HEALTHCHAIN: UM SMART CONTRACT PARA ARMAZENAMENTO E GERENCIAMENTO DE DADOS MÉDICOS DE PACIENTES

Victor de Souza Mendes
Instituto Federal da Bahia - Campus Salvador
Salvador – Bahia – Brasil
victor.me@hotmail.com

Allan Edgard Silva Freitas
Instituto Federal da Bahia - Campus Salvador
Salvador – Bahia – Brasil
allanedgard@gmail.com

***Abstract*—This work proposes a reliable solution for the storage and control of medical data for users of the health system. The objective of this work is to maintain the data in a form that can be shared securely between the health institutions that have permission to access this data. Contrary to what happens in the financial environment, for example, the regulatory agents present in the Brazilian medical system do not have effective solutions to make available and manage their data on a large scale. Each health agent stores its data independently, without sharing information and in different standards. Users end up having their medical history separated, a small part in each institution, and never get to see it in a complete and uniform way. Our proposal is that with the use of a blockchain and the Healthchain smart contract, healthcare system users can use a reliable, scalable and highly available medical system.**

Keywords: patient medical data, healthcare system users, healthcare institutions, control, storage.

***Resumo*—Este trabalho propõe uma solução confiável para o armazenamento e controle dos dados médicos para usuários do sistema de saúde. O objetivo deste trabalho é manter os dados de forma que possam ser compartilhados com segurança entre as instituições de saúde que têm permissão para acessar esses dados. Ao contrário do que ocorre no ambiente financeiro, por exemplo, os agentes reguladores presentes no sistema médico brasileiro não possuem soluções efetivas para disponibilizar e gerenciar seus dados em larga escala. Cada agente de saúde armazena seus dados de forma independente, sem compartilhamento de informações e em padrões diferentes. Os usuários acabam tendo seu histórico médico separado, uma pequena parte em cada instituição, e nunca chegam a vê-lo de forma completa e uniforme. Nossa proposta é de que com o uso de uma blockchain e do contrato inteligente Healthchain, os usuários do sistema de saúde possam utilizar um sistema médico confiável, escalável e altamente disponível.**

Palavras-chave: Dados médicos do paciente, Usuários do sistema de saúde, Instituições de saúde, Controle, Armazenamento.

I. INTRODUÇÃO

Desde o surgimento dos computadores a sociedade vem automatizando seus processos. Antes, as atividades costumavam ser custosas do ponto de vista de tempo e de recursos. Grande parte do trabalho manual passou a ser desempenhado e processado por máquinas, economizando tempo e recursos.

Seguindo o ritmo, a medicina incorporou inovações tecnológicas importantes, sobretudo acerca do diagnóstico. Municiados por aparelhos mais eficazes, médicos conseguiram chegar a conclusões que antes eram praticamente impossíveis. [13].

Porém, o avanço tecnológico na área de saúde como um todo, não se deu de forma igual. O manuseio dos dados de usuários é um dos pontos que possuem uma certa carência, muitas vezes armazenados apenas em papéis, não possuem um sistema de compartilhamento eficaz e seguro. Exemplo disso é um cenário onde um usuário faz uma consulta na instituição de saúde X e depois em uma Y, informações extraídas durante a primeira consulta não podem ser compartilhadas entre as duas instituições de forma rápida, segura e automática, muitas das vezes o usuário precisa levar esses dados fisicamente a segunda instituição, acarretando sérios problemas. Dentre eles, caso o usuário esteja inconsciente e a instituição médica responsável precise dos dados dele, de forma instantânea, como seria possível esse compartilhamento? A demora para receber determinadas informações importantes, pode impactar diretamente no início do tratamento.

Os dados médicos de um usuário do sistema de saúde, são dados extremamente sensíveis, não devem ser alterados por qualquer pessoa e muito menos perdidos. Em situações extremas, onde uma tomada de decisão precisa ser baseada em informações seguras, a vida de um usuário pode estar em jogo. Essas informações precisam estar disponíveis instantaneamente e podem fazer toda a diferença no futuro desse usuário.

Visto isso, o presente trabalho propõe a criação de um prontuário eletrônico de usuários, que será armazenado de forma segura e distribuída na rede Ethereum.

Os dados médicos estarão sob o controle do próprio usuário, que desta forma passará a ter autonomia sob suas próprias informações, decidindo quem pode visualizar e podendo compartilhar com qualquer pessoa ou instituição que precise dessas informações, além de possuir um histórico médico completo para cada usuário, disponibilizado na rede.

II. FUNDAMENTAÇÃO TEÓRICA

A. Prontuários Médicos

Segundo o Conselho Federal de Medicina (CFM), o prontuário médico é um documento elaborado pelo médico e é uma ferramenta fundamental para o seu trabalho. Nesse documento constam todos os dados relativos ao paciente, como:

- Histórico familiar
- Anamnese
- Descrição
- Evolução de sintomas
- Exames
- Tratamento e prescrição

Sua elaboração pode ser feita em um consultório ou em um hospital e possui informações valiosas para facilitar a assistência do médico ao paciente. Um ponto muito importante é que esse documento pertence ao paciente, que tem total direito de acesso. O médico e a instituição de saúde são responsáveis apenas por elaborar esses documentos e guardar de forma segura. No passado, o acesso ao prontuário era exclusivo do médico, porém atualmente esses documentos são elaborados de forma multidisciplinar. Dados fornecidos por enfermeiros, fisioterapeutas e nutricionistas, por exemplo, são de grande importância [11].

O próprio Conselho Federal de Medicina publicou em 2002 uma resolução que estabelece que prontuários médicos de papel precisam ser preservados por no mínimo 20 anos a partir do último registro. Contudo, seguindo a evolução tecnológica, a tendência é que esses registros em papel sejam substituídos por prontuários eletrônicos, e cada paciente possa carregar seu prontuário pessoal. Dessa forma, as informações são acrescentadas a cada atendimento e servirá para troca de informações entre médicos e instituições distintas [11].

Atualmente, segundo levantamento TIC Saúde 2021, feito pelo centro Regional de Estudos para o Desenvolvimento da Sociedade da Informação (Cetic), **88%** das instituições de saúde possuem algum tipo de sistema eletrônico para registros das informações dos usuários. Separadamente, na rede pública esse percentual chega a **74%** enquanto na rede privada está mais avançado, e chega na marca de **91%** [23].

B. Blockchain

1) Um pouco da história: Ao contrário do que muitos pensam, a tecnologia não foi criada por Satoshi Nakamoto, ela foi descrita pela primeira vez em 1991 pelos pesquisadores Stuart Haber e W. Scott Stornetta. Eles tinham o objetivo de criar uma solução computacional prática para gerenciar documentos digitais, de modo que não pudessem ser adulterados. Então, desenvolveram uma aplicação, usando o conceito de cadeia de blocos, criptograficamente protegidos, para armazenar os documentos com carimbo de data e hora. Em 1992, eles adicionaram na blockchain o conceito de Árvore de Merkle, o que tornou o projeto mais eficiente, uma vez que vários documentos puderam ser coletados em um único bloco. Contudo, o projeto não foi utilizado e sua patente expirou em 2004 [17].

No mesmo ano, o cientista da computação Hal Finney desenvolveu um protótipo para dinheiro digital, chamado Reusable Proof of Work (RPoW). Foi um grande avanço na história das criptomoedas, uma vez que resolveu o problema do gasto duplo. O sistema RPoW funcionou recebendo em troca, um token de prova de trabalho, baseado em HashCash. Esse token era do tipo não fungível, era assinado por RSA e poderia ser transferido de pessoa para pessoa. Ele era armazenado em um servidor confiável, que foi projetado para que usuários em todo o mundo pudessem verificar sua integridade em tempo real [17].

Apenas em 2008, Satoshi Nakamoto apresentou uma criptomoeda denominada bitcoin, cujas transações eram mantidas de forma distribuída por meio da blockchain, sendo que cada bloco de transações é submetido a rede e aceito por meio do algoritmo Proof of Work. A partir deste trabalho disseminou-se o uso da blockchain. [20]. Em seu projeto ele aprimorou aspectos indispensáveis para o futuro da tecnologia. Os blocos agora podem ser adicionados à cadeia, sem a necessidade de serem assinados por partes confiáveis. É utilizada uma rede ponto a ponto para registro de data e hora e verificação de cada troca. Dessa forma a cadeia de blocos pode ser gerenciada pela própria rede, sem a necessidade de uma autoridade central confiável. A tecnologia evoluiu tanto com esses aprimoramentos que se tornou a espinha dorsal das criptomoedas. O trabalho de Nakamoto alcançou fama mundial, na sequência muitas outras redes baseadas em blockchain foram desenvolvidas, todas com peculiaridades para atender cenários diferentes mas usando o Bitcoin como referência principal [17].

Dessa forma, a tecnologia evoluiu bastante dentro da área financeira, resolvendo problemas, como por exemplo, do gasto duplo. O aumento do uso dessa tecnologia pode ser expressado se analisado o crescimento da blockchain do bitcoin. Estima-se que atualmente ela cresce 58GB por ano, e seu tamanho total já ultrapassava a marca dos 300GB em 2020 [8].

A medida que a tecnologia vem se provando e sendo conhecida cada vez mais, o interesse de outras áreas também aumentou bastante, fazendo com que a blockchain venha sendo pulverizada nas mais diversas esferas da sociedade [1]. Essa tecnologia tem o potencial de mudar drasticamente os mais diversos setores e pode ser considerada uma das ferramentas que fazem parte da atual quarta revolução industrial [4]. O grau de adesão da blockchain a um determinado setor, depende das necessidades daquela área. Se determinada área, por exemplo, necessita manter seus dados íntegros e seguros, essa será uma boa área para uma aplicação blockchain, uma vez que esses são um dos benefícios inerentes da tecnologia [1], desse modo a blockchain vem sendo desenvolvida também em áreas como: saúde, cadeia de suprimentos, indústria de alimentos entre outras.

Em 2020, a Forbes fez uma lista com as 50 maiores empresas que adotam a tecnologia. Evidenciando a expansão nas mais diversas áreas, desde o rastreamento de créditos de carbono pelo segundo maior banco do mundo na China, até o pagamento de combustível, usando criptomoedas na Alemanha [12].

O desenvolvimento da tecnologia e sua expansão para as mais diversas áreas pode ser dividido em três estágios principais:

Figura 1: [17]

- **Blockchain 1.0:** Durante esse período a blockchain estava diretamente ligada ao setor financeiro, mais especificamente as moedas virtuais, sendo o bitcoin a primeira e mais utilizada aplicação dessa tecnologia. As criptomoedas trouxeram com elas novas possibilidades ao setor financeiro, podendo reduzir muitos dos custos causados pelas moedas físicas, como os custos de circulação. Durante esse período foi produzidos muitos aplicativos, incluindo o Bitcoin. A maioria desses aplicativos eram criptomoedas, e eram basicamente usadas para pequenos pagamentos, câmbio, jogos de azar e lavagem de dinheiro [27].
- **Blockchain 2.0:** Alguns fatores de extrema importância surgiram nesse estágio da tecnologia. Foi durante esse período que a blockchain incluiu os contratos inteligentes, propriedades inteligentes, aplicativos descentralizados(Dapps), organizações autônomas descentralizadas(DAOs) e corporações autônomas descentralizadas(DACs). Aliado a esses novos fatores, a blockchain 2.0, pode ser entendida como o momento no qual a tecnologia se espalhou dentro do setor financeiro, começou a ser usada para negociação de títulos, finanças da cadeia de suprimentos, instrumentos bancários, compensação de pagamentos, anti-falsificação, estabelecimento de sistemas de crédito e seguro mútuo. O setor financeiro foi uma área propícia para o desenvolvimento da tecnologia, por exigir altos níveis de segurança e integridade de dados, uma vez que essas são vantagens inerentes de aplicativos blockchain. A maior contribuição dessa fase foi justamente o uso de contratos inteligentes para interromper os sistemas tradicionais de moeda e pagamentos. A integração da blockchain com a tecnologia de contratos inteligentes vem ganhando muita notoriedade, grandes organizações estão gastando cada vez mais esforço, dinheiro e estudos. Recentemente, Ethereum, Codius e Hyperledger criaram linguagens de contrato programável e infraestrutura executável para implementar contratos inteligentes [27].
- **Blockchain 3.0:** Finalmente a blockchain 3.0 é a expansão da tecnologia para outras áreas além de moedas e finanças, como saúde, governo, ciência, cultura e artes. A blockchain 3.0 possui contratos inteligentes mais avançados, tendo como objetivo estabelecer uma unidade organizacional que faz e está sujeita às suas próprias leis e que funciona com alto grau de autonomia. Esse estágio visa a popularização da tecnologia nos aspectos de regulação e governança de sua descentralização na sociedade. A blockchain 3.0 trouxe a combinação da blockchain com os tokens, muito conhecidos e utilizados graças a Ethereum com sua padronização ERC20. Com base nesse padrão qualquer pessoa pode criar seu próprio token na rede Ethereum e este token pode representar qualquer direito ou valor. Os tokens podem validar virtualmente qualquer direito que exista na sociedade, incluindo, selos, identidade pessoal, diplomas acadêmicos, moedas, recibos, chaves, ingressos para eventos, ponto de descontos, cupons, ações e títulos [27].

2) O que é uma Blockchain: Uma blockchain pode ser descrita como um tipo de banco de dados. Um banco de dados diferente dos legados, possui um livro-razão[1] distribuído e é compartilhado entre os nós de uma determinada rede de computadores. Se tornou uma tecnologia notória por seu papel crucial em sistemas de criptomoedas, como o bitcoin, por manter as transações seguras e independentes de um órgão centralizador confiável. Diferente dos banco de dados comuns, onde as informações são armazenadas em tabelas, a blockchain agrupa seus dados em blocos. Esses blocos possuem limitação de armazenamento, e quando um bloco é completamente preenchido ele é fechado e vinculado ao bloco anterior por meio de criptografia, formando assim uma cadeia de blocos. Todas as informações desse bloco recém adicionado, são compiladas em um bloco subsequente, que por sua vez, quando for preenchido também será adicionado à cadeia. Uma vez adicionado, o bloco não pode ser mais alterado. Dessa forma

[1]O livro-razão é um registro de escrituração, que tem a finalidade de coletar dados cronológicos de todas as transações e organizá-las por contas individualizadas [25].

as transações são registradas permanentemente e podem ser visualizadas por qualquer pessoa [16].

3) Objetivos: O principal objetivo da blockchain é armazenar informações digitais de forma distribuída e de modo que não possam ser alteradas. Servindo de base para livros imutáveis ou registros de transações, que não podem ser adulteradas, excluídas ou destruídas. Com isso, também pode ser considerada como uma tecnologia de contabilidade distribuída (DLT, do inglês distributed ledger technology) [16].

Para a blockchain conseguir atingir esse objetivo e servir de banco de dados para informações vitais, alguns aspectos são indispensáveis nesse processo:

- **Descentralização:** A tecnologia não necessita de uma autoridade central reguladora, a rede possui regras padronizadas sobre como cada nó interage com a blockchain, garantindo que todas as transações sejam validadas e todas as transações válidas sejam adicionadas uma a uma [17].
  Em paralelo, em um cenário fictício, uma empresa possui seu banco de dados totalmente hospedado e gerenciado por ela mesma, em um hub de servidores que está localizado na sua sede. Todos os computadores, dentro da mesma sala. Essa é uma falha de segurança sem precedentes, o que acontece se faltar energia? E se a conexão for cortada? E se em alguma hipótese, acontecer um incêndio? Em todos os casos, todos os dados podem ficar indisponíveis por um tempo, podem ser corrompidos, ou até perdidos para sempre.
  A descentralização dessas informações permite que os dados sejam espalhadas dentre os nós da rede em vários locais físicos diferentes. Além de criar redundância, mantém a fidelidade dos dados armazenados. Se em algum cenário, um agente mal intencionado alterar um registro em uma instância do banco de dados, as outras instâncias não serão alteradas e poderão recusar a informação tendenciosa. Este sistema ajuda a estabelecer uma ordem exata e transparente dos eventos e nenhum nó único da rede consegue alterar as informações contidas na blockchain [16].
- **Transparência:** Todas as transações adicionadas à rede podem ser facilmente auditadas. Seja por meio de um nó pessoal, ou usando exploradores de blockchain, qualquer pessoa pode rastrear o bitcoin onde quer que ele vá. Se, por um exemplo, seus bitcoins forem roubados, o hacker poderá se manter anônimo, mas se ele mover ou gastar em algum lugar, isso seria conhecido [16].
- **Segurança:** Cada bloco dentro da rede é ligado ao anterior através do seu hash, ou seja, se um determinado bloco na cadeia for alterado, todos os subsequentes também precisaram ser alterados. Em blockchains públicas baseadas em criptomoedas, tipicamente, devido a descentralização do processo, um agente mal intencionado, se quiser adulterar alguma informação precisará assumir o controle de pelo menos 51% dos nós ligados à rede para sua modificação ser aceita. Tal ataque também exigiria uma quantidade imensa de dinheiro e recursos, pois seria necessário refazer todos os blocos já que agora os carimbos de data e hora além dos hashs seriam diferentes [16].

4) Como tudo funciona: É inerente da tecnologia, aspectos de descentralização, segurança e transparência. Mas como todos esses aspectos são atingidos na prática?

A arquitetura tradicional e bastante difundida da World Wide Web, usa em sua maioria, uma rede cliente-servidor. O servidor é o responsável por manter todas as informações necessárias em um único local, além de atualizar, adicionar, excluir e servir essas informações para os agentes autorizados a receber esses dados.

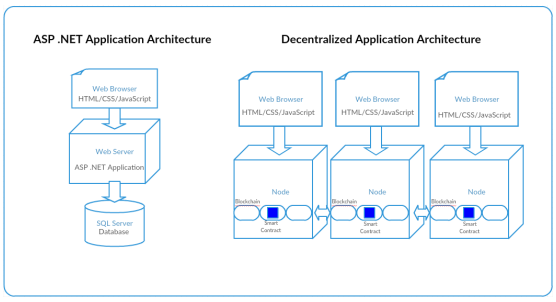

Figura 2: [24]

Já no caso da arquitetura blockchain, ela funciona tendo como base uma rede ponto a ponto (P2P), onde cada participante da rede mantém, aprova e atualiza novas entradas. O sistema é controlado, dessa forma, não só por participantes individuais, mas sim pelo conjunto de todos os agentes presentes na rede. Cada membro valida todos os registros e procedimentos, tornando os dados confiáveis e seguros. Dessa forma, partes que não necessariamente confiam uma na outra, conseguem chegar a um consenso comum [18]. Existem configurações públicas e privadas de blockchains. Em blockchains públicas normalmente há a medicação de criptomoedas e incentivos para proposição de blocos e alcance do consenso, em blockchains privadas, os membros são limitados e conhecidos, e normalmente não são utilizadas criptomoedas, o consenso ocorre sem incentivos adicionais na rede. Ao longo deste texto, assumimos o uso de blockchains públicas.

Portanto, a blockchain é representada por uma lista de blocos com transações ordenadas. Para que tudo funcione corretamente, os principais componentes da rede são:

- **Nó:** Pode ser um usuário ou um computador, dentro da arquitetura blockchain e possui uma cópia independente de todo livro razão.
- **Transação:** Menor bloco de construção, dentro de um sistema blockchain. Representa a mudança de estado em si que está sendo solicitada.

- **Mempool:** É como uma área de espera para as transações, antes de serem adicionadas a um bloco.
- **Bloco:** Armazena as transações e outras informações vitais, como hash do bloco anterior e timestamp da sua criação.
- **Cadeia:** A cadeia é formada por uma sequência de blocos, ordenados por ordem cronológica.
- **Mineiros:** Nós específicos, responsáveis por executar o processo de verificação do bloco antes dele ser adicionado à cadeia já existente.
- **Consenso:** Conjunto de regras a serem respeitadas para que à maioria dos nós presentes na rede cheguem a um consenso comum.

Mas de que forma esses componentes interagem para que a blockchain funcione?

Visto que, uma rede blockchain nada mais é que um livro razão, distribuído e público, entre seus participantes, onde qualquer agente participante da rede pode adicionar transações à rede, como a própria rede conseguiria distinguir se a transação foi realmente enviada por seu agente original, ou por um terceiro mal intencionado? Isto é, como garantir que uma pessoa não está se passando por outra, para enviar transações para a rede? Para resolver esse cenário, as redes blockchains usam as assinaturas digitais para validarem se uma transação é íntegra ou se é fraudulenta. A blockchain utiliza criptografia assimétrica para que cada agente da rede possa assinar suas transações. Cada usuário ao entrar na rede recebe uma chave privada e uma chave pública que todos podem ver, as duas quando usadas em conjunto cria uma identidade digital segura para autenticar o usuário e permitir que ele consiga fazer suas transações [10].

E então, é assim que o fluxo começa, um usuário assina a transação que ele deseja fazer, e essa transação é enviada para uma estrutura conhecida como mempool, uma das partes mais importantes, por ser a porta de entrada para a rede. Esse componente é frequentemente negligenciado, e pouco citado. Se trata da área de preparação dinâmica na frente da blockchain, permite a ordenação das transações, priorização das taxas de transação e construção geral dos blocos. Basicamente, é um conjunto de estrutura de dados, na memória de um Nó, que é responsável por armazenar transações candidatas antes de serem escritas em um bloco.

Cada nó possui o seu próprio mempool, que tenta permanecer sincronizado com outros nós da rede, mas como a comunicação da rede nem sempre é confiável, cada nó tem um pool de transações ligeiramente, ou às vezes significativamente, diferente. Além disso, os nós possuem políticas diferentes em relação a quais transações aceitam, preço mínimo do gás e limites de tamanho da mempool, por exemplo, podem influenciar no processo. Dessa forma, toda transação presente na rede já passou por alguma mempool e um fluxo de transação típico inclui as seguinte etapas [5]:

- Um usuário inicia uma transação de um Dapp ou Wallet, enviando fundos para outra conta ou contrato, por exemplo.
- O usuário assina essa transação com sua carteira.
- A transação é enviada pela carteira para um nó, geralmente chamado de nó de gateway.
- Esse nó valida a transação e adiciona ao seu mempool.
- O nó transmite essa transação para os seus pares presentes na rede.
- Os pares, por sua vez, validarão e moverão a transação recebida para o seu próprio mempool. Depois transmitirão a transação para pares adicionais, replicando a transação por toda a rede.
- Os nós que são mineradores após receberem a transação, além de também executarem o mesmo processo dos demais nós, tentarão adicionar a transação a um bloco.

A essa altura, é muito importante entender a estrutura dos blocos, é nele onde estarão armazenadas todas as transações validas na rede. O cabeçalho de um bloco pode variar de blockchain para blockchain, porém, existem quatro campos que são bastante comuns e cada um é importante para o funcionamento adequado da rede [21]:

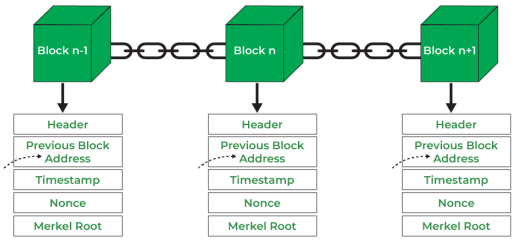

Figura 3: [14]

- **Hash do bloco anterior:** Esse valor é o que permite a ligação entre os blocos.
- **Timestamp:** Indica, aproximadamente, quando um bloco foi criado. Usado por contratos inteligentes para mensurar a taxa média de criação do bloco em relação ao valor alvo.
- **Raiz da transação:** Resume o conteúdo do corpo do bloco e ajuda a proteger a integridade das transações.
- **Nonce:** É um valor aleatório criado pelo criador do bloco. Usado no algoritmo de prova de trabalho, pelo seu minerador, para obter um valor de hash que seja aceito pela rede.

O corpo do bloco, por sua vez, é o responsável por armazenar as transações. A blockchain utiliza árvores Merkle para guardar as transações dentro dos blocos.

Para entender os próximos passos dentro desse fluxo, é necessário entender o que são funções hash criptográficas.

Uma função hash é qualquer função usada para mapear dados de entrada que possui um tamanho variável e retornar

uma saída de tamanho fixo. Ou seja, uma função hash pode ser usada para uma entrada de qualquer tamanho e o resultado será um hash de tamanho padronizado. Além disso, a função é determinística, para cada entrada distinta o resultado do hash será completamente diferente e a função quando aplicada a mesma entrada, não importando o número de tentativas, o resultado sempre será o mesmo. Esse aspecto é extremamente importante, pois permite a ”impressão digital”dos dados [7].

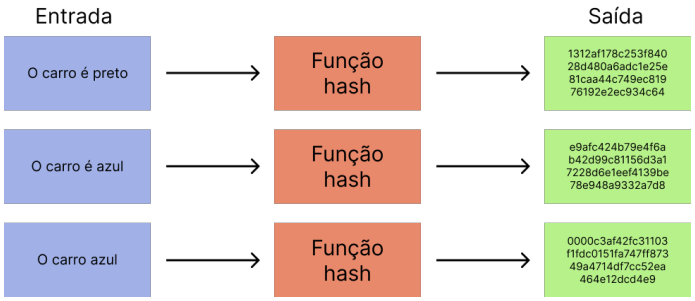

Figura 4: Funções hash

Se apenas uma pequena parte da entrada mudar, a saída também será completamente diferente. E como o resultado da função sempre tem o mesmo tamanho, grandes quantidades de dados podem ser identificadas utilizando o hash resultante. Em sistemas que gerenciam grandes quantidades de informações, como sistemas blockchain, os benefícios de poder armazenar e identificar dados com uma saída de tamanho fixo podem gerar grandes economias de armazenamento e processamento, ajudando a aumentar a eficiência do sistema [7].

C. Blockchain aplicada a saúde

Com a evolução tecnológica atual, usuários passaram a se acostumar com processos mais ágeis, digitais e de fácil acesso. Muitos problemas da área de saúde ainda ocorrem quanto a gestão dos dados médicos. Os sistemas legados, quando existem, são onerosos, lentos, muitas vezes inseguros e o usuário não possui nenhum controle sobre seus próprios dados e fluxos. Na maioria das vezes, o único acesso que o usuário possui a seus dados, ocorre por meio de papéis físicos, sejam eles, resultados de exames ou receitas médicas [3].

Armazenados nesses sistemas legados, os dados de saúde se tornam isolados e difíceis de serem compartilhados devido a falta de padronização dessas informações. Com isso, as partes que são interessadas nesses dados, precisam guardar esses registros por conta própria, se quiserem ter esses dados em mãos, não sendo possível nunca ter uma versão única da verdade [3].

Ao mesmo tempo que as informações fluem mais rápido pelo mundo, graças aos avanços tecnológico, a relação entre pacientes e médicos também estão mudando. Antes, mais paternalista, era comum pessoas e até famílias inteiras serem atendidas por um único médico durante toda a vida. Hoje em dia, os usuários estão reivindicando mais espaço e autoridade sobre suas intervenções médicas, é comum por exemplo, uma pessoa buscar mais de uma opinião médica para só depois tomar sua decisão. Com isso, cada vez mais os usuários não apenas podem, mas devem ter controle sobre seus próprios dados [3].

D. Ethereum

Descrito por Vitalik Buterin em 2013 e desenvolvido pelo Dr. Gavin Wood em abril de 2014 [19], o Ethereum é a segunda maior rede blockchain do mundo em capitalização de mercado, ficando atrás apenas do Bitcoin, e seu valor é estimado em mais de 540 bilhões de dólares [26]. O Ethereum é uma rede pública e sem permissão, qualquer pessoa pode baixar ou escrever algum software para se conectar à rede e começar a criar as transações, valida-las e minerar blocos sem a necessidade de fazer login ou se inscrever com qualquer outra organização. A rede é alimentada por Ether(ETH), que é a criptomoeda usada para pagar as transações e as taxas de gás. As taxas de gás são as taxas cobradas para o processamento de transações e a execução dos contratos inteligentes [26].

Além de toda a base da tecnologia blockchain, o maior diferencial trazido pelo Ethereum é a possibilidade de executar código através dos contratos inteligentes. Basicamente, o Ethereum é um blockchain com uma máquina de Turing determinística, embutida nele, e serve de base para criar aplicativos e organizações de maneira descentralizada, sem permissão e resistente à censura [9].

1) Máquina Virtual Ethereum: A máquina virtual ethereum (EVM, do inglês Ethereum Virtual Machine) é um computador canônico, cujo estado, todos os presentes na rede concordam. Cada participante da rede mantêm um cópia do estado desse computador e qualquer participante pode transmitir uma solicitação para que este computador execute cálculos arbitrários. Sempre que uma solicitação é transmitida, outros participantes na rede verificam, validam e executam o processamento. Essa execução muda o estado da EVM, que é confirmada e então propagada por toda a rede [9].

As solicitações de processamento são conhecidas como solicitações de transação. Todas as transações e o estado atual da EVM são armazenadas no blockchain, que por sua vez é armazenado e acordado por todos os nós [9].

Os mecanismos criptográficos da blockchain garantem que, a partir do momento que as transações sejam validadas e adicionadas a cadeia, elas não poderão ser adulteradas posteriormente. Os mesmos mecanismos também garantem que todas as transações sejam assinadas e executadas com as permissões apropriadas, ou seja, ninguém deve poder enviar ativos digitais de uma conta, exceto o próprio dono da conta [9].

2) Contratos Inteligentes: Os participantes da rede não escrevem um novo código toda vez que desejam solicitar um cálculo na EVM. Na verdade, os desenvolvedores de aplicativos carregam programas, que nada mais são que trechos de código reutilizáveis, para o estado da EVM. Os usuários, por sua vez, fazem solicitações para executar esses códigos com parâmetros variados [9].

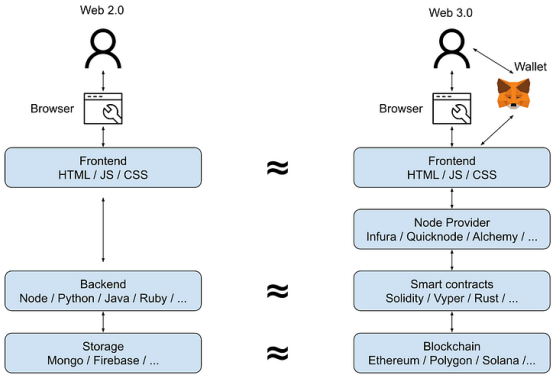

Figura 5: [15]

Por se tratar de uma rede aberta e democrática, qualquer desenvolvedor pode criar um contrato inteligente e torná-lo público na rede, usando blockchain como sua camada de dados, por uma taxa paga à rede. Depois do contrato disponibilizado na rede principal, qualquer usuário interessado, que possua o endereço do contrato poderá chama-lo para executar seu código, novamente por uma taxa paga à rede [9].

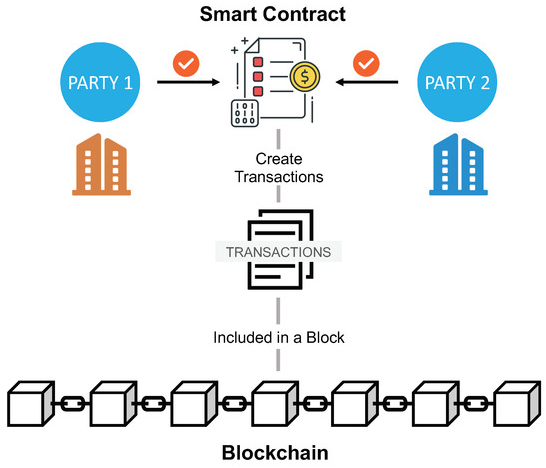

Figura 6: [28]

Dessa maneira, é por meio dos contratos inteligentes que, a rede Ethereum permite que os desenvolvedores criem e implantem aplicativos e serviços voltados para o usuário, arbitrariamente complexos, como: mercados, instrumentos financeiros, jogos ou um sistema de gerenciamento de dados médicos [9].

3) Solidity: O Solidity é um das linguagens disponíveis, de código aberto, para a construção de contratos inteligentes. Está entre os idiomas mais ativos e mantidos pela comunidade, além de possuir mais recursos que seus concorrentes, foi o escolhido para a criação do Healthchain. Algumas características do solidity são:

- Linguagem de alto nível orientada a objetos
- Linguagem de chaves, influenciada pelo C++
- Tipagem estática, o tipo da variável é conhecido durante a compilação
- Permite herança, um contrato pode estender outros contratos
- Permite o uso e criação de bibliotecas, para reutilização de código que pode ser chamado por diferentes contratos
- Suporta tipos complexos definidos pelo usuário

Existem outras linguagens que também podem ser utilizadas, cabe ao desenvolvedor escolher a que melhor se encaixa para ele. A maioria dessas linguagens, são relativamente amigáveis ao desenvolvedor. Se o programador tiver experiência com Python, provavelmente ele encontrará um idioma com sintaxe familiar [9].

No próximo tópico, discutiremos brevemente trabalhos relacionados que apresentam como a blockchain tem sido utilizada na saúde.

III. TRABALHOS RELACIONADOS

A. I. Kar. (2016). Estonian Citizens Will Soon Have the World's Most Hack-Proof Health-Care Records. [Online]. Available: http://qz. com/628889/this-eastern-european-country-is-moving-its-health-records-to-the-blockchain/

O trabalho fala sobre o perigo da exposição inerente dos dados pessoais, registros médicos e outros dados confidenciais por meio de hackers, mostrando o potencial da criação de uma rede dados baseados na tecnologia blockchain, dessa forma as informações permanecem seguras e invioláveis, ninguém pode mudá-lo sem deixar rastros e eventualmente, o público terá acesso a seus próprios registros de saúde e quaisquer alterações que sejam feitas nas mesmas.

B. W. Suberg. (2015). Factom's Latest Partnership Takes on US Health-care. [Online]. Available: http://cointelegraph.com/news/factoms-latest-partnership-takes-on-us-healthcare

O autor apresenta a parceria entre uma empresa de tecnologia e o maior provedor de serviços médicos dos EUA, a empresa fornecerá a tecnologia blockchain para garantir a integridade dos registros médicos e privacidade do paciente além de acrescentar a segurança e garantir a autenticidade de sequência de eventos através da criação de um software de segurança.

C. Conceição, A. F., Silva, F. S. C., Rocha, V., Locoro, A., and Barguil, J. M. M. (2018). Eletronic health records using blockchain technology.

O estudo mostra as diversas formas de aplicabilidade da criação de um software de armazenamento e segurança de dados de saúde utilizando a tecnologia blockchain. As informações anônimas e especificas acessíveis no banco de dados podem ajudar o controle de propagação de doenças transmissíveis e apoiar a notificação de epidemias.

D. Gordon J. W., Catalini C. (2018). Blockchain Technology for Healthcare: Facilitating the Transition to Patient-Driven Interoperability

O autor traz a visão da adaptação do foco interoperabilidade das instituições de saúde para a interoperabilidade orientada ao paciente. Em essência, a blockchain fornece uma estrutura de alto nível que permite ao paciente a interação com a segurança com várias partes interessadas, identificar-se em cada entidade e agregar seus dados de saúde de forma persistente. Traz também a importância da discussão de alguns pontos, dentre eles as considerações de privacidade e segurança, como exemplo, o aspecto anônimo, mas não privado, bem como questões de relacionadas ao gerenciamento de chaves do paciente e ao envolvimento do paciente.

E. Rede Nacional de Dados em Saúde

O governo brasileiro também já se mobilizou no sentido de melhorar o gerenciamento dos dados médico. A RNDS surgiu como um projeto piloto no estado de Alagoas, previsto pra ser iniciado em 2019. Porém, nesse mesmo período iniciou-se a pandemia do novo coronavírus e o projeto foi reorientado para receber e compartilhar informações relevantes para auxiliar no controle da situação de emergência da saúde pública [22].

Pressionado pela necessidade de controlar eficientemente a vacinação contra o vírus da Covid-19, a RNDS foi desenvolvida pelo governo Federal. A blockchain também é a base tecnológica da RNDS, mas só resolve alguns problemas. Se trata de uma rede privada, gerenciada e mantida pelo governo, ou seja os usuários continuam sem possuir poder nenhum sobre seus próprios dados. Por que o governo simplesmente não anonimizou os dados e usou uma rede já conhecida e testada mundialmente? pelo mesmo fato do governo ser contra as criptomoedas, nesse caso ele perderia o controle sobre esses dados e não teria vantagem com isto.

Em muitos aspectos esses trabalhos se relacionam com o Healthchain, cada um implementa a blockchain para resolver problemas no gerenciamento dos dados médicos, cada um com suas peculiaridades. Porém, o Healthchain tem a proposta de tratar os usuários como peça principal nesse cenário. Os dados serão abertos e anônimos, qualquer pessoa poderá ver os dados, mas não será possível fazer a relação entre dados e pessoas. Apenas o dono da conta Ethereum, e quem ele permitir, saberá qual conjunto de dados pertence a ele. Caberá apenas ao usuário decidir sobre o gerenciamento dos seus próprios dados.

IV. PROPOSTA

Esse projeto tem o objetivo de propôr uma solução para o gerenciamento de dados médicos. Buscando, por meio da tecnologia blockchain, resolver problemas relacionados a segurança, disponibilidade, compartilhamento e transparência dos dados.

Utilizando um contrato inteligente, é possível adicionar ao sistema de saúde um agente regulador, que no caso será a própria rede blockchain. Ao contrário do que ocorre no ambiente financeiro por exemplo, os agentes reguladores presentes no sistema médico brasileiro, não possuem soluções eficazes para disponibilizar e gerenciar seus dados em larga escala. Cada agente de saúde armazena seus dados de forma independente, sem compartilhamento de informações e em padrões distintos. Os usuários acabam tendo o seu histórico médico ”recortado”, uma pequena parte em cada instituição, e nunca consegue ver de forma integral e uniforme. O contrato Healthchain permitirá ao sistema médico ter uma solução confiável, escalável e altamente disponível. Além disso, fazendo outro paralelo com o sistema financeiro, assim como os bancos não serão os agentes reguladores centralizados, as instituições médicas também perderão essa responsabilidade. Passando dessa forma, o controle sobre os seus dados para os próprios usuários. Os usuários agora poderão ter acesso aos seus dados de forma integral, visualizar seu histórico completo e compartilhar com quem quiser. Também poderão vincular endereços confiáveis, para em casos de emergência, os usuários donos desse endereço consigam permitir que um agente médico visualize as informações de um terceiro, talvez em estado crítico.

Dessa maneira, as próximas seções serão dedicadas a explicar como o Healthchain foi construído.

A. Requisitos

De maneira geral, alguns requisitos precisaram ser cumpridos para que o objetivo final fosse alcançado, e também serviram como guia durante o desenvolvimento do contrato inteligente.

1) Requisitos Gerais:

- Será necessário construir um contrato inteligente compatível com a rede Ethereum
- Usuários poderão se conectar com o contrato inteligente disponível na rede Ethereum e criar sua própria ficha médica

- Médicos poderão se conectar com o contrato inteligente disponível na rede Ethereum e criar seu próprio perfil médico
- Os dados dos usuários serão anônimos, nenhum dado presente nas informações médicas serão vinculados a pessoa física
- Usuários poderão gerenciar as autorizações de acesso aos seus dados
- Médicos poderão, se autorizados, adicionar relatórios médicos aos registros de um usuário
- Os usuários poderão visualizar todas as suas informações médicas

2) Requisitos Específicos:

- O contrato inteligente será construído em solidity
- Apenas contas do tipo paciente poderão adicionar contatos confiáveis
- Apenas o usuário poderá gerenciar seus contatos confiáveis
- Apenas contas do tipo paciente poderá adicionar médicos confiáveis
- Apenas o usuário e seus contatos de confiança poderão gerenciar médicos confiáveis
- Apenas contas do tipo médico poderão adicionar criar relatórios médicos
- Apenas médicos autorizados podem criar relatórios médicos para um usuário

B. Estrutura

O Healthchain é baseado em três estruturas principais, pacientes, médicos e registros médicos. Cada um possui dados, limitações e recursos específicos dentro do sistema.

1) Paciente: Os pacientes representam os detentores dos dados médicos e os usuários do sistema, e o contrato armazena informações relevantes sobre sua saúde. As informações necessárias para a criação de um paciente são:

- Dono - o endereço ethereum dono daquela ficha médica
- Id - um identificador único para usuários
- Dia de aniversário - dia de nascimento do usuário
- Mês de aniversário - mês de nascimento do usuário
- Ano de aniversário - ano de nascimento do usuário
- País - país de moradia do usuário
- Estado - estado de moradia do usuário
- Cidade - cidade de moradia do usuário
- Notas importantes - alguma informação importante sobre a vida medica do usuário
- Doenças conhecidas - doenças conhecidas do usuário
- Alergias conhecidas - alergias conhecidas do usuário
- Tipo sanguíneo - tipo sanguíneo do usuário
- Gênero - gênero do usuário

```
// patient information
struct Patient{
    address owner;
    uint id;
    uint birthday;
    uint birth_month;
    uint birth_year;
    string country;
    string state;
    string city;
    string[] important_notes;
    string[] known_diseases;
    string[] known_allergies;
    BloodType blood_type;
    Gender gender;
}
```

Figura 7: Struct de pacientes

2) Médico: Os médicos são os responsáveis por criar relatórios médicos para os usuários, apenas médicos registrados podem adicionar informações médicas para um usuário. Ao contrário dos usuários eles são identificados e possuem informações pessoais como nome e CRM. As informações de um médico cadastrado são:

- Id - um identificador único para médicos
- Dono - o endereço ethereum representado pelo perfil médico
- Nome - o nome do médico
- CRM - conselho regional de medicina do médico
- Área - a área de atuação do médico
- Especialidade - a especialidade do médico

```
// doctor information
struct Doctor{
    uint id;
    address owner;
    string name;
    string crm;
    string area;
    string expertise;
}
```

Figura 8: Struct de médicos

3) Relatório Médico: Os relatórios se referem a uma interação médica entre um paciente e um médico. Apenas médicos cadastrados no contrato e autorizados pelo paciente podem criar relatórios médicos para esse paciente. Todos os relatórios de um usuário junto com sua ficha médica representam a sua vida médica.

- Paciente - identificador do usuário
- Médico - identificador do médico
- Sintomas - sintomas apresentados na consulta

- Diagnóstico - diagnóstico do cenário
- Prescrição - prescrição dada pelo médico
- Anotação - alguma anotação que o médico julgue importante

```
// medical reports from a medical appointment
struct MedicalReport{
    uint patient;
    uint doctor;
    string[] symptoms;
    string diagnostic;
    string prescription;
    string annotation;
}
```

Figura 9: Struct relatório médico

C. Funcionamento

Os pacientes são a estrutura principal desse ecossistema, é para eles que esse projeto foi construído, visando uma aplicação homogênea, transparente, segura, democrática e altamente disponível. Os usuários dentro do sistema são os reais detentores do poder sobre seus dados, eles possuem uma ficha médica e relatórios médicos. Relatórios médicos só podem ser criados por médicos que estejam autorizados para tal e usuários podem adicionar contatos confiáveis, que por sua vez, podem autorizar ou desautorizar um médico a criar relatórios médicos para a conta alvo.

A seguir, o esquema básico desse ecossistema:

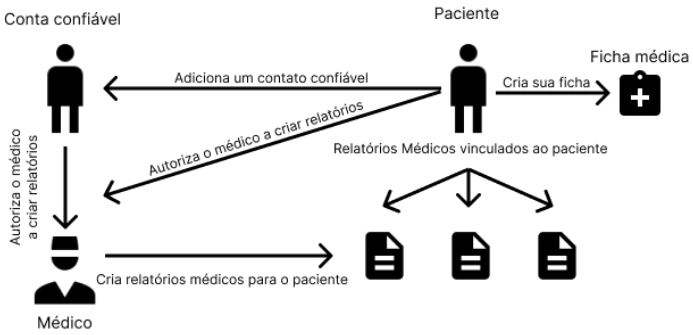

Figura 10: Esquema Healthchain

Dentro desse tópico serão explorados os mecanismos e funcionalidades de cada estrutura dentro do sistema, o que cada um pode fazer, o que não pode e suas limitações.

A ideia principal desse projeto é garantir que os dados médicos de um usuário realmente pertençam a ele, ou seja, o poder sobre esses dados deixaram de estar nas mãos dos órgãos médicos e de agentes centralizadores e passaram a estar nas mãos dos reais donos dessas informações.

Por se tratar de dados extremamente importantes e pessoais, eles precisam estar armazenados de forma segura, imutável e com alta disponibilidade. Sem depender de uma entidade central, já que os verdadeiros detentores do poder sobre seus dados são seus próprios donos. Para isso, foi escolhida a tecnologia blockchain. Como já mostrado nos tópicos anteriores, essa tecnologia dificulta muito que agentes mal intencionados consigam adulterar os dados já cadastrados no sistema. Com isso, o Healthchain foi construído visando ser disponibilizado dentro da rede principal do Ethereum, portanto, uma rede blockchain aberta em que todas as pessoas conectadas na internet possam visualizar e auditar todos os blocos e transações feitos utilizando esse contrato.

Uma das primeiras preocupações foram sobre a privacidade dos dados, uma vez que esse sistema está aberto em uma rede pública, como os dados de um paciente seriam privados a ponto de uma pessoa ou uma organização não conseguirem obter os dados e o histórico médico de um paciente específico?

A estratégia utilizada foi a anonimização dos dados, segundo a Lei Geral de Proteção de Dados (LGPD) [6], um dado anonimizado é aquele que, considerados os meio técnicos razoáveis no momento do tratamento, perde a possibilidade de associação, direta ou indireta, a um indivíduo. Com isso, a ficha médica de um usuário e os relatórios médicos não possuem nenhuma informação que permita associação entre esse dados e seu dono, como cpf, número de identidade ou nome.

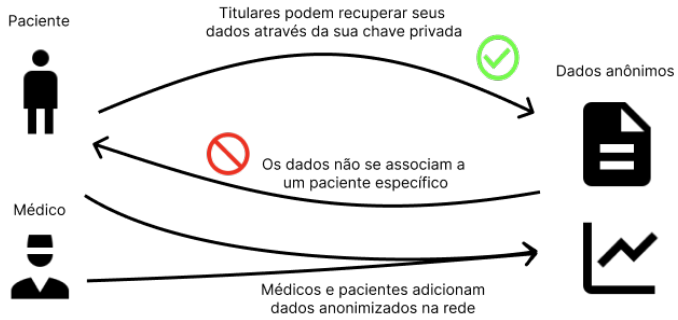

Figura 11: Anonimização dos dados

Os usuários criam suas fichas médicas e os médicos podem adicionar relatórios para eles. Todos esses registros são anonimizados, dessa forma, apenas usuários que possuem a chave privada que criou a ficha médica, consegue saber que aquele é um registro seu. Para todos os outros visualizadores da rede é um grande repositório de informações anônimas, mais à frente nesse documento será descrito como pesquisadores individuais, organizações de pesquisa e até o governo poderão utilizar esses dados para desenvolver ações, projetos e produtos.

Depois do contrato disponível na rede, usuários irão precisar de uma conta na rede Ethereum(uma chave pública e privada) e de fundos nessa conta para realizar transações. A primeira transação é criar sua ficha médica com suas informações básicas. A função ”create_medical_record”é chamada pelo

usuário usando sua conta Ethereum e os parâmetros necessários são:

```
// create medical records to one patient
function create_medical_record(
    uint _birthday,
    uint _birth_month,
    uint _birth_year,
    string memory _country,
    string memory _state,
    string memory _city,
    string[] memory _important_notes,
    string[] memory _known_diseases,
    string[] memory _known_allergies,
    uint _blood_type,
    uint _gender
) public returns (uint id){
```

Figura 12: Função create_medical_record

Notas importantes, doenças conhecidas e alergias conhecidas não são parâmetros obrigatórios, um usuário pode ter esse informação ou não. Porém, todos os outros parâmetros são obrigatórios, existe mais uma informação obrigatória, que é o endereço Ethereum de quem está criando a ficha médica mas esse parâmetro não precisa ser explicitado, o contrato inteligente consegue pegar o endereço de quem está enviando a transação e esse endereço obrigatoriamente será vinculado a ficha que está sendo criada.

```
200         Patient memory patient = Patient({
201             owner: msg.sender,
202             id: id,
203             birthday: _birthday,
204             birth_month: _birth_month,
205             birth_year: _birth_year,
206             country: _country,
207             state: _state,
208             city: _city,
209             important_notes: _important_notes,
210             known_diseases: _known_diseases,
211             known_allergies: _known_allergies,
212             blood_type: BloodType(_blood_type),
213             gender: Gender(_gender)
214         });
```

Figura 13: Criação de um paciente

Na linha 200 do código acima, é onde o endereço da conta que enviou a transação, é vinculada ao paciente que está sendo criado no contrato, dessa forma, uma conta Ethereum só é capaz de criar fichas médicas para o seu próprio endereço.

Depois da ficha criada, o usuário pode usar uma série de funções do contrato inteligente como:

- **Obter minha ficha médica (get_my_medical_record)**: Essa função retorna a ficha médica da conta que está invocando essa função do contrato, não requer parâmetros e é uma função do tipo view, que significa que não altera o estado da blockchain, apenas visualiza.

```
// get a medical record by sender address
function get_my_medical_record() public view returns(Patient memory){
    uint id = my_patient_record[msg.sender];
    return patient_records[id];
}
```

Figura 14: Retorna a ficha médica do usuário

- **Obter meus relatórios médicos (get_medical_reports)**: Essa função retorna uma lista de relatórios, caso possua, da conta que está invocando essa função do contrato, não requer parâmetros e é uma função do tipo view, que significa que não altera o estado da blockchain, apenas visualiza.

```
// get sender medical reports
function get_medical_reports() public view returns(MedicalReport[] memory){
    uint id = my_patient_record[msg.sender];
    return my_medical_reports[id];
}
```

Figura 15: Retorna os relatórios médicos do usuário

- **Obter meus médicos autorizados (get_my_authorized_doctors)**: Essa função retorna a lista de médicos autorizados a criar relatórios, caso possua, da conta que está invocando essa função do contrato, não requer parâmetros e é uma função do tipo view, que significa que não altera o estado da blockchain, apenas visualiza.

```
// get my authorized doctors
function get_my_authorized_doctors() public view returns(uint[] memory){
    uint patient_id = my_patient_record[msg.sender];
    return authorized_doctors[patient_id];
}
```

Figura 16: Retorna os médicos autorizados do usuário

- **Obter meus contatos confiáveis (get_my_trusted_accounts)**: Essa função retorna a lista de contatos confiáveis, caso possua, da conta que está invocando essa função do contrato, não requer parâmetros e é uma função do tipo view, que significa que não altera o estado da blockchain, apenas visualiza.

Os contatos confiáveis de uma conta podem gerenciar os médicos autorizados a criar relatórios médicos.

```
// get my trusted account list
function get_my_trusted_accounts() public view returns(uint[] memory){
    uint patient_id = my_patient_record[msg.sender];
    return trusted_accounts[patient_id];
}
```

Figura 17: Retorna os contatos de confiança do usuário

- **Autorizar um médico (give_doctor_permission)**: Essa função recebe o número de identificação do médico como parâmetro. Portanto, se esse médico já existir no sistema e se ele ainda não existir na lista de médicos autorizados daquela conta, esse médico será adicionado a lista de médicos autorizados da conta que invocou a função.

```
// the patient can add trusted doctor in your authorized doctor list and
function give_doctor_permission(uint _doctor_id) public
    exists_doctor(_doctor_id)
    doctor_not_authorized(my_patient_record[msg.sender], _doctor_id){

    authorized_doctors[my_patient_record[msg.sender]].push(_doctor_id);
}
```

Figura 18: Adiciona o médico a sua lista de médicos autorizados

- **Remove a permissão de um médico (remove_doctor_permission)**: Essa função recebe o número de identificação do médico como parâmetro. Portanto, se esse médico já existir no sistema e se ele já existir na lista de médicos autorizados daquela conta, esse médico será removido da lista de médicos autorizados da conta que invocou a função e consequentemente não poderá mais criar relatórios médicos para aquele usuário.

```
// the patient can remove doctors from your authorized doctor list
function remove_doctor_permission(uint _doctor_id) public
    exists_doctor(_doctor_id)
    doctor_authorized(my_patient_record[msg.sender], _doctor_id){
        uint patient_id = my_patient_record[msg.sender];
        remove_item(_doctor_id, patient_id, 0);
}
```

Figura 19: Remove o médico a sua lista de médicos autorizados

- **Adicionar/Remover um contato de confiança (add_trusted_accounts)/(remove_trusted_accounts)**: Essas duas funções recebem um número de identificação de um usuário. Esse usuário será removido ou adicionado da lista de confiança da conta que está chamando essas funções, dependendo da função que será chamada. Caso seja uma operação de adição, a conta adicionada poderá gerenciar os médicos autorizados daquela conta e caso seja de remoção, a conta removida não poderá mais participar do gerenciamento de médicos autorizados daquela conta.

```
// the patient can add trusted accounts in your trusted list
function add_trusted_account(uint _account_id) public
    exists_patient(_account_id)
    account_not_trusted(my_patient_record[msg.sender], _account_id){
        trusted_accounts[my_patient_record[msg.sender]].push(_account_id);
}

// the patient can remove trusted accounts from your trusted list
function remove_trusted_account(uint _account_id) public
    exists_patient(_account_id)
    account_trusted(my_patient_record[msg.sender], _account_id){
        remove_item(_account_id, my_patient_record[msg.sender], 1);
}
```

Figura 20: Adiciona ou remove contatos de confiança

É importante notar que, o dono da chave privada da conta Ethereum é o detentor do poder sobre os seus dados. Devido a estrutura dessas funções, uma conta Ethereum não pode se passar por outra, uma vez que o endereço que invoca essas funções já é automaticamente obtido sem depender de nenhum parâmetro. Ou seja, a conta que invoca essas funções só podem alterar os dados vinculados ao seu endereço Ethereum.

O usuário é o responsável por gerenciar seus contatos de confiança, ou seja, apenas o dono da conta pode adicionar ou remover contatos confiáveis a sua lista. E caso um usuário possua um contato de confiança, esse contato também poderá gerenciar a lista de médicos autorizados daquele usuário. Com isso, apenas o próprio dono da conta ou um contato confiável pode adicionar ou remover permissão de um médico para criar relatórios médicos para aquela conta.

A seguir serão listadas funções que também são invocadas por pacientes, mas dessa vez, como contatos de confiança, sendo capaz apenas nesse caso de usar essas rotinas. Essas funções são sobrecargas de funções já existentes.

- **Autorizar um médico (give_doctor_permission)**: Essa função é uma sobrecarga e recebe um parâmetro a mais além do número identificador do médico, recebe também o número identificador da conta do usuário. Possui as mesmas validações de antes, caso o médico exista e caso esse médico ainda não esteja autorizado nessa conta. Agora também verifica se a conta que está mandando a transação faz parte da lista de contatos confiáveis da conta passada por parâmetro, caso essas condições sejam verdadeiras, o médico é adicionado a lista de médicos autorizados para aquele usuário.

```
// trusted accounts can add trusted doctors in another authorized doctor list account
function give_doctor_permission(uint _account_id, uint _doctor_id) public
    exists_doctor(_doctor_id)
    only_trusted(_account_id)
    doctor_not_authorized(_account_id, _doctor_id){

    authorized_doctors[_account_id].push(_doctor_id);
}
```

Figura 21: Autoriza médico

- **Remove permissão de um médico (remove_doctor_permission)**: Essa função é uma sobrecarga e recebe um parâmetro a mais além do número identificador do médico, recebe também o número identificador da conta do usuário. Possui as mesmas validações de antes, caso o médico exista e caso esse médico já esteja autorizado nessa conta. Agora também verifica se a conta que está mandando a transação faz parte da lista de contatos confiáveis da conta passada por parâmetro, caso essas condições sejam verdadeiras, o médico é removido da lista de médicos autorizados para aquele usuário.

```
// trusted accounts can remove doctors from another authorized doctor list account
function remove_doctor_permission(uint _account_id, uint _doctor_id) public
    exists_doctor(_doctor_id)
    only_trusted(_account_id)
    doctor_authorized(_account_id, _doctor_id){
        uint patient_id = my_patient_record[msg.sender];
        remove_item(_doctor_id, patient_id, 0);
}
```

Figura 22: Remove permissão de um médico

Na sequência, outra estrutura vital nesse ecossistema são os médicos. Ao contrário das contas dos pacientes, os médicos são identificados no sistema, para vias de segurança. O usuário precisa saber quem é aquele médico, se ele realmente é um médico regularizado e apto para desempenhar esse papel, além de outras informações como sua área e sua especialidade. Para criar um médico o contrato disponibiliza a função ”create_doctor_record”:

```
// create doctor record
function create_doctor_record(
    string memory _name,
    string memory _crm,
    string memory _area,
    string memory _expertise
) public returns (uint id){
```

Figura 23: Criação de um médico

Apenas contas do tipo médico podem criar relatórios, e eles só podem criar relatórios para contas de usuários nas quais eles sejam autorizados, seja pelo próprio usuário ou por um contato de confiança daquele usuário. Atualmente, contas do tipo médico só podem invocar a função criar relatórios médicos (create_medical_report).

```
// create medical report
function create_medical_report(
    uint _patient_id,
    string[] memory _symptoms,
    string memory _diagnostic,
    string memory _prescription,
    string memory _annotation
) public doctor_authorized(_patient_id, my_doctor_record[msg.sender])
    exists_doctor(my_doctor_record[msg.sender])
    exists_patient(_patient_id)
    returns (MedicalReport memory){
```

Figura 24: Criação de um relatório médico

Um relatório médico representa uma consulta entre médico e usuário e possui algumas informações obrigatórias para ser criado, é necessário passar como parâmetro o número de identificação do usuário, os sintomas apresentados, o diagnóstico do médico além da prescrição e alguma anotação pertinente para o caso. Depois de adicionado, o usuário poderá visualizar o novo relatório criado dentre todos os outros anteriores. A ideia é que uma criança nascida depois da implementação do Healthchain, tenha todo o seu registro médico armazenado com segurança, organizado, completo e disponível para ele desde o momento do seu nascimento até o seu presente.

D. Fluxo de uso do sistema

Na imagem a seguir, será representado um possível fluxo de uma consulta médica entre um paciente e um médico em um hospital ou em um consultório.

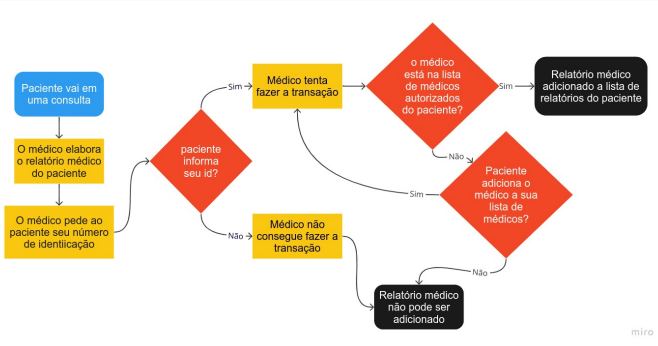

Figura 25: Fluxo de uma consulta

O próximo fluxo descreve uma situação onde o paciente está inconsciente e o seu contato de emergência gerencia os médicos autorizados da sua conta.

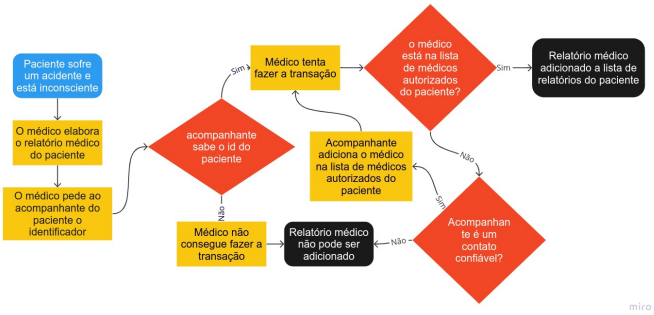

Figura 26: Fluxo de um acidente

Este fluxo pode ser aperfeiçoado ao longo do tempo, prevendo por exemplo que, a ficha médica de emergência de um paciente possa ser configurada na tela de bloqueio do seu celular, e uma das informações disponíveis seja o seu número identificador de usuário. Assim será possível para um médico, em caso de emergência, ver o identificador do usuário na tela de bloqueio do seu celular e buscar sua ficha médica completa.

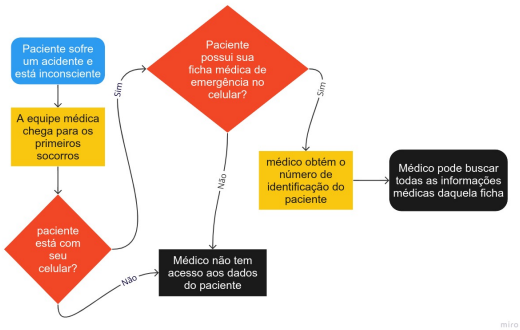

Figura 27: Fluxo de um acidente com a ficha médica de emergência configurada

V. Demonstração de Uso

Nesse tópico será feita uma demonstração básica de como um usuário poderá interagir com um médico, em uma consulta casual, e o seu relatório médico ser carregado no sistema.

Para exemplificar melhor, será demonstrado o uso do Healthchain, através de dois sistemas, capazes de se conectar ao contrato e enviar transações, um será para uso do médico, e será um sistema web, e o outro será um aplicativo mobile, para uso do usuário.

Na sequência, em um cenário fictício, o usuário Allan marcou uma consulta com o médico Victor. Agora vamos avançar passo a passo utilizando esse cenário.

- Allan se dirige ao consultório.
- Victor pergunta a Allan qual seu número de identificação, e digita o número identificador no seu sistema.

Figura 28: Tela de identificação do paciente

- Victor clica no botão de iniciar consulta, o sistema direciona para a próxima página, e ele pode começar a preencher o relatório.
- Allan explica o seu problema.
- Victor chega a sua conclusão e cria o relatório médico de Allan, mas não faz a transação ainda.

Figura 29: Tela de criação do relatório médico

- Allan concorda com o relatório, mas Victor ainda não é um médico autorizado dele

Figura 30: Tela de listagem de médicos autorizados

- Allan clica no botão de ”mais”para adicionar Victor a sua lista
- Victor informa seu número identificador, para que Allan possa autoriza-lo
- Allan faz a procura pelo número de identificação do médico

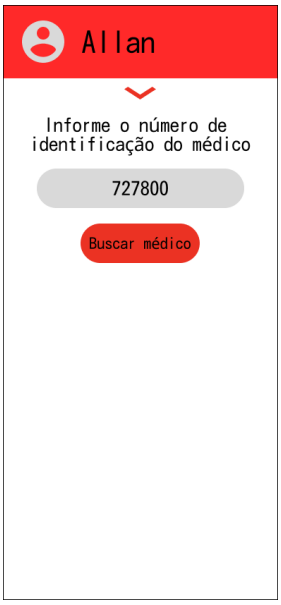

Figura 31: Procura de um médico cadastrado

- Caso Victor, seja um médico já cadastrado, o nome e o CRM dele são exibidos pelo sistema e Allan pode autoriza-lo

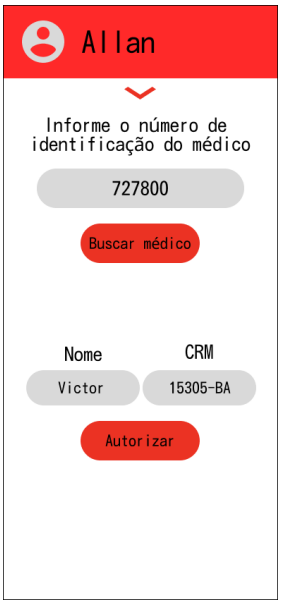

Figura 32: Autorização de um novo médico

- Depois da transação completa na blockchain, Allan pode verificar na sua lista que Victor agora está presente.

Figura 33: Novo médico adicionado com sucesso

- Victor então envia a transação de criação do relatório, apertando no botão ”Criar relatório”

Figura 34: Transação enviada com sucesso para a blockchain

- Assim que a transação for aceita pela rede, Allan já conseguirá ver em seus relatórios médicos, também o recém criado.

Figura 35: Novo relatório adicionado

VI. Considerações finais

Acompanhando a evolução tecnológica, a medicina conseguiu evoluir exponencialmente. A partir do século XX, os avanços na área foram muito significativos, aliado ao constante e veloz desenvolvimento tecnológico médicos e pesquisadores ganharam ferramentas poderosas para evoluir suas pesquisas, seus tratamentos e seus diagnósticos, por exemplo. Muitas doenças que antes praticamente sentenciavam uma pessoa a morte, hoje em dia são bastante controladas, dentre muitos outros exemplos na área [2].

Em contrapartida, os dados médicos receberam pouca atenção e seu gerenciamento não evoluiu no mesmo ritmo da medicina como um todo. Esse tema foi negligenciado por muito tempo e afeta toda a sociedade brasileira até os dias de hoje, os dados de um usuário do sistema de saúde, quando armazenados, são mantidos de forma desorganizada, heterogênea, insegura e indisponível. Acabam sendo criados grandes repositórios de informações recortadas, cada usuário possui um pedaço do seu histórico médico espalhado em várias instituições de saúde distintas que muitas vezes não possuem as mesmas normas e os mesmos métodos. Por consequência, nem os usuários, nem instituições de saúde, nem o governo e nem pesquisadores possuem uma visão macro organizada, objetiva e interoperável sobre esses dados. Muito tempo e dinheiro poderia ser poupado, além do mais importante, a vida de pessoas.

Quadros médicos delicados muitas vezes são inimigos do tempo, a decisão de um médico, um segundo mais veloz, pode fazer toda a diferença no caso. Por exemplo, uma mínima demora para obter uma informação simples, como o tipo sanguíneo de uma pessoa desacordada e desacompanhada, pode complicar ainda mais um quadro já complexo.

O Healthchain é a base para a construção e o desenvolvimento de uma plataforma aberta, interoperável, homogênea, segura e disponível para todos. Os dados são armazenados de forma segura usando as características intrínsecas da blockchain, são disponíveis para qualquer pessoa visualizar de forma aberta dentro da rede Ethereum e são anonimizados para assegurar a privacidade dos usuários. As informações são confiáveis, uma vez que não podem ser adulteradas e apenas são produzidas por médicos cadastrados e identificados que podem ser auditados facilmente.

Muitas melhorias futuras podem ser acopladas facilmente a esse ecossistema. Criação de funcionalidades para pesquisadores, por exemplo é uma delas. Podem ser criadas funções de filtragem que retornem dados de forma gratuita sobre essa base de dados confiável. Pesquisadores podem, por exemplo, solicitar através de uma dessas funções a quantidade de pessoas diabéticas na cidade de Salvador/BA dentre outras possibilidades.

O governo pode também se beneficiar dessa plataforma, uma vez que poderá analisar as informações importantes e criar ações mais assertivas para descobrir causas de possíveis surtos por exemplo. Outro ponto importante é sobre a distribuição de insumos, como vacinas, medicamentos e profissionais. Informações de vacinação podem ser facilmente acopladas ao contrato, e analisando os números de vacinados por região, o governo pode distribuir melhor as vacinas pelo território nacional por exemplo.

Portanto, informações abertas e anonimizadas são a melhor forma de gerenciar esse tipo de dado. Todos os envolvidos são beneficiados, e a sociedade como um todo tende a ganhar. A expectativa é que com a implementação do Healthchain, juntamente com sua evolução, os próximos bebês nascidos tenham todo o seu histórico médico rastreado de forma organizada desde o seu nascimento até o seu presente. Que médicos possam ter informações confiáveis disponíveis a todo o momento para embasar suas decisões. Que pesquisadores tenham uma base confiável de dados para realizar suas pesquisas, sem infringir a privacidade de ninguém. E que o governo possa ter ações mais assertivas de acordo com os dados analisados na rede.

Referências

[1] Raafat Abou Jaoude Joe e George Saade. “Aplicativos Blockchain – Uso em Diferentes Domínios”. Em: *Acesso IEEE* 7 (). DOI: 10.1109/ACCESS.2019.2902501.

[2] Raphael Abreu. *Descubra os 10 principais avanços da medicina e suas aplicações!* URL: https://blog.iclinic.com.br/avancos-da-medicina/.

[3] Dr. Abdullah Albeyatti. *MedicalChain*. URL: https://medicalchain.com/en/whitepaper/.

[4] Tomaso Aste, Paolo Tasca e Tiziana Di Matteo. "Blockchain technologies: The foreseeable impact on society and industry". Em: (2017).

[5] Blocknative. *What is the Mempool? - Your Intro to In-Flight Transactions*. URL: https://www.blocknative.com/blog/mempool-intro.

[6] Brasil. "LEI Nº 13.709 DE 14 DE AGOSTO DE 2018". Em: *Diário Oficial [da] República Federativa do Brasil* (14 de ago. de 2012). ISSN: 1677-7042. URL: https://www.planalto.gov.br/ccivil_03/_ato2015-2018/2018/lei/l13709.htm (acesso em 04/06/2023).

[7] Brian Curran. *What is a Merkle Tree? Beginner's Guide to this Blockchain Component*. URL: https://blockonomi.com/merkle-tree/.

[8] Decrypt. *Blockchain do bitcoin atinge 300 GB de tamanho*. URL: https://portaldobitcoin.uol.com.br/blockchain-do-bitcoin-atinge-300-gb-de-tamanho/#:~:text=O%20tamanho%20da%20blockchain%20do,Bitcoin%20nos%20%C3%BAltimos%2010%20anos..

[9] ethereum. *WHAT IS ETHEREUM?* URL: https://ethereum.org/en/developers/docs/intro-to-ethereum/.

[10] Euromoney. *How does a transaction get into the blockchain?* URL: https://www.euromoney.com/learning/blockchain-explained/how-transactions-get-into-the-blockchain.

[11] Aguiar Farina. *Prontuário Médico*. URL: https://portal.cfm.org.br/artigos/prontuario-medico/#:~:text=Nele%20constam%20%2C%20de%20forma%20organizada,indica%C3%A7%C3%B5es%20de%20tratamentos%20e%20prescri%C3%A7%C3%B5es..

[12] Forbes. *Blockchain 50: as maiores empresas que adotam a tecnologia*. URL: https://forbes.com.br/listas/2020/02/blockchain-50-as-maiores-empresas-que-adotam-a-tecnologia/.

[13] Universidade do Futebol. *A evolução da medicina na sociedade contemporânea*. Agosto de 2007. URL: https://universidadedofutebol.com.br/2007/08/09/a-evolucao-da-medicina-na-sociedade-contemporanea/.

[14] Geeks for Geeks. *Blockchain Structure*. URL: https://www.geeksforgeeks.org/blockchain-structure/.

[15] Yifei Huang. *Decoding Ethereum smart contract data*. URL: https://towardsdatascience.com/decoding-ethereum-smart-contract-data-eed513a65f76.

[16] Investopedia. *Blockchain: o que é, como funciona e como pode ser usado*. URL: https://www.investopedia.com/terms/b/blockchain.asp.

[17] Javatpoint. *Tutorial Blockchain*. URL: https://www.javatpoint.com/blockchain-tutorial.

[18] Anastasiia Lastovetska. *Blockchain Architecture Basics: Components, Structure, Benefits Creation*. URL: https://mlsdev.com/blog/156-how-to-build-your-own-blockchain-architecture.

[19] Antony Lewis. *A gentle introduction to Ethereum*. URL: https://bitsonblocks.net/2016/10/02/gentle-introduction-ethereum/.

[20] Satoshi Nakamoto. "Bitcoin: A peer-to-peer electronic cash system". Em: *Decentralized Business Review* (2008), p. 21260.

[21] Howard Poston. *Blockchain structure*. URL: https://resources.infosecinstitute.com/topic/blockchain-structure/.

[22] Ministério da Saúde. *Rede Nacional de Dados em Saúde - RNDS*. URL: https://www.gov.br/saude/pt-br/composicao/seidigi/rnds.

[23] Serasa. *Prontuário Eletrônico: entenda como funciona*. URL: https://serasa.certificadodigital.com.br/blog/prontuario-eletronico-do-pacientepep/prontuario-eletronico-entenda-como-funciona/#:~:text=O%20que%20%C3%A9%20o%20prontu%C3%A1rio,digital%20do%20conhecido%20prontu%C3%A1rio%20m%C3%A9dico..

[24] Elena Sinelnikova. *Decentralized applications development for .NET developers using Microsoft Visual Studio*. URL: https://blockgeeks.com/decentralized-applications-development-net-developers/.

[25] Carin Tom. *O que é o Livro Razão na contabilidade?* URL: https://contadores.contaazul.com/blog/livro-razao-contabilidade.

[26] Redação Warren. *Ethereum: tudo sobre a segunda maior criptomoeda do mundo*. URL: https://warren.com.br/magazine/ethereum/.

[27] Min Xu, Xingtong Chen e Gang Kou. "A systematic review of blockchain". Em: *Financial Innovation* 5 (dez. de 2019). DOI: 10.1186/s40854-019-0147-z.

[28] Yujian Zhang. *Toward Vulnerability Detection for Ethereum Smart Contracts Using Graph-Matching Network*. URL: https://www.mdpi.com/1999-5903/14/11/326.