Proposta de modelo para votação eletrônica utilizando Blockchain e Contratos Inteligentes

Model proposal for electronic voting using Blockchain and Smart Contracts

José Alves de Lima Neto [*1] **e Rafael Oliveira Vasconcelos** [†1]

[1] *Universidade Federal de Sergipe, DComp, São Cristóvão, SE, Brasil.*

Resumo

As eleições para escolha de representantes políticos são indiscutivelmente uma das mais importantes manifestações da democracia de uma nação e, por isso, também é um momento crítico e de intensa responsabilidade para aqueles que a operam. Por sua importância, é necessário que atenda vários princípios de segurança, integridade e transparência, os quais foram recentemente contestados no contexto das eleições brasileiras. Diante disto e da boa combinação das redes *blockchain* como agente benéfico para reforçar esses princípios, aplicando o método de pesquisa exploratória, este trabalho apresenta um modelo que aplica blockchain nas eleições brasileiras. Por fim, foram realizados testes funcionais e não funcionais. Os testes funcionais validaram a integridade e segurança do sistema ao permitir a submissão e verificação de votos de forma descentralizada, enquanto os testes não funcionais mostraram o possível custo da adoção deste tecnologia na eleição brasileira. Esses resultados reforçam o potencial da tecnologia blockchain para aprimorar o processo eleitoral brasileiro.

Palavras-chave: Eleições brasileiras. Votação eletrônica. Urna eletrônica. Blockchain. Contrato inteligente.

Abstract

The election to choose political representatives is undoubtedly one of the most important manifestations of a nation's democracy and, therefore, it is also a critical moment and of intense responsibility for those who operate it. Due to its importance, it is necessary to comply with several principles of security, integrity and transparency, which were recently challenged in the context of Brazilian elections. Faced with this challenge and recognizing the favorable combination of blockchain networks as a beneficial agent to reinforce these principles, applying the exploratory research method, this work presents a model that applies blockchain in Brazilian elections. Finally, functional and non-functional tests were carried out. Functional tests validated the system's integrity and security by allowing decentralized submission and verification of votes, while non-functional tests revealed the potential costs of adopting this technology in Brazilian elections. These results highlight the potential of blockchain technology to enhance the Brazilian electoral process.

Keywords: Brazilian elections. Electronic voting. Electronic ballot box. Blockchain. Smart contract.

Textolivre
Linguagem e Tecnologia

DOI: 10.1590/1983-3652.2025.50815

Seção:
Artigos

Autor Correspondente:
Rafael Oliveira Vasconcelos

Editor de seção:
Daniervelin Pereira
Editor de layout:
Leonardo Araújo

Recebido em:
25 de janeiro de 2024
Aceito em:
16 de outubro de 2024
Publicado em:
17 de dezembro de 2024

1 Introdução

Com o avanço das tecnologias, é natural que elas sejam inseridas cada vez mais em diversos cenários com o intuito de proporcionar melhorias, comodidade e segurança. Com o sistema eleitoral brasileiro não foi diferente. Com o principal intuito de trazer transparência e agilidade na apuração das eleições brasileiras, o sistema eleitoral brasileiro fez o uso de diversos recursos para gerir os processos de eleições federais, estaduais e municipais, que culminaram no uso das urnas eletrônicas (Sepúlvida; Paiva, 2019). Entretanto, existe uma recorrência acerca da confiabilidade no processo eleitoral (Machado Filho, 2021).

Algumas iniciativas buscam trazer mais integridade e mais confiabilidade a estes processos críticos. Por isso, ao surgir novas tecnologias que possam agregar novos recursos para alcançar este objetivo começaram a ser estudadas e aplicadas. Uma dessas tecnologias parte do conceito de redes descentralizadas conhecidas como *blockchain*. Esta tecnologia surgiu como livro-razão e contratos

*Email: jalneto@dcomp.ufs.br
†Email: rafael@dcomp.ufs.br

inteligentes públicos para armazenamento e validação de transações. Por este fato, os dados eram praticamente imutáveis e ao mesmo tempo acessíveis para todos os membros da rede descentralizada.

O uso de redes *blockchain* pode potencialmente contribuir para mitigar possíveis vulnerabilidades e aumentar a confiabilidade nos processos eleitorais (Hajian Berenjestanaki *et al.*, 2024; Vladucu *et al.*, 2023; Benabdallah *et al.*, 2022; Varaprasada Rao; Panda, 2023; Tanwar *et al.*, 2024). Nos Estados Unidos, embora a votação eletrônica não seja utilizada como alternativa única de votação nas eleições presidenciais, o Condado de King foi o primeiro região a permitir que os eleitores participassem da eleição do Conselho de Supervisores através de seus *smartphones* em 2020. Em 2018, outras regiões já haviam permitido o voto a distância para deficientes e cidadãos residentes no exterior. Em ambos os casos a votação utiliza a tecnologia *blockchain* para registro dos votos (Fornasier, 2022).

Dessa forma, este trabalho almeja contribuir com a discussão a cerca da confiabilidade dos meios eleitorais brasileiros, propondo um sistema de votação eletrônica que utiliza *blockchain* para armazenamento e contagem de votos, com o intuito de melhor a auditabilidade e integridade, respeitanto os princípios democráticos que uma eleição demanda.

1.1 Objetivos

Propor um modelo de votação eletrônica com base de dados descentralizada em redes *blockchain*, a fim de melhor a segurança, integridade e transparência do processo eleitoral brasileiro.

Para alcançar o objetivo geral, os seguintes objetivos específicos foram definidos:

- Apresentar as necessidades e vulnerabilidades do sistema atual de votação brasileiro, assim como os motivos que contribuem para falta de confiabilidade no processo por parte da população;
- Projetar um modelo de votação eletrônica baseado em *blockchain* focado em transparência, integridade e segurança, que poderia ser aplicado numa eleição oficial brasileira, destacando regras, especificando estrutura e comparando com o sistema atual;
- Realizar testes funcionais e não funcionais da implementação do modelo.

Ao alcançar esses objetivos específicos, espera-se contribuir para o avanço da área de votação eletrônica utilizando tecnologias de *blockchain*, e com a discussão recente acerca das UE brasileiras, promovendo maior transparência, segurança e confiança nos processos democráticos de eleição.

1.2 Método

Para o desenvolvimento deste trabalho foi realizada uma pesquisa exploratória sobre os temas "E--Voting", "Votação no Brasil", "Redes *blockchain*" e "Votação Eletrônica utilizando *blockchain*". Além disso, foi feita a pesquisa exploratória qualitativa dos serviços de votação no Brasil, as UEs e o sistema antigo baseado em cédula, além de outros serviços eletrônios de votação existentes no mundo. Após isso, foi proposto um modelo de votação eletrônica utilizando redes *blockchain*, a partir da definição de seus requisitos e domínios; será desenvolvida um contrato inteligente baseado no modelo; serão realizados testes das funcionalidade propostas e atendimento dos requisitos objetivados.

1.3 Trabalhos Relacionados

Sobre as eleições no Brasil, Ferrão *et al.* (2019) fez um estudo sobre as urnas eletrônicas e organizou em seu trabalho o histórico, evolução e falhas e desafios das UEs utilizadas no país. Similarmente, Machado Filho (2021) fez um trabalho de análise dos mecanismos de segurança das UEs, reuniu o histórico de todo o processo de votação do Brasil, incluindo o período anterior ao uso das UEs. Fez uma abordagem também baseada nos movimentos que pediam a volta do voto impresso no sistema eleitoral brasileiro. Em seu trabalho, ele conclui que o sistema adotado no Brasil garante eleições justas, mas não é a prova de falhas. Sobre urnas em geral, Brunazo (2014) detalhou todas as três gerações de urnas eletrônicas utilizadas em processos de eleição no mundo.

Quanto à democracia, Gritzalis (2002) destrinchou em seu trabalho vários princípios para um sistema seguro que visa atender alguns requisitos constitucionais. Outros trabalhos utilizam esse princípios como base para modelagem de seus projetos.

Fornasier (2022) dispôs em seu estudo uma análise da relação entre democracia e *blockchain*, e como isso pode ser associado a sistemas de votação. Destacou o cenário atual, suas fraquezas

e necessidades, além do seu potencial. Sepúlvida e Paiva (2019) expuseram vulnerabilidades das UEs usadas atualmente no sistema de votação brasileiro e apresentou também detalhes sobre redes *blockchain* e aplicações existentes, tanto de votação eletrônica no geral, como o *e-Estonia*, e votações com *blockchain*, como o *FollowMyVote*.

Sobre *E-Voting*, Gibson *et al.* (2016), Vladucu *et al.* (2023) e Hajian Berenjestanaki *et al.* (2024) fizeram uma revisão do tema com a associação com *blockchain*. Soares e Vasconcelos (2023) fizeram uma proposta de arquitetura distribuída para sistemas de votação eletrônica, e ressaltam a criptografia baseada em tempo como solução em potencial para armazenamento distribuído de votos. Lacerda (2019) implementou um projeto de votação eletrônica utilizando a rede *blockchain Ethereum*, de forma não pública e com nível de permissionamento controlado. Hjálmarsson *et al.* (2018) introduziram um sistema de *E-voting* baseado em *blockchain* que utiliza contratos inteligentes para alcançar eleições seguras e à baixo custo. Além disso, reforça que o uso de redes *blockchain* pode superar barreiras e limitações da implantação de sistemas de votação eletrônica, garantindo transparência e integridade. Este sistema proposto também faz o uso de redes *Ethereum*, que executar centenas de transações por segundo, utilizando os *smart contracts* para aliviar a carga, mas que para países de grande porte, seriam necessárias outras medidas para garantir o desempenho da rede.

O sistema proposto neste artigo diferencia-se de iniciativas existentes. No caso do e-Estonia, o sistema é altamente centralizado, um dos seus desafios é garantir que o processo seja protegido contra a coerção do eleitor, além de questões de privacidade e segurança dos canais de comunicação, principalmente em uma escala maior. Por outro lado, o modelo proposto utiliza uma abordagem baseada em *blockchain*, o que confere maior transparência e descentralização, permitindo que qualquer nó na rede audite o processo após a eleição. Enquanto o e-Estonia é altamente dependente da infraestrutura centralizada do governo para gerenciar os dados de votação, o uso de contratos inteligentes e registros imutáveis no *blockchain* reduz a necessidade de confiança em uma única entidade.

2 Fundamentação Teórica

2.1 Votação no Brasil

Em 1996 começou de forma gradual a transição para a votação através das urnas eletrônicas, sendo totalizada em 2000, quando todo o pleito fez uso da tecnologia (Machado Filho, 2021). A partir desse momento até os dias atuais, as eleições federais, estaduais e municipais são realizadas integralmente através das urnas eletrônicas (Figura 1 e Figura 2) de primeira geração.

Figura 1. Urna Eletrônica utilizada no Brasil em 1996, do tipo DRE
Fonte: Tse (1996).

Figura 2. Urna Eletrônica utilizada no Brasil, a partir de 2022, do tipo DRE
Fonte: Tse (2022).

Quanto às gerações das UEs, Brunazo (2014) detalhou as três gerações de urnas eletrônicas utilizadas em eleições.

- Primeira Geração - ***Direct Recording Electronic*** - ***DRE***: em português, Gravação Eletrônica Direta, surgiu nos anos 1990 e tinha como principal característica armazenar o voto apenas eletronicamente, sem possibilitar auditoria. Nestas urnas, a confiança no resultado era dependente apenas da confiabilidade no software instalado.
- Segunda Geração - ***Voter Verifiable Paper Audit Trail*** - ***VVPAT*** **ou** ***Independet Voter Verifiable Record*** - ***IVVR***: surgiu no ano 2000 e ficou conhecida por permitir uma auditoria através de registros independentes do sistema, para cada voto, além do registro digital das máquinas DRE. Uma aplicação desse tipo de máquina é permitir que o usuário confira seu voto, através de uma cédula impressa pela máquina no ato do registro e deposite em uma outra urna.
- Terceira Geração - ***End-to-End Verifiability*** - ***E2E Verifiability***: Não é representada por um modelo específico de máquinas mas por inovações que tinham como característica aprimorar ou facilitar o processo de auditoria das eleições. Todos esses sistemas tinham em comum a independência do software e a auditoria independente de ponta a ponta no processamento do voto.

No Brasil, a única geração de urna eletrônica até os dias atuais foram as urnas de primeira geração. Apesar disto, as urnas eletrônicas utilizadas nas eleições não foram do mesmo modelo. O TSE, que mantém e gerencia o processo de votação no Brasil, alterou a urna com o passar do tempo, inclusive para modernizá-la e trazer mais transparência aos processos de votação. Entretanto, pelo fato do método de registro de voto ser o DRE, considera-se que as UEs do Brasil são de primeira geração. Segundo Brunazo (2014), os outros países que utilizam votação eletrônica nos seus processos de eleição, já abandonaram o uso das urnas de primeira geração, sendo o Brasil o único a ainda utilizá-la.

Foram encontradas brechas de segurança nas UEs, através dos Testes Publicos de Segurança, organizado pelo TSE, e apresentadas em Aranha, Karam *et al.* (2013) e Aranha, Barbosa *et al.* (2019). Aranha, Barbosa *et al.* (2019) discutem ainda que a UE brasileira ainda não atende os requisitos mínimos de segurança e transparência e que não está desenvolvida o suficiente para uma aplicação de missão crítica de 20 anos. Sugere ainda a adoção de mais medidas de segurança, e a possível adoção de sistemas de *software* independente, como as outras duas gerações de urna. Para ele, o maior desafio da adoção destas tecnologias é o entendimento do TSE de que anexar registros físicos ao sistema compromete o sigilo do voto. Apesar disto, após discursos de ataque a segurança da UE em 2021, Aranha (2021) reforçou não haver evidências de fraude no uso das UEs brasileiras e que sua defesa ao voto impresso (urnas de segunda geração) é apenas por mais transparência ao sistema eleitoral.

2.2 Votação Eletrônica (E-Voting)

E-voting é caracterizado por todo modelo que faça uso de qualquer meio de tecnologia, parcial ou totalmente, durante seu processo de votação (Kersting; Baldersheim, 2004 apud Lacerda, 2019)

(Vladucu *et al.*, 2023; Benabdallah *et al.*, 2022). O modelo de votação eleitoral brasileiro atual, e de vários outros países, se encaixa neste modelo por utilizarem urnas eletrônicas e outros dispositivos. Fora do espectro eleitoral, são diversos os exemplos do uso do e-voting, como eleições informais, enquetes online, votação para programas de entretenimento, e eleições para cargos específicos.

Atualmente, o e-voting é prática mais recorrente devido ao avanço das tecnologias da informação. Em vários países, é comum associar o e-voting a UE, entretanto, existem outras abordagens de eleições com uso da tecnologia, inclusive de forma remota. Desde 2005, a Estônia permite a votação eleitoral de forma remota, através do aplicativo iVoting, a partir de qualquer dispositivo conectado à internet (Fornasier, 2022). Cerca de 46,7% do estonianos utilizam o sistema para votar[1]. Por depender de um ou mais canais de comunicação para realizar a eleição, estes sistemas apresentam desafios técnicos em relação à verificabilidade, confiabilidade, segurança, anonimato e confiança (Gibson *et al.*, 2016).

2.3 Requisitos para um Sistema de Votação Eletrônica Seguro

Gritzalis (2002) propôs alguns conceitos interessantes que foram usados como base para a definição de requisitos de outros trabalhos. Estes conceitos se complementam e se contradizem em algum ponto, sendo impossível projetar um sistema que atenda todos, entretanto, devem ainda ser considerados. Os princípios sugeridos por Gritzalis (2002) são baseados na premissa de atender seis requisitos constitucionais. Estes requisitos são Generalidade, Liberdade, Igualdade, Sigilo, Eleição Direta e Democracia, e deram origem a 15 princípios. A seguir, estes requisitos são listados, assim como os princípios gerados por cada um dos requisitos. Para um maior aprofundamento sobre os requisitos e seus princípios, recomenda-se a leitura do trabalhos de Soares e Vasconcelos (2023) e Gritzalis (2002).

- Generalidade;
  - Isomorfismo ao Processo Tradicional;
  - Elegibilidade ao Voto;
  - Incoercibilidade;
  - Não propaganda no sistema de votação;
  - Capacidade de voto inválido.
- Liberdade;
- Igualdade;
  - Igualdade de Candidatos;
  - Igualdade de Eleitores;
  - Voto único.
- Sigilo;
  - Sigilo;
  - Equilíbrio entre transparência e sigilo.
- Eleição Direta;
  - Voto não monitorado.
- Democracia.
  - Confiança e transparência;
  - Verificabilidade e prestação de contas;
  - Confiabilidade e segurança;
  - Simplicidade.

2.4 *Blockchain*

Blockchain é uma estrutura de dados onde só é permitida a adição de dados, assim, os dados são inseridos em blocos de modo que cada bloco possui o *hash* do bloco anterior (Nakamoto, 2008). Além disso, as redes *blockchain* funcionam sob o controle de um livro-razão, isto é, a cada vez que uma transação ocorre, o algoritmo do contrato inteligente é executado e o resultado é registrado no livro-razão de todos os nós. *Blockchain* é uma forma de tecnologia de contabilidade distribuída na qual as transações são registradas através de valores *hashes*, que são valores fixos gerados através de

[1] Fonte: site oficial do projeto e-Estonia (https://e-estonia.com/solutions/e-governance/e-democracy/)

um algoritmo *hash* para um código de valor variável. Se este valor for alterado e submetido ao mesmo algoritmo, o resultado será totalmente diferente. Isso pressupõe que, se um único bloco em uma cadeia for modificado, fica claro que a cadeia foi adulterada, e isso torna difícil alterar, *hackear* ou enganar o sistema. A partir dessa lógica, é possível aplicar tais conceitos para diversos contextos (Mingxiao *et al.*, 2017 apud Cadiz; Mariscal; Ceniza-Canillo, 2021) (Vladucu *et al.*, 2023). Nesse contexto, as diversas redes *blockchain* podem ser classificada em públicas ou privadas (também conhecidas como permissionadas).

Neste tipo de rede, surge ainda o conceito de *Smart Contracts*, ou Contratos Inteligentes, que são contratos programáveis executados automaticamente quando certas condições pré-definidas são atendidas. Estes contratos automatizam as transações e permitem que as partes cheguem a acordos sem a necessidade de uma entidade intermediária. Além disso, estes contratos são acessíveis por todos os usuário da rede, o que permite sua auditoria (Hjálmarsson *et al.*, 2018; Tanwar *et al.*, 2024).

2.5 *Blockchain* e E-Voting

Pelos benefícios que o conceito de *blockchain* traz, principalmente de imutabilidade de dados registados, contratos inteligentes públicos e possibilidade de acesso por todos os nós da redes, é possível aplicar este conceito em *e-voting*. Vários são os benefícios do uso de *blockchain* para votação. Uma vez que a integridade do sistema eleitoral é fundamental para integridade da própria democracia (Kohno *et al.*, 2004), o uso de tecnologias de *blockchain* são convenientes para um sistema de votação por possibilitar maior transparência e auditabilidade, maior eficiência, menor susceptibilidade a erros e falhas humanas, e redução de custos (Fornasier, 2022; Varaprasada Rao; Panda, 2023; Tanwar *et al.*, 2024; Hajian Berenjestanaki *et al.*, 2024).

Com a remodelagem do sistema, algumas brechas de segurança da UE atual podem ser resolvidas, como o possível desembaralhamento do RDV. Com os votos salvos em uma base única, é mais difícil identificar os votantes a seus respectivos votos. Além disso, outra vantagem seria a confiabilidade no sistema. É claro que a integridade da eleição ainda dependeria da boa-vontade daqueles que o desenvolvem, mas com os contratos inteligentes públicos, ou com iniciativas *opensource*, o sistema pode ser auditado e verificado acerca do seu funcionamento.

O uso de *blockchain* associado a "sistemas eletrônicos para votação estão sendo desenvolvido e aperfeiçoado em diversos lugares do mundo, com a proposta de gerar mais confiabilidade e clareza aos eleitores" (Sepúlvida; Paiva, 2019). Entretanto, um projeto de *E-voting*, para eleições de grande porte, como a eleição federal brasileira onde são computador centeas de milhões de votos, é necessário garantir o funcionamento e a viabilidade da estrutura.

Ademais, aplicar um novo modelo descentralizado de votação vai além da possibilidade técnica. "A implementação de sistemas de votação eletrônica com a tecnologia *blockchain* ainda encontra dificuldades (jurídicas), especialmente aquelas diretamente relacionadas às políticas vigentes em cada país" (Sepúlvida; Paiva, 2019). Segundo Gibson *et al.* (2016), alterar a forma como as pessoas votam tem muitas implicações sociais e políticas, principalmente porque o papel dos administradores eleitorais são completamente diferentes quando estas tecnologias estão envolvidas. Segundo Silva (2018), existe a possibilidade tanto técnica quanto jurídica de realizar a implantação da tecnologia *blockchain* no sistema eletrônico de votação brasileiro. Ele menciona ainda que esta tecnologia é relativamente nova e que deve ser testada para comprovar real eficácia, mas que isso não deve se tornar um empecilho à sua utilização.

2.6 *Ethereum, Solidity* e o *Projeto Remix*

O *Ethereum* é uma rede *blockchain open source*, que assim como o *Bitcoin*, permite a criação de um sistema econômico completo, com gerenciamento de contas e uma moeda nativa. Nesta rede, é possível criar e publicar contratos inteligentes, os quais são a lógica de negócio que é executada na rede, fazendo-se cumprir acordos de pagamento entre as partes.

Uma das linguagens mais utilizadas para escrever os contratos é *Solidity*[2]. Ela uma linguagem de

[2] Solidity: https://soliditylang.org/

alto nível orientada a contratos com traços da linguagem C e JavaScript. Assim como o *Ethereum*, *Solidity* também é uma tecnologia de código aberto.

Um conceito importante sobre as redes *Ethereum* e as *EVM* (*Ethereum Virtual Machine*) é o conceito de *Gas*. Embora publicar contratos na rede seja grátis, as transações são mantidas mediante um custo associado e cada vez que um usuário realiza uma transação, um valor é pre-reservado de sua carteira para arcar com esses custos. O *Gas* é a unidade de medida usada para mensurar esse custo, representando uma pequena fração de um *Ether*. Além de manter os custos, o *Gas* serve para uma outra finalidade que é garantir que os programas executados sejam finalizados, uma vez que possuem um limite de custo, que quando atingido, encerra a transação (Dannen, 2017).

Por fim, existe o *Remix Project*[3], que é uma ferramenta de desenvolvimento de contratos da rede *Ethereum*. Por meio desta ferramenta é possível, escrever, compilar, publicar e testar contratos, tanto em rede local ou na rede *Ethereum*. Após a criação, é possível usar uma conta real para se conectar ao contrato, mas para isso precisa ter saldo de *Ether* (moeda nativa da rede) para efetuar a transação.

3 Proposta de E-voting Baseada em Blockchain

Nesta seção é proposto um modelo de votação eletrônica descentralizado que terá como objetivo contribuir com mais transparência e segurança ao modelo atual através do uso de redes *blockchain*, e a demonstração de um contrato inteligente desenvolvido.

3.1 Proposta do Sistema

Este modelo foi pensado para atuar em conjunto com o sistema atual de votação no registro, armazenamento e acesso dos votos. Nesta proposta, o eleitor ainda teria que registrar seu voto através de urna eletrônica, mas a consulta ao código do algoritmo de consenso e aos votos (após o período de eleições), será público. Cada UE funcionará como um nó de submissão de votos. Apenas nós deste tipo podem submeter votos, e por isso, os endereços de rede desses nós são previamente armazenados na rede. Qualquer outro dispositivo que se conecte à rede poderá consultar seus próprios dados (poderá ver comprovante de votação e dados pessoais), consultar os dados dos candidatos, e após a eleição, verificar a quantidade de votos que cada candidato recebeu. Quanto ao período de eleição, este deve ser cadastrado previamente e ser disposto no *smart contract*, para que seja respeitado o tempo da sessão de votação e o momento em que as consultas de votos serão liberadas.

Dessa forma, uma definição formal da estrutura proposta é: uma composição de nós interligados baseados no conceito de *blockchain*, possuindo um contrato inteligente definido com o propósito de garantir as regras de uma votação eletrônica segura, que permite o voto único por eleitor, a submissão de votos apenas por nós do tipo UE previamente registrados, e a divulgação automática dos resultados após o fim da eleição para qualquer nó conectado.

3.2 Estruturas e Atores do Sistema

Atualmente, o sistema eleitoral brasileiro é mantido pelo Tribunal Superior Eleitoral (TSE), com o apoio dos Tribunais Regionais Eleitorais (TREs), ambos responsáveis por administrar todo o processo eleitoral, desde o desenvolvimento e manutenção das UE, registro prévio de eleitores e candidatos, e eleição e apuração dos votos. No modelo proposto, uma Autoridade Eleitoral, semelhante ao que o TSE representa hoje, ainda terá responsabilidade, entretanto, outros atores participarão do processo, conforme Tabela 1. Este modelo visa que com o uso da rede descentralizada, outros atores, além de uma Autoridade Eleitoral, possam acompanhar o processo. Quanto à estrutura, a comparação pode ser vista na Tabela 2.

O Ator **"Autoridade Eleitoral"** é a entidade responsável por gerir o pleito eleitoral. No Brasil é o TSE, e neste novo modelo a entidade terá o papel de produzir e projetar as novas UEs, desenvolver uma interface compatível e escrever um contrato inteligente para utilizar de maneira embarcada. Além disso, será preciso validar o registro dos candidatos e eleitores, que devem ser controlados para garantir a elegibilidade dos candidatos e prevenir a falsidade ideológica de eleitores. Deve também gerir todo

[3] *Remix Project*: https://remix-project.org/

Tabela 1. Comparação da relação de autoridade entre o modelo atual e o modelo proposto.

Responsabilidade	**Modelo Atual**	**Modelo Proposto**
Registrar Eleitores	Autoridade Eleitoral	Autoridade Eleitoral
Registrar Candidatos	Autoridade Eleitoral	Autoridade Eleitoral
Prover Estrutura	Autoridade Eleitoral	Autoridade Eleitoral
Votar	Eleitores	Eleitores
Registrar Votos	Autoridade Eleitoral	Blocos de registro da rede descentralizada
Armazenar Votos	Autoridade Eleitoral	Qualquer bloco da rede descentralizada
Apuração dos Votos	Autoridade Eleitoral	Eleitores, Candidatos, Autoridade, Cidadãos

Fonte: Autor.

Tabela 2. Comparação da estrutura do modelo atual e do modelo proposto.

Componente	**Modelo Atual**	**Modelo Proposto**
Dispositivo de Registro de Votos	UE de modelo DRE.	UE conectada a rede descentralizada.
Modelo de Rede	Durante as sessões de votação, a UE não é conectada a alguma rede.	Rede *blockchain* descentralizada com regras de registro e consulta.
Modo de contagem de Votos	Contagem através da leitura dos cartões de memória das máquinas.	Consulta através de qualquer nó, após fim do período de votação.

Fonte: Autor.

o processo de votação no dia da eleição (disposição das urnas em zonas eleitorais, fiscalização do processo, etc.).

O Ator **"Eleitor"** é o cidadão em plena execução dos seus direitos previamente identificado. Autentica-se durante o processo de votação e recebe confirmação que votou, embora não possa gerar nenhum comprovante de qual candidato escolheu.

O Ator **"Candidato"** é o cidadão em plenas condições de exercer o cargo ao qual concorre e que está apto a receber votos.

A **"Rede distribuída"** pode ser considerada um ator uma vez que possui grande importância no processo, é responsável por autenticar os eleitores durante o processo de votação, armazenar os votos, controlar as fases da votação e disponibilizar o resultado.

3.2.1 Regras da Eleição e Fases

O modelo foi projetado levando com o objetivo de atingir da melhor maneira os princípios definidos na Seção 2.3. A partir disso, foram definidas regras e premissas foram definidas.

Quanto à rede:

- O modelo de comunicação é o de redes *bockchain*;
- O modelo é composto por nós descentralizados que se comunicam entre si;
- Os nós executarão um *smart contract* com as regras de negócio da aplicação;
- Deve existir criptografia de dados durante a comunicação entre os nós, a fim de evitar interceptações;
- A rede deve ser não pressionada, para permitir a execução de transações por terceiros, facilitando a transparência durante a consulta;
- Qualquer nó conectado a rede poderá consultar votos e o contrato inteligente;
- Nós poderão executar papéis diferentes.

Quanto às regras de negócio:

- Cada eleitor pode registrar voto apenas uma vez;
- Nós do tipo Urna Eletrônica serão os únicos que podem registrar votos na rede. Seus endereços devem ser armazenados e pré-cadastrados antes da eleição.
- Qualquer outro nó não identificado como uma UE poderá consultar informações próprias (se for um eleitor), informações dos candidatos e consultar a contagem de votos ao final da eleição.
- O eleitor vota em um candidato através de um número de identificação único, semelhante ao que ocorre atualmente no sistema eleitoral brasileiro;
- Um eleitor pode verificar seu comprovante de votação, mas não pode obter qualquer comprovante de quem votou;
- Dados biométricos dos eleitores devem ser armazenados na rede para realizar autenticação;
- Votos nulos e brancos devem ser registrados;
- O contrato deve atender as restrições referentes as Fases da eleição;
- Nenhum voto é perdido ou adulterado, e caso isso acontecesse, seria possível a identificação através dos livro-razão dos blocos.

Quanto às fases da eleição:

- A eleição é dividida em três fases (Soares; Vasconcelos, 2023).
- Fase 1 - *Setup* da eleição: Primeira parte do processo, cuja autoridade central deve construir a rede, validar candidaturas, registrar os eleitores aptos na rede e montar o esquema de eleição.
- Fase 2 - Registro de Votos: Período em que as seções permanecem abertas e o sistema está apto a receber votos dos blocos autorizados.
- Fase 3 - Inicia-se após o encerramento da Fase 2 e corresponde ao período pós sessão de votação. Nesta fase, os votos são divulgados.
- Candidatos e Eleitores devem ser registrados apenas durante a Fase 1;
- Votos podem ser registrados apenas durante a Fase 2;
- A contagem de votos é publica ao final do período de eleição, ou seja, durante a Fase 3.

Quanto às regras não restritas à aplicação:

- A autoridade eleitoral deve garantir os princípios não restritos à aplicação (distribuição das urnas,

garantir privacidade durante o ato da votação, fiscalizar o processo e punir os infratores);

- Durante a eleição, a fraude por parte de votantes que cometam falsidade ideológica é reduzido uma vez que o usuário terá que se autenticar com dado biométrico único, previamente registrado na rede;
- Nenhum usuário terá provas físicas de qual candidato recebeu seu voto;
- A rede traz mais transparência pelo fato de que o contrato inteligente é publico e auditável, assim como a contagem de votos.

Durante o processo de votação, o fluxo esperado com o modelo proposto é o que segue:

1. O eleitor se autentica através de dado biométrico na UE;
2. A UE identifica o eleitor e permite seguir com o fluxo;
3. O eleitor digita um número;
4. A UE busca na rede os dados do candidato e exibe seu as informações, caso não exista candidato com o número informado o voto é disposto como nulo;
5. O eleitor confirma o voto;
6. A UE submete o voto na rede:
7. O *contrato inteligente* é executado e:
   - Verifica se está dentro do período de votação;
   - Verifica se o endereço de origem da UE é válido;
   - Verifica se o eleitor já votou;
   - Registra o voto numa estrutura de dados anônima, armazenando o voto associado apenas ao endereço de origem da UE;
   - Registra que o eleitor já submeteu seu voto;
   - Através dos algoritmos de consenso entre os nós, as informações são distribuídas entre todos os outros nós.
8. O eleitor recebe comprovante de votação;
9. Após o encerramento do tempo, as consultas à rede começarão a retornar os dados da votação.

Uma forma simplificada deste fluxo pode ser acompanhada na Figura 3.

Observações:

- O dispositivo de coleta de dados biométricos tanto no cadastro quanto no momento da votação também deve ser íntegro, podendo ser submetido a um outro algoritmo de validação;
- Com os dados biométricos registrados na rede, o eleitor seria capaz de votar de qualquer UE da rede;
- Este modelo de votação poderia ser aplicado em uma votação remota e online, com o auxílio de um dispositivo móvel capaz de verificar biometria com precisão. Se isso ocorresse, contribuiria de fato para atingir o princípio da generalidade (Seção 2.3), mas não garantiria a incoercibilidade por meio de terceiros.

Com este modelo de *E-Voting* é possível contribuir com a discussão acerca da confiabilidade do processo eleitoral brasileiro uma vez que é atacado o principal ponto de crítica do sistema atual. No sistema atual, a falta de confiabilidade baseia-se na premissa que a UE atual não é concordante com o princípio da independência de software, uma vez que sua execução depende do software e hardware embarcados. No modelo proposto, o contrato inteligente é público antes, durante e depois do período de eleição. Além disso, no modelo atual, a apuração dos votos também é contestada, e no modelo proposto, por se tratar de uma rede não permissionada para consulta de votos, a apuração também seria pública e auditável por quem desejar.

3.2.2 Contrato inteligente

Durante o *design* do modelo, foi desenvolvido um contrato inteligente para servir de apoio ao modelo apresentado. A rede *blockchain* utilizada durante o desenvolvimento do projeto foi a *Ethereum*, escolhida por sua popularidade e pela boa disponibilidade de escrever contratos inteligentes numa linguagem de alto nível, a *Solidity*. Um dos pontos da escolha foi a facilidade de implantação destes contratos (Dannen, 2017). Para isso, o ambiente de testes escolhido foi a IDE (*Integrated Development Environment*) *online* do *Remix Project*. Nesta plataforma, foi possível escrever o contrato,

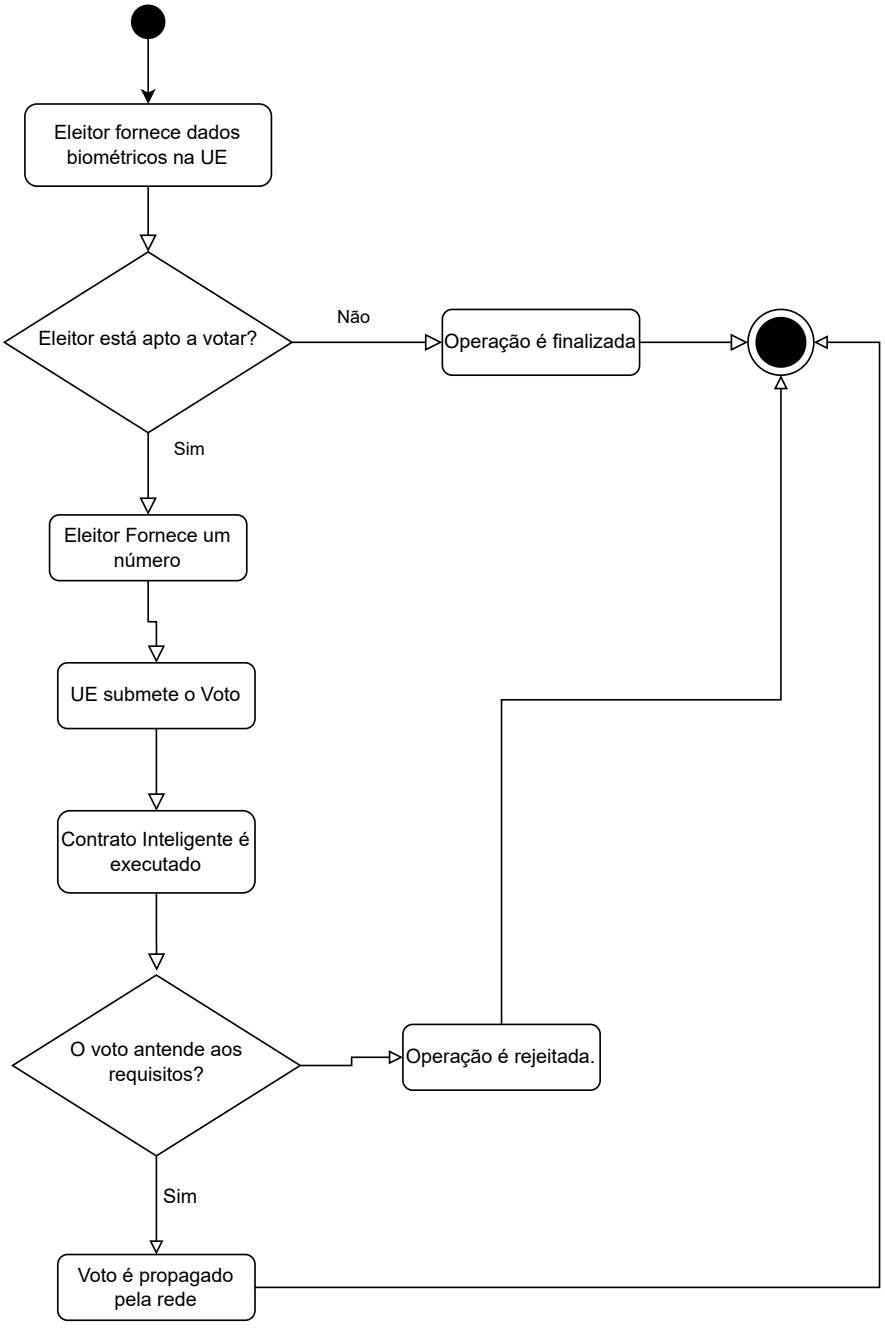

Figura 3. Diagrama de atividades do fluxo esperado de emissão do voto.
Fonte: Autor.

compilar, realizar o *deploy* e executar as transações de teste.

Durante a escrita do contrato, inicialmente foram definidas as estrutura de dados e as variáveis auxiliares, as quais podem ser vistas na Lista 1. As estruturas criadas foram "Voto" e "Candidato". Tendo essas estruturas, foram definidos "*mappings*". Esta estrutura de dados funciona como os dicionários (chave-valor) de outras linguagens de programação. Os *mappings* foram criados para armazenar "candidatos", "votos", "eleitores", "eleitoresVotantes" e "votosPorCadidato". Além disso, foi definido o contador de votos "totalVotos", as definições do horário de início e fim da votação, e um evento de "votoEmitido".

Lista 1. Contrato Inteligente desenvolvido utilizando Solidity.

```
pragma solidity ^0.8.0;

contract SistemaEleitoral {

        //Estrutura do voto
        struct Voto {
                address eleitor;
                int candidatoId;
        }

        //Candidatos participantes da eleicao
        struct Candidato {
                string NomeCandidato;
                string UrlFotoCandidato;
                string NomeVice;
                string UrlFotoVice;
                int Numero;
        }

        //Dicionario de eleitores
        mapping(string => bool) eleitores;

        //Dicionario de eleitores que votaram
        mapping(string => bool) eleitoresVotantes;

        //Dicionario de candidatos
        mapping(int => Candidato) candidatos;

        //Dicionario de votos
        mapping(uint => Voto) votos;

        //Numero de votos por candidato
        mapping(int => int) private votosPorCandidato;


        uint public totalVotos;

        //Hora do inicio da votacao
        uint public horaMinimadeVotos = 1664708400;

        //Hora do encerramento da votacao
        uint public horaMaximadeVotos = 1664740800;

        //Evento para notificar emissao de voto
        event VotoEmitido (address eleitor, int candidatoId);
}
```

Na Lista 2 foram definidas as funções para registro de eleitores e candidatos. Estas funções são de uso antes da eleição e são de responsabilidade da autoridade eleitoral responsável (atualmente seriam o TSE e os TREs).

Lista 2. Funções para registrar candidatos e eleitores.

```
function registrarCandidato(string memory NomeCandidato, string memory
```

```
    UrlFotoCandidato, string memory NomeVice, string memory UrlFotoVice,
    int Numero) public {
        uint horaAtual = obterHoraAtual();
        require(horaAtual < horaMinimadeVotos, "Candidatos nao podem ser
            registrados durante ou depois da eleicao.");
        candidatos[Numero] = Candidato(NomeCandidato, UrlFotoCandidato,
            NomeVice, UrlFotoVice, Numero);
        votosPorCandidato[Numero] = 0;
}

function registrarEleitor(string memory eleitor) public {
        uint horaAtual = obterHoraAtual();
        require(horaAtual < horaMinimadeVotos, "Eleitores nao podem ser
            registrados durante ou depois da eleicao.");
        eleitores[eleitor] = true;
}
```

No Lista 2, o método **registrarCandidato** (linha 2) recebe como parâmetro dados de um candidato apto a disputar a eleição e registra no *mapping* candidatos e votosPorCandidato. Estas informações são seu número único, e nome e endereço da imagem do candidato e seu vice. Este método verifica se esta na fase 1 da eleição. Similarmente, o método **registrarEleitor** (linha 10), recebe como parâmetro dados do eleitor. Neste caso é apenas uma *string*, mas num caso real seria o *hash* de alguma informação biométrica ou coisa do tipo. Verifica se está na fase 1 e registra que existe este eleitor no *mapping* eleitores. Na Lista 3 foi definida a função *obterHoraAtual* para obter a hora atual na rede em formato *Unix TimeStamp*.

Lista 3. Função auxiliar para obter hora atual em Unix TimeStamp do Contrato Inteligente desenvolvido.

```
function obterHoraAtual() public view returns (uint) {
        return block.timestamp;
}
```

Na Lista 4 está definida a função principal de registro de voto. Esta função recebe como parâmetro a identificação do eleitor e o número do candidato a ser votado. A partir disso, são feitas verificações para garantir que o voto a ser emitido é valido. Nas linhas 6, 9 e 12, é verificado se o precedimento está dentro do período de votação, se o eleitor é válido e se o eleitor já votou, respectivamente. Validado isso, o código verifica a existência ou não do candidato escolhido. Vale ressaltar, que para atender o princípio da capacidade do voto inválido (Seção 2.3), o sistema deve registrar votos inválidos ou nulos. Por isso, se o número do candidato não for encontrado, este é considerado um voto inválido, e é registrado com o número -1. Entretanto, para tratar o voto em branco, foi considerado o número fixo aleatório pré-definido, que associado como um "candidato" previamente cadastrado, e tem seu voto armazenado. Feito isso, o contexto inicia os processos de registro do voto: registrar o voto propriamente dito na lista "**votos**", somar o voto na estrutura "**votosPorCandidato**", marcar o eleitor como votante na estrutura "**eleitoresVotantes**", somar mais um voto no contador de votos e por fim emitir o evento "**VotoEmitido**".

Lista 4. Método para registrar o voto do Contrato Inteligente desenvolvido utilizando Solidity.

```
// Registra um novo voto
function emitirVoto(int candidatoId, string memory eleitorId) public {
        uint horaAtual = obterHoraAtual();

        // Verifica se esta em periodo de eleicao
        require(horaAtual >= horaMinimadeVotos && horaAtual <
            horaMaximadeVotos, "Votos nao podem ser registrados fora do
            periodo de eleicao.");

        // Verifica se o eleitor esta registrado
        require(eleitores[eleitorId], "Somente eleitores registrados podem
            votar.");

        // Verifica se o eleitor ja votou
```

```
        require(!eleitoresVotantes[eleitorId], "Eleitores podem votar
            apenas uma vez.");

        // Verifica se o candidato nao esta registrado
        if (candidatos[candidatoId].Numero == 0){
                // Adiciona o voto na contagem de votos nulos
                votosPorCandidato[-1]++;
                // Registra o voto
                votos[totalVotos] = Voto(msg.sender, -1);
        }
        else {
                // Adiciona o voto na contagem do candidato (inclui o
                    candidato "Branco")
                votosPorCandidato[candidatoId]++;
                // Registra o voto
                votos[totalVotos] = Voto(msg.sender, candidatoId);
        }

        // Armazena o eleitor na lista de eleitores votantes
        eleitoresVotantes[eleitorId] = true;

        // Incrementa o contador de votos
        totalVotos++;

        // Emite um evento de registro do voto
        emit VotoEmitido(msg.sender, candidatoId);
}
```

Por fim, na Lista 5 foram definidas as funções de consulta para candidatos e votos. As funções de candidatos buscam através de seu número de identificação e as funções de votos retornam o total de votos em um número bruto ou para cada candidato. Ambas as funções de consulta de voto respeitam o período final da eleição.

Lista 5. Funções de consulta do Contrato Inteligente.

```
// Obter Candidato
function getCandidato(int candidatoId) public view returns (Candidato
    memory) {
        return candidatos[candidatoId];
}

// Retorna o total de votos registrados
function getTotalVotos() public view returns (uint) {
        uint horaAtual = obterHoraAtual();

        // Verifica se o periodo de eleicao esta encerrado
        require(horaAtual >= horaMaximadeVotos, "A contagem de votos pode
            ser exibida apenas apos o periodo de eleicao.");

        return totalVotos;
}

// Total de votos registrados para cada candidato
function getVotosCandidato(int candidatoId) public view returns (int) {
        uint horaAtual = obterHoraAtual();

        // Verifica se o periodo de eleicao esta encerrado
        require(horaAtual >= horaMaximadeVotos, "A contagem de votos pode
            ser exibida apenas apos o periodo de eleicao.");

        return votosPorCandidato[candidatoId];
}
```

Algumas considerações sobre este modelo são que propositalmente que informações dos votos foram armazenadas de maneiras diferentes e redundantes com o intuito de poder trazer ainda mais uma camada de transparência e auditoria, uma vez que o valor de "**TotalVotos**" deve ser igual

a quantidade de objetos do tipo "**voto**" na lista "**votos**", que deve ser igual a soma dos valores da lista "**votosPorCandidato**", que por sua vez deve ser igual ao número de itens na lista "**eleitoresVotantes**". Embora o teste tenha sido realizado apenas com um nó, foi possível trocar as opções de servidores e executar algumas vezes as transações. Dessa forma, o plano da execução do contrato está contido na Tabela 3.

Tabela 3. Passos da execução do contrato inteligente.

Passo	**Ação**	**Descrição**
Passo 1	Preparação do ambiente	Corresponde a Fase 1 do processo. O contrato, contendo o horário atualizado do período de votação foi compilado e publicado na *Remix VM*.
Passo 2	Cadastro de Informações	Ainda durante a Fase 1 do processo, foram registrados dados de eleitores e candidatos. Também foi testado o registro e consulta de votos fora do período.
Passo 3	Registro de votos	Iniciada a Fase 2, os votos foram registrados de maneira aleatória, contendo apenas um voto nulo. Foram testados o registro de candidatos e eleitores e a consulta de votos fora de período.
Passo 4	Consulta de votos	Iniciada a Fase 3, os votos foram consultados e foram testados o registro de eleitores e candidatos e a emissão de votos fora do período de votação.

Fonte: Autor.

A implantação do contrato foi feita em um ambiente virtual (Figura 4) e o acesso do contrato na região de contratos publicados (Figura 5). Com o contrato acessível via interface, foi permitido executar as transações e funções especificadas, sendo possível cadastrar eleitores e candidatos, votar e verificar a cobertura das restrições, como não votar mais de uma vez e respeitar os períodos de votação para votar e consultar votos. Na Figura 4, é possível observar a área de *deploy* da plataforma *Remix*, onde é possível selecionar o ambiente da publicação, neste caso a *Remix VM* de *Shanghai*, a conta, definir o limite de *gas* e valores. Neste local é possível também selecionar o contrato a ser publicado, sendo que este deve ser compilado pelo *Remix*. Na Figura 5, é possível acompanhar a listagem de transações possíveis com o contrato publicado. Esta lista é gerada automaticamente pelo *Remix* após a publicação do contrato. Nessa listagem é possível preencher parâmetros e executar as transações.

Na Figura 6 é possível notar o status, informações e o resultado da transação. Neste caso, foi executa uma transação de registra eleitor. O mesmo acontece na Figura 7, que mostra a execução de uma transação de registrar um candidato, com destaque para as informações referentes a transação como o *hash*, custo, entrada e saída de dados. A Figura 8 mostra o acompanhamento de algumas dessas transações realizadas.

A Figura 9 mostra a execução da transação de emitir um voto. Na direita, é possível notar os parâmetros dos identificadores do candidato e do eleitor, sendo enviados. Na parte parte maior da

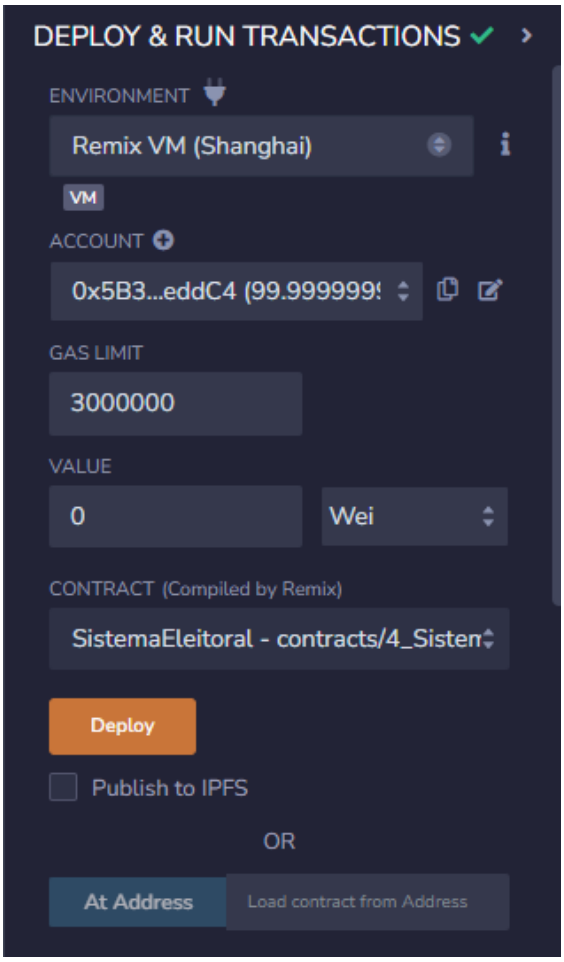

Figura 4. Área de *deploy* do *Remix*.
Fonte: Autor.

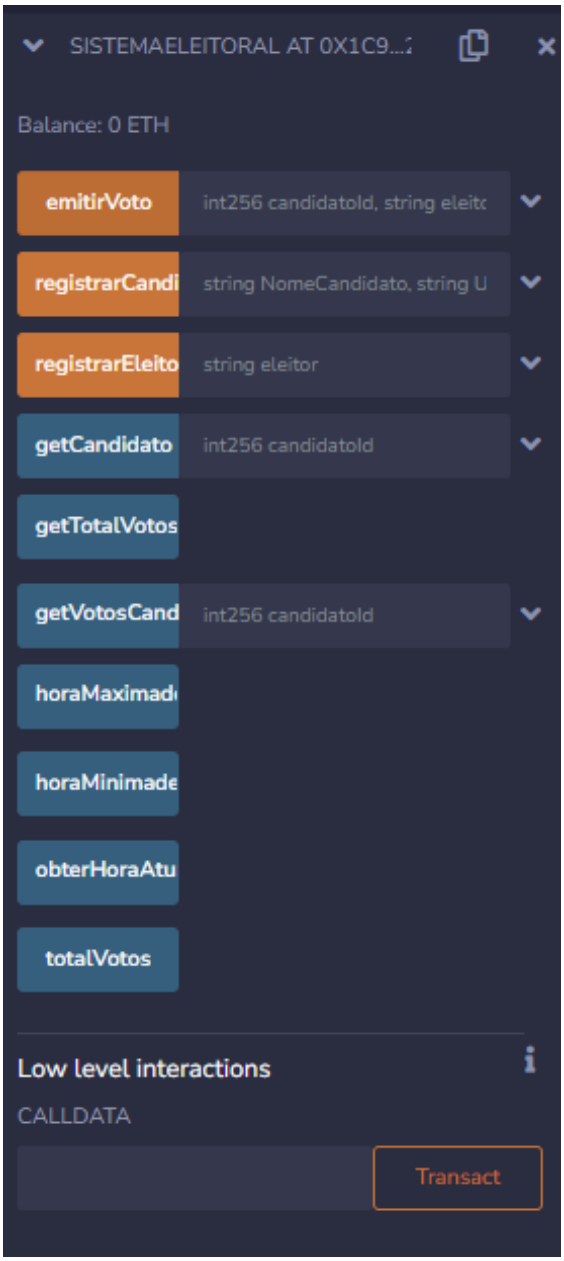

Figura 5. Transações disponíveis na área de contratos publicados.
Fonte: Autor.

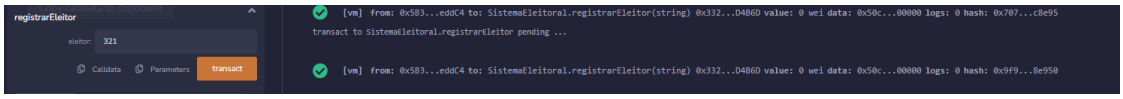

Figura 6. Execução da transação de registrar eleitor.
Fonte: Autor.

Figura 7. Execução da transação de registrar candidato.
Fonte: Autor.

Figura 8. Acompanhamento da execução das transações.
Fonte: Autor.

Figura 9. Execução da transação de emitir voto.
Fonte: Autor.

imagem pode-se acompanhar o resultado da transação.

Já a Figura 10 mostra outra parte importante do processo que é a obtenção de votos. Neste caso, na área direita foi submetida a transação para obter a quantidade de votos do candidato com identificador "1", e na janela de saída é possível ver o resultado, que mostra a contagem de 4 votos.

Figura 10. Obtenção de votos por candidato durante a fase 3.
Fonte: Autor.

Um dos pontos importantes da utilização destes contratos é a automação das regras de negócio e consequentemente, das restrições. Nas Figura 11, Figura 12 e Figura 13 é mostrado a manutenção da restrição de não poder registrar eleitores ou candidatos durante o período de votação, de um eleitor não poder registrar um voto mais de uma vez e a tentativa de obtenção de resultados antes do fim do período de votação. Na plataforma é possível acompanhar as transações que foram rejeitadas, assim como o motivo, neste caso, disparado pelo contrato.

Figura 11. Tentativas de cadastrar eleitores ou candidatos após o início sendo rejeitadas.
Fonte: Autor.

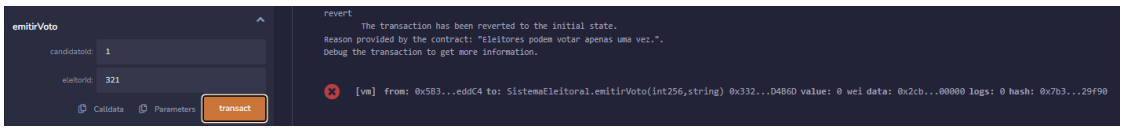

Figura 12. Tentativa rejeitada de votar mais de uma vez com o mesmo identificador de eleitor.
Fonte: Autor.

Por fim, a Figura 14 mostra a obtenção de votos nulos, também de suma importância neste tipo de votação, sendo representados pelo indicador -1, que no caso de testes foi submetido apenas uma vez.

Após a bateria de testes, pode-se concluir que cumpriu a expectativa da regra de negócio proposta, uma vez que com o contrato público, foi possível realizar as transações de acordo com as regras. Entretanto, por limitações, não foi possível garantir requisitos quanto segurança e resiliência.

Um fato interessante a se relatar após a execução dos testes é que o consumo de algumas transações podem consumir valores de *gas* relativamente altos. Embora os testes tenham sido realizados em um ambiente virtual grátis, é possível observar a quantidade de *gas* consumidos durante a transação. Em alguns casos, a transação chegou a consumir aproximadamente U$ 0,25, o que tornaria

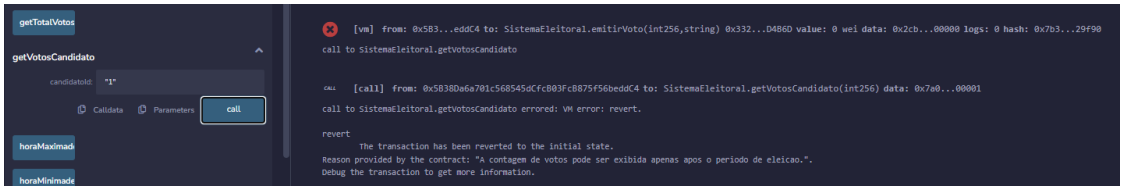

Figura 13. Tentativa rejeitada de obter quantidade de votos antes da fase 3.
Fonte: Autor.

Figura 14. Obtenção de votos nulos, representado pelo identificador -1.
Fonte: Autor.

uma possível aplicação real num contexto de eleições nacionais bem custoso, pois ao multiplicar pela quantidade de votos da eleição de 2022 resultaria numa quantia perto de R$ 150 milhões. Entretanto, há a possibilidade de executar o sistema em uma rede própria com o objetivo de reduzir o custo.

3.3 Desafios Jurídicos e Políticos

A implementação de tecnologias emergentes em eleições nacionais não envolve apenas desafios técnicos, mas também uma série de barreiras jurídicas e políticas. A legislação eleitoral brasileira teria que ser revisada para acomodar o uso de blockchain e contratos inteligentes, principalmente no que diz respeito à segurança do voto e à privacidade dos eleitores. A adoção de uma infraestrutura descentralizada levanta questões sobre quem teria a autoridade final para auditar e validar os resultados eleitorais.

Políticos e eleitores também podem ser resistentes à mudança, principalmente em um contexto onde o sistema eleitoral atual, baseado em urnas eletrônicas, é amplamente aceito, mas também alvo de questionamentos. A transparência oferecida pelo blockchain poderia resolver alguns problemas de confiança no processo, mas sua implementação prática dependeria de uma regulamentação clara e de um consenso político amplo, o que pode ser difícil de alcançar.

Além disso, jurisprudência internacional sobre o uso de blockchain em eleições ainda é escassa, embora iniciativas como o E-Estônia forneçam um ponto de partida para estudos comparativos. O debate sobre a aplicabilidade e a segurança de eleições baseadas em blockchain está em andamento, com defensores apontando suas vantagens em termos de transparência e adversários destacando a complexidade de regular tecnologias descentralizadas em cenários eleitorais críticos.

4 Conclusão

Este trabalho propôs um modelo de estrutura de votação eletrônica que pode ser aplicado em e-voting e com necessidade de garantir os requisitos de um sistema seguro, transparente e íntegro, como o cenário das eleições nacionais. Evidentemente este se trata de um trabalho inicial, não refletindo uma solução finalizada para aplicação em um cenário real. Este trabalho contribui com discussões recentes a cerca da confiabilidade das Urnas Eletrônicas do tipo DRE (*Direct Recording Eletronic*), propondo um modelo descentralizado com uma transparência, segurança e auditabilidade adicional, através do uso de contratos inteligentes em redes *blockchain* públicas.

Foram encontradas iniciativas que combinam votação eletrônica e redes descentralizadas. Diante disso, foi possível concluir que esta premissa era um campo de estudo inicial mas com grande potencial

para trazer as características almejadas para a discussão. Assim, foi proposto um modelo para uso em eleições contando ainda com o recurso das seções presenciais de votação, mas com interligação das urnas para registro e consulta dos votos, sendo a consulta pública após o encerramento do período de votação. Foi descrito também um *smart-contract* desenvolvido em Solidity e testado em ambiente *Ethereum*.

Para o uso deste modelo, evidentemente seria necessária uma complexidade maior de desenvolvimento, além é claro dos esforços necessários pelos órgãos competentes. Por se tratar de um contexto crítico, seu desenvolvimento deverá ser muito criterioso e transparente, para que sua confiabilidade seja aceita. Para trabalhos futuros, sugere-se o desenvolvimento do produto, com mais blocos descentralizados e maior atenção aos conceitos de criptografia entre estes blocos, além de testes de viabilidade, segurança e integridade destes possíveis sistemas descentralizados.

Referências

ARANHA, Diego. *'Não há evidência de fraude': Diego Aranha, professor defensor do voto impresso, rebate discurso bolsonarista*. [*S. l.*]: BBC News Brasil, 2021. Disponível em https://www.bbc.com/portuguese/brasil-58152337. Acessado em Outubro 2022.

ARANHA, Diego F; BARBOSA, Pedro YS; CARDOSO, Thiago NC; ARAÚJO, Caio Lüders; MATIAS, Paulo. The return of software vulnerabilities in the Brazilian voting machine. *Computers & Security*, Elsevier, v. 86, p. 335–349, 2019.

ARANHA, Diego F; KARAM, Marcelo Monte; MIRANDA, André de; SCAREL, Felipe. *Vulnerabilidades no software da urna eletrônica brasileira*. [*S. l.: s. n.*], 2013. Disponível em https://lasca.ic.unicamp.br/media/publications/relatorio-urna.pdf. Acessado em Outubro de 2022.

BENABDALLAH, Ali; AUDRAS, Antoine; COUDERT, Louis; EL MADHOUN, Nour; BADRA, Mohamad. Analysis of Blockchain Solutions for E-Voting: A Systematic Literature Review. *IEEE Access*, v. 10, p. 70746–70759, 2022. DOI: 10.1109/access.2022.3187688.

BRUNAZO, Amílcar. *Modelos e Gerações dos equipamentos de votação eletrônica*. [*S. l.: s. n.*], 2014. Disponível em http://www.brunazo.eng.br/voto-e/textos/modelosUE.htm. Acessado em Outubro 2022.

CADIZ, Jayson V; MARISCAL, Nicole Amber M; CENIZA-CANILLO, Angie M. An Empirical Analysis Of Using Blockchain Technology In E-Voting Systems. *In:* IEEE. 2021 1st International Conference in Information and Computing Research (iCORE). [*S. l.: s. n.*], 2021. p. 78–83.

DANNEN, Chris. *Introducing Ethereum and solidity*. [*S. l.*]: Springer, 2017. v. 1.

FERRÃO, Isadora Garcia; CHERVINSKI, João Otávio; SILVA, Sherlon Almeida da; KREUTZ, Diego; IMMICH, Roger; KEPLER, Fábio; ROSA RIGHI, Rodrigo da. Urnas Eletrônicas no Brasil: linha do tempo, evolução e falhas e desafios de segurança. *Revista Brasileira de Computação Aplicada*, v. 11, n. 2, p. 1–12, 2019.

FORNASIER, Mateus de Oliveira. A democracia e a tecnologia blockchain. *Sequência (Florianópolis)*, SciELO Brasil, v. 42, 2022.

GIBSON, J Paul; KRIMMER, Robert; TEAGUE, Vanessa; POMARES, Julia. A review of e-voting: the past, present and future. *Annals of Telecommunications*, Springer, v. 71, n. 7, p. 279–286, 2016.

GRITZALIS, Dimitris A. Principles and requirements for a secure e-voting system. *Computers & Security*, Elsevier, v. 21, n. 6, p. 539–556, 2002.

HAJIAN BERENJESTANAKI, Mohammad; BARZEGAR, Hamid R.; EL IOINI, Nabil; PAHL, Claus. Blockchain-Based E-Voting Systems: A Technology Review. *Electronics*, v. 13, n. 1, 2024. ISSN 2079-9292. DOI: 10.3390/electronics13010017. Disponível em: https://www.mdpi.com/2079-9292/13/1/17.

HJÁLMARSSON, Friðrik Þ; HREIÐARSSON, Gunnlaugur K; HAMDAQA, Mohammad; HJÁLMTÝSSON, Gísli. Blockchain-based e-voting system. *In:* IEEE. 2018 IEEE 11th international conference on cloud computing (CLOUD). [*S. l.: s. n.*], 2018. p. 983–986.

KERSTING, Norbert; BALDERSHEIM, Harald. *Electronic voting and democracy: a comparative analysis*. [*S. l.*]: Springer, 2004.

KOHNO, Tadayoshi; STUBBLEFIELD, Adam; RUBIN, Aviel D; WALLACH, Dan S. Analysis of an electronic voting system. *In:* IEEE. IEEE Symposium on Security and Privacy, 2004. Proceedings. 2004. [*S. l.: s. n.*], 2004. p. 27–40.

LACERDA, Matheus Miranda. Estudo sobre infraestruturas seguras de votação utilizando Blockchain, 2019.

MACHADO FILHO, Wagner Pereira. Vulnerabilidade da urna eletrônica: mecanismos de segurança e transparência para garantia da lisura no processo de votação. Monografia (graduação em Direito) – Departamento em Direito, Centro de Ciências Sociais Aplicadas, Universidade Federal de Sergipe, São Cristóvão, SE, 2021.

MINGXIAO, Du; XIAOFENG, Ma; ZHE, Zhang; XIANGWEI, Wang; QIJUN, Chen. A review on consensus algorithm of blockchain. *In:* IEEE. 2017 IEEE international conference on systems, man, and cybernetics (SMC). [*S. l.: s. n.*], 2017. p. 2567–2572.

NAKAMOTO, Satoshi. Bitcoin: A peer-to-peer electronic cash system. *Decentralized Business Review*, p. 21260, 2008.

SEPÚLVIDA, Felipe Rabelo; PAIVA, Cláudio Eduardo. Um estudo sobre o uso da tecnologia Blockchain para votação eletrônica. *Revista EduFatec: educação, tecnologia e gestão*, v. 2, n. 1, 2019. Disponível em: https://ric.cps.sp.gov.br/handle/123456789/5066.

SILVA, Matheus Passos. A Segurança da democracia e a blockchain. *Revista Projeção, Direito e Sociedade*, Social Science Research Network, v. 9, 2018.

SOARES, João Marcos; VASCONCELOS, Rafael Oliveira. Uma proposta de arquitetura distribuída para votação eletrônica. *Texto Livre*, v. 16, e42204, abr. 2023. DOI: 10.1590/1983-3652.2023.42204. Disponível em: https://periodicos.ufmg.br/index.php/textolivre/article/view/42204.

TANWAR, Sarvesh; GUPTA, Neelam; KUMAR, Prashant; HU, Yu-Chen. Implementation of blockchain-based e-voting system. *Multimedia Tools and Applications*, v. 83, n. 1, p. 1449–1480, jan. 2024. ISSN 1573-7721. DOI: 10.1007/s11042-023-15401-1. Disponível em: https://doi.org/10.1007/s11042-023-15401-1.

TSE. *Biblioteca da Justiça Eleitoral - Urna eletrônica: Modelo UE 1996*. [*S. l.: s. n.*], 1996. Disponível em https://bibliotecadigital.tse.jus.br/xmlui/handle/bdtse/711. Acessado em Outubro 2022.

TSE. *Conheça os seis modelos de urnas eletrônicas das Eleições 2022*. [*S. l.: s. n.*], 2022. Disponível em https://www.tse.jus.br/comunicacao/noticias/2022/Setembro/conheca-os-seis-modelos-de-urnas-eletronicas-das-eleicoes-2022. Acessado em Outubro 2022.

VARAPRASADA RAO, K.; PANDA, Sandeep Kumar. Secure Electronic Voting (E-voting) System Based on Blockchain on Various Platforms. *In:* SATAPATHY, Suresh Chandra; LIN, Jerry Chun-Wei; WEE, Lai Khin; BHATEJA, Vikrant; RAJESH, T. M. (ed.). *Computer Communication, Networking and IoT*. Singapore: Springer Nature Singapore, 2023. p. 143–151. ISBN 978-981-19-1976-3.

VLADUCU, Maria-Victoria; DONG, Ziqian; MEDINA, Jorge; ROJAS-CESSA, Roberto. E-Voting Meets Blockchain: A Survey. *IEEE Access*, v. 11, p. 23293–23308, 2023. DOI: 10.1109/access.2023.3253682.

Contribuições dos autores

José Alves de Lima Neto: Conceituação, Curadoria de dados, Investigação, Metodologia, Programas, Validação, Escrita – rascunho original, Escrita – revisão e edição; **Rafael Oliveira Vasconcelos**: Metodologia, Administração de projetos, Supervisão, Recursos, Validação, Escrita – rascunho original, Escrita – revisão e edição.