Zbornik radova Fakulteta tehničkih nauka, Novi Sad

UDK: 656.8
DOI: https://doi.org/10.24867/10BE09Malencic

PRIMENA ETHEREUM BLOKČEJN PLATFORME ZA RAZVOJ DECENTRALIZOVANE APLIKACIJE ZA GLASANJE

APPLICATION OF THE ETHEREUM BLOCKCHAIN PLATFORM FOR THE DEVELOPMENT OF A DECENTRALIZED VOTING SYSTEM

Nikola Malenčić, *Fakultet tehničkih nauka, Novi Sad*

Oblast – ELEKTROTEHNIKA I RAČUNARSTVO

Kratak sadržaj – *U ovom radu će biti predstavljene mogućnosti primene blokčejn tehnologije za podršku sistemima za glasanje i opisan razvoj jednog takvog sistema u vidu aplikacije na Ethereum blokčejn platformi. Aplikacija je implementirana na distribuiran i decentralizovan način kako bi se ispitale prednosti i mane softverskog rešenja ovog tipa.*
Ključne reči: *distribuirani sistemi, blokčejn, pametni ugovori.*
Abstract – *This work will present the possibilities for the application of blockchain technology in supporting voting systems and describe the development of such a system using the example of an Ethereum blockchain platform application. The application is implemented in a distributed and decentralized fashion, so as to explore the positive and negative aspects of a software solution of this type.*

Key words: *distributed systems, blockchain, smart contracts.*

1. UVOD

Tema ovog rada jeste implementacija softverskog rešenja koje predstavlja decentralizovanu aplikaciju za glasanje. Ideja je da se uporede prednosti i mane centralizovanih i decentralizovanih sistema kako bi se ustanovilo koji sistem više vredi razviti u različitim situacijama. U daljem tekstu uvoda će biti dat kratak pregled odnosa centralizovanih i decentralizovanih sistema. U drugom poglavlju će biti dat detaljan opis distribuiranih sistema kao celine. Treće poglavlje se bavi blokčejnom i tehnologijama distribuirane glavne knjige, dok je u četvrtom fokus na Ethereum-u i decentralizovanim aplikacijama [6]. U petom poglavlju su opisane korišćene tehnologije i alati, dok šesto i sedmo poglavlje predstavljaju opis rešenja i zaključak,

Centralizovani sistemi su tipično jednostavniji za implementaciju nego decentralizovani. Softver kod decentralizovanih sistema je veoma složen zbog brojnih problema konzistentnosti sistema. Decentralizovani sistem mora krajnjem korisniku da odaje utisak da intereaguje sa jednim koherentnim sistemom. Potrebni su protokoli koji će svakom članu sistema (čvoru) dati jedinstvenu spregu ka sistemu.

Takođe, u decentralizovanom okruženju, više čvorova mogu istovremeno da pristupaju jednom resursu, što znači da protokoli moraju da obezbede i konzistentnost podataka.

Decentralizovani sistemi se dobro skaliraju. Najrazličitiji uređaji koji implementiraju protokole sistema mogu lako da se pridruže, dok kod centralizovanih sistema to ne mora uvek biti slučaj. Uređaji koji učestvuju u sistemu mogu biti rasuti po celom svetu ali komuniciraju sa lokalnim uređajima kako ne bi bilo preteranog čekanja. Takođe, kod centralizovanih sistema povećanje broja korisnika može dovesti do toga da centralni serveri ne mogu da opsluže sve klijente, dok kod decentralizovanih sistema nema ovoga tipa problema.

Sa strane performansi pojedinačnih uređaja, kod decentralizovanih sistema se može javiti slabije iskorišćenje samih hardverskih jedinica. Ovo je zato što se tipično kod ovakvih sistema javlja mnogo komunikacije između čvorova koja nije vezana za samo rešavanje problema. Ova komunikacija služi kako bi se postigla konzistentnost i koherentnost sistema.

Sistemi sa centralizovanom arhitekturom su podložni otkazima. Kod ovakvih arhitektura je dovoljan mali kvar da paralizuje rad sistema. U distribuiranom okruženju sistem nastavlja da radi i ako neki čvorovi otkažu.

Bezbednost sistema je zanimljiva tema jer ne postoji očigledna prednost centralizovanog ili decentralizovanog načina rada.

S jedne strane, centralizovane sisteme je lakše osigurati jer je potrebno voditi računa samo o jednom kritičnom delu sistema, dok je kod drugih potrebno voditi računa o ponašanju svih čvorova sistema. Sa najnovijim protokolima neka decentralizovana rešenja uspevaju da nude veliku bezbednost, po nekim čak i bolju nego što nude centralizovani sistemi.

Jedna od najaktuelnijih tema u modernom dobu je privatnost. Popularnost decentralizovanih rešenja u velikoj meri proističe kao reakcija na monopol i neetičke postupke velikih tehnoloških kompanija u 21. veku. Ljudi gube poverenje u ove kompanije jer smatraju da im podaci nisu adekvatno zaštićeni i da se ti podaci koriste protiv njihovih interesa.

Zbog toga na svetsku scenu nastupaju velike platforme koje su bazirane na distribuiranim tehnologijama koje garantuju sigurnost i privatnost svih korisnika koji deluju na njima. Dakle, može se reći da decentralizovani sistemi nude veću zaštitu ličnih podataka nego centralizovani sistemi.

NAPOMENA:
Ovaj rad proistekao je iz master rada čiji mentor je bio dr Dušan Gajić, docent.

2. DISTRIBUIRANI SISTEMI

Distribuirani sistem je kolekcija autonomnih računskih elemenata koja se svojim korisnicima prikazuje kao jedan koherentni sistem [2]. Svaki pojedinačni računski element se naziva čvor. Čvorovi mogu biti ili hardverske jedinice ili softverski procesi. Moderni distribuirani sistemi se često sastoje od najrazličitijih vrsta čvorova, od računara visokih performansi do najsitnijih senzor uređaja. Osnovni princip je da su čvorovi u mogućnosti da deluju nezavisno, ali ukoliko se oni ignorišu međusobno onda nema smisla da se postavljaju u isti distribuirani sistem. U praksi su čvorovi isprogramirani da sarađuju jedni sa drugima što se realizuje razmenjivanjem poruke ili sa nekom vrstom deljene memorije.

Važno je primetiti da se kao posledica međusobnog delovanja raznih nezavisnih čvorova javlja odsustvo zajedničke vremenske reference. Naime, ne može se pretpostaviti da postoji nešto kao što je globalni sat. Ovaj nedostatak zajedničkog vremena dovodi do velikih problema što se tiče sinhronizacije i koordinacije između čvorova. Distribuirani sistem bi trebalo da se prikazuje korisnicima kao jedinstven i koherentan sistem. Neki kažu da krajnji korisnici uopšte ne bi trebalo da primećuju da rade sa procesima i podacima koji su rasprostranjeni po čitavoj distribuiranoj mreži. Ovo se uglavnom smatra preterivanjem, i neki stabilni konsenzus je na tome da bi sistem trebao da se ponaša onako kako korisnici očekuju da se ponaša. Kolekcija čvorova bi trebala da radi na isti način kakva god bila interakcija između korisnika i sistema. Distribuirani sistemi su uglavnom organizovani tako da imaju poseban softverski sloj koji je logički postavljen iznad operativnog sistema pojedinačnog računara koji je deo sistema. Ovaj sloj se popularno naziva middleware. Na neki način, middleware je distribuiranom sistemu isto što i operativni sistem jednom računaru. To je softverski sloj koji deluje kao upravljač resursima koji nudi svoje aplikacije kako bi se ti resursi efikasno podelili i iskoristili u mreži. Pored upravljanja resursima, može da nudi i druge servise koji su tipični za operativne sisteme. Ovo može uključivati servise za bezbednost, servise za kontrolu naloga, servise za oporavak od otkaza, servise za međuaplikacijsku komunikaciju itd. Razlika između ovih servisa i servisa operativnog sistema jeste to što se ovi servisi nude u mrežnom okruženju.

3. TEHNOLOGIJA DISTRIBUIRANE GLAVNE KNJIGE I BLOKČEJN

Distribuirana baza podataka je vrsta baze podataka kod koje se podaci čuvaju na više čvorova, tj. distribuirani su na više lokacija. Glavna knjiga (eng. Ledger) je knjiga koja služi za beleženje i sumiranje ekonomskih transakcija, sa debitima i kreditima u posebnim kolonama, takođe sa početnim i krajnjim stanjem dva računa. Dakle, distribuirana glavna knjiga, ili tehnologija distribuirane glavne knjige (eng. Distributed Ledger Technology – DLT) je vrsta distribuirane baze podataka u kojoj se pretpostavlja prisustvo malicioznih korisnika tj. čvorova.

Postoje tri glavne karakteristike tehnologija distribuiranih glavnih knjiga:

Jezik transakcija: Mora da postoji određeni format po kome će čvorovi da traže promenu stanja glavne knjige, tj. formalan način na koji će inicirati transakcije

Protokol: Moraju postojati pravila po kojima će se među učesnicima u distribuiranom sistemu postići konsenzus o tome koje će se transakcije prihvatiti i u kom redosledu će se upisati u glavnu knjigu

Model podataka: Trenutno stanje glavne knjige mora biti definisano određenim modelom podataka koji je odabran u vreme tehničke izvedbe sistema

Koncept tehnologija distribuirane glavne knjige postoji već dugo. Još 1982. godine su Lamport, Šostak i Pis opisali "problem Vizantijskih generala" i postavili pitanje kako će se računarski sistem nositi sa suprotstavljenim informacijama u neprijateljskom okruženju. 1999. godine istraživanja dovode do prvog algoritma koji na praktičan način rešava prethodno pomenuti problem. Ova tehnologija je osnova za novu generaciju aplikacija na bazi transakcija koje uspostavljaju transparentnost, odgovornost i poverenje u računarskim sistemima, a pri tom racionalizuju poslovne procese i pravna ograničenja kroz automatizaciju.

Blokčejn predstavlja jednu vrstu distribuirane baze podataka u kojoj se skladište digitalne transakcije. Ova tehnologija predstavlja podskup tehnologija distribuirane glavne knjige. Sa pojavom Bitcoin-a je blokčejn doveo do revolucije u distribuiranim sistemima i načinu na koju ljudi gledaju finansijsko poslovanje, lance isporuke, privatnost itd.

Postoje dve osnovne ideje koje razlikuju blokčejn od ostalih tehnologija distribuirane glavne knjige.

Prva je ideja da blokčejn bude projektovan tako da bi se postizao pouzdan i konzistentan dogovor između nezavisnih čvorova o zapisu događaja. Ovo se postiže mehanizmom konsenzusa koji omogućava da za svakog učesnika u mreži bude isti pogled na deljenju bazu podataka [4]. Dakle, učesnici u blokčejn mreži postižu konsenzus o promenama u deljenoj bazi podataka bez potrebe da se proverava integritet bilo kog čvora u mreži.

Druga ideja je specifična struktura podataka koja omogućava ovoj tehnologiji da reši problem dvostruke potrošnje, tj. onemogućava korisnicima da isti digitalni fajl kopiraju i prenose više puta. Blokčejn se iz ovih razloga može koristiti za razmenu valute, vrednosti ili drugih podataka centralnim bez potrebe za autoritetom.

Osnovna podela blokčejnova je na javne i privatne. Javnim sistemima može da pristupi bilo ko, dok privatnim mogu da pristupe samo autorizovani korisnici. Takođe je bitno uvideti da mogu postojati razne kontrole pristupa. Tako na nekim blokčejn sistemima može bilo ko i da čita i da piše na glavnu knjigu (Bitcoin, Ethereum), na nekima bilo ko može da čita, a samo autorizovani učesnici mogu da pišu (Sovrin), a na nekima postoji kontrola pristupa i za operacije čitanja i pisanja. Blokčejn koristi razne kriptografske tehnike za različite svrhe.

Prvenstveno, glavna knjiga u blokčejn sistema predstavlja lanac kriptografski povezanih blokova. Blok je skup određenog broja transakcija koje se istovremeno dodaju u lanac. U zaglavlju svakog bloka se nalazi referenca na prethodni blok u vidu kriptografskog heša tj. rezultata heš funkcije. Heš funkcije su jednosmerne (računanje inverzne funkcije je matematički zahtevno) matematičke funkcije koje na osnovu nekog skupa podataka daju heš, tj. vrednost fiksne dužine koja je dobijena na osnovu tog skupa podataka. Svrha ovoga je da se onemogući izmena blokova nakon dodavanja blokova u lanac. Ukoliko neki

učesnik pokuša da izmeni neki blok, ostali učesnici će to lako primetiti jer se heš tog izmenjenog bloka neće poklapati sa hešom koji je sadržan u zaglavlju novog bloka. Osim kriptografskog heša se u zaglavlju svakog bloka nalazi i koren Merkleovog stabla. Merkleovo stablo ili heš stablo je stuktura podataka koje predstavlja stablo kome je svaki terminalni čvor označen hešom bloka podataka, a svaki neterminalni čvor je označen hešom oznaka njegovih potomaka. Ovakva struktura omogućava pouzdanu i efikasnu pretragu i verifikaciju velikih struktura podataka.

Svako zaglavlje takođe sadrži vremenski otisak i nonce vrednost. Vremenski otisak predstavlja trenutak u vremenu kada je blok dodat u blokčejn. Nonce vrednost je slučajan broj koji se koristi samo jedanput, i ima ulogu u procesu validacije blokova.

U blokčejnu se takođe koristi infrastruktura javnih ključeva. Na blokčejnu postoje javne adrese kojima su pridružene određeni resursi (u primeru Bitcoin-a bi to bila neka količina Bitcoin-ova). Ideja je da svaki korisnik ima svoj privatan ključ, na osnovu kojeg on može da generiše javne ključeve ili adrese. Uz pomoć ovog privatnog ključa, korisnik može da dokaže posedovanje neke javne adrese, tj. adrese koja je izvedena iz tog ključa. Dakle, svi resursi se nalaze u glavnoj knjizi, korisnici putem svojih ključeva pristupaju glavnoj knjizi i dobijaju mogućnost da manipulišu resursima koji su pridruženi njihovim javnim adresama. Postizanje dogovora oko novog stanja glavne knjige je jedan od najvećih problema distribuiranih sistema. Konsenzus je proces postizanja dogovora između svih čvorova u mreži o ispravnom stanju podataka i cilj je da svi čvorovi dele iste podatke. Ovo se postiže konsenzus algoritmima. Ovi algoritmi osiguravaju da su upisani podaci u glavnu knjigu isti za sve čvorove u mreži i tako sprečavaju maliciozne učesnike da manipulišu podacima.

Osvrnućemo se na neke najpopularnije konsenzus algoritme.

Dokaz posla (eng. Proof of work) je konsenzus algoritma baziran na lutriji u kome se podrazumeva rešavanje matematičke zagonetke koja je zahtevna za izračunavanje kako bi se kreirao novi blok u sistemu. Najpoznatiji sistem baziran na dokazu posla je heškeš (Adam Bek 1997.) i služi za limitiranje spama i sprečavanje denial-of-service napada. Jedini način da se pronađe rešenje matematičke zagonetke su brute-force algoritmi koji manje više pogađaju tačno rešenje. Provera rešenja je jednostavna i ne zahteva kompleksna izračunavanja. Proces izračunavanja se naziva rudarenjem, a čvorovi u mreži koji pokušavaju da dođu do rešenja se nazivaju rudari. Rudari imaju ekonomski podsticaj jer pronalazak heša novog bloka dovodi do stvaranje novih resursa koji pripadaju rudaru (u nekim blokčejn sistema) i dobijaju proviziju od transakcija koje su validirane u tom bloku.

Dokaz uloga (eng. Proof of stake) je konsenzus algoritam koji je takođe baziran na lutriji i predstavlja generalizaciju dokaza posla. U ovom slučaju su čvorovi poznati kao validatori i oni validiraju transakcije kako bi zaradili proviziju od istih (ovde nema stvaranja novog resursa). Čvorovi se slučajno biraju za validatore blokova, a ova verovatnoća zavisi od veličine uloga nekog čvora. Ukoliko čvor A poseduje dva resursa, a čvor B jedan resurs, čvor A ima veću šansu da bude postavljen kao validator. Ključni aspekt je slučajnost, kako bi se izbegla situacija gde najbogatiji čvorovi stalno dobijaju ulogu validatora.

4. ETHEREUM I DECENTRALIZOVANE APLIKACIJE

Ethereum je jedna od najpoznatijih platformi baziranih na blokčejn tehnologiji. Ova platforma je otvorenog koda, javna i omogućuje razvoj i rad raznih projekata koji žele da iskoriste prednosti decentralizacije. Za razliku od Bitcoin-a, što je platforma koja isključivo služi za razmenu digitalne valute, na Ethereum-u mogu da se implementiraju najrazličitije funkcionalnosti.

Rane blokčejn aplikacije kao što su Bitcoin su jedino dozvoljavale korisnicima ograničen skup predefinisanih operacija. Za razliku od ovakvih projekata, Ethereum nudi korisnicima da prave svoje operacije. Ovo nas dovodi do koncepta Ethereum Virtualne Mašine (EVM). EVM je Ethereumovo okruženje za izvršavanje, i služi za izvršavanje pametnih ugovora. Pošto svaki Ethereum čvor izvršava Ethereum Virtualnu Mašinu, aplikacije izgrađene na ovoj bazi dobijaju pogodnosti decentralizacije bez potrebe da koriste neki posebno dizajniran blokčejn.

Slično kao i u Bitcoin-u, Ethereum ima svoj koncept rudarenja i svoju kriptovalutu Etar. Ethereum trenutno koristi dokaz posla kao konsenzus algoritam (kao i Bitcoin), ali je u planu prelazak na dokaz uloga. Do tada Etar stvara platforma i isplaćuje ga rudarima koji izračunaju heš novog bloka transakcija.

Pametni ugovori su programski kodovi koji na visokom nivou opisuju interakcije između korisnika i distribuirane glavne knjige. Ovi kodovi mogu da se kompajliraju u EVM bajtkod i da se izvršavaju na Ethereum Virtualnoj Mašini. Kao što Bitcoin ima svoj BitcoinScript, tako i Ethereum ima svoj jezik za pisanje pametnih ugovora, Solidity.

Razlika između ova dva je to što je Solidity Tjuring kompletan i nudi mnogo više mogućnosti, dok BitcoinScript podržava samo finansijske transakcije.

Pametni ugovori se postavljaju na Ethereum blokčejn gde dobijaju svoje javne adrese. Ovo zapravo znači da će kod ovih pametnih ugovora izvršavati mašine-čvorovi od rudara. Korisnici mogu pristupati ovom ugovoru putem te javne adrese i vršiti interakciju sa njim. Pojedinosti interakcije zavise od funkcionalnosti samog pametnog ugovora.

Na primer, Alisa i Bob žele da se opklade u 100 etra u ishod fudbalske utakmice. Oni se dogovore oko poverljive veb stranice za rezultate, i svako pošalje po 100 etra na javnu adresu pametnog ugovora. Nakon što se utakmica završi, pametni ugovor će proveriti rezultat utakmice na osnovu poverljive veb stranice. U zavisnosti od rezultata, pametni ugovor će automatski isplatiti 200 etra pobedniku opklade. Originalna zamisao Ethereuma je da on deluje kao „svetski računar“. Čvorovi mreže bi svi imali EVM kao midlver sloj, koji bi davao utisak da su svi računari deo jednog koherentnog sistema.

Tradicionalna veb aplikacija bi imala tri dela: frontend, bekend i bazu podataka. Kod decentralizovanih aplikacija postoji sličan pristup.

Frontend deo, koji predstavlja prednji deo aplikacije sa kojim korisnik intereaguje, može biti hostovan na nekom

centralnom serveru, a može biti i hostovan na nekim decentralizovanim platformama za skladištenje kao što su Svarm. Ovako bi se uspostavila potpuna decentralizacija.
Bekend deo, tj. serverski deo koji predstavlja logiku i spoj između podataka i frontenda, bi se implementirao kao pametni ugovor na nekoj blokčejn mreži.
Ovako se taj bekend ne bi izvršavao na nekom centralizovanom serveru, već bi ga izvršavali svi čvorovi mreže na potpuno distribuiran način.
Baza podataka bi u ovom slučaju bila sama distribuirana glavna knjiga blokčejn sistema. Zamisao osnivača Ethereum-a, Vitalik Buterina, je od početka bila da Ethereum ima ulogu svetskog računara gde čvorovi izvršavaju decentralizovane aplikacije. Ideja je da decentralizacija nudi brojne prednosti u finansijskoj, proizvodnoj, zdravstvenoj i informativnoj industriji.

5. KORIŠĆENE TEHNOLOGIJE I ALATI

Kao podrška razvoju okruženja za bavljenje blokčejn sistemom je iskorišćen Truffle Suite [5]. Truffle Suite obuhvata nekoliko alata za olakšan rad sa sistemima na bazi Ethereuma.
Ganache je personalni blokčejn namenjen za brzi razvoj Ethereum i Corda distribuiranih aplikacija. Može se koristiti tokom celog ciklusa razvoja, omogućavajući razvoj, pokretanje i testiranje decentralizovanih aplikacija u bezbednom i determinističkom okruženju. Ganache ima svoju desktop aplikaciju koja je iskorišćena u sklopu ovog rada da se simulira ponašanje Ethereum blokčejna. Razvijati decentralizovane aplikacije na primarnom Ethereum blokčejnu nije praktično iz više razloga.
Prvenstveno, transakcije na Ethereum-u koštaju. Zvog ovoga Ethereum ima svoju test mrežu, gde Ether nema vrednost, ali ni ona nije praktična za korišćenje. Ovo je zbog toga što je računarski zahtevno biti punopravni čvor na Ethereum mreži. Zato je korišćen Ganache, koji praktično simulira ponašanje pravog Ethereum blokčejn sistema.
Truffle je okruženje za rad sa blokčejnovima koji rade u sklopu Ethereum virtuelne mašine bazirano na NodeJS. NodeJS je okruženje otvorenog koda za izvršavanje JavaScript koda van pretraživača. Truffle olakšava razvoj i testiranje Ethereum aplikacija.
Neke stvari koje Truffle nudi su: ugrađeni mehanizmi za kompajliranje pametnih ugovora i linkovanje binarnih datoteka, mogućnost pisanja automatskih testova za brzi razvoj, pisanje skripti za podešavanje ugovora i migracija, upravljanje sa stanovišta mreža, upravljanje paketima i bibliotekama, interaktivna konzola.
Truffle se skida i instalira putem NPM-a (Node Packet Manager). Nakon podešavanja samog okvira, možemo započeti korišćenje Truffle-a.
Web3.js je JavaScript biblioteka koja omogućava rad sa Ethereum čvorovima [7]. Svaki Ethereum čvor ima RPC API i metode iz ovog API-ja je moguće pozivati putem metoda iz Web3.js biblioteke.
MetaMask je novčanik za kriptovalute i sprega pretraživača i blokčejna. Dolazi u formi plagina za pretraživač (Mozilla, Chrome..). Omogućava pametno upravljanje Ethereum adresama i ključevima iz pretraživača. Pomoću MetaMask-a, frontend aplikacija može da prosleđuje relevantne korisničke podatke ka blokčejnu.

6. REŠENJE

Predstavljeno rešenje sastoji se iz tri dela: distribuirane baze podataka, bekend aplikacije, i frontend aplikacije. Ulogu distribuirane baze podataka vrši sama Ethereum blokčejn platforma. Ovoj bazi pristupa bekend aplikacija, u vidu pametnog ugovora koji reguliše transakcije iz i ka blokčejnu. Frontend aplikacija, napisana u JavaScript-u, komunicira sa bekend aplikacijom u cilju ispravnog prikazivanja relevantnih podataka korisniku.
Aplikacija podržava sledeće mogućnosti: pravljenje novih decentralizovanih izbora, dodavanje novih kandidata na izbore, dodavanje novih učesnika (glasača) na izbore, glasanje na izborima, prikazivanje rezultata izbora. Trenutna verzija aplikacije omogućava isključivo deljenje informacija o izborima relevantnim korisnicima proverama javnih adresa korisnika. Ovo ograničenje se može prevazići u budućnosti dodavanjem servisa za članstvo (eng. membership service). Ovako bi korisnici mogli da učestvuju na više izbora istovremeno, a i mogli bi da prate rezultate drugih izbora na kojima ne učestvuju, ukoliko imaju dozvolu za to.

7. ZAKLJUČAK

Prednost ove decentralizovane aplikacije se ogleda u tome da pruža mnogo veću zaštitu nego tradicionalne centralizovane aplikacije. Ovakav pristup omogućava bezbedno i pouzdano deljenje informacija među korisnicima. Maliciozni učesnici koji imaju za cilj da kompromituju neke izbore imaju značajno redukovane mogućnosti u ovakvom decentralizovanom sistemu. S druge strane, ovakav sistem je mnogo zahtevniji za implementaciju, kao i za održavanje, od tradicionalnih centralizovanih sistema.

LITERATURA

[1] Dr Dušan Gajić, *Materijali sa predmeta Paralelni i distribuirani algoritmi i strukture podataka*, dostupno na: http://www.acs.uns.ac.rs/sr/node/237/4468699, poslednji pristup jul 2020.
[2] Maarten van Steen, Andrew S. Tanenbaum., *Distributed Systems (third edition)*, Maarten van Steen 2018.
[3] Don Tapscott, *Blockchain Revolution*, Penguin Random House LLC 2016.
[4] *Blockchain Consesnus*, dostupno na: *https://devopedia.org/blockchain-consensus* , poslednji pristup mart 2020.
[5] *Truffle* , dostupno na: www.trufflesuite.com, poslednji pristup jul 2020.
[6] *Ethereum*, dostupno na: https://ethereum.org/en/, poslednji pristup jul 2020.
[7] *Web3js*, dostupno na: https://web3js.readthedocs.io/en/v1.2.11/, poslednji pristup jul 2020.
[8] *Metamask*, dostupno na: https://metamask.io/, poslednji pristup jul 2020.

Kratka biografija:

Nikola Malenčić rođen je u Novom Sadu, Republika Srbija, 11. novembra 1996. godine. Osnovne akademske studije je upisao na Fakultetu tehničkih nauka Univerziteta u Novom Sadu 2015. godine. Diplomirao je 2019. godine.