Palestine Technical University-Kadoorie

Faculty of Engineering and Technology

Computer Systems Engineering Department

Special Topics in Computer Engineering (12140537)

Voting System Smart Contract.

Prepared By:

Fayha' AbuSalah – 201910068

Mays Qasem – 201910019

Mays Hawwa - 201910360

Submitted To:

Dr. Mahmoud Sawalha.

December 23, 2022

Table of Contents

1. Abstract

Electronic voting or e-voting has been used in varying forms since 1970s with fundamental benefits over paper-based systems such as increased efficiency and reduced errors. However, there remain challenges to achieve wide spread adoption of such systems especially with respect to improving their resilience against potential faults. Blockchain is a disruptive technology of current era and promises to improve the overall resilience of e-voting systems. The proposed scheme conforms to the fundamental requirements for e-voting schemes and achieves end-to-end verifiability. The report presents details of the proposed e-voting scheme along with its implementation using Remix platform.

2. The Voting Process

We now describe a typical interaction of a user with the proposed scheme based on our current implementation of the system. Typically, a voter logs into the system by providing his/her thumb impression. If the match is found, the voter is then presented with a list of available candidates with the option to cast vote against them. On the contrary, if the match is unsuccessful, any further access would be denied. This function is achieved using appropriate implementation of the authentication mechanism and predefined role-based access control management. Furthermore, it is also envisioned that a voter is assigned to their specific constituency and this information is used to develop the list of candidates that a voter can vote for. The assignment of voter to a constituency is rendered an offline process and therefore out of scope of this research. After a successful vote-cast, it is mined by multiple miners for validation following which valid and verified votes are added into public ledger. The security considerations of the votes are based on blockchain technology using cryptographic hashes to secure end-to-end verification. To this end, a successful vote cast is considered as a transaction within the blockchain of the voting application. Therefore, a vote cast is added as a new block (after successful mining) in the blockchain as well as being recorded in data tables at the backend of the database. The system ensures only one-person, one-vote (democracy) property of voting systems. This is achieved by using the voter's unique thumbprint, which is matched at the beginning of every voting attempt to prevent double voting. A transaction is generated as soon as the vote is mined by the miners

which is unique for each vote. If the vote is found malicious it is rejected by miners. After validation process, a notification is immediately sent to the voter through message or an email providing the above defined transaction id by which user can track his/her vote into the ledger. Although this functions as a notification to the voter however it does not enable any user to extract the information about how a specific voter voted thereby achieving privacy of a voter. It is important here to note that cryptographic hash for a voter is the unique hash of voter by which voter is known in the blockchain. This property facilitates achieving verifiability of the overall voting process. Furthermore, this id is hidden and no one can view it even a system operator cannot view this hash therefore achieving privacy of individual voters.

3. Voting System Smart Contract

Electronic voting systems have replaced paper-based systems, but even now, people doubt the voting system's ability to secure the data and defend against any attacks. The blockchain-based system can ensure transparent and publicly verifiable elections in the country. If implemented successfully, voting can be done using a mobile application that is attached to a blockchain system.

3.1 Features

- The owner of the contract can input one or more choices to be voted by people.
- The owner of the contract can specify the start time and end time for the voting period.
- A voter can vote for any choices set by the contract owner during the voting period.
- A voter can only vote once during the voting period.
- The smart contract can return the number of votes for each choice.
- Anyone can set up a voting system through the same smart contract.

3.2 Tools

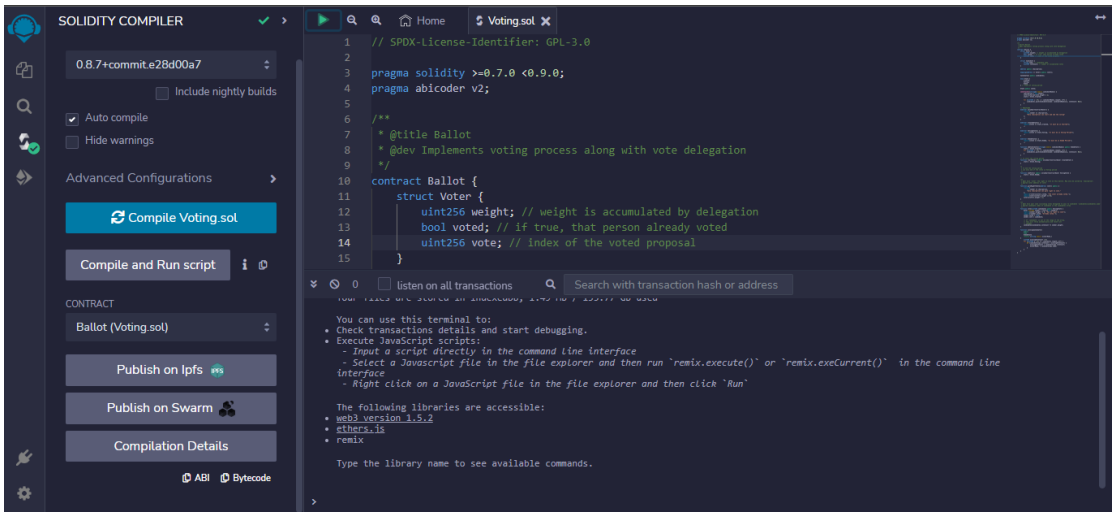
Remix IDE: https://remix.ethereum.org/

4. Implementation and Deployment

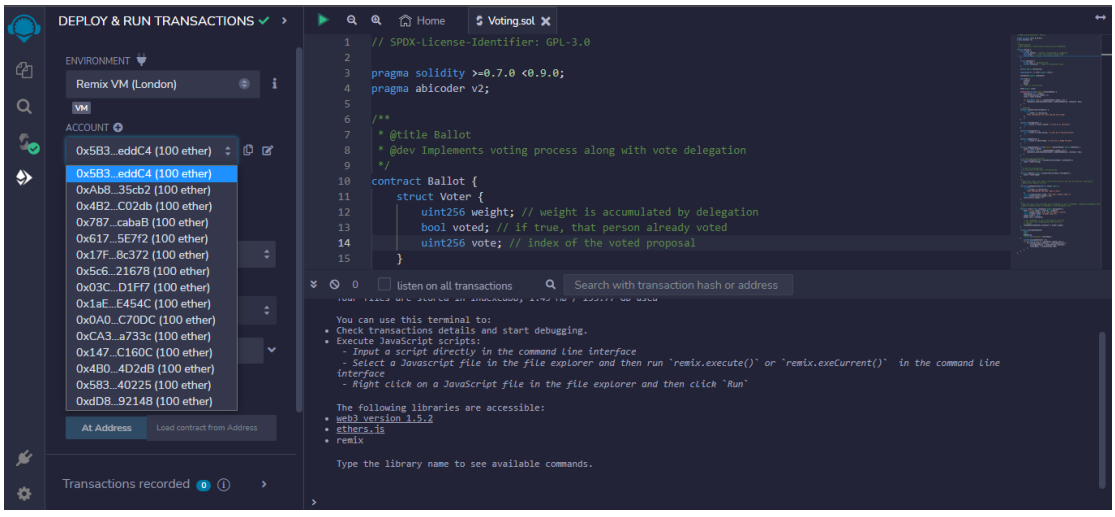
Implementation attached in voting.sol file

4.1 How to Deploy

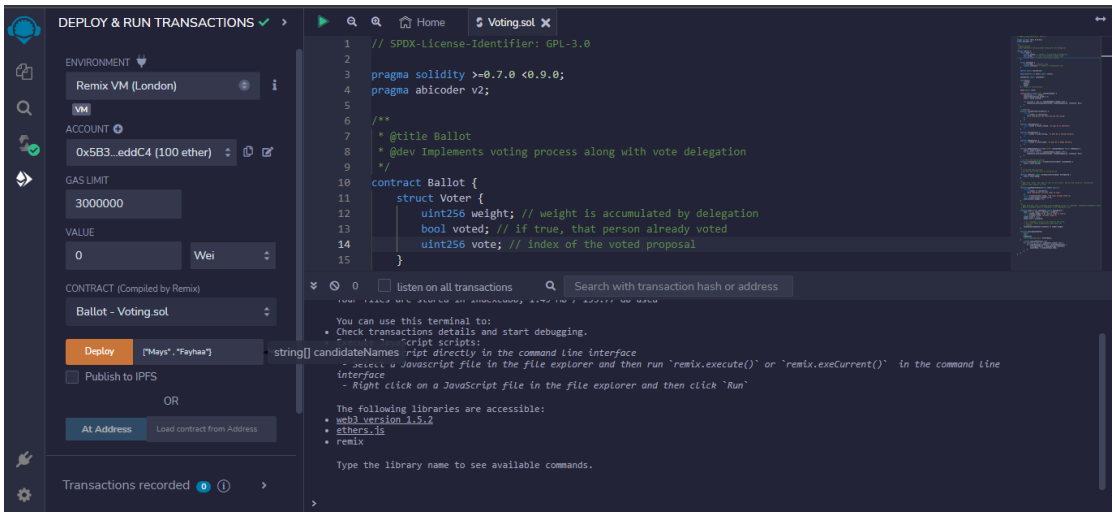
- Compile the code



- Select the smart contract owner address

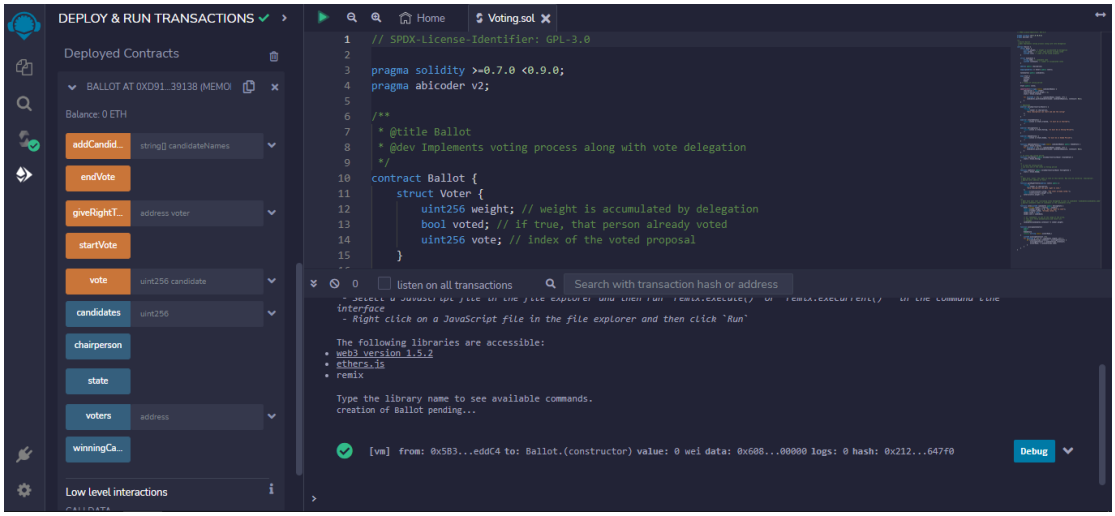


- Input candidates in Deploy button and then pressed Deploy the owner of the contract can input one or more choices to be voted by people.



4.2 How to Test

- After deploy the smart contract pressed startVote button, the owner of the contract can specify the start time and end time for the voting period.

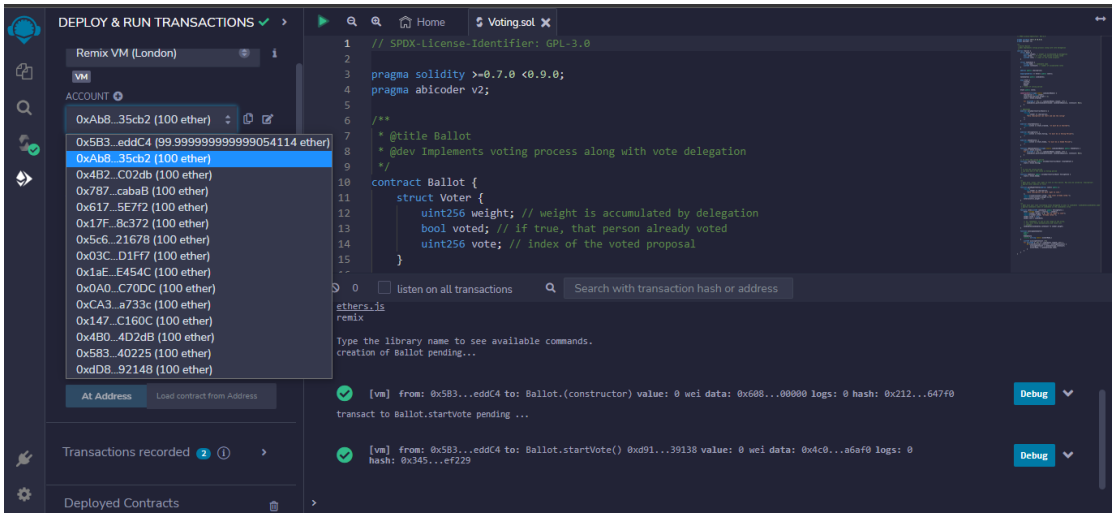


The owner specify the start time & go to select an address for the voters

- Change to another address, and copy it into giveRightToVote button

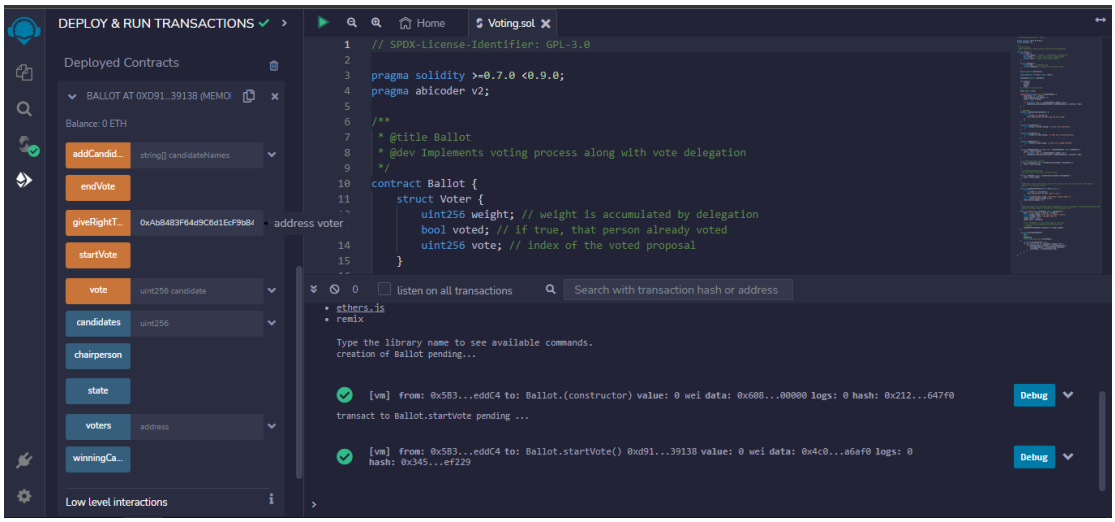
● A voter can vote for any choices set by the contract owner during the voting period.

● A voter can only vote once during the voting period.

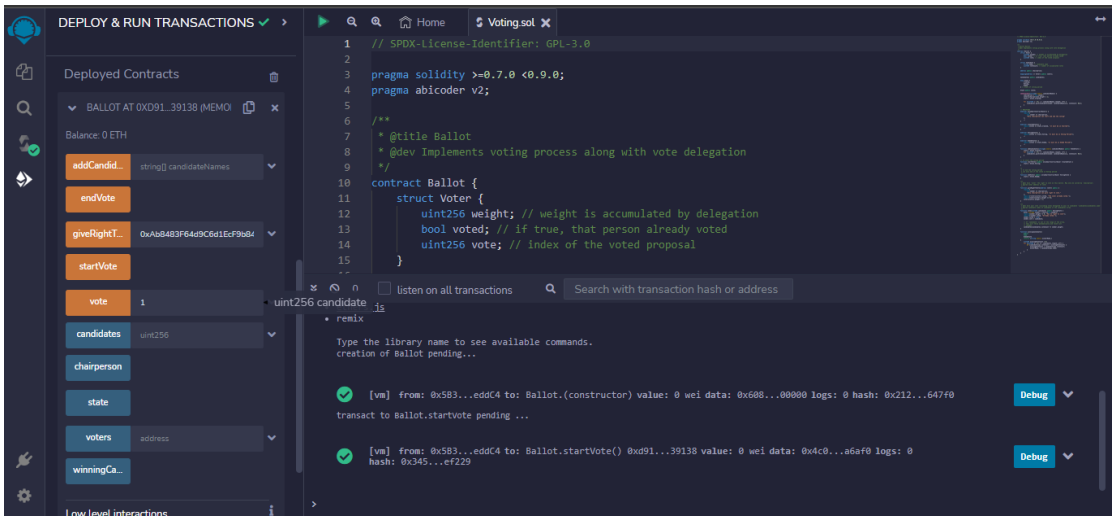


- Change back to owner address to give the right vote to other account

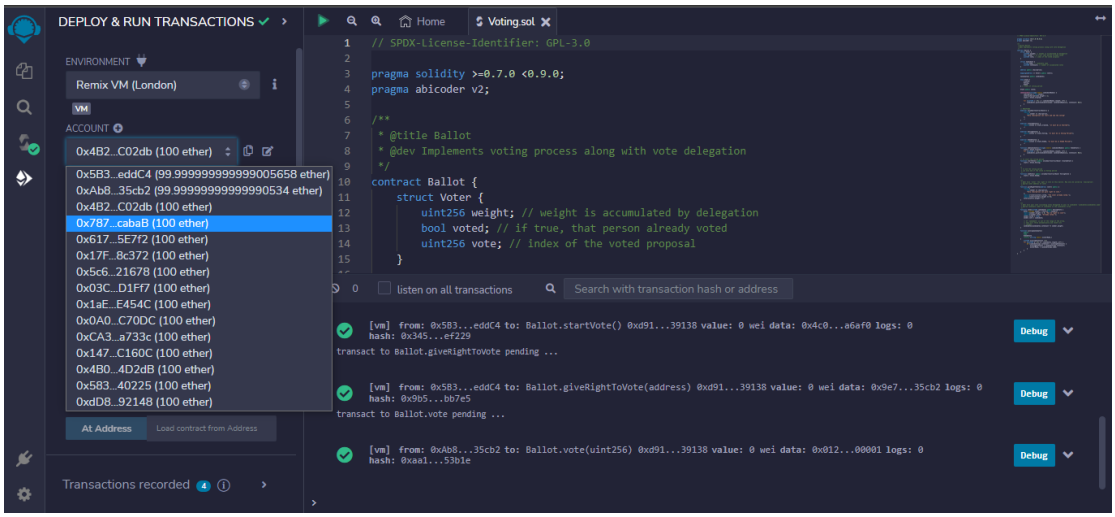
* notes: only owner of the contract can give right to vote



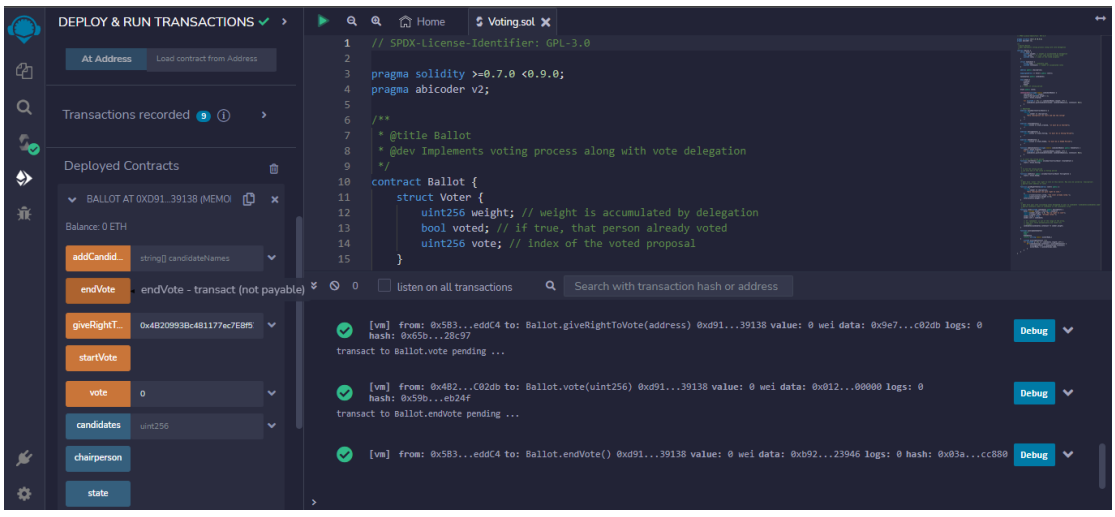
- Change back to address that already have right to vote. Input index of candidate and click vote



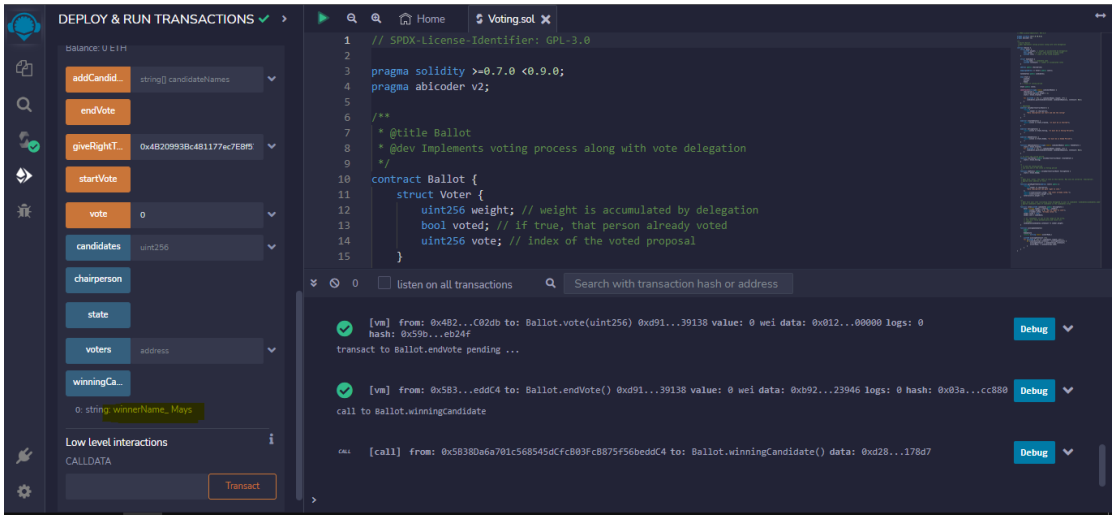
- Repeat step from 2-5 to give other account to have right and vote



- Change back to owner address to end the voting period

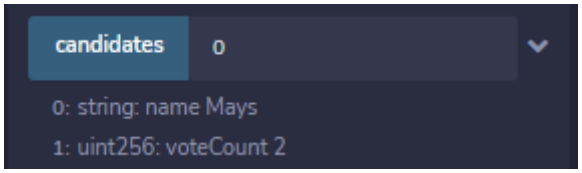


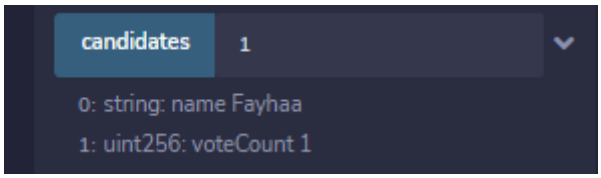
- Pressed WinningCandidate button



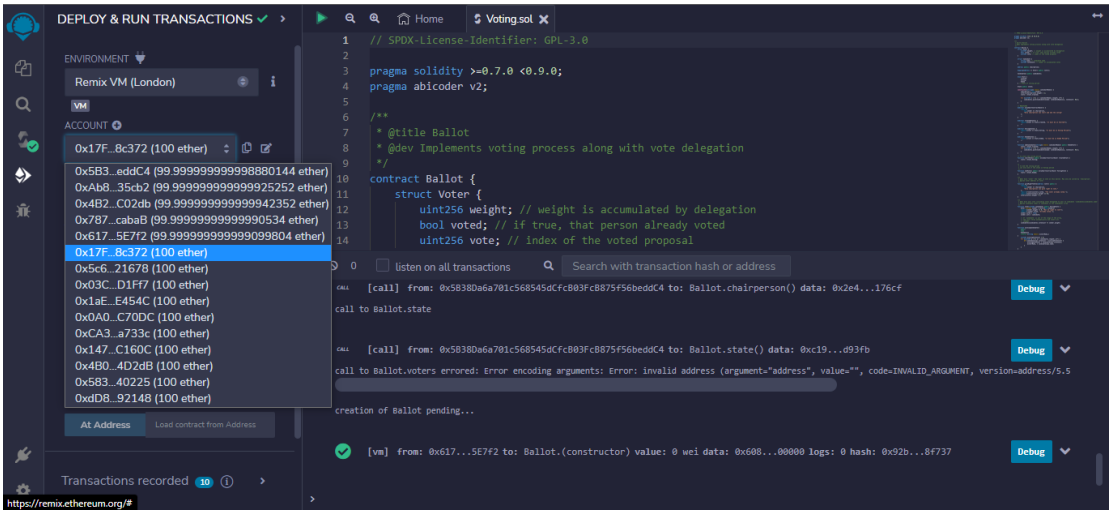
- Input index in candidates button to check the score

*The smart contract can return the number of votes for each choice.

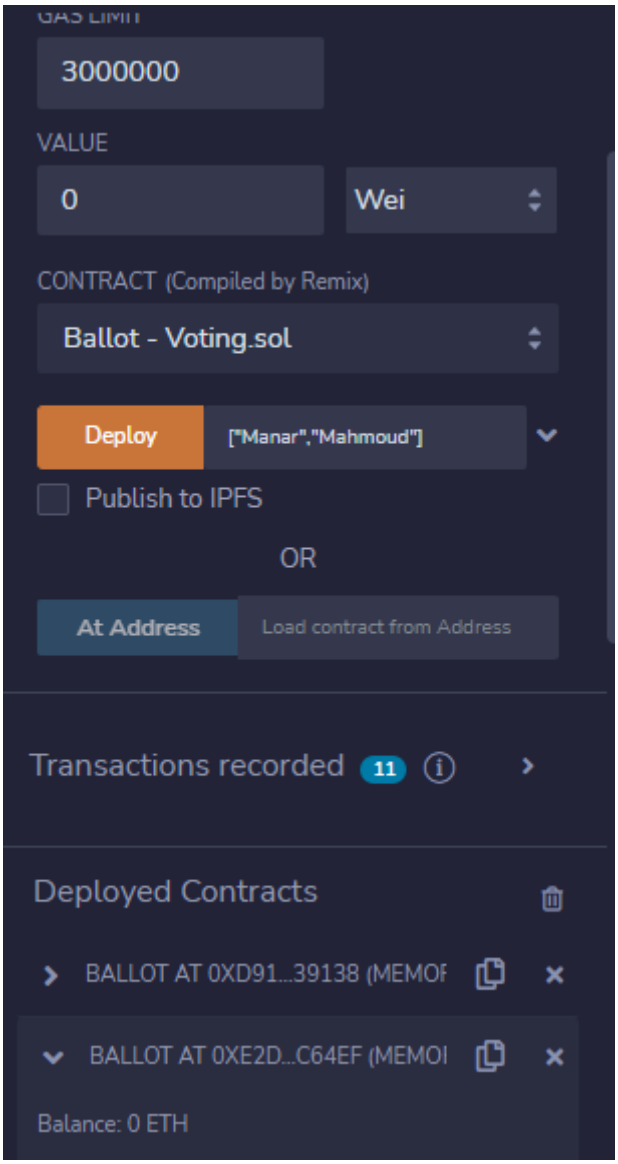




- Change to other account so anyone can set up a voting system through the same smart contract.



- Input candidates in addCandidates button, and so on



5. References

https://dev.to/niharrs/2-voting-smart-contract-2h7o

Lecture Notes on Data Engineering and Communications Technologies book series (LNDECT,volume 23)

https://core.ac.uk/download/pdf/155779036.pdf