

IT433 - BLOCKCHAIN TECHNOLOGY

END-SEMESTER PROJECT EVALUATION

**BLOCKCHAIN-BASED PREFERENTIAL VOTING SYSTEM WITH
DYNAMIC RE-VOTING**

A D MAHIT NANDAN - 211AI001

PRANAV KAPPARAD - 211AI026

SHAAD AKTHAR - 211AI032



ABSTRACT

- We propose a blockchain-based voting system that incorporates preferential voting (ranked-choice voting) and dynamic re-voting (allowing voters to modify their preferences within a voting window).
- The system will ensures transparency, tamper-proof results, and flexibility in voter decisions.
- Built using Solidity smart contracts on the Ethereum blockchain, it will supports secure, decentralized voting with vote counting and reallocation based on voter preferences.

INTRODUCTION

Voting Systems and Challenges:

- Traditional voting systems often face issues like lack of transparency, tampering, and inflexibility.
- Blockchain technology offers a way to decentralize and secure voting, making it transparent and resistant to fraud.
- Preferential voting is a more democratic method of voting, ensuring the winner has broad support.

LITERATURE REVIEW

1. **A Theoretical Examination of the Ranked Choice Voting Procedure:** It examines Ranked Choice Voting (RCV) in the context of social choice theory, highlighting its advantages over first-past-the-post (FPTP), such as reducing wasted votes and better reflecting voter preferences.
2. **BE-Voting: A Secure Blockchain Enabled Voting System:** A blockchain-based voting system designed for Bangladesh that enhances transparency, security, and privacy in elections. It offers a decentralized voting solution with anonymous voter data.
3. **ACB-Vote: Efficient, Flexible, and PrivacyPreserving Blockchain-Based Score Voting With Anonymously Convertible Ballots:** Presents a blockchain-based e-voting system utilizing score voting mechanisms with BBS+ signatures and convertibly linkable signatures, addressing multiple voting and computational overhead from complex range proofs.
4. **DTACB: Dynamic Threshold Anonymous Credentials With Batch-Showing:** Presents a dynamic threshold anonymous credential system that improves flexibility and scalability in decentralized environments by allowing dynamic adjustments of issuer participation without system rewinding.
5. **ElectAnon: A Blockchain-Based, Anonymous, Robust and Scalable Ranked-Choice Voting Protocol:** It focusses on enhancing voter anonymity, robustness, and scalability. By employing zero-knowledge proofs and off-chain ballot encoding, ElectAnon addresses privacy and scalability challenges.

PROBLEM STATEMENT

Current blockchain voting systems do not support advanced voting mechanisms like preferential voting, and once a vote is cast, voters cannot change their decision. This limits voter flexibility and doesn't guarantee a broad consensus for the winner.

OBJECTIVES

- Implement a blockchain-based preferential voting system using ranked-choice voting to ensure the winning candidate has majority support.
- Enable re-voting where voters can remove their vote.
- Ensure security, transparency, and integrity using a decentralized blockchain platform (Ethereum).

SYSTEM ARCHITECTURE

Overview of the Architecture:

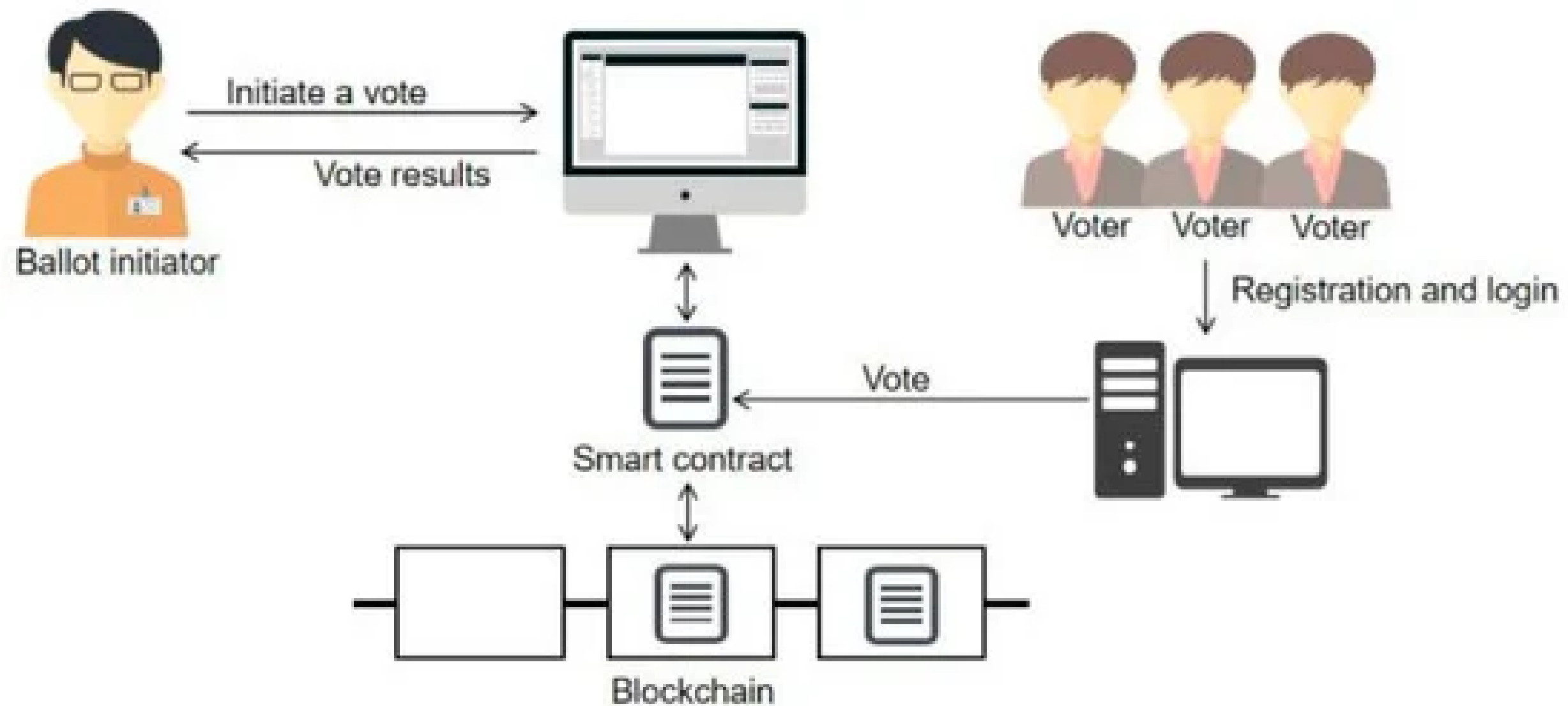
- User Interface (Front-End): Allows voters to register, rank their choices, and update their vote if needed. The interface interacts with the blockchain through a web3 provider.
- Smart Contract (Back-End): A Solidity-based smart contract deployed on Ethereum that:
 - Handles voter registration and eligibility.
 - Records voter preferences and supports dynamic re-voting.
 - Runs the vote counting and reallocation logic (round-based elimination).
- Blockchain (Ethereum): Ensures transparency, tamper-resistance, and decentralized storage of the votes.
- Consensus and Security: Smart contracts ensure that once a vote is cast, it is securely stored, and all vote changes are tracked on-chain.

SYSTEM ARCHITECTURE

Key Components:

1. Voter Registration Module: Registers voters, validates eligibility.
2. Voting Module: Records and updates ranked preferences; allows re-voting.
3. Vote Counting Module: Performs vote rounds and eliminates candidates, reallocating votes based on next preferences.
4. Dynamic Re-Voting: Logic to overwrite a voter's previous rankings within the voting window.

ARCHITECTURE



METHODOLOGY

Example of Preferential Voting:

- Voter 1 ranks the candidates as: Alice > Bob > Carol.
- Voter 2 ranks the candidates as: Bob > Alice > Carol.
- Voter 3 ranks the candidates as: Carol > Bob > Alice.

The smart contract processes the votes in the following way:

1. **Initial Vote Count:** Each voter's first preference is counted.
2. Alice receives 1 vote, Bob receives 1 vote, and Carol receives 1 vote.
3. **Elimination Process:** The candidate with the fewest votes, Carol, is eliminated.
4. **Redistribution of Votes:** Voter 3's vote is redistributed to their second preference, Bob. Now, Bob has 2 votes, while Alice has 1 vote.
5. **Final Check:** After redistribution, Bob has more than 50% of the remaining votes, making Bob the winner.

METHODOLOGY

Winner Determination

Algorithm 1 Winner Determination in Ranked Choice Voting

```
While remainingVotes > 0
    Find the loser: loser  $\leftarrow$  findLoser()
    Eliminate the loser: eliminated[loser]  $\leftarrow$  True
    Decrement remainingVotes by voteCounts[loser]
    Redistribute votes: RedistributeVotes()
    For each candidate:
        If voteCounts[candidate] > remainingVotes / 2
            Declare the winner: winner  $\leftarrow$  candidate
            Emit AnnounceWinner(winner)
        Return
    End For
End While
```

RESULTS AND ANALYSIS

ADD CANDIDATE FUNCTION

DEPLOY & RUN TRANSACTIONS

Transactions recorded 3

Deployed Contracts 1

VOTING AT 0X703...420E0 (MEMORY)

Balance: 0 ETH

addCandidate 0x5B38Da6a701c568545dCfcB03FcB875f56beddC4, "A"

addVote string[] preferences

removeVote

startVoting

stopVoting

candidateName address

candidates uint256

eliminated address

getVote address voterAddress

getWinner

isVoting

nameToAddress string

remainingVotes

totalVotes

voteCounts address

```
1 // SPDX-License-Identifier: MIT
2 pragma solidity >=0.5.0;
3
4 contract Voting {
5     bool public isVoting;
6     uint public totalVotes;
```

Transaction to Voting.addCandidate pending ...

[vm] from: 0x5B3...eddC4 to: Voting.addCandidate(address,string) 0x703...420E0 value: 0 wei data: 0x9a5...00000 logs: 0 hash: 0x83c...a4aaa

status 0x1 Transaction mined and execution succeed

transaction hash 0x83c9e1543f38f70fe0de2620f146651967edad792656f876fb1063854f3a4aaa

block hash 0xd9e3ac008d72942dfb9b31704e27e92e01ad9804997cdc41917eb7e488f9d45c

block number 190

from 0x5B38Da6a701c568545dCfcB03FcB875f56beddC4

to Voting.addCandidate(address,string) 0x703879Ca741c310b92A39D72B34C6D73D13420E0

gas 134999 gas

transaction cost 117390 gas

execution cost 95538 gas

input 0x9a5...00000

output 0x0001

decoded input { "address candidate": "0x5B38Da6a701c568545dCfcB03FcB875f56beddC4", "string name": "A" }

decoded output { "0": "bool: true" }

I'm here to help you!

RESULTS AND ANALYSIS

ADMIN-ONLY RESTRICTION FOR ADDING CANDIDATES

[illegible]

RESULTS AND ANALYSIS

ADD VOTE FUNCTION

DEPLOY & RUN TRANSACTIONS

Transactions recorded 6

Deployed Contracts 1

VOTING AT 0X703...420E0 (MEMORY)

Balance: 0 ETH

addCandidate

address candidate, string name

addVote

["A", "B", "C"]

removeVote

startVoting

stopVoting

candidateNa...

address

candidates

uint256

eliminated

address

getVote

address voterAddress

getWinner

isVoting

nameToAddress

string

remainingVotes

totalVotes

voteCounts

address

1 // SPDX-License-Identifier: MIT

2 pragma solidity >=0.5.0;

3

4 contract Voting {

5 bool public isVoting;

6 uint public totalVotes;

0

Listen on all transactions

Filter with transaction hash or address

[vm] from: 0x5B3...eddC4 to: Voting.addVote(string[]) 0x703...420E0 value: 0 wei

data: 0x392...00000 Logs: 1 hash: 0x32e...76ba9

Debug

status

0x1 Transaction mined and execution succeed

transaction hash

0x32e36f25411d09eb2a2d34d99bd898301fb6a66e14abb5b961aae5374b976ba9

block hash

0xacc711a2b0c3b687540c06db6df4b8d6e414e4550438fae63fdaff607e75e83f

block number

193

from

0x5B38D0a6a701c568545dCfcB03FcB875f56beddC4

to

Voting.addVote(string[]) 0x703879Ca741c310b92A39D72B34C6D73D13420E0

gas

280021 gas

transaction cost

243496 gas

execution cost

220892 gas

input

0x392...00000

output

0x0001

decoded input

{
 "string[] preferences": [
 "A",
 "B",
 "C"
]
}

decoded output

{
 "0": "bool: true"
}

I'm here to help you!

RESULTS AND ANALYSIS

ONE VOTE PER PERSON RESTRICTION

[illegible]

RESULTS AND ANALYSIS

GET WINNER FUNCTION

[illegible]

CONCLUSION AND FUTURE WORKS

CONCLUSION

- **Advancement in Electoral Modernization:** Blockchain and Ranked Choice Voting (RCV) enhance transparency, security, and fairness in elections.
- **Seamless Smart Contract Functionality:** Secure, auditable vote recording, candidate elimination, and vote redistribution validate blockchain's practicality.

FUTURE WORKS

- **User Interface:** Simplify for accessibility and broader participation.
- **Enhanced Security:** Integrate zero-knowledge proofs for privacy.
- **Gas Optimization:** Improve gas efficiency and assign gas limits effectively.
- **Multi-language Support:** Expand usability in diverse contexts.

REFERENCES

- 1)** E. Khan, M. H. Rahman, S. Mustary, I. Islam and S. R. Mahmud, "BE-Voting: A Secure Blockchain Enabled Voting System," 2022 4th International Conference on Electrical, Computer & Telecommunication Engineering (ICECTE)
- 2)** W. Xue, Y. Yang, Y. Li, H. H. Pang and R. H. Deng, "ACB-Vote: Efficient, Flexible, and Privacy- Preserving Blockchain-Based Score Voting With Anonymously Convertible Ballots," in IEEE Transactions on Information Forensics and Security
- 3)** C. Li, J. Ning, S. Xu, C. Lin, J. Li and J. Shen, "DTACB: Dynamic Threshold Anonymous Credentials With Batch-Showing," in IEEE Transactions on Information Forensics and Security
- 4)** Ceyhun Onur and Arda Yurdakul. 2023. ElectAnon: A Blockchain-based, Anonymous, Robust, and Scalable Ranked-choice Voting Protocol. Distrib. Ledger Technol. 2, 3, Article 19 (September 2023), 25 pages. <https://doi.org/10.1145/3598302>
- 5)** Nurmi, H., Palha, R.P. (2021). A Theoretical Examination of the Ranked Choice Voting Procedure. In: Nguyen, N.T., Kowalczyk, R., Motylska-Kuźma, A., Mercik, J. (eds) Transactions on Computational Collective Intelligence XXXVI