ISSN 1814-4225 (print)
ISSN 2663-2012 (online)

UDC 004.45 doi: 10.32620/reks.2026.2.15

Maksym HOLIKOV[1], Mariia RODINKO[1], Dmytro UZLOV[1], Artem BRONNIKOV[2]

[1]*V. N. Karazin Kharkiv National University*
[2]*Kharkiv National University of Radio Electronics*

A BLOCKCHAIN-BASED ELECTRONIC VOTING SYSTEM WITH AUTOMATED LEADERLESS BFT CONSENSUS

The article discusses *a decentralized electronic voting system based on blockchain technology.* ***This study aims*** *to improve the performance and fault tolerance of blockchain-based electronic voting systems by introducing the Automated Leaderless Byzantine Fault Tolerance (AL-BFT) consensus protocol.* ***This study aims*** *to develop and evaluate an electronic voting system model that applies the proposed AL-BFT consensus mechanism in a permissioned peer-to-peer network.* ***The methods*** *used include computer modeling of a peer-to-peer (P2P) network, implementation of a decentralized ledger, and experimental load testing of the consensus protocol. System performance is evaluated using key metrics, such as transaction latency, throughput (requests per second), fault tolerance threshold, and scalability.* ***The study results*** *include the development of a conceptual architecture for the electronic voting system, the identification of its core components, and the analysis of their interactions to ensure data integrity and the reliability of voting results. At each stage of the electoral process, data security is considered, and additional protection mechanisms are analyzed to enhance system robustness. Eliminating the leader election phase from the consensus process is a key feature of the proposed approach, thereby reducing coordination overhead and enabling more efficient agreement among nodes. The proposed AL-BFT protocol reduces transaction latency and improves throughput while maintaining the fault tolerance level of traditional Byzantine Fault Tolerance-based approaches. The results confirm improved efficiency compared to classical leader-based consensus mechanisms, particularly in small permissioned blockchain networks.* ***Conclusions.*** *A practical implementation of the system has been developed and tested under real simulated load conditions. The proposed solution ensures stable system operation and reliable consensus formation. The system can be effectively applied to university elections, organizational voting, and other scenarios that require transparency, security, and manipulation resistance*

Keywords*: electronic voting system, blockchain, consensus protocol, BFT protocol.*

1. Introduction

1.1. Motivation

In the digital age, electronic elections are becoming increasingly important in democratic processes. However, modern voting systems face significant challenges, including ensuring voter privacy, transparency in the electoral process, and cyberattack protection. Traditional centralized solutions are vulnerable to manipulation, whereas decentralized approaches have not yet achieved sufficient efficiency for large-scale implementation.

Electronic voting systems are classified based on several key criteria, including voting location, technology used, and security and privacy levels. The article [1] is a systematic cartographic study of current trends in securing electronic voting systems using modern cryptography and blockchain technologies. This study focuses on how these technologies contribute to increasing the security, transparency, and reliability of electoral processes. The study [2] focuses on the impact of decentralization on electronic voting systems, examining how decentralized technologies, such as blockchain, can transform election organizations. The paper [3] surveys recent trends in e-voting systems, focusing on the potential of blockchain technology to enhance security, transparency, auditability, and voting integrity. This survey addresses a gap in the existing literature by examining the latest advancements in blockchain-based e-voting, including architectural considerations, global trends, and ongoing development challenges. The significance and requirements of blockchain-based electronic voting systems are discussed in [4], highlighting the potential advantages and challenges of this technology. The article [5] explores the security challenges of IoT, including cloud and blockchain technologies, post-quantum cryptography, and evolutionary methods, considering how these technologies can be used to protect electronic voting systems. The study [6] focuses on a financial management system for cryptocurrencies that facilitates better collaboration and may have applications in financing and managing decentralized voting systems. The study [7] investigated how citizens perceive blockchain-based e-voting within the context of smart

cities. The findings reveal that citizens' perceptions of its ease of use and usefulness strongly determine their intentions to use this voting method.

Each approach has advantages and limitations that affect its application in real-world conditions. An analysis of the different types of electronic voting systems is presented in the State-of-the-Art subsection.

Blockchain, a technology characterized by high transparency and strong data protection, is becoming a promising tool for overcoming the challenges posed by centralized systems. The central principle of blockchain is decentralization, which distinguishes it from traditional centralized systems. In traditional systems, data are stored on a central server controlled by a single organization. On the other hand, the blockchain does not require a centralized authority. Each network participant (node) has equal access to information, and copies of the blockchain are simultaneously stored on many computers.

Notably, transparency in the blockchain is ensured because all participants have access to the transaction data stored on it. In public blockchains, anyone can view the data, ensuring complete openness. However, the blockchain maintains confidentiality even with this transparency, as users' personal data is replaced with cryptographic addresses. This allows for a balance between the system's openness and the protection of personal data.

The consensus protocol is a key component of a blockchain system. Consensus mechanisms ensure that participants agree on the only true version of the blockchain's data. In particular, the Byzantine Fault Tolerance (BFT) consensus protocol enables the system to maintain stability despite malicious actions by some network nodes.

Blockchain-based electronic voting systems can address several pressing issues, including ensuring voter privacy, preventing voter fraud, reducing election costs, and enhancing access to the electoral process for individuals with disabilities or those residing abroad. In addition, such systems can help increase public trust in election results, as participants can verify and attest each step of the process, and the data will be available to all stakeholders.

The implementation of this model will not only showcase the capabilities of technology in the electoral process but also highlight the electoral process's capabilities. It will also serve as a basis for further research and development in this area.

This will open new horizons for the development of democracy and for improving social processes. Additionally, the research and development of an efficient, secure, and transparent blockchain-based system is an important step toward creating a new generation of electoral platforms that ensure equal access to voting and minimize fraud risk.

1.2. State of the art

In many countries, centralized voting systems are traditional and involve storing and processing data on a single server.

They are characterized by a single control center that ensures that all information is stored and processed on centralized servers under the control of a single organization. This enables you to respond to technical malfunctions and control the process quickly.

Such systems provide a high level of data security through encryption and multi-factor authentication; however, they also create a single point of failure, which can put the entire process at risk in the event of an attack.

Centralization simplifies management and reduces the resources needed to maintain the system, making it easy to implement. Additionally, such systems ensure fast vote counting, enabling the prompt announcement of election results.

The Estonian i-voting system is one of the most famous centralized e-voting systems [8]. Since 2005, this system has allowed citizens to vote online during national elections. Reliable user identification through digital ID cards ensures the integrity of the voting process, allowing every citizen to participate in elections from anywhere in the world.

Centralized solutions ensure high-speed, efficient data processing by automating all voter registration and counting processes. However, in a centralized model, there is always a risk of compromising the central server, leading to manipulated results. In addition, centralization creates a "single point of failure" – technical problems or attacks on the central server can paralyze the electoral process [9].

Moreover, trust issues may raise concerns among voters about the objectivity and confidentiality of the vote, as the central body has access to all information, potentially undermining confidence in the system.

Decentralized electronic voting systems based on blockchain technology represent an innovative approach to the electoral process. They eliminate the need for trust in a central authority and instead rely on a network of nodes that ensure data integrity and high security. This approach reduces the risk of fraud and cyberattacks while ensuring voter privacy through cryptographic methods.

One of the main advantages is resistance to attacks, as distributed data storage makes it impossible for a single node to compromise the system. Transparency in voting also increases voter confidence by allowing the verification of results without the need for external observers. Simultaneously, the disadvantages include technical complexity (perhaps the key drawback), energy-intensive processes, and limited scalability, which can pose

ISSN 1814-4225 (print)
ISSN 2663-2012 (online)

problems for large-scale elections [10].

Successful implementations include Voatz [11], FollowMyVote [12], and Horizon State [13]. In the United States, Voatz has been used for military voting, providing security and transparency [11]. FollowMyVote allows voters to verify their votes, and Horizon State ensures data security through a decentralized registry. These projects demonstrate the potential of decentralization to create a secure and fair electoral process.

An innovative online voting system leveraging cloud-based hybrid blockchain technology is introduced in [14]. This approach is designed to be reliable, flexible, transparent, secure, and cost-effective, and aims to eliminate the persistent flaws in current voting systems.

In [15], a decentralized voting system that leverages the Ethereum blockchain and smart contracts is developed to maintain the integrity of the voting process. The system is tested using simulated voting data to mirror real-world scenarios, with particular emphasis on security, scalability, and user-friendliness.

Despite these challenges, decentralized voting systems have significant potential to change democratic processes. Their implementation requires overcoming technical and infrastructural barriers; however, they have already demonstrated efficiency and reliability in real-world settings.

In recent years, numerous studies have focused on improving the security, scalability, and efficiency of blockchain-based electronic voting systems. Modern research mainly concentrates on three directions:

— optimization of consensus mechanisms;
— improvement of scalability of permissioned blockchain networks;
— increasing voter anonymity and data protection.

In a systematic literature review [16], Zimba et al. provided a comprehensive analysis of the state of the art in blockchain consensus, highlighting BFT as a critical mechanism for high-speed, low-energy networks. The review underscores that BFT mechanisms are prioritized for their deterministic safety and high-speed finality. However, designers must manage the trade-off between decentralization and energy, as many improved BFT variants achieve higher throughput by reducing the validator pool or using hierarchical structures. Ultimately, the study advocates the development of standardized energy metrics and modular architectures to better evaluate these improved BFT protocols in real-world scenarios.

The survey [17] classified state-of-the-art BFT algorithms into three major categories based on their primary design goals: more efficient, more robust, and more available BFT. Unlike broader surveys that examine entire blockchain platforms, this paper isolates the consensus mechanisms themselves, describing “replication consensus” (how transactions are committed) and “leadership consensus” (how leaders are elected or rotated). The authors conclude by highlighting critical gaps in BFT protocols, including scalability, fairness of ordering, post-quantum security, and interoperability.

The paper [18] provides a systematic analysis of more than 200 publications to evaluate how consensus mechanisms have evolved from Proof-of-Work toward more efficient alternatives. The authors establish a structured framework that categorizes consensus mechanisms into four primary groups based on their central design goals: high throughput, strong security, low energy, and flexible scaling. To compare these mechanisms, the review uses specific standard indicators: throughput, block finality, fault tolerance, energy consumption, and communication complexity. A central conclusion of this study is that no single consensus mechanism optimally satisfies all performance requirements; every design involves an explicit trade-off within the “blockchain trilemma” of decentralization, security, and scalability.

Li et al. [19] presented the results of an improved consensus algorithm, BR-PBFT, that integrates PBFT with a bubble-sort algorithm and a reputation mechanism. The architectural innovations proposed in Li et al. paper, such as reputation-based selection, are identified as critical pathways to reducing the computational and communication costs that otherwise hinder large-scale BFT deployments.

Nguyen and Costa [20] analyzed the practical use of BFT mechanisms in digital voting systems and identified a critical security-usability trade-off:

— BFT ensures the immutability and integrity of votes, preventing tampering once they are recorded on the distributed ledger;
— advanced BFT protocols often require complex cryptographic key management; if a system is made too secure through advanced cryptography, it may become too difficult for the average voter to use;
— the authors highlight platforms like Agora (tested in Sierra Leone) and Follow My Vote as examples of systems attempting to balance these BFT-driven security features with public accessibility.

However, existing studies mainly rely on classical PBFT-type consensus protocols that require leader election or multi-phase communication. This limitation motivates the development of leaderless consensus mechanisms that can improve distributed voting system efficiency and robustness.

Many studies have demonstrated that traditional PBFT-based approaches suffer from high communication complexity and scalability limitations. Therefore, many researchers have proposed optimized or

modified consensus protocols to reduce latency and improve throughput. However, the problem of achieving fast consensus in small private blockchain networks used for internal elections remains poorly studied.

This study focuses on improving the efficiency of consensus in such systems by introducing an automated leaderless approach.

1.3. Objectives and Approach

This study aims to improve the performance and fault tolerance of blockchain-based electronic voting systems by introducing the Automated Leaderless Byzantine Fault Tolerance (AL-BFT) consensus protocol.

The improvements are evaluated using the following metrics:

— transaction latency;
— throughput (requests per second);
— fault tolerance threshold;
— scalability.

The proposed solution's effectiveness is confirmed through experimental load testing and comparison with traditional BFT-based approaches.

The article presents an integrated approach to creating an electronic voting system that combines blockchain technology with the proposed Automated Leaderless Byzantine Fault Tolerance consensus protocol. The following are the main contributions of this paper:

1. A modified consensus mechanism (AL-BFT) that eliminates the leader election phase and reduces consensus delay.
2. A blockchain-based voting architecture optimized for small permissioned networks (500–1000 users).
3. Experimental evaluation of the system using load testing metrics such as response time, throughput, and system stability.
4. Demonstration that the proposed approach improves system efficiency compared to classical leader-based BFT approaches.

Structure of the article. Section 1 provides an overview of existing electronic voting systems. The main problems and shortcomings of current solutions are outlined, underscoring the need for new technologies to enhance security and transparency. The protection of voter data and the possibility of electoral interference are considered. Traditional electronic systems have several vulnerabilities that can be mitigated using blockchain.

Section 2 analyzes blockchain technology and its application to electoral systems. The basic principles of blockchain functioning and its capabilities for ensuring trust in electronic voting are highlighted. Additionally, the issue of decentralization and its role in enhancing the system's reliability is considered. This study analyzes how blockchain can guarantee the transparency of the voting process without the need for a central trusted party.

Section 3 focuses on developing an electronic voting model. A conceptual diagram of the system is presented, and the main components are identified along with their interactions to ensure the integrity and reliability of voting results. Data security issues at each stage of the electoral process are considered separately. Attention is also focused on the possibility of integrating additional protection mechanisms to enhance the system's stability.

Section 4 presents the practical implementation results of the system using the selected technologies. Key technical solutions that ensure the system's stable operation under real load conditions are described, and the results of system testing are presented.

Section 5 discusses the obtained results. Section 6 presents the conclusions.

2. Application of blockchain technology in electronic voting systems

This section presents the theoretical basis of the proposed method and describes the blockchain components used to develop the proposed electronic voting model.

Cryptographic methods ensure security in blockchain systems. Each block and transaction is protected by digital signatures that prevent tampering with the data. Additionally, consensus mechanisms ensure that new transactions are added to the blockchain only after they are verified by most network participants. This process prevents double-spending and other forms of fraud. Each block in the blockchain comprises three main components:

1. *data* is information recorded in a block, which can be transactions, contracts, or other forms of data depending on the purpose of the blockchain;
2. *a block hash* is a unique cryptographic signature of each block that is created based on its content, and it is worth emphasizing that it is used to identify the block and confirm its integrity;
3. *the hash of the previous block* is an element that links each new block to the previous one, creating an unbroken chain of blocks. Therefore, this structure ensures the security and immutability of the blockchain: any change to a previous block is propagated to all subsequent blocks.

The blockchain records transactions organized in a block sequence (Fig. 1). Consequently, it creates a digital history of events that cannot be falsified [21, 22].

ISSN 1814-4225 (print)
ISSN 2663-2012 (online)

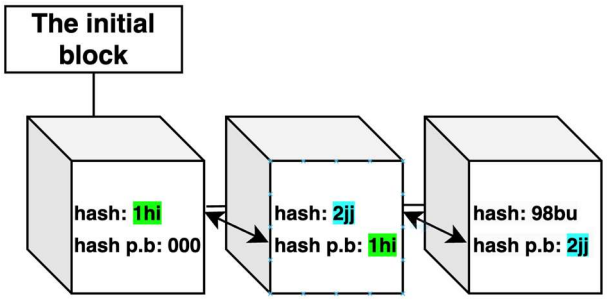

Fig. 1. The chain of blocks

The main components of the blockchain are also worth noting.

Nodes are the key elements of any blockchain system, performing the functions of storing, processing, and transmitting data. Computers, servers, or mobile devices that interact with each other via a communication protocol. There are different types of nodes; each has its own features and tasks. Full nodes store a complete copy of the blockchain, ensuring maximum security and decentralization. Lightweight nodes are optimized for devices with limited resources because they store only a subset of data and refer to full nodes for additional information. Mining nodes create new blocks through mining, and validation nodes confirm transactions in systems with alternative consensus mechanisms [23].

Nodes play an important role in blockchain-based e-voting systems. All votes are stored in a distributed ledger, enabling independent parties to verify the vote results. No single node can influence the data due to the blockchain's decentralized nature, ensuring its integrity and protection against tampering. Cryptographic methods ensure voter anonymity while allowing votes to be verified for authenticity. Lightweight nodes, in turn, make the system more accessible by allowing voters to cast their ballots on their own devices.

Nodes can participate in the voting process in an electronic voting system, ensuring the transparency and reliability of vote processing [24].

Another component is cryptography. It is the foundation of blockchain security, ensuring data protection, authentication, and transaction integrity. Asymmetric cryptography, which uses a pair of public and private keys, guarantees information confidentiality and reliability [25]. The public key creates a unique user address, and the private key signs transactions, confirming their authenticity.

Cryptography ensures anonymity: addresses hide user identities, and digital signatures protect data from tampering. In e-voting systems, ensure that votes are kept intact, authenticity is verified, and results remain transparent and reliable. Therefore, cryptography is a key element of blockchain security.

Consensus mechanisms ensure that participants agree on a single true version of the blockchain's data. The Byzantine Fault Tolerance (BFT) protocol guarantees consensus despite the presence of Byzantine (faulty or malicious) participants. In the context of electoral systems, BFT can be adapted to ensure fast and secure consensus among nodes that process votes. We propose the Automated Leaderless Byzantine Fault Tolerance consensus, which reduces the complexity of the algorithm and optimizes it for the specific requirements of the electoral process [14].

The advantages of using the AL-BFT are as follows:

1. Resistance to malicious actions and individual node errors;
2. The optimized algorithm allows for faster consensus, which is critical for timely vote counting.
3. Suitable for systems with a limited number of participants, which is typical for election commissions.

Restrictions:

1. Despite the simplifications, BFT implementation can still be technically challenging.
2. Requires certain computing resources to ensure efficient operation.
3. Electoral systems should ensure a high level of trust among participants, which may require additional authentication mechanisms.

3. Model of an electronic voting system using the AL-BFT

3.1. General system architecture

An electronic voting system consists of several key components that interact with each other via the Internet and the blockchain network. The architecture has a multilayered structure, including the following:

1. *a client layer*: the user interface through which voters and administrators interact with the system;
2. *a server layer*: an application server that processes client requests and interacts with the blockchain network;
3. *a blockchain layer*: a decentralized network of nodes that stores and validates voting transactions.

One of the system's main characteristics is its modular architecture. Modularity enables the system to be divided into separate functional components, each responsible for a specific task. This approach simplifies the development and subsequent support and further development of the system. If necessary, changing or updating a module will not affect the operation of other components. This is especially important for electronic voting systems, where any failure can jeopardize the process's integrity and reliability.

The main components of an electronic voting sys-

tem are users, a blockchain network, a consensus algorithm, an authentication service, a peer-to-peer network of nodes, and a user interface.

Users play a central role in the electoral process by performing functions related to interacting with the system. The user categories include voters, administrators, and candidates. Each of these groups interacts with the system through specialized web interfaces or mobile applications. Voters have unique credentials and private keys used to cryptographically authenticate and sign transactions (votes). Administrators manage the electoral process, ensuring that the system operates smoothly and complies with procedural requirements. Candidates can apply to participate in the election and view the results. This process can be formalized as a precedent diagram (Fig. 2).

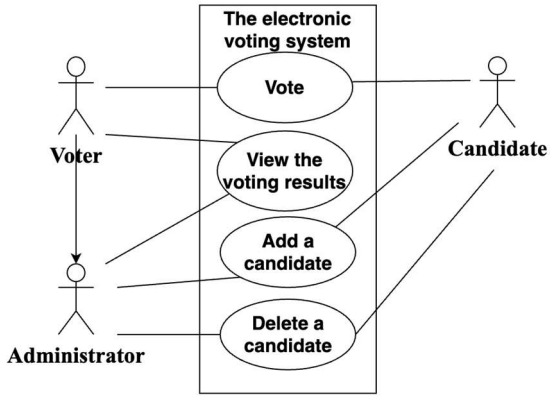

Fig. 2. The interaction precedents between voters, administrators, and candidates

The blockchain network, which serves as the main repository of electoral data and ensures its integrity and security, is a key element of the system. A blockchain is a decentralized database structured as a chain of blocks, each containing transaction information. The hash of the previous block is incorporated into each new block, forming an unchanging chain that makes changing data without detection impossible. The system guarantees the confidentiality and immutability of information thanks to cryptographic methods.

The AL-BFT algorithm is used to reach consensus among network nodes. This algorithm ensures the system's stability and reliability even in the presence of unscrupulous or faulty nodes. Such a mechanism is critical for decentralized systems that are susceptible to failures or malicious actions. It ensures decision consistency by allowing only reliable data to be added to the blockchain.

Another important component of the system that provides secure access to its functions is the authentication service. For this purpose, public-private key pairs are used to ensure each voter's identification is unique. During voting, the private key is used to verify the voter's identity, thereby eliminating the possibility of fraud or vote tampering.

The system is supported by a peer-to-peer network of nodes, each an independent entity that interacts with other nodes. This architecture eliminates the need for centralized management, increasing the system's resilience to attacks or failures of individual nodes. Each node stores a full copy of the blockchain and validates new blocks, ensuring high fault tolerance and system reliability (Fig. 3).

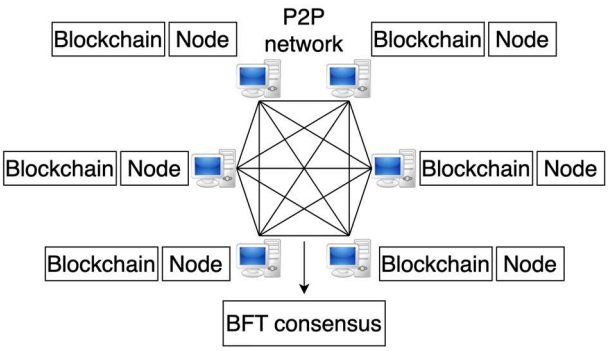

Fig. 3. The architecture of the electronic voting system

3.1.1. Interaction of system components

The first step in the voting process is voter authentication. When using the system, a voter must authenticate their identity by entering the login and password provided during registration. In the future, this can be achieved through various methods, such as digital signatures, biometric data, or multi-factor authentication (e.g., SMS or additional codes). Authentication is necessary to ensure that the voter is eligible and that no unauthorized user can access the system.

After successful authentication, the voter can choose among the candidates presented in the system. The voter can view the list of candidates and make a choice by clicking on the appropriate button or interface element.

Once the voter has made their choice, the system generates an encrypted transaction representing it. This transaction is signed with the voter's private key to verify the vote's authenticity. Cryptographic signatures ensure that only the voter can alter or falsify their vote.

The signed voting transaction is sent to a blockchain node for validation. The node receiving the transaction verifies the voter's signature and ensures that the vote is valid and that the voter has not already voted. Once the transaction has been verified, it is included in a new block created by the receiving node. This block is transmitted to other nodes in the network for validation along with other new transactions.

This process is shown in Fig. 4.

All nodes that receive the block are validated. In the

ISSN 1814-4225 (print)
ISSN 2663-2012 (online)

BFT algorithm, nodes must reach consensus that the block is correct and contains no malicious or incorrect transactions. BFT ensures that the network can reach the correct decision even if some nodes behave maliciously or fail. Once enough nodes confirm the block's validity, it is added to the blockchain. Election transactions are now considered officially registered, and the voter's choice cannot be changed or deleted. The blockchain network consists of many nodes that constantly synchronize their copies. Each node keeps a complete history of the blockchain, and all nodes regularly exchange information about their own blockchain states to ensure consistency. If one node detects that another node has a longer or more valid chain of blocks, it will download the new blocks to update its copy.

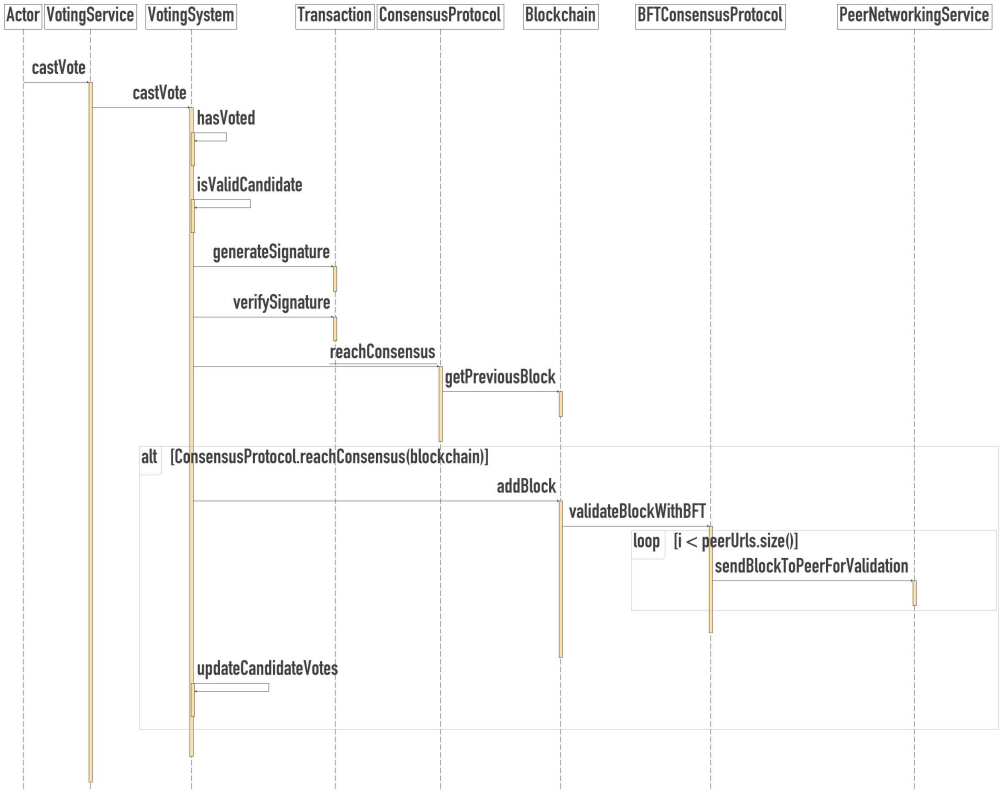

Fig. 4. Diagram of the sequence of creating a voter's vote [24]

3.2. The blockchain model and the use of a consensus mechanism

Blocks in the blockchain are organized in chronological order, forming a chain in which each block depends on the previous one. The connection between blocks is ensured by hashes, which guarantee data immutability: any attempt to modify it results in changes to the entire chain, making such interventions obvious.

The key components of a block are an index that determines its place in the chain, a hash of the previous block to ensure connectivity, transactions that contain voting data, and digital signatures for authentication. The timestamp allows you to track the chronology of events, which is important for voting transparency.

The process of creating a block is shown in Fig. 5.

Voting transactions include a voter ID, candidate selection, and a digital signature that protects the data from tampering. Hashing guarantees the integrity of information by creating a unique digital fingerprint for each block.

The genesis block, the initial element of the chain that contains the basic system settings, is of particular importance. The manual setting ensures the stability and functionality of the entire blockchain network.

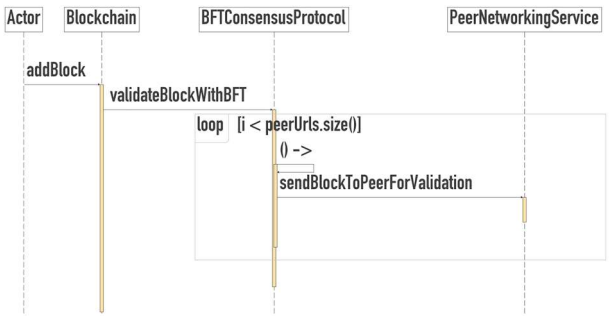

Fig. 5. Diagram of the block creation sequence

The proposed electronic voting model uses the AL-BFT consensus mechanism. The BFT algorithm ensures

consensus among nodes despite malicious or faulty participants. This is extremely important for voting systems, as it protects the process from sabotage by individual nodes or third-party attacks. The AL-BFT mechanism works as follows:

1. Each node in the network has a copy of the blockchain and participates in the validation process of new blocks.

2. When a new block is created, it is sent for verification to all nodes on the network. Each node independently validates the block by checking its hash and transactions.

3. After the block, each node votes for or against adding it to the chain.

4. The block is added to the blockchain only if more than 2/3 of the nodes agree on its validity. This ensures that the system can operate even in the presence of multiple malicious or faulty nodes.

The advantages of this AL-BFT implementation for a voting system include high resistance to malicious actions, independence from central control (i.e., consensus is achieved through node voting rather than centralized verification), data security, and fast validation.

The key feature of AL-BFT implementation in the voting system is automated consensus building without a leader to vote or a preliminary (pre) phase. This enables faster and more continuous agreement among nodes, thereby increasing network efficiency. This implementation minimizes the risk of delay in leader selection. This makes the validation process more reliable and faster, which is an important factor in ensuring the smooth functioning of the electronic voting system.

The security threshold of the AL-BFT consensus is defined by the classical BFT requirement $f < n/3$, where n is the total number of nodes and f is the number of Byzantine (faulty) nodes.

Because n nodes participate and each node sends a message to all other nodes, the total number of messages exchanged in the voting phase is proportional to n^2. Therefore, the communication complexity per decision (block) is typically $O(n^2)$.

4. A software implementation of the model and the results

4.1. A rationale for choosing technologies and tools

The development of an electronic voting system based on blockchain technology requires careful selection of technologies and tools that will ensure security, scalability, efficiency, and usability.

The Java programming language [26] was chosen for server-side development for the following reasons:

1. Java enables the creation of applications that can run on different operating systems without changing the code.

2. Thanks to JIT compilation and optimization, Java provides high application performance.

3. Several libraries and frameworks that simplify the development of complex systems;

4. Built-in security and memory management mechanisms reduce the vulnerability risk.

The Spring Boot framework [27, 28] was chosen to create RESTful [27] web services because of the following:

1. Auto-configuration and built-in server allow you to quickly launch the application;

2. Support for dependencies and modules simplifies scaling and code maintenance;

3. Easy integration with other security Spring modules, such as Spring Security [28].

Blockchain is used as the basis of the voting system due to its decentralized nature, which ensures independence and resistance to censorship. As mentioned earlier, records on the blockchain cannot be changed without the network's consent, ensuring the integrity of votes. Additionally, all transactions can be verified, which increases system trust.

The AL-BFT consensus is characterized by its resistance to malicious actions, ensuring that the system can function correctly even in the presence of malicious or faulty nodes. Additionally, it ensures compliance with voting requirements, providing a high level of trust and accuracy in distributed systems.

RSA [29] and SHA-256 [30] cryptographic algorithms are used to ensure security and authenticity. Asymmetric encryption for generating key pairs, signing transactions, and verifying signatures, and data hashing to create unique block and transaction identifiers, ensuring information immutability.

The Vue.js framework [31] was chosen for the client-side development because of the following:

1. Reactivity: Provides a fast and dynamic user experience;

2. Ease of learning: low entry threshold and clear component structure;

3. Flexibility: It easily integrates with other libraries and projects.

One of the most effective and widespread approaches is to use a RESTful API (Application Programming Interface) based on the REST (Representational State Transfer) architecture. This approach not only facilitates development but also opens up opportunities for scaling and integration in complex and distributed environments.

One of the key advantages of using the RESTful API is its standardization. By using the HTTP protocol for communication between the client and server, the

ISSN 1814-4225 (print)
ISSN 2663-2012 (online)

RESTful architecture adheres to well-established developer principles. All data operations are performed using standard HTTP methods, such as GET, POST, PUT, and DELETE, which greatly simplify the development and interaction among the system components. By adhering strictly to these standards, developers can easily integrate their applications with other services or systems using RESTful architecture.

Another important advantage of RESTful architecture is its scalability. When modern systems can process several simultaneous requests from thousands or even millions of users, an architectural approach that easily distributes load across servers or services is necessary. The RESTful API supports this distributed architecture by providing an efficient mechanism for horizontal scaling. This is especially true in microservice architectures, where each service can operate independently, providing flexibility and fault tolerance to the entire system.

Other tools and technologies were used:

1. Spring Security: implements user authentication and authorization mechanisms;
2. Lombok: reduces the amount of template code, thereby increasing readability and ease of support;
3. Axios: for making HTTP requests from the client side;
4. Vue Router and Vuex: Manage routing and application health.

4.2. Description of the P2P network for synchronizing blockchains between different nodes

A peer-to-peer (P2P) network is a key component of any decentralized blockchain-based system. A P2P network in a blockchain-based e-voting system enables nodes (system participants) to synchronize by exchanging information about new blocks, transactions, and the blockchain's status. This ensures data consistency between nodes and maintains consensus.

The basic idea behind a P2P network is that each node in the system acts as both a client and a server. This allows nodes to avoid relying on a centralized server to receive or exchange data. In our system, nodes constantly synchronize their blockchain copies to ensure that all network participants have the same, consistent blockchain history.

Block exchange and transactions between network nodes. The system uses HTTP requests between nodes to implement synchronization (Fig. 6), which is handled by the PeerNetworkingService. This service sends and receives requests to other nodes to obtain the current blockchain version and transfers new blocks for validation.

```
@Scheduled(fixedDelay = 15000) // 15000 15 seconds
public void synchronizeBlockchain() {
    List<String> peerUrls = Arrays.asList(
            "http://localhost:8080",
            "http://localhost:8081",
            "http://localhost:8082");
    ExecutorService executorService =
            Executors.newFixedThreadPool(peerUrls.size());
    for (String peerUrl : peerUrls) {
        executorService.submit(() -> {
            try {
                CompletableFuture<Blockchain> blockchainFuture =
                        peerNetworkingService.getBlockchainFromPeerAsync(peerUrl);
                Blockchain peerBlockchain = blockchainFuture.get();
                if (peerBlockchain != null && peerBlockchain.isValid() &&
                        peerBlockchain.getChain().size() > votingSystem.getChain().size()) {
                    peerService.setBlockchain(peerBlockchain);
                    votingSystem.setBlockchain(peerBlockchain);
```

Fig. 6. A snippet of node synchronization

Each node in the network can request and verify its neighbors' blockchains.

Synchronization is based on the following algorithm:

1. Regular updates: Each node sends requests to other nodes every 15 s to obtain their blockchains. If another node has a longer or more up-to-date chain, the node is synchronized with the local blockchain.

2. Chain comparison: After receiving a chain of blocks from another node, the local node checks the received chain's validity and compares it with its own. If the received chain is longer and passes all the checks, the local chain is replaced.

3. Sharing new blocks: If a node creates a new block, it can send it to its neighbors for validation. The neighbors check a new block for compliance with the consensus criteria (in our case, AL-BFT) and add it to their chain if it is successful.

Fig. 7 shows the implementation of sending a block to a P2P network.

```
public boolean sendBlockToPeerForValidation(String peerUrl, Block block) {
    try {
        String blockJson = objectMapper.writeValueAsString(block);
        HttpHeaders headers = new HttpHeaders();
        headers.setContentType(MediaType.APPLICATION_JSON);
        HttpEntity<String> request = new HttpEntity<>(blockJson, headers);
        ResponseEntity<Boolean> response =
                restTemplate.postForEntity(
                        url: peerUrl + "/api/vote/validateBlock", request, Boolean.class);
        return response.getBody() != null && response.getBody();
    } catch (JsonProcessingException e) {
        e.printStackTrace();
        return false;
    } catch (RestClientException e) {
        e.printStackTrace();
        return false;
    }
}
```

Fig. 7. Method of sending a block to P2P for validation

4.3. Test results

Load testing was performed using JMater, a widely used tool for evaluating the performance of web applications, APIs, databases, and other systems. The results are presented in Table 1. Based on this, we can draw the following conclusions:

1. 800 test iterations were conducted.
2. Requests are processed in 4.12 s on average;
3. The fastest request was processed in 0 s, the slowest was processed in 26343 s, and such a large time difference indicates insignificant fluctuations in system performance;
4. The high standard deviation value confirms that the test results have a small dispersion, i.e., the response time differs slightly from one request to another;
5. The average throughput is 29.1 requests per s;
6. On average, 4.86 KB of data is transferred per request.

In general, the system demonstrates the ability to withstand the load. Despite significant differences in processing time across request types, it efficiently handles tasks and can handle a given number of requests. This indicates the system's reliability and endurance under load. Note the stable operation of GET requests, which showed relatively consistent response times. This indicates efficient processing of simpler requests and the stability of certain system components, which positively impact its overall performance.

Additionally, the bar chart (Fig. 8) effectively illustrates the differences in the average response times across the request types.

The load testing confirmed the reliability and ability of the AL-BFT system to provide transaction finality. However, given the scope of the current study, the system was designed and tested to operate in small networks (permissioned blockchain) serving 500–1000 simultaneous voters. These limitations apply to the current software implementation and test environment. Further performance increases and system scaling to accommodate more extensive elections are possible by optimizing network settings, which are beyond the scope of this study.

Table 1

Load testing

Label	GET /api/vote/results	GET /api/vote/results	POST /api/vote/add-candidate	POST /api/vote	TOTAL
# Samples	200	200	200	200	800
Average	11	1	2	16465	4120
Min	1	0	0	532	0
Max	130	53	22	26343	26343
Std. Dev.	20,66	4,23	3,18	5526,73	7644,34
Error %	0.00%	0.00%	0.00%	0.00%	0.00%
Throughput	39.8/sec	39.8/sec	39.9/sec	7.3/sec	29.1/sec
Received KB/sec	11,31	35,16	10,92	1,96	12,44
Sent KB/sec	5,13	5,41	7,95	1,48	4,86
Avg. Bytes	291	903,5	280	276	437,6

ISSN 1814-4225 (print)
ISSN 2663-2012 (online)

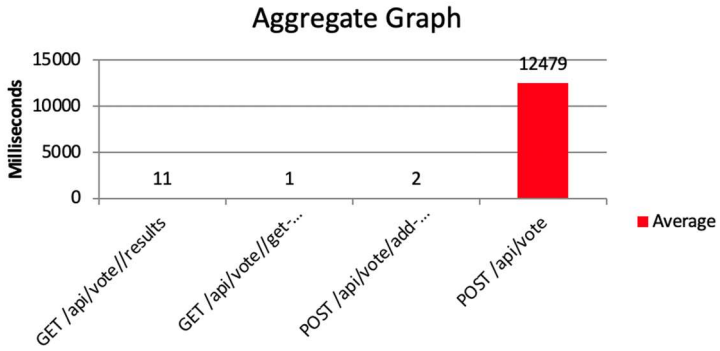

Fig. 8. A bar chart of the average request response time

5. Discussion

The development of a blockchain-based electronic voting model represents a modern approach to ensuring the security, transparency, and reliability of the electoral process. Key requirements for the system were identified during the work, which are critical for protecting voters' rights and ensuring trust in the voting results. The system must meet the highest standards, ensuring voter authentication, single-vote voting, choice privacy, vote validity, results transparency, and the possibility of distributed counting. Non-functional requirements focused on security, resistance to technical failures, scalability, accessibility, and user-friendliness, which make the system reliable and convenient for various voters. The system architecture is built on three layers: client, server, and blockchain layers, which ensure the system's modularity, scalability, and reliability. The modular architecture enables system improvements without compromising its basic functions and ensures resistance to high loads during mass elections. Cryptographic methods, such as asymmetric encryption and hashing, play a special role in ensuring security and anonymity. Additionally, the Byzantine Fault Tolerance algorithm used in the consensus process between nodes ensures that the system is resistant to malicious actions and technical failures.

The key element of the developed model is the blockchain, which guarantees voting data immutability, protects voter transactions, and ensures voting process transparency. The structure of blocks, including indexes, hashes, transactions, and timestamps, creates a reliable mechanism for storing information and preventing manipulation. The BFT consensus mechanism is crucial in ensuring data consistency across nodes and protecting the system against sabotage or failures of individual elements.

The practical implementation of a blockchain-based electronic voting system was thoroughly examined. The first part of the section focuses on justifying the choice of technologies and tools used to implement the system. The Java programming language, the Spring Boot framework for the server-side, and Vue.js for the client-side were chosen as the optimal solutions to ensure the system's security, scalability, and efficiency.

The primary arguments in favor of using blockchain are the elimination of centralized control and the guarantee of data immutability, both of which are important for a voting system. The AL-BFT protocol was proposed to ensure resistance to malicious nodes and to guarantee the correct functioning of the system even in the presence of problems with some network participants. Cryptographic mechanisms, such as RSA and SHA-256, were also considered to ensure transaction security and data integrity.

A detailed description of the project setup is presented, along with an explanation of the interaction between the client and server sides via the REST API. The importance of a simplified approach to development is emphasized here, particularly through the use of a microservices architecture, which enhances scalability and efficiency when processing large numbers of requests.

The focus was on the system's software components: the blockchain architecture, the transaction management module, and mechanisms for node synchronization. The use of the AL-BFT protocol for consensus and ensuring that all network participants agree on the addition of new blocks is of particular importance. This mechanism is critical for maintaining the reliability and integrity of the voting results.

Development and synchronization between system nodes via a Peer-to-Peer network were also covered. Algorithms for exchanging blocks and transactions between network nodes that ensure data consistency across different network participants are described. Special attention was paid to the regular updating of the blockchain by each node to maintain the current version.

The system demonstrates resilience under load, maintaining functionality even with significant variations in the processing times of different requests and handling a substantial volume of requests. The current throughput of the system meets existing needs, indicating significant

potential for further scaling.

The *scientific novelty* of this study lies in the development of an automated leaderless BFT consensus mechanism specifically tailored for electronic voting systems operating in small permissioned blockchain networks. Unlike traditional PBFT-based approaches, the proposed method eliminates the leader election phase and reduces consensus delay, which improves the voting system's overall performance. Table 2 presents a more detailed comparison of traditional BFT (PBFT) and AL-BFT with respect to key characteristics.

Practical implementation of the system based on the selected technologies is carried out. Setting up the project, implementing software components, and ensuring blockchain synchronization between network nodes are considered. Another important aspect is ensuring secure data exchange between nodes and preventing possible attacks. This paper describes the key technical solutions that enable the system to operate stably under real-world load conditions.

Table 2

Traditional BFT vs AL-BFT

Parameter	Traditional BFT (PBFT)	AL-BFT
Leader	Yes	No
Transaction latency	High (multiple communication phases and leader coordination)	Reduced (no leader election, optimized communication)
Throughput	Moderate	Higher due to reduced coordination overhead
Scalability	Limited	Improved for small networks (up to 1000 voters)
Fault tolerance threshold	Up to $(n-1)/3$	Up to $(n-1)/3$
Suitability for university elections	Limited	Suitable

This study demonstrates the practical possibility of using innovative technologies in electoral processes to ensure voters' reliability, confidentiality, and trust. The examples and implemented technical solutions can serve as a basis for further research in this area.

6. Conclusions

This study proposes a blockchain-based electronic voting system using the AL-BFT consensus mechanism. The system's conceptual scheme is presented, along with the main components and their interactions, to ensure the integrity and reliability of voting results. The developed model improves system efficiency by eliminating the leader election phase and reducing consensus delay.

The experimental load testing confirmed the stability and reliability of the proposed system. The proposed system is particularly suitable for small permissioned networks, such as university elections. Future research will focus on improving scalability and integrating advanced cryptographic protection mechanisms.

The proposed e-voting system can benefit university internal elections and polls by enabling fair, reliable voting with modern technology, thereby significantly reducing the risk of unauthorized tampering or vote manipulation.

One possible area for future improvement is the introduction of advanced Zero-Knowledge Proofs (ZKP) methods, such as zk-SNARKs or zk-STARKs.

Contributions of authors: conceptualization – **Mariia Rodinko, Maksym Holikov**; methodology – **Maksym Holikov, Mariia Rodinko**; formulation of tasks – **Mariia Rodinko, Maksym Holikov**; analysis – **Maksym Holikov, Mariia Rodinko, Dmytro Uzlov, Artem Bronnikov**; development of model – **Maksym Holikov, Mariia Rodinko**; software – **Maksym Holikov**; verification – **Mariia Rodinko, Artem Bronnikov**; analysis of results – **Mariia Rodinko, Dmytro Uzlov**; visualization – **Maksym Holikov**; writing – original draft preparation – **Maksym Holikov, Mariia Rodinko, Artem Bronnikov**; writing – review and editing – **Dmytro Uzlov**.

All the authors have read and agreed to the published version of this manuscript.

Conflict of Interest

The authors declare that they have no conflict of interest in relation to this research, whether financial, personal, authorship, or otherwise, that could affect the research and its results presented in this paper.

Financing

This study was conducted within the framework of the research project № 3-41-26 with partial financial support from the Ministry of Education and Science of Ukraine.

Data Availability

The work has associated data in the data repository.

Use of Artificial Intelligence

The authors have used artificial intelligence technologies within acceptable limits to provide their own verified data, as described in the research methodology section.

References

1. Aidynov, T., Goranin, N., Satybaldina, D., & Nurusheva, A. A systematic literature review of current trends in electronic voting system protection using modern cryptography. *Applied Sciences*, 2024, vol. 14, no. 7, article no. 2742. DOI: 10.3390/app14072742.

2. Almeida, R. L., Baiardi, F., Maesa, D. D. F., & Ricci, L. Impact of Decentralization on Electronic Voting Systems: A Systematic Literature Survey. *IEEE Access*, 2023, vol. 11, pp. 132389–132423. DOI: 10.1109/ACCESS.2023.3336593.

3. Ohize, H. O., Onumanyi, A. J., Umar, B. U., Ajao, L. A., Isah, R. O., Dogo, E. M., & Ibrahim, M. M. Blockchain for securing electronic voting systems: a survey of architectures, trends, solutions, and challenges. *Cluster Computing*, 2025, vol. 28, no. 2, article no. 132. DOI: 10.1007/s10586-024-04709-8.

4. El Kafhali, S. Blockchain-Based Electronic Voting System: Significance and Requirements. *Mathematical Problems in Engineering*, 2024, vol. 2024, article no. 5591147. DOI: 10.1155/2024/5591147.

5. Balogh, S., Gallo, O., Ploszek, R., Špaček, P., & Zajac, P. IoT security challenges: Cloud and blockchain postquantum cryptography and evolutionary techniques. *Electronics*, 2021, vol. 10, no. 21, article no. 2647. DOI: 10.3390/electronics10212647.

6. Zhang, B., Oliynykov, R., & Balogun, H. A Treasury System for Cryptocurrencies: Enabling Better Collaborative Intelligence. *Proceedings of Network and Distributed System Security (NDSS) Symposium 2019*, San Diego, CA, USA, 2019, 15 p. DOI: 10.14722/ndss.2019.23024.

7. Mannonov, K. M. U., & Myeong, S. Citizens' perception of blockchain-based e-voting systems: Focusing on TAM. *Sustainability*, 2024, vol. 16, no. 11, article no. 4387. DOI: 10.3390/su16114387.

8. Valimised. Introduction to i-voting. *Estonian National Electoral Committee*, 2023. Available at: https://www.valimised.ee/en/internet-voting/more-about-i-voting/introduction-i-voting (accessed 24 April 2025).

9. Lever, K. E. Single Points of Failure Within Systems-of-Systems. *Proceedings of the 14th Annual Post Graduate Symposium on the Convergence of Telecommunications, Networking and Broadcasting (PGNet)*, Liverpool, UK, 2013, pp. 183–188. Available at: https://www.researchgate.net/publication/268684111_Single_Points_of_Failure_Within_Systems-of-Systems (accessed 24 April 2025).

10. Isirova, K., Kiian, A., Rodinko, M., & Kuznetsov, A. Decentralized Electronic Voting System Based on Blockchain Technology Developing Principals. *CEUR Workshop Proceedings of The Third International Workshop on Computer Modeling and Intelligent Systems*, 2020, vol. 2608, pp. 211–223. DOI: 10.32782/CMIS/2608-17.

11. Specter, M. A., Koppel, J., & Weitzner, D. The ballot is busted before the blockchain: a security analysis of voatz, the first internet voting application used in U.S. federal elections. *Proceedings of the 29th USENIX Security Symposium*, USENIX Association, USA, 2020, pp. 1535–1552. DOI: 10.5555/3489212.3489299.

12. Follow My Vote. End-to-End Verifiable Blockchain Voting Software. 2023. Available at: https://followmyvote.com (accessed 24 April 2025).

13. Horizon State. Elections & Results Platform. 2023. Available at: https://horizonstate.com/horizon-state-solutions/horizon-state-elections/ (accessed 24 April 2025).

14. Jayakumari, B., Sheeba, S. L., Eapen, M., Anbarasi, J., Ravi, V., Suganya, A., & Jawahar, M. E-voting system using cloud-based hybrid blockchain technology. *Journal of Safety Science and Resilience*, 2024, vol. 5, no. 1, pp. 102–109. DOI: 10.1016/j.jnlssr.2024.01.002.

15. Singh, I., Kaur, A., Agarwal, P., & Idrees, S. M. Enhancing security and transparency in online voting through blockchain decentralization. *SN Computer Science*, 2024, vol. 5, article no. 921. DOI: 10.1007/s42979-024-03286-2.

16. Zimba, A., Phiri, K. O., Mulenga, M., & Mukupa, G. A systematic literature review of blockchain technology and energy efficiency based on consensus mechanisms, architectural innovations, and sustainable solutions. *Discover Analytics*, 2025, vol. 3, no. 1, article no. 14. DOI: 10.1007/s44257-025-00041-6.

17. Zhang, G., et al. Reaching consensus in the byzantine empire: A comprehensive review of bft consensus algorithms. *ACM Computing Surveys*, 2024, vol. 56, no. 5, pp. 1–41. DOI: 10.1145/3636553.

18. Shen, Z., Qu, Q., & Chen, X.-B. Blockchain Consensus Mechanisms: A Comprehensive Review and Performance Analysis Framework. *Electronics*, 2025, vol. 14, article no. 3567. DOI: 10.3390/electronics14173567.

19. Li, F., Kou, Y., Wang, G., & Xue, X. An effective consensus based on PBFT and reputation mechanism for service ecosystems. *International Journal of Web Information Systems*, 2025, vol. 21, no. 6, pp. 670–691. DOI: 10.1108/IJWIS-06-2025-0152.

20. Nguyen, C., & Costa, A. Blockchain-Based Digital Voting Systems: Security and Usability Analysis. *ITSI Transactions on Electrical and Electronics Engineering*, 2025, vol. 12, no. 1, pp. 1–6. DOI: 10.65521/itsi-teee.v12i1.141.

21. Zheng, Z., Xie, S., Dai, H.-N., Chen, X., & Wang, H. Blockchain challenges and opportunities: a survey. *International Journal of Web and Grid Services*, 2018, vol. 14, no. 4, pp. 352–375. DOI: 10.1504/IJWGS.2018.095647.

22. Dorfleitner, G., Muck, F., & Scheckenbach, I. Blockchain applications for climate protection: A global empirical investigation. *Renewable and Sustainable Energy Reviews*, 2021, vol. 149, article no. 111378. DOI: 10.1016/j.rser.2021.111378.

23. Elrom, E. The Blockchain Developer: A Practical Guide for Designing, Implementing, Publishing, Testing, and Securing Distributed Blockchain-based Projects. Berkeley, CA: Apress, 2019, 527 p. DOI: 10.1007/978-1-4842-4847-8.

24. Kushch, S., & Prieto-Castrillo, F. Blockchain for Dynamic Nodes in a Smart City. *Proceedings of 5th IEEE World Forum on Internet of Things (WF-IoT'19)*, Limerick, Ireland, 2019, pp. 29–34. DOI: 10.1109/WF-IoT.2019.8767336.

25. Zhang, R., Xue, R., & Liu, L. Security and Privacy on Blockchain. *ACM Computing Surveys*, 2019, vol. 52, no. 3, article no. 51, pp. 1–34. DOI: 10.1145/3316481.

26. Eckel, B. Thinking in Java. 4th ed. Upper Saddle River, NJ: Prentice Hall, 2006, 1057 p.

27. Spring Technologies. Spring Framework Documentation. 2024. Available at: https://spring.io/ (accessed 24 April 2025).

28. Baeldung. Spring Tutorial Series. 2024. Available at: https://www.baeldung.com/spring-tutorial (accessed 24 April 2025).

29. Tiwari, P., Choudhary, V., & Aman, S. Analysis and Comparison of DES, AES, RSA Encryption Algorithms. *Proceedings of 4th International Conference on Advances in Computing, Communication Control and Networking (ICAC3N)*, Greater Noida, India, 2022, pp. 1913–1918. DOI: 10.1109/ICAC3N56670.2022.10073996.

30. National Institute of Standards and Technology (NIST). Secure Hash Standard (SHS). *FIPS PUB 180-4*, 2015. DOI: 10.6028/NIST.FIPS.180-4.

31. Vue.js Team. Vue.js: The Progressive JavaScript Framework. 2024. Available at: https://vuejs.org (accessed 24 April 2025).

Received 07.05.2025, Received in revised form 22.01.2026
Accepted date 13.07.2026, Published date 31.07.2026

СИСТЕМА ЕЛЕКТРОННОГО ГОЛОСУВАННЯ НА ОСНОВІ БЛОКЧЕЙНУ З АВТОМАТИЗОВАНИМ КОНСЕНСУСОМ БЕЗ ЛІДЕРА AL-BFT

М. С. Голіков, М. Ю. Родінко, Д. Ю. Узлов, А. І. Бронніков

Предметом *вивчення в статті є децентралізована система електронного голосування на основі технології блокчейну.* ***Метою*** *дослідження є підвищення продуктивності та відмовостійкості систем електронного голосування на основі блокчейну шляхом впровадження протоколу консенсусу AL-BFT (Automated Leaderless Byzantine Fault Tolerance).* ***Завдання*** *полягало у розробці та оцінці моделі системи електронного голосування, яка застосовує запропонований механізм консенсусу AL-BFT у дозволеній одноранговій мережі. Використані* ***методи*** *включали комп'ютерне моделювання одноранговоі (P2P) мережі, реалізацію децентралізованого реєстру та експериментальне навантажувальне тестування протоколу консенсусу. Продуктивність системи оцінюється за допомогою ключових показників, таких як затримка транзакцій, пропускна здатність (запити за секунду), поріг відмовостійкості та масштабованість.* ***Результати*** *дослідження включають розробку концептуальної архітектури системи електронного голосування, визначення її основних компонентів та аналіз їхньої взаємодії для забезпечення цілісності даних і надійності результатів голосування. Безпека даних розглядається на кожному етапі виборчого процесу, а також аналізуються додаткові механізми захисту для підвищення стійкості системи. Ключовою особливістю запропонованого підходу є усунення фази обрання лідера в процесі консенсусу, що зменшує накладні витрати на координацію та забезпечує ефективнішу згоду між вузлами. Експериментальна оцінка демонструє, що запропонований протокол AL-BFT зменшує затримку транзакцій та покращує пропускну здатність, зберігаючи при цьому рівень відмовостійкості традиційних підходів, заснованих на візантійській відмовостійкості. Результати підтверджують підвищену ефективність порівняно з класичними механізмами консенсусу на основі лідерів, особливо в невеликих блокчейн-мережах з дозволом.* ***Висновки.*** *Практична реалізація системи була розроблена та протестована в умовах моделювання реального навантаження. Запропоноване рішення забезпечує стабільну роботу системи та надійне формування консенсусу. Систему можна ефективно застосовувати до виборів в університетах, організаційного голосування та інших сценаріїв, що вимагають прозорості, безпеки та стійкості до маніпуляцій.*

Ключові слова*: система електронного голосування, блокчейн, протокол консенсусу, протокол BFT.*

Голіков Максим Сергійович – аспірант кафедри комп'ютерних систем та робототехніки, Навчально-науковий інститут комп'ютерних наук та штучного інтелекту, Харківський національний університет імені В. Н. Каразіна, Харків, Україна.

Родінко Марія Юріївна – доктор філософії, доцент кафедри інтелектуальних програмних систем і технологій, Навчально-науковий інститут комп'ютерних наук та штучного інтелекту, Харківський національний університет імені В. Н. Каразіна, Харків, Україна.

Узлов Дмитро Юрійович – кандидат технічних наук, доцент, директор Навчально-наукового інституту комп'ютерних наук та штучного інтелекту, Харківський національний університет імені В. Н. Каразіна, Харків, Україна.

Бронніков Артем Ігорович – кандидат технічних наук, доцент кафедри комп'ютерно-інтегрованих технологій, автоматизації та робототехніки, Харківський національний університет радіоелектроніки, Харків, Україна.

Maksym Holikov – Postgraduate Student at the Department of Computer Systems and Robotics, Education and Research Institute of Computer Sciences and Artificial Intelligence, V. N. Karazin Kharkiv National University, Kharkiv, Ukraine,
e-mail: maksym.holikov@karazin.ua, ORCID: 0000-0003-4842-7823, Scopus Author ID: 59152954100.

Mariia Rodinko – Doctor of Philosophy, Associate Professor at the Department of Intelligent Software Systems and Technologies, Education and Research Institute of Computer Sciences and Artificial Intelligence, V. N. Karazin Kharkiv National University, Kharkiv, Ukraine,
e-mail: mariia.rodinko@karazin.ua, ORCID: 0000-0003-4692-9811, Scopus Author ID: 57194032767.

Dmytro Uzlov – Candidate of Technical Sciences, Associate Professor, Director of the Education and Research Institute of Computer Sciences and Artificial Intelligence, V. N. Karazin Kharkiv National University, Kharkiv, Ukraine,
e-mail: dmytro.uzlov@karazin.ua, ORCID: 0000-0003-3308-424X, Scopus Author ID: 57201780269.

Artem Bronnikov – Candidate of Technical Sciences, Associate Professor at the Department of Computer-Integrated Technologies, Automation and Robotics, Kharkiv National University of Radio Electronics, Kharkiv, Ukraine,
e-mail: artem.bronnikov@nure.ua, ORCID: 0000-0003-3096-7653, Scopus Author ID: 58375766100.