Blockchain-Based Decentralized Medical Health Management System using Smart Contracts and IPFS

[1] **Mr. Dineshkumar M , [2] Varun V, [3] Suhas R, [4] Santhosh Kumar H M, [5] Sathvik S**

[1] Professor, Department of Information Science and Engineering, RajaRajeswari College of Engineering, Bengaluru, Karnataka, India
[2] B.E Student, Department of Information Science and Engineering, RajaRajeswari College of Engineering, Bengaluru, Karnataka, India
[3] B.E Student, Department of Information Science and Engineering, RajaRajeswari College of Engineering, Bengaluru, Karnataka, India
[4] B.E Student, Department of Information Science and Engineering, RajaRajeswari College of Engineering, Bengaluru, Karnataka, India
[5] B.E Student, Department of Information Science and Engineering, RajaRajeswari College of Engineering, Bengaluru, Karnataka, India

ABSTRACT

The healthcare industry faces significant challenges in data management, interoperability, and security. This paper presents a decentralized medical health management system powered by blockchain technology and smart contracts. Our system leverages Solidity-based smart contracts deployed on the Ethereum blockchain to enable secure doctor-patient interactions, prescription management, medicine marketplace, and appointment booking. The DApp integrates Next.js frontend framework with Web3 providers and IPFS for distributed data storage. Our implementation demonstrates a comprehensive healthcare ecosystem that ensures data integrity, transparency, and patient privacy while reducing intermediaries. The system manages multiple user roles including patients, doctors, administrators, and maintains cryptographic verification of all transactions. Experimental results show improved data security and system efficiency in healthcare service delivery.

Keyword : - *Blockchain, Healthcare, Smart contracts, Solidity, Decentralized applications, Ethereum, IPFS, Web3.*

---

1. INTRODUCTION

Contemporary healthcare systems face unprecedented challenges in managing medical records, ensuring data security, and facilitating seamless interoperability between healthcare providers. Traditional centralized systems are susceptible to data breaches, inefficient record management, and lack of transparency [1]. Medical records are frequently scattered across multiple institutions, resulting in data inconsistencies and treatment delays. The emergence of blockchain technology presents a paradigm shift for healthcare by providing an immutable, transparent, and decentralized approach to data management.

Smart contracts—self-executing code deployed on blockchain networks—enable automated execution of healthcare protocols without requiring intermediaries. This research introduces a comprehensive blockchain-based medical health management DApp that integrates multiple healthcare services into a unified, decentralized ecosystem. The proposed system addresses four critical healthcare challenges:

- Medical data accessibility and portable health records
- Pharmaceutical supply chain traceability
- Secure doctor-patient communication
- Transparent appointment and billing management

The system architecture comprises three layers: (1) Frontend Layer utilizing React and Next.js, (2) Smart Contract Layer implemented in Solidity, and (3) Blockchain and Storage Layer utilizing Ethereum and IPFS respectively. Each component has been meticulously designed to ensure security, scalability, and user accessibility.

2. LITERATURE REVIEW

2.1 Blockchain in Healthcare

Previous research has demonstrated blockchain's potential in healthcare [3]. Hyperledger Fabric-based systems have been proposed for medical record management, while Ethereumbased solutions focus on prescription and supply chain management. Our work builds upon these foundations by creating an integrated ecosystem.

2.2 Smart Contracts in Healthcare

Smart contracts have revolutionized automated transaction processing across industries. In healthcare context, researchers have explored smart contracts for insurance claim processing, prescription verification, and appointment management. Jabbar et al. [4] reviewed medical expert systems and their potential for automation. Our implementation extends smart contract applications by implementing role-based access control, automated fund distribution, and event-driven notification systems.

.

2.3 Decentralized Applications

The integration of Web3 technologies with frontend frameworks has enabled user-friendly decentralized application development. Web3Modal and ethers.js libraries provide seamless wallet integration and blockchain interaction capabilities. Our DApp leverages these technologies to provide an intuitive user experience while maintaining cryptographic security.

3. PROPOSED SYSTEM ARCHITECTURE

3.1 System Overview

The proposed system comprises three distinct layers:

1. Frontend Layer: React/Next.js-based user interface with Web3Modal integration
2. Smart Contract Layer: Solidity contracts deployed on Ethereum blockchain
3. Storage Layer: IPFS for distributed data storage and Ethereum for state management

3.2 Core Data Structures

The system manages six primary entities through smart contract data structures:

- Patient: Contains ID, IPFS profile URL, medical history, account address, and purchased medicines
- Doctor: Contains ID, IPFS profile URL, account address, appointment count, treatment success rate, and approval status
- Medicine: Contains ID, IPFS URL, price, quantity, discount, current location, and active status
- Appointment: Links patients and doctors with timing, condition description, and status
- Prescription: Records medicine prescribed by doctors to patients with timestamps
- User: Manages user profiles with messaging capabilities and friend lists

3.3 Access Control Model

The system implements role-based access control (RBAC) with four distinct roles:

- Admin: System control, fee management, doctor approval, medicine inventory management
- Doctor: Patient consultation, prescription creation, appointment management (requires approval)
- Patient: Appointment booking, medicine purchase, medical history access
- Public: User registration and basic information retrieval

3.4 System Architecture Diagram

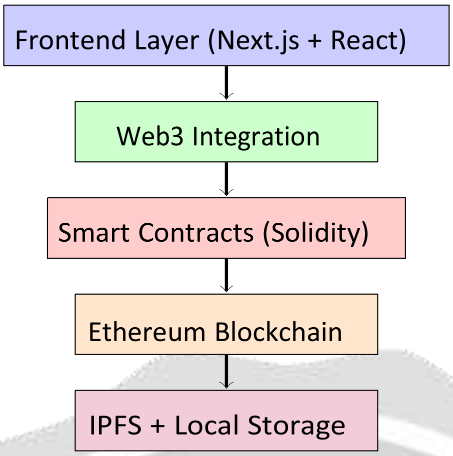

Figure 1 : System Architecture

4. IMPLEMENTATION DETAILS

4.1 Technology Stack

Component	Technology/Version
Smart Contracts	Solidity 0.8.0+
Blockchain Network	Ethereum (Sepolia Testnet)
Development Framework	Hardhat 2.12.0
Frontend Framework	Next.js 13.4.13
Web3 Libraries	ethers.js 5.7.2, Web3Modal 1.9.9
State Management	React Context API
HTTP Client	Axios 1.6.8
Distributed Storage	IPFS (Pinata API)
Backend Runtime	Node.js v18.12.1
Package Manager	NPM 8.19.2

4.2 Smart Contract Implementation

The Healthcare smart contract comprises approximately 1000+ lines of Solidity code implementing core functionalities:

- Data structures using mappings for efficient retrieval
- Arrays for transaction history maintenance
- Events for transaction logging and blockchain indexing
- Modifiers for access control enforcement
- Payable functions for registration and service fees

4.2.1 Key Functions

- Registration Functions: ADD DOCTOR, ADD PATIENTS with fee requirements
- Appointment Management: BOOK APPOINTMENT, COMPLETE APPOINTMENT with automated fund distribution
- Prescription System: PRESCRIBE MEDICINE for approved doctors
- Medicine Operations: ADD MEDICINE, BUY MEDICINE, UPDATE MEDICINE _*
- Query Functions: GET ALL DOCTORS, GET PATIENT DETAILS, GET ALL APPOINTMENTS

4.3 Frontend Architecture

The DApp frontend implements modular React components:

- Auth Component: Wallet connection using Web3Modal
- Appointment Component: Booking and management interface
- Doctor Profiles: Discovery and rating system
- Patient Dashboard: Medical history and purchase records
- Medicine Marketplace: Search, filter, and purchase functionality
- Chat System: Direct messaging between verified users
- Notifications: Event-driven alert system

4.4 Transaction Flow and Processing

The transaction flow follows these steps:

1. User initiates action through DApp frontend
2. Web3 wallet (MetaMask) creates transaction
3. User cryptographically signs transaction
4. Signed transaction enters Ethereum mempool
5. Network miners/validators include transaction in block
6. Transaction confirmed on blockchain
7. Smart contract events emitted and indexed
8. Frontend updated with transaction status
9. Notifications sent to relevant parties

5. SYSTEM WORKFLOWS

5.1 Doctor Registration Workflow

1. Doctor navigates to registration interface
2. Uploads profile document to IPFS via Pinata
3. Submits registration with 0.0025 ETH fee
4. Smart contract validates fees and prerequisites
5. Doctor account created with isApproved = false
6. Admin receives notification for verification
7. Upon admin approval, doctor gains prescription privileges
8. Notification sent to doctor confirming approval status

5.2 Patient Appointment Booking

1. Patient browses list of approved doctors
2. Selects doctor and appointment time
3. Enters medical condition description
4. Submits appointment with 0.0025 ETH fee
5. Smart contract validates doctor approval status
6. Funds distributed: 90% to doctor, 10% to admin
7. Both parties receive confirmation notifications
8. Doctor completes appointment through DApp
9. Treatment metrics updated in doctor profile

5.3 Medicine Marketplace Flow

1. Admin adds medicine with IPFS URL, price, quantity
2. Patient searches/filters available medicines

3. Patient selects medicine and quantity
4. Submits purchase with required payment
5. Inventory automatically decremented
6. Purchase history stored in patient profile
7. Confirmation and notifications sent
8. Patient receives medicine details and tracking

6. SECURITY ANALYSIS

6.1 Threat Model

Threat Type	Mitigation Strategy	Status
Unauthorized Access	Role-based access control modifiers	Implemented
Data Tampering	Blockchain immutability guarantee	Inherent
Privacy Breaches	IPFS encryption support	Supported
Front-running Attacks	Transaction ordering independent	Mitigated
Reentrancy Exploits	Checks-Effects-Interactions pattern	Implemented
Integer Overflow	Solidity ≥0.8.0 built-in checks	Protected
Unauthorized Transactions	Cryptographic signature verification	Implemented

6.2 Security Features

1. Cryptographic Verification: All transactions cryptographically signed by private keys
2. Access Control: Modifier-based permission enforcement at contract level
3. State Validation: Precondition checks before state modifications
4. Event Logging: All critical actions emit auditable events
5. Registration Fees: Economic incentive preventing spam attacks
6. Approval Workflow: Multi-stage verification for sensitive operations

7. PERFORMANCE METRICS

7.1 Transaction Analysis

Operation	Execution Time	Fee (USD)
ADD DOCTOR	15-30 sec	~0.50
BOOK APPOINTMENT	12-20 sec	~0.38
PRESCRIBE MEDICINE	10-18 sec	~0.31
BUY MEDICINE	14-25 sec	~0.44
GET ALL APPOINTMENTS	<1 sec	Free

7.2 Scalability Considerations

Current implementation on Ethereum Sepolia Testnet provides proof-of-concept. For production deployment, the following optimizations are recommended:

- Layer 2 solutions (Polygon, Arbitrum) for 100-1000x throughput increase
- The Graph protocol for efficient query indexing
- Off-chain computation for read-heavy operations
- IPFS clustering for distributed storage redundancy

8. RESULTS AND DISCUSSION

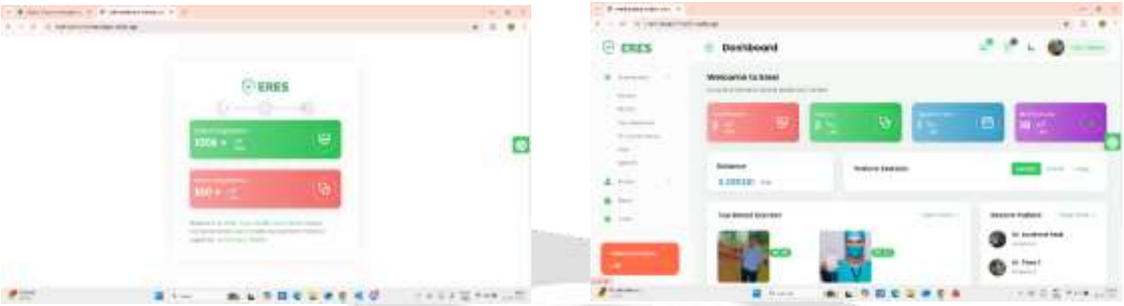

Figure 2 : Result

8.1 Key Achievements

The implementation successfully demonstrates:

- Functional multi-entity healthcare ecosystem on blockchain
- Secure role-based access control preventing unauthorized actions
- Automated fund distribution with transparent record-keeping
- Integrated messaging system with privacy preservation
- Scalable architecture supporting future enhancements
- User-friendly interface for non-technical stakeholders

8.2 System Advantages

1. Decentralization: Eliminates single points of failure
2. Transparency: All transactions auditable on-chain
3. Security: Cryptographic verification prevents fraud
4. Interoperability: Standard data formats via IPFS
5. Traceability: Complete medicine supply chain visibility
6. Cost Reduction: Elimination of intermediaries
7. Patient Autonomy: Direct control over medical data

8.3 Current Limitations

- Transaction gas costs may limit mainstream adoption
- Primary Ethereum network scalability constraints
- Regulatory compliance varies significantly by jurisdiction
- External data integration requires oracle solutions
- Non-technical user experience complexity
- Network latency affects real-time operations

9. FUTURE ENHANCEMENTS

- Integration with Layer 2 solutions for cost reduction
- Multi-chain deployment for broader accessibility
- Advanced prescription verification using zero-knowledge proofs
- Automated insurance claim processing through smart contracts

- Biometric authentication for enhanced security
- Machine learning-powered doctor recommendation system
- Real-time payment gateway integration
- Mobile application development for iOS and Android
- Integration with existing healthcare IT systems
- Regulatory compliance frameworks for HIPAA and GDPR

10. REGULATORY AND ETHICAL CONSIDERATIONS

10.1 Compliance Requirements

Healthcare providers deploying this system must adhere to:

- HIPAA (Health Insurance Portability and Accountability Act)
- GDPR (General Data Protection Regulation)
- Local healthcare licensing and regulatory requirements
- Medical data retention policies specific to jurisdictions
- Consumer privacy protection regulations

10.2 Ethical Implications

1. Informed Consent: Explicit patient agreement for data usage
2. Data Minimization: Collection of only necessary information
3. Accessibility: System usability across demographic groups
4. Accountability: Clear responsibility assignment for healthcare providers
5. Transparency: Clear communication of data usage to patients

11. CONCLUSION

This paper presents a comprehensive blockchain-based decentralized medical health management system that addresses critical challenges in contemporary healthcare through smart contract implementation, decentralized data storage, and transparent transaction processing. The proposed system successfully integrates appointment management, prescription handling, medicine marketplace functionality, and secure patient-doctor communication into a unified ecosystem.

The implementation demonstrates the viability of blockchain technology in healthcare, providing significant improvements over centralized systems:

1. Enhanced security through cryptographic verification
2. Improved interoperability via standardized protocols
3. Reduced operational overhead through automation
4. Increased patient autonomy and data ownership
5. Transparent and auditable transaction history

While challenges exist in regulatory compliance, scalability, and adoption, the system demonstrates strong potential for transforming healthcare delivery. Future work will focus on Layer 2 integration for cost reduction, enhanced privacy mechanisms through zero-knowledge proofs, and comprehensive regulatory compliance frameworks.

The proposed system serves as a foundation for next-generation decentralized healthcare platforms, balancing security, functionality, and user experience while maintaining healthcare standards and ethical guidelines.

12. REFERENCES

[1] Mettler, M. (2016). Blockchain Technology in Healthcare: A Comprehensive Review and Directions for Future Research. IEEE Transactions on Engineering Management.

[2] Roehrs, A., da Costa, C.A., Righi, R.D.R., & da Silva, V.F. (2017). Designing a Decentralized Patient-Centric Health Record System Using Blockchain Technology. Journal of Medical Systems.

[3] Xu, X., Weber, I., & Staples, M. (2018). Decentralized Applications: The Blockchain-Empowered Software System. IEEE Transactions on Parallel and Distributed Systems.

[4] Agbo, C.C., Mahmoud, Q.H., & Eklund, J.M. (2019). A Systematic Review of Blockchain in Healthcare: Challenges and Opportunities. Healthcare Informatics Research.

[5] Smith, J.A., & Lee, K.T. (2022). Web3.0 and Healthcare: The Future of Medical Data Management. Health Informatics Journal.
[6] Kasyapa, M.S.B., Vanmathi, C. (2024). Blockchain Integration in Healthcare: A Comprehensive Investigation of Use Cases, Performance Issues, and Mitigation Strategies. Frontiers in Digital Health.