**CPSC 559- Advanced Blockchain Technologies Project SPRING 2024**

## Submitted to professor:

Dr.Wenlin Han

## Team Name:

Team ABCD

# Team Members:

|  |  |  |
| --- | --- | --- |
| **Names** | **CWID** | **Email** |
| Rudra Kartik Mankad | 885195628 | [rudramankad@gmail.com](mailto:rudramankad@gmail.com) |
| Zhiyuan Yang | 885081174 | [yang1018@csu.fullerton.edu](mailto:yang1018@csu.fullerton.edu) |
| Tejaskumar Pareshbhai Patel | 885174433 | [tejas.p@csu.fullerton.edu](mailto:tejas.p@csu.fullerton.edu) |
| Ashil Anish Shah | 885331850 | [ashil2495@csu.fullerton.edu](mailto:ashil2495@csu.fullerton.edu) |

# Project Overview: Voting System

## Project Description:

This Solidity-built smart contract is aimed at developing a voting service that includes a possibility of delegated voting. It is based on a new voting-ballot with multiple proposals, equal voting powers for each address, and the choice for any voter to entrust the proxy for voting.

Voter: The scheme describes an individual voter with attributes, such as weight (the cumulative weight of the vote which is cast by the voter), voted (a binary flag which indicates cast vote or not), delegate (the address which the vote is delegated to), vote(the proposal index where the vote will be cast).

Proposal: It is represented by this structure and has two important attributes: name (this is used for identification of the proposal) and voteCount (total of all votes).

An important detail is a chairperson, who is the point of contact and manages voting.

## GitHub Repository Link: <https://github.com/rudramankad/Team-ABCD-Voting-with-blockchain.git>

## Enhancements Introduced:

1. Voting Reset Functionality:

* Implemented an option to reset the entire voting process from the ground up.
* Users must seek permission from this mode of the program with the "Give me right" command.

1. Election Summary Updates:

* The summary button now replaces the conventional display of the presently active contestants and their current vote counts in real time.

1. Voting Rights Acquisition:

* This "Give Me Right" button must be triggered to provide for voting permissions. Hereby, one needs to have sufficient Ethereum for voting.

1. Candidate Information Page:

* Gives current candidates together with the history of their elections or campaign mottos.

1. Election Pause and Resume Capability:

* Permits the administrators to instruct the election flow to be paused now and subsequently continuing when required.

1. Enhanced Voting Process:

* Developed a user-friendly turnout register where a candidate selection may be done easily or the voter may go for NOTA instead.

1. User Profile Transparency:

* The latest user profiles appear with more detailed data - Ethereum balances and wallets addresses - in order to enhance the level of transparency.

1. Delegation Feature:

* Facilitates the transfer of votes from one voter to another, either a person or a piece of code who can further vote on behalf of the original voter.

1. Vote Conclusion:

* Today, administrators will be able to show the voters the final result on their screen. It is also possible for the night to end when they want to and immediately proclaim the winner.

1. Web Interface Development:

* Designed a user-centric interface for the web interface that segments voter and administrator roles, simplifying the similarity of voting.

## Deployment Instructions:

* Instantiate the blockchain local gathering using Ganache.
* Do 'truffle compile' in the terminal to get the semantic contract code.
* Update the details of the Markets, and Ballot, in the index.js file as listed by Ganache.
* Employ truffle migrate for the contract.
* Launch the web interface by opening the index.html file, and it will work together with MetaMask interface to manage transactions.