Preserving Transparency and Integrity of Elections Utilising Blockchain Technology

Abdallah Al-Zoubi
Princess Sumaya University for Technology, Jordan

Mamoun Aldmour
Staffordshire University, UK

Rakan Aldmour
Staffordshire University, UK

Abstract: Digital voting is increasingly important in both established and emerging democracies. Some of the advantages of digital voting are faster vote count and tabulation; accurate results; increased voters' participation and convenience; and effective handling of complex electoral system formats that require laborious counting procedures. However, transparency, credibility, and integrity concerns, as well as the limited possibility of recount, usually make traditional digital voting systems unpopular. Digital voting using blockchain technology, however, is safe, transparent, and immutable, which makes it a suitable choice for future decentralized voting systems. In particular, the Ethereum blockchain is proposed as an appropriate platform for the backbone of an e-voting system due to its widespread use, transparency, consistency and provision of smart contracts. Initial piloting on the implementation of a blockchain-based voting framework in Jordan shows promising results on its transparency and integrity by incorporating a space for representatives and observers to monitor the election procedure and results as an additional measure to ensure its efficiency and reliability. The uptake of the proposed system calls for further debate and dialogue amongst governments and people, especially in developing countries where democracy is still in its infancy.

Keywords: e-Voting System, Blockchain Technology, Ethereum Platform, Digital Applications.

Introduction

Automatic voting systems are as old as democracy itself. An innovative method to safeguard against voter fraud while maintaining a sense of transparency using a ballot box containing a glass globe mounted in a frame was proposed by Samuel C. Jollie in 1858 (Jones, 2009). Early

http://doi.org/10.18080/jtde.v10n4.626

voting machines consequently dispensed entirely with any form of durable ballot and, in so doing, they provided reasonable secrecy. The first pushbutton voting machine appropriate for use in a general election in the United States was introduced by Anthony Beranek in 1881, thereafter improved and suggested by Jacob H. Myers in 1892. Four types of e-voting systems have mainly dominated the landscape since: direct recording electronic (DRE) voting machines; optical mark recognition (OMR); electronic ballot printers (EBPs); and Internet voting, where votes are cast from anywhere and transferred to a central counting server. In fact, digital voting commenced in the early 1960s by utilizing punched cards and subsequently evolved from basic transmission of tabulated results to a full-function online voting process. The degree of automation varied from marking a paper ballot to a comprehensive system encompassing vote input and recording, data encryption and transmission to servers, and consolidation and tabulation of election results (Abuidris *et al.*, 2019).

Estonia was the first country to fully implement an electronic voting system in local elections in 2005, and has since been legally bound to hold general elections using the Internet as a means of casting votes (Tsahkna, 2013). Today, there are over 34 countries which adapt and implement electronic voting in parliamentary elections, Pakistan being the latest when its National Assembly passed a bill approving electronic voting machine (EVM) voting as recently as November 2021. However, the main concerns with digital voting are security, privacy, trust, accuracy, and cost-effectiveness. Citizens worldwide are concerned about election security and integrity, particularly power interference, technical failure or mismanagement, hacking, fraud, forgery and unauthorised voting, as well as the manipulation of results (Kshetri & Voas, 2018). In addition, digital voting systems must comply with national and international standards and regulations. In fact, several countries, including Germany and Netherlands (Loeber, 2014), have abandoned e-voting in elections due to reliability issues, while others still endorse it (Jain, 2019). Certain voting machines are also prone to unexpected and inconsistent errors, making it difficult to ensure the authenticity, transparency and accuracy of results.

Kshetri and Voas proposed a blockchain-enabled e-voting system in 2013 that offers a solution to most of the existing problems of e-voting systems, as data is decentralised and distributed in a database shared by a peer-to-peer network. Every node in the network keeps a copy of voting data and stores it in blocks, which are chained together to make the ledger. The ledger may then be accessed by everyone in the network, thus ensuring the authenticity and transparency of voting data records. Blockchain-based voting may thus reduce fraud and increase accessibility over the Internet through computers and smartphones with encrypted-key and tamper-proof personal identification. Several blockchain-enabled voting systems have since been implemented with a variety of technologies and various degrees of uptake and success in several countries, including India, Russia, Malaysia, Colombia and Pakistan (Jafar,

http://doi.org/10.18080/jtde.v10n4.626

Aziz & Shukur, 2021; Giraldo, Barbosa Milton & Gamboa, 2020; Crowcroft, 2019; Syed *et al.*, 2019; Sherman *et al.*, 2019). In fact, blockchain technology uptake is globally sweeping the financial, business and public administration landscape with a significant scale of adoption in applications at organisations and institutions, in addition to an expectation of a massive $3.1 trillion worth of investments in the technology by 2030 (Madaan, Kumar & Bhushan, 2020).

In this paper, a blockchain voting system is proposed utilising the Ethereum platform, smart contracts, and supported by a decentralised database. The proposed blockchain system may offer citizens the opportunity to vote anonymously and transparently, to keep participants' records, and to ensure fraud-free results. A pilot experiment is designed to simulate an election in Jordan and to test its integrity and transparency with online presence of candidate representatives and worldwide independent observers to monitor the process and endorse its integrity.

Blockchain Technology

Blockchain was first introduced in 2008 as a ledger to execute bitcoin transactions across a distributed network to provide a mechanism for remote nodes to reach consensus on the state of a ledger of information (Hoiss, Seidenfad & Lechner, 2021). Blockchain technology has actually moved over the years from the phase of inception to rapid development and practical applications. A blockchain actually consists of data blocks linked together in a sequential order forming a continuous chain of immutable records, which are permanent and tamperproof. The chain begins with a genesis block that records the first transactions. The block is also assigned an alphanumeric string called a hash, which it uses to create its own hash to link to the next block. Each block is given a number and contains data on transactions and a time stamp of the event, its own hash address and that of the previous block. Blockchain also uses a computational process called consensus to validate a block's authenticity before it can be added to the chain. The nodes on the blockchain network must agree to the hash of the new block by verifying its correct calculation. Consensus ensures that all copies of the distributed ledger are in the same state. Each computer in the network thus maintains a copy of the ledger to avoid a single point of failure (Sheldon, 2021; Jeyasekar, 2020).

Blockchain is actually based on technologies that existed long before bitcoin appeared, such as the Merkle tree, which was proposed in 1979 to provide a data structure for verifying public records and digital signatures, and enabling multiple document certificates to live on a single block. In addition, mutually suspicious groups embody many of the elements of blockchain in the vault system established by David Chaum in 1979 to maintain and trust computer systems (Sherman *et al.*, 2019). Actually, the vault system is a public record-keeping arrangement where group members follow private transactions that protect individual privacy through

http://doi.org/10.18080/jtde.v10n4.626

physical security. The concepts of peer-to-peer (P2P) network and proof-of-work (PoW) to verify computational effort and deter cyberattacks also played an important role in the evolution of blockchain (Vivek *et al.*, 2020). Recently, blockchain was introduced in electronic voting systems due to its attractive feature of end-to-end verification (Hardwick *et al.*, 2018; Awsan & Othman, 2021; Puneet *et al.*, 2021; Anggorojati, 2020).

In 2014, Vitalik Buterin introduced the Ethereum platform as a decentralized open-source application with smart contract functionality, which extended the utilisation of blockchain technology beyond cryptocurrency (Buterin, 2014). Consequently, Ethereum provided developers with a platform for building decentralised applications in almost any field, utilising smart contracts that could be deployed to a live network because it is a secure, immutable, traceable and transparent platform. Ethereum has actually implemented smart contracts and provided developers with a means for application developments in many fields. Industries immediately began to recognise and explore the potential of blockchain and, as of the year 2014, the focus shifted from digital currency to the utilisation and development of blockchain applications beyond the financial landscape. The platform has actually attracted an active developer community that continues to this day. In fact, Ethereum network transactions exceeded 1 million per day in 2019, and consequently the Ethereum Foundation launched the Beacon Chain in preparation for Ethereum 2.0 (Cortes-Goicoechea & Bautista-Gomez, 2021).

The smart contract is a collection of code and data that resides at a specific address with a hash 66 characters long, and lives on the blockchain in an Ethereum-specific binary format called Ethereum Virtual Machine (EVM) bytecode, as shown in Figure 1. One may enter the address into a block explorer; like Etherscan, to see all of the transactions associated with the contract. A contract application binary interface or Arbitrary Binary Interface (ABI) is the standard way to interact with contracts in the Ethereum ecosystem, both from outside the blockchain and for contract-to-contract interaction. The smart contracts are deployed on Ganache, which only executes the byte-code representation. The ABI serves as an interface between two program modules. It defines how data structures and functions are accessed in machine code. When the compilation is successful, the smart contract can be deployed using a particular contract deployment transaction. When a transaction is executed, the contract can be referred to by its address, calculated as a hash function of the originating account and account nonce (number of transactions originated from an account). Consequently, the Ethereum user utilizes the address and ABI to interact with a smart contract. Hashing, a vital feature in blockchain operation, is the cryptographic process of converting an arbitrary input of variable size to an output of fixed size using a complex mathematical algorithm. Ethereum utilizes Keccak-256 hashing in a consensus engine called Ethash. Keccak-256 is part of the Secure Hash Algorithm (SHA)-3 standard released by the US National Institute of Standards and Technology (NIST)

http://doi.org/10.18080/jtde.v10n4.626

in 2015, but with slightly different parameters than the current SHA-3. It generates a cryptographic hash function that yields a 160-bit hash value consisting of 40 hexadecimal characters.

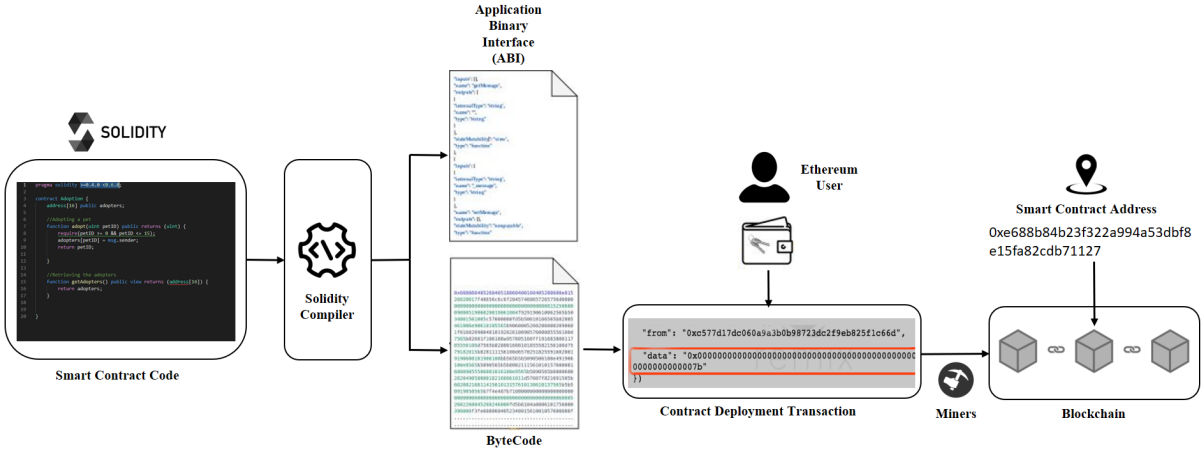

Figure 1. Blockchain Principle of Operation

There are two types of accounts in Ethereum, the first being an externally owned account (EOA), which is identified by a wallet address and controlled by a private key whose holder can transfer Ethers cryptocurrency and sign transactions. EAOs are linked to unique cryptographic key pairs, generated upon account creation. The public key address is used to reference the account, whereas the private key is used to sign a transaction before executing on the network to prove authenticity.

The second Ethereum account is, in fact, the smart contract, which may be considered an account controlled by its own code, or as an autonomous agent executed by the EVM, the core foundation and the main building blocks of any Decentralized Application (DApp). Once this code is deployed on the blockchain, the EVM will take care of running it as long as the conditions apply, and the contract may be publicly visited and viewed via its address with all associated transactions. Triggering functions in the smart contract can be performed from any account, as long as the address of the smart contract is known and the function caller has sufficient Ether to trigger it. A permissioned version of Ethereum exists, in addition to the public one, referred to as a private blockchain. In the public version blockchain, an EOA may send transactions to other addresses in the network using online explorers, such as Etherscan, while a central authority is needed to control and maintain its own ledger in the blockchain. In a country election process, for instance, a permissioned blockchain is usually preferred by governments in order to control the election process.

System Architecture

Ethereum has found popularity as a platform in e-voting systems and many examples have been showcased in the past few years (Khoury *et al.*, 2018; Shukla *et al.*, 2018; Yavuz *et al.*,

http://doi.org/10.18080/jtde.v10n4.626

2018; Rosasooria *et al.*, 2020; Park *et al.*, 2021). These focused on implementing Ethereum blockchain in e-voting systems governed by smart contracts to overcome problems associated with existing digital solution EVMs. Such platforms carry the promise of building trust among citizens on the transparency and openness of the election process, especially with the presence of a third party that monitors the validation, operations and procedures of voting, which is particularly important in emerging democracies.

The proposed architecture of the blockchain system designed to run the voting process consists of four main blocks, namely the client side, front-end web application, blockchain environment and the administrator front-end. The Ethereum platform forms the backbone of the development blockchain environment that includes a Truffle suite (a development environment for smart contracts), as well as InterPlanetary File System (IPFS) and Remix, which are used to test the smart contracts functionality online and consequently host the decentralized web application. The layout of the proposed election landscape is shown in Figure 2, where a country, region, state or city is usually divided into districts or election constituencies, and each accommodates a number of polling stations. Every polling station is connected through the Internet to the blockchain using Web3.js libraries to allow for interaction with an Ethereum node using HTTP, IPC or Web Socket.

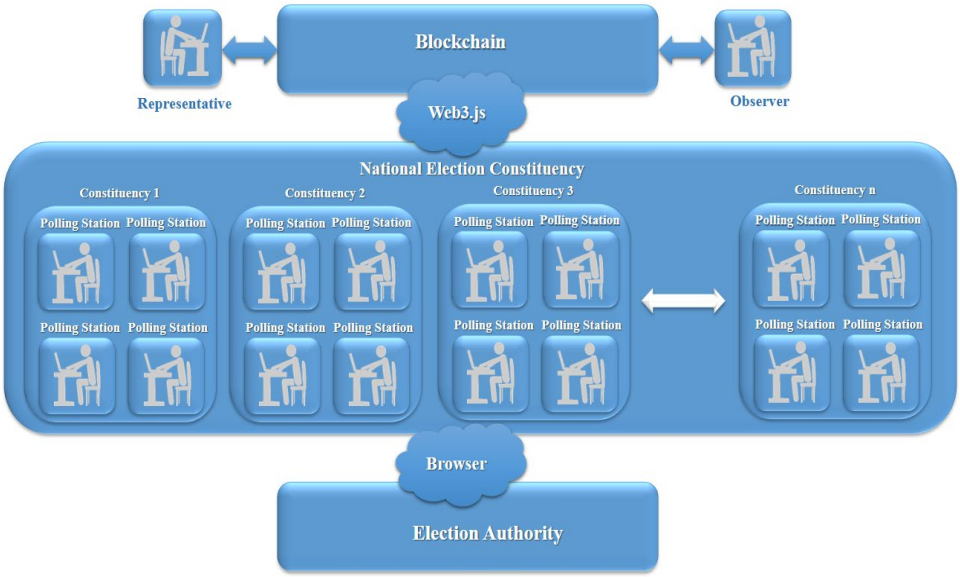

Figure 2. Layout of the Proposed Election Landscape

The voter at a certain polling station may access the voting system via an ordinary web browser by providing a unique national identification number. Once the vote is cast, the smart contract of the polling station automatically assigns the appropriate candidate a new count, and the process is repeated with each new voter in every polling station regardless of the constituency, as they all operate independently from each other using separate distinctive smart contract programming. Each smart contract of polling stations is assigned a transaction key address consisting of 42 characters, with a counter updated every 5 minutes. An observer, whether an

http://doi.org/10.18080/jtde.v10n4.626

international or domestic electoral official assigned by the election authority or a specific representative of a candidate, may follow all transactions of one or more smart contracts directly on the Etherscan platform (https://etherscan.io), which provides real-time display of all transactions on the Ethereum blockchain. By simply selecting and filtering the smart contract address, the observer and representative monitor all transaction details, such as block number, transaction hash, method, age, value and fees. The platform also provides details on internal transactions within the smart contract, tokens and Ether transfers, analytics and comments. The same principle applies to any type of election, whether presidential, congressional, parliamentary, state assembly, municipality, or even limited scale student union elections at universities.

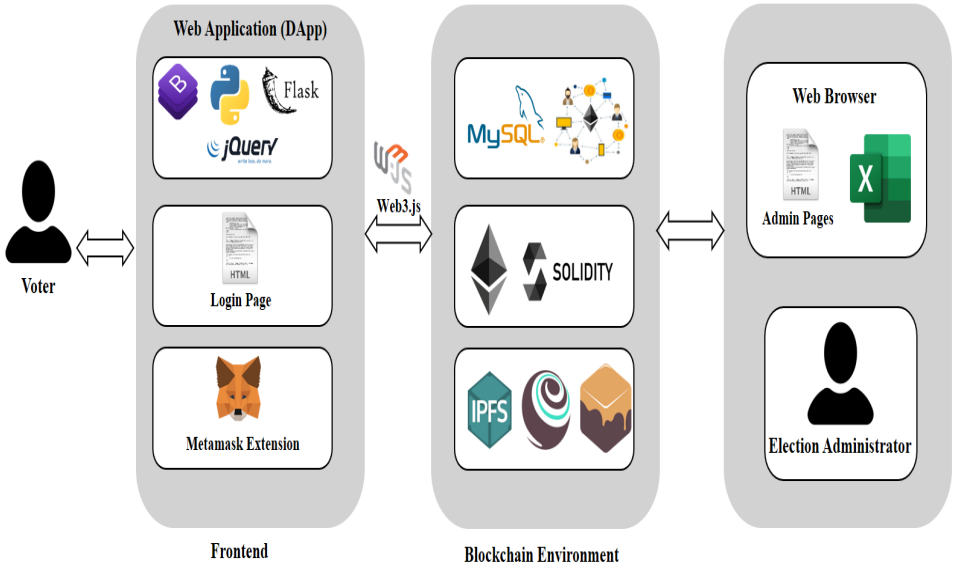

Figure 3. Blockchain Voting System Architecture Overview

In order to prepare the blockchain environment for the election process, as shown in Figure 3, Truffle is utilised to compile the smart contract, which is a set of protocols coded in Solidity to coordinate communication and to organize the flow of decisions amongst partners in the network (Ibrahim *et al.*, 2021). The Truffle framework actually allows migrating the smart contract to the local Ethereum blockchain utilizing default package manager for the JavaScript runtime environment Node.js (NPM). The smart contract necessary details and information, including its variables and functions, is migrated onto the blockchain in the form of an ABI file because of compilation. In fact, the ABI file helps connect the local Ethereum blockchain to the front end at the server side using web3.js. Simultaneously, Ganache, which is a local blockchain for rapid Ethereum distributed application development, is used across the entire development cycle to develop, deploy, and test decentralized applications (DApps) in a safe environment. Ganache actually initiates the local blockchain and provides the user with 10 accounts, each of which has a unique address that represents voters' access details in the actual election application. On the client side, the user and admin interfaces are hosted on the

http://doi.org/10.18080/jtde.v10n4.626

election authority server using different programming languages, like HTML, Python, and JavaScript, while formatting is performed with a cascading style sheet (CSS). The user can access these pages using a web browser, supported by a MetaMask functionality, to connect to the Ethereum platform. Meanwhile, the admin interface is responsible for management of policies and rules, including uploading a candidate list Excel file to the smart contract to be stored in mapping arrays, in addition to assuming authority to view the results on the output pages. Furthermore, the client user interface utilises Ethereum accounts to invoke the smart contracts and then generates a dropdown menu form to select the candidate.

Election authorities usually initiate the voter registration stage. When an election starts, these appropriate authorities define a trusted list of individuals who are eligible to vote. This might require a database for an identity verification service to securely authenticate and authorize eligible voters by the Civil Authority Department. Using such a service is necessary for the requirement of secure identity verification and authentication: by default, when using a blockchain platform, for each eligible voter, a unique identity wallet is generated. The voter's registration process and identity verification are executed in advance prior to creating accounts. The ID card is used to verify the voter encrypted information. Consequently, the voter enters the application website with access details, including the national number that every citizen in the country is assigned. MetaMask will then ask the user to provide access information, including the private key generated by Ganache, and hence open the voting page in the form of the dropdown menu. Once the voting task is completed, MetaMask generates a sequence to move an Ether coin from the user to the smart contract address. Casting votes is performed by the smart contract that contains a function that verifies the authenticity of the voting, and then a vote count is incremented.

Piloting the System

The flowchart in Figure 4 highlights the process of the voting application, which commences in initiating the blockchain environment to run locally on the desktop. The first step in the process is to read the data of the voting population from the server of the election authority by either uploading an Excel sheet containing all necessary details of each citizen, or reading it online directly by web service. In fact, election authorities should prepare the complete lists of voters months prior to election time. The voters' information includes details such as name, identification national number, sex, age, city and district. The list is uploaded to a secure MySQL database created by the admin at the election authority domain. An initial identity verification step is then launched based on the national identity number to check if the voter is included in the citizens list. A smaller candidate list is also created for each district

http://doi.org/10.18080/jtde.v10n4.626

containing all competing individuals or parties according to the election registration mechanisms.

The blockchain environment is launched using the Truffle framework to interact with the local Ethereum network and deploy smart contracts onto the blockchain. Ganache, meanwhile, creates the 10 accounts in the trial version, in the form of a public hash key containing 42 hex characters, and a private hash key of 66 hex characters. The public key is assigned to the voter as an anonymous identity equivalent to the national identity number, while the private key is used to access the system the same way as a password does. The voter then receives an amount of the Ethereum currency in the form of a gas coin. The account then enables the voter to connect to the Ethereum platform, via MetaMask extension within the browser in the polling station, with the private key as the identifier. The voter is considered eligible if found in the citizens list; otherwise an error message is displayed.

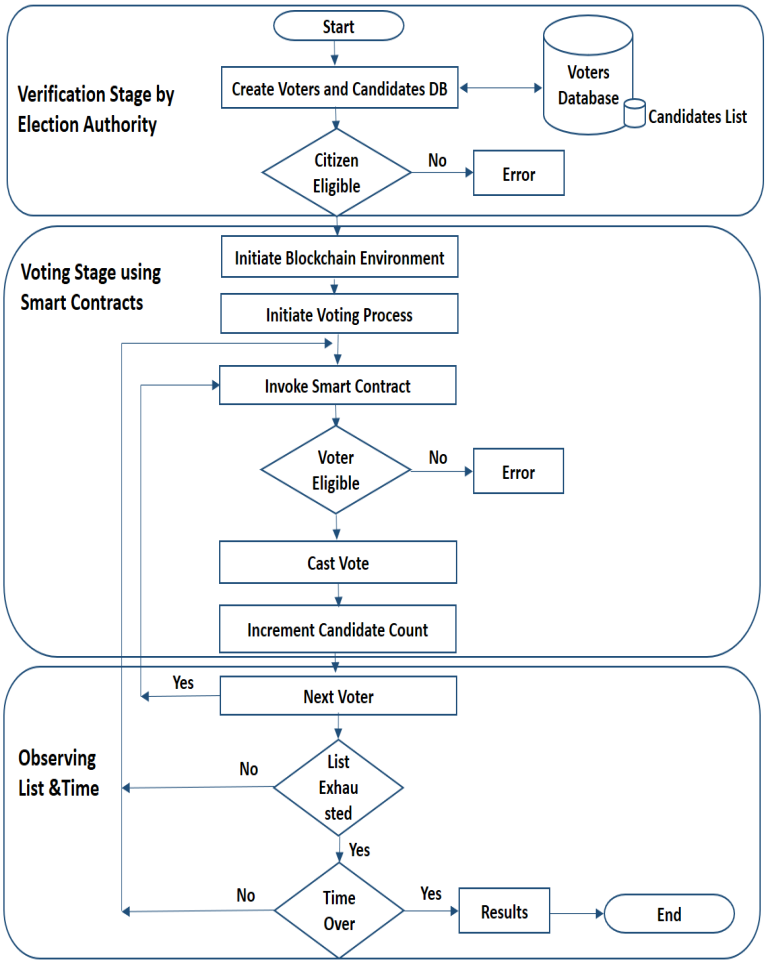

Figure 4. Flowchart of e-Voting System

The smart contract will also create a mapping array of the candidate list and stores it within its memory space, as shown in the example depicted in the screenshot of Figure 5. The details stored in the list pertain to the candidate data that consist of the names, IDs and vote counts. Once the voter accesses the blockchain election page in the user client side using the private

http://doi.org/10.18080/jtde.v10n4.626

key, the smart contract will check the eligibility criteria, and whether the vote has been cast. The voter then casts the vote and the smart contract approves the transaction and increments the count for the corresponding candidate. The smart contract then waits for the next voter, repeats the process and checks if the list of voters is exhausted, and finally inspects if the time allocated for the election is over. Once over, the mapping array accumulates the results for each district, which may then be transferred to the IPFS location for proper recording and archiving using smart contract events that may be utilised to communicate back with the admin who invoked the contract in the first place.

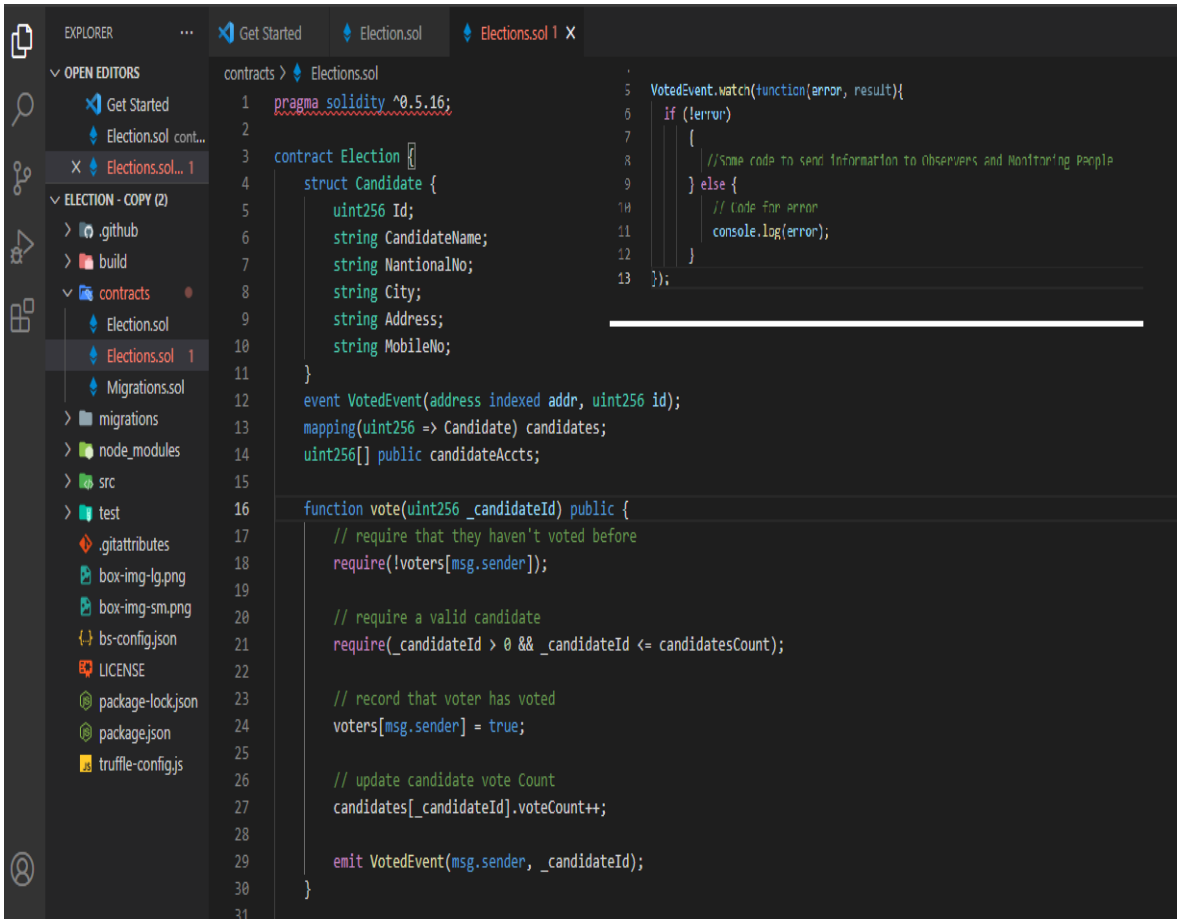

Figure 5. Solidity Code of the Smart Contract

In fact, events facilitate communication with the client through web3.js to obtain the value returned by the function to display in the user interface instead of the hash of the transaction, while the data passed is made available at the client side. In addition, every time an event is emitted, the data within the event is written into the blockchain logs, which are kept as a record of everything that happens within the contract, including the voter details and the candidates' vote count. The main features of events thus include the ability to log information and to trigger actions and return values to the invoking client. The example of the Solidity smart contract depicted shows the process when the Vote Function is invoked, in line 16. The event VotedEvent in line 12 is emitted, which returns the value of the voter account and candidate

http://doi.org/10.18080/jtde.v10n4.626

ID defined in the struct candidate in line 4, back to the invoker and simultaneously logs both his/her address and value in the blockchain.

Transactions performed through Ethereum blockchain smart contract are grouped into blocks connected by a chain. A new block is not created until the previous block is completed and this is the vital step in the process as shown in Figure 6. The blocks are ordered chronologically, and each block contains a cryptographic hash of the previous block and applying the SHA-256 algorithm.

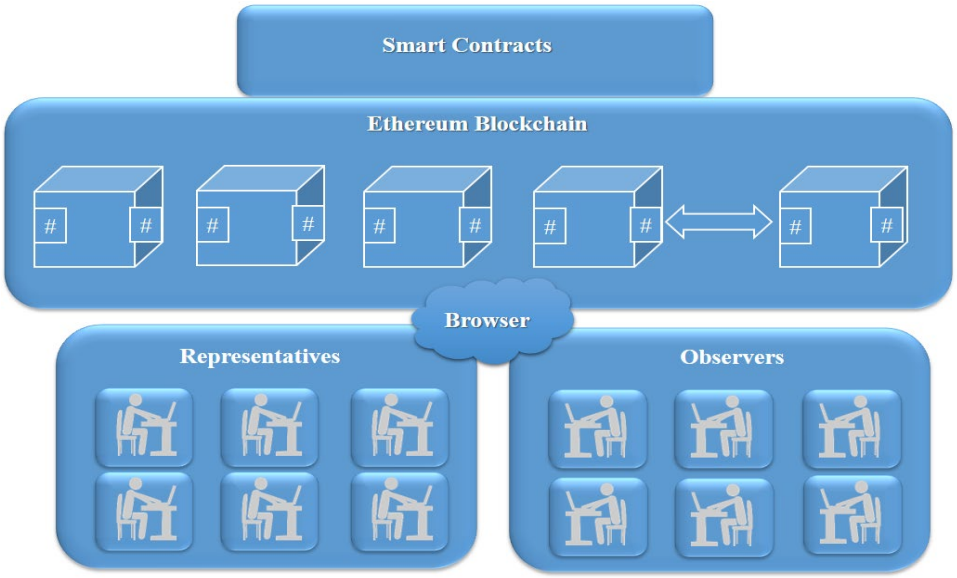

Figure 6. Observers' Role in the Blockchain Voting System

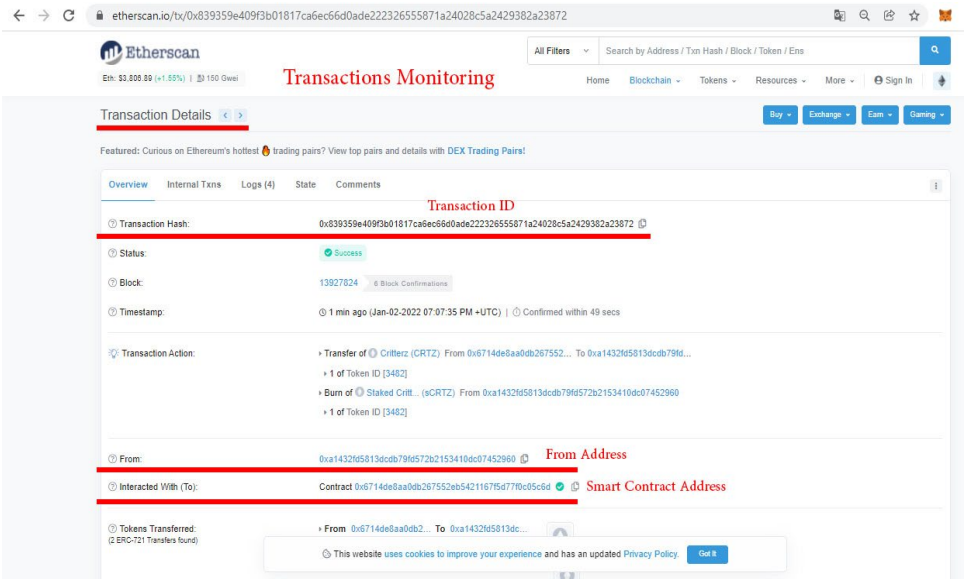

Figure 7. Etherscan View of Observers and Delegates

Observers assigned by the election authority and representatives of each candidate can subsequently monitor the election process by following the smart contract address in the Ethereum blockchain explorer at https://etherscan.io/, as shown in Figure 7. For each district, a vote is considered as a transaction by the smart contract. By following the transactions page,

http://doi.org/10.18080/jtde.v10n4.626

which contains data such as transaction hash, method, block, age, value fee and sending end, the observers and representatives can keep a close eye on the process and take note of any irregularities that might occur. Observers and representatives may then report irregularities to the election authority to take appropriate action immediately by checking all transactions made at the concerned smart contract address.

Every user, however, can trace any transaction performed by voters given their address in the Etherscan. The privacy of the voting process may only be compromised if unauthorised personnel have access to the source code, or the ABI file is decompiled and the bytecode of the smart contract in the Etherscan is revealed. In this case, an additional cryptographic technique may be required if the secret ballot voting principle is compromised using the public Ethereum blockchain network.

Jordan Case Study

The political system in Jordan is parliamentary with a hereditary monarchy, as stated in the first article of the Constitution. The parliamentary system adopted is a bicameral national assembly of a Senate appointed by the King and a House of Representatives elected by citizens. Elections of various types and for different offices have been part of the political and public life in Jordan since 1929, when the first general election was conducted during the era of Transjordan. Local, municipality, decentralization, unions, and other forms of elections have continued ever since; however, parliamentary elections were halted after periods of turbulence and unrest in the region until 1989. An independent election commission (IEC) oversees all public elections and is the authority responsible for administrating general and local polling processes. The IEC, in cooperation with the civil status department, prepares complete lists of voters in all districts and constituencies, months prior to election time. The commission may also seek the assistance of international observers to ensure accountable electoral management. This is a vital step to maintaining the credibility of elections and the transparency of electoral administration. The IEC thus follows specific executive instructions for the accreditation of international observers for the elections of parliament, as well as governorate and municipal councils, in order to provide an impartial and accurate assessment of the nature of election processes. International observers follow a detailed “code of conduct” to ensure that they respect the sovereignty of the host country, as well as the human rights and fundamental freedoms of its people, in addition to respecting the laws, authorities and bodies in charge of administering the electoral process.

Furthermore, the IEC reserves the power to approve the delegates or representative lists for individual candidates and political parties at the polling and counting centres for the general parliamentary and other elections. Delegates may enter the polling and counting centres and

http://doi.org/10.18080/jtde.v10n4.626

monitor the process, with only one delegate assigned to each polling room, while the candidate has the right to monitor all polling stations and the results extraction centre, which is limited to the candidate without delegates. The IEC issues accreditation cards to delegates and publishes the names on the IEC website. In fact, the IEC administered a transparent voting process for the 2020 parliamentary elections and mobilized over 12,000 youths as volunteers, implemented COVID-19 health measures protecting voter safety, and provided accommodation to ensure equal access to polling stations for persons with disabilities in all municipalities, including 12 pilot highly accessible polling stations.

In 2016, the government introduced a smart ID card, containing information that includes 18 data fields, with the name in Arabic and English, gender, place of birth, area of residence, blood type, and a distinct citizen number consisting of 10 digits in use since 1992. The smart card embeds a chip that stores biometric data such as an iris scan and fingerprint, and designed to accommodate data on health insurance, tax, pension, and voting at later stages. The new ID card, in a credit card format, will reinforce the infrastructure required for digital signatures and make it possible to introduce new online services. The creation of the ID card has actually been a top priority of the country's e-government program as a reliable online infrastructure for access to present and future e-services. The e-government program has in fact been launched in 2002 to improve service delivery and increase the involvement of citizens with ICT, and consequently create a foundation for an e-voting system.

Figure 8. Interface of the Voter Screen

The proposed blockchain-based voting system is prototyped in one constituency for fictitious parliamentary elections in Jordan, which consists, according to the present election law, of 12 governorates, each allocated a certain number of seats, including gender, ethnic minority and special quotas. A number of seats are also allocated to a national list, totalling 130 parliamentary seats. The prototype is depicted for third district in the capital, selected as a

http://doi.org/10.18080/jtde.v10n4.626

proof of concept. The demo page of the voter interface is shown in Figure 8, where the national ID is requested and entered. Access is then granted to the blockchain through the public key hash provided by the system. Figure 9 shows a snapshot of possible results for all candidates in that particular constituency.

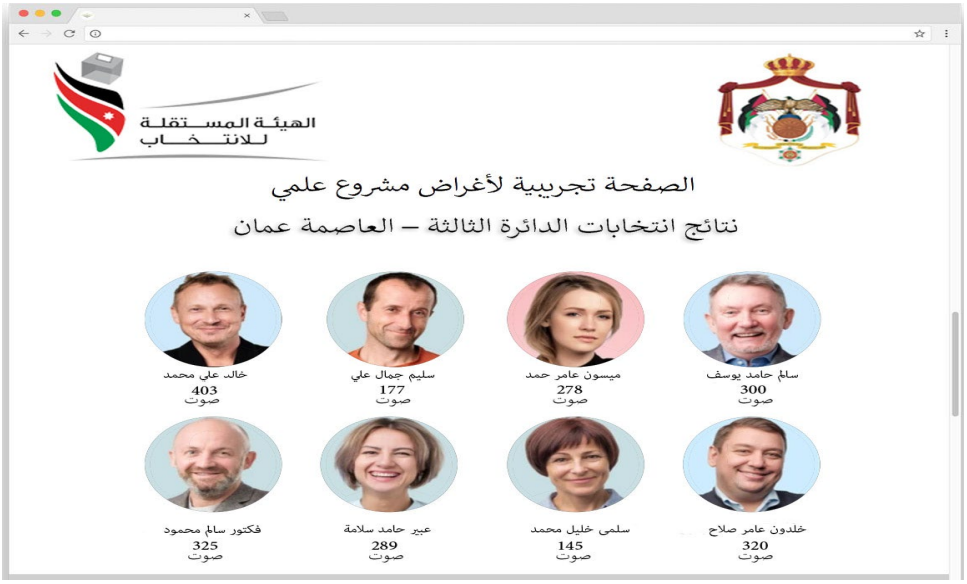

Figure 9. Interface of the Election Results

Almost all usual concerns regarding vote integrity and transparency have been raised, including the issue of assuring that votes have been counted as they were cast and not undetectably altered or discarded. However, a blockchain voting system running on a centralized network node with integrated biometric scanner has recently been developed in order to address the integrity of voters. This scheme allows data immutability while providing the user with security and control over their ballot. Recently, a blockchain based voting system (BBVS) has been proposed based on a private and centralized blockchain, using a Java development platform, implemented in a simulated environment, and then applied to parliamentary elections in Jordan. The BBVS applied a hierarchical election process, where a vote is cast at two levels, one for a group and the second for specific members within the group. A new algorithm has been introduced to maintain acceptable performance, both at the time of creating the blockchain for voters and candidates and at the time of vote casting (Malkawi, Yassein & Bataineh, 2021). Further research should address the challenges that blockchain brings in security, privacy, scalability and interoperability. Naturally, blockchain may not be appropriate for every application, and designers must evaluate its suitability before investing in its production.

Conclusions

A pilot model for a proposed e-voting system based on blockchain technology, which runs on the Ethereum platform, utilises IPFS for data storage and manipulation, and deploys an

http://doi.org/10.18080/jtde.v10n4.626

associated identity verification process, has been successfully implemented. The concept of smart contracts made programming the blockchain a smooth process that overcomes many of the limitations of conventional e-voting, such as a lack of transparency, security, trust or accuracy. The system may thus help in providing a reliable and secure voting process while reducing cost, saving time, and preserving the integrity of the election, whether it is local, regional or national, and regardless of its nature, being parliamentary, local authority, NGO or even private corporation. The system also promotes transparent democracy, which enables voters to easily cast their ballots from anywhere and consequently validate the final count.

The proposed model may be integrated further with identity authentication utilising artificial intelligence and machine learning algorithms for facial recognition, iris scanning or fingerprints. The advantage of the proposed system is its independence from traditional third-party involvement while maintaining integrity and transparency. Further improvements may be made to make the system versatile for elections at a large scale by integrating advanced identity authentication. The e-voting protocols may be improved further using different blockchain frameworks as well as real-time testing for large numbers of voters, in addition to increasing their confidence in the system.

References

Abuidris, Y., Hassan, A., Hadabi, A., & Elfadul, I. (2019). Risks and Opportunities of Blockchain Based on E-Voting Systems. 16th International Computer Conference on Wavelet Active Media Technology and Information Processing. https://doi.org/10.1109/ICCWAMTIP47768.2019.

Anggorojati, D. P. (2020). Implementation and Evaluation of Blockchain Based e-Voting System with Ethereum and Metamask. International Conference on Informatics, Multimedia, Cyber and Information System (ICIMCIS). https://doi.org/10.1109/ICIMCIS51567.2020.9354310

Awsan, A. H., & Othman, E. A. (2021). Online Voting System Based on IoT and Ethereum Blockchain. International Conference of Technology, Science and Administration (ICTSA). https://doi.org/10.1109/ICTSA52017.2021.9406528

Buterin, V. (2014). A Next-Generation Smart Contract and Decentralized Application Platform. Ethereum White Paper *3*(37), 2–1. Retrieved from https://scholar.google.co.kr/citations?view_op=list_works&hl=en&hl=en&user=DLP9gTAAAAAJ

Cortes-Goicoechea, M., & Bautista-Gomez, L. (2021). Discovering the Ethereum2 P2P Network. 3rd Conference on Blockchain Research & Applications for Innovative Networks and Services (BRAINS), 1–2. https://doi.org/10.1109/BRAINS52497.2021.9569801

Crowcroft, B. S. (2019). Trustworthy Electronic Voting Using Adjusted Blockchain Technology. *IEEE Access*, *7*, 24477-24488. https://doi.org/10.1109/ACCESS.2019.2895670

Giraldo, F. D., Barbosa Milton, C., Gamboa, C. E. (2020). Electronic Voting Using Blockchain and Smart Contracts: Proof of Concept. *IEEE Latin America Transactions*, *18*(10), 1743–1751. https://doi.org/10.1109/TLA.2020.9387645

Hardwick, F. S., Gioulis, A., Akram, R. N., & Markantonakis, K. (2018). E-Voting with Blockchain: An E-Voting Protocol with Decentralisation and Voter Privacy. IEEE International Conference on Internet of Things, IEEE Green Computing, Communications and IEEE Cyber, Physical and Social Computing and IEEE Smart Data. https://doi.org/10.1109/Cybermatics_2018.2018.00262.

Hoiss, T., Seidenfad, K., & Lechner, U. (2021). Blockchain Service Operations-A Structured Approach to Operate a Blockchain Solution. IEEE International Conference on Decentralized Applications and Infrastructures (DAPPS) UK (Online). https://doi.org/10.1109/DAPPS52256.2021

Ibrahim, M., Ravidran, K., Lee, H., Farooqui, O., & Mahmoud, Q. H. (2021). ElectionBlock: An Electronic Voting System using Blockchain and Fingerprint Authentication. IEEE 18th International Conference on Software Architecture Companion (ICSA-C). https://doi.org/10.1109/ICSA-C52384.2021.00033

Jafar, U., Aziz, M J. A., & Shukur, Z. (2021). Blockchain for Electronic Voting System—Review and Open Research Challenges. *Sensors*, *21*(17), 1–22. https://doi.org/10.3390/s21175874

Jain, K. P. (2019). Decentralized E-Voting Portal Using Blockchain. 10th International Conference on Computing, Communication and Networking Technologies (ICCCNT). https://doi.org/10.1109/ICCCNT45670.2019.8944820

Jeyasekar, S. K. (2020). A Competent and Accurate Blockchain based E-Voting System on Liquid Democracy. Second Conference on Blockchain Research and Applications for Innovative Networks and Services (BRAINS), 202–203. https://doi.org/10.1109/BRAINS49436.2020.9223308

Jones, D. W. (2009). Early Requirements for Mechanical Voting Systems. First International Workshop on Requirements Engineering for e-Voting Systems. https://doi.org/10.1109/RE-VOTE.2009.3

Khoury, D., Kfoury, E. F., Kassem, A., & Harb, H. (2018). Decentralized Voting Platform Based on Ethereum Blockchain. 2018 IEEE International Multidisciplinary Conference on Engineering Technology (IMCET). https://doi.org/10.1109/IMCET.2018.8603050

Kshetri, N., & Voas, J. (2018). Blockchain-Enabled E-Voting. *IEEE Software*, *35*(4), 95–99. https://doi.org/10.1109/MS.2018.2801546

Loeber, L. (2014). E voting in the Netherlands; past, current, future? Journal Proceedings of the sixth international conference on electronic voting (EVOTE), 43–46.

Madaan, L., Kumar, A., & Bhushan, B. (2020). Working Principle, Application Areas and Challenges for Blockchain Technology. IEEE 9th International Conference on Communication Systems and Network Technologies (CSNT). https://doi.org/10.1109/CSNT48778.2020.9115794

Malkawi, M., Yassein, M. B., & Bataineh, A. (2021). Blockchain Based Voting System for Jordan Parliament Elections. *International Journal of Electrical and Computer Engineering*, *11*(5), 4325–4335. http://doi.org/10.11591/ijece.v11i5.pp4325-4335

Park, S., Specter, M., Narula, N., & Rivest, R. L. (2021). Going from bad to worse: from Internet voting to blockchain voting. *Journal of Cybersecurity*, *7*(1). https://doi.org/10.1093/cybsec/tyaa025

Puneet, Chaudhary, A., Chauhan, N., & Kumar, A. (2021). Decentralized Voting Platform based on Ethereum Blockchain. International Conference on Advances in Electrical, Computing, Communication and Sustainable Technologies (ICAECT). https://doi.org/10.1109/ICAECT49130.2021.9392580

Rosasooria, Y., Mahamad, A. K., Saon, S., Isa, M A. M., Yamaguchi, S., & Ahmadon, M. A. (2020). E-Voting on Blockchain using Solidity Language. 3rd International Conference on Vocational Education and Electrical Engineering (ICVEE). https://doi.org/10.1109/ICVEE50212.2020.9243267

Sheldon, R. (2021). A Timeline and History of Blockchain Technology. Retrieved from https://whatis.techtarget.com/feature/A-timeline-and-history-of-blockchain-technology

Sherman, A. T., Javani, F., Zhang, H., & Golaszewski. E. (2019). On the Origins and Variations of Blockchain Technologies. *IEEE Security and Privacy*, *17*(1), 72–77. https://doi.org/10.1109/MSEC.2019.2893730

Shukla, S., Thasmiya, A. N., Shashank, D. O., & Mamatha, H. R. (2018). Online Voting Application Using Ethereum Blockchain. (2018). International Conference on Advances in Computing, Communications and Informatics (ICACCI). https://doi.org/10.1109/ICACCI.2018.8554652.

Sliusar, V., Fyodorov, A., Volkov, A., Fyodorov, P., & Pascari, V. (2021). Blockchain Technology Application for Electronic Voting Systems. IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (ElConRus), 2257–2261. https://doi.org/10.1109/ElConRus51938.2021.9396400

Stein, R., & Wenda, G. (2014). The Council of Europe and E-voting: History and impact of Rec (2004)11. 6th International Conference on Electronic Voting: Verifying the Vote (EVOTE). https://doi.org/10.1109/EVOTE.2014.7001139.

Syed, T. A., Alzahrani, A., Jan, S., Siddiqui, M. S., Nadeem, A., & Alghamdi, T. (2019). A Comparative Analysis of Blockchain Architecture and its Applications: Problems and Recommendations. *IEEE Access*, *7*, 176838–176869. https://doi.org/10.1109/ACCESS.2019.2957660

Tsahkna, A.-G. (2013). E-voting: Lessons from Estonia. *European View*, *12*(1), 59–66. https://doi.org/10.1007/s12290-013-0261-7

Vivek, S. K., Yashank, R. S., Prashanth, Y., Yashas, N., Namratha, M. (2020). E-Voting Systems Using Blockchain: An Exploratory Literature Survey. Second International Conference on Inventive Research in Computing Applications (ICIRCA). https://doi.org/10.1109/ICIRCA48905.2020.9183185

Yavuz, E., Koç, A. K., Çabuk, U. C., & Dalkılıç, G. (2018). Towards Secure E-Voting Using Ethereum Blockchain. 6th International Symposium on Digital Forensic and Security (ISDFS). https://doi.org/10.1109/ISDFS.2018.8355340