

TALLINNA TEHNIKAÜLIKOOL
Infotehnoloogia teaduskond

Rasmus Sild 134912IAPB

VÕRGUSTIKE ANALÜÜS JA VISUALISEERIMINE

bakalaureusetöö

Juhendaja: Tanel Tammet
doktorikraad

Tallinn 2018

Autorideklaratsioon

Kinnitan, et olen koostanud antud lõputöö iseseisvalt ning seda ei ole kellegi teise poolt varem kaitsmisele esitatud. Kõik töö koostamisel kasutatud teiste autorite tööd, olulised seisukohad, kirjandusallikatest ja mujalt pärinevad andmed on töös viidatud.

Autor: Rasmus Sild

22.05.18

Annotatsioon

Lõputöö eesmärgiks on lihtsustada võrgustike analüüsi protsessi ja teha see protsess kättesaadavamaks tavakasutajale. Selleks uuriti esmalt saadavalolevaid võrgustike analüüsimise tarkvarasid. Analüüsi tarkvarade kasutusmugavust ja funktsionaalsust. Analüüsist lähtuvalt loodi veebirakendus, mis peab olema võimalikult kasutajasõbralik ja sisaldama funktsionaalsust, mida võrgustike analüüsis kõige enam vaja läheb. Kasutaja saab laadida üles oma võrgustiku andmed ning need visualiseeritakse talle meeldival kujul.

Töö erinevates peatükkides annab autor edasi tarkvarade analüüsi tulemused, seletab lahti arendusprotsessi ja valminud rakenduse osad, ning toob välja kasutatud tehnoloogiad.

Töö tulemusena loodi veebirakendus, millega kasutajad saavad analüüsida oma võrgustiku andmeid lihtsalt ja kiirelt. Veebirakendus asub aadressil <https://rasmussild.github.io/data-viz/>, rakenduse lähtekood on kättesaadav aadressil <https://github.com/RasmusSild/data-viz>

Lõputöö on kirjutatud eesti keeles ning sisaldab teksti 36 leheküljel, 8 peatükki, 11 joonist.

Abstract

Analysis & visualisation of networks

The aim of this thesis is to simplify the process of analysing network, specifically graph data and make this process more accessible to regular users. An analysis was conducted to explore different downloadable network visualisation and analysis software. Points of analysis were the user experience and available functionality. A working web application was created based on the analysis, which is meant to be user-friendly and contain features that are the most important in network analysis. The user can upload their network data and it will be visualised to them in a pleasing manner.

The different chapters of this thesis contain the author's results of the software analysis, the development process and created features of the web application, and finally the technology used to create the web application.

With the resulting web application users can analyse their network data quickly and easily. The web application is located at the following web address: <https://rasmussild.github.io/data-viz/>, the application's source code can be found at <https://github.com/RasmusSild/data-viz>

The thesis is in Estonian and contains 36 pages of text, 8 chapters, 11 figures.

Lühendite ja mõistete sõnastik

CSV	<i>Comma-separated Values</i>
TSV	<i>Tab-separated Values</i>
JSON	<i>Javascript Object Notation</i>
CLI	<i>Command Line Interface</i>
HTML	<i>Hypertext Markup Language</i>
SVG	<i>Scalable Vector Graphics</i>
npm	<i>node package manager</i>
D3	<i>Data Driven Documents</i>
CTA	<i>Call to Action</i>
IDE	<i>Integrated Development Environment</i>

Sisukord

1 Sissejuhatus	9
2 Keskusmõõdud.....	11
2.1 Astmekeskus (degree centrality)	11
2.2 Läheduskeskus (closeness centrality)	12
2.3 Vahelisuskeskus (betweenness centrality)	12
2.4 Omavektorikeskus (eigenvector centrality)	13
3 Analoogilised lahendused.....	14
3.1 Gephi	14
3.2 Cytoscape.....	15
4 Veebirakenduse arendus	17
4.1 Funktsionaalsused ja vaated	17
4.2 Arenduskäik.....	18
5 Komponentid ja vaated	20
5.1 Avalehe komponent	20
5.2 Redaktori komponent	21
5.2.1 Graafi loomise viisard	21
5.2.2 Redaktor	22
5.3 Demo komponent	27
5.4 Abilehe komponent.....	28
6 Kasutatud tehnoloogiad	29
6.1 Javascript	29
6.2 Typescript.....	29
6.3 Angular	29
6.4 D3.js.....	30
6.5 Clarity	30
6.6 WebStorm.....	30
6.7 NodeJS ja npm.....	30
7 Võimalikud edasiarendused.....	32
7.1 Mobiiliversioon	32

7.2 Graafi kujundus	32
7.3 Andmetüüpide tugi	33
7.4 Keskusmõõdud.....	33
8 Kokkuvõte	34
Kasutatud kirjandus	35

Jooniste loetelu

Joonis 1. Tavaseadetega graaf Gephi-s.....	15
Joonis 2. Tavaseadetega graaf Cytoscape-s.....	16
Joonis 3. Avalehe vaade	20
Joonis 4. Viisardi esimese lehekülje vaade	21
Joonis 5. Viisardi teise lehekülje vaade.....	22
Joonis 6. Redaktori täisvaade	23
Joonis 7. Redaktori graafi osa.....	24
Joonis 8. Redaktori tööriistariba, avatud kesksuse sakk.....	25
Joonis 9. Kesksusmõtude täistabeli vaade.....	26
Joonis 10. Redaktori tööriistariba, avatud kohandamise sakk.....	27
Joonis 11. Abilehe vaade	28

1 Sissejuhatus

Võrgustikud annavad hea ülevaate isikute või objektide omavahelistest seostest. Võrgustikud koosnevad objektidest ja nendevahelistest lülidest. Sageli esitatakse võrgustikke graafidena. Graaf koosneb tippudest (objektidest) ja servadest (ühendustest objektide vahel). Kui on loodud võrgustik, mis esitab mingi kindla eluvaldkonna andmeid, saab võrgustikust välja lugeda palju huvitavat informatsiooni. Näiteks politseil võib olla olemas võrgustik, kus objektideks on telefoninumbrid ning ühendused on kõned teistele numbritele. Leides võrgustiku kõige olulisema lüli (näiteks telefoninumber, kuhu on kõige rohkem kõnesid tehtud), võib see anda informatsiooni mõne potentsiaalse kahtlusaluse kohta.

Võrgustike analüüsis on tähtsal kohal kesksusmõõdud. Need on erinevate algoritmide alusel saadud arvvaartused igale võrgustiku lülile, mis iseloomustavad lüli tähtsust võrgustikus mitmest erinevast aspektist.

Võrgustike ja graafide analüüsimiseks ja visualiseerimiseks on olemas mitmeid allalaetavaid tarkvarasi. Veebibrauseris töötavat rakendust, mis keskenduks võrgustike analüüsile ja just kesksusmõõtudele autoril ei õnnestunud internetist leida. Veebibrauseri eelis on kättesaadavus ja kasutusmugavus. Kasutaja ei pea suurt installeerimisfaili alla laadima ja installeerimisprotsessi läbi tegema, mis võtab aega. Tihti on ka allalaetavad tarkvarad tülikad kasutada ning neil on palju funktsionaalsusi, mida tavakasutajal vaja ei lähe.

Lõputöö eesmärgiks on võrgustike analüüsi protsess teha lihtsamaks, luues selle tarbeks veebirakenduse. Veebirakendus lubab kasutajal üles laadida oma võrgustikku puudutavad andmed, visualiseerib selle graafina, ning näitab kasutajale välja kesksusmõõte kasutades tema võrgustiku kõige olulisemad lülid. Rakendus luuakse inglise keelne, kuna on mõeldud ülemaailmseks kasutamiseks.

Lõputöö raames valminud veebirakendus asub aadressil
<https://rasmussild.github.io/data-viz/>, rakenduse lähtekood on kättesaadav aadressil
<https://github.com/RasmusSild/data-viz.>

2 Keskusmõõdud

Antud töös ja loodavas rakenduses on oluline osa graafi tippude keskusmõõtud. Keskusmõõdud on väga tähtsad selgitamiseks välja kõige olulisemad võrgustiku osad. Need kuvatakse loodavas rakenduses värviskaalana graafi tippude peal. Loodavasse rakendusse sai keskusmõõde valitud 4, kuigi neid on saadaval üle 10-ne. Need 4 sai valitud kahel põhjusel. Esiteks uuriti, millist tööriista kasutatakse korralikes kõige rohkem võrgustike analüüsiks. Selleks on IBM i2 Analyst's Notebook, kus ongi esitatud just need 4 mõõtu. Samuti viitab keskusmõõde kajastav Wikipedia artikkel[16] just eelkõige nendele keskusmõõtudele.

2.1 Astmekeskus (degree centrality)

Kõige esimene ja ilmselt lihtsamini mõistetav keskusmõõde. Tipu astmekeskus näitab, kui palju ühendusi (servi) konkreetsel tipul on. See mõõde aitab tuvastada kõige aktiivsemad tipud võrgustikus[1].

Astmekeskust saab mõõta 3 moodi:

- Ainult tipust väljaminevate ühenduste arv.
- Ainult tippu sissetulevate ühenduste arv.
- Kahe eelneva summa.

Loodavas veebirakenduses arvutatakse summa, kuna autori arvates annab see parema ülevaate tipu tähtsusest võrgustikus.

Astmekeskuse arvutamise üldine valem on järgmine[15]:

$$C_d = \frac{1}{N-1} d_i$$

kus d_i on tipu aste ja N suurim aste.

2.2 Läheduskesksus (closeness centrality)

Tipu läheduskesksuse mõõde mõõdab tipu lähedust teistele võrgustiku tippudele.

Tipul, millel on kõrge läheduskesksuse mõõde omab lühimaid teid teiste tippudeni. See mõõde võimaldab tipul suhtlust kiiremini vastu võtta ja edasi anda kui teised tipud võrgustikus.

Teave liigub kaugemale tipust ja tippu võrgustiku servast, mis on ühenduses vaid mõne üksiku tipuga. Neil on seetõttu väiksem läheduskesksuse mõõde[2].

Läheduskesksus on defineeritud kui[15]:

$$C(x) = \frac{1}{\sum_y d(x,y)}$$

kus $d(x,y)$ on kaugus tippude x ja y vahel.

2.3 Vahelisuskesksus (betweenness centrality)

Tipu vahelisuskesksus mõõdab iga tipu kaudu läbitavate teekondade arvu.

See mõõde tuvastab tippe, mis suudavad juhtida infovoogu võrgustiku erinevate osade vahel. Neid tippe nimetatakse väravaülemateks. Väravaülematel võib olla palju teid, mis neist läbi viivad, mis võimaldab neil suunata teavet enamusele võrgustiku teistele tippudele. Alternatiivselt võib neil olla ka vaid väike arv neid läbivaid teid, kuid neil on siiski tugev suhtlemisoskus, kui need eksisteerivad erinevate võrgustike klastrite vahel[3].

Vahelisuskesksust arvutatakse valemiga[14]:

$$c_B(v) = \sum_{s,t \in V} \frac{\sigma(s,t|v)}{\sigma(s,t)}$$

kus V on tippude kogumik, $\sigma(s,t)$ on lühimate teede arv tipust s tippu t ja $\sigma(s,t|v)$ on nende teede arv, mis läbi mingit tippu v , kuid mis pole s või t . Kui $s=t$, $\sigma(s,t)=1$ ja kui $v \in s$, siis $\sigma(s,t|v)=0$.

2.4 Omavektorkesksus (eigenvector centrality)

Omavektorkesksus mõõdab, kuidas on seotud tipp ja kui palju see võib otseselt mõjutada võrgustiku teisi ühendatud tippe[4].

Sarnaneb astmekesksusele, aga näitab kui seotud ja kui tugev mõju on võrgus: rohkem seoseid tugevamate (kõrgema omavektorkesksusega) tippudega annab tipule kõrgema omavektorkesksuse väärtuse. Seega saab tipul olla suurem omavektorkesksus, kui:

- Tal on rohkem ühendusi.
- Tal on rohkem tähtsaid ühendusi.

Omavektorkesksuse arvutamise valem on järgmine[15]:

$$C_e(v_i) = \frac{1}{\lambda_1} \sum_j A_{ij} C_E(v_j)$$

kus A_{ij} on graafi naabrusmaatriks, λ_1 on naabrusmaatriksi juhtiv omaväärtus ja summa väljendab kõigi tipu naabertippude kesksuste summat.

3 Analoolilised lahendused

Võrgustike ja üldiselt andmete visualiseerimise ja analüüsimise tarkvarasid on olemas mitmeid. Töö autor valis välja kaks vabavaralist tarkvaralahendust, mis on laiemalt levinud. Nendeks on Gephi ja Cytoscape. Neid proovis autor oma käega järgi, importides ühe oma testfaili ja analüüsid tulemust.

3.1 Gephi

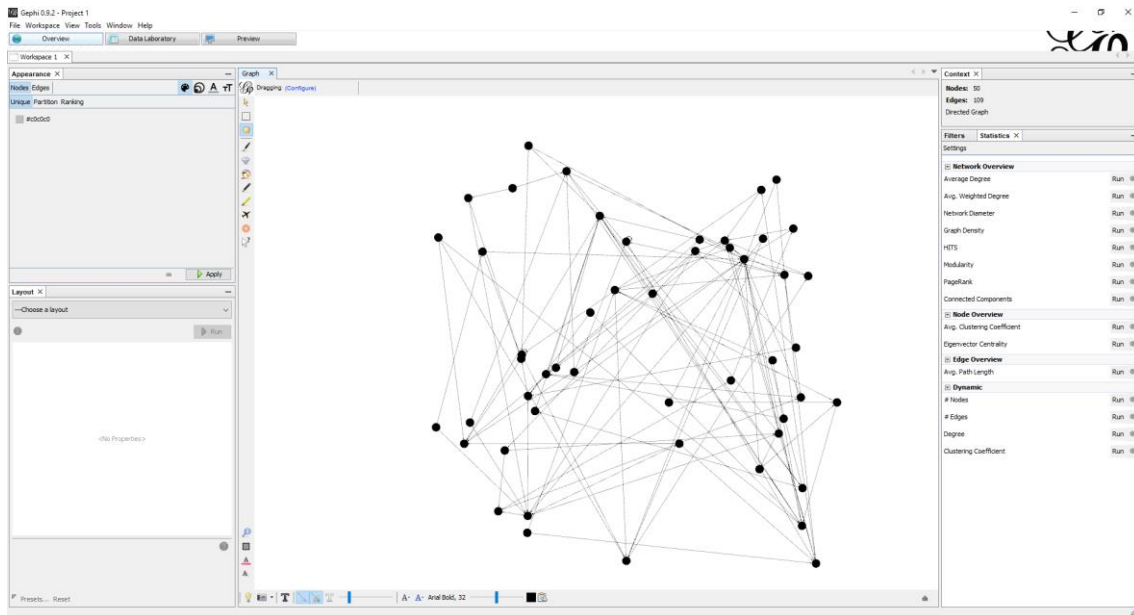
Gephi on interaktiivne visualiseerimis- ja uurimisplatvorm igat liiki võrkudele ja keerukatele süsteemidele ning dünaamilistele ja hierarhilistele graafidele[5].

Plussid:

- Kiire, saab hakkama väga suurte andmefailidega
- Väga palju võimalusi graafi kujundamiseks

Miinused:

- Kasutajaliides väga segane ja ajast maha jäänud, palju nuppe ja funktsionaalsusi. Antud tarkvaraga võtaks isegi millegi väga lihtsa visualiseerimine palju aega, seda peab õppima ennem kasutama
- Algselt genereeritud graaf ei näe kõige ilusam välja, on üleni musta värvi ja mitte midagi ütleb
- Keskusmõõde kuvab ainult graafikuna, mitte graafi enda peal



Joonis 1. Tavaseadete graaf Gephi-s

3.2 Cytoscape

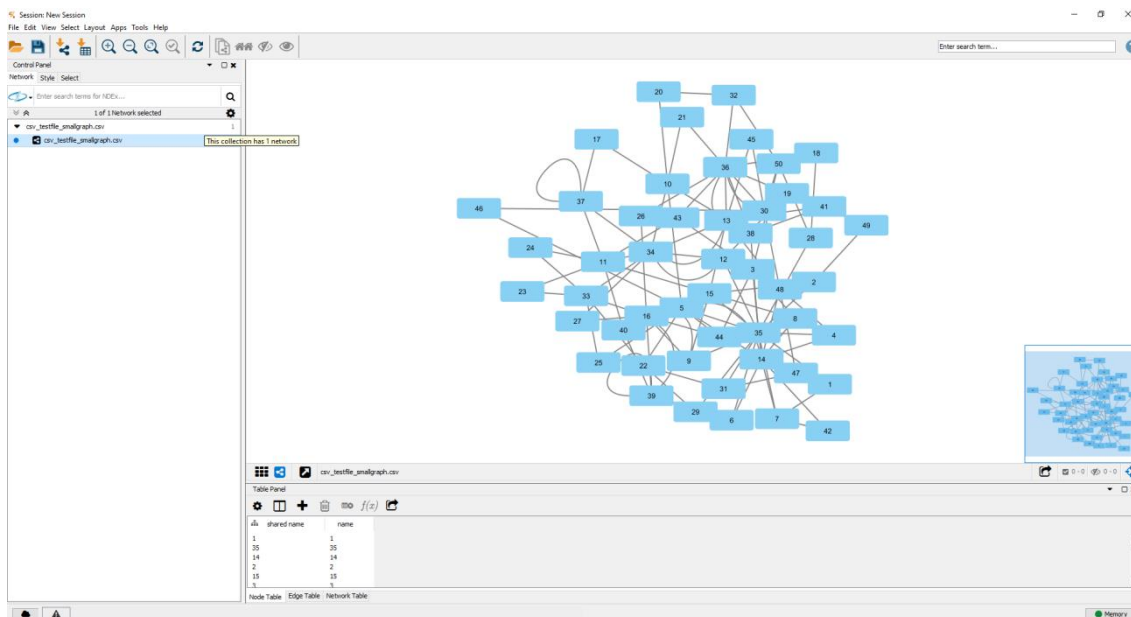
Cytoscape on avatud lähtekoodiga tarkvaraplatvorm molekulaarsete interaktsioonide võrgustike ja bioloogiliste rajatiste visualiseerimiseks ning nende võrgustike integreerimiseks annotatsioonide, geeniekspressiooni profiilide ja muude riiklike andmetega. Kuigi Cytoscape oli algselt mõeldud bioloogilisteks uuringuteks, on see nüüd üldine platvorm keerulisteks võrguanalüüsideks ja visualiseerimiseks[6].

Plussid:

- Kena ja moodne kasutajaliides
- Kiire

Miinused:

- Keskusmõõdud puuduvad täielikult, vaja lisa pluginat
- Installeerimisfail väga suur, umbes 300mb
- Genereeritud graafi tipud liiga üksteise otsas, paigutus võiks olla parem



Joonis 2. Tavaseadete graaf Cytoscape-s

Mõlema tarkvara puhul on vaja teada, kus kasutajale vajalik funktsionaalsus asub, tarkvara siseselt ringi otsides ei ole vajalikke asju kuigi lihtne leida. Funktsionaalsusi on aga mõlemad tarkvaral palju, ning kasutaja kes oskab kummagi tarkvaraga ringi käia, saab kindlasti efektiivseid tulemusi. Antud töös loodav rakendus aga püüab graafi loomise ja muutmise teha võimalikult lihtsaks ning vältida kasutaja frustratsioone.

Antud töös loodav rakendus keskendub graafi kesksusmõõtudele. Mõlemas allalaetavas tarkvaras puudus hea ja lihtne viis kesksusmõõte näha. Gephi oli võimalus arvutamiseks olemas, kuid tulemused kuvati eraldi graafikuna. Cytoscape tavaversioonis puudus kesksusmõõtude väljaarvutamise funktsionaalsus. Kiire internetiotsing aga andis mõista, et on võimalik seda siiski teha, installeerides selle tarvis plugina. See võtaks aga jällegi liigselt aega tavakasutaja seisukohast.

Allalaetavate tarkvarade analüüsis keskenduti eelkõige algtulemusele, mis saadi peale enda andmefaili üleslaadimist. Gephi puhul oli tulemus väga algeline ja võtaks palju aega, et graaf näeks välja värvirikkam ja annaks vajalikku informatsiooni. Cytoscape tegi ilusama graafi, kuid tippude paigutus oli tihtipeale üksteise otsas või väga lähedal, mis raskendas visuaalset pilti. Töös loodavas veebirakenduse proovitakse ülnimetatud puudusi vältida ja teha juba algtulemus visuaalselt meeldivaks.

4 Veebirakenduse arendus

Antud peatükis selgitatakse lahti veebirakenduse arenduskäik.

4.1 Funktsionaalsused ja vaated

Analoogiliste lahenduste analüüsis selgus, et allalaetavad tarkvarad on tavakasutajale liiga keerulised. Antud bakalaureusetöö eesmärk on luua lihtsam viis oma andmeid uurida ja visualiseerida. Rakendus peaks looma visualisatsiooni kiirelt, minimaalse konfigureerimisega. Tuginedes allalaetavate tarkvarade funktsionaalsustele sai kirja pandud funktsionaalsused, mis loodaval veebirakendusel kindlasti peaksid olema:

1. Rakendus toetab CSV (*Comma Separated Values*), TSV(*Tab Separated Values*) ja JSON(*Javascript Object Notation*) faile. CSV ja TSV on tõenäoliselt kõige laialtlevinud formaadid, millega andmeid lihtsalt ja kiirelt edasi anda. JSON sai valitud, kuna rakendus on veebipõhine ja JSON on veebis põhiliseks andmeedastusvahendiks
2. Rakendus arvutab välja eelnimetatud 4 kesksusmõõtu graafil ja visualiseerib need värviskaalana graafi tippudel
3. Kesksusmõõte saab sisse ja välja lülitada
4. Loodud graaf peab olema visuaalselt meeldiva kujundusega
5. Kasutaja saab valida suunatud ja suunamata graafi vahel
6. Kasutaja saab liigutada hiirega graafi tippe
7. Kasutaja saab muuta aktiivse kesksusmõõdu vähima väärtuse, keskväärtuse ja suurima väärtuse värve
8. Kasutaja saab muuta graafi servade värvi ja suurust

9. Kasutaja saab muuta graafi tippude suurust
10. Kasutaja saab muuta graafi ühenduste suunanoolte värvi (juhul kui on valitud suunatud graaf)
11. Kasutaja saab näha aktiivse kesksusmõõte väärtust kõikide tippude puhul tabelis
12. Kasutaja saab nupuvajutusega näha tabelit, kus on antud kõikide kesksusmõõtude väärtused kõikide tippude puhul
13. Kasutaja saab ülalnimetatud täistabelit sorteerida ja sealt väärtusi otsida
14. Kasutaja saab mingil viisil näha valitud tipu ühendusi teiste tippudega
15. Kasutaja saab näha tipu andmeid, minnes hiirega tipu peale
16. Kasutaja saab graafi brauseriaknas tervikuna liigutada
17. Kasutaja saab graafi sisse ja välja suumida

4.2 Arenduskäik

Rakenduse kasutajaliidese realiseerimiseks valiti raamistik Angular. Sellega oli autoril ka eelnev kogemus olemas. Valikus oli ka React, kuid autor polnud kindel, kui hästi see ühildub hiljem graafi visualiseerimist võimaldavate Javascripti teekidega. Arendust alustati esmalt uue Angular CLI (*Command Line Interface*) projekti loomisega. Selleks oli vaja installeerida serveripoolne Javascript raamistik NodeJS. Node-ga tuleb kaasa npm (Node Package Manager), mis sisaldab tuhandeid Javascripti teeke, mida on lihtne oma projekti lisada vaid mõne konsoolirea käsuga. Angular CLI endaga käib ka uue projekti loomine vaid ühe käsuga. Käsu sisestamisel genereeritakse projekt, mis on valmis kasutamiseks ja mida saab kohe brauseris avada. Mingit funktsionaalsust mõistagi see veel ei andnud, vaid ainult tühja „Tere tulemast“ stiilis lehe. Funktsionaalsuse lisamine oli aga nüüd lihtsustatud, kuna failide struktuur ja algseaded olid genereeritud projektis juba olemas. Seejärel saigi juba loodud rakenduse eri vaateid kirjeldavad Angulari komponendid.

Kõigepealt loodi rakenduse enda komponent. See komponent hoiab endas kõiki teisi komponente ja on esimene, mis käima läheb. Järgmisena oli vaja kindlasti avalehe

komponenti. Avalehe ülaossa loodi navigeerimise (päise) komponent, mis suunaks edasi teiste komponentide juurde.

Graafi redaktori loomiseks oli esmalt vaja viisi, kuidas graafe visualiseerida Javascripti keskkonnas. Selle jaoks sai valitud teek D3.js (*Data Driven Documents*, edaspidi D3). Variante oli veel, kuid autoril oli eelnevalt mingi kogemus antud teegiga olemas ja aega nappis, et midagi uut õppima hakata. Teegi sai lihtsalt lisada projekti npm abil.

Graaf ise sai loodud D3 ametliku graafi näite põhjal. Siin tekkis probleeme D3 ja Typescripti vahel. Kuna näide ise oli tehtud tavalises Javascriptis, siis võttis omajagu aega selle saamine Typescripti kompilaatorile meeldivale kujule. Kui baasgraaf sai valmis, siis tuli hakata talle proovima ette andma andmeid. Kuna JSON failid koosnevad Javascripti objektidest, siis polnud selle failitüübiga tegelemine kuigi raske. D3 võimaldab hõlpsasti lugeda CSV ja TSV faile, selle jaoks tulid kaasa eraldi meetodid.

Järgmisena uuris autor, kuidas saada andmed vajalikule kujule, et D3 graaf seda sisse suudaks lugeda. Vaja oli ka teeki või abivahendeid, mis hakkaks välja arvutama kesksusmõõtusid. Sellist teeki, mis suudaks arvutada välja kõik 4 vajalikku kesksusmõõtu autor ei leidnud. Silma jäid aga kaks teeki, mis tagastasid välja arvatud mõõtusi sarnasel kujul ja selle tõttu sai need valitud. Nendeks olid jsnx ja ngraph. Mõlemad said lisatud projekti npm kaudu. Ngraph võimaldas välja arvutada astme- ja läheduskesksuse, jsnx vahelikus- ja omavektorikesksuse. Seega teeb rakendus tegelikult taustal valmis kaks graafi andmestruktuuri kahes erinevas teegis. Siit saab ka kindlasti rakenduse kiirus suurema löögi, ning tuleviku edasiarenduses oleks seda vaja kindlasti adresseerida.

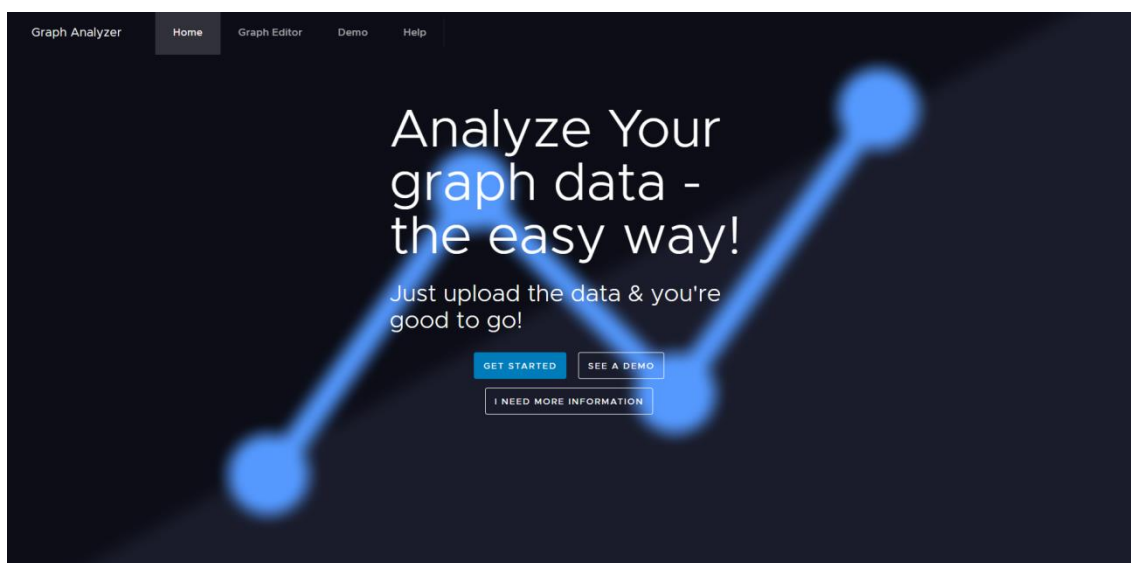
Mõtlemata tuli hakata ka rakenduse kujunduse peale. Siin oli targem kasutada mõnda olemasolevat raamistikku, mitte hakata ise kõike CSS-i kirjutama manuaalselt. See oleks töömahtu liigselt suurendanud. Töö autor soovis leida midagi, mis oleks tavakonfiguratsioonis lihtne ja minimaalne. Seda seetõttu, et graafi ennast on kasutajal võimalik teha väga kirjuks ja ülevoolavaks ning sealjuures siis võiks ülejäänud kasutajaliides olla värvide poolest lihtsakoelisem. Valikuks osutus raamistik Clarity, mis pole küll veel väga levinud, aga millel on hea Angulari tugi ja minimalistlik disain.

5 Komponentid ja vaated

Järgnevalt on kirjeldatud kõik rakenduse kasutajaliideses olevad vaated. Need on suuremad Angulari komponendid, mis võivad sisaldada väiksemad komponente.

5.1 Avalehe komponent

Avalehe ülaossa loodi navigeerimiseriba, mis võimaldab kiiret juurdepääsu kõikjale rakenduses. Rakendus sai loodud mõttega, et sellest võiks saada ka realselt kasutatav teenus. Selle tõttu on avalehel kasutatud viimasel ajal laialt levinud trendi kasutajakogemuse maailmas, milleks on CTA (*Call To Action*, eesti keeles võiks olla „Kutse tegevusele“). See tähendab, et lehe keskel on antud suure fondiga paar rakendust iseloomustavat ja tabavat lauset. Need laused peaksid kasutajat kutsuma rakendusega tutvuma ja tekitama huvi edasi uurida. Lausete alla kuvatakse nupud, mis suunavad rakenduse erinevatele lehtedele. Need dubleerivad avalehe ülaosas asetsevate navigeerimisnuppude funktsionaalsust, kuid neis on rohkem seletavat teksti. Nupp, mis viib redaktori lehele on tehtud meelega silmapaistvama kujundusega, kuna see on põhifunktsionaalsus, mida rakenduse autor tahab, et kasutaja kindlasti prooviks.



Joonis 3. Avalehe vaade

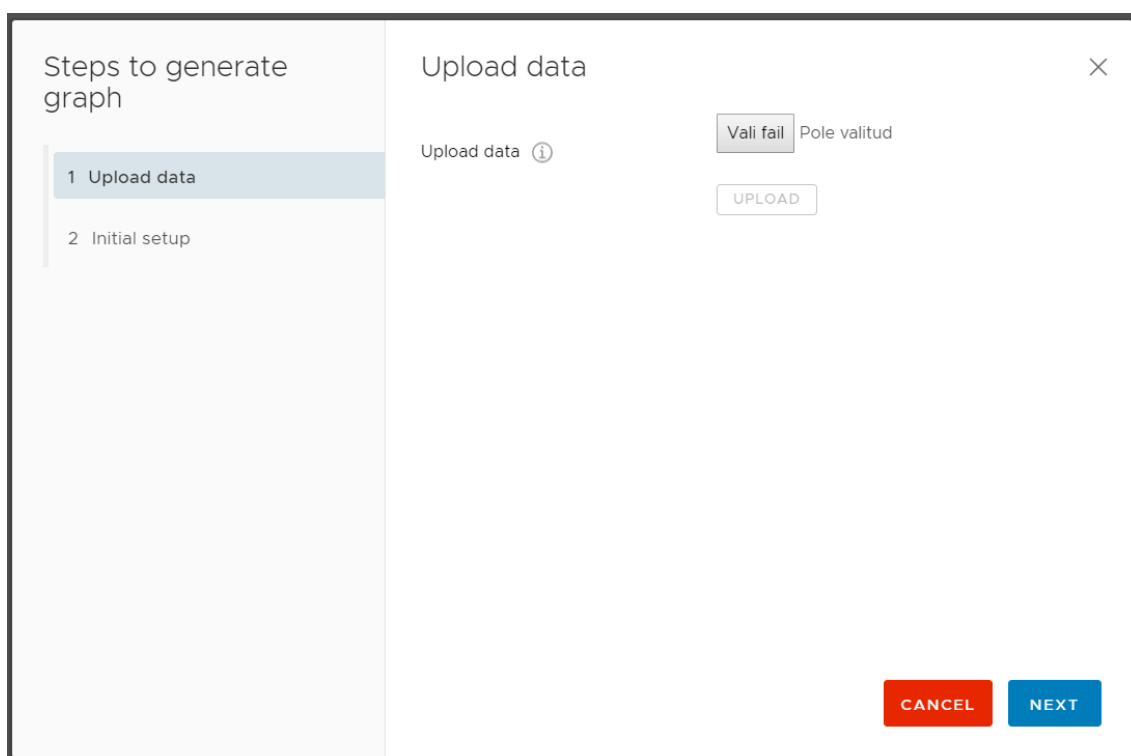
5.2 Redaktori komponent

Valides navigeerimismenüüst graafi redaktori, ilmub esimese asjana kasutajale uue graafi loomise viisard.

5.2.1 Graafi loomise viisard

Viisard on modaalaken, mis koosneb kahest leheküljest (sammust).

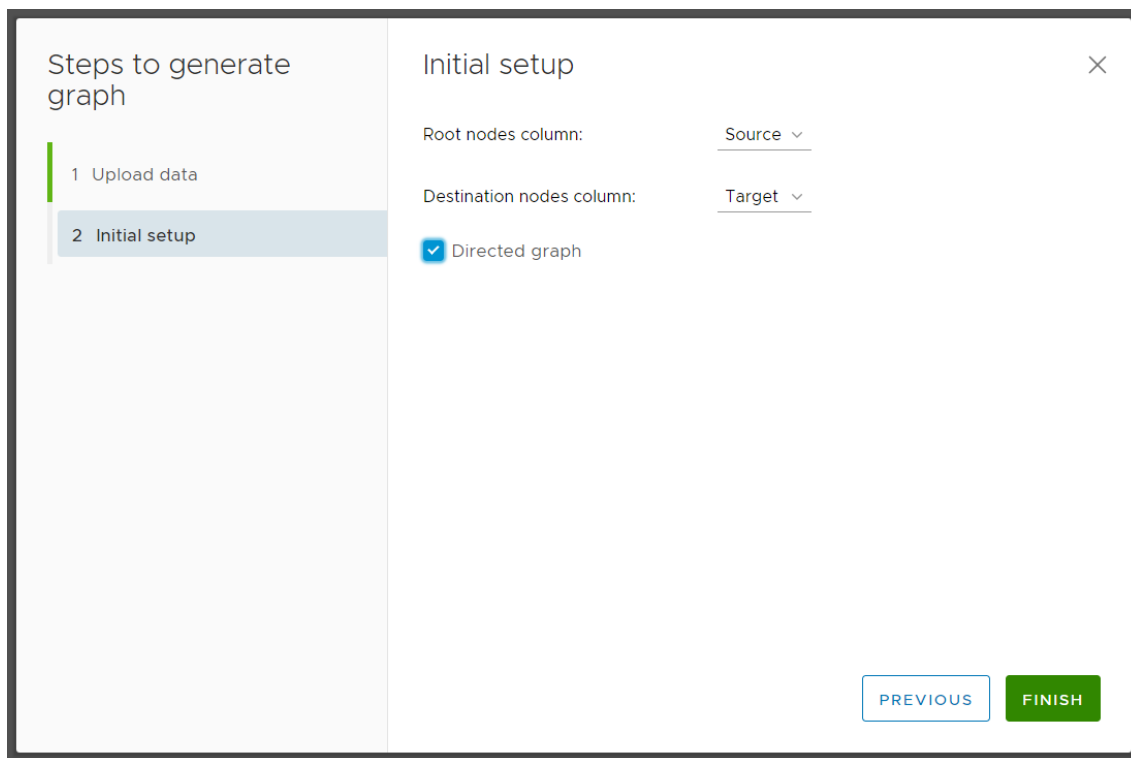
Esimesel leheküljel palutakse kasutajal üles laadida oma graafi andmeid sisaldav fail CSV, TSV või JSON formaadis. Andmefail peaks koosnema graafi ühendustest ridadena. Kui faili formaati ei toetata või faili ei suudetud lugeda, siis kuvatakse kasutajale vastav veateade.



Joonis 4. Viisardi esimese lehekülje vaade

Teisele leheküljele jõudes on rakendusel andmed sisse loetud ja kuvatakse 3 valikut. Esimene valik küsib, millisel veerul andmefailis on iga ühenduse algtypu identifikaator. Valik on antud valikukastina, kus on sees kõikide andmefaili veergude nimed. Teine valik küsib, millisel veerul andmefailis on iga ühenduse lõpptipu identifikaator. Valik on antud samuti valikukastina, kus on sees kõikide andmefaili veergude nimed. Kolmas

valik on märkeruut, mille sisse lülitades visualiseeritakse suunatud graaf (kuvatakse graafil ühenduste suunade nooli).



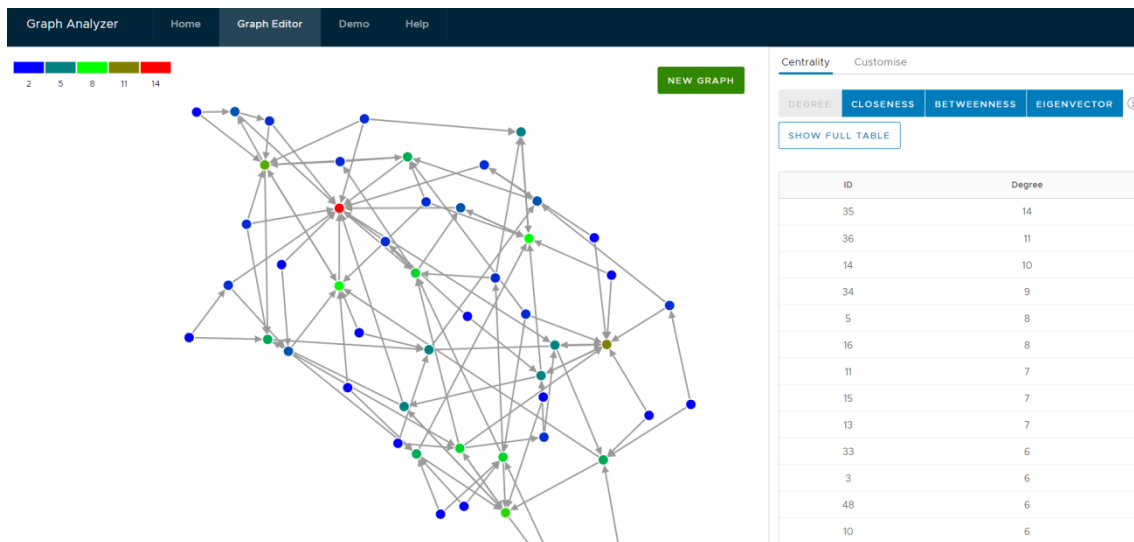
Joonis 5. Viisardi teise lehekülje vaade

Kui valikud on tehtud, vajutab kasutaja lõpeta nupule, millele järgnevalt läheb viisard kinni, redaktor avaneb ja rakendus proovib joonistada kasutaja üles laetud andmete põhjal graafi. Kui rakendus on selles edukas, siis kuvatakse visualisatsiooni aknasse joonistatud graaf. Kui tekkis viga, kuvatakse visualisatsiooni aknas veateade ning kasutajat innustatakse proovima mõnda teist faili.

5.2.2 Redaktor

Peale viisardi edukat läbimist kuvatakse kasutaja graafi redaktori vaade. Redaktor on leheküljel jaotatud kaheks.

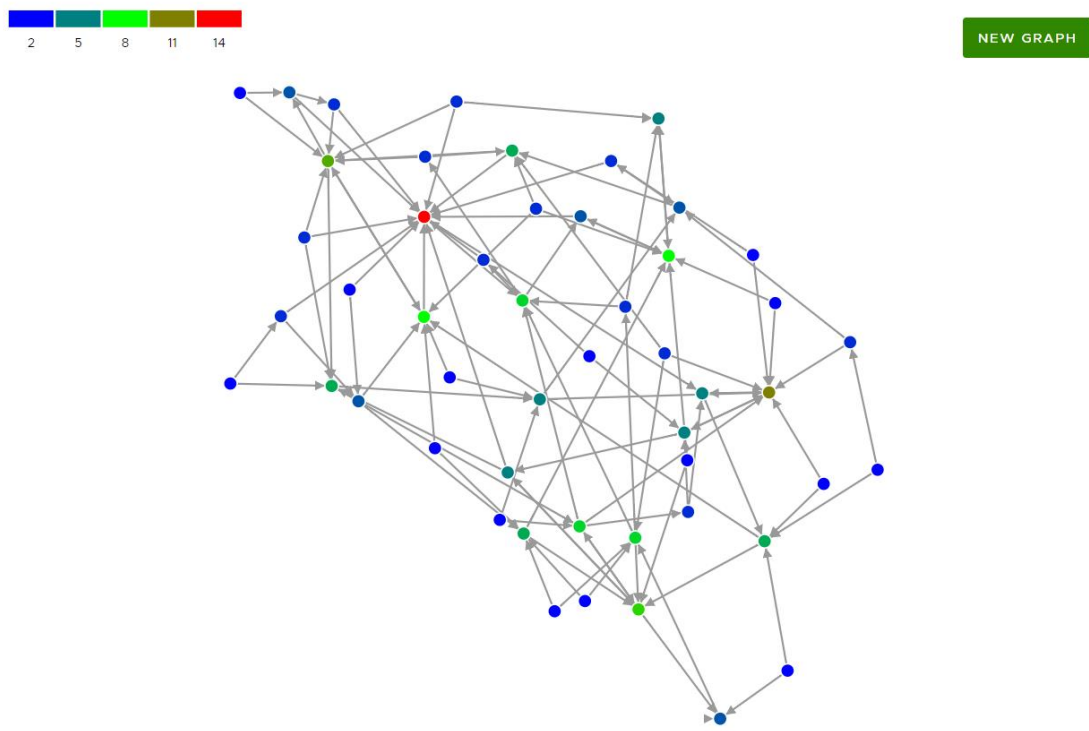
Vasakul pool ja 2/3 leheküljest moodustab graafi visualisatsiooni vaade, paremal pool 1/3 leheküljest moodustab tööriistariba.



Joonis 6. Redaktori täisvaade

Graafi aknas kuvatakse kasutaja poolt üles laetud andmete põhjal visualiseeritud graaf. Asetades hiirekursori graafi mõne tipu peale, kuvatakse infoaken antud tipu identifikaatoriga. Graafi igat tippu saab liigutada hiirega lohistades. Graafi ennast saab liigutada lohistades hiirega mõnda graafi osa, mis pole tipp. Graafi saab sisse ja välja suumida hiirerullikut kasutades. Vajutades hiirega mõne tipu peale jäävad nähtavaks ainult valitud tipp ning kõik tipud, mis on temaga ühenduses. Kui kasutaja lülitas sisse viisardist suunatud graafi, siis kuvatakse ka graafi servadele ühenduste suunanooled.

Graafi akna üleval vasakus servas kuvatakse kasutajale värvilegend, mis vastab hetkel valitud kesksusmõõdule. Üleval paremas servas on roheline nupp uue graafi tegemiseks. Seda vajutades viiakse kasutaja tagasi viisardi algusesse.



Joonis 7. Redaktori graafi osa

Tööriistariba koosneb omakorda kahest vaatest, mida saab sakkidel klikkides muuta. Esimese saki alt avaneb kesksuste vaade. Siin on antud 4 nuppu. Nendega on võimalik erinevaid kesksusmõõte sisse ja välja lülitada. Korraga saab aktiivne olla ainult üks kesksusmõõt.

Nuppude alla kuvatakse hetkel valitud kesksusmõõdu andmetabel. Tabelis on antud iga tipu kohta tema identifikaator ja tema kesksusmõõt. Vaikimisi on valitud astmekesksus, kuna see on uuele kasutajale eeldatavalt kõige lihtsamini mõistetav kesksusmõõt. Tabel on automaatselt sorteeritud kesksusmõõdu väärtuse kahanemise järjekorras, kuna eelkõige tunnevad kasutajad huvi tavaliselt just kõige kõrgema mõõdu väärtuse vastu. Kesksusmõõtude nuppude järel on antud väike infonupp, mis sisaldab vajadusel ka linki abilehele.

DEGREE	CLOSENESS	BETWEENNESS	EIGENVECTOR	
--------	------------------	--------------------	--------------------	---

SHOW FULL TABLE

ID	Degree
35	14
36	11
14	10
34	9
5	8
16	8
11	7
15	7
13	7
33	6
3	6
48	6
10	6
12	5

Joonis 8. Redaktori tööriistariba, avatud kesksuse sakk

Antud on ka täistabeli vaate nupp. Vajutades täistabeli vaate nupule, kuvatakse modaalkaken, kus on tabelina antud kõikide tippude kõikide kesksusmõõtude väärtused. Tabelit saab sorteerida kasvavalt ja kahanevalt, vajutades valitud veeru päise pealkirja peale. Tabelit saab filtreerida mingi otsingupäringu järgi vajutades valitud veeru päises paremal pool asuva filtreerimise ikooni peale. Kasutaja leiab need juhised ka modaalkna pealkirja lõpus oleva infonupu alt.

ID	Eigenvector	Betweenness	Closeness	Degree
1	0.17659	0.00105	0.35766	3
2	0.1371	0.02358	0.37121	3
3	0.22405	0.06166	0.42241	6
4	0.17477	0.00342	0.36842	3
5	0.22698	0.12423	0.44545	8
6	0.15554	0.01987	0.37405	3
7	0.19144	0.01301	0.36029	4
8	0.1448	0.00576	0.36842	3
9	0.14822	0.01784	0.392	4
10	0.12502	0.09367	0.39837	6
11	0.14295	0.0867	0.39837	7
12	0.16326	0.04879	0.42982	5
13	0.16234	0.08286	0.42982	7

Joonis 9. Kesksusmõõtude täistabeli vaade

Tööriistariba teise saki alt avaneb graafi kohandamise vaade. Siin saab kasutaja graafi visuaali muuta. Muuta saab tippude värve, võttes arvesse hetkel aktiivse kesksusmõõdu väärtusi. Muuta saab mõõdu miinimum väärtuse värvi, mõõdu keskvaartuse värvi ja mõõdu maksimumväärtuse värvi. Valitud kolmest värvist moodustatakse värviskaala, mis iseloomustab kesksusmõõdu muutumist graafil.

Lisaks on kohandamise vaates värvivalikud graafi servade ja servade suunanoolte jaoks (kui viisardist sai valitud suunatud graaf). Viimasena saab kasutaja muuta tippude ja servade suurusi. Kõiki stiilimuutusi saab kinnitada nupuvajutusega, mille tagajärjel ilmuvad muudatused koheselt graafile.

Color of minimum values:	<input type="color" value="#0000FF"/>
Color of median values:	<input type="color" value="#00FF00"/>
Color of maximum values:	<input type="color" value="#FF0000"/>
Color of graph links:	<input type="color" value="#808080"/>
Size of nodes: *	<input type="text" value="7"/>
Width of links: *	<input type="text" value="1"/>

Joonis 10. Redaktori tööriistariba, avatud kohandamise sakk

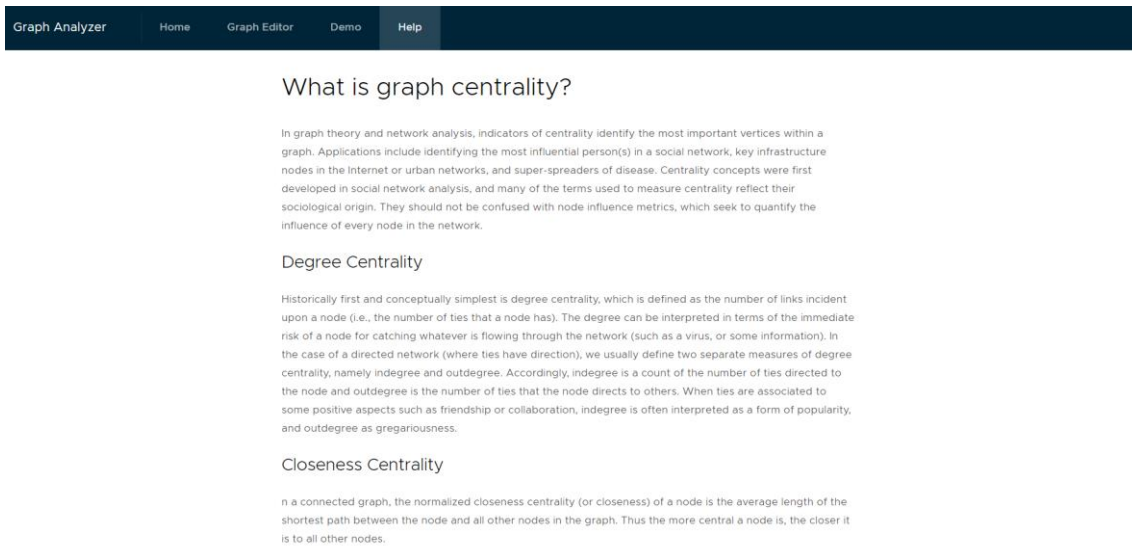
5.3 Demo komponent

Graafi loomine redaktoris eeldab alati enda faili üleslaadimist. Kuid kui kasutajal seda kohe pole võtta ja tahaks lihtsalt näha mida rakendus endast kujutab, siis otsustati luua demo komponent. Demo komponent on kasulik ka ettekandel funktsionaalsuse näitamiseks, kuna ei pea tegelema faili üleslaadimisega ja graafi vaade avaneb kohe.

Demo komponent kasutab redaktori komponenti, kuid mõningate kitsendustega. Esiteks on graafi loomiseks vajalik fail võetud töö autori testfailide hulgast ja loetakse lehekülje avamisel kohe sisse. See tähendab, et faili üleslaadimise viisardit ei kuvata üldse, vaid kohe on ees graafi vaade. Teiseks ei kuvata uue graafi loomise nuppu. Seda ei pidanud autor vajalikuks kuvada, kuna kasutaja saab hõlpsasti klikata ka redaktori nupule navigeerimise ribal.

5.4 Abilehe komponent

Abileht loodi kasutajale juhuks, kui midagi jääb rakenduse juures arusaamatuks. Abilehel on peamiselt informatsioon kesksusmõõtude kohta, kuid lehekülje allosast leiab ka juhendi graafi kohandamise jaoks. Redaktoris on kesksusmõõtude muutmise nuppude juures ja kohandamise vaate all ka infonupud, mis viitavad vajadusel abilehele.



The screenshot shows a dark blue navigation bar at the top with the following items: Graph Analyzer, Home, Graph Editor, Demo, and Help. The main content area is white and contains the following text:

What is graph centrality?

In graph theory and network analysis, indicators of centrality identify the most important vertices within a graph. Applications include identifying the most influential person(s) in a social network, key infrastructure nodes in the Internet or urban networks, and super-spreaders of disease. Centrality concepts were first developed in social network analysis, and many of the terms used to measure centrality reflect their sociological origin. They should not be confused with node influence metrics, which seek to quantify the influence of every node in the network.

Degree Centrality

Historically first and conceptually simplest is degree centrality, which is defined as the number of links incident upon a node (i.e., the number of ties that a node has). The degree can be interpreted in terms of the immediate risk of a node for catching whatever is flowing through the network (such as a virus, or some information). In the case of a directed network (where ties have direction), we usually define two separate measures of degree centrality, namely indegree and outdegree. Accordingly, indegree is a count of the number of ties directed to the node and outdegree is the number of ties that the node directs to others. When ties are associated to some positive aspects such as friendship or collaboration, indegree is often interpreted as a form of popularity, and outdegree as gregariousness.

Closeness Centrality

In a connected graph, the normalized closeness centrality (or closeness) of a node is the average length of the shortest path between the node and all other nodes in the graph. Thus the more central a node is, the closer it is to all other nodes.

Joonis 11. Abilehe vaade

6 Kasutatud tehnoloogiad

Siin peatükis on toodud antud töös loodud veebirakenduse loomisel kasutatud tehnoloogiad.

6.1 Javascript

JavaScript on firma Netscape Communications Corporation poolt väljatöötatud programmeerimise keel, mida kasutatakse dünaamilistel veebilehtedel. Javascript töötab kliendi poolel veebis ning selle ülesanne on kontrollida veebilehe käitumist[7].

6.2 Typescript

Typescript on Javascripti ülemhulk. See tähendab, et igasugune kehtiv Javascripti kood toimib ka Typescripti koodina. Typescript on loodud suuremõõtmeliste rakenduste arendamiseks Javascriptis.

Erinevalt Javascriptist kasutab Typescript staatilist tüüpimist ja klasse. Typescript aitab eelkõige vähendada Javascripti omapärast tingitud vigu. Typescript kompileerub brauseris Javascriptiks[8].

6.3 Angular

Angular on struktuurne raamistik, mis on loodud dünaamiliste veebirakenduste jaoks. Antud raamistik võimaldab kasutada HTML malli ja laiendada HTML-i süntaksit, mis väljendab rakenduse komponente selgelt ja lühidalt. Angular on ise kirjutatud TypeScript-vormingus[9].

Iga Angulari projekt koosneb komponentidest. Iga komponent ise koosneb HTML failist, Javascript (Angulari puhul Typescript) failist, CSS failist ja testide failist (antud projekti puhul ei kasutatud). Igat komponenti saab teise komponendi sees välja kutsuda, tekitades puu laadse struktuuri.

6.4 D3.js

D3.js on Javascripti teek andmete manipuleerimiseks ja visualiseerimiseks veebibrauseris. Teek loob SVG, HTML5 ja CSS standardite abiga veebilehitsejate tehnoloogiaga ühilduvaid dünaamilisi ja interaktiivseid visualiseeringuid[10].

6.5 Clarity

Töös loodud veebirakenduse kasutajaliidese kujunduse jaoks on kasutatud VMware[11] poolt loodud kasutajaliidese raamistikku Clarity. CSS-i süntaksi poolest on Clarity väga sarnane palju enam levinud Bootstrapile. Visuaalselt näeb aga Clarity töö autori hinnangul välja lihtsam ja puhtam kui Bootstrap, selle tõttu ka valik.

Tegu pole kuigi laialt levinud raamistikuga ning arendustöö alles käib, kuid antud töös loodava veebirakenduse tegemiseks sobis hästi. Clarity sisaldab ka Angulari komponente mõne keerulisema kasutajaliidese osa jaoks, mis oli suureks abiks arendusel.

6.6 WebStorm

WebStorm on JetBrainsi poolt loodud Javascripti IDE (integrated development environment – sisseehitatud arenduskeskkond). WebStorm pakub automaatset kooditäitmist, vigade tuvastamist ja erinevaid refaktoorimisvõimalusi[12]. WebStorm toetab juba tavakonfiguratsioonis HTML-i, CSS-i ja Javascripti, ning uuemates versioonides on olemas ka Angular CLI tugi.

6.7 NodeJS ja npm

„Node.js on tarkvaraplatvorm, mis võimaldab programmeerida serveripoolseid rakendusi kasutades Javascripti. Node-i siseselt kasutab see Google-i V8 mootorit koodi käivitamiseks ning sisaldab HTTP server teeki, võimaldades jooknutada veebiserverit ilma Apache või muu veebiserverita“[13].

„Node loodi aastal 2009 Ryan Dahl-i poolt. Aastal 2011 toodi välja ka npm, mis võimaldab lihtsalt hallata mooduleid“[13].

Antud töös kasutatakse ainult npm-i Javascripti teekide installeerimiseks. Serveripoolset NodeJS programmikoodi töös ei ole.

7 Võimalikud edasiarendused

Antud lõputöö raames valminud veebirakendust saaks edasi arendada mitmel viisil. Veebirakendusel ei ole kaugeltki nii palju funktsionaalsust kui allalaetavatel tarkvaradel, kuid rakendus on mõeldud prototüübina, millele saab modulaarselt funktsionaalsust hiljem juurde lisada.

7.1 Mobiiliversioon

Veebirakenduse versioon, mis valmis lõputöö esitamise tähtjaks ei toeta mobiilseadmeid. Rakendus on küll avatav, kui kujunduse poole pealt on väga raske kasutada ja mõned kujunduse elemendid kas nihkuvad paigast ära või ei tööta nii, nagu peaks. Rakendust arendades keskendus autor siiski laua- ja sülearvutite peale. Mobiiliversioonis tuleks näiteks redaktori kujundus täiesti ümber mõelda. Praegusel kujul, kui mahutada graaf ja tööriistariba korraga ekraanile, ei jääks graafiga tegutsemiseks piisavalt ruumi.

7.2 Graafi kujundus

Graafi kohandamise sätteid saaks lisada rakendusse juurde mitmeid.

Näiteks lisada graafi tippude kuju muutmise variandid. Praeguses rakenduses on tipud ainult ringikujulised ja seda muuta ei saa. Suuremates visualiseerimise tarkvarades saab tippe visualiseerida igasuguste geomeetriliste kujunditena, ka kasutaja poolt lisatud piltidena.

Saaks ka lisada graafi tippude paigutuse variante. Praegune lahendus üritab grupeerida seotud tippe klastritesse, vältides sealhulgas üksteise peale minemist. Üks alternatiivne paigutuse variant oleks graaf kujutada puuna.

7.3 Andmetüüpide tugi

Kasutaja poolt üleslaetavate failide tüüpide arvu saaks suurendada. Allalaetavatel tarkvaradel läheb toetatud failitüüpide arv juba kümnetesse ja sellega antud veebirakendus ei suudaks sammu pidada. Siiski saaks lisada toetatavaid andmetüüpe juurde, näiteks Exceli ja XML failitüübid. Kõik oleneb sellest, millised võimalused (teegid) on Javascriptil olemas nende failitüüpide sisselugemiseks ja JSON-iks konverteerimiseks.

7.4 Keskusmõõdud

Graafide keskusmõõde on olemas veel peale nende, mis loodud rakenduses saadaval on. Need 4, mis hetkel saadaval on, valiti eelkõige nende populaarsuse ajendil võrgustike analüüsis. Mõned keskusmõõdud annavad aga ka kohati liiga palju sarnast informatsiooni ja selle tõttu arvas autor, et saadaolevatest mõõtudest piisas antud töö jaoks.

8 Kokkuvõte

Käesolevas töös uuriti saadavalolevaid allalaetavaid tarkvarasid võrgustike (täpsemalt graafide) analüüsi ja visualiseerimise jaoks. Graafide analüüsis on tähtsal kohal kesksusmõõdud, millega saab leida graafi kõige olulisemaid liikmeid mitmete kriteeriumite alusel. Leiti, et olemasolevad tarkvarad on liiga keerulised kasutada ning neil kas täiesti puuduvad või on limiteeritud kesksusmõõtude välja arvutamise funktsionaalsus. Seetõttu seati eesmärgiks välja töötada lihtsustatud funktsionaalsusega lahendus, mis töötab veebibrauseris ja mida ei pea alla laadima.

Rakenduse loomisel pöörati tähelepanu eelkõige nendele funktsionaalsustele, mida graafide analüüsis kõige rohkem peaks vaja minema. Nendeks olid kindlasti graafi stiili muutmise võimalused, erinevad viisid graafi liigutada ja lähemalt uurida ning ka mõne kesksusmõõdu välja arvutamine ja nende kuvamine värviskaalana graafi tippudel.

Töö eesmärk sai täidetud. Valmis kasutuskõlblik veebirakendus, mis võimaldab kasutajal oma graafi andmed üles laadida. Rakendus genereerib andmetest graafi visualisatsiooni, mida saab mitut moodi manipuleerida ja ka kujundust muuta. Rakendus suudab välja arvutada 4 erinevat kesksusmõõtu, mille tulemused kuvatakse graafi iga tipu peal värvina. Kesksusmõõtude suurusi saab näha tabelis iga tipu kohta.

Rakendus valmis prototüübina, mida on võimalik täiendada veel lugematute funktsionaalsustega graafide analüüsimise valdkonnast. Rakendus töötab laua- ja sülearvutites.

Kasutatud kirjandus

- [1] Cecilia Mascolo “Centrality Measures” [WWW]
<https://www.cl.cam.ac.uk/teaching/1213/L109/stna-lecture3.pdf> (18.05.2018)

- [2] About closeness [WWW]
https://www.ibm.com/support/knowledgecenter/en/SS3J58_9.0.4/com.ibm.i2.anb.doc/about_closeness.html (18.05.2018)

- [3] About betweenness [WWW]
https://www.ibm.com/support/knowledgecenter/en/SS3J58_9.0.4/com.ibm.i2.anb.doc/about_betweenness.html (18.05.2018)

- [4] About eigenvector [WWW]
https://www.ibm.com/support/knowledgecenter/en/SS3J58_9.0.4/com.ibm.i2.anb.doc/about_eigenvector.html (18.05.2018)

- [5] Gephi [WWW] <https://github.com/gephi/gephi/wiki> (19.05.2018)

- [6] What is Cytoscape [WWW] http://www.cytoscape.org/what_is_cytoscape.html
(19.05.2018)

- [7] Javascript konspekt [WWW] http://puhang.tpt.edu.ee/raamatud/JavaScript_konspekt.pdf
(19.05.2018)

- [8] Typescript [WWW] <https://www.typescriptlang.org/> (19.05.2018)

- [9] AngularJS õppematerjal [WWW] http://www.cs.tlu.ee/teemad/get_file.php?id=400
(19.05.2018)

- [10] D3.js [WWW] <https://d3js.org/> (19.05.2018)

- [11] Clarity Design System [WWW] <https://vmware.github.io/clarity/> (19.05.2018)

- [12] JetBrains WebStorm IDE [WWW] <https://www.jetbrains.com/webstorm/>
(19.05.2018)

- [13] NodeJS [WWW] minitorn.tlu.ee/~jaagup/kool/java/kursused/14/progl/Node.docx
(19.05.2018)
- [14] Betweenness Centrality [WWW]
<http://jsnetworkx.org/api/#/v/v0.3.4/betweennessCentrality> (21.05.2018)
- [15] Katharine A. Anderson “Network Measures: Centrality and Prestige” [WWW]
https://www.andrew.cmu.edu/user/andersok/Katharine_A._Anderson/Social_Economic_and_Information_Networks_files/CentralitySlides.pdf (21.05.2018)
- [16] Centrality [WWW] <https://en.wikipedia.org/wiki/Centrality> (22.05.2018)