

A HYBRID BLOCKCHAIN-BASED KEY MANAGEMENT FRAMEWORK FOR ENHANCING SECURITY, TRUST, AND PERFORMANCE IN DIGITAL COMMUNICATION SYSTEMS

Yazeed Al Moaiad¹

Faculty of Computer and Information
Al-Madinah International University
Kuala Lumpur, Malaysia
yazeed.alsayed@mediu.edu.my

Abdullah Alsumaini²

Faculty of Computer and Information
Al-Madinah International University
Kuala Lumpur, Malaysia
BV071@lms.mediu.edu.my

Abstract— Secure key management is fundamental to the confidentiality, authentication, and integrity of digital communication, yet centralized key servers and single-chain blockchain designs present competing limitations in trust, privacy, auditability, and performance. This paper develops a hybrid blockchain-based key management framework that assigns frequent and sensitive lifecycle operations to a permissioned private blockchain while anchoring non-sensitive proofs to a public blockchain. A qualitative conceptual methodology was used, combining a concept-centric literature review, comparative analysis, framework design, and evaluation of an educational website prototype. The proposed architecture integrates users and devices, private-chain validators, smart-contract policy enforcement, public proof anchoring, and security monitoring. Smart contracts automate registration checks, key requests, authorization, rotation, revocation, and audit events without storing secret key material on-chain. The conceptual comparison indicates that the hybrid approach offers a more balanced security profile than centralized, public-only, or private-only alternatives: confidential operations remain controlled and responsive, while externally verifiable hashes strengthen integrity and accountability. The prototype demonstrates the framework and key lifecycle through an accessible web interface, but it is not a live blockchain implementation. The study concludes that hybrid blockchain can improve trust and operational efficiency only when accompanied by robust key custody, contract assurance, privacy-preserving metadata design, governance, and recovery mechanisms. Future work should deploy a working prototype, benchmark latency and throughput, test adversarial scenarios, and assess usability in realistic communication environments.

Keywords— hybrid blockchain; cryptographic key management; smart contracts; digital communication; cybersecurity; key lifecycle; auditability; Internet of Things

I. Introduction

Digital communication systems depend on cryptographic keys to protect confidentiality, authenticate users and devices, authorize access, and establish trustworthy sessions. The security of an encryption algorithm is therefore insufficient when the associated keys are generated, stored, distributed, rotated, recovered, or revoked incorrectly. As cloud services, Internet of Things (IoT) deployments, e-government platforms, health systems, and enterprise networks expand, key-management operations become more frequent, distributed, and difficult to govern. Contemporary cybersecurity guidance consequently treats asset governance, identity assurance, protection, detection, response, and recovery as connected responsibilities [16, 17, 27].

Traditional key management commonly relies on a central key server, certificate authority, hardware security infrastructure, or trusted administrative service. Centralized control may be efficient within a bounded organization, but it concentrates both technical and institutional risk. A server failure, credential compromise, malicious administrator, or configuration error can affect many users and devices simultaneously. In dynamic IoT and cross-domain communication, centralized repositories also struggle to provide transparent and independently verifiable records of who requested, approved, rotated, or revoked a key [10, 12].

Blockchain has been proposed as a distributed coordination mechanism for key events. Public blockchains provide openness, independent verification, and tamper-evident records, but they may expose metadata and introduce latency, cost, and scalability constraints. Private

blockchains protect confidentiality and offer faster validation, yet they provide less independent assurance when controlled by one organization or a small consortium. A hybrid architecture separates these concerns: a private blockchain performs sensitive and frequent operations, while a public blockchain stores selected hashes or proof references for external verification [2, 28, 29].

A. Research Problem and Objectives

Neither centralization nor a single blockchain type fully satisfies the combined requirements of secure digital communication. The research problem is to determine how public trust, private execution, automated lifecycle control, and usable explanation can be integrated without placing secret keys or sensitive metadata on a public ledger. The study therefore pursues four objectives: (1) identify limitations of centralized, public-only, and private-only key-management approaches; (2) design a hybrid framework for generation, distribution, use, rotation, and revocation; (3) analyze the contribution and risk of smart contracts; and (4) demonstrate the framework through an accessible website prototype.

B. Contributions

The paper contributes an integrated five-layer architecture, a security-conscious key-lifecycle workflow, and a comparative evaluation across security, privacy, performance, auditability, scalability, trust, and governance. It also distinguishes clearly between **operational records**, which remain within a permissioned environment, and **verification evidence**, which may be anchored publicly. This distinction is central: blockchain supports key governance and proof, but does not replace secure cryptographic key storage.

II. Background and Related Work

A. Cryptographic Key Management

Key management covers the entire life of cryptographic material: secure generation, registration, distribution, storage, authorized use, rotation, archival or recovery, and revocation. Each stage has distinct threats. Weak entropy undermines generation; insecure channels expose distribution; poor access control compromises storage; delayed rotation increases exposure; and ineffective revocation permits continued misuse. Group communication and IoT environments add membership churn, limited device capacity, multiple trust domains, and high event frequency [10, 18, 21].

Public-key infrastructure binds public keys to identities through signed certificates. It is mature and interoperable but depends on certificate authorities and revocation infrastructure. Symmetric distribution offers fast cryptographic operations but makes secret exchange difficult. Threshold cryptography reduces reliance on a single holder by dividing control, while attribute-based encryption supports fine-grained policy. These techniques remain valuable; the proposed framework does not replace them. Instead, it supplies a distributed governance and audit layer around key events.

B. Blockchain Models

A public blockchain permits broad participation and independent ledger inspection. These properties support integrity and auditability, but public consensus may be unsuitable for high-frequency lifecycle operations. A private blockchain restricts validation to authorized participants, enabling confidential and efficient processing at the cost of

broader public assurance. Consortium networks distribute validation among selected organizations, but require agreed governance. Hybrid blockchain combines private execution with public verification, allowing sensitive operations and metadata to remain controlled while selected commitments provide external evidence [1, 2, 3, 24].

Table 1 Blockchain models and their relevance to key management.

Model	Primary strength	Principal limitation	Key-management role
Public	Independent verification	Metadata, latency, cost	Proof and audit anchoring
Private	Speed and access control	Limited external trust	Sensitive lifecycle operations
Consortium	Shared institutional control	Complex governance	Cross-organization validation
Hybrid	Balanced execution and proof	Integration complexity	Private operation + public evidence

C. Smart Contracts and Security

Smart contracts can implement deterministic rules for key requests, access approval, rotation intervals, membership changes, revocation, and audit events. Automation reduces manual inconsistency and creates a traceable relationship between policy conditions and lifecycle actions [5, 23]. However, authorization errors, unsafe external calls, reentrancy, flawed upgrade paths, and incorrect state transitions can propagate at scale. Smart-contract quality assurance must therefore include threat modeling, code review, testing, static analysis, formal verification of critical properties, restricted administrative roles, and monitored deployment [14, 19, 20].

D. Research Gap

Existing studies demonstrate decentralized key management, blockchain-supported group keys, hybrid blockchain platforms, and contract-based access control. The remaining gap is a clear architecture that separates high-frequency private operations from publicly verifiable evidence while covering the complete lifecycle and its governance. A second gap concerns communication: technically accurate models are often difficult for decision makers and learners to understand. The present work addresses both gaps through a conceptual framework and an educational prototype.

III. Research Methodology

The study uses a qualitative, non-experimental, and conceptual research design. It synthesizes peer-reviewed articles, standards, technical guidance, and relevant implementation research rather than collecting participant data or measuring a deployed network. Sources were selected for direct relevance to blockchain architectures, key management, IoT security, smart contracts, secure software development, and web accessibility.

The methodology followed five stages. First, literature was analyzed by concept to identify recurring mechanisms, vulnerabilities, and design requirements. Second, centralized, public-only, private-only, and hybrid approaches were compared. Third, the findings were translated into a layered framework and lifecycle model. Fourth, a static website prototype was developed to explain the framework. Fifth, both framework and prototype were assessed qualitatively. Figure 1 shows the research sequence.



Figure 1 Research methodology used to develop and evaluate the framework.

The analytical criteria were security, privacy, performance, auditability, scalability, trust, and governance. Security concerns unauthorized access, modification, and misuse; privacy concerns exposure of keys and sensitive metadata; performance concerns latency and processing cost; auditability concerns traceability of events; scalability concerns growth in devices and requests; trust concerns reliance on a single authority; and governance concerns participation, upgrades, accountability, and dispute resolution. The website was assessed for content accuracy, clarity, navigation, visual consistency, and accessibility against established web-content principles [26]. Because no live deployment or user study was performed, the evaluation identifies expected properties and risks rather than measured effectiveness.

IV. Proposed Hybrid Key-Management Framework

A. Architectural Layers

The framework comprises five connected layers. The **user and device layer** includes people, applications, services, sensors, and gateways requesting secure communication. The **private blockchain performance layer** validates membership, identity or device status, and authorization; it also records confidential lifecycle events. The **smart-contract lifecycle layer** applies deterministic policy to requests, issuance, rotation, and revocation. The **public blockchain trust layer** stores non-sensitive commitments or hashes that enable later verification. Finally, the **monitoring and governance layer** correlates events, identifies anomalies, supports audits, and manages policy change.

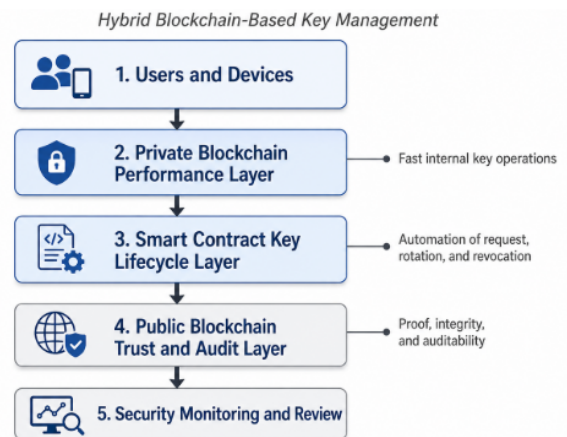


Figure 2 Five-layer hybrid blockchain key-management framework.

The separation of responsibilities shown in Figure 2 protects secret material and improves operational efficiency. Public records never contain encryption keys, private keys, plaintext identity data, or detailed access metadata. Instead, the public chain may contain a timestamped commitment derived from a lifecycle event. Authorized auditors can later compare a private record with its public commitment to detect unauthorized alteration.

B. Key Lifecycle Workflow

A user or device first registers through an approved identity and device-assurance process. When a secure session is required, the requester submits a key request to the private layer. Validators confirm status and context, and the smart contract evaluates role, policy, expiry, and risk conditions. If approved, a secure key-management service generates or retrieves the relevant key, transfers it through a protected channel, and records the event privately. A hash commitment can then be anchored publicly. Rotation replaces aging or exposed keys, while revocation immediately invalidates compromised or unauthorized material.

Table 2 Key lifecycle and required controls.

Stage	Hybrid operation	Security control
Register	Establish user or device status privately	Strong identity and enrollment
Request	Submit purpose, context, and requested scope	Authenticated channel
Validate	Contract evaluates status and policy	Least privilege; fail closed
Distribute	Release key through protected service	No secret on either ledger
Anchor	Publish hash of approved event	Minimal metadata non-linkable
Rotate	Replace key on schedule or risk trigger	Atomic status transition
Revoke	Invalidate compromised or expired key	Rapid propagation and audit

C. Smart-Contract Responsibilities

Contracts act as policy coordinators, not secret stores. They verify eligibility, enforce separation of duties, record approval states, trigger rotation or revocation, and create commitments for audit. Critical actions should require multi-party authorization or threshold approval. Contract updates must be transparent, versioned, independently reviewed, and protected by delay or emergency controls. The framework also requires an off-chain key vault or hardware security module because a ledger cannot securely perform all custodial functions by itself.

V. Conceptual Analysis and Findings

A. Security and Privacy

Compared with a single key server, distributed validation reduces dependence on one operational component and can make silent alteration of lifecycle records more difficult. Private execution protects key material and sensitive metadata, while signed events and public commitments improve integrity verification. Smart contracts can standardize authorization and revocation. These advantages remain conditional on endpoint security: compromised wallets, validators, administrative keys, or key vaults can still undermine the system. Hybrid architecture redistributes risk; it does not eliminate it [6, 10, 25].

Privacy depends on the data model. Hashes may still reveal relationships when events are predictable or linkable. Public commitments should therefore use domain-separated hashing, salt or nonce values where appropriate, aggregation, and minimal timestamps. Pairwise identifiers and controlled audit disclosure can reduce correlation. Private-chain access must be purpose-limited and monitored, because permissioned does not automatically mean private.

B. Trust and Auditability

The public layer supplies evidence that selected events existed at a particular time and were not silently rewritten after anchoring. This feature supports external assurance across organizations without giving the public access to operational records. Trust remains partly institutional: participants must still trust enrollment authorities, validator governance, key-custody procedures, and dispute-resolution processes. The result is a hybrid trust model that combines institutional accountability with cryptographic evidence rather than attempting to replace one with the other [7, 9, 15].

C. Performance and Scalability

Frequent requests, rotations, membership updates, and revocations are processed on the private chain, avoiding the confirmation delay and transaction cost of posting every event publicly. Only selected batches or commitments are anchored. This design is expected to improve response time and reduce public-chain load, especially in IoT environments. Ac-

tual benefits depend on validator topology, consensus protocol, anchor frequency, storage growth, network conditions, and key-vault capacity. Lightweight consensus and batching should be evaluated empirically before deployment [13, 24].

Table 3 Qualitative comparison. Expected performance assumes high-frequency operations remain private; empirical benchmarking is required.

Approach	Trust	Privacy	Speed	Audit
Centralized	Medium	Medium	High	Low
Public-only	High	Low	Low	High
Private-only	Medium	High	High	Medium
Hybrid	High	High	High (expected)	High

The comparative result in Table 3 supports four findings. First, hybrid blockchain offers the clearest separation between secret operations and public evidence. Second, smart contracts add value through consistent policy enforcement, but their defects become systemic risks. Third, auditability improves when private records can be matched to independent commitments. Fourth, performance claims remain hypotheses until validated on a defined platform and workload.

VI. Website Prototype

A static website named **HybridKey Shield** was developed as an educational companion to the framework. It presents the research problem, architecture, lifecycle, security comparison, and a stepwise demonstration. The interface uses consistent navigation, readable typography, high color contrast, and responsive content blocks. The demonstration shows how a request moves through private validation, contract logic, public proof anchoring, and later rotation or revocation.

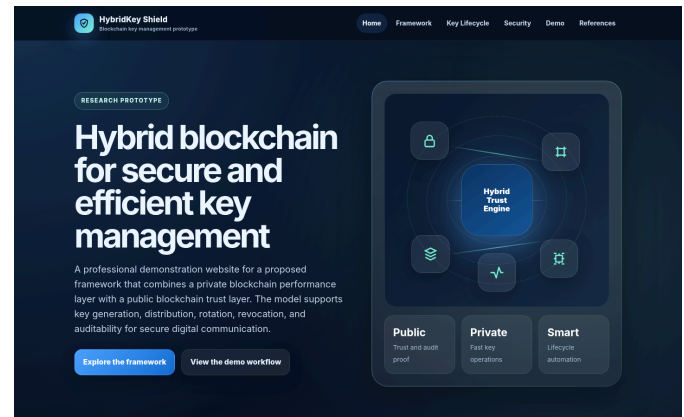


Figure 3 Home view of the educational website prototype.

The prototype was reviewed qualitatively for correctness, relevance, visual consistency, accessibility, and logical navigation. Its principal value is explanatory: it translates the conceptual architecture into a form that students, decision makers, and non-specialist stakeholders can explore. It does not connect to a live public or private blockchain, execute deployed smart contracts, manage real keys, or constitute evidence of production security. This distinction prevents the user interface from being mistaken for a validated cybersecurity system.

VII. Implementation and Governance Guidance

A deployment should proceed through bounded, evidence-based stages.

- 1) **Use-case and threat definition:** Specify participants, protected communications, adversaries, key types, latency targets, data classifications, and legal constraints.
- 2) **Governance design:** Define validator admission, public-anchor policy, contract ownership, administrative roles, incident response, liability, and dispute resolution.
- 3) **Cryptographic custody:** Use hardened key vaults or hardware security modules, authenticated channels, separation of duties, threshold approval for critical actions, and tested recovery.
- 4) **Contract assurance:** Apply the secure software development framework, independent review, automated testing, formal

checks for authorization invariants, staged upgrades, and continuous monitoring [14, 19, 20, 26].

- 5) **Pilot and measurement:** Compare the hybrid system with a centralized baseline under representative request rates, membership churn, validator failure, chain outage, and adversarial conditions.

The framework should also include a safe degradation strategy. Temporary unavailability of the public blockchain should not expose secret material or prevent emergency revocation. Anchors may be queued, signed locally, and published when service resumes, with clear evidence of the delay. Conversely, a private-chain compromise should trigger suspension, forensic review, and controlled re-establishment of trust rather than automatic acceptance of subsequent anchors.

VIII. Limitations and Future Research

The study is conceptual and does not provide a working blockchain network, cryptographic key exchange, deployed contract, or empirical performance benchmark. The comparison is qualitative and depends on design assumptions. The website prototype demonstrates concepts but was not evaluated through a controlled user study. Furthermore, blockchain governance and privacy requirements vary across sectors and jurisdictions.

Future work should implement a permissioned private chain, select a public anchoring mechanism, integrate a key-management service, and deploy audited lifecycle contracts. Experiments should measure request and revocation latency, throughput, public-anchor delay, storage growth, energy and transaction cost, resilience to validator loss, and recovery from network partition. Security evaluation should include unauthorized requests, contract upgrade abuse, metadata correlation, replay, key-vault compromise, malicious validator behavior, and denial of service. Further research should also examine post-quantum key encapsulation, threshold custody, multi-organization governance, and accessible recovery procedures.

IX. Conclusion

This paper proposed a hybrid blockchain-based framework for managing cryptographic keys in digital communication systems. The design assigns confidential and high-frequency lifecycle operations to a permissioned private blockchain, uses smart contracts to enforce policy, anchors non-sensitive commitments on a public blockchain, and adds monitoring and governance for accountability. Conceptual analysis indicates that this arrangement can offer a stronger balance of privacy, trust, auditability, and performance than centralized, public-only, or private-only alternatives.

The result is not that blockchain makes key management secure by default. Security depends on disciplined custody, minimal metadata, contract assurance, identity and device proofing, recovery, governance, and measurable operational performance. With these controls and empirical validation, the hybrid model provides a credible direction for resilient key management in distributed communication and IoT environments.

X. References

- [1] A. A. Al-awamy, N. Al-shaibany, A. Sikora, and D. Welte, "Hybrid consensus mechanisms in blockchain: A comprehensive review," *International Journal of Intelligent Systems*, vol. 2025, no. 1, Art. no. 5821997, 2025.
- [2] A. Alkhateeb, C. Catal, G. Kar, and A. Mishra, "Hybrid blockchain platforms for the Internet of Things: A systematic literature review," *Sensors*, vol. 22, no. 4, Art. no. 1304, 2022.
- [3] S. Almarri and A. Aljughaiman, "Blockchain technology for IoT security and trust: A comprehensive SLR," *Sustainability*, vol. 16, no. 23, Art. no. 10177, 2024.
- [4] M. Bordegoni, M. Carulli, and E. Spadoni, "Prototyping: Practices and techniques," in *Prototyping User Experience in Extended Reality*. Cham, Switzerland: Springer, 2023, pp. 29–47.
- [5] H. Chu, P. Zhang, H. Dong, Y. Xiao, S. Ji, and W. Li, "A survey on smart contract vulnerabilities: Data sources, detection and repair," *Information and Software Technology*, vol. 159, Art. no. 107221, 2023.
- [6] S. De Angelis, G. Zanfino, L. Aniello, F. Lombardi, and V. Sassone, "Evaluating blockchain systems: A comprehensive study of security and dependability attributes," in *CEUR Workshop Proceedings*, 2022.
- [7] A. De Salve, D. D. F. Maesa, P. Mori, L. Ricci, and A. Puccia, "A multi-layer trust framework for self-sovereign identity on blockchain," *Online Social Networks and Media*, vol. 37, Art. no. 100265, 2023.
- [8] A. Enaya, X. Fernando, and R. Kashef, "Survey of blockchain-based applications for IoT," *Applied Sciences*, vol. 15, no. 8, Art. no. 4562, 2025.
- [9] K. Huang, X. Zhang, Y. Mu, F. Rezaeiabagha, X. Du, and N. Guizani, "Achieving intelligent trust-layer for Internet-of-Things via self-redactable blockchain," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 4, pp. 2677–2686, 2020.
- [10] M. A. Kandi, D. E. Kouicem, M. Doudou, H. Lakhlef, A. Bouabdallah, and Y. Challal, "A decentralized blockchain-based key management protocol for heterogeneous and dynamic IoT devices," *Computer Communications*, vol. 191, pp. 11–25, 2022.
- [11] N. Karankar and A. Seth, "An IoT system for access control using blockchain and message queuing system," *EURASIP Journal on Information Security*, Art. no. 31, 2025.
- [12] M. Kharjana, F. H. Pohrmen, S. C. Sahana, and G. Saha, "Blockchain-based key management system in Named Data Networking: A survey," *Journal of Network and Computer Applications*, vol. 220, Art. no. 103732, 2023.
- [13] A. A. Maftai, A. I. Petrariu, V. Popa, and A. Lavric, "A blockchain framework for scalable, high-density IoT networks of the future," *Sensors*, vol. 25, no. 9, Art. no. 2886, 2025.
- [14] L. Marchesi, L. Pompianu, and R. Tonelli, "Security checklists for Ethereum smart contract development: Patterns and best practices," *Blockchain: Research and Applications*, Art. no. 100367, 2025.
- [15] N. Nadeem *et al.*, "Hybrid blockchain-based academic credential verification system," *Multimedia Tools and Applications*, vol. 82, no. 28, pp. 43991–44019, 2023.
- [16] National Institute of Standards and Technology, *Module-Lattice-Based Key-Encapsulation Mechanism Standard*, FIPS 203. Gaithersburg, MD, USA: NIST, 2024.
- [17] National Institute of Standards and Technology, *The NIST Cybersecurity Framework (CSF) 2.0*, NIST CSWP 29. Gaithersburg, MD, USA: NIST, 2024.
- [18] J. Ni, G. Fang, Y. Zhao, J. Ren, L. Chen, and Y. Ren, "Distributed group key management based on blockchain," *Electronics*, vol. 13, no. 11, Art. no. 2216, 2024.
- [19] Open Worldwide Application Security Project, *Application Security Verification Standard 5.0.0*, 2025.
- [20] Open Worldwide Application Security Project, *Smart Contract Top 10: 2025*, 2025.
- [21] J. C. Pérez-García, A. Braeken, and A. Benslimane, "Blockchain-based group key management scheme for IoT with anonymity of group members," *IEEE Transactions on Information Forensics and Security*, vol. 19, pp. 6709–6721, 2024.
- [22] D. Prasetyo, A. Utami, and T. G. Laksana, "Website based academic information system design using Extreme Programming method," *Journal of Informatics Information System Software Engineering and Applications*, vol. 6, no. 2, pp. 134–143, 2024.
- [23] A. Rashid, A. Masood, and A. U. R. Khan, "ACS-IoT: Smart contract and blockchain assisted framework for access control systems in IoT enterprise environment," *Wireless Personal Communications*, vol. 136, pp. 1331–1352, 2024.
- [24] S. Sahraroui and A. Bachir, "Lightweight consensus mechanisms in the Internet of Blockchain Things: Thorough analysis and research directions," *Digital Communications and Networks*, vol. 11, no. 4, pp. 1246–1261, 2025.
- [25] S. Soni and A. Singh, "A hybrid blockchain and smart contract framework for resilient IoT security in smart homes," *Frontiers in Blockchain*, vol. 8, Art. no. 1707911, 2025.
- [26] M. Souppaya, K. Scarfone, and D. Dodson, *Secure Software Development Framework (SSDF) Version 1.1*, NIST SP 800-218. Gaithersburg, MD, USA: NIST, 2022.
- [27] D. Temoshok *et al.*, *Digital Identity Guidelines*, NIST SP 800-63-4. Gaithersburg, MD, USA: NIST, 2025.
- [28] G. Tripathi, M. A. Ahad, and G. Casalino, "A comprehensive review of blockchain technology: Underlying principles and historical background with future challenges," *Decision Analytics Journal*, vol. 9, Art. no. 100344, 2023.
- [29] T. Wang, Q. Wu, J. Chen, F. Chen, D. Xie, and H. Shen, "Health data security sharing method based on hybrid blockchain," *Future Generation Computer Systems*, vol. 153, pp. 251–261, 2024.
- [30] Y. Al Moaiad, D. Alkhateeb, and M. Alokla, "Enhancing cybersecurity practices in Nigerian government institutions: An analysis and framework," *Journal of Electrical Systems*, vol. 20, no. 3, pp. 5964–5967, 2024.
- [31] Y. Al Moaiad, D. Alkhateeb, and M. Alokla, "Deep analysis of iris print and fingerprint for detecting drug addicts," *Journal of Electrical Systems*, vol. 20, no. 3, pp. 5639–5644, 2024.
- [32] Y. Al Moaiad, M. Alobed, and M. Alsakhnini, "Python solutions to address natural language challenges," *International Journal*, vol. 10, no. 3, pp. 3594–3603, 2024.
- [33] Y. Al Moaiad, M. Alobed, M. Alsakhnini, and A. M. Momani, "Challenges in natural Arabic language processing," *Edelweiss Applied Science and Technology*, vol. 8, no. 6, pp. 4700–4705, 2024.