

Opinnäytetyö (AMK)

Tietojenkäsittely

2019

Mira Pohjola

**E-TERVEYSPALVELUIDEN  
KÄYTETTÄVYYDEN  
KEHITTÄMINEN  
INTERAKTIIVISILLA  
GRAAFISILLA ELEMENTEILLÄ**

– CASE Hoito-ohjeet.fi

Mira Pohjola

# E-TERVEYSPALVELUIDEN KÄYTETTÄVYYDEN KEHITTÄMINEN INTERAKTIIVISILLA GRAAFISILLA ELEMENTEILLÄ

- CASE: Hoito-Ohjeet.fi

Opinnäytetyön tavoitteena oli kartoittaa e-terveyspalveluihin liittyviä yleisiä sekä sosiaali- että terveysalalle erityisiä käytettävyyteen sekä esteettömyyteen liittyviä ongelmia. Lisäksi työssä kartoitetaan ongelmien ratkaisemiseksi tarjolla olevia teknologioita sekä sovelluskehitysprosessin tarjoamia työkaluja. Tehdyn kartoituksen pohjalta opinnäytetyössä toteutetaan ammattilaisen laatimia hoito-ohjeita tarjoavalle terveysalan verkkopalvelulle uusi käytettävyyttä parantava visuaalinen interaktiivinen käyttöliittymä. Käyttöliittymän toteutuksen tavoitteet olivat parantunut käytettävyys, erityisesti liittyen palvelun käyttöön mobiililaitteilla, helppo ylläpidettävyys sekä helppo jatkokehittävyys. Lisäksi käyttöliittymälle asetettiin tiettyjä teknisiä vaatimuksia.

Opinnäytetyössä käy ilmi, että e-terveyspalveluiden käytettävyyden ongelmat ovat suurilta osin samoja kuin yleiset käytettävyyden ongelmat. Näiden palveluiden käyttöympäristöt ja käyttäjäryhmät kuitenkin korostavat käytettävyysongelmista johtuvien virheiden, käyttörajoitusten ja stressin vakavuutta ja seurauksia. Näitä ongelmia voidaan kuitenkin ratkaista käyttämällä tiettyjä tekniikoita sekä ohjelmistokehityksen prosesseja.

Teknisen toteutuksen suunnitteluun osallistui ohjausryhmä, joka koostui sekä teknisen, että terveysalan ammattilaisista. Lisäksi teknisessä toteutuksessa käytettiin teoriassa käsiteltäviä teknisiä apukeinoja käytettävyyden ja esteettömyyden varmistamiseksi.

Toteutettu käyttöliittymä ei täysin vastaa kaikkia sille asetettuja tavoitteita ja tutkimuksen tuloksena huomattiin, että osa tavoitteista ja teknisistä vaatimuksista ovat ristiriidassa keskenään. Ristiriitojen ratkaisemiseksi esitetään ratkaisuehdotuksia. Käytettävyyteen liittyvät tavoitteet voidaan kuitenkin katsoa täytetyksi niissä määrin, mikä oli työn kokonaistavoitteen kannalta.

ASIASANAT:

Käytettävyys, ohjelmistokehitys, esteettömyys

BACHELOR'S / MASTER'S THESIS | ABSTRACT

TURKU UNIVERSITY OF APPLIED SCIENCES

Information Technology

2019 | 29 pages

**Mira Pohjola**

Improving Usability of E-Health Applications with an Interactive Graphical Elements

Case: Hoito-ohjeet.fi

The goal of the thesis was to map out usability and accessibility challenges commonly affecting E-Health applications. Additionally, the work presents some technical tools and methods of software development and lifecycle management which can be used to mitigate the presented issues. The second part of the thesis covers designing and developing a new visual, interactive navigation module for Hoito-Ohjeet.fi e-health web application. The goal of the module was to heighten the usability and accessibility of the service. There were also some technical requirements placed for the new module.

The thesis outlines that most of the usability challenges faced by e-health applications rise from general web application usability issues. However, because of the nature of these services and the environments where these services are used in the errors, accessibility restrictions and user stress caused by poor usability have more severe consequences. However, there are technologies and software development methods which can be used to mitigate these issues.

A steering group consisting of healthcare professionals and software engineers took part in the planning phase of the navigation module. The technologies and methods for improving usability and accessibility presented by the thesis were also utilized in the development of the navigation module.

While the developed navigation module meets the requirements for usability and accessibility, not all placed technical and functional requirements were fulfilled due some of the requirements contradicting each other. Suggestions for resolving these issues, such as changing some technical aspects, were investigated and listed for future development of the navigation module.

**KEYWORDS:**

Usability, software development, accessibility

# SISÄLTÖ

<b>SANASTO</b>	<b>6</b>
<b>1 JOHDANTO</b>	<b>1</b>
<b>2 VERKKOSIVUMODUULIN KEHITYKSEEN TARVITTAVAT TEKNIIKAT</b>	<b>2</b>
2.1 Yleiset käytettävät tekniikat	2
2.2 TypeScript kehityskielenä	3
2.3 Käytettävyyttä ja esteettömyyttä tukevat tekniset ominaisuudet	4
<b>3 KÄYTTÖLIITTYMÄSUUNNITTELUN PERIAATTEET</b>	<b>6</b>
3.1 Verkkosivun ulkoasun suunnittelu	7
3.2 Käytettävyyssuunnittelu	8
<b>4 KÄYTETTÄVYYS JA KÄYTTÄJÄYSTÄVÄLLISYYS TERVEYS JA SOSIAALIALAN VERKKOPALVELUISSA</b>	<b>10</b>
4.1 Käytettävyysongelmat e-terveyspalveluissa	10
4.2 Käytettävyyden kehittäminen ja varmistaminen	11
<b>5 HOITO-OHJEET.FI-PALVELUN KUVAUS</b>	<b>14</b>
5.1 Käyttäjryhmät sekä käyttöympäristöt	14
5.2 Kehitystavoitteet	14
<b>6 IHMISKUVA-KÄYTTÖLIITTYMÄN SUUNNITTELU</b>	<b>16</b>
6.1 Käyttöliittymän kuvaus	16
6.2 Käytettävyyys ja käyttäjäystävällisyys	17
<b>7 IHMISKUVA-KÄYTTÖLIITTYMÄN TEKNINEN TOTEUTUS</b>	<b>21</b>
7.1 Teknisen toteutuksen vaatimukset	21
7.2 Toteutukseen käytetyt tekniikat	21
7.3 Teknisen toteutuksen ongelmat ja niihin liittyvät ratkaisuehdotukset	23
7.4 Tulosten arviointi	24
7.4.1 Esteettömyys	26
7.4.2 Ylläpidettävyys ja jatkokehitys	27
<b>8 LOPUKSI</b>	<b>29</b>

## LÄHTEET

30

## KUVAT

Kuva 1. Verkkosivun sisällön asettelu suurella ja pienellä näyttölaitteen koolla.	8
Kuva 2. Yksi uuden käyttöliittymäelementin funktionaalisen prototyypin työpöytä näkymistä.	17
Kuva 3. Ihmiskuva käyttöliittymän varhaisen version työpöytä näkymä (versio 1.0).	18
Kuva 4. Uuden käyttöliittymäelementin mobiilinäkymiä.	19
Kuva 5. Yksi teknisen toteutuksen näkymistä.	24
Kuva 6. Osa käyttöliittymän esiprosessoitua HTML-koodia, jossa näkyy sekä HTML role ja WAI-ARIA attribuuttien käyttöä.	26

## KUVIOT

Kuvio 1. Käytön sujuvuuteen liittyvät käytettävyyssperiaatteet muunnetulla Nielsenin heuristiikalla. (Kaipio & Vehmas, 2018)	12
Kuvio 2. Käytön luotettavuuteen liittyvät käytettävyyssperiaatteet muunnetulla Nielsenin heuristiikalla. (Kaipio & Vehmas, 2018)	13
Kuvio 3. Kehonkuva käyttöliittymän toimintopainikkeet.	20

## TAULUKOT

Taulukko 1. Käyttöliittymäelementin käytettävyyden arviointi käyttäen muunnettua Nielsenin heuristiikkaa	26
--	----

# SANASTO

ARIA	Accessible Rich Internet Applications. WAI-työryhmän laa- tima erityisesti dynaamisia verkkopalveluita koskettava es- teettömyystandardi.
Backend	Verkkopalvelun palvelin- sekä taustasovellukset
CSS	Cascading Style Sheets. Määrittely, jolla kuvataan miten verkkosivun HTML elementit näytetään
DOM	Document Object Model. HTML-sivun elementtien hierarki- nen rakenne
E-terveyspalvelu	Sosiaali- ja terveysalan digitalisoitu palvelu
Framework	Ohjelmakehys
Frontend	Verkkopalvelun käyttöliittymä
HTML	Hyper Text Markup Language. Kuvauskieli, jolla kuvataan miten selaimen tulisi esittää verkkosivun sisältö
SVG	Scalable Vector Graphics. XML perustainen skaalaava vek- torigrafiikkaformaatti.
W3C	World Wide Web Consortium. Kansainvälinen yhteisö, joka vastaa verkkoon liittyvien standardien ylläpidosta ja kehittä- misestä.
WAI	Web Accessibility Initiative. W3C:n työryhmä, joka vastaa es- teettömyyteen liittyvien strategioiden, ohjeistuksien ja stan- dardien kehittämisestä ja ylläpidosta.
Webpack	JavaScript moduulipaketoija. (Vepsäläinen 2018)
XML	Extensible Markup Language. Merkitsemiskieli, joka määrit- telee joukon sääntöjä dokumenttien enkoodaamiseen for- maatissa, joka sekä ihmis-, että koneluettava.

# 1 JOHDANTO

Sosiaali- ja terveysalan palvelut ovat tavallisesti olleet tavoitettavissa puhelimitse tai käymällä itse paikan päällä. Huonot kulkuyhteydet ja palveluiden rajoitetut aukioloajat aiheuttivat tällöin palveluiden tavoitettavuuteen liittyviä ongelmia, joka on johtanut palveluiden asteittaiseen siirtämiseen verkkoon. Palveluilla on palvelusta riippuen erilaisia käyttäjäryhmiä ja käyttötarkoituksia, joista yksi keskeisiä on tärkeän ja luotettavan terveyteen liittyvän tiedon ja tietämyksen jakaminen niin ammattilaisen kuin kansalaistenkin saataville. (Rosenlund ja Kinnunen 2018)

Näiden palveluiden käyttötarkoituksen ja käyttäjäryhmien takia on erityisen tärkeää, että palvelut ovat hyvin suunniteltuja, helppoja käyttää ja niiden sisältö on kaikille tasapuolisesti saatavilla. Huonosti suunnitellut tai käytettävyydeltään heikot sosiaali- ja terveysalan palvelut haittaavat ammattilaisten työtä aiheuttaen heille stressiä, ja toisaalta myös johtavat eriarvostumiseen kansalaisten keskuudessa. (Rosenlund ja Kinnunen 2018)

Tässä työssä kartoitetaan, miten yleisiä käyttöliittymä- ja käytettävyyssuunnittelun periaatteita soveltaen voidaan parantaa sosiaali- ja terveysalan palveluiden käytettävyyttä. Lisäksi työssä suunnitellaan ja toteutetaan näitä periaatteita toteuttava uusi tiedonhaku helpottava interaktiivinen käyttöliittymäelementti Hoito-ohjeet.fi-palveluun.

Opinnäytteen tutkimusmenetelmä on toiminnallinen ja tutkimusote konstrukttiivinen.

## 2 VERKKOSIVUMODUULIN KEHITYKSEEN TARVITTAVAT TEKNIIKAT

Verkkopalveluiden kehittämiseen on nykyisin saatavilla useita erilaisia tekniikoita, joista osa on rajattavissa backend ja osa frontend tekniikoiksi. Osa moderneista verkkosovellus ohjelmakehyksistä voidaan katsoa kuuluvaksi molempiin luokkiin. Tässä työssä puhuttaessa verkkosivumoduulista, rajataan toteutus frontend toteutukseen, ottamatta kantaa siitä, millä tekniikoilla verkkopalvelun palvelinsovellus on toteutettu.

### 2.1 Yleiset käytettävät tekniikat

Jokainen verkossa saatavilla oleva sovellus rakentuu verkko-ohjelmoinnin perustekniikoiden päälle. Nämä perustekniikat ovat HTML ja CSS. HTML, eli Hyper Text Markup Language on nimensä mukaisesti merkintäkieli, joka antaa verkkoselaimelle ohjeita siitä, millä tavalla verkkosivu tulisi näyttää ja mistä osista se koostuu. CSS, eli Cascading Style Sheets, on sen sijaan HTML:n tukena käytettävä tapa merkitä verkkosivujen elementtien erilaisia tyylejä kuten käytettäviä kirjasimia, värejä ja kokoja. Yhdessä nämä kaksi tekniikkaa mahdollistavat sivujen piirtämisen verkkoselaimessa. (Robbins 2018)

Molemmat edellä mainitut tekniikat ovat suunniteltu kehittymään ja edistymään niin, että kaikki vanhemmilla tekniikkaversioilla toteutetut sovellukset ja moduulit toimivat myös uudempien versioiden kanssa. Uusia versioita ja standardeja näistä tekniikoista julkaistaan edelleen ja nämä uudet versiot ovat mainitulla tavalla taaksepäin yhteensopivia, ja lisäävät uusia ominaisuuksia ja optimointeja vanhojen ominaisuuksien rinnalle. (Gustafson 2011)

Vuoden 2019 kesällä uusimmat versiot mainituista tekniikoista ovat HTML5 ja CSS3, jotka toivat mukanaan suuren määrän uusia, erityisesti mukautettavuutta, käytettävyyttä ja responsiivisuutta tukevia ominaisuuksia. Esimerkiksi HTML5:n mukana tulleiden erityisten elementtien tai elementeille asetettavan 'role'-attribuutin käyttö mahdollistavat sen, että näkövammaisille tarkoitetut ruudun lukulaitteet osaavat tulkita elementtien tarkoituksen. (Hogan 2013)



HTML ja CSS tekniikoilla voidaan siis näyttää sisältöä verkkosivuilla ja jo pelkästään näiden tekniikoiden avulla voidaan toteuttaa esimerkiksi yksinkertainen navigointi ja linkitys eri staattisten verkkosivujen välille.

Kuitenkin mikäli halutaan toteuttaa verkkosivuille monimutkaisempaa toiminnallisuutta tai vastavuoroisuutta käyttäjän kanssa, tarvitsee näiden tekniikoiden lisäksi verkkosivulla käyttää JavaScript-ohjelmointikieltä. JavaScript on ainoa ohjelmointikieli, jota verkkoselaimet tukevat, ja se suoritetaan käyttäjän selaimessa. Tämä ohjelmointikieli kykenee lukemaan ja manipuloimaan HTML:n muodostavaa DOM (Document Object Model) puuta, eli lukemaan, muokkaamaan, lisäämään ja poistamaan esitettäviä elementtejä. (Robbins 2018)

JavaScriptin käyttö mahdollistaa siis monimutkaisempien ja interaktiivisempien käyttöliittymien luomisen, jokseenkin sillä huomiolla, että selaimien tuki eri JavaScript versioihin voi vaihdella, joka luo uusia haasteita aidosti käytettävien ja käyttäjäystävällisten verkkosovellusten tekemiseen. (Robbins 2018)

Lisäksi laajamittainen JavaScriptin käyttö luo haasteita verkkopalvelun esteettömyydelle, sillä esimerkiksi ruudunlukulaitteet käsittelevät huonosti dynaamisesti tuotettua ja JavaScriptillä manipuloitavaa sisältöä. (Connor 2012)

Viime vuosina on kehitetty suuri määrä uusia verkkokehitystekniikoita ja viitekehyksiä, kuten AngularJS, VueJS ja TypeScript, jotka tarjoavat helpotusta ja laajennuksia puhtaaseen JavaScriptiin verraten. Koska verkkoselaimet kuitenkin suorittavat vain JavaScript-koodia, kaikkien näiden eri ohjelmakehysten lähdekoodi kuitenkin käännetään puhtaaksi JavaScriptiksi käyttäen erilaisia kyseiselle kehykselle tai kielelle tarkoitettua kääntäjiä. (Robbins 2018)

## 2.2 TypeScript kehityskielenä

TypeScript on JavaScriptin vahvasti tyypitetty ylijoukko, joka toteuttaa kaikki JavaScriptin ominaisuudet. TypeScript käyttää pohjana JavaScriptin syntaksia, jota se laajentaa tyyppimäärittäyksillä. TypeScript moduulien kanssa voidaan käyttää JavaScript moduuleja sekä ohjelmakirjastoja, TypeScript-moduulien ja kirjastojen lisäksi. Lisäksi TypeScript tulee aikaisessa vaiheessa uusimpia JavaScriptin ominaisuuksia. TypeScript-tiedostot

eroavat JavaScript-tiedostoista koodin syntaksin lisäksi ts-tiedostopäätteestä. Koska verkkoselaimet pystyvät suorittamaan vain JavaScriptiä, TypeScript-tiedostot käännetään JavaScript-tiedostoiksi. (Microsoft ei pvm)

Pääasiallinen syy TypeScriptin käyttöön kehityskielenä verkko-ohjelmoinnissa ovat koodin laadun parantaminen sekä virheiden vähentäminen. Vahva tyyppitys auttaa poistamaan heikosta tai dynaamisesta tyyppityksestä johtuvia tavallisimpia virheitä. Artikkelin 'To Type or Not to Type: Quantifying Detectable Bugs in JavaScript' mukaan noin viisi-toista prosenttia GitHubiin ladattujen JavaScript projektien koodista esiintyneistä ohjelmointivirheistä oltaisiin voitu havaita aikaisessa vaiheessa käyttämällä TypeScriptin tarjoamaa vahvaa tyyppitystä. (Gao; Bird ja Barr 2017)

Virheiden vähentämisen lisäksi vahvan tyyppityksen käyttö tarjoaa sovelluskehittäjän tueksi erilaisia apuja, kuten älykkäitä kehitystyökaluja. TypeScriptin vaatima rajapintaluokkien sekä metodien parametrien ja paluuarvotyyppien määrittäminen tekee TypeScript koodista lisäksi ekspressiivisempää ja helpommin ylläpidettävää. Ekspressiivinen koodi on helpommin jatkokehitettävää ja ymmärrettävämpää projektiin osallistuville uusille sovelluskehittäjille. (Tomaszewski 2018)

### 2.3 Käytettävyyttä ja esteettömyyttä tukevat tekniset ominaisuudet

Luvussa 2.1 esitellyt HTML 5 esteettömyyslaajennukset tarjoavat hyvän lähtökohdan esteettömien verkkopalveluiden kehitykselle. Osa verkkopalveluista nämä HTML standardin mukaiset ominaisuudet ovat riittävät siihen, että ruudunlukulaitteita käyttävät käyttäjät pystyvät selvittämään palvelun toiminnallisuuden sekä käyttämään palvelua. Rakenteeltaan monimutkaisemmat, runsaasti JavaScriptiä hyödyntävät ja dynaamista sisältöä tarjoavat verkkopalvelut joutuvat kuitenkin turvautumaan lisäksi muihin tekniikoihin. (Connor 2012)

Esteettömien verkkopalveluiden kehityksen tueksi on määritelty useita erilaisia standardeja. Suurin esteettömän internetin eteenpäin viejä on WAI (Web Accessibility Initiative), joka on W3C konsortion esteettömyystyöryhmä. WAI on laatinut suuren määrän verkon esteettömyyteen liittyviä strategioita, standardeja sekä ohjeistuksia. (W3C ei pvm)

Tunnetuin HTML 5 esteettömyyttä laajentava esteettömyysmäärittäminen on WAI-ARIA, 'Web Accessibility Initiative: Accessible Rich Internet Applications'-määrittäminen. ARIA määrittäminen sisältää runsaan määrän erityisesti dynaamista tai mediasisältöä sisältöä tarjoavien

verkkopalveluiden esteettömyyttä tukevia ominaisuuksia. Ruudunlukulaitteet osaavat käyttää ARIA-attribuutteja ja määrittelyjä hyväkseen tulkitessaan verkkopalvelun toimintaa. (Connor 2012)

Esimerkki merkittävästä ARIA-määrittelyn mukaisesta attribuutista on live-aria-attribuutti. Live-aria attribuutin kiinnittäminen HTML DOM-elementtiin kertoo ruudunlukulaitteelle, että elementti sisältää dynaamisesti päivittyvää sisältöä. Tällöin ruudunlukulaite osaa seurata sisällön muutoksia ja ilmoittaa käyttäjälle muuttuneesta sisällöstä. Ilman live-aria-attribuuttia ruudunlukulaitetta käyttävä käyttäjä ei saa tietoa dynaamisesti muuttuvasta elementin sisällöstä. (Web Accessibility Initiative 2017)

### 3 KÄYTTÖLIITTYMÄSUUNNITTELUN PERIAATTEET

Käyttöliittymän suunnittelu voidaan jakaa kahteen osaltaan toisistaan erottuvaan osaan; käyttöliittymän ulkoasun suunnitteluun sekä sen toiminnallisuuden suunnitteluun. Usein näitä kahta osaa käsitellään erillään toisistaan ja monissa tapauksissa ulkoasun ja toiminnallisuuden suunnittelu toteutetaan eri ihmisten tai eri tiimien toimesta. Näitä kahta osaa käyttöliittymän suunnittelusta ei kuitenkaan voida aidosti erottaa ja yhdessä ne muodostavat kokonaisuuden, josta nykyisin puhutaan esimerkiksi nimillä 'käytettävyys suunnittelu' (engl. usability design) sekä 'käyttäjäkokemuksen suunnittelu' (engl. user experience design). (Interaction Design Foundation ei pvm)

Nykyaikaisessa verkkokäyttöliittymien suunnittelussa pyritäänkin mahdollisimman korkeaan käytettävyteen ja tavoitettavuuteen. Verkkosovelluksen mahdollisten käyttäjien tulisi olla mahdollista käyttää sovellusta riippumatta heidän käyttösijainnistaan tai laitteesta, jolla he sovellusta käyttävät. Haastetta tähän tavoitteeseen tuovat eri järjestelmien ja verkkoselainten ominaisuudet ja eroavaisuudet. Esimerkiksi verkkoselainten tarjoama tuki uudempien verkkotekniikoiden, kuten HTML5:n ominaisuuksille ei ole yhdenvertainen (Gustafson 2011)

Verkkopalveluiden käytettävyttä ja tavoitettavuutta usein mitataan sillä, kuinka hyvin ne palvelevat ihmisiä, joilla on joitakin erityistarpeita, kuten ongelmia näön tai hienomotoriikan kanssa. On kuitenkin hyödyllistä katsoa käytettävyttä laajemmin niin, että tunnustetaan jokaisella käyttäjällä olevan erilaisia tarpeita ja rajoituksia. Usein nämä rajoitukset saattavat liittyä käytettävissä olevaan tekniikkaan. (Gustafson 2011)

Aaron Gustafson kirjassa Adaptive Webdesign toteaaakin, että verkkosivujen tärkein osa on sivun sisältö ja nimenomaan tämän sisällön tulisi olla saatavissa kaikille sivuston käyttäjille jollakin tavalla. Se, miten sisältö on saatavissa, voi vaihdella riippuen käyttäjän erityistarpeista, selaimesta tai päätelaitteesta, kuitenkin niin että sivusto toimii edelleen oikein. Ensimmäisenä periaatteena verkkosivujen toteutuksessa on siis se, että sisältö on aina tärkein ja saatavissa. Tämän pohjaperiaatteen toteuduttua voidaan verkkosovellukseen lisätä asteittaisia parannuksia, jotka nostavat käyttäjäkokemusta, mutta joiden puuttuminen ei vie käyttäjältä mahdollisuutta nähdä verkkosivun ydinsisältöä. (Gustafson 2011)

### 3.1 Verkkosivun ulkoasun suunnittelu

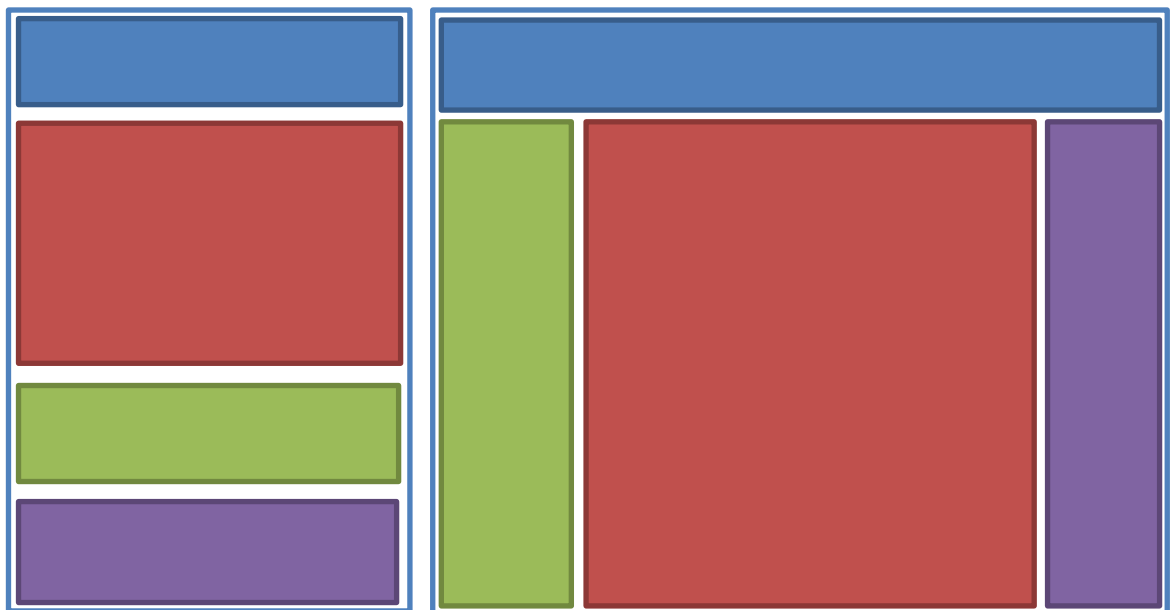
Verkkosivujen ulkoasun suunnittelussa tavallisesti kohdataan ongelmia, joita ei esiinny esimerkiksi suunniteltaessa sisältöä painotuotteisiin. Suurin näistä ongelmista on suuri näyttölaitteiden koon ja resoluution vaihtelu. Mikäli verkkokäyttöliittymä suunnitellaan ja toteutetaan samalla tavoin kuin painotuote käyttäen staattisia kokomäärytyksiä ja asette-luita, kyseinen käyttöliittymä näyttää hyvältä ja oikein toimivalta vain hyvin pienellä osalla laitteita. Tästä syystä verkkosivujen suunnittelussa pyritäänkin käyttämään mukautuvaa eli responsiivista ulkoasua, joka mukautuu käyttäjän päätelaitteen tai selaimen mukaan. (Keith 2018)

Responsiivisella suunnittelulla poistetaan tekniikasta johtuvia esteitä verkkosivun sisäl-lön tavoitettavuudelle ja päästään lähemmäksi edellisessä luvussa mainittua perustavoi-tetta. Käytännössä responsiivinen suunnittelu tarkoittaa sitä, että sivun elementeille an-netaan suhteellinen koko ja toisenaan myös sijainti. Suunnitteluprosessiin responsiivinen suunnittelu vaikuttaa niin, että sivulle ei voida luoda vain yhtä staattista ulkoasua, jonka oletetaan toimivan kaikilla eri päätelaitteilla. Sen sijaan sivulle luodaan joko useampi vaihtoehtoinen ulkoasu, tai vaihtoehtoisesti yksi ei-staattinen ja itsekseen mukautuva ulkoasu. (Keith 2018)

Modernissa verkkosivujen suunnittelussa usein lähdetään liikkeelle ns. 'mobile first', eli mobiililaitteet ensin-periaatteella. Tämän periaate johtuu kahdesta eri tekijästä; mobiili-laitteita on nykyään suuri verkkoselailuun käytetyistä päätelaitteista ja siitä, että sivu, joka toimii hyvin mobiililaitteella, toimii aina kohtuullisesti myös suurempikokoisella pää-telaitteella. Mobiililaitteet-ensin periaate on myös osaltaan sisältölähtöisen suunnittelun periaate, sillä kun sivusto suunnitellaan toimimaan hyvin rajatun kokoisella ruudulla, yli-määräinen karsiutuu pois ja suunnittelussa fokus ohjautuu varsinaisen sisällön esittämi-seen niin, että käyttäjä saa sivustosta helposti irti sen tärkeimmän sisällön. (Keith 2018)

Voidaankin ajatella niin, että mobiililaitteet-ensin periaatteella suunniteltu verkkosivu to-teuttaa Gustafsonin kirjassa esitettyä sisältökeskeisen suunnittelun periaatetta. Suurem-milla päätelaitteilla sivustoa käytettäessä sivustoon voidaan lisätä erilaisia helpotteita ja asteittaisia parannuksia. Esimerkkinä tästä olisi perinteinen sisällön asettele verkkosi-vulla. Mobiililaitteelle suunnitellussa verkkosivussa sivuston sisältö asetetaan yhteen sa-rakkeeseen, jolloin sisällön lukeminen vaatii vain vieritystä ylös tai alas, joka on mobiili-laitteille ominainen tapa esittää sisältöä. Suuremmalla näyttökoolla sisältö voidaan sen

sijaan jakaa kahteen tai useampaan sarakkeeseen, jolloin sisältö on suuremmalla näytöllä nähtävissä ilman vierittämistä. Molemmissa tapauksissa sisältö on käyttäjälle saatavissa yksinkertaisella tavalla ja sisällön jakaminen kahteen sarakkeeseen on vain parannus jo toimivan esitystavan lisäksi. Kuvattua esitystapaa hahmotetaan alla olevassa kuvassa, jossa vasemmalla puolella esitetään sisällönasettelu mobiilikäyttöliittymässä ja oikealla puolella saman sisällön mahdollinen asettelu työpöytäkäyttöliittymässä. Kuvan 1 vierekkäisistä ruutunäkymistä näkee, miten asettelulla voidaan työpöytäkäyttöliittymissä käyttää hyväksi näyttölaitteen suurempaa kokoa.



Kuva 1. Verkkosivun sisällön asettelu suurella ja pienellä näyttölaitteen koolla.

### 3.2 Käytettävyyssuunnittelu

Käyttöliittymäsuunnittelu on osa suurempaa suunnitteluprosessia, jota voidaan kutsua nimellä käyttäjäkokemussuunnittelu (engl. user experience design tai UX design), jossa suunnittelu kattaa muutakin kuin tuotteen ulkoasun ja toiminnallisuuden. Käyttäjäkokemussuunnittelussa otetaan huomioon koko tuotteen elinkaari ja koko prosessi, jonka käyttäjä käy tuotteen kanssa, alkaen esimerkiksi prosessista tuotteen hankkimiseen. Käyttöliittymä- ja käytettävyyssuunnittelu ovat keskeisiä osia käyttökokemussuunnittelua. (Interaction Design Foundation ei pvm)

Käytettävyyssuunnittelussa asetetaan tuotteelle kolme vaatimusta, jotka hyvin käytettävän tuotteen tulisi täyttää: Käyttäjän on helppo oppia tuotteen käyttö; Käyttäjän on helppo suorittaa haluamansa toiminto; Käyttäjän on helppo muistaa, miten tuotetta käytetään ja käyttäjän on helppo ottaa tuota takaisin käyttöön. (Interaction Design Foundation ei pvm)

Käytettävyyssuunnittelun yksi yleisimmin käytettävistä työkaluista on käytettävyystestaus, jossa kehitettävää järjestelmää tai toiminnallisuutta testataan oikeilla käyttäjillä. Tällaista testausta voidaan suorittaa joko tuotteen kehityksen loppuvaiheessa tai useaan kertaan tuotteen kehityksen aikana, jolloin tuotteen käyttäjäryhmän palaute voi vaikuttaa kehitysprosessiin. (Interaction Design Foundation ei pvm)

## 4 KÄYTETTÄVYYS JA KÄYTTÄJÄYSTÄVÄLLISYYS

### TERVEYS JA SOSIAALIALAN VERKKOPALVELUISSA

Käytettävyyden ja käyttäjäystävällisyyden merkitys julkisissa sekä terveyteen ja hyvinvointiin liittyvissä palveluissa on muita palveluita suurempi. Mainitun tyyppisten palveluiden digitalisoinnilla pyritään lisäämään palveluiden tavoitettavuutta, mahdollistamalla palvelun käytön ajasta ja paikasta riippumatta. Terveys- ja sosiaalialan palveluiden digitalisoinnilla pyritään lisäksi aktivoimaan ihmisiä oman terveytensä ja hyvinvointinsa ylläpitoon ja seurantaan sekä tuomaan terveyteen liittyvää tietoa avoimesti saataville. (Rosenlund ja Kinnunen 2018)

Vaikka palveluiden digitalisoinnilla pyritäänkin korottamaan niiden tavoitettavuutta, digitalisointi aiheuttaa kuitenkin jonkinasteista eriarvoistumista, koska erityisesti ikäihmisten, työelämän ulkopuolella olevien ja matalan koulutuksen saaneiden valmiudet näiden palveluiden käyttöön ovat muita alemmat. (Rosenlund ja Kinnunen 2018)

Koska digitaalisten terveystalveluiden tulisi olla tasa-arvoisesti kaikkien kansalaisten käytettävissä, käytettävyyden ja asiakas- sekä sisältölähtöisyyden huomioonottaminen palveluiden suunnittelussa ja toteutuksessa on erityisen tärkeää. Mikäli palvelu on hankala käyttää tai sen toiminta rajataan vain vähäiseen määrään toimintaympäristöjä, voidaan pahentaa palveluiden käyttöön liittyvää eriarvoistumista.

#### 4.1 Käytettävyysongelmat e-terveyspalveluissa

Sosiaali- ja terveysalalle tyypillisiä stressin aiheuttajia ovat olleet aikapaine, sosiaalidynaamiset ongelmat, hankalat potilaat sekä työn määrä, mutta uudemmat tutkimukset osoittavat, että aikaa vievät, huonosti suunnitellut ja vaikeasti käytettävät digitaaliset palvelut aiheuttavat kasvavissa määrin stressiä alan työntekijöille. Lisäksi voidaan osoittaa aikaa vievien digitaalisten palveluiden käytön aiheuttavan haittaa perinteiselle lääkäripotilassuhteelle. (Heponiemi, ym. 2017)

E-terveysjärjestelmien käyttö voi viedä jopa kolmanneksen terveysalan ammattilaisen päivittäisestä työajasta. Lisäksi järjestelmän käyttö esimerkiksi potilastapaamisten aikana haittaa potilaan ja lääkärin välistä kanssakäymistä, mikäli järjestelmän käyttö on hidasta tai tehotonta. Lisäksi terveysalan ammattilaiset toistuvasti arvioivat, että nämä



palvelut tukevat heidän työtään huonosti, riippumatta siitä ovatko he kokeneita järjestelmän käyttäjiä vai eivät. e-terveysjärjestelmien toimintavirheet ovat myös aiheuttaneet monissa tapauksissa vakavaa haittaa potilaiden terveydelle. (Kaipio ja Vehmas, Physicians as usability evaluators – first aid for poor EHR usability? 2018)

Terveys- ja sosiaalialan verkkopalveluiden käytettävyyden ongelmat ovat usein yleisiä käytettävyyteen liittyviä ongelmia, joiden merkitys korostuu toimintaympäristössä, jossa sovelluksen käyttöön tai toimintaan liittyvät virheet saattavat aiheuttaa odottamattomia seurauksia. E-terveyspalveluiden käytettävyysongelmien voidaan katsoa johtuvan kahdesta alkutekijästä; Ohjelman tai palvelun suunnittelussa ei ole otettu huomioon ohjelman lopullista käyttäjäryhmää ja ohjelma tai palvelu joko yrittää suorittaa liian montaa toimintoa kerralla tai siinä on huonosti huomioitu ohjelman tai palvelun pääasiallinen käyttötarkoitus. (Yen ja Bakken 2011)

Potilaat kokevat palveluiden käytön ongelmakohtien usein liittyvän niiden käytettävyyteen kuten ulkoasuun sekä huoleen palveluiden tietoturvasta. Erityisesti esiin nousevia käytettävyyden haittoja ovat palvelun rakenteen, käyttöliittymän tai navigoinnin monimutkaisuus, käytettävien painikkeiden koko ja asettelu sekä käytettävien kirjasimien koko ja tyyli. Eri esitystapojen, kuten videon tai kuvan ja tekstin yhdistämisen ja graafisten elementtien käytön koetaan parantavan palvelun käytettävyyttä. (Rosenlund ja Kinnunen 2018)

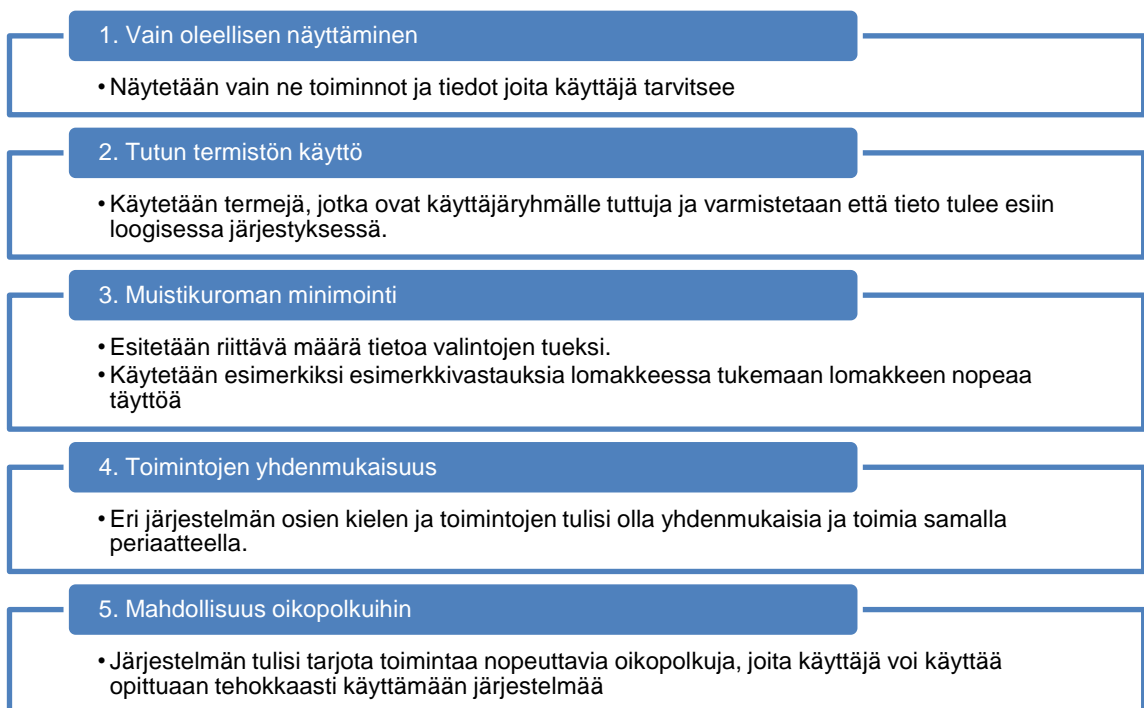
#### 4.2 Käytettävyyden kehittäminen ja varmistaminen

Käytettävyyden ja käyttäjäystävällisyyden huomioonottamista suositellaan koko ohjelmakehityksen ajan. Mahdollisia menetelmiä e-terveyspalvelun käytettävyyden arvioinnille ovat esimerkiksi loppukäyttäjien tekemä testaus sekä käyttäjäkokemukseen liittyvä heuristinen arviointi. (Rosenlund ja Kinnunen 2018)

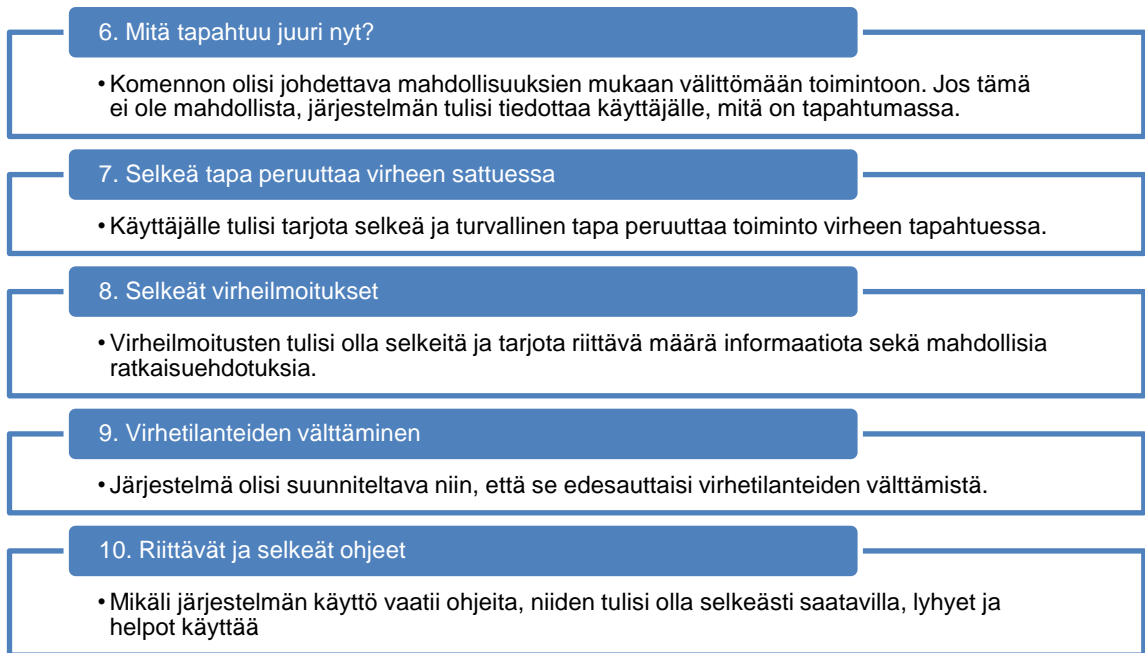
Monet Suomen terveysalan ammattilaiset ovat kyselyissä osoittaneet kiinnostusta osallistumaan e-terveyspalveluiden kehittämiseen sekä osoittaneet valmiutta keskustella palvelun käytettävyyteen liittyvistä seikoista esimerkiksi kollegoiden kanssa, joilla on yhteyksiä järjestelmän kehittäjiin. Kuitenkin useat käytettävyyden kehittämiseen osallistuneet terveysalan ammattilaiset ovat kokeneet, että järjestelmän kehittäjät eivät ole olleet kiinnostuneita saamastaan palautteesta tai heihin on ollut vaikea saada yhteyttä. (Kaipio ja Vehmas, Physicians as usability evaluators – first aid for poor EHR usability? 2018)

Sovellettaessa käytettävyyssuunnittelun ja käyttöliittymäsuunnittelun periaatteita sosi- ja terveysalan palveluihin olisi kuitenkin otettava huomioon alan erityispiirteitä. Artikkelissa Johanna Kaipio artikkelissa “Physicians as usability evaluators – first aid for poor EHR usability?” suosittelee lähestymistapaa, jossa jo järjestelmän suunnitteluvaiheessa otetaan prosessiin aktiivisesti mukaan myös terveysalan ammattilaisia. (Kaipio ja Vehmas, Physicians as usability evaluators – first aid for poor EHR usability? 2018)

Käytettävyysongelmat voidaan jakaa kahteen kategoriaan; Ongelmiin, jotka liittyvät käytön sujuvuuteen, ja ongelmiin, jotka liittyvät käytön luotettavuuteen, joihin vastaa 10 Jacob Nielsenin käytettävyysheuristiikkaa. Artikkelissa “Physicians as usability evaluators – first aid for poor EHR usability?” on käytetty mukautettua Nielsenin heuristiikkaa, joka vastaa paremmin nimenomaan e-terveyspalveluiden käytettävyystarpeisiin. Heuristiikat selityksineen on kuvattu kahdessa kuviossa niin, että käytön sujuvuuteen liittyvät periaatteet ovat Kuviossa 1 ja käytön luotettavuuteen liittyvät periaatteet Kuviossa 2. (Kaipio ja Vehmas, Physicians as usability evaluators – first aid for poor EHR usability? 2018)



Kuvio 1. Käytön sujuvuuteen liittyvät käytettävyysperiaatteet muunnetulla Nielsenin heuristiikalla. (Kaipio ja Vehmas, Physicians as usability evaluators – first aid for poor EHR usability? 2018)



Kuvio 2. Käytön luotettavuuteen liittyvät käytettävyyisperiaatteet muunnetulla Nielsenin heuristiikalla. (Kaipio ja Vehmas, Physicians as usability evaluators – first aid for poor EHR usability? 2018)

Heuristiikkaa voidaan käyttää erilaisten käytettävyyteen liittyvien ongelmien tunnistamiseen, lajitteluun sekä korjaamiseen.

## 5 HOITO-OHJEET.FI-PALVELUN KUVAUS

Hoito-ohjeet.fi palvelu on e-terveyspalvelu, joka on Varsinais-Suomen, Satakunnan ja Vaasan sairaanhoitopiirin yhteinen projekti. Palvelun tietosisällön ovat tuottaneet näissä kolmessa sairaanhoitopiirissä työskentelevät terveysalan ammattilaiset. Palvelun tietosisältö on avoimesti saatavilla ja yksi palvelun tavoitteista on tarjota luotettavaa, ammattilaisten tarjoamaa tietosisältöä sekä toisten ammattilaisten, että potilaiden saataville.

Tietosisältö palvelussa on kategorisoitu potilasontologian mukaan, jossa jokaiselle palvelussa saatavissa olevalle hoito-ohjeelle on asetettu ontologiasta löytyviä asiasanoja, joiden perusteella voidaan kategorisoida ja hakea hoito-ohjeita.

Palvelu on toiminnassa ja palvelun tietosisältöä voi hakea usealla eri tavalla, kuten aakkosellisesta listasta sekä käyttämällä hakuominaisuutta. (Danielsson-Ojala 2018)

### 5.1 Käyttäjryhmät sekä käyttöympäristöt

Palvelulla on kaksi merkittävää käyttäjäryhmää. Ensimmäisen käyttäjäryhmän muodostavat terveydenhuollon ammattilaiset, jotka vastaavat hoito-ohjeiden tuottamisesta palveluun sekä myös käyttävät muiden tuottamia hoito-ohjeita potilastyössä. Toinen palvelun käyttäjäryhmä ovat potilaat, jotka voivat tarvittaessa hakea palvelusta luotettavia, ammattilaisten laatimia hoito-ohjeita- (Danielsson-Ojala 2018)

Terveysalan ammattilaiset käyttävät palvelua sairaaloissa ja muissa hoitolaitoksissa, jossa usein käyttölaitteena on tietokone, kosketusnäyttöinen tabletti tai muu vastaava laite. Muut palvelunkäyttäjät käyttävät palvelua useissa eri käyttöympäristöissä, kuitenkin niin, että arvioidaan suuren osan palvelun käyttäjistä käyttävän palvelua mobiililaitteella. (Danielsson-Ojala 2018).

### 5.2 Kehitystavoitteet

Hoito-ohjeet.fi-palvelu on kehittyvä verkkopalvelu, joka pyrkii vastaamaan sekä terveydenhuollon ammattilaisten että asiakkaiden muuttuviin tarpeisiin. Projektin alkaessa palvelun kehityshaasteina koettiin tasalaatuinen sisällöntuottaminen sekä palvelun käytettävyyden kehittäminen ja palvelun käyttöasteen nostaminen. Uuden suunniteltavan ja

toteutettavan käyttöliittymäelementin toivotaan parantavan palvelun käytettävyyttä ja käyttäjäystävällisyyttä.

Terveysalan ammattilaiset, jotka käyttävät palvelua, kokevat puhtaasti tekstipohjaisten käyttöliittymien olevan haastavia käyttää kosketusnäytöllisillä päätelaitteilla. Samoin koetaan, että muiden palvelun käyttäjien voi olla haastava osata etsiä tarvitsemiaan hoito-ohjeita asiansanahauilla, sillä asiansanojen käyttö edellyttää, että käyttäjällä on tieto siitä, mitä asiansanoja hänen tulisi käyttää löytääkseen etsimänsä hoito-ohjeet.

Visuaalisen ja interaktiivisen käyttöliittymäelementin, jossa käyttäjä voi koskettamalla tiettyä kehonosaa, hakea kehonosaan liittyviä asiansanoja vastaavia hoito-ohjeita, katsotaan helpottavan yllä mainittuja käytettävyyteen liittyviä ongelmia.

Koska palvelu pystyy jo tarjoilemaan tietosisältöä useilla erilaisilla tavoilla, on kehitettävän uuden käyttöliittymäelementin tavoitteena tarjota vaihtoehtoinen hakutapa, joka soveltuu erityisesti käytettäväksi kannettavilla laitteilla sekä niille palvelun asiakkaille, joille tekstipohjaisten käyttöliittymien ja hakujen käyttö on haastavaa. (Danielsson-Ojala 2018)

## 6 IHMISKUVA-KÄYTTÖLIITTYMÄN SUUNNITTELU

Suunnitteluprosessi jaettiin kahteen erilliseen osaan, jossa kehitettävään ominaisuuteen soveltuvaa osaa ontologiasanastosta kehitti Päivi Lönnberg Tampereen ammattikorkeakoulun YAMK-tutkinnon päättötyönään. Päivi Lönnbergin työssä myös suoritettiin käytettävyystudkimusta sekä käytettävyydestausta eri suunnitteluvaiheissa olevilla käyttöliittymäelementin prototyypeillä. Toinen osa suunnitteluprosessia sisälsi varsinaisen kehitettävän ominaisuuden ulkonäön ja toiminnallisuuden suunnittelun, sekä käytettävyydestaustassa käytettyjen prototyyppien toteutuksen, joka on osa tätä opinnäytetyötä.

Suunnitteluprosessissa käyttöliittymän ulkoasua ja toiminnallisuutta kehitettiin Päivi Lönnbergin vertaisryhmille suorittamasta käytettävyydestaustasta saadun palautteen sekä opinnäytetyön ohjausryhmän antaman palautteen mukaisesti. Prosessin aikana luotiin useita prototyyppejä sekä ominaisuuden pääelementistä, eli ihmiskehon kuvasta, että muusta uuden hakuominaisuuden käyttöliittymästä.

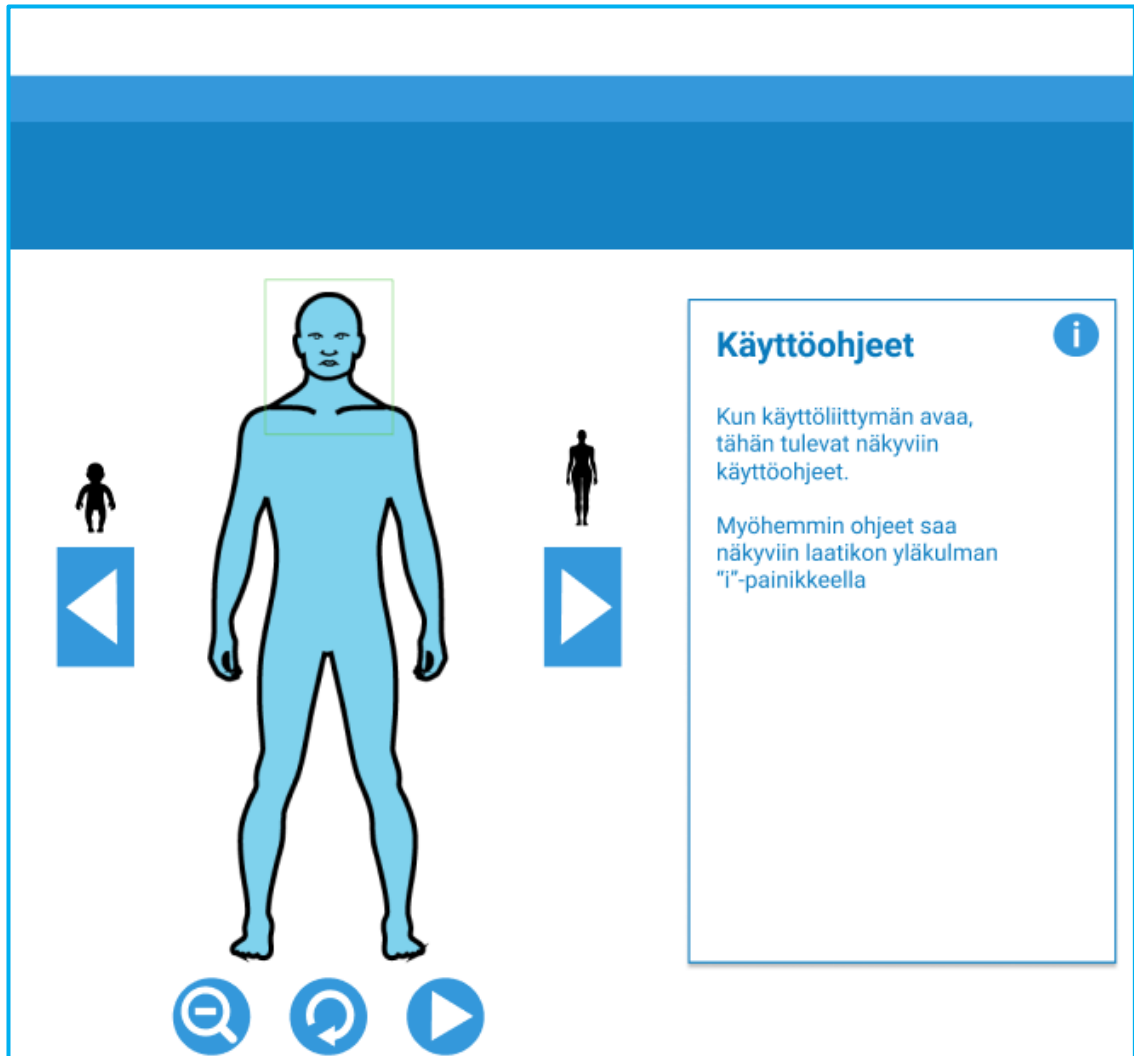
Prototyypeistä tehtiin osin funktionaalisia, jolloin ohjausryhmän sekä Päivi Lönnbergin testiryhmän oli mahdollista testata ja arvioida niiden varsinaista toiminnallisuutta.

Ohjausryhmän hyväksyttyä prototyypin, työssä siirryttiin käyttöliittymäelementin toteutusvaiheeseen, jossa tarkoituksena oli toteuttaa prototyyppiä vastaava käyttöliittymä niiltä osin, mikä työmäärältään vastaa tälle opinnäytetyölle varattua työmäärää.

### 6.1 Käyttöliittymän kuvaus

Toteutettava käyttöliittymä sisältää kolme pääasiallista elementtiä: Ihmiskehoa esittävä interaktiivinen kuva, jotka klikkaamalla tai koskettamalla voidaan hakea klikattua kehon osaa koskevia hoito-ohjeita; Sivu-valikko, jossa näytetään käyttäjälle tarvittavia käyttöliittymäelementin käyttöohjeita, ja johon ladataan haettavien hoito-ohjeiden tietosisältöä, sekä navigointielementti, joka sisältää painikkeet kehonkuvan vaihtamiseen, lähentämiseen, kääntämiseen sekä sivuvalikon avaamiseen ja sulkemiseen.

Kaikki kolme elementtiä haluttiin suunnitella niin, että ne ovat yhdenmukaisia ja yksiselitteisiä, sekä helppoja käyttää. Kuvassa 2 näkyvät käyttöliittymän kolme elementtiä niin, kuin ne ovat aseteltuna suurikokoisella näyttölaitteella.

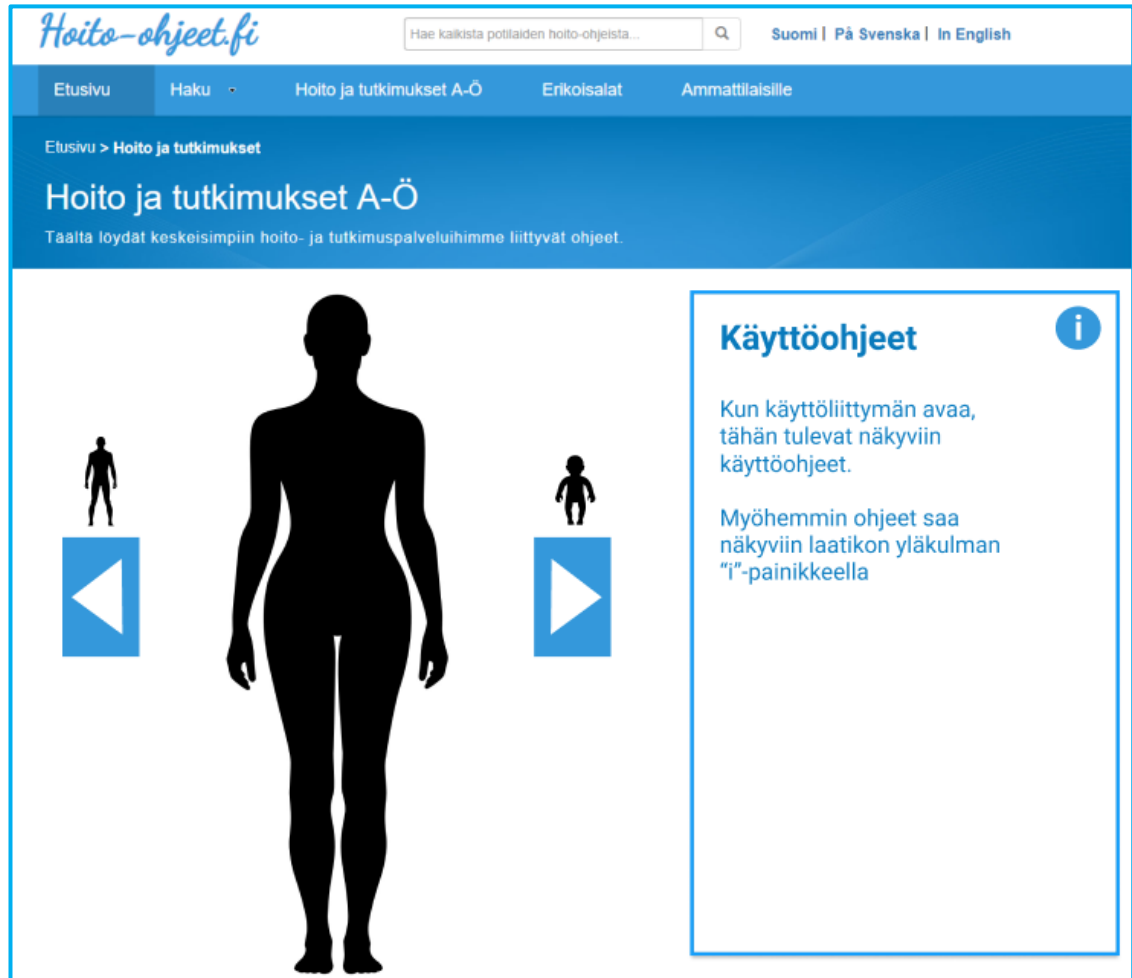


Kuva 2. Yksi uuden käyttöliittymäelementin funktionaalisen prototyypin työpöytänäkymistä.

## 6.2 Käytettävyys ja käyttäjäystävällisyys

Päivi Lönnbergin suorittamasta hakuelementtiin liittyvästä käytettävyyskyselystä nousi esiin useita kohtia, jotka otettiin huomioon käyttöliittymäelementin suunnittelussa ja toteutuksessa. Käytettävyyskysely tehtiin hakuelementin versiosta 1.0. joka on kuvassa 3. Käytettävyyskyselyn pohjalta ihmissiluetista muutettiin vaalea ja käyttöliittymään lisättiin tarvittavia toimintoja tarjoavia painikkeita. Muita käytettävyyskyselyssä nousseita asioita olivat esimerkiksi aktiivisen hakutuen toteuttaminen, sukupuolen huomioonottaminen,

mahdollisuus kuvan kääntämiseen sekä käyttöliittymän nopeuden ja helppokäyttöisyyden varmistaminen. Osa näistä ominaisuuksista oli jo versiossa 1.0. myös toteutettu (Lönnerberg 2019).

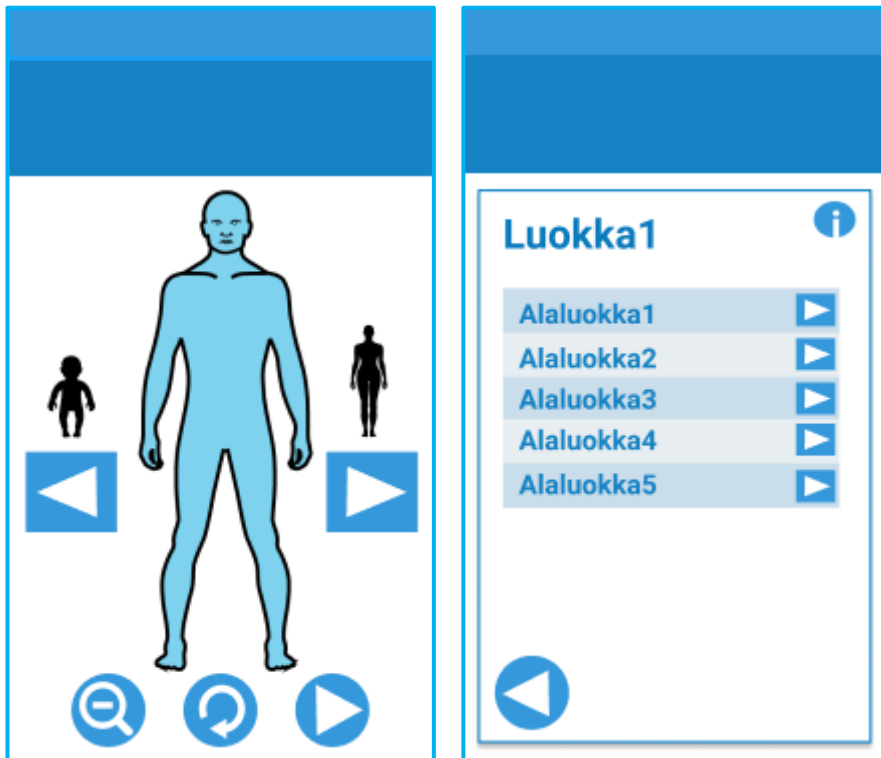


Kuva 3. Ihmiskuva käyttöliittymän varhaisen version työpöytänäkymä (versio 1.0).

Käyttöliittymä suunnitettiin ja toteutettiin noudattaen luvussa 3.1 kuvattua mobiililaitteiden periaatetta. Tämän periaatteen mukaan käyttöliittymä tulisi siis olla mahdollisimman helppo käyttää ensisijaisesti kannettavilla, pienikokoisilla laitteilla, joilla on kosketusnäyttö. Periaate ohjasi suunnitteluprosessia niin, että käyttöliittymä tulisi olla selkeä ja yksinkertainen ja käyttöliittymän painikkeet riittävän suurikokoisia, jotta niiden painaminen kosketusnäytöllä olisi helppoa.



Kuvassa 4 on mobiilikäyttöliittymän interaktiivinen prototyyppi kuvattuna kahdessa eri vaiheessa. Vasemmanpuoleisessa näkymässä käyttöliittymä näkyy niin, kuin käyttäjä näkee sen saapuessaan sivulle, jossa käyttöliittymäelementti sijaitsee. Oikeanpuoleisessa näkymässä käyttäjä on valinnut kehonkuvasta jonkun alueen, jolloin mobiilikäyttöliittymässä kehonkuvan päälle avataan navigointivalikko, jossa käyttäjä voi nähdä kehonosaan liittyviä tarkennushakuehdotuksia sekä siihen suoraan liittyviä hoito-ohjeita, mikäli näitä on saatavilla.



Kuva 4. Uuden käyttöliittymäelementin mobiilinäkymiä.

Lisäksi käyttöliittymän suunnittelussa otettiin huomioon muita käytettävyyteen liittyviä seikkoja, kuten navigoitavuus, joka näkyy suunnittelussa ja käyttöliittymän toiminnallisuudessa niin, että käyttöliittymän käyttöön vaadittavien interaktioiden määrä pyrittiin minimoimaan kaikissa käyttötapauksissa.

Käyttöliittymään liitettävään ihmiskehon kuvaan rajattuihin kehon osiin liitetään potilasontologian asiasanoja, joita käytetään myös muissa palvelun hakuominaisuuksissa sekä hoito-ohje-listauksissa. Yhdenmukaisen sanaston käyttö myös uudessa käyttöliittymäelementissä tukee palvelun käytettävyyden yhdenmukaisuutta.

Käytettävyyden tueksi käyttöliittymään lisättiin myös ominaisuus, jossa käyttäjän siirtäessä hiiren osoittimen jonkin kehonosan päälle kehonosa korostetaan ja käyttäjälle

näytetään pienikokoinen ohjeikkuna, jossa käyttäjälle kerrotaan kehonosaan liittyvät ontologiasanat. Lisäksi käyttöliittymän sivuikkunassa on ohjepainike, josta käyttäjä voi halutessaan klikkaamalla saada sivuikkunaan näkyviin käyttöliittymään liittyvän lyhyen käyttöohjeen.

Käyttäjän avatessa käyttöliittymän, käyttöohjeet ovat valmiiksi näkyvissä, jotta käyttäjä voisi nopeasti aloittaa käyttöliittymän käytön.

Käyttöliittymän painikkeisiin valittiin yleisesti tunnettavat symbolit kuvaamaan erilaisia käyttöliittymän toimintoja. Aiemmin esitetyissä käyttöliittymän prototyypin kuvissa olevat symbolit eivät olleet lopullisia valintoja. Lopullisista valitut painikkeet, niiden symbolit ja merkitys ovat kuvattuna kuviossa 3.



Kuvio 3. Kehonkuva käyttöliittymän toimintopainikkeet.

Painikkeilla, joilla hallinnoidaan sivu-valikkoa on merkittävämpi vaikutus mobiilikäyttöliittymässä, mutta käyttöliittymän yhdenmukaisen käytön varmistamiseksi painikkeet ovat käytettävissä samalla toiminnallisuudella myös työpöytäkäyttöliittymässä.

## 7 IHMISKUVA-KÄYTTÖLIITTYMÄN TEKNINEN TOTEUTUS

Tähän opinnäytetyöhön liittyvän työmäärän rajaamiseksi hyvin aikaisessa vaiheessa projektia päätettiin, että teknistä toteutusta ei tulla saattamaan tämän työn aikana täysin valmiiksi. Jatkokehityksen mahdollistaminen ja työn dokumentointi niin, että sitä on mahdollista jatkaa jonkun muun tahon toimesta myöhemmässä projektissa, otettiin osaksi kehitysprojektia. Samalla annetaan oikeus VSSH:lle tämän opinnäytetyön aikaansaannosten vapaaseen käyttöön kuitenkin niin, että alkuperäisen työn toteuttaja mainitaan raportoinnissa.

### 7.1 Teknisen toteutuksen vaatimukset

Käyttöliittymäelementistä pyrittiin kehittämään itsenäinen verkkosivumoduuli, joka olisi helppo liittää osaksi verkkopalvelua. Varsinainen verkkopalveluun liittäminen rajataan tämän työn ulkopuolelle. Vaatimus helposta palveluun liitettävyydestä asetti osaltaan vaatimuksia teknisen toteutuksen suunnittelulle sekä toteutukselle.

Toteutettavan käyttöliittymäelementin tärkein osa, eli ihmiskehoa esittävä navigointielementti haluttiin toteuttaa sivulle SVG-muotoisena vektorigrafiikkaelementtinä, jonka käyttö osaltaan takaa käyttöliittymäelementin toiminnan eri kokoisilla näyttölaitteilla.

Käyttöliittymäelementistä haluttiin lisäksi helposti ylläpidettävä niin, että ihmiskehon kuvaan asetettavia interaktiivisia alueita sekä niihin liittyviä potilasontologian sanaston osia voidaan lisätä, poistaa ja siirtää. Toiminnallisuutta siirtämiseen, poistamiseen tai lisäämiseen ei kuitenkaan itsessään toteuteta tämän työn aikana.

### 7.2 Toteutukseen käytetyt tekniikat

Jotta toteutettava käyttöliittymäelementti toteuttaisi ylempänä osiossa 7.1 mainittuja vaatimuksia, ihmiskuvaelementtiä käsitellään toteutuksessa SVG-formaatissa. Työssä käytetään lisäksi luvussa 2 mainittuja verkkosovelluksen kehittämiseen liittyviä perustekniikoita niin, että käyttöliittymäelementti pyritään kehittämään mahdollisimman responsiiviseksi, jotta se olisi helposti käytettävissä suurimmalla mahdollisella käyttäjäryhmällä.

Koska käyttöliittymäelementtiä voidaan käsitellä luvussa 3.1 määriteltynä asteittaisena parannuksena, voidaan elementin toteutuksessa kuitenkin käyttää sellaisia tekniikoita, jotka eivät toimi täysin optimaalisesti kaikissa käyttöympäristöissä. Toisaalta myös osan käyttöliittymäelementille asetetuista vaatimuksista täyttäminen edellyttää joidenkin sellaisten tekniikoiden käyttöä, joita esimerkiksi vanhemmat verkkoselaimet tukevat huonosti.

Jotta toteutus toteuttaisi vaatimusta helposta jatkokehitettävyydestä, toteutustekniikaksi valittiin TypeScript, joka luvussa 2.2 mainituilta osin parantaa selaimen kehitettävän koodin luettavuutta, vikasietoisuutta sekä jatkokehitettävyyttä. TypeScript-koodi pyrittiin kirjoittamaan niin, että sitä on helppo sekä dokumentoida, että kehittää edelleen myös tällä hetkellä tuntemattoman kolmannen osapuolen toimesta. Useista TypeScript alimoduuleista kuitenkin käännetään lopputuloksena yksi JavaScript-moduuli, joka on yksinkertaista liittää osaksi verkkopalvelua.

Modularisoidun lähdekoodin kääntäminen TypeScript:stä JavaScript:ksi suoritetaan käyttämällä TypeScript-kääntäjää sekä Webpack-ohjelmistoa. Webpack paketoii TypeScript-kääntäjän tuottamat JavaScript-tiedostot sekä tarvittavat ulkopuoliset kirjastot yhdeksi yksinkertaiseksi verkkomoduuliksi, joka voidaan ottaa käyttöön verkkopalvelussa.

Ylläpidettävyyden ja jatkokehitettävyyden kannalta koodin suunnittelu niin, että se on oikein modularisoitu ja tarjoaa hyvin suunniteltuja rajapintoja ja rajapintaluokkia oli erityisen tärkeää. Erityisesti kehonkuvaa koskevan ylläpidettävyyksivaatimuksen täyttämiseksi kehonkuvan navigointiin liittyvä toiminnallisuus erotettiin kuvaan liitettävistä alueista ja asiasanastosta niin, että alueita ja asiasanastoa on helppo muuttaa ilman että navigointi ja hakutoiminnallisuuksia tarvitsee muokata. Tätä erottelua pyrittiin toteuttamaan erottamalla kehonkuvan muodostama grafiikka ja siihen lisättävät interaktiiviset alueet toisistaan, jotta interaktiivisten alueiden muokkaaminen ei vaikuta kehonkuvan ulkonäköön.

Ohjelmakoodissa interaktiivisten alueiden sisältämää ryhmää alueita kutsutaan 'kontrollikerrokseksi', joka voidaan tarjota navigointia käsittelevälle alimoduulille rajapinnan kautta. Tämä mahdollistaa interaktiivisten alueiden muuttamisen mahdollisesti myös ohjelmakoodin suorituksen aikana.

### 7.3 Teknisen toteutuksen ongelmat ja niihin liittyvät ratkaisuehdotukset

Vaatimukset SVG-kuvaformaatin käytöstä ja ihmiskuvaan rajattavien alueiden helposta päivitettävyydestä ja ylläpidettävyydestä ovat osaltaan ristiriidassa keskenään. Koska SVG liitetään HTML DOM -puuhun, on jokaisen interaktiivisen alueen oltava SVG:ssä oma elementtinsä, jolle tulee määrittää koko ja muoto. Käytännössä alueiden lisääminen ja muokkaaminen vaatisi joko SVG-tiedoston muokkaamista tekstinmuokkaustyökalulla tai sen vektorigrafiikan muokkaamista kuvanmuokkaustyökaluilla, josta SVG-tiedosto on tallennettu.

SVG-tiedostojen muokkaaminen ohjelmallisesti on myös mahdollista. Tässä työssä ei ole kuitenkaan kartoitettu valmiiden ohjelmistojen saatavuutta. Jatkokehitysehdotuksina tähän ongelmaan ehdotan joko valmiin SVG-muokkaustyökalun käyttöönottoa tai käyttötapausten vaatimukset täyttävän muokkaustyökalun luomista erillisessä projektissa. Vaihtoehtoinen ratkaisu olisi vaihtaa SVG-kuvaformaatin sijaan käyttämään jotakin muuta kuvaformaattia sekä HTML5-canvas-elementtiä, johon alueiden määrittelemineen on merkittävästi yksinkertaisempaa. Kuvaformaatin ja lähestymistavan vaihtaminen kuitenkin aiheuttavat pakollisia muutoksia luotuun ohjelmamoduulin koodiin.

SVG-kuvaformaattilla interaktiivisena verkkoelementtinä on myös joitakin muita heikkouksia, joista tämän työn kannalta merkittävin on verkkoselainten vaihteleva tuki SVG:iden käytölle.

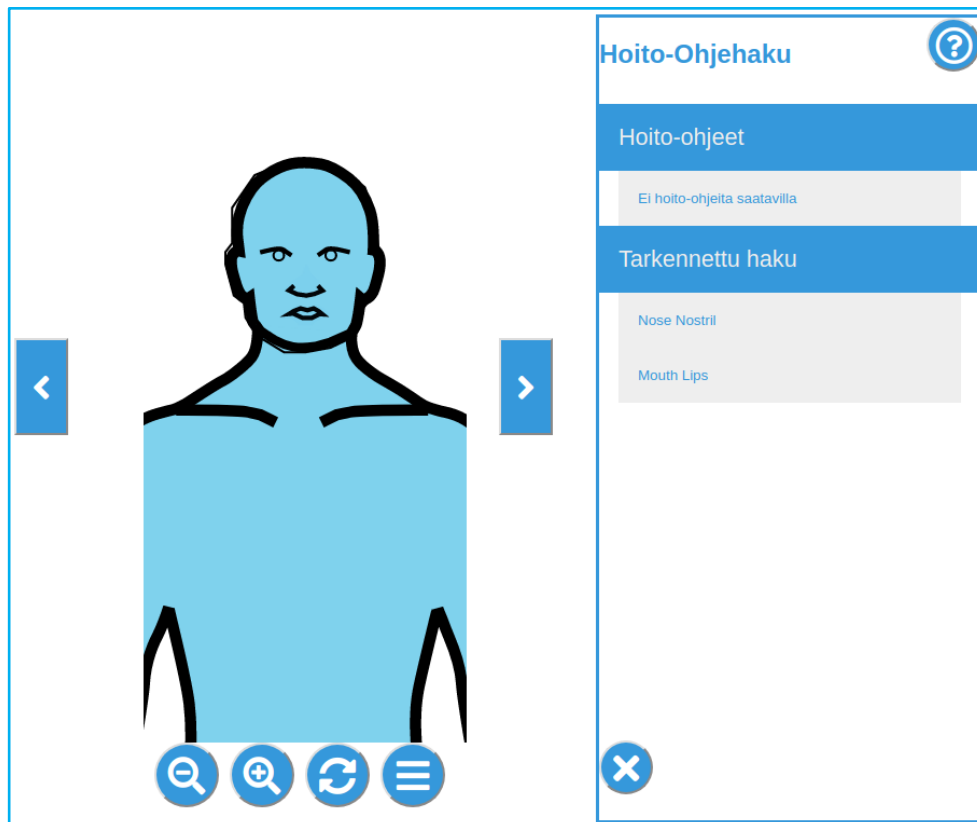
Osa verkkoselaimista asettaa rajoitteita SVG-elementtien interaktiivisille osille, niin että elementit, joilla ei ole täyttöä tai rajaa, eivät voi olla interaktiivisia. Nämä rajoitteet vaikeuttavat SVG-kuvan hyödyntämistä käyttöliittymäelementissä, sillä kehonosien rajauksia ei tahdota näkyviksi elementeiksi kuvaan, vaan SVG-kuvan halutaan olevan yksinkertainen ja puhdas ihmiskehon kuva. Interaktiivisten alueiden näkyvä rajaus tai kehystäminen voisi mahdollisesti tehdä kuvasta sekavan ja näin laskea käyttöliittymän käytettävyyttä.

Kehitystyössä yllä mainittu ongelma pystyttiin ratkaisemaan niin, että interaktiivisille alueille määriteltiin hyvin läpikuultava pohjakuvan värinen täyttö käyttäen RGBA väritystä. On kuitenkin oletettavaa, että sellaisilla selaimilla, mitkä eivät tuo läpikuultavia RGBA-väriarvoja, selain näyttää alueiden täytön ilman läpikuultoa, joka tekee käyttöliittymän käytöstä hyvin hankalaa.

Työssä ei laajamittaisesti kartoitettu eri selainten SVG-elementteihin tai RGBA-arvojen käyttöön liittyviä eroja.

#### 7.4 Tulosten arviointi

Tekninen toteutus vastasi hyvin aiemmin suunnitteluvaiheessa toteutettua funktionaalista prototyyppiä. Lisäksi teknisessä toteutuksessa otettiin huomioon suunnitteluvaiheessa saatu palaute. Kuvassa 5 on yksi lopullisen teknisen toteutuksen näkymistä. Sivuvälikon tekstit ja sisältö on luotu käyttöliittymämoduulin testausta varten eikä vastaa oikeaa moduulin sisältöä. Kuvan 5 navigointinäkymässä käyttäjä on klikannut ihmiskuvassa interaktiivista aluetta (pää), jolloin kuva on lähennetty alueelle ja sivuvälikkoon on tuotu nähtäville saatavilla olevat hoito-ohjeet sekä mahdollisuus tarkennetun haun suorittamiseen.



Kuva 5. Yksi teknisen toteutuksen näkymistä.

Lopullisen teknisen toteutuksen käytettävyyttä voidaan arvioida käyttäen luvussa 4.2 esiteltyä muunnettua Nielsenin heuristiikkaa. Taulukossa 1 käydään läpi miten toteutettu käyttöliittymäelementti vastaa heuristiikan eri kohtiin.

Heuristiikka	Miten käyttöliittymäelementti toteuttaa heuristiikan vaatimuksia
<b>1. Vain oleellisen näyttäminen</b>	Käyttöliittymän elementtien määrä on minimoitu ja jokaisella elementillä on tärkeä funktio käyttöliittymän käytössä
<b>2. Tutun termistön käyttö</b>	Käyttöliittymän käyttöön liittyvä sanasto on yleistä käyttöliittymien navigointiin liittyvää sanastoa.
<b>3. Muistikuorman minimointi</b>	Käyttöliittymäelementteihin lisätyt 'tooltip' tiedot, jotka saa näkyviin esimerkiksi pitämällä hiiren osoitinta jonkun käyttöliittymän osan päällä, avustavat käyttöliittymän käytön oppimisessa ja muistamisessa.
<b>4. Toimintojen yhdenmukaisuus</b>	Käyttöliittymässä käytetään samaa ontologiasanastoa kuin muuallakin palvelussa.
<b>5. Mahdollisuus oikopolkuihin</b>	Käyttöliittymän käytön yksinkertaisuuden takia oikopolkujen tarjoaminen ei ole oleellista.
<b>6. Mitä tapahtuu juuri nyt?</b>	Suurin osa käyttöliittymän toiminnallisuudesta tapahtuu hyvin lyhyellä vasteajalla, jolloin käyttäjä välittömästi näkee tekemänsä komennon tuloksen. Kuitenkin palvelimelta haettaessa hoito-ohjeita, näytetään käyttäjälle moduulin sivuvalikossa ilmoitus 'Haetaan hoito-ohjeita...', kunnes hoito-ohjeiden haku on kokonaan suoritettu loppuun.
<b>7. Selkeä tapa peruuttaa virheen sattuessa</b>	Käyttöliittymän sivuvalikko tarjoaa 'Takaisin'-painikkeen, jolla voidaan palata edelliseen näkymään tai tilanteeseen.
<b>8. Selkeät virheilmoitukset</b>	Mikäli käyttöliittymässä kohdataan toimintaan liittyvä virhe, näytetään käyttäjälle ruudulla virheilmoitus. Virheilmoituksen sisältöön ei kuitenkaan ole kattavasti otettu työssä kantaa.
<b>9. Virhetilanteiden välttäminen</b>	Käyttöliittymä on hyvin yksinkertainen ja virheet on minimoitu varmistamalla koodin toiminnan vakaus sekä se, että esimerkiksi toimintojen suoritusjärjestyksellä ei ole väliä käyttöliittymän toiminnan kannalta.

<p><b>10. Riittävät ja selkeät ohjeet</b></p>	<p>Käyttöliittymän käyttöön liittyvä ohje näkyy ruudulla käyttäjän avatessa käyttöliittymän. Ohjeen saa lisäksi takaisin näkyviin ohje-painikkeesta. Ohjeessa on yksinkertaisesti selitetty käyttöliittymän toiminnallisuus.</p>
---	--

Taulukko 1. Käyttöliittymäelementin käytettävyyden arviointi käyttäen muunnettua Nielsenin heuristiikkaa

Taulukon perusteella voidaan katsoa, että toteutettu käyttöliittymä on käytettävyydeltään hyvä. On kuitenkin otettava huomioon, että käyttöliittymä on hyvin visuaalinen ja käyttää suurta määrää JavaScript-tekniikkaa toimintaan. Nämä kaksi kohtaa osaltaan rajaavat käyttöliittymän käytettävyyttä ja esteettömyyttä, sillä visuaalisten käyttöliittymien käyttö voi olla haastavaa visuaalisesti rajoittuneille käyttäjille. Lisäksi JavaScript-koodia laajasti hyödyntävät käyttöliittymät toimivat heikosti tai rajoittuneesti joillakin vanhemmilla selaimilla ja päätelaitteilla. JavaScript:n käyttö tuo myös haasteita käyttöliittymän esteettömyyteen, luvussa 2 esitellyllä tavalla.

#### 7.4.1 Esteettömyys

Käyttöliittymän toteutuksessa on osin käytetty esteettömyyttä tukevia tekniikoita, kuten rooliattribuuttien asettamista HTML-elementeille, sekä ARIA-attribuuttien käyttöä niiltä osin miltä se on mielekästä käyttöliittymän käytön kannalta. Erityisesti aria-live-attribuutin käyttö käyttöliittymän sivuvalikon elementteihin, joissa on muuttuva sisältö, edistää käyttöliittymän esteettömyyttä. Kuvassa 6 näkyy osa käyttöliittymän prosessoidusta HTML DOM:sta, jossa näkyy sekä rooliattribuuttien, että ARIA määritysten käyttöä.

```

▼<div id="svg_side_info" class="grid-item layout_right" role="gridcell" style="grid-column: 4 / auto;">
  ▶<div id="svg_side_top" role="heading">_</div>
  ▼<div id="svg_side_middle" role="text" aria-live="polite">
    ▶<p id="svg_usage_instructions" style="display: none;">_</p>
    <button class="svg_accordion_menu svg_care_instructions active" role="button">Hoito-ohjeet</button>
    ▶<div class="svg_sidemenu_panel svg_care_instructions" id="sidemenu_care_instructions_panel" style="display: block;">_</div>
    <button class="svg_accordion_menu svg_care_instructions active" role="button">Tarkennettu haku</button>
    ▶<div class="svg_sidemenu_panel svg_care_instructions" id="sidemenu_search_panel" style="display: block;">_</div>
    </div>
  ▶<div id="svg_side_bottom">_</div>
</div>
▶<nav id="svg_navigation" class="grid-item layout_left" role="navigation grid">_</nav>

```

Kuva 6. Osa käyttöliittymän esiprosessoitua HTML-koodia, jossa näkyy sekä HTML role ja WAI-ARIA attribuuttien käyttöä.



Käyttöliittymässä käytettävässä SVG-elementissä on käytetty esteettömyyttä tukevia avusteita, kuten title- ja description-attribuuttien määrittämistä SVG elementille. SVG-elementit kuitenkin tarjoavat huonoa tukea esimerkiksi näppäimistön avulla navigoinnille. Lisäksi käyttöliittymässä käytetty SVG-kuva on rakenteeltaan monimutkainen ja ilman laajamittaista esteettömyystestausta on vaikea arvioida kuinka hyvin erilaiset ruudunlukulaitteet suoriutuvat sen prosessoimisesta.

Tämän takia interaktiivisen SVG-elementin käyttö esimerkiksi ruudunluku-laitteen kanssa voi olla hyvin haastavaa. Käyttöliittymän esteettömyyttä ei myöskään ole työn puitteissa testattu tai laajamittaisesti arvioitu. Käyttöliittymämodulin esteettömyyttä arviotaessa on kuitenkin huomioitava, että käyttöliittymä tarjoaa vaihtoehtoisen tavan hakea tietoa ja on erityisesti suunnattu käyttäjäryhmille, joille visuaalinen käyttöliittymä on helpompi tapa hakea tietoa kuin tekstipohjainen. Hoito-ohjeet.fi-palvelu tarjoaa lisäksi useita erilaisia tekstipohjaisia hakumahdollisuuksia, jotka voivat soveltua paremmin käytettäväksi ruudunlukulaitteen kanssa.

Tämä huomioiden käyttöliittymämodulin visuaalisen esteettömyyden tavoitteena oli tarjota käyttöliittymän toiminnasta riittävä määrä tietoa ruudunlukulaitteelle, jotta käyttäjälle selviää käyttöliittymän toiminnallisuus ja tarkoitus.

Mobiililaitteiden käyttö on käyttöliittymässä huomioitu niin, että käyttöliittymän elementeille on määritelty tapahtumakuuntelijoita, jotka voidaan laukaista myös kosketustapah- tumilla pelkän hiirenpainalluksen sijaan. Lisäksi käyttöliittymä on hyvin responsiivinen ja mukautuu käyttäjän päätelaitteen koon mukaan, kuitenkin säilyttäen saman toiminnalli- suuden.

Käyttöliittymän voidaan siis arvioida tarjoavan riittävä esteettömyys ja käytettävyys koh- dekäyttäjärühmälle, joka on esitelty luvussa 5.2.

#### 7.4.2 Ylläpidettävyys ja jatkokehitys

Ylläpidettävyys ja jatkokehityksen helppouden varmistaminen työssä pääasiallisesti to- teutettiin käyttämällä TypeScriptin tarjoamia tyyppi-, luokka- ja rajapintaluokka määritel- miä, sekä rakentamalla tuotettu ohjelmakoodi niin, että se on jaettu loogisiin kokonai- suuksin sekä alamoduuleihin.

Lisäksi käyttöliittymämoduuli tarjoaa rajapinnan, jonka kautta voidaan rekisteröidä SVG-navigointielementtiin kiinnitettäviä kontrollikerroksia sekä asettaa haluttuja asetuksia modulille. Näin pyritään siihen, että mahdolliset pienimuotoiset moduulin toimintaan liittyvät muutokset voidaan tehdä ilman, että olemassa olevaa ohjelmakomponenttien lähdekoodia tarvitsee muokata.

Lisäksi ylläpidon ja jatkokehityksen tueksi ohjelmakoodi kommentoitiin laajamittaisesti ja sen lisäksi laadittiin erillinen dokumentaatio, jossa ohjeistetaan moduulin käyttöä ja kirjataan ylös niitä osia, joita tulisi jatkokehityksessä edistää.

Ylläpidettävyyteen liittyviä tavoitteita ei kuitenkaan täysin saavutettu, johtuen SVG:n käsittelyyn liittyvistä ongelmista, jotka kuvataan tarkemmin luvussa 7.3.

Osia käyttöliittymämoduulin toteutuksesta tullaan suorittamaan myöhemmässä jatkokehitysprojektissa. Tällaisia osia ovat esimerkiksi kontrollikerrosten laatiminen kaikkiin SVG-elementteihin sekä moduulin liittäminen osaksi Hoito-ohjeet.fi-verkkopalvelua.

Lisäksi erityisesti SVG-elementin esteettömyyttä tulisi testata ja kehittää jatkoprojektin puitteissa.

## 8 LOPUKSI

Työssä kartoitettiin e-terveyspalveluiden käytettävyyden ja esteettömyyden ongelmia ja niihin liittyviä ratkaisuja sekä kehitettiin Hoito-ohjeet.fi palvelulle uusi visuaalinen käyttöliittymämoduli. Käyttöliittymämodulin tavoitteina olivat parempi käytettävyys mobiililaitteilla sekä helppo ylläpidettävyys ja jatkokehittävyys.

Työssä käy ilmi, että e-terveyspalveluiden käytettävyyden ongelmat eivät suuresti eroa tavallisista käytettävyyden ongelmista, mutta niiden seuraukset voivat olla merkittävän vakavia. Käytettävyyttä tulisi rakentaa järjestelmään aina suunnitteluvaiheesta lähtien ja loppukäyttäjien ja terveysalan ammattilaisten osallistuminen järjestelmän suunnitteluun ja testaamiseen on tärkeä osa helposti käytettävän ja käyttäjäystävällisen järjestelmän rakentamista. Hyvin suunniteltu ja käytettävä järjestelmä tukee terveysalan ammattilaisten työtä, tarjoaa potilaille tärkeää tietoa terveydestä ja auttaa vähentämään ammattilaisille aiheutuvaa stressiä.

Käyttöliittymämodulin toteutuksessa ei täysin saavutettu kaikkia asetettuja tavoitteita, johtuen joidenkin tavoitteiden ja vaatimusten ristiriitaisuudesta. Työssä toteutettu käyttöliittymäelementti kuitenkin vastaa kohderyhmän tarpeisiin ja on helposti käytettävä.

Graafiikkapohjaisen käyttöliittymäelementin käytön opettelu ja käyttö itsessään on yksinkertaista. Tutun ihmishahmon käyttäminen interaktiivisena navigointi elementtinä tekee palvelun käytöstä nopeaa ja helppoa, näin auttaen palvelua saavuttamaan suuremman käyttäjäryhmän.

## LÄHTEET

Connor, Joshua O. 2012. *Pro HTML 5 Accessibility*. 1st. Apress.

Danielsson-Ojala, Riitta. 2018. *Hoito-ohjeet.fi palvelun kuvaus ja jatkokehitystoiveet* (12 14).

experienceux. *What is usability testing?* experienceux. Accessed 08 14, 2019. <https://www.experienceux.co.uk/faqs/what-is-usability-testing/>.

Gao, Zheng, Christian Bird, ja Earl T. Barr. 2017. "To Type or Not to Type: Quantifying Detectable Bugs in JavaScript." *2017 IEEE/ACM 39th International Conference on Software Engineering (ICSE)*. Buenos-Aires: IEEE. 758-769. <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=7985711&isnumber=7985634>.

Gustafson, Aaron. 2011. *Adaptive Web Design*. 1st. Edited by Krista Stevens. Kelly McCarthy.

Häyrynen, Kristiina. 2018. "Sote kohtaa digitalisaation – haasteet ja mahdollisuudet." (Finnish Journal of EHealth and EWelfare) 10 (2-3).

Heponiemi, Tarja, Hannele Hyppönen, Tuulikki Vehko, Sari Kujala, Anna-Mari Aalto, Jukka Vänskä, ja Marko Elovainio. 2017. "Finnish physicians' stress related to information systems keeps increasing: a longitudinal three-wave survey study." *BMC Medical Informatics and Decision Making* (BMC Medical Informatics and Decision Making) 17. <https://bmcmmedinformdecismak.biomedcentral.com/articles/10.1186/s12911-017-0545-y>.

Hogan, Brian P. 2013. *HTML5 and CSS3 (2nd edition)*. The Pragmatic Programmers.

Interaction Design Foundation. "What is Usability?" *Interaction Design Foundation*. Interaction Design Foundation. Accessed 08 23, 2019. <https://www.interaction-design.org/literature/topics/usability>.

Interaction Design Foundation. "What is User Experience (UX) Design?" *Interaction Design Foundation*. Interaction Design Foundation. Accessed 08 23, 2019. <https://www.interaction-design.org/literature/topics/ux-design>.

- Kaipio, Johanna, ja Nico Vehmas. 2018. "Physicians as usability evaluators – first aid for poor EHR usability?" (Finnish Journal of EHealth and EWelfare) 10 (2-3).
- Kaipio, Johanna, Tinja Lääveri, Hannele Hyppönen, Suvi Vainiomäki, Jarmo Reponen, Andre Kushniruk, Elizabeth Borycki, ja Jukka Vänskä. 2017. "Usability problems do not heal by themselves: National survey on physicians' experiences with EHRs in Finland." *International Journal of Medical Informatics* (International Journal of Medical Informatics) 97: 266-281. <https://www.ncbi.nlm.nih.gov/pubmed/27919385>.
- Keith, Jeremy. 2018. *Resilient Web Design*. Keith, Jeremy,;
- Kushniruk, Andre. 2002. "Evaluation in the design of health information systems: application of approaches emerging from usability engineering." (Computers in Biology and Medicine) 32 (3).
- Lönnberg, Päivi. 2019. *Hoito-ohjeet.fi -sivuston sisällön ja käytettävyyden kehittäminen kehonkuva-hakutoiminnallisuudella*. Tampere.
- Microsoft. Accessed 07 23, 2019. <https://www.typescriptlang.org>.
- Miglliriosi, Heather. 2016. *Accessible SVGs*. Accessed 10 09, 2019. <https://css-tricks.com/accessible-svgs/>.
- Robbins, Jennifer Niederst. 2018. *Learning Web Design*. O'Reilly Media.
- Rosenlund, Milla, ja Ulla-Mari Kinnunen. 2018. "Ikäihmisten kokemukset terveydenhuollon sähköisten palvelujen käytöstä ja kokemusten hyödyntäminen palvelujen kehittämisessä – kuvaileva kirjallisuuskatsaus." (Finnish Journal of EHealth and EWelfare) 10 (2-3).
- The Usability People. *Nine types of usability problems with Electronic Health systems*. The Usability People. Accessed 8 8, 2019. <https://www.healthcareusability.com/article/nine-types-usability-problems-electronic-health-systems>.
- Tomaszewski, Jack. 2018. "Why TypeScript is the best way to write Front-end in 2019." *Medium*. 12 24. Accessed 07 23, 2019. <https://medium.com/@jtomaszewski/why-typescript-is-the-best-way-to-write-front-end-in-2019-feb855f9b164>.

- Vepsäläinen, Juho. 2018. *SurviveJs - What is Webpack?* Accessed 10 19, 2019. <https://survivejs.com/webpack/what-is-webpack/>.
- W3C. *Web Accessibility Initiative*. Accessed 10 19, 2019. <https://www.w3.org/WAI/>.
- Web Accessibility Initiative. 2017. *Accessible Rich Internet Applications (WAI-ARIA) 1.1*. Accessed 10 19, 2019. <https://www.w3.org/TR/wai-aria-1.1/>.
- Yen, Po-yin, ja Suzanne Bakken. 2011. "Review of health information technology usability study methodologies." *Journal of the American Medical Informatics Association (Journal of the American Medical Informatics Association)* 19 (3): 413-422. <https://academic.oup.com/jamia/article/19/3/413/2909188>.