Univerza v Ljubljani
Fakulteta za računalništvo in informatiko

Žan Kusterle

Odprta moderacija za demokratično preverjanje informacij

diplomsko delo

visokošolski strokovni študijski program
prve stopnje
računalništvo in informatika

Mentor: prof. dr. Janez Demšar

Ljubljana, 2018

Besedilo je oblikovano z urejevalnikom besedil LaTeX.

Fakulteta za računalništvo in informatiko izdaja naslednjo nalogo:

Tematika dela:

Spletni mediji imajo v primerjavi s tradicionalnimi zelo nizke stroške obratovanja in omogočajo objavljanje vsebin vsakemu uporabniku. To omogoča preprostejši dostop do ogromne količine informacij, prineslo pa je tudi poplavo vsebin vprašljive kakovosti. Avtomatsko preverjanje dejstev s trenutno tehnologijo (še) ni možno.

Možna rešitev je uporaba modrosti množic. Razvijte sistem, ki bo na preprost način omogočal vsakemu uporabniku, da ocenjuje resničnost dejstev v besedilih, objavljenih na spletu, in argumentira svojo oceno, hkrati pa vidi ocene in argumente drugih uporabnikov.

Kazalo

Povzetek

Abstract

1 Uvod **1**

2 Platforma za hrambo podpisanih podatkov **5**

2.1 Seznam zaupanja 7

2.2 Hranjenje podatkov 9

2.3 Tehnična izvedba 11

3 Tekoča demokracija za preverjanje trditev in anotacijo spleta **17**

3.1 Delegacije 18

3.2 Ocenjevanje 19

3.3 Povezovanje 26

3.4 Tehnična izvedba 31

4 Zaključek **37**

Literatura **41**

Seznam uporabljenih terminov

angleško	**slovensko**
trust metric	metrika zaupanja
whitelist	seznam zaupanja
liquid democracy	tekoča demokracija
cherry picking	selektivna izbira informacij
bias	pristranskost
confirmation bias	potrditvena pristranskost
cognitive dissonance	kognitivna disonanca
echo chamber	skupnost z enotnim prepričanjem
client	odjemalec
local storage	lokalna shramba
hash	zgoščena vrednost
like/dislike	všeček/nevšeček
content aggregator	združevalnik vsebine
ban	prepoved dostopa
HTML tag	značka HTML
domain specific language	domensko specifičen jezik

Povzetek

Naslov: Odprta moderacija za demokratično preverjanje informacij

Avtor: Žan Kusterle

Internet nam ponuja informacije v praktično neomejenih količinah. Zaradi tega se pojavi problem, da je relativno težko vedeti, ali je prebrano besedilo resnično ter nepristransko. Priljubljene platforme za deljenje vsebin uporabljajo priporočilne algoritme, ki predlagajo vsebino, katera nam bo najverjetneje všeč, ne glede na to, ali je potencialno zavajajoča. Ta problem je posebej opazen pri vsebinah s kontroverznimi trditvami, kjer večina gledalcev pripada določeni skupini, katere prepričanja ne odražajo prepričanj splošne populacije. Verodostojnost razmerja všečkov/nevšečkov je zato majhna. Zelo enostavno je tudi pripraviti zavajajočo vsebino s selektivnim izbiranjem informacij, ki potrjujejo vnaprej določena prepričanja, ignorirati pa druge relevantne informacije. Delo raziskuje možnosti za zmanjšanje teh problemov z uporabo odprte moderacije in preproste tekoče demokracije z možnostjo povezovanja rezultatov in vsebin na spletu. Predstavljen je dodatek za brskalnik Chrome, ki nam omogoča, da vidimo relevantne informacije na pravem mestu ob pravem času. Ciljnim uporabniki so ljudje, ki jim je objektivna resnica bolj pomembna, kot pa slišati stvari, s katerimi se že strinjajo.

Ključne besede: seznam zaupanja, preverjanje informacij, tekoča demokracija.

Abstract

Title: Open Moderation for Democratic Fact Checking

Author: Žan Kusterle

We consume a lot of information on the web, yet it is relatively hard to know whether news we read are true and unbiased. Mainstream content sharing platforms like YouTube implement recommendation algorithms that ensure the user is most likely to agree with what (s)he sees. Content that contains controversial claims often suffers from sampling bias that skews their like-dislike ratio. This shows that such platforms are rather useless to determine whether something is true or not. It is also easy to cherry pick facts that favour one side of the issue. This work presents a solution for mitigating these problems by using open moderation and a simple variation of liquid democracy with linking capabilities built on top and a Google Chrome extension that shows the relevant information when needed. Target users are people who value truth more than hearing what they already believe.

Keywords: whitelist, fact checking, liquid democracy.

Poglavje 1

Uvod

Ljudje so tekom evolucije večinoma živeli v manjših skupinah, kjer so se informacije širile samo med njenimi pripadniki. Nismo imeli načinov objektivnega preverjanja informacij, zato smo dali večjo težo informacijam, ki smo jih dobili od ljudi, ki so nam bili blizu in smo jim bolj zaupali. Večjo težo smo dajali tudi informacijam starejših ljudi, saj se je okolje spreminjalo bistveno počasneje kot danes in so bile njihove izkušnje vredne več. Tudi danes nam je zato dostikrat bolj pomembno kdo nekaj reče, kot kaj reče. Okolje v katerem živimo, želimo razumeti čim bolje, zato nas je nagonsko strah stvari, ki jih ne razumemo. Velikokrat imamo raje preproste odgovore na kompleksna vprašanja, kot da bi si priznali, da nekaterih stvari ne vemo. Bolj kot resnica so nam pomembne konkretne pozitivne ali negativne posledice, ki nam jih neko prepričanje prinese. Na vsak način želimo, da so naša prepričanja med seboj dosledna. Nedoslednost v naših prepričanjih nam povzroča nelagodje, ki mu pravimo kognitivna disonanca (*angl. cognitive dissonance*) [28]. Zato vse nove podatke interpretiramo na podlagi svojih obstoječih prepričanj. Večjo težo damo informacijam, ki naša prepričanja podpirajo, velikokrat pa ignoriramo informacije, ki so z našimi prepričanji v nasprotju. Temu pojavu pravimo potrditvena pristranskost (*angl. confirmation bias*) [29].

Internet nam ponuja informacije v praktično neomejenih količinah. Velika večina uporabnikov ne preverja informacij ali pa to počnejo redko. Odločitve,

ki jih sprejemamo na podlagi objektivnih dejstev, so boljše za delovanje sodobne družbe, zato je dostop do zanesljivih informacij ključnega pomena. Preverjanje splošnih trditev je zahteven proces, saj zanj po navadi potrebujemo odgovore na veliko več bolj podrobnih podvprašanj. Najbolj preproste in splošne trditve je iz tega razloga najtežje preveriti. Poleg tega, da vemo, ali je nekaj res, je pomembno tudi, kakšno težo damo različnim problemom. Velikokrat se ljudje strinjajo, da nek pojav predstavlja problem, vendar se ne strinjajo o pomembnosti tega problema. Nekateri so za preprečitev globalnega segrevanja pripravljeni žrtvovati gospodarsko rast, drugim pa se zdi, da materialno udobje, ki ga prinese gospodarska rast, odtehta negativne posledice globalnega segrevanja. V moderni družbi se prek interneta lahko povežemo z mnogimi ljudmi. Danes imamo več moči izbirati, v katerih skupnostih želimo sodelovati. To nam da veliko več intelektualne svobode, vendar ima to zaradi naših evolucijskih nagonov mnogokrat negativne posledice. Iz evolucijskih razlogov, opisanih v prvem odstavku, na spletu velikokrat raje sodelujemo v skupinah, ki potrjujejo naša že ustvarjena prepričanja. Zato na spletu velikokrat nastanejo skupnosti, kjer se med seboj vsi strinjajo, kljub temu, da njihova mnenja odstopajo od splošne populacije (*angl. echo chamber*). V skupnosti namenjeni proučevanju razlogov, zakaj je evolucija teorija zarote, se bo večina uporabnikov s tem strinjala. Z branjem objav na spletni strani take skupnosti zato ne bomo našli dokazov za evolucijo, naše prepričanje, da je evolucija laž, pa se bo tako samo še utrjevalo. Dostikrat tudi iščemo odgovore na napačna vprašanja, ki vsebujejo naše predpostavke. Namesto, da z iskalnikom iščemo 'zakaj je Zemlja ravna', je boljše vprašanje 'kakšne oblike je Zemlja'. S tem veliko lažje najdemo vsebino, ki bo objektivna, kot le vsebino, ki potrjuje naša prepričanja. Naše instinkte izkoriščajo tudi mediji, ki se zavedajo iracionalnih prepričanj in ideološke usmerjenosti svoje publike. Zato so finančno motivirani svojim bralcem predstaviti dejstva, za katera vedo, da so v skladu z njihovimi prepričanji. S tem dosežejo, da bralci radi berejo njihovo vsebino in tako z oglaševanjem zaslužijo več.

V diplomskem delu predstavljamo platformo, ki poenostavi preverjanje

informacij na spletu. Platforma je uporabnikom na voljo v obliki razširitve za brskalnik. Gradi na načelih tekoče demokracije in omogoča ocenjevanje zanesljivosti poljubnih trditev in vsebine na internetu. Nima centralnih moderatorjev, ki določajo, katero vsebino vidimo, ampak nam mogoča, da se sami odločimo, komu zaupamo. Zaupamo lahko posameznim uporabnikom ali pa si izberemo že narejen seznam uporabnikov, ki jih smatramo za vredne zaupanja. Seznam uporabnikov lahko naredi kdor koli in ga deli z drugimi. Omogoča nam tudi medsebojno povezovanje različnih anket. Ankete lahko dodajamo tudi k poljubnem besedilu na poljubni spletni strani. Programska koda našega dela je odprta in vidna vsem. Objavljena je na GitHub-u pod licenco MIT [13]. Na naši spletni strani [12] je vidna kratka demonstracija delovanja opisane platforme. Glavno okno razširitve je vidno na sliki 1.1.

Delo je razdeljeno na dva dela. V naslednjem poglavju opisujemo način shranjevanja podatkov in sezname zaupanja. Drugi del opisuje delovanje tekoče demokracije in dodatek za brskalnik Chrome, ki smo ga razvili v ta namen. Oba dela se končata z opisom tehnične izvedbe.

Slika 1.1: Prikaz ankete znotraj razširitve za brskalnik.

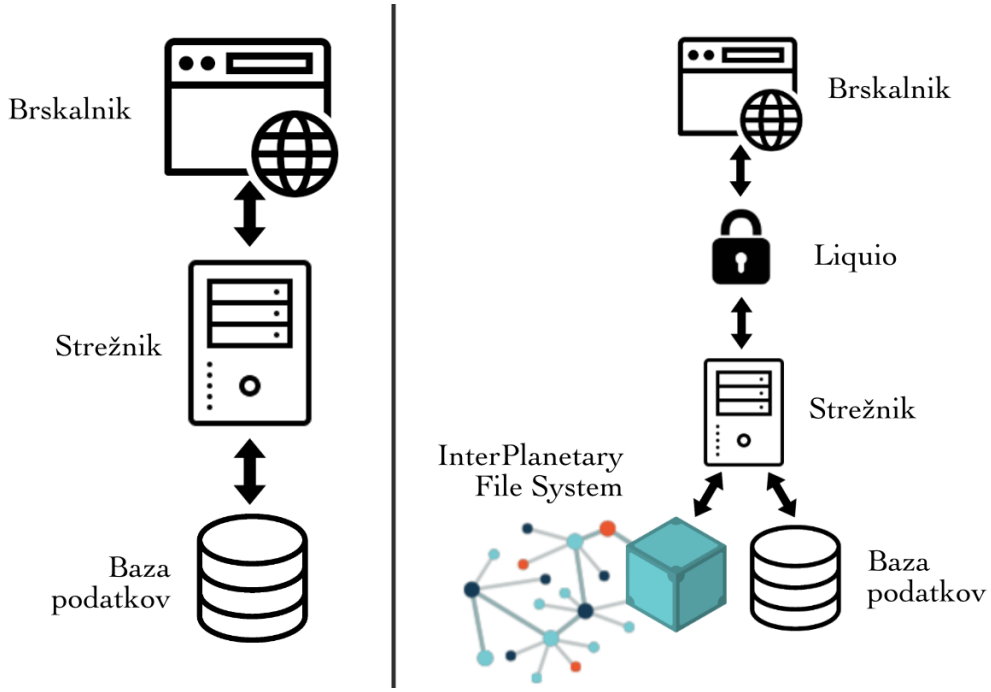
Poglavje 2

Platforma za hrambo podpisanih podatkov

Na internetu obstaja ogromno platform, kjer lahko uporabniki objavljajo vsebino, ki je nato vidna vsem ostalim obiskovalcem. Na takih platformah lahko vsebino popačimo tako, da si ustvarimo več uporabniških računov [16]. Za reševanje tega problema obstoječe platforme za registracijo pogosto zahtevajo e-naslov. Vsebino, ki jo ustvarijo uporabniki večina takih platform shranjuje centralizirano na lastnem strežniku, ki je v lasti posameznika ali podjetja. S tem imajo lastniki platforme popolno moč odločanja o tem, katera vsebina bo prikazana obiskovalcem. Z uporabo priporočevalnih algoritmov lahko promovirajo vsebino, ki se jim zdi primerna. Vsebino, ki se jim zdi neprimerna, pa lahko lahko odstranijo z moderacijo. Moderacija ponavadi odstranjuje le sporno vsebino, ne pa tudi vsebine nizke kvalitete.

Predstavljamo platformo kjer si lahko sami izberemo uporabnike, katerih vsebino želimo videti. Vsebino lahko torej objavlja vsak uporabnik, vendar pa ni nujno, da bo ta vsebina vidna tudi ostalim uporabnikom platforme. S tem demokratiziramo moderacijo, kar pomeni, da lahko različni uporabniki vidijo različno vsebino. To ima lahko tudi negativne posledice, saj imajo končni uporabniki možnost zaupati pristranskim uporabnikom. Vsebina je digitalno podpisana in se shranjuje na IPFS (InterPlanetary File System)

[10]. S tem zagotovimo, da je spreminjanje in odstranjevanje vsebine možno le, če poznamo geslo določenega uporabnika. Zato je platforma odporna na cenzuro. Vsebina na IPFS-ju je javna, vozlišča pa niso finančno motivirana oddajati vsebino, ki ima nizko uporabno vrednost glede na število bajtov. Naša platforma torej ni primerna, če želimo, da vsi obiskovalci vidijo enako vsebino. Primerjava arhitektur je vidna na diagramu 2.1.



Slika 2.1: Arhitektura klasične spletne aplikacije (levo) in naše platforme (desno).

Platforma je primerna za hranjenje ocen in komentarjev, s čimer se ukvarjamo v drugem delu tega diplomskega dela. Zanimiva uporaba naše platforme bi bila lahko tudi alternativa Wikipediji, kjer bi vsak videl drugačne podatke glede na to, komu zaupa [26].

2.1 Seznam zaupanja

Seznam zaupanja je množica uporabnikov, za katere mislimo, da objavljajo zanesljive informacije. Predstavljen je z datoteko formata HTML (Hypertext Markup Language), ki vsebuje vsaj eno HTML značko z atributom 'username'. Vrednost atributa je poljubno uporabniško ime uporabnika, katerega podatke želimo videti. Sezname zaupanja zato predstavimo z URL naslovom. Nov seznam tako lahko ustvari vsak, ki zna na internetu objaviti datoteko. Poleg uporabnikov na seznamu zaupanja so nam vedno vidni podatki trenutno prijavljenega uporabnika ter podatki uporabnikov, ki jim prijavljen uporabnik zaupa. Vsak uporabnik si lahko izbere nek obstoječ seznam zaupanja ali pa ustvari svojega.

Platforma ima privzeto nastavljenih več že narejenih seznamov zaupanja za uporabnike, ki si jih ne spremenijo. Privzeti seznami zaupanja so edini način, ki nam kot lastnikom platforme omogoča moderiranje vsebine. Privzeto so nastavljeni naslednji seznami zaupanja:

- Potrjevanje uporabnikov z metriko zaupanja.
- Potrjevanje uporabnikov z e-naslovom. Ta seznam ponuja podoben način registracije, kot mnogo obstoječih platforme in je trenutno še v fazi razvoja.
- Potrjevanje uporabnikov s telefonsko številko. Ta seznam trenutno še ne obstaja.

Metriko zaupanja predstavimo z uteženim usmerjenim grafom zaupanja med uporabniki [14]. Vozlišča predstavljajo uporabnike, povezave pa zaupanje med njimi. Vsak uporabnik lahko svoje zaupanje izkaže večim ostalim uporabnikom. Zaupanje je predstavljeno s številom med 0 (nezaupanje) in 1 (zaupanje). Nad tem usmerjenim grafom zaupanja uporabljamo algoritem PageRank [19], ki je biv prvotno razvit za razvrščanje spletnih strani v spletnih iskalnikih. Algoritmu najprej podamo majhen seznam začetnih uporabnikov, ki jim brezpogojno zaupamo. Metriko zaupanja torej lahko

uporabljamo za izračun seznamov zaupanja, ali si z njo le pomagamo. Lahko jo tudi popolnoma ignoriramo in za sestavo seznama zaupanja uporabimo druge vire informacij.

Nemogoče je narediti popoln seznam zaupanja, saj nihče nima dostopa do popolnoma objektivnih dejstev, zato naša platforma omogoča ustvarjanje seznamov vsakomur. Seznami zaupanja so inherentno subjektivni in so šibka točka celotne platforme, zato lahko na njih gledamo kot na dvorezen meč. Uporabnikom je omogočeno, da si izberejo izjemno popačen vzorec podatkov, bolj popačen, kot če bi bili prisiljeni uporabljati privzet seznam zaupanja. Platforma je torej namenjena predvsem posameznikom, ki jim je resnica najvišja prioriteta. To so redkejši ljudje, ki jim je bolj pomembno, kaj je res, kot pa slišati stvari, s katerimi se že strinjajo. Upamo, da bo z zmanjševanjem ovir za sodelovanje pri moderiranju na internetu veliko različnih seznamov zaupanja. Predvidevamo, da jih bo večina zelo pristranskih, kljub temu pa upamo, da bodo na voljo tudi bolj kakovostni seznami.

Vsi uporabniki so po registraciji popolnoma anonimni, znan je le njihov naslov IP. Uporabniki imajo po prijavi možnost, da si nastavijo javno prikazno ime in eno ali več spletnih strani (npr. profile socialnih omrežij, osebne spletne strani ...). Spletno stran je možno dodati samo v primeru, če ta vsebuje uporabnikovo uporabniško ime. S tem zagotovimo, da ima uporabnik nadzor nad vsebino spletne strani. To sicer še ne pomeni, da je uporabnik avtor spletne strani. Lahko gre namreč za spletno stran, kjer lahko vsebino objavlja vsak.

Obstaja možnost podkupovanja avtorjev seznamov zaupanja, da nas dodajo na njihov seznam. Podkupimo lahko tudi druge uporabnike, da nam izkažejo zaupanje, s čimer lahko posredno pridemo na sezname zaupanja, ki so ustvarjeni na podlagi metrike zaupanja.

2.2 Hranjenje podatkov

Za ustvarjanje novega uporabnika programsko generiramo naključno geslo v obliki 13 besed, ki jih lahko zapišemo na papir ali jih shranimo na svojem računalniku. Za razliko od večine sistemov za avtentikacijo, v naši platformi uporabniško ime ni potrebno za prijavo, zadostuje nam zgolj geslo. Ob prijavi se iz gesla deterministično generira par javnega in zasebnega ključa (v nadaljevanju par ključev). Besede in zasebni ključ se iz brskalnika nikoli ne prenesejo na strežnik. Uporabniško ime dobimo tako, da vzamemo zgoščeno vrednost javnega ključa in pogledamo samo prvih 16 znakov. To pomeni, da je možnost, da bi se dve uporabniški imeni prekrivali, manjša od 10^{-24}. Zasebni ključ se uporablja za digitalno podpisovanje podatkov [4], s čimer je ponarejanje teh podatkov nemogoče, tudi z neposrednim dostopom do podatkovne baze. Za preverjanje podpisov podatkov pa se uporablja javni ključ uporabnika. Iz teh razlogov ob izgubi gesla nimamo niti teoretične možnosti še kdaj podpisati podatke s tistim uporabnikom. V primeru izgube gesla imamo vedno možnost podatke prenesti na novega uporabnika. Iz tega razloga vidimo, da je vrednost uporabniškega računa odvisna samo od tega, v katere sezname zaupanja je vključen 2.1. V primeru da nek uporabnik zanika podpis določenega sporočila vemo, da so bile njegove besede za prijavo ukradene. Takega uporabnika ja najbolje takoj izključiti iz seznamov zaupanja, saj so vsa njegova prihodnja sporočila neverodostojna. Brez podpisovanja seznami zaupanja ne morejo delovati, saj bi vsak lahko dodajal vsebino v imenu poljubnega uporabnika. Podpisani podatki so shranjeni tudi na IPFS, kar zagotovi, da jih je nemogoče izbrisati. Kot avtorji platforme torej nimamo moči ustvarjati lažne podatke ali pa določene podatke izbrisati. Podatke poljubnega uporabnika lahko izvozimo v tekstovno datoteko in jih nato kasneje uvozimo drugam. To uporabnikom olajša dostop in hranjenje svojih podatkov in podatkov drugih uporabnikov. Do zdaj opisana platforma je izjemno generična saj omogoča podpisovanje poljubnih podatkov in ima posledično veliko različnih možnih aplikacij. Primerna je tam, kjer je vsebina vidna vsem, glavna razlika pa je, da si vsak sam izbere avtorje, ki jim zaupa.

Podatki so predstavljeni v obliki JSON (JavaScript object notation). Vsebovati morajo posebni polji ‘key‘ in ‘datetime‘. Datum in čas v polju ‘datetime‘ morata biti zapisana v formatu po standardu ISO 8601. Polje ‘key‘ pa mora biti seznam imen drugih polj. V primeru ima ključ vrednost ‘[’username’]‘, kar pomeni, da če enakemu uporabniku večkrat nastavimo raven zaupanja, se upoštevajo le zadnji podatki. V primeru, da se vrednosti polj v seznamu polja ‘key‘ ujemajo, je podatke mogoče posodobiti z novo verzijo. Strežnik torej nikoli ne vrne več sporočil z istim ključem, ampak samo najnovejšega. Sporočila nam strežnik ponuja za poljubno točko v času. Uporabnik sporočilom čas določi sam, kar pomeni, da lahko ustvarja sporočila tudi v preteklosti. V primeru, da so v našem seznamu zaupanja verodostojni uporabniki, to ni problem. Primeri podatkov so na slikah 2.2, 2.3 in 2.4:

ključ	**vrednost**
key	[’type’]
type	name
name	Janez Novak

Slika 2.2: Podatki, s katerimi si nastavimo javno prikazno ime.

ključ	**vrednost**
key	[’website’]
website	https://novak.janez
show	true

Slika 2.3: Podatki, s katerimi si nastavimo osebno spletno stran.

ključ	vrednost
key	['username']
username	qubybbhnrmicnbeu
trust	1.0

Slika 2.4: Podatki, s katerimi določenemu uporabniškemu imenu izkažemo zaupanje.

2.3 Tehnična izvedba

Ta del je implementiran s preprostim strežnikom, s katerim lahko interaktiramo preko vmesnika, narejenega po metodologiji REST [22]. Strežnik je napisan v jeziku Elixir, ki se prevede v Erlang in teče na virtualnem stroju Erlanga (BEAM) [6] [7]. Uporabljamo ogrodje Phoenix, ki je preprosto in minimalistično ogrodje za programiranje spletnih strežnikov [20]. Uporabljamo tudi knjižnico Ecto, ki ponuja domensko specifičen jezik, s katerim lahko enostavno komuniciramo z bazo podatkov. Knjižnica Plug pa je preprosta, a hkrati zelo elegantna knjižnica, s katero lahko na preprost način definiramo korake po katerih strežnik obdeluje zahteve. Na vsakem koraku se lahko odločimo, ali želimo že vrniti odgovor na zahtevek ali pa zahtevek predamo naslednjemu koraku. Podatke strežnik hrani lokalno v bazo Postgres ter hkrati tudi na IPFS. Vse podatke hranimo v eno samo tabelo, katere glavni stolpci so: *public_key*, *data*, *signature*, *ipfs_hash*. Iz podatkov iz glavnih stolpcev iz performančnih razlogov izpeljemo se nekaj dodatnih stolpcev: username, key, datetime, data_hash. Zanimive alternative za lokalno bazo bi bile tudi NoSQL baze, saj uporabljamo le dve nepovezani tabeli. Glavna prednost SQL baz je zagotavljanje konsistence med več tabelami ter združevanje več tabel v poizvedbah (*angl. join*). Med internetom

in strežnikom je postavljen posrednik Nginx 1.10.3 [17]. Vse skupaj teče na operacijskem sistemu Ubuntu 16.4.3, za varno komunikacijo prek protokola HTTPS pa uporabljamo certifikatno agencijo LetsEncrypt [24] [11].
Naslednji ukazi so uporabni za razvoj in vzdrževanje strežnika:

mix deps.get Namesti vse knjižnice, ki jih strežnik potrebuje za delovanje. Ta ukaz uporabimo, ko kloniramo repozitorij git, če želimo pognati strežnik.

mix phx.server Lokalno požene strežnik in nam omogoča razvoj. Vse spremembe kode takoj postanejo veljavne, ne da bi bilo potrebno ponovno zagnati strežnik.

mix edeliver update production Posodobi oddaljen strežnik z zadnjo verzijo kode na GitHub-u.

mix edeliver restart production Ponovno zažene oddaljen strežnik, s čimer začne delovati posodobljena verzija.

mix edeliver stop production Ugasne oddaljen strežnik.

Strežnik skrbi tudi za računanje privzetega seznama zaupanja iz metrike zaupanja. To periodično počne asinhron proces, ki se trenutno požene vsakih 10 sekund. Iz baze za vsakega uporabnika prebere komu zaupa in s kakšno težo. Nato iz teh podatkov ustvari utežen usmerjen graf, nad katerim požene algoritem PageRank s faktorjem zmanjševanja $d = 0,85$. Uporabimo nastavljiv seznam uporabnikov, ki jim popolnoma zaupamo in jim na začetku damo obratno (recipročno) vrednost velikosti tega seznama. Tako zagotovimo, da je vsota vrednosti vseh vozlišč enaka 1. Vsem ostalim uporabnikom pa damo na začetku vrednost 0. Nato za izračun vrednosti vsakega vozlišča (uporabnika) uporabimo preprost iterativen algoritem z 10 iteracijami. Po tem vzamemo največ 10 milijonov najbolje ocenjenih uporabnikov, ki imajo vrednost, večjo od 0,000000001. Uporabniška imena teh uporabnikov shranimo na disk v HTML datoteko, ki je potem javno dostopna na strežniku.

V nadaljevanju so predstavljene javno vidne funkcionalnosti API-ja z naslovom URL in metodo HTTP ter statusom odgovora. Tako izpišemo seznam vseh uporabnikov z uporabniškim imenom, nastavljenimi identifikacijskimi podatki (ime in spletno stran) ter seznamom drugih uporabnikov, ki jim zaupa:

GET /identities 200


```
{
  "data": [
    {
      "username": "qubybbhnrmicnbeu",
      "identifications": [
        {
          "type": "name",
          "value": "Janez Novak"
        },
        {
          "type": "website",
          "value": "https://twitter.com/janeznovak"
        },
        {
          "type": "website",
          "value": "http://janez.novak"
        }
      ],
      "trust_usernames": {
        "jztqktsdtofhwkhq": 1.0,
        "rknrnfutyematmgr": 0.5
      }
    },
    {
      "username": "jztqktsdtofhwkhq",
      "identifications": [
```

```
        { "type": "name", "value": "John Smith" }
      ]
    }
  ]
}
```

Izpišemo lahko seznam uporabniških imen na seznamu zaupanja, ki se nahaja na URL naslovu, katerega podamo s parametrom :url:

GET /whitelist/:url 200

```
{
  "data": [
    "qubybbhnrmicnbeu",
    "jztqktsdtofhwkhq",
    "rknrnfutyematmgr"
  ]
}
```

Poljubne podatke lahko podpišemo takole:

POST /messages 204

public_key Base 64 kodiran javni ključ;

message JSON objekt, ki mora vsebovati ključa 'key' in 'datetime';

signature Base 64 kodiran podpis sporočila.

```
{
  "public_key": "kp8Oy6nM3vfYvhoPLzxYIw...",
  "signature": "cF2BGn2j00wOjVawUwU2cH9...",
```

```
  "message": {
    "key":["identification"],
    "identification": "name",
    "value": "Janez Novak",
    "datetime": "2018-09-07T19:04:58.518Z"
  }
}
```

Besede, ki jih uporabljamo za podpisovanje podatkov, ponujajo več kot 128 bitov entropije. Za generacijo besed ter pretvarjanje besed v binarno obliko se uporablja implementacija iz BIP39 [15]. Celotna registracija (generiranje besed) se izvede izključno pri odjemalcu. Za podpisovanje podatkov se uporablja algoritem Ed25519, ki temelji na shemi EdDSA (Edwards-curve Digital Signature Algorithm) [5]. To je računsko varen algoritem, saj za napad z grobo silo potrebujemo povprečno 2^{140} operacij. Algoritem je primeren tudi iz drugih razlogov:

- majhna velikost javnega ključa (32 bajtov) in podpisa (64 bajtov),
- hitra generacija para ključev,
- hitro podpisovanje podatkov in preverjanje podpisov. Na povprečnem računalniku lahko preverimo 30 000 podpisov na sekundo.

Za branje shranjenih podatkov strežniku pošljemo naslednje podatke:

GET /messages 200

whitelist_url URL metrike zaupanja;

usernames seznam z vejico ločenih uporabniških imen, ki jim zaupamo.

datetime Datum v formatu ISO 8601, s katerim povemo, za katero točko v času želimo podatke.


```
{
  "data": [{
    "signature": "...",
    "public_key": "...",
    "ipfs_hash": "...",
    "username": "ldndlkoyicxincxw",
    "key": "Zemlja je ravna False-True",
    "datetime": "2018-09-10T20:05:29.673Z",
    "data": {
      "key": ["vote", "title", "unit"],
      "datetime": "2018-09-10T20:05:29.673Z",
      "title": "Something",
      "unit": "Unreliable-Reliable",
      "choice": 1
    }
  }]
}
```

Poglavje 3

Tekoča demokracija za preverjanje trditev in anotacijo spleta

V tem poglavju je opisana razširitev za brskalnike, ki temelji na tekoči demokraciji. Naša razširitev nudi možnost ocenjevanja poljubnega besedila. To besedilo je lahko trditev, URL spletne strani, opis kvantitativne veličine, naslov filma ... Ocenjujemo lahko tudi besedilo znotraj spletne strani. Stvarem, ki jih ocenjujemo, rečemo entitete. Razširitev shranjuje podatke o ocenah ter povezavah med entitetami na platformo, opisano v prejšnjem poglavju. Končnim uporabnikom ponuja hitre informacije o zanesljivosti spletnih strani ter posameznih trditev znotraj le-teh. Omogoča ocenjevanje poljubne entitete in povezovanje entitet med seboj. Na ta način nam razširitev omogoča ocenjevanje zanesljivosti določene spletne strani. Entitete lahko povežemo tudi na poljubno besedilo znotraj spletne strani. Razširitev nam nato pomaga, da vidimo relevantne informacije na pravem mestu ob pravem času. Razširitev nam omogoča, da rezultate vidimo v kontekstu spletne strani in nam zato ni treba odpirati dodatnih zavihkov v brskalniku. Vsa funkcionalnost je hitro dostopna iz razširitve. Na podlagi teh razlogov se nam je ločena spletna aplikacija zdela nepotrebna.

Tekoča demokracija združuje prednosti direktne in reprezentativne demokracije in s tem ustvari sistem, v katerem lahko ljudje direktno ocenjujejo entitete ali pa svoj glas delegirajo ljudem, ki jim zaupajo. Trenutno smo informacije sposobni preverjati samo ljudje, ki pa smo inherentno pristranski. Edini potencialno boljši način za preverjanje informacij je umetna inteligenca, ki pa danes še ni dovolj napredna, da bi lahko preverjala resničnost informacij. Naša platforma ni namenjena glasovanjem, na podlagi katerih se sprejema konkretne odločitve. Razlog za to je, da ne ponuja enakih rezultatov vsem uporabnikom, saj ima vsak lahko drugačen seznam zaupanja 2.1. V primeru, ko gre za glasovanje o praktičnih odločitvah, so tehnologije, ki ponujajo konsenz o trenutnem stanju (npr. Blockchain [3]), po navadi bolj primerne.
Na temeljnem nivoju gre v temu delu diplomskega dela za protokol, ki je definiran s strukturo podatkov, katere uporabniki podpisujejo. Iz tega razloga lahko vsak, ki ima ustrezno znanje, razvije program (razširitev, spletno aplikacijo, program za analizo podatkov ...), kateri uporablja enake podatke. To pomeni, da so ocene, ustvarjene z našo razširitvijo, vidne tudi drugim programom ter obratno.

3.1 Delegacije

Uporabniki lahko svojo volilno moč predajo drugim uporabnikom, ki jim zaupajo. To uporabnikom omogoča alternativno možnost sodelovanja. Uporabnike spodbuja k oceni zgolj entitete, tako da mislijo, da imajo zadostno znanje. S tem se poveča povprečna kakovost ocen na platformi. Delegacije pomagajo novim uporabnikom k enostavnejši udeležbi. S tem, ko svojo oceno razdelijo med več drugih oseb, prav tako prispevajo h končni oceni entitet. V primeru, da neko entiteto ocenimo neposredno, pa so delegacije za tisto entiteto ignorirane in se upošteva naša ocena. Ponujajo torej hibrid med direktno in reprezentativno demokracijo. Delegacije so utežene, kar pomeni, da imajo uporabniki možnost svoj glas neenakomerno razdeliti med več ostalih uporabnikov. Za implementacijo se uporabljajo isti podatki kot

za izkazovanje zaupanja v prvem delu tega diplomskega dela sekcije 2.4. Delegacije so tranzitivne, kar pomeni, da uporabniki ne delegirajo samo svojega glasu, temveč tudi glasove drugih uporabnikov, ki so svoj glas delegirali njim. Na primer, če uporabnik A delegira svoj glas uporabniku B, uporabnik B pa svoj glas delegira uporabniku C, ima uporabnik C volilno moč treh glasov.
Pogoj za učinkovito delegiranje je transparentnost celotne platforme. Nemogoče je delegirati svoj glas nekomu, za kogar ne vemo, kako glasuje. To odpre potencialno možnost podkupovanja za razliko od trenutnih volilnih sistemov, kjer volimo z listom papirja, za katerega nihče ne ve, čigav je. Delegacije lahko kadar koli spremenimo, s čimer spodbujamo in nagrajujejo kakovostno glasovanje. Uporabniki, katerim svoj glas delegira veliko ljudi, vedo, da lahko v primeru nizke kakovosti glasov hitro izgubijo svojo moč.
Namen delegacij je torej olajšati pridobivanje zanesljivih in kakovostnih ocen, zares uporabne pa postanejo šele pri večjem številu uporabnikov. Bolj globinski pregled delegativne demokracije in njenih prednosti je opisan v knjigi LiquidFeedback, ki je brezplačna [27].

3.2 Ocenjevanje

Ocenjevanje vsebine je pogosta funkcionalnost socialnih platform na internetu. YouTube nam omogoča ocenjevanje video posnetkov, IMDb ocenjevanje filmov, Amazon ocenjevanje izdelkov in knjig. Temeljna lastnost sistemov ocenjevanja je prikaz agregirane povprečne ocene, kar končnemu uporabniku poda informacije v bolj zgoščeni obliki. Mnoge razvite države za svoje volitve uporabljajo sisteme, ki nam omogočajo izbiro enega samega kandidata. Na IMDb-ju in mnogih drugih sistemih ocenjevanja pa filme ocenjujemo z oceno med 1 in 10. Če bi na volitvah vse kandidate prav tako ocenili z oceno med 1 in 10, bi s tem svoje želje lahko izražali bolj natančno. Tako bi lahko dvema kandidatoma zaupali približno enako, v trenutnih sistemih pa se moramo odločiti samo za enega. Velikokrat pride do problema, ko sta si dva

kandidata precej podobna in se glasovi volivcev razdelijo med njiju, nato pa zmaga tretji kandidat, ki ga večina ne podpira. Povprečne ocene uporabnikov so sicer na IMDb-ju zelo različne, ocene posameznega uporabnika pa so medsebojno še vedno relativno dosledne.

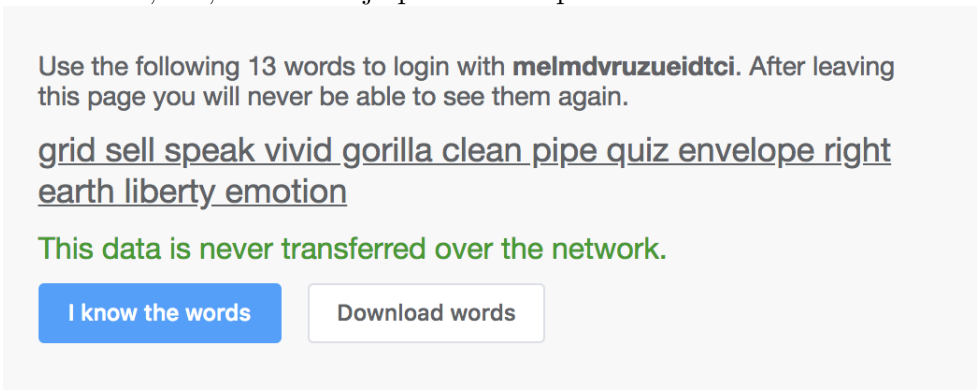
Vsaka entiteta ima določeno enoto, kar nam omogoča, da lahko pravilno izračunamo povprečje. Za računanje povprečja gledamo torej vse ocene neke entitete. Sistem ocenjevanja je izjemno generična rešitev in ima široke aplikacije. Ocenjujemo lahko, na primer, kakovost torrentov ali podnapisov. V ekosistemu NPM [18] ga lahko uporabljamo za preverjanje, ali so knjižnice varne in ne vsebujejo zlonamerne kode.

Entitete znotraj našega seznama zaupanja lahko pregledujemo znotraj razširitve. Razširitev nam ponuja seznam vseh entitet, entitet za specifično domeno ter možnost iskanja. To nam omogoča, da našo razširitev uporabljamo kot združevalnik internetnih vsebin (npr. novic). Pri velikem številu entitet je zelo pomembno, v kakšnem vrstnem redu jih prikažemo. Privzeto so entitete urejene po številu skupne volilni moči njenih ocen. Imamo tudi možnost urejanja po času, kar pomeni, da damo starejšim ocenam manjšo težo. To storimo tako, da določimo trajanje razpolovitve teže ocene [8]. Če razpolovitev traja dolgo, bodo nove ocene imele majhno prednost pred starejšimi. Če je trajanje 0, bo najnovejša ocena posamezne entitete edini faktor pri urejanju.

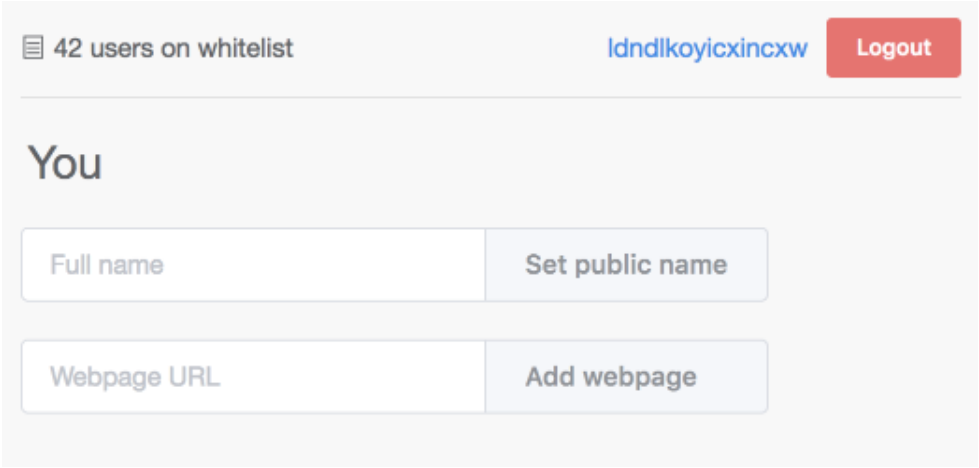
Večina omenjenih platform ponuja tudi možnost komentiranja. V primerjavi z direktnim ocenjevanjem nam komentarji omogočajo, da vnesemo bolj bogate informacije. Slaba stran tega pa je, da je (trenutno) nemogoče izračunati povprečen komentar. Podobno kot pri združevalnikih vsebine je najbolj bistvena lastnost sistemov komentiranja, v katerem vrstnem redu prikažemo komentarje. Za določanje vrstnega reda komentarjev se pogosto uporablja sistem ocenjevanja kakovosti komentarjev. Naša razširitev za ocenjevanje komentarjev in računanje vrstnega reda uporablja isti sistem kot za ocenjevanje trditev. S tem so komentarji del entitete, enako kot naslov in enota. Predstavljeni so s seznamom, kar omogoča neskončno globino odgovorov na

komentarje. Sistem komentiranja ima široko uporabnost, z njim bi lahko določali definicije za slengovske besede, podobno kot to počne Urban Dictionary.

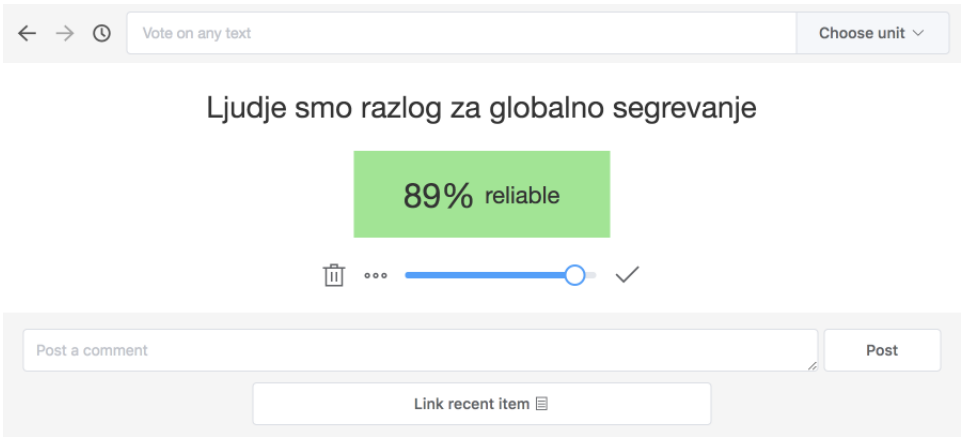
Na slikah 3.1, 3.2, 3.3 in 3.4 je prikazana uporaba razširitve:



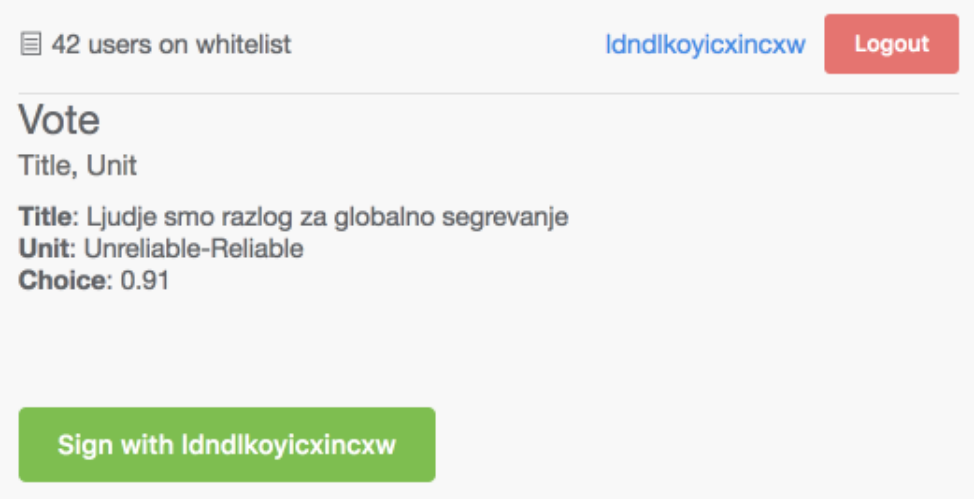
Slika 3.1: Generirano geslo, ki predstavlja novega uporabnika.



Slika 3.2: Svojim anonimnim uporabnikom lahko nastavimo tudi javno prikazno ime ter dodamo eno ali več spletnih strani.



Slika 3.3: Prikaz entitete s spektralno enoto.



Slika 3.4: Dialog za podpisovanje podatkov.

Za ocenjevanje zanesljivosti trditev, filmov, spletnih strani, komentarjev in v splošnem verjetnosti se uporabljajo spektralne enote. Trenutno so na voljo naslednje:

podpisana vrednost	**slovenski prevod**
Ratio	delež
Rating	ocena
False-True	resničnost
Unreliable-Reliable	zanesljivost
Anecdotal-Scientific	znanstvenost pristopa
Disagree-Agree	strinjanje
Irrelevant-Relevant	relevantnost
Useless-Useful	uporabnost

Za ocenjevanje verjetnosti trditev in zanesljivosti spletnih strani se uporabljajo *spektralne enote*. Vrednost ocene s spektralno enoto mora nujno biti na intervalu [0, 1]. Spektralne enote vsebujejo pomišljaj, ki ločuje dve nasprotni strani nekega spektra. Leva stran je predstavljena z vrednostjo 0, desna stran pa z 1. Ker je interval vrednosti majhen, lahko prikažemo aritmetično povprečje ocen, saj nimamo ekstremnih vrednosti, ki bi ga lahko popačile. Ocene iz drugih obstoječih virov lahko vedno preslikamo na ta interval: všečkom določimo vrednost 1, nevšečkom vrednost 0, za poljubne ocene pa uporabimo enačbo

$$(\text{ocena} - \min)/(\max - \min) \quad \textit{oziroma} \quad (\text{IMDb ocena} - 1)/9$$

Za podajanje približkov poljubnih veličin se uporabljajo kvantitativne enote. Od spektralnih enot se razlikujejo po tem, da omogočajo izbiro neomejene numerične vrednosti, za izračun povprečja pa se zato uporablja mediana. Trenutno so na voljo naslednje:

podpisana vrednost	**slovenski prevod**
Count	števec
Year(AD)	leto
Temperature(°C)	temperatura v stopinjah Celzija
Money(EUR)	denar v evrih
Money(USD)	denar v ameriških dolarjih
Length(m)	dolžina v metrih

Podatki za ocenjevanje različnih entitet so prikazani na slikah 3.5, 3.6, 3.7, 3.8, 3.9 in 3.10:

ključ	**vrednost**
title	Ljudje smo razlog za globalno segrevanje
unit	Unreliable-Reliable
choice	0.01

Slika 3.5: Ocena trditve s spektralno enoto.

ključ	**vrednost**
title	Število prebivalcev na Zemlji
unit	Count
choice	7000000000

Slika 3.6: Ocena števila s kvantitativno enoto.

ključ	**vrednost**
title	`https://en.wikipedia.org/wiki/Flat_Earth`
unit	Unreliable-Reliable
choice	0.93

Slika 3.7: Ocena spletne strani.

ključ	vrednost
title	`https://en.wikipedia.org/wiki/Flat_Earth`
anchor	Aristotle provided evidence for the spherical shape of the Earth on empirical grounds by around 330 BC
unit	False-True
choice	1.0

Slika 3.8: Ocena besedila na spletni strani.

ključ	vrednost
title	Earth is flat
unit	False-True
comments	['Dokaze za sferično obliko Zemlje imamo že od leta 330 pred našim štetjem.']

Slika 3.9: Ocena komentarja na trditev.

ključ	vrednost
title	Earth is flat
unit	False-True
comments	['Dokaze za sferično obliko Zemlje imamo že od leta 330 pred našim štetjem.', 'To je laž!']

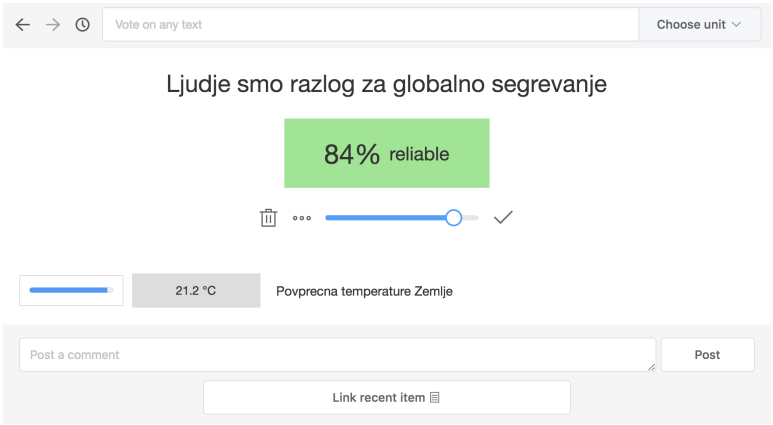
Slika 3.10: Ocena komentarja na komentar.

3.3 Povezovanje

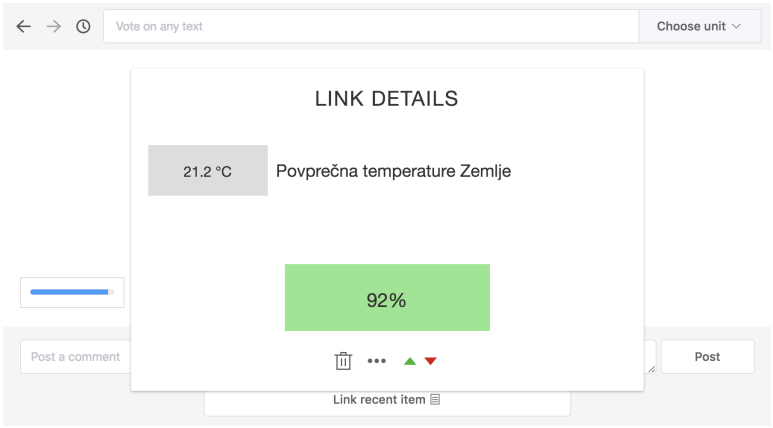
Vsako entiteto lahko povežemo s poljubno drugo in določimo relevantnost povezave med njima. Povezava med entitetama pomeni, da rezultat ene vpliva na rezultat druge. Na primer povprečna temperatura na Zemlji vpliva na resničnost globalnega segrevanja. Uporabnikom omogoča, da lažje najdejo razloge za določeno povprečno oceno neke entitete. S sistemom ocenjevanja so določene tudi same povezave. To pomeni, da lahko uporabniki za poljubno povezavo ene entitete k drugi povedo, kako relevantna se jim zdi. Za razliko od avtorja članka, ki ga piše ena sama oseba, nam povezovanje omogoča dodajanje povezave neki entiteti z glasovanjem s strani različno mislečih ljudi. V praksi se jih pogosto lahko uporablja za dodajanje posameznih študij k neki širši trditvi. Mišljeno je, da relevantnost povezave ocenimo neodvisno od rezultatov povezanih entitet. Iz tega razloga lahko povezava obstaja še preden uporabniki ocenijo povezane entitete. Za vsako entiteto imamo torej urejen seznam drugih povezanih entitet. To nam pride prav tudi za lažje raziskovanje entitet. Za večino kompleksnih trditev (npr. enoplačniški sistem zdravstva je boljši od ameriškega) obstajajo razlogi, ki trditev podpirajo, in razlogi, ki jo zavračajo. Zato dostikrat pride do nestrinjanja med ljudmi, saj se vsak osredotoči na argumente, ki podpirajo le eno stran. Lažje tudi ocenimo verjetnost, da je nekaj res, če imamo na voljo odgovore na bolj specifična vprašanja. Povezovanje entitet je prikazano na slikah 3.11, 3.12 in 3.13.

ključ	**vrednost**
key	['title', 'unit', 'reference_title', 'reference_unit']
title	Človeška aktivnost je razlog za globalno segrevanje.
unit	Unreliable-Reliable
reference_title	Povprečna temperatura na Zemlji
unit	Temperature(°C)
relevance	1.0

Slika 3.11: Podpisovanje ocene povezave.

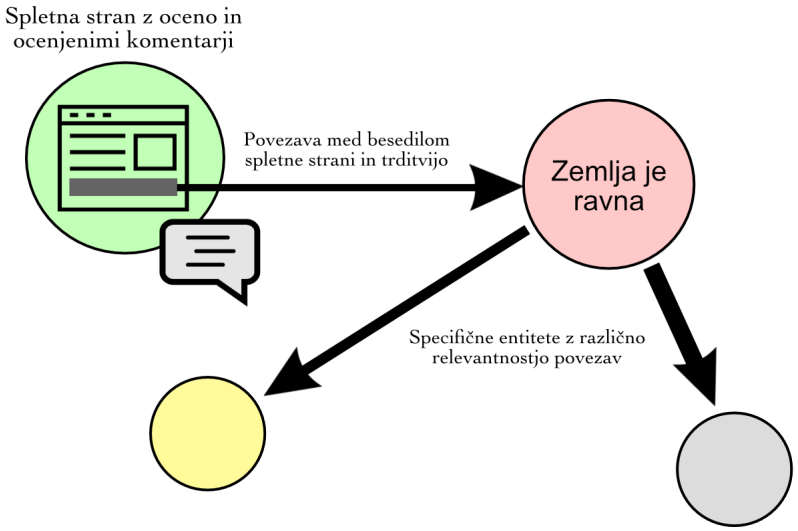


Slika 3.12: Prikaz povezane entitete.



Slika 3.13: Okno, kjer lahko ocenimo relevantnost povezave.

Povezovanje nam omogoča tudi bogatejšo anotacijo spleta. Entitete lahko dodajamo na poljuben URL naslov ali na besedilo znotraj spletne strani na tem naslovu. S tem nam omogoča, da vidimo relevantne, demokratično pridobljene informacije na pravem mestu ob pravem času. Povezujemo lahko tudi trditve z določeno temo. Teme so navadne entitete, ki jih ni smiselno ocenjevati, ampak k njim le povezovati druge entitete. Primer dokaj splošne teme so 'Družbeni problemi'. S tem uporabnikom omogočimo, da ocenijo, kako pomemben se jim zdi določen problem v primerjavi z ostalimi problemi. Tako lahko ocenimo, ali nam je npr. bolj pomembno zmanjšati globalno segrevanje, ali nam je bolj pomembna ekonomska rast. Povezave uporabnikom omogočajo razširitev oz. poglobitev svojega znanja o določeni temi. Uporabnikom ponujajo način najti novo vsebino znotraj razširitve, poleg pregleda vseh entitet oz. entitet na določeni domeni in iskanja po naslovu. Iz podatkov o ocenah in grafa zaupanja, ki nam predstavlja delegacije, z uporabo povezav izračunamo nov usmerjen graf, kjer so vozlišča entitete z ocenami, povezave pa so prav tako utežene. Primer grafa je viden na sliki 3.14.



Slika 3.14: Ocene in delegacije pretvorimo v graf viden zgoraj.

V znanosti poznamo meta študije. To so študije, ki analizirajo ugotovitve več različnih bolj poglobljenih študij glede istega vprašanja. Meta študije so idealne za vnos v našo platformo, za vnos pa so potrebni ljudje, ki podatke neke meta študije pretvorijo v obliko, primerno za prikaz v naši razširitvi. Z uporabo povezovanja bi lahko v razširitvi prikazovali tudi podatke, ki jih najdemo na različnih domenah na internetu (npr. AlternativeTo [2], ToSDR [23], IMDb [9], Reddit [21]). S tem lahko podatke več različnih platform vidimo v enotni obliki znotraj naše razširitve. To je zanimiva demonstracija fleksibilnosti razširitve, vendar s tem ne izkoriščamo prednosti seznamov zaupanja in tekoče demokracije, predstavljenih v tem delu. Vse javno dostopne podatke v obliki ocen, komentarjev in povezav lahko torej preslikamo v obliko, ki jo je naša razširitev zmožna prikazati. To bi storili tako, da bi napisali program, ki ga poganjamo periodično, recimo vsako uro. Ta program najprej prebere podatke na določeni domeni in jih pretvori v ustrezno obliko ter podpiše z nastavljenim uporabnikom. Če nato tega uporabnika dodamo na naš seznam zaupanja, bomo znotraj razširitve videli podatke, ki jih ta program vnaša. Podatki, ki bi jih lahko pridobili z pomočjo spletne platforme AlternativeTo so vidni na slikah 3.15 in 3.16.

ključ	vrednost
key	['title', 'unit']
title	Adobe Photoshop
unit	Rating
choice	0.6

ključ	vrednost
key	['title', 'unit']
title	GIMP
unit	Rating
choice	0.42

Slika 3.15: Posamezne ocene dveh podobnih neodvisnih programov.

ključ	vrednost
key	['title', 'unit', 'reference_title', 'reference_unit']
title	Adobe Photoshop
unit	Rating
reference_title	GIMP
unit	Rating
relevance	1.0

ključ	vrednost
key	['title', 'unit', 'reference_title', 'reference_unit']
title	GIMP
unit	Rating
reference_title	Adobe Photoshop
unit	Rating
relevance	1.0

Slika 3.16: Ocena povezave med programoma. Povezave na naši platformi so usmerjene, zato tu podatke podpišemo dvakrat za obe smeri.

3.4 Tehnična izvedba

Ta del diplomskega dela ima prav tako svoj strežnik, ki enako kot prvi uporablja jezik Elixir z ogrodjem Phoenix, opisanim v odseku 2.3. Strežnik v temu delu nima svoje podatkovne baze, temveč le zahteva podatke od strežnika, opisanega v prejšnjem delu, in jih posreduje razširitvi.
Razširitev za brskalnik uporablja Vue.js [25]. To je preprosta knjižnica (nekateri ji pravijo ogrodje), ki nam omogoča razvoj aplikacij s kompleksnimi uporabniškimi vmesniki. Glavne funkcionalnosti, ki nam jih ponuja, so reaktivnost podatkov in enkapsulirane komponente. Reaktivnost nam zagotovi, da bodo na zaslonu prikazani podatki vedno sinhronizirani z dejanskimi podatki oz. stanjem aplikacije. Enkapsulirane komponente nam omogočajo intuitiven način ločevanja uporabniškega vmesnika na več manjših delov.
Za pogoste komponente uporabljamo še dodatno knjižnico Elements-UI, ki temelji na Vue.js. Za pisanje pravil CSS uporabljamo Less, ki nam omogoča gnezdenje. Uporabniški vmesnik razširitve se nahaja znotraj Shadow DOM-a, kar nam zagotovi, da pravila CSS trenutne spletne strani ne povozijo naših pravil za prikaz dialoga, ki ga razširitev ponuja. Za prevajanje v čisti JavaScript, ki ga izvaja brskalnik, uporabljamo Webpack. Z uporabo Yarn-a zagotovimo, da so vsi naši paketi NPM zaklenjeni v točno določeno konfiguracijo.
Za hranjenje globalnega stanja razširitve uporabljamo Vuex, ki različnim komponentam omogoča dostop do istih podatkov. Znotraj projekta se Vue.js komponente ločijo po tem, kje hranijo svoje stanje. Bolj primitivne komponente, ki jih lahko uporabljamo na mnogo različnih mestih, svoje stanje hranijo lokalno, podatke pa dobijo prek atributov. Bolj kompleksne sestavljene komponente do svojega stanja dostopajo preko shrambe, kamor ga tudi zapisujejo. V nadaljevanju sta predstavljena dva Vuex modula, ki jih razširitev uporablja.
Strežnik deluje kot posrednik med strežnikom iz prejšnjega dela, ki hrani podatke, in razširitvijo. Podatke strežnik razširitvi vrne v naslednji obliki:

GET /api/nodes/:title 200 ali
GET /api/search/:query 200


```
{
  "data": [{
    "definition": {
      "unit": "Unreliable-Reliable",
      "title": "Something",
      "comments": [],
      "anchor": null
    },
    "data": {
      "results": {
        "voting_power": 1.0,
        "median": 1.0,
        "mean": 1.0,
        "contributions":[{
          "voting_power": 1.0,
          "username": "rknrnfutyematmgr",
          "choice": 1,
          "at_date": "2018-09-07T19:04:58.518Z"
        }]
      },
      "references": [],
      "inverse_references": [],
      "comments": []
    }
  }]
}
```


Strežnik s katerim komunicira razširitev najprej naredi zahtevek na strežnik, opisan v prvem delu diplomskega dela, in najde vse podatke uporabnikov z ustrezno strukturo na trenutnem seznamu zaupanja. Nato iz podatkov o zaupanju izračuna volilno moč vsakega uporabnika. Volilna moč se prenaša

samo, če uporabnik entitete ne oceni sam, zato jo je potrebno izračunati za vsako entiteto posebej. Zato se za izračun volilne moči uporablja memoizacija, ki teče v ločenem procesu. Sledi izračun povprečne ocene za vsako entiteto na podlagi ocen in podatka o volilni moči posameznih uporabnikov, nato pa se iz podatkov o ocenah relevantosti povezav izračunajo povezave med entitetami. Strežnik podatke nato pretvori v odjemalcu prijazno strukturo, vidno zgoraj, kjer so pripadajoči komentarji in povezane entitete vgnezdeni v podatkih entitet. Generične Vue.js komponente, ki se uporabljajo za delovanje razširitve so predstavljene na sliki 3.17. Na slikah 3.18 in 3.19 pa so opisani Vuex moduli. Koda za proženje dogodka za podpisovanje podatkov je prikazana na sliki 3.20.

Slika 3.17: Komponenta Node, ki je sestavljena iz označenih preprostih komponent. Prikazuje poljubno entiteto ter nam omogoča ocenjevanje, komentiranje in povezovanje drugih entitet.

ključ	**vrednost**
signDialogVisible	Določa, ali je dialog za podpisovanje podatkov in prijavo z uporabnikom odprt.
randomWords	Naključno generirane besede, ki nam služijo kot geslo novega uporabnika
seeds	Seznam prijavljenih uporabnikov. Podatki se izračunajo iz vnešenih besed in se uporabljajo za deterministično generiranje para ključev.
messages	Sporočila za podpisovanje.
whitelistUrl	Naslov URL trenutnega seznama zaupanja. Privzeto je nastavljen na vrednost https://sign.liqu.io/whitelist.html, v razvojnem okolju pa http://localhost:5000/whitelist.html.
whitelistUsernames	Seznam uporabniških imen na trenutnem seznamu zaupanja.
userMessages	Seznam sporočil uporabnika, katerega profil imamo odprt.

signItems: Podpiše trenutne podatke v polju 'messages'.

createUser: Generira seznam 13 besed.

downloadIdentity: Shrani seznam 13 besed na lokalno napravo.

login: Prijavi uporabnika s podanimi 13 besedami.

Slika 3.18: Vuex modul za podpisovanje podatkov in interakcijo s strežnikom iz prejšnjega poglavja.

ključ	**vrednost**
dialogVisible	Določa, ali je dialog odprt
nodes	Seznam entitet, ki jih dobimo iz strežnika.
definition	Definicija trenutno prikazane entitete.
currentPage	URL trenutne spletne strani.
currentSelection	Trenutno izbrano besedilo.
currentVideoTime	Trenuten čas na video posnetku, če se nahaja na strani.
activeDefinition	Definicija entitete, na katero smo se premaknili z miško.

initialize: Označi ocenjena besedila znotraj spletne strani in poskrbi, da bodo označena tudi takrat, ko se podatki o entitetah posodobijo. Prav tako začne opazovati, kdaj uporabnik neko besedilo označi in ga zapiše v stanje v polje 'currentSelection'. Nastavlja tudi polje 'currentPage' ter 'currentVideoTime'. Prav tako nastavlja polje 'activeDefinition', ko se z miško premaknemo čez ocenjeno besedilo.

loadNode: Naloži podano entiteto v polje 'nodes'.

setDefinition: V dialogu nas postavi na entiteto s podano definicijo.

setVote, setCommentVote, setReferenceVote: Odprejo dialog za podpisovanje podatkov.

Slika 3.19: Vuex modul za ocenjevanje in povezovanje entitet in ostale koncepte opisane v tem poglavju.

```
const messages = [
   {
      key: ["username"],
      username: "qubybbhnrmicnbeu",
      trust: 0.75
   }
]

window.dispatchEvent(new CustomEvent(
   "liquio-sign",
   { detail: messages }
))
```

Slika 3.20: Proženje dogodka, ki odpre okno za podpisovanje podatkov.

Poglavje 4

Zaključek

S predstavljeno platformo lahko enostavno ocenjujemo poljubno vsebino na internetu. Platforma vsebuje vse koristne lastnosti delegativne demokracije, kot so tranzitivne delegacije in transparentnost. Ocene in ostali podatki so digitalno podpisani, kar nam zagotavlja, da jih je res ustvaril določen uporabnik. Seznami zaupanja pa nam omogočajo, da sami določimo uporabnike, katerih podatke želimo videti. Naša platforma torej vsebuje koncepte, ki so na internetu zelo razširjeni že danes. Vsebuje generičen in fleksibilen sistem ocenjevanja, ki ga je mogoče uporabljati na kateri koli spletni strani. Pri razvoju uporabniškega vmesnika poskušamo uporabiti koncepte, ki so jih ljudje že navajeni iz drugih platform. Sistem je tudi zasnovan tako, da je na najnižjem nivoju decentraliziran ter s tem odporen na cenzuro, za kar je zaslužen IPFS.

Orodja za anotacijo spleta niso nič novega. Glavna posebnost naše platforme je koncept seznamov zaupanja. Večina obstoječih platform ima centraliziran sistem moderacije, na katerega končni uporabniki nimajo vpliva. Zavedamo se, da je uporabnost naše platforme povsem odvisna od števila uporabnikov s kakovostnimi podatki, pridobivanje uporabnikov pa ni odvisno samo od tehnologije in je po svoje precej bolj zahtevno.

Verzija trenutno objavljene razširitve še ne ponuja nekaterih funkcionalnosti, opisanih v diplomskem delu oz. nekatere funkcionalnosti še ne delujejo v ce-

loti. V tem delu opisana verzija bo verjetno na voljo do konca leta. Prostora za izboljšave in raziskovanje pa je še ogromno. V nadaljevanju podajamo nekaj temeljnih stvari, ki bi jih lahko naredili v prihodnosti:

- Za izračun relevantnosti povezave med entitetami bi lahko uporabljali metodo glajenja s prištevanjem [1]. S tem bi dali manjšo relevantnost povezavam, ki imajo malo ocen. Enako bi lahko počeli pri računanju povprečne ocene komentarja ali spletne strani. To metodo uporablja tudi IMDb za določanje najboljših 250 filmov. S prištevanjem lahko gladimo rezultate za entitete s spektralno enoto takrat, samo če vemo, da je ocena 100 % boljša od ocene 0 %. Za entiteto z naslovom 'Zemlja je ravna' nočemo dodajati ocen z verjetnostjo 50 %, saj bi že s tem zavajali.

- Nad ocenami bi lahko izvajali faktorsko analizo. Analizirali bi lahko, kako različne ocene sovpadajo. S tem bi lahko potencialno združevali več praktično enakih entitet v eno samo. Trenutno implementacije tega še ni, saj nimamo dovolj ocen za analizo.

- Možnost določanja, kako prepričani smo v svojo izbrano oceno. To bi storili z ocenjevanjem z normalno distribucijo, kjer večji standardni odklon pomeni manjšo prepričanost.

- Ugotavljanje, ali neka entiteta podpira ali zavrača trditev, h kateri je povezana. To bi počeli tako, da bi izračunali povprečno oceno za trditev, a bi pri tem gledali samo ocene uporabnikov, ki so ocenili, da je povezava relevantna.

- Izvoz in uvoz podatkov nekega uporabnika v tekstovno datoteko.

- Razširitev trenutno podatke dobi preko našega strežnika, lahko pa bi jih prebrala neposredno iz IPFS-ja s čimer bi povečali decentralizacijo platforme. Na ta način bi bila razširitev dosti bolj zahtevna za CPE in delovni pomnilnik, prav tako pa bi bila počasnejša zaradi omrežja. Za

implementacijo vmesnega strežnika smo se odločili, ker s tem povečamo dostopnost podatkov in omogočimo uporabo istega API-ja v različnih programih, napisanih v različnih programskih jezikih. Odločili smo se torej narediti kompromis in na začetku zmanjšati decentralizacijo na račun boljše dostopnosti. Strežnik je odprtokoden, zato ga lahko tudi zdaj vsak požene na svoji napravi. V začetni fazi nam je predvsem pomembno, da podatke ustrezno shranjujemo, saj so ti najbolj pomembni za uporabno platformo.

- Ocenjevanje entitet s historičnimi podatki (npr. povprečna temperatura Zemlje). Ocenam bi torej lahko izbirali poljuben datum. Tako lahko vsak dan v preteklosti ali prihodnosti ocenimo s poljubno numerično vrednostjo, namesto ene povprečne ocene pa vidimo graf povprečij po času.

- Delegiranje glasov nekemu uporabniku samo za entitete z določeno temo. Entitetam lahko dodajamo teme že zdaj z uporabo povezovanja. To storimo tako, da k temam (npr. znanost, politika, ekonomija) povežemo entitete, ki sodijo vanje.

- Posebna enota, ki nam omogoča ocenjevanje verjetnosti z standardnim odklonom, ki je nato pretvorjen in prikazan kot odstotek.

- Omogočanje deljenja podatkov s točno določenimi uporabniki, npr. tistimi, ki so na našem seznamu zaupanja. To bi počeli tako, da bi podatke šifrirali z javnim ključem vsakega uporabnika, ki mu jih želimo deliti. Podatke bi nato lahko dešifrirali samo uporabniki z pripadajočim zasebnim ključem. Te funkcionalnosti nismo implementirali, ker IPFS vozlišča niso finančno motivirana hraniti šifriranih podatkov, katerih vsebina jim je neznana.

Naša platforma temelji na konceptu seznamov zaupanja in ponuja demokratičen sistem za ocenjevanje, komentiranje in povezovanje vsebine na spletu.

Literatura

[1] Additive smoothing. https://en.wikipedia.org/wiki/Additive_smoothing. [Dostopano: 10. 5. 2018].

[2] Alternativeto. https://alternativeto.net. [Dostopano: 17. 9. 2018].

[3] Blockchain. https://en.wikipedia.org/wiki/Blockchain. [Dostopano: 10. 5. 2018].

[4] Digitalni podpis. https://en.wikipedia.org/wiki/Digital_signature. [Dostopano: 10. 5. 2018].

[5] Eddsa. https://en.wikipedia.org/wiki/EdDSA. [Dostopano: 26. 9. 2018].

[6] Elixir. https://elixir-lang.org. [Dostopano: 17. 9. 2018].

[7] Erlang. https://www.erlang.org. [Dostopano: 17. 9. 2018].

[8] Half life. https://en.wikipedia.org/wiki/Half-life. [Dostopano: 10. 5. 2018].

[9] Imdb. http://imdb.com. [Dostopano: 17. 9. 2018].

[10] Ipfs. https://ipfs.io. [Dostopano: 17. 9. 2018].

[11] Let's encrypt. https://letsencrypt.org. [Dostopano: 17. 9. 2018].

[12] Liquio. https://liqu.io. [Dostopano: 18. 9. 2018].

[13] Liquio code. https://github.com/zan-kusterle/Liquio. [Dostopano: 18. 9. 2018].

[14] Metrika zaupanja. https://en.wikipedia.org/wiki/Trust_metric. [Dostopano: 10. 5. 2018].

[15] Mnemonic code for generating deterministic keys. https://github.com/bitcoin/bips/blob/master/bip-0039.mediawiki. [Dostopano: 10. 5. 2018].

[16] Napad sybil. https://en.wikipedia.org/wiki/Sybil_attack. [Dostopano: 10. 5. 2018].

[17] Nginx. https://www.nginx.com. [Dostopano: 17. 9. 2018].

[18] Npm. https://www.npmjs.com. [Dostopano: 10. 5. 2018].

[19] Pagerank. https://en.wikipedia.org/wiki/PageRank. [Dostopano: 10. 5. 2018].

[20] Phoenix framework. https://phoenixframework.org. [Dostopano: 17. 9. 2018].

[21] Reddit. https://www.reddit.com. [Dostopano: 17. 9. 2018].

[22] Rest. https://en.wikipedia.org/wiki/Representational_state_transfer. [Dostopano: 17. 9. 2018].

[23] Terms of service; didn't read. https://tosdr.org. [Dostopano: 17. 9. 2018].

[24] Ubuntu. https://www.ubuntu.com. [Dostopano: 17. 9. 2018].

[25] Vue.js. https://vuejs.org. [Dostopano: 17. 9. 2018].

[26] Wikipedija. https://www.wikipedia.org. [Dostopano: 17. 9. 2018].

[27] J. Behrens, A. Kistner, A. Nitsche, and B. Swierczek. *The Principles of LiquidFeedback*. Interaktive Demokratie, 2014.

[28] L. Festinger. *A Theory of Cognitive Dissonance*. Mass communication series. Stanford University Press, 1962.

[29] Raymond S. Nickerson. *Confirmation Bias: A Ubiquitous Phenomenon in Many Guises*. 1998.