

Antti K. Koskela

**KETTERÄN OHJELMISTOKEHITYKSEN JA KEVYEN
KÄYTETTÄVYYSTESTAUKSEN
YHTEENSOVITTAMINEN: TAPAUSTUTKIMUS**



JYVÄSKYLÄN YLIOPISTO
TIETOJENKÄSITTELYTIETEIDEN LAITOS
2014

TIIVISTELMÄ

Koskela, Antti K.

Ketterän ohjelmistokehityksen ja kevyen käytettävyydestaustuksen yhteensovittaminen: tapaustutkimus

Jyväskylä: Jyväskylän yliopisto, 2014, 213 s.

Tietojärjestelmätiede, pro gradu -tutkielma

Ohjaaja: Leppänen, Mauri

Tutkielman aiheena on kevyen käytettävyydestaustuksen ja ketterän ohjelmistokehityksen yhteensovitus. Käytettävyydestaustaus on aiemmin ollut kalliissa käytettävyydelaboratorioissa harjoitettavaa ”salatiedettä”. Kevyiden käytettävyydestaustuksen menetelmien myötä on tarjoutunut aiempaa nopeampia ja kevyempiä tapoja suorittaa käytettävyydestaustaus, mutta niiden hyödyntäminen osana ketterää ohjelmistokehitystä on vielä vähäistä.

Tämän tutkimuksen tarkoituksena on selvittää, millä tavalla käytettävyydestaustaa voidaan tehdä ketterän ohjelmistokehityksen yhteydessä ja mitä hyötyjä ja kustannuksia tästä aiheutuu. Tutkimus koostuu kahdesta osasta, kirjallisuuskatsauksesta ja empiirisestä osasta. Kirjallisuuskatsaus käsittelee ketterän ohjelmistokehityksen piirteitä ja Scrum-menetelmää, käytettävyydestaustaa sekä tapoja, joilla käytettävyydestaustaa, erityisesti kevyitä menetelmiä, on pyritty aiemmin sovittamaan ketterään ohjelmistokehitykseen. Empiirinen osuus on toteutettu tapaustutkimuksena, jossa rakennetaan uusi käytettävyydestaustamalli, Nielsen+Krug-malli, integroidaan se Oy Samlink Ab:n (jäljempänä Samlink) SamScrum-projektimalliin sekä kokeillaan mallin toimivuutta neljän todellisen ohjelmistokehitysprojektin yhteydessä.

Nielsen+Krug-malli sisältää hyödynnettävinä menetelminä korttien lajittelun, heuristisen evaluoinnin ja yksinkertaistetun ääneen ajattelun. Mallin soveltamisen huomattiin tuottavan aiempia malleja edullisemmin hyödyllisiä havaintoja testattavana olevan järjestelmän käytettävyydestä, mutta lisäksi myös tietoa järjestelmän virheistä ja jatkokehitysmahdollisuuksista. Tutkimus tarjoaa ohjeet mallin hyödyntämiseksi sekä myöhemmissä tutkimuksissa että käytännön työssä, ja iteratiivista ohjelmistokehitystapaa soveltavissa organisaatioissa se voi tuottaa samankaltaisia havaintoja kuin tässä tutkimuksessa.

Asiasanat: ketterät menetelmät, scrum, käytettävyys, kevyt käytettävyydestaustaus, tapaustutkimus

ABSTRACT

Koskela, Antti K.

Integrating Discount Usability Engineering in Agile Software Development: Case Study

Jyväskylä: University of Jyväskylä, 2014, 213p.

Information Systems Science, Master's Thesis

Supervisor: Leppänen, Mauri

The subject of the thesis is the integration of discount usability engineering into agile software development. Earlier, usability testing used to be arcane science conducted in expensive usability laboratories. With the rise of more lightweight methods, usability testing can be done with much lower costs. However, incorporating them in agile software development seems to still be in its infancy.

This study consists of two distinct phases. First, in the literature review, a closer look is taken at the agile software development, especially Scrum, and the history and features of usability testing and engineering. Discount usability engineering is also introduced. A closer look is taken at some methods to integrate usability testing, especially discount usability engineering methods, to agile software development. The second part is a case study, where a new lightweight usability testing model, named Nielsen+Krug-model, is introduced and integrated into Samlink's software development framework, SamScrum. Nielsen+Krug-model is tested in four software development projects.

Nielsen+Krug-model includes such usability testing methods as card sorting, heuristic evaluation and simplified thinking aloud. Applying the model produced very encouraging results: it was significantly cheaper than earlier models and was not only able to generate findings about the usability of the systems, but also some bugs and feature requests as well. The findings were also generally well received in the project teams. This study offers instructions for using this model in the future research as well as practice. In organizations that are using iterative software development methods this model is expected to produce similar results.

Keywords: agile methods, Scrum, usability, discount usability engineering, case study

KUVIOT

KUVIO 1 Kuvaus scrumin iteratiivisesta työtavasta.....	21
KUVIO 2 Ketterän testauksen osa-alueet.....	22
KUVIO 3 Käytettävyyssuunnitteluparadigma.....	30
KUVIO 4 Käytettävyyssuunnittelun, -testauksen ja -arvioinnin suhde.....	31
KUVIO 5 Scrum ja käytettävyystestaus Summa-techin toteutustavassa	58
KUVIO 6 Ongelmien löytymisen ja koehenkilöiden määrän suhde.....	66
KUVIO 7 Koehenkilöiden määrän vaikutus ongelmien löytymiseen	67
KUVIO 8 Tutkimusprosessin kulku	83
KUVIO 9 SamScrum-projektimalli.....	96
KUVIO 10 SamScrum-malli, johon on sovitettu kevyt käytettävyystestaus	101
KUVIO 11 Käytettävyysongelmien löytyminen heuristisessa arvioinnissa	105
KUVIO 12 Käytettävyystestaussession läpivienti.....	108
KUVIO 13 Samlinkin käytettävyystestausprosessi	110
KUVIO 14 Käytettävyystestausmenetelmien projektikohtainen valinta	113
KUVIO 15 Virheiden ja jatkokehityspiirteiden osuus havainnoista	135
KUVIO 16 Hyväksytyjen ja hylättyjen havaintojen suhde eri projekteissa.....	138
KUVIO 17 Käytettävyystestauksen havaintojen tyypit eri projekteissa.....	139
KUVIO 18 Käytettävyystestauksen loppukyselyyn vastaajien rooli	141

TAULUKOT

TAULUKKO 1 Testauksen ja ketterien menetelmien vastakkaiset periaatteet..	24
TAULUKKO 2 Käytettävyyssmittarit eri tahojen mukaan	27
TAULUKKO 3 Esimerkki käyttäjätyyppien jaottelusta.....	45
TAULUKKO 4 Yhteen sovitettavia elementtejä ketterässä kehityksessä ja käyttöliittymäsuunnittelussa	52
TAULUKKO 5 Käytettävyystestauksen hinta eri menetelmillä	69
TAULUKKO 6 Nielsenin heuristiikat täydennettynä kuvauksilla	104
TAULUKKO 7 Projekteissa hyödynnetyt testausmenetelmät ja koehenkilöiden lukumäärät	124
TAULUKKO 8 Käytettävyystestaukseen kulunut aika iteraatiokohtaisesti eriteltynä.....	129
TAULUKKO 9 Käytettävyystestaukseen kulunut aika koehenkilöä kohden.....	129
TAULUKKO 10 Käytettävyystestauskertaan kulunut aika	130
TAULUKKO 11 Käytettävyystestaukseen havaintokohtaisesti kulunut aika..	131
TAULUKKO 12 Käytettävyystestauksen kustannukset sprinttiä kohden.....	133
TAULUKKO 13 Käytettävyystestauksen lopullinen hinta eri projekteissa.....	133
TAULUKKO 14 Havaintokohtainen hinta projekteissa.....	134

TAULUKKO 15 Käytettävyytestauksen havaintojen lukumäärä ja tyyppi eri sprinteissä.....	134
TAULUKKO 16 Käytettävyytestauksen havaintojen hyödyntäminen eri sprinteissä	136
TAULUKKO 17 Käytettävyytestauksen havaintojen hyväksyntä projekteittain ja iteraatioittain	137
TAULUKKO 18 Käytettävyytestauksen havaintojen hyväksyntä projekteittain	137
TAULUKKO 19 Loppupalautekyselyn tuloksia projekteittain	142

SISÄLLYS

1	JOHDANTO.....	10
1.1	Tutkimuksen tausta ja motivaatio.....	10
1.2	Tutkimustavoitteet, -ongelma ja -menetelmät.....	12
1.3	Tutkielman rakenne	14
2	KETTERÄ OHJELMISTOKEHITYS	15
2.1	Ketterä lähestymistapa.....	15
2.1.1	Aika ennen ketteryyttä	15
2.1.2	Agile-manifesti	17
2.1.3	Kuriositeeteista valtavirran käytännöiksi.....	18
2.2	Scrum esimerkkinä ketterästä kehitystavasta	20
2.3	Testaus ketterässä ohjelmistokehityksessä	21
2.4	Yhteenveto	24
3	KÄYTETTÄVYYSTESTAUS	26
3.1	Käytettävyys.....	26
3.2	Käytettävyystestaus osana suunnittelua	29
3.3	Käytettävyystestauksen tyyppejä ja menetelmiä	34
3.3.1	Käytettävyystestauksen tyyppejä	34
3.3.2	Käytettävyyden arvioimisen ja testaamisen menetelmiä.....	35
3.4	Kevyt käytettävyystestaus.....	38
3.4.1	Käytettävyystestauksen kehitys ja uusien ajatusten nousu	38
3.4.2	Käytettävyystestauksen murros ja kevyempien metodien nousu	39
3.4.3	Kevyen käytettävyystestauksen menetelmiä	40
3.4.4	Käytettävyystestauksen tulosten huomiointi.....	41
3.5	Käytettävyystestaukseen osallistuvat henkilöt	42
3.5.1	Käytettävyystestaaja	42
3.5.2	Koehenkilöt	43
3.5.3	Koehenkilöiden kohderyhmävastaavuus.....	44
3.6	Yhteenveto	45
4	KÄYTETTÄVYYSTESTAUS OSANA KETTERÄÄ OHJELMISTOKEHITYSTÄ	47
4.1	Ketterä käytettävyystestaus?.....	47
4.1.1	Käytettävyystestaus ketterässä ohjelmistokehityksessä.....	47
4.1.2	Käytettävyystestauksen ja ketterän ohjelmistokehityksen integrointi	49
4.2	Tapaustutkimuksia käytettävyystestauksen ja ketterän ohjelmistokehityksen integroinnista	53
4.2.1	XSBD-malli	53

4.2.2	RITE+Krug -menetelmä	54
4.2.3	Parsonsin, Lalin, Ryuin ja Langen esittämä integrointitapa	55
4.2.4	U-scrum-lähestymistapa	55
4.2.5	Autodesk.....	56
4.2.6	Skenaarioperusteisen suunnittelun ja XP:n yhdistäminen	57
4.2.7	Käytettävyydestaus Scrumin osana	57
4.2.8	UTUM-käytettävyydestausratkaisu.....	58
4.2.9	Synteesi ja johtopäätöksiä	59
4.3	Käytettävyydmetriikoiden hyödyntäminen.....	61
4.4	Testauksen ajoitus.....	63
4.5	Testauksen määrä	65
4.5.1	Koehenkilöiden määrä.....	65
4.5.2	Koehenkilöiden tyyppi.....	67
4.6	Käytettävyydestauksen hinta	68
4.7	Käytettävyydestauksen käytännön toteutuksesta.....	70
4.8	Käytettävyydestauksen sudenkuoppia.....	71
4.8.1	Suunnittelu ja toteutus saman sprintin aikana epäonnistuu	71
4.8.2	Onko järjestelmä käyttökelpoinen?	72
4.8.3	Testitapausten kattavuus käyttäjätarinoiden perusteella	72
4.8.4	Persoonien käyttö ja kohderyhmävastaavuuden merkitys	73
4.8.5	Käytettävyydestauksen tulosten luotettavuus	73
4.8.6	Käytettävyyden tärkeyden aliarviointi tai tulosten kyseenalaistaminen.....	74
4.9	Käytettävyydestauksen havaintojen validointi	74
4.10	Yhteenveto	77
5	TAPAUSTUTKIMUKSEN TOTEUTTAMINEN.....	80
5.1	Tutkimusmenetelmä	80
5.2	Tutkimuksen kohde.....	81
5.3	Tutkimusprosessi.....	82
5.4	Projektit joissa testausta tehdään.....	83
5.4.1	Sisäinen työsuunnittelun apuväline, RETU	84
5.4.2	Finanssialan asiakkaiden ajanvarausjärjestelmä, PA.....	84
5.4.3	Energia-alan asiakkaiden asiakaspalveluväylä, SH.....	85
5.4.4	Elintarvike-alan yritys, HE	85
5.5	Tiedonkeruu ja analysointi.....	85
5.5.1	Käytettävyydestauksen kustannukset.....	86
5.5.2	Käytettävyydestauksen hyödyllisyys.....	87
5.6	Testauksen toteutus.....	88
5.6.1	Käytettävyydestausmenetelmien testaus.....	88
5.6.2	Käytettävyydestauksen valmistelu	89
5.6.3	Koehenkilöiden rekrytointi.....	90
5.7	Nielsen+Krug-mallin toimivuuden validointi	91
5.8	Yhteenveto	92

6	SAMSCRUM JA NIELSEN+KRUG -MALLI.....	94
6.1	SamScrum	94
6.2	Käytettävyystestausmalli integroituna SamScrumiin.....	97
6.2.1	Vaatimukset Samlinkin käytettävyystestausmallille	98
6.2.2	Nielsen+Krug-malli	99
6.3	Testausmenetelmien kuvaus.....	102
6.3.1	Korttien lajittelu ja sen järjestäminen	102
6.3.2	Heuristinen arviointi ja sen suorittaminen.....	103
6.3.3	Yksinkertaistettu ääneen ajattelu	105
6.3.4	Käytettävyystestaussession kulku	107
6.4	Testauskäytäntöjä	108
6.4.1	Yleiskuvaus käytettävyystestausprosessista.....	109
6.4.2	Käytettävyystestausmenetelmien valinta.....	111
6.4.3	Koehenkilöiden rekrytointi.....	114
6.4.4	Testauksen loppukysely	114
6.5	Testaustulosten raportointi	115
6.5.1	Aiempi tutkimus	116
6.5.2	Raporttimalli	117
6.6	Yhteenveto	118
7	TUTKIMUSTULOKSET	119
7.1	Käytettävyystestauksen suorittamisesta yleisesti.....	119
7.1.1	Yleisvalmistelut	120
7.1.2	Koehenkilöiden rekrytointi koehenkilökantaa varten.....	120
7.1.3	Käytettävyystestauskerran valmistelu.....	120
7.1.4	Käytettävyystestaussession valmistelu ja kulku	121
7.1.5	Käytettävyystestauksen tulosten raportointimallin kehitys....	122
7.1.6	Käytettävyystestaussession tulosten käsittely	123
7.1.7	Käytettävyystestauskerran tulosten käsittely	123
7.2	Käytettävyystestauksen toteuttaminen projekteissa	124
7.2.1	RETU-projekti	124
7.2.2	PA-projekti	125
7.2.3	SH-projekti	127
7.2.4	HE-projekti	127
7.2.5	Yhteenveto.....	128
7.3	Käytettävyystestauksen kustannukset	128
7.3.1	Käytettävyystestaukseen kulunut aika	128
7.3.2	Käytettävyystestauksen kustannukset kokonaisuutena	132
7.4	Testaustulokset ja niiden hyödyllisyys.....	134
7.4.1	Havaintojen määrä ja tyyppi	134
7.4.2	Tulosten hyväksyntä ja hyödyllisyys	135
7.4.3	Saatu palaute	139
7.5	Nielsen+Krug-malli vs. Samlinkin vaatimukset	142
7.6	Yhteenveto	144

8	POHDINTA	146
8.1	Nielsen+Krug -malli	146
8.2	Kokeilu	147
	8.2.1 Kokeilun tavoitteet, lähtökohdat ja järjestelyt	148
	8.2.2 Testauksen kustannukset	148
	8.2.3 Testauksen hyödyt	151
8.3	Tulosten hyödyntäminen.....	153
8.4	Tulosten reliabiliteetti ja validiteetti	153
	8.4.1 Reliabiliteetti	154
	8.4.2 Validiteetti	154
9	YHTEENVETO	157
9.1	Tulokset ja johtopäätökset	157
9.2	Rajoitteet.....	159
9.3	Jatkotutkimusaiheet.....	160
	LÄHTEET	162
	LIITE 1 SAMLINKIN KOEHENKILÖTIETOKANNAN TIETOJEN KYSELYLOMAKE	175
	LIITE 2 KUVAUS PROJEKTEISTA	187
	LIITE 3 HAVAINTOJA KÄYTETTÄVYYSTESTAUSSESSIÖIDEN JÄRJESTELYISTÄ	189
	LIITE 4 KÄYTETTÄVYYSTESTAUSSESSION LOPPUKYSELY	193
	LIITE 5 KÄYTETTÄVYYSTESTAUKSEN ALOITUS- JA LOPETUSOHJEET ...	195
	LIITE 6 PROJEKTIRYHMIEN KOMMENTIT KÄYTETTÄVYYSTESTAUKSEN TULOKSISTA JA RAPORTEISTA ..	197
	LIITE 7 KÄYTETTÄVYYSTESTAUSRAPORTIN VERSIOT	204
	LIITE 8 KÄYTETTÄVYYSTESTAUKSEN LOPPUKYSELY	211
	LIITE 9 SOPIMUS KÄYTETTÄVYYSTESTAUKSEN TIETOJEN KÄYTÖSTÄ.	212

1 JOHDANTO

”If the user can’t use it, it doesn’t work” (Dray, 2008)

1.1 Tutkimuksen tausta ja motivaatio

Ketterä ohjelmistokehitys on viime vuosikymmenen aikana lyönyt läpi vakavasti otettavana haastajana perinteiselle vesiputousmallille (mm. Bygstad, Ghinea, & Brevik, 2008; Sohaib & Khan, 2010). Lähestymistavan mukaiset menetelmät kuten scrum (Schwaber & Sutherland, 2013) ja XP (Beck & Andres, 2005; Beck, 1999) ja Lean-ajatteluun pohjaava Kanban (Turner, Ingold, Lane, Madachy, & Anderson, 2012, s. 309–3010) on otettu yrityksissä käyttöön innolla ja useimmiten positiivisin tuloksin (Bird, 2010, s. 116; Dybå & Dingsøy, 2008, s. 843; Ghanam & Maurer, 2007, s. 8). Niillä on tavoiteltu muiden muassa kykyä reagoida nopeasti muutoksiin, nopeaa toimitusta, resurssien tehokasta käyttöä ja asiakkaalle syntyvän arvon asettamista kaiken muun edelle (Barksdale & McCrickard, 2012, s. 56). Ketterillä menetelmillä onkin saatu nostettua erityisesti tuottavuutta, asiakastyytyväisyyttä ja henkilöstön tyytyväisyyttä (Dybå & Dingsøy, 2008).

Ketterän kehittämisen menetelmille on olennaista myös ohjelmiston testaaminen, kuten XP:n yksi merkittävä osa-alue testivetoinen ohjelmistokehitys (engl. Test-Driven Development, TDD) (Beck, 1999, s. 74). Testauksen kohteena on kuitenkin pääasiassa ohjelmalogiikka. Sen sijaan menetelmien tuki ohjelmiston käytettävyyden testaamiseen on hyvin vähäistä (Constantine, 2001; Itkonen, Rautiainen, & Lassenius, 2005, s. 2–3; Kane, 2003, s. 40; Sohaib & Khan, 2010, s. 1). Useissa tapauksissa tämä on johtanut joko käytettävyydestauksen toteuttamiseen totutuilla, ketterään ja iteratiiviseen kehitykseen huonosti soveltuvilla tavoilla, käytettävyydestauksen sysäämiseen kokonaan asiakkaan hoidettavaksi tai pahimmassa tapauksessa käytettävyydestauksen unohtamiseen. Käytettävyys siis ei välttämättä ole parantunut (Constantine & Lockwood, 2003, s. 746). Ponnistelut asiakkaan saamiseksi aktiivisesti mukaan kehitystyöhön ovat toisaalta auttaneet käytettävyyden parantamisessa. (Budwig, Jeong, & Kelkar, 2009; Kane, 2003; Lee & McCrickard, 2007.)

Käytettävyys sen sijaan on likimain ajaton kysymys. New York Times nimitti käytettävyuden yhdeksi ”kuumaksi nousevaksi ammattialaksi” (engl. ”hot emerging career field”), mutta tosiasiaa alalla roikutaan pääosin vieläkin 90-luvun käytänteissä (Scott, 2009, s. 6). Moni alan uranuurtajista, kuten Jakob Nielsen, ovat esitelleet vaihtoehtoisia, usein kevyempiä käytettävyystestauksen malleja, metodologioita tai filosofioita, mutta niiden vaikutus on jäänyt vähäiseksi, ja nykyaikaisia, monista yhteen pelaavista moduuleista koostuvia järjestelmiä ja ohjelmia testataan vanhanaikaisilla ja vajavaisilla tavoilla (Scott, 2009, s. 7). Käytettävyysuunnittelusta (engl. usability engineering) on kuitenkin hiljalleen tullut valtavirtaa ja vaikka sen perspektiivi asioihin on erilainen, sillä ja ketterillä menetelmillä on sama tavoite: hyvä ohjelmistokehitys (Sohaib & Khan, 2010, s. 1).

Ohjelmisto tai järjestelmä voi olla käyttäjätarinoiden (engl. user story) (Lekman ym., 2012) perusteella näennäisen valmiiksi toteutettu, mutta kehittäjäkeskeisen lähestymistavan ja ketterien menetelmien kuten scrumin ja XP:n vahvan keskittymisen toiminnallisiin vaatimuksiin laadullisten sijasta (Sohaib & Khan, 2011, s. 1) seurauksena käyttöliittymä voi olla hajanainen ja käyttäjäkokemus heikko (Meszaros & Aston, 2006; Sohaib & Khan, 2010). Tiedemaailma on hiljalleen herännyt tunnustamaan peruseräiteiltään tai jopa kulttuureiltaan erilaisen toimintatapojen yhteensovittamisen vaikeuden (Bygstad ym., 2008, s. 375; Lee, 2006, s. 2; Sohaib & Khan, 2010, s. 5). Käytettävyysseikat, ja erityisesti -puutteet, ovat saaneet myös huomattavasti julkisuutta erityisesti VR:n alkuun pahasti epäonnistuneen verkkokauppauudistuksen seurauksena, josta vapaaehtoistyönä valmistunut käytettävyysraportti löysi alkuun toista sataa ongelmaa (Puskalahti, 2011), ja käyttäjien palautteen perusteella ongelmia löytyi vielä kymmeniä lisää (Turun Sanomat, 2012).

Toisaalta dokumentoidut kokemukset niin ketterästä kehityksestä (Layman, Williams, & Cunningham, 2006, s. 654) kuin käytettävyystestauksesta (Meszaros & Aston, 2006) sinällään ovat olleet pääosin positiivisia. Myös melko vähäiset kokemukset näiden yhteensovittamisesta ovat olleet pääosin lupaavia (McInerney & Maurer, 2005, s. 23; Singh, 2008), mutta yleisesti tunnustetaan, että erityisesti käytännön kokemuksia kaivataan lisää (Parsons, Lal, Ryu, & Lange, 2007, s. 177).

Aiemmassa tutkimuksessa käytettävyystestauksen hyötyjä on eritelty jonkin verran (mm. Bias & Mayhew, 2005; Donahue, 2001; Nielsen & Landauer, 1993; Nielsen, 1994b). Tämän tutkimuksen kaltainen lähestymistapa, siis mahdollisimman kevyen käytettävyystestausmallin löytäminen ketterään kehitykseen yhteensovittettavaksi, on kuitenkin harvinaisempi. Kevyen käytettävyystestauksen kustannuksia on eritelty melko harvoin (poikkeuksina ainakin Krug (2000), jonka mukaan kevyt käytettävyystestaus koko projektin aikana voi maksaa esimerkiksi 3 900 dollaria, ja Nielsen (1994b), joka mainitsee projektikohtaiseksi kustannukseksi 7 440 dollaria) ja perinteisen käytettävyystestauksen (mm. Krugin (2000) mukaan vähintään 5 000 dollaria testauskertaa kohden, Nielsenin ja Landauerin (1993) mukaan pienessäkin projektissa vähintään 9 400 dollaria ja jokaisen koehenkilön osalta tuhansia dollareita) kustannukset eivät ole vertailukelpoisia.

Tämä tutkimus tuottaa aiempia tutkimuksia yksityiskohtaisemman kuvan kevyen käytettävyydestäustuksen kustannuksista.

1.2 Tutkimustavoitteet, -ongelma ja -menetelmät

Tämän tutkimuksen tarkoituksena on etsiä vastausta siihen, voidaanko käytettävyydestäukseen löytää ketterään ohjelmistokehitykseen hyvin nivoutuva, todella kevyt malli, joka takaisi käytettävyydestäustuksen toteutumisen myös sellaisissa projekteissa, joissa resurssit ovat vähissä. Löytämällä tarpeeksi kevyt ja helppo toimintatapa voitaisiin varmistaa, että ainakin jonkinlainen käytettävyydestäustaus voitaisiin tehdä missä tahansa projektissa. Projekteissa, joissa resursseja on käytettävissä enemmän, voidaan käytettävyydestäustauskin toteuttaa laajemmin. Tämän tutkimuksen tarkoituksena on tuottaa aiempia tutkimuksia yksityiskohtaisempi kuva myös kevyen käytettävyydestäustuksen hyödyistä ja kustannuksista. Tutkielman tutkimusongelma voidaan ilmaista seuraavasti:

Miten kevyt käytettävyydestäustaus voidaan sovittaa ketterään ohjelmistokehitykseen?

Tutkimusongelma voidaan jäsentää seuraaviksi tutkimuskysymyksiksi:

- Millä tavalla käytettävyydestäustaus voidaan tehdä ketterän ohjelmistokehityksen yhteydessä?
- Mitä hyötyjä ja kustannuksia käytettävyydestäustuksen teosta ketterän ohjelmistokehityksen yhteydessä aiheutuu?

Tutkimuskysymyksillä pyritään saamaan konkreettista ja vertailukelpoista tietoa kevyen käytettävyydestäustuksen ja ketterän ohjelmistokehityksen yhteensovittamisesta. Tutkimuskysymysten asettamisella pyritään tavoittamaan kaksi keskeisintä ulottuvuutta tutkimusaiheesta, yhteensovituksesta saatavat hyödyt ja sen kustannukset.

Tutkimus koostuu kirjallisuuskatsaukset ja empiirisestä osuudesta. Kirjallisuuskatsauksessa tarkastellaan ketterän ohjelmistokehityksen ja käytettävyydestäustuksen historiaa ja piirteitä ja esitellään kevyt käytettävyydestäustaus. Lisäksi tarkastellaan joitakin tapoja, joilla käytettävyydestäustaus, erityisesti kevyitä menetelmiä, on pyritty aiemmin sovittamaan ketterään ohjelmistokehitykseen. Tutkimuksen empiirinen osio on toteutettu tapaustutkimuksena, jonka tavoitteena on selvittää, mitkä ovat kevyiden käytettävyydestäustausmenetelmien ketterään ohjelmistokehitykseen integroinnin hyödyt ja kustannukset käytännön tilanteissa. Tutkimusta varten kirjallisuuskatsauksen pohjalta kehitetään uusi kevyen käytettävyydestäustuksen malli (Nielsen+Krug-malli), joka integroidaan yhteistyöorganisaatio Samlinkin ketterään ohjelmistokehitysmalliin, SamScrumiin. Mallia kokeillaan neljässä keskenään erilaisessa projektissa, joiden aikana mallin kustannuksista ja hyödyllisyydestä kerätään tietoa. *Kustannuksia* arvioidaan testäuksen eri vaiheisiin kuluvalle ajalle sekä testaamisesta aiheutuvilla suorilla kustan-

nuksilla. Mallin *hyödyllisyydestä* saadaan tietoa mittaamalla käytettävyydestä havaintojen lukumäärää ja projektiryhmien reagoitua havaintoihin. Tapaustutkimuksessa havaittiin kustannusten osalta, että kokonaisuutena malli oli huomattavasti useimpia aiempia yhteensovitusmalleja edullisempi. Projektiryhmien palautteen perusteella testaus oli myös hyödyllistä. Lisäksi projektissa, jossa testaus saatiin tehtyä pisimpään, järjestelmän laatu parani sekä koehenkilöiltä kerätyn palautteen, että uusien käytettävyyssongelmien määrän vähentymisen perusteella. Tutkimuksen perusteella käytettävyydestä voidaan suositella tehtäväksi sellaisissa sovelluskehitysprojekteissa, joissa käytettävyydellä ylipäänsä on merkitystä.

Käytettävyydestä integrointia ketterän kehittämisen yhteyteen on tutkittu verraten vähän, ja usein varsin yleisellä tasolla (Ghanam & Maurer, 2007, s. 8–9). Empiirisiä tutkimuksia aiheesta on tehty vähän, ja usein tutkimuksen fokus on ollut käytettävyydestä sijasta esimerkiksi käytettävyyssuunnittelussa yleensä (mm. McInerney & Maurer, 2005; Salah, 2011) tai muussa testauksessa. Joitakin tapaustutkimuksia on tehty, joissa on saatu rohkaisevia tuloksia käytettävyydestä, tai käytettävyydestä lähellä olevan käytettävyyssuunnittelun (Sy, 2007, s. 112), huomioimisesta osana ketterää kehitystä mahdollisimman aikaisessa vaiheessa projektia, esimerkiksi paperiprototyyppejä käyttäen (Meszaros & Aston, 2006, s. 289). Lisäksi on tehty joitakin kirjallisuuskatsauksia (esim. Sohaib & Khan, 2010).

Esimerkkeinä käytettävyydestä käsittelevistä tai sivuavista empiirisistä tutkimuksista voidaan mainita Høeghin ym. (2006) vertaileva tutkimus käytettävyyssarvioinnin eri metodeista ja Meszarosin ja Astonin (2006) tutkimus käytettävyydestä sovittamisesta ketterään ohjelmistokehittämiseen. Lisäksi käytännön kokemuksia tietyn organisaation kokemuksista käytettävyydestä ja ketterien menetelmien yhteensovittamisesta on käsitelty jonkin verran (Budwig ym., 2009; Lee, McCrickard, & Stevens, 2009; McInerney & Maurer, 2005 (UCD); Sy, 2007; Talby, Hazzan, Dubinsky, & Keren, 2006; Winter, Rönkkö, Ahlberg, & Hotchkiss, 2011), ja lisäksi on tehty joitakin tutkimuksia käytettävyyden huomioimisesta tiettyä menetelmää käytettäessä (Singh, 2008).

Keveydestä käytettävyydestä ja -suunnittelusta on ensimmäiset tutkimukset julkaistu jo 90-luvulla (Cooper, 1995; Nielsen, 1995a) ja ajatuksia on erityisesti akateemisessa maailmassa viety myöhemmin myös eteenpäin (Kane, 2003, s. 40). Valitettavasti edes alan uranuurtaja Jakob Nielsenin ponnistelut käytettävyydestä arkipäiväistämiseksi ja kustannusten keventämiseksi eivät ole jostakin syystä tavoittaneet suuria massoja (Krug, 2006). Joitakin malleja tai tutkimuksia käytettävyydestä sovittamisesta ketterään sovelluskehitykseen on laadittu, osa melko kattaviakin, mutta useimmiten käytännön kokemukset mallien toiminnasta puuttuvat (Kane, 2003, s. 40).

1.3 Tutkielman rakenne

Tutkielma rakentuu yhdeksästä luvusta. Luvut 2, 3 ja 4 muodostavat kirjallisuuskatsauksen. Toisessa luvussa esitellään ketterä ohjelmistokehitys ja kuvataan scrum esimerkkinä ketterästä menetelmästä. Lisäksi tarkastellaan hiukan ketterää testausta. Kolmannessa luvussa käsitellään käytettävyydestausta sekä erikoisuutena kevyttä käytettävyydestausta. Neljännessä luvussa tarkastellaan käytettävyydestaustan ja ketterän ohjelmistokehityksen yhteensovittamista, sekä teoreettisesti että aiempien tapaustutkimusten tarkastelun kautta. Lisäksi tutustutaan tarkemmin erityisesti yhteensovituksen kannalta kriittisiin kysymyksiin, kuten testausten ajoitukseen, määrään, hintaan ja joihinkin sudenkuoppiin.

Luvut 5, 6 ja 7 käsittelevät tutkimuksen empiiristä osuutta, tapaustutkimusta kevyen käytettävyydestaustamallin ja ketterän ohjelmistokehityksen yhteensovittamisesta. Viides luku kuvaa tutkimusmenetelmän, tiedonkeruumenetelmät sekä aineiston analysointiin käytettävät menetelmät sekä projektit, joissa kehitettyä mallia hyödynnetään. Lisäksi käsitellään joitakin testausten käytännön kysymyksiä. Kuudennessa luvussa kuvataan seikkaperäisesti kehitetty käytettävyydestaustamalli, Nielsen+Krug-malli. Samoin esitetään scrumiin perustuva Samlinkin projektimalli SamScrumiin, johon integrointi tapahtuu. Nielsen+Krug-mallin testausmenetelmät ja -käytännöt kuvataan, ja erityisesti esitetään käytettävyydestaustan tulosten raportointikäytännöt. Seitsemännessä luvussa esitetään havainnot käytettävyydestaustan suorittamisesta sekä tapaustutkimuksen tulokset. Tuloksissa esitellään testausten kustannukset ja hyödyt, joita on havaittu neljässä testausmallia soveltaneessa projektissa.

Luvussa 8 esitellään tiivistetysti Nielsen+Krug-malli sekä sen soveltamisesta saadut tulokset, verrataan tuloksia aiempien tutkimusten tuloksiin, kerrotaan tämän tutkimuksen tulosten hyödyntämismahdollisuuksista sekä suoritetaan tutkimuksen reliabiliteetti- ja validiteettitarkastelu. Tutkielman viimeisessä luvussa esitetään yhteenveto, jossa luodaan tiivistetty kuva tutkimustuloksista ja niistä tehdyistä johtopäätöksistä sekä kerrotaan tutkimuksen rajoitteista ja jatko-tutkimusaiheista.

2 KETTERÄ OHJELMISTOKEHITYS

*"Vesiputousmalli on kuollut."
"Ei ole, mutta sen tulisi olla."
(Boehm, 1988, s. 1)*

Tämän luvun tarkoituksena on kuvata ketterän ohjelmistokehityksen peruspiirteet. Luvussa käsitellään ensin ketterän ohjelmistokehityksen taustaa ja periaatteita. Sitten kuvataan scrumia esimerkkinä ketterästä menetelmästä sen yleisyyden ja relevanttiuden tutkielman empiiriselle osuudelle vuoksi. Lisäksi käsitellään lyhyesti ketterää ohjelmistotestausta.

2.1 Ketterä lähestymistapa

Tämä alaluku kuvaa ketterän lähestymistavan syntyyn johtaneita syitä, arvoja ja periaatteita. Ketterän kehityksen taustalla vaikuttavaa iteratiivista ja inkrementaalista kehitystä ja sen tietynlaista antiteesiä, vesiputousmallia, käsitellään jotta saataisiin kattava käsitys ketterän ohjelmistokehityksen taustoista. Sitten tutustutaan Agile-manifestiin, joka kuvaa ketterän ohjelmistokehityksen perusteet. Lopuksi tarkastellaan ketterän ohjelmistokehityksen nousua laajasti tunnetuksi ja hyväksytyksi ohjelmistokehitystavaksi.

2.1.1 Aika ennen ketteryyttä

Perinteinen, viime vuosikymmeninä hallinnut metodi ohjelmistokehityksessä on ollut niin sanottu vesiputousmalli. Vesiputousmalli on kehitetty teollisuuden pohjalta, jossa tuotokset voidaan esimerkiksi liukuhihnatyönä valmistaa kuvatulla tavalla (Benington, 1983, s. 3). Se on puhtaimmillaan vahvasti vaiheistettu, jolloin jokainen vaihe seuraa edellistä, eikä edelliseen vaiheeseen palata (Sohaib

& Khan, 2010, s. 2). Edellisen vaiheen tuotokset toimivat seuraavan vaiheen toiminnan pohjana, kunnes määriteltyjen vaiheiden jälkeen tuotoksena on lopullinen tuote.

Käytännössä siis projektit aloitetaan vaatimusmäärittelyllä kaikille ominaisuuksille (feature), jolloin saadaan täydellinen listaus toiminnallisista ja ei-toiminnallisista ominaisuuksista. Tämän jälkeen suunnitellaan koko järjestelmän arkkitehtuuri, minkä jälkeen kehittäjät toteuttavat sen ja lopuksi on laadunvarmistus ja -testaus (Beck, 1999, s. 70; D. Cohen, Lindvall, & Costa, 2004, s. 3). Proessin kesto mitataan kuukausissa tai vuosissa. Jokaisen vaiheen tuotoksia käytetään syötteenä seuraavalle vaiheelle aina julkaisuun saakka. (Sy, 2007, s. 779)

Vesiputousmalli määriteltiin ensimmäistä kertaa ohjelmistokehityskäyttöön jo vuonna 1956 (Benington, 1956), ja se oli eri muodoissaan hallitseva menetelmä vuosikymmenten ajan. Yksi syy vesiputousmallin suureen suosioon oli sen asema Yhdysvaltain julkisella sektorilla: julkisen sektorin projekteissa vaadittu projektimalli ja dokumentaatio ohjasivat toteutuksen vesiputousmaiseen top-down -malliin (Benington, 1983, s. 3). Yhdysvaltain Puolustusministeriön (Department of Defence, DoD) asetus DOD-STD-2167A (Department of Defence, 1988) vuodelta 1988, sekä edeltäjänsä DOD-STD-2167 (Department of Defence, 1985) vuodelta 1985, käytännössä määrittelivät vesiputousmallisen ohjelmistokehityksen ainoaksi sallituksi tavaksi ohjelmistokehitysprojekteissaan. Tämä tapahtui implisiittisesti ohjaamalla tuotettujen artefaktien määrää ja tyyppiä erityisen dokumentaatiopainotteisesti ja ajoittamalla projektin vaiheet ("milestone", virstanpylväs) tuotettujen dokumenttien perusteella. Yhdysvaltain Puolustusministeriön ollessa suuri asiakas moni iso IT-yritys mukautti käytäntönsä sen vaatimusten mukaisiksi. DOD-STD-2167A korvattiin myöhemmin vähemmän rajoittavalla standardilla MIL-STD-498, jonka puolestaan siviilipuolen standardit (ISO, IEEE) myöhemmin pääosin ovat korvanneet. (Larman & Basili, 2003.)

Ohjelmistokehityksessä vesiputousmalli on kerännyt vuosien saatossa kritiikkiä joustamattomuutensa vuoksi (Aigner, 2002; Beck, 1999; Bird, 2010; D. Cohen ym., 2004; Larman & Basili, 2003; Paetsch, Eberlein & Maurer, 2003). Tiukasti noudatettuna se ei mahdollista esimerkiksi uusien ominaisuuksien lisäämistä tai aiemmin suunniteltujen muuttamista enää suunnitteluvaiheen loputtua. Tällöin voi olla mahdotonta esimerkiksi tehdä muutoksia projektin laajuuteen (engl. scope). Lisäksi vesiputousmallissa ohjelmiston testausvaihe ajoittuu kehitys-/implementointivaiheen jälkeen, jolloin havaitut isot virheet tai ongelmat voivat olla todella kalliita korjata (Aigner, 2002, s. 2). Vesiputousmallilla toteutetut projektit olivat myös tyypillisesti varsin pitkiä, ja projektit epäonnistuivatkin usein aikatauluhaasteiden vuoksi (Sohaib & Khan, 2010, s. 1).

1990-luvun puolivälissä ohjelmistokehittäjät ja tutkijat alkoivat ensimmäistä kertaa ilmaista huolensa siitä, että nopeasti kehittyvä teknologia ja teollisuus yhdistettynä yhä korkeampiin odotuksiin ohjelmistoille ja järjestelmille tekevät vanhanmallisesta ohjelmistokehitystavasta kehnon, jopa mahdottoman (Barksdale & McCrickard, 2012; Beck, 1999; Larman & Basili, 2003). Tällöin esitellyt uudet ohjelmistonkehitysmallit olivat iteratiivisia ja inkrementaalisia. Merkittävimpana esimerkkinä mainittakoon Rational Unified Process, RUP (Bygstad

ym., 2008, s. 375; Jacobson, Booch, & Rumbaugh, 1999). Iteratiivinen ja inkrementaalinen ohjelmistokehitys ei ollut silloinkaan mikään uusi keksintö, vaan se pohjautui ainakin Shewhartin 1930-luvulla Bell Labsille määrittelemiin lyhyisiin ”plan-do-study-act”-sykleihin, joita hyödynnettiin laadunparannustyössä (Shewhart, 1986). (Larman & Basili, 2003, s. 47–50.)

Useimmille ketterille menetelmille keskeinen nopea lisäarvon tuottaminen asiakkaille tuli viimeistään 1980-luvun puolivälissä Gilbin (1981, 1985) töissä esille (Larman & Basili, 2003, s. 51). 1980-luvulla myös Boehm (1988) julkaisi spiraalimallinsa, joka lienee ensimmäinen hyvin dokumentoitu, toteutusjärjestyksen riskin perusteella määrittävä evolutionäärinen ohjelmistokehitystapa (Larman & Basili, 2003, s. 51).

2.1.2 Agile-manifesti

Ketterä ohjelmistokehitys perustuu 17 ohjelmistokehityksen asiantuntijan vuonna 2001 julkaisemaan *Agile-manifestiin* (engl. Agile Manifesto) (Fowler & Highsmith, 2001). Se julkaistiin eräänlaisena vastavoimana byrokraattiselle ja jähmeälle ohjelmistokehitykselle (Beck ym., 2001; Fowler & Highsmith, 2001). Se määrittelee joukon arvoja ja toimintaperiaatteita vaihtoehdoksi perinteiselle, usein vesiputousmalliksi kutsutulle ohjelmistokehitystavalle (Barksdale & McCrickard, 2012, s. 54–55). *Ketteryys* taas voidaan määritellä esimerkiksi kyvykkyydeksi sekä luoda muutosta että reagoida muutoksiin liikevoittojen luomiseksi muuttuvassa ja epävarmassa liiketoimintaympäristössä (Highsmith, 2002). Ketterä ohjelmistokehitys vaatii jossain määrin ketteryyttä koko organisaatiolta, joten on syytä määritellä ketteryys myös laajemmassa näkökulmassa.

Ketterän manifestin teesit olivat olleet esillä jo ennen manifestin julkaisua, mutta se oli ensimmäinen ketterää kehitystä yhdistävä julkilausuma, joka pyrki pitämään esillä eri ketterien menetelmien yhteisiä piirteitä, jotka erottavat ne vesiputousmallin mukaisesta ohjelmistokehityksestä. Agile-manifesti määrittää ketterän lähestymistavan arvot seuraavalla tavalla (Beck ym., 2001):

”Löydämme parempia tapoja tehdä ohjelmistokehitystä, kun teemme sitä itse ja autamme muita siinä. Kokemuksemme perusteella arvostamme:

- **Yksilöitä ja kanssakäymistä** enemmän kuin menetelmiä ja työkaluja
- **Toimivaa ohjelmistoa** enemmän kuin kattavaa dokumentaatiota
- **Asiakasyhteistyötä** enemmän kuin sopimusneuvotteluja
- **Vastaamista muutokseen** enemmän kuin pitäytymistä suunnitelmassa

Jälkimmäisilläkin asioilla on arvoa, mutta arvostamme ensiksi mainittuja enemmän”.

Manifestin mukaisesti ketterät ohjelmistokehitysmenetelmät mahdollistavat nopean reagoinnin muutoksiin. Toisin sanottuna ketterässä kehityksessä fokus on reagoivassa toiminnassa ennustavan sijasta, ihmisissä ennemmin kuin rooleissa ja prosessi on itseohjautuva (Bird, 2010, s. 3).

Ketterä manifesti sisältää edellä kuvattujen arvojen lisäksi 12 periaatetta (Fowler & Highsmith, 2001). Ne täydentävät ja konkretisoivat ketterän kehityksen arvoja. Periaatteet on muotoiltu ohjeiden muotoon, ja ne ovat seuraavat¹:

- *Asiakastyytyväisyys*: tärkein tehtävämme on pitää asiakas tyytyväisenä toimittamalla nopeasti ja jatkuvasti arvokas sovellus.
- *Hyväksy muutos*: toivotamme tervetulleeksi muuttuvat vaatimukset, myös projektin loppupuolella.
- *Julkaise aikaisin ja usein*: toimita toimiva sovellus usein, parin viikon - parin kuukauden välein. Pyri mahdollisimman nopeaan toimitusyhteyteen.
- *Toimiva sovellus* on ensisijainen edistymisen mittari.
- *Tasainen tahti*: ketterät prosessit suosivat kestäväää kehitystä. Rahoittajien, kehittäjien ja käyttäjien on pystyttävä ylläpitämään tasaista työtahtia jatkuvasti.
- *Tiivis yhteistyö*: toteuttajien ja liiketoiminnan tuntijoiden on toimittava yhdessä päivittäin koko projektin ajan.
- *Suora keskustelu* välillisen viestinnän sijaan: Tehokkain kommunikointikeino on kasvokkain keskustelu.
- *Luottamus tekijöihin*: rakenna projektit motivoituneiden yksilöiden ympärille. Anna heille ympäristö ja heidän tarvitsemansa tuki ja luota että he tekevät työnsä.
- *Tekninen loistokkuus*: jatkuva tekniseen laatuun ja hyvään suunnitteluun panostaminen parantaa ketteryyttä.
- *Yksinkertaisuus* – tekemättä jätetyn työn määrän maksimointi – on olennaista.
- *Itseohjautuvuus*: parhaat arkkitehtuurit, vaatimukset ja suunnitelmat kehittyvät tiimeistä, jotka organisoivat itse toimintansa.
- *Itsetarkastelu*: tiimi pysähtyy miettimään säännöllisin väliajoin, kuinka tulla vielä tehokkaammaksi ja säätää toimintatapojaan sen mukaisesti.

Agile-manifestia on myöhemmin kritisoitu muun muassa epämääräisyydestä ja epätieteellisyydestä, jotka ovat johtaneet epäselvyyteen keskeisten termien merkityksestä ja tavoista, joilla ketteriä menetelmiä on mielekästä hyödyntää eri organisaatioissa (Laanti, Similä, & Abrahamsson, 2013).

2.1.3 Kuriositeeteista valtavirran käytännöiksi

Perinteisten ja ketterien menetelmien olennaisimpina eroina voidaan pitää kahta olennaista eroa tavassa suhtautua asiakkaisiin (Highsmith, 2002). Ensinnäkin perinteisesti oletetaan, että asiakkaalla ei ole käsitystä siitä, mitä hän projektilta haluaa, mutta kehittäjillä on riittävä osaaminen ja ymmärrys täydellisen määrittelyn tekemiseksi projektin aluksi. Ketterässä ohjelmistokehityksessä oletetaan,

¹ Yllä olevan listan suomennoksessa on käytetty pohjana Puumalan ja Tolvasen (2011) suomenkielistä esitystä Fowlerin ja Highsmithin (2001) 12 periaatteen listasta.

ettei kummallakaan osapuolella ole täydellistä käsitystä vaatimuksista, vaan ne työstetään yhdessä iteraatioiden aikana projektin edetessä. Toiseksi perinteisessä ohjelmistokehityksessä ajatellaan asiakkaan olevan lyhytnäköinen, jolloin kehittäjän on tarpeen tulevaisuuteen varautuakseen tehdä ylimääräistä työtä ja toteuttaa esimerkiksi ylimääräisiä ominaisuuksia. Ketterässä ohjelmistokehityksessä pyritään sen sijaan yksinkertaisuuteen ja maksimoimaan tekemättä jätettävän työn määrä (Chan & Thong, 2009, s. 804–805; Fowler & Highsmith, 2001, s. 5).

Vesiputousmallilla toteutetuissa projekteissa ei tiukasti noudatettuna ole mahdollista enää tehdä muutoksia esimerkiksi järjestelmän suunniteltuihin ominaisuuksiin tai suunnitella lisää ominaisuuksia (Aigner, 2002, s. 2), kun taas ketterissä menetelmissä suuretkin muutokset ovat iteratiivisen toimintatavan vuoksi mahdollisia (Bird, 2010, s. 1). Testausvaiheen ajoittuminen kokonaisuudessaan toteutuksen jälkeen osaltaan tyypillisesti varsin pitkät ja esimerkiksi aikatauluhaasteiden takia tyypillisesti epäonnistuvia projekteja (Sohaib & Khan, 2010, s. 1).

Dokumentoidut kokemukset ketterästä kehityksestä ovat olleet pääosin positiivisia (Layman ym., 2006, s. 654). Alan ongelmista on tehty jonkin verran tutkimusta eri näkökulmista, esimerkiksi ketterään kehitykseen siirtymisen ongelmista (mm. Nerur, Mahapatra, & Mangalaraj, 2005) ja ketterien kehitysmenetelmien hyväksynnästä organisaatiossa (Chan & Thong, 2009), mutta tutkimustulokset ovat yhä paljolti anekdoottisia ja yksittäisiin tapaustutkimuksiin perustuvia (Chow & Cao, 2008) ja kaiken kaikkiaan rajoittuneita ja niiden laatu vaihtelee (Dybå & Dingsøy, 2008, s. 852).

Syn (2007) mukaan ketterälle kehitykselle on ominaista inkrementaalinen kehitys, jossa jokainen vaihe sisältää määrittely-, suunnittelu-, toteutus- ja testausvaiheet ja johtaa vakaaseen toimivaan (engl. stable) ohjelmaversioon, joka sisältää uusina aina osajoukon lopullisen ohjelmaversioon ominaisuuksista ja mahdollisesti korjauksia aiempien versioiden ongelmiin. Kun koko projektin aikataulu ja esimerkiksi toteutettavia ominaisuuksia ei ole ennalta sovittu, toteutusta tai projektin määrittelyä on helppo muuttaa ja esimerkiksi lisäominaisuudet voidaan toteuttaa joustavasti.

Ketterä kehitys ei kuitenkaan ole mikään ”hopealuoti”-ratkaisu (Laanti ym., 2013). Joustava toteutus, jossa ei välttämättä lyödä lukkoon aikatauluja, budjettia tai toteutettavia ominaisuuksia, ei useinkaan ole asiakkaan liiketoiminnan vaatimusten vuoksi tälle mieleinen (Hoda, Noble, & Marshall, 2009). Tällöin voi olla järkevää tai jopa pakollista muodostaa liiketoimintamalli sellaiseksi, että projekteja voidaan myydä kiinteä- tai ainakin kattohintaisina. Myös erilaisia kompromisseja sopimuskäytäntöihin liittyen on esitetty (Poppendieck & Poppendieck, 2003) Asiakkaalle voidaan myydä ensin muutama sprintti, joiden päätyttyä ja tulokset nähtyään asiakas voi vielä perua koko projektin toteuttamisen. Projektin hinta ja aikataulu voidaan sopia kiinteiksi, mutta toteuttavista ominaisuuksista voidaan joustaa. Joskus voi myös olla tarkoituksenmukaista olla kertomatta asiakkaalle ensinkään, että yritys käyttää ketteriä menetelmiä ohjelmistokehityksensä (Hoda ym., 2009, s. 4–5). Eri projektien ja tilanteiden erilaiset vaatimukset

ovatkin johtaneet siihen, että organisaatiot ja yritykset ovat kehittäneet käyttöönsä erilaisia sovellutuksia, osioita ja hybridiversioita ketteristä menetelmistä (Senapathi & Srinivasan, 2012; VersionOne, 2013). Tähän toisaalta kannustettiin jo Agile-manifestissa (Fowler & Highsmith, 2001, s. 6).

2.2 Scrum esimerkkinä ketterästä kehitystavasta

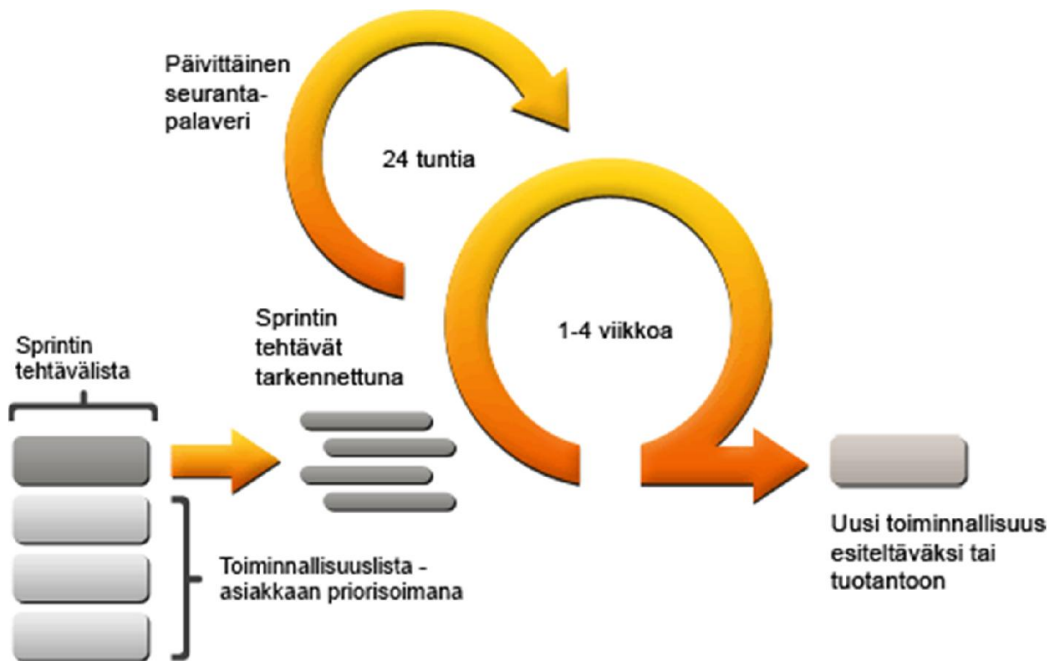
Ketterän lähestymistavan mukaisia ohjelmistokehitysmenetelmiä on useita. Strode, Huff, Hope ja Link (2012) mainitsevat ainakin 13, joista tunnetuimpina ja merkittävimpinä scrum (Schwaber & Sutherland, 2013, s. 4) XP (Beck & Andres, 2005; Beck, 1999) ja Lean-ajatteluun pohjaava Kanban (Anderson, 2010; Turner ym., 2012, s. 309–3010) ja edellisten eri variantit, kuten Scrumban (Ladas, 2009), scrumin ja XP:n hybridit (VersionOne, 2013). Scrum eri variantteineen on ketteristä ohjelmistokehitysmenetelmistä käytetyin (VersionOne, 2013). Lisäksi se on tutkielman tapaustutkimusosuuden kannalta relevantti menetelmä. Tästä syystä se on valittu tässä yksityiskohtaisempaan käsittelyyn.

Scrum on viitekehys, jonka avulla voidaan ratkaista monimutkaisiakin ongelmia kehitettäessä tuotteita luovasti ja tuottavasti mahdollisimman korkealla lisäarvolla (Schwaber & Sutherland, 2013, s. 3). Scrumin taustalla on empiirinen prosessinhallintateoria, ja sillä voidaan katsoa olevan kolme tukijalkaa: läpinäkyvyys, tarkastelu ja sopeuttaminen (Schwaber & Sutherland, 2013, s. 3). *Läpinäkyvyydelle* (engl. transparency) on tärkeää yhteinen sanasto ja yhteinen ”valmiin” määritelmä (engl. Definition of done, DoD). *Tarkastelu* (engl. inspection) hoidetaan sprinttikatselmoineissa, ja sillä ohjataan suoritusta. Tarkastelun tulosten perusteella *sopeutetaan* (engl. adaptation) suoritusta neljässä ajankohdassa, jotka scrumissa on varattu tähän tarkoitukseen, sprintin suunnittelupalaverissa (engl. Sprint Planning), päiväpalaverissa (engl. Daily Scrum), sprinttikatselmuksessa (engl. Sprint Review) ja sprintin retrospektiivissä (engl. Sprint Retrospective). (Schwaber & Sutherland, 2013, s. 3–4.)

Scrum on viitekehyksen lisäksi iteratiivinen ketterä menetelmä, joka on keskittynyt määrittelemään projektinhallinnan käytäntöjä (Bird, 2010, s. 82). Ohjelmistokehitys tapahtuu inkrementaalisesti ja iteratiivisesti, ja projektiryhmä on itseohjautuva (Dybå & Dingsøyr, 2008, s. 835; Schwaber, 2002). Käytännössä kehitystyö tapahtuu muutaman viikon, korkeintaan kuukauden mittaisissa *sprinteissä*, joiden aluksi on suunnitteluvaihe ja lopuksi katselmointi. Sprintit pysyvät koko projektin ajan yhtä pitkinä, niiden aikana ei muuteta sisältöä, kehitystiimin koostumusta ei muuteta eikä laatutavoitteita lasketa. Mikäli näyttää siltä, että sprintin tavoitteita ei pystytä toteuttamaan, projektitiimi kommunikoi tuotemistajan kanssa ja toteuttaa sprintin tavoitteet vain osittain. Projektiryhmällä on päiväpalaverit, joilla varmistetaan, että kaikki ryhmän jäsenet pysyvät perillä projektin etenemisestä. Kuvio 1 kuvaa iteratiivisen toimintatavan syklejä. (Lekman ym., 2012; Schwaber & Sutherland, 2013, Schwaber, 2002)

Scrumissa tuotteelta halutut ominaisuudet laitetaan tuotteen kehitysjonoon (engl. product backlog), jossa ne järjestetään usein arvon, riskin, prioriteetin ja

tarpeellisuuden perusteella. Kehitysjonosta ominaisuuksia poimitaan sprintti-suunnittelussa sprintissä toteutettaviksi (sprintin tehtävälisalle). Kehitysjonon voi katsoa elävän ja olevan olemassa yhtä pitkään kuin tuotteenkin: se muuttuu tuotteen ja sen ympäristön mukana kunkin hetkisiä vaatimuksia mukailen, eikä kaikkia sen kohteita välttämättä toteuteta ikinä, jos niillä ei ole tarpeeksi liiketoiminta-arvoa. (Schwaber & Sutherland, 2013, s. 11-12)



KUVIO 1 Kuvaus Scrumin iteratiivisesta työtavasta (EPiServer World, 2009)

2.3 Testaus ketterässä ohjelmistokehityksessä

Tässä alaluvussa käsitellään testausta osana ketterää ohjelmistokehitystä. Tästä käytetään usein termiä "ketterä ohjelmistotestaus" (engl. *agile testing* tai *agile software testing*) (Crispin & Gregory, 2009; Itkonen ym., 2005, s. 2).

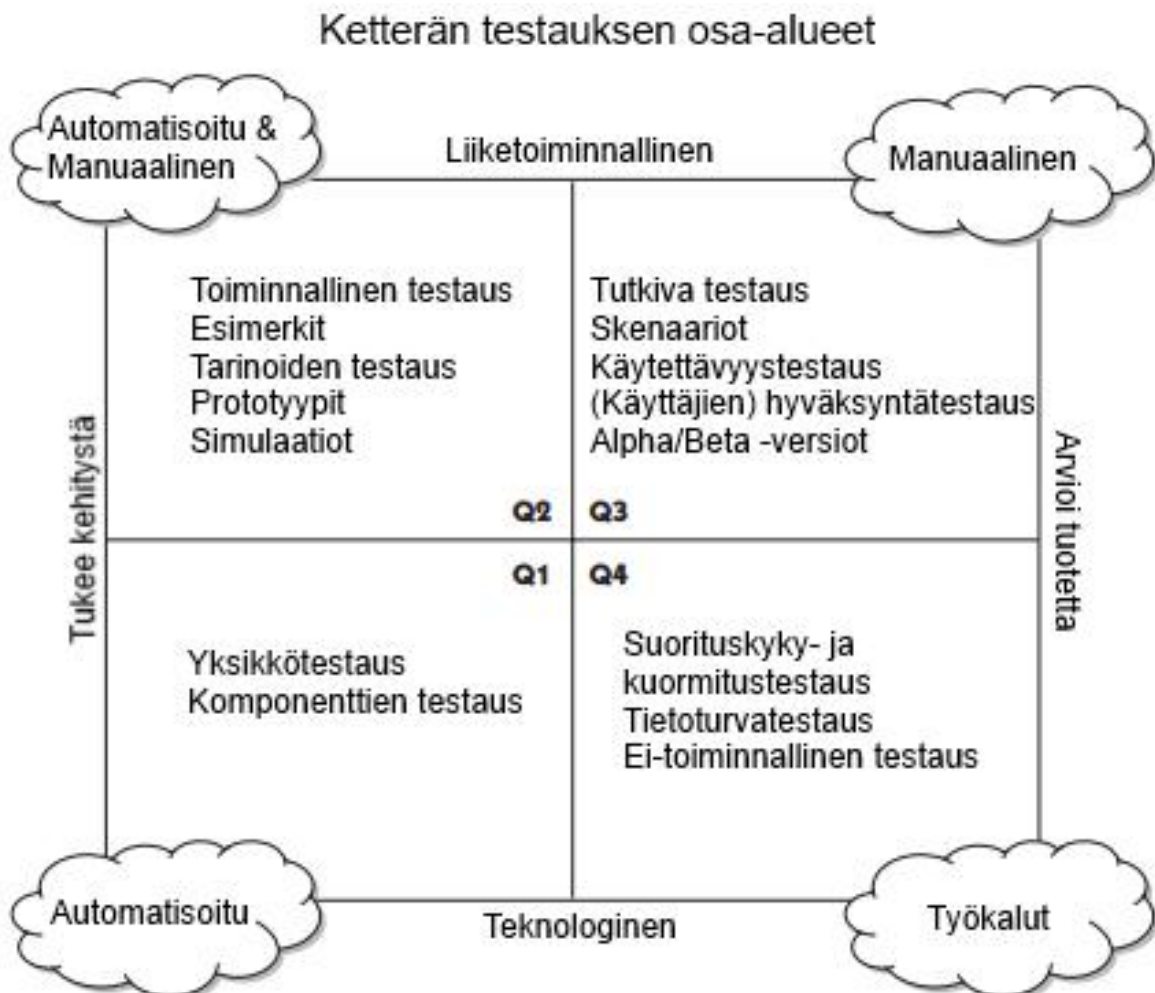
Ketterät menetelmät muuttavat huomattavasti perinteistä tapaa tehdä ohjelmistotestausta, sillä siinä kehitys ja testaus on yhdistetty täysin erottamattomasti. Tämä voi vaatia suurtakin ajatus- ja toimintatapojen muutosta - esimerkiksi voidaan mainita XP:n TDD (Test-Driven Development) (Beck, 1999). XP-menetelmän mukainen kehittäminen rakentuu testien laatimisen kautta vaatimuksia vastaavan tuotteen inkrementaaliseen rakentamiseen. (Sohaib & Khan, 2010, s. 5, 2011, s. 1)

Ketterän testauksen voi katsoa koostuvan seuraavista testauksen tyypeistä tai osa-alueista (kuvion 2 mukaisesti):

- Automatisoitu testaus
- Sekä automatisoitu että manuaalinen testaus
- Manuaalinen testaus

- Työkaluilla tehtävä testaus

Testaustyyppit eroavat toisistaan paitsi sen suhteen, validoivatko vai arvioivatko ne tuotetta, myös sen suhteen tukevatko ja testaavatko ne liiketoimintalogiikkaa vai teknologiaa (Crispin & Gregory, 2009, s. 98). Kuviossa 2 vasemman puolimmaisiet neljännekset (Q1 ja Q2) tukevat tuotteen kehitystä ja vaikuttavat pitkälti vaatimusten muodostamiseen. Q1, tuotteen kehitystä tukeva teknologinen testaus, koostuu pitkälti TDD:n mukaisista testeistä ja on automatisoitavissa, kun taas Q2, tuotteen kehitystä tukeva liiketoiminnallinen testaus, on luonteeltaan laajempi kokonaisuus ja sisältää korkeammalla tasolla tapahtuvaa testausta, joka määrittää tuotteen ulkoisen laadun. Q2-neljänneksen testaustyyppit tukevat tuotteen kehitystä erityisesti siinä mielessä, että ne havainnollistavat ja vahvistavat tuotteen haluttua toimintaa liiketoiminnallisessa mielessä. Myös aikaisen vaiheen versiot ja prototyypit muun muassa käyttöliittymistä kuuluvat tähän neljännekseen. Osa neljänneksen testaustyypeistä on automatisoitavissa, osa ei.



KUVIO 2 Ketterän testauksen osa-alueet (Crispin & Gregory, 2009, s. 98)

Oikean puolen neljännekset sisältävät tuotetta arvioivat testausmenetelmät. Tämän puolen testaustyyppit antavat palautetta tuotteen ongelmista tai kehityskohdeista. Q3 sisältää liiketoiminnallisessa mielessä järjestelmää arvioivan testauksen tyyppijä, ja yksi näistä osa-alueista on käytettävyytestaus. Neljäs neljännes sisältää testaustyyppijä, jotka arvioivat tuotetta teknologisesta näkökulmasta, kuten suorituskyky- ja kuormitustestausta sekä tietoturvatestausta. Nämä testaustyyppit suoritetaan usein työkalujen avulla, esimerkiksi kuormitustestausta JMeter-työkalun (Apache Software Foundation, 2013) avulla. (Crispin & Gregory, 2009, s. 98–103.)

Ketterän testauksen käytännöistä ketterää kehitystä parhaiten tukee kuitenkin nopea palaute (engl. feedback loop), jolloin mahdollisimman aikaisessa vaiheessa esitellään asiakkaalle tai tämän edustajalle tuotos, johon saadaan palaute, jonka perusteella toteutusta voidaan ohjata. Käytännössä tämä palautevaihe voi ajoittua jo aivan ensimmäiseen suunnitteluvaiheeseen (XP:ssä ensimmäistä iteraatiota edeltävä suunnitteluvaihe, scrumissa sprintti 0), jossa rautalankamalleja tai paperiprototyyppijä esittelemällä ja niiden avulla järjestelmän toimintaa simuloimalla voidaan saada asiakkaalta tai käyttäjältä palautetta. Palautteen perusteella seuraavan iteraation tai sprintin työjonoa voidaan muuttaa (Crispin & Gregory, 2009, s. 192).

Itkonen, Rautiainen ja Lassenius (2005) käsittelevät tutkimuksessaan yleisemmällä tasolla testauksen ja ketterän ohjelmistokehityksen yhteensovittamista. He esittelevät ketterien menetelmien tarjoamaa ohjeistusta (tai sen puutetta) testaamiseen, yhteensovittamisen ongelmia ja sivuavat myös mahdollisia keinoja erilaisten ajattelumallien ja kulttuurien yhteensovittamiseksi. Fokus on erityisesti aikamääreissä ja iteratiivisuudessa. Tutkijat listaavat merkittävimmiksi haasteiksi (1) nopean aikataulun, joka johtaa rajalliseen aikaan käytettäväksi testaukseen, (2) muuttuvat määrittelyt, joiden vuoksi käytettävissä on jatkuvasti epätäydellinen dokumentaatio, johon perustaa testitapaukset, (3) epäformaalin kommunikaation, jonka vuoksi kehittäjät ja liiketoimintaihmiset on hankala saada toimimaan yhdessä testauksen edistämiseksi, (4) toimivan sovelluksen toimittamisen merkinä edistyksestä, joka aiheuttaa jatkuvaa kuormaa laadunvarmistukselle ja jatkuvan tarpeen testaukselle jo aikaisin sovelluksen kehityksessä ja viimeisenä (5) tavoitteen yksinkertaisuuteen, jonka vuoksi erityisesti kompleksiseksi koettu testaus jää helposti hyvin vähäiselle huomiolle (Itkonen ym., 2005, s. 2). He myös erittelevät testauksen ja ketterien menetelmien periaatteiden olennaisimpia eroja, jotka hankaloittavat yhteensovittamista. Näitä on kuvattu taulukossa 1.

Taulukon 1 ongelmia paikkaamaan Itkonen ym. (2005) esittävät joidenkin testauskäytäntöjen integroimista ketterään ohjelmistokehitykseen. Iteratiiviseen ja inkrementaaliseen ketterään ohjelmistokehitykseen ei sovellu perinteinen testauksen vaiheistettu suorittaminen. Jos ohjelmistokehitys kuitenkin jaetaan päivän, iteraation (esimerkiksi sprintti) ja julkaisun mittaisiin jaksoihin, jotka sitten yhdessä muodostavat tuotteen elinkaaren hallinnan, testausaktiviteetteja voidaan kohdistaa siihen ajanjaksoon, johon ne parhaiten. Lyhin ajanjakso, yksi päivä, käsittää päivittäisen tehtävien ja ominaisuuksien toteuttamisen ja niihin

liittyen erityisesti yksikkötestit ja muu työ laadun eteen: pariohjelmointi, standardien ja parhaiden käytäntöjen noudattaminen sekä päivittäiset tai jatkuvat koostamiset (engl. build). Iteraatiotasolla kuvioihin tulee hyväksyntätestaus sekä integraatio- ja regressiotestaus ja erilaiset testausmenetelmät todellisilla käyttäjillä tai heidän edustajillaan. Julkaisun mittaisella ajanjaksolla tutkijat ehdottavat suoritettavaksi esimerkiksi tavallista laajempaa hyväksyntätestausta.

TAULUKKO 1 Testauksen ja ketterien menetelmien vastakkaiset periaatteet

Testauksen periaate	Ketterien menetelmien periaate
Testaus suoritetaan itsenäisesti.	Kehittäjät laativat itse omat testinsä. Testaaja on osa kehitystiimiä.
Testaus vaatii erityisiä testaustaitoja.	Kehittäjät testaavat kehityksen ohessa. Asiakkaan rooli on merkittävä, ja hän osallistuu aktiivisesti laadunvarmistustyöhön.
Oraakkelin käyttö tulosten vertailussa. ²	Automaattiset testit paljastavat virheet.
Tuhon aiheuttaminen tavoitteena.	Kehittäjät keskittyvät rakentamaan laadukasta ohjelmistoa ja toimivia ominaisuuksia.
Saavutetun laadun arviointi.	Luottamus laadukkuuteen seurauksena hyvien käytäntöjen seuraamisesta.

Itkosen ym. (2005, s. 6–7) mukaan monet ketterät menetelmät tarjoavat useita keinoja laadun parantamiseksi ja varmistamiseksi, mutta osa ei. Eri aikajän-teistä ”päivittäinen työ” on menetelmissä yleensä parhaiten huomioitu, mutta tutkijat suosittelevat testaajan roolin käyttöönottoa, mikäli menetelmässä ei sitä muuten olisi. Iteraation mittaiselle aikajän-teelle tutkijat suosittelevat esimerkiksi tutkivan testauksen järjestämistä.

2.4 Yhteenveto

Tässä luvussa käsiteltiin ketterän kehityksen taustaa, sen eroja sen eräänlaiseen antiteesiin, perinteiseen vesiputousmalliseen ohjelmistokehitystapaan. Ketterä ohjelmistokehitys on lisääntyvässä määrin haastanut perinteisen ohjelmistokehitystavan. Sen peruseriaatteet, eräänlaiset ydinarvot, on esitelty Agile-manifestissa: yksilöt ja kanssakäyminen, toimiva ohjelmisto, asiakasyhteistyö ja muutokseen vastaaminen. Ne pyrkivät tarjoamaan keinoja ohjelmistojen laadun parantamiseen.

Tähänastisten kokemusten perusteella ketterän ohjelmistokehitystavan mukaiset menetelmät, kuten scrum, ovatkin jo osoittautuneet hyödyllisiksi parannuksiksi aikaisempiin tapoihin tehdä ohjelmistokehitystä. Ketterä kehitys lisäksi muuttaa perinteistä tapaa tehdä testausta, ja osa perinteisen testauksen ja

² Oraakkelin käyttö, niin sanottu ”oraakkeliiongelma” (engl. oracle problem) viittaa tilanteeseen, jossa testattavan järjestelmän toiminta antaa jonkin tuloksen, ja joudutaan päättämään muiden lähteiden (”oraakkeliin”) perusteella onko tulos oikea. (Bertolino, 2001, s. 75)

ketterän kehityksen periaatteista on ristiriidassa. Yhteensovittaminen voi vaatia joustamista ketterän kehityksen periaatteista, kuten tiimin moniosaajista, ja sen sijaan voi olla mielekästä säilyttää esimerkiksi testaajan rooli.

3 KÄYTETTÄVYYSTESTAUS

*Supposing is good, but finding out is better.
(Twain, 1940, s. 324)*

Tämän luvun tarkoituksena on esitellä käytettävyydestaustuksen perusteet. Käsitely aloitetaan käytettävyyden ja siihen liittyvien käsitteiden määrittelyllä. Sitten kuvataan käytettävyydestausta osana käytettävyyssuunnittelua ja käytettävyydestaustuksen roolia ketterässä ohjelmistokehityksessä. Tämän jälkeen käsitellään käytettävyydestaustuksen eri tyyppejä ja menetelmiä, jonka jälkeen esitellään kevyt käytettävyydestaustaus. Lopuksi tarkastellaan vielä käytettävyydestaustukseen osallistuvia henkilöitä ja heidän roolejaan.

3.1 Käytettävyys

Jotta yleensä ottaen olisi mielekästä käsitellä käytettävyydestaustaa, on ensin määriteltävä, mitä käytettävyys tarkoittaa ja mikä oikeastaan on käytettävää? Erilaisia laatumääreitä voidaan toki esittää – esimerkiksi hyödyllisyys, tehokkuus, vaikuttavuus, tyydyttävyyys, opittavuus ja saavutettavuus – mutta niillä ei saavuteta kokonaisvaltaista käsitystä. Käytettävyydelle on esitetty kirjallisuudessa hyvin monenlaisia määritelmiä. Rubin (2008, s. 4) määrittelee käytettävän tuotteen seuraavasti: *”Käyttäjä voi käyttää tuotetta tai palvelua haluamallaan tavalla, jolla hän odottaa voivansa käyttää sitä, ilman esteitä, epäröintiä tai kysymysten esittämistä.”* ISO 9241-11-standardin mukaan käytettävyys on: *”mitta sille, miten hyvin määrättyt käyttäjät voivat käyttää tuotetta määrättyssä käyttötilanteessa saavuttaakseen määritetyt tavoitteet tuloksellisesti, tehokkaasti ja miellyttävästi”* (International Organization for Standardization, 1998; Jokela, 2010, s. 9 (suomenkielinen käännös ISO-standardin tekstistä); Sohaib & Khan, 2010, s. 2).

Tässä tutkielmassa käytettävyys on luontevaa käsittää Rubinin (2008) kuvaamalla tavalla. Käyttäjän odotuksiin ja hänen kohtaamiinsa esteisiin ja epävar-

muuteen keskittyvä määritelmä sopii hyvin tutkimukseen, jossa keskitytään käytettävyydestäukseen ja erityisesti kevyisiin käytettävyydestäusmenetelmiin. Esimerkiksi ISO-standardin (1998) mukainen määritelmä keskittyy tavoitteiden tarkempaan määrittelyyn ja mittareihin, kuten tuloksellisuus, tehokkuus ja miellyttävyys, joiden mittaaminen kevyitä käytettävyydestäusmenetelmiä hyödynnettäessä on epäedullista ja vaikeaa (Göransson, Gulliksen, & Boivie, 2003; Marty & Twidale, 2005).

Käytettävyyden käsitteen konkretisoimiseksi on esitetty erilaisia käytettävyydsmittareita. Taulukossa 2 esitetään kolme yleistä käytettävyydsmittareiden järjestelmää, ISO 9241-11-standardin (1998), Shneidermanin (1987) ja Nielsenin (1993) mittareita. Welie, Veer ja Eliënsin (1999) mukaan nämä kolme, verraten laajasti käytettyä luokittelua ovat pohjimmiltaan hyvin yhteneväisiä. ISO 9241-11 -standardi lähestyy mittareita varsin teoreettisesta näkökulmasta, siinä missä Shneiderman ja Nielsen tarkastelevat järjestelmien tehokkuutta käyttäjän näkökulmasta. Siten erityisesti Shneidermanin ja Nielsenin mittarit ovat pitkälti samat. ISO 9241-11-standardissa suorituskyyky vastaa paljolti Shneidermanin suorituskyykyä ja oppimiseen kuluvaan aikaan, jotka taasen vastaavat Nielsenin suorituskyykyä ja opittavuutta. ISO 9241-11-standardissa tehokkuus tarkoittaa käytön tehokkuutta, joka Shneidermanin mittareissa ilmaistaan säilyttävyytenä ajan kuluessa ja käyttäjävirheiden määränä. Nämä mittarit mittaavat käyttäjän kyykyä käyttää järjestelmää ongelmitta sekä yhden session aikana että tauon jälkeen. Ne vastaavat Nielsenin muistettavuutta sekä virheitä ja turvallisuutta, joista jälkimmäinen, kuten Shneidermanin käyttäjävirheiden määräkin, mittaa paljolti järjestelmän käytön intuitiivisuutta. ISO 9241-11-standardin määrittämä tyytyväisyys löytyy sellaisenaan myös Shneidermanilta ja Nielseniltä.

TAULUKKO 2 Käytettävyydsmittarit eri tahojen mukaan (Welie ym., 1999, s. 3)³

ISO 9241-11	Shneiderman	Nielsen
Suorituskyyky (Efficiency)	Suorituskyyky (Speed of performance)	Suorituskyyky (Efficiency)
	Oppimisaika (Time to learn)	Opittavuus (Learnability)
Tehokkuus (Effectiveness)	Säilyttävyyys ajan kuluessa (Retention over time)	Muistettavuus (Memorability)
	Käyttäjävireiden määrä (Rate of errors by users)	Virheet ja turvallisuus (Errors/Safety)
Tyytyväisyys (Satisfaction)	Subjekttiivinen tyytyväisyys (Subjective satisfaction)	Tyytyväisyys (Satisfaction)

Nielsen on myöhemmin (Nielsen, 2012) hiukan päivittänyt taulukossa 2 kuvattua mittaristoaan. Tuoreemmassa mittaristossaan aiemmin muodossa "Virheet ja turvallisuus" ollut mittari on yksinkertaisesti muodossa "Virheet" (engl. errors). Varsinaiseksi mitattavaksi määreeksi on määritetty virheiden määrä, niiden va-

³ Taulukon soluissa on suluissa ilmaistu alkuperäinen, englanninkielinen termi

kavuus ja se, miten helposti käyttäjä pystyy palaamaan järjestelmän käyttöön virheen jälkeen. Uuden mittariston viiden päämääreen lisäksi Nielsen (2012) mainitsee hyödyllisyyden eli käyttökelpoisuuden (engl. utility), sillä järjestelmän käytettävyydellä ei ole mitään arvoa, jos sillä ei ole käyttötarkoitusta tai siitä ei ole hyötyä (Bankston, 2003, s. 5; Dicks, 2002).

Modernimpi ja kokonaisvaltaisempi käsitys tuotteen käytettävyyteen liittyvistä ominaisuuksista on käyttäjäkokemus (engl. User Experience, UE/UX). ISO (2009) määrittelee standardin 9241-210 kansainvälisessä luonnoksessa (engl. Draft International Standard, DIS) käyttäjäkokemuksen *käyttäjän havainnoiksi ja reaktioiksi järjestelmän, tuotteen tai palvelun käyttöön tai odotettuun käyttöön*. Termi voidaan kuitenkin määritellä monella eri tavalla riippuen tilanteesta ja perspektiivistä: tarkastellaanko käyttäjäkokemusta ilmiönä, tutkimusalana vai käytännön työnä (Bevan ym., 2011, s. 5). Esimerkki tutkimusalaan liittyvästä määritelmästä on Usability.gov-sivuston (Assistant Secretary for Public Affairs, 2013a) määritelmä käyttäjäkokemuksesta: *laajaa termi, joka käsittää tutkimuksen, joka keskittyy tuotteen, sivuston tai järjestelmän käytön helppouden suunnittelun ja käyttäjätyytyväisyyden tason tutkimukseen*. Harper (2011) toisaalta kuvaa käyttäjäkokemuksen kattotermiksi, *joka kuvaa kaikki elementit, joista käyttäjän kokemuksen laatu koostuu hänen käyttäessään tiettyä ohjelmistoartefaktia*. Hänen mukaansa käyttäjäkokemus sisältää käyttäjäkeskeisen suunnittelun, toteutuksen ja testauksen. Käyttäjäkokemus mielletään toisaalta yleisesti holistiseksi ja subjektiiviseksi – se sisältää siis tunteet, motivaation ja toiminnan tietyssä fyysisessä ja sosiaalisessa kontekstissa ja keskittyy koettuun käytettävyyteen varsinaisten ominaisuuksien sijaan (Law, Roto, Hassenzahl, Vermeeren, & Kort, 2009; Wiklund-Engblom, Hassenzahl, Bengs, & Sperring, 2009). Käyttäjäkokemus on sisällöltään laaja termi, eikä sellaisenaan ole pääosin riittävän yksityiskohtainen laajaan käyttöön tässä tutkielmassa. Tässä tutkielmassa tarkasteltavien käytettävyydestestauksen menetelmien ei voida sanoa välttämättä tuottavan dataa järjestelmän käyttäjäkokemuksesta, sillä käyttäjäkokemus voi sisältää myös tunteet, kontekstin ja motivaation, joita on hyvin vaikea testata luotettavasti, joten käyttäjäkokemus-termin sijasta keskitytään tässä suppeamman käytettävyyss-termin tarkasteluun.

Miksi jokin järjestelmä tai tuote sitten on huonosti käytettävä? Syitä voi olla monia. Vikaa voi olla niin suunnittelussa, toteutusprosessissa, markkinoiden ja tarjoaman kohtaamisessa kuin toteuttavassa tiimissäkin. Yleisimmät syyt ovat Rubinin (2008, s. 6) mukaan:

1. Kehitystyö keskittyy laitteeseen tai järjestelmään.
2. Kohdeyleisö laajentuu tai vaihtuu.
3. Käytettävien tuotteiden suunnittelu on vaikeaa.
4. Tiimin jäsenet eivät välttämättä työskentele hyvin yhteen.
5. Järjestelmä ei aina toteudu suunnitellun kaltaisena.

Ensimmäinen ongelma on teknologian aallonharjalla ratsastaville toimittajaorganisaatioille hyvin tuttu: loppukäyttäjän tarpeiden ja toiveiden sijasta kehityksessä keskitytään teknologian tarjoamiin mahdollisuuksiin. Joskus voidaan aja-

tella, että käyttäjät ovat kyllä joustavia ja tottuvat epäintuitiiviseenkin järjestelmään, ja kehittäjät toisaalta yleensä työskentelevät mieluiten teknisten ongelmien ratkaisemisen parissa asiakkaan toiveiden tiedustelun ja toteuttamisen sijaan – ja teknisten taitojensa perusteella heidät on yleensä myös rekrytoitu. Usein kehittäjät myös toteuttavat käyttöliittymät ja muut ominaisuudet sellaisiksi kuin itse haluaisivat, sen sijaan että ajattelisivat huomattavasti ei-teknisemmän loppukäyttäjän parasta. Paljolti tähän liittyy myös toinen ongelma: kohdejoukko ei välttämättä ole tunnettu, tai sama kuin aiemmin. Teknologiset laitteet ovat yhä useamman käyttäjän saatavilla, ja harva järjestelmä on enää vain teknologian harrastajien käytössä – esimerkiksi mobiilisovellusta voi sen kehittäjän lisäksi olla käyttämässä kehittäjän isoäiti, tai jokin muu käyttäjä, jonka kiinnostukset ja taidot ovat jotain aivan muuta kuin kehittäjän. (Rubin, 2008, s. 7–8.)

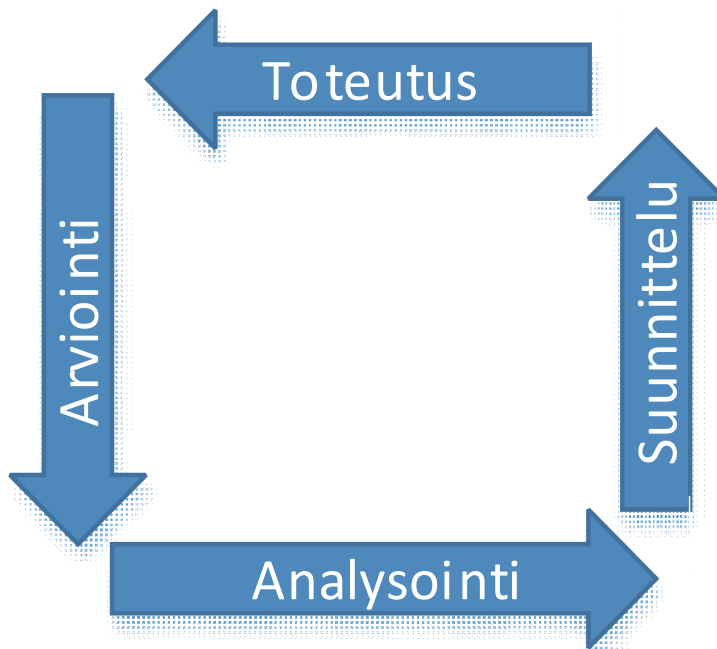
Kolmas ongelma liittyy käytettävyyden ottamiseen vakavasti. Harvoissa organisaatioissa on vielä nykyäänkään riittävää käytettävyydosaamista käytettävien järjestelmien toteuttamiseksi ja vaarana onkin, että käytettävyyteen suhtaudutaan kuin se olisi osa ”maalaisjärkeä” ja että se seuraisi automaattisesti ”järkevästä” suunnittelupäätöksistä. Käytettävän tuotteen aikaansaaminen vaatii kuitenkin järjestelmällistä työskentelyä käytettävyyden eteen ja käytettävyyden huomioimista suunnittelussa. (Rubin, 2008, s. 9.)

Neljäs ongelma syntyy tehtävänjaosta ja työntekijöiden erikoistumisesta: esimerkiksi järjestelmän käyttöliittymä, käyttöohjeet ja dokumentointi sekä ohjelmaan integroitu ohjeistus ovat todennäköisesti eri henkilöiden tai tiimien toteuttamia. Jos esimerkiksi termistöjä ei ole sovittu tai niitä ei joku tarkasta, käyttökokemus voi olla hyvin epäyhtenäinen. Käytettävyydestä tällä ongelmaa voidaan kartoittaa ja välttää testaamalla eri komponentteja yhdessä. Esimerkiksi annetaan ohjekirjan luonnos koehenkilölle käyttöön tämän suorittaessa tehtäviä järjestelmässä. Tehtävänjakoon ja erikoistumiseen liittyy myös viides ongelma: suunnitelmat ja toteutus voivat olla kaukanakin toisistaan. Suunnitteluosaaminen ja toteutukseen vaadittavat taidot ovat yleensä eri henkilöillä, ja mikäli kommunikaatio heidän välillään ei toimi, tai esimerkiksi asiakas vaatii radikaaleja muutoksia tai lisätöitä järjestelmään suunnittelun valmistuttua ja kehittäjät toteuttavat nämä itsenäisesti, voi lopputulos olla lopulta kaukanakin alkuperäisistä suunnitelmista, joissa esimerkiksi käytettävyys on voitu huomioida hyvinkin. (Rubin, 2008, s. 9–11.)

3.2 Käytettävyydestä osana suunnittelua

Käytettävyydestä tai käytettävyyttä yleensä ei voida käsitellä keskittymättä hiukan myös käytettävyyssuunnitteluun. Käytettävyyssuunnittelu (engl. usability engineering) sisältää englanninkielisenä terminä myös käytettävyydestä ja -arvioinnin, siis hyvinkin pitkälti kaiken käytettävyyden eteen tehtävän työn projekteissa. Esimerkiksi Butler (1996) kuvaa käytettävyyssuunnittelun tarjoavan systemaattiset menetelmät ja työkalut sellaisten käyttöliittymien suunnitteluun, jotka ovat ymmärrettävissä, opittavissa ja luotettavasti käytettävissä,

mutta kuitenkin painottaa käytettävyyssuunnittelun sisältävän niin analysoinnin, suunnittelun, toteuttamisen kuin arvioinninkin. Kuvio 3 havainnollistaa Butlerin (1996, s. 60) näkemystä. Kuviossa arviointi sisältää myös testauksen (Butler, 1996, s. 69).

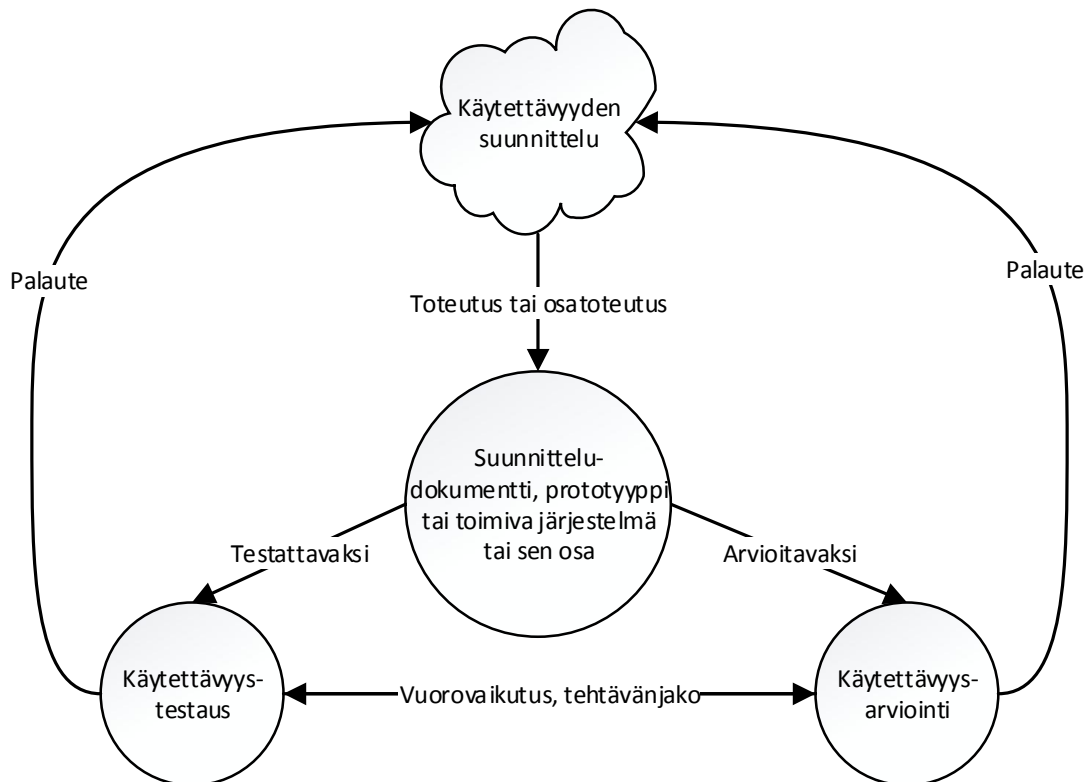


KUVIO 3 Käytettävyyssuunnitteluparadigma (mukaillen Butler, 1996, s. 60)

Dumas (1999) luonnehtii käytettävyyssuunnittelun olevan tuotteen arvioimista sen käytettävyyden parantamiseksi, mutta myös tuotteen suunnittelu- ja valmistusprosessin parantamiseksi. Leen ja McCrickardin (2007) mukaan käytettävyyssuunnittelun tavoitteena on ohjelmistotuotteiden opittavuuden, tehokkuuden, muistettavuuden, virheettömyyden ja käyttäytyvyvyyden parantaminen. Toisaalta Jacob Nielsenin popularisoima edullinen käytettävyyssuunnittelu (engl. discount usability engineering) (Ghanam & Maurer, 2007, s. 3; Nielsen, 1995a) tarkoittaa keskittymistä edullisiin käytettävyyssuunnittelun ja -testauksen metodeihin. Tällöin tavoitteena ei suinkaan ole absoluuttisesti optimaaluisimman tuloksen saavuttaminen, vaan viedä käytettävyys yksinkertaisilla työkaluilla ja toimintatavoilla riittävälle tasolle, jotta se tuottaa merkittävää lisäarvoa (Kane, 2003, s. 40). Nielsenin termi sisältää vähintään yhtä paljon suunnittelua kuin testausta, joten suomennos voi olla hieman hämäävä, ja tässä työssä käytetäänkin termiä *edullinen käytettävyystestaus* (engl. discount usability testing) viittaamaan erityisesti Nielsenin käyttämiin edullisiin käytettävyystestausmenetelmiin.

Kuvattujen termien käyttö, sekä suomeksi että englanniksi, ei ole täysin vaikiintunut. Tässä tutkielmassa käytetään termiä *käytettävyysuunnittelu* ylemmän tason terminä kuvaamaan kaikkea käytettävyyden eteen tehtävää työtä, ja ero-

tuksena tästä termiä *käytettävyyden suunnittelu* (engl. usability design) kuvaamaan ainoastaan varsinaista konkreettista suunnitteluaktiviteettia. *Käytettävyydestaus ja -arviointi* taas käsittävät ne aktiviteetit, joiden tuloksena tuotetaan jollakin menetelmällä palautetta jonkin ohjelmistoartefaktin käytettävyydestä. Käytettävyydestaus ja -arviointi toimivat siis käytännössä ikään kuin takaisinkytkentänä käytettävyyden tai käyttöliittymän suunnitteluprosessille, antaen sille palautetta esimerkiksi jokaisesta käyttöliittymäsuunnittelun iteraation tuloksena syntyneestä käyttöliittymäversiosta. Käytettävyyden suunnittelun kannalta jatkuva palaute käyttöliittymän arvioinnin tai testauksen avulla on välttämätöntä (Lee & McCrickard, 2007). Kuvio 4 havainnollistaa näiden aktiviteettien suhdetta: käytettävyyden suunnittelu (osana muuta suunnittelua) tuottaa toteutuksen tai osatoteutuksen kautta jonkin järjestelmän, sen osan tai siihen liittyvän artefaktin, johon voidaan kohdistaa käytettävyydestausta tai -arviointia, jotka taas vuorovaikutuksessa keskenään tuottavat palautetta järjestelmän käytettävyydestä, ja tämän palaute taas osaltaan ohjaa suunnittelua. Käytettävyyden eteen tehtävä työ on siis ihannetapauksessa luonteeltaan iteratiivista: lyhyet suunnittelu-, toteutus- ja arviointi- tai testausvaiheet seuraavat toisiaan.



KUVIO 4 Käytettävyyssuunnittelun, -testauksen ja -arvioinnin suhde

Käytettävyyssuunnittelun lisäksi myös käytettävyyden suunnittelu on määritelty eri lähteissä ja eri aikaan eri tavoin. Gouldin (1995) määritelmä käytettävyyden suunnittelulle sisältää myös paljon testausta ja muita tuoreemmissa määritelmässä erityisesti käytettävyyssuunnittelun piiriin kuuluvista aktiviteeteista,

mikä osaltaan on yksi osoitus termien sekavasta käytöstä myös englanniksi. Esimerkiksi Göransson, Lif ja Gulliksen (2003) yhdistävät käytettävyyden suunnittelun RUP-menetelmään siten, että valtaosa siihen kuuluvista aktiviteeteista tapahtuu projektin alkuvaiheessa muun muassa käyttöliittymäsuunnittelun ja vaatimusmäärittelyn osalta, ja toteutuksen aikana voidaan tehdä myös perinteisemmin käytettävyyssuunnittelun piiriin kuuluvia asioita kuten käytettävyyden arviointia ja käytettävyydestäusta. Heidän mallissaan käytettävyyden suunnittelu ikään kuin sovittaa käyttäjakeskeisen suunnittelun (engl. User-Centred Design, UCD) osaksi RUP-menetelmää (Göransson, Lif, ym., 2003). Göransson, Gulliksen ja Boivie (2003) tarkentavat toisaalta Gouldin (1995) määritelmää kuvaamalla käytettävyyden suunnittelua käytettävyyssuunnittelun ja interaktiosuunnittelun (engl. interaction design) yhdistelmäksi, jossa käyttäjakeskeisyys on suuressa roolissa (Göransson, Gulliksen, ym., 2003, s. 10). He myös kuvaavat käytettävyyden suunnittelua projektin eri vaiheissa: alkuvaiheessa suunnitteluun kuuluu lähinnä vaatimusmäärittely (engl. requirements analysis), varsinaisen järjestelmän kehittämisen aikana iteratiivinen suunnittelu (engl. iterative design) ja projektin lopuksi käyttöönotto (engl. deployment). Vaatimusmäärittely toisaalta jatkuu läpi projektin toteutuksen, ja käytettävyyden suunnitteluprosessi (engl. usability design process) voikin täydentää käyttötapausten tarjoamia toiminnallisia vaatimuksia käytettävyyden osalta.

Käytännössä käytettävyyssuunnittelu, mahdollisen toteutusvaiheen seurauksena, tuottaa tuloksena suunnitteludokumentteja, prototyypin tai toimivan ohjelmiston tai sen osan, jota sitten arvioidaan ja mahdollisesti testataan. Käytettävyyssarviointi ja -testaus tuottavat dataa käytettävyyden suunnittelun käyttöön. Käytettävyydestäusta ja -arviointi ovat pitkälti toisiaan täydentäviä aktiviteetteja. Kuvio 4 havainnollistaa tätä suhdetta ja erityisesti suunnittelun ja toteutuksen sekä arvioinnin ja testauksen välistä takaisinkytkentää, jossa arvioinnin ja testauksen tulokset ohjaavat suunnittelua ja toteutusta (Bankston, 2003, s. 11; Genov, 2005). Käytettävyydestäusta ja -arviointia on syytä tarkastella lähemmin, sillä ne, käytettävyyden suunnittelun sijaan, ovat tämän tutkimuksen kannalta relevantteimpia.

Käytettävyydestäusta on testausmuoto tai -prosessi, jossa järjestelmää tai tuotetta testataan sen käytettävyyden suhteen ja arvioidaan, missä määrin se täyttää sille asetetut käytettävyyksivaatimukset. Siinä missä toiset tutkijat painottavat, että termi käytettävyydestäusta ei sisällä käytettävyyssarviointia (mm. Bailey, Allan, & Raiello, 1992; Jeffries & Desurvire, 1992), toiset katsovat esimerkiksi heuristisen evaluoinnin olevan käytettävyydestäustamenetelmä siinä missä esimerkiksi korttien lajittelukin (mm. Greenberg & Buxton, 2008; Negru, 2011). Useimmissa lähteissä kuitenkin termit määritellään siten, että käytettävyydestäustuksen ja käytettävyyden arvioinnin erona on koehenkilöiden hyödyntäminen (mm. de Kock, van Biljon, & Pretorius, 2009; Jeffries & Desurvire, 1992; Nielsen, 1994a). Käytettävyydestäustuksen menetelmissä koehenkilöt ovat mukana, käytettävyyssarviointimenetelmissä ei käytetä koehenkilöitä vaan arvioinnin suorittaa käytettävyydestäustaja tai -asiantuntija (Foraker Labs, 2011; Holzinger, 2005, s. 72). Toiset tutkijat painottavat testattavien henkilöiden kohderyhmävastaavuutta (Rubin, 2008, s.

21), toisten mielestä vastaavuus ei ole välttämätöntä kunhan kokeen järjestäjä huomioi koehenkilön tason tai tyyppin arvioidessaan tuloksia (Krug, 2006).

Oman hankaluutensa termien tarkoituksenmukaiseen määrittelyyn tuo vielä termien kääntäminen. Siinä missä suomeksi ero käytettävyydestestauksen ja -arvioinnin välille on melko helppo tehdä edellä kuvatulla tavalla, englanniksi termejä *usability evaluation*, *usability inspection* ja *usability testing* käytetään paljolti tarkoittamaan samaa asiaa (esim. Nielsen, 1995a).

Tässä tutkielmassa noudatetaan rajanvetoa, jonka mukaan käytettävyydestestauksessa hyödynnetään aina koehenkilöitä, jotka halutussa määrin edustavat tuotteen loppukäyttäjiä, ja *käytettävyydsarviointi* puolestaan hyödyntää menetelmiä, joissa ei käytetä koehenkilöitä. Silloin kun koehenkilöitä käytetään, heidän tarkoituksenaan on tuoda loppukäyttäjän näkemys mukaan testaukseen. Käytettävyydestestauksella voidaan arvioida sitä, kuinka helposti käyttäjät saavuttavat ohjelmaa käyttäessään tavoitteensa (Ferreira, Noble, & Biddle, 2007, s. 56). Nielsenin määritelmä käytettävyydestestaukselle on Sohaib ja Khanin (2011) mukaan:

”[Käytettävyydestestaus] on menetelmä tuotteen arvioimiseksi testaamalla sitä käyttäjillä. Se on laadullisten testausmetodien kokoelma, joiden tavoitteena on siis hahmottaa, kuinka hyvin tuote tai järjestelmä sopii käyttötarkoitukseensa. ” (Sohaib & Khan, 2011, s. 51-52)

Käytettävyydsarviointi tarkoittaa tässä tutkielmassa käytettävyyden arvioinnin tai testaamisen metodia, johon ei osallistu koehenkilöitä – siis kyseessä on esimerkiksi testaajan itse suorittama arviointi kuten heuristinen evaluointi. Määritelmä ei ole aukoton, koska englanninkielistä termiä *usability evaluation method* käytetään joskus myös koehenkilöitä sisältävistä arvioinnin tai testauksen metodeista (esim. Frøkjær & Larusdottir, 1999). Tässä työssä käytettävää määritelmää tukee kuitenkin mm. Grayn ja Salzmanin (1998) tutkimus, jossa käsite määritellään seuraavasti:

”Käytettävyydsarviointimetoiteita käytetään ihmisen ja järjestelmän vuorovaikutuksen sellaisten piirteiden arviointiin, joita parantamalla voidaan lisätä käytettävyyttä. Niitä hyödynnetään useimmiten tarvearvioinnin jälkeen mutta ennen beta-testausta” (Gray & Salzman, 1998, s. 206)

Käytettävyydestestauksella ja käytettävyyden arvioinnilla on koehenkilöiden hyödyntämisen lisäksi joitakin muitakin eroja, tärkeimpänä se, että ne löytävät josain määrin erilaisia havaintoja. Oikeat käyttäjät voivat löytää vain puolet siitä havaintojen määrästä mitä ammattilaiset tekevät (Sauro, 2010), mutta tehdyt havainnot ovat erilaisia (Bruun, Gull, Hofmeister, & Stage, 2009, s. 1626; Frøkjær & Larusdottir, 1999; Sauro, 2010) ja arviointimenetelmät tuottavat herkemmin virheellisiä havaintoja (Frøkjær & Larusdottir, 1999).

3.3 Käytettävyydestestauksen tyyppejä ja menetelmiä

Tässä alaluvussa esitellään käytettävyydestestauksen tyyppejä ja yleisimpiä menetelmiä. Tässä vaiheessa ei vielä eritellä kevyiden ja ”perinteisten” menetelmien välillä, joskin osa esitellyistä menetelmistä on myös kevyen käytettävyydestestaukseen tarkoitettuja. Esityksessä käydään läpi erityisesti Rubinin (2008, s. 17–38) erittäin kattava luokittelu käytettävyydestestausmenetelmistä ja niiden ryhmitte-lystä käytettävyydestestaus-tyyppihin ja verrataan sitä muun muassa Nielsenin (1994b, 1995a) huomattavan yksinkertaisiin kevyen käytettävyydestestauksen me-
netelmiin. Käytettävyydestestaus-tyypit ovat siis ikään kuin käytettävyydestestaus-
menetelmien luokittelutapa, joskin osa menetelmistä voi olla luokiteltavissa use-
ampaan tyyppiin. Käytettävyydestestaus-tyypit kertovat paitsi käytettävyydestestaus-
menetelmän tyyppistä, myös sen sijainnista tai suhtautumisesta tuotteen kehitys-
prosessiin tai sen vaiheeseen.

3.3.1 Käytettävyydestestauksen tyyppejä

Tässä esitellään käytettävyydestestauksen tyyppejä Rubinin (2008) mukaan. Näihin tyyppihin viitataan myös Nielsenin kevyitä käytettävyydestestausmenetelmiä esi-
teltäessä. Tämä käytettävyydestestausmenetelmien lajittelutapa on käytännöllinen
määriteltäessä sitä, mihin rajalliset käytettävyydestestausresurssit kannattaa missä-
kin projektin vaiheessa käyttää. Tyypit esitellään suunnilleen siinä järjestyksessä,
jossa ne (tai niitä edustavat käytettävyydestestauksen menetelmät) ovat relevant-
teja järjestelmän kehitysprosessin edetessä. Näitä tyyppejä voisi nimittää myös
käytettävyydestestauksen vaiheiksi. Niiden eroina ovat käytettävät käytettävyy-
testausmenetelmät ja erityisesti kvantitatiivisen ja kvalitatiivisen testauksen
suhde sekä koehenkilöiden ja kokeen valvojan välisen interaktion määrä (Rubin,
2008, s. 27–28).

Tutkiva tai muodostava testaus (engl. explorative or formative study) on usein
tuotteen kehityksen alku- tai keskivaiheeseen sijoittuva käytettävyydestestauksen
tyyppi (Rubin, 2008, s. 29–33). Käyttäjätyyppit ovat jo selvillä, ja käyttötapauksista
on ainakin jonkinlainen käsitys. Testauksen kohteena voi olla esimerkiksi kor-
kean tason navigaatio ja sivuston alustava asettelu. Tämä on usein ensimmäinen
vaihe, jossa voidaan havaita fundamentaalisia suunnitteluongelmia. Voitaisiin
ehkä sanoa, että testaustyyppi testaa tuotteen runkoa tai luurankoa (Rubin, 2008,
s. 34). Tutkivassa testauksessa järjestelmää tai tuotetta käydään läpi systemaatti-
sesti, mutta ikään kuin seurataan virheiden tai ongelmien ”hajua” ja pyritään löy-
tämään esimerkiksi käyttötapausten pohjalta ongelmia, joita automatisoidulla tai
täysin orjallisesti testitapauksia seuraavalla testauksella ei pystyttäisi löytämään
(Crispin & Gregory, 2009, s. 201). Muodostavan testauksen tarkoituksena on
käyttöliittymän tai järjestelmän parantaminen sen kautta, että etsitään sen hyvät
ja huonot piirteet ja esitetään parannusehdotuksia. Tyypillisenä testaustapana
voidaan pitää ääneen ajattelua (Nielsen, 1993, s. 170).

Arvioiva tai summaava testaaminen (engl. assessment or summative test) suoritetaan yleensä tuotteen ollessa vähintäänkin perusosiltaan pitkälle tai jopa valmiiksi suunniteltu. Jos edelliset testauksen tyypit testasivat tuotteen runkoa tai luurankoa, tämä testaustyyppi testaa ”lihaa luiden ympärillä”. Olettaen, että tuotteen konsepti on kunnossa, tämän vaiheen tai tyyppin menetelmät testaavat konseptin toteutuksen onnistumista. Testauksen kohteena on realististen tehtävien toteuttaminen mahdollisimman toimivalla prototyypillä. (Rubin, 2008, s. 34–35.)

Validoiva tai vahvistava testaus (engl. validation or verification test) on myöhäisemmän vaiheen testaustyyppi, jolla testataan laadullisten vaatimusten täyttymistä ja esimerkiksi aiempien ongelmien korjausten onnistumista. Lisäksi voidaan testata esimerkiksi järjestelmän lopullisen laitteiston ja ohjelmiston yhteen toimivuutta tai pitkien työnkulkujen tai käyttötapausten toteutumista suunnitellusti. (Rubin, 2008, s. 35–37.)

Vertaileva testaus (engl. comparison test) tarkoittaa radikaalisti erilaisten toteutustapojen paremmuuden selvittämiseksi järjestettyä testiä. Tätä tyyppiä voidaan käyttää projektin missä tahansa vaiheessa: alkuvaiheessa voidaan testata vaikkapa erilaisten navigaatioelementtien intuitiivisuutta, ja loppuvaiheessa laitaa tuote kilpailijan vastaavaa tuotetta vastaan testiin. (Rubin, 2008, s. 37–38.)

3.3.2 Käytettävyyden arvioimisen ja testaamisen menetelmiä

Käytettävyydestestauksen menetelmiä yhdistää yleensä se, että niissä tarkastellaan ja arvioidaan käyttäjän tai edustavan koehenkilön selviytymistä erilaisista tehtävistä tuotteen parissa (Rubin, 2008, s. 19–20). Menetelmiä esitellään useissakin lähteissä, mutta tässä erittelyssä pohjana oleva Rubinin (2008) listaus on kattavimmasta päästä. Menetelmät on esitetty siinä järjestyksessä, jossa niitä todennäköisesti käytetään tosiasiallisissa projekteissa.

Etnografinen tutkimus tapahtuu ensisijaisesti käyttäjän ”luonnollisessa ympäristössä” ja tuotteen luonnollisessa käyttötilanteessa. Sen avulla kerätään tietoa siitä, keitä käyttäjät itse asiassa ovat, mitä käyttötapauksia ja tavoitteita heillä on suhteessa tuotteeseen ja mikä on konteksti, jossa he työskentelevät tuotteen kanssa. Etnografinen tutkimus voi olla apuna käyttäjäprofiilien, persoonien⁴, skenaarioiden ja tehtäväkuvausten (engl. task descriptions) luomisessa. (Blomquist & Arvola, 2002; Rubin, 2008, s. 16.)

Osallistava suunnittelu (engl. participatory design) ottaa yhden tai useamman edustavan käyttäjän mukaan kehitystiimiin, jolloin käyttäjien tietämys ja näkemys saadaan projektiin mukaan jo suunnitteluvaiheessa. Riskinä on, että käyttäjä on tiimissä sisällä aivan samoin kuin kehittäjätkin, jolloin tuotteelle sokaistuu. Lisäksi tällaisissa projekteissa tehdään usein sisäistä kehitystä, jolloin

⁴ **Persoon**a (engl. persona) on tarkka kuvaus hypoteettisesta käyttäjästä ja hänen tavoitteistaan (Blomquist & Arvola, 2002, s. 197). Persoonalla voi olla nimi ja kasvot, ja sen tavoitteena on edustaa käyttäjää ilman käyttäjän projektiin mukaan ottamisen vaivaa ja riskejä.

käyttäjien edustaja voi varoa palautteen antamista työkavereilleen. (Rubin, 2008, s. 17.)

Ryhmähaastattelututkimus (engl. focus group research) on usein ryhmässä oissa jonkinlaisen tuotteen, tuotesuunnitelman tai sen esiasteen arvioimista osallistujien näkökulmista (Rubin, 2008, s. 17). Työskentely voi olla esimerkiksi työpajamaista, ja menetelmää todennäköisimmin hyödynnetään hyvin aikaisessa vaiheessa projektia (Krug, 2006).

Kysely (engl. survey) on käytännöllinen metodi ”yleisen käyttäjän” löytämiseksi hyvin suurten populaatioiden joukosta ja hänen mielipiteidensä ja toiveidensa hahmottamiseksi. Kyselyitä voidaan käyttää missä vaiheessa tahansa kehitystä: alkuvaiheessa potentiaalisten käyttäjien toiveiden ja vaatimusten kartoittamiseksi, ja myöhemmin heidän mielipiteidensä ja arvioidensa kysymiseksi ensin suunnitelmista, sitten prototyypeistä ja lopulta tuotteesta. (Rubin, 2008, s. 17–18.)

Läpikäynnit (engl. walk-throughs) ovat kätevä tapa testata tiettyjen käyttötapausten sujuvuutta ja niiden toteuttamiseen liittyviä ongelmia. Tällöin käyttötapausten ja erityisesti käyttäjätyyppien on jo pääpiirteittäin oltava selvillä. Läpikäynnin suorittaa todellinen käyttäjä tai häntä toivottavasti edustava henkilö. (Rubin, 2008, s. 18.)

Avoin ja suljettu korttien lajittelu (engl. card sorting) ovat menetelmiä, jotka käyvät rakenteen, kategorioiden, hierarkian ja termien testaamiseen, ja ne mittaavat pääosin näiden johdonmukaisuutta. Testissä koehenkilölle annetaan esimerkiksi järjestelmän eri näkymiä ilman otsikkoja tai kategorioita ja häntä pyydetään nimeämään näkymät tai kategorisoimaan ne (avoin korttien lajittelu) tai yhdistämään valmiiksi annetut kategoriat, otsikot tai nimet näkymiin (suljettu korttien lajittelu). (Nielsen, 1995a; Rubin, 2008, s. 18)

Paperiprototyypityksessä (engl. paper prototyping) jokin järjestelmän näkymä tai ominaisuus esitellään paperilla käyttäjälle tai koehenkilölle ja voidaan joko vain esittää kysymyksiä prototyypin perusteella tai käyttää useita paperilla olevia näkymiä simuloimaan jopa kokonaista läpikäyntiä. Tämä on kevyt ja halpa tapa testata näkymiä ja skenaarioita jo ennen kuin järjestelmän kehitys on aloitettu, tai ennen kuin mitään testattavaa on saatu aikaiseksi. (Rubin, 2008, s. 18–19.)

Heuristinen evaluointi tai ammattilaisen arvio (engl. heuristic or expert evaluation) on yleensä projektin ulkopuolisen, käytettävyyssiantuntijan suorittama arvio tuotteen käytettävyydestä sen käyttäjäkunta huomioiden. Parhaassa tapauksessa kyseisellä asiantuntijalla on myös kohdealuetietämystä, jolloin hän voi entistä paremmin arvioida tuotteen tai järjestelmän toimivuutta (Rubin, 2008, s. 19). Heuristisessa evaluoinnissa tukeudutaan usein johonkin listaan käytettävyyden mittareista johdetuista heuristiikoista, joista tunnetuin (Kılıç, Delice & Güngör, 2009, s. 935) lienee Nielsenin 10-kohtainen lista (Nielsen, 1995b; suomennoksissa käytetty myös Nummiahho, 2003):

1. Järjestelmän tilan näkyvyys (engl. Visibility of system status)
2. Käytä käyttäjien omaa kieltä (engl. Match between system and the real world)

3. Käyttäjän kontrolli ja vapaus (engl. User control and freedom)
4. Yhtenäisyys ja standardien noudattaminen (Consistency and standards)
5. Virhetilanteiden ehkäisy (engl. Error prevention)
6. Muistikuormituksen vähentäminen (engl. Recognition rather than recall)
7. Käytön joustavuus ja tehokkuus (engl. Flexibility and efficiency of use)
8. Suunnittelun estetiikka ja minimalistisuus (engl. Aesthetic and minimalist design)
9. Käyttäjien tukeminen virheiden tunnistamisessa ja niistä palautumisessa (engl. Help users recognize, diagnose, and recover from errors)
10. Käyttäjän apu ja tuki, sekä dokumentaatio (engl. Help and documentation)

Ensimmäisen heuristiikan mukaan järjestelmän täytyy ilmaista tilansa selkeästi. Käyttäjän täytyy tietää, mitä tapahtuu ja mitä hänen pitää tehdä. *Toinen heuristiikka* määrittää, että järjestelmän sanaston täytyy vastata tosielämässä vastavassa tilanteessa käytettävää sanastoa, eikä niinkään ottaa kantaan järjestelmän teknisiin puoliin. *Kolmannen heuristiikan* mukaan käyttäjille tulisi tarjota mahdollisuus liikkua eri tiloihin ja niistä pois luontevasti, sillä järjestelmien käyttö perustuu usein paljolti yritykseen ja erehtymiseen. Tästä syystä myös peruuta- ja tee uudelleen -tyyppiset napit ovat hyödyllisiä. *Neljäs heuristiikka* edellyttää, että järjestelmä on yhtenäinen kokonaisuus ja noudattaa standardeja, jotta käyttäjien ei tarvitse pohtia, tarkoittavatko samat sanat ja symbolit samaa, käyttäytyykö järjestelmä samankaltaisissa tilanteissa samalla tavalla, ja toisaalta sitä, noudattaako järjestelmä standardeja yleensä. *Viidennen heuristiikan* mukaan hyvääkin virheilmoitusta parempi on aina virhetilanteen ehkäisy ennalta. *Kuudennen heuristiikan* voisi ilmasta myös muodossa ”tunnistaminen muistamisen sijasta”, jolloin käyttäjän pitäisi pystyä pääättelemään eri nappien ja näkymien merkityksiä muistamatta niitä. *Seitsemäs heuristiikka* edellyttää, että kokeneelle käyttäjälle tarjotaan esimerkiksi näppäimistöopolkuja ja mahdollisuus käyttöliittymän mukauttamiseen omaan käyttöönsä sopivaksi. *Kahdeksas heuristiikka* painottaa yksinkertaisuutta: vain relevantit asiat ovat näkyvissä, koska jokainen uusi elementti kilpailee edellisten kanssa huomiosta. *Yhdeksäs heuristiikka* koskee tilannetta, jossa virhe on jo tapahtunut ja siitä pitäisi palautua: virheilmoitusten tulisi olla selkokielisiä (epämääräisten virhekoodien sijaan) ja virheen vaatimat toimenpiteet pitäisi tehdä selväksi. *Kymmenes* ja tämän listan viimeinen *heuristiikka* käsittelee käyttäjälle kohdistettuja ohjeita: vaikka järjestelmän normaalisti tulisi olla käytettävissä ilman erillisiä ohjeita, niin jos ja kun ohjeita kuitenkin tarvitaan, niiden tulisi olla helposti saavutettavissa ja konkreettisia. (Nielsen, 1995b)

Nielsenin lista on koostettu alun perin yhdessä Molichin kanssa (Molich & Nielsen, 1990; Nielsen & Molich, 1990), mutta Nielsen on sittemmin täydentänyt ja parannellut sitä 249 käytettävyysohjeen analysoinnin perusteella (Nielsen, 1995b). Nielsen pitää omaa listaansa käyttökelpoisena, ja sitä laajempia sinänsä

perusteltuina mutta mahdollisesti hankalasti hyödynnettävinä (mm. Tognazzini, 2003).

Varsinaisessa *käytettävyytestauksessa* (engl. usability testing) tarkastellaan käyttäjien tai heitä edustavien koehenkilöiden toimintaa järjestelmän tai sen prototyypin parissa tosielämän tehtäviä suorittamassa. Eri lähteissä testausta jaotellaan hiukan eri tavoin: ainakin hoksaus- ja käyttötapaustestaukseen (Krug, 2006) tai formaalien testien läpikäyntiin ja iteratiiviseen testaukseen, jolla pyritään paljastamaan käytettävyysspuutteita tuotteessa (Rubin, 2008, s. 19–20). Yksi näistä menetelmistä on myös ääneen ajattelu tai yksinkertaistettu ääneen ajattelu (Krug, 2000, 2006).

3.4 Kevyt käytettävyytestaus

“The biggest myths are that usability testing is hard, users are difficult to reach, you need many participants for a formative – or diagnostic – test, and you need a formal report.” (Quesenbery, 2009)

Tässä alaluvussa tuodaan esille nykyajan vaatimuksia testaustapojen keventämiselle, niiden perusteluja ja mahdollisia toteutustapoja. Lisäksi esitellään käytettävyytestausta osana edullista käytettävyyssuunnittelua (engl. discount usability engineering) (Krug, 2000; Nielsen, 1995a). Aiemmin esiteltyistä käytettävyytestauksen menetelmistä pyritään valitsemaan sellaisia, joista saadaan maksimaalinen hyöty ja joita voidaan soveltaa mahdollisimman kevyesti. Erityisesti keskitytään Nielsenin (1995a) tutkimuksiin aiheesta.

Tässä tutkielmassa puhutaan *kevyestä käytettävyytestauksesta*, kun käsitellään erityisesti ketterään ohjelmistokehitykseen soveltuvia käytettävyytestausmenetelmiä. Näitä menetelmiä on esitelty useasti edullista käytettävyytestausta käsittelevissä artikkeleissa (Krug, 2006; Nielsen, 1995a). Tutkielmassa on periaatteellisista syistä vältetty käyttämästä termiä ketterä käytettävyytestaus, sillä varsinaiset käytettävyytestauksen metodit eivät ketterään kehitykseen sovitettuina muutu: niitä vain pyritään soveltamaan soveltuvien osien ja varsinkin kevyillä tavoilla. (Budwig ym., 2009, s. 3077; Kane, 2003, s. 44; Lee & McCrickard, 2007, s. 2.)

3.4.1 Käytettävyytestauksen kehitys ja uusien ajatusten nousu

Käytettävyytestaus oli pitkään vain hienoissa ja kalliissa käytettävyysslaboratorioissa suoritettavaa ”salatiedettä”. Jo 1990-luvulla kuitenkin tieteellisissä piireissä alkoi nousta kritiikkiä tätä näkemystä vastaan (esim. Cooper, 1995; Nielsen, 1995a), ja ajatuksia on kehitelty ainakin akateemisessa maailmassa hiljalleen eteenpäin. Syystä tai toisesta alan käytännöt ovat kuitenkin kehittyneet varsinkin

hitaasti, eivätkä edes alan uranuurtaja Jacob Nielsenin ponnistelut käytettävyydestäuksen virtaviivaistamiseksi ja arkipäiväistämiseksi ole juuri saavuttaneet suuria massoja (Krug, 2006).

Miljoonia maksavat käytettävyydelaboratoriot ja kalliit käytettävyydestäuskonsultit, ja niiden seurauksena kallis käytettävyydestäminen ovat arkea vielä nykypäivänäkkin. Laboratoriosta ei toki ole haittaa – ellei se ohjaa koko käytettävyydelprosessia ja siirrä huomiota pois tärkeistä asioista – mutta välttämätön se ei ole. Moni testi on luonnollisuuden vuoksi hyödyllistä tehdä muualla kuin laboratoriossa, joskaan ei voida yksiselitteisesti sanoa esimerkiksi kenttätutkimusta laboriotutkimusta luotettavammaksi tai paremmaksi (Kallio, 2005). Testäympäristö on siis syytä valita testin mukaan. Käytettävyydestäus taas on todennäköisesti mielekkäintä ottaa osaksi projektiryhmän työtapoja ulkopuolisten konsulttien palkkaamisen sijasta (Spencer, 2008). (Rubin, 2008, s. 93–95).

3.4.2 Käytettävyydestäuksen murros ja kevyempien metodien nousu

Ohjelmistokehitys on joitakin poikkeuksia lukuun ottamatta muuttunut entistä nopeatemposemmaksi ja kehityssyklit ovat lyhentyneet. Tämä mahdollistaa nopean palautteen keräämisen ja virheiden korjaamisen jopa vain tuntien varoitusaajalla, erityisesti nettipohjaisissa järjestelmissä (Scott, 2009, s. 8). Toisaalta muutosten nopea vienti tuotantoon saattaa kuitenkin tarkoittaa testäukseen suunnatun huomion vähentymistä, ja nopeat päivitykset (engl. upgrade) aiheuttavat korkeamman tason suunnittelun unohtumisen (Scott, 2009, s. 9). Kaiken kaikkiaan mahdollisuus nopeaan käyttöönottoon ja palautteen saantiin ja yleensäkin erittäin nopeaan kehityssykliin siirtää pois huomiota suunnittelusta ja testäuksesta: ne on ikään kuin ulkoistettu asiakkaalle tai käyttäjälle. Useasti, hiukan negatiivissävytteisesti, puhutaankin varsinkin suurten yritysten (kuten Microsoft) ulkoistaneen beta-testäuksen kuluttaja-asiakkailleen (esim. Kniskern, 2013; Weiss, 2000).

Käytettävyyden ja sen testäamisen perusasiat eivät ole muuttuneet mihinkään: aktiivinen käyttäjän mukaan ottaminen, iteratiivinen ja monialainen suunnittelutyö ja käyttäjien ja teknologian onnistunut ”naittaminen” ovat olennaisimmat asiat yhä vieläkin (Scott, 2009, s. 9). Scott (2009, s. 11) listaa tärkeimmiksi haasteiksi ja mahdollisuuksiksi:

1. Käytettävyyden sovittamisen ketteriin menetelmiin
2. Ubikviteetin (”läsnäolo kaikkialla”, ”jokapaikan tietotekniikan”) hyväksikäyttö esimerkiksi tiedonlouhinnan kautta
3. Kevyt käytettävyydestäus siirrettävällä testäusasemalla (engl. ”Lab in a bag”)
4. Menetelmien muuntaminen kontekstuaalisiksi (testäamisen vieminen kentälle)
5. Käyttäjien hyödyntäminen suunnittelijoina

Scottin (2009) mukaan käytettävyys sinällään on universaalia, mutta siihen liittyvät menetelmät ja käsitteet ovat kehittyneet ja käytettävyystestauksenkin tulisi kehittyä. Hän mainitsee yhtenä tärkeimpänä edistysaskeleena ”lab in a bag” -tyyppisen kevyen ja mobiilin käytettävyystestauksen, jossa prototyyppi, tietokone, kamera ja muut tarvittavat laitteet kulkevat testaajan mukana koehenkilöiden luo. Myös Nielsenin (1994b) jo aiemmin esittelemät ”sissimenetelmät” (engl. guerrilla methods) pyrkivät huomioimaan kontekstin testeissä paremmin ja toteuttamaan testit kevyemmin (Scott, 2009, s. 11). Scottin erittelemistä haasteista ja mahdollisuuksista oikeastaan IT:n läsnäoloa kaikkialla lukuun ottamatta kaikki liittyvät tavalla tai toisella ketterään ja moderniin ohjelmistokehitykseen: ketteriin menetelmiin sovittamiseen, kevyeen käytettävyystestaukseen, asiakas-yhteistyöhön kontekstualisoinnin kautta ja käyttäjien hyödyntämiseen.

Toisaalta jotain alan kestävästä kysymyksistä kertoo se, että jo Nielsen 17 vuotta Scottia aiemmin käytettävyysuunnittelumallissaan (engl. usability engineering model, Nielsen, 1992a) listaa tärkeimmiksi elementeiksi aikaisen käyttäjien mukaan ottamisen, käyttäjien hyödyntämisen suunnittelijoina, käyttöliittymän eri osien (yhteneväisyyden) koordinoinnin ja käyttöliittymien iteratiivisen kehittämisen testaustulosten pohjalta (Sohaib & Khan, 2010, s. 1) - siis pitkälti samoja kysymyksiä, joita Scott esitti myöhemmin ja jotka ovat yhä avoimia käytettävyystestausta ketterään kehitykseen sovitettaessa!

Käytännön käytettävyystestauksessa on hiljalleen siirrytty kattavasta ja formaalista käytettävyystestauksesta ketterämpään ja vähemmän kattavaan ”good enough” -tulokseen pyrkivään käytettävyystestaukseen – tai jopa ”better than nothing” (Marty & Twidale, 2005, s. 4-5), eli pyritään tekemään pienin mahdollinen määrä työtä, jolla saadaan kuitenkin jotain tuloksia.

3.4.3 Kevyen käytettävyystestauksen menetelmiä

Nielsenin (1995a) artikkeli esittelee kevyitä, tarkasteluun ja ”riittävän hyvään” tulokseen pyrkiviä kevyen käytettävyystestauksen menetelmiä. Seuraavassa on kuvattu yhden todellisen käytettävyystestausteprojektin menetelmät siinä järjestyksessä, jossa niitä hyödynnettiin (Nielsen, 1995a, s. 98):

1. Suljettu korttien lajittelu kategorioiden löytämiseksi
2. Kuvakkeiden intuitiivisuuden testaus
3. Korttien ja kuvakkeiden yhdistäminen
4. Kotisivun läpikäynti (engl. walkthrough)

Joka testin jälkeen suunnitelmia parannettiin seuraavaa testiä varten. Suunnittelu oli siis iteratiivinen prosessi, jossa jokainen testikerta tuotti palautetta seuraavaa suunnitelmaa varten (Nielsen, 1995a, s. 100). Toisaalta, yleisellä tasolla Kane (2003) listaa Nielsenin (1994b, s. 4, 1995a, s. 98) useissa tutkimuksissaan esittelemät halvan käytettävyystestauksen menetelmät seuraavasti:

1. Skenaariot

2. Yksinkertaistettu ääneen ajattelu (engl. simplified thinking aloud)
3. Heuristinen evaluointi
4. Korttien lajittelu eri muodoissaan.

Halvat ja kevyet käytettävyydestaustamenetelmät luottavat tilastotieteellisillä menetelmillä analysoitavissa olevan aineiston ja suuren mittakaavan luotettavien testien sijaan enemmän testajan ammattitaitoon ja kykyyn tulkita tuloksia. Lisäksi kevyen käytettävyydestaustuksen johtavana ajatuksena on, että vähänkin käytettävyydestausta on parempi kuin ei lainkaan käytettävyydestausta: jos investoinnilla edes yhteen halpaan käytettävyydestaustsessioon löydetään muutama kriittinen käytettävyysongelma, jotka muutoin olisivat paljastuneet vasta tuotannossa, on investointi todennäköisesti ollut kannattava (Krug, 2006; Nielsen, 1995a, s. 100).

Sopivan käytettävyydestaustamenetelmän valinta pitkälti ratkaisee, saadanko testauksesta hyödyllisiä tuloksia, ja oikean menetelmän valintaan onkin syytä kiinnittää huomiota. Sama käytettävyydestaustamenetelmä ei välttämättä sovi eri projekteihin, sillä testauksen onnistumiseen vaikuttavat monet seikat: muun muassa projektin tyyppi, organisaatiokulttuuri ja aikataulu. Menetelmä, jolla on tietyssä projektissa ja tietyssä organisaatiossa havaittu 100 % ongelmista, voi tuottaa kelvottomia tuloksia toisessa organisaatiossa tai toisenlaisessa projektissa (Wixon, 2003, s. 31).

Jeffries, Miller, Wharton ja Uyeda (1991) vertailevat tutkimuksessaan heuristista evaluointia, käytettävyydestausta, käytettävyysohjeiden (engl. guidelines) arviointia ja kognitiivista läpikäyntiä. Tässä tutkimuksessa tarkasteltava heuristinen evaluointi on käytännössä yhdistelmä heidän määritelmiensä mukaisesta heuristisesta evaluoinnista ja käytettävyysohjeiden arvioinnista ja käytettävyydestausta täsmää hyvin pitkälti yksinkertaiseen ääneen ajatteluun. Heidän mukaansa heuristinen evaluointi tuotti kokonaisuutena eniten tuloksia, mutta suuri osa havainnoista oli prioriteetiltaan vähäisiä ja testaus vaati, että käytettävissä oli useita käytettävyyssiantuntijoita. Käytettävyydestausta tuotti suhteessa eniten prioriteetiltaan suuria havaintoja, mutta oli menetelmistä kallein. Käytettävyysohjeiden arviointi ja kognitiivinen läpikäynti on mahdollista suorittaa myös ilman käytettävyyssosaamista. Ne kuitenkin jättivät huomiotta tärkeitäkin havainnoita.

3.4.4 Käytettävyydestaustuksen tulosten huomiointi

Käytettävyydestestien tulokset voivat vaikuttaa ainakin uusia käyttäjätarinoita luovina tai käyttäjien koulutusta ohjaavina (Meszaros & Aston, 2006). Käytettävyydestausta johtaa jo pelkästään luonteensakin vuoksi (testaa ihmisen hankalasti ennustettavaa käyttäytymistä) yleensä muutoksiin ohjelmistossa tai vaatimuksissa (Ferreira ym., 2007, s. 57), sillä paitsi toteutuksia, myös määrittelyitä on hankala tehdä ihmisten toimintaa riittävän hyvin ennustaen.

Tosielämän esimerkkinä käytettävyydestausta voidaan suorittaa siten, että julkaisun (release) loppuun varataan aika testauksen tulosten huomioimiselle,

jotta ne saadaan jo samaan julkaisuun korjattua, tai ne voidaan sijoittaa seuraavan iteraation tai julkaisun työlistalle (engl. backlog). Joissakin tapauksissa käytettävyydestä on tehty vain epävirallisesti, ja tulosten huomiointi on projektiryhmän suopeuden varassa. (McInerney & Maurer, 2005, s. 23.)

3.5 Käytettävyydestä osallistuvat henkilöt

Tässä alaluvussa esitellään testaukseen osallistuvat henkilöt ja heihin kohdistuvia vaatimuksia ja suosituksia perustuen aiempaan tutkimukseen. Ensin esitellään itse käytettävyydestä, joka voi myös toimia käytettävyydestä sessioiden valvojana, ja sitten koehenkilöt, joiden kohdalla keskitytään erityisesti vaatimuksiin kohderyhmävastaavuudesta.

Käytettävyydestä järjestämiseksi tarvitaan aina vähintään yksi henkilö, itse käytettävyydestä. Käytettävyydestä roolin suuruus voi vaihdella, ja useita hänen rooliinsa kuuluvia tehtäviä voidaan automatisoida (esimerkiksi käytettävyyden kerääminen koesessioista) (West & Lehman, 2006, s. 1). Koehenkilöitä testaukseen voi osallistua kerralla yksi tai useampi testausmenetelmän vaatimuksista riippuen, ja useita testausseksioita järjestettäessä koehenkilöiden määrä tietenkin nousee. Kaikki testaukseen osallistuvat henkilöt on syytä saada, jos mahdollista, mukaan projektin ulkopuolisista, jolloin näkökulma ei ole yhtä herkästi vääristynyt (Rubin, 2008, s. 46–47).

3.5.1 Käytettävyydestä

Käytettävyydestä on käytettävyydestä järjestämisen kannalta olennaisin henkilö. Hän muun muassa toteuttaa käytännön käytettävyydestä sessiot. Tässä roolissa hänestä käytetään usein nimitystä käytettävyydestä valvoja (Rubin, 2008, s. 46–47). Silloinkin kun käytettävyydestä sessioiden järjestämiseen on käytettävissä suurempi työryhmä, on varsinainen käytettävyydestä valvoja tärkein yksittäinen henkilö (Rubin, 2008, s. 45–46).

Kuten muidenkin käytettävyydestä osallistuvien henkilöiden, myös käytettävyydestä olisi parasta tulla projektiryhmän ulkopuolelta. Tavoitteena on sen välttäminen, että testaukseen osallistuvilla henkilöillä olisi liikaa tietämystä järjestelmästä tai testauksen tehokkuutta vähentäviä intressejä, kuten testaus tulosten hyvyteen perustuva tulospalkkio. Jos kuitenkin ketään muuta ei ole saatavilla, jo se, että testauksesta jotakin ymmärtävä projektitiimin jäsen itse testaa järjestelmää, on parempi kuin ei testata lainkaan. Mikäli erityistä testausasiantuntijaa (esimerkiksi psykologitaustalla tai edes markkinoinnista tietävää) ei ole saatavilla, kuka tahansa auttavilla sosiaalisilla taidoilla (Krug, 2006) varustettu projektitiimin ulkopuolinen, tai hätätapauksessa sisäinenkin, työntekijä kelpaa. Vuoro ja roolit voivat myös kiertyä eri henkilöillä. (Rubin, 2008, s. 46–47.)

3.5.2 Koehenkilöt

Eri projekteihin sopivat erilaiset koehenkilöt. Toivotut taustamuuttajat voivat vaihdella suurestikin sekä sen mukaan, minkälaisia järjestelmän loppukäyttäjät ovat, että sen mukaan, kuinka tarkkaa kohderyhmävastaavuutta kokeessa vaaditaan. Yleisesti voidaan sanoa, että koehenkilöiden tulisi vastata loppukäyttäjää, ja heiltä vaaditaan testausmenetelmien kannalta välttämättömiä kykyjä, kuten ääneen ajattelu -tyyppisissä testeissä kykyä selostaa tekemisiään kokeen valvojalle ja internetin kautta suoritettavissa korttien lajittelu -tyyppisissä kokeissa riittävät tietokoneenkäyttötaidot testiohjelman käyttämiseksi. Yleinen vaatimus on myös niin sanotun sisäisen tiedon vähäisyys. Koehenkilöllä ei saa olla sellaista tietoa ohjelmasta, mikä helpottaa ohjelman käyttöä. Hänellä ei myöskään saa olla intressejä tulosten vääristelyyn (esimerkiksi vaikeuksiaan vähättelemällä, esitietokyselyssä taitojaan pimittämällä tai korttien lajittelussa huijaamalla) esimerkiksi paremman palkkion tai kokeen valvojan miellyttämisen toivossa. (Krug, 2006.)

Organisaation sisäiset koehenkilöt, siis esimerkiksi saman yrityksen mutta eri osaston työntekijät, ovat siitä hankalia, että heillä on usein sisäistä tietoa, liikaa kohdealueosaamista ja kaiken lisäksi liikaa pyrkimystä hyvin tuloksiin, jotta he pystyisivät todella edustamaan loppukäyttäjää, vaikka olisivatkin varsinaisen projektiryhmän ulkopuolelta. Muutamaankin käyttötarkoitukseen heistäkin on, esimerkiksi testauksen testaamiseen, hyvin aikaisen vaiheen tutkivaan testaukseen (perustavaa laatua olevien ongelmien löytämiseksi) ja ”parhaan tapauksen” testaukseen (so. kuinka optimaalisessa tilanteessa kokenut käyttäjä pärjää järjestelmälle). Sisäisenkin koehenkilö kannattaa tuki valita mahdollisimman hyvin loppukäyttäjää vastaavasti. (Rubin, 2008, s. 133.)

Useimmissa tapauksissa sisäisiä koehenkilöitä parempia ovat tuttavat, naapurit ja sukulaiset, jotka ominaisuuksiltaan vastaavat loppukäyttäjää. He ovat usein käyttökelpoisempia, sillä heiltä puuttuu sisäpiiritieto ja epärealistinen kohdealueosaaminen. Edellytyksenä on kuitenkin, että testauksen kohde esitellään heille tavalla, jolla heille ei tule paineita miellyttää kokeen valvojaa. Näin he ovat todennäköisesti ihan päteviä koehenkilöitä – ja helppoja ja nopeita hankkia. (Krug, 2006; Rubin, 2008, s. 134.)

Koehenkilöitä voi myös kerätä yrityksen nettisivujen ja erilaisten kampanjoiden kautta, ja ylläpitää heistä tietokantaa, josta poimia eri sessioita varten soveltuvia koehenkilöitä. Tällöin on huomioitava otannan mahdollinen vinous. Vapaaehtoisesti, oma-aloitteisesti ilmoittautuneet koehenkilöt voivat olla yrityksen faneja ja tuotteiden pitkäaikaisia käyttäjiä. Lakien ja säädösten puitteissa myös muita kautta kerättyjä listoja ja tietoja asiakkaista voidaan käyttää erityisesti kokeneiden koehenkilöiden keräämiseen. (Rubin, 2008, s. 134–135.)

Yrityksellä voi olla myös myynnin kautta kontakteja asiakkaisiin (tai potentiaalsiin asiakkaisiin), jolloin esimerkiksi suuren yritysasiakkaan organisaatiosta voi löytyä vapaaehtoisia koehenkilöitä hyvinkin moneen eri käyttäjäprofiiliin. Erilaiset yhteisöt, kuten käyttäjäryhmät, ammattiyhteisöt, LinkedIn-ryhmät, tuki- ja harrastusfoorumit, opiskelijajärjestöt ja jopa seurakunnat ja etujärjestöt

voivat olla sopivia väyliä tietynlaisten koehenkilöiden rekrytointiin. Koehenkilöitä voi kerätä myös lehti-ilmoituksilla. Tällöin saa todennäköisesti yhteyden teknisesti vähemmän kokeneisiin koehenkilöihin. Työvoimatoimistojen kautta voi saada esimerkiksi maahanmuuttajia kansainvälistä testausta varten, ja heidän kauttaan voi saada mainostettua mahdollisuutta päästä koehenkilöksi. (Rubin, 2008, s. 136–137, 141–143.)

3.5.3 Koehenkilöiden kohderyhmävastaavuus

Vaatus koehenkilöiden kohderyhmävastaavuudesta on jakanut tiedemaailmaa ja käytettävyydestä harjoittajia. Toisten mukaan se on ensiarvoisen tärkeää ja pienikin virhe koehenkilöiden poiminnassa tekee tulokset epävalideiksi (Rubin, 2008, s. 115), kun taas toisten mukaan kohderyhmävastaavuudesta ei yleensä kannata stressata liikaa, kunhan muistaa arvioida tuloksia koehenkilöiden ja lopullisten käyttäjien mahdolliset erot muistaen (Krug, 2006).

Voitaneen sanoa, että tulokset ovat luotettavimpia, kun koehenkilöt ovat lopullisia käyttäjiä, tai ainakin näitä vastaavia henkilöitä. Otoksen koossa pätee viisaus ”mitä suurempi, sen parempi”, mutta toisaalta jo yhdenkin koehenkilön otos on äärettömän paljon parempi kuin ei testausta ollenkaan (Krug, 2006). Pieni otos kuitenkin aiheuttaa suuren riskin – mitä jos ainoa testattava koehenkilö onkin syystä tai toisesta erityisen epäsopeva ja loppukäyttäjää huonosti kuvaava? Pienillä koehenkilöiden otoksilla täytyy olla erityisen varovainen, ettei koehenkilöihin satu erityisen epäsopevaa ja huonosti edustavaa henkilöä (Nielsen, 1993, s. 175).

Optimaalisen otoksen koon määrittämiseen vaikuttavat ainakin tavoiteltu luottamusaste, testaukseen saatavilla olevien resurssien määrä, soveltuvien koehenkilöiden saatavuus ja yksittäisen testisession pituus (Rubin, 2008, s. 125). Riittävä otos tyypillisimpien käytettävyysohjelmien löytämiseksi voisi olla 4-5 koehenkilöä kustakin käyttäjätyypistä (Rubin, 2008, s. 126), ja testien toistaminen riittävän monta kertaa muutosten onnistumisen todentamiseksi ja eri vaiheissa toteutettavien ominaisuuksien testaamiseksi (Krug, 2006). Useasti testaaminen toisaalta oikeuttaa pienemmän määrän koehenkilöitä testiä kohden (Rubin, 2008, s. 126).

Viiden käyttäjän ”riittävyys” ei kannata hirttäytyä liikaa kiinni. Koikeissa viidellä koehenkilöllä on voitu löytää 55–100 % järjestelmässä olevista tunnetuista käytettävyysohjelmista (Faulkner, 2003, s. 379). Viiden koehenkilön otannalla löydettiin samassa tutkimuksessa keskimäärin 85 % ongelmista (Faulkner, 2003, s. 382). Toisaalta on havaittu, että testattavien käyttäjien määrää tärkeämpää on testitapausten kattavuus, ainakin mikäli koehenkilöt on hyvin valittu (Lindgaard & Chattratichart, 2007, s. 9).

RITE-menetelmä (McGinn & Chang, 2013) taasen painottaa nopeita iteraatioita prototyypin kehityksessä, jolloin voidaan testata jopa vain yhdellä käyttäjällä kerrallaan, mutta tarjoilla jokaiselle uudelle koehenkilölle sellainen uusi versio järjestelmästä, jossa edellisen session löydökset on huomioitu. Tällöin testikertoja on erittäin paljon, joten koehenkilöidenkin kokonaislukumäärä hiljalleen

nousee. Jos on mahdollista rekrytoida paljon koehenkilöitä, voidaan testattavalle järjestelmälle selvittää käyttäjäprofiilit tai persoonat ja näille osaamistasot (esimerkiksi noviisi, keskimääräinen, ekspertti). Esimerkki tästä on esitetty taulukossa 3.

TAULUKKO 3 Esimerkki käyttäjätyyppien jaottelusta (Rubin, 2008, s. 125)

Käyttäjäprofiili	Noviisi	Ekspertti
Rahastoasiantuntija	4	4
Pankkitoimihenkilö	4	4

Yleensä ottaen käytettävyydestä testataan järjestelmän ”komponenteista” kaikkein odottamattomimmin käyttäytyvää – siis käyttäjää, ihmistä. Tämän vaatimuksen, eli ennalta arvaamattoman ja joskus epärationaalisen käytöksen, toteuttaa melkein kuka tahansa ihminen (Ferreira ym., 2007, s. 57), mikä puoltaa kohderyhmävastaavuuden pienempää merkitystä. Ulkoisten ja hyvin kohderyhmää vastaavien koehenkilöiden rekrytoinnissa on ongelmansa: sessioista koituu vaivaa ja kuluja, joten siihen ei välttämättä ole aina resursseja. Erityisesti kiireessä tai projektien alkuvaiheessa voikin olla järkevintä rekrytoida koehenkilöt mahdollisimman helposti, käytännössä siis usein sellaisia työkavereita, joilla on kohdealueosaamista (ns. domain-tietämys) mutta ei osuutta projektiin, tai esimerkiksi sopivan alan opiskelijoita. Toisaalta joudutaan huomioimaan erityisen spesifien koehenkilöiden hankkimisen vaikeus – voi olla mielekästä luopua ainakin suuren mittakaavan käytettävyydesteistä ja keskittyä enemmän esimerkiksi heuristiseen evaluointiin (Nielsen, 1994b, s. 9–11). (Sy, 2007, s. 123–124.)

3.6 Yhteenveto

Tässä luvussa määriteltiin käytettävyyteen liittyviä peruskäsitteitä ja toimintoja sekä kuvattiin käytettävyyssuunnittelu ja sen osana erityisesti käytettävyydestäusta, sen merkittävimpiä menetelmätyyppejä ja menetelmiä. Lopuksi pohdittiin myös käytettävyydestäukseen osallistuvia henkilöitä ja heidän roolejaan.

Käytettävyydestäuksen merkittävyttä ei aina tiedosteta, ja se jää raskaiden menetelmiensä ja kalleutensa vuoksi usein tekemättä. Ratkaisuksi näihin ongelmiin on esitetty kevyttä käytettävyydestäusta, jolloin kevyitä käytettävyydestäuksen menetelmiä toteutetaan mahdollisimman tehokkailla tavoilla. Kevyen käytettävyydestäuksen menetelmät pohjautuvat yleensä tavallisen käytettävyydestäuksen menetelmiin, mutta kvantitatiivisten tulosten sijasta saadaan kvalitatiivisia tuloksia ja niiden tilastollisen analysoinnin sijaan luotetaan testajan ammattitaitoon tärkeimpien havaintojen löytymisen varmistamisessa. Kevyen käytettävyydestäuksen eri toteutuksissa kohderyhmien kokoon ja lopullisten käyttäjien vastaavuuteen on suhtauduttu hyvin eri tavoin, mutta tulosten

luotettavuuden varmistamiseksi kohderyhmävastaavuudesta on ainakin erikoisempien käyttäjäryhmien kohdalla huolehdittava. Koehenkilöiden määrää tärkeämpi kysymys näyttäisi olevan testitapausten kattavuus.

4 KÄYTETTÄVYYSTESTAUS OSANA KETTERÄÄ OHJELMISTOKEHITYSTÄ

Tämän luvun tarkoitus on antaa käsitys keskeisimmistä kysymyksistä koskien käytettävyydestä suoritusta osana ketterää ohjelmistokehitystä sekä tavoista, joilla yhteensovitus on aiemmissa tapauksissa tehty. Ensin käsitellään käytettävyydestä suoritusta ja ketterän ohjelmistokehitystavan integrointia yleisellä tasolla ja ”kahden eri maailman” keskeisimpiä eroavaisuuksia ja yhteneväisyyksiä. Seuraavaksi tarkastellaan aiempaa tutkimusta aiheesta ja perehdytään muutama tapaukseen. Tämän jälkeen pohditaan tiettyjä olennaisia kysymyksiä, kuten käytettävyyden hyödyllisyyttä, käytettävyydestä suoritusta oikeaa ajoitusta, määrää ja hintaa sekä perehdytään käytettävyydestä suoritukseen käytännössä. Lopuksi käsitellään vielä joitakin käytettävyydestä suoritusta sudenkuoppia ja tapoja käytettävyydestä suoritusta havaintojen validointiin.

4.1 Ketterä käytettävyydestä suoritusta?

Tässä alaluvussa käsitellään yleisellä tasolla käytettävyydestä suoritusta ja ketterän ohjelmistokehityksen integrointia. Erityisesti keskitytään siihen, miten ketterässä ohjelmistokehityksessä käytettävyydestä suoritusta yleensä tehdään, mitkä piirteet käytettävyydestä suoritusta ja ketterästä ohjelmistokehityksestä ovat luontevasti yhteensovitettavissa ja mitkä piirteet voivat tuottaa enemmän ongelmia.

4.1.1 Käytettävyydestä suoritusta ketterässä ohjelmistokehityksessä

Ketterä kehitys ei itsessään ota kantaa käytettävyydestä suoritusta toteuttamiseen (Bankston, 2003, s. 3; Budwig ym., 2009, s. 3076; Constantine, 2001; Itkonen ym., 2005, s. 2–3; Kane, 2003, s. 40; Sohaib & Khan, 2010, s. 1). TDD:n (Test-Driven Development) ja hyväksymistestä suoritusta (engl. acceptance testing) varjossa käytettävyyden testä suoritusta ei juuri anneta huomiota (Sohaib & Khan, 2010, s. 5, 2011,

s. 1). Ketterä kehitys ja käytettävyydestä pyrkivät kuitenkin pohjimmiltaan samaan asiaan: hyvään ohjelmistoon (Sohaib & Khan, 2010). Hyvä ohjelmisto on tarkoituksenmukaisessa mitassa myös hyvin käytettävä, ja toisaalta jotta ohjelmisto olisi käytettävä, sillä täytyy pystyä suorittamaan tehtävät tehokkaasti.

Organisaatioiden siirtyessä hyödyntämään ketterää ohjelmistokehitystä käytettävyydestä menetelmät on saatettu ottaa käyttöön sellaisenaan niitä ketterään kehitystapaan mitenkään räätälöimättä. Pahimmassa tapauksessa ketterän ohjelmistokehitystavan noudattaminen on voinut johtaa myös käytettävyydestä laiminlyöntiin, eikä käytettävyys siis välttämättä ole parantunut (Constantine & Lockwood, 2003, s. 746). Ketterien menetelmien yleistymisen myötä moni myös käytettävyydestä hyödyttävä toimintatapa tai kulttuurimuutos on kuitenkin tullut tärkeämmäksi. Esimerkiksi ponnistelu asiakkaan saamiseksi aktiivisesti mukaan kehitystyöhön on toisaalta auttanut käytettävyyden parantamisessa, aktiivisempi kommunikaatio sekä nopea ja iteratiivinen ja inkrementaalinen työskentelytapa ovat sopineet hyvin myös käytettävyydestä tulleelle, ja ne ovatkin pääosin myös parantaneet käytettävyyttä (Budwig ym., 2009, s. 3077; Kane, 2003, s. 43–44; Lee & McCrickard, 2007, s. 2).

Sekä käytettävyyden että ketterien menetelmien kannalta asiakas, ja erityisesti asiakkaan käyttäjät, on tärkein osapuoli: ilman heitä ei ole tarvetta ohjelmistotuotannolle tai edes liiketoimintaa yleensä (Itkonen ym., 2005, s. 1; Sohaib & Khan, 2011, s. 5). Käytettävyydestä pyritään arvioimaan sitä, kuinka hyvin käyttäjät saavuttavat haluamansa tavoitteet kehitettävää ohjelmaa tai järjestelmää käyttäessään. Käytettävyydestä sessioiden, siis esimerkiksi yksinkertaisen äänen ajattelun, suurin etu verrattuna automatisoituihin testeihin (kuten yksikkötestien) on se, että ne voivat auttaa löytämään merkittäviäkin ongelmia järjestelmässä, joka noudattaa täysin määrittelyjä (Ferreira ym., 2007, s. 7). Tämä voi olla merkki puutteellisesta määrittelystä tai yksinkertaisesti irrationaalista ja vaikeasti ennustettavissa olevasta inhimillisestä käytöksestä. Käytettävyydestä on luonteeltaan hyvin vaikeasti automatisoitavissa, joten se täydentää esimerkiksi yksikkötestejä ja käyttäytymisohjatun suunnittelun (engl. Behavior-Driven-Development, BDD) mukaisten testien (Solis & Wang, 2011) ajamista tuomalla testituloksiin mukaan muutoin vaikeasti simuloitavissa olevan tekijän, ihmisen. Vaikka monilla muillakin testaustavoilla on perusteensa ja paikkansa, varsinaista käyttäjätestausta on ihmisluonteen monimutkaisuuden ja arvaamattomuuden vuoksi hyvin vaikea simuloida tai automatisoida.

Suurin osa niin automaattisesta kuin manuaalisestakin testauksesta testaa sitä, toteuttaako järjestelmä sille asetetut vaatimukset. Tämä on tietyllä tavalla osaltaan myös käytettävyydestä: ollakseen käytettävä (engl. usable) järjestelmän täytyy olla hyödyllinen (engl. useful), ja mahdollistaessaan käyttäjiensä haluamien tehtävien toteuttamisen järjestelmän voi katsoa olevan hyödyllinen (Bankston, 2003, s. 5). Osa järjestelmän käytettävyydestä tulee testatuksi muun testauksen yhteydessä. Voisi ehkä ajatella, että käytettävyyden perusvaatimus on jo tällä täytetty. Käytettävyydestä tuleekin testatuksi muun testauksen yhteydessä. Voisi ehkä ajatella, että käytettävyyden perusvaatimus on jo tällä täytetty. Käytettävyydestä tuleekin testatuksi muun testauksen yhteydessä.

4.1.2 Käytettävyydestä ja ketterän ohjelmistokehityksen integrointi

Erilaisia integrointitapoja on esitetty useita. Esimerkiksi Krug (2006) käsittelee käytettävyydestä liittämistä mihin tahansa ohjelmistokehitystapaan. Kane (2003) taas löytää ketterien menetelmien peruseräkkeistä paljon hyvin kevyen käytettävyydestä kanssa yhteensopivaa. Hänen mukaansa (1) käyttäjien ymmärtäminen sopii hyvin yhteen yksilöiden ja vuorovaikutuksen kanssa ja (2) toimivan ohjelman painottaminen puolestaan sopii kevyelle käytettävyydestä hyvin, mutta sovitettuna siten, että käyttäjän ja käyttöliittymän välisen vuorovaikutuksen täytyy toimia, varsinaisella ohjelmalla ei niin väliä. Tämä tarkoittaa, että kevyen käytettävyydestä näkökulmasta prototyyppi tai järjestelmän malli riittää. (3) Vuorovaikutus asiakkaan kanssa on selkeästi käytettävyyttä edistävä tekijä - erityisesti jos varsinaisia loppukäyttäjiä ei ole saatavilla, jolloin asiakkaan edustaja voi toimia tällaisen edustajana. (4) Muutokseen reagointi on sen sijaan lähtökohtaisesti hiukan ristiriidassa käytettävyydestä kanssa, sillä käytettävyydestä toteuttaminen vaatii aina jonkin verran suunnitelmista, vähintäänkin täytyy tietää testitapaukset ja mahdollisesti suunnitella koehenkilöiden valinta ja rekrytointi, mutta toisaalta taas kevyt käytettävyydestä voi toimia muutoksen katalyyttinä. Yleisimpien ketterien menetelmien ja käytettävyydestä yhteensovittamisessa on muutenkin ongelmansa: XP:stä puuttuu yleensä kulloinkin toteutettavaltakin osiolta järjestelmästä "tuotantomalli", jota testata ennen toteutusta, ja toisaalta scrumin käyttäjät eivät yksinään anna juuri eväitä testaukselle (Sohaib & Khan, 2010, s. 5). Toiseksi usein mainitaan asiakkaan edustaja, jolla on projektissa suuri vastuu asiakkaan näkökulman esiintuomisessa. Puhdasoppisessa XP:n soveltamisessa asiakkaan edustaja on jopa istutettu täysin toimittajaorganisaation tiloihin, joissa hän osallistuu tuotteen toteutukseen erityisesti testejä laatimalla ja validoimalla (Beck & Andres, 2005, s. 58). Asiakkaan edustaja valvoo käytännössä myös käyttäjän etua tuotteen tai järjestelmän kehityksessä, mutta hän ei välttämättä edusta todellista loppukäyttäjää.

Integrointia on myös useasti kokeiltu käytännössä (Lee ym., 2009; McGinn & Chang, 2013; Parsons ym., 2007; Singh, 2008; Sy, 2007). Integrointitavat yleensä perustuvat yksinkertaisuudessaan siihen, että sovitetaan eri toimintatapojen iteratiivisuus yhteen, varmistetaan, että projektitiimissä on tavalla tai toisella mukana käytettävyyssosaamista ja määritellään tapa, jolla palaute annetaan ja siihen reagoidaan. Eroja esimerkiksi palautteen käsittelyn ja sidosryhmien sitouttamisen välillä on paljon, eikä tulosten validointiin tieteellisesti kestäväällä metodilla näyttäisi useinkaan olevan keskitytty riittävästi.

Ohjelmiston ja järjestelmän suunnittelu on ketterässä ohjelmistokehityksessä aina luonteeltaan iteratiivista. Käytettävyydestä tuottaa suunnittelua ohjaavaa dataa ja on siten myös luonteeltaan iteratiivista. Erityisesti käyttöliittymäsuunnittelun iteraatioiden kesto on usein totuttua sprinttisykliä lyhempi: viikkojen sijasta päiviä tai tunteja. Suunnitelma muuttuu tai kehittyy siis nopeasti kokemusten ja palautteen pohjalta. (Ferreira ym., 2007, s. 52.)

Iteratiivisuus sopii siis hyvin sekä ketterän ohjelmistokehityksen että käytettävyydestestauksen harjoittajille. Kehittäjäryhmään (tai varsinaisen kehittäjäryhmän lisäksi) tarvitaan ainakin vastuuhenkilö käytettävyydestestaukselle (erotuksena ketterän kehitystavan moniosajisiin), pelkän asiakkaan edustajan (toimittajan tiloissa oleskelevan asiakkaan edustajan, kuten XP:ssä) sijasta täytyy olla tekemisissä myös loppukäyttäjia tai vastaavia. Kehittäjien toteutuksenaikaisen testauksen, kuten yksikkötestien, koodikatselmointien ja nopean toiminnallisen testauksen, ja asiakkaan suorittaman hyväksyntätestauksen lisäksi on pyrittävä saattamaan käytettävyydestestaus osaksi kehitysprosessia. Käyttäjätarinat voidaan ottaa osaksi skenaarioperusteista suunnittelua. (Sohaib & Khan, 2010, s. 6.)

Ketterän testauksen osa-alueista, joita on käyty läpi kuviossa 2, käytettävyydestestaus kuuluu likimain kokonaisuudessaan manuaaliseen testaukseen. Se antaa arvioita tuotteesta ja on selvästi lähempänä liiketoimintalogiikan näkökulmasta testaamista kuin varsinaista teknologian testaamista. Tällaista testausta on erittäin vaikea automatisoida, koska se on usein kontekstiriippuvaista ja kokeen tulokset riippuvat paljolti koehenkilön kokemuksesta ja jopa vaistoista (Crispin & Gregory, 2009, s. 190). Toisaalta osa käytettävyydestestauksesta, erityisesti aikaisien prototyyppien osalta, voi kuulua myös toiseen neljännekseen. Käytettävyydestestaus-osa-alueen sisältä voidaan vielä eritellä esimerkiksi seuraavia menetelmiä tai osa-alueita (Crispin & Gregory, 2009, s. 203–204):

- Persoona-testaus
- Navigaatio
- Vertailu kilpailijoihin

Tällöin erilaiset läpikäynnit ovat pääosin persoona-testausta (eri persoonia edustavia koehenkilöitä suorittamassa erityyppisiä tehtäviä), erilaiset korttien lajitte-
lut (avoin, suljettu tai välimuoto) testaavat navigaation termejä ja rakennetta, samoin osaltaan läpikäynnit, ja itse tai koehenkilöiden kanssa suoritettavat koesessiot kilpailijoiden tuotteilla ovat vertailua kilpailijoihin ja näiltä oppimista. Eräs yleisimpiä käytettävyydestestauksesta ketterään ohjelmistokehitykseen tuotuja osioita on persoona-testaus. On myös yleistä että käytettävyyssuunnittelusta ainakin osa suoritetaan ennen toteutuksen aloittamista. (McInerney & Maurer, 2005, s. 21.)

Käytettävyydestestaus kuuluu laadullisen (engl. non-functional) testauksen metodeihin. Tieteellisen tarkastelun kestävien tulosten saaminen vaatii tilastollisesti merkittävän otoksen keräämistä, mikä usein on haastavaa ja aikaa vievää. Käytettävyydestestaus voi kuitenkin tuottaa sekä kvalitatiivisia että kvantitatiivisia tuloksia. Käytettävyydestestaus on ketterässä ohjelmistokehityksessä perinteisesti ajateltu osaksi hyväksyntätestausta: kun asiakas testaa, että järjestelmä täyttää vaatimukset, käytettävyys tulee samalla testatuksi (Dicks, 2002, s. 28). Mikäli käytettävyydelle ei kuitenkaan ole kirjattu vaatimuksia hyväksyntätestausta varten, eikä asiakas oma-aloitteisesti testata järjestelmää loppukäyttäjällä tai sopivasti edustavalla koehenkilöllä, ei varsinaisen käytettävyys välttämättä tule testatuksi. Siksi onkin esitetty käytettävyydestestauksen järjestämistä itsenäisenä osana hyväksyntätestausta (Bankston, 2003, s. 11; Lee & McCrickard, 2007, s. 4).

Tässä kuitenkin syntyy ongelma: käytettävyydestestauksen havaintoja ei voida enää korjata samassa sprintissä tai julkaisussa, vaan ne ovat jo päätyneet asiakkaalle ja työllistävät projektitiimiä muiden korjauksien lisäksi. Jos taas testaus tehdään liian paljon ennen hyväksyntätestausta, muutokset ehditään tehdä mutta regressiotestausta ei, eikä taaskaan tiedetä mitä asiakkaan nähtäväksi toimitettiin (Kane, 2003, s. 44). Toisaalta voidaan ajatella, että asiakas tulee suorittamaan käytettävyydestausta joko tietoisesti tai osana muuta testausta riippumatta siitä, suorittaako testaaaja sitä vai ei, ja pahimmassa tapauksessa asiakkaan havainnot johtavat asiakastytyvyyden laskuun (Donahue, 2001, s. 33).

Sohaib ja Khan (2011, s. 6) ehdottavat XP:n ja edullisen käytettävyydestestauksen integroinnista seuraavaa:

1. Skenaarioiden käyttö käyttäjätarinoiden ohella
2. Korttien lajittelu julkaisun suunnittelun ohella
3. Heuristinen arviointi hyväksyntätestauksen ohessa
4. Yksinkertaistettu ääneen ajattelu -testaus käyttöönottoaiheessa

Sohaibin ja Khanin (2011, s. 6) mainitsevat menetelmät perustuvat itse asiassa Nielsenin (1995a) esittelemiin edullisen käytettävyydestestauksen menetelmiin (ks. kohta 3.4.3), ja niitä voidaan soveltaa myös muussa kuin XP:n mukaisessa ketterässä kehityksessä.

McInerney ja Maurer (2005) puolestaan tutkivat käyttäjakeskeisen suunnittelun sovittamista ketterään ohjelmistokehitykseen. He haastattelivat kolmea käytettävyyssammattilaista heidän kokemuksistaan oikeissa projekteissa. Haastateltavat työskentelivät eri organisaatioissa ja eri projekteissa keskenään, mutta kaikki työskentelivät ketterissä projekteissa. Havainnot käytettävyyssuunnittelun ja ketterän ohjelmistokehityksen yhteensovittamisesta voivat antaa myös suuntaviivoja käytettävyydestestauksen ja ketterän ohjelmistokehityksen yhteensovittamiselle esimerkiksi roolien tai yleisten toimintatapojen osalta. Minkä tahansa järjestelmän käytettävyys on paljolti seurausta sen suunnittelun onnistumisesta – niin käyttöliittymäsuunnittelun (Butler, 1996, s. 10) kuin muunkin suunnittelun (Ferré, Juristo, Windl, & Constantine, 2001). Läheisen suhteen vuoksi myös käyttäjakeskeisen suunnittelun, joka sisältää paljon käyttöliittymäsuunnittelun kanssa yhteisiä elementtejä, ja ketterän ohjelmistokehityksen yhteensovittamisen tarkastelu on mielekästä.

McInerneyn ja Maurerin (2005) tutkimus paljastaa useita olennaisia eroja käyttäjakeskeisen suunnittelun ja ketterän ohjelmistokehityksen lähestymistavoissa. Ensimmäinen ero liittyy koehenkilöiden hyödyntämiseen: käyttäjakeskeisessä suunnittelussa pyritään hyödyntämään noin viittä koehenkilöä, jotka valitaan ennalta suunniteltujen profiilien perusteella. Ketterässä kehityksessä taas muutokset viedään tuotantoympäristöön mahdollisimman nopeasti, ja palautetta saadaan todellisilta käyttäjiltä. Toinen eroavaisuus koskee järjestelmän arviointia: ketterässä kehityksessä kunkin iteraation lopuksi arvioidaan tuotantoon vietävää koodia ja sen hyvyyttä, kun taas käyttäjakeskeisessä suunnittelussa arviointi aloitetaan jo hyvin karkeista suunnitelmista. Eri järjestelmäversioille tai muutoksille haetaan hyväksyntä ketterässä kehityksessä esittelemällä toimivaa

ohjelmakoodia tai järjestelmää, kun taas käyttäjäkeskeisessä suunnittelussa käytetään menetelmiä kuten tutkivaa testausta ja arvioidaan esimerkiksi käyttäjätyytyväisyydestä kerättyjen tietojen perusteella järjestelmän hyvyttä. (McInerney & Maurer, 2005, s. 23)

Taulukko 4 kuvaa McInerneyn ja Maurerin (2005, s. 22) mukaan merkittävimpiä ketterän ohjelmistokehityksen eroja käyttöliittymäsuunnittelun menettelytapoihin verrattuna, ja joitakin ehdotuksia yhteensovittamisen helpottamiseksi. Ketterässä ohjelmistokehityksessä järjestelmää kehitetään vertikaalisissa siivuissa, eli ominaisuus tai käyttäjätarina kerrallaan. Käyttöliittymäsuunnittelun osalta tähän voidaan mukautua suunnittelemalla pientä osaa ohjelmistosta kerrallaan ja etenemällä suunnittelusta hyvin nopeasti varsinaiseen toteutukseen. Ketterässä ohjelmistokehityksessä käyttöliittymäsuunnittelijan roolia ei yleensä tunneta, mutta tällöin käyttöliittymäsuunnittelukin voi olla koko tiimin vastuulla, jolloin tiimin jäsenistä se, jolla on osaamista, aikaa ja kiinnostusta, ottaa tehtävät hoitaakseen. Toisissa tapauksissa on voitu käyttää myös kiertävää käytettävyyssiantuntijan roolia (Lee & McCrickard, 2007). Järjestelmän toteutus tehdään ketterässä ohjelmistokehityksessä ohjelmoijien yhteistyönä, ja tapa millä käytettävyyssiantuntija voi tukea kehitystä, on olla käytettävissä kehityksen aikana ad hoc -tyyppisiin keskusteluihin erityisesti järjestelmän käytettävyyssuunnitteluun liittyen. (McInerney & Maurer, 2005)

TAULUKKO 4 Yhteensovittavia elementtejä ketterässä kehityksessä ja käyttöliittymäsuunnittelussa

Menettelytapa ketterässä ohjelmistokehityksessä	Mahdollisia vaikutuksia käyttöliittymäsuunnitteluun
Kukin iteraatio keskittyy vertikaalisiin siivuihin ohjelmistosta.	Käyttöliittymäsuunnittelu keskitetään pienen osaan ohjelmistoa, ja edetään nopeasti konseptista koodiin.
Ei ole käyttöliittymäsuunnittelijan roolia.	Käyttöliittymäsuunnittelu voi olla tiimin vastuulla.
Ohjelmoijat (esimerkiksi pariohjelmointina) suunnittelevat ja toteuttavat yhtä aikaa.	Käyttöliittymäsuunnittelijan täytyy olla valmiina "ad hoc"-tyyppisiin keskusteluihin.

McInerneyn ja Maurerin (2005) haastatteleminen käytettävyyssammattilaisten projekteissa käytettävyyssuunnittelusta ainakin osa tehtiin etukäteen. Asiakkaan edustaja tiimeissä oli usein yksi yrityksen työntekijöistä, jolla oli kohdealue- tai tuotetietämystä.

4.2 Tapaustutkimuksia käytettävyydestauksen ja ketterän ohjelmistokehityksen integroinnista

Aiempia yrityksiä ja tapoja yhdistää ketterä ohjelmistokehitys ja käytettävyydestaus ketteräksi käytettävyydestaukseksi on useita. Tässä alaluvussa esitellään joitain niistä keskittyen erityisesti tapoihin, joilla on ratkaistu integroinnin keskeisimpiä ongelmia kuten asiakasnäkökulman mukaanottoa, käytettävyydestauksen tulosten nopeaa käsittelyä ja projektiryhmän viestinnän tehostamista. Nämä tapaustutkimukset antavat osviittaa siitä, kuinka kevyen käytettävyydestauksen ja ketterän ohjelmistokehityksen yhteen sovittaminen kannattaa tehdä.

4.2.1 XSBD-malli

Lee, McCrickard ja Stevens (2009) kehittivät Meridium-yritykselle XP:n ja käytettävyyssuunnittelun yhdistävän XSBD-mallin (eXtreme Scenario-Based Design). Mallilla pyrittiin paikkaamaan XP:n (ja muiden vastaavien ketterien menetelmien kuten scrumin) kehittäjäkeskeisyyttä ja varmistamaan lopputuotteen käytettävyys (Lee ym., 2009, s. 1).

XSBD-mallin tarkoitus on vastata ketterän kehityksen ja käytettävyydestauksen suurimpiin ongelma-kohtiin. Mallin mukaan toimintaa ohjaavat seuraavat periaatteet (Lee ym., 2009, s. 7):

- Koko projektitiimi pidetään selvillä asiakkaan tärkeimmistä prioriteeteista, mukaan lukien käytettävyydestauksen havainnot.
- Väittämät⁵ ovat käytännöllinen keino eri käyttöliittymäsuunnitteluvaihtoehtojen tallettamiseen ja arviointiin.
- Käytettävyyssuunnittelun ja ketterän ohjelmistokehitystiimin työn synkronoinnin varmistamiseksi tarkastellaan aktiivisesti toteutusten tasavauhdissa pysymistä.
- Käyttöliittymän suunnittelu toteutetaan inkrementaalisesti päätösten ja testauspalautteen perusteella.

XSBD otettiin käyttöön Meridium-yrityksessä tuotekehitysprojektissa. Mallin "tulikaste" saatiin projektissa, jossa vanhanaikaista, papereiden käyttöön perustuvaa järjestelmää parantamaan kehitettiin kosketusnäytölliselle laitteelle soveltuva ohjelmisto. Ketterät menetelmät olivat paljolti vieraita projektiryhmälle, ja käytettävyyssiantuntijan toimiminen mukana projektissa oli myös heille uutta. Projektiryhmän päästyä yli alun hankaluuksista tuloksia kuvattiin kautta

⁵ Väittämät (engl. "claims") ovat ominaisuuksien kuvauksia, jotka sisältävät vaihtokauppoja eri suunnitteluvaihtoehtojen välillä. Ne sisältävät erityisesti eri vaihtoehtojen positiivisia ja negatiivisia vaikutuksia käyttöliittymälle. (Lee, McCrickard, & Stevens, 2009, s. 3)

linjan hyväksi. Tulosten perusteella tutkijat esittävät seuraavia parannuksia ketterään ohjelmistokehitykseen erityisesti projekteissa, joissa projektiryhmässä on mukana käytettävyyssiantuntija (Lee ym., 2009, s. 7):

1. Usein ristiriidassa olevien kehitystiimin ja käytettävyyssiantuntijoiden intressit on saatava pidettyä tasapainossa.
2. Uusia käyttöliittymäversioita on syytä toimittaa säännöllisesti, jotta on mahdollista saada käyttäjiltä ja asiakkailta jatkuvasti palautetta, jonka perusteella sitä voidaan parantaa entisestään.
3. Liiketoimintaan, kehitykseen ja käytettävyyssaktiviteetteihin osallistuvien ihmisten on syytä työskennellä yhdessä päivittäin.
4. Tehokkain tapa kehitettävän järjestelmän käytettävyyden saamiseksi selville on perinteinen käytettävyystestaus.
5. Edistymisen tärkein mittari on toimiva ohjelmisto, joka täyttää vaatimukset ainakin karkealla tasolla.
6. Jatkuva keskittyminen hyvään suunnitteluun, niin käytettävyydessä kuin ohjelmoinnissakin, lisää ketteryyttä.
7. Yksinkertaisuuden sopiva painottaminen sekä käyttöliittymässä että koodissa on olennaista.

4.2.2 RITE+Krug-menetelmä

McGinn ja Shang (2013, s. 66) kehittivät aiemman, suorastaan kaoottisen käytettävyystestausmallinsa korvaamiseksi kevyen käytettävyystestausmallin, jonka he nimesivät RITE+Krug -menetelmäksi. RITE+Krug-menetelmän keskeisimpinä piirteinä sen kehittäjät listaavat erityisesti Krugilta (2006) omaksuttua 3-4 koehenkilön käyttämistä testauskertaa kohden ja tulosten koostamista ja purkua nopeasti testauksen valmistuttua (McGinn & Chang, 2013, s. 64), testien toistamista usein ja kenties tärkeimpänä menetelmälle ominaisena piirteenä olennaisimpien sidosryhmien sitouttaminen sessioiden seuraamiseen verkkokonferenssina (McGinn & Chang, 2013, s. 64). RITE+Krug -menetelmässä sessio on jaettu kahdelle päivälle, jolloin kolme ensimmäistä koehenkilöä testataan ensimmäisenä päivänä ja neljäs toisena päivänä, jolloin tulokset myös puretaan ja käsitellään sidosryhmien kanssa. Tämä menettelytapa mahdollistaa myös väljemmän sessioiden ajoittamisen (McGinn & Chang, 2013, s. 64-65).

RITE+Krug -menetelmää hyödynnettiin (ilmeisesti, tästä ei mainintaa tutkimuksessa) tutkijoiden työnantajalla Oraclella tuotekehitysprojekteissa. Perinteisen, hitaan ja paljolti kyseenalaistettuja tuloksia tuottaneen mallin korvaaminen uudella ja kevyemmällä mallilla tuotti erinomaisia tuloksia: käytettävyystestauksen tulokset hyväksyttiin laajemmin kuin koskaan ennen, projektin sidosryhmät saatiin osallistumaan erinomaisesti (osallistuminen oli alkuun pakollista kunkin sidosryhmän yhdelle edustajalle, mutta sana kiiri ja vapaaehtoisia seuraajia sessioille alkoi kertyä) ja testaus tuotti tuloksia nopeammin kuin koskaan aiemmin. (McGinn & Chang, 2013.)

4.2.3 Parsonsin, Lalin, Ryuin ja Langen esittämä integrointitapa

Parson, Lal, Ryu ja Lange (2007, s. 177) esittävät useista tapauksista keräämäänsä kokemukseen perustuen seuraavia muutoksia ketterän ohjelmistokehitystavan ja käytettävyydestäuksen yhteensovittamiseksi:

- Käytettävyydestäusprosessi sovitetaan ketterään, iteratiiviseen ohjelmistokehitysprosessiin.
- Vaatimukset (käytettävyyden osalta) liitetään käyttäjätarinoihin, käyttäjäprofiileihin tai persooneihin.
- Sallitaan toteutuksen aikainen suunnittelu ja prototyypin muodostuminen iteratiivisesti.
- Käytetään toimivaa ohjelmaa tai sen osaa prototyyppinä.
- Varmistetaan, että toteutustiimissä on myös käyttöliittymäosaamista.
- Testataan käyttöliittymää säännöllisesti ja kaikissa projektin vaiheissa.
- Erikoistuneen ja projektille omistetun käytettävyyssiantuntijan sijaan käytetään samaa käytettävyyssiantuntijaa useissa projekteissa, tai tarvittaessa turvaudutaan alihankkijaan.

Käytettävyyssuunnittelun ja -testauksen sovittaminen yhteen ketterän ohjelmistokehitystavan kanssa on mahdollista menettämättä ketteryyden etuja, mutta kuitenkin parantaen ketterää ohjelmistokehitystapaa käytettävyyden huomiointin osalta merkittävästi. (Parsons ym., 2007.)

4.2.4 U-scrum-lähestymistapa

Scrumin asema yleisimpänä ketteränä menetelmänä (VersionOne, 2013) on synnyttänyt tutkimusta nimenomaan käytettävyydestäuksen soveltamisesta scrumin yhteydessä. Aiemmassa tutkimuksessa (Federoff ym., 2008; Patton, 2002) on tunnistettu useita ongelmia – jopa siinä optimaalisessa tilanteessa, että käytettävyyssiantuntijoita on scrumtiimin osana. Ongelmia ovat olleet ainakin scrumin tuoteomistajan roolin heikentyästä seuraava käytettävyyden näkökulman unohtaminen, tuoteomistajan puutteellinen käytettävyyden osaaminen ja motivaatio, sekä ketterien menetelmien puutteellinen tai jopa rajoittava näkemys käytettävyyden huomiointiin. (Singh, 2008.)

Yhden lähestymistavan käytettävyyden huomiointiin scrumissa esittää U-scrum, jossa tuoteomistajan tehtävä on jaettu kahdelle henkilölle, joista toinen kantaa vastuun käytettävyyden tai käyttäjäkokemuksen kehittämisestä (Singh, 2008, s. 555) ja toinen tuoteomistajan rooli on perinteisempi. Tällaisessa lähestymistavassa käytettävyyteen kuuluu kuitenkin resursseja jo huomattavan paljon ja vaikka käyttäjätyytyväisyys U-scrumia noudatettaessa onkin parempi kuin perinteisemmissä scrum-projekteissa, ei se ehkä riitä perusteeksi pelkästään käytettävyyden keskittävän henkilön hankkimiseksi muutoin kuin käytettävyydeltään

kriittisissä tai todella suurissa projekteissa. U-scrum ei ota toisaalta kantaa käytettävyyden testaamiseen vaan tarjoaa lähinnä uudenlaisen tavan jakaa vastuita erityisesti käytettävyyssuunnittelun yhteydessä.

4.2.5 Autodesk

Kun Alias alkoi harjoittaa ketterää ohjelmistokehitystä, käytettävyyssuunnittelutiimi päätti samalla kehittää käytettävyyssuunnitteluprosessiaan. Alias valmisti 3D-suunnitteluohjelmistoja, jotka nyt myöhemmin on sulautettu osaksi Autodeskin ohjelmistovalikoimaa Autodeskin ostettua yhtiön vuonna 2006. Vuonna 2002 varsinainen ohjelmistokehitystiimi oli jo ottamassa innolla ketteriä menetelmiä käyttöön, ja käytettävyyssuunnittelutiimi (jonka vastuulla myös käytettävyydentestaus oli) halusi varmistaa, että tuotteet kehitetään jatkossakin asiakkaan tarpeiden mukaisiksi. Käytettävyyssuunnittelutiimi kehitti vuosien varrella omaa toimintatapaansa paremmin ketterään ohjelmistokehitykseen sopivaksi, ja tulokset olivat yllättävänkin hyviä. (Sy, 2007, s. 129.)

Ketterien menetelmien käyttöönoton myötä suurempi osa tuotteesta saatiin suunniteltua, käytettävyyssarviointi/-testaus saatiin kattamaan koko tuotteen julkaisuelinkaari (testaus ensimmäisestä suunnittelusprintistä viimeiseen toteutussprinttiin asti), tärkeimpiin osioihin pystyttiin keskittymään ensimmäisenä, käytettävyydentestauksen tulosten pohjalta tuotetta voitiin muuttaa nopeasti seuraavaan iteraatioon, vaatimuksia (engl. requirements) voidaan parantaa ja tarkentaa iteratiivisesti ja kehittäjät pysyvät paremmin suunnitelmissa. (Sy, 2007, s. 129–130.)

Varsinaiset havainnot ja käytännön ohjeet, joita Sy (2007, s. 130–131) esittää, ovat seuraavat:

1. Ketterän kehityksen kommunikointitapa pienentää kuilua käytettävyyssdatan keräämisen ja sen perusteella muutosten tekemisen välillä.
2. Koska ketterä kehitys on käyttäjäkeskeistä, kehitystiimi voi nojautua käyttäjän (tai asiakkaan) mielipiteisiin, vaikka oikeampi toimintatapa olisi perustaa päätökset käyttäjän toiminnan seuraamiseen. Näissä tilanteissa käytettävyydentestaajat voivat helpottaa päätöksentekoa.
3. Tavallisia käytettävyydentestaus tapoja voi yleensä käyttää myös ketterässä ohjelmistokehityksessä testaustapoina, mutta niiden ajoitusta ja tarkasteluyksikön kokoa voi joutua muuttamaan.
4. Aivan kuten ketterä ohjelmistokehitys on luonteeltaan iteratiivista, voi käytettävyyssuunnitteluprosessi olla myös iteratiivinen ja koostua inkrementaalisista minijulkaisuista. Suunnitteluprosessi voi lisäksi olla toteutusprosessia edellä esimerkiksi yhden sprintin verran.
5. Prototyyppien esittelyt ja päivittäiset keskustelut ovat paljolti korvanneet yksityiskohtaisemmat dokumentit kuten käytettävyydentestiraportit ja käyttöliittymämäärittelyt

4.2.6 Skenaarioperusteisen suunnittelun ja XP:n yhdistäminen

Lee ja McCrickard (2007) tutkivat tapoja käytettävyyden huomioimiseen XP-menetelmässä, skenaarioperusteista suunnittelua (engl. Scenario-Based Design, SBD) hyödyntämällä. He havaitsivat, että SBD- ja XP-menetelmillä on paljon yhteistä: iteratiivinen kehitys, ihmiskeskeisyys ja tiimin koordinointi ja kommunikaatio (Lee & McCrickard, 2007, s. 3). Suunnittelijoiden ja kehittäjien välisessä kommunikaatiossa käytettiin apuna keskitettyä suunnittelupöytäkirjaa (engl. Central Design Record, CDR) ja pyrittiin säilyttämään XP-menetelmän piirteet.

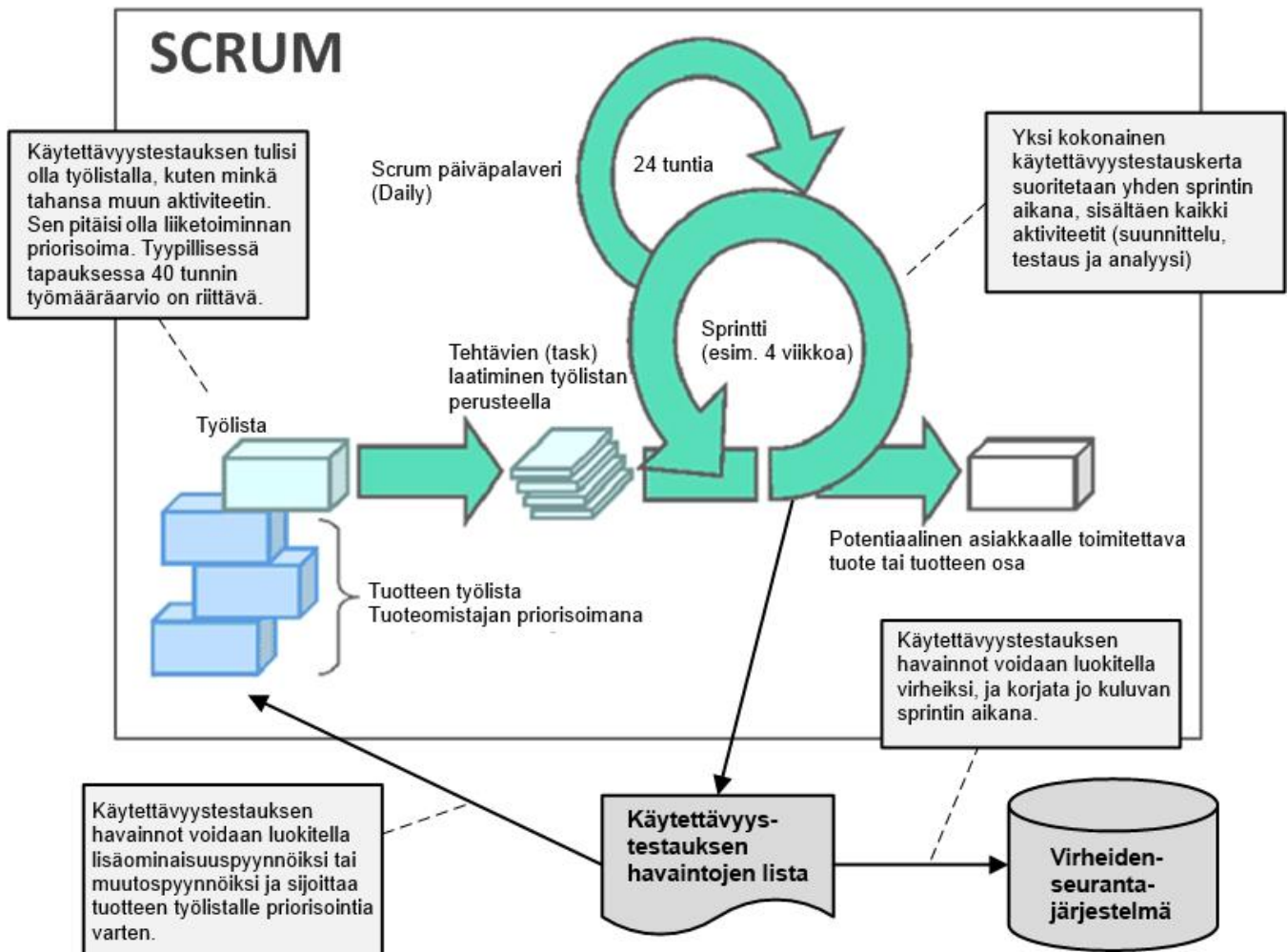
Tutkimuksessa havaintoja kokeiltiin useissa pienissä ohjelmistokehitystie-
meissä. Havaittiin, että projektin edetessä CDR-menetelmän noudattaminen vä-
hitellen unohtui ja tiimi keskittyi XP:n mukaiseen toimivan prototyypin toteutta-
miseen. Tämä aiheutti käytettävyyssarvioinnin tulosten huomiotta jättämisen ja
johti jatkuviin käytettävyyssongelmiin, jotka jäivät myös viimeisiin toteutet-
tuihin prototyyppisiin (Lee & McCrickard, 2007, s. 3). Tutkijat päättelivät, että
CDR-menetelmän noudattaminen vähitellen unohtui erityisesti siksi, että käytet-
tävyyssarvioinnin tulosten ja järjestelmän näkymien välillä ei ollut riittävää yhe-
teenkuuluvuutta. Heidän mukaansa olennaista prosessin toimivuuden kannalta
on saada käytettävyydestauksen tai -arvioinnin ja ohjelmistokehityksen välille
yhteys, joka tukee käytettävyydestauksen tai -arvioinnin tulosten perusteella jär-
jestelmän iteratiivista parantamista. Tutkijat myös ehdottavat, että tehtävän käy-
tettävyydestauksen määrän, ja siten myös kulujen, karkaaminen käsistä pyritään
estämään priorisoimalla käytettävyydestausta vaativat elementit tai näkymät
kunkin iteraation osalta keskitetyn suunnittelupöytäkirjan mukaisten väittämien
perusteella tai vaihtoehtoisesti tekemällä hyvin kohdennettua käytettävyydes-
tausta tai -arviointia kevyillä käytettävyydestausmenetelmillä (engl. discount
usability methods). (Lee & McCrickard, 2007, s. 3–4.)

4.2.7 Käytettävyydestaus scrumin osana

Blogikirjoituksissaan Northrop (2011, 2012) käsittelee kokemuksiaan käytettä-
vyydestaudesta ketterissä projekteissa, joissa hän on työskennellyt Summa-tech-
-nimisessä konsultointipalveluita tarjoavassa yrityksessä. Hän käsittelee asiaa
erityisesti projektinhallinnallisesta näkökulmasta. Hänen mukaansa käytettä-
vyyden varmistamiseksi tarvitaan erityisiä toimia, kuten varsinaista käytettä-
vyydestausta. Silloin kun (1) järjestelmän käyttäjät ovat laaja, ei-teknisesti suun-
tautunut ryhmä tai he tulevat asiakkaan organisaation ulkopuolelta, (2) käyttä-
jien tekemien virheiden kustannus on korkea tai (3) erilaisia vaihtoehtoja käyttö-
liittymätoteutuksille on paljon eivätkä alustan parhaat käytännöt ole vielä va-
kiintuneet.

Northrop (2011) kuvaa tavan sovittaa yhteen käytettävyydestaus scrumin
kanssa tavalla, joka on hänen mielestään helppo, nopea ja halpa. Kuviossa 5 ku-
vataan käytettävyydestauksen aktiviteettien nivoutuminen scrumin iteratiivi-
seen työtapaan. Mallissa jokaisen sprintin aikana suoritetaan yksi käytettävyydes-
tauskerta – siis testataan yhdellä tai useammalla koehenkilöllä järjestelmän

käytettävyyttä. Testauskerran havainnoista muodostetaan lista, jonka kohteet liitetään työlialle joko virheinä tai lisäominaisuustoiveina. Testauksen tulisi olla omana työlialan kohteenaan siinä missä muidenkin iteraation aikaisten tehtävien, mutta olennaisena erona esimerkiksi käyttäjätarinoihin käytettävyydestaus ei käytännössä valmistu lainkaan. Se on ikään kuin tehtävä, jonka priorisointi tulee tehdä joka sprintillä uudestaan. Siten voidaan ajatella myös käytettävyydestauskulkevan mukana työlialalla koko projektin ajan (Northrop, 2011).



KUVIO 5 Scrum ja käytettävyydestaus Summa-techin toteutustavassa (Northrop, 2011)

4.2.8 UTUM-käytettävyydestausratkaisu

Winter, Rönkkö, Ahlberg ja Hotchkiss (2011) esittelevät tapaustutkimuksessaan UTUM-käytettävyydestausratkaisun, jolla he pyrkivät luomaan kompromissin ketterän kehityksen ja käytettävyydestausmenetelmien välille. Heidän mukaansa UTUM on hybridiratkaisu ketteryyden ja muodollisuuden (engl. formalism) välillä, ja täyttää sekä organisaation tarpeet muodolliselle käytettävyydestaukselle että ketteryydelle. Heidän mukaansa ketterien menetelmien yleisosajat ovat ihanne, joka ei tosielämässä toteudu, ja erityisesti testaajat ja kehittäjät

ovat erillisiä rooleja (Winter ym., 2011, s. 278). Siksi UTUM tarjoaa työkalut siihen, että eri rooleissa työskentelevät projektiryhmien jäsenet voivat työskennellä yhdessä. UTUM itsessään sai alkunsa tilanteesta, jossa asiakkaan edustajia ei ole käytettävissä validoimaan toteutusta, mutta esimerkiksi hyväksymistestaukseen halutaan loppukäyttäjän näkemys.

UTUM-testauksessa testauspäällikkö (engl. test leader) johtaa testausta, jossa hyödynnetään yhtä koehenkilöä kerrallaan, mieluiten mahdollisimman neutraalissa ympäristössä, jotta koehenkilö tuntee olonsa mukavaksi. Testauksen aikana koehenkilö suorittaa testattavalla laitteella erilaisia käyttötapauksia testauspäällikön seurattessa suoritusta ja tehdessä muistiinpanoja. Koehenkilöä pyydetään lopuksi täyttämään SUS-mittaristoon⁶ perustuva kysely laitteen käyttöliittymän käytettävyydestä. (Winter ym., 2011.)

Koehenkilöä kiitetään ja hänelle annetaan jonkinlainen palkinto testauksen osallistumisesta, esimerkiksi elokuvalippu. Testaussession jälkeen testauspäällikkö koostaa muistiinpanoista käytettävyydestestauksen havainnot ja halutut metriikat. Käytännössä tulosten koostamiseen käytetään esimerkiksi laskentataulukkoa, ja numeerisesta datasta koostetaan myös kuvioita, ja kaikesta tästä datasta voidaan koostaa raportti. Laajassa organisaatiossa hyväksyntää käytettävyydestestauksen tuloksille ei tutkijoiden mukaan voida saada ilman muodollista raportointia. (Winter ym., 2011.)

Winterin ym. (2011) lähestymistapa kompromissiin muodollisuuden ja ketteryyden välillä on lähtökohtaisesti tehdä ketterää kehitystä, mutta tuoda mukaan muodollisuutta vain sen verran, kuin organisaation vaatimusten täyttämiseksi on pakollista. Tämä vaikuttaa esimerkiksi käytettävyydestestauksen tulosten raportointitapaan.

4.2.9 Synteesi ja johtopäätöksiä

Käytettävyydestestauksen ja ketterän ohjelmistokehityksen yhteensovittamista on yritetty useasti, ja monilla eri tavoilla. Usein kevyisiin käytettävyydestestauksen menetelmiin on päädytty joko eksplisiittisesti, jolloin ne ovat saattaneet olla lähtökohtana yhteensovittamista suunniteltaessa, tai implisiittisesti, jolloin niihin on päädytty toimivampana käytettävyydestestauksen menetelmäkokoelmana. Tulokset ovat olleet rohkaisevia, mutta eivät pelkästään positiivisia.

Ketterästä testauksesta yleensä on sanottu, että sen tavoitteena on yksinkertaisimman mahdollisen tavan löytäminen sen varmistamiseksi, että asiakkaalle toimitettava ohjelmisto täyttää tämän vaatimukset (Crispin & Gregory, 2009, s. 26). Käytettävyydestestauksen ja ketterän ohjelmistokehityksen yhteensovittamisyrityksissä on usein keskitytty joko ketterän kehityksen ”hengen” mukaisesti keveisiin toimintatapoihin (esimerkiksi McGinn & Chang, 2013) tai sitten hyvinkin tunnolliseen menetelmien integrointiin. Yhteistä useimmille yrityksille on ollut

⁶ SUS (engl. System Usability Scale, järjestelmän käytettävyydsmittaristo) on Brooken (1996) kehittämä mittaristo järjestelmän käytettävyyden mittaamiseen yksinkertaisia valmiita kysymyksiä käyttämällä.

käytettävyydestestauksen aikataulujen sovittaminen ketterän kehityksen iteraatioihin, esimerkiksi siis scrumin sprintteihin. Joskus tämä prosessi on aloitettu yksinkertaisesti tiivistämällä ja pilkkomalla vesiputousmallin mukaisesti tehtävä projekti esimerkiksi 90-päiväisiin sykleihin (Abdelnour-Nocera & Sharp, 2008).

Ketterällä ohjelmistokehitystavalla ja kevyellä käytettävyydestestauksella on paljon yhteistä. Hyvin yhteen sopiviksi piirteiksi tai käytänteiksi on havaittu iteraatiivinen ja inkrementaalinen toimintatapa, ketterän ohjelmistokehitystavan painottama jatkuva kommunikaatio sekä tiimin sisällä että asiakkaan suuntaan, yksinkertaisuuden ihannoiti suunnittelussa ja toteutuksessa (Lee ym., 2009; Lee & McCrickard, 2007), tulosten nopea toimittaminen ja prototyyppien aktiivinen hyödyntäminen.

Ketterän ohjelmistokehityksen ja käytettävyydestestauksen yhteensovittamisen ongelmakohtia on ratkottu useilla eri tavoilla. Tärkeää on ainakin saada välitettyä ristiriidat kehitysryhmän sisällä (esimerkiksi käytettävyyssasiantuntijoiden ja kehittäjien välillä), jotta resursseista ei kilpailla turhaan ja projektitiimin tavoitteet ovat yhtenevät (Lee ym., 2009). Projektitiimin lisäksi sidosryhmät täytyy saada vakuutettua käytettävyyden tärkeydestä – tähän kenties toimivin keino on käytettävyydestestaussessioiden seuraamisen mahdollistaminen, tai ainakin tallenteiden tekeminen ja esittely tarvittaessa (McGinn & Chang, 2013), sekä organisaation yleiset muodollisuusvaatimukset täyttävä käytettävyydestestauksen havaintojen raportointi (Winter ym., 2011). Tätä ongelmaa helpottaa ketterän ohjelmistokehityksen suosima ja ketterään käytettävyydestestaukseen hyvin sopiva aktiivinen ja epäformaali jatkuva kommunikointi.

Sopiva kompromissi koehenkilöiden määrän ja tyyppien suhteen on suuri kysymys. Käytettävyydestestaussessioita suositellaan yleensä järjestettäväksi mieluummin usein ja vähäisemmällä osallistujamäärällä kuin harvoin ja suuremmalla otannalla (McGinn & Chang, 2013). Koehenkilöiden kohderyhmävastavuudesta täytyy kantaa ainakin jonkin verran huolta, sillä pienellä otannalla ja äärimmäisen vääränlaisella koehenkilöllä saadaan helposti todella vinoutuneita tuloksia – toisaalta on myös painotettu testaaajan arviointikykyä tuloksia tulkittaessa. Lisäksi kysymys siitä, missä määrin tuoteomistaja tai asiakkaan edustaja voi toimia loppukäyttäjän edustajana, jää osittain auki (Sy, 2007).

Ketterässä ohjelmistokehityksessä tiimi koostuu moniosaajista, mutta useimmissa yrityksissä (mm. Lee ym., 2009; Lee & McCrickard, 2007; Singh, 2008) sovittaa käytettävyydestestaus osaksi sitä tiimissä on mukana käytettävyyteen keskittyvä henkilö, joka on vastuussa ainakin käytettävyydestestauksen järjestämisestä. Joissain tutkimuksissa kuitenkin testaaajan ja kehittäjän rooli tunnustetaan erilaiseksi ja erilliseksi (esim. Winter ym., 2011). Vastuu voi myös olla projektiryhmän sisällä kiertävällä roolilla, tai sitä varten voidaan hyödyntää alihankkijoita. Useissa tapauksissa huomiota on kiinnitetty myös tulosten erittäin nopeaan raportointiin ja pohdittu sopivaa määrää tuotettavalle dokumentaatiolle. Toisaalta on nähty ensiarvoisen tärkeäksi toimittaa organisaation muodolliset vaatimukset täyttävä raportti käytettävyydestestauksen havainnoista, ja siten myös nähty arvokkaaksi käyttää aikaa raportin koostamiseen (Winter ym., 2011).

Kaikista olennaisimpana käytettävyydestäuksen menetelmänä myös ketterän ohjelmistokehityksen yhteydessä pidetään usein perinteistä, koehenkilöitä hyödyntävää käytettävyydestäusta (Lee ym., 2009; McGinn & Chang, 2013; Winter ym., 2011). Sen hyödyntäminen näyttää ensiarvoisen tärkeältä, mikäli riittävät resurssit ovat käytettävissä.

4.3 Käytettävyydmetriikoiden hyödyntäminen

Käytettävyydmetriikat voivat osaltaan antaa hyvin konkreettisen kuvan järjestelmän käytettävyydestä ja tukea tai kritisoida koehenkilöiden tarkasteluun perustuvan testauksen tuloksia. Erityisesti metriikoiden avulla on myös mahdollista seurata järjestelmän kehitystä eri testauskertojen välillä.

Metriikoiden käyttäminen vain siksi, että saataisiin jotain numeerista dataa, ei ole mielekästä, vaan se voi jopa olla haitallista. Metriikoiden tehtävä on auttaa projektitiimiä, joten ne on valittava oikein, ja ketterässä ohjelmistokehityksessä niiden on syytä olla yksinkertaisia (Crispin & Gregory, 2009, s. 437). Lean-menetelmässä (Poppendieck & Poppendieck, 2003) suositellaan sen mittaamista, mikä on esimerkiksi muutosten tuotto prosentti (engl. Return On Investment, ROI) tai kuinka nopeasti virheet saadaan korjattua, sen sijasta että mitattaisiin esimerkiksi absoluuttista virheiden määrää, joka voi olla merkki huonolaatuisesta koodista, hitaasta testauksesta, kattavasta testauksesta tai erittäin aktiivisesta asiakkaasta, tai vaikkapa työmääräarvioiden paikkansapitävyyttä, joka taasen voi olla erittäin raskas mitattava ja tuottaa hyödyttömiä dataa. (Crispin & Gregory, 2009, s. 72–74,76.)

Scott (2009, s. 10) toteaa, että klassinen käytettävyydestäus, kuten heuristinen arviointi, laboratoriossa testaus ja laskennalliset mallit mittaavat paljolti vanhentunutta tai alkeellista dataa, kuten käytön tehokkuutta tai tarkkuutta, modernimpien mittareiden, kuten esimerkiksi vaikuttavuuden, lojaliteetin tai osallistavuuden, sijasta.

Nielsenin (1993) mukaan käytettävyydmetriikoiden käyttökelpoisuutta laskee se, että testattavien koehenkilöiden määrän täytyy olla verraten korkea, jotta dataa saadaan mitattua tarpeeksi riittävän luottamusvälin aikaan saamiseksi (Nielsen, 2001). Mikäli lukuja halutaan kuitenkin kerätä, mahdollisia mielekkäitä kvantitatiivisesti käytettävyyttä mittaavia suureita voivat olla ainakin seuraavat (Nielsen, 1993, s. 194–195):

- Aika, joka koehenkilöiltä kuluu tietyn tehtävän suorittamiseen
- Tehtävien (tai tehtävän osioiden) määrä, joka koehenkilöiltä onnistuu tietyssä ajassa
- Onnistuneiden ja epäonnistuneiden testitapausten suhde.
- Aika, joka kului ongelmista palautumiseen
- Toisiaan välittömästi seuranneiden ongelmien määrä

- Koehenkilön hyödyntämien eri komentojen tai toimintojen määrä (joko absoluuttinen määrä, tai sitten erilaisten toimintojen tai komentojen määrä)
- Kokonaan käyttämättä jääneiden toimintojen tai komentojen määrä
- Niiden toiminnallisuuksien määrä, jotka käyttäjä muistaa session jälkeen
- Frekvenssi, jolla koehenkilö turvautuu ohjeisiin, ja aika joka kului ohjeita tarkastellessa
- Frekvenssi, jolla ohjeet ratkaisivat koehenkilön ongelmat
- Koehenkilön positiivisten ja negatiivisten toteamusten suhde session aikana
- Koehenkilön selkeiden ilon tai turhautumisen ilmausten määrät
- Se osuus koehenkilöistä, jotka arvelisivat käyttävänsä kokeiltua järjestelmää mieluummin kuin sen kilpailijaa
- Niiden kertojen määrä, kun koehenkilö törmäsi ylitsepääsemättömään ongelmaan
- Se osuus koehenkilöistä, jotka löysivät tehokkaat työtavat, verrattuna niihin käyttäjiin, jotka käyttivät tehottomia tapoja (käyttökel-poinen vain, jos on useita tapoja toteuttaa tehtävä)
- "Kuolleen" ajan määrä, eli aika jolloin koehenkilö odottaa järjestelmää tai järjestelmä koehenkilöä, esimerkiksi järjestelmän hitauden tai koehenkilön pohdinnan vuoksi
- Kerrat, jolloin koehenkilö joutuu sivuraiteille kesken tehtävänsä.

Toisaalta esimerkiksi ääneen ajattelu menetelmänä vaikeuttaa luotettavien tietojen keräämistä ja painottaa enemmänkin koehenkilön ajatusten tulkintaa (Nielsen, 1993, s. 195–196). Nielsen (2001) esittelee myöhemmin paljon yksinkertaisemman mittariston käytettävyyden, tai erityisesti käytettävyyden muutosta iteraatioiden välillä, mittaamista varten:

- onnistumisprosentti
- tehtävään kuluva aika
- epäonnistumisten määrä
- käyttäjien subjektiivinen tyytyväisyys.

Nielseninkin mukaan käytettävyyden eteen tehtävän työn paras puolestapuhuja on ROI: vaikka luku on viime vuosien aikana hiukan laskenutkin, on ROI yhä korkea. Nielsen (2008) arvioi, että kaikista "matalimmalla roikkuvat hedelmät" on nyt saatu napattua ja kaikista selkeimmät käytettävyysspainajaiset saatu korjattua. Hän kuitenkin huomauttaa, että vieläkin käytettävyys on mainio sijoitus. Holzinger ym. (2005) listaavat testaussessioiden luonnollisiksi mittareiksi sen, kuinka kauan tehtävän suorittamiseen kestää (engl. time to perform a task) ja suorituksessa ilmenevät virheet (engl. cognitive overload).

Vaikka joissain tutkimuksissa pidettiin käytettävyyshetimitriikoita tarpeellisina (de Kock ym., 2009; Genov, 2005; Lee ym., 2009; McInerney & Maurer, 2005 (UCD); Preece, 1994; Winter ym., 2011), toisissa tapauksissa (Genov, 2005 (joissain tapauksissa); Göransson, Gulliksen, ym., 2003; Marty & Twidale, 2005; Nielsen, 2001; Scott, 2009) niitä voitiin pitää tarpeettomina tai jopa haitallisina ja anekdoottisia havaintoja, joiden keräämiseen käytetyt menetelmät pääosin keskittyvät, niitä merkittävämpinä. Kaiken kaikkiaan voitaneen sanoa, että koska luotettavien käytettävyyshetmitriikoiden kerääminen on todennäköisesti työlästä ja siten kevyeen käytettävyyshetmitestaukseen kehnosti yhteensopivaa, ei niiden keräämiseen kannattane liikaa keskittyä.

4.4 Testauksen ajoitus

Tapaustutkimuksissa on havaittu, että käytettävyyshetmitestaus on syytä tehdä mahdollisimman aikaisessa vaiheessa eikä esimerkiksi melkein valmiille tuotteelle, kuten ennen ensimmäistä julkaisua (Meszaros & Aston, 2006, s. 292; Rubin, 2008, s. 331). Ensimmäinen käytettävyyshetmitesti kannattanee suorittaa jo ns. ”nollasprintissä” (Sy, 2007, s. 118), mikäli suinkin mahdollista. Myös interaktiosuunnittelun (engl. interaction design) ja XP:n yhdistämistä tutkittaessa on ehdotettu käyttäjän ottamista mukaan interaktiosuunnitteluun alusta asti ja sitten palautteen perusteella käyttäjätarinoiden parantamista iteratiivisesti (Aigner, 2002, s. 7–8).

Aikaisessa vaiheessa voi olla hyödyllistä kerätä palautetta esimerkiksi karkeista prototyypeistä sisäisiltä koehenkilöiltä, kuten projektitiimin jäseniltä. Tämä on nopea ja vaivaton tapa kerätä palautetta jotain tiettyä toiminnallisuutta koskien, ja vieläpä ilman varsinaisten sessioiden järjestämistä ja siitä aiheutuvaa vaivaa. Sisäiseltä koehenkilöltä voi saada palautteen parissa minuutissa ilman byrokratiaa (Sy, 2007, s. 121). Sisäisten koehenkilöiden lisäksi tuote tai järjestelmä kannattaa kuitenkin näyttää myös asiakkaalle mahdollisimman aikaisessa vaiheessa, jopa niin aikaisessa, että jotkut kuten Crispin ja Gregory (2009, s. 105), kannattavat sen sisällyttämistä osaksi testaussuunnitelman tehtäväkortteja (muuta tällaisia kortteja ovat muun muassa integraatiotestauskortit ja yksikkötestauskortit).

Testattaessa aikaisin testauksen kohde on luultavasti jotain aivan muuta kuin valmis tuote. Todennäköisemmin testattavana on aikainen prototyyppi, tai ehkä vasta suunnitelmia tai luonnoksia. Esimerkiksi Khan (1997, s. 320) esittää kehityksen aloittamista paperiprototyypeillä, ja myös ainakin ensimmäiset testit sisältäisivät siis toimivan järjestelmän sijasta paperiprototyyppejä. Joka tapauksessa testattava tuote pitää tuntea, joten mitä tahansa testataankaan, täytyy testaajalla olla aikaa tutustua testattaviin materiaaleihin huolellisesti ennen testejä. (Rubin, 2008, s. 181–182.)

Davis ja Venkatesh (2004, s. 32) ehdottavat hyväksyntätestauksen aloittamista jo ennen kuin ensimmäistäkään koodiriviä on kirjoitettu, jolloin ensimmäi-

nen hyväksyntätestaus tehtäisiinkin vaatimusmäärittelylle. Osana tätä he esittivät, että vaikka käytettävyyden testaaminen, ennen kuin on mitään prototyyppiä testattavana, ei ole mahdollista, hyödyllisyys ennustaa hyvin käyttäjien tavoitteiden toteutumista, ja täten hyödyllisyyden testaaminen vaatimusmäärittelyn hyväksyntätestauksen avulla osittain korvaa aikaisen vaiheen käytettävyydestausta. Toisaalta heidän tutkimuksensa antoi myös viitteitä siitä, että aikaisen vaiheen käytettävyydestausta paperiprototyypeillä tai yksinkertaisten mallien avulla ennusti hyvin myöhempiä tuloksia (Davis & Venkatesh, 2004, s. 43).

Käytettävyydestaustan tulosten nopea esittely projektiryhmälle voi saada projektitiimin sitoutumaan käytettävyyteen ja parantamaan kommunikaatiota. Joissain tapauksissa vapaaehtoinen, päiväpalaverin jälkeen järjestettävä käytettävyysspalaveri on kerännyt säännöllisesti pääosan projektitiimistä kokoon. Näissä palavereissa voi käsitellä esimerkiksi sessioiden parhaat menestystarinat, kontekstuaalista tietoa kuten yllättävät käyttötapaukset, toivomukset uusille ominaisuuksille, merkittävimmät käytettävyyssongelmat ja selkeät virheet. Esityksen jälkeen projektiryhmä saa päättää suhtautumisestaan ominaisuustoiveisiin (engl. feature request) ja yllättäviin käyttötapauksiin. Virheet voidaan viedä suoraan sprintin tehtävälisään. (Sy, 2007, s. 127.)

Osallistumisesta käytettävyydestaustassessioiden seuraamiseen on myös saatu rohkaisevaa palautetta. RITE+Krug- menetelmässä eri sidosryhmät velvoitettiin lähettämään yksi päätösvaltainen edustaja seuraamaan sähköisesti käytettävyydestaustassessioita, jotta sidosryhmät saataisiin sitoutettua käytettävyyden huomiointiin. Tämä toimi, kun edustajat olivat ensimmäisissä sessioissa todistaneet koehenkilöiden hankaluuksia, ja ei kestänyt kauaakaan, kun parhaimmillaan koko projektiryhmä seurasi sessiota sähköisesti. (McGinn & Chang, 2013.)

On huomionarvoista, että käytettävyydestaustauksessa voi paljastua muitakin kuin käytettävyyteen liittyviä ongelmia. Eräässä tutkimuksessa havaittiin yhteensä 102 ongelmaa tuotteessa, mutta näistä vain 71 oli käytettävyyssongelmia: loput olivat virheitä, ominaisuustoiveita, prototyypin rajoitteita tai muita ongelmia. Kun luvuista poistetaan toisteiset ongelmat (eri testiryhmien raportoimat samat käytettävyyssongelmat), oli muiden kuin käytettävyyssongelmien lukumäärä melkein yhtä suuri kuin käytettävyyssongelmien. (Kessner, Wood, Dillon, & West, 2001, s. 19.)

Käytettävyydestaustan palautteen nopea käsittely on myös hyvin linjassa Agile-manifestissa (Fowler & Highsmith, 2001) mainitun nopean muutokseen reagoinnin kanssa. Riippumatta siitä ovatko koehenkilöt varsinaisesti asiakkaan edustajia vai eivät, palautteen saaminen loppukäyttäjältä, tai loppukäyttäjää vastaavalta koehenkilöltä, voi aiheuttaa suuriakin muutoksia kehitystyöhön, jolloin tavoitteena on tietysti loppukäyttäjän käyttäjäkokemuksen parantaminen.

4.5 Testauksen määrä

Tässä alaluvussa käsitellään aiemmassa tutkimuksessa esiteltyjä näkemyksiä testauksen optimaalisimmasta määrästä. Tähän liittyy olennaisesti koehenkilöiden määrä ja tyyppi.

4.5.1 Koehenkilöiden määrä

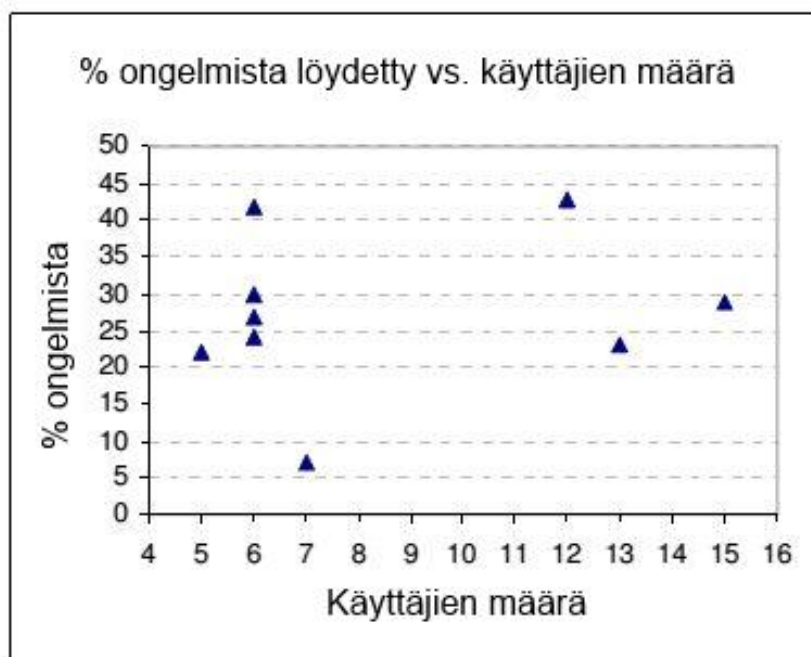
Perinteisessä käytettävyydestestauksessa koehenkilöitä tarvitaan usein paljon, jotta saadaan kvantitatiivisia tuloksia, mutta kevyessä käytettävyydestestauksessa heitä tarvitaan vähemmän. Toteutettaessa kevyt käytettävyydestaus osana ketterää ohjelmistokehitystä joudutaan miettimään jopa pienintä mahdollista testauksen määrää, jolla saadaan yleensä ottaen mitään tuloksia (Marty & Twidale, 2005, s. 4-5). Cooperin (2004) mukaan testaus voidaan tehdä jopa ilman koehenkilöitä (Blomquist & Arvola, 2002, s. 1).

Alan tutkimus on pääosin keskittynyt kehittämään parempia tapoja testata sen sijaan, että pyrittäisiin kehittämään testaustapoja mahdollisimman tehokkaiksi (Martin, Rooksby, Rouncefield, & Sommerville, 2007, s. 9). Useimmille organisaatioille mahdollisimman tehokas ja järkevä tapa käyttää rajalliset testausresurssit olisi kuitenkin kenties absoluuttisia parhaita testauskäytänteitä ja -työkaluja hyödyllisempi tutkimuskohde. Toisaalta on myös argumentoitu, että tiedemaailman liiallinen keskittyminen optimaalisen (siis esimerkiksi 85 % kattavuuden tarjoavan) koehenkilömäärän etsintään on ollut haitallista, sillä merkittävämpää olisi ehkä ollut kehittää mahdollisimman tehokkaita tapoja käyttää rajalliset testausresurssit (Martin ym., 2007, s. 9) tai, kuten Marty ja Twidale (2005, s. 4) argumentoivat, edes etsiä pienintä mahdollista koehenkilöiden määrää, jolla saadaan varmasti aikaiseksi tuloksia. Wixon (2003, s. 31) myös huomauttaa, että keskittyminen vain siihen, että optimaalinen määrä ongelmia saadaan havaittua, ei riitä, sillä havaitut ongelmat pitäisi myös saada korjattua. Hänen mukaansa pahimmillaan keskittyminen optimaalisen koehenkilömäärän tavoitteluun aiheuttaa resurssien keskittymisen väärin aktiviteetteihin.

Testauksen ”oikea määrä” määräytyy aina organisaation ja/tai asiakkaiden tarpeiden mukaan. Useinkaan testausresurssseja ei ole tuhlattavaksi asti, ja siksi täytyykin keskittyä vain olennaisimpiin asioihin tai vain tärkeimmän asiakkaan toiveiden mukaiseen testaukseen (Martin ym., 2007, s. 9). Vähäininkin määrä käytettävyydestestausta on kuitenkin enemmän kuin ei lainkaan (Gray & Salzman, 1998, s. 207; Krug, 2006), ja on todennäköisesti hyödyllistä kunhan koehenkilö ei ole vääränlainen.

Optimaalinen määrä koehenkilöitä riippuu monista seikoista. Nielsen (1994b, s. 5, 2000) on arvioinut 3-6 koehenkilöä koesessiota kohden antavan parhaan tuoton investoinnille. Virzi (1992) on päätenyt viiteen koehenkilöön. Hertzumin ja Jacobsenin (2001) mukaan riittäväksi määräksi on yleensä arvioitu 4-5 käyttäjää, ja Jones (2003) mainitsee pyrkivänsä 3-5 koehenkilöön iteraatiota kohden.

Toisaalta on myös havaittu, ettei koehenkilöiden määrällä ja vakavien käytettävyysongelmien ilmi tulemisella testausseisioissa ole korrelaatiota (Lindgaard & Chattratichart, 2007, s. 7), mikä voidaan havaita kuvioista 6. Kuvioista nähdään, että kyseisessä tutkimuksessa käyttäjien määrää lisäämällä ei voitu varmistaa ongelmien kattavaa löytymistä. Parhaat tulokset, joissa molemmissa löydettiin 40–45 % ongelmista, saatiin kuudella ja kahdellatoista koehenkilöllä. Sen sijaan korrelaatio on verraten vahva ongelmien löytymistodennäköisyyden ja testitapausten kattavuuden välillä. Toiset tutkimukset ovat osoittaneet, että koehenkilöiden määrän lisääminen parantaa tulosten luotettavuutta: siinä missä viidellä koehenkilöllä löytyi keskimäärin 85 % ongelmista (vaihteluväli 55–100 %), jo kymmenellä koehenkilöllä löytyi 95 % ja vaihteluväli oli kutistunut ollen n. 82–100 % (Faulkner, 2003, s. 382). Tätä havainnollistaa kuvio 7. Tyypillisesti vakavimmat ongelmat löytyvät jo hyvin pienellä määrällä koehenkilöitä (Virzi, 1992), mutta jotta mahdollisimman suuri osa järjestelmästä saataisiin testattua, on iteraatioita syytä olla useita, ja oppimisvaikutuksen välttämiseksi samoja koehenkilöitä ei tule käyttää useasti. Merkittävä osa suuresta vaihtelusta löydettyjen ongelmien määrässä, mikä esiintyy esimerkiksi Faulknerin (2003) tutkimuksessa, johtuu testajaista ja heidän erilaisesta tulkinnastaan tulosten vakavuuden suhteen (Hertzum & Jacobsen, 2001, s. 12).

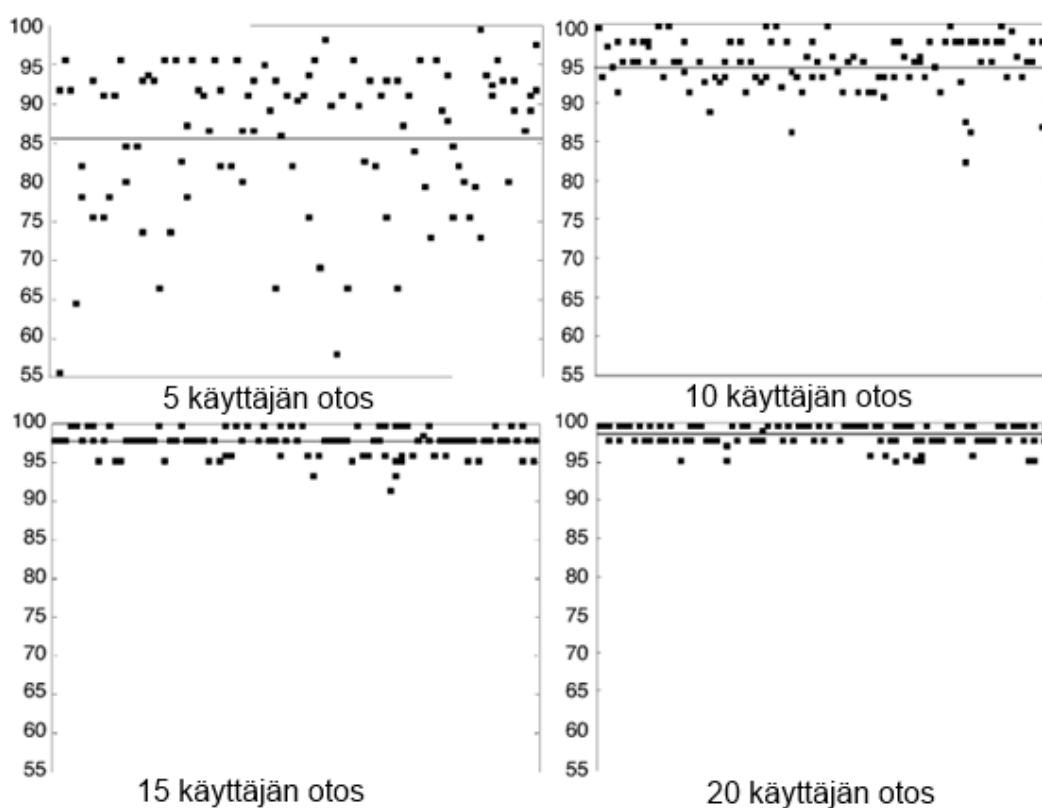


KUVIO 6 Ongelmien löytymisen ja koehenkilöiden määrän suhde (Lindgaard & Chattratichart, 2007, s. 1421)

Vähäinen määrä testausta, erityisesti pienellä koehenkilöiden määrällä, ei riitä todistamaan mitään trendeistä eikä tuota tuloksia, jotka olisivat luotettavasti yleistettävissä koko populaatioon. Silti on parempi testata kuin olla testaamatta. Tuloksia täytyy vain kohdella sellaisina kuin ne ovat. Tällaisella testauksella ei myöskään voida varsinaisesti validoida tuotetta, siis varmistua siitä että se on

virheetön, vaan lähinnä voidaan havaita siinä olevia virheitä ja sitten pyrkiä korjaamaan niitä. (Dicks, 2002)

Toisaalta on myös huomionarvoista, että siinä missä moni tutkija (mm. Nielsen, 1994b, 2000; Northrop, 2012; Virzi, 1992) kuvaa useilla (kuten yli viidellä) koehenkilöillä testaamista laskevien rajahyötyjen (engl. diminishing returns) takia tarpeettomaksi, sillä jokainen uusi koehenkilö tuottaa suurella todennäköisyydellä pääosin samoja havaintoja kuin aiemmat, on tällä toisaalta aiempia havaintoja validoiva vaikutus (Faulkner, 2003, s. 4). Kunkin havainnon uskottavuus ja hyväksyttävyyys projektiryhmän parissa kasvaa, kun voidaan osoittaa saman ongelman ilmeneen useammalla koehenkilöllä. On myös huomattu, että erityisesti kevyiden käytettävyystestausmenetelmien piirteenä on suuri riippuvuus testaajan ammattitaidosta (Hertzum & Jacobsen, 2001, s. 12; Kessner ym., 2001, s. 33).



KUVIO 7 Koehenkilöiden määrän vaikutus ongelmien löytymiseen (Faulkner, 2003, s. 382)

4.5.2 Koehenkilöiden tyyppi

Ohjelmistokehittäjiltä tyypillisesti puuttuu kokemus käyttäjäkeskeisestä kehittämisestä, jolloin heidän on hankala arvioida, kauanko aikaa vaaditaan käytettävyydeltään hyvän järjestelmän. He sopivat siis huonosti koehenkilöiksi – paitsi tietenkin tapauksessa, jossa käyttäjät ovat tyypillisimmin ohjelmistokehittäjiä.

Organisaatiossa on usein hyvin vähäinen määrä käytettävyyssiantuntijoita, joten näiden hyödyntäminen useammassa projektissa soveltuvin osin – U-scrumin tapauksessa tuotteen käytettävyyden omistajina – tehostaa heidän resurssiensa käyttöä, kun tavanomaisemmat liiketoiminnasta vastaavat ihmiset voivat hoitaa tuoteomistajan (engl. product owner) työt miettimättä käytettävyyttä (Singh, 2008, s. 559).

Ketterissä menetelmissä, erityisesti XP:ssä, puhutaan toimittajan tiloissa sijaitsevasta (engl. on-premises) asiakkaasta. Sillä tarkoitetaan asiakkaan edustajaa, joka osallistuu muiden muassa testitapausten tai esimerkiksi käyttäjätarinoiden määrittelyyn. Houkutus tällaisen asiakkaan edustajan käyttämiseksi käytettävyydestestauksen primäärisenä koehenkilönä voi olla suuri, mutta niin ei välttämättä kannata toimia. Asiakkaan edustaja ei useinkaan edusta loppukäyttäjää (Jokela, 2010, s. 72), sillä projektin tai tuotteen tilannut asiakas usein itse asiassa tarjoaa tuotteen tietylle käyttäjäryhmälle, joka voi olla esimerkiksi sen asiakkaat tai tietty, erikoistunut osa sen työntekijöistä, tai esimerkiksi asiakkaan alihankkijat. Tällöin koehenkilöt kannattaa valikoida käyttäjätarinoiden tai persoonien perusteella asiakkaan edustajan hyödyntämisen sijasta.

4.6 Käytettävyydestestauksen hinta

Aiemmissa tutkimuksissa ei useinkaan ole erikseen mainittu käytettävyydestestauksen rahallista hintaa, eikä se välttämättä olisi aina mielekästäkään. Tietyn toimintamallin kustannukset riippuvat mallin ominaispiirteiden lisäksi ainakin valuutasta ja työn hinnasta organisaatiossa, ja tähän vaikuttavat organisaation piirteiden lisäksi muun muassa verotus ja palkan sivukulujen määrä sekä yleinen hintataso yhteiskunnassa, jossa mallia hyödynnetään. Seuraavassa annetaan muutamia esimerkkejä käytettävyydestestauksen hinnoista.

Krugin (2000) mukaan tieteellinen käytettävyydestestaus maksaa 20 000 – 50 000 dollaria, ja nykyaikainen käytettävyydestestaus yrityksissä noin 5 000 – 15 000 dollaria testauskertaa kohden. Krugin mukaan hänen oma mallinsa maksaa noin 300 dollaria testauskertaa kohden, mikä ei tosin sisällä mitään hintaa ajankäytölle. Tämän laskutavan mukaan hän mainitsee pitkän projektin kaiken käytettävyydestestauksen maksavan yhteensä esimerkiksi 3 900 dollaria.

Nielsen ja Landauer (1993) koostavat tutkimuksessaan aiempien tutkimusten pohjalta käytettävyydestestauksen hintaa pienissä, keskikokoisissa ja suurissa ohjelmistokehitysprojekteissa. Taulukossa 5 on esitetty heidän koosteensa eri koluokkia edustavissa projekteissa hyödynnettävien testausmenetelmien kustannuksista. Heidän mukaansa käytettävyydestestauksen hinta pienehköissäkin projekteissa on yli 16 000 dollaria, jos hyödynnetään koehenkilöitä, ja ilman koehenkilöitä mutta useilla arvioijilla tehtäessä heuristinen evaluointi maksaa yli

15 000 dollaria sekin. Lisäksi he laskevat aiempien tutkimusten pohjalta, että käytettävyydestä valmistelu maksaa noin 5 580 dollaria (3 400)⁷, yhdellä koehenkilöllä testaaminen maksaa noin 3 280 dollaria (2 000). Vastaavasti heuristisen evaluoinnin valmistelu maksaa 6070 (3 700) ja 7 880 (4 800) dollarin välillä, ja yhden uuden evaluoijan käyttö maksaa lisää 689 (420) ja 853 (520) dollarin välillä. Kevyen käytettävyydestä valmistelu maksaa 4 270 (2 600) dollaria valmistelun osalta ja 673 (410) dollaria jokaisen uuden koehenkilön osalta.

Toisessa tutkimuksessaan Nielsen (1994b) kuvaa kolmella koehenkilöllä tehdyn käytettävyydestä valmistelu maksaneen 7 440 (4 650) dollaria⁸. Bias ja Mayhew (2005) mainitsevat kirjassaan lukuisiakin erilaisia esimerkkejä erilaisista käytettävyyssuunnittelun kustannuksista, mutta käyttäjätestausta hyödyntäviä esimerkkejä ovat muun muassa käytettävyyden testaaminen konseptointivaiheessa, jonka kustannukseksi arvioitiin 30 800 (25 400) dollaria⁹ yksityiskohtainen käyttöliittymien testaus, jonka kustannukseksi 22 koehenkilöllä on arvioitu 39 300 (32 400) dollaria¹⁰.

TAULUKKO 5 Käytettävyydestä valmistelu eri menetelmillä (mukaillen Nielsen & Landauer, 1993, s. 211)

Projektin koko	Käyttäjättestaus			Heuristinen evaluointi		
	Koehenkilöitä	Hinta dollareina ¹¹	Hinta euroissa ¹²	Evaluointien määrä	Hinta dollareina	Hinta euroissa ¹³
Pieni	7	16 400 (10 000)	12 000	9	15 400 (9 400)	11 300
Keskikokoinen	15	30 200 (18 000)	22 100	16	22 300 (13 600)	16 300
Suuri	20	75 500 (46 000)	55 200	21	54 500 (33 200)	39 900

Tärkeä näkökulma kustannuksiin on tietysti myös hyötysuhde tai kustannustehokkuus. Nielsen ja Landauer (1993) kuvaavat hyötysuhteen (hyödyt / kustannukset) vaihtelevan heidän kartoittamissaan tutkimuksissa välillä 3,8 – 247, ollen sitä korkeampi, mitä suurempi projekti oli kyseessä. Samassa tutkimuksessa yksittäisen käytettävyyssongelman löytämisen hyödyksi arvioitiin suuressa projektissa 24 600 (15 000) dollaria, ja pienessä projektissa 1 640 (1 000) dollaria. Dona-

⁷ Hinnat on ilmaistu vuoden 2014 dollareissa, suluissa alkuperäinen, tutkimuksessa mainittu hinta vuoden 1993 dollareissa. Muuntamisessa on huomioitu inflaatio ja kuluttajahintaindeksin muutos, ja apuna on käytetty US Inflation Calculator -sivuston laskuria (CoinNews, 2014). Summat on lopuksi pyöristetty kolmen merkitsevän numeron tarkkuuteen

⁸ Muunnettu vuoden 1994 dollarit (suluissa) vuoden 2014 dollareiksi samalla menetelmällä kuin alaviitteessä 7.

⁹ Alkuperäinen arvo (suluissa) vuoden 2005 dollareina

¹⁰ Muuntamisessa käytetty Aktian valuuttamuunninta (Aktia Oyj, 2014)

¹¹ Katso alaviite 7.

¹² Muuntamisessa käytetty Aktian valuuttamuunninta (Aktia Oyj, 2014)

¹³ Muuntamisessa käytetty Aktian valuuttamuunninta (Aktia Oyj, 2014)

hue (2001) mainitsee anekdoottisena havaintona, että yksinkertainen käytettävyysoongelma, joka päättyi sovelluksen tuotantoversioon käytettävyytestauksen laiminlyömisen vuoksi, maksoi yritykselle 900 000 dollaria. Tutkijoiden mukaan ongelma olisi voitu havaita erittäin yksinkertaisella käytettävyytestauksella.

4.7 Käytettävyytestauksen käytännön toteutuksesta

Aikaisemmin käytettävyytestaus tapahtui laboratorioissa, joissa testien valvonta ja tarkastelu suoritettiin lähinnä etäältä, esimerkiksi heijastavan ikkunan läpi viereisestä huoneesta. Menetelmien kehittyessä epäformaalimpi ”vieressä istumisen” (engl. sit-by) -tyyli on yleistynyt, jolloin erityisesti tutkivassa testauksessa jatkokysymysten esittäminen on helpompaa (Rubin, 2008, s. 54). Tämä tyyli myös yhdistyy erittäin luontevasti esimerkiksi ääneen ajatteluun menetelmänä. Ohjaaja voi kuitenkin päästä vaikuttamaan testin lopputuloksiin ja näin huonontaa tulosten laatua. Lisäksi saadut tulokset ovat paljolti testaajan/ohjaajan ammattitaidosta kiinni. Sessioiden järjestämistä suositellaan usein tehtäväksi muualla kuin ”laboratoriossa”, mieluiten luontaisessa käyttöympäristössä, jotta koehenkilöt olisivat mahdollisimman luonnollisia. Toisaalta toisissa tutkimuksissa tutkimuksen tulosten osalta ei ole havaittu merkittävää eroa laboratorio- ja kenttätutkimusten välillä (Kallio, 2005), joten ei voida ainakaan yksiselitteisesti sanoa, mikä tutkimusmenetelmä tuottaa varmin luotettavia tuloksia.

Koehenkilöiden täytyy tuntea olonsa mukavaksi session aikana. Session järjestäjien tehtävänä on varmistaa tämän toteutuminen (Krug, 2000, s. 156–157). Yksi olennainen asia on viitata koehenkilöihin jollakin neutraalilla termillä, joka ei ole alentava tai halventava – huonoja esimerkkejä ovat ainakin koe-eläin, koe-kaniini, ehkä myös koehenkilö. Koehenkilön suoriutumista ei ole syytä kommentoida, mutta jos suoritus jää selkeästi johonkin hankaluuteen jumiin, koehenkilön voi auttaa siitä yli viimeistään hänen selkeästi turhautuessa (Krug, 2006; Nielsen, 1993, s. 183,190). Koehenkilöiden kanssa voi olla syytä allekirjoittaa salassapitosopimus (riippuen tietysti projekteista, joihin he ottavat osaa), ja heille kannattaa mainita, ettei heidän tietojään tulla jakamaan eteenpäin eikä heitä tulla yksilöinä käsittelemään. Videotallenteen käyttöä varten kannattaa pyytää lupa selkokieli-sellä ja lyhyellä sopimuksella (Nielsen, 1993, s. 189).

Koehenkilön pyytäminen ajattelemaan ääneen tuotetta käyttäessään voi paljastaa arvokasta tietoa esimerkiksi päättelyketjuista jotakin tehtävää suoritettaessa ja tietenkin tarkempaa tietoa koehenkilön suhtautumisesta eri ominaisuuksiin tuotteessa. Se kuitenkin muuttaa tilannetta epärealistisemmaksi (juuri kukaan ei ajattele ääneen tositilanteessa, ääneen ajattelu saattaa helpottaa käyttöä) ja kuluttaa aikaa. Erityisesti jos ajankäyttöä mitataan, ääneen ajattelu on huono tapa (Rubin, 2008, s. 54). Osa tutkijoista on ehdottanut koeasetelmaa, jossa kaksi koehenkilöä voisi yhdessä käyttää järjestelmää ja keskustella keskenään, jolloin tilanne olisi huomattavasti luontevampi (Holzinger, 2005, s. 73). Tätä voidaan kutsua *rakentavaksi vuorovaikutukseksi* (engl. constructive interaction).

Jos aikaa on käytettävissä paljon, session nauhoitteen voi käydä koehenkilön kanssa vielä läpi ja pohtia ongelmakohtia uudelleen, mikä voi antaa lisätietoa siitä, mikä oli vaikeaa ja mitä voisi korjata (Rubin, 2008, s. 55). Myös session aikana ilmenneet vaikeudet kannattaa käydä nopeasti läpi ja painottaa koehenkilölle tämän panoksen ja avun tärkeyttä ja hyödyllisyyttä (Nielsen, 1993, s. 183). Ennen mitään pohdintaa tai keskustelua kannattaa kuitenkin pyytää koehenkilöä täyttämään käyttäjätyytyväisyyttä mittaava kysely, mikäli tällaista tietoa kerätään (Nielsen, 1993, s. 191). Mikäli on mahdollisuus järjestää useita testisessioita, aiempien sessioiden koehenkilöiltä kannattaa kerätä palaute paitsi session yleisistä järjestelyistä, myös käyttötapausten realistisuudesta. Palautteen perusteella voi olla mahdollista paitsi muuttaa testitapauksia realistisemmiksi tai hyödyllisemmiksi, jopa parantaa varsinaisia käyttötapauksia, jos jotain oikeasti merkittävää tulee ilmi. (Sy, 2007, s. 124.)

4.8 Käytettävyydestauksen sudenkuoppia

Käytettävyydestauksessa on aina ongelmansa riippumatta toteutustavasta. Edes hyvin valitut menetelmät ja osaava henkilökunta niitä suorittamassa eivät välttämättä takaa onnistumista. Näiden sudenkuoppien välttäminen käytettävyydestausta ja ketterää ohjelmistokehitystä yhteen sovittaessa korostuu, sillä erityisesti pienehköistä otannoista aiheutuvat sudenkuopat ovat tavallistakin suuremmassa roolissa kevyessä käytettävyydestauksessa. Tässä alaluvussa kuvataan ongelmia ja esitetään niihin ratkaisuja, joiden pohjalta sudenkuoppien aiheuttamia ongelmia voidaan välttää kevyen käytettävyydestauksen ja ketterän ohjelmistokehityksen yhteen sovittavassa mallissa.

4.8.1 Suunnittelu ja toteutus saman sprintin aikana epäonnistuvat

Empiirisissä kokeissa on huomattu, että suunnittelu ja toteutus saman sprintin aikana onnistuvat huonosti (Budwig ym., 2009). Vaatimusten jatkuva muuttuminen aiheuttaa hankaluuksia UX-tiimille, mikäli kommunikaatiossa on ongelmia. Heikkolaatuinen vaatimusmäärittely vaikeuttaa myös oikeanlaisten tuotosten aikaansaantia. Myös toteutuksen koordinointi ulkoisten tai toisella paikkakunnalla sijaitsevien tiimien kanssa voi olla hankalaa. (Budwig ym., 2009, s. 3077.)

Hybridimallina edellä kuvattujen ongelmien ratkaisemiseksi on esitetty UX-tiimiä tai UX-scrumtiimiä, joka toimii varsinaisen scrumtiimin (toteutustiimin) ulkopuolella (Budwig ym., 2009). Tiimillä on oma tuoteomistaja, ja se toimii yhdestä kahteen sprinttiä toteutustiimin edellä ja järjestää esimerkiksi neljännesvuosittain "design vision"-sprinttejä, joiden aikana suunnitellaan seuraavan 3-6 kuukauden toteutusta. UX-tiimi tekee tiivistä yhteistyötä liiketoimintayksikön (engl. business unit) kanssa, jotta vaatimusmäärittelyn laatu olisi mahdollisimman korkea. Eri tiimit sijaitsevat samassa paikassa koordinointikuorman vähentämiseksi. (Budwig ym., 2009, s. 3078, 3080–3083.)

Toteutuksen edellä kulkeva UX-prosessi ottaa vastaan palautetta käytettävyystestaukselta ja näiden pohjalta suunnittelee toteutusta. Mallilla on saavutettu laadukkaammat vaatimukset, niiden helpompi kommunikointi ja toteutus. Mallilla on kuitenkin ongelmansa. UX-tiimin ajasta voi kulua kohtuuttoman suuri osa vaatimusmäärittelyn tarkentamiseen. Palautetta toteutustiimiltä ei saada ajoissa, jos suunnittelu tehdään ennen toteutusta ja palaute tulee vasta myöhemmin toteutuksen jälkeen. Lisäksi testauksen tuottamat uudet ominaisuudet ja virheenkorjaukset voivat vaikeuttaa aikataulussa pysymistä. (Budwig ym., 2009, s. 3079–3080.)

4.8.2 Onko järjestelmä käyttökelpoinen?

Helppokäyttöinen ei tarkoita käyttökelpoista (vrt. engl. *usable* vai *useful*?). Se, että järjestelmä tai palvelu on helppokäyttöinen, ei takaa, että käyttäjillä olisi syytä käyttää sitä. Onkin vähintäänkin tärkeää testata, että järjestelmää käyttämällä voidaan saavuttaa sen käytölle asetetut tavoitteet (Dicks, 2002, s. 4).

Kevyt ja melko suoraviivainen keino sen varmistamiseksi, että käytettävyystestauksen avulla saadaan testattua myös järjestelmän hyödyllisyyttä, voisi olla järjestelmän käytettävyystestauksessa käytettävien testitapausten laatiminen käyttäjätarinoiden pohjalta. Krug (2006) sivuaa tätä esittäessään, että käytettävyystestaussessioiden kulun pitäisi noudattaa koehenkilöiden todellisia tarpeita. Samoin Sohaib ja Khan (2011, s. 6) esittävät, että XP:n tarinoihin liitetään konteksti ja niistä saadaan luotua uskottavia skenaarioita käytettävyystestauksen koehenkilöitä varten. Mikäli scrumin käyttäjätarinoiden tai XP:n tarinoiden kattavuus on tarpeeksi hyvä, niiden pohjalta suoritetut käytettävyysseisiot varmistavat, että kaikki toiminnallisuus järjestelmästä tulee testatuksi, ainakin mikäli testausresurssit riittävät.

4.8.3 Testitapausten kattavuus käyttäjätarinoiden perusteella

Kaiken olennaisen testaaminen on hankalaa. Prototyypin tai järjestelmän jonkin version kaiken toiminnallisuuden testaaminen vaatii, että niitä jokaista varten on suunniteltu testitapaukset, kevyessä käytettävyystestauksessa siis todennäköisesti skenaariot ”ääneen ajattelu”-testausta varten. Aiemmassa tutkimuksessa on havaittu, että kokeneetkaan käytettävyystestaajat eivät välttämättä osaa laatia testitapauksia kaikkia vaatimuksia kattavasti (Kessner ym., 2001, s. 35). Onkin ehdotettu että ”riittävän monen koehenkilön” sijaan pitäisi keskittyä ”riittävän kattaviin testitapauksiin” (Lindgaard & Chattrachart, 2007, s. 1).

Scrumin ja kevyen käytettävyystestauksen yhteensovittamisessa ongelma voitaneen ratkaista sillä, että jokaista käyttäjätarinaa kohden laaditaan vähintään yksi sen toteuttava skenaario, jota varioidaan eri koehenkilöille ja mahdollisesti eri persoonille sopivaksi. Testaajan täytyy kuitenkin tasapainoilla saatavilla olevien resurssien ja testitapausten kattavuuden välillä: paitsi että kattavien testitapausten laatimiseen kuluu huomattavasti aikaa, niiden toteuttaminen testausseisioissa myös vaatii enemmän resursseja, tai vähentää yksittäisten testitapausten

toistoja. Tähän liittyen Kessner ym. (2001, s. 36) huomauttavat, että testitapausten kattavuus ja oikeellisuus todennäköisesti onnistuu paremmin, kun testausryhmä voi kommunikoida kattavasti varsinaisen toteuttajaryhmän kanssa, mikä sopiikin ketterään toimintatapaan hyvin.

Kessner ym. (2001) painottavat asiakkaan mukaanottoa testaussuunnitelmaa tehtäessä. Ketterässä kehityksessä asiakas otetaan kuitenkin mukaan käyttäjätarinoita tehtäessä, ja käyttäjätarinat ovatkin periaatteessa "täydellisiä" niiden tullessa käytettävyydestä käsiin. Testaaja voi niiden pohjalta laatia testaussuunnitelman. Käytännössä käytettävyydestä saatu palaute voidaan antaa projektiryhmälle, joka päättää siihen reagoinnista, ja palaute menee asiakkaalle projektiryhmän läpi suodatettuna. Tarvittaessa palautteen pohjalta voidaan tehdä muutosehdotuksia tai virheraportteja.

4.8.4 Persoonien käyttö ja kohderyhmävastaavuuden merkitys

Krugin (2006) mukaan koehenkilöiden kohderyhmävastaavuudesta ei kannata kantaa liikaa huolta, mutta suurin osa tutkijoista pitää sitä tärkeänä. Myös Krug muistuttaa, että koehenkilöiden mahdollinen eroavuus loppukäyttäjistä tulee kuitenkin huomioida tuloksia arvioidessa. On kuitenkin huomionarvoista, että todennäköisesti kaikki käyttäjät kärsivät kehnosta käytettävyydestä ainakin jonkin verran, vaikka taitava käyttäjä voikin pärjätä hankalankin ohjelman taitamaton paremmin (Kessner ym., 2001, s. 36). Koehenkilöiden loppukäyttävyyden vastaavuus pitäneen kustannusten minimointiin pyrkiessä arvioida aina tapauskohtaisesti.

Mikäli koehenkilöiden ja loppukäyttäjien erot eivät niinkään ole tiedoissa kuin esimerkiksi demografisissa seikoissa, eroja voidaan jonkin verran paikata käyttämällä persoonia. Persoonan omaksuminen voi auttaa huonostikin loppukäyttäjää vastaavan koehenkilön kykyä havaita ongelmia (Lindgaard & Chattratichart, 2007, s. 1423). Persoonan käyttö voi olla erityisen hyödyllistä siis silloin, kun koehenkilöiden kohderyhmävastaavuus on tärkeää, mutta varsinaiseen käyttäjäryhmään saatavilla olevia koehenkilöitä on todella heikosti tarjolla.

4.8.5 Käytettävyydestä tulosten luotettavuus

Jyrkin kritiikki käytettävyydestä kohtaan on esitetty sen tulosten aukottomuuden, kattavuuden ja tieteellisen hyödyntämisen kannalta. Ilmeisesti siis tuloksia sinällään ei kiistetä, eikä menetelmien käyttöä yrityksissä paheksuta, mutta kevyiden ja nopeiden käytettävyydestä tulosten väittäminen tilastollisesti merkittäviksi ja tieteellisen tarkastelun kestäviksi ei ole sopivaa (Nielsen, 1994b, s. 3). Mikään määrä käytettävyydestä ei myöskään voi "todistaa tuotteen olevan käytettävä" tai "paljastavan kaikki käytettävyysongelmat" (Jokela, 2010, s. 70).

Kevyessä käytettävyydestä kvanitatiivisen, tilastollisen analyysin kohteeksi kelpaavan datan sijasta kerätään kvalitatiivista dataa, jolloin koeses-

sion valvojan rooli ja ammattitaito korostuu. Krugin (2006) mukaan testaussession valvojan tehtävään sopii kuka vain kohtuullisen sosiaalinen ja tarkkailukykyinen henkilö, ja tehtävässä kehittyy nopeasti paremmaksi, mutta toisaalta ammattitaidoton session valvoja voi olla huomattavasti ammattitaitoista valvojaa tehottomampi tehtävässään. Toisaalta myös kokeneiden testaajien tulokset voivat poiketa toisistaan suuresti (Kessner ym., 2001, s. 2). Olennaista kuitenkin lienee, että vakavien käytettävyysongelmien löytämisessä eri ryhmät olivat yhtenäisempiä tuloksissaan, suurimmat erot ilmenivät vähemmän merkittävien ongelmien suhteen (Kessner ym., 2001, s. 33).

4.8.6 Käytettävyyden tärkeyden aliarviointi tai tulosten kyseenalaistaminen

Eräs vakavimmista sudenkuopista liittyy käytettävyyden huomioinnin vaativaan ajattelutavan muutokseen: mikäli projektitiimi ei sitoudu huomioimaan käytettävyyttä, käytettävyydestaus voi jäädä päälle liimatuksi näpertelyksi, eikä esimerkiksi testauksen tuloksia seuraa kukaan. Ilmeisimmillään tämä on silloin, kun käytettävyyssasiantuntijan rooliin otetaan uusi, usein nuori ja kokematon henkilö, jonka saamia tuloksia sitten vähätellään (Jokela, 2010, s. 76). Toisaalta, oli käytettävyydestaaja kuka tahansa, tulosten vähättely ja tyrmääminen on aina mahdollista.

Näitä ongelmia pienentää paljolti se, jos suurimmille kriitikoille voidaan suoraan havainnollistaa löydettyjä ongelmia. Käytettävyydestaustessio kannattaa nauhoittaa, jotta sitä voidaan tarkastella jälkeinpäin, mutta myös mahdollisuus seurata tapahtumaa "livenä" on arvokas ja voi vakuuttaa erityisesti käytettävyydestauksen arvon kyseenalaistavat jäsenet organisaatiossa (Rubin, 2008, s. 48.)

4.9 Käytettävyydestauksen havaintojen validointi

Tässä alaluvussa tarkastellaan aiempien käytettävyydestauksen ja ketterän ohjelmistokehityksen yhdistämistä tarkastelevien tutkimusten tapoja validoida havaintonsa. Käytettävyydestauksen tulosten validiteetti on pystyttävä varmistamaan, sillä onhan tärkeää, että paitsi että havaintoja tehdään, ne myös ovat hyödyllisiä. Yritysmailmassa tulosten hyödyllisyyden osoittamiseen riittää se, että ne koetaan hyödylliseksi: käytäntö siis ikään kuin osoittaa tietojen validiuden. Tämä ei kuitenkaan riitä tieteellisessä tutkimuksessa tietojen oikeellisuuden tai hyödyllisyyden takeeksi. Alaluvussa kerrotaan, kuinka aiemmissä tutkimuksissa havaintojen validointi on suoritettu. Tarkastelussa painotetaan erityisesti sellaisia organisaatioita, jotka ovat hyödyntäneet ketterää ohjelmistokehitystapaa ja siihen sopivia käytettävyydestausmenetelmiä.

Budwig, Jeong ja Kelkar (2009) toimivat tapaustutkimuksessaan organisaationsa käyttäjäkokemus- ja suunnittelutiimissä yhdessä scrumtiimien kanssa

projekteissa. Tutkimusraportissaan he lähinnä raportoivat tiimensä jäsenten kokemuksi ja mielipiteitä käytettävyydestä, eivätkä noudattaneet mitään sen formaalimpaa tapaa tulosten validoimiseksi (Budwig ym., 2009, s. 3077, 3079).

Nielsen (1995a) toimi tapaustutkimuksessaan projektiryhmän jäsenenä ja toisena kahdesta käytettävyyssuunnittelijasta. Hän raportoi, kuinka projektissa toteutettiin käytettävyystestaus. Tutkimuksessa käytettävät käytettävyystestausmenetelmät jossain määrin validoivat toinen toisensa – korttien lajittelulla selvitetiin ensin kategoriat, joiden pohjalta suunniteltiin kuvakkeet, joiden intuitiivisuus testattiin ja joiden pohjalta suoritettiin suljettu korttien lajittelu järjestelmän paperiprototyypille. Lopulta suoritettiin läpikäyntejä järjestelmän paperiprototyypillä. Tulosten validointi perustuu Nielsenin johtopäätöksiin ja arvioon projektin hyvästä menestyksestä.

Holzinger ym. (2005) tarkastelevat XP:n ja käytettävyyssuunnittelun integrointia, jonka tulosta he kutsuvat nimellä ”Extreme Usability”, erityisesti kouluttamisen näkökulmasta. He kuvaavat tärkeimmiksi käytettävyydestä mitattaviksi muuttujiksi tehtävän suorittamiseen kuluvan ajan ja tehtävien suorituksessa syntyvät ongelmat. Heidän mukaansa nopeasti iteraatiot tuottavat nopeasti tarkasteltavia tuloksia ja lopulta parantavat merkittävästi käytettävyyttä. Nämä tulokset voidaan validoida asiakkaan edustajan avulla tai jopa asiakkaan edustajaa koehenkilönä käyttäen. (Holzinger ym., 2005, s. 2–4)

Itkosen ym. (2005) tutkimuksessa esitettiin, että ketterää ohjelmistokehitysprosessia voidaan parantaa ottamalla käyttöön testauskäytäntöjä. Tärkeänä tulosten kannalta pidettiin ajankohtia, joiden kohdalla suoritetaan tietynlaista testausta ja jotka antavat testaajalle ja kehittäjälle mahdollisuuden arvioida tuloksia ja ohjata työtään. Tällöin kehittäjä saa siis nopeasti palautteen järjestelmän laadusta. Tutkijat mainitsevat tehneensä tutkivaa testausta ja mainitsevat kokemustensa olleen rohkaisevia, mutta eivät kuvaa tapaa, jolla tuloksia on validoitu (Itkonen ym., 2005, s. 6). Lisäksi heidän mukaansa testaaminen on syytä tehdä niin aikaisin, että tuloksiin voidaan reagoida ja julkaisutason testauksen sijaan tulee keskittyä lyhemmän aikavälin testaukseen (Itkonen ym., 2005, s. 6).

Lee ja McCrickard (2007) käsittelevät tutkimuksessaan keskitetyn suunnittelupöytäkirjan (engl. Central Design Record, CDR) yhdistämistä Extreme Programming -menetelmään. Tapaustutkimuksissaan kehittäjät tekevät vähintään jokaisen iteraation lopussa kevyen käytettävyyssarviointin, mutta lisäksi suoritetaan läpikäyntejä asiakkaan edustajien kanssa. Tapaamisia, joissa arviointia ainakin välillä suoritettiin, pidettiin viikoittain, millä pyrittiin korvaamaan tiimiin kuuluvan asiakkaan puutetta (Lee & McCrickard, 2007, s. 5). Suunnitelmallisempi testaus ajoitettiin iteraation loppuun. Iteraation lopussa suoritettava arviointi suoritettiin käytännössä osana hyväksyntätestausta. Asiakas antoi käytettävyysspalautetta ja vahvisti järjestelmän toiminnallisuuden oikeellisuuden. Nopeiden ja kevyiden käytettävyyssarviointien lisäksi julkaisusyklin lopussa suoritettiin laajemman kuvan saamiseksi ja aiempien havaintojen validoimiseksi perusteellisempi käytettävyystestaus, jossa pyrittiin käyttämään mahdollisimman edustavia koehenkilöitä (Lee & McCrickard, 2007, s. 9). Tutkijat eivät esitelleet muuta tapaa havaintojen validoimiseksi.

Marty ja Twidale (2005, s. 5) toteavat, että kevyen käytettävyydestestauksen tulosten validointi voi itse asiassa olla likimain mahdotonta (Marty & Twidale, 2005, s. 5). Toisaalta, he ovat tutkineet erittäin kevyen käytettävyydestestauksen menetelmiä 36 tapauksessa ja toteavat saaneensa hyödyllisiä tuloksia niistä jokaisessa. 20 tapauksessa käytettiin menetelmänä yksinkertaista ääneen ajattelua ja 16 tapauksessa konstruktivistista interaktiota, jossa kaksi vapaaehtoista koehenkilöä pohti tapoja suorittaa tehtäviä järjestelmän avulla (Marty & Twidale, 2005, s. 6). He pyrkivätkin tilanteeseen, jossa pienimmillä mahdollisilla resursseilla saadaan hyödyllisiä tuloksia – siis pyrkivät pysymään juuri ja juuri sen rajan yläpuolella, jota Thomas (1996, s. 113) kuvaa pikaisen ja likaisen (engl. "quick and dirty") muuttumista nopeaksi ja saastaiseksi (engl. "fast and filthy"). Heidän mukaansa tällaisessa käytettävyydestestauksessa piilee kaksi haastetta: (1) vähäisimmän mahdollisen ajan käyttämiseen hyödyllisten tulosten saamiseksi ja (2) vain yhdeltä testikerralta saatujen tulosten luotettavuus. Nämä molemmat ovat ongelmia erityisesti validoinnin kannalta: tulosten saaminen ei sinällään ole hankalaa, mutta tulosten tulisi olla aina hyödyllisiä ja luotettavia. Marty ja Twidalen (2005) mukaan heidän menetelmänsä, jossa käytettävyydestestaussessio suoritetaan puolessa tunnissa, tähtää minkä tahansa määrän valideja tuloksia tuottamiseen lyhyessä ajassa ja näiden tulosten perusteella järjestelmän parantamiseen tavalla, joka hyödyttää käyttäjiä. He saivat 18 tunnin aikana järjestettyä 36 testausseksiota, joiden aikana toteutettiin 83 testitapausta ja tulosten pohjalta tuotettiin yli 500 suositusta tai ehdotusta, jotka hyväksyttiin koehenkilöillä. Lisäksi hyväksyntä havainnoille haettiin vapaamuotoisesti projektiryhmän edustajilta, käytettävyydestaajilta ja sessioiden tarkastelijoilta, ja jokaisessa tapauksessa havaintoja pidettiin vakavina käytettävyysongelmina (Marty & Twidale, 2005, s. 8). Tutkijat arvelevat, että heidän havaintonsa kuvaavat pääosin sivuston isoimpia ja ilmeisimpiä ongelmia ja niiden korjaaminen on niin sanotusti "matalalla roikuvan hedelmän poimimista". Havaintojen kannalta olennaista ei ollut edes se, montako tehtävää koehenkilöt saivat suoritettua tai kauanko niiden suorittamisessa kesti, vaan tärkeintä oli päästä tarkkailemaan käyttäjää tämän käyttäessä järjestelmää, huomioida ongelmat ja saada niiden pohjalta tuloksia, joita voitiin käyttää järjestelmän parantamiseksi. Tuloksia ei kuitenkaan validoitu ulkopuolista asiantuntijaa käyttäen, eivätkä tutkijat pitäneet sitä edes tärkeänä. Tutkijoiden mukaan olennaista oli, että käytettävyydestestauksen havainnot olivat käyttökelpoisia järjestelmän parantamisessa.

RITE+KRUG -menetelmässä (vrt. kohta 4.2.2) McGinn ja Chang (2013) validoivat käytettävyydestestauksen tulokset velvoittamalla sidosryhmät lähettämään edustajansa seuraamaan käytettävyydestestaussessioita. Sessioista otettiin lisäksi nauhoitteet siltä varalta, että tapahtumia olisi tarve tarkastella myöhemmin. Tämän toimintatavan seurauksena sidosryhmät ottivat tulokset vakavasti. Lisäksi tulosten huomioinnista seuranneiden korjausten tai parannusten validiteetti testattiin seuraavilla testauskerroilla (McGinn & Chang, 2013, s. 66).

Nielsen (1994b) on myös pohtinut kevyiden käytettävyydestestausmenetelmien tulosten luotettavuutta moneen otteeseen. Eräässä tutkimuksessa (Nielsen,

1994b) käyttöliittymä oli testattu useamman kerran yhdellä koehenkilöllä iteraatiota kohden ja tuloksia validoitiin kahdella eri menetelmällä: (1) koehenkilöille annettiin arvioitavaksi paranneltu tai alkuperäinen käyttöliittymä, ja (2) edellisen kokeen testaajilta ja satunnaiselta koehenkilöryhmältä kysyttiin, kumpi käyttöliittymistä oli mieleisempi. Ensimmäisessä kokeessa testattiin peräti 152 koehenkilöä, joita testaamassa oli 38 kokematonta testaajaa. Kaikkien näkymien ja mitattavien muuttujien osalta havaittiin parannusta, joskin muutokset eivät olleet aina suuria. Toisessa vaiheessa edelliseen kokeeseen osallistujat suosivat hyvin vahvasti paranneltua käyttöliittymää, mutta satunnaiset koehenkilöt jakautuivat melko tasan vanhan ja uuden käyttöliittymän välillä. Tutkimus (Nielsen, 1994b) siis osoitti, että siinä missä kevyiden käytettävyydestestausmenetelmien hyödyntämisen seurauksena käyttöliittymien laatu parani, se myös osoitti kokemattomien testaajien pystyvän arvioimaan käyttöliittymien käytettävyyttä paremmin koehenkilöitä hyödyntämällä kuin intuiutionsa varassa. (Nielsen, 1994b, s. 6–7.)

Yhteenvedona voidaan todeta, että aiemmassa tutkimuksessa kevyen käytettävyydestestauksen havaintojen validointia ei ole yleensä suoritettu. Käytännöt muutenkin ovat aika vaihtelevia eivätkä aina erityisen kurinalaisia. Muun muassa Kessner (2001) ei tutkimuksessaan ole edes varma, mitä menetelmiä hyödyntäen osa käytettävyydestestauksen havainnoista on saatu, mutta tulkitsee silti havaintojen olevan hyödyllisiä. Toisaalta Dicks (2002) on todennut kevyestä käytettävyydestestauksesta, että vaikka sen menetelmät eivät vastaakaan tieteellisen tutkimuksen täsmällisyyttä, ne voivat silti tuottaa hyödyllisiä tuloksia. Ne ovatkin riittävän hyviä, jotta niiden tuloksista on hyötyä, vaikka tieteellistä täsmällisyyttä ei noudatettaisikaan. Usein on sanottu havaintojen ikään kuin validoivan itsensä hyödyllisyydellään. Toisaalta niiden validoinnissa on voitu käyttää yksinkertaisesti palautetta projektiryhmältä (Haikara, 2007; Marty & Twidale, 2005) tai asiakkaalta (Holzinger ym., 2005).

Monikaan tutkija ei aiemmin ole erityisesti kiinnittänyt huomiota käytettävyydestestauksen havaintojen validointiin, vaan ne on luovutettu sellaisinaan projektiryhmille ja projektiryhmät ovat saaneet tehdä arvioinnin niiden validiteetista, esimerkiksi yhdessä käytettävyydestaajan kanssa (Haikara, 2007; Krug, 2000, 2006; Marty & Twidale, 2005). Toisaalta esimerkiksi Rubin toteaa käytettävyydestestauksesta yksinkertaisesti, että ” - - on parempi testata kuin olla testaamatta” (Rubin, 2008, s. 27).

4.10 Yhteenvedo

Tässä luvussa käsiteltiin käytettävyydestausta, erityisesti kevyitä menetelmiä, osana ketterää ohjelmistokehitystä. Käytettävyydestestauksen ja ketterän ohjelmistokehityksen yhteensovittamista on tutkittu aiemminkin. Vaikka ketterä ohjelmistokehitys ei sinällään otakaan kantaa käytettävyydestestauksen suorittamiseen, monilta osin se tukee käytettävyydestestauksen aktiviteetteja. Asiakkaan aktiivinen osallistuminen projektiin, aktiivinen kommunikaatio sekä iteratiivisuus ja inkre-

mentaalisuus toimivat hyvin yhteen, mutta myös ristiriitoja löytyy, sillä käytettävyydestä, toisin kuin esimerkiksi yksikkötestaus XP:ssä, ei ole määritelty osaksi kehitysprosessia, käytettävyyssarviointi ei varsinkaan aikaisen vaiheen sprinteissä oikein sovellu osaksi hyväksymistestausta, testauksessa haluttaisiin käyttää loppukäyttäjiä asiakkaan edustajan sijasta ja käytettävyydestä on vastuussa yleensä sitä opiskellut henkilö, mikä soveltuu huonosti ketterien menetelmien yleisosaaja-ajatteluun. Lisäksi erityisesti käyttöliittymäsuunnittelu aiheuttaa, nimensä mukaisesti, suunnittelua ennen toteutusta, mikä soveltuu hiukan huonosti yhteen ketterän kehityksen kanssa (McInerney & Maurer, 2005).

Luvussa tutustuttiin myös useisiin tapaustutkimuksiin ketterän ohjelmistokehityksen ja käytettävyydestä huolehtimisen yhteensovittamisesta. Näiden pohjalta voidaan päätellä, että tärkeää on ainakin saada vältettyä ristiriidat kehitysryhmän sisällä, esimerkiksi käytettävyyssuunnittelijoiden ja kehittäjien välillä, sekä saada vakuutettua projektiryhmä käytettävyyden tärkeydestä, löytää sopiva kompromissi koehenkilöiden määrän ja tyyppien suhteen ja löytää sopiva tapa hoitaa testaajan tai käytettävyydestä huolehtijan rooli. Lisäksi hyvin yhteen sopiviksi piirteiksi tai käytänteiksi havaittiin iteratiivinen ja inkrementaalinen toimintatapa, ketterän ohjelmistokehitystavan painottama jatkuva kommunikaatio sekä tiimin sisällä että asiakkaan suuntaan, yksinkertaisuuden ihannoiti, tulosten nopea toimittaminen ja prototyyppien hyödyntäminen (Lee & McCrickard, 2007). Eri menetelmiä käyttävissä organisaatioissa yhteen sovittaminen on tehty eri tavoin, ja näiden eri tapojen hyvistä ja huonoista puolista on pyrittävä oppimaan tulevia mallielementtejä kehitettäessä.

Luvussa pureuduttiin myös aiemman tutkimuksen havaintoihin joistakin alan, ja siten myös yhteensovituksen, olennaisimmista kysymyksistä. Käytettävyyshavainnot havaittiin mahdollisesti hyödyllisiksi, mutta kevyeen käytettävyydestä huolehtimiseen huonosti soveltuviksi työläytensä vuoksi. Käytettävyydestä huolehtimisen ajoituksesta havaittiin, että käytettävyydestä on syytä aloittaa jo verraten aikaisin kehityksen aikana ja toistaa usein. Käytettävyydestä huolehtimisen oikeasta määrästä löytyy useita näkemyksiä: vähäkin testaus on parempi kuin ei mitään, ja varsinkin tarvittava määrä ratkeaa sen mukaan, mitä testaukselta halutaan. Tuotteen hyvää käytettävyyttä on hankala tai jopa mahdotonta todentaa testauksella, ellei testausta ole valmis tekemään varsin paljon, ja toisaalta yksittäisiä tai pienellä otannalla saatuja tuloksia ei voi yleistää, mutta hyödyllistä tietoa tuotteen käytettävyydestä voidaan saada jo vähäiselläkin määrällä testausta.

Luvussa käytiin myös läpi käytettävyydestä huolehtimisen keskeisimpiä sudenkuoppia. Havaittiin, että käytettävyys kärsii, jos suunnittelu ja toteutus pyritään toteuttamaan rinnakkain, saman sprintin aikana. Ollakseen käytettävä (engl. usable) järjestelmällä täytyy olla selkeä tarve ja hyöty (engl. useful). Testitapausten kattavuus havaittiin myös edellytykseksi käytettävyydestä huolehtimisen onnistumiseksi. Koehenkilöiden tarkoituksenmukainen vastaavuus loppukäyttäjien huomioitiin, mutta vaadittavat toimenpiteet riippuvat paljolti projektin ja loppukäyttäjien luonteesta. Lisäksi persoonilla voidaan parantaa koehenkilöiden roolin omaksumista, joskaan puuttuvia tietoja tällä ei toki voida paikata. Käytettävyydestä huolehtimisen tulokset voivat vaihdella suurestikin riippuen siitä, kuka ja

kuinka kokenut henkilö testauksen suorittaa ja tulokset arvioi. Testaajan merkitys korostuu entisestään kevyissä käytettävyystestausmenetelmissä, joissa tilastollinen analyysi jätetään tietoisesti pois ja havainnot ovat luonteeltaan enemmän anekdoottisia ja vaativat tulkintaa. Käytettävyystestaus, tai jopa käytettävyys yhtenä tuotteen laadun mittareista, voi jäädä hyvin vähälle huomiolle jo pelkästään alan tai sen harjoittajien arvostuksen puutteen vuoksi. Lisäksi on huomioitava, että pienellä otannalla saadut havainnot eivät voi olla tilastollisesti yleistettävissä.

Lopuksi käsiteltiin käytettävyystestauksen tuottamien havaintojen validointia. Aiemmassa tutkimuksessa tavat havaintojen validointiin ovat olleet varsin vaihtelevat, ja usein käytännön on katsottu osoittavan tulosten hyödyllisyyden. Usein on voitu käyttää projektiryhmää tai asiakasta apuna.

5 TAPAUSTUTKIMUKSEN TOTEUTTAMINEN

Tässä luvussa kerrotaan, miten käytettävyydestauksen ja ketterän ohjelmistokehityksen integrointia on tutkittu empiirisesti. Tutkimuksen tavoitteena on selvittää, *miten kevyitä käytettävyydestausmenetelmiä voidaan integroida ketterään ohjelmistokehitykseen ja mitkä ovat kevyiden käytettävyydestausmenetelmien ketterään ohjelmistokehitykseen integroinnin hyödyt ja kustannukset käytännön tilanteissa*. Ensiksi kuvataan ja perustellaan valittu tutkimusmenetelmä. Toiseksi kerrotaan tutkimuksen kohteena olevasta organisaatiosta. Kolmanneksi kuvataan tutkimusprosessi, jolla luodaan integrointimalli ja kokeillaan sitä käytännössä neljässä projektissa, jotka myös kuvataan. Viidenneksi kerrotaan tiedonkeruusta ja analysoinnista ja kuudenneksi esitellään lyhyesti testauksen toteutustapaa käytännössä. Lopuksi käsitellään vielä tapoja tapaustutkimuksen tulosten validointiin.

5.1 Tutkimusmenetelmä

Tutkimustavoitteen mukainen tutkimus voitaisiin toteuttaa ainakin kyselytutkimuksena ja tapaustutkimuksena. *Kyselytutkimuksen* tarkoituksena on tiedon kerääminen tietämyksen, asenteiden ja käyttäytymisen kuvaamiseksi, vertaamiseksi ja selittämiseksi (Pfleeger & Kitchenham, 2001, s. 16). Kyselytutkimuksen vahvuutena on perinteisesti pidetty suhteellisen helppoa järjestämistä, yksinkertaista arvostelemista, hyvin suoraviivaista tiettyjen muuttujien arvojen ja suhteiden selville saamista sekä sitä, että niiden tulokset voidaan yleistää samaan populaatioon ja usein myös samankaltaisiin muihin populaatioihin. Lisäksi niiden uudelleenkäyttö on helppoa, jolloin myös tuloksia voidaan vertailla eri ryhmien, ajankohtien tai sijaintien välillä (Newsted, Munro, Schwarz, & Cu, 2014). Kyselytutkimuksen heikkoutena on niiden rajoittuneisuus: ne ovat ikään kuin kaappaus tietyn hetken tilanteesta tietyssä paikassa, eivätkä ne tarjoa yhtä vahvaa osoitusta kausaliiteetista asioiden välillä kuin laajemmat tutkimukset tai yhtä laajaa kuvausta tarkasteltavasta ilmiöstä kuin tapaustutkimukset. Lisäksi tulosten

yleistämässä täytyy noudattaa varovaisuutta ja vastaajien edustavuuteen täytyy kiinnittää huomiota (Kitchenham & Pfleeger, 2002a). Kyselytutkimukset eivät myöskään sovellu konstruktiviseen tutkimukseen. (Newsted ym., 2014.)

Tapaustutkimuksessa ilmiötä voidaan tutkia sen luonnollisessa kontekstissa (Runeson & Höst, 2008). Se sopii hyvin vastaamaan kysymykseen ”kuinka?” ja ”miten?” (Yin, 2009). Tällöin voidaan selvittää ilmiöiden taustalla olevia syitä, erilaisia näkökantoja ja selityksiä. Tapaustutkimuksen heikkoutena on se, että tuloksia ei voida yleistää, vaan ne ovat valideja vain ko. kontekstissa.

Tämän tutkimuksen menetelmäksi valittiin tapaustutkimus siksi, että tällä tavalla päästään luomaan uusi toimintamalli ja käytännössä testaamaan sen toimivuutta. Tapaustutkimukseen tarjoaa oivalliset mahdollisuudet myös se seikka, että tutkija toimii organisaatiossa, jolla on tarve ja kiinnostus uudelleen toimintatapaan. Tapaustutkimus on tyypiltään toteutusta tarkasteleva tapaustutkimus (Commonwealth Association for Public Administration and Management, 2010) ja sitä voi luonnehtia kuvailevaksi (engl. descriptive) ja kehittäväksi (Runeson & Höst, 2008; Yin, 2009).

Tapaustutkimusta on kritisoitu kerättävän datan suurpiirteisestä käsitteilystä ja tutkimuksen pitkästä kestosta, jonka tuloksena saadaan heikosti yleistettävissä olevia tuloksia, joiden dokumentaatio on hankalassa muodossa (Runeson & Höst, 2008; Yin, 2009). Kritiikkiin voidaan tässä tutkimuksessa vastata siten, että tulokset ja menettelytavat dokumentoidaan tarkasti, jotta niiden hyödyllisyys ja menettelytavan validius voidaan vahvistaa. Niinpä dokumentaatio laaditaan sellaiseen muotoon, että käytettävyydestä malli on sopivin osin helposti toistettavissa ja muissa organisaatioissa hyödynnettävissä. Tulokset eivät sellaisenaan ole yleistettävissä, mutta toimintamalli ja havainnot sen toimivuudesta voivat tarjota hyödyllisiä tietoja käytettävyydestä suorittamiseen muissa organisaatioissa.

5.2 Tutkimuksen kohde

Tapaustutkimuksen kohdeorganisaationa on Samcom Oy (jäljempänä Samcom), joka on Jyväskylän suurimpia sähköisten palvelujen toimittajia (Samlink, 2014a). Sillä on asiakkaita sekä yksityiseltä että julkiselta sektorilta. Samcom on osa Samlinkia, joka on Suomen johtava monipankkijärjestelmien toimittaja. Samcom on viime vuosina kasvanut muutaman työntekijän nyrkkipajasta yli sata henkilöä työllistäväksi, monipuolisia palveluita ja isojakin järjestelmäkehitysprojekteja toteuttavaksi toimijaksi, ja toimii osana noin 500 henkilön konsernia (Samlink, 2014b).

Nopea kasvu ja suurentuneet projektit ovat aiheuttaneet omat haasteensa totutuille toimintatavoille. Yrityksen toimintatapojen yhtenäistämiseksi on kehitetty Samlinkin oma projektimalli SamScrum, jolla Samlinkin Ratkaisuliiketoiminta-yksikön projektit toteutetaan yhä enenevässä määrin (Litmanen, 2013b). Projektimallin kehitys aloitettiin vuoden 2012 aikana, ja yksi virstanpylväs saavutettiin vuoden 2012 syksyllä, kun ensimmäiset mallia noudattavat projektit

aloitettiin (Litmanen, 2013b, s. 3). SamScrum, kuten esikuvansa scrum (Lekman ym., 2012; Schwaber & Sutherland, 2013; Schwaber, 2002), ei ota kantaa käytettävyydestä toteuttamiseen, mutta kuvailee erityisesti asiakkaan suorittaman hyväksymistestauksen suoritusta. Kuten aiemmin tarkastellussa kirjallisuudessa (Crispin & Gregory, 2009; McInerney & Maurer, 2005; Northrop, 2011), myös SamScrumin tapauksessa testauksen tulokset syötetään ensisijaisesti seuraavien sprintin tehtävälistalle sen sijaan, että niihin reagoitaisiin välittömästi.

Käytäntö on osoittanut, että useimmat menestyvät yritykset eivät ole sänneet ottamaan ketteriä menetelmiä käyttöön suin päin, vaan ovat ennemminkin soveltaneet niitä soveltuvien osien ja aikaisempaan organisaatiokulttuuriinsa pohjaten (Bygstad ym., 2008, s. 375). Samlinkin tapauksessa scrumia on täydennetty erityisesti yrityksen käyttöön tulevien tuotosten kuvauksella ja projektin virstanpylväiden määrittelyllä. SamScrumissa (Litmanen, 2013b) ei kuitenkaan ole otettu kantaa käytettävyydestä tai käytettävyyteen ylipäätään millään tavalla. Käytännössä käytettävyys huomioidaan kyllä toteutuksessa kautta linjan, mutta enemmänkin tarpeen ja mahdollisuuksien mukaan kuin kurinalaisesti. Käytettävyyden suunnittelu tehdään osana muuta suunnittelutyötä, ja käytettävyys on laadullisena vaatimuksena huomioitu jo määrittelyvaiheessa. Yhtenäistä käytäntöä käytettävyyden testaukseen ei kuitenkaan ole. Tämä puute on tunnistettu, ja testausta varten halutaan kehittää malli. Samlink on myös tämän tutkimuksen tekijän työpaikka, joten resurssit ja hyväksyntä tutkimukselle on saatavissa helpommin kuin täysin ulkopuolisen yhteistyökumppanin kautta. Nykyisen toimintatavan haasteet ovat myös hyvin tunnettuja. Organisaatio tarjoaa mainion ympäristön sen tutkimiselle, miten kevyttä käytettävyydestä voidaan tehdä ja miten sitä varten kehitettävä malli toimii ketterää ohjelmistokehitystä tekevässä organisaatiossa.

5.3 Tutkimusprosessi

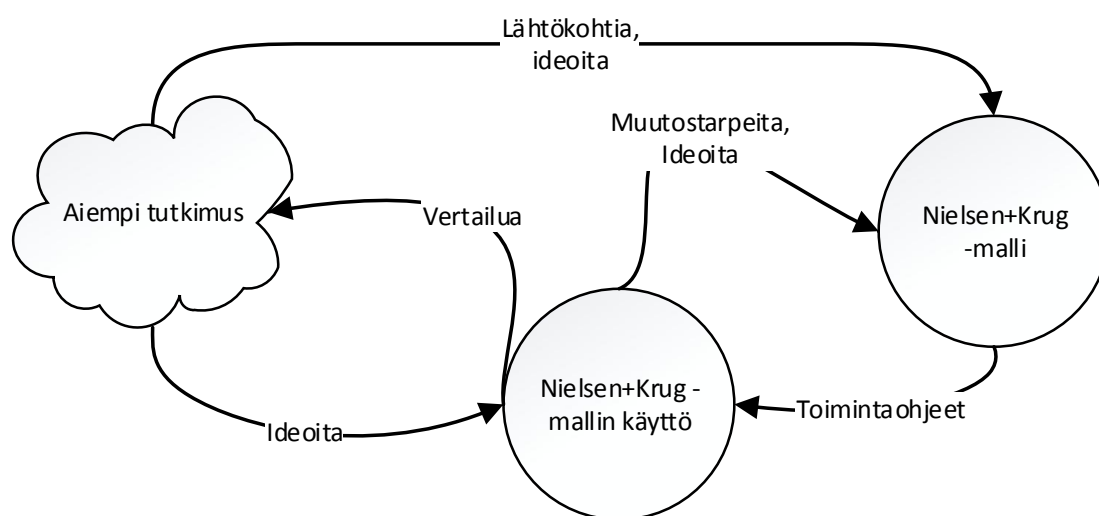
Tässä alaluvussa kuvataan tutkimuksessa hyödynnettävää tutkimusprosessia ja sen vaiheita. Kuvio 8 kuvaa tutkimusprosessin kulun. Aiempi tutkimus (Luvut 2, 3 ja 4) tarjoaa lähtökohdat ja ideoita, joiden pohjalta tässä työssä rakennetaan käytettävyydestä ja SamScrum-mallin mukaisen ketterän ohjelmistokehityksen integrointiratkaisu. Tätä ratkaisua kutsutaan nimellä *Nielsen+Krug* -malli. Se tarjoaa toimintaohjeet käytettävyydestä teolle käytännössä. Nielsen+Krug -malli kuvataan luvussa 6.

Nielsen+Krug -mallin toimivuutta kokeillaan käytännössä neljässä ohjelmistokehitysprojektissa. Nämä projektit ovat Samlinkin todellisia ohjelmistokehitysprojekteja, jotka on valittu niillä perusteilla, että projektiryhmästä ainakin projektipäällikkö, tuoteomistaja tai scrummaster on ilmaissut olevansa kiinnostunut osallistumaan pilottiin, projektin aikataulu on tutkimuksen kannalta sopiva ja projekti on sisällöltään sopiva tutkimusta varten. Projektit ovat:

- Sisäisen työsuunnittelun apuväline (RETU)

- Finanssialan asiakkaiden ajanvarausjärjestelmä (PA)
- Energia-alan asiakkaiden asiakaspalveluväylä (SH)
- Elintarvikealan yrityksen verkkosivut (HE)

Ohjelmistokehitysprojekteissa hyödynnetään Nielsen+Krug-mallia projektin tarpeet parhaiten täyttävällä tavalla mallin mukaista käytettävyydestä. Havainnot raportoidaan projektiryhmille, ja niistä pyydetään palautetta. Saadun palautteen perusteella mukautetaan jatkossa sekä käytettävyydestä toteutustapaa että erityisesti raportointikäytänteitä, jotta havainnot saadaan projektiryhmille mahdollisimman helposti hyödynnettävään muotoon. Tarkemmin käytettävyydestä käytännön järjestelyistä kerrotaan luvussa 5.6.



KUVIO 8 Tutkimusprosessin kulku

Nielsen+Krug -mallin käyttö käytännössä tarjoaa mallin toimivuudesta kokemuksia ja arvioita, joita vertaillaan aiempaan tutkimukseen. Kokemuksia käytetään myös Nielsen+Krug -mallin parantamiseen. Nämä muutostarpeet ja ideat syntyvät käytännön kokemusten ja projektiryhmien sekä koehenkilöiden antaman palautteen pohjalta. Palautetta kerätään systemaattisesti jokaisen testaussektion päätteeksi, jokaisen testausiteraation havaintojen toimituksen yhteydessä ja vielä tutkimuksen loppuun.

5.4 Projektit joissa testausta tehdään

Tässä alaluvussa kuvataan lyhyesti projektit, joissa käytettävyydestä tehtiin. Projektit eroavat toisistaan suurestikin. RETU on sisäiseen käyttöön kehitetty järjestelmä, PA on moniin muihin projekteihin ja järjestelmiin liittyvä finanssialan asiakkaalle räätälöity järjestelmä, SH on useille asiakkaille toimitettava Samlinkin oma tuote ja HE-projektissa testataan järjestelmää, joka on yhdistelmä useiden eri toimittajien verkkopohjaista ratkaisusta.

5.4.1 Sisäinen työsuunnittelun apuväline, RETU

Projektissa kehitetään sisäinen järjestelmä resursoinnin apuvälineeksi. Järjestelmä on web-pohjainen ja koko henkilöstön käytössä. Henkilöstön edustajilla on eri rooleja järjestelmässä, ja siten eri toiminnot ovat heille aktiivisia. Roolit määrittyvät sen mukaan, onko työntekijä osaaaja, resurssipyyntöjen tekijä, resursoija tai esimies. Useat työntekijät ovat useamman kuin yhden roolin edustajia. Tästä projektista käytetään lyhennettä RETU.

Järjestelmän testaus on tärkeää, koska esiteltäessä henkilöstölle uusia järjestelmiä ja toimintatapoja on muutosvastarinta aina suuri ongelma. Kiinnittämällä huomiota järjestelmän käytettävyyteen saadaan toisaalta helpotettua järjestelmän hyväksymistä, toisaalta myös annettua osoitus siitä, että työntekijöiden palautteesta ja mielipiteistä todella välitetään. Käyttäjistä on tunnistettu muutamia suurempia käyttäjäryhmiä: resursoijat (käyttävät kaikkia toimintoja), resurssipyyntöjen tekijät (kuten projektipäälliköt ja ratkaisukonsultit, käyttävät monia toimintoja), esimiehet (käyttävät vain joitakin toimintoja) ja osaaajat sekä loput (lähinnä tarkastelevat tietoja).

Nielsen+Krug-mallia testataan kahdesti, ensin julkaisun aikoihin (käytännön rajoitteista aiheutunut aikataulu), ja toisen kerran ensimmäisen ylläpitospintin loppupuolella, noin kuukauden kuluttua ensimmäisestä käytettävyydestäuskerrasta. Tarkoituksena on tarjota projektiryhmälle riittävästi aikaa ensimmäisen käytettävyydestäuskerran havaintoihin reagoimiseen.

5.4.2 Finanssialan asiakkaiden ajanvarausjärjestelmä, PA

Tässä tapauksessa testattavia projekteja on itse asiassa kaksi, mutta niiden käytettävyydestä suoritetaan samalla kertaa. Kyseessä on finanssialan asiakkaan ajanvarausjärjestelmä, joka on upotettu heidän uusille nettisivuilleen. Uusien nettisivujen toteutus on liitännäisprojekti, jolle havaintoja myöskin tuotetaan. Koehenkilöt edustavat kuluttaja-asiakkaan ja ylläpitäjäkäyttäjän rooleja. Tästä projektista käytetään lyhennettä PA.

Testauksen tavoitteena on saada tuloksia erityisesti ajanvarausjärjestelmän käytettävyydestä, mutta samalla myös nettisivuista muuten ja koko järjestelmän intuitiivisuudesta käyttäjän kannalta. Erilaiset, eri käyttäjäryhmiä edustavat koehenkilöt jakautuvat käytännön syistä eri sprintteihin siten, että ensimmäisessä sprintissä testataan kuluttajakäyttäjillä, toisessa sprintissä asiakkaan edustajilla ja järjestelmän käyttäjätuen henkilöstöllä, ja kolmannessa sprintissä vielä kuluttajakäyttäjillä.

Projektin ensimmäisellä testauskerralla tuotetaan dataa sekä nettisivu- että ajanvarausprojektille, toisessa vaiheessa testataan vain ajanvarausjärjestelmää ja sen ylläpitotoimintoja, ja kolmannessa vaiheessa taas nettisivu- ja ajanvarausnäkyä. Tulokset siis jakautuvat kahden erillisen projektin hyödynnettäviksi, mutta painopisteenä on ajanvarausprojekti. Testauskerrat ajoittuvat siten, että ne tehdään noin kolmen viikon välein ja kolmas ja viimeinen testauskerta ajoittuu

projektin viimeiseen toteutusprinttiin, ja sen havaintoihin voidaan reagoida tarpeen mukaan vielä saman sprintin aikana tai vasta seuraavassa sprintissä, joka on projektin ensimmäinen ylläpitosprintti.

5.4.3 Energia-alan asiakkaiden asiakaspalveluväylä, SH

Projektissa testauksen kohteena on eräänlainen energia-alan yrityksen kuluttaja-asiakkaille suunnattu myyntiportaali. Myyntiportaalin lopullisessa versiossa on muun muassa sopimusten hallinta ja monia muita käyttötapauksia, mutta siinä vaiheessa kun käytettävyydestä tehdään, järjestelmästä on vielä varsin keskeneräinen. Ymmärrettävistä syistä käytettävyydellä on suurin merkitys kuluttajakäyttäjien kannalta, ja lisäksi heidän käytettäväkseen tulevat osiot järjestelmästä ovat pisimmälle toteutetut, joten niiden testaaminen on järkevää priorisoida tässä vaiheessa tärkeimmäksi. Tästä projektista käytetään lyhennettä SH.

Projektin aikataulun vuoksi käytettävyydestä voidaan tämän tutkimuksen puitteissa tehdä vain yhdessä toteutusprintissä. Tällä testauskerralla koehenkilöt edustavat kuluttaja-asiakkaan roolia ja testaavat tyypillisimmän ja projektin tässä vaiheessa pisimmälle kehitetyn käyttötapauksen, eli uuden sähkösisopimuksen teon.

5.4.4 Elintarvike-alan yritys, HE

Tässä projektissa testataan yrityksen yhden toimialan nettisivuja, jotka koostuvat useamman eri brändin sivustoista, joiden eri osiot on suunnattu eri käyttäjryhmille. Testattava järjestelmä on valmis, mutta käytettävyydestä toteutettiin osana uudistusprojektin suunnittelua. Samlink siis tarjoaa siis asiakkaalle konsultointia uudistusprojektin suunnittelussa. Tästä projektista käytetään lyhennettä HE.

Tämän järjestelmän testaus tehdään yhteistyössä Samlinkin käytettävyyssiisiantuntijan kanssa, ja tulokset toimitetaan hänelle. Projektin aikataulu on joustava eikä työskentely ole sprinttimallista, joten testaus tehdään muiden aikataulujen niin salliessa. Testaussuunnitelman sisältää Samlinkin käytettävyyssiisiantuntijan laatimat testitapaukset, ja lisäksi käytettävyyden arvioinnin Nielsenin 10 heuristiikan suhteen. Testauksen tuloksista koostetaan raportti asiakkaalle osana uudistusprojektin suunnittelua. Testitapaukset ja sivuston testattavat osat on suunniteltu siten, että dataa saadaan kerättyä erityisesti sivuston kuluttajakäyttäjän kokemasta käytettävyydestä.

5.5 Tiedonkeruu ja analysointi

Tutkimuksen aikana kerätään monenlaista tietoa, jota analysoidaan syvemmän tietämyksen saavuttamiseksi tutkimuksen kohdealueesta ja sen selvittämiseksi, kuinka hyvin tutkimuksen tavoitteet saavutetaan. Erityisesti kerätään laajasti ja

monipuolisesti tietoa Nielsen+Krug-mallin toimivuudesta. Tutkimustavoitteiden mukaisesti kiinnostavat mallin käytön kustannukset ja hyödyt käytännön tilanteessa. *Kustannuksia* arvioidaan testauksen eri vaiheisiin kuluvalle ajalle sekä testaamisesta aiheutuvilla suorilla kustannuksilla. Mallin *hyödyllisyydestä* saadaan tietoa mittaamalla käytettävyydestä havaintojen lukumäärää ja projektiryhmien reagoitua havaintoihin. Tulosten reliabiliteetin ja validiteetin parantamiseksi oma toiminta tutkimuksen aikana pyritään dokumentoimaan mahdollisimman tarkasti. Sen tuloksena saadut pöytäkirjat tehtävistä muutoksista ja esimerkiksi dokumenttien ja raporttien eri versiot lisätään liitteeksi tähän tutkimukseen.

5.5.1 Käytettävyydestä testauksen kustannukset

Testaukseen kuuluva aika on mittarina erittäin tärkeä, koska yksi Samlinkin vaatimuksista mallille on sen soveltuvuus jopa projekteihin, joissa testausaikaa on käytettävissä vain muutamia tunteja. Silloinkin, kun resursseja on käytettävissä enemmän, on tehokkuus ensiarvoisen tärkeää, sillä aikaa on harvoin kuitenkaan paljon käytettävissä. Testauksen eri vaiheisiin lasketaan mukaan (1) testauksen valmistelu, ei kuitenkaan käytettävyydestä testauksen yleisvalmistelu, kuten koehenkilötietokannan kerääminen ja käytettävyydestä testaukseen käytettävien tietokoneiden asentaminen, (2) käytettävyydestä testausseisot ja (3) tulosten koostaminen ja analysointi, sekä niiden toimittaminen projektiryhmälle sekä projektiryhmän kysymyksiin ja tarkennuspyyntöihin vastaaminen. Koska malli halutaan pitää kevyenä, ei erikseen ajoiteta palaveria tulosten käsittelyyn, vaan projektiryhmät saavat käsitellä tulokset omaan aikatauluunsa sopivalla tavalla. Tähän käsittelyyn kuluva aika ei myöskään käytännön syistä voida seurata, ja koska projektiryhmien erilaisten tilanteiden vuoksi tulosten analysointiin kuluva ajassa on erittäin suuria eroja, sen mittaaminen ei välttämättä olisi myöskään mielekäästä. Lisäksi projektiryhmille on jätetty vapaus päättää tulosten käsittelystä vapaasti esimerkiksi osana sprintin retrospektiiviä tai päiväpalaverin jälkeen.

Muu testauksen kustannuksiin huomioitava erä on koehenkilöille annettavan palkkion arvo. Muut testauksesta mahdollisesti koituvat erät, kuten koehenkilöille tarjottavien virvokkeiden arvo tai testausseisoin muistiinpanojen koostamiseen tarvittavien muistiinpanovälineiden arvo ovat kokonaisuuden kannalta merkityksellisiä, joten niitä ei oteta mukaan tarkasteluun.

Nielsen+Krug-mallin kustannuksia pyritään analysoimaan vertaamalla niitä muiden integrointi- tai käytettävyydestä testauksen kustannuksiin ja resurssivaatimuksiin. Tätä valitettavasti vaikeuttaa se, että sekä eri integrointitapojen että testausapojen kustannuksista on saatavilla tietoja vaihtelevasti ja ne ovat peräisin eri ajoilta, jolloin muun muassa valuutan arvokin voi olla eri. Vertailua kuitenkin tehdään sikäli kuin se on mielekäästä, ja lisäksi vertaillaan tuloksia tutkimuksessa olevien projektien välillä. Yli 10 vuotta vanhat summat muunnetaan valuuksiksi nykyarvoiksi. Tieto ajankäytöstä yhdistettynä muihin kuluihin, kuten koehenkilöiden palkkioihin ja käytettyjen ohjelmistojen lisenssikuluihin, mahdollistaa tietojen jatkojalostamisen ja esimerkiksi sen selvittämisen, paljonko yksittäisen

hyväksytyyn havainnon tuottaminen lopulta maksoi. Kulut saadaan seurattua omaa toimintaa dokumentoimalla. Projekti-, käytettävyydestaustakerta-, koehenkilö- ja havaintokohtaiset hinnat lasketaan ja esitetään muodossa, joka mahdollistaa niiden vertailun myöhemmissä tutkimuksissa.

5.5.2 Käytettävyydestaustuksen hyödyllisyys

Hyödyllisyyttä arvioidaan kerätyn palautteen mukaan ja käytettävyydestaustasta tehtyjen havaintojen lukumäärän ja niiden hyödyntämistapojen perusteella. Palautetta hyödyllisyydestä kerätään (1) jokaisen käytettävyydestaustassession jälkeen koehenkilöiltä, (2) jokaisen käytettävyydestaustakerran tulosten toimittamisen jälkeen projektipäälliköiltä ja (3) tutkimuksen lopuksi kootusti kaikilta projektiryhmiltä.

Koehenkilöiltä kerättävä data on hyödyllistä lähinnä käytettävyydestaustakäytäntöjen kehittämiseksi ja sen varmistamiseksi jatkossa, että itse testaussessio on mahdollisimman sujuva. Tämä on erittäin tärkeää jo pelkästään siksi, että käytettävyydestaustajan vastuulla on koehenkilön olon saattaminen mukavaksi testauksen aikana. Dataa käytetään lisäksi mahdollisesti projektien tarpeisiin, mutta tutkimusta varten analysointia ei säännönmukaisesti tehdä.

Dataa havaintojen lukumäärästä ja tavoista, joilla ne hyödynnetään, kerätään jokaisen käytettävyydestaustakerran jälkeen pyytämällä projektipäälliköiltä havaintokohtainen palaute raportista. Tästä datasta saadaan selville testauksen tuottamien havaintojen hyöty, eli se, millä tavalla projektiryhmät näkivät tarpeelliseksi reagoida havaintoihin. Palautteen lisäksi dokumentoidaan käytettävyydestaustassa tehtyjen havaintojen määrä, tavat joilla projektiryhmät reagoivat tuloksiin, edellisestä muuttujasta johdettu tulosten hyväksyntäprosentti, sekä tavat, joilla varsinaista käytettävyydestaustaa, tulosten raportointia tai jotain muuta prosessin piirrettä kehitetään eri testauskertojen välillä. Tästä datasta ja käytettävyydestaustuksen kustannuksista kerätyistä tiedoista koostetaan tieto käytettävyydestaustuksen projekti-, sprintti-, havainto- ja koehenkilökohtaisista kustannuksista.

Tutkimuksen lopuksi suoritetaan vielä kysely projektiryhmien jäsenille heidän kokemuksistaan mallin hyödyntämisestä. Tämän *käytettävyydestaustuksen loppukyselyn* tuloksista saadaan johdettua projektiryhmien hyväksyntä hyödynnetylle käytettävyydestaustamallille. Lisäksi kyselyn avulla saadaan kartoitettua jatkokehitysideoita ja mahdollisesti jatkotutkimusaiheita. Kysely pyrittiin laatimaan siten, että siihen vastaaminen aiheuttaisi mahdollisimman vähän vaivaa projektiryhmien jäsenille, mutta tarjoaisi kuitenkin kattavasti tietoa projektiryhmien kokemuksesta käytettävyydestaustuksen hyödyllisyydestä. Jotta saataisiin selville, kuinka hyvin tutkimuksessa kehitetty käytettävyydestaustamalli toimi osana ketterää ohjelmistokehitystä, haluttiin palautetta erilaisilta projektiryhmien jäseniltä, eikä vain projektipäälliköiltä, joiden kautta valtaosa havainnoista mallin mukaan kuljetettiin. Kysely sisältää vain yhden pakollisen kysymyksen: *"Tuottiko käytettävyydestaustaus hyödyntämiskelpoisia tuloksia?"* ja vastausvaihtoehdoiksi tarjotaan *"Kyllä, paljon tai pääosin"*, *"Kyllä, jossain määrin tai jonkin verran"*

ja "Ei". Tämä oli tärkein tieto, jota haluttiin kerätä. Kysymykseen saa myös halutessaan antaa avoimen vastauksen. Lisäksi kysytään, "Oliko käytettävyydestä havaintojen hyödyntäminen helppoa?", johon pyydetään vastaus 3-portaisella luokittelulla, kuten edellisessäkin kysymyksessä. Lisäksi myös avoimen vastauksen antaminen mahdollistetaan, jotta saataisiin kerättyä mahdollisimman kattavasti laadullista dataa vastaajilta. Lopuksi kysytään vielä vastaajien kokemuksia mallin hyvistä ja huonoista puolista, jotta viimeistään saadaan esille parannuskohteita ja kritiikkiä jatkokehitystä ajatellen. Näiden lisäksi vastaajille esitetään useita, vapaaehtoisia tarkentavia tai taustatietoja keräviä kysymyksiä, kuten tieto projekteista, joissa hän on ollut mukana, hänen roolistaan niissä ja avoimia palautekenttiä.

Loppupalautekyselyn vastauksista voidaan johtaa projektiryhmien yleinen hyväksyntä käytettävyydestäusmallille. Tutkimuksen aikana kerätystä datasta koostetaan lisäksi tiedot siitä, miten käytettävyydestäusmallia kehitettiin tutkimuksen aikana, mitkä olivat projektiryhmien mielipiteet mallin toimivuudesta ja eri käytettävyydestäuskertojen havainnoista.

5.6 Testauksen toteutus

Tässä alaluvussa kuvataan tapa, jolla käytettävyydestäusta käytännössä toteutetaan. Ensin kuvaillaan tapa, jolla valittuja käytettävyydestäusmenetelmiä testataan, ennen kuin ne otetaan käyttöön. Sitten kuvataan valmistelut, jotka täytyy tehdä, ennen kuin tapaustutkimukseen kuuluva käytettävyydestäus saadaan aloitettua. Lopuksi kerrotaan alkuvaiheen koehenkilöiden rekrytoinnista.

5.6.1 Käytettävyydestäusmenetelmien testaus

Nielsen+Krug-malliin valittiin testausmenetelmiksi korttien lajittelu, heuristinen evaluointi ja yksinkertainen ääneen ajattelu. Seuraavassa luvussa kerrotaan tarkemmin menetelmistä ja niiden valinnan perusteista. Ennen kuin käytettävyydestäusessioiden suorittaminen aloitettiin, laadittua mallia kokeiltiin käytännössä osana tutkimuksen valmisteluja. Kokeiluja varten rekrytoitiin muutamia, taustoiltaan erilaisia koehenkilöitä, valmisteltiin tarkoitusta varten menetelmiksi korttien lajittelu ja yksinkertaisen ääneen ajattelun tehtävät sekä laadittiin ensimmäinen versio loppukyselystä. Testattaviksi järjestelmiksi valittiin nimeltä mainitsemattoman yhdistyksen jäsenrekisterijärjestelmä, eBay-verkkokauppa ja Concrete5-sisällönhallintajärjestelmä. Valinnat tehtiin tutkijan intuitioon luottaen. Tarkoituksena oli saada testattavaksi keskenään hyvin erilaisia järjestelmiä jotta testausmenetelmän testaus tuottaisi mahdollisimman kattavasti havaintoja erittäin pienelläkin otannalla.

Testausmenetelmien testauksen tuloksena valittujen järjestelmien yhteydessä tehtiin seuraavia havaintoja:

1. Session aikana tarvittavien tietojen on oltava erittäin selkeässä muodossa koehenkilön käytettävissä.
2. Persoonan omaksuminen oli koehenkilöiden mielestä melko helppoa, joten se lienee jatkossa turvallisin menetelmä käyttää.
3. Session aikana kirjattuihin huomioihin kannattaa liittää kellonaika, jolloin ne on tehty, ja tulkinnan helpottamiseksi kannattaa käyttää videotallenteellekin tulevaa tietokoneen kellonaikaa. Tästä seuraa, että testaajan täytyy istua siten, että hän näkee tietokoneen kellonajan huomioita tehdessään.
4. Huomioiden tekoa varten tarvitaan paperi, johon voi nopeasti kirjata vapaita huomioita, mutta myös merkitä, onnistuuko jokin tehtävä vai ei. Testitapauksen kulun kuvaava pöytäkirja toimii tarkoituksessa, jos siinä on muutoin tarpeeksi tilaa merkintöjen tekemiselle.

Testausmenetelmien kehittämistä ei lopeteta todellisten käytettävyystestausseisoiden alkaessa, vaan niitä kehitetään edelleen koehenkilöiltä ja projektiryhmiltä saadun palautteen perusteella. Tämän projekteissa tapahtuvaa käytettävyystestauksista edeltävän menetelmien testauksen tarkoituksena oli testata valittujen menetelmien yleinen soveltuvuus käytettävyystestaukseen ja varmistua siitä, että projektien puitteissa varsinaisten käytettävyystestausmenetelmien kehittämiseen ei enää tarvitse käyttää paljon resursseja.

5.6.2 Käytettävyystestauksen valmistelu

Sen jälkeen, kun käytettävyystestausmalli oli laadittu ja tiedettiin projektit, joissa mallia hyödynnetään, täytyi varsinaisen käytännön vaiheen valmistelu hoitaa. Näihin valmisteluihin kuului lisenssien hankinta tarvittavia ohjelmia varten, käytettävyystestauksista varten tarvittavien laitteiden hankinta ja asentaminen sekä koehenkilötietokannan laatiminen ja perustaminen.

Koehenkilöiden rekrytoinnista kerrotaan tarkemmin alla. Lisenssien hankkiminen oli varsin suoraviivaista, sillä käyttöjärjestelmän ja varusohjelmien lisenssit toimeksiantajayrityksellä oli tarjota jo valmiiksi, ja varsinaisten käytettävyystestausohjelmistojen ja -palveluiden lisenssit olivat ilmaisia ja helposti hankittavissa. Käytettävyystestaukseen käytettävän tietokoneen asentaminen oli sekin melko yksinkertaista, mutta ruuduntallennukseen käytettävä ohjelmisto vaati joidenkin tuntien konfigurointityön, jotta videon tallennus saatiin onnistumaan varmasti. Kutakin käytettävyystestausprojektia varten laadittiin erikseen käytettävyystestausseisoiden yhteydessä järjestettävä kysely, jonka avulla saatiin suuntaa-antava, numeerinen arvo järjestelmän käytettävyydelle. Tämän kyselyn laatimisesta kerrotaan tarkemmin luvussa 6.4.4.

5.6.3 Koehenkilöiden rekrytointi

Osana tapaustutkimusta Samlinkin käyttöön perustettiin ja ylläpidettiin tietokanta potentiaalisista koehenkilöistä. Koehenkilöt rekrytoitiin siten, että he vastaavat riittävässä määrin loppukäyttäjää. Tietokannassa olevia koehenkilöitä pyritään hyödyntämään koehenkilöinä eri projekteissa tai eri menetelmissä, ei kuitenkaan samassa projektissa ja metodissa useaan kertaan oppimisvaikutuksen välttämiseksi (vrt. Nielsen, 1995a; Rubin, 2008). Rekrytoinnit kohdistettiin yliopiston postituslistoille, sikäli kuin se on tarkoituksenmukaista, ja tuloksia täydennettiin muilla keinoilla tarvittaessa.

Aivan ensimmäisissä testeissä koehenkilöt voivat olla organisaation sisäisiä, mutta kuitenkin henkilöitä, joilla ei ole mitään tekemistä tuotteen tai projektin kanssa. Sisäiseen käyttöön tulevissa projekteissa koehenkilöinä voidaan vapaammin käyttää sisäisiä koehenkilöitä. Potentiaaliset koehenkilöt voivat täyttää omat tietonsa verkkolomakkeella tietokantaan ennen testaussessiota, jotta aikaa taustatietojen keräämiseen ei kulu enää varsinaisen session aikana. Tietokantaa varten laadittiin rekisteriseloste.

Koehenkilöiden rekrytoinnissa oltiin joustavia, mutta kuitenkin huolehdittiin siitä, että koehenkilöt ovat tarkoituksenmukaisella tavalla loppukäyttäjää vastaavia. Jos testauksen kohteena oli selvästi eri käyttäjärooleille kohdistettuja näkymiä tai muutoin tarvetta useiden käyttäjäroolien testaukseen, pyrittiin rekrytoimaan rooleja vastaavia koehenkilöitä. Roolit voidaan laatia persoonien pohjalta, tai voidaan käyttää testaussuunnitelmassa, määrittelyssä tai käyttäjätarinoissa käytettyjä rooleja tai käyttäjätyyppjä.

Rubin (2008, s. 126–131) kuvaa koehenkilöiden esitietojen keräämistä ja sopivien henkilöiden valintaa projektien koehenkilöiksi seuraavasti:

1. Tarkasta vielä profiileista haluttujen käyttäjien taustat
2. Tunnista kriteerit koehenkilöiden valintaan
3. Muodosta esitietokyselyn kysymykset
4. Järjestä kysymykset perusteltuun järjestykseen
5. Varmista kyselyn helppo läpikäynti
6. Testaa kyselyä ja paranna sitä
7. Tee lista "oikeista vastauksista" -> vain vastaajat jotka antavat oikeita vastauksia kelpaavat koehenkilöiksi.

Rekrytoinnin helpottamiseksi Samlinkin tapauksessa koehenkilöiden rekrytointi jaettiin kahteen vaiheeseen: ensin kerättiin vapaaehtoisia koehenkilöitä koehenkilökantaan, ja lopullinen rekrytointi hoidettiin sitten koehenkilökantaan annettujen vastausten perusteella.

5.7 Nielsen+Krug-mallin toimivuuden validointi

Nielsen+Krug-mallin toimivuuden Samlinkin käytössä osoittaa käytäntö. Tässä tutkimuksessa mallin toimivuutta mitataan ensisijaisesti sillä, mikä on projektiryhmien kokemus mallin hyödyllisyydestä, ja toissijaisesti myös sillä, kuinka paljon mallin sisältämien menetelmien hyödyntämiseen projekteissa tarvitaan resursseja – onhan tavoitteena mahdollisimman kevyen mallin kehittäminen. Projektiryhmien kokemusta ja mielipiteitä arvioidaan sekä testauskertakohtaisesti, jolloin mitataan erityisesti havaintojen hyödyllisyyttä yleisellä tasolla, mutta erityisesti painotetaan mielipiteitä koko tutkimuksen loppupalautekyselystä, jonka avulla pyritään keräämään palautetta koko tutkimuksen tulosten hyödyllisyydestä.

Aiemmissa tutkimuksissa käytettävyydestäusmallin tuloksia, käytännössä siis mallin hyvyttä, on validoitu ja arvioitu varsin vaihtelevin käytännöin. Chamberlain, Sharp ja Maiden (2006) tarkkailivat kenttätutkimuksessaan käyttäjakeskeisen suunnittelun ja ketterän ohjelmistokehityksen yhteensovittamista ja sen haasteita. Kolmea projektitiimiä samasta organisaatiosta seurattiin muutamman tunnin ajan viikossa kuuden kuukauden ajan. Tarkastelun tueksi järjestettiin lisäksi 10 haastattelua, joihin pyrittiin saamaan mahdollisimman laajasti erilaisia työntekijöitä eri projektiryhmistä. Tulokset koostuivat siis työntekijöiden seuraamisesta tehdyistä havainnoista, kokouksissa käsitellyistä asioista ja haastattelujen perusteella. Tutkimus tuotti havainnoinnin ja haastattelujen tuloksena ehdotuksia käyttäjakeskeisen suunnittelun ja ketterän ohjelmistokehityksen yhteensovittamisesta ja havaintoja sen suurimmista ongelmista. Näitä tuloksia ei erikseen validoitu, vaan ne perusteltiin kerätyllä aineistolla. (Chamberlain ym., 2006, s. 145–146.)

Haikara (2007) kuvaa toimintatutkimuksessaan projektia, jossa kehitettiin ketterää kehitystä hyödyntävälle organisaatiolle malli interaktiosuunnitteluprosessin laajentamiseksi persoonia hyödyntäväksi. Hän kuvaa rooliaan osallistujatarkkailijaa (engl. participant-observer) -tyyppiseksi ja mainitsee, että havaintoja kerättiin koko projektin aikana. Projektin lopussa järjestettiin lisäksi neljä haastattelua, jotta saatiin paremmin selville, mitä projektiryhmän muut jäsenet ajattelivat mallista. Empiiriset havainnot luokiteltiin ja niiden pohjalta arvioitiin mallin onnistumista eri ajankohtina kahden eri kriteerin (tiedostaminen, tyytyväisyys) suhteen (Haikara, 2007, s. 155).

Hartson, Andre ja Williges (2001) arvioivat tutkimuksessaan käytettävyyserviointimenetelmien hyvyttä ja sen tarkasteluun sopivia kriteereitä. He toteavat, että lopulta menetelmiä yhdistää vain se, että ne tuottavat listan havaintoja. Merkittävää on tämän listan havaintojen laatu sekä tuotetun listan kattavuus (engl. thoroughness), validiteetti (engl. validity) ja luotettavuus (engl. reliability) (Hartson ym., 2001, s. 28–29). Kattavuus tarkoittaa lyhykäisyydessään sitä, kuinka suuren osan järjestelmän ongelmista menetelmä kykenee löytämään. Va-

liditeetti taasen mittaa todellisten ongelmien osuutta havainnoista, ja luotettavuus sitä, kuinka hyvin menetelmä tuottaa yhteneviä tuloksia testajaista riippumatta (Hartson ym., 2001, s. 15–18,21–22).

Høegh ym. (2006) vertailevat käytettävyytestausraporttien ja käytettävyytestaussessioiden seuraamisen vaikutusta projektiryhmän jäsenten suhtautumiseen käytettävyyteen. Tutkimus koostui tapaustutkimuksesta, jossa tarkasteltiin projektiryhmän jäsenten mielipiteitä eri vaihtoehdoista, sekä kenttätutkimuksesta. Kenttätutkimuksessa tutkittiin tarkemmin käytettävyytestausraporttien piirteitä, ja se koostui varsinaisen toiminnan lisäksi kyselystä, jossa selvitettiin raportin osioiden tärkeys. Tutkimuksessa sekä raportit että sessioiden seuraaminen nähdään tärkeiksi tavoiksi sekä välittää tietoa että saada tulokset hyväksytyiksi. Raportit toivat tarkkaa tietoa tietyistä ongelmista, kun taas sessioiden seuraaminen vakuutti projektiryhmän jäsenet havaintojen merkittävydestä (Høegh ym., 2006, s. 194).

Holzinger (2005) kuvaa tutkimuksessaan käytettävyytestauksen ja -arvioinnin menetelmiä ja niiden hyviä ja huonoja puolia. Tutkimus on luonteeltaan kirjallisuuskatsaus. Tutkimuksessa alleviivataan käytettävyyden eteen tehtävän työn tärkeyttä ja painotetaan sitä, että koehenkilöiltä on ennemmin pyrittävä saamaan tietoa siitä, mitä he todella tekevät kuin mitä he arvelevat tekevänsä. Lisäksi painotetaan sitä, että käytettävyyssarviointia täytyy täydentää koehenkilöitä hyödyntävällä testauksella, ja sitä, että epäsuoraa käytettävyytestausta, kuten kyselyitä tai haastatteluita, pitäisi täydentää suoralla käytettävyytestauksella kuten ääneen ajattelu -menetelmällä (Holzinger, 2005, s. 73–74).

5.8 Yhteenveto

Tässä luvussa kuvattiin tapaustutkimuksen toteutus. Tutkimusmenetelmäksi valittiin tapaustutkimus, jonka avulla kerätään tietoa kevyen käytettävyytestauksen ja ketterän ohjelmistokehityksen yhteensovittamiseksi kehitetyn Nielsen+Krug-mallin toimivuudesta. Tutkimuksen kohteena on Jyväskyläläinen Samcom, johon viitataan emoyhtiönsä Oy Samlink Ab:n nimellä brändisyistä. Samlinkin käytössä oleva SamScrum-projektimalli ei huomioi käytettävyytestausta, ja tapaustutkimuksessa tutkitaankin Nielsen+Krug-mallin soveltamista yhteen SamScrumin kanssa. Tapaustutkimuksen tutkimusprosessi koostuu Nielsen+Krug-mallin kehittämisestä aiempaan tutkimukseen pohjaten, sen kokeilusta käytännössä ja jatkokehityksestä kokeilun havaintojen perusteella.

Mallia kokeillaan neljässä projektissa, jotka ovat keskenään hyvin erilaisia monin tavoin. Tietoa kerätään erityisesti mallin kustannuksista ja hyödyistä. Tämä tapahtuu keräämällä tieto testaukseen kuluva ajasta ja muista kustannuksista, sekä projektiryhmiltä palaute kaikista käytettävyytestauksen havainnoista ja lopulta mallin toimivuudesta yleisellä tasolla.

Tapaustutkimuksen aluksi koostetaan Nielsen+Krug-malli, testataan siihen valitut käytettävyytestausmenetelmät ja havaintojen perusteella laaditaan mie-

lekkäin tapa hyödyntää niitä osana Nielsen+Krug-mallia. Käytettävyystestausmenetelmien hyödyntämistä varten täytyy lisäksi rekrytoida koehenkilöt ja organisoida varsinaiset käytettävyystestaussessiot. Tapaustutkimuksen tulosten validointiin saatiin heikosti malleja aiemmista tutkimuksista, mutta sovellettavaksi menetelmäksi valittiin projektiryhmiltä saatava palaute. Mikäli testaus on todellakin hyödyllistä, täytyy projektiryhmiltä saatavan palautteen osoittaa se.

6 SAMSCRUM JA NIELSEN+KRUG -MALLI

Tämän luvun tarkoituksena on kuvata tässä tutkimuksessa kehitetty tapa sovittaa yhteen ketterä SamScrum-menetelmä ja Nielsen+Krug-käytettävyydestaustamalli. Ensiksi kuvataan Samlinkin käytössä olevat SamScrum-menetelmä. Toiseksi kerrotaan Samlinkin vaatimuksista käytettävyydestaustuksen suhteen ja kuvataan tässä työssä muodostettu Nielsen+Krug-malli upotettuna SamScrum-menetelmään. Tarkoituksena on, että SamScrum tarjoaa muokkausten jälkeen mahdollisuuden käytettävyydestaustuksen suorittamiseen ja käytännönläheisiä keinoja testausten toteuttamiseksi projektissa kuin projektissa. Kolmanneksi kuvataan testausmenetelmät, joita mallissa noudatetaan. Neljänneksi käsitellään konkreettisia testauskäytänteitä, joita on tarkoitus noudattaa uuden mallin yhteydessä.

6.1 SamScrum

Samlinkin toimintatapojen yhtenäistämiseksi on kehitetty yrityksen oma projektimalli, SamScrum, jolla Samlinkin Ratkaisuliiketoiminta-yksikön projektit toteutetaan yhä enenevässä määrin (Litmanen, 2013b). Projektimallin kehitys aloitettiin vuoden 2012 aikana, ja yksi virstanpylväs saavutettiin vuoden 2012 syksyllä, kun ensimmäiset mallia noudattavat projektit aloitettiin (Litmanen, 2013b, s. 3). SamScrum, kuten esikuvansa scrum (Schwaber & Sutherland, 2013, s. 4), ei ota kantaa käytettävyydestaustuksen toteuttamiseen, mutta kuvailee erityisesti sprintin jälkeisen testausten suoritusta. Kuten aiemmin tarkastellussa kirjallisuudessa (Crispin & Gregory, 2009; McInerney & Maurer, 2005; Northrop, 2011), myös SamScrumin tapauksessa testausten tulokset syötetään ensisijaisesti seuraavien sprinttien tehtävälistaan sen sijaan, että niihin reagoitaisiin välittömästi.

Käytäntö on osoittanut, että useimmat menestyvät yritykset eivät ole sännäneet ottamaan ketteriä menetelmiä käyttöön suin päin, vaan ovat ennemminkin soveltaneet niitä soveltuvien osien ja aikaisempaan organisaatiokulttuuriinsa

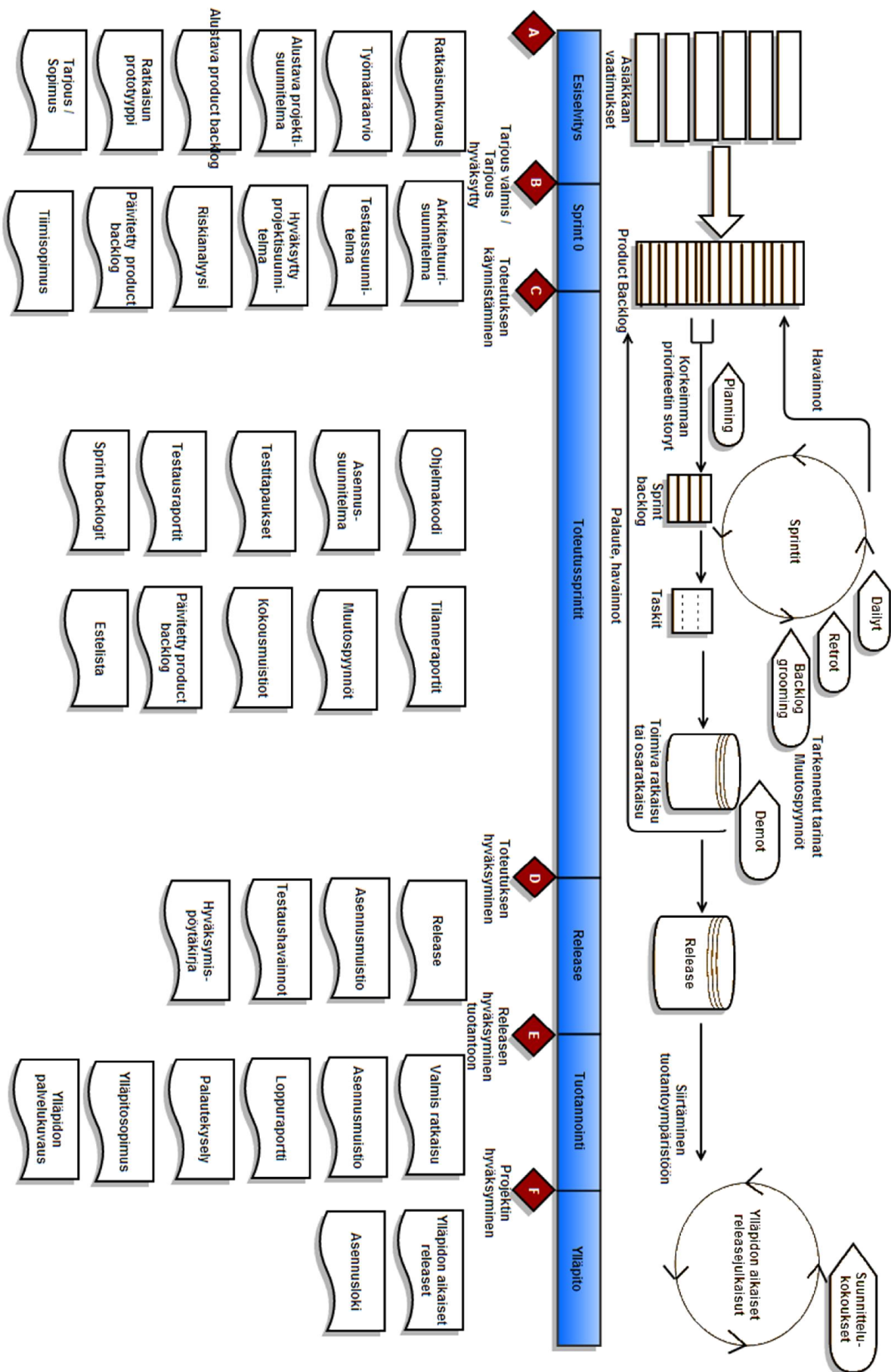
pohjaten (Bygstad ym., 2008, s. 375). Samlinkin tapauksessa mallia on täydennetty erityisesti kuvauksella yrityksen käyttöön tulevien tuotosten kuvauksella. Projektimallin täydentäminen erilaisia käytettävyydestausmalleja tarjoavalla osiolla on yksi tutkimuksen päätavoitteista.

Kuvio 9 kuvaa Samlinkin käytössä olevan projektimallin pääpiirteissään. Malli perustuu scrumiin ja yleisiin ketteriin käytäntöihin, mutta siinä on määritetty projektin mahdolliset tuotokset, kuten dokumentaatio ja prototyypit, sekä suoritettavat aktiviteetit tarkemmin kuin ”tavallisessa” scrumissa (vertaa kuvio 1). Se sisältää tavoite- ja kattotyömäärät ja niitä vastaavat hinnat, ja lisäksi siihen on lisätty Samlinkin sisäisessä termistössä määritellyt projektin virstanpylväät (engl. milestones).

SamScrum-projektimallissa esiselvitysvaiheessa, ennen varsinaisen projektin alkua, asiakkaan vaatimusten perusteella laaditaan mahdollisesti ratkaisukuvaus, työmääräarvio, alustava projektisuunnitelma, prototyyppi ratkaisusta ja joka tapauksessa sopimus, mahdollisesti ensi vaiheessa tarjouksena. Mallin ensimmäinen virstanpylväs on tarjouksen hyväksyntä, josta seuraa toteutusta valmisteleva 0-sprintti. Tämän sprintin aikana valmistellaan ensimmäinen versio tuotteen työlistasta ja mahdollisesti sellaisia dokumentteja kuin arkkitehtuuri-suunnitelma, testaussuunnitelma, tässä vaiheessa jo hyväksytty projektisuunnitelma, projektin riskianalyysi ja tiimisopimus.

Projektin toinen virstanpylväs käynnistää jo varsinaisen toteutuksen, jonka aikana noudatetaan tavanomaisia scrumin käytänteitä ja joka sisältää useita scrumin mukaisia sprinttejä. Näiden sprinttien aikana prioriteetiltaan suurimmat käyttäjätarinat poimitaan sprinttien työlistalle, käyttäjätarinat muutetaan pienemmiksi tehtäviksi (engl. task), jotka sitten jaetaan kehittäjien kesken ja toteutetaan. Sprinttien aikana voidaan järjestää päiväpalavereita, retrospektiivi ja työlistan siistimispalaveri. Sprintin aikainen testaus tuottaa havaintoja suoraan tuotteen työlistalle. Aina sprintin lopussa tuotetaan toimiva ratkaisu tai osaratkaisu, josta järjestetään yksi tai useampia havaintoesityksiä (”demo”) asiakkaalle. Viimeistään tämän esityksen seurauksena asiakkaalta saadaan palautetta, jonka perusteella voidaan tehdä uusia kohteita tuotteen työlistalle, tarkentaa käyttäjätarinoita tai tehdä muutospyyntöjä. Toteutusprinttien aikana voidaan tuottaa ainakin seuraavanlaista dokumentaatiota: varsinaista ohjelmakoodia, asennussuunnitelma, testitapaukset, testausraportit, sprinttien työlistat, projektin tilannereportit muutospyyntöt, kokousmuistiot, tuotteen työlistan päivitetty versio ja projektin estelista.

Viimeistä toteutusprinttiä seuraa virstanpylväs *Toteutuksen hyväksyminen*. Sen seurauksena sovelluksen julkaistulle versiolle toteutetaan hyväksyntätestaus, jonka seurauksena julkaisu joko viedään tuotantoon tai palautetaan toteutusprintteihin. Tämän vaiheen tuotoksia voivat olla varsinainen julkaistu ohjelmisto, asennusmuistio, testaushavainnot ja hyväksymispöytäkirja.



KUVIO 9 SamScrum-projektimalli (Litmanen, 2013a)

Seuraavan virstanpylvään, eli tuotantoon hyväksymisen, jälkeen seuraa tuotantovaihe, jossa julkaistu ohjelmistoversio siirretään tuotantoympäristöön. Tämän vaiheen tuotoksia voivat olla valmis ratkaisu, asennusmuistio, loppuraportti, palautekysely, ylläpitosopimus ja ylläpidon palvelukuvaus. Tätä vaihetta seuraa viimeinen projektin virstanpylväs, projektin hyväksyminen. Viimeinen projektimallin vaihe on ylläpito, jonka aikana työskentely on iteratiivista koostuen sprinteistä tai sprinttimäisestä työskentelystä, jonka ohjaamiseksi järjestetään suunnittelukokouksia ja jonka tuloksena saadaan uusia julkaisuja. Tässä vaiheessa tuotoksia voivat olla ylläpidon aikaiset julkaisut ja asennusloki. (Litmanen, 2013a)

SamScrum-projektimallissa on määritelty eri vaiheissa tuotettava dokumentaatio. Tämä määrittely on kuitenkin lähinnä ohjeellinen ja kuvaa, mitä kaikkea malli täydellisesti suoritettuna tuottaa. Projektimalli muistuttaa, kuten kuvista 9 voi nähdä, pääpiirteiltään hyvin pitkälti normaalia scrumia. Mallista on syytä mainita sprintti 0, jonka aikana laaditaan suunnitelmia tai niiden pohjia, usein sen perusteella mitä asiakas vaatii. Tärkein tuotos lienee hyväksytty projektisuunnitelma, joka määrittää paljolti muun muassa sen, mistä aktiviteeteista asiakas on valmis projektin aikana maksamaan, ja muun muassa sen, kuuluuko käytettävyytestaus projektiin lainkaan. Tässä tutkimuksessa suurin huomio keskittyy toimitussprinttien aikaiseen työhön, sillä niiden aikana muodostuvat ne artefaktit, joihin käytettävyytestausta on mielekkäintä kohdistaa.

Malli ei ota kantaa käytettävyyteen, sen suunnitteluun tai testaamiseen, vaan toimintatavat ja projektin lopputuotteiden käytettävyyden taso on projektiryhmän itse päätettävissä. Tällä hetkellä kukin projektiryhmä huomioi käytettävyyden oma-aloitteisesti ja joustavasti siten kuin se kulloinkin luontevimmin onnistuu. Käyttöliittymäsuunnittelijat, ohjelmistokehittäjät, testaajat ja asiakkaan edustajat huomioivat käytettävyyden omassa roolissaan muiden tehtäviensä ohella. Käytännössä näin joustava toimintapa johtaa kuitenkin hyvin vaihtelevaan lopputuotteiden laatuun. Suunnittelutehtävissä käytettävyys yleensä huomioidaan, sillä suunnittelu tehdään määrittelyn pohjalta ja määrittelyssä käytettävyydelle on yleensä asetettu vaatimuksia, mutta toteutuksen ja testauksen osalta käytettävyys jää usein hyvin pienelle huomiolle. Usein käytettävyys tuleekin vastaan vasta hyväksyntätestausvaiheessa, tai mikä pahinta, tuotantoon viennin jälkeen. Hyväksyntätestauksen suorittaa asiakas, mutta valitettavasti asiakkaan edustaja ei useinkaan vastaa tosiasiallista käyttäjää (Jokela, 2010, s. 72).

6.2 Käytettävyytestausmalli integroituna SamScrumiin

Tässä alaluvussa tarkastellaan aiempiin lukuihin pohjautuen valittuja keinoja käytettävyytestausmallin yhteensovittamiseksi osaksi SamScrumia. Ensiksi määritellään vaatimuksia Sanlinkin käytettävyytestausmallille. Toiseksi kuvataan kyseinen malli, jota kutsutaan Nielsen+Krug-malliksi. Eri tavoista valitaan Samlinkille sopivimmat, valinnat perustellaan ja mietitään niiden vaihtoehtoja.

6.2.1 Vaatimukset Samlinkin käytettävyydestestausmallille

Samlinkin projektimalli on kevyt, eikä siihen haluta liittää paljon etukäteissuunnittelua. Tarjous- ja määrittelyvaiheessa käytettävyys on toki huomioitu osana muuta suunnittelua. Tavoitteena on löytää nimenomaan testausprosessia tukeva malli, joka voidaan suorittaa erittäin vähilläkin resursseilla ja jonka voi suorittaa vähällä osaamisella. Samlinkin tarpeisiin tulevalle käytettävyydestestausmallille on keskusteluissa esimiesteni ja mallin kehittäjien kanssa johdettu seuraavat vaatimukset:

1. Mallin tulee olla sovellettavissa myös projekteissa, joissa käytettävyydestestaukseen on uhrattavissa vain muutamia tunteja koko projektin aikana.
2. Mallin on toimittava saumattomassa yhteistyössä SamScrum-mallin kanssa, esimerkiksi integroituna osaksi iteratiivista työtapaa.
3. Käytettävyydestestauksen tulokset on saatettava nopeasti ja luontevasti projektiryhmän hyödynnettäviksi.
4. Työkalut eivät saa olla kalliita, joskaan niiden ei tarvitse myöskään olla ilmaisia.

Ensimmäinen vaatimus on olennaisin. Resurssit projekteissa ovat jatkuvasti vähissä, ja työmääräarvioihin on vaikea saada mahtumaan rajaansa enempää testausta, saati sitten erityisesti käytettävyydestestausta. Mallin tulee tarjota työkaluja käytettävyydestestaukseen erilaisissa projekteissa, joissa resursseja on käytettävissä eri määrä, myös vain muutamia tunteja.

Toinen vaatimus määrittää, että käytettävyydestestauksen menetelmät on saatava sovitettua Samlinkilla yleisesti käytettävään projektimalliin, SamScrumiin. Tämä on ensiarvoisen tärkeää, jotta eri projektiryhmille saadaan tarjottua käytettävyydestestauksen suorittamiseen tapoja, joita on helppo hyödyntää osana projektityöskentelyä.

Kolmas vaatimus, tulosten nopea projektiryhmälle toimittaminen, pohjautuu rajallisiin resursseihin ja ainaiseen kiireiseen kuten ensimmäinenkin vaatimus. Neljäs vaatimus on vähemmän pakottava, ja on lähinnä yksi esimerkki rajallisista resursseista: yhteen, varsin spesifiin käyttötarkoitukseen ei aina ole varaa ostaa kallista työkalua. Tämänkin takia valittiin kevyet käytettävyydestestausmenetelmät kalliin käytettävyydestestauslaboratorion sijasta.

Marty ja Twidale (2005, s. 6) ovat tutkineet pienintä mahdollista testauksen määrää, ja siten myös työmäärää, jolla voidaan saada varmasti joitakin tuloksia. Thomas (1996, s. 113) kuvaa tämän olevan raja, jolloin ”pikainen ja likainen” testaus muuttuu ”nopeaksi ja saastaiseksi” (engl. ”fast and filthy”), eli menetelmän keveyden tavoittelussa mennään liian pitkälle. Martyn ja Twidalen (2005) mukaan tällaisessa käytettävyydestestauksessa piilee kaksi haastetta: (1) vähäisimmän mahdollisen ajan käyttämiseen hyödyllisten tulosten saamiseksi ja (2) vain yhdeltä testikerralta saatujen tulosten luotettavuus. Samlinkille olennaista on löytää keinoja käytettävyydestestauksen toteuttamiseen erilaisilla käytettävissä olevilla

resursseilla, ja toisaalta pystyä ennustamaan tietyn tyyppisten käytettävyydestestausmenetelmien käyttöön vaadittavat resurssit, jotta esimerkiksi työmääräarviot saadaan tehtyä mahdollisimman tarkasti.

Joitakin arvioita käytettävyydestestaukseen tarvittavasta ajasta ja muista resursseista voidaan löytää aiemmista integrointikokemuksista. Varsin keveitä menetelmiä hyödyntävä Krug (2000, 2006) kuvaa valmistelun ja testien organisoinnin jälkeen testaukseen kuluneen vain yhden päivän: aamupäivän aikana suoritetaan varsinainen testaus, ja lounastauolla tai iltapäivän aikana käsitellään tulokset. Hiukan raskassoutuisempaa menettelytapaa hyödyntää McGinnin ja Changin (2013) kehittämä RITE+Krug -menetelmä, jossa testaus suoritetaan kahden päivän aikana, jolloin toisena päivänä tulokset voidaan jo käsitellä projektiryhmän kanssa. Hyvin käytännönläheisiä esimerkkejä löytyy muitakin. Northrop (2011, 2012) arvioi yhden käytettävyydestestausiteraation vievän keskimäärin noin 30 tuntia, joskin hän painottaa tuntimäärän vaihtelevan suuresti eri projektien välillä riippuen esimerkiksi siitä, montako koehenkilöä halutaan testata. Hänen mukaansa testausiteraation valmisteluun kuluu noin 15 tuntia, varsinaiseen testaukseen 10 tuntia ja tulosten analysointiin 5 tuntia. Koko sprintin ajaksi hän suosittelee varattavan 40 tuntia käytettävyydestestausta varten, ja käytettävyydestestausta tehtäväksi kerran sprinttiä kohden (Northrop, 2011).

6.2.2 Nielsen+Krug-malli

Käytettävyydestestauksella laajennettu SamScrum-malli työstettiin hyödyntämällä aiempaa tutkimusta erityisesti kevyistä käytettävyydestestausmenetelmistä (Krug, 2000, 2006; McGinn & Chang, 2013; Nielsen, 1992a, 1994b, 1995a, 2000) ja käytettävyydestestauksen ja ketterän ohjelmistokehityksen yhteensovittamisesta (Lee ym., 2009; Lee & McCrickard, 2007; McGinn & Chang, 2013; Northrop, 2011, 2012; Parsons ym., 2007; Singh, 2008; Sy, 2007). Tulos on nähtävissä kuviossa 10.

Laajennettuun SamScrum-malliin on lisätty toimitussprinttien aikaiseksi aktiviteetiksi *käytettävyydestestaus*, jonka havainnot viedään *tärkeyden arvioinnin* kautta projektin työlistalle. Toteutusprinttien tuotoksiksi on lisätty *käytettävyydestestitapaukset*, jotka pohjautuvat käyttäjätarinoihin tai testitapauksiin, ja *käytettävyydestestausraportit*. Julkaisukohtaisiin tuotoksiin kuuluu *käytettävyydestestauksen yhteenveto*, jossa kuvataan lyhyesti havainnot ja niihin reagointi. Kuvioista puuttuu käytettävyydestestaussuunnitelma, koska sen katsotaan olevan osa testaussuunnitelmaa, joka mallissa on jo valmiiksi.

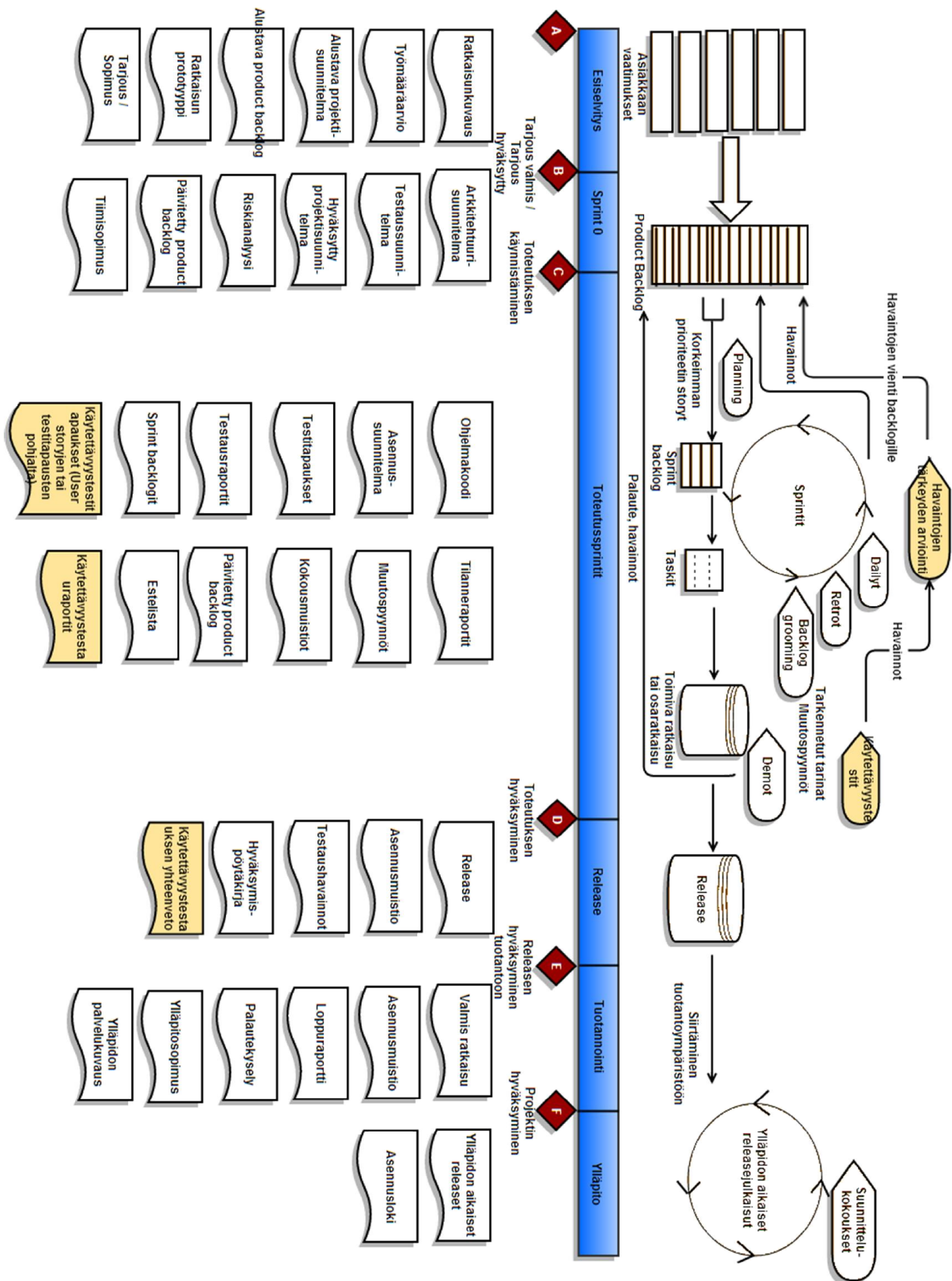
Malliin lisätyt osiot on korostettu kuviossa vaalean keltaisella värillä. Mallin lopullisessa dokumentaatiossa käytettävyyssaktiviteettien ja -tuotosten värjämiselle muista erottuviksi ei ole syytä, mutta tässä vaiheessa se on tehty kuvion selventämiseksi. Käytettävyydestestauksen aktiviteettien toteuttaminen projektissa ei ole pakollista, kuten ei ole kuvatun dokumentaationkaan tuottaminen. SamScrum ei ole mallina tarkoitettu käyttäjiään kahlitsevaksi, joten muutkin tuotokset ja aktiviteetit ovat kuviossa enemmän ohjeellisia kuin pakollisia.

Esitetty malli integraatiosta pohjautuu erityisesti aiemmin esiteltyyn RITE+Krug-malliin (McGinn & Chang, 2013, s. 61–63), mutta myös muista integrointitavoista on ammennettu ohjeita ja kokemuksia. Lisäksi mallia varaudutaan päivittämään ja kehittämään myös tutkimuksen käytännön vaiheen aikana tehtävien havaintojen perusteella.

RITE+Krug-mallissa keskitytään edullisiin ja ketteriin käytettävyydestä menetelmiin, joissa koehenkilöitä on minimaalinen määrä, mutta projektin osapuolilta vaaditaan osallistumista testisessioihin, ja se vie aikaa. McGinn ja Chang (2013) toisaalta järjestivät sessioiden valvonnan verkkokonferenssina, jossa osallistujat pystyivät seuraamaan session etenemistä omilta koneiltaan. Erityisesti tällaisen mahdollisuuden tarjoaminen voi olla arvokasta. Samlinkin mallissa varaa osapuolien osallistumisen vaatimiseen ei ole, mutta siihen tarjotaan tietenkin mahdollisuus. Toisaalta RITE+Krug -mallin kenties suurin vahvuus on se, että kun käytettävyysongelmat saatetaan sessioiden kautta sidosryhmien tietoon ja niiden ratkaisuvaihtoehdot käydään koko projektiryhmän kanssa läpi, saadaan niiden olemassaolo tunnustettua ja projektiryhmä sitoutettua ratkaisemaan ne (McGinn & Chang, 2013, s. 66).

Samlinkille kehitettyä mallia kutsutaan McGinn ja Changin esimerkkiä (2013) noudattaen *Nielsen+Krug-malliksi*, sillä siinä yhdistetään Nielsenin edullisen käytettävyydestäuksen ja Krugin menetelmiä, jotta saadaan aikaiseksi joukko suosituksia ja ohjeita projektin tyyppin ja käytettävissä olevien resurssien perusteellaärkevimpien testausmallien hyödyntämiseen. Tärkeimpinä Nielsen+Krug-mallin menetelminä pidetään tässä korttilajittelua ja ääneen ajattelu -testaussessioita. Samlinkin käyttöön näiden lisäksi otetaan myös heuristinen arviointi, joka on mahdollista toteuttaa ilman koehenkilöiden rekrytointia.

Mallin selkärangana ovat yksinkertaista ääneen ajattelu menetelmän hyödyntävät käytettävyydestäussessiot, joita yleensä ottaen eri integrointitavoissa pidetään merkittävimpänä käytettävyydestäuksen menetelmänä (Holzinger, 2005; Lee ym., 2009). Tilanteisiin, joissa varsinaiseen käytettävyydestäuksen järjestämiseen ei ole resursseja tai joissa on resursseja sekä varsinaisen käytettävyydestäuksen että käytettävyyssarvioinnin tekoon, tarjotaan ohjeet heuristisen arvioinnin toteuttamiseen. Tämä lista tarjotaan muodossa, joka mahdollistaa tulosten helpon raportoinnin riittävällä tarkkuudella ja sisältää tiedon noudatettavista heuristiikoista (Nielsen & Molich, 1990; Nielsen, 1994b, 1995b). Heuristinen arviointi voi lisäksi sopia suoritettavaksi esimerkiksi järjestelmään tutustumisen yhteyteen, kun esimerkiksi käyttötapausten pohjalta laaditaan käytettävyydestäussuunnitelmaa. Tarvittaessa voidaan tehdä myös korttien lajittelu, erityisesti yksinkertaisen ääneen ajattelun yhteydessä. Samoja koehenkilöitä voidaan hyödyntää sekä korttien lajittelussa että yksinkertaisessa ääneen ajattelussa. Hyödynnettävät menetelmät on kuvattu tarkemmin seuraavassa alaluvussa.



KUVIO 10 SamScrum-malli, johon on sovitettu kevyt käytettävyysestaus

Samlinkin mallissa testausta voidaan tehdä sekä projektien toteutuksen aikana, että valmiille järjestelmälle. Mikäli toteutus on vielä kesken, kunkin sprintin aikana tehdyn käytettävyystestauksen havainnot syötetään arvioinnin jälkeen lähitökohtaisesti seuraavan sprintin tehtävälisterille (engl. sprint backlog) (kuten mm. McInerney & Maurer, 2005, s. 23). Testauskertojen välille jätetään niin paljon aikaa, että aiemman testauskerran huomiot on voitu käydä läpi ja jos tarpeen, korjata.

6.3 Testausmenetelmien kuvaus

Tässä alaluvussa kuvaillaan lyhyesti SamScrumiin yhteen sovitettavien käytettävyystestausmenetelmien piirteitä. Nämä menetelmät ovat korttien lajittelu, heuristinen arviointi ja yksinkertainen ääneen ajattelu. Ne kuuluvat ns. Nielsenin (1994, 1995a) edullisiin käytettävyystestauksen metodeihin, ja Sohaib ja Khan (2011, s. 6) nostavat ne esille integrointimallissaan. Menetelmät on valittu eri tutkijoiden raporttoimien hyvien kokemusten (Jørgensen, 1989; Nielsen, 1994b, 1995a) ja niiden yleisen hyväksynnän (Vredenburg, Mao, Smith, & Carey, 2002) takia sekä sen perusteella, että ne tukevat hyvin toisiaan. Korttien lajittelu voidaan yhdistää muihin menetelmiin, ja heuristinen arviointi tukee hyvin yksinkertaistettua ääneen ajattelua (Frøkjær & Larusdottir, 1999). Kaikista merkittävimpänä menetelmänä pidetään yksinkertaista ääneen ajattelua. Se on menetelmistä ainoa, jolla saadaan järjestelmän käyttäjien realistisia kommentteja ja jossa käyttäjiä (tai näiden edustajia) voidaan tarkastella verraten luonnollisessa tilanteessa. Se on myös menetelmä, jota kohtaan Samlinkin projektipäälliköt ovat osoittaneet suurinta kiinnostusta, ja siksi siihen keskitytään tässä eniten. Tässä alaluvussa kuvataan käytettävyystestausmenetelmien kulku erityisesti Samlinkin tapauksessa ja SamScrumiin liitettynä, ja kerrotaan valinnoista ja rajoituksista, joita tutkimuksessa on tehty Samlinkille toimitettavaa mallia varten.

6.3.1 Korttien lajittelu ja sen järjestäminen

Korttien lajittelun kuvaus yleisellä tasolla löytyy jo aiempaa luvusta 3.3.2. Tässä yhteydessä menetelmä kuvataan yksityiskohtaisemmin tavalla, jolla sitä hyödynnetään Nielsen+Krug-mallissa. Menetelmällä pyritään saamaan selville käyttäjien ”mentaalinen malli” järjestelmän informaatioavaruudesta (Ghanam & Maurer, 2007; Nielsen, 1995a, s. 98). Korttien lajittelumenetelmiksi on vakiintunut kaksi erilaista: avoin ja suljettu lajittelu. Menetelmät eroavat toisistaan hiukan koehenkilön tehtävän osalta: avoimessa korttien lajittelussa koehenkilölle annetaan kasa kortteja, ja pyydetään ryhmittelemään ne ja lopuksi nimeämään ryhmät. Suljetussa lajittelussa koehenkilölle on annettu kategoriat tai ryhmät valmiiksi, ja häntä pyydetään lajittelemaan kortit niihin. Puolisuljetussa korttien lajittelussa koehenkilö saa valmiit ryhmät tai kategoriat, mutta voi halutessaan

tehdä niihin muutoksia. (Foraker Labs, 2010; Kane, 2003; Nielsen, 1995a; Rubin, 2008, s. 18.)

Korttien lajittelussa koehenkilö saa eteensä termejä ja hierarkian kohteita ja saa tehtäväkseen lajitella ne mielestään loogisiin kokonaisuuksiin (Foraker Labs, 2010). Tyypillisimmät käyttötarkoitukset Samlinkin projekteissa ovat termistön organisointi (esimerkiksi SharePoint-intranetin termijoukot), sivustorakenteen ja näkymien järjestyksen selvittäminen sekä navigaation tai sivustohierarkian loogisimman rakenteen organisointi. Moniportainen sivustorakenne voi olla syytä selvittää vaiheissa.

Nielsen+Krug-mallissa session järjestäjä (käytettävyystestaaaja) saa lajiteltavat termit tai hierarkian osiot projektiryhmältä tai koostaa ne itse aiemman materiaalin pohjalta, laatii niistä online-testin ja testaa sen itse tai sisäisen koehenkilön kanssa. Seuraavaksi valitaan tai määritellään koehenkilöt, ellei testausta tehdä varsinaisen käytettävyystestaussession ohessa, ja kutsutaan heidät osallistumaan testiin. Lopuksi koostetaan tulosten pohjalta esitys hierarkiasta tai termistöstä.

Menetelmä vie aikaa noin kymmenestä viiteentoista minuuttia per koehenkilö, jopa vähemmän verkossa toteutettuna, kunhan järjestelmä on helppokäyttöinen. Session valmistelussa aikaa kuluu kokeen laajuudesta riippuen vartista pariin tuntiin. Jos hierarkia tai termistö koostuu termeistä, jotka vaativat kohdealueosaamista, on huomioitava, että koehenkilöiden täytyy ymmärtää termit – heiltä täytyy siis löytyä ainakin jonkin verran kohdealueosaamista. Koehenkilöllä ei tarvitse välttämättä olla kovin syvää kohdealueosaamista, kunhan hän ymmärtää termit. Tiivistetysti voidaan sanoa, että mitä enemmän koehenkilöitä käytetään, sen parempi. Korttien lajittelun kohdalla kokeen toistamisella ei ole yhtä suurta merkitystä kuin erityisesti käytettävyystestaussessioiden toistamisella, mutta toisaalta myöskään oppimisvaikutus ei ole yhtä voimakas, joten samoja koehenkilöitä voidaan käyttää jonkin verran uudestaan.

6.3.2 Heuristinen arviointi ja sen suorittaminen

Heuristinen arviointi (Nielsen, 1995c; Rubin, 2008) menetelmänä on jo kuvattu aiemmin luvussa 3.3.2. Nielsenin kymmenen käytettävyyshauristiikkaa on riittävän laaja, tunnettu ja läpikotaisin koeteltu heuristiikka, jotta uskallamme ottaa sen käyttöön Samlinkin kevyessä käytettävyystestaussmallissa. Läpikäyntejä varten varsinaisen heuristiikan tueksi otetaan jokin niitä selittävä ja esimerkkejä tarjoava lähde, kuten *Designing Web Interfaces - Principles and Patterns for Rich Interaction* – kirjan tueksi kirjoitetun web-sivuston (Nell, 2009) kuvallisilla esimerkeillä varustettu täydentävä listaus kustakin Nielsenin (1995b) heuristiikasta.

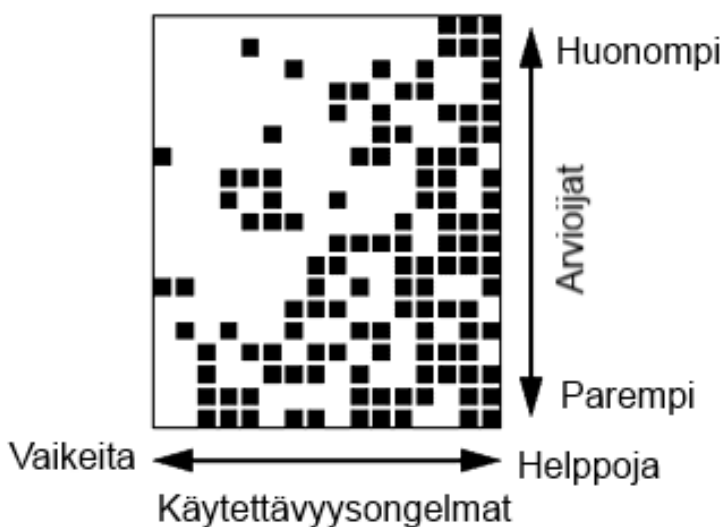
Taulukko 6 kuvaa Nielsenin (1995b) heuristiikat täydennettynä niiden selkokielisellä kuvauksella, joiden muotoilussa on käytetty apuna Nummiahon (2003) suomennoksia. Tässä muodossa heuristiikkoja voidaan käyttää sekä heuristisen evaluoinnin tukena että yksinkertaisen ääneen ajattelun havaintojen ryhmittelyssä.

TAULUKKO 6 Nielsenin heuristiikat täydennettynä kuvauksilla (Nielsen, 1995b)

Heuristiikan nimi	Heuristiikan kuvaus
Järjestelmän tilan näkyvyys (palaute käyttäjälle)	Käyttäjän pitäisi aina saada palautetta kaikista tekemistään komennoista ja valinnoista. Mikäli käyttäjä ei mitenkään havaitse, että järjestelmä tekee/teki jotain, olettaa tämä luultavasti, että jotain on pielessä, ja esimerkiksi yrittää samaa komentoa turhaan uudelleen. Välitön palaute antaa käyttäjälle tunteen järjestelmän hallinnasta ja ohjauksesta ja tehostaa oppimista. Vähintään yli kymmenen sekuntia kestävästä toiminnosta pitäisi antaa käyttäjälle arvio sen kestosta (esimerkiksi täytyvällä palkilla).
Käytä käyttäjien omaa kieltä	Termien, käsitteiden ja kuvakkeiden tulisi olla tuttuja käyttäjälle ja niiden pitäisi tarkoittaa sitä, mitä käyttäjä odottaa niiden tarkoittavan. Tieto tulisi esittää luonnollisessa ja loogisessa järjestyksessä.
Käyttäjän kontrolli ja vapaus	Jotta käyttäjä uskaltaisi kokeilla eri toimintoja, tulisi joka toiminnosta tarjota mahdollisuus palata edelliseen tilaan (kumoa / tee uudelleen, peruuta) tai jopa pois koko järjestelmästä. Jos jollekin toiminnolle, kuten tiedoston poistolle, ei pystytä tarjoamaan kumoavaa toimintoa, tulisi käyttäjälle kertoa tästä ja pyytää vahvistusta toiminnon suorittamiseen.
Yhtenäisyys ja standardien noudattaminen	Tietyn komennon tai valinnan tulisi toimia yhdenmukaisesti kaikissa järjestelmän osissa. Myös käyttöliittymän osien sijoittelun tulisi olla yhdenmukaista eri näytöissä.
Virhetilanteiden ehkäisy	Jopa selkeitä virheilmoituksia tärkeämpää on suunnitella käyttöliittymä niin, että käyttäjälle näkyviä virhetilanteita syntyy mahdollisimman vähän.
Muistikuorman vähentäminen (tunnistaminen muistamisen sijasta)	Tietojen muistaminen on syytä jättää koneelle, ei käyttäjälle. Jo kertaalleen annetun tiedon muistamista on esimerkiksi turha vaatia järjestelmän muissa osissa. Koska ihminen tunnistaa asioita paljon helpommin kuin muistaa ulkoa, pitäisi käyttäjälle tarjota mahdollisuuksien mukaan mieluiten vaihtoehtoja, joista valita tai ainakin malli- tai oletusarvoisia vastauksia.
Käytön joustavuus ja tehokkuus	Kokenut käyttäjä kokee usein tarvetta järjestelmän käytön tehostamiseen erilaisilla oikopoluilla. Tyypillisiä oikopolkuja ovat komentojen suorittaminen hiiren eri nappuloiden tai näppäinyhdistelmien avulla. Oikopolut eivät saa olla ainoa keino tietyn toiminnon suorittamiseen, vaan niiden tulee olla vaihtoehtona, jollekin helppokäyttöiselle ja selkeästi merkitylle keinolle (esimerkiksi valikosta valitsemiselle). Järjestelmän käyttöä voidaan tehostaa myös tarjoamalla käyttäjälle oletusarvoja.
Suunnittelun estetiikka ja minimalistisuus	Käyttöliittymässä ei saisi olla turhaa tietoa. Harvoin käytettävä tieto tulisi piilottaa esimerkiksi erilliseen ikkunaan tai alivalikkoon. Osat pitäisi ryhmitellä niin, että ne helpottavat asioiden löytämistä. Käyttäjälle tulisi tarjota selkeä etenemistapa asioiden tekoon, mutta mahdollistaa myös vaihtoehtoisia etenemistapoja.
Käyttäjien tukeminen virheiden tunnistamisessa ja niistä palautumisessa	Virheilmoitukset pitäisi esittää käyttäjän ymmärtämällä kielellä, niiden tulisi selkeästi ja kohteliaasti kertoa mikä on ongelmana ja ohjata käyttäjää korjaamaan virheen.
Käyttäjän apu ja tuki, sekä dokumentaatio	Paras vaihtoehto olisi, jos järjestelmää pystyisi käyttämään ilman erillisiä ohjeita. Tähän kuitenkin päästään vain harvoin. Järjestelmän pitäisi pystyä tarjoamaan käyttäjälle apua juuri siitä käyttöliittymän osasta, jota käyttäjä sillä hetkellä käyttää (kontekstisidonnaisuus). Dokumentaation pitäisi olla jäsennelty niin, että asiat löytyvät helposti ja nopeasti. Käyttäjälle pitäisi konkreettisesti kertoa, mitä tämän tulee tehdä tietyn tehtävän suorittamiseksi.

Edellä kuvatulla tavalla toteutettuna heuristinen evaluointi muistuttaa Jeffriesin ym. (1991) esittelemien heuristisen evaluoinnin ja käytettävyysohjeiden arvioinnin yhdistelmää, jonka etuina on käytettävyysohjeiden arvioinnin helppous ja heuristisen evaluoinnin hyötysuhde.

Nielsenin (1995c) mukaan heuristinen arviointi kestää yleensä yhdestä kahteen tuntiin, ja mikäli järjestelmä on niin monimutkainen, ettei sen arviointi kestä pidempään, on arviointi syytä jakaa useampaan osaan. Mahdollisimman kattavien havaintojen keräämiseksi arviointia olisi syytä tehdä useamman arvioijan - kuvio 11 havainnollistaa eri käytettävyyсарvioijien kykyä löytää eri havaintoja järjestelmästä. Paraskaan arvioija ei löytänyt kahta vaikeimmaksi arvioitua ongelmaa, eikä kaikkia helpompiaakaan ongelmia, ja toisaalta huonoin arvioija löysi muutaman helpoimmista ongelmista. Toisaalta paras hinta-laatu-suhde saatiin neljällä arvioijalla, joka Nielsenin mukaan sopii teoriaan siitä, että optimaalinen määrä arvioijia on kolmesta viiteen (Nielsen, 1995c). Jos resursseja ei ole käytettävyydestä koehenkilöillä, eikä edes useiden henkilöiden suorittamaan käytettävyyсарviointiin, on yhdenkin henkilön kerran suorittama käytettävyyсарviointi parempi kuin ei mitään. Heuristinen arviointi ei myöskään vaadi paljon resursseja, vaan lähinnä ammattitaitoa ja herkkyyttä arvioijalta.



KUVIO 11 Käytettävyysongelmien löytyminen heuristisessa arvioinnissa (Nielsen, 1995c)

6.3.3 Yksinkertaistettu ääneen ajattelu

Yksinkertaista ääneen ajattelua hyödyntävä käytettävyystestaussessio suoritetaan aina kun siihen riittää resursseja. Testaus on syytä suorittaa useilla tavoilla ja oppimisvaikutuksen välttämiseksi aina eri henkilöillä. Jos samoja henkilöitä joudutaan käyttämään (esimerkiksi koehenkilöiden rekrytoinnin vaikeuden vuoksi), on toistot toteutettava ajallisesti mahdollisimman etäällä toisistaan. Erittäin vältettävä saman koehenkilön käyttämistä kahdessa perättäisessä testausseiossa. Testaus tehdään kuitenkin maksimissaan kerran iteraatiota kohden.

Nielsenin (1993, s. 170) mukaan ennen testausta tulee laatia testaus suunnitelma, jossa varsinaisten testien järjestelyt suunnitellaan, määritellään testauksen budjetti ja siinä tarvittavat resurssit ja testataan pilottitapauksella testin kulku. Testitapausten testaaminen esimerkiksi sisäisillä tai muuten ”helppoilla” koehenkilöillä on hyödyllistä siksi, että silloin voidaan vielä havaita helposti virheitä testitapauksissa ja voidaan hahmottaa esimerkiksi testien ajallinen kesto paremmin varsinaisilla edustavilla käyttäjillä tehtävää testausta varten.

Käytettävyydestaajan tärkein tehtävä session aikana – menetelmästä riippumatta – on saada koehenkilö tuntemaan olonsa mukavaksi ja luontevaksi¹⁴ samalla kun tekee huomioita tavoista, joilla tämä käyttää järjestelmää. Tärkeä osa tässä tehtävässä on testaus session alku ja tehtävän esittely. Nielsenin (1993, s. 188–189) ja Krugin (2000, s. 158–159) mukaan tyypillisesti kannattaa mainita seuraavaa:

- Testissä testataan järjestelmää, ei käyttäjää.
- Testattava järjestelmä ei ole testaajan tekemä, joten hänellä ei ole henkilökohtaista sidettä järjestelmään (paitsi jos testaaja itse asiassa on järjestelmän kehittäjä, jolloin asiasta ei kannattane juuri puhua).
- Testauksen tuloksia tullaan käyttämään järjestelmän parantamiseksi, joten lopullinen järjestelmä luultavasti poikkeaa testattavasta.
- Mainita mahdollisesta salassapitosopimuksesta.
- Muistutus, että testiin osallistuminen on vapaaehtoista ja testaus voidaan lopettaa milloin vain.
- Vakuuttaa, että kerättävä data tullaan pitämään salassa.
- Selostaa mahdollisesti tehtävä videotalliointi ja sen käyttötarkoitus.
- Mainita siitä, että kaikenlaiset kysymykset ovat tervetulleita ja arvokasta dataa, mutta että niihin ei välttämättä voida aina vastata testin aikana.
- Mahdolliset testikohtaiset ohjeet, kuten kehotus ajatella ääneen.
- Pyyntö kysyä tarvittaessa tarkentavia kysymyksiä ennen testauksen aloittamista.

Näihin tyypillisesti mainittaviin asioihin lisätään vielä joitakin yritys kohtaisia asioita. Lopullinen listaus Nielsen+Krug-mallin mukaisen käytettävyydestaustaus session aluksi mainittavista asioista on kuvattu liitteessä 5.

Muistiinpanot sessiosta täytyy tehdä mieluusti jo session aikana, mutta viimeistään heti session jälkeen, muutoin ne unohtuvat (Krug, 2000, s. 157). Sen lisäksi, että havainnoidaan koehenkilöä käyttämässä järjestelmää käytettävyydestaustapausten mukaisissa tilanteissa, niissä projekteissa, joihin menetelmä sopii, järjestetään myös korttien lajittelu, ja käytettävyydestaustauksen havaintojen lisäksi saadaan järjestelmän informaatioavaruuden mentaalinen malli. Lisäksi jokainen koehenkilö täyttää *käytettävyydestaustauksen loppukyselyn*, jonka avulla saadaan vielä

¹⁴ ”suojella koehenkilöiden itsetuntoa” (Krug, 2000, s. 155)

kerättyä koehenkilöiden mielipiteitä järjestelmästä sekä erityisesti kvantitatiivista palautetta järjestelmän käytettävyydestä SUS-mittaristosta mukautetun lomakkeen avulla. Lisäksi loppukyselyn avulla saadaan kuulla koehenkilöiden mielipiteitä käytettävyydestäustavan kehittämiseksi.

Testaussessiot nauhoitetaan ruutukaappauksen ja äänen osalta. Nauhoituksia käytetään tarpeen mukaan muistin tukena. Testaussessioiden aikana koehenkilöiden toimintaa havainnoidaan, ja havaintojen perusteella muodostetaan raportti käytettävyydestäustuksen havainnoista. Testauksen tuloksena havaitut ongelmat ja niiden korjaamiseksi tehdyt toimenpiteet raportoidaan.

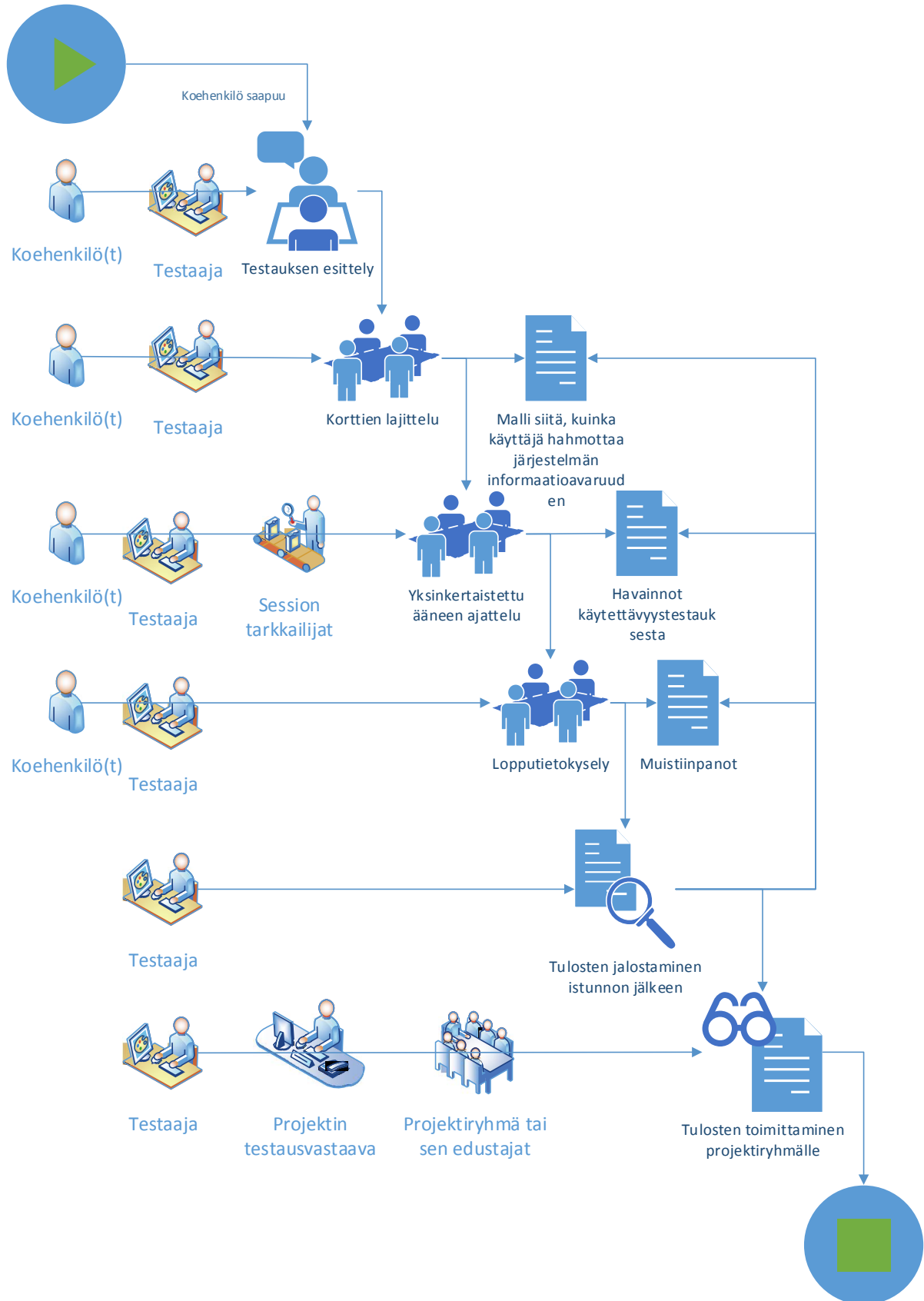
6.3.4 Käytettävyydestäustaussektion kulku

Samlinkin uuden käytettävyydestäustausprosessin mukainen käytettävyydestäustaussektion läpivienti on kuvattu kuviossa 12. Kuviossa kuvataan siis yhden koehenkilöitä hyödyntävän käytettävyydestäustaussektion toteuttaminen alkaen koehenkilön saapumisesta sessioon. Kaikki kolme testausmenetelmää – korttien lajittelu, yksinkertaistettu ääneen ajattelu ja lopputietokysely – ovat vaihtoehtoisia, eikä niitä ole pakko toteuttaa käytettävyydestäustaussektion aikana.

Testaussessio alkaa koehenkilön olon mukavaksi tekemisellä ja testaustapahtuman esittelyllä. Koehenkilölle tarjotaan juotavaa ja kerrotaan hiukan testauksen syystä ja testauksen järjestävästä organisaatiosta. Tämän jälkeen järjestetään korttien lajittelu, jos sille on tarvetta. Korttien lajittelun tuloksena saadaan malli käyttäjän informaatioavaruudesta. Seuraavaksi järjestetään yksinkertaistettu ääneen ajattelu, jossa käyttäjän toiminnan seuraamisen tuloksena saadaan havaintoja käytettävyydestäustauksesta, ja viimeisenä koehenkilöä pyydetään vielä täyttämään käytettävyydestäustauksen loppukysely. Näiden testausmenetelmien tuloksista saadaan koottua käytettävyydestäustausraportti, joka toimitetaan projektiryhmän tarkasteltavaksi.

6.4 Testauskäytäntöjä

Tässä alaluvussa kuvataan Nielsen-Krug-mallin käytännön toteutuksen vaiheita ja piirteitä. Ensin esitetään yleiskuvaus mallin mukaisesta käytettävyydestäustausprosessista. Toiseksi kerrotaan, kuinka eri projekteissa valitaan sovellettavat käytettävyydestäustausmenetelmät. Kolmantena käsitellään koehenkilöiden rekrytointia, ja neljäntenä kuvataan testauksessa hyödynnettävä loppukysely. Lopuksi käydään vielä läpi tapa, jolla käytettävyydestäustauksen havainnot raportoidaan projektiryhmälle.



KUVIO 12 Käytettävyytestaussession läpivienti

6.4.1 Yleiskuvaus käytettävyytestausprosessista

Tapaustutkimuksessa tietoa järjestelmien käytettävyydestä hankitaan tekemällä käytettävyytestausta eri menetelmillä. Samlinkin käyttöön kehitetty käytettävyytestausprosessi on hyvin yleisellä tasolla kuvattu kuviossa 13. Kuviossa on määritelty kussakin vaiheessa (mahdollisesti) tuotettavat artefaktit, vaiheen nimi ja vaiheen todennäköinen suorittaja. Kuviossa on testaussessio kuvattu yksinkertaistetun ääneen ajattelun perusteella, mutta sessio voidaan myös toteuttaa ilman koehenkilöitä esimerkiksi heuristista evaluointia hyödyntäen. Alkaen testausseesion suunnittelusta päättyen työlistan täydennykseen toistetaan iteraatiokohtaisesti – tätä havainnollistaa kuvion iteraatio-nuoli.

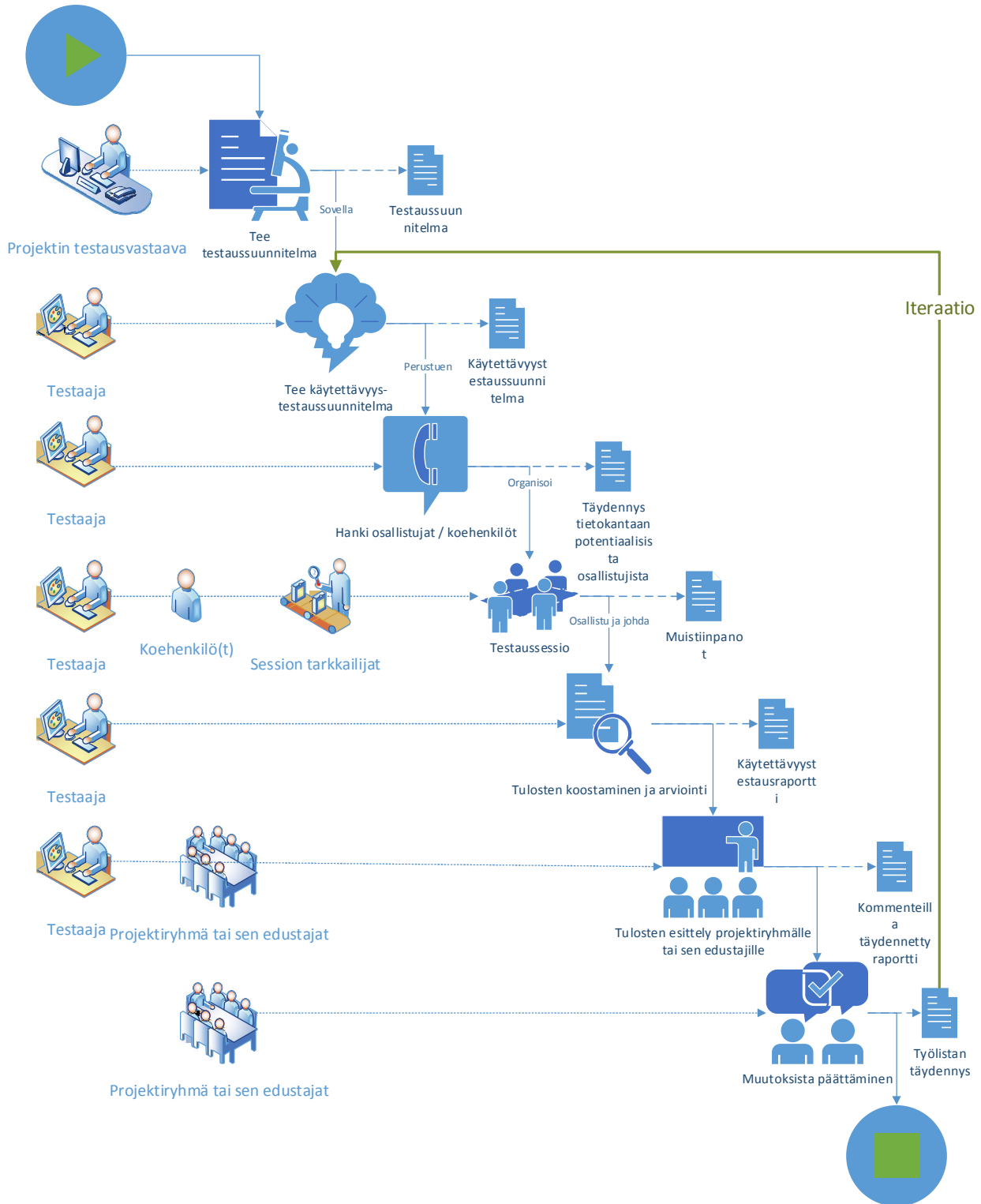
Kuviossa aktiivisiksi rooleiksi on Samlinkin organisaation pohjalta tunnistettu projektin testausvastaava, testaaja, koehenkilö, session tarkkailijat ja projektiryhmä. Testaaja ja projektin testausvastaava voivat olla sama henkilö, ja mahdolliset session tarkkailijat todennäköisesti koostuvat projektiryhmästä, joten nämä roolit eivät ole täysin erillisiä. Varsinaisen testaajan sijasta käytettävyytestauksen voi myös suorittaa kuka tahansa henkilö, joka on määritetty hoitamaan käytettävyytestausta, ja hän voi myös tarvittaessa laatia käytettävyytestaus suunnitelman testaus suunnitelman pohjalta. Tämä onkin prosessin ensimmäinen vaihe, sillä kevytkin käytettävyytestaus täytyy tehdä suunnitelmallisesti. Tämän jälkeen hoidetaan koehenkilöiden rekrytointi ja täydennetään koehenkilötietokanta osallistumistietojen pohjalta. Varsinaiseen testaussessioon voi osallistua tarkkailijoita, joskaan tämän järjestämiseen ei ollut tämän tutkimuksen puitteissa mahdollisuuksia, ja lisäksi tietysti testaaja ja koehenkilö tai koehenkilöt.

Testausseesioiden jälkeen testaaja laatii muistiinpanojen pohjalta käytettävyytestausraportin, joka toimitetaan projektiryhmän edustajille ja mahdollisesti käydään heidän kanssaan läpi. Projektiryhmä voi täydentää raporttia kommentteillaan. Seuraavassa vaiheessa projektiryhmä tekee vielä raportin perusteella tarvittavat päivitykset työlistaan. Mikäli testausta tehdään useamman sprintin aikana, tässä vaiheessa palataan uudestaan käytettävyytestaus suunnitelman päivittämiseen, eli toiseen toteutusaskeleseen.

Työlistan päivitys vaatii havaintojen validoinnin. Käytettävyytestauksen havaintoja validoidaan useilla eri tavoin. Havaintojen merkityksen ja prioriteetin liiketoiminnan näkökulmasta arvioi ensin projektipäällikkö, jonka jälkeen havainnoista viedään työlistalle ne, joihin pyritään jollakin aikataululla reagoimaan. Havainnot voidaan käsitellä lisäksi projektiryhmän kesken. Kuten useissa aiemmissakin tapauksissa (mm. Haikara, 2007; Marty & Twidale, 2005), myös Samlinkin tapauksessa tulosten validointi hoidetaan keräämällä projektiryhmiltä palaute niiden validoinnista.

Samlinkin raportointikäytäntöä, joka on kuvattu alaluvussa 6.5, pyritään kehittämään siihen suuntaan, että projektiryhmien on mahdollisimman helppoa kommentoida havaintojen validiteetti suoraan raporttiin ilman ylimääräistä työtä. Projektiryhmien kommenttien perusteella toimitettavia havaintoja voi-

daan myös hienosäätää sen lisäksi, että raportointikäytäntöä kehitetään. Joissakin projekteissa voi myös olla mahdollista hyödyntää käytettävyyssiantuntijaa tai ulkopuolista käytettävyystestausosaamista havaintojen validoinnissa.



KUVIO 13 Samlinkin käytettävyystestausprosessi

6.4.2 Käytettävyydestaustausmenetelmien valinta

Edellä on kuvattu kolme käytettävyydestaustausmenetelmää, joita voidaan soveltaa projektin tarpeiden mukaisesti. Suunnitteluvaiheessa voidaan toteuttaa projektista riippuen joko avoin tai suljettu korttien lajittelu kevyimmällä mahdollisella tavalla, eli verkossa toteutettuna. Tähän löytyy erilaisia työkaluja, niin ilmaisia kuin maksullisiakin. Useimmissa projekteissa ilmaisista löytynee Samlinkin käyttöön riittävä ratkaisu (Tullis, 2013). Vaihtoehtona koehenkilöitä vaativille käytettävyydestaustausmenetelmille menetelmävaihtoehtoihin otetaan mukaan heuristinen evaluointi (Nielsen, 1995c; Rubin, 2008). Siihen voidaan turvautua silloin, kun resursseja ei ole muutamaa tuntia enempää tai projekti on erittäin aikaisessa vaiheessa. Kaavio soveltuvimman käytettävyydestaustausmenetelmän valinnasta löytyy luvusta 6.4.2.

Tavoitteena on, että rautalankamallien, luonnosten (engl. mock-up) tai viimeistään ensimmäisen prototyypin ollessa olemassa aloitetaan käytettävyydestaustausseisot. Nämä sessiot voidaan aloittaa jo paperiprototyypeilläkin, koska on havaittu, että koehenkilöiden on ilmeisesti helpompi kommentoida selvästi keskeneräistä järjestelmää, jota paperiprototyyppi edustaa, kuin valmiimman oloista järjestelmää esimerkiksi tietokoneella (Holzinger ym., 2005, s. 2). Testauksen aloitusaikatauluun vaikuttaa kuitenkin monta muuttujaa, kuten asiakkaan hyväksyntä testaukselle, integraatioiden toimivuus, demoympäristön valmistuminen, käytettävissä olevien resurssien määrä ja projektin vaiheistus. Samoja koehenkilöitä hyödynnetään mahdollisuuksien mukaan eri käytettävyydestaustausmenetelmissä. Esimerkiksi palkkion saamiseksi voidaan vaatia, tai ainakin toivoa:

1. taustatietolomakkeen täyttämistä
2. korttien lajittelua verkossa
3. varsinaista yksinkertaista ääneen ajattelua hyödyntävää käytettävyydestaustausseisiota ja
4. session päätteeksi täytettävää yksinkertaista palautelomaketta

Projektissa sovellettavien käytettävyydestaustausmenetelmien valinta riippuu projektin tyypistä ja käytettävissä olevista resursseista. Kuvio 14 kuvaa prosessin, jonka avulla voidaan käytettävissä olevien resurssien määrän perusteella arvioida, mitä käytettävyydestaustausmenetelmiä projektissa voidaan hyödyntää. Käytännössä kuvaus vaadittavista resursseista on suuntaa-antava, mutta antaa kuitenkin peruskäsityksen siitä, mitä minkäkin menetelmän soveltaminen vaatii.

Korttien lajittelu on kuvattu suoritettavaksi yksinkertaistetun ääneen ajattelun yhteydessä, ja sisältyvän tarvittaessa sen resurssivaatimukseen. Käytännössä joko avoin tai suljettu korttien lajittelu kuitenkin voidaan tarvittaessa toteuttaa myös muusta käytettävyydestaustauksesta erillisenä, mutta tällöin koehenkilöt täytyy sitä varten rekrytoida erikseen. Tässä tutkimuksessa korttien lajittelua ei testata eikä määritellä erikseen suoritettavana menetelmänä, vaan se toteutetaan muun käyttäjätestauksen yhteydessä.

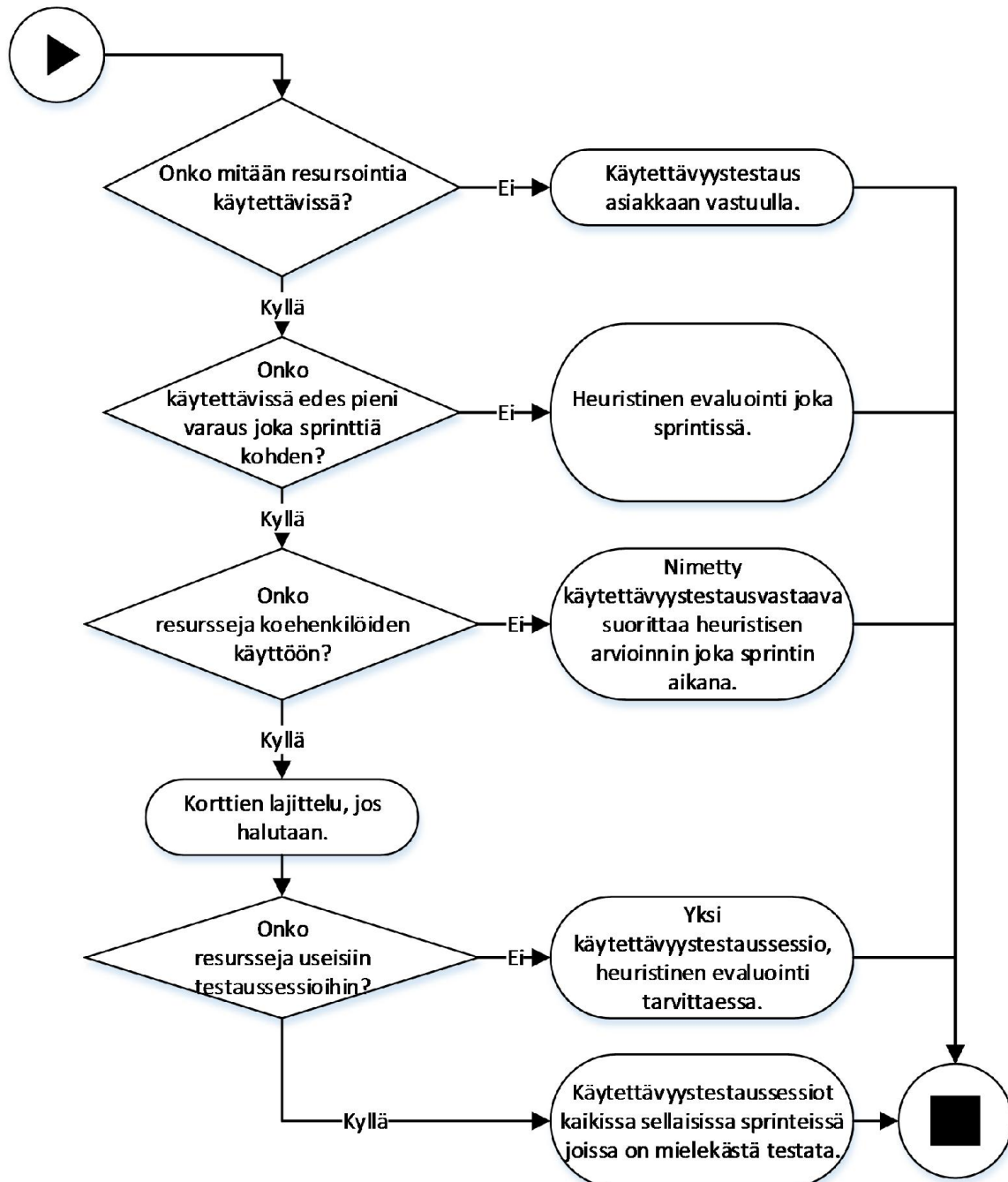
Kuvio 14 kuvaa projekteissa hyödynnettävät eri käytettävyydestausvaihtoehdot. Kuvio antaa yksinkertaiset ohjeet siihen, millä tavoilla käytettävyydestaus voidaan toteuttaa. Mikäli minkäänlaista resursointia käytettävyyden huomiointia varten ei ole, käytettävyydestaus jää asiakkaan vastuulle. Mikäli käytettävissä on edes jonkinlainen resursointi, esimerkiksi yhdestä kymmeneen tuntiin koko projektin aikana, se voidaan hyödyntää suorittamalla heuristinen evaluointi valmiin raportointimallin avulla. Evaluoinnin voi tehdä kuka vain projektiryhmän jäsen hyödyntämällä lisäksi esimerkiksi Nielsenin (1995b) 10 heuristiikkaa. Jos projektissa on käytettävissä esimerkiksi noin yhden työpäivän resurssivaraus sprinttiä kohden, voidaan tarvittaessa nimittää käytettävyydestausvastaava, joka voi suorittaa heuristisen evaluoinnin jokaisen sprintin aikana. Käytettävyydestaajan rooli voi myös olla kiertävä. Koehenkilöiden hyödyntäminen puolestaan vaatii resursseja vähintään noin kahden työpäivän verran sellaisia sprinttejä kohden, jolloin testausta halutaan tehdä, ja lisäksi koehenkilöt täytyy voida palkita jotenkin.

Mikäli käytettävissä olevat resurssit riittävät koehenkilöiden hyödyntämiseen, voidaan yksinkertaista ääneen ajattelua hyödyntävä käytettävyydestaus sessio järjestää, ja tarvittaessa järjestää myös korttien lajittelu. Korttien lajittelu kannattaa järjestää viimeistään ensimmäisen käytettävyydestaus sessioon aikana. Mikäli resursseja on vain yhden käytettävyydestaus sessioon järjestämiseen, käytettävyydestausvastaava voi lisäksi tehdä heuristisen evaluoinnin jopa joka sprintissä, mikäli resurssit riittävät siihen. Mikäli resurssit riittävät useiden käytettävyydestaus sessioiden järjestämiseen, sellainen kannattaa järjestää aina kun se on projektin vaiheen perusteella mielekästä.

Eri vaihtoehdot käytettävyydestausjärjestämiseen kuvion 14 mukaisesti ovat siis seuraavat:

- Kevyimmillään käytettävyydestaus on *asiakkaan vastuulla* ja käytettävyyteen liittyvät havainnot tulevat ilmi todennäköisimmin hyväksyntätestausajan aikana.
- Toiseksi kevyimmässä vaihtoehdossa *joku projektiryhmän jäsenistä tekee järjestelmälle heuristisen evaluoinnin valmiiden ohjeiden ja heuristiikkojen perusteella.*
- Kolmannessa vaihtoehdossa resurssit eivät vielä riitä käytettävyydestaus sessioiden järjestämiseen, mutta *nimetty käytettävyydestausvastaava voi suorittaa heuristisen evaluoinnin jokaisessa sprintissä, jossa se on mielekästä tehdä.*
- Neljännessä vaihtoehdossa *käytettävyydestausvastaava suorittaa käytettävyydestaus sessioon todellisilla koehenkilöillä sekä mahdollisesti heuristisen evaluoinnin.*
- Viidennessä vaihtoehdossa *käytettävyydestaus sessio voidaan järjestää joka sprintissä.*

Neljännessä ja viidennessä vaihtoehdossa voidaan tarvittaessa myös suorittaa korttien lajittelu joko käytettävyystestaussessioiden yhteydessä tai esimerkiksi niitä edeltäen.



KUVIO 14 Käytettävyystestausmenetelmien projektikohtainen valinta Nielsen+Krug-mallissa

6.4.3 Koehenkilöiden rekrytointi

Koehenkilöiden kohderyhmävastaavuuden vaatimukset on pääteltävä projekti-kohtaisesti. Eroja erityisesti kokemuksen määrässä pyritään kompensoimaan käyttämällä persoonia, joihin koehenkilöt voivat samaistua. Tutkimusten (esim. Lindgaard & Chattratichart, 2007, s. 1423) mukaan persoonan omaksuminen auttaa huonostikin haluttua vastaavaa koehenkilöä tuottamaan oikeampia havain- toja.

Sisäisiä koehenkilöitä voidaan käyttää kolmessa eri tilanteessa: (1) sisäisten projektien testauksessa, (2) erittäin aikaisen vaiheen testauksessa, jos ulkoisten koehenkilöiden rekrytointiin on huonosti aikaa ja (3) kun he ovat kaikesta huolimatta edustavia. Tällöinkin koehenkilöiksi rekrytoidaan henkilöitä, joilla ei ole tietämystä varsinaisesta projektista. Useimmiten pyritään kuitenkin käyttämään ulkoisia koehenkilöitä, joita rekrytoidaan sekä avoimesti että verkostojen kautta. Käytännössä rekrytointi hoidetaan ensisijaisesti yliopiston sähköpostilistojen kautta. Tarvittaessa rekrytointia voidaan hoitaa myös esimerkiksi sosiaalisen median tai ilmoitustaulujen kautta, riippuen siitä, minkälaisia koehenkilöitä halutaan rekrytoida.

6.4.4 Testauksen loppukysely

Käytettävyydestestauksen jälkeen suoritetaan vielä loppukysely, jossa kerätään kvantitatiivista dataa järjestelmästä ja tietoja, joilla pyritään ohjaamaan myös käytettävyydestestausprosessin kehitystä. Loppukyselyssä kysytään yksinkertaisilla kysymyksillä koehenkilön antamaa arvosanaa 1-5 erilaisille järjestelmän käytettävyyttä mittaaville määreille, joiden laatimisessa on käytetty ohjeena *UX for the masses* -sivuston (UX for the masses, 2010a) esimerkkiä. Sivuston laati- massa lomakkeessa (UX for the masses, 2010b)¹⁵ on yhdeksän kenttää, joihin voi vastata asteikolla 1-5. Lomake perustuu alun perin Brooken (1996) *järjestelmän käytettävyydsmittareihin* (engl. System Usability Scale, SUS), mutta niitä yksinker- taistettiin entisestään Samlinkin käyttöä varten. Esimerkki käytettävyydestestaus- kerran loppukyselystä löytyy liitteestä 4.

Käytettävyydestestauksessa on mielekästä kerätä havaintojen ja kvantitatiivi- sen datan lisäksi kommentteja käyttäjiltä. Nämä kommentit ovat enemmän mie- lipiteitä kuin valideja faktoja, mutta niidenkin pohjalta voidaan saada hyödyllisiä tietoja, joita käyttää esimerkiksi uusien käyttäjätarinoiden pohjana tai käyttäjien koulutuksen ohjaukseen (Meszaros & Aston, 2006, s. 4). Nämä kommentit kerä- tään myös osana loppukyselyä.

UX for the masses (2010a) -sivuston lomaketta ei otettu sellaisenaan käyttöön, sillä se on alkuperäisessä muodossaan melko sekava. Osa kentistä on positiivisia (esimerkiksi ”järjestelmä oli mielestäni helppokäyttöinen”) ja osa negatiivisia

¹⁵ Haettu osoitteesta [http://www.uxforthemasses.com/blog/wp-con- tent/uploads/2010/09/System-usability-scale-questionnaire.pdf](http://www.uxforthemasses.com/blog/wp-content/uploads/2010/09/System-usability-scale-questionnaire.pdf)

(kuten ”mielestäni järjestelmä oli varsin kömpelö”), joten sekä vastaajan ajatusprosessin että testaajan arvosteluprosessin yksinkertaistamiseksi tutkimusta varten muutettiin kaikki kysymykset tyypiltään positiivisiksi. Tällöin virheiden määrä pienenee lomakkeen luotettavuuden kuitenkin laskematta (Sauro, 2013). Kymmenenneksi kysymykseksi on lisätty järjestelmän hyödyllisyyttä mittaava kysymys, koska hyödyllisyys on myös käytettävyyden kannalta yksi keskeisimpiä asioita. Käytettävyydestaustakertojen välillä lomaketta hiottiin vain hiukan kysymysten kieliasun osalta. Eri versiot kyselystä löytyvät liitteestä 4.

Muutosten seurauksena on mahdollista seurata järjestelmän käyttäjiltään saamaa kokonaisarvosanaa, ja sen laskenta on suoraviivaisempaa kuin alkupe-
räisellä SUS-menetelmällä. Koehenkilö ei myöskään joudu pohtimaan kysymyksiä niin pitkään, eikä todennäköisesti tule vahingossa vastanneeksi väärin. Kyse-
lylomakkeen toimivuutta testattiin jokaisen muutoksen jälkeen yksittäisillä koehenkilöillä, jotka eivät kuuluneet mihinkään projektiin. Heiltä saadun palautteen perusteella tehtiin lomakkeen jatkokehitystä. Otoksena tässä vaiheessa oli kolme henkilöä.

6.5 Testaustulosten raportointi

Seuraavaksi käsitellään tapaa, jolla käytettävyydestaustauksen tulokset on mielek-
käintä kommunikoida projektiryhmälle. Ensin käydään läpi aiemmissa tutki-
muksissa esiteltyjä huomioita käytettävyydestaustauksen raportoinnista. Sitten näi-
hin havaintoihin pohjautuen johdetaan raporttimalli Samlinkin käyttöön. Tämän
raportin käytöstä saadun palautteen perusteella raporttia kehitetään entisestään.

Erityisesti otettaessa uutta mallia käyttöön on tärkeää pystyä raportoimaan
havainnot siten, että muutostavastarintaa voidaan ehkäistä jo ennalta, ja yleisesti-
kin tulosten raportoinnissa on pyrittävä välttämään testauksen ja ohjelmistoke-
hityksen vastakkainasettelua (Barksdale & McCrickard, 2012; C. F. Cohen, Birkin,
Garfield, & Webb, 2004; Cooper, 2004; Lee ym., 2009). Toisaalta myös käytettä-
vyysasiantuntijoiden ja ohjelmistokehittäjien tavoitteet ovat luonnostaan osin ris-
tiriidassa: siinä missä käytettävyydasiantuntijoiden huolena on käyttäjien tarpei-
den täyttyminen, huolehtivat kehittäjät ensisijaisesti järjestelmän toiminnallisten
vaatimusten täyttymisestä ja luotettavuudesta (Lee ym., 2009, s. 5).

Käytettävyydestaustaus on hyödytöntä, jos kukaan ei käytä tuloksia mihinkään.
Samlinkin tapauksessa tätä pyritään välttämään paitsi käytettävyydestaustausmallin
keveydellä (ei luoda turhaa byrokratiaa) myös mahdollistamalla sessioiden tal-
lenteiden tarkastelu sekä toimittamalla tulokset projektiryhmän tai sen edusta-
jien arvioitaviksi nopeasti.

6.5.1 Aiempi tutkimus

Järjestelmälle on asetettu ei-toiminnalliset tai laadulliset vaatimukset, ja käytettävyyteen liittyviä vaatimuksia voi olla niissä mukana, ja riippuen järjestelmästä, niiden prioriteetti voi vaihdella. On kuitenkin olennaista, että käytettävyydestä on jokin tavoite olemassa, ja tavoitteiden perusteella on myös helpompaa priorisoida korjaukset tärkeimpiin ongelmiin (Lee ym., 2009, s. 5). Vaatimukset vaikuttavat siihen, miten raportti on mielekästä laatia ja havainnot priorisoida.

Testausraportti laaditaan lyhyesti mutta ongelmat selkeästi kuvaavasti. Koska sessiot nauhoitetaan, tarvittaessa voidaan myös antaa projektiryhmän käyttöön katkelmia nauhoitteista, ja lisäksi voidaan käyttää kuvakaappauksia havainnollistamaan ongelmakohtia, kuten alan tutkimuksessa usein suositellaan (mm. Assistant Secretary for Public Affairs, 2013b). Osa raporttia on koehenkilöiltä käytettävyydestä loppukyselyssä kerätyn kvantitatiivisen datan perusteella laskettu järjestelmän käytettävyyden kokonaisarvosana asteikolla 0-100, ja merkittävimmät huomiot kyselyn tuloksista. Loppukysely vastaa osaltaan tarpeeseen validoida testaaajan tekemiä havaintoja, sillä numeerisen arvostelun ja testaaajan havaintojen yhteensopivuutta voidaan tarkastella, ja loppukyselyn avoimet kysymykset toisaalta tuovat vielä entistä vahvemmin esille käyttäjän Nielsen, Overgaard, Pedersen ja Stage (2005) arvioivat tutkimuksessaan käytettävyydestä raporttien hyödyllisyyttä mobiilijärjestelmän kehittäjille ja suunnittelijoille. Tavoitteena oli selvittää, mitkä tiedot raportissa ovat heille tärkeimpiä. Tutkimus suoritettiin ”perinteisillä”, varsin kattavilla käytettävyydestä raportteilla, mutta tuloksista voi saada hyödyllistä tietoa siitä, mikä tieto on kevyessä käytettävyydestä hyvä sisällyttää hyvin pelkistettyihin raportteihin. Tutkimuksessa mobiilijärjestelmän kehittäjille näytettiin erilaisia käytettävyydestä raportteja ja heiltä kysyttiin, mitkä raportin sisältämät tiedot olivat heille tärkeimpiä. Kehittäjät suhtautuivat tutkimukseen innokkaina, mutta olivat hiukan skeptisiä tulosten hyödyllisyyttä kohtaan. Käytettävyyssession keräämistä arvosteltiin esimerkiksi sillä perusteella, että tosielämän käyttäjät olivat pääasiassa jo eläneet järjestelmän kanssa vuosia ja oppineet käyttämään sitä. Täten datan kerääminen uusilta käyttäjiltä olisi hyödytöntä.

Tutkijat koostivat seuraavan listan havainnoista siitä, mitkä osat käytettävyydestä raporttien sisällöstä kehittäjät kokivat itselleen merkityksellisiksi (Nielsen ym., 2005, s. 402):

1. Lista käytettävyysongelmista, jossa on kunkin ongelman seikkaperäinen kuvaus, on erittäin tärkeä osa raporttia.
2. Seikkaperäinen selostus, kuvaus tai loki käyttötapausten etenemisestä ongelman tapahtuessa on myös tärkeä osa raporttia.
3. Tiedot testisession toteutuksesta (kuten session toteutustapa, tehtävänänot sekä koehenkilöiden tyypit ja mielipiteet) ovat kehittäjille tärkeitä, mutta voivat johtaa tulosten vähättelyyn ja selittelyyn tehtävänänotta tai koehenkilöistä johtuviksi.

4. Yleisemmät ilmaukset ja arviot käytettävyydestä ja muut havainnot eivät ole kovin käyttökelpoisia.
5. Metodispesifiset tiedot (esim. NASA-TLX) ilman kattavaa kontekstia ja avaamista raportissa eivät ole hyödyllisiä kehittäjille.

Yleisenä huomiona todettiin, että positiivinen käytettävyysspalautte on kehittäjille turhaa. Se ei auta heitä kehittämään järjestelmää paremmaksi (Nielsen ym., 2005, s. 401). Toisaalta havaittiin, että kehittäjien ottaminen mukaan käytettävyysspalautteen arviointiin ja se, että he pääsivät vaikuttamaan palautteen esitystapaan, sitoutti heidät myös korjaamaan havaittuja ongelmia (Nielsen ym., 2005, s. 401). Mikäli kehittäjät eivät ottaneet palautetta tosissaan, videotaltiointi sessiosta mainittiin toimivaksi tavaksi vakuuttaa kehittäjä ongelmista järjestelmästä, mutta video täytyisi käytännössä leikata esityskelpoiseksi ja tähän kuluu paljon aikaa.

6.5.2 Raporttimalli

Nielsen+Krug-mallissa raportissa käytettävä kriittisyysluokitus täsmentyy testauksen aikana. Tavoitteena on löytää sellainen malli havaintojen priorisoinniseksi, että havaintojen siirtäminen onnistuu vaivattomasti minkä tahansa projektin työlistalle, ja toisaalta havaintojen tulkinta on vaivatonta myös työlistaa näkemättömälle työntekijälle tai asiakkaalle. Lisäksi kuhunkin ongelmaan tai huomioon esitetään kommentti, ensisijaisesti parannus- tai korjausehdotus, jotta saadaan laskettua kynnystä raportin havaintojen huomiointiin (Hornbæk, 2008, s. 268).

Sekä osa testaaajista että projektipäälliköistä¹⁶ toivoi monipuolisempia tilastoja tai metriikoita mukaan raporttiin. Nielsenin (2001) määrittelemät olennaisimmat metriikat¹⁷, onnistumisprosentti, epäonnistumisten määrä, käyttäjien subjektiivinen tyytyväisyys ja tehtävään kuluva aika, voitaisiin myös yhdistää anekdoottisiin havaintoihin seuraavalla tavalla: onnistumisprosentti ja epäonnistumisten määrä ilmaistaan suoraan havaintojen kriittisyysluokituksessa: jos havainto aiheutti epäonnistumisen, ja siten myös laski onnistumisprosenttia, se aiheuttaa saman tien havainnon prioriteetin nousun kriittiseksi, ja jos ongelma on niin vakava, että testitapausta on mahdoton suorittaa, prioriteetti voi olla vielä korkeampi. Näin Nielsenin nämä mittarit täsmäivät Holzingerin (2005) mittariin virheiden määrästä. Käyttäjien subjektiivinen tyytyväisyys taasen saadaan selville mahdollisella loppukyselyllä. Tehtävään kuluva aika voidaan huomioida vasta äärimmäisissä tapauksissa, jolloin se voidaan kirjata havaintoon ylös. Eriytyisen vahva argumentti ajan mittaamista vastaan on sen metriikan epäluotettavuus, kun käytettävänä menetelmänä on yksinkertaistettu ääneen ajattelu: ääneen ajattelu hidastaa huomattavasti tehtävistä suoriutumista (Rubin, 2008, s. 54). Tällä tavoin metriikat voidaan ottaa kiinteäksi, mutta helposti huomioitavaksi osaksi käytettävyydestauksen havaintojen raportointimallia.

¹⁶ Projektipäällikkö J ja Testaaja M, katso myös liite 6, jossa projektiryhmien täydelliset kommentit.

¹⁷ Katso luku 4.3, jossa kuvaus metriikoista ja niiden vertailua kilpaileviin metriikoihin.

Raportti toimitetaan sähköpostitse projektiryhmälle, ja tarvittaessa järjestetään lyhyt palaveri esimerkiksi päiväpalaverin yhteydessä tulosten purkamiseksi. Raportit myös arkistoidaan sähköisesti. Lisäksi projektiryhmien kysymyksiin, koskien havaintoja tai niiden korjausehdotuksia, voidaan vastailla tarpeen mukaan. Esimerkiksi kuvakaappausten toimittaminen ongelmakohtista tai muutoin havaintoihin liittyen on vaivatonta, sillä testausseSSIoiden aikana näyttö ja web-kameran kuva nauhoitetaan myöhemmin tarkasteltavaan videomuotoon.

Raportin eri pääversiot löytyvät liitteestä 7. Liitteestä ilmenevät olennaisimmat kehitysvaiheet. Raporttia on kehitetty liitteessä 6 esitetyn palautteen ja osin myös liitteessä 3 kuvattujen havaintojen perusteella. Palaute on kerätty projektiryhmältä ja Samlinkin käytettävyyssiantuntijalta.

Lopullinen versio raportista sisältää varsinaisen raporttidokumentin (jonka sisältö voidaan lähettää esimerkiksi suoraan sähköpostina), ja siihen liittyvän Excel-muotoisen koosteen käytettävyyssitestauksen havainnoista. Varsinainen raportti voi sisältää yleisen kuvauksen ja johdannon lisäksi korttien lajittelun ja mukautetun SUS-kyselyn tulosten kuvaukset, kuten kuvaajat eri mittareiden arvojen kehityksestä.

6.6 Yhteenveto

Tässä luvussa esiteltiin SamScrum-projektimalli, Nielsen+Krug-käytettävyyssitestausmalli sekä se, miten jälkimmäinen on integroitu SamScrum-malliin. Esitelty malli yhdistää Nielsenin laajalti koetellut ja kentällä arvonsa todistaneet kevyen käytettävyyssuunnittelun menetelmät Samlinkin käytössä olevan projektimallin vaiheisiin ja tarjoaa käytännönläheiset ohjeet niiden soveltamiseksi projekteissa.

SamScrumiin yhteen sovitettaviksi käytettävyyssitestauksen tai -arvioinnin menetelmiksi valittiin korttien lajittelu, heuristinen arviointi ja yksinkertaistettu ääneen ajattelu. Näiden menetelmien avulla saadaan täytettyä Samlinkin tarpeet käytettävyyssitestausmallille. Korttien lajittelu mahdollistaa käyttäjien sisäisen informaatiomallin hahmottamisen, ja erityisesti sovelluksen navigaatorakenteen muokkaamisen sen pohjalta, ja se on varsin nopeasti toteutettavissa oleva menetelmä. Heuristinen evaluointi voidaan tehdä tilanteissa, joissa koehenkilöiden hyödyntämiseen ei ole resursseja tai muutoin mahdollisuutta – sen toteuttaminen vie keveimmillään vain tunteja, mutta tuottaa kuitenkin huomattavan määrän havaintoja. Lisäksi hyödynnetään yksinkertaista ääneen ajattelua aina kun mahdollista.

Lisäksi tässä luvussa kuvattiin laajennetun Samlinkin mallin testauskäytäntöjä, käytettävyyssitestausprosessi ja käytettävyyssitestaukseen liitettävä loppukysely. Testaus tulosten raportointikäytännön kehittämistä varten tutustuttiin aiemmissa tutkimuksissa käytettyihin tapoihin tulosten raportoinnista ja laadittiin niiden pohjalta lähtökohta Samlinkin raportointikäytännöille.

7 TUTKIMUSTULOKSET

Tämän luvun tarkoituksena on kertoa, miten Nielsen+Krug-mallia hyödynnettiin käytännössä Samlinkin projektityöskentelyssä ja millaisia kokemuksia siitä saatiin. Ensin kuvataan yleisiä havaintoja ja piirteitä käytettävyydestauksen suorittamisesta käytännössä, erityisesti valmisteluista ja tulosten käsittelystä. Toiseksi käsitellään tapaa ja menetelmiä, joilla käytettävyydestausta toteutettiin Samlinkin päivittäisessä projektityöskentelyssä. Kolmanneksi käsitellään yksityiskohtaisesti käytettävyydestauksen aiheuttamia kustannuksia mallin kokeilun aikana. Neljänneksi arvioidaan mallin kokeilusta saatuja hyötyjä. Viidenneksi käydään vielä läpi yleisen tason havaintoja käytettävyydestauksen suorittamisesta käytännössä. Kun käsitellään tuloksia ja toimintatapoja projekteissa, käytetään projekteista liitteessä 2 esiteltyjä lyhenteitä. Samasta liitteestä löytyy myös tarkempi kuvaus kustakin projektista.

7.1 Käytettävyydestauksen suorittamisesta yleisesti

Tässä alaluvussa käsitellään yleisiä piirteitä toteutettuun käytettävyydestauksen suorittamiseen ja erityisesti valmisteluun liittyen. Ensin kuvataan yleisvalmisteluita, jotka tehtiin jotta käytettävyydestaus yleensä ottaen voitiin aloittaa. Sitten käsitellään tavat, joilla koehenkilöitä rekrytoitiin koehenkilötietokantaan. Tämän jälkeen käydään läpi ensin käytettävyydestaustakerran ja sitten käytettävyydestaustaussession valmistelut ja käytettävyydestaustaussession kulku. Sitten kerrotaan käytettävyydestauksen tulosten raportointimallin kehitysvaiheista. Lopuksi käsitellään vielä käytettävyydestaustaussession ja -kerran tuottamien tulosten, eli käytettävyydestauksen havaintojen, käsittelytavasta.

7.1.1 Yleisvalmistelut

Käytettävyytestauskäyttöön asennettiin yksi kannettava tietokone, jolle asennettiin kattava valikoima selaimia, ruuduntallennukseen käytettävä ohjelmisto *Open Broadcaster Software* ja tarvittavat ohjelmistot etäyhteyksiä varten. RETU-projektin käytettävyytestauksen molempia iteraatioita, ja PA-projektin toisen iteraation testausta varten asennettiin vastaavat ohjelmistot myös toiseen tietokoneeseen, sillä näissä projekteissa tehtiin osa testauksesta ympäristössä, jossa sai käyttää vain hyvin rajatun tyyppisiä tietokoneita. Tätä toista konetta ei kuitenkaan ollut mielekäästä käyttää muiden projektien testaukseen siinä olevien rajoitteiden vuoksi.

Korttien lajittelua varten luotiin ilmainen käyttäjätili *Optimal Workshop*-palveluun. Koehenkilöiden tietojen ylläpitoa ja keräämistä varten perustettiin kysely *LimeSurvey*-kyselyalustalle, joka saatiin ilmaiseksi käyttöön yhteistyökumppani Norppandalotti Softwarelta. Samlinkin käyttöön kehitetyt käytettävyytestauksen yleisohjeet on kuvattu liitteessä 5. Ne kuvaavat yleiset tehtävät käytettävyytestauksen alkuun ja loppuun. Ohjeet on kehitetty pohjautuen useisiin valmiisiin ohjeisiin (Krug, 2000, s. 158–160; Northrop, 2012; Rubin, 2008, s. 78), ne on mukautettu Samlinkin käyttöön sopiviksi ja niitä on paranneltu tutkimuksen aikaisten havaintojen perusteella. Kaiken kaikkiaan käytettävyytestauksen valmisteluun kului kymmeniä tunteja, jos mukaan lasketaan eri työkalujen vertailu ja valinta ja testauksen toteutustavan suunnittelu sekä havaintojen raportointikäytäntöjen kehittäminen.

7.1.2 Koehenkilöiden rekrytointi koehenkilökantaa varten

Koehenkilöiden rekrytointi tehtiin varsin opportunistisesti: koehenkilötietokantaan oltiin valmiita ottamaan kenen tahansa tiedot, ja tietokannan sisältämiä henkilöitä rekrytoitiin sitten projektien tarpeiden mukaan. Rekrytointi hoidettiin hyödyntäen useita yliopiston postituslistoja sekä nykyisille opiskelijoille että alumneille. Lisäksi koehenkilötietokantaan tietonsa ilmoitti useita muilta koehenkilöiltä, eli niin sanotusti puskaradion kautta, tiedon asiasta saaneita henkilöitä.

Projektien vaatimukset koehenkilöiden tyyppille olivat melko joustavat, joten rekrytointia voitiin tehdä varsin laajasti. Osa koehenkilötietokantaan tietonsa antaneista potentiaalisista koehenkilöistä ei kuitenkaan lopulta osallistunut yhteenkään käytettävyytestaussessioon. Kysely, jonka tiedot koehenkilötietokantaan tallennettiin, löytyy liitteestä 1.

7.1.3 Käytettävyytestauskerran valmistelu

Niissä projekteissa, joissa käytettiin korttien lajittelua, laadittiin lajiteltavat kortit järjestelmän navigaatiohierarkian perusteella ja luotiin projektille korttienlajittelutehtävä *Optimal Workshop*-sivustolle. Käytettävyytestauksen lopputietokysely perustettiin samalle *LimeSurvey*-palvelimelle, jossa myös koehenkilötietokanta

sijaitsee. Lopputietokyselyllä kerättiin kvantitatiivista dataa järjestelmän käytettävyydestä, ja lisäksi palautetta käytettävyydestä järjestelyiden onnistumisesta koehenkilön näkökulmasta. Kyselystä kerrottiin tarkemmin luvussa 6.4.4.

Käytettävyydestä varten valmisteltiin käytettävyydestäsuunnitelma. Tämä laadittiin opportunistisesti niiden materiaalien pohjalta, joita projektissa oli käytettävissä. Tämä tarkoittaa, että sen pohjana käytettiin testaus suunnitelmaa, käyttötapauksia, testitapauksia tai jotain muuta määrittystä. Jokaisen käytettävyydestäsuunnitelman aluksi suunnitelma joka tapauksessa täytyi päivittää.

7.1.4 Käytettävyydestäsuunnittelun valmistelu ja kulku

Tässä aluvuossa kuvataan valmistelut erityisesti sellaista käytettävyydestäsuunnittelua varten, jonka aikana hyödynnetään testausmenetelmänä yksinkertaista ääneen ajattelua ja mahdollisesti korttien lajittelua, ja voidaan kerätä myös lopuksi kyselyn avulla lisätietoa. Nämä ohjeet ovat varsin konkreettisella tasolla, mutta olennaisia Nielsen+Krug-mallissa hyödynnettävien käytettävyydestäsuunnittelun menetelmien hyödyntämisen onnistumisen kannalta.

Viimeistään kun testaus session ajankohta on sovittu koehenkilön kanssa, täytyy tehdä tilavaraus käytettävyydestäsuunnittelun järjestämistä varten. Ennen session alkua valmistellaan käytettävyydestäsuunnittelun käytettävä työasema testattavana olevan projektin tarpeet huomioiden ja varmistetaan työaseman toimivuus. Koska käytettävyydestäsuunnittelun aikana sekä koehenkilö että käytettävyydestäsuunnittelija puhuvat todennäköisesti session aikana paljon, on tärkeää varata jotain juotavaa molemmille. Käytettävyydestäsuunnittelijan tulee myös varata itselleen muistiinpanovälineet ja projektin käytettävyydestäsuunnitelma avuksi testaus session ohjauksessa.

Käytettävyydestäsuunnittelun alkuun kuuluu salassapitosopimuksen allekirjoittamisella. Malli sopimuksesta löytyy liitteestä 9. Koehenkilölle mainitaan (Krugin (2000, s. 158–160) esimerkkiä mukailen) ainakin seuraavat asiat:

- Testissä testataan järjestelmää, ei käyttäjää.
- Testattava järjestelmä on keskeneräinen.
- Testattava järjestelmä ei ole testajan tekemä, joten hänellä ei ole henkilökohtaista sidettä järjestelmään (paitsi jos testaja itse asiassa on järjestelmän kehittäjä, jolloin asiasta ei kannattane juuri puhua).
- Muistutus, että testiin osallistuminen on vapaaehtoista ja testaus voidaan lopettaa milloin vain.
- Selostaa mahdollisesti tehtävä videotalliointi ja sen käyttötarkoitus.
- Vakuuttaa, että kerättävä data tullaan pitämään salassa.
- Mainita siitä, että kaikenlaiset kysymykset ovat tervetulleita ja arvokasta dataa, mutta että niihin ei välttämättä voida aina vastata testin aikana.
- Pyyntö kysyä tarvittaessa tarkentavia kysymyksiä ennen testauksen aloittamista.

Lisäksi selitetään testauksessa hyödynnettävien menetelmien piirteet ja testauksen kulku.

Nielsen+Krug-mallin kokeilussa käytettävyytestaustaussektion kesto vaihteli noin 45 minuutista tuntiin ja viiteentoista minuuttiin. Session alusta erilaisiin valmisteluihin ja koehenkilön olon tekemiseen mukavaksi, kuten kahvin tai virvoitusjuomien tarjoamiseen, kului noin viisi minuuttia. Tämän jälkeen allekirjoitettiin salassapitolomake, jonka jälkeen aloitettiin käytettävyystestaus. Hyödynnettävät menetelmät vaihtelivat projektin mukaan, ja niiden hyödyntämiseen kului valtaosa ajasta. Järjestys oli aina sama. Ensimmäisenä tehtiin korttien lajittelu, jos sitä hyödynnettiin, sillä tässä vaiheessa termistö ei ole järjestelmän käytön takia vielä tuttu koehenkilölle. Tämän jälkeen tapahtui varsinainen käytettävyystestaus, eli yksinkertaistettu ääneen ajattelu. Lopuksi toteutettiin käytettävyystestauksen loppukysely.

Projektiryhmät esittivät toiveen mahdollisuudesta sessioiden seuraamiseen, ja yksittäisten havaintojen kohdalla myös käyttäjien ongelmat tai havainnon tärkeys saatettiin kyseenalaistaa. Sessioiden seuraaminen pyritään myöhemmin mahdollistamaan sähköisesti esimerkiksi yrityksen sisäverkkoon perustettavan ”striimauspalvelimen” kautta. Mahdollisuus sessioiden seuraamiseen tai ainakin tulosten todentamiseen videotallenteita tarkkailemalla sitouttaa henkilöstöä ja vahvistaa heidän uskoaan käytettävyystestauksen tuloksiin, ja etänä toteutettuna se ei myöskään häiritse koehenkilöä. Samlinkilla ei ole käytettävissä käytettävyystestauslaboratoriota, jossa olisi esimerkiksi käytettävissä yksisuuntaisen tarkastelun mahdollistava peili-ikkuna.

Käytettävyystestauksen lopuksi koehenkilöä pyydetään täyttämään loppukysely, jos sellainen projektia varten on laadittu, ja annetaan palkinto testiin osallistumisesta. Session jälkeen pyritään käsittelemään muistiinpanot välittömästi ja laatimaan niistä havainnot raportointimallin mukaisesti.

7.1.5 Käytettävyystestauksen tulosten raportointimallin kehitys

Käytettävyystestauksen tulosten raportointiin kehitettiin malli, jota voidaan hyödyntää myöhemmin menetelmiä hyödynnettäessä. Mallin eri kehitysversioita voi tarkastella liitteestä 7. Lomakkeen kehitys tehtiin projektiryhmiltä saadun palautteen sekä käsitteellisen-analyttisen tutkimuksen havaintojen perusteella.

Käytännössä ensimmäinen versio raportista laadittiin ”Krugin (2000, 2006) hengessä” erittäin yksinkertaiseksi, lyhyeksi listaksi tärkeimmistä havaituista ongelmista, ja selkeyden vuoksi havainnot ryhmiteltiin Nielsenin (1995b) erittäin tunnettujen heuristiikkosten mukaan. Tällainen listaus oli projektipäälliköiden mieleen, mutta ei antanut tarpeeksi tietoa ongelmien riittävän tarkkaan hahmottamiseen ja korjaamiseen. Lisäksi havaintoja ei voinut helposti järjestellä haluamallaan tavalla, esimerkiksi heuristiikkosten sijasta prioriteetin mukaan. Kehittäjien toiveesta ja monien tarkennuspyyntöjen myötä raportointitapa muutettiin hiukan kattavammaksi ja ongelmat pyrittiin kuvaamaan tarkemmin. Kaikkiin ongelmiin pyrittiin tarjoamaan hyvin konkreettinen ratkaisuehdotus.

7.1.6 Käytettävyytestaussionsession tulosten käsittely

Käytettävyytestaussionsession aikana muistiinpanot tehtiin samaan paperiin, johon oli laadittu testitapaukset. Tällöin muistiinpanot saatiin tehtyä sen testitapauksen kohdalle, jota suorittaessa ne ilmenivät, ja ne oli melko vaivatonta session jälkeen siirtää Excel-muotoiseen raporttiin. Excel-raporttia varten tulokset nimettiin (nimenä toimi hyvin yksinkertainen kuvaus ongelmasta, kuten ”navigaation toiminta ei ole käyttäjälle selvä”), niistä laadittiin kuvaus, jossa ongelma pyrittiin määrittelemään tarkasti, ilmaistiin mihin heuristiikkaan ne liittyvät ja mikä niiden prioriteetti on sekä ehdotettiin ratkaisua ongelmaan. Myöhemmissä versioissa raportointimallista lisäksi ilmaistiin se, monellako käyttäjällä ongelma ilmeni.

Muistiinpanot käytettävyytestaussionsiosta siirrettiin raporttimuotoon, siis ensimmäisissä iteraatioissa tekstidokumenttiin heuristiikoittain järjesteltynä, ja sittemmin laskentataulukon kurinalaisemmin ja tarkemmin eriteltyinä, heti kun mahdollista. Usein tämä tapahtui välittömästi käytettävyytestaussionsession jälkeen, mutta aina aikaa ei ollut välittömästi session jälkeen vaan havainnot voitiin kirjata vasta päivän viimeisen käytettävyytestaussionsession jälkeen.

Arvio havaintojen prioriteetista perustui koehenkilöiden havainnointiin, ja luokittelussa noudatettiin aluksi Bugzillan (Roczek & Tryon, 2014) luokitusta, jota sittemmin täydennettiin hiukan laajemmaksi muilta harjoittajilta¹⁸ mallia katsoen, sekä Yhdysvaltain Puolustusministeriön standardiin MIL-STD-882D (Department of Defence, 2000) pohjautuen.

7.1.7 Käytettävyytestauskerran tulosten käsittely

Käytettävyytestaustuksen tulokset toimitettiin projektiryhmälle käsiteltäväksi, ja ryhmä sai käsitellä tulokset haluamallaan tavalla. Jotta tulosten hyväksynnästä saatiin dataa, tämän tutkimuksen puitteissa projektiryhmiltä pyydettiin palaute tavoista, joilla kuhunkin havaintoon reagoitiin. Tämän lisäksi havainnot kävi läpi Samlinkin käytettävyyssiantuntija. Yhdessä projekteista voitiin lisäksi hyödyntää riippumattoman, ulkopuolisen tahon käytettävyytestaustuksen tuloksia osin samalle projektille, jota tässä tutkimuksessa testataan.

Tulosten pohjalta projektiryhmät laativat uusia käyttäjätarinoita (tai muita kohteita) työlialle, tai vaihtoehtoisesti loivat uusia virheitä havaintojen pohjalta työlialle tai virheidenseurantajärjestelmään. Kohteet tai virheet voitiin poimia kuluvan, seuraavan tai myöhemmän sprintin työlialle, mutta useinkaan kuluvan sprintin työlialle ei enää voitu ottaa muita uusia kohteita, paitsi mahdollisesti kaikista pahimpia virheitä, joiden korjaus on syytä tehdä ennen hyväksyntätestausta. Osa havainnoista voitiin myös eri syistä hylätä ja jättää käsittelemättä.

Projektiryhmillä kesti parista päivästä yli kuukauteen toimittaa kommenttinsa käytettävyytestauskerran tuloksiin, siis havaintoihin ja loppukyselyn koosteeseen. On kuitenkin huomionarvoista, että palaute kerättiin ensisijaisesti

¹⁸ <http://www.wikihow.com/Categorize-Defects> (luettu 1.5.2014) ja Sarafinchan (2010)

tutkimuksen tarpeita varten ja tilanteessa, jossa pintapuolisempi ja kevyempi palaute riittää, myös tarpeellinen palaute saataisiin nopeammin. Tietoa siitä, kuinka nopeasti havaintojen toimituksen jälkeen projektiryhmät käsittelivät tulokset tai kauanko käsittelyyn kului, ei ole tiedossa.

7.2 Käytettävyydestaustuksen toteuttaminen projekteissa

Tässä alaluvussa kerrotaan käytettävyydestaustuksen toteutuksesta varsinaisissa projekteissa. Erityisesti tarkastellaan tapoja, joilla käytettävyydestausta projekteissa tehtiin. Projektien esittely löytyy liitteestä 2 ja luvusta 5.4. Yhteenvedo eri projekteissa hyödynnetyistä testausmenetelmistä ja koehenkilöistä on esitetty taulukossa 7. Varsinaisten käytettävyydestaustamenetelmien lisäksi niissä projekteissa, joissa hyödynnettiin koehenkilöitä, suoritettiin lopuksi myös käytettävyydestaustuksen loppukysely kvantitatiivisen käytettävyyssdatan keräämiseksi.

TAULUKKO 7 Projekteissa hyödynnetyt testausmenetelmät ja koehenkilöiden lukumäärät

Projektin tunnus	Testauskertojen lukumäärä	Käytetyt testausmenetelmät	Koehenkilöiden lukumäärät testauskerroittain
RETU	2	Yksinkertaistettu ääneen ajattelu	1: 4 2: 5 Yhteensä: 9
PA	3	Korttien lajittelu Yksinkertaistettu ääneen ajattelu	1: 6 2: 4 3: 5 Yhteensä: 15
SH	1	Korttien lajittelu Yksinkertaistettu ääneen ajattelu	1: 6 Yhteensä: 6
HE	1	Heuristinen evaluointi	Ei koehenkilöitä

7.2.1 RETU-projekti

RETU-projektissa käytettävyydestaustaus jouduttiin useista syistä johtuen aloittamaan varsin myöhäisessä vaiheessa. Käytettävyydestaustaus suoritettiin kahdessa peräkkäisessä sprintissä, joista edellinen oli viimeinen toteutussprintti ja jälkimmäinen ensimmäinen ylläpitosprintti. Testattavana olivat ne järjestelmän näkyvät, jotka ovat useimmiten käytössä, siis resursoijien ja resurssipyyntöjen tekijöiden kuten projektipäälliköiden ja ratkaisukonsulttien käyttämät toiminnallisuudet. Projektissa hyödynnettiin yksinkertaistettua ääneen ajattelua sekä kyseleä käytettävyyssdatan keräämiseksi.

Ensimmäisellä kerralla oli neljä ja jälkimmäisellä kerralla viisi koehenkilöä. Projektin koehenkilöt olivat sisäisiä, ja heidät pyrittiin valitsemaan vapaaehtoisiksi ilmoittautuneista siten, että otannasta saatiin edustava, testitapaukset vas-

tasivat koehenkilön käyttäjäroolia ja oppimisvaikutus saatiin minimoitua. Erityisesti viimeisin vaatimus vaikeutti rekrytointia, koska järjestelmän käyttöönotto ajoittui hyvin pian ensimmäisen käytettävyystestauskerran jälkeen, ja suuri osa henkilöstöstä oli ehtinyt käyttää järjestelmää ainakin jonkin verran toiseen käytettävyystestauskertaan mennessä.

Molemmilla kerroilla testattiin samoja osia järjestelmästä. Projektin terminologiassa *jatkokehitys* tarkoittaa normaalin kehityksen osana myöhemmin toteutettavia ominaisuuksia, siinä missä muissa projekteissa jatkokehityksellä viitataan mahdollisesti myöhemmässä projektissa toteutettavia ominaisuuksia. Siten tämän projektin osalta havainnot, jotka projektipäällikön määritelmän mukaan ovat *jatkokehitystoiveita*, luokitellaan työlistalle uusina tai vanhoina kohteina. Varsinaiseen jatkokehitys-kategoriaan luokitellaan havainnot, joita ei tässä vaiheessa edes harkita toteutettavan, mutta jotka olisi kuitenkin hyödyllistä ottaa mukaan, ellei projektipäällikkö kuvaa havainnon olevan duplikaatti, pienimerkityksinen tai määrittelyn vastainen. Näissä tapauksissa havainto luokitellaan hylätyksi.

RETU-projektissa käytettävyystestausuunnitelma laadittiin hyödyntäen yksityiskohtaisesti laadittuja testitapauksia tukena. Käytettävyystestausuunnitelman pohjaksi otettiin suoraan testitapaukset sopivasti järjesteltyinä ja karsittuina, ja niitä täydennettiin muistiinpanoilla testauksen aluksi ja lopuksi suoritettavista tehtävistä.

Projektissa ei hyödynnetty korttien lajittelua, sillä navigaatorakenne oli verraten yksinkertainen ja syvimmillään kaksitasoinen. Yksinkertaista ääneen ajattelua hyödyntäen koehenkilön suoritettavaksi annettiin hänen roolinsa huomioiden, mutta ei kuitenkaan välttämättä vain sen mukaisesti, valittuja testitapauksia suoritettavaksi. Lopuksi koehenkilöiltä kerättiin kyselyn avulla palaute järjestelmän käytettävyydestä.

RETU-projektissa projektiryhmä käsitteli käytettävyystestauksen tulokset siten, että projektipäällikkö, joka toimi myös tuoteomistajana ja testauspäällikkönä, arvioi havaintojen validiteetin, pyysi tarvittaessa tarkennuksia ja vei lopulta haluamansa havainnot työlistalle. Havaintoja saatiin todella paljon. Tuloksista varsin suuri osa kuitenkin hylättiin joko siksi, että nykyinen toiminnallisuus on jo määrittelyn mukainen tai siksi, että havainnon merkitys on pienehkö eikä sitä ehditä ottaa käsiteltäväksi. Kiireen vuoksi ensimmäisen sprintin havaintoja ei juuri ehditty huomioida seuraavaa sprinttiä varten.

Ensimmäisessä käytettävyystestauskerrassa aikaa kului 3,25 tuntia koehenkilöä kohden, ja toisella kerralla 2,65 tuntia. Toisella testauskerralla testattavia oli yksi enemmän, ja valmisteluun kului vähemmän aikaa testitapausten ollessa hio- tumpia.

7.2.2 PA-projekti

PA-projekti oli tutkimuksessa mukana olevista projekteista laajin. Projektissa havaintoja ehdittiin kerätä varsin monesta käyttöliittymästä, ja huomattava osa havaintoihin ehdotetuista korjauksista ehdittiin myös huomioida testauskertojen välillä. Käytettävyystestausta tehtiin kolmessa peräkkäisessä sprintissä, joista

viimeisin oli projektin viimeinen toteutusprintti. Testattavana olivat kaikissa iteraatioissa järjestelmän kuluttajakäyttäjälle suunnatut osiot, toisessa iteraatioissa lisäksi myös järjestelmän ylläpitäjäkäyttäjille suunnatut osiot. Projektissa käytettävyydestä tehtiin varsin opportunistisesti ja sen mukaan, minkä testaaminen missäkin vaiheessa projektia oli hyödyllisintä. Ensimmäisellä ja kolmannella testauskerralla tuotettiin käytettävyydestä havaintoja myös liitännäisprojektille, kun taas toisella testauskerralla, jolloin päästiin hyödyntämään koehenkilöinä ylläpitäjäkäyttäjää vastaavia käyttäjiä, testattiin myös järjestelmän ylläpitonäkymiä. Toinen testausiteraatio toteutettiin Samlinkin tiloissa Espoossa, johon ylläpitäjäkäyttäjät voitiin saada paikalle, ensimmäinen ja kolmas iteraatio Jyväskylässä, tällöinkin Samlinkin tiloissa. Ensimmäisellä kerralla oli kuusi, toisella kerralla neljä ja kolmannella kerralla viisi koehenkilöä.

Käytettävyydestä suunnitelma laadittiin myös PA-projektissa testitapausten pohjalta. Testitapauksia jouduttiin karsimaan ja hiukan muokkaamaan, jotta ne vastasivat hyvin kuluttajakäyttäjän tyypillistä käyttötapausta. Yksinkertaistetun ääneen ajattelun ja loppukyselyn lisäksi PA-projektin liitännäisprojektia varten järjestettiin korttien lajittelu, jota varten järjestelmän suunniteltu navigaatorakenne muutettiin järjesteltäviksi ”korteiksi” *OptimalWorkshop*-järjestelmän *OptimalSort*-työkaluun. Käytettävyydestä aloitettiin korttien lajittelulla. Tämän jälkeen käyttäjä suoritti tehtävänannon perusteella käyttötapausten yksinkertaista ääneen ajattelua hyödyntäen, ja lopuksi kerättiin vielä kyselyllä käyttäjän oma arvio järjestelmän käytettävyydestä.

PA-projektissa tulokset toimitettiin testausvastaavalle ja projektipäällikölle, jotka kävivät ne läpi, pyysivät tarvittaessa tarkennuksia, loivat työlialle uusia kohteita havaintojen pohjalta ja kävivät sitten muun projektiryhmän kanssa ne läpi. Liitännäisprojektissa projektipäällikkö ja scrummaster koostivat tuloksista koosteen asiakkaalle toimitettavaksi, ja tulosten pohjalta tehtiin pienehköjä päivityksiä testiympäristöön, jotta testaus tilanne saatiin toteutettua mahdollisimman sujuvaksi. Liitännäisprojektille toimitettiin lisäksi järjestelmän navigaatorakennetta varten korttien lajittelun tulokset. Vertaamalla tuloksia järjestelmän testaus hetken sivustorakenteeseen voitiin havaita, että käyttäjillä oli jonkin verran vaikeuksia navigaation hahmottamisessa ja käyttäjien mentaalinen malli järjestelmän informaatioavaruudesta erosi hiukan kuvitellusta. Kaksi navigaation ylä-tason kohdetta oli niin lähellä toisiaan, että näiden välillä oli vaikea jakaa alata-sion navigaatiokohteita. Lisäksi varsinainen käyttäjätestaus paljasti, että myös muutamien alata-sion kohteiden välillä oli ajoittain hankala tehdä eroa.

Ensimmäisessä käytettävyydestä testauskerrassa aikaa kului 3,92 tuntia koehenkilöä kohden. Tähän vaikuttivat käytettävyydestä suunnitelman laatimisen vaatima aika ja tekniset ongelmat, sillä ensimmäisellä testauskerralla pyrittiin myös järjestämään testaus hyvin ennakkoluulottomasti muualla kuin Samlinkin tiloissa etäyhteyksien avulla. Etäyhteyksien toimivuus osoittautui vähemmän luotettavaksi, ja myöhemmissä iteraatioissa testaus ei järjestetty muualla kuin Samlinkin toimipisteellä.

Toisella testauskerralla aikaa kului 2,63, ja kolmannella kerralla 2,95 tuntia. Toisella testauskerralla testattavia oli yksi enemmän kuin ensimmäisellä, ja valmisteluun kului käytettävyytestaussyunnitelman päivitystarpeista huolimatta vähemmän aikaa, sillä projektipäällikkö hoiti pääosan aikataulutuksesta. Testauksen kohteena oli kuitenkin osia, joita ensimmäisellä kerralla ei testattu, joten osa testitapauksista piti laatia uusiksi, samoin kuin koehenkilöille annettavista tehtävänannoista. Kolmas käytettävyytestauseraatio oli järjestelyiltään hyvin pitkälti ensimmäisen kaltainen. Koehenkilöiden rekrytointiin kului enemmän aikaa kuin ensimmäisellä tai toisella kierroksella, koska innokkaimmat vapaaehtoiset oli jo hyödynnetty, eikä tällä kertaa haluttu käyttää sisäisiä koehenkilöitä.

7.2.3 SH-projekti

SH-projektissa käytettävyytestaus tehtiin yhdessä projektin toteutusprintissä. Testattavana oli tavallisille kuluttaja-asiakkaille suunnattu osa järjestelmästä, ja koehenkilöiksi rekrytoitiinkin tätä vastaavia henkilöitä (kuusi koehenkilöä). Hyödynnettäviä testausmenetelmiä olivat yksinkertaistettu ääneen ajattelu ja loppukysely.

Käytettävyytestauksen havaintojen merkitystä olivat arvioimassa osa projektin kehittäjistä, projektipäällikkö ja käytettävyyssiantuntija. Projektipäällikkö ja käytettävyyssiantuntija esittivät joitakin tarkentavia kysymyksiä havaintoihin liittyen. Pääosa havainnoista oli kuitenkin melko yksiselitteisiä, eikä kriittisyysluokitukseltaan erittäin kriittisiä havaintoja tehty.

SH-projekti oli testauksen näkökulmasta yksinkertaisin tutkimuksen projekteista. Testattavana oli tarkoitukseltaan hyvin selkeä tuote, ja koehenkilöiksi haettiin yksinomaan kuluttajakäyttäjiä. Käytettävyytestaussyunnitelma laadittiin tuotteen käyttäjätarinoiden pohjalta, ja koehenkilöt rekrytoitiin varsin vähällä vaivalla. Ainoana käytettävyytestausmenetelmänä oli yksinkertaistettu ääneen ajattelu, sillä korttien lajittelulle ei ollut navigaation yksinkertaisuuden ja vakiintumattomuuden vuoksi tarvetta. Testaus SH-projektissa tehtiin vain yhdessä sprintissä.

7.2.4 HE-projekti

HE-projektin käytettävyytestaus suoritettiin elintarvikealan yrityksen nettisivuille. Testaus tehtiin Samlinkin käytettävyyssiantuntijan johdolla, ja testausmenetelmä oli heuristinen läpikäynti – siis yhdistelmä kognitiivista läpikäyntiä ja heuristista evaluointia. Käytännössä tässä hybridissä siis tehdään sekä kognitiivinen läpikäynti (suppeasti) että vapaamuotoisempi heuristinen evaluointi. Projektissa Samlinkin käytettävyyssiantuntija toimi toisena käytettävyytestaajana, ja hän myös arvioi havaintojen validiteetin.

HE-projekti erosi muista projekteista siten, että ainoa siinä hyödynnetty Samlinkin mallin mukainen testausmenetelmä oli heuristinen evaluointi, joka suoritettiin osana heuristista läpikäyntiä. Lisäksi siinä arviointi tehtiin muun

konsultoinnin ohessa, ei osana sprinttimäistä, iteratiivista työskentelyä. Kokonaisuudessaan projektissa aikaa käytettävyytestaukseen kului 8,5 tuntia, josta 7,5 tuntia yhden päivän aikana, ja siitä 6 tuntia testaukseen ja loput raportin koostamiseen. Myöhemmin aikaa kului vielä yksi tunti havaintojen jatkokäsittelyyn ja käytettävyyssiantuntijan kysymyksiin vastaamiseen. Tämä aika kului siis kurinalaiseen kognitiiviseen läpikäyntiin ja vapaamuotoisempaan heuristiseen evaluointiin yhteensä. Tosiasiassa heuristinen evaluointi vei alle puolet tästä ajasta tuottaen silti enemmän havaintoja kuin kognitiivinen läpikäynti.

7.2.5 Yhteenveto

Huolimatta siitä, että tutkimuksessa mukana olleet projektit olivat keskenään varsin erilaisia, oli käytettävyydestauksen toteutus projekteissa pääpiirteissään varsin suoraviivaista ja oli loogista valita tietyt menetelmät hyödynnettäviksi tietyissä projekteissa. Testauksen aikana ilmeni joitakin teknisiä ongelmia, ja useita testauksen piirteitä, erityisesti tulosten raportointiin käytettävää raportointipohjaa, kehitettiin tutkimuksen aikana palautteen ja tehtyjen havaintojen perusteella eteenpäin. Testauskertoja järjestettiin yhteensä seitsemän, joista kuudessa hyödynnettiin koehenkilöitä ja yhdessä ei. Tulosten validoinnissa käytettiin apuna erityisesti projektiryhmien kommentteja, mutta toissijaisesti myös muiden testaajien havaintoja tai käytettävyyssiantuntijan kommentteja.

7.3 Käytettävyydestauksen kustannukset

Tässä alaluvussa käsitellään käytettävyydestauksesta aiheutuneita kustannuksia. Tärkein kustannuserä on käytettävyydestaukseen kulunut aika. Lisäksi käydään läpi muut testauksesta aiheutuneet kustannukset. Näitä kustannuksia käsitellään projekteittain, sprinteittäin ja koehenkilöittäin.

7.3.1 Käytettävyydestaukseen kulunut aika

Käytettävyydestaukseen kulunut aika vaihteli projekteittain ja iteraatioittain (sprinteittäin) suuresti. Kuluneeseen aikaan vaikutti ainakin se, oliko projektille valmiina testaussuunnitelmaa tai käyttötapauksia, kuinka monimutkainen projektissa kehitettävä järjestelmä ja sen testausympäristö oli, minkälaisia koehenkilöitä vaadittiin ja paljonko havaintoja kertyi. Taulukko 8 kuvaa ajankäytön projekteittain ja iteraatioittain aktiviteettityypin (valmistelu, testaus, purku ja analysointi) mukaisesti eriteltynä. Iteraatiot on järjestetty kronologisesti. PA-projektin toisessa iteraatiossa valmisteluun kuluva aikamäärä jäi pieneksi siksi, että projektipäällikkö hoiti koehenkilöiden rekrytoinnin. HE-projektissa, jossa käytettävyydestausmenetelmät olivat heuristinen evaluointi ja kognitiivinen läpikäynti, testauksen valmistelut hoiti Samlinkin käytettävyyssiantuntija. Testauksessa käytettiin tällöinkin hyväksi lähtökohtaisesti samaa raporttimallia kuin muiden

projektien havaintojen raportoinnissa, mutta siihen täydennettiin kognitiivisen läpikäynnin tehtäväkuvaukset.

Käytettävyydestäukseen kuluva aika mitattiin projekti- ja käytettävyydestäuskertakohtaisesti, sekä jaoteltuna kussakin (1) valmisteluun, (2) testaukseen ja (3) tulosten purkuun ja analysointiin. Lisäksi aikaa arvioitiin käytettävien menetelmien suhteen: heuristista evaluointia käytettiin merkittävästi vain yhdessä projektissa, muissa projekteissa sitä käytettiin osana testitapausten valmistelua. Korttien lajittelua käytettiin yhdessä projektissa jokaisen käytettävyydestäussession aluksi, muissa kahdessa projektissa sitä ei käytetty. Valmisteluun kuuluvaan aikaan ei ole laskettu varsinaista pohjatyötä, kuten käytettävyydestäuksen koehenkilöiden tietojen ylläpitoon tarkoitettun järjestelmän luomista ja ensimmäistä rekrytointivaihetta, jossa vasta haettiin vapaaehtoisia käytettävyydestäusta varten. Sen sijaan projektikohtainen rekrytointi on laskettu valmisteluun mukaan, samoin kuin käytettävyydestäussessioiden aikataulutusta ja järjestelyä.

TAULUKKO 8 Käytettävyydestäukseen kulunut aika iteraatiokohtaisesti eriteltynä

Projekti	Ite- raatio	Kulunut aika (tuntia)			Yhteensä
		Valmistelu	Testaus	Purku ja analysointi	
RETU	1	3,00	8,00	2,00	13,00
PA	1	7,00	10,50	6,00	23,50
PA	2	0,50	5,00	5,00	10,50
SH	1	2,75	6,00	5,25	14,00
RETU	2	4,00	5,00	4,25	13,25
PA	3	3,75	5,00	6,00	14,75
HE	1	0,00	6,00	2,50	8,50

Taulukko 9 kuvaa käytettävyydestäukseen kulunutta aikaa koehenkilöä kohden. Erityisesti varsinaisen testauksen organisointiin kuuluva aika väheni koko ajan myöhempien iteraatioiden myötä rutiinin kehittyessä ja toimintatavan täsmentyessä. RETU- ja PA-projektien ensimmäisten iteraatioiden osalta tekniset ongelmat ja toimintatavan vakiintumattomuus näkyy korkeampana ajankäyttönä kuin myöhemmissä iteraatioissa.

TAULUKKO 9 Käytettävyydestäukseen kulunut aika koehenkilöä kohden

Projekti	Ite- raatio	Kulunut aika koehenkilöä kohden (tuntia)			Yhteensä
		Valmis- telu	Testaus	Purku ja analy- sointi	
RETU	1	0,75	2,00	0,50	3,25
PA	1	1,17	1,75	1,00	3,92
PA	2	0,13	1,25	1,25	2,63
SH	1	0,46	1,00	0,88	2,33
RETU	2	0,80	1,00	0,85	2,65
PA	3	0,75	1,00	1,20	2,95

Valmisteluun kuluva aika oli vaikea saada vähennettyä, sillä käytettävyydestäussuunnitelman päivittämiseen ja testisessioiden aikataulutukseen kului aina

jonkin verran aikaa. Poikkeuksen tästä muodosti PA-projektin iteraatio 2, jossa, kuten jo aiemmin on mainittu, projektipäällikkö hoiti koehenkilöiden rekrytointin. Tulosten purun ja analysoinnin osalta rutiini myös kehittyi, mutta tutkimuksen myötä ilmeni, että tulosten raportointi täytyy tehdä verraten yksityiskohtaisella tasolla, ja täten tulosten koostamiseen kului aikaa. Lisäksi jo alusta alkaen oli selvää, että ongelmiin täytyy tarjota ratkaisuehdotus, koska käytettävyydestaajalle on verraten selvää, mikä milloinkin on ongelman syy ja hänen on helpompi ehdottaa mahdollista korjausta kuin kehittäjien tai projektipäällikön. Myös näiden ehdotusten laatimiseen kului aikaa, erityisesti jos havaintoja oli paljon.

Taulukossa 10 on kuvattu käytettävyydestaukseen kulunut aika iteraatiota kohden yhteensä, ja jaoteltuna eri aktiviteetteihin. Heuristista evaluointia hyödynnettiin tässä tutkimuksessa yhdessä projektissa, ja silloinkin osana heuristista läpikäyntiä, jossa evaluoinnin lisäksi tehdään kognitiivinen läpikäynti. Näihin kului yhteensä aikaa 8,5 tuntia, joka jakautui karkeasti ottaen puoliksi eri menetelmien välille. Tällöin heuristiseen evaluointiin kului suunnilleen vastaava aika kuin 1-2 koehenkilön kattavaan käytettävyydestaukseen, eli noin 4-5 tuntia. Heuristisen läpikäynnin valmisteluja, käytännössä siis kognitiivisen läpikäynnin valmistelua, ei ole huomioitu ajankäytössä. Itse heuristinen evaluointi ei käytännössä vaatinut valmistelua, jos havaintojen raportointiin käytettiin samaa dokumenttipohjaa kuin muissakin menetelmissä.

TAULUKKO 10 Käytettävyydestauskertaan kulunut aika

Projekti	Iteraatioiden määrä	Kulunut aika keskimäärin käytettävyydestauskertaa kohden (tuntia)			
		Valmistelu	Testaus	Purku ja analysointi	Yhteensä
RETU	2	3,50	6,50	3,13	13,13
PA	3	3,75	6,83	5,67	16,25
SH	1	2,75	6,00	5,25	14,00
HE	1	0,00	6,00	2,50	8,50

Taulukossa 11 kuvataan käytettävyydestauskertakohtaisesti havaintoon kulunut aika minuutteina. Havaintojen lukumäärä sisältää myös virheet. Erityisen korkeat arvot, 39 minuuttia PA-projektin ensimmäisessä sprintissä, ja 42 minuuttia saman projektin kolmannessa sprintissä, ovat selitettävissä sillä, että ensimmäisen sprintin aikana teknisiä haasteita testauksen järjestelyissä oli huomattavasti (etäyhteys ei toiminut ja yksi käytettävyydestaussessio jouduttiin uusimaan) ja kolmannen sprintin aikana havaintoja kertyi melko vähän, sillä kriittisimmät havainnot oli saatu jo korjattua aiempien sprinttien palautteen perusteella.

Nielsen+Krug-mallissa testaus työllisti pääsääntöisesti vain yhtä työntekijää, varsinaista käytettävyydestaajaa, ja lisäksi projektiryhmää siltä osin kuin he käsittelivät testauksen tuloksia tai auttoivat testauksen valmisteluissa. Projektiryhmän jäsenten työmäärä oli kuitenkin pääsääntöisesti vähäinen, ja testaus oli heidän näkökulmastaan melko vaivatonta.

TAULUKKO 11 Käytettävyytestaukseen havaintokohtaisesti kulunut aika

Projekti	Iteraatio	Havaintojen lukumäärä	Havaintoon kulunut aika (minuutteina)
RETU	1	33	23,64
PA	1	36	39,17
PA	2	33	19,09
SH	1	28	30,00
RETU	2	46	17,28
PA	3	21	42,14
HE	1	58	8,79

Testaukseen kului menetelmistä ja toimintatavoista riippuen 8,5 tunnista yli 25 tuntiin sprinttiä kohden. Käytännössä pelkkään heuristiseen evaluointiin voi arvioida menneen puolesta työpäivästä yhteen työpäivään, sillä tällä kertaa sitä ei päästy kokeilemaan yksinään, vaan samalla suoritettiin myös kognitiivinen läpikäynti samalle järjestelmälle. Koehenkilöitä hyödyntävään testaukseen meni aikaa kaiken kaikkiaan 10,5 tunnista 23,5 tuntiin sprinttiä kohden. Käytännössä siis resursseja vaadittiin huomattavasti vähemmän kuin esimerkiksi Nielsenin (1994b) arvioima, minimissään 50 tuntia sprinttiä kohden, mutta toisaalta vaihtelu oli melko suurta.

Jos samassa projektissa testausta tehtiin useampien iteraatioiden aikana, testaukseen kuuluva aika hiukan vähentyi ensimmäistä seuraavien sprinttien aikana. Tähän vaikutti paitsi tutkijan harjaantuminen testauksen järjestämiseen, myös se, että ensimmäistä iteraatiota varten käytettävyytestaussuunnitelma täytyi laatia muiden materiaalien, kuten testaussuunnitelman, käyttötapausten ja prototyypin, pohjalta, mikä oli hetkittäin työlästäkin. Sprinttikohtaiseen käytettyyn aikaan oli suuri vaikutus silläkin, kuka hoiti koehenkilöiden rekrytoinnin.

Koehenkilökohtainen testauksen kesto valmistelu mukaan lukien oli hiukan yli kahdesta tunnista yli neljään tuntiin. Ensimmäisissä testauskerroissa valmistelu kesti aina pidempään kuin myöhemmissä iteraatioissa, sillä ensimmäisellä kerroilla täytyi saada laadituksi käytettävyytestaussuunnitelma, jota myöhemmin voitiin hyödyntää vain pienin päivityksin. Lisäksi testaaajan rutiini kehittyi selvästi myöhempiä testausiteraatioita kohden, jolloin aikaa kului entistä vähemmän asioiden suunnitteluun. Koehenkilöiden rekrytointi toisaalta muuttui myöhemmissä sprinteissä hiukan ensimmäisiä hankalammaksi.

Ajallisesti käytettävyytestaus ajoittui testauskerrasta riippuen 1-3 päivälle. Tähän on laskettu mukaan testaus, tulosten analysointi ja koostaminen ja toimitaminen projektiryhmälle. Valmistelu tyypillisesti jakautui testauskertaa edeltävien viikkojen ajalle siten, että maksimissaan noin kaksi viikkoa ennen ensimmäistä testausseksiota suunniteltiin mahdolliset ajat sessioille ja lähetettiin kutsu potentiaalisille koehenkilöille. Tämän jälkeen hallinnoitiin ilmoittautumisia sitä mukaa kun niitä tuli, ja mikäli oli tarvetta, noin viikko ennen ensimmäistä sessiota suoritettiin täydennysrekrytointi. Käytettävyytestaussuunnitelman laatiminen tai päivitys tehtiin aikataulujen salliessa, kuitenkin ennen ensimmäistä käytettävyytestaussessiota. Näistä aktiviteeteista ainut merkittävästi aikaa kulluttava oli suunnitelman laatiminen, sillä rekrytointi oli melko suoraviivaista.

Käytettävyytestaussessiot ajoittuivat, riippuen testaajan aikatauluista ja erityisesti koehenkilöiden rekrytoinnin onnistumisesta, 1-3 päivälle. Raportti testauksen tuloksista koostettiin mahdollisuuksien mukaan jo sitä mukaa kun käytettävyytestaussessioita saatiin toteutettua, mutta tyypillisesti samana päivänä viimeisten käytettävyytestaussessioiden kanssa. Raportin lähettäminen projektiryhmälle tapahtui joko samana tai seuraavana päivänä riippuen siitä, vaatikko se vielä hiomista.

Käytännössä käytettävyytestaussessioiden, tulosten koostamisen ja jakamisen projektiryhmälle mahdollistaminen yhteen päivään venyttää päivän melko pitkäksi. PA-projektin iteraation 2 käytettävyytestaus tehtiin päivän aikana, ja päivä oli noin 10-tuntinen. Tyypillisimmin tulokset saatiin toimitettua samana tai seuraavana päivänä, kun viimeiset testaussessiot järjestettiin, eli usein testauksen toisena, kolmantena tai neljäntenä päivänä riippuen siitä, kuinka testaussessiot saatiin järjestettyä. Keskimääräisesti testaukseen kului työtunteja yhden tai kahden työpäivän verran sprinttiä kohden.

7.3.2 Käytettävyytestauksen kustannukset kokonaisuutena

Testauksen eri vaiheisiin kulunut aika oli suurin testauksesta aiheutuva kustannus, mutta myös muita, pienempi kustannuseriä oli. Mukaan kustannuksiin on laskettava koehenkilöille maksettava korvaus heidän ajastaan: sisäisten koehenkilöiden tilanteessa siis normaali sisäisen työn hinta, ja ulkoisten koehenkilöiden tapauksessa heille maksettava palkkio testaukseen osallistumisesta.

Koehenkilöiden ajan hintana käytetään tässä tutkimuksessa kolmea eri arvoa: sisäisten koehenkilöiden hinta on 50 euroa tunnilta, joka on laskennallinen sisäisen työn kustannus. Ulkoisten koehenkilöiden hinta on joko 9,75 euroa tunnilta, mikäli heille annettiin palkkioksi Finnkinon vapaalippu, tai 0 euroa, kun käytettiin asiakkaan edustajia, joille ei maksettu korvausta osallistumisesta testaukseen.

Taulukossa 12 kuvataan käytettävyytestauksen kustannuksia sisältäen ajan ja koehenkilöiden kustannukset, ja kuvataan koehenkilökohtaiset käytettävyytestauksen kustannukset. Koehenkilökohtainen kustannus on relevantti mittari useista syistä. Ensinnäkin se on vertailukelpoinen joihinkin aiempiin (Krug, 2000; Nielsen & Landauer, 1993) ja erityisesti tuleviin tutkimuksiin, ja toiseksi se tuottaa erittäin konkreettista tietoa, jonka avulla arvioida käytettävyytestauksen mielekkyyttä ja esimerkiksi sopivaa koehenkilöiden määrää eri tapauksissa. Taulukosta on nähtävissä, että myös niissä projekteissa, joissa käytettiin sisäisiä koehenkilöitä, jotka olivat ulkoisia kalliimpia, valtaosa kustannuksista muodostui käytettävyytestaajan tekemästä työstä. Iteraatiokohtainen kustannus vaihteli 625,00 ja 1233,50 euron välillä niissä projekteissa, joissa hyödynnettiin koehenkilöitä, ja koehenkilöitä hyödyntämättömän HE-projektin osalta kustannus oli vain 425,00 euroa.

TAULUKKO 12 Käytettävyytestauksen kustannukset sprinttiä kohden

Pro- jekti	Sprint- ti	Sprintin ai- kana kulu- nut aika (h)	Ajan hinta	Muut kustan- nukset	Kustannukset yhteensä	Kustannukset koehenkilöä kohden
RETU	1	13,00	650,00 €	200,00 €	850,00 €	212,50 €
PA	1	23,50	1 175,00 €	58,50 €	1 233,50 €	205,58 €
PA	2	10,50	525,00 €	100,00 €	625,00 €	156,25 €
SH	1	14,00	700,00 €	58,50 €	758,50 €	126,42 €
RETU	2	13,25	662,50 €	250,00 €	912,50 €	182,50 €
PA	3	14,75	737,50 €	48,75 €	786,25 €	157,25 €
HE	1	8,50	425,00 €	-	425,00 €	-

Taulukko 13 kuvaa käytettävyytestauksen kustannukset projektikohtaisesti, sekä projekteittain koehenkilöä kohden keskiarvoisesti. Koehenkilökohtainen hinta käytettävyytestaukselle vaihteli projektien välillä suuresti riippuen toisaalta projektin järjestelyistä, koehenkilöiden tyypistä ja siten hinnasta, sekä vaatimuksista saataville tuloksille. PA-projektissa koehenkilöt olivat pääosin varsin edullisia (2 sisäisen koehenkilön hinta oli 50 euroa, 9 ulkopuolelta rekrytoidun koehenkilön 9,75 euroa ja 2 asiakasta edustavan koehenkilön 0 euroa), mutta toisaalta monimutkainen testiympäristö ja muita projekteja kompleksisempi raportointikäytäntö (testauksella tuotettiin havaintoja kahdelle eri projektille) aiheutti sen, että aikaa testaukseen kului koehenkilöä kohden keskimäärin enemmän kuin muissa projekteissa. Koko tutkimuksen aikana kallein testausiteraatio oli PA-projektin ensimmäinen sprintti, jossa aikaa kului käytettävyytestaussuunnitelman laatimiseen, teknisten ongelmien ratkomiseen ja testiympäristöön tutustumiseen.

TAULUKKO 13 Käytettävyytestauksen lopullinen hinta eri projekteissa

Pro- jekti	Projektin käytettä- vyytestauksen koko- naiskustannus	Koehenki- löiden luku- määrä	Koehenkilöiden kes- kihinta projektissa	Koehenkilö- kohtainen kus- tannus
RETU	1 762,50 €	9	50,00 €	195,83 €
PA	2 644,75 €	13	15,94 €	203,44 €
SH	758,50 €	6	9,75 €	126,42 €
HE	425,00 €	-	-	-

Taulukko 14 kuvaa projektien käytettävyytestauksen kustannukset vielä havaintokohtaisesti. Havaintojen hinnat käyttäjätestausta tehtäessä vaihtelivat 22,31–29,39 euron välillä. Jos mukaan lasketaan vain hyväksytyt havainnot, havaintokohtainen hinta nousee välille 32,65 - 40,99 euroa. Vain heuristista läpikäyntiä (jonka osana heuristinen evaluointi) hyödyntävä HE-projekti tuotti havaintoja huomattavasti halvemmalla: havaintokohtainen hinta oli 7,33 euroa, ja hyväksytyinkin havainnon hinta vain 7,87 euroa.

TAULUKKO 14 Havaintokohtainen hinta projekteissa

Pro- jekti	Havain- tojen lu- kumäärä	Hyväksytyjen havaintojen lukumäärä	Koko projektin kustannus	Havaintokoh- tainen hinta	Hyväksytyn havainnon hinta
RETU	79	43	1 762,50 €	22,31 €	40,99 €
PA	90	81	2 644,75 €	29,39 €	32,65 €
SH	28	23	758,50 €	27,09 €	32,98 €
HE	58	54	425,00 €	7,33 €	7,87 €

7.4 Testaustulokset ja niiden hyödyllisyys

Tässä alaluvussa käsitellään käytettävyydestauksen tuottamia havaintoja ja niiden hyödyntämistä sekä niiden hyväksyntää. Havainnot esitetään luokiteltuna sekä sen mukaan, minkä tyyppisiä kohteita (virhe, jatkokehityspiirre tai käytettävyysongelma) ne tuottavat työlialle, että sen mukaan, miten projektiryhmä päättää reagoida niihin. Jälkimmäisestä luokittelusta saadaan johdettua myös tulosten hyväksyntäprosentti. Projektiryhmien yksityiskohtaiset kommentit havainnoista ja testaustuloksista yleensä löytyvät liitteestä 6, mutta ne käydään lyhyesti läpi myös tässä alaluvussa siinä järjestyksessä, jossa testaus toteutettiin.

7.4.1 Havaintojen määrä ja tyyppi

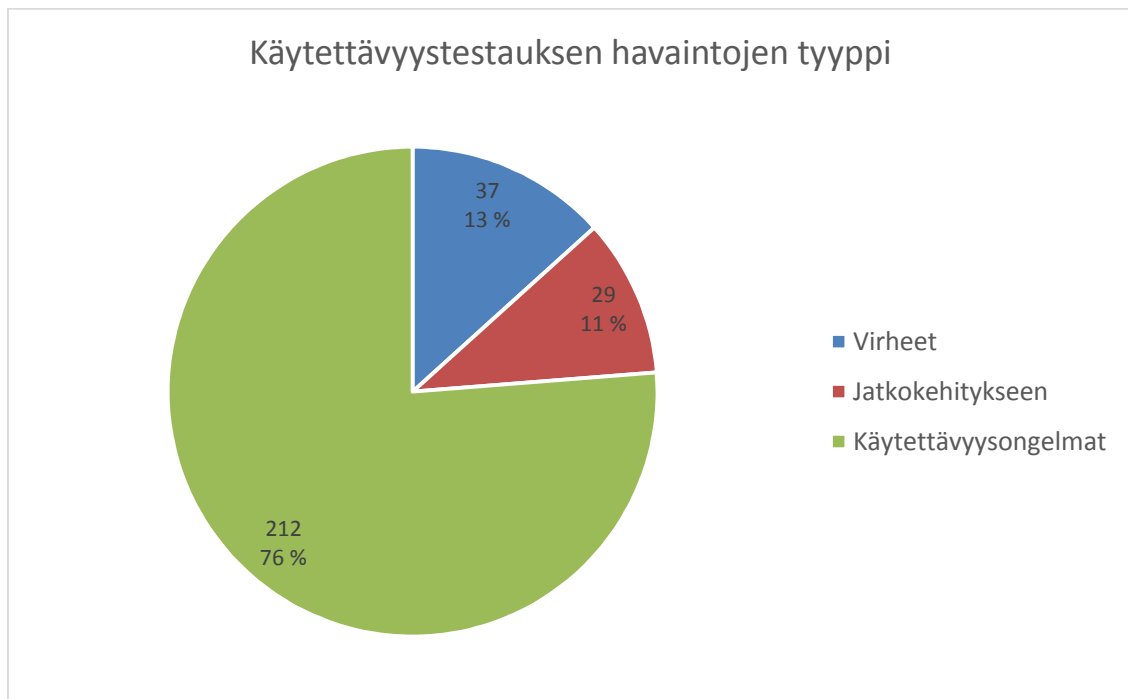
Käytettävyydestaus tuotti melko paljon havaintoja, ja erityisesti ensimmäisten sprinttien, joissa käytettävyydestausta tehtiin, tulosten käsittelyssä havaintojen verraten suuri määrä aiheutti sen, että niiden käsittely vei paljon aikaa projektiryhmiltä. Taulukko 15 kuvaa havaintojen määrän ja tyyppin eri projektien eri sprinteissä.

TAULUKKO 15 Käytettävyydestauksen havaintojen lukumäärä ja tyyppi eri sprinteissä

Pro- jekti	Sprint- ti	Havaintojen lukumäärä	Havainnon tyyppi		
			Virhe järjes- telmässä	Käytettävyyso- ngelma	Jatkokehi- tystoive ¹⁹
RETU	1	33	6	33	0
PA	1	36	4	34	2
PA	2	33	3	29	4
SH	1	28	1	27	1
RETU	2	46	5	46	0
PA	3	21	3	17	4
HE	1	58	4	52	6

¹⁹ Jatkokehitystoiveisiin on laskettu PA-projektin sprintti ensimmäisen ja kolmannen sprintin osalta kaikki liitännäisprojektin havainnot, jotka projektipäällikön mukaan toimitettiin asiakkaalle jalostuksen jälkeen.

Käytettävyydestauksen havainnot on luokiteltu *järjestelmän virheisiin*, jolloin järjestelmä on toteutettu virheellisesti tai määrittelyn vastaisesti, *käytettävyysoongelmiin*, eli ongelmiin jotka eivät johdu määrittelyn vastaisesta toteutuksesta vaan esimerkiksi ongelmallisesta määrittelystä, sekä *jatkokehitystoiveisiin*, joita ei ole sisällytetty projektin määrittelyyn tai ne on eksplisiittisesti rajattu projektin ulkopuolelle. Tämä jaottelu on projektinhallinnallisesta näkökulmasta ja erityisesti toteutusta ohjaavasta näkökulmasta relevantti: jatkokehitystoiveet rajataan kuluvan projektin ulkopuolelle ja määritellään erikseen, virheet ovat pakosti korjattavia ja usein virheellisen toteutuksen seurausta, käytettävyysongelmat taas voivat olla sellaisia, joihin reagoidaan jonkinlaisella korjauksella tai sellaisia, joihin ei syystä tai toisesta reagoida. Käytettävyysongelmat täytyy kuitenkin erikseen käsitellä oikean reagoitavan ratkaisemiseksi. Virheitä oli kaiken kaikkiaan noin 11 % havainnoista, jatkokehitystoiveita 13 %, ja loput havainnoista olivat käytettävyysongelmia. Kuvio 15 havainnollistaa tätä suhdetta.



KUVIO 15 Virheiden ja jatkokehityspiirteiden osuus havainnoista

7.4.2 Tulosten hyväksyntä ja hyödyllisyys

Verrattuna taulukkoon 15, taulukko 16 kuvaa havainnot luokiteltuna sen mukaan, miten projektiryhmä reagoi niihin, ei niinkään sen mukaan, millä aikataululla niihin reagoidaan. Havainnot on siis jaettu (1) havaintoihin, jotka lisättiin työlisterille uusina havaintoina, (2) havaintoihin, jotka olivat jo listalla vanhoina havaintoina, (3) havaintoihin, jotka toimitettiin asiakkaalle arvioitavaksi ja (4) havaintoihin jotka hylättiin joko duplikaatteina, pienimerkityksisinä, määrittelyn vastaisina sekä (5) havaintoihin, jotka jätettiin täysin huomiotta muusta syystä.

Duplikaattien puute lienee ainakin heikko signaali sen puolesta, että havaintojen raportointimalli oli toimiva myös sen ehkäisyssä, ettei samoja havain-toja raportoitu useasti. Heuristisen läpikäynnin tulokset HE-projektissa hyödynnettiin osana olemassa olevan asiakkuuden hoitoa. Havaitut virheet toimitettiin ylläpitoprojektille korjattavaksi ja muut havainnot koostettiin asiakkaalle toimitettavaksi osana konseptointiprojektin muita materiaaleja. PA-projektin liiteprojektille tuotettiin ensimmäisen ja kolmannen käytettävyydestestauskerran aikana yhteensä 33 havaintoa, jotka toimitettiin projektipäällikön suodatuksen jälkeen asiakkaalle arvioitaviksi. Havainnot olivat paljolti samoja ulkopuolisen palveluntarjoajan toteuttaman, huomattavasti perusteellisemmän ja raskaamman käytettävyydestestausprojektin kanssa.

TAULUKKO 16 Käytettävyydestestauksen havaintojen hyödyntäminen eri sprinteissä

Pro- jekti	S p r i n t t i	Ha- vain- tojen luku- määrä	Hyväksytyt				Hylätyt			Muu Ei huo- mi- oitu
			Uu- tena työlis- talle	Jo työ- lis- talla	Jat- ko- ke- hitys	Asiak- kaan arvioi- ta- vaksi	Dup- li- kaatti	Pieni- mer- kityk- sinen	Vastoin määritte- lyä tai muutoin virheelli- nen	
RETU	1	33	14	5	0	0	0	2	6	6
PA	1	36	4	7	2	22 ²⁰	0	0	1	0
PA	2	33	20	0	4	4	0	1	4	0
SH	1	28	18	4	1	0	0	1	4	0
RETU	2	46	11	13	0	0	0	19	3	0
PA	3	21	2	0	4	12 ²¹	0	1	2	0
HE	1	58	48	0	6	0	0	3	1	0

Johtuen tulosten raportointimallin kehittymisestä projektin aikana, tutkimuksen alkupuolelta ei ole käytettävissä aivan yhtä tarkkaa tietoa projektiryhmien reaktioista käytettävyydestestauksen havaintoihin ja havaintojen priorisoinnista kuin tutkimuksen loppupuolelta. Tämä osaltaan vaikutti ainakin RETU-projektin ensimmäisen iteraation suureen huomiotta jääneiden havaintojen määrään.

Käytettävyydestestauksen tulosten hyväksyntää seurattiin sillä, kuinka suuri osa käytettävyydestestauksen havainnoista nähtiin sellaisiksi että ne aiheuttavat toimenpiteitä, ja keräämällä projektiryhmien jäseniltä palautetta sekä heille toimitettujen havaintojen hyödyllisyydestä että Nielsen+Krug-mallin toimivuudesta yleensä. Taulukossa 17 esitetään prosenttiosuus, jolla käytettävyydestestauksen havainnot eri projekteissa hyväksyttiin. Tämä taulukko siis jalostaa eteenpäin taulukon 16 kuvausta havaintoihin reagoinnista. Hyväksynnäksi on määriteltä

²⁰ Tähän lukuun laskettu mukaan 21 PA-projektin liitännäisprojektin havaintoa, jotka projektipäällikön mukaan toimitettiin hiukan jalostettuina asiakkaalle arvioitaviksi. Näistä havainnoista ei ollut saatavissa tämän tarkempaa tietoa tämän tutkimuksen puitteissa.

²¹ Kuten alaviitteessä 20, tähän lukuun on laskettu 12 PA-projektin liitännäisprojektin havaintoa.

se, että tulos johtaa johonkin toimenpiteisiin. Tällä määrittelyllä havainnot, jotka on luokiteltu niin pienimerkityksiksi, että niihin ei reagoida vaikka ne olisivatkin sinällään valideja, lasketaan siis hylätyiksi. Tämä melko jyrkältä vaikuttava luokittelutapa puolustaa paikkaansa siksi, että tutkimuksessa kehitetyn kevyen käytettävyydestestauksen mallin tavoitteena on tuottaa nimenomaan hyödyllisiä tuloksia - mikäli tuotettu havainto on niin pienimerkityksinen, että siihen ei voida projektin aikana reagoida, sitä on turha edes raportoida projektiryhmälle. Mikäli pienimerkityksiset havainnot laskettaisiin hyväksytyiksi, varsinkin RETU-projektin toisen iteraation aikaisen testauksen tulosten hyväksyntä näyttäisi erilaiselta: 52 %:n sijaan se olisi 93 %, sillä kyseisessä iteraatiossa hyvin suuri osa havainnoista arvioitiin niin pienimerkityksiksi, että niihin ei validiudesta huolimatta tulla kiinnittämään huomiota. Huomioimatta jätetyt havainnot on myös laskettu hylätyiksi, koska ne eivät johda mihinkään toimenpiteisiin.

TAULUKKO 17 Käytettävyydestestauksen havaintojen hyväksyntä projekteittain ja iteraatioittain

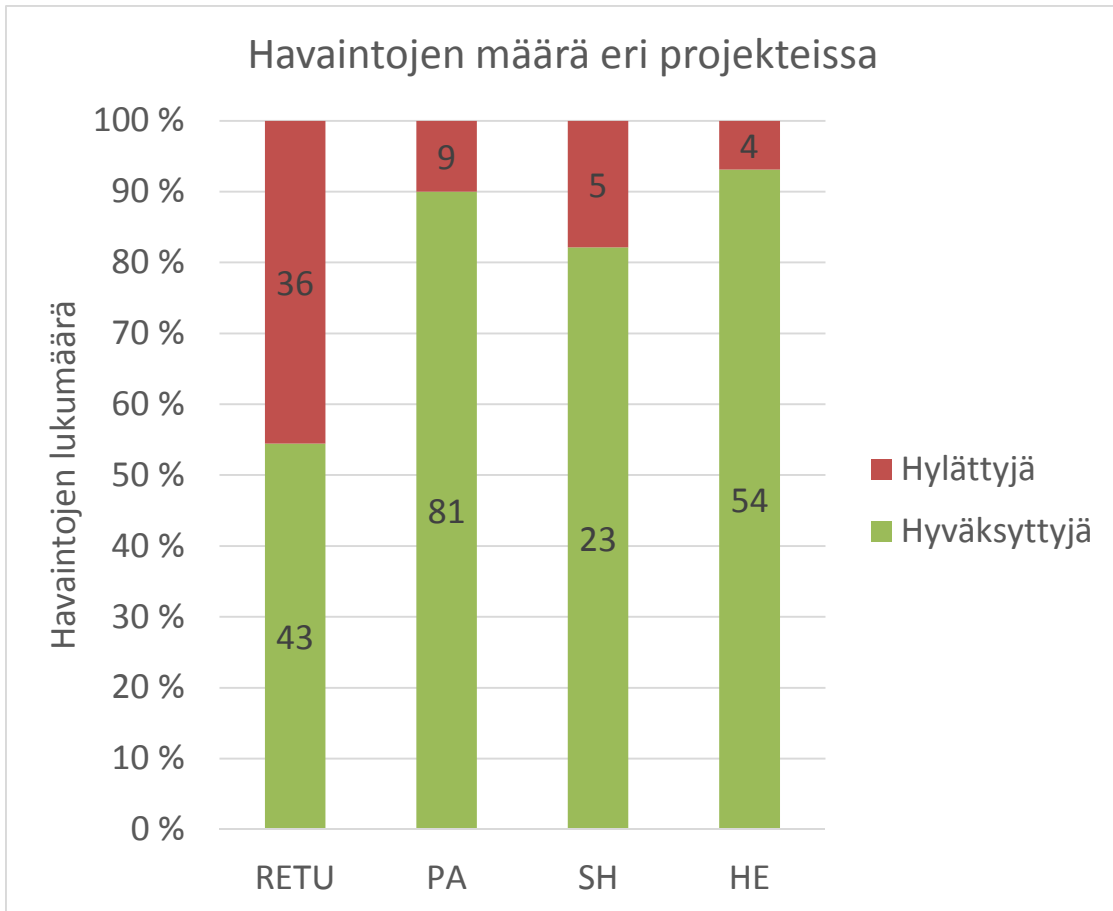
Pro- jekti	Sprintti	Havaintojen lukumäärä	Hyväksytyt	Hylätyt	Ei huomioitu	Hyväk- symis-%
RETU	1	33	19	8	6	58 %
RETU	2	46	24	22	0	52 %
PA	1	36	35 ²²	1	0	97 %
PA	2	33	28	5	0	85 %
PA	3	21	18	3	0	86 %
SH	1	28	23	5	0	82 %
HE	1	58	54	4	0	93 %

Taulukossa 18 kuvataan havaintojen hyväksymisprosentti projektitasolla. Laskelmissa ja luokittelussa on noudatettu samaa logiikkaa kuin taulukossa 17, mutta eri iteraatioiden havaintojen lukumäärät on yhdistetty, jotta tulokset ovat projektikohtaisia. Hyväksymisprosentit on vielä kuvattu havainnollisemmassa muodossa pylväsdiagrammina kuviossa 16.

TAULUKKO 18 Käytettävyydestestauksen havaintojen hyväksyntä projekteittain

Pro- jekti	Sprinttien lukumäärä	Havaintojen lukumäärä	Hyväksytyt	Hylätyt	Ei huomioitu	Hyväk- symis-%
RETU	2	79	43	30	6	54 %
PA	3	90	81	9	0	90 %
SH	1	28	23	5	0	82 %
HE	1	58	54	4	0	93 %

²² Arvo sisältää liitännäisprojektin havainnot (yhteensä 21 kappaletta), jotka projektipäällikön mukaan toimitettiin jalostuksen jälkeen kaikki asiakkaalle, joten ne on tässä tutkimuksessa luokiteltu hyväksytyiksi, asiakkaalle toimitetuiksi havainnoiksi.

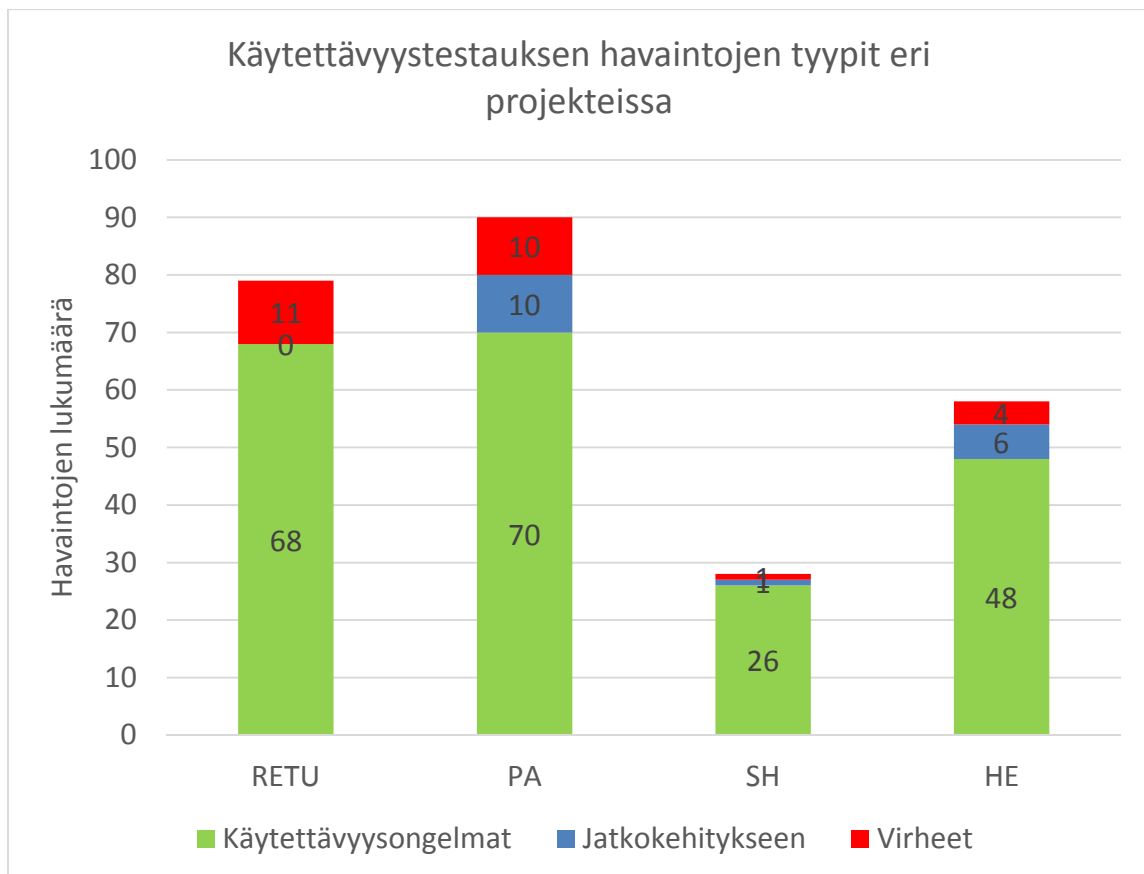


KUVIO 16 Hyväksytyjen ja hylättyjen havaintojen suhde eri projekteissa

Kuvio 17 sisältää tiedon havaintojen tyypeistä projekteittain eriteltyinä virheisiin, jatkokehityspiirteisiin ja käytettävyysongelmiin. Virheet ovat tilanteita, joissa järjestelmä selvästi käyttäytyy virheellisesti ja odottamattomalla tavalla. Jatkokehitykseen luokiteltavat havainnot taasen ovat projektipäällikön mukaan myöhemmin mahdollisesti projektin työlialle otettavia toivottavia ominaisuuksia tai toiminnallisuuksia, jotka eivät kuitenkaan tässä vaiheessa kuulu toteutettaviin ominaisuuksiin. Kuvioista voidaan nähdä, että RETU-projektissa tällaisia ei nähty olevan lainkaan, vaikka projektiryhmälle niitä toimitettiin – tyypillisesti tämä johtui siitä, että toiminnallisuus oli jo työlialla (ja siten hyväksyttiin kyllä havaintona, mutta luokiteltiin vanhaksi havainnoksi) tai oli vastoin määrittelyä (ja siten hylättiin). Virhetilanteita ilmeni jokaisessa projektissa vähintään yksi, jopa siten, että joka käytettävyystestauskerralla ilmeni vähintään yksi virhe.

Tässä tutkimuksessa havaittiin, että ainakin siinä tapauksessa, että koehenkilöiltä ei vaadita tarkkaa kohderyhmävastaavuutta eikä testattavia käyttäjäryhmiä ole useita, ei koehenkilöitä tarvita kovin montaa. Jo neljä koehenkilöä riitti tuottamaan vähemmän merkittävien havaintojen suodattamisen jälkeenkin niin paljon havaintoja, joiden prioriteetti vaihteli kriittisestä kosmeettiseen, että projektiryhmällä ei ollut mitään mahdollisuutta korjata kaikkia havaintoja seuraavan sprintin käytettävyystestauskertaa mennessä, ja erityisesti projektiryhmien

kehittäjäjäsenet kokivat havaintojen määrän jo liiankin suureksi. Tämä oli erityisen selvää ensimmäisellä testauskerralla. Tällöin on mielekkäämpää testata hiukan harvemmillä koehenkilöillä, mutta mahdollisesti useampia kertoja, jotta ainakin merkittävimmät havainnot saataisiin aina korjattua testauskertojen välillä, eivätkä koehenkilöt törmäisi niihin yhä uusissa testeissä aina uudestaan. Myös projektiryhmien jäsenten suhtautuminen käytettävyytestaukseen pysyy tällä tavalla positiivisempänä.



KUVIO 17 Käytettävyytestauksen havaintojen tyypit eri projekteissa

7.4.3 Saatu palaute

Tässä kohdassa tarkastellaan ensin projektien yhteydessä testauskertojen jälkeen projektiryhmiltä saatuja palautteita. Nämä palautteet koskivat paljolti havaintojen raportointitapaa, mutta osin myös havaintojen laatua tai tyyppiä. Sen jälkeen kerrotaan loppupalautekyselyn tuloksista.

RETU-projektin ensimmäisen testauskerran palaute suuntautui lähinnä raportointimallin parantamiseen, kuten erään projektipäällikön kommentti: *"Yleiskommenttina olisi hyvä jos skenaariot, jossa ongelmia tapahtuu, olisi tarkemmin listattuna. Silloin on helpompi tehdä kehitysehdotus."* Tarkempia tietoja havainnosta kävattiin ja varsin yleisellä tasolla oleva käytettävyytestausraportti aiheutti melko paljon kysymyksiä. Samansuuntaista palautetta saatiin PA-projektin ensimmäi-

sestä testauskerrasta, jolloin tuloksia tarkasteleva testaaja ehdotti tulosten ryhmittelyä näkymien perusteella, sekä toivoi raportissa mainittavan sovelletun SUS-arvon havainnollistamista kuvaajien avulla. Projektipäällikkö toivoi myös tietoa näkymästä sekä havaintojen priorisointia havainnon kriittisyyden ja yleisyyden suhteen. Lisäksi tarkempaa tietoa mittareista olisi hänen mukaansa hyvä nähdä. Sekä tieto näkymästä että kuvaajat, havaintojen kriittisyys ja tarkempaa tietoa mittareista otettiin raporttiin mukaan myöhemmissä vaiheissa. Lisäksi näiden kahden ensimmäisen käytettävyydestestauskerran jälkeen raporttiin lisättiin havaintojen kuvaus Excel-muodossa.

PA-projektin toisen käytettävyydestestauskerran havaintoja pidettiin kaikkiaan hyvinä, ja palautteissa monen niistä arvioitiin johtuneen käytettävyydestestauskerralla testatun ylläpito näkymän konseptoinnista. SH-projektin osalta havainnoista toivottiin tarkempaa, esimerkiksi kuvakaappauksin täydennettyä kuvausta. Ratkaisuehdotuksia havaittuihin ongelmiin pidettiin hyödyllisinä ja hyvinä. Projektipäällikkö toivoi, että projektiryhmä olisi päässyt laajemmin mukaan käytettävyydestestauksen suunnitteluun. Kuvakaappausten liittäminen raporttiin osoittautui haastavaksi, mutta niiden toimittaminen pyynnöstä oli helppoa, sillä jokaisesta käytettävyydestestaussesta tehtiin nauhoitus, josta voitiin tarvittaessa ottaa jälkepäin kuvakaappaukset. SH-projektin käytettävyydestestauksen havainnoista koostettiin uusia käyttäjätarinoita ja virheitä projektin työlistalle.

RETU-projektin toisella käytettävyydestestauskerralla Excel-muotoinen havaintojen raportointimalli sai positiivista palautetta selkeytensä vuoksi. Projektin toinen käytettävyydestestaussesta tuotti paljon havaintoja, mutta niistä huomattava osa hylättiin projektipäällikön mukaan niiden matalahkon merkityksen vuoksi, sillä muutoksille ei ollut riittävää liiketoiminnallista syytä. Käytettävyydentuntijan mukaan RETU-projekti kärsi käytettävyydestestauksen suorituksista huomattavasti optimaalista tilannetta myöhemmässä vaiheessa. Jos testaus olisi tehty aiemmin, olisi myös tulosten käsittelyyn jäänyt enemmän aikaa.

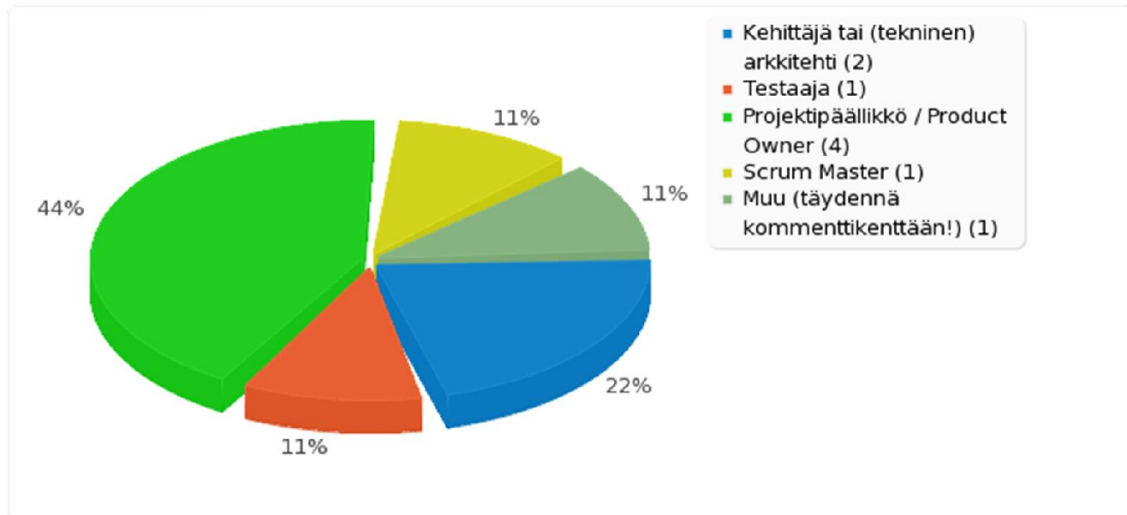
PA-projektin kolmannella käytettävyydestestauskerralla havaintojen määrä väheni huomattavasti aiempiin iteraatioihin verrattuna. Projektissa voitiin myös selvästi havaita aiempien sprinttien testauksessa tehtyjen havaintojen korjausten vaikutus koehenkilöiden kokemaan käytettävyyteen. Havaintojen määrän vähentymisen lisäksi mukautettu SUS-arvo nousi aiempiin käytettävyydestestaukseroihin verrattuna. Mukautetun SUS-arvon mittareita havainnollistavat kuvaajat saivat pääosin kiitosta, vaikka kymmenen mittarin kuvaaminen yhdessä kuvajassa on hankalaa. Excel-muotoinen raportti sai myös kiitosta ja parannusehdotuksia tehtyihin havaintoihin pidettiin hyvinä. Parannusehdotuksina esitettiin muun muassa käytettävyydestestauskerralla hyödynnetyn järjestelmäversion havainnollistamista myös testiraportissa esimerkiksi kuvakaappauksin. Tätä parannusta ei tämän tutkimuksen puitteissa voitu tehdä, mutta videotallenteista tämäkin tieto olisi ollut vaivatonta saada raporttiin mukaan.

Käytettävyydestestauksen loppupalautekysely suunnattiin niiden projektien projektiryhmille, joita varten Nielsen+Krug-mallin mukaista käytettävyydes-

tausta järjestettiin. Potentiaalisia vastaajia oli 10, ja heistä yhdeksän vastasi kyselyyn. Jokaisesta projektista saatiin vähintään 1 vastaaja, ja niistä projekteista joissa oli projektipäällikkö, saatiin vastaus ainakin häneltä ja yhdeltä muulta vastaajalta (projektiryhmän jäseneltä tai käytettävyyssasiantuntijalta).

Kuvio 18 kuvaa loppupalautekyselyn vastaajien roolin projektiryhmässä. "Muu" vastannut henkilö toimi käytettävyyssasiantuntijana. Valtaosa vastaajista oli projektipäälliköitä, ja vastaukset saatiinkin kaikkien mukana olleiden projektien projektipäälliköiltä paitsi HE-projektilta, jolla ei vielä tässä vaiheessa ollut projektipäällikköä.

Käytettävyydestäuksen loppupalautekysely sisälsi 7 kysymystä, joista vain yksi, "Tuottiko käytettävyydestäus hyödyntämiskelpoisia tuloksia?", oli pakollinen. Muut kysymykset koskivat vastaajan taustatietoja tai tarjosivat mahdollisuuden käytettävyydestäusmallin kommentointiin tai kehitysideoiden esittämiseen. Kysely lähetettiin kaikille vastaajille yhtä aikaa tutkimuksen viimeisen käytettävyydestäuskerran valmistuttua. Kysely kokonaisuudessaan löytyy liitteestä 8.



KUVIO 18 Käytettävyydestäuksen loppukyselyyn vastaajien rooli projekteissa

Kyselyn vastauksissa esitettiin jonkin verran parannusehdotuksia käytettävyydestäusprosessiin ja tulosten raportointitapaan. Jotkut parannusehdotuksista, kuten "Olisi hyvä, jos havainnot voisi sortata kriittisyyden mukaan", johtuivat ilmeisesti väärinymmärryksistä – ei esimerkiksi huomattu joitakin havaintojen raportointiin käytettävän Excel-tiedoston sarakkeita, tai mahdollisuutta järjestää havainnot tiedostossa jo olevan kriittisyydenluokituksen perusteella. Joissakin kommentteissa myös keuhuttiin havaintojen raportointitavan myönteistä kehitystä tutkimuksen aikana, esimerkiksi "Havaintolomakkeesta käytiin keskustelua ja se saatiin toimivaksi toiselle kierrokselle", "Ensimmäisen kierroksen löydöksiä ei ollut tarkasti linkattu järjestelmän osiin, toisen vaiheen havainnot oli helppo hyödyntää" ja yksinkertaisesti "Prosessi kehittyi iteraatioiden välillä."

Yleisesti raportointimallia myös pidettiin toimivana, josta kertovat kommentit kuten "Excel on nopea pohja. Kysymysten asettelu oli hyvää", "Hyviä ja konkreettisia kehitysehdotuksia projektin loppukäyttäjänäkökulman käytettävyyteen." Mallia

kehuttiin myös kevyeksi ja vaivattomaksi projektiryhmän näkökulmasta, ”PO/PP:n näkökulmasta se ei kuormittanut paljoakaan projektia vaan tapahtui hyvin itseohjautuvasti”, ”Valmiit ehdotukset helpottivat” ja ”Projektilta ei vaadittu paljon erityisjärjestelyitä etukäteen, itse testaus toimi itsenäisesti ja jututtamieni testaaajien mukaan konsepti oli hyvä, tuloksista tuli suht selkeä raportti nopeasti.”

Taulukko 19 kuvaa loppupalautekyselyyn vastaajien vastaukset projekteittain. Jokainen käytettävyydestä loppupalautekyselyyn vastanneista projektiryhmien jäsenistä (n=9) piti testauksen tuottamia havaintoja hyödyllisinä. Valtaosan vastaajista mukaan käytettävyydestä tuotti paljon tai pääosin hyödyllisiä tuloksia, kun taas loppujen vastaajien mukaan hyödyllisiä havaintoja saatiin jossain määrin tai ainakin jonkin verran. Lisäksi havaintojen hyödyntämistä (vapaaehtoinen kysymys, ”Oliko käytettävyydestä havaintojen hyödyntäminen helppoa?”) pidettiin melko helppona, mutta selvästi vastaajat suhtautuivat hyödyntämisen helppouteen kielteisemmin kuin havaintojen hyödyllisyyteen. Raportointimallissa tai -käytänteissä on siis vielä puutteita.

TAULUKKO 19 Loppupalautekyselyn tuloksia projekteittain

Pro- jekti	Vastaajien luku- määrä ²³	Tuottiko käytettävyydestä hyödyllisiä tuloksia?			Oliko käytettävyydestä havaintojen hyödyntäminen helppoa?		
		Kyllä, paljon tai pääosin	Kyllä, jossain määrin tai jonkin verran	Ei	Oli erittäin helppoa ja suoraviivaista	Oli mutta ei kovin helppoa	Ei
RETU	2	1	1	0	0	2	0
PA	5	3	2	0	2	1	0
SH	4	2	2	0	2	1	0
HE	1	1	0	0	0	1	0

7.5 Nielsen+Krug-malli vs. Samlinkin vaatimukset

Samlinkin käyttöön tulevalle käytettävyydestä mallille esiteltiin kohdassa 6.2.1 seuraavat vaatimukset:

1. Jonkinlainen käytettävyydestä on saatava toteutettua jopa projektissa, jossa siihen on uhrattavissa vain muutamia tunteja koko projektin aikana.
2. Mallin on toimittava saumattomassa yhteistyössä SamScrum-mallin kanssa, esimerkiksi integroituna osaksi iteratiivista työtapaa.
3. Muutokset on saatava nopeasti ja luontevasti projektiryhmän hyödynnettäviksi.
4. Työkalut eivät saa olla kalliita, joskaan niiden ei tarvitse myöskään olla ilmaisia.

²³ Osa vastaajista oli mukana useammassa projekteissa

Tapaustutkimuksen perusteella voidaan sanoa tavoitteiden pääosin täyttyneen. Vaatimus 1, joka koskee jonkinlaisen testauksen toteuttamista jopa vain muutamien tuntien ollessa käytössä, saadaan täytettyä hyödyntämällä heuristista evaluointia. Heuristista evaluointia ei tässä tutkimuksessa käytetty aidosti yksinään, vaan osana heuristista läpikäyntiä. Heuristiseen läpikäyntiin, tulosten raportointi mukaan lukien, kului noin 8,5 tuntia, ja tästä valtaosa kului kognitiiviseen läpikäyntiin. Heuristiseen evaluointiin voidaan arvioida kuluneen noin 3-4 tuntia. Myös koehenkilöitä hyödyntävä käytettävyytestaus saadaan toteutettua kaikki työvaiheet mukaan lukien kahdessa työpäivässä, mikäli koehenkilöiden rekrytoinnissa onnistutaan hyvin.

Vaatimus 2 käsittelee mallin sovittamista Samlinkin projektityömalli SamScrumiin. Käytettävyytestauksen yhteensovittaminen iteratiivisen työskentelytavan kanssa osoittautui pääosin ongelmattomaksi. Hankaluuksia kuitenkin aiheutti testaaminen mahdollisimman aikaisessa vaiheessa: toisaalta projektiryhmä ei koe, että testattavaa olisi välttämättä tarpeeksi, toisaalta testattavaa ympäristöä ei välttämättä ole valmiina. Seuraus myöhäisessä vaiheessa testaamisesta on kuitenkin muutosten tekemisen muuttuminen hankalaksi.

Kolmas vaatimus, muutosten nopea toimitus, onnistui korkeintaan kohutuullisesti. Tulokset saatiin toimitettua projektiryhmille riittävällä tarkkuudella viimeistään viimeisiä testaussessioita seuraavana päivänä. Käytännössä projektiryhmät eivät useinkaan ehtineet käsitellä tuloksia ennen kuin päiviä myöhemmin, joten tulosten käsittelyssä käytettävyytestaukseen kuluva aika ei ollut kriittinen tekijä. Toisaalta projektiryhmille toimitettujen havaintojen määrä aiheutti jo osaltaan sen, että niiden läpikäynti saattoi olla hidasta.

Työkalujen edullisuus, neljäs vaatimus, oli kenties kaikista helpoimmin täytettävissä. Työkaluista kallein oli käytettävyytestaukseen käytettävä tietokone, joka tässä tapauksessa saatiin kuitenkin kehityskäytöstä poistettujen koneiden reservistä ilman ylimääräisiä kuluja. Laitteistovaatimukset käytettävälle koneelle riippuvat sekä testattavasta järjestelmästä että käytettävissä työkaluista, mutta ainakin tässä tutkimuksessa käytetyt työkalut olivat varsin kevyitä. Optimal Workshop-palvelu saatiin pienimuotoiseen käyttöön ilmaiseksi. Mikään testauskoneelle asennetuista sovelluksista ei maksanut mitään tai saatiin käyttöön yrityksen olemassa olevien lisenssien kautta (esimerkiksi Microsoft Office). LimeSurvey on myös ilmainen ohjelmisto. Tässä tutkimuksessa sen käyttöön tarvittava palvelin saatiin yhteistyökumppanin kautta, mutta jatkossa kevyen Linux-palvelimen ylläpidosta kertyy pienehkö kustannus.

Tiivistetysti voitaneen sanoa, että ensimmäinen vaatimus saatiin pääosin täytettyä. Lyhimmillään testaukseen kului noin päivä, mutta pelkkä heuristinen evaluointi olisi ollut tehtävissä vielä nopeammin. Toinen vaatimus saatiin täytettyä pienellä varauksella, sillä testaustapa toimi erittäin hyvin yhteen iteratiivisen työskentelytavan kanssa, mutta testauksen aloittamiselle riittävän aikaisessa vaiheessa projektia on haastavaa saada hyväksyntä projektiryhmiltä. Myös kolmas vaatimus pääpiirteissään täytettiin, ja niiltä osin kuin toiminnassa oli hitautta, hitauden merkitys jäi vähäiseksi, sillä projektiryhmät eivät ehtineet kuitenkaan

käsitellä tuloksia välittömästi. Neljäs vaatimus saatiin täysin täytettyä, sillä käytettävyydestestauksessa käytettyä, melko edullista tietokonetta ja pieniä materiaalikuluja lukuun ottamatta käytetyt työkalut olivat ilmaisia.

7.6 Yhteenveto

Tässä luvussa kuvattiin, miten Nielsen+Krug-mallia hyödynnettiin käytännössä Samlinkin projektityöskentelyssä ja millaisia kokemuksia siitä saatiin. Todettiin, että mallin mukaisen käytettävyydestestauksen toteuttamiseen ei tarvita monimutkaisia työkaluja ja testauksen järjestelyt olivat melko yksinkertaisia. Myös koehenkilöiden rekrytointi koehenkilötietokantaa varten pystyttiin toteuttamaan varsin kevyesti.

Käytettävyydestestauksen valmistelut olivat myös varsin kevyet ja käytettävyydestestaus suunnitelma laadittiin olemassa olevien materiaalien pohjalta käyttäen pohjana esimerkiksi testaus suunnitelmaa, käyttötapauksia, testitapauksia tai jotain muuta määrittelyä. Luvussa kuvattiin myös käytettävyydestestaus session kulku ja sovellettavien menetelmien valinta siten, että ohjeiden perusteella testaus on mahdollista suorittaa onnistuneesti.

Tutkimuksen aikana raportointimallia täydennettiin Excel- taulukolla, joka sisälsi yksityiskohtaisen kuvauksen havaituista ongelmista sekä korjausehdotukset niihin ja mahdollisuuden järjestää havaintoja esimerkiksi prioriteetin tai yleisyyden mukaan. Projektiryhmä voi mallin mukaan käsitellä havaintonsa haluamallaan tavalla ja aikataululla, mutta joka tapauksessa havainto joko hylätään, se päättyy työlisterille tai virheidenseurantajärjestelmään.

Nielsen+Krug-mallia kokeiltiin yhteensä 30 koehenkilöllä, kolmella eri menetelmällä neljässä eri projektissa yhteensä seitsemän käytettävyydestestauksen aikana. RETU-projektin kahden käytettävyydestestauksen aikana havaintoja saatiin 79 kappaletta ja testaukseen kului sprintin aikana keskimäärin hiukan alle 2 vuorokautta. Projektissa kuitenkin havaintojen hyväksyntä oli projekteista kehnoin, sillä melkein puolet havainnoista hylättiin, joskin usein liiketoiminnallisiin syihin tai projektin määrittelyyn vedoten. PA-projektissa testausta tehtiin kolmen käytettävyydestestauksen verran, ja koehenkilöinä hyödynnettiin sekä asiakkaan edustajia, sisäisiä koehenkilöitä että ulkoisia koehenkilöitä. Havaintoja saatiin yhteensä 90 hyväksyntäprosentinkin ollessa 90 %. Huomattava osa havainnoista saatiin lisäksi korjattua ennen kolmatta testauskertaa, mikä johti käytettävyyden parantumiseen mukautetulla SUS-mittaristolla mitattuna. Aikaa testaukseen kului iteraatiota kohden hiukan yli kaksi päivää, josta kuitenkin maininnan arvoinen poikkeus oli toinen käytettävyydestestauskerta jonka kaikki valmistelut, testaus ja analysointi tehtiin yhden työpäivän aikana. PA-projektin testauksen ohessa tuotettiin havaintoja myös projektin liitännäisprojektin käyttöön, ja erityisesti korttien lajittelun tuottama tieto oli sille hyödyllistä. SH-projektin yhteen käytettävyydestestaukseen kului hiukan alle kaksi päivää ja havaintoja saatiin yhteensä 28 kappaletta, ja niiden hyväksyntä oli 82 %. PA-, RETU- ja SH-

projekteissa hyödynnettiin yksinkertaista ääneen ajattelua ja PA-projektissa lisäksi korttien lajittelua. HE-projektissa, aiemmista projekteista poiketen, hyödynnettiin vain heuristista läpikäyntiä, ja havaintoja saatiin yhden käytettävyydestaustakerran aikana peräti 58 kappaletta. Havaintojen hyväksyntä oli 93 %

Projektien käytettävyydestaustakohtainen kustannus vaihteli 625,00 eurosta 1233,50 euroon niiden projektien osalta, joissa hyödynnettiin koehenkilöitä, ja vain heuristista läpikäyntiä hyödyntävä käytettävyydestaustakerta maksoi 425,00 euroa. Koehenkilökohtaiset kustannukset vaihtelivat 126,42 euron ja 203,44 euron välillä. Havaintokohtaiset kustannukset vaihtelivat heuristisen läpikäynnin 7,33 eurosta runsaasti koehenkilöitä ja muun muassa korttien lajittelua (jonka tekeminen kuluttaa aikaa yksinkertaistetulta ääneen ajattelulta tuottamatta kuitenkaan uusia havaintoja) hyödyntävän PA-projektin 29,39 euroon.

Tutkimuksen lopuksi tehtiin vielä palautekysely projektiryhmien jäsenille. 9 vastaajasta kaikki pitivät käytettävyydestausta Nielsen+Krug-mallin avulla vähintään jossain määrin hyödyllisenä, ja yli puolet (5) piti sitä erittäin hyödyllisenä. Kommenttien perusteella testausta kannattaa jatkaa tulevaisuudessakin, mutta mallissa on vielä kehitettävää muun muassa havaintojen raportoinnin suhteen.

8 POHDINTA

Tämän tutkimuksen tavoitteena oli tutkia kevyen käytettävyydestestauksen yhteensovittamista ketterän ohjelmistokehityksen kanssa. Työssä rakennettiin kirjallisuuskatsauksen pohjalta Nielsen+Krug-mallia, joka koostuu kolmesta käytettävyydestestausmenetelmästä. Malli yhdistettiin Samlinkin SamScrum-projektimalliin ja testattiin neljässä ohjelmistokehitysprojektissa. Seuraavassa kuvataan ensin Nielsen+Krug-malli ja verrataan sitä kirjallisuuteen. Toiseksi luvussa kuvataan tiivistetysti mallin käyttöä projekteissa ja niistä saatuja kokemuksia. Kokemuksia verrataan vastaaviin aiempiin tutkimuksiin. Kolmanneksi luvussa kerrotaan, miten tutkimuksen tuloksia voidaan hyödyntää. Lopuksi suoritetaan tutkimuksen reliabiliteetin ja validiteetin tarkastelu.

8.1 Nielsen+Krug -malli

Tutkimuksessa kehitettiin käytettävyydestestausmalli, joka koostuu kolmesta käytettävyydestestausmenetelmästä, korttien lajittelusta (Nielsen, 1995a; Rubin, 2008), heuristisesta evaluoinnista (Kane, 2003; Nielsen, 1995c; Rubin, 2008) ja yksinkertaisesta ääneen ajattelusta (Kane, 2003; Krug, 2000; Nielsen, 1994b, 1995a). Malli nimettiin Nielsen+Krug-malliksi. Malli määrittelee testauksen piiriin kuuluvat aktiviteetit sekä mahdollisesti tuotettavan dokumentaation. Sen vaiheet ja toimintatavat sovitettiin Samlinkilla hyödynnettävän SamScrum-ohjelmistokehitysmallin toimintatapoihin sopiviksi.

Mallin osia ovat aktiviteetteina käytettävyydestestaus, havaintojen tärkeyden arviointi projektiryhmän kesken ja niiden vienti työlistalle. Mallin tuloksina toteutusiteraatioiden aikana voidaan saada sellaisia dokumentteja kuten käytettävyydestestitapaukset ja julkaisukohtaisesti käytettävyydestestauksen yhteenveto.

Mallin merkittävimmät erot aiempiin integrointiesityksiin (Lee ym., 2009; Lee & McCrickard, 2007; Parsons ym., 2007; Singh, 2008; Sy, 2007) verrattuna löytyvät sen näkökulmasta. Malli ei ota kantaa käytettävyyden suunnitteluun tai

koko käytettävyyssuunnitteluprosessiin, vaan keskittyy nimenomaan käytettävyydestauksen ja ketterän ohjelmistokehityksen yhteensovittamistavan löytämiseen. Tämän rajauksen ansiosta löydettiin laajalti integroitavissa oleva, joustava ja kevyt käytettävyydestausmalli. Nielsen+Krug -mallissa keskitytään yleisen käytettävyydestauksen, kuten käytettävyysslaboratoriota vaativan laajan mitta-kaavan käytettävyydestauksen sijasta kevyisiin käytettävyydestauksen menetelmiin. Tässä mielessä asiaa tarkastellaan samoin kuin McGinnin ja Changin (2013) RITE+Krug-menetelmässä, joskin erona on keskittyminen sidosryhmien sitouttamisen sijasta toisaalta testauksen resurssivaatimuksiin ja toisaalta toimivan tulosten raportointitavan löytämiseen. Molemmissa tutkimuksissa keskitytään myös huomattavasti toimivien testausapojen hyödyntämiseen.

Mallin keveydessä ei pyritty yhtä pitkälle kuin Martyn ja Twidalen (2005) äärimmäisen ketterän käytettävyydestauksen mallissa. XSBD-malli (Lee ym., 2009) keskittyy skenaarioihin perustuvan käytettävyyssuunnitteluratkaisun kehittämiseen, mutta integrointi tapahtui XP:n suhteen, johon erona tässä tutkimuksessa oli integrointi scrumin kanssa. Kuten UTUM-mallissakin (Winter ym., 2011), myös Nielsen+Krug-mallissa yhteensovitus aloitettiin olemassa olevasta ketterästä toimintatavasta, johon tuotiin osaksi käytettävyydestausaktiiviteetti, joka aiheutti olemassa olevalle toimintatavalle formaaleja vaatimuksia koskien käytettävyydestauksen havaintojen raportointia. XSBD-mallissa taas hyödynnetään väittämiä (engl. claims) ja skenaarioita, jotka koettiin tämän tutkimuksen kohdeorganisaatiolle vieraiksi. Niiden sijasta valittiin scrumin piiristä käyttäjätarinat ohjaamaan käytettävyydestausuunnitelman valmistelua. Tavassa, jolla mallin toiminnot sovitettiin yhteen scrumin kanssa, otettiin mallia myös Northropin (2011, 2012) integrointitavasta, jossa käytettävyydestaus on yhtenä kohteena työlistalla, mutta erona varsinaisiin käyttäjätarinoihin se ei varsinaisesti valmistu lainkaan. Vertailu Northropin malliin muutoin ei ole mielekäästä, sillä hänen mallinsa on hyvin yleisellä tasolla eikä juuri anna konkreettisia ohjeita.

8.2 Kokeilu

Tässä alaluvussa käsitellään Nielsen+Krug-mallin kokeilua käytännössä ja siitä saatuja tuloksia. Kokeilun toteutustapaa ja tuloksia verrataan aiempiin tutkimuksiin. Ensin kerrotaan kokeilun tavoitteista ja lähtökohdista sekä kokeilussa mukana olleita projekteista. Sitten kuvataan kokeilun järjestelyjä. Kolmanneksi kuvataan kokeilussa havaittuja käytettävyydestauksen kustannuksia ja mallin soveltamisesta saatuja hyötyjä verraten niitä aiempien tutkimusten havaintoihin niiltä osin kuin se on mahdollista ja mielekäästä.

8.2.1 Kokeilun tavoitteet, lähtökohdat ja järjestelyt

Nielsen+Krug-mallin kokeilun tärkein tavoite oli saada yksityiskohtaisempaa tietoa mallin hyödyntämisestä käytännössä, sekä myös kehittää joitakin mallin piirteitä, kuten testauksen havaintojen raportointikäytänteitä, paremmin toimiviksi. Tämän lisäksi haluttiin kerätä tietoa mallin hyödyntämisestä koituvista kustannuksista, erityisesti mallin hyödyntämiseen kuluvasta ajasta, sekä tietysti mallin hyödyntämisestä koituvista hyödyistä.

Nielsen+Krug-mallia kokeiltiin käytännössä Samlinkin projektityöskentelyssä. Testauksen kohteeksi valittiin projekteja varsin opportunistisesti: tavoitteena oli valita keskenään erilaisia projekteja, mutta erittäin suuri vaikutus oli myös projektien sopivalla aikataululla. Kaikki projektit soveltuivat kuitenkin tapaustutkimukseen siksi, että projektipäälliköiden arvioiden perusteella ne voisivat hyötyä käytettävyytestestauksesta.

Aiempiä tapaustutkimuksia erityisesti käytettävyytestestauksen yhteensovittamisesta ketterän ohjelmistokehityksen kanssa ei ole tehty kovin paljon, vaikka yhteensovittamista varmasti tehdään käytännössä jatkuvasti. Kuten useissa aiemmissakin malleissa (mm. Lee ym., 2009), myös tässä tutkimuksessa käytettävyytestaus koehenkilöitä hyödyntäen arvioitiin tärkeimmäksi käytettävyytestestauksen tavaksi. Lee ja McCrickard (2007) pitivät oman tapaustutkimuksensa perusteella äärimmäisen tärkeänä sitä, että käytettävyytestestauksen tulokset saadaan projektiryhmissä käsiteltyä. Tässä tutkimuksessa tämä pyrittiin varmistamaan sillä, että havaintoihin tarjottiin aina ratkaisumalli ja ne raportoitiin niin tarkasti, että projektiryhmä hahmottaa, mistä niissä on kyse. Tämä eroaa suuresti ainakin Krugin (2006, 2010) kevyestä tavasta raportoida käytettävyytestestauksen havainnot, jolloin tulosten purkamiseen käytetään esimerkiksi lounastauko ja käytettävyytestestausraportti koostuu muutamasta ranskalaisesta viivasta. Tämä tutkimus on toisaalta poikkeuksellisen laaja siksi, että mallin soveltamisen kohteena olevia projekteja oli varsin useita. Projektit itsessään eivät olleet kuitenkaan kovin laajoja.

8.2.2 Testauksen kustannukset

Käytettävyytestestauksen kustannuksia on eritelty aiemmin luvussa 7.3. Tässä tutkimuksessa on voitu laskea yksiselitteisesti hinta sprintin ja koehenkilön kustannuksille niiden projektien osalta, joissa käytettävyytestestausta tehtiin.

Kustannuksia kertyi testaukseen kuluvasta ajasta ja koehenkilöiden palkkioista. Työkalujen tai testaukseen käytettävän työaseman hankinnasta ei tässä tutkimuksessa aiheutunut kustannuksia. Projektikohtaiset kustannukset vaihtelivat 425,00 ja 2644,75 euron välillä riippuen iteraatioiden ja koehenkilöiden määrästä sekä hyödynnettävistä käytettävyytestestausmenetelmistä. Käytettävyytestestauskertakohtainen kustannus vaihteli 425,00 ja 1233,50 euron välillä. Koehenkilökohtainen kustannus, niissä projekteissa joissa koehenkilöitä hyödynnettiin, oli 126,42 - 203,44 euron välillä. Poikkeuksetta merkittävin kustannuserä oli käytet-

tävyystestaajan testaukseen käyttämä aika, korkeimmillaan koehenkilöistä aiheutuneet kustannukset olivat yhteensä 250,00 euroa käytettävyystestauskerran aikana, jolloin hyödynnettiin viittä sisäistä koehenkilöä.

Kustannusten vertaaminen aiempaan tutkimukseen on hankalaa, sillä eri ajankohtina, eri yhteiskunnissa, eri organisaatioissa ja eri valuutoissa ilmaistut kustannukset eivät yksinkertaisesti ole keskenään vertailukelpoisia. Tässä tutkimuksessa kehitetyn mallin keveyden hahmottamiseksi tehdään kuitenkin muutamia vertailuja kustannuksista ja erityisesti ajankäytöstä tämän ja aiempien tutkimusten välillä.

Käytettävyystestauksen valmisteluun kuluva ajasta valtaosa kului koehenkilöiden rekrytointiin tai testaussessioiden aikataulutukseen. Itse testaussessioiden valmistelu oli lopulta varsin nopeaa, ja samoin oli käytettävyystestaus suunnitelman laatiminen testaussuunnitelman tai esimerkiksi käyttötapausten pohjalta. Aiemmistä tutkijoista Nielsen ei useimmissa tutkimuksissaan ota kantaa siihen, kuka rekrytoinnin hoitaa ja kuinka kauan se kestää, joskin hän arvioi hyvin suuntaa-antavaksi sprinttikohtaisesti käytettäväksi ajaksi vähintään 50 tuntia ja mainitsee esimerkkinä 11 arvioijalla teetetyn heuristisen evaluoinnin vieneen 105 tuntia (Nielsen, 1994b). Tämä tarkoittaa siis noin 9,55 tuntia arvioijaa kohden, mikä sisältää myös projektiryhmän havaintojen arviointiin kuluttaman ajan. Krug (2000, 2006) hoiti rekrytoinnit ja aikataulutuksen itse, ja toisaalta rohkaisikin rekrytoimaan vaikka naapureita, mutta hän ilmoittaa varsin epätarkasti eri vaiheisiin kuluneen ajan. Hänen tapauksessaan testaukseen sanotaan kuluvan aamupäivä, ja hyvin epävirallinen ja yleiselle tasolle jätettävä kooste tulokista käsitellään jo iltapäivällä, esimerkiksi tunnin mittaisen session aikana, mutta valmisteluun kuluva aika jää hämärään. Moni muukaan testaukseen ja sen valmisteluun kuluva ajasta arvioita esittäneistä tutkijasta (kuten Nielsen, 1995a; Northrop, 2011, 2012), ei ilmeisesti yleensä ole ottanut laskelmiin mukaan rekrytointiin ja aikataulutukseen kuluvaan aikaan, ja silti merkittävä osa ajasta on heidän tapauksissaan kulunut erilaisiin valmisteluihin. Monikaan tutkijoista, kuten McGinn ja Chang (2013), ei edes ilmoita valmisteluihin kulunutta aikaa. Toisaalta heidän RITE+Krug-menetelmässään, joka osin toimii Samlinkin mallin esikuvana, kuuluu varsinaiseen testaukseen ja tulosten käsittelyyn kaksi työpäivää neljällä koehenkilöllä. Sy (2007) taasen mainitsee vain, että Autodeskillä käytettävyystestaussessiot kestivät tunnin tai kaksi riippuen siitä, mitä testattiin, mutta valmisteluun tai tulosten purkamiseen kuluvaan aikaan ei eritellä. Tavattoman moni tutkija yksinkertaisesti mainitsee koehenkilöitä hyödyntävään käytettävyystestaukseen kuluvan ajan olevan mittava (mm. Desurvire, 1994; Holzinger, 2005; Nielsen, 1992b, 1994a).

Tässä tutkimuksessa havaittiin, että kevyt käytettävyystestaus vähensi erityisesti valmisteluun kuluvaan aikaan, kun käytettävyystestaus suunnitelma laaditaan joustavasti jo valmiiden materiaalien pohjalta ja osittain myös testaajan intuition luottaen. Tulosten analysointi ja raportointi vei huomattavasti aikaa, vaikka raportointia varten laadittiin dokumenttipohja. Osa ajasta kului kuitenkin juuri tämän dokumenttipohjan laatimiseen ja hiomiseen.

Krugin (2000, 2006) nopeuteen testauksessa ei päästy, koska raportoinnissa pyrittiin kattavuuteen: sen sijaan, että havainnoista poimitaan vain kaikista ilmeisimmät ja vakavimmat, haluttiin suodatus tehdä vasta projektiryhmän toimesta, jolloin testaajan tehtäväksi jää varsin suuren havaintomäärän jalostaminen sellaiseen muotoon, jossa projektiryhmä voi käsitellä sen ja päättää, mitkä havainnoista viedään työlistalle. Projektin aikana täsmentyi myös, että käytettävyydestä paljastaa varsinaisten käytettävyyso Ongelmien lisäksi uusia ominaisuuksia ja myös virheitä (engl. bugs), joista erityisesti jälkimmäiset on syytä saattaa tiedoksi myös projektiryhmälle, mikä aiheuttaa merkittäviä vaatimuksia havaintojen raportointitarkkuudelle. Mikäli havaintoja ei kuvata tarpeeksi tarkasti, aikaa kuluu tarkennuspyyntöihin vastaamisessa tai havaintoja voi jäädä hyödyntämättä.

Kustannusten osalta on todettava, että koehenkilöiden palkkioissa ja muissa oheiskustannuksissa säästettiin verrattuna useisiin aiempiin tutkimuksiin. Verrattuna Krugin (2000, 2006) maksamiin palkkioihin (50–300 dollaria tunnilta) koehenkilöt saatiin rekrytoitua melko edullisesti, ulkoisten koehenkilöiden palkkion ollessa noin 10 euroa tunnin mittaisesta sessiosta ja sisäisten koehenkilöiden laskennallisten kustannuksen ollessa noin 50 euroa tunnilta. Lisäksi johtuen mallin suoraviivaisuudesta ja kulujen vähäisyydestä saatiin sprinttikohtaisen testauksen hinta pidettyä matalampana kuin aiemmissa tutkimuksissa, pois lukien Krugin menetelmä, johon verrattuna tulosten jalostamiseen ja raportointiin käytettiin enemmän aikaa. Krug (2000) myös mainitsee projektin aikaiseen käytettävyydestä kuluvaan esimerkiksi 3 900 dollaria, joka nykyarvoon ja euroiksi muutettuna²⁴ vastaa noin 3920 euroa. Tässä tutkimuksessa pitkäkestoisimman PA-projektin aikana tehtiin kolme iteraatiota käytettävyydestä, ja tämän projektin käytettävyydestä kokonaiskustannus oli noin 2640 euroa. Toiseksi kalleimmaksi tuli RETU-projekti, jossa koehenkilöt olivat sisäisiä, ja täten kalliimpia. RETU-projektin käytettävyydestä maksoi kokonaisuutena 1760 euroa. HE-projekti oli kokonaisuutena projekteista edullisin, johtuen siitä, että siinä sovellettiin vain yhtä menetelmää, heuristista läpikäyntiä, ja täten testaus oli varsin vaivaton ja nopea tehdä.

Kaikkien projektien koehenkilökohtainen kustannus oli huomattavasti matalampi kuin esimerkiksi Nielsenin ja Landauerin (1993) arvio kevyen käytettävyydestä koehenkilön hinnasta (673 dollaria, noin 491 euroa²⁵) – itse asiassa kalleimmassakin projektissa koehenkilökohtainen kustannus oli alle puolet Nielsenin ja Landauerin arviosta, vaikka tässä tutkimuksessa projektin käytettävyydestä järjestämisen kiinteät kustannuksetkin on laskettu mukaan koehenkilökohtaiseen kustannukseen. SH-projektin osalta käytettävyydestä oli varsin vaivaton ja ongelmaton, ja siten sen kustannuksetkin jäivät erittäin mataliksi. Tämän tutkimuksen osalta myös havaintokohtainen kustannus on laskettu.

Käytettävyydestä kerran (projektin näkökulmasta siis iteraatiokohtaisen käytettävyydestä) hinta oli heuristisen evaluoinnin osalta 425,00 euroa ja

²⁵ Muuntamisessa käytetty Aktian valuuttamuunninta (Aktia Oyj, 2014)

yksinkertaista ääneen ajattelua sekä mahdollisesti korttien lajittelua hyödyntävien projektien osalta 625,00 ja 1233,50 euron välillä. Aiemmissa tutkimuksissa on niukasti mainintoja vastaavista kustannuksista, ja esimerkiksi Krugin (2006) ilmoittaman 300 dollarin lisäksi ajankäytöstä koituvia kustannuksia joudutaan arvailemaan. Käytettävyykertoaikohtaisia kustannuksia ei ole ehkä tässä tutkimuksessa mielekästä vertailla aiempaan tutkimukseen, sillä jo projekti- ja koehenkilökohtaisista kustannuksista on saatu melko hyvä kuva siitä, että Nielsen+Krug-malli on erittäin edullinen aiempiin malleihin verrattuna.

8.2.3 Testauksen hyödyt

Nielsen+Krug-malli tuotti vähäisin kustannuksin paljon havaintoja erityisesti käytettävyydestä, mutta myös joistakin virheistä järjestelmässä. Mallin todettiin tarjoavan mahdollisuuden käytettävyydestestauksen toteuttamiseen projektin resursseihin ja tyyppiin sopivalla tavalla ja tuottaa havaintoja, joihin on mahdollista reagoida projektiryhmälle parhaiten sopivalla tavalla. Nielsen+Krug-malli havaittiin toteutuksen aikana soveltuvaksi iteratiivisen työskentelytavan kanssa, ei niinkään SamScrum-mallista riippuvaiseksi. Näin ollen sen sovittaminen yhteen minkä tahansa iteraatioihin, kuten scrumin tapauksessa sprintteihin, perustuvan työskentelytavan tai projektimallin kanssa, on todennäköisesti mahdollista.

PA-projektissa testausta päästiin tekemään kolmessa peräkkäisessä sprintissä ja projektiryhmä ehti reagoida huomattavaan osaan käytettävyydestestauksen havainnoista toisen ja kolmannen sprintin välissä. Tämä näkyi välittömästi koehenkilöiden järjestelmälle antamien käytettävyyssarvosanojen paranemisena (ensimmäisen ja toisen käytettävyydestestauskerran arvosanat olivat 78,33 ja 76,25, kolmannen käytettävyydestestauskerran 84) ja myös käytettävyydestestauksen tuottamien havaintojen määrän merkittävänä vähenemisenä. Vaikuttaisi siis siltä, että käytettävyydestestausmalli tuotti havaintoja, joihin esitetyllä tavalla reagoimalla järjestelmän käytettävyyttä voitiin todella parantaa. Lisäksi projektiryhmien kommentit loppupalautekyselyyn antavat tukea sille oletukselle, että käytettävyydestestausta voidaan tehdä myös kevyellä ja projektiryhmän resursseja minimaalisesti kuluttavalla tavalla.

Tässä tutkimuksessa kehitetty malli antaa ohjeet muutaman keskeisimmän käytettävyydestestausmenetelmän soveltamiseen ja tarjoaa valmiin raportointipohjan ja -tavan, jolla käytettävyydestestauksen havainnot voidaan saattaa projektiryhmälle tiedoksi. Mallia pidettiin projektiryhmän kannalta varsin ongelmattomana, ja sen arvioitiin antavan kilpailuetua muihin toimijoihin verrattuna, jos se tuotteistetaan ja otetaan mukaan tarjouksiin²⁶.

Käytettävyydestestauksen tulosten vastaanotto projekteissa oli kaksijakoinen. Toisaalta tuloksiin oltiin tyytyväisiä, toisaalta havaintojen suuri määrä saattoi yllättää projektin kehittäjät ja aiheuttaa negatiivisen vastareaktion. Suuri havainto-

²⁶ Loppupalautteesta poimittuja kommentteja

jen määrä saattoi myös johtaa projektiryhmän suorittamaan aggressiiviseen karshintaan, kuten kävi RETU-projektin toisessa sprintissä, jossa havainnoista hylättiin pienimerkityksisinä 41 %, ja havaintojen hyväksyntäprosentti oli vain 52. Projektikohtaiset tiedot projektiryhmän reagoinnista ja kommentteista havaintoihin löytyy liitteestä 6.

Jo ensimmäisten käytettävyydestaustakertojen tulosten toimituksen jälkeen saadun palautteen perusteella havaittiin, että tärkeämpää kuin tulosten salamanopea toimitus, on tulosten riittävän perusteellinen kuvaaminen, jotta projektiryhmä tietää tarkasti, mikä ongelma on ja osaavat priorisoida sen oikein, ja konkreettinen esitys kunkin ongelman ratkaisemiseksi. Raportointitapaa kehitettiin sellaiseen suuntaan, että havainnot olisi helppo raportoida alusta asti riittävän laajasti, jotta niihin reagointi olisi helppoa eikä tarkentavia tietoja tarvitsisi kysyä. Projektiryhmän jäsenet tai Samlinkin käytettävyyssasiantuntija esittivät havainnoista usein tarkentavia kysymyksiä. Kysymyksiin vastattiin joko tarkentamalla ongelman tai ratkaisuehdotuksen kuvausta, liittämällä siihen kuvakaappauksia tai tarjoamalla mahdollisuus videotaltioinnin tarkasteluun.

Havainnoista huomattava osa hylättiin eri syistä, kaikissa projekteissa yhteensä 21 %. Tyypillinen tilanne oli, että havaintoa itsessään ei kiistetty, eikä sen käyttäjille aiheuttamia ongelmia myöskään yritetty väittää perusteettomiksi, mutta siihen ei aiottu reagoida syystä tai toisesta. Esimerkkejä tällaisista tilanteista on muun muassa (1) rajalliset kehitysresurssit ja merkitykseltään melko pieni havainto tai työmäärältään suuri muutos, (2) järjestelmä toimii määrittelyn mukaisesti ja toiminnallisuuden muuttaminen vaatisi muutosta myös määrittelyyn ja (3) integraatiot tai muut projektit estävät muutoksen tekemisen. Esitettyjä syitä, sinällään perusteltujen havaintojen hylkäämiseen vastaan on mahdollista argumentoida. Ensimmäinen syy olisi voitu paljolti välttää tekemällä testaus aiemmin, jolloin resursseja muutoksen tekoon olisi voinut vielä olla enemmän ja toisaalta muutos olisi myös voinut vaatia vähemmän työtä. Toinen syy vaatii parempaa kommunikaatiota asiakkaan kanssa ja sitä, että asiakas ottaa käytettävyydestaustan tulokset tosissaan. Kolmas syy nivoutuu osittain ensimmäiseen: jos havainto olisi tehty aiemmin, integraatio olisi kenties voitu toteuttaa toisella tavalla.

Kokeilun aikana havaittiin, että käytettävyydestaustaus tuotti myös jonkin verran havaintoja virheistä, jolloin testattava järjestelmä selkeästi käyttäytyi virheellisesti ja esimerkiksi määrittelyn vastaisesti tai päätyi tietyn syötteen seurauksena täysin käyttökelvottomaan tilaan. Näitä oli peräti 10 % kaikista tehdyistä havainnoista. On huomionarvoista, että näitä tilanteita ei lainkaan pyritty luomaan, vaan virheet ilmenivät normaalien käytettävyydestestitapausten suorituksen aikana. Virheitä raportoitiin jokaisen käytettävyydestaustakerran aikana vähintään yksi kappale, ja ne kirjattiin pääosin myös virheinä projektien työlistoille.

Tutkimuksen loppupalautekyselyn perusteella Nielsen+Krug-mallia pidettiin hyödyllisenä ja sen tuottamia havaintoja pääosin helposti hyödynnettävissä olevina. Havaintojen raportointimallia kehitettiin tutkimuksen aikana palautteen perusteella parempaan suuntaan, mutta ilmeisesti jatkokehitystä kaivattiin.

Myös mukautettua SUS-arvoa pidettiin hyödyllisenä ja ongelmien korjausehdotuksia arvokkaina.

8.3 Tulosten hyödyntäminen

Tämän tutkimuksen tuloksia voidaan hyödyntää sekä jatkotutkimuksessa että käytettävyydestestauksen käytännön työssä. Aiempaa empiiristä tutkimusta käytettävyydestestauksen ja ketterän ohjelmistokehityksen yhteensovittamisesta on tehty (mm. Lee ym., 2009; Meszaros & Aston, 2006; Sy, 2007; Talby ym., 2006; Winter ym., 2011), mutta ei kovin paljon. Tämä tutkimus tarjoaa arvokasta lisätietoa yhteensovittamistavoista ja kevyiden käytettävyydestestausmenetelmien hyödyntämisestä järjestelmäkehitysprojekteissa. Jatkotutkimuksen kannalta tämä tutkimus on tuottanut mielenkiintoista dataa kahden keskenään erilaisen mutta samoihin tavoitteisiin pyrkivän tutkimusalan yhteensovittelusta. Tässä tutkimuksessa myös koostetaan laajalti eri näkemyksiä ketterän ohjelmistokehityksen ja käytettävyydestestauksen yhteensovittelusta sekä käytettävyydestestauksen eri lähestymistavoista – erityisesti kevyestä käytettävyydestestauksesta. Myöhempiä tutkimusta varten myös tämän tutkimuksen lähdeluettelo voi olla hyödyllinen, sillä se sisältää varsin laajalti alan keskeisintä tutkimusta. Lisäksi tämän tutkimuksen rajausten vuoksi mielenkiintoisia jatkotutkimusaiheita on paljonkin. Jatkotutkimusaiheita kuvataan tarkemmin luvussa 9.3.

Tutkimuksessa on rakennettu uudenlainen malli käytettävyydestestauksen ja ketterän ohjelmistokehityksen yhteensovittamiseksi. Tämä malli voi sellaisenaan tai soveltuvin osin olla hyödynnettävissä myös muissa organisaatioissa. Myös jo pelkästään eri kevyiden käytettävyydestestausmenetelmien kuvaukset ja ohjeet niiden soveltamisesta ovat hyödyllisiä, ja mahdollisesti sellaisenaan käytettävissä muissa organisaatioissa. On perusteltua olettaa, että kuvatulla mallilla käytettävyydestestauksen ja ketterän kehityksen yhteensovittamiseen on saavutettavissa samankaltaisia tuloksia muissakin organisaatioissa kuin Samlinkilla, sillä tässä tutkimuksessa mallia testattiin useissa, keskenään varsin erilaisissa projekteissa ja niistä saadut havainnot muistuttivat paljolti toisiaan.

8.4 Tulosten reliabiliteetti ja validiteetti

Tässä alaluvussa käsitellään tutkimuksen reliabiliteettia ja validiteettia, siis sitä, ovatko havainnot toistettavissa ja miten hyvin tutkimus kokonaisuutena ja toisaalta muuttujat tutkimuksessa mittaavat sitä, mitä niiden on tarkoitus mitata. Hirsjärven ym. (2013) mukaan vaatimukset validiteetille ja reliabiliteetille eivät välttämättä tapaustutkimuksessa päde samalla tavoin kuin esimerkiksi kyselytutkimuksessa, mutta niitä pyritään kuitenkin arvioimaan.

8.4.1 Reliabiliteetti

Reliabiliteetti kuvaa mittaustuloksen toistettavuutta, eli sillä voidaan mitata tutkimustulosten ja väitteiden luotettavuutta: johtuuko tutkimustulos vain sattumasta vai kyetäänkö tulokset riippumattomasti toistamaan? Tämä voidaan myös ilmaista siten, että reliabiliteetti kuvaa sitä, kuinka todennäköisesti eri tutkijat voivat saada tutkimuksen toistamalla samoja tuloksia (Runeson & Höst, 2008). Jos mittaus on täysin reliaabeli, siihen eivät vaikuta satunnaisvirheet eivätkä olosuhteet.

Yksittäistä tapaustutkimusta sellaisenaan ei voida koskaan toistaa, sillä jokainen tilanne ja ympäristö ovat ainutkertaisia. Tämän tapaisen tapaustutkimuksen suorittamista samalla lailla toisessa ympäristössä on kuitenkin helpotettu sillä, että tutkimuksen vaiheet ja toteutustapa on dokumentoitu yksityiskohtaisesti, mikä parantaa reliabiliteettia (Hirsjärvi ym., 2013). Myös havaintojen raportoinnissa käytetty raporttimalli, käytettävyydestä käytetty kysely, tiedonkeruussa käytetty kyselytutkimus ja koehenkilöiden rekrytointiin käytetty lomake löytyvät liitteistä. Lisäksi liitteessä 3 on tutkijan omia muistiinpanoja ja havaintoja koskien käytettävyydestä järjestelyjä. Tutkimuksen tällaista toistettavuutta kutsutaan myös *ulkoiseksi reliabiliteetiksi* (LeCompte & Goetz, 1982; McLeod, 2007). Tutkimuksen *sisäisellä reliabiliteetilla* puolestaan tarkoitetaan sitä, kuinka pitkälti sama mittaustapa tuottaa keskenään samankaltaisia havaintoja tutkijasta riippumatta (LeCompte & Goetz, 1982; McLeod, 2007). Tämä ei tapaustutkimuksessa ole helposti tarkasteltavissa, joten reliabiliteetin osalta keskitytään ulkoisen reliabiliteetin varmistamiseen.

8.4.2 Validiteetti

Validiteetti ilmaisee sen, miten hyvin tutkimuksessa käytetty mittaus- tai tutkimusmenetelmä mittaa juuri sitä tutkittavan ilmiön ominaisuutta, mitä on tarkoituskin mitata, siis mittaako tutkimus sitä, mitä sen avulla on tarkoitus selvittää (Hirsjärvi ym., 2013). Kvalitatiivisessa tutkimuksessa validiteetin kannalta olennaisinta on se, onko selitys luotettava, eli ovatko ilmiön kuvaus ja siihen liitetyt selitykset ja tulkinnat yhteensopivia.

Validiteetti on jaettavissa kolmeen osaan: konstruktiio- eli rakennevaliditeettiin, sisäiseen validiteettiin ja ulkoiseen validiteettiin (Runeson & Höst, 2008; Yin, 2009). *Rakennevaliditeetilla* tarkoitetaan sitä, kuinka hyvin tutkimuksessa käytetty menetelmä mittaa sitä asiaa, mitä tutkimuksessa oli tarkoitus selvittää. Rakennevaliditeetin tarkastelu on tapaustutkimuksessa ongelmallista. Tämä johtuu siitä, että tutkimusaineiston hankinta ja tutkimuksen teko yleensäkin perustuu subjektiivisiin päätelmiin eikä yksinkertaisia mittareita yleensä ole saatavilla (Yin, 2009). Rakennevaliditeetin lisäämiseen Yin (2009) esittää kolme keinoa. Ensimmäinen näistä keinoista on tutkimusaineiston kerääminen useasta eri lähteestä, eli triangulaatio. Yksinkertaistettuna tämä tarkoittaa erilaisten menetelmien, tutkijoiden, tietolähteiden tai teorioiden yhdistämistä tutkimuksessa sen

reliabiliteetin ja validiteetin lisäämiseksi ja se voidaan tehdä aineiston, tutkijoiden, teorioiden tai menetelmien, tai joskus useiden näistä suhteen (Tuomi & Sarajarvi, 2013). Todistusketjun rakentaminen eli tutkimusvaiheiden raportointi on toinen keino. Kolmas Yinin rakennevaliditeettia lisäävä keino on tutkimusraportin luetuttaminen asiantuntijoilla.

Tässä tutkimuksessa on kehitetty malli kevyen käytettävyydestäuksen ja ketterän ohjelmistokehityksen yhteensovittamiseksi, ja sen hyvyttä on arvioitu sen aiheuttamalla kustannuksilla ja tuottamalla hyödyillä, ja pyritty vertaamaan näitä aiempiin tutkimuksiin. Kustannuksiksi laskettiin testaukseen kulunut aika sekä koehenkilöille maksetut palkkiot, tai mikäli käytettiin sisäisiä koehenkilöitä, laskettiin heidänkin käyttämälleen ajalle arvo. Työaika seurattiin Samlinkin työajanseurantajärjestelmän kautta. Työn hintana käytettiin sisäisen työn laskennallista arvoa, 50 euroa. Koska projektiryhmille haluttiin jättää vapaus käsitellä havainnot haluamallaan tavalla, eikä käsittely siten ollut mallin piirissä, projektiryhmän käyttämää aikaa ei tässä tutkimuksessa mitattu. Hyötyjä arvioitiin toisaalta tarkastelemalla käytettävyydestäuksen tuottamia havaintoja ja projektiryhmien hyväksyntäprosenttia niille, ja toisaalta suorittamalla tutkimuksen lopuksi kysely projektiryhmien jäsenille.

Tämän tutkimuksen osalta sovelletaan menetelmätriangulaatiota, sillä dataa kerätään tapaustutkimuksen aikaisen havainnoinnin lisäksi kyselytutkimuksella. Tietojen keruu perustui havainnointiin, erityisesti kustannusten osalta, vapaamuotoiseen kyselyyn käytettävyydestäuksen tuottamiin havaintoihin reagointitapojen selvittämiseksi sekä kurinalaisempaan kyselyyn tutkimuksen lopuksi sen selvittämiseksi, kuinka hyödyllisenä projektiryhmät pitivät käytettävyydestäusmallia. Vapaamuotoinen kysely projektiryhmän reaktioista suunnattiin ensisijaisesti projektipäällikölle, mutta ainakin yhdessä projektissa vastaus saatiin projektipäällikön sijasta käytettävyyssiantuntijalta. Loppupalautekyselyyn saatiin vastauksia kaikkien projektiryhmien edustajilta, ja kohtuullisen kattavasti myös projektiryhmän erityyppisiltä jäseniltä, siis sekä projektipäälliköiltä, testaajilta, kehittäjiltä että käytettävyyssiantuntijaltakin.

Loppupalautekyselyn avulla haluttiin kerätä vielä tietoa projektiryhmien tyytyväisyydestä käytettävyydestäusmalliin. Tällaisessa kyselytutkimuksessa, kuten tapaustutkimuksessa, on tutkijan aiheuttaman vääristymän (engl. *researcher bias*) riski suuri, ja se voi ilmetä ainakin tavassa, jolla kysymys kysytään, kysymysten määrässä, vastausvaihtoehtojen ja kategorioiden laatimisessa sekä kyselyn ohjeissa (Kitchenham & Pfleeger, 2002b). Vääristymä syntyy helposti vahingossa, ja siten sen ehkäiseminen vaatii oman toiminnan tarkkailua. Lisäksi tulokset raportoidaan avoimesti ja kuvataan johtopäätöksiin johtanut analyysiprosessi yksityiskohtaisesti. Kyselyn toimivuutta testattiin ennen sen lähettämistä potentiaalisille vastaajille antamalla se täytettäväksi kahdelle koehenkilölle: tutkijalle ja projektiryhmän jäsentä vastaavalle henkilölle joka ei kuitenkaan kuulu varsinaiseen otantaan. Heidän kommenttinsa perusteella kyselyä voitiin vielä hioa ennen sen lähettämistä lopullisille vastaajille.

Kerättyjä tietoja kustannuksista voidaan pitää erittäin luotettavina, sillä myös kohdeorganisaation edustajat ovat tarkastaneet tiedot ajankäytöstä. Muut

kustannukset taas on laskettu tositteiden perusteella, joten nekin ovat erittäin luotettavia. Tiedot hyödyistä on kerätty sekä projektien aikana että loppupalautekyselyn avulla projektiryhmiltä. Projektien aikana havaintojen luokittelu perustui siihen, miten projektiryhmä reagoi havaintoon, eli luodaanko sen perusteella esimerkiksi uusia kohteita työlisterille tai virheidenseurantajärjestelmään. Näitä tietoja voidaan pitää melko luotettavina, joskin ensimmäisten käytettävyydestäuskertojen osalta luokittelussa oli epätarkkuutta johtuen havaintojen raportointimallin kehityksestä tutkimuksen aikana. Loppupalautekyselyn vastauksiin täytyy suhtautua kriittisemmin. Kaikki kyselyyn vastaajat pitivät mallia hyödyllisenä tai melko hyödyllisenä. On kuitenkin huomionarvoista, että useimmat palautteet sisälsivät myös rakentavaa kritiikkiä tai parannusehdotuksia, ja havaintojen hyödyntämistä pidettiin tyypillisimmin korkeintaan melko helpo-pona. Loppupalautekyselyyn sai vastata halutessaan anonyymisti, mutta anonyymien vastausten sävy ei merkittävästi poikennut ei-anonyymien vastausten sävystä. Vastauksia voitaneen pitää kohtuullisen luotettavina.

Tutkimuksen *sisäinen validiteetti* puolestaan tarkastelee kausaliiteetin validiutta, eli sitä, aiheutuvatko havaitut muutokset muuttujien arvoissa niistä tekijöistä, joista niiden arvellaan johtuvan (Runeson & Höst, 2008). Tässä tutkimuksessa ei kuitenkaan tutkittu tällaisia kausaalisuhteita, joten sisäistä validiteettiä ei tarkastella tämän lähemmin.

Ulkoinen validiteetti kuvaa sitä, kuinka tutkimuksen tuloksia voidaan yleistää (Runeson & Höst, 2008). Tapaustutkimusten tulosten yleistämisessä on syytä noudattaa varovaisuutta. Yin (2009) ehdottaa tutkimuksen ulkoisen validiteetin parantamiseksi sen testaamista useammassa tapauksissa tai useiden kilpailevien teorioiden kokeilua samassa projektissa. Vaikka kilpailevia teorioita ei voida testata, tehdään vertailua aiempiin yhteensovitusmalleihin. Mallia on lisäksi kokeiltu käytännössä, jolloin kokeilun kohteena oli useita erilaisia projekteja, joiden toteutusteknologiat, projektiryhmien kokoonpanot ja asiakkaat olivat keskenään varsin erilaisia, joten mallin toimivuudesta saatiin kerättyä tietoa useissa erilaisissa tilanteissa. Tutkimusmenetelmä ja erityisesti kehitetyn testausmallin soveltamistapa on kuvattu niin yksityiskohtaisesti, että on syytä arvioida, että samaa menetelmää noudattaen tutkimuksen kohdeorganisaatiota vastaavassa organisaatiossa voitaisiin saada samankaltaisia tuloksia. Ainakin voidaan olettaa, että sovellettaessa mallia Samlinkin muissa projekteissa, tulokset ovat samankaltaisia. Malli on lisäksi varsin yleisellä tasolla liittymäkohdiltaan ketterään kehitystapaan, joten se lienee ainakin jossain määrin sovellettavissa useimmissa iteratiivista ohjelmistokehitystä hyödyntävissä organisaatioissa.

Runeson ja Höst (2008) pitävät tutkimuksen validiteetin kannalta tärkeänä myös sitä, että organisaatiossa vietettiin riittävästi aikaa siihen hyvin tutustumiseksi jotta myös tapaustutkimuksen tulokset voitiin ymmärtää oikein. Tässä tapauksessa organisaatiossa on vietetty riittävästi aikaa siihen tutustumiseksi varsin perusteellisesti, tutkimusraportin kirjoittamishetkellä yli kolme vuotta. Yhteenvetona voidaan todeta, että yllä esitetyistä puutteista huolimatta tutkimuksen reliabiliteettia ja validiteettia voidaan pitää kohtuullisen hyvänä, erityisesti ottaen huomioon, että kysymyksessä on tapaustutkimus.

9 YHTEENVETO

Tämän tutkimuksen tavoitteena on ollut ensiksikin tehdä katsaus aiempaan tutkimukseen tavoista sovittaa yhteen käytettävyydestaus ja ketterä kehitys. Erityisesti keskityttiin kevyeen käytettävyydestaukseen, joka sopii lähtökohdiltaan parhaiten yhteen ketterän kehityksen kanssa. Tämän kirjallisuuskatsauksen pohjalta laadittiin Nielsen+Krug-malli käytettävyydestauksen ja ketterän kehityksen yhteensovituksesta. Tutkimuksen toisessa vaiheessa testattiin tätä mallia käytännössä. Tapaustutkimuksen kohteena oli suomalainen ohjelmistoyhtiö Samlink, jossa sovelletaan ketterää kehitystä SamScrum-projektimallin mukaisesti. Tässä luvussa esitetään tapaustutkimuksen päätulokset, esitetään niiden pohjalta johtopäätöksiä sekä ehdotetaan mielenkiintoisia jatkotutkimusaiheita.

9.1 Tulokset ja johtopäätökset

Tässä tutkimuksessa kehitetty Nielsen+Krug-malli koostuu kolmesta käytettävyydestausmenetelmästä, korttien lajittelusta (Nielsen, 1995a; Rubin, 2008), heuristisesta evaluoinnista (Kane, 2003; Nielsen, 1995c; Rubin, 2008) ja yksinkertaisesta ääneen ajattelusta (Kane, 2003; Krug, 2000; Nielsen, 1994b, 1995a). Malli sovitettiin Samlinkilla hyödynnettävään SamScrum-ohjelmistokehitysmalliin, ja se määrittelee testauksen piiriin kuuluvat aktiviteetit sekä mahdollisesti tuotettavan dokumentaation.

Nielsen+Krug -mallissa keskitytään kevyisiin käytettävyydestauksen menetelmiin. Tulokset raportoidaan kattavammin kuin esimerkiksi Martyn ja Twidalen (2005) tai Krugin (2000) mallissa, sillä projektiryhmät pitivät sitä tärkeänä ja pitivät erityisesti konkreettisia korjausehdotuksia hyvinä. Yhteensovitus tehtiin XP:n sijasta scrumin kanssa, jolloin otettiin mallia mm. Northropin (2011, 2012) integrointitavasta, mutta huomattavasti konkreettisemmat ohjeet tarjoten. Testauksen järjestely käytännössä sai paljon vaikutteita RITE+Krug-mallista (McGinn & Chang, 2013), mutta malli haluttiin pitää kevyempänä ja erityisesti

sisällyttää mukaan testausmenetelmä myös tilanteeseen, jossa käytettävyydestä resursseja on käytettävissä hyvin vähän.

Mallia kokeiltiin käytännössä neljässä projektissa. Tavoitteena oli saada yksityiskohtaisempaa tietoa mallin hyödyntämisestä käytännössä, sekä myös kehittää joitakin mallin piirteitä, kuten testauksen havaintojen raportointikäytänteitä, paremmin toimiviksi. Tämän lisäksi haluttiin kerätä tietoa mallin hyödyntämisestä koituvista *kustannuksista*, erityisesti mallin hyödyntämiseen kuluva ajasta, sekä tietysti mallin hyödyntämisestä koituvista *hyödyistä*. Kustannuksiin laskettiin testaukseen kuluva aika ja suorat kustannukset, joita testauksesta aiheutui. Hyötyjä mitattiin käytettävyydestä tuottamien havaintojen ja niiden hyväksyntäprosentin mukaan, sekä projektiryhmien jäsenille suunnatulla loppupalautekyselyllä. Verrattuna muihin kirjallisuuskatsauksissa käsiteltyihin malleihin Nielsen+Krug-malli osoittautui huomattavan kevyeksi ja edulliseksi, ja projektiryhmien jäsenet olivat siihen pääsääntöisesti tyytyväisiä.

Tutkimuksessa havaittiin, että käytettävyydestä saadaan verrattain pienilläkin resursseilla huomattava määrä havaintoja, joiden prioriteetti vaihtelee kriittisestä kosmeettiseen. Varsinaisten käytettävyyteen liittyvien havaintojen lisäksi saatiin myös merkittävä määrä havaintoja virheistä. Merkittävin ero aiempaan tutkimukseen kevyistä käytettävyydestä menetelmistä havaittiin ajankäytössä. Tässä tutkimuksessa huomattavan pieni osa ajasta kului valmisteluun, suurin osa ajasta kului varsinaiseen testaukseen tai välittömiin testauksen valmisteluihin, ja melkein yhtä paljon aikaa kului tulosten analysointiin ja raportointiin. Pisimmillään aikaa kului yhteensä nelisen tuntia koehenkilöä kohden, nopeimmillaan selvittiin reilulla kahdella tunnilla. Havaintokohtaisen ajan mittaamisessa havaittiin, että heuristisella evaluoinnilla uusia havaintoja saatiin noin viidessä minuutissa havaintoa kohden, kun taas koehenkilöitä hyödyntävässä käytettävyydestä menetelmässä aikaa kului noin 17 minuutista yli 40 minuuttiin projektista riippuen, keskiarvon ollessa noin 27 minuuttia.

Verrattuna aiempien yhteensovitusmallien kustannuksiin, tässä tutkimuksessa testaus saatiin toteutettua varsin edullisesti. Koehenkilökohtainen kustannus (yksinkertaista ääneen ajattelua ja osin myös korttien lajittelua hyödyntävien projektien osalta) oli kalleimmassa projektissa noin 200 euroa, kun aiemmissa tutkimuksissa mm. Nielsenin ja Landauerin (1993) mukaan koehenkilökohtainen kustannus kevyitä menetelmiä hyödyntäessään voi olla edullisimmillaan noin 490 euroa²⁷. Tässä tutkimuksessa hinta oli siis korkeimmillaankin vain noin puolet tästä. Käytettävyydestä kokonaisen projektin osalta maksoi kalleimmillaan noin 2 640 euroa, johon sisältyi kolme käytettävyydestä menetelmää, 13 koehenkilön hyödyntäminen ja menetelminä sekä yksinkertaistetun ääneen ajattelun että korttien lajittelun soveltamisen. Krugin mukaan (2000) hänen kuvaamaansa kevyen käytettävyydestä menetelmän käyttöön voi kuluja esimerkiksi 3 920 euroa koko projektin aikana, mikä ei sisällä mitään hintaa kuluneelle työajalle, joten tämäkin kustannus saatiin alitettua.

Havaintojen voidaan katsoa tukevan aiempien tutkimusten johtopäätöstä siitä, että *on todennäköisesti parempi tehdä käytettävyydestä menetelmää kuin olla tekemättä*

²⁷ Nykyarvoon muunnettu arvo vuoden 1993 dollareista.

(Dicks, 2002; Rubin, 2008), koska projektiryhmät olivat tyytyväisiä käytettävyydestä tuloksiin. Lisäksi Nielsenin (1995a) ja Krugin (2006, 2010) tapa perustella kevyttä käytettävyydestäusta, *on parempi testata vähän kuin ei ollenkaan* saa myös tukea projektiryhmien palautteesta. Siitä, minkä verran käytettävyydestäusta lopulta vaikutti projekteissa kehitettävien järjestelmien laatuun, ei voida saada kovin tarkkaa tietoa, sillä kontrolliryhmää ei ollut käytettävissä eikä järjestelmille voitu tehdä esimerkiksi ulkopuolisen asiantuntijan arviota ennen ja jälkeen käytettävyydestäuksen havaintoihin reagoinnin sen selvittämiseksi, paransiko havaintoihin reagoiminen todella järjestelmän laatua. Toisaalta projektissa, jossa testausta saatiin tehtyä pisimpään (3 sprintin ajan), järjestelmän laatu parani sekä koehenkilöiltä kerätyn palautteen että uusien käytettävyysongelmien määrän vähentymisen perusteella.

Tutkimuksen perusteella käytettävyydestäusta voidaan suositella tehtäväksi sellaisissa sovelluskehitysprojekteissa, joissa käytettävyydellä ylipäänsä on merkitystä. Tähän tutkimukseen oli valittu mukaan vain sellaisia projekteja, joissa käyttäjien tyytyväisyydellä on tosiasiallisesti merkitystä, mutta silti voitiin havaita, että kaikista pienin havaintojen hyväksyntäprosentti oli projektilla, joka toteutti sisäiseen käyttöön tulevaa järjestelmää. Käytettävyyttä ei sisäiseen käyttöön tulevassa järjestelmässä pidetä ehkä yhtä suurena arvossa kuin esimerkiksi kuluttajien käyttöön tulevan järjestelmän kohdalla. Tämä on toisaalta perusteltuakin – sisäisiä käyttäjiä voidaan kouluttaa ja opastaa, siinä missä kuluttajakäyttäjä saattaa esimerkiksi pettyä järjestelmään ja vaihtaa kilpailijan tuotteeseen.

9.2 Rajoitteet

Tutkimuksen käytännön vaiheen tulokset on saatu yhdestä organisaatiosta ja rajallisesta määrästä projekteja ja testauskertoja, joten tulosten yleistettävyys voidaan kyseenalaistaa. Tarkastelun kohteena olevan organisaation ominaispiirteet vaikuttivat tutkimuksen tuloksiin. Tutkimuksessa testattiin eri projekteissa eri järjestelmien hyvinkin erilaisia osia ja rekrytoitiin parhaiten soveltuvia koehenkilöitä, mutta valtaosa koehenkilöistä oli kuitenkin hyvin väljillä kriteereillä kuluttajakäyttäjän edustajaksi valittuja. Testattavat projektit olivat lisäksi pienehköjä.

Tapaustutkimuksessa mukana olleista projekteista ei saatu liiketoiminnan vaatimusten vuoksi kerättyä palautetta aina samalla tavalla, eikä myöskään kyselytutkimusta saatu tehtyä koko toivotulle otannalle muun muassa henkilöstömuutosten vuoksi, joten on mahdollista, että otos ei ole niin kuvaava kuin voitaisiin toivoa. Lisäksi yksi tutkittavan organisaation vaatimuksista kehitettävälle käytettävyydestäusmallille oli tulosten nopean raportoinnin mahdollistaminen, ja tätä varten tutkimuksen aikana kehitettiin iteratiivisesti raportointilomake tarkoitusta varten. Tämän seurauksena havaintojen raportointitapa, ja myös niihin saatu palaute projektipäälliköiltä, kehittyi tutkimuksen aikana. Tämä hiukan heikentää tulosten vertailukelpoisuutta eri iteraatioiden välillä ja vaikeuttaa esimer-

kiksi tulosten hyväksynnän arvioimista eksaktisti. Lisäksi raportoidessaan havaintoihin reagoivia projektipäälliköt jättivät joitakin havaintoja käsittelemättä ilmoittamatta syytä.

Yksi merkittävä rajoite liittyy käytettävyydestä kusten kustannuksien arviointiin. Tällaisessa tutkimuksessa mielekkäintä olisi laskea, paljonko käytettävyydestä voidaan säästää myöhempiä kuluja, mutta tämän arviointi on erittäin työlästä (Nielsen & Landauer, 1993). Kun tarkastellaan testauksesta aiheutuneita kustannuksia, tulee helposti se kuva, että mitä matalampi havaintokohtainen kustannus on, sitä paremmin testaus on onnistunut. Näin ei kuitenkaan välttämättä ole, sillä projektissa aikaisessa vaiheessa (selvästi keskeneräisen) järjestelmän testaaminen tuottaa todennäköisesti vähemmän havaintoja kuin valmiin ja erittäin hankalakäyttöisen järjestelmän testaaminen. Jälkimmäinen tuottaa todennäköisesti huomattavasti halvemmalla havaintoja, mutta havaintoihin reagoiminen voi tässä vaiheessa olla jo kallista. Samoin pieni havaintojen määrä myöhäisessä testausiteraatiassa voi myös validoida aiempia korjauksia, kuten tässä tutkimuksessa todennäköisesti kävi PA-projektin sprintin 3 kohdalla.

Tutkijan aiempi kokemus, niin opintojen kuin työelämänkin osalta, on hyvin pitkälti ohjelmistokehityksen alalta ja fokus on ollut teknisessä toteutuksessa tai liiketoiminta-asioissa testauksen ja käytettävyyden sijasta. Motiivi tämän tutkimuksen tekoon on syntynyt käytännön kokemuksista, mutta ammattitaito käytettävyyden arviointiin tai testaamiseen on pääosin kehittynyt tutkimuksen aikana. Siksi tulosten muodostamisessa ja arvioinnissa ei ole ollut tukena laajaa ammattitaitoa ja kokemusta, joka olisi epäilemättä auttanut tulosten objektiivisuuden ja täydellisyyden varmistamisessa.

9.3 Jatkotutkimusaiheet

Yksi tutkimuksen aikana eniten työtä aiheuttanut tehtävä oli sopivan tavan löytäminen käytettävyydestä havaintojen raportoimiseksi. Kysymys siitä, sopiiko nykyinen, kokemusten perusteella hiottu raportointilomake ja -tapa kaikille projekteille, tai onko kaikille projekteille sopivaa raportointitapaa yleensä ottaen edes olemassa, jäi avoimeksi ja vaatii jatkotutkimusta. Tutkimuksessa ei myöskään voitu vertailla erilaisten koehenkilöotantojen tuottamia tuloksia ja sitä, paljonko niissä on eroa. Varsin suuri ja liiketoiminnallisesta näkökulmasta kiinnostava jatkotutkimus liittyy käytettävyydestä myymiseen asiakkaille: asiakkaat suhtautuvat työmääräarvioihin varsin kriittisesti joka tapauksessa, joten sen selvittäminen, millä tavalla asiakas saadaan alusta asti hyväksymään myös käytettävyydestä osana projektia ja sen kuluja, olisi hyödyllistä.

Tutkimusprojektin aikana myös Samlinkilla heräsi kiinnostusta käytettävyydestä tuotteistamiseen sekä kysymys siitä, missä määrin testaus olisi jatkossa syytä tehdä itse ja missä määrin hankkia ulkopuolelta. Nämä kysymykset tarjoavat työhön vaikuttavien seikkojen lisäksi mielenkiintoisia jatkotutkimusaiheita ja -ongelmia. Millä tavoin käytettävyydestä, tai käytettävyy-

suunnittelua yleensä, on voitu tuotteistaa ja minkälaisia liiketoimintamahdollisuuksia käytettävyyden hyväksi tehtävä työ tarjoaa? Minkälaisella menestyksellä käytettävyyssuunnittelu on huomioitu organisaation sisäisenä tai ulkoisena aktiviteettina?

Nielsen+Krug-mallin jatkokehitys kaiken kaikkiaan tarjoaa mielenkiintoisia aiheita jatkotutkimukselle. Käytettävyydestä loppukyselyyn annettujen projektiryhmien edustajien vastausten perusteella voidaan päätellä, että mallissa olisi vielä kehittämistä. Osa projektiryhmien edustajista olisi halunnut päästä mukaan suunnittelemaan käytettävyydestä tapauksia. Tämä johtaisi suureen kompromissiin keveyden suhteen, sillä suuri osa mallin keveydestä pohjautui käytettävyydestä suunnitelman joustavaan ja opportunistiseen laatumiseen olemassa olevien materiaalien, kuten käyttötapauksien, skenaarioiden tai testaus-suunnitelman pohjalta, ja perusteellisempi käytettävyydestä suunnitelman toteuttaminen aiheuttaisi mallin muuttumisen huomattavasti raskaammaksi. Osa kaipasi myös suullista havaintojen ”purkutilaisuutta” käytettävyydestä ja projektiryhmän kesken tai muita parannuksia havaintojen raportointitapaan, kuten enemmän kuvakaappauksia. Käytettävyydentutkija esitti myös toiveen päästä seuraamaan testausseksiota, mutta tätä ei toisaalta käytännön syistä (kuten aikataulun vuoksi) ja toisaalta koehenkilön suojaamiseksi häiriötekijöiltä toteutettu tämän tutkimuksen puitteissa. Kahta jälkimmäistä puutetta voitaisiin luultavasti helpottaa jonkin verran mahdollistamalla käytettävyydestä seuranta reaaliaikaisesti organisaation sisäverkossa sijaitsevan palvelimen kautta. Näiden parannusten tai muutosten huomiointi mallissa ja käytettävyydestä uusissa, tämän tutkimuksen projekteja riittävässä määrin vastaavissa projekteissa tarjoaisi mahdollisuuden verrata käytettävyydestä kustannuksia ja hyötyjä hiukan erilaisten mallien välillä.

Tässä tutkimuksessa jäi myös avoimeksi se, kuinka paljon käytettävyydestä tosiasiallisesti paransi järjestelmän käytettävyyttä. Kahden samanlaisen projektin toteuttaminen siten, että toisessa olisi mukana käytettävyydestä ja toisessa ei, ei liene mielekkäällä tavalla mahdollista, mutta parempaa osviittaa käytettävyydestä vaikutuksista saataisiin kenties ottamalla tarkasteltaviksi pidempiä projekteja, joissa käytettävyydestäkin (esimerkiksi Nielsen+Krug-mallin mukaisesti) voitaisiin tehdä esimerkiksi kymmenen sprintin aikana. Tämä ei valitettavasti tässä tutkimuksessa ollut mahdollista, mutta tarjoaa mielenkiintoisen jatkotutkimusaiheen.

LÄHTEET

- Abdelnour-Nocera, J., & Sharp, H. (2008). Adopting Agile in a Large Organisation. Teoksessa P. Abrahamsson, R. Baskerville, K. Conboy, B. Fitzgerald, L. Morgan, & X. Wang (Toim.), *Agile Processes in Software Engineering and Extreme Programming* (s. 42–52). Springer Berlin Heidelberg.
- Aigner, W. (2002, 5. helmikuuta). Combining Extreme Programming & Interaction Design.
- Aktia Oyj. (2014). Valuuttamuunnin - Aktia. Haettu 4.6.2014 osoitteesta <http://www.aktia.fi/fi/valuuttamuunnin>
- Anderson, D. J. (2010). *Kanban: successful evolutionary change in your software business*. Sequim, Wash.: Blue Hole Press.
- Apache Software Foundation. (2013). Apache JMeter - Apache JMeter™. jmeter.apache.org. Haettu 13.4.2014 osoitteesta <https://jmeter.apache.org/>
- Assistant Secretary for Public Affairs. (2013a, 6. kesäkuuta). Glossary | Usability.gov. Haettu 6.1.2014 osoitteesta <http://www.usability.gov/what-and-why/glossary/u/index.html>
- Assistant Secretary for Public Affairs. (2013b, 10. syyskuuta). Reporting Usability Test Results. Haettu 1.1.2014 osoitteesta <http://www.usability.gov/how-to-and-tools/methods/reporting-usability-test-results.html>
- Bailey, R. W., Allan, R. W., & Raiello, P. (1992). Usability testing vs. heuristic evaluation: A head-to-head comparison. Teoksessa *Proceedings of the Human Factors and Ergonomics Society Annual Meeting* (Vsk. 36, s. 409–413). SAGE Publications.
- Bankston, A. (2003). *Usability and User Interface Design in XP*. Fairfax, VA: CC Pace Systems.
- Barksdale, J. T., & McCrickard, D. S. (2012). Software product innovation in agile usability teams: an analytical framework of social capital, network governance, and usability knowledge management. *International Journal of Agile and Extreme Software Development*, 1(1), 52.
- Beck, K. (1999). Embracing change with extreme programming. *Computer*, 32(10), 70–77.
- Beck, K., & Andres, C. (2005). *Extreme programming explained: embrace change* (2. painos). Boston, MA: Addison-Wesley.
- Beck, K., Beedle, M., Bennekum, A. van, Cockburn, A., Cunningham, W., Fowler, M., ... Thomas, D. (2001). Ketterän ohjelmistokehityksen julistus. Haettu 22.6.2013 osoitteesta <http://agilemanifesto.org/iso/fi/>
- Benington, H. D. (1956). Symposium on advanced programming methods for digital computers. Teoksessa *Symposium on Advanced Programming Methods for Digital Computers: Washington, D.C., June 28, 29, 1956*. Washington, D.C.: Office of Naval Research, Dept. of the Navy.
- Benington, H. D. (1983). Production of Large Computer Programs. *IEEE Annals of the History of Computing*, 5(4), 350–361.

- Bertolino, A. (2001). Software testing. *SWEBOK*, 69.
- Bevan, N., Blom, J., Blythe, M., Buie, E., Christou, G., Cockton, G., ... Wright, P. (2011). User experience white paper: Bringing clarity to the concept of user experience. Teoksessa V. Roto, E. Law, A. Vermeeren, & J. Hoonhout (Toim.), *Result from Dagstuhl Seminar on Demarcating User Experience, September 15-18, 2010*. Dagstuhl seminar on Demarcating User Experience.
- Bias, R. G., & Mayhew, D. J. (Toim.). (2005). *Cost-justifying usability: an update for an Internet age* (2. painos). Amsterdam; Boston: Morgan Kaufman.
- Bird, M. S. (2010). *Utilizing agile software development as an effective and efficient process to reduce development time and maintain quality software delivery*.
- Blomquist, Å., & Arvola, M. (2002). Personas in Action: Ethnography in an Interaction Design Team. Teoksessa *Proceedings of the Second Nordic Conference on Human-computer Interaction* (s. 197–200). New York, NY, USA: ACM.
- Boehm, B. W. (1988). A spiral model of software development and enhancement. *Computer*, 21(5), 61–72.
- Brooke, J. (1996). SUS - A quick and dirty usability scale. Teoksessa *Usability evaluation in industry* (s. 189–195). London; Bristol, Pa: Taylor & Francis.
- Bruun, A., Gull, P., Hofmeister, L., & Stage, J. (2009). Let your users do the testing: a comparison of three remote asynchronous usability testing methods (s. 1619–1628). ACM Press.
- Budwig, M., Jeong, S., & Kelkar, K. (2009). When user experience met agile: a case study. Teoksessa *CHI '09 Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (s. 1619–1628). New York, NY, USA: ACM Press.
- Butler, K. A. (1996). Usability engineering turns 10. *Interactions*, 3(1), 58–75.
- Bygstad, B., Ghinea, G., & Brevik, E. (2008). Software development methods and usability: Perspectives from a survey in the software industry in Norway. *Interacting with Computers*, 20(3), 375–385.
- Chamberlain, S., Sharp, H., & Maiden, N. (2006). Towards a Framework for Integrating Agile Development and User-Centred Design. Teoksessa P. Abrahamsson, M. Marchesi, & G. Succi (Toim.), *Extreme Programming and Agile Processes in Software Engineering* (Vsk. 4044, s. 143–153). Berlin, Heidelberg: Springer Berlin Heidelberg.
- Chan, F. K. Y., & Thong, J. Y. L. (2009). Acceptance of agile methodologies: A critical review and conceptual framework. *Decision Support Systems*, 46(4), 803–814.
- Chow, T., & Cao, D.-B. (2008). A survey study of critical success factors in agile software projects. *Journal of Systems and Software*, 81(6), 961–971.
- Cohen, C. F., Birkin, S. J., Garfield, M. J., & Webb, H. W. (2004). Managing conflict in software testing. *Communications of the ACM*, 47(1), 76–81.
- Cohen, D., Lindvall, M., & Costa, P. (2004). An Introduction to Agile Methods. Teoksessa *Advances in Computers* (Vsk. 62, s. 1–66). Elsevier.

- CoinNews. (2014, 15. toukokuuta). Inflation Calculator | Find US Dollar's Value from 1913-2014. Haettu 20.5.2014 osoitteesta <http://www.usinflationcalculator.com/>
- Commonwealth Association for Public Administration and Management. (2010). Overview of Case Study Models and Methodology. A Shared Library of Public Administration Case Studies (CAPAM/Commonwealth Secretariat Project).
- Constantine, L. L. (2001, 4. joulukuuta). forUse Newsletter #12. Uutiskirje.
- Constantine, L. L., & Lockwood, L. A. D. (2003). Usage-centered software engineering: an agile approach to integrating users, user interfaces, and usability into software engineering practice. Teoksessa *Proceedings 25th International Conference on Software Engineering: ICSE 2003, 3-10 May 2003, Portland, Oregon USA*. (s. 746-747). International Conference on Software Engineering: ICSE 2003., Portland, Oregon USA.: IEEE.
- Cooper, A. (1995). *About face: the essentials of user interface design*. Foster City, California: IDG Books Worldwide.
- Cooper, A. (2004). *The inmates are running the asylum*. Indianapolis, IN: Sams.
- Crispin, L., & Gregory, J. (2009). *Agile testing: a practical guide for testers and agile teams*. Upper Saddle River, NJ: Addison-Wesley.
- Davis, F. D., & Venkatesh, V. (2004). Toward preprototype user acceptance testing of new information systems: implications for software project management. *IEEE Transactions on Engineering Management*, 51(1), 31-46.
- De Kock, E., van Biljon, J., & Pretorius, M. (2009). Usability evaluation methods: mind the gaps (s. 122-131). ACM Press.
- Department of Defence. DOD-STD-2167, Military standard: Defense system software development, DOD-STD-2167 (1985).
- Department of Defence. DOD-STD-2167A, Military standard: Defense system software development, DOD-STD-2167A (1988).
- Department of Defence. MIL-STD-882D, System Safety Program Requirements (2000).
- Desurvire, H. W. (1994). Faster, cheaper!! Are usability inspection methods as effective as empirical testing? Teoksessa *Usability inspection methods* (s. 173-202).
- Dicks, R. S. (2002). Mis-usability: on the uses and misuses of usability testing. Teoksessa *SIGDOC '02 Proceedings of the 20th annual international conference on Computer documentation* (s. 26-30). SIGDOC '02, New York, NY, USA: ACM Press.
- Donahue, G. M. (2001). Usability and the bottom line. *IEEE Software*, 18(1), 31-37.
- Dray, S. (2008). Can't use equals Defunct » UX Quotes- Useful Quotes on User Experience. Haettu 2.8.2013 osoitteesta <http://www.uxquotes.com/author/susan-dray/cant-use-defunct/>
- Dumas, J. S. (1999). *A practical guide to usability testing* (Uudistettu painos.). Exeter, England; Portland, Or: Intellect Books.
- Dybå, T., & Dingsøy, T. (2008). Empirical studies of agile software development: A systematic review. *Information and Software Technology*, 50(9-10), 833-859.

- EPiServer World. (2009, 3. joulukuuta). Project model. *EPiServer World - Community*. Haettu 12.1.2014 osoitteesta <http://world.episerver.com/Get-Started/Complex-EPiServer-CMS-Projects/Project-model/>
- Faulkner, L. (2003). Beyond the five-user assumption: Benefits of increased sample sizes in usability testing. *Behavior Research Methods, Instruments, & Computers*, 35(3), 379–383.
- Federoff, M., Villamor, C., Miller, L., Patton, J., Rosenstein, A., Baxter, K., & Kelkar, K. (2008). Extreme Usability: Adapting Research Approaches for Agile Development. Teoksessa *CHI '08 Extended Abstracts on Human Factors in Computing Systems* (s. 2269–2272). New York, NY, USA: ACM.
- Ferré, X., Juristo, N., Windl, H., & Constantine, L. (2001). Usability Basics for Software Developers. *IEEE Softw.*, 18(1), 22–29.
- Ferreira, J., Noble, J., & Biddle, R. (2007). Agile Development Iterations and UI Design (s. 50–58). IEEE.
- Foraker Labs. (2010). Usability First - Methods - Card Sorting | Usability First. *Usability First*. Haettu 22.9.2013 osoitteesta <http://www.usabilityfirst.com/usability-methods/card-sorting/>
- Foraker Labs. (2011). Usability First - Usability Glossary - Usability Testing | Usability First. *Usability First*. Haettu 23.3.2014 osoitteesta <http://www.usabilityfirst.com/glossary/usability-testing/>
- Fowler, M., & Highsmith, J. (2001). The agile manifesto. *Software Development*, 9(8), 28–35.
- Frøkjær, E., & Larusdottir, M. K. (1999). Prediction of usability: Comparing method combinations. *Managing Information Technology Resources in Organizations in the Next Millennium*, Hershey, USA.
- Genov, A. (2005). Iterative usability testing as continuous feedback: A control systems perspective. *Journal of Usability Studies*, 2005(1), 18–27.
- Ghanam, Y., & Maurer, F. (2007). *Discount Usability Testing*.
- Gilb, T. (1981). Evolutionary development. *ACM SIGSOFT Software Engineering Notes*, 6(2), 17–17.
- Gilb, T. (1985). Evolutionary Delivery versus the "waterfall model". *ACM SIGSOFT Software Engineering Notes*, 10(3), 49–61.
- Gould, J. D. (1995). How to design usable systems. Teoksessa R. M. Baecker (Toim.), *Readings in human-computer interaction: toward the year 2000* (2. painos, s. 93–121). San Francisco, Calif: Morgan Kaufmann Publishers.
- Gray, W., & Salzman, M. (1998). Damaged Merchandise? A Review of Experiments That Compare Usability Evaluation Methods. *Human-Computer Interaction*, 13(3), 203–261.
- Greenberg, S., & Buxton, B. (2008). Usability Evaluation Considered Harmful (Some of the Time). Teoksessa *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (s. 111–120). New York, NY, USA: ACM.

- Göransson, B., Gulliksen, J., & Boivie, I. (2003). The usability design process - integrating user-centered systems design in the software development process. *Software Process: Improvement and Practice*, 8(2), 111-131.
- Göransson, B., Lif, M., & Gulliksen, J. (2003). Usability Design-Extending Rational Unified Process with a New Discipline. Teoksessa J. A. Jorge, N. Jardim Nunes, & J. Falcão e Cunha (Toim.), *Interactive Systems. Design, Specification, and Verification* (Vsk. 2844, s. 316-330). Berlin, Heidelberg: Springer Berlin Heidelberg.
- Haikara, J. (2007). Usability in Agile Software Development: Extending the Interaction Design Process with Personas Approach. Teoksessa G. Concas, E. Damiani, M. Scotto, & G. Succi (Toim.), *Agile Processes in Software Engineering and Extreme Programming* (Vsk. 4536, s. 153-156). Berlin, Heidelberg: Springer Berlin Heidelberg.
- Harper, S. (2011, 20. syyskuuta). UX the Ghost [#ux]. *Bugs Become Features...*
- Hartson, H. R., Andre, T. S., & Williges, R. C. (2001). Criteria For Evaluating Usability Evaluation Methods. *International Journal of Human-Computer Interaction*, 13(4), 373-410.
- Hertzum, M., & Jacobsen, N. E. (2001). The Evaluator Effect: A Chilling Fact About Usability Evaluation Methods. *International Journal of Human-Computer Interaction*, 13(4), 421-443.
- Highsmith, J. A. (2002). *Agile software development ecosystems*. Boston: Addison-Wesley.
- Hirsjärvi, S., Remes, P., Sajavaara, P., & Sinivuori, E. (2013). *Tutki ja kirjoita* (15.-17. painos). Helsinki: Tammi.
- Hoda, R., Noble, J., & Marshall, S. (2009). Negotiating Contracts for Agile Projects: A Practical Perspective. Teoksessa P. Abrahamsson, M. Marchesi, & F. Maurer (Toim.), *Agile Processes in Software Engineering and Extreme Programming* (Vsk. 31, s. 186-191). Berlin, Heidelberg: Springer Berlin Heidelberg.
- Høegh, R. T., Nielsen, C. M., Overgaard, M., Pedersen, M. B., & Stage, J. (2006). The Impact of Usability Reports and User Test Observations on Developers' Understanding of Usability Data: An Exploratory Study. *International Journal of Human - Computer Interaction*, 21(2), 173-196.
- Holzinger, A. (2005). Usability Engineering Methods for Software Developers. *Communications of the ACM - Interaction design and children*, 48(1), 71-74.
- Holzinger, A., Errath, M., Searle, G., Thurnher, B., & Slany, W. (2005). From extreme programming and usability engineering to extreme usability in software engineering education (XP+UE -> XU). Teoksessa *Computer Software and Applications Conference, 2005. COMPSAC 2005. 29th Annual International* (Vsk. 2, s. 169-172 Vol. 1). Computer Software and Applications Conference, 2005. COMPSAC 2005. 29th Annual International.
- Hornbæk, K. (2008). Usability Evaluation as Idea Generation. Teoksessa E. L.-C. Law, E. T. Hvannberg, & G. Cockton (Toim.), *Maturing Usability* (s. 267-286). London: Springer London.

- International Organization for Standardization. (1998). *ISO 9241-11: Ergonomic Requirements for Office Work with Visual Display Terminals (VDTs): Part 11: Guidance on Usability*.
- International Organization for Standardization. (2009). ISO DIS 9241-210: 2010. Ergonomics of human system interaction-Part 210: Human-centred design for interactive systems. *International Standardization Organization (ISO)*. Switzerland.
- Itkonen, J., Rautiainen, K., & Lassenius, C. (2005). Towards understanding quality assurance in agile software development. Teoksessa *ICAM 2005*.
- Jacobson, I., Booch, G., & Rumbaugh, J. (1999). *The unified software development process*. Reading, Mass: Addison-Wesley.
- Jeffries, R., & Desurvire, H. (1992). Usability testing vs. heuristic evaluation: was there a contest? *ACM SIGCHI Bulletin*, 24(4), 39–41.
- Jeffries, R., Miller, J. R., Wharton, C., & Uyeda, K. (1991). User interface evaluation in the real world: a comparison of four techniques (s. 119–124). ACM Press.
- Jokela, T. (2010). *Navigoi oikein käytettävyyden vesillä*. Väylä-Yhtiöt.
- Jones, C. P. (2003). Lessons learned from discount usability engineering for the US federal government. *Technical Communication*, 50(2), 232–246.
- Jørgensen, A. H. (1989). Using the thinking-aloud method in system development. Teoksessa S., Michael J International Conference on Human-Computer Interaction & G. Salvendy (Toim.), *Proceedings of the Third International Conference on Human-Computer Interaction, Boston, Massachusetts, September 18-22, 1989*. Amsterdam ; New York: Elsevier.
- Kallio, T. (2005). Usability testing of mobile applications: A comparison between laboratory and field testing. *Journal of Usability Studies*, 1(4-16), 23–28.
- Kane, D. (2003). Finding a place for discount usability engineering in agile development: throwing down the gauntlet. Teoksessa *Proceedings of the Agile Development Conference, 2003*. (s. 40–46). Agile Development Conference, 2003., IEEE.
- Kessner, M., Wood, J., Dillon, R. F., & West, R. L. (2001). On the reliability of usability testing. Teoksessa *Proceedings of CHI EA '01* (s. 97). CHI '01 Human Factors in Computing Systems, ACM Press.
- Khan, B. H. (1997). *Web-based Instruction* (1. painos). Educational Technology.
- Kılıç Delice, E., & Güngör, Z. (2009). The usability analysis with heuristic evaluation and analytic hierarchy process. *International Journal of Industrial Ergonomics*, 39(6), 934–939.
- Kitchenham, B., & Pfleeger, S. L. (2002a). Principles of survey research part 2: designing a survey. *ACM SIGSOFT Software Engineering Notes*, 27(1), 18–20.
- Kitchenham, B., & Pfleeger, S. L. (2002b). Principles of survey research part 4: questionnaire evaluation. *ACM SIGSOFT Software Engineering Notes*, 27(3), 20.
- Kniskern, K. (2013, 27. helmikuuta). Windows Phone 7.8 delayed: from beta tester to guinea pig? *LiveSide.net*. Blogi.

- Krug, S. (2000). *Don't make me think!: a common sense approach to Web usability* (1. painos). Indianapolis, Indiana: Que.
- Krug, S. (2006). *Don't make me think!: a common sense approach to Web usability* (2. painos). Berkeley, California: New Riders.
- Krug, S. (2010). *Rocket surgery made easy: the do-it-yourself guide to finding and fixing usability problems* (1. painos). Berkeley, California: New Riders.
- Laanti, M., Similä, J., & Abrahamsson, P. (2013). Definitions of Agile Software Development and Agility. Teoksessa F. McCaffery, R. V. O'Connor, & R. Messnarz (Toim.), *Systems, Software and Services Process Improvement* (Vsk. 364, s. 247–258). Berlin, Heidelberg: Springer Berlin Heidelberg.
- Ladas, C. (2009). *Scrumban - Essays on Kanban Systems for Lean Software Development*. Saetle, WA: Modus Cooperandi Press.
- Larman, C., & Basili, V. R. (2003). Iterative and incremental developments. a brief history. *Computer*, 36(6), 47–56.
- Law, E. L.-C., Roto, V., Hassenzahl, M., Vermeeren, A. P. O. S., & Kort, J. (2009). Understanding, scoping and defining user experience: a survey approach. Teoksessa *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (s. 719). SIGCHI Conference on Human Factors in Computing Systems, Boston, MA: ACM Press.
- Layman, L., Williams, L., & Cunningham, L. (2006). Motivations and measurements in an agile case study. *Journal of Systems Architecture*, 52(11), 654–667.
- LeCompte, M. D., & Goetz, J. P. (1982). Problems of Reliability and Validity in Ethnographic Research. *Review of Educational Research*, 52(1), 31–60.
- Lee, J. C. (2006). Embracing agile development of usable software systems. Teoksessa *CHI '06 Extended Abstracts on Human Factors in Computing* (s. 1767). Human Factors in Computing, Montréal, Québec, Canada: ACM Press.
- Lee, J. C., & McCrickard, D. S. (2007). Towards extreme (ly) usable software: Exploring tensions between usability and agile software development. Teoksessa *Agile Conference (AGILE), 2007* (s. 59–71).
- Lee, J. C., McCrickard, D. S., & Stevens, K. T. (2009). Examining the Foundations of Agile Usability with eXtreme Scenario-Based Design. Teoksessa *AGILE'09* (s. 3–10). 2009 Agile Conference, IEEE.
- Lekman, L., Eskelinen, A., Heiramo, P., Heljo, S., Lindström, J., Nyman, R., ... Ziegler, L. (2012, 3. helmikuuta). Uusi suomenkielinen Scrum Guide. *larelekman.com*. Haettu 22.4.2013 osoitteesta <http://larelekman.com/scrumguide/>
- Lindgaard, G., & Chattratichart, J. (2007). Usability testing: what have we overlooked? Teoksessa *Proceedings of the SIGCHI Conference on Human Factors in Computing* (s. 1415). Conference on Human Factors in Computing, San Jose, California, USA: ACM Press.
- Litmanen, S. (2013a). SamScrum. Teoksessa *Samcom wiki*.
- Litmanen, S. (2013b, 4. maaliskuuta). *SamScrum - Ketterää kehitystä Jyväskylässä*. Jyväskylä.

- Martin, D., Rooksby, J., Rouncefield, M., & Sommerville, I. (2007). "Good" Organisational Reasons for "Bad" Software Testing: An Ethnographic Study of Testing in a Small Software Company. *Teoksessa Proceedings of the 29th international conference on Software Engineering* (s. 602–611). The 29th international conference on Software Engineering, Minneapolis, MN: IEEE.
- Marty, P. F., & Twidale, M. B. (2005). Extreme discount usability engineering.
- McGinn, J. J., & Chang, A. R. (2013). RITE+ Krug: A Combination of Usability Test Methods for Agile Design. *Journal of Usability Studies*, 8(3), 61–68.
- McInerney, P., & Maurer, F. (2005). UCD in agile projects: dream team or odd couple? *interactions*, 12(6), 19–23.
- McLeod, S. A. (2007). What is Reliability? | Simply Psychology. *Simply Psychology*. Haettu 8.6.2014 osoitteesta <http://www.simplypsychology.org/reliability.html>
- Meszaros, G., & Aston, J. (2006). Adding Usability Testing to an Agile Project. *Teoksessa Agile Conference, 2006* (s. 289–294). IEEE.
- Molich, R., & Nielsen, J. (1990). Improving a human-computer dialogue. *Communications of the ACM*, 33(3), 338–348.
- Negru, S. (2011). Laboratory 11 - Human-Computer Interaction Laboratory. *www.uaic.ro*. Haettu 23.3.2014 osoitteesta <http://profs.info.uaic.ro/~stefan.negru/hci2011/lab11.html>
- Nell, T. (2009, 17. toukokuuta). 6 Tips for a Great Flex UX: Part 5. *Designing Web Interfaces Principles and Patterns for Rich Interaction*.
- Nerur, S., Mahapatra, R., & Mangalaraj, G. (2005). Challenges of migrating to agile methodologies. *Communications of the ACM*, 48(5), 72–78.
- Newsted, P., Munro, S. H. M., Schwarz, A., & Cu, T. (2014). A Tutorial on Survey Instruments. *aisnet.org*. Haettu 23.4.2014 osoitteesta <http://aisnet.org/?MISSurveyTutorial>
- Nielsen, C. M., Overgaard, M., Pedersen, M., & Stage, J. (2005). Feedback from Usability Evaluation to User Interface Design: Are Usability Reports Any Good? Teoksessa M. Costabile & F. Paternò (Toim.), *Human-Computer Interaction - INTERACT 2005* (Vsk. 3585, s. 391–404). Springer Berlin Heidelberg.
- Nielsen, J. (1992a). The usability engineering life cycle. *Computer*, 25(3), 12–22.
- Nielsen, J. (1992b). Finding usability problems through heuristic evaluation. *Teoksessa Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (s. 373–380). Conference on Human Factors in Computing Systems, Monterey, USA: ACM Press.
- Nielsen, J. (1993). *Usability Engineering*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc.
- Nielsen, J. (1994a). Usability inspection methods (s. 413–414). ACM Press.
- Nielsen, J. (1994b, 1. tammikuuta). Guerrilla HCI: Article by Jakob Nielsen. *www.nngroup.com*. Haettu 3.9.2013 osoitteesta <http://www.nngroup.com/articles/guerrilla-hci/>

- Nielsen, J. (1995a). Applying discount usability engineering. *IEEE Software*, 12(1), 98–100.
- Nielsen, J. (1995b). 10 Heuristics for User Interface Design: Article by Jakob Nielsen. *www.nngroup.com*. Haettu 24.1.2014 osoitteesta <http://www.nngroup.com/articles/ten-usability-heuristics/>
- Nielsen, J. (1995c, 1. tammikuuta). Heuristic Evaluation: How-To: Article by Jakob Nielsen. *www.nngroup.com*. Haettu 24.1.2014 osoitteesta <http://www.nngroup.com/articles/how-to-conduct-a-heuristic-evaluation/>
- Nielsen, J. (2000, 19. maaliskuuta). Why You Only Need to Test with 5 Users. *www.nngroup.com*. Haettu 9.3.2014 osoitteesta <http://www.nngroup.com/articles/why-you-only-need-to-test-with-5-users/>
- Nielsen, J. (2001, 21. tammikuuta). Usability Metrics. *www.nngroup.com*. Haettu 22.9.2013 osoitteesta <http://www.nngroup.com/articles/usability-metrics/>
- Nielsen, J. (2008, 22. tamikuuta). Usability ROI Declining, But Still Strong. *www.nngroup.com*. Haettu 24.1.2014 osoitteesta <http://www.nngroup.com/articles/usability-roi-declining-but-still-strong/>
- Nielsen, J. (2012, 4. tammikuuta). Usability 101: Introduction to Usability. *www.nngroup.com*. Haettu 17.9.2013 osoitteesta <http://www.nngroup.com/articles/usability-101-introduction-to-usability/>
- Nielsen, J., & Landauer, T. K. (1993). A mathematical model of the finding of usability problems. Teoksessa *Proceedings of the INTERACT '93 and CHI '93 Conference on Human Factors in Computing Systems* (s. 206–213). Conference on Human Factors in Computing Systems, Amsterdam, The Netherlands: ACM Press.
- Nielsen, J., & Molich, R. (1990). Heuristic evaluation of user interfaces. Teoksessa *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (s. 249–256). Conference on Human Factors in Computing Systems, Seattle, Washington, USA: ACM Press.
- Northrop, B. (2011, 20. marraskuuta). Usability Testing on Agile Projects (part 1) | Summa Blog. *summa-tech*.
- Northrop, B. (2012, 1. huhtikuuta). Usability Testing on Agile Projects (part 2) | Summa Blog. *summa-tech*.
- Nummiaho, A. (2003, 16. lokakuuta). PPT - Heuristinen arviointi. *hut.fi*. Haettu 31.3.2014 osoitteesta http://www.soberit.hut.fi/T-76.115/03-04/palautukset/groups/PPT/lu/docs/heuristinen_arviointi_esittely.html
- Paetsch, F., Eberlein, A., & Maurer, F. (2003). Requirements engineering and agile software development. Teoksessa *Proceedings of the Twelfth IEEE International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises, 2003. WET ICE 2003*. (s. 308–313). Twelfth IEEE

- International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises, IEEE Comput. Soc.
- Parsons, D., Lal, R., Ryu, H., & Lange, M. (2007). *Software Development Methodologies, Agile Development and Usability Engineering*. Teoksessa *Australasian Conference on Information Systems*.
- Patton, J. (2002). *Hitting the Target: Adding Interaction Design to Agile Software Development*. Teoksessa *OOPSLA 2002 Practitioners Reports* (s. 1–9). New York, NY, USA: ACM.
- Pfleeger, S. L., & Kitchenham, B. (2001). Principles of survey research: part 1: turning lemons into lemonade. *ACM SIGSOFT Software Engineering Notes*, 26(6), 16.
- Poimala, S., & Tolvanen, P. (2011, 8. lokakuuta). Ketteryyks haltuun: Ketterän kehityksen yleiset periaatteet - Sininen Meteoriitti Oy. Haettu 26.2.2014 osoitteesta <https://www.meteoriitti.com/Artikkelisarjat/Ketteryyks-haltuun/Ketteryyks-haltuun-Ketteran-kehityksen-yleiset-periaatteet/>
- Poppendieck, M., & Poppendieck, T. D. (2003). *Lean software development: an agile toolkit*. Boston, Mass.: Addison-Wesley.
- Preece, J. (1994). *Human-computer interaction*. Wokingham, England ; Reading, Mass: Addison-Wesley Pub. Co.
- Puskalahti, A. (2011). *VR verkkokaupan käytettävyyssarvio* (Asiantuntija-arvio) (s. 68).
- Quesenbery, W. (2009, 10). Usability Testing Versus Expert Reviews :: UXmatters. Haettu 5.12.2013 osoitteesta <http://www.uxmatters.com/mt/archives/2009/10/usability-testing-versus-expert-reviews.php>
- Roczek, D., & Tryon, R. (2014, 9. tammikuuta). QA/Bugzilla/Fields/Severity - The Document Foundation Wiki. *Wiki of The Document Foundation*. Haettu 27.3.2014 osoitteesta <https://wiki.documentfoundation.org/QA/Bugzilla/Fields/Severity>
- Rubin, J. (2008). *Handbook of usability testing: how to plan, design, and conduct effective tests* (2. painos). Indianapolis, IN: Wiley Pub.
- Runeson, P., & Höst, M. (2008). Guidelines for conducting and reporting case study research in software engineering. *Empirical Software Engineering*, 14(2), 131–164.
- Salah, D. (2011). A framework for the integration of user centered design and agile software development processes. Teoksessa *Proceedings of the 33rd International Conference on Software Engineering* (s. 1132–1133). The 33rd International Conference on Software Engineering, Waikiki, Honolulu, Hawaii: ACM Press.
- Samlink. (2014a). Monikanavaiset ja tehokkaat IT-palvelut finanssitoimialalle. Haettu 24.3.2014 osoitteesta <http://samlink.fi/Mita/Finanssi/>
- Samlink. (2014b). Samlinkin avainluvut. *www.samlink.fi*. Haettu 11.3.2014 osoitteesta <http://samlink.fi/Me/Samlink-lyhyesti/Avainluvut/>

- Sarafinchan, Q. J. (2010, 15. joulukuuta). Defects - Severity and Priority. *Splat Fx()*. Haettu 1.5.2014 osoitteesta <http://www.splatfx.com/defects-severity-and-priority>
- Sauro, J. (2010, 10. kesäkuuta). Can users self-report usability problems?: Measuring Usability. *Measuringusability.com*. Haettu 10.3.2014 osoitteesta <http://www.measuringusability.com/blog/self-reporting.php>
- Sauro, J. (2013, 18. kesäkuuta). 10 Things to Know About the System Usability Scale (SUS): Measuring Usability. *http://www.measuringusability.com/*. Haettu 20.2.2014 osoitteesta <http://www.measuringusability.com/blog/10-things-SUS.php>
- Schwaber, K. (2002). *Agile software development with Scrum*. Upper Saddle River, NJ: Prentice Hall.
- Schwaber, K., & Sutherland, J. (2013). *The Scrum Guide - The Definitive Guide to Scrum: The Rules of the Game*. Scrum.org.
- Scott, K. M. (2009). FEATURE: Is usability obsolete? *interactions*, 16(3), 6.
- Senapathi, M., & Srinivasan, A. (2012). Understanding post-adoptive agile usage: An exploratory cross-case analysis. *Journal of Systems and Software*, 85(6), 1255–1268.
- Shewhart, W. A. (1986). *Statistical method from the viewpoint of quality control*. New York: Dover.
- Shneiderman, B. (1987). *Designing the user interface: strategies for effective human-computer interaction*. Addison-Wesley.
- Singh, M. (2008). U-SCRUM: An Agile Methodology for Promoting Usability. Teoksessa *In proceeding of: Agile, 2008. AGILE '08*. (s. 555–560). AGILE '08, Toronto, ON: IEEE.
- Sohaib, O., & Khan, K. (2010). Integrating usability engineering and agile software development: A literature review. Teoksessa *International Conference on Computer Design and Applications (ICCD), 2010* (s. V2–32–V2–38). International Conference on Computer Design and Applications, Qinhuangdao, China: IEEE.
- Sohaib, O., & Khan, K. (2011). Incorporating discount usability in extreme programming. *International Journal of Software Engineering and its Applications*, 5(1), 51–62.
- Solis, C., & Wang, X. (2011). A Study of the Characteristics of Behaviour Driven Development. Teoksessa *37th EUROMICRO Conference on Software Engineering and Advanced Applications (SEAA), 2011* (s. 383–387). 37th EUROMICRO Conference on Software Engineering and Advanced Applications, Oulu, Finland: IEEE.
- Spencer, D. (2008, 28. tammikuuta). Why I don't offer a usability testing service. *Information architecture, interaction design and much more*.
- Strode, D. E., Huff, S. L., Hope, B., & Link, S. (2012). Coordination in co-located agile software development projects. *Journal of Systems and Software*, 85(6), 1222–1238.
- Sy, D. (2007). Adapting usability investigations for agile user-centered design. *Journal of Usability Studies*, 2007(2), 112–132.

- Talby, D., Hazzan, O., Dubinsky, Y., & Keren, A. (2006). Agile Software Testing in a Large-Scale Project. *IEEE SOFTWARE*, 30–37.
- Thomas, B. (1996). Quick and dirty usability tests. *Usability evaluation in industry*. London: Taylor & Francis, 107–114.
- Tognazzini, B. (2003). AskTog: First Principles of Interaction Design. *AskTog*. Haettu 24.1.2014 osoitteesta <http://www.asktog.com/basics/firstPrinciples.html>
- Tullis, T. (2013, 3. helmikuuta). Card-sorting Tools. *www.measuringuserexperience.com*. Haettu 4.9.2013 osoitteesta <http://measuringuserexperience.com/CardSorting/index.htm>
- Tuomi, J., & Sarajärvi, A. (2013). *Laadullinen tutkimus ja sisällönanalyysi* (1. painos). Tammi.
- Turner, R., Ingold, D., Lane, J. A., Madachy, R., & Anderson, D. (2012). Effectiveness of kanban approaches in systems engineering within rapid response environments. *Procedia Computer Science*, 8, 309–314.
- Turun Sanomat. (2012, 6. tammikuuta). VR:n verkkokaupan käytettävyys takkuua yhä - Kotimaa - Turun Sanomat. *www.ts.fi*. Uutissivusto. Haettu 4.9.2013 osoitteesta <http://www.ts.fi/uutiset/kotimaa/296159/VRn+verkkokaupan+kaytet+avyys+takkuua+yha>
- Twain, M. (1940). *Mark Twain in eruption: hitherto unpublished pages about men and events* (1. painos). New York: Harper.
- UX for the masses. (2010a, 17. syyskuuta). Usability testing hints, tips and guidelines. *UX for the masses*. Haettu 11.1.2014 osoitteesta <http://www.uxforthemasses.com/usability-testing/>
- UX for the masses. (2010b, 17. syyskuuta). System usability scale questionnaire. *UX for the masses*. Haettu osoitteesta <http://www.uxforthemasses.com/blog/wp-content/uploads/2010/09/System-usability-scale-questionnaire.pdf>
- Weiss, T. (2000, 28. syyskuuta). Amazon apologizes for price-testing program that angered customers - Computerworld. Haettu 2.12.2013 osoitteesta http://www.computerworld.com/s/article/51392/Amazon_apologizes_for_price_testing_program_that_angered_customers
- Welie, M. V., Veer, G. C. V. D., & Eliëns, A. (1999). Breaking down Usability. Teoksessa *Proceedings of Interact '99* (s. 613–620). Press.
- VersionOne. (2013). *7th Annual State of Agile Development Survey*.
- West, R., & Lehman, K. (2006). Automated Summative Usability Studies: An Empirical Evaluation. Teoksessa *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (s. 631–639). New York, NY, USA: ACM.
- Wiklund-Engblom, A., Hassenzahl, M., Bengs, A., & Sperring, S. (2009). What Needs Tell Us about User Experience. Teoksessa T. Gross, J. Gulliksen, P. Kotzé, L. Oestreicher, P. Palanque, R. O. Prates, & M. Winckler (Toim.), *Human-Computer Interaction – INTERACT 2009* (Vsk. 5727, s. 666–669). Berlin, Heidelberg: Springer Berlin Heidelberg.

- Winter, J., Rönkkö, K., Ahlberg, M., & Hotchkiss, J. (2011). Meeting Organisational Needs and Quality Assurance through Balancing Agile and Formal Usability Testing Results. Teoksessa Z. Huzar, R. Koci, B. Meyer, B. Walter, & J. Zendulka (Toim.), *Software Engineering Techniques* (Vsk. 4980, s. 275–289). Berlin, Heidelberg: Springer Berlin Heidelberg.
- Virzi, R. A. (1992). Refining the Test Phase of Usability Evaluation: How Many Subjects Is Enough? *Human Factors: The Journal of the Human Factors and Ergonomics Society*, 34(4), 457–468.
- Wixon, D. (2003). Evaluating Usability Methods: Why the Current Literature Fails the Practitioner. *interactions*, 10(4), 28–34.
- Vredenburg, K., Mao, J.-Y., Smith, P. W., & Carey, T. (2002). A survey of user-centered design practice. Teoksessa *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (s. 471). Conference on Human Factors in Computing Systems '02, Minneapolis, MN: ACM Press.
- Yin, R. K. (2009). *Case study research: design and methods* (4th ed.). Los Angeles, Calif: Sage Publications.

LIITE 1 SAMLINK KOEHENKILÖTIETOKANNAN TIETOJEN KYSELYLOMAKE

SAMLINK

Käytettävyydestauksen koehenkilökysely

Tämän kyselyn täyttäneitä henkilöitä voidaan kutsua koehenkilöiksi käytettävyydestejä varten. Kyselyssä tiedustellaan muutamia taustatietoja mielenkiinnon kohteistasi ja arvioita taidoistasi joidenkin laitteiden tai palveluiden käytössä. Näiden tietojen perusteella valitaan koehenkilöitä eri projekteihin.

Rekisteröidy osallistuaksesi tutkimukseen

Halutessasi rekisteröityä tähän tutkimukseen kirjaa tietosi alapuolelle. Tietoja käytetään vain käytettävyydestauksen koehenkilöiden valintaan ja kutsumiseen. Tietojen antamisen jälkeen pääset jatkamaan kyselyyn.

Etunimi:

Sukunimi:

Sähköpostiosoite:

Turvallisuuskysymys:

Rekisteriseloste löytyy osoitteesta <https://dl.dropboxusercontent.com/u/18835906/rekisteriseloste/rekisteriseloste.pdf>

Ensimmäinen sivu, rekisteröityminen.

SAMLINK

Käytettävyydestauksen koehenkilökysely

Tämän kyselyn täyttäneitä henkilöitä voidaan kutsua koehenkilöiksi käytettävyydestejä varten. Kyselyssä tiedustellaan muutamia taustatietoja mielenkiinnon kohteistasi ja arvioita taidoistasi joidenkin laitteiden tai palveluiden käytössä. Näiden tietojen perusteella valitaan koehenkilöitä eri projekteihin.

Kiitos rekisteröitymisestäsi ja osallistumisestasi kyselyyn.

Tutkimuksen hallinta Antti K. Koskela - Samlink (antti.koskela@samlink.fi)

JATKA KYSELYYN!
(Automaattinen uudelleenohjaus 5 sekunnin kuluttua...)

Rekisteriseloste löytyy osoitteesta <https://dl.dropboxusercontent.com/u/18835906/rekisteriseloste/rekisteriseloste.pdf>

Toinen sivu, perustietoja ja kiitokset rekisteröitymisestä.

Antti K. Koskela - Samlink

-> minä ▾

Hyvä Testi,

Olet rekisteröitynyt tutkimukseen Käytettävyydestauksen koehenkilökysely.

Vastataksesi tutkimukseen klikkaa alla olevaa linkkiä:

<http://kyselyt.norppandalotti.info/index.php?lang=fi&sid=62458&token=wg7gbhs9unqyms>

Ylläolevasta linkistä pääset myös myöhemmin muokkaamaan antamiasi vastauksia!

Paina alla olevasta linkistä päästäksesi vastaamaan uudestaan (esimerkiksi antaaksesi jonkun muun tiedot hänen puolestaan):

<http://kyselyt.norppandalotti.info/index.php?lang=fi&sid=62458&newtest=Y>Jos sinulla on kysymyksiä tutkimuksesta ota yhteyttä: antti.koskela@samlink.fi.

Sähköpostiin tuleva vahvistus rekisteröitymisestä ja linkki, jonka kautta voi täydentää tietojaan tai antaa uuden henkilön tiedot.



SAMLINK

Käytettävyydestauksen koehenkilökysely

Tämän kyselyn täyttäneitä henkilöitä voidaan kutsua koehenkilöiksi käytettävyydestejiä varten. Kyselyssä tiedustellaan muutamia taustatietoja mielenkiinnon kohteistasi ja arvioita taidoistasi joidenkin laitteiden tai palveluiden käytössä. Näiden tietojen perusteella valitaan koehenkilöitä eri projekteihin.

Tervetuloa käytettävyydestauksen koehenkilöiden taustatietokyselyyn! Tämän kyselyn tietoja käytetään Samlinkin Sähköisen asiointin projektien käytettävyydestauksen koehenkilöiden valinnassa. Käytettävyydestaus tehdään Antti Koskelan pro gradu -tutkielman puitteissa.

Kyselyyn vastaamisessa kestää arviolta 5-10 minuuttia, ja se koostuu henkilötietojen ja, osaamiseen ja kiinnostukseen liittyvistä kysymyksistä. Tietoja käytetään vain mahdollisimman relevanttien koehenkilöiden valintaan käytettävyydestausessioita suunniteltaessa.

Pakolliset kysymykset on osoitettu * -merkillä.

Mikäli kyselyyn liittyy kysymyksiä, ne voi lähettää osoitteeseen antti.koskela@samlink.fi.

[Seuraava >>](#)

[Poistu kyselystä ja tyhjennä vastaukset.](#)

Rekistenseloste löytyy osoitteesta <https://dl.dropboxusercontent.com/u/18835906/rekistenseloste/rekistenseloste.pdf>

Varsinaisen kyselyn ensimmäinen sivu: tervetuloivotus ja ohjeita.

SAMLINK

Käytettävyydestauksen koehenkilökysely

Tämän kyselyn täyttäneitä henkilöitä voidaan kutsua koehenkilöiksi käytettävyydesteitä varten. Kyselyssä tiedustellaan muutamia taustatietoja mielenkiinnon kohteistasi ja arvioita taidoistasi joidenkin laitteiden tai palveluiden käytössä. Näiden tietojen perusteella valitaan koehenkilöitä eri projekteihin.

0% 100%

Henkilötiedot

Tällä sivulla kysellään joitakin henkilötietoja. Näitä tietoja käytetään, jos kutsumme sinut käytettävyydestaustausseesioon.

Varsinaisten käytettävyydestaustausseesioden tuloksia (kuten pöytäkirjaa) ei sellaisenaan näytetä muille kuin session valvojille (ja pyynnöstä koehenkilölle itselleen), eikä henkilöllisyyksiä ikinä yhdistetä käytettävyyseraportteihin.

1 Sukupuolesi

Nainen Mies Ei vastausta

2 Puhelinnumerosi

? Jos haluat että sinuun otetaan yhteyttä puhelimitse (tekstiviestillä) sähköpostin sijasta, annathan puhelinnumerosi tähän!

3

Perhekuntasi tyyppi?

Valitse sopiva vaihtoehto

Asun yksin
 Asun yhdessä kaverini tai jonkun muun kanssa
 Asun yhdessä puolisoni kanssa
 Asun yhdessä puolisoni ja lastemme kanssa
 Asun yhdessä lasteni kanssa
 Asun yhdessä vanhempieni kanssa
 Ei vastausta

? Minkälaisista henkilöistä perhekuntasi koostuu? Puolisolla tarkoitetaan tässä myös avopuolisoa tai tyttö- tai poikaystävää.

<< Edellinen Seuraava >>

Poistu kyselystä ja tyhjennä vastaukset.

Jatka myöhemmin

Rekisteriseloste löytyy osoitteesta <https://dl.dropboxusercontent.com/u/18835906/rekisteriseloste/rekisteriseloste.pdf>

Varsinaisen kyselyn toinen sivu: henkilö- ja taustatietoja

Tilan säästämiseksi loppuosa kyselystä on tässä kuvakaappausten sijasta teksti/html-muodossa.

Kyselyn tässä osiossa tiedustellaan joitakin osaamisesi alueita. Tarvitsemme erilaisella osaamisella varustettuja koehenkilöitä, joten meille on tärkeää tietää ainakin karkea arviosi osaamisestasi mm. internet-asioissa ja toimisto-ohjelmissä.

Lisäksi tiedustelemme joitakin asioita työelämäkokemuksistasi, jotta saamme kohdistettua sinulle relevanteimpia kyselyitä.

Kysymyksissä, joissa pyydämme sinua arvioimaan asteikolla 1-5 kuinka samaa mieltä olet väittämien kanssa, noudatetaan seuraavaa asteikkoa:

1 = ei ollenkaan samaa mieltä

3 = ei samaa eikä eri mieltä

5 = täysin samaa mieltä.

4 Osaan käyttää sujuvasti seuraavia ohjelmia tai palveluita:

	1	2	3	4	5	Ei vastausta
Internet-selaimen käyttö	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
Hakukoneiden ja -palveluiden käyttö	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
Intranet-järjestelmät (SharePoint, Liferay yms.)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
Wiki-järjestelmät (Wikipedia, Confluence yms.)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>

? Kuinka hyväksi arvioit osaamisesi seuraavilla osa-alueilla?

5 Osaan käyttää sujuvasti seuraavia toimisto-ohjelmia:

	1	2	3	4	5	Ei vastausta
Tekstinkäsittely (esim. MS Office Word, OpenOffice/LibreOffice Writer)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>

	1	2	3	4	5	Ei vastausta
Diashow (esim. MS Office PowerPoint, OpenOffice/LibreOffice Impress)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
Taulukkolaskenta (esim. MS Office Excel, OpenOffice/LibreOffice Calc)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
Muut toimisto-ohjelmat (esim. Access, Notes...)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>

6 Työ- tai muu elämäntilanteesi?

	Koulu- lainen (pe- rusk.)	Opis- keak. (2.aste)	Opis- kelija (kor- keak.)	Elä- ke- Työ- läi- Työn- teki- jä	Alempi toimi- hen- kilö	Ylempi toimi- hen- kilö tai joh- taja	Muu tai ei Yrit- täjä tausta	Ei vas- vas- tausta
1. tär- kein teh- tä- väsi	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
2. tär- kein teh- tä- väsi	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
3. tär- kein teh- tä- väsi	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>

? Valitse tässä päätoiminen ja sivutoimiset työtilanteesi: 1. valinta on olennaisin, muut voit hyvällä omalla tunnolla jättää kohtaan "ei vastausta" jos haluat.

Jos olet esimerkiksi osa-aikaisesti töissä ja opiskelet päätoimisesti, valitse opiskelija 1. valinnaksesi ja työtilannettasi kuvaava valinta 2. valinnaksesi, ja jos sinulla esimerkiksi on lisäksi sivutoiminen yritys, laita se 3. valinnaksesi.

Jos olet keskikokoisen tai suurehkon yrityksen johtaja, laita yrittäjä ensisijaiseksi valinnaksesi ja ylempi toimihenkilö toiseksi!

***7 Osaan käyttää itsenäisesti ja monipuolisesti seuraavia laitteita ja tunnen hyvin niiden ominaisuudet:**

	1	2	3	4	5
Pöytätietokone	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Kannettava tietokone	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Älypuhelin kosketusnäytöllä	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Muu älypuhelin (ei kosketusnäyttöä)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Perinteinen matkapuhelin	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Tabletti (esim. iPad)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

?

Käytetään seuraavaa asteikkoa:

1 = laite täysin vieras

3 = keskimääräinen käyttäjä, itselle tärkeimmät toiminnallisuudet tuttuja

5 = suvereeni käyttäjä, käytän esimerkiksi laitetta työkseni ja tunnen valtaosan toiminnallisuuksista

8

Minkä verran arvelet sinulla olevan kokemusta seuraavista toimialoista?

1 = Ei kokemusta

2 = Tiedän suunnilleen mistä alalla on kysymys

3 = Ala on tuttu mutta ei merkittävää kokemusta

4 = Olen opiskellut tai työskennellyt alalla tai siihen liittyvissä tehtävissä
5 = Olen työskennellyt alalla vuosia tai tutkinut sitä.

	1	2	3	4	5	Ei vastausta
Maa-, metsä- ja kalatalous	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
Kaivostointi	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
Teollisuus	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
LVI- ja huoltoala (Sähkö-, kaasu- ja lämpöhuolto, jäähdytysliiketoiminta, vesihuolto, viemäri- ja jätevesihuolto, jätetuho ja muu ympäristön puhtaanapito)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
Rakentaminen	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
Tukku- ja vähittäiskauppa	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
Auto- ja konehuolto	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
Kuljetus ja varastointi	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
Majoitus- ja ravitsemistointi	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
Informaatio ja viestintäala	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
Rahoitus- ja vakuutustointi	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
Kiinteistöalan toiminta	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
Ammatillinen, tieteellinen ja tekninen toiminta (erityisesti tutkimus)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>

	1	2	3	4	5	Ei vastausta
Ammatillinen, tieteellinen ja tekninen toiminta (erityisesti asiantuntijapalvelut)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
Julkinen hallinto	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
Maanpuolustus tai muu sotilaallinen toiminta	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
Koulutus	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
Terveys- ja sosiaalipalvelut	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
Taiteet, viihde ja virkistys	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
Kansainvälisten organisaatioiden ja toimielinten toiminta	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>



Toimialaluokituksena on käytetty Tilastokeskuksen luokitusta.

Tämä kysymys osaltaan määrittää, minkälaisista kyselyistä sinulle lähetetään jatkossa. Kysymykseen ei silti ole pakko antaa vastausta!

Kiinnostuksen kohteet ja sessioon osallistuminen

Tällä sivulla kartoitamme kiinnostuksen kohteitasi, jotta osaamme suunnata sinulle sinua ainakin potentiaalisesti kiinnostavia käytettävyytestaussessioita (asiakkaamme ja projektimme ovat vaihtelevilta toimialoilta).

***9 Mihin käyttäjärooleihin arvelisit osaavasi eläytyä? Roolista annetaan tarkempi kuvaus ennen käytettävyytestaussessiota.**

Valitse kaikki sopivat vaihtoehdot

- Kuluttajakäyttäjä
- Sähköinen asiakaspalvelija
- Asiakaspalvelijoiden esimies

***10 Minkätyyppisiin käytettävyytestaustuntoihin olisit valmis osallistumaan?**

Valitse kaikki sopivat vaihtoehdot

- Käytettävyytestaustessio, jossa pääset kokeilemaan vapaasti testattavaa järjestelmää
- Käytettävyytestaustessio, jossa toteutat ennalta määrättyjä tehtäviä testattavalla järjestelmällä
- Käytettävyytestaustessio, jossa pääset ilmaisemaan mielipiteesi järjestelmän ulkoasusta
- Web-kysely, jossa voit arvioida testattavan järjestelmän navigaation toimivuutta
- Web-kysely, jossa pääset arvioimaan testattavan järjestelmän terminologisuutta
- Ei mikään edellisistä

***11 Valmiutesi osallistua Jyväskylässä järjestettävään käytettävyytestaustuntoon?**

- Kyllä
- Ei



Varsinaiset testaustunnot järjestetään Jyväskylän alueella, projektista riippuen esimerkiksi Samcomin toimipisteellä (Kauppakatu 28B), yliopiston kampuksilla, kahvilassa tai vaikka koehenkilön kotona - mikä sopii osapuolille ja testauksen luonteelle parhaiten. Ne tullaan kuitenkin järjestämään Jyväskylässä, joten näihin sessioihin osallistumisen edellytyksenä on mahdollisuus olla paikalla sessiossa Jyväskylässä.

Käytettävyytestaukseen osallistuminen on mahdollista myös muualla kuin Jyväskylässä, mutta se vaatii eri metodeita tai ainakin aikataulujen sopimista tarkemmin.

12

Mikä olisi mieluisin palkkio käytettävyytestaustessioista? Kuvittele tässä tehtävässä, että kaikki palkkiot ovat rahallisesti saman arvoisia.

Valitse kaikki sopivat vaihtoehdot

- Subway-lahjakortti tai -lahjakortteja
- Finnkino-leffalippu tai -leffalippuja
- Spotify-lahjakortti

- Filmtown-lahjakortti
- Netflix-lahjakortti
- Muu:

? Verotuksellisista syistä palkkiota ei valitettavasti voi valita vapaasti esimerkiksi session jälkeen useista vaihtoehdoista (se olisi silloin palkkion saajan verotettavaa tuloa), mutta tällaisella kyselyllä varmistetaan että voidaan tarjota palkkiota joka kuitenkin kelpaa koehenkilöille.

13 Missä paikoissa käytät internetiä?

	En koskaan	Harvoin	Joskus	Usein	Yleensä	Ei vastausta
Kotona	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
Työpaikalla	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
Koulussa/yliopistolla	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
Kahvilassa tai muilla julkisilla paikoilla	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>

? Jos jokin kohta ei kuvaa lainkaan tilannettasi (esimerkiksi et opiskele), anna kohdalle ("koulussa/yliopistolla") arvoksi "En koskaan".

*14 Millä laitteilla käytät internetiä?

	En koskaan	Harvoin	Joskus	Usein	Yleensä
Pöytätietokoneella	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Kannettavalla tietokoneella	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Tabletilla (esim. iPad)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Älypuhelimella	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

15

Minkä verran seuraavat toimialat sinua kiinnostavat?

	1	2	3	4	5	Ei vastausta
Maa-, metsä- ja kalatalous	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
Kaivostointa	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
Teollisuus	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
LVI- ja huoltoala (Sähkö-,	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>

	1	2	3	4	5	Ei vastausta
kaasu- ja lämpöhuolto, jäähdytysliiketoiminta, vesihuolto, viemäri- ja jätevesihuolto, jätetuhoolto ja muu ympäristön puhtaanapito)						
Rakentaminen	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
Tukku- ja vähittäiskauppa	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
Auto- ja konehuolto	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
Kuljetus ja varastointi	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
Majoitus- ja ravitsemistoiminta	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
Informaatio ja viestintäala	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
Rahoitus- ja vakuutustoiminta	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
Kiinteistöalan toiminta	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
Ammatillinen, tieteellinen ja tekninen toiminta (erityisesti tutkimus)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
Ammatillinen, tieteellinen ja tekninen toiminta (erityisesti asiantuntijapalvelut)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
Julkinen hallinto	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
Maanpuolustus tai muu	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>

	1	2	3	4	5	Ei vastausta
sotilaallinen toiminta						
Koulutus	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
Terveys- ja sosiaalipalvelut	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
Taiteet, viihde ja virkistys	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
Kansainvälisten organisaatioiden ja toimielinten toiminta	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>



Toimialaluokituksena on käytetty Tilastokeskuksen luokitusta.

Tämä kysymys osaltaan määrittää, minkälaisista kyselyistä sinulle lähetetään jatkossa. Kysymykseen ei ole pakko antaa vastausta!

1 = ei yhtään, 5 = paljon.

Lopuksi

Lopuksi esitämme sinulle vielä pari kysymystä tähän kyselyyn liittyen.

16 Mistä sait kuulla tästä kyselystä?

Valitse kaikki sopivat vaihtoehdot

- Suoraan kyselyn järjestäjältä
- Tuttava vinkkasi/linkkasi
- Postituslistalta
- Internetistä, mistä?
- Muualta, mistä?
- Muu:

17 Terveiset ja palaute kyselystä

Vastaus

LIITE 2 KUVAUS PROJEKTEISTA

Tässä dokumentissa kuvataan lyhyesti projektit, joissa käytettävyydestäusta tehtiin. Projektit eroavat toisistaan keskenään suurestikin: RETU on sisäiseen käyttöön kehitetty järjestelmä, PA on moniin muihin projekteihin ja järjestelmiin liitettävä finanssialan asiakkaalle räätälöity järjestelmä, SH on useille asiakkaille toimitettava Samlinkin oma tuote ja HE-projektissa testataan järjestelmää joka on yhdistelmä useiden eri toimittajien verkkopohjaista ratkaisusta.

1. Sisäinen työsuunnittelun apuväline, **RETU**

Projektissa kehitetään sisäinen järjestelmä resursoinnin apuvälineeksi. Järjestelmä on web-pohjainen ja koko henkilöstön käytössä. Henkilöstön edustajilla on eri rooleja järjestelmässä, ja siten eri toiminnot ovat heille aktiivisia. Roolit määrittyvät sen mukaan, onko työntekijä osaja, resurssipyynnön tekijä, resursoija tai esimies. Useat työntekijät ovat useamman roolin edustajia.

Järjestelmän testaus on tärkeää, koska esitellessä henkilöstölle uusia järjestelmiä ja uusia toimintatapoja on muutosvastarinta aina suuri ongelma. Kiinnittämällä huomiota järjestelmän käytettävyyteen saadaan toisaalta helpotettua järjestelmän hyväksymistä, toisaalta myös annettua osoitus siitä, että työntekijöiden palautteesta ja mielipiteistä todella välitetään. Käyttäjistä on tunnistettu muutamia suurempia käyttäjäryhmiä: resursoijat (käyttävät kaikkia toimintoja), resurssipyynnön tekijät (kuten projektipäälliköt ja ratkaisukonsultit, käyttävät monia toimintoja), esimiehet (käyttävät vain joitakin toimintoja) ja osajat sekä loput (lähinnä tarkastelevat tietoja).

Apuvälinettä testataan kahdesti, ensin julkaisun aikoihin (käytännön rajoitteista aiheutunut aikataulu), ja toisen kerran kun valtaosa ensimmäisen testauksen havainnoista on saatu paikattua, noin kuukauden kuluttua ensimmäisestä testistä.

Tästä projektista käytetään lyhennettä **RETU**.

2. Finanssialan asiakkaiden ajanvarausjärjestelmä, **PA**

Tässä tapauksessa testattavia projekteja on itse asiassa kaksi, mutta niiden käytettävyydestä suoritetaan samalla kertaa. Kyseessä on finanssialan asiakkaamme ajanvarausjärjestelmä, joka on upotettu heidän uusille nettisivuilleen. Koehenkilöt edustavat kuluttaja-asiakkaan roolia.

Testauksen tavoitteena on saada tuloksia erityisesti ajanvarausjärjestelmän käytettävyydestä, mutta samalla myös nettisivuista muuten ja koko järjestelmän intuitiivisuudesta käyttäjän kannalta. Erilaiset, eri käyttäjäryhmiä edustavat koehenkilöt jakautuvat (valitettavasti) eri sprintteihin siten, että ensimmäisessä sprintissä testataan kuluttajakäyttäjillä, toisessa sprintissä asiakkaan edustajilla ja järjestelmän käyttäjätuen henkilöstöllä, ja kolmannessa sprintissä vielä kuluttajakäyttäjillä.

Projektin ensimmäisessä testaussprintissä tuotetaan dataa sekä nettisivu-että ajanvarausprojektille, toisessa vaiheessa testataan vain ajanvarausjärjestelmää ja sen ylläpitotoimintoja, ja kolmannessa vaiheessa taas nettisivu- ja ajanvarausnäkyymiä. Tulokset siis jakautuvat kahden erillisen projektin hyödynnettäväksi, mutta painopisteenä on ajanvarausprojekti.

Tästä projektista käytetään lyhennettä **PA**.

3. Energia-alan asiakkaiden asiakaspalveluväylä, **SH**

Projektissa testauksen kohteena on eräänlainen energia-alan yrityksen kuluttaja-asiakkaille suunnattu myyntiportaali. Myyntiportaalin lopullisessa versiossa on muun muassa sopimusten hallinta, mutta tässä vaiheessa testattavassa versiossa on uuden sähkö sopimuksen teko.

Tästä projektista käytetään lyhennettä **SH**.

4. Elintarvike-alan yritys, **HE**

Tässä projektissa testataan yrityksen yhden toimialan nettisivuja. Menetelmänä on heuristisen evaluoinnin ja kognitiivisen läpikäynnin yhdistävä heuristinen läpikäynti. Käytännössä tässä hybridissä siis tehdään sekä kognitiivinen läpikäynti (suppeasti) että vapaamuotoisempi heuristinen evaluointi. Arviointiin osallistuu useampi käytettävyydestaaja, joista olen yksi.

Tästä projektista käytetään lyhennettä **HE**.

LIITE 3 HAVAINTOJA KÄYTETTÄVYYSTESTAUSSESSIÖIDEN JÄRJESTELYISTÄ

Ensimmäisen käytettävyystestauskerran havainnot

Projekti:	RETU
Osanottajat:	Sisäisiä koehenkilöitä: 1 resursoija, 1 projektipäällikkö, 1 esimies ja 1 osaaja.
Menetelmät:	Yksinkertaistettu ääneen ajattelu Kysely
Havainnot:	<ul style="list-style-type: none"> • Osanottajat antoivat mielellään palautetta järjestelyistä, myös rakentavaa kritiikkiä. • Jokaisessa istunnossa havainnot kertyi niin paljon, että tuloksia oli pakko karsia rankasti loppuraporttia varten. • Myös hiljaiseksi oletetut käyttäjät pärjäsivät hyvin yksinkertaistettua ääneen ajattelua käytettäessä, korkeintaan kaipasivat hiukan rohkaisua. • Vaikka koehenkilöt vaikuttaisivat yhtä tyytyväisiltä tai tyytymättömiltä johonkin järjestelmän ominaisuuteen tai mitattavaan suureeseen, kuten helppokäyttöisyyteen, he voivat antaa hyvin erilaisia arvioita kysyttäessä heidän arviotaan siitä asteikolla 1-5. • Koehenkilöt eivät yleensä turvautuneet ohjemateriaaleihin, vaikka sellaisia olisi tarjolla. • Projektipäällikön suhtautuminen käytettävyystestaukseen erittäin positiivinen. • Sisäisten koehenkilöiden rekrytointi ja tapaamisten bookaus oli vaivatonta.

Toisen käytettävyystestauskerran havainnot

Projekti:	PA
Osanottajat:	6 kuluttajakäyttäjää simuloivaa koehenkilöä
Menetelmät:	Suljettu korttien lajittelu Yksinkertaistettu ääneen ajattelu Kysely
Havainnot:	<ul style="list-style-type: none"> • Rekrytointiin ja tapaamisten organisointiin kului aikaa huomattavasti enemmän kuin sisäisten koehenkilöiden kohdalla. • VPN-yhteyksiin ei voi luottaa, joten testaaminen muualla kuin toimistolla on vähintäänkin haastavaa.

	<ul style="list-style-type: none"> • Asiakas suhtautui erittäin positiivisesti – erityisesti, kun testauksesta ei tullut lisäkuluja! • Muutaman päivän varoitusajalla (rekrytointi edellisenä viikonloppuna, testisessiot torstaina ja perjantaina) on mahdollista saada pienestäkin joukosta muutamia vapaaehtoisia. • Käyttäjien taitojen välillä suuria eroja. • Ainakin osa täysin vieraistakin koehenkilöistä olivat luontevia tilanteessa. • ”Tehtävänannon” selkeydellä on suuri merkitys, toisaalta se ei saa olla liian laaja tai ohjaava. • Projektipäällikön kommentti raportin saatuaan: ”Täytyy sanoa, että tämä olisi sittenkin kannattanut tehdä jo kuukausi sitten...” • Projektiryhmiltä toive raportin järjestämisestä esimerkiksi ”näkymittäin” heuristiikkaperusteisen jaottelun sijasta. Toisaalta heuristiikkoihin perustuvaa jaottelua projektipäällikkö piti hyvänä, ja projektipäällikkö toivoi seuraavaa: <ul style="list-style-type: none"> ○ Jatkossa havainnot Excel-tiedostoon, jossa <ul style="list-style-type: none"> ▪ tieto sekä näkymästä että ▪ heuristiikasta, että ▪ arvio vakavuudesta.
--	--

Kolmannen käytettävyytestauskerran havainnot

Projekti:	PA
Osanottajat:	2 asiakkaan edustajaa, 2 järjestelmän tukihenkilöä
Menetelmät:	Korttien lajittelu Yksinkertaistettu ääneen ajattelu Kysely
Havainnot:	<ul style="list-style-type: none"> • Rekrytoinnin ja aikataulutuksen hoiti projektipäällikkö, jolloin testaajalle jäävä työmäärä oli huomattavasti pienempi • Testitapausten hiominen edellisen iteraation testeistä kesti noin tunnin, infran valmistelu toista tuntia, testaus noin 4 tuntia ja tulosten analysointi ja jalostaminen myöskin noin 4 tuntia. • Vaikka raportointikäytäntö ei ole erityisen kevyt (vrt. esim. Krugin vastaavaan) niin tällä menettelytavalla koko käytettävyytestausiteraatio on päivässä ohi, tulosten käsittely jää seuraavalle päivälle.

	<ul style="list-style-type: none"> • Asiakkaan edustajat, erityisesti jos ovat projektin johtoa, ovat haastavia koehenkilöitä sillä heidän näkökulmansa voi olla varsin kriittinen eikä todennäköisesti edusta loppukäyttäjän näkökulmaa <ul style="list-style-type: none"> ○ Heidän kommenttinsa voivat kyllä olla erinomaisia, tulosten tulkinta vaatii jälleen vain oikeanlaista suhtautumista. ○ Erityisesti korttien lajittelun tuloksiin täytyy suhtautua varauksella • Käytettävyydestä paljastaa myös usein bugeja käytettävyyssongelmien lisäksi
--	--

Neljännän käytettävyydestäuskerran havainnot

Projekti:	SH
Osanottajat:	6 kuluttajakäyttäjää simuloivaa koehenkilöä
Menetelmät:	Yksinkertaistettu ääneen ajattelu Kysely
Havainnot:	<ul style="list-style-type: none"> • Kun rekrytointi aloitetaan tarpeeksi ajoissa, kuten pari viikkoa ennen istuntoja, on tarpeeksi osanottajia helppo saada. • 6 koehenkilöä tuottaa todella paljon havainnot • Vaikka varsinaisten testitapausten läpikäynti olisi nopeaa, testauksessa kuluu silti helposti vähintään 45 minuuttia. • Osa koehenkilöistä käyttää tavattoman pitkään loppukyselyyn vastaamisessa – tämä täytyy huomioida ajoituksessa • Käytettävyydestä paljastaa myös usein bugeja käytettävyyssongelmien lisäksi

Viidennen käytettävyydestäuskerran havainnot

Projekti:	RETU
Osanottajat:	Sisäisiä koehenkilöitä: 1 esimies, 1 toimistohenkilö, 3 osaajaa
Menetelmät:	Yksinkertaistettu ääneen ajattelu Kysely
Havainnot:	<ul style="list-style-type: none"> • Sisäinen rekrytointi nopeaa ja vaivatonta, ja istuntojen aikataulutus helppoa • Testaus, analysointi ja tulosten toimittaminen onnistuvat yhden päivän aikana, mutta päivä venyy helposti aika pitkäksi


	<ul style="list-style-type: none"> • 30–45 min testaussessioiden sopiminen tunnin päähän toisistaan on liian tiuha tahti: tuloksia ei ehdi yhtään koostaa sessioiden välissä eikä mahdollisiin valmisteluihin jää aikaa. Lisäksi koehenkilön myöhästyminen session alusta aiheuttaa helposti ketjureaktion. • Havainnot kertyvät tuhattomasti. Jos ei haeta testeiltä kattavuutta, vaan riittää, että joitakin vakavia ongelmia löytyy, ei koehenkilöitä kannata rekrytoida kovin paljon.
--	---

Kuudennen käytettävyystestauskerran havainnot

Projekti:	PA
Osanottajat:	6 kuluttajakäyttäjää simuloivaa koehenkilöä
Menetelmät:	Korttien lajittelu Yksinkertaistettu ääneen ajattelu Kysely
Havainnot:	<ul style="list-style-type: none"> • Järjestelmän käytettävyys oli jo aika hyvä, joten havainnot ei kertynyt paljon.

LIITE 4 KÄYTETTÄVYYSTESTAUSSESSION LOPPUKYSELY

Tässä liitteessä esitellään kysely, joka esitettiin koehenkilöille käytettävyystestausession jälkeen kvantitatiivisen datan keräämiseksi.



SAMLINK käytettävyystestauksen loppukysely - PROJEKTINNIMI

Järjestelmän käytettävyys

Tämän sivun monivalintakysymyksissä käytettävä asteikko on seuraava:

1 = Täysin eri mieltä
 2 = Jokseenkin eri mieltä
 3 = Ei samaa eikä eri mieltä
 4 = Jokseenkin samaa mieltä
 5 = Täysin samaa mieltä

*** Käytettävyystestausistunnon päiväys**

Muotoilu: dd.mm.yyyy

? Tätä kenttää käytetään tulosten seurantaan ja validointiin.

Järjestelmän käytettävyys.	1	2	3	4	5	Ei vastausta
Järjestelmä on hyödyllinen	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Järjestelmä on yksinkertainen	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Järjestelmä on helppokäyttöinen	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Järjestelmää käyttäessä ei tarvitse teknistä tukea	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Järjestelmän eri toiminnot sopivat hyvin yhteen	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Järjestelmän ulkoasu oli yhtenäinen	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Järjestelmän käytön opettelu ei vie aikaa	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Järjestelmän käyttö on vaivatonta ja sujuvaa	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Järjestelmän käyttäminen on miellyttävää	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Järjestelmän sujuvaan käyttö ei vaadi harjoittelua	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Avoin palaute koskien järjestelmää

? Tuliko järjestelmää koskien jotain vielä mieleen? Tähän kenttään voit jättää halutessasi anonymiä palautetta tai kommentteja!

Avoin palaute koskien koejärjestelyitä

? Koejärjestelyitä ja mm. kysymyksiä kehitetään palautteen perusteella! Tähän kenttään voit antaa anonymisti palautetta jonka avulla pyrimme parantamaan käytettävyystestausprosessiamme.

Rekisteriseloste löytyy osoitteesta <https://dl.dropboxusercontent.com/u/18835906/rekisteriseloste/rekisteriseloste.pdf>

Loppukyselylomakkeen Versio 1 (käytetty sisäisessä resursoinnin aputyökalun kehitysprojektin testauksessa)

SAMLINK

Samlink käytettävyydestäuksen loppukysely - PA

Järjestelmän käytettävyys

Tämän sivun monivalintakysymyksissä käytettävä asteikko on seuraava:

1 = Täysin eri mieltä
2 = Jokseenkin eri mieltä
3 = Ei samaa eikä eri mieltä
4 = Jokseenkin samaa mieltä
5 = Täysin samaa mieltä

*** Käytettävyydestäustustunnon päiväys**

Muotoilu: dd.mm.yyyy

? Tätä kenttää käytetään tulosten seurantaan ja validointiin.

Järjestelmän käytettävyys.

	1	2	3	4	5	Ei vastausta
Järjestelmä on hyödyllinen	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Järjestelmä on yksinkertainen	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Järjestelmä on helppokäyttöinen	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Järjestelmää käyttäessä ei tarvitse teknistä tukea	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Järjestelmän eri toiminnot sopivat hyvin yhteen	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Järjestelmän ulkoasu on yhtenäinen	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Järjestelmän käytön opettelu on erittäin nopeaa	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Järjestelmän käyttö on vaivatonta ja sujuvaa	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Järjestelmän käyttäminen on miellyttävää	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Järjestelmän sujuva ja tehokaskaan käyttö ei vaadi harjoittelua	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Avoin palaute koskien järjestelmää

? Tuliko järjestelmää koskien jotain vielä mieleen? Tähän kenttään voit jättää halutessasi anonyymiä palautetta tai kommentteja!

Avoin palaute koskien koejärjestelyitä

? Koejärjestelyitä ja mm. kysymyksiä kehitetään palautteen perusteella! Tähän kenttään voit antaa anonyymisti palautetta jonka avulla pyrimme parantamaan käytettävyydestäustustusprosessiamme.

Rekisteriseloste löytyy osoitteesta <https://dl.dropboxusercontent.com/u/18835906/rekisteriseloste/rekisteriseloste.pdf>

LIITE 5 KÄYTETTÄVYYSTESTAUKSEN ALOITUS- JA LOPETUSOHJEET

Tämä dokumentti kuvaa sellaisen käytettävyytestaussionsession kulun, joka sisältää ääneen ajattelua, mutta sopii pienin muutoksin myös esimerkiksi korttien lajittelun järjestämistä varten.

Käytettävyytestauksen valmistelu:

- Varataan tila testausta varten.

Juuri ennen käytettävyytestaussionsiota:

- Valmistellaan työasema.
 - Varmistetaan että tarvittavat yhteydet ja näytönkaappaus toimivat.
 - Hankitaan ulkoinen hiiri ja mahdollisesti näppäimistö.
 - Säädetään testattava nettisivu kotisivuksi kaikille selaimille.
 - Lisätään työpöydälle (tai muualle) pikakuvakkeet mahdolliseen korttien lajitteluun ja loppukyselyyn.
- Varataan juotavaa koehenkilölle (ja mahdollisesti itselle).
- Varataan muistiinpanovälineet ja pöytäkirja tai testaussuunnitelma.
- Varmistetaan että koehenkilön palkkio on saatavilla.
- Tyhjennetään käytettyjen selainten välimuistit.

Käytettävyytestaussionsession aluksi:

- Allekirjoitetaan salassapitosopimus.
- Mainitaan seuraavat asiat:
 - Testissä testataan järjestelmää, ei käyttäjää.
 - Testattava järjestelmä on keskeneräinen.
 - Testattava järjestelmä ei ole testaajan tekemä, joten hänellä ei ole henkilökohtaista sidettä järjestelmään (paitsi jos testaaja itse asiassa on järjestelmän kehittäjä, jolloin asiasta ei kannattane juuri puhua).
 - Muistutus, että testiin osallistuminen on vapaaehtoista ja testaus voidaan lopettaa milloin vain.
 - Selostaa mahdollisesti tehtävä videotalliointi ja sen käyttötarkoitus.
 - Vakuuttaa, että kerättävä data tullaan pitämään salassa.
 - Mainita siitä, että kaikenlaiset kysymykset ovat tervetulleita ja arvokasta dataa, mutta että niihin ei välttämättä voida aina vastata testin aikana.

- Pyyntö kysyä tarvittaessa tarkentavia kysymyksiä ennen testauksen aloittamista.
- Selostetaan ”yksinkertaistetun ääneen ajattelun” kulku.
- Suoritetaan korttien lajittelu

Käytettävyytestaussionsession loppuksi:

- Pyydetään täyttämään loppukysely.
 - Jos loppukyselyyn liittyy vaatimuksia esim. näkökulmasta, muista mainita!
- Annetaan palkinto testiin osallistumisesta.
- Annetaan koehenkilölle käyntikortti, jotta osaa ottaa myöhemmin yhteyttä halutessaan.

LIITE 6 PROJEKTIRYHMIEN KOMMENTIT KÄYTETTÄVYYSTESTAUKSEN TULOKSISTA JA RAPORTEISTA

Projekti ja iteraatio:	RETU 1
Raporttiversio:	1
Havaintojen määrä ja hyödyntäminen:	<p>33 käytettävyystestauksen havaintoa, joista 6 bugia, 8 lisäominaisuustoivetta.</p> <p>14 havaintoa kirjattiin uusina löydöksinä listalle, jäljellejäävistä 5 oli jo listalla, osa katsottiin duplikaateiksi (joh-tuen taustalla olevasta toiminnallisuudesta) ja osa aiheet-tomiksi ja loput merkitykseltään niin pieniksi ettei niitä tässä vaiheessa otettu työlistalle.</p>
Kommentit:	<p>Projektipäällikkö S:n kommentit:</p> <ul style="list-style-type: none"> • ”Yleiskommenttina olisi hyvä jos skenaariot, jossa ongelmia tapahtuu, olisi tarkemmin listattuna. Sil-loin on helpompi tehdä kehitysehdotus.” <p>Ensimmäisen testausiteraation aikana suuri painoarvo oli tulosten raportointitavan kehittämisellä</p>

Projekti ja iteraatio:	PA 1
Raporttiversio(t):	1 2 (kehitetty testaaaja M:n ja projektipäällikkö J:n ensimmäi-sestä raportista antaman palautteen perusteella)
Havaintojen määrä ja hyödyntäminen:	<p>15 käytettävyystestauksen havaintoa, joista 1 bugi, 2 lisäominaisuustoivetta.</p> <p>4 havaintoa kirjattiin työlistalle uusina löydöksinä, 7 oli jo jossain muodossa listalla tai työn alla, 1 löydös toimitettiin asiakkaalle arvioitavaksi, 1 hylättiin ja 2 havaintoa arvioi-tiin jatkokehitystarpeiksi.</p>
Kommentit:	<p>Testaaaja M:n kommentit:</p> <ul style="list-style-type: none"> • Tuloksia voisi raapustaa myös näkymä-kohtaisesti koska niitä on järjestelmässä kuitenkin melko vä-hän ja on siten projektin/testauksen kannalta hel-pommin jäsennettävissä kehitykseen • Lopun yhteenvedoon olisi mukava saada hieman tarkemmin kuvattua (myös vaikkapa käppyy-röitä/numeroita havainnollistaen) miten käyttäjät

haluavat järjestelmän käytettävyyttä parannettavan esim. mitkä asiat ei tee järjestelmästä yksinkertaista (keho arvosana) vaikka estetiikassa sanotaan että yleisilme on yksinkertainen ja selkeä.

Projektipäällikkö J:n kommentit:

- Kokonaisuutena oikein hyvä rapsa. Tästä rapsamallista saa samalla kertaa management-tason summaryä että riittävän konkreettista pohjaa havaintojen läpikäynnille projektitiimin ja kenties myös asiakkaan kanssa.
- Havaintojen jaottelu: tuo Nielsen on hyvä ja yleisesti tunnistettu tapa jäsenellä käytettävyyssasi-oita, joten se on oletusarvoisena ihan paikallaan. Tykkään omassa management-kohderyhmässäni siitä jaottelusta. Projekteja varmaan monissa tapauksissa käytännössä helpottaa, jos havaintojen jaottelua tehdään tapauskohtaisesti muillakin tavoilla, esim. näkymäpohjainen lähestymistapa, tai kenties jopa suoraan projektiin backlogin storyihin määpäten (tämä riippuisi toki hyvin paljon projektin backlogin tasosta ja esitystavasta, ja vastavuoroisesti myös projektin backlogin teossa pitäisi jo alusta asti sitten huomioida tuo Nielsen tai muu käytettävyydestauksen mukainen jaottelu (mahdollisuuksien mukaan)). Pitkässä juoksussa yleinen jaottelumalli (esim. Nielsen) on varmaan kokonaistehokkain ja sen kautta saa yhteismitallisempaa dataa yli projektien. Noilla optionaalisilla jaottelutavoilla mallia voisi kuitenkin hyvin tuotteistaa ja käytettävyydestausta vaikka skaalata mukavasti erilaisten projektien erilaisiin tarpeisiin ja erilaisiin budjetteihin. Eli tyyliin oletuksena Nielsen, mutta jos haluatte sijoittaa enemmän niin jaotellaan myös näkymä- tai storypohjaisesti.
- Havaintojen priorisointi: Just a thought. Olisiko mahdollista saada raporttiin hiukan selkeämmin näkymiin havaintojen numeerisia tai muulla mittarilla mitattavia painoarvoja? Ainakin häntä- ja kärkipään havainnoista, helpottaisi löytämään absoluuttisesti häiritsevimmät pullonkaulat ja sudenkuopat ja ärsytyksen kohteet testajien mielestä (ja

	<p>vastaavasti myös parhaiten toimivat asiat)? Tyyliin top x havainnot, jotka ärsyttää / innostaa eniten. Tässä oleellista olisi myös, kuinka moni testaaja kokee saman asian ärsyttäväksi / kivaksi.</p> <ul style="list-style-type: none"> • Mittarit: Nuo mittarit olisi varmaan hyvä listata johonkin näkyviin. Parhaimmat ja huonoimmat tulokset tuottavat mittarit tuossa onkin kuvattuna, se on hyvä. Käytännössä voisi myös olla hyvä käydä mittareista pientä keskustelua projektin kanssa ennen käytettävyydestestauksen suorittamista. • Statiistikat, johdannot, yhteenvedot: Käytettävyydestestauksen yleistä suorittamista ja ajatusmaailman takana olisi hyvä avata vähän raportin alussa. Johdannossa voisi hiukan kuvata testauksen strategian pääpiirteitä, esim. minkälaisia käyttäjäryhmiä on tunnistettu, kuinka monta testaajaa mistäkin käyttäjäryhmästä osallistui testaukseen, millä tavalla testaus käytännössä suoritettiin jne. Vastaavasti mittareita, ja statiistikkoja voisi vähän lisätä / tuoda niitä raportissa enemmän esille, vaikka omana loppuyhteenvetonaan, liitteenä tai muulla tavalla. • Kaiken kaikkiaan hyviä havaintoja, ilman muuta kannattaa ottaa toinen kuluttajakäyttäjien sessio.
--	--

Projektin ja iteraatio:	PA 1 (liitännäisprojekti)
Raporttiversio:	1
Havaintojen määrä ja hyödyntäminen:	21 havaintoa, joista 3 bugia. Havainnot toimitettu asiakkaalle.
Kommentit:	Projektipäällikkö JM: <ul style="list-style-type: none"> • Havainnot toimitettiin "supistettuna" asiakkaalle.

Projektin ja iteraatio:	PA 2 (näkyvä AV)
Raporttiversio:	3

Havaintojen määrä ja hyödyntäminen:	21 havaintoa, joista 3 bugia ja 1 jatkokehitysidea. 9 suoraan työlistalle (joista 1 bugi), 4 viedään asiakkaan arvioitavaksi, 4 lisätään jatkokehityslistalle (joista 1 bugi), 1 havainto on validi mutta hylätään ja 3 hylätään aiheettomina tai määrittelyyn vastaisina (joista 1 bugi).
Kommentit:	Ei kommentteja tähän liittyen.

Projekti ja iteratio:	PA 2 (näkyvä YP)
Raporttiversio:	3
Havaintojen määrä ja hyödyntäminen:	12 havaintoa. 11 suoraan työlistalle, 1 hylättiin määrittelyjen vastaisena.
Kommentit:	Projektipäällikkö J: <ul style="list-style-type: none"> • Aika paljon näkyy, että UI:ta ei ole konseptoitu ja siihen liittyvät ohjeistukset on tulkinnanvaraiset. Toki itse toteutuskin on vielä keskeneräinen, varsinkin hallintanäkymän osalta. • Mutta hyviä havaintoja kaiken kaikkiaan.

Projekti ja iteratio:	SH 1
Raporttiversio:	3
Havaintojen määrä ja hyödyntäminen:	28 havaintoa, joista 2 bugeja. Havainnoista 18 (sisältäen 1 bugin) uutena listalle, 4 vanhaa havaintoa, 1 jatkokehitykseen, 1 hylätään pienimerkityksisenä ja 4 (sisältäen 1 bugin) ei valideja
Kommentit:	Ei kommentteja.

Projekti ja iteratio:	HE
Raporttiversio:	2.1 (muokattu raportin versio 2 kognitiivisen läpikäynnin ja heuristisen evaluoinnin käyttöön sopivaksi)
Havaintojen määrä ja hyödyntäminen:	Kognitiivisen läpikäynnin tuloksena 27 havaintoa, joista 1 bugi. Heuristisen evaluoinnin tuloksena 31 havaintoa, joista 1 bugi.

	<p>Yhteensä 58 havaintoa joista 2 bugeja.</p> <p>Näistä 48 uusina työlistalle (sisältäen 2 bugia), 6 jatkokehitykseen, 3 hylätään pienimerkityksisinä ja 1 määrittelyn vastaisena.</p>
Huomiot:	Järjestelmä oli tuotantokäytössä.
Kommentit:	Ei erikseen kommentteja.

Projekti ja iteratio:	RETU 2
Raporttiversio:	3
Havaintojen määrä ja hyödyntäminen:	<p>46 havaintoa, joista 5 bugia.</p> <p>11 havaintoa jatkokehitykseen uusina havaintoina, 13 jatkokehitykseen jo listalla olevina, 19 hylättiin vähämerkityksisinä tai kehitysresurssien puutteessa ja 3 hylättiin määrittelyn vastaisina (esimerkiksi ongelma joka on ilmennyt ei-tuetulla selaimella kuten Safarilla).</p>
Kommentit:	<p>Projektipäälliköltä:</p> <ul style="list-style-type: none"> • Yleisesti ottaen Excel oli selkeämpi kuin aiemmin. • Löydöksistä moni liittyi näyttöihin/toimintoihin jotka on tietoisesti jätetty alhaiselle prioriteetille niiden vähäisen käytön/vähäisen käyttäjämäärän vuoksi. Eli löydökset ovat kyllä valideja mutta osalle ei aiota silti tehdä mitään koska muutoksille ei ole riittävää liiketoiminnallista syytä. <p>Käytettävyyssiantuntijalta:</p> <ul style="list-style-type: none"> • Hyviä löydöksiä. Tämä oli siitä hyvä testi, että se osoittaa MIKSI käyttäjätutkimus pitää tehdä HYVISSÄ AJOIN ennen tuotteen/palvelun lopullista toimitusta (eli vähintään 4 sprinttiä ennen viimeistä sprinttiä, joka yleensä käytetään viimeistelyyn. Viimeisen sprintin aikana ei ole enää mahdollisuutta tehdä käytettävyyssparannuksia. • Käyttäjätutkimuksissa nousee lähes poikkeuksetta esiin tärkeitä parannuksia, joiden korjaamiseen

	PRODUCT OWNERIN TÄYTYY OSATA VARATA resursseja vielä ennen palvelun tai tuotteen lopullista lanseeraamista.
--	---

Projekti ja iteraatio:	PA 3
Raporttiversio:	4
Havaintojen määrä ja hyödyntäminen:	9 havaintoa, joista 1 bugi. Havainnoista 2 otetaan työn alle, 4 jatkokehitykseen, 3 havaintoa ei tehdä koska määrittelyjen vastaisia tai toteutetaan toisessa projektissa.
Kommentit:	<p>Testaaja M:</p> <ul style="list-style-type: none"> • Yleisesti, tuo esitystapa on mun mielestä hyvä että esitetään ensin suorat ja yksiselitteiset mittaustulokset jollain havainnollisella graafisella tavalla ja niissä vertauskelpoisia lukemia esim. juuri tuo SUS ja lopulta suorat parannusehdotukset kohta kohdalta. • Testihenkilöiden otanta olisi voinut olla isompikin mutta ymmärrän aika- ja resurssirajoitteet. • Hyvää projektin kannalta näissä kuvaajissa on että nähdään jonkinlainen progressio, ilmeisesti on menty projektin käytettävyydessä parempaan suuntaan tällä kertaa. • Käytettävyysomittarit-kuvaaja on toisaalta melkoinen viivaviidakko mutta tarkemmalla syynillä siitä saa kyllä oleellisen osumaan silmään, ei kai tuota kannata jakaa useampaankaan kuvaajaan. • Projektin kannalta grafiikka ja mittarit on konkreettisia projektin kehityksen kannalta, käytettävyysomittareista voi poimia suoraan heikoimmat paunat saaneet kohdat ja kohdistaa mahdollisesti jatkokehitystä ja ideointia niihin • Havainnot-osio on hyvin raportoitu ja kehitysehdotukset implementoitavia • Lukijana tarvitsisin enemmän metatietoa kuvaajista ja tuloksista esim. missä ympäristössä on toimittu milläkin testikierroksella. <ul style="list-style-type: none"> ○ Eri kierroksilla käytettyjen järjestelmän versioiden ulkonäkö olisi mukava nähdä kuvankaappauksin että voisi arvata mitkä

	<p>asiat ovat huonontaneet/parantaneet käytettävyyttä mutta menee jo ehkä liian hieromisen puolelle.</p> <ul style="list-style-type: none"> ○ Onko testihenkilöt samoja kierroksittain eli onko testikierrosten vertailut keskenään täysin valideja?
--	---

Projekti ja iteraatio:	PA 3 (liitännäisprojekti)
Raporttiversio:	4
Havaintojen määrä ja hyödyntäminen:	12 havaintoa, joista 2 bugia. Havainnot toimitettu asiakkaalle.
Kommentit:	<p>Oma kommentti, vertailu ulkopuolisen yhtiön testausraporttiin:</p> <ul style="list-style-type: none"> • Yhtenevää: <ul style="list-style-type: none"> ○ Havainto siitä, että sisällön löytäminen nettisivuilta on navigaation osittaisen päällekkäisyyden vuoksi hankalaa. ○ Osa tekstistä sivuilla on vaikeaselkoista (erityisesti valkoinen teksti kuvan päällä). ○ Epäselvyys eri pankkien sivuista ja yleisistä sivuista. ○ Nettisivujen ei arvioitu palvelevan asiakkaita hirveän hyvin (myyntihenkisyys sen sijaan oli selvää). ○ Näkyvä valikko kaikista pankeista helpotaisi, joskaan se ei tullut käytettävyydestä tulkittua sellaisenaan ilmi. • Eriävää: <ul style="list-style-type: none"> ○ Raportissa huomio pitkälti verkkopankin ja nettisivujen eroissa <p>Projektipäällikkö JM:</p> <ul style="list-style-type: none"> • Havainnot toimitettiin "supistettuna" asiakkaalle.

LIITE 7 KÄYTETTÄVYYSTESTAUSRAPORTIN VERSIOT

Tässä liitteessä esitellään yleisellä tasolla käytettävyystestauksen tulosten raportointiin käytettävien dokumenttien piirteet.

Versio 1

Ensimmäinen versio raportista on word-dokumentti, joka on tehty Samlinkin dokumenttipohjaan. Havainnot on liitetty taulukkoon jossa ne on jaoteltu Nielsenin heuristiikkojen perusteella. Alla kuvakaappaus.

YLEISTÄ

1 HUOMIOT

Huomiot on järjestetty Nielsenin 10 heuristiikan mukaan.

Puhun käyttäjistä usein yksikössä, vaikka ilmiö olisi tullut ilmi useamman käyttäjän kohdalla. Tämä lista on jo suodatettu siten, että mukana on vain olennaisimmat huomiot. Otanta ei ollut tilastollisesti merkittävä, havainnot ovat anekdoottisia.

Järjestelmän tilan näkyvyys	
Järjestelmän ja reaalimaailman vastaavuus	
Käyttäjän kontrolli ja vapaus	
Yhtenäisyys ja standardointi	
Virheiden estäminen	
Muistikuorman vähentäminen ("Tunnistaminen muistamisen sijasta")	
Käytön joustavuus ja tehokkuus	
Suunnittelun estetiikka ja minimalistisuus	
Käyttäjien tukeminen virheiden tunnistamisessa ja niistä palautumisessa	
Käyttäjän apu ja tuki, sekä dokumentaatio	

1.1.1 Bugit tai epäloogisuudet

1.1.2 Lisäominaisuustoiveet

1.1.3 Kvantitatiivista dataa

Versio 2

Raportin versiossa 2 huomioitiin palaute RETU- ja PA-projektien 1. iteraation käytettävyydestä huomioiden raportointitavasta. Se sisältää word-dokumentin, joka sisältää kuvaukset bugeista, lisäominaisuustoiveista ja erityisesti kvantitatiivista dataa SUS-menetelmää mukaillen. Tämä word-tiedosto ja Excel-tiedosto, jossa ovat varsinaiset käytettävyydestä havainnot, toimitetaan sähköpostin liitteinä projektiryhmälle.

1 HAVAINNOT

1.1 Käytettävyydestä havainnot

Erillisessä excel-tiedostossa.

1.1.1 Bugit tai epäloogisuudet

1.1.2 Lisäominaisuustoiveet

1.1.3 Kvantitatiivista dataa

Parhaat arvosanat mittareista:

Kehnoimmat arvosanat mittareista:

|

Excelin rakenne ja heuristiikkojen ja kriittisyyden valinnat näkyvät seuraavissa kuvakaappauksissa.

Käytettävyydestäraportti						
Projekti						
Näköma, d jolla	Ongelman nimi	Ongelman kuvaus	Kategoria	Heuristiikka, jota rikotaa	Ratkaisuehdote	Prioriteetti
1	Etusivu	Etusivu koostuu näköinen	UI	8. Suunnittelun esteettikilla ja minimalistisuus	Etusivulta voisi poistaa osan välikkyistä ja häiritsevistä elementeistä.	Critical

Heuristiikat:

1	Järjestelmän tilan näkyvyys
2	Järjestelmän ja reaali maailman vastaavuus
3	Käyttäjän kontrolli ja vapaus
4	Yhtenäisyys ja standardointi
5	Virheiden estäminen
6	Muistikuorman vähentäminen ("Tunnistaminen muistamisen sijasta")
7	Käytön joustavuus ja tehokkuus
8	Suunnittelun estetiikka ja minimalistisuus
9	Käyttäjien tukeminen virheiden tunnistamisessa ja niistä palautumisessa
10	Käyttäjän apu ja tuki, sekä dokumentaatio

Kriittisyysluokitus tässä vaiheessa:

E	F	G
		Prioriteetti
1	Show stopper	Suoritus keskeytyy aina, järjestelmä on käyttökelvoton
2	Critical	On suuri riski, että suoritus keskeytyy tai päättyy virheeseen
3	Medium	Ongelma, mutta ei tule normaalitilanteessa todennäköisesti keskeyttämään suoritusta
4	Minor	Hidaste, pieni este, hämmentäjä, ei tule normaalitilanteessa ikinä keskeyttämään suoritusta.
5	Cosmetic / suggestion	Kosmeettinen ongelma tai merkitykseltään pieni parannusehdotus.

Versio 3

Tämä käytettävyydestäusraportin versio kehitettiin taas edellisten palautteen perusteella. Raportin Excel-osioon otettiin mukaan tieto siitä, monellako koehenkilöllä ongelma ilmeni: tämä helpotti tulosten kirjausta ja toisaalta sen arviointia, kuinka vakava ongelma on kyseessä.

Käytettävyydestäusraportti										
Projekti X										
Näkymä, jolla ongelma ilmeni										
id	ongelman nimi	ongelman kuvaus	Kategoria	Heuristiikka, jota rikotaan	Ratkaisuehdotus	Prioriteetti	Ilmeni käyttäjillä			
				8. Suunnittelun esteettisyys ja minimaalisuus			1	2	3	4
1	Etusivu koottisen näköinen	Etusivulla on paljon elementtejä. Käyttäjä ei tiennyt mistä painaa, vaan sulki selaimen tutustumatta sivuun enempäs.	UI		Etusivulta voisi poistaa osan välikkyistä ja häiritsevistä elementeistä.	2 Critical	x	x	x	

Word-tiedostoa ei tässä vaiheessa enää sellaisenaan lähetetä mihinkään, vaan se toimii pohjana projektiryhmälle lähetettävän sähköpostin sisällölle. Bugit raportoidaan word-dokumentissa.

Versio 4

Versio 4 on viimeisin projekteissa käytetty versio raporttimalleista. Word-dokumentin otsakkeet on muutettu paremmin sähköpostiin kopioituviksi, ja itse word-dokumentti ei enää sisällä muuta tietoa kuin kuvauksen kvantitatiivisesta datasta. Bugit on siirretty Excel-tiedostoon kategorialla "bugi" (tiedosto myös sisältää bugien lukumäärän laskenta-automatiikan) ja lisäominaisuuspyynnöt ovat normaaleita havaintoja prioriteetilla 7.

Käytettävyydestauksen havainnot

Havainnot ovat erillisessä [excel-tiedostossa](#).

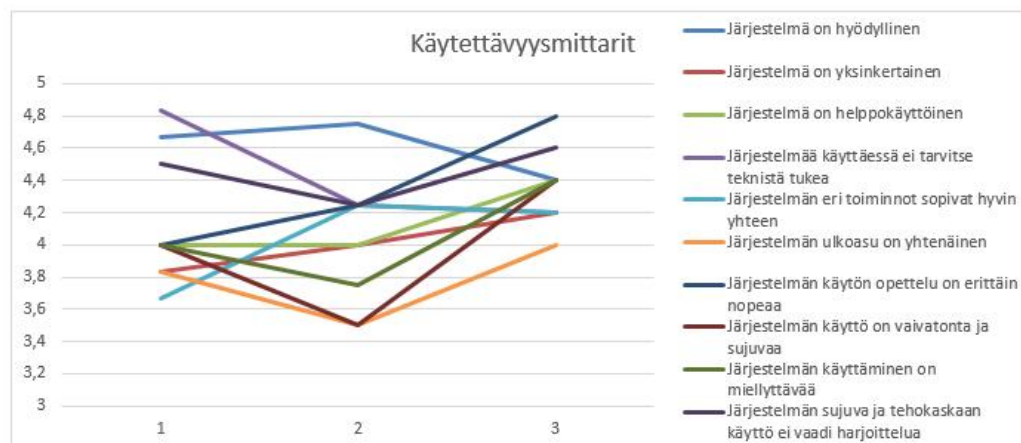
Bugit tai epäloogisuudet

Löytyvät [excelistä](#) kategorialla "bugi".

Lisäominaisuustoiveet

Löytyvät [excelistä](#) prioriteetilla 7.

Kvantitatiivista dataa



Excel-dokumentin prioriteettiluokitus on muutettu 7-portaiseksi ja jokaiselle havainnolle on lisätty kentät liiketoiminnallisen merkityksen ja kommentin lisäämiseksi – nämä ovat projektipäällikön/ muun projektiryhmän edustajan käyttöä varten.

Käytettävyyssitestausraportti		Projekti:	Bugeja yht: 0	Ilmeni käyttö:						
Näkymä, jolla ongelma ilmeni	Ongelman nimi	Ongelman kuvaus	Kategoria	Heuristiikka, jota rikotaan	Ratkaisuehdotus	Prioriteetti	Yks	Liiketoiminnallisen merkitys	Liiketoiminnan kommentit	
1 Etusivu	Etusivu kaotettiin näkökentän	Etusivulla on paljon elementtejä. Käyttäjää ei tiennyt mistä painaa, vaan sulki sovelimen tutustumatta sivuun enempää.	UI	8. Suunnittelun esteettöitä ja minimalistisuus	Etusivulta voisi poistaa osan välikäytöstä ja hääritsevistä elementteistä	2 Critical	xxxxx	5	2 Esitetään asiakkaalle	Esitetään asiakkaalle, että uimahallin osasta flash- ja sivedit-elementtejä. Tähän tarkuuteen kuitenkin asiakkaan hyväksyntä.

Alla kuvakaappaus kriittisyysluokituksesta.

Prioriteetti	
1 Show stopper	Suoritus keskeytyy aina, järjestelmä on käyttökelvoton
2 Critical	Suoritus keskeytyy virheeseen ainakin tietyillä arvoilla, workaround vaatii käyttäjältä paljon työtä.
3 Major	Suoritus voi keskeytyä, mutta ongelma on kierrettävissä vähäisellä työmäärällä.
4 Normal	Ongelma joka vaikeuttaa suoritusta, mutta ei tule normaalitilanteessa todennäköisesti keskeyttämään suoritusta.
5 Minor	Hidaste, pieni este, hämmentäjä, ei tule normaalitilanteessa ikinä keskeyttämään suoritusta.
6 Cosmetic / trivial	Kosmeettinen ongelma tai pieni hämmentäjä, ei vaaranna suoritusta.
7 Suggestion	Parannusehdotus.

Versio 5

Versio 5 on viimeisten tutkielmaan kuuluneiden käytettävyystestauskertojen palautteen perusteella kehitetty versio raportista. Excel-tiedosto on sama kuin versiossa 4, mutta word-muotoiseen sähköpostipohjaan on lisätty historiatieto eri ympäristöistä ja konfiguraatioista, joita testauksessa on käytetty. Tämä tieto täydentää erityisesti SUS-arvoja, jotka useampia käytettävyystestauskertoja käsittävissä projekteissa sisältää jo paljon historiatietoakin joka kuvaa arvojen kehitystä.

Käytettävyystestauksen havainnot

Havainnot ovat erillisessä [excel-tiedostossa](#).

Bugit tai epäloogisuudet

Löytyvät [excelistä](#) kategorialla "bugi".

Lisäominaisuustoiveet

Löytyvät [excelistä](#) prioriteetilla 7.

Kvantitatiivista dataa

[Käytettävyystestauksessa käytetyt ympäristöt ja konfiguraatiot:](#)

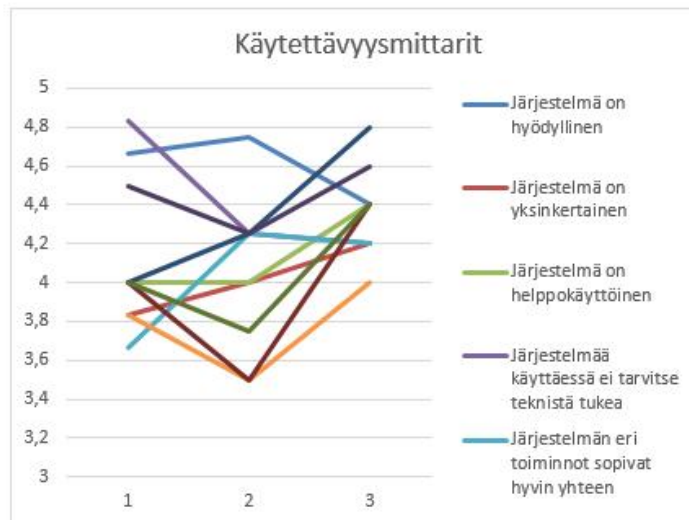
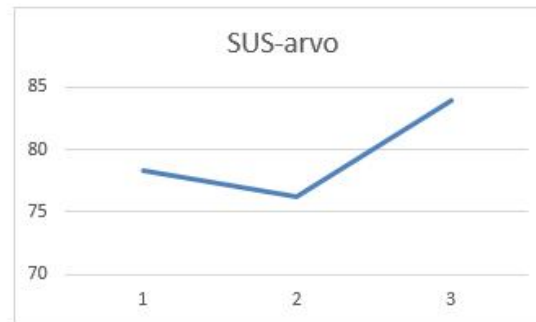
Testauskerta	Konfiguraatio
1	lorem ipsum
2	dolor sit amet
3	etc etc

[Samlinkin käyttöön muokattu SUS-arvo 84 \(ed. 76,25\).](#)

Otanta oli 5 kuluttaja-asiakasta simuloivaa koehenkilöä.



[Arvosanat asteikolla 1-5.](#)

|



LIITE 8 KÄYTETTÄVYYSTESTAUKSEN LOPPUKYSELY

Tämä kysely lähetettiin käytettävyystestaukseen osallistuneiden projektiryhmien jäsenille.

Käytettävyystestausprojektin loppupalautekysely

Kyselyssä on yksi pakollinen kysymys (ja se on kyselyn ensimmäisenä), muiden rooli on tarjota lisä- tai taustatietoja.

Palaute

1 Tuottiko käytettävyystestaus hyödyntämiskelpoisia tuloksia?
Valitse sopiva vaihtoehto

Kyllä, paljon tai pääosin

Kyllä, jossain määrin tai jonkin verran

Ei

Kirjoita kommentisi tänne:

2 Pääasiallinen roolisi projektiryhmässä?
Valitse sopiva vaihtoehto

Kehittäjä tai (tekninen) arkkitehti

Testaaja

Projektipäällikkö / Product Owner

Scrum Master

Käytettävyyssiantuntija

Muu (täydennä kommenttikenttään!)

Ei vastausta

Kirjoita kommentisi tänne:

3 Projekti(t) jossa olit mukana jossakin roolissa. Voit halutessasi tarkentaa vastausta esimerkiksi roolin osalta.
Valitse kaikki sopivat vaihtoehdot

RETU

Pankkien Ajanvaraus

SP.n

Sähkökkä

Raisio

4 Oliko käytettävyystestauksen havaintojen hyödyntäminen helppoa? Mitä siinä suhteessa oltaisiin voitu tehdä vielä paremmin?
Valitse sopiva vaihtoehto

Oli erittäin helppoa ja suoraviivaista

Oli mutta ei kovin helppoa

Ei

Ei vastausta

Kirjoita kommentisi tänne:

5 Mikä testauksessa toimi hyvin?

6 Mikä olisi voinut toimia paremmin, kehitysehdotuksia?

7 Sähköpostiosoite josta voi kysellä lisätietoja

? Anna tämä osoite vain jos sinulta saa kysellä lisätietoja. Vastaukset ovat anonyymejä, joten ilman tätä tietoa en tiedä keitä kysellä jos jotain jää auki.

[Lähetä](#)

[Poistu kyselystä ja tyhjennä vastaukset](#)

[Jatka myöhemmin](#)

Rekisteriseloste löytyy osoitteesta <https://dl.dropboxusercontent.com/u/18835906/rekisteriseloste/rekisteriseloste.pdf>

LIITE 9 SOPIMUS KÄYTETTÄVYYSTESTAUKSEN TIETOJEN KÄYTÖSTÄ

Osapuolet

Tämä salassapitosopimus ("sopimus") on laadittu seuraavien osapuolten välille:
Käytettävyystestaussessioiden järjestäjäorganisaatio, Yhtiö Oy:n edustajana,

_____. Jäljempänä
"Yhtiö".

Käytettävyystestaussession osanottaja,

_____. Jäljempänä
"osanottaja".

Sopimuksen tarkoitus

Käytettävyystestaussession aikana osanottaja saa tietoja projekteista ja järjestelmistä, joita Yhtiö toimittaa tai toteuttaa asiakkailleen. Nämä tiedot ovat yleensä salassa pidettävää materiaalia, ja tämän sopimuksen tarkoituksena on estää näiden tietojen päätyminen kolmansille tahoille. Sopimuksen allekirjoittamisen seurauksena osanottajaa koskevat tietojen salassapidon suhteen pääpiirteissään samat vaatimukset kuin Yhtiön työntekijöitä. Myöskin Yhtiöllä on oikeus käyttää käytettävyystestauksesta kertyviä tietoja vain rajatusti.

Salassapitovelvollisuus

Osanottaja sitoutuu pitämään luottamuksellisen tiedon salassa sekä olemaan ilmaise-matta saamaansa luottamuksellista tietoa kolmansille tahoille. Osanottaja sitoutuu olemaan käyttämättä haltuunsa saamaansa luottamuksellista tietoa missään muussa tar-koituksessa kuin kohdan 3.2 mukaisen käyttötarkoituksen täyttämiseen. Yhtiö käyttää saamiaan tietoja vain käytettävyystestauksen kannalta tarkoituksenmukaisiin käyttötar-koituksiin.

Luottamuksellisen tiedon määrittely ja yksilöinti

"Luottamuksellinen tieto" sisältää käytettävyystestaustunnon tehtävien suorittamiseen vaadittavat materiaalit, jotka jaetaan session aikana, projektien aihe, asiakkaat, mah-dolliset henkilötiedot ja muut järjestelmissä säilytettävä tai käsitettävä data sekä järjes-telmien käyttötarkoitus ja ulkoasu. Käytettävyystestaussession kulku ja käytettävät me-netelmät ovat salassa pidettävää tietoa.

Sopimuksen mukainen salassa pidettävän tiedon käyttötarkoitus

Osanottaja saa käyttää saamiaan tietoja käytettävyystestaussession aikana testin teh-tävien suorittamiseen. Osanottajalla ei ole lupaa jakaa käytettävyystestaussession ai-kana saamiaan projekteja, järjestelmiä tai Yhtiön asiakkaita koskevia tietoja ulkopuoli-sille.

Yhtiö saa käytettävyytestaussession aikana tietoja liittyen testattavien järjestelmien laatuun (erityisesti käytettävyyteen). Nämä tiedot ovat Yhtiön käytettävissä myöhemmin projektien kannalta tarkoituksenmukaisella tavalla. Henkilötietojen säilytys on kuvattu käytettävyytestauksen koehenkilötietokannan rekisteriselosteessa²⁸. Yhtiön käyttöön tulevia tietoja ei luovuteta ulkopuolisille. Videotallenteista ei luovuteta edes lyhyitä otteita konsernin ulkopuolelle ilman osanottajan lupaa.

Yhtiön keräämiä tietoja voidaan osanottajia yksilöimättä käyttää myös tieteellisiin tarkoituksiin.

Muut ehdot

Salassapitoaika

Tämän sopimuksen mukaiset salassapitovelvoitteet säilyvät voimassa viisi (5) vuotta luottamuksellisen tiedon luovutuspäivästä lukien.

Vahingonkorvaus

Sopimusta rikkonut osapuoli on velvollinen korvaamaan loukatulle osapuolelle täysimääräisesti sopimusrikkomuksesta aiheutuneen vahingon määrän kattaen myös välilliset vahingot.

Sopimuksen voimassaolo

Sopimus tulee voimaan molempien osapuolten allekirjoitettua sen. Sopimus koskee ainoastaan käytettävyytestaussessioiden liittyvää osanottajan ja Yhtiön välistä kanssakäymistä ja ko. kanssakäymisen aikana luovutettuja tietoja.

Velvollisuuksien siirto

Tätä sopimusta koskevat muutokset on tehtävä kirjallisesti osapuolten allekirjoituksin vahvennettuna. Tätä sopimusta ei saa siirtää kolmannelle osapuolelle ilman toisen osapuolen etukäteistä kirjallista suostumusta.

Sovellettava laki ja riitojen ratkaiseminen

Tähän sopimukseen sovelletaan Suomen lakia. Tästä sopimuksesta ja sen tulkinnasta johtuvat riitaisuudet pyritään ratkaisemaan ensisijaisesti neuvottelemalla.

Allekirjoitukset

Yhtiö Oy:n edustaja

(pvm ja paikka)

(nimenselvennös)

Käytettävyytestaussession osanottaja

(pvm ja paikka)

(nimenselvennös)

²⁸ <https://dl.dropboxusercontent.com/u/18835906/rekisteriseloste/rekisteriseloste.pdf>