

Blockchain and Smart Contract for Digital Certificate Verification

Dr. S.RamaDevi
Dept. of ECE
Andhra University college of
Engineering for women
Maddilapalem,
Visakhapatnam, AP, India

Killada Veera Venkata Padmavathi Sirisha
Dept. of ECE
Andhra University college of
Engineering for women
Maddilapalem,
Visakhapatnam, AP, India

Murupaka Meghana
Dept. of ECE
Andhra University college of
Engineering for women
Maddilapalem,
Visakhapatnam, AP, India

Jakkula Harshitha
Dept. of ECE
Andhra University college of Engineering for women
Maddilapalem, Visakhapatnam, AP, India

Pillala Sriharika
Dept. of ECE
Andhra University college of Engineering for women
Maddilapalem, Visakhapatnam, AP, India

Abstract - The increasing incidents of forged academic certificates and the delays involved in manual verification have created a strong demand for secure digital credential authentication systems. This paper presents the implementation of a Blockchain and Smart Contract based Digital Certificate Verification System designed to provide tamper-proof certificate storage and instant authenticity validation. The developed system integrates student registration, academic record entry, cryptographic hash generation, blockchain transaction storage, QR code generation, and employer-side certificate verification into a unified app platform.

In the implemented model, each student certificate is converted into a unique hash value using cryptographic algorithms, and the generated hash is securely stored in an Ethereum blockchain through Solidity smart contract execution. A QR code is issued for every student, enabling employers or authorized institutions to verify the academic certificate by scanning the QR code. During verification, the system compares the regenerated certificate hash with the immutable blockchain record to identify whether the certificate is genuine or tampered.

Experimental testing confirms that the system performs secure certificate issuance, rapid verification, and effective tamper detection with minimal manual intervention. The developed platform significantly reduces certificate fraud, improves verification speed, and increases trust in digital academic credential management.

Keywords - *Keywords: Blockchain, Smart Contract, Digital Certificate Verification, Cryptographic Hashing, QR Code Authentication, Ganache, Tamper-Proof Verification.*

I. INTRODUCTION

In the digital era, academic certificates and educational credentials play a significant role in higher education admissions, employment recruitment, professional accreditation, and skill validation. Educational institutions issue a large number of certificates every year, making secure certificate management and authenticity verification increasingly important. Traditional certificate verification systems generally depend on centralized databases, paper-based records, and manual communication between employers and educational institutions. These approaches can introduce

delays, increase administrative overhead, and create opportunities for certificate forgery and unauthorized modification. Consequently, there is a growing need for secure, transparent, and efficient mechanisms for issuing, storing, and verifying digital academic credentials. Blockchain technology has emerged as a promising approach to addressing these challenges by providing decentralized and tamper-evident record management [1].

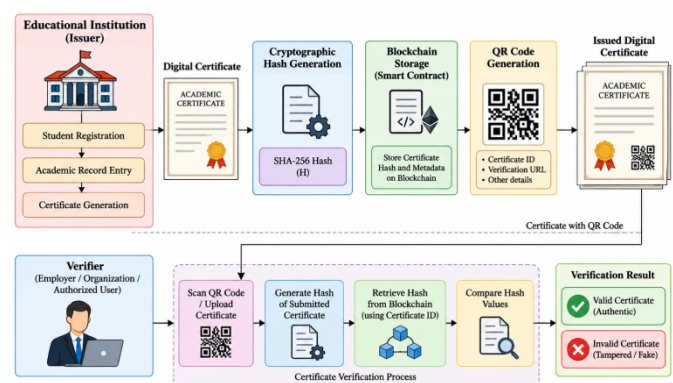


Fig. 1. Illustrative Blockchain Certificate Verification Flowchart

Blockchain is a distributed ledger technology that maintains records across a network of participating nodes. Its fundamental characteristics include decentralization, cryptographic integrity, transparency, and resistance to unauthorized modification. These properties make blockchain suitable for applications that require reliable record verification without depending entirely on a centralized authority. In academic certificate management, blockchain can be used to maintain verifiable records of issued credentials and support independent authentication by employers and other authorized parties. A systematic investigation of blockchain-based academic certificate verification systems highlights the potential of this technology to improve trust, integrity, and efficiency in educational credential management [1].

Smart contracts extend blockchain functionality by enabling predefined operations to be executed automatically

when specified conditions are satisfied. The Ethereum blockchain supports smart contracts developed using Solidity, allowing certificate registration and verification operations to be implemented programmatically. A blockchain-based certificate management system can use smart contracts to register certificate identifiers, associate them with issuing institutions, and verify whether submitted credentials correspond to previously registered records. Such automation can reduce manual intervention and improve the consistency of verification procedures. Ethereum-based decentralized certificate verification approaches have explored the use of smart contracts to establish reliable and transparent mechanisms for academic credential authentication [2].

Although blockchain provides tamper-evident storage, storing complete academic certificates directly on the blockchain may introduce challenges related to storage costs, scalability, and the confidentiality of personal information. To address these limitations, decentralized storage technologies such as the InterPlanetary File System (IPFS) can be integrated with blockchain platforms. In this architecture, the certificate document can be maintained in decentralized storage while its cryptographic hash or content identifier is recorded on the blockchain. Smart contracts can then facilitate the verification of the associated record. Research on integrating IPFS, blockchain, and smart contracts has investigated this approach as a means of reducing certificate fraud and improving the management of academic credentials [3].

Cryptographic hashing is an essential component of digital certificate authentication. A cryptographic hash function converts digital information into a fixed-length output that serves as a fingerprint of the input data. When a certificate is registered, its hash can be calculated and stored on the blockchain. During verification, the system calculates the hash of the submitted certificate and compares it with the previously registered value. If the values match, the submitted document is consistent with the registered digital record. If they differ, the system identifies a mismatch that may indicate modification, corruption, or the submission of an unregistered document. Blockchain-based approaches to digital certificate forgery verification have investigated the use of cryptographic mechanisms and smart contracts to strengthen certificate authentication and detect unauthorized changes [4].

In addition to secure storage and cryptographic verification, accessibility is an important consideration in practical certificate authentication systems. Quick Response (QR) codes can simplify the verification process by encoding a certificate identifier or a reference to a verification service. Employers and other authorized users can scan the QR code to retrieve the corresponding certificate information and initiate verification. When integrated with a blockchain-based platform, QR-code-assisted access can reduce the effort required to locate certificate records and support convenient verification. A decentralized certificate issuance and verification architecture can further organize the certificate lifecycle, from registration and issuance to subsequent authentication [5].

Recent developments have also explored privacy-preserving verification mechanisms. Although blockchain can improve record integrity, recording sensitive personal information directly on a public blockchain may create privacy concerns because blockchain records can remain accessible for extended periods. Zero-knowledge proofs offer a potential

approach for demonstrating that certain claims are valid without revealing all the underlying information. Their application to academic record verification represents a promising direction for balancing trustworthy credential authentication with data privacy [6]. Other recent studies have continued to investigate blockchain-based academic certificate verification and secure credential authentication, demonstrating the ongoing relevance of this research area [7], [8].

To address the limitations of conventional certificate verification methods, this paper presents a Blockchain and Smart Contract-Based Digital Certificate Verification System. The proposed implementation integrates student registration, academic record entry, cryptographic hash generation, blockchain transaction storage, QR-code generation, and certificate verification into a unified application platform. The system uses an Ethereum-compatible blockchain environment and Solidity smart contracts to register certificate-related information and support authenticity checks. During certificate issuance, the application generates a cryptographic hash for the relevant certificate information and stores the hash on the blockchain. A QR code is generated to facilitate access to the corresponding verification process. During verification, the submitted certificate is processed to generate a new hash, which is compared with the registered hash retrieved through the blockchain-based verification mechanism. If both values match, the certificate content is consistent with the registered record. Otherwise, the system reports a mismatch, indicating that the submitted document may have been altered or may not correspond to the registered certificate. This process reduces dependence on manual verification and provides a systematic method for detecting inconsistencies in digital credentials.

The primary objective of this research is to develop a practical, secure, and efficient platform for academic certificate authentication using blockchain technology, smart contracts, and cryptographic hashing. The proposed system aims to improve the integrity of digital academic records, simplify verification procedures, reduce administrative effort, and increase confidence in electronically issued certificates. By integrating blockchain-based record management with a user-friendly verification mechanism, the implementation provides a foundation for more trustworthy digital credential management.

II. PROBLEM STATEMENT

Traditional academic certificate verification systems rely on centralized databases, paper-based records, and manual verification procedures, which can be time-consuming, costly, and vulnerable to certificate forgery and unauthorized modification. Employers and other organizations often need to contact educational institutions to validate the authenticity of submitted certificates, resulting in delays and increased administrative workload. Furthermore, centralized storage systems may be susceptible to unauthorized access, data manipulation, and single points of failure. These limitations reduce the reliability, transparency, and efficiency of academic credential management.

Therefore, there is a need for a secure, transparent, and efficient digital certificate verification system that can authenticate academic credentials while minimizing manual intervention. The proposed system addresses these challenges

by integrating blockchain technology, smart contracts, cryptographic hashing, and QR-code-based verification. Blockchain maintains tamper-evident certificate records, while smart contracts automate certificate registration and verification operations. Cryptographic hash comparison helps identify modifications to registered certificate content, and QR codes provide convenient access to the verification process. The primary objective is to develop a reliable and user-friendly platform that reduces certificate fraud, improves verification efficiency, and enhances trust in digital academic credentials.

III. RELATED WORK

Blockchain technology has gained considerable attention in the field of digital credential management because of its ability to provide decentralized, transparent, and tamper-evident record storage. Several research studies have investigated blockchain-based approaches to academic certificate issuance, authentication, and verification. These studies explore different techniques involving distributed ledgers, smart contracts, cryptographic hashing, decentralized storage, and privacy-preserving mechanisms to overcome the limitations of conventional certificate verification systems.

Rustemi et al. [1] presented a systematic literature review of blockchain-based systems for academic certificate verification. The authors examined existing research and identified major themes, contributions, and research gaps in the application of blockchain technology to educational credentials. Their study highlighted the potential of blockchain to support immutable digital certificates, improve verification efficiency, and reduce dependence on centralized authorities. However, the review also identified opportunities for further investigation into practical implementation, scalability, interoperability, and the adoption of blockchain-based verification systems.

The study presented in [2] investigated a decentralized certificate management approach using Ethereum blockchain technology and smart contracts. Ethereum provides a programmable blockchain environment in which smart contracts can automate predefined operations associated with certificate registration and verification. Such an approach can reduce manual intervention and improve the consistency of credential authentication. Nevertheless, practical deployment requires consideration of transaction costs, blockchain performance, smart contract security, and the accessibility of the verification interface.

Sultana et al. [3] proposed an IPFS-blockchain smart contract-based conceptual framework to reduce certificate fraud in the academic field. Their approach combines blockchain technology with cryptographic protection and the InterPlanetary File System (IPFS) to manage academic certificate information. The framework uses blockchain records to support integrity verification while IPFS provides a mechanism for storing larger encrypted data objects outside the blockchain. This approach helps address storage limitations associated with recording complete documents directly on a blockchain. However, practical deployment must also consider data availability, access control, confidentiality, and the management of decentralized storage references.

The work reported in [4] investigated digital certificate forgery verification using blockchain technology and smart contracts. The study addresses the problem of validating digital certificates and identifying fraudulent or modified credentials. Blockchain-based verification can provide a trusted reference against which submitted certificate information can be checked. Smart contracts can further automate the verification process and reduce the need for repeated manual confirmation by issuing institutions. The effectiveness of such a system depends on the reliability of the original certificate registration process and the security of the verification mechanism.

The research described in [5] focused on decentralized certificate issuance and verification using Ethereum blockchain technology. Decentralized approaches can support certificate management across multiple participants while providing a mechanism for verifying previously registered credentials. Smart contracts can be used to organize certificate-related operations and maintain consistent verification rules. However, system usability, transaction-processing overhead, secure identity management, and the integration of certificate issuance with existing institutional systems remain important considerations for practical implementation.

The study presented in [6] explored a blockchain-based academic record verification system incorporating zero-knowledge proofs. Zero-knowledge proof techniques can enable a party to demonstrate the validity of a claim without revealing all the underlying information. This capability is particularly relevant to academic credential verification, where sensitive student information should be protected. Although privacy-preserving verification can reduce unnecessary disclosure of personal data, its integration may introduce additional computational complexity and implementation requirements.

Recent work on academic certificate verification has also continued to investigate blockchain technology as a means of improving credential authenticity and reducing the risk of fraudulent academic records [7]. These approaches emphasize the importance of reliable verification mechanisms that allow authorized parties to establish whether a credential corresponds to a legitimate record. Similarly, the research described in [8] addresses secure academic certificate authentication using blockchain-based mechanisms. Such developments demonstrate the continuing interest in decentralized credential management and the need for practical systems that combine security, reliability, and ease of use.

Research Gap

The reviewed studies demonstrate the potential of blockchain, smart contracts, cryptographic techniques, and decentralized storage to improve academic certificate verification. However, developing a practical verification platform requires these technologies to be integrated into a coherent workflow that supports certificate registration, issuance, authentication, and convenient access by authorized users.

The proposed system addresses this implementation requirement by integrating student registration, academic record entry, cryptographic hash generation, blockchain-based hash storage, QR-code generation, and certificate verification into a unified application. A Solidity smart contract manages the blockchain-related operations, while cryptographic hash

comparison helps determine whether the submitted certificate content matches the registered record. The QR-code-based interface simplifies access to the verification process and reduces the effort required from employers and other authorized users.

Unlike approaches that require users to interact directly with blockchain tools, the proposed implementation emphasizes an accessible application interface combined with automated verification. Its principal objective is to provide a practical and user-friendly mechanism for checking certificate integrity while maintaining a tamper-evident reference on the blockchain.

IV. PROPOSED METHODOLOGY

The proposed system implements a blockchain-based digital certificate verification workflow using an application interface, cryptographic hashing, an Ethereum-compatible blockchain, Solidity smart contracts, and QR-code-assisted verification. The methodology is organized into certificate registration, hash generation, blockchain recording, QR-code generation, and certificate verification. The blockchain stores the certificate fingerprint and associated reference information rather than requiring the complete certificate document to be written directly to the chain.

A. System Architecture

The system contains the following logical components: (i) an institution or administrator interface for entering student and certificate details; (ii) a certificate-processing module that prepares the certificate data and calculates its cryptographic hash; (iii) a Solidity smart contract deployed on an Ethereum-compatible blockchain; (iv) a QR-code generator that encodes a certificate identifier or verification reference; and (v) a verifier interface used by employers or other authorized users. The verifier retrieves the registered hash through the smart contract and compares it with the hash calculated from the submitted certificate.

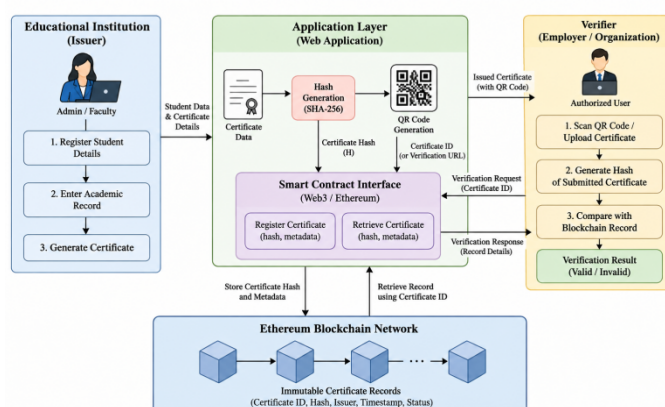


Fig. 2. Illustrative System Architecture

- Institution/administrator: enters student details and registers a certificate.
- Hash-generation module: creates a fixed-length digital fingerprint from the certificate data.
- Blockchain and smart contract: store and retrieve the registered hash and certificate identifier.

- QR-code module: creates a scannable reference to the certificate verification page or identifier.
- Verifier: submits or selects the certificate and checks whether its hash matches the registered hash.

B. Certificate Data Preparation

Let C denote the digital certificate content selected for verification. Before hashing, the system must use a consistent representation of the certificate data. If the system hashes structured fields, the fields should be serialized in a fixed order and format. If it hashes the complete file, the same file bytes must be used during later verification; even a change in file encoding or metadata may change the resulting hash.

$$C = \text{Serialize}(\text{StudentID}, \text{CertificateID}, \text{Name}, \text{Qualification}, \text{Institution}, \text{IssueDate})$$

Here, $\text{Serialize}(\cdot)$ denotes a deterministic serialization operation. The exact fields should be limited to those actually collected by the implementation. The student identifier and certificate identifier should be unique within the issuing institution.

C. Cryptographic Hash Generation

A cryptographic hash function maps the prepared certificate representation to a fixed-length digest. For a system using SHA-256, the registered certificate hash is calculated as follows:

$$H_{\text{reg}} = \text{SHA-256}(C)$$

The digest H_{reg} acts as the certificate's digital fingerprint. A change in the input is expected to produce a different digest with overwhelming probability. The hash is not encryption and cannot be used to recover the original certificate. For structured data, deterministic serialization is important so that the same certificate details produce the same digest.

D. Blockchain Registration Using a Smart Contract

After calculating H_{reg} , the application invokes a Solidity smart-contract function to register the certificate identifier and its hash on the Ethereum-compatible blockchain. A simplified logical record can be represented as:

$$R = (\text{CertificateID}, H_{\text{reg}}, \text{IssuerID}, \text{Timestamp}, \text{Status})$$

The smart contract can reject duplicate certificate identifiers and restrict registration to authorized issuers, provided these controls are implemented in the contract. Once the transaction is accepted and confirmed, the record can be queried during verification. Only the fields required for verification should be stored on-chain; sensitive personal information should not be written to a public blockchain.

A simplified registration rule is:

$$\text{Register}(\text{CertificateID}, H_{\text{reg}}) \text{ if } \text{CertificateID} \notin \text{BlockchainRecords}$$

This expression describes the intended logic and is not Solidity source code. Access control, duplicate handling, and

transaction confirmation must be enforced by the actual smart-contract implementation.

E. QR-Code Generation

Following successful registration, the application generates a QR code containing a certificate identifier or a verification URL that references the identifier. The QR code is an access mechanism; it does not itself prove that a certificate is genuine. Authenticity is determined by the verification process and the blockchain record retrieved for the corresponding identifier.

$$QR = \text{Encode}(\text{CertificateID or VerificationURL})$$

F. Certificate Verification

When a verifier scans the QR code, the application extracts the certificate identifier or opens the verification page. The system obtains the registered digest H_{chain} from the smart contract and calculates a fresh digest from the submitted certificate using the same preparation and hashing procedure.

$$H_{\text{sub}} = \text{SHA-256}(C_{\text{sub}})$$

$$V = 1, \text{ if } H_{\text{sub}} = H_{\text{chain}}; \quad \text{otherwise } V = 0$$

Here, C_{sub} represents the submitted certificate content, H_{sub} is its computed hash, H_{chain} is the registered hash retrieved from the blockchain, and V is the verification result. If $V = 1$, the submitted content matches the registered record. If $V = 0$, the content does not match the record or no valid record is available. A mismatch indicates inconsistency; it should not automatically be interpreted as proof of deliberate fraud.

G. Algorithmic Workflow

1. Start the application and connect to the Ethereum-compatible blockchain.
2. Authenticate the institution or administrator, where authentication is supported.
3. Enter the student and certificate details and assign a unique certificate identifier.
4. Prepare a deterministic representation of the certificate content.
5. Calculate the certificate digest H_{reg} using the selected cryptographic hash function.
6. Submit the certificate identifier and H_{reg} to the Solidity smart contract.
7. Wait for transaction confirmation and verify that the record was registered.
8. Generate a QR code containing the certificate identifier or verification URL.
9. For verification, scan the QR code and retrieve the corresponding blockchain record.
10. Calculate H_{sub} from the submitted certificate using the same data-preparation procedure.
11. Compare H_{sub} with H_{chain} and display a match, mismatch, or record-not-found result.
12. End the process.

H. Verification Performance Measures

The implementation can be evaluated using verification time, registration transaction time, and detection accuracy on a test

set containing original, modified, and unregistered certificates. The following expressions define useful measurements when corresponding experimental data are collected.

$$T_{\text{verify}} = t_{\text{result}} - t_{\text{request}}$$

$$\text{Detection Rate (\%)} = (\text{Tampered files correctly flagged} / \text{Tampered files tested}) \times 100$$

$$\text{False Acceptance Rate (\%)} = (\text{Invalid files accepted} / \text{Invalid files tested}) \times 100$$

The measured values should be reported from actual experiments. No performance figures are assumed in this methodology. Registration latency and verification latency should be measured separately because blockchain transaction confirmation and local hash computation have different costs.

I. Security and Implementation Considerations

The integrity of the verification result depends on secure initial registration, issuer authorization, the correct handling of certificate identifiers, and the use of a consistent hashing procedure. A blockchain record can establish that submitted content matches a registered digest, but it cannot independently prove that the original certificate was legitimately issued. The application should avoid placing unnecessary personal data on-chain and should protect administrative credentials and signing keys. QR codes should contain references rather than sensitive student data.

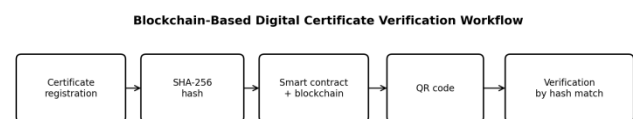
J. Methodology Scope

This methodology describes the workflow supported by the proposed application: certificate registration, cryptographic hash storage through a Solidity smart contract, QR-code-assisted access, and hash-based verification. IPFS storage and zero-knowledge proofs are not assumed to be implemented features; they are discussed only as possible approaches in related work unless they are present in the actual system

V. RESULTS AND DISCUSSIONS

This section discusses the expected functional outcomes of the proposed blockchain-based digital certificate verification workflow. The available project context defines certificate registration, SHA-256 hash generation, Solidity smart-contract recording on an Ethereum-compatible blockchain, QR-code generation, and verification by comparing a newly calculated hash with the registered hash. No actual test logs, application screenshots, or measured performance values were provided; therefore, the figures below are illustrative diagrams/mock-ups and the discussion does not claim experimentally measured performance.

A. Overall System Workflow



Illustrative workflow based on the proposed methodology

Fig. 3. Illustrative workflow of the proposed certificate registration and verification system.

The workflow begins when an authorized institution enters certificate details through the application. The certificate data are processed consistently and converted into a SHA-256 digest. The certificate identifier and digest are then registered through a Solidity smart contract. After the blockchain transaction is confirmed, the system generates a QR code containing a certificate identifier or verification reference. During verification, the QR code directs the verifier to the relevant record, and the submitted certificate is checked against the registered digest.

B. Hash-Based Authenticity Verification

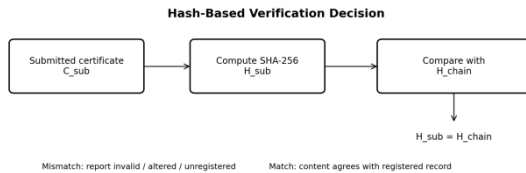


Fig. 4. Hash comparison used to determine whether submitted content matches the registered record.

Let H_{sub} denote the SHA-256 hash calculated from the submitted certificate and H_{chain} denote the hash retrieved from the blockchain. The system reports a match when $H_{sub} = H_{chain}$. A match means that the submitted content corresponds to the registered fingerprint. If the hashes differ, the system should report a mismatch or request further review. A mismatch can result from document modification, a different file representation, an incorrect certificate, or a missing/incorrect record; it is not by itself proof of intentional fraud.

$$V = 1, \text{ if } H_{sub} = H_{chain}; \text{ otherwise } V = 0$$

C. Illustrative Verification Results

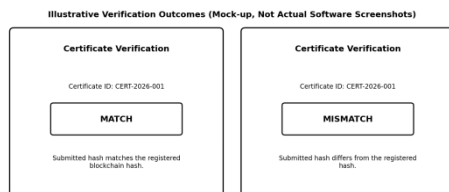


Fig. 5. Illustrative mock-up of match and mismatch outcomes; these are not screenshots of the implemented application.

The mock-up shows two possible outcomes. In the match case, the submitted certificate digest equals the value stored on the blockchain, so the application can display that the certificate content matches the registered record. In the mismatch case, the digest differs, and the application should flag the record for further checking. For a complete implementation, the interface should also distinguish a certificate identifier that is not found from a certificate whose hash does not match.

ADDRESS	BALANCE	TO	FROM
0x0001E6886e3D954c488016fC848534985F3A968	98.80 ETH	TO	FROM
0x77C5a328726e2f08ad7803d0177E47C6A86e7887	182.80 ETH	TO	FROM
0xb2791347c02607a776474918f72694cA978D61	100.00 ETH	TO	FROM
0x2b1Ea26C9F8158a3fa97251A9ec11240608279	100.00 ETH	TO	FROM
0xf098732a9f597A802182b05808e1a7651f55192	100.00 ETH	TO	FROM

Fig. 6. Illustrate the real time Output

D. Discussion of Functional Benefits

The proposed workflow combines cryptographic hashing with blockchain recordkeeping. Hashing provides a compact fingerprint for comparison, while the blockchain serves as a tamper-evident reference for the registered digest. The smart contract can standardize registration and retrieval operations, and the QR code can reduce the effort needed to access the verification process. These functions can help reduce dependence on manual confirmation with the issuing institution. Their reliability still depends on correct initial issuance, appropriate issuer authorization, secure management of signing keys, and consistent certificate-data preparation.

E. Performance Evaluation Required

Quantitative claims should be added only after executing tests on the actual application and blockchain environment. The following measures are suitable for evaluation:

- Registration latency: time from submitting a registration transaction to confirmed registration.
- Verification latency: time from initiating verification to displaying the result.
- Tamper-detection rate: proportion of modified test certificates correctly reported as mismatches.
- False acceptance rate: proportion of invalid or unregistered certificates incorrectly accepted.
- Transaction cost: blockchain gas used by the registration and verification transactions, where applicable.

For example, verification latency can be calculated as $T_{verify} = t_{result} - t_{request}$. The tamper-detection rate can be calculated as $(\text{modified certificates correctly flagged} / \text{modified certificates tested}) \times 100\%$. These metrics should be populated with measured values from repeated trials; no numerical results are asserted here.

F. Limitations

A blockchain can preserve a registered hash but cannot independently establish that the certificate was legitimate when first issued. If incorrect or fraudulent data are registered by an unauthorized or compromised issuer, the blockchain may preserve that incorrect record. In addition, hashing the full file can produce a mismatch if benign metadata or encoding changes. The implementation should define whether it hashes the complete file bytes or a canonical representation

of selected fields. Sensitive student data should not be stored directly on a public blockchain unless a suitable privacy and access-control design is in place.

G. Conclusion of Results and Discussion

The described design provides a clear mechanism for registering certificate fingerprints and comparing submitted certificate content with blockchain records. The workflow supports a match/mismatch decision and uses QR codes as a convenient entry point for verification. The figures in this document are explanatory illustrations rather than evidence of completed experiments. Actual application screenshots, test cases, and measured performance results should be inserted after testing the implemented system.

VI. CONCLUSION

This paper presents a blockchain and smart contract-based digital certificate verification system designed to improve the security, reliability, and efficiency of academic certificate authentication. The proposed system integrates cryptographic hashing, Ethereum blockchain technology, Solidity smart contracts, and QR-code-based verification to establish a systematic mechanism for certificate registration and validation. By storing the cryptographic hash of a registered certificate on the blockchain, the system provides a tamper-evident reference for checking the integrity of submitted academic credentials.

The verification process involves generating a hash of the submitted certificate and comparing it with the corresponding hash retrieved from the blockchain. A matching hash indicates that the submitted certificate content is consistent with the registered record, whereas a mismatch identifies a potential inconsistency or modification. Furthermore, QR-code integration simplifies access to the verification process, while smart contracts support automated certificate registration and retrieval operations. These features can reduce dependence on manual verification procedures and improve the accessibility of digital academic credential authentication.

The proposed approach offers a practical foundation for reducing certificate forgery, strengthening trust in digital credentials, and simplifying academic record verification. However, the effectiveness of the system depends on the authenticity of the initially registered information, appropriate issuer authorization, secure management of cryptographic keys, and consistent certificate-data processing. Future work may focus on integrating decentralized storage technologies such as the Inter Planetary File System (IPFS), incorporating privacy-preserving verification techniques, and evaluating the system's scalability, transaction costs, and verification latency under different workloads.

Overall, the proposed system demonstrates how blockchain technology and smart contracts can support secure and

transparent digital certificate management. With further implementation and experimental validation, the approach has the potential to provide a reliable solution for educational institutions, employers, and other organizations requiring efficient academic credential verification.

ACKNOWLEDGMENT

We would like to express our sincere gratitude to our esteemed guide, **Dr. S. Rama Devi**, Department of Electronics and Communication Engineering (ECE), Andhra University College of Engineering for Women, for her valuable guidance, continuous encouragement, and constant support throughout the successful completion of this project. Her insightful suggestions, technical expertise, and constructive feedback have played a significant role in helping us complete this project successfully.

We extend our heartfelt thanks to the management and faculty members of **Andhra University College of Engineering for Women** for providing the necessary facilities, resources, and academic environment to carry out this project. We, **Killada Veera Venkata Padmavathi Sirisha, Murupaka Meghana, Jakkula Harshitha and Pillala Sriharika**, express our sincere appreciation to our guide and all the faculty members who supported us throughout the project. We also thank our families and friends for their constant motivation, encouragement, and support. Finally, we acknowledge everyone who directly or indirectly contributed to the successful completion of this project.

REFERENCES

- [1] A. Rustemi, F. Dalipi, V. Atanasovski, and A. Risteski, "A Systematic Literature Review on Blockchain-Based Systems for Academic Certificate Verification," *IEEE Access*, vol. 11, pp. 64679–64696, 2023. doi: 10.1109/ACCESS.2023.3289598.
- [2] "The Decentralized Smart Contract Certificate System Utilizing Ethereum Blockchain Technology," *Procedia Computer Science*, vol. 230, pp. 923–934, 2023. doi: 10.1016/j.procs.2023.12.035.
- [3] S. A. Sultana, C. Rupa, R. P. Malleswari, and T. R. Gadekallu, "IPFS-Blockchain Smart Contracts Based Conceptual Framework to Reduce Certificate Frauds in the Academic Field," *Information*, vol. 14, no. 8, Art. no. 446, 2023. doi: 10.3390/info14080446.
- [4] "A Verification of Digital Certificate Forgery Using Blockchain Technology and Smart Contract," in *Proceedings of the 2024 8th International Conference on Information Technology (InCIT)*, 2024. doi: 10.1109/InCIT63192.2024.10810617.
- [5] "Decentralized Certificate Issuance and Verification System Using Ethereum Blockchain Technology," *Journal of Network and Computer Applications*, 2025. doi: 10.1016/j.jnca.2025.104190.
- [6] "A Zero-Knowledge Proof-Enabled Blockchain-Based Academic Record Verification System," *Sensors*, vol. 25, no. 11, Art. no. 3450, 2025. doi: 10.3390/s25113450.
- [7] "Academic Certificate Verification Using Blockchain Technology," in *2025 IEEE International Conference on Contemporary Computing and Communications (InC4)*, 2025.
- [8] "Secure Academic Certificate Authentication Using Blockchain Technology," in *2025 9th International Conference on Computational System and Information Technology for Sustainable Solutions (CSITSS)*, 2025. doi: 10.1109/CSITSS67709.2025.11294218.