CENTRO UNIVERSITÁRIO CAMPO LIMPO PAULISTA
PROGRAMA DE MESTRADO EM CIÊNCIAS DA COMPUTAÇÃO - PMCC

TIAGO LEORATTO

MODELO PARA CRIAÇÃO DE PRONTUÁRIO ELETRÔNICO DE PACIENTE BASEADO EM BLOCKCHAIN INTEGRADO A BANCO DE DADOS ORIENTADO A DOCUMENTOS

Campo Limpo Paulista
2021

CENTRO UNIVERSITÁRIO CAMPO LIMPO PAULISTA
PROGRAMA DE MESTRADO EM CIÊNCIAS DA COMPUTAÇÃO - PMCC

TIAGO LEORATTO

MODELO PARA CRIAÇÃO DE PRONTUÁRIO ELETRÔNICO DE PACIENTE BASEADO EM BLOCKCHAIN INTEGRADO A BANCO DE DADOS ORIENTADO A DOCUMENTOS

Esse documento corresponde à dissertação apresentada à Banca Examinadora para defesa no curso de Mestrado em Ciência da Computação da Faculdade Campo Limpo Paulista.

Orientador: Prof. Dr. Marcelo de Paiva Guimarães

Campo Limpo Paulista, 07 de maio de 2021.

O presente trabalho foi realizado com apoio da
Coordenação de Aperfeiçoamento de Pessoal de Nível
Superior - Brasil (CAPES) - Código de Financiamento 001.

Ficha catalográfica elaborada pela

Biblioteca Central da Unifaccamp

L612m

Leoratto, Tiago

Modelo para criação de prontuário eletrônico de paciente baseado em *blockchain* integrado a banco de dados orientado a documentos / Tiago Leoratto. Campo Limpo Paulista, SP: Unifaccamp, 2021.

Orientador: Prof°. Dr. Marcelo de Paiva Guimarães

Dissertação (Programa de Mestrado Profissional em Ciência da Computação) – Centro Universitário Campo Limpo Paulista – Unifaccamp.

1. *Blockchain*. 2. *BigchainDB*. 3. Prontuário médico. I. Guimarães, Marcelo de Paiva. II. Centro Universitário Campo Limpo Paulista. III. Título.

CDD-005.75

Ano de 2020 ficará na história da humanidade.

AGRADECIMENTOS

Agradeço a minha esposa Débora pela paciência e carinho que me deu durante toda a jornada. A meu orientador que com calma e conhecimento auxiliou meu caminho.

RESUMO

Leoratto, Tiago. **Modelo para criação de Prontuário eletrônico de Paciente baseado em blockchain integrado a banco de dados orientado a documentos**. 2021. 70 f. Dissertação (Mestrado em Ciências da Computação) -Centro Universitário Campo Limpo Paulista, Campo Limpo Paulista, 2021.

As aplicações baseadas em blockchain são caracterizadas pela descentralização do processamento e gerenciamento dos dados, e pela possibilidade de o usuário ser o gestor dos seus dados. Portanto, o uso dessa tecnologia tem despertado interesse na área da saúde, principalmente para o desenvolvimento de prontuários eletrônicos. Este trabalho tem como objetivo apresentar um modelo de prontuário eletrônico baseado em blockchain integrado a um banco de dados orientado a documentos (noSQL). O modelo visa dar aos pacientes a gestão de seus próprios registros médicos e conceder acesso a quem desejar. Além disso, as entidades médicas e de pesquisa configuradas como validadores de blockchain terão acesso ao histórico do paciente para consultas de dados de forma segura e coesa. O modelo desenvolvido faz parte da pesquisa em realidade virtual e recuperação neurofuncional (BRAINN_VR Initiative), Comitê de Ética em Pesquisa em Seres Humanos - CAAE 35771314.4.0000.5404, em desenvolvimento no Instituto Brasileiro de Neurociências e Neurotecnologia (BRAINN). Resultado de testes iniciais apontam que o modelo e a implementação são viáveis como solução para implementação de prontuários eletrônicos.

Palavras-chave: Blockchain, BigchainDB, Prontuário médico.

ABSTRACT

Leoratto, Tiago. **Modelo para criação de Prontuário eletrônico de Paciente baseado em blockchain integrado a banco de dados orientado a documentos**. 2021. 70 f. Dissertação de pesquisa (Mestrado em Ciências da Computação) -Centro Universitário Campo Limpo Paulista, Campo Limpo Paulista, 2021.

Blockchain-based applications are characterized by the decentralization of processing and management of data, and the possibility of the user being the manager of their data. Therefore, the use of this technology has aroused interest in the health area, especially for the development of electronic medical records. This work aims to present a model for electronic records based on blockchain integrated with a document-oriented database(noSQL). The model aims to give patients the management of their own medical records and grant access to whoever wishes. In addition, medical and research entities configured as blockchain validators will have access to patient history for secure data consultations. The developed model is part of the research in virtual reality and neurofunctional recovery (BRAINN_VR Initiative), Human Research Ethics Committee - CAAE 35771314.4.0000.5404, under development at the Brazilian Institute of Neuroscience and Neurotechnology (BRAINN). Results of initial tests show that the model and the implementation are viable as a solution for the implementation of electronic medical records.

Keywords: Blockchain, BigchainDB, Eletronic medical records.

LISTA DE FIGURAS

LISTA DE TABELAS

LISTA DE ABREVIATURAS E SIGLAS

AB	Atenção Básica
AES	Advanced Encryption Standard
API	Application Programming Interface
BFT	Byzantine Fault Tolerance
BSON	Binary JSON
CPF	Cadastro da pessoa física
DAB	Departamento de Atenção Básica
DPoS	Delegated Proof os stack
e-SUS	SUS eletrônico
IP	Internet Protocol address
IPFS	InterPlanetary File System
JSON	JavaScript Object Notation
PEP	Prontuário eletrônico do paciente
PoA	Proof of authority
PoI	Proof of impostance
POMR	Acronym for Problem-Oriented Medical Record
PoS	Proof of stack
PoW	Proof of work
REST	Representational State Transfer
RSA	Acronym for Ron Rivest, Adi Shamir e Leonard Adleman
SOAP	Acronym for Subjective, Objective, Assessment and Plan
SUS	Sistema Único de Saúde

SUMÁRIO

1 INTRODUÇÃO

Com os eventos que acontecem ao longo da vida, o ser humano recorre a instituições médicas para solicitar atendimento em saúde. Todo atendimento inclui o envolvimento e a participação de uma variedade de profissionais. Além disso, estes atendimentos aos pacientes podem ocorrer em diferentes locais, hospitais, clínicas e outras instituições de saúde. Para realização destes atendimentos são necessárias múltiplas informações de diferentes fontes, que vão garantir a continuidade do processo de cuidado do paciente. Tais informações precisam ser organizados de modo a produzir um contexto que servirá de apoio para tomada de decisão sobre o tratamento ao qual o paciente deverá ser submetido e orientar o processo de atendimento à saúde do paciente. Essas informações, conforme a Resolução CFM n° 1.638/2002, definem o prontuário médico, o qual é o documento único constituído de um conjunto de informações, sinais e imagens registradas, geradas a partir de fatos, acontecimentos e situações sobre a saúde do paciente e a assistência a ele prestada, de caráter legal, sigiloso e científico, que possibilita a comunicação entre membros da equipe multiprofissional e a continuidade da assistência prestada ao indivíduo (BRASIL, 2002).

Conforme a Cartilha sobre Prontuário Eletrônico (CFM; SBIS, 2012), os prontuários médicos eram inicialmente registrados em papel e ficavam disponíveis somente para um profissional, possuíam baixa mobilidade e ficavam sujeitos a ilegibilidade, ambiguidade, possíveis perdas de informação, falta de padronização e dificuldades de pesquisas coletivas. Além disso, o armazenamento demandava grandes espaços físicos, pois cada atendimento poderiam gerar vários documentos. Com o intuito de minimizar as limitações dos registros médicos em papel, os computadores foram incorporados nas rotinas das instituições médicas. Desde então, inúmeras

instituições médicas têm desenvolvido, implementado e avaliado diferentes modelos de prontuário médico eletrônico (HYPPöNEN et al., 2014). A necessidade de tornar a prestação de serviços médicos mais seguros, tornou a adoção de sistemas de prontuário eletrônico de pacientes (PEP) imprescindível. A tecnologia permite uma série de facilidades tanto para o atendimento clínico, como para a administração e pesquisa científica.

Com o uso da tecnologia inúmeras soluções de sistemas e de banco de dados foram criadas e utilizadas para atender a demanda das instituições médicas, porém esses sistemas não foram projetados para gerenciar prontuários médicos entre instituições ao longo da vida (AZARIA et al., 2016). Sendo assim, se um prestador de saúde fizer alterações nos registros de um paciente é provável que outro prestador não esteja ciente dessas alterações, quando os prestadores trabalham com sistemas diferentes que não se comunicam (LI et al., 2019). Além disso, os prontuários médicos dos pacientes são compostos de informações privadas e sensíveis, necessitando atenção no quesito segurança e transparência nos dados. Outro ponto importante do prontuário está no serviço à pesquisa na área de saúde. Dele pode-se colher informações, elaborar estatísticas e indicadores, beneficiando o ensino, a assistência e o planejamento em saúde.

Grande parte da apreensão em torno da segurança de dados e privacidade do paciente é alimentada por violações de segurança nos registros de saúde do paciente. Estudos revelam que as violações de dados de saúde estão crescendo (PARANJAPE et al., 2019). A principal preocupação dos pacientes é que eles têm pouco ou nenhum controle sobre suas informações depois de terem sido fornecidas a uma instituição de saúde. Os pacientes querem uma visão maior de como seus dados são usados, quem tem acesso a eles e quando eles são modificados (PARANJAPE et al., 2019).

Nos últimos anos blockchain despertou o interesse da área da saúde, tanto

que diversos artigos abordam o assunto. Segundo HöLBL et al. (2018) e AGBO; MAHMOUD; EKLUND (2019), é notável o aumento de iniciativas acadêmicas e de indústrias para o uso da tecnologia de blockchain na área de saúde. Com a adoção de blockchain, os prontuários médicos não dependerão de um armazenamento central e, além disso, possibilitará que os próprios pacientes façam a gestão dos seus dados. Isso ajudará a garantir um nível mais alto de privacidade e a transferir o controle para os pacientes, além disso, pesquisadores poderão se beneficiar da disponibilidade de um amplo conjunto de dados médicos, que podem ser usados para realizar estudos sobre patologias existentes (KARAFILOSKI; MISHEV, 2017).

1.1 Objetivos

Este trabalho tem como objetivo desenvolver um modelo de criação de prontuários eletrônico de pacientes estruturado baseado em blockchain, que terá como caraterísticas a descentralização da informação, privacidade, controle das informações pelo proprietário e armazenamento das informações em um banco de dados orientado a documento e escalável. Para atender o desenvolvimento do modelo, este trabalho apresenta os ativos[1] necessários , bem como os processos de criação de cada um.

O modelo desenvolvido neste projeto está inserido em uma blockchain permissionada, na qual os participantes são conhecidos. A Figura 1.1 mostra a visão geral da arquitetura proposta, na qual pacientes e instituições de pesquisa e saúde, utilizando uma aplicação, serão os atores de um cenário de utilização de prontuários de pacientes e terão suas informações armazenadas de forma descentralizada dentro da blockchain.

[1]Um ativo pode representar qualquer objeto físico ou digital. O ativo no contexto deste trabalho representa os prontuários médicos e as regras de acesso para os mesmos. Cada ativo é associado a proprietários, que são responsáveis pela sua criação.

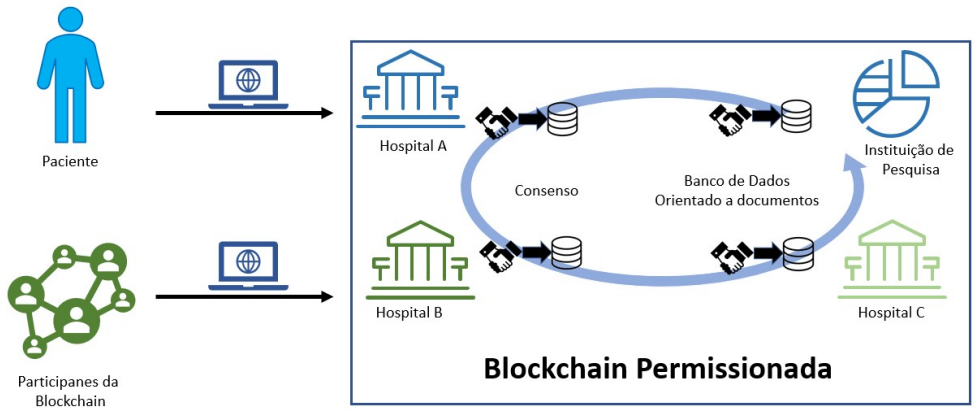


Figura 1.1: Visão geral da proposta

Os nós que estão dentro da blockchain serão configurados previamente para que sejam responsáveis por entrar em consenso e validar as informações submetidas dentro da blockchain. Como benefício por efetuar o consenso entre todos os envolvidos da blockchain, as instituições médicas e de pesquisa terão acesso as informações dos prontuários armazenados. Adicionalmente, serão utilizados algoritmos de criptografia para a segurança dos dados armazenados no banco de dados.

1.2 Contribuições

A contribuição principal deste trabalho é um modelo para criação de protuário eletrônico de paciente baseado em blockchain. O modelo desenvolvido visa atender um cenário de uma blockchain permissionada com consenso baseado em votos com armazenamento das informações em um banco de dados orientado a documentos. Como contribuição secundária é apresentado uma arquitetura de sistema que implementa o modelo.

Com o modelo pretende-se obter alguns benefícios:

- Através de um consenso baseado em votos das instituições configuradas como validadoras, visa-se obter agilidade nas informações a serem persistidas.
- Com a utilização de um banco de dados orientado a documentos ter facilidade em obter as informações armazenadas oriundas dos prontuários dos pacientes.
- Todas as informações armazenadas serão vinculadas a seus proprietários, dessa forma, somente os proprietários poderão visualizar ou conceder permissão a outros participantes da blockchain. Através do uso de criptografia no prontuário do paciente armazenado, somente quem tiver permissão conseguirá visualizar as informações.
- Facilitar auditoria de todas as informações, uma vez que não poderão ser apagadas e serão cronologicamente armazenadas. O histórico do paciente estará acessível independentemente de onde foi o atendimento.
- A arquitetura descentralizada proporcionada pela a blockchain garantirá que a perda de informações seja dificultada, uma vez que as informações estarão replicadas pelas instituições configuradas como validadoras dentro da blockchain.

1.3 Organização do Trabalho

No Capítulo 2 são apresentados os conceitos de blockchain, da estrutura do prontuário utilizada, conceitos e tipos de criptografias e também a arquitetura utilizada pelo BigchainDB. No Capítulo 3 são apresentados os trabalhos relacionados. Já no Capítulo 4, é demonstrado o modelo desenvolvido para contemplar a solução, o fluxo de criação de cada ativo e também exemplos da utilização do modelo. Finalmente, no Capítulo 5, são expostas as conclusões obtidas nesse trabalho.

2 CONCEITOS

Neste capítulo serão descritos os conceitos fundamentais utilizados para a construção do modelo desenvolvido. Os conceitos permeiam sobre a ótica da utilização da blockcahin para criação de prontuários de pacientes. É apresentado os tipos de blockchain existentes, formas de armazenamento, mecanismos de consenso e ferramentas e tecnologias que serão utilizadas para o desenvolvimento do modelo criado.

2.1 Blockchain

Os primeiros estudos que originaram o conceito blockchain ocorreram no início da década de 90, sendo inspirado no algoritmo de ordenação de data e hora, que foi usado para impedir a violação de documentos. Porém, a definição original foi criada em 2008 com a publicação do artigo "Bitcoin: A Peer-to-Peer Electronic Cash System"publicado por Satoshi Nakamoto(NAKAMOTO, 2009).

A blockchain é uma tecnologia que fornece uma plataforma de autenticação, descentralizada, transparente e que aplica uma abordagem orientada por consenso para facilitar as interações de várias entidades através do uso de um livro-razão [1] (CYRAN, 2018). A blockchain, segundo TASATANATTAKOOL; TECHAPANUPREEDA (2018), é uma forma de armazenamento de dados não centralizada, confiável e difícil de utilizar para fins fraudulentos. Conforme SWAN (2015), blockchain é um livro-razão que não pode ser alterado e facilita o processo de registro

[1] Livro Razão é um registro que tem a finalidade de coletar dados cronológicos de todas as transações registradas e organizar individualmente.

de transações de um certo domínio. Já SARAF; SABADRA (2018), definem como um livro-razão distribuído em uma arquitetura peer-to-peer [1], onde todos os nós conectados possuem uma cópia dos dados, sem precisar de um banco de dados centralizado.

A blockchain é uma cadeia de blocos divididos em duas partes:

Cabeçalho: estão as informações de identificação do bloco, como seu identificador único, a data e hora de sua criação.

Dados: estão as transações que podem ser qualquer tipo de dado ou atividade pertinentes ao cenário que a blockchain esta sendo utilizada.

As transações na blockchain utilizam o conceito de controle de ativo pelo proprietário.Os ativos são controlados pelos proprietários, com isso somente o dono pode transferir o controle para outro participante. A blockchain efetua a validação de propriedade da transação, se inválido, o bloco não é adicionado. Também é validado o gasto duplo [2] de um ativo, ou seja, a blockchain garante que um ativo não seja transferido mais de um vez indevidamente. Estas validações são cruciais para manter a integridade. Por isso, a blockchain implementa algoritmos de consenso para evitar falhas. Esses algoritmos são utilizados para que haja um acordo de validação dos blocos quando submetidos na blockchain.

Após validação, o bloco é inserido na blockchain. O novo bloco tem um

[1] Segundo TANENBAUM; WETHERALL (2010), em uma rede peer-to-peer, os nós agem como clientes e servidores para os outros nós da rede. Quando comparado ao modelo cliente/servidor, onde só há um servidor recebendo e processando requisições, no modelo peer-to-peer os nós compartilham responsabilidades de servir a outros nós. Assim, não há um ponto de controle único. É importante notar que para essa troca de informações ocorra, os nós têm de concordar com um conjunto de regras previamente definidas para a comunicação.

[2] Também conhecido com *double spending*, ocorre quando é efetuado a tentativa de utilizar o mesmo ativo duas vezes um ativo que não é mais da proprietário dele.

apontador para o bloco anterior, dessa forma, se qualquer informação dentro do bloco for alterada, automaticamente invalida os blocos da frente sucessivamente, pois os apontamentos estarão inválidos. A Figura 2.1 mostra um exemplo de cadeia que é formada conforme os blocos são inseridos na blockchain.

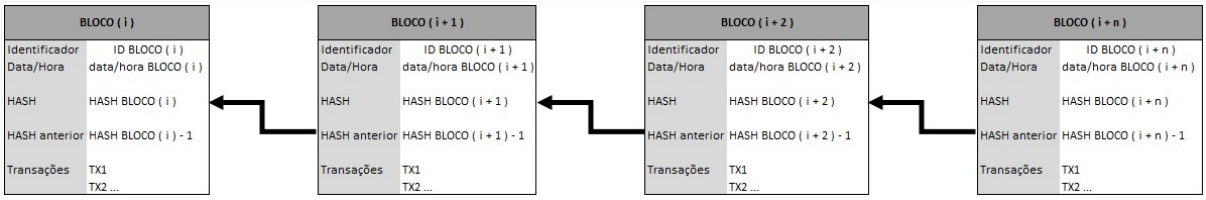


Figura 2.1: Cadeia de blocos na blockchain

Algumas características de destaque da blockchain são (**?**):

Descentralização da informação: Na blockchain a informação está dispersa entre os participantes fisicamente distribuídos, evitando que somente um participante tenha o poder sobre a informação. A cada bloco inserido na cadeia, esse é replicado para todos os participantes.

Disponibilidade: Após a inserção do bloco na blockchain por um participante, esse esta disponível a todos, mesmo que um dos participantes saia do sistema.

Privacidade: Através de técnicas de criptografia a blockchain faz com que as informações de origem e destino sejam anônimas. Na criação de uma transação é vinculado uma chave criptografada denotando o proprietário. Essa chave é utilizada nas futuras transações.

Integridade: Como a cada bloco inserido na blockchain recebe um identificador único e também um apontador para o bloco anterior, esse enlace entre os blocos confere que as informações sejam íntegras, visto que se o conteúdo de qualquer bloco seja alterado irá desencadear apontadores inválidos.

Imutabilidade: Todo o conteúdo de um bloco é criptografado e esse valor é referenciado no bloco posterior. Qualquer alteração que ocorra dentro bloco altera o valor criptografado e assim ferindo a característica de integridade.

Auditabilidade: A auditabilidade na blockchain permite verificar que todas as informações contidas nela são válidas. A auditabilidade faz uso das características da integridade e imutabilidade.

Outro ponto referente a blockchain é sua evolução em relação a sua aplicabilidade. Conforme SWAN (2015), a blockchain tem 3 momentos de evolução:

Blockchain 1.0 : aplicações em criptomoedas [3]. Quando a tecnologia de blockchain e criptomoeda chegaram no mercado em meados de 2009, uma das intenções principais era eliminar a interação de terceiros em todos os tipos de transferência financeiras. Na essência, a blockchain 1.0 é um livro-razão que registra as transações ao longo do tempo.

Blockchain 2.0 : usos relacionados a contratos inteligentes que vão muito além de transações em dinheiro. Nessa utilização, em essência, os contratos são executados automaticamente quando condições pré estabelecidas são satisfeitas. Dessa forma, quando duas partes efetuam uma transação entre si, somente após da verificação pelo contrato é que a transação é concluída. Esses tipos de aplicação foram inicialmente criadas em meados de 2015 com a Ethereum.

Blockchain 3.0 : apresenta uma nova forma de registrar as transações no livro-razão sem ser por blocos, através *Directed Acyclic Graph* ou *DAG*. Comparado a uma estrutura de dados em blocos regular que opera em uma cadeia reta,

[3]Uma criptomoeda é um ativo digital projetado para funcionar como um meio de troca, criado e armazenado eletronicamente no Blockchain usando técnicas de criptografia para controlar transações financeiras

os sistemas baseados em *DAG* têm mais amplitude, pois a rede é semelhante a uma árvore em expansão ao invés de uma cadeia reta de blocos.

LIM et al. (2019) fizeram um levantamento de blockchain disponíveis e analisaram os critérios de segurança, flexibilidade, quantidade de transações por segundo e mecanismo de consenso. Também em KUO; ZAVALETA ROJAS; OHNO-MACHADO (2019a) fizeram um estudo com objetivo de comparar as mais populares blockchain usando um método sistemático. Neste estudo foram encontradas 35 blockchains e dessas 10 foram ranqueadas como as mais populares e podem ser visualizadas na Tabela 2.1:

Tabela 2.1: Blockchains populares. Adaptado de KUO; ZAVALETA ROJAS; OHNO-MACHADO (2019a)

Blockchain	**Site Oficial**
Bitcoin	https://bitcoin.org
Ethereum	https://ethereum.org
Zcash	https://z.cash
Litecoin	https://litecoin.com
Dash	https://www.dash.org
PeerCoin	https://www.peercoin.net
Ripple	http://tipple.com
Monero	https://getmonero.org
MultiChain	https://multichain.com
Hyperledger	https://hyperledger.org

2.1.1 Tipos de Blockchain

Conforme a necessidade da utilização da blockchain, deve ser analisada e avaliada qual o tipo terá melhor aderência, que pode ser pública ou privada, sendo permissionada ou não.

Blockchain Pública: esse tipo permite que qualquer pessoa participe e através das transações possa atualizar dados dentro dela. Os dados armazenados são acessíveis por todos os participantes, e esse ponto deve ser levado em conta nos casos em que a privacidade dos dados deve ser preservada.

Blockchain Privada: é controlada por uma organização que é de confiança de todos os membros. Essa organização é responsável por permitir ou não os acessos dentro da blockchain

As blockchains públicas podem ser permissionadas ou não. Quando permissionadas, membros escolhidos antecipadamente que se tornam responsáveis pela validação das transações.

BURKE (2020) efetuou um comparativo entre os tipos da blockchain que pode ser visualizado na Tabela 2.2. Nele é comparado o tipo de participação (pública ou privada) com o nível de consentimento (com permissão ou sem permissão).

Tabela 2.2: Tipos de blockchain. Adaptado de BURKE (2020)

Participação / Consentimento	Pública	Privada
Sem permissão	Totalmente aberto a todos. Qualquer pessoa pode ingressar. As contas são criadas anonimamente usando funções limitadas. O nível de transparência da transação pode ser limitado para proteger a confidencialidade e fornecer anonimato. Exemplo: Público negociando criptomoedas	Por padrão, não há possibilidade.
Permissionada	Partipantes limitados. As contas são criadas através de regras. Transações podem ser escritas e lidas pelos partipantes. Exemplo: Pacientes e médicos participantes de processo médico	Participantes pertencem a um grupo. Contas são criadas através de regras. Transações podem ser escritas e lidas pelos partipantes. Exemplo: Funcionários de um departamento dentro de uma empresa.

Em geral, a tecnologia blockchain é útil nos casos de uso em que há mais de uma autoridade administrativa e há um déficit de confiança entre essas partes

(CHOWDHURY et al., 2018). Para tanto CHOWDHURY et al. (2018) e WüST; GERVAIS (2018) criaram fluxos que auxiliam a tomada de decisão para usar ou não a blockchain. A Figura 2.2 demonstra que inicialmente deve ser observado se há necessidade de armazenar as alterações dos estados do sistema, ou seja, deve se preocupar com a imutabilidade dos dados. Em seguida, é perguntado se existem muitos participantes que irão inserir informações. Após é verificado se existe uma terceira parte confiável para o processo, caso existe pode ser centralizado o processamento. Se todos os participantes são desconhecidos, uma blockchain pública pode ser a melhor opção. Se todos os participantes são conhecidos e confiáveis também não há necessidade de uma blockchain. Caso contrário, deve ser avaliado se existe restrição das informações, havendo um grupo restrito de verificação o melhor a ser adotado é uma blockchain privada, não havendo, uma blockchain permissionada pode ser implementada.

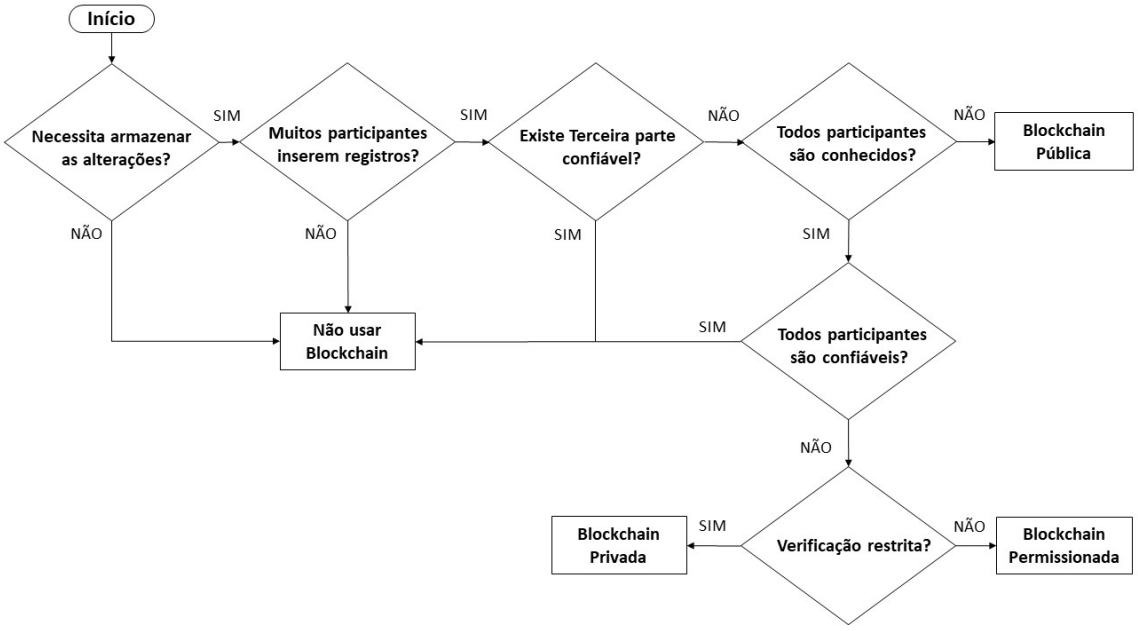


Figura 2.2: Fluxo para determinar quando usar blockchain. Adaptado de WüST; GERVAIS (2018)

2.1.2 Mecanismos de Consenso

O consenso tem como objetivo que os computadores cheguem a um acordo sobre um determinado valor (TANENBAUM; STEEN, 2008). No caso da blockchain, eles devem chegar a um acordo de qual bloco deverá ser adicionado no final da cadeia. No final do acordo, todos os computadores deverão ter a mesma cadeia. BASHIR (2018) define o mecanismo de consenso como um conjunto de passos que são dados por todos ou pela maioria dos nós para entrar em um acordo a respeito de um estado proposto ou valor. Existem diversos mecanismos para efetuar o consenso na blockchain:

Prova de trabalho(*Proof of Work*, Pow): primeiro mecanismo de consenso, iniciado com a criptomoeda bitcoin. Opera com o princípio de que é oneroso criar um bloco de transações na blockchain, mas é fácil validá-las. Os computadores que são encarregados de manter a segurança do blockchain, conhecidos como mineradores, trabalham para resolver uma função matemática chamada hash. Esta tarefa é extremamente repetitiva e, portanto, computacionalmente cara. Os computadores competem para encontrar um hash com propriedades específicas. O computador que encontra a resposta primeiro pode adicionar um novo bloco de transações na blockchain e são recompensados de alguma forma, na maioria das vezes com uma parcela de criptomoeda utilizada na blockchain em questão.

Prova de participação(*Proof of Stake*, PoS): nesse mecanismo o criador de um novo bloco é escolhido de forma aleatória, por um sorteio. Suas chances de ser escolhido é diretamente proporcional a sua participação na blockchain. Isso acarreta em maior participação dos membro, maior interesse e consequentemente menor número de falhas.

Prova de participação delegada (*Delegated Proof Of Stake*, DPoS): este al-

goritmo se propõe a melhorar a segurança do DoS, no qual não ocorre sorteio e sim uma eleição dos validadores pelas partes interessadas. Isso fornece o benefício do controle semi central da rede, como eficiência e velocidade, mantendo os recursos de descentralização (ALIAGA; HENRIQUES, 2017).

Prova de atividade(*Proof of Authority*, PoA): é um consenso híbrido que aproveita os benefícios dos algoritmos PoW e PoS. Nele o bloco minerado precisa ser assinado por N mineradores para ser válido. Neste sentido, mesmo que algum dos nós possua mais da metade de todas as moedas, ele não conseguirá controlar a criação de novos blocos por conta própria (ALSUNAIDI; ALHAIDARI, 2019).

Prova de importância (*Proof of Importance*, PoI): semelhante ao PoS, entretanto além de levar em consideração o fator monetário, considera também notoriedade, saldo e número de transações enviadas e recebidas (ALSUNAIDI; ALHAIDARI, 2019).

Byzantine Fault Tolerance (BFT): um novo bloco é gerado em uma rodada e, em cada rodada, um nó primário é escolhido segundo regras. Ele é o responsável pela condução da organização. Todo o processo está divido em três fases: pré-preparado, preparado, restabelecer. Em cada fase um nó deve receber os votos de 2/3 de todos os nós para poder passar para fase seguinte. O algoritmo requer que cada nó tenha conhecimento de todos os outros nós. Por isto, este algoritmo não pode ser utilizado em blockchain públicas (ALIAGA; HENRIQUES, 2017).

Ripple: é um algoritmo de consenso que utiliza sub-redes coletivamente confiáveis dentro de uma rede ampla. Dentro da rede existem os nós que podem ser servidores, os quais participam no processo de consenso ou são somente clientes. Se a aprovação relativa a uma organização atinge 80%, então ela pode ser armazenada na blockchain (ALIAGA; HENRIQUES, 2017).

O artigo de ALSUNAIDI; ALHAIDARI (2019) examina, compara e classifica diversos mecanismos de consenso. A Figura 2.3 apresenta a classificação em dois tipos de algoritmos, os baseados em provas e os baseados em votos. Uma diferença entre essa classificação é que nos algoritmos baseados em votos necessitam que os nós que irão fazer a validação do processo sejam identificados antecipadamente. Também ALSUNAIDI; ALHAIDARI (2019) apresenta em uma tabela que é possível verificar que os algoritmos de consenso baseados em votos são direcionados para blockchains permissionadas e privadas já os consensos baseados por prova para blockchains públicas.

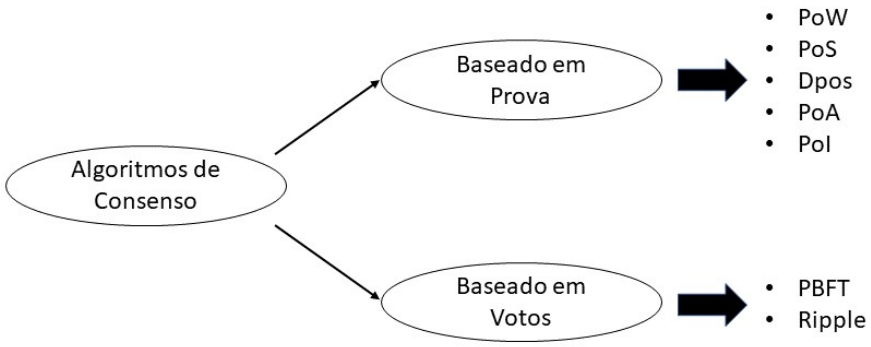


Figura 2.3: Classificação dos algoritmos de consenso. Adaptado de ALSUNAIDI; ALHAIDARI (2019)

2.2 Armazenamento

Qualquer tipo de informação pode ser armazenado em uma blockchain, quem a implementa define o tipo de informação contida na organização. As informações são criptografadas e assinadas digitalmente para garantir autenticidade e segurança. As transações são estruturadas em blocos e cada bloco contém um hash criptografado para o bloco anterior na blockchain. Os blocos são adicionados em uma ordem cronológica linear(LINN; KOO, 2016).

No caso dos prontuários médicos as informações são dinâmicas e podem ser volumosas. Algumas alternativas para o armazenamento de dados com a utilização da blockchain devem ser observados e analisados para uma melhor aderência da solução. Haja vista que uma base de dados com qualidade e de fácil consulta pode ser utilizada para prever epidemias, curar doenças, melhorar a qualidade de vida e evitar mortes. A medida que as informações são coletadas, os médicos poderão oferecer opções de tratamento com base nos dados de outros pacientes com condições semelhantes, fatores genéticos e estilo de vida.

O ambiente de dados compartilhado de uma blockchain fornece um amplo conjunto de dados, incluindo pacientes de diferentes origens. Com o intuito de coletar dados de saúde ao longo da vida de um paciente, a blockchain oferece dados ideais para estudos. Entretanto, o armazenamento de dados em grande quantidade na blockchain é um problema, seja devido tamanho limitado pelo protocolo ou pelas taxas de organização que podem ser cobradas. Felizmente, existem maneiras de armazenamento que podem auxiliar e minimizar esse problema. Entre as opções podem ser citados:

Sistemas de arquivos ponto a ponto: Permite o compartilhamento de arquivos entre os usuários do sistema. Os arquivos ficam guardados em cada usuário e podem ser compartilhados quando solicitados. Porém, para ocorrer o comparilhamento dos arquivos as duas pontas, o solicitante e receptor devem estar conectados. Além disso, a busca não pode ser efetuada pelo conteúdo dos arquivos e somente pelo seu endereço. Um exemplo desse sistema é o IPFS [4], The InterPlanetary File System.

Armazenamento descentralizado na nuvem: Semelhante ao sistema de arquivos ponto a ponto, entretanto seu armazenamento é efetuado em estruturas

[4] https://ipfs.io/

não locais. Do ponto de vista do usuário quando submetido o arquivo esse estará disponível a todos mesmo que ele não esteja conectado. Atualmente existem projetos nesse modelo, por exemplo, Storj[5] e Ethereum Swarm[6].

Bancos de dados: A forma mais tradicional de armazenar dados em sistemas computadorizados. Os dados são armazenados em repositório dedicados para este fim e posteriormente podem facilmente ser extraídos para análise. Os bancos de dados utilizam linguagens de programação para a manipulação e consulta dos dados. A arquitetura dos bancos de dados são voltadas para aumentar a velocidade de armazenamento e busca dos dados. Exemplo de bancos são MySQL[7] e MongoDB.

BigchainDB: Um software com propriedades de um banco de dados e com características da blockchain. Sua estrutura é suportada pelo banco de dados MongoDB e o protocolo Tendermint. Como tem propriedades de um banco de dados, há facilidade em efetuar consultas para extração de informações contidas nele. Esta solução é usada neste trabalho (detalhes serão apresentados na seção 2.4)

2.3 Criptografia

Segundo CORON (2006), o objetivo da criptografia é construir esquemas ou protocolos que ainda podem realizar certas tarefas, mesmo na presença de um adversário. Uma tarefa básica na criptografia é permitir que os usuários se comuniquem com segurança em um canal inseguro, de forma a garantir a privacidade e a autenticidade de suas transmissões.

[5]https://storj.io/
[6]https://swarm.ethereum.org/
[7]https://www.mysql.com/

Neste trabalho o procedimento de criptografia foi embasado no artigo de RAMESH (2018), no qual é efetuado a criptografia de arquivos que são armazenados em sistemas de arquivos para posterior consulta. Assim, serão abordados dois tipos de criptografia: simétrica e assimétrica.

2.3.1 Criptografia simétrica

Esse tipo de criptografia utiliza somente uma chave tanto para o processo de criptografar quanto descriptografar. Essa chave é compartilhada com todas as partes enolvidas no processo. Assim, após os dados serem criptografados, quem tiver a chave poderá descriptografar sem problema. A Figura 2.4 ilustra esse procedimento.

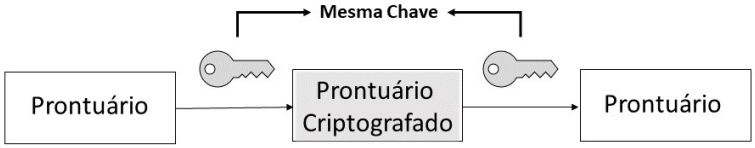


Figura 2.4: Criptografia simétrica

Neste trabalho será utilizado o algoritmo *Advanced Encryption Standard* (AES) do pacote *PyCryptodome* da linguagem *Python*. O desenvolvimento da AES iniciou em 1997 como alternativa para ataques de força bruta para o governo dos Estados Unidos. Nesse algoritmo é utilizado uma chave, também conhecida como chave secreta, de um tamanho definido, neste trabalho será utilizado uma chave de 128 bits, porém existem também de 192 e 256 bits. O governo dos Estados Unidos classifica as informações em confidencial, secreto e *top secret*, sendo que para confidencial e secreto qualquer tamanho é suficiente, já para *top secret* é necessário a utilização de uma chave de 256 bits (ROUSE, 2020).

Conforme o tamanho da chave utilizada o algoritmo efetua rodadas para concluir a criptografia. Uma chave de 128 bits (16 caracteres) por exemplo, efetua

10 rodadas. A cada rodada do algoritmo são realizadas 4 etapas: *AddRoundKey*, *SubBytes*, *ShiftRows* e *MixColumns*. Na última rodada, porém, a operação *MixColumns* não é executada (SOUZA; BORGES; CUNHA, 2008).

O algoritmo AES é utilizado no modelo desenvolvido para criptografar o prontuário médico. Uma chave de 16 caracteres é criada de forma aleatória, em seguida é utilizado as funções de criptografia AES que efetuam a criptografia do prontuário utilizando a chave criada.

2.3.2 Criptografia assimétrica

Na criptografia assimétrica, também conhecida como de chave pública, o algoritmo usa um par de chaves, sendo uma pública e outra privada. A chave pública é utilizada para criptografar e a chave privada para descriptografar. A chave pública pode ser disponibilizada, porém a privada deve ser mantida em segurança. A Figura 2.5 ilustra esse procedimento.

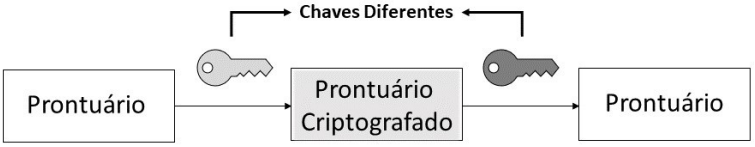


Figura 2.5: Criptografia assimérica

Para implementação do modelo será utilizado o algoritmo *RSA* do pacote *PyCryptodome* da linguagem *Python*. Neste algoritmo a chave de criptografia é pública e é diferente da chave de descriptografia, que é secreta (privada). No RSA, esta assimetria é baseada na dificuldade prática da fatoração do produto de dois números primos grandes (ROUSE, 2018).

No modelo desenvolvido nesta dissertação, o algoritmo RSA é utilizado para

criptografar a chave simétrica utilizada na criptografia do prontuário eletrônico. Após o prontuário ser criptografado de forma simétrica, a chave utilizada é criptografada utilizando todas as chaves públicas dos proprietários. Dessa forma, somente através das chaves privadas de cada proprietário será possível descriptografar a chave utilizada para criptografar o prontuário.

Cada paciente e instituição de saúde e pesquisa terão um par de chave, pública e privada, quando criados na blockchain. Essas chaves serão utilizadas para efetuar a validação de propriedade dos ativos criados. Assim, somente o proprietário das chaves poderá tomar uma ação sobre o ativo criado.

2.4 BigchainDB

O BigchainDB[8] é construído em cima de um banco de dados distribuído com alto rendimento, alta capacidade, baixa latência e uma linguagem de consulta NoSQL eficiente (GMBH, 2018). Também utiliza uma ferramenta para efetuar a comunicação dentro da rede e efetuar o consenso através de algoritmo BFT. Entrevendo as qualidades do BigchainDB, foi esta a solução utilizada para a implementação do modelo proposto.

O BigchainDB foi anunciado em 2016 e é chamado de banco de dados blockchain, por possuir propriedades de um banco de dados tradicional e ter características de uma blockchain. Atualmente Interplanetary Database Foundation[9], uma organização que concentra pesquisas na tecnologia de blockchain, é responsável pela manutenção dele.

[8]https://www.bigchaindb.com/
[9]https://ipdb.io/

A estrutura fundamental do BigchainDB é o *node*. Ele é formado pelo software BigchainDB, MongoDB[10] e Tendermint[11]. O MongoDB é o banco de dados de código aberto que opera utilizando a linguagem de consulta NoSQL[12], ele é o responsável pelo armazenamento dos dados oriundos do BigchainDB, fornecendo as características de banco de dados ao ambiente. O Tendermint adiciona um algoritmo de consenso baseado em votos e também a descentralização da informação através de um protocolo de rede ponto a ponto.

O BigchainDB também adiciona um artificio chamado de pipeline da blockchain. No pipeline de blockchain, os blocos são adicionados sem esperar que o bloco atual seja acordado pelos outros nós. A validação de blocos não é feita durante a adição de blocos, mas eventualmente por um processo de votação entre nós. Isso gera ganhos de desempenho (SAHOO; BARUAH, 2018). Para esse processo o BigchainDB utiliza dois dispositivos:

Conjunto de transações ou Backlog: recebe as transações novas que desejam ser inseridas no blockchain, elas podem estar desordenadas e inválidas.

Blockchain: contém os blocos ordenados e com as transações válidas.

A Figura 2.6 mostra o ciclo de vida de uma transação efetuada no BigchainDB. Um usuário da blockchain, através de uma aplicação, submete uma transação ao BigchainDB (1). O BigchainDB fornece os métodos de criptografia necessária e também faz algumas validações iniciais para as transações. A transação é enviada para o Tendermint realizar o processo de consenso. Como parte desse

[10] https://www.mongodb.com/

[11] https://tendermint.com/

[12] Os bancos de dados NoSQL (também conhecidos como "não apenas SQL") não são tabulares e armazenam dados de maneira diferente das tabelas relacionais. Existe uma variedade de tipos com base em seu modelo de dados. Os principais tipos são documento, valor-chave e gráfico(MONGODB, 2020).

consenso, são empacotadas as transações submetidas em um bloco e esse bloco sofre uma votação para avaliar sua validade (2). Quando é alcançado um consenso, é enviada uma confirmação do bloco criado (3). Com o bloco criado, é enviado ao banco de dados MongoDB (4), em seguida o banco de dados envia a confirmação do bloco armazenado (5). Logo após, é enviado para a aplicação a confirmação de todo o processo ao usuário (6).

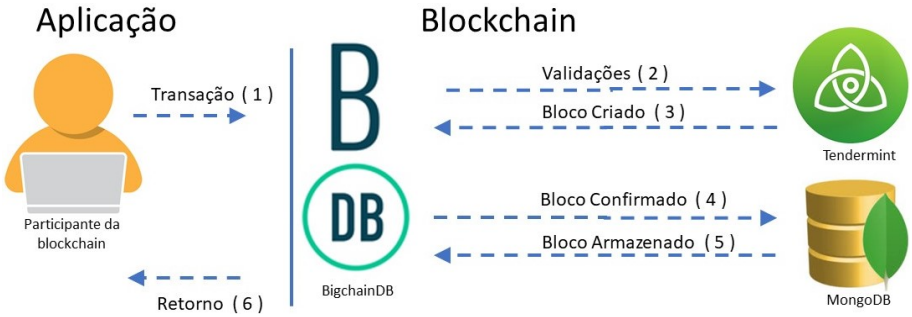


Figura 2.6: Ciclo de vida no BigchainDB. Adaptado de DHAMEJA (2018)

O BigchainDB utiliza o conceito de ativo para qualquer organização e esse é armazenado no banco de dados. Os ativos podem ser qualquer tipo de informação, como por exemplo, uma anotação de um atendimento médico a um paciente. Esses ativos são criados através da transação *create*, a qual, além de criar, associa o proprietário. O proprietário do ativo é identificado por uma chave pública criptografada e inserido no ativo criado. Além da chave pública, uma chave privada também é assinada no ativo, assim se o par de chaves não pertencer ao mesmo usuário, o ativo não será criado. Quando o ativo é criado um identificador é gerado. Com o identificador único, posteriormente o ativo pode ser referenciado para consultas e até mesmo transferidos para outros usuários. A transferência do ativo ocorre através da transação *transfer*, onde é necessário informar a chave privada do proprietário aprovando a transferência.

Os principais objetivos do BigchainDB comparados com blockchain e com banco de dados distribuídos são ilustrados na Tabela 2.3.

Tabela 2.3: Objetivos BigchainDB. Adaptado de GMBH (2018)

	Blockchain	**Banco de dados distribuídos**	**BigchainDB**
Descentralização	X		X
Tolerancia a falhas	X		X
Imutabilidade	X		X
Controle de ativo pelo proprietário	X		X
Alta taxa de organização		X	X
Baixa latência		X	X
Indexação e consulta de dados estruturados		X	X

No armazenamento dos dados o BigchainDB trabalha com as tabelas de transações, blocos e ativos. Na tabela de blocos estão todos os blocos ordenados uns aos outros e na tabela de ativos, todos os ativos criados. Já na tabela de transações, ela é utilizada como o artifício de backlog. As transações submetidas que sejam inválidas, por dados inválidos ou por conter ativos inexistentes ou ainda por causa da tentativa de transferir um ativo mais de uma vez indevidamente, são descartadas e não inseridas na tabela.

2.4.1 MongoDB

O MongoDB é um banco de dados de documentos com a escalabilidade e flexibilidade para consultas e indexações necessárias (MONGODB, 2017). Ele armazena arquivos no formato BSON(*Binary JSON*), que é uma forma de representação binária da popular JSON (*JavaScript Object Notation*). Conjuntos de documentos são armazenados em coleções (*collections*). E um conjunto de coleções denota um banco de dados.

O ponto positivo do MognoDB é sua flexibilidade. Sua estrutura orientada a documentos permite gravar os dados com facilidade e sem necessidade de muitas adaptações na aplicação. Diferente em um banco relacional onde muitas vezes é

necessário ajustar a aplicação para se adequar a estrutura do banco de dados. A Figura 2.7 apresenta um comparativo dos termos utilizados entre SQL e o MongoDB.

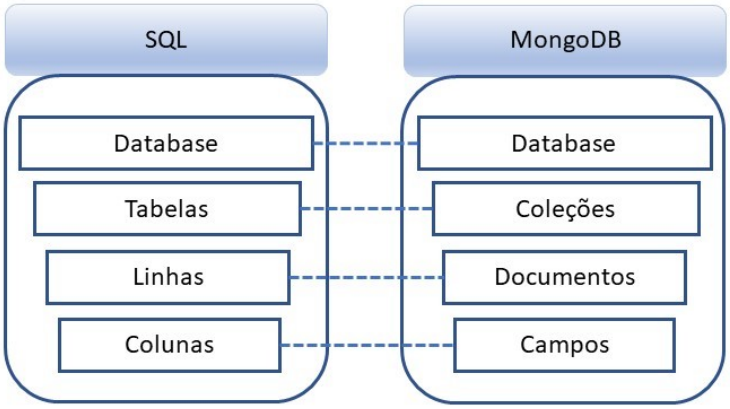


Figura 2.7: Comparativo de termos SQL e MongoDB.

Conforme CHODOROW; DIROLF (2010), o MongoDB foi projetado desde o início para ser expansível. Seus dados orientados a documentos permite dividir dados automaticamente em vários servidores. Pode equilibrar dados e carregamento em diversos computadores, redistribuindo documentos automaticamente. Isso permite aos desenvolvedores focar na programação do aplicativo, não no dimensionamento. Quando eles precisam mais capacidade, eles podem simplesmente adicionar novos computadores e deixar o banco de dados organizar tudo.

De acordo com JOSE; ABRAHAM (2017) explora os méritos do MongoDB destacando algumas características:

- Flexibilidade: Esquema não é rígido, ou seja, não exige uma estrutura fixa como ocorre nos bancos relacionais. Armazena os dados em documentos e é facilmente mapeando nas linguagens de programação atuais.
- Consultas: Consultas dinâmicas. Possibilidade de ordenação, indexação e fácil agregação das informações armazenadas.

- Particionamento: Facilidade na escalabilidade horizontal, pois suporta a adição de novas máquinas na arquitetura.

- Facilidade de uso: Disponível gratuitamente e é de fácil instalação, configuração e utilização.

- Alta performance: Fornece consultas rápidas suportadas pelos documentos armazenados reduzindo as entradas e saídas no banco de dados

2.4.2 Tendermint

A capacidade de tolerar máquinas que falham de maneira arbitrária é conhecida como tolerância a falhas bizantina. Essa teoria tem décadas, mas as implementações de software só se tornaram populares recentemente, devido em grande parte ao sucesso da blockchain (KWON, 2014).

Na implementação do modelo nesta dissertação o software que fará a comunicação entre os nós validadores dentro da blockchain será o Tendermint. Além da comunicação, o Tendermint será responsável pelo efetuar o consenso de qual bloco será adicionado no final da cadeia.

O Tendermint é um software para replicação segura e consistente de dados entre muitas máquinas. O Tendermint utliza um algoritmo de consenso BFT que funciona mesmo que até 1/3 das máquinas falhem de maneira arbitrária. Ele consiste em dois componentes técnicos principais: um mecanismo de consenso em blockchain, chamado Tendermint Core, que garante que as mesmas transações sejam registradas em todas as máquinas na mesma ordem; e uma interface de aplicativo, genérica denominada Application BlockChain Interface, que permite que as transações sejam processadas em qualquer linguagem de programação. Ao contrário de

outras soluções de blockchain e consenso, que vêm pré-empacotadas com máquinas de estado incorporadas, os desenvolvedores podem usar o Tendermint para replicação de aplicativos escritos em qualquer linguagem de programação e ambiente de desenvolvimento adequado para eles(KWON, 2014).

No Tendermint os participantes são chamados de validadores, e eles se revezam propondo blocos de transações e votando neles. A adição de um bloco novo na cadeia ocorre através de rodadas (*rounds*) no Tendermint. Em cada rodada o bloco pode ter alguns estados possíveis, que são: nova altura (*NewHeight*), proposta (*Propose*), pré-voto (*Prevote*), pré-confirmação (*Precommit*), e confirmação (*Commit*). Os estados também são chamados de passo (*step*). Um bloco a ser validado esta em uma determinada altura (tamanho da blockchain), rodada e passo. Um bloco é dito efetivado pelo Tendermint quando assinado e difundido por mais de 2/3 do poder de votação dos validadores (MITRA, 2019).

A Figura 2.8 ilustra como o Tendermint trabalha com os estados. Inicialmente, é designado um proponente entre os validadores, que transmite a todos os validares uma proposta de um novo bloco a ser inserido no final da blockchain (MITRA, 2019). No início do pré-voto cada validador deve tomar uma decisão. Se o validador receber uma proposta aceitável para a rodada atual, ele assina e transmite uma previsão para o bloco proposto. Se o validador não receber nenhuma proposta ou uma proposta inválida, o validador assina um valor nulo para o bloco. Durante o pré-voto todos os validadores transmitem seus votos entre si (KWON, 2014). No início do passo de pré-confirmação, cada validador toma uma nova decisão, se recebeu mais de 2/3 da previsão para um determinado bloco aceitável, o validador assina e transmite uma pré-efetivação para o bloco. No final da pré-confirmação, se o bloco recebeu 2/3 de votos válidos, o bloco avança para a confirmação (KWON, 2014). Na confirmação, primeiramente, os validadores devem receber o bloco que foi pré-confirmado. Feito isso, eles assinam e transmitem sua confirmação. Logo

após, deve ser esperado receber pelo menos 2/3 de pré-confirmações para o bloco e somente assim o bloco é adicionado na blockchain. Finalizando, a cadeia de blocos é incrementada em mais 1 para mostrar que o bloco foi adicionado e transmite o novo estado (KWON, 2014).

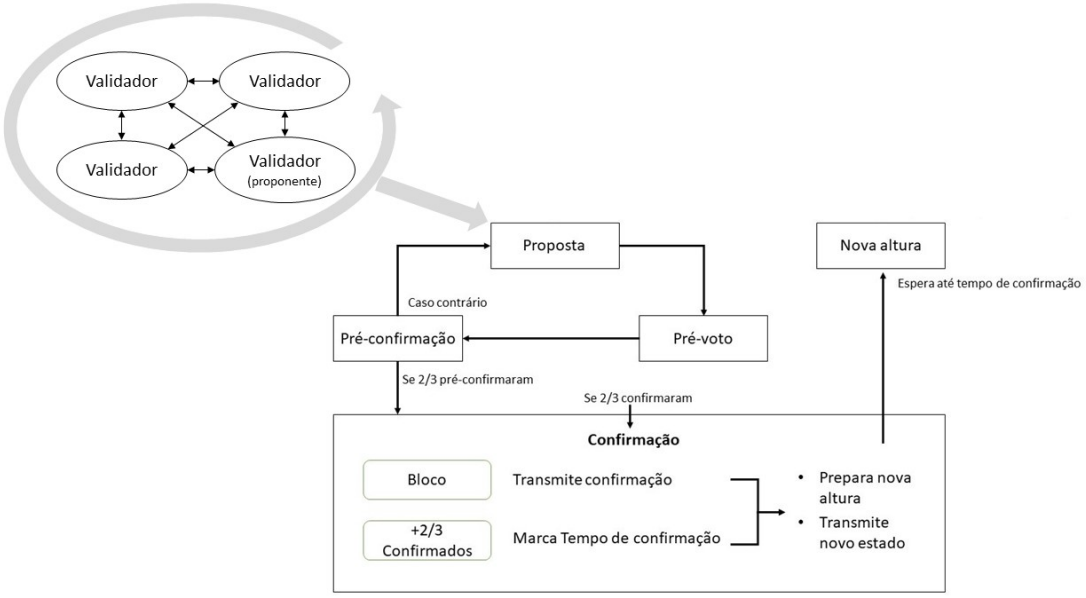


Figura 2.8: Etapas de uma rodada do Tendermint. Adaptado de KWON (2014).

2.5 Prontuários médicos Orientados ao Problema

Com intuito de facilitar as consultas de dados dentro dos prontuários, o modelo desenvolvido armazenará os prontuários de forma estruturada. Assim, quando necessário efetuar uma pesquisa no prontuário, será possível direcionar as condições de busca para um determinado conteúdo.

No final da década de 60, Lawrence Weed (JAROUDI; PAYNE, 2019) lançou uma proposta para organização interna do prontuário. Sua abordagem ficou co-

nhecida como Registro médico Orientado ao Problema (*Problem-Oriented Medical Record*, POMR). Ele defendia que todas as anotações, terapêuticas, diagnósticos, deveriam ser relacionados a um problema específico de saúde, constituindo assim, uma lista de problemas, numa árvore hierárquica na qual cada problema seria um ramo principal. No POMR, cada registro de evolução, resultado de exame, é armazenado de forma indexada por problema, de acordo com a estrutura SOAP (COSTA, 2001):

S - *Subjective*: descreve a queixa principal do paciente. O médico pode registrar o histórico da doença atual, que o paciente descreverá. Alguns pontos poderão ser incluídos como:

- sintomas relevantes
- histórico cirúrgico
- histórico familiar
- medicamentos atuais

O - *Objective*: sinais observados pelo médico, tais como peso, altura, sinais vitais e medições;

A - *Assessment*: resultados de exames e conclusões, tal como um diagnóstico ou progresso após um tratamento;

P - *Plan*: próximos passos para o paciente. Esta etapa pode incluir a atribuição de tratamentos, solicitação de testes adicionais ou referências a especialistas.

O problema com o formato SOAP é que ele não reconhece o efeito crucial da última avaliação. Ao começar uma nova anotação, colocar os dados na no Objetivo dá a impressão de que o médico não olha os dados antes de questionar o

paciente. Este não é o caso. Os médicos, sempre que possível, examinarão todos os dados relacionados ao problema antes de consultar o paciente (MEYERS; MILLER; NAEYMI-RAD, 1998).

Apesar da utilização da estrutura SOAP ocorrer a mais de quatro décadas e existirem alguns pontos a serem avaliados, existe uma estratégia do Ministério da Saúde Brasileira que o utiliza. O e-SUS AB é uma estratégia do Departamento de Atenção Básica (DAB) para reestruturar as informações da Atenção Básica (AB) em nível nacional (BRASIL, 2019). O e-SUS AB faz referência ao processo de informatização qualificada do Sistema Único de Saúde (SUS) em busca de um SUS eletrônico (e-SUS) e tem como objetivo concretizar um novo modelo de gestão de informação que apoie os municípios e os serviços de saúde na gestão efetiva da AB e na qualificação do cuidado dos usuários (BRASIL, 2019).

O modelo adotado pelo Sistema e-SUS AB para estruturação da funcionalidade de Prontuário Eletrônico do Cidadão tem como elemento central a estrutura SOAP (BRASIL, 2019).

2.5.1 Interoperabilidade por padrões médicos

A área médica está mudando continuamente, por isso é muito difícil definir uma representação de informação padronizada que seja válida para todos os dados que precisam ser armazenados (FRADE et al., 2013). Além de um padrão para armazenar as informações, há também, a questão dessas informações interoperarem entre as instituições para atender as necessidades dos pacientes.

Existem padrões aplicados no campo médico para cobrir aspectos da comu-

nicação de dados médicos. Health Level 7 (HL7)[13] é uma organização sem fins lucrativos de desenvolvimento de padrões que foi criada em 1987 para desenvolver padrões para sistemas de informação hospitalar. A HL7 é uma comunidade internacional de especialistas em informação de saúde que colaboram para desenvolver padrões para o intercâmbio de informação em saúde e interoperabilidade dos sistemas de saúde (BENDER; SARTIPI, 2013). Em 1995, HL7 v3 fora criada, na qual busca estruturar todas as informações como por exemplo, entidades em funções, participantes em atos, através de funções específicas. Embora tenha uma abordagem orientada a objetos e a linguagem de marcação XML, HL7 v3 não foi adotado pela indústria devido a sua complexidade (BENDER; SARTIPI, 2013).

Outro padrão desenvolvido pela HL7 é FHIR [14]. Ele utiliza interface REST e tenta representar todos as entidades e procedimentos da saúde como recursos. Entre os recursos mais usados no FHIR são o paciente, medicação, médico, dispositivo e observação. A ideia básica por trás do FHIR é construir um conjunto básico de recursos que, individualmente ou combinados, atendam a muitos casos. Os recursos do FHIR querem definir a estrutura e o conteúdo de informações que são transmitidas entre sistemas. Para o resto das informações, que não são cobertas pelas especificações do FHIR, existem extensões que podem ser usadas para construir dados personalizados (FRADE et al., 2013).

Outra organização que visa padronizar a comunicação de dados médicos é a openEHR [15]. Ela é uma organização independente e sem fins lucrativos que facilita a criação e transmissão de informações médicas baseadas em uma implementação de código aberto. As recomendações openEHR são baseadas em um modelo para atender requisitos éticos e legais, modelos para características de dados clínicos e modelos para estruturas complexas, tais como documentos ou relatórios, dependendo

[13] https://www.hl7.org/
[14] https://hl7.org/FHIR/
[15] https://www.openehr.org/

do caso de uso (FRADE et al., 2013).

2.6 Blockchain para prontuários médicos

As características da blockchain despertou interesse de diversas áreas, e uma delas é área da saúde. Através de HöLBL et al. (2018) e também AGBO; MAHMOUD; EKLUND (2019) ocorreu aumento de iniciativas acadêmicas e de indústrias para o uso da tecnologia de blockchain na área de saúde. Grande parte dessas iniciativas são focadas para o uso em cenários de prontuários médicos eletrônicos na blockchain.

Os prontuários médicos implementados com blockchain não dependerão de um armazenamento central e os pacientes serão os gestores de seus dados. Isso ajudará a garantir um nível mais alto de privacidade e a transferir o controle para os pacientes, além disso, pesquisadores poderão se beneficiar da disponibilidade de um amplo conjunto de dados médicos, que podem ser usados para realizar estudos sobre patologias existentes (KARAFILOSKI; MISHEV, 2017).

Vários fatores devem ser considerados na escolha da blockchain para o desenvolvimento de aplicativos na área da saúde. Ao comparar as diferentes soluções de blockchain mais comuns, é importante entender os requisitos técnicos da necessidade que se busca resolver. Em artigos como AGBO; MAHMOUD (2019) e KUO; ZAVALETA ROJAS; OHNO-MACHADO (2019b) apresenta-se o desafio de propor uma plataforma adequada e fazem comparativos entre as plataformas de blockchain mais populares atualmente com foco na área de saúde.

No contexto de um protuário médico, pode-se exemplificar a utilização da blockchain da seguinte forma: um paciente está em uma consulta com um especi-

alista A, é gerado um identificador único para o bloco e informado a data e hora de sua criação, já as informações da consulta são inseridos no bloco no campo de dados, assim como a propriedade das informações são transferidas ao paciente. O especialista submete o bloco para a blockchain, após a validação, esse faz parte da cadeia. Logo após, o paciente vai ao especialista B para outra consulta. O processo ocorre da mesma forma, porém o novo bloco a ser inserido faz referência ao último bloco inserido na cadeia. Assim, quando o paciente efetuar uma consulta, seus dados estarão cronologicamente íntegros. Como o paciente é proprietário de suas informações, ele pode disponibilizar suas informações a quem lhe interessar, como um novo especialista ou até mesmo uma instituição de pesquisas médicas.

Apesar das boas perspectivas, a blockchain ainda tem algumas limitações entre as quais o armazenamento de dados. Essencialmente, a blockchain foi projetada com uma abordagem de armazenar transações financeiras utilizando blocos com suas respectivas informações e não grandes volumes de dados como pode ocorrer em um prontuário médico.

3 TRABALHOS CORRELATOS

Neste capítulo serão descritos alguns trabalhos na área de saúde que utilizam a blockchain para seus objetivos. Serão apresentados soluções que utilizam a blockchain para a área de saúde destacando estudos relacionados a criação de prontuários eletrônicos. No final, é efetuado um comparativo das soluções apresentadas com modelo desenvolvido neste trabalho.

3.1 Blockchain na área da saúde

As revisões sistemática de HöLBL et al. (2018) e TANDON et al. (2020) efetuadas em publicações entre os anos 2015 e 2019 indicam que pesquisas na tecnologia blockchain empregada na área de saúde tem aumentado. ALONSO et al. (2019) demonstram de forma geral a viabilidade de utilização da blockchain na área da saúde nas seguintes frentes:

- Identificação dos pacientes.
- Rastreabilidade das ações médicas.
- Registros de informações médicas.
- Redução de tempo para interoperabilidade de dados.

Outras frentes que também estão interessados nas características da blockchain são por exemplo a cadeia de suprimentos e planos de saúde. As blockchains

podem fornecer um método confiável para rastrear produtos farmacêuticos durante todo o processo de fabricação e distribuição, reduzindo assim o problema de falsificação de medicamentos. Nos planos de saúde a blockchain poderia combater as fraudes através de sua característica de imutabilidade (ACADEMY, 2019).

Entretanto, um dos casos de usos mais populares da blockchain na área de saúde é a manutenção de prontuários médicos(AGBO; MAHMOUD; EKLUND, 2019). Diversos artigos descrevem soluções e experiências da utilização da blockchain para o âmbito de prontuários médicos. O MIT Media Lab desenvolveu o MedRec (AZARIA et al., 2016). Alunos de pós-graduação utilizaram a blockchain da Ethereum para criar um sistema que representa as permissões de propriedade e visualização de dados compartilhados por membros de uma rede ponto a ponto. A Figura 3.1 demonstra que na blockchain ficam armazenados a identificação do paciente, informação referente as permissões e as referências dos prontuários. Em posse das referências aos prontuários é efetuado a consulta direto no banco de dados da instituição médica. Essa abordagem pode dificultar a pesquisas genéricas dentro dos prontuários, pois cada instituição pode armazenar os prontuários de formas diferentes.

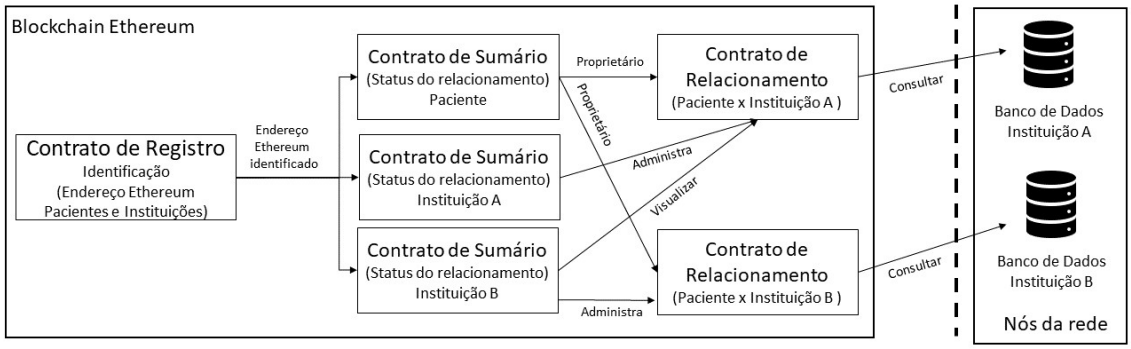


Figura 3.1: Organização do Medrec. Adaptado de AZARIA et al. (2016)

O framework Ancile (DAGHER et al., 2018) se baseia no projeto MedRec,

entretanto utiliza-se de um cliente Ethereum chamado de Quorum [1]. A proposta consiste de manter maior parte das informações dos clientes nos bancos de dados das instituições, somente armazenando os apontadores para os dados e as permissões na blockchain. A Figura 3.2 mostra também outro ponto que o Ancile se difere, que é o algoritmo de consenso, no qual não há necessidade de ser baseado em prova e sim baseado em votos.

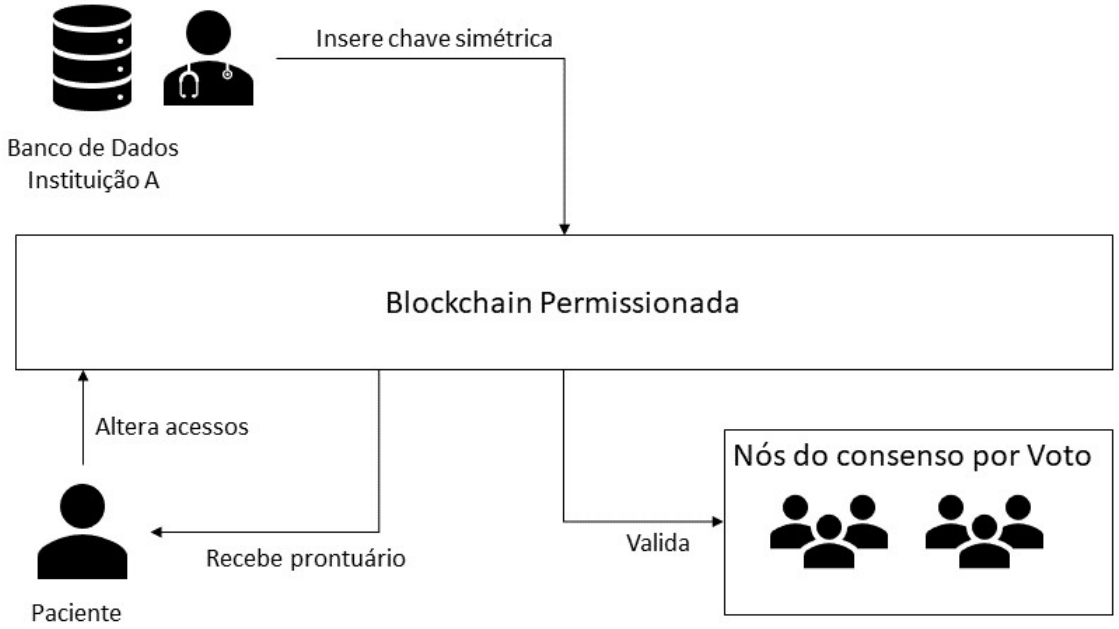


Figura 3.2: Arquitetura Ancile. Adaptado de DAGHER et al. (2018)

A Zhealth (ZHEALTH, 2020) é a primeira blockchain de saúde brasileira. Esta solução é desenvolvida na solução Hyperledger[2] que é um projeto dentro do guarda-chuva de projetos da Linux Foundation. A Zhealth tem como proposta a partir de uma rede blockchain, que confere segurança, autenticidade e controle de acesso, pretendem unificar os dados pessoais e empresariais de saúde em uma rede compartilhada por todos os participantes. Em sua essência a Zhealth se utiliza da blockchain para solucionar o problema de alteração indevida dos dados.

[1] https://consensys.net/
[2] https://www.hyperledger.org/

O modelo UniRec (QUAINI; ROEHRS; RIGHI, 2018) consiste em integrar, rastrear e segurar prontuários médicos dentro de uma blockchain privada. Através da blockchain Ethereum e de um sistema de arquivos ponto a ponto os dados do paciente são compartilhados entre as instituições de saúde. Na Figura 3.3 é possível verificar a utilização de um sistema de arquivos (IPFS) para armazenamento dos prontuários. Dessa forma, uma busca de informações específicas também ficam comprometidas, pois os arquivos podem ser dos mais diversos formatos.

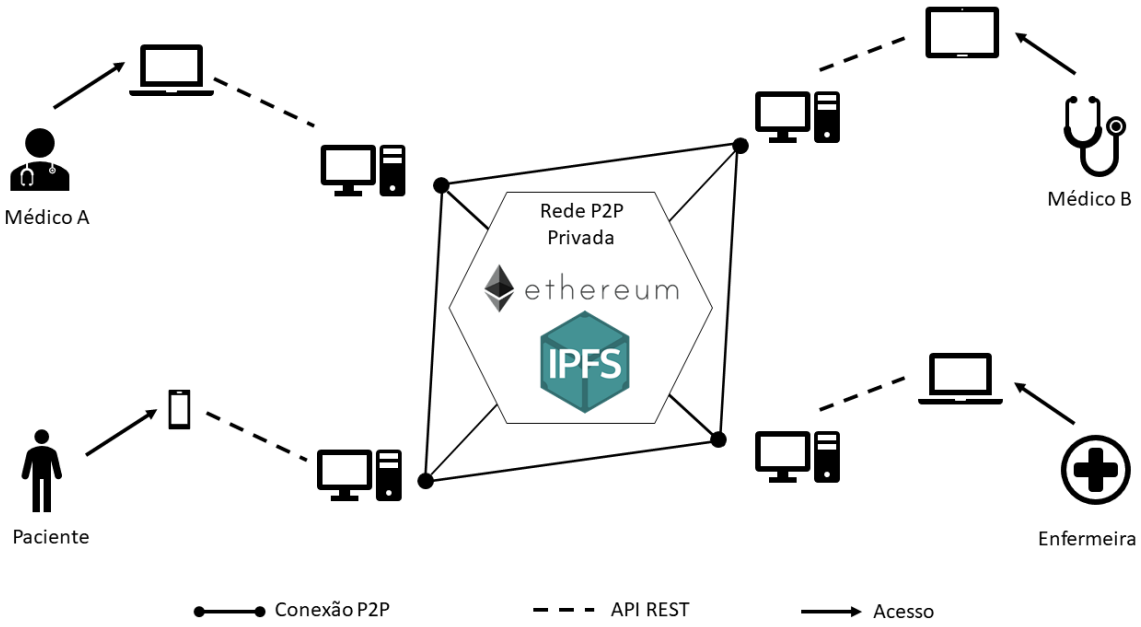


Figura 3.3: Arquitetura UniRec. Adaptado de QUAINI; ROEHRS; RIGHI (2018)

MediChain (ROUHANI et al., 2018) é uma implementação baseada na solução do Hyperledger dividido em módulo de controle de acesso, armazenamento na nuvem e interface com usuário. Os arquivos são criptografados e armazenados em um repositório na nuvem, mantendo somente o vínculo dos arquivos na MediChain.

Existe o protótipo RAMESH (2018), que efetua o processo de criptografia e utiliza o armazenamento dos arquivos em sistemas de arquivos e guardando o apontamento em banco de dados. Já o framework proposto por BELLOD CISNEROS;

AARESTRUP; LUND (2018) utiliza a blockchain para um cenário de dados genômicos na qual os metadados são armazenados em banco de dados e os arquivos em um sistema de arquivos ponto-a-ponto.

A Tabela 3.1 apresenta um comparativo das contribuições descritas anteriormente e o modelo desenvolvido neste trabalho.

Tabela 3.1: Comparativo das contribuições avaliadas e o modelo proposto neste trabalho

	Moeda	Consenso	Blockchain	Armazenamento	Open Source	Prontuário Estruturado	Paciente Proprietário	Contribuição
MedRec	Sim	Prova	Permissionada	Banco de dados relacional	Sim	Não	Sim	Projeto/Implementação
Ancile	Sim	Voto	Permissionada	-	Sim	Não	Sim	Framework
UniRec	Sim	Prova	Permissionada	Sistema de arquivos	Sim	Não	Sim	Modelo/Protótipo
Zhealth	Não	-	Permissionada	-	Não	Não	Sim	API/Implementação
Medichain	Não	-	Permissionada	Nuvem	Sim	Não	Sim	Implementação
Medblocks	Não	Voto	Permissionada	Sistema de arquivos	Sim	Não	Sim	Protótipo
Modelo desenvolvido	**Não**	**Voto**	**Permissionada**	**Banco de dados de documentos**	**Sim**	**Sim**	*Sim*	**Modelo/Protótipo**

Com o modelo desenvolvido, pretende-se avançar na criação de prontuários de pacientes, tratando todo o processo como ativos com seus respectivos proprietários. Os pacientes, as instituições médicas e de pesquisa, prontuários dos pacientes e as permissões de visualização serão consistidos na blockchain através de um consenso baseado em votos e armazenados em um banco de dados orientado a documentos. Outro destaque do modelo é o armazenamento dos prontuários seguindo um estrutura orientada ao problema. Isso facilitará as consultas específicas quando necessário, buscando as informações de forma mais assertivas de forma geral. Adicionalmente, como contribuição secundária, uma arquitetura para implementação do modelo é apresentada.

4 MODELO PARA CRIAÇÃO DE PRONTUÁRIO ELETRÔNICO DO PACIENTE BASEADO EM BLOCKCHAIN

Nesse capítulo será apresentado a visão geral do modelo desenvolvido bem como uma arquitetura para sua implementação. Também serão apresentados os ativos e os processos para contemplar a criação de um prontuário eletrônico de paciente baseado em blockchain.

4.1 Modelo para prontuário eletrônico

O modelo desenvolvido consiste em oito tipos de ativos. Os ativos serão criados com características distintas, entretanto, funcionarão integrados. São eles:

Paciente: Terá um identificador que será utilizado para efetuar buscas, podendo ser, por exemplo, o CPF do paciente. Também será guardado as suas chaves públicas para efetuar as transações e criptografia;

Dados do paciente: Conterá informações como nome, data de nascimento, sexo entre outras. Esse ativo será somente criado pelo paciente e quando houver necessidade de atualização, o paciente deverá criar um novo ativo desse tipo com as novas informações;

Responsáveis: Utilizado para que o paciente possa vincular outras pessoas para também serem co-responsáveis de suas informações. Exemplo é pai e mãe de uma criança, ou um paciente com incapacidade;

Instituição: Ativo de cada um das instituições de saúde ou pesquisa que irão utilizar o sistema. Também serão guardadas as suas chaves públicas para efetuar as transações e criptografia;

Prontuário: Terá as informações do prontuário do paciente na estrutura SOAP;

Acesso: Criado após a criação do ativo do prontuário e também quando o paciente desejar dar acesso ao prontuário a alguém. Armazenará o identificador do ativo prontuário e todas as chaves públicas que têm acesso ao prontuário;

Anexo: Armazenará os apontamentos dos arquivos que sejam vinculados ao prontuário do paciente. Esses apontamentos podem ser através de um sistema de arquivos ou utilizando algum mecanismo dentro da base de dados diretamente;

Admin: Serão vinculados a todos os ativos de prontuários criados. Esse ativo será criado somente em tempo de configuração da blockchain. Nele serão armazenadas as chaves públicas dos nós configurados para efetuar o consenso;

A Figura 4.1 apresenta o relacionamento entre os ativos. As setas pretas informam qual é a informação que os ativos terão em comum para ocorrer o vínculo entre eles. As setas em verde demonstram qual é a chave privada que deve ser informada para a criação do ativo, caso a chave privada informada não seja par da chave pública, o ativo não será criado. Já a seta vermelha demonstra a transferência do ativo de prontuário para o paciente, denotando a propriedade do prontuário ao paciente. O único ativo que poderá ser transferido é o ativo do prontuário.

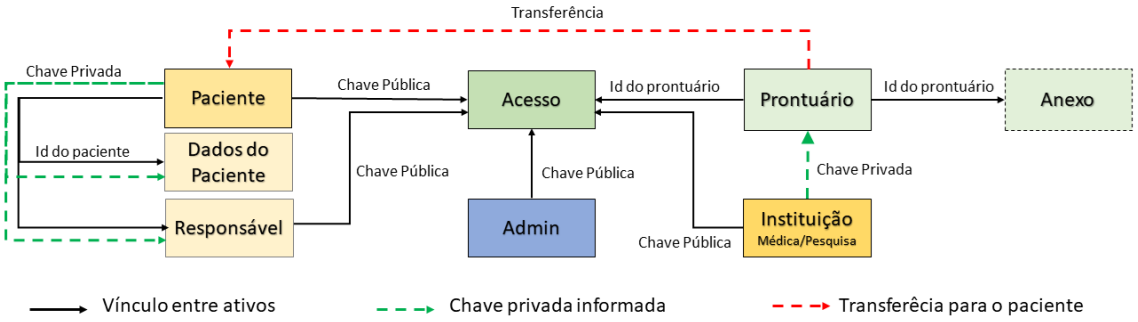


Figura 4.1: Tipos de ativos dentro do modelo e seus relacionamentos

A Figura 4.2 apresenta o processo para cadastrar um paciente. Após informar seu identificador, o sistema gera um par de chaves e disponibiliza através de arquivos para o paciente guardar e a criação do ativo é submetido na blockchain. Após o consenso, o ativo já esta armazenado e é vinculado um identificador único a esse ativo, assim o sistema solicita se o paciente deseja informar seus dados pessoais. Caso o paciente informe seus dados, um ativo do tipo dados do paciente será criado. Em seguida, ocorre o mesmo processo para o ativo de responsáveis. Se o paciente desejar vincular outras pessoas como co-responsáveis dos seus dados o sistema irá solicitar as informações e criar o ativo dentro da blockchain.

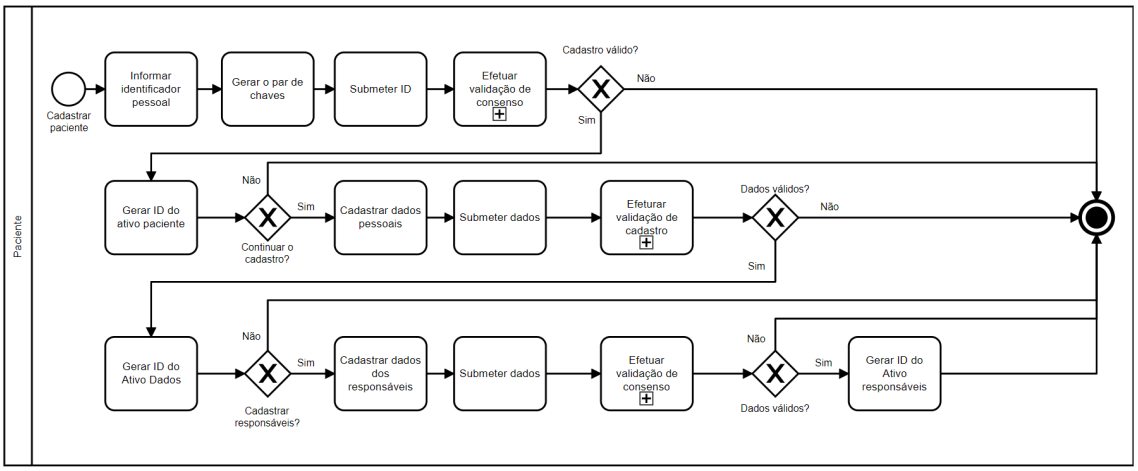


Figura 4.2: Processo para cadastro do paciente

Nos ativos de dados do paciente e responsáveis será acrescido um campo de

data em suas criações. Assim, o paciente ao buscar esses ativos será limitado ao dado mais atual. Alguns dados podem mudar com o passar do tempo e podem ser atualizados nos cadastros de dados, como por exemplo peso e altura, ou até mesmo o responsável pelo paciente. Essas informações poderão ser atualizadas.

A Figura 4.3 apresenta a criação do ativo do tipo instituição de saúde e de pesquisa. Cada ativo criado tem um par de chaves associado que serão disponibilizados através de arquivos e será submetido a consenso na blockchain, atingido o consenso, o ativo é armazenado com seu identificador único.

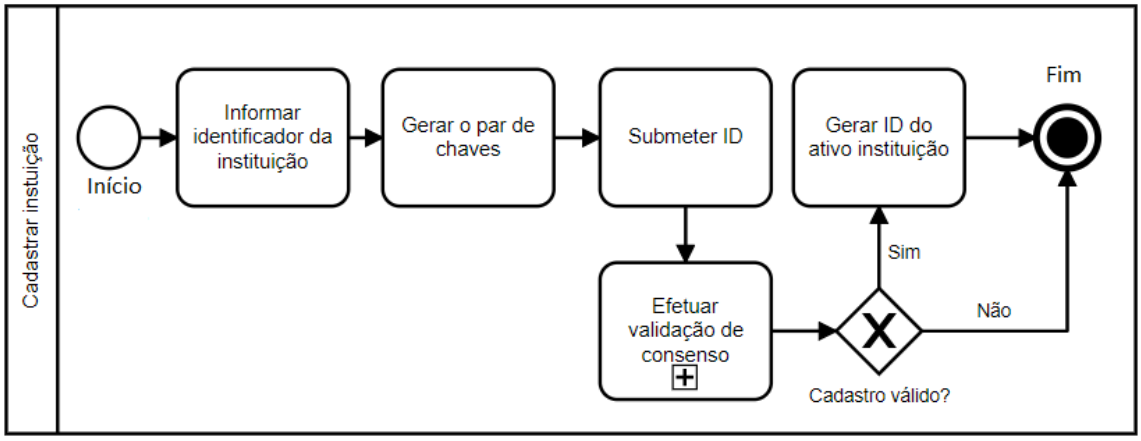


Figura 4.3: Processo de cadastro de uma instituição

Com os ativos do paciente e das instituições na blockchain, já é possível criar um ativo do tipo prontuário. O ativo de prontuário seguirá a abordagem utilizada para a confecção de um prontuário de registro médico orientado ao problema. Nessa abordagem os registros são armazenados na estrutura SOAP (COSTA, 2001):

S - Subjective: sintomas do paciente;

O - Objective: sinais observados pelo médico;

A - Assessment: resultados de exames e conclusões, tal como um diagnóstico;

P - Plan: conduta, por exemplo um tratamento.

A Figura 4.4 descreve o fluxo para a criação do ativo de prontuário na blockchain. O paciente vai até uma instituição para efetuar algum procedimento médico. A instituição acessa o sistema e informa seu identificador (1), após o sistema encontrar a instituição é solicitado a identificação do paciente (2). Com os atores identificados, o sistema solicita os dados estruturados do prontuário (3). Em seguida, o sistema solicita para a instituição os arquivos com as chaves privadas para submeter o prontuário para a blockchain (4). O sistema efetua validações, como por exemplo, se o par de chaves informado é válido. Após, solicita o consenso da transação (5). Obtido o consenso, o sistema gera o identificador do ativo (6). Com o ativo criado, o sistema submete a transferência do prontuário ao paciente através de uma nova transação (7). Obtido o consenso da transação de transferência (8), o sistema conclui o processo (9).

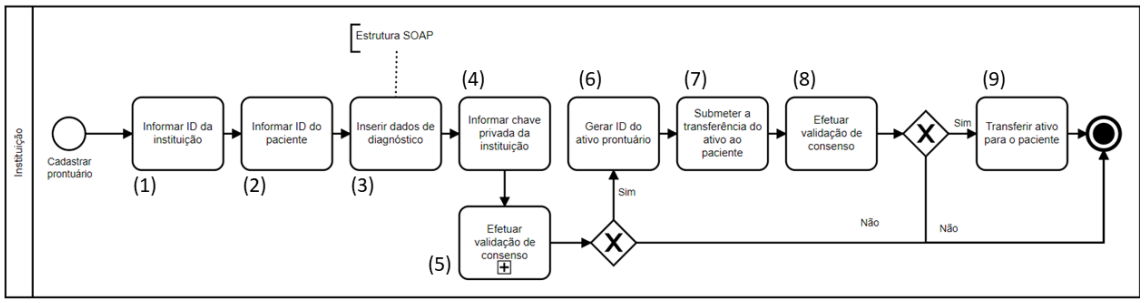


Figura 4.4: Processo para cadastro prontuário do paciente

O ativo de prontuário terá uma característica de múltiplos donos, ou seja, quando for criado o ativo e transferido ao paciente, o sistema vai verificar se para o paciente existem outros responsáveis (ativo de responsáveis) e caso existam eles deverão ser registrados no ativo como proprietários também.

Outra característica do ativo de prontuário é que eles serão criptografados. Antes da instituição submeter ativo de prontuário para a blockchain, é gerado uma

chave de sessão de forma aleatória e essa é utilizada para criptografar o ativo de forma simétrica. Logo após, a chave de sessão utilizada será criptografada de forma assimétrica utilizando as chaves públicas de todos os donos, da instituição que executou o procedimento criando o ativo de acesso. O ativo do tipo acesso terá o identificador do ativo prontuário e todas as chaves públicas que têm acesso ao prontuário.

O ativo de acesso também será criado quando o paciente desejar permitir que outras pessoas vejam as informações do seu prontuário. A Figura 4.5 descreve um paciente (1) dando permissão a uma instituição que deseja visualizar seu prontuário. Através de uma lista dos identificadores dos prontuários (2), o paciente seleciona qual irá conceder permissão (3). O sistema solicita o identificador da instituição (4) e a chave privada do paciente (5). É validado se as chaves do paciente são válidas e se o prontuário pertence ao paciente. O sistema localiza o ativo de acesso do paciente para o prontuário em questão (6). Utilizando a chave privada do paciente é possível descriptografar a chave de sessão que esta no ativo (7). Utilizando a chave pública da instituição, o sistema criptografa a chave de sessão (8) e submete o ativo de acesso para a blockchain (9). Obtido o consenso o ativo de acesso é criado (10) .

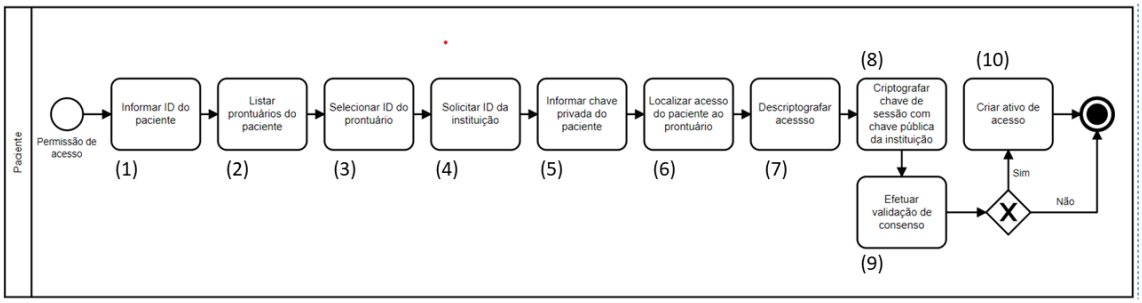


Figura 4.5: Fluxo dar acesso

Em caso de necessidade de armazenar arquivos de vídeo, imagens ou som é sugerido armazenar o arquivo em um sistema de arquivos e somente o apontador dele dentro da base de dados. Isso auxilia a minimizar problemas de indexação

e pesquisa na base de dados. Um exemplo dessa solução é possível analisar no artigo BELLOD CISNEROS; AARESTRUP; LUND (2018), onde são tratados e armazenados dados de DNA. Também é possível utilizar algum mecanismo para armazenamento dos arquivos dentro da base de dados de forma fragmentado. No modelo é possível criar um ativo do tipo anexo. Nele terá o identificador do prontuário e o apontador para o arquivo armazenado em um sistema de arquivos ou em tabelas do banco de dados. Com o banco de dados MongoDB é possível utilizar o *GridFS*[1] que armazena o arquivo em fragmentos dentro do banco de dados e cria um identificador apontando para todos os fragmentos.

4.2 Implementação do modelo

A implementação permitiu a criação de todos os atores de um cenário de prontuários médicos eletrônicos e submetidos em uma blockchain permissionada e também para atender os processos descritos na Seçao 4.1. A escolha de um blockchain pública permissionada garante que sejam conhecidos os validadores da rede e que esses tenham total interesse que as informações submetidas sejam confiáveis.

A Figura 4.6 mostra uma visão geral do modelo desenvolvido neste trabalho. O paciente e uma instituição médica (1) já adicionados na blockchain geram o prontuário (2). Havendo arquivos a serem adicionados, são inseridos apontamentos deles no prontuário (2.1). O prontuário é criptografado e armazenado no banco de dados (3). A chave utilizada na criptografia do prontuário é criptografada de forma assimétrica utilizando as chaves públicas de todos os proprietários do prontuário, criando o ativo de acesso (4). Caso seja solicitado visualização(5) por uma nova instituição (6) é necessário criar um novo ativo do tipo acesso (7).

[1] https://docs.mongodb.com/manual/core/gridfs

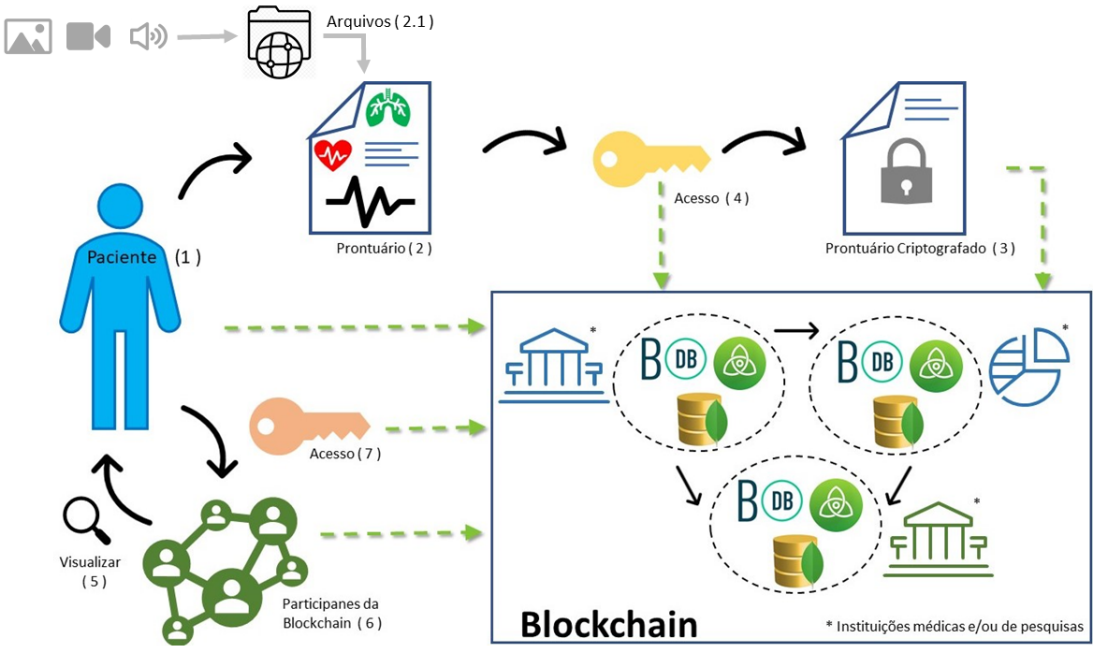


Figura 4.6: Visão geral do modelo desenvolvido

Vislumbrando melhor utilização do modelo desenvolvido e possíveis integrações a outros sistemas, partiu-se para a criação de uma Interface de Programação de Aplicativos (*application programming interface*, API). API REST, é uma arquitetura para fornecer padrões entre sistemas de computador na web, tornando mais fácil a comunicação entre sistemas. Esse tipo de interface é baseada no protocolo HTTP e basicamente utiliza quatro comandos para interagir:

GET: retornar um registro específico ou uma coleção;

POST: criar um novo registro;

PUT: atualizar um registro. Dentro de uma blockchain não é possível efetuar alteração de registros, isso afeta a característica de imutabilidade de uma blockchain;

DELETE: remover um registro. Também não pode ser utilizada dentro da blockchain por causa da característica de imutabilidade dos registros.

Foi utilizado a linguagem de programação Python [2] em conjunto com framework [3] Flask para o desenvolvimento. O Flask é um micro framework, o qual visa ter suas funcionalidades simples mas de forma extensível. Uma das extensões que o Flask disponibiliza é *Flask-RESTPlus* [4]. Através de configurações é possível construir uma API REST e expor sua documentação de fácil entendimento.

Através do *Flask-RESTPlus* foram criadas as interfaces necessárias para atender a proposta deste trabalho. A Figura 4.7 apresenta a lista de interfaces criadas, bem como a documentação criada para auxiliar o entendimento de cada uma. Em cada uma é possível verificar o formato do arquivo de entrada e de saída pré-configurados. Ainda é possível efetuar testes unitários em cada uma das interfaces.

Figura 4.7: Lista de Interfaces criadas para interação do modelo desenvolvido

Com as interfaces criadas, se fez necessário criar uma aplicação para utilizá-las e efetuar a interação com os usuários finais. Para a criação da aplicação foi utilizado outra extensão do Flask chamada de *WTForms* [5]. Essa extensão é utilizada

[2] https://www.python.org/downloads/
[3] é um conjunto de bibliotecas que auxiliam o desenvolvimento de aplicações.
[4] https://flask-restplus.readthedocs.io/en/stable/
[5] https://flask.palletsprojects.com/en/1.1.x/patterns/wtforms/

para trabalhar com formulários que podem validar e submeter as informações para as interfaces criadas. Primeiramente foi criada uma tela inicial para diferenciar o perfil do usuário, seja paciente ou uma instituição. Através da Figura 4.8 é possível verificar as funcionalidades de cada perfil após direcionamento da aplicação.

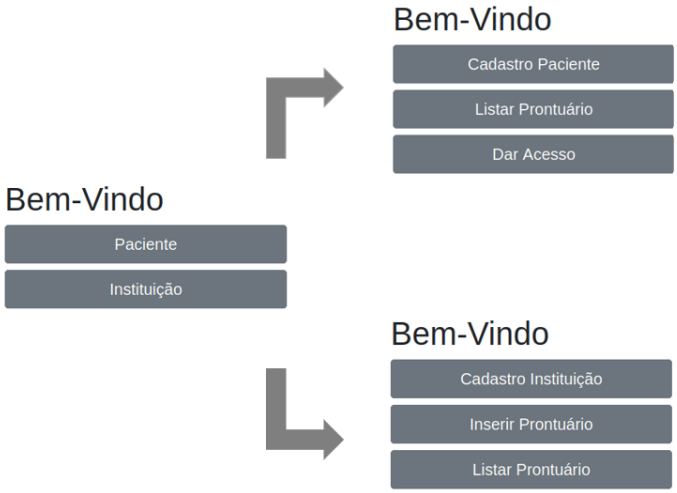


Figura 4.8: Tela inicial da aplicação com direcionamento conforme perfil

Na criação de um paciente a aplicação solicita o identificador e o nome do paciente. Após é necessário submeter às informações na blockchain. A Figura 4.9 apresenta em caso de sucesso a criação do paciente através do identificador da transação e também é necessário efetuar o *download* do arquivo contendo as chaves privadas do paciente. Esse arquivo deverá ser guardado para futura identificação dentro da blockchain. Através desse processo o paciente estará validado e armazenado na blockchain. De forma semelhante ocorre o cadastro de uma instituição médica ou de pesquisa dentro da blockchain. Para a instituição também é disponibilizado arquivo com as chaves privadas que deverão ser guardadas em sigilo.

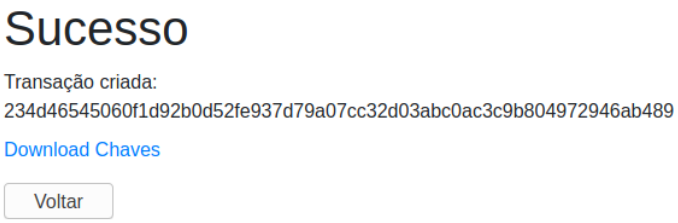


Figura 4.9: Tela de paciente criado com sucesso

A opção de inserir um prontuário é somente disponibilizada no perfil da instituição. Nessa opção a aplicação solicita o identificador da instituição e do paciente. Com essas informações a aplicação identifica os atores e guarda as chaves públicas de cada um. A Figura 4.10 demonstra a tela de cadastro do prontuário na estrutura SOAP e também um campo de cabeçalho para auxiliar nas visualizações futuras.

Figura 4.10: Tela de cadastro de prontuário do paciente com sucesso

Nesse momento a instituição deve informar seu arquivo contendo as chaves privadas para que assim seja possível validar sua identificação. Após, a aplicação submete o prontuário para a blockchain e em caso de sucesso retorna o identificador do prontuário dentro da blockchain. Vale lembrar que no momento da submissão do prontuário é feito a transferência da propriedade para o paciente e também criado o ativo de acesso.

Para listar os prontuários de um paciente específico é utilizada a opção de listar prontuários, que está disponível tanto para o perfil de paciente quanto instituição. Nessa funcionalidade a aplicação solicita o identificador do paciente e da instituição. Com esta informação, a aplicação retorna todos os prontuários do paciente caso existam. Como é possível verificar na Figura 4.11, a lista retorna somente o identificador e cabeçalho do prontuário. Dessa forma, não é possível identificar o paciente e muito menos as informações contidas no prontuário. Para que ocorra a visualização do conteúdo do prontuário é necessário ter acesso. Sendo assim, a aplicação verifica se existe um ativo do tipo acesso vinculado ao prontuário requerido e que contenha a chave pública do solicitante. Caso exista, a aplicação solicita o arquivo de chave privada para efetuar a descriptografia das informações contidas no prontuário do paciente.

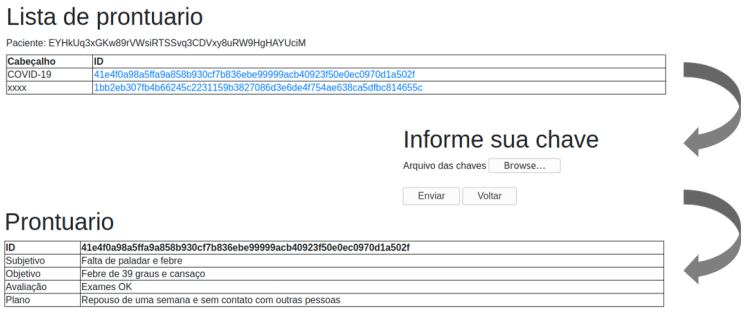


Figura 4.11: Tela com a lista e visualização dos prontuários do paciente

O paciente tem a possibilidade de dar acesso a qualquer instituição médica ou de pesquisa. Para que isso ocorra, a aplicação solicita o identificador do paciente e retorna os prontuários existentes. Na lista é possível selecionar o prontuário que deseja dar acesso. Depois de selecionado, a aplicação requisita o identificador da instituição que o paciente irá dar acesso, também deve ser informado o arquivo de chave privada do paciente. Caso ocorra tudo com sucesso, é criado um novo ativo do tipo acesso e assim a nova instituição poderá visualizar o prontuário do paciente.

4.3 Ambiente operacional

Para a implementação da solução, inicialmente foi criado uma máquina virtual utilizando *Oracle VM Virtual Box* [6] com o sistema operacional *Ubuntu 20.04.1 LTS* [7].

Depois de efetuado a instalação do sistema operacional, foram liberadas algumas portas para todo o tráfego de entrada e saída de informações. Também foi configurado um endereço de IP estático para a máquina virtual, isso se fez necessário para que ocorra a identificação da máquina. Em seguida, foi efetuada instalação e configuração do *node* que é composto pelo BigchainDB, Tendermint e MongoDB.

Neste trabalho foi instalado a última versão estável do BigchainDB (versão 2.2.1 [8]). Através do tutorial disponível (BIGCHAINDB, 2019), foi possível efetuar a instalação e configuração. Em seguida, foi instalado o banco de dados MongoDB [9]. O BigchainDB necessita a versão do MongoDB igual ou superior a 3.4. A versão do Tendermint utilizada e sugerida pelo BigchainDB é a 0.31.5.

Conforme a Seção 2.4.2, o Tendermint é um software para replicação em muitas máquinas e funciona mesmo que até 1/3 das máquinas falhem. Para que possa ser validada sua funcionalidade é sugerido que no mínimo sejam configuradas quatro *nodes* na blockchain. Dessa forma, a blockchain irá funcionar mesmo que uma das máquinas falhe, somente quando duas ou mais máquinas falharem o Tendermint não alcançará o consenso.

[6] https://www.oracle.com/virtualization/technologies/vm/downloads/virtualbox-downloads.html
[7] https://ubuntu.com/download/desktop
[8] https://pypi.org/project/BigchainDB/
[9] https://www.mongodb.com/try

Para que os *nodes* se comuniquem e façam o compartilhamento das informações dentro da blockchain é necessário configurar os membros validadores. Esses membros serão responsáveis pelo consenso dentro da blockchain. Cada *node* é identificado pelo seu endereço de IP, sua chave pública e identificador do Tendermint. A chave pública de cada *node* é obtido no arquivo *priv_validator.json* dentro do diretório de configuração da instalação do Tendermint. Já o identificador do Tendermint no *node* é encontrado através do comando *node_id* no terminal do sistema operacional.

Com posse destas informações, todos os *nodes* compartilham com os demais membros para que assim possam criar e configurar a blockchain e o bloco inicial da cadeia. Essa configuração é efetuada no arquivo *genesis.json* no diretório de configuração do Tendermint. Deve ser informado o nome da blockchain que está sendo configurada e também as chaves públicas dos membros que farão as validações da blockchain. Após inserir as informações no arquivo *genesis.json* ele deve ser compartilhado com os membros validadores da blockchain.

Nesse momento todos os membros da blockchain compartilham da mesma blockchain, faltando somente informar como os membros são encontrados. Para isso no arquivo *config.toml* dentro do diretório de configuração do Tendermint é necessário informar os endereços IPs e os identificadores de cada membro. Esta informação deve ser inserida em todos os membros no formato *node_id@IP*.

Com as instalações e configurações efetuadas é possível iniciar os serviços do BigchainDB e Tendermint para que possam a ser utilizados na aplicação desejada. O BigchainDB sugere a utilização de um monitor de processos para gerenciamento automático. Esses monitores são interessantes, pois podem ser configurados para que caso ocorra algum problema os processos possam ser reiniciados automaticamente. Para o protótipo descrito na Seção 4.2 todos os serviços foram iniciados

manualmente.

4.3.1 Cenários em ambiente operacional

Os testes têm como objetivo avaliar o tempo utilizado para a criação de um prontuário do paciente dentro do modelo proposto. Para a tomada dos tempos foram criados programas que geram arquivos com os tempos de início e fim de cada execução, e os tempos inicio e fim de cada transação submetida na blockchain para a criação de um prontuário de paciente.

Foram criadas quatro máquinas virtuais. Cada máquina deve ser considerada uma instituição médica ou de pesquisa. Ainda, cada máquina é um nó na blockchain que fará a validação do consenso quando submetido uma transação para a blockchain. A quantidade de máquinas foi escolhida para testar o algoritmo de consenso em caso de falha, visto que ele deve encontrar o consenso mesmo que 1/3 das máquinas falhem. O número de nós que podem falhar, N_{falhas}, sem interromper o processamento em um cenário com N nós validadores é dado por:

$$N_{falhas} = \lfloor (N-1)/3 \rfloor \tag{4.1}$$

Vale salientar que os testes foram criados em uma rede interna configurada para que as máquinas pudessem se comunicar. Dessa forma não foi previsto nenhuma avaliação referente a possíveis interferências de comunicação entre as máquinas.

Foram criados quatro cenários de testes para avaliar tempos de execução para a criação e consulta dos prontuários. Para execução dos testes foram criados programas que efetuam as operações do protótipo criado. Além disso, a criação do paciente e da instituição foram efetuadas através do protótipo, isso se fez necessário para que as chaves privadas de cada um fossem inseridas de forma fixa nos testes.

Outro ponto a ser salientado é o conteúdo dos prontuários. Para simular o conteúdo dos prontuários foi efetuado um procedimento randômico de frases para cada um dos campos do prontuário na estrutura SOAP. Desta forma, cada prontuário terá em média cinco mil caracteres alfanuméricos. Os cenários são:

Cenário 1: Criação de 2.000 prontuários divididos em quatro máquinas de forma concorrente. Esse cenário simula quatro instituições efetuando o atendimento a pacientes de forma simultânea.

Cenário 2: Criação de 1.750 prontuários divididos em quatro máquinas de forma concorrente. Durante o processo de criação foi desligada uma máquina, ficando somente três para o restante das transações. A criação dos prontuários deve continuar, pois o consenso é encontrado mesmo que 1/3 das máquinas falhem.

Cenário 3: Criação de 2.000 prontuários divididos em quatro máquinas de forma concorrente. A quantidade de caracteres nestes prontuários foi quinze vezes maiores que do Cenário 1. O programa de criação foi alterado para que cada prontuário fosse criado contendo em média setenta e cinco mil caracteres alfanuméricos.

Cenário 4: Consulta dos 5.750 prontuários criados nos cenários anteriores. Esse cenário demonstra a consulta de prontuário tanto pelo paciente quanto por uma instituição que tenha permissão para visualizar as informações.

A Figura 4.12 apresenta os resultados dos tempos totais para a criação dos prontuários de paciente em cada um dos cenários descritos. O tempo médio nos cenários ficou em 4,1 segundos e assim é possível verificar que para os cenários 1 e 2 a maior parte das criações ficaram na média ou abaixo. Já o cenário 3, a maioria ficou acima da média do tempo. Além disso, tanto o tamanho do prontuário quanto

a queda de uma das máquinas influenciaram nos tempos finais totais. Os tempos totalizados em minutos de cada cenário ficaram em 27, 33 e 36 respectivamente.

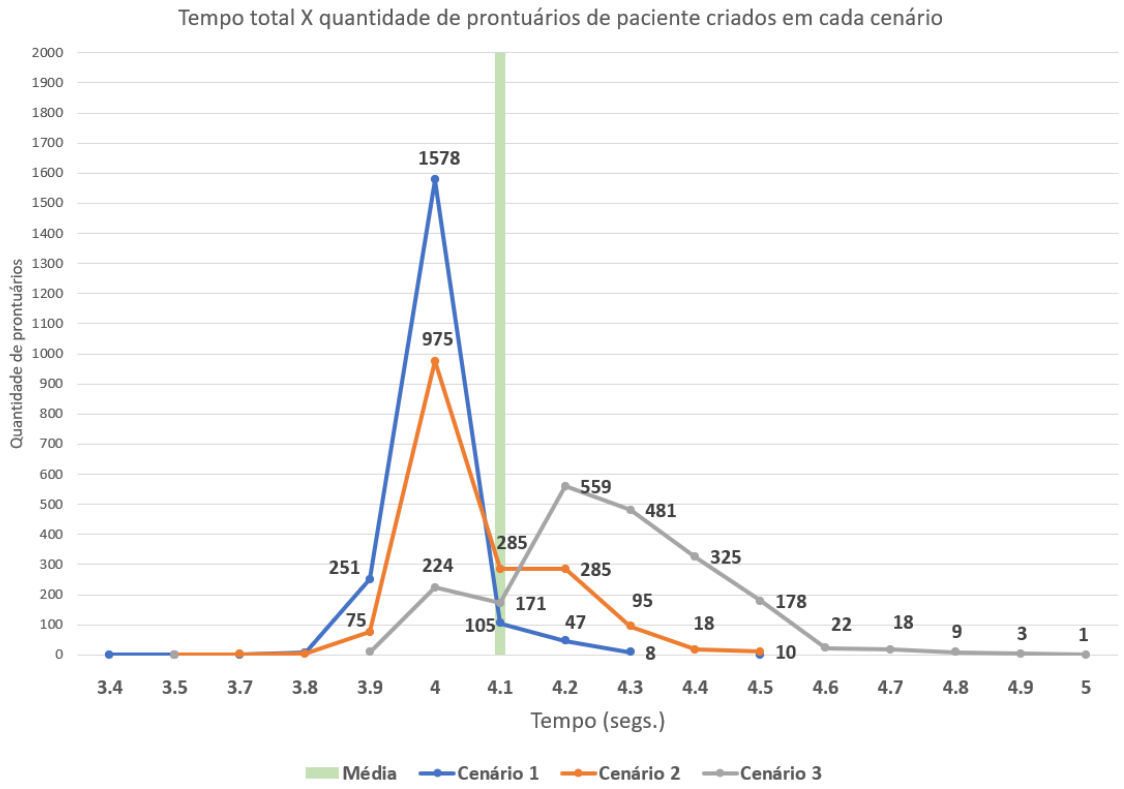
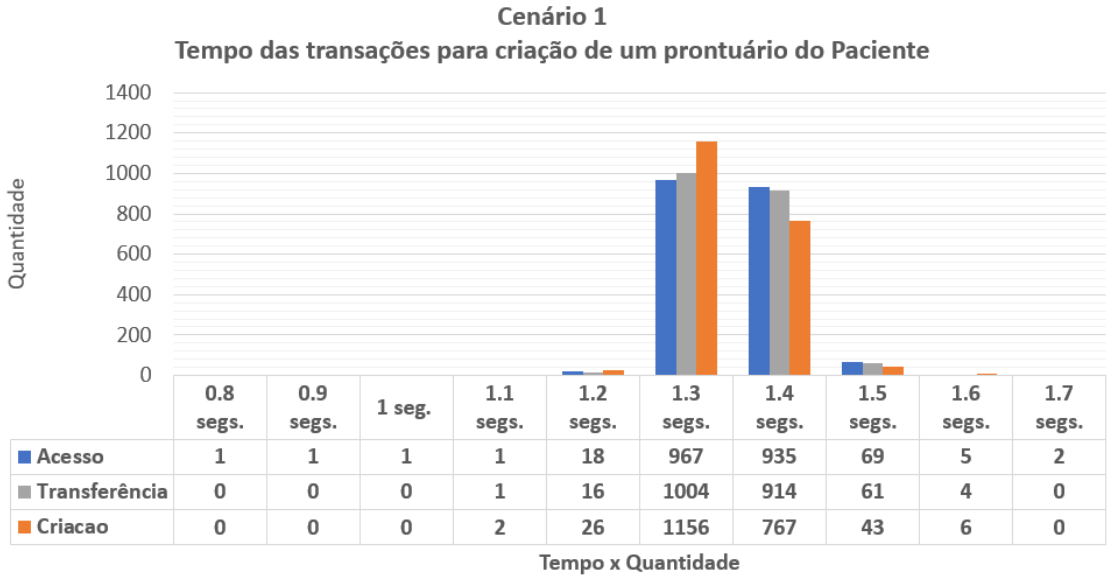


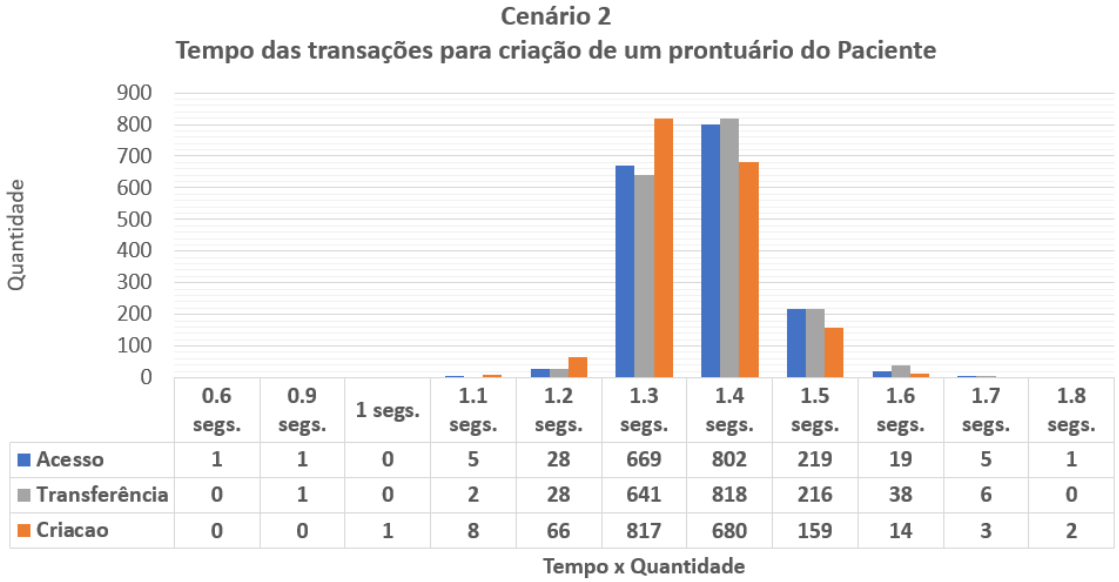
Figura 4.12: Tempo total X quantidade de prontuários de pacientes criados

O processo de criação de um prontuário é composto de três transações. A primeira transação é a criação do ativo prontuário. Em seguida, ocorre a transferência do ativo para o paciente e a última transação é a criação do ativo de acesso. As Figuras 4.13, 4.14 e 4.15 apresentam os tempos de cada transação em cada um dos cenários. Além das transações que são submetidas na blockchain o processo executa a criptografia do prontuário que não foi considerado na tomada dos tempos.



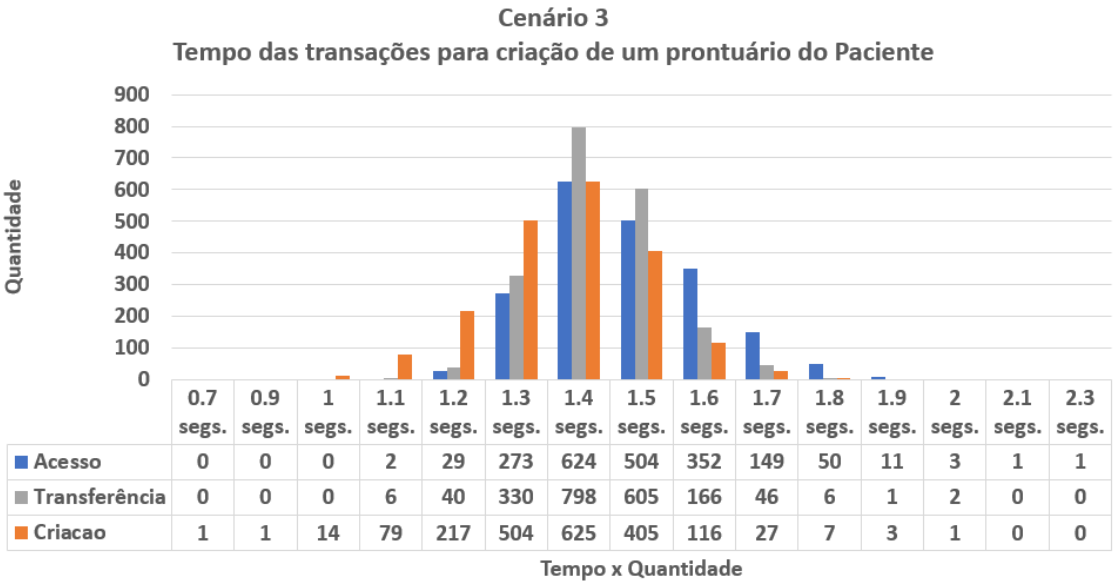
	0.8 segs.	0.9 segs.	1 seg.	1.1 segs.	1.2 segs.	1.3 segs.	1.4 segs.	1.5 segs.	1.6 segs.	1.7 segs.
Acesso	1	1	1	1	18	967	935	69	5	2
Transferência	0	0	0	1	16	1004	914	61	4	0
Criacao	0	0	0	2	26	1156	767	43	6	0

Figura 4.13: Tempo das transações X quantidade de prontuários no cenário 1



	0.6 segs.	0.9 segs.	1 segs.	1.1 segs.	1.2 segs.	1.3 segs.	1.4 segs.	1.5 segs.	1.6 segs.	1.7 segs.	1.8 segs.
Acesso	1	1	0	5	28	669	802	219	19	5	1
Transferência	0	1	0	2	28	641	818	216	38	6	0
Criacao	0	0	1	8	66	817	680	159	14	3	2

Figura 4.14: Tempo das transações X quantidade de prontuários no cenário 2



	0.7 segs.	0.9 segs.	1 segs.	1.1 segs.	1.2 segs.	1.3 segs.	1.4 segs.	1.5 segs.	1.6 segs.	1.7 segs.	1.8 segs.	1.9 segs.	2 segs.	2.1 segs.	2.3 segs.
Acesso	0	0	0	2	29	273	624	504	352	149	50	11	3	1	1
Transferência	0	0	0	6	40	330	798	605	166	46	6	1	2	0	0
Criacao	1	1	14	79	217	504	625	405	116	27	7	3	1	0	0

Figura 4.15: Tempo das transações X quantidade de prontuários no cenário 3

Para o cenário de consulta dos prontuários, o programa percorre todos os ativos do tipo prontuário que estão na blockchain. Para cada prontuário é encontrado o ativo de acesso correspondente. Com o ativo de acesso e a chave privada do paciente é efetuado a descriptografia do prontuário. Para cada prontuário o processo demorou em média 0,5 segundos e concluindo a consulta total em quarenta e oito minutos.

Outro teste efetuado que difere dos cenários criados é o tamanho máximo do prontuário. Para tal foi criado um programa que vai incrementando a quantidade de caracteres dentro do prontuário até que ocorra um erro no processo. O processo acusou erro quando o tamanho do prontuário chegou a 183.800 caracteres de conteúdo. O erro reportado é uma limitação da solução para o tamanho do arquivo JSON enviado para a blockchain.

4.4 Cenários de simulação

A simulação teve como objetivo avaliar o desempenho da rede para validar a tarefa de criação e validação de prontuários usando quatro configurações que variam de tamanho de bloco de dados, latência de rede e jitter (entrega de pacote).

A configuração da simulação consistiu em cinco máquinas virtuais (8 núcleos e 24 GB de RAM) rodando em um servidor Huawei FusionServer RH2288H V3 com 2 processadores Intel Xeon E5-2690 v3, 128 gb DDR4 RAM e 2x disco rígido de 2 TB. As máquinas virtuais foram aplicadas para simular cinco universidades localizadas em cinco diferentes regiões brasileiras (norte, nordeste, centro-oeste, sudeste, sul). Para a simulação, a máquina 1 ou nó 1 (norte), foi a responsável pela criação dos prontuários e também por consenso. As demais máquinas, ou nós, realizaram o consenso exclusivamente. Foi criado um script para disparar 32 execuções do processo de criação de prontuários para rodar em paralelo. Em cada execução foram anotados os tempos de início e fim para posterior avaliação. A Figura 4.16 descreve o design da arquitetura de rede para as simulações.

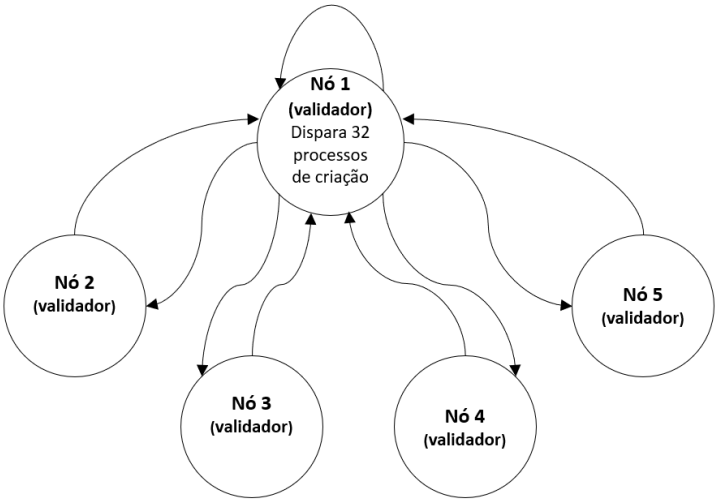


Figura 4.16: Arquitetura de rede para as simulações

Foi utilizada a ferramenta NetEm[10] para emular as adversidades da rede nessas regiões. Baseamos os parâmetros de simulação no relatório do Centro de Estudos e Pesquisas em Tecnologia de Redes e Operações (2021). A Tabela 4.1 descreve os cenários que combinam as adversidades principais relacionadas à latência e ao jitter das redes de dados. Esses parâmetros dificilmente aparecem isolados no ambiente real. Para cada simulação, criamos 20.000 prontuários e variamos os tamanhos dos pacotes com os valores 15, 20, 40 e 80 Kilobytes. Testamos outros tamanhos de prontuários, no entanto, foram necessários ajustes nas configurações padrões do Tendermint.

Tabela 4.1: Simulação: cenários variando latência e jitter

	Nó 1 (Norte)	**Nó 2 (Nordeste)**	**Nó 3 (Centro Oeste)**	**Nó 4 (Sudeste)**	**Nó 5 (Sul)**
Cenário 1					
Nó 1 (Norte)	Latência (0ms) Jitter (0ms)	Latência (0ms) Jitter (0ms)	Latência (0ms) Jitter (0ms)	Latência (0ms) Jitter (0ms)	Latência (0ms) Jitter (0ms)
Cenário 2					
Nó 1 (Norte)	Latência (50ms) Jitter (6ms)	Latência (55ms) Jitter (6ms)	Latência (65ms) Jitter (7ms)	Latência (60ms) Jitter (6ms)	Latência (77ms) Jitter (8ms)
Cenário 3					
Nó 1 (Norte)	Latência (50ms) Jitter (4ms)	Latência (55ms) Jitter (4ms)	Latência (65ms) Jitter (5ms)	Latência (60ms) Jitter (4ms)	Latência (77ms) Jitter (6ms)
Cenário 4					
Nó 1 (Norte)	Latência (50ms) Jitter (2ms)	Latência (55ms) Jitter (2ms)	Latência (65ms) Jitter (4ms)	Latência (60ms) Jitter (2ms)	Latência (77ms) Jitter (5ms)

4.4.1 Resultados da simulação

A Figura 4.17 mostra que a arquitetura de software desenvolvida suportou um alto volume de transações (60.000 transações para criação de 20.000 prontuários) considerando diferentes configurações. A combinação da latência da rede e do

[10]Emuladores de rede são ferramentas importantes para fazer pesquisas e desenvolvimento relacionado a protocolos e aplicativos de rede. Com a emulação de rede é possível realizar testes de cenários de rede realistas de maneira controlada, que não é possível usando apenas dispositivos de rede reais sem recursos de emulação. Um dos emuladores de redes mais populares é NetEm (JURGELIONIS et al., 2011).

tamanho do prontuário influenciou o tempo de consenso. Mas, avaliando apenas a latência de rede e o jitter, não houve influência, já que o cenário 2 seria o pior e não foi o que aconteceu. A configuração 3 foi o pior cenário, com uma média de 41 prontuários por minuto. A configuração 1 foi a melhor, com uma média de 47 prontuários por minuto. O desempenho da rede Blockchain depende também do mecanismo de consenso escolhido. Escolhemos o algoritmo BFT, então em nossos cenários, os quatro nós mais rápidos são suficientes para chegar ao consenso. Na média geral dos cenários, o modelo demorou 0,38 segundos para criar um prontuário.

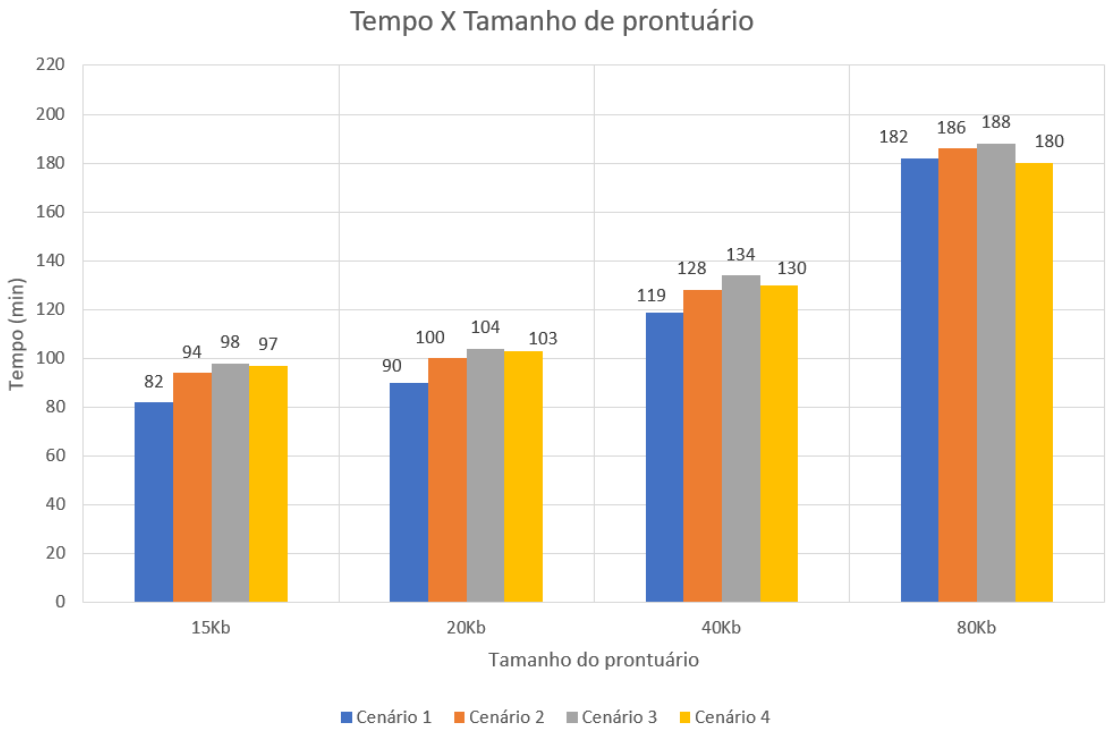


Figura 4.17: Simulação: Tempo X Tamanho de prontuário

5 CONCLUSÕES

A blockchain possibilita o compartilhamento das informações dos prontuários entre profissionais da saúde e empossa o paciente das informações desse documento. Também os pacientes podem controlar os acessos aos seus dados e com isso, o controle passa a ser de propriedade do paciente. Este trabalho apresenta a utilização de uma alternativa para armazenar prontuários médicos em um cenário de uso da blockchain. Apesar de algumas soluções já terem demonstrado serem promissoras, na sua maioria utilizam blockchains que são suportadas por criptomoedas ou necessitam de pagamentos de taxas. Os diferenciais do modelo proposto são: (i) utilização de um banco de dados orientado a documentos e (ii) um mecanismo de consenso que não necessita de mineração para validação. Além disso, foi abordada uma ideia de utilizar um prontuário estruturado e também sugere a criptografia dos dados mesmo que a solução esteja em um blockchain permissionada. Assim, os dados serão visualizados somente pelos participantes que os proprietários permitirem.

5.1 Contribuições

Com a criação dos tipos de ativos específicos para contemplar a solução proposta e o protótipo desenvolvido neste trabalho, foi possível confirmar as características da blockchain, como privacidade (par de chaves para identificação e criptografia das informação), descentralização e propriedade das informações. Além disso, nos testes, foi possível verificar o algoritmo de consenso funcionando em cenários de falhas dos validadores. O algoritmo baseado em votos mostrasse eficaz, visto que os validadores são conhecidos entre eles e esses tenham somente interesse nas informações geradas e não em outros tipos de recompensas.

A estrutura SOAP utilizada na confecção dos prontuários dos pacientes neste trabalho, fora motivada a ser a escolhida por existir uma estratégia do Ministério da Saúde Brasileira que a utiliza. Essa estratégia faz referência ao processo de informatização qualificada do Sistema Único de Saúde com objetivo concretizar um novo modelo de gestão de informação que apoie os serviços de saúde na gestão efetiva do cuidado dos pacientes.

As simulações se aproximaram de um cenário de utilização do mundo real. Foram utilizados dados informados pelo Centro de Estudos e Pesquisas em Tecnologia de Redes e Operações em tempo de pandemia do COVID-19. Perante a esses dados os tempos de validação dos prontuários na blockchain satisfatórios tanto para criação quanto para a consulta das informações.

5.2 Trabalhos Futuros

Durante os testes e simulações foram utilizadas consultas básicas para validar as informações persistidas, por exemplo, validar a quantidade de ativos do tipo prontuário gravados no fim de cada teste. E como discutido no trabalho, o armazenamento das informações geradas dentro de um solução baseada na blockcahin deve ser um ponto a ser avaliado com atenção. Visto que os bancos de dados estão em cada uma das instituições que efetuam o consenso é possível criar estratégias de indexação e extração das informações geradas pela solução proposta. Por exemplo, avaliar mecanismos de aprendizado de máquinas para para realizar estudos sobre patologias existentes ou identificação de novas.

Por fim, com avanço constante das tecnologias, outras sugestões para pesquisas futuras podem ser direcionadas para avaliação de utilização de algoritmos de criptografia para identificação dos participantes e também do prontuário do paciente.

REFERÊNCIAS

ACADEMY, B. **Casos de Uso Blockchain**: Setor de Saúde. Acesso em 07 de novembro de 2020, https://academy.binance.com/pt/articles/blockchain-use-cases-healthcare.

AGBO, C. C.; MAHMOUD, Q. H. Comparison of blockchain frameworks for healthcare applications. **Internet Technology Letters**, Ontario, Canada, v.2, n.5, p.e122, 2019.

AGBO, C. C.; MAHMOUD, Q. H.; EKLUND, J. M. Blockchain Technology in Healthcare: A Systematic Review. **Healthcare**, Ontario, Canada, v.7, n.2, p.56, June 2019. Number: 2 Publisher: Multidisciplinary Digital Publishing Institute.

ALIAGA, Y. E. M.; HENRIQUES, M. A. A. Uma comparação de mecanismos de consenso em blockchains. **X Encontro de Alunos e Docentes do DCA/FEEC/UNICAMP (EADCA)**, Campinas, Brasil, p.4, 2017.

ALONSO, S. G. et al. Proposing New Blockchain Challenges in eHealth. **Journal of Medical Systems**, Sydney, Australia, v.43, n.3, p.64, Feb. 2019.

ALSUNAIDI, S. J.; ALHAIDARI, F. A. **A Survey of Consensus Algorithms for Blockchain Technology**. Sakaka, Saudi Arabia: IEEE, 2019. 1–6p.

AZARIA, A. et al. **MedRec**: Using Blockchain for Medical Data Access and Permission Management. Vienna, Austria: IEEE, 2016. 25–30p.

BASHIR, I. **Mastering Blockchain**: Distributed ledger technology, decentralization, and smart contracts explained, 2nd Edition. Birmingham, UK: Packt Publishing, 2018. OCLC: 1032154406.

BELLOD CISNEROS, J. L.; AARESTRUP, F. M.; LUND, O. Public Health Surveillance using Decentralized Technologies. **Blockchain in Healthcare Today**, Denmark, v.1, 2018.

BENDER, D.; SARTIPI, K. **HL7 FHIR**: An Agile and RESTful approach to healthcare information exchange. Porto, Portugal: IEEE, 2013. 326–331p.

BIGCHAINDB. **Set Up BigchainDB, MongoDB and Tendermint**. Acesso 09 de junho de 2020, http://docs.bigchaindb.com/projects/server/en/latest/simple-deployment-template/set-up-node-software.html.

BRASIL. Resolução CFM n° 1.638/2002. **Diário Oficial da União**, Brasília, DF, 09 agosto 2002. Seção 1, p.184-5.

BRASIL. e-SUS Atenção Básica : manual do sistema com prontuário eletrônico do cidadão pec – versão 3.2. **Ministério da Saúde. Secretaria de Atenção Primária à Saúde**, Brasília, DF, 2019.

BURKE, T. **Business Analysis in the Blockchain Age**. Acesso em 09 de junho de 2020, https://go.iiba.org/whitepaper/blockchain.

Centro de Estudos e Pesquisas em Tecnologia de Redes e Operações. **COVID-19 Impactos na qualidade da internet no Brasil**. Acesso em 11 de fevereiro de 2021, https://ceptro.br/assets/publicacoes/pdf/2021.01.05-relatorio-covid.pdf.

CFM, C. F. d. M.; SBIS, S. B. d. I. e. S. **A Certificação de Sistemas de Registro Eletrônico de Saúde**. 2012.

CHODOROW, K.; DIROLF, M. **MongoDB**: The Definitive Guide. 1st.ed. Springfield, Missouri: O'Reilly Media, Inc., 2010.

CHOWDHURY, M. J. M. et al. Blockchain Versus Database: A Critical Analysis. **2018 17th IEEE International Conference On Trust, Security And Privacy In Computing And Communications/ 12th IEEE International**

Conference On Big Data Science And Engineering (TrustCom/BigDataSE), New York, NY, USA, p.1348–1353, Aug. 2018. ISSN: 2324-9013.

CORON, J.-S. What is cryptography? **IEEE Security & Privacy Magazine**, Luxembourg, Luxembourg, v.4, n.1, p.70–73, Jan. 2006.

COSTA, C. G. A. d. **Desenvolvimento e avaliação tecnologica de um sistema de prontuario eletronico do paciente, baseado nos paradigmas da World Wide Web e da engenharia de software**. 2001. Tese (Doutorado em Ciência da Computação) — .

CYRAN, M. A. Blockchain as a Foundation for Sharing Healthcare Data. **Blockchain in Healthcare Today**, Stamford, Connecticut, Mar. 2018.

DAGHER, G. G. et al. Ancile: Privacy-preserving framework for access control and interoperability of electronic health records using blockchain technology. **Sustainable Cities and Society**, Amsterdam, Netherlands, v.39, p.283–297, May 2018.

DHAMEJA, G. **Lifecycle of a BigchainDB Transaction**. Acesso em 06 de marco de 2020, https://blog.bigchaindb.com/lifecycle-of-a-bigchaindb-transaction-c1e34331cbaa.

FRADE, S. et al. **Survey of openEHR storage implementations**. Porto, Portugal: IEEE, 2013. 303–307p.

GMBH, B. **BigchainDB 2.0 The Blockchain Database**. Berlim, Germany, 2018.

HöLBL, M. et al. A Systematic Review of the Use of Blockchain in Healthcare. **Symmetry**, Basel, Switzerland, v.10, n.10, p.470, Oct. 2018. Number: 10 Publisher: Multidisciplinary Digital Publishing Institute.

HYPPöNEN, H. et al. Impacts of structuring the electronic health record: A systematic review protocol and results of previous reviews. **International Jour-**

nal of Medical Informatics, Amsterdam, Netherlands, v.83, n.3, p.159–169, Mar. 2014.

JAROUDI, S.; PAYNE, J. D. Remembering Lawrence Weed: A Pioneer of the SOAP Note. **Academic Medicine**, Washington, D.C., v.94, n.1, p.11, jan 2019.

JOSE, B.; ABRAHAM, S. Exploring the merits of nosql: A study based on mongodb. **2017 International Conference on Networks Advances in Computational Technologies (NetACT)**, Thiruvanthapuram, India, p.266–271, July 2017.

JURGELIONIS, A. et al. An Empirical Study of NetEm Network Emulation Functionalities. **2011 Proceedings of 20th International Conference on Computer Communications and Networks (ICCCN)**, Lahaina, HI, USA, p.1–6, July 2011.

KARAFILOSKI, E.; MISHEV, A. Blockchain solutions for big data challenges: A literature review. **IEEE EUROCON 2017 -17th International Conference on Smart Technologies**, Ohrid, Macedonia, p.763–768, July 2017.

KUO, T.-T.; ZAVALETA ROJAS, H.; OHNO-MACHADO, L. Comparison of blockchain platforms: a systematic review and healthcare examples. **Journal of the American Medical Informatics Association**, Oxford, England, v.26, n.5, p.462–478, May 2019.

KUO, T.-T.; ZAVALETA ROJAS, H.; OHNO-MACHADO, L. Comparison of blockchain platforms: a systematic review and healthcare examples. **Journal of the American Medical Informatics Association**, [S.l.], v.26, n.5, p.462–478, May 2019.

KWON, J. **Tendermint**: consensus without mining. 2014.

LI, P. et al. DMMS: A Decentralized Blockchain Ledger for the Management of Medication Histories. **Blockchain in Healthcare Today**, Stamford, Connecticut, v.2, Jan. 2019.

LIM, Y. H. et al. Blockchain Technologies in E-commerce: Social Shopping and Loyalty Program Applications. **Social Computing and Social Media. Communication and Social Communities**, Cham, p.403–416, 2019.

LINN, L. A.; KOO, M. B. Blockchain For Health Data and Its Potential Use in Health IT and Health Care Related Research. **ONC**, https://www.healthit.gov/sites/default/files/11-74-ablockchainforhealthcare.pdf, 2016.

MEYERS, K. C.; MILLER, H. J.; NAEYMI-RAD, F. Problem focused knowledge navigation: implementing the problem focused medical record and the O-HEAP note. **Proceedings of the AMIA Symposium**, Orlando, Florida, p.325–329, 1998.

MITRA, R. **What is Tendermint Core? The Most Comprehensive Guide Ever**. Acesso em 09 de junho de 2020, https://blockgeeks.com/guides/tendermint/.

MONGODB. **Building Enterprise-Grade Blockchain Databases with MongoDB**. 2017.

MONGODB. **What is NoSQL?** Acesso em 20 de novembro de 2020, https://www.mongodb.com/nosql-explained.

NAKAMOTO, S. **Bitcoin**: a peer-to-peer electronic cash system. 2009.

PARANJAPE, K. et al. Implementation Considerations for Blockchain in Healthcare Institutions. **Blockchain in Healthcare Today**, Stamford, Connecticut, v.2, Jul. 2019.

QUAINI, T.; ROEHRS, A.; RIGHI, C. A. d. C. a. R. d. R. UNIREC: AN ARCHITECTURE PROPOSAL FOR INTEGRATING DISTRIBUTED ELECTRONIC HEALTH RECORDS USING BLOCKCHAIN. **International Conferences WWW/Internet 2018 and Applied Computing 2018**, Budapest, Hungary, p.167–174, 2018.

RAMESH, S. **Introducing MedBlocks - Storing Medical Records Securely on the Interplanetary File System using Blockchain technology**. Acesso em 11 de junho de 2020, https://medium.com/medblocks/introducing-medblocks-storing-medical-records-securely-on-the-interplanetary-file-system-using-20f4e88c9bda.

ROUHANI, S. et al. MediChainTM: a secure decentralized medical data asset management system. **2018 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData)**, Halifax, Canada, Jul 2018.

ROUSE, M. **RSA algorithm (Rivest-Shamir-Adleman)**. Acesso em 10 de junho de 2020, https://searchsecurity.techtarget.com/definition/RSA.

ROUSE, M. **Advanced Encryption Standard (AES)**. Acesso em 10 de junho de 2020, https://searchsecurity.techtarget.com/definition/AdvancedEncryptionStandard.

SAHOO, M. S.; BARUAH, P. K. HBasechainDB – A Scalable Blockchain Framework on Hadoop Ecosystem. **Supercomputing Frontiers**, Cham, p.18–29, 2018.

SARAF, C.; SABADRA, S. Blockchain platforms: A compendium. **2018 IEEE International Conference on Innovative Research and Development (ICIRD)**, Bangkok, Thailand, p.1–6, May 2018.

SOUZA, R. A.; BORGES, F.; CUNHA, G. N. da. O algoritmo AES: apresentação e descrição da estrutura. **Resumos do I Encontro Acadêmico de Modelagem Computacional do Laboratório Nacional de Computação Científica**, Petrópolis - RJ, 2008.

SWAN, M. **Blockchain**: blueprint for a new economy. 1st.ed. [S.l.]: O'Reilly Media, Inc., 2015.

TANDON, A. et al. Blockchain in healthcare: A systematic literature review, synthesizing framework and future research agenda. **Computers in Industry**, Amsterdam, Netherlands, v.122, p.103290, Nov. 2020.

TANENBAUM, A. S.; STEEN, M. van. **Distributed Systems**: principles and paradigms. 2th.ed. USA: Prentice Hall International, 2008.

TANENBAUM, A. S.; WETHERALL, D. J. **Computer Networks**. 5th.ed. USA: Prentice Hall Press, 2010.

TASATANATTAKOOL, P.; TECHAPANUPREEDA, C. Blockchain: Challenges and applications. **2018 International Conference on Information Networking (ICOIN)**, Chiang Mai, Thailand, p.473–475, Jan. 2018.

WüST, K.; GERVAIS, A. **Do you Need a Blockchain?** Zug, Switzerland, 2018. 45–54p.

ZHEALTH. **Zhealth Soluções Blockchain para o setor da saúde**. Acessado: 2020-03-16, http://zhealth.com.br/.