Journal Homepage: -www.journalijar.com

INTERNATIONAL JOURNAL OF ADVANCED RESEARCH (IJAR)

ArticleDOI:10.21474/IJAR01/13801
DOI URL: http://dx.doi.org/10.21474/IJAR01/13801

RESEARCH ARTICLE

PROPOSAL OF A DATA MODEL FOR MANAGING BLOKCHAIN-BASED TRUST BETWEEN ACTORS IN AN ELECTORAL PROCESS

Kanga Koffi[1], Kamagate Béman Hamidja[1], Coulibaly Tiekoura[1] and Oumtanaga Souleymane[2]

1. Laboratoire des Sciences et Technologies de l'Information et de la Communication, Ecole Supérieure Africaine des TIC, LASTIC-ESATIC, Abidjan , Côte d'Ivoire , 18bp 1501 Abidjan 18.
2. Laboratoire de Recherche en Informatique et Télécommunication, Institut National Polytechnique Félix HouphouëtBoigny , Côte d'Ivoire , Yamoussoukro.

Manuscript Info

Manuscript History
Received: 28 September 2021
Final Accepted: 30 October 2021
Published: November 2021

Key words:
Blokchain, Datasecurity, Modeling, Transaction, Cryptography

Abstract

In this paper we propose a tool (reference data model) for the improvement of trust, based on blokchain technologies, between the different actors of an electoral process. Our contribution focuses in a first step on the implementation of a data model having all these public attributes and thus public classes whose methods are coupled to cryptographic techniques. In a second step, we propose an underlying formalism of this model using a matrix representation of the different actors, the different transactions and the working criteria allowing to validate these transactions and blockchains. This formalism allows to find a transaction performed by one of the actors of the electoral chain and also the data block in which this transaction is validated. Also, an algorithm allowing to reinforce the trust is proposed.

Introduction:-

The various electoral processes, especially in Africa, are generally the object of confligate tensions due to a lack of trust between actors. Thus, the problem of a data model for the management of trust between electoral proccess actors deserves to be addressed in order to find an acceptablesolution.

In the field of information science, a data model describes the way data is represented in an organization, an information system or a database. For some authors [1], a data model represents a structural foundation, represented as a well-defined graphical characterization of a business information system.

As for electoral process, according to [2], it can be defined as all operations necessary for the proper conduct of elections. This process at various levels could include preparation of electoral lists, organization of electoral campaign, various formalities in preparation for the poll, the holding of polling stations and voting procedures, the modalities of centralization and counting of results, the training of "electoral officers" and the supervision of opinion polls or even the international observation of elections

Thus, in the field of data management of an electoral process, the execution of these different operations generates data that are stored in different forms (paper, flat files, etc.). To do this, some authors [3] have been able to set up processes for automation and storage of data from this process, all necessary and contributing to the strengthening of trust between actors, but when these data are derived from a model set up by professionals (technician in modeling

Corresponding Author:- Kanga Koffi
Address:- Laboratoire des Sciences et Technologies de l'Information et de la Communication, Ecole Supérieure Africaine des TIC, LASTIC-ESATIC, Abidjan , Côte d'Ivoire , 18bp 1501 Abidjan 18.

and engineer) and validated by the various actors, it is even better and more reassuring [13]. Also, in a sub-Saharan context, where it is recurrent to note crises of confidence during the electoral processes betweenelectoralproccessactors (organizers, voters, parties to vote, observers (NGOs, political parties and other various entities, ...)), it is therefore necessary to have consensual tools (data model) secured, recognized by all and not modifiable by one of the actors of the electoral chain. Hence the use of blockchain technologies coupled with a consensual data model.

The rest of our paper is organized as follows:
1. In Section 2, we present the state of the art
2. In section 3, we set out our problematic
3. In Section 4, we illustrate our contribution
4. Section 5 is devoted to a discussion and we end with a conclusion in section 6.

In the latter, we will outline some perspectives.

State of the art

Blockchain-based e-voting security model

The security of data on the Internet is mostly achieved through the use of passwords and double authentication, electronic certificates, digital signatures and integrity checks by hashing. Double authentication, one of most used for data security consists in sending short message or e-mail confirmations to the different users of the applications. In the literature, this way of securing has shown its limits in terms of security. This is why some authors propose the use of blokchain technologies coupled with cryptographic techniques. Indeed, with this coupling of techniques the possible modifications of data by a third party of the blokchain network proves to be painful or even impossible. This guarantees the integrity and viability of an authentication. Considering the possibilities [5] offered by blokchain and its coupling with cryptography, this technology is increasingly integrated in several fields. These include banking, insurance and most recently electronic voting.

In [17], the authors start from the observation that in a network-based voting system, the problems of reticular security (of the networking) and the confidentiality of the communications increased. Faced with these observations, the security of electronic voting is becoming a popular concern and is urgent and necessary. Therefore, to overcome these concerns, YI et al. present techniques to exploit the blokchain in P2P networks in order to improve the security of electronic voting and the data transiting on this network. To achieve their goal, they first propose a synchronized model of voting data based on blokchain technologies in order to avoid possible tampering. Secondly, a voter identity proofing model using elliptic curve cryptography techniques is used to ensure authenticity and non-repudiation. Thirdly, an opt-out model allowing voters to change their vote from a predefined date is proposed. Finally they integrate the above different models to result in a blokchain based e-voting system for multiple candidates.

Electronic voting systems [6][8]

In 2017, authors [6] proposed a contribution to the use of blokchain technologies for securing elections. In their paper, they implemented a decentralized e-voting protocol without the existence of a trusted third party because according to them, existing e-voting systems use blokchain for storage and disregard the actual voting process. That is why they propose in their paper a protocol whose scheme includes 4 steps that are:
1. The execution of a voting application: this execution generates a private key and a public key for security management
2. Voter identification (constituting a signature of the vote cast)
3. The actual vote (constituting the crucial information, the object of the vote) which is recorded
4. Validation of voting data using a consensus algorithm on a server.

In [13], a blokchain-based voting audit system is implemented. It aims, to facilitate audits on the methods and approaches of existing electronic voting systems (dedicated electronic voting machines, optical scan voting machines, manual ballot printing, Internet voting). To achieve these objectives, the authors present some advantages (voter identification and authentication, voter privacy, accuracy, transparency and verifiability, ballot integrity, availability) and disadvantages (related to the use of passwords and double authentication) of these voting systems. They conclude by looking at the auditing aspects of the voting processes and the data generated. The results of their

work are summarized in a tool set up called (ABVS) using blokchain in its operation which includes three stages (the preparation of elections, the vote itself, the counting and verification of results).

In [15], KADAM, Snehal, CHAVAN, Khushaboo, KULKARNI, Ishita, et al. propose a review of some works on electronic voting using blokchains. In this paper, the authors consider blokchain technologies as usable services, applicable to the security and implementation of voting processes and data. They present a comparative study of these different works and propose an implementation of them taking into account the electoral context of each nation.

Some data models for storing voting data

Data model for managingelectionsacross multiple organizations [18]

This workiscarried out within the framework of the implementation of an application allowingorganizations to electtheirrepresentatives online. The data model resultingfromthiswork (figure 1) is a relational model, includingnine (9) basic files that are :

1. Organization: to store all organizationsinitiatingelections
2. Election:represents the actualvoting
3. Role:allowsyou to determine the differentroles to befilledduringelectoralprocesses
4. Candidate:represents the different candidates in electoralcompetition
5. Voter:represents all the people who have to elect the candidates throughtheirchoice. Thesechoices are stored in the file called "**vote".**
6. ROLE_APPARTENIR :this file allows a candidate to change rolebetweenelections. It ischaracterized by the attributes (**id_organization, election_id, roleid)**

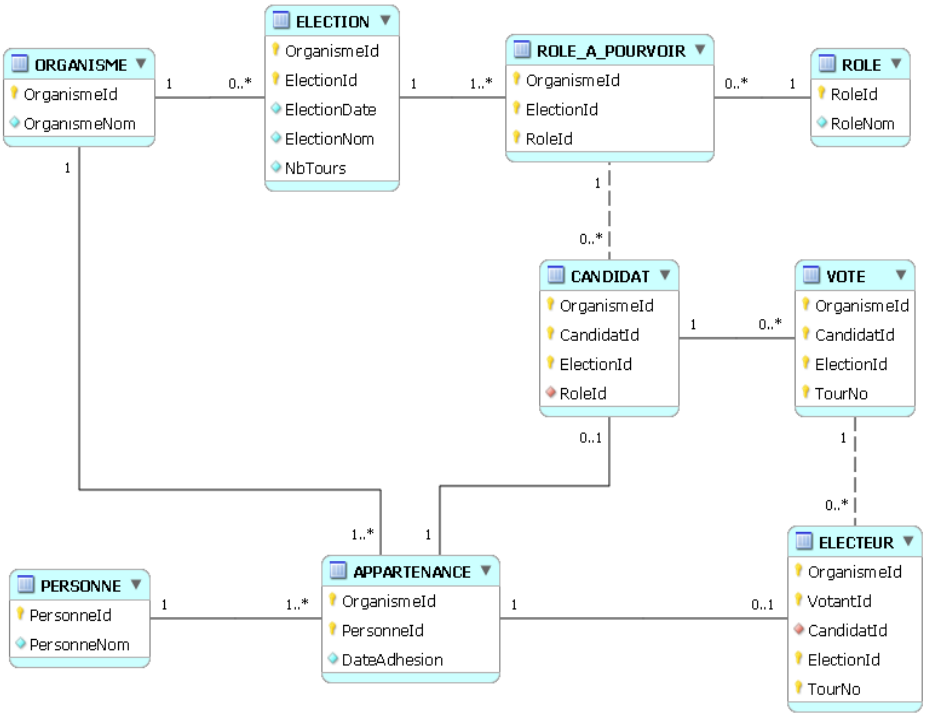

Figure 1:- Agency Voting Management Data Model.

The limits generated by this model are linked to the type of model and the type of data stored. This model being a relational model, it implements then contains in its semantics all the limits and defects related to the relational model. Namely:

- data locality: data from this model can only be stored on a server characterized by an **IP** address and accesses. This situation means that the data can be exposed and possibly modified by malicious third parties.
- scaling (in data size, geographically)
- Shared memory, Shared disk, Shared nothing

As for the limits related to storage, this model does not specify in which form (clear text, encrypted, hashed etc..) the data will be stored. Also the conditions of validation of the vote (start and end time, number of voters, voting parties are not illustrated) are not defined

Voting management systems: the case of Haiti [19]

This work is the result of work carried out by students of the University of Picardie (France). It represents the static part of the various models carried out for the implementation of an integrated system of electronic voting. On this model we see six (6) classes as well as the different methods associated with these classes (figure 2).

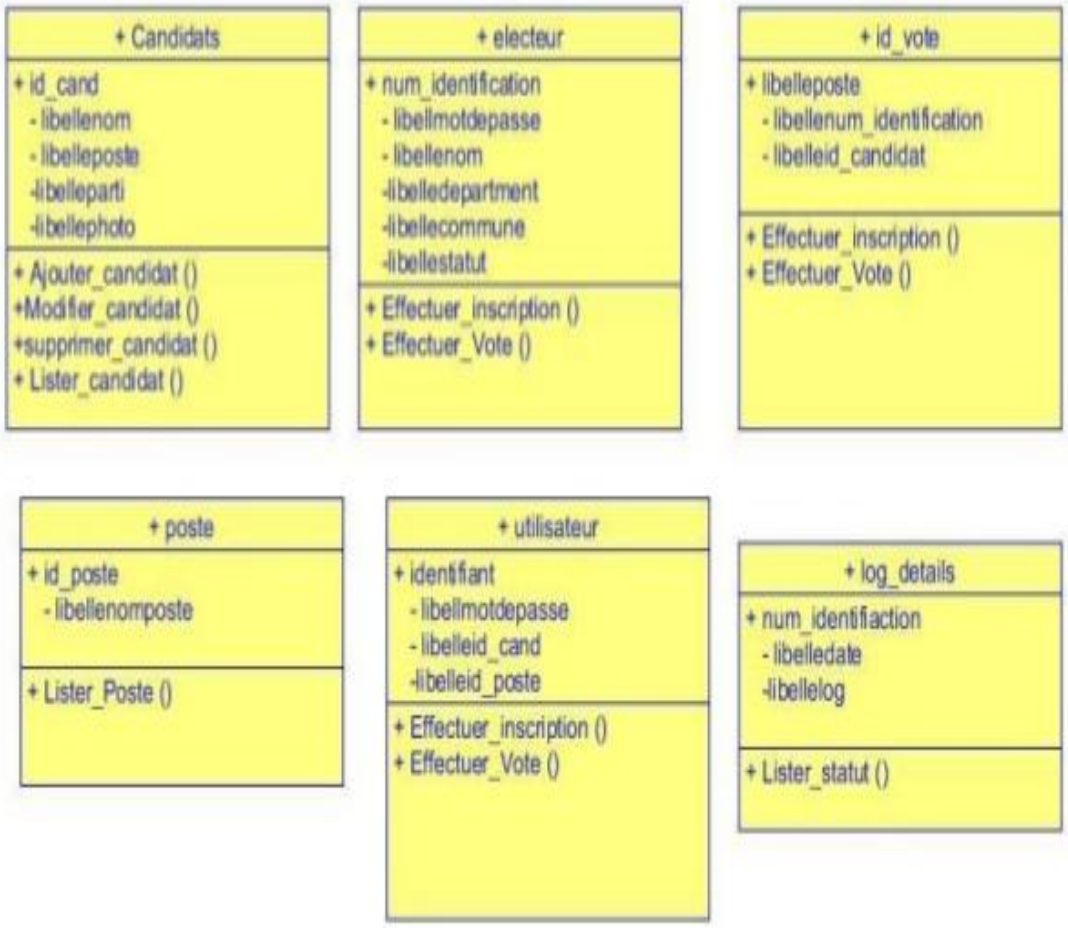

Figure 2:- Data model file from [19].

On this model, no element allowing to secure the stored data appears. Also in this work, the authors propose an architecture that does not allow a distributed storage of the model data. This work does not mention the voting criteria, and participation in the vote. All things that can taint the credibility of the vote and therefore the lack of confidence between the actors and also in the quality of data that this model could store.

A remote voting system [20]

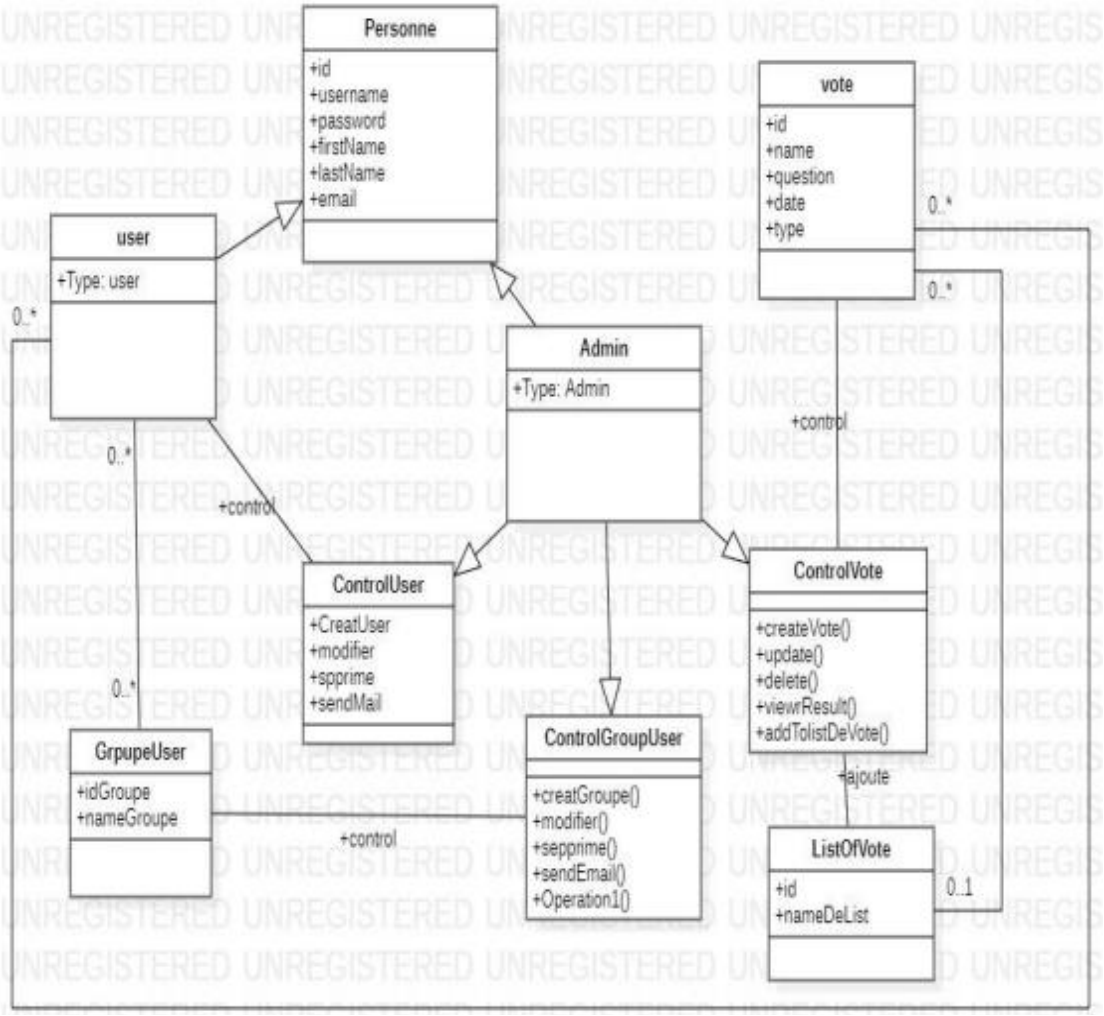

Figure 3:- Data model for remote voting [20].

On this remote voting management system, the authors present elements allowing the management of the participants in the voting system. These are the persons represented by the classesnamed“persons”, “admin” and “user”. The limits of this system lie in the appearance of duplicates appearing in the classes persons and user. This does not facilitate trust between these doubly represented actors. Also, this model does not specify the criteria for voting and validation of the vote of each actor. Although the act of voting is materialized by the class "Vote" on this model, the paper does not specify the conditions relating to the securing of each vote.

Limitations of different blokchain-based e-voting technologies

In view of the variousworkspresentedabove, several limitations emerge. Indeed, to ourknowledge, theseworks do not takeintoaccount the aspects related to data storage. These aspects couldallow to manage the opinions of the voting parties on the quality of the voting process. Also the voter being the onlyactor in the vote in thesevariousworks ;they use theseprotocols set up withoutknowingwhatthisprotocolwillproduce as output data in terms of quality (possibility of corruption and loss or disappearance). In thislogic, the failure to takeintoaccount the freedom of the vote (time limits) whichremainsverycomplexlegallythan in itsrealizationisalsonoted. Since the aspects related to data storage are not addressed, itgoeswithoutsayingthat the aspects related to the underlying data models are not addressedeither. In addition, the lack of conformity of the proposed architectures with the electoralsystems (in terms of components and functionality), the compatibility of the security system capable of triggering confidence between the various stakeholders, the nature and form of data likely to guarantee the credibility of the electoral process are shortcomings due to the absence of a reference model (in terms of data).

Problems
Fromthisliteraryreview, itappearsthat excellent researchwork has been carried out. This work has led to the implementation of electronicvotingtools, security of said ballots and votes, hybridmethods for security (blokchain and cryptography) of whichwe have presented the mostrepresentative in terms of theirfunctionality and interesting architecture. However, theseworks and toolsproducedstillsufferfromevilsmakingit impossible to strengthen the trust between the actors in the various ballots and votes. The question that arises in these conditions is the following:isthere a reference data model aroundwhichthesedifferentworks and toolscouldbeimplementedsothat trust between the differentactors can beimproved? All thisisnecessary for a smooth and peacefulelectoral process in ourdeveloping countries.

Contribution
literature review showhigh level of quality of the research work carried out with a view to securing electoral processes. However, issues concerning data storage from various processes are not taken into account. Among these are the models underlying the various tools and applications.

To this end, we present a tool (data model) (Figure 2) that could be consensual between the different actors involved in an electoral process in order to guarantee their confidence.

Contribution modelling
Our model consists of the following classes:
"Actors": this class allows to manage all the actors of a poll. This management consists in their creation, their update (using the **modification()** and **deletion()** methods**)** if possible, errors appear in their characteristics. Thus, each actor is characterized by its identifier (**Id_acteur**). This identifier is unique for each actor. Also, the actors in the voting process and operation are characterized by a registration date (**registration_date**), a name (**actor_name**) and possibly a contact (**actor_contact**)

"Voter": he is the main actor in the voting process. His action, his participation or his non-participation constitutes a transaction to be secured, to store data and to publish to all the actors of the blockchain. He is characterized by an identifier (**id_voter**), a name (**name_voter**), a location (the place of voting or residence)

"Organisers"(this is an actor in charge of organising the vote. As such, in an e-voting context, he will be in charge of the physical existence of the voters, of their participation in the voting process and also of the management of the logistics related to the vote). The organizer is characterized by an identifier (**id_org**), a description (**description_org**). Its intervention is made public and validated by all the actors: guarantee of transparency and equity in the treatment.
"Observer" (it is an actor in charge of the conformity of the various transactions carried out by the voters and the organizers with regard to the various occurrences of work criteria (work_criteria). This class is characterized by an identifier (**id_observor**), an observation start date (**debut_observ**) and an observation end date (**fin_observ**).

"The work criteria"(voting criteria). This class defines the voting criteria or the set of criteria related to a transaction initiated by an actor. The validation of transactions in a block is conditioned by the strict respect of the set of criteria related to this transaction which itself is visible and monitored by all the actors. This set of criteria is characterized by an identifier (**id_critere**), a description (**description_critere**) and a value (**val_critere**). This set is subject to creation, modification or deletion accepted by all the actors. Therefore, in its content, it becomes consensual and therefore a source of lesser contestation through the signs (**+) in** front of each attribute and method.

"The transaction"(the vote itself) represents the actual voting operation. It is linked to the actors so that all actors can recognize the different transactions by each party.

"The blockchain "(storing the result of the transaction). Each block is characterized by an index (**block_index**), a hash of the current block **(hash**) representing a small string of bits characterizing the block; this hash is obtained using a hash function or method, a creation date (**date_creation**) which is the date on which the block is created. This date is used to time-stamp the blocks implemented during a poll, the data of the block (**data_block**), a signature (**signature_block**) allowing us to know the actor who initiated a validated transaction in the block, the set of proofs of work (**proof_of_work**) and the hash of the previous block (**hash_precedent**) also obtained by our hash function.

All these classes are constituted in their formalisms of 3 parts (figure 4) which are:
the name of the class, the list of attributes and the methods associated with these classes. The symbols (**+) in** front of the attributes and methods mean that these elements are accessible and visible by all the actors. All things guaranteeing trust through the identification, the proof of works of the actors, their different transactions and the blocks in which these transactions are validated. Also, the blokchain being a distributed database, it allows to publish, make available the validated data through replication functions on the different nodes of the chain

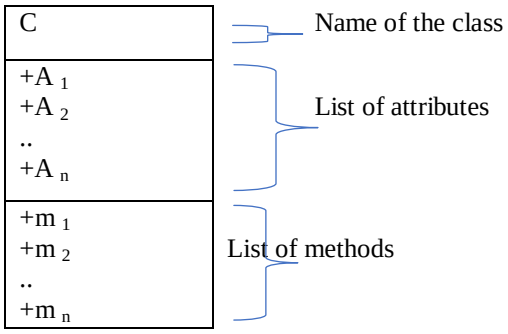

Figure 4:- Class formalism.

In Figure 5, the set designated by the term "multiplicity" represents the links or relationships between these classes. In fact, this term designates the minimum and maximum number of times that a given class participates in a multiplicity. Formally, it is written:

Multiplicity = (min, max) and takes these values in the set **£ = (1.. *, 1..1, 0..*, 0..1).**

In figure 5, the relation designated in **A** is a reflexive relation. It materializes the fact that the present block depends on the previous block. In other words, the hash of the current block depends on the hash of the previous block. A modification of the block number (i) also leads to the modification of its hash and the following hash(i+1) .Thus from modification to modification one ends up with an operation which becomes computationally impossible because it takes enough time.

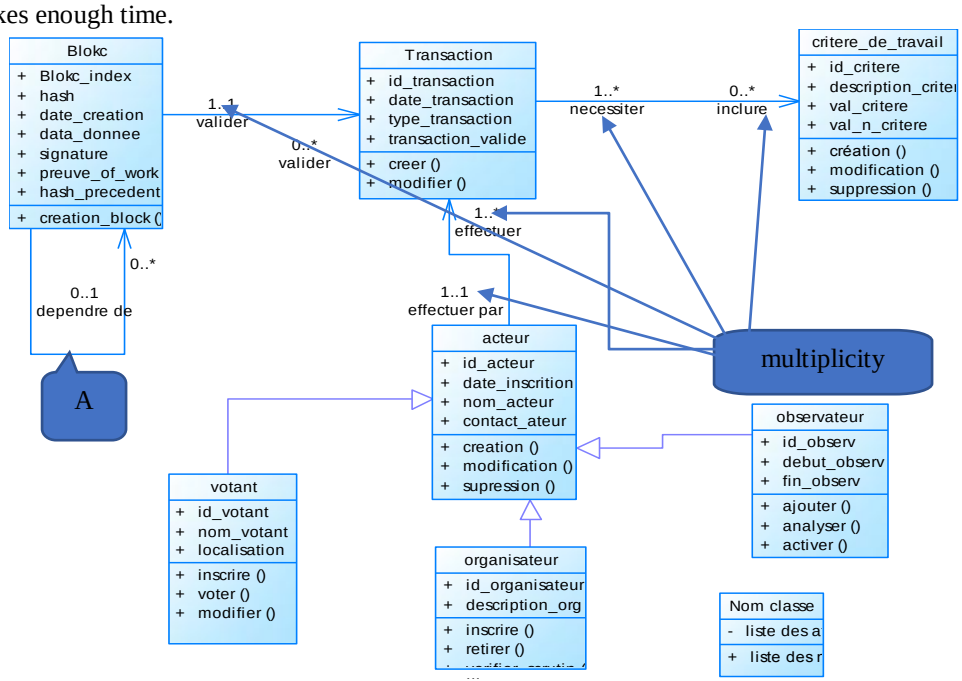

Figure 5:- Trust management data model.

Formalization
Formally, this trust management model is represented by four (4) major sets defined as follows:

- A = all the actors involved in the voting process. It is made up of the voters, the organisers and the observers.
  $A = \{A_1, A_2, A_{3,\dots}A_n\}$
- T = set of transactions performed by the actors in the process
  $T = \{T_1, T_2, T_{3,\dots}T_k\}$
- C = the set of criteria or rules that govern the voting process. This set of rules represents a consensual basis defining the criteria of trust between the actors whose strict respect must be guaranteed.
  $C = \{C_1, C_2, C_{3,\dots}C_Q\}$
- B = set of blocks in which the transactions of set T are validated. In this set, the hash of each block depends on the hash of the previous block. It is defined as follows:
  $B = \{B_1, B_2, B_{3,\dots}B_j\}$.

Thus, we define the following matrices:

Formalization of transactions carried out by an actor

$AT = (A_s, T_d)$ where $(1 \le s \le n, 1 \le d \le k)$, n is the maximum of actors and k the maximum of transactions .

$$AT = (A_s, T_d) = \begin{cases} 1 \text{ if } A_s \text{ carried out transaction } T_d \\ \text{while respecting the work criteria related to the} \\ \text{transaction} \\ 0 \quad \text{otherwise} \end{cases}$$

For example;

Let A be a set of 4 actors and T a set of 4 transactions (here we choose a number of actors equal to the number of transactions because an actor can perform only one transaction (vote).

So these conditions we obtain a square matrix which is presented as follows:

$$AT = (A_s, T_d) = \begin{pmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 \end{pmatrix}$$

On this matrix the rows represent the actors and the columns represent the transaction

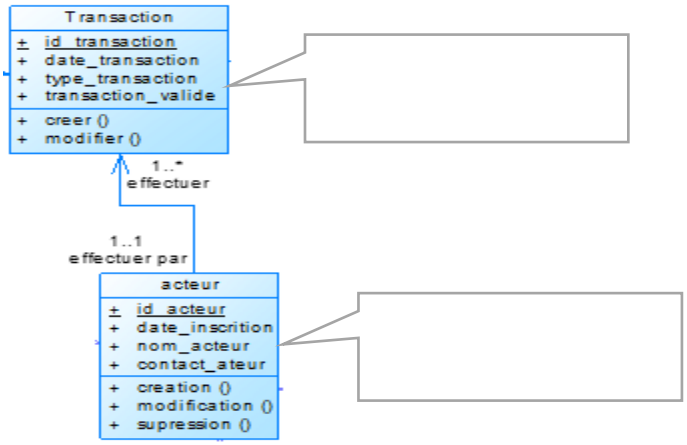

Figure 6:- Sub-model representing the transactions made by an actor.

Formalization of transaction validations by a set of criteria

- $TC = (T_d, C_x)$ with $(1 \le d \le k, 1 \le x \le q)$ k is the maximum of the transactions and q is the maximum of the work criteria to be respected and .

$$TC = (T_d, C_x) = \begin{cases} 1 \text{ if } T_d \text{ respect the work critteria} \\ Cx \\ \text{related to the transaction} \\ 0 \quad \text{othewise} \end{cases}$$

Illustrative example:

We always take the same number (4) of transactions in the AT. matrix and a number of six work criteria that these transactions must meet. Thus, our matrix $TC = (T_d, C_x)$ with $1 \le d \le 4, 1 \le x \le 6)$ is as follows;

$$TC = (T_d, C_x) = \begin{pmatrix} 1 & 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 1 & 1 & 1 \end{pmatrix}$$

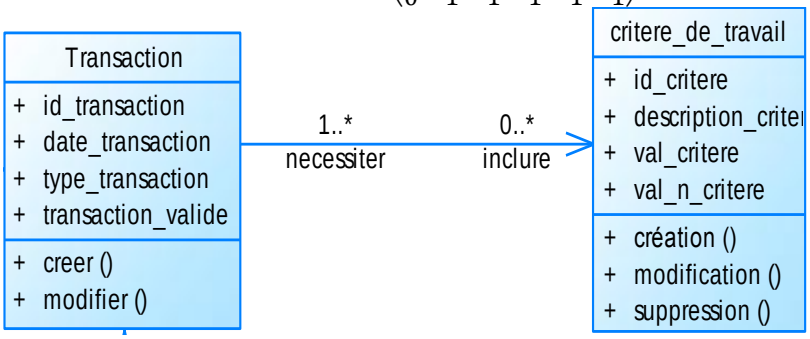

Figure 7:- Representative sub-model of our formalization.

Formalization of transactions stored in blocks

- TB =(T_d, B_i) where $(1 \leq d \leq k, 1 \leq i \leq j)$ k is the maximum of transactions and j is the maximum of blocks in the voting chain.

- $$TB = (T_d, B_i) = \begin{cases} 1 \text{ if } T_d \text{ is validated in the block } B_i \\ 0 \text{ otherwise} \end{cases}$$

Also since a transaction is committed archive in one and only one block, we must have a number of transaction equal to the number of block (so a square matrix)

For example:

$$TB = (T_d, B_i) = \begin{pmatrix} 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

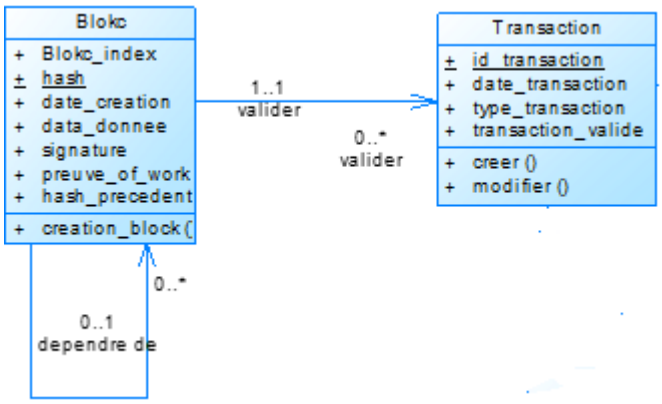

Figure 8:- Representative sub-model of transaction hashing in blocks.

Formalization and representation of transaction traces carried out by an actor

To know that a transaction performed by an actor $\mathbf{A_s}$is validated in a block $\mathbf{B_i}$according to the given criteria $\mathbf{C_x}$(representing the constraints), it is necessary that:

$(AT = (A_s , T_d) = 1)$ and $(TB = (T_d , B_i) = 1)$

Constrained that $(TC = (T_d, C_x) = 1)$

$$AT = (A_s , T_d) = \begin{pmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 \end{pmatrix} \text{ and } TB = (T_d, B_i) = \begin{pmatrix} 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

$$\text{Constrained that } TC = (T_d, C_x) = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 \end{pmatrix} = 1$$

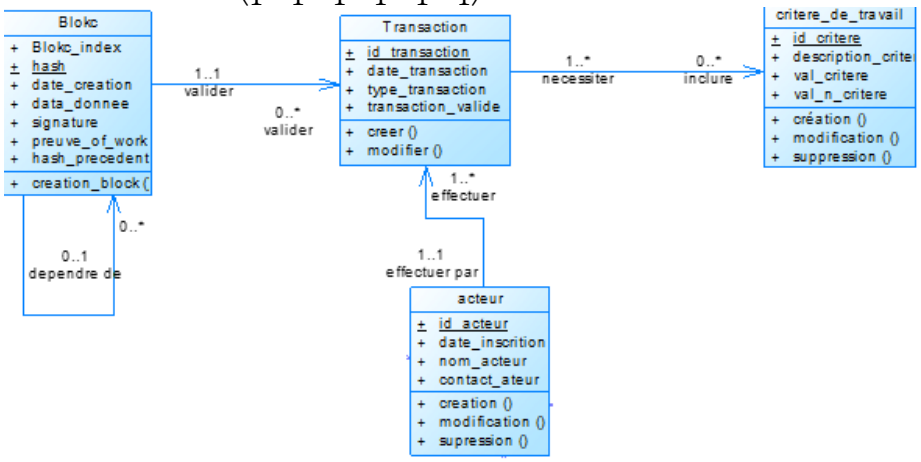

Figure 9:- Representative sub-model of the trace of a transaction performed by an actor

Proposal of algorithms

First algorithm: detection of the list of criteria involved in the validation of transactions carried out by a given actor

Description

Our algorithm takes as input the list of actors in the vote and produces as output two lists C' and T' containing respectively the criteria that participated in the validation of the elements of the list of transactions. These steps are the following:

- **Step 1: selection of actors to vote**
- **Step 2: for each actor, list all transactions made**

(These first two (2) steps take into account our formalization in section 4.2.1 as well as the associated sub-model (through the "actors" and "transaction" classes)

- **Step 3: for each transaction to list the criteria that contributed to its validation**

This step is based on our formalization in 4.2.2 and the associated sub-models (through the "transaction" and "criteria" classes)

The pseudo code of our algorithm is as follows:

Algorithm: list_criteria_by_transaction
Input: A = list of actors
Output: T'= transaction list = Ø
C'= list of working criteria per transaction= Ø
Begin
/Actor's career path
For any A_s belonging to A do
//transaction flow
For all T_d belonging to T do
If (A_s, T_d) = 1 then
Display (A_s performed transaction T_d)
//course of work criteria
For any C_x apartment at C do
If (T_d, C_x) = 1 then
Display ("transaction C_x is validated by criterion C_x")
C' =C'+C_x

Otherwise
Display ("transaction T_d is not released by C_x")
End if
End for
T'= T'+T_d
Otherwise
Display ("A_s did not perform the T_d transaction")
End if
End for
End for
End

Second algorithm: List of blocks and transactions
Description
This algorithm takes as input the list of transactions produced by the previous algorithm '**list_criteria_by_transaction**. It outputs the list of blocks, the number of transactions and the number of blocks involved in securing these transactions.
It includes the following steps:

- **Step 1: selection of transactions produced in algo 1**
  Verification of the constraints linked to the work criteria that validated each transaction. This step takes into account our formalization in point 4.2.3
- **Step 2: block selection**
  Search in the blocks, the blockchains having validated this transaction as well as their signature. This search is done taking into account the constraints associated with our formalization in 4.2.4 and the data sub-model

Algorithm: list_archive_transaction
Input: T' = list of completed transactions related to work item C'.
Output: block list = Ø (containing transactions and previous blocks)
Begin
//transaction flow
For all T'_abelonging to T' do
//block course
For all B'_ibelonging to B' do
If (T'_a,B'_i) = 1 then
Display("transaction $T'_a$is in block $B'_{I''}$)
B' = B'+ B'_i
otherwise
Display ("Transaction$T'_a$is in block $B'_{I''}$)
endif
End for
Display ("engaged blocks are B'")
End for
End

Discussion:-

The use of blokchain technologies coupled with cryptographic techniques for the management of trust deserves that the different actors of the different electoral processes are well identified. In our case the actors are the voters, the organizers and observers. Also, the working conditions (proof of work) and participation must be agreed upon by each actor. Here these conditions are made public by assigning public scopes to the different attributes of the different classes of our reference model. Thus, the public character of all the attributes and methods of the classes of our model also confers a public character to the classes and thus to the model. Moreover, the different nodes intervening in the block chain must be available and access the block network.

Under these conditions our proposed data model will allowbetter storage to make data available to every electoral actor and better durability.

The adoption of our reference model for the eventual implementation of management tools for electoral processes could reduce and eliminate possible disputes during and after the various elections. For sub-Saharan countries, its adoption could increase confidence between the various actors and stabilize disputes.

At the technical level, our reference model could give an idea of the size of the data of an electoral process in order to guide the different organizers in the choice of hardware and implementation platform (programming and storage). It could be implemented in structured (relational, spreadsheet) and unstructured (NoSQL) databases

Conclusion And Outlook:-

The objective of the present work is the implementation of tools (data model) for the improvement of trust between the different actors of an electoral process based on blockchain technologies coupled with cryptography. To do this, we reviewed the most representative works to our knowledge.

From these works, we have identified the limitations and have made our contribution which consists in the adoption of a reference data model made up of classes and methods that could lead to an improved trust between the different electoral actors.

The future work concerning the implementation of this reference model could turn to the use of big data technologies for the storage and availability of data. Indeed, these technologies allowing to store and make available large masses of data, could be useful for the management of the confidence of the various actors in these electoral data.

Bibliography:-

1. https://www.memoireonline.com/10/13/7478/m_Processus-electoral-et-contestation-de-resultat-en-Afrique-subsaharienne-Cas-de-la-RDC1.html accessed on 04/03/2020
2. https://fr.talend.com/blog/2017/07/31/conception-de-modeles-de-donnees-et-bonnes-pratiques-partie-2/ accessed on 04/03/2020
3. IONESCU, Alexandra, et al. Voting and Territorial Reform in Central and Eastern Europe. Electoral administration and politics in post-communist Romania. Studia Politica. Romanian Political Science Review, 2012, vol. 12, no. 4, pp. 539-554.
4. ALI, Muneeb, NELSON, Jude, SHEA, Ryan, et al. Blockstack: A global naming and storage system secured by blockchains. In: 2016 {USENIX} Annual Technical Conference ({USENIX}{ATC} 16). 2016. p. 181-194.
5. CRESITELLO-DITTMAR, Ben. Application of the blockchain for authentication and verification of identity. Independent Paper, 2016.
6. LIU, Yi and WANG, Qi. An E-voting Protocol Based on Blockchain. IACR Cryptology ePrint Archive, 2017, vol. 2017, p. 1043.
7. MOURA, Teogenes and GOMES, Alexandre. Blockchain voting and its effects on election transparency and voter confidence. In : Proceedings of the 18th annual international conference on digital government research. 2017. p. 574-575.
8. KSHETRI, Nir and VOAS, Jeffrey. Blockchain-enabled e-voting. IEEE Software, 2018, vol. 35, no. 4, pp. 95-99.
9. AZOUVI, Sarah, MCCORRY, Patrick, and MEIKLEJOHN, Sarah. Betting on Blockchain Consensus with Fantômette. 2018. 1805.
10. PANJA, Somnath and ROY, Bimal Kumar. A secure end-to-end verifiable e-voting system using zero knowledge based blockchain. IACR Cryptology ePrint Archive, 2018, vol. 2018, p. 466.
11. ADIPUTRA, Cosmas Krisna, HJORT, Rikard, and SATO, Hiroyuki. A proposal of blockchain-based electronic voting system. In: 2018 Second World Conference on Smart Trends in Systems, Security and Sustainability (WorldS4). IEEE, 2018. p. 22-27.
12. COOLEY, Rafer, WOLF, Shaya, and BOROWCZAK, Mike. Blockchain-Based Election Infrastructures. In: 2018 IEEE International Smart Cities Conference (ISC2). IEEE, 2018. pp. 1-4.
13. PAWLAK, Michał, GUZIUR, Jakub, and PONISZEWSKA-MARAŃDA, Aneta. Voting process with blockchain technology: auditable blockchain voting system. In International Conference on Intelligent Networking and Collaborative Systems. Springer, Cham, 2018. pp. 233-244.
14. PATEL, Hiren M., PATEL, Milin M., and BHATT, Tejas. Election Voting Using Block Chain Technology. International Journal of Scientific Research and Review, 2019, vol. 7, no. 05, pp. 1-4.

15. KADAM, Snehal, CHAVAN, Khushaboo, KULKARNI, Ishita, et al. Survey on Digital E-Voting System by using Blockchain Technology. International Journal of Advance Scientific Research and Engineering Trends, 2019.
16. JAIN, Harsh, OAK, Rajvardhan, and BANSAL, Jay. Towards Developing a Secure and Robust Solution for E-Voting using Blockchain. In: 2019 International Conference on Nascent Technologies in Engineering (ICNTE). IEEE, 2019. p. 1-6.
17. YI, Haibo. Securing e-voting based on blockchain in P2P network. EURASIP Journal on Wireless Communications and Networking, 2019, vol. 2019, no. 1, pp. 1-9.
18. https://www.developpez.net/forums/d1343738/general-developpement/alm/modelisation/schema/gestion-votes/ (expires on 07/08/2020)
19. https://fr.slideshare.net/ingcarlosphilippe/conception-et-ralisation-dun-systme-intgr-de-vote-electronique-cas-haiti (expires on 07/08/2020)
20. Yousfi, I. (2021). Design and implementation of a remote voting system. Case study: Department of Computer Science (Doctoral dissertation, FACULTY: MATHEMATICS AND COMPUTING DEPARTMENT: COMPUTER SCIENCE-OPTION: RTIC).