



UNIVERSIDADE FEDERAL DO RIO GRANDE DO NORTE CENTRO DE
TECNOLOGIA
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA E DE
COMPUTAÇÃO



Blockchain no Ecossistema Tecnológico da Plataforma revELA: no contexto da esclerose lateral amiotrófica (ELA)

Aleika Lwiza Alves Fonsêca

Orientador: Prof. Dr. Ricardo Alexsandro de Medeiros Valentim

Dissertação de Mestrado apresentada ao
Programa de Pós-Graduação em Engenharia
Elétrica e de Computação da UFRN (área de
concentração: Engenharia de Computação)
como parte dos requisitos para obtenção do
título de Mestrado em Ciências

Nº de Ordem PPgEEC:

Natal/RN, janeiro de 2025

Universidade Federal do Rio Grande do Norte - UFRN
Sistema de Bibliotecas - SISBI
Catalogação de Publicação na Fonte. UFRN - Biblioteca Central Zila Mamede

Fonseca, Aleika Lwiza Alves.

Blockchain no ecossistema tecnológico da Plataforma revELA:
no contexto da esclerose lateral amiotrófica (ELA) / Aleika
Lwiza Alves Fonseca. - 2025.

74 f.: il.

Dissertação (mestrado) - Universidade Federal do Rio Grande
do Norte, Centro de Tecnologia, Programa de Pós-Graduação em
Engenharia Elétrica e de Computação, Natal, RN, 2025.

Orientação: Prof. Dr. Ricardo Alexsandro de Medeiros
Valentim.

1. Registro Eletrônico em Saúde - Dissertação. 2. Esclerose
Lateral Amiotrófica - Dissertação. 3. Registro Nacional da ELA -
Dissertação. 4. Blockchain - Dissertação. 5. Hyperledger Fabric
- Dissertação. I. Valentim, Ricardo Alexsandro de Medeiros. II.
Título.

RN/UF/BCZM

CDU 004:61(043.3)

*Ao meu noivo,
Luiz Gustavo Maia de Lima.*

Agradecimentos

Agradeço aos meus pais, Nilton Fonsêca e Avani Alves, por terem me concedido toda a base e apoio que precisava para chegar até aqui hoje. Pela criação que me deram e me ajudaram a me tornar a pessoa que sou hoje. Sempre me mostraram que a educação era o caminho para uma vida melhor, isso sempre me impulsionou a estudar cada vez mais.

Ao meu futuro esposo Luiz Gustavo, por ser meu melhor amigo, maior incentivador e por não medir esforços para fazer de tudo por nós. Agradeço por toda a paciência que ele teve nos momentos em que estive ansiosa, estressada ou triste achando que nada ia dar certo e em todas essas vezes ele me acalmou e mostrou que eu ia conseguir, mostrando que ele acredita mais em mim do que eu mesma.

Ao meu orientador Dr. Ricardo Valentim, por ser uma referência quanto ao conhecimento e experiências de vida. Por servir de inspiração como uma pessoa que sempre pensa além e que traz ótimas discussões. Cada encontro ou oportunidade de escutá-lo é uma aula cheia de ensinamentos e coisas novas.

Agradeço à todos com quem tive a oportunidade de trabalhar dentro do Laboratório de Inovação Tecnológica em Saúde (LAIS), foi com a ajuda dessas pessoas e com o apoio financeiro vindo dos projetos do laboratório que desenvolvemos tantas soluções digitais na área da saúde, em destaque para o Registro Nacional da ELA que foi a base para este trabalho, e pude publicar o meu primeiro artigo em revista internacional de alto impacto (coisa que eu nunca imaginei).

Não poderia deixar de agradecer especialmente a Ingridy Barbalho por sempre ter feito mais por mim do que realmente precisava fazer. Por ter tido muita paciência nas vezes que não

estava rendendo bem em minhas entregas de tarefas, por ter nossos momentos de desabafo nas reuniões e, enfim, por me acompanhar e apoiar desde o desenvolvimento dos projetos, entrega do meu trabalho de TCC, publicação de artigos e agora no meu mestrado.

Resumo

O Registro Nacional da ELA é uma solução de saúde digital que tem como objetivo coletar os dados epidemiológicos dos diversos pacientes com Esclerose Lateral Amiotrófica (ELA) no Brasil. Diante do uso de soluções tecnológicas como essa, de registro eletrônico de dados em saúde, surgem as preocupações relacionadas à segurança, integridade dos dados e importância de uma plataforma interoperável para o compartilhamento com outros sistemas. Para sanar tais preocupações, a tecnologia *blockchain* vem atraindo bastante atenção em diversas áreas que já enxergam as possibilidades de progresso com a sua incorporação. Tendo em vista as carências por segurança, auditabilidade, interoperabilidade e para tornar o Registro Nacional da ELA cada vez mais seguro e confiável, este trabalho apresenta a configuração de uma rede *blockchain* com o *framework Hyperledger Fabric* e a implementação de uma API, desenvolvida com Node.js, usada para integrar a *blockchain* ao Registro Nacional da ELA. Para avaliar o desempenho da solução proposta foi utilizada a ferramenta *Hyperledger Caliper*, esta que é referência para benchmark de soluções *blockchain*. Os resultados mostram a viabilidade do uso dessa tecnologia e a relevância do seu uso é destacada no decorrer do texto. Apesar de a análise de performance não ter sido satisfatória, com latência média acima de 19s e com valores de *throughput* abaixo de 100 transações por segundo, os dados obtidos permitem identificar pontos de melhoria na implementação e otimização da rede.

Palavras-chave: Registro Eletrônico em Saúde. Esclerose Lateral Amiotrófica. Registro Nacional da ELA. *Blockchain*. Hyperledger Fabric.

Abstract

The National ALS Registry is a digital health solution that aims to collect epidemiological data from various patients with Amyotrophic Lateral Sclerosis (ALS) in Brazil. When using technological solutions such as this one, for electronic health data recording, concerns arise regarding security, data integrity, and the importance of an interoperable platform for sharing with other systems. To address these concerns, blockchain technology has been attracting a lot of attention in several areas that already see the possibilities for progress with its incorporation. Considering the needs for security, auditability, and interoperability, and to make the National ALS Registry increasingly secure and reliable, this paper presents the configuration of a blockchain network with the Hyperledger Fabric framework and the implementation of an API, developed with Node.js, used to integrate the blockchain with the National ALS Registry. To evaluate the performance of the proposed solution, the Hyperledger Caliper tool was used, which is a reference for benchmarking blockchain solutions. The results show the viability of using this technology and the relevance of its use is highlighted throughout the text. Although the performance analysis was not satisfactory, with average latency above 19s and throughput values below 100 transactions per second, the data obtained allows us to identify points for improvement in the implementation and optimization of the network.

Keywords: Electronic Health Record. Amyotrophic Lateral Sclerosis. National ALS Registry. Blockchain. Hyperledger Fabric

Sumário

Lista de Figuras	iv
Lista de Tabelas	v
Lista de Símbolos e Abreviaturas.....	vi
Produção Acadêmica.....	vii
Capítulo 1 Introdução	8
1.1 Problemática	10
1.2 Objetivos.....	12
1.2.1 Objetivo Geral.....	12
1.2.2 Objetivos Específicos	12
1.3 Estrutura da dissertação	13
Capítulo 2 Referencial Teórico.....	14
2.1 Esclerose Lateral Amiotrófica (ELA)	14
2.2 Projeto RevELA.....	15
2.2.1 Registro Nacional da ELA	16
2.3 Blockchain	18
2.3.1 Principais conceitos	20
2.3.1.1 Blocos	20
2.3.1.2 Hash	20
2.3.1.3 Transação.....	21
2.3.1.4 Consenso.....	21
2.3.1.5 Contratos Inteligentes (<i>smart contracts</i>)	21
2.4 Hyperledger Fabric Framework	22
2.4.1 Pares, Organizações e Ordenadores	22
2.4.2 Identidade, <i>Wallet</i> e <i>Gateway</i>	23
2.4.3 Contratos inteligentes (<i>chaincode</i>)	23
2.4.4 Provedor de Serviços de Associação	23
2.4.5 Usuários da rede	24
2.5 Análise de performance.....	24

2.5.1 Taxa de envio	25
2.5.2 Latência	25
2.5.2.1 Latência máxima.....	26
2.5.2.2 Latência mínima	26
2.5.2.3 Latência média.....	26
2.5.3 Taxa de transferência de transações.....	26
2.6 Blockchain em Sistemas de Informação em Saúde.....	27
2.6.1 Resultados e Discussões.....	28
Capítulo 3 Materiais e Métodos	30
3.1 Hyperledger Fabric Network	30
3.2 Hyperledger Caliper.....	33
Capítulo 4 Implementação e Resultados	35
4.1 Contratos inteligentes (<i>smart contract</i>)	35
4.1.1 RegisterPatient.....	36
4.1.2 RegisterAccessPatientData	36
4.1.3 UpdatePatientData	37
4.1.4 GetPatient.....	39
4.1.5 GetPatientHistory.....	39
4.1.6 GetAllPatients	40
4.2 API para comunicação entre os sistemas e a rede blockchain	41
4.2.1 Estabelecer uma conexão segura com a rede blockchain Hyperledger Fabric.....	42
4.2.2 Validar se a requisição recebida pela API possui um <i>token</i> válido.....	43
4.2.3 Definir as rotas que executarão os métodos definidos pelos contratos inteligentes	43
4.3 Utilização da API pelo Registro Nacional da ELA.....	43
4.3.1 Classe <i>BlockchainIntegration</i>	43
4.3.2 Classe <i>JwtHelper</i>	46
4.4 Interagindo com a rede <i>blockchain</i>	47
4.4.1 Retornando todos os pacientes cadastrados na rede	47
4.4.2 Retornando dados do paciente pelo CPF	49
4.4.3 Retornando histórico de alterações do paciente pelo CPF	51
4.5 Análise de performance.....	52
4.5.1 Operações de escrita na rede	54
4.5.2 Operações de leitura na rede.....	55
Capítulo 5 Discussões	57

Capítulo 6 Considerações Finais	59
6.1 Principais Contribuições e Impactos sociais	60
6.2 Trajetória acadêmica	61
6.3 Limitações e Desafios.....	62
6.4 Trabalhos Futuros	63
Referências	64

Lista de Figuras

Figura 2.1: Infográfico com os produtos do projeto RevELA	16
Figura 2.2: Fluxo de homologação dos dados inseridos pelo paciente no autorrelato	17
Figura 2.3: Processo de transação em uma rede blockchain	20
Figura 3.1: Etapas para a configuração da rede blockchain	32
Figura 4.1: Fluxo de integração do Registro Nacional da ELA com a blockchain	35
Figura 4.2: Detalhes de implementação da função RegisterPatient	36
Figura 4.3: Detalhes de implementação da função RegisterAccessPatientData	37
Figura 4.4: Detalhes de implementação da função UpdatePatientData	38
Figura 4.5: Detalhes de implementação da função GetPatient	39
Figura 4.6: Detalhes de implementação da função GetPatientHistory	40
Figura 4.7: Detalhes de implementação da função GetAllPatients	41
Figura 4.8: Métodos principais da classe BlockchainIntegration	44
Figura 4.9: Métodos auxiliares da classe BlockchainIntegration	45
Figura 4.10: Método createJWTToken da classe JwtHelper	46
Figura 4.11: Retorno dos dados dos pacientes cadastrados na rede	48
Figura 4.12: Mais detalhes do retorno dos dados do registro do paciente	49
Figura 4.13: Retorno de um paciente filtrado pelo CPF	50
Figura 4.14: Retorno do histórico de alterações de um paciente pelo CPF	51
Figura 4.15: Configuração dos testes utilizando como exemplo o da função <i>registerPatient</i>	52
Figura 4.16: Configuração dos testes para a função <i>getPatientHistory</i>	53

Lista de Tabelas

Tabela 4.1: Sumário das métricas para cada uma das funções avaliadas do contrato inteligente	54
---	----

Lista de Símbolos e Abreviaturas

API	<i>Application Programming Interface</i>
ELA	Esclerose Lateral Amiotrófica
HTTP	<i>Hypertext Transfer Protocol</i>
JWT	<i>JSON Web Tokens</i>
LGPD	Lei Geral de Proteção de Dados
MSP	<i>Membership Service Provider</i>
ODS	Objetivos de Desenvolvimento Sustentável
ONU	Organização das Nações Unidas
PEP ELA	Prontuário Eletrônico para Paciente com ELA
PRISMA	<i>Preferred Reporting Items for Systematic Reviews and Meta-Analyse</i>
PROSPERO	Registro Prospectivo Internacional de Revisões Sistemáticas
RES	Registro Eletrônico em Saúde
RN	Rio Grande do Norte
RNDS	Rede Nacional de Dados em Saúde
RSL	Revisão Sistemática de Literatura
SUS	Sistema Único de Saúde
TPS	Transações por segundo

Produção Acadêmica

Artigos, Livros e Capítulos de Livros

Título	Link	Tipo
Digital health solution for monitoring and surveillance of Amyotrophic Lateral Sclerosis in Brazil	https://doi.org/10.3389/fpubh.2023.1209633	Artigo
Blockchain in Health Information Systems: A Systematic Review	https://www.mdpi.com/1660-4601/21/11/1512	Artigo
Electronic health record for patients with amyotrophic lateral sclerosis: a Blockchain approach	https://zenodo.org/records/14337625	Anal

Registro de Software e Patentes

Título	Número Registro	Tipo	Data do Registro
Registro Nacional da ELA	BR512023002160-1	Registro de Software	Expedido em 2023
Observatório Nacional da ELA	BR512023002161-0	Registro de Software	Expedido em 2023
Autonomus	BR512023001526-1	Registro de Software	Expedido em 2023
Autonomus Mobile	BR512023001573-3	Registro de Software	Expedido em 2023

Capítulo 1

Introdução

A Esclerose Lateral Amiotrófica (ELA) é uma doença rara neurodegenerativa caracterizada pela morte progressiva de neurônios motores, que leva à paralisia fatal (Barbalho *et al.*, 2023; Ralli *et al.*, 2019). Os pacientes com ELA recebem cuidados paliativos e tratamentos que ajudam a aumentar a sobrevida do paciente em cerca de 2 a 3 meses, esses cuidados visam dispor uma melhor qualidade de vida, já que não foi encontrada, até hoje, uma cura para a doença (Mao; Zhang; Chen, 2015). Devido a sua heterogeneidade, o monitoramento da progressão da doença é complexo e requer idealmente uma abordagem multidisciplinar (Radunović; Mitsumoto; Leigh, 2007). Mesmo sendo amplamente estudada, profissionais de saúde enfrentam o desafio de proporcionar cuidados de suporte abrangentes e melhorar a qualidade de vida dos pacientes, dada a ausência de tratamentos definitivos, no qual os pilares do tratamento dos pacientes com ELA centram-se no controle dos sintomas (Radunović; Mitsumoto; Leigh, 2007; Ralli *et al.*, 2019).

Diante da necessidade de constantes estudos voltados à entender cada vez mais sobre a doença, os registros eletrônicos, ou prontuários eletrônicos, desempenham um papel fundamental no cuidado de pacientes com ELA (Granit *et al.*, 2022; Wei; Denny, 2015). Essas ferramentas permitem a coleta e o armazenamento eficaz de informações clínicas cruciais, incluindo históricos médicos, detalhes de tratamentos, medições de capacidade funcional e progressão da doença (Fernandes *et al.*, 2021). Esses registros oferecem grandes benefícios para os pacientes, fornecendo um histórico completo de seu estado de saúde e permitindo uma

melhor coordenação do cuidado multidisciplinar. No entanto, é importante destacar que, apesar dos benefícios, a adoção generalizada de prontuários eletrônicos na ELA ainda é escassa, existindo poucos ao redor do mundo (Barbalho; Fernandes, 2021; Barbalho, 2023).

Apesar da existência de soluções de saúde digital voltadas aos estudos sobre a ELA, elas apresentam alguns problemas relacionados à diversidade de dados e fontes, o que representa a falta de integração e interoperabilidade entre os sistemas. Cada sistema estabelece e aplica suas próprias iniciativas (estruturas lógicas e arquitetura tecnológica), muitas vezes desenvolvidas em infraestruturas centralizadas e alguns perdem a capacidade de integração com outras plataformas ou instituições (Barbalho *et al.*, 2022). É nesse contexto que surge o conceito da interoperabilidade, que significa a promoção de ambientes digitais integrados, cuja característica central se baseia no compartilhamento e troca de informações consistentes e precisas entre diferentes sistemas ou aplicações (Marcondes, 2016).

Para doenças que, assim como a ELA, precisam de um acompanhamento multidisciplinar e que necessitam de um longo histórico clínico dos pacientes muitas vezes fragmentados entre diferentes instituições de saúde, a utilização de um sistema interoperável e seguro reflete em ganhos consideráveis e simplifica bastante o acompanhamento da evolução da doença e histórico clínico dos pacientes.

Na busca por uma ferramenta para solucionar tais desafios, o uso da tecnologia *blockchain* pode desempenhar um papel fundamental (Shahnaz; Qamar; Khalid, 2019; Tanwar; Parekh; Evans, 2020). Sua arquitetura descentralizada permite que múltiplas organizações compartilhem informações de maneira uniforme, utilizando um livro-razão (*ledger*) comum, o que facilita a interoperabilidade entre sistemas distintos sem a necessidade de intermediários (Dagher *et al.*, 2018). Quanto à imutabilidade e auditabilidade dos dados registrados, a tecnologia garante um histórico confiável das transações, essencial para rastrear alterações e detectar inconsistências (Bari; Qamar; Khalid, 2021). Além disso, o uso da *blockchain* ajuda a

manter a integridade das informações, fornecendo transparência e fortalecendo a segurança no acesso aos dados (Antwi *et al.*, 2021; Mantey *et al.*, 2021).

Atualmente, o uso da blockchain no contexto da ELA, ainda é pouco explorado. Embora existam muitos estudos sobre o uso dessa tecnologia em sistemas de saúde (Fonseca *et al.*, 2024), os casos voltados especificamente para a ELA são raros. Soluções adaptadas às necessidades específicas da ELA, como o acompanhamento multidisciplinar e o compartilhamento de dados sensíveis entre profissionais de diferentes áreas, ainda requerem maior atenção e desenvolvimento. Isso ressalta a necessidade de mais pesquisas e desenvolvimentos voltados para adaptar a tecnologia blockchain para essa aplicação.

Neste contexto, essa dissertação de mestrado tem como objeto de pesquisa a utilização da tecnologia blockchain aplicada ao gerenciamento de dados dos pacientes com Esclerose Lateral Amiotrófica (ELA). O objetivo foi garantir a segurança, a auditabilidade e a imutabilidade em relação ao registro digital das informações dos pacientes com ELA, aspecto que visou dar mais integridade e confiança aos usuários do Registro Nacional da ELA, e servir como ferramenta base para os demais projetos integrados ao registro que utilizem estruturas semelhantes dentro da Plataforma revELA.

1.1 Problemática

No Brasil, os dados sobre a epidemiologia da Esclerose Lateral Amiotrófica (ELA) ainda são escassos. Alguns estudos apresentam a incidência e prevalência da doença, porém apenas a nível estadual ou municipal (Castro-Costa *et al.*, 1999; Linden Junior *et al.*, 2013; Loureiro *et al.* 2012). Na perspectiva de proporcionar avanços em pesquisas relacionadas a ELA, o RevELA, projeto de abrangência nacional desenvolvido pelo Laboratório de Inovação Tecnológica em Saúde (LAIS) da Universidade Federal do Rio Grande do Norte (UFRN), conta com um conjunto de iniciativas e desenvolvimento produtos (desenvolvimento de tecnologias,

ações e protocolos) voltados às lacunas existentes no cenário da ELA no Brasil. Dentre as iniciativas já desenvolvidas está o Registro Nacional da ELA¹ no Brasil. Esse registro consiste em uma solução de saúde digital que tem como objetivo coletar os dados epidemiológicos dos diversos pacientes com ELA no Brasil. A partir dos dados coletados pelo registro, é possível entender melhor como está a distribuição geográfica dos casos, avaliar os principais fatores demográficos e melhorar os cuidados com os indivíduos que possuem a doença (Barbalho *et al.*, 2023).

Apesar de ser uma solução já avançada em termos de desenvolvimento e implantação no ambiente real e ser integrada com as demais soluções do projeto RevELA, o Registro Nacional da ELA é uma solução de saúde digital que não é interoperável, o que dificulta que os dados sejam compartilhados de forma eficaz entre diferentes provedores de cuidados em saúde. Diante da necessidade de tornar o Registro Nacional da ELA cada vez mais seguro e confiável, interoperável e capaz de garantir a privacidade e segurança dos dados, o uso da tecnologia blockchain pode desempenhar um papel fundamental (Shahnaz; Qamar; Khalid, 2019). Ao fornecer um sistema mais seguro e transparente para compartilhar informações médicas e coordenar cuidados entre os membros da equipe de saúde, a blockchain pode simplificar a comunicação e garantir que todos os profissionais envolvidos tenham acesso aos dados mais recentes e precisos dos pacientes (Shi *et al.*, 2020).

No contexto da ELA, a implementação da tecnologia blockchain na Plataforma RevELA pode fortalecer a colaboração entre profissionais de saúde e otimizar o acompanhamento multidisciplinar e, em última instância, aspecto que melhora a qualidade dos cuidados oferecidos aos pacientes com ELA. Diante desse tema, foram definidas as seguintes questões de pesquisa:

¹ <https://revelanos.lais.ufrn.br/>

Q1: Quais são os principais desafios na implementação da blockchain nos registros eletrônicos em saúde?

Q2: Como a tecnologia blockchain pode melhorar a segurança e integridade dos dados médicos dos pacientes com ELA na Plataforma RevELA?

As questões de pesquisa foram formuladas com base na seguinte hipótese: “A implementação da tecnologia blockchain na Plataforma RevELA implica na melhoria da segurança e integridade dos dados médicos dos pacientes com ELA. Portanto, representa uma ferramenta mais qualificada e adequada para o compartilhamento das informações clínicas dos pacientes com ELA, especialmente porque o ambiente exige cuidados à saúde multidisciplinar, ou seja, entre diferentes profissionais de saúde.”

1.2 Objetivos

1.2.1 Objetivo Geral

Propor, desenvolver e implementar uma arquitetura *blockchain* para o gerenciamento de dados de pacientes e usuários do Registro Nacional da ELA de forma segura, auditável, imutável e interoperável.

1.2.2 Objetivos Específicos

- 1) Realizar uma revisão sistemática sobre o uso da tecnologia *blockchain* em sistemas de informação em saúde;
- 2) Avaliar os potenciais benefícios do uso da *blockchain* na segurança e integridade dos dados nos sistemas de informação em saúde;
- 3) Propor uma arquitetura *blockchain* para a Plataforma revELA;

- 4) Desenvolver e implementar um protótipo de sistema blockchain para a Plataforma revELA; e
- 5) Testar e validar a arquitetura blockchain especificada e prototipada.

1.3 Estrutura da dissertação

Esta dissertação está organizada em seis capítulos. O presente capítulo contextualiza o cenário da ELA e a sua problemática quanto à interoperabilidade e segurança dos dados, descreve também as questões de pesquisa e os objetivos a serem alcançados. O Capítulo 2 aborda os conceitos importantes para o entendimento dos principais assuntos tratados neste trabalho, detalhando mais sobre a ELA, o que é o Projeto revELA e o Registro Nacional da ELA, os principais conceitos sobre blockchain e sobre o *framework Hyperledger Fabric*, descreve sobre as variáveis utilizadas nas análises de performance da rede, e por fim, apresenta uma revisão sistemática com o estudo correlato que contribuiu com a discussão sobre a utilização da tecnologia blockchain em sistemas de informação em saúde, buscando entender o potencial da blockchain para melhorar a gestão de dados em saúde. O Capítulo 3 explica a metodologia utilizada para o desenvolvimento da solução proposta. O Capítulo 4 apresenta os detalhes técnicos da implementação e resultados, descrevendo sobre a lógica e código desenvolvidos para a criação da API e sua integração com o Registro Nacional da ELA, além de mostrar os resultados das análises de performance. O Capítulo 5 apresenta uma discussão mais ampla sobre os resultados deste trabalho, respondendo às questões de pesquisa. E, por fim, o Capítulo 6 apresenta as considerações finais, as principais contribuições e impactos sociais, um pouco sobre a minha trajetória acadêmica, as limitações, os desafios e a perspectiva de trabalhos futuros.

Capítulo 2

Referencial Teórico

2.1 Esclerose Lateral Amiotrófica (ELA)

A Esclerose Lateral Amiotrófica (ELA) é uma doença rara neurodegenerativa que afeta as células nervosas do cérebro e da medula espinhal, é caracterizada por uma progressiva paralisia muscular que reflete a degeneração dos neurônios motores, normalmente causando a morte do indivíduo devido ao bloqueio respiratório (Pagnini et al. 2012; Wijesekera; Leigh, 2009). Cerca de 10% dos pacientes com ELA têm uma forma lenta da doença com sobrevida de 10 anos ou mais, no entanto, a grande maioria tem uma sobrevida muito mais limitada após o diagnóstico (Longinetti; Fang, 2019). Geralmente a doença leva a paralisia progressiva e morte dentro de 2 a 5 anos após o início dos sintomas (Cook *et al.*, 2021).

Embora seja relativamente rara em comparação a outras condições neuromusculares, sua prevalência tem aumentado nos últimos anos, as projeções mostram que o número de casos de ELA no mundo passará de 222.801 em 2015 para 376.674 em 2040, representando um aumento de 69% (Arthur *et al.*, 2016). Os pacientes com ELA enfrentam desafios significativos, principalmente no diagnóstico, que muitas vezes é demorado, variando entre 9 e 12 meses, dificultando o tratamento que proporciona qualidade de vida ao paciente (Hulisz, 2018).

Devido a sua heterogeneidade, o monitoramento da progressão da doença é complexo e requer idealmente uma abordagem multidisciplinar (Radunović; Mitsumoto; Leigh, 2007). O acompanhamento por profissionais de saúde, tais como: fisioterapeutas, terapeutas ocupacionais, fonoaudiólogos, médicos pneumologistas, gastroenterologistas e assistentes sociais pode apresentar evolução significativa nos indicadores de sobrevida, com a redução da

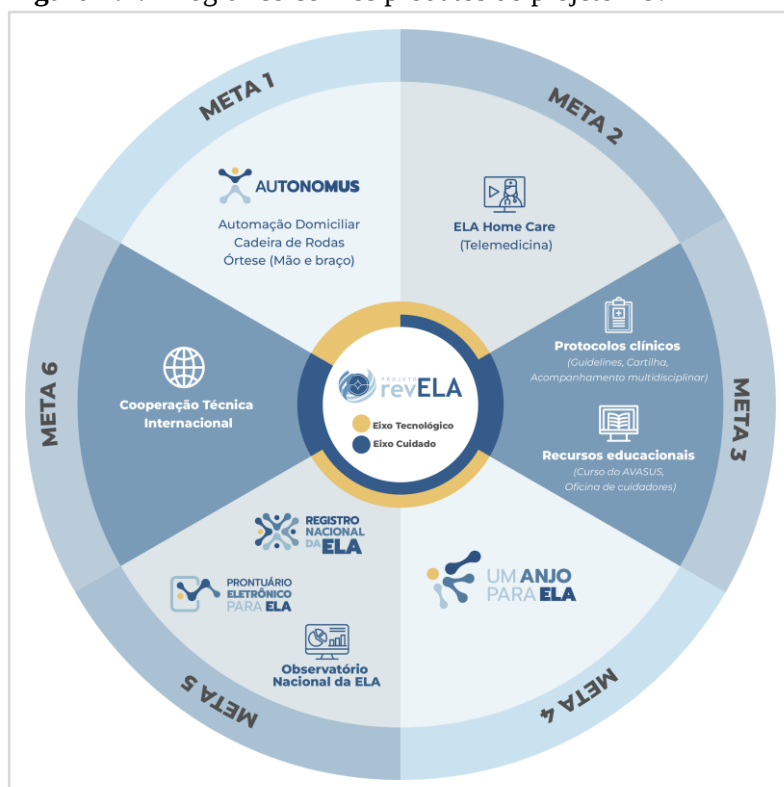
probabilidade de óbito em 45% ao longo de um período de cinco anos (Kiernan *et al.*, 2011). A coordenação eficiente da equipe multidisciplinar é essencial para proporcionar cuidados adequados e melhorar a qualidade de vida dos pacientes (Hogden *et al.*, 2017).

Mesmo sendo amplamente estudada, profissionais de saúde enfrentam o desafio de proporcionar cuidados de suporte abrangentes e melhorar a qualidade de vida dos pacientes, dada a ausência de tratamentos definitivos para a ELA, no qual os pilares do tratamento dos pacientes com ELA centram-se no controle dos sintomas (Radunović; Mitsumoto; Leigh, 2007; Ralli *et al.*, 2019).

2.2 Projeto RevELA

O projeto RevELA é caracterizado pelo desenvolvimento de produtos, pesquisas e estudos voltados aos cuidados para pacientes com ELA. Tais iniciativas são voltadas ao monitoramento, tratamento e definição de protocolos para o acompanhamento dos pacientes. O projeto é desenvolvido pelo LAIS/UFRN e financiado pelo Ministério da Saúde do Brasil através do Termo de Execução Descentralizada - TED nº 132/2018, e tem como objetivo o desenvolvimento de pesquisas e estudos voltados para tecnologias e inovações para tratamento, monitoramento e definição de protocolos para doenças do neurônio motor, com especial atenção para a Esclerose Lateral Amiotrófica. A Figura 2.1 apresenta as seis metas e seus respectivos produtos tecnológicos e protocolos relacionados ao projeto RevELA.

Figura 2.1: Infográfico com os produtos do projeto RevELA



Fonte: Adaptado de Barbalho *et al.* (2023)

Dentre os produtos do projeto está o Registro Nacional da ELA, que é uma plataforma desenvolvida para a coleta de dados epidemiológicos dos pacientes com ELA no Brasil, com o objetivo de armazenar os dados essenciais para melhorar a vigilância e monitoramento dos pacientes com ELA no Brasil, populando uma base de dados estruturada (Barbalho *et al.*, 2023).

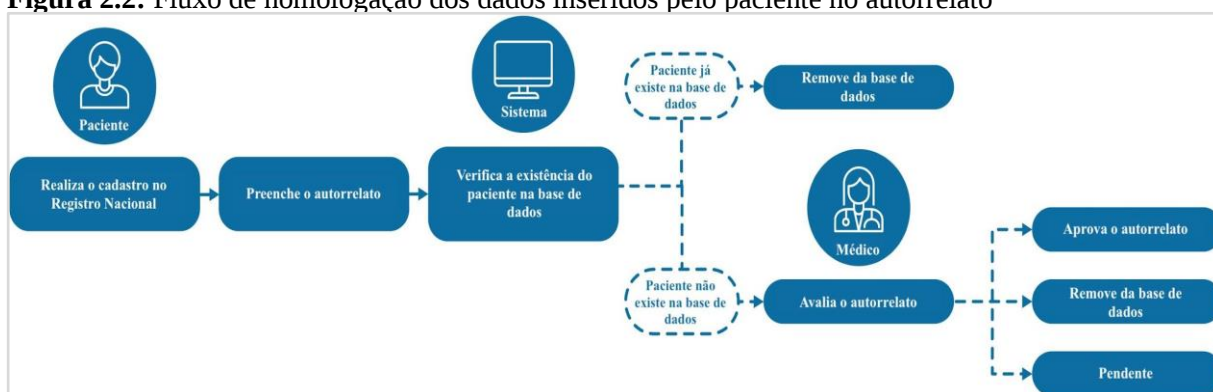
2.2.1 Registro Nacional da ELA

O Registro Nacional da ELA tem o objetivo de coletar as informações dos pacientes com ELA em todo território nacional a partir de três formulários principais: registro, autorrelato e *followup*. O formulário de registro e do *followup* são preenchidos pelo médico neurologista e coletam dados clínicos e de identificação. Já o autorrelato é respondido pelo próprio paciente, de forma que o paciente realiza seu cadastro na plataforma e insere suas informações sobre a doença. Conforme pode ser visto na Figura 2.2, o fluxo do autorrelato consiste em:

1. O paciente cadastra-se na plataforma e preenche o formulário de autorrelato.

2. O formulário fica pendente de aprovação pelos médicos que fazem parte do comitê responsável pela homologação desses pacientes. A homologação consiste na validação das informações passadas pelo paciente. Caso seja validado, o autorrelato é incluído no estudo e fica com o status de Homologado. Caso as informações não sejam verídicas, é utilizado o status de Retirado da Pesquisa, indicando que o caso não se trata do diagnóstico de ELA.

Figura 2.2: Fluxo de homologação dos dados inseridos pelo paciente no autorrelato



Fonte: Adaptado de Barbalho *et al.* (2023)

O Registro Nacional da ELA foi lançado em agosto de 2021 e está disponível em uma página web² de acesso público. O registro pode ser acessado por qualquer pessoa que atenda aos níveis de permissões. Atualmente o registro conta com mais de 700 pacientes cadastrados e com 25 médicos homologados. No total, foram preenchidos mais 400 registros e conta com mais de 300 autorrelatos homologados distribuídos entre os estados brasileiros. O estado do Rio Grande do Norte, que foi onde se iniciou o estudo, possui a maior quantidade de registros preenchidos pelo médico, sendo 156 pacientes, o que representa cerca de 36% do total. Seguido por Minas Gerais e São Paulo com 115 (27%) e 40 registros (9,3%), respectivamente.

Dos 27 estados brasileiros, 23 deles possuem algum paciente cadastrado. Apesar de não ter alcançado todos os estados da Região Norte (Acre, Amazonas, Roraima e Tocantins), isso

² <https://revelanos.lais.ufrn.br/>

mostra como a plataforma caminha bem para a coleta de dados dos pacientes com ELA em todo o território do país. Além disso, pode ser visto como uma peça chave para ser utilizado de forma interoperável por outros sistemas, para que seus dados possam ser utilizados pelos mais diversos estudos e consiga ajudar os pacientes com iniciativas de políticas públicas, fornecendo assim o suporte e auxílio a eles e suas famílias.

2.3 Blockchain

A tecnologia blockchain foi inicialmente proposta em 2008 por Satoshi Nakamoto como a base para o funcionamento do Bitcoin, uma moeda digital descentralizada (Nakamoto, 2008). Desde então, o conceito de blockchain evoluiu além do domínio financeiro, sendo adotado em diversos setores como recursos energéticos, *internet of things* (IoT), votação, governo e cultura, que já enxergam as possibilidades de progresso com a sua incorporação (Agbo; Mahmoud; Eklund, 2019; Hussien *et al.*, 2019; Khezzr *et al.*, 2019; Saeed *et al.*, 2022).

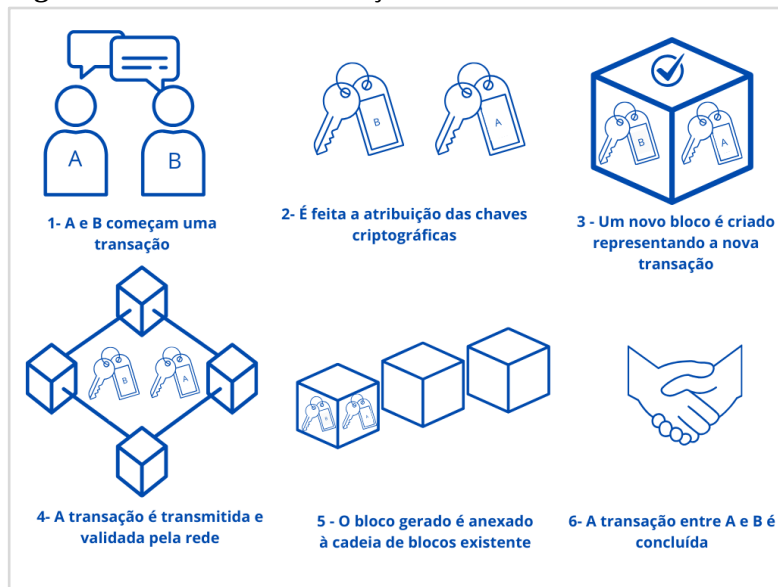
Blockchain é essencialmente um livro-razão (*ledger*) distribuído que registra transações, em uma série de blocos ordenados pelo tempo, de forma imutável, segura e auditável, sem a necessidade de uma autoridade central. Cada bloco contém uma lista de transações assinadas entre endereços junto com um *hash* do bloco anterior na cadeia (Cunningham *et al.*, 2021). O bloco *hash* anterior faz com que as informações na blockchain permaneçam imutáveis. A blockchain torna possível adicionar novos dados às transações; no entanto, não permite nenhuma alteração nelas. As transações são assinadas pela chave privada do usuário para garantir que a fonte de dados seja autenticada e não negada, e então o processo de *hash* é adicionado para verificar a integridade da transação e confirmar que ela é imutável (Pournaghi; Bayat; Farjami, 2020).

Veja na Figura 2.3 um exemplo bem simples do processo de uma transação em uma rede blockchain que ilustra como é utilizada criptografia, consenso e os registros distribuídos.

De forma resumida, pode-se atribuir as seguintes etapas:

1. **Início da transação:** A transação é iniciada entre duas partes, A e B, que desejam realizar uma operação, como a troca de informações ou ativos digitais.
2. **Atribuição de chaves criptográficas:** Cada parte (A e B) possui chaves criptográficas que garantem a segurança e autenticidade da transação. Essas chaves são utilizadas para assinar digitalmente a transação, assegurando que ela foi gerada pelas partes legítimas.
3. **Criação de um novo bloco:** A transação, juntamente com as assinaturas criptográficas, é agrupada em um novo bloco. Este bloco contém os detalhes da transação, como quem enviou, quem recebeu e os dados transferidos.
4. **Transmissão e validação:** O bloco contendo a transação é transmitido para a rede blockchain. A rede, composta por nós (ou *peers*) distribuídos, valida a transação por meio de um mecanismo de consenso, garantindo que ela seja legítima e que não haja duplicidade ou fraude.
5. **Anexação à blockchain:** Após a validação, o bloco é anexado à cadeia de blocos existente (blockchain), criando um registro permanente e imutável da transação.
6. **Conclusão da Transação:** A transação é considerada concluída e registrada de forma segura. As partes envolvidas (A e B) podem confiar que o registro está protegido contra alterações e acessível para auditorias futuras.

Figura 2.3: Processo de transação em uma rede blockchain



Fonte: Autoria Própria

2.3.1 Principais conceitos

A seguir são apresentados os principais conceitos que envolvem a blockchain. Todos esses conceitos apresentados nas subseções a seguir são baseados nas seguintes referências: Gaur *et al.* (2018), Sharma, Borah e Namasadura (2021), Chaudhry e Yousaf (2018) e Bach, Mihaljevic e Zagar (2018):

2.3.1.1 Blocos

O bloco é uma unidade de armazenamento que contém um conjunto de transações ou dados. Cada bloco inclui uma lista de transações registradas, um identificador único (*hash*) e o *hash* do bloco anterior, criando uma ligação sequencial entre os blocos.

2.3.1.2 Hash

O *hash* é um código único gerado por uma função matemática que representa os dados contidos no bloco. Qualquer alteração nos dados altera o *hash*, garantindo a integridade.

2.3.1.3 Transação

Uma transação é o registro de uma ação dentro da blockchain, como a transferência de um ativo digital, dados ou a execução de um contrato inteligente. Após ser criada, a transação é enviada à rede, validada pelos nós e registrada em um bloco.

2.3.1.4 Consenso

O consenso é o processo pelo qual os nós da rede concordam sobre a validade de novas transações e blocos. É essencial para a descentralização, pois elimina a necessidade de uma autoridade central. Como em uma rede blockchain, cada nó pode receber dados necessários em momentos diferentes, um mecanismo de consenso é necessário para determinar qual nó deve adicionar o novo bloco na blockchain (Sharma et al. 2021).

Alguns dos principais algoritmos de consenso são (Chaudhry e Yousaf 2018, Bach et al. 2018):

- *Proof of Work* (PoW): Exige que os mineradores resolvam problemas matemáticos complexos.
- *Proof of Authority* (PoA): Seleciona validadores com base na quantidade de tokens que possuem. É mais eficiente energeticamente.
- *Byzantine Fault Tolerance* (BFT): Usado em blockchains privadas para alcançar consenso rápido e eficiente.

2.3.1.5 Contratos Inteligentes (*smart contracts*)

O princípio básico dos contratos inteligentes é criar um acordo legal entre duas ou mais partes, em que cada parte é obrigada a cumprir com suas obrigações contratuais (Sharma et al. 2021). Por exemplo, em uma aplicação de saúde, um contrato inteligente pode liberar automaticamente um registro médico ao ser verificado o consentimento do paciente.

2.4 Hyperledger Fabric Framework

O *Hyperledger Fabric* é um dos *frameworks* de blockchain privado mais utilizados e foi desenvolvido sob a fundação Linux como parte do projeto Hyperledger³. Ele é projetado para aplicações empresariais que requerem segurança, escalabilidade e modularidade. A principal característica do Fabric é sua estrutura permissionada, na qual os participantes da rede são identificados e possuem permissões definidas, garantindo maior controle sobre o acesso aos dados.

Com base em sua documentação, disponível no site oficial do *framework*⁴, foram selecionados alguns dos principais conceitos que são fundamentais para o entendimento completo deste trabalho.

2.4.1 Pares, Organizações e Ordenadores

Na arquitetura do *Hyperledger Fabric*, os pares (*peers*) participam da rede blockchain sendo responsáveis por armazenar os dados e executar contratos inteligentes (*smart contracts* ou *chaincodes*). Eles podem ser classificados em dois tipos principais: pares que endossam transações e pares que apenas armazenam o histórico. Os pares pertencem às organizações, que representam entidades específicas dentro da rede, como hospitais ou laboratórios, garantindo um nível adicional de segregação e controle de acesso.

Os ordenadores (*orderers*) desempenham um papel fundamental como ponto de administração da rede, pois chegam ao consenso e garantem que todas as transações do bloco dentro da rede sejam executadas em ordem com base no valor acordado.

³ <https://hyperledger-fabric.readthedocs.io/en/release-2.2/whatis.html#hyperledger-fabric>

⁴ <https://hyperledger-fabric.readthedocs.io/en/release-2.2/whatis.html>

2.4.2 Identidade, *Wallet* e *Gateway*

A identidade dos participantes da rede é gerenciada por meio de um sistema de credenciais digitais, assegurando que apenas usuários autorizados possam interagir com o sistema. Elas determinam as permissões exatas sobre recursos e acesso a informações que os atores têm em uma rede blockchain.

As identidades são armazenadas em carteiras digitais (*wallets*), que funcionam como repositórios seguros de certificados e chaves privadas necessárias para autenticação e assinatura de transações.

Já o *gateway* atua como um ponto de acesso centralizado para simplificar a interação com a rede, especialmente para aplicações cliente, como sistemas de prontuários eletrônicos.

2.4.3 Contratos inteligentes (*chaincode*)

Os contratos inteligentes, implementados como *chaincode* no *Hyperledger Fabric*, são programas que definem a lógica de negócios a ser aplicada na rede. Eles automatizam processos, garantindo que as regras acordadas sejam seguidas sem a necessidade de intervenção humana.

2.4.4 Provedor de Serviços de Associação

O Provedor de Serviços de Associação, comumente referenciado como MPS que é a sigla que deriva do nome em inglês *Membership Service Provider*, é o mecanismo responsável por gerenciar as identidades e permissões na rede *Hyperledger Fabric*. Na prática, ele é um conjunto de pastas que são adicionadas à configuração da rede e é usado para definir uma organização tanto internamente (as organizações decidem quem são seus administradores) quanto externamente (permitindo que outras organizações validem que as entidades têm autoridade para fazer o que estão tentando fazer).

Ele trabalha em conjunto com a *Certificate Authority* (CA), que emite certificados digitais para autenticar os participantes. Essas estruturas são fundamentais para assegurar que cada organização ou usuário na rede seja confiável e opere dentro dos limites de sua autorização.

2.4.5 Usuários da rede

O conceito de usuários é importante para garantir que apenas entidades autorizadas possam interagir com a rede blockchain e está diretamente relacionado com o MSP, explicado no tópico anterior. A criação de usuários envolve a definição das identidades digitais para os administradores e usuários comuns que irão realizar transações, validando que apenas aqueles que possuem autorização possam executar determinadas ações. Existem dois tipos de usuários:

- Administradores
 - Responsáveis por gerenciar a rede, incluindo a configuração de canais, políticas e a administração de identidades.
 - Têm permissões elevadas e podem realizar ações administrativas.
- Usuários
 - Entidades que realizam transações na rede, como criar, ler, atualizar e excluir dados.
 - Têm permissões limitadas com base nas políticas definidas pela rede.

2.5 Análise de performance

A análise de performance implica em medir e avaliar o desempenho de uma rede blockchain sob diferentes condições de carga de trabalho, submetendo a rede a um número diferente de transações enviadas durante diferentes intervalos de tempo (Dinh *et al.*, 2017). Essa

análise é importante para identificar gargalos, otimizar a configuração da rede e garantir que ela possa atender aos requisitos de desempenho esperados.

Com o intuito de ajudar na avaliação de forma precisa os atributos que impactam no desempenho, o *Hyperledger Performance and Scalability Working Group*⁵ desenvolveu um artigo em que define os termos e métricas relevantes aplicáveis em várias plataformas de tecnologia de razão distribuída, em inglês, *Distributed Ledger Technology (DLT)*.

As métricas fornecem *insights* sobre a capacidade da rede de processar transações, a eficiência com que as transações são confirmadas e a estabilidade da rede sob diferentes cargas de trabalho (Pongnumkul; Siripanpornchana; Thajchayapong, 2017). Por fim, as métricas ajudam a identificar áreas de melhoria e a garantir que a rede blockchain possa operar de forma eficiente.

2.5.1 Taxa de envio

A taxa de envio de transações por segundo indica o número de transações enviadas para a rede por segundo. Esse valor é configurado pelas ferramentas de benchmarking para especificar a carga de trabalho utilizada (Gaur *et al.*, 2018).

2.5.2 Latência

A latência de transação é uma métrica essencial para entender a eficiência e a capacidade de resposta de uma rede blockchain. Ela mede o tempo total necessário para que uma transação passe por todas as etapas de processamento e seja confirmada na rede, esses tempos são chamados de tempo de conclusão e o tempo de implantação, respectivamente, e a latência mede a diferença entre eles (Pongnumkul; Siripanpornchana; Thajchayapong, 2017).

⁵ HYPERLEDGER Blockchain Performance Metrics. 2018. Available at: <https://www.lfdecentralizedtrust.org/learn/publications/blockchain-performance-metrics>. Access in: 20 nov. 2024

Essa métrica pode afetar a experiência do usuário e a eficiência da rede, visto que, baixa latência significa que as transações são confirmadas rapidamente, o que é essencial para aplicações que exigem respostas rápidas.

$$latência = t_{implantação} - t_{envio}$$

2.5.2.1 Latência máxima

A partir do conceito da latência, é derivado o de latência máxima ($latência_{max}$) que é o maior tempo registrado para uma transação ser confirmada na rede. Esta métrica ajuda a identificar picos de latência que podem indicar problemas de desempenho ou gargalos na rede.

$$latência_{max} = \max(t_{latência})$$

2.5.2.2 Latência mínima

A latência mínima ($latência_{min}$) é o menor tempo registrado para uma transação ser confirmada na rede. Isto é, essa métrica diz respeito ao menor valor de tempo dentre todas as latências individuais coletadas ($t_{latência}$).

$$latência_{min} = \min(t_{latência})$$

2.5.2.3 Latência média

A latência média ($latência_{media}$) é o tempo médio que leva para uma transação ser confirmada na rede. O cálculo da latência média é feito somando-se o tempo das latências coletadas individualmente ($t_{latência}$) dividido pelo número de coletas (N).

$$latência_{media} = \sum t_{latência} / N$$

2.5.3 Taxa de transferência de transações

A taxa de transferência de transações, ou apenas *throughput*, mede o número de transações confirmadas pela rede por segundo, sendo importante para avaliar a capacidade da

rede de processar transações de forma eficiente (Hyperledger [...], 2018; Pongnumkul; Siripanpornchana; Thajchayapong, 2017).

2.6 Blockchain em Sistemas de Informação em Saúde

Apesar de ser originalmente desenvolvida como uma solução para o registro de transações em moedas, a tecnologia de blockchain vem atraindo bastante atenção em diversas áreas que já enxergam as possibilidades de progresso com a sua incorporação em diversos setores, estendendo-se além de aplicações de uso financeiro e representando um forte aliado aos sistemas de saúde em problemáticas relacionadas à segurança, privacidade e compartilhamento de dados. Para compreender o atual cenário do uso dessa tecnologia dentro do campo da saúde digital, foi elaborada uma Revisão Sistemática de Literatura (RSL) com o objetivo de explorar as aplicações da blockchain nos sistemas de informação em saúde, destacando suas vantagens e seus desafios. Dessa forma, é possível entender o potencial da blockchain para melhorar a gestão de dados em saúde.

A RSL foi baseada nas diretrizes de revisão sistemática propostas por Kitchenham (2004) e conduzida com base na lista de verificação Preferred Reporting Items for Systematic Reviews and Meta-Analyses (PRISMA) (Page *et al.*, 2021). Além disso, foi registrada no International prospective register of systematic reviews (PROSPERO), sob o número de registro CRD42024564961⁶. As buscas foram realizadas em 31 de outubro de 2023 em seis bases de dados, são elas: ACM Digital Library, Web of Science (WOS), ScienceDirect (Elsevier), IEEE Explore, PubMed e Springer. Os detalhes da execução do protocolo da RSL foram publicados por Fonsêca *et al.* 2024.

⁶ https://www.crd.york.ac.uk/PROSPERO/display_record.php?RecordID=564961

2.6.1 Resultados e Discussões

Após a execução do protocolo de pesquisa, dos 4864 estudos encontrados nas seis bases de dados pesquisadas, apenas 73 deles foram incluídos e selecionados para uma análise mais detalhada. Os resultados mostraram que, dentre as aplicações da blockchain em sistemas de informação em saúde, o uso em prontuários eletrônicos representa 63% dos estudos incluídos. Além disso, a blockchain foi usada em estudos que abordaram Internet das Coisas Médicas (8,2%) e para compartilhamento de dados durante a pandemia de COVID-19 (6,8%).

Quanto às práticas e modelos mais utilizados na implementação da blockchain dos estudos selecionados, a maioria fez uso do Ethereum (57.9%), seguidos de alguma das versões do *Hyperledger* (*Hyperledger Fabric*, *Hyperledger Aries* e *Hyperledger Indy*) com 35.1%, ou ainda, fizeram uso tanto do Ethereum quanto do Hyperledge (5.3%). Apenas um dos trabalhos fez o uso do Exonum, uma estrutura de blockchain de código aberto.

Analisando a permissividade da rede, ou seja, quem pode participar da validação das transações e da visibilidade dos dados armazenados na blockchain, considerando-se que apenas 34 dos estudos citaram sobre esse aspecto; destes, 16 (47,1%) eram redes privadas, 9 (26,5%) públicas, 3 (8,8%) consórcio e os 6 restantes (17,6%) utilizavam mais de um dos métodos e foram classificados para esta revisão sistemática como híbridos.

Os resultados mostraram também o aumento no número de publicações por ano e, usando como referência o país de cada autor, foi possível notar a cooperação científica internacional sobre o tema. E por fim, identificou-se quais as principais problemáticas que se buscava solucionar com o uso da blockchain dentre: segurança, privacidade, compartilhamento e distribuição de dados, chegando aos seguintes dados: 45 dos estudos fizeram uso da tecnologia blockchain para resolver problemas de segurança, outros 38 se interessaram em utilizá-la para fins de compartilhamento de dados, 25 deles fizeram uso dos benefícios da tecnologia quando

o assunto é privacidade e, por fim, 4 deles utilizaram a descentralização como meio de solucionar o problema enfrentado.

Baseado nos resultados apresentados anteriormente, através da RSL foi possível notar que a blockchain está se tornando uma tecnologia madura e amplamente aceita, visto o seu alcance global e aumento contínuo dos estudos na área. A adoção em sistemas de informação de saúde provou oferecer benefícios importantes em termos de privacidade, segurança, compartilhamento de dados e interoperabilidade, além de exemplificar a sua alta aplicabilidade.

Quanto a sua utilização em sistemas de informação em saúde, a blockchain se destacou pelo uso no gerenciamento de informações médicas sobre pacientes como: histórico médico, resultado de exames, prescrição e registro de saúde. As características dessa tecnologia como: a descentralização, onde os dados são compartilhados e armazenadas dentre as organizações participantes da rede; imutabilidade, já que o conteúdo das transações não podem ser modificados ou apagados; e a interoperabilidade, visto que os usuários que precisam ter acesso aos dados podem ter acesso a eles de forma segura sendo adicionados a rede, resultam em uma plataforma ideal para uso em sistemas de saúde, garantindo um compartilhamento seguro e a integridade dos dados.

No entanto, apesar das vantagens promissoras, a implementação de blockchain na saúde enfrenta desafios como padronização, questões regulatórias e integração, além de exigir mais pesquisas sobre impacto econômico e escalabilidade. É essencial criar um ambiente que promova soluções seguras e eficazes, a colaboração entre pesquisadores e profissionais é vital para transformar desafios em oportunidades. O interesse global pode impulsionar avanços significativos na qualidade dos serviços de saúde pública.

Capítulo 3

Materiais e Métodos

Este capítulo apresenta as ferramentas utilizadas para a configuração e testes de performance da rede e dos contratos inteligentes criados para serem utilizados pelos projetos da Plataforma revELA. O *Hyperledger Fabric framework* foi utilizado como a base para a construção da rede e como principal suporte para criação e utilização dos contratos inteligentes. O *Hyperledger Caliper*, que também é um *framework*, foi utilizado para *benchmarking* de desempenho da rede blockchain.

Os testes para análise de performance da rede *blockchain* foram realizados em ambiente local com processador Intel(R) Core(TM) i7-10510U CPU@1.80GHz 2.30GHz, 16GB de memória RAM rodando no sistema operacional Windows 10 Home Single Language. As versões dos softwares utilizados foram: *Hyperledger Fabric* 2.5.9, *Hyperledger Caliper* 0.6.0, Docker 27.0.3 e Node.js 18.20.4.

3.1 Hyperledger Fabric Network

A rede blockchain utilizada foi desenvolvida com o *framework Hyperledger Fabric*, que é uma infraestrutura usada na construção de aplicações baseados em blockchain, fornecendo uma arquitetura que define funções entre os nós, execução de contratos inteligentes e serviços de consenso que podem ser configurados.

Essa ferramenta foi escolhida devido a três fatores principais: primeiramente, por ser uma blockchain permissionada, ou seja, os usuários participantes podem ter diferentes permissões de acesso e isso é uma boa alternativa para o contexto de gerenciamento de dados de pacientes por parte de diferentes organizações; é uma plataforma grátis, não tendo custos

para ser utilizada e, por fim, foi a rede blockchain privada mais utilizada dentre os estudos analisados na RSL de Fonsêca et al. 2024, mostrando sua relevância entre outros trabalhos.

O *framework* disponibiliza de uma rede já configurada para que os desenvolvedores possam validar seus contratos inteligentes e suas aplicações, utilizando imagens Docker⁷ em ambiente local. Essa rede de teste foi utilizada neste trabalho e suas características são:

- Inclui dois nós pares (*peers nodes*): são máquinas trabalhadoras, responsáveis por validar as transações e gravarem nos ledgers. Além disso, os pares também executam os contratos inteligentes que contém a lógica de negócio da rede.
- Inclui um nó de ordenação (*ordering node*): possuem uma função administrativa dentro da rede, garantindo que todas as transações do bloco sejam executadas em ordem com base no valor acordado.

A rede de teste representa uma infraestrutura ideal para a validação da aplicação desenvolvida neste trabalho, pois para a criação de uma rede blockchain são necessárias diferentes máquinas, já que o conceito de blockchain é voltado à questão da descentralização, onde os dados são replicados em cada uma e o trabalho é dividido entre as diferentes máquinas. Porém, no ambiente de desenvolvimento, a utilização de diferentes máquinas físicas apenas para essa criação/validação é algo custoso e demanda um maior trabalho. Então, é possível criar uma rede blockchain totalmente virtual utilizando os *containers* Dockers com uma configuração simples pré-estabelecida pela rede de exemplo do *Hyperledger Fabric*.

Após a inicialização da rede, são necessárias quatro etapas para a configuração completa exemplificadas na Figura 3.1 e descritas a seguir.

⁷ <https://www.docker.com/>

Figura 3.1: Etapas para a configuração da rede blockchain



Fonte: Autoria Própria

1. Criação do canal: O canal representa a forma de comunicação entre os pares (*peers*) da rede. Com um canal de comunicação privado estabelecido entre as máquinas ou membros da rede é possível configurar para que apenas organizações convidadas tenham visibilidade das mensagens que são enviadas dentro do canal, ficando “invisível” para outros membros da rede. Após a criação do canal, os nós que fazem parte dele devem ser adicionados.

2. Criação e implantação dos contratos inteligentes: Os contratos inteligentes criados nada mais são do que arquivos da linguagem JavaScript que fazem uso do *Hyperledger Fabric* SDK para Node.js e definem a lógica de negócio, descrevendo as funções que serão utilizadas para a interação com a rede. Com os contratos inteligentes criados, é feito o empacotamento e a implantação na rede, permitindo que a lógica seja

aplicada corretamente e que as transações sejam registradas de forma segura e imutável pela blockchain.

3. Criação e inscrição dos usuários: A criação de usuários é importante para garantir que apenas entidades autorizadas possam interagir com a rede blockchain. Esta etapa envolve a criação de identidades digitais para os administradores e usuários que irão realizar transações, de forma que, apenas usuários com permissões adequadas podem realizar determinadas ações. Primeiramente, é necessário criar um usuário administrador, pois este será o responsável por gerenciar a rede e adicionar outros usuários a ela. Em seguida, com o administrador já criado, é possível adicionar novos usuários na rede.

4. Desenvolvimento de uma aplicação para interagir com a rede: Após a configuração completa da rede é necessário criar uma aplicação cliente para interagir com ela. A criação dessa API será descrita em detalhes a seguir na seção 4.2.

3.2 Hyperledger Caliper

O *Hyperledger Caliper*⁸ é uma ferramenta de *benchmark* que permite aos usuários medir o desempenho de diferentes soluções de blockchain, tendo suporte hoje para *Hyperledger Besu*⁹, *Ethereum*¹⁰ e *Hyperledger Fabric*¹¹. Com o Caliper é possível definir diversos cenários de testes e medir indicadores como taxa de transação por segundo, latência de transação e taxa de sucesso de transações. O Caliper dá suporte para as seguintes métricas de performance: transação/rendimento de leitura, latência de transação/leitura (mínimo, máximo, médio,

⁸ <https://www.lfdecentralizedtrust.org/projects/caliper>

⁹ <https://github.com/hyperledger/besu>

¹⁰ <https://github.com/ethereum/>

¹¹ <https://github.com/hyperledger/fabric>

percentil) e consumo de recursos (tais como: CPU, memória, E/S de rede, entre outros). Sua documentação¹² é bem abrangente e cobre diversos aspectos da ferramenta, desde a instalação até a execução dos *benchmarks*, interpretação dos resultados e fornece exemplos práticos para ajudar os usuários a obterem o máximo disponibilizado pela plataforma.

Para executar um *benchmark* com o *Hyperledger Caliper*, é preciso configurar pelo menos três tipos de arquivos que definem os parâmetros do teste, a configuração da rede blockchain, e as transações específicas que serão enviadas para a rede:

1. O arquivo de configuração do *benchmark* (*benchmark.yaml*): define os parâmetros do teste, incluindo o número de rodadas de testes a serem usados ao gerar a carga, a carga de trabalho, e as métricas a serem coletadas.
2. O arquivo de configuração da rede (*networkConfig.yaml*): define a configuração da rede blockchain, incluindo os nós, canais, contratos e identidades. Esse arquivo é necessário para que o Caliper possa enviar e avaliar transações à rede *Hyperledger Fabric*.
3. Os módulos de *workload*: definem as transações específicas que serão enviadas para a rede durante o benchmark, isto é, os módulos *workload* interagem com o contrato inteligente implantado durante a rodada de *benchmark*. Cada módulo de *workload* é um arquivo JavaScript que implementa a lógica de envio de transações.

Ao final da execução de todos os módulos e cenários de testes que foram configurados, o Caliper gera um relatório em formato HTML, no diretório de *workspace*, com informações sobre nome do arquivo de configuração do benchmark, número de sucessos e falhas de envio das transações, taxa de envio, latência e número de transações processadas por segundo. Os resultados são apresentados através de tabelas, onde as colunas representam as métricas avaliadas e as linhas correspondem aos módulos criados.

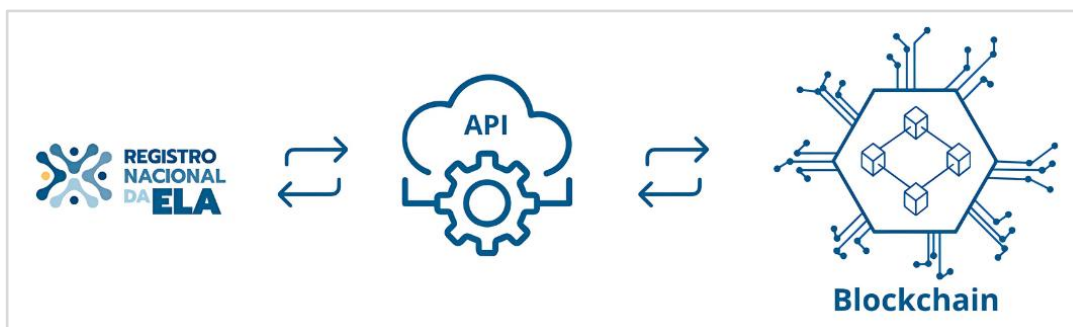
¹² <https://hyperledger-caliper.github.io/caliper/v0.6.0/getting-started/>

Capítulo 4

Implementação e Resultados

Este capítulo apresenta os detalhes técnicos da implementação e resultados, descrevendo sobre a lógica e código desenvolvidos para a criação da API e sua integração com o Registro Nacional da ELA, além de mostrar os resultados das análises de performance. Veja na Figura 4.1 uma ilustração de como essa integração ocorre.

Figura 4.1: Fluxo de integração do Registro Nacional da ELA com a blockchain



Fonte: Autoria própria

4.1 Contratos inteligentes (*smart contract*)

Os contratos inteligentes são responsáveis por definir a lógica de negócio que é aplicada aos dados armazenados no ledger. Eles definem as funções e as regras para criar, ler, atualizar e excluir ativos no ledger. O contrato inteligente criado para este trabalho define a lógica para o gerenciamento dos registros de pacientes. Para isso, define as seguintes funções:

Figura 4.3: Detalhes de implementação da função RegisterAccessPatientData

```
// RegisterAccessPatientData registra o acesso aos dados de um paciente no ledger
async RegisterAccessPatientData(context, cpf, name, doctorName, doctorCRM, accessDate) {
  // Cria um objeto de acesso com os dados fornecidos
  const patientAccess = {
    cpf,
    name,
    doctorName,
    doctorCRM,
    accessDate,
    action: 'ACCESS',
  };

  // Converte o objeto paciente para uma string JSON e depois para um Buffer
  const patientAccessBuffer = Buffer.from(stringify(sortKeysRecursive(patientAccess)));

  // Armazena o par chave-valor no ledger
  await context.stub.putState(cpf, patientAccessBuffer);

  // Retorna a string JSON do acesso
  return JSON.stringify(patientAccess);
}
```

Fonte: Autoria Própria

4.1.3 UpdatePatientData

A função de *UpdatePatientData* recebe da requisição e salva os mesmos parâmetros da função *RegisterPatient*, diferindo apenas a action, que na operação de atualização é passado o valor “UPDATE”, como pode ser visto na Figura 4.4. Sempre que o dado do paciente (nome, data de nascimento, sexo, cor, entre outras informações de identificação) ou de se registro forem modificados, será salvo no ledger para se manter com os dados atualizados.

Figura 4.4: Detalhes de implementação da função `UpdatePatientData`

```
// UpdatePatientData atualiza os dados de um paciente no ledger com o CPF fornecido
async UpdatePatientData(context, cpf, name, motherName, dateOfBirth, phone, currentCity, sex, race, doctorName, doctorCRM, registry) {
  // Cria um objeto paciente com os dados fornecidos especificando o tipo de ação como UPDATE
  const patient = {
    cpf,
    name,
    motherName,
    dateOfBirth,
    phone,
    currentCity,
    sex,
    race,
    doctorName,
    doctorCRM,
    registry,
    action: 'UPDATE',
  };

  // Converte o objeto paciente para uma string JSON e depois para um Buffer
  const patientBuffer = Buffer.from(stringify(sortKeysRecursive(patient)));

  // Armazena o par chave-valor no ledger
  await context.stub.putState(cpf, patientBuffer);

  // Retorna a string JSON do paciente
  return JSON.stringify(patient);
}
```

Fonte: Autoria Própria

Perceba que, assim como as funções *RegisterPatient* e *RegisterAccessPatientData*, o *UpdatePatientData* salva no *ledger* um par de chave-valor, onde a chave é o CPF do paciente. No *Hyperledger Fabric* é necessário que cada registro salvo no *ledger* seja identificado por uma chave única, logo, ao tentar salvar mais de um registro para o mesmo paciente, o novo valor irá sobrescrever o já existente, mantendo apenas o valor mais recente para a chave CPF. Por exemplo: Foi feito o registro do paciente com CPF 111.111.111-11 salvando os dados no ledger através da função *RegisterPatient*. Quando o paciente tem seus dados atualizados, a função *UpdatePatientData* é chamada para salvar as novas informações no *ledger*. Nesse momento, já existe no *ledger* um registro para o paciente de CPF 111.111.111-11 (que foi feito pela função *RegisterPatient*), como não é possível manter no *ledger* dois registros com a mesma chave, apenas aquele feito com a função *UpdatePatientData* (operação mais recente) será mantido. Embora o valor seja sobrescrito, o histórico de transações mantém um registro de todas as operações realizadas. Isso significa que é possível consultar o histórico dos pacientes para ver todas as versões anteriores do valor associado ao CPF.

4.1.4 GetPatient

Esta função recebe como parâmetro o CPF do paciente que se deseja recuperar as informações. Como explicado no tópico anterior, o ledger irá manter apenas o registro mais atualizado para o paciente, então serão retornados sempre os dados da última operação, podendo ser de uma ação de “*CREATE*”, “*UPDATE*” ou “*ACCESS*”. Esta função é interessante para entender qual a atualização mais recente realizada para aquele paciente. Detalhes da função estão na Figura 4.5.

Figura 4.5: Detalhes de implementação da função GetPatient

```
// GetPatient retorna os dados do paciente com o CPF fornecido
async GetPatient(context, cpf) {
  // Recupera o paciente com o CPF fornecido do ledger
  const patientJSON = await context.stub.getState(cpf);

  // Se o paciente não existir, uma exceção é lançada
  if (!patientJSON || patientJSON.length === 0) {
    throw new Error(`O paciente de CPF: ${cpf} não existe`);
  }

  // Retorna a string JSON do paciente
  return patientJSON.toString();
}
```

Fonte: Autoria Própria

4.1.5 GetPatientHistory

A função *GetPatientHistory* foi implementada para retornar todas as operações que foram realizadas para um dado paciente passando como parâmetro o CPF. Além do registro mais recente que está salvo no *ledger*, ela retorna o histórico de todas as versões anteriores de registros salvos. Esta função é ideal para a auditoria e acompanhamento do histórico de modificações realizadas para os dados de identificação do paciente e seu registro clínico. Seu código pode ser visto com mais detalhes na Figura 4.6.

Figura 4.6: Detalhes de implementação da função `GetPatientHistory`

```

// GetPatientHistory retorna o histórico de alterações do paciente com o CPF fornecido
async GetPatientHistory(context, cpf) {
  // Cria um iterador para percorrer o histórico de alterações do paciente
  const iterator = await context.stub.getHistoryForKey(cpf);
  const patientHistory = [];

  // Se o paciente não existir, uma exceção é lançada
  if (!iterator || iterator.length === 0) {
    throw new Error(`O paciente de CPF: ${cpf} não existe`);
  }

  // Enquanto houver registros no histórico, adiciona o registro ao array de resultados
  while (true) {
    // Obtém o próximo registro do histórico
    const history = await iterator.next();

    // Se houver um registro, adiciona o registro ao array de resultados
    if (history.value) {
      const record = {
        txId: history.value.txId,
        timestamp: history.value.timestamp,
        isDelete: history.value.isDelete.toString(),
        value: history.value.value.toString('utf8')
      };
      patientHistory.push(record);
    }

    // Se não houver mais registros, sai do loop, encerrando o iterador e retornando o array de resultados
    if (history.done) {
      await iterator.close();
      return JSON.stringify(patientHistory);
    }
  }
}

```

Fonte: Autoria Própria

4.1.6 GetAllPatients

A função *GetAllPatients* retorna os dados mais recentes de todos os pacientes que estão registrados no ledger. Detalhes da função podem ser vistos na Figura 4.7.

Figura 4.7: Detalhes de implementação da função GetAllPatients

```
// GetAllPatients retorna todos os pacientes do ledger
async GetAllPatients(context) {
  const allPatients = [];

  // Cria um iterador para percorrer todos os registros no ledger
  const iterator = await context.stub.getStateByRange('', '');

  // Obtém o primeiro paciente
  let patient = await iterator.next();

  // Enquanto houver pacientes, adiciona o paciente ao array de resultados
  while (!patient.done) {
    // Converte o valor do paciente de um buffer para uma string
    const strValue = Buffer.from(patient.value.value.toString()).toString('utf8');
    let record;

    // Tenta converter a string JSON para um objeto
    try {
      record = JSON.parse(strValue);
    } catch (err) {
      record = strValue;
    }

    // Adiciona o paciente ao array de resultados
    allPatients.push(record);

    // Obtém o próximo paciente
    patient = await iterator.next();
  }

  // Retorna a string JSON de todos os pacientes
  return JSON.stringify(allPatients);
}
```

Fonte: Autoria Própria

4.2 API para comunicação entre os sistemas e a rede blockchain

Pensando na utilização da rede blockchain e dos contratos inteligentes pelos diversos projetos que compõem a Plataforma revELA e com a finalidade de abstrair toda a complexidade da integração direta entre os sistemas e a rede, foi criado um programa para atuar como uma API atuando como um intermediário entre as aplicações.

O *Hyperledger Fabric* é uma das primeiras plataformas a dar suporte à criação de contratos inteligentes em linguagens de programação de uso geral, como Java, Go e Node.js.

Devido a essa facilidade e ao *Hyperledger Fabric* SDK permitir que aplicativos interajam com uma rede de blockchain Fabric utilizando o Node.js, a API foi implementada utilizando-se a linguagem de programação JavaScript. A escolha da linguagem foi baseada principalmente na familiaridade com a linguagem.

4.2.1 Estabelecer uma conexão segura com a rede blockchain Hyperledger Fabric

1. Carrega o arquivo de perfil de conexão que contém informações sobre a rede como os nós disponíveis, a organização, os canais, os pares, os ordenadores e as credenciais. Nesse arquivo estão as configurações necessárias para que a aplicação saiba como se comunicar com a rede.
2. Configura a *Wallet*: o *wallet* armazena as chaves privadas e certificados, isto é, a identidade de um usuário ou organização pertencente a rede. Essa configuração é importante para verificar se uma identidade necessária está disponível na *wallet* para autenticação.
3. Conecta ao *gateway*: o *gateway* é ponto de entrada para a interação com a rede. Ele é responsável por gerenciar a conexão e a comunicação entre a API e a rede. Para sua configuração são utilizados a *wallet* e os dados carregados no arquivo de perfil de conexão.
4. Obtém o canal da rede: uma vez conectado, o cliente acessa o canal especificado no perfil de conexão. Um canal pode ser entendido como uma rede privada dentro da *blockchain* onde transações específicas são registradas.
5. Obtém o contrato inteligente: os contratos definem as funções que podem ser invocadas na blockchain e através dele é possível executar as transações e consultas.

4.2.2 Validar se a requisição recebida pela API possui um *token* válido

Uma abordagem comum para autenticação em APIs é o uso de *tokens* JWT (*JSON Web Tokens*), onde cada aplicação autorizada que deseja se comunicar com a API precisa gerar um JWT e incluir esse *token* em cada requisição. A API desenvolvida faz essa validação do *token* antes de permitir o acesso, garantindo uma camada a mais de segurança a nível de requisição.

4.2.3 Definir as rotas que executarão os métodos definidos pelos contratos inteligentes

As rotas são o meio pelo qual os sistemas que acessam a API podem armazenar e recuperar os dados diretamente da rede.

4.3 Utilização da API pelo Registro Nacional da ELA

Para abstrair a lógica de comunicação do Registro Nacional da ELA com a API das outras partes do código do sistema, foram criadas duas classes: *BlockchainIntegration* e *JwtHelper*.

4.3.1 Classe *BlockchainIntegration*

Essa classe possui três métodos principais: *registerPatientBlockchain*, *registerPatientAccess* e *updatePatientBlockchain*, mostrados na Figura 4.8, e seis métodos auxiliares que implementam alguma busca ou formatação dos dados, listados na Figura 4.9. Os métodos principais representam as ações que o sistema executará dentro da rede blockchain. Cada um deles possui uma estrutura bem semelhante: montam a requisição buscando pelos dados que serão enviados, geram o token que validará a comunicação e, por fim, fazem uma chamada HTTP do tipo *POST* para as rotas configuradas pela API.

Figura 4.8: Métodos principais da classe BlockchainIntegration

```

app > Http > Requests > BlockchainIntegration.php
1  <?php
2
3  namespace App\Http\Requests;
4
5  use GuzzleHttp\Client;
6  use App\Helpers\JwtHelper;
7  use App\Medico;
8  use App\Municipio;
9  use App\Uf;
10 use App\Opcao;
11 use App\Paciente;
12 use App\Registro;
13 use App\RegistroPerguntaResposta;
14 use App\User;
15
16 class BlockchainIntegration {
17     // Métodos de integração com a blockchain
18
19     // Método para registrar um paciente na blockchain
20 > public function registerPatientBlockchain($registry){ ...
55     }
56
57     // Método para registrar o acesso de um paciente na blockchain
58 > public function registerPatientAccess($patient, $doctor){ ...
87     }
88
89     // Método para atualizar um paciente na blockchain
90 > public function updatePatientBlockchain($registry){ ...
126     }
127

```

Fonte: Autoria Própria

Figura 4.9: Métodos auxiliares da classe BlockchainIntegration

```

128 // Métodos auxiliares
129
130 // Método para converter a string para UTF-8
131 > private function convert($data){ ...
133 }
134
135 // Método para obter os dados do registro
136 > private function getRegistryData($registry){ ...
138 }
139
140 // Método para filtrar os dados do registro retornando apenas a pergunta e a resposta
141 > private function filterRegistry($answers) { ...
143 }
144
145 // Método para reindexar o array
146 private function reindexArray(&$array)
147 { ...
148 }
149
150 // Métodos para agrupar respostas formadas por mais de uma pergunta
151 private function groupingAnswersID10($return, $registry)
152 { ...
153 }
154
155 private function groupingAnswersID93($return, $registry)
156 { ...
157 }
158 }
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246

```

Fonte: Autoria Própria

A classe *BlockchainIntegration* é instanciada dentro das classes *FormularioController* e *PacienteController*, estas possuem métodos chamados quando um registro é criado, quando um paciente possui seus dados modificados ou quando a página de detalhes do paciente é acessada. De forma mais detalhada:

- O método *registerPatientBlockchain* é chamado dentro do método *store* da classe *FormularioController*, sendo responsável por enviar para a rede os dados de um paciente quando ele tem o seu registro cadastrado;
- O método *registerPatientAccess* é chamado dentro do método *show* da classe *PacienteController* e registra as informações sobre o médico e o paciente sempre que os dados de um paciente são acessados.
- O método *updatePatientBlockchain* é chamado pelo método *update* da classe *FormularioController*, *updateDadosIdentificacao* e *updateDadosPessoais* da classe

PacienteController, armazenando as informações atualizadas do paciente e do seu registro.

4.3.2 Classe *JwtHelper*

Essa classe possui apenas um método, o *createJWTToken*. Esse método acessa a variável de ambiente que armazena a *JWT_SECRET*, que é uma chave secreta que serve para assinar e verificar a autenticidade de tokens JWT, e monta o *payload* com as informações úteis que serão enviadas juntamente com a requisição, como por exemplo: tempo de expiração do token (*exp*) e hora em que o token foi gerado (*iat*). Por fim, o token criado é retornado. Veja a implementação desse método na Figura 4.10.

Figura 4.10: Método *createJWTToken* da classe *JwtHelper*

```
app > Helpers > JwtHelper.php
1  <?php
2
3  namespace App\Helpers;
4
5  use Firebase\JWT\JWT;
6  use Firebase\JWT\Key;
7
8  class JwtHelper {
9
10     // Cria um token JWT
11     public static function createJWTToken($data) {
12         // Recupera a chave secreta definida no arquivo .env
13         $key = env('JWT_SECRET');
14
15         // Define o payload do token
16         $payload = [
17             'iat' => time(), //timestamp de quando o token foi criado
18             'exp' => time() + 60*60, //timestamp de quando o token irá expirar
19             'data' => $data, // Dados que serão armazenados no token
20             'institution' => "registroNacionalELA" // Identificação de qual sistema que gerou o token
21         ];
22
23         // Retorna o token JWT com codificação HS256
24         return JWT::encode($payload, $key, 'HS256');
25     }
26 }
```

Fonte: Autoria Própria

4.4 Interagindo com a rede *blockchain*

Como descrito na seção anterior, sempre que um novo paciente tem seu registro cadastrado pelo médico através do sistema Registro Nacional da ELA, os dados são salvos na *blockchain* e para validar se foram salvos corretamente, é possível fazer requisições para as rotas que retornam os dados dos pacientes. Nas subseções a seguir, serão mostrados os resultados retornados por cada uma das funções que realizam a leitura dos dados salvos na rede. Os dados dos pacientes e médicos apresentados são fictícios e gerados apenas para fins de exemplificação.

4.4.1 Retornando todos os pacientes cadastrados na rede

Para retornar as informações de todos os pacientes salvos na *blockchain*, é utilizada a rota “/api/patients” e serão retornados os dados da última atualização realizada para aquele paciente. Veja no exemplo da Figura 4.11 em que são listados dois pacientes diferentes. A última atualização do paciente de nome “Juliana Marina Silva Costa” foi uma ação do tipo “*UPDATE*” e a última atualização para o paciente de nome “João Carlos Maciel” foi uma ação de “*CREATE*”. As respostas do registro foram colapsadas para que fosse possível listar mais de um paciente, mas a Figura 4.12 mostra mais detalhes de como as respostas são retornadas. As informações do médico salvas junto aos dados do paciente, dizem respeito ao médico responsável pelo seu cadastro no sistema.

Figura 4.11: Retorno dos dados dos pacientes cadastrados na rede

```
2 // http://localhost:3000/api/patients
3
4 ▼ [
5 ▼ {
6     "action": "UPDATE",
7     "cpf": "07139666024",
8     "currentCity": "Natal/RN",
9     "dateOfBirth": "20/06/1987",
10    "doctorCRM": "1111",
11    "doctorName": "Marcelo Augusto de Souza",
12    "motherName": "Maria Carla Silva",
13    "name": "Juliana Marina Silva Costa",
14    "phone": "(84) 99676-3521",
15    "race": "preto",
16 ▶    "registry": [↔],
118    "sex": "F"
119  },
120 ▼ {
121    "action": "CREATE",
122    "cpf": "0_1",
123    "currentCity": "São Paulo",
124    "dateOfBirth": "01/01/2000",
125    "doctorCRM": "123456",
126    "doctorName": "Dr. Alice",
127    "motherName": "Mother Name",
128    "name": "João Carlos Maciel",
129    "phone": "123456789",
```

Fonte: Autoria Própria

Figura 4.12: Mais detalhes do retorno dos dados do registro do paciente

16	▼	"registry": [
17	▼	{
18		"question": "Data da inclusão",
19		"answer": "28/11/2024"
20		},
21	▼	{
22		"question": "Data do primeiro sintoma",
23		"answer": "01/01/2024"
24		},
25	▼	{
26		"question": "Idade do paciente no início dos sintomas",
27		"answer": "36 anos"
28		},
29	▼	{
30		"question": "Data do diagnóstico",
31		"answer": "09/09/2024"
32		},
33	▼	{
34		"question": "Idade no momento do diagnóstico",
35		"answer": "37 anos"
36		},
37	▼	{
38		"question": "Município de residência no início da doença",
39		"answer": "Natal"
40		},
41	▼	{
42		"question": "Tempo de moradia nesse município",
43		"answer": "05 Ano(s) e 05 Mes(es)"
44		},
45	▼	{
46		"question": "Cidade onde nasceu",
47		"answer": "Natal"
48		},
49	▼	{
50		"question": "Estado onde nasceu",
51		"answer": "Rio Grande do Norte"

Fonte: Autoria Própria

4.4.2 Retornando dados do paciente pelo CPF

É possível buscar apenas por um paciente passando o CPF através da rota “api/patient/{cpf}”. A Figura 4.13 mostra os dados do paciente com CPF **49399966038**. Esta busca também retorna a última operação que foi realizada para o paciente que, como pode ser

visto na imagem, foi a ação de “*UPDATE*”. Logo, os dados retornados são as informações mais recentes daquele paciente.

Figura 4.13: Retorno de um paciente filtrado pelo CPF

```
// http://localhost:3000/api/patient/49399966038

{
  "action": "CREATE",
  "cpf": "49399966038",
  "currentCity": "Natal/RN",
  "dateOfBirth": "20/06/1985",
  "doctorCRM": "1111",
  "doctorName": "Augusto de Souza",
  "motherName": "Marta Calisto",
  "name": "Nathalia Calisto Rodrigues",
  "phone": "(84) 99876-5246",
  "race": "branco",
  "registry": [
    {
      "question": "Data da inclusão",
      "answer": "02/12/2024"
    },
    {
      "question": "Data do primeiro sintoma",
      "answer": "01/01/2024"
    },
    {
      "question": "Idade do paciente no início dos sintomas",
      "answer": "38 anos"
    },
    {
      "question": "Data do diagnóstico",
      "answer": "09/09/2024"
    },
    {
      "question": "Idade no momento do diagnóstico",
      "answer": "39 anos"
    },
    {
      "question": "Município de residência no início da doença",
      "answer": "Natal"
    }
  ]
}
```

Fonte: Autoria Própria

4.4.3 Retornando histórico de alterações do paciente pelo CPF

Para ter uma visão do histórico de modificações realizadas para um dado paciente, é possível utilizar a rota “api/patienthistory/{cpf}”. Veja um exemplo na Figura 4.14 que mostra um trecho do histórico de ações realizadas para o paciente de CPF **49399966038**. Na Figura 4.14 é possível observar que houve uma operação do tipo “ACCESS” seguida de um “UPDATE”, a ordem das transações é exibida em ordem decrescente, da mais recente para a mais antiga. Para essa primeira versão, não é possível identificar diretamente quais os campos do paciente foram atualizados, pois sempre são salvos todos os campos quando é feita alguma alteração nas respostas.

Figura 4.14: Retorno do histórico de alterações de um paciente pelo CPF

```
// http://localhost:3000/api/patienthistory/49399966038

[
  {
    "txId": "52c3d1e7eb7a63a2bc929d62ec8aceaa30f5051369964f30fb237adb63c2f314",
    "timestamp": {↔},
    "isDelete": "false",
    "value": {
      "action": "UPDATE",
      "cpf": "49399966038",
      "currentCity": "Natal/RN",
      "dateOfBirth": "20/06/1985",
      "doctorCRM": "1111",
      "doctorName": "Augusto de Souza",
      "motherName": "Marta Calisto",
      "name": "Nathalia Calisto Rodrigues",
      "phone": "(84) 99876-5246",
      "race": "branco",
      "registry": [↔],
      "sex": "F"
    }
  },
  {
    "txId": "1b530f7ac09b2bc86b98af6799568bbdcf511e9d90c207058ffec2f6e9d1b65d",
    "timestamp": {↔},
    "isDelete": "false",
    "value": {
      "accessDate": "2024-12-02 13:59:00",
      "action": "ACCESS",
      "cpf": "49399966038",
      "doctorCRM": "1111",
      "doctorName": "Augusto de Souza",
      "name": "Nathalia Calisto Rodrigues"
    }
  }
],
```

Fonte: Autoria Própria

4.5 Análise de performance

Os testes foram configurados para terem uma duração de 60s e com taxa de envio fixa, enviando transações de entrada em um intervalo fixo de 100 transações por segundo. Veja na Figura 4.15 detalhes de como foi montado o arquivo de configuração para a *registerPatient*. Já os testes para a função *getPatientHistory* foram realizados com uma configuração diferente dos demais por demandar um esforço computacional maior e não conseguir concluir a análise de performance retornando erros. Para esta função em específico, foram enviadas apenas 2 transações por segundo, como pode ser visto na Figura 4.16 que dá detalhes dessa configuração.

Figura 4.15: Configuração dos testes utilizando como exemplo o da função *registerPatient*.

```

benchmarks > ! ehrccBenchmark.yaml
1  ∨ test:
2      name: ehrcc-contract-benchmark
3      description: Benchmark for Eletronic Health Record Chaincode
4  ∨ workers:
5      number: 2
6  ∨ rounds:
7  ∨     - label: registerPatient
8        description: Register patient benchmark
9        txDuration: 60
10 ∨     rateControl:
11         type: fixed-rate
12 ∨     opts:
13         tps: 100
14 ∨     workload:
15         module: workload/registerPatient.js
16 ∨     arguments:
17         patients: 10
18         contractId: ehrcc
19         txDetails: true

```

Fonte: Autoria Própria

Figura 4.16: Configuração dos testes para a função `getPatientHistory`.

```

76 - label: getPatientHistory
77   description: Read patient history benchmark
78   txDuration: 60
79   rateControl:
80     type: fixed-rate
81     opts:
82       tps: 2
83   workload:
84     module: workload/getPatientHistory.js
85     arguments:
86       patients: 10
87       contractId: ehrcc
88     txDetails: true
89

```

Fonte: Autoria Própria

Para a análise foi utilizado o relatório gerado pela própria ferramenta Caliper que retorna os valores referentes à taxa de transferência, latência, taxas de erro e outros aspectos que ajudam a avaliar a eficiência e a capacidade da rede *blockchain* em processar transações. Infelizmente, na versão utilizada desta ferramenta, existe a limitação de que não é possível coletar a latência e o *throughput* de cada iteração dos testes de performance. Esses pontos poderiam ser utilizados para a geração de gráficos, cálculo de desvio padrão, mediana e média, o que enriqueceria ainda mais as análises, dando uma visão da existência ou não de valores extremos e discrepantes dentre os que foram coletados. Na Tabela 4.1, foi adicionado o sumário das métricas para cada uma das funções avaliadas.

Tabela 4.1: Sumário das métricas para cada uma das funções avaliadas do contrato inteligente.

Função	Sucesso	Falha	Taxa de envio (TPS)	Latência máxima (s)	Latência mínima (s)	Latência média (s)	Throughput (TPS)
registerPatient	19390		81.1	33.10	0.88	18.29	80.4
getAllPatients	6002		100.0	0.05	0.01	0.01	100.0
registerAccessPatient	17830		88.2	25.16	0.58	13.99	87.5
updatePatient	18160		81.7	23.70	0.07	16.25	81.2
getPatient	6002		100.0	0.03	0.01	0.01	100.0
getPatientHistory**	122		2.0	41.24	1.37	24.60	1.3

**A função *getPatientHistory* faz uso de uma configuração diferente das demais funções para análise das métricas.

Fonte: Autoria Própria

A coluna “*Sucesso*” indica o total de sucesso das requisições realizadas, isto é, o total de transações concluídas sem nenhuma falha. Já a coluna “*Falha*” indica o total de falhas, que como pode ser visto na Tabela 4.1, não ocorreram falhas durante a execução dos testes. A tabela mostra a latência mínima, máxima e média em segundos, e por fim, o *throughput* dado em transações por segundo (TPS).

Para a análise das outras métricas, as funções foram agrupadas pelo tipo de operação que elas realizam no *ledger*, podendo ser uma operação de leitura ou escrita. Os critérios de avaliação são bem similares, por isso a escolha de agrupá-las.

4.5.1 Operações de escrita na rede

As funções que realizam escrita na *blockchain* são: *registerPatient*, *registerAccessPatient* e *updatePatient*. Através dos resultados da tabela, é possível notar que

para as operações de escrita, o *throughput* não alcança o valor da taxa de envio, ou seja, as transações enviadas não são confirmadas imediatamente pela rede, apresentando um atraso. Apesar disso, um *throughput* dentro da faixa de 50s a 150s apresenta um desempenho razoável para a rede de teste do Hyperledger Fabric, considerando que existem limitações da configuração padrão. Esses valores de referência foram definidos com base nas análises feitas por Diegues (2020) em seu estudo.

Tomando como base a latência média, houve uma grande variação entre as operações de leitura, com menos de 1 segundo, e as operações de escrita, que chegaram a mais de 24 segundos. O principal motivo visto para as latências mais elevadas são limites de *hardware* do ambiente de execução dos testes, que mesmo tendo uma configuração boa para ambientes de desenvolvimento, já estava utilizando mais de 50% da memória RAM com a execução do Docker e dos agentes relacionados à execução da rede *blockchain* em si.

4.5.2 Operações de leitura na rede

As operações de leitura costumam ser mais leves quando comparadas a uma operação de escrita, pois as operações de leitura na rede são quase que imediatas, enquanto as operações de escrita não (Diegues, 2020, p. 118). As funções que realizam leitura são: *getAllPatients*, *getPatient* e *getPatientHistory*.

A *getPatientHistory* obteve um resultado diferente das outras funções de escrita e apresentou uma performance bem ruim com taxas bastantes altas de latência. Os principais fatores podem estar relacionados ao código atual não aplicar paginação e um limite de registros retornados por páginas. A busca pelos históricos é uma operação que tende a ser mais lenta devido a sua extensão.

Contudo, como esperado, as outras operações de leitura, que tendem a ser mais rápidas, apresentaram melhores valores de *throughput*. Diferentemente das operações de escrita, apresentaram uma latência média na faixa de 10 milissegundos, sendo um bom resultado.

Valores de latência média mais baixos indicam que a rede demora menos tempo a responder aos pedidos de transações.

Capítulo 5

Discussões

A tecnologia *blockchain* proporciona uma maior segurança e integridade dos dados de saúde dos pacientes com ELA ao fornecer um sistema descentralizado e criptograficamente seguro para o armazenamento e compartilhamento de informações. Por meio de sua arquitetura imutável, os registros são protegidos contra alterações não autorizadas e fraudes, o que assegura a integridade e a confiabilidade dos dados dos pacientes. Além disso, os mecanismos de consenso da *blockchain* garantem que todas as transações de dados sejam validadas antes de serem incorporadas ao sistema, isso cria um histórico auditável que pode ser rastreado de maneira transparente por todas as partes autorizadas.

Além da segurança, a *blockchain* mantém a interoperabilidade ao permitir que diferentes sistemas de saúde e organizações compartilhem dados de maneira padronizada, devido ao uso dos contratos inteligentes, e segura, o que reduz os sistemas isolados de informação, como hospitais ou clínicas que armazenam dados fora da rede, e assim possibilita um cuidado mais integrado e eficiente. No contexto da Plataforma RevELA, essa capacidade facilita a colaboração entre pesquisadores, médicos e instituições, e acelera a análise de dados e o desenvolvimento de tratamentos personalizados para pacientes com ELA. Assim, a combinação de segurança e interoperabilidade promove um ecossistema de saúde mais efetivo, pois contribui para melhorar diretamente a qualidade do atendimento e o avanço da pesquisa clínica.

Apesar das vantagens promissoras, o uso da tecnologia *blockchain* em sistemas de saúde enfrenta dificuldades durante a sua fase de implementação como foi apontado em Fonsêca et al. 2024, a *blockchain* frequentemente apresenta problemas relacionados a escalabilidade, isso

resulta em tempos de transação mais longos e atraso no processamento dos dados gerados. As operações realizadas para geração e validação das chaves criptográficas assim como as validações das transações por meio do mecanismo de consenso, requerem um grande esforço computacional, portanto este é um fator de alerta para a expansão da rede em aplicações que necessitam crescer cada vez mais. Ainda relacionado às taxas de transação, existe uma limitação quanto às transferências por segundo, o que pode resultar em um atraso e congestionamento da rede em momentos de alta demanda da rede (Fonseca *et al.*, 2024).

Essas mesmas dificuldades foram observadas durante os testes de performance aplicados ao trabalho, que mostraram que apesar da rede funcionar bem, quando submetida a cargas de consumo e envio de dados mais pesadas, apresentou um desempenho não tão satisfatório quanto o esperado, com taxa de transferência boas, mas com uma latência alta para operações de escrita, quando comparadas à latência de operações de leitura. A latência média esperada deveria ser de menos de 1s, mas estava com mais de 13s. De certa forma, é compreensível resultados não tão satisfatórios visto que os testes foram realizados em ambiente local com limitações de hardware, como CPU e memória. Além disso, foi utilizada a rede *test-network* do *framework Hyperledger Fabric*, que possui uma configuração básica (dois *peers* e um *ordering service* simples) que não considera otimização para altas performances.

Capítulo 6

Considerações Finais

A integração do Registro Nacional da ELA com a tecnologia de *blockchain* representa uma proposta inovadora e altamente promissora para enfrentar os desafios no gerenciamento de informações de pacientes com a ELA. Diante da necessidade de um registro eletrônico robusto, essa abordagem oferece uma solução abrangente e que representa apenas um primeiro passo para a integração da *blockchain* em outros projetos da Plataforma revELA.

O uso da *blockchain* permite que informações críticas sejam compartilhadas com segurança entre diversas partes interessadas na saúde do paciente. Além disso, a imutabilidade dos registros na *blockchain* garante a integridade das informações ao longo do tempo, o que é de extrema importância no contexto da ELA, uma doença progressiva que requer um histórico preciso do tratamento.

Essa interoperabilidade proporciona uma camada adicional de segurança aos dados dos pacientes, protegendo contra violações de privacidade e garantindo que apenas as partes autorizadas tenham acesso aos registros. A integração entre o Registro Nacional da ELA e a tecnologia de *blockchain* é um avanço significativo no aprimoramento dos cuidados e no suporte aos pacientes com ELA. Essa combinação de recursos tem o potencial de revolucionar o cenário da ELA, beneficiando pacientes, profissionais de saúde e pesquisadores. De forma que, os pacientes se beneficiarão com diagnósticos mais precisos, tratamentos personalizados e descobertas inovadoras, melhorando a qualidade de vida e promovendo um cuidado mais integrado.

Os resultados apresentados validam a hipótese norteadora deste trabalho, no qual, a integração entre a tecnologia *blockchain* e o Registro Nacional da ELA representa uma melhoria na segurança e integridade dos dados de pacientes com ELA, assim como, respondem às questões de pesquisa. A Q1 é respondida com as análises feitas através da Revisão Sistemática da Literatura apresentada no tópico 2.6. A Q2 é respondida através deste trabalho que mostra que a *blockchain* é a ferramenta indicada para a gestão mais eficaz e segura de seus dados de saúde.

6.1 Principais Contribuições e Impactos sociais

A solução de saúde digital desenvolvida é apenas o início para a interoperabilidade, segurança e auditabilidade de dados dos sistemas que fazem parte da Plataforma revELA. O foco deste trabalho foi apresentar a viabilidade do uso de uma rede *blockchain* integrada ao Registro Nacional da Esclerose Lateral Amiotrófica (ELA). Todavia, o uso de *Application Programming Interface* (API) e dos contratos inteligentes desenvolvidos nesta dissertação de mestrado não se limitam apenas a este estudo. Outros projetos como o Prontuário Eletrônico para Paciente com ELA e o Observatório Nacional Brasileiro da ELA, que registram e consomem dados de pacientes que vivem com a Esclerose Lateral Amiotrófica, também podem se beneficiar com todas as vantagens proporcionadas pela tecnologia *blockchain*.

A solução proposta pode servir, inclusive, como um modelo para a integração com a Rede Nacional de Dados em Saúde (RNDS) do Ministério da Saúde do Brasil. Logo, se destaca como uma camada de segurança e proteção de dados disponibilizados em conformidade com a Lei Geral de Proteção de Dados (LGPD) do Brasil. De forma que, além de impactar diretamente todos os pacientes que vivem com ELA distribuídos em todo o território nacional, também tem o potencial de escalar e impactar todos os usuários do Sistema Único de Saúde (SUS) do Brasil.

A *blockchain* está em constante evolução e por isso é utilizada em diversas aplicações em todo o mundo, ela tem ganhado força no contexto da saúde global, particularmente depois da pandemia de covid-19. Este trabalho apresenta uma grande oportunidade de utilizar o que há de mais robusto em termos de novas tecnologias e segurança, que permeia os sistemas de informação em saúde, os quais fazem uso intensivo de dados sensíveis de usuários do SUS e de vários serviços de saúde. Portanto, comumente esses sistemas têm que lidar com fatores como segurança, confidencialidade e interoperabilidade.

Neste contexto, este trabalho de mestrado, ao contribuir com o Registro Nacional da Esclerose Lateral Amiotrófica (ELA), uma tecnologia que faz parte do ecossistema de saúde digital do Projeto revELA, contribui diretamente com Objetivos de Desenvolvimento Sustentável (ODS) da Agenda 2030 da Organização das Nações Unidas (ONU), principalmente: 3 Saúde e bem Estar, 9 Indústria, Inovação e Infraestrutura e 16 Paz, Justiça e Instituições Fortes.

6.2 Trajetória acadêmica

Meu contato junto aos projetos relacionados com a Esclerose Lateral Amiotrófica se iniciou em 2021 quando ingressei como pesquisadora no projeto “Desenvolvimento Científico e Tecnológico Aplicado a Esclerose Lateral Amiotrófica” pelo Laboratório de Inovação Tecnológica em Saúde (LAIS), este que foi o projeto principal responsável por desenvolver as soluções da Plataforma revELA. Entre os anos de 2021 e 2023, atuei principalmente no Registro Nacional da ELA, no Observatório Nacional da ELA e no Prontuário Eletrônico da ELA.

Em 2023, fiz minha primeira publicação como uma das autoras principais do artigo com título “*Digital health solution for monitoring and surveillance of Amyotrophic Lateral Sclerosis in Brazil*”¹³, publicado na revista internacional *Frontiers in Public Health*. Ainda em 2023, foi

¹³ <https://www.frontiersin.org/journals/public-health/articles/10.3389/fpubh.2023.1209633/full>

feito o registro de software dos seguintes produtos dos quais colaborei: Registro Nacional da ELA, Observatório Nacional da ELA, Autonomus e Autonomus Mobile.

Em 2024, publiquei meu primeiro artigo de Revisão Sistemática da Literatura com título “*Blockchain in Health Information Systems: A Systematic Review*”¹⁴, publicado na revista *International Journal of Environmental Research and Public Health (IJERPH)*. Além disso, tive um trabalho aceito como anal no XX Congresso Brasileiro de Informática em Saúde (CBIS) 2024 com o título “*Electronic health record for patients with amyotrophic lateral sclerosis: a Blockchain approach*”¹⁵.

6.3 Limitações e Desafios

Entender o funcionamento e como operar uma rede *blockchain* foi um desafio bem considerável e que demandou bastante tempo, já que boa parte do desenvolvimento do trabalho foi dedicado à realização de cursos e busca de materiais que abordassem os conceitos e aplicações práticas da *blockchain*. A maioria dos materiais disponíveis que abordam o assunto de maneira aprofundada estão em inglês e usam termos muito técnicos que dificultam o entendimento por aqueles que são novos nessa área.

A solução proposta foi desenvolvida e testada apenas em ambiente local, sem validação em ambientes de homologação ou produção. Essa limitação decorre do fato de que a implementação de uma *blockchain* requer uma infraestrutura mais robusta e escalável, ou seja, sistemas computacionais com maior capacidade de processamento, armazenamento e conectividade para lidar com a execução de contratos inteligentes, gerenciamento de múltiplos nós e sincronização em tempo real dos dados distribuídos na rede. Atualmente, esses requisitos excedem os recursos utilizados no Projeto revELA, o que inviabilizou a realização de testes em

¹⁴ <https://www.mdpi.com/1660-4601/21/11/1512>

¹⁵ <https://zenodo.org/records/14337625>

cenários reais ou mais complexos, limitando, assim, a avaliação de seu desempenho e integração.

6.4 Trabalhos Futuros

Ainda existe etapas de melhorias que podem ser desenvolvidas, quais sejam:

1. Desenvolver uma interface para acessar os dados dos pacientes facilitando o acompanhamento e auditoria das informações com uma visualização mais amigável, não sendo disponível apenas através de rotas HTTP requisitadas via navegador.
2. Melhorar a performance da rede desenvolvendo uma versão mais simplificada utilizando apenas os artefatos necessários para o seu funcionamento.
3. Disponibilizar a estrutura *blockchain* criada em ambiente de homologação para validar seu funcionamento e performance em servidores com infraestrutura mais adequada.

Referências

AGBO, Cornelius C.; MAHMOUD, Qusay H.; EKLUND, J. Mikael. Blockchain Technology in Healthcare: A Systematic Review. **Healthcare**, v. 7, p. 56-56, 2019. DOI <https://doi.org/10.3390/healthcare7020056>

ANTWI, McSeth; ADNANE, Asma; AHMAD, Farhan; HUSSAIN, Rasheed; REHMAN, Muhammad Habib ur; KERRACHE, Chaker Abdelaziz. The case of HyperLedger Fabric as a blockchain solution for healthcare applications. **Blockchain: Research and Applications**, v. 2, p. 100012-100012, 2021. DOI <https://doi.org/10.1016/j.bcr.2021.100012>.

ARTHUR, Karissa C.; CALVO, Andrea; PRICE, T. Ryan; Geiger, JOSHUA, T.; CHIÒ, Adriano; TRAYNOR, Bryan J. Projected increase in amyotrophic lateral sclerosis from 2015 to 2040. **Nature communications**, v. 7, p. 12408–12408, 2016. PMC4987527[pmcid]. DOI <https://doi.org/10.1038/ncomms12408>

BACH, L. M.; MIHALJEVIC, B.; ZAGAR, M. Comparative analysis of blockchain consensus algorithms. In: INTERNATIONAL CONVENTION ON INFORMATION AND COMMUNICATION TECHNOLOGY, ELECTRONICS AND MICROELECTRONICS (MIPRO), 41., 2018, Croatia. **Proceedings [...]** Croatia, 2018. p. 1545–1550.

BARBALHO, Ingridy; FERNANDES, Felipe. Electronic health record in brazil: prospects and technological challenges. **PROSPERO**, 2021. CRD42021233219.

BARBALHO, Ingridy M. P.; FERNANDES, Felipe; BARROS, Daniele M. S.; PAIVA, Jailton C.; HENRIQUES, Jorge; MORAIS, Antonio H. F.; COUTINHO, Karilany D.; COELHO NETO, Gilate, C.; CHIORO, Arthur; VALENTIM, Ricardo A. M. ‘Electronic health records in brazil: Prospects and technological challenges. **Frontiers in Public Health**, v. 10, 2022.

BARBALHO, Ingridy Marina Pierre. **Plataforma RevELA: Uma solução de saúde digital para a gestão do cuidado e vigilância dos pacientes com Esclerose Lateral Amiotrófica no Brasil**. 2023. 146f. Tese (Doutorado na área de Engenharia de Computação) – Programa de Pós-Graduação em Engenharia Elétrica e de Computação, Universidade Federal do Rio Grande do Norte, Natal, 2023.

BARBALHO, Ingridy; FONSÊCA, Aleika; FERNANDES, Felipe; HENRIQUES, Jorge; GIL, Paulo; NAGEM, Danilo; LINDQUIST, Raquel; LIMA, Thaisa; SANTOS, João Paulo Queiroz dos; PAIVA, Jailton; MORAIS, Antonio H. F.; DOURADO JÚNIOR, Mário E. T.; VALENTIM, Ricardo A. M. ‘Digital health solution for monitoring and surveillance of Amyotrophic Lateral Sclerosis in Brazil’. **Front Public Health**, v. 11, 24 ago. 2023. DOI <https://doi.org/10.3389/fpubh.2023.1209633>.

BARI, Nadi; QAMAR, Usman; KHALID, Ayesha. Efficient Contact Tracing for pandemics

using blockchain. **Informatics in Medicine Unlocked**, v. 26, p. 100742, 2021. DOI <https://doi.org/10.1016/j.imu.2021.100742>.

CASTRO-COSTA, Carlos M. De; ORIÁ, Reinaldo B.; MACHADO-FILHO, João A.; FRANCO, Maria T. G.; DINIZ, Débora L. O.; GIFFONI, Sílvia D.; SANTOS, Terezinha J. T.; CUNHA, Francisco M. B. Da; BRUIN, Veralice S. De; TEIXEIRA, Carlos A. C. Amyotrophic lateral sclerosis: clinical analysis of 78 cases from fortaleza (northeastern brazil). **Arquivos de Neuro-Psiquiatria**, v. 57, p. 761–774, 1999.

CHAUDHRY, Natalia; YOUSAF, Muhammad Murtaza (2018), Consensus algorithms in blockchain: Comparative analysis, challenges and opportunities. *In: INTERNATIONAL CONFERENCE ON OPEN SOURCE SYSTEMS AND TECHNOLOGIES (ICOSST)*, 12., Pakistan, 2018. **Proceedings [...]** Pakistan, 2018. p. 54–63.

COOK, Suzanne F. *et al.* A Descriptive Review of Global Real World Evidence Efforts to Advance Drug Discovery and Clinical Development in Amyotrophic Lateral Sclerosis. **Frontiers in neurology**, v. 12, p. 770001, 2021. DOI doi:10.3389/fneur.2021.770001

CUNNINGHAM, J.; DAVIDGE, G.; DAVIES, N.; DEVANEY, S.; HOLM, S.; HARDING, M.; LEEMING, G.; NEUMANN, V.; AINSWORTH, J. Blockchain Native Data Linkage. **Front. Blockchain**, v. 4, p. 667388, 2021. DOI 10.3389/fbloc.2021.667388

DAGHER, Gaby G.; MOHLER, Jordan; MILOJKOVIC, Matea; MARELLA, Praneeth Babu. Ancile: Privacy-preserving framework for access control and interoperability of electronic health records using blockchain technology. **Sustainable Cities and Society**, v. 39, p. 283–297, 2018. DOI <https://doi.org/10.1016/j.scs.2018.02.014>.

DIEGUES, Ana Catarina Dinis. **Benchmarking de Plataformas Blockchain**. 2020. 144f. Dissertação (Mestrado em Engenharia Informática) - Faculdade de Ciências e Tecnologia, Departamento de Engenharia, Universidade de Coimbra, Coimbra, 2020.

DINH, T. T. A.; WANG, J.; CHEN, G. LIU, R.; OOI, B. C.; TAN, K.-L. (2017). BLOCKBENCH: a framework for analyzing private blockchains. *In: INTERNATIONAL CONFERENCE ON MANAGEMENT OF DATA*, 2017, Chicago. **Proceedings [...]** Chicago: ACM, 2017. p. 1085–100. DOI 10.1145/3035918.3064033

FERNANDES, Felipe; BARBALHO, Ingridy; BARROS, Daniele; VALENTIM, Ricardo; TEIXEIRA, César; HENRIQUES, Jorge; GIL, Paulo; DOURADO JÚNIOR, Mário. Biomedical signals and machine learning in amyotrophic lateral sclerosis: a systematic review. **Biomedical engineering online**, v. 20, n. 1, p. 61–61, 2021.

FONSECA, Aleika L. A.; BARBALHO, Ingridy Marina Pierre; FERNANDES, Felipe; ARRAIS JÚNIOR, Ernando; NAGEM, Danilo Alves Pinto; CARDOSO, Pablo Holanda; VERAS, Nicolás Vinícius Rodrigues; FARIAS, Fernando Lucas de Oliveira; LINDQUIST, Ana Raquel; SANTOS, João Paulo Q. dos. Blockchain in Health Information Systems: A Systematic Review. **International Journal of Environmental Research and Public Health**, v. 21, n. 11, p. 1512, 2024. DOI <https://doi.org/10.3390/ijerph21111512>

GAUR, Nitin *et al.* **Hands-on Blockchain with Hyperledger**: Building decentralized applications with Hyperledger Fabric and Composer. Birmingham: Packt Publishing, 2018.

460 p.

GRANIT, Volkan; GRIGNON, Anne-Laure; WUU, Joanne; KATZ, Jonathan; WALK, David; HUSSAIN, Sumaira; HERNANDEZ, Jessica; JACKSON, Carlayne; CARESS, James; YOSICK BS, Tom; SMIDER, Nancy; BENATAR, Michael. Harnessing the power of the electronic health record for ALS research and quality improvement: CReATe CAPTURE-ALS and the ALS Toolkit. **Muscle & Nerve**, v. 65, p. 154-161, 2022. DOI <https://doi-org.ez18.periodicos.capes.gov.br/10.1002/mus.27454>.

HOGDEN, Anne; FOLEY, Geraldine; HENDERSON, Robert D.; JAMES, Natalie; AOUN, Samar M. Amyotrophic lateral sclerosis: improving care with a multidisciplinary approach. **Journal of multidisciplinary healthcare**, v. 10, p. 205–215, 2017. Disponível em: <https://pubmed.ncbi.nlm.nih.gov/28579792>. Acesso em: 20 out. 2024.

HULISZ, Darrell. Amyotrophic lateral sclerosis: disease state overview. **American Journal of Managed Care**, v. 24, p. 320-326, 2018.

HUSSIEN, H. M.; YASIN, S. M.; UDZIR, S. N. I.; ZAIDAN, A. A.; ZAIDAN, B. B. A Systematic Review for Enabling of Develop a Blockchain Technology in Healthcare Application: Taxonomy, Substantially Analysis, Motivations, Challenges, Recommendations and Future Direction. **Journal of medical systems**, v. 43, n. 10, p. 320-320, 2019. DOI <https://doi.org/10.1007/s10916-019-1445-8>

HYPERLEDGER Blockchain Performance Metrics. 2018. Available at: <https://www.lfdecentralizedtrust.org/learn/publications/blockchain-performance-metrics>. Access in: 20 nov. 2024.

KHEZR, S.; MONIRUZZAMAN, M.; YASSINE, A.; BENLAMRI, R. Blockchain Technology in Healthcare: A Comprehensive Review and Directions for Future Research. **Appl. Sci.**, v. 9, p. 1736, 2019. DOI <https://doi.org/10.3390/app9091736>

KIERNAN, Matthew C.; VUCIC, Steve; CHEAH, Benjamin C.; TURNER, Martin R.; EISEN, Andrew; HARDIMAN, Orla; BURRELL, James R.; ZOING, Margaret C. Amyotrophic lateral sclerosis. **The Lancet**, v. 377, n. 9769, p. 942–955, 2011. DOI [https://doi.org/10.1016/S0140-6736\(10\)61156-7](https://doi.org/10.1016/S0140-6736(10)61156-7)

KITCHENHAM, Barbara. **Procedures for performing systematic reviews**. Keele, Staffs: Keele University, Department of Computer Science, Software Engineering Group and Empirical Software Engineering National ICT Australia Ltd., ST5 5BG, 2004. Relatório técnico.

LINDEN JUNIOR, Eduardo; BECKER, Jefferson; SCHESTATSKY, Pedro; ROTTA, Francisco Tellechea; MARRONE, Carlo Domenico; GOMES, Irenio. Prevalence of amyotrophic lateral sclerosis in the city of porto alegre, in southern brazil. **Arquivos de Neuro- Psiquiatria**, v. 71, p. 959–962, 2013.

LONGINETTI, Elisa; FANG, Fang. Epidemiology of amyotrophic lateral sclerosis: an update of recent literature. **Current opinion in neurology**, v. 32, p. 771-776, 2019. DOI [doi:10.1097/WCO.0000000000000730](https://doi.org/10.1097/WCO.0000000000000730)

LOUREIRO, Marli P. S.; GRESS, Claudio H.; THULER, Luiz C. S.; ALVARENGA, Regina M. P.; LIMA, José M. B. Clinical aspects of amyotrophic lateral sclerosis in rio de janeiro/brazil. **Journal of the Neurological Sciences**, v. 316, p. 61–66, 2012.

MANTEY, Eric Appiah; ZHOU, Conghua; ANAJEMBA, Joseph Henry; OKPALAOGUCHI, Izuchukwu M.; CHIADIKI, Onyeachonam Dominic-Mario. Blockchain-Secured Recommender System for Special Need Patients Using Deep Learning. **Front. Public Health**, v. 9, 2021. DOI <https://doi.org/10.3389/fpubh.2021.737269>

MAO, Zhijuan; ZHANG, Suming; CHEN, Hong. Stem cell therapy for amyotrophic lateral sclerosis. **Cell Regeneration**, v. 4, p. 1–4, 2015.

MARCONDES, Carlos Henrique. Interoperabilidade entre acervos digitais de arquivos, bibliotecas e museus: potencialidades das tecnologias de dados abertos interligados. **Perspectivas em Ciência da Informação**, v. 21, n. 2, p. 61–83, 2016.

NAKAMOTO, Satoshi. **Bitcoin: A Peer-to-Peer Electronic Cash System**. 2008. Disponível em: <https://bitcoin.org/bitcoin.pdf>. Acesso em: 2 jul. 2024.

PAGE, M. J.; MCKENZIE, J. E.; BOSSUYT, P. M.; BOUTRON, I.; HOFFMANN, T. C.; MULROW, C. D.; SHAMSEER, L.; TETZLAFF, J. M.; AKL, E. A.; BRENNAN, S. E. *et al.* The PRISMA 2020 statement: An updated guideline for reporting systematic reviews. **BMJ**, v. 372, n. 71, 2021.

PAGNINI, Francesco *et al.* Pain in Amyotrophic Lateral Sclerosis: a psychological perspective. **Neurological sciences: official journal of the Italian Neurological Society and of the Italian Society of Clinical Neurophysiology** v. 33, p. 1193-1196, 2012. DOI [doi:10.1007/s10072-011-0888-6](https://doi.org/10.1007/s10072-011-0888-6)

PONGNUMKUL, S.; SIRIPANPORNCHANA, C.; THAJCHAYAPONG, S. Performance Analysis of Private Blockchain Platforms in Varying Workloads. *In: INTERNATIONAL CONFERENCE ON COMPUTER COMMUNICATION AND NETWORKS (ICCCN)*, 26., 2017, Vancouver. **Proceedings [...]** Vancouver, 2017. p. 1–6. Available at: [10.1109/ICCCN.2017.8038517](https://doi.org/10.1109/ICCCN.2017.8038517). Access in: 20 fev. 2024.

POURNAGHI, Seyed Morteza; BAYAT, Majid; FARJAMI, Yaghoub. MedSBA: a novel and secure scheme to share medical data based on blockchain technology and attribute-based encryption. **Journal of Ambient Intelligence and Humanized Computing**, v. 11, p. 4613–4641, 2020. Disponível em: <https://link.springer.com/article/10.1007/s12652-020-01710-y>. Acesso em: 20 jul. 2023.

RADUNOVIĆ, Aleksandar; MITSUMOTO, Hiroshi; LEIGH, P. Nigel. Clinical care of patients with amyotrophic lateral sclerosis. **The Lancet Neurology**, v. 6, p. 913-925, 2007. DOI [https://doi.org/10.1016/S1474-4422\(07\)70244-2](https://doi.org/10.1016/S1474-4422(07)70244-2).

RALLI, Massimo; LAMBIASE, Alessandro; ARTIC, Marco; VINCENTIIS, Marco de; GRECO, Antonio. Amyotrophic Lateral Sclerosis: Autoimmune Pathogenic Mechanisms, Clinical Features, and Therapeutic Perspectives. **The Israel Medical Association Journal**, v. 21, p. 438-443, 2019.

SAEED, Huma; MALIK, Hassaan; BASHIR, Umair; AHMAD, Aiesha; RIAZ, Shafia; ILYAS, Maheen; BUKHARI, Wajahat Anwaar; KHAN, Muhammad Imran Ali. Blockchain technology in healthcare: A systematic review. **PLOS ONE**, v. 17, n. 4, 2022. DOI <https://doi.org/10.1371/journal.pone.0266462>

SHAHNAZ, Ayesha, QAMAR, Usman; KHALID, Ayesha. Using Blockchain for Electronic Health Records. **IEEE Access**, v. 7, p. 147782-147795, 2019. Disponível em: <https://ieeexplore.ieee.org/document/8863359>. Acesso em: 5 fev. 2024.

SHARMA, Pratima; BORAH, Malaya Dutta; NAMASUDRA, Suyel. Improving security of medical big data by using Blockchain technology. **Computers & Electrical Engineering**, v. 96, p. 107529, 2021. DOI <https://doi.org/10.1016/j.compeleceng.2021.107529>.

SHI, Shuyun; HE, Debiao; LI, Li; KUMAR, Neeraj; KHAN, Muhammad Khurram; CHOO, Kim-Kwang Raymond. Applications of blockchain in ensuring the security and privacy of electronic health record systems: A survey. **Computers & Security**, v. 97, p. 101966-101966, 2020. DOI <https://doi.org/10.1016/j.cose.2020.101966>

TANWAR, S.; PAREKH, K.; EVANS, R. Blockchain-based electronic healthcare record system for healthcare 4.0 applications. **J. Inf. Secur. Appl.**, v. 50, p. 102407, 2020. DOI <https://doi.org/10.1590/1981-5344/2735>.

WEI, Wei-Qi; DENNY, Joshua C. Extracting research-quality phenotypes from electronic health records to support precision medicine. **Genome Medicine**, v. 7, p. 41-41, 2015. DOI <https://doi.org/10.1186/s13073-015-0166-y>.

WIJESEKERA, L. C.; LEIGH, P. Nigel. Amyotrophic lateral sclerosis. **Orphanet journal of rare diseases**, v. 4, p. 1-22, 2009. DOI <https://doi.org/10.1186/1750-1172-4-3>