Smart Land Registration in BD: A Blockchain-Based Architecture for Decentralized Land Governance and Secure Registration

Thesis Final_Defense – Fall 2025

30 January 2026

Presented By,

Khondokar Saim – 221902353

Hasnat Zamil Sayad– 221902001

Ibrahim Al Rifat– 221902208

Supervisor's-

Ms. Farhana Akter Sunny
Assistant Professor, Dept of CSE at GUB

Mr. Mozdaher Abdul Quader
Lecturer, Dept of CSE at GUB

Department of Computer Science and Engineering
Green University of Bangladesh

Table of Contents

Introduction

Background

- Georgia, Sweden, and the Netherlands have already adopted blockchain

- Still face corruption, delays, and data security risks

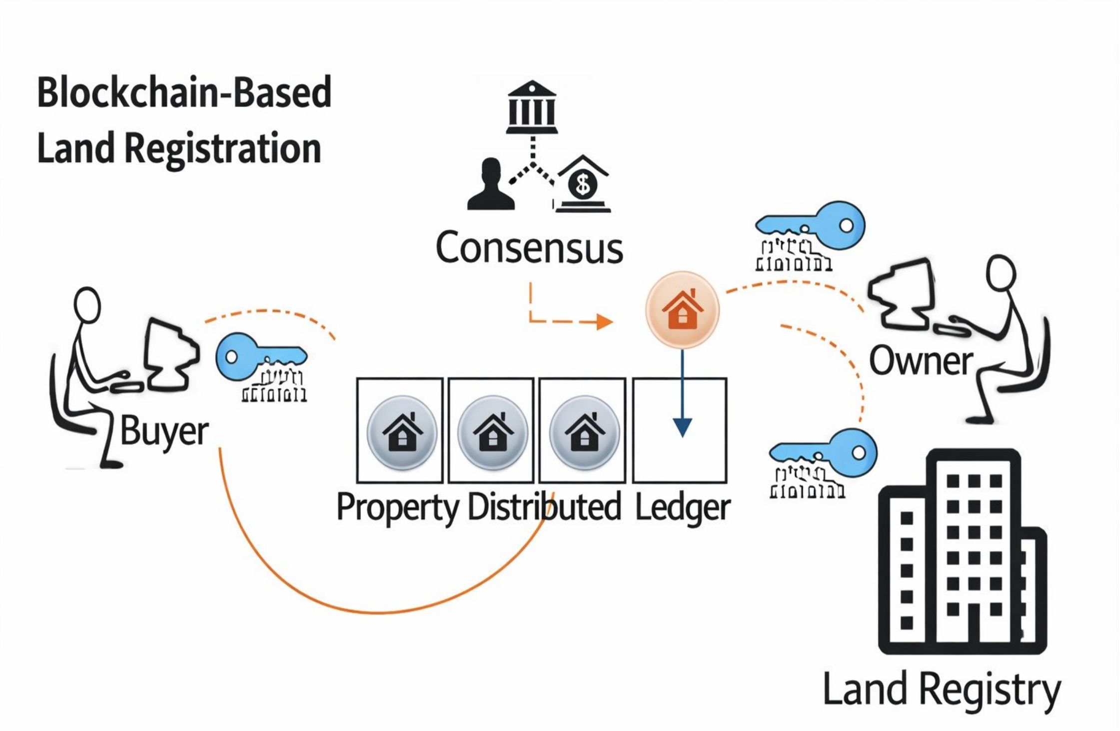


Figure 1.0 : *Blockchain-Based Land Registration Workflow*

Problem Statement (1/3)

- Land services face fake documents and long delays.
- Different government offices keep separate land records, which creates confusion.
- Central online systems are also risky, such as the 2023 hacking of 50 million citizen records.
- Normal blockchain systems like PoW and DPoS are too costly and slow for government use.

Problem Statement (2/3)

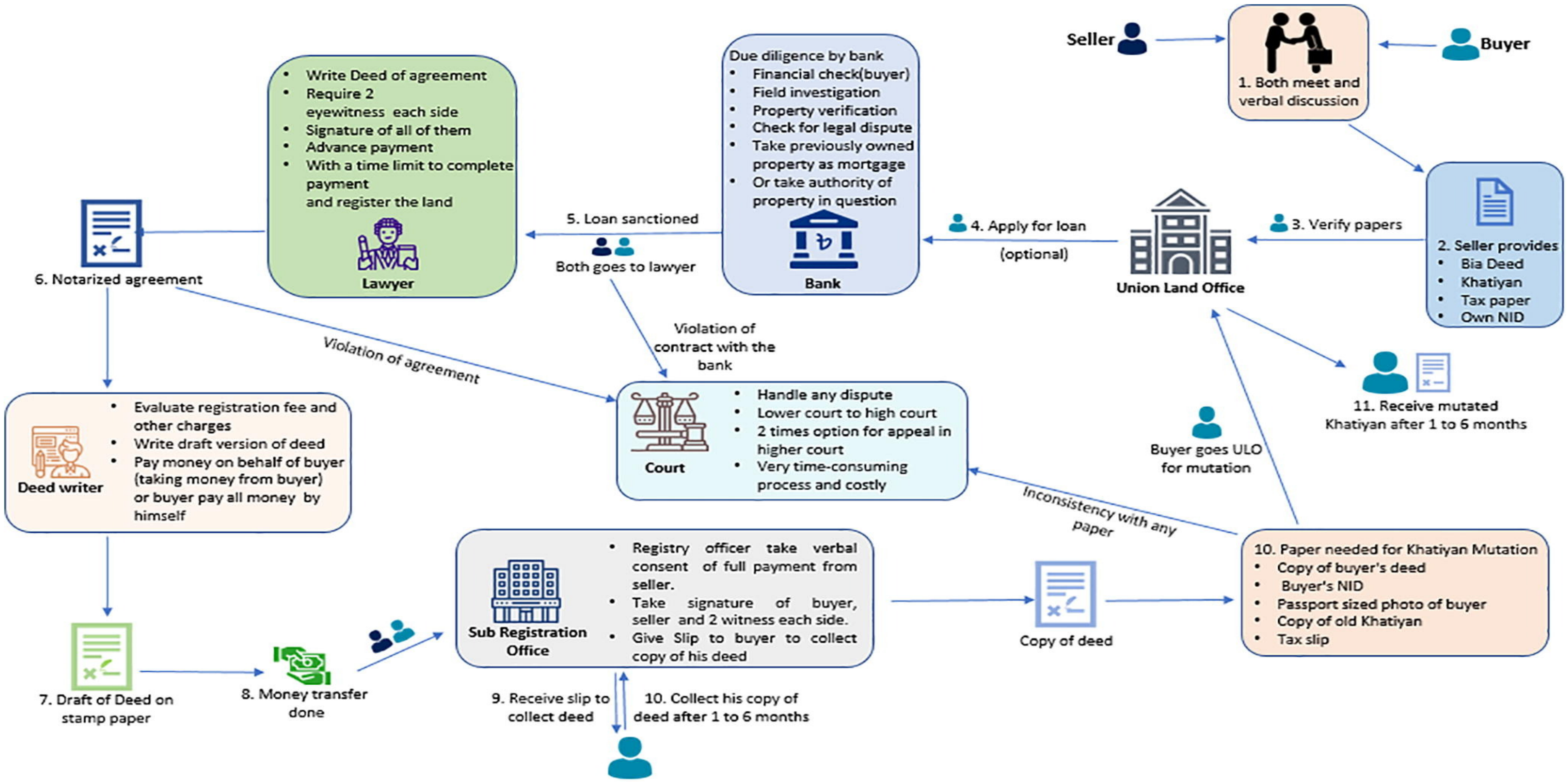


Figure 2.0 : *Existing land title management system of Bangladesh*

Problem Statement (3/3)

- The process is highly manual and fragmented, involving many intermediaries (lawyer, deed writer, bank, sub-registry, court, land office), which causes long delays, repeated verification, and frequent human errors.

- Because it mostly depends on paper documents and verbal checks, records can be easily changed or faked and lengthy court cases.

- There is no single trusted system, so mismatched records often lead to disputes and long delays, resulting in disputes and 1-6 month delays even after registration.

Motivation

- There is a strong need for a secure, scalable, and tamper-resistant land registration framework.
- Large projects like Purbachal New Town reveal that current systems cannot securely handle high transaction volumes.
- Real-world corruption cases and unofficial payments in land services highlighted systemic weaknesses.
- Major data breaches exposed the risks of centralized land record databases.
- Successful blockchain-based land registry initiatives in other countries showed practical benefits.

Objectives

- To develop an Property Transaction Process via Digital Portal to handle stamp obtaining and stamp duty payment digitally.
- To develop an Digital Verification Protocols for Title Transfers to manage ownership transfer transactions between buyers and sellers.
- To design a Network Structure for the System to support structured and transparent transaction flow.
- To get Gas cost Analysis for different transaction operations in Land Registration.

Literature Review

Related Studies (1/6)

Title: A Blockchain-based Land Title Management System for Bangladesh

Methodology:

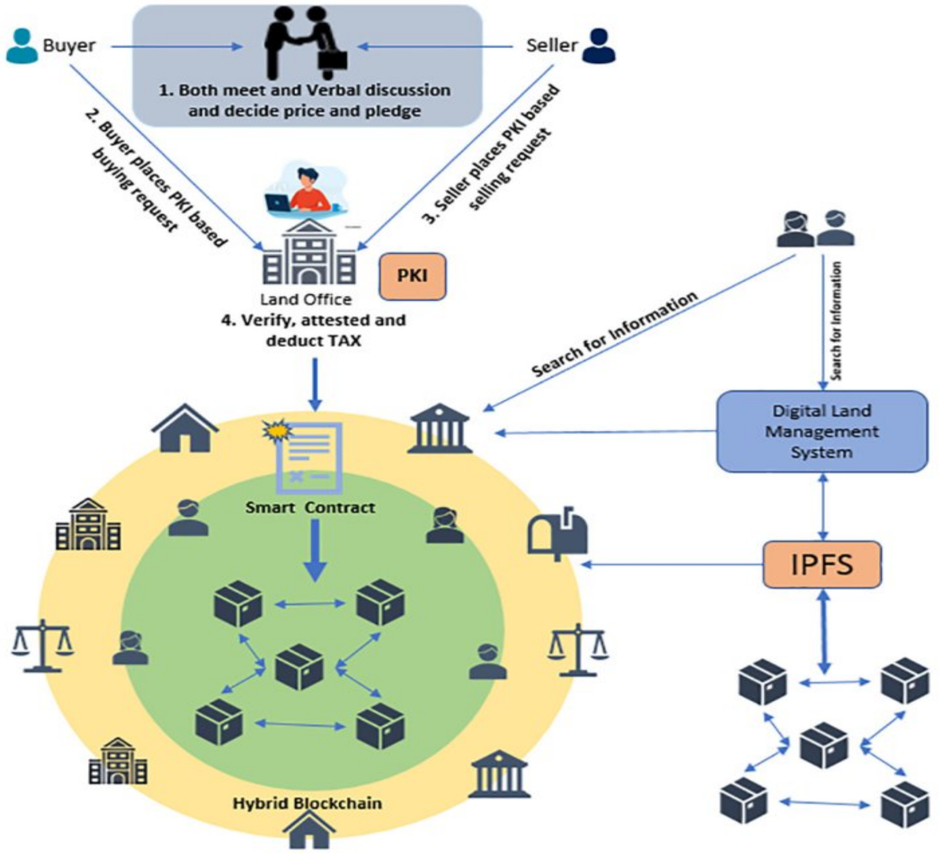


Figure 3.0 : *Proposed incremental Blockchain adoption model for land title management of Bangladesh.*

Related Studies (2/6)

- **Achievement:**

- Introduced phase wise Public to Hybrid blockchain roadmap.
- Implemented Ethereum smart contract prototype reducing cost and delays.

Limitations:

- Needs strong government laws and institutional adoption.
- Public blockchain mining not scalable for all citizens.
- Gas price volatility makes real-world cost uncertain.

Reference:

K. M. Alam, J. M. A. Rahman, A. Tasnim and A. Akther,"Blockchain-Based Land Title Management System for Bangladesh," Journal of King Saud University – Computer and Information Sciences, 2020, doi: 10.1016/j.jksuci.2020.10.011.

Related Studies (3/6)

Title: Distributed Ledger Technology based Land Transaction System with Trusted Nodes Consensus Mechanism

Methodology:

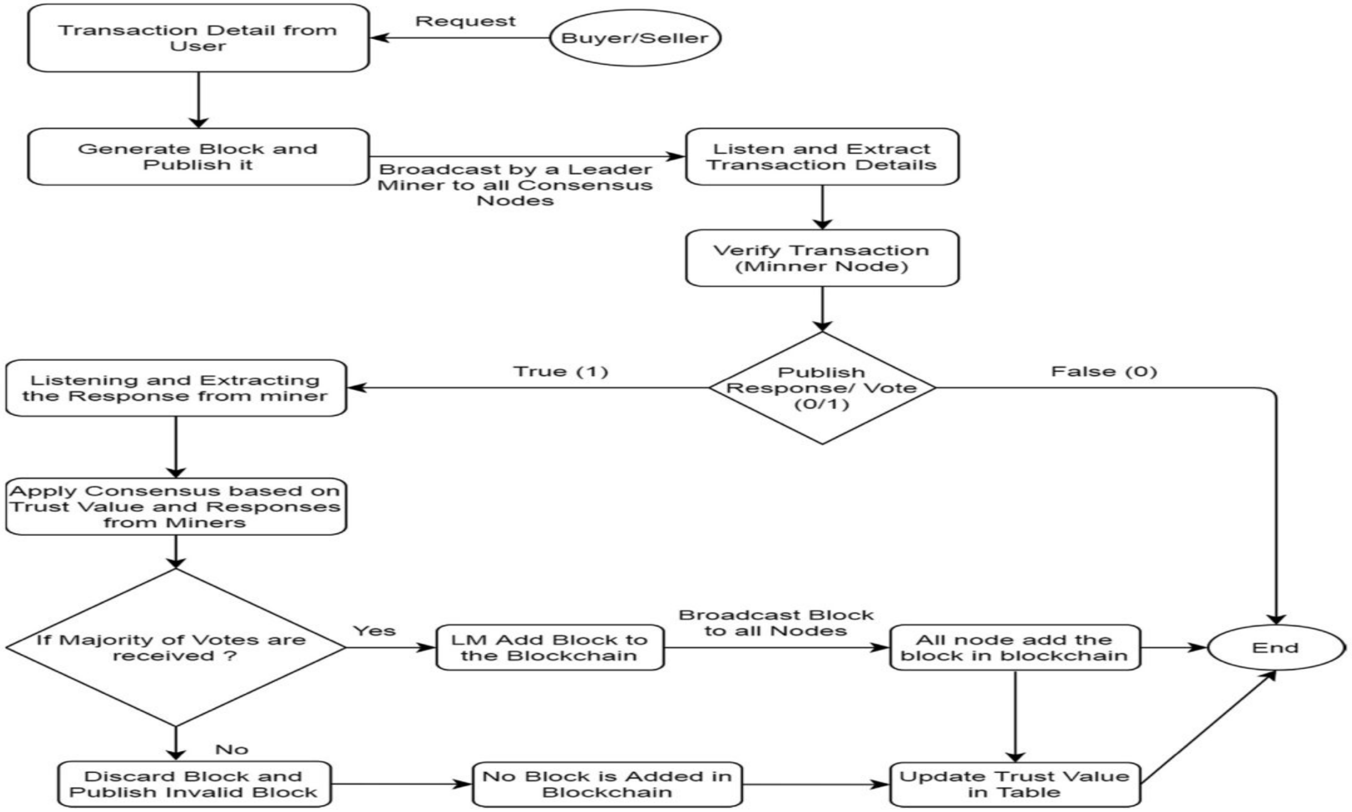


Figure 4.0 : *Work Flow of the Proposed System*

Related Studies (4/6)

Achievement:

- Introduced Trusted Node Consensus Algorithm (TNCA) using miner trust values
- Reduced message overhead by 58.94% and improved block addition time by 26.44% vs PoW

Limitations:

- Trust value table maintenance increases computation cost in large networks.
- Framework assumes government-controlled validator participation.
- Consensus accuracy depends strongly on correct trust evaluation.

Reference:

A. S. Yadav, S. Agrawal and D. S. Kushwaha,"Distributed Ledger Technology Based Land Transaction System With Trusted Nodes Consensus Mechanism," Journal of King Saud University – Computer and Information Sciences, 2021, doi: 10.1016/j.jksuci.2021.02.002.

Related Studies (5/6)

Title: A Blockchain Based Land Registration and Ownership Management System for Bangladesh

Methodology:

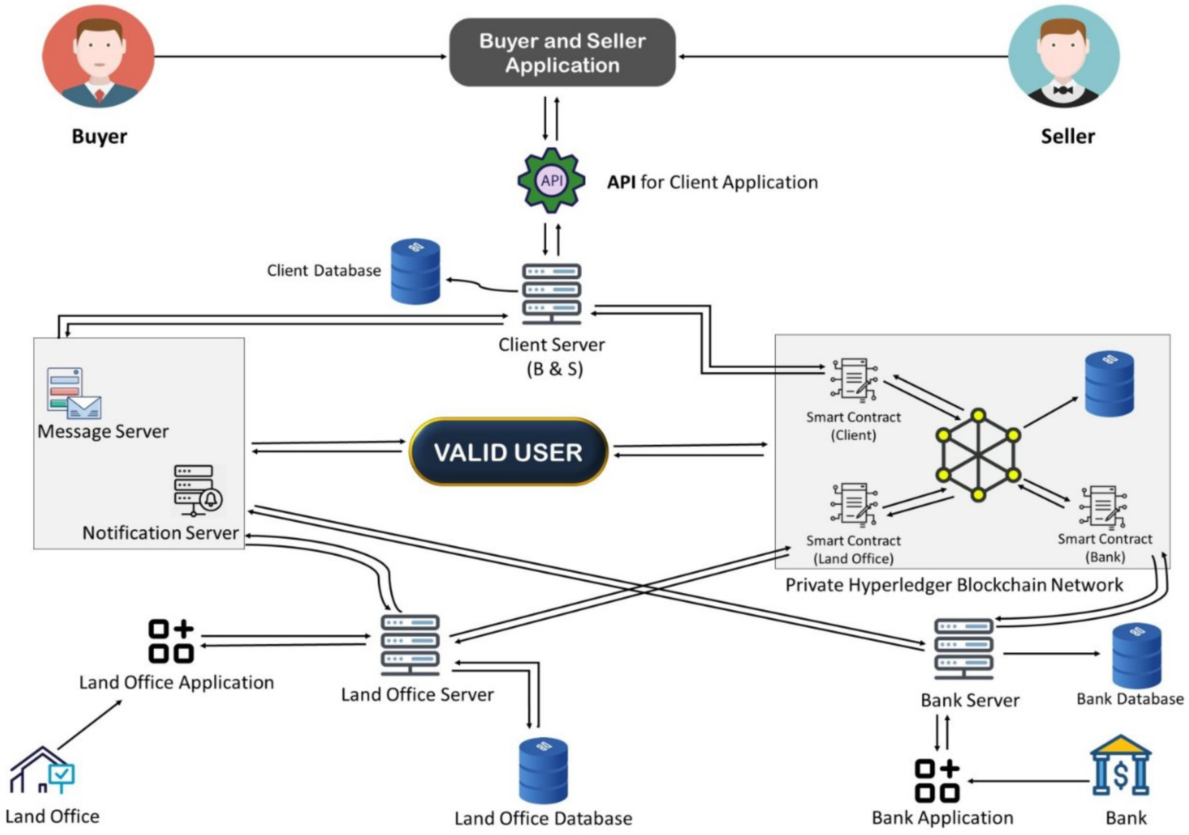


Figure 5.0 : *System architecture of the proposed system.*

Related Studies (6/6)

Achievement:

- Proposed a permissioned Hyperledger Fabric-based land ownership system.
- Improved transparency and removed intermediaries to reduce fraud.

Limitations:

- No consensus mechanism defined for buyer–seller transaction completion.
- Assumes full participation of banks and government stakeholders.

Reference:

R. I. Shithy, N. Mohammad, H. N. A. Ruhullah, S. M. Y. Oni and Md. Al-Amin, "A Blockchain Based Land Registration and Ownership Management System for Bangladesh," in Proceedings of the 2021 4th International Conference on Blockchain Technology and Applications (ICBTA 2021), Xi'an, China, Dec. 2021, pp. 1–7, doi: 10.1145/3510487.3510501.

Research Gap

Table 1.0 : *Research Gap from Existing Systems..*

Gap	What Existing Systems Lack	Why Our work is needed
1	Large-scale testing with realistic networks	Our work evaluates performance up to 1000 validator nodes, matching district-level deployment in Bangladesh
2	Cross-consensus comparison across families	our compares PoW, DPoS, RLSCA, TNCA to justify a government-suitable consensus choice
3	Clear governance, onboarding, and legal integration	our work aligns validation with administrative jurisdictions and supports existing land laws
4	Detailed gas consumption and transaction cost benchmarking for core land registration operations	our work provides operation-wise gas profiling (deployment, registration, mutation, identity setup), ensuring economic scalability and practical feasibility for Bangladesh
5	Privacy-safe decentralized architecture	our work combines permissioned blockchain + IPFS, reducing data leakage and corruption risks

Methodology

Block Diagram of Research Methodology

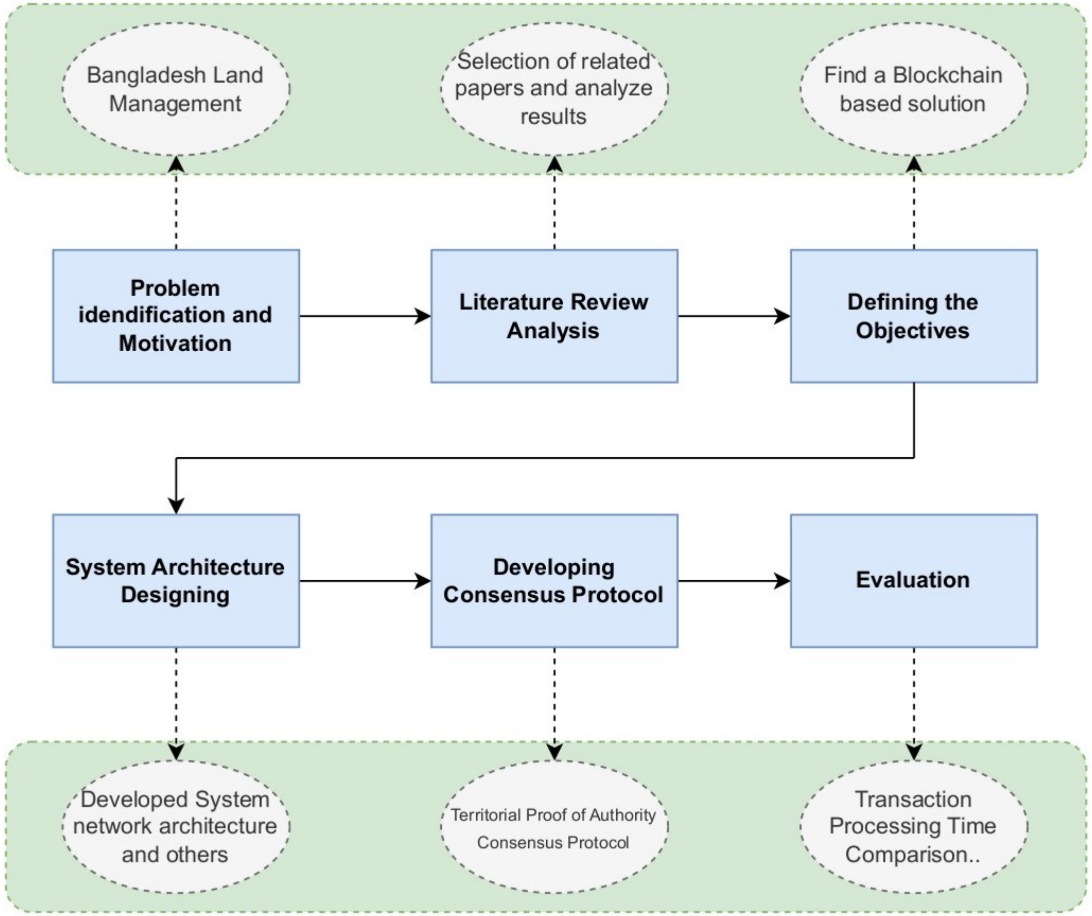


Figure 6.0 : *Block Diagram of Research Methodology Overview.*

Network Structure for the System (1/3)

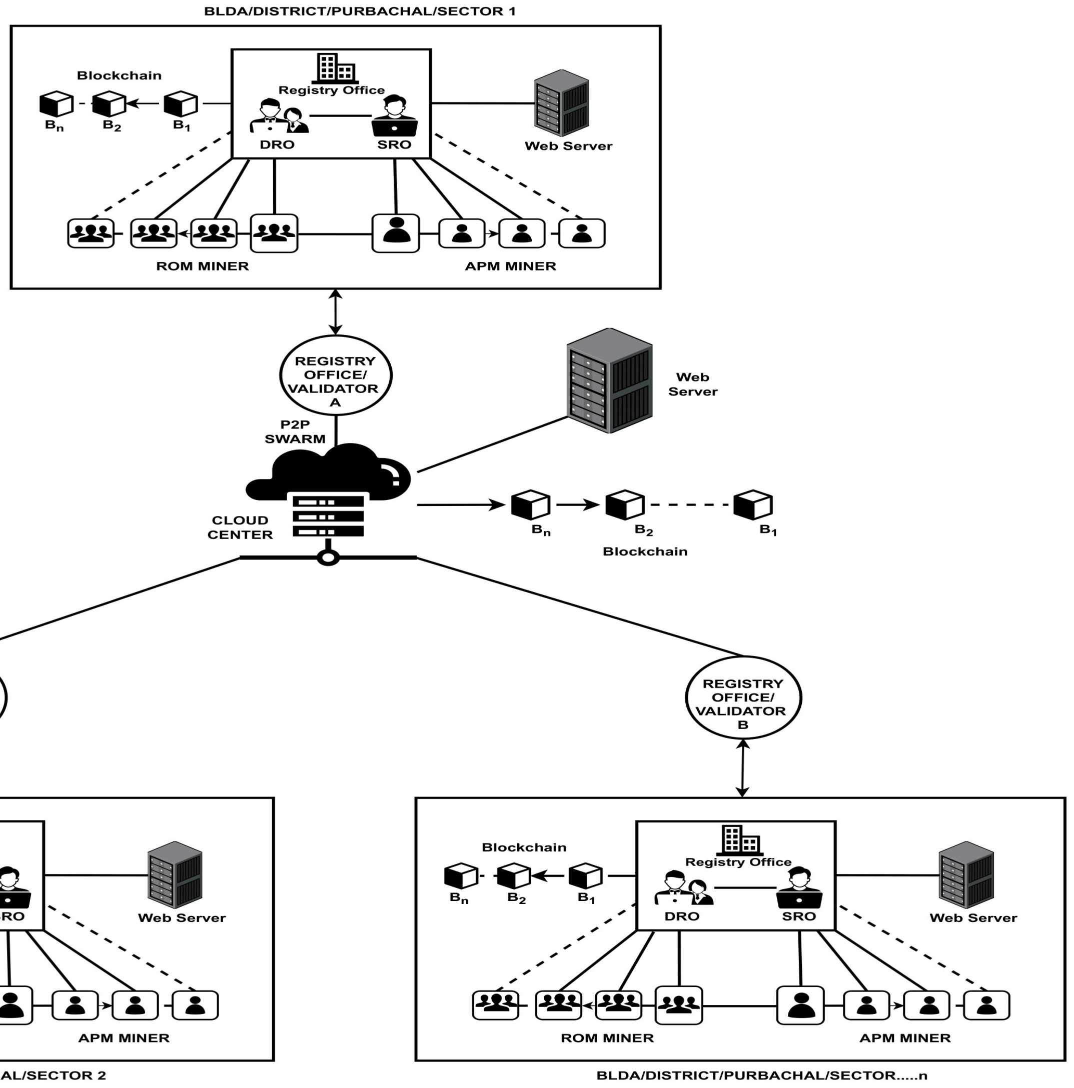


Figure 7.0 : *Network Structure for the System.*

Property Transaction Process via Digital Portal (2/3)

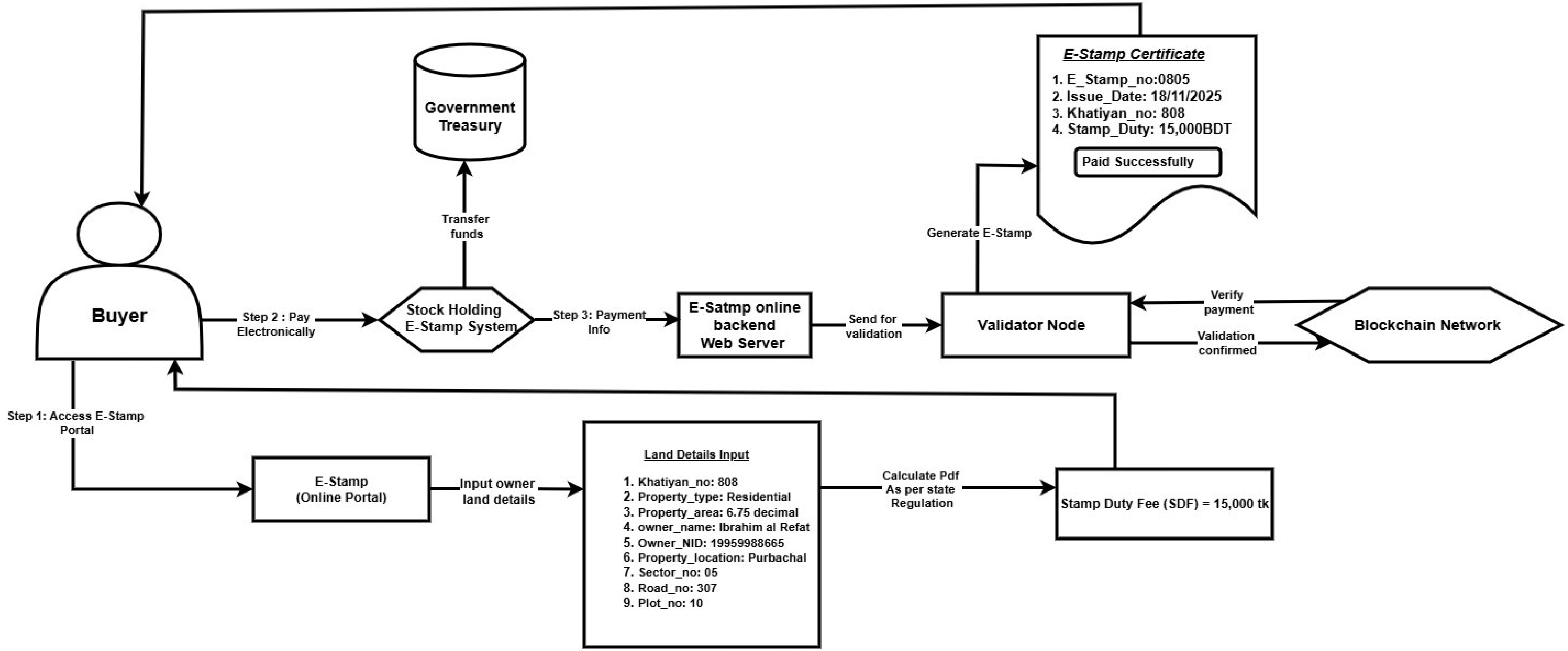


Figure 7.0 : *Property Transaction Process via Digital Portal.*

Digital Verification Protocols for Title Transfers (3/3)

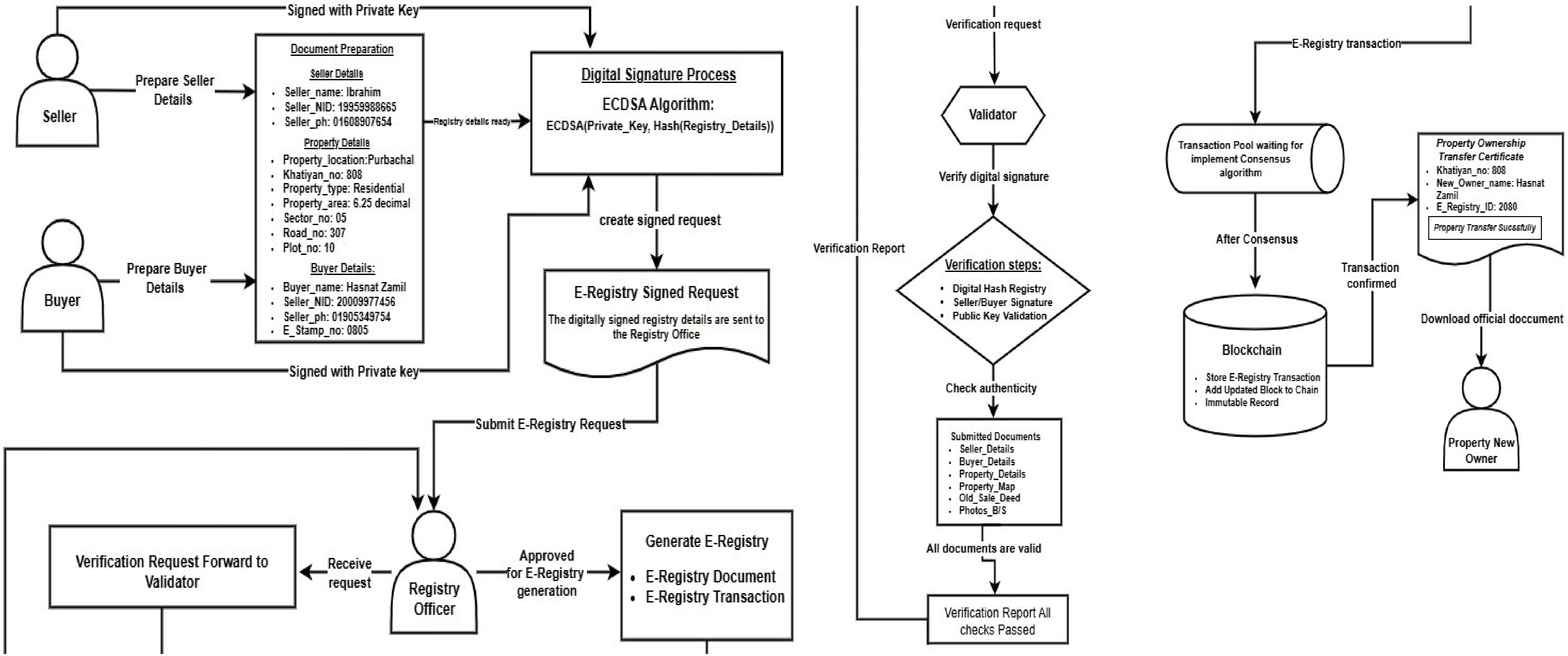


Figure 8.0 : *Digital Verification Protocols for Title Transfers.*

Data Collection Methods (1/2)

Territorial Message Exchange Reduction & Transaction Processing Time Data Collection Method.

Table 2.0 : Territorial Message Exchange Reduction & Transaction Processing Time Data Collection Method.

Aspect	Description
Data Sources	Realtime simulation-generated blockchain event logs, validator performance data, transaction timestamps, and network message traces; baseline data for PoW and other consensus protocols.
Tools	Python-based simulation in Google Colab; structured logging using CSV/JSON; data handling with Pandas.
Sampling	Stratified sampling across 64 districts with network sizes ranging from 10–1000 nodes and multiple transaction runs per configuration.
Validity & Reliability	Ensured through event-driven data capture, controlled simulation settings, repeated runs, and immutable audit trails.

Data Collection Methods (2/2)

Gas Consumption and Cost Data Collection Method.

Table 3.0 : Gas Consumption and Cost Data Collection Method..

Rubric Item	Content
Sources of Data	**Primary:** Transaction receipts (gasUsed, txHash, status) from on-chain executions **Secondary:** IPFS land records (metadata + hashes) for realistic scenarios
Tools / Instruments	• Web3 simulation interface + IPFS connector • Remix Ethereum IDE (Solidity execution) • MetaMask (signing) • Ganache (local chain + dummy ETH)
Sampling Technique & Size	• Purposive sampling from **100 IPFS land records** • **50** for Initial Khatian Registration • **50** for Ownership Mutation (Transfer)
What Was Measured	• Smart Contract Deployment • Initial Khatian Registration • Khatian Mutation Transfer • User Identity Registration

Data Analysis Methods (1/2)

Territorial Message Exchange Reduction & Transaction Processing Time Data Anaysis Method.

Table 4.0 : Territorial Message Exchange Reduction & Transaction Processing Time Data Anaysis Method...

Aspect	Description
Techniques	Statistical and model-based analysis comparing TPACP with PoW and other consensus protocols.
Processing	Event logs are aggregated per transaction and network size to compute message exchange and processing time metrics.
Software	Python (Google Colab) using Pandas, NumPy, and Matplotlib.
Research Alignment	Analysis evaluates scalability (message reduction), efficiency (processing time), and integrity (verification outcomes).

Data Analysis Methods (2/2)

Gas Consumption and Cost Data Analysis Method

Table 5.0 : Gas Consumption and Cost Data Analysis Method

Rubric Item	Content
Techniques Used	Descriptive analysis of gasUsed per operation (representative/average values) .Cost modelling: Gas Cost = gasUsed × gasPrice
How Data Were Processed	• Execute each function in Remix • Extract receipt data (gasUsed, status, txHash) from Web3/Ganache/Remix • Compute cost using benchmark 20 Gwei
Software / Platform	• Remix + Solidity • MetaMask + Ganache • Web3 interface
Aligned with Research Questions	• Evaluates economic feasibility (transaction cost) • Evaluates operational efficiency (gas consumption) for core land registry operations

Results and Discussion

Key Finding (1/5)

Simulation Environment:

Table 6.0 : Simulation Environment:

Main Category	Sub Category	Description
Simulation Setup	Network Model	District-based blockchain network reflecting Bangladesh’s land registry system
	Validator Structure	Validators represent Sub-Registry and District Registry offices
	Network Scale	Network size evaluated from 10 to 1000 nodes for scalability testing
Technical Environment	Operating System	Ubuntu 22.04.5 LTS
	Distributed Storage	InterPlanetary File System (IPFS)
	Programming Tools	Python 3, Solidity (Smart Contract Development)
	Consensus Baselines	PoW, DPoS, RLSCA, TNCA + Gas Consumption Analysis
	Performance Metrics	Message exchange volume, transaction processing time, and gas consumption cost for land operations
	Experimental Stability	Multiple iterations with statistical averaging for reliable results

Key Finding (2/5)

Performance Metrics:

Table 7.0 : Performance Metrics

Territorial Communication Efficiency	
What it Measures	Message exchange overhead during the consensus process
Why it Matters	Lower message overhead improves scalability, especially in poor or unstable network conditions
Transaction Processing Time	
What it Measures	Time taken from property submission to final immutable blockchain record
Why it Matters	Faster processing reduces administrative delays and limits opportunities for corruption
Gas Consumption and Cost Analysis	
What it Measure	Gas units required by smart contract operations such as deployment, khatian registration, ownership mutation, and identity setup
Why it Matters	It shows how expensive blockchain transactions are and whether the system is affordable to use at scale in Bangladesh

Key Finding (3/5)

Visual Result – Message Exchange Reduction

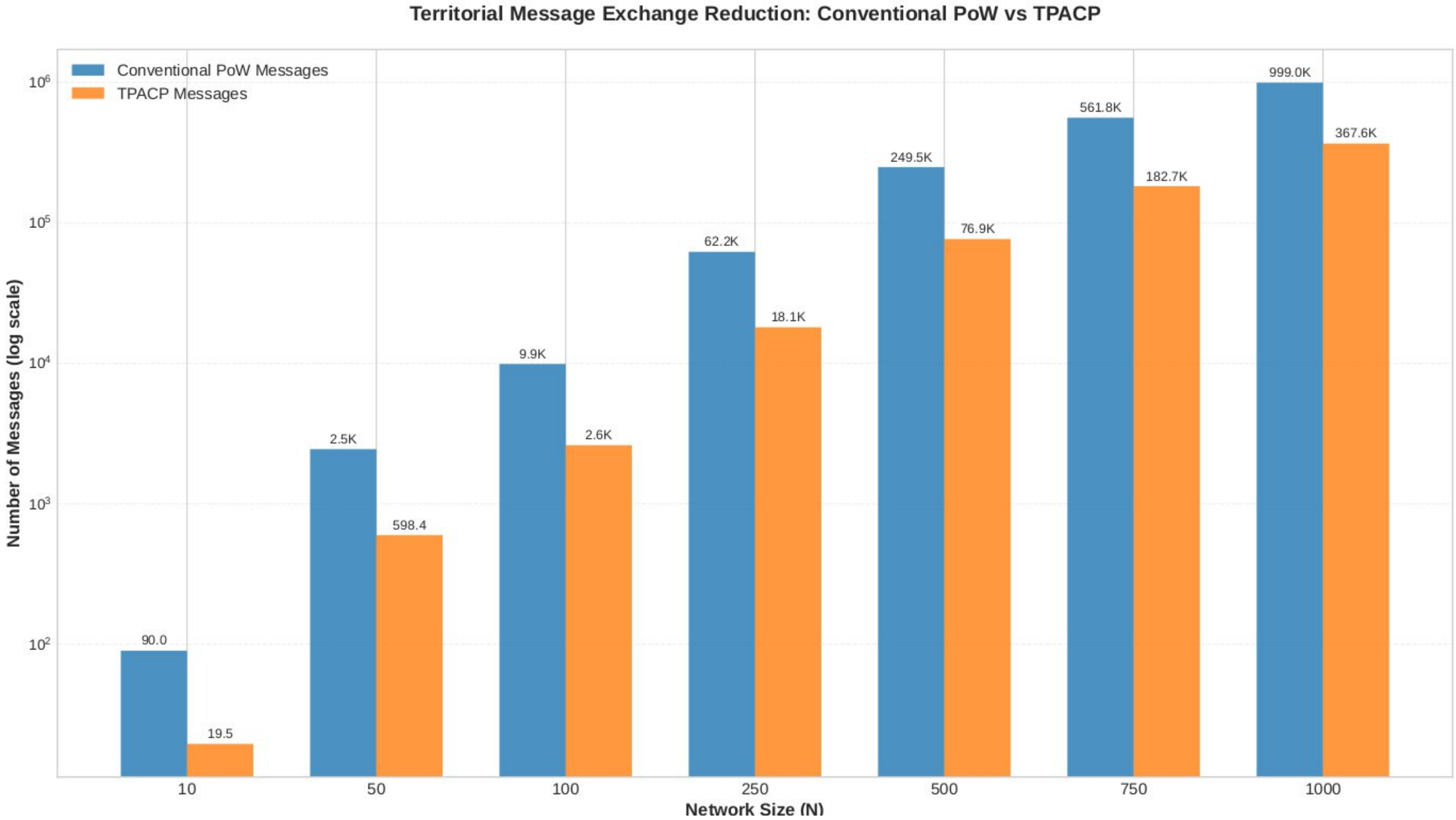


Figure 9.0 : *Territorial Messeage Exchange Reduction: Conventional PoW vs TPACP.*

Key Finding (4/5)

Visual Result – Transaction Processing Time

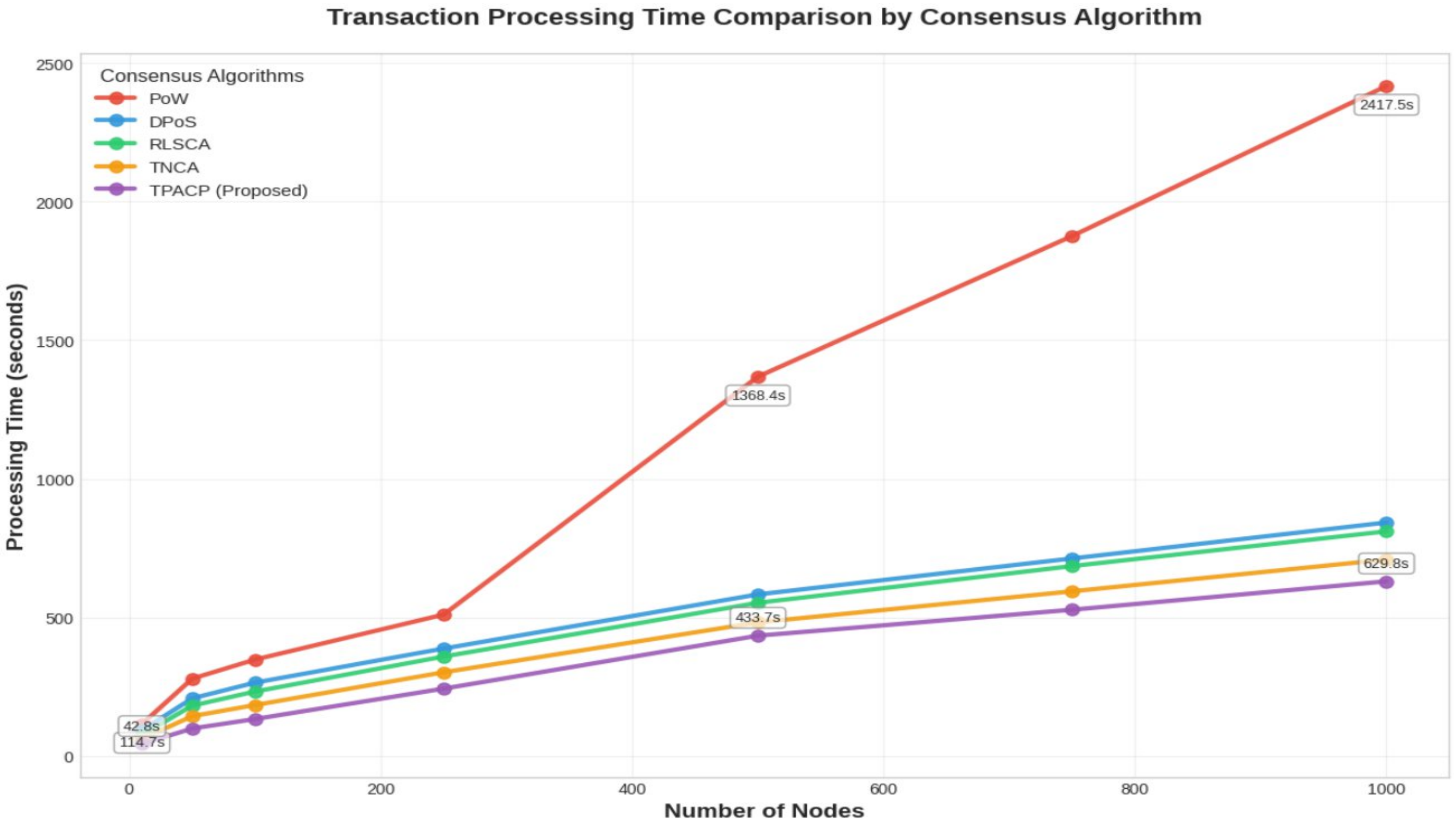


Figure 10.0 : *Transaction Processing Time Comparison by Consensus Algorithm.*

Key Finding (5/6)

Visual Result – Gas Consumption and Cost for Different Transaction Operations

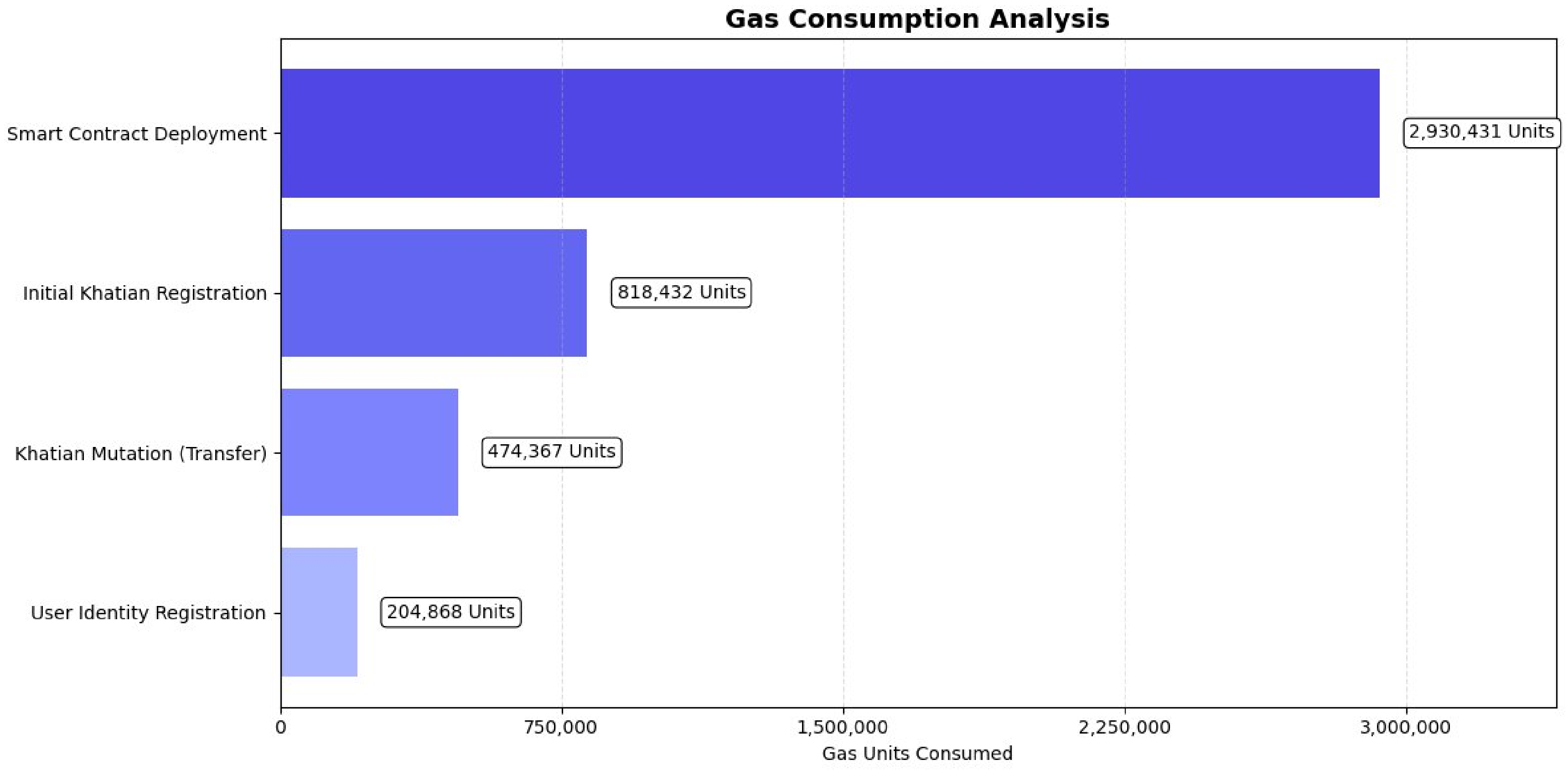


Figure 11.0 : *Gas Consumpton Analysis for Different Transaction Operations.*

Key Finding (6/6)

Core outcomes of the Visual Result

- Average 56.8% reduction in message overhead vs PoW
- Up to 74% faster transaction processing at large scale
- Stable performance from 10 to 1000 nodes
- Overall, routine transactions such as registration (818,432 g.units),mutation (474,367 g.units), and identity setup (204,868 g.units) consume far less gas than deployment (2,930,431 g.units) and resulting the system is economically scalable for Bangladesh land regestration scenario.

Comparison

Table 8.0 : *Comparison with Existing Related Studies with proposed TPACP*

Aspect	**Existing Related Studies**	**Limitation in Prior Works**	**Contribution of Proposed TPACP**
Land Registry Blockchain Models	Shithy et al. proposed blockchain-based land registration frameworks	Mostly conceptual; lacks full-scale economic and governance evaluation	TPACP introduces territorially governed blockchain suitable for Bangladesh administration
Transaction Processing Time	A. S. Yadav et al., Processing times reported for PoW, PoS, DPoS, and TNCA (e.g., at 100 nodes: PoW 358.2 s, DPoS 252.4 s, TNCA 153.6 s)	Evaluated only up to 350 nodes and without advanced authority-based optimization	TPACP achieves lower times at larger scale (e.g., 132.9 s at 100 nodes, 629.0 s at 1,000 nodes), demonstrating scalable and faster processing
Gas Consumption Benchmarking	Alam et al., Final Avg. gas costs reported (e.g., deployment 3,084,664, khatian entry 442,408, mutation 499,334, user 215,650)	Values reported without unified workflow or benchmark consistency	TPACP provides standardized gas benchmarks (deployment 2,930,431, registration 818,432, mutation 474,367, identity 204,868) enabling direct, reproducible comparison

Performance Analysis (1/3)

Table 9.0 : *Territorial Message Exchange Reduction Compared to Proof of Work (PoW)*

Network Size (N)	Conventional PoW Messages	TPACP Messages	Reduction (%)
10	90	19.5	78.3
50	2,450	598.4	75.6
100	9,900	2,613.6	73.6
250	62,250	18,052.5	71.0
500	249,500	76,855.0	69.2
750	561,750	182,733.8	67.5
1000	999,000	367,632.0	63.2

- **10 nodes:** TPACP reduces message overhead by **78.3%** (highest efficiency).
- **500 nodes:** Still achieves **69.2%** reduction, showing good scalability
- **1000 nodes:** Maintains **63.2%** reduction, proving robustness for large nationwide networks

Performance Analysis (1/3)

Table 10.0 : *Transaction Processing Time Comparison (Seconds)*

Node Count	PoW	DPoS	RLSCA	TNCA	TPACP
10	114.7	92.3	75.4	61.2	42.8
50	278.6	207.4	181.3	143.2	98.7
100	347.5	264.3	232.1	183.4	132.9
250	509.8	387.2	358.4	301.5	242.6
500	1,368.4	582.6	552.3	481.7	433.7
750	1,876.2	712.4	684.9	593.8	527.3
1,000	2,417.5	842.1	810.3	708.4	629.0

At 1000 nodes, TPACP also outperforms other protocols:

- **25.2% better than TPoS**
- **22.3% better than RLSCA**
- **11.1% better than TNCA**

Performance Analysis (1/3)

Table 11.0 : *Gas Consumption and Cost for Different Transaction Operations*

Operation (Transaction Type)	Gas Units	Gas cost (20 Gwei per Unit)
Smart Contract Deployment	2,930,431	58,608,620
Initial Khatian Registration	818,432	16,368,640
Khatian Mutation (Transfer)	474,367	9,487,340
User Identity Registration	204,868	4,097,360

- Smart contract deployment has the highest gas cost (2,930,431 gas), reflecting the main one-time setup expense.
- Routine land operations like khatian registration (818,432 gas) and mutation/transfer (474,367 gas) consume much less gas, ensuring practicality.
- User identity registration requires the lowest gas (204,868 gas), showing efficient authentication processing.

Conclusion

Contributions

- **TPACP Consensus Methodology**
  Introduces a territorial, performance-weighted consensus protocol aligned with district-based land administration.

- **Performance-Driven Evaluation Framework**
  Assesses scalability using message exchange volume and transaction processing time across 10–1000 nodes, benchmarked against PoW, DPoS, RLSCA, and TNCA.

- **Gas Consumption & Cost Analysis**
  Provides numerical gas usage and transaction cost measurements for core land operations, enabling objective assessment of economic feasibility.

Limitations

- Tests were done in a simulation, not in a real government environment.
- All districts were assumed to have similar systems and documents, which is not always true in practice.
- Very heavy transaction loads were not tested.
- These issues can be improved in future real-world implementations.

Future Works

- In future, TPACP can be improved by selecting validators more dynamically based on real-time network performance.
- Since this work is simulation-based, a real pilot implementation across districts in Bangladesh would strengthen its practical validity.
- Smart contracts can be further optimized to reduce gas usage for repeated operations like registration and ownership transfer.

References (1/2)

[1] K. M. Alam, J. A. Rahman, A. Tasnim, and A. Akther, “A blockchain-based land title management system for bangladesh,” Journal of King Saud University-Computer and Information Sciences, vol. 34, no. 6, pp. 3096–3110, 2022.

[2] A. S. Yadav, S. Shikha, S. Gupta, and D. S. Kushwaha, “The efficient consensus algorithm for land record management system,” IOP Conference Series: Materials Science and Engineering, vol. 1022, p. 012090, jan 2021.

[3] A. S. Yadav, S. Agrawal, and D. S. Kushwaha, “Distributed ledger technology based land transaction system with trusted nodes consensus mechanism,” Journal of King Saud University-Computer and Information Sciences, vol. 34, no. 8, pp. 6414– 6424, 2022.

[4] A. M. Munir Hussain, “Reputation-based leader selection consensus algorithm with rewards for blockchain technology,” Computers, vol. 14, no. 1, p. 20, 2025.

[5] R. I. Shithy, N. Mohammad, H. A. Ruhullah, S. Y. Oni, and M. A. Amin, “A blockchain based land registration and ownership management system for bangladesh,” in Proceedings of the 2021 4th International Conference on Blockchain Technology and Applications, pp. 94–100, 2021.

[6] S. Nakamoto, “Bitcoin: A peer-to-peer electronic cash system,” 2008.

[7] D. Larimer, “Delegated proof-of-stake (dpos),” Bitshare whitepaper, vol. 81, p. 85, 2014.

References (2/2)

[8] S. M. H. Bamakan, A. Motavali, and A. B. Bondarti, “A survey of blockchain consensus algorithms performance evaluation criteria,” Expert Systems with Applications, vol. 154, p. 113385, 2020.

[9] W. Wang, D. T. Hoang, P. Hu, Z. Xiong, D. Niyato, P. Wang, Y. Wen, and D. I. Kim, “A survey on consensus mechanisms and mining strategy management in blockchain networks,” Ieee Access, vol. 7, pp. 22328–22370, 2019.

[10] B. O. Ansah, W. Voss, K. O. Asiama, and I. Y. Wuni, “A systematic review of the institutional success factors for blockchain-based land administration,” Land use policy, vol. 125, p. 106473, 2023.

[11] S. Banerjee, K. Kumar, P. Masulkar, R. Amin, and S. K. Dwivedi, “Blockchain and ipfs-based reliable land registry system,” Security and Privacy, vol. 5, no. 5, p. e236, 2022

[12] J. P. Paavo, R. Rodr´ıguez-Puentes, and U. E. Chigbu, “Practicality of blockchain technology for land registration: A namibian case study,” Land, vol. 14, no. 8, p. 1626, 2025.

[13] K. S. and G. Sarath, “Securing land registration using blockchain,” Procedia Computer Science, 2020.

[14] R. M. Zein and H. Twinomurinzi, “Blockchain technology in lands registration: A systematic literature review,” JeDEM-eJournal of eDemocracy and Open Government, vol. 15, no. 2, pp. 1–36, 2023.

Q&A Session

Any Questions or Suggestions?

TPACP Algorithm (1/2)

Algorithm 1 Performance-Weighted Eligibility Selection

Require: TransactionPool, NodeHistory
Ensure: QualifiedValidators

Initialize performanceScores $\leftarrow \emptyset$
for each node in NetworkNodes **do**
    if node.isNew **then**
        node.baseScore $\leftarrow 30$
    else
        node.baseScore $\leftarrow$ node.historicalPerformance $\times 0.7 +$ node.responseSpeed $\times 0.3$
    end if
    if node.recentValidations > 0 **then**
        accuracy $\leftarrow$ node.correctValidations / node.recentValidations
        node.adjustedScore $\leftarrow$ node.baseScore $\times (0.8 + 0.2 \times$ accuracy$)$
    else
        node.adjustedScore $\leftarrow$ node.baseScore
    end if
    performanceScores[node.id] $\leftarrow$ node.adjustedScore
end for
avgScore $\leftarrow$ calculateAverage(performanceScores)
threshold $\leftarrow$ avgScore $\times 0.85$
QualifiedValidators $\leftarrow$ {node | performanceScores[node.id] $\geq$ threshold}
return QualifiedValidators

Algorithm 1.0: *Performance-Weighted Node Eligibility*

Algorithm 2 Territorial Verification

Require: QualifiedValidators, TransactionDetails
Ensure: PrimaryValidator, SecondaryValidators

transactionTerritory $\leftarrow$ TransactionDetails.district
localValidators $\leftarrow$ { node $\in$ QualifiedValidators | node.territory = transactionTerritory }
if count(localValidators) > 0 **then**
    sort(localValidators, by=performanceScore, descending=true)
    PrimaryValidator $\leftarrow$ localValidators
else
    regionalValidators $\leftarrow$ { node $\in$ QualifiedValidators | node.region = transactionTerritory.region }
    sort(regionalValidators, by=performanceScore, descending=true)
    PrimaryValidator $\leftarrow$ regionalValidators
end if
externalValidators $\leftarrow$ { node $\in$ QualifiedValidators | node.territory $\neq$ transactionTerritory }
groupedValidators $\leftarrow$ groupBy(externalValidators, key=district)
SecondaryValidators $\leftarrow$ empty list
for each territoryGroup in groupedValidators **do**
    sort(territoryGroup, by=performanceScore, descending=true)
    append territoryGroup[0] to SecondaryValidators
end for
if count(SecondaryValidators) > 5 **then**
    sort(SecondaryValidators, by=performanceScore, descending=true)
    SecondaryValidators $\leftarrow$ first 5 elements of SecondaryValidators
end if
return PrimaryValidator, SecondaryValidators

Algorithm 2.0: *Territorial Verification Framework*

TPACP Algorithm (2/2)

Algorithm 3 Integrity Assessment

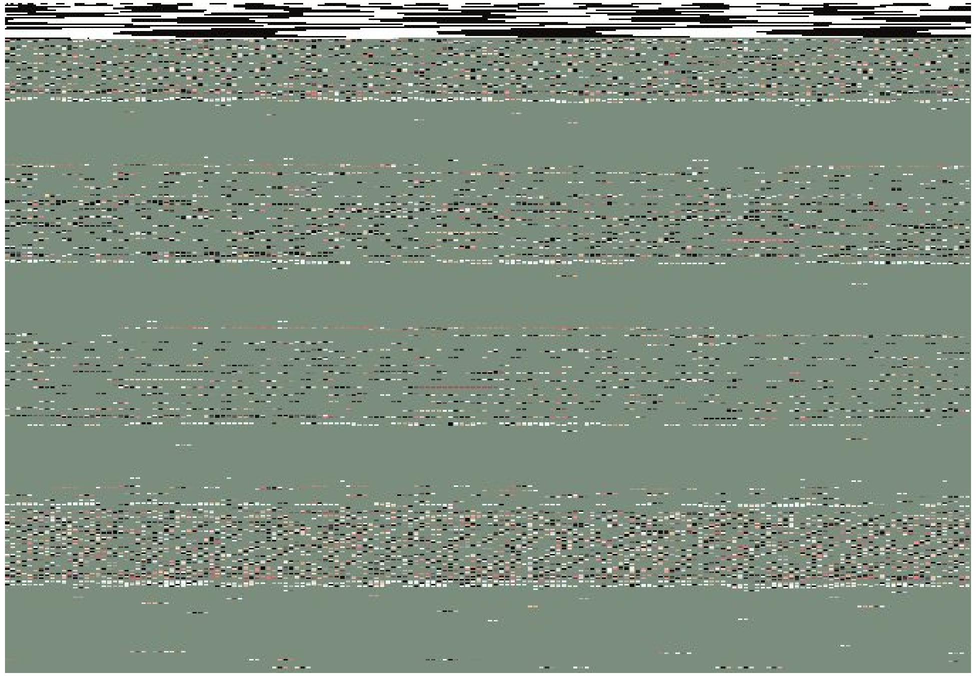
```
Require: PropertyDocuments, PrimaryValidator, SecondaryValidators
Ensure: VerificationResult, DocumentScores
documentScores ← empty map
primaryResults ← PrimaryValidator.verify({ documentAuthenticity: true, witnessVerification: true, localRegistryCheck: true })
secondaryResults ← empty list
for each validator in SecondaryValidators do
    result ← validator.verify({ documentHashes: true, ownershipHistory: true, boundaryConsistency: true })
    append result to secondaryResults
end for
for each document in PropertyDocuments do
    primaryScore ← getScore(primaryResults, document)
    secondaryScores ← empty list
    for each result in secondaryResults do
        append getScore(result, document) to secondaryScores
    end for
    avgSecondary ← average(secondaryScores)
    documentScores[document.id] ← (primaryScore × 0.6) + (avgSecondary × 0.4)
end for
if all values in documentScores > 75 then
    VerificationResult ← "APPROVED"
else if any value in documentScores < 40 then
    VerificationResult ← "REJECTED"
else
    VerificationResult ← "REQUIRES_MANUAL_REVIEW"
end if
return VerificationResult, documentScores
```

Algorithm 3.0: Document Integrity Assessment

Algorithm 4 Immutable Finalization

```
Require: VerificationResult, DocumentScores, TransactionDetails
Ensure: BlockStatus, AuditRecord
AuditRecord ← CreateAuditRecord
AuditRecord.timestamp ← currentTime
AuditRecord.transactionID ← TransactionDetails.id
AuditRecord.territory ← TransactionDetails.district
AuditRecord.documentScores ← DocumentScores
AuditRecord.validators ← [PrimaryValidator.id] ∪ {v.id | v ∈ SecondaryValidators}
if VerificationResult = "APPROVED" then
    docHashes ← calculateHashes(PropertyDocuments)
    signatures ← collectSignatures(PrimaryValidator, SecondaryValidators)
    newRecord ← createRecord(TransactionDetails, docHashes, signatures)
    if TransactionDetails.propertyValue > HIGH_VALUE_THRESHOLD then
        storeWithRedundancy(newRecord, 3)
    else
        storeWithRedundancy(newRecord, 2)
    end if
    updatePerformanceScores(PrimaryValidator, SecondaryValidators, "positive")
    BlockStatus ← "FINALIZED"
    notifyOffices(TransactionDetails.district, newRecord)
else if VerificationResult = "REJECTED" then
    BlockStatus ← "REJECTED"
    updatePerformanceScores(PrimaryValidator, SecondaryValidators, "negative")
    AuditRecord.rejectionReason ← identifyLowScoringDocuments(DocumentScores)
else
    BlockStatus ← "PENDING_MANUAL_REVIEW"
    routeToManualReview(TransactionDetails, DocumentScores)
end if
storeAuditRecord(AuditRecord)
return BlockStatus, AuditRecord
```

Algorithm 3.0: *Immutable Record Finalization*



Thank You