

Material Design -muotokielen käyttäminen Web-sovelluksen suunnittelussa

Joonas Nahkala

Opinnäytetyö

Lokakuu 2018

Liiketalouden ala

Tradenomi (AMK), tietojenkäsittelyn tutkinto-ohjelma

Tekijä(t) Nahkala, Joonas	Julkaisun laji Opinnäytetyö, AMK	Päivämäärä Lokakuu 2018
	Sivumäärä 43	Julkaisun kieli Suomi
		Verkojulkaisulupa myönnetty: x
Työn nimi Material Design -muotokielen käyttäminen Web-sovelluksen suunnittelussa		
Tutkinto-ohjelma Tietojenkäsittelyn tutkinto-ohjelma		
Työn ohjaaja(t) Kiviaho, Niko		
Toimeksiantaja(t) Growthsetters Oy		
Tiivistelmä <p>Sovelluksen käyttökokemukseen vaikuttaa käyttöliittymään liittyvät käytettävyysongelmat. On tärkeää karsia käytettävyysongelmat käyttöliittymästä, jotta käyttäjän sovelluksen testaus ei jää lyhyeksi kokeiluksi.</p> <p>Usein käytettävyysongelmat karsitaan kehittämällä uusia versioita sovelluksesta ja testaamalla ratkaisuja käyttäjillä, kunnes saadaan haluttu käytettävyyden taso. Käytettävyysongelmien karsintaan kuluu paljon aikaa, minkä takia tutkittiin, miten valmiiksi testattuja suunnitteluperiaatteita voi hyödyntää uuden sovelluksen suunnittelussa.</p> <p>Tutkimuksen tavoitteena oli suunnitella käyttöliittymä Googlen Material Design -muotokielen avulla ja selvittää, miten muotokielen käyttö integroidaan suunnitteluprosessiin. Tutkimuksessa pyrittiin selvittämään myös, miten käytettävyys ilmenee Material Designin suunnitteluperiaatteissa.</p> <p>Tutkimus toteutettiin kvalitatiivisena tutkimuksena. Siinä pyrittiin selvittämään, minkälaisia käytettävyysongelmia käyttäjätestauksella havaittiin. Tutkimus antoi tietoa sivujen rakenteiden luontiin, sisällön asetteluun, visuaaliseen tyyliin ja tarkoituksellisten animaatioiden luontiin Material Designissa.</p> <p>Tutkimuksen tuloksena syntyi toimiva prototyyppi sovelluksesta. Tuloksista havaittiin, että Material Designin avulla on mahdollista luoda käytettävä pohja sovelluksen käyttöliittymälle. Käyttäjätestauksesta havaittiin, että sovelluskohtaisten toiminallisuuksien kanssa kannattaa arvioida käytettävyyttä heuristiikoilla ja testata ominaisuuksia käyttäjillä.</p>		
Avainsanat (asiasanat) Käyttöliittymäsuunnittelu, käytettävyys, käyttäjakeskinen suunnittelu, material design		
Muut tiedot		

Author(s) Nahkala, Joonas	Type of publication Bachelor's thesis	Date October 2018 Language of publication: Finnish
	Number of pages 43	Permission for web publication: x
Title of publication Using Google Material Design to design a Web application		
Degree programme Business Information Technology		
Supervisor(s) Kiviaho, Niko		
Assigned by Growthsetters Oy		
Abstract <p>User experience with an application depends on the usability issues related to the application. It is important to resolve usability issues so that the user's trial run of the application will not be a short one.</p> <p>Usability issues are often resolved by developing new versions of the application and testing the newly developed solutions with users until a desired standard of usability is reached. It can take a great deal of time to fix all the usability problems in an application, which is why the research was conducted on how already established design guidelines could be used in designing a new application.</p> <p>The purpose of the study was to design an application using Google's Material Design system and understand how using Material Design fits into the design process. Another goal of the study was to understand how usability is valued in the Material Design principles.</p> <p>The research was carried out with the qualitative method, and it uncovered what types of usability issues were found through user testing. The research also shed light on how pages are structured, how content is laid out, how visual the style is and how purposeful animations are created in Material Design.</p> <p>The result of the study was a usable prototype of the application. The research clarified that the guiding principles in Material Design can be used to make a foundation for a usable application. User testing revealed that the application specific functionalities are still subject to usability issues, and that they should go through heuristic analysis and user testing.</p>		
Keywords/tags (subjects) User interface design, usability, user-centered design, material design		
Miscellaneous		

Sisältö

1	Johdanto	5
2	Tutkimusasetelma	5
2.1	Opinnäytetyön toimeksiantaja ja tavoitteet	5
2.2	Tutkimusongelma ja -kysymykset	6
2.3	Tutkimusmenetelmät	6
3	Teoreettinen viitekehys	6
3.1	Käytettävyys	6
3.2	Käyttäjäkokemus	8
3.3	Käyttäjäkeskeinen suunnittelu	8
3.4	Konventiot	9
3.5	Käyttäjätestaus	9
3.6	Heuristinen arviointi	10
3.7	Material Design	12
3.7.1	Ympäristö	12
3.7.2	Rakenne	17
3.7.3	Navigaatio	20
3.7.4	Väri ja typografia	22
3.7.5	Liike ja transformointi	26
4	Käyttöliittymän suunnittelu	28
4.1	Vaatimusmäärittely	28
4.2	Rautalankamallit	29
4.3	Prototypisointi	31
4.4	Käyttäjätestaus	34

5 Tulokset	35
5.1 Lopullinen käyttöliittymä	35
5.2 Käyttäjätestauksen tulokset	38
5.3 Material Design suunnitteluprosessissa.....	41
6 Pohdinta.....	42
Lähteet	43

Kuviot

Kuvio 1. Materiaalin muuttuva korkeus ja leveys	14
Kuvio 2. Taustan tummentaminen.....	15
Kuvio 3. Korkeustason ilmaiseminen varjoilla.....	16
Kuvio 4. Varjon intensiivisyys	17
Kuvio 5. Palstat, urat ja marginaalit	18
Kuvio 6. Inkrementti.....	20
Kuvio 7. Hierarkia navigoinnissa.....	21
Kuvio 8. Värien käyttö eri elementeissä.....	23
Kuvio 9. Tekstin korostaminen läpinäkyvyydellä	24
Kuvio 10. Material Designin kirjasinskaala	25
Kuvio 11. Kirjasinskaalan käyttö.....	26
Kuvio 12 Liikkeen kaari näytöllä	27
Kuvio 13 Rautalankamalli etusivusta ja tehtäväpaketin valinnasta	30
Kuvio 14. Ensimmäinen prototyyppi	31
Kuvio 15 Painikkeen varjo	32
Kuvio 16 Kirjautumissivun lomakkeet	32
Kuvio 17 Väriteema	33
Kuvio 18 Tekstin painotus	34
Kuvio 19 Kirjautumissivu	35
Kuvio 20 Etusivu ja ryhmän sekä tilin hallinta.....	36

Kuvio 21 Haastetehtävät ja haasteen ohjeet	37
Kuvio 22 Liikkeen suoritus ja ohjeet.....	38
Kuvio 23 Taustan tummennus prototyyppissä.....	39
Kuvio 24 Liikkeiden selaus	40
Kuvio 25 Haastetehtävien avaus	41

Käsitteet

Affordansi: Havaittava toiminnan mahdollisuus sovelluksen elementissä.

Asettelu: Tekstin tai kuvien järjestys sivulla

Iterointi: Tuotteen kehitysmenetelmä, jossa toistetaan työvaiheita, kuten toimivan version tuottamista ja sen arviointia, kunnes saadaan haluttu tulos.

Järjestelmä: Tietokoneohjelma tai palvelukokonaisuus

Käyttäjäkokemus (UX): Käyttäjän tunteet tuotteen käytön suhteen.

Käyttöliittymä (UI): Kokoelma mekanismeja, joiden avulla käyttäjä ja laite ovat vuorovaikutuksessa keskenään.

Mobiili: Puhelimille ja tableteille tarkoitettu ohjelmistoympäristö.

Pysäytyspiste: Ennalta määritetty piste, jossa toiminto vaihdetaan tai keskeytetään.

Prototyyppi: Kokeiluversio sovelluksesta

Työpöytä: Työpöytä tietokoneille tarkoitettu ohjelmistoympäristö.

Tietoarkkitehtuuri: Tapa, jolla tieto strukturoidaan järjestelmässä.

1 Johdanto

Käytettävyys käyttöliittymissä vaikuttaa tuotteen ensivaikutelmaan ja käyttäjäkokemukseen paljon. Tuotteen käytettävyyteen on varattava resursseja suunnitteluvaiheessa. Käyttöliittymäsuunnitteluun kuluu ylimääräistä aikaa ja resursseja silloin, kun käytettävyysongelmiin ei osata tuottaa toimivia ratkaisuja. Tähän on avuksi luotu suunnitteluohjeita, kuten Googlen Material Design -muotokieli.

Tämän tutkimuksen tavoitteena on tutkia, miten käytettävyys ilmenee Material Design -muotokielessä sekä miten sitä voidaan hyödyntää käyttöliittymän suunnittelussa. Hyvin suunniteltu käyttöliittymä antaa käyttäjän suorittaa heidän tehtävänsä tehokkaasti. Miellyttävä käyttäjäkokemus tarkoittaa myös käyttäjän todennäköistä paluuta samaan sovellukseen sen kokeilun jälkeen.

Tehtävänä on suunnitella käyttöliittymä Web-sovellukseen käyttämällä Googlen nykyisiä suunnitteluperiaatteita. Työssä käydään läpi käyttöliittymän suunnitteluprosessi läpi, jonka lopuksi testataan sovelluksen käytettävyyttä käyttäjätestauksella.

2 Tutkimusasetelma

Tässä luvussa kerrotaan, mitä työssä tutkitaan, millä tutkimusmenetelmällä tutkitaan sekä mikä on tutkimuksen tavoite.

2.1 Opinnäytetyön toimeksiantaja ja tavoitteet

Tutkimuksen toimeksiantaja on digitaalisten palveluiden kehittäjä Growthsetters Oy. Toimeksiantaja kehittää Game of Skills -nimistä sovellusta, johon tehtävänä on suunnitella käyttöliittymä noudattaen Material Design -muotokielen suunnitteluperiaatteita ja testaamalla käyttäjillä, onko käyttäjäkokemus miellyttävä.

2.2 Tutkimusongelma ja -kysymykset

Tutkimusongelma on, miten käyttöliittymän suunnitteluprosessiin voidaan integroida Material Design -muotokielen käyttö.

Tutkimuskysymykset ovat:

- Mikä on käytettävyyden merkitys Material Design -muotokielessä?
- Miten Material Design -muotokieltä voidaan hyödyntää Game of Skills -sovelluksen käyttöliittymän suunnittelussa?

2.3 Tutkimusmenetelmät

Tutkimuksessa käytetään kvalitatiivista eli laadullista tutkimusmenetelmää. Laadullisessa tutkimuksessa pyritään johtopäätöksiin ilman tilastollisia menetelmiä tai muita määrällisiä keinoja. Tällä menetelmällä pyritään ymmärtämään ilmiö syvällisesti ja tuottaa kerätystä aineistosta järkevä tulkinta tutkimuksen tuloksissa. (Kananen 2008, 24.)

Käytettävyyttä ja käyttäjäkokemusta arvioidaan aluksi heuristisella arvioinnilla tuottamalla useita iteraatioita käyttöliittymästä, kunnes tavoitetaan haluttu käytettävyyden taso. Käyttöliittymästä luodaan interaktiivinen prototyyppi, jota voidaan testata käyttäjillä.

3 Teoreettinen viitekehys

Tässä luvussa käsitellään tutkimuksen tietoperustaa. Ensin käydään läpi käytettävyys, jonka jälkeen esitellään mitä on käyttäjäkeskeinen suunnittelu, esteettisyys ja Material Design.

3.1 Käytettävyys

Käytettävyydellä tarkoitetaan tuotteen käyttäjän kykyä käyttää tuotetta sen suunniteltuun käyttötarkoitukseen ilman tarpeettoman pitkää perehtymistä tuotteeseen (Krug 2006, 5).

Hyvä käytettävyys säästää yrityksen resursseja. Ominaisuus on halvempi vaihtaa sen suunnitteluvaiheessa sen sijaan, että käytetään resursseja huonoksi todetun ominaisuuden korjaamiseen sen käyttöönoton jälkeen. (Sinkkonen, Kuoppala, Parkkinen & Vastamäki 2006, 15.)

Nielsen (1993, 26) määrittelee 5 piirrettä käytettävyydelle seuraavasti:

- Opittavuus – Kuinka nopeasti käyttäjä oppii suorittamaan perustehtävät järjestelmässä?
- Tehokkuus – Kuinka tehokkaasti järjestelmän kokenut käyttäjä kykenee suorittamaan perustehtäviä?
- Muistettavuus – Osaako käyttäjä käyttää järjestelmä käyttötauon jälkeen?
- Virheettömyys – Kuinka monta virhetta käyttäjä tekee järjestelmää käyttäessä ja kuinka vakavia kyseiset virheet ovat?
- Miellyttävyys – Kuinka miellyttävä järjestelmä on käyttää?

Opittavuus

Opittavuus on yksi käytettävyyden olennaisimpia piirteitä, koska useimmat järjestelmät tulisi olla helppo oppia ja käyttäjän ensimmäinen kokemus järjestelmän kanssa koostuu käyttöliittymän oppimisesta. Opittavuus on myös helpoin piirre mitata. Siinä katsotaan, kuinka pitkään käyttäjällä kuluu aikaa saavuttaa määritetyn pätevyyden tason järjestelmän kanssa. Pätevyyden taso voi olla käyttäjän kyky suorittaa jokin tehtävä järjestelmässä. (Mts. 27–29.)

Tehokkuus

Tehokkuus viittaa kokeneen käyttäjän kykyyn suorittaa tehtäviä tehokkaasti. Järjestelmän tehokkuutta voidaan mitata määrittelemällä aluksi tietty taso, jolla käyttäjä voidaan todeta kokeneeksi. Sitten testataan, kuinka paljon aikaa kuluu tyypillisiin tehtäviin kokeneiden käyttäjien kanssa. (Mts. 30.)

Muistettavuus

Muistettavuus on tärkeä piirre käytettävyydelle silloin kun sovellus ei kuulu käyttäjän päivittäiseen käyttöön, mutta on silti tarpeellinen esimerkiksi luomaan raportteja muutaman kuukauden välein. Muistettavuutta mitataan testaamalla käyttötauolta

palaneen käyttäjän ajankäyttöä tyypillisten tehtävien suorittamiseen. Testin jälkeen käyttäjälle voidaan tehdä myös muistitesti. Muistitestissä kysytään käyttäjältä nimetä tai piirtää toiminnon kuvake sen kuvauksen perusteella. (Mts. 31–32.)

Virheettömyys

Käyttäjän tekemät virheet tulisi pitää minimissä. Virheellä viitataan mihinkä tahansa tekoon, jolla käyttäjä ei saavuta haluamaansa tavoitetta. Näiden tekojen yhteenlaskulla voidaan mitata virheettömyys. Kaikkia virheitä ei tarvitse laskea, kuten sellaisia, jotka eivät vaikuta käyttäjän työhön. Vakavimmat virheet lasketaan erikseen ja niiden estäminen pitäisi olla korkein prioriteetti virheettömyyttä ajatellessa. Näihin lasketaan virheet, joita käyttäjä ei huomaa tai jotka tuhoavat käyttäjän työn. (Mts. 32–33.)

Miellyttävyys

Miellyttävyys on tärkeä muistaa silloin, kun järjestelmää suunnitellaan vapaa-ajan käyttöön. Yksinkertaisin tapa mitata miellyttävyttä on tehdä subjektiivinen kysely järjestelmän käyttökokemuksesta. Nämä ovat yleensä hyvin lyhyitä kyselyitä, joissa kysytään käyttäjältä, kuinka vahvasti he ovat samaa mieltä jonkin toteamuksen kanssa esimerkiksi 1-5 skaalalla. (Mts. 33–35.)

3.2 Käyttäjäkokemus

Käyttäjäkokemus (UX) tarkoittaa käyttäjän jokaista interaktiota tuotteen kanssa ja minkälaisia tunteita se herättää tuotteen käytön suhteen. Tähän vaikuttaa se kuinka hyödylliseksi, helpoksi ja miellyttäväksi käyttäjä tuntee tuotteen käytön olevan. Käyttäjäkokemukseen vaikuttaa myös asiat, jotka eivät suorastaan liity käytettävyyteen. Käyttäjäkokemukseen sovelluksessa vaikuttaa sen visuaalinen tyyli, brändi, äänet ja interaktiot. (Garret 2002, 3–4.)

3.3 Käyttäjäkeskeinen suunnittelu

Hyvä suunnittelu vaatii tehokasta vuorovaikutusta varsinkin koneesta ihmiseen. Hyvä vuorovaikutus kertoo käyttäjälle, mitkä toiminnot ovat saatavilla, mitä järjestelmässä tapahtuu ja mitä kukin toiminto tekee. Vuorovaikutus on yhä tärkeämpää silloin, kun käyttäjä tekee virheen sovelluksen käytössä. Hyvän suunnittelun erottaa huonosta

siitä, miten helposti käyttäjä voi toipua virhetilanteista. Virhetilanteen syntyessä järjestelmän pitäisi ilmoittaa käyttäjälle, mistä ongelma tarkalleen johtuu. Käyttäjä voi sitten suorittaa tarvittavat toimenpiteet virhetilanteesta palautumiseen. Tämän prosessin sujuvuus vaikuttaa käyttökokemukseen suuresti. (Norman 2002, 7–8.)

Käyttäjäkeskeinen suunnittelu on suunnittelufilosofia, jota hyödyntämällä varmistetaan siitä, että tuote suunnitellaan kohderyhmä mielessä ja että se palvelee käyttäjän tarpeita. Käyttäjän tarpeiden ymmärtäminen saavutetaan havainnoinnin avulla, koska käyttäjät eivät tarkalleen tiedä mistä ongelmista heidän tarpeet tulevat ja miten ne voidaan määritellä. Havainnoinnista luodaan arvio käyttäjän tarpeista, jota pyritään tarkemmin määrittämään iteroimalla ongelman näkökulmaa sekä määrittämään. Tuloksena on tuote, joka vastaa käyttäjän tarpeita. (Mts. 8.)

3.4 Konventiot

Konventiot ovat vakiintuneita käytäntöjä, joita käyttäjä opettelee käyttämään ensimmäistä kertaa sovelluksen kanssa. Konventioiden käytöllä voidaan tehdä käyttöliittymästä hyvin intuitiivinen. Esimerkkinä yksi vakiintunut käytäntö verkkokaupoissa on kuvakkeen käyttö ostoskärrylle, jonka käyttäjä voi nopeasti tunnistaa muissa verkkokaupoissa sen opetteluun jälkeen. Käytäntö vakiintuu yleensä vain silloin, jos se osoitetaan toimivaksi. (Krug 2006, 34.)

Vakiintuneilla käytännöillä on kuitenkin kulttuurillinen rajoitus, koska käyttäjän katse voi mieluiten etsiä Web-sivun valikoita jossain kulttuureissa näytön vasemmalta laidalta ja muissa näytön oikealta laidalta. Tähän vaikuttaa se luetaanko tekstiä kulttuurissa vasemmalta oikealle tai päinvastoin. Tästä syystä sovelluksen suunnittelussa on huomioitava kohderyhmä. (Sinkkonen, Kuoppala, Parkkinen & Vastamäki 2006, 37.)

3.5 Käyttäjätestaus

Käyttäjätestauksella on mahdollista saada määrällistä sekä laadullista tietoa testeistä. Molemmat ovat tärkeitä, koska ensimmäisellä mitataan käytettävyyttä ja toisella arvioidaan käyttäjäkokemusta. (Mitchell 2007, 3.)

Käyttäjätestaus suoritetaan suunnitteleamalla tehtäviä, joita käyttäjä tekisi oikeasti sovellusta käyttäessä. Testin vetäjä tarkkailee käyttäjän etenemistä jokaisen tehtävän kanssa, jonka aikana käyttäjää kehoitetaan ajattelemaan ääneen selvittääkseen miten he lähestyvät tuntematonta käyttöliittymää. (Mts. 3–4.)

Käyttäjätestauksessa ei yleensä ehdi testata kaikkea, joten monimutkaiset käyttöliittymän alueet priorisoidaan tehtävälstaan. Tehtävissä ei kannata käyttää samoja termejä mitä käyttäjä näkee käyttöliittymässä, koska se muuten johdattaa käyttäjän suoraan tehtävän loppuun. (Mts. 14–15.)

Testin vetäjä kirjaa ylös käyttäjän virheet, kuten esimerkiksi väärälle sivulle navigointi. Virheiden määrä ja tehtäviin käytetty aika ovat mittareita käyttöliittymän käytettävyydelle. Jokaisen tehtävän jälkeen voidaan kerätä subjektiivista dataa käyttäjältä kysymällä heidän mielipidettä tehtävän vaikeudesta esimerkiksi 1-10 skaalalla. Jokaisen käyttäjän testistä kootaan keskimääräiset tulokset, joista voidaan selvittää, mikä on käyttöliittymän heikoin alue. (Mts. 58.)

3.6 Heuristinen arviointi

Heuristisen arvioinnin tarkoituksena on parantaa sovelluksen käytettävyyttä ja käyttäjäkokemusta tarkistamalla, noudattaako se tunnistettuja käyttöliittymien suunnitteluperiaatteita. Heuristisella arvioinnilla voidaan tunnistaa virheitä jo prototyypin suunnittelun alkuvaiheessa. (Soini, 9–10.)

Arvioinnissa ei ole realistista olettaa löytävänsä kaikki käytettävyysongelmat sovelluksesta tekemällä arvioinnin yksin. Tämä johtuu siitä, että eri evaluoijat löytävät erilaisia ongelmia. Yksi evaluoija löytää keskimäärin 35% käytettävyysongelmista heuristisessa arvioinnissa. Suositeltu evaluoijien määrä on viisi, jonka jälkeen löydettyjen ongelmien määrä vähenee jokaista lisättyä evaluoijaa kohti. (Nielsen 1990, 156).

Nielsen (Nielsen 1995) käyttää seuraavia heuristiikkoja yleisinä suunnitteluperiaatteina käytettävyyden arviointiin:

Näytä järjestelmän tila

Järjestelmän tulisi aina pitää käyttäjän tietoisena sovelluksessa tapahtuvista asioista paikoissa, jossa se sopii. (Mt.)

Järjestelmän ja todellisen maailman vastaavuus

Järjestelmän pitäisi esittää tieto loogisessa järjestyksessä selkeällä kielellä. Käyttäjän on helpompi ymmärtää arkikieltä ja oikean eleman konventioita kuin teknisiä termejä, jotka ovat jääneet kehityksen jäljiltä käyttöön. (Mt.)

Käyttäjän kontrolli ja vapaus

Käyttäjät usein erehtyvät toiminnallisuuden valinnasta ja siitä syystä tarvitsevat paluu toiminnon viemään käyttäjän edelliseen tilaan järjestelmässä ilman pidennettyä dialogia. (Mt.)

Yhtenäisyys ja standardit

Käyttäjien ei tarvitse miettiä, onko eri sanojen tai toiminnallisuuksien merkitys sama. Järjestelmän pitäisi seurata yhtenäistä standardia. (Mt.)

Virhetilanteiden esto

Virhetilanteiden esto on mahdollista järjestelmän suunnittelun avulla, joka estää ongelman syntymästä. Virhetilanteet tulisi aina estää tai muuten antaa käyttäjälle mahdollisuus vahvistaa heidän tekonsa. (Mt.)

Tunnistettavuus ennen muistamista

Käyttäjän muistikuorma tulisi pitää minimissä jättämällä komponentit ja toiminnot esiin käyttöliittymässä. Näkymästä siirtyminen toiseen ei edellytä käyttäjän muistaa tietoja edellisestä näkymästä. Tarvittavat tiedot pitäisi olla näkyvillä tai helposti saatavilla sopivassa paikassa. (Mt.)

Joustavuus ja tehokkuus

Kokeneen käyttäjän vuorovaikutusta voidaan nopeuttaa järjestelmässä luomalla pikanäppäimiä tai sallimalla muita tapoja tuoda esiin useimmin käytetyt toiminnot (mt).

Esteettinen ja minimalistinen suunnittelu

Dialogien ei pitäisi sisältää tarpeetonta tietoa. Jokainen tietoa sisältävä komponentti pitäisi olla tarkoituksenmukainen, koska kaikki näytöllä oleva tieto kilpailee muiden näkyvillä olevien tietojen tärkeydestä. (Mt.)

Virheiden tunnistaminen ja niistä toipuminen

Virheistä pitäisi luoda helppolukuinen viesti, jossa selitetään mikä ongelma on ja ehdottaa ratkaisu ongelmaan (mt).

Opastus ja dokumentaatio

Järjestelmä on aina parempi, jos sitä voi käyttää ilman dokumentaatiota, mutta joskus se voi tulla tarpeen. Opastus ja dokumentaatio pitäisi olla helppo löytää ja niiden ohjeet keskittyä käyttäjän tehtävään sekä listata konkreettiset vaiheet tehtävän toteutukseen. (Mt.)

3.7 Material Design

Googlen luoma Material Design on visuaalinen muotokieli, joka on jatkuvasti päivitetty suunnitteluohje ja kokoelma työkaluja tehty käyttöliittymän suunnitteluun. Material Designin tarkoituksena on tuottaa hyviin suunnitteluperiaatteisiin pohjautuva, muokattava ja yhtenäinen käyttäjäkokemus jokaisella alustalla. Material Design muokaa fyysistä maailmaa mallintamalla valon ja varjojen piirteet käyttöliittymien visuaaliseen tyyliin. Muotokielen visuaalinen ilme perustuu printtimediaan. Värien ja tyhjän tilan käyttö on tarkoitus luoda hierarkia elementeille käyttöliittymässä. Käytettävyyden kannalta elementit ilmiselvästi erottuvat toisistaan. Muotokielessä hyödynnetään myös komponenttien liikettä, jolla luodaan merkityksellisiä siirtymiä käyttäjien johdosta. (Material Design n.d.)

3.7.1 Ympäristö

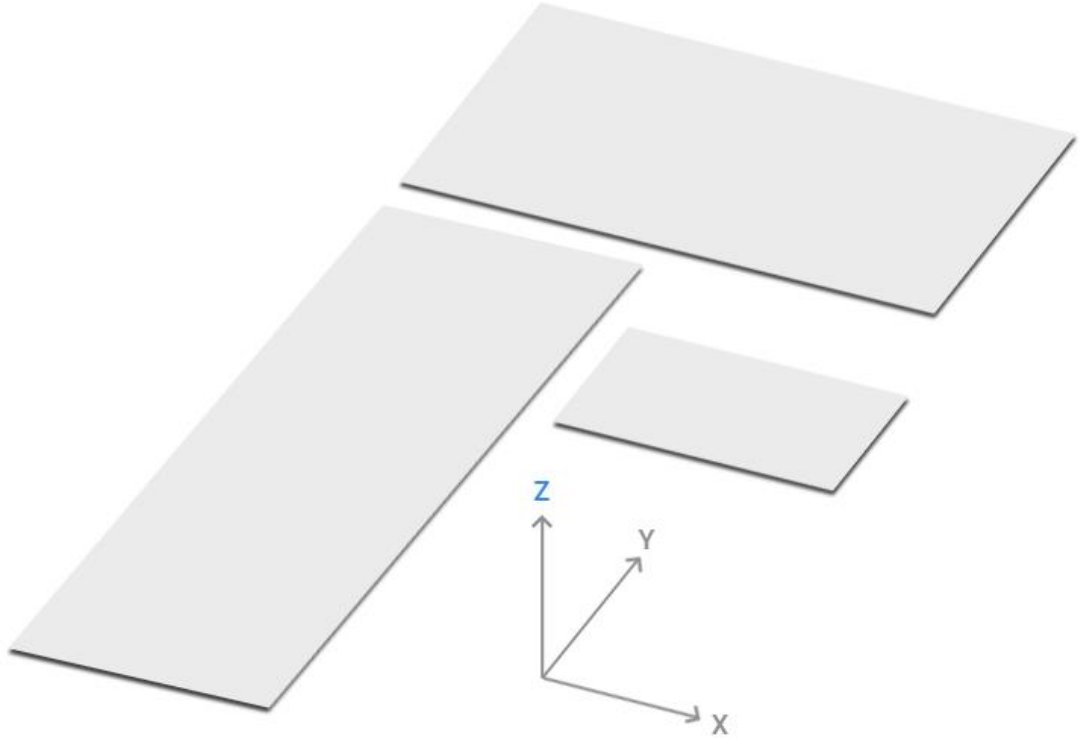
Material Designissa asemoidaan komponentteja kolmiulotteisessa ympäristössä. Näkymän valaisee virtuaaliset valot, jotka luovat materiaalille ympäröivät varjot. Varjot tuodaan esiin hyödyntämällä päällekkäisten elementtien korkeuseroa. Web-sivuilla z-akselia käytetään kerrostamiseen eikä perspektiivin luomiseen. 3D-maailmaa emuloidaan manipuloimalla y-akselia. (Material Design n.d.)

Materiaalin piirteet

Muotokielessä käytetään materiaalia kielikuvana, jonka kuvataan omaavaan seuraavia piirteitä:

- kiinteä
- varaa uniikin paikan tilassa
- läpitunkematon
- muuttuva muoto
- vaihtaa kokoaan vain tasollaan
- taipumaton
- voi liittää muuhun materiaaliin
- voi erottaa ja jakaa
- voidaan luoda tai tuhota
- liikkuu millä tahansa akselilla

Materiaalilla on muuttuvat x- ja y-mitat, mutta sen paksuus on absoluuttinen. Materiaalin paksuus vastaa yhtä pikseliä näytöllä, jonka pikselitiheys on 160 (ks. kuvio 1).



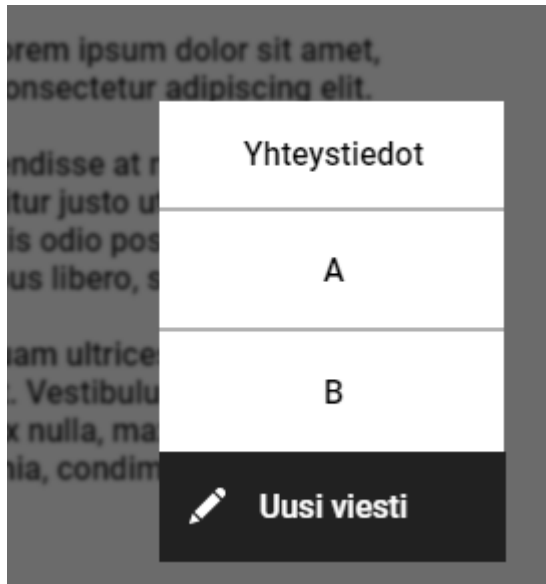
Kuvio 1. Materiaalin muuttuva korkeus ja leveys

Objektit Material Designissa jakaa samanlaisia piirteitä fyysisten objektien kanssa. Fyysisessä maailmassa objekteja voidaan pinota tai liittää yhteen, mutta niitä ei voi liikuttaa toisten läpi. Objektit myös luovat varjoja ja heijastavat valoa. Nämä säännöt ovat käyttäjälle ennestään tuttuja ja niitä voidaan käyttää yhtenäisesti eri sovelluksissa. (Mt.)

Scrim

Scrim on tapa muuttaa komponenttien visuaalista hierarkiaa tummentamalla tai lisäämällä läpinäkyvyyttä komponentteihin. Sen tarkoitus on ohjata käyttäjän huomio toiseen tärkeämpään komponenttiin näytöllä. (Mt.)

Tämä esimerkiksi voi olla valikko, jonka käyttäjä avaa painikkeesta, joka sillä hetkellä tummentaa muut komponentit näkymässä (ks kuvio 2).



Kuvio 2. Taustan tummentaminen

Korkeustaso

Korkeustaso on suhteellinen pituus z-akselilla kahden pinnan välillä. Korkeustason käyttö mahdollistaa elementtien liikuttamisen toisten edessä tai takana. Korkeustaso kertoo käyttäjälle miten päällekkäiset elementit käyttäytyvät sovelluksessa ja se auttaa ohjaamaan käyttäjän huomion korkeimpaan elementtiin visuaalisessa hierarkiassa. (Mt.)

Visuaalinen hierarkia on tärkeä osa suunnittelua, koska sillä voidaan varmistaa, että käyttäjät näkevät ja ymmärtävät mahdollisimman suuren osan sivun sisällöstä. Kaikki visuaaliset vihjeet kuvaavat selkeästi näkymän komponenttien välisiä suhteita. (Krug 2006, 5.)

Korkeustaso voidaan näyttää varjoilla tai muilla visuaalisilla vihjeillä kuten täyttöväreillä tai läpinäkyvillä pinnoilla. (ks kuvio 3).



Kuvio 3. Korkeustason ilmaiseminen varjoilla

Elementtien oletuskorkeudet

Jokaisella elementillä on oletus korkeustaso riippuen komponentin tyypistä. Esimerkiksi dialogi -elementillä on sama korkeustaso kuin muilla dialogeilla. Komponentit muuttavat omaa korkeustasoja reagoimalla tapahtumaan järjestelmässä tai käyttäjän kosketukseen. Oletuskorkeus vaihtelee ympäristöstä, alustasta ja sovelluksesta riippuen. Korkeustason tarkoitus mobiilinäytöillä on näyttää visuaalisia vihjeitä osoittaakseen komponentin muuttumista interaktiiviseksi. Työpöydällä oletuskorkeudet ovat matalampia, koska työpöydällä voidaan hyödyntää muita visuaalisia vihjeitä, kuten ulkoviivoja, jotka aktivoituvat hiiren osoittimen liikkeestä. (Material Design n.d.)

Reunat auttavat havainnollistamaan materiaalin taktiilisen laadun. Ne erottavat käyttöliittymän elementit tunnistettaviin komponentteihin. Esimerkiksi yläpalkin reunat kertovat käyttäjälle, että se ei kuulu sisältöön ja sitä voi vierittää itsenäisesti. Oletuksena Material Designin komponentit käyttävät varjoja näyttääkseen reunat. Vaihtoehtoisesti pinnoille voi myös antaa eri värin tai läpinäkyvyyden, mutta reunojen on luotava tarpeeksi suuri kontrasti muita pintoja kohtaan, jotta ne voidaan selvästi erottaa toisistaan. (Mt.)

Varjot

Varjot voi ilmaista korkeuseron pintojen välillä tavalla jolla muut tekniikat eivät voi. Pituutta kahden pinnan välillä voidaan muokata varjon koon ja diffuusion mukaan. Mitä tarkempi ja pienempi varjo on, sitä lähempänä pinta on. Kauempana olevat pinnat luovat suuremman ja pehmeämmän näköisen varjon. (Mt.)

Varjot tuovat näkyviin päällekkäisten pintojen reunat ja sen korkeustason. Varjoja on käytettävä johdonmukaisesti, koska se auttaa käyttäjän ymmärtämään, että kyseessä on korkeustason muutos, kun komponentin varjo kasvaa sen liikuessa (ks. kuvio 4).



Kuvio 4. Varjon intensiivisyys

3.7.2 Rakenne

Material Designin asettelulla (engl. layout) suositellaan yhdenmukaisen rakenteen suunnittelua eri alustoilla ja ympäristöissä sekä eri kokoisilla näytöillä hyödyntämällä yhdenmukaisia elementtejä ja välejä (Material Design n.d).

Pikselitiheys

Näytön pikselitiheys ja resoluutio vaihtelevat alustasta riippuen. Laitte riippumattomat pikselit ja skaalautuvat pikselit ovat yksikköjä, jotka tarjoavat joustavan tavan sovittaa yhdenmukaisen käyttöliittymän monelle alustalle. Pikselitiheys tarkoittaa kuinka monta pikseliä mahtuu yhdelle tuumalle. Mitä korkeampi pikselitiheys sitä

suurempi määrä pikseleitä tuumaa kohti. Tästä syystä käyttöliittymän elementit näyttävät suuremmilta matalalla pikselitiheydellä ja pienemmiltä korkealla pikselitiheydellä. (Mt.)

Tiheys riippumattomat pikselit

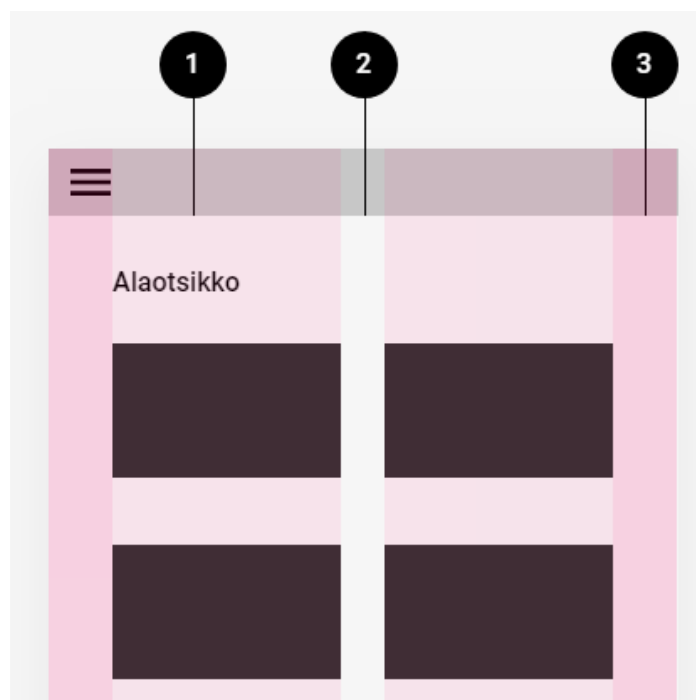
Tiheys riippumattomuus tarkoittaa yhdenmukaisten elementtien esittämistä eri tiheyksillä. Tiheys riippumattomat pikselit (dp) ovat joustavia yksiköitä, jotka skaalautuvat luomaan yhdenmukaiset mitat eri kokoisille näytöille. Material Design käyttää tiheys riippumattomia pikseleitä näyttääkseen elementtejä yhdenmukaisesti erilaisilla näytöillä. (Mt.)

Responsiivinen asettelu

Material Designin responsiivinen asettelu mukautuu näytön koon ja orientaation mukaan, jotta sommittelu pysyy yhdenmukaisena eri kokoonpanoissa. (mt).

Material Designin asettelu jakautuu kolmeen elementtiin (ks. kuvio 5):

1. Palstat
2. Urat
3. Marginaalit



Kuvio 5. Palstat, urat ja marginaalit

Palstat

Sisältö laitetaan palstojen sisälle näytöllä. Palstan leveys määritellään prosentteina kiinteän yksiköiden sijaan, jotta sisältö voi mukautua näytön koon mukaan. Palstojen määrä riippuu siitä mihin pysäytyspisteeseen (engl. breakpoint) näytön koko sijoittuu. (Mt.)

Urat

Urat ovat aukkoja palstojen välissä. Ne auttavat erottamaan sisältöä. Urien leveydet ovat kiinteitä arvoja jokaisella pysäytyspisteellä. Pienemmillä näytöillä urilla on vastaavasti pienempi leveys. Leveämmät urat sopivat paremmin suuremmille näytöille luodakseen tilaa palstojen välille. (Mt.)

Marginaalit

Marginaali tarkoittaa tilaa sisällön ja näytön vasemman ja oikean reunan välillä. Marginaalien leveydet ovat kiinteitä arvoja jokaisella pysäytyspisteellä. Marginaalin leveys vaihtelee näytön koon mukaan. Leveämmät marginaalit luovat tilaa sisällön ympärille isoimmilla näytöillä. Ihanteellinen luettavuus saavutetaan käyttämällä 40-60 kirjainta riviä kohden. (Mt.)

Pysäytyspisteet

Pysäytyspiste on joukko ennalta määritettyjä näytön leveyksiä, jossa asettelu vaihtuu sovittamaan sisällön ruutuun paremmin. Jokainen pysäytyspiste määrää, kuinka monta palstaa, uraa ja marginaalia käytetään. (Mt.)

UI-alueet

Tässä työssä käyttöliittymään (engl. user interface) viitataan lyhenteellä UI.

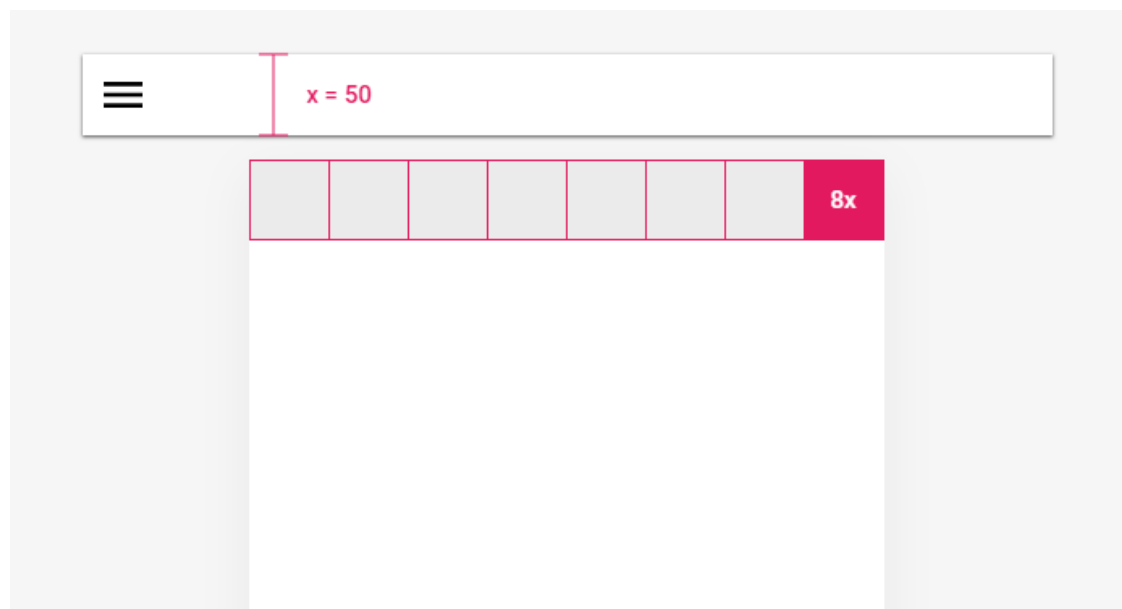
Käyttöliittymän asettelu muodostuu useammasta UI-alueesta, kuten sivupalkista, sisällöstä ja yläpalkista. Nämä alueet pitäisi näyttää yhdenmukaisilta eri laitteilla samalla sopeutuen eri kokoisten näyttöjen pysäytyspisteisiin. UI-elementit työpöydällä pitäisi järjestää niin, että ne ovat yhdenmukaisia mobiililaitteiden käyttöliittymissä.

Jotkut alueet voivat olla liikkumattomia ja toiset väliaikaisia, kuten sivupalkki mobiililaitteella, jonka käyttäjä voi tuoda esille. Työpöydällä on tarpeeksi tilaa näyttää sivupalkki ja sisältö vierekkäin. (Mt.)

Inkrementit

Inkrementti on tapa mitata elementtien koko ja asemointi sovelluksessa. Sillä on sama korkeus ja leveys, mutta niitä voi olla useampi vierekkäin. (Mt.)

On suositeltua käyttää jonkin sovelluksen perus elementin korkeutta, kuten yläpalkkia inkrementtinä (ks. kuvio 6).



Kuvio 6. Inkrementti

3.7.3 Navigaatio

Navigaatio antaa käyttäjän liikkua sovelluksen läpi. Navigaation voi toteuttaa usealla eri keinolla. Nämä voi olla navigointikomponentteja tai sisältökomponentteja, joihin on upotettu navigointi affordanssina. (Material Design n.d.)

Navigoinnin tavat

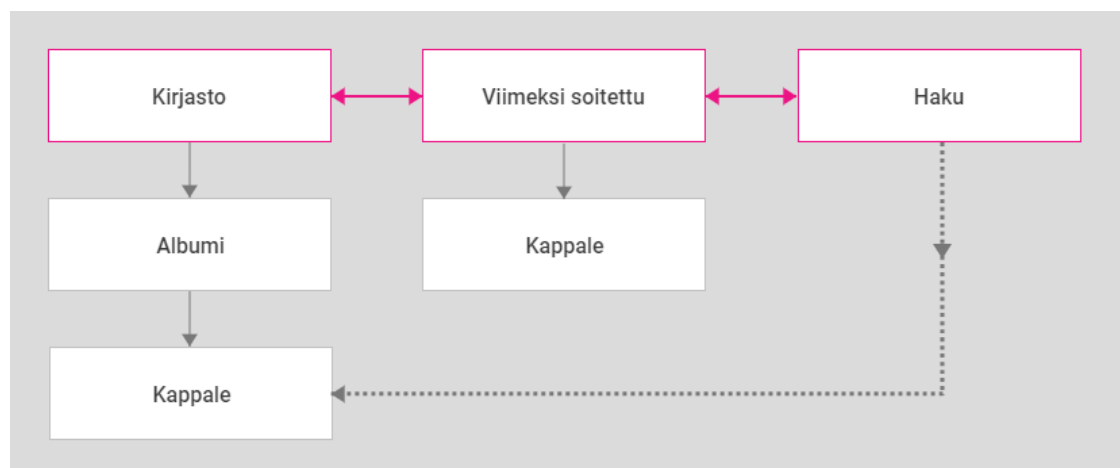
Riippuen sovelluksen tietoarkkitehtuurista, käyttäjä voi liikkua kolmeen suuntaan navigoinnilla:

Sivuttainen navigaatio tarkoittaa näkymän vaihtamista toiseen samalla tasolla hierarkiassa. Sovelluksen päänavigointi pitäisi antaa käyttäjän liikkua kaikkiin osiin hierarkian ylimmällä tasolla. (Mt.)

Eteenpäin navigointi tarkoittaa hierarkiassa siirtymistä sitä seuraavalle tasolle. Hierarkiassa voi eteenpäin siirtyä komponenteilla, jossa on upotettu navigointi (kortit, listat tai kuvat), painikkeilla, linkeillä tai hakutoiminnolla. (Mt.)

Taaksepäin navigointi tarkoittaa hierarkiassa nousua. Tämän navigaatio tavan käyttäytymisen määrää alustan konventiot sovelluksessa. (Mt.)

Esimerkkinä tässä musiikkisovelluksessa käyttäjä voi etsiä kappaleen joko seuraamalla hierarkiaa tai sitten ohittaa muut näkymät käyttämällä haku (ks. kuvio 7).



Kuvio 7. Hierarkia navigoinnissa

Kotisivun rooli navigoinnissa

Kotisivulla on tärkeä rooli hierarkian opittavuudessa. Hierarkian ylimpänä on aina kotisivu. Kotisivulta pitää tulla käyttäjälle selväksi sivun tunnus ja mitä sillä tehdään. Sivun hierarkia pitäisi olla myös nähtävissä, jotta käyttäjä tietää mitä missäkin voidaan tehdä ja miten sivut on järjestetty. Pysyvä navigaatio on hyvä tapa muistuttaa käyttäjää sovelluksen hierarkiasta. (Krug 2006, 95.)

3.7.4 Väri ja typografia

Väriin ja typografian käyttö auttaa ilmaisemaan sovelluksen brändiä sekä visuaalista hierarkiaa. Material Designissa on järjestelmällinen tapa soveltaa värejä eri UI-elementteihin. Tyypillisesti edustettavaan brändiin valitaan kaksi väriä. Näistä käytetään eri sävyjä elementin tyyppin perusteella. (Material Design n.d.)

Ensisijainen väri

Ensisijainen väri on useiten käytetty väri sovelluksen näkymissä ja komponenteissa. Toista väriä ei tarvitse välttämättä käyttää, jos muita elementtejä täyttää ensisijaisen värin eri sävyillä. (Mt.)

Toissijainen väri

Toissijainen väri antaa mahdollisuuden erottaa sovelluksen muista uniikeilla yhdistelmillä. Sitä pääosin käytetään aksenttina pienemmissä komponenteissa. Toissijaiset värit parhaiten käyvät painikkeisiin, lomakekomponentteihin, edistymispalkkeihin, linkkeihin ja otsikkoihin. (Mt.)

Tässä esimerkissä käytetään ensisijaista väriä, sen tummempaa sävyä sekä toissijaista väriä (ks. kuvio 8).



Kuvio 8. Värien käyttö eri elementeissä

WCAG standardit tekstin luettavuudelle

Kaiken tekstin pitäisi olla luettavissa ja vastata esteettömyyden standardeja. WCAG (Web Content Accessibility Guidelines) -standardissa on määritetty minimi kontrastisuhte tekstille. Tämän standardin minimi kontrastisuhte on 4.5:1 tekstille sivuilla ja kuvissa. Isokokoiselle tekstille (vähintään 18 tai 14 pistekoon lihavointi) minimi kontrastisuhte on 3:1. (WCAG 2.0, 1.4.)

Tekstin korostaminen läpinäkyvyydellä

Tekstin väriä voi muuttaa sen tarkoituksen mukaan sovelluksessa. Tekstin luettavuuden voi kuitenkin säilyttää paremmin muokkaamalla tekstin läpinäkyvyyttä sen värjäämisen sijaan. (Material Design n.d.)

Tumma teksti (värikoodi #000000) vaalealla taustalla (#FFFFFF) käyttää seuraavia läpinäkyvyyden tasoja:

Otsikot käyttävät 100% läpinäkyvyyttä. Tekstikappaleet ja ohjetekstit käyttävät 60% läpinäkyvyyttä. Poistetut valinnat esimerkiksi lomakkeissa käyttävät 38% läpinäkyvyyttä (ks. kuvio 9).

Otsikot	
100%	#000000
Ohjetekstit	
60%	#000000
Poistettu valinta	
38%	#000000

Kuvio 9. Tekstin korostaminen läpinäkyvyydellä

Typografia

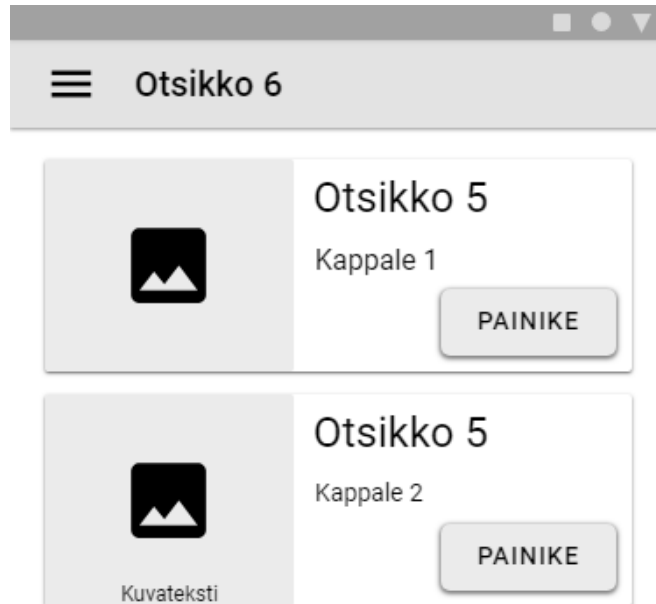
Typografialla näytetään sisältö selkeimmällä ja tehokkaimmalla tavalla. Material Design käyttää omaa kirjasinskaalaa, jonka kategorioille on luotu oma käyttötarkoitus ja tyyli. (Mt.)

Kirjasinskaalaan kuuluu 13 kategoriaa (ks. kuvio 10).

Skaalan kategoria	Kirjasintyyppi	Fontti	Koko	Aakkoslaji	Kirjainväli
H1	Roboto	Light	96	Lause	-1,5
H2	Roboto	Light	60	Lause	-0,5
H3	Roboto	Regular	48	Lause	0
H4	Roboto	Regular	34	Lause	0,25
H5	Roboto	Regular	24	Lause	0
H6	Roboto	Medium	20	Lause	0,15
Alaotsikko 1	Roboto	Regular	16	Lause	0,15
Alaotsikko 2	Roboto	Medium	14	Lause	0,1
Kappale 1	Roboto	Regular	16	Lause	0,5
Kappale 2	Roboto	Regular	14	Lause	0,25
PAINIKE	Roboto	Medium	14	Suuraakkoset	0,75
Kuvateksti	Roboto	Regular	12	Lause	0,4
YLIVIIVA	Roboto	Regular	10	Suuraakkoset	1,5

Kuvio 10. Material Designin kirjasinskaala

Kirjasinskaala esiintyy tekstinä eri UI-elementeissä. Jokaista kategorian attribuuttia voi muokata sovelluksen tyylin mukaan (ks. kuvio 11).



Kuvio 11. Kirjasinskaalan käyttö

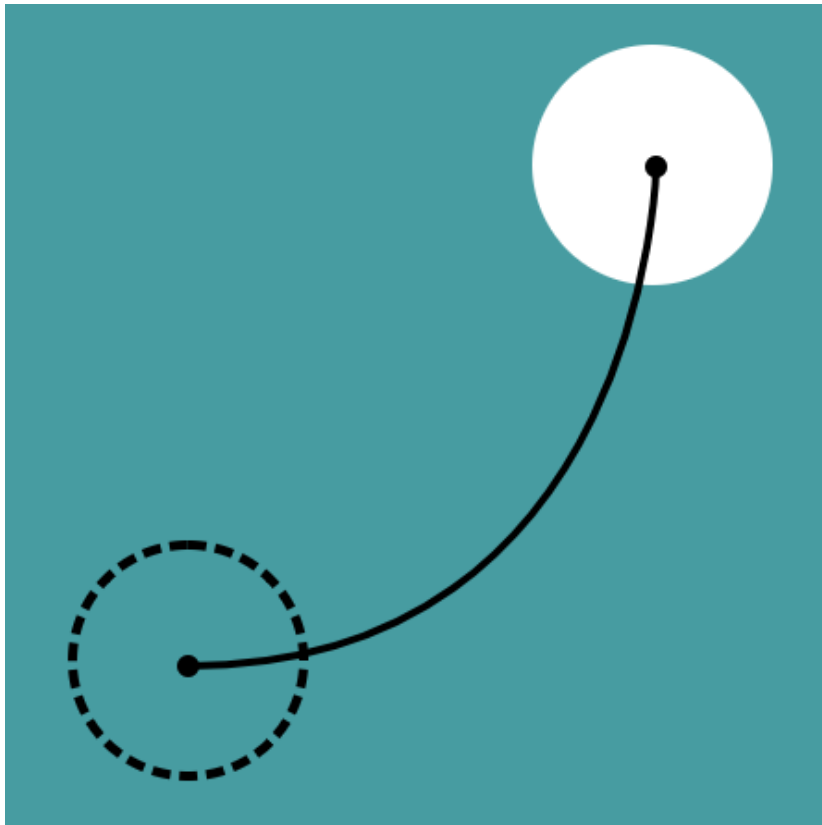
3.7.5 Liike ja transformointi

Liikkeen Material Designissa on tarkoitus kuvata komponenttien suhdetta tilassa sekä niiden funktiota sulavilla animaatioilla. Liikkeellä havainnollistetaan, miten sovellus on organisoitu ja mitä toimintoja on käytettävissä. Sillä voidaan kohdistaa käyttäjän katse uuteen näkymään tai vihjata mitä tapahtuu, jos käyttäjä suorittaa eleen. Käyttäjän huomio voidaan kanssa viedä pois hetkeksi aikaa taustalla latautuvasta näkymästä. (Material Design n.d.)

Liikkuva materiaali on responsiivinen ja se reagoi tarkalleen liikuttamalla komponenttia siitä mistä liike aloitetaan. Fyysisessä maailmassa objektin kiihdytystä rajoittaa sen paino ja kitka. Material Designin komponentit käyttäytyvät samalla tavalla. Komponenttien liike luo yhtenäisen kokemuksen sovelluksessa ohjaamalla käyttäjän katsetta ja se lisää tuotteen esteettisyyteen yksinkertaisilla siirtymillä. (Mt.)

Komponenttien liikkeessa tai tilan vaihtuessa liikkeen on oltava tarpeeksi hidas, jotta käyttäjä ymmärtää mitä siirtymässä tapahtuu, mutta animaation ei pitäisi hidastaa työnkulkua. Siirtymien pitäisi olla lyhyitä, sillä käyttäjät näkevät niitä usein. Animaatioiden kestoa on suositeltavaa vaihdella, koska liikkeen pituus vaihtelee eri näkymissä. (Mt.)

Liike kahden pisteen välillä seuraa koveraa kaarta. Nostamalla jonkin objektin fyysisessä maailmassa vaatii ponnistuksen. Ylöspäin liikkuvan komponentin pitäisi aloittaa liike samaan tyyliin hitaalla, matalalla nousulla, joka jyrkistyy loppua kohti. Alaspäin liikkuvan komponentin tapauksessa seurataan samaa sääntöä, mutta päinvastaisessa järjestyksessä, koska fyysisessä maailmassa painovoima nopeuttaisi liikkeen alkua (ks. kuvio 12).



Kuvio 12 Liikkeen kaari näytöllä

Yhdellä akselilla liikkuvat komponentit eivät seuraa kaarta. Nämä liikkeet ovat helpompia hahmottaa ja siitä syystä voivat liikkua hieman nopeammin. Komponentit,

jotka tulevat tai poistuvat näytöltä liikkuvat yhdellä akselilla. Kun näytölle tuodaan elementtejä, on annettava käyttäjälle helposti seurattava suora liike, joka hidastuu loppua kohti. Näytöltä poistuvat elementit eivät vaadi käyttäjän huomiota, jonka takia näissä elementeissä animaatiot ovat nopeampia. (Mt.)

Siirtymät

Siirtymän tyyppin perusteella käyttäjä näkee, mihin suuntaan sovelluksen hierarkiassa liikutaan. Siirtymän aikana UI-elementit, jotka transformoidaan, luokitellaan meneviksi, tuleviksi tai pysyviksi elementeiksi. Elementin kategoria vaikuttaa transformoinnin tapaan. (Mt.)

Pysyvän elementin transformointi alkaa ja loppuu näytöllä. Nämä elementit voi olla esimerkiksi navigaatiokuvakkeita. Menevä elementti, kuten otsikko poistuu näytöltä häivytyksellä. Tulevan elementin transformointi tuodaan näytölle häivytyksellä. (Mt.)

Häivytytys tarkoittaa elementin läpinäkyvyyden hidasta muutosta. Sillä saadaan aikaan sulavan näköinen siirtymä paljastamalla uusi elementti häivytetyn elementin takaa. (Mt.)

4 Käyttöliittymän suunnittelu

Tässä luvussa käydään läpi käyttöliittymän suunnittelun prosessi suunnittelusta toteutukseen ja testaukseen.

4.1 Vaatimusmäärittely

Game of Skills -pelin suunnittelussa ideoitii aluksi toiminnalliset vaatimukset, jotka pitäisi sisältää ensimmäiseen prototyyppiin. Sen jälkeen luotiin käyttötarina pelin käytöstä. Pelin tarkoituksena on kannustaa pelaajia suorittamaan muutamasta harjoitusliikkeestä muodostettuja haastetehtäviä. Pelissä seurataan pelaajan sekä ryhmän kehitystä viikon pituisissa jaksoissa. Haastetehtävän suorittamalla pelaaja edistää ryhmän kehitystä viikkotavoitetta kohti.

Käytettävyyksvaatimukset on sovelluksen suunnittelun ensimmäinen vaihe.

Suunnittelijana on tärkeää tietää, mikä on käyttäjän tavoite, missä tuotetta

käytetään, mitä käyttäjät ovat varsinaisesti tekemässä käyttäessään tuotetta ja mitä

vaatimuksia näistä seuraa tuotteen käytettävyydelle. (Sinkkonen, Kuoppala, Parkkinen & Vastamäki 2006, 15.)

Käyttötarina pelille:

Pelaaja kirjautuu sisään. Uuden käyttäjän tapauksessa pelaaja luo tilin ja liittyy ryhmään. Pelaaja sitten valitsee ensimmäisen haastetehtävän, jonka suorituksen yksityiskohdista pelaaja näkee tiedot ennen tehtävän aloittamista. Pelaaja päättää milloin aloittaa liikkeen suorituksen, koska suorituksilla on aikarajoite. Haastetehtävän suoritettua pelaaja menee takaisin etusivulle, josta näkee yhteenvedon suorituksista.

Käyttötarinasta luotiin rakenne sovelluksen eri sivuille. Tästä pystyi viemään kaikki ideat rautalankamalleihin, jossa käyttöliittymää voisi testata ja iteroida.

4.2 Rautalankamallit

Käyttöliittymän iterointi aloitettiin rautalankamalleista. Rautalankamalleja käytettiin sisällön priorisoitiin ja asemointiin. Rautalankamallit tehtiin Wireframe.cc -sivustolla. Wireframe on verkkosovellus, jolla suunnitellaan rautalankamalleja usealle alustalle.

Rautalankamallista luotiin käyttöliittymä, joka mukaili käyttötarinan toiminnallisuuksia. (ks kuvio 13).



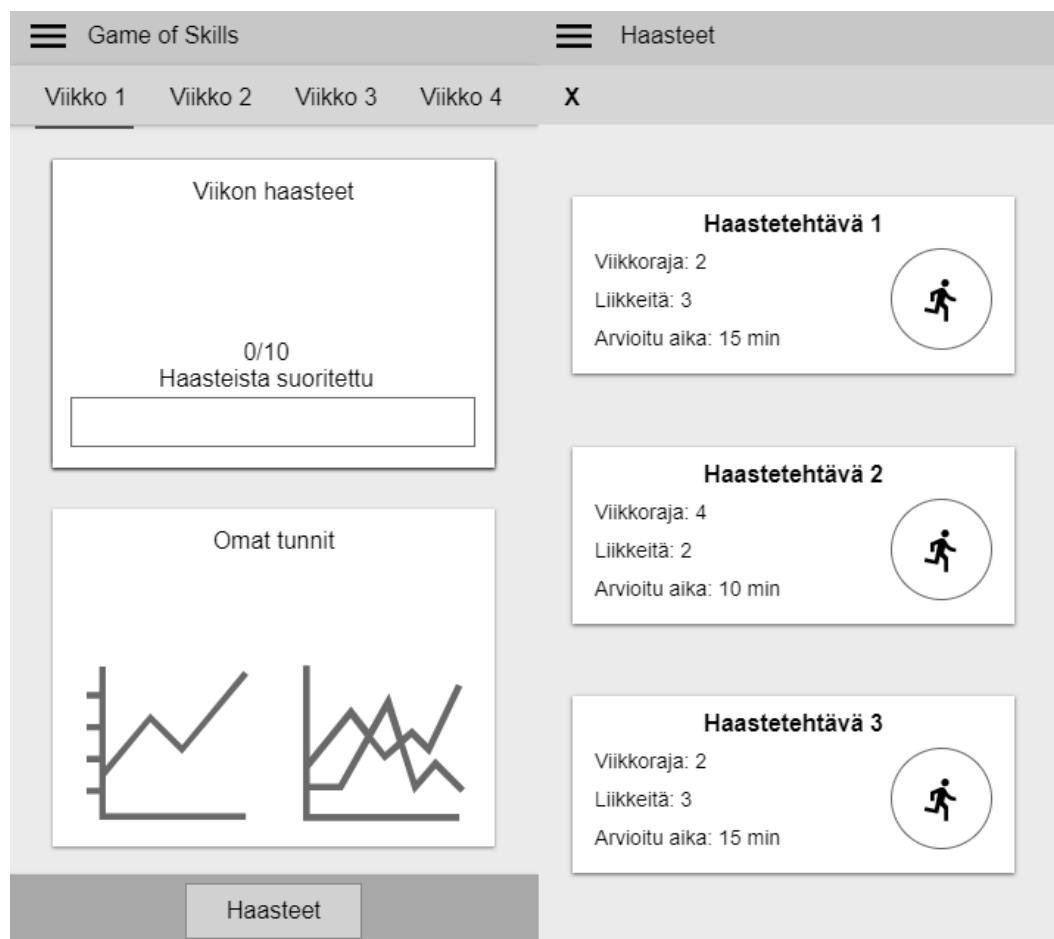
Kuvio 13 Rautalankamalli etusivusta ja tehtäväpaketin valinnasta

Käyttöliittymän ensimmäiseen versioon muutettiin aluksi pisteiden tyypit prosenteista pelkäsi liikkeiden suorituksen määräksi selkeyden vuoksi. Tehtäväpaketit uudelleennimettiin haastetehtäviksi. Ensimmäisten rautalankamallien jälkeen siirryttiin UXPin -sovellukseen, joka mahdollisti sovelluksen testauksen interaktiivisessa prototyypissä. UXPin on verkkosovellus, jolla suunnitellaan ja toteutetaan käyttöliittymän prototyyppejä varhaisimmista suunnittelun vaiheista valmiiseen ulkoasuun ja rakenteeseen asti.

4.3 Prototypisointi

Tämän työn prototyyppien arviointiin osallistui 4 henkilöä. Käyttöliittymän rakennetta ja toiminnallisuuksia muokattiin heuristiikkojen avulla. Prototyypeistä syntyi 3 iteraatiota. Ensimmäinen versio arvioitiin 24.7.2018. Arvioinnissa huomioitiin, että käyttäjän ei tarvitse nähdä tulevia viikkoja ja sen sijaan on hyödyllisempää nähdä nykyhetkinen tilanne sekä menneet viikot.

Taaksepäin navigointi -painike todettiin helpommaksi tunnistaa taaksepäin-nuolena rastin sijaan. Näin painike vastaa nykyistä nettiselainten konventiota sivujen eteen- tai taaksepäin selauksessa (ks. kuvio 14).



Kuvio 14. Ensimmäinen prototyyppi

Toinen iteraatio arvioitiin 8.8.2018. Tässä iteraatiossa otettiin käyttöön animaatiot näkymien siirtymille, joiden funktiona on ohjata käyttäjän katse oikeaan paikkaan

siirtymän alusta seuraavalle näkymälle. Käytettävyyden kannalta painikkeille luotiin varjot, jotta käyttäjä tunnistaisi ne korkeammalla visuaalisessa hierarkiassa (ks. kuvio 15).



Kuvio 15 Painikkeen varjo

Kirjautumissivulle luotiin kaksi lomaketta rekisteröitymiselle sekä kirjautumiselle. Käyttäjä valitsee jommankumman lomakkeen painikkeella. Tämä säästää tilaa mobiilinäytöllä sekä vähentää käyttäjän kognitiivista kuormaa (ks. kuvio 16).

Game of Skills



UUSI KÄYTTÄJÄ KIRJAUDU SISÄÄN

Luo uusi käyttäjä

Käyttäjänimi:

Ryhmäkoodi:

Salasana:

LUO KÄYTTÄJÄ

Kuvio 16 Kirjautumissivun lomakkeet

Kolmas iteraatio arvioitiin 29.8.2018. Etusivun rakennetta muutettiin tässä iteraatiossa, jotta viikoittaisten liikkeiden tyypeistä voisi näyttää tietoa. Tässä vaiheessa prototyypin lisättiin myös väriteema.

Väriteema sovelluksessa perustui olemassa olevaan Game of Skills -verkkokauppaan. Musta toimii ensisijaisena värinä ja oranssi toissijaisena värinä. Molemmista väreistä käytetään eri sävyjä käyttöliittymän komponentin tyyppin mukaan (ks. kuvio 17).



Kuvio 17 Väriteema

Sovelluksen fontiksi valittiin Roboto. Tekstien paikkaa visuaalisessa hierarkiassa säädellään tiedon tärkeyden mukaan. WCAG -standardin mukaan, minimiläpinäkyvyys mustalla taustalla on 49%. Keltaisella taustalla korkeimman kontrastin luo musta teksti, jonka minimiläpinäkyvyys on 67% luettavalle tekstille.

Näin alhaisia lukuja ei tarvinnut käyttää tekstin läpinäkyvyyden kanssa, mutta sillä painotettiin tiettyjä tietoja ja painikkeiden tekstejä. Kaikki interaktiiviset elementit ja otsikot sovelluksessa saavat korkeimman läpinäkyvyyden tason. Tekstikappaleet ja aliotsikot käyttävät 90% läpinäkyvyyttä. Poikkeuksena ovat muut tiedot, kuten

inaktiiviset välilehdet, joissa on alhaisempi läpinäkyvyys aktiiviseen välilehteen verrattuna (ks. kuvio 18).



Kuvio 18 Tekstin painotus

4.4 Käyttäjätestaus

Käyttäjätestit toteutettiin 6 eri käyttäjällä. Testattavat henkilöt olivat kaikenikäisiä urheilijoita tai urheilun harrastajia. Kaikille käyttäjille annettiin UXPin -sivun linkki, jonka kautta he pääsivät testaamaan prototyyppiä omilla puhelimillaan. Testit toteutettiin henkilökohtaisesti käyttäjien kanssa, joista kirjoitettiin ylös käyttäjän kommentit ja onnistumiset tehtävien kanssa.

Käyttäjätesti aloitettiin kertomalla käyttäjälle, mistä testissä on kyse ja mihin tarkoitukseen sovellusta käytetään. Käyttäjiä sitten pyydettiin suorittamaan tehtäviä prototyyppissä. Jokaisen tehtävän lopussa käyttäjältä kysyttiin, oliko annetun tehtävän toteutuksessa vaikeuksia ja mikä mahdollisesti johti käyttäjän virheeseen. Testin aikana käyttäjän annettiin myös esittää huomioita tai ehdotuksia sovelluksen käytöstä. Testiaineiston pohjalta toteutettiin vielä yksi iteraatio prototyyppistä.

5 Tulokset

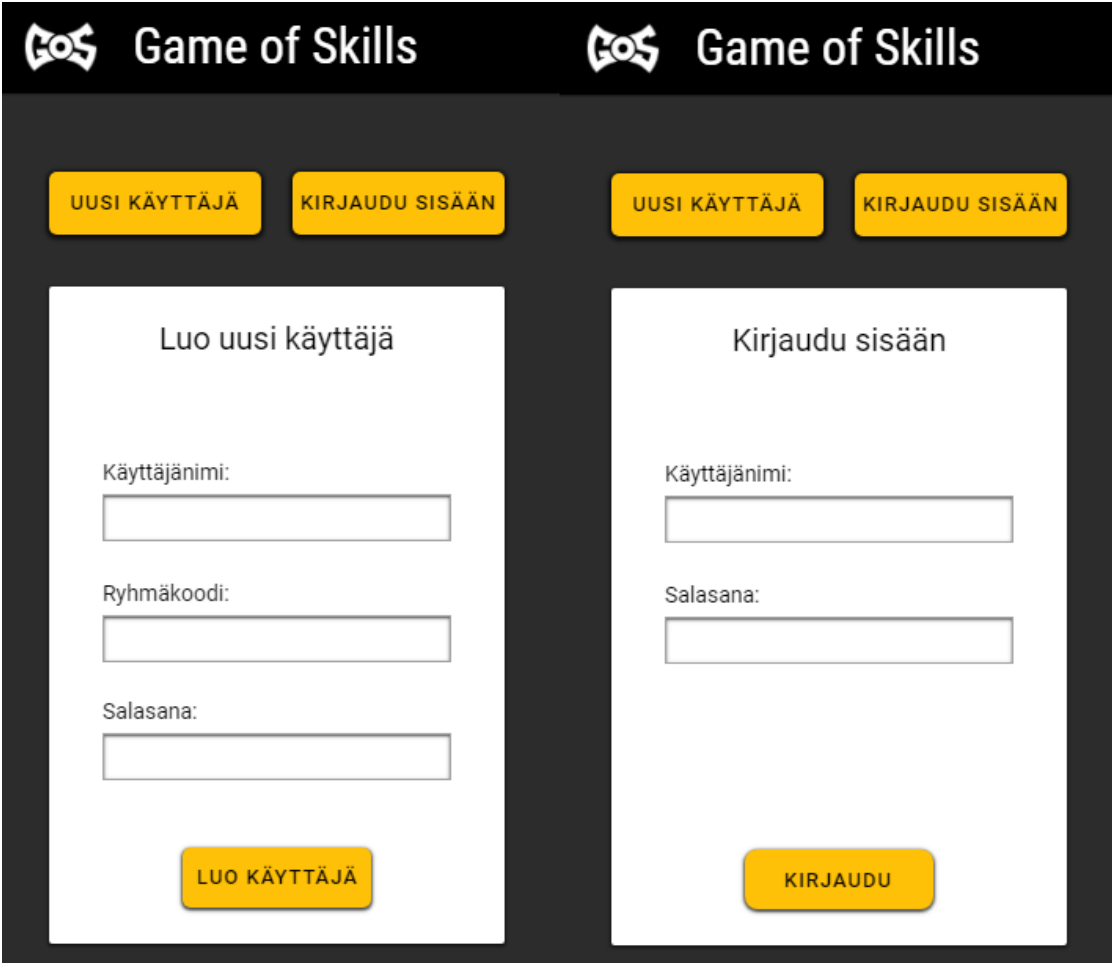
Tässä luvussa esitellään tutkimuksen tulokset. Luvussa näytetään myös lopullinen versio sovelluksen prototyypistä sekä käyttäjätestauksen tulokset.

5.1 Lopullinen käyttöliittymä

Tutkimuksen tuloksena syntyi käyttöliittymä sovelluksen mobiiliversiolle.

Käyttäjätestauksen tuloksien ymmärtämiseksi käydään läpi tässä vaiheessa käyttöliittymän toiminnallisuudet.

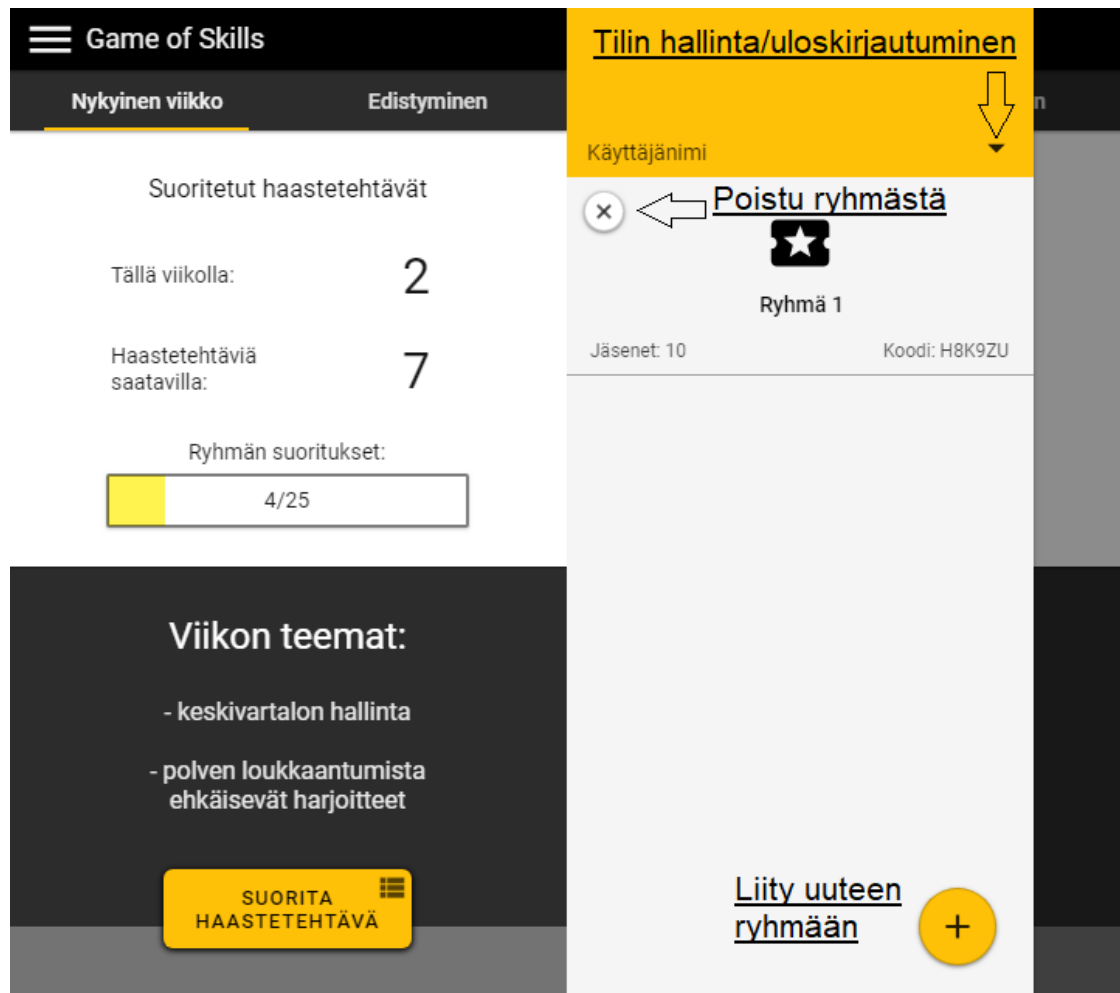
Kirjautumissivu on sovelluksen ensimmäinen sivu. Tällä sivulla käyttäjä valitsee, kirjautuuko sisään vai luodaanko uusi käyttäjä (ks. kuvio 19).



The image shows two side-by-side screenshots of the 'Game of Skills' mobile application interface. Both screenshots feature a dark grey header with the 'GOS Game of Skills' logo and title. Below the header, there are two yellow buttons: 'UUSI KÄYTTÄJÄ' and 'KIRJAUDU SISÄÄN'. The left screenshot shows the 'Luo uusi käyttäjä' (Create new user) form, which includes three input fields: 'Käyttäjänimi:' (Username), 'Ryhmäkoodi:' (Group code), and 'Salasana:' (Password), followed by a yellow 'LUO KÄYTTÄJÄ' button. The right screenshot shows the 'Kirjaudu sisään' (Login) form, which includes two input fields: 'Käyttäjänimi:' (Username) and 'Salasana:' (Password), followed by a yellow 'KIRJAUDU' button.

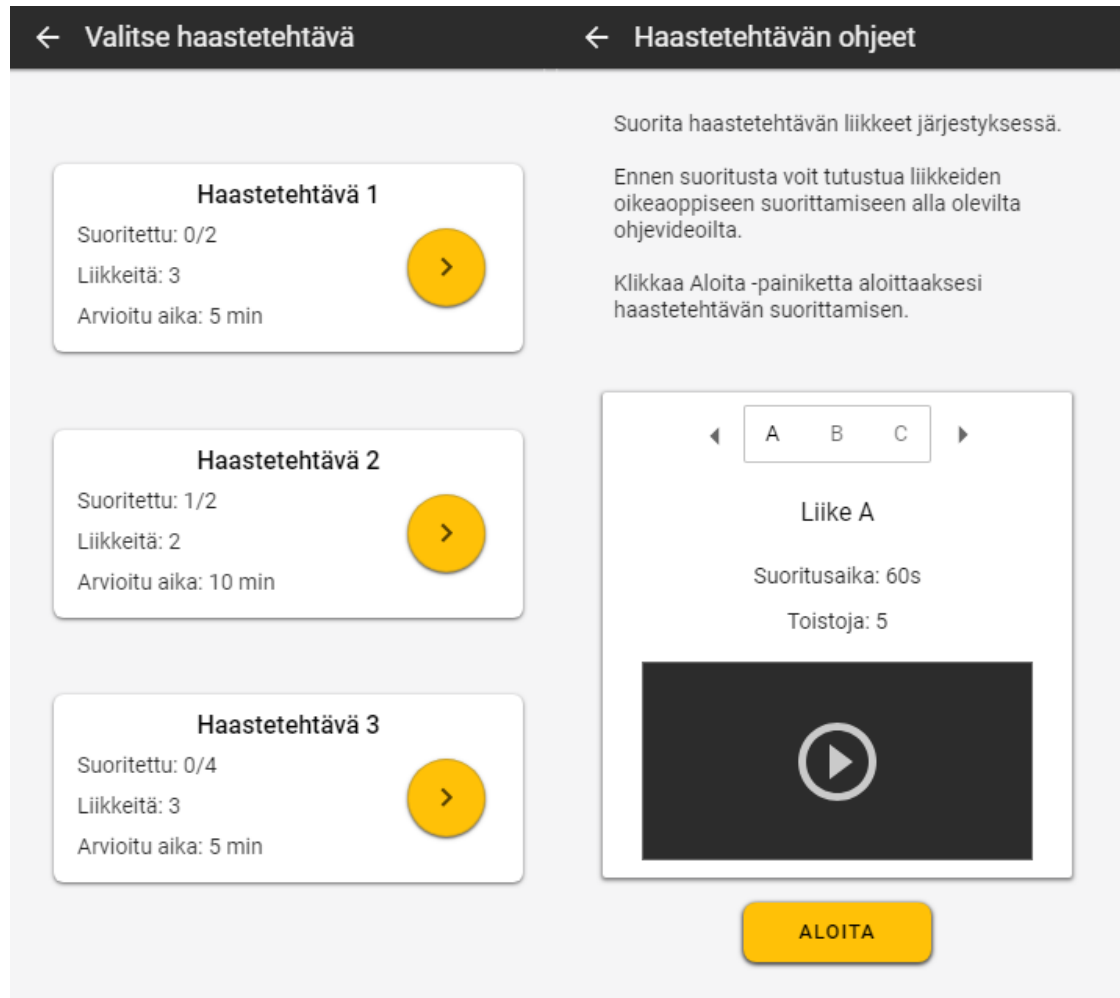
Kuvio 19 Kirjautumissivu

Etusivulla näkee yhteenvedon suoritetuista haasteista sekä minkä tyyppisiä haasteita nykyisellä viikolla on. Tilin ja ryhmän hallinta löytyy vasemman yläkulman painikkeen takaa. Kuviossa on myös alleviivattu kommentteilla ryhmän ja tilin hallinnan toiminnot (ks. kuvio 20).



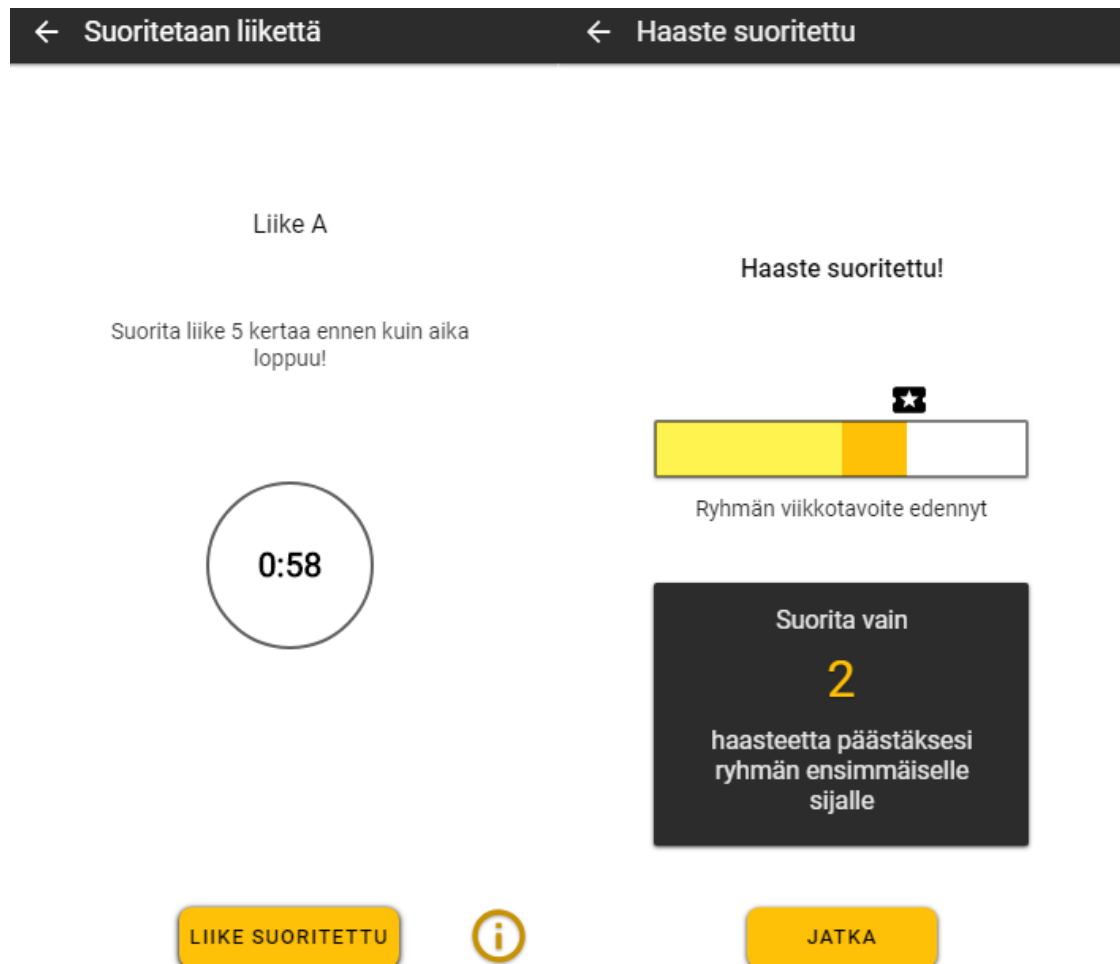
Kuvio 20 Etusivu ja ryhmän sekä tilin hallinta

Sovelluksen painikkeissa käytetään pyöreitä reunoja. Tämä visuaalinen sääntö tehdään selväksi kirjautumissivun ensimmäisistä painikkeista, jotta käyttäjä helpommin tunnistaa haastetehtävien listan kortit interaktiivisina elementteinä. Haastetehtävän valinnut käyttäjä voi selata liikkeiden läpi ja katsoa ohjevideot liikkeiden oikeaoppisesta suorituksesta (ks. kuvio 21).



Kuvio 21 Haastetehtävät ja haasteen ohjeet

Liikkeen suorituksen aikana keskellä näyttöä näytetään jäljellä oleva aika ja kuvake, jolla saa liikkeen ohjevideon esille. Suorituksen jälkeen visualisoidaan ryhmän edistymisen sekä kannustetaan pelaajaa jatkamaan toiseen haasteeseen heidän sijoituksen perusteella (ks. kuvio 22).

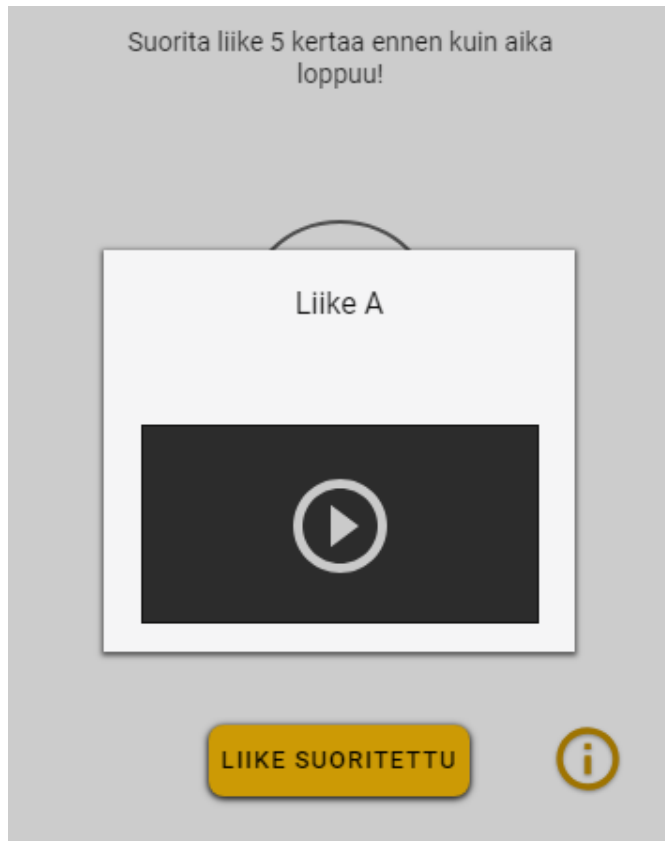


Kuvio 22 Liikkeen suoritus ja ohjeet

5.2 Käyttäjätestauksen tulokset

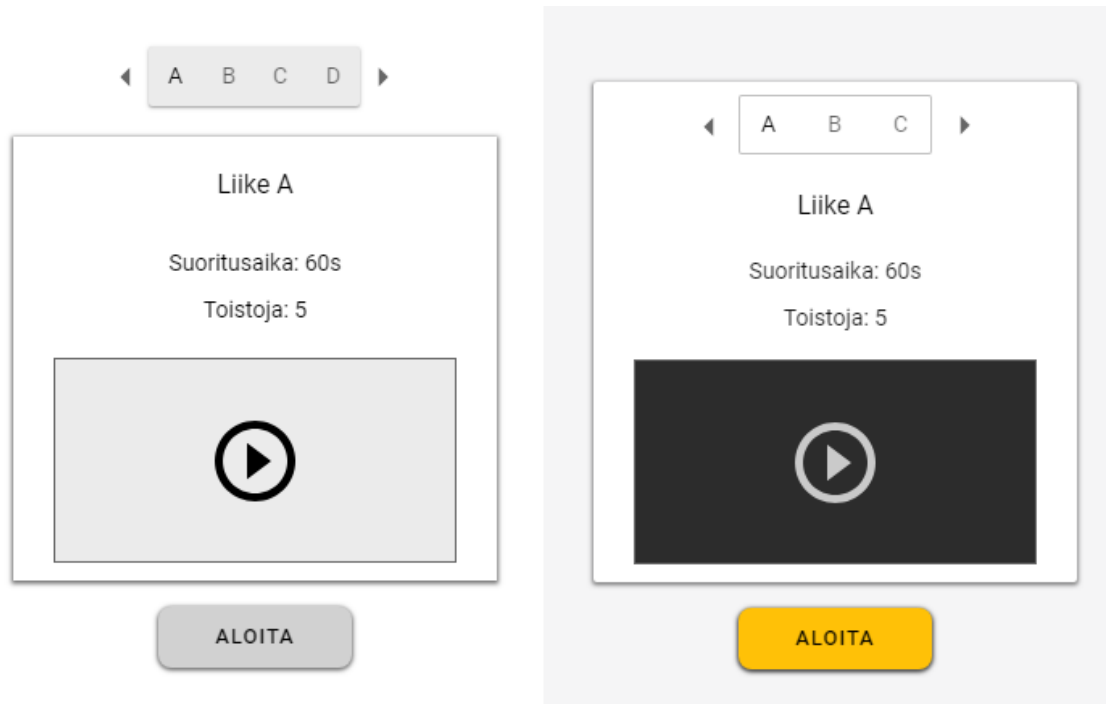
Käyttäjätestissä oli kolme tehtävää. Ensimmäisessä oli tarkoitus luoda käyttäjätili ja liittyä ryhmään. Toinen tehtävä oli haastetehtäviin tutustuminen ja haastetehtävän valinta. Viimeisenä oli haasteen suoritus. Jokaisesta tehtävästä tehtiin havaintoja käyttäjän tekojen ja kommenttien perusteella.

Merkittävimpiin havaintoihin testauksessa kuului se, että vain yksi kuudesta käyttäjästä ei välittömästi tunnistanut Material Designissa käytettyä taustan tummennusta konventiona, jota näpäyttämällä suljetaan edessä oleva ikkuna (ks. kuvio 23).



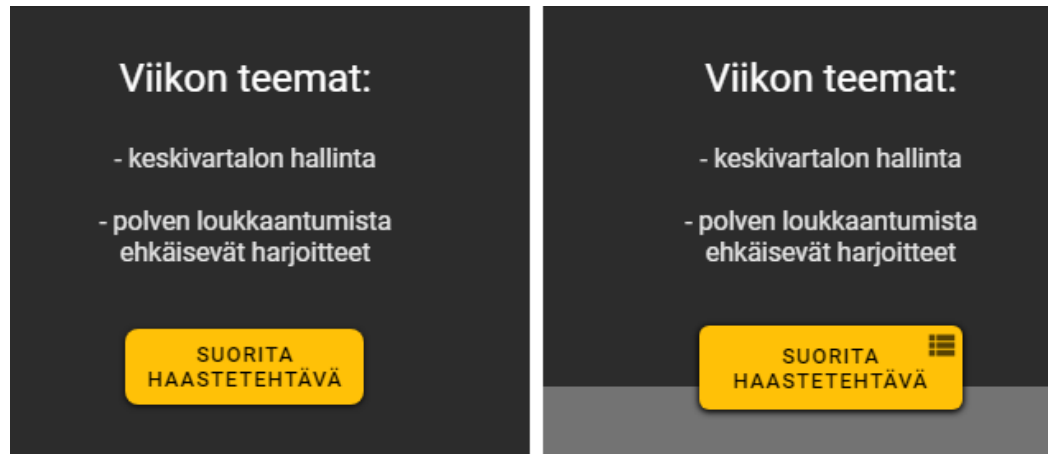
Kuvio 23 Taustan tummennus prototyypissä

Eniten hämmennystä prototyypissä aiheutti liikkeiden selaus, joka oli testiversiossa irrallinen ikkuna liikkeistä. Lopullisessa prototyypissä liikkeiden selauksen painikkeet siirrettiin korttien sisälle. Kuvion vasemmalla puolella näkyy vanha asettelu ja oikealla uusi (ks. kuvio 24).



Kuvio 24 Liikkeiden selaus

Toinen käytettävyysoongelma havaittiin etusivun ”Suorita haastetehtävä” -painikkeen kanssa, jolla navigoidaan haastetehtäviin. Tämä painike ei ollut tarpeeksi keskeisessä roolissa sivulla ja jossain tapauksissa ei ollut käyttäjän ensimmäinen vaihtoehto, kun käyttäjää pyydettiin tutustumaan haastetehtäviin. Kuviossa vasemmalla puolella testiversio ja oikealla uusin (ks. kuvio 25).



Kuvio 25 Haastetehtävien avaus

Muut käyttäjän havainnot epäselvistä toiminnallisuuksista otettiin huomioon loppullisen käyttöliittymään. Ylipäättään kaikilta käyttäjiltä jäi positiivinen kuva sovelluksen käytöstä. Käytettävyysoongelmia löydettiin kaksi. Näistä havainnoista sai vastauksen tutkimuskysymykseen ”Mikä on käytettävyyden merkitys Material Design -muotokielessä?”.

5.3 Material Design suunnitteluprosessissa

Material Designin suunnitteluperiaatteista sai pohjan sivujen rakenteelle ja sisällön asettelulle. Visuaalisessa suunnittelussa Material Designin värityökalu auttoi luomaan automaattisesti eri sävyjä väreistä, jotka oli valittu väriteemaan. Värityökalusta sai selvitettyä myös, mikä tekstin väri luo eniten kontrastia kunkin värin päällä ja mikä jokaisen värin alhaisin luettava tekstin läpinäkyvyys on.

Käyttöliittymän suunnitteluprosessin havainnollistamalla sai vastauksen tutkimuskysymykseen ”Miten Material Design -muotokieltä voidaan hyödyntää Game of Skills -sovelluksen käyttöliittymän suunnittelussa?”.

6 Pohdinta

Tutkimuksen tavoitteena oli soveltaa Material Designia käyttöliittymän suunnitteluprosessissa ja testata, missä käytettävyyssongelmia havaittaisiin.

Tutkimus osoittaa, että sovelluksen käyttöliittymän suunniteluun on mahdollista soveltaa Material Designin suunnitteluperiaatteita alusta asti, joka käyttäjätestauksen perusteella auttoi luomaan hyvin käytettävän pohjan sovellukselle. Heuristinen arviointi ja käyttäjätestaus ovat silti hyödyllisiä tapoja löytää käytettävyyssongelmia, mutta Material Designin suunnitteluperiaatteita noudattamalla säästyy ylimääräisten iteraatioiden kehitykseltä prototypisoinnissa.

Työssä kehitetyn prototyypin toiminnallisuuksien määrä rajaa sen, kuinka monipuolisesti Material Designin eri osia voidaan soveltaa käyttöliittymän suunnittelussa ja testata niiden mahdollisia käytettävyyssongelmia. Testauksessa oli myös käytetty rajallinen määrä käyttäjiä. Testauksesta sai käsityksen siitä, minkälaiset käytettävyyssongelmat ovat yleisimpiä sovelluksessa, joka on suunniteltu Material Designin avulla.

Tutkimustulokset tulevat olemaan hyödyllisiä sekä toimeksiantajalle, että käyttäjille. Toimeksiantaja saa prototyypin sovelluksen mobiiliversiosta, jonka voi viedä jatkokehitykseen ja käyttäjät saavat käytettävän sovelluksen.

Koska Material Design on jatkuvasti päivittyvä dokumentti, Googlen testaamat nykypäivän konventiot käyttöliittymissä päivitetään muotokieleeseen jatkuvasti (Material Design n.d). Tästä syystä Material Design on hyödyllinen resurssi käyttöliittymäsuunnittelijoille.

Lähteet

Garret, J. 2002. The Elements of User Experience: User-Centered Design for the Web and Beyond. Thousand Oaks: New Riders Publishing. Viitattu 30.7.2018.

Kananen, J. 2008. Kvali : kvalitatiivisen tutkimuksen teoria ja käytänteet. Jyväskylä: Jyväskylän Ammattikorkeakoulu. Viitattu 10.4.2018.

Krug, S. 2006. Älä pakota minua ajattelemaan : tervejärkinen käsitys web-käytettävyydestä. Helsinki: Readme.fi. Viitattu 24.4.2018.

Material Design. N.d. Google. Viitattu 11.5.2018. <https://material.io/design/>.

Mitchell, P. 2007. A step-by-step guide to usability testing. Lincoln (NE): iUniverse cop. Viitattu 22.8.2018.

Nielsen, J. 1995. 10 Heuristics for User Interface Design. Nielsen Norman Group, Training, and Consulting. Viitattu 5.5.2018. <https://www.nngroup.com/articles/ten-usability-heuristics/>.

Nielsen, J. 1993. Usability engineering. Boston: Academic Press. Viitattu 3.4.2018.

Norman, D. 2002. The design of everyday things. New York: Basic Books. Viitattu 8.5.2018.

Sinkkonen, I., Kuoppala, H., Parkkinen, J. & Vastamäki, J. 2006. Käytettävyyden psykologia. Helsinki: Edita, IT Press. Viitattu 24.4.2018.

Soini, J. 2016. Web-sovellusten käytettävyyden arviointi heuristiikkojen avulla. Kandidaatintyö. Oulun Yliopisto, tietojenkäsittelytieteiden koulutusohjelma. Viitattu 9.7.2018. <http://urn.fi/URN:NBN:fi:oulu-201612103257>.

W3C. 2008. Verkkosisällön saavutettavuusohjeet (WCAG) 2.0. W3C-suositus. Viitattu 29.7.2018. <https://www.w3.org/Translations/WCAG20-fi/WCAG20-fi-20110216/>.