BLOCKCHAINS ALÉM DAS CRIPTOMOEDAS: UMA ANÁLISE PARA E-VOTING

BLOCKCHAINS BEYOND CRYPTOCURRENCY: AN E-VOTING ANALYSIS

2. Administração Pública

Guilherme Zamberlam Pomini, UEM, Brasil, guizamberlam@hotmail.com

Wagner Igarashi, UEM, Brasil, wigarashi@uem.br

Deisy Cristina Corrêa Igarashi, UEM, Brasil, dccigarashi@uem.br

Resumo

Blockchain é uma tecnologia que permite a construção de um "livro razão" distribuído em uma rede, de modo que todo registro escrito seja inalterável. Inicialmente, a maioria das *blockchains* modeladas lidava apenas com criptomoedas, seguindo o exemplo explosivo da moeda que popularizou a tecnologia e se tornou quase sinônimo da mesma, o Bitcoin. Pensando nisso, o presente trabalho propõe uma aplicação da tecnologia *blockchain* fora do mercado de criptomoedas. Por meio da implementação de um Aplicativo Descentralizado em um *blockchain* baseado em Ethereum, foi possível modelar um sistema funcional que gerencia uma eleição virtualmente. O protótipo foi analisado em relação às principais características de uma eleição e julgado como uma possível aplicação da tecnologia *blockchain*.

Palavras chave: *Blockchain*; Smart contract; Eleição virtual.

Abstract

Blockchain is a technology that allows the construction of a "ledger book" distributed on a network, so that every record is written unalterable. Most blockchain models will only deal with the example initially with cryptocurrency currency that has become popular from blockchain technology. With that in mind, the present work of the project is an application of blockchain technology for the cryptocurrency market. By implementing a decentralized application on an Ethereum-based blockchain, it was possible to model a functional system that manages an election virtually. The technology was thought in relation to the characteristics of a selection and judged as a possible application of blockchain.

Keywords: *Blockchain; Smart contract; Virtual election.*

1. INTRODUÇÃO

Com a tecnologia cada vez mais presente na vida social, a maioria dos processos que anteriormente eram manuais, tem migrado para meios digitais. Isto devido ao ganho de maior rapidez e segurança e a demanda por acompanhar as necessidades da sociedade. Mas, mesmo com os avanços tecnológicos alguns processos ainda são feitos de forma analógica, sem mudar drasticamente sua estrutura desde que foram propostos, como por exemplo o processo eleitoral.

Em inúmeros países o esquema de votação ainda é realizado com voto impresso, necessitando de estruturas de gerenciamento e estando vulnerável a adulterações. Saindo da esfera dos

processos analógicos, parte dos processos que tiveram alguma modernização são reféns de órgãos controladores ou empresas, que não atribuem as partes interessadas efetiva transparência ao processo. Uma alternativa nasceu com as moedas virtuais (Nakamoto, 2008), ao introduzir a tecnologia de Blockchain, a qual possibilitou melhorar processos existentes e aumentar a segurança e liberdade dos usuários do sistema.

Neste sentido, analisar a tecnologia *blockchain* sob a ótica de solucionar/melhorar processos que possuem problemas ou limitações na forma em que são realizados atualmente se configura como um potencial adicional a esta tecnologia. Neste sentido, o objetivo desta pesquisa é analisar a viabilidade de aplicação da tecnologia de Blockchain em sistema de votações virtuais. Assim, busca-se responder o seguinte questionamento: Quais elementos indicam como viável a utilização da tecnologia de *blockchain* em sistema de votações virtuais?

O presente estudo se justifica no contexto de que é importante que desenvolvedores e pesquisadores saibam diferenciar o que é factível de ser implementado com base na tecnologia de *blockchain*.

2. FUNDAMENTAÇÃO TEÓRICA

2.1 Blockchain

Viriyasitavat e Hoonsopon (2019) definem uma *blockchain* como sendo "uma tecnologia que permite a imutabilidade e integridade nos dados, mantendo um registro de transações em diversos nós distribuídos conectados por uma rede *peer to peer*". Cong e He (2019) afirmam que uma *blockchain* na sua forma mais simples é um banco de dados distribuído que gerencia de forma autônoma uma lista de transações inseridas em blocos, seguros de qualquer alteração indevida.

Essencialmente, uma *blockchain* é um banco de dados distribuído que guarda um registro de todas as transações que ocorreram na rede, tendo um comportamento similar ao de um livro-razão público com crescimento contínuo. A *blockchain* nasceu como o sistema base para o Bitcoin como forma de eliminar a necessidade de um terceiro elemento (normalmente uma empresa) como entidade validadora de uma transação entre outras duas entidades (Nakamoto, 2008). Utilizando técnicas criptográficas e de consenso, o Bitcoin conseguiu surgir como a primeira moeda virtual livre de qualquer governo. Sendo assim, é possível dizer que o objetivo de toda *blockchain* é criar um sistema de banco de dados descentralizado onde indivíduos e empresas podem guardar informações em um ambiente auto-sustentável e independente de poderes.

As principais características de uma *blockchain* são: **transparência** - ao permitir a todos os usuários o acesso aos dados, todos podem trabalhar como auditores validando os dados; s**egurança** - a *blockchain* funciona como um banco de dados imutável, ninguém possui controle sobre a *blockchain*, nem mesmo os seus criadores; **confiança** - o sistema executa complexos protocolos para garantir um consenso na rede antes de executar cada transação, garantindo que nada será perdido ou alterado; **descentralização** - como todo o sistema é gerenciado pelos nós da rede, não existe a necessidade de intermediários, deixando todo o sistema nas mãos dos usuários.

2.2 Estrutura de uma Blockchain

Uma *blockchain* é formada por uma cadeia de blocos interligados, onde cada bloco possui uma referência (um código *hash*, uma função matemática que codifica um bloco em letras e números) para o bloco anterior, chamado de bloco pai (Zheng, Xie, Dai, Chen & Wang, 2018). Nesta cadeia, cada bloco

serve como armazenamento para as transações (os dados da *blockchain*) e para alguns meta-dados da *blockchain* que funcionam como características chaves de toda a cadeia. O primeiro bloco de uma *blockchain* é chamado de Bloco Gênesis. A Figura 1 ilustra como uma *blockchain* é organizada.

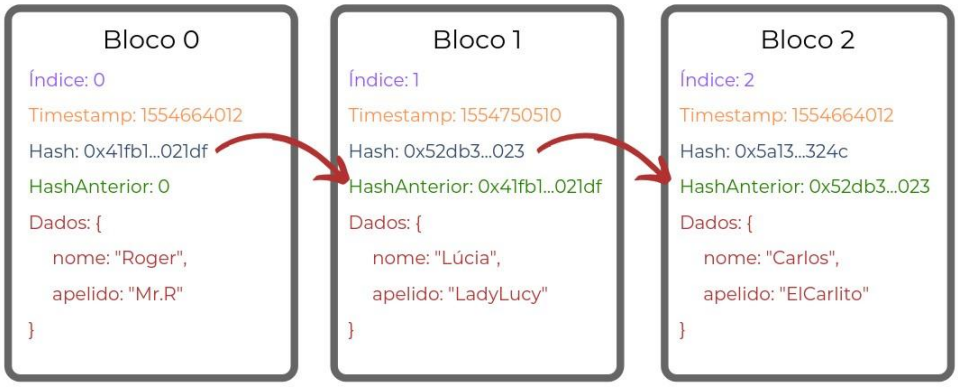

Figura 1: Estrutura de uma *blockchain* Fonte: Rocha (2019)

Com base na figura 1 pode-se dividir uma *blockchain* em dois principais componentes, o Bloco, que é o responsável pela conexão entre os dados da cadeia, e as Transações, que fazem parte de um Bloco e guardam os dados de tudo o que foi processado.

Ao armazenar o *hash* do bloco anterior o processo de ataque à uma *blockchain* se torna inviável. Caso alguma transação seja alterada em um bloco, todo o seu *hash* será alterado, já que tanto transações quanto o *hash* do bloco anterior fazem parte do cálculo do *hash* atual. Com o seu *hash* alterado, os blocos seguintes não terão mais a referência ao bloco alterado, tornando fácil de identificar quando um bloco não pertence à cadeia.

A estrutura de uma *blockchain* pode ser implementada de diversas maneiras, sendo a Pública, Privada, e de Consórcio as mais utilizadas.

Em uma *blockchain* pública todo nó participante pode fazer parte do processo de consenso, é virtualmente impossível de ser alterada já que todo nó é um validador. Os seus registros de transações são públicos para todos de uma forma descentralizada com anonimato aos usuários (Zheng et al., 2018). No entanto, a sua eficiência em processamento é o preço por mais segurança e descentralização, já que é necessário tempo para propagar as transações e blocos à todos os nós da rede.

Vindo em contrapartida à *blockchain* pública, em uma *blockchain* privada somente nós com permissão podem acessar a *blockchain*, normalmente sendo restrita a somente uma organização ou empresa. Com esse processo a eficiência de processamento na rede é alta, mas o sistema acaba gerando uma centralização de todos os nós e perda do anonimato. Com essa centralização e a restrição de quem pode escrever e ler as transações da *blockchain*, o risco de algum nó conhecido tentar (e conseguir) alterar os registros é existente, tendo que ser controlado pela empresa detentora da *blockchain* (Zheng et al., 2018). Normalmente utilizada quando uma empresa precisa de um sistema rápido e com controle mais rígido dos usuários.

Por fim, a *blockchain* de Consórcio utiliza um meio-termo entre a *blockchain* pública e a *blockchain* privada, sendo controlada por um consórcio ao invés de somente uma organização como tentativa de descentralizar os nós participantes, mas ainda possuindo restrições sobre quem pode participar (Zheng et al., 2018). As principais características de uma *blockchain* foram sintetizadas no quadro 1.

Característica	*blockchain* pública	*blockchain* privada	*blockchain* de Consórcio
Permissão de Participação	Sem permissão	Com permissão	Com permissão
Proprietário	Ninguém	Uma organização	Várias organizações
Permissão de Leitura	Pública	Pode ser limitada	Pode ser limitada
Permissão de Escrita	Pública	Nós aprovados	Nós aprovados
Imutabilidade	Praticamente impossível de alterar	Pode ser alterada	Pode ser alterada
Centralização	Descentralizada	Centralizada	Parcialmente descentralizada
Velocidade de Validação	Baixa	Alta	Alta
Anonimato dos Participantes	Sim	Não	Não

Quadro 1: Características de cada arquitetura

2.3 Protocolos de Consenso

Uma parte crítica de uma ***blockchain*** é chegar em um consenso entre os nós na rede, afinal, a ***blockchain*** é um mecanismo para gerar confiança em um ambiente inseguro como a internet. Para isso são necessárias técnicas que garantam de uma forma efetiva uma visão uniforme do estado dos registros e a ordem dos eventos.

Essa visão única da ***blockchain*** é gerada por um consenso entre os nós participantes, seguindo regras pré-definidas, os nós decidem quais blocos são válidos e qual caminho seguir dentro da cadeia de blocos.

No entanto, produzir e manter um consenso em um ambiente descentralizado sem uma autoridade reguladora não é uma tarefa trivial. Para isso, os protocolos das ***blockchain***s incentivam a responsabilidade e exatidão de uma parte dos nós da rede (Cong & He, 2019). Tais nós competem entre si de uma forma a obter a confiança da maioria da rede de acordo com o protocolo aplicado.

A segurança de um protocolo de consenso depende da suposição que os mineradores (os responsáveis por gerar novas moedas) são racionais, ou seja, do pressuposto que é mais conveniente para um minerador seguir as regras do que tentar atacar o sistema. Para essa suposição se sustentar, os mineradores recebem incentivos econômicos para realizarem longas tarefas computacionais (Atzei, Bartoletti & Cimoli, 2017).

Dito isso, os principais protocolos de consenso são explicados a seguir.

2.3.1. Proof of Work (POW)

O protocolo ***Proof of Work*** (Prova de Trabalho) é o mais conhecido e utilizado principalmente na arquitetura do Bitcoin. O protocolo consiste em encontrar um "***nonce***", um número qualquer que quando junto das transações do bloco produz um ***hash*** que atende as restrições do sistema. Por exemplo, na rede do Bitcoin todo bloco precisa ter uma quantidade específica de zeros no início do seu ***hash*** para ser um bloco válido, com a quantidade de zeros representando a dificuldade de encontrar tal "***nonce***", sendo incrementada com o passar do tempo para se adequar aos novos hardwares e técnicas de processamento.

Com isso, o esforço necessário para encontrar um *hash* válido é exponencial ao número de zeros necessários, mas o processo de validação de um "*nonce*" é feito simplesmente executando a função de *hash* uma única vez com o valor encontrado (Crosby, Pattanayak, Verma & Kalyanaraman, 2016). Segundo Bentov, Gabizon e Mizrahi (2016), O propósito de um sistema que utilizada *Proof of Work* é chegar em um consenso sobre o histórico do registro de dados, sincronizando as transações e deixando os usuários seguros contra tentativas de *double-spending*.

No entanto, o processo de POW é muito caro computacionalmente, já que as máquinas necessitam fazer uma imensa quantidade de cálculos criptográficos como forma de encontrar o "*nonce*" necessário. Isso faz com que a necessidade de hardware dedicado, como placas de vídeo especializadas para o processo de criptografia, sejam necessários para produzir um resultado satisfatório, além de exigir um alto custo de energia comparável ao de um pequeno país (Kiayias, Russell, David, & Oliynykov, 2017). Outro lado negativo do POW é que todo o processamento utilizado é perdido (Laurie & Clayton, 2004), sendo utilizado unicamente para encontrar o "*nonce*" e não tendo serventia para outros processos.

2.3.2. Proof of Stake (POS)

O protocolo *Proof Of Stake* surgiu como alternativa ao alto custo do *Proof of Work*, ao invés de ser baseado em cálculos criptográficos para resolver um problema, o POS tenta minimizar o custo de operação da escolha de um bloco válido ao adicionar confiança nos usuários baseado na quantidade de "*tokens*" (a moeda da rede) que cada usuário possui (Kiayias et al., 2017).

Ao contrário do POW, onde cada nó da rede investe processamento e eletricidade no processo, no POS um nó é escolhido em um processo pseudo-randômico onde os nós com mais *tokens* da rede possuem maiores chances. O nó escolhido então investe seus próprios *tokens* no processo de validação de um bloco, gerando assim uma confiança onde o nó validador escolhido não fará nenhuma ação maliciosa pois o seu próprio investimento está em risco.

Contudo, esse processo de gerar consenso com base nos nós mais "ricos" da rede pode gerar problemas como a centralização da validação (e recebimento de dividendos por isso) da parcela mais rica da rede, negligenciando grande parte do sistema.

2.4 Bitcoin

Quando se fala de *blockchain* a primeira coisa que surge na mente das pessoas é o Bitcoin. Nascido junto do *blockchain*, o Bitcoin será abordado neste trabalho devido ao seu caráter de primeira implementação viável do uso de *blockchain* em larga escala, como um exemplo de como a tecnologia de *blockchain* pode ser utilizada no cotidiano da sociedade.

Idealizado em 2008 por um grupo de programadores sob o pseudônimo de Satoshi Nakamoto (NAKAMOTO, 2008), a ideia por traz do Bitcoin é criar um método de pagamento que forneça segurança às partes envolvidas sem a necessidade de um intermediário, utilizando provas criptográficas de grande esforço computacional em uma rede distribuída mundialmente. Dessa forma, o Bitcoin não possui valor como ouro ou joias, mas sim a partir da sua confiança e aceitação das pessoas (Devries, 2016). Segundo Barber, Boyen, Shi e Uzun (2012) o ecossistema projetado para o Bitcoin foi engenhosamente projetado de forma a garantir incentivos econômicos aos participantes, atraindo inúmeros investidores.

Como o Bitcoin não possui um órgão emissor, algum mecanismo precisa existir para gerar novas moedas na rede, tal processo na rede Bitcoin é chamado de mineração (Antonopoulos, 2014). Os nós chamados de "mineradores" na rede competem entre si para resolver um problema computacional, sempre o

mesmo problema, mas com a dificuldade sendo incrementada ao longo do tempo e da necessidade da rede. Na rede Bitcoin esse problema a ser resolvido é utilizado para validar as transações pendentes, garantindo a segurança da rede e gerando como prêmio uma certa quantidade de Bitcoins ao nó vencedor. Dessa forma, a rede usa os próprios usuários para validar as suas transações, pagando aos mesmos pelo seu esforço. Com isso é fácil identificar que o sistema do Bitcoin utiliza *Proof of Work* como forma de gerar uma moeda virtual (Androulaki, Karame, Roeschlin, Scherer & Capkun, 2013).

Contudo, como forma de garantir a estabilidade do Bitcoin, a geração da moeda possui restrições, sendo a mais importante o processo de "*halving*". Desde o seu nascimento o Bitcoin possui um limite para a quantidade de moedas que serão geradas na rede para evitar uma inflação na rede, 21 milhões, que teoricamente será atingido no ano de 2140. Com isso, a recompensa dada aos mineradores de Bitcoin é cortada pela metade (caracterizando o processo de *halving*) a cada 4 anos, valorizando a moeda conforme o tempo.

O Bitcoin tem como potencial inovar as formas de pagamento existentes, assim como a internet e o e-mail inovaram as comunicações (ULRICH, 1892). Removendo intermediários, mas ainda garantindo segurança, não possuindo órgãos emissores e sendo feito totalmente online, fazer uma transação entre duas partes, de qualquer lugar do mundo nunca foi tão fácil.

2.5 Smart Contracts

Dentro do contexto de aplicações que podem tirar proveito das características da *blockchain* temos os contratos inteligentes.

O conceito dos Contratos Inteligentes (do inglês, Smart Contracts) foi proposta por Szabo (1994) sendo definida como um protocolo de transações que executam os termos de um contrato, traduzindo componentes legais de um contrato formal para código de programação e os executando via software e hardware. Com isso, um contrato inteligente possui a vantagem de não necessitar de um intermediário para executar as suas ações legais (Atzei et al., 2017). Contudo, somente definir um contrato em formato de código não garante que o mesmo será executado com segurança conforme planejado. Para tornar o contrato realmente "inteligente" o mesmo é inserido em uma *blockchain*, fazendo uso dos seus protocolos de consenso e transparência para prover proteção ao contrato e aos usuários (Luu, Chu, Olickel, Saxena & Hobor, 2016).

Executado em uma rede de *blockchain*, um Smart Contract tem primeiramente suas regras definidas e programadas para reagirem a eventos na *blockchain*. Conforme um evento ocorre (como um depósito no contrato), a própria rede da *blockchain* reage e executa os termos definidos no contrato, realizando as transferências necessárias em outras transações imediatamente. Um exemplo simples de um Smart Contract pode ser um script para conversão de moedas. Um usuário qualquer A pode criar um script que aceite receber uma certa quantidade de moedas X e que devolva uma certa quantidade de moedas Y. Ao publicar o script em uma *blockchain*, todos poderão ver como o contrato se comporta e como utilizá-lo. O usuário A (dono do contrato) pode abastecer o contrato com suas moedas Y para realizar o câmbio, e quando desejar poderá sacar as moedas X depositadas por outros usuários no contrato, seguro da condição criada que somente o criador do contrato poderá sacar as moedas. Seguindo este mesmo conceito é possível criar contratos complexos com a garantia de segurança, transparência e eficiência da rede de *blockchain*.

2.6 Aplicações Descentralizadas

Uma aplicação descentralizada (do inglês, Descentralized Application, mais conhecido por dApp) é uma aplicação que possui toda ou grande parte de sua estrutura distribuída na rede. Normalmente um dApp possui como *backend* um Smart Contract, utilizando a estrutura de uma *blockchain* como servidor.

Antonopoulos e Wood (2018) citam que as maiores vantagens de um dApp sobre uma aplicação centralizada, sendo essas:

- Resiliência: ao ter o seu código definido por um Smart Contract em uma *blockchain*, a disponibilidade da aplicação depende puramente da disponibilidade da rede e não de um único servidor.
- Transparência: o seu código pode ser inspecionado e toda interação com o dApp será armazenada na *blockchain*.
- Resistência à censura: devido a natureza de imutabilidade de um Smart Contract, nem mesmo o criador do contrato pode alterar as regras depois que ele chega na *blockchain*. Com isso, virtualmente ninguém pode ser impedido de interagir com um dApp.

Com base nestas características é factível dizer que a modelagem de dApps pode vir a mudar e reinventar até mesmo a estrutura da web (Antonopoulos & Wood, 2018), dando mais poder aos seus usuários e removendo os vínculos de dependências existentes.

2.7 Ethereum

Normalmente quando se fala de *blockchain*s, a primeira utilização que vêm a mente de qualquer pessoa é o Bitcoin. Vindo logo atrás está o Ethereum, que atualmente já conquistou o posto da *blockchain* mais utilizada do mundo, com uma quantidade até duas vezes maior de transações que a do Bitcoin (Ossinger, 2020). No entanto, mesmo com as pessoas acreditando que as duas *blockchain*s competem entre si (o que não deixa de ser verdade), elas possuem mais diferenças do que semelhanças.

O Ethereum ao contrário do Bitcoin não nasceu com o intuito de se tornar uma criptomoeda, os seus criadores citam o Ethereum como uma grande máquina de estados descentralizada, onde as transações realizadas na rede alteram o estado do mundo compartilhado pelos seus usuários (Wood, 2014). O estado na *blockchain* não guarda somente os dados das transações, mas também uma máquina de estados que pode ser alterada de bloco para bloco executando códigos em uma máquina virtual chamada EVM (Ethereum Virtual Machine).

Com isso, é possível ver o poder da rede do Ethereum, comumente visto como um computador extraordinariamente poderoso (Hildenbrandt et al., 2018). Por consequência da sua EVM, o Ethereum consegue atingir o seu objetivo de executar aplicações descentralizadas e Smart Contracts de uma maneira que permite uma expansão das funcionalidades de uma *blockchain* comum.

2.8 Sistema Eleitoral

Outra aplicação sensível a questões de segurança, confiabilidade e rastreabilidade são os sistemas eleitorais, os quais podem tirar proveito das características da *blockchain*.

Um sistema eleitoral define regras de como um processo de votações é executado e como o seu resultado é gerado. A forma mais conhecida de um sistema eleitoral são as eleições organizadas por um governo, normalmente para decisões de liderança no governo ou para referendos. No entanto, um sistema eleitoral abrange desde a política de um país até votações informais como escolhas de popularidade.

Contudo, mesmo com as variações de detalhes entre cada tipo de eleição, todos compartilham uma característica em comum: indivíduos votam e esperam que o seu voto seja computado corretamente, respeitando as regras, para gerar um resultado (Farrell, 2011). Em uma votação o elemento mais importante para as partes é a contagem correta dos votos, garantindo que não existe fraude no processo eleitoral.

Dada esta introdução sobre sistemas eleitorais é necessário lembrar que a mesma abrange inúmeros campos de estudo, e uma teoria mais aprofundada sobre a temática sairia do escopo do presente trabalho.

2.9 Trabalhos Correlatos

Durante o desenvolvimento deste trabalho foram estudados artigos sobre *blockchain*, Bitcoin e suas implementações. No âmbito das implementações de *blockchain* foram priorizadas as de matéria diferente da área financeira, com foco em sistemas de eleições virtuais.

No artigo de Crosby et al. (2016) são apresentadas diversas aplicações teóricas baseadas em *blockchain*s, sendo citado brevemente a ideia de votações virtuais. Com base no conceito apresentado neste artigo, nasceu a ideia de expandir e investigar a viabilidade de um sistema para votações virtuais. Já no trabalho de Hjálmarsson, Hreiðarsson, Hamdaqa, Hjálmtysson (2018) são expostos os problemas e dificuldades dos sistemas atuais para votações em eleições, desde o sistema de voto em papel ao eletrônico. O artigo cita principalmente a falta de confiança e auditabilidade dos sistemas atuais e propõe o uso da tecnologia de *blockchain* juntamente com Contratos Inteligentes como um serviço para permitir eleições mais seguras.

Similarmente, o trabalho de Hanifatunnisa e Rahardjo (2017) propõe o uso de *blockchain* em um sistema de eleições, mas com a utilização de Contratos Inteligentes, propondo um processo baseado em "turnos" de votação para cada nó da rede. O artigo também produz uma análise sobre o espaço de armazenamento necessário para o sistema de *blockchain*, e o tempo de processamento da rede com base na quantidade de nós. Os autores concluem que, embora os valores de tempo e armazenamento cresçam com a quantidade de nós da rede, ainda estão em valores adequados com a capacidade computacional atual, tornando o sistema viável de ser implementado.

2.9.1 Análise dos Trabalhos Correlatos

Todos os trabalhos analisados propõem o uso de *blockchain*s em um processo eleitoral, em especial em eleições governamentais, que é o tipo de eleição mais suscetível a críticas. Em especial o trabalho de Hjálmarsson et al. (2018) se preocupa com o conceito de atingir um estado chamado de "Democracia Líquida". Segundo Hjálmarsson et al. (2018), em um estado de Democracia Líquida, um eleitor pode a qualquer momento revisar como o seu voto foi feito, garantindo que a sua escolha foi contada propriamente no processo eleitoral. A preocupação com uma eleição transparente também é exposta por Crosby et al. (2016) e Hanifatunnisa e Rahardjo (2017), que levantam dúvidas se os processos atuais de eleição conseguem assegurar que nenhuma entidade interfere no resultado de uma votação.

3. DESENVOLVIMENTO

Tendo em vista que o objetivo deste trabalho é a utilização da tecnologia de *blockchain* na área votações virtuais, o foco deste trabalho não será na construção da *blockchain* em si, e sim na sua utilização. Foi modelado um protótipo para gerenciar uma eleição online simples entre dois candidatos.

A eleição modelada terá a característica especial de um administrador, o próprio endereço que criou o contrato na *blockchain*. O administrador terá a função de conceder a permissão de voto aos eleitores que

fizeram a requisição. Desta forma, somente usuários validados pelo administrador poderão realizar um voto na eleição.

A eleição terá o seguinte funcionamento: 1 - O eleitor se conecta com a sua chave privada; 2 - O eleitor faz a requisição da permissão de voto para o administrador; 3 - O Administrador analisa a requisição e aprova, garantindo ao eleitor o direito de voto; 4 - O eleitor escolhe um candidato e confirma seu voto; 5 - Os usuários podem visualizar o resultado da eleição, bem como os votos enviados para o contrato.

Neste capítulo é apresentada a implementação do protótipo realizada ao longo do estudo para simulação de uma eleição online utilizando *blockchain*.

3.1 Materiais e Métodos

O protótipo do estudo foi construído como uma aplicação descentralizada (dApp) utilizando a linguagem de programação Javascript com a utilização do ambiente Node.Js para execução do código. Por sua vez, para permitir a comunicação entre a linguagem de comunicação e a plataforma de *blockchain* Ethereum, foi utilizada a biblioteca Web3. O protótipo foi dividido em duas partes principais o *frontend* (responsável pela interação com o usuário) e o *backend* (responsável pela interação com o *blockchain* Ethereum.

Para o *frontend* foi utilizado HTML e CSS, utilizando o framework Bootstrap e React. Já para o *backend* foi necessária a utilização da linguagem Solidity, necessária para a definição do Smart Contract da eleição. Junto da linguagem Solidity, foram utilizadas as ferramentas, Ganache e o Metamask.

3.2 Arquitetura do Smart Contract

O contrato de eleição foi modelado com base na arquitetura apresentada na figura 2.

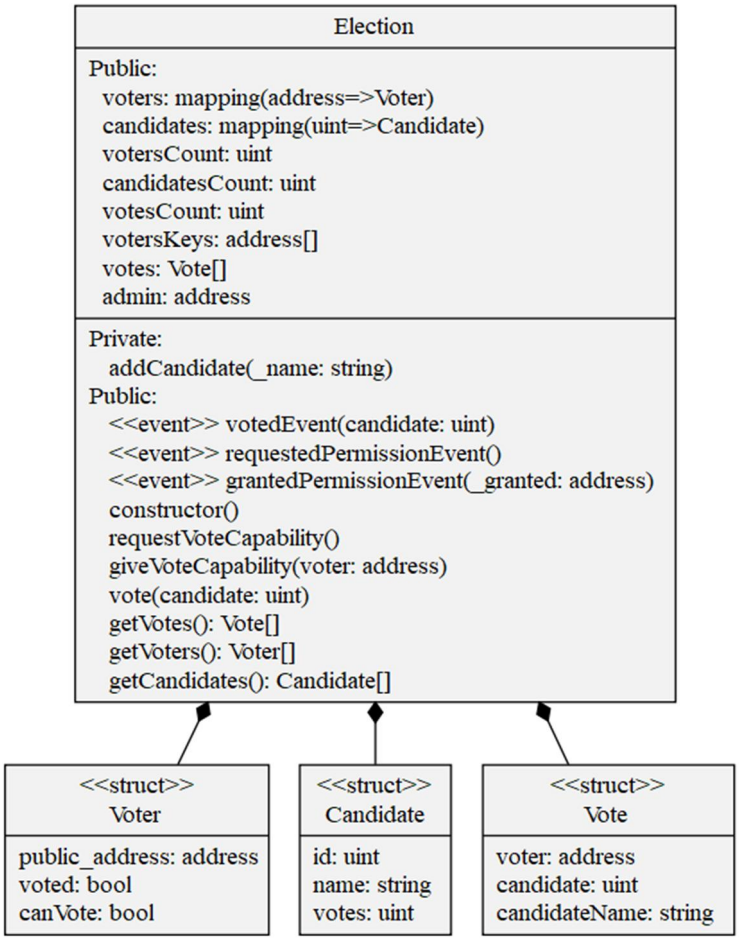

Figura 2: Arquitetura do Contrato de Eleição

De acordo com a figura 2, como a linguagem solidity é orientada à objetos, para melhor controle dos dados foram modeladas algumas classes que simbolizam modelos pertencentes ao contrato da eleição. Além das classes, o contrato também possui métodos auxiliares para interagir com a *blockchain* e fazer as validações necessárias. A seguir será descrito como foi definida cada entidade do contrato.

A primeira classe a fazer parte do contrato é a classe Voter que simboliza um eleitor, com a informações: se pode votar, se já votou; e qual é o endereço registrado (sua chave pública).

A próxima classe do contrato é a Candidate, que faz a abstração das informações de um candidato no sistema: o id (para melhor controle e comunicação com o backend), name para o frontend da aplicação, e votes que armazena a quantidade total de votos do candidato.

É necessária a classe Vote que simboliza e guarda as informações de um voto na eleição para o sistema ter transparência no momento da contagem e validação dos votos. Um voto possui como atributos: uma referência ao endereço do eleitor (a chave pública da classe Voter), o id do candidato cujo o voto é destinado, e o nome do candidato.

Para o funcionamento do contrato é necessário a definição de informações e controles gerais da eleição, que farão parte do estado da aplicação. Por isso foi criada a classe Election.

3.3 Interface do Usuário

Para o usuário interagir com o sistema de uma forma simples, foi criado uma interface web que se comunica em tempo real com o Smart Contract e a *blockchain*. A interface foi dividida em três telas chamadas de: Votação, Resultados e Eleitores.

A interface de votação gerencia todo o processo de votação para o eleitor com base no estado das informações do contrato. Os dois primeiros estados são: eleitor ainda não solicitou a permissão de voto e eleitor aguardando a confirmação da permissão, ilustrados na Figura 3.

Figura 3: Telas de permissão de voto

Após o registro e autorização do administrador, o eleitor poderá votar (Figura 4).

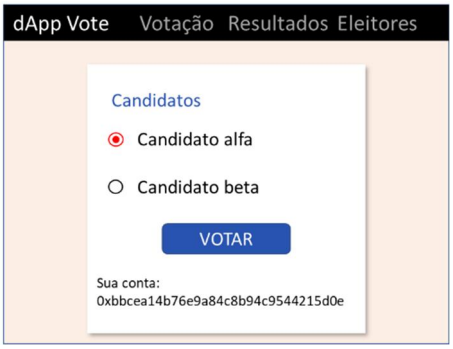

Figura 4: Tela de Votação - Com permissão de voto

Após a transação do voto ser aceita pelo contrato, o eleitor terá concluído a sua participação no processo eleitoral, levando à última tela do protótipo (Figura 5).

Figura 5: Tela de Votação - Voto realizado

O protótipo tem mais duas telas onde são apresentados: os resultados da votação (Figura 6); e outra contendo o controle de eleitores (Figura 7), sendo possível identificar quais são os eleitores e se estes já votaram.

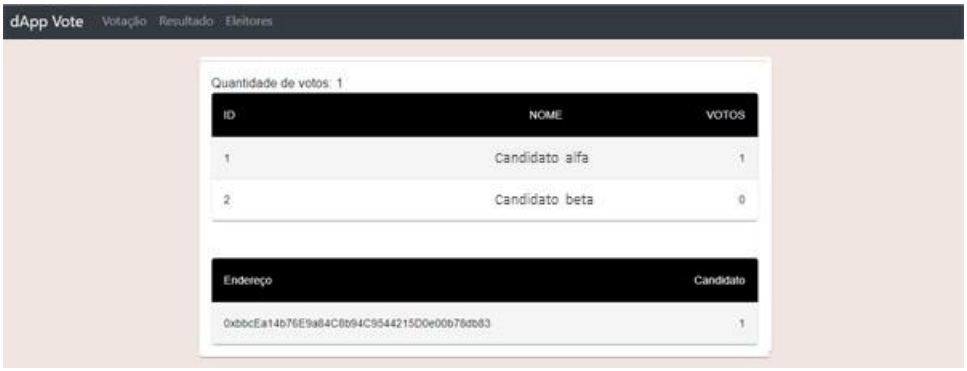

Figura 6: Resultados da votação

Figura 7. Controle de eleitores

4. ANÁLISE DE RESULTADOS

Tendo em vista que o sistema proposto visa criar um sistema eleitoral, o mesmo foi avaliado com base nos princípios básicos de uma votação descritos por Corry (2009): anonimato do voto, unicidade do voto por pessoa, elegibilidade do eleitor, transparência dos votos, contagem correta dos votos e confiabilidade do sistema.

O protótipo gerado conseguiu cumprir com todos os requisitos de uma votação segura. Através da sua interface e Smart Contract foi possível conseguir: 1- Anonimato do voto - os votos no sistema não podem ser vinculados a pessoas físicas já que são identificados unicamente por uma chave criptográfica pública; 2 - Unicidade do voto - por meio de lógicas presentes na formulação do Smart Contract é impossível o mesmo endereço realizar mais de um voto; 3 - Eligibilidade do eleitor - com a utilização de um administrador da eleição, somente os eleitores autorizados pelo mesmo podem realizar um voto no sistema; 4 - Transparência dos votos - todos os usuários do sistema podem recuperar e visualizar a lista dos votos presentes no contrato, sendo assim, a eleição é auditável e transparente; 5 - Contagem correta dos votos - todos os votos são corretamente computados por meio da lógica do Smart Contract; 6 – Confiabilidade - o sistema é confiável com base nos princípios de uma *blockchain*, nenhuma transação e nem mesmo o Smart Contract podem ser alterados uma vez que são escritos na *blockchain*.

A partir desta análise pode-se concluir que o sistema eleitoral proposto no estudo abrange as características necessárias para uma votação segura.

No protótipo desenvolvido a *blockchain* possui um comportamento semelhante ao de um “backend” de aplicações convencionais. A *blockchain* armazena tanto a estrutura da lógica da aplicação da eleição (o Smart Contract) quanto as informações, se comportando como um banco de dados.

Através da utilização da *blockchain* foi possível criar uma aplicação que não depende de um servidor e utiliza a própria rede de nós para executar suas funções, sem agregar muita dificuldade de implementação. Além disso, por conta de todos os protocolos de funcionamento da *blockchain* expostos na Fundamentação Teórica, o sistema por utilizar da tecnologia também incorpora as mesmas vantagens, gerando uma aplicação confiável e com a segurança que os seus dados não serão adulterados.

De modo a avaliar a execução do protótipo, foram realizados testes automatizados. Embora a própria interface tenha tratamentos para evitar que um usuário faça uma operação não permitida no contrato, um usuário com conhecimento de programação ainda poderia interagir com o contrato através de um código próprio.

Como forma de garantir que o contrato modelado não possui falhas de segurança e que as funcionalidades estão de acordo com o esperado, foram escritos testes automatizados para as principais funcionalidades. Os testes são executados na *blockchain* disponibilizada pelo Ganache através das ferramentas contidas no Truffle, sendo assim os testes simulam um ambiente real de operação.

A Figura 8 mostra que todos os testes criados foram executados com sucesso,comprovando a segurança do contrato desenvolvido.

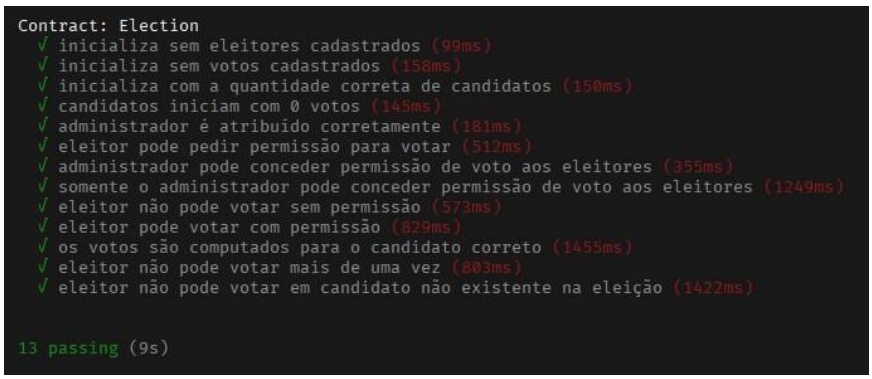

Figura 8: Resultado dos testes automatizados

5. CONCLUSÃO

Blockchain é uma tecnologia que permite a construção de um “livro razão” distribuído em uma rede, de forma que todo registro escrito é inalterável. Inicialmente, a maioria das *blockchain*s modeladas lidava somente com criptomoedas, seguindo o exemplo da moeda que popularizou a tecnologia e virou quase sinônimo da mesma, o Bitcoin. Com o passar dos anos as *blockchain*s foram ganhando maturidade no mercado, novos tipos de aplicações utilizando a tecnologia foram surgindo, ficando evidente o potencial de aplicação de uma *blockchain* além do mercado financeiro.

Neste contexto, o presente estudo teve como objetivo o estudo de possíveis aplicações da tecnologia de *blockchain* fora do mercado de criptomoedas/financeiro, de forma a resolver um problema tangível no mundo real. Para isso, foi feito um levantamento das tecnologias presentes em uma *blockchain* e implementado um protótipo de aplicação para eleições virtuais.

Ao final, verificou-se que o protótipo modelado tem limitações, entretanto, o mesmo gerou resultados satisfatórios, produzindo um sistema funcional que gerencia uma eleição de forma online e

descentralizada. A partir do protótipo e dos estudos realizados pôde-se concluir que a utilização de sistemas com ***blockchain*** pode ir além da área de criptomoedas, tendo grande aplicabilidade a outros problemas reais.

Como possibilidades de trabalhos futuros pode-se listar: a melhoria no sistema de autenticação para utilização do sistema, podendo usar tecnologias como reconhecimento facial; separação de fases bem definidas para o processo de votação.

REFERÊNCIAS

Androulaki, E.; Karame, G. O.; Roeschlin, M.; Scherer, T.; Capkun, S. (2013). Evaluating user privacy in bitcoin. In: *International Conference on Financial Cryptography and Data Security*. NEC Laboratories Europe: Springer, p. 34–51.

Antonopoulos, A. M. (2014). *Mastering Bitcoin:* unlocking digital cryptocurrencies. Sebastopol, California: "O'Reilly Media, Inc.".

Antonopoulos, A. M.; Wood, G. (2018). *Mastering ethereum:* building smart contracts and dapps. [S.l.]: O'reilly Media.

Atzei, N.; Bartoletti, M.; Cimoli, T. (2017). A survey of attacks on ethereum smart contracts (sok). In: SPRINGER. *International conference on principles of security and trust*. Università degli Studi di Cagliari, Cagliari, Itália. p. 164–186.

Barber, S.; Boyen, X.; Shi, E.; Uzun, E. (2012). Bitter to better—how to make bitcoin a better currency. In: SPRINGER. *International conference on financial cryptography and data security*. University of California, Berkeley, USA. p. 399–414.

Bentov, I.; Gabizon, A.; Mizrahi, A. (2016). Cryptocurrencies without proof of work. In: SPRINGER. *International conference on financial cryptography and data security*. Department of Computer Science,Technion, Haifa,Israel. p. 142–157.

Cong, L. W.; HE, Z. (2019). *Blockchain* disruption and smart contracts. *The Review of Financial Studies*, Oxford University Press, v. 32, n. 5, p. 1754–1797.

Corry, C. E. (2009). *Vote Fraud And Election Issues*. Disponível em: <http://www.ejfi.org/Voting/Voting.htm#fraud>. technology: Beyond bitc

Crosby, M.; Pattanayak, P.; Verma, S.; Kalyanaraman, V. (2016). *Blockchain* oin. *Applied Innovation*, v. 2, n. 6-10, p. 71.

Devries, P. D. (2016). An analysis of cryptocurrency, bitcoin, and the future. *International Journal of Business Management and Commerce*, v. 1, n. 2, p. 1–9.

Farrell, D. M. (2011). *Electoral systems: A comparative introduction*. London, England: Macmillan International Higher Education.

Hanifatunnisa, R.; Rahardjo, B. (2017). *Blockchain* based e-voting recording system design. In: IEEE. *2017 11th International Conference on Telecommunication Systems Services and Applications (TSSA)*. Lombok, Indonesia. p. 1–6.

Hildenbrandt, E.; Saxena, M.; Rodrigues, N.; Zhu, X.; Daian, P.; Guth, D.; Moore, B.; Park, D.; Zhang, Y.; Stefanescu, A. et al. (2018). Kevm: A complete formal semantics of the ethereum virtual machine. In: IEEE. *2018 IEEE 31st Computer Security Foundations Symposium (CSF)*. p. 204–217.

Hjálmarsson, F. Þ.; Hreiðarsson, G. K.; Hamdaqa, M.; Hjálmtysson, G. (2018). *Blockchain*-based e-voting system. In: IEEE.` *2018 IEEE 11th International Conference on Cloud Computing (CLOUD)*. San Francisco, CA, USA. p. 983–986.

Kiayias, A., Russell, A., David, B., Oliynykov, R. (2017). Ouroboros: A Provably Secure Proof-of-Stake Blockchain Protocol. In: Katz, J., Shacham, H. (eds) *Advances in Cryptology – CRYPTO 2017*. Lecture Notes in Computer Science, vol 10401. Springer, Cham. https://doi.org/10.1007/978-3-319-63688-7_12.

Laurie, B.; Clayton, R. (2004). Proof-of-work proves not to work; version 0.2. In: ONLINE. *Workshop on Economics and Information, Security*. Cambridge, United Kingdom.

Luu, L.; Chu, D.-H.; Olickel, H.; Saxena, P.; Hobor, (2016). A. Making smart contracts smarter. In: ASSOCIATION FOR COMPUTING MACHINERY. *Proceedings of the 2016 ACM SIGSAC conference on computer and communications security*. New York, NY, United States. p. 254–269.

Nakamoto, S. (2008). *Bitcoin: A peer-to-peer electronic cash system*. Disponível em: <https://bitcoin.org/bitcoin.pdf>.

Ossinger, J. (2020). *Ethereum Becoming More Than Crypto Coder Darling, Grayscale Says*. Disponível em: <https://www.bloomberg.com/news/articles/ 2020-12-04/ethereum-becoming-more-than-crypto-coder-darling-grayscale-says>.

Rocha, W. G. C. da. (2019). *Entendendo blockchain*. Disponível em: <https://bitbaysolucoes.com.br/blog/articles/entendendo-*blockchain*.html>.

Szabo, N. (1994). Smart contracts. *Unpublished manuscript Online*.

Viriyasitavat, W.; Hoonsopon, D. (2019). *Blockchain* characteristics and consensus in modern business processes. *Journal of Industrial Information Integration*, Elsevier, v. 13, p. 32–39.

Wood, G. (2014). Ethereum: A secure decentralised generalised transaction ledger. *Ethereum project yellow paper*, v. 151, n. 2014, p. 1–32.

Zheng, Z.; Xie, S.; Dai, H.-N.; Chen, X.; Wang, H. (2018). *Blockchain* challenges and opportunities: A survey. *International Journal of Web and Grid Services*, Inderscience Publishers (IEL), v. 14, n. 4, p. 352–375.