



Android-sovelluksen käyttäjäkokemuksen (UX) kehittäminen

Mikko Myllys

OPINNÄYTETYÖ
Huhtikuu 2019

Tieto- ja viestintäteknikka
Ohjelmistotekniikka

TIIVISTELMÄ

Tampereen ammattikorkeakoulu
Tieto- ja viestintätekniikan koulutus
Ohjelmistotekniikka

MYLLYS, MIKKO:

Android-sovelluksen käyttäjäkokemuksen (UX) kehittäminen

Opinnäytetyö 57 sivua, joista liitteitä 12 sivua
Huhtikuu 2019

Tämän opinnäytetyön tavoitteena oli tutkia käyttäjäkokemuksen ja sen mittaamisen teoriaa, jota sovellettiin opinnäytetyön kohteena olevan Android-aloitusnäyttösovelluksen arviointiin ja testaamiseen. Arviointimenetelmänä käytettiin asiantuntija-arviointia, joka pohjautui heuristisiin periaatteisiin. Käytettävyyden testauksessa käytettiin käyttäjätestausta. Tarkoituksena oli tuottaa dokumentti, joka tarjoaa ratkaisuja sovelluksesta löytyviin käytettävyyssongelmiin ja jota voidaan käyttää apuna sovelluksen käyttäjäkokemuksen kehittämiseen tulevaisuudessa. Työn toimeksiantajana oli Jidoka Technologies Oy ja asiakkaana Kuntotimpurit, jolle sovellus oli alun perin kehitetty.

Tuloksena löydettiin yhteensä noin 80 käytettävyyssongelmaa, jotka lajiteltiin vakavuusasteen perusteella luokkiin. Yli puolet ongelmista sijoittui vakavuusasteikon vakavammalle puolelle, mikä kertoo testien tuoneen onnistuneesti esille vakaviakin käytettävyyssongelmia. Vakavimmille käytettävyyssongelmille kehitettiin ratkaisuehdotuksia ja niiden avulla kehitettiin kehitysideoita sovelluksen jatkokkehitystä varten.

Saatujen tulosten perusteella huomattiin sovelluksen käytettävyydessä paljon parannettavaa. Parhaiten sovelluksen käyttäjäkokemusta voidaan kehittää keskittymällä vakavimpien ongelmien korjaamiseen ensimmäisenä, jonka jälkeen sovellusta voidaan arvioida uusiksi tarpeen vaatiessa. Tämän dokumentin tulokset tarjoavat kaikki tarpeelliset kehitysehdotukset käyttäjäkokemuksen parantamista varten kyseisessä aloitusnäyttösovelluksessa.

ABSTRACT

Tampereen ammattikorkeakoulu
Tampere University of Applied Sciences
Degree Programme in ICT Engineering
Software Engineering

MYLLYS, MIKKO:
User Experience (UX) Development in Android Application

Bachelor's thesis 57 pages, appendices 12 pages
April 2019

The purpose of this thesis was to study the theory of user experience and its measurement techniques which were then used to evaluate and to test a real Android launcher application. The expert evaluation was based on heuristic principles. It was used to measure the usability of the application from the Android developers' perspective. After that the application was tested on real users. The objective was to produce a document that provides solutions to the usability problems found in the application. The results can be used to develop the application's user experience in the future. The project was commissioned by Jidoka Technologies Oy and the client was Kuntotimpurit.

About 80 usability problems were found during the evaluation. The problems were sorted into different classes based on the severity of the problem. More than half of the problems were severe ones which indicates that the tests successfully brought up some serious usability problems. Solutions to the most severe usability problems were developed. Based on these solutions a set of improvement plans was created for further development of the application.

The findings indicate that the application has a lot to improve on the user experience. The best way to improve the overall user experience is to fix the most severe usability problems first. After taking care of the most severe problems the application can be then evaluated and tested again. The results of this document provide all the necessary development ideas for improving the user experience in this specific launcher application.

Key words: user experience, usability, android, evaluation, testing

SISÄLLYS

1	JOHDANTO	6
2	TARKOITUS JA TAVOITTEET	8
3	KÄYTTÄJÄKOKEMUS JA KÄYTETTÄVYYS	9
	3.1 Käyttäjäkokemus	9
	3.2 Käytettävyys ja sen merkitys	10
4	ANDROIDIN KÄYTETTÄVYYSSUUNNITTELUPERIAATTEET	12
	4.1 Android kehitys- ja käyttöympäristönä	12
	4.2 Saavutettavuus	12
	4.2.1 Avustava teknologia	13
	4.2.2 Sovelluksen rakenne	13
	4.2.3 Väri ja kontrasti	14
	4.2.4 Ulkoasu ja typografia	15
	4.2.5 Kirjoitus	15
	4.2.6 Ääni ja liike	16
5	KÄYTETTÄVYYDEN ARVIOIMINEN	17
	5.1 Heuristiikka ja arviointimenetelmät	17
	5.2 Shneidermanin kahdeksan kultaista sääntöä	19
	5.2.1 Pyri johdonmukaisuuteen	19
	5.2.2 Hae yleistä käytettävyyttä	20
	5.2.3 Anna selkeää palautetta	20
	5.2.4 Suunnittele dialogit johtamaan lopputulokseen	21
	5.2.5 Tarjoa yksinkertainen virheenkäsittely	21
	5.2.6 Salli helppo toimintojen peruminen	22
	5.2.7 Tue käyttäjän hallintakäsitystä	22
	5.2.8 Vähennä lyhytkestoisien muistin kuormitusta	22
	5.3 Nielsenin heuristiset periaatteet	23
	5.3.1 Järjestelmän tilan näkyvyys	23
	5.3.2 Tuotteen ja tosielämän vastaavuus	24
	5.3.3 Käyttäjän kontrolli ja vapaus	25
	5.3.4 Yhdenmukaisuus	25
	5.3.5 Virheiden ehkäisy	26
	5.3.6 Tunnistaminen mieluummin kuin muistaminen	26
	5.3.7 Joustavuus ja tehokkuus	27

5.3.8	Estetiikka ja minimalistinen suunnittelu	27
5.3.9	Virhetilanteista palautuminen	28
5.3.10	Opastus ja ohjeistus.....	28
6	ALOITUSNÄYTTÖSOVELLUKSEN ARVIOINTI JA TESTAUS	29
6.1	Asiantuntija-arvioinnin järjestelyt	29
6.2	Asiantuntija-arvioinnin toteutus	29
6.3	Käyttäjätestauksen järjestelyt.....	30
6.4	Käyttäjätestauksen toteutus	31
7	TULOKSET	37
7.1	Testikäyttäjien palaute	37
7.2	Asiantuntija-arvioinnin käytettävyysongelmat.....	37
7.3	Käyttäjätestauksen käytettävyysongelmat	40
7.4	Ratkaisuehdotukset.....	41
8	POHDINTA	43
	LÄHTEET	44
	LIITTEET	46
	Liite 1. Asiantuntija-arvioinnissa käytetty lomake	46
	Liite 2. Käyttäjätestauksessa käytetty lomake.....	53

1 JOHDANTO

Käyttäjäkokemus terminä nousee nykyään pinnalle hyvin usein teknologiasta puhuttaessa. Järjestelmät keskittyvät yhä enemmän ihmisten ympärille ihmisten sekä tietokoneiden välisen vuorovaikutuksen lisääntyessä. Teknologian tarkoitus on parantaa elämänlaatua ja tuoda arkipäiväisiin askareisiin jotain lisäarvoa. Hyvä käyttäjäkokemus mahdollistaa teknologian viemisen tähän suuntaan. Tuotteen vastatessa käyttäjälle tämän haluamalla tavalla ja ratkaistessa käyttäjältä löytyviä ongelmia kasvaa käyttäjän tyytyväisyys, mikä muodostaa käyttäjälle positiivisia kokemuksia tuotteen käytöstä. Hyvä käyttäjäkokemus saa henkilön todennäköisemmin käyttämään tuotetta uudestaan jatkossakin. (Shape 2018.)

Tämän opinnäytetyön tarkoituksena on tutkia käyttäjäkokemusta ja soveltaa käyttäjäkokemuksen arviointi- ja testausmenetelmiä asiakkaalle kehitettyyn Android-sovellukseen. Asiakkaan sovelluksesta yritetään löytää mahdollisimman kattavasti käytettävyyso ongelmia, jonka jälkeen ongelmat käsitellään ja niille kehitetään ratkaisuehdotuksia. Tavoitteena on muodostaa dokumentti, jota voidaan käyttää hyväksi sovelluksen mahdollisessa jatkokehityksessä.

Toimeksiantajana opinnäytetyölle oli Jidoka Technologies Oy, jossa opinnäytetyön tekijä oli työskennellyt jo reilut puoli vuotta ennen opinnäytetyön aloittamista. Jidoka on erikoistunut IoT-sovelluksiin, etäohjattaviin näyttöratkaisuihin, Web-to-teutuksiin ja mobiilikehitykseen. Opinnäytetyön aikana toimivalle tutkimus ja tuotekehitys -tiimille projekti oli ensimmäinen Android-asiakasprojekti laatuaan.

Tarve opinnäytetyölle syntyi, kun aiemmin asiakkaalle kehitettyä sovellusta päätettiin lähteä uudistamaan ykkösprioriteetin kohdistuessa sovelluksen käytettävyyteen. Sovellus oli ollut asiakkaalla käytössä jo useamman kuukauden ajan kesästä 2018 lähtien. Kehitystyö oli kuitenkin vielä kesken ja seuraavaksi kehityksen kohteeksi valitui sovelluksen käyttäjäkokemus.

Työn toisessa luvussa käydään tarkemmin läpi tutkimuksen tarkoitusta ja tavoitteita. Kolmannessa luvussa perehdytään käyttäjäkokemuksen teoriaan ja luvussa neljä käytettävyyttä käsitellään juuri Android-ympäristön kannalta. Luvussa

viisi esitetään erilaisia tapoja käytettävyyden mittaamiseen ja arviointiin, jonka jälkeen luvussa kuusi kerrotaan tätä tietoa hyödyntämällä tehtyjen tutkimusten etenemisestä. Luvussa seitsemän kootaan lopulta tutkimusten tulokset yhteen.

2 TARKOITUS JA TAVOITTEET

Opinnäytetyössä käsiteltävä aloitusnäyttösovellus sai alkunsa kesällä 2018, kun toimeksiantaja Jidoka sai asiakasprojektin Kuntotimpureilta. Kuntotimpurit on kehittänyt elämyspalvelun Iloi.fi, joka tuottaa liikuntaa, kulttuuria ja luontoelämyksiä palvelutaloille ja hoivakodeille verkossa. Projektin tarkoitus oli luoda isokokoisilla Android-taulutietokoneilla toimiva aloitusnäyttösovellus, johon olisi yhdistetty elämyspalvelun verkossa tarjoamat palvelut sekä haluttuja Android-sovelluksia ja toiminnallisuuksia. Taulutietokoneiden avulla hoivakotien ja palvelutalojen asukit ja hoitohenkilökunta pääsevät käsiksi elämyspankin tarjoamiin elämyksiin nopeasti ja vaivatta.

Sovelluksesta saatiin kehitettyä demoversio, joka annettiin asiakkaan käyttöön kesän 2018 lopussa. Asiakkaan kanssa tehtiin tiivistä yhteistyötä sovelluksesta käytön aikana löytyneiden ongelmien korjaamiseen ja uusien toiminnallisuuksien kehittämiseen. Kehitystyö jatkui läpi syksyn. Vuoden 2018 lopussa opinnäytetöiden aiheita miettiessä Jidokalla päädyttiin ideaan lähteä tutkimaan Kuntotimpureille luodun aloitusnäyttösovelluksen kehittämistä erityisesti käyttäjäkokemuksen kannalta. Koska sovelluksen käyttäjäryhmänä oli pääasiassa vanhempia ihmisiä ja hoitohenkilökuntaa, pitäisi sovelluksessa kehittää erityisesti käytettävyyttä ja helppokäyttöisyyttä, jotta henkilöt, joilta ei välttämättä löydy kokemusta tietoteknisten laitteiden käytöstä, pystyvät käyttämään tuotetta vaivattomasti.

Opinnäytetyön aihe rajattiin lopulta käsittelemään sovelluksen kehitystyön eri alueista käyttäjäkokemusta. Tavoitteena oli perehtyä käyttäjäkokemukseen ja sen arviointimenetelmiin. Tämän jälkeen arviointi- ja testausmenetelmiä sovellettaisiin aloitusnäyttösovellukseen ja arviointi- sekä testitulosten pohjalta luotaisiin löydettyihin käytettävyysongelmiin ratkaisuja ja kehitysehdotuksia. Opinnäytetyön tulosten on tarkoitus toimia apuna käyttäjäkokemuksen parantamisessa kehitystyön jatkuessa.

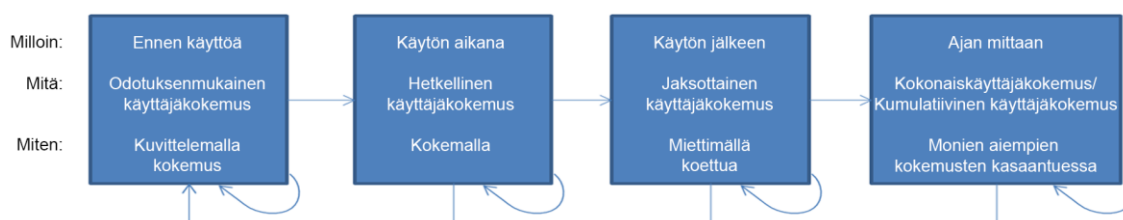
3 KÄYTTÄJÄKOKEMUS JA KÄYTETTÄVYYS

3.1 Käyttäjäkokemus

Käyttäjäkokemus käsitteenä on erittäin laaja. Se sisältää monia osa-alueita, kuten visuaalisen suunnittelun, käytettävyyden, estetiikan ja psykologisen sekä sosiologisen tutkimuksen. Kokemus ei rajoitu pelkästään tuotteen käyttämiseen, vaan se rakentuu kaikista tuotteen elämänkaaren vaiheista, joissa käyttäjä on jotenkin mukana. Näitä vaiheita ovat mm. tuotteen mainoskampanjat ja asiakastuki. Käyttäjäkokemusta ei voida määritellä yleisellä tasolla, vaan se muuttuu ympäristön ja olosuhteiden vaikutuksesta. Koska kokemus on riippuvainen käyttäjän tunteista, on hyvä muistaa, että suunnittelun ja kehityksen keskipisteenä tulee olla käyttäjä itse, ei sovellus tai sen kehittäjä. (Interaction Design Foundation n.d.)

On hyvin yleistä, että käyttäjäkokemuksen ja käyttöliittymän merkitys sekoittuu puhekielessä. On totta, että monia samoja asioita on otettava huomioon molempia suunniteltaessa, mutta täytyy kuitenkin muistaa, että termit eivät tarkoita samaa. Käyttöliittymä keskittyy tuotteen ulkonäköön ja tekniseen puoleen, kun taas käyttäjäkokemus on kokonaisvaltaisempi elämys sisältäen myös käyttöliittymän. Käyttäjäkokemus keskittyy enemmän abstrakteihin osa-alueisiin, kuten vuorovaikutukseen ja tunteisiin. (UserTesting 2018.)

Koska käyttäjäkokemus on sidoksissa aikaan, voidaan kokonaiskäyttäjäkokemus jakaa osiin kuvion 1 mukaisesti. Käyttäjäkokemus rakentuu käyttöhetkellä syntyvien kokemusten lisäksi myös epäsuorista kokemuksista, jotka vaikuttavat käyttäjän näkemykseen tuotteesta. Näitä kokemuksia käyttäjälle kehittyy käyttöhetkeä ennen ja sen jälkeen. Ne voivat olla esimerkiksi muiden käyttäjien mielipiteitä ja omia käytönjälkeisiä kokemuksia. (Roto, Law, Vermeeren & Hoonhout 2011.)



KUVIO 1. Käyttäjäkokemuksen ajanjaksot nimineen ja tapahtumineen (Roto ym. 2011, muokattu)

Odotuksenmukainen käyttäjäkokemus muodostuu, kun käyttäjä kuvittelee käyttävänsä tuotetta. Tämä kokemus voi syntyä ennen ensimmäistä käyttökertaa, käytön aikana tai sen jälkeen. Hetkellinen käyttäjäkokemus taas muodostuu käytön aikana, kun käyttäjä on suorassa vuorovaikutuksessa tuotteen kanssa. Tätä seuraava kokemus on jaksottainen käyttäjäkokemus, joka muodostuu jokaisen käyttökerran jälkeen käyttäjän miettiessä miltä tuotteen käyttäminen tuntui. Kokonaiskäyttäjäkokemus tai kumulatiivinen käyttäjäkokemus sisältää kaikkia edellä mainittuja tyyppisiä ja se alkaa heti käyttäjän saatua mitä tahansa informaatiota tuotteesta ensimmäistä kertaa. (Roto ym. 2011.)

3.2 Käytettävyys ja sen merkitys

Käytettävyys on laatuattribuutti, eli järjestelmän toimintakykyä kuvaava ei-toiminnallinen vaatimus. Hyvä käytettävyys tarkoittaa, että tuote on helppokäyttöinen ja käyttäjä pystyy käyttämään tuotetta tehokkaasti. Käytettävyysasiantuntija Jakob Nielsenin mukaan käytettävyys koostuu viidestä eri osasta: opittavuudesta, tehokkuudesta, muistettavuudesta, virheettömyydestä ja miellyttävyydestä. Opittavuus kertoo siitä, kuinka nopeasti käyttäjä oppii käyttämään sovellusta. Mitä paremmin käyttäjä sisäistää tuotteen toiminnan jo ensimmäisten käyttökertojen aikana, sitä parempi on tuotteen opittavuus. Tehokkuus kuvastaa jonkin tietyn tehtävän suorittamiseen käytettyä aikaa. Muistettavuus määritellään sen perusteella, miten hyvin käyttäjä muistaa ohjelman toiminnan palattuaan käyttämään sitä tauon jälkeen. Virheettömyys koostuu järjestelmän teknisistä ongelmista ja käyttäjän tekemistä virheistä. On otettava huomioon paljonko käyttäjä voi tehdä virheitä, ovatko virheet vakavia ja pystyykö käyttäjä palautumaan niistä. Miellyttävyys kertoo, kuinka mieluisaksi käyttäjä tuntee tuotteen käytön. (Nielsen 2012.)

Käytettävyys ei yksinään vielä tee tuotteesta käytännöllistä vaan sen kanssa tulee miettiä myös hyödyllisyyttä. Jos käytettävyys määrittelee tuotteen helppokäyttöisyyden, niin hyödyllisyys määrittelee, tarjoaako tuote toiminnallisuudet käyttäjän tarpeisiin. Tuote voi olla helppo ja mukava käyttää, mutta jos se ei tarjoa käyttäjälleen tarvittavia toimintoja, on tuote epäkäytännöllinen, käyttäjä ei tarvitse sitä. Vastaavasti tuotteen tarjotessa tarvittavat toiminnot, mutta ollessa vaikeakäyttöinen, se on silti epäkäytännöllinen. Jos halutun toiminnon suorittaminen on niin hankalaa, että toimintoa ei saada suoritettua loppuun asti, ei käyttäjä hyödy siitä mitenkään. (Nielsen 2012.)

Jos käytettävyttä ei ole otettu huomioon tuotetta suunniteltaessa tarpeeksi hyvin, voi se pahimmillaan johtaa tuotteen hylkäämiseen. Epämiellyttävyyden ja turhautuneisuuden tunne voi johtua esimerkiksi pitkistä latausajoista tai epäselvästä käyttöliittymästä. Ensivaikutelma tuotteesta syntyy todella nopeasti ja sillä on suuri vaikutus siihen, haluaako käyttäjä lopulta käyttää tuotetta. Käytettävyyden ollessa hyvä voidaan säästää myös rahallisesti. Kun käyttäjän ei tarvitse ottaa jatkuvasti yhteyttä tuotetukeen, säästetään tukipalveluissa. Myös kouluttamisen tarve vähenee käytettävyyden parantuessa. (Segue Creative Team 2017.)

4 ANDROIDIN KÄYTETTÄVYSSUUNNITTELUPERIAATTEET

4.1 Android kehitys- ja käyttöympäristönä

Suunniteltaessa sovellusta Androidille tulee ottaa huomioon sovellusalustakohtaiset vaatimukset. Androidin versioissa on poikkeavuuksia mm. laiterajapinnan tarjoamien toimintojen osalta. Myös kohdelaitteet vaikuttavat suunniteluun, sillä esimerkiksi tabletille kehittäessä on otettava huomioon isompaan näyttöön liittyviä asioita, joita ei välttämättä tarvitse miettiä pienemmälle kännykälle tehtävässä sovelluksessa.

Android tarjoaa kattavan kokoelman suunnittelussa käytettäviä periaatteita ja käytäntöjä yleisellä tasolla, joiden avulla suunnitellusta sovelluksesta saadaan mahdollisimman käyttäjäystävällinen. Ohjeet opastavat kehittämään sovelluksen, jota pystyy käyttämään kuka tahansa riippumatta käyttäjästä. Ohjeiden noudattaminen mahdollistaa myös parhaan mahdollisen yhteensopivuuden, suorituskyvyn, tietoturvan ja paljon muuta Androidin ollessa sovellusalustana. Nämä ohjeet ovat kuitenkin tehty yleisiksi suuntaa antaviksi ohjeiksi ja on suositeltavaa, että niitä sovelletaan ja muutetaan aina sovelluskohtaisesti sovelluksen käyttäjien kohderyhmät huomioon ottaen. (Android Developers n.d.)

4.2 Saavutettavuus

Saavutettavuudella tarkoitetaan sovelluksen helppoa lähestyvyyttä käyttäjän rajoitteista riippumatta. Ihmiset poikkeavat toisistaan ja siksi on tärkeää huomioida, että kaikki käyttäjät pystyvät vuorovaikutukseen tuotteen kanssa yhtä helposti. Saavutettavuuden parantuessa tuote tavoittaa yhä useamman käyttäjän. Se ei helpota ainoastaan rajoittuneiden henkilöiden toimintaa vaan kaikkien sovelluksen käyttäjien toimintaa ja näin ollen parantaa kaikkien käyttäjien käyttäjäkokemusta. (Android Developers n.d.)

Jos käyttäjä on esimerkiksi värisokea, voi häneltä jäädä ymmärtämättä sovelluksessa väreillä viestityt asiat kokonaan tai jos käyttäjällä on lukivaikeus ei hän välttämättä saa selkeästäkään tekstistä mitään irti. Käyttäjän ei tarvitse edes olla rajoittunut, jotta hänellä voisi ilmetä hankaluuksia sovelluksen käytössä. Tällaisia tilanteita huomioiden Android tarjoaa ohjeita myös saavutettavuuden parantamiseksi.

4.2.1 Avustava teknologia

Avustava teknologia tarkoittaa laitteen mahdollistamia teknologioita, joilla voidaan mahdollistaa rajoittuneen käyttäjän toiminta sovelluksessa. Tiedon välittäminen käyttäjältä sovellukselle ja toisin päin voidaan hoitaa monella eri tapaa. Tekstin kokoa voidaan automaattisesti suurentaa, sovellus voi lukea sisällön käyttäjälle tai sisältöä voidaan lukea pistenäyttöä käyttäen. (Material Design n.d.)

Avustavan teknologian käyttö voi tulla hyödylliseksi tilanteessa, jossa käyttäjä ei pysty käyttämään käsiään esimerkiksi valmistaessaan ruokaa. Tällöin puheentunnistus toimii hyödyllisenä apuvälineenä ja käyttäjä voi antaa sovellukselle kommentoja puheen avulla ilman, että hänen täytyy koskettaa laitetta. Sovellus voidaan suunnitella käyttämään tekstistä puheeksi -synteesiä, jolla muutetaan kirjoitettu teksti puhuttuun muotoon ja sovellus pystyy näin myös vastaamalla käyttäjälle ilman, että käyttäjän tulee olla laitteen ääressä.

4.2.2 Sovelluksen rakenne

Eri elementtien ja toimintojen huoliteltu sijoittaminen sovelluksessa on tärkeää, jotta sovelluksessa liikkuminen olisi käyttäjälle mahdollisimman helppoa ja selkeää. Käyttäjän tulisi tietää koko ajan missä päin sovellusta hän on ja mikä sovelluksessa on tärkeää. Näitä elementtejä voidaan korostaa mm. liikkeillä, muodoilla, väreillä ja tekstillä. Seuraavaan aktiviteettiin siirtyminen ja sieltä poistuminen tulee olla tehty tarpeeksi selkeäksi ja liian moni päällekkäinen aktiviteetti voi tehdä sovelluksessa navigoimisesta sekavaa. (Material Design n.d.)

Hierarkia on tärkeä osa sovelluksen rakennetta suunniteltaessa. Käyttöliittymä kannattaa pitää mahdollisimman yksinkertaisena ja turhat elementit suositellaan karsittavaksi pois. Tärkeämmät toiminnot ja elementit sijoitetaan näytön yläreunaan, yhtä tärkeät rinnakkain ja vähemmän tärkeät alas. Näin pidetään tärkeysjärjestys selkeänä ja loogisena käyttöliittymässä. (Material Design n.d.)

Elementtejä kannattaa myös ryhmitellä samankaltaisten elementtien kanssa. Tiedon esittämisen järjestyksessä on hyvä miettiä mitkä elementit kiinnittävät ensimmäiseksi käyttäjän huomion ja elementin kadotessa näytöltä mihin huomio seuraavaksi kiinnittyy. Näin käyttäjän liikkumisesta sovelluksessa voidaan saada mahdollisimman sulavaa ja tieto saadaan luontevasti sovellusta selatessa. (Material Design n.d.)

4.2.3 Väri ja kontrasti

Värit esiintyvät hyvin tärkeässä osassa käyttöliittymän kannalta. Niillä pystytään korostamaan informaation tunnelmaa, sävyä ja tärkeyttä (Material Design n.d.). Hyvänä esimerkkinä perinteinen punainen hylkää-painike ja vihreä hyväksy-painike. Punaisella värillä viestitään yleensä negatiivista ja vihreällä värillä positiivista.

Kontrasti ilmoittaa, kuinka paljon kaksi väriä poikkeaa toisistaan. Jos kontrasti jää liian pieneksi, ei informaatio erotu välttämättä tarpeeksi selkeästi taustastaan ja sitä on hankala huomata. Näin voi käydä, kun vaalean taustan päälle asetetaan valkoista tekstiä ja tumman taustan päälle mustaa tekstiä. World Wide Web Consortium, joka ylläpitää WWW:n standardeja, suosittelee pienen tekstin kontrastisuhteeksi taustaan verrattuna 4.5:1 ja suuren tekstin kontrastisuhteeksi 3:1. Näin informaatio saadaan selkeästi näkyviin ja sitä on helpompi lukea. (Material Design n.d.)

4.2.4 Ulkoasu ja typografia

Interaktiiviset painikkeet ja kosketuksen rekisteröivät elementit tulee tehdä tarpeeksi suuriksi. Suositus on vähintään 48 x 48 dp:tä eli näytön tiheydestä riippumattonta pikseliä. Jos kuitenkin haluaa kohdistaa sovelluksen mahdollisimman laajalle käyttäjäkunnalle, kannattaa käyttää suurempia elementtejä sen ollessa mahdollista. (Material Design n.d.)

Sovelluksen ulkoasun on hyvä mukautua eri näyttöjen kokoihin. Jos tuotetta testataan vain isoilla näytöillä, voi joitain elementtejä leikkaantua tai piiloutua pienemmille näytöille siirryttäessä. Mukautuva ulkoasu on responsiivisen sovelluksen perusta. Myös tekstin on hyvä olla skaalautuva, sillä käyttäjä pystyy hyvin usein säätämään fonttikokoa laitteesta ja selaimesta. Tekstin kokoa määriteltäessä käytetäänkin näytön tiheydestä riippumattomien pikselien sijaan skaalautuvia pikseleitä. (Material Design n.d.)

4.2.5 Kirjoitus

Kirjoituksia lisätessä on hyvä ottaa huomioon, miten teksti käsitellään näytönluokhuohjelmilla. Näytönlukijat lukevat kaiken tekstin sivuilta, myös piilotetun tekstin. Piilotettu teksti voi olla esimerkiksi vaihtoehtoista tekstiä, mikä pidetään piilossa, kunnes elementtiä muutetaan toiminnoilla. Tekstin on hyvä olla lyhyttä, mutta kuvaavaa, jotta käyttäjä saa tekstin tarkoituksen nopeasti selville. (Material Design n.d.)

Elementtejä, kuten painikkeita, valintaruutuja ja valikkoja kannattaa nimetä niin, että käyttäjä tietää mitä siitä tapahtuu, ei miltä elementti näyttää. Jos elementillä on eri tiloja, sen tekstinä tulee näyttää juuri sen hetkinen tila. Elementin suorittama toiminto ilmoitetaan ilman ohjeita siitä, miten toiminto tulee aktivoida. Eri syötelaitteilla voidaan aktivoida sama toiminto eri tavoin, esimerkiksi kosketuksella aktivoitava painike voidaan aktivoida myös erillistä näppäimistöä käyttäen. (Material Design n.d.)

4.2.6 Ääni ja liike

Sovelluksen sisältäessä ääniä kannattaa miettiä, kuinka näitä ääniä voitaisiin ilmentää kuuroille tai heikommin kuuleville. Mahdollisia vaihtoehtoja ovat mm. tekstitykset, käännökset ja visuaaliset vihjeet. Turhien äänien toistamista, kuten taustamusiikki, tulisi välttää. Jos tällaisia ääniä kuitenkin halutaan käyttää, käyttäjän tulee pystyä pysäyttämään äänet halutessaan. (Material Design n.d.)

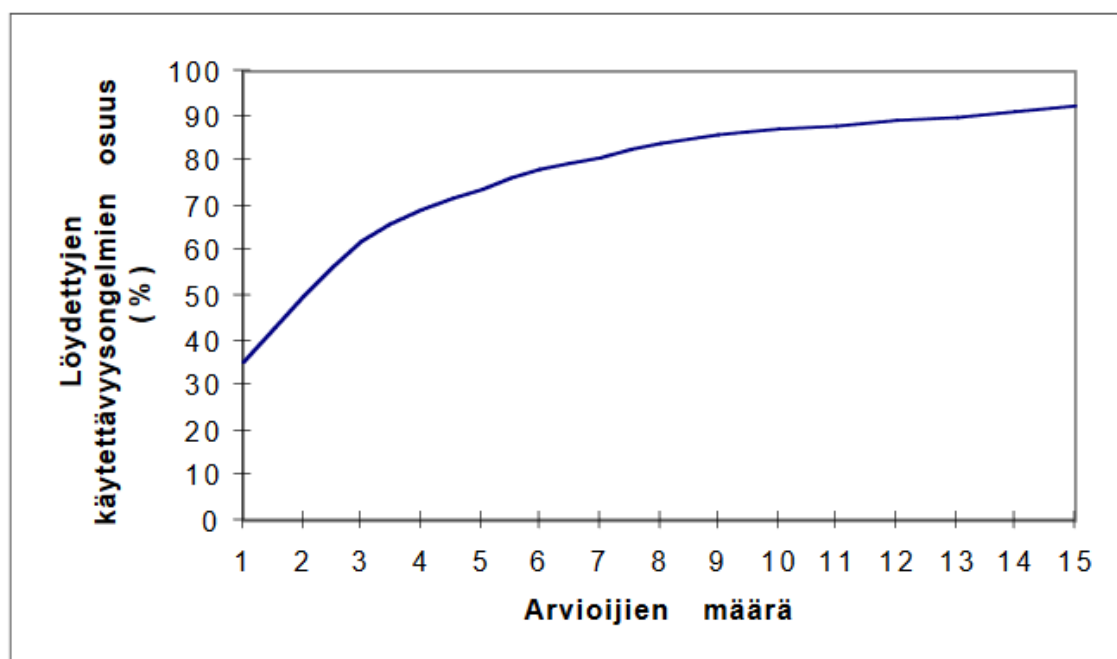
Liikkeellä voidaan viestiä kohdennuksen vaihtamisesta näkymien välillä, esimerkiksi valittuna oleva elementti voidaan suurentaa muita elementtejä isommaksi. Liikkuvan ja välähtelevän sisällön pitäisi olla pysäytettävissä tai piilotettavissa, jos se kestää yli viisi sekuntia. Välkkyvä sisältö ei saisi välkkyä yli kolmea kertaa sekunnissa. (Material Design n.d.)

5 KÄYTETTÄVYYDEN ARVIOIMINEN

5.1 Heuristiikka ja arviointimenetelmät

Heuristiikalla tarkoitetaan ohjelmistosuunnittelussa käytettävyyteen liittyviä muis-tisääntöjä. Heuristisessa arvioinnissa keskitytään tuotteen käytettävyydsongel-mien löytämiseen käyttämällä tuotetta ja tutkimalla sen käyttöä ennalta määritel-tyjen heurististen sääntöjen avulla. Tuotteen käytettävyyden arvioiminen tätä me-netelmää käyttäen mahdollistaa nopean ja tehokkaan käytettävyydsongelmien löy-tämisen. Arvioinnin suorittavat asiantuntijat, joilla on kokemusta käytettävyyden arvioimisesta ja suunnittelusta, eivät käyttäjät. Siksi on hyvä muistaa, että heuris-tisen arvioinnin lisäksi tulee testausta suorittaa myös käyttäjillä. (Käyttötuotteen heuristinen arviointi n.d.)

Asiantuntija-arviointi ei vaadi suuria valmisteluja ja sitä voidaan käyttää useassa tuotekehityksen eri vaiheessa. Arviointia suorittavilla asiantuntijoilla tulisi löytyä kokemusta joko sovelluksen toiminnasta tai käytettävyyden arvioimisesta. Ideaa-lisessa tilanteessa arvioijat ovat sekä käytettävyydsasiantuntijoita että sovelluksen suunnittelijoita. Testaukseen kannattaa hankkia yleensä kolmesta viiteen eri asi-antuntijaa, sillä yksittäinen asiantuntija ei huomaa läheskään kaikkia ongelma-kohtia sovelluksesta. Yksi asiantuntija löytää noin 35 % ongelmista, kolme asian-tuntijaa noin 60 % ongelmista ja viisi asiantuntijaa noin 75 % ongelmista (kuvio 2). Viittä useammasta arvioijasta ei saada enää niin paljoa hyötyä, että se olisi tehokkuuden kannalta tarpeellista. (Riihiahho n.d., 2)



KUVIO 2. Asiantuntija-arvioijien määrä suhteessa löydettyjen käytettävyyssongelmien osuuteen (Riihiahho n.d., 3)

Käyttäjätestaus toimii vastapainona heuristiselle arvioinnille, jossa sovelluksen käyttöä tarkastellaan ainoastaan asiantuntijoiden kannalta. Käyttäjätestauksessa saadaan selville käyttäjien kohtaamia ongelmia realistisessa käyttötilanteessa. Käyttäjää voidaan haastatella ennen ja jälkeen testausta sekä heille voidaan luoda kysymyksiä, joita ohjaaja esittää käyttämisen aikana testaajalle. Käyttäjää seurataan koko testauksen ajan hänen suorittaessaan ennalta luotuja tehtäviä sovelluksen käyttöön liittyen. Testaustilanteet voidaan tallentaa esim. videolle tai äänitteeksi myöhempää läpikäymistä varten. (Riihiahho 2000, 8.)

Käyttäjätestauksen suorittavat koehenkilöt kannattaa valita mahdollisimman hyvin sovelluksen loppukäyttäjäkuntaa ajatellen. Noin kolmesta kuuteen testaajaa per kohderyhmä on yleensä sopivasti, sillä liian monta testaajaa vie paljon ylimääräistä aikaa tuomatta lisää hyötyä. Käyttäjätestauksia tehdään yleensä tuotekehityksen keski- ja loppuvaiheilla, kun käyttäjälle on tarjottavana jo jonkinlainen demo tai valmis tuote. Testitehtävät tulee laatia kirjoitettuun tarinamuotoon, mutta ne eivät saa ohjata käyttäjää tehtävien tekemisessä. Yhden tehtävän sopiva kesto on noin 2-20 minuuttia, jotta tehtävä pysyy vielä helposti analysoitavana. (Torp n.d.)

5.2 Shneidermanin kahdeksan kultaista sääntöä

Tunnetuimpia heuristisen arvioinnin listoja ovat Jakob Nielsenin ja Ben Shneidermanin heuristiikat. Shneiderman on tietojenkäsittelytieteen professori, joka perusti myös Marylandin yliopistoon kuuluvan ihmisen ja tietokoneen vuorovaikutusta tutkivan laboratorion vuonna 1983. Hänen työtänsä ihmisen ja tietokoneen vuorovaikutuksen sekä informaation visualisoinnin parissa arvostetaan laajasti maailmalla. (Association for Computing Machinery 2014.)

Shneiderman päätti tiivistää käyttöliittymäsuunnittelun perusteet muutamaksi ytimekkääksi periaatteeksi, joita muut opiskelijat ja suunnittelijat pystyisivät käyttämään hyväkseen tulevaisuudessa. Näin syntyi Shneidermanin kahdeksan kultaista sääntöä. Säännöt sopivat suurimmalle osalle vuorovaikutteisia järjestelmiä, mutta niitä on myös laajennettu ja muunneltu ajan saatossa. Esimerkiksi Jakob Nielsen käytti näitä sääntöjä luodessaan oman listansa heuristisista periaatteista. (Shneiderman n.d.)

5.2.1 Pyri johdonmukaisuuteen

Samanlaisissa tilanteissa samalla tavalla toimiminen helpottaa sovelluksen käytön opettelua. Johdonmukaisuuden saavuttamiseksi sovelluksessa tulee käyttää samaa termistöä ohjeistuksissa, valikoissa ja toiminnoissa. Myös värit, elementtien asettelu, isot kirjaimet, fontit ja muut ulkoasuun vaikuttavat asiat tulee suunnitella yhtenäisiksi koko sovelluksessa. (Shneiderman n.d.)

Käyttäjät tottuvat paljon nopeammin tuotteen digitaaliseen ympäristöön, kun tuotteessa noudatetaan johdonmukaisuutta. Jos hyväksymisdialogissa käyttäjälle annetaan vihreä hyväksy-painike ja punainen peruuta-painike, olisi kaikissa samanlaisissa valintadiologeissa hyvä käyttää näitä samoja värejä ja tekstejä, joihin käyttäjä on jo ensimmäisessä valintadialogissa tottunut.

5.2.2 Hae yleistä käytettävyyttä

Käyttäjien eroavaisuudet tulee ottaa huomioon käytettävyyttä suunniteltaessa tarjoamalla yksinkertaisia ja helppokäyttöisiä toiminnallisuuksia, mutta rajoittamatta kokeneemmilta käyttäjiltä tehokkaamman käytön mahdollistavat monimutkaisemmat toiminnot. Aloittelijoille voidaan antaa enemmän käyttöön liittyviä ohjeita ja kokeneemmat käyttäjät voivat hyväksikäyttää esimerkiksi pikanäppäimiä. (Shneiderman n.d.)

Sovelluksen tarjotessa helppokäyttöisen ja tehokkaan toimintaympäristön käyttäjälle tämän kokemuksesta ja rajoitteista riippumatta laajennetaan samalla tuotteen käyttäjäkuntaa. Myös käyttökokemus parantuu ja se voi johtaa hyvään palautteeseen käyttäjiltä sekä menestyksekkäämpään markkinointiin.

5.2.3 Anna selkeää palautetta

Käyttäjien on tiedettävä koko ajan mitä sovelluksessa tapahtuu ja missä he ovat. Tämä voidaan mahdollistaa hyvän ja selkeän palautteen avulla. Toiminnoista on annettava käyttäjälle helposti ymmärrettävä palaute, josta käyttäjä saa selville mitä tapahtuu tai on tapahtunut. Palaute pitää olla ymmärrettävissä, vaikka käyttäjä olisi täysin aloittelija tietotekniikan kanssa ja ei välttämättä ymmärrä täysin tietotekniikkatermistöä. (Wong 2018.)

Palautteen ei aina tarvitse olla näytölle ilmestyvää tekstiä, vaan se voidaan ilmaista myös visuaalisesti. Esimerkiksi painiketta painaessa painike voi vaihtaa väriä, suurentua tai upota ja kaikki nämä tehosteet viestivät käyttäjälle, että painike rekisteröi painalluksen. Jos minkäänlaista palautetta ei anneta toiminnosta, käyttäjä voi hämmentyä, eikä hänellä ole varmuutta, onko toiminto suoritettu onnistuneesti vai ei.

5.2.4 Suunnittele dialogit johtamaan lopputulokseen

Peräkkäisiä toimintoja suoritettaessa tulee miettiä toimintojen järjestystä, eli toimintaketjulla on oltava alku-, väli- ja loppuvaiheet. Kun käyttäjä pääsee toimintaketjun loppuun ja suorittaa viimeisen tarvittavan toiminnon, tulee käyttäjälle ilmoittaa koko ketjun onnistuneesta suorittamisesta. Tämä tuottaa käyttäjälle onnistumisen tunteen ja käyttäjä voi unohtaa ketjun aikana mahdollisesti kehittyneet varasuunnitelmat ja keskittyä seuraaviin toimintoihin. (Shneiderman n.d.)

Verkkokaupasta ostaessaan käyttäjä etsii ensiksi haluamansa tuotteen tai tuotteet. Tämän jälkeen hän lisää ostoksensa ostoskoriin. Valittuaan haluamansa tuotteet hän suuntaa kassalle ja maksamaan. Onnistuneen tilauksen jälkeen käyttäjä saa ilmoituksen onnistuneesta maksusta ja mahdollisesti myös tilausvahvistuksen sähköpostiinsa. Tässä vaiheessa käyttäjä ymmärtää, että tilauksen toimintaketju on suoritettu onnistuneesti loppuun ja hän voi unohtaa tilaamiseen liittyvät tapahtumat ja keskittyä muihin asioihin.

5.2.5 Tarjoa yksinkertainen virheenkäsittely

Mitä vähemmän käyttäjä voi tehdä virheitä käyttöliittymässä, sen parempi. Käyttöliittymä kannattaa suunnitella ennalta virheitä ehkäiseväksi sen sijaan, että jokaisesta virheestä annetaan vain virheilmoitus. Kaikkia virheitä ei voida kuitenkaan välttää käyttöliittymäsuunnittelullakaan, joten virheen ilmaantuessa siitä on ilmoitettava käyttäjälle selkeästi ja käyttäjää on opastettava palautumaan virheestä askel askeleelta. (Wong 2018.)

Syötekentät voidaan rajoittaa hyväksymään pelkästään aakkosia tai numeroita tarpeen mukaan sekä syötteen pituutta voidaan rajoittaa. Kun käyttäjän ei haluta suorittavan tiettyä toimintoa, voidaan toiminnon aktivoiva elementti kytkeä pois päältä ja näin estää ei-toivotun toiminnon suorittaminen vahingossa. Toinen vaihtoehto virheilmoituksen lisäksi on pitää käyttöliittymä muuttumattomana, jos virhe on esimerkiksi niin lievä, ettei virheilmoitusta tarvita.

5.2.6 Salli helppo toimintojen peruminen

Käyttäjän tulee pystyä perumaan tekojaan. Peruminen tulee mahdollistaa niin monessa tilanteessa kuin mahdollista, sillä se lievittää käyttäjän huolta mahdollisista virheistä. Tämän ansiosta käyttäjä voi myös uskaltautua kokeilemaan uusien vaihtoehtojen valitsemista sovelluksessa, kun pelkoa virheiden peruuttamattomuudesta ei ole. (Shneiderman n.d.)

Esimerkiksi verkkokaupoissa ennen tilauksen lähettämistä käyttäjällä on mahdollisuus palata takaisin muuttamaan tilaukseen liittyviä tietojaan, kuten yhteystietoja tai tilattavia tuotteita. Takaisin palaaminen mahdollistaa ostosten nopeamman tekemisen, kun käyttäjän ei tarvitse olla ostostensa kanssa liian varovainen virheiden välttämiseksi.

5.2.7 Tue käyttäjän hallintakäsitystä

Käyttäjälle tulee muodostaa hallinnan tunne. Tämä tarkoittaa sitä, että käyttäjä hallitsee sovellusta eikä sovellus käyttäjää. Sovelluksen tulee pyrkiä suorittamaan mahdollisimman moni toiminto käyttäjän toimesta sen sijaan, että sovellus päättäisi käyttäjän puolesta. Yllätykset ja muutokset sovelluksen normaalissa käytöksessä ärsyttävät käyttäjää. (Shneiderman n.d.)

Hallinnan tunne korostuu kokeneemmilla käyttäjillä, jotka haluavat hyödyntää kertynyttä kokemustaan. Käyttäjän tulee pystyä tehostamaan sovelluksen käyttöä esimerkiksi poistamalla aloittelijoille tarkoitettuja ohjeita tai kytkemällä päälle tehokkaampia, kokeneemmille käyttäjille tarkoitettuja ominaisuuksia.

5.2.8 Vähennä lyhytkestoisen muistin kuormitusta

Käyttöliittymän yksinkertaisuus ja informaatiohierarkia korostuu, kun käytettävyyttä tarkastellaan lyhytkestoisen muistin kuormituksen kannalta. Koska ihminen pystyy ylläpitämään noin viittä eri asiaa lyhytkestoisessa muistissa yhtä ai-

kaa, tulisi sovelluksen pyrkiä käyttämään tunnistamista muistamisen sijaan. Näkymästä toiseen siirryttäessä ei kannata pakottaa käyttäjää muistamaan edellisen näkymän informaatiota, vaan jos uudessa näkymässä tulee käyttää edellisen näkymän tietoja, kannattaa näkymät yhdistää. (Wong 2018.)

Puhelinnumeroita ei tarvitse kirjoittaa uudestaan, kun ne on kerran lisätty puheliin ja pitempien lomakkeiden tiedot tulisi tiivistää kompaktisti yhteen näkymään käyttäjälle. Myös nettisivujen sijainnit on hyvä näyttää käyttäjälle, ettei käyttäjän tarvitse muistaa, minkä sijainnin hän on sivulle valinnut. (Shneiderman n.d.)

5.3 Nielsenin heuristiset periaatteet

Jakob Nielsen on tanskalainen professori, joka valmistui Tanskan teknillisestä yliopistosta Kööpenhaminassa opiskeltuaan ihmisen ja tietokoneen välistä vuorovaikutusta. Hän perusti Nielsen Norman Group -järjestön yhdessä Donald A. Normanin kanssa ja hän on kirjoittanut useita teoksia käytettävyyteen ja käyttöliittymiin liittyen. (Nielsen Norman Group n.d.)

Nielsen kehitti kymmenen nyrkkisääntöä heuristista arviointia varten. Säännöt kehittyivät yhteistyössä Rolf Molichkin kanssa. Nämä kymmenen sääntöä on tarkoitettu avuksi vuorovaikutuksen suunnittelemisessa, eikä niitä tule noudattaa sokeasti, vaan niitä tulee soveltaa jokaiseen järjestelmään omalla tavallaan. Nielsen tiivisti heuristisen listansa 249 eri käytettävyysongelman pohjalta niin, että ne olisivat mahdollisimman yksinkertaisesti selitetty. (Nielsen 1994.)

5.3.1 Järjestelmän tilan näkyvyys

Järjestelmän tilan näkyvyys kertoo siitä, kuinka hyvin käyttäjä näkee järjestelmän sen hetkisen tilan laitteella. Ihmisillä on pyrkimys pystyä ennakoimaan ja kontrolloimaan tilanteita ja yleensä mitä enemmän informaatiota on tarjolla, sen paremmin pystytään tekemään valintoja. Käyttäjän tulee saada tietää, onko laitteen ja käyttäjän välinen vuorovaikutus onnistunut vai onko tapahtunut virheitä. Välittömän palautteen ansiosta käyttäjä ei jää miettimään tapahtunutta, vaan pystyy

jatkamaan toimintaansa heti, oli seuraava tehtävä sitten palautuminen virheestä tai onnistuneen vuorovaikutuksen jälkeinen sovelluksen käytön jatkaminen. (Harley 2018.)

Järjestelmän tilaa voidaan ilmentää monin eri tavoin. Latausnäkyvässä käyttäjälle voidaan kertoa tekstillä mitä tapahtuu ja latauspalkilla kuinka paljon kokonaisprosessista on suoritettu. Verkkokaupasta junan paikkaa varatessaan käyttäjä painaa haluamaansa vapaata paikkaa ja paikka vaihtaa värinsä vihreäksi, jolloin värinvaihto viestii käyttäjälle onnistuneesta toiminnosta. Musiikkisoittimen ollessa päällä ja kappaleen soidessa voi kännykän lukitusnäytöllä näkyä kappaleen pituus ja senhetkinen eteneminen.

5.3.2 Tuotteen ja tosielämän vastaavuus

Tuotteessa tulee käyttää käyttäjälle mahdollisimman selkeitä ja ymmärrettäviä termejä ja symboleita, sillä asioiden tuttuus tuo lisää käyttömukavuutta. Käyttäjälle tosielämästä tuttujen käsitteiden käyttäminen sovelluksessa ei vaadi käyttäjältä uuden opettelua, vaan käyttäjä pystyy automaattisesti yhdistämään sovelluksen ulkopuolelta opittuja asioita sovellukseen. (Kaley 2018.)

Väreillä, tekstityypeillä, kuvakkeilla ja monilla muilla visuaalisilla elementeillä pystytään yhdistämään tosielämän ilmiöitä virtuaaliseen maailmaan. Hyvä esimerkki tuotteen ja tosielämän vastaavuudesta on käyttöjärjestelmistä löytyvä roskakori. Käyttäjä pystyy yhdistämään tarkoituksen poistaa asioita roskakorin avulla. Käyttäjä tietää, että halutessaan päästä eroon tiedostoista hänen tulee siirtää ne roskakoriin poistettavaksi. Myös värit vihreä ja punainen yhdistetään tosielämässä yleensä onnistumiseen ja virheeseen ja näitä väriyhdistelmiä voi nähdä erittäin monessa sovelluksessa.

5.3.3 Käyttäjän kontrolli ja vapaus

Käyttäjälle tulee tarjota mahdollisuus peruuttaa toimintojaan ja tapahtumia ilman ylimääräisiä dialogeja, eli eräänlainen hätäuloskäynti. Jos käyttäjä tekee vahingossa virheen, hänen ei tarvitse huolehtia tulevasta vaan hän voi palata takaisin pisteeseen ennen virheen suorittamista. Toimintojen perumisen lisäksi tulee tukea myös mahdollisuutta tehdä toiminto uudelleen. (Liyanage 2016.)

Verkkosivuille tiedostoja ladatessa latauspalkin ympäriltä voi löytyä Peruuta-painike. Tämän avulla käyttäjä pystyy keskeyttämään latauksen, jos lataus on aloitettu vahingossa tai se kestää käyttäjän mielestä liian pitkään. Verkkoselaimilla sivuja selatessa voi tulla tarve päästä takaisin aiemmin vierailuille sivustoille esim. väärälle sivulle siirryttäessä. Tätä varten selaimiin on luotu Edellinen-painike edelliselle sivulle siirtymistä varten.

5.3.4 Yhdenmukaisuus

Sovelluksessa säilytettävän yhdenmukaisuuden ansiosta käyttäjä ei jää miettimään tarkoittavatko sanat, tilanteet ja toiminnot samaa. Standardien ansiosta sovellus pystyy tarjoamaan käyttäjälleen yhtenäisen sanaston ja toimintatavat. On kuitenkin muistettava, että yleisiä standardeja ei pystytä täysin noudattamaan, koska sovellukset poikkeavat toisistaan, vaan standardeihin tulee mukautua. Suurin osa suunnittelutyöstä tuleeekin yleensä niiden elementtien suunnittelusta, joihin yleiset standardit eivät sovi, jolloin ne on suunniteltava yksilöllisesti soveluksittain. (Nielsen 1999.)

Eri sivuilla käytettävien samojen elementtien tulisi myös näyttää samalta. Sovelluksessa näkyvät tiettyjä kategorioita ilmaisevat kuvakkeet tulisi esittää aina samassa kontekstissa sivusta riippumatta. Esimerkkinä huonosta yhdenmukaisuudesta on etusivulla kysymysmerkillä varustettu kuvake, josta käyttäjä pääsee usein kysytyihin kysymyksiin. Seuraavalla sivulla samainen kysymysmerkillä varustettu kuvake kuitenkin vie käyttäjän sivuston ohjeisiin.

5.3.5 Virheiden ehkäisy

Hyviä virheviestejä vielä parempi ratkaisu on koittaa välttää virheet kokonaan. Käyttäjää ei ikinä tule syyttää virheistä, sillä helppo virheiden tekeminen viestii huonosta suunnittelusta, josta sovelluksen suunnittelija on vastuussa. Virheiden mahdollisuus tulee poistaa kokonaan tai mahdollisten virheiden ilmaantuessa käyttäjää on opastettava siitä palautumiseen yksinkertaisesti. (Laubheimer 2015.)

Sen sijaan, että lentomatkaa varatessaan käyttäjä joutuisi syöttämään lähtö- ja paluupäivän tekstinä, tarjotaan käyttäjälle kalenteri, josta hän voi valita päivämäärät. Kalenteri on myös suunniteltu niin, ettei käyttäjä pysty valitsemaan paluupäivää ennen lähtöpäivää. Toinen esimerkki virheiden ehkäisystä on ehdotusten tarjoaminen haussa. Kun käyttäjä alkaa kirjoittamaan haluamaansa hakua, ehdottaa hakukenttä erilaisista vaihtoehdoista, jotka voivat muodostua käyttäjän antamista merkeistä. Kaikki ehdotukset ovat valideja vaihtoehtoja hakuun, joten oman hakukohteensa näkyessä ehdotuksissa ja sen valittuaan käyttäjä on suorittanut automaattisesti virheettömän haun. Tällä voidaan ehkäistä esimerkiksi kirjoitusvirheitä. (Laubheimer 2015.)

5.3.6 Tunnistaminen mieluummin kuin muistaminen

Käyttäjälle kannattaa esittää asioita, joita hänen ei tarvitse muistaa, mutta jotka hän osaa tarpeen tullen tunnistaa. Tunnistaminen on muistamista helpompaa ja vähemmän muistia kuormittavaa, sillä tunnistaminen hyötyy enemmän vihjeistä kuin muistaminen. Tämän takia tärkeät tiedot, toiminnot ja elementit tulisi olla käyttäjän nähtävissä tai ainakin helposti saatavilla, ettei ylimääräistä muistamista tarvita. (Budi 2014.)

Hyvä esimerkki muistamisesta on sisäänkirjautuminen. Käyttäjän tulee muistaa käyttäjätunnus ja salasana päästäkseen kirjautumaan sisään. Käyttäjätunnuksista annetaan hyvin harvoin mitään vihjeitä, joten niiden käyttäminen vaatii enemmän muistamista. Tunnistamisen yksi selkeä esimerkki on sovelluksesta löytyvät valikot. Valikoissa on tietty määrä vaihtoehtoja, mistä käyttäjä voi valita,

joten vaikka valittavaa vaihtoehtoa ei muistaisikaan, voi sen näkeminen valikossa palauttaa oikean vaihtoehdon mieleen. (Budi 2014.)

5.3.7 Joustavuus ja tehokkuus

Jokaisella sovelluksella on kokeneempia ja kokemattomampia käyttäjiä. Molemmat käyttäjäryhmät tulisi ottaa huomioon, kun suunnitellaan sovelluksen toimintaa. Kokeneemmat käyttäjät voivat haluta toimintoja, joiden avulla he pystyvät työskentelemään tehokkaammin ja ohittamaan vaiheet, jotka ovat hyödyllisiä aloittelijoille, mutta hyödyttömiä heille. (Nielsen 1994.)

Aloittelijan on hyvä etsiä toimintoja valikoiden kautta, sillä hän ei vielä välttämättä edes tiedä mitä kaikkia toimintoja sovelluksesta löytyy, mutta kokeneemman käyttäjän jo tuntiessa mahdolliset toiminnot, pystyy hän suorittamaan niitä pikanäppäinten avulla. Pikanäppäimet tehostavat kokeneemman käyttäjän toimintaa huomattavasti. Myös verkkokaupassa asioidessa ensimmäistä kertaa käyttäjä joutuu syöttämään tilaustietonsa, mutta seuraavilla tilauskerroilla käyttäjälle voidaan tarjota viimeksi käytettyjä tilaustietoja, jottei hänen tarvitsisi käyttää aikaa tietojen syöttämiseen uudelleen.

5.3.8 Estetiikka ja minimalistinen suunnittelu

Mitä enemmän informaatiota käyttäjälle näytetään kerralla, sitä hankalampi käyttäjän on löytää näkymästä yksittäistä tietoa, sillä kaikki ylimääräinen tieto vie näkyvyyttä halutulta tiedolta. Sen takia dialogien ei tulisi sisältää turhaa tai harvoin tarvittavaa tietoa. (Nielsen 1994.)

Epäselvä käyttöliittymä, josta halutun toiminnon tai tiedon löytäminen kestää liian pitkään, karkottaa mahdolliset käyttäjät. Näkymää, josta löytyy ainoastaan tarvittavat toiminnot ja tiedot, on huomattavasti mukavampi käyttää ja yksinkertaisesti suunniteltu näkymä näyttää myös käyttäjän silmissä kauniimmalta. Sovelluksen tekstien kannattaa olla mahdollisimman lyhyitä ja ytimekkäitä. Selkeän kirjaintyypin, kuten Arial, valitseminen parantaa luettavuutta huomattavasti.

5.3.9 Virhetilanteista palautuminen

Käyttäjää tulee opastaa virheen ilmaantuessa. Ystävällinen, selkeä ja rakentava virheilmoitus samalla sekä auttaa käyttäjää palautumaan virheestä että opettaa käyttäjää toimimaan jatkossa samanlaisen virheen ilmaantuessa. Virhe tulee selittää käyttäjälle mahdollisimman tarkasti, mutta se tulee myös ilmoittaa yksinkertaisella kielellä, jotta kuka tahansa käyttäjä pystyy ymmärtämään sen. (Nielsen 2001.)

Osa lomakkeisiin täytettävistä tiedoista voi olla sovelluksen mukaan pakollisia. Pakollisen tiedon täyttämättä jättäminen ja yritys siirtyä eteenpäin aiheuttaa tuolloin virheen. Yleensä tällainen virhe ilmoitetaan lomakkeessa lukevalla punaisella tekstillä tai korostamalla kentän väriä, jotta käyttäjä tietää missä lomakkeen kentässä virhe on tapahtunut. Tämä on yksinkertainen, mutta toimiva ratkaisu virheen ilmoittamiseen ja käyttäjän ohjaamiseen.

5.3.10 Opastus ja ohjeistus

Sovelluksen tulisi olla tarpeeksi helppokäyttöinen, ettei sen käyttämiseen tarvittaisi ohjekirjaa. Jos käyttäjä kuitenkin jää jumiin johonkin toimintoon tai tilanteeseen, on avun ja ohjeiden oltava helposti saatavilla. Ohjeiden tulee keskittyä juuri käyttäjän kohtaamaan virheeseen ja toiminnot ongelman ratkaisemiseksi tulisi ilmoittaa askel askeleelta. (Nielsen 1994.)

Hyvä esimerkki tehokkaasta ja käytännöllisestä opastuksesta on sovellusten ja sivustojen Usein kysytyä -osio. Sieltä käyttäjät löytävät yleisimpiin ongelmatilanteisiin laaditut selkeät selitykset ja toimintatavat. Usein kysytyä -osio voi säästää ylimääräisiä yhteydenottokertoja asiakastukeen sekä tarvetta selata läpi käyttöohjedokumentteja.

6 ALOITUSNÄYTTÖSOVELLUKSEN ARVIOINTI JA TESTAUS

6.1 Asiantuntija-arvioinnin järjestelyt

Asiantuntija-arvioinnissa käytettiin hyväksi Nielsenin kymmentä heuristista periaatetta, joiden pohjalta käytettävyysongelmia lähdettiin etsimään. Lomakkeen mallina toimi heuristisen arviointilomakkeen pohja (Torp n.d.). Arviointilomake sisälsi kymmenen kohtaa, joista jokaisessa perehdyttiin tarkemmin aina yhden periaatteen näkökulmasta huomioituihin ongelmiin (ks. Liite 1).

Arviointiin osallistuvia asiantuntijoita oli kolme: opinnäytetyön tekijä itse ja kaksi muuta kollegaa. Arviointi suoritettiin Jidokalla erillisessä testaamiselle varatussa tilassa. Testilaitteena oli 23-tuumainen Android 4.4.4 KitKat -mobiilikäyttöjärjestelmällä toimiva taulutietokone. Aloitusnäyttösovellus asennettiin ja päivitettiin laitteeseen valmiiksi ennen arvioinnin alkamista. Jokainen arvioija täytti arviointilomakkeensa henkilökohtaisesti, jonka jälkeen kaikki lomakkeet kerättiin talteen myöhempää analysoimista varten.

6.2 Asiantuntija-arvioinnin toteutus

Arviointilomake jakoi arvioinnin kolmeen osaan: ongelmien listaukseen, niiden priorisointiin ja ratkaisuehdotuksiin. Ensimmäisessä vaiheessa sovelluksesta etsittiin heuristisen muistilistan avulla käytettävyysongelmia. Ongelmien ilmetessä ne kirjattiin niiden rikkoman heuristisen periaatteen alle. Jokaisesta ongelmasta annettiin pienimuotoinen kuvaus ja kerrottiin, kuinka se rikkoo heuristiikkaa.

Toisessa vaiheessa aiemmin löydetyille ongelmille annettiin vakavuusluokitus sen mukaan, kuinka harmillinen ongelma on käytettävyyden kannalta. Asteikko on viisiosainen, jossa numero yksi tarkoittaa vakavinta mahdollista ongelmaa ja numero viisi tarkoittaa olematonta ongelmaa.

Viimeisessä vaiheessa jokainen arvioija valitsi mielestään viisi tärkeintä havaittua ongelmaa ja merkitsi ne taulukkoon. Tämän jälkeen arvioija antoi ongelmista henkilökohtaiset ratkaisuehdotuksensa. Tällä tavoin samoille ongelmille saatiin ratkaisuja useamman eri arvioijan näkökulmasta ja monista eri vaihtoehdoista voidaan valita parhaiten sopiva toteutus.

6.3 Käyttäjätestauksen järjestelyt

Käyttäjätestaus suoritettiin Jidokalla samoissa tiloissa, missä asiantuntija-arvioinkin tehtiin. Laitteena käytettiin samaa taulutietokonetta (ks. 6.1). Testikäyttäjiä saatiin rekrytoitua yhteensä kaksi kappaletta ja molempien kohdalla koko käyttäjätestaukseen aikaa kului puolesta tunnista tuntiin. Testihenkilöt pyrittiin valitsemaan mahdollisimman hyvin kohdekäyttäjryhmän mukaan eli käyttäjien tuli olla mahdollisimman kokemattomia Android-käyttöjärjestelmän kanssa ja ensikertalaisia testattavan aloitusnäyttösovelluksen käytössä. Näin saataisiin esille realistisia ongelmia, joita teknologiasta tietämätön henkilö voisi sovellusta käyttäessään kohdata.

Myös käyttäjätestausta varten luotiin testilomake. Lomakkeen alussa kerrottiin lyhyellä tekstillä testikäyttäjälle hieman taustaa testattavasta sovelluksesta ja miten testijärjestelyt etenevät. Käyttäjälle painotettiin, että arvioinnin kohteena on sovelluksen käytettävyyys eikä käyttäjän suoriutuminen. Näin käyttäjä voi keskittyä mahdollisimman luonnolliseen sovelluksen käyttöön ilman suorituspaineita.

Pohjustuksen jälkeen aloitettiin haastattelulla. Käyttäjältä kysyttiin yhteensä neljä esikysymystä. Kolme ensimmäistä kysymystä käsittelivät käyttäjän aiempaa kokemusta sekä Android-käyttöjärjestelmästä että taulutietokoneiden käytöstä. Tarkoituksena oli kartoittaa tarkemmin, miten hyvin testikäyttäjä vastaa sovelluksen kohderyhmää. Viimeisellä kysymyksellä haettiin käyttäjän suhtautumista sovellusta kohden ennen varsinaisten testitehtävien aloittamista.

Tehtäviä käyttäjätestissä oli kuusi kappaletta. Tehtävät koitettiin rakentaa niin, että ne käsittelisivät sovelluksen tärkeimmät toiminnallisuudet. Tärkeimpiä toiminnallisuuksia ovat navigointi aloitusnäyttösovelluksen, kolmannen osapuolten

sovellusten ja Ilo-elämyspankin välillä, elämyspankin videoiden toistoon mukautetun mediasoittimen käyttö ja aloitusnäyttösovelluksen sisäänrakennetun muistion käyttö. Jokaisesta tehtävästä mitattiin käyttäjältä kulunut aika ja jos käyttäjä ei saanut tehtävää suoritettua loppuun onnistuneesti, merkittiin käytetyksi ajaksi viiva. Käyttäjän tehdessä tehtävää sovelluksen ja käyttäjän välisestä vuorovaikutuksesta kirjattiin koko ajan ylös huomioita ja ongelmia sitä mukaa, kun niitä ilmeni. Huomioiksi kirjattiin myös käyttäjän sovelluksessa tehdyt valinnat, vaikka tehtävä olisikin suoritettu loppuun onnistuneesti. Näiden muistiinpanojen avulla pystyttiin jälkeenpäin tutkimaan, toimiko käyttäjä luonnollisesti tehokkaimmalla mahdollisella tavalla ja oliko käyttäjän työskentely sovelluksella sulavaa.

Testitehtävien jälkeen kysyttiin testikäyttäjältä vielä jälkikysymyksiä. Kysymyksiä oli kuusi kappaletta ja niillä oli tarkoitus selvittää sovelluksen käytöstä käyttäjälle muodostuneen kokemuksen piirteitä. Osa kysymyksistä oli avoimia kysymyksiä ja osa suljettuja. Avoimissa kysymyksissä käyttäjä pystyy kertomaan ajatuksensa sovelluksesta omin sanoin, kun taas suljetuissa kysymyksissä käyttäjän tulee valita sopivin vaihtoehto (Mishra 2018). Tuntemusten ja mielipiteiden lisäksi käyttäjiltä kysyttiin vapaasti, millaisia muutoksia he itse haluaisivat sovellukseen kokemuksen parantamiseksi.

6.4 Käyttäjätestauksen toteutus

Käyttäjätestauksen esikysymyksillä saatiin kartoitettua molempien testikäyttäjien aikaisempaa tietämystä. Kumpikin haastateltava sanoi joskus kuulleensa Androidista, muttei tunne käyttöjärjestelmää sen paremmin. Android-laitteita molemmat olivat käyttäneet ja omistivat itsekin ainakin yhden sellaisen. Ensimmäinen käyttäjä kertoi omistavansa puhelimen ja taulutietokoneen, jota on käyttänyt useita kertoja. Toinen käyttäjästä kertoi omistavansa Android-puhelimen ja koittaneensa myös taulutietokoneen käyttöä muutaman kerran aikaisemmin muttei itse omistanut tällaista. Molempien testikäyttäjien tapauksessa käyttö oli ollut peruskäyttöä eli esimerkiksi viestimistä ja internetin selaamista. Sulava peruskäyttö ei vaadi vielä syvempää tietämystä käyttöjärjestelmästä.

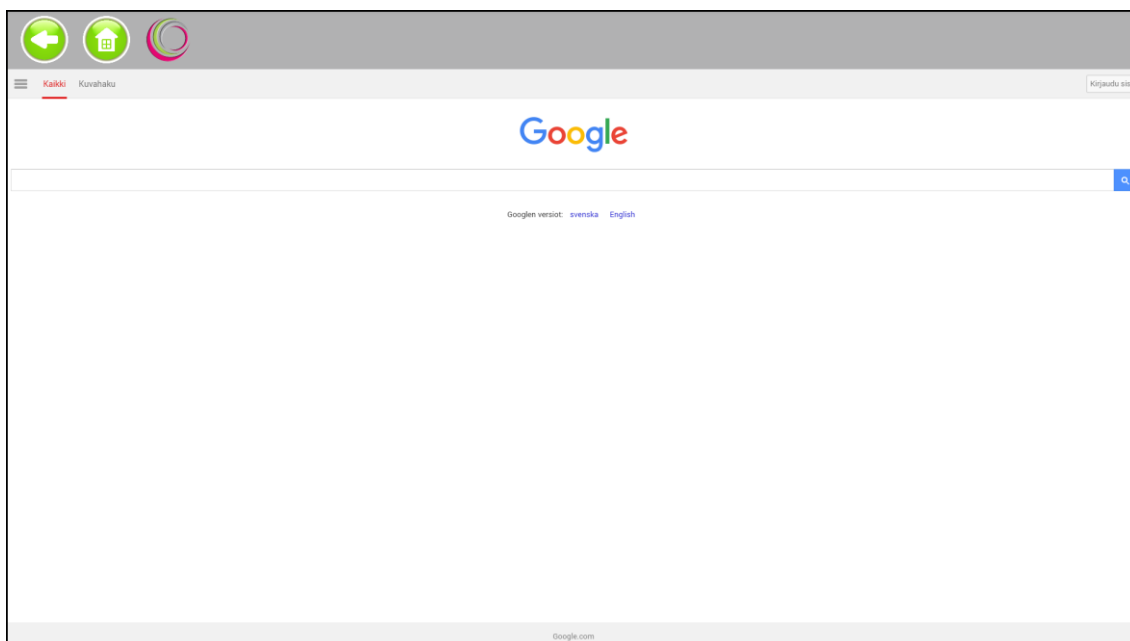
Sovelluksen käytöstä kysyttiin myös ennakkoarjatuksia. Ensimmäinen käyttäjä uskoi, että pärjää tehtävien teossa melko hyvin ja sovellus olisi jokseenkin helppokäyttöinen. Toinen käyttäjä taas uskoi suoriutuvansa tehtävistä heikommin, sillä hän ei uskonut tietävänsä sovelluksesta tarpeeksi käyttääkseen sitä sujuvasti. Kumpikaan käyttäjistä ei ollut kuitenkaan täysin varma, että voisi luottaa sovelluksen helppokäyttöisyyteen. Osa kokonaiskäyttäjäkokemuksesta muodostuu jo ennen varsinaista sovelluksen käyttöä ja tähän vaikuttavat käyttäjän aiemmat kokemukset muista samanlaisista sovelluksista tai palveluista. Kumpikin testikäyttäjä osoitti jonkin verran epävarmuutta sovellusta kohden, mikä kertoo siitä, että käyttäjä tuntee oman aikaisemman kokemuksensa olevan riittävä sovelluksen sujuvaan käyttöön ainoastaan, jos sovellus on todellakin tarpeeksi helppokäyttöinen.

Jo heti ensimmäisen tehtävän alussa molemmilla käyttäjillä ilmeni ongelmia sovelluksen käytön kanssa. Tehtävässä testattiin, kuinka helposti käyttäjä pystyy navigoimaan sovelluksen avulla Iloi.fi-elämyspankkiin ja etsimään sieltä haluamansa videon katsottavaksi. Kumpikin käyttäjä oli tekemisissä sovelluksen kanssa ensimmäistä kertaa ja vaistonvaraisesti lähti hakemaan tietä elämyspankkiin selaimen kautta. Tarjolla olisi ollut etusivulla näkyvä Elämyksiin-painike (kuva 1), jolla käyttäjät olisivat päässeet suoraan elämyspankin etusivulle. Painike näyttää kuitenkin liikaa staattiselta kovalta, eikä siitä erota syvyyttä, mikä yleensä painettavasta painikkeesta löytyy. Myös teksti ”Elämyksiin” voisi olla käyttäjää paremmin opastava. Toiselta käyttäjistä kyseinen tehtävä jäi kesken ja toiselta suorittamiseen kului tarpeettoman pitkä aika ensikertalaisellekin.



KUVA 1. Sovelluksen aloitusnäky

Toisessa tehtävässä käyttäjien täytyi navigoida takaisin sovelluksen aloitusnäky-
mään videonäkymästä. Ensimmäinen käyttäjä suoriutui tehtävästä melko nope-
asti. Ensiksi käyttäjä koitti palata aina aloitussivulle asti painamalla Edellinen-pai-
niketta (kuva 2) tarpeeksi monta kertaa, mutta huomattuaan, ettei sillä pääse se-
laimen kotisivua pidemmälle hän tajusi käyttää talolta näyttävää Koti-painiketta,
joka vie aloitusnäky-mään. Toinen käyttäjä ei kuitenkaan selviytynyt tehtävästä
yhtä nopeasti, vaan jäi pyörimään selainnäky-mään tajuamatta Koti-painikkeen
tarkoitusta. Lopulta käyttäjä sai auki natiivin navigaatiopalkin, josta hän palasi
aloitusnäky-mään natiivilla Takaisin-painikkeella. Työkalupalkin painikkeet eivät
siis olleet sovellusta ensikertaa käyttävälle henkilölle tarpeeksi selkeitä toiminnal-
lisuuksiltaan.



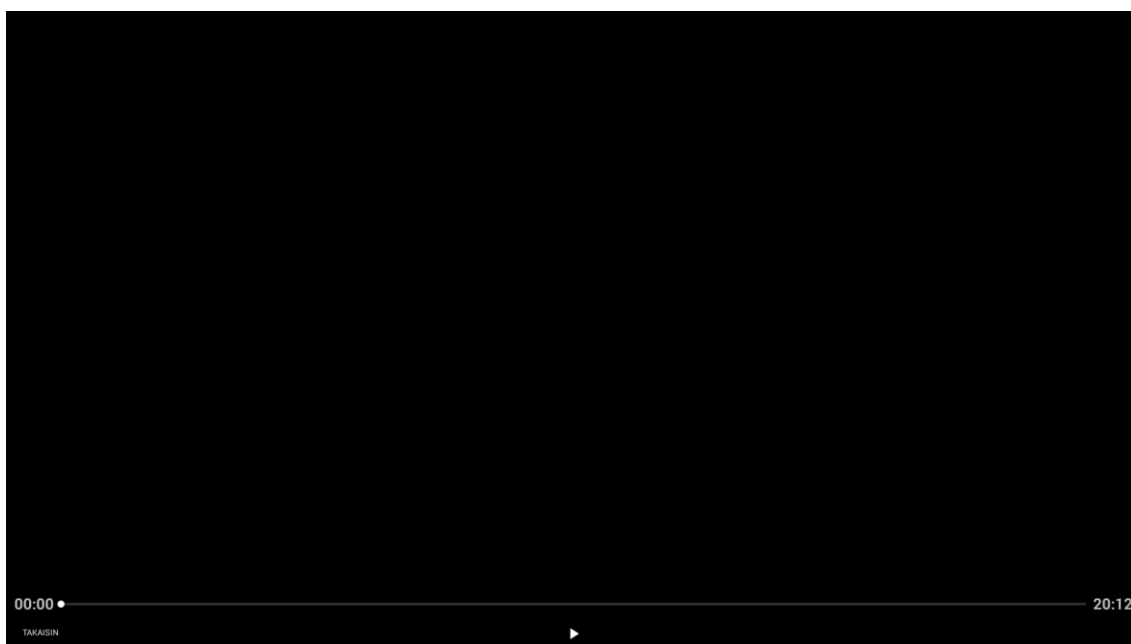
KUVA 2. Selainnäkömä räätälöidyillä työkalupalkilla

Tehtävässä kolme täytyi hakea Ilmatieteen laitoksen verkkosivuilta tämänpäiväinen sääennuste ja kirjoittaa se ylös aloitusnäkömän muistilistalle (kuva 1). Tehtävällä testattiin navigointia ja muistilistan toimintaa. Ensimmäinen käyttäjä pääsi verkkosivulle, jonka jälkeen hän koitti saada säätilatietoja kopioitua leikepöydälle. Tämä osoittautui kuitenkin liian hankalaksi ja hän päätti muistaa säätilatiedot, palata etusivulle ja kirjoittaa tiedot muistilistalle ulkomuistista. Toinen käyttäjä löysi verkkosivulta myös säätilatiedot, muttei yrittänyt edes kopioida niitä vaan päätti muistaa ne. Palattuaan aloitusnäkömään käyttäjä joutui hetken tunnistamaan muistilistaa ennen kuin pääsi kirjoittamaan siihen. Tehtävästä huomattiin, että kopioiminen ja liittäminen on ensikertalaiselle hankalaa selainnäkömässä sekä muistilista voisi olla paremmin tunnistettavissa aloitusnäkömässä.

Neljännessä tehtävässä testattiin navigoimista käyttäjän haluamalle verkkosivulle ja sitä kautta elämyspankkiin. Molemmat pääsivät haluamalleen verkkosivustolle melko nopeasti, mutta navigointi elämyspankkiin tuotti toiselle käyttäjälle hankaluuksia. Käyttäjä mietti pitkään, mitä kautta hän pääsisi elämyspankkiin nopeimmin ja päätyi tekemään Google-haun, mikä lopulta lisäsi roimasti aikaa tehtävän suorittamiseen. Työkalupalkissa olevasta logosta painamalla pääsee suoraan elämyspankin etusivulle selainnäkömässä, mutta painiketta oli hankala tun-

nistaa käytettäväksi siihen tarkoitukseen. Tästä painikkeesta puuttui myös yhdenmukainen valkoinen tausta, joka muista työkalupalkin painikkeista löytyi ja sen takia käyttäjä voi hämääntyä.

Tehtävässä viisi piti käyttäjän katsella kolmea haluamaansa videota elämyspankista. Molemmat käyttäjät onnistuivat löytämään ja toistamaan kolme videota videonäkymässä (kuva 3) onnistuneesti. Videonäkymästä palaaminen sovelluksen aloitussivulle aiheutti hämmennystä molemmilla käyttäjillä. Ensimmäinen käyttäjä koitti saada auki natiivin navigointipalkin laitteesta ja siirtyi aloitusnäky-mään natiivia Takaisin-painiketta käyttäen. Toinen käyttäjä koitti palata aloitusnäky-mään käyttämällä selainnäkömön työkalupalkin Takaisin-painiketta, mutta onnistui navigoimaan itsensä vain selaimen kotisivulle. Lopulta käyttäjä palasi sovelluksen etusivulle natiivia Takaisin-painiketta käyttäen, eikä työkalupalkin Koti-painiketta käyttäen. Tästä tehtävästä voitiin taas huomata navigoinnin haastavuus, kun käyttäjälle ei tarjota yhtä ja selkeää tapaa päästä haluamaansa paikkaan sovel-luksessa. Monta erilaista tapaa tehdä sama asia voi hämmentää käyttäjää.



KUVA 3. Räätelöity mediasoitin videonäkymässä

Viimeisessä eli kuudennessa tehtävässä käyttäjien tuli avata Yle Areena aloitusnäkömystä, katsoa sieltä haluamaansa videota ja palata takaisin aloitusnäky-mään. Sovelluksen avaaminen onnistui käyttäjiltä virheettää, niin myös videon tois-

taminen. Molemmat käyttäjät pääsivät myös onnistuneesti takaisin aloitusnäky-
mään, mutta toiselle käyttäjälle natiivipainikkeiden löytäminen tuotti hieman vii-
västystä, sillä hän oli tehtävän suorittamisen aikana laittanut navigaatiopalkin pii-
loutumaan automaattisesti näytön alareunaan. Tästäkin tehtävästä voidaan nos-
taa esille ongelmia navigaatiossa.

7 TULOKSET

7.1 Testikäyttäjien palaute

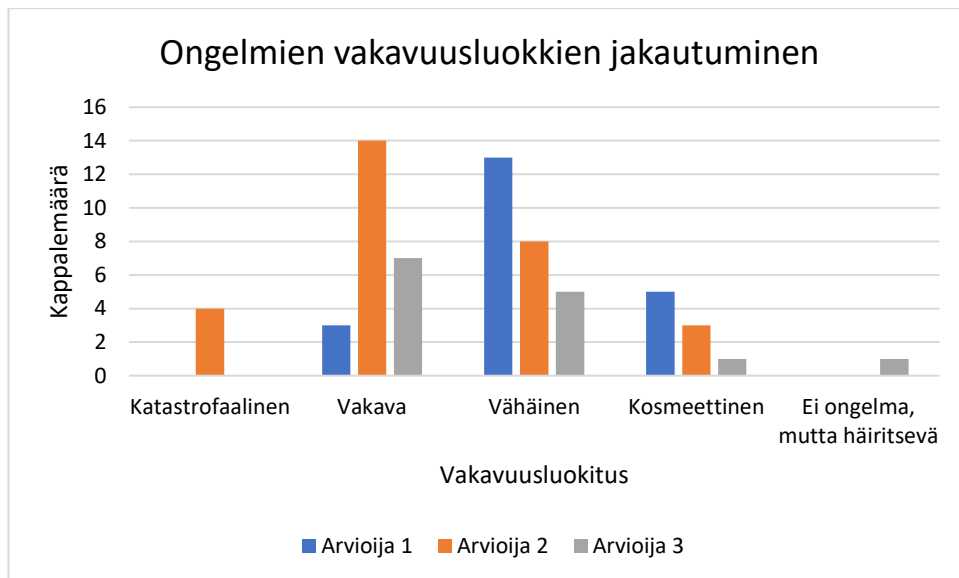
Käyttäjätestauksen lopuksi saadusta palautteesta poimittiin hyviä ja huonoja puolia sovelluksesta. Käyttäjien mukaan sovelluksen kuvakkeet olivat tarpeeksi isoja, jotta heikkonäköisemmätkin henkilöt näkevät ne. Haastateltavat painottivat myös, että sovellusta on helppo käyttää, kunhan oppii sovelluksen toimintatavat ensimmäisten käyttökertojen aikana. Ensikertalaiselle kuitenkin sovelluksen käyttäminen voi tuottaa ongelmia.

Muutosehdotuksena toinen haastateltavista mainitsi näytönsäästäjän lisäämisen sovelluksen seisoessa käyttämättä tarpeeksi pitkään. Asteikolla nolasta kymmeneen ensimmäinen testikäyttäjä antoi arvosanaksi yhdeksän ja toinen viisi, mutta molempien käyttäjien kommentteista sai selville, että sovelluksen käytettävyys vaatii parantelua.

7.2 Asiantuntija-arvioinnin käytettävyysongelmat

Käytettävyysongelmia merkittiin arvioinnin aikana ylös yhteensä 64 kappaletta. Näistä merkinnöistä moni kohdistui samaan ongelmaan, sillä ongelmasta saatettiin tehdä omanlaisensa merkintä jokaisen arvioijan kohdalla. Ensimmäinen arvioija merkitsi ylös 21 ongelmaa, toinen arvioija 29 ongelmaa ja kolmas arvioija 14 ongelmaa. Merkintöjä kertyi jokaisella arvioijalla yhdestä viiteen kappaletta jokaista heuristista periaatetta kohden.

Arvioinnin toisesta vaiheesta muodostettiin kuvio 3, joka havainnollistaa ongelmien vakavuusluokkien jakautumista. Selkeästi suurin osa ongelmista määriteltiin vakaviksi tai vähäisiksi ongelmiksi. Katastrofaalisia, eli korkeimman prioriteetin saaneita ongelmia merkittiin neljä kappaletta ja yhtä ongelmaa ei pidetty käytettävyyden kannalta haitallisena tässä sovelluksessa. Kosmeettisiakin ongelmia löytyi, mutta melko vähän kahteen seuraavaan luokkaan nähden.



KUVIO 3. Ongelmien jakautuminen eri vakavuusluokkiin

Matalampien prioriteettien ongelmiksi, eli häiritseviksi ja kosmeettisiksi ongelmiksi nostettiin mm. seuraavanlaisia ongelmia:

- Jos kolmannen osapuolen sovellusta ei ole asennettu laitteelle, sitä ei näytetä millään lailla käyttöliittymässä ennen sovelluksen valitsemista.
- Osa painikkeiden teksteistä ei ole tarpeeksi kuvaavia.
- Iloi:n kotisivuille vievä painike ei selainnäkömön työkalupalkissa näytä tarpeeksi paljon painikkeelta, enemmänkin staattiselta kuvalta.
- Selainnäkömössä näkyy käyttäjälle turhaa informaatiota.

Tärkeimmät ongelmat valittiin arvioinnin kolmannessa vaiheessa. Jokaiselta arvioijalta saatiin viisi eniten priorisoitua ongelmaa. Nämä ongelmat ovat asiantuntija-arvioinnin kannalta kaikista tärkeimpiä ja ne otetaan ensimmäisenä huomioon sovellusta kehitettäessä asiantuntija-arvioinnin pohjalta. Loogisesti tähän vaiheeseen valitut ongelmat olivat suurin osa vaiheen kaksi ongelmistä kaikista vakavimmin luokiteltuja ongelmia.

Vakavimmista ongelmista painotettiin eniten seuraavia ongelmia:

- Selaimesta puuttuu kokonaan Seuraava-painike.
- Sovellusta käytettäessä joudutaan käyttämään kolmea erilaista Takaisin-painiketta, kun halutaan palata eri tilanteista takaisin.
- Sovellus ei tarjoa käyttäjälleen minkäänlaista opastusta.

- Käytön aikana on mahdollista päästä käsiksi laitteen toimintoihin, kuten asetuksiin, liian helposti, vaikka niin ei olisi tarkoitus tapahtua.

Seuraava-painikkeen puuttuminen selainnäkyssä rajoittaa käyttäjän kontrollia ja voi vaikuttaa negatiivisesti hallinnan tunteeseen. Käyttäjä voi liikkua selaimessa nopeasti vain toiseen suuntaan, eli taaksepäin muttei eteenpäin, mikä tekee myös käytöstä tehottomampaa. Siirryttäessä vahingossa taaksepäin liian monta askelta joudutaan tekemään navigointi halutulle sivulle uudestaan sen sijaan, että sille voitaisiin siirtyä nopeasti Seuraava-painikkeen avulla.

Sovellus tarjoaa käyttäjälleen mahdollisuuden palata takaisin, mutta toiminto on toteutettu eri tavoin eri näkymissä. Mediasoittimessa, selainnäkyssä ja siirryttäessä kolmannen osapuolen sovelluksiin kaikkialla käytetään erilaisia Takaisin-painikkeita eripuolilla näyttöä. Tämä sotii yhdenmukaisuuden periaatetta vastaan, sillä käyttäjälle tulisi tarjota sama toiminto samalla tavalla kaikkialla, jottei käyttäjä ala miettimään, mistä tämän saman toiminnon voi suorittaa eri näkymissä.

Käyttäjän ollessa jumissa missä tahansa tilanteessa hänelle tulisi tarjota opastusta. Sovellus ei kuitenkaan tarjoa tätä missään tilanteessa, vaan ilmoittaa ainoastaan satunnaisista virheistä, muttei tee mitään auttaakseen käyttäjää palautumaan niistä. Myöskään käyttöohjeita ei ole tarjolla missään lähteessä. Opastus ja ohjeistus puuttuu siis kokonaan ja vaikka onkin oleellista kehittää sovellus niin, että ohjeistusta tarvittaisiin mahdollisimman vähän, on ohjeita silti annettava tarpeen niin vaatiessa.

Aloituspainikkeen sovelluksen perimmäinen tarkoitus on tarjota käyttäjälleen rajoitettu käyttöympäristö, joka sisältää halutunlaiset toiminnot. Sovellusta käyttäessä käyttäjän on mahdollisuus kuitenkin päästä käsiksi esimerkiksi laitteen asetuksiin, sillä tilapalkin voi avata näytön pyyhkäisyllä ja navigointipalkissa on ylimääräisiä toimintoja, kuten sovellusten välillä siirtyminen, mitä käyttäjän ei tarvitsisi pystyä tekemään.

7.3 Käyttäjätestauksen käytettävyysongelmat

Käyttäjätestauksessa ilmenneet käytettävyysongelmat olivat samanlaisia kuin asiantuntija-arvioinnissa. Käyttäjätestauksessa löytyneitä ongelmia oli määrällisesti vähemmän kuin asiantuntija-arvioinnissa, koska käyttäjätestauksessa jokainen tehtävä mennään läpi vain kerran testikäyttäjää kohden. Testauksen avulla voitiin kuitenkin havaita muutamia ongelmia, joita asiantuntija-arvioijat eivät taas huomanneet. Ongelmista poimittiin eniten sovelluksen käyttämistä haittaavat tapaukset.

Tehtäviä suorittaessa ylös nousseita tärkeimpiä ongelmia olivat:

- Painikkeita ei tunnistettu ja osattu yhdistää oikeisiin toimintoihin.
- Takaisin-painikkeiden erilaisuus ja toiminta hämmensi käyttäjiä.
- Muistilistaa on hankala tunnistaa muistilistaksi.

Painikkeiden toimintojen ollessa käyttäjälle epäselviä käyttäjä pyrkii välttämään painikkeiden käyttöä. Tuotteen ja tosielämän vastaavuus ei toteudu, jos käyttäjä ei osaa yhdistää painiketta oikeaan toimintoon. Tämä kertoo siitä, että painikkeissa käytetty grafiikka ja teksti eivät olleet tarpeeksi kuvaavia ja helposti ymmärrettävissä käyttäjien kannalta.

Jo asiantuntija-arvioinnissakin kriittiseksi ongelmaksi noussut erilaisten Takaisin-painikkeiden määrä ilmeni uudestaan käyttäjätestauksessa. Yhdenmukaisten toimintatapojen puuttuminen hidastaa huomattavasti sovelluksen käyttöä ja näin myös vähentää tehokkuutta. Käyttäjä joutuu käyttämään ylimääräistä aikaa samanlaisen painikkeen etsimiseen, kun sama toiminto on sijoitettu näytöllä erilaisen elementin taakse.

Muistilistan tunnistaminen osoittautui käyttäjille hankalaksi, mikä johtui myöskin tuotteen ja tosielämän vastaavuuden puutteesta. Iso kehystetty vihreä alue näytöllä ei viesti vielä käyttäjälle paljoka siitä, että kyseessä olisi tekstin tuottamiseen tarkoitettu kenttä.

7.4 Ratkaisuehdotukset

Takaisin-painikkeiden erilaisuuteen on olemassa selkeä ratkaisu, jotta kaikkialla sovelluksen rajaamassa käyttöympäristössä päästäisiin liikkumaan takaisin samalla tavalla. Kaikki ylimääräiset Takaisin-painikkeet natiivin painikkeen lisäksi on poistettava ja kaikissa näkymissä on oltava näkyvillä natiivi navigointipalkki näytön alaosassa. Tämä sen takia, että käyttäjän suunnistaessa aloitusnäky-
mästä kolmannen osapuolen sovellukseen ainut tapa suunnistaa takaisin on käyttämällä natiivia Takaisin-painiketta. Sen takia natiivin takaisin paluun käyttäminen myös aloitusnäyttösovelluksessa itsessään olisi kaikista loogisin vaihtoehto.

Sovelluksen käyttäessä räätälöityä selainäkymää, puuttuu selaimen yläosasta sen oma työkalupalkki kokonaan. Tämän tilalla oleva oma työkalupalkki ei sisällä Seuraava-painiketta, mikä rajoittaa käyttäjän liikkumista selaimessa. Ongelma voidaan korjata lisäämällä työkalupalkkiin uusi elementti, joka on Seuraava-painike. Painike tulee muistaa sijoittaa mahdollisimman loogisesti muiden työkalupalkin elementtien sekaan.

Opastuksen ja ohjeistuksen puuttuminen korostuu varsinkin silloin, kun sovelluksen käytettävyydessä on puutteita ja puutteet johtavat virhetilanteisiin. Käyttäjän tehdessä virheitä, tulee sovelluksen ilmoittaa virheen lisäksi, miten seuraavaksi tulee toimia. Virheviesteihin tulee siis lisätä käyttäjää opastavaa tekstiä. Sen lisäksi sovellukseen voidaan lisätä eri näkymiin elementit, joista pääsee sovelluksen ohjeisiin, jotka voivat sijaita omassa näkymässään.

Käyttäjän mahdollisuutta päästä käsiksi laitteen muihin toiminnallisuuksiin aloitusnäyttösovelluksen lisäksi pitää rajoittaa estämällä laitteen tilapalkin pyyhkäiseminen alas. Tämä voidaan hoitaa pitämällä tilapalkki piilotettuna halutuissa tilanteissa ohjelmallisesti. Myös navigointipalkissa sijaitseva painike, joka avaa viimeksi käytetyt sovellukset näkyviin, voidaan ohjelmallisesti piilottaa, jottei käyttäjä vahingossakaan joudu sovelluksen rajaaman ympäristön ulkopuolelle.

Joitain painikkeita ei tunnistettu ja näin ollen niitä ei myöskään käytetty tarkoituksenmukaisesti. Painikkeista tulisi tehdä enemmän painikkeiden näköisiä, että

käyttäjää selvästi erottaa ne elementeiksi, joita voi painaa. Tämä voidaan saavuttaa esimerkiksi lisäämällä painikkeisiin syvyyttä ja saada ne näyttämään siltä kuin ne olisivat hiukan koholla, kuten painikkeet reaali maailmassakin ovat. Kaikissa painikkeissa tulisi myös säilyttää sama teema, jotta käyttäjä osaa tunnistaa kaikissa näkymissä elementit painikkeiksi käytettyään yhtä painiketta kerran.

Muistilistaa voidaan parantaa tekemällä siitä käyttäjälle enemmän oikean muistilistan näköinen. Siihen voidaan lisätä viivoitus riveittäin, kuten osassa oikeitakin vihkoja on. Sen lisäksi muistilistan ollessa tyhjä, kun käyttäjä ei ole siihen vielä mitään kirjoittanut, voidaan listaan lisätä väliaikainen täyteteksti hieman himmeämmällä värillä. Teksti voi kertoa muutamalla sanalla käyttäjälle, että siihen voi kirjoittaa. Tätä käyttäjän on vaikea olla huomaamatta ja se voi avustaa käyttäjää painamaan muistilistasta, jolloin näppäimistön automaattisesti ilmestyessä painalluksen jälkeen käyttäjä varsin tajuaa, että hänen on mahdollisuus kirjoittaa kyseiseen kenttään.

8 POHDINTA

Testien ja arviointien avulla saatiin esille paljon hyödyllistä tietoa aloitusnäyttösovelluksen käytettävyydestä. Tulokset olivat kattavat ja niiden avulla pystyttiin miettimään käyttäjäkokemuksen parantamiseksi tarvittavia toimia. Täytyy kuitenkin muistaa, että arvioinnin suoritti kolme henkilöä ja käyttötestin kaksi henkilöä, mistä johtuen tulokset eivät ole täysin aukottomia. Testitapauksissa kaikki muis-tiinpanot kirjattiin käyttäjän suoriutumisen perusteella ja yksikään testikäyttäjä ei varmasti suorittaisi kaikkia tehtäviä samalla tavalla, vaikka testikäyttäjiä olisi ollut kymmeniä. Tämän takia voidaan todeta, että tutkimus tuotti arvokkaita tuloksia sovelluksen jatkokehittämisen kannalta, mutta kaikkia käytettävyysoongelmia ei varmastikaan löydetty. Sovelluksen jatkokehitysvaiheissa tässä työssä suoritettut testit ja arviot on mahdollista myös suorittaa uudestaan, jolloin saataisiin varmasti erilaisia tuloksia sovelluksen muututtua.

Ongelmien laadun ja määrän perusteella voidaan sanoa, että sovelluksessa on paljonkin parannettavaa käytettävyydessä. Suurin osa ongelmista painottui vakavuusasteikon vakavampaan päähän, mikä viestii kehittäjille käytettävyyden kehittämisen tarpeesta. Jos suurin osa ongelmista olisi keskittynyt vähäisten, kosmeettisten ja vain häiritsevien ongelmien kohdalle, voisi käytettävyyden parantamiseen riittää käyttöliittymän ja toimintojen hiominen, mutta suuremmat käytettävyysongelmat voivat vaatia isompiakin rakenteellisia muutoksia sovelluksessa.

Katastrofaalisten ja vakavien ongelmien ratkaisemisella saavutetaan kaikista eniten kehitystä käyttäjäkokemuksen parantamisen kannalta. Jos sovellusta meina-taan jatkokehittää, kannattaa näihin ongelmiin pureutua ensimmäisinä. Vähäisten, kosmeettisten ja vain häiritsevien ongelmien korjaamisen tärkeyttä voidaan arvioida uudestaan sen jälkeen, kun ylimmän prioriteetin ongelmat on ratkaistu.

Kokonaisuutena työ ja tulosten analysoiminen onnistui ongelmitta. Mitään en-nakko-odotuksia ei tulosten kannalta ollut ja kaikki saadut tiedot auttavat sovel-luksen kehittämisessä. Tämän dokumentin käytettävyysongelmille luotujen rat-kaisujen avulla sovellusta kehittäessä sen käytettävyyden tulisi parantua huomattavasti, mikä voidaan varmasti huomata tulevaisuuden testeissä.

LÄHTEET

Android Developers. N.d. Accessibility overview. Luettu 23.2.2019. <https://developer.android.com/guide/topics/ui/accessibility>

Android Developers. N.d. Design for Android. Luettu 22.2.2019. <https://developer.android.com/design/>

Association for Computing Machinery. 2014. People of ACM – Ben Shneiderman. Luettu 28.2.2019. <https://www.acm.org/articles/people-of-acm/2014/ben-shneiderman>

Budiu, R. 2014. Memory Recognition and Recall in User Interfaces. Luettu 5.3.2019. <https://www.nngroup.com/articles/recognition-and-recall/>

Harley, A. 2018. Visibility of System Status. Luettu 4.3.2019. <https://www.nngroup.com/articles/visibility-system-status/>

Interaction Design Foundation. N.d. User Experience (UX) Design. Luettu 1.2.2019. <https://www.interaction-design.org/literature/topics/ux-design>

Kaley, A. 2018. Match Between the System and the Real World: The 2nd Usability Heuristic Explained. Luettu 4.3.2019. <https://www.nngroup.com/articles/match-system-real-world/>

Laubheimer, P. 2015. Preventing User Errors: Avoiding Unconscious Slips. Luettu 4.3.2019. <https://www.nngroup.com/articles/slips/>

Liyanage, E. 2016. 10 Usability heuristics explained. Luettu 4.3.2019. <https://medium.com/@erangat/10-usability-heuristics-explained-caa5903faba2>

Material Design. N.d. Accessibility. Luettu 23.2.2019. <https://material.io/design/usability/accessibility.html>

Mishra, R. 2018. This is all you need to know to conduct a UX Survey. Luettu 19.3.2019. <https://uxplanet.org/this-is-all-you-need-to-know-to-conduct-a-ux-survey-50400af45920>

Nielsen, J. 1994. 10 Usability Heuristics for User Interface Design. Luettu 28.2.2019. <https://www.nngroup.com/articles/ten-usability-heuristics/>

Nielsen, J. 1999. Do Interface Standards Stifle Design Creativity? Luettu 4.3.2019. <https://www.nngroup.com/articles/do-interface-standards-stifle-design-creativity/>

Nielsen, J. 2001. Error Message Guidelines. Luettu 5.3.2019. <https://www.nngroup.com/articles/error-message-guidelines/>

Nielsen, J. 2012. Usability 101: Introduction to Usability. Luettu 17.2.2019. <https://www.nngroup.com/articles/usability-101-introduction-to-usability/>

Nielsen Norman Group. N.d. Jakob Nielsen. Luettu 28.2.2019.

<https://www.nngroup.com/people/jakob-nielsen/>

POLUT. N.d. Käyttötuotteen heuristinen arviointi. Luettu 27.2.2019.

http://mlab.uiah.fi/polut/Design/tyokalu_heuristinen_arvio.html

Riihiaho, S. N.d. Käytettävyyden arviointi ilman käyttäjiä. Luettu 27.2.2019.

<http://www.soberit.hut.fi/T-121/T-121.600/asiantuntija-arviot.pdf>

Riihiaho, S. 2000. Experiences with usability evaluation methods. Luettu

27.02.2019. https://www.researchgate.net/publication/2485077_Experiences_with_Usability_Evaluation_Methods

Roto, V., Law, E., Vermeeren, A. & Hoonhout, J. 2011. User Experience White

Paper. Luettu 9.2.2019. <http://www.allaboutux.org/files/UX-WhitePaper.pdf>

Segue Creative Team. 2017. What is Usability and Why is it Important to Appli-

cation Development? Luettu 18.2.2019. <https://www.seguetech.com/usability-application-development/>

Shape. 2018. 5 Reasons why UX design is important. Luettu 7.4.2019.

<http://shape.ma/5-reasons-why-ux-design-is-important/>

Shneiderman, B. N.d. The Eight Golden Rules of Interface Design. Luettu

28.2.2019. <http://www.cs.umd.edu/~ben/goldenrules.html>

Torp, T. N.d. Asiantuntija-arviointi - lomake. Ohjelmistotekniikan mobiiliprojekti - opintojakson oppimateriaali. DOCX-tiedosto. Tabula. TAMK.

Torp, T. N.d. Käytettävyyden arviointi – verkkoluento 13 min. Ohjelmistotekniikan mobiiliprojekti -opintojakson oppimateriaali. MP4-tiedosto. Tabula. TAMK.

UserTesting. 2018. UI vs UX: What's the difference between user interface and

user experience? Luettu 15.2.2019. <https://www.usertesting.com/blog/ui-vs-ux/>

Wong Euphemia. 2018. Shneiderman's Eight Golden Rules Will Help You De-

sign Better Interfaces. Luettu 1.3.2019. <https://www.interaction-design.org/literature/article/shneiderman-s-eight-golden-rules-will-help-you-design-better-interfaces>

LIITTEET

Liite 1. Asiantuntija-arvioinnissa käytetty lomake

1 (7)

Asiantuntija-arviointi

Tekijä:

Päivämäärä:

Lisää taulukoihin tarvittaessa rivejä tai poista turhat rivit. Hyödynnä annettuja apukysymyksiä.

1. Ongelmien listaus

1. Järjestelmän tilan näkyvyys

Ongelma	Kuvaus	Miten rikkoo heuristiikkaa
1.1		
1.2		
1.3		
...		

- Onko käyttäjä tietoinen mitä tuotteen käytössä tapahtuu?
- Tarjoaako tuote sopivaa palautetta, tuleeko palaute nopeasti?

2. Tuotteen ja tosielämän vastaavuus

Ongelma	Kuvaus	Miten rikkoo heuristiikkaa
2.1		
2.2		

2.3		
...		

- Käytetäänkö tuotteessa käyttäjälle tuttua kieltä, sanoja, lauseita ja konsepteja vai tuoteorientoituneita termejä?
- Käytetäänkö tosielämän käytäntöjä, esiintyykö tieto luonnollisessa ja loogisessa järjestyksessä?

3. Käyttäjän kontrolli ja vapaus

Ongelma	Kuvaus	Miten rikkoo heuristiikkaa
3.1		
3.2		
3.3		
...		

- Valitseeko käyttäjä vahingossa ominaisuuksia?
- Tarjoaako tuote ulospääsyn ei-halutusta tilasta ilman pitkää dialogia?
- Tarjoaako tuote undo- ja redo-ominaisuudet?

4. Yhdenmukaisuus

Ongelma	Kuvaus	Miten rikkoo heuristiikkaa
4.1		
4.2		
4.3		

...		
-----	--	--

- Joutuuko käyttäjä ihmettelemään tarkoittavatko eri sanat, tilanteet ja toiminnot samoja asioita?
- Onko tuotteessa yhteneväiset toimintatavat?

5. Virheiden ehkäisy

Ongelma	Kuvaus	Miten rikkoo heuristiikkaa
5.1		
5.2		
5.3		
...		

- Hyvien virheilmoitusten sijaan tärkeää on, että virheitä ei esiinny. Miten tämä toteutuu tuotteessa?

6. Tunnistaminen mieluummin kuin muistaminen

Ongelma	Kuvaus	Miten rikkoo heuristiikkaa
6.1		
6.2		
6.3		
...		

- Ovatko objektit, toiminnot ja valinnanmahdollisuudet näkyviä?
- Täytyykö käyttäjän muistaa tietoa eri toiminnallisuuksien välillä?
- Ovatko käyttöohjeet näkyvissä tai helposti saatavissa aina kuin niitä tarvitaan?

7. Joustavuus ja tehokkuus

Ongelma	Kuvaus	Miten rikkoo heuristiikkaa
7.1		
7.2		
7.3		
...		

- Tukeeko tuote sekä kokeneita että uusia käyttäjiä? Onko kokeneille käyttäjille tarjolla oikopolkuja, jotka nopeuttavat tuotteen käyttöä?
- Mahdollistaako tuote usein käytettyjen toimintojen räätälöinnin?

8. Estetiikka ja minimalistinen suunnittelu

Ongelma	Kuvaus	Miten rikkoo heuristiikkaa
8.1		
8.2		
8.3		
...		

- Sisältävätkö toiminnallisuudet tietoa, joka ei ole oleellista tai sitä harvoin tarvitaan? Jokainen "turha" tietosisältö kilpailee oleellisen tiedon kanssa ja vähentää oleellisen tiedon näkyvyyttä.

9. Virhetilanteista palautuminen

Ongelma	Kuvaus	Miten rikkoo heuristiikkaa
9.1		
9.2		
9.3		
...		

- Ilmoitetaanko virhetilanteet selkokielellä (ei virhekoodeilla)?
- Kertovatko virhetilanteet selkeästi ongelman?
- Ehdottaako tuote selkeän ratkaisun ongelmaan?

10. Opastus ja ohjeistus

Ongelma	Kuvaus	Miten rikkoo heuristiikkaa
10.1		
10.2		
10.3		
...		

- Tarjoaako tuote opastusta ja ohjeistusta, vaikka tuotteen on hyvä toimia ilman käyttöohjeita?
- Onko opastukseen liittyvä tieto helposti löydettävissä, onko se fokusoitunut käyttäjän tehtäviin, kertooko se askeleet, miten jatketaan ja ovatko ne riittävän lyhyet?

3. Tärkeimpien ongelmien valinta ja ratkaisuehdotukset

Valitse viisi omasta mielestäsi tärkeintä ongelmaa ja listaa ne alla olevaan taulukkoon. Anna oma myös oma ratkaisuehdotuksesi ongelmalle.

Ongelma	Ongelman prioriteetti	Ratkaisu
X.X		
X.X		
X.X		
X.X		
X.X		

Liite 2. Käyttäjätestauksessa käytetty lomake

1 (5)

Käyttäjätestaus

Testattavan sovelluksen pohjustus testikäyttäjälle sekä testin rakenne:

Testattavana on isokokoisille Android-laitteille suunniteltu sovellus. Sovelluksen tarkoitus on tuoda Iloi.fi-verkkosivuston elämyspankin tarjoamat elämykset käytettäväksi myös tableteille.

Ensiksi pidetään pienimuotoinen esikysely, jonka jälkeen suoritetaan varsinaiset testitehtävät ja lopuksi vielä pidetään jälkikysely sovelluksen käytöstä.

Onko jotain, jota haluaisit kysyä ennen aloittamista?

1. Esikysymykset

- Kuinka hyvin tunnet Android-käyttöjärjestelmän?
 - 1) En ole ikinä kuullutkaan Androidista.
 - 2) Olen kuullut Androidista, mutta en tiedä siitä sen syvällisemmin.
 - 3) Tiedän millainen Android-käyttöjärjestelmä on.

- Edelliseen lisäten, oletko koskaan käyttänyt Android-laitteita?
 - 1) En ole tai en ainakaan tietoisesti.
 - 2) Kyllä olen, mutta en itse omista Android-laitteita.
 - 3) Kyllä olen ja omistan itsekin ainakin yhden Android-laitteen.

- Oletko koskaan aikaisemmin käyttänyt tabletteja eli taulutietokoneita.
 - 1) En tai en ainakaan tietoisesti.
 - 2) Olen käyttänyt yhden tai vain muutaman kerran.
 - 3) Olen käyttänyt useita kertoja.

- Kuinka hyvin uskot suoriutuvasi testitehtävistä tietämättä vielä tehtäviä?
 - 1) Huonosti, sillä en usko, että tiedän sovelluksesta tarpeeksi käyttäkseni sitä sujuvasti.
 - 2) Melko hyvin, sillä uskon, että sovellusta on jokseenkin helppo käyttää.
 - 3) Todella hyvin, sillä uskon, että sovellusta on helppo käyttää.

2. Testitehtävät

Testitehtäviä suorittaessa suorituksesta mitataan siihen käytettävä aika. Uusi tehtävä jatkuu aina vanhan tehtävän lopputilanteesta. Jos käyttäjä joutuu umpikujaan, eikä saa tehtävää suoritettua onnistuneesti, merkitään käytetyksi ajaksi viiva ja testiä pitävä henkilö asettaa sovelluksen seuraavan tehtävän lähtötilanteeseen.

- 1) Haluat tietää enemmän lihaskunnosta. Etsi Iloi.fi-elämyspankista haluamasi lihaskuntovideo ja aloita sen katselu. Katseltuasi videota muutaman sekunnin ajan pysäytä video.

Lähtötilanne: Sovelluksen aloitusnäkyvä
Lopputilanne: Videonäkymä, jossa pysäytetty video

Käytetty aika sekunteina:

Huomioita/ongelmia:

- 2) Palaa takaisin sovelluksen aloitusnäkyvä.

Lähtötilanne: Videonäkymä
Lopputilanne: Sovelluksen aloitusnäkyvä

Käytetty aika sekunteina:

Huomioita/ongelmia:

- 3) Tarkista haluamasi paikkakunnan tämänhetkinen sää ilmatieteenlaitoksen sivuilta ja kirjoita se ylös sovelluksen aloitusnäkyvän muistilistalle.

Lähtötilanne: Sovelluksen aloitusnäkyvä

Lopputilanne: Sovelluksen aloitusnäkyvä, jossa muistilistaan kirjoitettuna säätilatieto

Käytetty aika sekunteina:

Huomioita/ongelmia:

- 4) Mene haluamallesi verkkosivulle. Verkkosivulle päästyäsi suunnista Iloi.fi-elämyspankin etusivulle.

Lähtötilanne: Sovelluksen aloitusnäkyvä

Lopputilanne: Iloi.fi-elämyspankin etusivu

Käytetty aika sekunteina:

Huomioita/ongelmia:

- 5) Haluat katsella vielä lisää videoita Iloi.fi-elämyspankista. Katso kolmea haluamaasi videota, kutakin noin kymmenen sekuntia ennen uuden videon etsimistä. Katsottuasi kolmea videota palaa sovelluksen aloitusnäkympään.

Lähtötilanne: Iloi.fi-elämyspankin etusivu
Lopputilanne: Sovelluksen aloitusnäkympä

Käytetty aika sekunteina:

Huomioita/ongelmia:

- 6) Katso Yle Areenasta hetki haluamaasi videota ja palaa sen jälkeen aloitussivulle.

Lähtötilanne: Sovelluksen aloitusnäkympä
Lopputilanne: Sovelluksen aloitusnäkympä

Käytetty aika sekunteina:

Huomioita/ongelmia:

3. Jälkikysymykset

- Kuinka kuvailisit sovellusta muutamalla sanalla?

Vastaus:

- Turhauttiko sovelluksen käyttäminen sinua jossain vaiheessa? Jos turhautti, niin missä vaiheessa?

1) Kyllä

Kommentti:

2) Ei

- Mistä tykkäsit eniten/vähiten sovelluksessa?

Eniten:

Vähiten:

- Oliko sovellusta mielestäsi helppo käyttää?

1) Kyllä

2) Ei

- Jos voisit muuttaa yhden asian sovelluksessa, mikä se olisi?

Vastaus:

- Paljonko antaisit sovellukselle kokonaisarvosanaksi asteikolla 0-10?

Arvosana: