



Tietojärjestelmän käyttöliittymän opittavuus

Oulun yliopisto
Tieto- ja sähkötekniikan tiedekunta
Tietojenkäsittelytiede
LuK-tutkielma
Tommi Kyllönen
2023

Tiivistelmä

Tietojärjestelmät ovat enemmän ja enemmän osa meidän arkipäiväistä elämäämme. Yksi tärkeimmistä asioista tietojärjestelmissä on se, että me ihmiset oppisimme käyttämään niitä. Tässä tutkielmassa selvitettiin miten tietojärjestelmästä löytyvät ominaisuudet vaikuttavat sen opittavuuteen. Käytettävyyteen ja opittavuuteen etsittiin sopivia aiempia tutkimuksia kirjallisuuskatsauksen avulla ja samalla vastattiin seuraavanlaiseen tutkimuskysymykseen: Mitkä tekijät vaikuttavat tietojärjestelmien käyttöliittymien opittavuuteen?

Eri tutkimusten avulla tietojärjestelmän käyttöliittymän opittavuudelle löydettiin viisi keskeistä ominaisuutta, joita hyödyntämällä tietojärjestelmän käyttöliittymän opittavuuden toteutumista voitiin parantaa. Osa näistä ominaisuuksista rajoittui ihmisen fyysisiin ominaisuuksiin ja osalla pyrittiin takaamaan yksinkertainen ja helposti seurattava kokonaisuus käyttöliittymässä. Kirjallisuuden avulla löydettiin myös opittavuuden esteitä, jotka toteutuessaan vaikuttivat heikentävästi, niin käyttöliittymän ymmärtämiseen, kuin tietojärjestelmän suosioon.

Avainsanat

Käytettävyys, opittavuus

Ohjaaja

FT, yliopistonlehtori Mikko Rajanen

Sisällysluettelo

Tiivistelmä.....	2
Sisällysluettelo	3
1. Johdanto.....	4
2. Tutkimusmenetelmä	5
3. Aiempi tutkimus	6
3.1 Tietojärjestelmän käyttöliittymän käytettävyyden perusteet	6
3.2 Tietojärjestelmien opittavuus	7
4. Pohdinta	13
5. Yhteenveto.....	17
Lähteet.....	19

1. Johdanto

Tietojärjestelmät kasvattavat jatkuvasti osuuttaan ihmisten arkielämässä. Suurin tähän vaikuttava tekijä on se, että internetiä käyttää jopa 90 % taloudellisesti kehittyneiden maiden asukkaista (Morley et al., 2018). Tietojärjestelmiä kehitettäessä on tärkeää toteuttaa niille annetut tekniset ja toiminnalliset ominaisuudet siten, että ne toimivat halutulla tavalla. On myös tärkeää miettiä mihin tietojärjestelmää käytetään. Etenkin, jos tietojärjestelmän käyttäjänä on ihminen, jolle tulee luoda erillinen käyttöliittymä toimintaa varten, niin käyttäjän vaatimukset tulisi olla teknisten ja toiminnallisten vaatimusten ohella yhtä tärkeitä (Maguire, 2001).

Tietojärjestelmistä puhuttaessa esiin nousee usein niiden käytettävyys, eli kuinka helppoa sen käyttäminen käyttäjälle todellisuudessa on (McNamara & Kirakowski, 2005). Ihmiset käyttävät mielellään niitä tietojärjestelmiä, joiden käytön he kokevat helpoksi. Jokainen ihminen on kuitenkin erilainen, joten saman tietojärjestelmän käyttäminen voi toisesta henkilöstä tuntua helpolta ja toisesta taas vaikealta (Ko et al., 2004).

Tutkielman tarkoituksena on tarkastella, miten tietojärjestelmän käyttöliittymästä löytyvät elementit ja ominaisuudet vaikuttavat tietojärjestelmän käytettävyyteen ja edelleen sen opittavuuteen. Olemme tottuneet käyttämään tietojärjestelmiä, mutta emme tule ajatelleeksi, mitä kaikkea niiden kehityksessä tulee ottaa huomioon, jotta niiden käyttö olisi mahdollisimman sujuvaa. Siksi tämän tutkielman avulla on tarkoitus tuoda esille, niin aiheesta kiinnostuneille, kuin alaa opiskeleville, kuinka laaja kokonaisuus käytettävyys käyttöliittymissä on. Etenkin keskityttäessä opittavuuteen keinot, joilla tietojärjestelmän opittavuutta voidaan kehittää, on hyvä tiedostaa, jotta voi mahdollisuuksien mukaan tarjota asiakkaille sujuvamman käyttökokemuksen. Konkreettisenä tutkimusongelmana siis toimii, miten tietojärjestelmän käyttöliittymän ominaisuudet vaikuttavat sen opittavuuteen. Tutkielmassa pyrin vastaamaan yhteen tutkimuskysymykseen, jonka avulla tutkimusongelma ratkaistaan: Mitkä tekijät vaikuttavat tietojärjestelmien käyttöliittymien opittavuuteen?

Luvussa kaksi esittelen tutkielman tekoon käytetyn tutkimusmenetelmän. Kolmannessa luvussa käsittelen määritellyn aiheen aiempien tutkimusten avulla. Siinä perehdyn tietojärjestelmien käyttöliittymiin ja niiden käytettävyyteen. Lisäksi perehdyn tietojärjestelmien opittavuuteen, sekä sen tärkeyteen. Neljännessä luvussa pohdin aiempien tutkimusten löydöksiä peilaten tutkimuskysymykseen. Lopuksi viidennessä luvussa teen yhteenvedon tuloksista.

2. Tutkimusmenetelmä

Tutkielman tutkimusmenetelmänä käytetään kirjallisuuskatsausmenetelmää. Valittu kirjallisuuskatsausmenetelmä on kokoava eli integroiva kirjallisuuskatsaus. Kokoavassa kirjallisuuskatsauksessa aiemmista tutkimuksista kerättävä tieto rajataan kriittisesti vastaamaan kirjallisuuskatsauksen tekijän aihetta. Lopputuotoksena saadaan uusia näkökulmia sisältävä teksti (Torraco, 2016).

Tämän tutkielman aineisto on kerätty useista hakukoneista ja hakutietokannoista löytyvästä kirjallisuudesta. Google Scholar ja Scopus toimivat pääasiallisina lähteinä aiheeseen liittyvän kirjallisuuden haussa. Lisäksi kirjallisuutta on haettu ACM Digitalin tietokannoista ja IEEE Xploresta.

Relevantin kirjallisuuden löytämiseen käytettiin hakusanoja ”information system usability”, ”software usability”, ”learnability attributes”, ”what is usability”, ”learnability”, ”what is learnability” ja ”learnability in information systems”. Tuloksia haettaessa rajasin hakutuloksia tarpeen mukaan tietojenkäsittelytieteiden alle, jolloin hakutuloksista löytyi aihetta vastaavaa kirjallisuutta tarkemmin kuin laajemmalla haulilla. Laajemmalla haulilla oli myös omat hyvät puolensa, sillä sen avulla aihetta varten etsittävä tietoa pystyi tarkastelemaan hieman kontekstin ulkopuolelta, joka toi uusia ideoita kirjallisuuden haussa. Kirjallisuuden avulla oli tarkoitus selvittää kuinka tietojärjestelmien käytettävyys vaikuttaa niiden opittavuuteen. Empiiristä tutkimusaineistoa ei kerätä.

3. Aiempi tutkimus

Tässä luvussa tarkastellaan aiheeseen liittyvää kirjallisuutta, etenkin käyttöliittymien, käytettävyyden ja opittavuuden pohjalta.

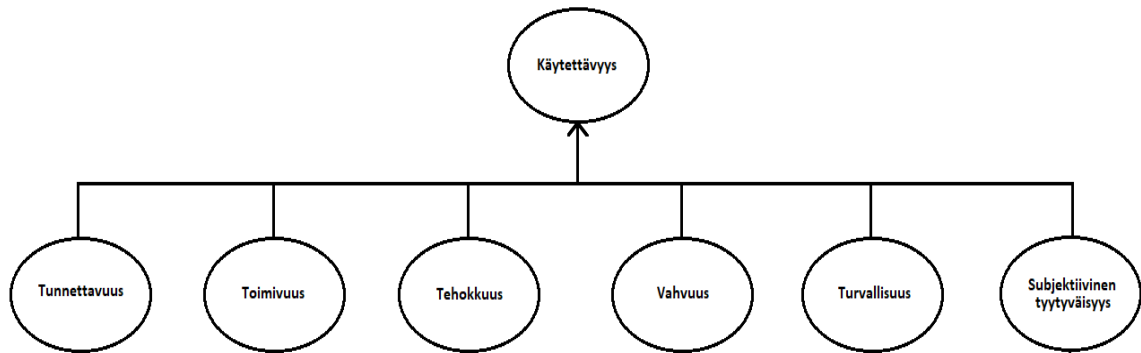
3.1 Tietojärjestelmän käyttöliittymän käytettävyyden perusteet

Käytettävyys on laaja käsite, sillä sitä ei voida yksinkertaisesti mitata millään mittarilla, joka tekee sen mittaamisesta monimutkaista (Lewis, 2014). Tietojärjestelmän käyttöliittymän tulee siis olla helppo ja miellyttävä käyttää, sillä muuten vaarana on asiakkaan kyllästyminen vaikeakäyttöiseen järjestelmään, jonka myötä käyttäjämäärä vähenee (Haramundanis, 2001).

ISO-standardit on luotu yhdenmukaistamaan asioiden tekemistä, ja ne ovat alan asiantuntijoiden hyväksymiä menetelmiä (International Organization for Standardization, n.d.). Käytettävyys määritellään ISO 9241-11:2018 standardin mukaan vaikuttavuutena, tehokkuutena ja tyytyväisyytenä, jotka käyttäjä voi saavuttaa järjestelmää, tuotetta tai palvelua hyödyntäessään tietyssä käyttöympäristössä (International Organization for Standardization, 2018).

Käytettävyys ei ole tietojärjestelmän käyttöliittymän yksinkertainen ominaisuus, vaan se koostuu viidestä käytettävyysominaisuudesta, jotka ovat opittavuus, tehokkuus, muistettavuus, virheettömyys, sekä tyytyväisyys (Nielsen Jakob, 1994, p. 26; Holzinger, 2005). Opittavuus tarkoittaa järjestelmän käytön oppimisen helppoutta, niin että käyttäjä pystyy suorittamaan tekemänsä tehtävän nopeasti (Nielsen Jakob, 1994, p. 26). Tehokkuus puolestaan on opitun järjestelmän tehokasta ja tuloksellista käyttöä. Muistettavuudella pyritään takaamaan käyttäjän sujuva toimiminen järjestelmän kanssa, vaikka edellisestä käyttökerrasta olisikin jo aikaa (Nielsen Jakob, 1994, p. 26). Virheettömyydellä pyritään minimoimaan käyttäjän mahdollisuudet virheiden tekemiseen järjestelmää käytettäessä ja samalla mahdollisten virheiden korjaaminen tai niistä palautuminen tulisi olla helppoa. Tyytyväisyydellä tarkoitetaan käyttäjän saamaa mielihyvää järjestelmän käytön aikana (Nielsen Jakob, 1994, p. 26). Kun jotain keskeistä käytettävyysominaisuutta muutetaan parempaan suuntaan voi jonkun toisen käytettävyysominaisuuden vaikutus järjestelmässä mennä huonompaan suuntaan. Lisäksi joskus on myös hankala arvioida missä tärkeysjärjestyksessä käytettävyysominaisuuksia lähdetään järjestelmään toteuttamaan (Santoso & Schrepp, 2018).

Käytettävyys on vuorovaikutuskeskeinen ominaisuus ainakin kahden eri ympäristön välillä, esimerkiksi ihminen ja tietokone (Lewis, 2014). Käytettävyys voidaan myös jakaa kahteen määritelmään summatiiviseen ja formatiiviseen arviointiin. Summatiivinen käytettävyys arviointi keskittyy mittareilla mitattavaan käytettävyyteen, ja sen mukaan tuote on käytettävyydeltään hyvä, jos se sille tarkoitettussa käytössä toimii tehokkaasti ja tuottaa tyytyväisyyttä. Formattiivinen käytettävyys arviointi puolestaan keskittyy käytettävyysongelmiin, kuten niiden havaitsemiseen, ehkäisemiseen ja poistamiseen. Tällöin tuotteen käytettävyys koetaan sitä paremmaksi mitä vähemmän siitä on löydettävissä käytettävyysongelmia (Lewis, 2014). Grossman et al., (2009) esittävät summatiivisen käytettävyyden arviointina, jossa tietojärjestelmän käytettävyyttä verrataan kilpailevaan järjestelmään tai sille asetettuihin käytettävyyden vaatimuksiin. Formattiivisen käytettävyyden arvioinnin kohdalla pyritään parantamaan järjestelmän käyttöliittymää havaittujen käytettävyysongelmien avulla (Grossman et al., 2009).



Kuva 1. Käytettävyysominaisuuksien taksonomia (Muokattu Alonso-Ríos et al., 2010 kuvasta 1)

Alonso-Ríos et al., (2010) ovat määritelleet tietojärjestelmän käytettävyysominaisuuksien taksonomian, jonka avulla pystytään kattamaan minkä tahansa tietojärjestelmän käytettävyysnäkökohdat erilaisissa käyttökonteksteissa (Kuva 1). Luodun taksonomian tarkoituksena on selkeyttää käytettävyyden määrittelyssä löytyviä attribuutteja ja jakaa niitä eri tasoille helpottamaan niiden arviointia ja käytettävyyttä. Lisäksi tämän taksonomian hyöty nähdään käytettävyystutkimuksissa, jossa käytettävyysattribuuttien päällekkäisyys tietojärjestelmän muiden ominaisuuksien kanssa saadaan estettyä (Alonso-Ríos et al., 2010). Tunnettavuudella pyritään saamaan järjestelmästä käyttäjälle ymmärrettävä, opittava ja muistettava. Toimivuudella pyritään tarjoamaan käyttäjälle tarvittavat toiminnallisuudet ja helpottamaan erilaisten käyttäjien mukautumista järjestelmän käyttöön (Alonso-Ríos et al., 2010). Tehokkuuden avulla satsatut resurssit pyritään tarjoamaan tarkoituksenmukaisina tuloksina. Vahvuutta käytetään virheiden torjumiseen (Alonso-Ríos et al., 2010). Turvallisuudella tarkoitetaan järjestelmän käytöstä aiheutuvien riskien ja vahinkojen välttämistä. Subjekttiivisen tyytyväisyyden tarkoituksena on tuottaa käyttäjälle kiinnostusta järjestelmää kohtaan (Alonso-Ríos et al., 2010).

Tarkasteltaessa muita määritelmiä käytettävyyden attribuuteille havaitaan neljä useimmin esiintyvää attribuuttia: tyytyväisyys, tehokkuus, opittavuus ja vaikuttavuus. Näistä kahdelle ensimmäiselle löytyy oma paikkansa taksonomian ylimmältä tasolta, mutta kaksi seuraavaa ovat osana tunnettavuus- ja toimivuuskäsitteitä, joissa niiden määrittelystä saadaan entistä kokonaisvaltaisempi (Alonso-Ríos et al., 2010).

3.2 Tietojärjestelmien opittavuus

Opittavuus on yksi käytettävyyden viidestä käytettävyysominaisuudesta. Opittavuus tarkoittaa järjestelmän käytön oppimisen helppoutta, niin että käyttäjä pystyy suorittamaan tekemänsä tehtävän nopeasti (Nielsen Jakob, 1994, p. 26). Rafique et al., (2012) puolestaan määrittelevät opittavuuden käyttäjän helppoudeksi suoriutua järjestelmän perustehtävistä ensimmäisellä käyttökerralla. Opittavuutta pidetään joissakin tapauksissa tärkeimpänä käytettävyyden käytettävyysominaisuuksista (Nielsen Jakob, 1994, p. 27). Tietojärjestelmien kohdalla opittavuus lähtee aina liikkeelle tietämättömyydestä järjestelmän käyttöä kohtaan. On toki mahdollista, että käyttäjä hyödyntää jotain jo muista tietojärjestelmistä omaksuttuja taitoja tämän kyseisen

tietojärjestelmän käyttämiseen. Tällöin lähtökohta oppimiselle on parempi kuin jollakin toisella järjestelmän käyttäjällä (Nielsen Jakob, 1994, p. 28). Assimilaatioteorian mukaan menestys ongelmanratkaisussa pohjautuu mielekkääseen oppimiseen. Tällainen oppimistapahtuma syntyy, kun uutta opittaessa henkilö pystyy muodostamaan yhteyksiä uusien tietojen, sekä pitkäkestoisesta muistista löytyvien tietojen välille (Davis & Wiedenbeck, 1998). Tämän teorian mukaan on myös helpompaa oppia käyttämään uutta tietojärjestelmää, jos pystyy luomaan yhteyksiä jo aiemmin samankaltaisen tietojärjestelmän toiminnallisuuksiin.

Opittavuutta pidetään käytettävyyden helpoimpana mitattavana käytettävyyssominaisuutena. Tämä johtuu siitä, että on helppo sanoa, onko käyttäjä oppinut tietojärjestelmän käytön, jos mitattavaksi kohteeksi on valittu esimerkiksi tietyn tehtävän tekemiseen kulunut aika (Grossman et al., 2009). Kun opittavuudelle voitiin määritellä summatiivinen ja formatiivinen arviointimenetelmä, on se mahdollista myös opittavuudelle. Summatiivisella arvioinnilla luodaan kokonaisarvio järjestelmän oppimiskelpoisuudesta. Formatiiivinen arviointi keskittyy oppimisongelmien paljastamiseen (Grossman et al., 2009). Tietojärjestelmällä voi olla monitasoisia käyttäjiä, joten silloin on hyvä eritellä opittavuutta mitattaessa eri lähtötasossa olevat käyttäjät esimerkiksi aiemmin paljon tietojärjestelmiä käyttäneet henkilöt ja aiemmin vähän tietojärjestelmiä käyttäneet henkilöt (Nielsen Jakob, 1994, p. 29). On hyvä muistaa, että oppimistilanteen tulee olla käyttäjälle mieluinen opittavan asian lisäksi. Osa käyttäjistä nostaa tietyissä tilanteissa tunteen omistuneesta järjestelmän oppimisesta korkeammalle kuin konkreettisen opitun taidon (Vujošević et al., 2019). Tietojärjestelmää käytettäessä tulee eteen useammin suoritettavia tehtäviä ja harvemmin suoritettavia tehtäviä. Vaikka suunnittelu vaiheessa tuleekin varmistua siitä, että useasti suoritettavat tehtävät on helppo suorittaa, niin on myös varmistuttava harvemmin suoritettavien tehtävien helposta opittavuudesta (Benyon, 2019, p. 33).

Grossman et al., (2009) ovat luoneet viisi aspektia, jotka tulisi ottaa huomioon tietojärjestelmän opittavuutta arvioitaessa ja suunniteltaessa. Nämä aspektit ovat tehtävien kulku, tietoisuus, paikannus, ymmärtäminen ja siirtymät. Tehtävien kulussa pyritään varmistamaan se, että käyttäjä pystyy löytämään tapahtumasarjat tietyn toiminnallisuuden suorittamiseksi. Jos tämä on puutteellista, niin on mahdollista, että käyttäjä tietää mitä tulisi järjestelmällä tehdä, mutta ei tiedä mistä aloittaa (Grossman et al., 2009). Tietoisuudella pyritään tuomaan käyttäjälle tieto siitä, mitä kaikkia työkaluja tai toimintoja järjestelmästä löytyy. Jos eri työkalut ovat piilotettuna monien toimintojen taakse, voi se vaikeuttaa kokonaisvaltaista järjestelmän käyttöä ja oppimista (Grossman et al., 2009). Paikantamisessa on kyse käyttäjän kyvystä paikantaa järjestelmästä löytyvät toiminnot. Hyvin tehdyn dokumentaation avulla pystytään merkittävästi vähentämään tämän aspektin negatiivista vaikutusta järjestelmän opittavuuteen (Grossman et al., 2009). Ymmärtämisen kohdalla käyttäjä on tietoinen ja pystyy paikantamaan työkalun tai toiminnon, mutta ei ymmärrä miten sitä käytetään. Ymmärtämistä voidaan tehostaa esimerkiksi apujärjestelmien, tai dokumentaatiosta löytyvien media elementtien avulla (Grossman et al., 2009). Siirtymät kuvaavat käyttäjän valintoja käyttäen toiminnallisuuksia eri tavoilla. Järjestelmät usein tarjoavat eri keinoja käyttää toiminnallisuuksia pohjautuen käyttäjän taitotasoon. Esimerkiksi jonkin toiminnallisuuden voi suorittaa näppäinyhdistelmillä, tai perinteisesti klikkailemalla. Siirtymien avulla etenkin tehokkuus järjestelmän käyttöä kohtaan kasvaa ja siksi on ehdotettu opittavuuden helpottamiseksi vähitellen esiin tulevia tapoja, jotka nopeuttavat ja tehostavat toiminnallisuuksia (Grossman et al., 2009).

Tietojärjestelmän design muotoilulla on merkitystä sen kokonaisvaltaisessa opittavuudessa. Tärkeää on pyrkiä käyttämään johdonmukaisia elementtejä designin

muotoilussa, koska tällöin käyttäjän ei tarvitse käyttää aikaa monimutkaisten muotoilujen tulkittamiseen, järjestelmän käyttämisen oppimisen sijaan. Näihin elementteihin lukeutuvat esimerkiksi napit ja liukukytkimet (Benyon, 2019, p. 69). Design muotoilujen rinnalle voidaan määritellä neljä suunnitteluperiaatetta, joiden avulla pyritään parantamaan opittavuutta tietojärjestelmässä. Nämä suunnitteluperiaatteet ovat näkyvyys, johdonmukaisuus, tuttuus ja affordanssi (Benyon, 2019, p. 117). Näkyvyysperiaatteessa pohjaututaan psykologiseen periaateeseen, jonka mukaan on helpompaa tunnistaa asiat kuin muistaa. On siis tärkeää tuoda näkyville saatavilla olevat asiat eri toiminnallisuuksien suorittamiseen. Jos näkyviin tuominen ei ole mahdollista myös havainnollistaminen on hyvä keino (Benyon, 2019, p. 117). Johdonmukaisuus on vahvasti sidonnainen aiemmin esitettyyn design muotoiluun, mutta on myös tärkeässä roolissa esimerkiksi kielen käytössä tietojärjestelmässä. Käsitteellisyys ja fyysiset elementit tulee olla johdonmukaisesti toteutettu opittavuuden takaamiseksi (Benyon, 2019, p. 117). Tuttuudella pyritään tarjoamaan käyttäjälle tietojärjestelmän sisäinen kieli ja symbolit mahdollisimman helposti tunnistettavina. Tilanteessa, jossa joudutaan käyttämään tuntemattomampia ilmauksia ja symboleja, on metaforien avulla asian tutummaksi muuttaminen hyvä keino. Johdonmukaisuudella siis pyritään pitämään esitetyt elementit ja kieli samassa linjassa monimutkaisuuden välttämiseksi, kun tuttuudella pyritään esittämään ilmaukset ja symbolit tunnistettavalla tavalla epäselvyyksien välttämiseksi (Benyon, 2019, p. 117). Affordanssilla tarkoitetaan asioiden suunnittelua niin, että käyttäjän on helppo ymmärtää mihin kyseistä asiaa käytetään. Eri asioilla on kulttuurin mukaan vakiintuneet käyttötavat ja toiminnallisuudet, jolloin ihmiset olettavat automaattisesti näiden asioiden toimivan opitulla halutulla tavalla. Esimerkiksi elementtien muotoilu napeiksi saa käyttäjät painamaan niitä, ilman tätä asiayhteyttä käyttäjä ei välttämättä ymmärrä mikä funktio tällä elementillä on (Benyon, 2019, p. 117).

Tietojärjestelmän opittavuudelle on löydettävissä kuusi esteettä, jotka ovat suunnittelun, valikoiman, koordinoinnin, käytön, ymmärtämisen ja tiedon esteet. Kun loppukäyttäjä kohtaa uutta tietojärjestelmää opitellessaan esteen oppimiselle punnitsee hän usein, onko ratkaisun etsiminen kannattavaa vai ei (Ko et al., 2004). Jos loppukäyttäjä kokee, että epäonnistuminen yrittäessä löytää ratkaisua esteeseen ei ole suuri riski, pyrkii hän vielä opettelemaan järjestelmän käyttöä, muuten mahdollisuutena on, että hän vaihtaa johonkin toiseen samantapaiseen järjestelmään (Ko et al., 2004). Suunnittelun esteet syntyvät, kun järjestelmää kehitetään ilman huolellista suunnittelua. Lisäksi täysin uuden koodin luominen jo olemassa olevan käytön sijaan on haitaksi, koska se lisää virheiden riskiä toiminnallisuuksien toiminnassa (Ko et al., 2004). Valikoiman esteiden olemassaolon syyksi nousee useiden samankaltaisten tietojärjestelmien tarjoamat käyttömahdollisuudet ilman asianmukaista käytön opastusta. Jos kaksi samankaltaista järjestelmää tarjoavat samanlaisia käyttömahdollisuuksia, toinen paremmalla opastuksella ja toinen huonommalla, voi loppukäyttäjä ilman syvempää perehtymistä valita oman oppimisen kannalta väärän järjestelmän, jolloin kallisarvoista järjestelmän käytön oppimisen aikaa menee hukkaan (Ko et al., 2004). Koordinoinnin esteiksi nousevat usein odottamattomat yhdistelmät eri toimintojen välillä. Kun loppukäyttäjä odottaa tiettyä lopputulosta todellinen lopputulos ei vastaakaan odotuksia (Ko et al., 2004). Käytön esteet muodostuvat tietämättömyydestä käyttöliittymää kohtaan, sen käyttämisestä ja sen käyttäytymisestä. Osalle järjestelmiä annetaan tueksi ohjekirjoja (Ko et al., 2004). Ymmärtämisen esteet tulevat eteen järjestelmän tuottaman informaation aikana. Jos loppukäyttäjä tekee jonkin virheen toiminnassaan, niin järjestelmän tulisi pystyä ilmaisemaan ratkaisu onnistuneen suorituksen tekemiseen, niin että loppukäyttäjä ymmärtää sen (Ko et al., 2004). Tiedon esteet koostuvat loppukäyttäjän hankaluudesta

saada tietoa järjestelmän toiminnasta. Esimerkiksi jonkin toiminnallisuuden löytyminen useasta eri paikasta voi vahvistaa tiedon estettä (Ko et al., 2004).

Tietojärjestelmien opittavuutta tarkasteltaessa on löydettävissä viisi keskeistä ominaisuutta. Nämä ominaisuudet kytetään usein käyttöliittymästä löytyviin materiaaleihin (Haramundanis, 2001). Näiden ominaisuuksien mukaan tietojärjestelmän käyttöliittymän materiaalien tulee siis olla muistettavia, loogisia, rekonstruoitavia, johdonmukaisia ja visuaalisia. Näitä ominaisuuksia oikealla tavalla hyödyntäen osana tietojärjestelmän käyttöliittymän kehitystä saadaan käyttöliittymästä helposti opittava (Haramundanis, 2001). Muistettavuutta tarkastellessa esiin nousee ihmisen fyysiset rajoitteet muistamiselle. Ihmisen muisti voidaan jakaa kolmeen tasoon lyhytaikaiseen, keskipitkään aikaväliin ja pitkään aikaväliin. Mitä pidempään asia tulee muistaa sitä, merkittävämpi se on järjestelmän toiminnassa (Haramundanis, 2001). Tämän fyysisen rajoitteen vuoksi käyttöliittymistä tulisi tehdä yksinkertaisia, jotta ne olisivat mahdollisimman mieleenpainuvia. Jos käyttöliittymästä löytyy pitkiä etenemisketjuja tietyn tehtävän suorittamiseen, voi se olla käyttäjälle liian pitkälle muistettava, jolloin se heikentää järjestelmän opittavuutta (Haramundanis, 2001). Loogisuudessa on kyse siitä, miten helposti käyttöliittymästä on havaittavissa askel askeleelta eteneviä selkeitä tapoja suorittaa haluttu tehtävä. Rekonstruoitavuus puolestaan taataan sillä, että tieto mitä käyttäjä käyttöliittymästä saa tietyn tehtävän suorittamista varten on tarpeeksi johdonmukainen, joten tämä toimintaketju on helposti rekonstruoitavissa, eli uudelleen muodostettavissa hänen mielessään seuraavalla kerralla, eikä erillistä tarkastuslistaa tehtävän suorittamiseen tarvita (Haramundanis, 2001). Johdonmukaisuus on jaettavissa kahteen osaan sisäiseen johdonmukaisuuteen ja ulkoiseen johdonmukaisuuteen. Sisäisessä johdonmukaisuudessa on tarkoituksena käyttää yleisiä käsitteitä ja tyyliä tietojärjestelmässä, jotta käyttäjän ei tarvitse kokea järjestelmän käytössä haasteelliseksi esimerkiksi käytettyjä sanavalintoja. Tämä helpottaa tiedonkulkua käytön yhteydessä käyttäjälle (Haramundanis, 2001). Ulkoisessa johdonmukaisuudessa on kyse järjestelmän käyttäjälle tuottamasta informaatiosta. Tämä tarkoittaa sitä, että esimerkiksi järjestelmän tuottamat tulokset tulee olla samanlaisessa muodossa joka puolella järjestelmää. Visuaalisuus on käyttöliittymän opittavuudelle tärkeä ominaisuus, koska näkö on aisteista tärkein oppimisen kannalta (Haramundanis, 2001).

Pretorius & van Biljon (2010) ovat tutkimuksessaan tarkastelleet kuinka ICT-aidot vaikuttavat tietojärjestelmän opittavuuteen. Tutkimuksen kohdejärjestelmänä on oppimisen hallintajärjestelmä, jota arvioivat avoimen etäyliopiston opiskelijat. Tutkimukseen valittiin 23 opiskelijasta 10, joista viisi on miehiä ja viisi naisia. Heidät on myös jaettu kahteen ryhmään ICT-taitojen perusteella ja näistä ryhmistä käytetään nimityksiä ICT-asiantuntijaosallistujat ja ei-asiantuntijat. Osallistujilla on myös aiempaa kokemusta kyseisen järjestelmän käyttämisestä.

	Tehtävä 1 Lähetä word-tiedosto	Tehtävä 2 Lähetä PDF-tiedosto	Tehtävä 3 Lähetä word-tiedosto	Tehtävä 4 Lähetä word-tiedosto uudelleen
Minimi	27	27	10	13
Maksimi	161	237	63	87
Keskiarvo	68.67	115.60	34.70	37.60

Taulukko 1. Tutkimuksen tehtäväkohtaiset suoritusajat (Muokattu Pretorius & van Biljon, 2010 taulukosta 2)

Pretorius & van Biljon (2010) sisällyttivät tutkimukseensa neljä eri tehtävää, joiden suoritusaikojen ja virheiden määrän tarkastelulla pyrittiin selvittämään, onko ICT-taidoilla merkitystä valitun tietojärjestelmän käytettävyyteen ja opittavuuteen. Tutkimuksessa havaitaan, että ilman hyviä ICT-taitoja käyttäjien on hankala ymmärtää virheilmoituksia ja tietoteknisiä termejä, jonka seurauksena samoja virheitä toistettiin (Pretorius & van Biljon, 2010). Tehtävien tekemiseen käytetty aika on taulukoitu käyttäjiä erittelemättä vähimmäis-, enimmäis- ja keskiarvoaikoihin (Taulukko 1). Esimerkiksi ensimmäinen ja kolmas tehtävä ovat sama, jonka tarkoituksena on havainnoida, kuinka nopeasti käyttäjät oppivat tietyn tehtävän toimintaperiaatteen. Tuloksena huomattiin, että keskiarvo suoritus aika pienentyi yli 50 prosenttia. Tämä tukee käsitystä järjestelmän helposta opittavuudesta ja korostaa käyttäjien ICT-taitojen merkitystä järjestelmän käytettävyydelle ensimmäisellä käyttökerralla (Pretorius & van Biljon, 2010).

Tietojärjestelmien opittavuutta arvioitaessa voidaan määritellä tietojärjestelmästä löytyvät oppimista tukevat oppimiskykytekijät. Nämä oppimiskykytekijät ovat ymmärrettävyys, toiminnallisuus ja sopeutuvuus (Karume & Masese, 2016). Ymmärrettävyyden tavoitteena on tarjota käyttäjälle tiettyä tehtävää varten sopiva komponentti ja antaa riittävä dokumentaatio kyseisen komponentin käyttöolosuhteista ja käyttötarkoituksista. Toiminnallisuus kuvaa järjestelmän kykyä tehdä se työ mikä sille on määritetty. Eri moduulien välinen yhteistoiminta on siis tärkeä varmistaa, jotta toiminnallisuus toteutuu järjestelmän toiminnassa (Karume & Masese, 2016). Sopeutuvuus on tietojärjestelmän kyky mukautua uuteen ympäristöön suorittaen sille määritellyt tehtävät lähes alkuperäisellä tavalla (Karume & Masese, 2016).

Karume & Masese (2016) ovat tutkimuksessaan tarkastelleet ja arvioineet sosiaalisten ohjelmistojen opittavuuteen vaikuttavia käyttäjä- ja ohjelmistopohjaisia tekijöitä. Tutkimukseen osallistunut henkilö sai satunnaisesti, joko WhatsAppin, Facebookin tai Twitterin arvioitavakseen sosiaalisesti ohjelmistoksi. Kaiken kaikkiaan 345 vastaajaa palautti arvionsa (Karume & Masese, 2016). Merkittävimpinä käyttäjäpohjaisina tekijöinä mahdollisen oppimisen esteeksi nousi ajanpuute perehtyä ohjelmiston yksityiskohtaiseen tutkimiseen, heikot IT-taidot ja yleisesti sosiaalisten ohjelmistojen vaikea ymmärrettävyys ja käyttäminen. Merkittävimmät ohjelmistopohjaiset tekijät mahdollisen oppimisen esteeksi olivat taas sosiaalisten ohjelmistojen nopea kasvu, virheellisten syötteiden huono käsittely, monimutkainen tiedonsaaminen, huono dokumentaatio ja ohjeistus, sekä uusien päivitysten tuleminen lyhyellä aikavälillä (Karume & Masese, 2016). Käyttäjäpohjaiset tekijät kertovat enemmän käyttäjien omista rajoitteista ja valmiuksista käyttää kyseisiä ohjelmistoja ja ohjelmistopohjaiset tekijät sitoutuvat ohjelmiston toimintaan ja ominaisuuksiin. Nopeasti kehittyvä ohjelmistoala tuo väistämättä kovaa kilpailua markkinoille, jonka takia uudet järjestelmät ovat yhä monimutkaisempia uusien toiminnallisuuksien kanssa. Tärkeimpänä löydöksenä voidaan sanoa, että kehitettävistä järjestelmistä tulisi tehdä helposti opittavia myös vähän IT-taitoja omaaville henkilöille, sekä dokumentaatio ja ohjeistus tulisi olla sillä tasolla, että käyttäjä voi itse korjata kohtaamansa virhetilanteen tai ongelman (Karume & Masese, 2016).

Leung et al., (2010) ovat tutkimuksessaan tutkineet, kuinka monikerroksinen käyttöliittymä vaikuttaa iäkkäämpien aikuisten tietojärjestelmän opittavuuteen mobiiliympäristössä. Tätä tutkimusta varten he kasasivat kaksi testiryhmää. Toisessa ryhmässä oli 16 henkilöä 65–81-vuotiaita ja toisessa 16 henkilöä 21–36-vuotiaita. Näille ryhmille teetätettiin tehtäviä eri käyttöliittymänäkymissä. Toinen käyttöliittymänäkymä oli monikerroksinen ja toinen yksikerroksinen niin sanottu kontrollinäkymä. Monikerroksista näkymää hyödyntänyt ryhmä suoritti perustehtävät rajoitettujen

toiminnallisuuksien käyttöliittymässä ja vaativimmat tehtävät kaikki toiminnallisuudet sisältävässä käyttöliittymässä. Kontrollinäkömää hyödyntänyt ryhmä suoritti molempien tasojen tehtävät kaikki toiminnallisuudet sisältävässä käyttöliittymässä (Leung et al., 2010). Tutkimuksen tuloksista selvisi, että monikerroksisesta käyttöliittymänäkymästä oli enemmän hyötyä vanhemmalle testiryhmälle, kuin nuoremmalle testiryhmälle. Hyöty ei kuitenkaan ollut niin merkittävä, kuin hypoteeseissa oletettiin (Leung et al., 2010). Esimerkiksi monikerroksisessa näkymässä opitut taidot perustehtävistä eivät auttaneet suorittamaan vaativampia tehtäviä kontrollinäkömässä. Toisaalta monikerroksisesta käyttöliittymänäkymästä ei koettu olevan haittaa oppimisen kannalta. Vanhemmat testiryhmän jäsenet kokivat monikerroksisen käyttöliittymänäkymän vähemmän monimutkaiseksi, kuin kontrollinäkömän (Leung et al., 2010).

4. Pohdinta

Tämän tutkielman tavoitteena oli selvittää miten tietojärjestelmän käyttöliittymästä löytyvät ominaisuudet vaikuttavat tietojärjestelmän opittavuuteen. Tietojärjestelmän käyttöliittymään liittyy vahvasti sen käytettävyys, eli kuinka helppoa käyttöliittymän käyttäminen käyttäjälle on (McNamara & Kirakowski, 2005). Opittavuus puolestaan on yksi käytettävyuden viidestä käytettävyysominaisuudesta ja tarkoittaa järjestelmän käytön oppimisen helppoutta, niin että käyttäjä suoriutuu tekemistään tehtävistään nopeasti (Nielsen Jakob, 1994, p. 26). Jos käyttäjällä on kokemusta samankaltaisista järjestelmistä, niin oppiminen on helpompaa tällaiselle käyttäjälle verrattuna käyttäjään, joka ei ole vastaavaa järjestelmää nähnyt (Nielsen Jakob, 1994, p. 28). Assimilaatioteoria puoltaa myös oppimisen helppoutta, jos käyttäjä pystyy luomaan yhteyksiä uuden tiedon ja pitkäkestoisesta muistista löytyvän tiedon välille (Davis & Wiedenbeck, 1998). Nykypäivänä monet eri tietojärjestelmät ovat enemmän tai vähemmän samankaltaisia, jolloin se jo itsessään helpottaa niiden käytön oppimista. Kovin uniikkeja käyttöliittymiä näkee nykypäivänä melko vähän, mutta jos sellaisia esiintyy, niin oman kokemuksen mukaan ne ovat melko sekavia ja vaikeita ymmärtää.

Opittavuudelle on löydettävissä viisi keskeistä ominaisuutta, jotka ovat muistettavuus, loogisuus, rekonstruoitavuus, johdonmukaisuus ja visuaalisuus. Näiden oikeanlaisella toteutuksella on tarkoitus turvata opittavuuden onnistuminen tietojärjestelmää kehitettäessä (Haramundanis, 2001). Haramundanis (2001) ei ole määritellyt näiden opittavuuden ominaisuuksien tärkeyttä tietyssä järjestyksessä, mutta esiin nousee väistämättä ihmisen fyysisten ominaisuuksien rajoitteet, jolloin opittavuuden ominaisuuksista tärkeimpinä onnistuneen toteutuksen kannalta voidaan nostaa esiin muistettavuus ja visuaalisuus. Ko et al., (2004) määrittelemät opittavuuden esteet ja Haramundanis (2001) määrittelemät opittavuutta tukevat ominaisuudet ovat siis toisiaan täydentäviä ja tukevia tekijöitä tietojärjestelmien opittavuutta arvioitaessa. Toisin sanoen esteiden syntymistä ehkäistään ominaisuuksien oikeanlaisella toteuttamisella. Jos ominaisuuksien heikko onnistuminen johtaa esteiden syntymiseen on vaarana, että tietojärjestelmän käyttäjämäärä vähenee ja vastaavat kilpailevat tietojärjestelmät saavat nämä käyttäjät itselleen, koska toteutus on parempi (Haramundanis, 2001; Ko et al., 2004).

Helposti opittavaa tietojärjestelmää suunnitellessa lähtökohtana on teknisten ominaisuuksien toteuttaminen opittavuutta edistävällä tavalla. Tämä ei kuitenkaan takaa onnistunutta lopputulosta kaikkien käyttäjien keskuudessa, koska jokaisella meillä on erilainen lähtökohta uuden järjestelmän käyttämiseen. Useassa eri tutkimuksessa nostettiin esille ICT- ja IT-taitojen merkitystä tietojärjestelmän oppimisen helppoudessa. Pretorius & van Biljon (2010) toivat tutkimuksessaan esille erot, joita kohdejärjestelmän avulla suoritetuista tehtävistä saatiin kahden ryhmän välille. Toisen ryhmän henkilöitä kutsuttiin ICT-asiantuntijoiksi ja toisen ryhmän henkilöitä ei-asiantuntijoiksi. Kaikki kohdehenkilöt olivat aiemmin käyttäneet kohdejärjestelmää (Pretorius & van Biljon, 2010). Karume & Mase (2016) puolestaan saivat selville tutkimuksessaan, että merkittävimmät käyttäjäpohjaiset tekijät kohdejärjestelmän opittavuudelle olivat heikot IT-taidot, ajanpuute järjestelmän yksityiskohtaiseen tutustumiseen ja kokemus kohdejärjestelmän kaltaisten järjestelmien vaikeasta käytettävydestä ja ymmärrettävyydestä. Esiin siis nousi ajankäytöllisen ja käyttäjän taitojen lisäksi ennakkoletukset käytössä olevasta järjestelmästä (Karume & Mase, 2016). Näistä tutkimuksista nousee esiin se, kuinka tärkeää on tehdä tietojärjestelmistä mahdollisimman selkokieliä, jotta jokainen käyttäjä ymmärtäisi mitä tietyn tehtävän tekeminen vaatii.

Lisäksi mahdollisissa virhetilanteissa käyttäjälle tulisi ilmaista mahdollisimman selkokielisesti mistä virhe johtuu ja kuinka hän voi sen korjata. Etenkin, jos kohdekäyttäjryhmässä on esimerkkitapauksien mukaisesti eri taitotasoisia tietoteknisellä osa-alueella, niin vain tietoteknisissä piireissä käytettävät termit eivät sovi sellaisenaan tehtävänantoihin ja virheviesteihin. Jos näiden termien käyttäminen on kuitenkin välttämätöntä, niin yhtenä ratkaisuna voi olla lisäselityksen antaminen termille tai esimerkkien avulla havainnoiminen halutusta syötteestä tai toiminnosta. Nämä havainnot ovat ristiriidassa Haramundanis (2001) määrittelemän johdonmukaisuus ominaisuuden kanssa ja lisäävät Ko et al. (2004) määrittämiä käytön, sekä ymmärtämisen esteitä.

Leung et al., (2010) toivat tutkimuksessaan esille monikerroksisen käyttöliittymänäkymän, jonka avulla pyrittiin parantamaan iäkkäämpien aikuisten oppimista mobiiliympäristössä. Tutkimuksen tulosten perusteella ei havaittu hypoteeseihin pohjautuen haluttua lopputulosta oppimisen kannalta, mutta kyselyiden mukaan monikerroksinen käyttöliittymänäkymä koettiin luonnollisempana ja mukavampana tapana oppia sen yksinkertaisuuden takia (Leung et al., 2010). Tutkimuksessa ei havaittu, että monikerroksinen käyttöliittymänäkymä olisi heikentänyt tietojärjestelmän opittavuutta. Mitä vanhemmat testiryhmäläiset monikerroksista käyttöliittymänäkymää käyttivät sitä, suuremman hyödyn he oppimisen kannalta siitä saivat. Monikerroksisen käyttöliittymänäkymän koettiin myös alentavan omaksumisen esteitä (Leung et al., 2010). Omaksumisen este terminä oli eriävä muusta kirjallisuudesta, koska sitä ei ollut tätä ennen tullut vastaan. Esimerkiksi verrattaessa Ko et al., (2004) opittavuuden esteisiin, suoraa viittausta omaksumisen esteeseen ei ole. Itse ajattelen omaksumisen esteen koostuvan osittain Ko et al., (2004) määritellystä tiedon esteestä. Tiedon esteessä on siis kyse siitä, kun käyttäjällä on hankaluuksia saada tietoa järjestelmän toiminnasta. Esimerkiksi joidenkin ominaisuuksien löytäminen voi liian huonolla tiedolla olla mahdotonta (Ko et al., 2004). Monikerroksisen käyttöliittymänäkymän käyttäminen osana tietojärjestelmän opittavuuden parantamista iäkkäämmille aikuisille toimi Leung et al., (2010) tutkimuksessa, mutta tätä ei ole laajemmin tutkittu tämän jälkeen. Leung et al., (2010) tutkimuksessa käytettiin mobiililaitteena Nokia E61i puhelinta, joka verrattuna nykyajan älypuhelimiin on todella vanhaa teknologiaa. Nykyajan älypuhelimien näytöt ovat esimerkiksi paljon suurempia, jolloin on mahdollista, että jo näytön koko voi omalta osaltaan selkeyttää käyttöliittymän käyttöä. Lisäksi yhä vanhemmat ihmiset ovat ottaneet käyttöön mobiililaitteita ja käyttävät niitä jokapäiväisesti, joten uudella tämän tyyppisellä tutkimuksella olisi mahdollista selvittää nykytilanne, onko tämän kaltaisista käyttöliittymäratkaisuista apua oppimisen kannalta iäkkäämmille aikuisille. Myös mahdollisia testiryhmien ikäjakaumia olisi tarpeen tarkastella uudelleen, koska Leung et al., (2010) määrittelemän 21–36-vuotiaiden testiryhmä ei enää tänä päivänä olisi kovin relevantti, yleistyneen mobiililaitteiden käytön myötä. Myös otoskoko voisi olla suurempi, kuin tutkimuksen 16 +16. En ole itse nähnyt missään mahdollisuutta valita käyttöliittymää oman taitotason mukaan, mutta näkisin tämän olevan mahdollisuus auttaa juuri iäkkäämpiä ihmisiä käyttöliittymien käytössä ja niiden oppimisessa.

Yksi keskeinen esiin noussut tekijä tietojärjestelmän opittavuuden kannalta on johdonmukaisuus. Haramundanis (2001) yksi viidestä keskeisestä ominaisuudesta on johdonmukaisuus ja Benyon (2019) tuo esille johdonmukaisuuden design muotoilussa ja suunniteluperiaatteissa. Haramundanis (2001) tekee jaottelun johdonmukaisuudelle sisäiseen ja ulkoiseen johdonmukaisuuteen. Sisäinen johdonmukaisuus keskittyy käytettävien käsitteiden ja tyylien johdonmukaisuuteen, jolloin käyttäjälle ei tule haasteita ymmärtää asiayhteyksiä. Ulkoisessa johdonmukaisuudessa järjestelmän tuottama informaatio, kuten syötteet ovat johdonmukaisessa muodossa (Haramundanis,

2001). Tällaista jakoa Benyon (2019) ei tee. Benyon (2019) on ikään kuin sisällyttänyt nämä molemmat Haramundanis (2001) jaotukset yhden johdonmukaisuuden alle, sekä design muotoilussa, että suunnitteluperiaatteissa. Tällöin Benyon (2019) määrittelemä johdonmukaisuus kuvastaa lähinnä Haramundanis (2001) sisäistä johdonmukaisuutta, johon on sisällytetty ulkoinen johdonmukaisuus. Johdonmukaisuudella käsitteellisyys ja fyysiset elementit ovat samassa linjassa monimutkaisuuksien välttämiseksi. Näiden kahden tutkimuksen suurin ero on julkaisuajankohta. Melkein 20 vuodessa ala on kehittynyt paljon, jolloin eri tekijöiden ja elementtien tärkeys tietojärjestelmissä on voinut vaihdella. Voidaan kuitenkin todeta niin, että näinkin tärkeä tekijä kuin johdonmukaisuus on kuitenkin pysynyt saman kaltaisena parin vuosikymmenen aikana. Termi ja sen vaatimukset ovat toki tarkentuneet, mutta pääajatus on sama. Siksi on mielenkiintoista huomata, että teknologia kehittyi todella nopeasti, mutta perusasiat esimerkiksi suunnittelun taustalla pysyvät lähes muuttumattomina.

Loogisuus eri muodoissa ja käyttöliittymän ymmärtäminen nousivat myös esiin useassa eri tutkimuksessa. Grossman et al., (2009) määrittelivät viisi aspektia, jotka tulisi ottaa huomioon tietojärjestelmän opittavuutta arvioitaessa ja suunnitellessa. Tehtävien kulku on yksi näistä aspekteista ja sillä pyritään varmistamaan tietyn toiminnallisuuden suorittamiseen vaadittujen tapahtumasarjojen helppo löytäminen. Haramundanis (2001) puolestaan määritteli loogisuuden yhdeksi keskeiseksi ominaisuudeksi opittavuuden kannalta. Loogisuudella halutaan varmistaa helposti havaittavat toimintatavat halutun tehtävän suorittamiseksi. Nämä kaksi termiä ovat määritelmiltään lähellä toisiaan, joten voidaan olettaa niiden olevan siis sama tekijä opittavuutta varmistettaessa. Grossman et al., (2009) tutkimuksessaan vertailivat erilaisia tapoja mitata opittavuutta ja havaita oppimiso ongelmia ja näin loivat viisi aspektia. Loogisuus tekijänä on helpompi esittää, kuten Haramundanis (2001) on sen määritellyt. Hyvällä loogisuudella käyttäjä saadaan helposti muistamaan ja oppimaan toimintatapoja eri toiminnallisuuksien suorittamiseen.

Tietojärjestelmän opittavuutta helpottavana tekijänä koettiin olevan myös monialainen ymmärrettävyys. Grossman et al., (2009) määrittelemästä viidestä aspektista ymmärtäminen nousee esiin tätä tekijää kuvailtaessa. Ymmärtäminen on heikkoa, jos käyttäjä on tietoinen ja paikantamaan tarjolla olevat työkalut ja toiminnot, mutta ei ymmärrä, miten niitä käytetään. Hyvällä dokumentaatiolla ymmärtämistä voidaan parantaa (Grossman et al., 2009). Ymmärtämisen rinnalle sopii osittain myös Benyon (2019) määrittelemistä suunnitteluperiaatteista affordanssi. Affordanssilla pyritään suunnittelemaan asiat niin, että käyttäjän on helppo ymmärtää mihin sitä käytetään. Affordanssi on hyvin kulttuuri keskeinen, joten toisessa kulttuurissa eri symboli tai esittämistapa voi tarkoittaa toista, kuin toisessa kulttuurissa (Benyon, 2019). Karume & Masese (2016) ovat määritelleet tietojärjestelmän opittavuuden arviointiin kolme oppimiskykytekijää, joista asiayhteyden nousee ymmärrettävyys. Ymmärrettävyydellä pyritään tarjoamaan käyttäjälle komponentti tiettyä toiminnon suorittamista varten ja lisäksi tarjoamaan dokumentaatiota ymmärrettävyyden parantamiseksi (Karume & Masese, 2016). Nämä kaikki kolme nostoa ovat toistensa lähellä, vaikka joltain osin eroavat toisistaan. Ymmärtämisen aspektilla ja ymmärrettävyyden oppimiskykytekijällä on yhteistä dokumentaation tarjoaminen käyttäjälle, mutta ymmärtämisen aspekti ei ole välttämättä aina niin tekniselle tasolle pohjautuva, kuin ymmärrettävyyden oppimiskykytekijä on. Molemmat ovat varmasti tärkeitä omalla tasollaan ja näiden yhdistäminen tietojärjestelmää toteutettaessa tuo käyttäjälle ymmärrettävää lopputuloksen. Affordanssi suunnitteluperiaatteena on hieman eroava näistä kahdesta edellä mainitusta ymmärtämiseen liittyvästä tekijästä. Affordanssilla pyritään auttamaan käyttäjää hahmottamaan, mihin eri asioita käytetään ei niinkään miten niitä käytetään. Esiin nousee vahvasti visuaalinen puoli, kuten Benyon (2019) on todennut esitettävillä

asioilla olevan kulttuurinen eroavaisuus tarkoituksissaan. Affordanssi on siis lähempänä symboliikkaan tai elementtien visuaalisen muotoilun sitomista kulttuurisiin normeihin.

5. Yhteenveto

Tässä tutkielmassa selvitettiin miten tietojärjestelmästä löytyvät ominaisuudet vaikuttavat sen opittavuuteen. Käytettävyyteen ja opittavuuteen etsittiin sopivia aiempia tutkimuksia kirjallisuuskatsauksen avulla. Tutkimuksista löydettiin ominaisuuksia, joita tietojärjestelmän materiaaleista tulisi löytyä. Lisäksi tutkimukset osoittivat näiden ominaisuuksien onnistumisen merkityksen oppimisen esteiden ehkäisyssä. Tärkeimpinä huomioina nostettiin esiin tietojärjestelmän käyttöliittymän johdonmukaisuutta, niin termistössä kuin virhetilanteissakin. Huomionarvoista oli myös käyttäjän ominaisuuksien ja järjestelmän ominaisuuksien yhteen sovittaminen. Eli tietojärjestelmää suunniteltaessa tulisi tarkkaan miettiä kaikki mahdolliset käyttäjäryhmät ja tämän myötä toteuttaa käyttöliittymä ymmärrettävällä tavalla juuri näille käyttäjille.

Tietojärjestelmien helppo oppiminen on tärkeää digitalisoituvassa maailmassa. Käyttäjiä yritetään koskella tuomalla tietojärjestelmiin mitä hienompia toiminnallisuuksia, mutta usein ne ovat liian vaikeita käyttää, jolloin käyttäjät eivät käytä näitä järjestelmiä. Tietojärjestelmän vaikea opittavuus voi heikentää sen suosiota ihmisten käytössä. Varsinkin, jos käyttäjä törmää järjestelmän käyttöä opitellessaan ylitsepääsemättö mään ongelmaan, mahdollisuutena on, että hän vaihtaa johonkin toiseen samantapaiseen, helpommin opittavaan järjestelmään. Jotta tietojärjestelmän opittavuutta voidaan parantaa, on ensin tunnistettava ongelmat, jotka vaikuttavat oppimiseen.

Tutkimuskysymyksen kannalta löydettiin monia opittavuuteen vaikuttavia tekijöitä. Tärkeimpiä löydettyjä tekijöitä kirjallisuuden pohjalta ovat johdonmukaisuus, loogisuus ja ymmärtäminen. Tutkimusongelmaan peilaten nämä tekijät eivät välttämättä ole yksiselitteisiä, sillä ne voivat yhdessä toisten tekijöiden kanssa muodostaa yhtenäisen kokonaisuuden. Johdonmukaisuuden rooli opittavuudessa on pitää esitetyt elementit ja kieli samassa linjassa monimutkaisuuksien välttämiseksi. Johdonmukaisuudesta löydettiin pari eri määritelmää, mutta sisällöltään ne vastasivat samaa asiaa. Hyvällä johdonmukaisuudella käyttäjän ei tarvitse käyttää voimavaroja tulkitsemaan käyttöliittymän elementtien asetteluja tai käytettyä kieltä. Loogisuudella opittavuuden kannalta on myös tärkeä rooli. Hyvällä loogisuudella varustetusta tietojärjestelmästä käyttäjän on helppo suorittaa haluttu tehtävä, kun toimintatavat ovat loogisesti esitetty. Tämän helppouden kautta käyttäjä voi muistaa jatkossa, miten eri tehtävät tai toiminnallisuudet suoritetaan, kun loogisuus on kunnossa. Ymmärtämiselle löydettiin eniten vastineita kirjallisuudesta verrattuna löydettyihin tekijöihin. Ymmärtämisellä pyritään auttamaan käyttäjää hahmottamaan mihin ja miten eri asioita käytetään. Esiin nousi myös tarjottava dokumentaatio, jonka avulla käyttäjä voi perehtyä tarkemmin tarjolla oleviin toiminnallisuuksiin ja työkaluihin ja tätä kautta kasvattaa ymmärtämistä tietojärjestelmää kohtaan ja samalla parantaa opittavuutta. Ymmärtämisen kautta opittavuus paranee myös kulttuurisidonnaisten elementtien muotoilun ja symboliikan käytön myötä.

Opittavuutta on tutkittu paljon ja tutkimusaineistoa tulee koko ajan lisää useilta eri toimialoilta. Kiinnostavaa jatkotutkimusta tästä aiheesta voisi tehdä saavuttavuuteen peilaten, eli miten opittavuus ja saavutettavuus tietojärjestelmissä kytkeytyvät toisiinsa. Lisäksi jatkotutkimusta voisi toteuttaa siitä näkökulmasta, kuinka yleistä on, että tietojärjestelmää suunniteltaessa ei ole määritelty kohderyhmää tarpeeksi tarkasti, jolloin tietojärjestelmän ymmärrettävyys ja sitä kautta opittavuus heikentyvät. Kvantitatiivisessa tutkimusmenetelmässä kerätyn tiedon pyritään esittämään tärkeimmät löydökset numeerisesti, jonka avulla voidaan paljastaa käyttäytymismalleja ja trendejä. Tyypillisiä

tiedon keräystapoja ovat haastattelut ja kyselyt (Goertzen, 2017). Tätä tutkimusmenetelmää hyödyntäen edellä mainituissa aiheen laajentamistutkimuksissa voidaan saada selville tarkempia tuloksia tämänhetkisestä tilanteesta tietojärjestelmien opittavuuden näkökulmasta. Lisäksi tutkimusta voisi tehdä erittelemällä tietokonepohjaiset käyttöliittymät ja mobiilikäyttöliittymät. Mitä eri elementtejä näiden välillä on ja mitä asioita tulee ottaa huomioon niiden kehittämisessä.

Lähteet

- Alonso-Ríos, D., Vázquez-García, A., Mosqueira-Rey, E., & Moret-Bonillo, V. (2010). Usability: A Critical Analysis and a Taxonomy. *International Journal of Human-Computer Interaction*, 26(1), 53–74. <https://doi.org/10.1080/10447310903025552>
- Benyon, D. (2019). *Designing user experience: a guide to HCI, UX and interaction design* (Fourth edition). Pearson Education Limited.
- Davis, S., & Wiedenbeck, S. (1998). The effect of interaction style and training method on end user learning of software packages. *Interacting with Computers*, 11(2), 147–172. [https://doi.org/10.1016/S0953-5438\(98\)00026-5](https://doi.org/10.1016/S0953-5438(98)00026-5)
- Goertzen, M. J. (2017). Chapter 3. Introduction to Quantitative Research and Data. *Library Technology Reports*, 53(4), 12–18.
- Grossman, T., Fitzmaurice, G., & Attar, R. (2009). A survey of software learnability: Metrics, methodologies and guidelines. *Conference on Human Factors in Computing Systems - Proceedings*, 649–658. <https://doi.org/10.1145/1518701.1518803>
- Haramundanis, K. (2001). Learnability in information design. *ACM SIGDOC Annual International Conference on Computer Documentation, Proceedings*, 7–11. <https://doi.org/10.1145/501516.501519>
- Holzinger, A. (2005). Usability engineering methods for software developers. *Communications of the ACM*, 48(1), 71–74. <https://doi.org/10.1145/1039539.1039541>
- International Organization for Standardization. (2018). *Ergonomics of human-system interaction — Part 11: Usability: Definitions and concepts* (ISO 9241-11:2018). Haettu 12.11.2022 osoitteesta: <https://www.iso.org/obp/ui/#iso:std:iso:9241:-11:ed-2:v1:en>
- International Organization for Standardization. (n.d.). *Standards*. Haettu 12.11.2022 osoitteesta: <https://www.iso.org/standards.html>
- Karume, S. M., & Masese, N. B. (2016). Software and User Based Factors Influencing Social Software Learnability. *International Journal of Science and Research (IJSR) ISSN*, 7(1), 628-633.
- Ko, A. J., Myers, B. A., & Aung, H. H. (2004). Six learning barriers in end-user programming systems. *Proceedings - 2004 IEEE Symposium on Visual Languages and Human Centric Computing*, 199–206. <https://doi.org/10.1109/VLHCC.2004.47>
- Leung, R., Findlater, L., Mcgrener, J., Graf, P., & Yang, J. (2010). Multi-Layered Interfaces to Improve Older Adults Initial Learnability of Mobile Applications. *ACM Transactions on Accessible Computing (TACCESS)*, 3(1). <https://doi.org/10.1145/1838562.1838563>
- Lewis, J. R. (2014). Usability: Lessons Learned ... and Yet to Be Learned. *International Journal of Human-Computer Interaction*, 30(9), 663–684. <https://doi.org/10.1080/10447318.2014.930311>

- Maguire, M. (2001). Methods to support human-centred design. *International Journal of Human-Computer Studies*, 55(4), 587–634. <https://doi.org/10.1006/IJHC.2001.0503>
- McNamara, N., & Kirakowski, J. (2005). Defining usability: Quality of use or quality of experience? *IEEE International Professional Communication Conference*, 200–204. <https://doi.org/10.1109/IPCC.2005.1494178>
- Morley, J., Widdicks, K., & Hazas, M. (2018). Digitalisation, energy and data demand: The impact of Internet traffic on overall and peak electricity consumption. *Energy Research & Social Science*, 38, 128–137. <https://doi.org/10.1016/J.ERSS.2018.01.018>
- Nielsen Jakob. (1994). *Usability Engineering*. Morgan Kaufmann Publishers Inc., San Francisco, CA, United States.
- Pretorius, M., & van Biljon, J. (2010). Learning management systems: ICT skills, usability and learnability. *Interactive Technology and Smart Education*, 7(1), 30–43. <https://doi.org/10.1108/17415651011031635>
- Rafique, I., Weng, J., Wang, Y., Abbasi, M. Q., Lew, P., & Wang, X. (2012). Evaluating software learnability: A learnability attributes model. *2012 International Conference on Systems and Informatics, ICSAI 2012*, 2443–2447. <https://doi.org/10.1109/ICSAI.2012.6223548>
- Santoso, H. B., & Schrepp, M. (2018). Importance of user experience aspects for different software product categories. *Communications in Computer and Information Science*, 886, 231–241. https://doi.org/10.1007/978-981-13-1628-9_21
- Torraco, R. J. (2016). Writing Integrative Literature Reviews: Guidelines and Examples. *Human Resource Development Review*, 4(3), 356–367. <https://doi.org/10.1177/1534484305278283>
- Vujošević, D., Kovačević, I., & Vujošević-Janičić, M. (2019). The learnability of the dimensional view of data and what to do with it. *Aslib Journal of Information Management*, 71(1), 38–53. <https://doi.org/10.1108/AJIM-05-2018-0125>