

TALLINNA TEHNIKAÜLIKOOL
Infotehnoloogia teaduskond

Tarvo Tammejuur 176293IDDR

Küünalmustrite tuvastamise ja tulemuse visualiseerimise rakenduse prototüübi arendus

Diplomitöö

Juhendaja: Kristiina Hakk
Doktorikraad

Tallinn 2021

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 Tarvo Tammejuur

17.05.2021

Annotatsioon

Antud töös on vaatluse all finantsgraafikute tehniline analüüs, täpsemalt küünalgraafikutelt defineeritud mustrite otsimine, selle automatiseerimine ja tulemuste kuvamine. Vaatluse all on kauplemisplatvormid, ning sarnaseid lahendusi pakkuvad rakendused.

Diplomitöö käigus luuakse veebirakendus, mille tehnoloogiline valik põhineb eesmärgil luua üldkasutatav lahendus, mis kataks täna turul saadaval olevate rakenduste puudusi, jäädes piisavalt sõltumatuks välistest keskkondadest ja tehnoloogiast.

Lõputöö on kirjutatud eesti keeles ning sisaldab teksti 23 leheküljel, 7 peatükki, 16 joonist, 4 tabelit.

Abstract

Development of Candlestick Patterns Recognition and Visualization Software Prototype

The thesis covers technical analysis, more specifically finding patterns in candlestick charts, its automation and visualisation of results. Bigger trading platforms and libraries, with similar functionality, will be analyzed and weakpoints specified.

From that, a more general-use web application, in candlestick pattern recognition domain, will be developed with minimal dependencies to other libraries and platforms.

The thesis is in Estonian and contains 23 pages of text, 7 chapters, 16 figures, 4 tables.

Lühendite ja mõistete sõnastik

API	<i>Application Programming Interface</i> . Liides olemasoleva tarkvaraga suhtlemiseks
Brauser	Rakendusporgramm internetis veebilehtede kuvamiseks
Bugi	<i>Bug</i> . Tarkvara arenduse käigus tekkinud soovimatu tulemus või viga
Cache	„Vahemälu“. Koht, kus hoitakse andmeid nende kiireks uuesti kasutamiseks
Clipboard	Lõikelaud. Mälu operatsioonisüsteemis lõikamise-kopeerimise- kleepimise funktsionaalsuse kasutamiseks.
CSS	<i>Cascading Style Sheets</i> . Märgistuskeel, milles küljendatakse veebileht
GET-päring	Üks HTTP meetoditest. Kasutatakse ressursi pärimiseks
HMR	<i>Hot Module Replacement</i> . Webpack-dev-server veebiserveri lisafunktsioon veebilehel moodulite reaalajas väljavahetamiseks, ilma täieliku brauseri värskendusega
HTML	<i>HyperText Markup Language</i> . Märgistuskeel, milles märgendatakse veebilehti. Seab paika dokumendi struktuuri
HTTP	<i>Hypertext Transfer Protocol</i> . Protokoll teabe edastamiseks arvutivõrkudes kasutades klient-server ja päring-vastus suhtlust
HTTP klient	Rakendus, millega HTTP päringuid tehakse, nt. veebibrauser
JSON	<i>JavaScript Object Notation</i> . Tekstvormingus ja programmeerimiskeelest sõltumatu andmevahetusvorming, mis põhineb JavaScripti programmeerimiskeele alamhulgal
Konsool	Käsurea kasutajaliides. Näiteks „brauseri konsool“
NPM	<i>Node Package Manager</i> . Node käitussüsteemi paketi haldur JavaScript pakettide haldamiseks
OOP	<i>Object-Oriented Programming</i> . Objektide põhine programmeerimise paradigma
Plugin	Pistikprogramm. Tarkvaramoodul, mis annab lisa funktsionaalsust teisele süsteemile.
REST	<i>Representational state transfer</i> . Arhitektuuriline moodus serveriga suhtluse loomiseks

Teek	Funktsioonide, moodulite vms komponentide kogum korduvkasutamise eesmärgil
Tooltip	Infot/ liidese komponenti seletav lühikirjeldus
URI	<i>Uniform Resource Identifier</i> . Internetis oleva infoallika üheseks määramiseks mõeldud sõne
URL	<i>Uniform Resource Locator</i> . Internetiaadress, mida kasutatakse infoallikate leidmiseks ja kasutamiseks internetis
UI	<i>User Interface</i> . Kasutaja ja programmi funktsioonide vaheline liides - kasutajaliides
UX	<i>User experience</i> . Kasutajakogemus, emotsioonid toote/ teenuse kasutamisel

Sisukord

1 Sissejuhatus	10
2 Tehniline analüüs.....	11
2.1 Küüналgraafikud	13
2.2 Küüналgraafikute mustrid	13
3 Sarnased, olemasolevad lahendused	17
4 Lähteülesanne ja rakenduse teostus	20
4.1 Sisend.....	20
4.2 Väljund	22
4.3 Andmestik.....	23
4.4 Tehnoloogiate valik	27
5 Tulemus	29
6 Edasiarendus	30
7 Kokkuvõte	32
Kasutatud kirjandus	33
Lisa 1 – Lihtlitsents lõputöö reprodutseerimiseks ja lõputöö üldsusele kättesaadavaks tegemiseks	37

Jooniste loetelu

Joonis 1. Graafikutüübid, küünlad [7].	12
Joonis 2. Swedbank (SWED-A.ST), 5-päeva küünalgraafiku näide [9].	13
Joonis 3. Doji star muster.	14
Joonis 4. Dragonfly doji muster.	14
Joonis 5. Gravestone doji muster.	15
Joonis 6. Hammer muster.	15
Joonis 7. Hanging man muster.	16
Joonis 8. Spinning top muster.	16
Joonis 9. Andmete sisestamise näide.	21
Joonis 10. Küünla osade valik.	21
Joonis 11. Andmete kuvamise näide.	22
Joonis 12. Binance API küünla näide JSON-struktuuris [31].	24
Joonis 13. Coinbase API küünla näide JSON-struktuuris [32].	24
Joonis 14. Marketstack JSON küünla näide [36].	26
Joonis 15. Polygon JSON küünla näide [37].	26
Joonis 16. AlphaVantage JSON küünla näide [39].	26

Tabelite loetelu

Tabel 1. CandleScanner plussid-miinused [26].....	17
Tabel 2. MetaTrader plussid-miinused [27].	18
Tabel 3. ProRealTime plussid-miinused [28].....	18
Tabel 4. Binance ja Coinbase JSON künla tähendus.....	25

1 Sissejuhatus

Tehnilist analüüsi kasutatakse finantsvahendi (edaspidi vahend), nt valuuta, aktsia, krüptoraha, fondid jms, hinna ennustamiseks turuandmete ajaloo järgi. Alguse saanud Jaapanist, nüüdseks üle maailma kasutatavat tehnilise analüüsi meetodi isaks peetakse Munehisa Homma't [1], [2], kes avastas, et peale hinna ja pakkumise-nõudluse suhte on turg tugevalt mõjutatud ka kauplejate emotsioonidest [3].

Visuaalsetest tehnilise analüüsi meetoditest üheks tuntuimaks on küünalgraafikud, mis kuvavad vahendi hinna kõikumisi konkreetsel ajaperioodil ja näitavad kauplejate emotsiooni visuaalselt graafikul tekkinud mustrite näol [4].

Turg on viimaste aastakümnetega tehnoloogiliselt kiiresti arenenud. Ei piisa enam "näpuga järje ajamisest". Kadumas on aeg, kus käsitsi graafikutelt mustreid otsitakse. Et püsida konkurentsivõimeline tuleb tehnoloogia arenguga kaasa minna. Loodud on kümneid suuri kauplemisplatvorme, igaüks oma võimaluste/ laienditega, kuid turul puuduvad üldkasutatavad, konkreetsemat probleemi lahendavad rakendused, mis ei sõltuks operatsioonisüsteemist, finantsvahendi tüübist, ega kauplemisplatvormist.

Ülesanne on luua ühe-leheline veebirakendus, mis küünalgraafikutelt eeldefineeritud mustreid kasutaja sisestatud andmestikust üles leiab ja graafikul visuaaliseerida suudab.

Diplomitöö käigus võetakse vaatluse alla praegu turul olevad sarnased lahendused, nii suurte kauplemisplatvormide kui ka väiksemate teekide näol. Analüüsitakse nende positiivseid ja negatiivseid omadusi. Selle põhjal selgitatakse välja täpsemad nõuded loodavale rakendusele. Analüüsitöö tulemusena selgitatakse ka välja tehnoloogiline valik, et luua üldkasutatav ja paindlik lahendus graafiku andmestikult küünal-mustrite otsimiseks ja tulemuste kuvamiseks küünalgraafikul.

2 Tehniline analüüs

Peamine finantsvahendi uurimise viis on tehniline analüüs. Tehnilise analüüsi erinevus fundamentaalanalüüsi ees on see, et see ei keskendu ettevõtte finantstulemustele, ega tulevikuväljavaadetele, nt aruanded, prognoosid jms [5]. Sellest tulenevalt on vahendi uurimisel võtmeküsimuseks sõna „mis?“, mitte „miks?“, kogu informatsioon peegeldub hinnas ja seda mõjutab inimeste emotsioonid [6].

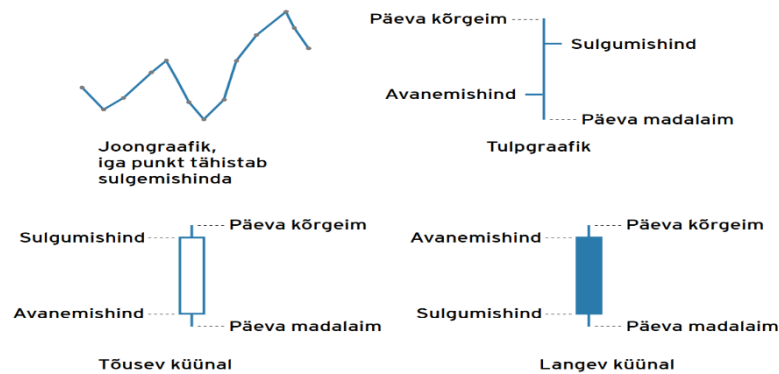
Tehniline analüüs uurib hinnaliikumisi ajas – tekkib aegrida. Seda visualiseeritakse peamiselt graafikuna ja ta on kaupleja peamine töövahend, millelt seaduspärasusi/mustreid otsitakse. Lisainformatsiooni pakuvad erinevad indikaatorid jpm andmetega seotud väljatöötatud reeglistikud.

Graafiku tüüpe on mitmeid ja nad kuvavad andmeid erineva infotasemega, enamlevinud on: joongraafik, tulpgraafik ja küünalgraafik:

Joongraafik. Väljendub punktijadana, mis on ühendatud joonega. Iga punkt näitab sulgemishinda mingil ajahetkel. Sellist graafikutüüpi kasutatakse peamiselt kiire ülevaate saamiseks, ning madalamatasemelist analüüsi sellel ei tehta, sest ta pakub vähe informatsiooni.

Tulpgraafik. Kujutatud vertikaalsete joontena, mille külgedelt ulatuvad välja väiksemad horisontaalsed jooned. Võrreldes eelmise graafikutüübiga pakub tulpgraafik rohkem infot, sest näha on lisaks perioodisest hinnaliikumist – kõrgeimat ja madalaimat (vertikaalse joone algus ja lõpp-punkt) ja avanemis- ning sulgemishinda (vertikaalse joone asetus horisontaalse joone suhtes), mingil ajahetkel.

Küünalgraafik. Sarnane tulpgraafiku pakutava informatsiooniga, kuid erinev esitusviis ja sellepärast lihtsam lugeda. Avanemis -ja sulgemishinda näitavad piklikud ristkülikud (moodustab küünla keha), ning kõrgeimat ja madalaimat hinda tähistavad ristkülikust ulatuvad vertikaalsed jooned (moodustab küünlatahi/ varju). Keha värvus annab aimu millise trendiga on antud ajavahemikul tegemist: seest roheline tähendab tõusu, s.t. hinna sulgemishind on suurem kui avanemishind. Punane värvus viitab langusele, s.t. hinna sulgemishind on alla avanemishinna. Rohelist ja punast küünlakeha värvust võib asendada ka vastavalt valge ja „täidetud“ värvusega, nagu allpool toodud pildil (vt Joonis 1).



Joonis 1. Graafikutüübid, küünlad [7].

Tulenevalt diplomitöö teemast käsitleb autor järgnevalt küünalgraafikuid ja selle mustreid täpsemalt.

2.1 Küüналgraafikud

Küüналgraafikul kuvab konkreetset ajahetke üks küüнал. Küüналgraafik koosneb valitud perioodis kõrvuti asetsevatest küüналde jadast [5]. Kuvades nt. viiepäeva graafikut, siis iga küüнал tähistab viiepäevast perioodi ja selle hinnamuutust (vt Joonis 2). Mida pikem on küüнал, seda suurem on hinnamuutus. Kuna küüналgraafikud on visuaalselt hästi kirjeldavad ja mustrite nimetused graafikul olevaga vastavuses (nt. küüналgraafiku muster „Hammer“ kujutab endast haamrit) on neid kauplejal kerge lugeda ja tuvastada. Küüналgraafikud on versatiilsed ja neid on võimalik käsitleda teistest kaupleva tööriistadest täiesti iseseisvalt [8].



Joonis 2. Swedbank (SWED-A.ST), 5-päeva küüналgraafiku näide [9].

2.2 Küüналgraafikute mustrid

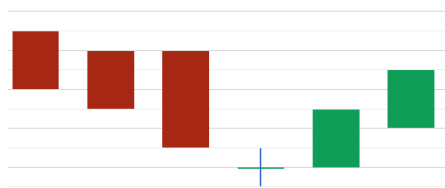
Küüналgraafikute mustrid jaotatakse enamasti kolme gruppi: tõusu (nn. „pulli-tüüpi“, ingl. k. „bullish“) ja languse (nn. „karu-tüüpi“, ingl. k. „bearish“) trendi mustrid [10]. Enamasti kasutatakse lühemaid perioode, ning nn. päeva-graafikut. Üldjuhul vajavad trendimuutust indikeerivad mustrid ka nn. järgmise-päeva-kinnitust [11]. See tähendab, et mustrit ei tasu lõpuni tehingu tegemiseks usaldada ning võtta esialgu, kuni kinnituseni, hoiatusena.

Taolisi mustreid on leitud 50-60 [12], kuid diplomitöö käigus käsitletakse kuut mustrit. Töö käigus implementeeritakse järgmised küünalgraafiku mustrid: Doji star, Dragonfly doji, Gravestone doji, Hammer, Hanging man doji, Spinning top.

Doji star

Muster võib esineda nii tõusu - kui ka langusperioodi ajal ja hoiatab potentsiaalse trendimuutuse eest. Viimane küünal on enne mustrit formuleerimist tavaliselt kõrgem. Antud mustrile on iseloomulik puuduv või minimaalne küünla keha ja selle tahti/ varju niipalju, et küünal näeks tähe moodi välja (vt Joonis 3). Madalaima ja kõrgeima hinna muutus peaks olema võimalikult võrdne.

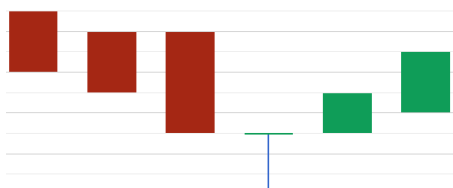
Kauplemisel tasub vaadata graafiku erinevaid perioode, et kindlaks teha turu suund [13]. “Doji star” on üldjuhul harva esinemissagedusega ja ei viita alati tegutsemisvajadusele, vaid on eelkõige tähelepanu äratamiseks potentsiaalse trendi muutuse eel [14].



Joonis 3. Doji star muster.

Dragonfly doji

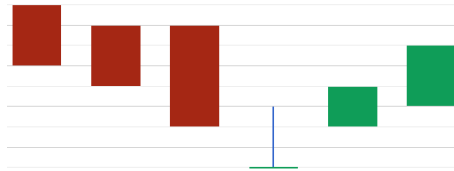
Sarnaselt “Doji star”-ile, võib esineda mõlemapoolse trendi lõpus, kuid usaldusväärsem on see langustrendil. Antud mustril on puuduv (või minimaalne) küünla keha. Ülemine vari on võrdeline kehaga või võimalikult lühike. Moodustab “T”-tähe [15] (vt Joonis 4). Nagu “doji”-tüüpi küünaltele omane, oodatakse ära järgmine küünal, et kinnitada trendimuutus. Esineb harva ja pigem usaldada kõrgemat tehingute volüümi antud perioodis, et muster trendimuutusele viitaks [16].



Joonis 4. Dragonfly doji muster.

Gravestone doji

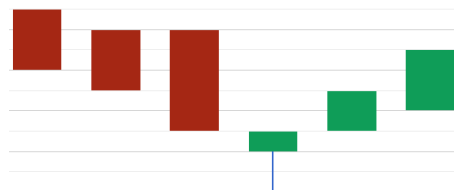
Meenutab “*Dragonfly doji*”-i vastupidist versiooni. Mustri moodustab tagurpidi “T”-täht (vt Joonis 5). Pikem ülemine vari, ning alumine vari võimalikult lühike või võrdeline puuduva kehaga [17]. Esineb langus- kui ka tõusutrendil, kuid tihedamini tõusutrendil ja vajab sarnaselt “*Dragonfly doji*”-le järgmist n-ö kinnitus-küünalt ja tugevat volüümi mustri perioodil [18].



Joonis 5. Gravestone doji muster.

Hammer

Langustrendil tõusule viitav muster. Meenutab kujult haamrit (vt Joonis 6). Küünla keha peaks olema väike, ning alumine vari peab olema vähemalt kaks korda suurem kui küünla keha. Küünlal on puuduv või võimalikult väike ülemine vari [19]. Antud muster on samade tingimuste korral usaldusväärsem kui „*Hanging man*“ ja ei vaja pikka eelnevat samasuunalist hinnaliikumist [20]. Mustri kinnituseks on järgnev küünal, mis peab lõpetama kõrgemal kui mustri sulgemishind [21].



Joonis 6. Hammer muster.

Hanging man

Sarnaneb välimuselt “*Hammer*”-ile, kuid tõusutrendil viitab langusele. Muus osas reeglid hinna-piiridega eelmise mustriiga samad [19]. Keha peaks jääma päevasisesel kauplemisel ülespoole, et seda kindlaks kindlaks mustriks lugeda võiks. Kui küünla avanemise ja kõrgeim hind on samad, siis on tegemist ühe tugevama languse signaaliga.

Mida pikem küünla alumine vari, seda tugevamat vastupanu näitavad nn. karud, kes müügiga hinda alla suruvad, millega ostjad sammu pidada ei suuda [22]. “*Hanging man*” vajab kinnitust, et trend muutumas on. Kinnituseks on järgmine küünal antud perioodil, mis peab olema hinnalt madalam, milleta on antud muster lihtsalt hoiatuseks [23].

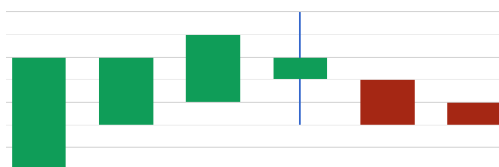


Joonis 7. Hanging man muster.

Spinning top

Võib esineda nii langus - kui ka tõusutrendil. Spinning top omab võrdlemisi väikest keha, kuid väljaulatuvad varjud võiksid olla sarnase pikkusega, ning kehast pikemad (vt Joonis 8). Mida pikemad on mõlemad varjud seda tugevamale hinna-võitlusele viitavad [24].

Sarnaneb kujult “*Doji star*”-le, kuid on suurema kehaga. Mõlemad mustrid viitavad turu “võrdsusele” antud perioodis. Lisainformatsiooni saamiseks on soovitatav ka teisi indikaatoreid uurida [14]. Üksinda võivad viidata trendi jätkumisele, kus üks pool ei kaalu veel teist üle. Antud mustrit esineb tihti, mis tähendab, et tuleks vaadata ka muid tehnilise analüüsi indikaatoreid, et kindlaks teha turu liikumise soov [25].



Joonis 8. Spinning top muster.

3 Sarnased, olemasolevad lahendused

Turult võib leida tooteid ja teenuseid, mis pakuvad küünalmustrite tuvastust graafikul, kuid leitud lahendustel on puudusi. Järgnevalt toob autor välja mõned potentsiaalsed, enamlevinud platvormid, nende positiivsed ja negatiivsed küljed.

Konkreetselt küünalgraafikute mustreid otsiva rakendusena saab välja tuua CandleScanner'i (vt. Tabel 1).

Tabel 1. CandleScanner plussid-miinused [26].

Plussid	Miinused
Lai valik andmeallikaid tasulises versioonis.	Piiratud valik andmeallikaid tasuta versioonis. Peamiselt päeva-lõpu seisuga
Konfigureeritavad küünla osad, mustrite tuvastamise tingimused	Tasuline. Eluaegse litsentsiga pakutakse ainult üks aasta programmi edasisi uuendusi
Iseseisev rakendus	Ainult Windows operatsioonisüsteemi rakendusena

Kauplejate seas populaarsemaid kauplemisplatvorme MetaTrader (vt Tabel 2).

Tabel 2. MetaTrader plussid-miinused [27].

Plussid	Miinused
Levinud platvorm, palju kasutajaid	Küünalmustrite tuvastamine eraldi pluginana. Ei ole platvormi sisseehitatud osa.
Kõrge funktsionaalsusega. Pluginate arendamise võimekus	Veebiversioonil ja mobiilirakendustel puudub pluginate tugi
Windows, Linux, MacOS tugi. Lisaks, Android ja iOS tugi	Tasuta versioonis kaubeldakse virtuaalrahaga. Kauplemiseks mitte-virtuaalse rahaga on vajalik täisversioon, mis on tasuline.

Hea klienditoe ja pika kogemusega kauplemisplatvorm ProRealTime (vt Tabel 3).

Tabel 3. ProRealTime plussid-miinused [28].

Plussid	Miinused
Hea turgude ja vahendite valik	Turuandmed maksavad eraldi
Windows, Linux, MacOS tugi	Tasuta versioonis päevalõpu graafiku andmed
Kõrge funktsionaalsusega	Küünalgraafikute mustrid ja selle osad ei ole konfigureeritavad

Vabavaralistest lahendustest võib veel välja tuua järgmised teegid:

- [SpiralDevelopment/candlestick-patterns](#) [29] – seotud konkreetse andmete formaadiga. Saadaval ainult terminalirakendusena.
- [hackingthemarkets/candlestick-screener](#) [30] – seotud konkreetse andmete formaadiga, andmeallika, ning turu liigiga.
- [cm45t3r/candlestick](#) [31] – seotud konkreetse andmete formaadiga. Saadaval ainult terminalirakendusena.
- [erwin-beckers/CandleLight](#) [32] – seotud konkreetse kauplemisplatvormiga (*MetaTrader* plugin).
- [bjlbernal/candlestick_patterns](#) [33] – pPuudub kasutajaliides, ning sisendi-väljundi haldus.

Väljatoodud teekide peamiseks miinuseks on seotus konkreetsete vahendite, andmete formaadi ja struktuuriga. Lisaks on need seotud kindla programmeerimiskeelega, millele ei ole juurde ehitatud kasutajaliidest. See annab küll paindlikkuse kasutajaliides juurde ehitada, kuid piirab meid olemasoleva koodibaasiga, mille haldamine võib potentsiaalselt olla hoopis ajakulukam ja piiravam tuleviku arenduste suhtes. Sellest tulenevalt on autor otsustanud luua rakenduse ilma ülaltoodud programmideta.

4 Lähteülesanne ja rakenduse teostus

Ülesanne on luua ühe-leheline veebirakendus, mis eeldefineeritud mustreid kasutaja sisestatud andmestikust üles leiab ja graafikul visualiseerida suudab.

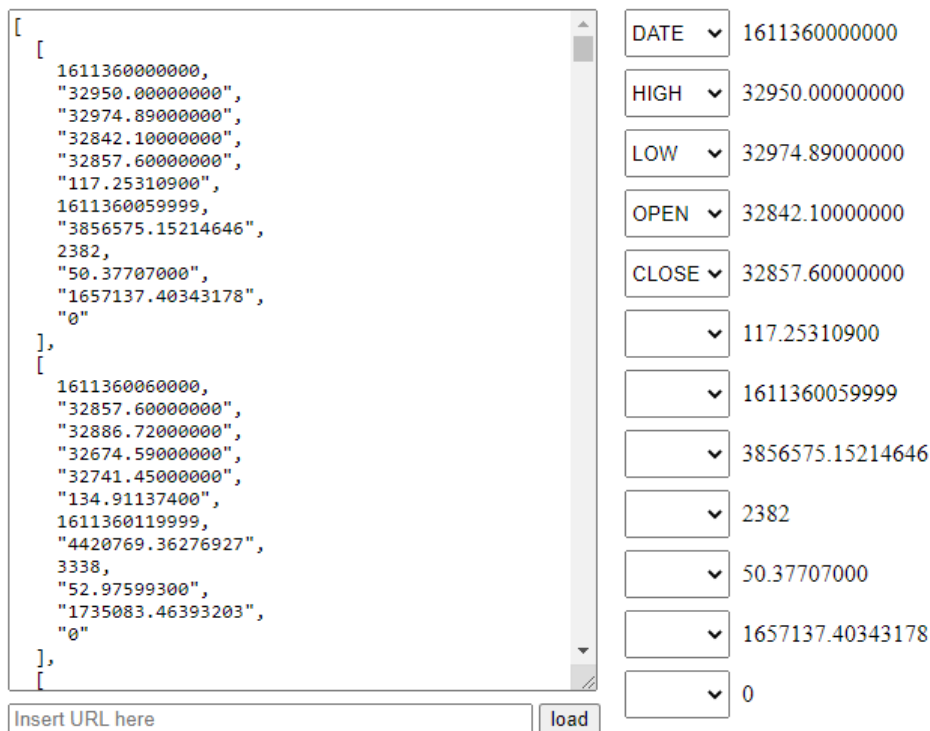
Tingimused:

- Toetada sisendina küünalgraafiku andmeid JSON (JavaScript Object Notation) formaadis:
 - vabatekstina;
 - päringu vastusena URL (Uniform Resource Locator) kaudu.
- Visualiseerida etteantud andmed küünalgraafikuna.
- Kuvada graafikul kohti ajas, mis viitavad, tuvastatud küünalgraafiku mustri järgi potentsiaalsele trendimuutusele.
- Defineerida kuus küünalgraafiku mustrit.
- Võimalus tulemust pildina salvestada ja jagada:
 - link pildi allalaadimiseks;
 - nupp pildi objekti kopeerimiseks lõikelauale.
- Arendada programmi põhi selliselt, et kasutataks minimaalselt erinevaid tehnoloogiaid, et mitte piirata rakenduse paindlikust edasiarenduseks.

4.1 Sisend

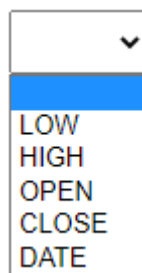
Sisendiks on JSON-tüüpi küünalgraafiku andmestik, rakenduses on seda võimalik ette anda kahel viisil. Üheks võimaluseks on sisestada andmestik JSON-kujul HTML (HyperText Markup Language) *textarea* vormielementi. Teine võimalus on sisestada URL *input*-tüüpi HTML elementi. Vastavale nupule vajutades tehakse antud URL-i (andmestiku teenusepakkuja) vastu päring ja saadakse vastus. Selleks on küünalgraafiku JSON-kuju, mis kirjutatakse korrektselt vormindatult esialgsesse *textarea*-tüüpi HTML elemendi väärtuseks.

Peale graafiku andmete sisestamist ilmub massiiv-tüüpi andmestiku puhul HTML *select* element, iga massiivis oleva künla osa kohta (vt Joonis 9).



Joonis 9. Andmete sisestamise näide.

Kasutajale kuvatakse esimese leitud künla väljade väärtused. Antud liideses tuleb kasutajal viia väli vastavusse künla osaga (vt Joonis 10), et graafiku generaator teaks mis väljaga on andmestikus tegu.



Joonis 10. Künla osade valik.

Väljad, mis ei ole künla osad tuleb jätta valimata (rippmenüüst esimene, vaikimisi valik). Rakenduse korrapäraseks töötamiseks tuleb ära kirjeldada kõik künla osad. Seda tuleb teha vaid esimese leitud künla kohta, sest andmestikus künla väljade asukohad/järjekord ei muutu.

Pärast andmete sisestamist on võimalik kasutajal vajutada „Add mapping“ nupule (HTML „button“-element) ning salvestab antud struktuuri brauseri vahemällu. Näiteks, JSON massiiv-tüüpi andmestiku puhul salvestatakse künla osade järjekord antud struktuuris.

Kui andmestikule on brauseri vahemälus juba olemasolev JSON künla struktuuri kirjeldus olemas, siis on võimalik see valida rippmenüüst. Nii ei pea uut struktuuri iga kord andmestiku sisestamisel kirjeldama, vaid saab olemasolevat ära kasutada.

Kui andmestikuga ja struktuuri kirjeldamise/ valimise toimingud on tehtud saab kasutaja vajutada „Show chart“ nupule, et kuvada küünalgraafik leitud mustritega.

4.2 Väljund

UI (User Interface) pool on minimaalse disainiga. Kuvatud on ainult rakenduse tööks vajalikud elemendid. Peale „Show chart“ nupule vajutust kuvatakse kasutajale standardne küünalgraafik, erisusega, et x-teljel jookseb tavapärase kuupäeva rea asemel nimistu leitud küünalmustritega. Autor leidis, et selline vaade on kasutajale arusaadavam, sest eesmärk on just mustreid otsida, ning selline pilt annab parima ülevaate. Küünalgraafikul saab märgitud iga mustri viimane küünal (või ainukene, kui muster koosneb ühest küünlast). Lisaks, küünlale hiirega peale minnes kuvatakse vihjemullina infot konkreetse küünla osade kohta s.h. kõik leitud mustrid küünlal (vt Joonis 11).



Joonis 11. Andmete kuvamise näide.

Kuna tulemus on graafiline, siis on autor otsustanud kasutada tulemuse jagamise funktsionaalsuse tagamiseks kahte erinevat viisi. Google Charts teek pakub funktsiooni nimega *getImageURI* [34], mis annab tulemuse graafiku *Base64* enkodeeritud pildi *URI* (Uniform Resource Identifier) kujul. Järgnevalt tutvustab autor lähemalt kuidas antud *URI*'ga luuakse tulemuse salvestamise ja kopeerimise võimekus.

Allalaadimislingi tegemiseks tuleb luua HTML'i nn. ankrulement, millele lisatakse atribuut nimega *download* ja viide antud *Base64* andmete *URI*'le [35]. See võimaldab lingile klikates graafik, koos leitud mustritega, arvutisse salvestada.

Clipboard'i manipuleerimise toe pakub *Clipboard API* (Application Programming Interface) [36], mis hetkel on veel nn. „töötava mustandi“ (ingl. k *working draft*) staatuses [37], kuid pakub juba rakendusele vajalikku funktsionaalsust. Rakendus kasutab *Base64* pildiandmete *URI*, millest eraldab *Base64* osa, dekodeerib selle, ning tekitab vajaliku *Blob*-tüüpi [38] objekti, *URI*'st saadud pildi tüübi järgi. Antud objekt on *Clipboard API write* funktsiooni sisendiks, mis võimaldab pildi salvestada otse lõikelauale. Niimoodi on võimalik rakenduse tulemust lihtsasti edasi kopeerida teistesse programmidesse.

4.3 Andmestik

Andmestik, mida rakendus mustrite tuvastamiseks kasutab, võib olla mitmes erinevas formaadis *JSON*'i struktuur. Andmestik võib pärineda andmepakkujatelt, kes küünalgraafikute andmeid üle *REST* (Representational state transfer) teenuse kättesaadavaks teevad. Andmepakkujateks võivad olla ka kauplemisplatvormid ise.

Võimalik künla JSON-kuju struktuur krüptovaluuta platvormi Binance API näitel (vt Joonis 12):

```
[
  1611362820000,
  "32450.52000000",
  "32484.53000000",
  "32436.26000000",
  "32470.32000000",
  "42.65275800",
  1611362879999,
  "1384501.91745361",
  1057,
  "20.50098800",
  "665516.25057661",
  "0"
]
```

Joonis 12. Binance API künla näide JSON-struktuuris [39].

Coinbase API näide (vt Joonis 13) on sarnane eelnevaga, kuid kuvab andmeid vähem ja teises järjekorras.

```
[
  1618388220,
  64512.2,
  64548.33,
  64525,
  64545.52,
  2.0441104
]
```

Joonis 13. Coinbase API künla näide JSON-struktuuris [40].

Antud struktuuri tähendus on vastavalt järjekorras selline (vt Tabel 4):

Tabel 4. Binance ja Coinbase JSON künla tähendus.

Binance	Coinbase
Ajatempel	Ajatempel
Alghind	Madalaim hind
Suurim hind	Suurim hind
Madalaim hind	Alghind
Lõpphind	Lõpphind

Iga künal on massiiv-tüüpi objekt enda väärtustest, kindlas järjekorras. Näha on ka, et ajatempel (ingl. k. *timestamp*) on erinevas formaadis. Tegemist on „Unix epoch“ (edaspidi epoch) ajatempliga. See näitab sekundite arvu ajast 00:00:00 (UTC) 1. jaanuar 1970. Konventsiooni kohaselt on tegemist sekunditega, kuid on rakendusi mis defineerivad seda kui millisekundeid epoch-ajast. See on erisus millega rakenduse arendades arvestada tuleks, sest erinevad teenused pakuvad erinevat kuupäeva formaati.

Sellisel kujul andmeid jagavaid andmepakkujaid on veel, kuid ei hõlma kõiki võimalikke formaate. On tavapärase, et formaat võib olla ka JSON-objekti kujul. Üks suurimaid börsi API (Application Programming Interface)-sid [41] pakuvad Marketstack [42] (vt Joonis 14) ja Polygon [43] (vt Joonis 15) kasutavad variatsiooni JSON-objekti kujust, kus iga künla osa on eraldi “võti”.

```

"data": [
  {
    "open": 129.8,
    "high": 133.04,
    "low": 129.47,
    "close": 132.995,
    "volume": 106686703.0,
    "adj_high": 133.04,
    "adj_low": 129.47,
    "adj_close": 132.995,
    "adj_open": 129.8,
    "adj_volume": 106686703.0,
    "split_factor": 1.0,
    "symbol": "AAPL",
    "exchange": "XNAS",
    "date": "2021-04-09T00:00:00+0000"
  },
  [...]
]

```

Joonis 14. Marketstack JSON künla näide [44].

```

{
  "status": "OK",
  "from": "2020-10-14T00:00:00.000Z",
  "symbol": "AAPL",
  "open": 324.66,
  "high": 326.2,
  "low": 322.3,
  "close": 325.12,
  "volume": 26122646,
  "afterHours": 322.1,
  "preMarket": 324.5
}

```

Joonis 15. Polygon JSON künla näide [45].

Kolmas, vähempopulaarsem variant on massiiv ajatemplitest, mille väärtuseks on künla osad. Näiteks kasutab taolist kirjapilti AlphaVantage [46] (vt Joonis 16).

```

"2020-11-09": {
  "1. open": "117.8800",
  "2. high": "119.7400",
  "3. low": "115.2700",
  "4. close": "115.5300",
  "5. volume": "8992152"
}

```

Joonis 16. AlphaVantage JSON künla näide [47].

Kuna ei ole mõistlik luua tuge kõikidele võimalikele andmekujudele, tuleb skoopi piirata ja valida enam-levinud formaadid. Diplomitöö käigus realiseeritakse variant, kus andmestik on massiiv künlaosade massiividest, et demonstreerida apliksiooni

rakendatavust. Hilisema edasiarendusena valmiks ka ülejäänud kaks formaati, mis kataks enamuse vajaminevatest andmete JSON struktuuridest.

4.4 Tehnoloogiate valik

Et luua operatsioonisüsteemist sõltumatut lahendust, mis oleks kasutajatele kergesti kättesaadav, on mõistlik see arendada veebiaplikatsioonina. Selleks kasutab autor webpack-dev-server'it [48], mis omakorda sisemiselt kasutab Express'i [49] veebiraamistikku. Tegemist on mahult kerge teegiga, mis võimaldab kiiresti veebiserveri üles seada ja on piisavalt populaarne [50], [51] ja ajas lihvitud, et ei sea piiranguid ka projekti kasvades. Rakenduse sisu serveerib Webpack'i plugin „HtmlWebpackPlugin“.

Et mugavdada arendustööd, on autor konfigureerinud webpack-dev-server'i selliselt, et projektiga seotud failides muudatuste tegemisel laetakse ka veebileht automaatselt uuesti. Lisaks on konfigureeritud ka *Hot Module Replacement* (HMR), mis automaatselt asendab muudetud moodulid serveris. Tänu HMR'ile ei lähe kaotsi rakenduse olek, nt. vormi sisestuselementidesse kirjutatud väärtused.

Rakendus töötab Node.js käitussüsteemina, ning programmeerimiskeelena kasutab TypeScripti, tüübikindlat JavaScripti. Typescript'i koodi on tänu selle tüübikindlusele lihtne lugeda ja hallata, tekib vähem vigu. Lisaks on TypeScript täielikult tagasiühilduv JavaScriptiga, mis tähendab, et rakendust on võimalik installeerida keskkondadesse kus viimane toetatud on [52].

Kood on kirjutatud järgides objektorienteeritud programmeerimise (edaspidi OOP) põhimõtteid. Näitena: küünalmustrid on erinevad klassid, mis sisaldavad ainult endale vajalikku informatsiooni (nt mustri enda tuvastamise reeglid), ning neid mustreid haldab ühtselt (mustrid implementeerivad üldist liidest) eraldi klass. Visuaalse poole struktuuri paneb paika hüperteksti märgistuskeel (ingl. k. *HyperText Markup Language*, edaspidi HTML) ja küljendamise/ stiili annab kaskaadlaadistik (ingl. k. *Cascading Style Sheets*, edaspidi CSS).

Tulenevalt Node'i kasutamisest on võimalik valida mitme paketi halduri vahel. Taoline programm haldab teeke, mida rakendus kasutada saab. Selliselt saab arendaja ühest tsentraalsest kohast vajamineva kätte ja kindel olla, et ka teistel arendajatel/ rakenduse installeerimistel ka versiooniti kõik töötaks. Tuntuimad on *Node Package Manager*

(edaspidi NPM) ja Yarn [53]. Peamisi erinevusi nende vahel on pakettide installeerimise kiirus [54]. NPM installeerib mitut paketti ükshaaval, aga Yarn teeb seda võimalusel samaaegselt. Antud rakenduses ei ole pakettide installeerimise kiirus suureks argumendiks, sest rakendust hoitakse väliste sõltuvuste poolest minimaalsena ja kuna NPM on vaikumisi valitav ja lihtsasti kasutatav, suurema kasutajaskonnaga ja sellesttulenevalt rohkem silutud, seega sai NPM valitud paketihalduriks.

Rakenduses võib kasutada vabalt valitud HTTP (Hypertext Transfer Protocol) klientrakendust, sellest ei sõltu otseselt rakenduse funktsionaalsus. Autor kasutab HTTP-kliendina „Axios”t. Üks laialt kasutatavamaid, hea dokumentatsiooni, ning jõudlusega Node-keskkonnas kasutamiseks mõeldud teek [55]. Lisaks näitavad tihedad uuendused [56], et projekt on kaasaegne, ning vigade (*bugide*) ilmnemisel nendega tegeletakse. Projekt on suure kasutajabaasiga, mis annab enesekindluse, et aastatega on tarkvara hästi silutud. Välja võib veel tuua alternatiivi „node-fetch“ [57], mis kasutab Fetch API-t, kuid sellele puudub veel brauseri „Internet Explorer“ tugi (27.04.2021 seisuga) [58] – selle tarbeks tuleks teha antud brauserile eraldi tugi [59]. Populaarne ja lihtsasti kasutatav on ka teek nimega „got“ [60].

Cache'i võimekuse pakub brauseri enda „localStorage“. Tuntud ka nimega „Web Storage“, mis pakub võimalust salvestada infot brauseri n-ö vahemällu, mis oma olemuselt on sarnane „sessionStorage“-ga, kuid puudub aegumisaeg [61]. See tähendab, et „localStorage“-is olevad andmed jäävad sinna ka brauseriakna sulgemisel [62]. Erand on „Incognito“ privaatses brausimise režiim. Sellisel juhul puhastatakse ajalugu, s.h. „localStorage“, kui sulgetakse „Incognito“ akna viimane avatud leht.

Küünalgraafikute kuvamiseks on autor valinud „Google Charts“ teegi, mis sai valitud oma kasutuslihtsuse poolest, ning see on tasuta tarkvara. Arendatud Google'i enda poolt, seega autori silmis on tegemist töökindla projektiga. Võimalikke graafiku kuvamise kandidaatide oli veel, nt. „CanvasJS“ [63] ja „JSCharting“ [64], kuid tegemist on tasuliste programmidega. Ühe potentsiaalse teegina saab veel välja tuua „D3.js“-nimelise tööriista [65]. Mõeldud peamiselt andmeteadlastele, kuid funktsionaalsuse mõttes täidaks oma rolli ka antud diplomitöös küünalgraafiku kuvamisel. Sarnaselt „Google Charts“-ile on tegemist tasuta tarkvaraga, kuid ei osutunud väljavalituks, sest eeldab teegi sügavuti tundmist [66]. Lisaks, sisaldab funktsioone mida antud töös vaja ei lähe. Lahendust võib kaaluda, kui küünalgraafikud ei ole ainukene andmete kuvamise viis.

5 Tulemus

Valminud on diplomitöö käigus veebirakendus küünalmustrite tuvastamise funktsionaalsusega, ning tulemuse visualiseerimise põhi. Antud rakendus töötab ootuspäraselt autori poolt testitud brauserite Microsoft Edge, Mozilla Firefox ja Google Chrome viimaste versioonidega (25.04.2021 seisuga). Rakendust on valitud tehnoloogiate tõttu võimalik hõlpsasti erinevatesse keskkondadesse üles seada, ning laiendada.

Kasutaja saab sisestada graafiku andmestiku kokkulepitud formaadis, koos nõutud JSON-struktuuri kirjeldamise infoga. Andmestik võib olla käsitsi sisestatud või vastusena kolmanda osapoole (nt aktsiate kauplemisplatvormi vms) REST-teenuse kaudu. Selleks tehakse HTTP GET-päring vastavalt kasutaja sisestatud URL'ile. Tagastatakse JSON-formaadis tekst – küünalgraafiku andmestik, mida graafikul koos mustritega peale nupule vajutamist kuvatakse. Andmete uuesti sisestamisel või pärimisel kirjutatakse olemasolev graafik uue tulemusega üle.

Loodud on *cache*'i võimekus brauseri „localStorage“ kasutamise näol. Hetkel kasutatakse seda JSON-andmestikus küünla struktuuri kirjelduste salvestamiseks. Selliselt ei pea kasutaja iga andmestiku sisestamise järel JSON küünla struktuuri uuesti kirjeldama vaid saab selle rippmenüüst valida.

Tehtud on ka veahaldus erinevate toimingute tarbeks. Kasutajale kuvatakse hetkel programmi vead brauseri konsooli kaudu, kasutades „console.error“ funktsiooni võimekust. Edasiarenduse käigus on plaan implementeerida kasutajasõbralikum lahendus.

Saadud graafilist tulemust (pilti) on võimalik salvestada otse arvutisse või kopeerida lõikelauale. Lahendus katab kaks nõuet tulemuse lihtsaks jagamiseks ja salvestamiseks.

6 Edasiarendus

Autori idee on antud töö võtta aluseks edasiarenduseks, ning implementeerida ka järgnev:

Täiendavad andmestiku formaadid

Implementeerida lisaks kaks andmepakkujate peamist andmestiku formaati, mis katavad juba enamuse turul pakutavatest JSON-struktuuridest.

URL-i põhjal tehtud päringute salvestamine

Võimalik URL-päringuid brauseri vahemällu salvestada ja kasutajale valikud kuvada selliselt, et puuduks vajadus aadresse iga kord uuesti sisestada.

Kasutajaliidese disain

Hetkel on kasutusel minimaalne disain CSS elementide loogiliseks kuvamiseks. Lisada kasutajasõbralikum disain. Graafiku kuva teha vastavalt ekraani suurusele skaleeruvaks.

Kasutaja defineeritud küünalmustrid

Küünalmustrite tingimused ei ole rangelt defineeritud. Olenevalt kirjandusest võib leida erinevaid interpretatsioone. Luua võimalus kasutajal defineerida mustreid ja kirjeldada küünla osasid.

Võimalus valida andmestikust „korduv struktuuri osa“

Andmestikus on küünalde massiiv ja „korduv osa“ vihjab ühe küünla struktuurile. Andmestiku siseselt struktuur ei muutu. Seega võiks kasutajal olla võimalus defineerida ise andmestikus „korduv osa“, ja selle küünla osad kirjeldada. See annab paindlikus kasutajal kasutada ükskõik millist andmestiku struktuuri.

Veateadete kuvamine

Luua ühtne, häid tavaid [67] järgiv, veateadete kuvamise viis.

Iseuenev graafik URL-ist päringu tegemisel

URL-põhiseid päringuid tehes küsitakse API käest juba kindla perioodiga andmeid, nt. 4-tunni graafik. See tähendab, et iga nelja tunni tagant tekkib uus küünal. Kasutajal võiks olla lisavalik kui tihti graafikut automaatselt uuendatakse.

Notifikatsioonid

Kasutajat temale valitud viisil (nt email) teavitada mingi kindla mustri tekkimisest.

Veebimajutus

Majutada rakendus kättesaadavaks üldsusele.

7 Kokkuvõte

Antud diplomitöö idee tuli autori soovist automatiseerida küünalgraafikute andmestikust mustrite otsimine ja tulemuse visualiseerimine. Saadaval on sarnaseid lahendusi, kuid enamuse on seotud konkreetse kauplemisplatvormiga. Mis omakorda, olenevalt ettevõttest, lisapiirangud seab. Autor nägi võimalust luua konkreetsele probleemile üldkasutatav rakendus, et puuduksid piirangud finantsvahendi tüübile (valuuta, aktsia, krüptoraha, fondid, jpmis), operatsioonisüsteemile ja kauplemisplatvormile.

Analüüsi tulemusena selgus, et uurimuse all olnud populaarsemad andmepakkujad kasutavad küünalgraafiku sisu hoidmiseks JSON-formaati, kuid selle struktuur ei ole kuidagi standardiseeritud. Selle tarbeks implementeeris autor ühe enamlevinud andmestiku formaadi, ning tõi töös veel välja kaks struktuuri, millega rakendus arvestada võiks, et katta ülejäänud andmepakkujate formaat. See on ka edasiarenduse ideena välja toodud. Implementeeritud formaat on loodud selliselt, et kasutajal on võimalik seda korduvkasutada samat tüüpi andmestiku peal. Kasutajale on loodud ka liides andmestikus küünla JSON struktuuri osade ümberpaigutamiseks, et tagada paindlikus erinevate andmepakkujate osas.

Diplomitöö raames vaadeldavad kauplemisplatvormid olid peamiselt veebipõhised, kuid suure kasutajaskonnaga platvorme võis näha sõltuvana ka konkreetsest operatsioonisüsteemist, töölauarakenduse näol. Sellest tingituna valmis lahendus veebirakendusena. Nii on võimalik rakendust kasutada laiemal kasutajaskonnal. Konfigureeriti veebiserver, ning rakendust testiti edukalt populaarsemate brauserite viimaste versioonidega.

Ülaltoodu põhjal valmis püstitatud eesmärkidele vastav veebirakenduse prototüüp, mis tuvastab etteantud andmestiku põhjal defineeritud küünalgraafikute mustreid ja visualiseerib neid interaktiivse küünalgraafiku peal. Rakendust on võimalik kasutada sõltumata kauplemisplatvormist, finantsvahendi tüübist, ning operatsioonisüsteemist. Loodud lahendus ei sea arvuti võimsusele kõrgeid nõudmisi, ega piira kasutajat kindla turu valikuga. Välja on ka toodud mõtteid edasiarenduseks, et luua mugavam, veel üldkasutatavam, ning funktsioonide rikkam lahendus.

Kasutatud kirjandus

- [1] „Candlestick Chart Patterns History,” [Võrgumaterjal]. Available: <https://www.vskills.in/certification/tutorial/candlestick-chart-patterns-history>. [Kasutatud 10. aprill 2021].
- [2] „Japanese Candlesticks,” Traderslog, [Võrgumaterjal]. Available: <https://www.traderslog.com/japanese-candlesticks>. [Kasutatud 10. aprill 2021].
- [3] „Japanese Candlesticks,” Corporate Finance Institute, [Võrgumaterjal]. Available: <https://corporatefinanceinstitute.com/resources/knowledge/trading-investing/japanese-candlestick>. [Kasutatud 15. aprill 2021].
- [4] G. Scott, „Understanding Basic Candlestick Charts,” Investopedia, [Võrgumaterjal]. Available: <https://www.investopedia.com/trading/candlestick-charting-what-is-it>. [Kasutatud 13. aprill 2021].
- [5] Äripäev, „Investeerimise teejuht,” 2007. [Võrgumaterjal]. Available: https://www.nasdaqbaltic.com/files/tallinn/bors/koolitus/Investeerimise_teejuht.pdf. [Kasutatud 15. märts 2021].
- [6] „Introduction to Candlesticks,” [Võrgumaterjal]. Available: https://school.stockcharts.com/doku.php?id=chart_analysis:introduction_to_candlesticks. [Kasutatud 12. aprill 2021].
- [7] Äripäev, „Joograafik, tulograafik ja küünalgraafikud,” %1 *Investeerimise teejuht*, Tallinn, Tallinna Raamatutrükikoda, 2007, p. 175.
- [8] B. Imambaks, „Why are candlestick charts so popular today?,” Think Binary, [Võrgumaterjal]. Available: <https://think-binary.com/why-are-candlestick-charts-so-popular-today/>. [Kasutatud 25. aprill 2021].
- [9] „Yahoo Finance,” [Võrgumaterjal]. Available: <https://finance.yahoo.com/quote/SWED-A.ST?p=SWED-A.ST&.tsrc=fin-srch>. [Kasutatud 16. aprill 2021].
- [10] „16 candlestick patterns every trader should know,” 2. jaanuar 2020. [Võrgumaterjal]. Available: <https://www.ig.com/en/trading-strategies/16-candlestick-patterns-every-trader-should-know-180615>. [Kasutatud 15. aprill 2021].
- [11] P. C. Charting, „Candlestick reversal signals need confirmation,” [Võrgumaterjal]. Available: <https://www.profitablecandlestickcharting.com/confirmation>. [Kasutatud 19. aprill 2021].
- [12] „Major Candlestick Signals,” Profitable Candlestick Charting, [Võrgumaterjal]. Available: <https://www.profitablecandlestickcharting.com/candlestick-signals>. [Kasutatud 19. aprill 2021].
- [13] „Doji Star Candlestick Pattern: Complete Guide,” PatternsWizard, [Võrgumaterjal]. Available: <https://patternswizard.com/doji-star-candlestick-pattern/>. [Kasutatud 18. aprill 2021].
- [14] J. Chen, „Doji Definition,” Investopedia, [Võrgumaterjal]. Available: <https://www.investopedia.com/terms/d/doji.asp>. [Kasutatud 17. aprill 2021].

- [15] „Dragonfly Doji Candlestick Pattern: Full Guide,“ PatternsWizard, [Võrgumaterjal]. Available: <https://patternswizard.com/dragonfly-doji-candlestick-pattern/>. [Kasutatud 19. aprill 2021].
- [16] C. Mitchell, „Dragonfly Doji Candlestick Definition and Tactics,“ Investopedia, [Võrgumaterjal]. Available: <https://www.investopedia.com/terms/d/dragonfly-doji.asp>. [Kasutatud 20. aprill 2021].
- [17] „Gravestone Doji Candlestick Pattern: Full Guide,“ PatternsWizard, [Võrgumaterjal]. Available: <https://patternswizard.com/gravestone-doji-candlestick-pattern/>. [Kasutatud 18. aprill 2021].
- [18] J. Chen, „Gravestone Doji Definition,“ Investopedia, [Võrgumaterjal]. Available: <https://www.investopedia.com/terms/g/gravestone-doji.asp>. [Kasutatud 20. aprill 2021].
- [19] PatternsWizard, „Hammer Candlestick Pattern: Complete Guide,“ [Võrgumaterjal]. Available: <https://patternswizard.com/hammer-candlestick-pattern/>. [Kasutatud 9. aprill 2021].
- [20] TheLiveTradeRoom, „Candlestick patterns – Umbrella Lines,“ [Võrgumaterjal]. Available: <https://thelivetraderoom.com/candlestick-patterns-umbrella-lines/>. [Kasutatud 20. aprill 2021].
- [21] C. Mitchell, „Hammer Candlestick,“ Investopedia, [Võrgumaterjal]. Available: <https://www.investopedia.com/terms/h/hammer.asp>. [Kasutatud 19. aprill 2021].
- [22] PatternsWizard, „Hanging Man Candlestick Pattern: Trading Guide,“ [Võrgumaterjal]. Available: <https://patternswizard.com/hanging-man-candlestick-pattern/>. [Kasutatud 20. aprill 2021].
- [23] C. Mitchell, „Hanging Man Candlestick Definition and Tactics,“ Investopedia, [Võrgumaterjal]. Available: <https://www.investopedia.com/terms/h/hangingman.asp>. [Kasutatud 20. aprill 2021].
- [24] „Spinning Top Candlestick Pattern: What is it?,“ PatternWizard, [Võrgumaterjal]. Available: <https://patternswizard.com/spinning-top-candlestick-pattern/>. [Kasutatud 20. aprill 2021].
- [25] C. Mitchell, „Spinning Top Candlestick,“ Investopedia, [Võrgumaterjal]. Available: <https://www.investopedia.com/terms/s/spinning-top.asp>. [Kasutatud 21. aprill 2021].
- [26] „CandleScanner,“ [Võrgumaterjal]. Available: <https://www.candlescanner.com/products/candlescanner/>. [Kasutatud 25. märts 2021].
- [27] „MetaTrader 5 Trading Platform,“ [Võrgumaterjal]. Available: <https://www.metatrader5.com/en/trading-platform>. [Kasutatud 25. märts 2021].
- [28] „ProRealTime,“ [Võrgumaterjal]. Available: <https://www.prorealtime.com/>. [Kasutatud 24. märts 2021].
- [29] SpiralDevelopment, „candlestick-patterns,“ [Võrgumaterjal]. Available: <https://github.com/SpiralDevelopment/candlestick-patterns>. [Kasutatud 15. märts 2021].
- [30] hackingthemarkets, „candlestick-screener,“ [Võrgumaterjal]. Available: <https://github.com/hackingthemarkets/candlestick-screener>. [Kasutatud 15. märts 2021].

- [31] „Candlestick - Candlestick patterns detection,“ [Võrgumaterjal]. Available: <https://github.com/cm45t3r/candlestick>. [Kasutatud 15. märts 2021].
- [32] E. Beckers, „CandleLight,“ [Võrgumaterjal]. Available: <https://github.com/erwin-beckers/CandleLight>. [Kasutatud 15. märts 2021].
- [33] B. Bernal, „candlestick_patterns,“ [Võrgumaterjal]. Available: https://github.com/bjlbernal/candlestick_patterns. [Kasutatud 15. märts 2021].
- [34] G. Charts, „Printing PNG Charts,“ [Võrgumaterjal]. Available: <https://developers.google.com/chart/interactive/docs/printing>. [Kasutatud 18. aprill 2021].
- [35] [Võrgumaterjal]. Available: https://www.w3schools.com/tags/att_a_download.asp. [Kasutatud 20. mai 2021].
- [36] [Võrgumaterjal]. Available: https://developer.mozilla.org/en-US/docs/Web/API/Clipboard_API. [Kasutatud 20. mai 2021].
- [37] [Võrgumaterjal]. Available: <https://w3c.github.io/clipboard-apis/>. [Kasutatud 20. mai 2021].
- [38] [Võrgumaterjal]. Available: <https://developer.mozilla.org/en-US/docs/Web/API/Blob>. [Kasutatud 20. mai 2021].
- [39] „Market Data Endpoints,“ Binance, [Võrgumaterjal]. Available: <https://binance-docs.github.io/apidocs/>. [Kasutatud 20. märts 2021].
- [40] „Data Endpoints,“ Coinbase, [Võrgumaterjal]. Available: <https://developers.coinbase.com/api/v2>. [Kasutatud 20. märts 2021].
- [41] „Top Market Data API Providers (2020-2021),“ [Võrgumaterjal]. Available: <https://digital.hbs.edu/platform-digit/submission/top-stock-market-api-providers-a-guide-for-investors/>. [Kasutatud 1. aprill 2021].
- [42] „Marketstack API,“ [Võrgumaterjal]. Available: <https://marketstack.com>. [Kasutatud 10. märts 2021].
- [43] „Polygon API,“ [Võrgumaterjal]. Available: <https://polygon.io>. [Kasutatud 10. märts 2021].
- [44] „API Documentation,“ marketstack, [Võrgumaterjal]. Available: <https://marketstack.com/documentation>. [Kasutatud 21. märts 2021].
- [45] „Api Documentation,“ Polygon, [Võrgumaterjal]. Available: <https://polygon.io/docs>. [Kasutatud 21. märts 2021].
- [46] „Alpha Vantage API Documentation,“ [Võrgumaterjal]. Available: <https://www.alphavantage.co/documentation>. [Kasutatud 10. märts 2021].
- [47] „Alpha Vantage API Documentation,“ Alpga Vantage, [Võrgumaterjal]. Available: <https://www.alphavantage.co/documentation/>. [Kasutatud 15. aprill 2021].
- [48] „Webpack DevServer,“ [Võrgumaterjal]. Available: <https://webpack.js.org/configuration/dev-server>. [Kasutatud 1. aprill 2021].
- [49] „Express.js web framework,“ [Võrgumaterjal]. Available: <https://expressjs.com>. [Kasutatud 15. märts 2021].
- [50] A. Goel, „10 Best Web Development Frameworks,“ [Võrgumaterjal]. Available: <https://hackr.io/blog/web-development-frameworks>. [Kasutatud 22. märts 2021].
- [51] „webpack dev-server,“ Openbase, [Võrgumaterjal]. Available: <https://openbase.com/js/webpack-dev-server>. [Kasutatud 25. märts 2021].

- [52] A. Sharma, „Why You Should Use TypeScript for Developing Web Applications,“ DZone, [Võrgumaterjal]. Available: <https://dzone.com/articles/what-is-typescript-and-why-use-it>. [Kasutatud 27. aprill 2021].
- [53] K. Khosravi, „JavaScript package managers compared: Yarn, npm, or pnpm?,“ [Võrgumaterjal]. Available: <https://blog.logrocket.com/javascript-package-managers-compared/>. [Kasutatud 16. märts 2021].
- [54] „Yarn vs npm – Best Package Manager 2021,“ [Võrgumaterjal]. Available: <https://www.positronx.io/yarn-vs-npm-best-package-manager>. [Kasutatud 26. märts 2021].
- [55] „14 Best Node.js HTTP Request Libraries,“ Openbase, [Võrgumaterjal]. Available: <https://openbase.com/categories/js/best-nodejs-http-request-libraries?orderBy=RECOMMENDED&>. [Kasutatud 5. aprill 2021].
- [56] [Võrgumaterjal]. Available: <https://openbase.com/js/axios/maintenance>. [Kasutatud 12. märts 2021].
- [57] „node-fetch,“ [Võrgumaterjal]. Available: <https://www.npmjs.com/package/node-fetch>. [Kasutatud 23. aprill 2021].
- [58] „Fetch API,“ MDN Web Docs, [Võrgumaterjal]. Available: https://developer.mozilla.org/en-US/docs/Web/API/Fetch_API. [Kasutatud 27. aprill 2021].
- [59] A. Bece, „How to polyfill JavaScript fetch function for Internet Explorer,“ DEV Community, [Võrgumaterjal]. Available: <https://dev.to/adrianbdesigns/how-to-polyfill-javascript-fetch-function-for-internet-explorer-g46>. [Kasutatud 27. aprill 2021].
- [60] „got,“ [Võrgumaterjal]. Available: <https://www.npmjs.com/package/got>. [Kasutatud 27. aprill 2021].
- [61] Mozilla, „Mozilla MDN Web Docs - Local Storage,“ [Võrgumaterjal]. Available: <https://developer.mozilla.org/en-US/docs/Web/API/Window/localStorage>. [Kasutatud 18. aprill 2021].
- [62] M. Burn, „Quora,“ [Võrgumaterjal]. Available: <https://www.quora.com/Is-localStorage-deleted-once-the-user-closes-the-browser-or-the-tab>. [Kasutatud 18. aprill 2021].
- [63] „CanvasJS,“ [Võrgumaterjal]. Available: <https://canvasjs.com>. [Kasutatud 15. märts 2021].
- [64] „JSCharting,“ [Võrgumaterjal]. Available: <https://jscharting.com/>. [Kasutatud 18. aprill 2021].
- [65] „D3.js,“ [Võrgumaterjal]. Available: <https://d3js.org>. [Kasutatud 11. aprill 2021].
- [66] K. Sumanth, „D3.js vs. Google Charts: A Data Scientist’s Review,“ [Võrgumaterjal]. Available: <https://humansofdata.atlan.com/2014/09/d3-js-versus-google-charts>. [Kasutatud 2. märts 2021].
- [67] Y. Deshpande, „Oops, something went wrong!,“ [Võrgumaterjal]. Available: <https://uxdesign.cc/oops-something-went-wrong-621efcc8319d>. [Kasutatud 18. aprill 2021].

Lisa 1 – Lihtlitsents lõputöö reprodutseerimiseks ja lõputöö üldsusele kättesaadavaks tegemiseks¹

Mina, Tarvo Tammejuur

1. Annan Tallinna Tehnikaülikoolile tasuta loa (lihtlitsentsi) enda loodud teose „Küünalmustrite tuvastamise ja tulemuse visualiseerimise rakenduse prototüübi arendus“, mille juhendaja on Kristiina Hakk
 - 1.1. reprodutseerimiseks lõputöö säilitamise ja elektroonse avaldamise eesmärgil, sh Tallinna Tehnikaülikooli raamatukogu digikogusse lisamise eesmärgil kuni autoriõiguse kehtivuse tähtaja lõppemiseni;
 - 1.2. üldsusele kättesaadavaks tegemiseks Tallinna Tehnikaülikooli veebikeskkonna kaudu, sealhulgas Tallinna Tehnikaülikooli raamatukogu digikogu kaudu kuni autoriõiguse kehtivuse tähtaja lõppemiseni.
2. Olen teadlik, et käesoleva lihtlitsentsi punktis 1 nimetatud õigused jäävad alles ka autorile.
3. Kinnitan, et lihtlitsentsi andmisega ei rikuta teiste isikute intellektuaalomandi ega isikuandmete kaitse seadusest ning muudest õigusaktidest tulenevaid õigusi.

17.05.2021

¹ Lihtlitsents ei kehti juurdepääsupiirangu kehtivuse ajal vastavalt üliõpilase taotlusele lõputööle juurdepääsupiirangu kehtestamiseks, mis on allkirjastatud teaduskonna dekaani poolt, välja arvatud ülikooli õigus lõputööd reprodutseerida üksnes säilitamise eesmärgil. Kui lõputöö on loonud kaks või enam isikut oma ühise loomingu tegevusega ning lõputöö kaas- või ühisautor(id) ei ole andnud lõputööd kaitsvale üliõpilasele kindlaksmääratud tähtajaks nõusolekut lõputöö reprodutseerimiseks ja avalikustamiseks vastavalt lihtlitsentsi punktidele 1.1. ja 1.2, siis lihtlitsents nimetatud tähtaja jooksul ei kehti.