Document protection using Blockchain technology

Ayushi Kumari[1], Bhavani B[2], Bindu M[3], Harshitha V M[4]
[1,2,3,4]*Dept. Computer Science, Sir M Visvesvaraya Institute of Technology, Bangalore, India*

Abstract **-Digital documents are widely used in education, business, law, and government, but they can be easily copied, edited, or misused. Traditional verification systems depend on centralized databases, which create security risks and do not provide clear proof of authenticity. To solve this problem, this paper presents a blockchain-based document protection system that ensures secure storage, verification, and tracking of digital files. The system uses SHA-256 hashing to generate a unique fingerprint for each document and stores it on the Ethereum blockchain through smart contracts. The actual file is uploaded to IPFS, providing decentralized and scalable storage. A QR code is generated for every document, allowing quick public verification through a web interface developed using Flask. Audit logs, role-based access, and encryption further improve transparency and security. Testing shows that the system is reliable, tamper-proof, and efficient for real-time verification. This framework can be applied to certificates, legal records, academic documents, and forensic data, offering a trustworthy solution for institutions that require secure digital document management.**

I. INTRODUCTION

Digital Document Integrity (DDI) refers to the framework and methodologies used to ensure the authenticity, reliability, and trustworthiness of digital documents—such as legal contracts, academic certificates, and medical records—throughout their entire lifecycle. It includes the processes of creation, storage, verification, and transfer while preventing unauthorized modification. The importance of strong digital document protection is paramount, as these documents are now central to critical operations including court proceedings, university admissions, financial transactions, and corporate governance. The ability to validate a document's authenticity directly influences legal rulings, academic integrity, and organizational transparency. Digital documents are more vulnerable than physical records due to factors such as ease of duplication, undetectable alterations, reliance on centralized storage systems (creating single points of failure), and the absence of inherent, verifiable proof of provenance. The foundation of reliable digital document management is the Immutable Chain of Integrity (CoI), a secure, chronologically ordered ledger that records every action performed on a document. It transparently logs the 'Four Ws'—Who accessed it, When the action occurred, Why it was accessed, Where it was stored, and How it was modified—throughout its lifecycle. The objective of this project is to develop a system that protects a digital document from its creation to its final use, ensuring its integrity remains intact and verifiable by authorized parties at any time. In traditional environments, as a document moves through multiple stakeholders (e.g., creators, reviewers, verifiers, and auditors), the likelihood of tampering, unauthorized access, or disputes over its history increases significantly. Existing centralized databases and manual logging practices are inadequate to provide the level of trust required for legal and academic acceptance. Through a detailed review of current literature and technological solutions, we have identified that blockchain technology is uniquely capable of addressing these issues. Its core characteristics—decentralization, immutability, and transparency—offer a strong foundation for establishing a tamper-proof Chain of Integrity. Therefore, integrating blockchain into document management can significantly enhance traceability, accountability, and long-term data assurance.

II. EXISTING LITERATURE SURVEY

A. Title: "A Comprehensive Study on Blockchain Applications for Digital Evidence Management"
Author: Rajesh Kumar et al. (2021) Kumar and co-researchers explored how blockchain technology can be applied to manage and secure digital evidence efficiently. Their paper emphasizes how distributed

ledgers improve transparency, authenticity, and immutability of forensic data.

B. Title: "A Blockchain-Oriented Framework for Managing Forensic Evidence"

Author: John Smith et al. (2022) Smith and his research team introduced a blockchain-based evidence management structure designed to preserve data integrity and enable full traceability during forensic analysis

C. Title: "QR Code Integrated Blockchain Model for Evidence Verification and Tracking"

Author: Anil Kumar et al. (2023) Anil Kumar and co-authors developed a blockchain framework enhanced with QR code technology for streamlined forensic evidence management. Each QR code acts as a secure digital tag linked to blockchain records, simplifying evidence identification, retrieval, and verification while improving accountability in forensic operations.

D. Title: "AI-Enhanced Blockchain Mechanisms for Digital Evidence Authentication"

Authors: M. Johnson, D. Lee, and C. Wong (2024) Johnson and his collaborators proposed an innovative model that merges artificial intelligence with blockchain to automate evidence validation and detect anomalies. Their approach enhances reliability, reduces manual intervention, and improves the accuracy of forensic data authentication across investigation networks.

E. Title: "Blockchain-Based Chain-of-Custody for Preserving Forensic Data Integrity"

Authors: L. Chen, R. Banerjee, and T. Wilson (2025) Chen and the research team presented a blockchain-powered chain-of-custody protocol that maintains continuous evidence traceability. Their system ensures every transaction related to digital evidence is securely recorded, verifiable, and resistant to manipulation, strengthening data integrity throughout the forensic process.

III. METHODOLOGY

The methodology of the Blockchain-Based Document Protection System is designed to deliver a transparent, tamper-proof, and verifiable approach to document authentication by combining blockchain, decentralized storage, and cryptographic techniques. The primary aim is to ensure that every document's authenticity and integrity can be verified without relying on centralized intermediaries. This system replaces traditional manual verification processes, which are prone to forgery and data loss, with a decentralized and immutable digital process. The methodology follows a multi-phase workflow that involves user interaction, document hashing, blockchain recording, decentralized storage, and verification through QR codes. The system operates with clearly defined user roles: Administrator, Investigator, and Public Verifier. The Administrator manages the platform, user accounts, and blockchain network configuration. The Investigator (such as an institutional staff member) uploads and manages documents, while the Verifier or Public User can check document authenticity without needing a login. The process begins when an authorized user logs into the system through the web interface developed using Flask (Python). Flask handles user authentication, session management, and API communication. When a user uploads a document, the system immediately computes a SHA-256 cryptographic hash of the file. This hash acts as the document's unique digital fingerprint—any modification in the file, even a single byte, will generate a completely different hash value, enabling instant detection of tampering. Once the hash is generated, the document is uploaded to IPFS (InterPlanetary File System), a decentralized storage protocol that divides the file into smaller chunks and distributes them across several peer nodes. IPFS generates a Content Identifier (CID)—a unique code referencing the file's location in the decentralized network. This ensures that even if one node fails, the file remains retrievable through other nodes, guaranteeing both persistence and high availability. After generating the CID, the system communicates with the Ethereum blockchain through Web3.py, a Python library that allows interaction with deployed smart contracts. The blockchain stores both the SHA-256 hash and the IPFS CID as part of a transaction record. This transaction is permanently recorded on the blockchain, where it cannot be altered or deleted due to blockchain's immutability feature. The recorded transaction includes the uploader's ID, timestamp, and document metadata, ensuring a complete chain of custody. Once the blockchain transaction is confirmed, the system receives a transaction hash and block number, serving as verifiable proof of existence. Simultaneously, a QR code is generated using Python's qrcode library. This QR code encodes the verification

URL or blockchain transaction details and is attached to the digital or printed copy of the document. Anyone can scan this QR code to verify document authenticity through the system's public verification page. Flask's backend continuously maintains audit logs for every operation—uploads, verifications, and administrative activities—in a secure JSON-based log file. Each log record includes timestamps, user roles, and actions performed, ensuring transparency and accountability. The system also incorporates Zero-Knowledge Proof (ZKP) principles for privacy-preserving verification. This ensures that verification confirms the authenticity of a document without exposing its confidential contents. The methodology integrates real-time synchronization between IPFS, the blockchain ledger, and local metadata storage. Flask periodically checks blockchain node status and connection integrity using background threads, ensuring uninterrupted service. This hybrid architecture ensures both decentralized immutability and local reliability. Overall, this methodology creates a secure, scalable, and efficient document verification system. By combining Flask, Ethereum, IPFS, and cryptographic hashing, it ensures that digital evidence remains tamper-proof and verifiable across institutional and research domains. The workflow ensures end-to-end transparency—from document creation to verification—making it ideal for educational certifications, research data, and forensic records.

IV. GAP ANALYSIS

A. Partial Decentralization

Most existing solutions rely on centralized components for authentication or verification, which undermines the principle of full decentralization. This dependency introduces potential single points of failure and reduces overall system trustworthiness.

B. Incomplete Evidence Lifecycle Coverage

Several studies focus mainly on secure storage or verification but fail to manage the entire evidence lifecycle—from acquisition and transfer to analysis and long-term archiving. This incomplete chain-of-custody reduces evidential continuity and weakens its admissibility in court.

C. Scalability and Performance Limitations

Current blockchain-based forensic systems experience high latency, increased transaction costs, and reduced throughput when handling large volumes of digital evidence. These performance constraints limit real-time operations and large-scale adoption in law enforcement environments.

D. Privacy and Confidentiality Concerns

While blockchain ensures transparency, it can expose sensitive metadata on public ledgers. Few studies have effectively implemented privacy-preserving methods such as Zero-Knowledge Proofs (ZKPs) or permissioned blockchain networks to safeguard confidential forensic information.

E. Poor Interoperability with Existing Systems

Most existing frameworks are standalone prototypes and lack integration with traditional forensic tools, legal databases, and evidence management systems. This lack of interoperability prevents smooth adoption by investigative and judicial institutions.

F. Limited Legal and Regulatory Alignment

Although blockchain provides immutability and traceability, many works fail to align with legal standards regarding evidence admissibility, data protection, and jurisdictional requirements. This gap restricts the implementation of blockchain-based forensic systems in real-world judicial scenarios.

V. PROPOSED SYSTEM CONFIGURATION

The proposed system configuration integrates web technology, blockchain, and decentralized storage into a cohesive framework that guarantees transparency, resilience, and scalability. It employs a five-layered architecture—Frontend, Backend, Blockchain, Storage, and Database—each layer performing specialized operations within the system. The frontend interface is built using HTML, CSS, JavaScript, and Bootstrap 5, providing a responsive and user-friendly dashboard for all user roles. It enables administrators to manage users and blockchain settings, investigators to upload and verify evidence, and public users to verify authenticity via QR code scanning. The user interface focuses on simplicity, using AJAX-based interactions to ensure smooth transitions without full page reloads. The backend uses Flask 2.3+, a lightweight Python web framework that manages routes, user sessions, form validations, and blockchain communication. Flask serves as the middleware between the web frontend and the blockchain layer through Web3.py. It is responsible for generating document hashes, connecting to IPFS nodes, and

invoking smart contract functions for document registration and verification. The backend also provides RESTful APIs for mobile and external integrations. The blockchain layer uses Ethereum, deployed locally on Ganache for testing and Truffle Suite for contract management. Smart contracts, written in Solidity (version 0.8.x), are designed to handle three main functions: document registration (storing SHA-256 hash, uploader ID, and timestamp), document verification (validating hashes and timestamps), and audit logging (recording every blockchain interaction for transparency). Once a smart

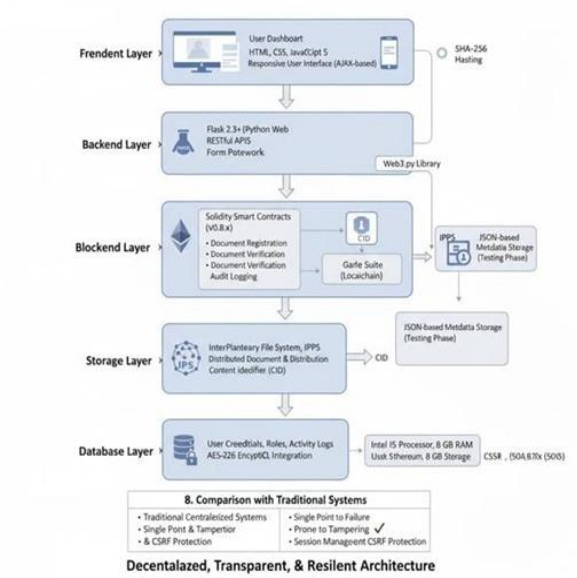

Fig 1. System architecture Diagram

contract is deployed, it functions autonomously, ensuring tamper-proof data records. For decentralized storage, IPFS (InterPlanetary File System) is integrated for distributed document storage. When a document is uploaded, it is split into blocks and distributed across multiple IPFS peers. IPFS generates a unique Content Identifier (CID) for each file, ensuring content-addressable storage. The CID is stored on the blockchain to establish a link between the document's data and its cryptographic proof. This layer enhances redundancy, scalability, and data persistence. A lightweight JSON-based storage is used for metadata and local backups during the testing phase, with provisions for SQLite or MySQL integration in production. The database stores user credentials, roles, and activity logs while ensuring all sensitive data is encrypted using AES-256 encryption. The system runs efficiently on a machine with an Intel i5 processor, 8 GB RAM, and 10 GB storage. Flask operates on Port 5000, Ethereum on Port 8545, and IPFS on Port 5001. These ports can be customized based on deployment needs. The system supports both Windows and Linux platforms, ensuring cross-compatibility. Security is enforced through cryptographic hashing, encryption, and role-based access control. SHA-256 protects document integrity, AES-256 secures credentials, and Flask's CSRF protection ensures safe user sessions. Traditional centralized systems rely on a single database or institution for verification, which can be easily manipulated. In contrast, this proposed configuration ensures no single point of failure. Blockchain immutability prevents tampering, while IPFS redundancy ensures uninterrupted document availability. Together, these technologies create a decentralized, transparent, and resilient architecture suitable for institutional adoption.

VI. IMPLEMENTATION DETAILS

The implementation of the Blockchain-Based Document Protection System involves coordinated integration of multiple technologies to form a fully functional decentralized verification environment. Each module was designed to work independently yet cohesively within the ecosystem. The development begins with setting up a local Ethereum blockchain using Ganache. Smart contracts are developed using Solidity to define functionalities such as document registration, hash verification, and transaction retrieval.

These contracts are compiled and deployed through Truffle, which connects with the Flask backend using Web3.py. The deployment process also initializes blockchain accounts and gas configurations. The Flask backend file, MainEnhanced.py, handles authentication, evidence upload, and blockchain communication. When an investigator uploads a document, Flask generates its SHA-256 hash using Python's hashlib library. The file is stored in IPFS, and its CID and hash are recorded on the blockchain via smart contract. Once confirmed, Flask stores the transaction hash, block number, and timestamp in its local metadata database. A QR code is generated using Python's qrcode and Pillow libraries, embedding the blockchain verification link.

This allows users to scan and verify authenticity through the web interface. Verification involves

comparing blockchain-stored hashes with recomputed local hashes. Matching values confirm authenticity; mismatches indicate tampering. All verification requests are logged with timestamps in the audit trail. Security is reinforced using AES-256 encryption for metadata, SHA-256 hashing for documents, and session timeouts for user management. Flask logs every upload, verification, and login action into an audit JSON file for accountability. Extensive testing confirmed that blockchain transactions are successfully recorded and validated. IPFS data redundancy was verified through multi-peer access. The end-to-end process—from upload to QR verification—was tested, ensuring consistent performance with an average verification time of two seconds. The modular design allows future integration with Zero-Knowledge Proofs, NFT-based certificates, and interoperability with public Ethereum networks. This implementation demonstrates how blockchain, IPFS, and Flask can cohesively deliver a transparent, decentralized, and tamper-proof document management system, serving as a foundation for future secure evidence management and academic data protection.

Additionally, the system's architecture ensures scalability and adaptability for diverse institutional needs. Future enhancements may include automated smart contract deployment pipelines and integration with distributed AI agents for real-time anomaly detection. Such improvements will further strengthen the platform's robustness, making it suitable for government, academic, and forensic record management applications.

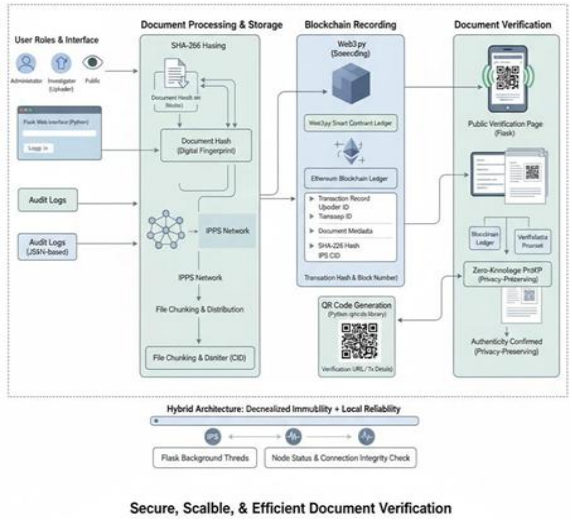

Fig .2. Implementation Flow Diagram

VII. BENEFITS OF PROPOSED SYSTEM

A. Data Integrity and Verification

The proposed blockchain-based forensic framework ensures that every piece of digital evidence is timestamped and immutably stored on the blockchain. This guarantees the authenticity and integrity of the data, preventing any unauthorized modification or deletion. The cryptographic verification process provides tamper-proof validation at every stage of the investigation.

B. Privacy and Controlled Access

Blockchain technology supports secure and permissioned sharing of forensic data among authorized users. Sensitive evidence information can be shared within closed networks while maintaining confidentiality. This controlled access minimizes the risk of unauthorized disclosure or tampering during evidence transmission.

C. Elimination of Intermediaries

Traditional forensic systems often depend on centralized authorities or intermediaries for verification. In the proposed framework, blockchain eliminates third-party dependence by automating authentication and validation through smart contracts. This decentralization reduces processing delays, lowers operational costs, and enhances the overall efficiency of the forensic workflow.

D. Immutability of Records

Once recorded on the blockchain, evidence-related data becomes permanent and unalterable. The immutable nature of blockchain ensures that no entity can modify or remove any record without detection. This feature preserves the reliability and evidential value of digital records for long-term legal and investigative purposes.

E. Transparency and Accountability

The proposed system enhances transparency and accountability among all participants involved in the investigation. Every access or modification is permanently logged, and all stakeholders can view the same verified version of the data in real time. This shared visibility promotes trust, prevents disputes, and ensures clear traceability of all forensic actions.

VIII. CONCLUSION

This study demonstrates that blockchain technology provides a dependable foundation for protecting and maintaining the integrity of digital evidence by recording every transaction in a secure transparent and immutable manner when integrated with the interplanetary file system ipfs and smart contracts it establishes a trustworthy system for evidence storage verification and traceability across its entire lifecycle the inclusion of zero-knowledge proofs zkps and qr-based methods further reinforces privacy controlled access and audit efficiency without reducing user convenience the modular framework supports scalability flexibility and ongoing system enhancement for practical forensic applications although legal standardization and system integration remain areas for improvement the proposed model shows strong potential to redefine digital evidence handling creating a more credible transparent and secure ecosystem for investigative and judicial processes.

FUTURE WORK

In the future, this system can be improved by connecting it to public blockchain networks like Ethereum or Polygon for wider verification. Artificial Intelligence (AI) can be added to detect unusual activities and give security alerts automatically. The use of Zero-Knowledge Proofs (ZKPs) can help verify documents without showing their contents, keeping privacy safe.

A mobile version of the system can be developed so users can upload and verify documents easily. Scalability can be improved using layer-2 blockchain solutions to reduce cost and delay. The system can also be integrated with government, academic, and legal databases for real-world use. These improvements will make the system faster, more secure, and suitable for large-scale applications.

REFERENCES

[1] D. Kim, S. Y. Ihm, and Y. Son, "Two-level blockchain system for digital crime evidence management," *Sensors*, vol. 21, no. 9, p. 3051, 2021.

[2] S. Rao, S. Fernandes, S. Raorane, and S. Syed, "A Novel Approach for Digital Evidence Management Using Blockchain," *Proceedings of the International Conference on Recent Advances in Computational Techniques (IC-RACT)*, June 2020.

[3] I. Hingorani, R. Khara, D. Pomendkar, and N. Raul, "Police complaint management system using blockchain technology," *2020 3rd International Conference on Intelligent Sustainable Systems (ICISS)*, pp. 1214–1219, Dec. 2020.

[4] K. Tabassum, H. Shaiba, S. Shamrani, and S. Otaibi, "e-Cops: An Online Crime Reporting and Management System for Riyadh City," *2018 1st International Conference on Computer Applications & Information Security (ICCAIS)*, Riyadh, pp. 1–8, 2018.

[5] P. Kormpho, P. Liawsomboon, N. Phongoen, and S. Pongpaichet, "Smart Complaint Management System," *2018 Seventh ICT International Student Project Conference (ICT-ISPC)*, Nakhonpathom, pp. 1–6, 2018.

[6] M. Archana, S. Durga, and K. Saveetha, "Online Crime Reporting System," *International Journal of Advanced Networking Applications (IJANA)*. [Online]. Available: https://www.ijana.in/ papers /82.pdf

[7] A. T. Dini, E. G. Abete, M. Colombo, J. Guevara, B. S. Menchon Hoffmann, and M. C. Abeledo, "Analysis of implementing blockchain technology to the Argentinian criminal records information system," *2018 Congreso Argentino de Ciencias de la Informática y Desarrollos de Investigación (CACIDI)*, pp. 1–3, 2018.

[8] J. Yang, J. Wen, B. Jiang, and H. Wang, "Blockchain-Based Sharing and Tamper-Proof Framework of Big Data Networking," *IEEE Network*, vol. 34, no. 4, pp. 62–67, Jul./Aug. 2020.

[9] M. Singh, G. S. Aujla, A. Singh, N. Kumar, and S. Garg, "Deep-learning-based blockchain framework for secure software-defined industrial networks," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 1, pp. 606–616, Jan. 2021.

[10] Z. Zheng and S. Xie, "An overview of Blockchain Technology: Architecture, Consensus and Future Trends," *IEEE 6th International Congress on Big Data*, 2017.

[11] M. Humayun, N. Jhanjhi, B. Hamid, and G. Ahmed, "Emerging smart logistics and transportation using IoT and blockchain," *IEEE*

Internet of Things Magazine, vol. 3, no. 2, pp. 58–62, Jan. 2020.

[12] A. Adewuyi, H. Cheng, Q. Shi, J. Cao, A. MacDermott, and X. Wang, "Ctrust: A dynamic trust model for collaborative applications in the Internet of Things," *IEEE Internet of Things Journal*, vol. 6, no. 3, pp. 5432–5445, 2019.

[13] B. Pourghebleh, K. Wakil, and N. Navimipour, "A comprehensive study on the trust management techniques in the Internet of Things," *IEEE Internet of Things Journal*, vol. 6, no. 6, pp. 9326–9337, 2019.

[14] S. Guleng, C. Wu, X. Chen, X. Wang, T. Yoshinaga, and Y. Ji, "Decentralized trust evaluation in vehicular Internet of Things," *IEEE Access*, vol. 7, pp. 15980–15988, 2019.

[15] L. Wei, Y. Yang, J. Wu, C. Long, and B. Li, "Trust management for Internet of Things: A comprehensive study," *IEEE Internet of Things Journal*, 2022.

[16] Ethereum Foundation, "Ethereum Whitepaper," 2025. [Online]. Available: https://ethereum.org/en/whitepaper

[17] Protocol Labs, "InterPlanetary File System (IPFS) Documentation," 2025. [Online]. Available: https://docs.ipfs.io

[18] D. Shim, "QRCode.js: A JavaScript Library for QR Code Generation," 2025. [Online]. Available: https://github.com/davidshimjs/qrcodejs