: https://www.blockcerts.org/about.html. [Accessed 2 January 2025].

[124] M. Jirgensons and J. Kapenieks, "Blockchain and the future of digital learning credential, assessment and management," *Journal of Teacher Education for Sustainability,* vol. 20, no. 1, pp. 145-156, 2018.

[125] "About Digital Diplomas," Maryville University, [Online]. Available: https://www.maryville.edu/digitaldiploma. [Accessed 6 August 2024].

[126] M. Themistocleous, K. Christodoulou and E. Iosif, "Academic certificates issued on blockchain - The case of the University of Nicosia and Block.co," in *Supporting Higher Education 4.0 with Blockchain*, 2023.

[127] P. Thevoz, "How can anyone check the authenticity of my University Certificate on the Blockchain ?," 11 Iulie 2016. [Online]. Available: https://medium.com/@PhilippeThevoz/how-can-anyone-check-the-authenticity-of-my-university-certificate-on-the-blockchain-7c3003b993d4. [Accessed 21 Ianuarie 2024].

[128] T. Tahlil, S. S. Gomasta and A. B. M. S. Ali, "AlgoCert: Adopt Non-transferable NFT for the Issuance and Verification of Educational Certificates using Algorand Blockchain," in *IEEE Asia-Pacific Conference on Computer Science and Data Engineering*, Gold Coast, Australia, 2022.

[129] M. Choi, S. R. Kiran, S.-C. Oh and O.-Y. Kwon, "Blockchain-Based Badge Award with Existence Proof," *Applied Sciences,* vol. 9, no. 12, p. 2473, 2019.

[130] "What are OPEN BADGES?," [Online]. Available: https://openbadges.org/. [Accessed 29 March 2025].

[131] M. Baldi, F. Chiaraluce, M. Kodra and L. Spalazzi, "Security analysis of a blockchain-based protocol for the certification of academic credentials," 2019. [Online]. Available: https://doi.org/10.48550/arXiv.1910.04622.

BUPT

[132] S. Gangwar and A. Chaurasia, "Blockchain-based Authentication and Verification System for Academic Certificate using QR Code and Decentralized Applications," *International Journal of Computer Applications*, vol. 186, no. 26, 2024.

[133] M. Fartitchou, I. Lamaakal, K. E. Makkaoui, Z. E. Allali and Y. Maleh, "BlockMEDC: Blockchain Smart Contracts System for Securing Moroccan Higher Education Digital Certificates," *IEEE Access*, vol. 13, pp. 39152 - 39175, 2025.

[134] "Self-assessment/evaluation framework sheet for the purpose of awarding the annual qualification for teachers in pre-university education," [Online]. Available: https://www.scoalatulca.ro/wp-content/uploads/2023/09/F07-FISA-DE-AUTOEVALUARE-EVALUARE-CADRU-DIDACTIC.pdf. [Accessed 10 April 2025].

[135] "Procedure regarding the establishment of teaching-learning-evaluation didactic norms in pre-university education, in the 2025-2026 school year, as a result of the application of the measures adopted by Law No. 141/2025 on some fiscal-budgetary measures," [Online]. Available: https://www.edu.ro/sites/default/files/Proiect_Procedura_constituire_NDP.pdf. [Accessed 5 September 2025].

[136] "Evaluation form for ranking candidates participating in the teaching staff mobility stages for the 2025-2026 school year," [Online]. Available: https://www.isj.tm.edu.ro/public/data_files/specializari/fisier-6445.pdf. [Accessed 11 April 2025].

[137] [Online]. Available: https://www.isjcj.ro/wp-content/uploads/2025/04/Fisa-de-gradatie-de-merit-PROFESORI-TIC.pdf. [Accessed 7 July 2025].

[138] "Merit rankings 2025," [Online]. Available: https://www.isjbihor.ro/resurse/gradatii-de-merit/2525-2025-gradatii-de-merit-2025. [Accessed 1 July 2025].

[139] G. Deep, J. Sidhu and R. Mohana, "Insider Threat Prevention in Distributed Database as a Service Cloud environment," *Computers & Industrial Engineering*, vol. 169, no. 4, p. 108278, 2022.

[140] G. Habib, S. Sharma, S. Ibrahim, I. Ahmad, S. Qureshi and M. Ishfaq, "Blockchain Technology: Benefits, Challenges, Applications, and Integration of Blockchain Technology with Cloud Computing," *Future Internet*, vol. 14, no. 11, p. 341, 2022.

[141] A. Parsovs, "Solving the Estonian ID Card Crisis: the Legal Issues," in *Proceedings of the 17th ISCRAM Conference*, Blacksburg, USA, 2020.

[142] S. Semenzin, D. Rozas and S. Hassan, "Blockchain-based application at a governmental level: disruption or illusion? The case of Estonia," *Policy and Society*, vol. 41, no. 3, p. 386–401, 2022.

[143] J. A. B. Moya, J. Ayoade and M. A. Uddin, "A Zero-Knowledge Proof-Enabled Blockchain-Based Academic Record Verification System," *Sensors*, vol. 25, no. 11, p. 3450, 2025.

[144] D. L. Fekete and A. Kiss, "Toward Building Smart Contract-Based Higher Education Systems Using Zero-Knowledge Ethereum Virtual Machine," *Electronics*, vol. 12, no. 3, p. 664, 2023.

[145] "Ethereum development environment for professionals," [Online]. Available: https://hardhat.org/. [Accessed 4 August 2025].

[146] "Run JavaScript Everywhere," [Online]. Available: https://nodejs.org/en. [Accessed 15 January 2025].

[147] "Hardhat network," [Online]. Available: https://hardhat.org/hardhat-network/docs/overview. [Accessed 11 July 2025].

BUPT

[148] "Ethereum Holesky: All You Need to Know and How to Use It," [Online]. Available: https://blog.quicknode.com/ethereum-holesky-testnet-all-you-need-to-know/. [Accessed 5 February 2025].

[149] "What is Sepolia? A Beginner's Guide to Ethereum Test Networks," [Online]. Available: https://getblock.medium.com/what-is-sepolia-a-beginners-guide-to-ethereum-test-networks-866663a26698. [Accessed 31 July 2025].

[150] C. McCabe, A. I. C. Mohideen and R. Singh, "A Blockchain-Based Authentication Mechanism for Enhanced Security," *Sensors*, vol. 24, no. 17, p. 5830, 2024.

[151] "MetaMask: The Beginner's Guide," [Online]. Available: https://medium.com/@decryptmedia/metamask-the-beginners-guide-6111143f2581. [Accessed 19 August 2025].

[152] R. Singh, Z. Pervez and H. Tewari, "Blockchain-Enabled NextGen Service Architecture for Mobile Internet Offload," *Future Internet*, vol. 15, no. 5, p. 173, 2023.

[153] M. Pacheco, G. Oliva, G. K. Rajbahadur and A. Hassan, "Is My Transaction Done Yet? An Empirical Study of Transaction Processing Times in the Ethereum Blockchain Platform," *ACM Transactions on Software Engineering and Methodology*, vol. 32, no. 3, pp. 1-46, 2023.

[154] K. Okonkwo, "Top Three Libraries for Web3 Developers," [Online]. Available: https://metamask.io/news/top-three-libraries-for-web3-developers. [Accessed 5 February 2025].

[155] "The industry leading RPC API," [Online]. Available: https://www.alchemy.com/rpc-api. [Accessed 9 March 2025].

[156] "Top Tools for Smart Contract Development : IDE's, Languages, Framework, Testing & Auditing," [Online]. Available: https://www.rapidinnovation.io/post/tools-for-smart-contract-development. [Accessed 9 August 2025].

[157] "Solidity," [Online]. Available: https://docs.soliditylang.org/en/v0.8.30/. [Accessed 20 August 2025].

[158] M. N. Uddin, A. H. M. A. Hasnat, S. Nasrin, M. S. Alam and M. A. Yousuf, "Secure File Sharing System Using Blockchain, IPFS and PKI Technologies," in *International Conference on Electrical Information and Communication Technology*, Khulna, Bangladesh, 2021.

[159] "What is IFPS?," [Online]. Available: https://aioz.network/blog/what-is-ipfs. [Accessed 31 July 2025].

[160] "Decentralized file sharing, explained," [Online]. Available: https://cointelegraph.com/explained/decentralized-file-sharing-explained. [Accessed 31 July 2025].

[161] R. Sidharthan and V. R. Balasaraswathi, "Secured Land Registration using Ethereum Blockchain and IPFS," in *International Conference on Applied Artificial Intelligence and Computing*, Salem, India, 2024.

[162] "Django vs. Flask: Which Is the Best Python Web Framework?," [Online]. Available: https://blog.jetbrains.com/pycharm/2023/11/django-vs-flask-which-is-the-best-python-web-framework/. [Accessed 20 August 2025].

[163] "web3.py," [Online]. Available: https://web3py.readthedocs.io/en/stable/overview.html. [Accessed 10 March 2025].

[164] D. Stevenson, "What is Firebase? The complete story, abridged.," [Online]. Available: https://medium.com/firebase-developers/what-is-firebase-the-complete-story-abridged-bcc730c5f2c0. [Accessed 5 February 2025].

BUPT

[165] D. L. Silaghi, A. C. Artenie and D. E. Popescu, "Optimizing Teacher Portfolio Integrity with a Cost-Effective Smart Contract for School-Issued Teacher Documents," *Computers*, vol. 14, no. 9, p. 395, 2025.

[166] "ChainList - Helping users connect to EVM-powered networks," [Online]. Available: https://chainlist.org/chain/11155111. [Accessed 5 February 2025].

[167] "Faucet-The multi-chain faucet for everyone.," [Online]. Available: https://faucet.triangleplatform.com/polygonzkevm/cardona. [Accessed 13 September 2025].

[168] "Solidity 0.8.30 Release Announcement," [Online]. Available: https://soliditylang.org/blog/2025/05/07/solidity-0.8.30-release-announcement/. [Accessed 11 June 2025].

[169] K. Duan, G. Pang and Y. Lin, "Exploring the current status and future opportunities of blockchain technology adoption and application in supply chain management," *Journal of Digital Economy*, vol. 2, pp. 244-288, 2023.

[170] J. Alongi, "React.js X Next.js - Server-side User Interface Components," [Online]. Available: https://medium.com/swlh/next-x-react-d289f006c9eb. [Accessed 7 February 2025].

[171] "Web3Eth methods," [Online]. Available: https://docs.web3js.org/guides/web3_eth/methods/. [Accessed 3 February 2025].

[172] "Linking and Navigating," [Online]. Available: https://nextjs.org/docs/app/getting-started/linking-and-navigating. [Accessed 4 February 2025].

[173] A. Pandey, J. Rout, A. Soni and S. K. Nanda, "Blockchain based Digital Multimedia Content Authentication System: using IPFS and Ethereum," in *IEEE International Conference on Blockchain and Distributed Systems Security*, Pune, India, 2024.

[174] M. Kolomoyets and Y. Kynash, "Modern approaches to interact Smart-Contracts in React.js development with ThirdWeb framework," in *IEEE 18th International Conference on Computer Science and Information Technologies*, Lviv, Ukraine, 2023.

[175] X. Sun, F. R. Yu, P. Zhang, Z. Sun, W. Xie and X. Peng, "A Survey on Zero-Knowledge Proof in Blockchain," *IEEE Network*, vol. 35, no. 4, pp. 198 - 205, 2021.

[176] H. Eren, Ö. Karaduman and M. T. Gençoğlu, "Security Challenges and Performance Trade-Offs in On-Chain and Off-Chain Blockchain Storage: A Comprehensive Review," *Applied Sciences*, vol. 15, no. 6, p. 3225, 2025.

[177] "Forever Isn't Free: The Cost of Storage on a Blockchain Database," [Online]. Available: https://medium.com/ipdb-blog/forever-isnt-free-the-cost-of-storage-on-a-blockchain-database-59003f63e01. [Accessed 1 August 2025].

[178] A. Jain, C. Jain and K. Krystyniak, "Blockchain transaction fee and Ethereum Merge," *Finance Research Letters*, vol. 58, p. 104507, 2023.

[179] S. Leonardos, B. Monnot, D. Reijsbergen, E. Skoulakis and G. Piliouras, "Dynamical analysis of the EIP-1559 Ethereum fee market," in *Proceedings of the 3rd ACM Conference on Advances in Financial Technologies*, 2021.

[180] "EIP-1559 | How much was my gas fee again…?," [Online]. Available: https://ingeun92.medium.com/eip-1559-how-much-was-my-gas-fee-again-6a8cbbe26477. [Accessed 6 August 2025].

[181] D. Koutmos, "Network Activity and Ethereum Gas Prices," *J. Risk Financial Manag*, vol. 16, no. 10, p. 431, 2023.

BUPT

[182] E. Shao and D. Rajapaksa, "Miner competition and transaction fees," *Journal of Economic Behavior & Organization*, vol. 227, p. 106736, 2024.

[183] J. P. Cruz, Y. Kaji and N. Yanai, "RBAC-SC: Role-Based Access Control Using Smart Contract," *IEEE Access*, vol. 6, pp. 12240 - 12251, 2018.

[184] Y. Ding, J. Jin, J. Zhang, Z. Wu and K. Hu, "SC-RBAC: A Smart Contract based RBAC Model for DApps," in *International Conference on Human Centered Computing*, 2019.

[185] I. Achour, S. Ayed and H. Idoudi, "On the Implementation of Access Control in Ethereum Blockchain," in *International Conference on Innovation and Intelligence for Informatics, Computing, and Technologies (3ICT)*, Zallaq, Bahrain, 2021.

[186] M. Baldauf, E. Sonnleitner and M. Kurz, "Exemplary Ethereum Development Strategies Regarding Security and Gas-Saving," *Electronics*, vol. 13, no. 1, p. 117, 2024.

[187] J. X. Valverde, "Advanced gas optimization tips for Solidity," [Online]. Available: https://coinsbench.com/advanced-gas-optimizations-tips-for-solidity-85c47f413dc5. [Accessed 29 August 2025].

[188] H. Achiando, "40 Tips to Optimize Smart Contract Gas Cost," [Online]. Available: https://www.linkedin.com/pulse/optimizing-smart-contract-gas-cost-harold-achiando. [Accessed 29 August 2025].

[189] [Online]. Available: https://solidityscan.com/. [Accessed 1 June 2025].

[190] "Remix IDE of Ethereum foundation integrates Solidityscan to help developers identify vulnerabilities effortlessly," [Online]. Available: https://chainwire.org/2024/07/23/remix-ide-of-ethereum-foundation-integrates-solidityscan-to-help-developers-identify-vulnerabilities-effortlessly/?utm_source=chatgpt.com&mfk=rdm1zK3wRg%2BQCRY4q0Wsosmz01Tml0KQEVdgn6QM5AYQupzPKqlSXgV1D6IgM%2B55A3xdySElOMAn. [Accessed 27 August 2025].

[191] "Blockscout Integrates SolidityScan to Enhance Security," [Online]. Available: https://solidityscan.com/discover/blockscout-integrates-solidityscan/. [Accessed 1 June 2025].

[192] A. Chetia, "Why the SafeMath Library Is Becoming Obsolete in Solidity Development," [Online]. Available: https://coinsbench.com/why-the-safemath-library-is-becoming-obsolete-in-solidity-development-1717928d2970. [Accessed 30 August 2025].

[193] "Bitnodes," [Online]. Available: https://bitnodes.io/. [Accessed 13 April 2025].

[194] "Bitcoin," [Online]. Available: https://www.blockchain.com/explorer/assets/btc. [Accessed 6 June 2025].

[195] A. T. Elizabeth Durant, "Digital Diploma debuts at MIT," [Online]. Available: https://news.mit.edu/2017/mit-debuts-secure-digital-diploma-using-bitcoin-blockchain-technology-1017. [Accessed 26 March 2025].

[196] M. Cocosatu, "Diploma Supplement - Instrument of the Quality Assurance Process in the Romanian Higher Education?," *European Integration Realities and Perspectives*, pp. 868-872, 2012.

[197] E. Zhu, J. Zhang, J. Yan, K. Chen and C. Gao, "N-gram MalGAN: Evading machine learning detection via feature n-gram," *Digital Communications and Networks*, vol. 8, no. 4, pp. 485-491, 2022.

[198] "Mastering LightGBM: An In-Depth Guide to Efficient Gradient Boosting," [Online]. Available: https://medium.com/@mohtasim.hossain2000/mastering-lightgbm-an-in-depth-guide-to-efficient-gradient-boosting-8bfeff15ee17. [Accessed 9 August 2025].

BUPT

[199] "Sentiment Analysis of Customer Reviews: A Guide for Beginners," [Online]. Available: https://medium.com/@14duttatanushreeya/sentiment-analysis-of-customer-reviews-a-guide-for-beginners-4a5f2194a121. [Accessed 9 August 2025].

[200] "Avalanche Bridge: Revolutionizing Cross-Chain Transfers," [Online]. Available: https://medium.com/avalancheavax/avalanche-bridge-revolutionizing-cross-chain-transfers-59e327780b93. [Accessed 10 August 2025].

[201] "Oasis Network," [Online]. Available: https://docs.oasis.io/general/oasis-network/. [Accessed 10 August 2025].

[202] O. Beigel, "Bitcoin Wallet Guide, Reviews, and Comparison," [Online]. Available: https://99bitcoins.com/bitcoin-wallet/. [Accessed 10 March 2025].

[203] F. Ren, B. Zhao, J. Wang, J.-X. Zhou and T.-Y. Xie, "Enhancing Blended Learning Evaluation Through a Blockchain and Searchable Encryption Approach," *Electronics,* vol. 14, no. 5, p. 1039, 2025.

[204] [Online]. Available: https://coinmarketcap.com/currencies/ethereum/. [Accessed 7 August 2025].

BUPT