<u>Blockchain-Powered E-Voting System</u>

Abstract:

In recent years, there has been a growing interest in using electronic voting systems to replace traditional paper-based voting systems due to the advantages they offer, such as speed, efficiency, and reduced risk of errors and fraud. However, existing e-voting systems have several limitations, such as manual voting and mining processes, lack of anonymity in voter data, and security concerns.

In this paper, we present the design and implementation of our modified e-voting system, which offers a reliable and secure solution for electronic voting that ensures transparency, security, anonymity, and ease-of-use. The system is designed to be transparent and secure, with blockchain integration providing an extra layer of security and transparency. The automated voting and mining processes eliminate the need for manual intervention and make the system faster and more efficient.

We evaluate the performance of our modified e-voting system by comparing it with the previously designed e-voting system. Our system offers significant improvements over traditional e-voting systems, with enhanced security, transparency, and user-friendliness. Our modified e-voting system can be used for various applications, including voting in political elections, shareholder meetings, and boardroom decisions.

Introduction:

Blockchains are sophisticated systems that operate on the principles of decentralization, distribution, and immutability. They serve as secure and transparent ledgers, capable of storing data in either permissioned or permissionless environments. At their core, blockchains enable a collective of users to collaboratively maintain a shared ledger within a particular community. One key characteristic is that, during regular blockchain network operations, transactions become unalterable once they have been incorporated into a block[2].

The potential benefits of electronic voting systems are numerous, including the ability to streamline the voting process, increase voter turnout, and reduce the risk of errors and fraud. However, many existing e-voting systems still suffer from significant limitations, including manual voting and mining processes, a lack of anonymity in voter data, and concerns over security and transparency.

To address these challenges, we have modified a previously designed blockchain-based e-voting system[3]. Our modified e-voting system incorporates several advanced features and functionalities that aim to enhance the transparency, security, and ease-of-use of the voting process.

One of the key enhancements we have made to the system is the automation of the entire voting process, which eliminates the need for manual intervention and makes the process faster and more efficient. We have also automated the mining process, making it faster, more efficient, and organized.

In addition, we have implemented advanced features like maintaining voter anonymity by encrypting their voter ID, adding voting time, tracking the number of blocks voted, and also the average block creation time. These features not only ensure voter privacy but also increase the transparency of the voting process.

Furthermore, we have transformed the static page of the e-voting web application into a dynamic one, allowing for greater interactivity and ease-of-use. Our modified e-voting system represents a significant improvement over existing systems and demonstrates the potential of blockchain technology to revolutionize the way we conduct elections.

In summary, our advanced features and functionalities aim to create a more secure, transparent, and efficient e-voting system that can increase voter confidence in the electoral process. By addressing the limitations of existing e-voting systems, we believe our modified system has the potential to significantly improve the way we conduct elections.

System Architecture:

The e-voting system architecture consists of a front-end web application built using HTML, CSS, and JavaScript, a back-end server built using Flask, and two Python scripts for automating the voting and mining processes.

The front-end web application allows voters to cast their votes securely and anonymously. The application is designed to be user-friendly and intuitive, with a modern and responsive design. The voting process is automated using the voting.py script, which sends encrypted votes to the pool at a rate of 10 votes per second. The votes are encrypted using the SHA-256 algorithm, ensuring that they cannot be tampered with.

The back-end server is responsible for processing the votes and mining new blocks. It is built using Flask, a lightweight web framework for Python. The server is designed to be scalable and can handle a large number of requests simultaneously. The mining process is automated using the mining.py script, which mines and appends a new block every 10 seconds. The script ensures that unmined transactions are processed and added to the blockchain in a timely and efficient manner.

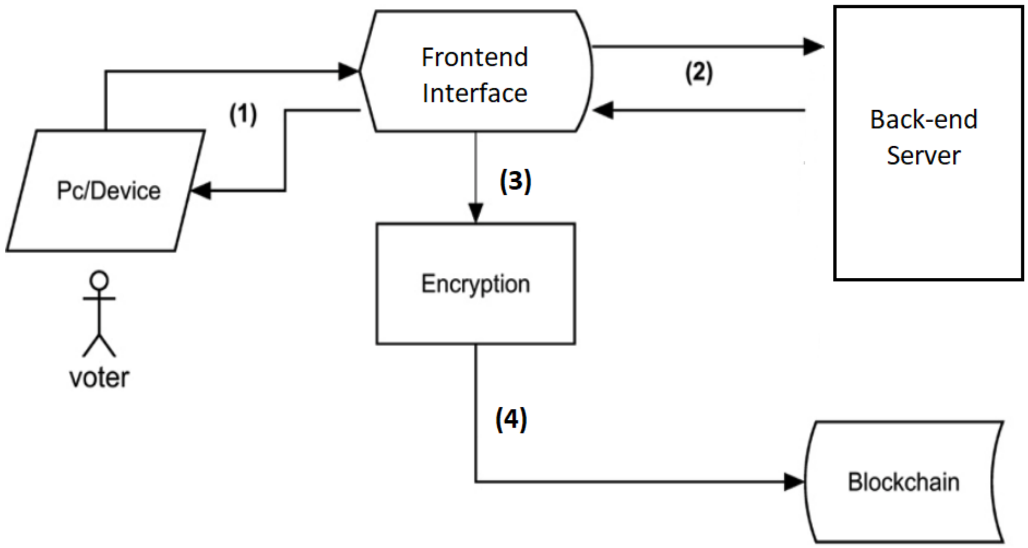

Figure 1: "E-Voting system simplified architecture"

The e-voting system also includes several advanced features and functionalities that enhance the transparency, security, and ease-of-use of the voting process. These include maintaining voter anonymity by encrypting their voter ID, adding voting time, tracking the number of blocks mined, the vote count for each party and also the average block creation time. The details of these features are dynamically updated in the web application, allowing voters to monitor the progress of the election in real-time.

Overall, the system architecture is designed to be robust, scalable, and secure, ensuring that the voting process is efficient and tamper-proof. The system represents a significant improvement over existing e-voting systems, demonstrating the potential of blockchain technology to revolutionize the way we conduct elections.

Implementation Details:

The blockchain-based e-voting system was implemented using a variety of tools and technologies to ensure transparency, security, and efficiency of the voting process. The system was built using Python, with the Flask web framework used to create the frontend user interface, while the backend scripts, voting.py and mining.py, were also written in Python. To interact with the blockchain network, the class and methods written are used, which provided the core functionality for maintaining the tamper-evident and tamper-resistant digital ledger of all the votes cast in the system. Additionally, the Python Requests library was used to enable communication between the backend and the blockchain network.

To ensure the security and anonymity of the voting process, the SHA256 encryption algorithm was used to encrypt each vote before it was added to the blockchain. This ensured that each vote was anonymous and could not be tampered with once added to the blockchain.

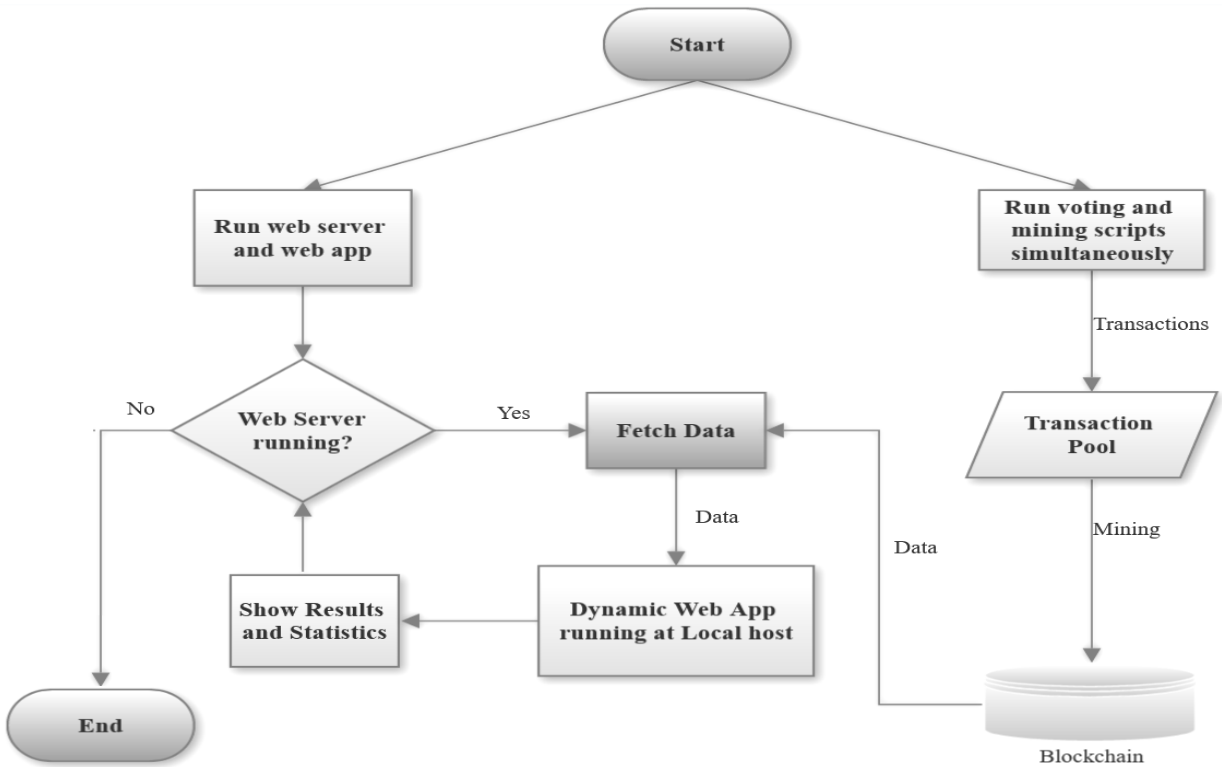

Figure 2: "Flowchart of the E-Voting System"

To automate the voting process, the *voting.py* script was designed to cast *10,000 votes* at a rate of *10 votes per second*. The mining.py script was also automated to mine unmined transactions after every 10 seconds, ensuring that the blockchain network was continuously updated with the latest transactions.
To add advanced features to the e-voting system, the voting and mining times were recorded, along with the number of blocks voted and the average block creation time. These details were dynamically displayed in the frontend user interface, enabling voters to track the progress of the voting and mining processes in real-time.

Furthermore, to ensure transparency and accountability of the e-voting system, advanced features and functionalities were incorporated to allow for real-time monitoring of the voting process. These features included the ability to show the leading party, to calculate the number of blocks in the blockchain, the total voting time, and the average mining time and the total vote gain.

The leading party is displayed in real-time by comparing their gained votes, with the party having the highest vote count identified and displayed in the frontend user interface as the leading party. This was made possible by implementing a vote counting functionality that continuously updated the vote count and determined the leading party.

The number of blocks in the blockchain was calculated using the built-in method for determining the length of the chain. This method returned the number of blocks in the chain, which was displayed in the web app for users to monitor the progress of the voting process.

The total voting time was calculated by subtracting the voting starting time from the time of the last block in the chain. The voting starting time was recorded when the first vote was cast, while the time of the last block was obtained from the timestamp of the last block added to the chain.

Voting_time = last block timestamp - first transaction timestamp

The average mining time was calculated by first determining the total time spent mining blocks. This was determined by subtracting 10 seconds (the time delay between each block) multiplied by the number of blocks from the total voting time. The resulting value was then divided by the number of blocks to obtain the average mining time per block.

Total_mining_time = Voting_time - 10(No_of_blocks)*

All these calculations were performed dynamically in the web app using Python's Flask web framework and were updated in real-time as new blocks were added to the blockchain. The resulting values were displayed to users in the web app, providing them with a clear understanding of the progress and performance of the e-voting system.

By incorporating these advanced features and functionalities, the implementation of the blockchain-based e-voting system ensured transparency, security, and efficiency of the voting process. The use of blockchain technology and automation streamlined the voting process and increased voter confidence in the electoral process.

Results and Evaluation:

The implementation of the modified e-voting system proved to be successful in improving the efficiency and scalability of the voting and mining processes. By automating the voting and mining process, we were able to test the system with **10,000 votes** casted in about **17 minutes**, which is a significant improvement over the previous manual system.

In addition, by limiting the maximum number of transactions per block to approximately 100 through a fixed delay between mining blocks, we were able to achieve an average mining time of 0.06 seconds and an average of 96 transactions per block. The total number of blocks in the blockchain was recorded as 105, providing a clear indication of the level of activity and participation in the voting process.

Mining Statistics	
Total Voting time	*17 min 27 sec*
Total Blocks mined	*105*
Total Mining time	*7 sec*
Transactions per block	*96*
Mining time per block	*0.0673 sec*

Figure 3: "Table of result obtained"

The modified e-voting system was evaluated based on its ability to improve the efficiency, scalability, and security of the voting and mining processes. The results indicate that the implementation of an automated system using blockchain technology and limiting the maximum number of transactions per block can significantly improve the efficiency and scalability of the voting and mining processes.

The use of SHA256 encryption algorithm ensures the security and anonymity of the votes cast in the system. The dynamic display of the number of blocks, total voting time, and average mining time in the frontend user interface also provides transparency and accountability, increasing voter confidence in the electoral process.

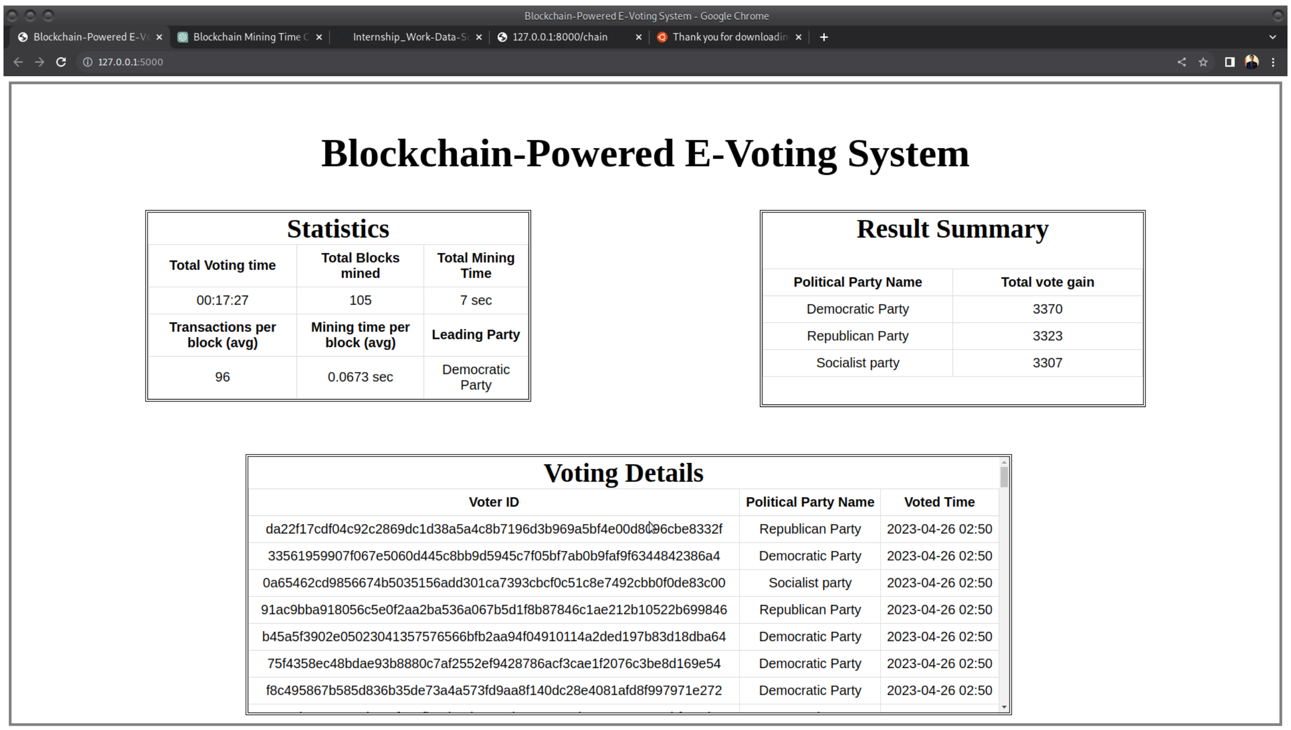

Figure 4: "Voting results"

However, it is important to note that further testing and evaluation of the system is required to ensure its long-term viability and security. This includes testing the system under different scenarios, such as increased voting traffic or attempts at tampering with the blockchain, and evaluating the effectiveness of the security measures in place.

Conclusion:

In conclusion, the implementation of a blockchain-based e-voting system and the Flask web framework has proven to be a viable solution for improving the efficiency, scalability, security, and transparency of the voting and mining processes. The automation of the voting and mining processes using the voting.py and mining.py scripts has significantly improved the speed and accuracy of the voting process, while the use of SHA256 encryption algorithm ensures the security and anonymity of the votes cast.

The advanced features and functionalities incorporated in the e-voting system, such as the real-time monitoring of the number of blocks, total voting time, and average mining time in the frontend user interface, provide transparency and accountability, increasing voter confidence in the electoral process.

However, further testing and evaluation of the system is necessary to ensure its long-term viability and security. Continued research and development in the field of blockchain-based e-voting systems are essential to identify and mitigate any potential vulnerabilities in the system.

Overall, the implementation of a blockchain-based e-voting system holds great promise for revolutionizing the electoral process by providing a more efficient, secure, and transparent means of conducting elections.

References:

1. Deloitte, "What is Blockchain?", https://www2.deloitte.com/content/dam/Deloitte/uk/Documents/Innovation/deloitte-uk-what-is-blockchain-2016.pdf

2. NISTIR 8202, "Blockchain Technology Overview", https://doi.org/10.6028/NIST.IR.8202

3. Ramesh Adhikari, "E-voting-system-using-blockchain-and-python", https://github.com/me-hem/E-voting-system-using-blockchain-and-python

4. Ahmed Ben Ayed, "A Conceptual Secure Blockchain-Based Electronic Voting System", https://www.researchgate.net/publication/341498272_A_Conceptual_Secure_Blockchain_Based_Electronic_Voting_System

5. Carmen Holotescu, Victor Holotescu, Tudor Holotescu, "Understanding Blockchain Technology and how to get involved", https://www.researchgate.net/publication/346463547_Understanding_Blockchain_Technology_and_how_to_get_involved