Departamentul de Inginerie ȘI INFORMATICĂ, Universitatea spiru Haret

Disciplina/Laborator: Proiect informatic ÎN ECHIPĂ

Anul 3, Semigrupa 3042

sistem de votare bazat pe blockchain

studenți: Turbureanu Georgiana – Carmen

Voicu Ionel

# Overview

## Descrierea proiectului

|  |  |
| --- | --- |
|  | În prezenta lucrare s-a urmărit crearea unui sistem de votare bazat pe tehnologia Blockchain în limbajul JAVA. Totodată, am simulat și o zi de votare folosindu-ne de librăria Faker, generându-se astfel populația României. Iar pentru o mai bună vizualizare a unui scrutin în timp real, am creat un Dashboard în PowerBI unde datele pot fi vizualizate în timp real. |

## Scopul și obiectivele proiectului

|  |  |
| --- | --- |
|  | Scopul proiectului este să încercăm să criptăm datele cu ajutorul tehnologiei Blockchain pentru ca preferințele fiecărei persoane în parte să fie confidențiale, să fie stocate pe un server, unde accesul este limitat și să fie prezentate publicului într-o manieră prietenoasă. |

## Cerințe de nivel înalt

|  |  |
| --- | --- |
|  | Pentru a implementa anumite task-uri în programul NetBeans 12.0 IDE trebuie mai întâi să fie instalat de pe următorul link:<https://netbeans.apache.org/download/nb120/nb120.html> .  După ce am instalat NetBeans – ul de pe acest link eu, Turbureanu Georgiana – Carmen am creat un proiect numit „SistemDeVotareBazatPeBlockchain”format din 7 clase de Java. Acestea sunt: Block.java, ClientManager.java, Main.java, MesageStruct.java, NetworkManager.java, ServerHandler.java și ServerManager.java.  Clasele ClientManager, ServerHandler și ServerManager sunt bazate pe noțiunile învățate în acest semestru de la disciplina „Programare paralelă, concurentă și distribuită”.  În clasa Block.java am implementat o clasă interioară numită Vote care este alcătuită din variabilele: voterId, voterName și voteParty. Aici fiecare candidat are un id, un nume și la final se va realiza în funcție de rezultatul obținut după votare, un vot de petrecere. Variabila previousHash marchează hash-ul anterior și variabila blockHash marchează blocul de hash.  În clasa ClientManager.java am marcat canalul de comunicație care comunică cu serverul și pe baza a celor trei algoritmi: BJP, INC și BSP cu ajutorl cărora pot să adaug o criptare în block-ul generic. Dacă nu se poate conecta la server se va afișa un mesaj de eroare, în sensul că clientul „nu se poate conecta la server”. Există o funcție „startClient” care ne întâmpină cu un mesaj „Bine ați venit la mașina de vot!”. Pentru a continua procesul de votare trebuie să tastez pe y. După acest lucru, afișez un mesaj „Introduceți ID-ul alegătorului:”, apoi se afișează un mesaj „Introduceți numele alegătorului” și la final se afișează un mesaj „Vot pentru petreceri:”. După ce se introduce votul, îl rog pe votant printr-un mesaj să introducă corect indexul pentru fiecare candidat. Dacă nu respectă aceste instrucțiuni se va afișa un mesaj care indică faptul că votul este nevalid. Dacă dorește să pună alt vot se poate realiza acest lucru. Se creează un cifru folosindu-se de algoritmii: AES, ECB și PKCS5Padding. Există un obiect de decriptare folosindu-se de aceeași algoritmi. Este o funcție ce verifică validitatea block-ului. Este o funcție ce trimite un mesaj către clienți. După ce se realizează aceste instrucțiuni informațiile se stochează într-un folder numit „blockchain\_data” fiind situat pe Desktop.  În clasa Main.java am declarat adresa serverului care este „localhost” și portul acestuia este 6777. Prin mai multe mesaje am afișat meniul principal și prezint un anumit ajutor dacă anumiți votanți se încurcă la selectarea votului sau la utilizarea aplicației mele.  În clasa MesageStruct.java am realizat o structură pentru comunicarea între server și client. Tot aici prezint 3 tipuri de mesaje și descrierea acestora.  În clasa NetworkManager.java furnizez operațiunea de rețea. Am creat o funcție în care trimit un mesaj la canalul de comunicație. Prin funcția „receiveMsg” se încearcă să primească un mesaj de la socket.  În clasa ServerHandler.java am pregătit și am așteptat mesajele de la un client specificat cu by\_socket.  În clasa ServerManager.java sunt marcate toate comunicațiile de rețea din partea serverului. Sunt gestionate ID-urile clienților și se păstrează o hartă între ID-ul clientului și canalul comunicației unui client. Tot aici se pot conecta clienții noi. |
|  |  |

## Minuta fiecărei ședinte online și livrabile

|  |  |  |
| --- | --- | --- |
| Data minutei | Probleme / Minuta | Soluții de rezolvare |
| 15.03.2021 | - întâlnire inițială pentru stabilirea obiectivelor | N/A |
| 22.03.2021 | - PowerBI nu se poate conecta la baza Oracle | - se vor testa diferite soluții găsite pe internet  - dacă nicio solutie nu va funcționa vom trece la o bază PostgreSQL |
| 05.04.2021 | - scriptul Java are erori de compilare cand se ajunge la functia de decriptare  - s-a ales server-ul PostgreSQL pentru că s-a reusit conectarea printr-un Direct Query cu PowerBI | - se întreabă colegii pentru sugestii;  - fiind în prima faza a dezvoltării scriptului, vom aloca mai multe zile pentru debugging  - în ultima fază îl vom întreba pe dl profesor |
| 12.04.2021 | - progresul fiecăruia | N/A |
| 19.04.2021 | - problema din data de 05.04.2021 a fost rezolvată | N/A |
| 26.04.2021 | - un prim demo din partea amândurora | N/A |
| 11.05.2021 | - prezentarea primei versiuni  - completarea fișierului .docx | N/A |
| 19.05.2021 | - prezentarea versiunii finale | N/A |

|  |  |
| --- | --- |
| Data livrabilului | Cuprinsul livrabilului |
| 11.05.2021 | <https://github.com/VoicuIonel/blockchain_voting_day> |
| 19.05.2021 | <https://github.com/VoicuIonel/blockchain_voting_day> |

## Modalități de implementare

|  |  |
| --- | --- |
|  | Ca și modalitate de implementare, ne-am folosit de diagrama Gantt, diagramă ce poate fi văzută la punctul 7. |

## Plan Implementare

|  |  |
| --- | --- |
|  | Ne-am folosit de metodologia Agile, unde soluția inițială a evoluat ajungându-se să simulăm și o zi de votare, folosindu-ne de date fictive. Ne-am împărțit în două roluri, Scrum Master și echipa de dezvoltare. Duratele iterative de timp, numite și sprinturi, au fost din trei în trei săptămâni, ajungând în final să avem trei sprinturi.  Primul pas a fost să creăm o listă de task-uri pentru fiecare membru din echipă și să alegem tehnologiile în care se va lucra, după cum urmează:   * JAVA  sistem de votare bazat pe blockchain * Python * crearea tabelelor/view-urilor * generarea de date fictive și inserarea în baza PostgreSQL * simularea unei zi de votare prin inserarea datelor în bază pentru fiecare oră în parte * Microsoft PowerBI  crearea unui Dashboard * Microsoft Visio  generarea diagramei Gantt |

## Timeline și calendarul desfășurării activităților



## Echipa și rolurile fiecărui membru al echipei

|  |  |  |  |
| --- | --- | --- | --- |
| Student | Rol | Atribuții principale | Task-uri principale |
| Turbureanu Georgiana - Carmen | Scrum Master | Asignarea de task-uri | * setarea ședințelor săptămânale |
| Developer | Crearea blockchain-ului în Java | * aplicarea unui blockchain deja existent pentru proiectul nostru * criptarea/decriptare block hash-ului * crearea unui server cât și a unui client |
| Voicu Ionel | Data Architect | Configurarea unei baze de date | * instalarea și setarea unui server PostgresSQL * crearea de tabele/view-uri * generarea de date false pentru fiecare județ din România folosind librăria Faker * inserarea datelor inițiale |
| Simularea unei zi de votare | * crearea unui eșantion aleatoriu din baza de date * împărțirea eșantionului în mod aleator la 14, reprezentând numărul de ore în care populația are drept de vot |
| PowerBI Dashboard | * conectarea la Dashboard * crearea unui Dashboard în Microsoft PowerBI și testarea acestuia |
| Database Security | Limitarea accesului | * crearea de useri și limitarea drepturilor de acces * acces limitat la baza de date pentru o listă predefinită de IP-uri |

## Bibliografie

1. Cursul și laboratorul de la disciplina “Programare paralelă, concurentă și distribuită” a domnului Conf. univ. dr. Eugen Zăhărescu

2. https://github.com/hardik22317/Blockchain-based\_E-Voting/find/master