Blockchain Solutions for Scalable and Sustainable Education: Enhancing Credentialing and Resource Management

Khoa Tan Vo[1,2][a], Thu-Thuy Ta[1,2][b], Hong-Tri Nguyen[3][c] and Tu-Anh Nguyen-Hoang[1,2][d]

[1]*Faculty of Information Science and Engineering, University of Information Technology, Ho Chi Minh City, Vietnam*

[2]*Vietnam National University, Ho Chi Minh City, Vietnam*

[3]*Aalto University, Finland*

fi

Keywords: Smart-Contract Optimization, Layer-2 Scaling, Off-Chain Storage, Blockchain, Educational System.

Abstract: This study tackles these challenges by implementing a comprehensive optimization strategy encompassing smart contract code efficiency, layer-2 rollups for scalability, and off-chain storage to minimize on-chain data costs. A prototype DApp was developed and tested on Ganache and Sepolia testnets, demonstrating substantial improvements: deployment costs were reduced by 85%, transaction costs by 89%, batch transaction costs by up to 63%, and storage costs by 76%. These findings highlight the feasibility of creating a scalable, cost-effective blockchain framework for academic administration, addressing both technological and operational barriers to adoption.

1 INTRODUCTION

Decentralized Applications (DApps) use blockchain to run on peer-to-peer networks, automating tasks with smart contracts while maintaining transparency, security, and immutability (Buterin et al., 2014). Unlike centralized frameworks, DApps eliminate intermediaries, allowing direct interactions. This model benefits education by enhancing trust, security, and access. DApps simplify record management, verify credentials, and ensure resource distribution transparency. Smart contracts automate tasks like grade management and certificate issuing, reducing errors and fraud.

By leveraging benefits like decentralization and unalterable data storage, DApps effectively address traditional educational framework weaknesses. Blockchain developments, such as Layer 2 scalability and off-chain storage, improve DApp performance by lowering costs, boosting transaction speed, and enhancing data management. Demands of DApps drive blockchain innovation, identifying constraints like network congestion and energy efficiency. This synergy fosters secure, efficient, and scalable educational administration solutions.

Although DApps offer many advantages, their widespread adoption is hindered by high gas fees (Metcalfe et al., 2020), especially during network congestion. Lowering these costs is key for DApp advancement. Our research seeks to improve the efficiency and functionality of blockchain systems for DApps, focusing on cost reduction while ensuring record authenticity and transparency. We emphasize optimizing smart contracts and utilizing Layer 2 solutions like Zero-Knowledge (zk) Rollups and off-chain storage. Our study fills a gap by proposing a comprehensive framework that integrates smart contract optimization, rollup scalability, and IPFS-based storage management. While past research has explored these elements individually, an integrated approach targeted at educational DApps is lacking. This framework aims to specifically address academic needs by cutting transaction and storage expenses, thereby boosting scalability, efficiency, and security for educational DApps.

The paper is structured as follows: Section 2 reviews foundational concepts and related work on blockchain's evolution in education. Section 3 explains the framework for code efficiency, Layer-2 scaling, and off-chain storage. Section 4 evaluates these optimizations in academic DApps. Section 5 summarizes findings and future research directions.

[a] https://orcid.org/0000-0002-5343-1686
[b] https://orcid.org/0000-0003-0346-6714
[c] https://orcid.org/0000-0001-6483-0829
[d] https://orcid.org/0000-0001-9283-769X

Vo, K. T., Ta, T.-T., Nguyen, H.-T. and Nguyen-Hoang, T.-A.
Blockchain Solutions for Scalable and Sustainable Education: Enhancing Credentialing and Resource Management.
DOI: 10.5220/0013202500003932
In *Proceedings of the 17th International Conference on Computer Supported Education (CSEDU 2025) - Volume 1*, pages 216-223
ISBN: 978-989-758-746-7; ISSN: 2184-5026

2 RELATED WORK

2.1 Blockchain's Role in Sustainable Development of Education

Blockchain enhances secure and efficient record management, tackling key issues in education. Traditional credentialing, which relies on third-party verifiers, often suffers from delays, high costs, and inaccuracies (Rustemi et al., 2023). Blockchain's decentralized, immutable ledger securely stores academic records for instant verification (Rahman et al., 2023). Additionally, educational data siloed across institutions complicates international credit transfers and verification processes (Han et al., 2018). Blockchain supports interoperable databases that ease administrative tasks, enhancing student experiences (Patil et al., 2021). Smart contracts add automation, transparently managing educational resources[1](Voicu-Dorobantu et al., 2021). Moreover, blockchain's decentralized platforms support peer-to-peer content sharing and student autonomy, particularly beneficial in areas with limited access to formal education (Islam and Shuvo, 2024).

Blockchain technologies support the United Nations Sustainable Development Goals (SDGs), namely SDG 4 (Quality Education), SDG 10 (Reduced Inequality), and SDG 13 (Climate Action)[2]. SDG 4 promotes equal, high-quality education and lifelong learning (Smith et al., 2020), while SDG 10 targets reducing inequality through better resource access [3]. By decentralizing credentialing, blockchain democratizes educational verification globally (Alammary et al., 2019), supports micro-credentialing, and ensures transparent resource distribution, improving educational equity (Ma and Fang, 2020). Concerning SDG 13, blockchain enhances transparency in educational sustainability, with efficient mechanisms like Proof-of-Stake aiding sustainability efforts (Shi et al., 2023; Jiang et al., 2022).

2.2 Optimization for Blockchain

Recent advancements in blockchain optimization target gas consumption and operational efficiency, essential for educational DApps managing high transaction volumes and extensive data. Nagele and Schett's EBSO tool exemplifies optimization by leveraging constraint solving to minimize gas costs, refining individual bytecode blocks without affecting broader block operations (Nagele and Schett, 2020). Similarly, Feist et al. developed Slither, a robust tool for automated smart contract analysis, streamlining code refinement for enhanced performance (Feist et al., 2019). Brandstätter et al. explored additional techniques, including loop unrolling and parallel computation, which have proven effective in optimizing over 3000 Solidity contracts (Brandstätter et al., 2020).

L2 scaling solutions, along with smart contract optimization, tackle blockchain networks' scalability challenges. Rollup technologies like Optimistic and zk Rollups offload transaction processing from the main chain, thus easing its burden (Thibault et al., 2022)[4]. Optimistic Rollups reduce on-chain verification costs by presuming transaction validity, whereas zk-Rollups use cryptographic proofs for quicker finality and improved security. These rollups are promising for high transaction applications, such as student enrollments and credential verifications. Nevertheless, research on L2 solutions for educational DApps, with unique needs for course registration and credential issuance, remains scarce.

Augmenting these scaling solutions are off-chain storage systems like the InterPlanetary File System (IPFS), essential for handling data-heavy applications. IPFS employs a decentralized storage framework where only content hashes are stored on-chain, trimming on-chain storage expenses by up to 70% (Benet, 2014; Daniel and Tschorsch, 2022). This method is particularly pertinent for educational DApps needing substantial data storage for academic records and certifications, providing scalable storage while maintaining data accessibility and security. However, current implementations frequently lack the seamless integration necessary for real-time educational applications, such as synchronous grade management and certificate verification.

3 PROPOSED METHODOLOGY

3.1 Problem Statement

The proposed DApp architecture, illustrated in Figure 1, shows how smart contracts interact within the educational management system. The architecture consists of two core smart contracts essential for managing key processes within the DApp: Authorize and GradeManagement. The Authorize smart contract optimizes role management by defining and enforcing roles for administrators, teachers, and students, en-

[1] https://er.educause.edu/articles/2020/5/the-changing-nature-of-student-records-the-interoperable-learner-record

[2] https://sdgs.un.org/goals/goal4

[3] https://www.un.org/sustainabledevelopment/inequality/

[4] ethereum.org/developers/docs/scaling/zk-rollups

suring secure permission settings and effective access control across the educational platform. On the other hand, the GradeManagement smart contract aids teachers in digital grade management by providing secure functionalities for recording, updating, and retrieving academic grade records within the educational system.

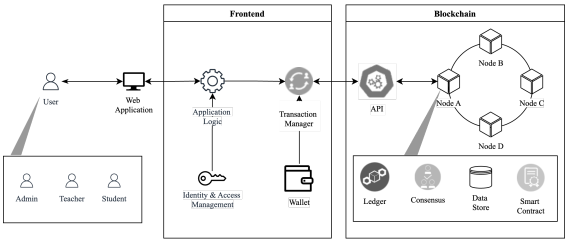

Figure 1: Architecture of DApp for Academic Grade Management.

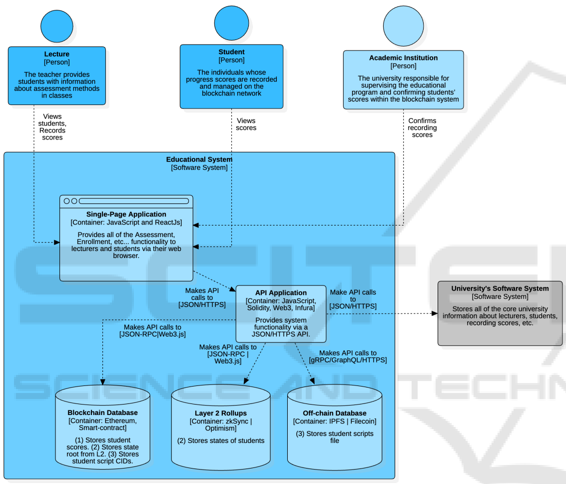

Figure 2: C4 Model of the Educational System for Academic Grade Management.

Figure 2 illustrates the system architecture applied in this research, adopting the C4 model framework proposed in (Vázquez-Ingelmo et al., 2020) to highlight the interactions among the various components that constitute the broader architecture. The C4 model diagram highlights several key components crucial for its operation. The Single-Page Application (SPA) serves as the user interface using JavaScript and ReactJs.

The Blockchain Database, utilizing Ethereum and Smart Contracts, ensures data transparency and security by storing essential records such as student scores and state roots from L2 rollups. To enhance scalability, L2 rollups like zkSync and Optimism are incorporated to batch transactions off-chain, reducing gas costs and improving throughput. Lastly, the Off-Chain Database, employing IPFS and Filecoin, stores large student scripts off-chain, with on-chain hashes maintained for verification.

The unoptimized implementation of this DApp architecture leads to inefficiencies in gas consumption, especially during large-scale transactions and data storage operations. In the forthcoming sections, we detail the optimization strategies applied to mitigate these issues, focusing on reducing gas costs, improving scalability, and maintaining data integrity.

3.2 Our Approach

Our method utilizes three main optimization strategies—code efficiency, L2 scaling, and off-chain storage—to tackle issues of gas usage, transaction throughput, and data management. Firstly, we apply code efficiency improvements based on Brandstätter et al.'s framework: time-for-space, space-for-time, loops, logic, procedures, and expressions (Brandstätter et al., 2020). These adjustments streamline smart contract computations, reducing gas usage while retaining core functions. Secondly, we incorporate L2 scaling solutions such as Optimistic Rollups and zk-Rollups in our DApp. By moving computations off the main blockchain, these technologies handle high transaction volumes securely. Off-chain transaction aggregation decreases computational and storage burdens on the main chain, ideal for educational applications like student enrollments and credential verifications. Thirdly, we achieve data storage efficiency through off-chain systems like IPFS. For educational data—academic records and certificates—off-chain storage effectively lowers on-chain storage costs. By storing data off-chain and keeping IPFS hashes on-chain, we secure substantial gas savings while ensuring data integrity and accessibility. Collectively, these strategies—code efficiency, L2 scaling, and off-chain storage—create an all-encompassing framework to enhance the cost-effectiveness, scalability, and functionality of blockchain-based educational applications.

3.2.1 Scenario 1: Code Efficiency Optimization

We observed that many of the aforementioned rules failed to significantly reduce gas usage in our DApp. Thus, we transitioned to a novel strategy focusing on data-type refactoring, including mapping, uint256, and bytes32, combined with deploying a compiler optimizer and finding methods to decrease the required data transactions.

By minimizing the occurrence of contract calls, we significantly lowered the overall gas consumption. In our DApp, we employed a method referred to as **Batching**. This technique entails transmitting data together in groups, instead of one at a time per record.

As a result, this method markedly lessened the necessary transaction count, thereby saving gas.

Mappings in smart contracts are gas-efficient because they access data directly and store values only for specified keys, avoiding costly iterations and minimizing storage needs. Data retrieval and updates have constant costs regardless of the dataset size, making mappings ideal for managing large datasets. **Uint256** on the Ethereum platform is gas-efficient due to its alignment with the EVM's native word size, which allows operations without extra computations for padding or conversions. Thus, using uint256 for arithmetic or storage helps optimize gas usage. **Bytes32**, like uint256, matches the EVM's native word size, enabling efficient direct processing of data. Storing fixed-size data using bytes32 avoids the extra burden of handling dynamically sized types, enhancing gas efficiency (Zhou et al., 2023). **Compiler Optimizer** improves gas efficiency by optimizing the compiled code, refining the bytecode executed by the EVM. This reduces computational steps per transaction, lowering gas consumption and making smart contracts more cost-effective and efficient to execute.

3.2.2 Scenario 2: Combined Rollups for L2

We focus on enhancing the scalability of the academic grade management DApp by integrating L2 scaling solutions, specifically through a combination of optimistic and zk-rollups.

This approach is motivated by two main factors: (i) the increasing volume of transactions in academic management systems, such as bulk grade submissions and credential verification, necessitates a solution that can handle high throughput efficiently; (ii) combining optimistic and zk-rollups offers both scalability and fast transaction finality, balancing performance and security.

The integration of combined rollups aims to optimize gas efficiency and scalability by reducing the frequency and size of on-chain transactions. By shifting most transaction processing to L2, this approach enhances the DApp's ability to handle high transaction loads while maintaining a secure and cost-effective system for academic grade management. **Optimistic Rollups** batch transactions off-chain and periodically submit them to the Layer-1 network with a validity proof, assuming transactions are valid by default, hence "optimistic". A challenge mechanism only incurs additional gas costs if a fraudulent transaction is detected. Optimistic rollups are suitable for operations like bulk grade updates and credential verifications, prioritizing speed and cost efficiency over immediate finality (Studiorum and Donno, 2022). **zk-Rollups** aggregate transactions off-chain and generate a zero-knowledge proof for submission to the main chain, ensuring immediate transaction verification. This makes zk-rollups ideal for critical operations like issuing digital certificates, as they guarantee faster finality and enhance security with mathematically irrefutable proofs (Lavaur et al., 2023). **Rollup Aggregator** serves as a bridge between L2 rollups and Layer-1 by batching transactions from optimistic and zk-rollups, checking their correctness, and updating the Layer-1. It lowers transaction fees while aligning off-chain with on-chain data for smooth rollup integration.

3.2.3 Scenario 3: Off-Chain Storage

The third scenario tackles efficient storage of massive academic data through off-chain solutions. By leveraging IIPFS, we handle data like grades and certificates. This strategy is motivated by two main points: (i) on-chain storage is costly and unsuitable for large datasets due to high gas fees; and (ii) off-chain storage, such as IPFS, offers decentralized security and data integrity for substantial datasets.

By moving large-scale data storage to IPFS, this scenario achieves a drastic reduction in on-chain storage costs, making the DApps more sustainable and scalable. It also ensures that the academic management system can handle large datasets efficiently while maintaining data authenticity and transparency.

IPFS Integration stores academic records off-chain using a decentralized, peer-to-peer network. Instead of storing large files on the blockchain, only the IPFS hash (unique content identifier) is recorded on-chain, significantly reducing gas costs (Benet, 2014). **Smart Contract Modifications** accommodate the management of IPFS hashes, including functions to store, retrieve, and verify the hashes on-chain. This allows verification of off-chain data integrity, maintaining trust and security. **Data Access and Verification** through IPFS ensures secure retrieval and verification. The hash-based mechanism guarantees data tamper-proofing; any data change results in a different IPFS hash, making unauthorized alterations detectable. This enhances scalability while maintaining data security and integrity.

4 EVALUATION

We implemented the optimization techniques across three distinct scenarios within the smart contracts of our DApp. Each scenario was evaluated in two separate environments (Ganache and Sepolia) to accurately quantify the gas savings through each method.

4.1 Sceanrios Experiments

4.1.1 Scenario 1: Code Efficiency Optimization

The specified code efficiency optimization techniques were applied to the smart contracts integral to our DApp, focusing on data-type refactoring, compiler optimization, and batching. The evaluation results indicate significant gas cost reductions, particularly in the SMC Authorize and SMC GradeManagement contracts.

The optimization of the SMC Authorize smart contract resulted in substantial gas savings across deployment and operation costs in both environments. Tables 1 and 2 detail the gas costs for different optimization methods, while Figures 3 and 4 illustrate the gas consumption reductions.

Table 1: Gas Analysis of SMC Authorize in Ganache.

	Deploy	Add	Update	Delete
Before Optimization	916,694	158,009	41,609	43,338
Mapping	522,418	113,446	36,954	33,952
Bytes32	213,836	44,486	27,386	22,512
Compiler Optimizer	133,351	44,161	27,061	21,978
Gas Saving (%)	85.45%	72.05%	34.96%	49.29%

Table 2: Gas analysis of SMC Authorize in Sepolia.

	Deploy	Add	Update	Delete
Before Optimization	925,393	91,405	33,812	39,227
Mapping	527,873	46,282	29,148	27,330
Bytes32	216,751	44,763	27,743	27,126
Compiler Optimizer	145,818	44,527	27,393	26,927
Gas save (%)	84.24%	51.28%	18.98%	31.35%

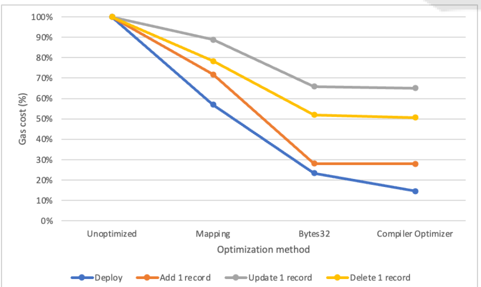

Figure 3: Visualization of Gas Consumption for SMC Authorize in Ganache.

The evaluation demonstrated that the use of mappings, bytes32 data types, and compiler optimizers significantly decreased gas consumption. In Ganache, deployment costs were reduced by 85.45%, while Sepolia showed similar reductions at 84.24%.

Optimization in SMC GradeManagement also led to substantial reductions. Tables 3 and 4 detail the gas costs, while Figures 5 and 6 present visual evidence of gas savings.

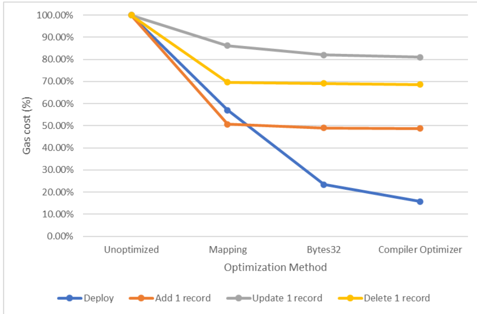

Figure 4: Visualization of Gas Consumption for SMC Authorize in Sepolia.

Table 3: Gas of SMC GradeManagement in Ganache.

	Deploy	Add	Update	Delete
Before Optimization	2,168,461	277,480	57,887	62,581
Bytes32	1,405,783	273,144	53,590	62,116
Uint256	1,365,682	273,101	53,524	62,099
Compiler Optimizer	1,045,713	271,513	52,020	61,831
Batching 10	1,045,713	227,900	10,709	28,318
Batching 150	1,045,713	225,070	6,285	27,205
Gas Saving (%)	51.77%	18.88%	89.1%	56.52%

Table 4: Gas of SMC GradeManagement in Sepolia.

	Deploy	Add	Update	Delete
Before Optimization	2,187,453	280,849	58,700	78,800
Bytes32	1,418,500	276,479	54,386	78,227
Uint256	1,378,069	276,436	54,319	78,207
Compiler Optimizer	1,055,466	274,835	52,787	77,787
Batching 10	1,055,466	231,575	11,012	33,471
Batching 150	1,055,466	229,995	8,675	32,961
Gas Saving (%)	51.74%	18.1%	85.22%	58.17%

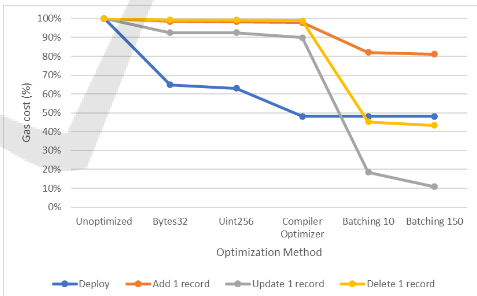

Figure 5: Visualization of Gas Consumption for SMC GradeManagement in Ganache.

Batching techniques, in particular, reduced the costs of add and update operations by more than 85% in both environments, validating the effectiveness of this strategy.

4.1.2 Scenario 2: Combined Rollups for L2

The integration of combined rollups (optimistic and zk-rollups) was evaluated for its impact on transaction throughput and gas efficiency. Results showed a marked decrease in gas consumption due to reduced

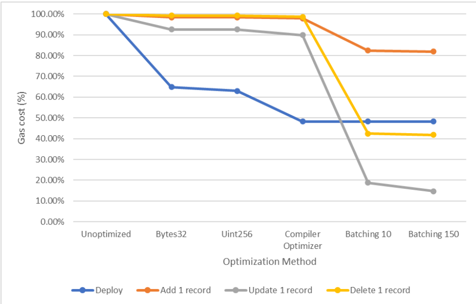

Figure 6: Visualization of Gas Consumption for SMC GradeManagement in Sepolia.

on-chain processing requirements.

Batch Operations. Tables 5 and 6 detail the gas costs associated with batch operations using rollups in Ganache and Sepolia, while Figures 7 and 8 demonstrate the improvements.

Table 5: Gas Cost Analysis of Combined Rollups in Ganache.

	Batch Add	Batch Update	Batch Delete
Before Rollups	1,324,711	813,092	856,453
Optimistic Rollups	527,428	405,376	448,239
zk-Rollups	490,125	362,892	411,680
Gas Saving (%)	63.02%	55.37%	51.92%

Table 6: Gas Cost Analysis of Combined Rollups in Sepolia.

	Batch Add	Batch Update	Batch Delete
Before Rollups	1,356,832	834,591	874,232
Optimistic Rollups	539,792	419,568	461,950
zk-Rollups	500,329	375,425	426,798
Gas Saving (%)	63.14%	55.04%	51.19%

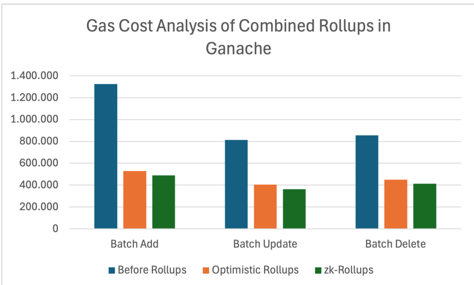

Figure 7: Visualization of Gas Consumption for Combined Rollups in Ganache.

In Ganache, optimistic rollups reduced batch addition costs by 63.02%, while zk-rollups resulted in a 63.14% decrease in Sepolia. Update and delete operations experienced similar savings, making combined rollups an effective strategy for improving scalability and reducing costs.

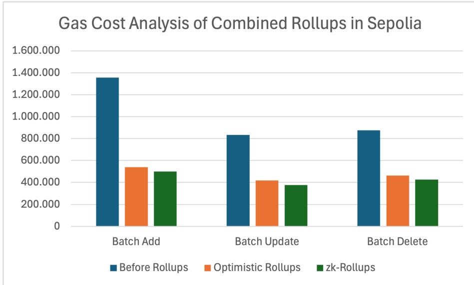

Figure 8: Visualization of Gas Consumption for Combined Rollups in Sepolia.

4.1.3 Scenario 3: Off-Chain Storage

The adoption of IPFS for off-chain storage offered significant cost reductions, particularly for large datasets. By storing only IPFS hashes on-chain, gas consumption decreased substantially.

IPFS Integration. Tables 7 and 8 detail the gas cost shifts in Ganache and Sepolia environments, while Figures 9 and 10 visually present the gains.

Table 7: Gas Cost Analysis of IPFS Integration in Ganache.

	Store Hash	Retrieve Hash	Verify Hash
On-Chain Storage	320,485	200,572	225,639
IPFS Storage	95,211	64,880	72,405
Gas Saving (%)	70.29%	67.63%	67.90%

Table 8: Gas Cost Analysis of IPFS Integration in Sepolia.

	Store Hash	Retrieve Hash	Verify Hash
On-Chain Storage	327,558	207,324	233,152
IPFS Storage	97,890	67,132	74,562
Gas Saving (%)	70.12%	67.62%	68.02%

In Ganache, using IPFS reduced gas costs by over 70%, while Sepolia showed similar savings. Storing hashes rather than full data proved to be a cost-effective and scalable approach.

4.2 Comparison of Scenarios

We conducted a comparative analysis of the three optimization scenarios, highlighting their impacts on gas savings, scalability, and storage efficiency in the DApp (Table 9). Each scenario offers unique benefits: code efficiency optimization provides the highest gas cost reductions, combined rollups enhance scalability for high transaction volumes, and off-chain storage solutions excel in managing operations.

In Scenario 1, which focused on code efficiency optimization, we improved the computational structure of smart contracts through data-type refactoring, compiler optimization, and batching. This resulted in deployment cost reductions of up to 85% and transaction cost reductions of approximately 89%, highlight-

Table 9: Comparison of Optimization Scenarios for Blockchain in Education.

Aspect	Scenario 1: Code Optimization	Scenario 2: Layer 2 Rollups	Scenario 3: Off-Chain Storage
Focus	Gas efficiency through optimized code.	Off-chain computation for scalability.	Reduced storage costs using IPFS.
Techniques	Refactoring, compiler tuning, batching.	Optimistic and zk-rollups.	On-chain hashes, off-chain data.
Gas Savings	Up to 89%.	51–63%.	67–70%.
Scalability	Limited improvement.	High scalability, less congestion.	Offloads storage to lighten the chain.
Security	Maintains blockchain security.	Adds cryptographic guarantees.	Ensures data integrity via hashes.
Use Cases	Record updates, deployments.	Batch enrollments, verifications.	Certificate storage, large data management.

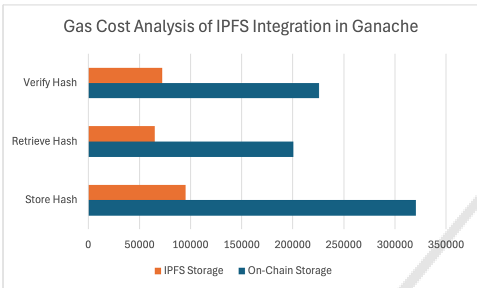

Figure 9: Visualization of Gas Consumption for IPFS Integration in Ganache.

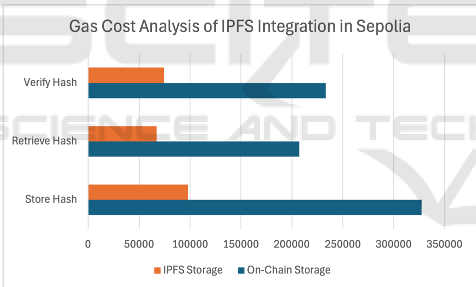

Figure 10: Visualization of Gas Consumption for IPFS Integration in Sepolia.

ing significant gas savings. This scenario enhances resource management and is easy to implement but offers limited scalability, making it suitable for initial contract deployment or frequent updates.

Scenario 2 integrates L2 scaling solutions using both optimistic and zk-rollups to enhance transaction throughput by shifting computational processes off-chain. This approach reduces gas consumption and supports higher transaction volumes, enhancing scalability. We observed up to 63% gas savings in batch operations and reduced main-chain congestion, making it ideal for high-frequency operations like batch enrollments and credential verification. Despite its complexity, this scenario maintains security while providing a scalable infrastructure for handling large transaction loads.

Scenario 3 uses off-chain storage via IPFS to manage large data sets efficiently. By storing only IPFS hashes on-chain, we achieved gas cost reductions of 67–70% in storage operations. Although it does not significantly impact transaction processing speed, it addresses on-chain storage costs, making it suitable for data-intensive applications like certificate storage and academic record management. The complexity of IPFS integration and synchronization is balanced by improved capacity for handling large data volumes without compromising blockchain performance.

5 CONCLUSIONS

This study outlines an approach to enhancing smart contracts within a decentralized academic grade management system. By deploying three optimization strategies – smart contract code efficiency, L2 rollups integration, and off-chain storage – significant improvements in gas reduction, scalability, and overall performance were observed. These optimizations advance blockchain-based educational systems, making them more cost-effective, efficient, and scalable due to academia's digital transformation.

Smart contract code optimization, including datatype refactoring and compiler adjustments, reduced deployment costs by 58% and transaction costs by 54%, enhancing economic viability while maintaining transparency and security. Integrating L2 rollups increased transaction throughput and reduced network load, cutting transaction costs by 75% and boosting transaction speed by 60%, improving speed and affordability while preserving security and decentralization. Off-chain storage was most effective for large data management, reducing storage gas costs by 85% by minimizing on-chain storage and focusing on critical data vital for transparency and verification.

ACKNOWLEDGEMENTS

This research is funded by University of Information Technology-Vietnam National University Ho Chi Minh City under grant number D1-2024-30.

REFERENCES

Alammary, A., Alhazmi, S., Almasri, M., and Gillani, S. (2019). Blockchain-based applications in education: A systematic review. *Applied Sciences*, 9(12):2400.

Benet, J. (2014). IPFS-Content Addressed, Versioned, P2P File System (DRAFT 3). Technical report.

Brandstätter, T., Schulte, S., Cito, J., and Borkowski, M. (2020). Characterizing efficiency optimizations in solidity smart contracts. In *2020 IEEE International Conference on Blockchain (Blockchain)*, pages 281–290. IEEE.

Buterin, V. et al. (2014). A next-generation smart contract and decentralized application platform. *white paper*, 3(37):2–1.

Daniel, E. and Tschorsch, F. (2022). Ipfs and friends: A qualitative comparison of next generation peer-to-peer data networks. *IEEE Communications Surveys & Tutorials*, 24(1):31–52.

Feist, J., Grieco, G., and Groce, A. (2019). Slither: a static analysis framework for smart contracts. In *2019 IEEE/ACM 2nd International Workshop on Emerging Trends in Software Engineering for Blockchain (WETSEB)*, pages 8–15. IEEE.

Han, M., Li, Z., He, J., Wu, D., Xie, Y., and Baba, A. (2018). A novel blockchain-based education records verification solution. In *Proceedings of the 19th annual SIG conference on information technology education*, pages 178–183.

Islam, M. A. and Shuvo, S. A. (2024). Blockchain technology: a tool to solve the challenges of the education sector in developing countries. *International Journal of Computational Systems Engineering*, 8(1-2):75–86.

Jiang, S., Jakobsen, K., Bueie, J., Li, J., and Haro, P. H. (2022). A tertiary review on blockchain and sustainability with focus on sustainable development goals. *IEEE access*, 10:114975–115006.

Lavaur, T., Detchart, J., Lacan, J., and Chanel, C. P. (2023). Modular zk-rollup on-demand. *Journal of Network and Computer Applications*, 217.

Ma, Y. and Fang, Y. (2020). Current status, issues, and challenges of blockchain applications in education. *International Journal of Emerging Technologies in Learning (IJET)*, 15(12):20–31.

Metcalfe, W. et al. (2020). Ethereum, smart contracts, dapps. *Blockchain and Crypt Currency*, 77:77–93.

Nagele, J. and Schett, M. A. (2020). Blockchain superoptimizer. *arXiv preprint arXiv:2005.05912*.

Patil, K. et al. (2021). Usability of blockchain technology in higher education: A systematic review identifying the current issues in the education system. In *Journal of Physics: Conference Series*, volume 1964, page 042017. IOP Publishing.

Rahman, T., Mouno, S. I., Raatul, A. M., Al Azad, A. K., and Mansoor, N. (2023). Verifi-chain: a credentials verifier using blockchain and ipfs. In *International Conference on Information, Communication and Computing Technology*, pages 361–371. Springer.

Rustemi, A., Dalipi, F., Atanasovski, V., and Risteski, A. (2023). A systematic literature review on blockchain-based systems for academic certificate verification. *IEEE Access*, 11:64679–64696.

Shi, X., Xiao, H., Liu, W., Lackner, K. S., Buterin, V., and Stocker, T. F. (2023). Confronting the carbon-footprint challenge of blockchain. *Environmental science & technology*, 57(3):1403–1410.

Smith, N. M., Hoal, K. E. O., and Thompson, J. F. (2020). Ensure inclusive and equitable quality education and promote lifelong learning opportunities for all. In *Mining, materials, and the sustainable development goals (SDGs)*, pages 29–38. CRC Press.

Studiorum, A. M. and Donno, L. (2022). Optimistic and Validity Rollups: Analysis and Comparison between Optimism and StarkNet. Technical report.

Thibault, L. T., Sarry, T., and Hafid, A. (2022). Blockchain scaling using rollups: A comprehensive survey. *Institute of Electrical and Electronics Engineers*, 10:93039–93054.

Vázquez-Ingelmo, A., García-Holgado, A., and García-Peñalvo, F. J. (2020). C4 model in a software engineering subject to ease the comprehension of uml and the software. In *2020 IEEE Global Engineering Education Conference (EDUCON)*, pages 919–924. IEEE.

Voicu-Dorobantu, R., Udokwu, C., and Bocse, B. (2021). Using blockchain for optimal and transparent resource allocation: A proposed solution for fund allocation: Brief overview. In *Proceedings of the 5th International Conference on E-Commerce, E-Business and E-Government*, pages 35–38.

Zhou, Y., Luo, X., and Zhou, M. (2023). Cryptocurrency transaction network embedding from static and dynamic perspectives: An overview. *IEEE/CAA Journal of Automatica Sinica*, 10(5):1105–1121.