

Τεχνολογίες Blockchain και Εφαρμογές

Ναταλία Βοριζανάκη Π20035, Θεοδώρα Δόριζα Π20244

10 Ιουλίου 2024

To documentation δημιουργήθηκε μέσω overleaf

1 Ανάλυση Κώδικα

Το smart contract "PollManager" επιτρέπει στους χρήστες να δημιουργούν και να συμμετέχουν σε ψηφοφορίες στο blockchain του Ethereum. Κάθε ψηφοφορία έχει έναν δημιουργό, μια ερώτηση για την οποία γίνεται η ψηφοφορία, και καταγράφει τις ψήφους για τις απαντήσεις "ναι" και "όχι". Το συμβόλαιο περιλαμβάνει λειτουργικότητα για τη δημιουργία ψηφοφοριών, την υποβολή ψήφων, το κλείσιμο ψηφοφοριών και την ανάκτηση αποτελεσμάτων και λεπτομερειών εκλογών.

1.1 Election

```
struct Election {  
    address creator;  
    string question;  
    bool isOpen;  
    uint256 yesVotes;  
    uint256 noVotes;  
    uint256 startTimestamp;  
    uint256 endTimestamp;  
    mapping(address => bool) hasVoted;  
    Voter[] votes;  
}
```

address creator: Η διεύθυνση του χρήστη που δημιούργησε την ψηφοφορία.

string question: Η ερώτηση που τίθεται για ψηφοφορία.

bool isOpen: Μια λογική τιμή που υποδεικνύει εάν η ψηφοφορία είναι ανοιχτή.

uint256 yesVotes: Ο αριθμός των ψήφων "ναι".

uint256 noVotes: Ο αριθμός των ψήφων "όχι".

uint256 startTimestamp: Η χρονική στιγμή έναρξης της ψηφοφορίας.

uint256 endTimestamp: Η χρονική στιγμή λήξης της ψηφοφορίας.

mapping(address => bool) hasVoted: Ένας χάρτης που καταγράφει αν μια διεύθυνση έχει ήδη ψηφίσει.

Voter[] votes: Μια λίστα με τις ψήφους που έχουν υποβληθεί.

1.2 Voter

```
struct Voter {  
    address voterAddress;  
    bool vote;  
}
```

voterAddress: Η διεύθυνση του ψηφοφόρου.

vote: Η ψήφος του.

1.3 Events

```
event ElectionCreated(uint256 electionID, string question, address creator);  
event ElectionConcluded(uint256 electionID, address concludedBy);  
event VoteSubmitted(uint256 electionID, address voter, bool vote);
```

ElectionCreated: Χρησιμοποιείται όταν δημιουργείται μια νέα ψηφοφορία.


ElectionConcluded: Χρησιμοποιείται για την λήξη μιας ψηφοφορίας.

VoteSubmitted: Χρησιμοποιείται όταν υποβάλλεται μια ψήφος.

1.4 Συναρτήσεις


```
function createElection(string memory question) public returns (uint256 electionID) {  
    electionID = elections.length;  
    Election storage newElection = elections.push();  
    newElection.creator = msg.sender;  
    newElection.question = question;  
    newElection.isOpen = true;  
    newElection.startTimestamp = block.timestamp;  
    emit ElectionCreated(electionID, question, msg.sender);  
}
```

createElection : Δημιουργεί μια νέα ψηφοφορία.

```
function closeElection(uint256 electionID) public {  infinite gas
    require(electionID < elections.length, "Invalid election ID");
    require(msg.sender == elections[electionID].creator, "Unauthorized action");
    require(elections[electionID].isOpen, "Election is already closed");

    elections[electionID].isOpen = false;
    elections[electionID].endTimestamp = block.timestamp;
    emit ElectionConcluded(electionID, msg.sender);
}
```

closeElection: Κλείνει μια ανοιχτή ψηφοφορία.

```
function submitVote(uint256 electionID, bool vote) public {  infinite gas
    require(electionID < elections.length, "Invalid election ID");
    require(elections[electionID].isOpen, "Election is closed");
    require(!elections[electionID].hasVoted[msg.sender], "You have already voted");

    Election storage currentElection = elections[electionID];

    Voter memory newVote = Voter({
        voterAddress: msg.sender,
        vote: vote
    });

    currentElection.votes.push(newVote);
    currentElection.hasVoted[msg.sender] = true;

    if (vote) {
        currentElection.yesVotes++;
    } else {
        currentElection.noVotes++;
    }

    emit VoteSubmitted(electionID, msg.sender, vote);
}
```

submitVote: Υποβάλλει μια ψήφο σε ανοιχτή ψηφοφορία, μόνο αν ο χρήστης δεν έχει ήδη ψηφίσει.

```
function getResults(uint256 electionID) public view returns (uint256 yesCount, uint256 noCount, uint256 totalVotes) {
    require(electionID < elections.length, "Invalid election ID");

    Election storage currentElection = elections[electionID];
    yesCount = currentElection.yesVotes;
    noCount = currentElection.noVotes;
    totalVotes = currentElection.votes.length;
}
```

getResults: Επιστρέφει τα αποτελέσματα μιας ψηφοφορίας. Συγκεκριμένα, επιστρέφει τον αριθμό των θετικών ψήφων, των αρνητικών ψήφων αλλά και του συνόλου των ψήφων.

```
function getElectionDetails(uint256 electionID) public view returns (string memory question, address creator, bool isOpen, uint256 start, uint256 end) {
    require(electionID < elections.length, "Invalid election ID");

    Election storage currentElection = elections[electionID];
    question = currentElection.question;
    creator = currentElection.creator;
    isOpen = currentElection.isOpen;
    start = currentElection.startTimestamp;
    end = currentElection.endTimestamp;
}
```

getElectionDetails: Επιστρέφει στοιχεία μιας ψηφοφορίας. Επιστρέφονται, η ερώτηση , η διεύθυνση του δημιουργού, αν η ψηφοφορία είναι ανοιχτή, η χρονική στιγμή έναρξης και λήξης της ψηφοφορίας.

2 Παράδειγμα Χρήσης

2.1 Τα βήματα που ακολουθήσαμε για την εκτέλεση του Smart Contract:

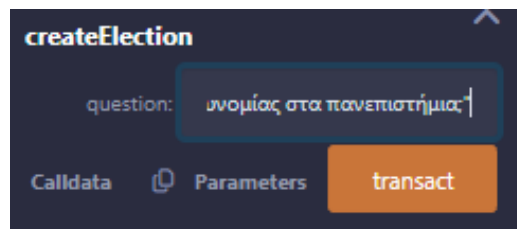
Compile

Ξεκινήσαμε κάνοντας Compile τον κώδικα μέσω του Solidity Compiler.

Deploy

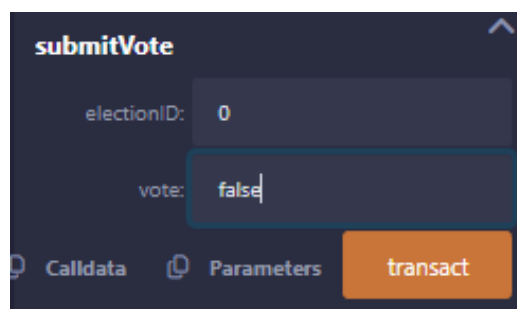
Συνεχίσαμε κάνοντας Deploy and Run Transactions.

Εκτέλεση Συναρτήσεων



The screenshot shows the 'createElection' function interface. The 'question' input field is highlighted with a blue border and contains the text 'υνομίας στα πανεπιστήμια'. Below the input field, there are three buttons: 'Calldata', 'Parameters', and 'transact'. The 'transact' button is orange and is the most prominent.

Για να δημιουργήσουμε την πρώτη ψηφοφορία, θέσαμε ως ερώτηση ‘Συμφωνείτε με την παρουσία αστυνομίας στα πανεπιστήμια’



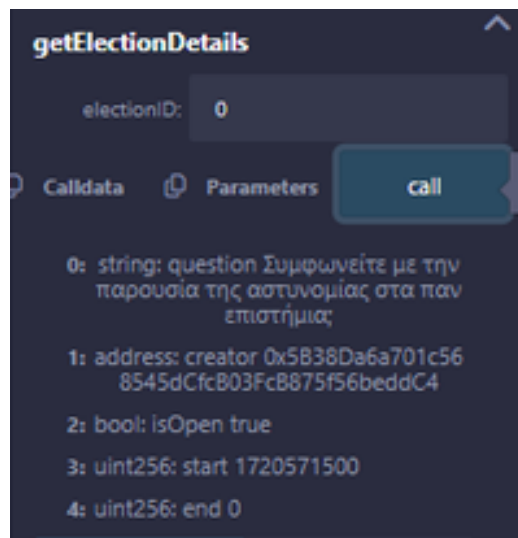
The screenshot shows the 'submitVote' function interface. The 'electionID' input field contains the value '0'. The 'vote' input field is highlighted with a blue border and contains the value 'false'. Below the input fields, there are three buttons: 'Calldata', 'Parameters', and 'transact'. The 'transact' button is orange and is the most prominent.

Στην συνέχεια υποβάλλαμε την ψήφο μας.

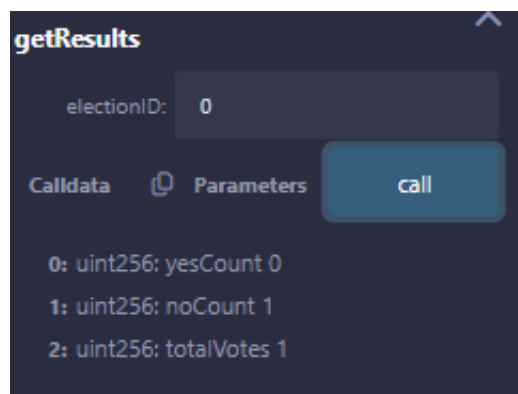
```
[vm] from: 0x583...eddC4 to: PollManager.submitVote(uint256,bool) 0x0fC...9A836 value: 0 wei data: 0x612...00000 logs: 0 hash: 0x7b5...6a368
transact to PollManager.submitVote errored: Error occurred: revert.

revert
The transaction has been reverted to the initial state.
Reason provided by the contract: "You have already voted".
You may want to cautiously increase the gas limit if the transaction went out of gas.
```

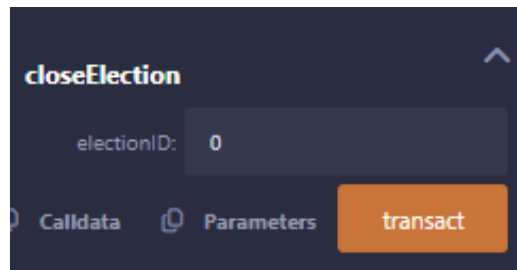
Βλέπουμε το αποτέλεσμα της προσπάθειας να υποβάλλουμε και δεύτερη ψήφο απο την ίδια διεύθυνση.



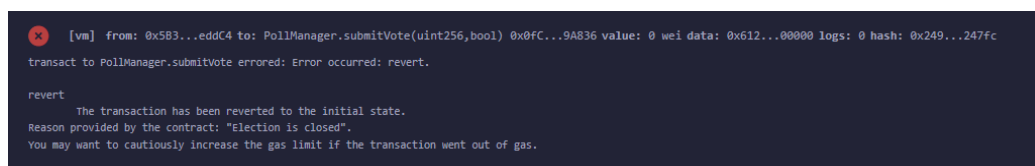
Καλώντας την getElectionDetails λάβαμε τις πληροφορίες της ψηφοφορίας.



Λάβαμε τα αποτελέσματα της ψηφοφορίας.



Λήξαμε την ψηφοφορία.



Αποτέλεσμα προσπάθειας να ψηφίσουμε αφού έχει κλείσει η ψηφοφορία.