MedVault

Software Design Document

Líder do projeto:

Jorge Abrão Neto

Membros:

David Samuel Tavares de Moraes

Lucas Mendes da Silva

Paulo Victor Rocha de Almeida

Instrutor:

Sérgio Carvalho

2023

Sumário

1. Overview do projeto

1.1. Instrutor

Este trabalho foi concebido sob a orientação do Professor Sérgio Teixeira de Carvalho, no âmbito da disciplina de Sistemas Distribuídos. O propósito deste projeto final foi a aplicação prática dos conceitos abordados ao longo do curso.

1.2. Contextualização

É de extrema importância nos sistemas de saúde a proteção dos dados médicos contra possíveis ataques mal intencionados capazes de lesar o paciente. Assim, torna-se indispensável ter mecanismos que garantam a segurança, privacidade e disponibilidade desses dados apenas para entidades autorizadas mediante a permissão prévia do paciente.

A título de ilustração acerca dos riscos citados, podemos destacar um incidente ocorrido em 2020 envolvendo o Ministério da Saúde que resultou na exposição dos dados de aproximadamente 243 milhões de brasileiros, incluindo pessoas já falecidas. Informações sensíveis, tais como endereço, número de telefone, nome, CPF, estado clínico e restrições médicas, nas mãos de indivíduos mal intencionados podem ocasionar uma série de implicações adversas como crimes relacionados à falsidade ideológica, ameaças à vida do paciente e uma variedade de danos que englobam aspectos financeiros e emocionais.

Diante deste cenário, o projeto MedVault utiliza-se da tecnologia de blockchain, mais precisamente o framework Hyperledger Sawtooth, aliada ao InterPlanetary File System (IPFS) para assegurar a integridade do prontuário médico de cada paciente, garantindo uma cadeia cronológica e impedindo alterações retroativas ou

adulterações.

No contexto deste projeto, podemos destacar como principais stakeholders

a. Pacientes e familiares que buscam segurança e confiabilidade de seus registros médicos prezando pela inviolabilidade dessas informações.

b. Profissionais de saúde que necessitam ter acesso em tempo real a informações precisas e imutáveis de maneira que possam ser confiáveis para orientar diagnósticos e tratamentos.

c. Órgãos regulatórios e entidades legais assegurando a conformidade das práticas adotadas com as normas de proteção de dados e privacidade.

1.3. Objetivos do projeto

Como solução para os problemas anteriormente descritos, o MedVault surge com a proposta de fornecer uma maneira segura e descentralizada de armazenar e compartilhar informações médicas sensíveis. Para isso, utiliza contratos inteligentes para gerenciar o acesso às informações armazenadas no IPFS por meio de um proxy desenvolvido, enquanto o Hyperledger Sawtooth é usado para garantir a integridade e a imutabilidade dos dados. A comunicação com o proxy é realizada a partir de uma interface web, sendo o proxy o responsável por se comunicar com o IPFS e o Hyperledger Sawtooth.

1.4. Discussões de relevância

Ao lidarmos com a maioria das blockchains, podemos nos deparar com problemas relacionados à escalabilidade. Por isso, o projeto emprega o conceito de armazenamento off-chain, armazenando informações fora da cadeia de blocos principal. Com essa abordagem, a rede de blockchain passa a lidar com volumes maiores de dados sem comprometer sua eficiência e desempenho, assegurando que as informações médicas estejam disponíveis em tempo real sem comprometer a integridade e a segurança dos dados que é garantida graças à estrutura imutável da blockchain.

Com essa estratégia off-chain, a blockchain é utilizada apenas para registrar referências ou hashes que comprovem a autenticidade e integridade dos dados armazenados no IPFS. Com isso, é reduzido significativamente o tamanho e a sobrecarga da blockchain, permitindo que grandes volumes de informações médicas sejam tratados com eficiência e maior grau de escalabilidade.

O armazenamento off-chain pode proporcionar alguns riscos relacionados à segurança, principalmente em relação à comunicação do proxy com o banco de dados. Esse banco de dados será responsável por armazenar as informações da requisição feita pelo profissional da área da saúde e a autorização do paciente em relação a disponibilização das informações. Todavia, essas informações estarão fortemente criptografadas e, caso o banco de dados seja violado, não comprometerão a segurança do sistema.

A tecnologia da blockchain também proporciona a descentralização dos dados, permitindo que eles sejam distribuídos de forma segura entre as partes autorizadas, reduzindo o risco de violações de privacidade e a centralização excessiva de informações médicas, o que torna o sistema tolerante à falhas.

Outro ponto abordado no projeto é o InterPlanetary File System (IPFS), sendo a solução escolhida para o armazenamento distribuído de dados. O IPFS é uma rede ponto a ponto que utiliza hashes criptográficos para endereçar conteúdo, permitindo

que os dados sejam armazenados de forma descentralizada, recuperados eficientemente e resistam à censura.

A rede IPFS é composta por uma série de nós interconectados, onde cada nó armazena um conjunto de dados e pode servir como ponto de acesso para solicitações de dados. Essa arquitetura distribuída e resistente à censura não apenas melhora a velocidade e a disponibilidade do acesso aos dados, mas também reduz a carga sobre os pontos centrais, superando os desafios tradicionais de escalabilidade e centralização de armazenamento de dados.

1.5. Trabalhos relacionados

[1] Madine, M. M., Al-Ayyoub, M., Jararweh, Y., & Al-Zoubi, H. (2020). Blockchain for Giving Patients Control Over Their Medical Records. IEEE Access, 8, 217487-217500.

O artigo fala sobre como usar contratos inteligentes baseados na blockchain Ethereum para dar aos pacientes controle sobre seus dados médicos de forma descentralizada, imutável, transparente, rastreável e confiável. Os autores propõem um sistema chamado PHR-BC que permite aos pacientes armazenar, compartilhar e revogar o acesso aos seus registros de saúde usando a blockchain. Eles também apresentam uma avaliação de desempenho e segurança do sistema.

[2] Kumar, S., Bharti, A. K., & Amin, R. (2021). Decentralized secure storage of medical records using Blockchain and IPFS: A comparative analysis with future directions. Security and Privacy, 4(5), e173.

O artigo apresenta um estudo detalhado das soluções de armazenamento seguro de dados de saúde baseadas em IPFS e Blockchain. Ele analisa as soluções existentes e suas arquiteturas, que podem facilitar a pesquisa e o desenvolvimento futuros das tecnologias emergentes de IPFS e Blockchain. Ele também propõe um modelo conceitual de uma plataforma descentralizada de armazenamento de

registros médicos usando IPFS e Blockchain, que pode oferecer vantagens como segurança, privacidade, disponibilidade, escalabilidade e interoperabilidade.

[3] Kumar, S., Bharti, A. K., & Amin, R. (2021). Secure and Scalable Decentralized Supply Chain Management Using Blockchain and IPFS. IEEE Communications Magazine, 59(10), 68-74.

O artigo propõe uma estrutura coletiva usando IPFS, contrato inteligente Ethereum e organização de repositório descentralizado para automatizar o processo e a troca de dados entre as partes interessadas na cadeia de suprimentos. O artigo apresenta os desafios e as vantagens da aplicação da blockchain na gestão da cadeia de suprimentos, bem como um estudo de caso de um sistema de rastreamento de produtos farmacêuticos usando a estrutura proposta. O artigo também avalia o desempenho, a segurança e a escalabilidade do sistema.

2. Requisitos

2.1. Requisitos de usuários

- Registro de paciente: O sistema deve permitir o registro de novos pacientes, coletando informações como nome, data de nascimento, gênero, endereço e outras informações relevantes para a identificação do paciente.

- Registro de histórico médico: O sistema deve permitir que os médicos registrem o histórico médico de um paciente, incluindo diagnósticos, tratamentos anteriores, alergias, cirurgias realizadas, medicamentos prescritos e outras informações médicas relevantes.

- Compartilhamento controlado de informações: O sistema deve permitir que os pacientes compartilhem suas informações médicas com outros profissionais de saúde ou instituições de forma controlada, ou seja, o paciente deve ter a capacidade de definir quais informações podem ser compartilhadas e com quem.

2.2. Requisitos funcionais

- Armazenamento descentralizado: O sistema deve utilizar a blockchain para armazenar os registros médicos de forma descentralizada, garantindo a imutabilidade e a segurança dos dados. Além disso, os arquivos médicos devem ser armazenados no IPFS, que oferece um sistema de armazenamento distribuído e resiliente.

- Registro/Auditoria de transações: Todas as transações envolvendo o histórico médico devem ser registradas na blockchain, incluindo a criação, modificação e compartilhamento de registros. Isso garante a transparência e a rastreabilidade das ações realizadas no sistema.

- Acesso Controlado: O acesso às informações médicas deve ser estritamente controlado, permitindo que apenas profissionais de saúde autorizados e pacientes tenham permissão para visualizar e atualizar os registros.

2.3 Requisitos não-funcionais

- Tempo de resposta: O sistema deve responder às solicitações do usuário de forma rápida e eficiente, garantindo tempos de resposta aceitáveis. O uso da off chain busca garantir que essa latência seja baixa.

- Manutenibilidade: A solução deve ser de fácil manutenção e atualização, permitindo correções de bugs, melhorias e expansões futuras.

- Padrões e conformidade técnica: O sistema deve aderir a padrões técnicos estabelecidos, como os padrões do Hyperledger para blockchain, garantindo a compatibilidade com outras soluções e facilitando a integração.

- Criptografia dos dados: No contexto do MedVault, a criptografia é um aspecto crucial da segurança das informações médicas sensíveis. Ela ajuda a proteger os dados de serem lidos ou alterados por pessoas não autorizadas. Portanto, a criptografia é um aspecto não funcional que contribui para a proteção das informações, mas não determina diretamente o que o sistema faz.

3. Fundamentos de SD relacionados ao projeto

3.1. Princípios de sistemas distribuídos

- Transparência: Um SD deve oferecer aos usuários e aplicativos a ilusão de que estão lidando com um único sistema de computação, em vez de várias máquinas separadas. No nosso caso, o usuário e os médicos possuem a impressão de que estão apenas se comunicando com um servidor qualquer, enquanto na verdade estão fazendo requisições para ler/modificar blocos específicos armazenados em uma blockchain descentralizada.

- Segurança: A segurança dos dados e recursos em um sistema distribuído é um princípio fundamental. Mecanismos de autenticação, autorização e criptografia que serão implementados através do Hyperledger irão garantir a segurança dos dados.

- Escalabilidade: A capacidade de um sistema distribuído de lidar com um aumento na carga de trabalho ou no número de usuários é essencial. Em nosso projeto, fizemos uso do Hyperledger Fabric, o qual utiliza uma arquitetura modular que permite a criação de canais privados e a divisão do processamento de transações entre diferentes nós da rede. Essa abordagem, conhecida como sharding ou particionamento, pode aumentar a capacidade de processamento da rede, permitindo a execução simultânea de transações em diferentes partes da blockchain. Além disso, o IPFS implementa uma estratégia de cache local, em que os nós armazenam temporariamente os arquivos acessados recentemente. Isso ajuda a melhorar a velocidade e eficiência do acesso aos arquivos, uma vez que eles podem ser servidos a partir do cache, evitando a necessidade de buscá-los na rede novamente. Além disso, a solução de armazenamento off-chain acarreta no uso da blockchain apenas para registrar referências ou hashes que comprovem a autenticidade e integridade dos dados no IPFS, evitando a sobrecarga da blockchain com grandes arquivos médicos, permitindo um maior grau de escalabilidade. Com todos esses fatores, alcançamos um maior nível de escalabilidade visto que o servidor fica o menos sobrecarregado possível graças às tecnologias utilizadas.

- Consistência: Em sistemas distribuídos, onde os dados são replicados e podem ser acessados por várias entidades concorrentes, é importante garantir a consistência dos dados. Em nosso caso, temos o IPFS usando o hash dos arquivos para verificar a integridade e garantir que eles não tenham sido corrompidos ou modificados (já que possuem um hash diferente para cada alteração), além do algoritmo de consenso do Hyperledger Proof of Elapsed Time (PoET), que não garante a consistência dos dados, mas trabalha em conjunto com outros componentes para garantir que a rede concorde sobre a ordem das transações. Portanto, a consistência dos dados

é mantida por meio de uma combinação do PoET com alguns mecanismos de validação de transações implementados e pelas e arquiteturas de armazenamento. Esses componentes trabalham juntos para garantir que apenas transações válidas sejam registradas no ledger e que o estado do ledger seja atualizado de maneira confiável e consistente.

3.2. Fundamentos de arquiteturas de sistemas distribuídos e dos estilos arquiteturais

- Peer-to-peer: Arquitetura de redes em que cada nó coopera entre si para prover serviços um ao outro, sem a necessidade a priori de um servidor central. Todos os pares são clientes e servidores. No projeto, os arquivos são divididos em blocos no IPFS e distribuídos no Hyperledger Sawtooth por meio de um hash criptográfico.

- Blockchain: Tecnologia que permite o armazenamento de blocos interligados em uma cadeia. Os dados são cronologicamente consistentes, pois não é possível excluir ou modificar a cadeia sem o consenso da rede. O projeto faz a utilização do Hyperledger Sawtooth.

3.3. Fundamentos de paradigmas de comunicação em sistemas distribuídos

- Comunicação baseada em troca de mensagens: Nesse paradigma, os componentes de um sistema distribuído se comunicam trocando mensagens. Cada componente pode enviar uma mensagem para outro componente, que então pode processar a mensagem e responder com uma mensagem de volta. Esse paradigma é geralmente implementado usando protocolos de comunicação, como o Transmission Control Protocol/Internet Protocol (TCP/IP) ou o User Datagram Protocol (UDP).

3.4. Robustez em sistemas distribuídos: nomeação, coordenação, consenso, consistência e replicação, e tolerância a falhas

- Nomeação: O Hyperledger Sawtooth utiliza o conceito de endereços de recursos (Resource Addresses) para nomear entidades dentro do sistema. Esses endereços são gerados de maneira determinística a partir dos dados da transação, garantindo a unicidade e consistência dos identificadores. A nomeação adequada e a garantia de unicidade são importantes para evitar conflitos e assegurar a integridade dos dados.

- Coordenação: A coordenação em um sistema Hyperledger Sawtooth é facilitada pelo mecanismo de validação de transações distribuídas. Cada transação é submetida para validação em diferentes nós, e um algoritmo de consenso é usado para garantir que os nós cheguem a um acordo sobre a validade e a ordem das transações. Isso requer coordenação eficiente entre nós para sincronizar e validar as transações de forma consistente.

- Consenso: O Hyperledger Sawtooth utiliza um mecanismo de consenso modular, o que significa que diferentes algoritmos de consenso podem ser implementados e escolhidos de acordo com as necessidades do sistema. Algoritmos de consenso como o Proof of Elapsed Time (PoET) e o Practical Byzantine Fault Tolerance (PBFT) são comumente usados no Sawtooth para alcançar um consenso confiável e resistente a falhas, mesmo em ambientes distribuídos adversos.

- Consistência e replicação: A consistência dos dados é garantida no Hyperledger Sawtooth por meio de um modelo de transações em estado de

finalização (Transaction Family State). Cada transação atualiza o estado do blockchain de maneira atômica, garantindo que as operações ocorram de forma consistente e correta. Além disso, a replicação dos dados em vários nós aumenta a disponibilidade e a resistência a falhas do sistema, permitindo a continuidade do serviço mesmo quando alguns nós estão indisponíveis.

- Tolerância a falhas: O Hyperledger Sawtooth é projetado para ser tolerante a falhas, tanto de nós quanto de redes. Os nós do Sawtooth possuem mecanismos de detecção de falhas e são capazes de se recuperar e se reconfigurar em caso de falhas. Além disso, o Sawtooth possui recursos para lidar com nós maliciosos ou desonestos, garantindo a segurança e a confiabilidade do sistema.

4. Resultados

4.1 Design arquitetural

O design arquitetural foi descrito utilizando o seguinte Diagrama. Nele podemos observar alguns componentes do sistema como o Hyperledger Sawtooth, o Web Service, o Proxy e o sistema de arquivos IPFS.

Basicamente, o Web Service funciona como uma interface para facilitar a comunicação do sistema com seus usuários. Essa interface fará a comunicação com o Proxy, que será responsável pela comunicação com o banco de dados, com o IPFS e a blockchain.

Inicialmente o paciente poderá adicionar um arquivo no sistema por meio do web service. O proxy ficará responsável por fazer o upload desse arquivo criptografado no IPFS e irá fornecer as informações necessárias a blockchain para criar um

request, no caso, hashs e outras devidas informações que são organizadas como forma de endereçamento dos arquivos no InterPlanetary File System.

Ficará a cargo do banco de dados o armazenamento do histórico de concessões e informações sobre os usuários, tais dados irão garantir a interação com a blockchain. Por meio dessa interação, a blockchain irá verificar se o usuário que está realizando a requisição tem permissão concedida para acessar o arquivo. Em caso positivo, o Hyperledger Sawtooth irá enviar ao proxy o link de endereçamento do arquivo no IPFS. Em posse dessas informações, o proxy irá descriptografar o link de endereçamento, recuperar o arquivo médico no InterPlanetary File System e retorná-lo ao usuário por meio do web service.

4.2 Design dos dados

Os tipos de dados criados para o funcionamento do sistema, de maneira geral, foram:

Doctor: Entidade que possui os atributos CPF, name e type.

Patient: Entidade que possui os atributos CPF, name, records e type.

Records, que são as formas de registro, possuem: id_record, title, bundle_hash e requests.

Requests são requisições feitas pelo médico à blockchain, possuem os atributos: Doctor_cpf, request_id e request_status.

Os Patients vão conter Records, que são registros médicos que contém, além das informações do paciente, também a lista de Requests com dados sobre solicitações feitas anteriormente. Esses Records poderão ser acessados pelo médico assim que o paciente fornecer os dados necessários para permissão de sua visualização.

5. Limitações, trabalhos futuros e perspectivas do Projeto

Neste capítulo são discutidas as limitações do estudo e as sugestões para futuros estudos.

5.1 Limitações

Este projeto decorreu ao longo de um semestre e cada integrante do grupo possuía mais de uma atividade (fora o projeto). O curto espaço de tempo somada a outras atividades (trabalho e extracurriculares) prejudicou o desenvolvimento, visto que o tema e as estruturas trabalhadas eram de grande complexidade. Consequentemente, não conseguimos integrar a proxy à blockchain.

5.2 Trabalhos futuros

Ao longo do desenvolvimento deste projeto identificaram-se questões correlatas que permitiriam o desenvolvimento de outros estudos para ampliar o entendimento do fenômeno estudado.

No caso, durante o projeto surgiu a ideia de se buscar a eliminação do uso de uma estrutura centralizada para armazenar os tokens de acesso