



OULUN YLIOPISTO  
UNIVERSITY of OULU

# **Web-sovellusten käytettävyyden arviointi heuristiikkojen avulla**

Oulun Yliopisto  
Tietojenkäsittelytieteiden  
laitos  
LuK-tutkielma  
Jaakko Soini (2367851)  
4.12.2016

## Tiivistelmä

Heuristinen evaluointi on menetelmä, jota käytetään ohjelmistojen käytettävyyden arviointiin. Sen tavoitteena on paljastaa mahdollisia käytettävyyssongelmia ohjelmistossa. Heuristisessa evaluoinnissa asiantuntijat vertaavat ohjelmistoa tai sen prototyyppiä yleisesti tunnettuihin periaatteisiin eli heuristiikkoihin. Tämä kirjallisuuskatsaus esittelee tunnettuja heuristiikkoja ja niiden soveltamista erityisesti web-sovelluksiin, millaisia omia heuristiikkoja web-sovellusten käytettävyyden arviointiin kuuluu, mitä eroja ja yhtäläisyyksiä niillä on perinteisiin heuristiikkoihin verrattuna ja kuinka nämä heuristiikat ovat kehittyneet perinteisistä heuristiikoista.

### *Avainsanat*

käytettävyys, heuristiikat, heuristinen evaluointi, web-sovellukset

### *Tutkielman ohjaajat*

Mikko Rajanen

## Alkusanat

Moni ohjelmisto tai web-sovellus voi tuntua sen suunnittelijalle, projektin vetäjälle, ohjelmoijalle tai graafikolle maailman helpoimmalta asialta käyttää ja lähestyä. Mutta ennen pitkää nämä ohjelmistokehittäjät huomaavat, että kuka tahansa sovellusta käyttävä ja siitä hyötyvä henkilö, siis sovelluksen käyttäjä, ei välttämättä ymmärrä tai osaa käyttää sovellusta niin hyvin kuin heidän toivoisi. Tämän ongelman vuoksi käytettävyyttä ja käyttäjäkokemusta on tutkittu jo pitkään, ja ongelmien korjaamiseksi on laadittu eri menetelmiä.

Tämän kirjallisuuskatsauksen avustamisesta erityiset kiitokset ohjaajalle Mikko Rajaselle sekä kannustajina toimineille opiskelijatovereille, ystäville ja sukulaisille.

# Sisällys

1.	Johdanto .....	5
2.	Käytettävyyteen ja heuristiikkoihin liittyvät määritelmät .....	6
2.1	Käytettävyys .....	6
2.2	Käyttäjäkokemus .....	7
2.3	Halvat käytettävyyden arviointimenetelmät.....	7
2.4	Heuristiikat .....	8
2.5	Heuristinen evaluointi .....	9
3.	Web-sovellukset ja käytettävyys .....	11
3.1	Web-sovellusten erot sovellusohjelmiin.....	11
3.3	Käytettävyyden arviointia .....	12
4.	Web-sovellukset ja heuristinen arviointi .....	14
4.1	Nielsenin heuristiikat sovellettuna web-ympäristöön .....	14
4.2	Heuristiikat ja web-sovellusten houkuttelevuus.....	16
4.3	Heuristiikkojen kritiikkiä ja heikkouksia .....	18
5.	Yhteenveto .....	19
	Lähteet.....	21

# 1. Johdanto

Käytettävyydellä on ollut suuri merkitys jo 90-luvulla, mutta sen tärkeys on kasvanut teknologian yleistyessä. Jakob Nielsenin mukaan käytettävyuden merkitys muuttui, kun tietokoneiden käyttö siirtyi pieniltä erikoistuneilta käyttäjäryhmiltä massoille, jolloin käytön konteksti laajentui (Nielsen, 1993, s. 8). Nykyään käyttäjien odotukset web-sovelluksia kohtaan ovat kasvaneet ja sivustojen oletetaan toimivan aina ja tiedon oletetaan löytyvän helposti ja nopeasti (Nielsen & Loranger, 2006).

Pääasiallisesti kirjallisuuskatsaukseni tarkoituksena on tutkia web-sovellusten käytettävyuden arvioinnin eroja perinteisiin sovellusohjelmiin tai peleihin sekä koostaa kattava yleiskäsitys web-sovellusten ja sivustojen vaatimista erikoisuuksista, joita perinteiset ohjelmistot eivät yleensä vaadi. Lisäksi pyrin tunnistamaan eri työkaluja ja menetelmiä, joista web-kehittäjät ja -suunnittelijat voivat hyötyä käytettävyuden parantamisessa.

Tässä kirjallisuuskatsauksessa keskityn suurimmaksi osaksi *heuristiseen evaluointiin*, joka on laajalti hyödynnetty käytettävyudenarviointimenetelmä, ja pyrin tutkimaan lähteitä, jotka painottavat heuristista evaluointia web-kontekstissa. Esittelen kirjallisuuskatsauksessa löytämiäni heuristiikkoja, sekä miten käytettävyysasiantuntijat voivat soveltaa näitä heuristiikkoja osana käytettävyuden arviointia.

Käytetyimpiä hakukoneita tieteellisille lähteille tässä tutkimuksessa olivat *Scopus* ja *Google Scholar* -palvelut. Hakutermeihin kuuluivat mm. ”Web usability”, ”usability evaluation”, ”heuristic” ja ”heuristic evaluation”. Monet lähteistä löytyivät myös hyvin tunnettujen lähteiden viittauksia tulkitsemalla.

Luvussa kaksi esitellään yleisellä tasolla käytettävyyteen liittyviä käsitteitä ja heuristiikkoja sekä heuristista evaluointia. Luvussa kolme tarkastellaan web-kontekstia ja käytettävyyttä sekä esimerkiksi web-kehityksen asettamia vaatimuksia käytettävyuden arvioinnille. Luku neljä perustuu web-sovellusten ja sivustojen arviointiin heuristiikkojen avulla; luvussa tuodaan esille lukuisia kirjallisuudessa mainittuja heuristiikkoja web-sovellusten ja sivustojen arviointia varten. Luku viisi toimii yhteenvetona sekä analyysinä tutkielman sisällöstä.

## 2. Käytettävyyteen ja heuristiikkoihin liittyvät määritelmät

Tässä luvussa esitellään käytettävyyteen, käyttäjäkokemukseen ja erityisesti heuristiseen arviointiin liittyviä termejä ja käsitteitä yleisellä tasolla. Määritelmiin web-sovellusten suhteen ei vielä oteta kantaa. Eri käsitteet esitellään hyödyntäen eri standardeja ja määritelmiä eri henkilöiltä.

### 2.1 Käytettävyys

Käytettävyyttä käsitellään esimerkiksi ISO 9241 -standardin osassa 210, joka on tarkoitettu tietokonepohjaisten vuorovaikutteisten järjestelmien ihmiskeskeiselle suunnittelulle (Suomen Standardisoimisliitto SFS [SFS], 2010). ISO 9241-210 on käännetty Suomeksi vuonna 2011 ja se on vahvistettu suomalaisiksi kansalliseksi standardiksi. Koska kyseessä on standardi, siinä ei käydä läpi yksityiskohtaisesti ihmiskeskeisen suunnittelun menetelmiä ja tekniikoita (SFS, 2010).

ISO 9241-210 -standardin määritelmä käytettävyydestä on lainattu aikaisemmasta käytettävyyttä käsittelevästä ISO 9241-11 -standardista (ISO 9241-11, 1998). Sen mukaan käytettävyys on ”mitta, miten hyvin määrätyt käyttäjät voivat käyttää järjestelmää, tuotetta tai palvelua tietyssä käyttötilanteessa saavuttaakseen määritetyt tavoitteet tuloksellisesti, tehokkaasti ja tyytyväisinä” (SFS, 2010). ISO 9241-11 -standardia aikaisempi IEEE:n vuonna 1990 laatima 610.12 -standardi puolestaan määrittelee käytettävyyden järjestelmän tai sen komponenttien helppoudeksi, jonka avulla käyttäjä pystyy opettelemaan valmistamaan syötteitä ja tulkitsemaan niiden tulosteita. (IEEE Std 610.12-1990).

Tunnettuja käytettävyyteen liittyviä teoksia tehnyt Jakob Nielsen (1993, s. 26) ei kuitenkaan määrittele käytettävyyttä ohjelmiston yksiulotteisena ominaisuutena, vaan toteaa sen koostuvan perinteisesti viidestä eri attribuutista: opittavuus, tehokkuus, muistettavuus, virheettömyys ja tyytyväisyys (*learnability, efficiency, memorability, errors, satisfaction*). Attribuuttien määritelmät ovat Nielsenin (1993) mukaan:

- *Opittavuus* tarkoittaa, että käyttäjä pystyy käytön aloittamisen jälkeen omaksua kyseisen järjestelmän toiminnan ja ominaisuudet nopeasti.
- *Tehokkuus* tarkoittaa, että kun käyttäjä on oppinut järjestelmän käytön, hänen on mahdollista käyttää järjestelmää nopeasti ja tuottoisasti.
- *Muistettavuus* tarkoittaa, että järjestelmän käyttöä ei tarvitse opetella kokonaan uudelleen, jos käyttäjä ei ole käyttänyt järjestelmää vähään aikaan, vaan käyttäjä pystyy palauttamaan sen mieleensä nopeasti.
- *Virheettömyys* tarkoittaa, että järjestelmän tavallisesta käytöstä ei aiheudu usein virhetilanteita. Mahdollisten virheilmoitusten tulee myös olla käyttäjille ymmärrettäviä ja käyttäjien pitää pystyä palautumaan virhetilanteista mahdollisimman helposti. Käyttäjän toimet eivät myöskään saa johtaa katastrofaalisiin virhetilanteisiin.

- *Tyytyväisyydellä* tarkoitetaan, että järjestelmän käytön ylipäättään tulisi olla käyttäjille miellyttävää.

Käytettävyyden mittaamiseen tarvitaan käyttäjädataa, yleensä mahdollisimman monelta käyttäjältä. Riippumatta siitä kuinka monella tai millaisella käyttäjällä järjestelmän käytettävyyttä testataan, on tärkeää, että käytettävyyttä arvioidaan tietäen tarkasti käyttäjäryhmä ja käyttötilanne, sillä eri käyttäjäryhmien priorisoitavat käytettävyyssattribuutit eivät aina ole samoja, juurikin käyttäjän ja heidän tehtäviensä tunteminen on erittäin tärkeässä roolissa, kun käytettävyyttä tarkastellaan. (Nielsen, 1993).

Mutta Nielsenin (1993, s. 24-25) käytettävyys on vain pieni osa suurempaa, järjestelmän hyväksyttävyyden (*system acceptability*) kokonaisuutta, kun taas järjestelmän hyväksyttävyyden puolestaan koostuu sosiaalisesta hyväksyttävyydestä (*social acceptability*) ja käytännöllisestä hyväksyttävyydestä (*practical acceptability*). Käytettävyys on osana järjestelmän hyödyllisyydessä (*usefulness*), jonka lisäksi käytännölliseen hyväksyttävyyteen luetaan myös kriteerejä kuten hinta, luetettavuus sekä yhteensopivuus.

## 2.2 Käyttäjäkokemus

Käytettävyys voidaan käsittää myös osana laajempaa termiä, käyttäjäkokemus (*user experience*). ISO 9241-210 -standardi määrittelee käyttäjäkokemukseksi ne ”henkilön havainnot ja vasteet, jotka ovat seurausta tuotteen, järjestelmän tai palvelun käytöstä ja/tai ennakoitusta käytöstä” (SFS, 2010, s. 16). Tämän standardin mukaan myös suunnittelun tulisi kohdistua koko käyttäjäkokemukseen, ei vain käytettävyyteen, jotta se voidaan lukea ihmiskeskeiseksi (*human-centered*) suunnitteluksi (SFS, 2010, s. 22).

Käyttäjäkokemusta huomioiva suunnittelu ottaa huomioon paljon muita kriteerejä, jotka eivät kuulu itse käytettävyyden määritelmään, kuten organisatoriset vaikutukset, käyttäjän näkökulmat tuotemerkeistä ja sen mainonnasta. (SFS, 2010, s. 22).

## 2.3 Halvat käytettävyyden arviointimenetelmät

Halvat käytettävyyden arviointimenetelmät (*discount usability*) ovat Nielsenin esittelemiä menetelmiä, joiden avulla ohjelmistokehittäjät ja muut alalla toimivat voivat oppia ja selvittää uusia heuristiikkoja nopeasti ja vaivattomasti, näiden menetelmien opettaminen ryhmälle voi viedä vain puoli seminaari-päivää. (Nielsen, 1994, s.25). Nämä ovat käytettävyydenarviointitekniikoita, joihin kuuluu lukuisia nykyään laajasti käytettyjä ja tunnettuja tekniikoita, kuten tässäkin kirjallisuuskatsauksessa esiteltävä heuristinen evaluointi ja heuristiikat. Heuristisen evaluoinnin lisäksi halpoihin käytettävyyden arviointimenetelmiin Nielsen lukee mukaan käyttäjän ja tämän tehtävien observoinnin, käyttöskenaarioiden arvioinnin ja testauksen paperi- tai toiminnallisten prototyyppien avulla sekä yksinkertaistetun ääneen ajattelun käyttäjällä (Nielsen, 1993, s. 17-20).

Halpojen käytettävyydenmenetelmien ei ole tarkoitus olla ”täydellisiä” ratkaisuja, sillä niin sanotut täydelliset tai parhaat menetelmät ovat erittäin laajoja ja ohjelmistokehittäjät

voivat juuri niiden työmäärän paljoudesta johtuen välttää käytettävyyden arviointia kokonaan (Nielsen, 1993, s 17). Nielsenin vuonna 1989 esittelemät halvat käytettävyyden arviointimenetelmät oli tarkoitettu vaihtoehdoksi näille työmäärältään suurille kvantitatiivisille käytettävyyden arviointimenetelmille. Kaksikymmentä vuotta myöhemmin julkaisemassaan artikkelissa Nielsen kritisoi monien yritysten soveltamia arviointimenetelmiä, kuten paperiprototyypoinnin käytön vähäisyyttä ja sitä kuinka monet ohjelmistokehitystiimit käyttävät paljon aikaa laatiessaan monimutkaisia esityksiä kehitettävästä tuotteesta ennen varsinaisten käyttäjien palautteen keruuta. (Nielsen, 2009).

## 2.4 Heuristiikat

Nielsenin määritelmän mukaan heuristiikat ovat ”tunnistettuja käyttöliittymien suunnitteluperiaatteita” (Nielsen, 1993, s. 155). Hän määrittelee niiden olevan toisessa kirjassaan myös lisäksi ”yleisiä sääntöjä, jotka näyttävät kuvaavan käytettävien käyttöliittymien ominaisuuksia” (Nielsen, 1994, s. 28).

Nielsen luettelee kymmenen perusheuristiikkaa, joiden avulla yleisimmät ja suurin osa käytettävyysongelmista voidaan löytää. Muutoin käytettävissä olevia heuristiikkoja olisi tuhansia (Nielsen, 1993). Alun perin Nielsen määritteli kymmenen perusheuristiikkaa Rolf Molichin kansa vuonna 1990, mutta päivitti listaa vuonna 1994. Uusi lista (Nielsen, 1994, s. 30) perustui Nielsenin tutkimukseen 249 eri käytettävyysongelmasta. Nielsen on myös laatinut heuristiikoistaan verkkoartikkelin (Nielsen, 1995).

- Yksinkertainen ja luonnollinen dialogi [1993] / Esteettinen ja minimalistinen suunnittelu [1994]: Ohjelma kertoo käyttäjälle vain tarvittavan tiedon tiettyä käyttötilannetta varten. Sen ei tulisi kertoa käyttäjälle asiaankuulumatonta tai harvoin tarvittavaa tietoa. Liiallinen asiaankuulumaton tieto vähentää tarpeellisen tiedon näkyvyyttä.
- Puhu käyttäjän kieltä [1993] / Vastaavuus järjestelmän ja tosimaailman välillä [1994]: Ohjelman tulee käyttää termejä ja nimityksiä, joita käyttäjä voi ymmärtää. Käyttöliittymässä tulisi välttää asiantuntijasanastoa, jota varsinaisen käyttäjä ei välttämättä ymmärrä.
- Minimoi muistettavien asioiden määrä [1993] / Tunnistaminen mieluummin kuin muistaminen [1994]: Käyttäjän ei itse tarvitse muistaa asioita vaiheesta toiseen. Sen sijaan käyttöliittymän osat ja toiminnot liittyvät toisiinsa niin, että kyseisten toimintojen välinen yhteys on pääteltävissä. Myös ohjeiden tulisi olla saatavilla niitä tarvittaessa.
- Yhdenmukaisuus [1993] / Yhdenmukaisuus ja standardit [1994]: Ohjelman toimintojen ja termien tulisi merkitä samaa asiaa eri puolilla ohjelmaa. Ohjelman tulee noudattaa käyttöjärjestelmän käytäntöjä.
- Palaute [1993] / Järjestelmän tilan selkeys [1994]: Ohjelman tulee pitää käyttäjä tiedotettuna järjestelmän sen hetkisestä tilanteesta sopivalla palautteella tarpeeksi lyhyessä ajassa.



- Selkeät poistumistiet [1993] / Käyttäjän hallinta ja vapaus [1994]: Jos käyttäjä joutuu tilaan, josta hän haluaakin palata takaisin, hänen pitää pystyä poistumaan tilasta helposti. Ohjelman tulee tukea kumo- ja tee uudelleen -toimintoja.
- Oikotiet [1993] / Joustavuus ja käytön tehokkuus [1994]: Edistyneille käyttäjille tulee tarjota oikoteitä ja pikatoimintoja, sekä mahdollisuus muokata niitä omaa työntekoa varten. Aloittelija ei yleensä huomaa näitä, mutta ne eivät haittaa hänen työtään.
- Hyvät virheilmoitukset [1993] / Auta käyttäjää tunnistamaan, määrittää ja palautua virhetilanteista [1994]: Selkokielliset virheilmoitukset, ei pelkkiä virhekoodeja, jotka kertovat virhetilanteen yksityiskohdat tarkasti ja ehdottavat tilanteeseen sopivaa ratkaisua.
- Virheiden ennaltaehkäisy [1993, 1994]: Selkeä ohjelmiston ja käyttöliittymän rakenne ehkäisee vahingollisia virhetilanteita. Varoita käyttäjää ennen kuin hän tekee peruuttamattomia muutoksia.
- Ohjeet ja dokumentaatio [1993, 1994]: Vaikka hyvin suunniteltua ohjelmaa pitäisikin pystyä käyttämään ilman ohjekirjaa, ohjeiden hyvällä saatavuudella helpotetaan käyttäjän taakkaa ongelmatilanteiden sattuessa.

Nämä ovat eräät tunnetuimmista heuristiikoista, mutta heuristiseen evaluointiin on laadittu muitakin vastaavanlaisia listoja. Eräs näistä on esimerkiksi Gerhardt-Powalsin periaatteet (1996). Heidän heuristiikkansa eivät ole niin tunnettuja kuin Nielsenin, mutta ne ottavat kantaa samoihin ongelmiin.

## 2.5 Heuristinen evaluointi

Heuristinen evaluointi on yksi käytettävyyden arviointimenetelmä, jota Nielsen (1993, s. 155) kuvailee yleisellä tasolla sellaiseksi prosessiksi, jossa käyttöliittymää tutkitaan ja yritetään saada mielipide, mikä käyttöliittymässä on hyvää ja mikä huonoa. Tarkempi ja muodollisempi määrittäminen heuristisesta evaluoinnista on, että se on prosessi, jossa pyritään systemaattisesti havaitsemaan mitkä osat käyttöliittymästä ovat tiettyjen heuristiikkojen mukaisia ja mitkä eivät, ja näillä tiedoilla selvittää ja parantaa ohjelmiston käytettävyyttä. Yksinkertaisemmin heuristisen evaluoinnin voi myös määrittellä arvionniksi, joka perustuu määriteltyihin peruseräisiin (De Jong & Van Der Geset, 2000).

Heuristinen evaluointi suoritetaan pääpiirteissään niin, että siinä toimii evaluoija (*evaluator*) tai useampi evaluoija yhdessä. Vaikka heuristista evaluointia voi toteuttaa vain yksi evaluoija, suosittelee Nielsen (1993, s. 155), että useampi evaluoija suorittaisi evaluointia, sillä eri evaluoijat havaitsevat eri käytettävyysongelmia kuin toiset; lisäämällä evaluoijien määrää voidaan löytää useampia ongelmia ohjelmistosta kuin vain yhdellä evaluoijalla. Nielsen suosittelee vähintään kolmen evaluoijan käyttöä, mielellään viiden (Nielsen, 1993, s. 156). Jokaisen evaluoijan tulisi suorittaa evaluointi yksin, jotta muiden evaluoiden kommentit ja havainnot eivät vaikuta heidän tuloksiinsa.

Heuristiseen evaluointiin on mahdollista liittää muitakin rooleja kuin vain evaluoija. (Nielsen, 1994, s. 25-29). Havainnoija (*observer*, tai ns. koevastaava eli *experimenter*) voi avustaa evaluoijaa tarvittaessa, kuten voi olla tarpeen, jos esimerkiksi prototyyppi ei toimi halutulla tavalla tai sen on puutteellinen, tai jos evaluoija ei tunne sovellusaluetta tarpeeksi hyvin ja tarvitsee lisäinformaatiota. Vaikka havainnoija olisikin mukana heuristisessa evaluoinnissa, tulee evaluoijien silti suorittaa evaluointi yksitellen. Kun evaluoijat ovat tutkineet käyttöliittymää yksi heuristiikka kerrallaan, kunnes kaikki heuristiikat on käyty läpi, he laativat joko kirjallisen raportin tai raportoivat suullisesti omalle havainnoijalleen. Heuristinen evaluointi kuuluu Nielsenin halpoihin käytettävyyden arviointimenetelmiin, joiden etuja on se, että yhden evaluoijan työurakka voidaan suorittaa vain parissa tunnissa (Nielsen, 1994, s. 27).

### 3. Web-sovellukset ja käytettävyys

Tässä luvussa esitellään, miten web -sovellusten konteksti eroaa usein sovellusohjelmistojen kontekstista käytettävyyden kannalta. Millaisia haasteita web-ympäristöt asettavat käytettävyydelle ja sen arvioinnille, sekä miten käytettävyysarviointi muuttaa web-sovellusten kehitysprosessia sovellusohjelmistoista.

#### 3.1 Web-sovellusten erot sovellusohjelmiin

Eroja web-sovellusten ja sovellusohjelmien välillä on lukuisia, mutta yleisimpiä näistä ovat esimerkiksi suuri määrä erilaisia sidosryhmiä, paljon käyttäjien tehtäviä ja vuorovaikutustapoja, jatkuva kehitys ja lyhyemmät revisiosykliit sekä monimutkaisempien teknologisten ratkaisujen käyttö (Bruno, Tam, Thom, 2005). Lisäksi web-sovelluksia on usein mahdollista käyttää useilla eri laitteilla, mikä lisää suunnittelijan taakkaa käyttäjän ja sovelluksen välisen vuorovaikutuksen suunnitteluun (Agarwal & Venkatesh, 2002).

Nielsen (2000, s. 10) kuvailee web-sivuille ominaisena tapana käyttäjän näkökulmasta sitä, että käytettävyys tulee näkyviin käyttäjälle usein ennen kuin hän on edes päättänyt käyttää sivustoa. Tämä ja suuri määrä kilpailevia sivustoja asettaa käytettävyyden yhdeksi tärkeimmistä ominaisuuksista web-sovelluksille. Kuusi vuotta myöhemmin kirjassaan *Prioritizing Web Usability* Nielsen ja Loranger (2006) painottavat edelleen, että käyttäjien odotukset web-sovelluksilta ovat korkeat ja huonosti suunnitellun sivuston näkee jo kaukaa. Eroja on myös käyttäjien navigoidessa web-sovelluksiin, joita he haluavat käyttää. Nielsenin ja Lorangerin (2006, s. 36) mukaan käyttäjät hyödyntävät hakukoneita päästäkseen haluamaansa palveluun usein.

Vaikka web-sovellukset voidaan määritellä eri tavalla kuin perinteiset sovellusohjelmat, erot niiden välillä eivät ole aina niin selvät. Web-sovelluksen ja sovellusohjelman eroihin vaikuttaa nykyään usein myös ”webbimäistyminen” (*webize*). Termillä tarkoitetaan, että monia sovellusohjelmia on mahdollista käyttää nykyään web-käyttöliittymän kautta esimerkiksi mobiilisovelluksella. Web-sovellusten käytöstä on tullut paljon yksinkertaisempaa ja käyttäjät ovat alkaneet tottua siihen, miten web-sovellukset yleensä toimivat (Oinas-Kukkonen, 2013, s. 71).

Heuristiikat voivat soveltua tietyn tyyppisille web-ohjelmille, tai niitä voidaan hyödyntää laajasti kaikille web-sovelluksille. De Jongin mukaan (2000) web-sovellusten kehityksessä ja arvioinnissa käytettävät heuristiikat voivat olla joko genre-spesifisiä, jotka keskittyvät tietyn tyyppiin web-sovelluksiin, esimerkiksi web store -sovelluksiin tai heuristiikat voivat kohdistua tiettyihin web-sovelluksen toimintoihin, kuten navigaatioon tai ulkoasuun. Käytettävyyden sanotaan olevan yksi web-sovellusten tärkeimmistä laatuominaisuuksista, joihin luetaan usein myös turvallisuus (*security*) ja luotettavuus (*reliability*) (Fernandez et al. 2011).

#### 3.2 Web ja käytettävyys

Käytettävyyteen ja sen arviointiin liittyen on määritelty useita kansainvälisiä standardeja, jotka tarjoavat jokainen eri näkökulman käytettävyyteen. Aikaisemmin tässä kirjallisuuskatsauksessa katsoimme standardia ISO 924-210, mutta vastaavia

käytettävyyteen liittyviä standardeja on useampiakin (Marghescu, 2009). Esimerkiksi ISO 14598 – Osa 1 kuvaa ohjelmistojen laadunarviointiprosessia, kun taas ISO 25000 -standardi keskittyy laatuvaatimuksiin ja niiden arviointiin. (Marghescu, 2009). Vaikka käytettävyydelle on lukuisia määritelmiä, niistä laajalti käytetyin on kuitenkin Nielsenin vuoden 1993 määritelmä (Matera, Rizzo & Carughi, 2006, s. 4), joka esiteltiin aikaisemmin tämän tutkielman kappaleessa 2.1 Nielsenin määritelmien pohjalta on mahdollista laatia tarkempia, mitattavissa olevia muuttujia joiden avulla käytettävyyttä ja sen arviointia voidaan lähestyä systemaattisesta näkökulmasta (Matera, 2006, s. 5).

HCI-alalla yleisesti hyväksytty määritelmä vastaa juuri Nielsenin määritelmää (Fernandez et al. 2011). Yleisesti hyväksytty käytettävyyden määritelmä ohjelmistotekniikan näkökulmasta on ISO 9126-1 standardin mukainen määritelmä. Näitä määritelmiä erottaa esimerkiksi niiden suhtautuminen käyttäjään. ISO 9126-1 standardin määritelmän mukaan käytettävyys nähdään ominaisuutena, joka vaikuttaa ohjelmistotuotteen laatuun ja standardi antaa ymmärtää, että käytettävyyttä voidaan arvioida web-kehityksen varhaisissa vaiheissa, eikä käyttäjän vuorovaikutuksen järjestelmän kanssa tarvitse olla osa käytettävyyden arviointiprosessia. (Fernandez et al. 2011).

Materan ja kumppaneiden (2006) mukaan web-ohjelmistojen käytettävyyden määrittelyä ja arviointia varten on kuitenkin tehtävä tarkennuksia, jotta käytettävyyden arviointi sopii tähän ohjelmistoluokkaan. Heidän mukaansa yleisimmin toistuvat tehtävät web-ympäristössä ovat tiedon etsiminen joko suoralla haulla tai selaamalla ja etsityn tiedon ymmärtäminen ja tiettyjen palvelujen, kuten lataamisen (*download*) käyttö. Web-käytettävyys muodostuu siten niistä ominaisuuksista, jotka tukevat näiden tehtävien tehokkuutta, opittavuutta ja tyytyväisyyttä (Matera et al. 2006, s. 6).

### 3.3 Käytettävyyden arviointia

Käytettävyysarviointi (*Usability evaluation*) koostuu metodeista, joiden tarkoituksena on arvioida ja mitata käyttöliittymän käytettävyysominaisuuksia ja tunnistaa mahdollisia ongelmakohtia (Ivory & Hearst, 2001). Käytettävyyden arviointimenetelmät (*UEM = Usability Evaluation Method*) ovat kehittyneet samalla, kun Web-pohjaisten käyttöliittymien merkitys on kasvanut. Yleisesti eri käytettävyydenarviointimenetelmät voidaan jakaa empiirisiin menetelmiin ja tarkastelumenetelmiin. Nielsenin heuristiikat ja muut heuristiikat ovat tarkastelumenetelmiä (*Inspection methods*), jotka tässä tapauksessa kohdistetaan web-artefakteihin, kuten Web-käyttöliittymään. (Fernandez et al. 2011, s. 3).

Web-sovellukset ovat nykyään merkittävässä roolissa niin yksityisen kuin julkisen sektorin palveluissa, ja sovelluksen käytettävyykskään ei jää pieneen rooliin menestyksen kannalta. Vuonna 2011 julkaistussa artikkelissa Fernandez, Insfran ja Abrahão tutkivat nimenomaan web-sovelluksia varten laadittuja käytettävyyden arviointimenetelmiä systemaattisella lähestymistavalla. Fernandez kumppaneineen painottaa myös, että käytettävyyden arviointimenetelmän valinta riippuu usein myös käytettävyyden määrittelystä, sillä kuten aikaisemmin todettiin, käytettävyyden ja käytettävyyden evaluoinnin määritelmät eroavat toisistaan hieman. (Fernandez et al. 2011).

Koska käytettävyys on yksi web-sovellusten tärkeimmistä laatuvaatimuksista, perinteiset ohjelmistojen kehitysprosessit ovat mukautuneet tähän lisäämällä arviointimetodien

käyttöä tuotteen koko elinkaaren ajalla (Matera et al. 2006). Tämän kaltainen iteratiivinen suunnittelu lähestyy käytettävyyttä evolutiivisena ominaisuutena, joka kehittyy tuotteen kehitysprosessin aikana vaihe vaiheelta. Materan (2006) mukaan käytettävyyden arviointi on testauksen jatke, jossa hyödynnetään prototyyppisiä ja verrataan löydöksiä vaatimuksiin niiden verifiointia varten. ISO/IEC 14598-1 standardi käsittelee käytettävyyden arviointiprosessia, jonka Marghescu (2009) kiteyttää neljään eri vaiheeseen:

1. Käytettävyysvaatimusten määrittäminen (*Define usability requirements*)
2. Arvioinnin tarkentaminen (*Specify the evaluation*)
3. Arvioinnin suunnittelu (*Plan the evaluation*)
4. Arvioinnin toteuttaminen (*Execute the evaluation*)

ISO/IEC 14598-1 standardi on Marghescun (2009, s. 138) mukaan esitellyistä standardeista laajamittaisin, sillä se tarjoaa tarkimman ohjeistuksen käytettävyyden arviointiin. Muut standardit eivät ole yhtä tarkkoja arviointiprosessin suhteen, mutta hänen esittelemistään standardeista ja niiden ohjeprosesseista löytyy kuitenkin lukuisia päällekkäisyyksiä.

Eri käytettävyyden arviointimenetelmät sopivat eri kehitysprosessin vaiheisiin. Eri käytettävyyden arviointimenetelmillä on omat vaatimuksensa organisaatiolle ja sen työntekijöille. Esimerkiksi Nielsenin (1993, s. 155-160) mukaan heuristinen arviointi vaatii usein asiantuntijan, joka toimii evaluoijana erillään kehittäjistä.

## 4. Web-sovellukset ja heuristinen arviointi

Tässä luvussa esitellään heuristiikkoja ja suunnitteluperiaatteita nimenomaan Web-sovelluksille tarkoitettuna. Kappale 4.1 käsittelee K. Instonen soveltamia Nielsenin heuristiikkoja, ja luvussa 4.2 käsitellään eri asiantuntijoiden tutkimuksia heuristisesta evaluoinnista web-kontekstissa. Lopuksi käsitellään heuristisen evaluoinnin heikkouksia ja sen saamaa kritiikkiä.

### 4.1 Nielsenin heuristiikat sovellettuna web-ympäristöön

Instone (1997) on laatinut version Nielsenin kymmenestä heuristiikasta, joka tarkoittaa näiden heuristiikkojen käyttöä web-ympäristössä. Esimerkiksi De Jong ja Van Der Geset (2000) toteavat, että Nielsenin heuristiikat ovat eräs tunnetuimmista käytettävyyden arviointiteoksista. Heidän mukaansa Instonen (1997) sovellus Nielsenin heuristiikoista on kuitenkin yleinen, ja sitä voidaan soveltaa edelleen myös muissa konteksteissa. Instonen tulkinnat Nielsenin heuristiikoista jättävät kuitenkin vaihtelevissa määrin pääteltävää suunnittelijalle tai arvioijalle, koska osa niistä on esitetty joskus hyvin yleisellä, joskus tarkalla tasolla. Nielsenin heuristiikat keskittyvät enemmän käytettävyyksivaatimusten kuvaamiseen, kun taas Instonen sovelletut heuristiikat tulkitsevat mahdollisia ongelmia ja niiden ratkaisuja (De Jong & Van Der Geset 2000, s. 320-321). Instonen vuoden 1997 julkaisu on kuitenkin paljon käytetty viite monissa web-suunnittelua käsittelevissä julkaisuissa.

Instone (1997) esittelee kaikki Nielsenin heuristiikat järjestyksessä ja niiden alkuperäisessä muodossaan, ja tarkoittaa sitten, kuinka web-sovellukset vaativat heuristiikkojen soveltamista. Esimerkiksi Nielsenin alkuperäisissä heuristiikoissa 1. heuristiikka on ”*Visibility of system status*”, jonka mukaan järjestelmän pitäisi pystyä pitämään käyttäjä perillä siitä, missä osassa ohjelmistoa hän on. Instonen (1997) mukaan web-sovelluksissa tämä tarkoittaa, että jokaiselta sivulta löytyy tieto, missä osassa koko web-sovellusta käyttäjä on milläkin hetkellä, ja että linkit sovelluksen muihin osiin on selvästi merkitty.

Osa Instonen esittämistä heuristiikoista ei juuri muutu, kun niitä verrataan Nielsenin heuristiikkoihin. Esimerkiksi Nielsenin 2. heuristiikka on vastaavuus järjestelmän ja tosimaailman välillä, jonka yhteydessä Instone (1997) painottaa vain että ”käyttäjän kielen puhuminen” voi olla haaste. Instonen 2. heuristiikka ei juurikaan tarkenna Nielsenin heuristiikkaa (De Jong & Van Der Geset, 2000, s. 319). Kolmannen heuristiikan, eli käyttäjän hallinta ja vapaus -heuristiikan kohdalla Instone toteaa, että web-selaimet tarjoavat usein edellisen- ja seuraava sivu -toiminnot, mutta on silti hyvä käytäntö antaa käyttäjälle selkeä mahdollisuus palata takaisin aloitussivulle.

Yhdenmukaisuus ja standardit on Nielsenin neljäs heuristiikka, ja Instonen mukaan yleisin esimerkki kyseisen heuristiikan huomioimattomuudesta on epäjohtonmukainen kieliasu. Tämä tilanne on usein yhteydessä ensimmäiseen heuristiikkaan, jonka mukaan käyttäjän pitäisi aina pystyä käsittämään nopeasti ja vaivattomasti sovelluksen sen hetkinen tila ja sijaintinsa sovelluksessa. Web-sovellusten tapauksessa heuristiikan

mainitsevat ”standardit” viittaavat esimerkiksi web-teknologian standardeihin, kuten HTML ja CSS -kieliin (Instone, 1997).

Viidennen heuristiikan mukaan virhetilanteita tulisi välttää niitä ehkäisemällä. Instonen (1997) mukaan HTML-lomakkeiden täytön rajoitteet ja mahdollinen virheiden ehkäisy voidaan toteuttaa esimerkiksi JavaScriptiä käyttämällä. Nielsen (2001) on julkaissut tähän heuristiikkaan liittyviä ohjeita virhetilanteiden käsittelystä, nimenomaan HTML-lomakkeisiin tähdäten. Yksi hänen ohjeistaan sanoo, että virhetilanteen tapahtuessa käyttäjän virheellistä syötettä ei tulisi poistaa, vaan se pitäisi säilyttää, jotta käyttäjä voi muokata ja korjata sen.

Vaikka Nielsen (1993) ja Instone (1997) molemmat mainitsevat seuraavan esiteltävän heuristiikan vasta toiseksi viimeisenä listoissaan, käsitellään se tässä yhteydessä siitä syystä, että ne liittyvät virheiden välttämiseen ja ehkäisyyn. Nielsenin 9. heuristiikan mukaan virheilmoitukset tulisi ilmaista yksinkertaisella ja selkeällä kielellä, ei pelkillä virhekoodeilla, ja lisäksi ehdottaa ratkaisua. Instone painottaa, että mahdollisimman hyvästä virheen ehkäisystä huolimatta virheitä tulee tapahtumaan ja että ratkaisun tarjoaminen on tärkeää (Instone, 1997).

Nielsenin (1993) heuristiikkojen mukaan käytön ei tulisi nojata monimutkaisiin ohjeisiin, eikä käytön pitäisi vaatia käyttäjiä muistamaan ohjelmiston eri osien tarjontaa ulkoa. Tämän sijasta käyttäjälle tarjottavan ohjeistuksen tulisi olla selvästi näkyvillä ja saatavilla. Instonen (1997) soveltamissa heuristiikoissa web-kontekstissa hän toteaa Nielsenin ”*Recognition rather than recall*” -heuristiikan olevan hyvin tiukassa suhteessa Nielsenin määrittelemän ”järjestelmän tilan selkeys” -heuristiikan kanssa. Käytettävyyden kannalta on tärkeää, että käyttäjä pystyy tunnistamaan sijaintinsa sivustolla muistamatta tarkkaa reittiä kotisivulta sen hetkiselle sivulle. Hyvät nimikkeet ja helposti tunnistettavat linkit ovat tärkeitä käyttäjän huomiota ajatellen (Instone, 1997, s. 3).

Nielsenin seitsemäs heuristiikka (Nielsen, 1993) on suunnattu nimenomaan kokeneille käyttäjille ja heidän tarpeidensa tukemista varten. Heuristiikan mukaan edistyneille käyttäjille tulee tarjota oikoteitä ja pikatoimintoja. Noviisikäyttäjät tuskin huomaavat näitä ominaisuuksia, mutta ne ovat tärkeitä kokeneimmille käyttäjille (Nielsen, 1993). Yksi parhaimpia web-pohjaisten sovellusten ja sivustojen käyttöä nopeuttavia ominaisuuksia on selainten tarjoama kirjanmerkkiominaisuus (Instone, 1997). Nielsenin (1993) seitsemännen heuristiikan mukaan sovelluksen tulee olla joustava ja käytön tehokasta. Instonen (1997) mukaan useat tätä heuristiikkaa tukevat ominaisuudet tulevat suoraan web-selaimen kautta, joten web-suunnittelijoiden täytyy huomioida näiden ominaisuuksien tukeminen web-sovellusta suunnitellessa ja sen toteutuksessa. Monimutkaiset ja pitkät linkkien kautta navigoinnin takana olevat sivustot voivat kuitenkin olla tietyille käyttäjille tärkeitä, joten kirjanmerkintäominaisuuden tukeminen tällaisille sivustoille on pidettävä mielessä. Esimerkiksi sellaisten käyttöliittymärunkojen käyttöä tulisi välttää, jotka vaikeuttavat kirjanmerkkien käyttöä (Instone, 1997). Tätä heuristiikkaa Instone (1997) käsittelee esimerkein huomattavasti tarkemmin kuin aikaisemmin esiteltyjä heuristiikkoja. Hän antaa myös teknisiä esimerkkejä, kuten suosittelee URL -osoitteen uudelleenkäyttöä ja säilyvyyttä. Lomakkeiden yhteydessä hän suosittelee GET -komennon käyttöä POST -komennon sijasta, sillä GET -komentoa

käytettäessä URL -osoite muuttuu, joten esimerkiksi haun tulokset on mahdollista asettaa kirjanmerkkinä.

Kuten aikaisemmat heuristiikat ovat jo painottaneet, käyttäjän tulisi olla selvillä sijainnistaan sivustolla (1. heuristiikka) ja hänen pitäisi pystyä selkeästi navigoimaan sivustolla (3. heuristiikka). Kahdeksas heuristiikka (Nielsen, 1993) painottaa esteettistä ja minimaalista suunnittelua. Liiallinen ja asiaankuulumaton tieto web-sovelluksessa on käyttäjälle häiriö ja hidastaa sovelluksen käyttöä. Myös tiedon esittämistä asteittain kasvavassa järjestyksessä suositellaan. Tällöin käyttäjälle relevantein tieto löytyy nopealla silmäyksellä ja tarkempi tieto syvemmällä lukemisella. Minimalistisen suunnittelun yhteydessä on hyvä myös huomioida, että sisältö on sopiva web-ympäristöön. Looginen järjestys, tiedon jakaminen eri osiin ja niiden yhdistäminen linkeillä auttaa tämän tavoitteen saavuttamisessa (Instone, 1997).

Vaikka Nielsen (1994) määritteli, että tuen ja ohjeiden tulee olla heti käyttäjän saatavilla, Instone ehdottaa, että hyvin yksinkertaisille sivuille ohjeet eivät välttämättä ole tarpeellisia, mutta painottaa silti, että hiemankaan monimutkaisemmille sovelluksille ohjeiden tulisi olla selvästi esillä, ja niiden pitäisi olla eksakteja siihen tilanteeseen liittyen. Ohjeiden tulisi olla spesifisiä ongelmaan liittyen, eikä yhdessä suuressa dokumentissa, josta käyttäjä joutuu vaivautumaan niiden etsintään (Instone, 1997).

Instone sovelletut web-heuristiikat poikkeavat jossain määrin Nielsenin heuristiikoista, mutta De Jong (2000) olettaa, että Instone heuristiikat perustuvat käytännön web-design harjoitukseen, eivätkä tieteellisiin tutkimuksiin, kuten Nielsenin alkuperäiset heuristiikat. Heuristiikkojen käytön arvo on sitä suurempi, mitä enemmän kokemusta suunnittelija osaa tuoda niiden soveltamiseen (De Jong, 2000, s. 320).

## 4.2 Heuristiikat ja web-sovellusten houkuttelevuus

Nielsenin heuristiikat ovat luonteeltaan yleisiä, ja suuri osa Instone (1997) heuristiikoista antaa vain hieman ohjeita ja suuntaa web-käytettävyyteen. Molemmat mainitsevat minimalistisen ja esteettisen suunnittelun tärkeyden, mutta varsinaisia ohjeistuksia sen toteuttamiselle ei anneta. Web-käyttöliittymät haastavat perinteiset sovellusohjelmistojen käyttöliittymät, minkä vuoksi sitä varten on kehitetty kolmivaiheinen malli web-sovellusten ja -sivustojen houkuttelevuutta koskien. Lisäksi käytettävyyden käsitettä täytyy määritellä tarkemmin, jotta sitä voidaan soveltaa tehokkaasti web-kontekstissa. (Sutcliffe, 2001).

Web-käytettävyyden suunnittelu painottuu usein käytön intuitiivisuuden suunnitteluun. Nämä ovat edelleen tärkeitä ominaisuuksia, mutta sivuston ulkoasu ja vetovoimaisuus ovat usein ensimmäiset tekijät, joiden perusteella käyttäjän oleskeluaika (engl. *Dwell time*) määräytyy. Tämä rohkaisee käyttäjää tutkimaan sovellusta pidempään ja ominaisuudet, kuten käytön nopeus ja sivustolla navigointi tulevat tärkeiksi ominaisuuksiksi vasta käytön aloittamisen jälkeen. (Sutcliffe, 2001).

Sutcliffe (2001) määrittää web-sovellusten houkuttelevuuden (engl. *Attractiveness*) koostuvan kolmesta eri muuttujasta tai vaiheesta:



- *Innostumisella* hän tarkoittaa käyttäjän kiinnostuksen herättämistä tai jännitystä sivuston käyttöä kohtaan. Tämä ominaisuus tukee käyttäjän ongelman ratkaisua ja innostaa käyttäjää selaamaan sivustoa.
- *Motivaatio* ilmenee käyttäjän halusta käyttää web-sivustoa. Siihen voi liittyä lukuisia tekijöitä, kuten tarve ostaa tuotteita, itsetunto, uteliaisuus tai halu oppia uusia asioita. Motivaation mittaaminen on käytettävyyden kannalta vaikeaa, koska käyttäjien väliset henkilökohtaiset erot ovat vaikeita tunnistaa. Motivaatiota suunnitelmassa kannattaa Sutcliffen mukaan keskittyä käyttäjäryhmien motivaatioon.
- *Havaittu käyttökelpoisuus* tarkoittaa käyttäjän näkemystä sivuston hyödyntämisestä, ja se koostuu motivaatiosta ja innostumisesta.

Näiden ominaisuuksien avulla Sutcliffe (2001) luo pohjan heuristiikoille, joilla web-sivuston käytettävyyttä ja houkuttelevuutta voidaan arvioida ja parantaa visuaalisen suunnittelun kautta. Sutcliffe hyödyntää myös muiden tutkijoiden tekemiä määritelmiä ja suunnitelmia, kuten Nielsenin (1993) heuristiikkoja ja Kristofin (1995) kirjaa nimeltä *Interactivity by design: creating and communicating with new media*. Sutcliffe jakaa määrittelemänsä heuristiikat yleisiin ja sisältö-riippuvaisiin heuristiikkoihin. Hän määrittelee muutamia yleisiä ohjeita web-sivustojen houkuttelevuudelle ja esteettisyydelle:

- *Järkevä värien käyttö*: Värien tulisi olla tasapainossa, taustavärien tulisi olla värikylläisyydeltään alhaisia, ja korkeita värikylläisiä värejä ei kannata käyttää useampaa kuin 2-3 väriä samaan aikaan.
- *Symmetrisyyteen* tulisi tähdätä, mutta eri muotoja voi hyödyntää paremman ja muistettavamman visuaalisen ilmeen saavuttamiseksi.
- *Jäsennelty ja johdonmukainen sijoittelu*: Kuten Instone (1993) ja Nielsenin heuristiikkoja soveltaessaan mainitsi, sivuston sisällön tulisi olla sijoitettu loogisesti ja johdonmukaisesti. Tähän ohjeistukseen voidaan lisätä, että ruudukon (engl. *Grid*) käyttö on suositeltavaa, ja niiden tulee olla suorakulmioita, joiden korkeus-leveys suhde ei ole yli 5:3 (Sutcliffe, 2001).
- *Tasojen hyödyntäminen* (engl. *Depth of Field*) kuvien yhteydessä kiinnittää huomion paremmin. Esimerkiksi haalennetun kuvan käyttäminen taustana tuo lisäsyvyyttä sen päälle aseteltuihin elementteihin.
- *Median hyödyntäminen* on hyvä työkalu käyttäjän huomion kiinnittämiseksi; kuvat, musiikki ja videot vetävät huomion puoleensa paremmin kuin staattinen teksti.
- *Personalisointi*: Sivusto voidaan saada tuntumaan inhimillisemmältä ihmisten kuvien ja puheen avulla. Etenkin e-kauppasovelluksissa kohtelias kielen käyttö ja käyttäjän kehuminen ovat hyviä keinoja tuotteiden oston rohkaisuun.

- Epätavallisten tai haastavien kuvien käyttö lisää muistettavuutta ja houkuttelevuutta.

Erityyppiset käyttöliittymät tekevät usein jo suuren vaikutuksen käyttäjän mielipiteeseen sen käytettävyydestä. Vuonna 2006 tehdyssä tutkimuksessa saman web-sivuston sisältö esitettiin käyttäjille erityyppisissä käyttöliittymissä. Perinteisen valikkopohjaisen käyttöliittymän nähtiin saavan parempia käytettävyyssarvioita, mutta metaforiin perustuvan, tyylieltyemmän käyttöliittymän todettiin olevan visuaalisesti miellyttävämpi (De Angeli et al. 2006).

Sutcliffen yleiset heuristiikat sopivat hyvin käyttäjän ensireaktion arvioimiseen ja houkuttelevuuden arvioimiseen, mutta ne yksin eivät riitä sivuston käytettävyyden arviointiin Sutcliffe (2001). Web-sivuston houkuttelevuuden kannalta sisältö on kuitenkin ratkaiseva tekijä (Bruno et al. 2005), joten Sutcliffe määritteli myös heuristiikkoja, jotka liittyvät nimenomaan web-sivun sisällön esittämiseen ja laatuun. Näissä ohjeistuksissa hän painottaa tähtäämistä sisällön yhdenmukaiseen visuaaliseen tyyliin, sivuston identiteetin ja brändin näkyvyyteen, ja sisällön yhdistämiseen käyttäjän motivaatioon, aikeisiin ja vaatimuksiin. (Sutcliffe, 2001)

### 4.3 Heuristiikkojen kritiikkiä ja heikkouksia

Heuristiikkoihin on jo pitkään liitetty useita heikkouksia ja ominaisuuksia, joita heuristisella evaluoinnilla on usein vaikea ratkaista. Vaikka yksi arvioija voi löytää heuristiikkojen avulla vakaviakin käytettävyysongelmia, hän ei kuitenkaan voi ottaa huomioon kaikkia sovelluksessa olevia käytettävyysongelmia. Yhden evaluoijan arvioidaan löytävän noin 35% sovelluksen käytettävyysongelmista heuristiikkojen avulla. Vaikka heuristinen evaluointi ei ole systemaattinen lähestymistapa löytää ratkaisuja käytettävyysongelmiin, vaan nimenomaan löytää ja tulkita käytettävyysongelmia, moniin tunnistettuihin käytettävyysongelmiin voidaan usein löytää korjaus helposti ja nopeasti. Heuristinen arviointi käytettävyyden tarkastusmetodinä vaatii kuitenkin osaamista ja ymmärrystä käytettävyyssperiaatteista sekä web-kontekstin tuntemusta. (Nielsen, 1993, s. 156-159; Matera, 2006, s. 17-18).

Nielsenin mukaan kokemattomat evaluoijat tuottavat huonoja tuloksia, ja kokeneiden käytettävyyssiantuntijoiden tuottamat arvioinnit ovat lähes kaksi kertaa parempia heuristisessa evaluoinnissa verrattuna noviisievaluoijiin. Evaluoijiksi sopivat parhaiten ns. kaksinkertaiset asiantuntijat, joilla on kokemusta sekä käytettävyyden arvioinnista että sovellusalueesta (Nielsen, 1993, s. 161).

Hvannberg, Law ja Lérusdóttir (2007) tutkivat Nielsenin ja Gerhardt-Powalsin heuristiikkojen käytännön eroja heuristisessa evaluoinnissa sekä pyrkivät selittämään noviisiarvioijien ongelmia heuristisen evaluoinnin soveltamisessa. Tutkimuksen tuloksista he tulkitsivat, että Gerhardt-Powalsin heuristiikat vaikeuttivat arviointia enemmän kuin Nielsenin. Arvioijat kokivat ongelmia myös kokemuksen ja aikavaatimusten kanssa molempia heuristiikkoja hyödyntäessään.

## 5. Yhteenveto

Tämän kirjallisuuskatsauksen tarkoituksena oli tutkia, miten heuristista evaluointia hyödynnetään käytettävyyden arviointimenetelmänä web-sovelluksille ja -sivustoille. Lisäksi oli tarpeellista yrittää selvittää, millaisia perinteisistä sovelluksista poikkeavia vaatimuksia web-ympäristö asettaa käytettävyyden arvioinnille ja suunnittelulle.

Tutkimuskysymykset kirjallisuuskatsaukselle olivat seuraavat:

- *Asettaako web-ympäristö haasteita ohjelmistojen yleisille käytettävyyden suunnittelu- ja arviointiperiaatteille vai soveltuvatko ne sellaisenaan web-kontekstiin?*
- *Millaisia heuristiikkoja ja ohjeita on sovellettu web-käytettävyyden heuristiseen evaluointiin?*

Heuristisen evaluoinnin on todettu olevan yksi hinta-tehokkuus suhteeltaan parhaista käytettävyyden arviointimenetelmistä, mutta heuristiikat ovat usein yleisellä tasolla, ja niiden soveltaminen vaatii paljon kokemusta ja toimialan tuntemusta. Ohjelmistojen yleiset käytettävyyden arviointiperiaatteet, kuten Nielsenin heuristiikat ja heuristinen evaluointi, antavat hyvän perustan käytettävyyden arvioinnille, mutta web-sovellusten käytettävyyden voidaan olettaa poikkeavan eri määrissä perinteisten ohjelmistojen käytettävyydestä (Nielsen, 1993; Matera ym., 2006).

Kysymys web-sovellusten eroista perinteisiin sovellusohjelmiin on tärkeä, mutta tässä kirjallisuuskatsauksessa keskityttiin erityisesti web-sovelluksien heurististen evaluoinnin käytäntöihin ja yksityiskohtiin. Osa esitellyistä heuristiikoista saa paljon vaikutteita Nielsenin (1993 ja 1994) heuristiikoista, ja tarjoaa pintaraapaisun syvempään web-suunnitteluun (De Jong & Van der Geset, 2000).

Heuristiikkoja on tarjolla suunnittelijoille eri tarkkuuksilla ja esitystavoilla. Nielsenin (1993 ja 1994) heuristiikat ovat hyvin yleispäteviä ja yksinkertaisia, joten niiden soveltaminen tarkempana eri konteksteihin on hyvinkin mahdollista. Kun heuristiikkoja tarkennetaan, kuten esimerkiksi tässä kirjallisuuskatsauksessa esitelty Sutcliffen (2001) heuristiikat web-sivustojen houkuttelevuudelle, niiden pidemmälle soveltaminen voi vaikeutua. Ehkä Nielsenin heuristiikat ovat saavuttaneet suuren suosion juuri tästä syystä. De Jong ja Van der Geset (2001) mainitsevat, että heuristiikkoja on mahdollista esittää kolmella eri tavalla: ohjeina, kysymyksinä ja vaatimuksina. Vaikka näiden eri esitystapojen vaikutuksiin evaluointia kohtaan ei ole tässä kirjallisuuskatsauksessa kiinnitetty erityistä huomiota, De Jong ja Van der Geset (2001) olettavat niillä olevan merkitystä arvioijaan ja lopullisen arvioinnin tulokseen. Eri tyyppisiä heuristiikkoja on mahdollista soveltaa eri vaiheessa web-suunnittelu prosessia, esimerkiksi ohjeistavat heuristiikat sopivat paremmin suunnitteluvaiheeseen, kun taas kysymyksinä esitetyt heuristiikkoja voidaan käyttää paremmin juuri evaluointivaiheeseen.

Nielsenin tai Gerhardt-Powalsinkaan heuristiikat eivät ole syntyneet tyhjästä, vaan ne ovat kehittyneet muiden tutkimusten ja ohjeistusten pohjalta. Vaikka nämä kaksi yleisemmin tunnettua heuristiikkalistaa sisältävät osittain samoja asioita, ne lähestyvät

käytettävyyden arviointia silti hieman eri näkökulmasta, jonka tuloksena myös lopulliset heuristiikat ovat erilaisia. Riittävätkö nämä yleispätevät heuristiikat? Nielsenin heuristiikat julkaistiin yli kaksikymmentä vuotta sitten, ja niitä käytetään monissa lähteissä edelleen. Ehkä ne riittävät asiantuntijoille, ja ohjeistuksien soveltaminen oikealla tavalla jätetään heidän kokemuksensa varaan.

Heuristiikkoja on sovellettu moniin ongelmiin ja tilanteisiin. Niiden pohjalta määriteltyjä web-heuristiikkoja ovat esimerkiksi K. Instonen määrittelemät käytettävyysheuristiikat (Instone, 1997), jotka perustuvat Nielsenin heuristiikkoihin. Heuristiikat eivät esiinny pelkästään tutkimuksissa ja akateemisissa julkaisuissa, vaan ne ovat vuosien aikana muodostaneet pohjan eri standardeille ja ohjeistuksille. Esimerkiksi Nielsenin ja Molichin alkuperäisiä heuristiikkoja on sovellettu jo vuonna 1996 Ameritechin ohjeistuksessa ohjelmistokehittäjille (Schumacher, 1996). Vaikka heuristista evaluointia suositellaan kokeneille käytettävyydasiantuntijoille, se on kuitenkin halpa ja tehokas työkalu myös vasta-alkaville web-suunnittelijoille ja -kehittäjille.

Tämä kirjallisuuskatsaus on aineiston laajuuden huomioon ottaen hyvin suppea. Siinä keskitytään yleisiin heuristiikkoihin ja niiden soveltamiseen web-kontekstissa, sekä tutkitaan web-sovellusten houkuttelevuuteen ja käyttöön liittyviä heuristiikkoja hieman tarkemmin. Materiaalia ja viittauksia muihin tutkimuksiin on runsaasti, mutta niitä kaikkia ei ole mahdollista sovittaa noin kahteenkymmeneen sivuun. Suurin osa viittauksista ja lähteistä oli saatavilla Oulun Yliopiston tunnuksilla, joten viitteiden hankinnassa ei ollut paljon ongelmia, mutta laaja määrä tietoa tähän aiheeseen liittyen johtaa suppeaan esitystapaan. Lisäksi lienee syytä ilmoittaa, että vaikka tässä kirjallisuuskatsauksessa keskitytään web-sovelluksiin ja niiden heuristiseen evaluointiin, se on vain yksi monista arviointimenetelmistä, ja tähän sovellusalueeseen sopivat monet muutkin menetelmät.

Itseäni jatkotutkimusta varten kiinnostavat etenkin suunnitteluratkaisujen arvioiminen sekä nimenomaan keskittyminen niihin suunnitteluratkaisuihin, jotka vaikuttavat käyttäjien mielipiteisiin, käyttäytymiseen ja toimintaan. Tulevan tutkimuksen kannalta, esimerkiksi pro gradu -tutkielmassa, heuristiikat itsessään ovat niin laaja aihe, että niiden syvälinen tutkiminen on haaste, mutta varsinaisiin käytännön toteutuksiin liittyvä tutkiminen ja arviointi ovat tähtäimessäni tulevaisuutta varten.

## Lähteet

- Bruno, V., Tam, A., & Thom, J. (2005). Characteristics of web applications that affect usability: A review. Proceedings of the 17th Australia Conference on Computer-Human Interaction: Citizens Online: Considerations for Today and the Future, Canberra, Australia. 1-4.
- De Angeli, A., Sutcliffe, A., & Hartmann, J. (2006, June). Interaction, usability and aesthetics: what influences users' preferences? In Proceedings of the 6th conference on Designing Interactive systems (pp. 271-280). ACM.
- De Jong, M., & Van Der Geset, T. (2000). Characterizing web heuristics. *Technical Communication*, 47(3), 311-325.
- Fernandez, A., Insfran, E., & Abrahão, S. (2011). Usability evaluation methods for the web: A systematic mapping study. *Information and Software Technology*, 53(8), 789-817
- Gerhardt-Powals, J. (1996). Cognitive engineering principles for enhancing human-computer performance. *International Journal of Human-Computer Interaction*, 8(2), 189-211.
- Hvannberg, E. T., Law, E. L. C., & Lérusdóttir, M. K. (2007). Heuristic evaluation: Comparing ways of finding and reporting usability problems. *Interacting with computers*, 19(2), 225-240.
- IEEE standard glossary of software engineering terminology (1990). doi:10.1109/IEEESTD.1990.101064
- Ivory, M. Y. and Hearst, M. A. (2001) The state of the art in automating usability evaluation of user interfaces. *ACM Comput. Surv.* 33 (4), 470–516
- Kristof, R. & Satran, A. (1995) *Interactivity by design: creating and communicating with new media*. Adobe Press, Mountain View CA.
- Nielsen, J. (1992). Finding usability problems through heuristic evaluation. *Conference on Human Factors in Computing Systems - Proceedings*, Bellcore, Morristown, United States. 373-380.
- Nielsen, J. (1993). *Usability Engineering*. AP Professional.
- Nielsen, J., & Mack, R. L. (cop. 1994). *Usability inspection methods*. New York: Wiley.
- Nielsen, J (1994). *10 Usability Heuristics for User Interface Design*. Lainattu 7.1.2016, saatavilla: <https://www.nngroup.com/articles/ten-usability-heuristics/>
- Nielsen, J. (2000). *Designing Web Usability*. New Riders Publishing.
- Nielsen, J., & Loranger, H. (2006) *Prioritizing Web Usability*. New Riders Publishing
- Nielsen, J. (2001). *Error message guidelines*. Lainattu 14.11.2016, saatavilla: <https://www.nngroup.com/articles/error-message-guidelines/>
- Nielsen, J. (2009). *Discount Usability: 20 years*. Lainattu 6.1.2016, saatavilla: <https://www.nngroup.com/articles/discount-usability-20-years/>

Marghescu, D. (2009). Usability evaluation of information systems: A review of five international standards. In *Information Systems Development* (pp. 131-142). Springer US.

Matera, M., Rizzo, F., & Carughi, G. T. (2006). Web usability: Principles and evaluation methods. In *Web engineering* (pp. 143-180). Springer Berlin Heidelberg.

Oinas-Kukkonen, H., & Oinas-Kukkonen H. (2013). *Humanizing the Web*. Palgrave Macmillan.

Schumacher, R. (1996). *Ameritech graphical user interface standards and design guidelines*. Lainattu 30.11.2016, saatavilla: <http://www.theuxbookmark.com/wp-content/uploads/2010/08/ameritech-gui-std-design-guidelines.pdf>

Suomen Standardoimisliitto SFS. (2010). SFS-EN ISO 9421-210: *Ihmisen ja järjestelmän vuorovaikutuksen ergonomia. Osa 210: Vuorovaikutteisten järjestelmien käyttäjäkeskeinen suunnittelu*.

Sutcliffe, A. (2001). *Heuristic evaluation of website attractiveness and usability*.