

## **Datan visualisointi**

### **Monitorointi-ohjelmiston kattavuus visualisointiohjelmistona**

Matias Perämäki

Opinnäytetyö

Toukokuu 2019

Tietojenkäsittelyn ala

Tradenomi liiketalouden tutkinto-ohjelma

Tietojenkäsittely

|  |                                     |                                   |
|--|-------------------------------------|-----------------------------------|
| Tekijä(t)<br>Perämäki, Matias  | Julkaisun laji<br>Opinnäytetyö, AMK | Päivämäärä<br>Toukokuu 2019       |
|  | SIVUMÄÄRÄ<br>53                     | Julkaisun kieli<br>Suomi          |
|  |                                     | Verkojulkaisulupa<br>myönnetty: x |
| Työn nimi<br><b>Datan visualisointi</b><br>Monitorointi-ohjelmiston kattavuus visualisointiohjelmistona  |                                     |                                   |
| Tutkinto-ohjelma<br>Tietojenkäsittely  |                                     |                                   |
| Työn ohjaaja(t)<br>Immonen, Jarkko   |                                     |                                   |
| Toimeksiantaja(t)<br>Roima Intelligence  |                                     |                                   |
| <p>Tiivistelmä</p> <p>Datan määrä ja sen merkitys kasvaa jatkuvasti sekä kiihtyvästi ympärillämme. Nykyään käytännössä jokainen yritys tuottaa ja myöskin tarvitsee dataa toimiakseen kannattavasti. Yritysten tavat käsitellä dataa vaihtelevat kuitenkin merkittävästi. Datan analysoinnilla voidaan tehostaa yrityksen toimintaa, luoda uusia palveluita tai kehittää olemassa olevia palveluita. Data ei kuitenkaan itse itsessään tuo helposti tulkittavaa dataa, vaan jotta dataa voitaisiin analysoida, täytyy dataa visualisoida. Visualisoinnilla voidaan vaikuttaa erittäin paljon datan analysointiin ja sen tehokkuuteen. Visualisoinnin jälkeen data tulee vielä analysoida, mutta analysointia helpottaakseen data tulee visualisoida oikein, selkeästi ja tehokkaasti.</p> <p>Opinnäytetyön päätavoitteena oli selvittää toimeksiantajan Monitorointi-ohjelmiston kattavuus visualisointiohjelmistona. Lisäksi tavoitteena oli selvittää mitä tarkoittaa datan visualisointi ja miten visualisointi toteutetaan selkeästi, mutta samalla kuitenkin mahdollisimman tehokkaasti.</p> <p>Teoreettisessa viitekehyksessä paneuduttiin dataan, sen eri muotoihin ja datan ja tiedon väliseen suhteeseen. Teoriaosuudessa käsiteltiin pääasiassa datan visualisointia, sen eri muotoja, visualisointityyppejä ja visualisointiin liittyviä ominaisuuksia. Lopuksi käsiteltiin vielä datan visualisoinnin konkreettisia hyötyjä yritykselle ja sen liiketoiminnalle.</p> <p>Tuloksena saatiin kriteeristö, jonka avulla toimeksiantajan ohjelmistoa arvioitiin. Arvioinnista luotiin kehityssuunnitelma, jonka kehitysehdotusten pohjalta Monitorointi-ohjelmistoon toteutettiin toteutettavissa olleet kehitysehdotukset. Luotua kriteeristöä voidaan hyödyntää joko suoraan tai vähäisellä soveltamisella myös muiden ohjelmistotuotteiden visualisoinnin kattavuuden arviointiin.</p> |                                     |                                   |
| Avainsanat (asiasanat)<br>Data, visualisointi, Roima Intelligence, ohjelmisto, kvalitatiivinen tutkimus  |                                     |                                   |
| Muut tiedot (Salassa pidettävät liitteet)  |                                     |                                   |

|  |  |   |
|--|--|---|
| Author(s)<br>Perämäki, Matias  | Type of publication<br>Bachelor's thesis | Date<br>May 2019<br>Language of publication:<br>Finnish |
|  | Number of pages<br>53                    | Permission for web publication: x                       |
| Title of publication<br><b>Data visualization</b><br>Monitorointi software's coverage as visualization software  |  |   |
| Degree programme<br>Business Information Systems   |  |   |
| Supervisor(s)<br>Immonen, Jarkko   |  |   |
| Assigned by<br>Roima Intelligence  |  |   |
| Abstract<br><br><p>The amount and importance of data is constantly and exponentially growing around us. Today, practically every company produces and needs data to operate profitably. The way companies handle data varies significantly. Data analysis can enhance company's performance, create new services, or develop existing ones. However, the data itself does not easily produce data that can be interpreted, yet, data needs to be visualized for analysis. Visualization can greatly affect data analysis and its effectiveness. After visualization, the data should still be analyzed, however, in order to facilitate the analysis, the data should be visualized correctly, clearly and efficiently.</p> <p>The main goal of the thesis was to find out the coverage of assignor's Monitorointi software as visualization software. In addition, the goal was to find out what is meant by data visualization and how visualization is carried out clearly but at the same time as efficiently as possible.</p> <p>The theoretical framework focused on data, its different forms, and the relationship between data and knowledge. The theoretical part mainly focused on data visualization, its different form, visualization types and visualization related features. Finally, the concrete benefits of data visualization for the company and its business were further discussed.</p> <p>The result was the criteria by which the assignor's software was evaluated. From the evaluation, a development plan was created on the basis of which development proposals were implemented in the Monitoring software. The created criteria can be utilized, either directly or with minor application, to evaluate the visualization of other software products.</p> |  |   |
| Keywords/tags (subjects)<br>Data, visualization, Roima Intelligence, software  |  |   |
| Miscellaneous (Confidential information)   |  |   |

## Sisältö

|          |   |           |
|----------|---|-----------|
| <b>1</b> | <b>Johdanto</b> .....                                       | <b>4</b>  |
| <b>2</b> | <b>Tutkimusasetelma</b> .....                               | <b>5</b>  |
| 2.1      | Toimeksiantajan esittely .....                              | 5         |
| 2.2      | Tutkimuksen tavoitteet ja rajaukset .....                   | 6         |
| 2.3      | Tutkimusmenetelmät.....                                     | 6         |
| 2.4      | Tutkimuskysymykset.....                                     | 7         |
| <b>3</b> | <b>Data</b> .....   | <b>8</b>  |
| 3.1      | Data käsitteenä.....  | 8         |
| 3.2      | Data ja tieto .....   | 10        |
| 3.3      | Datatyypit .....  | 12        |
| <b>4</b> | <b>Datan visualisointi</b> .....                            | <b>13</b> |
| 4.1      | Visualisointi käsitteenä.....                               | 14        |
| 4.2      | Milloin data tulisi visualisoida tekstin sijaan? .....      | 14        |
| 4.3      | Selkeä ja tehokas visualisointi .....                       | 14        |
| 4.4      | Visualisointitavat.....                                     | 16        |
| 4.5      | Visualisointitapojen ominaisuudet .....                     | 26        |
| 4.6      | Visualisoitavan datan tyyppi.....                           | 27        |
| 4.7      | Datan visualisoinnin hyödyt .....                           | 29        |
| <b>5</b> | <b>Datan visualisointi Monitorointi-ohjelmistossa</b> ..... | <b>31</b> |
| 5.1      | Ohjelmiston esittely .....                                  | 31        |
| 5.2      | Visualisointiohjelmiston kriteeristö .....                  | 34        |
| 5.3      | Ohjelmiston vertailu .....                                  | 36        |
| 5.4      | Toimintasuunnitelma .....                                   | 38        |
| <b>6</b> | <b>Monitorointi-ohjelmiston kehittäminen</b> .....          | <b>40</b> |
| 6.1      | Lisättävät kaaviotyypit .....                               | 40        |
| 6.2      | Lisättävät ominaisuudet .....                               | 46        |
| <b>7</b> | <b>Pohdinta</b> .....                                       | <b>47</b> |
| 7.1      | Tutkimuksen luotettavuus.....                               | 50        |

|                                |    |
|--------------------------------|----|
| 7.2 Kehitysmahdollisuudet..... | 51 |
|--------------------------------|----|

|                      |           |
|----------------------|-----------|
| <b>Lähteet .....</b> | <b>52</b> |
|----------------------|-----------|

## **Kuviot**

|   |    |
|---|----|
| Kuvio 1. Datan määrä ajan suhteen (Salo 2013, 10).....                          | 9  |
| Kuvio 2. Datasta tietämykseen (Salo 2014, 33).....                              | 11 |
| Kuvio 3. Arvon kehitys datasta tietämykseen (Salo 2013, 27) .....               | 12 |
| Kuvio 4. Pylväskaavio .....   | 17 |
| Kuvio 5. Bullet-kaavio .....  | 17 |
| Kuvio 6. Mittari .....  | 18 |
| Kuvio 7. Viivakaavio .....  | 19 |
| Kuvio 8. Aluekaavio.....  | 19 |
| Kuvio 9. Hajontakaavio .....  | 20 |
| Kuvio 10. Kuplakaavio .....   | 21 |
| Kuvio 11. Piirakkakaavio.....   | 22 |
| Kuvio 12. Lämpökartta .....   | 22 |
| Kuvio 13. Puukartta.....  | 23 |
| Kuvio 14. Laatikko-janakaavio .....   | 24 |
| Kuvio 15. Vesiputouskaavio .....  | 25 |
| Kuvio 16. Tutkakaavio .....   | 25 |
| Kuvio 17. Ohjelmiston käyttöliittymästä otettu kuvakaappaus 1 .....             | 32 |
| Kuvio 18. Ohjelmiston käyttöliittymästä otettu kuvakaappaus 2 .....             | 32 |
| Kuvio 19. Ohjelmiston käyttöliittymästä otettu kuvakaappaus 3. ....             | 33 |
| Kuvio 20. Vesiputouskaavion luomiseen käytetty koodi omassa luokassaan. ....    | 41 |
| Kuvio 21. Kuplakaavion luontiin käytetty koodi omassa luokassaan. ....          | 42 |
| Kuvio 22. Kuplakaavion datan hakemiseen käytetty koodi. ....                    | 43 |
| Kuvio 23. Puukartan datan hakemiseen käytetty koodi. ....                       | 43 |
| Kuvio 24. Puukartan luomiseen käytetty koodi omassa luokassaan.....             | 44 |
| Kuvio 25. Tutkakaavion luontiin käytetty koodi omassa luokassaan. ....          | 45 |
| Kuvio 26. Toimintasuunnitelman perusteella lisätyt kaaviot käyttöliittymässä. . | 46 |

|  |    |
|--|----|
|  | 3  |
| Kuvio 27. Tietokerrosten luominen eräässä kaaviotyypissä.....            | 47 |
| Kuvio 28. Kaavion luonti toimintasuunnitelman toteuttamisen jälkeen..... | 47 |
| Kuvio 29. Vaihteluvälikaavio (pylväs) .....                              | 48 |

## **Taulukot**

|   |    |
|---|----|
| Taulukko 1. Kriteeristöä muodostettu taulukko.....  | 35 |
| Taulukko 2. Arvioitavan ohjelmiston visualisointimenetelmien kattavuus<br>kriteerien perusteella.....   | 37 |
| Taulukko 3. Arvioitavan ohjelmiston visualisoinnin ominaisuuksien kattavuus<br>kriteerien perusteella.....  | 38 |
| Taulukko 4. Arvioitavan ohjelmiston visualisointimenetelmien kattavuus<br>kriteerien perusteella toimintasuunnitelman toteutuksen jälkeen.....      | 48 |
| Taulukko 5. Arvioitavan ohjelmiston visualisoinnin ominaisuuksien kattavuus<br>kriteerien perusteella toimintasuunnitelman toteutuksen jälkeen..... | 49 |

# 1 Johdanto

”Data on kuin maaperässä piileskelevä malmi: pelkkä sen olemassaolo ja varastointi eivät tuota arvoa” (Markkula & Syväniemi 2015, 36).

Datan määrä ja merkitys kasvavat jatkuvasti kaikkialla ympärillämme. Käytännössä jokainen yritys tuottaa ja myöskin tarvitsee dataa toimiakseen ollenkaan, tai etenkin kannattavasti. Se, miten yritys hyödyntää dataa, vaihtelee huomattavasti erilaisten yritysten välillä. Datan analysoinnilla voidaan tehostaa yrityksen toimintaa, luoda uusia palveluja, kehittää palveluita ja paljon muuta, riippumatta yrityksen toimialasta.

Data ei itse suorita edellä mainittuja tehtäviä, vaan se täytyy ensin tuoda lukijalle nähtäväksi ja tulkittavaksi. Jotta dataa voitaisiin kuitenkin tulkita, pitää data ensin visualisoida. Datan esittämistavalla voidaan muuttaa täysin datan tulkitsemistapoja. Tämän vuoksi on olemassa hyvin paljon erilaisia datan visualisointimenetelmiä, kuten esimerkiksi monet erilaiset kaaviot, taulukot, mittarit, kartat ja kuviot.

Datan visualisointi ei kuitenkaan riitä informaation hyödyntämiseen. Tätä varten dataa tulee osata vielä analysoida oikein. Analysointi onkin tärkein vaihe datan hyödyntämisessä, sillä väärin analysoitu data antaa vääriä tuloksia, joista tehdyt johtopäätökset voivat olla hyvinkin arvaamattomia yrityksen liiketoiminnan, tai jopa tulevaisuuden kannalta. Analysointi ja siitä tehtävät johtopäätökset ovatkin täysin tulkitsejan vastuulla. Analysointi voikin olla erittäin haastavaa, etenkin jos visualisointi on toteutettu heikosti tai puutteellisesti. Tämän takia sekä visualisointi, että analysointi onkin toteutettava alan asiantuntijoiden toimesta, jotta vääriä analysoinneilta vältyttäisiin.

Tämän opinnäytetyön tarkoituksena on selvittää, mitä on data ja datan erilaiset ilmenemismuodot sekä datan erilaiset tyypit, mitä visualisointi selkokielellä tarkoittaa, miten sitä tulee esittää, jotta esitys olisi tehokasta mutta silti selkeää, sekä mitä konkreettisia hyötyjä yritys tehokkaasta visualisoinnista ja sitä kautta analysoinnista voi saada. Tämän lisäksi selvitetään, onko toimeksiantajan Monitorointi-ohjelmisto riittävän kattava visualisointiohjelmisto visualisointimenetelmien ja menetelmiin liit-

tyvien ominaisuuksien osalta käyttäen hyödyksi teoriaosuuden pohjalta tehtyä kriteeristöä. Kriteeristön avulla voidaan arvioida minkä tahansa ohjelmiston tai sovelluksen kattavuutta visualisointiohjelmistona.

Tutkimustuloksena saadaan kehityssuunnitelma, jossa esitetään kehitysehdotukset puuttuvien ominaisuuksien toteuttamiseen, sekä visualisointitapojen ominaisuuksien lisäämiselle ja muokkaamiselle. Tuloksissa myös kuvataan kehitysprosessi ja sen tulokset.

## 2 Tutkimusasetelma

Tässä luvussa esitellään toimeksiantaja sekä käsitellään tutkimuksessa käytettyjä tutkimusmenetelmiä, rajataan tutkittavaa asiaa ja esitellään tutkimuksen tutkimuskysymykset.

### 2.1 Toimeksiantajan esittely

Toimeksiantajana tälle tutkimustyölle toimii Roima Intelligence. Roima on Suomessa, Ruotsissa ja Alankomaissa toimiva ohjelmistokehitysyritys, joka valmistaa ohjelmistoja valmistavalle teollisuudelle. Roima on varsin uusi yritys, sillä se on perustettu vasta vuonna 2014. Siitä huolimatta Roiman juuret ovat syvällä historiassa, sillä siihen on yhdistynyt vuosien saatossa useita yrityksiä ja kartoituksia uusille yritysosoille suoritetaan jatkuvasti. Roimalla on tällä hetkellä toimistoja kahdeksalla eri paikkakunnalla, joista viisi sijaitsee Suomessa, kaksi Ruotsissa ja yksi Alankomaissa. Suomen toimipisteet sijaitsevat Espoossa, Porissa, Seinäjoella, Tampereella (2) ja Turussa. Ruotsissa paikkakunnat ovat Linköping ja Västerås. Alankomaissa toimisto sijaitsee Eindhovenissa. Toimipisteissä työskentelee jo yli 250 ohjelmistojen ja myynnin, sekä markkinoinnin asiantuntijaa ja määrä kasvaa jatkuvasti. Liikevaihto oli vuonna 2018 noin 34 miljoonaa euroa. Asiakkaita on yli 300. (Roima n.d.)

Roiman ydinajatuksena on teollisuuden kannattavuuden tukeminen Euroopassa tarjoamalla valmistavalle teollisuudelle ohjelmistoja, joiden avulla yritykset voivat kehittää ja tehostaa toimintaansa kannattavammaksi. Ohjelmistojen avulla asiakasta autetaan löytämään mahdolliset toimintaan negatiivisesti vaikuttavat pullonkaulat ja



muut aikaa vievät prosessit. Ohjelmistojen käyttöönoton myötä saadaan parempi kannattavuus ja tehokkuus. (Roima n.d.)

## 2.2 Tutkimuksen tavoitteet ja rajaukset

Tutkimuksen tarkoituksena on selvittää Roiman Monitorointi – ohjelmiston tämän hetkisen tilan riittävyttä datan visualisoinnin osalta. Löytyykö ohjelmistosta riittävän kattavan visualisointikeinot, vai tarvitaanko siihen lisäyksiä kaavioiden tai asetusten osalta? Lisäksi tarkoituksena on yleisesti ottaen selvittää, mitä on data ja minkälaisia erilaisia datatyyppejä on olemassa ja mitä tarkoittaa datan visualisointi, miten visualisointia toteutetaan tehokkaasti selkeydestä tinkimättä ja mitä konkreettista hyötyä datan visualisoinnista on yrityksille. Tavoitteena on myös luoda toimintasuunnitelma, jonka avulla ohjelmistoa jatkokehitetään. Toimintasuunnitelman perusteella ohjelmistoon toteutetaan havaitut puutteet osana tätä tutkimusta.

Visualisointitapoja on olemassa lukemattomia määriä. Kaikkien, tai edes puolien tapojen tutkimiseen menisikin käsittämätön määrä aikaa, joten tutkittavia tapoja on syytä rajata jollain tapaa. Tässä tutkimuksessa käsitellään vain tilastollisia graafeja, eli kaavioita. Näitäkin kaavioita arvioidaan olevan noin sata erilaista, joista vain pientä osaa käytetään säännöllisesti (Koponen & Hildén, 2019, 179), joten tässä tutkimuksessa perehdytään vain näihin tiettyihin, yleisimmin käytettyihin kaavioihin.

## 2.3 Tutkimusmenetelmät

Tässä opinnäytetyössä käytetään kvalitatiivista, eli laadullista tutkimusmenetelmää, jolla selvitetään datan ja datan visualisoinnin määritelmät ja erilaiset tyypit, sekä tutkittavan ohjelmiston riittävyys visualisointisovelluksena visualisoinnin ja siihen liittyvien ominaisuuksien osalta. Tutkimuksessa on myös kehitystutkimuksen piirteitä, sillä osana tutkimusta toteutetaan toimeksiantajan ohjelmistoon uusia ominaisuuksia, jotka pohjautuvat tähän tutkimukseen.

Datasta ja datan visualisoinnista, sekä analysoinnista löytyy valtava määrä erilaisia kirjoituksia, artikkeleita ja tieteellisiä julkaisuja. Tällöin, kun dataa on saatavilla suuria määriä erilaisista näkökulmista, tulee kvalitatiivinen, eli laadullinen tutkimus ajankoh-

taiseksi. Laadullisessa tutkimuksessa tutustutaan tutkittavan ilmiön teorioihin mahdollisimman laajalla näkökulmalla. Ilmiötä pyritään ymmärtämään mahdollisimman syvällisesti ilman määrällistä tutkimusta. Laadullinen tutkimus pureutuu yleensä vain yhteen asiaa käyttäen tutkimuksessa lauseita numeroiden sijasta. Laadullisessa tutkimuksessa ollaan usein kiinnostuneita prosesseista, sekä merkityksistä, kuinka ihminen tulkitsee tutkittavaa ilmiötä. (Kananen 2008, 26.)

Kvalitatiivista tutkimusmenetelmää hyödynnetään tietoperustan hankinnassa käyttäen useita sekä kirjallisia-, että internet-lähteitä. Lisäksi tutkimuksessa käytetään internet-kurssilla saatuja tietoja. Erilaisten lähteiden avulla muodostetaan selkeä ja laaja kuva datasta ja datan visualisoinnista. Näiden lähteiden avulla voidaan arvioida ohjelmiston kattavuus visualisointisovelluksena.

Tutkimuksen tuloksena saadaan selkeä ymmärrys datasta ja datan visualisoinnin eri menetelmistä sekä tarpeellisuudesta. Tutkimustuloksissa ensimmäisenä esitellään toimeksiantajan Monitorointi-ohjelmisto. Laajan teoriaosuuden pohjalta luodaan kriteeristö, jonka avulla arvioidaan visualisointimenetelmien ja menetelmiin liittyvien ominaisuuksien kattavuutta erilaisten visualisointitarpeiden kannalta. Tämän kriteeristön avulla arvioidaan toimeksiantajan Monitorointi-ohjelmiston kattavuutta visualisointiohjelmistona niin visualisointimenetelmien, kuin menetelmiin liittyvien ominaisuuksien osalta. Kriteeristöä voidaan hyödyntää, joko suoraan tai vähäisellä soveltamisella myös muiden ohjelmistotuotteiden visualisoinnin kattavuuden arviointiin. Arvioinnin perusteella luodaan toimintasuunnitelma, jossa kerrotaan ohjelmiston kehitysehdotukset. Lopuksi toimintasuunnitelma kehitysehdotukset toteutetaan ohjelmistoon siltä osin kuin se on toteutettavissa toimeksiantajan käyttämien teknologioiden puitteissa.

## 2.4 Tutkimuskysymykset

Tutkimuskysymykset, sekä niiden alikysymykset tässä tutkimuksessa ovat seuraavanlaiset:

- Mitä on datan visualisointi?
  - Mitä hyötyä datan visualisoinnista on?

- Miten esittää dataa tehokkaasti mutta selkeästi?
- Onko toimeksiantajan Monitorointi-ohjelmisto riittävän kattava visualisointiohjelmistona?

Ensimmäiseen tutkimuskysymykseen vastataan kirjallisuuden perusteella luvussa 4.1 ja sen ensimmäiseen alikysymykseen luvussa 4.5 ja toiseen alikysymykseen luvussa 4.3.

Tutkimuksessa selvitetään erilaisten visualisoitavan datan tyyppien parhaita visualisointitapoja. Arvioitavan visualisointisovelluksen olemassa olevia visualisointitapoja verrataan kirjallisuudessa mainittuihin tapoihin, joiden avulla tehdään johtopäätöksiä sovelluksen kattavuudesta ja luodaan mahdollinen toimintasuunnitelma sovelluksen kehittämiseksi, mikäli sovellus havaitaan puutteelliseksi. Tämän toimintasuunnitelman avulla vastataan toiseen tutkimuskysymykseen.

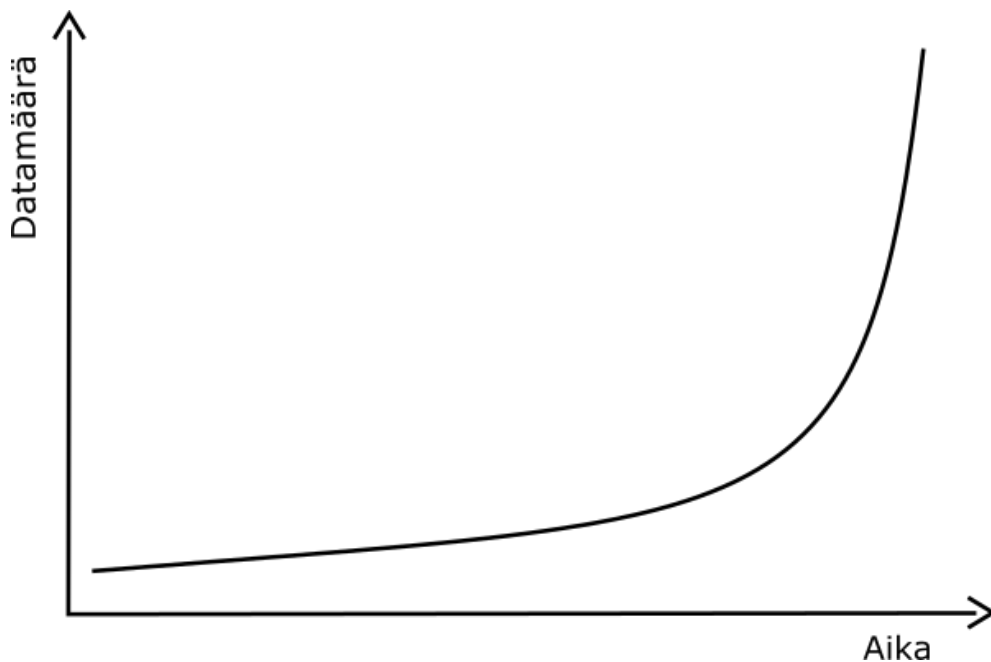
### **3 Data**

Tämän osion tarkoituksena on selvittää mitä data on, minkälaisia datatyyppisiä on olemassa, sekä mikä ero on datalla ja tiedolla ja mikä näiden käsitteiden välinen suhde on.

#### **3.1 Data käsitteenä**

Dataksi luokitellaan kaikki kirjallinen, digitaalinen, visuaalinen, sekä audiovisuaalinen materiaali. Historian saatossa datan luonne on muuttunut hyvinkin merkittävästi. Mitä syvemmälle historiaan halutaan mennä, sitä vähemmän dataa halutusta ajasta on enää saatavilla. Vain ihmiskunnan kannalta merkittäväksi koettu, sekä kestäväälle alustalle tuotettu data on enää saatavilla. Esimerkkeinä voidaan käyttää esimerkiksi luolamaalauksia ja tärkeät poliittiset ja tieteelliset julkaisut. Nykyään teknologia mahdollistaa yhä suuremman datamäärän turvallisen tallentamisen digitaalisessa muodossa ja tästä johtuenkin ihmiskunta tuottaa ja myös tallentaa yhä kiihtyvässä tahdissa haalimaansa dataa. Kuinka paljon dataa sitten on ja kuinka nopeasti datan määrä kasvaa? Siitä ei ole yksiselitteistä varmaa vastausta ja vaikka tulos voitaisiinkin esittää, olisi tulos jo julkaistaessa pahasti vanhentunutta tietoa. Varmaa onkin vain

se, että dataa on hyvin paljon ja datan määrä kasvaa hyvin nopeasti ja kiihtyvästi. Tästä tuleekin nimitys big data, joka on tämän vuosikymmenen yksi kuumimmista trenditermeistä. (Salo 2013, 10–11.)



Kuvio 1. Datan määrä ajan suhteen (Salo 2013, 10)

Vaikka ihminen onkin datan massiivisen kasvun aiheuttanut, ihminen itse ei tuota kaikkea muodostuvaa dataa, vaan uudet teknologiset ratkaisut, verkkoon kytketyt esineet (IoT, Internet of Things), tuottavat dataa myöskin kiihtyvässä tahdissa. Tällaisia esineitä ovat nykyään hyvinkin monenlaiset, jopa arkipäiväiset esineet, kuten älypuhelimet, älytelevisiot, jääkaapit, sääasemat, ajotietokoneet, valvontakamerat ja lukuisat muut laitteet. Niin ihmisen kuin esineiden tuottamat datat voidaan lajitella eri tavoin sen ominaisuuksien mukaan. (Salo 2013, 12–13.)

### **Staattinen ja virtaava data**

Data voidaan lajitella sen liikkuvuuden mukaan. Paikallaan pysyvään ja liikkuvaan dataan. Tietovarastossa oleva data on paikallaan pysyvää dataa. Data, joka on näennäisesti liikkeessä, on liikkuvaa dataa. Esimerkkinä paikallaan pysyvää dataa on yrityksen tietokannassa oleva data. Data niin sanotusti joudutaan louhimaan esiin. Liikkuvan datan esimerkkinä voidaan käyttää valvontakameran tuottamaa dataa. Kamera tuottaa jatkuvasti suuren määrän dataa, riippuen kameran resoluutiotasosta. Suuren tallennustilan vaatimisen takia tätä videomateriaalia ei välttämättä ole syytä

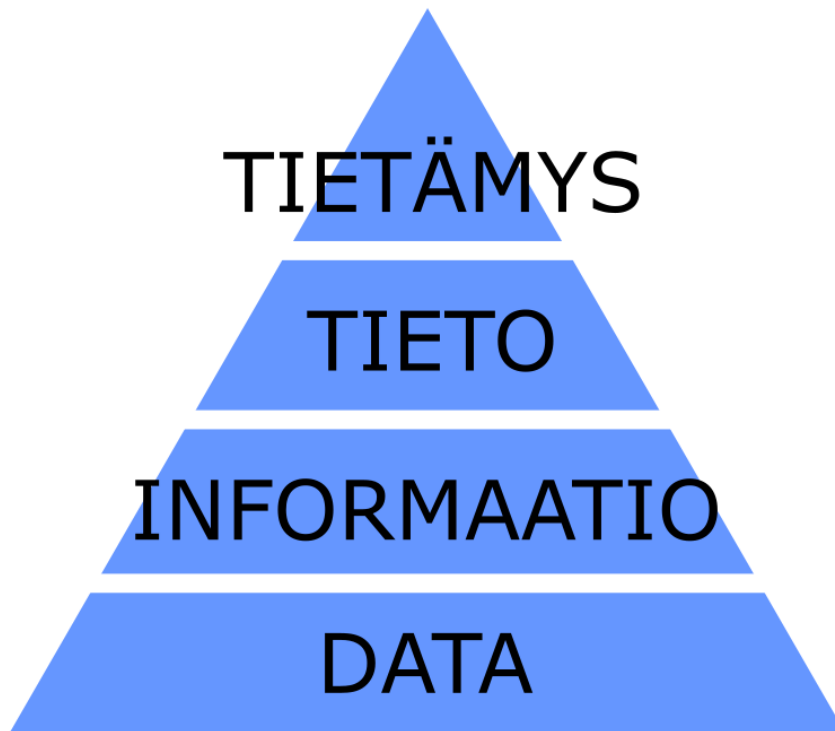
säilyttää kuitenkin pysyvästi, mutta tämä ei tarkoita sitä, etteikö dataa voisi kuitenkin hyödyntää tavalla tai toisella. (Salo 2013, 23–24.)

### **Strukturoitu ja strukturoimaton data**

Yksi tunnetuimmista tavoista on datan jakaminen sen rakenteen perusteella strukturoituun ja strukturoimattomaan dataan. Strukturoitu data on sellaista dataa, jolla on selkeä rakenne, kun taas strukturoimaton on päinvastoin rakenteetonta. Välimuodolle on myös annettu nimi semistrukturoitu data. Esimerkkinä strukturoimattomasta datasta voidaan käyttää edellisen kappaleen tavoin valvontakameran sisältöä. Videoon voidaan lisätä esimerkiksi avainsanoja, jotka tekevät videosta semistrukturoidun. Strukturoitua dataa puolestaan on esimerkiksi tietojärjestelmissä oleva data. (Salo 2013, 21–26.)

## **3.2 Data ja tieto**

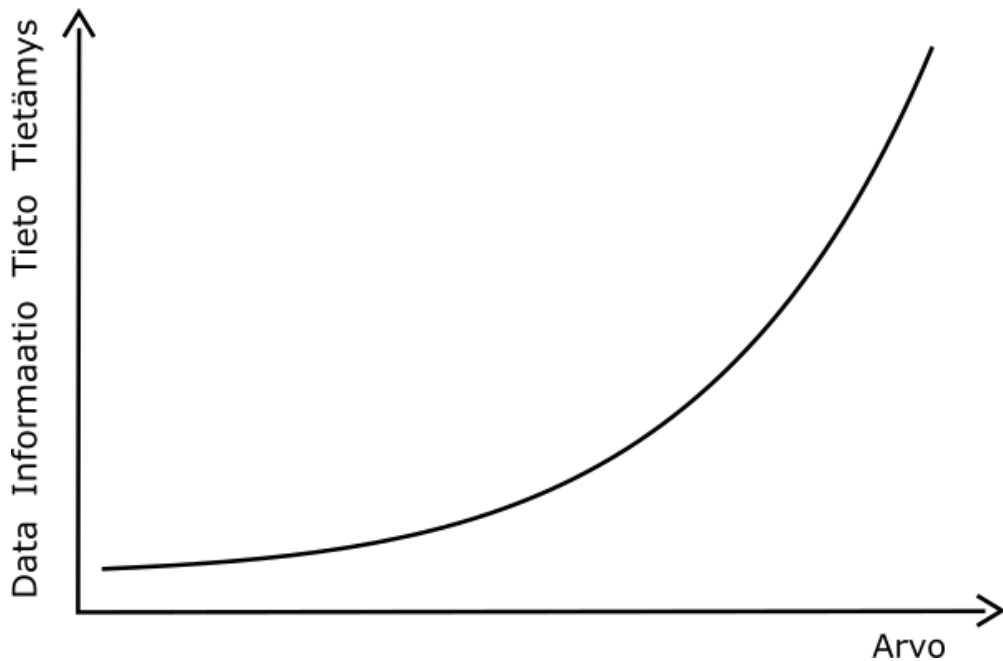
Arkikielessä moni mieltää sanat data ja tieto toistensa synonyymeiksi. Näin ei kuitenkaan ole täsmällisellä määrittelyllä. Datalla tarkoitetaan raaka-ainetta, josta muodostetaan informaatiota. Informaatiosta taas luodaan tietoa, joka jalostetaan tietämykseksi. Tieto ja tietämys määritelmiensä mukaisesti liittyy olennaisesti ihmiseen ja ihmisen ymmärrykseen, joten parhaimmillakaan algoritmeilla ei voida saada näitä osa-alueita tehtyä, ainakaan vielä. Edellä mainittujen käsitteiden suhdetta toisiinsa voisi mallintaa seuraavalla kuviolla:



Kuvio 2. Datasta tietämykseen (Salo 2014, 33)

Suhdeluku käsitteiden välillä ei ole kuitenkaan oheisen kaavion tavoin vakio tai suoraviivainen, mutta kaavio antaa kärjistetyn kuvan käsitteiden suhteesta. Pohjimmainen ajatus kaaviossa on kuitenkin se, että datasta jalostettu informaatio lisää tiedon määrää, mikä lähtökohtaisesti lisää myös tietämyksen määrää. Kuinka arvokasta tämä tietämys yritykselle on? Joskus se ei ole minkään arvoista, joskus taas tietämys voi olla elinehto yrityksen toiminnan kannalta. Huomioitavaa on myös se, että tietämyksen lisääntyminen ei ole itsestään selvyyttä ja myöskin se, että tiedosta voi tehdä väärä johtopäätöksiä, niin koneille luodut tekoälyt, kuin erehtyväiset ihmisetkin. (Salo 2014, 32–33.)

Toinen lähestymistapa edellä mainittuun asiaan on datasta saadun hyödyn vertaileminen. Data itsessään ei ole hyödyllistä, mutta datasta jalostettu tietämys voi olla erittäin arvokasta yrityksen toiminnan kannalta. Tästä suhteesta voidaankin luoda seuraavanlainen kaavio:



Kuvio 3. Arvon kehitys datasta tietämykseen (Salo 2013, 27)

Datan arvo siis kehittyi, mitä pidemmälle datasta saatua informaatiota jalostetaan. (Salo 2013, 26–28.)

### 3.3 Datatyypit

Dataa voidaan lajitella muutamiin erilaisiin datatyyppeihin. Näissä datatyypeissä vaihtelevat datatyypin kriittisyys yrityksen liiketoiminnan kannalta, sekä sen käyttömahdollisuudet erilaisilla osa-alueilla. (Väre 2019, 13.)

#### **Analyttinen data**

Analyttinen data tarkoittaa yrityksen varastoimasta datasta jalostettua lisädataa matematiikan ja tilastotieteen avulla. Tässä tapauksessa lisädataan pyritään saamaan uusia näkökulmia liittämällä jo olemassa olevaan dataan yrityksen ulkopuolelta tulevaa dataa. (Väre 2019, 15–16.)

#### **Master data**

Suomen kielelle käännettynä master data tarkoittaa avaintietoa. Nimensä mukaisesti avaintiedolla tarkoitetaan liiketoiminnan ytimessä olevaa dataa, eli dataa, joka on yrityksen liiketoiminnan kannalta kaikkein tärkeintä. Avaintiedon hallitseminen on kriittistä yrityksen jokapäiväisen liiketoiminnan kannalta. Avaintiedon voidaan kuvailla muodostavan datan perustukset ja kantavat rakenteet. (Väre 2019, 16.)

**Metadata**

Suomeksi metadata tarkoittaa määrittelydataa. Sillä hallinnoidaan sekä tiedostoja, että dokumentteja. Määrittelydatassa annetaan määritelmät ja/tai vaatimukset sen määrittelemälle datalle. Määrittelydatan määrittelemisen ja hallinnoiminen on avainasemassa avaintiedon laadun varmistamisessa. (Väre 2019, 16–17.)

**Referenssidata**

Referenssidatalla tarkoitetaan luokittelutietoa. Luokittelutiedolla ryhmitellään muuta yrityksen keräämää tai luomaa dataa. Luokittelutiedossa voidaan antaa myös ennalta määriteltyä dataa, jolla voidaan varmistaa muussa datassa käytettyjen datojen yhteneväisyys. Luokitteludata on kriittisessä roolissa luotettavan raportoinnin kannalta. (Väre 2019, 17.)

**Transaktiodata**

Transaktiodatalla tarkoitetaan tapahtumatietoa. Tapahtumatietoa kertyy yrityksen varsinaisesta liiketoiminnasta saadusta datasta. Toisin sanoen tapahtumatieto on dataa, joka kertyy yrityksessä yrityksen jokapäiväisestä toiminnasta. (Väre 2019, 17.)

**Big data**

Nopeasti lisääntyvää tapahtuma tietoa kutsutaan big data datatyypiksi. Big datalle tyypillistä on sen lähes hallitsematon muodostuminen, jolloin datan tallentaminen ei ole mahdollista tai järkevää. Big datalla tarkoitetaan usein myös yrityksen toiminnan ulkopuolella olevaa ja muodostuvaa dataa. (Väre 2019, 18.)

## 4 Datan visualisoiminen

Tämän osion tarkoituksena on selvittää mitä visualisointi käsitteenä tarkoittaa, sekä millainen on selkeä ja samalla tehokas visualisointi. Osuudessa käydään lävitse yleisimpiä kaaviotyyppisiä ja niiden vahvuuksia ja käyttötarkoituksia sekä selvitetään millä visualisointitavalla erityylisten datojen vertailua kannattaa visualisoida. Tämän osion avulla voidaan arvioida kattavasti visualisointisovelluksen riittävyttä sovelluksena visualisointimahdollisuuksien osalta. Tämä osio toimii myös pohjana sovellusta kehittävästä toimintasuunnitelman laatimisesta.



## 4.1 Visualisoiminen käsitteenä

Data itsessään on verrattavissa raaka-aineisiin; raaka-aine on arvokasta, mutta sellaisenaan se on hyvin harvoin käyttökelpoista. Raaka-aine tulee jalostaa tavalla tai toisella käytettävään muotoon, jotta sillä olisi oikeasti käytännön arvoa. (Nayak n.d.)

Datan visualisointi on graafinen esitys informaatiosta ja datasta. Käyttämällä visuaalisia elementtejä, kuten taulukoita, kaavioita ja karttoja, datan visualisoinnin työkalut tarjoavat helpon tavan nähdä ja ymmärtää trendejä, poikkeavia havaintoja ja kaavoja datassa. (Tableau n.d.)

## 4.2 Milloin data tulisi visualisoida tekstin sijaan?

Vaikka visualisointi onkin tehokas tapa välittää tietoa, se ei kuitenkaan ole järkevää kaikissa tilanteissa. Kultainen sääntö tähän on seuraava: jos jokin voidaan kuvata vähintään yhtä selkeästi sanojen avulla, visualisointi ei ole tarpeellista. Visualisoinnin tarkoituksena on helpottaa lukijan työtä hahmottaa dataa ja sen analysointia. Visualisointi on selvä silloin kun lukija ymmärtää kaavion sisällön ja sisällön avulla saa vastauksen kysymykseen tai luo uusia näkökulmia visualisoitavaan asiaan. Mikäli edellä mainittuja kriteerejä ei tapahdu, on visualisointi turhaa. Joskus visualisoinnista tulee sekava, mikä voi johtua usein väärästä visualisointivalinnasta. (Koponen & Hildén, 2019, 29–30.)

## 4.3 Selkeä ja tehokas visualisointi

Visualisointia luodessa on tärkeää ottaa huomioon erilaisia asioita, niin kohdeyleisöstä kuin itse visualisoinnista. Kohdeyleisöstä tulisi huomioida seuraavat asiat:

- henkilökohtaiset ominaisuudet (ikä, kulttuuri)
- asiantuntevuus
- käyttötapa.

Käyttäjän henkilökohtaiset ominaisuudet voivat vaikuttaa merkittävästi informaation analysointitapoihin ja -mahdollisuuksiin. Esimerkiksi fonttikoolla voi olla hyvin suuri

merkitys eri ikäryhmillä. Toisaalta taas kulttuurilliset erot voivat tuoda esiin ristiriitaisuuksia. Esimerkiksi lukusuunnan vaihtuvuus voi vaikuttaa merkittävästi tulkintaan. Visualisointitapaa valittaessa on myös hyvä huomioida tiedon vastaanottajan asiantuntevuus alalla. Asiantuntijat ja noviisit tarvitsevat erilaisen pohjustuksen aiheeseen. Lisäksi huomioitavaa on vielä se, että millä alustalla esittämistapaa tarkastellaan (mobiili, tietokone). Esimerkiksi valitun esittämistavan tulisi olla riittävän kokoinen helpon tulkinnan mahdollistamiseksi. Kun kohderyhmään liittyvät tekijät on selvitetty, tulisi valitun visualisointitavan omata vielä seuraavat ominaisuudet:

- ymmärrettävä (digestibility)
- selkeä (Clarity)
- luotettava (trust)
- tuttuus (familiarity)
- miellyttävä (delight).

Ymmärrettävällä kaaviolla tarkoitetaan käytännössä sitä, että esittämistavassa ei tule olla muuta kuin tarvittava informaatio, ei mitään ylimääräistä. Selkeys viittaa osittain samaan kuin ymmärrettävyys. Selkeydellä tarkoitetaan sitä, että informaatio tuodaan selkeästi esiin helppolukuisella esittämistavalla. Luotettavuus luodaan selkeällä esittämistavalla. Kun informaatio tuodaan esille selkeästi ja tehokkaasti, ei käyttäjällä ole tällöin syytä epäillä informaation oikeellisuutta. Tuttuus saadaan sisällytettyä esittämistapaan käyttämällä käyttäjälle ennestään tuttuja visualisointitapoja, esimerkiksi yksinkertaiset kaaviot. Miellyttävä esittämistapa on helppolukuinen, jolloin informaatiota on miellyttävä lukea ja analysoida. Kun esittämistapaan otetaan huomioon kaikki edellä mainitut ominaisuudet, saadaan luotua miellyttävä, selkeä ja tehokas visualisointitapa. (Nayak n.d.)

Datan visualisoinnilla voi olla kaksi erilaista tavoitetta. Visualisointi voi olla joko selittävä tai tutkiva. Selittävä visualisointi antaa selvän vastauksen haluttuun kysymykseen. Tällainen kysymys voisi olla esimerkiksi: mikä toimipiste tuotti suurimman liikevaihdon viime vuonna? Tässä tapauksessa visualisointiin haluttaisiin yrityksen jokai-

nen toimipiste ja jokaisen toimipisteen liikevaihto. Visualisoinnista nähtäisiin heti selkeästi suurimman liikevaihdon omaava toimipiste, eli se antaa suoraan vastauksen esitettyyn kysymykseen. Tutkiva visualisointi ei suoranaisesti vastaa tiettyyn kysymykseen, vaan visualisoinnista on tarkoituksena herätä kysymyksiä, joihin valittu visualisointitapa vastaa. Tällainen visualisointi voisi visualisoida esimerkiksi yrityksen toimipisteen liikevaihtoa koko toimipisteen elinkaaren ajalta. Tätä visualisointia tutkiessa lukijalle voi herätä uusia kysymyksiä: ”Onko myynti kehittynyt?”, ”Mikä kvartaali on ollut paras toimipisteen historiassa?”, ”Mikä kvartaali on ollut huonoin toimipisteen historiassa?”. Näihin kysymyksiin visualisoinnista löytyy vastaus. (Nayak n.d.)

#### 4.4 Visualisointitavat

Datan visualisoimiseen on lukemattomia erilaisia tapoja yksinkertaisista kaavioista ja taulukoista kolmiulotteisiin taideteoksiin sekä taulukoista kuplakaavioihin, joissa voi helposti esittää jopa neljää eri muuttujaa. (Nayak n.d.) Kaikkien esitystyyppien lävitsekäymisessä ei olisi mitään järkeä, joten tässä luvussa käydään vain yleisimmät esittämiseen käytetyt kaaviotyypit, tilastolliset kaaviot, ja niiden esittämisessä huomioon otettavat asiat.

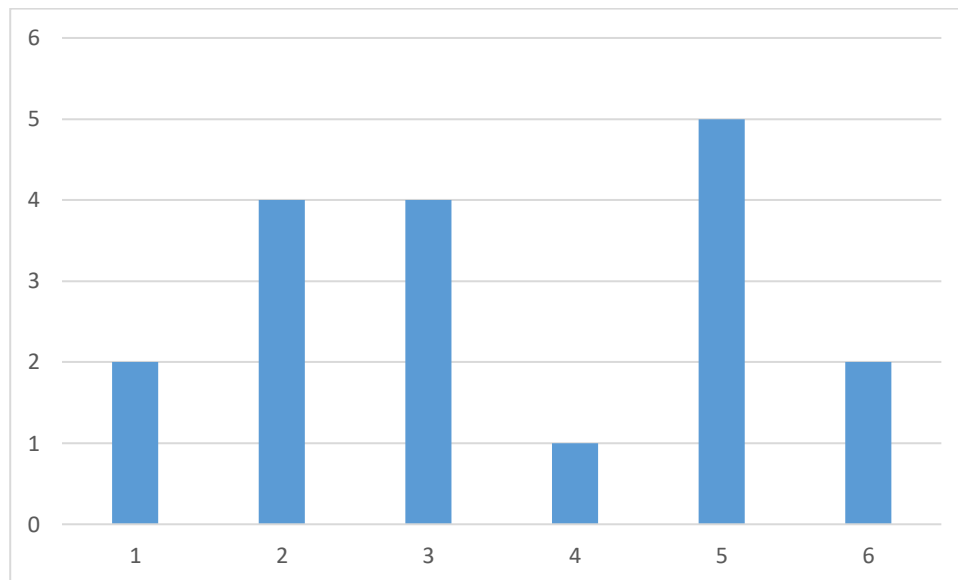
##### **Pylväskaavio (Bar Chart)**

Pylväskaavio on yksi yleisimmin käytetyistä kaaviotyypeistä. Pylväskaavion voi asettaa sekä vaaka-, että pystysuoraan. Pystysuorassa pylväskaaviossa käytetään usein x-akselilla kvantitatiivisia (numeerisia) arvoja (esimerkiksi vuosiluku), kun taas vaakasuorassa pylväskaaviossa kvalitatiivisia (laadullisia) (esimerkiksi kategoria) arvoja. Vaakasuoraa pylväskaaviota suositellaan kuitenkin enemmän. Tämä siksi, että vaakasuorassa pylväskaaviossa on enemmän tilaa kategorioiden teksteille ja allekkain olevia palkkeja voi lukea kuin listaa. Y-akselilla on kummassakin tapauksessa aina numeerinen arvo. (Koponen & Hildén 2019, 180.)

Pylväskaavio on yksinkertainen ja helposti ymmärrettävä, mutta samanaikaisesti erittäin tehokas, mitkä ovatkin kaavion vahvuuksia. (Nayak N.D.) Kaaviotyyppiä käytetään yleisimmin eri kategorioiden vertailuun, sijoituksiin, osuuden näyttämiseen sekä toissijaisesti datan muutokseen aikavälillä. (SAP n.d.)

Pylväskaaviosta on olemassa myös variaatioita, kuten pinottu pylväskaavio absoluuttisilla tai suhteellisilla arvoilla. Pylväskaaviossa voidaan helposti esittää myös useita kategorioita yhdessä kaaviossa eri värisillä tai sävyisillä pylväillä. (SAP n.d.) Pinotut pylväskaaviot ovat validi vaihtoehto piirakkakaavioille. (Koponen & Hildén 2019, 181.)

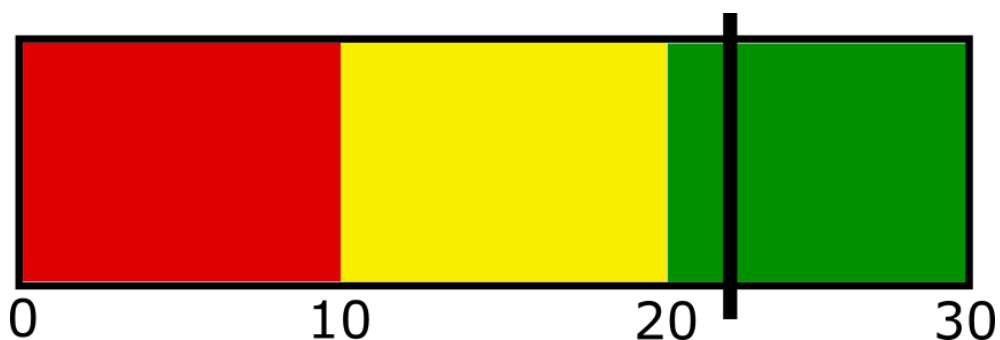
Kaaviota luodessa, tulisi aina kysyä itseltään, että voiko informaation visualisoida pylväskaaviolla, sillä sen tuttuus ja yksinkertaisuus tuottavat varmasti halutun vaikutuksen. (Nayak n.d.)



Kuvio 4. Pylväskaavio

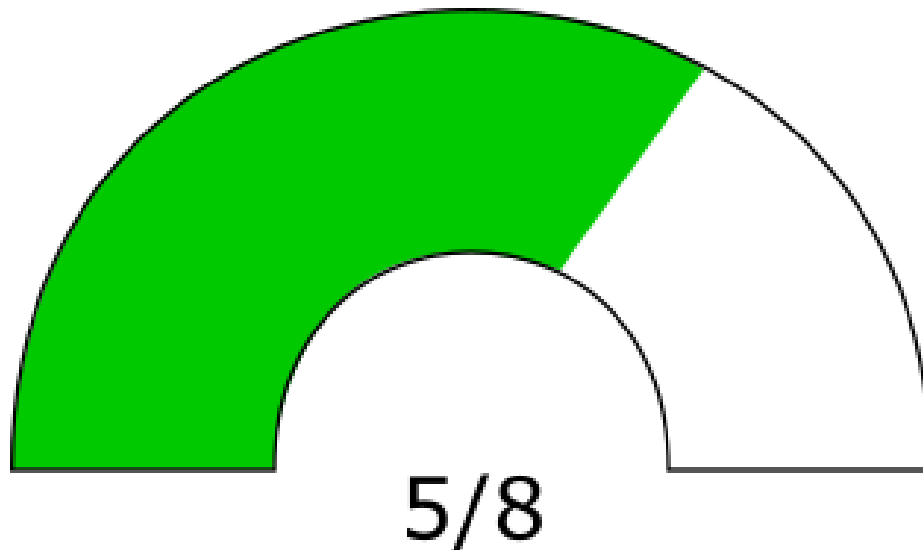
### Bullet-kaavio (Bullet Chart)

Bullet-kaavio on muunnelma perinteisestä pylväskaaviosta, jolla voidaan ilmaista halutun prosessin etenemistä määränpäähän. Kaaviolla voidaan esittää dataa, joka pohjautuu raja-arvoihin tai tavoitteisiin. (SAP n.d.)



Kuvio 5. Bullet-kaavio

Bullet-kaavio ajaa saman asian kuin erilaiset mittarit, mutta säästävät tilaa muulle informaatiolle, minkä takia bullet-kaavioiden käyttämistä suositellaan mittareiden sijasta. (SAP n.d.)

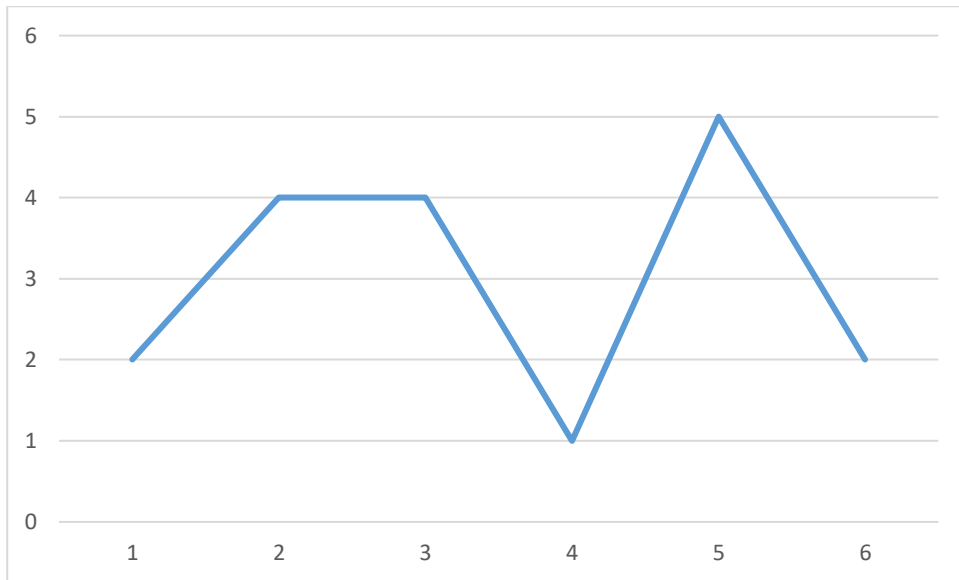


Kuvio 6. Mittari

#### **Viivakaavio (Line Chart)**

Viivakaavio on myös yksi yleisimmin käytetyistä kaaviotyypeistä. Viivakaaviota käytetään yleisimmin esittämään datan muutosta ajan funktiona, sillä viivakaaviosta on helposti havaittavissa muutoksen trendi. (Nayak n.d.)

Viivakaaviossa tulisi olla molemmilla akseleilla kvantitatiiviset (numeeriset) arvot. Yleensä havaintopisteet merkitään pilkuilla, mutta suuressa datajoukossa selkeyden vuoksi pilkkuja tulisi välttää. Viivakaavion käyttäminen onnistuu myös, mikäli datajoukon pisteet eivät jakaudu tasaisesti (esimerkiksi puuttuva havainto). Viivakaaviota ja pystysuoraa pylväskaaviota voidaan käyttää enimmäkseen samaan tarkoitukseen. Viivakaavio havainnollistaa paremmin jatkuvuutta, muutosta ja trendiä, kun taas pylväskaavio korostaa yksittäisiä datapisteitä ja vertailua näiden välillä. Lisäksi viivakaavio sallii huomattavasti suuremman datajoukon ilman kaavion sekavoitumista. (Koponen & Hildén 2019, 184–185.)



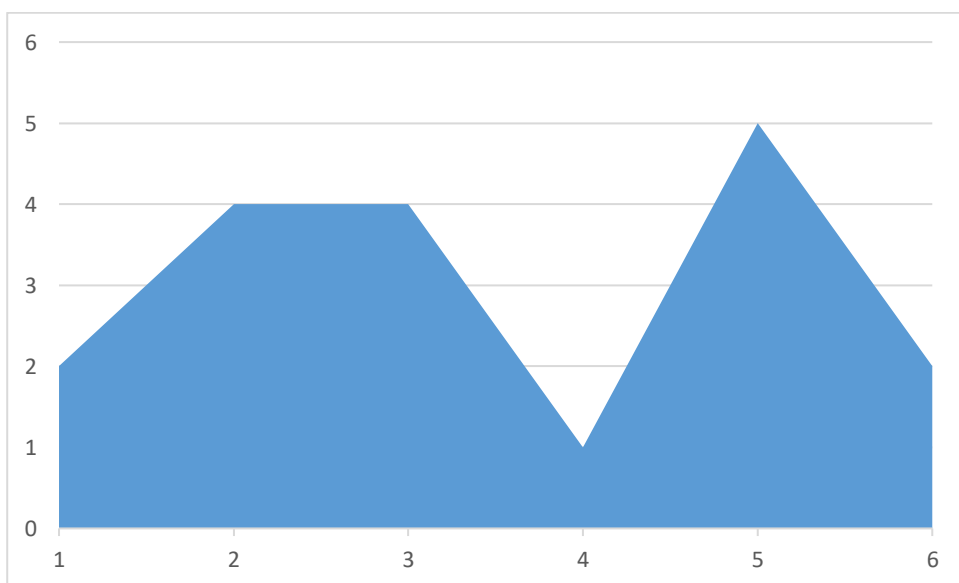
Kuvio 7. Viivakaavio

### Aluekaavio (Area Chart)

Aluekaavio on kuin viivakaavio, mutta viivan ja x-akselin välinen alue on täytetty.

Aluekaaviosta voi tehdä myös pinotun, joko absoluuttisella pinoamisella tai suhteellisella pinoamisella. Pinotussa aluekaaviossa huomioitavaa on se, että viivojen väliin jäävät alueet tulee olla täytetty, sillä täyttö on merkki lukijalle siitä, että kaavio on pinottu. (Koponen & Hildén 2019, 185.)

Aluekaavio on hyödyllinen silloin, kun halutaan visualisoida kumulatiivista arvoa, esimerkiksi liikevoittoa. (Nayak n.d.)

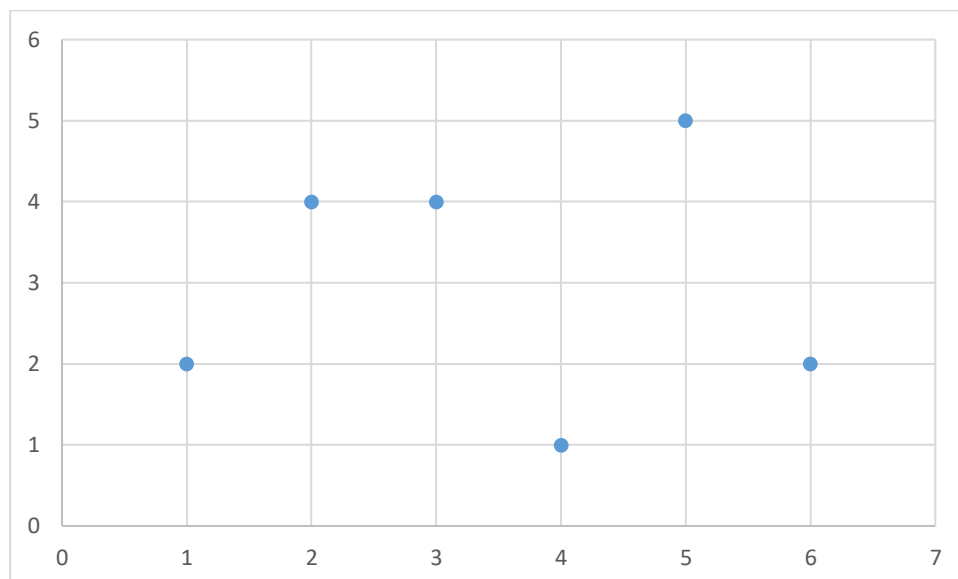


Kuvio 8. Aluekaavio

### Hajontakaavio (Scatter Chart)

Hajontakaavio on erinomainen kaaviotyyppi näyttämään muuttujien välistä korrelaatiota, eli riippuvuutta, sekä havainnollistamaan mahdollista muuttujien välistä kaavaa. Kaaviosta voi helposti havaita niin negatiiviset, kuin positiivisetkin korrelaatiot ja myöskin mahdolliset pisteistä muodostuvat rykelmät tai muista pisteistä selvästi erottuvat poikkeavat havainnot. (Nayak n.d.)

Mahdolliset rykelmät hajontakaaviossa kertoo mahdollisesti korrelaatiosta ja hajaantuneiden pisteiden välillä korrelaatiota ei puolestaan ole havaittavissa. Mikäli korrelaatiota epäillään, on tavanomaista asettaa x-akselille aiheutettu asia ja y-akselille syyn aiheuttaja. Hajontakaavioon asetetun trendiviivan avulla lukija voi helpommin arvioida korrelaatiota muuttujien välillä. Nouseva trendiviiva kertoo positiivisesta korrelaatiosta, kun taas laskeva kertoo negatiivisesta korrelaatiosta. (Koponen & Hildén 2019, 190–191.)

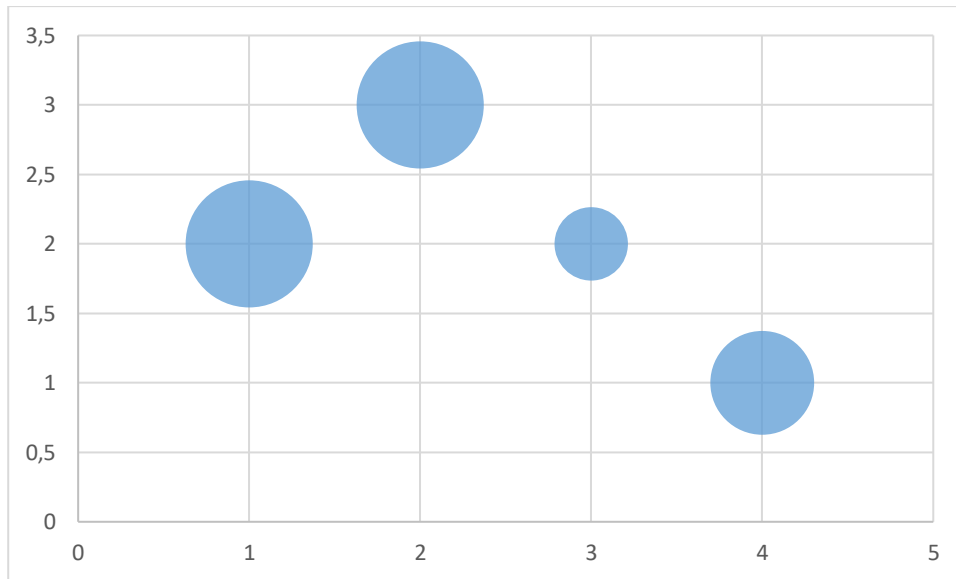


Kuvio 9. Hajontakaavio

### Kuplakaavio (Bubble Chart)

Kuplakaavio on verrattavissa hajontakaavioon, sillä sen esitystapa ja käyttötarkoitukset ovat hyvin samanlaiset. Kuplakaavio erottuu hajontakaaviosta siten, että kuplakaaviossa nimensä mukaisesti pisteiden sijaan esitetään kuplia, joihin voidaan lisätä kolmaskin muuttuja mukaan: pinta-ala. Tällöin pystytään selkeästi yhdellä kaaviolla esittämään kolmen tai neljän eri muuttujan kaavio, jos tuodaan pinta-alan lisäksi myös värit mukaan. (Nayak n.d.)

Kuplakaaviossa tulisi huomioida se, että kuplan sijainti on lukijalle helpompi tulkittava asia, kuin ympyrän pinta-ala, joten toissijainen tai tukeva informaatio tulisi asettaa kuplan pinta-alaksi. (Koponen & Hildén 2019, 193.)



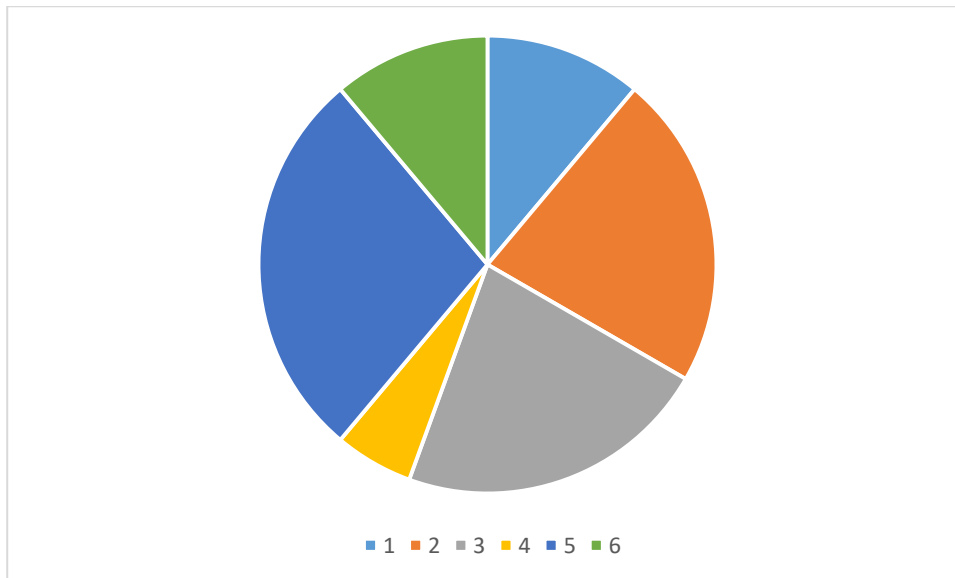
Kuvio 10. Kuplakaavio

### **Piirakkakaavio (Pie Chart)**

Ensimmäinen ja tärkein sääntö piirakkakaaviossa on se, että sen käyttämistä tulisi välttää, mikäli mahdollista. (SAP n.d.) Esimerkiksi pinottu pylväskaavio on usein huomattavasti parempi vaihtoehto piirakkakaavioille. Tietyissä tilanteissa piirakkakaavio voi olla kuitenkin hyödyllinen. Piirakkakaaviolla voidaan vertailla muuttujien osuuksien suuruutta suhteessa toisiinsa, mutta tarkkoja vertailuja ei voida ongelmitta suorittaa, sillä kulmien väliset erot ovat huomattavasti vaikeampia havaita kuin pituuden tai pinta-alan erot. (Koponen & Hildén 2019, 188)

Piirakkakaavion kanssa tulee toisaalta olla hyvin tarkka, sillä pienet erot eivät piirakkakaaviosta erotu helposti. Lisäksi piirakkakaavion käyttöä tulisi välttää, kun muuttujia on useita; yli viiden muuttujan piirakkakaavioita ei tulisi luoda, sillä usean muuttujan piirakkakaavio voi mennä hyvinkin sekavaksi johtuen pienistä piirakansiivuista. (Nayak n.d.)





Kuvio 11. Piirakkakaavio

**Lämpökartta (Heat map)**

Lämpökartassa pystytään helposti esittämään kolmen eri muuttujan muutokset.

Kartta muistuttaa perinteistä taulukkoa, mutta taulukkoon on sulautettu kolmas ulottuvuus mukaan värien muodossa, jolloin väri toimii myös muuttujan muodossa.

(Nayak n.d.)

Lämpökarttaa käytetään yleensä jakauman esittämiseen. Kartassa tulee ottaa huomioon värien erottuvuus; mikäli värien eroavaisuus on pientä, tulee lämpökartan käyttämistä harkita uudelleen. (SAP n.d.)

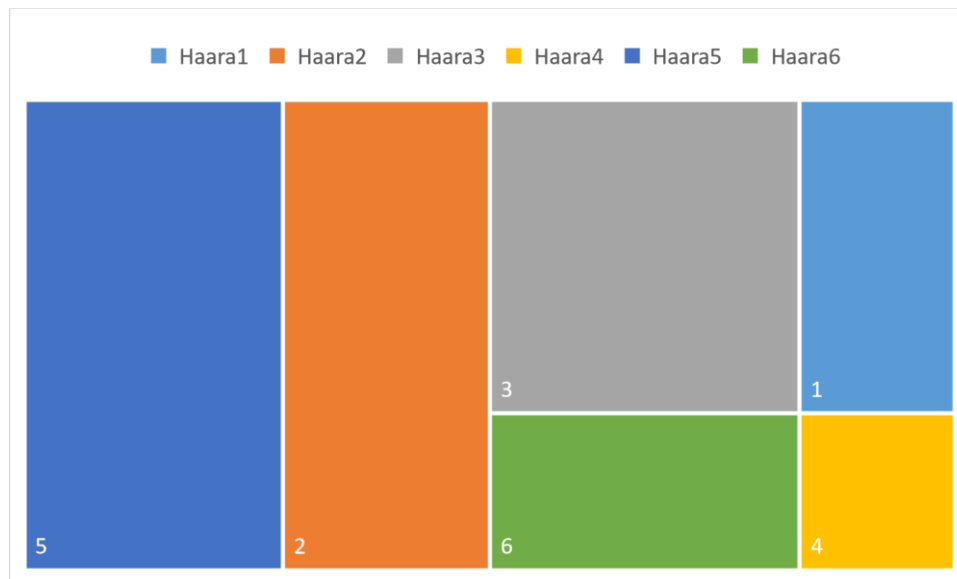
|    |    |    |            |
|----|----|----|------------|
| 2  | 15 | 50 | Muuttuja 1 |
| 15 | 31 | 0  | Muuttuja 2 |
| 12 | 24 | 45 | Muuttuja 3 |
| 1  | 2  | 3  |            |

Kuvio 12. Lämpökartta

**Puukartta (Treemap)**

Puukartta muistuttaa hieman lämpökarttaa niin ulkomuodoltaan, kuin käyttötarkoitukseltaan. Puukartassa ei kuitenkaan ole akseleita, vaan muuttujia on käsitelty värin ja koon muodossa. (Nayak n.d.)

Kuten lämpökartassakin, puukartassa tulee värien olla riittävän erilaiset selkeän kokonaisuuden luomiseksi. Värit eivät ole pakollisia tulkinnan saamiseksi, mutta ne helpottavat tulkinnan tekemistä. (SAP n.d.)



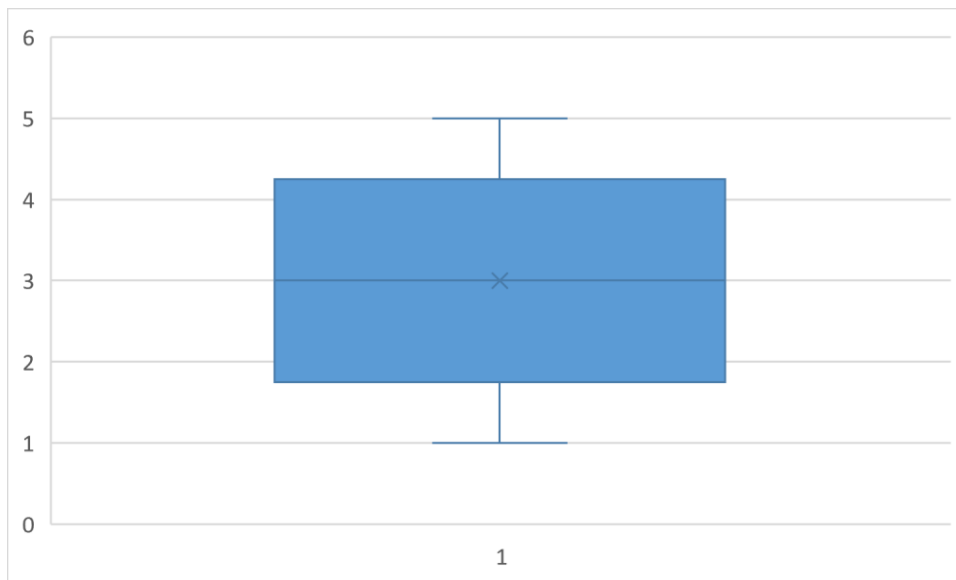
Kuvio 13. Puukartta

### Laatikko-janakaavio (Box plot)

Laatikko-janakaaviolla voidaan tehokkaasti esittää muuttujan arvojen jakautumista. Kaaviosta saadaan selville muuttujan mediaani, mediaanin molempien puolien lähimmät kvartaalit, sekä koko arvojoukon suurin sekä pienin arvo. Tällöin jo nopealla silmäyksellä on helppo selvittää koko arvojoukon arvojen jakautuminen, sekä arvojen tiheys eri kvartaaleilla. (Nayak n.d.)

Laatikko-janakaaviolla voidaan lisäksi esittää mahdolliset poikkeavat havainnot erillisillä pisteillä kaavion ulkopuolella. (SAP n.d.)

Laatikko-janakaaviosta on olemassa useita eri variaatioita. Tämä mahdollistaa kattavan kaaviopankin laatikko-janakaavion osalta, mutta toisaalta luo myös ongelman erilaisten variaatioiden tulkinnalle. Suuri variaation määrä aiheuttaa sen, että kaaviota voidaan tulkita virheellisesti aiheuttaen vääriä johtopäätöksiä, mikäli käytetty kaavio ei ole lukijalle tuttu. Tukeyn laatikko-janakaavio on kuitenkin muodostunut laatikko-janakaavion perinteisimmäksi versioksi (kuvio 14). Laatikko-janakaavio on melko epätavallinen kaaviotyyppi tieteellisten julkaisujen ulkopuolella, joten kaaviota tulisi käyttää harkiten. (Koponen & Hildén 2019, 196–197.)

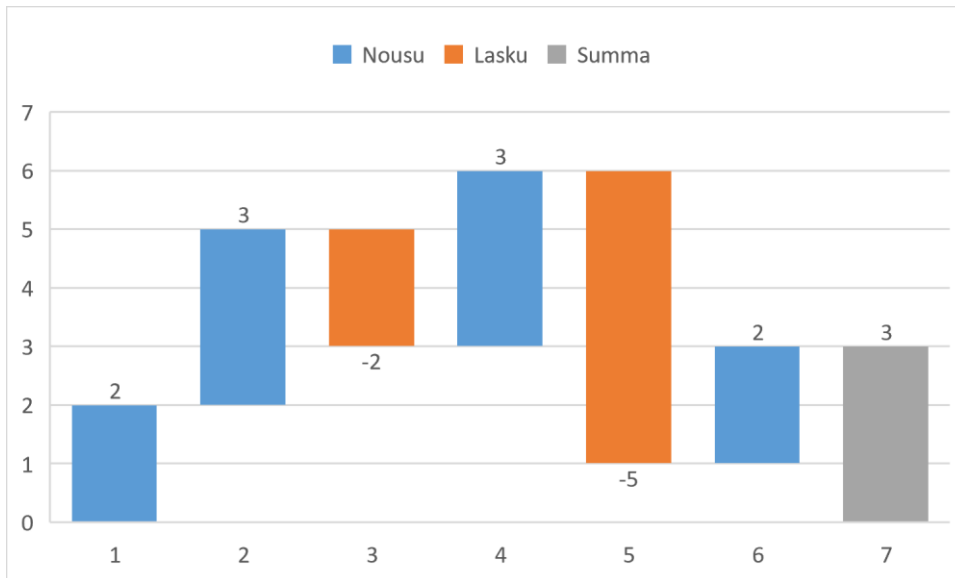


Kuvio 14. Laatikko-janakaavio

### Vesiputouskaavio (Waterfall Chart)

Vesiputouskaavio muistuttaa visuaalisesti pylväskaaviota, mutta käyttötarkoitus eroaa perinteisestä pylväskaaviosta merkittävästi. Vesiputouskaaviota käytetään esittämään kumulatiivinen tulos muuttujan arvoista tietyllä aikavälillä. Kaaviolla pyritään näyttämään muuttujan vaihtelua positiivisen ja negatiivisen välillä. (SAP n.d.)

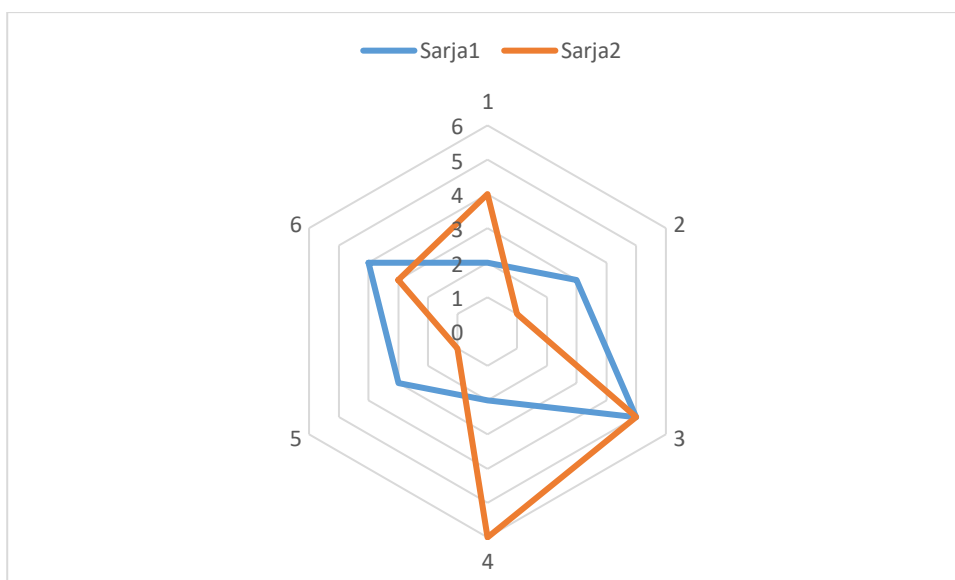
Kaaviossa positiiviset ja negatiiviset lisäykset tulisi visualisoida siten, että lukija voi helposti ymmärtää, mikä lisäys on negatiivista ja mikä positiivista. Lisäksi vesiputouskaavioon kuuluu oleellisesti kaikkien lisäysten lopullinen summa, joka esitetään kaavion päätteeksi viimeisenä muuttujana. Summa tulisi myös esittää siten, että lukija mieltää sen selkeästi lopulliseksi summaksi. (SAP n.d.)



Kuvio 15. Vesiputouskaavio

### Tutkakaavio (Radar Chart)

Tutkakaavio on yhdistelmä useammasta kaaviosta. Tavallisesta poiketen kaaviot eivät mene vasemmalta oikealle, vaan kiertävät ympyrän tai monikulmion kehää. Tutkakaavion avulla voidaan helposti erottaa eri kategorioiden arvojen jakaantuminen eri osa-alueille ja kaaviosta on helppo erottaa poikkeavat havainnot. Kaaviosta näkee selkeästi ja helposti mitkä kategoriat saavat suurimmat ja pienimmät arvot kullakin osa-alueella. (SAP n.d.)



Kuvio 16. Tutkakaavio

## 4.5 Visualisointitapojen ominaisuudet

Ainoa todellinen tavoite visualisoinnissa on maksimoida sen selkeys. (Koponen & Hildén 2019, 207.)

Pelkkä visualisointitavan valitseminen ei kuitenkaan riitä luomaan helppolukuista ja samalla tehokasta visualisointia. Jokaisella kaaviolla on olemassa erilaisia ominaisuuksia, kuten akseleita, selitteitä, tietokerroksia ja värityksiä, joita tulee ottaa huomioon. Yleinen ohje kaikkiin kaavioita koskeviin ominaisuuksiin on: vähemmän on enemmän. Jokaisen yksityiskohdan kaaviossa tulisi olla merkitsevä, älä sisällytä kaavioon mitään ylimääräistä. (SAP n.d.)

### **Otsikko (Headline)**

Visualisoinnissa tulisi olla mukana otsikko, joka herättää lukijan mielenkiinnon, sekä kertoo visualisoitavan ydinasian selkeästi. (IDRC n.d.)

### **Akselit (Axis)**

Akselit tulisivat olla selkeät ymmärrettävyyden lisäämiseksi. Ymmärrettävien visualisointien akseleilla on nimet ja nimet on kirjoitettu vaakatasoon helppolukuisuuden takaamiseksi. (IDRC n.d.) Lisäksi akselien tulisi alkaa nollasta, mikäli mahdollista. Jos akselia ei aloiteta nollasta, antaa visualisointi väärän kuvan suhteellisista osuuksista, mikä voi johtaa vääriin analysointeihin ja täten vääriin johtopäätöksiin. (HubSpot n.d.)

### **Lähteet (Sources)**

Luotettavuuden lisäämiseksi kaaviossa tulee olla mainittuna mahdolliset lähteet. (IDRC n.d.)

### **Tietokerrokset (Data labels)**

Lisää helppolukuisuutta tuo tietokerrokset, joissa kerrotaan muuttujan tarkka arvo. Tällöin lukijan ei itse tarvitse arvioida arvoa akseleista. Tietokerroksen tulisi kuitenkin mahtua kaavion sisään tai välittömään läheisyyteen. (Ficek 2015)

### **Selitteet (Legends)**

Mikäli useita eri kaavioita, tulee visualisoinnissa olla selitteet. Tällöin lukija voi tunnistaa helposti eri kategorioille kuuluvat kaaviot. (HubSpot n.d.) Selitteiden tulisi olla

mahdollisimman lähellä elementtiä, johon sillä viitataan. (Koponen & Hildén 2019, 216.)

### **Trendiviiva (Trendline)**

Mikäli kaavion tarkoituksena on esittää visualisoitavan osa-alueen trendiä, visualisointiin voidaan lisätä trendiviiva, joka kertoo selkeästi muuttujan trendin. (HubSpot n.d.)

### **Järjestys (Ordering)**

Datajoukot tulisi aina järjestää jonkin dataan liittyvän ominaisuuden perusteella, kuten arvojen suuruuden mukaan. Aakkosellinen tai muu irrelevantti järjestystapa on harvoin hyvä vaihtoehto. Mikäli kaaviossa on useita arvoja samoissa kategorioissa, järjestys muodostetaan ensimmäisen kategorian mukaan. (Koponen & Hildén 2019, 208–209.)

### **Värit (Colors)**

Kuten kaikkea muutakin, värejä tulee käyttää suurella varovaisuudella. Väriä valitessa tulee ottaa huomioon erilaiset värisokeudet. (Nayak n.d.) Väreillä tulee olla myös jokin merkitys. Värejä voi käyttää kiinnittämään lukijan huomion haluttuun dataan, korostamalla kyseisen elementin värystä. (IDRC n.d.) Mikäli visualisointi vaatii värejä, esimerkiksi uuden ulottuvuuden luontiin, tulisi eri väreiksi valita yhden värin eri sävyjä. Tällöin visualisoinnista ei tule nopeasti sateenkaaren kirjavaa väriloistoa, vaan silmille mukava ja värisokeille ystävällinen vaihtoehto. (Rabah n.d.)

## **4.6 Visualisoitavan datan tyyppi**

Ennen visualisointitavan päättämistä on hyvä pysähtyä miettimään, mitä konkreettisesti halutaan visualisoida, mihin kysymykseen halutaan vastaus, minkä tyyppistä dataa visualisoidaan ja minkälaisia arvoja esittämistavassa tullaan käyttämään. Esittämistapaa valitessa voi kuitenkin seuraaviin yleistyksiin. (SAP n.d.)

### **Muutos ajan suhteen**

Kun halutaan visualisoida datan muutosta ajan suhteen tai halutaan havaita trendejä valitusta datasta, tulisi valita jokin seuraavista kaavioista:

1. viivakaavio

2. pylväskaavio
3. bullet-kaavio.

Tällöin y-akselilla on yleensä datan saamat arvot ja x-akselilla on kuvattuna ajan muutos tietyn väliajoin, esimerkiksi kuukausittain, kvartaaleittain tai vuosittain. (SAP n.d.)

### **Vertailu**

Kun halutaan vertailla eri kategorioita, joilla ei ole olennaista järjestystä, seuraavat kaaviot toimivat parhaiten:

1. pylväskaavio
2. vesiputouskaavio.

Vertailua luodessa on hyvä huomioida se, että akseleiden järjestyksellä ei ole suurta merkitystä, joten x- ja y-akselit voidaan helposti vaihtaa toisinpäin. Tällöin saadaan lisää tilaa kategorioiden kuvauksille. Toinen akseleista toimii aina kategorioiden esittäjänä ja toinen lukumäärän esittäjänä. (SAP n.d.)

### **Sijoitukset**

Kun halutaan tutkia kategorioiden tai muuttujien sijoituksia halutulla osa-alueella, tai esittää suurimmat ja/tai pienimmät arvot halutusta datasta, seuraavat kaaviot toimivat parhaiten:

1. pylväskaavio
2. kuplakaavio.

Pylväskaavion pylväät ovat helppolukuisia ja on helposti järjestettävissä kokojärjestykseen. Kuplakaavion pinta-alan vertailu toimii myös hyvin tällaisessa tilanteessa. (SAP n.d.)

### **Osuus kokonaisuudesta**

Vertaillessa muuttujien osuutta useamman eri muuttujan kokonaisuudesta, toimivimman kokonaisuuden luo:

1. pylväs- ja aluekaavio (pinottu)

## 2. piirakkakaavio.

Osuutta visualisoidessa tavallinen pylväs- tai aluekaavio ei ole kuitenkaan paras ratkaisu, vaan pinotut pylväs- ja aluekaaviot. Tällöin pylväät ja viivat ovat kumulatiivisia, jolloin muuttujien osuudet ovat päällekkäin ja tällöin helposti vertailtavissa koko kokonaisuuteen. Piirakkakaaviota käyttäessä tulee edelleen huomioida kategorioiden lukumäärä selkeyden säilyttämiseksi. (SAP n.d.)

### **Jakauma**

Tietyn kategorian arvojen jakautumista visualisoitaessa tulisi käyttää seuraavia kaavioita:

1. laatikko-janakaavio
2. lämpö- ja puukaavio
3. tutkakaavio.

Arvojen jakaumasta halutaan saada nopeasti selville tietyn arvojoukon jakautuminen kategorioissaan. Tällöin muuttujia on yleensä useampia, joten tavalliset kaaviot eivät ole vaihtoehtona. (SAP n.d.)

### **Korrelaatio**

Korrelaatiota visualisoitaessa käytetään seuraavia kaavioita:

1. hajontakaavio
2. kuplakaavio.

Korrelaatiota (riippuvuutta) visualisoitaessa halutaan nähdä muuttujien mahdollinen riippuvuus toisistaan. Tähän parhaiten soveltuu pistemäiset visualisointitavat havainnoimaan muuttujien vaihtelua ja mahdollisia kaavoja. Kuplakaavion avulla saadaan useampia muuttujia samaan kaavioon. (SAP n.d.)

## 4.7 Datan visualisoinnin hyödyt

Vanhan sanonnan mukaisesti: kuva kertoo enemmän kuin tuhat sanaa. Ihminen pystyy käsittelemään huomattavan paljon nopeammin ja tehokkaammin visualisointiin



liitetyn monimutkaisen datan, kun pitkiä tekstipohjaisia raportteja. Raporttien toinen ongelma on niiden laatimisen hitaus, minkä takia raporttien pohjalta tehdyt päätökset tulevat pitkällä viiveellä. Tätä ongelmaa ei visualisoinnilla ole, joten visualisoinnista saatuun dataan voidaan reagoida välittömästi. Visualisoinnilla voidaan havaita muuttuvia, hälyttäviä trendejä, jotka vaativat välitöntä reagoitua, minkä avulla voidaan saada etulyöntiasemaa kilpailijoihin nähden sekä minimoida trendin muutoksista koituvat vahingot tai seuraamukset. (SAS n.d.)

Visualisoinnin avulla voidaan myös havaita mahdollisia korrelaatioita muuttujien välillä. Osa korrelaatioista on itsestään selviä, mutta osa korrelaatioista voi tulla myös suurena yllätyksenä, joita ei voisi havaita ilman visualisointia. Myös mahdolliset poikkeavat arvot ovat havaittavissa visualisoinnista. (SAS n.d.)

Visualisointi on universaali ja nopea tapa välittää tietoa, jolloin tieto on ymmärrettävissä lähes poikkeuksetta tiedon vastaanottajasta riippumatta. (SAS n.d.) Datan viestinnän kannalta tärkeintä onkin, että vastaanottaja ymmärtää näkemänsä datan. (Deloitte 2017)

Hyvin suunniteltu visualisointikokonaisuus antaa suuripiirteisen kuvan yrityksen sen hetkisestä tilasta tai halutuista yrityksen osa-alueista tai osastoista. Säännöllisesti päivitetyllä kokonaisuudella saadaan selville jopa lähes reaaliajassa mahdolliset ongelmakohdat, sekä ongelmien ratkaisut. Lisäksi kokonaisuudella voidaan tarkkailla erilaisten ratkaisujen vaikutusta muihin osa-alueisiin. Kaikki tämä on nähtävissä jopa vain pienellä vilkaisulla, käyttämättä aikaa datan analysointiin numeroista tai teksteistä. (Yin ym. 2014)

### **Visualisointi ja raportointi**

Visualisointi ja raportointi sekoitetaan liian usein keskenään. Tällaiset visualisointikokonaisuudet eivät ole kuitenkaan verrattavissa raportointiin. Visualisoinnilla ei saada syvää kuvaa kuvatussta ilmiöstä, tai yksityiskohtaisia raportteja, toisin kuin raporteilla. Yleisesti myös virheellisesti luullaan, että visualisoinnilla tarkoitetaan kokoelmaa taulukoita. Visualisoinnissa voi olla myös taulukoita, mutta se ei kuitenkaan tarkoita sitä. Taulukot eivät yleensä ole tehokkaita tapoja visualisoida dataa. Visualisointi ei ole

taidetta, sillä visualisoinnin on tarkoitus olla mahdollisimman helposti ymmärrettävä ja tulkittava, mikä tarkoittaa mahdollisimman yksinkertaista visualisointia. Visualisointi ei esitä uskomuksia tai mielipiteitä, vaan perustuu täysin kerättyyn riippumattomaan tietoon. (Yin ym. 2014)

## 5 Datan visualisointi Monitorointi-ohjelmistossa

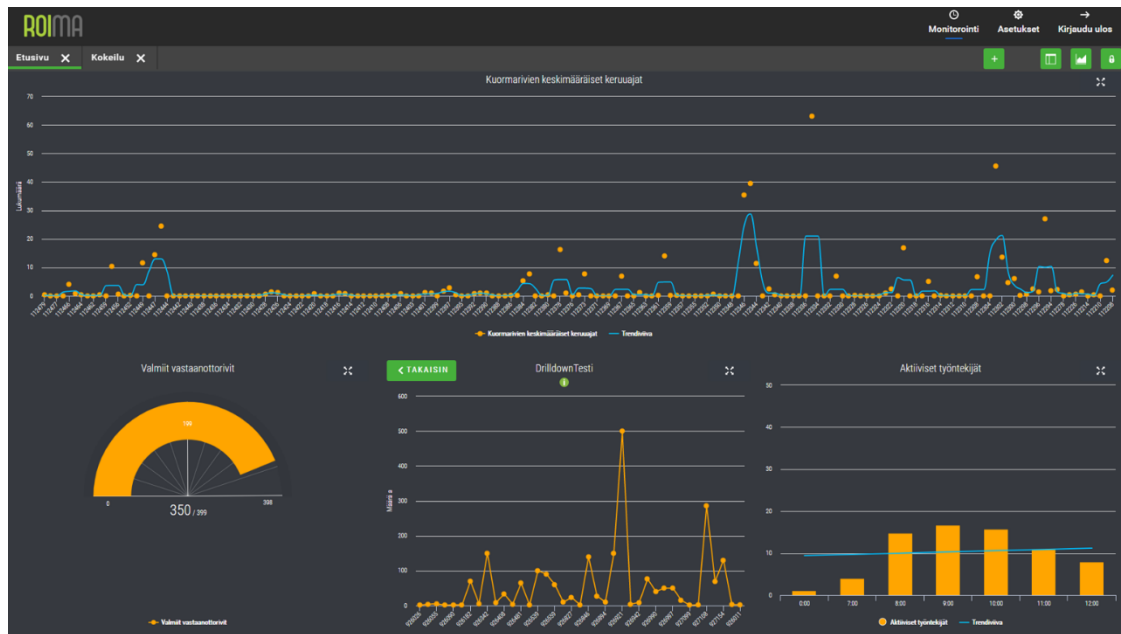
Tämän osion tarkoituksena on koota yhteen edellisissä luvuissa kerätty teorian tieto, luoda teoriasta kriteeristö, joka määrittää kattavan visualisointisovelluksen ja vertailla toimeksiantajan sovellusta kriteeristöön, sekä luoda vertailun pohjalta toimintasuunnitelma ohjelmiston jatkokehittämistä varten. Sitä ennen esitellään toimeksiantajan sovellus visualisointimenetelmien ja niihin liittyvien ominaisuuksien osalta. Muut ominaisuudet eivät ole oleellisia tämän tutkimuksen kannalta, joten ne jätetään huomiotta.

### 5.1 Ohjelmiston esittely

Roimassa kehitettiin vuosien 2018 ja 2019 aikana Monitorointi-nimistä ohjelmistoa, jonka tarkoituksena on visualisoida käyttäjän (asiakkaan) luomaa ja/tai keräämää dataa. Ohjelmiston käyttäjä voi tuoda esiin haluamaansa dataa, haluamastaan tietokannasta haluamallaan esitysmenetelmällä. Kun tätä tutkimusta alettiin toteuttamaan, oli ohjelmistossa saatavilla seuraavat visualisointitavat:

- pylväskaavio (myös pinottu)
- palkkikaavio (vaakatasossa oleva pylväskaavio)
- viivakaavio
- aluekaavio (myös pinottu)
- piirakkakaavio
- mittari
- hajontakaavio

- vaihteluvälikaavio
- taulukko.



Kuvio 17. Ohjelmiston käyttöliittymästä otettu kuvakaappaus 1

Yllä olevassa kuvassa on esiteltyä erilaisia kaaviotyyppejä. Tässä kuvassa kaaviotyyppejä ovat hajontakaavio, mittari, viivakaavio ja pylväskaavio (ylhäältä alaspäin ja vasemmalta oikealle lueteltuna). Lisäksi visualisointimenetelmien ominaisuuksista ovat näkyvissä otsikot, y-akselin nimi, selitteet, infoteksti (esim. lähteille) ja trendiviivat.

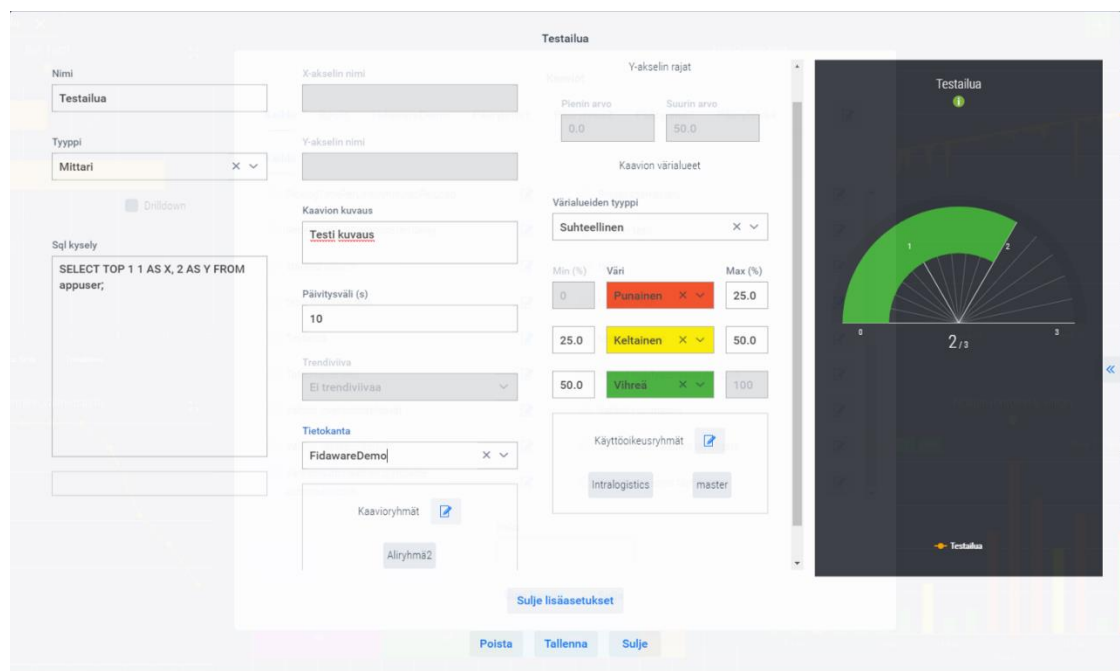


Kuvio 18. Ohjelmiston käyttöliittymästä otettu kuvakaappaus 2

Yllä olevassa kuvassa uusina kaavioina ovat aluekaaviosta muodostettu vaihteluväli-kaavio (yläoikea) ja pinottu pylväskaavio (alhaalla keskellä). Uutena ominaisuutena huomataan väritys.

Visualisointitavan lisäksi käyttäjä voi muokata visualisointitapaa lukuisilla erilaisilla asetuksilla, joita ovat seuraavat ominaisuudet:

- otsikko
- kuvaus (kuvauksessa voidaan kertoa lisätietoja kaaviosta, jonka avulla voidaan selittää mahdollisesti yksityiskohtia, lähteitä tai muuta huomioitavaa asiaa)
- x-akselin nimi
- y-akselin nimi
- y-akselin vaihteluväli
- päivitystiheys
- kuvaajan väritys (väriytyksen saa myös vaihtelevaksi arvojen suhteellisen tai absoluuttisen eroavaisuuden mukaan).



Kuvio 19. Ohjelmiston käyttöliittymästä otettu kuvakaappaus 3.

Yllä olevassa kuviossa on esiteltyä kaavion luominen Monitorointi-ohjelmistossa. Kuvasta huomataan hyvin edellä mainitut ominaisuudet. Lisänä on myös muita ominaisuuksia, jotka eivät suoraan liity visualisointiin, joten näiden ominaisuuksien tulkinta jätetään tämän tutkimuksen ulkopuolelle.

Kaaviota luodessaan käyttäjä hakee tietokannasta haluamansa datan SQL-koodikielen komennon avulla, jolloin käyttäjä voi lauseketta muokkaamalla vaihtaa visualisoitavaa dataa helposti.

### **Esimerkki**

Yritys X haluaa visualisoida aktiivisten työntekijöiden määrä kunakin tasatuntina erään paikkakunnan toimipisteellä. Tällöin asiakas valitsee kyseisen toimipisteen tietokannan. Asiakas nimeää kaavion ”Aktiiviset työntekijät”, jotta kaaviosta on huomattavissa välittömästi, mistä kaavio kertoo. Kaaviotyyppiä asiakas valitsee pylväskaavion, koska kokee tämän selkeimmäksi vaihtoehdoksi ja koska vertailuun pylväskaavio on yksi parhaista vaihtoehdoista. Asiakas haluaa tämän lisäksi jokaiseen palkkiin näkyviin palkin arvon selkiyttääkseen arvojen vaihtelua, joten hän asettaa kaavioon tietokerrokset näkyviin. Y-akselin nimeksi laitetaan ”Lukumäärä”, välttääkseen väärinymmärryksiä. Lisäksi hän muuttaa palkin väriksi keltaisen, koska keltainen on yritykselle omainen väri. Työntekijöiden aktiivisuus ei ole asiakkaan mielestä erityisen tärkeää tietoa, joten päivitysväliksi asetetaan yksi tunti. Muita asetuksia asiakas ei koe hyödylliseksi, joten asiakas päättää olla käyttämättä niitä.

## **5.2 Visualisointiohjelmiston kriteeristö**

Jotta ohjelmiston kattavuutta voitaisiin tarkastella visualisointimenetelmien osalta, tulee olla olemassa kriteeristö, johon ohjelmiston kattavuutta voidaan verrata. Tämän kriteeristön avulla voidaan siis arvioida toimeksiantajan ohjelmiston visualisointimenetelmien ja siihen kuuluvien ominaisuuksien kattavuutta. Edellisessä pääluvussa kerättyä teoretistä tietoa hyödyntäen luodaan kriteeristö visualisointimenetelmien ja niihin liittyvien ominaisuuksien kattavuuden vertailulle.

Kriteeristössä on kaksi eri osa-aluetta, jotka koskevat visualisointimenetelmiä sekä visualisoinnin ominaisuuksia. Visualisointimenetelmissä arvioidaan menetelmien kat-

tavuutta ja ominaisuuksissa arvioidaan menetelmiin lisättävien ominaisuuksien kattavuutta. Näiden kahden osa-alueen avulla voidaan helposti arvioida ohjelmiston kattavuutta.

### Visualisointimenetelmät

Ohjelmistossa tulee olla riittävän kattavat visualisointimenetelmät. Ohjelmistossa tulee olla valittavana jokaiselle visualisoitavalle datatyypille vähintään yksi sille hyvin soveltuva visualisointimenetelmä. Seuraavista ryhmistä tulisi visualisointiohjelmistossa olla vähintään yksi visualisointitapa:

- viivakaavio, pylväskaavio tai bullet-kaavio (muutos ajan suhteen)
- pylväskaavio tai vesiputouskaavio (vertailu)
- pylväskaavio tai kuplakaavio (sijoitukset)
- pylväs-/aluekaavio (pinottu) tai piirakkakaavio (osuus kokonaisuudesta)
- laatikkojana-kaavio, puukartta tai tutkakaavio (jakauma)
- hajontakaavio tai kuplakaavio (korrelaatio).

Taulukko 1. Kriteeristöä muodostettu taulukko.

|                       | Kaavio 1            | Kaavio 2          | Kaavio 3       |
|-----------------------|---------------------|-------------------|----------------|
| Muutos ajan suhteen   | Viivakaavio         | Pylväskaavio      | Bullet-kaavio  |
| Vertailu              | Pylväskaavio        | Vesiputouskaavio  |                |
| Sijoitukset           | Pylväskaavio        | Kuplakaavio       |                |
| Osuus kokonaisuudesta | Pylväskaavio (pino) | Aluekaavio (pino) | Piirakkakaavio |
| Jakauma               | Laatikko-janakaavio | Puukartta         | Tutkakaavio    |
| Korrelaatio           | Hajontakaavio       | Kuplakaavio       |                |

Taulukon avulla voidaan vertailla ohjelmiston kattavuutta visualisointimenetelmien osalta. Vertailtavasta ohjelmistosta tulisi löytyä vähintään yksi visualisointitapa jokaiselta riviltä. Tällöin ohjelmiston kattavuudeksi saadaan 100 %, eli ohjelmistolla pystytään visualisoimaan erilaisia tilanteita kattavasti. Erilaisten tilanteiden kannalta olisi suotavaa, että ohjelmistosta löytyisi kaikki taulukosta löytyvät visualisointimenetelmät.

### **Visualisoinnin ominaisuudet**

Ohjelmiston visualisoinnissa tulisi olla riittävästi ominaisuuksia, joiden avulla visualisointia pystytään selkeyttämään ja tehostamaan. Näitä ovat:

- x-akselin nimi
- y-akselin nimi
- otsikko
- paikka lähteille
- y-akselin arvojoukon alkaminen nolasta
- tietokerrokset
- selitteet
- trendiviiva
- väriasetukset
- järjestys.

Vertailtavasta ohjelmistosta tulee löytyä vähintään edellä mainitut ominaisuudet selkeän ja tehokkaan visualisointimenetelmän luomiseksi.

### **5.3 Ohjelmiston vertailu**

Tässä luvussa suoritetaan toimeksiantajan Monitorointi-ohjelmiston vertailu edellisessä luvussa luotuun kriteeristöön, jonka perusteella luodaan ohjelmiston kehittämiseen perustuva toimintasuunnitelma.

#### **Visualisointimenetelmien vertailu**

Arvioidaan ohjelmiston visualisointimenetelmien kattavuutta taulukon avulla. Taulukossa on merkattu vihreällä taustavärillä ohjelmistosta löytyvät visualisointimenetelmät ja punaisella taustavärillä ohjelmistosta puuttuvat visualisointimenetelmät. Keltaisella taustavärillä on merkattuna ohjelmistossa eri tavalla toteutetut visualisointimenetelmät, joiden tavoitteena on visualisoida samantyylliset datatyypit, mutta eivät

ole suositelluimpia visualisointitapoja. Kriteereinä on käytetty tässä pääluvussa luotua kriteeristöä.

Taulukko 2. Arvioitavan ohjelmiston visualisointimenetelmien kattavuus kriteerien perusteella.

|                       | Kaavio 1            | Kaavio 2          | Kaavio 3       |
|-----------------------|---------------------|-------------------|----------------|
| Muutos ajan suhteen   | Viivakaavio         | Pylväskaavio      | Bullet-kaavio  |
| Vertailu              | Pylväskaavio        | Vesiputouskaavio  |                |
| Sijoitukset           | Pylväskaavio        | Kuplakaavio       |                |
| Osuus kokonaisuudesta | Pylväskaavio (pino) | Aluekaavio (pino) | Piirakkakaavio |
| Jakauma               | Laatikko-janakaavio | Puukartta         | Tutkakaavio    |
| Korrelaatio           | Hajontakaavio       | Kuplakaavio       |                |

Taulukosta huomataan, että Monitorointi-ohjelmistosta puuttuu jakauman esittämiseen parhaiten soveltuvat visualisointitavat. Muutos ajan suhteen, vertailu, sijoitukset ja osuuden vertailut ovat ohjelmistossa visualisoitavissa. Jakauman visualisointimiseksi ohjelmistossa ei ole täysin kyseiseen tyyppiin sopivaa menetelmää. Visualisoitavien tyyppien kattavuus on siis 83 %. Kaikkien kaavioiden kattavuus on 53 %, kun korvaavia kaaviovaihtoehtoja ei huomioida. Keltaisella taustavärillä merkattu bullet-kaavio on ohjelmistossa toteutettu mittarin muodossa, mittarin käyttöä ei kuitenkaan suositella, mikäli bullet-kaavio on mahdollinen (SAP n.d.). Mittarin avulla voidaan visualisoida bullet-kaavion tavoin tavoitetta tai päämäärää, joten kaavion tuottama visualisointitarve on katettu. Lisäksi keltaisella merkittyä laatikkojana-kaavion korvaa osittain vaihteluvälikaavio, joka esittää arvojen vaihteluvälin minimi- ja maksimiarvon avulla. Laatikkojana-kaavion avulla voidaan kuitenkin yksittäisellä kaaviolla antaa suurempi määrä informaatiota, minkä takia laatikkojana-kaavio on suositellumpi vaihtoehto kaavionvalintaan.

Vertailusta huomataan, että Monitorointi-ohjelmistossa on puutteita visualisointimenetelmien osalta.

### Visualisoinnin ominaisuuksien vertailu

Arvioidaan ohjelmiston visualisointimenetelmien ominaisuuksien kattavuutta taulukon avulla. Taulukossa on merkattu vihreällä taustavärillä ohjelmistossa olevat ominaisuudet ja punaisella taustavärillä ohjelmistosta puuttuvat ominaisuudet. Keltaisella taustavärillä on merkattu ohjelmistossa puutteellisesti toteutetut ominaisuudet.



Lisäksi taulukossa on esitetty huomioita ominaisuuksien toteutuksesta ohjelmistossa. Kriteereinä on käytetty tässä pääluvussa luotua kriteeristöä.

Taulukko 3. Arvioitavan ohjelmiston visualisoinnin ominaisuuksien kattavuus kriteerien perusteella.

|                            | Huomioita   |
|----------------------------|---|
| X-akselin nimi             | Vain x-akselin omaaville kaavioille                     |
| Y-akselin nimi             | Vain y-akselin omaaville kaavioille                     |
| Otsikko                    |   |
| Paikka lähteille           | Erillinen paikka lisäinformaatiolle                     |
| Y-akselin arvojoukon arvot | Minimi- ja maksimiarvon voi asettaa                     |
| Tietokerrokset             |   |
| Selitteet                  | Automaattinen, kun useampi kuin yksi kaavio             |
| Trendiviiva                | Useita trendiviivatyyppisiä                             |
| Väriasetukset              | Useita värejä saatavilla, mutta ei yhden värin sävyinä  |
| Järjestys                  | SQL-komennon avulla voidaan järjestää halutulla tavalla |

Taulukosta huomataan, että vain tietokerrokset puuttuvat kokonaan Monitorointi-ohjelmistosta. Ominaisuuksien kattavuus on siis 80 %, kun puutteellisesti toteutetuista ominaisuuksista ei huomioida. Väriasetukset ovat toteutettu puutteellisesti, sillä ohjelmistossa on saatavilla täysin eri värejä, mutta ei yhden värin eri sävyjä, mikä on suotavampaa värisokeiden kannalta ja samalla vältetään sateenkaariefektin syntymistä (Rabah n.d.).

Vertailusta huomataan, että Monitorointi-ohjelmistossa on puutteita visualisointimenetelmien ominaisuuksien osalta.

## 5.4 Toimintasuunnitelma

Edellisessä kappaleessa huomattiin puutteita Monitorointi-ohjelmistossa niin visualisointimenetelmien, kuin niiden ominaisuuksienkin osalta. Ohjelmistosta puuttuu useita kriteeristössä mainittuja menetelmiä, sekä kaksi korvaavilla tavoilla toteutetuista menetelmistä. Huomioitavaa on myös, että ohjelmistosta puuttuu jakaumaan sopivimmat menetelmät kokonaan. Menetelmien lisäksi, ohjelmistosta puuttuu yksi ominaisuus, joka lisäisi visualisoinnin helppolukuisuutta ja yksi ominaisuus on toteutettu puutteellisesti kriteeristön perusteella. Huomioitavaa on kuitenkin se, että ohjelmistosta löytyy kriteeristön ulkopuolella olevia visualisointimenetelmiä. Kattavan

visualisointiohjelmiston luomiseksi, luodaan toimintasuunnitelma ohjelmiston kehittämiseksi. Ohjelmiston ominaisuudet, jotka eivät liity visualisointiin, jätetään huomiotta.

### **Suunnitelma**

Toimintasuunnitelma toteutetaan Roima Intelligencen Monitorointi-ohjelmistolle. Suunnitelma toteutetaan, sillä ohjelmistossa havaittiin puutteita. Toimenpiteiden tarkoituksena on kehittää ohjelmistoa siten, että ohjelmisto vastaa visualisointiohjelmiston tarpeisiin mahdollisimman kattavasti, ottaen huomioon ohjelmistossa käytettyjen teknologioiden mahdollisuudet.

Tutkimusta aloittaessa haluttiin varmistaa ohjelmiston kattavuus visualisointiohjelmistona. Ohjelmistossa oli jo ennen tutkimusta melko kattava kaaviovalikoima, mutta sitä haluttiin tarkastella lisää. Tutkimuksen teoriaosuuden perusteella tehdyn kriteeristön pohjalta tehdyssä vertailussa havaittiin huomattavia puutteita visualisointimenetelmien osalta, sekä pieniä puutteita menetelmiin liittyvissä ominaisuuksissa. Jotta ohjelmisto täyttäisi kriteerit, tulee ohjelmistoon lisätä neljä uutta kaaviota ja yksi puuttuva ominaisuus. Lisäksi ohjelmistosta löytyy kaksi vaihtoehtoista kaaviota, sekä osittain virheellisesti toteutetut väriasetukset. Ohjelmistoon lisätään seuraavat kaaviotyypit:

- vesiputouskaavio
- kuplakaavio
- puukartta
- tutkakaavio.

Kaaviotyyppien lisäksi ohjelmistoon lisätään seuraava ominaisuus:

- tietokerrokset.

Tutkimuksessa havaitut puutteet toteutetaan Monitorointi-ohjelmistoon osana tätä tutkimusta kevään 2019 aikana.

## 6 Monitorointi-ohjelmiston kehittäminen

Ohjelmisto oli jo ennen tutkimuksen aloittamista toimintalogiikan puitteissa valmis, mutta kaaviotyypin kattavuutta haluttiin selvittää. Tästä syystä ohjelmiston toimintalogiikkaan ei oteta kantaa millään tavalla.

Tässä luvussa esitellään ja käsitellään ohjelmistoon lisätyt kaaviotyypit ja ominaisuudet. Osiot sisältävät esimerkkejä ohjelmiston koodista, jotka käsittelevät lisättyjä kaaviotyyppejä ja ominaisuuksia. Lisäksi osioissa näytetään toteutuneet visualisoinnit kuvakaappauksin sisältäen konkreettisia esimerkkejä.

### 6.1 Lisättävät kaaviotyypit

Toimintasuunnitelmassa (luku 5.4) lueteltiin neljä kaaviotyyppiä, jotka ohjelmistoon tulee lisätä. Tässä luvussa esitellään kaaviotyyppien toteutus ja konkreettinen esimerkki toteutetusta kaaviotyypeistä.

Monitorointi-ohjelmisto on toteutettu käyttäen Vaadin-sovellusalustaa. Vaadin-sovellusalustassa on saatavilla suuri valikoima erilaisia kaavioita, jotka soveltuvat suurimpaan osaan visualisointitapauksista. (Vaadin n.d.) Kaikki ohjelmiston kaaviot on toteutettu Vaadin-sovellusalustan teknologioilla käyttäen Java-ohjelmointikieltä.

#### **Vesiputouskaavio**

Vesiputouskaavio oli yksi sovelluksesta puuttunut kaaviotyyppi. Kuten kaikki muutkin kaaviotyypit, myös tämä kaaviotyyppi toteutettiin omaan luokkaansa (class), jossa määritellään kaavion ominaisuudet käyttäjän asettamien arvojen mukaan.

Itse vesiputouskaavion luominen toteutetaan Vaadin-sovellusalustalla hyvinkin samanlaisesti, kuin minkä tahansa kaavion luominen. Halutut datapisteet haetaan/asetetaan (tässä itse datahakua ei ole esitetty) datajoukkoon, joka asetetaan kaavion datajoukoksi. Lisäksi kaaviotyypiksi tulee valita haluttu kaavio (Waterfall). Mahdolliset käyttäjän asettamat ominaisuudet asetetaan kaavion ominaisuuksiin, näitä ovat x- ja y-akselin nimet, y-akselin arvot, kaavion nimi eli otsikko ja tietokerrokset. Lisäksi selitteet poistetaan yksittäisen kaavion tapauksessa. Vesiputouskaaviossa asetetaan erikseen viimeiseksi kategoriaksi summa, joka kertoo edellisten kategorioiden summan (ks. kuvio 20).

```

public class WaterfallChart extends Chart {
    private static final long serialVersionUID = 1L;

    public WaterfallChart (ICustomDashBoard dashboardBL, Chartscript chartscript) {
        Configuration conf = this.getConfiguration();
        this.setId(chartscript.getId() + "");
        conf.setTitle(chartscript.getChartname());
        DataSeries series = new DataSeries();
        try {
            series = dashboardBL.getData(chartscript, false);
        } catch (Throwable e) {
            new ErrorNotification(Translator.getText("error.sql.not.valid") + " " + chartscript.getChartname(),
                e.getMessage());
            return;
        }
        series.setName(chartscript.getChartname());
        WaterFallSum balance = new WaterFallSum(Translator.getText("lbl.total"));
        series.add(balance);
        series.setName(chartscript.getChartname());
        PlotOptionsWaterfall plotOptions = new PlotOptionsWaterfall();
        series.setPlotOptions(plotOptions);
        conf.addSeries(series);
        conf.getxAxis().setType(AxisType.CATEGORY);
        conf.getyAxis().setTitle(new AxisTitle(chartscript.getAxisname()));
        conf.getxAxis().setTitle(new AxisTitle(chartscript.getXaxisname()));
        conf.getyAxis().setMin(chartscript.getMinYValue());
        conf.getyAxis().setMax(chartscript.getMaxYValue());
        conf.getLegend().setEnabled(false);
        Tooltip tooltip = new Tooltip();
        tooltip.setValueDecimals(1);
        tooltip.setHeaderFormat("{series.name}<br/>");
        tooltip.setPointFormat("{point.name}: <b>{point.y}</b>");
        conf.setTooltip(tooltip);
        if (chartscript.getDataLabels().equals("Y")) {
            DataLabels dataLabels = new DataLabels(true);
            dataLabels.setFormat("{y}");
            plotOptions.setDataLabels(dataLabels);
        } else
            plotOptions.getDataLabels().setEnabled(false);
    }
}

```

Kuvio 20. Vesiputouskaavion luomiseen käytetty koodi omassa luokassaan.

### Kuplakaavio

Kuplakaavio oli yksi sovelluksesta puuttunut kaaviotyyppi. Kuten kaikki muutkin kaaviotyypit, myös tämä kaaviotyyppi toteutettiin omaan luokkaansa (class), jossa määritellään kaavion ominaisuudet käyttäjän asettamien arvojen mukaan.

Kuplakaavion luominen tapahtuu merkittävässä määrin samalla tavalla, kuin minkä tahansa muunkin kaavion luominen. Haetaan/asetetaan data datajoukkoon, jonka jälkeen asetetaan käyttäjän luoman arvot kaavion ominaisuuksiin. Huomioitavaa tässä kaaviotyypissä on se, että hakiessa dataa, data tulee erilaisessa muodossa. Tätä varten tuli luoda oma metodi (getBubbleData()), joka palauttaa datan oikeassa muodossa. Lisäksi datatyyppin vaihdon takia, kuplakaavion tietokerrokset tuli asettaa myös (x, y, z) muotoon (ks. kuvio 21).

```

public class BubbleChart extends Chart {
    private static final long serialVersionUID = 1L;

    public BubbleChart (ICustomDashBoard dashboardBL, Chartscript chartscript) {
        DataSeries series = new DataSeries();
        try {
            series = dashboardBL.getBubbleData(chartscript);
        } catch (Throwable e) {
            return;
        }
        series.setName(chartscript.getChartname());
        this.setId(chartscript.getId() + "");
        Configuration conf = this.getConfiguration();
        conf.getLegend().setEnabled(false);
        conf.getChart().setType(ChartType.BUBBLE);
        conf.setTitle(chartscript.getChartname());
        conf.getyAxis().setTitle(new AxisTitle(chartscript.getAxisname()));
        conf.getxAxis().setTitle(new AxisTitle(chartscript.getXaxisname()));
        conf.getyAxis().setMin(chartscript.getMinYValue());
        conf.getyAxis().setMax(chartscript.getMaxYValue());
        DataLabels dataLabels = new DataLabels(true);
        dataLabels.setFormat("{x}" + ", " + "{y}" + ", " + "{point.z}");
        if (!chartscript.getDataLabels().equals("Y"))
            dataLabels.setEnabled(false);
        PlotOptionsBubble plotOptions = new PlotOptionsBubble();
        plotOptions.setMinSize("5");
        plotOptions.setDataLabels(dataLabels);
        series.setPlotOptions(plotOptions);
        conf.setPlotOptions(plotOptions);
        conf.setSeries(series);
    }
}

```

Kuvio 21. Kuplakaavion luontiin käytetty koodi omassa luokassaan.

Yleensä data tulee (x, y) pareina, mutta koska kuplakaaviossa on 3 ulottuvuutta, tulee data (x, y, z) -muodossa. Kuplakaaviossa kaikki muuttujat ovat numeromuodossa, joten data tallennetaan Number-datatypeihin. Jokainen rivi käydään lävitse ja jokaisen rivin x-, y- ja z-arvo tallennetaan omaan tietue (DataSeriesItem3d) muuttujaansa. Lopuksi tietuejoukko (DataSeries) viedään väritettäväksi, jonka jälkeen tietuejoukko palautetaan kaaviolle datapisteiksi (ks. kuvio 22).

```

public DataSeries getBubbleData(Chartscript chartscript) throws FidawareException {
    if (chartscript == null)
        return null;
    GenericTableData data = cache.executeQuery(chartscript, new ArrayList<String>());
    DataSeries series = new DataSeries();
    try {
        for (int i = 0; i < data.getRowData().size(); ++i) {
            Object[] res = data.getRowData().get(i).toArray();
            Number y;
            Number z;
            Number x;
            if (res[0] == null || res[1] == null || res[2] == null)
                continue;
            y = toNumber(res[1]);
            x = toNumber(res[0]);
            z = toNumber(res[2]);
            series.add(new DataSeriesItem3d(x, y, z));
        }
    } catch (Throwable e) {
        logger.error(e);
        e.printStackTrace();
        throw new FidawareException(Translator.getText("error.chartscript.execution.failed", chartscript.toString())
            + "\r\n\r\n" + e.getMessage());
    }
    return getChartItemColors(chartscript, series, 0, 0, new ArrayList<Chartcolor>());
}

```

Kuvio 22. Kuplakaavion datan hakemiseen käytetty koodi.

### Puukartta

Puukartta oli yksi sovelluksesta puuttunut kaaviotyyppi. Kuten kaikki muutkin kaaviotyytit, myös tämä kaaviotyyppi toteutettiin omaan luokkaansa (class), jossa määritellään kaavion ominaisuudet käyttäjän asettamien arvojen mukaan.

Puukartta eroaa huomattavasti tavallisista kaavioista, sillä puukartassa ei ole ollenkaan akseleita. Tämän takia puukartalle ei voida asettaa akseleiden nimiä tai arvoja. Tästä syystä puukaavion luomiseen tarvittava koodikin on huomattavasti lyhyempi kuin muiden. Puukaaviossa on erilaisia asetuksia myös palojen pilkkomiselle, mutta tässä käytetty ”SLICEANDDICE” on oletusasetus. Lisäksi, vaikka puukartta ottaakin sisäänsä (x, y) -muotoisia datapisteitä, tarvitsee puukartta nämä datapisteet eri tietotyyppillä (TreeSeries). Tämän vuoksi puukartalle on luotu oma metodi (getTreeData()) datan hakemiselle (ks. kuvio 23).

```
public class TreemapChart extends Chart {
    private static final long serialVersionUID = 1L;

    public TreemapChart (ICustomDashBoard dashboardBL, Chartscript chartscript) {
        TreeSeries series = new TreeSeries();
        try {
            series = dashboardBL.getTreeData(chartscript);
        } catch (Throwable e) {
            return;
        }
        series.setName(chartscript.getChartname());
        Configuration conf = this.getConfiguration();
        conf.getChart().setType(ChartType.TREEMAP);
        conf.setTitle(chartscript.getChartname());
        PlotOptionsTreemap plotOptions = new PlotOptionsTreemap();
        plotOptions.setAlternateStartingDirection(true);
        plotOptions.setLayoutAlgorithm(TreeMapLayoutAlgorithm.SLICEANDDICE);
        series.setPlotOptions(plotOptions);
        conf.setPlotOptions(plotOptions);
        conf.setSeries(series);
    }
}
```

Kuvio 23. Puukartan datan hakemiseen käytetty koodi.

Puukartan datan hakeminen toimii lähes samalla tavalla kuin aiemmin kuvattu kuplakaavion datanhaku. Eroina ovat vain datatyyppi (TreeSeriesItem), joka ottaa (x, y, z) pisteiden sijasta vain (x, y) pareja, jossa y-arvon tulee olla kokonaisluku, sekä tässä tilanteessa x-akselilla numeroiden sijasta hyväksytään myös tekstiarvo, jolloin data tallennetaan Object-datatyyppiin (ks. kuvio 24).

```

public TreeSeries getTreeData(Chartscript chartscript) throws FidawareException {
    if (chartscript == null)
        return null;
    GenericTableData data = cache.executeQuery(chartscript, new ArrayList<>());
    TreeSeries series = new TreeSeries();
    SimpleDateFormat sdf = new SimpleDateFormat("HH:mm");
    try {
        for (int i = 0; i < data.getRowData().size(); ++i) {
            Object[] res = data.getRowData().get(i).toArray();
            int y;
            Object x;
            if (res[0] == null || res[1] == null)
                continue;
            y = (int) toNumber(res[1]);
            x = res[0];
            if (res[0].getClass().equals(java.sql.Timestamp.class))
                series.add(new TreeSeriesItem(sdf.format(x), y));
            else
                series.add(new TreeSeriesItem(x.toString(), y));
        }
    } catch (Throwable e) {
        logger.error(e);
        e.printStackTrace();
        throw new FidawareException(Translator.getText("error.chartscript.execution.failed", chartscript.toString())
            + "\r\n\r\n" + e.getMessage());
    }
    List<Chartcolor> colors = findChartcolorByChartscript(chartscript);
    for (TreeSeriesItem item : series.getData()) {
        item.setColorIndex(ChartColor.valueOf(colors.get(0).getColor()).getId());
    }
    return series;
}

```

Kuvio 24. Puukartan luomiseen käytetty koodi omassa luokassaan.

### Tutkakaavio

Tutkakaavio oli yksi sovelluksesta puuttunut kaaviotyyppi. Tämä kaaviotyyppi toteutettiin samaan luokkaan (class) muiden yhdistelmäkaavioiden kanssa, koska kaikki nämä kaaviotyypit voivat sisällyttää useamman kuin yhden kaavion, myös erilaisia kaavioita, esimerkiksi pylväs- ja viivakaavion.

Tutkakaavio on yhden tai useamman kaavion joukko, jossa normaalista poiketen kaaviot etenevät ympyrän kehällä vaaka- tai pystysuoran sijaan. Tutkakaavioon voi lisätä pylväs-, viiva-, hajonta-, tai aluekaavioita (ks. kuvio 26). Yksittäisen kaavion datan hakeminen ja sisällyttäminen toteutetaan samalla tavalla kuin ne olisi luotu yksittäisenä. Nyt tässä tapauksessa jokainen kaavio lisätään erikseen yhteen kaavioon luoden yhdistelmäkaavion. Kaaviota luodessaan käyttäjän on mahdollista valita esitystapaksi tutkakaavio (polar), jolloin kaaviosta luodaan tutkakaavio. Ilman tätä asetusta asettelu olisi tavallisen kaavion tapainen. Huomioitavaa on tässä myös se, että perinteisiä akseleita ei ole, joten tutkakaaviossa ei huomioida y-akselille asetettuja arvoja, sekavuuden välttämiseksi. Yksittäisestä kaaviosta poiketen, tutkakaaviossa selitteitä ei poisteta, sillä niiden puuttuminen aiheuttaisi epäselvyyttä eri elementtien tarkoituksesta. Muuten kaavioon asetettavat ominaisuudet vastaavat tavallisen kaavion luomista (ks. kuvio 25).

```

public class CombinationChart extends Chart {

    private static final long serialVersionUID = 1L;

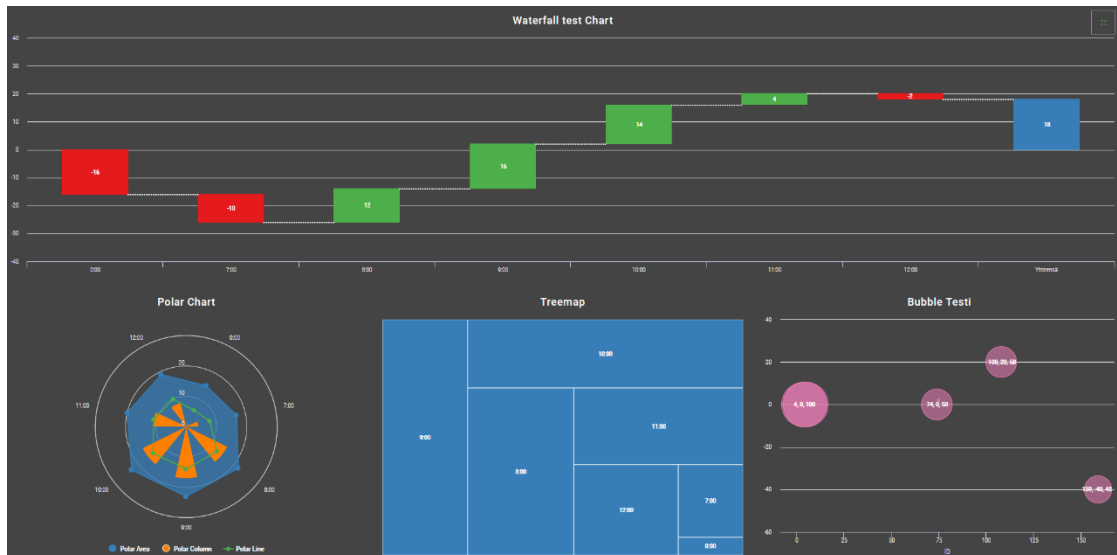
    public CombinationChart(ICustomDashBoard dashboardBL, Chartscript chartscript,
        List<Chartscript> combinationCharts) {
        List<Chartscript> charts;
        if (combinationCharts.size() == 0)
            charts = dashboardBL.getChartsByMasterChart(chartscript);
        else
            charts = combinationCharts;
        if (charts.size() == 0)
            return;
        Configuration conf = this.getConfiguration();
        if (chartscript.getCharttype().contains("polar")) {
            conf.getChart().setPolar(true);
            Pane pane = new Pane(0, 360);
            conf.addPane(pane);
        } else if (chartscript.getCharttype().contains("timeline"))
            this.setTimeline(true);
        this.setId("" + chartscript.getId());
        conf.setTitle(chartscript.getChartname());
        conf.getxAxis().setType(AxisType.CATEGORY);
        conf.getxAxis().setTitle(new AxisTitle(chartscript.getXaxisname()));
        conf.getTooltip().setShared(true);
        sortCharArray(charts);
        for (Chartscript chart : charts) {
            DataSeries series = new DataSeries();
            try {
                series = dashboardBL.getData(chart, chartscript.getCharttype().contains("timeline"));
            } catch (FidawareException e) {
                new ErrorNotification(Translator.getText("error.sql.not.valid") + " " + chartscript.getChartname()
                    + ", " + chart.getChartname(), e.getMessage());
            }
            series.setName(chart.getChartname());
            if (chart.getCharttype().equalsIgnoreCase("column")) {
                PlotOptionsColumn plotOptions = new PlotOptionsColumn();
                series.setPlotOptions(plotOptions);
            } else if (chart.getCharttype().equalsIgnoreCase("line")) {
                PlotOptionsLine plotOptions = new PlotOptionsLine();
                series.setPlotOptions(plotOptions);
            } else if (chart.getCharttype().equalsIgnoreCase("spline")) {
                PlotOptionsSpline plotOptions = new PlotOptionsSpline();
                series.setPlotOptions(plotOptions);
            } else if (chart.getCharttype().equalsIgnoreCase("scatter")) {
                PlotOptionsScatter plotOptions = new PlotOptionsScatter();
                series.setPlotOptions(plotOptions);
            } else if (chart.getCharttype().equalsIgnoreCase("area")) {
                PlotOptionsArea plotOptions = new PlotOptionsArea();
                series.setPlotOptions(plotOptions);
            }
        }
    }
}

```

Kuvio 25. Tutkakaavion luontiin käytetty koodi omassa luokassaan.

Alla olevassa kuvakaappauksessa (ks. kuvio 26) on esiteltynä edellä mainitut neljä erilaista kaaviota, jotka lisättiin sovellukseen. Kuviossa ylhäällä oleva kaavio on vesiputouskaavio ja alarivissä vasemmalta oikealle lueteltuna ovat tutkakaavio, puukartta ja kuplakaavio. Esimerkkikaavioissa data ei varsinaisesti kerro mitään tulosta tai vastausta kysymykseen, vaan kaaviot ovat toteutettu vain kaavion demonstraatiota varten. Vesiputouskaaviossa vihreä väri tarkoittaa positiivista muutosta ja vastaavasti punainen väri tarkoittaa negatiivista muutosta. Sininen väri kertoo lopullisen summan edellä olevien kategorioiden osalta. Muissa kaavioissa väreillä ei visualisoida mitään, vaan ovat esimerkkeinä erilaisista väriasetuksista. Näiden kaaviotyyppien lisäysten jälkeen ohjelmisto vastaa lähes täydellisesti edellisessä pääluvussa asetettuja kaaviovaatimuksia.





Kuvio 26. Toimintasuunnitelman perusteella lisätyt kaaviot käyttöliittymässä.

## 6.2 Lisättävät ominaisuudet

Toimintasuunnitelmassa (luku 5.4) lueteltiin yksi visualisoinnin ominaisuus, joka ohjelmistoon tulee lisätä suuremman kattavuuden luomiseksi. Tässä luvussa esitellään ominaisuuden toteutus, sekä kuvakaappaus kaavionluonnin käyttöliittymästä toteutetun muutoksen jälkeen.

Kuten edellä mainittiinkin, Monitorointi-ohjelmisto toteutettiin käyttäen Vaadin-sovellusalustaa. Vaadin-alustan kaaviokomponentista löytyy kattavat ominaisuudet visualisoinnin selkeyttämiseen. (Vaadin n.d.)

### Tietokerrokset

Tietokerrokset toteutettiin ohjelmistoon käyttämällä DataLabels-ominaisuutta, joka asettaa jokaisen havaintopisteen välittömään läheisyyteen halutun arvon. Yleensä arvolla halutaan näyttää y-akselin tarkka arvo, sillä x-akselilla oleva kategoria on helpposti tulkittavissa. Kuviossa 25 on esitetty kuplakaavion tietokerrosten luominen, mikä eroaa tavallisesta datapisteestä kolmannella ulottuvuudella (pinta-ala), jolloin tietokerrokseen on hyvä lisätä myös pinta-alan arvo. Tietokerrokseen tässä tapauksessa lisättiin myös x-akselin arvo, sillä kuplakaaviota käytetään numeerisissa yhteydessä, ilman kategorioita. Tietokerrokset ovat poistetaan kaaviosta käyttäjän näin halutessa, jolloin tietokerrokset otetaan pois käytöstä.

```

DataLabels dataLabels = new DataLabels(true);
dataLabels.setFormat("{point.x}" + ", " + "{point.y}" + ", " + "{point.z}");
if (!chartscript.getDataLabels().equals("Y"))
    dataLabels.setEnabled(false);

```

Kuvio 27. Tietokerrosten luominen eräässä kaaviotyypissä.

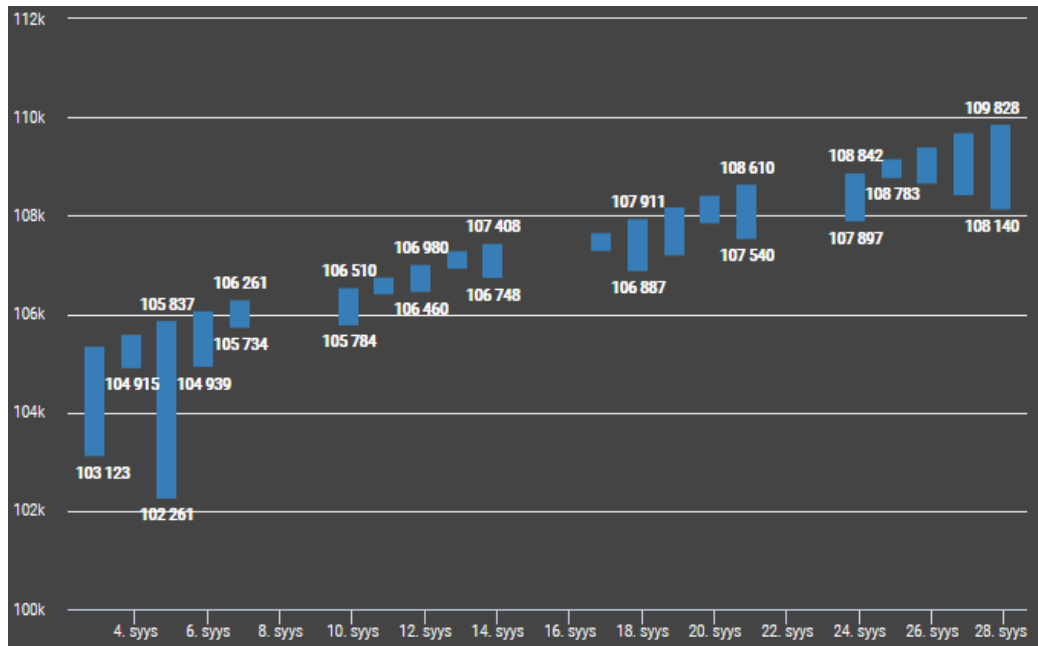
Ominaisuuden lisäämisen jälkeen, käyttäjä voi valita kaaviota luodessa tietokerrokset kaavioon aktiiviseksi valintalaatikon (checkbox) avulla. Lisäämisen jälkeen ohjelmisto täyttää kaikki edellisessä pääluvussa mainitut kriteerit (ks. kuvio 28).

Kuvio 28. Kaavion luonti toimintasuunnitelman toteuttamisen jälkeen.

## 7 Pohdinta

Monitorointi-ohjelmistossa havaittiin tutkimustuloksissa puutteita, jotka korjattiin osana tätä tutkimusta. Ohjelmiston visualisointimenetelmien kattavuus muutosten jälkeen on kuvattuna alla olevassa taulukossa (ks. taulukko 4). Huomioitavaa on myös se, että vaihtoehtoisia kaavioita ei ohjelmistoon korjattu. Tämä siksi, että vaihtoehtoiset kaaviot koettiin riittävän kattavaksi tässä tilanteessa. Bullet-kaavion osalta käytössä oleva teknologia (Vaadin) ei mahdollista bullet-kaaviota. Bullet-kaaviota vastaava mittari kuitenkin ajaa riittävässä määrin bullet-kaavion tavoitteen, joten puute ei ole merkittävä.

Laatikko-janakaavion vaihtoehtoisena toteutuksena ohjelmistossa on käytettynä vaihteluvälikaaviota, joka kertoo muuttuja suurimman ja pienimmän arvon (ks. kuvio 18). Laatikko-janakaaviossa saattaa esiintyä myös tulkintaongelmia (Koponen & Hildén 2019, 196–197), jolloin kaavion hyödyt voivat jäädä minimaaliseksi. Vaihteluvälikaavio muistuttaa, joko pylväskaaviota tai aluekaaviota (käyttäjä voi valita), joten se on helpommin tulkittavissa sen tuttuuden vuoksi (Nayak n.d.) (ks. kuvio 18 ja 29).



Kuvio 29. Vaihteluvälikaavio (pylväs)

Vaihteluvälikaaviossa pylvään ala- ja yläreunan avulla ilmaistaan vaihteluvälin ylin ja alin arvo. Tällöin pylväs havainnollistaa kahden arvon välin, minkä välille haetut arvot jakautuvat.

Taulukko 4. Arvioitavan ohjelmiston visualisointimenetelmien kattavuus kriteerien perusteella toimintasuunnitelman toteutuksen jälkeen.

|                       | Kaavio 1            | Kaavio 2          | Kaavio 3       |
|-----------------------|---------------------|-------------------|----------------|
| Muutos ajan suhteen   | Viivakaavio         | Pylväskaavio      | Bullet-kaavio  |
| Vertailu              | Pylväskaavio        | Vesiputouskaavio  |                |
| Sijoitukset           | Pylväskaavio        | Kuplakaavio       |                |
| Osuus kokonaisuudesta | Pylväskaavio (pino) | Aluekaavio (pino) | Piirakkakaavio |
| Jakauma               | Laatikko-janakaavio | Puukartta         | Tutkakaavio    |
| Korrelaatio           | Hajontakaavio       | Kuplakaavio       |                |

Visualisointimenetelmissä käytettävät ominaisuudet olivat ennen tutkimusta jo melko kattavalla tasolla, vain yksi ominaisuus oli puutteellinen. Lisäksi väriasetukset olivat virheelliset/puutteelliset. Alla olevassa taulukossa on kuvattu ohjelmiston kattavuus tämän tutkimuksen jälkeen (ks. taulukko 5). Väriasetukset jätettiin vielä ainakin toistaiseksi ennalleen, sillä eri värit koettiin tärkeämmäksi, kuin maksimaalinen selkeys värien osalta. Lisäksi ohjelmiston suhteellinen ja absoluuttinen väritys tarvitsee eri värejä sävyjen sijaan, jolloin voidaan korostaa poikkeuksellisen pieniä tai suuria arvoja.

Taulukko 5. Arvioitavan ohjelmiston visualisoinnin ominaisuuksien kattavuus kriteerien perusteella toimintasuunnitelman toteutuksen jälkeen.

|                            | Huomioita   |
|----------------------------|---|
| X-akselin nimi             | Vain x-akselin omaaville kaavioille                     |
| Y-akselin nimi             | Vain y-akselin omaaville kaavioille                     |
| Otsikko                    |   |
| Paikka lähteille           | Erillinen paikka lisäinformaatiolle                     |
| Y-akselin arvojoukon arvot | Minimi- ja maksimiarvon voi asettaa                     |
| Tietokerrokset             | Valitaan päälle tai pois                                |
| Selitteet                  | Automaattinen, kun useampi kuin yksi kaavio             |
| Trendiviiva                | Useita trendiviivatyyppisiä                             |
| Väriasetukset              | Useita värejä saatavilla, mutta ei yhden värin sävyinä  |
| Järjestys                  | SQL-komennon avulla voidaan järjestää halutulla tavalla |

Visualisointimenetelmien kriteeristötaulukkoa luodessa on helppo huomata pylväskaavion niin sanottu ylivoimaisuus, sillä kaavio sopii suurimpaan osaan erilaisista visualisointitilanteista. Pylväskaavion avulla voitaisiin visualisoida neljä kuudesta visualisointitapauksesta. Tästä huolimatta pylväskaavio ei välttämättä ole kuitenkaan aina se mielekkäin vaihtoehto, joten vaikka pylväskaavio kattaakin suuren osan visualisointitarpeista, tulisi visualisointiohjelmistosta löytyä myös muitakin vaihtoehtoja erilaisia tilanteita varten. Lisäksi huomioitavaa on se, että taulukon (ks. taulukko 1) perusteella ohjelmistoon riittäisi kolme kaaviotyyppiä. Tämäkään ei kuitenkaan ole kovin mielekästä, sillä vaikka kolmella kaaviolla pystyttäisiinkin visualisoimaan suurin osa visualisointitarpeista, tulee varmasti myös eteen tilanteita, joihin mikään näistä kolmesta kaaviosta ei sovi. Tämän takia ohjelmistosta tulisi löytyä useampia kaavioita

kuin vain minimivaatimukset. Mitä suuremman kattavuuden koko taulukosta (ks. taulukko 1) saa, sitä kattavampi ohjelmisto on ja sopii paremmin huomattavasti laajempaan skaalaan visualisointitarpeista.

Visualisointimenetelmien ominaisuudet ovat suuressa roolissa selkeän ja tehokkaan visualisoinnin luomisessa. Joitakin ominaisuuksia ei kaikissa tilanteissa tarvita (esimerkiksi akselit), mutta ominaisuuden tulisi silti olla käytettävissä, mikäli siihen ominaisuuteen liittyviä kaaviotyyppisiä ohjelmistosta löytyy. Jotkut näistä ominaisuuksia tuntuvat usein kovin itsestäänselviltä, mutta ottaessa kyseisen ominaisuuden pois, tulkinta vaikeutuukin huomattavasti. Huomioitavaa on myös se, että kaaviota luodessa tulisi aina muistaa myös kaavion tulkitsijaa. Tulkitsija ei välttämättä ole pohjustautunut mitenkään kaavioon, jolloin hänellä ei ole minkäänlaista pohjatietoa. Tällöin kaikki oleellinen tieto tulisi olla kaaviossa valmiina.

## 7.1 Tutkimuksen luotettavuus

Kuten aiemmissa luvuissa on sanottukin, visualisointimenetelmiä on lukemattomia määriä. Tämän perusteella ei voidakaan sanoa täsmällisesti, mikä kaavio tiettyyn tilanteeseen on ehdottomasti paras. Lisäksi kohdeyleisö vaikuttaa kaavion valintaan, jolloin toinen kaavio sopii paremmin toiselle kohdeyleisölle kuin joku toinen kaavio. Lisäksi kaaviotyyppihin liittyy aina henkilökohtaisia mielipiteitä niiden paremmuudesta ja mitä ominaisuuksia tai värejä kaaviolla tulisi olla.

Tutkimuksessa on pyritty hakemaan erilaisia mielipiteitä erilaisista näkökulmista mahdollisimman laajan näkökulman saamiseksi. Tutkimuksessa on otettu kantaa kaikkein yleisimpiin kaavioihin (tilastolliset kaaviot), joita käytetään yleisesti kaikkialla, joten kaaviotyyppit ovat enimmäkseen tuttuja kaikille. Jotkin monimutkaiset visualisointimenetelmät saattavat soveltua tiettyihin tarpeisiin paremmin, mutta ne eivät ole yleisesti lukijoiden tiedossa, jolloin näiden tulkitseminen on vaikeaa, mikä sotii visualisointimenetelmän ydinajatusta vastaan: kaavion tulee olla mahdollisimman selkeä. Näiden syiden ja perusteluiden valossa voidaan todeta, että tutkimuksen tulokset ovat luotettavia, kun tarkastelu kohdennetaan vain yleisimpiin menetelmiin, eli tilastollisiin kaavioihin.

## 7.2 Kehitysmahdollisuudet

Sovellukseen jäi vielä kaksi kaaviotyyppiä ja yksi visualisointiin liittyvä ominaisuus, jotka olivat korvaavilla tai erilaisella tavalla toteutettuja. Näiden vaihtoehtojen toteuttamisella saataisiin ohjelmistoon 100 % kattavuus kriteerien perusteella. Toisaalta toinen näistä kaaviotyypeistä ei ole toteutettavissa nykyisellä käytössä olevalla teknologialla.

Visualisointimenetelmistä löytyy lukemattomia erilaisia teoksia erilaisista näkökulmista. Tästä syystä tutkimusta voisi jatkaa, hankkimalla vielä enemmän tietoa edellä käsitellyistä visualisointimenetelmistä tai niiden hyödyistä saaden vielä kattavamman kuvan visualisointitavoista. Toisaalta jatkotutkimuksessa voisi perehtyä myös esimerkiksi monimutkaisempiin tai erilaisiin visualisointitapoihin ja niiden mahdollisuuksiin. Hieman erilaisen näkökulman tutkimusaiheeseen voisi saada myöskin asiantuntija-haastatteluilla, tai esimerkiksi luomalla kyselyn, jossa pyydetään osallistujaa valitsemaan kaavioita selkeyden ja tehokkuuden perusteella. Tällaisesta tutkimuksesta saataisiin parhaiten kaavioiden lukijoiden mielipiteitä otettua huomioon.

Toisaalta taas luonnollinen jatkumo tälle tutkimukselle, olisi tehdä tutkimus datan analysoinnista ja minkälaisen datan analysoinnista olisi eniten hyötyä. Lisäksi datan visualisoinnin ja sitä kautta datan analysointiin liittyvien hyötyjen tutkimus auttaisi yrityksiä kehittämään liiketoimintaansa datan visualisoinnin avulla.

Kaaviotyyppejä voi myös aina luoda lisää, mikäli käytetty teknologia vain tukee vielä uusia kaaviotyyppejä. Mitä kattavampi kaaviopankki, sitä kattavampi visualisointiohjelmisto on erilaisiin tarkoituksiin. Lisäksi suurempi kaaviopankki takaisi jokaisen käyttäjän tyytyväisyyden vaihtoehtoihin omien mieltymystensä perusteella.

## Lähteet

Roima Int. N.d. Roima Intelligence OY. Viitattu 28.4.2019. <https://www.roimaint.fi>.

Kananen, J. 2008. Kvalitatiivisen tutkimuksen teoria ja käytänteet. Jyväskylä:

Jyväskylän yliopistopaino 2008.

Markkula, Y. & Syväniemi, A. 2015. Analytiikkamatka: datasta tietoon ja tiedolla johtamiseen. Helsinki: Suomen Liikekirjat 2015.

Koponen, J. & Hildén, J. 2019. Data visualization handbook. Helsinki: Aalto University 2019.

Salo, I. 2013. Big Data – Tiedon vallankumous. Jyväskylä: Docendo 2013.

Väre, T. 2019. Master Data. Helsinki: Alma Talent 2019.

Salo, I. 2014. Big Data & Pilvipalvelut. Jyväskylä: Docendo 2014.

Nayak, A. N.d. Introduction to Data Visualization. Viitattu 5.4.2019. <https://www.udemy.com/introduction-to-data-visualization/>.

Data visualization beginner's guide: a definition, examples, and learning resources. N.d. Julkaisu Tableau – sivustolla. Viitattu 5.4.2019. <https://www.tableau.com/learn/articles/data-visualization>.

The Data Visualization Handbook. N.d. Julkaisu SAP – sivustolla. Viitattu 8.4.2019. <https://www.sap.com/documents/2018/04/8cb8821e-fc7c-0010-87a3-c30de2ffd8ff.html>.

IDRC. N.d. 10 Data Visualization Tips. Viitattu 15.4.2019. <https://www.idrc.ca/sites/default/files/sp/Documents%20EN/10-data-visualization-tips-en.pdf>.

Rabah, S. N.d. UX Color Theory – Applying Color Knowledge to Data Visualization. Viitattu 15.4.2019. [http://cdn2.hubspot.net/hub/111084/file-708877165-pdf/docs/ebooks/eBook-UX-Color-Theory\\_Applying-Color-Knowledge-to-Data-Visualization.pdf](http://cdn2.hubspot.net/hub/111084/file-708877165-pdf/docs/ebooks/eBook-UX-Color-Theory_Applying-Color-Knowledge-to-Data-Visualization.pdf).

Ficek, A. 2015. Data Visualization Techniques – How to Tell Your Story to Make it Stick. Viitattu 15.4.2019. <https://www.cehd.umn.edu/olpd/mesi/spring/2015/ficek-datavis.pdf>.

HubSpot. N.d. Data Visualization 101: How to Design Charts and Graphs. Viitattu 15.4.2019. [https://cdn2.hubspot.net/hub/53/file-863940581-pdf/Data\\_Visualization\\_101\\_How\\_to\\_Design\\_Charts\\_and\\_Graphs.pdf](https://cdn2.hubspot.net/hub/53/file-863940581-pdf/Data_Visualization_101_How_to_Design_Charts_and_Graphs.pdf).

SAS. N.d. Data Visualization – What it is and why it matters. Viitattu 15.4.2019. [https://www.sas.com/en\\_us/insights/big-data/data-visualization.html](https://www.sas.com/en_us/insights/big-data/data-visualization.html).

- Deloitte. 2017. Data visualization: Why a picture can be worth a thousand clicks. Viitattu 15.4.2019. [https://www2.deloitte.com/content/dam/Deloitte/za/Documents/za\\_DeloitteZA\\_CFO\\_Insights\\_data\\_visualisation\\_Nov17.pdf](https://www2.deloitte.com/content/dam/Deloitte/za/Documents/za_DeloitteZA_CFO_Insights_data_visualisation_Nov17.pdf).
- Yin, M., Hollender, D., Condelli, L., Shewitz, D., Duffy, A. & Movit, M. 2014. The Power of Data Visualization: Advanced Presentations of NRS Data. Viitattu 15.4.2019. [https://nrsweb.org/sites/default/files/Data\\_Visualization\\_Guide\\_508.pdf](https://nrsweb.org/sites/default/files/Data_Visualization_Guide_508.pdf).
- Vaadin. N.d. Viitattu 4.5.2019. <https://vaadin.com>.