Blockchain Technology Mini project

Project Title: Blockchain-Based Application for Health-Related Medical Records

Group Members:

- **Tanay Jadhav(29)**
- **Ritesh Shevare(30)**
- **Om Taskar(31)**
- **Pratik Kotkar(32)**

Problem Statement:

The healthcare industry faces significant challenges in securing sensitive patient data, which is often stored in centralized systems vulnerable to breaches, unauthorized access, and tampering. This project proposes **SwasthyaChain**, a blockchain-based decentralized application (dApp) that ensures the integrity, security, and privacy of electronic health records (EHR). The solution leverages blockchain technology and smart contracts to create an immutable, auditable, and secure platform for managing health-related records, while addressing scalability issues using off-chain storage.

Objectives:

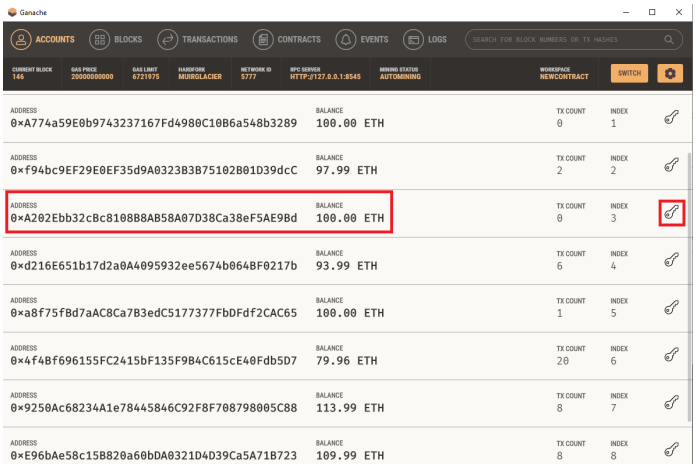
- **Data Integrity**: Ensure the immutability of medical records by storing cryptographic hashes on a secure blockchain network.
- **Patient Privacy**: Protect patient identities and their medical information through cryptographic encryption while enabling authorized access.
- **Transparency**: Provide real-time access logs and history of data modifications for auditing purposes by patients and healthcare providers.
- **Decentralized Data Storage**: Distribute medical records across a blockchain network, eliminating single points of failure.
- **User-Friendly Interface**: Create a simple, intuitive platform for healthcare providers and patients to manage medical records seamlessly.

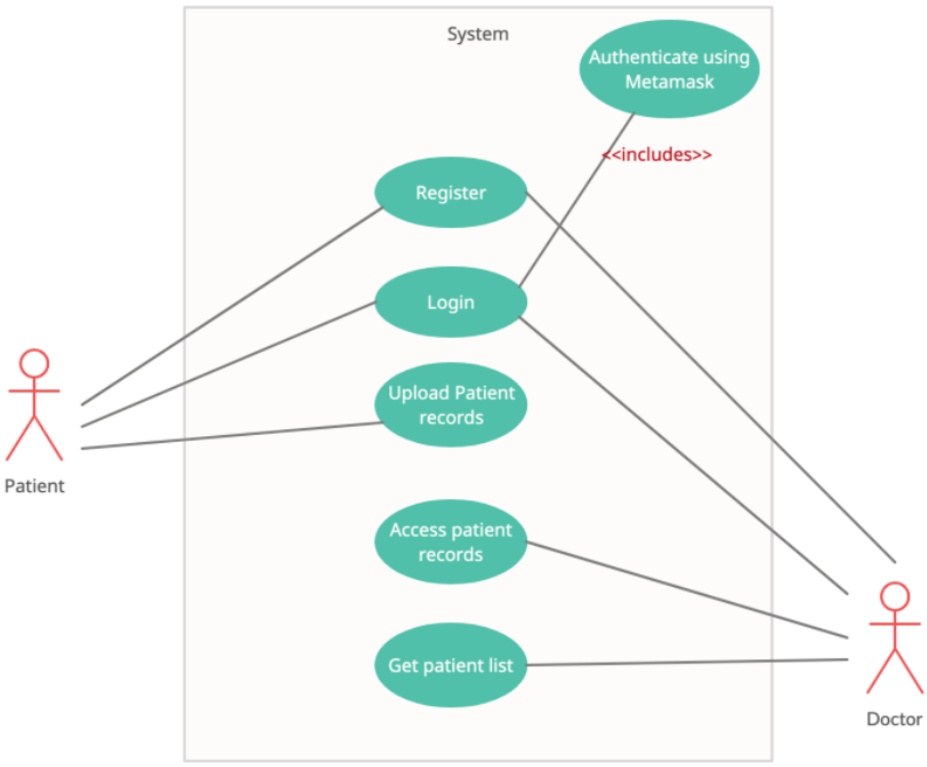
System Architecture:

The system comprises several key components:

1. **User Interface (UI)**: A web-based application where users (patients and healthcare providers) can register, view, and manage medical records.
2. **Smart Contracts**: Developed in Solidity, these contracts handle the logic for user registration, record access, and permissions.
3. **Blockchain Network**: A private Ethereum or Hyperledger network stores cryptographic hashes of medical records, ensuring immutability and transparency.
4. **Off-Chain Storage**: Sensitive medical records are stored off-chain using encrypted storage on IPFS (InterPlanetary File System), while cryptographic hashes are stored on the blockchain to verify the records' authenticity.

5. **Access Control System**: Role-based access control mechanisms ensure only authorized individuals (e.g., doctors and patients) can view or modify medical records.





Implementation Details:

- **Frontend**: Built using React.js, providing a responsive and user-friendly interface for both patients and healthcare providers.
- **Backend**: Node.js is used to handle the server-side logic and interactions with the blockchain network.
- **Blockchain**: A private Ethereum or Hyperledger network is used to store cryptographic hashes of medical records.
- **Smart Contracts**: Solidity-based smart contracts control the access permissions and audit trails of medical records.
- **Off-Chain Storage**: Patient records are stored off-chain using IPFS, while only the cryptographic hashes are stored on the blockchain for scalability.

Features:

1. **Patient Registration**: A secure registration system with identity verification to ensure that only authorized individuals can access their medical records.
2. **Record Access and Sharing**: Patients can control who accesses their records, with doctors or healthcare providers being able to view the records for a limited time, as defined by the patient.
3. **Audit Logs**: Every interaction with a medical record, such as viewing, modifying, or sharing, is recorded on the blockchain, providing transparency for both patients and healthcare providers.
4. **Encryption**: Off-chain medical records are encrypted using the patient's private key, ensuring that only authorized individuals can decrypt and access the data.
5. **Decentralized Storage**: Cryptographic hashes of the medical records are stored on the blockchain, while the actual medical data is stored off-chain in IPFS, maintaining scalability and efficiency.

Installation and Configuration (Technical Details):

1. **Dependencies**:
    - Install Node.js and npm.
    - Install Ganache for local blockchain simulation.
    - Install IPFS Desktop for decentralized storage.
    - Add MetaMask as a browser extension for managing blockchain transactions.
2. **Smart Contract Deployment**:
    - Install Truffle globally using npm install truffle -g.
    - Compile and deploy the smart contracts with truffle compile and truffle migrate.
    - Update the deployed contract address in the front-end application.

3. **Running the dApp**:
    - Connect MetaMask to the local blockchain.
    - Start the IPFS Desktop application for off-chain data storage.
    - Start a local server to run the web application (npm start).

Conclusion:

The **SwasthyaChain** project provides a blockchain-based solution for the secure, transparent, and decentralized management of medical records. By integrating blockchain technology with off-chain storage (IPFS), this system ensures data integrity, patient privacy, and transparency in healthcare data management. Future improvements may include the integration of advanced cryptographic techniques such as zero-knowledge proofs (ZKPs) and enhancing interoperability with existing healthcare systems.