

# Web-sovellusten käytettävyyshaasteet

TURUN YLIOPISTO  
Tietotekniikan laitos  
TkK-tutkielma  
Tietotekniikka  
Toukokuu 2025  
Jeremias Salonen

TURUN YLIOPISTO  
Tietotekniikan laitos

JEREMIAS SALONEN: Web-sovellusten käytettävyyshaasteet

TkK-tutkielma, 20 s.  
Tietotekniikka  
Toukokuu 2025

---

Web-sovellusten käytettävyys on tärkeä osa niiden toimivuutta, käyttäjäkokemusta ja menestystä. Tämä tutkimus käsittelee web-sovellusten käytettävyyshaasteita, sekä keinoja niiden tunnistamiseen ja ehkäisyyn. Tutkimus on toteutettu kirjallisuuskatsauksena, ja siinä analysoidaan aiempien tutkimuksien pohjalta käytettävyysongelmia viidessä pääkategoriassa: käyttöliittymän, sisällön, suorituskyvyn, tyydyttävyyden ja kehityksen haasteet.

Käyttöliittymän haasteet liittyvät epäselvään navigointiin sekä graafisen suunnittelun virheisiin. Sisällön haasteet puolestaan heikentävät luettavuutta ja ymmärrettävyyttä, mikä heikentää käyttäjän kykyä omaksua sovelluksessa esitettävää tietoa. Suorituskyvyn haasteet, kuten hidas vasteaika, tekevät sovelluksen käyttämisestä hitaampaa. Tyydyttävyyden haasteet korostavat käyttäjän subjektiivista käyttökokemusta ja kehityksen haasteet vaikuttavat sovelluksen ylläpitoon.

Käytettävyyden haasteiden huomiotta jättäminen voi johtaa käyttäjien turhautumiseen, käyttäjien tekemien virheiden kasvamiseen, tehokkuuden heikkenemiseen ja ylläpidon vaikeutumiseen. Huono käytettävyys on yleisesti haitallista sovellukselle, mutta yksittäisten tekijöiden tunnistaminen voi olla vaikeaa. Tutkimus tarjoaa kattavan yleiskuvan web-sovellusten käytettävyyshaasteista, jonka avulla voi oppia tuntemaan käytettävyyshaasteiden eri kategoriat.

Tutkimuksen tuloksissa korostuu, että käytettävyyshaasteet ovat ehkäistävissä tai korjattavissa kehityksen aikana, mikäli ne tunnistetaan riittävän aikaisin. Iteratiivinen käyttäjätestaus on erityisen tärkeä käyttäjäpalautteen keräämisessä ja siten haasteiden tunnistamisessa.

Asiasanat: käytettävyys, web-sovellukset, käyttöliittymäsuunnittelu

# Sisällys

<b>1</b>	<b>Johdanto</b>	<b>1</b>
<b>2</b>	<b>Tausta</b>	<b>3</b>
2.1	Käytettävyys . . . . .	3
2.2	Web-sovellus . . . . .	6
<b>3</b>	<b>Käytettävyysshaasteet</b>	<b>9</b>
3.1	Käyttöliittymän haasteet . . . . .	10
3.2	Sisällön haasteet . . . . .	12
3.3	Suorituskyvyn haasteet . . . . .	13
3.4	Tyydyttävyyden haasteet . . . . .	16
3.5	Kehityksen haasteet . . . . .	16
3.6	Haasteiden ehkäisy . . . . .	17
<b>4</b>	<b>Yhteenveto</b>	<b>19</b>
	<b>Lähdeluettelo</b>	<b>21</b>

# 1 Johdanto

Web-sovellukset ovat tärkeä osa jokapäiväistä elämää. Terveyspalvelut, verkkopankit sekä useat muut julkisen ja yksityisen sektorin palvelut ovat tarjolla webissä. Näitä sovelluksia käyttävät kaiken ikäiset ja taitoiset käyttäjät. Palveluiden takaaminen kaikille edellyttää, että sovellusten käyttäminen on sujuvaa kaikille käyttäjille. Web-sovellukset tulisi kehittää ihmisten käytettäväksi ja on tärkeää, että ihmiset pystyvät myös käyttämään niitä. [1] Sovellus voi tehdä kaiken tarvittavan, mutta se on täysin hyödytön, jos käyttäminen on mahdotonta.

Web-sovelluksen hyvän käytettävyyden seurauksena käyttäjien tehtävien suorittaminen on nopeampaa ja tehokkaampaa. Nopean sovelluksen käyttökokemus on miellyttävämpää, koska käyttäjä ei joudu odottamaan sovelluksen vastausta. Heikon käytettävyyden seurauksena sovelluksen käyttö on vaikeampaa ja hitaampaa, mikä voi johtaa käyttäjien määrän pienenemiseen. Käytettävyys voi myös vaikuttaa käytettävyyteen arvostelujen välityksellä, koska monissa arvosteluissa kiinnitetään huomiota käytettävyyteen. [2] Hyvät arvostelut ovat tärkeitä varsinkin kaupallisille sovelluksille, koska käyttäjät voivat arvosteluiden perusteella valita kilpailevan sovelluksen. Huonon käytettävyyden seurauksena voi olla käyttäjän tyytymättömyyden lisäksi myös mahdolliset rasitusvammat, erityisesti toistuvien liikkeiden seurauksena. [1]

Käytettävyyden kehittämisessä on tärkeä tunnistaa käytettävyyteen kohdistuvia haasteita, jotta nämä voidaan ottaa huomioon. Tiedossa olevat haasteet voidaan hel-

pommin välttää ja korjata valmiista sovelluksesta. Jotta käytettävyyshaasteet pystytään tunnistamaan ja ehkäisemään, tässä tutkielmassa tutkitaan web-sovellusten käytettävyyttä seuraavien tutkimuskysymysten avulla:

**TK1:** Mitä web-sovellusten käytettävyyshaasteita on olemassa?

**TK2:** Miten web-sovellusten käytettävyyshaasteita on pyritty ratkaisemaan?

Tutkielma on toteutettu kirjallisuuskatsauksena. Tieto on haettu hakulauseella ”*Web AND Usability AND challenge*”, tietokannoista IEEE ja ACM Digital Library. Tuloksia löytyi yhteensä 50, joista poistettiin epärelevantit. Tämän lisäksi ”Web of Science - tietokannasta haettiin samoilla hakutermeillä, minkä tuloksena saatiin 1363 tutkimusta. Nämä rajattiin vuosikymmenen mukaan 2020-luvulle ja poimittiin relevantit aineistot.

Luvussa 2 käsitellään tutkimuksen taustaa. Taustaluvussa avataan käytettävyyden määritelmiä ja mittareita, sekä web-sovelluksen rakennetta. Luvussa 3 käsitellään käytettävyyshaasteita. Haasteet on jaettu kategorioihin käytettävyyden määritelmien avulla. Luku 4 on yhteenvetoluku, jossa kootaan työn havainnot ja vastataan tutkimuskysymyksiin.

## 2 Tausta

Web-sovellusten käytettävyys perustuu moniin eri tekijöihin, jotka vaikuttavat käyttäjäkokemukseen. Jotta voidaan analysoida web-sovellusten käytettävyysaasteita, on ensin tarkasteltava käytettävyyden määritelmää ja web-sovelluksen rakenteellisia ominaisuuksia.

Web-sovellusten rakenne on oleellista ymmärtää käytettävyyttä käsiteltäessä. Käytettävyysaasteet saattavat johtua sovelluksen teknisistä rajoitteista, joten sovelluksen tekniset ominaisuudet on hyvä tuntea. Web-sovelluksen eri osien tunnistaminen auttaa hahmottamaan eri osiin vaikuttavat käytettävyysaasteet.

### 2.1 Käytettävyys

Käytettävyys jaetaan ISO 9241-11-standardin [3] mukaan kolmeen ryhmään: tehokkuuteen (engl. effectiveness), suorituskykyyn (engl. efficiency) ja tyydyttävyyteen (engl. satisfaction). *Tehokkuus* tarkoittaa käyttäjän tavoitteiden saavuttamisen tarkkuutta ja täydellisyyttä. Tehokkuus vaikuttaa käyttäjän tekemiin virheisiin ja tehtävien lopputuloksen laatuun. *Suorituskyky* mittaa käyttäjän tavoitteiden saavuttamisen tarkkuutta ja täydellisyyttä suhteessa käytettyihin resursseihin. Yleisesti tavoitteiden saavuttamisessa kulutettava resurssi on aika, joten suorituskyky viittaa sovelluksen käytön nopeuteen. *Tyydyttävyyys* on käyttäjän positiivinen asenne sovellusta kohtaan. Käytettävyys muodostuu näiden tekijöiden kokonaisuudesta.

Jos yhdessä näistä tekijöistä on huomattava heikkous, huononee koko sovelluksen käytettävyys. [4]

Käytettävyyden toinen laajasti käytetty määritelmä jakaa käytettävyyden viiteen ryhmään. Käytettävyys jaetaan Jacob Nilsenin[5] mukaan opittavuuteen, suorituskyykyyn, muistettavuuteen, virheisiin ja tyydyttävyyteen. Yhteiset ryhmät ISO-standardin kanssa ovat suorituskyyky ja tyydyttävyyys. *Opittavuus* viittaa sovelluksen käytön oppimisen helppouteen. Sovelluksen tulisi olla uudelle käyttäjälle helppo ja nopea oppia. Opittavuus on jokaisen käyttäjän kohtaama tekijä, koska järjestelmän käyttäminen edellyttää käytön oppimista. Opittavuuteen liittyy myös toinen ominaisuus: *muistettavuus*. Sovellus, jonka muistettavuus on hyvä, mahdollistaa käytön jatkamisen tauon jälkeen ilman sovelluksen oppimista uudelleen. Varsinkin sovellusta satunnaisesti käyttäville hyvä muistettavuus on tärkeää sujuvan käyttökokemuksen takaamiseksi. *Virheet* viittaavat käyttäjän tekemiin virheisiin ja niiden vaikutuksiin. Käyttäjien tulisi tehdä sovellusta käyttäessä vähän virheitä, ja tapahtuneista virheistä pitää pystyä palautumaan helposti. Sovellus ei pysty estämään kaikkia käyttäjän tekemiä virheitä, mutta mahdollisten virheiden määrää voi vähentää. Virheiden mahdollinen vakavuus on pyrittävä minimoimaan, jotta käyttäjän virheestä ei johdu suuria haittoja. [5]

Käytettävyyden mittaamisessa hyödynnetään mittareita jokaista ISO 9241-11-standardin [3] ryhmää kohden. Näiden mittarien yhteisellä tuloksella voidaan havainnollistaa sovelluksen käytettävyyttä ryhmittäin eriteltynä.

**Tehokkuuden** mittaamisessa käytetään viittä mittaria. [4]

**Binäärinen** tehtävien suorittaminen merkitsee kuinka monta tehtävää käyttäjä suoritti oikein, suoritti valitun ajan jälkeen ja ei suorittanut ollenkaan.

**Tarkkuuden mittarit** määrittävät kuinka monta virhettä käyttäjät tekivät tehtäviä suorittaessa tai tehtävien lopputuloksessa.

**Muistaminen** viittaa siihen kuinka paljon käyttäjät muistavat sovelluksen käyttöliittymästä. Muistettava tieto on yleisesti elementtejä käyttöliittymästä.

**Täydellisyys** viittaa tehtävien suorittamisen kattavuuteen. Täydellisyys mittaa tehokkuutta tilanteissa joissa käyttäjä saavutti eriävän ratkaisun tehtävälle, mikä ei kuitenkaan ole väärä.

**Lopputuloksen laadulla** mitataan tehtävien suoritusta laajemmin. Esimerkiksi työn tai oppimisen laatua mitataan tällä mittarilla.

**Suorituskyvyn** mittaamiseen käytetään kuutta mittaria. [4]

**Aika** viittaa tehtävien suorittamiseen kulunutta aikaa. Aika voidaan mitata kokonaiselta tehtävältä tai vain tietyistä osista tehtävää.

**Syöttönopeus** voidaan mitata joko kaiken sovellukseen tulevan syötteen nopeutena tai oikein syötettyjen arvojen syöttönopeutena.

**Psyykkistä vaivannäköä** on vaikea mitata sellaisenaan. Käyttäjien psyykkistä kuormitusta havainnollistetaan esimerkiksi sykkeen vaihtelevuutena.

**Käyttökuviot** viittaa sovelluksen käytössä tapahtuviin kuvioihin. Näitä kuvioita voi mitata toistuvien tapahtumien kuten hiiren painalluksien avulla.

**Oppimisen** mittareilla viitataan mittauksiin suorituskyvyn vaihtelussa. Suorituskyvyn kasvaminen sovelluksen pysyessä samana voi olla seuraus käyttäjän oppimiselle.

**Tyytyväisyyttä** mitataan kolmella mittarilla. [4]

**Mieltymys** mittaa mitä käyttöliittymää käyttäjät suosivat. Mieltymys mitataan laittamalla käyttäjät valitsemaan käyttöliittymien välillä.

**Käytön helppous** on käyttäjän yleinen tyytyväisyys. Helppous voidaan mitata suoraan käyttäjän palautteena.

**Tarkat suhtautumiset** viittaavat käyttäjien suhtautumista käyttöliittymään tarkemmin. Käyttäjän suhtautuminen käyttöliittymään voidaan kysyä suoraan käyttäjältä kirjallisena palautteena.

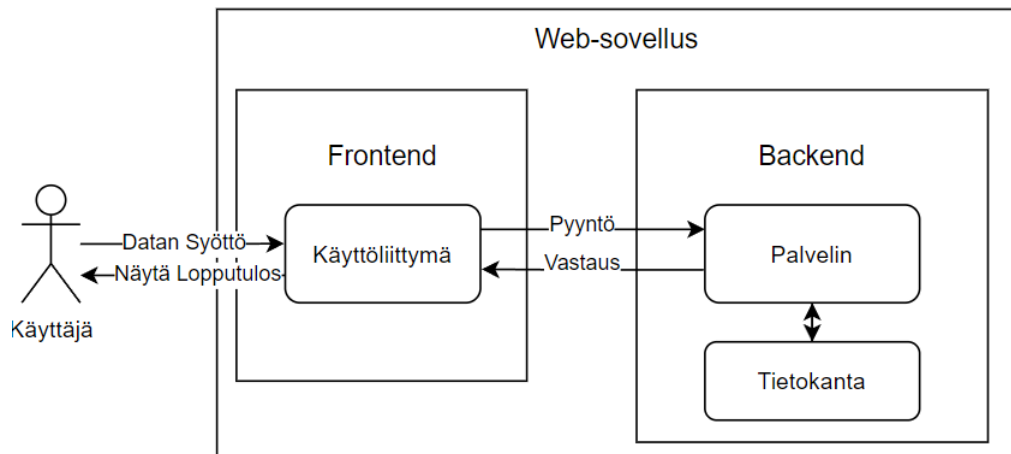
## 2.2 Web-sovellus

Web-sovellukset ovat sovellusohjelmia, joita käytetään verkkoselaimien avulla. Niiden käyttö ei edellytä sovelluksen asentamista käyttäjän laitteelle, vaan selain pystyy avaamaan sovelluksen verkon välityksellä. Web-sovellukset perustuvat asiakas-palvelin-arkkitehtuuriin, jossa palvelin varastoi tietoa ja asiakas pääsee käsiksi näihin tietoihin. Käyttäjä pystyy käyttämään palvelimella sijaitsevaa web-sovellusta selaimen avulla. [6]

Useimmat web-sovellukset kuitenkin koostuvat laajemmasta arkkitehtuurista kuin pelkkä yksittäinen asiakas-palvelin suhde. Kolmitasoisessa arkkitehtuurissa sovellus erotellaan kolmeen erilliseen tasoon. Ensimmäinen taso vastaanottaa käyttäjän pyynnöt ja välittää ne seuraavalle tasolle. Ensimmäinen taso myös välittää toisen tason vastauksen ja esittää sivun käyttäjälle. Sovelluksen toinen taso on palvelintaso, joka vastaa sovelluksen toiminnan logiikasta. Palvelinsovellus toimii linkkinä ensimmäisen tason pyyntöjen ja kolmannessa tasossa sijaitsevan tiedon välillä. Palvelin käsittelee pyydetyn tiedon ja välittää sen takaisin ensimmäiselle tasolle. Kolmas taso on tietokanta, johon sovelluksen tiedot on tallennettu. Tietokanta sisältää kaikki sovelluksen tarvitsevat tiedot ja hallitsee sitä palvelimen pyyntöjen mukaan. Palvelimen pyynnöistä riippuen tietokantaan voidaan lisätä tietoa, tietokannassa olevaa tietoa voidaan muokata tai haluttu tieto voidaan palauttaa palvelimelle. [7]

Web-sovellusten yhteydessä kolmitasoisessa arkkitehtuurin ensimmäistä tasoa

kutustaan frontendiksi. Sovelluksen frontend on kaikki, mitä käyttäjä näkee selaimen avulla. Ne osat joiden kanssa käyttäjä on suoraan tekemisissä, ovat sovelluksen frontendiä, mistä johtuen suurin osa käytettävyyshaasteista liittyy suoraan frontendiin. Frontendin suhde muihin web-sovelluksen komponentteihin on esitetty kuvassa 2.1.



Kuva 2.1: Web-sovelluksen komponentit

Sovelluksen graafinen käyttöliittymä (engl. graphical user interface, GUI) tarkoittaa kaikkia graafisia elementtejä joiden kanssa käyttäjä on vuorovaikutuksessa. Web-sivut ovat dokumentteja, jotka muotoillaan HTML-kielillä. HTML on merkintäkieli, jonka avulla määritetään, mitä tyyppiä mitkäkin sivun elementit ovat. Selaimet pystyvät lukemaan HTML-dokumentin ja esittämään sivun oikein muodostettuna. [8]

HTML kuvaa kuitenkin vain sivun rakennetta ja sen tukena toimivat kielet CSS ja JavaScript, joiden avulla voidaan muokata sovelluksen tyyliä ja toiminnallisuutta. Tyylien lisäämiseen käytetään CSS-kieltä, jonka avulla voidaan määrittää, miltä sivun elementtien kuuluu näyttää. JavaScript on perustana käyttöliittymän kaikelle toiminnallisuudelle. Sivun joka koostuu pelkästään HTML- ja CSS-määrittelyistä on hyvin staattinen. JavaScriptin avulla sivulle voidaan luoda sovelluslogiikkaa, interaktiivisia elementtejä ja muita toiminnallisuuksia. [8]

Jotta web-dokumentille voidaan luoda interaktiivisuutta JavaScriptin avulla, tarvitaan rajapinta, joka mahdollistaa dynaamisen muokkaamisen JavaScriptin avulla. Tämä onnistuu dokumenttioliomallin eli DOM:n (engl. Document Object Model) avulla. DOM jäsentää sivun HTML-esityksen hierarkkiseksi puuksi, jonka jokaista solmua vastaa HTML-elementti. DOM-puun elementtejä voidaan lukea, lisätä, poistaa ja muokata JavaScriptin avulla. Selain päivittää esitettävän sivun DOM-puussa tapahtuneiden muutoksien jälkeen. [9]

Sovelluksen osat jotka toimivat erillään käyttäjän laitteesta ovat osa sovelluksen backendia. Backendin osat ja suhde muihin web-sovelluksen komponentteihin on esitetty kuvassa 2.1. Kuten kuvassa esitetään, käyttäjä käyttää backendin toimintoja välillisesti frontendin kautta ja backend on yhteydessä tietokantaan. Frontend ja backend toimivat yhdessä siten, että frontendin komponentit tekevät pyyntöjä backendille ja saavat tarvitsemiaan resursseja. Vaikka käyttäjä ei käytä suoraan backendiä eikä tietokantaa, voivat backend ja tietokanta vaikuttaa epäsuorasti käytävyyteen esimerkiksi verkosta johtuvien viiveiden seurauksena. [8]

## 3 Käytettävyysaasteet

Käytettävyysaasteet ovat tekijöitä, jotka vaikuttavat sovelluksen käytön tehokkuuteen, miellyttävyyteen ja nopeuteen. Ne voivat johtua suunnittelusta, puutteellisesta testaamisesta, teknisistä rajoitteista tai sovelluksen rakenteen monimutkaisuudesta. Käytettävyysaasteet saattavat tulla esiin sovelluksen toiminnallisuudessa, sisällön esityksessä, sovelluksen suorituskyvyssä sekä käyttäjän yleisessä tyytyväisyydessä.

Hyvä käytettävyys on tärkeä osa sovelluksen käyttökokemusta ja siksi käytettävyysaasteiden tunnistaminen on tärkeää sovelluksen menestyksen kannalta. Jos käytettävyysaasteita ei huomioida, seurauksia voi olla sovelluksen virheellinen käyttö, käyttäjän turhautuminen ja sovelluksen käytön lopettaminen. Varsinkin monimutkaisissa järjestelmissä, joissa halutaan suorittaa useita tehtäviä nopeasti ja tarkasti, pienetkin käytettävyysaasteet voivat kasvaa suuriksi haitoiksi.

Käytettävyysaasteet voidaan jakaa viiteen eri kategoriaan: käyttöliittymä, sisältö, suorituskyky, tyydyttävyyden ja kehitys. Jokainen kategoria vaikuttaa käytettävyyteen eri tavoin ja niiden välillä esiintyy myös vuorovaikutusta. Käyttöliittymä muodostaa käyttäjän yhteyden sovellukseen ja on siksi tärkeä osa sovelluksen esitystä. Käyttöliittymä vaikuttaa suoraan sovelluksen käytön intuitiivisuuteen. Sisältö liittyy tiedon esittämiseen ja oikeellisuuteen, kun taas suorituskyvyllä viitataan järjestelmän käytön nopeuteen ja tehokkuuteen. Tyytyväisyys kuvaa käyttäjän sub-

jektiivista kokemusta sovelluksesta ja kehityksen haasteet liittyvät taustalla oleviin haasteisiin sovelluksen kehityksessä ja ylläpidossa.

Seuraavissa alaluvuissa esitetään, miten käytettävyyshaasteiden viisi kategoriaa ilmenevät käytännössä. Kategorioista esitetään esimerkkitapauksia, joita on löytynyt tutkimuksissa. Näiden konkreettisten esimerkkien avulla voidaan muodostaa kattava kuva yleisimmistä käytettävyyshaasteista. Alla olevassa taulukossa 3.1 on esitetty, mihin kategorioihin kukin esitetty lähde liittyy.

Taulukko 3.1: Lähteiden kategoriat

Artikkeli	Käyttöliittymä	Sisältö	Suorituskyky	Tyydyttävyyys	Kehitys
Alfayez ja Al-Tawriy [10]	x	x	x		
Otaiza, Rusu ja Roncagliolo [11]	x	x			
Elahi, Maqueda ja Morato [2]		x			
Brinck, Gergle ja Wood [12]			x		x
Nik Ahmad, Hamid ja Mohd Lokman [13]			x	x	
Romano Bergstrom, Olmsted-Hawala, Chen et al. [14]			x		

### 3.1 Käyttöliittymän haasteet

Käyttöliittymän ongelmat vaikuttavat suoraan käyttökokemukseen, koska käyttäjä vuorovaikuttaa sovellukseen käyttöliittymän avulla. Huono käyttöliittymä laskee sovelluksen tehokkuutta kasvattaa käyttäjän tekemiä virheitä ja heikentää tyydyttävyyttä tekemällä sovelluksen käytöstä monimutkaisempaa.

Basrahin yliopiston tutkimuksessa mitattiin yliopiston sivujen käytettävyyttä kahden arvioijan avulla. [10] Tutkimuksen tuloksena sivulta löytyi 3143 ongelmaa käytettävyyteen liittyen. Käytettävyyden ongelmat jaettiin neljään pääryhmään: *sisältö, navigointi ja linkit, suorituskyky* sekä *käyttöliittymä*. Tutkimuksessa löytyi

yhteensä 693 ongelmaa liittyen käyttöliittymän suunnitteluun. Nämä ongelmat jaettiin kahdeksaan luokkaan haasteiden tunnistamisen helpottamiseksi. Sivun esteettisyyttä pidettiin epäammattimaisena ja heikosti visuaalisesti vetoavalta. Sivulla oli epäjohdonmukaisuutta fonteissa sekä asettelussa. Fonttien koko, väri ja tyyli eivät olleet yhdenmukaisia, myös taulukot ja kuvaajat olivat väärin kohdennettuja. Sivun painikkeet ja linkit eivät antaneet kuvaa käyttötarkoituksestaan. Osa valikoista olivat joko poissa näkyvässä tai muuten saavuttamattomia. Sivuilta löytyi rikkinäisiä kuvia, jotka eivät näyttäneet haluttua sisältöä. Sivujen otsikoissa oli ongelmia siten, että niitä ei ollut ollenkaan tai ne olivat sopimattomia. Sovelluksen sivutus ei ollut selvää. Taustalla olevien tekstien lukeminen oli vaikeaa, pienestä fonttikoosta ja huonoista väriyhdistelmistä johtuen. [10]

Linkit yhdistävät sovelluksen sivut keskenään, mistä johtuen vialliset linkit voivat tehdä joidenkin sivujen avaamisesta mahdotonta. Navigointiin liittyviä ongelmia löytyi yhteensä 1391 [10] ja nämä ongelmat jaettiin seitsemään luokkaan. Sivuilta löytyi rikkinäisiä linkkejä, joiden avaamisesta seurasi tyhjä sivu tai virheilmoitus. Jotkut sivun linkeistä olivat hyödyttömiä, koska niiden käyttämisen jälkeen käyttäjä pysyy samalla sivulla. Sivulla löytyi linkkejä joiden nimi ei vastaa sisältöä. Sivuilta puuttui kokonaan mahdollisuus sisäiselle haulle. Sivujen välillä liikkuminen edes ja takaisin koettiin vaikeaksi, sekä halutun tiedon löytäminen edellytti monta vaihetta. Sovelluksesta löytyi sivuja, jotka eivät sisältäneet ollenkaan linkkejä muihin sivuihin. Joillain sivuilla taas oli monta linkkiä, jotka avaavat saman sisällön.

Kaupallisten sovellusten käytettävyyttä mittaavassa tutkimuksessa [11] osa kriittisimmistä ongelmista käytettävyydessä liittyivät käyttöliittymään. Tutkimuksessa tutkittiin eri arviointimenetelmien tehokkuutta käytettävyyden arvioimisessa ja tuloksena tehokkain arviointimenetelmä on monien eri menetelmien käyttäminen yhdessä. Arviointimenetelmät eivät vaikuta käytettävyyshaasteisiin, mutta sovelluksista löydetty ongelmat listattiin kriittisyyden mukaan. Sovelluksien kriittisiä ongel-

mia liittyen käyttöliittymään olivat edelliselle sivulle navigoimisen mahdottomuus, hiirellä sivun vierittäminen ei toiminut, avattuja linkkejä ei merkitty, sivulla on liikaa tietoa ja linkkejä, uloskirjautuminen on mahdotonta ja haku ei toiminut. Moni näistä ongelmista liittyy sivun linkkeihin ja niiden huonoon toteutukseen. [11]

Tutkimuksista [10], [11] selviää, että käyttöliittymän osalta suurin haaste käytettävyydelle ovat linkit ja niiden toteuttaminen. Käytettävyyden takaamiseksi on tärkeää, että sovelluksen sivuilla voi liikkua taaksepäin ja jo avatut linkit pystyy erottamaan avaamattomista. Tämä tekee sovelluksen selaamisesta ja tiedon etsimisestä sulavampaa. Myös käyttöliittymän ulkonäkö vaikuttaa käytettävyyteen. Epäammattimaisen näköinen ja epäselkeä käyttöliittymä tekee käyttämisestä epämiellyttävää.

## 3.2 Sisällön haasteet

Sovelluksen sisältö on sovelluksesta löytyvää informaatiota. Sisällön haasteet voivat vaikuttaa käyttäjät kykyyn ymmärtää ja löytää tarvittavaa tietoa sivulta. Väärä tai vanhentunut sisältö voi antaa käyttäjälle väärää tietoa, mikä ei ole käytettävyyden kannalta hyvä asia.

Basrahin yliopiston tutkimuksessa löytyi yhteensä 861 sisältöön liittyvää haastetta, jotka jaettiin viiteen kategoriaan: *puuttuva tieto*, *vanhentunut tieto*, *tyhjä sivu*, *kieliopilliset virheet* ja *asiaankuulumaton sisältö*. Yleisimmät sisällölliset ongelmat liittyivät sisällön paikkansapitävyyteen ja tarpeellisuuteen, sekä tiedon ilmaisemiseen. [10]

Kaupallisten sovelluksen käytettävyyttä mittaavassa tutkimuksessa sovelluksista löytyi sisältöön liittyviä haasteita, joiden kriittisyys oli merkitty pienemmäksi kuin käyttöliittymään ja navigointiin liittyvät haasteet. Sisältöön liittyviä haasteita olivat: *tärkeän ja epäoleellisen tiedon erottelu*, *haut palauttavat tyhjiä sivuja* ja *hakutuloksien järjestyksen epäloogisuus*. [11]

Sovelluksen sisältöön vaikuttaa sivujen luettavuus, eli kuinka helppoa sisällön lukeminen on. Huono luettavuus tekee sivusta vaikeaselkoisen ja epätyytyttävän. Luettavuuteen vaikuttaa tekijät kuten tekstien fontit, sisällön sijoittelu, värien kontrastit ja graafisten elementtien asiaankuuluvuus. Graafisten elementtien asiaankuuluvuus viittaa elementteihin, jotka eivät paranna sivua mitenkään, vaan ovat pelkästään haitaksi, kuten turhat kuvat. Fonttikoko, fonttityyli, riviväli ja värien kontrasti vaikuttavat erityisesti tekstien luettavuuteen. Sisällön sijoittelu ja graafisten elementtien asiaankuuluvuus vaikuttavat koko sivun luettavuuteen. [2]

Yhteenvetona sisällön haasteista voidaan todeta, että haasteet liittyvät sisällön muotoiluun ja laatuun. Sivulla olevan tiedon puuttuminen, virheellisyys, vanhentuminen ja epäoleellisuus vaikuttavat negatiivisesti sisällön laatuun. Sisällön erotteleminen vaikuttaa myös käytettävyyteen. Käytettävyys heikkenee, jos sivun sisältö ei ole aseteltu käyttäjälle ymmärrettävään muotoon. Sisällön ymmärrettävyyteen taas vaikuttaa luettavuus.

### 3.3 Suorituskyvyn haasteet

Yksi käytettävyyden tekijöistä on suorituskyky. Sovelluksen hidas lataaminen on käyttäjälle huomattavin seuraus huonosta suorituskyvystä. Hitaan sovelluksen käyttäminen on turhauttavaa ja voi saada käyttäjän lopettamaan sovelluksen käyttämisen.

Basrahin yliopiston käytettävyytutkimuksessa löytyi 198 haastetta, jotka luokiteltiin suorituskyvyn haasteiksi. Nämä haasteet jaettiin neljään ryhmään: *Vieraiden kielten puuttuminen, avattujen ja avaamattomien sivujen erottelu, heikko suorituskyky ja puuttuvat yhteydenottotiedot*. Koska Basrahin yliopisto sijaitsee Irakissa, on yliopiston kotisivu arabiaksi. Sivun ei tue muita kieliä, kuten englantia, mikä tekee käyttämisen arabiaa puhumattomille erittäin vaikeaksi. Avattujen ja avaamattomien sivujen erottelu liittyy enemmän käyttöliittymään, koska avatut ja avaamattomat

sivut erotellaan vaihtamalla avatun linkin väri käyttöliittymässä. Linkkien erottelu luokitellaan kuitenkin suorituskyvyn haasteeksi, koska sivujen turha uudelleenavaaminen aiheuttaa ylimääräistä kuormitusta sovellukselle ja vaikuttaa siten enemmän suorituskykyyn. Yleisellä heikolla suorituskyvyllä viitataan sovelluksen rakentamiseen, jossa tarvitaan monta askelta ennen kuin saavutetaan haluttua tietoa. [10]

Avattujen ja avaamattomien sivujen erottelu liittyy enemmän käyttöliittymään, koska avatut ja avaamattomat sivut erotellaan vaihtamalla linkin väri. Linkkien erottelun voisi tästä johtuen luokitella myös navigoinnin haasteeksi, mutta sivujen turha uudelleenavaaminen aiheuttaa ylimääräistä kuormitusta sovellukselle ja vaikuttaa siten enemmän suorituskykyyn.

Tom Brinckin kirjassa *Usability for the Web* [12] avataan haasteita web-sovelluksen suorituskyvyille. Huono vasteaika, eli sovelluksen hidas reagointi käyttäjän syötteeseen, on haaste suorituskyvyille. Sivun vasteaika on aika käyttäjän syötteestä sivun vastaamiseen. Vasteaika koostuu tiedonsiirron ajasta, palvelimen prosessoinnin viemästä ajasta ja selaimelta sivun esittämiseen kuluvasta ajasta. Jos yhdessä näistä osa-alueista on ongelma, kasvaa koko sovelluksen vasteaika. Graafisten elementtien liiallinen ja väärä käyttö aiheuttavat sovelluksen vasteajan kasvamista, koska selaimen pitää ladata kaikki sivun elementit ennen kun sivu on valmis käyttäjälle. [12]

Monialustaisten sovellusten käytettävyyttä mittaavassa tutkimuksessa arvioitiin käytettävyyttä tehokkuuden, suorituskyvyn ja tyydyttävyyden kohentamisen näkökulmasta. [13] Tutkimus suoritettiin palkkaamalla käyttäjiä testaamaan sovellusta ja keräämällä heiltä palautetta. Tutkimuksessa löydetty käytettävyyshaasteet jaettiin tehokkuuden, suorituskyvyn ja tyydyttävyyden mukaan. Suorituskyvyn haasteet liittyivät käyttäjien hämmennykseen. Käyttäjät eivät tieneet miten tietoa pitäisi syöttää, koska kysytyn tiedon nimi ja muoto olivat epäselviä. Valikoiden nimet olivat myös virheellisiä, mikä sekoitti käyttäjiä. Käyttäjien sekaantumisesta johtuen

sovelluksen käyttämiseen kuluu enemmän aikaa ja suorituskyky on siten huonompi. [13]

Iteratiivisen käytettävyydestaamisen vaikutusta valmiin sovelluksen käytettävyyteen selvitettiin tutkimuksella kehityksen yhteydessä. Testaus toteutettiin neljänä iteraationa osana sovelluksen kehitystä. Ensimmäisissä iteraatioissa testeihin osallistui asiantuntijoita, mutta myöhemmissä iteraatioissa osallistujat olivat pääsääntöisesti aloittelijoita. Tutkimuksen tuloksena iteratiivinen testaaminen todettiin hyödylliseksi ja nostettiin kehityksiä, joita voi tehdä seuraavassa tutkimuksessa. [14]

Tutkimuksessa löytyi suorituskykyyn vaikuttavia haasteita. Varsinkin kahdessa ensimmäisessä iteraatioissa vakavimmat ongelmat liittyvät käyttäjän hämmennykseen. Ensimmäisessä iteraatioissa käyttäjiltä puuttui selvät ohjeet, joiden mukaan sovellusta tulisi käyttää. Sivut sisälsivät ammattikieltä ja huonosti ilmaistuja lauseita ja kuvakkeet olivat epäselviä. Toisessa iteraatioissa käyttäjiltä puuttui tarvittava ohjeistus tiettyjen tehtävien suorittamiseen ja nimikkeet olivat epäselviä. Nämä tekijät tekevät sovelluksesta epäselvän käyttäjälle ja hidastavat tehtävien suorittamista. [14] Suorituskykyhaasteiden lisäksi tästä voidaan päätellä sovelluksen opittavuuden olevan melko heikko, koska käyttäjillä oli ongelmia sovelluksen käytön oppimisessa.

Suuri osa suorituskyvyn haasteista liittyy sovelluksen käyttöliittymään ja rakenteeseen. Suoraan näistä johtuvia haasteita suorituskyvylle ovat graafisten elementtien väärä käyttö, liian monivaiheinen rakenne ja avattujen ja avaamattomien sivujen erottelu. Käyttöliittymän käytettävyyden heikkenemisen lisäksi myös suorituskyky kärsii käyttöliittymän ja rakenteen virheistä. Käyttäjien sekaantuminen ja käyttöliittymän huono ymmärtäminen heikentävät myös suorituskykyä. Suorituskyky heikkenee myös suuresti, jos sovellus ei tue käyttäjän osaamia kieliä.

### 3.4 Tyydyttävyyden haasteet

Tyydyttävyyttä heikentävät ongelmat tekevät sovelluksen käyttämisestä epämiellyttävää. Käyttäjä voi lopettaa sovelluksen käyttämisen, jos hän kokee sen liian epätydyttäväksi. Tyydyttävyys sitoutuu muihin edellä mainittuihin kappaleisiin, koska käyttöliittymän, sisällön tai suorituskyvyn ollessa huonoja, ei sovelluksen käyttäminen ole kovin miellyttävää. Tyydyttävyyteen vaikuttaa eniten käyttäjien kokemukset eikä niin paljon sovelluksen tekninen toteutus.

Monialustaisten sovellusten käytettävyyttä mittaavassa tutkimuksessa tyydyttävyyteen luokitellut käyttäjäpalautteet oli kerätty kyselylomakkeen avulla. Käyttäjien mielestä valikoissa oleva tieto oli epäselvää ja huonosti sanoitettua. Tämän lisäksi joidenkin käyttäjien mielestä käyttöliittymä on liian tekninen, kun taas toisten käyttäjien mielestä käyttöliittymä on liian yksinkertainen. [13] Tyydyttävyys on siis hyvin subjektiivista, koska sama käyttöliittymä voi olla jollekin käyttäjälle aivan liian tekninen ja toiselle yksinkertainen. Sovelluksen kehityksessä ja käytettävyyden mittaamisessa onkin tärkeä huomioida sovelluksen tulevat käyttäjät. Tästä johtuen tulevien käyttäjien oletettu tekninen osaaminen ja yleiset mieltymykset tulee huomioida kehityksessä.

### 3.5 Kehityksen haasteet

Kehitykseen liittyvät tekijät voivat myös vaikuttaa sovelluksen käytettävyyteen. Haasteet vaikuttavat sovelluksen ylläpitämiseen kuten mahdollisiin päivityksiin. Sovelluksen koodi harvoin pysyy täysin samana, vaan vaatii korjauksia ja päivityksiä. Käytettävyyden ylläpitämisen kannalta on edun mukaista, että päivityksen saadaan julkaistua mahdollisimman nopeasti ja virheettömästi.

Web-sovelluksen tuotannossa on yleisesti mukana monta ohjelmoijaa samaan aikaan. Eri ohjelmoijien työtyyleissä voi olla eroja ja he voivat käyttää esimerkiksi eri

tietorakenteita. Ohjelman laajentuessa näistä eroavaisuuksista voi tulla haasteita ylläpitämisen kannalta. Sovelluksen elementtien päivittäminen tulee paljon monimutkaisemmaksi, jos koodi ei ole yhdenmukaista. [12]

Huono dokumentaatio on käytettävyyshaaste sovelluksen ylläpitäjälle. Jos sovellusta ylläpitää eri taho kuin sovelluksen kehittäjät, on dokumentaatio tärkeä osa ylläpitäjien ymmärtämistä sovelluksen toiminnasta. Sovelluksen toimiessa niin kuin pitää, ei huono dokumentaatio välttämättä vaikuta juuri sovelluksen ylläpitoon, mutta mahdollisen virheen korjaamisessa tai sovelluksen päivittämisessä dokumentaatio on tärkeä apu ylläpitäjälle. [12]

Kehityksen haasteilla ei ole suoraa vaikutusta käyttäjään, mutta välillisiä vaikutuksia sovelluksen käytettävyyteen voi esiintyä, jos esimerkiksi sovelluksen päivittäminen vaikeutuu huomattavasti. Käytettävyyteen liittyvien virheiden tunnistaminen ei paranna käytettävyyttä, jos ongelmia ei pystytä korjaamaan.

## 3.6 Haasteiden ehkäisy

Basrahin yliopiston tutkimuksessa [10] pääteltiin, että web-sovelluksen suunnittelijoiden kannattaa huomioida tiettyjä peruseriaatteita kehityksen aikana. Tutkimuksessa listattiin periaatteita, joiden avulla akateemisen verkkosivun kehittäjät pystyvät tekemään sovelluksia paremmalla käytettävyydellä. Tutkimuksessa löydettyt käytettävyysongelmat voivat myös auttaa tutkimuksen kohteena olleen sovelluksen kehittäjiä käytettävyyden parantamisessa.

Luettavuuden haasteita esittelevässä tutkimuksessa listataan konkreettiset korjaukset joillekin mainituista haasteista. Hyvän luettavuuden voi varmistaa, kun fonttikoko ei ole liian pieni, riviväli on tarpeeksi suuri, fonttityyli on selkeä, lauseet eivät ole liian pitkiä ja kuvat ovat relevantteja. Esitetyt korjaukset ovat selkeitä korjauksia mainituille haasteille, koska haasteiden tunnistamisen jälkeen korjaukset ovat hyvin yksinkertaisia. [2]

Myös monialustaisten sovellusten käytettävyyttä mittaavassa tutkimuksessa [13] pääteltiin, että käytettävyyttä voidaan parantaa korjaamalla tutkimuksessa mainitut haasteet. Korjauksien oletetaan parantavan sovelluksen tehokkuutta, suorituskykyä ja tyydyttävyyttä.

Iteratiivinen käytettävyytstudkimus paransi sovelluksen käytettävyyttä jokaisella iteraatiolla. [14] Iteratiivisen testaamisen avulla voidaan ehkäistä käytettävyyshaasteita ja tunnistaa ongelmat käytettävyydessä kehityksen aikana. Iteratiivinen testaaminen kuluttaa kuitenkin resursseja kehitysvaiheessa, mutta korjaukset kehitysvaiheessa voivat säästää suuremmalta määrältä korjaamista ja testaamista kehityksen loppuvaiheessa.

Käytettävyyshaasteita voidaan ehkäistä tunnistamalla yleisimmät käytettävyyshaasteet kehityksen alkaessa. Kun yleisimmät haasteet tunnistetaan, on niihin helppompi kiinnittää huomiota kehityksen aikana. Jos käytettävyyshaasteet eivät ole tiedossa, voi tutkimuksen yhteydessä suorittaa iteratiivista käytettävyytstudkimusta. Käytettävyytstudkimuksesta saadaan tietoa kehitysvaiheessa olevan sovelluksen käytettävyyden ongelmista ja ne voidaan korjata välittömästi. Käytettävyyttestaaminen on muutenkin tärkeä osa käytettävyyden parantamista, koska haasteiden tunnistamisen jälkeen korjaaminen ei ole yleisesti kovin monimutkaista.

## 4 Yhteenveto

Tutkimuksen tarkoituksena oli selvittää web-sovellusten käytettävyyshaasteita ja niiden ratkaisuja. Tutkimuksessa pyrittiin vastaamaan kahteen tutkimuskysymykseen. Tutkimuskysymyksenä TK1 oli ”*Mitä web-sovellusten käytettävyyshaasteita on olemassa?*” Tutkimuksessa selvisi, että web-sovellusten käytettävyyshaasteet voidaan jakaa viiteen pääkategoriaan: Käyttöliittymän haasteet, sisällön haasteet, suorituskyvyn haasteet, tyydyttävyyden haasteet ja kehityksen haasteet.

Tutkimuskysymyksenä TK2 oli ”*Miten web-sovellusten käytettävyyshaasteita on pyritty ratkaisemaan?*” Käytettävyyshaasteiden ratkaisemisen avain on haasteiden tunnistamisessa. Moniin haasteista voidaan varautua kehityksessä ja ne voidaan korjata tunnistamalla. Esimerkiksi navigointiin liittyviä haasteita voidaan vähentää suunnittelemalla selkeitä ja johdonmukaisia rakenteita navigoinnille. Suorituskykyä voidaan parantaa optimoimalla sivujen rakenteita. Iteratiivinen käyttäjättestaus on keskeinen keino haasteiden tunnistamiselle ja korjaamiselle sovelluksen kehityksen aikana.

Tutkimuksen rajausta oli aika laajaa, koska kaikista mahdollisista käytettävyyshaasteista ei saa kovin tarkkaa kuvaa. Tutkimuksen olisi voinut rajata yhteen tai kahteen kategoriaan, joka olisi mahdollistanut kategorioiden yksityiskohtaisemman käsittelyn. Laajasti kaikkien kategorioiden käsittely mahdollisti kuitenkin hyvän yleiskuvan käytettävyyshaasteista.

Jatkossa web-sovellusten käytettävyyttä voitaisiin parantaa keskittymällä tar-

kemmin käyttäjien tarpeisiin. Tulevaisuuden tutkimukset voisivat keskittyä tekoälyn ja koneoppimisen hyödyntämiseen käytettävyyden parantamisessa, esimerkiksi käyttäjätestauksen tukena. Tulevaisuuden tutkimukset voisivat myös tutkia yksityiskohtaisemmin esitettyjä käytettävyyshaasteita laajemman yleiskuvan sijaan.

Tutkimus osoittaa, että käytettävyyshaasteiden ymmärtäminen ja aktiivinen testaaminen ovat tärkeä osa web-sovelluksen kehitystä. Tutkimuksessa on listattu käytettävyyshaasteita, mutta tulevaisuudessa web-sovellukset voivat sisältää täysin uusia käytettävyyshaasteita.

# Lähdeluettelo

- [1] A. Seffah, M. Donyaee, R. B. Kline ja H. K. Padda, ”Usability measurement and metrics: A consolidated model”, *Software Quality Journal*, vol. 14, nro 2, s. 159–178, kesäkuu 2006, ISSN: 0963-9314, 1573-1367. DOI: 10.1007/s11219-006-7600-8.
- [2] E. Elahi, A. Maqueda ja J. Morato, ”Web Readability Challenges”, teoksessa tammikuu 2023, s. 446–454, ISBN: 978-3-031-21437-0. DOI: 10.1007/978-3-031-21438-7\_35.
- [3] International Organization for Standardization. ”ISO 9241-11:2018 Ergonomics of Human-System Interaction – Part 11: Usability – Definitions and Concepts”. (2018), url: <https://www.iso.org/standard/63500.html> (viitattu 24.04.2025).
- [4] K. Hornbæk, ”Current practice in measuring usability: Challenges to usability studies and research”, *International Journal of Human-Computer Studies*, vol. 64, nro 2, s. 79–102, helmikuu 2006, ISSN: 10715819. DOI: 10.1016/j.ijhcs.2005.06.002.
- [5] J. Nielsen, *Usability Engineering*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1994, ISBN: 9780080520292.
- [6] M. Jazayeri, ”Some trends in web application development”, teoksessa *Future of Software Engineering (FOSE '07)*, Minneapolis, MN, USA: IEEE, toukokuu 2007, s. 199–213, ISBN: 978-0-7695-2829-8. DOI: 10.1109/FOSE.2007.26.

- 
- [7] X. Liu, J. Heo ja L. Sha, ”Modeling 3-tiered Web applications”, teoksessa *13th IEEE International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems*, ISSN: 2375-0227, syyskuu 2005, s. 307–310. DOI: 10.1109/MASCOTS.2005.40.
- [8] H. M. Abdullah ja A. M. Zeki, ”Frontend and backend web technologies in social networking sites: Facebook as an example”, teoksessa *2014 3rd International Conference on Advanced Computer Science Applications and Technologies*, Amman, Jordan: IEEE, joulukuu 2014, s. 85–89, ISBN: 978-1-4799-1845-4. DOI: 10.1109/ACSAT.2014.22.
- [9] D. Flanagan, *JavaScript: The Definitive Guide*, 7. painos. O’Reilly Media, 2020, ISBN: 978-1-491-95202-3.
- [10] Z. Alfayez ja H. Al-Tawriy, ”Usability Evaluation of Educational Websites: A case Study of University of Basrah Website”, *Solid State Technology*, vol. 63, s. 9865–9886, 15. tammikuuta 2021.
- [11] R. Otaiza, C. Rusu ja S. Roncagliolo, ”Evaluating the usability of transactional web sites”, teoksessa *2010 Third International Conference on Advances in Computer-Human Interactions*, St Maarten, Netherlands Antilles: IEEE, 2010, s. 32–37, ISBN: 978-1-4244-5693-2. DOI: 10.1109/ACHI.2010.27.
- [12] T. Brinck, D. Gergle ja S. d. Wood, ”Usability in Software Development”, teoksessa *Usability for the Web*, sarja Interactive Technologies, T. Brinck, D. Gergle ja S. d. Wood, toim., San Francisco: Morgan Kaufmann, 2002, s. 339–364, ISBN: 978-1-55860-658-6. DOI: 10.1016/B978-155860658-6.50012-2.
- [13] N. A. Nik Ahmad, N. I. M. Hamid ja A. Mohd Lokman, ”Performing usability evaluation on multi-platform based application for efficiency, effectiveness, and satisfaction enhancement”, *International Journal of Interactive Mobile Tech-*

*nologies (iJIM)*, vol. 15, nro 10, s. 103, 25. toukokuuta 2021, ISSN: 1865-7923.

DOI: 10.3991/ijim.v15i10.20429.

- [14] J. C. Romano Bergstrom, E. L. Olmsted-Hawala, J. M. Chen ja E. D. Murphy, "Conducting iterative usability testing on a web site: challenges and benefits", *J. Usability Studies*, vol. 7, nro 1, s. 9–30, marraskuu 2011.