A Proposed Blockchain-Based Voting System with User Authentication through Biometrics

M. M. da Silva, A. A. A Silva, N. F. Junior, E. T. Ueda, F. D. Perreira, A. S. dos Santos,
A. E. Guelfi and S. T. Kofuji

Abstract — **Transparency and security in an electoral process are fundamental to the legitimacy of the results and the confidence of voters. Thus, it is necessary to assess opportunities to improve traditional voting systems. Among the main problems is the lack of transparency, due to the impossibility of a voter checking their vote and the lack of access to the source from which the results are obtained. Another problem is mobility, due to the impossibility of performing remote voting, as traditional voting systems continue to require the physical presence of the voter in an electoral zone. Thus, the objective of this work is to propose a voting system that is functional, transparent, safe, and accessible to everyone. Voters can vote through a mobile application with biometric authentication using fingerprint and password access. In our proposal, votes are registered in an Ethereum Blockchain through a Smart Contract, allowing the voter to check their vote. The results are expected to collaborate with the evolution of studies necessary to improve traditional voting systems, especially in fundamental aspects such as security, transparency, and mobility.**

Index Terms — **Biometrics, Blockchain, Voting, Smart Contract, Ethereum.**

I. INTRODUCTION

A voting system allows the individual to choose a choice from a group of existing options and enables a population to decide in a democratic way who its representatives will be.

The main requirements for a functional voting system should be: guarantee of anonymity [14]; transparency; possibility of investigation and audit [21]; mobility[10]; only authorized voters can vote; security against double voting; security against voter coercion; calculation of elections only after the end of the electoral process [17]; and decentralization, that is, the data source where votes are registered cannot be centralized in any entity [9];

Traditional voting systems do not meet all of these requirements. Transparency is not achieved, as the source of the data where the votes are recorded is not made available to voters, making any kind of public audit of the results impossible. There is no mobility, as voters have to go to the polling booth. There is complete centralization, as there is dependence on a trusted third party for storing, calculating, processing, and guaranteeing the anonymity of a vote.

To consider the aim of a decentralized voting system and in accordance with the security, verifiability, and auditing requirements, it is necessary to evaluate the viable technologies and existing studies related to the theme.

Each electoral system has specific rules. In Brazil, the electoral code is sanctioned by law number 4,737 of July 15, 1965, where issues such as vote secrecy, participating public, parties, preparatory acts, among many others are addressed. Therefore, a voting system for political offices in Brazil must obligatorily adhere to all the clauses described in this law.

The basic requirements of a voting system presented in this work do not take into account the rules described in electoral codes used in public elections for political office. In other words, the scope of this proposal serves voting systems governed by local rules, such as elections for associations, condominiums, or clubs.

In addition to this Introduction, this paper contains a short explanation of the main concepts used and the main guiding references in chapter II. Chapter III brings the proposal of this paper, while chapter IV presents and discusses the results achieved. Finally, chapter V resumes the main points discussed and presents the conclusions.

II. BLOCKCHAIN, ETHEREUM AND BIOMETRY

A. Blockchain

The study that gave rise to the first Blockchain was published by [15]. In this approach, an electronic payment system is based on cryptographic evidence, allowing transactions to be carried out directly with each other, without the need for a trusted third party.

In a Blockchain, a digital currency is a chain of digital signatures. To avoid double spending of the same currency, only the oldest transaction involving one currency will be valid. Transactions must be propagated on a public network and

M. M. da Silva, Instituto de Pesquisas Tecnológicas (IPT), São Paulo, Brazil, celo.moro@hotmail.com.

A. A. A. Silva, Instituto de Pesquisas Tecnológicas (IPT) / Universidade de São Paulo (USP) / Universidade Paulista (UNIP) / Centro Universitário SENAC, São Paulo, Brazil, anderson@uol.com.br.

N. F. Junior, Universidade de São Paulo (USP), São Paulo, Brazil, norisjunior@gmail.com.

E. T. Ueda, Instituto de Pesquisas Tecnológicas (IPT), São Paulo, Brazil, eduardoueda@ipt.br.

F. D. Pereira, Centro Universitário Eurípides de Marília, São Paulo, Brazil, prof.fabiopereira@gmail.com.

A. S. dos Santos, Instituto de Pesquisas Tecnológicas (IPT), São Paulo, Brazil, alesan@ipt.br.

A. E. Guelfi, Universidade do Oeste Paulista (UNOESTE), São Paulo, Brazil, guelfi@unoeste.br.

S. T. Kofuji, Universidade de São Paulo (USP), São Paulo, Brazil, kofuji@usp.br.

network participants use the same system to reach an agreement on the order of transactions received, without central authority.

Ownership of a digital currency in a Blockchain is secured through symmetric encryption [16]. The public key is used to send a certain value to the recipient, who will be the only one able to assume this property through their respective private key.

To provide security against time fraud, in a Blockchain there is a timestamp for each hash corresponding to a block of transactions.

To prevent blocks from being processed by dishonest participants, the Blockchain uses a stress test as part of the block mining process, which requires the miner to devote a certain amount of computational resources and energy to the generation of a new block, in order to make attempts to generate a false blockchain economically disadvantageous. [7]

In a Blockchain, a currency is not manipulated individually. According to [15], to allow an amount to be divided and combined, transactions contain multiple inputs and outputs.

To maintain privacy in transactions, the model proposed by [15] maintains the public keys used in anonymous transactions.

B. Ethereum

According to [7], Ethereum is a platform for developing and deploying reliable decentralized applications, denominated *Smart Contracts*.

The possibility of creating and executing the *Smart Contracts* is the main difference between Ethereum Blockchain and the Blockchain used by Bitcoin [2].

In Ethereum, the consensus mechanism between network nodes guarantees not only the immutability of transactions but also of *Smart Contracts*. This mechanism, therefore, ensures that the writing code of a *Smart Contract* is tamper proof, even in a distributed and decentralized execution model.

Smart Contracts are systems that automatically move digital assets according to previously established and immutable rules [4].

According to [7], the *Smart Contracts* communicate via a transaction. When receiving a message, the code of the *Smart Contract* is executed and this potentially may or may not send a message to the other *Smart Contract* account, which in turn will react according to their code, and so on.

Processing messages and transactions involves computational cost and in Ethereum, the party requesting the execution must establish the maximum amount of GAS[1] units it is willing to pay and what value in Ether it will pay for each unit spent [7].

C. Biometry

The current study uses fingerprint-based biometrics as a method for authentication because it is considered secure, is widespread among *mobile* users, and extensive documentation is available on the APIs for development in *Android* and *IOS* operating systems [18].

Based on identification through fingerprint irregularities, the capture of the digital image is performed by optical means, being digitally processed through a system capable of identifying the dactyloscopic characteristics, and comparing them with a database record, to determine access or not [19].

D. Hypotheses that guide this research

[22] present a voting system that utilizes the Ethereum Blockchain, where voters can use their Ethereum wallet or an *Android* device to submit their vote. The work is intended only for small elections and the authors warn about the scalability of the Ethereum network for large votes, recommending in-depth studies on the topic. The authors also warn that when using a public Blockchain, there may be a risk of guaranteeing the anonymity of a voter and do not recommend this for official and critical elections.

[3] propose a conceptual model for protecting credentials within digital wallets using biometric methods, denominated *BioWallet*. This model employs user authentication through biometrics and an access password. The transaction is only allowed if both validations are positive. The work by [3] is not directed at voting systems. The authors propose double authentication, through biometrics and an access password to increase the security of the digital wallets used to carry out financial transactions.

The work of [17] allows voting through a *mobile* application or through an authentication terminal. All communication between the authentication center and the central election office is carried out through a secure channel, integrating existing facilities in traditional voting systems with a new authentication layout based on biometrics.

The work of [9] focuses on recording votes in a Blockchain. According to the authors, traditional voting systems are extremely vulnerable to tampering when managed by an organization with full control over the system and database. The traces of possible tampering could be easily erased, thus making any type of audit difficult. The presented solution uses a permissioned Blockchain, which is nothing more than a private Blockchain network, developed and maintained centrally by one or more companies or institutions. The authors focus their work exclusively on recording data in a Blockchain, not addressing which access mechanisms are available to voters or which forms of integration are possible.

[13] propose a Blockchain-based voting system. Voter authentication involves a trusted third party denominated *Trusted Third Party* (TTP). This entity is responsible for verifying if a voter is entitled to vote, by consulting the authenticating organization through an encrypted secret message. The authors propose using the same Blockchain used by Bitcoin. Voter authentication is not performed at the time of voting. In this way, the Blockchain can receive vote records from anyone. However, after the conclusion of the election, authentication is performed, where only votes corresponding to voters with voting rights are computed.

III. Proposal

This section presents the proposal for the development of a

[1] GAS is not an acronym but a term used to refer to fees charged for processing a message or transaction in an Ethereum Blockchain.

mobile system with authentication mechanisms, which can be carried out only through the user's own smartphone. In addition, it also details a *Smart Contract* developed for the Ethereum Blockchain for registration of votes, and a conceptual model of applications for *backend* responsible for the integration between the *mobile* application and the *Smart Contract Ethereum Blockchain*, in addition to the orchestration of services for user authentication and voting rights verification.

The scope of this proposal includes the use of intermediaries such as Infrura and third-party software such as the Android Biometrics API. Furthermore, details about the creation of Blockchain accounts, coercion actions, or counting of voting results are also not considered. All these points can be considered as vulnerable features and they are discussed in more detail in section V Conclusion.

A. Conceptual Model of the Technical Solution

Like the solution proposed in the work by [22], in the current work, the vote registration is performed through a *Smart Contract Ethereum Blockchain*, which ensures that the program used to record a vote cannot be changed, thus increasing confidence in the system.

However, in the proposal presented in this research work, the voter uses a *mobile* app to vote, therefore, there will be an integration between the application and *Smart Contract* when registering the vote.

The *mobile* application presented in this proposal is developed for the platform *Android* [8]. The choice of platform takes into account the cost of the development tools and the platform's compatibility with the web3j API [23], which is required to communicate with the Blockchain through the application.

Figure 1 illustrates the overview of the functional voting system proposed in this study:

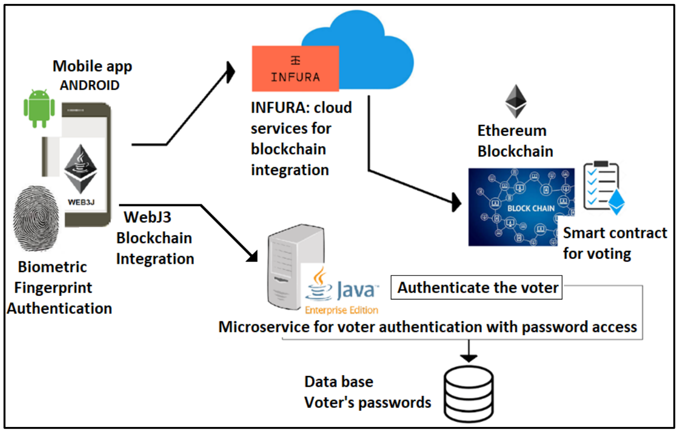

Mobile app
ANDROID
INFURA
INFURA: cloud services for blockchain integration
Ethereum Blockchain
WEB3J
BLOCK CHAIN
WebJ3 Blockchain Integration
Smart contract for voting
Biometric Fingerprint Authentication
Java
Enterprise Edition
Authenticate the voter
Microservice for voter authentication with password access
Data base
Voter's passwords

Fig. 1. Overview of the Blockchain-based Voting System with Biometric Authentication Proposed in this Study.

Figure 1 shows an overview of this proposal, with the main functional modules, from biometrics and the Android voting application, to integration with additional authentication services and blockchain integration.

As a result of this proposal, it is expected that a functional voting system will be obtained based on Blockchain with user authentication using biometrics and access through a *mobile* application.

B. Authentication with access password

A voter's navigation journey through the *mobile* application is initiated by authentication with an access password. In other words, the first authentication process (there is a second process as well) is performed through an application on the user's mobile. The application has a *login* screen, where voters are required to enter their username and password.

These data are used in a Hash-based Message Authentication Code (HMAC) function to produce a final *hash* that represents the voter's credentials. In computing, a *hash* is a sequence of bits generated by a scatter algorithm. HMAC functions use an existing *hash* generation function in conjunction with a secret key [5] to generate a final *hash*.

The secret key used in the HMAC function is the user name and password used in the voting application. The other component of the HMAC is the International Mobile Equipment Identity (IMEI) of the voter device. The IMEI is unique for each mobile device. These data are accessible on all mobile devices [1].

This final *hash* generated in the application running on the voter's mobile device is sent to a central server, where there is software developed in Java language and a database where the voter's access credentials and the IMEI of the mobile device are stored.

This software generates a new *hash* through an HMAC function, using the voter's password and the IMEI registered in the database. If the final *hash* generated is the same as the *hash* generated in the mobile app, user authentication becomes successful.

It is important to note that the voter and their mobile device are directly linked to the authentication process. Authentication will only be successful if the voter uses their mobile device at the time of authentication. It is not possible to vote via other mobile devices.

The registration of the access password of the voter and the mobile IMEI is not part of the scope of this work. However, this study is based on the premise that the registration of these data was previously carried out by the entity responsible for organizing the election. In this way, each voter already has their name and access password registered, in addition to the IMEI corresponding to their mobile. The application consults this database to authenticate voters.

The hash is generated twice. The first hash is generated in the mobile application, based on the user and password entered by the voter and the device IMEI, programmatically obtained through the Android platform API in the application used for the vote.

This hash is sent to the central server, where the second hash is generated based on the user, password, and IMEI of the voters' device registered in the database.

Authentication is only successful if the values of the hash codes are the same.

Figure 2 illustrates the authentication process involving the user and the password and IMEI of the voter’s device:

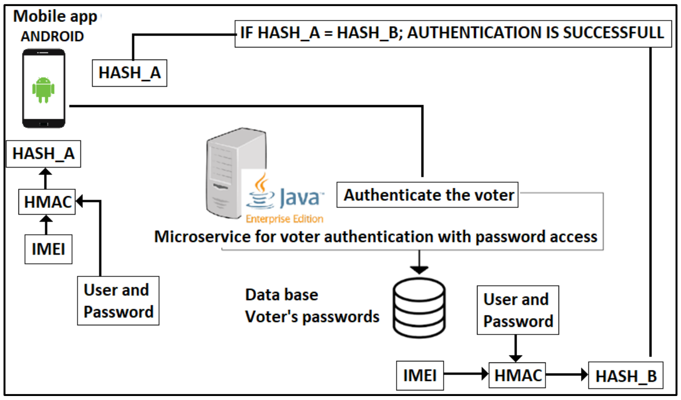


Fig. 2. Voter authentication with password access.

C. Application main screen

Continuing the voter journey using the application, successfully authenticated voters at *login* advance to the next step and gain access to the main screen of the application.

Figure 3 illustrates the systemic integrations performed in this step:

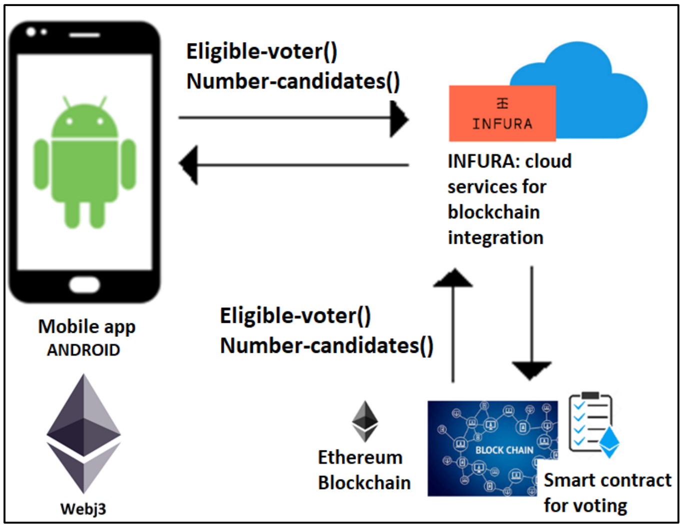


Fig. 3. Systemic integrations performed on the main application screen

As can be seen in Figure 3, the main screen performs two queries on the Blockchain: (1) a check if the voter has already voted; and (2) a list of candidates participating in the election.

For the application to communicate with the Blockchain it is necessary to have a Blockchain account. In addition to the application user password, the voter must have an account on the Ethereum Blockchain. The credentials of this account are used for each communication with the Blockchain.

The creation of Blockchain accounts for each voter is the responsibility of the election organizer. However, the Blockchain account creation process is beyond the scope of this work.

To carry out the experiment, a local file on the *mobile* device is used, where the Blockchain credentials are saved and are used by the application in each communication with the Blockchain. Therefore, in this authentication, the user does not need to interact directly, as this will be performed automatically in all communications between the application and the Blockchain through a library that is compatible with the *Android* operating system, called web3j [23].

The web3j library is used in the programming of the mobile application and allows remote execution of all the functions of a *Smart Contract* in an Ethereum Blockchain. However, an intermediary between the application and the Blockchain is also required for the integration to be successful.

The intermediary role is performed through the Infura services [11]. Infura has mechanisms for connecting to the Blockchain network. Figure 4 illustrates the integration possibilities using the Ethereum Blockchain ecosystem offered by Infura.

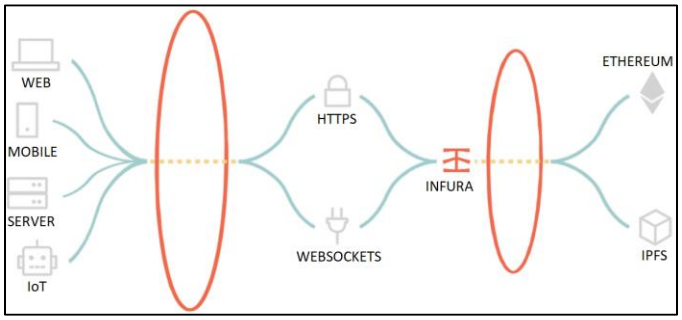


Fig. 4. Integrations with Ethereum Blockchain through Infura

As can be seen in Figure 4, it is possible to connect to an Ethereum Blockchain through APIs provided by Infura. The focus of Infura is to provide infrastructure and services for remote communication with an Ethereum Blockchain, simplifying projects involving this type of integration. All communications between the application and the Blockchain performed in this proposal use Infura through the HTTPS protocol.

The choice of Infura was focused on simplifying the proposed solution, as it solves several infrastructure-related concerns. In this way, it is possible to focus on the development of functionalities in the application.

However, in a real election, Infura would represent a third party to participate in the voting system, which raises important points of attention related to security that are not addressed in this study.

The *Smart Contract* used in this study is developed in Solidity language [20].

As can be seen in Figure 3, the methods used in the main screen of the application (Eligible voter and number of Candidates) have the same name as the functions declared in the *Smart Contract Blockchain*. This is one of the web3j library definitions.

The first functionality, to be run on the main screen of the application, checks if the voter has voted yet. At this point, the application's Eligible() voter method makes a query to the corresponding function in the *Smart Contract*, which notifies as to whether the voter has already voted. If the voter has already voted, an informational message is displayed and it will not be possible to vote. Otherwise, a Vote button is displayed and the voter can start the voting process.

Another function of the application's main screen informs the number of candidates participating in the election. In the same way, the application again queries the Blockchain, but now using the Quantity_Candidates() method, which queries the corresponding function in the *Smart Contract* and returns the number of candidates to be presented on the application's main screen.

D. *Fingerprint Biometric Authentication*

Voters who have not voted yet can proceed to the next step and begin the voting process. At this point, a second authentication is performed, this time using fingerprint biometrics.

This authentication is performed directly on the mobile device, without any communication with external systems. It is a feature of the *Android* operating system used for application programming.

Possible vulnerabilities related to the fingerprint biometrics process in the Android operating system are not addressed in this study.

Taking into account the use of *login* with user password and mobile device IMEI to access the application and the additional use of biometrics, it can be stated that the proposal presented in this research paper is characterized as a dual authentication solution. Dual authentication is also addressed in the works of [3] and [17].

E. *Conducting a vote*

Voters with positive biometric authentication in the application can proceed to the next step, where they can select a candidate to cast their vote.

The *Smart Contract* does not allow more than one vote for the same public key in the Blockchain. In this way, double voting is impossible.

As can be seen in Figure 5, the List_Candidates() and vote() methods enable consultation of the list of candidates to be displayed on the screen and performance of the vote.

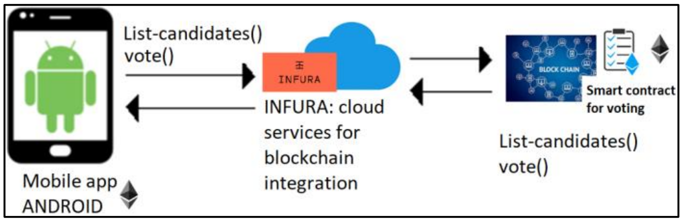


Fig. 5. Integrations with Ethereum Blockchain through Infura

Figure 5 presents the systemic integrations existing in the voting stage.

The *Smart Contract* used for voting also has a function for registering candidates and should only be used when registering candidates by the entity responsible for the election.

However, the registration of candidates is not addressed in this proposal and the work is based on the premise that candidates have already been registered by the entity responsible for the election.

The application presents the voter with a screen with the list of all candidates participating in the election. When selecting a candidate, the application executes the *Smart Contract* vote() function and their vote is registered in the Blockchain.

F. *Proof of voting*

Every transaction carried out on the Blockchain has a unique identifier. Upon completion of the vote, the application presents the voter with a voting slip with the unique identifier of the Blockchain transaction, the unique identifier of the program responsible for processing the vote, and the public key of the Blockchain account used for the vote.

Through the transaction's unique identifier, any voter can consult their vote via the Internet and confirm that the candidate registered in the vote is really the one chosen by them. Another important point concerns immutability, as Blockchain technology ensures that the transaction is immutable. These features increase system reliability and security.

Although voters can consult their transaction, all existing data do not compromise vote secrecy, given that the public key of the Blockchain account used in the transaction is exclusive to the voter. Thus, there are important gains in terms of auditing and confidence.

G. *Verification of the election result*

An important advantage of using a public Blockchain in an election is the possibility of the voter checking the legitimacy of the registration of their vote and calculation of the final result without depending on third parties.

The proposal presented in this paper demonstrates how a voter can check their vote, but reading all votes in the Blockchain and calculating the final result are beyond the scope of this research work.

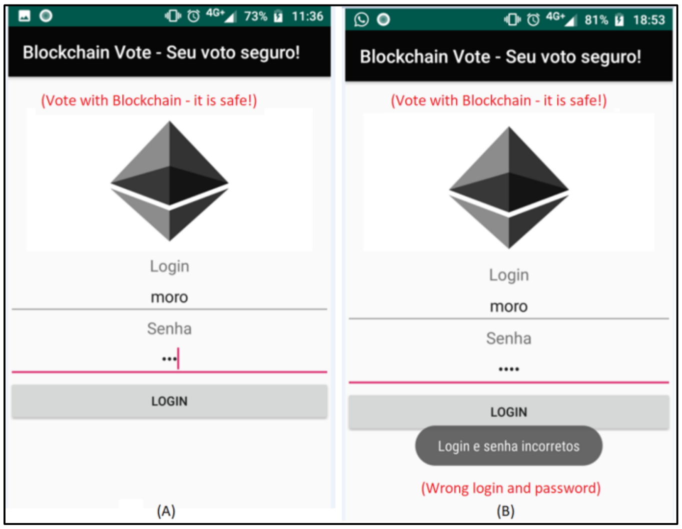
However, the design of the technical solution presented in the proposal of this work is also viable for the reading of all votes existing in the Blockchain, making the calculation of the final result feasible. In this way, future works may extend this research work, approaching ways of calculating the final result from the technical solution presented in this proposal.

IV. Experiment and Analysis of Results

Validation of the proposal is performed by comparing compliance with the main requirements of a voting system. The system proposed in this study and the others discussed during the research work are evaluated for each requirement.

A. *Mobility, Dual Authentication, Biometrics and Double Vote*

The proposed system presented in this study uses an application for cell phones and the navigation journey begins with the voter authentication screen. Figure 6A shows the screen used in the authentication process.



(A)

Blockchain Vote - Seu voto seguro!

(Vote with Blockchain - it is safe!)

Login

moro

Senha

•••

LOGIN

(B)

Blockchain Vote - Seu voto seguro!

(Vote with Blockchain - it is safe!)

Login

moro

Senha

••••

LOGIN

Login e senha incorretos

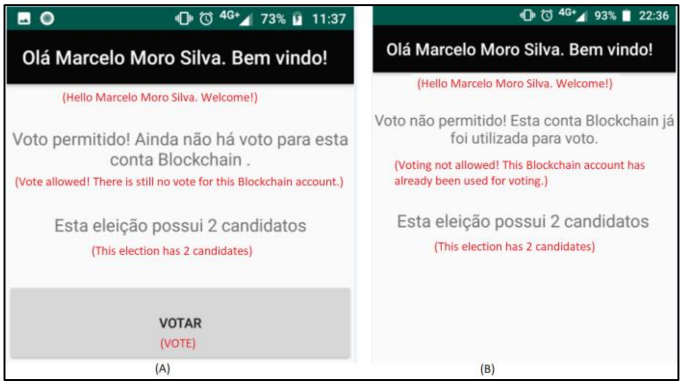
(Wrong login and password)

Fig. 6. Login screens with access password. 6A: password entry. 6B: wrong login and password.

In case of authentication failure, the voter will not be able to access the application's functionalities and an informational message will be presented, as shown in Figure 6B.

This functionality represents the first level of authentication existing in the experiment. This authentication uses the *login*, password, and IMEI of the voter's device. In this way, only the voter will be able to vote through their cell phone and it will not be possible to impersonate a voter through a device that is not included in their register.

Authenticated voters will have access to the home screen of the application, illustrated in Figure 7.



(A)

Olá Marcelo Moro Silva. Bem vindo!

(Hello Marcelo Moro Silva. Welcome!)

Voto permitido! Ainda não há voto para esta conta Blockchain .

(Vote allowed! There is still no vote for this Blockchain account.)

Esta eleição possui 2 candidatos

(This election has 2 candidates)

VOTAR

(VOTE)

(B)

Olá Marcelo Moro Silva. Bem vindo!

(Hello Marcelo Moro Silva. Welcome!)

Voto não permitido! Esta conta Blockchain já foi utilizada para voto.

(Voting not allowed! This Blockchain account has already been used for voting.)

Esta eleição possui 2 candidatos

(This election has 2 candidates)

Fig. 7. Main page with number of candidates and vote confirmation. 7A: Vote allowed. 7B: Voting not allowed.

This screen in Figure 7 has two functions: querying the number of candidates in the election and querying whether the vote has already been taken. This information is obtained directly from the Blockchain, by consulting the *Smart Contract*.

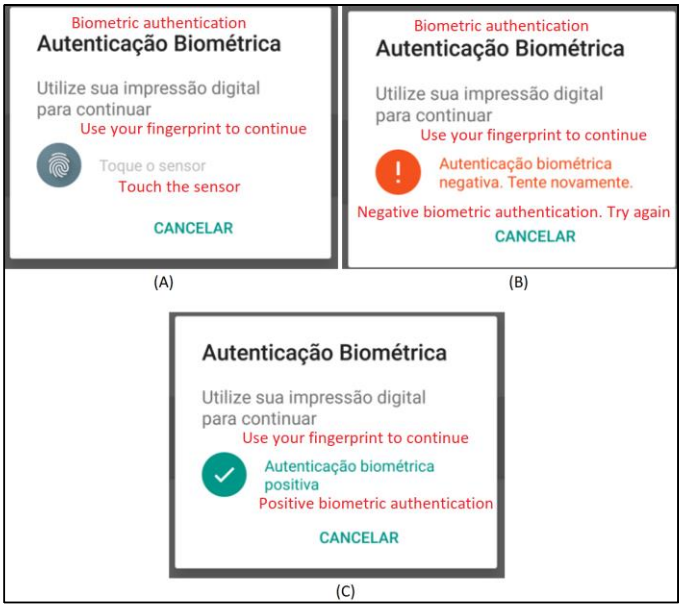
It is important to note that voting control for a voter is in the Blockchain and not the application. The guarantee of preventing double voting is offered by the Blockchain technology, through the programming of the *Smart Contract*. The application only consults a *Smart Contract* function that warns if the voter has already voted. In this way, the logic in the application allows voting only if the voter has not yet voted.

The voter's public key used for a vote is stored in the Blockchain and consulted for every new vote. If there is a vote linked to a certain public key in the Blockchain, it cannot be used again.

If the public key has not been used in a vote, the voter receives the message in Figure 7A, otherwise the message in Figure 7B.

As can be seen in Figure 7B, in addition to informing the voter that the vote has already been taken, the application does not have the button used for the vote, making it impossible to attempt to vote through the application. However, even if there is a button, *Smart Contract* would not allow a second vote.

Voters who have not yet voted can cast the ballot. The application developed in this experiment uses the cell phone biometric fingerprint sensor. Therefore, only mobile devices with this feature are compatible with the application. The fingerprint biometric collection page is shown in Figure 8:



(A)

Biometric authentication

Autenticação Biométrica

Utilize sua impressão digital para continuar

Use your fingerprint to continue

Toque o sensor

Touch the sensor

CANCELAR

(B)

Biometric authentication

Autenticação Biométrica

Utilize sua impressão digital para continuar

Use your fingerprint to continue

Autenticação biométrica negativa. Tente novamente.

Negative biometric authentication. Try again

CANCELAR

(C)

Autenticação Biométrica

Utilize sua impressão digital para continuar

Use your fingerprint to continue

Autenticação biométrica positiva

Positive biometric authentication

CANCELAR

Fig. 8. Biometric data collection. 8A: biometric entry. 8B: Negative biometric authentication. 8C: Positive biometric authentication

The screen shown in Figure 8A is presented after the user clicks on the Vote button, available on the application's home page.

At this point, the voter is asked for a sample of their fingerprint. The *Android* operating system has mechanisms for recording and managing biometric data not covered in this study. However, it is imperative that the mobile device user has registered their fingerprints on the mobile device for this feature to work in the application.

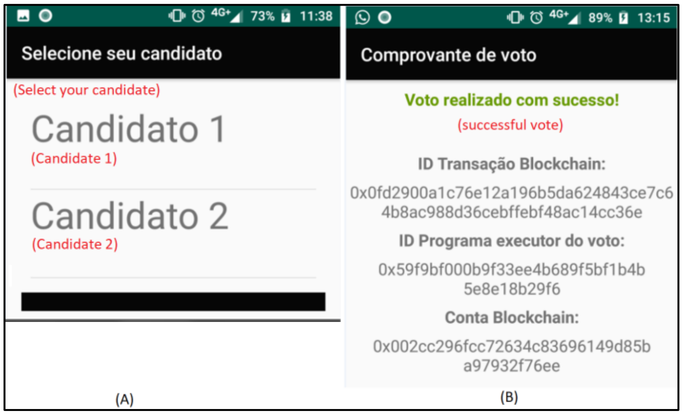
Upon collecting the voter's fingerprint, the application performs an assessment and access to the voting page is authorized only if authentication is successful. Figure 8B shows the negative biometric authentication screen.

Biometric authentication represents the second level of authentication that exists in the application. Thus, it can be stated that the solution presented in this experiment uses double authentication. Figure 8C shows the positive biometric authentication screen.

After performing password and biometric authentication, the voter is authorized to vote.

The first step in a voting process is the presentation of the list of candidates.

Figure 9A shows the screen with the list of candidates for voting after successful authentication.



Selecione seu candidato

(Select your candidate)

Candidato 1
(Candidate 1)

Candidato 2
(Candidate 2)

[redacted]

Comprovante de voto

Voto realizado com sucesso!
(successful vote)

ID Transação Blockchain:
0x0fd2900a1c76e12a196b5da624843ce7c64b8ac988d36cebffebf48ac14cc36e

ID Programa executor do voto:
0x59f9bf000b9f33ee4b689f5bf1b4b5e8e18b29f6

Conta Blockchain:
0x002cc296fcc72634c83696149d85ba97932f76ee

Fig. 9. Ongoing votting. 9A: candidate list. 9B: proof of voting.

The list of candidates is registered in the Blockchain and can be obtained by consulting the *Smart Contract*. This feature increases the security against tampering with registered candidates, since this registration is carried out at the time of creation of the *Smart Contract*. It is not possible to change them once they have been created.

Each candidate list item has an event. By clicking on one of the candidates, the vote is processed through a call to *Smart Contract* where there is a function used to record the vote.

After registering the successful vote, the *Smart Contract* returns data from the transaction performed to the application. These data are informed in a voting slip, which contains the program ID. This is the ID of the *Smart Contract* responsible for processing the vote. This feature is fundamental as it characterizes the adulteration prevention of the program. Only one *Smart Contract* is used in voting, therefore, all votes must be processed by the same *Smart Contract*, which is guaranteed immutability through Blockchain technology block processing rules.

Another datum on the voucher is the Blockchain account. This is the public key used for voting. As stated in previous sections, each voter can only vote once, so this public key can only be used once.

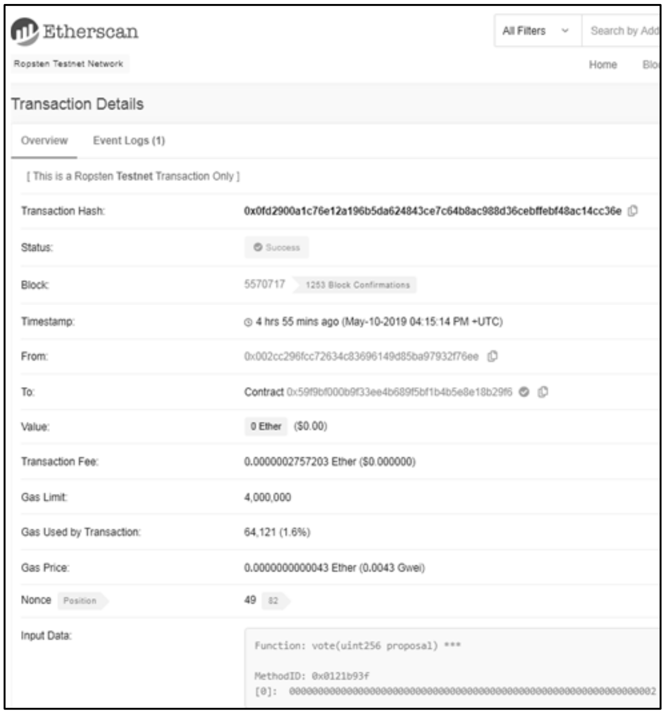
Figure 9B demonstrates the voting slip presented to the voter after a successful vote has been made. Registration of the vote performed by *Smart Contract* results in a transaction on the Blockchain and the receipt informs the *ID* of this event. This transaction has a guarantee of immutability through the Blockchain network, where the vote performed is registered. Thus, the proposal presented in this study has a guarantee against tampering with the program used in the election and the vote performed.

This experiment uses the Ethereum Ropsten Test Network [6]. It is a Blockchain Ethereum network used for testing. This network allows consultation of all transactions carried out through an internet website. The entire *Smart Contract* implementation was developed in Solidity language.

Through the transaction *ID* generated when registering the vote, the voter can consult their transaction on the Internet. The transaction *ID* is defined in the *Transaction Hash* parameter. The *ID* of the executing program of the vote is defined in the parameter *To*, which means that the *Smart Contract* corresponds to the destination of a transaction. Finally, the Blockchain account corresponds to the *From* parameter which means the origin of the transaction.

[15] created the Blockchain technology and eliminated the need for a trusted third party to guarantee security in banking transactions. However, the proposal presented in this study does not eliminate the need for a trusted third party in a voting system. The figure of a person responsible for the election is maintained for functions such as registration of voter data, authentication of voters, registration of Blockchain keys, and publication of the *Smart Contract* used in the election.

Figure 10 shows a website query [6] where the transaction used in this experiment can be checked:



Transaction Details

[This is a Ropsten Testnet Transaction Only]

Transaction Hash:	0x0fd2900a1c76e12a196b5da624843ce7c64b8ac988d36cebffebf48ac14cc36e
Status:	Success
Block:	5570717 1253 Block Confirmations
Timestamp:	4 hrs 55 mins ago (May-10-2019 04:15:14 PM +UTC)
From:	0x002cc296fcc72634c83696149d85ba97932f76ee
To:	Contract 0x59f9bf000b9f33ee4b689f5bf1b4b5e8e18b29f6
Value:	0 Ether ($0.00)
Transaction Fee:	0.0000002757203 Ether ($0.000000)
Gas Limit:	4,000,000
Gas Used by Transaction:	64,121 (1.6%)
Gas Price:	0.0000000000043 Ether (0.0043 Gwei)
Nonce Position	49 82
Input Data:	Function: vote(uint256 proposal) *** MethodID: 0x0121b93f [0]: 0000000000000000000000000000000000000000000000000000000000000002

Fig. 10. Result of the Internet query of the transaction performed in the experiment related to a vote through the application

B. Comparison between works

The channel for accessing the system is not addressed in the study presented by [22]. However, other works referenced in this study have the access channel as their main motivation, such as the work of [17] which presents a proposal based on mobile applications with the aim of increasing people's participation and engagement in the electoral process.

[17] propose integrating the traditional voting system with a mobile application. The authors involve a third party responsible for the authentication of voters and election control. This is a common feature of the experiment presented in this

study, where a third party is also needed for voter authentication, as mentioned earlier.

[17] use biometric authentication through fingerprint. Non-biometric data is also used for authentication, where a four-digit authorization code must be entered. Dual authentication is employed in the proposal presented in this article, using fingerprint and password biometrics. Therefore, both have an additional layer of security in a voting system.

[17] do not use Blockchain technology and keep the base where votes are registered centralized in the organ responsible for the election. On the other hand, aiming at less centralization of responsibilities in the body responsible for the authentication of voters, the proposal presented in this study uses a public Blockchain, where all voters can access and check the results, demonstrating an additional gain in transparency and security.

The concern with data source tampering and control in a voting system is the focus of the work of [9]. As an alternative, the authors propose a solution based on Blockchain.

The proposal uses a permissioned Blockchain, where the nodes responsible for processing blocks are limited to a previously registered group. This is a difference from the proposal presented in the current study, where a public Ethereum Blockchain is used.

[9] adopt specific mechanisms for processing blocks and mining, aimed at controlling the participating nodes and the order in which each one must process a block.

This control does not exist in the solution proposed in the current study, which uses the block processing and mining mechanisms of the Ethereum Blockchain network. Controlling the nodes participating in the network can be a security differential in a Blockchain-based system. However, the proposal presented in this dissertation does not address possible block tampering attacks and their characteristics.

The work by Lee et al. (2016) mentions the use of a Blockchain, but the focus is on the authentication of a voter before interacting with the Blockchain, without compromising the confidentiality of their identity, even to those responsible for authentication.

To do so, the authors involve a trusted third party, in addition to the agency responsible for the election in the authentication process.

During the authentication process there is an exchange of information between the agency and the trusted third party. Each one keeps a part of the voter's information and one does not have access to the other's part. In this way, the author proposes an authentication model that guarantees voter anonymity. The proposal presented does not address possible fraud in the body responsible for the election during the authentication process.

Lee et al. (2016) use the same Blockchain used in Bitcoin. However, implementation details are not covered by the authors.

The proposal presented in the current study demonstrates the integration between a mobile application and a Blockchain, enabling the possibility of improving the experience in a voting system, without sacrificing security, obtained through dual authentication, biometrics, and Blockchain technology. This is an important contribution of this work, considering the difficulty of finding works dedicated to improving the user experience in a voting system during the research stage.

TABLE 1
Comparison of Proposals for Voting Systems

Num	Requirements	[22]	[17]	[9]	[13]	Proposal for this study
1	Mobility	✗	✓	✗	✗	✓
2	Anonymity	✓	✓	✓	✓	✓
3	Dual Authentication	✗	Biometric / Non Biometric	✗	Biometric / Non Biometric	Biometric / Non Biometric
4	Biometry	✗	Voice/Fingerprint	✗	✗	Fingerprint
5	Anti-tampering of the vote	✓	✗	✓	✓	✓
6	Anti-tampering of the program	Ethereum Smart Contract	✗	✗	✗	Ethereum Smart Contract
7	Double voting	✓	✗	✗	✗	✓
8	Anticoercion	✗	✗	✗	✗	✗
9	No Intermediaries	✓	✗	✗	✗	✗

As can be seen in Table 1, the proposal presented in this study is shown to adhere to 7 out of the 9 main requirements of a voting system. They are: (1) mobility: voters can use their mobile phone to vote; (2) anonymity: each voter has their public key on the Blockchain and there is no link between this key and the voter's identity. The vote only registers the public key, therefore, it is not possible to identify the voter; (3) double authentication: through biometrics and an access password; (4) biometrics: authentication uses fingerprint biometrics; (5) anti-tampering of the vote: Blockchain technology guarantees the immutability of the vote; (6) program anti-tampering: the vote is registered on the Blockchain through a Smart Contract that has immutability guaranteed by the Blockchain technology; (7) double vote: the system does not allow the same vote key to be used more than once. This is an important contribution of this study. The following are not covered; (8) anti-coercion, which occurs when someone forces someone to vote for a particular candidate and (9) the presence of intermediaries, represented by Infura, which guarantees the integration between the Blockchain and the application.

V. Conclusion

This work is dedicated to collaborating with research focused on increasing transparency and security, and providing a better user experience in voting systems. Therefore, it presents a proposal for a functional voting system based on Blockchain technology, accessible through an application for cell phones, with user authentication through biometrics and an access password. This model is applicable to elections that are not regulated by law or with pre-established rules.

Mobility is an important feature in a voting system, as it can increase voter engagement and participation in an election, in addition to adapting a voting system to the context provided by digital transformation, where applications make people's lives easier in everyday tasks, such as paying bills or checking the bank balance.

Through the experiment presented, it was possible to integrate an application for mobile phones with biometric and non-biometric authentication of an Ethereum Blockchain, where there is a *Smart Contract* used for votes in execution. The smart contract code used in this paper is available at: https://github.com/marceloMoro/Ethereum_Projects/blob/main/Election.sol.

The feasibility of the proposal was proven by performing functional tests of the following features: authentication by password access; biometric authentication; consultation of the candidate list; taking of votes; casting votes with invalid data; taking of votes with data from voters who have already voted. All tests were successful.

After surveying the main works related to the topic, it was possible to identify the main requirements in a voting system, as shown in Table 1. The evaluation of the proposal presented in this study on these requirements is shown in Table 2, where there is a comparison with the other proposals presented in studies addressed in this research paper.

Among nine main requirements in a voting system, the proposal presented in this study proved to be adherent to 7, namely: (1) mobility: voters can use their mobile phone to vote; (2) anonymity: each voter has their public key on the Blockchain and there is no link between this key and the voter's identity. The vote only registers the public key, therefore, it is not possible to identify the voter; (3) double authentication: through biometrics and an access password; (4) biometrics: authentication uses fingerprint biometrics; (5) anti-tampering of the vote: Blockchain technology guarantees the immutability of the vote; (6) anti-tampering of the program: the vote is registered in the Blockchain through a *Smart Contract* which has immutability guaranteed by Blockchain technology; (7) double vote: the system does not allow the same vote key to be used more than once.

The votes registered through the proposal presented in this study are immutable, according to the processing and consensus criteria of the Ethereum Blockchain network. The consultation of votes is public, so any voter can confirm their vote. This feature represents an important advance in transparency in a voting system.

Biometric authentication is linked to the IMEI of the voter's mobile device. This feature is critical to increasing voter security, especially in attempts to use third-party data in a vote.

A. *Limitations and future works*

The proposal presented in this study has some limitations which can be addressed by future works:

- Anti-coercion: prevention of coercion of voters by parties interested in a given result. Voting systems must have a means to prevent coercion [17]. The current work does not address this functionality and its implementation in a system and in functional voting can be explored in future works;
- No intermediaries: election in which the voter interacts directly with the Blockchain, without intermediaries. The use of Infura to communicate with the Blockchain represents a point of risk, considering the possibility of interception and leakage of confidential voter information. In future work, it is possible to evaluate the replacement of this intermediary;
- Third-party software: biometrics uses the API available through the *Android* operating system. Possible vulnerabilities in the API or *Android* operating system that could be exploited by fraudsters or interceptors are not evaluated and may be addressed in future work. Furthermore, some biometric readers are able to differentiate living tissue from dead tissue. This feature adds an extra layer of security to the biometric authentication process. However, the current work does not explore this kind of vulnerability;
- Blockchain: The scalability of Blockchain technology is also a restriction cited in the works of [12] and [22]. There are outstanding issues regarding the scalability of Blockchain technology for elections involving many voters, which are not covered in this study. The creation of the Blockchain account is another sensitive point of the process, not addressed in this study. It is necessary that the generation process does not compromise voting secrecy. Incorporating a functionality for creating a Blockchain account in the application used for voting may be an alternative evaluated in future work;
- Finally, the calculation of the final result of the election is not part of the scope of this research work.

Acknowledgments: this research had the technical support of the Huawei-USP partnership. The smart contract code used in this paper is available at https://github.com/marceloMoro/Ethereum_Projects/blob/main/Election.sol.

REFERENCES

[1] ALOUL, F.; ZAHIDI, S.; EL-HAJJ, W. Two factor authentication using mobile phones. 2009 IEEE/ACS International Conference On Computer Systems And Applications, [s.l.], 2009, p.641-644.

[2] ÁLVAREZ-DÍAZ, N.; HERRERA-JOANCOMARTÍ, J.; CABALLERO-GIL, P. Smart contracts based on blockchain for logistics management. Proceedings Of The 1st International Conference On Internet Of Things And Machine Learning - Iml '17, [s.l.], 2017. ACM Press. DOI: 10.1145/3109761.3158384.

[3] BENLI, E. et al. BioWallet: A Biometric Digital Wallet. Icons 2017: The Twelfth International Conference on Systems, Venice-Italia, p.38-41, 10 mar. 2017.

[4] BUTERIN, V. Ethereum White Paper: A next generation Smart Contract e Descentralized Application Platform. 2013. Available in <https://github.com/Ethereum/wiki/wiki/White-Paper>. Last accessed August 25, 2021.

[5] DEEPAKUMARA, J.; HEYS, H.; VENKATESAN, R. FPGA implementation of MD5 hash algorithm. Canadian Conference On Electrical And Computer Engineering 2001. Conference Proceedings (cat. No.01th8555), [s.l.], p.1-6, 16 maio 2001. DOI: 10.1109/CCECE.2001.933564.

[6] ETHERSCAN. Ropsten Testnet Explorer. Site Ropsten Etherscan. https://ropsten.etherscan.io/. Last accessed August 25, 2021.

[7] FERREIRA, F. L. Blockchain e Ethereum: Aplicações e Vulnerabilidades. 2017. 37 f. TCC (Graduação) - Curso de Ciência da Computação, Instituto de Matemática e Estatística, Universidade de São Paulo, São Paulo, 2017.

[8] GOOGLE. O que é o Android. Site Android. Available in https://www.android.com/intl/pt-BR_br/. Last accessed August 25, 2021.

[9] HANIFATUNNISA, R.; RAHARDJO, B. Blockchain based e-voting recording system design. 2017 11th International Conference On Telecommunication Systems Services And Applications (TSSA), [s.l.], p.1-6, out. 2017. IEEE. DOI: 10.1109/TSSA.2017.8272896.

[10] ILLAKIYA, T. et al. E-voting system using biometric testament and cloud storage, 2017 Third International Conference on Science Technology Engineering & Management (ICONSTEM), Chennai, 2017, pp. 336-341. DOI: 10.1109/ICONSTEM.2017.8261305.

[11] INFURA. The Infura Ethereum API. Site Infura. https://infura.io/product/ethereum. Last accessed August 25, 2021.

[12] KSHETRI, N. and VOAS, J. Blockchain-Enabled E-Voting, in IEEE Software, vol. 35, no. 4, pp. 95-99, July/August 2018. DOI: 10.1109/MS.2018.2801546.

[13] LEE, K. et al. Electronic Voting Service Using Block-Chain. Journal Of Digital Forensics, Security And Law, [s.l.], p.123-135, 2016. Embry-Riddle Aeronautical University/Hunt Library. DOI: 10.15394/jdfsl.2016.1383.

[14] MCCORRY, P.; SHAHANDASHTI, S. F.; HAO, F. A Smart Contract for Boardroom Voting with Maximum Voter Privacy. Financial Cryptography And Data Security, [s.l.], p.357-375, 2017. Springer International Publishing. DOI: 10.1007/978-3-319-70972-7_20.

[15] NAKAMOTO, S. Bitcoin: A Peer-to-Peer Eletronic Cash System, 2008. Available in <https://bitcoin.org/bitcoin.pdf>. Last accessed August 25, 2021.

[16] JUNIOR, N. et al. Lightweight and Secure Publish-Subscribe System for Cloud-Connected Ultra Low Power IoT Devices. Journal of Communication and Information Systems, v. 36, n. 1, p. 100-113, 2021. DOI: 10.14209/jcis.2021.11

[17] PETCU, D.; STOICHESCU, D. A. A hybrid mobile biometric-based e-voting system. 2015 9th International Symposium On Advanced Topics In Electrical Engineering (atee), [s.l.], p.37-42, maio 2015. IEEE. DOI: 10.1109/ATEE.2015.7133676.

[18] PINHEIRO, J. M. Biometria nos Sistemas Computacionais: Você é a Senha. Rio da Janeiro: Ciência Moderna, 2008.

[19] SILVA, M. S.; SIQUEIRA FILHO, V. Biometria através de Impressão Digital. Cadernos Unifoa, Rio de Janeiro, v. 15, n. 1, p.19-28, 15 abr. 2011.

[20] SOLIDITY. Solidity V.0.8.7. Site Solidity. https://solidity.readthedocs.io. Last accessed August 25, 2021.

[21] TAPSCOTT, D.; TAPSCOTT, A. Blockchain Revolution: Como a tecnologia por trás do bitcoin está mudando o dinheiro, os negócios e o mundo. São Paulo: Senai-SP, 2017. 392 p.

[22] YAVUZ, E.; KOÇ, A. K.; ÇABUK, U. C and DALKILIÇ, G. Towards secure e-voting using ethereum blockchain, 2018 6th International Symposium on Digital Forensic and Security (ISDFS), Antalya, 2018, pp. 1-7. DOI: 10.1109/ISDFS.2018.8355340.

[23] WEB3LABS. Develop with Ethereum with the JVM. Site Web3j-sdk. https://web3j.io. Last accessed August 25, 2021.

Marcelo Moro da Silva is Master Degree in Computer Engineering from IPT and Graduate Studies in Web and Mobile Software Development from UFSCAR. Currently he is Senior Software Engineering Manager and has been working with projects envolving Mobile and Internet in financial sector since 2014 in big companies like Itau-Unibanco and Santander Bank.

Anderson Aparecido Alves da Silva received his PhD. Degree in Computer Engineering from USP in 2016, his Master’s degree in Computer Engineering from IPT in 2010, his MBA in Systems Analysis from FECAP in 1993 and his Data Processing degree from FIEO in 1991. Anderson has over 25 years of experience in the IT field. Currently he is a professor in several undergraduate and graduate courses in Sao Paulo and develops his second postdoctoral project in machine learning security at USP.

Norisvaldo Ferraz Junior is a PhD student at USP. He received his Master’s Degree in Computer Engineering from IPT in 2016, his MBA in the security of systems and environments from FASP in 2007 and his Bachelor degree in Systems Analysis in 2003. He works at FUNDACENTRO since 2005 as an analyst in science and technology. His research interests include security in Internet of Things, wireless sensor networks, machine learning, and intrusion prevention systems.

Eduardo Takeo Ueda holds a PhD in Electrical Engineering from EP/USP (2012), Master in Computer Science from IME/USP (2007), Specialist in Health Informatics from EPM/UNIFESP (2014), Graduated in Computer Engineering from UNIVESP (2019), and Graduated in Mathematics from UNESP (2000) . He is currently a Professor at the Senac University Center of São Paulo (SENAC), and Professor/Advisor of the Professional Master's Program in Applied Computing at the Technological Research Institute of the State of São Paulo (IPT). His lines of research are Cryptography and Information Security, with interest mainly in the following topics: Algorithms and Cryptographic Protocols, Authentication and Authorization Models, and Artificial Intelligence Techniques in Cybersecurity.

Fabio Dacêncio Pereira holds a Bachelor's Degree in Computer Science from the Euripides de Marília University Center (2002), a Master's in Computer Science from the Eurípides University Center of Marília (2004) and a Ph.D. in Electrical Engineering from USP (2009). He is currently a professor at the Centro Universitário Eurípides de Marília. He has experience in Computer Science and is currently the manager of the Marília Technological Innovation Center (CITec-Marília).

With a PhD in Transport Engineering (Poli /USP), a Master's in Computer Science (IME/USP) and a Bachelor's Degree in Computer Science (UFMT), **Alessandro Santiago dos Santos** is currently the business support manager for digital technologies at IPT. He is the coordinator and professor of the Professional Master's course in Applied Computing at IPT.

Adilson Eduardo Guelfi is a Doctor in Eletric Engineering at the Electronic Engineering Department (USP). He has over 22 years of experience in training, education, research, project development, professional and innovation skills in information security area. Until the year 2017, Adilson held the position of regular professor in the master’s program in Computer Engineering at the FIPT. Currently, Adilson has the position of Dean of Research and Graduate Studies at UNOESTE since 2014.

Sergio Takeo Kofuji holds a Bachelor’s Degree in Physics from the University of Sao Paulo (1985), a Master’s Degree in Electrical Engineering from the University of Sao Paulo (1988) and a PhD in Electrical Engineering from the University of Sao Paulo (1995), He is currently Professor Dr. at Polytechnic School of the University of Sao Paulo. Currently He has been focusing in Projects related to IoT, 5G communications, Smart Objects, Machine Learning and Artificial Intelligence, applied to Smart Cities and Smart Farms.