Univerza v Ljubljani
Fakulteta za računalništvo in informatiko

Andraž Povše

Identifikacija najugodnejših poti pri nakupu kriptovalut

diplomsko delo

univerzitetni študijski program
prve stopnje
računalništvo in informatika

Mentor: doc. dr. Dejan Lavbič

Ljubljana, 2018

Besedilo je oblikovano z urejevalnikom besedil LaTeX.

Fakulteta za računalništvo in informatiko izdaja naslednjo nalogo:

Tematika naloge:

Tehnologija veriženja blokov je v zadnjem času zelo prodoren in pogosto uporabljen način za reševanje problemov na različnih poslovnih področjih. Na omenjeni tehnologiji temeljijo tudi priljubljene kriptovalute, ki so postale zanimiva alternativa vlagateljem finančnih sredstev, ki so nekoliko bolj nagnjeni k tveganju. S pojavom vedno večjega števila kriptovalut in tudi menjalnic, ki omogočajo njihovo trgovanje, je postalo vlaganje v novejše in ne tako razširjene kriptovalute, precej zapleteno. Glede na to, da večina menjalnic s kriptovalutami omogoča dostop preko točk API, v okviru diplomskega dela predlagajte rešitev, ki uporabniku omogoča najugodnejši nakup kriptovalute. To pogosto predstavlja trgovanje na različnih menjalnicah z različnimi trgovalnimi pari, kjer manjše število trgovalnih akcij ne pomeni nujno tudi najugodnejše poti. Predlagano rešitev implementirajte in kritično ovrednotite na izbranih scenarijih ter primerjajte z obstoječimi pristopi.

Kazalo

Seznam uporabljenih kratic

kratica	angleško	slovensko
API	application programming interface	Aplikacijski programski vmesnik
JSON	Javascript Object Notation	Javascript objektna notacija
PoW	Proof of Work	Dokaz o delu
ICO	Initial Coin Offering	Začetna ponudba kriptovalute. Podobno kot IPO pri delnicah.
ERC20	Ethereum Request for Comment 20	Etherium zahtevek za komentar 20
SPA	Single Page Application	Enostranska spletna aplikacija
SEPA	Single Euro Payments Area	Enotno območje plačil v evrih

Povzetek

Naslov: Identifikacija najugodnejših poti pri nakupu kriptovalut

Avtor: Andraž Povše

Rast trenda trgovanja kriptovalut na menjalnicah, enostavnost uporabe, povečana medijska pozornost in ideja hitrega zaslužka so v letu 2017 pritegnile veliko število novih uporabnikov. Trgovanje poteka tako, da se uporabnik registrira, na menjalnico nakaže denar in kupi izbrano kriptovaluto. Lahko se zgodi, da le te ni na voljo, zato je potrebno ustvariti račun še drugje, poslati sredstva in tam izvesti novo menjavo. Vseh stroškov, ki nastanejo tekom pridobitve željene kriptovalute se uporabniki velikokrat ne zavedajo. Cilj diplomske naloge je izdelava optimalne poti pri nakupu izbrane kriptovalute. Na podlagi podatkov, prejetih iz dostopnih točk API, algoritem izdela omrežje menjalnih parov in v njem poišče najugodnejšo pot za nakup. Uporabnik ima pregled nad izbrano potjo in lahko po njenih korakih izvrši transakcije in menjave. Algoritem je izdelan v programskem jeziku Python in postavljen kot spletna aplikacija v okoljih Flask in Vue. Za hrambo podatkov je uporabljena nerelacijska podatkovna baza MongoDB.

Ključne besede: kriptovalute, menjalnice, borze, najkrajša pot, nakup kriptovalut.

Abstract

Title: Identifying optimal paths for purchasing cryptocurrencies.

Author: Andraž Povše

Increase of trend in trading cryptocurrencies, ease of use, surge in media attention and the idea of making quick money were contributing factors for massive growth in users in the year 2017. To start trading, the user must first register on the desired exchange to which he transfers his funds and exchanges them for the chosen cryptocurrency. In the event that the target cryptocurrency is not available on this exchange, he must create another account on a different exchange, send the currency there and execute another trade. Many times users do not realise all the fees that take place during this process. The goal of this Bachelor's thesis is to create an algorithm that provides the end user with optimal path for purchasing a specific cryptocurrency. Based on data, provided by API points, the algorithm generates a network of trading pairs in which it finds the optimal path for purchase. The user can then make actions based on the provided path. The algorithm is made in programming language Python, using MongoDB database and set up as a web application using Flask and Vue frameworks.

Keywords: cryptocurrency, cryptocurrency exchange, shortest path, purchasing cryptocurrencies.

Poglavje 1

Uvod

Prvo srečanje s trgom kriptovalut lahko za uporabnika predstavlja velik izziv. Če kriptovaluta ni na voljo na izbrani menjalnici, se moramo registrirati na več različnih, izvršiti več menjav in transakcij med menjalnicami. Do tega prihaja zato, ker imajo dovoljenja za poslovanje z denarjem le nekatere menjalnice, ravno te pa pogosto nimajo široke izbire različnih kriptovalut.

Lahko se zgodi, da čeprav prvotna menjalnica ponuja izbrano kriptovaluto, ta pot ni optimalna zaradi višjih provizij ali tečaja. V takem primeru je zato boljše, če se registriramo drugje in tako preko več menjalnic izvedemo nakup ciljne kriptovalute. Večinoma takšno poizvedovanje vzame veliko časa in zahteva vsaj osnovno poznavanje nekaterih menjalnic ter provizijskih tečajev.

Zaradi želje po minimizaciji stroškov pri poti do izbrane kriptovalute smo v diplomskem delu razvili rešitev, ki na podlagi začetne valute in višine investicije izračuna optimalno pot za nakup končne kriptovalute. Da bi ustvarili omrežje menjalnih parov, v katerem iščemo najugodnejšo pot, potrebujemo podatke o trenutnih tečajih, provizijah pri menjavah in pošiljanju. Najprej smo ustvarili algoritem, ki se je povezal z menjalničnimi točkami API in drugimi relevantnimi spletnimi stranmi, prejeli podatke in jih zapisali v podatkovno bazo. Algoritem za izdelavo omrežja in iskanje optimalne poti nato pridobi podatke iz podatkovne baze in na koncu izdela ter grafično prikaže optimalno pot. Kot zaključek bo celoten algoritem deloval kot spletna apli-

kacija in s tem omogočil enostavnejšo uporabo.

V sledečem poglavju bomo opisali področja, ki se navezujejo na vsebino izdelane rešitve. Poglavje vsebuje pregled področja kriptovalut, opis iskanja optimalnih poti in za konec opis obstoječih sorodnih rešitev. Po pregledu področja bomo v tretjem poglavju opisali tehnologije, ki smo jih uporabili, arhitekturo celotne rešitve in načrtovanje omrežja, v katerem bomo kasneje iskali najugodnejšo pot. Četrto poglavje vsebuje opis dejanske implementacije rešitve. V začetku poglavja je opisano pridobivanje in filtriranje podatkov, kateremu sledi opis dejanske izdelave omrežja z iskanjem optimalne poti ter na koncu prenos rešitve na spletno okolje. V petem poglavju smo izdelano rešitev primerjali s sorodnimi in evaluirali optimalne poti pri naši rešitvi. Zadnje poglavje diplomskega dela predstavlja sklepne ugotovitve z možnimi izboljšavami.

1.1 Motivacija

Začetnik trgovalec lahko meni, da s svojim trgovanjem ustvarja dobiček, v resnici pa ga ustvarja samo menjalnica, kjer trguje. Provizije, ki nastanejo ob pošiljanju valut med menjalnicami in trgovanjem znotraj nje, so lahko velike in imajo tako vpliv na naše trgovanje. Provizije ob menjavi so zaračunane v obliki deleža (po navadi okoli 0,1 % in 0,25 %), medtem ko so stroški nakazil na in z menjalnic večinoma fiksne vrednosti. Cilj uporabnika naj bo, da pošilja kriptovalute med menjalnicami v čim večjih vrednostih, saj se s tem zmanjša procentualna vrednost stroška pošiljanja. Naslednji problem, na katerega naletimo, je število menjalnic, ki jih je v tem trenutku več kot 200 [1]. Kriptovaluta, ki jo želimo kupiti, je lahko na voljo v več menjalnicah, mi pa se moramo odločiti, kje jo bomo kupili. Pri tem problemu bi uporabnik moral porabiti čas, da bi menjalnice pregledal, se odločil, katera bi njegovem primeru uporabe omogočala najbolj optimalen nakup, se registrirati in na koncu izvršiti menjavo.

Ker poleg različnih menjalnic obstajajo tudi mnogi menjalni pari z istimi

kriptovalutami (kriptovaluto EOS lahko na primer kupimo z valuto EUR, USD, kriptovalutami BTC, ETH, BNB, USDT ...), je potrebno izbrati tudi najugodnejši par. Kot bomo videli v poglavju 4.3.2, najugodnejša pot ni vedno tista, ki vsebuje najmanj menjav.

V diplomskemu delu smo želeli ta postopek avtomatizirati, a za razliko od sorodnih rešitev, opisanih v poglavju 2.5, ohraniti transparentnost in uporabniku prikazati dejansko pot, ki jo je potrebno izvršiti. Tako smo ustvarili program, ki namesto uporabnika pregleda menjalnice, iz katerih pridobi podatke o stroških transakcij, provizij ob menjavi, vrstah menjalnih parov in trenutnih tečajev. S temi podatki izdela omrežje menjalnih parov, po katerih išče najugodnejšo pot do ciljne kriptovalute. Izračunana pot se uporabniku prikaže kot omrežje, kateremu lahko s svojim trgovanjem sledi in tako pride do najkrajše poti za nakup kriptovalute.

1.2 Cilji

Cilj rešitve v diplomskem delu je povezovanje podatkov menjalnic v skupno aplikacijo in s tem olajšati odločanje o poteku nakupa določene kriptovalute, saj imamo vse potrebne podatke na enem mestu. Aplikacija bi služila uporabnikom, ki želijo kupiti določeno kriptovaluto, a ne vedo, kje se naj registrirajo in kakšne menjave naj izvedejo, da pridejo do optimalne poti. Rešitev naloge je koristna tudi za iskanje najugodnejših poti pri kakšnih drugih problemih ali kot nadgradnja obstoječih storitev. Podoben problem bi lahko nastal pri nakupu delnice, ki kotira na različnih borzah. Tudi tam bi potrebovali najugodnejšo pot za nakup, saj bi bila delnica v menjalnih parih povezana z več različnimi valutami, ki niso nujno enake naši izhodiščni valuti.

Podoben problem bi lahko obstajal tudi kot iskanje najcenejše poti pri letalskem letu iz začetne točke A do končne točke B, kjer lahko letimo z direktnim ali pa s kombinacijo letov.

Poglavje 2

Pregled področja

V tem poglavju bomo pregledali področja, na katera se naša rešitev nanaša. V začetku bomo pregledali kriptovalute in tehnologijo veriženja blokov, ki jo uporabljajo. Nato se bomo posvetili menjalnicam kriptovalut, ki služijo kot glavni vir podatkov pri naši rešitvi. Izbrane menjalnice bomo na kratko opisali, oziroma utemeljili njihov izbor. Za konec poglavja sledi opis iskanja optimalnih poti in pregled dveh sorodnih rešitev.

2.1 Kriptovalute

Večina kriptovalut ima omejeno končno število kovancev, ki so ali bodo nastali. Ker so zaloge navzgor omejene, se rast zanimanja po določeni kriptovaluti odrazi s povečanjem njene cene. Podobno logiko lahko najdemo tudi v ozadju cen plemenitih kovin, ki imajo poleg praktične uporabe visoko ceno tudi zaradi njihove redkosti. Trenutno po svetu uporabljamo fiat denar (npr. EUR, USD itd.), ki ni količinsko omejen, njegovo število pa je prilagodljivo. Tako lahko s povečanjem števila fiat denarja v obtoku, povišamo stopnjo inflacije in s tem znižamo dejansko vrednost posamezne enote denarja [2].

Pojav kriptovalut se je začel leta 2009 s kriptovaluto Bitcoin, ki jo je Satoshi Nakamoto opisal že v letu 2008 [3]. Od takrat naprej je Bitcoin rasel na popularnosti in ceni, nastajalo pa je tudi vedno več novih kriptovalut.

V letu 2014 je obstajalo že okoli 67 različnih kriptovalut, med katerimi so poleg Bitcoina (88,2 %) sorazmerno velik tržni delež glede na ostale imeli še Litecoin (5,2 %), Ripple (1,8%), Peercoin (1,3 %), Omni (0,9 %), NXT (0,5 %) in Namecoin (0,5 %) [4]. Od sedmih zgoraj naštetih kriptovalut so trenutno med prvimi 100 po tržni kapitalizaciji le še štiri. Etherium [5], ki je trenutno druga najvišje uvrščena kriptovaluta glede na tržno ktalizacijo, je nastal julija 2015 preko pristopa zbiranja prvotnih sredstev ICO.

Proti koncu leta 2013 je cena Bitcoina zrasla iz približno 120 $ v oktobru na 1.149 $ v začetku decembra. Omenjena rast je bila zelo verjetno posledica manipulacije cene s strani dveh algoritmov za trgovanje poimenovanih Marcus in Willy [6]. Algoritma naj bi se obnašala sumljivo, Marcus je deloval kot zgolj eden uporabniški račun in je med februarjem ter septembrom leta 2013 skupno nabral 335.898 Bitcoinov. Ko je Willy končal z nakupi, se je kot skupina 49 uporabniških računov pojavil algoritem Marcus. Vsak izmed teh uporabniških računov je drug za drugim kupil Bitcoine v vrednosti natančno 2,5 milijonov dolarjev. Zaradi teh nakupov je torej sledila takšna rast cene v decembru. Nato se je zgodil padec cene, ki je svoj minimum (204 $) dosegel komaj januarja leta 2015. S koncem padca se je sprva začela počasna rast, ki se je v letu 2017 spremenila v izjemno hitro in nestabilno rast cene z najvišjo vrednostjo v decembru, ki je znašala skoraj 20.000 $.

Vzporedno z rastjo cene in popularnosti prvotne kriptovalute je od leta 2014 naprej nastajalo ogromno novih kriptovalut. Sprva počasneje, najbolj očitna rast novih kriptovalut na trgu pa je bila vidna v letu 2017, korelirana z izjemno rastjo cene že obstoječih kriptovalut. Na trgu je bilo torej zaradi rasti cen obstoječih kriptovalut veliko sredstev, ki so jih uporabniki vlagali v novo nastale kriptovalute večinoma preko pristopa za zbiranje sredstev ICO [7]. Zanimivo dejstvo je, da večina nastalih kriptovalut ni bila zgrajena kot popolnoma nova tehnologija, temveč so bile izdelane na obstoječih kriptovalutah, ki to omogočajo. Primer kriptovalute, ki v svojem omrežju omogoča izdajo novih kriptovalutnih kovancev, je Etherium. Na njegovem omrežju tako najdemo več kot 800 različnih kriptovalut, tipa ERC20, ki so

dejansko nastale kot program znotraj okolja Etherium [8].

2.2 Tehnologija veriženja blokov - Blockchain

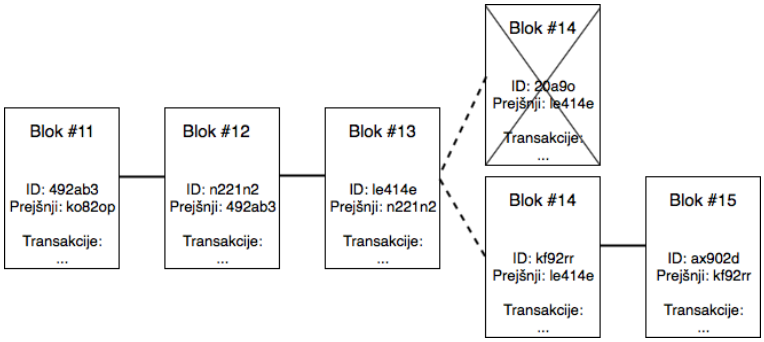
Ena izmed prednosti, ki jih podajajo kriptovalute, je uporaba tehnologije veriženja blokov. Preko uporabe te tehnologije se znebimo centralnega sistema, ki je odgovoren za potrjevanje transakcij in tako ustvarimo omrežje, v katerem se transakcije potrjujejo med uporabniki - vozlišči [9, 10].

Primer tehnologije veriženja blokov, kjer se uporablja pristop dokaza o delu (POW), najdemo pri kriptovaluti Bitcoin [11]. Vsako vozlišče v omrežju ima lastno kopijo javne knjige nakazil (angl. *ledger*) [12], ki je pri vseh konsistentna. Javna knjiga nakazil vsebuje podatke o vseh transakcijah, ki so se že zgodile. S pošiljanjem kriptovalute ustvarimo novo transakcijo, ki jo naše vozlišče posreduje sosednjim vozliščem in se tako širi po omrežju, kjer vsako vozlišče posodobi svojo knjigo nakazil.

Korak, ki sledi, je razlog, zakaj ta postopek imenujemo tehnologija veriženja blokov [13]. Bitcoin omrežje zgoraj omenjene transakcije zapakira v enakomerno velike bloke, ki vsebujejo fiksno število transakcij in povezavo do prejšnjega bloka. Vse transakcije znotraj enega bloka so, gledano s strani omrežja Bitcoin, nastale v istem času. Transakcije, ki smo jih ustvarili, vendar še niso znotraj bloka, se imenujejo nepotrjene transakcije. Da bi novo izdelani blok dodali na obstoječo verigo, je potrebno uganiti naključno število, ki je skupaj z vsebino prejšnjega bloka enako definiranemu rezultatu. Temu postopku pravimo rudarjenje. Pridobitev rešitve je računsko zahtevna operacija, a se zaradi skupnega dela pri ugibanju rezultata celotnega omrežja trenutno nov blok doda v verigo vsakih 10 minut. V primeru, da več vozlišč reši problem v istem času, se bo ustvarilo več vzporednih verig (glej sliko 2.1). Bitcoin omrežje izbere verigo, ki je najdaljša, saj verjetnost, da bi več vozlišč rešilo problem v istem času večkrat zaporedoma, hitro pada. Vozlišče, ki prvo izračuna rešitev problema in tako doda nov blok v verigo, dobi za nagrado tudi določeno število Bitcoinov ter prispevke vseh transakcij,

ki so v bloku [14]. Ker vozlišče pridobi tudi prispevke transakcij, so transakcije, ki plačajo večje prispevke, hitreje dodane v blok in potrjene. Ko bo doseženo končno število Bitcoinov (21.000.000), bo zmagovalno vozlišče prejelo le prispevke transakcij znotraj bloka.

Čas potrditve naše transakcije in višine prispevkov so tako odvisne od trenutne zasičenosti omrežja in kriptovalute. V primeru, da je omrežje zasičeno, bo z istim prispevkom transakcija trajala dlje časa in bo potrebno prispevek povečati, če želimo ohraniti enak čas kot takrat, ko je omrežje nenasičeno. To je bilo opazno predvsem v letu 2017, ko so bili nekateri projekti ICO razprodani v roku nekaj deset sekund. Takrat so uporabniki želeli svojo transakcijo poslati čim hitreje in so tako za transakcije kriptovalute Etherium plačevali tudi do 6.600 $ [15].



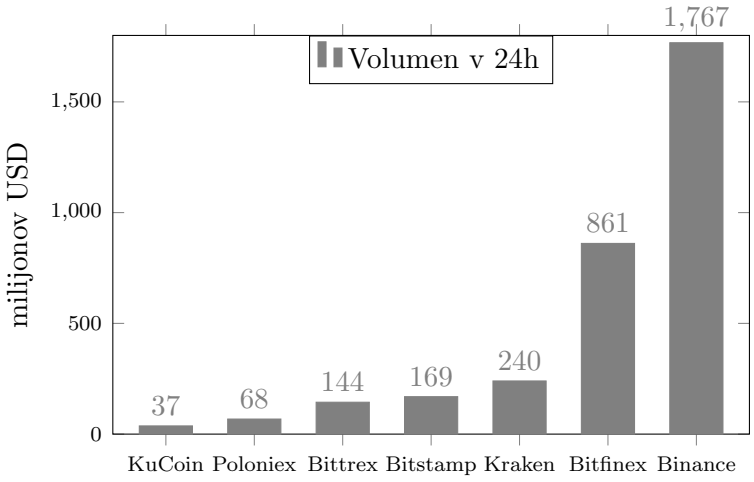
Slika 2.1: Primer dveh vzporednih verig. Ko je bil nov blok dodan v spodnjo verigo je bila ta izbrana in zgornja zavržena.

2.3 Menjalnice kriptovalut

Pri svoji rešitvi smo uporabljali podatke z menjalnic Bitstamp [16], Kraken [17], Bitfinex [18], KuCoin [19], Poloniex [20], Bittrex [21] in Binance [22]. Menjalnice smo izbrali na podlagi njihove likvidnosti in ponudbe menjalnih parov. Ker omogočajo menjavo fiat valut v kriptovalute samo nekatere menjalnice, smo vključili tudi te. To so trenutno Bitstamp, Kraken in Bi-

tfinex. Verjetno pa bo skozi čas menjave fiat valut za kriptovalute omogočalo vedno več menjalnic.

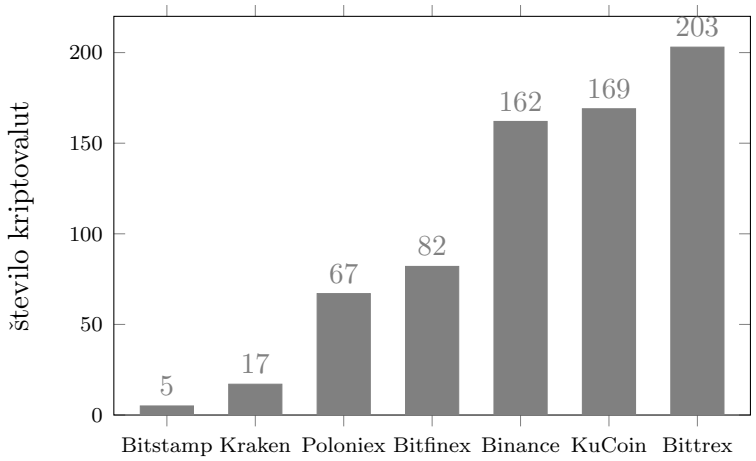
Dnevni promet menjalnic nam pove, kakšna je vrednost vseh menjav, ki so se zgodile v zadnjih 24 urah. Zaradi različnih zunanjih dejavnikov dnevni prometi menjalnic nihajo iz dneva v dan. V veliko primerih opazimo, da se s povečevanjem dnevnega prometa, povečujejo tudi cene kriptovalut. Na stolpičnem grafikonu na sliki 2.2 je prikazan promet izbranih menjalnic, za dan 18.7.2018.



Slika 2.2: Dnevni promet menjalnic, vir CoinMarketCap, 18.7.2018.

Poleg samega prometa oziroma likvidnosti menjalnice pa je za naš razvoj pomembna tudi ponudba kriptovalut. Zaradi tega smo nekatere menjalnice izbrali kljub njihovi majhni likvidnosti, saj je raznolika ponudba kriptovalut pomembna, če želimo omogočiti poizvedovanje po optimalni poti za čim večje število obstoječih kriptovalut. Na grafu 2.3 si lahko ogledamo število kriptovalut, s katerimi trgujejo izbrane menjalnice. Poleg števila samih kriptovalut, ki so na voljo na posamezni menjalnici, lahko primerjamo tudi število menjalnih parov. Tako ima na primer menjalnica Binance 162 različnih kriptovalut in 376 menjalnih parov (Coin Market Cap, 25. 7. 2018). Menjalni par nam pove, katero kriptovaluto lahko zamenjamo za katero. Tako imamo lahko na

primer le eden menjalni par oblike EUR/BTC, ki nam pove, da lahko kupimo ali prodamo Bitcoin za EUR. Na isti menjalnici pa imamo lahko tudi več menjalnih parov, preko katerih pridemo do iste kriptovalute. Primeri, ko pridemo do iste ciljne kriptovalute, so naslednji menjalni pari, ki omogočajo nakup kriptovalute LTC: BTC/LTC, ETH/LTC, EUR/LTC.



Slika 2.3: Število različnih kriptovalut, ki so na voljo v posamezni menjalnici, dne 25.7.2018.

Izbrane menjalnice bomo v spodnjih podpoglavjih opisali in pojasnili, zakaj smo jih vključili v razvoj aplikacije.

2.3.1 Bitstamp

Bitstamp [16] je ena izmed najstarejših menjalnic kriptovalut, z nastankom v Sloveniji. Pojavila se je že leta 2011 in omogočala trgovanje z Bitcoinom. V letu 2013 je svoj sedež premaknila v London. Zaradi vdora v sistem je leta 2015 zaustavila delovanje za en teden [23]. Ukradenih je bilo okrog 19.000 Bitcoinov, ki jih je Bitstamp povrnil v celoti [24].

Menjalnica je popularna za trgovalce v Evropi, saj podpira nakazila SEPA in s tem služi kot most med fiat denarjem (na primeru Bitstampa EUR in USD) in kriptovalutami. Sčasoma so dodali tudi druge bolj popularne kripto-

valute, kot so Litecoin, Ripple, Etherium in Bitcoin Cash. Vse kriptovalute, ki jih ponujajo, imajo menjalne pare z Bitcoinom, evrom in ameriškim dolarjem.

2.3.2 Binance

Binance [22] je trenutno najbolj likvidna menjalnica kriptovalut. Polega tega ima tudi zelo velik nabor menjalnih parov in ponudbo različnih kriptovalut.

Ustanovljena leta 2017 je prva izmed menjalnic, ki je uvedla svojo kriptovaluto – Binance Coin. Kriptovaluta Binance Coin je na trg prišla v juliju 2017 preko pristopa za zbiranje sredstev ICO, kjer so zbrali 15 milijonov dolarjev [25]. Binance Coin je ERC20 kovanec, narejen na omrežju Etherium. Če na menjalnici plačujemo provizije ob nakupu in prodaji z njihovo kriptovaluto, jih s tem lahko prepolovimo. Velik del novih uporabnikov je menjalnica pritegnila zaradi njihovega programa, kjer za vsako osebo, ki se registrira kot tvoj referent, prejmeš delež njenih plačanih provizij ob menjavah.

2.3.3 KuCoin

KuCoin [19], ustanovljen avgusta 2017 smo izbrali zato, ker se na njej najprej pojavi veliko novih kriptovalut. Torej je to pogosto prva menjalnica, kjer lahko kupimo neko popolnoma novo kriptovaluto, ki je komaj prišla na trg. Po številu kriptovalut, ki so na voljo, je primerljiva z menjalnico Binance. Razlika med njima pa je v tem, da Binance ne dodaja kriptovalut, ki so še tako nove, in raje počaka na takšne, ki so že bolj stabilne.

Tudi KuCoin menjalnica ima svojo kriptovaluto, imenovano KuCoin Shares. Če to kriptovaluto posedujemo na njihovi menjalnici, si lahko s tem zmanjšamo provizije ob nakupu in prodaji za do 30 % [26]. Poleg tega pa prejemamo tudi tako imenovani KuCoin bonus, ki je odvisen od dnevnega prometa menjalnice in števila KuCoin Shares, ki jih posedujemo. Bonus se izplačuje vsakodnevno, sestavljen pa je iz 50 % dnevnih provizij, ki jih je menjalnica prejela. Bonus se razdeli med vse imetnike kriptovalute.

2.3.4 Poloniex

Ena izmed prvih popularnih menjalnic z velikim izborom kriptovalut za tiste čase. Ustanovljena je bila januarja 2014 s sedežem v Delaware, ZDA [27]. Zaradi slabega skaliranja menjalnice in velikega navala novih uporabnikov je pogosto delovala počasi. Zaradi teh problemov so novi in stari uporabniki iskali alternative in odhajali iz menjalnice Poloniex.

Posebnost te menjalnice je tudi trgovanje z vzvodom, kjer si lahko na nek način izposodimo kriptovalute. V letu 2018 jo je za okoli 400 milijonov dolarjev kupilo podjetje Circle [28].

2.3.5 Bitfinex

Bitfinex [18] je s svojim delovanjem začel leta 2012 kot menjalnica Bitcoinov. Prvotno so se torej ukvarjali le s trgovanjem Bitcoinov, kasneje pa so dodajali tudi nove kriptovalute in s tem skrbeli, da niso padli v zaostanek za ostalimi menjalnicami, ki so nastajale. Podjetje je trenutno registrirano na britanskih Deviških otokih, s sedežem v Hong Kongu [29]. Po dnevnem prometu je izmed naših menjalnic na drugem mestu, po številu kriptovalut na voljo pa na četrtem. V primerjavi z ostalima menjalnicama, ki ponujata fiat nakazila, ima Bitfinex na voljo veliko več različnih kriptovalutnih parov, kar poveča verjetnost, da bo uporabnik lahko ciljno kriptovaluto kupil kar na prvi menjalnici. Zaradi visokega števila kriptovalut na voljo je to za veliko uporabnikov edina menjalnica, ki jo uporabljajo od nakazila fiat denarja do nakupa ciljnih kriptovalut.

Ena izmed bolj kontroverznih povezav v svetu kriptovalut je povezava podjetja Bitfinex s kriptovaluto Tether [30]. Tether je kriptovaluta, katere ustvarjalci zagotavljajo, da je vsaka enota kriptovalute Tether podprta z eno enoto ameriškega dolarja. Kritiki pravijo, da to ne drži in obtožujejo Tether, da samo izdaja več kriptovalut brez povečevanja dejanskega finančnega položaja v dolarjih. Povezujejo ga tudi z manipulacijo cene kriptovalut pozno v letu 2017, ko je cena Bitcoina dosegla skoraj 20.000 dolarjev, saj naj bi

bil odgovoren za 50 % te ogromne rasti [31]. Preiskava povezave med rastjo kriptovalut in valuto Tether je opisana v članku [32].

2.3.6 Kraken

Menjalnica Kraken [17] s sedežem v San Franciscu je bila ustanovljena leta 2011 kot menjalnica s pari med valutami EUR, BTC in LTC. Trenutno se nahaja na prvem mestu po likvidnosti menjalnega para BTC/EUR. Je prva menjalnica kriptovalut, ki je imela trenutni tečaj Bitcoina in njegov volumen prikazan na Bloomberg terminalu.

Zaradi možnosti nakazila SEPA in večje ponudbe kriptovalut kot Bitstamp, je pogosto izbira uporabnikov, ki živijo v Evropski uniji, da se registrirajo na zgolj eni menjalnici, kjer lahko dobijo izbrano kriptovaluto. Njihov vmesnik je sicer manj optimiziran kot konkurenčni, zaradi česar se uporabniki, ki redno izvršujejo menjave, raje poslužujejo drugih menjalnic.

2.3.7 Bittrex

Menjalnico, ustanovljeno leta 2014, s sedežem v ameriškem mestu Seattle, so ustvarili trije strokovnjaki za računalniško varnost. Poleg Bitfinexa in Poloniexa jo lahko štejemo kot pionirje menjalnic s kriptovalutami. Verjetno ravno zaradi ozadja ustanoviteljev ni bila deležna nobenega vdora v sistem [33].

Dobri odnosi z uporabniki, pogosto posodabljanje in dodajanje novih kriptovalut so Bittrex [21] pripeljali na prvo mesto po številu kriptovalut na menjalnici izmed naših izbranih menjalnic. Ko pa to povežemo še z dejstvom, da se v menjalnico ni zgodil noben vdor, opazimo, zakaj je ravno ta menjalnica prva izbira mnogih trgovalcev kriptovalut po vsem svetu.

2.4 Iskanje optimalnih poti

Iskanje optimalne poti si lahko razlagamo na več različnih načinov. Iščemo lahko najhitrejšo pot, najcenejšo pot, najbolj likvidno pot ali pa uporabniku

najbolj prijazno pot. Pri iskanju najhitrejše bi tako vzeli čas transakcije med eno menjalnico in drugo ter fiat nakazila. Pri najcenejši poti gledamo stroške, ki nastanejo ob nakupih in pošiljanju, za najbolj likvidno preverimo, kakšen je promet za določeno kriptovaluto v zadnjih 24 urah. Če pa bi iskali uporabniku najbolj prijazno pot, pa bi verjetno izbrali menjalnice, ki jih že uporablja, ali pa svetovali registracijo na takšnih, kjer je to najbolj enostavno.

Pri gradnji naše rešitve smo se odločili kot optimalno izbrati pot, po kateri bomo imeli na cilju največje možno število ciljne kriptovalute.

Glede na to, za katero optimalno pot se odločimo, ustvarimo ustrezen graf z uteženimi povezavami. Za iskanje poti po takšnem grafu naletimo na klasičen problem iskanja najkrajše poti med dvema vozliščema v omrežju. Najbolj znan algoritem za reševanje takšnega problema je verjetno algoritem Dijkstra [34] s časovno zahtevnostjo $\mathcal{O}(|E| + |V| \log |V|)$. Za našo uporabo pa žal ne pride v poštev, saj ne omogoča iskanja po omrežju z negativno uteženimi povezavami. Izbrani algoritem za izdelavo naše rešitve je tako postal Bellman Ford [35], ki je s časovno zahtevnostjo $\mathcal{O}(|V||E|)$ sicer počasnejši, vendar dovoljuje negativno utežene povezave. Poleg iskanja najkrajše poti po omrežju bi lahko algoritem Bellman Ford uporabili tudi za detekcijo negativnih ciklov.

2.5 Sorodne rešitve

Na spletu obstaja nekaj rešitev, ki nudijo storitev nakupa kriptovalut brez računov na posameznih menjalnicah. Te storitve verjetno uporabljajo podobne pristope iskanja optimalnih poti pri nakupih kriptovalut kot naša rešitev. V naslednjih podpoglavjih bomo opisali dve takšni storitvi.

Podobno rešitev pa smo našli tudi v članku [36], kjer se opisana rešitev od naše razlikuje tako, da ne išče poti od začetne fiat valute do končne kriptovalute, ampak najugodnejše cikle. Pri opisani rešitvi je bilo ugotovljeno, da lahko pride do ugodnih ciklov, preko katerih pridobimo na vrednosti, vendar je pogosto najugodnejši cikel izmed vseh možnih, še vedno takšen, da bi

preko njega izgubili na vrednosti.

2.5.1 Changelly

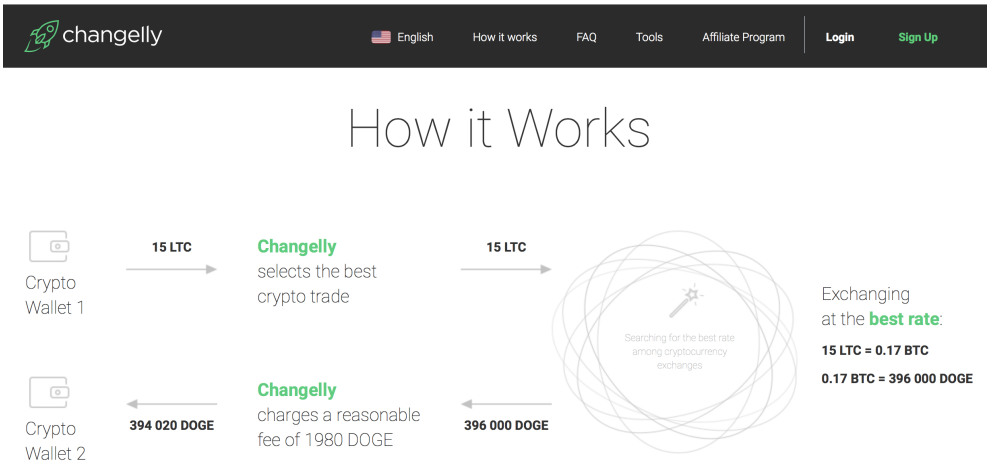
Prva sorodna rešitev, ki smo jo našli, je storitev Changelly [37]. Gre za spletno aplikacijo, ki nam ponuja enostaven način nakupa oziroma menjave izbrane kriptovalute. Na strani imajo seznam kriptovalut, ki jih je mogoče kupiti, in seznam valut, s katerimi je mogoče plačati. Tako lahko na strani izvedemo nakup izbrane kriptovalute preko kreditne kartice ali pa obstoječe kriptovalute, ki jo nakažemo na podani naslov. Celotna aplikacija daje glede na njeno zasnovo prednost ravno menjavam ene kriptovalute za drugo in ne direktnim nakupom s kreditno kartico (čeprav so omogočeni). Z našo rešitvijo jo lahko primerjamo preko skupne začetne valute, USD in končne kriptovalute, ki je na voljo pri obeh aplikacijah (glej podrobnosti v poglavju 5.1).

V ozadju aplikacije so verjetno uporabniški računi storitve Changelly, ki izvršijo menjave glede na optimalno pot za nakup ciljne kriptovalute na podlagi začetne valute in vrednosti. Po izvršenih menjavah aplikacija pošlje ciljno kriptovaluto na naslov denarnice, ki jo je vnesel uporabnik. Strošek, ki ga storitev Changelly zaračunava, je provizija v višini 0,5 % nakupa. Prav tako pa velja omeniti, da ob nakupu s kreditno kartico ta strošek narase na 5 % s strani storitve Changelly in dodatnih 5 % za posrednika Simplex, ki procesira kreditne kartice.

2.5.2 Coinmama

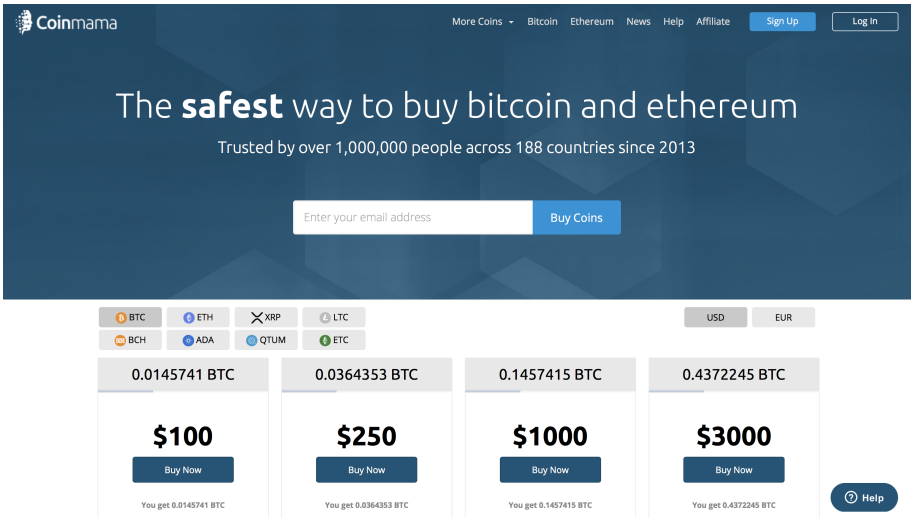
Spletna storitev Coinmama [38] je namenjena nakupu kriptovalut s kreditno kartico ali pa preko dveh kriptovalut, ki jih sprejemajo kot plačilno sredstvo. Zaradi poslovanja s fiat denarjem in kreditnimi karticami so stroški pri storitvi višji in znašajo 5.9 % pri nakupu kriptovalut ter dodatnih 5.0 % pri nakazilu denarja s kreditno kartico. Če stroške primerjamo s storitvijo Changelly, opazimo, da so ob enakih pogojih malenkost višji.

Aplikacija nam ponuja izbiro 8 različnih kriptovalut, ki jih lahko kupimo.



Slika 2.4: Spletna aplikacija Changelly.

Glede tega je bolj omejena kot Changelly in naša razvita rešitev. Zaradi manjše izbire je možno, da posluje samo na eni menjalnici kriptovalut in ji tako niti ni potrebno iskati optimalne poti po več menjalnicah. Lahko pa vseeno izvaja iskanje najcenejše poti po različnih menjalnih parih znotraj iste menjalnice in tako zagotovi bolj konkurenčne cene ali višje profite.



Slika 2.5: Spletna aplikacija Coinmama.

Poglavje 3

Načrt izdelave

Znotraj načrta izdelave bomo predstavili tehnologije, uporabljene pri izdelavi rešitve. Te so za izdelavo samega algoritma, programski jezik Python in podatkovna baza MongoDB, za prenos na spletno okolje pa ogrodji Flask in Vue. Opisali bomo celotno arhitekturo nastalega sistema. Kako uporabnik preko čelnega dela komunicira z algoritmom in iz kod le-ta prejme podatke, potrebne za izdelavo optimalne poti. Poglavje bomo zaključili z detajlnim načrtom omrežja odvisnosti. Tam bomo opisali, kako so predstavljena vozlišča, povezave in uteži pri gradnji ter podali primer izdelanega omrežja in optimalne poti v njem.

3.1 Uporabljene tehnologije

3.1.1 Python

Python je tolmačitveni programski jezik, ki je nastal leta 1990 in podpira tudi objektno usmerjeno programiranje [39]. Njegov najpogostejši primer uporabe je razvijanje spletnih aplikacij, podatkovno rudarjenje in strojno učenje. Pri razvijanju spletnih aplikacij so na voljo ogrodja Django, Flask, Jade in mnoga druga. V diplomski nalogi bomo za prenos na spletno okolje uporabili ogrodje Flask [40]. Primer preproste funkcije, ki izpiše znakovni niz in vrednost spremenljivke v tem programskem jeziku, vidimo na odseku

kode 3.1.

```
def primer_funkcije(ime_osebe):
    print("Pozdravljen, ", ime_osebe, "!")

# Klic funkcije s parametrom
primer_funkcije("Janko")
```

Slika 3.1: Primer funkcije in izpisa v programskem jeziku Python. Izpisana vrednost je 'Pozdravljen, Janko!'.

Za izdelavo aplikacije v programskem jeziku Python smo se odločili zaradi velikega števila knjižnic, ki so na voljo in poznavanjem jezika. Tako smo za ekstrakcijo podatkov s spletnih strani uporabili knjižnico BeautifulSoup [41], pri izdelavi omrežja pa knjižnico NetworkX [42]. V pomoč je prišla tudi knjižnica FuturesSession [43], s pomočjo katere smo lahko klice API na menjalnice izvajali asinhrono. S tem smo pospešili čas izvajanja teh klicev za približno 4-krat.

3.1.2 MongoDB

MongoDB spada med tako imenovane podatkovne baze NoSQL [44]. Za njih je značilno, da za povezovanje podatkov ne uporabljajo klasičnega pristopa tabel in relacij, kot to počnejo relacijske podatkovne baze. Prednost takšnih novih pristopov je predvsem povečana skalabilnost podatkovne baze in hitrost. Seveda pa obstajajo tudi slabosti nerelacijskih podatkovnih baz, pogosto v konsistentnosti podatkov in izolacijah transakcij [45].

Med bolj znane podatkovne baze NoSQL sodijo Cassandra, MongoDB in Neo4j. Podatkovna baza MongoDB namesto tabel in relacij za shranjevanje uporablja dokumente, zapisane v formatu BSON, ki je binarna oblika formata JSON [46]. Za enolično identifikacijo shranjenih dokumentov uporablja posebni ključ z nazivom '_id', ki se nastavi samodejno, ali pa ga podamo sami. Format JSON je zelo podoben slovarju v programskem jeziku Python, kar

nam omogoča enostavno shranjevanje in ohranjanje podatkovne strukture. Aplikacija bo v podatkovni bazi hranila najnovejše podatke v obliki slovarjev. Zaradi tega ne potrebujemo relacij med podatki, temveč zgolj prostor, kjer lahko ohranimo strukturo podatkov. Za uporabo podatkovne baze MongoDB v programskem jeziku Python smo uporabili knjižnico PyMongo [47]. V spodnjem primeru (glej sliko 3.2) si lahko ogledamo inicializacijo podatkovne baze in shranjevanja podatkov v obliki JSON oziroma BSON.

```
from pymongo import MongoClient
client = MongoClient('localhost', 27017)
# Ustvari bazo z imenom 'database'
db = client['database']

nova_oseba = {"ime": "Janko",
        "priimek": "Novak"}

# Zapis v podatkovno bazo
db.osebe.insert_one(nova_oseba)
```

Slika 3.2: Primer povezovanja, ustvarjanja in vnosa podatkov v MongoDB podatkovno bazo.

3.1.3 Flask in Vue

Za izdelavo spletne aplikacije smo se odločili uporabiti okolji Flask in Vue. Čelni del aplikacije bo tako ustvarjen s pomočjo okolja Vue js, zaledni del pa v okolju Flask.

Flask je mikro okolje za izdelavo spletnih aplikacij v programskem jeziku Python. Takšno mikro okolje je optimalno za izdelavo naše aplikacije, saj izdelujemo zelo osnovno spletno aplikacijo na eni strani. Za razliko od večjih programskih okolij za izdelavo spletnih aplikacij v Pythonu, kot je na primer Django, Flask vsebuje majhno število odvisnosti do zunanjih knjižnic [48]. Tako sicer nekaterih funkcionalnosti, kot je na primer registracija uporabnikov, ne dobimo avtomatsko in bi jo v primeru uporabe morali napisati sami

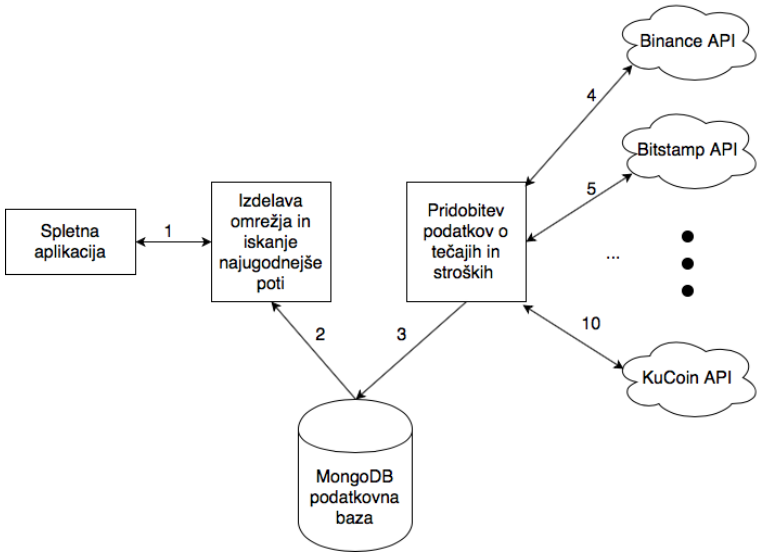
ali pa uporabiti ustrezno knjižnico in s tem dejansko izničili bistvo mikro okolja. Odločitev za zaledni del v okolju Flask izhaja iz uporabe enakega programskega jezika kot pri izdelanem algoritmu in preprostosti tega mikro ogrodja.

Vue je okolje za izdelavo čelnega dela spletnih aplikacij, napisano v programskem jeziku Javascript. Uporaba okolja je zelo široka, saj ga lahko uporabimo samo na delu obstoječe spletne aplikacije ali pa okoli njega zgradimo celotno. Prav tako je v okolju Vue možno ustvarjati napredne spletne aplikacije tipa SPA, kjer je ena spletna stran razdeljena na več odsekov različnih HTML, CSS in JS datotek [49]. Za čelni del v okolju Vue smo se tako odločili ravno zaradi SPA pristopa izdelave spletne aplikacije.

3.2 Arhitektura sistema

Celoten sistem bo sestavljen iz štirih večjih delov, ki komunicirajo med seboj. Prvi, zaledni, del aplikacije v enakomernih intervalih izvaja klice na menjalnice preko dostopnih točk API. Pod zaledni del aplikacije spada tudi program, ki bo ustvaril omrežje odvisnosti menjalnih parov in po njih iskal optimalno pot. Podatki, ki jih pridobimo iz točk API in kasneje uporabimo pri gradnji omrežja, bodo hranjeni v podatkovni bazi MongoDB. Čelni del aplikacije bo komuniciral s programom za izdelavo omrežja in iskanje optimalne poti.

Na sliki 3.3 so številčno označene povezave med posameznimi deli aplikacije. Povezava 1 bo vsebovala podatke, ki jih uporabnik vnese v aplikacijo. To so podatki o želeni ciljni kriptovaluti, začetni valuti (EUR ali USD) in količini začetne valute. Povezava 2 pridobi iz podatkovne baze vse podatke, ki so potrebni za izdelavo omrežja odvisnosti. Povezava 3 zapisuje podatke, ki smo jih pridobili preko klicev API v podatkovno bazo. Preostale povezave, oštevilčene od 4 do 10, pa predstavljajo klice na dostopne točke API in odzive s podatki v JSON obliki. Več o dejanskem pridobivanju in filtriranju podatkov je opisano v poglavju 4.1.



Slika 3.3: Arhitektura spletne aplikacije.

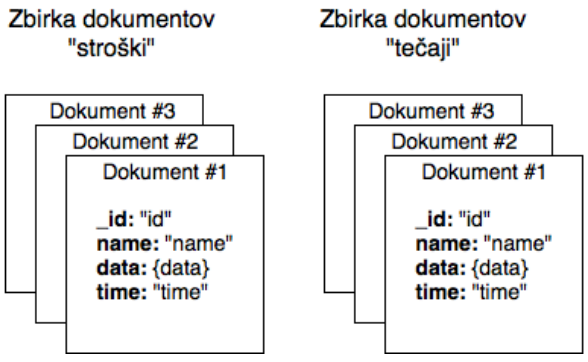
Preko klicev API iz menjalnic pridobimo podatke o tečajih kriptovalut in stroških menjave ter pošiljanja. Klici se izvajajo v intervalih, in sicer klici za posodobitev tečaja vsakih 5 minut, klici za posodobitev stroškov pa vsakodnevno. Razlog za različne posodobitve je hitrost spreminjanja posameznih podatkov. Tako se na posamezni časovni interval zgodi nov zapis, oziroma ker hranimo samo najnovejše vrednosti lahko rečemo, da se zgodi kar prepis v podatkovni bazi.

Pri samemu tečaju kriptovalutnih parov bi bilo še bolj optimalno pridobivanje podatkov v živo ali na primer vsakih nekaj deset sekund. Zaradi prosto dostopnih in brezplačnih dostopnih točk API, pa to pri večini menjalnic ni mogoče, saj bi bili s tem njihovi strežniki potencialno preobremenjeni. Pri pridobivanju stroškov pošiljanja in provizij pri menjavi smo se omejili na 24 ur, saj so spremembe manj pogoste.

3.3 Shema podatkovne baze MongoDB

Podatke, ki jih pridobimo preko dostopnih točk API in ostalih načinov, opisanih v poglavju 4.1, hranimo v podatkovni bazi MongoDB. Znotraj podatkovne baze imamo dve zbirki dokumentov. Prva izmed zbirk dokumentov vsebuje podatke o stroških menjalnic, druga pa podatke o trenutnih tečajih, podatke s strani CoinMarketCap in tečaj za fiat par EUR/USD. Za enolično identifikacijo dokumentov znotraj zbirke dokumentov služi avtomatsko generiran ključ '_id', ki smo ga omenili v poglavju 3.1.2, dodatno pa smo dodali še ključ 'name', ki kot vrednost vsebuje ime menjalnice oziroma strani, iz katere smo pridobili podatke. Tudi ta ključ enolično identificira dokument znotraj posamezne zbirke dokumentov.

Znotraj obeh zbirk dokumentov se nahajajo dokumenti enakih oblik, kot vidimo na sliki 3.4, razlikujejo se le v tem, kaj dejansko hranimo v atributu 'data'. Pri zbirki dokumentov, imenovanih stroški, v atributu hranimo slovar vseh stroškov, ki se delijo na nakazila na menjalnico, nakazila iz menjalnice in provizijo menjalnice, medtem ko pri zbirki dokumentov, imenovani tečaji, hranimo dvonivojski slovar, ki vsebuje podatke o vseh menjalnih parih in njihovih trenutnih tečajih. Pri posebnih primerih, kot so na primer podatki s spletne strani CoinMarketCap, je slovar enonivojski in vsebuje oznako kriptovalute ter njeno ceno v valuti USD.



Slika 3.4: Shema podatkovne baze mongoDB.

3.4 Načrt omrežja odvisnosti menjalnih parov

Pri gradnji omrežja potrebujemo podatke o menjalnih parih, njihovih tečajih, imenih menjalnic in stroškov pošiljanja ter provizij. Vozlišče smo predstavili kot kriptovaluto v menjalnici, povezavo pa kot pošiljanje oziroma menjavo te kriptovalute.

Poleg običajnih vozlišč, ki vsebujejo enake tipe podatkov, bomo v omrežju zgradili tudi dve posebni vozlišči. To sta vozlišči START in GOAL, ki služita kot začetno in končno vozlišče. Potrebujemo ju, ker je vozlišč s ciljno kriptovaluto lahko več, s tem pa pridobimo eno končno vozlišče. Povezave med vozliščem s ciljno kriptovaluto in vozliščem GOAL bodo seveda imele utež, enako 0. Vozlišče START pa služi začetnemu izhodišču, iz katerega potekajo povezave na menjalnice, ki sprejemajo fiat nakazila denarja.

Podatki, ki jih bo vsebovalo vsako običajno vozlišče v omrežju, so:

- Ime vozlišča, preko katerega izvemo ime menjalnice in valuto. Primer poimenovanja za menjalnico Bitstamp in kriptovaluto Bitcoin je *bitstamp_BTC*. Za takšen način poimenovanja smo se odločili, ker z njim pridobimo enolična imena in podatke o vozlišču.

- Podatek o proviziji pri menjavi. Podatek je zapisan kot delež, ki ga menjalnica računa ob menjavi. Provizija pri menjavi je znotraj iste menjalnice enaka za vse menjalne pare. Provizijo, v višini 0,1 % bi tako zapisali kot *0,001*.

- Strošek pošiljanja valute iz te menjalnice. Razlikujejo se glede na valuto in menjalnico. Zapisan je v enaki valuti kot je valuta vozlišča.

- Podatek o količini valute v tem vozlišču. Ta podatek potrebujemo predvsem pri gradnji samega omrežja, saj preko njega in tečaja menjave izračunamo novo količino ter vrednost novega vozlišča.

- Vrednost tega vozlišča v USD. Podatek uporabimo pri uteževanju povezav med vozlišči. Vrednost je zapisana v USD,ker je pri kriptovalutah to bolj pogosta pretvorba kot pa na primer v EUR. Vrednost bi lahko alternativno hranili tudi v valuti BTC.

Načrtovani podatki, ki jih bo hranila vsaka povezava med vozlišči, so naslednji:

- Podatek o tipu povezave. Povezave smo razdelili na dva različna tipa, in sicer 'e' in 's'. Tip povezave 'e' nam pove, da gre za povezavo znotraj iste menjalnice, zgodi pa se menjava ene valute za drugo. Povezava tipa 's' nam pove, da gre za povezavo iz ene na drugo menjalnico. Povezava tega tipa ima na obeh vozliščih enako valuto, a drugo menjalnico. Primer povezave tipa 'e' je menjava valute EUR za BTC na menjalnici Bitstamp, medtem ko je primer povezave tipa 's' pošiljanje valute BTC iz menjalnice Bitstamp na menjalnico Binance.

- Podatek o teži povezave v valuti USD. Uteži povezav potrebujemo pri iskanju optimalnih poti, saj po navadi iščemo najkrajšo oziroma najcenejšo pot. Izračun teže povezave je odvisen od tipa povezave. Pri teži povezave je uporabljena valuta USD, saj obstaja veliko različnih kriptovalut, ki bi jih morali v nasprotnem primeru kasneje pretvarjati v skupno valuto.

Kot smo opisali že zgoraj, je izračun teže povezave odvisen od tipa povezave. Težo povezave med vozliščema N1 in N2 izračunamo na način, prikazan na sliki 3.5. Pri računanju vrednosti (angl. *value*) vozlišča smo uporabili pristop množenja količine kriptovalute v vozlišču s ceno te kriptovalute v valuti USD (pridobljene s spletne strani CoinMarketCap).

V vozlišču ohranjamo samo najvišjo količino valute in njeno vrednost v USD. Če pride v vozlišče nova povezava, ki bi prinesla nižjo količino, to količino upoštevamo samo pri utežitvi nove povezave, vrednosti v vozlišču pa ne spreminjamo. V primeru, ko pride nova povezava, ki prinese višjo

$$weight(N1, N2) = \begin{cases} N1.sending\_fee, & \text{tip='s'} \\ \\ N1.value - N2.value, & \text{tip='e'} \end{cases}$$

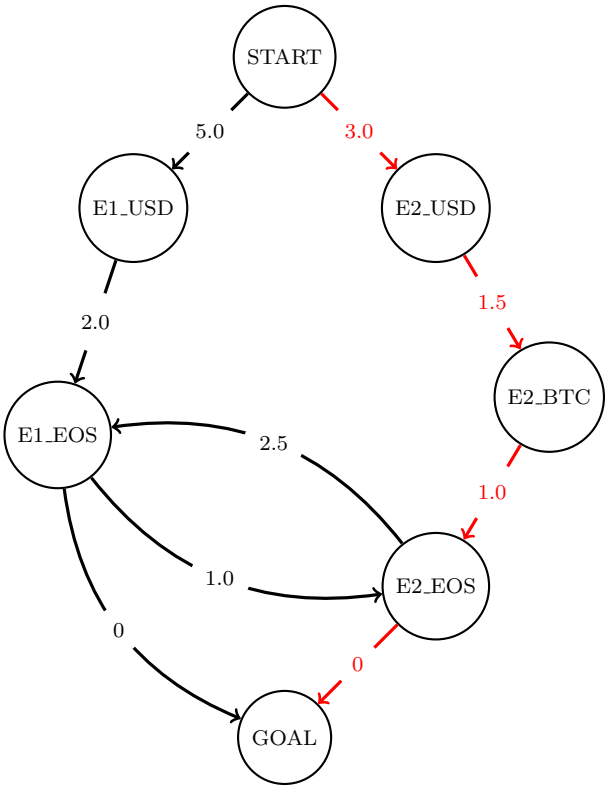
Slika 3.5: Izračun teže povezave glede na tip povezave.

količino pa jo upoštevamo pri utežitvi nove povezave in spremenimo količino ter vrednost v ciljnem vozlišču.

Na spodnjem primeru si bomo ogledali primer omrežja in optimalne poti v njem. Za imena menjalnic v vozliščih je uporabljen znak E. Povezave med vozlišči so naštete v tabeli 3.1. Pri primeru predpostavljamo, da je šlo za nek izmišljeni začetni znesek, provizije ob menjavi in stroški pošiljanja so prav tako znani le kot uteži povezav. Na nastalem omrežju na sliki 3.6 tako vidimo povezave med vozlišči. Opazimo, da optimalna pot vsebuje več menjav kot alternativna, a je kljub temu boljša. Tudi v nadaljevanju diplomske naloge bomo opazili takšne primere na pravih podatkih. Kot smo določili zgoraj, so končne povezave do vozlišča GOAL utežene z 0, ostale pa glede na njihov tip.

Začetek	Cilj	Tip povezave	Utež
START	E1_USD	's'	5,0
START	E2_USD	's'	3,0
E1_USD	E1_EOS	'e'	2,0
E2_USD	E2_BTC	'e'	1,5
E2_BTC	E2_EOS	'e'	1,0
E1_EOS	E2_EOS	's'	1,0
E2_EOS	E1_EOS	's'	2,5
E1_EOS	GOAL	'e'	0
E2_EOS	GOAL	'e'	0

Tabela 3.1: Podatki o povezavah med vozlišči.



Slika 3.6: Primer omrežja. Pot označena z rdečo barvo je optimalna.

Poglavje 4

Implementacija predloga rešitve

V prejšnjem poglavju smo predstavili tehnologije, ki smo jih uporabili in načrt za izdelavo aplikacije. V nadaljevanju bomo opisali postopek izdelave predloga rešitve. Na začetku poglavja si bomo ogledali pridobivanje in filtriranje podatkov, prejetih s strani menjalnic in ostalih relevantnih spletnih strani. Sledi pregled postopka izdelave omrežja v Pythonovi knjižnici NetworkX in dejanska izdelava vozlišč ter povezav med našimi vozlišči. V nastalemu omrežju seveda obstaja več poti do ciljne kriptovalute, vendar je pogosto le ena optimalna. Ogledali si bomo primerjavo te optimalne poti, z ostalimi možnimi potmi do cilja. Zaradi že omenjenih razlogov smo izbrali algoritem Bellman Ford, katerega izbiro bomo potrdili s podpoglavjem o negativno uteženih povezavah in s primerom takšne povezave. Prav tako bomo predstavili primer rešitve, kjer optimalna pot ni tista, za katero je potrebno najmanjše število menjav. Poglavje bomo zaključili s prenosom rešitve na spletno okolje in prikazom spletne aplikacije.

4.1 Pridobivanje in filtriranje podatkov

Za pridobivanje podatkov o trenutnih tečajih smo uporabili točke API, ki so jih posamezne menjalnice ponujale. Obstajajo omejitve dostopa do teh točk, ki pa jih zaradi klica funkcij na dovolj velike časovne intervale nismo presegli.

Pri pridobivanju cen transakcij, provizij in tečajev smo pri menjalnicah, ki so to omogočale, dostopali do točk API (glej tabelo 4.1). Pri menjalnicah, ki takšnih dostopnih točk niso imele na voljo, pa smo uporabili pristop pridobitve HTML dokumenta spletne strani, s katerega smo nato izluščil podatke o cenah transakcij posameznih kriptovalut. Pri pridobivanju provizije določene menjalnice pa smo vrednosti zapisali direktno v program, saj gre za eno vrednost, ki je enotna celotni menjalnici. Pri provizijah smo upoštevali vrednost brez kakršnih koli bonitet. Tako nismo upoštevali zmanjševanja provizije pri menjalnicah KuCoin in Binance, ki bi jih lahko dosegli z nakupom njihovih kriptovalut in njihovi uporabi za plačevanje provizij. Podatke, ki smo jih prejeli, smo nato shranili v podatkovno bazo MongoDB. Kot smo omenili že pri utemeljitvi izbora podatkovne baze v poglavju 3.1.2, bomo hranili samo najnovejše podatke. Vrednosti so shranjene v obliki slovarja, ki omogoča hiter dostop do želenih podatkov.

Pri pridobivanju podatkov s strani menjalnic smo se srečali tudi z nekaj problemi. Prvi problem, ki je bil dejansko samo majhna ovira, je bil ta, da so nekatere menjalnice omogočale dostop do cen transakcij preko vmesnika API, medtem ko so jih druge imele zapisane samo v tabeli. To smo rešili z zgoraj opisanim postopkom izluščevanja podatkov. Naslednji problem je različno poimenovanje kriptovalut na različnih menjalnicah. Nekatere menjalnice so se tako odločile, da za ime kriptovalute uporabijo zgolj 3 znake. Tako je pri pridobivanju podatkov iz menjalnice Bitfinex prišlo do problemov za kriptovalute MIOTA (IOT), QTUM (QTM), MITH (MTH). Neskladno z ostalimi je tudi poimenovanje kriptovalute BTC na menjalnici Kraken, ki ima ime XBT. Zaradi takšnih neskladij pri poimenovanju smo morali take primere identificirati in jih skladno preimenovati. Do teh problemov lahko pride tudi pri sicer redkih dogodkih, ko se ime kriptovalute spremeni. Ob takšnih primerih nekatere menjalnice ime posodobijo takoj, spet druge rabijo za to več časa.

Menjalnica	Stroški API	Tečaji API	Provizija
Bitstamp	/	`https://www.bitstamp.net/api/v2/trading-pairs-info/`	0,25 %
Bitfinex	/	`https://api.bitfinex.com/v1/symbols`	0,20 %
Kraken	/	`https://api.kraken.com/0/public/AssetPairs`	0,26 %
KuCoin	`https://api.kucoin.com/v1/market/open/coins`	`https://api.kucoin.com/v1/market/open/symbols`	0,10 %
Binance	`https://www.binance.com/assetWithdraw/getAllAsset.html`	`https://api.binance.com/api/v3/ticker/price`	0,10 %
Bittrex	`https://bittrex.com/api/v1.1/public/getcurrencies`	`https://bittrex.com/api/v1.1/public/getmarketsummaries`	0,25 %
Poloniex	`https://poloniex.com/public?command=returnCurrencies`	`https://poloniex.com/public?command=returnTicker`	0,25 %

Tabela 4.1: Podatki o dostopnih točkah API in menjalnih provizijah pri menjalnicah.

4.1.1 Menjalnice kriptovalut

Funkcije, ki dostopajo do menjalnic in pridobivajo podatke, smo izvedli v različnih časovnih intervalih. Ker se stroški pošiljanja spreminjajo bolj redko, smo funkcijo nastavili na ponovitev izvajanja vsakih 24 ur. Tečaji se spreminjajo bolj pogosto, zato smo funkcijo nastavili na ponovitev vsakih 5 minut. V primeru, da bi bil tudi ta časovni interval predolg, bi ga lahko tudi skrajšali, vendar obstajajo omejitve glede frekvence dostopanja do dostopnih točk API. Poleg tečajev in stroškov pošiljanja ter menjav smo pridobili tudi podatke o cenah kriptovalut v valuti USD s strani CoinMarketCap [1, 50] in menjalni tečaj za par EUR/USD [51].

Primer JSON odziva iz dostopne točke API menjalnice Poloniex je prikazan na sliki 4.1. Uporabili smo podatke o menjalnemu paru in zadnjo ceno ("last"). Menjalni par smo pridobili tako, da smo naziv para razdelili na dva dela (BTC in BCN) in tako shranili vrednosti kot slovar, prikazan na sliki 4.3.

V funkciji, prikazani na sliki 4.2, smo si podatke shranili na način, da bomo lahko do njih enostavno in hitro dostopali. Primer izgleda strukture, v kateri smo shranili podatke, si lahko ogledamo na sliki 4.3. Prvi nivo slovarja hrani kot ključ naziv kriptovalute, kot vrednost pa slovar njenih menjalnih parov in menjalnega tečaja. Drugi nivo slovarja tako hrani kot ključ ime kriptovalute drugega dela para, kot vrednost pa menjalni tečaj prvega dela para z drugim. Primer iz spodnjih podatkov bi bila menjava kriptovalute ETH za LSK po menjalnemu tečaju 1 LSK = 0,01074519 ETH.

4.1.2 Vrednosti kriptovalut s strani CoinMarketCap

Pri določanju vrednosti posameznega vozlišča ali stroška pošiljanja (oba podatka v valuti USD), smo uporabili vrednost s spletne strani CoinMarketCap. Vrednost je na strani izračunana kot povprečna vrednost tečajev vseh menjalnic, ki s to kriptovaluto trgujejo. Zaradi te povprečne vrednosti lahko v nekaterih primerih vrednost odstopa od realne na menjalnici, ki jo trenutno

```
{
    "BTC_BCN": {
        "baseVolume": "41.09105285",
        "high24hr": "0.00000050",
        "highestBid": "0.00000048",
        "id": 7,
        "isFrozen": "0",
        "last": "0.00000048",
        "low24hr": "0.00000047",
        "lowestAsk": "0.00000049",
        "percentChange": "0.00000000",
        "quoteVolume": "84286006.66310813"
    }
}
```


Slika 4.1: Primer odgovora v JSON obliki iz menjalnice Poloniex.

```
def getPairsAndPricesAtPoloniex(api_data):
    data = api_data[1]

    prices_at_poloniex = {}
    for key in data.keys():
        prices_at_poloniex.setdefault(key.split("_")[0],
                                      {}).update({key.
                                      split("_")[1] :
                                      float(data[key]['
                                      last'])})

    return prices_at_poloniex
```

Slika 4.2: Urejanje in shranjevanje JSON odgovora v Python slovar.

```
{
    "BTC":{
        "BCN": 3.4e-07,
        "BLK": 1.525e-05, ...},
    "XMR":{
        "BCN": 2.147e-05, ...},
    "ETH":{
        "LSK": 0.01074519,
        "STEEM": 0.00315999, ...}
}
```


Slika 4.3: Struktura shranjenih menjalnih parov iz menjalnice Poloniex.

spremljamo. Največkrat lahko pride do takšne situacije pri kriptovalutah z manjšo tržno kapitalizacijo in s prometom. Pri pridobivanju podatkov s strani CoinMarketCap smo se omejili na 500 najvišje uvrščenih kriptovalut glede na njihovo tržno kapitalizacijo.

Na sliki 4.4 vidimo prvih 17 kriptovalut, uvrščenih glede na tržno kapitalizacijo in njihove atribute. Uporabili smo podatke o ceni v valuti USD. Pridobljene podatke smo shranili v slovar, tako da smo lahko do cene posamezne kriptovalute dostopali preko enostavnega klica v slovar, kjer so kot ključi uporabljeni nazivi kriptovalut. Tako smo lahko pri gradnji omrežja enostavno uporabili podatke o vrednosti posamezne kriptovalute v USD. S tem smo zagotovili enako ceno za posamezno kriptovaluto ne glede na to, v katerem vozlišču smo in tako poenotil primerjavo količine z vrednostjo. Če bi izbrali pristop cene na posamezni menjalnici, bi dobili bolj točne rezultate s strani vrednosti. Če bi iskali najcenejšo pot glede na vrednost, pa bi lahko prišlo do primerov, kjer bi dobili višjo vrednost, ampak nižjo količino. Tudi ta način izpeljave je torej odvisen od uporabnikovih nadaljnjih planov s ciljno kriptovaluto – takojšnja prodaja ali hranjenje.

#	Name	Market Cap	Price	Volume (24h)	Circulating Supply	Change (24h)	Price Graph (7d)
1	Bitcoin	$140,198,953,057	$8,160.76	$4,288,623,019	17,179,650 BTC	-0.18%	
2	Ethereum	$46,743,040,715	$462.76	$1,744,776,076	101,009,001 ETH	-0.47%	
3	XRP	$17,748,269,498	$0.451430	$198,084,959	39,315,683,476 XRP *	-0.02%	
4	Bitcoin Cash	$14,146,238,987	$819.36	$560,856,055	17,265,050 BCH	0.02%	
5	EOS	$7,319,204,942	$8.17	$650,825,040	896,149,492 EOS *	-0.92%	
6	Stellar	$5,641,862,399	$0.300620	$72,072,706	18,767,431,579 XLM *	-3.70%	
7	Litecoin	$4,792,182,885	$83.15	$285,178,632	57,630,782 LTC	-0.70%	
8	Cardano	$4,134,906,936	$0.159482	$49,404,640	25,927,070,538 ADA *	-1.53%	
9	IOTA	$2,774,222,942	$0.998091	$28,998,843	2,779,530,283 MIOTA *	-2.02%	
10	Tether	$2,500,962,640	$0.997536	$2,515,786,036	2,507,140,346 USDT *	-0.19%	
11	TRON	$2,428,968,055	$0.036944	$340,177,787	65,748,111,645 TRX *	-0.92%	
12	Monero	$2,152,594,464	$132.35	$29,320,046	16,264,830 XMR	-4.29%	
13	NEO	$2,144,815,744	$33.00	$66,306,080	65,000,000 NEO *	-2.22%	
14	Dash	$1,952,602,443	$237.46	$93,948,461	8,223,003 DASH	-0.67%	
15	Ethereum Classic	$1,758,805,378	$17.01	$198,596,543	103,405,783 ETC	1.46%	
16	NEM	$1,569,991,865	$0.174444	$21,642,757	8,999,999,999 XEM *	-0.88%	
17	VeChain	$1,401,384,576	$2.53	$1,281,590	554,545,494 VEN *	-3.15%	

Slika 4.4: Prikaz prvih 17 kriptovalut na strani CoinMarketCap (dne 30.7.2018).

4.2 Izdelava omrežja odvisnosti menjalnih parov

Za izdelavo dejanskega omrežja odvisnosti smo uporabili pristop, ki je opisan v poglavju 3.4 in prikazan v psevdokodi 1. Pri izdelavi smo uporabili Pythonovo knjižnico NetworkX [42, 52] , ki omogoča enostavno gradnjo omrežij in podpira ogromno funkcij po omrežjih. V odseku kode na sliki 4.5 si lahko pogledamo preprost primer, v katerem ustvarimo usmerjen graf, mu dodelimo nekaj vozlišč z atributi in jih povežemo med seboj s povezavami, ki prav tako vsebujejo atribute. Slika 4.6, ki predstavlja nastalo omrežje, je vizualizirana s funkcijo *draw_networkx*. Povezave so označene kot premice, kjer širši del premice predstavlja cilj povezave.

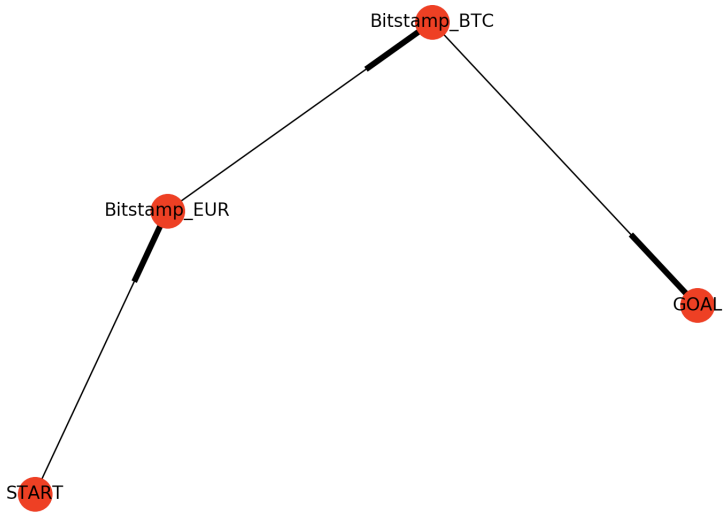
```
import networkx as nx

# Inicializacija usmerjenega grafa
G=nx.DiGraph(directed=True)

# Dodajanje vozlisc z atributi. Prvi atribut je ime
                                vozlisca.
G.add_node('START', value=0, count=0)
G.add_node('Bitstamp_EUR',  count=500,
    exchange_fee=0.002, sending_fee='no', value=500)

G.add_node('Bitstamp_BTC', count=0.11,
    exchange_fee=0.002, sending_fee=0.001, value=499)
G.add_node('GOAL', value=0, count=0)

# Dodajanje povezav med vozlisci.
G.add_edge('START','Bitstamp_EUR', type='s',  weight=0)
G.add_edge('Bitstamp_EUR','Bitstamp_BTC', type='e',
                                weight=1)
G.add_edge('Bitstamp_BTC', 'GOAL', type='s', weight=0)
```

Slika 4.5: Gradnja usmerjenega grafa s knjižnico NetworkX.

Algoritem 1 Izdelava celotnega omrežja.

```
1: Initilize network
2: for each exchange ∈ list_of_fiat_exchanges do
3:     network.nodes ← exchange_currency
4: end for
5: for each exchange ∈ list_of_all_exchanges do
6:     for each pair1, list_of_pairs2 ∈ exchange do
7:         for each pair2 ∈ list_of_pairs2 do
8:             network.nodes ← exchange_pair2
9:         end for
10:        network.nodes ← exchange_pair1
11:    end for
12: end for
13: for each exchange ∈ list_of_all_exchanges do
14:    for each cryptocurrency ∈ exchange do
15:        start_node ← cryptocurrency
16:        for each node ∈ network.nodes do
17:            if node.currency == cryptocurrency then
18:                network.edge ← (node.currency,
                                   cryptocurrency, type = s)
19:            end if
20:        end for
21:    end for
22:    for each pair1, list_of_pairs2 ∈ exchange do
23:        for each pair2 ∈ list_of_pairs2 do
24:            network.edge ← (pair1, pair2, type = e)
25:        end for
26:    end for
27: end for
```

Pri izdelavi omrežja smo ustvarili dodatni vozlišči START in GOAL. Njihov namen je narediti skupen začetek in konec za računanje optimalne poti. S takšnim pristopom je vsako vozlišče, ki vsebuje našo izbrano ciljno valuto (ne glede na menjalnico), povezano z vozliščem GOAL preko povezave, katere utež je 0. Prav tako pa iz vozlišča START potekajo povezave do menjalnic, ki sprejemajo izbrano fiat valuto. Povezave te vrste so uteženе s ceno, ki jo te menjalnice zaračunavajo ob pošiljanju fiat denarja k njim.



Slika 4.6: Grafični prikaz grafa nastalega iz zgornje kode.

4.2.1 Postopek izdelave vozlišč

Izdelave celotnega omrežja smo se lotili po korakih. Na začetku smo ustvarili vozlišča, ki so vsakič enaka, to sta zgoraj omenjeni START in GOAL vozlišči. V naslednjem delu algoritma smo ustvarili vsa vozlišča, ki vsebujejo fiat valuto. Izdelava teh vozlišč je odvisna od izbora začetne valute – v primeru, da smo izbrali valuto USD, se bodo ustvarila vozlišča, ki vsebujejo valuto USD. Skupno število teh vozlišč je 3, saj imamo med menjalnicami 3 takšne, ki sprejemajo fiat nakazila denarja. Posebnost ustvarjanja začetnih vozlišč je tudi ta, da je v njih zapisana vrednost in količina valute, saj jo poznamo z

nakazila. Ta vozlišča smo že v tem koraku tudi povezali z začetnim, START, vozliščem.

Sledi bolj splošen del algoritma, ustvarjanje vseh preostalih vozlišč, ki jih je skupno 723. Ta številka predstavlja vse kriptovalute, ki so dostopne na vsaki izmed menjalnic in ne število različnih kriptovalut. Ustvarjamo jih tako, da gremo skozi slovar vseh menjalnic, ki smo ga pridobili iz podatkovne baze. V vsakem slovarju menjalnice imamo nove slovarje, ki predstavljajo menjalne pare. Iz teh podatkov ugotovimo ime menjalnice in kratico kriptovalute, preko katerih lahko izdelamo vozlišče. Le-tem ob kreiranju zapisujemo vrednost in število valute enako 0, saj do njih še nismo ustvarili povezav in tako ne poznamo teh podatkov v vozlišču. Pri tem delu algoritma dodajamo tudi povezave do končnega, GOAL, vozlišča. V pogojnem stavku preverimo, ali je trenutna valuta enaka ciljni valuti in to vozlišče povežemo s ciljnim, kot to prikazuje slika 4.7.

```
    if (cryptocurrency == goal_currency):
        network.add_edge(node_name, 'GOAL', type='s',
                                            weight=0)
```

Slika 4.7: Povezava s končnim vozliščem - GOAL.

4.2.2 Postopek izdelave povezav med vozlišči

Nastala vozlišča je med seboj potrebno še povezati. Za izdelavo povezav in ustrezne zapise novih vrednosti in števila valute v posameznih vozliščih, moramo začeti v tistih vozliščih, ki že vsebujejo omenjene atribute. To so na začetku vozlišča, ki vsebujejo začetne fiat valute. Začnemo pri njih in postopoma dodajamo povezave ter osvežujemo atribute v ostalih vozliščih.

Kot smo omenili že v načrtu omrežja v poglavju 3.4, obstajata dva tipa povezav, pošiljanje in menjava. Pri povezavi tipa pošiljanje ('s') smo preverili vsa ostala vozlišča, ki vsebujejo isto valuto in do njih ustvarili povezavo z ustrezno utežjo.

Povezave tipa menjava ('e') pa se dogajajo znotraj iste menjalnice, tako da jih ustvarimo, ko iteriramo skozi slovar vsake menjalnice. Pri tem smo pozorni na izračun stroškov, ki nastanejo pri menjavi in posodabljanju vrednosti in števila valute v vozliščih, kot to prikazuje slika 4.8. V primeru, da je nova vrednost višja kot stara, posodobimo njo in količino valute.

```
if (network.node[edge_end]['value'] < new_value):
    network.node[edge_end]['value'] = new_value
    network.node[edge_end]['count'] = new_amount
```

Slika 4.8: Osveževanje atributov v vozlišču.

Pri določanju vrednosti posameznega vozlišča ali stroška pošiljanja smo uporabili vrednost te valute v USD s spletne strani CoinMarketCap. Vrednost na strani je pridobljena kot povprečna vrednost tečaja vseh menjalnic, ki s to valuto trgujejo. Zaradi tega lahko pri katerih kriptovalutah, ki imajo nižjo tržno kapitalizacijo in manj prometa, prihaja do odstopanj med to povprečno vrednostjo in dejansko vrednostjo na izbrani menjalnici.

4.3 Iskanje optimalnih poti po omrežju

Za iskanje optimalnih poti smo uporabili algoritem za iskanje najkrajše poti, Bellman Ford, ki je dostopen kot funkcija v knjižnici NetworkX. Razlog za izbiro algoritma Bellman Ford je v tem, da v omrežju obstajajo povezave z negativnimi utežmi (več v podpoglavju 4.3.1). Zaradi tega, kakšen drug algoritem, kot je na primer Dijkstra, ne deluje.

Za delovanje algoritma je potrebno povezavam podati atribut uteži. To smo storili, kot je opisano v poglavju 3.4. Uteži povezav so tako odvisne od tipa povezave, na podlagi katerega se izračuna utež v valuti USD. Izračunana je glede na strošek pošiljanja ali pa spremembo v vrednosti.

Kot optimalno pot smo si izbrali pot, po kateri na ciljnem vozlišču dobimo največjo količino in posledično vrednost. V primeru, da bi želeli uporabiti kakšne druge kriterije, bi morali prilagoditi samo računanje uteži za povezave.

Za naslednji primer smo v program vnesli želeno kriptovaluto: ETH, začetno fiat valuto: USD in količino: 500. Kot najcenejšo povezavo nam je vrnil naslednji seznam, ki nam pove postopek nakupov in menjav. Optimalna pot v našem primeru je torej pošiljanje USD na menjalnico Kraken, kjer jih zamenjamo za kriptovaluto ETH. Algoritem nam kot rezultat vrne seznam, prikazan na sliki 4.9.

```
['START', 'kraken_USD', 'kraken_ETH', 'GOAL']
```

Slika 4.9: Seznam, ki vsebuje potek optimalne poti za nakup kriptovalute ETH.

Poleg optimalne poti pa obstaja še več poti, po katerih lahko pridemo do želene kriptovalute. V tabeli 4.2 so prikazana vsa vozlišča, ki imajo povezavo, ki se konča v vozlišču GOAL. To so torej vsa vozlišča, ki vsebujejo ciljno kriptovaluto. V tabeli 4.2 vidimo razlike med količinami, ki jih dobimo na katerem vozlišču. V vrstici, kjer vidimo podatke iz vozlišča 'kraken_ETH', sta število in vrednost najvišji. S tem smo prepričani, da je algoritem izbral optimalno pot.

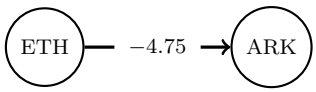
Naziv vozlišča	Število	Vrednost (v USD)
kucoin_ETH	1,1794	493,48
poloniex_ETH	1,1760	492,05
binance_ETH	1,1777	492,74
bitstamp_ETH	1,1783	492,99
bitfinex_ETH	1,1769	492,44
bittrex_ETH	1,1744	491,39
kraken_ETH	1,1845	495,59

Tabela 4.2: Tabela vozlišč, ki vsebujejo ciljno kriptovaluto, njihovo število in vrednost.

4.3.1 Negativno utežene povezave

Do negativno uteženih povezav pride zaradi različnih menjalnih tečajev pri parih kriptovalut, njihovih trenutnih vrednostih in razmika med nakupno ter prodajno ceno. Ker obstaja veliko menjalnic kriptovalut in posledično tečajev za isto kriptovaluto, hitro pride do asimetrije informacij, zaradi katere lahko pridobimo ali izgubimo vrednost.

Na sliki 4.10 si lahko ogledamo primer, kako lahko nastane takšna negativno utežena povezava. Podatki so pridobljeni iz menjalnice Binance in strani CoinMarketCap. Tečaj ETH/ARK je znašal 0,002258, medtem ko je bila vrednost v valuti USD enega ETH enaka 461,41 $ in enega ARK 0,952896 $. Strošek, ki bi nastal pri takšni menjavi kot provizija, pa znaša 0,416 $. Pri računanju vrednosti v USD smo vzeli samo podatek o ceni na menjalnici Binance in ne povprečja vseh (kot nam ga vrne klic vmesnika API). Vidimo, da se ustvari negativno utežena povezava zaradi asimetrije med menjalnim tečajem ETH/ARK in dejansko vrednostjo posamezne kriptovalute iz para. Na prvi pogled pomislimo, da bi lahko zaradi takšne asimetrije ustvarjali velike dobičke. V realnosti pa so prikazane cene samo trenutne in so veljale ob zadnji menjavi, ki se je zgodila. Če bi torej hoteli takšno menjavo izvršiti z velikimi količinami valute ETH, bi se tečaj najverjetneje spreminjal tekom izvršitve menjave. Razlog spreminjanja bi bil povečanje povpraševanja z naše strani, zaradi česar bi rasla nakupna cena valuti ARK. Seveda bi se lahko zgodil tudi dogodek, da bi nam uspelo celotno menjavo velike količine valute ETH izpeljati po planirani ceni, vendar je to manj verjetno.



Slika 4.10: Primer negativno utežene povezave.

Vozlišče	Količina	Vrednost
ETH	1	416,41 $
ARK	422,87	421,17 $

Slika 4.11: Podatki o vozliščih.

Ker v algoritmu vrednosti računamo s podatki s strani CoinMarketCap, kjer je podano povprečje vseh menjalnic in parov, lahko pride tudi do od-

stopanj. Tako se lahko zgodi, da je cena, ki jo pridobimo s spletne strani CoinMarketCap, nižja ali višja kot dejanska cena na trenutni menjalnici. Občasno pride tudi do različnih cen istih kriptovalut na različnih menjalnicah, ki jih lahko izkoristimo tako, da na eni menjalnici prodamo in na drugi kupimo isto kriptovaluto. Temu postopku pravimo arbitraža, obstaja pa veliko računalniških programov, ki te priložnosti iščejo in jih poizkusijo izkoristiti v svoj prid.

Poleg razlik v tečajih med menjalnicami najdemo tudi razlike v tečajih glede na to, s katero kriptovaluto je izbrana valuta v menjalnem paru. Na sliki 4.12 vidimo primer dvajsetih menjalnih parov s kriptovaluto BNB. Trije pari so sicer enaki z razliko menjalnic, ostali pa so si različni med seboj. Pri parih iz menjalnice Binance lahko izračunamo razliko med najvišjo in najnižno ceno kriptovalute BNB. Najvišjo ceno dosega pri menjalnemu paru s kriptovaluto ONT (ONT/BNB), najnižjo pa pri paru s kriptovaluto BTC (BNB/BTC). Razlika med najvišjo in najnižjo ceno je tako 0.39 $, kar znaša malo manj kot 3 % vrednosti kriptovalute BNB. Če to primerjamo s ceno menjave na menjalnici Binance, ki znaša 0,1 % opazimo, da je razlika velika.

V naslednjem podpoglavju 4.3.2 bomo na podobnem primeru videli, kakšen vpliv ima lahko ta razlika v tečajih. Zaradi nje se v nekaterih primerih namreč bolj splača izvesti več menjav kot pa samo eno enostavno menjavo.

4.3.2 Primer optimalne poti

V nekaterih primerih ugotovimo, da boljša pot za nakup določene kriptovalute vsebuje več menjav, kot bi jih bilo minimalno potrebno. Na primeru na sliki 4.13 je razvidno ravno to. V tabelah 4.3 in 4.4 smo prikazali podatke o atributih vozlišč, vrednosti posameznih kriptovalut in tečaje menjalnih parov, ki smo jih uporabili. Vsi podatki so pridobljeni iz menjalnice Binance in CoinMarketCap (za prikaz USD vrednosti kriptovalut). Stroški pri menjavi so 0,1 %, kar v našem primeru znaša ravno okrog 8 $ na menjavo.

V tabeli 4.4 opazimo, da vozlišče ICX*, do katerega smo prišli preko več menjav, vsebuje večjo količino kriptovalute, kot vozlišče ICX. Ti dve

Binance Coin Markets

#	Source	Pair	Volume (24h)	Price	Volume (%)
1	Binance	BNB/BTC	$35,919,942	$13.48	42.43%
2	Binance	BNB/USDT	$31,355,797	$13.49	37.04%
3	FCoin	BNB/USDT	** $5,046,330	$13.45	5.96%
4	Binance	BNB/ETH	$4,369,542	$13.54	5.16%
5	Binance	ADA/BNB	$1,172,785	$13.61	1.39%
6	Binance	EOS/BNB	$1,063,785	$13.56	1.26%
7	LBank	BNB/USDT	$733,618	$13.41	0.87%
8	Binance	XLM/BNB	$520,021	$13.52	0.61%
9	Binance	ETC/BNB	$379,495	$13.51	0.45%
10	Binance	NEO/BNB	$323,140	$13.57	0.38%
11	Binance	ICX/BNB	$301,163	$13.61	0.36%
12	Binance	LTC/BNB	$287,498	$13.62	0.34%
13	Binance	XRP/BNB	$277,842	$13.59	0.33%
14	Binance	POLY/BNB	$232,205	$13.62	0.27%
15	Binance	BCC/BNB	$227,986	$13.52	0.27%
16	Binance	ONT/BNB	$188,577	$13.87	0.22%
17	Binance	TUSD/BNB	$162,826	$13.51	0.19%
18	Binance	SKY/BNB	$128,154	$13.66	0.15%
19	Binance	MCO/BNB	$111,773	$13.61	0.13%
20	Binance	LOOM/BNB	$107,165	$13.55	0.13%

Slika 4.12: Pari s kriptovaluto BNB. Vir: CoinMarketCap, 1.8.2018.

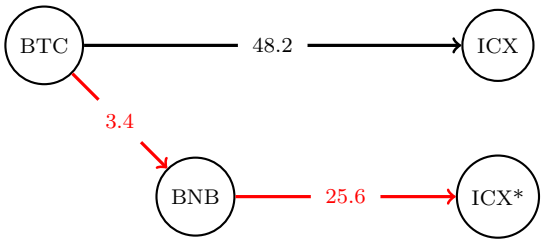
vozlišči vsebujeta enak tip kriptovalute, razlikujeta se le v poti, ki je pripeljala do njih. Tako opazimo, da preko direktnega nakupa kriptovalute ICX s kriptovaluto BTC dobimo manjšo količino in vrednost (kljub samo enkratnem strošku menjave) kot pa ob drugi poti. Pri drugi poti na začetku kupimo kriptovaluto BNB z BTC, nato pa kriptovaluto ICX z BNB. V podanem primeru, kjer bi porabili 1 BTC v vrednosti 8.008 $, bi tako z izbiro optimalne poti pridobili približno 19 $.

Valuta/par	Cena/tečaj
BTC	8.008,65 $
BNB	13,31 $
ICX	1,11 $
BTC/ICX	0,0001393
BTC/BNB	0,001661
BNB/ICX	0,08358

Tabela 4.3: Podatki o cenah in tečajih valut.

Vozlišče	Količina	Vrednost
BTC	1	8.008,65 $
ICX	7.171,54	7.960,41 $
ICX*	7.188,84	7.979,62 $
BNB	601,45	8.005,25 $

Tabela 4.4: Podatki o vozliščih. Vozlišče ICX* vsebuje isto valuto kot vozlišče ICX, le da smo do njega prišli po drugi poti.



Slika 4.13: Primer optimalne poti je označen z rdečimi povezavami. Na povezavah so prikazane uteži povezav. Tokrat je vsebovala več menjav kot jih je minimalno potrebno. Vir podatkov: Binance, CoinMarketCap, dne 31.7.2018.

4.4 Prenos na spletno okolje

Prenos na spletno okolje je še zadnji korak pred dokončanjem celotne arhitekture sistema na sliki 3.3. Ustvarili bomo čelni in zaledni del, medtem ko imamo algoritme, ki komunicirajo z menjalnicami in ostalimi spletnimi stranmi že implementirane. Ob zagonu strežnika jih inicializiramo, saj se po tem izvajajo samostojno.

Ustvarili bomo čelni in zaledni del, medtem ko imamo algoritme, ki komunicirajo z menjalnicami in ostalimi spletnimi stranmi že implementirane. Ob zagonu strežnika jih inicializiramo, saj se po tem izvajajo samostojno.

Pri prenosu na spletno okolje smo uporabili ogrodji Flask in Vue. Zaledni del aplikacije smo napisali v okolju Flask, ker uporablja enak programski jezik, kot smo ga uporabili pri gradnji aplikacije – Python. Klice API dostopnih točk, filtriranje in zapis podatkov smo nastavili na avtomatsko izvajanje, ki se ponavlja v intervalih. Klici in zapis podatkov o tečajih se izvedejo vsakih 5 minut, medtem ko se klici in zapis podatkov o stroških pošiljanja in menjav izvedejo vsakih 24 ur. Ob inicializaciji zalednega dela aplikacije, poženemo tudi Python programe, ki se povezujejo do menjalnic v časovnih intervalih. Ti programi se izvajajo v ozadju, v okviru strežniškega programa, kjer se funkcija po pretečenem intervalu ponovno pokliče in izvede.

Za čelni del smo uporabili okolje Vue, saj je primerno za gradnje tako imenovanih spletnih aplikacij na eni strani (SPA). Ogrodje bi bilo še posebej uporabno, če bi na sami strani imeli še več oken, ki bi se med seboj neodvisno posodabljala. Naš čelni del aplikacije bo razdeljen na dva dela: vnos podatkov in prikaz rezultatov ter grafični prikaz optimalne poti. Za izdelavo GET zahtevka na sliki 4.14, ki pošlje podatke na zaledni del aplikacije, smo uporabili knjižnico Axios [53]. Podatki, ki so podani kot parametri, so bili pridobljeni iz vnosnih oken, kamor jih je vnesel uporabnik. Stran, na katero pošljemo zahtevek, je v našem primeru localhost, ker razvijamo na lastnem računalniku.

Celotna aplikacija se izvaja kot REST storitev, kjer čelni del kliče zaledni del preko GET zahtevka. Funkcija zalednega dela aplikacije nato sprejme

```
axios({
    method: 'get',
    url: `http://localhost:5000/shortestPath`,
    params: {
      currency: this.currency,
      investedAmount: this.investedAmount,
      targetCryptocurrency:
          this.targetCryptocurrency
    },
    headers: {'Access-Control-Allow-Origin': '*'}
  })
```

Slika 4.14: Izdelava GET zahtevka s pomočjo knjižnice Axios.

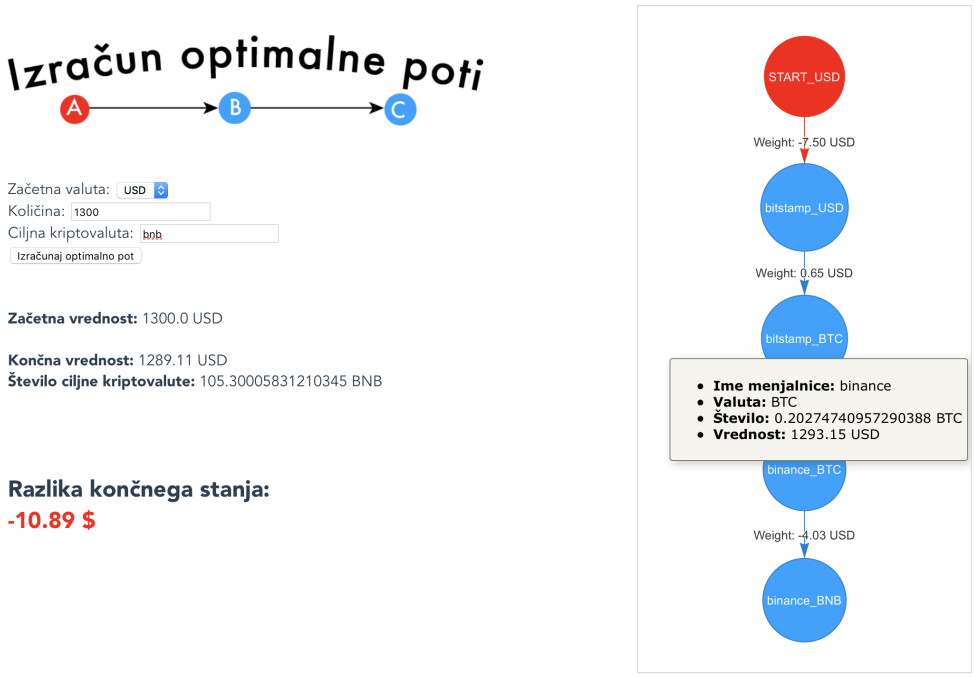
podatke, na podlagi njih izdela oziroma poišče optimalno pot in rezultate vrne kot odziv čelnemu delu. Na podlagi podatkov iz odziva čelni del grafično prikaže optimalno pot in ostale rezultate.

4.5 Prikaz najugodnejše poti (spletna aplikacija)

Za grafični prikaz optimalne poti na čelnem delu smo izbrali JavaScript knjižnico Vis [54]. S pomočjo nje lahko vizualiziramo optimalno pot od začetka do ciljne kriptovalute.

Pri gradnji omrežja in računanju optimalne poti smo povezave utežili na takšen način, da je pozitivna utež povezave pomenila izgubo valute, negativno utežena povezava pa je pomenila, da smo valuto pridobili. Pri grafičnem prikazu bi se takšen sistem lahko izkazal kot nerazumljiv za končne uporabnike. Prikaz uteži povezav smo zato spremenili na način, da negativno utežena povezava pomeni izgubo valute in pozitivna pridobitev. Vozlišče 'GOAL' smo pri prikazu izpustili, saj nam je služilo samo pri izdelavi sku-

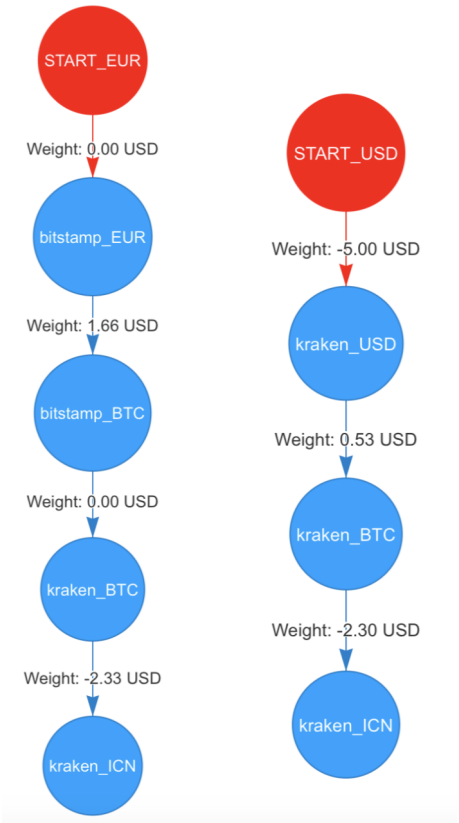
pnega končnega vozlišča v omrežju na sliki 4.15. Podatke o posameznem vozlišču smo prikazali, ko se s kurzorjem postavimo na izbrano vozlišče 4.15. S tem lahko vidimo ime menjalnice, valuto, število valute in vrednost tega vozlišča v USD.



Slika 4.15: Spletna aplikacija. Desni del prikazuje graf optimalne poti, levi del vsebuje vnos podatkov in rezultate..

Na sliki 4.16 vidimo grafični prikaz najugodnejše poti pri nakupu kriptovalute ICN z valutama EUR in USD. Pri obeh izračunih smo kot končno kriptovaluto nastavili valuto ICN, kot začetno vrednost pa valuti USD oziroma EUR v vrednosti 577 USD (v tistem času je bilo to približno enako 500 EUR). Povezave vsebujejo uteži, tako vidimo, da je nakazilo valute EUR na menjalnico Bitstamp zastonj, medtem ko je nakazilo valute USD na menjalnico Kraken uteženo s ceno 5 USD. Glede na razlike med potmi, ki sta nastali ob različnih začetnih valutah lahko sklepamo, da je menjalni tečaj EUR/BTC

na menjalnici Bitstamp trenutno ugodnejši kot menjalni tečaj USD/BTC na menjalnici Kraken, saj smo na prvi pridobili večjo utež povezave.



Slika 4.16: Grafični prikaz poti v čelnem delu aplikacije. Začetna valuta na levi je EUR na desni USD. Začetni vrednosti v USD sta enaki.

Še eden primer različne optimalne poti, ki nastane zaradi različnih začetnih vrednosti, je pri manjših nakupih, kjer pridejo fiksni stroški bolj do izraza, medtem ko pri večjih nakupih pot narekujejo boljši menjalni tečaji.

Poglavje 5

Evaluacija predlaganega pristopa

V sledečem poglavju bomo izdelano rešitev primerjali s sorodnimi. Za primerjavo bomo uporabili primer uporabe, ko uporabnik želi s trenutno fiat valuto USD kupiti izbrano kriptovaluto. Pri tem ne bomo upoštevali stroškov posameznih storitev, saj nas zanima zgolj, kako ugodno pot najdejo njihovi algoritmi. Evaluirali bomo našo rešitev, ki nam je podala optimalno pot, opisali probleme, ki bi lahko nastali ob praktični uporabi in omejitve, ki nastopajo.

5.1 Primerjava s sorodnimi rešitvami

Sorodne rešitve, opisane v poglavju 2.5, bomo primerjali med seboj in z izdelano rešitvijo. Cilj primerjave je ugotoviti, kako se razlikuje število prejete končne kriptovalute ob fiksnem znesku fiat valute na vhodu. V primerjavi med vsemi tremi rešitvami bomo vzeli kriptovalute, ki jih ponujajo vse rešitve. Zaradi omejitev širine tabele bomo storitve zapisali na krajši način: Changelly (CH), Coinmama (CM), naša rešitev (R). Da bo primerjava bolj objektivna, bomo pri dveh sorodnih rešitvah odstranili dodatne provizije platform. Tako moramo vrednosti, dobljene pri storitvi Changelly,

povečati za 0,5025 % (provizija 0,5 %) in vrednosti pri storitvi Coinmama povečati za 6,2699 % (provizija 5,9 %). Pri obeh torej ne upoštevamo provizije ob izvršitvi menjave, prav tako pa ne upoštevamo provizije ob nakazilu denarja do aplikacije. Da pridemo do enakih kriterijev tudi pri naši aplikaciji, moramo pri izračunu začetni vrednosti dodati strošek pošiljanja fiat valute na menjalnico preko bančnega nakazila (v primerih v tabeli 5.1 je bila tako prišteta vrednost večinoma 5 $ in 7,5 $). Stolpci v tabeli 5.1 vsebujejo začetno valuto, ciljno kriptovaluto, vrednost začetne investicije in število ciljne kriptovalute, ki jo dobimo preko posamezne storitve.

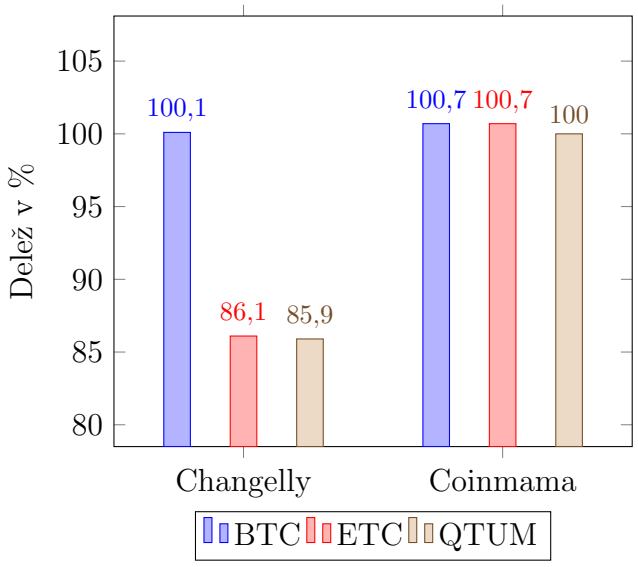
Začetek	Konec	Začetna vrednost	CH	CM	R
USD	BTC	1.000 $	0,15301	0,1541	0,1536
USD	ETH	1.000 $	2,7035	2,7170	2,7030
USD	ETC	1.000 $	52,0225	60,7504	60,8957
USD	QTUM	1.000 $	150,5217	176,2279	176,5474
USD	ADA	1.000 $	/	8.558,9777	8.538,4076
USD	BNB	1.000 $	69,0321	/	80,4331

Tabela 5.1: Primerjava dobljenega števila kriptovalute glede na izbrano storitev. Podatki pridobljeni dne 8.8.2018

Opazimo, da so vrednosti pri nekaterih kriptovalutah zelo podobne, pri drugih pa različne od storitve do storitve. Pri nakupu kriptovalute QTUM prejmemo z uporabo storitve Changelly približno 26 kovancev manj, kot pri uporabi naše razvite rešitve ali storitve Coinmama (provizije niso vštete). Razlogi v takšnih odstopanjih so v volatilnosti trga, kjer lahko v določenih trenutkih razliko naredijo že sekunde. V tem primeru je cena kriptovalute QTUM padala in verjetno algoritem storitve Changelly še ni uspel spremeniti vrednosti na nove. Pri nakupu kriptovalute ADA bi preko storitve Coinmama prejeli večje število, kot z uporabo naše rešitve. Razlog je lahko ta, da ima storitev Coinmama na menjalnicah odprte račune, ki pogosto trgujejo, zaradi česar ima nižje provizije ob menjavah. Cene v tabeli 5.1 so prikazane brez

provizij, ki bi jih zaračunale aplikacije.

Naslednjo primerjavo v tabeli 5.2 smo izvedli v 10-minutnih časovnih oknih, kjer smo primerjali količine prejete kriptovalute glede na izbrano rešitev. Prvo primerjavo smo tako izvedli s preizkusom ob 12:06 uri, naslednjo ob 12:16 zadnjo pa ob 12:46 uri. Pri vseh primerih je bila začetna valuta USD v količini 1.000. Tudi tukaj smo odšteli provizije, ki jih aplikacije zaračunavajo, oziroma nakazila začetne fiat valute pri naši rešitvi. Prejete količine s strani rešitev Coinmama in Changelly smo delili s količino končne kriptovalute pri naši aplikaciji. S tem smo dobili delež, ki je lahko višji ali nižji od potencialne količine, prejete s strani naše aplikacije. Opazimo, da so količine zelo podobne, še posebej med našo rešitvijo in spletno aplikacijo Coinmama. Pri primerjavi s storitvijo Changelly občasno opazimo razlike tudi do slabih 15 % v našo korist. Na grafu na sliki 5.1 smo prikazali povprečje petih poizkusov pri obeh rešitvah za omenjene kriptovalute.



Slika 5.1: Primerjava povprečne pridobljene vrednosti sorodnih rešitev z našo.

Zaradi volatilnosti celotnega trga kriptovalut so potrebne višje provizije, kot bi jih morda želeli. Ker te storitve ponujajo svoj tečaj za nekaj deset sekund ali par minut, lahko v tem času cena kriptovalute zraste ali pade.

Poizkus	Končna kriptovaluta	Changelly	Coinmama
1	BTC	100,02 %	100,82 %
2	BTC	100,16 %	100,67 %
3	BTC	100,06 %	100,74 %
4	BTC	100,04 %	100,81 %
5	BTC	100,40 %	100,30 %
1	ETC	86,25 %	100,95 %
2	ETC	86,35 %	101,08 %
3	ETC	86,29 %	100,87 %
4	ETC	86,20 %	100,50 %
5	ETC	85,53 %	100,22 %
1	QTUM	86,03 %	100,10 %
2	QTUM	86,01 %	100,24 %
3	QTUM	85,98 %	100,14 %
4	QTUM	85,93 %	100,10 %
5	QTUM	85,30 %	99,44 %

Tabela 5.2: Deleži količine kriptovalut, normirane z rezultatom izdelane rešitve. Podatki pridobljeni 16.8.2018, 12:06 - 12:46

Zaradi tega je pomemben volumen takih storitev, saj s tem uravnovesijo primere, kjer so pri menjavi dejansko izgubili zaračunano provizijo, in primere, kjer so zaradi rasti cene prejeli še večjo končno količino, kot obljubljeno.

Prav tako se pojavijo dodatni stroški in tveganja, povezana s financiranjem preko kreditnih kartic. V izdelavi naše rešitve smo za fiat nakazila upoštevali transakcijo SEPA. V primeru, da bi upoštevali nakazilo preko kreditne kartice se tudi pri naši rešitvi pojavi strošek v višini 5 % nakazane vrednosti (menjalnica Bitstamp), nekatere menjalnice pa nakazila preko kreditnih kartic niti ne omogočajo.

5.2 Optimalne poti

Namenjena uporaba izdelane rešitve je ugotovitev optimalne poti pri nakupu ciljne kriptovalute. Tako je naš namen ugotoviti pot po omrežju, kjer z začetno investicijo dobimo čim večjo količino končne kriptovalute. Izdelali smo algoritem, ki je pridobil podatke s seznama menjalnic in jih shranil v podatkovno bazo. S temi podatki smo ustvarili vozlišča, jih povezali in s tem ustvarili omrežje z uteženimi povezavami. Po njem smo poiskali najkrajšo pot med začetnim in končnim vozliščem, izpisali relevantne podatke za uporabnika ter grafično prikazali potek poti. Uporabnik rešitve bi lahko potem sam izvedel menjave in pošiljanja glede na prikazano pot.

Pri izdelavi omrežja in računanju optimalne poti smo za tečaje kriptovalutnih parov vzeli tečaj zadnje izvedene menjave. Tako je tečaj sicer ustrezen, vendar bi se lahko ob nakupih večjih količin spremenil, saj bi bilo naše povpraševanje po istem tečaju večje od trenutne ponudbe. Seveda je zgoraj omenjena sprememba odvisna od trenutnega trenda. Tako bi v primeru trenda naraščajoče cene morali dvigovati ceno, da bi kupili želeno količino, v primeru padajočega trenda pa bi verjetno dobili celotno število po tečaju zadnje menjave.

Optimalna pot je tako najbolj ustrezna samo v trenutku, ko je narejena. V primeru, da bi hoteli točne podatke, bi bilo potrebno API ali drug pristop

povezave vzdrževati konstantno, hkrati pa pridobiti tudi podatke o knjigah naročil. V knjigi naročil so zapisana vsa trenutno aktivna naročila, s podatki o želenem tečaju in količini. Preko nje bi tako lahko videli, ali bomo lahko celotno menjavo izvedli po trenutnem tečaju ali bomo morali tečaj menjave povečati.

Poglavje 6

Sklepne ugotovitve

Ustvarili smo algoritem, ki na podlagi podatkov, pridobljenih iz menjalnic kriptovalut in z ostalih strani ustvari omrežje menjalnih parov, v katerem lahko poiščemo optimalno pot. Celotna koda spletne aplikacije je dostopna na GitHub repozitoriju [55], delujoča rešitev pa na spletnem naslovu `https://optimalnapot.herokuapp.com`. Namenjena je iskanju najkrajše poti, ko želimo izbrano kriptovaluto kupiti s fiat valuto EUR ali z USD. Kot rezultat prejmemo prikaz te poti, ki jo lahko samostojno izvršimo. Pri pregledu sorodnih rešitev smo videli dve storitvi, ki v ozadju verjetno izvajata podobne algoritme za iskanje najcenejše poti. Storitvi omogočata nakup oziroma menjavo kriptovalute, kjer se vsa logika zgodi v ozadju, uporabnik pa samo prejme končno količino ciljne kriptovalute.

V članku [36] opisan problem bi lahko bil v naši rešitvi izveden kot iskanje negativnih ciklov v omrežju. Ko najdemo negativen cikel, lahko izvedemo opisane menjave in pridobimo na vrednosti. Takšna storitev izkorišča asimetrije informacij in volatilnosti trga kriptovalut. Kot smo omenili v poglavju 4.3.1 in prikazali na primerih, obstajajo negativne povezave, kjer z menjavami pridobimo na vrednosti. Dejanska storitev bi bila težje izvedljiva, saj se takšni negativni cikli, s katerimi bi lahko pridobili na vrednosti, hitro izravnajo že ob manjših količinah menjav. Kljub temu pa bi lahko z avtomatiziranim in s hitrim pristopom uspešno izvajali tako imenovano arbitražo

med menjalnicami. [56].

Našo rešitev iskanja najugodnejše poti med začetno fiat valuto in končno kriptovaluto bi lahko nadgradili na več različnih načinov. Prvi je enak kot sorodne rešitve. Tako bi kot podjetje ustvarili račune na raznih menjalnicah kriptovalut in glede na povpraševanje uporabnikov aplikacije iskali optimalne poti, izvrševali menjave ter dostavljali ciljne kriptovalute. Prednost tega pristopa je predvsem prijaznost do uporabnika, saj mu ni potrebno ustvarjati računov pri različnih menjalnicah kriptovalut. Zaradi volatilnosti pa bi se morali poslužiti provizij na podoben način kot storitvi, opisani v poglavju 2.5. Drugi način nadgradnje bi bila aplikacija, kjer bi menjave kriptovalut izvajali na uporabniških računih uporabnikov naše aplikacije. Uporabnik bi ob podatkih, potrebnih za računanje optimalne poti, vnesel še lastne dostopne ključe API iz menjalnic, kjer je registriran. Algoritem bi na podlagi menjalnic, ki jih je uporabnik vnesel, iskal poti le po teh menjalnicah, našel optimalno in izvršil vse potrebne menjave. Storitev bi s tem avtomatsko izvršila prikazano pot na uporabniških računih uporabnika. Prednost takšnega načina nadgradnje je v transparentnosti aplikacije, saj so vidni vsi koraki. Prav tako nam ne bi bilo obvezno zaračunavati provizij, saj se menjave ne dogajajo na naših uporabniških računih.

Idejo iskanja optimalne poti ob nakupu bi lahko uporabili tudi pri drugih področjih. Primer takšne uporabe je nakup delnic, ki kotirajo na različnih borzah. V tem primeru bi imeli začetno valuto in več možnih vstopov na borze, ki poslujejo z raznimi valutami. Iskali bi optimalno pot, kjer bi za naš vložek dobili čim večje število želenih delnic.

Literatura

[1] CoinMarketCap OpCo LLC., "Cryptocurrency market capitalizations — coinmarketcap." Dosegljivo: https://coinmarketcap.com. [Dostopano: 19. 7. 2018].

[2] Investopedia LLC., "Fiat money." Dosegljivo: https://www.investopedia.com/terms/f/fiatmoney.asp. [Dostopano: 22. 8. 2018].

[3] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system." Dosegljivo: https://nakamotoinstitute.org/bitcoin/. [Dostopano: 28. 8. 2018].

[4] Brilliant British Ltd, "The rise fall (and rise fall) of the top 10 cryptocurrencies from 2014-2018." Dosegljivo: https://merchantmachine.co.uk/cryptocurrencies/. [Dostopano: 22. 8. 2018].

[5] Ethereum Foundation (Stiftung Ethereum), "Etherium project." Dosegljivo: https://www.ethereum.org. [Dostopano: 7. 8. 2018].

[6] N. Gandal, J. Hamrick, T. Moore, in T. Oberman, "Price manipulation in the bitcoin ecosystem," *Journal of Monetary Economics*, št. 95, str. 86–96, 2018.

[7] M. Galka, "Four years of token sales, visualized in one graphic." Dosegljivo: https://elementus.io/blog/token-sales-visualization/. [Dostopano: 8. 8. 2018].

[8] "Erc20 token standard - the etherium wiki." Dosegljivo: https://theethereum.wiki/w/index.php/ERC20_Token_Standard. [Dostopano: 8. 8. 2018].

[9] Z. Zheng, S. Xie, H.-N. Dai, in H. Wang, "Blockchain challenges and opportunities: A survey," *Work Pap.*, 2016.

[10] Z. Zheng, S. Xie, H. Dai, X. Chen, in H. Wang, "An overview of blockchain technology: Architecture, consensus, and future trends," v zborniku *Big Data (BigData Congress), IEEE International Congress on*, Honolulu, HI, USA, str. 557–564, IEEE, junij 2017.

[11] F. Tschorsch in B. Scheuermann, "Bitcoin and beyond: A technical survey on decentralized digital currencies," *IEEE Communications Surveys & Tutorials*, št. 18, zv. 3, str. 2084–2123, 2016.

[12] Bitcoin Project, "Kako bitcoin deluje?." Dosegljivo: https://bitcoin.org/sl/kako-deluje. [Dostopano: 8. 8. 2018].

[13] S. Turšič, "Trgovalni sistem nad digitalno valuto bitcoin," diplomska naloga, Fakulteta za računalništvo in informatiko, Univerza v Ljubljani, 2014.

[14] M. D'Aliessi, "How does the blockchain work." Dosegljivo: https://medium.com/@micheledaliessi/how-does-the-blockchain-work-98c8cd01d2ae. [Dostopano: 9. 7. 2018].

[15] CodeTract, "Bat ico, usd 35 million in 24 seconds, gas and gasprice." Dosegljivo: https://medium.com/@codetractio/bat-ico-usd-35-million-in-24-seconds-gas-and-gasprice-6cdde370a615. [Dostopano: 8. 8. 2018].

[16] Bitstamp Ltd., "About us - bitstamp." Dosegljivo: https://www.bitstamp.net/about_us/. [Dostopano: 19. 7. 2018].

[17] Payward Inc., "Kraken bitcoin exchange." Dosegljivo: https://www.kraken.com. [Dostopano: 19. 7. 2018].

[18] iFinex Inc., "Bitfinex - bitcoin, litecoin and ethereum exchange and margin trading platform." Dosegljivo: https://www.bitfinex.com. [Dostopano: 26. 7. 2018].

[19] Phoenixfin Pte. Ltd., "Kucoin exchange." Dosegljivo: https://www.kucoin.com. [Dostopano: 19. 7. 2018].

[20] Poloniex LLC., "Poloniex - bitcoin/digital asset exchange." Dosegljivo: https://poloniex.com. [Dostopano: 19. 7. 2018].

[21] Bittrex Inc., "Bittrex cryptocurrency exchange." Dosegljivo: https://bittrex.com. [Dostopano: 23. 7. 2018].

[22] Binance, "Binance - blockchain and crypto asset exchange." Dosegljivo: https://www.binance.com. [Dostopano: 19. 7. 2018].

[23] Z. Whittaker, "Bitstamp exchange hacked, $5m worth of bitcoin stolen." Dosegljivo: https://www.zdnet.com/article/bitstamp-bitcoin-exchange-suspended-amid-hack-concerns-heres-what-we-know/. [Dostopano: 19. 7. 2018].

[24] S. Ember, "Bitcoin exchange bitstamp resumes services." Dosegljivo: https://dealbook.nytimes.com/2015/01/09/bitcoin-exchange-bitstamp-resumes-services/. [Dostopano: 19. 7. 2018].

[25] P. Ambler, "From zero to crypto billionaire in under a year: Meet the founder of binance." Dosegljivo: https://www.forbes.com/sites/pamelaambler/2018/02/07/changpeng-zhao-binance-exchange-crypto-cryptocurrency/#390a79b81eee. [Dostopano: 20. 7. 2018].

[26] Doris, "Trading fee discount function – kucoin help center." Dosegljivo: https://kucoin.zendesk.com/hc/en-us/articles/360000512294-Trading-Fee-Discount-function. [Dostopano: 19. 7. 2018].

[27] Bloomberg L.P., "Poloniex, llc: Private company information - bloomberg." Dosegljivo: https://www.bloomberg.com/research/stocks/private/snapshot.asp?privcapId=309559982. [Dostopano: 23. 8. 2018].

[28] J. Verhage, "Goldman-backed circle agrees to buy crypto exchange poloniex." Dosegljivo: https://www.bloomberg.com/news/articles/2018-02-26/goldman-backed-circle-buys-digital-exchange-poloniex. [Dostopano: 23. 7. 2018].

[29] S. Higgins, "Bitfinex examined: Inside the troubled bitcoin exchange's history." Dosegljivo: https://www.coindesk.com/bitfinex-examined-bitcoin-exchange/. [Dostopano: 22. 8. 2018].

[30] Tether Limited, "Tether." Dosegljivo: https://tether.to. [Dostopano: 23. 8. 2018].

[31] W. Suberg, "Research: Tether, bitfinex 'manipulation' reason behind 2017 bitcoin price highs." Dosegljivo: https://cointelegraph.com/news/research-tether-bitfinex-manipulation-reason-behind-2017-bitcoin-price-highs. [Dostopano: 6. 8. 2018].

[32] J. M. Griffin in A. Shams, "Is bitcoin really un-tethered?." Dosegljivo na SSRN: https://ssrn.com/abstract=3195066, 13 junij, 2018.

[33] Bittrex Support Team, "Bittrex about us." Dosegljivo: https://support.bittrex.com/hc/en-us/articles/115003684411. [Dostopano: 23. 7. 2018].

[34] E. W. Dijkstra, "A note on two problems in connexion with graphs," *Numerische mathematik*, št. 1, zv. 1, str. 269–271, 1959.

[35] R. Bellman, "On a routing problem," *Quarterly of applied mathematics*, št. 16, zv. 1, str. 87–90, 1958.

[36] F. Bortolussi, Z. Hoogeboom, in F. W. Takes, "Computing minimum weight cycles to leverage mispricings in cryptocurrency market networks," *arXiv preprint arXiv:1807.05715*, 2018.

[37] Fintechvision Limited, "Changelly." Dosegljivo: `https://changelly.com`. [Dostopano: 8. 8. 2018].

[38] NBV International s.r.o., "Coinmama." Dosegljivo: `https://www.coinmama.com`. [Dostopano: 8. 8. 2018].

[39] Python Software Foundation, "History and license – 3.7.0 documentation." Dosegljivo: `https://docs.python.org/3/license.html`. [Dostopano: 23. 8. 2018].

[40] A. Ronacher, "Flask (a python microframework)." Dosegljivo: `http://flask.pocoo.org`. [Dostopano: 20. 7. 2018].

[41] L. Richardson, "Beautiful soup documentation." Dosegljivo: `https://www.crummy.com/software/BeautifulSoup/bs4/doc/`. [Dostopano: 7. 7. 2018].

[42] NetworkX Developers, "Networkx tutorial, creating a graph." Dosegljivo: `https://networkx.github.io/documentation/networkx-1.10/tutorial/tutorial.html`. [Dostopano: 7. 7. 2018].

[43] R. McFarland, "Asynchronous python http requests for humans." Dosegljivo: `https://github.com/ross/requests-futures`. [Dostopano: 23. 7. 2018].

[44] S. Edlich, "Nosql databases." Dosegljivo: `http://nosql-database.org`. [Dostopano: 23. 8. 2018].

[45] B. M. Sasaki, "Graph databases for beginners: Acid vs. base explained." Dosegljivo: `https://neo4j.com/blog/acid-vs-base-consistency-models-explained/`. [Dostopano: 7. 8. 2018].

[46] "Bson (binary json) serialization." Dosegljivo: http://bsonspec.org. [Dostopano: 23. 8. 2018].

[47] MongoDB Inc., "Pymongo 3.7.1 documentation." Dosegljivo: http://api.mongodb.com/python/current/tutorial.html. [Dostopano: 7. 8. 2018].

[48] K. Das, "Introduction to flask." Dosegljivo: https://pymbook.readthedocs.io/en/latest/flask.html. [Dostopano: 9. 8. 2018].

[49] "Vue.js introduction." Dosegljivo: https://vuejs.org/v2/guide/. [Dostopano: 9. 8. 2018].

[50] N. Pisk, "Realno-časovna vizualizacija gibanja tečajev kriptovalut," diplomska naloga, 2018.

[51] M. Vergel, "Currency converter api." Dosegljivo: https://free.currencyconverterapi.com. [Dostopano: 24. 7. 2018].

[52] A. Hagberg, P. Swart, in D. Schult, "Exploring network structure, dynamics, and function using networkx," , Los Alamos National Lab.(LANL), Los Alamos, NM (United States), 1 januar, 2008.

[53] M. Zabriskie in N. Uraltsev, "Axios." Dosegljivo: https://www.npmjs.com/package/axios. [Dostopano: 31. 7. 2018].

[54] Almende B.V., "vis.js - a dynamic, browser based visualization library.." Dosegljivo: http://visjs.org. [Dostopano: 31. 7. 2018].

[55] A. Povše, "Github repozitorij rešitve." Dosegljivo: https://github.com/andrazpovse/cryptoShortestPath. [Dostopano: 15. 8. 2018].

[56] J. Kokeš, "Control strategy to trade cryptocurrencies," *International Journal of Business and Management*, št. 5, zv. 1, str. 62–69, 2017.