SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

ZAVRŠNI RAD br. 5753

Sustav za pohranu DNA

Matej Šaravanja

Zagreb, lipanj 2018.

SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA
ODBOR ZA ZAVRŠNI RAD MODULA

Zagreb, 15. ožujka 2018.

ZAVRŠNI ZADATAK br. 5753

Pristupnik: **Matej Šaravanja (0036487164)**
Studij: Računarstvo
Modul: Računarska znanost

Zadatak: **Sustav za pohranu DNA**

Opis zadatka:

Cijena sekvenciranja genoma opada puno brže od Mooreovog zakona i trenutno se ljudski genom može sekvencirati za manje od 1000 $. Zbog toga se mnogi odlučuju sekvencirati svoj genom. Vrlo je važno taj genom sigurno pohraniti na što više mjesta i istovremeno osigurati da mu može pristupiti samo korisnik koji je genom pohranio. U završnom radu potrebno je razviti sustav za raspodijeljenu pohranu genoma koristeći blockchain tehnologiju.
Programski kod je potrebno komentirati i pri pisanju pratiti neki od standardnih stilova. Kompletnu aplikaciju postaviti na repozitorij Github.

Zadatak uručen pristupniku: 16. ožujka 2018.
Rok za predaju rada: 15. lipnja 2018.

Mentor:
Izv. prof. dr. sc. Mile Šikić

Djelovođa:
Doc. dr. sc. Tomislav Hrkać

Predsjednik odbora za završni rad modula:
Prof. dr. sc. Siniša Srbljić

Hvala mentoru izv.prof.dr.sc. Mili Šikiću, obitelji i prijateljima na pruženoj podršci i motivaciji za pisanje ovog Završnog rada.

SADRŽAJ

1. Uvod

Cijena sekvenciranja genoma opada puno brže od Mooreovog zakona i trenutno se ljudski genom može sekvencirati za manje od 1000 američkih dolara. Zbog toga se mnogi odlučuju sekvencirati svoj genom. Kako navodi Patrick Lin, autor teksta na Forbesu: "Ispitivanje genoma najavljuje novo doba medicine. Kako se približavamo cijeni od 100 dolara, pacijenti se mogu liječiti s mnogo većom preciznošću, skrojenom prema njihovom osobnom genetskom kodu." Dalje u navodu stoji da "najveća zabrinutost kod sekvenciranja genoma proizlazi iz problema sigurnosti i zaštite korisničkih podataka" [1]. Naime, zakonodavci u SAD-u žele omogućiti pristup podatcima testiranja poslodavcima, čime u pitanje dolazi omogućavanje prava zaposlenicima na zdravstvenu zaštitu [2].

Omogućavanje pristupa podatcima poslodavcima još je jedan od problema s kojim se suočavaju oni koji se odluče na sekvenciranje genoma jer zakoni ne poznaju termin vlasnika genomskih podataka. Sudeći prema svemu navedenom, jedini način zadržavanja kontrole nad osobnim genetskim podatcima je - ne testirati se, što poništava cijelu poantu najsuvremenije medicinske genomske tehnologije. [1]

U ovom radu testirat će se hipoteza da je navedeni problem sigurnog načina pohrane osjetljivih podataka poput ljudskog genoma te zaštite privatnosti donora genoma rješiv u vidu hibridnog sustava korištenjem distribuirane i decentralizirane tehnologije *blockchain* i centralizirane baze podataka.

U poglavlju 2 dodatno će se ući u srž problema i upoznati s razmišljanjima ljudi koji su adresirali problem. Zatim, u trećem poglavlju, će se predstaviti korištene tehnologije i implementacija prijedloga rješenja. U četvrtom poglavlju komentirat će se rezultati rješenja, a u petom moguća poboljšanja implementiranog sustava. U završnom poglavlju stajat će kratak osvrt na rad.

2. Problem

U suvremenoj medicini postoji velika potreba za sekvenciranjem i obradom genoma. Rezultati istraživanja provedenih nad određenim genomom donose obostranu korist i znanstvenoj zajednici i samom vlasniku genoma. Ljudski DNA sadrži informacije o očekivanoj životnoj dobi, sklonosti depresiji i šizofreniji, etničkom podrijetlu, očekivanom kvocijentu inteligencije, pa čak i političke sklonosti [1]. Poznavanjem tih informacija ljudi bi efikasnije mogli predvidjeti i otkriti potencijalne zdravstvene probleme te se fokusirati na prevenciju istih prije nego što uopće dođe do komplikacija. Time se može produžiti životni vijek čovjeka, popustiti pritisak na zdravstveni sustav, a u konačnici i obogatiti proračun zdravstva jer prevencija je uglavnom povoljnija od liječenja. Međutim, postoji i druga strana medalje koja primjerice, jednom kad je genom sekvenciran i rezultati obrade su poznati, omogućuje osiguravateljskim kućama odbijanje zahtjeva klijenta za zdravstveno osiguranje jer je iz rezultata obrade vidljivo da osoba ima veliku mogućnost obolijevanja od raka dojke [1].

Sve navedeno ukazuje na potrebu zaštite genoma, odnosno zaštitu poveznice između osobe i genoma. Suvremena medicina treba bazu genoma nad kojom može vršiti ispitivanje, a ljudi bi željeli znati više o sebi. U posljednje vrijeme pojavljuju se i ideje o uvođenju blockchain tehnologije u svrhu zaštite zdravstvenih podataka.

U tehničkom opisu pod nazivom *Blockchain za podatke u zdravstvu: Pogled iz perspektive podataka* (engl. *Blockchain for healthcare records: A data perspective*), autori Mian Zhang i Yuhong Ji raspravljaju o problemu zaštite evidencije zdravstvenih podataka pojedinca i kao glavne probleme adresiraju privatnost i integritet podataka. Navode da je jedno od rješenja i čuvanje svih podataka na blockchainu [3]. Uzmemo li u obzir da jedan ljudski genom sadrži otprilike 3 milijarde baznih parova[1] i da svaki par zauzima 2 bita, dolazimo do iznosa od 6 milijardi bitova, što je približno 720MB [4]. Ukoliko znamo je da je krajem 2017. godine danas najpoznatiji blockchain, onaj

[1]Za više informacija o genomu i baznim pročitajte u poglavlju 4.3.

Bitcoinov, iznosio otprilike 150GB [5], jasno je da je sustav o kojem na počcetku tehničkog opisa govore Zhang i Ji, gotovo nemoguće ostvariti jer bi mu trebalo samo 208 genoma da dostigne 150GB, s tim da bi imao i mnogo veću prosječnu godišnju stopu rasta od Bitcoinovog blockchaina koja iznosi 20GB (otprilike 27 genoma). Dalje u opisu predlažu korištenje klasične centralne baze podataka u kojoj bi se čuvali podatci, a sažetak (engl. *hash*), kao referenca na podatak, u blockchainu.

Blockchain kao moguće rješenje problema osiguravanja zaštite genome također predlaže i filozof David Koepsell u razgovoru s Patrickom Linom [1].

Uzevši u obzir sve navedene probleme, mogu se istaknuti 3 najznačajnija:

1. Postoji potreba za sekvenciranjem genoma kako bi suvremena osobna medicina što brže napredovala i postizala značajnije rezultate.
2. Zaštita privatnosti donora genoma - osoba ne želi da itko ima pristup njezinim podatcima koji je mogu direktno identificirati kao vlasnika genoma.
3. Pohrana i osiguranje nepromjenjivosti pohranjenog genoma.

3. Rješenje

3.1. Uvod

Pristupio sam rješavanju problema dijeljenja genoma iz dvije različite perspektive.

1. Prva je ona donora genoma. Ukoliko sam ja sekvencirao svoj genom i želim pomoći znanstvenoj zajednici u daljnjem razvoju, želim to učiniti na jednostavan i siguran način, a bilo bi dobro doznati nešto novo o sebi te imati i mogućnost zarade u cijelom procesu.

2. Iz druge perspektive, ukoliko sam ja organizacija koja se bavi proučavanjem genoma i otkrivanjem međusobnih različitosti na temelju kojih donosim zaključke istraživanja, želim imati na jednom mjestu jednostavan pristup velikoj količini podataka i dnevniku istraživanja kojemu možemo pristupiti i donor genoma i ja kao organizacija. Također ne želim da itko može urediti ili lažirati moje istraživanje provedeno nad određenom sekvencom genoma čime bi donor dobio netočno izvješće, a moj rad bi bio kompromitiran.

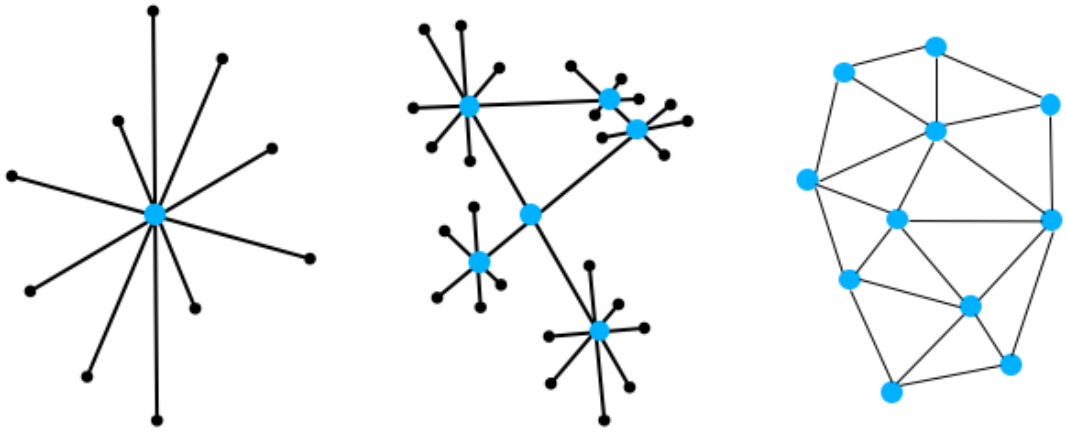
Poznavajući sve probleme i sve aktore procesa, odlučio sam se na hibridni model koji spaja klasičnu, centraliziranu bazu podataka u kojoj će podatci biti spremljeni, blockchain koji osigurava nepromjenjivost i raspodijeljenost unesenih podataka, te jednostavnog korisničkog sučelja koji može razumjeti svaki prosječan korisnik računala.

3.2. Korištene tehnologije

U sljedećim podnaslovima govorit ću o tehnologijama korištenim prilikom implementacije rješenja.

3.2.1. Blockchain

Blockchain ili prevedeno na hrvatski, lanac blokova jednosmjerni je lanac podatkovnih blokova u kojem svaka nova karika, odnosno blok, ovisi o vrijednosti prve starije karike i ima poveznicu na nju. Povezivanje blokova u lanac temeljeno je na kriptografiji kako bi se osigurala sigurnost i što veća razina privatnosti [6], [7]. Blockchain predstavlja jedno od rješenja distribuirane knjige zapisa, u kojoj ne postoji centralna kontrola kao u, primjerice, klasičnom bankarskom sustavu. Ilustracija koja prikazuje razliku između blockchaina i centraliziranog sustava nalazi se na slici 3.1.



Slika 3.1: Lijevo se nalazi prikaz centraliziranog sustava poput bankarskog. U sredini je primjer decentraliziranog sustava poput recimo blagajni. Desno se nalazi primjer distribuirane knjige zapisa u kojoj ne postoji centralni autoritet.

U računalnom smislu, blockchain je obična jednosmjerna povezana lista čiji su elementi blokovi podataka koji mogu biti predstavljeni u primjerice JSON[1] obliku i mogu sadržavati bilo koje informacije od kojih su dvije ključne: sažetak prethodnog bloka i *timestamp*[2]. Spomenuti sažetak bloka se računa na osnovu sažetka prethodnog bloka, podataka unutar bloka i *timestampa*. Blokove možemo zamisliti kao papir na kojem se nalazi popis podataka/transakcija, a blockchain kao fascikl u kojem su papiri složeni prema vremenu nastajanja [7]. Ono što osigurava nepromjenjivost unesenih podataka je raspodijeljenost sustava i međusobni konsenzus svih računala u mreži.

Blockchain mrežu čine korisnici koji dodaju podatke u blokove i tzv. rudari koji razvijaju mrežu. Svaki rudar, tj. računalo svakog rudara u mreži ima vlastitu kopiju

[1] JavaScript Object Notation: object = {key: "value"}

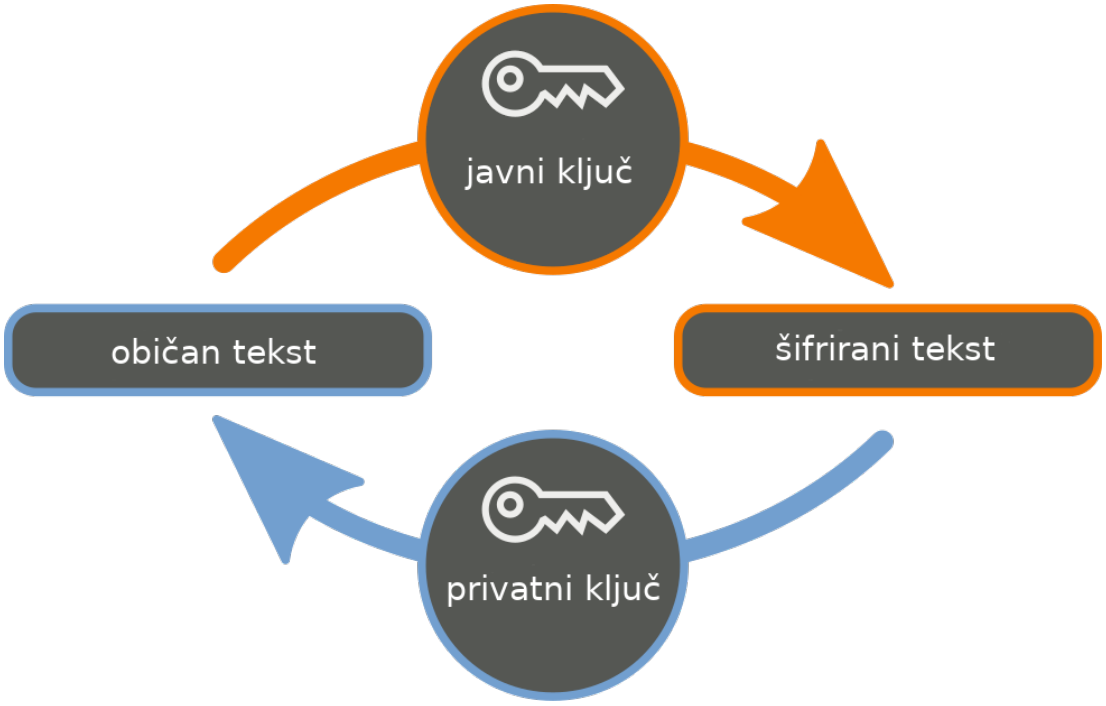
[2] Vrijeme u milisekundama proteklo od 1.1.1970. do određenog datuma

blockchaina (vlastitu kopiju fascikla). Prilikom dodavanja podatka u blockchain, korisnik odašilje svim rudarima u mreži poruku da želi dodati novi podatak. Taj podatak se provjerava (primjerice ukoliko je podatak transakcija, provjerava se ima li pošiljatelj dovoljno novca) i sprema u listu s ostalim podatcima budućeg bloka. U međuvremenu rudari "rudare" novi blok. Ono što oni zapravo rade jest rješavanje vremenski složenih matematičkih operacija, tzv. *"proof-of-work"* (recimo pogađanje sažetka koji počinje s deset nula) koji troše struju i opterećuju procesor računala rudara. Prvi rudar koji izrudari novi blok, tj. otkrije rješenje zadatka, odašilje ostalim rudarima u mreži da je uspješno dodao novi blok u mrežu, a ostali rudari verificiraju novi blok. Ukoliko više od 50% ostalih rudara verificira ispravnost odaslanog lanca s novim blokom, taj lanac postaje referentan i jedini ispravan u mreži (konsenzus (engl. *consensus algorithm*)). U tom slučaju rudar koji je izrudario novi blok biva nagrađen od sustava (recimo jednim novčićem). Međutim, ukoliko je rudar zlonamjeran i promijeni podatak u nekom od prethodnih blokova (recimo, napravi nekoliko lažnih transakcija) time automatski mijenja oznaku (sažetak) tog bloka i ostali rudari čim naiđu na nekonzistentnost među blokovima odbacuju cijeli taj lanac i proglašavaju ga nevažećim.

3.2.2. RSA i AES

RSA[3] u kriptografiji predstavlja ime za algoritam šifre javnog ključa. Prvi je algoritam koji se koristio za potpisivanje (utvrđivanje izvornosti poruke) te za šifriranje podataka [8], a sastoji se od privatnog (engl. *private key*) i javnog ključa (engl. *public key*). Funkcionira tako da korisnik A koji želi poslati poruku korisniku B, šifrira poruku (engl. *plain text*) javnim ključem korisnika B. Kad primi poruku, korisnik B svojim privatnim ključem (koji je tajan i ne dijeli ga ni s kim) dešifrira pristiglu poruku (engl. *cipher text*). Ilustracija rada RSA algoritma prikazana je ispod.

[3]Rivest, Shamir i Adlerman: Prezimena autora RSA algoritma.



Slika 3.2: RSA algoritam

AES[4] jedan je od kriptografskih algoritama za zaštitu digitalnih podataka. AES standard temelji se na simetričnom Rijndael algoritmu, a kao standard razvijen je da bi postupno zamijenio DES[5], čija sigurnost (DES je duljine 56 bita, dok AES može iznositi do 256 bita) u današnje vrijeme više nije dovoljna [9]. AES je poput DES-a algoritam simetričnog ključa što znači da se isti ključ koristi i za šifriranje i za dešifriranje poruke.

3.2.3. PostgreSQL

PostgreSQL je objektno-relacijski sustav za upravljanje bazama podataka s naglaskom na proširivost i usklađenost s normama. Kao poslužitelj baze podataka, njegove primarne funkcije su sigurno pohranjivanje podataka i vraćanje podataka kao odgovor na zahtjeve drugih aplikacija. Može podnositi opterećenja od malih sustava s jednim korisnikom, sve do velikih sutava s mnoštvom korisnika i zahtjeva [10]. PostgreSQL je sukladan ACID-u[6] što ga čini jednim od najkorištenijih sustava za upravljanje bazama podataka [11].

[4]Napredni enkripcijski standard (engl. *Advanced Encryption Standard*)

[5]Algoritam simetričnog ključa (engl. *Data Encryption Standard*)

[6]ACID - Nedjeljivost, konzistentnost, izolacija, izdržljivost (engl. *Atomicity, Consistency, Isolation, Durability*)

3.2.4. Django

Django je platforma otvorenog koda (engl. *open source*) pisana u programskom jeziku Python, a namijenjena je jednostavnijem razvoj web aplikacija. Svaki put prilikom razvijanja nove web aplikacije, programeri se susreću s istim problemima: potrebno je napraviti autentifikaciju korisnika, upravljačku ploču za administratora stranice, HTML forme, sustav za dodavanje datoteka i slično. Django se sastoji od gotovih često korištenih komponenti što olakšava i ubrzava razvoj. [12].

3.2.5. Python

Python je programski jezik opće namjene, interpretiran i visoke razine. Po automatskoj memorijskoj alokaciji, Python je sličan programskim jezicima kao što su Perl, Ruby, Smalltalk itd. Python dopušta programerima korištenje nekoliko stilova programiranja. Objektno orijentirano, strukturno i aspektno orijentirano programiranje stilovi su dopušteni korištenjem Pythona te ova fleksibilnost čini Python programski jezik sve popularnijim [13].

3.2.6. Git

Git je distribuirani sustav koji služi za upravljanje izvornim kodom i praćenje promjena u računalnim datotekama i koordinaciju rada na tim datotekama između više osoba [14]. Besplatan je i otvorenog koda što ga uz jednostavnost korištenja čini najboljim sustavom te vrste [15].

3.3. Implementacija

Rješenje je implementirano iz 3 dijela:

1. Blockchain dio sustava koji služi osiguranje transparentnosti javnih podataka (transkacija) i nepromjenjivost podataka unesenih u bazu podataka.

2. Baza podataka drugi je ključan dio sustava. Ona sadrži javne podatke (genome) u nešifriranom obliku koji se poslužuju krajnjim korisnicama, te čvorove (adrese) rudara koji održavaju blockchain mrežu.

3. Sučelje, kao jedini dio koji donori genoma vide, poveznica je između blockchaina i baze podataka. Ono putem *API-ja*[7] komunicira s blockchainom te tako omogućuje svakom prosječnom korisniku jednostavan unos podataka i obavljanje transakcija na blockchainu.

Također je predviđeno da rješenje koriste tri tipa korisnika:

1. Prvi tip korisnika su rudari koji održavaju blockchain mrežu, proizvode nove blokove, verificiraju primljene i poslane podatke te tako osiguravaju transparentnost i nepromjenjivost podataka.

2. Donori genoma su korisnici koji putem sučelja učitavaju svoje genome u bazu podataka i njihove *hasheve* u blockchain. Oni u svakom trenutku mogu pristupiti svim relevantnim informacijama vezanim uz genom koji su učitali u mrežu poput popisa svih organizacija koje su kupile genom, opisa istraživanja (odgovora) koje su organizacije ostavile na taj genom te verifikaciji svih transakcija, pohranjenih istraživanja ili nepromjenjivosti učitanog genoma.

3. Organizacija je korisnik s javnim profilom. Ona može pristupiti popisu svih učitanih genoma, kupiti učitane genome te ih potom preuzeti. Nakon provedenog istraživanja nad genomom, ima mogućnost pohraniti jedan ili više odgovora (rezultata istraživanja) koje mogu vidjeti i korisnici koji su učitali genom, a i same organizacije kojima ti odgovori mogu služiti kao svojevrsni dnevnik zapisa.

[7] Aplikacijsko programsko sučelje (engl. *Application Programming Interface, API*) ili sučelje za programiranje aplikacija je skup određenih pravila i specifikacija koje programeri slijede tako da se mogu služiti uslugama ili resursima operacijskog sustava. [16]

3.3.1. Blockchain

Blockchain (3.2.1) je implementiran kao samostalna Django (3.2.4) aplikacija. Razvijen je tako da rudari, neovisno o korisnicima sučelja mogu preuzeti kod s git (3.2.6) repozitorija i lokalno pokrenuti instancu koja s ostalima rudarima u mreži komunicira putem API-ja.

Opis bloka

Lanac se sastoji od blokova u sljedećem obliku:

```
{
    "index": "(int) redni_broj_bloka",
    "data": "(dict) podatci",
    "timestamp": "(int) vrijeme_nastanka_bloka",
    "previous_hash": "(str) hash_prethodnog_bloka",
    "hash": "(str) hash_bloka",
    "proof": "(int) dokaz_o_obavljenom_rudarenju"
}
```

Više o načinu nastanka sažetka bloka možete pročitati u poglavlju 3.3.1

Vrijednost *data* sastoji se od tri različite vrste podataka: *uploads*, *downloads* i *transactions*.

```
{
    "uploads": "lista_učitavanja",
    "downloads": "lista_odgovora",
    "transactions": "lista_transakcija"
}
```

Svaka navedena vrsta podataka ima svoje posebnosti koje su navedene u sljedećim isječcima koda.

1. Učitavanja (engl. *uploads*) predstavljaju sve podatke o učitanim genomima.

```
{
    "genome_hash": "hash_učitanog_genoma",
    "owner_public_key": "javni_ključ_donora",
    "timestamp": "vrijeme_učitavanja"
}
```

Više o načinu nastanka javnog ključa možete pročitati u poglavlju 3.3.1.

2. Kupovine (engl. *transactions*) je lista podataka o transakcijama između dva korisnika sustava:

```
{
    "sender_key": "javni_ključ_pošiljatelja",
    "receiver_key": "javni_ključ_primatelja",
    "timestamp": "vrijeme_nastanka",
    "amount": "iznos",
    "transaction_hash": "hash transakcije",
    "signature": "potpis_transakcije"
}
```

Više o načinu nastanka sažetka i potpisa možete pročitati u poglavlju 3.3.1.

3. Preuzimanja (engl. *downloads*) je lista podataka o odgovorima. Odgovori su zapravo rezultati istraživanja provedenog nad određenim genomom koje organizacije pohranjuju kroz sučelje sustava.

```
{
    "response_hash": "hash_odgovora",
    "organization_public_key": "javni_ključ_organizacije",
    "signature": "potpis_odgovora",
    "genome_hash": "hash_istraživanog_genoma",
    "timestamp": "vrijeme_nastanka"
}
```

Inicijalizacija lokalne instance blockchaina

Nakon što rudar preuzme izvorni kod s git repozitorija, može se registrirati na blockchain mrežu i rudariti. Pokretanje programa i registraciju na mrežu čini sljedećom naredbom.

```
./manage.py runserver <IP_ADRESA_RUDAREVOG_RACUNALA>
```

Argument <IP_ADRESA_RUDAREVOG_RAČUNALA> je obvezan i bez njega se program ne može pokrenuti budući da se prilikom pokretanja sustava ta adresa šalje u bazu podataka[8] u tablicu Čvorovi (engl. *Nodes*) (3.3.2). Zatim iz te tablice u bazi povlači adrese svih računala u mreži koje su bitne za izvršavanje konsenzusa, o kojem će više riječi biti u nastavku.

Generiranje sažetka

Često spominjani sažetak bloka jedna je od ključnih informacija koju svaki blok sadrži. *Hash* funkcija je svaki algoritam koji od podatka proizvoljne dužine generira podatak fiksne duljine [17]. Prilikom implementacije funkcije koristio sam SHA256[9] algoritam za kriptografski sažetak koji podatke šifrira jednosmjerno, odnosno tako da se više ne mogu vratiti u početno stanje. Prilikom izračunavanja sažetka bloka u obzir sam uzeo *timestamp* nastanka bloka, podatke u bloku (u obliku niza znakova (engl. *string*)), te sažetak prethodnog bloka što čini blokove sigurnima i teško manipuliranim. Naime, SHA256 algoritam sažetke generira tako da i najmanja promjena ulaznih podataka u funkciju sažetka uvelike utječe na izlaz što je vidljivo u sljedećem primjeru.

Kao konstantu za sažetak prethodnog bloka uzmimo sljedeći niz znakova:
"40670c9ca50548c2cffbe64ac276849a0f0c40f6e0f6a40d0d237f49b599bb38".
U tablici 3.1 prikazano je djelovanje SHA256 funkcije.

[8] O načinu implementacije i izgledu tablica baze podataka čitajte u poglavlju 3.3.2.
[9] Sigurni algoritam za kriptografski sažetak (engl. *Secure Hash Algorithm*)

Vrijeme / Podatci	1528230470.189587
Završni rad	074a3b79d24fccc4ce8d481ec8270217 e25405ba05f9aacc8115acf0e77c0321
završni rad	f62370759d256d721c3e3453171240eb 4e363c368dd25928a96f5efc969dfcfb

Tablica 3.1: Prikaz računanja sažetka bloka

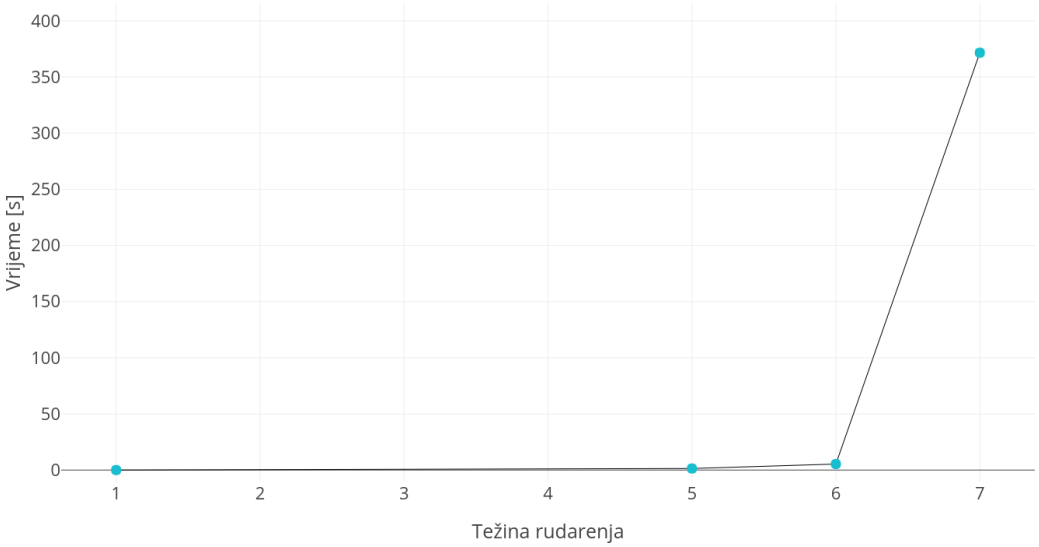
Na primjeru smo se uvjerili kako i najmanja promjena ulaznih podataka (malo i veliko slovo Z) prilikom generiranja sažetka uvelike utječe na izlaz i konačan sažetak. Ovo je vrlo važna osobina blockchaina jer prilikom provjere ispravnosti lanca tijekom konsenzusa, svaki rudar za sebe ponovno izračunava sažetak svakog bloka i time validira ispravnost primljenog lanca i zbog toga je jako teško, gotovo nemoguće, promijeniti bilo koji podatak unutar blockchaina.

Proof-of-work

Proof-of-work (3.2.1) je rad koji rudar obavlja kako bi potrošnjom resursa dokazao da je izrudario novi blok. Svaki blockchain ima tzv. težinu rudarenja novog bloka (engl. *proof-of-work difficulty*). Težina rudarenja novog bloka (engl. *difficulty*) je broj u intervalu [0, 64] koji predstavlja koliko početnih nula mora imati izračunati sažetak u funkciji rudarenja. U funkciju ulaze broj (*i*), koji je inicijaliziran na nulu i težina izračuna prethodnog bloka (*dif*). Zatim se računa sažetak niza znakova kojeg zajednički čine *i* i *dif*. Svakom iteracijom u kojoj se ne pogodi ispravan sažetak, *i* se uvećava za jedan. Broj koji na kraju u kombinaciji s težinom izračuna prethodnog bloka vrati sažetak koji ima *difficulty* vodećih nula je novi dokaz (engl. *proof*) izrudarenog bloka.

Težina	Vrijeme rudarenja	Novi dokaz (broj iteracija)
1	0.00003	3
5	1.37	771 986
6	5.34	3 140 321
7	371.63	222 768 594

Tablica 3.2: Primjer povećavanja vremena rudarenja novog bloka s obzirom na težinu rudarenja



Slika 3.3: Graf ovisnosti vremena o težini rudarenja

Potpisivanje i verifikacija

Potpisivanje i verifikacija poslanih odnosno primljenih transakcija također je jedna od bitnijih značajki blockchaina. Potpisom transakcije pošiljatelji osiguravaju odaslanu transakciju (ili bilo koji drugi podatak, u ovom slučaju i odgovor organizacije) od zlonamjernih rudara, a verifikacijom ostali rudari i korisnici ustvrđuju autentičnost primljene transakcije. Funkcionira na principu privatnog i javnog ključa. Funkcija potpisivanja na ulaz prima privatni ključ pošiljatelja i sažetak transakcije (ili bilo kojeg drugog podatka).

```
def sign_transaction(private_key, transaction_hash)
```

Korištenjem SHA256 algoritma generira se sažetak sažetka transakcije na kojeg zatim pošiljatelj potpiše s privatnim ključem. Taj potpis (engl. *signature*) zatim se još jednom šifrira, ovaj put simetričnim AES algoritmom samo radi lakšeg prijenosa u obliku niza znakova.

Kada rudar (ili neki drugi korisnik) vide pristiglu transakciju, oni je moraju verificirati kako bi je dodali u budući blok. Funkcija verifikacije na ulaz prima javni ključ pošiljatelja transakcije, sažetak transakcije i potpis.

```
def verify(public_key, transaction_hash, signature_aes)
```

U ovoj se funkciji također napravi sažetak sažetka transakcije na koji se zajedno s potpisom primijeni verifikator kreiran s javnim ključem pošiljatelja. Ukoliko je izlaz funkcije verifikacije istinit (engl. *true*), tada će transakcija biti dodana u novi blok jer rudar može biti siguran da nitko nije prepravio primjerice javni ključ primatelja i time sebi neovlašteno priskrbio novčiće.

Konsenzus (engl. *consensus algorithm*)

Konsenzus algoritam predstavlja odabir važećeg i referentnog lanca u blockchain mreži. Svaki lanac sadrži listu registriranih čvorova (adresa) ostalih rudara. Prilikom rudarenja novog bloka poziva se funkcija konsenzusa koja dohvaća lanac sa svakog čvora te provjerava njegovu valjanost. Ukoliko su 2 lanca valjana, kao pobjednik izlazi onaj koji u trenutku ima više blokova. Ukoliko oba imaju jednak broj blokova, čeka se novi blok koji će odrediti novi, važeći lanac u mreži. U ovoj implementaciji, funkcija konsenzusa poziva se odmah prilikom pokretanja lokalne instance blockchaina kako bi svaki rudar odmah na početku preuzeo važeći lanac.

Novčanik (engl. *Wallet*)

Svaki korisnik sustava prilikom prvog ulaska u sustav (registracija za organizaciju, pokretanje lokalne instance za rudara i doniranje genoma za donora) dobije vlastiti novčanik generiran od strane sustava. Novčanik se sastoji od privatnog i javnog ključa generiranih RSA algoritmom koje sustav nigdje ne bilježi nego se samo generiraju i pošalju korisniku. Privatni ključ korisniku služi za potpisivanje transakcija i podataka, a javni kao vanjski identifikator na koji mu ostali korisnici mogu poslati transakciju ili verificirati njegov potpis. Ukoliko korisnik želi doznati stanje svog računa, mora prilikom upita predati i privatni i javni ključ.

3.3.2. Baza podataka

Baza podataka drugi je dio rješenja nastao iz potrebe pohrane genoma koji su preveliki da bi se čuvali unutar samog blockchaina. Baza podataka nije i u ovom slučaju ne smije biti "izvor istine" (engl. *source of truth*) jer bi se time izgubila poanta blockchaina. Iz baze se zatraženi podatci dohvaćaju ako i samo ako sažetci (reference) tih podataka postoje u važećem lancu unutar blockchain mreže.
Sastoji se od 4 tablice čije ću funkcionalnosti pobliže objasniti u sljedećim pododjeljcima.[10]

Tablica *Genome*

Redak u ovoj tablici nastaje u trenutku kad donor genoma u sučelju učita datoteku sa svojim genomom, a sastoji se od 4 polja:

1. Javni ključ vlasnika genoma (engl. *owner public key*) jedina je oznaka donora u bazi podataka. Ovaj podatak se sprema kako bi kasnije korisnik mogao prodati svoje podatke organizacijama.

2. Sažetak genoma (engl. *genome hash*) služi u svrhu jedinstvenosti genoma u sustavu. Ne mogu se dodati 2 ista genoma u sustav.

3. Datoteka (engl. *file*) čiji sadržaj je genom

4. Jedinstveni identifikator genoma (engl. *genome unique identifier*). U trenutku učitavanja datoteke (ili uređivanja retka u ovoj tablici) formira se niz znakova kojeg čine sažetak genoma, javni ključ korisnika i *timestamp* učitavanja. Sažetak tog niza znakova tvori jedinstveni identifikator genoma koji donoru služi za verifikaciju integriteta i autentičnosti učitane datoteke i njezinog sadržaja.

Tablica *Nodes*

Tablica čvorova sadrži samo adrese računala svih rudara u blockchain mreži.

Tablica *Organizations*

Budući da je profil svake organizacije javan, podatci o toj organizaciji spremljeni su u bazi u ovoj tablici. Ona se sastoji od sljedećih polja:

[10] Svaki redak svih tablica uvijek sadrži datum nastanka (engl. *date created*) i datum posljednjeg uređivanja (engl. *date modified*) tako da to neću posebno opisivati u svakom pododjeljku.

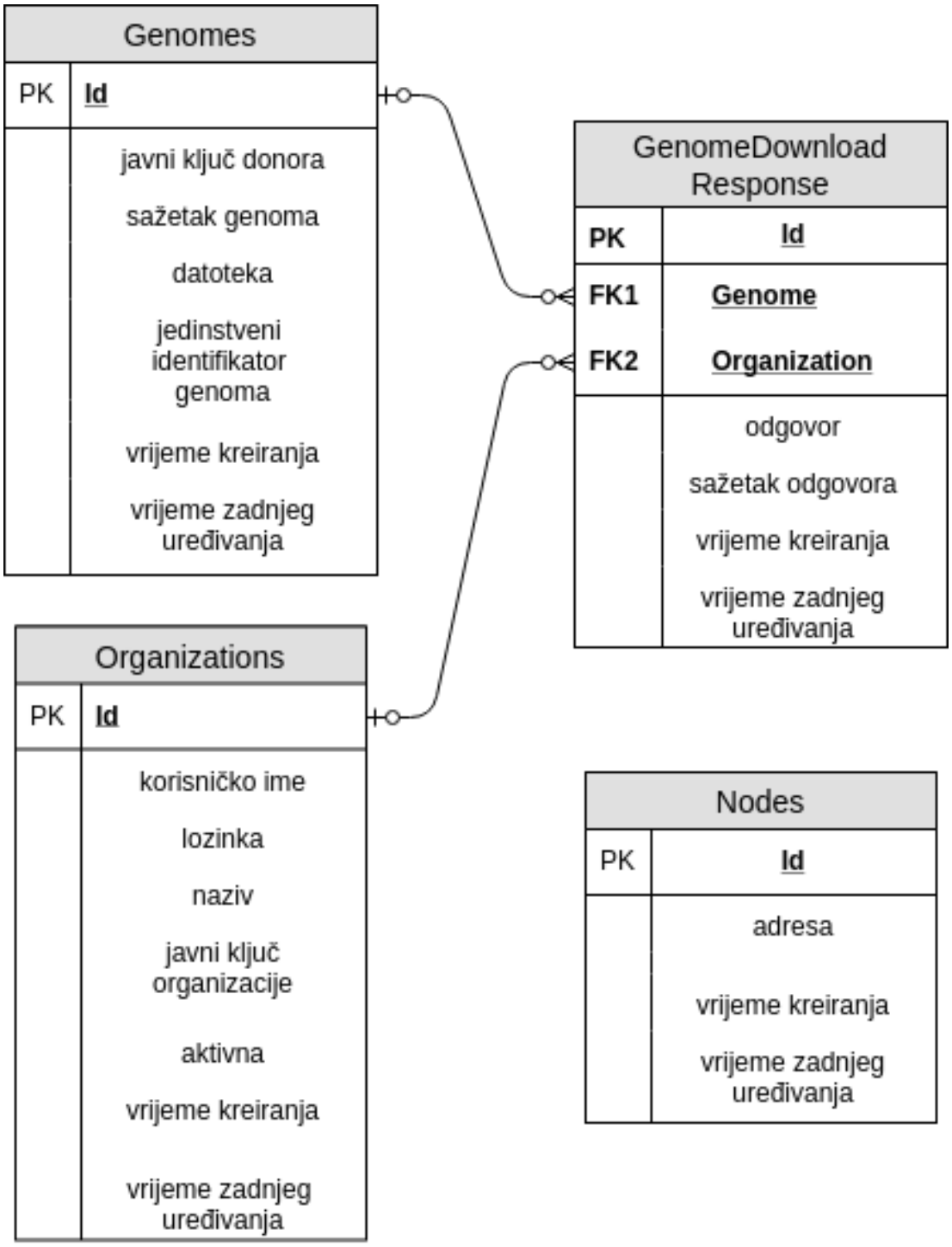
1. Korisničko ime (engl. *username*) koje se koristi za spajanje u sustav
2. Lozinka (engl. *password*)
3. Ime organizacije (engl. *name*)
4. Javni ključ organizacije (engl. *organization public key*). Koriste ga donori kako bi mogli verificirati istinitost odgovora.
5. Aktivna (engl. *is active*) je polje koje određuje je li profil organizacije aktivan. Ukoliko profil nije aktivan, organizacija ne može pristupiti korisničkom sučelju za kupnju i preuzimanje genoma.

Tablica *GemomeDownloadResponse*

Svaki put kad organizacija provede istraživanje nad genomom i odluči to podijeliti s vlasnikom genoma i istovremenom voditi svoj dnevnik zapisa, kreira se redak u ovoj tablici.

1. Genom, referenca na genom za koji se dodaje odgovor.
2. Organizacija, referenca na organizaciju koja dodaje odgovor.
3. Odgovor (engl. *response*), može biti bilo kakav tekst.
4. Sažetak odgovora (engl. *response hash*) nastaje na temelju niza znakova dobivenog od sažetka transakcije, u kojoj je organizacija platila donoru pravo na korištenje genoma, i *timestampa* kreiranja odgovora

Svaka od ovih tablica ima zabranu uređivanja kako bi se dodatno osigurala autentičnost unosa.



Slika 3.4: E-R dijagram baze podataka

3.3.3. Poslužitelj i sučelje

Sučelje je implementirano kao samoostalna Django(3.2.4) aplikacija koja s blockchain mrežom komunicira putem API-ja. Koriste ga sve tri vrste korisnika:

1. Donori genoma jednostavno i sigurno mogu dodati svoj genom u blockchain mrežu te pratiti korištenje i istraživanja vezana uz isti. Mogu i verificirati autentičnost i integritet podataka koje su učitali u sustav.

2. Organizacije na jednom mjestu imaju pristup velikoj količini podataka gdje u nekoliko klikova mogu zakupiti prava na korištenje i naknadno dodavati rezultate istraživanja provedenih nad određenim genomom.

3. Rudari, tj. budući rudari, u svakom trenutku mogu vidjeti koliko čvorova održava blockchain mrežu i isplati li im se po tome rudarenje na toj mreži

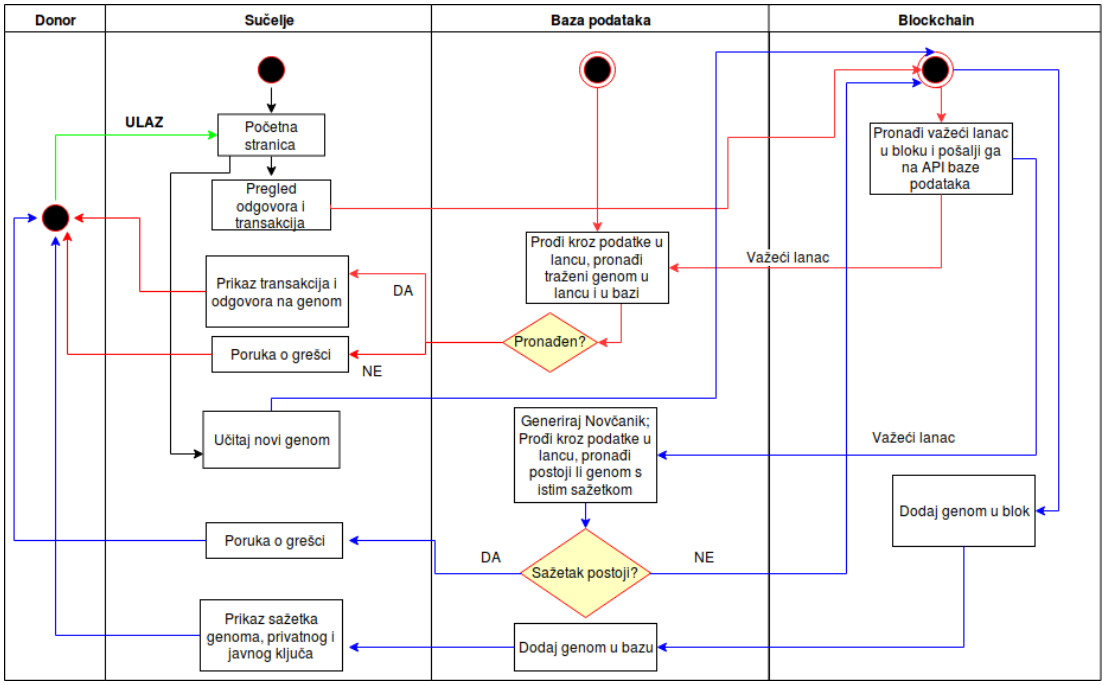
Također, sve vrste korisnika u svakom trenutku mogu vidjeti popis svih transakcija na mreži i svaku pojedinačno verificirati. Više o protoku pojedine vrste korisnika kroz sučelje i sustav u cjelini čitajte u poglavlju 3.4.

3.4. Protok korisnika

3.4.1. Donor genoma

U potpunosti anoniman korisnik, tj. bilo koja osoba koja pristupi sučelju može učitati datoteku sa svojim genomom i pohraniti ga u blockchain mrežu i bazu podataka. Prilikom učitavanja genoma, od sadržaja datoteke stvara se sažetak i jedinstveni identifikacijski sažetak genoma te se generiraju privatni i javni ključ. Zahtjev za dodavanje unesenih podataka šalje se putem API-ja na blockchain mrežu, a u bazi se kreira redak u tablici *Genome* (3.3.2). U slučaju uspješne pohrane podataka, na sučelju se donoru prikažu privatni i javni ključ, sažetak te jedinstveni identifikacijski sažetak genoma. Sustav ne bilježi nikakve informacije o korisniku, a sve što zna nalazi se u kreiranom retku u tablici (engl. *Genome*). Bitno je za napomenuti da sustav ne zna je li isti donor pohranio dva različita genoma budući da se za svako učitavanje novog genoma generiraju novi privatni i javni ključ i sustav ih bilježi kao dva različita donora. U slučaju neuspješnog pohranjivanja genoma (sažetak genoma se već nalazi u blockchain mreži ili se dogodi nekakva druga greška), korisniku se ispisuje odgovarajuća poruka.

Pomoću sažetka genoma i dodijeljenog mu javnog ključa korisnik u svakom trenutku na stranici *Responses* može vidjeti sve transakcije i odgovore koje su organizacije pohranile za taj genom. Također, na stranici *Verify* u svakom trenutku može koristeći se sažetkom i jedinstvenim identifikacijskim sažetkom genoma vidjeti je li genom koji se trenutno predstavlja kao genom s tim sažetkom uistinu genom koji je donor pohranio. Ukoliko se jedinstveni identifikacijski sažetak koji je korisnik unio (koji mu je sustav generirao prilikom pohrane) i onaj pohranjen u bazi ne podudaraju, tada postoji vjerojatnost da je netko mijenjao podatke vezane za taj genom.[11].

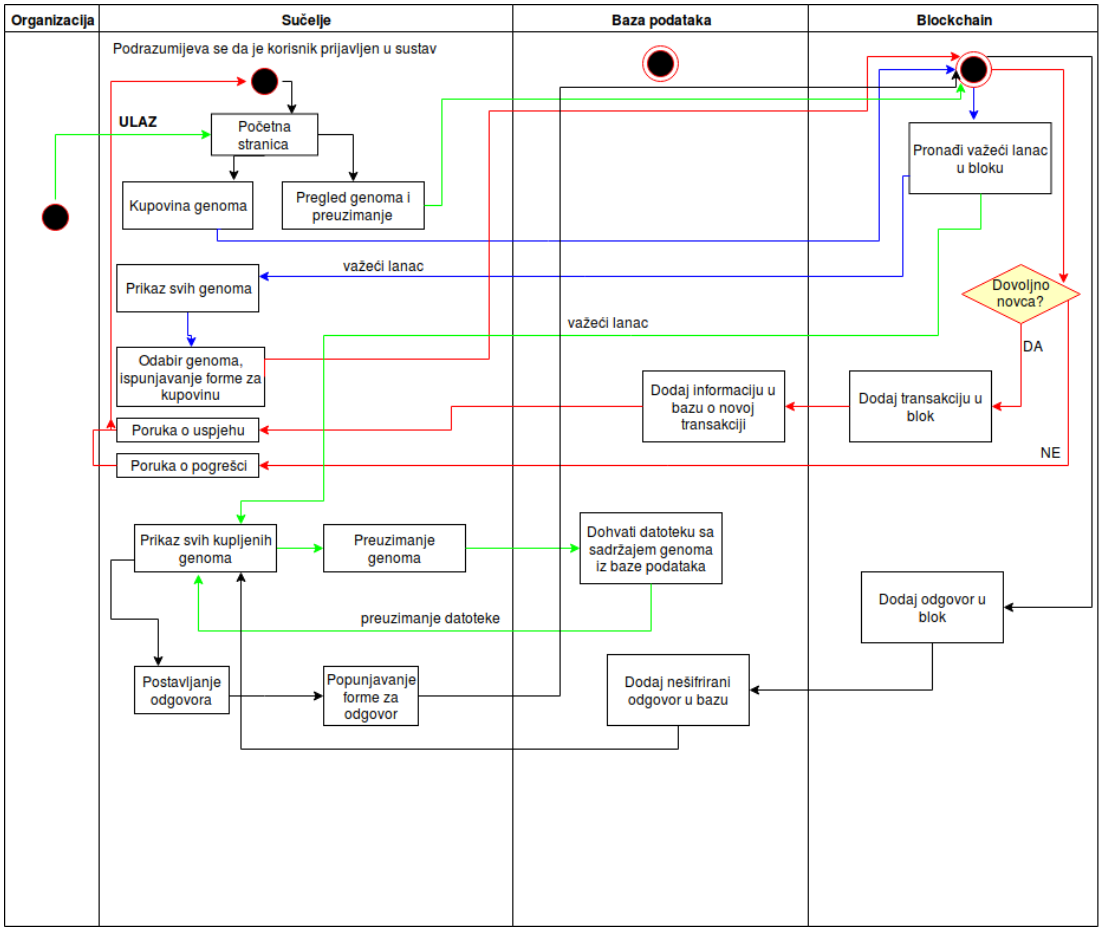


Slika 3.5: Dijagram aktivnosti donora genoma

[11] Više o načinu generiranja jedinstvenog identifikacijskog sažetka pročitajte u poglavlju 3.3.2

3.4.2. Organizacija

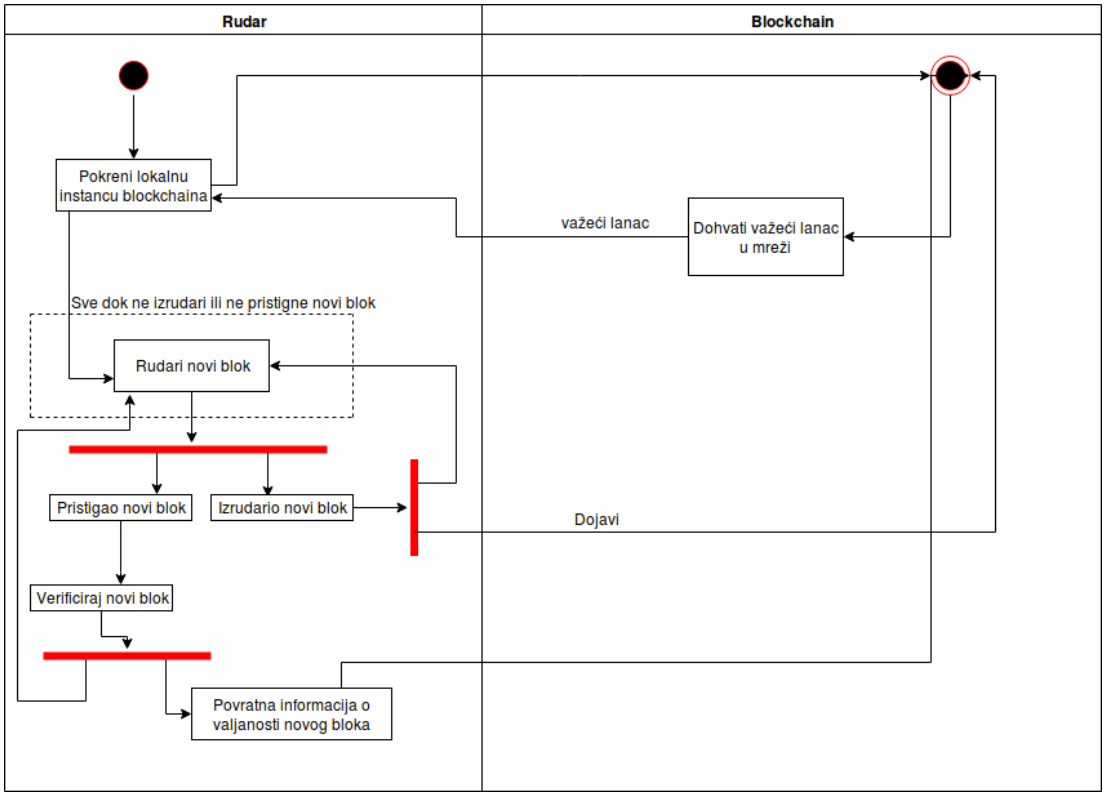
Organizacija je korisnik koji želi kupiti pravo na korištenje pohranjenog genoma. U sustav se registrira s korisničkim imenom i lozinkom. Prilikom uspješne registracije sustav generira privatni i javni ključ organizacije koji ona koristi za potpisivanje transakcije pri kupnji genoma i potpisivanje odgovora (engl. *response*) pri pohrani odgovora o provedenom istraživanju u sustav. Nakon prijave u sustav, dvije su organizaciji najbitnije stranice: Kupovina (engl. *Buy*) te Preuzimanja (engl. *Download*). Na stranici za kupovinu genoma organizacija odabire genom koji želi kupiti te potpisuje transakciju sa svojim privatnim ključem. Jednom kad se ta transakcija nađe u važećem lancu u blockchain mreži bit će vidljiva na stranici s prikazom svih transakcija (engl. *Transactions*), a organizacija će na stranici *Download* moći preuzeti kupljeni genom, odnosno pohraniti odgovor za određeni genom.



Slika 3.6: Dijagram aktivnosti organizacije

3.4.3. Rudar

Rudar je korisnik sustava koji održava blockchain mrežu i brine se o autentičnosti svih podataka pristiglih u mrežu. Na slici 3.7 prikazan je proces rudarenja.



Slika 3.7: Dijagram aktivnosti rudara

4. Rezultati

Sustav je testiran lokalno na jednom računalu s četiri instance blockchain sustava, te po jednom instancom poslužitelja za sučelje i poslužitelja za bazu podataka.

4.1. Konfiguracija računala

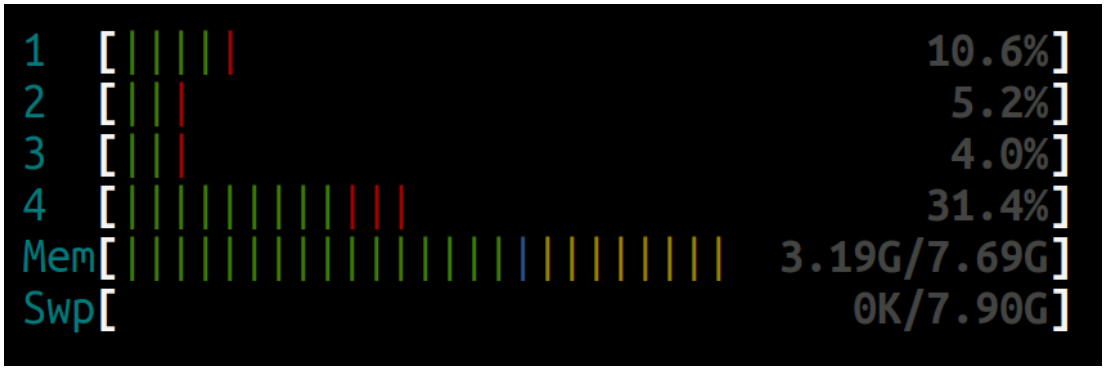
Budući da je rudarenje zahtjevan postupak koji troši mnogo resursa, a vrijeme rudarenja razlikuje se od računala do računala, smatram bitnim za napomenuti o kojoj konfiguraciji je riječ. Računalo na kojem je provedeno testiranje ima 8GB DDR4 RAM-a (2133 MHz), četverojezgreni Intel i5-6300HQ (2.30 GHz) procesor, grafičku karticu Nvidia GeForce GTX 960M 4GB GDDR5, te SSD[1].

4.2. Postavke blockchaina

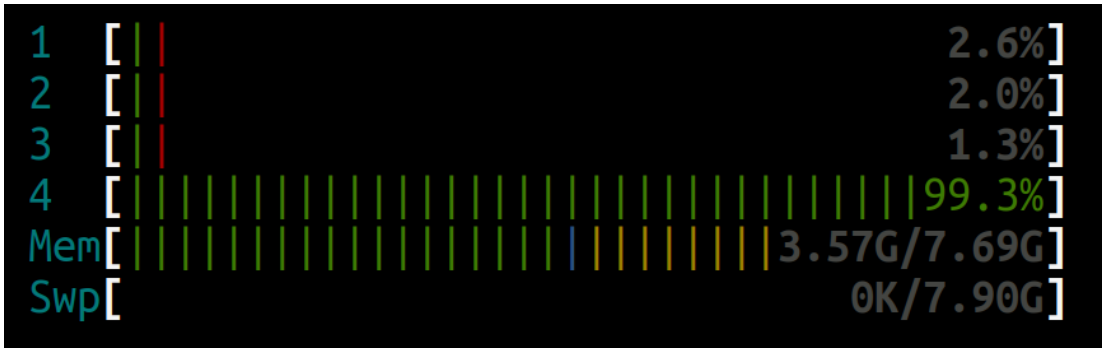
Pokrenute su četiri instance blockchain mreže (četiri rudara) na različitim vratima (engl. *port*), a težina rudarenja postavljena je na 7[2]. Na slici 4.1 vidljivo je opterećenje računala u ovisnosti o broju procesa rudarenja.

[1] Pogon čvrstog stanja (engl. *Solid State Disk*).

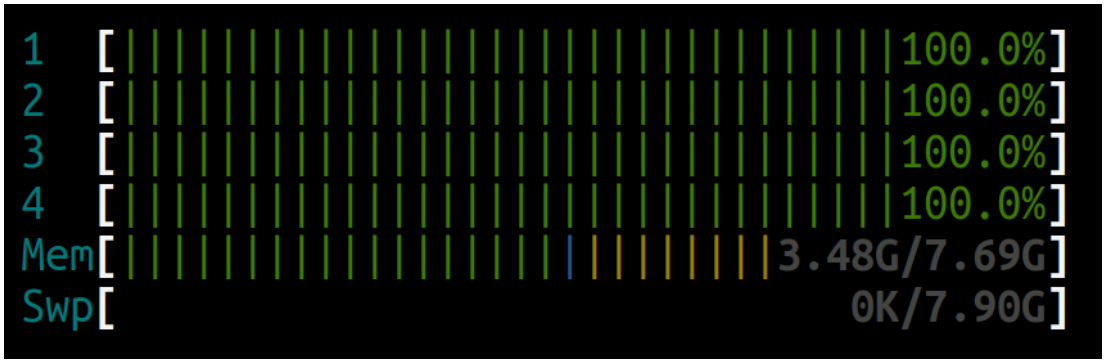
[2] Ovisnost vremena o težini rudarenja pogledajte u tablici 3.2



(a) Opterećenje računala u uobičajenom načinu rada bez rudarenja



(b) Opterećenje računala s jednim rudarom



(c) Opterećenje računala s četiri rudara

Slika 4.1: Opterećenje računala u ovisnosti o broju aktivnih procesa rudarenja

4.3. Podaci

Podaci za testiranje sustava su maksimalno iznosili 20MB, a preuzeti su sa stranica američkog NCBI-ja (engl. *National Center for Biotechnology Information*)[3]. U biologiji, genom nekog organizma jest ukupni genetski materijal istog, tj. svi njegovi nasljedni podatci kodirani u DNK (kod nekih virusa u RNK). Time su obuhvaćeni kako geni tako i ne-kodirajuće sekvence DNK [18]. Sastoji se od parova nukleotida koji su monomerske jedinice nukleinskih kiselina čija se baza (u slučaju DNK) sastoji od purina adenina (A) i gvanina (G), te pirimidina citozina (C) i timina (T) [19]. Primjer dijela sadržaja datoteke genoma prikazan je na slici

```
ACCCTCAGGAGCTGCTGCACCCCTAGGTAATCCCGATCCAACATCGAGGT
AGAACTCTCCAAGAAAATTACGCTGTTATCCCTACGGTAACTTATTCCTC
ATCCCCGAAAAAGCAACACCTGCATGGAAGTTGGCACTTCACCAAATGCT
TTCTCCGCAGTTGCCCCAACTAAAGCTACAAGCACATAACCAAGCCGCCT
TTAAAAAAGAAAATTAGCCCCTAGGCGACCAAACCAACTTGTGCAACTAA
CTCGTCCCACGATTACACCCCCGCTTCTTCACGAGGATATCAAATTCAAG
AACCTCCGTCTTGCCATTCATACCAGCCCCCATTCAAAGGGCAAATGATT
AGATACCGCGGCCGTTTAACTAAGTCACCGGGCAGGCAGGACCCTTTATA
GGCGATGTTTTTGGTAAACAGGCGAAGTCCCCACATTTGCCGAGTTCCTT
TACCAGGCCGCTTGAGTTAGCAGACAGACAAAAGCGAACTATGCTTTCCC
ATCAAAAATTAGGACTGCTTAATACACAAAAGGCCTACTCTACTTCTTTA
CCTGCAACCCCGCAGGACTCCCCCGTAAAATAGTCACAACATAAAACCCG
ACGCCACGGGCTACATAAAACAAGCTTTAACGCCCTCGTTCGGATAGATG
CGAGACCCAATTAACGTGTTTCCCTGCGCATTTTTAACCCATGGCTCAGG
AAGCTTATCCCTCCTTGTCTACGCTTTGAATTCCCATTAAGGATTTAATC
ACCTTAATAGCCAAACAAACATGGCTAACACCAATTTCCGGGGAAACCAG
CATTTCACATCTAATGTACCCTCAAACCTTACTATTGCAACAGTAAAAGC
ACACTAGCTCGCTCAAATTCGGGTACGGCAACTCTTTACCAAAATACTCA
TACAGGGCTTTACTCTTTCTTTCCTACTCTTAGATATTTCACGATATTTC
CCCCTTTAGCTCTTACTACGAGTTTCACTGATAACCTACTACTCCTGCTT
TGGTAAACTTAAAACATCACAGTTTTATTATTCCCTCACCCTTACAAGGC
```

Slika 4.2: Dio sadržaja datoteke genoma

[3]Stranica za preuzimanje genoma: ftp://ftp.ncbi.nih.gov/genomes/

4.4. Veličina lanca

Za primjer će u sustav biti dodan genom čija veličina iznosi 4113027 B (oko 3.9 MB). Ta datoteka s genomom se čuva u bazi podataka, a u bloku (lancu) samo referenca na nju. Ukoliko pogledamo lanac nakon rudarenja novog bloka s pristiglim genomom, vidimo da se u listi *uploads* nalazi novi podatak ukupne duljine 728 B (sažetak genoma iznosi 64 B, a šifrirani javni ključ donora 664 B). Dodamo li tome ključeve[4] i interpunkcijske znakove JSON formata podatka u bloku, dolazimo do maskimalne (fiksne) duljine podatka o učitanom genomu od 771 B (oko 0.0007 MB).

Dobiveni rezultat od 771B znači da se ovim načinom implementacije u lancu veličina podatka smanjila za otprilike 5000 puta. Ukoliko uzmemo u obzir da je veličina ljudskog genoma za koji je ovaj sustav napravljen otprilike 720 MB ([4]) dolazimo do uštede prostora od nešto više od 900000 puta. Da je lanac velik i nespretan za korištenje (rudari stalno preuzimaju i šalju podatke putem interneta), ne bi postojala velika mreža rudara, kao što je slučaj malog lanca nad kojim se može rudariti i pri manjim brzinama interneta. Valja napomenuti da što je veća mreža rudara koji održavaju blockchain mrežu, tim je manja šansa da zlonamjeran rudar preuzme kontrolu nad čitavom mrežom. Stoga, što je manji lanac, tim je veći broj rudara, a s povećanjem broja rudara povećava se i povjerenje donora u čitav sustav.

[4] *JavaScript Object Notation* - JSON, "key": "value"

5. Diskusija

Vratimo se na poglavlje 2 i pokušajmo odgovoriti na tamo navedena tri najznačajnija problema:

1. Potreba za sekvenciranjem genoma zadovoljava se nepovratno iz dana u dan, pogotovo iz razloga što cijena sekvenciranja opada.

2. Donori su u potpunosti anonimni. Nitko ne zna čiji genom proučava i istražuje, a sam donor ima u potpunosti transparentan i jednostavan pristup informacijama vezanim uz svoj genom.

3. Genomi su pohranjeni na efikasan način, a njihova sigurnost se očituje u količini verifikacija koju obavlja svaki dio sustava neovisno jedan o drugom. Uz to sve i sam donor, korisnik sustava, u bilo kojem trenutku može verificirati integritet i autentičnost pohranjenog genoma koristeći jedinstveni identifikator genoma.

Sustav implementiran u ovom završnom radu nije savršen, nego tek prva iteracija testiranja postavljene hipoteze i problema anonimne i transparentne pohrane te uporabe genoma s kojim se suvremena medicina susreće. Postoji nekoliko razloga radi kojih bi cijeli sustav potencijalno mogao biti u opasnosti.

Broj rudara

Što ako se dogodi da sustav u jednom trenutku sadrži neku veliku brojku od, recimo, milijun genoma, još toliko transakcija i odgovora na genome. Postojali bi ljudi (donori) koji bi se oslanjali na informacije pohranjene u sustav, prema tome prevenirali bolesti, mijenjali način života i sl. Također bi postojao i pozamašan broj organizacija koje bi možda svoj poslovni model temeljile na postojanju sustava koji sadrži toliki broj genoma i u kojem mogu pratiti svoj dnevnik zapisa. Ukoliko se dogodi da iz nekog razloga broj rudara koji održavaju mrežu blockchaina padne na neki vrlo malen broj kao što je pet, netko zlonamjeran vrlo lako može ovladati čitavim sustavom čime automatski dolazi u priliku mijenjati sve informacije koje su korisnici pohranili, ruši

vrijednost kriptovalute koju su donori zaradili prodajom prava na korištenje genoma itd. Izgubilo bi se povjerenje u sustav, donori bi odustali od dodavanja genoma zbog nikakve zarade, a organizacijama bi se izgubio smisao korištenja sustava koji možda ne nudi autentične podatke. Iz svih navedenih razloga vrlo je važno da u sustavu postoji što veći broj rudara kako bi cijela mreža bila sigurnija i vjerodostojnija za korištenje od strane sve tri vrste korisnika.

Napad 51%

Napad 51% svodi se zapravo na broj rudara u sustavu. Zlonamjeran rudar stvori dovoljan broj čvorova (više od 50% trenutnog broja čvorova) koji kreiraju novi lanac blokova s izmijenjenim podatcima. Tada si čvorovi u vlasništvu zlonamjernog rudara međusobno vrlo brzo potvrđuju autentičnost i u trenutku kad 51% rudara u sustavu potvrdi da je to ispravan blok, pravi važaći lanac blokova nad kojim rudari preostalih 49% rudara biva odbačen kao nevažeći i tako napadač ovladava sustavom [20]. Nijedan sustav temeljen na blockchainu ne može se oduprijeti ovom napadu ukoliko ne postoji dovoljan broj poštenih rudara. Primjerice, na Bitcoinovom blockchainu takav napad još uvijek nije uspješno izveden.

Baza podataka

Baza podataka nije osigurana od hakera. Njoj se može pristupiti kroz administratorsko sučelje u kojem se potencijalno mogu mijenjati/brisati postojeći podaci. Napravljen je određeni sloj zaštite u kojem se onemogućuje uređivanje i brisanje unesenih podataka bilo kojem korisniku sučelja, ali što ako haker ipak uspije nekako promijeniti podatke? U tom slučaju događa se scenarij gubitka povjerenja u sustav i gubitka zarade za donore, ali barem je osigurana privatnost donora. Kako bi se poboljšala sigurnost podataka u bazi moguće je primjerice šifrirati sve podatke simetričnim algoritmom poput AES-a čiji bi se ključ generirao nasumično. Moguće je uvesti dvostruku autentifikaciju prilikom pristupa sučelju da korisnik koji pristupa osim korisničkog imena i lozinke unese i kod primljen putem SMS-a. Samu bazu podataka možemo dodatno zaštiti i zabranom pristupa kroz postgres tako da se za svaki pristup bazi kreira novi korisnik s minimalno potrebnim ovlastima i briše odmah nakon odjave ili isteka određenog vremena, a sve to bi nadzirao neki centralni autoritet poput skupa sistemskih administratora

Skalabilnost

Sustav je testiran lokalno, na svega četiri instance blockchaina. Cacheiranje[1] nije implementirano što znači da rudar uvijek preuzima cijeli lanac, te šalje kao svoj odgovor cijeli lanac natrag svakom rudaru u mreži. Dugoročno to nije održivo zbog brzine i cijene održavanja cijelog sustava.

[1] Pohrana podataka koji se ne mijenjaju često, ali iz kojih se često čita u radnu memoriju kako bi posluživanje bilo što brže.

6. Zaključak

U suvremenoj medicini i znanosti o podatcima (engl. *data science*) postoji potreba za sekvenciranjem genoma kako bi se razvila osobna medicina, tj. kako bi liječnici lakše uočili i tretirali zdravstvene probleme dizajnirane prema svakom pojedincu posebno. U današnje vrijeme vrlo je lako i povoljno sekvencirati svoj genom i samim time se automatski nameće pitanje pohrane i zaštite privatnosti donora.

Ovaj rad ponudio je odgovor na to pitanje u vidu spajanja dviju oprečnih tehnologija: centralizirane baze podataka, te decentraliziranog i distribuiranog blockchaina. Baza podataka služi kao skladište podataka kojem se može pristupiti ako i samo ako referenca na taj podatak postoji u trenutnom važećem lancu u mreži blockchaina. Blockchain mrežu održavaju rudari koji za svoj rad bivaju nagrađeni određenom količinom kriptovalute, donori za svoj doprinos razvoju medicine imaju mogućnost zaraditi od organizacija koje kupuju pravo na korištenje tog genoma, te saznaju nešto o svom tijelu budući da organizacije imaju mogućnost voditi dnevnik zapisa u sustavu o istraživanjima provedenim nad kupljenim genomima. Uz sve to, riješen je i problem privatnosti podataka donora koji su u ovoj implementaciji u potpunosti anonimni pojedinci o kojima nitko u sustavu ne zna ništa. Rezultat, barem što se tiče veličine lanca je obećavajući, budući da se radi o malim količinama podataka koji se čuvaju u lancu.[1]

Kao što je već rečeno u poglavju 5, sustav nije savršen i poboljšanja su uvijek moguća. Neka od mogućih bila bi primjerice uvođenje *cacheiranja* čime bi se ubrzala razmjena podataka između rudara, poboljšanje sigurnosti baze podataka, korištenje oblaka (engl. *cloud*) na kojem bi se prilikom eventualnog napada 51% vrlo brzo mogao pokrenuti dovoljan broj poštenih instanci blockchaina i sl.

[1] Vidi 4.4.

LITERATURA

[1] Patrick Lin. Blockchain: The missing link between genomics and privacy? https://www.forbes.com/sites/patricklin/2017/05/08/blockchain-the-missing-link-between-genomics-and-privacy/#339a82e14b77. Pristupljeno: 13. veljače 2018.

[2] Dan Mangan. Your boss could demand you get genetic testing and hand over the results, if this congressional bill becomes law. https://www.cnbc.com/2017/03/10/employers-could-demand-genetic-testing-under-congressional-bill.html. Pristupljeno: 13. veljače 2018.

[3] Mian Zhang i Yuhong Ji. Blockchain for healthcare records: A data perspective. https://doi.org/10.7287/peerj.preprints.26942v1. Pristupljeno: 20. svibnja 2018.

[4] Reid J. Robison. How big is the human genome? https://medium.com/precision-medicine/how-big-is-the-human-genome-e90caa3409b0, 2018. Pristupljeno: 20. svibnja 2018.

[5] Statista. Blockchain size. https://www.statista.com/statistics/647523/worldwide-bitcoin-blockchain-size/. Pristupljeno: 21. svibnja 2018.

[6] Satoshi Nakamoto. Bitcoin: A peer-to-peer electronic cash system. https://bitcoin.org/bitcoin.pdf, 2008. Pristupljeno: 13. veljače 2018.

[7] Denis Arunović. Što je u stvari blockchain i kako radi? https://www.bug.hr/tehnologije/sto-je-u-stvari-blockchain-i-kako-radi-3011, 2018. Pristupljeno: 4. lipnja 2018.

[8] Wikipedia. Rsa. https://en.wikipedia.org/wiki/RSA_(cryptosystem), 2018. Pristupljeno: 4. lipnja 2018.

[9] CARNetCERT. *AES algoritam*, 2003.

[10] Wikipedia. Postgresql. https://en.wikipedia.org/w/index.php?title=PostgreSQL&oldid=844146841, 2018. Pristupljeno: 4. lipnja 2018.

[11] Host4ASP. The commonly used database management systems of web hosting. https://host4asp.net/commonly-used-database-management-systems/, 2014. Pristupljeno: 4. lipnja 2018.

[12] Django. Django. https://tutorial.djangogirls.org/en/django/, 2018. Pristupljeno: 4. lipnja 2018.

[13] Wikipedia. Python (programski jezik). https://hr.wikipedia.org/w/index.php?title=Python_(programski_jezik)&oldid=4653573, 2018. Pristupljeno: 4. lipnja 2018.

[14] Wikipedia. Git. https://en.wikipedia.org/w/index.php?title=Git&oldid=844389685, 2018. Pristupljeno: 4. lipnja 2018.

[15] G2Crowd. The top 6 version control systems. https://www.g2crowd.com/categories/version-control-systems#highest_rated, 2018. Pristupljeno: 4. lipnja 2018.

[16] Wikipedija. Api. https:///hr.wikipedia.org/w/index.php?title=API&oldid=4266194, 2018. Pristupljeno: 4. lipnja 2018.

[17] Wikipedija. Hash function. https://en.wikipedia.org/w/index.php?title=Hash_function&oldid=844015490, 2018. Pristupljeno: 5. lipnja 2018.

[18] Wikipedija. Genom. https://hr.wikipedia.org/w/index.php?title=Genom&oldid=5026208, 2018. Pristupljeno: 7. lipnja 2018.

[19] Wikipedija. Nukleotidi. https:https://hr.wikipedia.org/wiki/Nukleotidi, 2018. Pristupljeno: 7. lipnja 2018.

[20] Investopedia. 51% attack. https://www.investopedia.com/terms/1/51-attack.asp, 2018. Pristupljeno: 7. lipnja 2018.

Sustav za pohranu DNA

Sažetak

U suvremenoj medicini postoji velika potreba za sekvenciranjem i obradom genoma. Cijena sekvenciranja genoma opada puno brže od Mooreovog zakona i trenutno se ljudski genom može sekvencirati za manje od 1000 američkih dolara te se zbog toga mnogi odlučuju sekvencirati svoj genom. Najveća zabrinutost kod sekvenciranja genoma proizlazi iz problema sigurnosti i zaštite korisničkih podataka. U ovom radu testirat će se hipoteza da je navedeni problem sigurnog načina pohrane osjetljivih podataka poput ljudskog genoma te zaštite privatnosti donora genoma rješiv u vidu hibridnog sustava korištenjem distribuirane i decentralizirane tehnologije *blockchain* i centralizirane baze podataka. Rješenje je implementirano iz tri dijela: *blockchaina*, baze podataka i sučelja. Predviđeno je da ga koriste tri tipa korisnika: donori genoma, organizacije koje kupuju i provode istraživanja nad genomom i rudari koji održavaju *blockchain* mrežu. Iako rješenje ima svojih problema poput zaštite baze podataka i skalabilnosti, ono pokazuje da se ovim pristupom distribuirane pohrane genoma može osigurati veće povjerenje donora u očuvanje integriteta i autentičnosti pohranjenih podataka.

Ključne riječi: genom, sekvenciranje, zaštita genoma, blockchain, baza podataka, raspodijeljeno, anonimnost

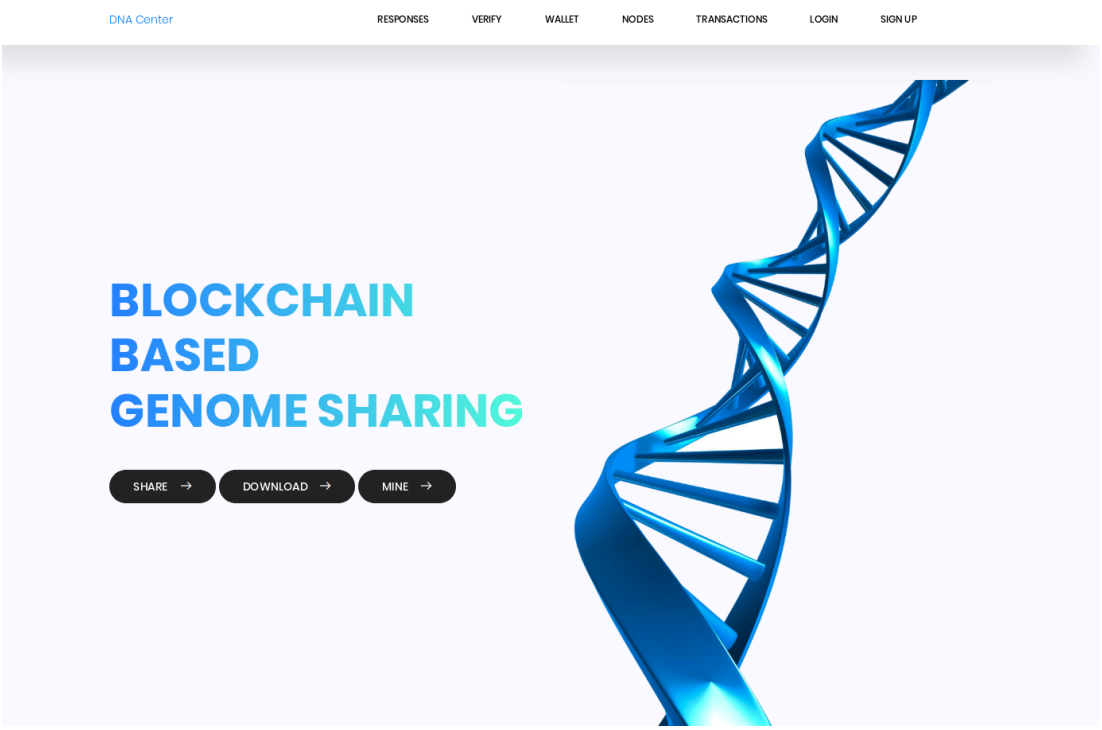
A Service for DNA Backup

Abstract

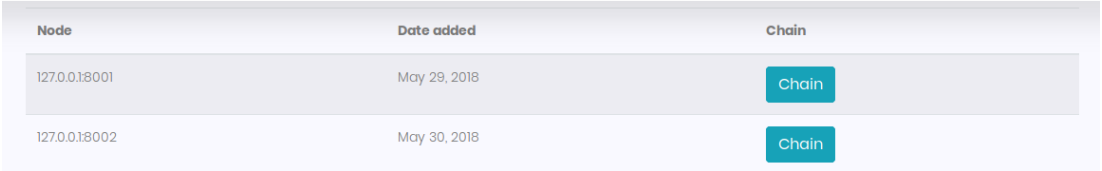
In modern medicine there is a great need for genome sequencing and processing. Genome sequencing price falls much faster than Moore's law and currently human genome can be sequenced for less than 1000$ and that's why many people decide to sequence their genome. The biggest concern with genome sequencing arises from the security and protection of user data. This paper will test the hypothesis that the aforementioned problem of safe storage of sensitive data such as the human genome and the protection of the donor genome's privacy can be solved in the form of a hybrid system using distributed and decentralized blockchain technology and centralized database. The solution is implemented from three parts: blockchain, database and interface. It is anticipated that it will be used by three types of users: genome donors, organizations that buy genomes and conduct research and miners who maintain the blockchain network. Although the solution has its own problems such as database protection and scalability, it shows that this approach of distributed genomic storage can provide greater donor confidence in preserving the integrity and authenticity of stored data.

Keywords: genome, sequencing, genome protection, blockchain, database, distributed, anonymity

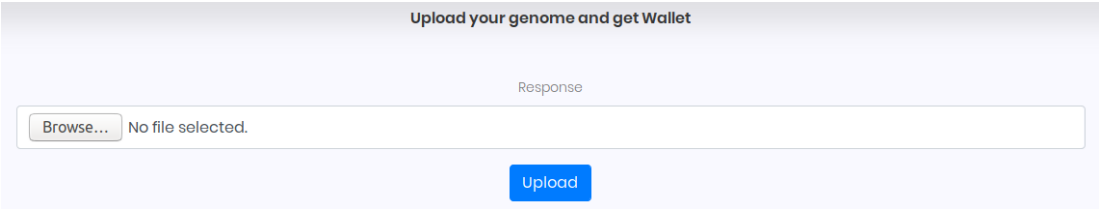
Dodatak A
Funkcije sučelja



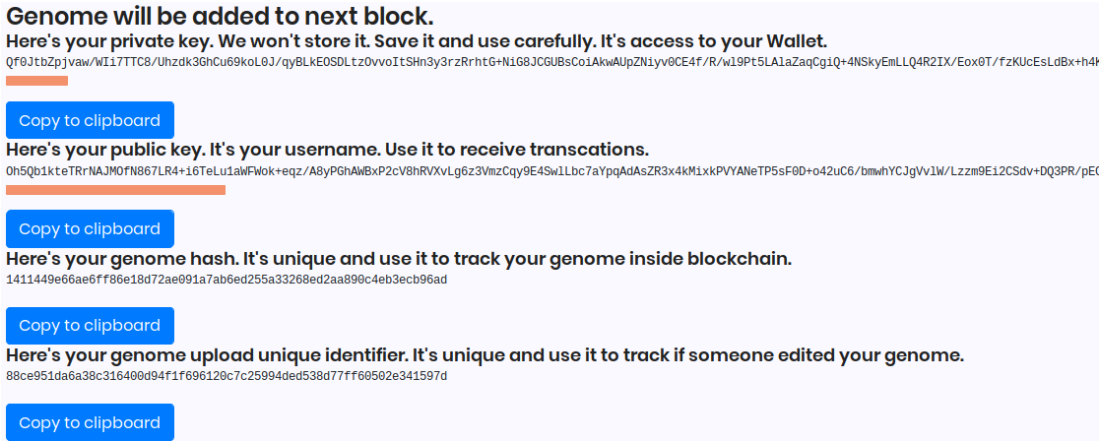
Slika A.1: Početna stranica



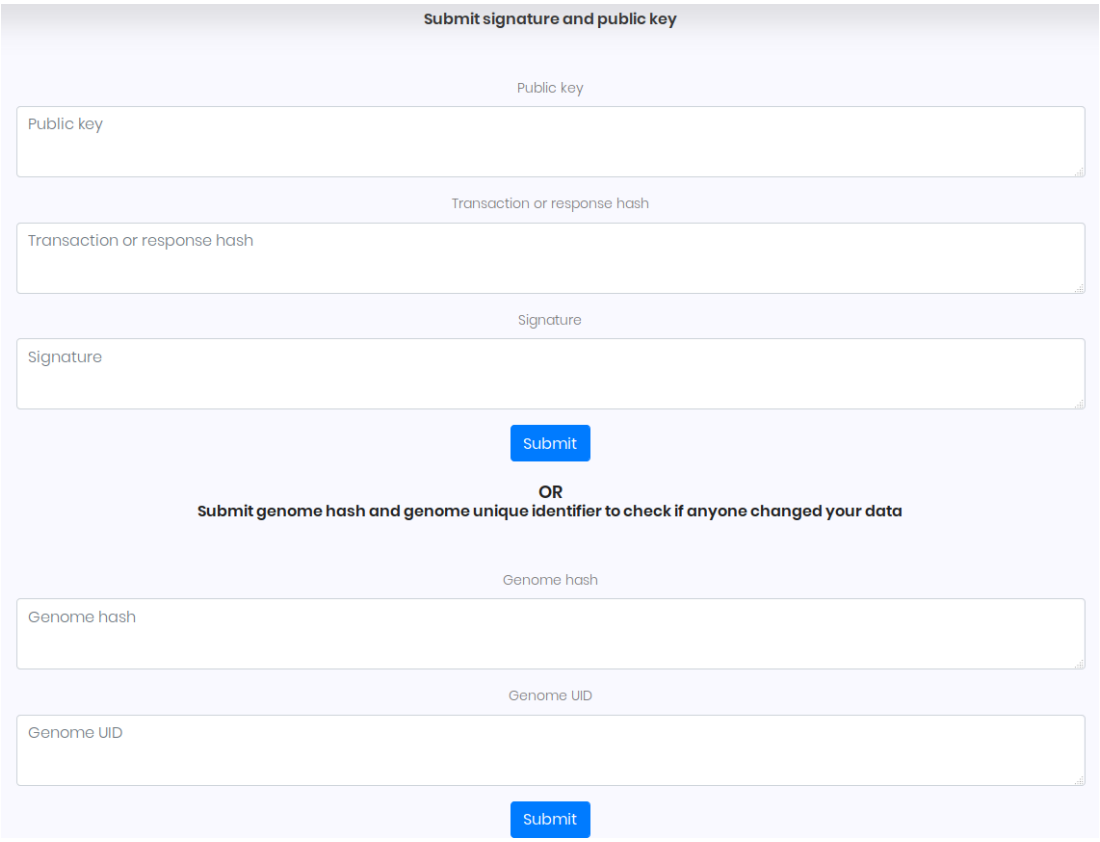
Slika A.2: Popis čvorova koji održavaju mrežu



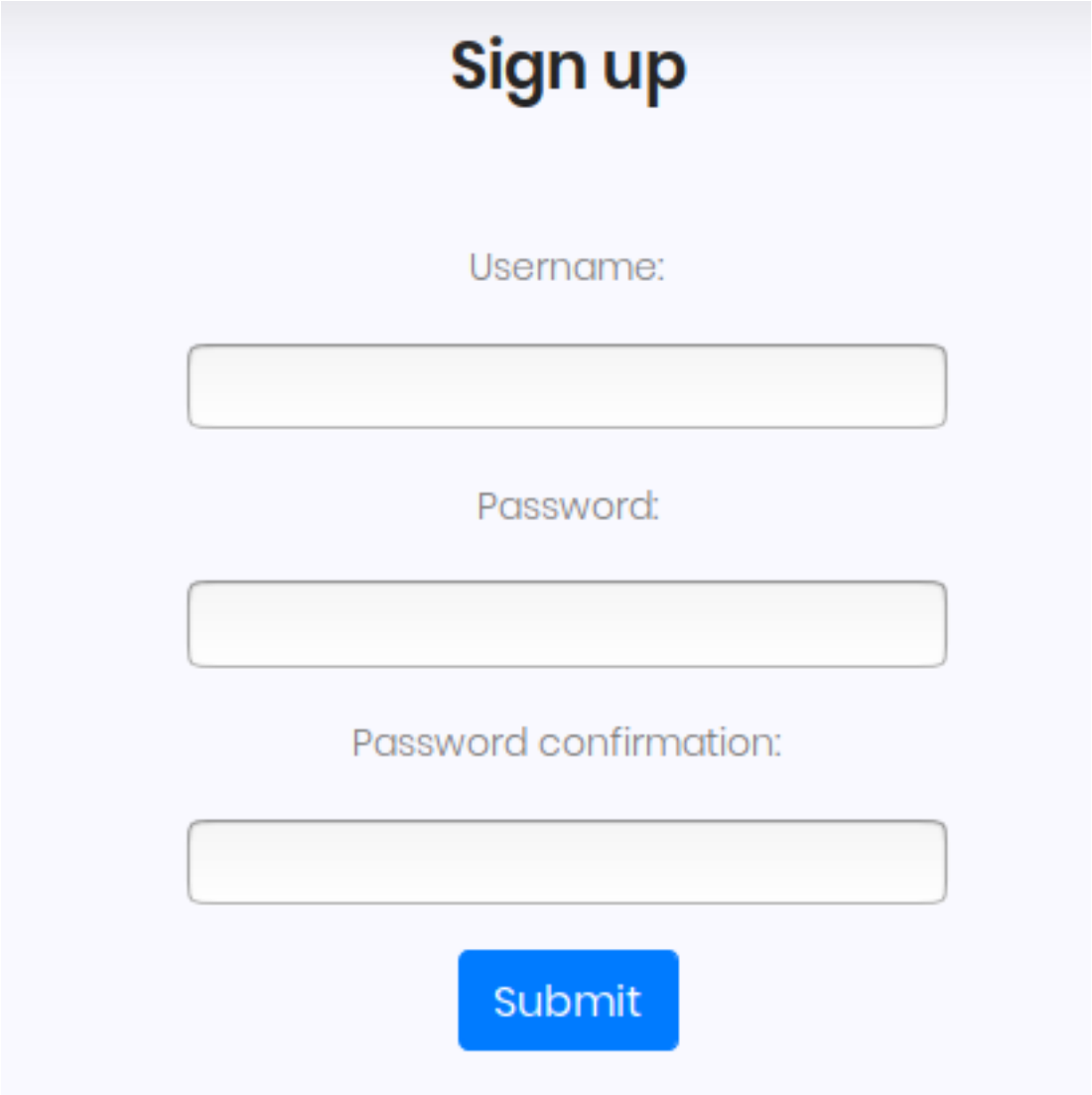
Slika A.3: Obrazac za dodavanje genoma u sustav



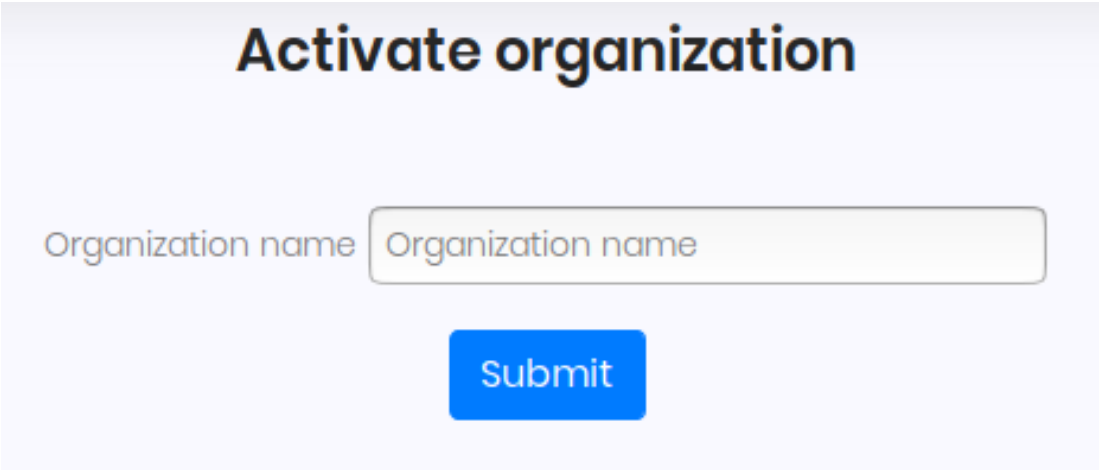
Slika A.4: Po uspješnom dodavanju donor dobije ključeve i sažetke genoma



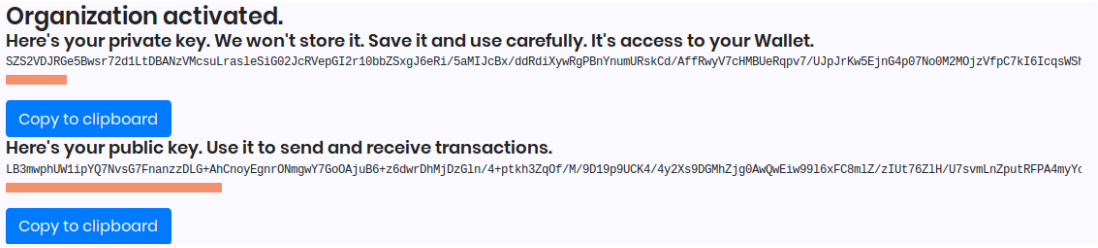
Slika A.5: Stranica za verifikaciju transakcija, odgovora i genoma



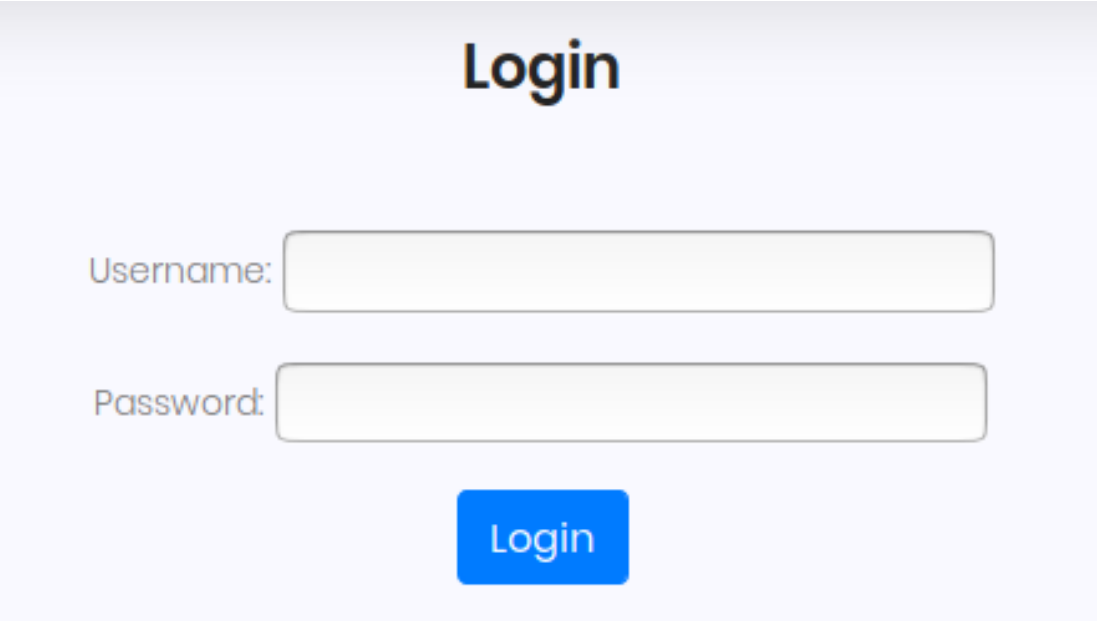
Slika A.6: Obrazac za registraciju organizacije



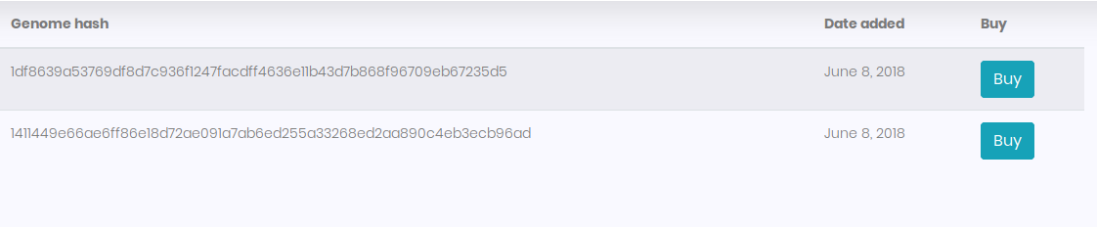
Slika A.7: Obrazac za aktivaciju organizacije



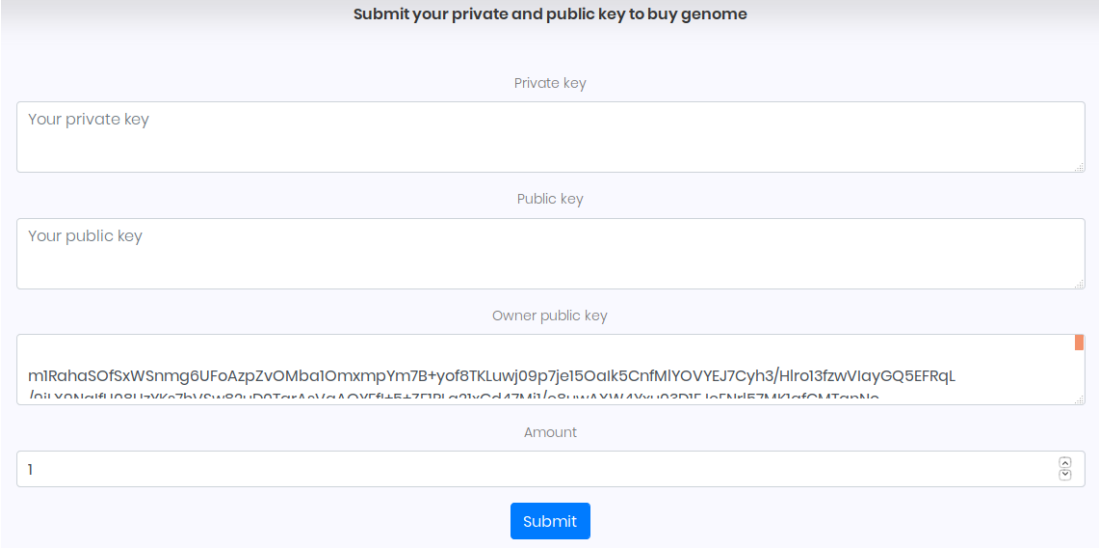
Slika A.8: Po uspješnoj aktivaciji, organizaciji se na sučelju prikazuju ključevi



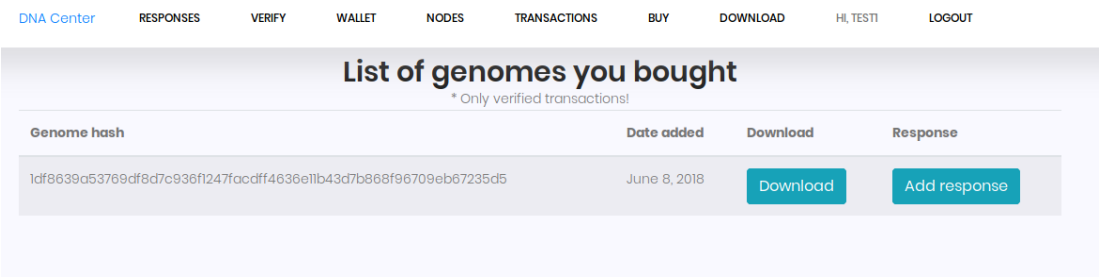
Slika A.9: Obrazac za prijavu organizacije u sustav



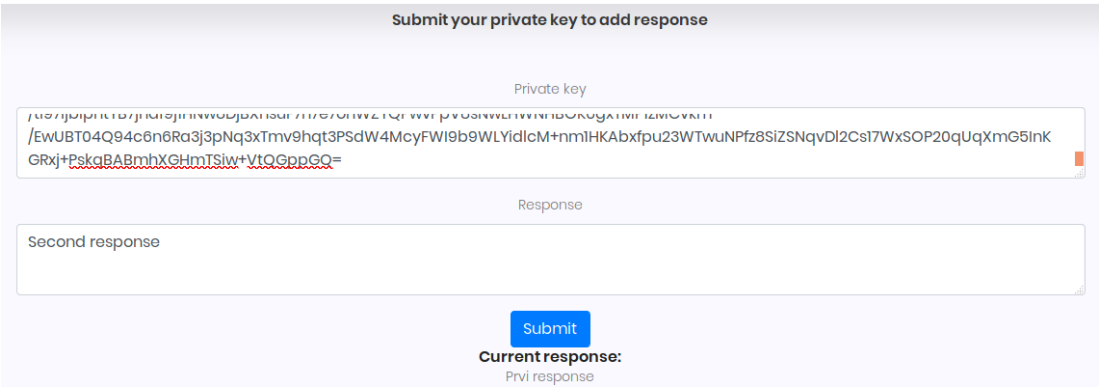
Slika A.10: Stranica s popisom dostupnih genoma za kupovinu



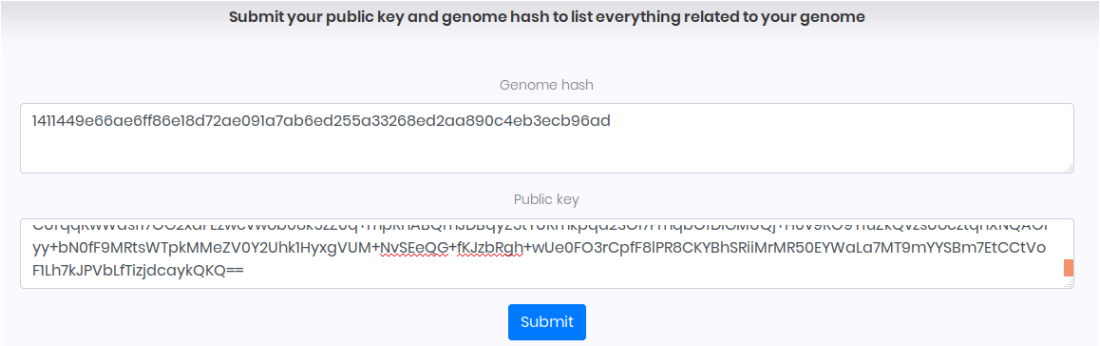
Slika A.11: Obrazac za kupovinu genoma



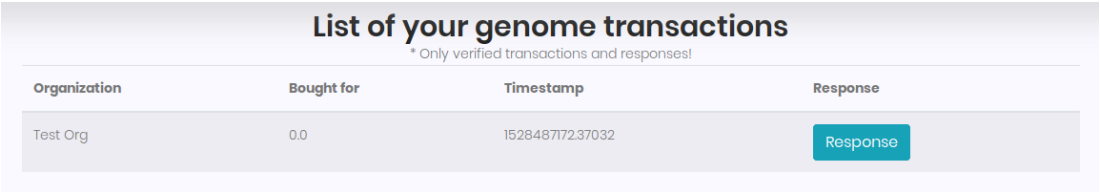
Slika A.12: Stranica s popisom kupljenih genoma



Slika A.13: Obrazac za dodavanje odgovora na genom



Slika A.14: Obrazac za dohvat svih informacija i transakcija vezanih uz genom



Slika A.15: Stranica s popisom svih transakcija i odgovora vezanih uz određeni genom

Slika A.16: Stranica za pregled jednog odgovora