Blockchain-Voting

Introducere

In cadrul acestui proiect am implementat un sistem de votare bazat pe tehnologia Blockchain. Scopul de a avea un astfel de sistem este de a defini o solutie ce face aproape imposibila fraudarea voturilor si de a creste astfel nivelul de incredere al votantilor. Votul este un instrument esential pentru orice formatiune democratica, iar modul in care acesta se desfasoara in mod traditional este problematic si vulnerabil din multe puncte de vedere si permite coruperea alegerilor extrem de facil.

Proiectul a fost scris folosind **Python** si **Flask**. Ne-am concentrat pe implementarea conceptelor de baza din Blockchain astfel incat sa ne creionam o idee cat mai clara despre cum functioneaza in spate cat si pentru a avea controlul deplin a ceea ce se intampla.

Usecase-ul concret pe care l-am folosit este un sistem de votare in cadrul facultatii pentru stabilirea celei mai populare materii, iar aceasta alegere sa fie realizata intr-un mod cat mai corect.

Structura proiectului

Componentele proiectului sunt urmatoarele:

- Blockchain -> clasa care defineste blockchain-ul in care se retin tranzactiile si in care se definesc comportamentele specifice
- Miner -> este privit ca o entitate separata, ce este o clasa derivata din Blockchain; Acesta ruleaza independent si proceseaza tranzactiile noi, adaugand astfel noi blocuri. De asemenea, aceasta comunica cu restul de mineri din retea pentru a se pune de comun acord asupra blockchain-ului si pentru a procesa tranzactiile.
- Voter -> este tot o entitate separata, ce reprezinta user-ul care realizeaza tranzactiile. Tinand cont de natura problemei (sistem de votare), fiecare user poate primi de la sistem maxim 1 token pe care il poate cheltui pe 1 vot (exprimand cine primeste votul).
- Admin -> este o entitate separata, un user special; Acesta poate efectua tranzactii in numele sistemului userilor, astfel incat aceastia sa poata primi token-ul corespunzator votului lor; Practic, adminul se ocupa de distribuirea dreptului de vot (pe baza unui CNP/Nr. Matricol, etc.); Aceasta entitate face legatura dintre sistemul de votare descentralizat si identitatea persoanelor (sistem centralizat).

In continuare vom prezenta fiecare componenta.

Blockchain

Atribute

- MINING_DIFFICULTY : Numarul de biti de 0 cu care sa inceapa hashul
- MINER_REWARD : cat primeste fiecare miner pentru un bloc nou
- blockchain : o lista de blocuri, fiecare bloc continand metadate si o lista de tranzactii

Metode

- verifyPOW() : verifica daca un block are proof of work-ul valid
- verifyBlockchain(blockchain) : verifica daca un blockchain este valid din toate punctele de vedere. Acesta este primit ca input la metoda
- getBlockchainAvailableFunds(blockchain) : verifica cate token-uri mai are fiecare user in urma tranzactiilor din blockchainul respectiv. Face balanata fiecarui user
- verifyTransactionSignature(sender, signature, transaction) : valideaza ca tranzactia respectiva a fost semnata de userul care a postat-o
- appendBlock(block) : appenduieste un block la blockchain, dupa ce il verifica corespunzator.

Miner

Atribute

- transaction_pool : lista de tranzactii care inca nu au fost adaugate pe blockchain
- friendlyMiners : lista cu adresa altor mineri, folosita pentru a comunica cu acestia

Metode

- PoW() : metoda ce are ca scop gasirii pow-ului unui block
- resolveConflicts() : metoda ce are ca scop rezolvarea diferentelor dintre blockchain-uri, pe baza principiului celui mai lung lant
- appendTransaction() : metoda ce are ca scop adaugarea unei noi tranzactii in tranzaction_pool dupa ce a fost verificata in prealabil
- mine() : metoda ce are ca scop minarea tranzactiilor din transaction_pool
- runMiner() : o metoda ce este apelata in mod repetat pentru a verifica daca au aparaut noi tranzactii ce trebuiesc procesate

Voter

Atribute

- private_key : cheia privata pentru semnarea tranzactiilor. Se foloseste algoritmul RSA.
- public_key : cheia publica ce tine loc si de identificator. Aceasta este transmisa tuturor si este folosita pentru a valida tranzactiile semnate.

Metode

- getAvailableFunds() : metoda ce returneaza cantitatea de token-uri pe care userul le mai are in urma tranzactiilor de pe blockchain
- createTransaction() : metoda ce creeaza si posteaza catre un miner o noua tranzactie.

Admin

Are aceleasi atribute ca voterul, singura diferenta fiind faptul ca public key-ul asociat este recunoscut de catre sistem, dandu-i astfel un drept de a face tranzactii catre useri.