FINAL ASSIGNMENT (END EVALUATION)

Secure Permissioned Blockchain Simulation for Medical Supply Chain

The pharmaceutical industry plays a pivotal role in global healthcare but faces persistent challenges related to authenticity, traceability, and secure data sharing among stakeholders. Counterfeit medicines continue to infiltrate supply chains, prescription misuse remains unchecked, and billing or insurance claim processes are often delayed due to fragmented systems and siloed databases. Traditional centralized solutions are ill-suited to address these issues because they are vulnerable to data tampering, single points of failure, and lack transparency across stakeholders.

A decentralized approach using blockchain technology has emerged as a promising alternative. Blockchain's core properties such as immutability, distributed consensus, and transparency make it an ideal foundation for building systems that ensure end-to-end drug traceability, enforce medicinal correctness, and automate billing workflows. However, most existing blockchain-based healthcare prototypes still depend heavily on off-chain databases for storing sensitive data, undermining the very decentralization and security they aim to achieve.

Overview

To address these limitations, we propose a secure, role-based, blockchain-powered simulation of a healthcare supply management system. This system combines a Node.js linkage layer with smart contracts to simulate a permissioned blockchain network where only authorized stakeholders—manufacturers, intermediaries (e.g., distributors, quality checkers), pharmacists, patients, and insurers—can participate.

All critical data and state transitions are stored directly on the blockchain. Node.js serves as an intermediary for orchestrating frontend interactions and blockchain operations without relying on a traditional database. A key enhancement in this system is the implementation of time-based expiry tracking for drug units, ensuring that expired drugs cannot be dispensed or transferred.

Additionally, each NFT representing a drug unit is mapped to a unique identifier that, in a real-world deployment, would correspond to a QR code printed on the physical packaging. This enables easy scanning and verification of authenticity throughout the supply chain. We simulated QR code scanning with the ID based mapping for each NFT.

Core Functionalities

1. Tracking individual drug units across their lifecycle from manufacturing to patient consumption using blockchain, eliminating centralized databases and data silos.

2. Validating prescriptions and enforcing medicinal correctness in a fully decentralized manner.

3. Automating billing and insurance claim processing directly within the blockchain.

4. Implementing fine-grained, role-based access control to protect sensitive patient and transaction data from unauthorized access.

5. Using Non-Fungible Tokens (NFTs) to represent drug units digitally, enabling secure and traceable ownership transfers along the supply chain.

6. Enforcing time-based expiry for drug units using the blockchain's `block.timestamp`, preventing any transfer or dispensation of expired drugs.

7. Simulating intermediaries (distributors, quality control agents) who verify drug batches at various stages before reaching pharmacists.

8. Mapping NFTs to QR codes(ID's) for physical verification, enabling stakeholders to scan and validate drug authenticity.

System Architecture

The system is designed as a web-based decentralized application (DApp) with four main components:

Component	Description
Frontend	A React.js or Next.js-based DApp providing dashboards for all roles. Users interact through a secure and intuitive interface.
Node.js Backend	Acts as a middleware layer to orchestrate frontend requests, communicate with smart contracts via Web3.js or Ethers.js, manage user sessions, and prepare transactions.
Smart Contract	A smart contract to simulate permission based blockchain working that stores all application state, drug metadata, role-based access controls, and expiry logic. Manage the drug lifecycle states, validate prescriptions, handle billing and insurance logic, implement expiry tracking, and enable NFT-based ownership transfers mapped to QR codes.

Data Flow

1. The frontend sends requests to the Node.js backend for actions such as registering drugs, validating prescriptions, scanning QR codes, and dispensing medicine.

2. The Node.js backend prepares transactions, interacts with smart contracts, and sends responses back to the frontend.

3. The blockchain stores all records immutably, ensuring data integrity and traceability.

4. Smart contracts use `block.timestamp` to check drug expiry and ensure only non-expired drugs can move through the supply chain.

5. Intermediaries verify batches by scanning QR codes and updating on-chain verification statuses before forwarding drugs to the next stakeholder.

Role-Based Access Control

Role	Capabilities
Manufacturer	Register drug batches, mint NFTs representing each unit with metadata including expiry timestamp and QR code mapping, transfer NFTs to intermediaries.
Intermediary	Verify drug batches using QR codes, perform quality checks, approve/reject batches, transfer verified NFTs to pharmacists.
Pharmacist	Verify prescriptions, dispense drugs by transferring NFTs to patients (only if not expired and verified), update drug status.
Patient	View active prescriptions, track NFT-based drug ownership, and access billing and insurance data.
Insurer	Monitor dispensation events, automatically process claims, and settle payments based on smart contract rules.

Ownership of each drug unit is represented as an NFT. As the drug moves from manufacturer to intermediary to pharmacist to patient, the ownership of the corresponding NFT is transferred securely on-chain. Before any transfer, the smart contract verifies that:

- The drug's expiry date has not passed (using `block.timestamp`).

- The intermediary verification status is set to "approved."

- When a drug is dispensed to the patient, the NFT is burned to indicate that the drug has been consumed.

Security and Privacy Considerations

1. The smart contract handles role based access.
2. Sensitive metadata such as patient details is encoded and stored directly within smart contract state variables, minimizing exposure.
3. Privacy is enhanced through access-controlled smart contract functions and simulated private blockchain channels.
4. Expiry validation and intermediary verification are enforced entirely on-chain to prevent unauthorized sale or use of expired or unverified drugs.
5. NFT ↔ QR code mapping allows physical units to be validated against their digital twins securely.

Assumptions

1. The system simulates a permissioned blockchain where only pre-approved participants can interact with the network.

2. The contract owner (administrator) is responsible for admitting new participants by assigning them roles: Manufacturer, Intermediary, Pharmacist, and Insurer.

3. Patients are allowed to register themselves directly but are limited to patient-specific actions.

4. All other roles must be explicitly defined and assigned by the contract owner to ensure only verified entities participate in critical operations.

5. Each role inherits the basic privileges of a patient but adds additional capabilities based on their responsibilities.

6. Drug expiry times are pre-defined at the time of minting NFTs by manufacturers and enforced directly by the smart contract logic.

7. Intermediary verification of drug batches is simulated by mapping NFTs to QR codes, which would correspond to physical scanning in real-world deployments.