RectorDApp: Decentralized Application for Managing University Rector Elections

Jesús Rosa-Bilbao
UCASE Software Engineering Research Group
School of Engineering
University of Cadiz
Puerto Real, Cádiz, Spain
jesus.rosabilbao@alum.uca.es, 0000-0002-4378-5229

Juan Boubeta-Puig
UCASE Software Engineering Research Group
Department of Computer Science and Engineering
University of Cadiz
Puerto Real, Cádiz, Spain
juan.boubeta@uca.es, 0000-0002-8989-7509

2021 IEEE International Conference on Service-Oriented System Engineering (SOSE) | 978-1-6654-3477-5/21/$31.00 ©2021 IEEE | DOI: 10.1109/SOSE52839.2021.00024

***Abstract*—Blockchain is a distributed and secure database that can be applied to all types of transactions. Blockchain technology is growing in popularity because it allows for the development of applications whose information is traceable, immutable, transparent and reliable. Given the advantages that blockchain provides over other traditional systems, in this paper we present a decentralized application, called RectorDApp, for the management of university rector voting in a private, but transparent and immutable way, being able to verify publicly and in real time the election results. RectorDApp, capable of interacting with the Ethereum public blockchain network, was developed using the Truffle framework and the MetaMask software. The results demonstrate that RectorDApp is a highly useful application to address the digital transformation of university rector elections.**

***Index Terms*—Blockchain, DApp, Digital transformation, Smart Contract, University, Voting**

I. Introduction

The right to vote is universal and in most countries there is no obligation to exercise it or to disclose the vote since it is considered secret. This right is regulated by the laws of each country that establish the different types and forms of voting. As an example, the Uruguayan Constitution states that voting is a duty as a citizen and an obligation. Usually, any person, who possess the nationality and is registered in the electoral roll, may vote; however, in some countries, those who have been convicted of any crime do not have this right [1].

Nevertheless, the evolution of voting management systems, from the first elections to the most recent ones, is not comparable to other systems developed in other industrial sectors, where technology has advanced rapidly. In recent times, the greatest advance made with respect to voting has been the possibility of requesting a postal vote, which is becoming increasingly successful. Additionally, no solution has ever been proposed for the management of invalid votes, which are those that are wrongly cast in a given election and are, therefore, null.

The aim of this paper is the application of blockchain technology to voting systems, proposing an alternative through a modern and telematic approach so that any authorized person can vote from anywhere without the need to travel and eliminating the typology of the invalid vote. This is because a person who has the right to vote, and decides to exercise it, will vote for one of the available candidates or blank, but it can never be considered a null vote, since there is no evidence of the invalidity of the vote.

Specifically, this proposal is focused on a voting system for the election of a university rector that has some particular characteristics with respect to a general voting system, such as the type of staff to which the voter belongs (research teaching, student, administration and services, etc.) and the percentage that their vote represents. Therefore, a Decentralized Application (DApp), called RectorDApp, has been developed for the management of university rector voting in a private way, but transparent and immutable at the same time, being able to verify publicly and in real time the election results.

Therefore, we also aim to answer the following Research Questions (RQs):

- **RQ1**: Are blockchain-based systems appropriated to ensure data traceability, immutability and transparency in a user-friendly way?
- **RQ2**: Is blockchain suitable for running an electronic voting system?
- **RQ3**: What advantages does the use of the blockchain technology provide to electronic voting systems?

The rest of the paper is organized as follows. Section II describes the background of blockchain. Section III presents RectorDApp, our DApp for rector voting management. Section IV presents a case study where our proposal has been applied. Section VI describes the work related to RectorDApp. Finally, Section VII presents conclusions and lines of future work.

II. Background

This section describes the blockchain technology as well as the frameworks used in this work.

A. Blockchain

Blockchain [2] can be defined as a distributed database in which transactions are registered and confirmed. Each trans-

This work was supported by the Spanish Ministry of Science and Innovation and the European Regional Development Funds under project FAME [grant number RTI2018-093608-B-C33].

2642-6587/21/$31.00 ©2021 IEEE
DOI 10.1109/SOSE52839.2021.00024

action executed has to be verified, registered and combined with other transactions to produce new blocks that will then be copied in the same nodes of the participating network, thus creating a type of distributed network [3].

A blockchain network involves a series of phases to create and secure blocks. These networks must perform a set of tasks, which are as follows [4]:

- Collecting and computing data in blocks.
- Using cryptography as a means of securely unifying blocks.
- Sharing blocks with all peers belonging to the network.
- Validating and authenticating blocks.
- Maintaining consensus among the different parties that belong to the network.

When a transaction is finalized within the blockchain network, the information is recorded and shared with all users of the participating network. These records form blocks that are time-stamped to organize them in a sequential order, thus avoiding duplication or errors, since these blocks are immutable. Any member who wants to query the history of the blockchain network will obtain the same transaction history and in the same order.

In a blockchain network, there is a fundamental element which is the hash function [4]. This function is a cryptographic algorithm that makes the link between the different blocks of the blockchain network impossible to break, since it uses the information of each block to create a unique string of characters for each block, identifying it univocally.

Each block, therefore, has a hash that is added to the data of the next block, i.e., when a new block is formed, it includes the hash of the previous block, among other information. Thanks to this, if a block is manipulated at any time, it will cause the entire block chain to change, thus showing evidence that it has been manipulated. This process is repeated for every block that makes up the blockchain network.

Currently, there are several platforms based on blockchain technology [5]. Because of the characteristics of our proposal, we have chosen Ethereum [6], a blockchain platform used to create open source DApps.

One of the most important components of a blockchain network is smart contracts. A smart contract is an agreement that describes how a transaction is to be carried out between various agents in the network. They can be executed to deal with information that is transmitted or has been extracted from some interconnected systems.

Blockchain technology seeks a more equitable approach, thus promoting collaboration. Ethereum allows us to write business logic and agreements in the form of smart contracts, which will be automatically executed when their conditions are satisfied by both parties. Moreover, thanks to smart contracts, we can automate certain processes in a secure way.

These smart contracts are implemented using high-level programming languages. As an example, Solidity [7] is one of the languages supported by Ethereum.

B. Truffle

Truffle [8] is a development environment that provides resources to interact with the Ethereum blockchain network. It also enables the compilation, linking and implementation of smart contracts.

It allows for the creation of scripts for automated testing of smart contracts. It also has an interactive console for direct communication with smart contracts.

C. MetaMask

MetaMask is a software that allows us to run Ethereum DApps directly in our browser without the need to run a full Ethereum network. It consists of a user interface where we can own multiple accounts and sign blockchain transactions [9].

Regarding security, we could say that, so far, MetaMask is quite secure, as it has not suffered any damage or attacks that have resulted in cryptocurrency theft. It uses a security system that keeps all local keys encrypted, so that even the DApps themselves cannot access them.

MetaMask has a backup system that is based on 12 words that are in a specific order. In case an account is stolen, these words are the ones that would give us the opportunity to recover it. In addition, it has the support of the entire community of developers who constantly update the source code at the slightest error detected [10].

III. Our Proposal

In this section, we present our approach (see Fig. 1), which interacts with the Ethereum network and which RectorDApp is based on. The approach is composed of two layers. The design time layer includes the process that will take place at design time (left-hand side of Fig. 1) while the runtime layer covers the process to be executed at runtime (right-hand side of Fig. 1).

Following the numbering sequence in Fig. 1, the steps involved from the definition of the smart contract until the end user interacts with it through the web application are explained below:

1) First of all, the logic necessary for the DApp is defined. The smart contract can be defined with the Solidity programming language.
2) When the contract has been defined, the contract is syntactically validated. For this purpose, the Truffle framework is used.
3) The migration process consists of transforming the smart contract implemented in Solidity into a kind of interface. Moreover, in this step the smart contract is deployed in the Ethereum network. Just before the deployment of the smart contract, and at design time, is when the candidates to be part of the voting are defined. Internally, the smart contract will create an extra “candidate” who will be the recipient of the blank votes.
4) To support the DApp in a user-friendly way, the deployment of a web application is conducted. This web application, through the elements arranged in it, will be in charge of invoking the specific functions of the

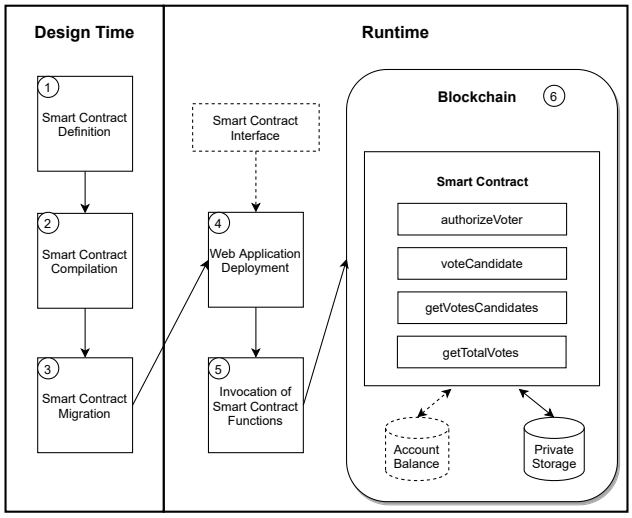

Fig. 1. Overview of the proposed approach

previously deployed contract, such as authorizing a person to vote, viewing the votes in real time or simply voting.

5) Once the user has chosen the operation to be performed, the DApp is responsible for automatically invoking the necessary smart contract functions.
6) When the functions have been executed by the DApp, they are validated and executed on the Ethereum blockchain network.

IV. Case Study

This section describes the application of our approach (see Section III) for the management of university rector elections.

A. Description

This case study is based on the elections for rector of the University of Cadiz (UCA) that took place in 2019 [11]. Next, we will analyze some of the most relevant aspects of the smart contract that we have implemented in this work to provide RectorDApp with the business logic.

The smart contract mainly considers an administrator —in our case, it would be the UCA— that plays the role of contract owner. This administrator is the only responsible entity able to authorize which people will be able to participate in the elections. In order to authorize a person to participate in the elections, the following information is required:

- The public address of the person to be authorized.
- The weight of a person's vote, which depends on the university staff type to which he/she belongs (see Table I).
- The working place of that person (census), i.e., in case the elections were face-to-face, the place where he/she would be entitled to vote. Although the census is not relevant for electronic voting, it will allow us to obtain voting statistics according to voter's census.

Table I shows the abbreviations of UCA staff that have the possibility to vote in the elections:

TABLE I
Vote weightings for UCA staff types

UCA staff	Percentage of vote
PDVP	53%
ALU	28%
PAS	8%
PDINVP	6%
PNDVP	5%

- **PDVP**: Tenured Doctoral Professors.
- **PNDVP**: Tenured Teachers.
- **PDINVP**: Non-tenure Research/Teaching Staff.
- **PAS**: Administration and Services Staff.
- **ALU**: Students Staff.

It is worth nothing that this information can be found in more detail from the provisional results of the second round of the UCA elections published in [12].

Therefore, only users authorized to vote in the elections will be able to do it. Likewise, all users, whether authorized or not, will be able to query in real time the partial or total results of the voting.

B. Implementation

As previously mentioned in Section III, we used Truffle in order to develop and deploy the web application that interacts with the blockchain network. In addition, the Solidity language was used to create the smart contract. This is where all the logic necessary for the correct operation of the contract was implemented. For instance, only the administrator can control which people have the permissions to vote.

Due to the inherent limitations of the Solidity language, e.g. it does not allow to declare variables with decimals, the percentages specified in Table I will be represented by points over 100 in our DApp. That is, if a student has a 28% voting percentage, 1 student vote will be worth 28 points.

Fig. 2 shows the main view of RectorDApp, our application developed for the management of rector voting. We can observe through this web application that the contract owner can authorize a person by typing the requested data (see Subsection IV-A). Once the person is authorized, he/she can choose the option to vote for the desired candidate or to vote blank, as shown in the lower view of Fig. 2.

Let's consider a concrete example of a user voting. Firstly, a person is authorized to vote by the administrator. Secondly, this person casts his/her vote by using the web application. Finally, the vote is correctly registered in the blockchain, as illustrated in Fig. 3.

V. Discussion

This section discusses the answers to the RQs stated in Section I:

- **RQ1: Are blockchain-based systems appropriated to ensure data traceability, immutability and transparency in a user-friendly way?**
  Yes, as we have seen, our RectorDApp proposal uses blockchain technology [13]. Thanks to the use of

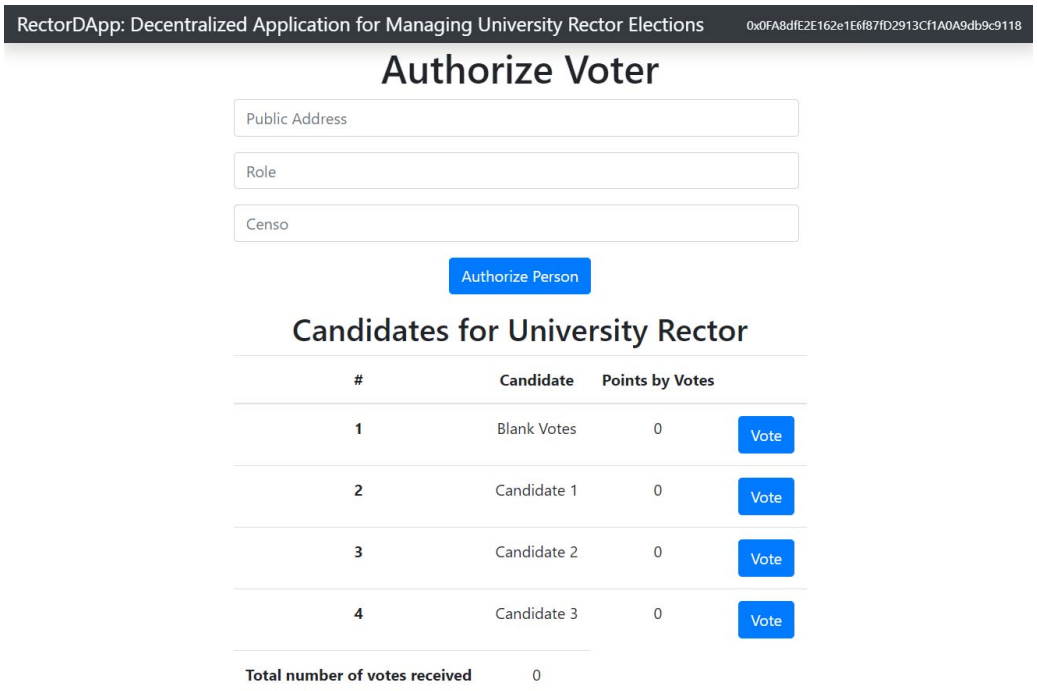

Fig. 2. Web application supporting the case study

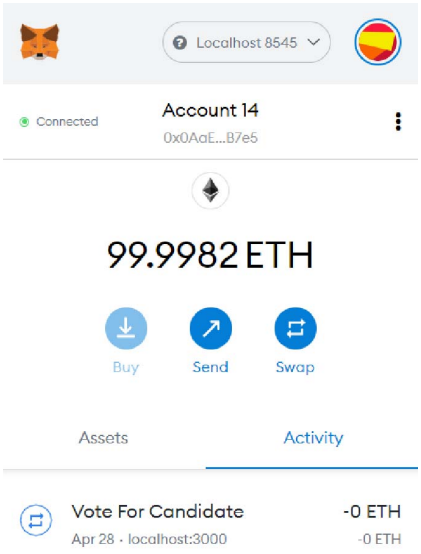

Fig. 3. MetaMask interface after taking a vote

blockchain, we can state that RectorDApp is able to ensure data traceability, immutability and transparency because of the intrinsic characteristics from the design of the blockchain technology.

Additionally, RectorDApp allows non-experts in blockchain to take advantage of all these characteristics in a transparent manner, as RectorDApp provides them with a graphical user interface.

- **RQ2: Is blockchain suitable for running an electronic voting system?**
  Yes, as we have stated in Section II, the design of blockchain technology prevents information from being manipulated. It also provides the possibility to implement business logic through smart contracts. Moreover, it ensures data consistency and integrity. Additionally, this technology has some benefits over traditional voting systems such as increasing efficiency and reducing errors [14].
- **RQ3: What advantages does the use of the blockchain technology provide to electronic voting systems?**
  The traditional voting process relies on slow and laborious counting. Thanks to the blockchain technology, voting results are visible anywhere in a country and removes organizational difficulties in processing voting data. In addition, data are copied to all nodes of the network using the distributed technology of blockchain networks. This means that there is no single point of failure and information cannot be lost. Moreover, the use of blockchain-based voting applications allows people to cast their vote from anywhere in the world. Therefore, it will be accessible to people who cannot attend their polling stations or who due to difficulties cannot exercise their right to vote.

VI. Related Work

In recent years, some blockchain-based works have been proposed for election management.

Dagher et al. [15] propose the creation of a DApp for a voting system using some technologies similar to those used in this paper, such as MetaMask or Web3, and in the same Ethereum blockchain network. However, the main difference with our proposal is the weight of votes; that is, in Dagher's proposal all votes are equally worth, with no difference between voters. In our case, as explained in Section IV, each voter, depending on the university staff type to which he/she belongs, will have a different weight with respect to the final result of the elections.

Giraldo et al. [16] propose the creation of a DApp for the University of Quindío. This work also makes use of the Ethereum blockchain network and similar technologies used in RectorDApp, such as Truffle or MetaMask. Although this DApp is also applied to a university context, Giraldo's proposal does not take into account the weighting of votes according to university staff types to which the voters belong.

On the other hand, Vivek et al. [17] develop a voting management application with different technologies, such as Angular or Amazon Web Services. Moreover, Vivek's proposal uses another blockchain network called Hyperledger Sawtooth, whose characteristics are different from those provided by Ethereum. As an example, Hyperledger Sawtooth allows us to choose the consensus mechanism or the type (public or private) of the blockchain network to be used. So, despite being an application for voting management, Vivek's proposal is different from ours since it does not guarantee data transparency. Moreover, depending on the consensus mechanism chosen, the system may be less secure or unreliable.

Liu et al. [18] propose a protocol based on the integration of blockchain and e-voting. Specifically, the properties satisfied by the protocol are verifiability, dependability, consistency, auditability, anonymity and transparency. However, Liu et al.'s proposal does not provide a user-friendly graphical interface. In our case, we propose a web application that allows non-experts in the blockchain technology to interact with it in a transparent way.

VII. Conclusions and Future Work

In this paper, we proposed an approach that allows for the interaction with the Ethereum blockchain network.

For implementing this approach, we used novel technologies such as Truffle, MetaMask and the Ethereum blockchain. These were key to successfully develop this approach, making it possible the compilation and deployment of smart contracts in the Ethereum network. Moreover, these technologies allowed us to interact with smart contracts through our application created for this purpose: RectorDApp.

RectorDApp is a decentralized application for university rector voting management. The use of blockchain in this voting system provides us with intrinsic characteristics of this technology, such as immutability, transparency, reliability and traceability of information.

Our proposed application makes it possible for a system administrator, such as the UCA, to authorize people who have the possibility of participating in a university rector elections. In addition, the authorized people can exercise their right to vote taking into account which is the university staff type to which they belong, what implies that their votes are weighted differently. Thanks to the use of blockchain technology, we eliminate the term null vote in elections, since this is meaningless in a telematic system. RectorDApp also offers the opportunity to obtain partial or total election results in real time, such as which candidate has the highest number of points at a given time or how many people have voted so far.

As lines of future work, we plan to develop a mobile application that allows any authorized user to vote and interact with the blockchain network without the need to access the web application. On the other hand, we will work on adapting the proposed smart contract so that it can be used in any type of voting. This might entail the modification of its logic, in case it is not required that all users access all the election results or that these are not shown until the voting is concluded.

References

[1] The New York Times, "Can Felons Vote? It Depends on the State," https://www.nytimes.com/2018/04/21/us/felony-voting-rights-law.html, [Accessed April 29, 2021].

[2] D. Drescher, *Blockchain Basics: A Non-Technical Introduction in 25 Steps*, 1st ed. Apress, 2017.

[3] A. Preukschat, *Blockchain: la revolución industrial de internet*, 1st ed. Gestión 2000, 2017.

[4] M. Ananthanarayanan, R. Mishra, and V. Chakka, "How Integrated Process Management Completes the Blockchain Jigsaw," *Digital Systems & Technology*, pp. 1–16, 2018.

[5] D. J. Yaga, P. M. Mell, N. Roby, and K. Scarfone, "Blockchain Technology Overview," NIST, Gaithersburg, MD, NIST Pubs 8202, Oct. 2018.

[6] Ethereum, "Ethereum blockchain." https://www.ethereum.org/, [Accessed April 28, 2021].

[7] "Solidity documentation." https://solidity-es.readthedocs.io/es/latest/, [Accessed April 28, 2021].

[8] ConsenSys Software Inc, "Truffle suite." https://www.trufflesuite.com/truffle, [Accessed April 28, 2021].

[9] MetaMask, "MetaMask," https://metamask.io/, [Accessed April 28, 2021].

[10] ——, "MetaMask Community," https://community.metamask.io/, [Accessed April 28, 2021].

[11] University of Cadiz, "Rector Elections 2019." https://votacionesrector2019.uca.es/, 2019, [Accessed April 28, 2021].

[12] ——, "Rector election statistics 2019." https://votacionesrector2019.uca.es/estadisticas/, 2019, [Accessed April 28, 2021].

[13] X. Xu, I. Weber, and M. Staples, *Architecture for Blockchain Applications*. Cham: Springer International Publishing, 2019.

[14] K. M. Khan, J. Arshad, and M. M. Khan, "Secure digital voting system based on blockchain technology," *Int. J. Electron. Gov. Res.*, vol. 14, no. 1, p. 53–62, Jan. 2018. [Online]. Available: https://doi.org/10.4018/IJEGR.2018010103

[15] G. G. Dagher., P. B. Marella., M. Milojkovic., and J. Mohler., "BroncoVote: Secure Voting System using Ethereum's Blockchain," in *Proceedings of the 4th International Conference on Information Systems Security and Privacy*, INSTICC. SciTePress, 2018, pp. 96–107.

[16] F. Giraldo, M. Barbosa, and C. Gamboa, "Electronic Voting Using Blockchain And Smart Contracts: Proof Of Concept," *IEEE Latin America Transactions*, vol. 18, no. 10, pp. 1–9, 2020.

[17] S. K. Vivek, R. S. Yashank, Y. Prashanth, M. Yashas, and N. Namratha, "E-Voting System using Hyperledger Sawtooth," in *Proceedings of 2020 International Conference on Advances in Computing, Communication and Materials*, 2020, pp. 29–35.

[18] Y. Liu and Q. Wang, "An e-voting protocol based on blockchain," *IACR Cryptol. ePrint Arch.*, vol. 2017, p. 1043, 2017.