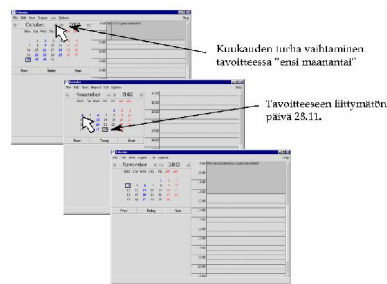


Käyttöliittymien arviointimenetelmät



Kälin arviointimenetelmiä Käyttäjien kanssa ja ilman

Arviointi loppukäyttäjien avulla (user testing)

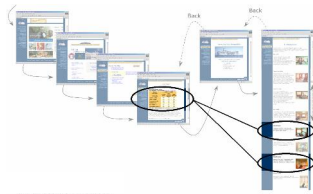
- n käytettävyystestaus (usability testing)
- n käytettävyysläpikäynti (usability walkthrough)
- n *hyödyllisyysläpikäynti* (utility walkthrough)
- n käyttäjätarkkailut (user observations)
- n käyttölokien keruu (software logging)
- n [fokusrhmäkeskustelut (focus groups)]

Arviointi ilman käyttäjiä (inspection, expert review)

- n kognitiivinen läpikäynti (cognitive walkthrough)
- n *simulointitestaus* (simulation-based expert review, usage simulation)
- n heuristinen arviointi (heuristic evaluation)
- n heuristinen läpikäynti (heuristic walkthrough)
- n käyttäjän tehtävien ja toimenpiteiden mallinnus (GOMS, KLM)
- n [katselemointitilaisuudet (consistency inspection, standards inspection)]

Copyright © 2006 / Sari A. Laakso, Antti Latva-Koivisto

Simulointitestaus: menetelmä



Simulointitestauksessa [Laakso02] asiantuntija selvittää testitapauksen suorittamiseen tarvittavan oikean käyttösekvenssin. Sen jälkeen hän paikantaa ongelmakohdat simuloimalla käyttösekvenssin suorittamista useaan kertaan ja tarkastelemalla toimenpiteitä ja ajatustyötä, joiden suorittamista käyttöliittymä käyttäjältä vaatii.

Simulointitestaus Menetelmän vaiheet

- n Lähtökohdaksi arvioija selvittää jonkin tosielämässä käyttäjälle eteen tulevan konkreettisen käyttötilanteen
- n Arvioija arvioi ensin, mikä olisi valitussa käyttötilanteessa käyttäjälle paras loppuratkaisu riippumatta siitä, millaisella järjestelmällä tilanne hoidetaan.
- n Seuraavaksi arvioija selvittää, mikä on käyttöliittymän tarjoama ensisijainen oikea polku parhaan loppuratkaisun saavuttamiseksi:
 - n Mitkä mekaaniset toimenpiteet käyttäjän pitää tehdä
 - n Mitkä mentaaliset toimenpiteet käyttäjän pitää tehdä:
 - mitkä tiedon palaset hänen on näytöltä etsittävä ja luettava,
 - mitä toimenpiteitä hänen on tehtävä päässään, esim. laskutoimituksia, vaihtoehtojen suhteuttamista toisiinsa ja päätösten tekemistä.
- n Arvioija simuloi useaan kertaan oikean sekvenssin kaikkine toimenpiteineen (mekaaniset ja mentaaliset).

Simulointitestaus

Oikea suorituspolku

- n Hyödyllinen ajatusmalli: oikean polun suorittamisessa ikään kuin simuloidaan sellaista käyttäjää, jolla on
 - n käyttöliittymän toimintalogiikasta (epärealistisen) täydellinen tietämys (ei simuloida todellisen käyttäjän mahdollista realistista harhailua ja väärin toimintojen käynnistämistä),
 - n mutta käyttöliittymän tietosisällöstä realistinen tietämys (esimerkiksi käyttäjä ei voi etukäteen tietää, onko ravintolassa tilaa haluttuna iltana; käyttäjä tietää, että McDonald's tarjoaa hampurilaisia ja pikaruokaa).
- n Tarkoituksena on selvittää ensisijaisesti hyödyllisyys- ja tehokkuusongelmia, ei kälin opittavuusongelmia.
 - n Vertaa kognitiiviseen läpikäyntiin, jossa oletetaan käyttäjän ensimmäinen käyttökerta eli käyttäjä ei tunne tämän kälin toimintalogiikkaa lainkaan → Opittavuusongelmia
 - n Menetelmät olettavat eri ääripäät toimintologiikan tuntemuksesta.

Copyright © 2006 / Antti Latva-Koivisto, Sari A. Laakso

Simulointitestaus

Paras loppuratkaisu

- n Oikea polku johtaa parhaaseen mahdolliseen loppuratkaisuun.
 - n Arvioijan tietämys on eri asia kuin käyttäjän tietämys realistisessa käyttötilanteessa.
 - n Arvioijan on (viime kädessä) hankittava kaikki todelliseen tarjontaan liittyvä tieto, jotta hän pystyy päättämään, mikä olisi tässä tilanteessa käyttäjälle paras oikea loppuratkaisu.
 - n Järjestelmän avulla selville saatava paras ratkaisu voi olla eri kuin todellisuudessa paras loppuratkaisu.
 - n Tämä saattaa johtua esimerkiksi käyttöliittymän tietosisältöongelmasta: järjestelmässä ei ole sellaista tietosisältöä, jonka avulla todellisuudessa parhaan loppuratkaisun voisi löytää.

Copyright © 2006 / Antti Latva-Koivisto, Sari A. Laakso

Simulointitestaus

Oikean polun simulointi

- n Arvioija jäljittelee askel askeleelta käyttäjän ajatuksia ja toimenpiteitä oikean suorituspolun tekemisessä, ja simuloi useaan kertaan samaa sekvenssiä.
- n Toimenpiteet on simuloitava niin yksityiskohtaisesti kuin mahdollista. Älä simuloi ylimalkaisesti (esim. "Tästä se sitten katsoisi sopivan bussin").
 - n Simuloi jokainen käyttäjän toimenpide (esim. vierittää puoli näyttöä alaspäin, siirtyy seuraavalle sivulle, siirtyy takaisin).
 - n Simuloi tarvittavien tiedon palasten etsiminen ja lukeminen näytöltä (käyttäjä lukee, että "perillä vasta 14.40, liian myöhään", sitten katsoo edellistä: "perillä jo 13.00, mutta tässä on hidas vaihto" jne.). Vihje: Näytä aina etsittävät ja luettavat kohdat hiiren kursorilla tai kynällä, jotta huomaat itsekkin, mitä pitää katsoa.
 - n Käytä koko ajan todellisia syötteitä. Esim. käyttäjän kotiosoite voi olla "Maaherrankatu 25 D 9", mutta ei "asdf" tai "xxxxx".

Copyright © 2006 / Sari A. Laakso, Antti Latva-Koivisto

Simulointitestaus

Oikean polun simulointi loppuun asti

- n Myös järjestelmän käyttöön (näennäisesti) liittymättömät toimenpiteet täytyy simuloida.
- n Simulointi on vietävä loppuun asti, eli siihen saakka, kunnes käyttäjä on saavuttanut tavoitteen, joka käyttötapauksessa on kuvattu (esim. *kunnes käyttäjä on ajanut autolla perheineen hotelli Mesikämmeneen ja mennyt huoneeseensa*).
- n Käyttötilanteen loppupään simulointi voi paljastaa yllättäviä ongelmakohtia käyttöliittymästä (erityisesti puuttuvaa tietosisältöä), jos käyttäjällä tuleekin vaikeuksia päästä parhaaseen loppuratkaisuun.
 - n Esimerkiksi ajokartan avulla ei oikeasti löydäkään perille, tai hotellihuone ei vastaakaan sitä, mitä varausjärjestelmä käyttäjälle lupasi.

Copyright © 2006 / Sari A. Laakso, Antti Latva-Koivisto

Simulointitestaus: vihjeitä

Simulointitestausvihjeitä Käyttötilanteiden valinta

- n Toisin kuin peruskurssilla, simulointitestaukseen mukaan otettavia käyttötilanteita ei kannata valita umpimähkään.
- n Testaukseen kannattaa valita realistisista käyttötilanteista niitä, jotka johtavat sellaiseen oikeaan sekvenssiin, joka paljastaa mahdollisimman paljon ongelmakohtia (kysehän on *testauksesta!*). Näin saadaan maksimaalinen kattavuus pienellä määrällä testitapauksia.
- n Valinnassa kannattaa suosia mm. sellaisia realistisia käyttötilanteita, joissa käyttäjältä puuttuu parhaan loppuratkaisun löytämiseen vaadittavaa tietämystä. Tästä seuraa oikeassa sekvenssissä vertailun tarvetta.

Simulointitestausvihjeitä Käyttötilanteiden valinta: vertailu

- n Valitse sellainen realistinen tilanneasetelma, jossa käyttäjällä on parhaan loppuratkaisun kanssa muutama muu hyvä kilpaileva vaihtoehto (vähintään noin 2 muuta) sekä useita huonoja vaihtoehtoja.
 - n Oikeassa simulointisekvenssissä käyttäjän täytyy punnita hyviä vaihtoehtoja keskenään ja arvioida, mikä olisi paras tässä tilanteessa. Käyttäjän pitää pystyä myös nopeasti havaitsemaan huonot vaihtoehdot huonoiksi (ilman sokkovalintaa).
- n Jos käyttäjä ei pysty järjestelmän avulla erottamaan parasta vaihtoehtoa muista sitä lähellä olevista vaihtoehdoista (tai edes huonoista vaihtoehdoista), järjestelmässä on tietosisältöongelma.
- n Jos vaihtoehtojen erottaminen toisistaan vaatii paljon toimenpiteitä ja usealla eri näytöllä käymistä, käyttöliittymässä on tehokkuusongelmia (sekä mekaanisia että mentaalisia).

Copyright © 2006 / Antti Latva-Koivisto, Sari A. Laakso

Simulointitestausvihjeitä Vertailumerkinnät kuvasarjoihin

- n Oikeasta sekvenssista tehtävän kuvasarjan (ks. Käyttöliittymät-kurssi) näyttökuviin tehtävät vertailumerkinnät auttavat löytämään sellaisia käyttöliittymän ongelmakohtia, jotka liittyvät käyttäjän mielessä tapahtuvaan vaihtoehtojen vertailemiseen ja päätöksentekoon.
- n Vertailumerkinnöillä ympäröidään ne yksittäiset tiedon palaset, joiden perusteella käyttäjä päättää, että kyseinen vaihtoehto on hyvä tai huono.
 - n Älä ympäröi kaikkea dataa, jota käyttäjä voisi näytöltä lukea tai katsoa, vaan ainoastaan ne datapalaset, jotka *vaikuttavat* juuri tässä tapauksessa juuri tämän käyttäjän valintapäätökseen.
- n Jos vertailumerkinnät päätyvät yhden näytön sisällä hajalleen, tai mikä pahempaa, kuvasarjan usealle eri näytölle, käyttöliittymässä on todennäköisesti mielessä pitämisen ongelma (mentaalinen tehokkuus).

Copyright © 2006 / Antti Latva-Koivisto, Sari A. Laakso

Simulointitestausvihjeitä Vertailutilanteet ja sokkovalinta

- n Usein vertailutilanteissa parhaaseen mahdolliseen loppuratkaisuun johtavan oikean suorituspolun verralla käyttäjä joutuu tekemään sokkovalinnan.
- n Sokkovalinta ilmenee simulointisekvenssistä esimerkiksi näin:
 - n *"Kokeilempa klikata vaikkapa tuota ravintolaa." "Ai, tämä onkin täynnä."*
 - n *"Tuo voisi varmaan olla jokin hyvä elokuva." "Ei kun tämä onkin jokin romanttinen rakkausjuttu."*
 - n *"Näistä ei oikein tiedä, ostan nyt vaikka tuon matkaoppaan."*
- n Näissä esimerkeissä käyttäjä ei pysty tekemään käyttöliittymän tarjoamilla tiedoilla perusteltua valintaa. Tästä käyttöliittymän kohdasta puuttuu tietosisältöä sekä *Critical Differences* – pattern.

Copyright © 2006 / Antti Latva-Koivisto

Simulointitestausvihjeitä Ongelmalliset variaatiot ja toteumat

- n Kun olet simuloinut käyttötapauskelle yhden toteuman, kysy itseltäsi, olisiko todellisuudessa olemassa tämän tilanneasetelman sellaisia ilmeisiä variaatioita tai saman tilanneasetelman vaihtoehtoisia toteumia, joissa käykin käyttäjän kannalta huonosti, eli jokin realistinen ongelma estää häntä pääsemästä parhaaseen loppuratkaisuun.
 - n Esimerkkejä ongelmista, jotka voivat johtaa käyttäjän huonoon lopputulokseen: Käyttäjä ei tiedäkään ravintolan osoitetta, käyttäjä on myöhässä jonkin ulkoisen syyn takia, käyttäjä hukkaa järjestelmästä muistiin kirjoittamansa tiedon, käyttäjä luulee salin 3 sijaitsevan Tennispalatsin yläkerrassa, käyttäjä saa flunssan...
- n Simuloi tämä ongelmallinen sekvenssi. Jos käyttäjä päättyy siinä huonoon loppuratkaisuun, eikä järjestelmä tee mitään sen estämiseksi, käyttöliittymässä on usein jokin hyödyllisyysongelma.

Copyright © 2006 / Antti Latva-Koivisto

Simulointitestausvihjeitä Muiden käyttötilanteiden ongelmat

- n Jos testauksen kuluessa tuntuu siltä, että käyttöliittymässä voisi olla ongelma, joka ei kuitenkaan näy käsillä olevan käyttötapauksen simuloinnista, älä jää siihen jumiin.
 - n Voit kirjoittaa mieleesi tulleen ongelman muistiin, mutta laita se sen jälkeen sivuun ja pysy käsillä olevassa tilanteessa. Älä jää jumiin miettimään tilanteeseen liittymättömiä näkökulmia, jotka voivat olla todellisen käytön kannalta täysin epäoleellisia ja joiden järkevyyttä et saa selville ilman realistista käyttötilannetta.
 - n Saatuasi edellisen tilanteen simulointitulokset valmiiksi, voit seuraavaksi etsiä sellaista käyttötilannetta ja sille mahdollista simulointisekvenssiä, josta muistiin kirjaamasi ongelma tulisi esille.
- n Jos huomaat käyttöliittymässä olevia tietoja tai toimintoja, joita ei ole tarvittu missään simulointisekvenssissä, voit yrittää löytää realistisen tilanteen ja sille suorituspolun, jossa kyseisen toiminnon käyttäminen olisi optimaalista. Jos sellaista ei löydy, toiminto on todennäköisesti turha.

Copyright © 2006 / Antti Latva-Koivisto

Simulointituloksina käyttöliittymäongelmia

Tähän osioon on kerätty esimerkkejä simulointitestauksen tyypillisesti tuottamista käyttöliittymäongelmista. Nämä luettelot on tarkoitettu simulointitestauksen opiskeluvaiheen avuksi, jotta arvioijan olisi helpompaa päästä jyvälle siitä, miten simulointitestaus tuottaa ongelmahavaintoja ja millaisia löytäneet ongelmakohdat voivat olla.

Simulointituloksina käliiongelmia Ongelmien havaitseminen

- n Jotta testaaaja voi havaita simulointitestauksella ongelmia, hänen on opittava simuloinnin lisäksi olemaan tietoinen siitä, mitkä toimenpiteet hän joutuu simulointisekvenssissä tekemään.
- n Toisen tekemästä oikeellisesta simulointisekvenssistä on helpompi nähdä ongelmia, koska tällöin ei tarvitse tehdä kahta kognitiivisesti raskasta asiaa (oikeellinen simulointi + simulointisekvenssin piirteiden havainnointi) samaan aikaan.
- n Simulointisekvenssistä tehtävä kuvasarja auttaa ongelmien havaitsemisessa. Tällöin oikea polku on tallennettuna ulkoiseen muistivälineeseen ja sekvenssi on helppo suorittaa monta kertaa. Oman kognitiivisen kapasiteetin voi käyttää simulointisekvenssistä näkyvien ongelmien havainnointiin.

Copyright © 2006 / Antti Latva-Koivisto

Simulointituloksina käliiongelmia Ensin läpimeno, sitten käytettävyys

- n Ensin testataan, meneekö käyttötapauksen suorittaminen läpi, eli onko järjestelmällä mahdollista saavuttaa paras loppuratkaisu:
 - n Aukot tietosisällössä
 - n Aukot toiminnoissa
- n Jos tehtävän suoritus menee läpi, arvioidaan käytettävyyttä:
 - n Tehokkuusongelmat (käyttäjältä vaadittu turha työ)
 - Turhat toimenpiteet (mekaaninen tehokkuus)
 - Turha mentaalitö (mentaalinen tehokkuus)
 - n (Opittavuusongelmat – käyttöliittymän tarjoama oikea polku ei vastaa käyttäjällä mielessä olevaa strategiaa)

Copyright © 2006 / Sari A. Laakso

Simulointituloksina käliiongelmia Aukot tietosisällössä ja toiminnoissa

- n Onko tavoite ylipäätään mahdollista saavuttaa järjestelmän avulla, vai puuttuuko tarvittavia tietoja tai toimintoja:
 - n Tehtävän suorittamisessa tarvittavan tiedon esille kaivamiseen tarvitaan ulkoista lähdettä, kuten toista ohjelmaa tai paperilähdettä, tai muuten suoritus ei etene (esim. Stockmannin myymälän katuosoitteen selvittäminen Reittiopasta varten Googlella; potilaan edellisten hoitotietojen etsiminen arkistomapista).
 - n Tavoitetta ei voi saavuttaa loppuun asti järjestelmän avulla, vaan suoritus katkeaa kesken kokonaan tai epäoptimaalisesta kohdasta (esim. hotellihuoneiden hinnat ja varaustilanteen voi selvittää webistä, mutta huoneen varaaminen on tehtävä puhelimella tai menemällä paikan päälle; ajo-ohjetta hotelliin ei ole lainkaan).
 - n Tehtävää ei voi tehdä ollenkaan, ei edes osittain.

Copyright © 2006 / Sari A. Laakso

Simulointituloksina käliiongelmia Tehokkuusongelmat (turha työ)

- n Käytön tehokkuutta heikentävät turhat toimenpiteet, esimerkkejä:
 - n Järjestelmä pakottaa käyttämään käsillä olevan tilanteen kannalta turhia toimintoja tai pakottaa navigoimaan turhan mutkan kautta.
 - n Toimintoja on käynnisteltävä vuorotellen eri paikoista, mikä aiheuttaa edestakaisen navigoinnin kahden tai useamman näytön välillä.
 - n Käyttäjän on syötettävä sama tieto kahteen tai useampaan kertaan eri paikkoihin taikka tehtävä samat säädöt monta kertaa, kun ohjelma unohtaa ne välillä.
 - n Käyttäjä kopioi toisesta järjestelmästä manuaalisesti tietoja (jotka järjestelmä voisi hakea automaattisesti)

Copyright © 2006 / Sari A. Laakso, Antti Latva-Koivisto

Simulointituloksina käliiongelmia Tehokkuusongelmat (turha mentaaliyö)

- n Käytön tehokkuutta heikentävä turha mentaaliyö eli 'miettimistyö', esimerkkejä:
 - n Käyttäjän on pideltävä mielessään vertailtavia vaihtoehtoja, koska vaihtoehdot on esitetty käyttöliittymässä eri paikoissa eikä niitä saa samanaikaisesti näkyville. Vaihtoehtoisesti tehtävän suorittamisessa tarvitaan ulkoista apuvälinettä, kuten kynää ja paperia tai itse tehtyä Excel-taulukkoa.
 - n Kun käyttäjä löytää itselleen sopivia vaihtoehtoja (esimerkiksi sopivia hotellivaihtoehtoja Tukholmasta), hänen on yritettävä pitää löytämänsä hyvät vaihtoehdot mielessään sen sijaan, että ohjelma muistaisi ne.
 - n Käyttäjän pitää laskea päässä jotakin sen perusteella, mitä näytöllä näkyy (esim. elokuvan päättymisajan laskeminen lisäämällä kesto alkamisaikaan tai ensi viikon torstain päivämäärän laskeminen näytöllä näkyvän tämän päivän päiväyksen perusteella).

Copyright © 2006 / Sari A. Laakso, Antti Latva-Koivisto

Simulointituloksina käliiongelmia Opittavuusongelmat (vaikea keksiä)

- n Mitkä toimenpiteet tai niiden seuraukset ovat vaikeita keksiä tai ymmärtää? Esimerkkejä:
 - n Toimenpiteitä, joita on vaikea keksiä:
 - Kohde, johon liittyvät toimenpiteet eivät ole näkyvissä, esim. kuva tai tekstinpätkä, josta ei voi tajuta, että sitä voisi klikata.
 - Toimintopainikkeen tai linkin harhaanjohtava otsikko, joka ei vaikuta oikealta, vaikka todellisuudessa on se oikea.
 - Käyttöliittymän komponentin sisältöä huonosti kuvaava otsikko: käyttäjä on mahdotonta päätellä, mitä syötekenttään pitäisi kirjoittaa ja missä muodossa.
 - n Toimenpiteiden seurauksia, joita on vaikea ymmärtää:
 - Ohjelman antama vaikeaselkoinen palaute, josta on vaikea päätellä, mitä ohjelma teki, esimerkiksi lähtikö viesti nyt vai ei.

Copyright © 2006 / Sari A. Laakso

Simulointitestausta: esimerkkejä

- n Unicafe
- n VR: junalipun ostaminen
- n Silja Line

Simulointitestausharjoitus 1: Unicafe

- n Esimerkki tyypillisestä käyttötilanteesta:

Torstaina 5.10. klo 12 Taneli on Helsingin keskustassa yliopiston kielikeskuksen aulassa. Luento on juuri päättynyt, ja Tanelilla on nälkä.

- n Taneli tietää, että lähellä sijaitsevat mm. *Porthanian* ja *päärakennuksen* Unicafe-opiskelijaravintolat, eikä *Metsätalonkaan* ravintola ole kaukana. Myös *Ylioppilasaukion* ravintolan hän tietää olevan kävelymatkan päässä.
- n Taneli ei tiedä, mitä ruokaa missäkin paikassa on tänään tarjolla, eli mihin hänen kannattaisi mennä syömään.

- n Tehtävä: Mitä vikaa seuraavassa simulointitestausten sekvenssissä on?

Simulointitestausharjoitus 2: VR

n Käyttötilanne:

n Espoossa Tapiolassa asuva opiskelija Harri on menossa siskonsa ja tämän miehen luokse Seinäjoelle viettämään joulua. Sisko on kutsunut koko perheen paikalle joulun pyhiksi. Harrilla on viimeinen tentti perjantaina 22.12.2006 klo 14-18. Hän tietää, että hänellä saattaa olla sen jälkeen vielä joululahjaostoksia tekemättä. Siskon luokse voisi sinänsä mennä jo perjantaina, mutta viimeistään aattoamuna on syytä olla paikalla.

n Mitkä ongelmat simulointitestausta tuo esille?

Copyright © 2006 / Antti Latva-Koivisto

Lippu (VR:n järjestelmä)

VR E-LIPUN TILAUSVAHVISTUS

Tulosta Ohje

31.10.2006, klo 20:56

Kiitos tilauksestasi.

Tulosta tämä vahvistus ja ota se mukaasi matkalle. Vahvistus sisältää matkustamiseen tarvittavat lipputiedot, jotka on esitettävä junassa lipuntarkastuksen yhteydessä. Lippu on henkilökohtainen, joten lipun lisäksi tulee konduktöörille esittää myös henkilöllisyystodistus.

Junalippu, Opiskelija (OP): Latva-Koivisto, Antti **Hinta: 21,10 €**

Lipputarkiste: **-4 UYT G5R-**

Lauantai 23.12.2006 **13:06 Helsinki** **16:21 Seinäjoki**
InterCity 49 2. lk

Vaunu 2, Ikkunapaikka 85, Yläkerta

Hinta yhteensä: 21,10 €

Alv 8 %: 1,56 €

Tilausnumero: E73001328869

Lipputiedot lähetetään myös sähköpostiin:
Antti.Latva-Koivisto@iki.fi

Lipputiedot lähetetään myös tekstiviestinä matkapuhelimeen:
+358 405782747

Copyright © 2006 / Antti Latva-Koivisto

Esimerkki 3: Silja Line Hyttivertailun päätöksenteko

Kiteytetty käyttötapaus

Perheen tavoite: Nelihenkinen perhe harkitsee syyslomalle Tukholman-risteilyä, mutta heillä ei ole tietoa hyttivaihtoehdoista eikä hinnoista.

Tilatietoja

- n Perhe tietää, että Tukholmaan lähtee laivoja joka päivä.
- n Heillä on etukäteen mielessään noin 200-250 euron matka.
- n Vanhemmat tietävät, että aiemmillla Tallinnan-matkoilla lapsista (5 ja 7 v.) on ollut kivaa katsella hytin ikkunasta ulos.
- n <Ks. laivojen ja hyttien tiedot Siljan web-sivuilta. >

RISTELYINNÄT

Palva Tukholmaan risteily
Silja Symphony, Silja Serenade 22.8.2002 - 16.9.2002

Hytti	Lahdet	Lämpö	Lähö
Suhte	toisella ja	toisella ja	toisella ja
Tweed II-luokka	198	252	200
Ensimmäinen luokka	224	278	260
Perusseläluokka	200	310	400
Suosittu luokka	300	304	470
Silja-luokka I/II	302	470	554
Suomenmeren luokka I/II	402	542	718

Koko hytin lähtään sisällyttämään risteilyä (jokaisella paapalalla) kahdeksan 1-4 henkilöä.

* C-asteen ja Silja-luokan lähtö on sisällyttämään jokaisen aluksen ja risteilyä. Luvut C-asteen luokassa lähtö on sisällyttämään jokaisen aluksen ja risteilyä.

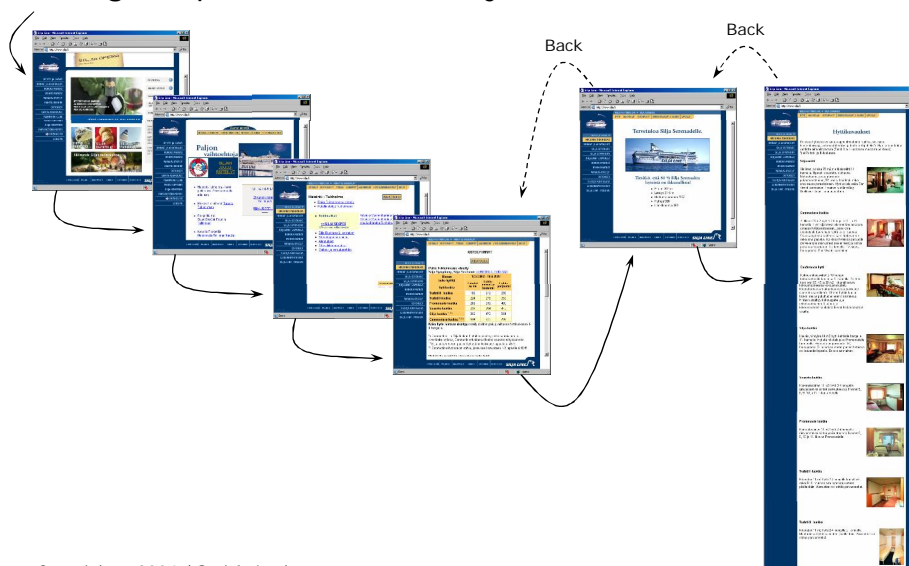
** Silja-luokan meren puolella hyttien on kahdeksan lapsella 3-45 v.

† 1-asteen hyttien on sisällyttämään jokaisen aluksen ja risteilyä.

Copyright © 2004 / Sari A. Laakso

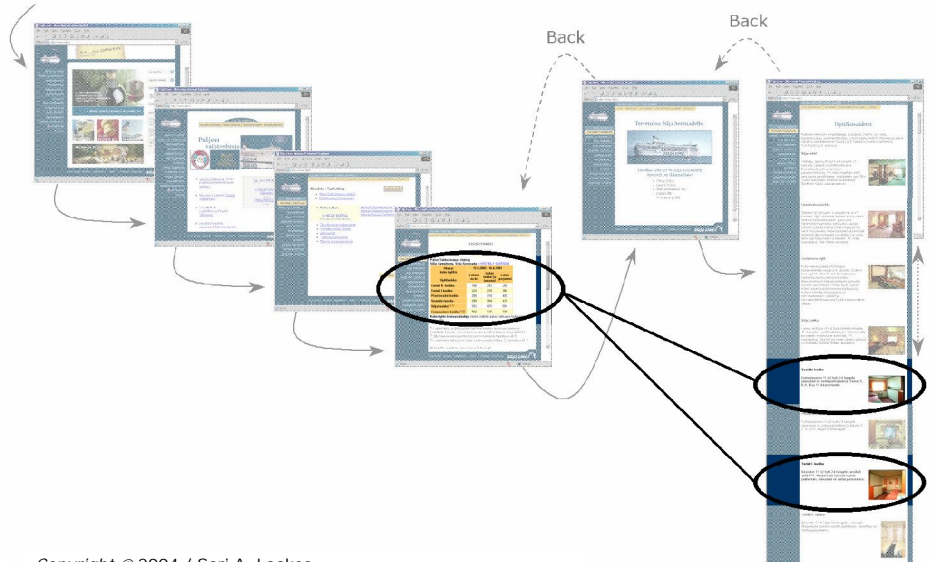
Siljan sivusto: Löytynyt ongelmakohta

Navigointipolku vertailutietojen saamiseksi



Copyright © 2004 / Sari A. Laakso

Siljan sivusto: Löytynyt ongelmakohta Päätöksenteossa tarvittavat tiedot erillään



Copyright © 2004 / Sari A. Laakso

Tee parannusehdotus:

RISTEILYHINNAT

AIKATAULU

Päivä Tukholmassa -risteily
Silja Symphony, Silja Serenade >>RISTEILY SVITISSÄ

Hinnat koko hyiltä	12.8.2002– 16.6.2003		
	Lähdöt su-ke	Lähte torstai ja lauantai	Lähte perjantai
Turisti II-luokka	186	242	290
Turisti I-luokka	224	278	350
Promenade-luokka	268	318	400
Seaside-luokka	299	394	470
Silja-luokka ^{*) 1)}	352	470	554
Commodore-luokka ^{*) 2)}	504	620	706

Koko hytin hintaan sisältyy risteily (välitön paluu) valitussa hyttiluokassa 1-4 hengelle.

^{*)} Commodore- ja Silja-luokan hintoihin sisältyy erikoisaamiainen ja viivokkeita hytissä, Commodore-luokassa lisäksi osaston erityispalvelut.

¹⁾ Silja-luokan meren puolen hytissä on lisävuode lapselle á 45 €.

²⁾ Commodore-hytissä on sohva, josta saa lisävuoteen 1-2 lapselle á 45 €

Seaside-luokka

Korkeatasoinen 11 m² hytti 2-4 hengelle (alavuoteet voi siirtää parivuoteeksi). Kannet 5, 8, 9, 10 ja 11. Ikkuna merelle.



Promenade-luokka

Korkeatasoinen 11 m² hytti 2-4 hengelle (alavuoteet voi siirtää parivuoteeksi). Kannet 8, 9, 10 ja 11. Ikkuna Promenadelle.



Turisti I-luokka

Ikkunaton 11 m² hytti 2-4 hengelle kansilla 5 sekä 9-11. Muutamissa yhteissä vuoteet päällekkäin. Alavuoteet voi siirtää parivuoteeksi.



Turisti II-luokka

Ikkunaton 11 m² hytti 2-4 hengelle 2. kannella. Muutamissa yhteissä vuoteet päällekkäin. Alavuoteet voi siirtää parivuoteeksi.



Lähde: www.silja.fi

Simulointitestaus: objektiivisuus?

Simulointitestaus Objektiivisuus ilman käyttäjiä?

- n Voivatko simulointitestauksen tulokset olla objektiivisia, vaikka menetelmä ei ole empiirinen?
- n Käytettävyydestestauksen ja käytettävyysläpikäynnin tulokset perustuvat empiriaan, jolloin saadut havainnot sinänsä ovat objektiivisia valittujen käyttäjien ja testitapausten osalta.
 - n Tulosten oikeellisuus (validiteetti) riippuu siitä, voidaanko havainnoista tehdä induktio, eli voidaanko ne yleistää. Tämä riippuu mm. siitä, edustavatko valitut käyttäjät ja testitapaukset todellista käyttöä.
- n Simulointitestauksen tulokset ovat objektiivisia, koska ne perustuvat deduktioon (hyödyllisyyden & tehokkuuden osalta).
 - n Tulosten validiteetti riippuu siitä, vastaavatko deduktiossa käytetyt oletukset todellisuutta. Simulointitestauksessa esim. oletetaan, että todellisessa käytössä esiintyy valitun käyttötapausten edustamia tilanteita, ja että eri loppuratkaisut on periaatteessa mahdollista laittaa paremmuusjärjestykseen kussakin käyttötilanteessa.

Copyright © 2006 / Antti Latva-Koivisto

Simulointitestaus: yhteenvedo

Simulointitestaus Yhteenvedo

- n Etsi käyttöliittymän tarjoama oikea polku parhaaseen mahdolliseen loppuratkaisuun. Vertaa parhaaseen todelliseen loppuratkaisuun.
- n Huomioi käyttäjän tietämys: ei voi tietää parasta ratkaisua etukäteen.
- n Asettele tilanne siten, että parhaalle ratkaisulle on useita kilpailukykyisiä vaihtoehtoja (tietosisältöongelmat, mielessä pitelyn ongelmat, mekaaninen tehokkuus).
- n Simuloi käyttötapaus loppuun asti.
- n Etsi testauksessa sellaisia tilanneasetelmia ja realistisia toteumia, joissa jokin syy voi estää käyttäjää pääsemästä parhaaseen loppuratkaisuun.
- n Simuloi kaikki käyttäjän toimenpiteet orjallisesti (klikkaukset, tiedon palasten etsiminen ja lukeminen näytöltä, mentaaliset toimenpiteet)
- n Simuloi moneen kertaan ja tee kuvasarja, jotta pystyt havaitsemaan ongelmia. Opettele tiedostamaan, mitkä toimenpiteet teet simuloinnissa.

Käyttöliittymän hyödyllisyysläpikäynti



Käyttöliittymän hyödyllisyysläpikäynti (utility walkthrough) on kehitteillä oleva oma menetelmä (Laakso & Latva-Koivisto), joka soveltuu suunnitteluvaiheessa olevan käyttöliittymän ongelmien löytämiseen.

Hyödyllisyysläpikäynti Käytettävyys vs. hyödyllisyys

Käytettävyysläpikäynti (usability walkthrough) [Bias91]

- n Vetäjä johtaa yhteissimulointia, jossa käyttäjät ehdottavat aina seuraavaa toimenpideaskelta
- n Tulokset painottuvat opittavuuteen, kuten käytettävyystestissä, mutta muitakin käytettävyyteen (ja hyödyllisyyteenkin) liittyviä näkökohtia saattaa tulla esille

Hyödyllisyysläpikäynti (utility walkthrough)

- n Simuloinnin aikana vetäjä voi demota kälin toimintalogiikkaa ja jopa muuttaa käliä lennossa piirtämällä, koska nyt ei testata käytettävyyttä
- n Käli toimii työkaluna selvitystyössä:
 - n puuttuvia käyttötapauksia
 - n epäselviä/muutettavia työkäytäntöjä ja päätöksentekoprosesseja
 - n päätöksenteossa tarvittavaa tietosisältöä ja toimintoja

Hyödyllisyysläpikäynti Opittavuus vs. päätöksenteko

- n Käytettävyyssläpikäynti
 - n Tarkoitus selvittää järjestelmän opittavuusongelmia.
 - n Läpikäynti tehdään suunniteltujen käliratkaisujen ehdoilla: käyttäjä vedätetään suunnitellun polun läpi, jotta saadaan selville, missä kohdissa polulla on ongelmia.
 - n Kälän toimintalogiikkaa ei saa muuttaa kesken läpikäynnin.
- n Hyödyllisyysläpikäynti
 - n Tarkoitus selvittää, missä kohdissa käyttäjän päätöksenteko eroaa suunnittelijan design-yrityksestä: strategiat, tietosisältö, toiminnot.
 - n Käliratkaisu on vain työkalu käyttäjän päätöksentekoprosessin selvittämistä varten. Läpikäynnissä edetään käyttäjän ongelmanratkaisustrategian mukaan, ja asennoidutaan siten, että proto ja kuvasarja voivat olla pahasti väärin.
 - n Käliratkaisuja saa ja usein pitääkin muuttaa lennossa: piirrä/leikkele/liimaa protoasi läpikäynnin edetessä. (Vasta jälkeinpäin arvioidaan, mitä muutoksia oikeasti kannattaa tehdä.)

Copyright © 2004 / Sari A. Laakso

Hyödyllisyysläpikäynti Läpikäyntipalaverin kulku

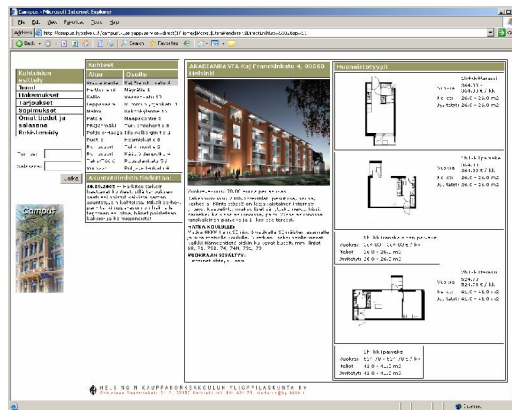
- n Kuvaile testitapaus suullisesti ja jätä paperi käyttäjälle.
- n Avaa järjestelmän alkutila näkyville.
- n Etene vaihe kerrallaan.
 - 1. "Mitä seuraavaksi yrittäisit tehdä?" Tyydy suulliseen kuvaukseen, älä yritä saada käyttäjää väkisin 'painamaan nappia'.
 - 2. Demoa käyttäjän yrittämä toimenpide protollasi jotenkin.
 - Jos käyttäjä aloitti valitsemallasi strategialla, siirry eteenpäin.
 - Jos käyttäjä aloitti eri strategialla kuin protosi sekvenssi:
 - Hyppää protossa sellaiseen kuvaan, jossa näkyy käyttäjän yrittämä toimenpide, tai
 - korjaa protoasi piirtämällä/leikkaamalla ja anna käyttäjän tehdä se, mitä hän yritti.
- n Jos käyttäjä ei osunut johonkin suunnittelemaasi vaiheeseen, esim. ei hae ravintolaa kilometrin säteellä rautatieasemasta, selvitä lopuksi:
 - n Demoa toimenpide ja kysy: "Mitä ajattelet tästä?"

Pyydä tarvittaessa ääneenajattelua!

Copyright © 2004 / Sari A. Laakso

Hyödyllisyysläpikäynti Esimerkki: KY:n asuntohakujärjestelmä

Olin muutama vuosi sitten suunnittelemassa käyttöliittymää KY:lle tulevaan järjestelmään, jolla opiskelijat hakevat ylioppilaskunnan asuntoja ja jolla asuntotoimisto tarjoaa vapautuvia asuntoja jonon kärjessä oleville opiskelijoille.



Copyright © 2006 / Antti Latva-Koivisto

Hyödyllisyysläpikäynti Esimerkki: KY:n asuntohakujärjestelmä

- n Hyödyllisyysläpikäynnissä edettiin pitkin asuntotoimiston työnkulkua.
- n Eräs vuokrasopimus oli irtisanottu, ja asuntotoimiston virkailija oli tekemässä vapautuvasta asunnosta tarjousta jonon kärjessä olevalle opiskelijalle.
- n Käsin piirretystä käyttöliittymäprototyypistä oli auki näyttö, jolla virkailija näki vapautuneen asunnon, kyseisen kohteen jonon kärjessä olevat opiskelijat, ja jonka avulla hän pystyi tekemään vapautuvasta asunnosta valitulle opiskelijalle tarjouksen, joka lähtee sähköpostilla.

Copyright © 2006 / Antti Latva-Koivisto

Hyödyllisyyslöpikäynti

Esimerkki: KY:n asuntohakujärjestelmä

- n "Mitä seuraavaksi yrittäisit tehdä?"
- n "No yrittäisin tehdä haun kohteen osoitteella." (Käyttöliittymään ei ollut suunniteltu mitään hakua.)
- n "Eli jotakin tällaista, vai?" (Piirrän uuden näytön ja siihen hakukentän, johon voi laittaa kohteen osoitteen.)
- n "Niin, siihen laittaisin Ida Aalbergin tie 1 B 23 ja katsoisin sitten, kenellä asunto nyt on ja sitten hakisin sen asukkaan puhelinnumeron ja sähköpostiosoitteen."
- n Osoittautui, että asuntotarjoukseen tarvittiin nykyisen asukkaan yhteystiedot, jotta uusi asukas voi sopia asuntoon tutustumisesta ja ennen kaikkea avaimen saamisesta suoraan nykyisen asukkaan kanssa. Aiemmassa käyttötilanneselvityksessä olimme saaneet sen käsityksen, että avaimet kulkisivat aina asuntotoimiston kautta. (Parempi ratkaisu ei tietenkään edellyttänyt hakua asunnon osoitteella, vaan korjatussa käyttöliittymässä nykyisen asukkaan yhteystiedot laitettiin oletuksena valmiiksi asuntotarjoukseen.)

Copyright © 2006 / Antti Latva-Koivisto

Hyödyllisyyslöpikäynti

Esimerkki: NCC:n tuntikirjanpito



Työnjohtaja

- n työntekijöiden työtuntikirjaus
 - n aikatyöt ja urakkatyö
 - n sairastumiset ja muut poissaolot
- n urakkasopimukset
- n lainatyöntekijät ja uudet työntekijät

Vastaava työnjohtaja

- n tarkistukset, sopimukset

Palkanmaksu

- n erikoistapausten käsittely ja tarkistukset

Copyright © 2004 / Sari A. Laakso

Hyödyllisyyslöpikäynti Testitapaukset

Testitapauksina käytettiin...

- n haastattelujen perusteella etukäteen valmisteltuja tilanteita sekä
- n käyttäjän omalla työmaalla käynnissä olevaa todellista työtilannetta.

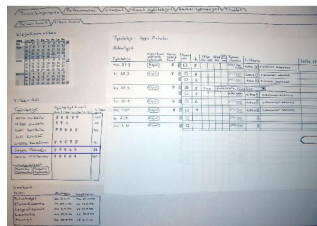
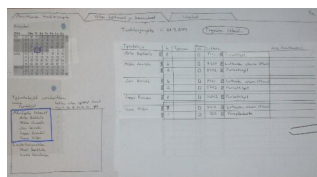


Copyright © 2004 / Sari A. Laakso



Oman tilanteen etu:
Käyttäjän pitkäkestoisessa muistissa aktivoituu paljon sellaista tietoa, jota hän ei helposti pystyisi tuottamaan ulos vieraan esimerkkitilanteen pohjalta.

Hyödyllisyyslöpikäynti Esim. 1: Puuttuva käyttötapaus



Löytynyt ongelma ja tulos

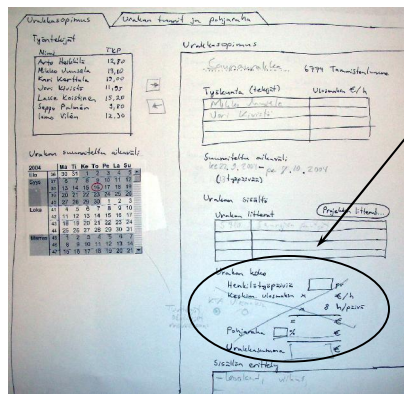
- n Päivakohtaisen kirjausnäytteen tarve oli tullut esille aiemmin, ja käli tuki tätä nyt hyvin.
- n Testikäyttäjä täyttäessään omien työntekijöidensä todellisia tunteja edellispäiviltä: "Tämä on erilainen ajattelutapa..."
- n Selvisi, että on monotonisia työkausia, jolloin päivittäinen tuntikirjaus teettäisi turhaa työtä.
- n => Kaksi kt:ta jäsenyi erilleen.
- n => Käliin viikkonäkymä ja toiminto *Kopioi kollegan viikko pohjaksi.*

Copyright © 2004 / Sari A. Laakso

Näyttökuvapiirrosten esimerkkitilanne on kuvitteellista.

Hyödyllisyysläpikäynti

Esim. 2: Väärä päätöksentekoprosessi

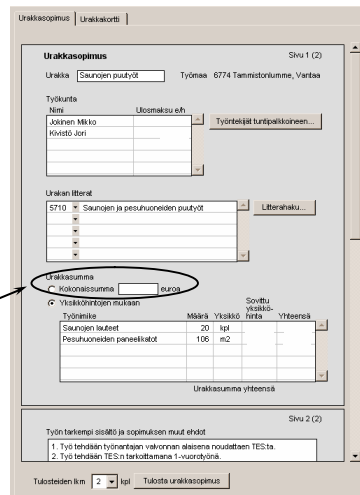
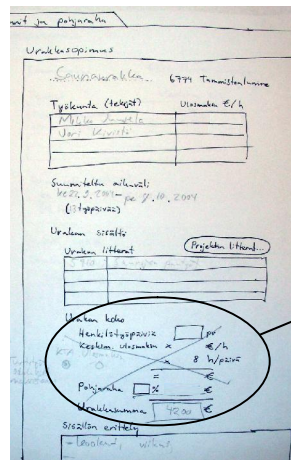


Löytynyt ongelma ja tulos

- n Käli mm. tuki urakkahinnan laskemista ulosmaksuhintojen ja arvioitun työmäärän perusteella.
- n Todellinen urakkahinnasta sopimisen päätöksenteko perustui mm.
 - n työnjohtajan omiin ja muiden tekemiin aiempiin vastaavanlaisiin urakoihin ja
 - n miesten ehdotukseen.
- n => Päätöksentekoprosessista valtaosa järjestelmän ulkopuolelle (neuvottelut).

Kaikki esimerkkidata on kuvitteellista.
Copyright © 2004 / Sari A. Laakso

Esim. 2: Päätöksentekoprosessi ulkopuolelle



Kaikki esimerkkidata on kuvitteellista.
Copyright © 2004 / Sari A. Laakso

Esim. 2: Päätöksentekodatana aiemmat urakat

Vertailudata:

- Omat aiemmat urakat tai
- muiden samanlaiset

Copyright © 2004 / Sari A. Laakso

Kaikki esimerkkidata on kuvitteellista.

Hyödyllisyysläpikäynti Käytännön vihjeitä

- n Läpikäynti kannattaa usein järjestää käyttäjien työpaikalla.
- n Hyvä määrä käyttäjiä on noin 1-2 tai enintään 3 kerralla.
- n Alkuun kannattaa laatia käyttäjän tositilanteeseen perustuvia tehtäviä, loppuun etukäteen valmisteltuja.
- n Läpikäynti tehdään yhdellä paperiprotolla, jota vetäjä käsittelee. Myös käyttäjät kirjoittavat syötteitä protoon.
- n Vetäjä saa myös demota, koska opittavuudella ei merkitystä.
- n Kun esille tulee uutta tietoa työtehtävistä, vetäjä voi nopeasti
 - n korjata proton käliratkaisua paremmin uuteen tietoon sopivaksi ja muuttaa kälin toimintalogiikkaa lennossa tai
 - n luonnostella uuteen tilanteeseen sopiva uusi kälin palanen ja ottaa simuloimalla siitä välittömästi palautetta käyttäjältä.
- n Vetäjän ei kuitenkaan pidä yleisesti kysellä mielipiteitä tai yrittää saada käyttäjiä suunnittelemaan ratkaisuja.

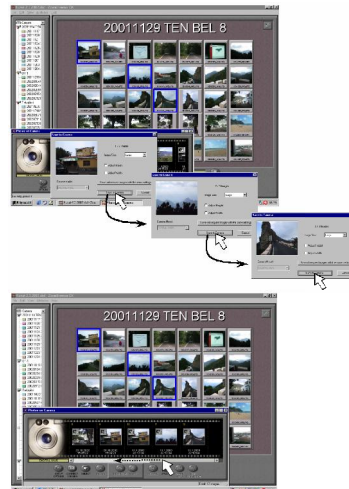
Copyright © 2004 / Sari A. Laakso

Käyttäjätarkkailu



Käyttäjätarkkailu Käyttäjän tehtävät ja käliongelmat

- n Käyttäjätarkkailua voidaan käyttää käyttötapausten selvittämisen lisäksi myös käyttöliittymän arviointimenetelmänä.
- n Tarkkailija seuraa käyttäjän toimintaa ja yrittää selvittää,
 - n mitä tehtävää käyttäjä on tekemässä ja
 - n mitä ongelmia käyttöliittymässä on.



Keskeisiä käyttösekvenssipohjaisia menetelmiä

	Käytettä- vyydestä Usability test	Käytett.- läpikäynti Usability walkthrough	Hyödyll.- läpikäynti Utility walkthrough	Simulointi- testaus Simulation- based expert review	Kognit. läpikäynti Cognitive walkthrough	Käyttäjä- tarkkailu User observation
Käyttäjät	X	X	X			X
Käyttösekv.	X	X	X	X	X	X
Tulokset: <u>1 Hyödyllisyys</u> Toiminnot ja tietosisältö <u>2 Käytettävyys</u> Tehokkuus Opittavuus Muistettav. Virheet	(Tehokkuus) Opittavuus (Muistettav.) Virheet	(Hyödyllisyys) (Tehokkuus) Opittavuus (Muistettav.) (Virheet)	Hyödyllisyys Tehokkuus	Hyödyllisyys Tehokkuus (Opittavuus) (Muistettav.) (Virheet)	Opittavuus	Hyödyllisyys Tehokkuus (Opittavuus) Muistettavuus (Virheet)
Keskeisiä eroja	Käyttäjien vapaa eteneminen myös harhapoluilla.	Käyttäjät pidetään ohjatusti testipolulla.	Edetään optimipolkua. Muuttuu välillä haastattelu- pohjaiseksi käyttötapaus- ja työnkulku- selvitykseksi.	Edetään kälän tarjoamaa optimipolkua.	Edetään kälän tarjoamaa optimipolkua.	Ei etukäteen laadittuja testitehtäviä. Tehtävät ilmenevät lennossa.

Muita arviointimenetelmiä

- Lokin keruu
- Tehtävämallinnus (GOMS, KLM)
- Heuristinen arviointi
- Heuristinen läpikäynti
- Katselmointi
- Fokusryhmät

Lokin keruu

- n Lokin kerääminen vastaa käyttäjätarkkailujen tekemistä tietämättä, mitä käyttäjät yrittävät saada toimenpiteillään aikaan.
- n Lokitietojen kunnolliseen tulkitsemiseen tarvitaan tietoa käyttäjän työnkuvasta ja tyypillisistä käyttötilanteista. Arvioijan on ymmärrettävä, mitä käyttäjä on yrittänyt saada aikaan (käyttötapauksen tavoiteosa), kun hän on tuottanut lokiin kirjatut tapahtumat.

Copyright © 2004 / Sari A. Laakso

Lokin keruu

Esimerkki 1: Unixin komentoloki

- n Käyttäjien antamista unix-komennoista voitaisiin päätellä
 - n mitä virheellisiä muiden järjestelmien käytöstä oppimiaan komentoja käyttäjät yrittävät
 - yrittävät: *dir*
 - pitäisi antaa: *ls*
 - n millaisia parametreja he eivät osaa antaa, esim.
 - yrittävät: *chmod * read all*
 - pitäisi antaa: *chmod a+r **
- n Jos käyttäjien virheellisistä yrityksistä selviää, mitä he yrittävät, voidaan esimerkiksi tehdä aliaksia.
- n Ei-toivottuja sivuvaikutuksia pitäisi osata välttää, esim. miten tulkitaan
 - *chmod rwx all*

Copyright © 2004 / Sari A. Laakso

Lokin keruu

Esimerkki 2: Yahoos loki



Yläreunan painikkeiden klikkausten määrä kaksinkertaistui, kun kuviin kuuluvat tekstit vaihdettiin sinisiksi ja alleviivatuiksi.



CHI 99 - User Interface Design for the Web -tutoriali

YAHOO!

Lokin keruu

Vahvuuksia ja ongelmia

- n Vahvuuksia
 - n Objektivistista dataa, joka perustuu käyttäjien todelliseen toimintaan.
 - n Toimenpidesekvenssit mukana datassa.
- n Ongelmia
 - n Tulkinta perustuu paljolti arvailuun, koska lokidatasta ei selviä, mitä käyttäjä yrittää.
 - Lokissa on ikään kuin pelkkä testitehtävän suorituspolku, mutta lokista puuttuu tieto siitä, mitä tehtävää käyttäjä on tekemässä. Tätä voidaan arvailla - tai selvittää käyttäjätarkkailuilla tai haastatteluilla.
 - Lokista ei näy, miksi käyttäjä teki niin kuin teki, vrt. käyttäjän reaktiot ja ääneenajattelu käytettävyydestissä.
 - n Menetelmää voidaan käyttää vain valmiin, jo käytössä olevan järjestelmän käyttöliittymän arviointiin.

Copyright © 2004 / Sari A. Laakso

Tehtävämallinnus

GOMS ja KLM

GOMS

- n Käyttäjän tehtävät mallinnetaan hierarkioina, joissa alimmalla tasolla on yksittäisiä toimenpiteitä.

Näppäintasomallin (Keystroke-Level Model, KLM) aika-arvioita

- n Näppäimenpainallus 0.35 s
- n Hiirellä osoittaminen 1.10 s
- n Mentaalioperaatio 1.35 s
- n Vasteaika 1.2 s
- n Näppis/hiiri-siirtymä 0.4 s

Ks. [Preece94, s. 419-424, 685-687]

Copyright © 2004 / Sari A. Laakso

Esimerkki: Wordin *Save As*

- | | |
|----------------------------------|--------------|
| n Käsi hiirelle | 0.4 |
| n Kursori <i>File</i> -valikkoon | |
| n Mieti | 1.35 |
| n Siirrä hiiri <i>File</i> :en | 1.10 |
| n Valitse <i>Save as</i> | |
| n Klikkaa menu auki | 0.35 |
| n Mieti | 1.35 |
| n Siirry <i>Save as</i> :iin | 1.10 |
| n Klikkaa <i>Save as</i> | 0.35 |
| n Ohjelma kysyy tiedostonimeä | |
| n Vasteaika | 1.2 |
| n Mieti tiedostonimi | 1.35 |
| n Kirjoita tiedostonimi | 8*0.35 |
| n Paina Enteriä | 0.35 |
| Yhteensä | 13.05 |

Tehtävämallinnus

Vahvuuksia ja ongelmia

- n Vahvuuksia
 - n Ottaa kantaa tärkeimpään ja ensisijaisesti korjattavaan käyttöliittymäongelmien osa-alueeseen: tehokkuuteen.
 - n Käyttäjän toimenpidesekvenssit arvioinnissa mukana.
- n Ongelmia
 - n Sopii lähinnä kahden käliratkaisun tehokkuusvertailuun rajallisissa tapauksissa.
 - n Menetelmä ei huomioi päätöksenteon suorittamiseen kuluvaa kognitiivista työtä.
 - Hyvä ja huono datan organisointi tuottavat saman aika-arvion, vaikka ne aiheuttavat hyvin erilaisen määrän kogn. työtä.
 - Jos käyttöliittymä vaatii enemmän toimenpiteitä ('klikkauksia') vähemmällä kognitiivisella työllä, menetelmä tuottaa sille isomman aika-arvion – vaikka päätöksentekoon tarvittavan kognitiivisen työn vuoksi tämä käyttöliittymä oikeasti veisi vähemmän aikaa kuin vertailukäyttöliittymä.

Copyright © 2004 / Sari A. Laakso

Heuristinen arviointi

- n Heuristinen arviointi [Nielsen90] on eräs asiantuntija-arvioiden tekotapa. Nielsenin esittämä heuristinen arviointi perustuu ns. 10 kohdan tarkistuslistan eli kymmenen heuristiikan läpikäyntiin, mutta muitakin tarkistuslistoja on käytetty.
- n Arvioijalla on lähtökohtana näyttökuvat ja tarkistuslista. Joskus hänelle toimitetaan lisäksi käyttöskenaarioita.
- n Arvioija ottaa esiin näytön kerrallaan ja käy läpi listan kaikki kohdat.
- n Esimerkki: icalin heuristinen arviointi

Copyright © 2004 / Sari A. Laakso

Heuristinen arviointi

Nielsenin heuristiikat 1 / 2

- 1 Järjestelmän tilan näkyvyys
 - n Näytä aina käyttäjälle, missä tilassa järjestelmä on
 - n Tarjoa nopea palaute (feedback)
- 2 Luonnollinen ilmaisutapa
 - n Käytä käyttäjälle tuttuja käsitteitä ja kieltä
 - n Esitä tieto luonnollisessa loogisessa järjestyksessä
- 3 Kontrolli käyttäjän käsissä
 - n Käyttäjän on pystyttävä poistumaan mistä tahansa tilasta
 - n Tarjoa perumismahdollisuus (undo/redo)
- 4 Yhdenmukaisuus ja standardit
 - n Käytä johdonmukaisesti samoja termejä joka paikassa
 - n Noudata toteutusympäristön standardeja
- 5 Virheiden estäminen
 - n Yritä välttää käyttäjän virheet jo etukäteen
- 6 Tunnistaminen ja muistaminen
 - n Tee toimenpiteistä ja niiden kohteista näkyviä (visible)
 - n Älä pakota käyttäjää muistamaan tietoa, jonka jämä voisi näyttää

Heuristinen arviointi

Nielsenin heuristiikat 2 / 2

7 Käytön tehokkuus

- n Tarjoa pikavalintanäppäimiä (shortcuts)
- n Anna käyttäjän räätälöidä useimmin tarvitsemaansa toimintoja

8 Minimalistinen suunnittelu

- n Karsi käyttöliittymästä pois sellainen tieto, jota käyttäjä ei tarvitse
- n Sijoita harvemmin tarvittavat tiedot pidemmän polun päähän

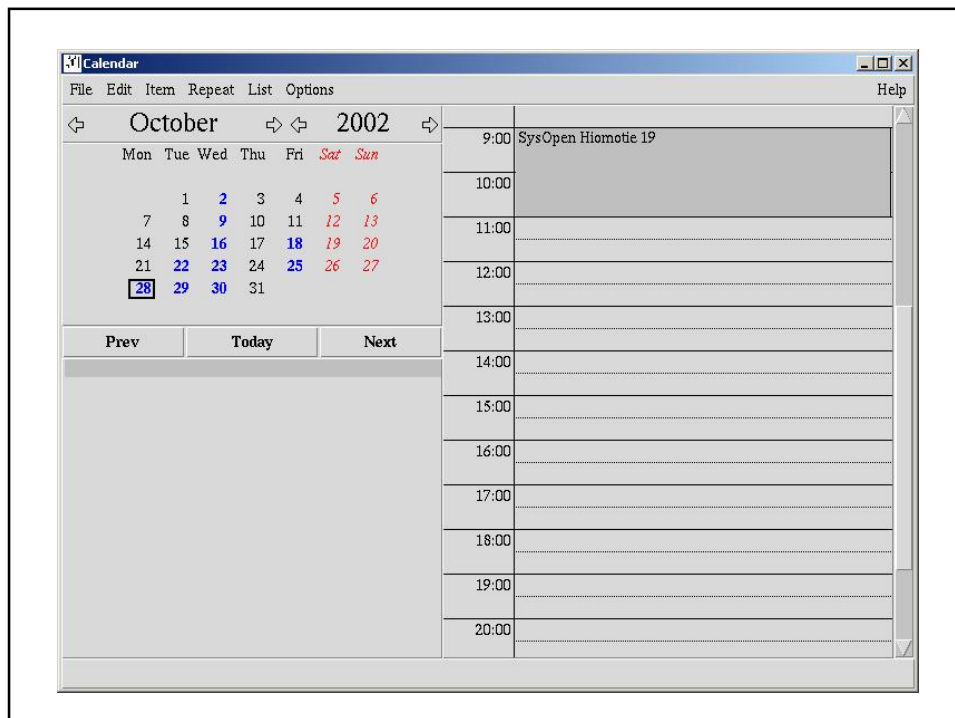
9 Virheistä toipuminen

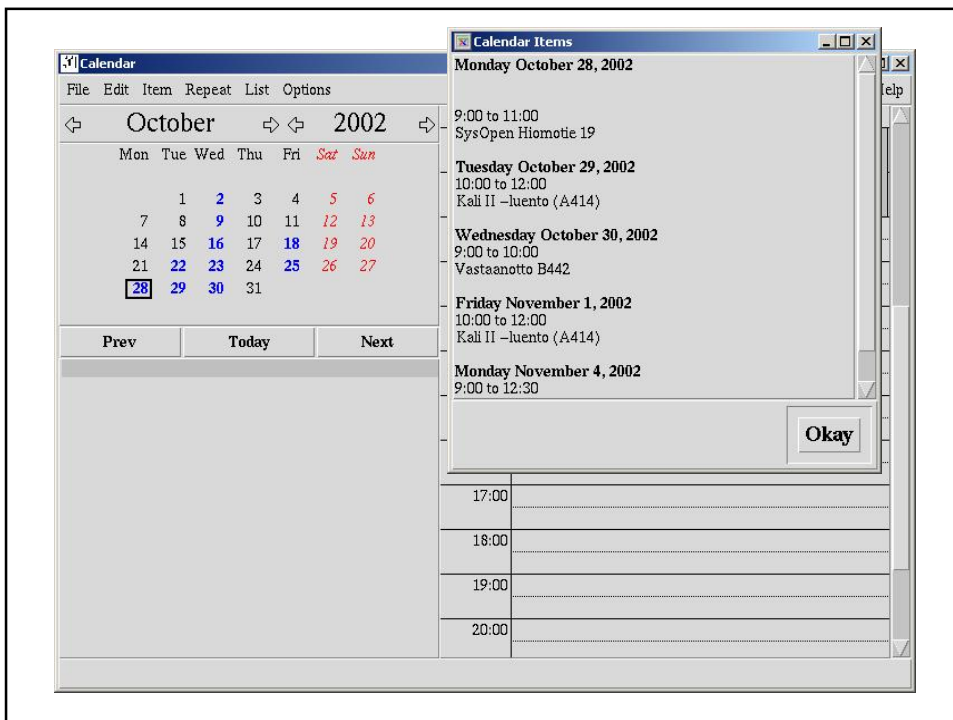
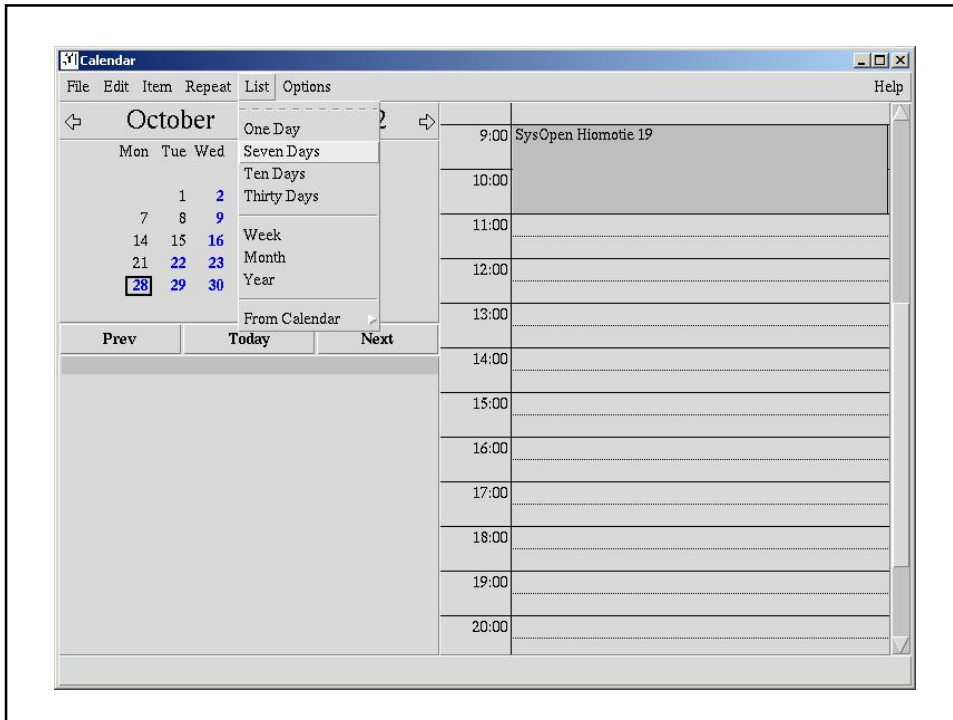
- n Esitä virheilmoitukset ymmärrettävällä kielellä
- n Kuvaa virheilmoituksessa ongelma ja esitä siihen ratkaisuja

10 Opastus ja käyttöohjeet

- n Useimmissa tapauksissa on tarkoituksenmukaista, että käyttäjä pystyy käyttämään ohjelmaa myös ilman käyttöohjeita
- n Kun laadit käyttöohjeen, esitä käyttäjien tehtävien mukaisesti jäsennehtyinä toimenpiteet, jotka käyttäjän on tehtävä tavoitteensa saavuttamiseksi
- n Älä laadi liian laajoja käyttöohjeita

Ks. [Nielsen02a]





Heuristinen arviointi

Ongelmia

- n Menetelmä ei perustu käyttäjän toiminnan arviointiin.
- n On helppo nähdä, että tarkistuslistan 10 korkean tason periaatetta ovat yleisellä tasolla oikeellisia, mutta monet keskeiset käliongelmat eivät jää kiinni, koska tarkistuslistan kohdat eivät osu ollenkaan
 - n kontekstisidonnaisiin ongelmiin eivätkä
 - n käyttäjän toimintosekvenssin (usein toistuviinkaan!) ongelmakohtiin.
- n Menetelmä tuottaa paljon virheellisiä ongelmahavaintoja, ks. esim. [Sears97], [Lauesen05, s. 19-20].

Copyright © 2004 / Sari A. Laakso

Heuristinen läpikäynti

- n Heuristisessa läpikäynnissä (heuristic walkthrough) [Sears97] arvioija tarkastelee käyttäjän toimenpidesekvenssiä.
- n Ongelmia etsitään kognitiivisesta läpikäynnistä lainattujen neljän apukysymyksen avulla.
- n Lisäksi ongelmia etsitään käyttämällä Nielsenin tarkistuslistaa [Nielsen94].
- n Tuottaa menetelmän kehittäjän mukaan [Sears97] oikeellisempia tuloksia kuin heuristinen arvio:
 - n Enemmän oikeita oleellisia ongelmia
 - n Ei vääriä ongelmia (näennäisvikoja)

Copyright © 2006 / Sari A. Laakso, Antti Latva-Koivisto

Katselmointi

- n Katselmointia voidaan tehdä
 - n yhdenmukaisuuden tarkistamista varten (saman ohjelman sisällä tai saman tuoteperheen eri ohjelmien välillä)
 - n standardeja vasten tai
 - n käyttöliittymäohjeistoja (guidelines) vasten.
- n Ongelmia
 - n Yhdenmukaistaminen voi tuottaa hyviä tai huonoja käliratkaisuja tapauksesta riippuen.
 - n Jos standardien tai käliohjeistojen käyttöliittymäratkaisut ovat huonoja, katselmointi tuottaa vääriä tuloksia.
 - n Katselmointi ei tuota minkäänlaista arviointitietoa siitä, onko käyttöliittymä kelvallinen todellisissa käyttötilanteissa.

Copyright © 2004 / Sari A. Laakso

Ks. esim. [Wixon94]

Fokusryhmät

- n Fokusryhmät kootaan tyypillisesti 6-9:stä kohderyhmän käyttäjästä. Palaverissa keskustellaan uuden konseptin ominaisuuksista ja toiminnallisuudesta tai arvioidaan suunniteltua järjestelmää näyttökuvia katsomalla.
- n Ongelmia
 - n Design-ratkaisuja ei pidä arvioida kysymällä käyttäjien mielipiteitä. Se, mitä käyttäjät *sanovat*, on monesti eri asia kuin se, mitä he todellisuudessa *tekisivät*.
 - n Fokusryhmäpalaverit voivat olla hyviä esimerkiksi markkinointivaihtoehtojen toimivuuden arvioinnissa, mutta fokusryhmien käyttäminen *käliratkaisujen* arviointiin tuottaa satunnaisia tuloksia, joiden pohjalta käliratkaisuja ei pidä ryhtyä muuttamaan. Käytä luotettavampia menetelmiä, kuten läpikäyntejä, simulointipohjaisia asiantuntija-arvioita, käyttäjätarkkailuja tai käytettävyydestä.

Copyright © 2004 / Sari A. Laakso

Ks. [Nielsen02b]

Kälin arviointimenetelmiä Käyttäjien kanssa ja ilman

Arviointi loppukäyttäjien avulla (user testing)

- n käytettävyydestaus (usability testing)
- n käytettävyydläpikäynti (usability walkthrough)
- n hyödyllisyysläpikäynti (utility walkthrough)
- n käyttäjätarkkailut (user observations)
- n käyttölokin keruu (software logging)
- n [fokusrhmäkeskustelut (focus groups)]

Arviointi ilman käyttäjiä (inspection, expert review)

- n kognitiivinen läpikäynti (cognitive walkthrough)
- n simulointitestausta (usage simulation)
- n heuristinen arviointi (heuristic evaluation)
- n heuristinen läpikäynti (heuristic walkthrough)
- n käyttäjän tehtävien ja toimenpiteiden mallinnus (GOMS, KLM)
- n [katselmointitilaisuudet (consistency inspection, standards inspection)]

Copyright © 2006 / Sari A. Laakso, Antti Latva-Koivisto

Muita arviointimenetelmiä

Eräänlaisia kehysmenetelmiä

	Lokin keruu Software logging	Tehtävä-mallinnus GOMS, KLM	Heuristinen arviointi Heuristic evaluation	Heuristinen läpikäynti Heuristic walkthrough	Katselmointi Inspection	Fokusrhmät Focus groups
Käyttäjät	X					X
Käyttösekv.	X	X		X		
Tulokset: <u>1 Hyödyllisyys</u> Toiminnot ja tietosisältö <u>2 Käytettävyys</u> Tehokkuus Opittavuus Muistettav. Virheet	Tehokkuus (Opittavuus) (Muistettav.) Virheet	Tehokkuus	Opittavuus Muistettav. Virheet	(Hyödyllisyys) (Tehokkuus) Opittavuus Muistettav. Virheet		
Huomioita	Käyttäjän tehtävät ja tavoitteet jäävät arvailujen varaan.	Vaatii vertailu-asetelman (vähintään 2 eri käliiratkaisua).	Löytää vain 'yleisiä käytettävyysongelmia' ja niistäkin vain ne, jotka osuvat tarkistuslistaan. Oleellisemmat kontekstisidonnaiset (tilanteeseen liittyvät) ongelmat jäävät huomaamatta. Löytää myös paljon vääriä ongelmia (naennäisvikoja).	Arvioija suorittaa käyttöliittymällä ensin tehtäviä ja käyttää apuna vastaavia kysymyksiä kuin kogn. läpikäynnissä (ei orjallisesti). Lopuksi arvioidaan käyttäen Nielsenin heuristiikkoja.	Ei arvioi itse käliiratkaisua. Esim. yhdenmukaisuus voi parantaa tai huonontaa käliiratkaisuja.	Eraanlainen mielipideselvitys. Eri asia kuin arviointimenetelmä. Designia ei pidä arvioida käyttäjien <i>puheiden</i> perusteella.

Lähteitä arviointimenetelmistä 1/4

Käytettävyystestaus

- Nielsen93 Nielsen J.,
Usability Engineering.
Academic Press, San Diego, CA, 1993. Ks. luku 6.
- Rubin94 Rubin J.,
Handbook of Usability Testing: How to Plan, Design, and Conduct Effective Tests.
John Wiley & Sons, USA, 1994.

Käytettävyysläpikäynti

- Bias91 Bias R.,
Walkthroughs: Efficient Collaborative Testing.
IEEE Software, Vol. 8, No. 5, 1991, s. 94-95.
- Bias94 Bias R. G.,
The Pluralistic Usability Walkthrough: Coordinated Empathies.
Teoksessa Nielsen J., Mack R.L. (toim.), *Usability Inspection Methods*. John Wiley & Sons, New York, 1994, s. 63-76.

Lähteitä arviointimenetelmistä 2/4

Kognitiivinen läpikäynti

- Lewis97 Lewis C., Wharton C.,
Cognitive walkthrough.
Teoksessa Helander M., Landauer T., Pradhu P. (toim.), *Handbook of human-computer interaction*. Elsevier Science B.V., 1997, s. 717-732.
- Wharton94 Wharton C. et al.,
The Cognitive walkthrough Method: A Practitioner's Guide.
Teoksessa Nielsen J., Mack R.L. (toim.), *Usability Inspection Methods*. John Wiley & Sons, New York, 1994, luku 5.
- Valorinta05 Valorinta J.,
Kognitiivinen läpikäynti ja käytettävyystestaus käyttöliittymän arviointimenetelminä.
Pro gradu –tutkielma, Helsingin yliopisto, TKTL, 2005.

Lähteitä arviointimenetelmistä 3/4

Lokin keruu ja tehtävämallinnus

- Preece94 Preece J. et al.,
Human-Computer Interaction.
Addison Wesley, 1994. Ks. osa IV, s. 595-708.

Heuristinen arviointi

- Nielsen90 Nielsen J., Molich R.,
Heuristic Evaluation of User Interfaces.
CHI'90 Conf. Proc., ACM, New York, 1990, s. 249-256.
<http://portal.acm.org/citation.cfm?id=97281>
- Nielsen94 Nielsen J.,
Heuristic Evaluation.
Teoksessa Nielsen J., Mack R.L. (toim.), *Usability Inspection Methods*. John Wiley & Sons, New York, 1994, s. 25-62.
- Nielsen02a Nielsen J.,
Ten Usability Heuristics.
http://www.useit.com/papers/heuristic/heuristic_list.html

Heuristinen läpikäynti

- Sears97 Sears A.,
Heuristic Walkthroughs: Finding the Problems Without the Noise.
International Journal of Human-Computer Interaction, Vol. 9, No. 3, 1997, s. 213-234.

Lähteitä arviointimenetelmistä 4/4

Katselmointi

- Wixon94 Wixon D. et al.,
Inspections and design reviews: Framework, history, and reflection.
Teoksessa Nielsen J., Mack R.L. (toim.), *Usability Inspection Methods*. John Wiley & Sons, New York, 1994, s. 79-104.

Fokusryhmät

- Nielsen02b Nielsen J.,
The Use and Misuse of Focus Groups.
<http://www.useit.com/papers/focusgroups.html>