# Decentralized Voting

## An Ethereum-based Decentralized Voting Platform

Authors: Luke Reddick
Advisor: Dr. Liran Ma

TCU
COLLEGE OF SCIENCE & ENGINEERING
DEPARTMENT OF COMPUTER SCIENCE

## Introduction

Voting is the civic duty of the people in a democracy, Unfortunately, the archaic methods of voting still haunt us in the 21st century. Long queues, inconvenient drives, and the perceived insignificance of a single vote deter many from participating. We need something that can provide the infrastructure for a voting system that is efficient, secure, mobile, and traceable. Thankfully, blockchain technology enables us to facilitate a voting infrastructure that can modernize voting the way we see fit.

## Problem

- Need for increased security in voting- both in implementation and voter trust
- Voting processes are archaic; require an update
- Limited existing decentralized electronic voting systems.
- Existing solutions are as bureaucratic as common physical ballot voting.
- Existing solutions are domain specific (Horizon State Technology)
- Existing solutions are complicated to many users.

## Solution

Voting on Distributed Ledger Technology (DLT), or blockchain, provides an infrastructure that can facilitate trustless and secure transactions, or votes, on a network that users can reach anywhere. To create such a decentralized voting application that reaches people, we want to leverage an existing blockchain network that people can reach without the requirement of becoming a blockchain node. Ethereum provides tools to access blockchain nodes without the user becoming a node, specifically the MetaMask browser plugin.

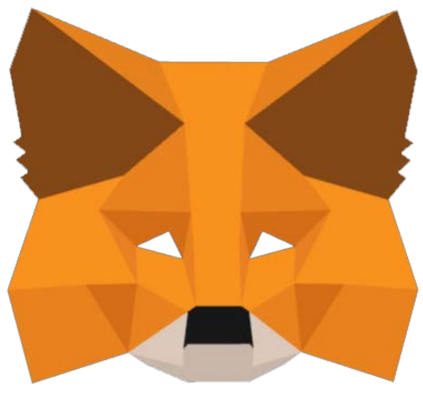
## Roles

- **Administrator:**
  - Creates the voting lobby, sets lobby to public or private, adds candidates to lobby, and sets the date and time the voting period will end. Administrators can vote during the voting period.
- **Voter:**
  - Joins lobby of choice with associated MetaMask wallet, receives voting token upon registration, votes before time ends.
- **Candidates:**
  - Added to lobby by administrator by wallet address, tokens are received by this address during voting phases, candidates can vote.

## Processes

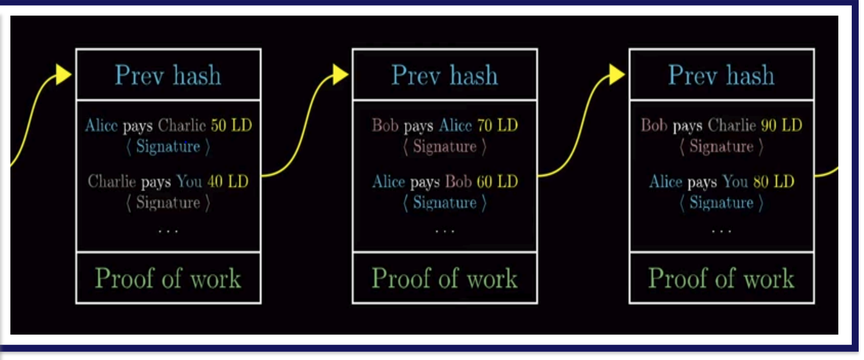
- **Registration:**
  - Search for the lobby their administrator has created, or create one themselves. Joining a lobby will either require no authentication, or a administrator-set password. Once authorized, the user will be granted a voting token and become eligible to vote.
- **Voting:**
  - The voting period has a set date and time to end, thus all voting participants have until the set time to cast their voting token. Each voter wallet gets one vote. Once a voter knows who they wish to vote for, they simply select that candidate and a voting token is removed from the voter wallet and added to the candidates wallet.
- **Tally:**
  - The tally phase is extremely simple since we do not need phsyical entities counting each individual vote. The network handles the votes in real time and once voting is over, the candidate wallet token amounts only need to be compared. The largest amass of voting tokens is declared the winner and etched into the blockchain history, viewable for all of the lobbies participants.

## Technology Stack



### Ethereum Features

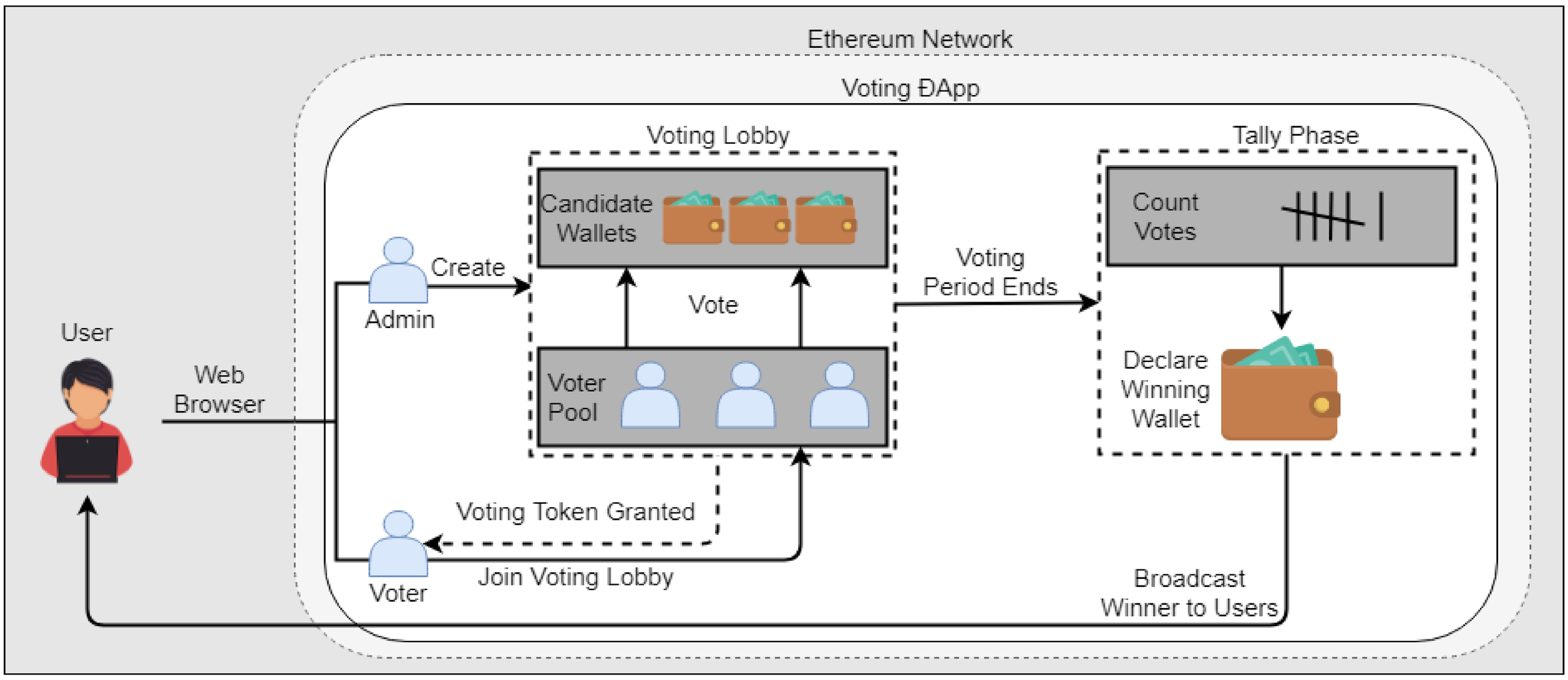
Ethereum caters to a variety of different use cases by assessing the computational power of transactions via gas, a way to accurately define the cost of various and complex transactions occuring on the network. Gas is measured in Gwei, which you can think of as a nano-Ether, or 10-9 ETH [2]. This allows the network to calculate transaction costs at a nano-Ether accurate level. For our voting platform, this means our transactions are eligible to be reflected in a block via gas. The Ethereum Virtual Machine and its support of Smart Contracts built on Solidity allow us to fulfill the full stack decentralized application we desire. This network is a haven for dApp developers entering into the new age Web 3.0.



## Abstract

This research covers a comprehensive implementation of a blockchain based voting platform. Blockchain, in its infancy, has shown remarkable use cases with cryptocurrencies and we would like to expand upon its possibilities. Voting is a system ripe with opportunity for blockchain; it requires security, consensus, and portability- all qualities inherited from blockchain technology. In this research, we discuss the appeal of blockchain technology and why we want to elevate voting to 21st century technology. Next, we discuss the design of the voting application by assessing all roles and processes involved. After a survey of the voting business logic and design, we will discuss how this will be implemented from a technical standpoint. Finally, we acknowledge any related works in the design of this project that are leveraged or used as models.


## Platform Design




## Implementation

To inherit blockchain properties we need to create a blockchain network to host our voting application. Fortunately, Ethereum provides the infrastructure for the development of such applications without the overhead of creating a new blockchain network from scratch. On the Ethereum network, these applications are known as Decentralized Applications, which we will refer to as Dapps (read as “dee-apps”).

With a Dapp on the Ethereum network, we inherit all the desirable traits of blockchain technology as well as the concurrency of working with an updated and relevant network that is always improving. To create such an application, we need to write some software. The allure of Ethereum lies in its Turing complete virtual machine that handles code written in the new dapp development language: Solidity. Here is a Hello, world!” contract written in Solidity:

```
pragma solidity ^0.4.11;
contract helloWorld {
    function renderHelloWorld( ) returns ( string ) {
        return 'helloWorld';
    }
}
```

This is a glance at an Ethereum Smart Contract, a necessary technology that will be the foundation of our voting application. Smart Contracts act as Ethereum nodes that exist as pieces of code, written at the developer’s leisure. Encoded into the blockchain, Smart Contracts allow other users on the Ethereum network, specifically our voting platform users, to interact with the code. This communication occurs on a Remote Procedure Call (RPC) server governed by web3.js, Ethereum’s JSON RPC client built with JavaScript [2]. Therefore, if a user wants to cast their ballot, their computer/networking device will invoke function calls written into the Smart Contract by talking to the contract node to carry out the necessary actions to complete a casted ballot.

## Future Work

Free time for this research was scarce during this last year, allowing time to flesh out the blueprint and theory of this application; however there is the possibility of working on an actual implementation of a voting Dapp. I hope to pursue the actual development of this project as an Ethereum smart contract, and expanding the research to more members to help guide the application to deployment.


## Acknowledgements

I, Luke Reddick, would like to personally thank Dr. Ma for all his guidance, support, and enthusiasm in working with me on this research. Without his passion and drive to work with me, I would never have found the motivation to see this project through.
Dr. Ma’s unbridled spirit has pushed me to continue expanding on this project with post-graduate implementation work. Thank you, Dr. Ma.


## About the Author and Technologies Used

Luke Reddick is a senior Computer Science Major with a Mathematics minor from Kansas City. The application will be written in Solidity to be deployed on the Ethereum blockchain with authentication schemes provided by MetaMask.

## References


[1] C. Pathirana, “What is ethereum?” A Medium Corporation, 2017.
[2] V. Buterin and G. Contributors, “Ethereum white paper: A nextgeneration smart contract and decentralized application platform,” 2015. [Online]. Available: https://github.com/ethereum/wiki/wiki/White-Paper
[3] C. Jeria, J. Moreau, K. Serrano, F. Pangilinan, T. Huang, Kumavis, and D. Finlay, “Metamask: Brings ethereum to your browser,” 2016. [Online]. Available: https://metamask.io/
[4] B. Giang and the Crpyto Kitties Team, “Version 2.0 cryptokitties: Collectible and breedable cats empowered by blockchain technology white pa-purr,” 2017. [Online]. Available: https://www.dropbox.com/s/ a5h3zso545wuqkm/CryptoKitties WhitePapurr V2.pdf?dl=0