

# Smart Contracts for Automating Corporate Governance and Contractual Compliance

Razakova Rayxan Saylaubekovna

Department of Psychology, Mamun University, Urganch, Uzbekistan

## ABSTRACT

Corporate governance and contractual compliance rest on obligations, such as approvals, disclosures, covenant tests, payments and reporting deadlines, that are specified in documents and fulfilled by people, with the result that obligations are missed, delayed and expensively audited. Smart contracts, programs on a distributed ledger that execute obligations automatically when their conditions are met, promise to remove this gap between specification and execution for obligations that can be codified. This paper develops a framework for deciding which governance and compliance obligations to automate and examines three enforcement regimes in a simulation of 2,000 obligations, of which 75% are codifiable: manual compliance, rule-based workflow automation and smart-contract enforcement with manual fallback for non-codifiable obligations. Smart-contract enforcement reduces missed obligations from 206 to 64, mean delay from 5.0 to 1.3 days and cost per obligation by 45%, and raises audit completeness from 70% to 99%, but introduces 29 rigidity incidents in which automatic execution was inappropriate given circumstances the contract could not observe. Rule-based workflow captures about half the improvement without rigidity. The paper argues that the automation decision turns on codifiability, on the cost of rigidity relative to the cost of omission, and on the availability of trusted inputs, and proposes governance arrangements for the smart contracts themselves.

**Keywords:** Smart contracts; corporate governance; contractual compliance; automation; distributed ledger; rigidity.

*Journal of Teacher Education and Research* (2026). DOI: 10.36268/JTER/21203

## INTRODUCTION

Corporate governance operates through obligations: board approvals above thresholds, disclosure within deadlines, related-party transaction reviews, covenant compliance tests, dividend restrictions, and the reporting that evidences all of these. Contractual compliance operates the same way: payments on milestones, service credits on breaches, rebates on volumes, notices on events. The obligations are specified in constitutions, policies and contracts and fulfilled by people who must remember them, interpret them and act, and the evidence that they were fulfilled is assembled afterward for audit. Obligations are missed, fulfilled late and documented incompletely, at a cost that governance and compliance functions exist to contain [1].

Smart contracts offer to collapse specification and execution. A smart contract encodes an obligation as a program on a distributed ledger that executes when its conditions are satisfied, with the execution itself recorded immutably [2], [3]. A covenant test can run automatically against reported figures, a milestone payment can release when delivery is attested, a disclosure can publish when a threshold is crossed. The obligation cannot be forgotten, its execution is immediate, and its record is complete.

The promise is bounded. Only obligations that can be fully specified in advance and whose conditions can be observed by the ledger are codifiable; the rest require judgement. And automatic execution is rigid: a contract that pays on attested delivery pays even when the attestation is wrong or the circumstances have changed. This paper develops a framework for the automation decision, examines the trade-offs in simulation, and proposes governance for smart contracts in corporate settings.

---

**Corresponding Author:** Razakova Rayxan Saylaubekovna, Department of Psychology, Mamun University, Urganch, Uzbekistan, e-mail: razakova\_rayxan@mamunedu.uz

**How to cite this article:** Saylaubekovna, R.R. (2026). Smart Contracts for Automating Corporate Governance and Contractual Compliance. *Journal of Teacher Education and Research*, 21(2):22-26.

**Source of support:** Nil

**Conflict of interest:** None

---

## BACKGROUND

### Governance and compliance as obligation systems

Agency theory locates governance in the mechanisms that align managers with shareholders, including monitoring and bonding arrangements that generate obligations [1]. Compliance management operationalises these and contractual obligations through policies, controls and evidence. The gap between obligation and fulfilment is a recognised source of governance failure and audit cost.

### Smart contracts

Szabo's original conception of smart contracts [2] was of contractual clauses embedded in systems so that breach is costly or impossible. Blockchain platforms made the conception practical by providing a shared, tamper-evident execution environment [3]. Legal scholarship has examined the relationship between smart contracts and legal contracts and identified incompleteness and rigidity as the central problems: contracts are deliberately incomplete to allow for unforeseen circumstances, and code cannot be [4], [5];

the broader question of when code should substitute for law is examined in [6].

### Oracles and trusted inputs

A smart contract executes on data available to the ledger. Obligations that depend on external facts, such as delivery, financial results or market prices, require oracles that bring those facts on-chain, and the contract's reliability is bounded by the oracle's [3]. Governance obligations often depend on internal facts, such as approvals and financial figures, which must be attested by trusted parties.

## FRAMEWORK FOR THE AUTOMATION DECISION

Obligations are classified on three dimensions. Codifiability: can the obligation's trigger, action and parties be fully specified in advance without recourse to judgement? Observability: can the trigger be established from data the ledger can trust, whether on-chain, oracle-fed or attested by an accountable party? Rigidity cost: what is the harm if the obligation executes automatically in circumstances the specification did not anticipate, relative to the harm of its being missed or delayed under manual fulfilment?

Obligations that are codifiable, observable and low in rigidity cost are candidates for smart-contract enforcement. Codifiable obligations with higher rigidity cost suit rule-based workflow, which automates tracking and prompting while leaving execution to a person. Non-codifiable obligations remain manual, with the ledger used for evidence only. The framework thus produces a three-regime allocation rather than a single choice.

Examples illustrate the allocation. A dividend restriction tied to a leverage covenant is codifiable (the test is arithmetic), observable (the figures are attested in reporting) and low in rigidity cost (a blocked dividend can be released on correction), so it suits smart-contract enforcement. A board approval threshold is codifiable and observable but its rigidity cost is high, since blocking a transaction automatically may harm the firm when the threshold is crossed for benign reasons; workflow that routes the transaction for approval suits it better. A related-party transaction review requires judgement about whether a party is related and whether terms are fair, and is non-codifiable; the ledger can evidence the review but not perform it.

## SIMULATION DESIGN

The harness generates 2,000 obligations with complexity uniform on [0, 1]; obligations with complexity below 0.75 are codifiable (75%). Three regimes are compared. Manual compliance misses 12% of obligations, fulfils with a mean delay of 5 days, costs 1.0 per obligation and achieves 70% audit completeness. Rule-based workflow misses 6%, delays 2 days, costs 0.7 and achieves 85%. Smart-contract enforcement misses 1% of codifiable obligations, executes with 0.1 days' delay at cost 0.4 and 99% audit completeness, but 3% of codifiable executions are rigidity incidents; non-codifiable obligations fall back to manual handling. The parameters are design values reflecting the qualitative properties of each regime; the harness quantifies the resulting trade-offs.

## RESULTS

### Compliance outcomes

Table 1 and Figure 1 report outcomes. Smart-contract enforcement reduces missed obligations from 206 to 64 (–69%), mean delay

from 5.0 to 1.3 days (–74%) and cost per obligation from 1.00 to 0.55 (–45%), and raises audit completeness from 70% to 99%. Rule-based workflow achieves roughly half of each improvement: 97 missed obligations, 2.0 days' delay, cost 0.70 and 85% completeness. The residual misses and delay under smart contracts arise entirely from the 25% of obligations that fall back to manual handling.

### Rigidity

Smart-contract enforcement produces 29 rigidity incidents, cases in which an obligation executed automatically although circumstances the contract could not observe made execution inappropriate, for instance a payment released on an attestation later found erroneous, or a disclosure triggered by a transient data error. The other regimes produce none, because a person reviews before acting. Whether 29 rigidity incidents are an acceptable price for 142 fewer missed obligations depends on their relative cost: where rigidity incidents are reversible and cheap, the trade is favourable; where they are irreversible, as with public disclosures or external payments, it may not be.

### The codifiability boundary

The 25% of obligations that are non-codifiable account for all residual misses and delay under smart-contract enforcement. The marginal value of smart contracts is therefore bounded by codifiability, and attempts to push the boundary by codifying obligations that require judgement would raise rigidity incidents faster than they reduce misses.

### Sensitivity

The results scale with the codifiable share and the rigidity rate. A lower codifiable share, as in governance regimes heavy in discretionary obligations, raises residual misses under smart contracts and narrows their advantage over workflow; a higher rigidity rate, as with poor input assurance, raises rigidity incidents. The ordering of regimes on misses, delay, cost and audit completeness is robust, since it follows from the regimes' qualitative properties; what varies is whether the rigidity cost outweighs the gain, which the framework leaves to the obligation-level assessment.

## GOVERNANCE OF SMART CONTRACTS

### Specification governance

A smart contract is a governance instrument and should be approved as one: its specification reviewed by the owners of the obligation it encodes, its code audited, and its deployment authorised through the same channels as the policy or contract it implements. Changes to the contract are changes to the obligation and require the same authority.

### Input governance

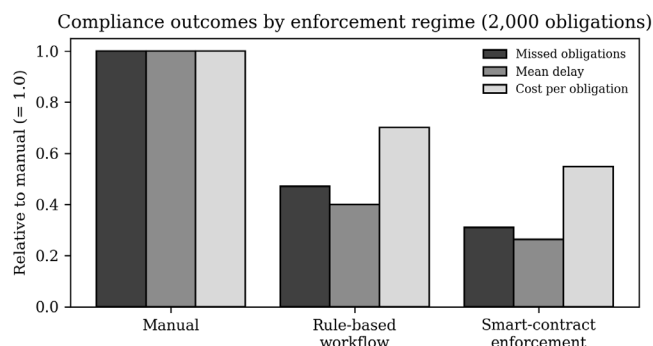
Because execution depends on inputs, the attestation and oracle arrangements that supply them need defined accountability: who may attest, what evidence attestation requires, and how erroneous attestations are corrected. Rigidity incidents in the simulation are largely input failures, and input governance is the principal control.

### Override and reversal

Smart contracts for corporate use should include governed pause and reversal mechanisms, exercised by accountable parties and recorded on the ledger. The mechanisms re-introduce a measure of

**Table 1:** Compliance outcomes by enforcement regime (2,000 simulated obligations; 75% codifiable)

| Regime                     | Missed obligations | Mean delay (days) | Cost per obligation | Audit completeness | Rigidity incidents |
|----------------------------|--------------------|-------------------|---------------------|--------------------|--------------------|
| Manual compliance          | 206                | 5.0               | 1.00                | 70%                | 0                  |
| Rule-based workflow        | 97                 | 2.0               | 0.70                | 85%                | 0                  |
| Smart-contract enforcement | 64                 | 1.3               | 0.55                | 99%                | 29                 |

**Figure 1:** Missed obligations, mean delay and cost per obligation by enforcement regime, relative to manual compliance.

discretion at the cost of some immediacy; the framework's rigidity-cost dimension indicates where they are essential.

### Evidence and audit

The ledger record of executions is the audit evidence, which is why completeness approaches 99%. Auditors should have read access to the ledger and to the contract specifications, and audit procedures should shift from sampling executions to verifying that the contracts implement the obligations correctly.

### IMPLICATIONS

For governance functions, the framework implies a portfolio approach: automate the codifiable, observable, low-rigidity obligations, which in the simulation are the majority; apply workflow to the rest of the codifiable set; and concentrate human judgement on the non-codifiable remainder. For contract design, it implies drafting obligations with codifiability in mind, separating the clauses that can execute automatically from those that require interpretation. For regulators and auditors, it implies accepting ledger records as primary evidence and focusing assurance on specifications and inputs.

### Organisational change

Automating obligations changes the work of governance and compliance staff from fulfilment to specification, input assurance and exception handling. The staff who previously tracked and executed obligations become the designers and reviewers of contracts and the handlers of rigidity incidents. Functions should plan this transition, since the skills differ, and should expect the volume of exception handling to rise initially as contracts encounter circumstances their specifications did not anticipate, before specifications are refined.

### Platform choice

Corporate smart contracts can run on public ledgers, permissioned consortium ledgers or private ledgers. Public ledgers maximise

tamper-evidence but expose obligations to public view and to platform risks the firm does not control; private ledgers offer control but weaker independence of the record. Permissioned ledgers shared with counterparties, auditors and regulators balance the two for most corporate purposes, and the choice should be made on the independence that the audit function and counterparties require.

### Legal standing

Automatic execution does not settle the legal question of what was agreed. Where a smart contract and the governing legal document diverge, as when code executes an outcome the parties did not intend, the legal document usually prevails and the execution must be reversed. Firms should treat the legal document as the authoritative specification, the code as its implementation, and the reconciliation between them as a control, with contract templates designed so that codified clauses map one-to-one onto legal clauses.

### LIMITATIONS

The regime parameters are design values reflecting qualitative properties rather than measured performance; the codifiability threshold and rigidity rate are illustrative. Obligations are independent, whereas governance obligations interact. Legal enforceability, platform risk and the cost of building and auditing contracts are not modelled. All reported values are simulation outputs.

### CONCLUSION

Smart contracts can automate the codifiable majority of governance and compliance obligations, and in simulation they cut missed obligations by two-thirds, delay by three-quarters and cost by nearly half while making audit evidence almost complete, at the price of rigidity incidents that manual and workflow regimes avoid. The automation decision should rest on codifiability, observability and rigidity cost, obligations should be allocated across regimes accordingly, and the smart contracts themselves should be governed as the instruments of governance they are.

### REFERENCES

- Bhaskara Raju Rallabandi. (2024). 6G Cross-Layer Intent Engine: Automated Translation from Business Requirements to Network Policies and Real-Time Scheduling Controls. *International Journal of Data Science and IoT Management System*, 3(1), 44–48. <https://doi.org/10.64751/ijdim.2024.v3.n1.pp44-48>
- Kumar, Ashok, and Arpit Jain. "Image smog restoration using oblique gradient profile prior and energy minimization." *Frontiers of Computer Science* 15, no. 6 (2021): 156706.
- Bhaskara Raju Rallabandi. (2019). Comprehensive Design and Field Performance Assessment of Early 5G NSA Private Networks for Industrial IoT Campuses. *International Journal of Engineering Research and Science & Technology*, 15(1), 22–26. <https://doi.org/10.62643/ijerst.2019.v15.n1.pp22-26>



- Bhanushali, B. (2023). Blockchain for trade finance: Enhancing cross-border payment transparency, reducing processing time, and mitigating fraud through distributed ledger systems. *International Journal on Recent and Innovation Trends in Computing and Communication*, 11(9), 5690–5697. <https://www.ijritcc.org/index.php/ijritcc/article/view/11932>
- Khan, S. (2019). Sales force security compliance: An in-depth study of GDPR, HIPAA, and PCI-DSS enforcement in cloud-based CRM systems and their implications for global enterprises. *International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering*, 8(9), 2211–2219. <https://doi.org/10.15662/IJAREEIE.2019.0809010>
- Ahuja, R. (2023). Implementation of AI-driven input validation in user interfaces for preventing injection attacks and ensuring data sanitation through adaptive filtering and pattern recognition. *International Journal on Recent and Innovation Trends in Computing and Communication*, 11(8), 884–891. <https://www.ijritcc.org/index.php/ijritcc/article/view/11936>
- Rangappa, A. S. (2021). Database security in cloud environments: Exploring multi-tenant risks, data isolation challenges, and compliance mechanisms in hybrid and public clouds. *International Journal on Recent and Innovation Trends in Computing and Communication*, 9(2), 50–57. <https://www.ijritcc.org/index.php/ijritcc/article/view/11921>
- Mahajan, R. A., Dey, R., Khan, M., Mohd Su'ud, M., Alam, M. M., & Jadhav, P. (2025). Enhancing personalization in IoT-based health monitoring via generative AI and transfer learning. *Egyptian Informatics Journal*, 32, Article 100788. <https://doi.org/10.1016/j.eij.2025.100788>
- Dwivedi, D. (2020). Adoption of secure software updates in application lifecycle for preventing supply chain tampering and maintaining trust through cryptographically signed patches. *Journal of Critical Reviews*, 7(10), 6974–6980. <https://doi.org/10.69980/jcr.v7i10.13406>
- Harshitha, G., Kumar, S., Rani, S., & Jain, A. (2021, November). Cotton disease detection based on deep learning techniques. In 4th Smart Cities Symposium (SCS 2021) (Vol. 2021, pp. 496–501). IET.
- Anchala, S. (2023). Integration of confidential computing in multi-cloud environments for enhancing data privacy and regulatory compliance through encrypted processing and isolated execution. *International Journal on Recent and Innovation Trends in Computing and Communication*, 11(9), 5698–5709. <https://www.ijritcc.org/index.php/ijritcc/article/view/11933>
- Gupta, A. (2021). Deployment of advanced endpoint detection in remote work environments for preventing malware intrusions and data exfiltration through behavioral analytics and AI-driven responses. *NeuroQuantology*, 19(9), 1206–1224. <https://doi.org/10.48047/nq.2021.19.9.NQ21199>
- Rangappa, A. S. (2020). Deployment of cross-platform code scanning tools in polyglot DevSecOps projects for uniform security testing and language-agnostic vulnerability coverage. *Journal of Critical Reviews*, 7(11), 5000–5006. <https://doi.org/10.69980/jcr.v7i11.13402>
- Mahadasu, S. S. (2024). Open RAN-based on-demand 6G agricultural networks: Adaptive edge infrastructure for autonomous farming systems. *International Journal of Data Science and IoT Management System*, 3(4), 96–102. <https://doi.org/10.64751/ijdim.2024.v3.n4.1289>
- Bhaskara Raju Rallabandi. (2019). Enhancing 5G NSA Performance with Lightweight Network Data Analytics Loops. *International Journal of Engineering Research and Science & Technology*, 15(4), 57–61. <https://doi.org/10.62643/ijerst.2019.v15.n4.pp57-61>
- Aggarwal, A. (2022). Cloud security in artificial intelligence workloads: Protecting model training, data confidentiality, and AI-generated insights in cloud-based machine learning pipelines. *NeuroQuantology*, 20(3), 1367–1382. <https://doi.org/10.48047/nq.2022.20.3.NQ23004>
- Khan, S. (2022). The use of artificial intelligence in ESG (environmental, social, and governance) investing: A study on AI models for sustainability analytics in asset and wealth management. *International Journal of Multidisciplinary and Scientific Emerging Research*, 10(1), 431–439. <https://doi.org/10.15662/IJMSEERH.2022.1001046>
- Jain, Arpit, Adesh Kumar, and Sanjeev Sharma. "Comparative design and analysis of mesh, torus and ring NoC." *Procedia Computer Science* 48 (2015): 330–337.
- Chatrath, A. (2024). The role of encryption and key management in cloud security: Balancing performance, scalability, and confidentiality in multi-tenant and distributed data storage systems. *International Journal of Innovative Research in Science, Engineering and Technology*, 13(7). <https://doi.org/10.15680/IJRSET.2024.1307187>
- Dwivedi, D. (2022). Application of AI-powered vulnerability scanners in legacy applications for identifying unpatched security flaws and weak dependencies through semantic code analysis. *International Journal on Recent and Innovation Trends in Computing and Communication*, 10(12), 617–623. <https://www.ijritcc.org/index.php/ijritcc/article/view/11918>
- Kumar, S., Jain, A., & Moparthi, N. R. (2022). Enhanced method of object tracing using extended Kalman filter via binary search algorithm.
- Dwivedi, D. (2019). The use of homomorphic encryption in cloud security: Enabling secure data processing, confidential query execution, and privacy-preserving cloud applications. *International Journal of Innovative Research in Science, Engineering and Technology*, 8(4), 4690–4698. <https://doi.org/10.15680/IJRSET.2019.0804136>
- Talasila, D. (2023). Enhancing cloud security with intrusion detection and prevention systems (IDPS): Comparative evaluation of signature-based, anomaly-based, and AI-powered detection models. *International Journal on Recent and Innovation Trends in Computing and Communication*, 11(3), 828–836. <https://www.ijritcc.org/index.php/ijritcc/article/view/11945>
- Mahadasu, S. S. (2025). Distributed AI-RAN inference placement for joint radio optimization and edge workload acceleration. *International Journal of AI Electronics and Nexus Energy*, 1(1), 39–44. <https://doi.org/10.64751/7ep35m50>
- Khan, S. (2022). Sales force security in mobile and remote work environments: Addressing authentication challenges, secure synchronization, and device-level threats. *International Journal of Innovative Research in Computer and Communication Engineering*, 10(12), 8735–8743. <https://doi.org/10.15680/IJIRCE.2022.1012040>
- Jain, Jaishree, Arpit Jain, Saurabh Kumar Srivastava, Chaman Verma, Maria Simona Raboaca, and Zoltán Illés. "Improved security of e-healthcare images using hybridized robust zero-watermarking and hyper-chaotic system along with RSA." *Mathematics* 10, no. 7 (2022): 1071.
- Bhanushali, B. (2024). Tokenization of real-world assets through block chain for improving liquidity, enabling fractional ownership, and reducing transaction costs in global financial markets. *International Journal of Advanced Research in Education and Technology*, 11(4), 1809–1816. <https://doi.org/10.15680/IJARETY.2024.1104071>
- Chatrath, A. (2021). Integration of AI-powered forensic analysis in cloud breach investigations for accelerating root cause identification and evidence collection through log correlation techniques. *International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering*, 10(8). <https://doi.org/10.15662/IJAREEIE.2021.1008051>
- Kumar, S., Sharma, K., Kumar, P. A., Jain, A., Bhagat, S. K., & Singh, P. (2024, September). An improved particle swarm approach for energy-aware location-aided routing in mobile ad-hoc network. In 2024 7th International Conference on Contemporary Computing and Informatics (IC3I) (Vol. 7, pp. 1119–1124). IEEE.
- Mahadasu, S. S. (2025). AI-enhanced service management and orchestration for rApp and xApp lifecycle automation in Open RAN. *Communications on Applied Nonlinear Analysis*, 32(2s), 794–807. <https://doi.org/10.52783/cana.v32.6716>
- Aggarwal, A. (2021). Implementation of machine learning in cloud SIEM solutions for enhancing threat detection accuracy and reducing alert fatigue through correlation-based anomaly detection. *NeuroQuantology*, 19(10). <https://doi.org/10.48047/nq.2021.19.10.NQ21226>
- Ahuja, R. (2022). Application of static application security testing (SAST) in CI/CD pipelines for early detection of insecure coding practices through syntax tree analysis and custom rule sets. *International Journal on Recent and Innovation Trends in Computing and Communication*, 10(2), 80–87. <https://www.ijritcc.org/index.php/ijritcc/article/view/11922>

- Jain, A. (2024, December). Energy-Efficient Signal Processing Architectures for IoT Networks Design. In 2024 International Conference on Emerging Technologies and Innovation for Sustainability (EmergIN) (pp. 362–367). IEEE.
- Devulapalli, N. (2020). The evolution of database security from centralized architectures to distributed cloud-native systems: A comparative and historical perspective. *Turkish Online Journal of Qualitative Inquiry*, 11(4), 2954–2963. <https://doi.org/10.69980/j1064z97>
- Ahuja, R. (2023). The role of artificial intelligence and machine learning in DevSecOps: Leveraging predictive analytics, automated threat detection, and anomaly identification for secure software delivery pipelines. *International Journal on Recent and Innovation Trends in Computing and Communication*, 11(2), 381–389. <https://www.ijritcc.org/index.php/ijritcc/article/view/11946>
- Devulapalli, N. (2020). Implementation of compliance-as-code in regulated DevSecOps environments for automating audit readiness and reducing human error through scripted control validation. *Journal of Critical Reviews*, 7(1), 2651–2657. <https://doi.org/10.69980/jcr.v7:i1.13405>
- Anchala, S. (2020). Cloud security challenges in supply chain systems: Addressing vendor risks, data integrity, and third-party service provider vulnerabilities. *Journal of Critical Reviews*, 7(2). <https://doi.org/10.69980/jcr.v7:i2.13401>
- Aggarwal, A. (2021). Utilization of continuous vulnerability scanning in DevSecOps pipelines for preemptive threat detection and secure code deployment through automated security checks. *NeuroQuantology*, 19(1), 427–442. <https://doi.org/10.48047/nq.2021.19.1.NQ21055>
- Devulapalli, N. (2019). Post-quantum cryptography for database security: Preparing for the impact of quantum computing on encrypted data storage and secure queries. *Journal of Critical Reviews*, 6(1), 614–620. <https://doi.org/10.69980/jcr.v6:i1.13396>
- Kaur, M., AlZubi, A. A., Jain, A., Singh, D., Yadav, V., & Alkhayyat, A. (2023). DSCNet: deep skip connections-based dense network for ALL diagnosis using peripheral blood smear images. *Diagnostics*, 13(17), 2752.
- Talasila, D. (2020). Deployment of dynamic data masking in cloud-based applications for protecting sensitive information and enhancing privacy controls through contextual access filtering. *Journal of Critical Reviews*, 7(9), 4056–4062. <https://doi.org/10.69980/jcr.v7:i9.13403>
- Anchala, S. (2023). Utilization of blockchain technology in cloud storage systems for enhancing data integrity and preventing unauthorized modifications through decentralized validation. *International Journal on Recent and Innovation Trends in Computing and Communication*, 11(5), 629–637. <https://www.ijritcc.org/index.php/ijritcc/article/view/11942>
- Rangappa, A. S. (2020). Integration of behavioral analytics in developer workflows for detecting insider threats and malicious activities through activity monitoring and contextual profiling. *International Journal of Innovative Research in Science, Engineering and Technology*, 9(1), 13375–13383. <https://doi.org/10.15680/IJRSET.2020.0901064>
- Gupta, A. (2021). Application of cyber deception strategies in military networks for enhancing operational security and misleading attackers through adaptive honeypots and decoy systems. *NeuroQuantology*, 19(4), 317–333. <https://doi.org/10.48047/nq.2021.19.4.NQ21068>
- Mahadasu, S. S. (2024). Real-time network digital twin synchronization for predictive capacity planning in multi-vendor private 5G systems. *International Journal of Engineering Research and Science & Technology*, 20(4), 483–488. <https://doi.org/10.62643/ijerst.2024.v20.n4.4405>
- Jain, J., & Jain, A. (2022). Securing e-Healthcare Images Using an Efficient Image Encryption Model. *Scientific Programming*, 2022(1), 6438331.
- Chatrath, A. (2021). Implementation of AI in recruitment platforms for improving candidate screening and diversity inclusion through bias-free resume parsing and skill matching. *International Journal of Innovative Research in Computer and Communication Engineering*, 9(12). <https://doi.org/10.15680/IJIRCC.2021.0912046>
- Raval, M., Pandit, T., & Upadhyay, D. (2026). LLM is not all you need: A systematic evaluation of ML vs. foundation models for text and image based medical classification (arXiv:2601.16549). [arXiv:2601.16549](https://arxiv.org/abs/2601.16549)
- Gupta, A. (2021). Implementation of cyber threat hunting in hybrid cloud environments for early breach detection and threat containment through behavior-based analysis tools. *NeuroQuantology*, 19(12), 1312–1329. <https://doi.org/10.48047/nq.2021.19.12.NQ21342>
- Bhaskara Raju Rallabandi. (2023). AGENTIC-AI xAPPS: O-RAN MULTI-AGENT CONTROL FOR ENTERPRISE POLICY, SCHEDULING & SPECTRUM. *International Journal of Data Science and IoT Management System*, 2(4), 54–58. <https://doi.org/10.64751/ijdim.2023.v2.n4.pp54-58>
- Talasila, D. (2023). Implementation of cloud-based disaster recovery models for minimizing business downtime and ensuring operational continuity through geographically distributed backup systems. *International Journal on Recent and Innovation Trends in Computing and Communication*, 11(8), 867–874. <https://www.ijritcc.org/index.php/ijritcc/article/view/11934>
- Bhanushali, B. (2022). Blockchain-driven carbon credit trading systems for improving environmental transparency, reducing double counting, and ensuring accountability in climate finance. *International Journal on Recent and Innovation Trends in Computing and Communication*, 10(11), 292–300. <https://www.ijritcc.org/index.php/ijritcc/article/view/11927>
- Jain, Arpit, Nageswara Rao Moparthy, A. Swathi, Yogesh Kumar Sharma, Nitin Mittal, Ahmed Alhussen, Zamil S. Alzamil, and MohdAnul Haq. "Deep Learning-Based Mask Identification System Using ResNet Transfer Learning Architecture." *Computer Systems Science & Engineering* 48, no. 2 (2024): 341.

