

Tuomas Kaartoluoma

KÄYTTÄJÄKESKEINEN SUUNNITTELU RAKENNUSSUUNNITTELUOHJELMAN KEHITTÄMISESSÄ

TIIVISTELMÄ

Tuomas Kaartoluoma: Käyttäjäkeskeinen suunnittelu rakennussuunnitteluohjelman kehittämisessä
Pro gradu -tutkielma
Tampereen yliopisto
Master's Degree Programme in Human-Technology Interaction
Huhtikuu 2020

Käyttäjäkeskeinen suunnittelu on runsaasti tutkittu ja paranneltu konsepti. Sen on sanottu olevan avain tuotteen menestykseen käyttäjäkeskeisessä suunnittelussa. Sen katsotaan tuovan lisäarvoa organisaatioille sekä kehittäjille. Käyttäjäkeskeinen suunnittelu ei kuitenkaan ole yksiselitteistä ja väärin sovellettuna siitä voi koitua enemmän harmia kuin hyötyä. Tämän takia tässä tutkielmassa noudatetaan tarkasti käyttäjäkeskeisen suunnittelun periaatteita ja selvitetään sen hyödyllisyyttä.

Tässä tutkielmassa käytetään käyttäjäkeskeistä suunnittelua rakennussuunnitteluohjelman kehittämiseen. Kehitysprosessissa selvitetään ohjelman ongelmakohtia ja tarjotaan niihin parannuksia, joita käyttäjät pääsevät arvioimaan. Kehitysprosessin päätteeksi arvioidaan, miten käyttäjäkeskeinen suunnittelu suoriutui tästä ja miten sen hyödyntäminen vaikutti lopputulokseen.

Tutkimusongelmana on ohjelman käyttöliittymän kehittäminen, jotta siitä saadaan tehokkaampi käyttäjille, jotka ovat käyttäneet ohjelmaa jo useita vuosia. Tutkimusmenetelmänä käytetään toimintatutkimusta. Toimintatutkimus suoritetaan kolmella iteraatiokierroksella, jossa ensimmäisessä tutustutaan ohjelmaan sekä sen ongelmiin ja seuraavilla kierroksilla arvioidaan prototyyppiä.

Tutkimus oli onnistunut ja sen avulla saatiin tuotettua lisäarvoa käyttäjille. Lisäarvona käyttäjille saatiin suunniteltua ratkaisuja, jotka helpottavat sekä nopeuttavat heidän työskentelyään. Tärkeää oli, että käyttäjät pääsivät itse vaikuttamaan lopputulokseen.

Avainsanat: käyttäjäkeskeisyys, käyttäjäkeskeinen suunnittelu, käytettävyys, toimintatutkimus

Tämän julkaisun alkuperäisyys on tarkastettu Turnitin OriginalityCheck -ohjelmalla.

Sisältö

1. Johdanto	1
2. Käyttäjakeskeisyys	3
2.1. Käsitteitä	3
2.2. Käyttäjakeskeinen suunnittelu	4
2.3. Käyttäjakeskeisen suunnittelun periaatteet	8
2.4. Käyttäjakeskeinen suunnitteluprosessi	9
2.4.1. Suunnittelututkimus	10
2.4.2. Suunnittelu	11
2.4.3. Suunnittelun arviointi	12
2.5. Yhteenveto	13
3. Tutkimusmenetelmä	14
3.1. Toimintatutkimus	14
3.2. Kehityskohde: Wode	15
3.2.1. Sweco	15
3.2.2. Wode	15
4. Käytännön kehitysprosessi	17
4.1. Kehitysprosessin käynnistäminen	17
4.2. Tilannekartoitus	17
4.2.1. Tilannekartoituksen suunnittelu	17
4.2.2. Toteutus	21
4.2.3. Analyysi/tulokset	21
4.2.4. Johtopäätökset	25
4.3. Prototyypin kehittäminen	26
4.4. Prototyypin testaaminen	27
4.4.1. Suunnitelma	28
4.4.2. Toteutus	30
4.4.3. Analyysi/tulokset	30
4.4.4. Johtopäätökset	32
4.5. Prototyypin jatkokehittäminen	32
4.6. Prototyypin testaaminen	33
4.6.1. Suunnitelma	33
4.6.2. Toteutus	42
4.6.3. Analyysi/tulokset	43
4.6.4. Johtopäätökset	45
5. Tutkimuksen johtopäätökset ja keskustelua	46
6. Yhteenveto	48
Lähteet	49
Liitteet	

1. Johdanto

Maon ja muiden [2005] mukaan käyttäjakeskeinen suunnittelu on monitieteinen suunnittelutapa, joka perustuu käyttäjien aktiiviseen osallistamiseen parantaakseen käyttäjän ja heidän tehtäviensä vaatimusten ymmärtämistä sekä suunnittelun ja sen arvioinnin iterointia. Tässä tutkielmassa käyttäjakeskeiseen suunnitteluun keskitytään laittamalla käyttäjät, heidän tarpeensa ja vaatimuksensa suunnitteluprosessin keskelle. Termi *käyttäjakeskeinen suunnittelu* sai alkunsa Donald Normanin tutkimuslaboratoriosta Kalifornian yliopistossa 1980-luvulla, mutta käyttäjakeskeisen suunnittelun katsotaan lähteneen liikkeen Normanin ja Draperin seminaarityöstä vuonna 1986. Käyttäjakeskeiselle suunnittelulle on annettu useita määritelmiä ja suunnitteluperiaatteita vuosien varrella. Kaikki suunnitteluperiaatteet ovat kuitenkin hyvin samankaltaisia eikä niissä ole juuri eroa kuin termistö. Tässä tutkielmassa keskitytään noudattamaan standardin ISO 9241-210 [2019] määrittelemiä käyttäjakeskeisen suunnittelun periaatteita.

Tässä tutkielmassa käydään läpi kehitysprosessi käyttäen käyttäjakeskeisen suunnittelun periaatteita ja suunnitteluprosessia. Tutkielman kohteena on rakennussuunnitteluohjelma rakennus-, energia- ja ympäristöalan asiantuntijayrityksessä, jossa sitä käytetään kattoristikoiden suunnitteluun. Kehitysprosessin tavoitteena on selvittää, miten käyttäjakeskeinen suunnittelu toteutetaan käytännössä, ja miten käyttäjakeskeisyyden korostaminen vaikuttaa lopputulokseen. Tätä lähdetään selvittämään toteuttamalla toimintatutkimus, joka noudattaa käyttäjakeskeisen suunnittelun periaatteita ja suunnitteluprosessia.

Toimintatutkimus tarjoaa systemaattiseen yhteistyöhön perustuvan lähestymistavan. Sen tarkoituksena on tehdä tutkimusta ihmisten kanssa, jotka kohtaavat todellisia ongelmia jokapäiväisessä elämässään. Toimintatutkimus keskittyy siis hyvin kontekstuaalisiin, paikallisiin ratkaisuihin. [Hayes 2011] Sen tarkoituksena on ratkaista käytännön ongelmia tutkimustiedon avulla. Tutkimus alkaa ongelmasta tai kehityksen alla olevasta kohteesta, jonka ratkaisemiseen tarvitaan tutkimustietoa. Toimintatutkimuksen keskeisiä piirteitä ovat ongelmakeskeisyys ja käytännönläheisyys. [Tiainen *et al.* 2015] Iteratiivinen sykli tai spiraali on yksi toimintatutkimuksen olennaisia ominaisuuksia [List 2006]. Valitsin toimintatutkimuksen tutkimusmetodikseni sillä se sopii hyvin yhteen käyttäjakeskeisen suunnittelun kanssa. Molemmissa iteroidaan prosessia, että saadaan haluttu lopputulos. Käyttäjakeskeinen suunnittelu tullaan toteuttamaan toimintatutkimuksen tavoin ja noudattamalla ISO 9241-210 [2019] standardin mukaisia suunnittelun periaatteita.

Tutkimuksessa tullaan tekemään kolme iteraatiokierrosta. Ensimmäisellä kierroksella kartoitetaan nykyistä ohjelmaa, sen käyttäjiä, mahdollisia puutteita ja etuja sekä koi-tetaan selvittää, millaisia muutoksia halutaan tulevaan ohjelmaan. Toiselle kierrokselle

rakennetaan prototyyppi ensimmäisen kierroksen tulosten perusteella. Kolmannella kierroksella on paranneltu versio prototyypistä ja sitä arvioidaan. Kolmannen kierroksen jälkeen tehdään lopullinen analysointi ja kirjoitetaan tulokset puhtaaksi.

Luvussa 2 käydään tutkielman teoreettista taustaa läpi, painopisteenä on käyttäjakeskeisyys ja käyttäjakeskeinen suunnittelu. Aluksi käydään läpi käsitteiden määritelmiä, jotta ne ymmärretään oikein, jonka jälkeen siirrytään käyttäjakeskeisen suunnittelun pariin, avataan sen käsitettä, suunnitteluperiaatteita ja suunnitteluprosessia.

Luvussa 3 käydään läpi tutkielman tutkimusmenetelmä eli toimintatutkimus. Samassa luvussa esitellään kohdeorganisaatio ja heidän rakennussuunnitteluohjelmansa.

Luvussa 4 on kuvattu varsinainen kehitysprosessi. Kehitysprosessi kuvataan iteraatiokierrosten mukaan järjestyksessä. Ensimmäisenä on ohjelman tilanteen kartoitus, jonka jälkeen on kaksi kierrosta prototyypin kehittämistä ja arviointia. Kehitysprosessi koostuu haastatteluista, analyyseistä sekä prototyypin suunnittelusta ja arvioinnista.

Luvussa 5 on tutkimuksen johtopäätökset ja keskustelua. Siinä käydään läpi, miten tutkimus onnistui, miten käyttäjakeskeinen suunnittelu sopi tähän tutkimukseen ja ehdotetaan jatkotoimenpiteitä. Luku sisältää myös tutkimuksen lopputulokset.

Luvussa 6 on vielä yhteenveto tutkielman sisällöstä.

2. Käyttäjakeskeisyys

Tässä luvussa esitellään käyttäjakeskeisen suunnittelun keskeisiä termejä ja syvennyttään paremmin siihen, mitä käyttäjakeskeinen suunnittelu pitää sisällään. Käyttäjakeskeistä suunnittelua käydään aluksi yleisesti läpi ja sen määritelmään. Seuraavaksi käydään käyttäjakeskeisen suunnittelun periaatteita ja lopuksi esitellään käyttäjakeskeinen suunnitteluprosessi.

2.1. Käsitteitä

Käyttäjakeskeinen suunnittelu (eng. user-centred design) on lähestymistapa järjestelmien suunnitteluun ja kehittämiseen. Sen tavoitteena on tehdä interaktiivisista järjestelmistä käyttökelpoisempia keskittymällä järjestelmän käyttöön ja käyttäjiin sekä heidän ergonomiaansa. Käyttäjakeskeisessä suunnittelussa hyödynnetään myös osaamista käytettävyydestä ja tekniikoista. Käyttökelpoisista järjestelmistä voi olla useita etuja, kuten parempi tuottavuus, parempi käyttäjien hyvinvointi, stressin välttäminen, parempi käyttömahdollisuus ja vähentynyt vahinkojen vaara. [ISO 9241-210:2019] Käyttäjakeskeistä suunnittelua ja sen eri vaiheita käydään vielä tarkemmin läpi kappaleissa 2.2, 2.3 ja 2.4.

Käyttäjä (eng. user) on henkilö, joka on vuorovaikutuksessa järjestelmän, tuotteen tai palvelun kanssa. Järjestelmän, tuotteen tai palvelun käyttäjät sisältävät henkilöt, jotka käyttävät järjestelmää, käyttävät järjestelmän tuotosta/tuotetta ja henkilöt, jotka tukevat järjestelmää (mukaan lukien henkilöt, jotka ylläpitävät järjestelmää ja kouluttavat sen käyttöön). [ISO 9241-11:2018] Käyttäjakeskeisessä suunnittelussa käyttäjällä tarkoitetaan henkilöitä, jotka tulevat käyttämään suunnittelun kohteena olevaa järjestelmää. Tässä tutkielmassa käyttäjät ovat Swecon työntekijöitä, jotka tulevat käyttämään suunniteltua ohjelmaa.

Käyttöliittymä (eng. user interface) sisältää kaikki vuorovaikutteisen järjestelmän komponentit (ohjelmisto tai laitteisto), jotka antavat tietoa ja hallintalaitteita käyttäjälle, jotta hän voi suoriutua tietyistä tehtävistä interaktiivisen järjestelmän kanssa. [ISO 9241-110:2006]

Vuorovaikutteinen järjestelmä (eng. interactive system) on yhdistelmä laitteistoa ja/tai ohjelmistoja ja/tai palveluita ja/tai ihmisiä, joiden kanssa käyttäjät ovat vuorovaikutuksessa tiettyjen tavoitteiden saavuttamiseksi. Tähän sisältyy tarvittaessa pakkaus, käyttäjän dokumentaatio, verkko ja ihmisen apu, tuki ja koulutus. [ISO 9241-11:2018]

Käytettävyys (eng. usability) voidaan määritellä mittana, minkä mukaan voidaan arvioida kuinka järjestelmän käyttäjät suoriutuvat erilaisissa käyttötilanteissa saavuttaakseen halutun tavoitteen tuloksia tuottavasti, tehokkaasti ja tyytyväisenä. Määritetyt käyttäjät, tavoitteet ja käyttökonteksti viittaavat käyttäjien, tavoitteiden ja käyttöympäristön yhdistelmään. Termiä *käytettävyys* käytetään myös tarkentajana viittaamaan suunnittelu-

tietoon, osaamiseen, toimintoihin ja suunnittelun ominaisuuksiin, jotka edistävät käytettävyyttä, kuten käytettävyyssiantuntemus, käytettävyyssammattilainen, käytettävyystekniikka, käytettävyyssmenetelmä, käytettävyyden arviointi ja käytettävyyden heuristiikka. [ISO 9241-11:2018]

Nielsenin [1994] mukaan käytettävyys ei ole vain yksi käyttöliittymän ominaisuus. Käytettävyydellä on monia osatekijöitä ja yleisesti se jaetaan seuraaviin viiteen käytettävyyden ominaisuuteen:

- Opittavuus: Järjestelmän tulisi olla helposti opittava, jotta käyttäjä voi nopeasti aloittaa työskentelyn.
- Tehokkuus: Järjestelmän pitäisi olla tehokas käyttää, kun käyttäjä on oppinut järjestelmän käytön hän voi työskennellä tuottavasti.
- Muistettavuus: Järjestelmä tulisi olla helposti muistettava, jotta satunnainen käyttäjä pystyy palaamaan järjestelmän käyttöön, vaikka viime käyttökerrasta olisi kulunut jo aikaa, ilman että täytyy opetella kaikkea alusta.
- Virheet: Järjestelmällä tulisi olla pieni virhetaso, jotta käyttäjät tekisivät mahdollisimman vähän virheitä käytön aikana, ja jos he tekevät virheitä, heidän pitäisi pystyä toipumaan niistä helposti.
- Tyytyväisyys: Järjestelmän tulisi olla miellyttävä käyttää. [Nielsen 1994]

Käyttäjäkokemus (eng. user experience) on käyttäjän havaintoja ja vastatoimia, jotka muodostuvat järjestelmän, tuotteen tai palvelun käytöstä ja/tai ennakoidusta käytöstä. Käyttäjän havainnot ja vastatoimet sisältävät käyttäjän tunteet, uskomukset, mieltymykset, käsitykset, mukavuuden, käyttäytymisen ja saavutukset, jotka tapahtuvat ennen käyttöä, käytön aikana ja sen jälkeen. Käyttäjäkokemus on seurausta brändikuvasta, esityksestä, toiminnallisuudesta, järjestelmän suorituskyvystä, interaktiivisuudesta ja järjestelmän, tuotteen tai palvelun apuominaisuuksista. Se muodostuu myös käyttäjän sisäisestä ja fyysisestä tilasta, joka on muodostunut aiemmista kokemuksista, asenteista, taidoista, kyvyistä ja persoonallisuudesta sekä käyttökontekstista. [ISO 9241-11:2018]

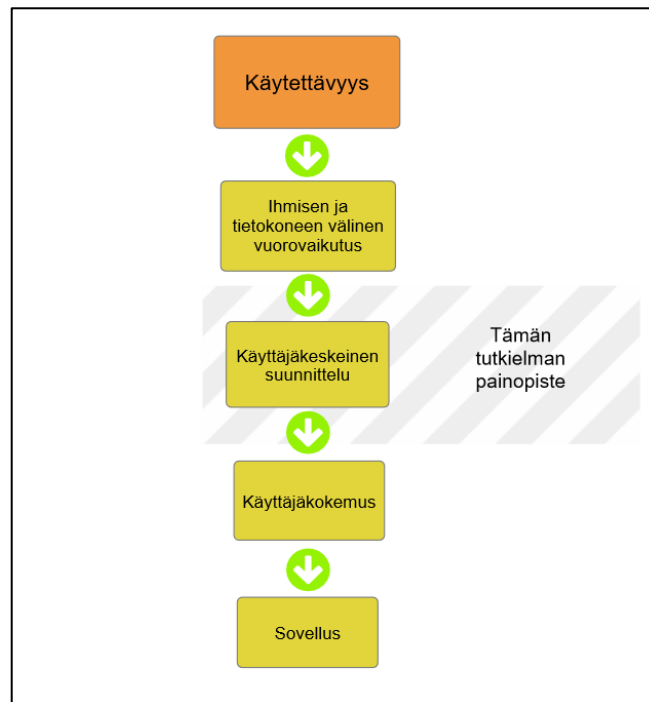
2.2. Käyttäjäkeskeinen suunnittelu

Termi *käyttäjäkeskeinen suunnittelu* sai alkunsa Donald Normanin tutkimuslaboratoriosta Kalifornian yliopistossa 1980-luvulla. Termin käyttö laajeni, kun Norman ja Draper [1986] julkaisivat kirjan: *User-Centered System Design: New Perspectives on Human-Computer Interaction*. Maon ja muiden [2005] mukaan käyttäjäkeskeinen suunnittelu on monitieteinen suunnittelutapa, joka perustuu käyttäjien aktiiviseen osallistumiseen parantaakseen käyttäjän ja heidän tehtäviensä vaatimusten ymmärtämistä sekä suunnittelun ja arvioinnin iterointia. Sitä pidetään avaimena tuotteen hyödyllisyyteen ja käytettävyyteen sekä tehokkaaseen tapaan ylittää perinteisen järjestelmäkeskeisen suunnittelun rajoitukset [Mao *et al.* 2005]. Gulliksen ja muiden [2005] mukaan käyttäjäkeskeinen suunnittelu on

prosessi, joka keskittyy käytettävyyteen koko kehitysprosessin ajan ja edelleen koko järjestelmän elinkaaren ajan. Normanin ja Jurekin [2018] mukaan käyttäjäkeskeinen suunnittelu on lähestymistapa paljon käytettyjen järjestelmien suunnittelulle löytämällä ja ymmärtämällä käyttäjien tarpeet heti projektin alussa.

Lowdermilkin [2013] mielestä monet kehittäjät pitävät käytettävyyden harjoittamista subjektiivisena. Nielsen [1994] väitti, että monet kehittäjät eivät käytä käytettävyyssuunnittelua, koska he pitivät sitä pelottavana sen monimutkaisuuden vuoksi, liian aikaa vievänä ja liian kalliina toteuttaa. Kehittäjät uskovat, että päätökset erilaisista käytettävyyden menetelmistä ovat vain mielivaltaisia ja he voivat tehdä omat päätöksensä omien mieltymyksiensä mukaan. Lisäksi monet näistä päätöksistä tehdään ilman mitään perusteita käyttäjän näkökulmaan. Ymmärtääksemme, miksi käyttäjäkeskeinen suunnittelu on tärkeää, meidän täytyy tietää mitä se on ja mitä se ei ole. [Lowdermilk 2013]

Käyttäjäkeskeinen suunnittelu ei ole käytettävyyttä. Käytettävyys on tutkimus siitä, kuinka ihmiset suhtautuvat mihinkin tuotteeseen. Ihmisen ja tietokoneen vuorovaikutus (human-computer interaction, HCI) on juurtunut käytettävyyteen, mutta se keskittyy siihen, kuinka ihmiset suhtautuvat tietojenkäsittelytuotteisiin. Käyttäjäkeskeinen suunnittelu on syntynyt ihmisen ja tietokoneen vuorovaikutuksesta. Se on ohjelmistojen suunnittelumenetelmä kehittäjille ja suunnittelijoille. Periaatteessa se auttaa heitä luomaan sovelluksia, jotka vastaavat käyttäjän tarpeita. [Lowdermilk 2013] Kuva 1 auttaa selkeyttämään näiden suhdetta toisiinsa.



Kuva 1. Suhde käytettävyyden, ihmisen ja tietokoneen vuorovaikutuksen, käyttäjäkeskeisen suunnittelun ja käyttäjäkokemuksen välillä.

Voidaan sanoa, että harjoittamalla käyttäjäkeskeistä suunnittelua saadaan taattua hyvä käytettävyys ohjelmistossa ja se on se tärkein asia. Ottamalla käyttäjät kehittämisprosessin keskiöön saadaan poistettua epäselvyydet ja pystytään keskittymään käyttäjän tarpeisiin. [Lowdermilk 2013] On paljon todennäköisempää, että järjestelmällä on korkea käytettävyys, jos ymmärretään käyttäjiä ja suunnitellaan heidän tarpeidensa mukaan [Norman ja Jurek 2018].

Lisäksi on vielä käyttäjäkokemus. Käyttäjäkokemusta käytetään yleensä kuvaamaan koko kokemusta ohjelmistosta. Se ei kata pelkästään toiminnallisuutta vaan myös sen kuinka houkuttelevaa ja mukavaa sovellusta on käyttää. Käyttäjäkeskeinen suunnittelu voidaan toteuttaa sen varmistamiseksi, että sovelluksen käyttö takaa hyvän käyttökokemuksen. [Lowdermilk 2013] McNamara ja Kirakowski [2006] kirjoittivat, että käyttäjäkokemus ottaa huomioon tuotteen ja käyttäjän välisen laajemman suhteen tutkiakseen henkilön henkilökohtaista kokemusta siitä.

Käyttäjäkeskeinen suunnittelu ei ole subjektiivista. Koko käytettävyyden ala ja kaikki sen taustalla olevat metodologiat tulevat monilta tieteenaloilta. Ergonomian, psykologian, antropologian ja monien muiden alojen soveltamisen kautta käytettävyys juurtuu tieteelliseen tietoon. Se on kaukana subjektiivisesta ajattelusta tai arvailusta. Käyttäjäkeskeisen suunnittelun prosessi toimii subjektiivisia oletuksia vastaan. Se vaatii todisteita siitä, että suunnittelun ideat ovat tehokkaita. Kaikki suunnittelupäätökset, jotka on tehty tarkkailemalla tai kuuntelemalla käyttäjää eivät perustu mielijohteisiin tai henkilökohtaisiin mielityksiin. Käyttäjäkeskeisen suunnitteluprosessin aikana kerättyä tietoa vastaan on vaikea väittää, etteikö se kehittäisi ohjelmistoa. [Lowdermilk 2013] Käyttäjäkeskeinen suunnitteluprosessi toimii subjektiivisia oletuksia vastaan. Tarvitaan todisteita, että voidaan todeta suunnitteluratkaisujen olevan tehokkaita. Jos käyttäjäkeskeinen suunnittelu tehdään oikein, sovelluksesta tulee käyttäjiä viehättävä. Tällöin jokainen suunnitteluratkaisu on tehty kuuntelemalla käyttäjiä, eivätkä ratkaisut perustuneet omiin henkilökohtaisiin mielityksiin. [Ferrández-Pasto *et al.* 2017]

Käyttäjäkeskeinen suunnittelu ei ole pelkästään sommittelua. Tämä on ehkä yleisin väärinymmärrys. Jotkut uskovat, että käyttäjäkeskeiset suunnittelun harjoittajat keskittyvät vain estetiikkaan. Vaikka estetiikka voi olla tärkeää, se ei kata koko kuvaa. Käyttäjäkeskeisyys on enemmän kuin vain keskittymistä asioiden ulkonäköön tai näyttävien animaatioiden luomista. Käyttäjäkeskeinen suunnittelu takaa, että tutkimme, kuinka tehokas ohjelma on saavuttamaan suunnitellun tarkoituksen. Kuitenkin käytettävyydetutkimus voi tunnistaa ohjelman käyttöliittymän virheitä, jotka vaikeuttavat tehtävien suorittamista. Tässä tapauksessa ohjelman käyttöliittymällä on iso rooli tavoitteen saavuttamisessa, kuitenkin olisi virhe pitää sitä ainoana tekijänä. [Lowdermilk 2013] Vredenburg ja muut [2002] tutkivat käyttäjäkeskeisen suunnittelun vaikutuksia. Kyselyyn osallistui yli 100 käyttäjäkeskeisen suunnittelun harjoittajaa. Keskeisinä tuloksina löydettiin parannusta

tuotteen hyödyllisyydessä ja käytettävyydessä. Tämän tutkimuksen perusteella voidaan todeta, että käyttäjäkeskeisellä suunnittelulla on muitakin vaikutuksia kuin tuotteen ulkonäön parannukset.

Käyttäjakeskeinen suunnittelu ei ole virheraportti. Käyttäjakeskeisen suunnittelun harjoittaminen on enemmän kuin vain mahdollisuuden antamista käyttäjille raportoida virheistä ja ongelmista ohjelmassa. Vaikka käyttäjiä kannattaa kuunnella niin se ei tarkoita, että käyttäjätutkimus on pelkästään virheiden raportoimista. Jos pelkästään korjaat ohjelmaa aina, kun saat ilmoituksen virheestä, etkä perehdy siihen mitä käyttäjä on yrittänyt saada aikaiseksi virheen kohdatessaan, et saa merkityksellistä tietoa siitä, kuinka voit parantaa ohjelmaa kokonaisuutena. Tutkimalla ja kyselemällä kysymyksiä, jotka eivät liity virheeseen saat paremman käsityksen siitä käyttävätkö käyttäjät ohjelmaa suunnitellulla tavalla. Ominaisuuksia voidaan toteuttaa uudelleen, jotta työnkulut olisivat selkeämpiä tai sitten käyttäjän kanssa voidaan keskustella ja tuottaa uusia ideoita siitä, kuinka tuottaa ohjelmalla enemmän arvoa. [Lowdermilk 2013]

Käyttäjakeskeinen suunnittelu ei ole häiriötekijä. Se auttaa meitä keskittymään käyttäjän ydintarpeisiin. Se varmistaa, että saamme ensin tärkeää tietoa ennen kuin lähdemme yrittämään saada ongelmaa sopimaan tekniikkaan. On olemassa todellisia teknisiä rajoituksia, jotka on käsiteltävä, mutta kehittäjät keskittyvät yleensä näihin ongelmiin ensin. Käyttäjakeskeinen suunnittelu auttaa meitä etenemään käyttäjäkeskeisistä vaatimuksista teknisiin ratkaisuihin. Se on tarkoituksenmukainen lähestymistapa, joka auttaa meitä suorittamaan tehtävät oikeassa järjestyksessä. Sen tarkoituksena on auttaa keskittymään tärkeisiin asioihin. Se varmistaa, että keskitymme oikeisiin asioihin: käyttäjien tarpeiden tyydyttämiseen oikeilla teknologisilla ratkaisuilla. [Lowdermilk 2013] Shah ja Robinson [2007] mainitsevat, että yleisesti ottaen käyttäjien sisällyttäminen tuotekehitykseen parantaa tuotteen suunnittelua, käyttöliittymää, toiminnallisuutta, käytettävyyttä ja laatua.

Käyttäjakeskeisessä suunnittelussa pidetään käyttäjät mukana koko ohjelman suunnittelun, toteutuksen ja testauksen ajan. Tällöin pystytään keskittymään tuotekehityksessä käyttäjien tarpeisiin ja tuotteen käytettävyyteen. Tuotteesta tulee käyttäjäystävällisempi, kun osataan huomioida käyttäjän odotukset tuotteelta ennen kuin sitä lähdetään kehittämään pidemmälle. [Baek *et al.* 2008]

Käyttäjakeskeinen suunnittelu on erittäin laaja käsite, jonka määrittely on vuosien varrella kehittynyt monien eri tahojen toimesta. Sen käytöstä on todistettavan paljon hyötyä tuotteen/järjestelmien kehityksessä tai parantamisessa. Vaikka käyttäjakeskeinen suunnittelu on tehokas ja hyödyllinen menetelmä, on hyvä huomioida, että sitä ei välttämättä voida hyödyntää kaikkiin projekteihin.

2.3. Käyttäjakeskeisen suunnittelun periaatteet

Norman jatkoi käyttäjakeskeisen suunnittelun kehittämistä kirjassaan: *The Psychology of Everyday Things* [Norman 1988]. Kirja myöhemmin vaihtoi nimeään sen tunnetumpaan muotoon: *The Design of Everyday Things* [Norman 1988]. Norman [1988] tarjoaa neljä olennaista ehdotusta kuvaamaan minkälaisia suunnittelun tulokset tulisi olla:

1. Aina pitäisi olla helposti pääteltävissä, mitkä toiminnot ovat mahdollisia käyttää milloin tahansa.
2. Tee asioista näkyviä, mukaan lukien järjestelmän käsitteellinen malli, vaihtoehtoiset toiminnot ja toimintojen tulokset.
3. Tee järjestelmän nykyisen tilan arviointi helpoksi.
4. Seuraa luonnollista kartoitusta aikomusten ja vaadittujen toimintojen välillä; toimintojen ja niistä aiheutuvien vaikutusten välillä; ja näkyvien tietojen ja järjestelmän tilan tulkinnan välillä.

Nämä suositukset sijoittavat käyttäjän keskelle suunnittelua. Suunnittelijan tehtävänä on helpottaa käyttäjän tehtävää ja varmistaa, että käyttäjä pystyy hyödyntämään tuotetta suunnitellulla tavalla ja minimoimalla vaivan oppia käyttämään sitä. Norman huomautti, että tuotteiden mukana tulevat pitkät, hankalasti ymmärrettävät käsikirjat eivät ole käyttäjakeskeisiä. Hän ehdottaa, että tuotteiden mukana tulisi pieni lehtinen, joka voitaisiin lukea nopeasti ja se hyödyntäisi käyttäjän tietämystä. [Abrams *et al.* 2004]

Norman ei jättänyt suunnittelijaa vain yllämainitun neljän ehdotuksen varaan. Norman [1988] ehdotti seitsemää suunnittelun periaatetta suunnittelijan tehtävän helpottamiseksi:

1. Käytä tietoa sekä maailmasta, että päästä. Rakentamalla käsitteellisiä malleja, kirjoita käsikirjoja, jotka ovat helposti ymmärrettäviä ja jotka on kirjoitettu ennen suunnittelun toteuttamista.
2. Yksinkertaista tehtävien rakenne. Varmista ettei käyttäjän lyhytaikaismuisti tai pitkäaikaismuisti ei ylikuormitu. Keskimäärin käyttäjä pystyy muistamaan viisi asiaa kerralla. Varmista, että tehtävä on johdonmukainen ja tarjoaa mentaalisia apuvälineitä tiedon noutamiseen pitkäaikaismuistista. Varmista, että käyttäjällä on kontrolli tehtävästä.
3. Tee kaikesta näkyvää. Yhdistä toteutuksen ja arvioinnin kuulut. Käyttäjän pitäisi pystyä selvittämään kohteen käyttötarkoitus katsomalla oikeita nappeja tai laitteita toiminnon suorittamiseksi.
4. Suunnittele esitykset oikein. Yksi tapa tehdä asioista ymmärrettäviä on käyttää grafiikkaa.
5. Hyödynnä sekä luonnollisten että keinotekkoisten rajoitusten voimaa antaaksesi käyttäjälle tunteen, että on vain yksi asia tehtävänä.

6. Suunnittele virheiden varalle. Valmistaudu kaikkien mahdollisten virheiden varalla, tällä tavalla käyttäjälle annetaan mahdollisuus palautua mahdollisista tehdyistä virheistä.
7. Kun kaikki muu epäonnistuu, standardisoi. Luo kansainvälinen standardi, jos jostain ei voi suunnitella ilman mielivaltaisia esityksiä.

Vuonna 1987 Ben Shneiderman esitti samankaltaiset suunnittelun periaatteet, mutta hän nimesi ne kahdeksaksi kultaiseksi säännöksi [Shneiderman 1987]. Vielä myöhemmin Jakob Nielsen sovitti nämä samat konseptit kehittääkseen heuristiikan käytettävyystekniikalle [Nielsen 1994]. Molemmat, Norman ja Shneiderman, ovat yrittäneet tiivistää hyvän suunnittelun muutamaa periaatteeseen. Molempien periaatteet ovat hyvin samankaltaisia, niissä löytyy suurimmat erot käytetyissä termeissä. Suunnittelija voi käyttää apunaan toista, molempia tai ei kumpaakaan niin halutessaan. Normanin työ korosti tarvetta tutkia käyttäjien tarpeita ja toiveita sekä tuotteen käyttötarkoitusta, tarve saada käyttäjät osallistumaan tuotteen kehitykseen sen tarkoitettussa ympäristössä oli luonnollinen kehitys käyttäjäkeskeisen suunnittelun alalla [Abrams *et al.* 2004].

Gulliksen ja muiden [2005] mielestä muiden kehittämät suunnittelun periaatteet eivät olleet tarpeeksi kattavia ja ne olivat liian abstrakteja, jotta niitä olisi helppo hyödyntää käytännössä. Tästä syystä he kehittivät omat 12 periaatetta käyttäen apuna aikaisempia töitä [Gulliksen *et al.* 2005]. Heidän muodostamansa periaatteet ovat hyvin samankaltaiset kuin standardin ISO 9241-210 [2019] periaatteet. ISO 9241-210 [2019] periaatteet ovat:

1. Suunnittelu perustuu käyttäjän, tehtävien ja ympäristöjen selkeään ymmärrykseen.
2. Käyttäjät ovat mukana suunnittelussa ja kehittämisessä.
3. Suunnittelua ohjaa ja tarkentaa käyttäjäkeskeinen arviointi.
4. Prosessi on toistuva.
5. Suunnittelu kattaa koko käyttökokemuksen.
6. Suunnittelutiimi sisältää monitieteisiä taitoja ja näkökulmia.

Molemmat, Gulliksen ja muut [2005] sekä ISO 9241-210 [2019], pitävät käyttäjän suunnittelun keskiössä. Heidän mielestään on myös tärkeää pitää käyttäjä mukana koko suunnitteluprosessin ajan ja prosessin on oltava iteratiivinen. Myös suunnittelutiimin monialaisuutta pidetään tärkeänä käyttäjäkeskeisessä suunnittelussa. Tässä tutkielmassa tullaan keskittymään standardin ISO 9241-210 [2019] määrittelemiin suunnitteluperiaatteisiin.

2.4. Käyttäjäkeskeinen suunnitteluprosessi

Käyttäjäkeskeinen suunnitteluprosessi käsittää kolme vaihetta: suunnittelututkimus, suunnittelu ja suunnittelun arviointi. Suunnittelututkimuksessa on tarkoitus saada selville

keitä käyttäjät ovat ja mitkä heidän tarpeensa ovat. Toisessa vaiheessa, suunnittelussa, suunnittelija lähtee suunnittelemaan löydöstensä perusteella (esimerkiksi lähdetään kehittämään käyttöliittymää). Kun malli on laadittu, se arvioidaan käyttäjien kanssa ja tehdään muutoksia tarpeiden mukaan. Nämä kolme vaihetta ovat vain ydintoiminnot käyttäjakeskeisessä suunnitteluprosessissa. [Williams 2009] Käyttäjakeskeisen suunnittelun tarkoituksena on asettaa käyttäjät suunnitteluprosessin keskipisteeseen, käyttäjä osallistuu järjestelmän suunnitteluun, suunnitelman toteutukseen ja testaukseen [Cagiltay *et al.* 2008].

2.4.1. Suunnittelututkimus

Suunnittelututkimuksen aikana on tarkoitus arvioida keitä käyttäjät ovat ja mitkä heidän tarpeensa ovat. Korkealla tasolla suunnittelututkimukseen sisältyy tyypillisesti suunnittelu, suorittaminen, analysointi ja tutkimusdatan raportointi, ja jokainen näistä tyypillisesti sisältää useita vaiheita [Williams 2009].

Suunnittelututkimuksen suunnittelu yleensä sisältää keskittymisen yritykseen tunnistukseen sen tavoitteet, rajoitteet ja oletukset. Kun tiedetään tavoitteet ja rajoitteet, suunnittelija voi tehdä tietoisia päätöksiä mitä tutkia ja mikä menettelytapa on paras. Ensimmäiseksi on tärkeää määrittää yrityksen sidosryhmät. Yleensä he ovat hankkeen liiketoiminnan asiakkaita ja muita henkilöitä yrityksessä, joilla on jokin osuus projektissa. Seuraavaksi on hyvä haastatella sidosryhmiä, että he voivat ilmaista tarpeensa. On tärkeää muistaa, että sidosryhmien tarpeet eivät ole käyttäjien tarpeita. [Williams 2009] Käyttäjakeskeisen suunnitteluprosessin tulisi omaksua laaja näkökulma eikä keskittyä pelkästään loppukäyttäjiin, jos halutaan tuottaa onnistunut uusi palvelu. Iteratiivinen kehitysprosessi on välttämätöntä aloittaa tunnistamalla merkittävimmät sidosryhmät ja niiden vaikutukset käyttäjakeskeisen suunnittelun soveltamiseen. Useiden sidosryhmien osallistuminen suunnitteluprosessiin vaatii huolellista projektisuunnittelua ja hallintaa, koska kukin sidosryhmä arvostelee konseptin, vaatimukset ja käyttökokemuksen eri näkökulmasta. [Alamäki ja Dirin 2015]

Suunnittelututkimuksen suorittaminen käsittää yleisesti neljä osa-aluetta. Ensimmäisenä tehdään taustatutkimus aiheesta. Jos suunnittelija ei tiedä etukäteen aiheesta, on hyvä tehdä taustatutkimusta ennen kuin aloittaa haastattelemaan käyttäjiä, jotta suunnittelija voi olla mahdollisimman valmistautunut. Toisena on hyvä tutkia kilpailevien yritysten työtä. Suunnittelija arvioi tai tarkastelee tutkimukseen liittyviä tai kilpailevia nettisivuja, sovelluksia ja niin edelleen. Kolmantena on haastattelut käyttäjien kanssa. Käyttäjähaastatteluita on monenlaisia, mutta keskityn kahteen tyyppiin, sillä niitä kahta käytän tutkielmassani, henkilöhaastattelut ja etähaastattelut. Vähemmän tunnettuja, mutta silti tehokkaita tekniikoita, ovat muun muassa päivä elämässä -tutkimus tai päiväkirjat. Yksi kaikista tehokkaimmista haastattelu tekniikoista on henkilöhaastattelu [Williams 2009].

Henkilöhaastatteluissa on vielä useita vaihtoehtoja. Suunnittelija voi seurata asiayhteyteen liittyvää lähestymistapaa, eli seurata käyttäjää normaalissa ympäristössä ja tarkkailla käyttäjää. Jos projektiin sisältyy uuden sovelluksen suunnittelua, suunnittelija varjostaisi käyttäjää ja tarkkailisi heidän tekemäänsä työtä kehitettävän sovelluksen valossa. Sen sijaan, että suorittaisi asiayhteyshaastattelun, suunnittelija voi suorittaa muodollisen haastattelun kasvotusten kysymällä sekoitusta suunnitelluista ja tutkivista kysymyksistä sekä suljetuista ja avoimista kysymyksistä käyttäjältä. [Williams 2009] Haastattelut käyttäjien kanssa voivat tarjota syvää, rikasta tietoa suunnitteluprosessille. Lisäksi käyttäjän mielipide, vaikkakin se olisi subjektiivinen, on tärkeä näkökohta suunnittelijalle. Avoimet kysymykset antavat mahdollisuuden tutkia käsillä olevaa aihetta, koska haastateltava voi vastata vapaammin ja haastattelija voi koettaa tiedustella aihetta enemmän saadakseen lisätietoja. [Ledbury 2018]

Etähaastattelut voidaan tehdä joko puhelimitse tai internetin välityksellä. Internetin välityksellä tehdyt haastattelut sisältävät jonkin sovelluksen, jota käyttävät sekä haastattelija, että haastateltava. Sen avulla pitäisi pystyä jakamaan näyttö ja mahdollisesti jotain toisia tapoja olla vuorovaikutuksessa. [Williams 2009] Etähaastattelujen etuna on, että niiden taltioiminen on helppoa. Haastattelujen taltioiminen on tärkeää varsinkin, puolistrukturoiduissa haastatteluissa, joita on paljon tässä tutkielmassa. Wilsonin [2014] mukaan puolistrukturoidut haastattelu hyötyvät ääni- ja videonauhoituksesta, se auttaa huomattavasti datan analysointivaiheessa.

Neljäntenä suunnittelututkimuksen toteutuksen osa-alueena on tutkimuksen toissijaiset muodot. Nämä sisältävät kyselylomakkeet, hakukoneoptimointi datan läpikäymistä (Search Engine Optimization, SEO), analytiikkadata, lokitiedostot ja tiedon keräämistä asiakastuesta, teknisestä tuesta ja/tai markkinoinnista. [Williams 2009]

Suunnittelututkimuksen analysointi voi tapahtua monin eri tavoin, se riippuu paljon suoritettujen tutkimusten tyypistä, kerätyistä tiedoista ja yleisesti projektin tavoitteesta. Käyttäjäkeskeinen tutkimus on yleensä laadullista, niin yleisiä analyttisiä tekniikoita ovat raportit (haastattelun tulokset), listat (esimerkiksi luettelo aikaisemmista oletuksista ja tärkeimmät havainnot) ja klusterointi (datan tarkistaminen ja yhtäläisyyksien yhdistäminen). [Williams 2009] Wilson [2014] suosittelee käytettävän datan analysointi työkaluja, jos haastattelut sisältävät paljon avoimia kysymyksiä ja haastateltavia on paljon. On myös hyvä huomioida, että usein nämä työkalut eivät ole helppoja oppia ja voivat viedä viikkoja tai jopa kuukausia hallita kunnolla [Wilson 2014].

2.4.2. Suunnittelu

Kun ongelma on perusteellisesti ymmärretty, suunnitteluprosessin seuraava vaihe on ideoida ratkaisuja. Voi olla hyödyllistä kehittää visuaalisia esityksiä, jotka osoittavat, miten käyttäjät ovat vuorovaikutuksessa tuotteen kanssa. Ideointi -vaihe voidaan myös toteuttaa

yhteistyössä käyttäjien tai sidosryhmien kanssa erilaisten työpajojen muodossa. Niissä laaditaan ratkaisuja, jotka auttaisivat parantamaan heidän ongelmiansa prototyypisuunnitteluun liittyvissä kysymyksissä. [Graham *et al.* 2019]

Suunnitteluvaihe sisältää aivoriisiä ja konseptointia ja ensimmäisten versioiden luonnostelua suunnittelututkimuksen perusteella. Käyttäjakeskeisen suunnittelun kirjallisuudessa on kirjoitettu vähän siitä, miten suunnittelijat tuovat taktisesti tutkimustulokset suunnitteluprosessiin. Kun luonnoksia on tehty yksi tai useampia, suunnittelija käyttää asiaankuuluvaa ohjelmistoa tuottamaan lisää versioita tuotteesta. Tuotteesta voidaan kehittää muun muassa sivukarttoja, rautalankamalleja tai prototyyppejä. [Williams 2009]

Yksi suunnittelun metodeista on kuvakäsikirjoitus. Kuvakäsikirjoitus tekee laajasta visiosta yksityiskohtaisen etenemismallin. Siinä näytetään kuva kovalta, miten käyttäjä etenee ja suorittaa tehtäviä uudessa järjestelmässä. Se auttaa suunnittelijaa miettimään, miten käyttäjä käytännössä tulee suorittamaan kaikki tehtävät. Useasti suunnittelijat lähtevät toteuttamaan suurta visiotaan keskittymättä pieniin vaiheisiin. Kuvakäsikirjoitus auttaa keskittymään niihin pieniin vaiheisiin, joita tarvitaan suoriutumaan isommista tehtävistä. Jokaista vaihetta ei ole pakko kuvittaa, on tärkeää, että ydintehtävät on selkeästi kuvitettu ja niihin liittyvät vaiheet. [Holtzblatt, Wendell ja Wood 2005] Tässä tutkielmassa on käytetty kuvakäsikirjoituksen mukaista prototyyppeä.

Prototyypit voivat vaihdella monimutkaisuuden ja täsmällisyyden tason mukaan, tarkoittaen kuinka tarkasti prototyyppi vastaa odotettua lopputuotetta. Prototyypit voivat olla monimuotoisia, mukaan lukien sellaisten välineiden käyttö, jotka eivät välttämättä liity lopputuotteeseen. Esimerkiksi muistiinpanot julistetaululla voivat olla arvokas prototyyppi, jos ne ilmaisevat selkeästi sovelluksen ominaisuudet. Hyödyllisintä on aloittaa yksinkertaisemmasta suunnittelusta, joka säästää aikaa ja rahaa. Minimalistiset prototyypit auttavat suunnittelijoita arvioimaan ideoita käyttämättä huomattavia resursseja ideoihin, jotka voidaan viime kädessä hylätä. Suunniteltaessa ensimmäisiä prototyyppejä voi olla hyödyllistä tehdä useampia vaihtoehtoja, joita käyttäjät voivat arvioida. [Graham *et al.* 2019]

2.4.3. Suunnittelun arviointi

Suunnittelun arviointi sisältää tyypillisesti sen käytettävyyden testaamisen. Käytettävyydellä itsellään on rikas historia ja runsaasti kirjallisuutta ja sitä pidetään omana aiheenaan. Muodollisen käytettävyydestauksen lisäksi on myös muita tapoja arvioida, muun muassa heuristiset tai asiantuntija-arvostelut, tyytyväisyys lomakkeet ja läpikäynnit. [Williams 2009]

Käyttäjäkeskeiselle suunnittelulle on ominaista monivaiheinen ongelmanratkaisuprosessi, joka vaatii suunnittelijoita:

- Analysoimaan ja ennakoimaan kuinka käyttäjät todennäköisesti käyttävät tuotetta;
- Testaamaan heidän oletustensa paikkansapitävyyttä reaali maailman testeissä todellisten käyttäjien kanssa;
- Testaamaan jokaisessa suunnittelu- ja tuotantoprosessin vaiheessa luomalla todistuspiiri, jolla vahvistetaan tai muutetaan alkuperäisiä vaatimuksia.

Toistaakseen prosessin, suunnittelijoiden on tiedettävä kuinka käyttäjät käyttävät tuotetta reaali maailmassa. Tätä varten kehitetään mahdollisimman todellisia testejä, joita käydään läpi käyttäjien kanssa ja näitä tuloksia hyödynnetään alkuperäisten suunnitteluvaatimusten vahvistamiseksi tai muuttamiseksi. [Grott 2019]

Kun prototyypit on suunniteltu, on kriittistä arvioida ne käyttäjien kanssa ja hienosäätää niitä tai esittää ne tulevaa käyttöä varten. Arvioivan tutkimuksen tarkoituksena on selvittää: onko suunnitelma käyttökelpoinen (esimerkiksi onko se helppo oppia ja käyttää); hyödyllinen (tukeeko se käyttäjiä tavoitteiden saavuttamisessa ja tehtävien suorittamisessa); ja toivottava (parantaako se käyttökokemusta). Tätä varten suunnittelu ja suunnittelun arviointi vaiheet ovat iteratiivisia, kun käyttäjät antavat palautetta alkuperäisistä suunnitelmista, prototyyppejä tarkistetaan, arvioidaan ja tarkistetaan uudelleen. [Graham *et al.* 2019]

2.5. Yhteenveto

Käyttäjäkeskeinen suunnittelu sai alkunsa 1980-luvulla ja se on laaja käsite, jonka määrittely on vuosien varrella kehittynyt monien eri tahojen toimesta. Sen käytöstä on todistettavan paljon hyötyä tuotteen/järjestelmien kehityksessä tai parantamisessa.

Käyttäjäkeskeiselle suunnittelulle on kehitetty useita eri periaatteita monien tahojen osalta. Useat periaatteet ovat hyvin samankaltaisia, niissä löytyy suurimmat erot käytetyissä termeissä. Suunnittelija voi käyttää apunaan yhtä, useita tai ei mitään niin halutessaan. Tässä tutkielmassa tullaan keskittymään standardin ISO 9241-210 [2019] määrittelemiin suunnitteluperiaatteisiin.

Käyttäjäkeskeisen suunnittelun tarkoituksena on asettaa käyttäjät suunnitteluprosessin keskipisteeseen. Käyttäjäkeskeisen suunnitteluprosessi kattaa kolme vaihetta: suunnittelututkimus, suunnittelu ja suunnittelun arviointi. Kaikkia vaiheita tullaan hyödyntämään tutkimuksen aikana ja niitä tullaan iteroimaan. Käyttäjät pidetään koko suunnitteluprosessin ajan tutkimuksen keskiössä.

3. Tutkimusmenetelmä

Tutkimusta ohjaa kaksi tutkimuskysymystä:

1. Miten käyttää käyttäjäkeskeistä suunnittelua parantamaan ohjelman tehokkuutta?
2. Miten käyttää käyttäjäkeskeistä suunnittelua kehittämään käyttöliittymää ohjelman tehokkaille käyttäjille?

Ensimmäistä kysymystä haluan tutkia, koska haluan selvittää, mitkä toiminnot ja osat alueet toimivat heidän nykyisessä ohjelmassansa ja mitä voisi kehittää. Toinen tutkimuskysymys on tärkeämpi, sen avulla haluan selvittää kuinka voin hyödyntää heiltä kerättyä tietoa, että pystyn suunnittelemaan käyttöliittymän vastaamaan heidän tarpeitaan. Vastauksia haetaan noudattamalla ISO 9241-210 [2019] standardin mukaisia suunnittelun periaatteita.

Aluksi minun täytyy tutustua ohjelmaan ja ohjelman käyttöön ja sitä kautta lähteä etsimään kehittämisen kohteita. Sweco ja Swecon työntekijät ovat tässä tärkeässä roolissa, ilman heitä käyttäjäkeskeistä suunnittelua ei voida toteuttaa, joten on tärkeää, että he ovat mukana kehitysprojektissa.

Tässä luvussa esitellään tutkimusmenetelmä sekä kehityksen kohteena oleva yritys ja heidän ohjelmansa.

3.1. Toimintatutkimus

Toimintatutkimus tarjoaa systemaattiseen yhteistyöhön perustuvan lähestymistavan. Toimintatutkimus on monitieteinen. Sen tarkoituksena on tehdä tutkimusta ihmisten kanssa, jotka kohtaavat todellisia ongelmia jokapäiväisessä elämässään. Toimintatutkimus keskittyy siis hyvin kontekstuaalisiin, paikallisiin ratkaisuihin. [Hayes 2011] Toimintatutkimus on laadullista tutkimusta, jonka ideana on käyttää tutkimustietoa kehittääkseen ja muuttaakseen käytännön toimia ja samaan aikaan tuottaa uutta tietoa tutkimukseen. Sen tarkoituksena on ratkaista käytännön ongelmia tutkimustiedon avulla. Tutkimus alkaa ongelmasta tai kehityksen alla olevasta kohteesta, jonka ratkaisemiseen tarvitaan tutkimustietoa. Toimintatutkimuksen keskeisiä piirteitä ovat ongelmakeskeisyys ja käytännönläheisyys. [Tiainen *et al.* 2015]

Iteratiivinen sykli tai spiraali on yksi toimintatutkimuksen olennaisia ominaisuuksia. Se käyttää iteraatiota kehittämään tutkittavaa asiaa kierros kerralla. [List 2006] Susmanin ja Everedin [1978] mukaan jokaisen syklin tulisi kattaa seuraavat vaiheet:

- Diagnostiikka. Ongelman tunnistaminen tai määrittäminen.
- Toimintasuunnittelu. Harkitaan vaihtoehtoisia toimintatapoja ongelman ratkaisemiseksi.
- Toimenpiteet. Valitaan toimintatapa.
- Arviointi. Tutkitaan toiminnan seuraukset.
- Oppimisen määrittäminen. Tunnistetaan yleiset havainnot.

Valitsin toimintatutkimuksen tutkimusmetodiksi sillä se sopii hyvin yhteen käyttäjakeskeisen suunnittelun kanssa. Molemmissa iteroidaan prosessia, että saadaan haluttu lopputulos. Käyttäjakeskeinen suunnittelu tullaan toteuttamaan toimintatutkimuksen tavoin ja noudattamalla ISO 9241-210 [2019] standardin mukaisia suunnittelun periaatteita.

Tutkimuksessa tullaan tekemään kolme iteraatiokierrosta. Ensimmäisellä kierroksella kartoitetaan nykyistä ohjelmaa, sen käyttäjiä, mahdollisia puutteita, etuja ja koitetaan selvittää, millaisia muutoksia halutaan tulevaan ohjelmaan. Toiselle kierrokselle rakennetaan prototyyppi ensimmäisen kierroksen tulosten perusteella. Kolmannella kierroksella on paranneltu versio prototyypistä ja sitä arvioidaan. Kolmannen kierroksen jälkeen tehdään lopullinen analysointi ja kirjoitetaan tulokset puhtaaksi.

3.2. Kehityskohde: Wode

Kohdeorganisaatio on yritys Sweco ja heidän ohjelmansa Wode. Tässä kappaleessa esitellään Sweco ja kehityskohteena oleva ohjelma Wode.

3.2.1. Sweco

Tutkimuksen kohdeorganisaationa toimii Sweco. Sweco on rakennus-, energia- ja ympäristöalan asiantuntijayritys, joka suunnittelee tulevaisuuden kaupunkeja ja kestävämpää yhteiskuntaa. Yrityksellä on 17 000 työntekijää, joista 2 500 työskentelee Suomessa. Sweco on Euroopan johtava suunnittelun ja konsultoinnin asiantuntijayritys, jonka liikevaihto on 1,9 miljardia euroa.

Kehitettävän ohjelmiston parissa työskentelee kuitenkin vain alle 20 työntekijää. He ovat työskennelleet Swecolla useita vuosia suunnittelun parissa. Swecolla jokaisella yksilöllä on merkittävä vaikutus lopputulokseen. Osaamisen kehittäminen on tärkeää, jotta he pystyvät vastaamaan asiakkaiden odotuksiin. Sen vuoksi jatkuvan kehittymisen ja tehävissä onnistumisen mahdollistaminen on tärkeää.

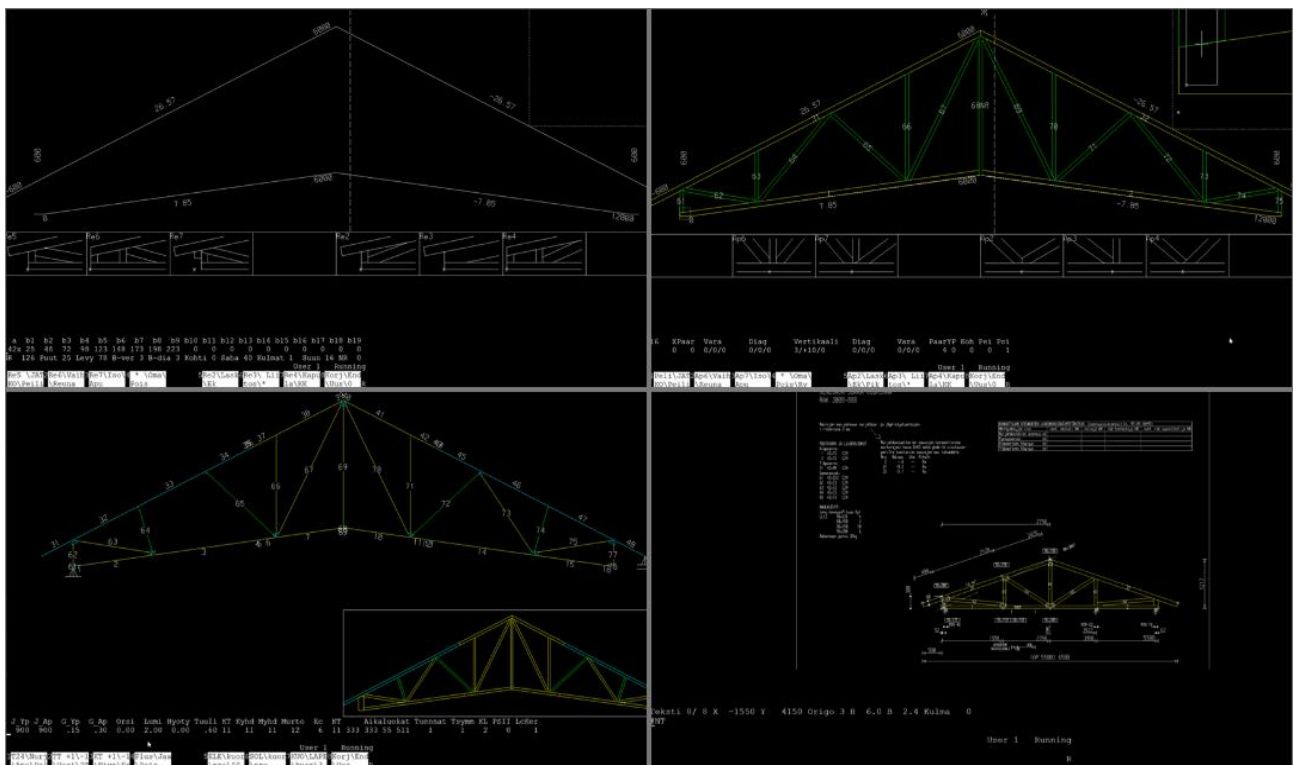
3.2.2. Wode

Swecolla on käytössään suunnitteluohjelma, jota kutsutaan nimellä Wode. Wode on Swecon kehittämä ohjelma. Se on tehty 1980-luvulla käyttäen ohjelmointikieltä BASIC. BASIC kehitettiin vuonna 1964 John Kemenyn toimesta ja se oli suuressa suosiossa 1980-luvulla [Lorenzo 2017]. BASIC on kehittynyt vuosien varrella useiden eri murteiden muodossa, mutta Wode -ohjelma ei ole pysynyt nykytrendien mukana. Woden käyttöliittymä näyttää samalta kuin silloin, kun se tehtiin 80-luvulla. Vaikka kehityskohteena on Wode niin alkuperäiseen ohjelmaan ei olla tulossa tekemään muutoksia, vaan ohjelmaa kirjoitetaan uudelleen C# ohjelmointikielelle ja muutokset tehdään siihen.

Kehityskohteena on suunnitella tehokas käyttöliittymä puisten naulalevyristikoiden suunnitteluun. Laajempi viitekehys on CAD (computer aided design), jossa keskeistä on suunnitella rakenteen geometria. Tavoitteena on tehdä ohjelma, joka on nopea käyttää, se on opastava ja johon on mahdollista liittää uusia käyttöä avustavia automaatiomenetelmiä.

Kehitettävällä ohjelmalla rakenteet suunnitellaan toistaiseksi vain yksi ristikko kerrollaan, jolloin suunnittelunäkymä on pääosin kaksiulotteinen. Ohjelmassa on valmiina tietokantaosuus, erillinen ratkaisin ja yksinkertainen käyttöliittymä. Ohjelmalla voidaan tällä hetkellä mallintaa ja laskea yksinkertainen ristikko. Ohjelmasta puuttuu tehokäyttöön soveltuva käyttöliittymä, joka kattaa koko ristikon geometrian suunnitteluprosessin. Tärkein yksittäinen toiminto on yhden liitoksen geometrian suunnittelu, jossa määritellään liittyvät osat, niiden sahaukset sekä naulalevyn koko ja sijainti.

Koska kokonaisen ohjelman käyttöliittymän suunnittelu on liian työläs projekti yhteen tutkielmaan, tässä tullaan keskittymään muutamiin tiettyihin ongelmiin ja siihen, miten ne voidaan ratkaista käyttäjakeskeisellä suunnittelulla. Aluksi tutustun Wode -ohjelmaan, että saan ymmärryksen ohjelman käytöstä, kuvassa 2 on muutama näyttökuvaa Wode -ohjelmasta.



Kuva 2. Wode -ohjelman käyttöliittymän kuvia.

4. Käytännön kehitysprosessi

Tässä luvussa käydään läpi, kuinka toimintatutkimus ja käyttäjäkeskeisen suunnittelu toteutettiin käytännössä. Tutkimuksen aikana suoritettiin kolme iteraatiokierrosta. Ensimmäisellä kierroksella kartoitettiin nykyistä ohjelmaa ja seuraavilla kahdella kierroksella arvioitiin ja kehitettiin prototyyppiä.

4.1. Kehitysprosessin käynnistäminen

Käyttöliittymän kehitysprosessi sai alkunsa, kun Swecon tutkimus- ja kehitysjohtaja otti yhteyttä Tampereen yliopistoon etsien tutkielman tekijää. Tapasin hänet 25.09.2019, jolloin kävimme ongelmaa läpi ja keskustelimme, minkälainen projekti olisi kyseessä. Tässä vaiheessa tutkimussuunnitelma ja -kysymykset olivat vielä avoimet. Tutkimussuunnitelmaa ja -kysymyksiä hiottiin vuoden 2019 loppuun. Oli tärkeää kohdentaa suunnitelma ja tutkimuskysymykset tarkaksi, ettei tutkielmasta koituisi liian isoa ja sitä ei kerittäisi aikataulun puitteissa saattaa loppuun. Tutkimuksen varsinainen suorittaminen alkoi tammikuussa 2020 ja loppuun saattaminen suunniteltiin huhtikuulle 2020. Tänä aikana tarkoitus oli suorittaa kolme iteraatiokierrosta toimintatutkimuksen tavoin.

4.2. Tilannekartoitus

Tilannekartoituksen ideana oli selvittää nykyisen ohjelman käyttötapoja ja tarkoitusta, mahdollisia käytön haasteita ja ongelmia sekä etsiä parannuskeinoja. Tilannekartoitus ohjelmalle tehtiin puolistrukturoidun haastattelun avulla. Puolistrukturoidussa haastattelussa esitetään kaikille haastateltaville samat tai melkein samat kysymykset samassa järjestyksessä [Saaranen-Kauppinen ja Puusniekka 2006]. Puolistrukturoitu haastattelu on osittain järjestelty ja osittain avoin haastattelu ja formaaliudessaan se sijoittuu täysin strukturoidun lomakehaastattelun ja teemahaastattelun välille [Hirsjärvi ja Hurme 2001]. Saaranen-Kauppisen ja Puusniekan [2006] mukaan puolistrukturoituja haastatteluja voi toisinaan kutsua myös teemahaastatteluiksi, varsinkin jos siinä esitetään tarkkoja kysymyksiä tietyistä teemoista, mutta ei välttämättä käytetä samoja kysymyksiä kaikkien haastateltavien kanssa. Haastattelukysymysten muodostamisessa käytettiin apuna käyttäjäkeskeisen suunnittelun periaatteita.

4.2.1. Tilannekartoituksen suunnittelu

Haastatteluiden toteuttaminen aloitettiin tammikuun puolessa välissä. Haastattelut pidettiin Swecon Tampereen toimistolla työhuoneissa. Haastatteluissa oli läsnä tutkija sekä haastateltava, sillä oli tärkeää taata haastatteluiden yksityisyys ja keskeytymättömyys. Ensimmäisiä haastateltavia pyydettiin esittelemään myös ohjelman käyttöä ennen kuin siirryttiin haastattelukysymyksiin. Koska minulla ei vielä ollut kovin hyvää käsitystä oh-

ohjelman käytöstä, oli tärkeää nähdä, miten ohjelmaa käytetään käytännössä. Ohjelman läpikäynnin jälkeen haastattelukysymyksiin tehtiin pieniä muutoksia. Ensimmäisiin haastatteluihin, jotka sisälsivät ohjelman läpikäynnin, varattiin aikaa tunti ja siitä seuraaviin haastatteluihin varattiin puoli tuntia.

Sovimme Swecon kanssa etukäteen, koska tulen toimistolle haastattelemaan. Sovimme sellaisen päivän, että siellä olisi varmasti väkeä paikalla. Haastateltaviksi valittiin ohjelman tämänhetkisiä käyttäjiä. Haastateltavien kanssa ei sovittu erikseen tiettyä ajan-kohtaa haastattelulle vaan katsottiin päivän mittaan kelle sopisi mikäkin hetki.

Haastattelukysymykset rakennettiin kolmen teeman ympärille. Teemat olivat:

- Taustatiedot
- Ohjelman käyttö
- Ohjelman arviointi

Taustatiedot ja ohjelman käyttö -teemat sisälsivät helppoja ja yksinkertaisia kysymyksiä ja ne auttoivat rentouttamaan haastateltavaa kyseiseen tilanteeseen. Kun haastateltavaa oli saatu ”lämmiteltyä” kysymyksiin siirryttiin ohjelman arviointiin ja hieman hankalampiin kysymyksiin. Haastattelurunko pysyi melkein samanlaisena kaikille haastateltavilla. Oli tiedossa, että kysymykset saattaisivat hieman muuttua ja/tai lisääntyä ohjelman läpikäyntien jälkeen. Seuraavaksi käydään läpi haastattelukysymyksiä ja perusteluita, miksi juuri nämä kysymykset valittiin tätä tutkimusta varten (katso Liite 1. Haastattelukysymykset).

Taustatiedot

1. Kuinka kauan olet työskennellyt tämän ohjelman parissa?
2. Arvio omasta tietoteknisestä osaamisestasi?

Näillä kysymyksillä selvitettiin, kuinka kokeneita käyttäjät olivat ja pyrittiin saamaan selkeyttä siihen, että millaisen tietoteknisen osaamisen ohjelman käyttäminen vaatii. Tietoteknistä osaamista arvioitiin asteikolla 1-5.

Ohjelman käyttö

1. Kuinka usein käytät ohjelmaa?
2. Paljonko käytät aikaa ohjelman parissa?
3. Kauanko kesti opetella käyttämään ohjelmaa?

Näillä kysymyksillä jatkettiin vielä haastateltavan rentouttamista tilanteeseen. Halusin myös selvittää, kuinka usein ja kuinka kauan he käyttävät ohjelmaa. Ohjelman opettelu-kesto oli myös tärkeää tietää, sillä siitä saa tietoa, kuinka helposti opittava se on uusille käyttäjille.

Ohjelman arviointi

1. Koetko ohjelman käytön helpoksi?
2. Millaisena koet käytön vaatiman aikamäärän? Onko käytön tehokkuus hyvä?

Näiden kysymysten avulla halusin arvioida, että onko ohjelma helppo käyttää vai onko ohjelma mahdollisesti kovin monimutkainen ja se vie turhaa aikaa tehokkuudesta. Käytön vaatimalla aikamäärällä halusin selvittää, kuinka tehokas ohjelma on eli tehdäänkö siinä niin sanottuja turhia vaihteita, mitkä voisi mahdollisesti hoitaa tehokkaamminkin.

3. Onko ohjelman käytön muistettavuus helppoa? Tarvitsetko tukea muistamiseen? Tarvitsetko lisää ohjeita muistamiseen?
4. Kuinka helposti ohjelmassa tapahtuu virheitä? Haluaisitko selkeämmät virheilmoitukset?

Kysymyksillä haluttiin selvittää, kuinka hyvin ohjelma tukee muistamista. Käyttäjän muisti on rajallinen, joten ohjelman tulee tukea muistamista. Tähän myös vaikuttaa, se kuinka kauan käyttäjät ovat työskennelleet ohjelman parissa. Muistaminen ja virheiden tekeminen kulkee paljolti käsi kädessä. Näillä kahdella kysymyksellä halusin tietää, kuinka paljon muistamista ohjelma vaatii ja tapahtuuko virheet niistä vai jostain muusta syystä. Virheilmoitusten selkeys auttaa myös selvittämään niiden lähteen ja estää tekemästä niitä jatkossa.

5. Onko käyttö miellyttävää?

Mitä miellyttävämpää ohjelman käyttö on, sitä tuottavampaa se on. Jos ohjelman käyttö on miellyttävää se nostaa työntekijöiden tehokkuutta. [Duncan 2013] Koska tutkimuksessa pyrittiin myös nostamaan tehokkuutta, oli tärkeä tietää, kuinka miellyttävää nykyisen ohjelman käyttö on ja pystyisikö sitä kehittämään.

6. Mistä tykkäät eniten ohjelmasta?
7. Mistä tykkäät vähiten ohjelmasta?

Näiden kysymysten avulla koitin kartoittaa, mistä voisi löytyä ongelmakohtia ja mihin osiin ohjelmassa ollaan tyytyväisiä. Nämä auttavat keskittämään huomioita oikeisiin ongelma-alueisiin ja jättämään toimivat osat ennalleen.

8. Onko ohjelmassa jotain toimintoa mitä käytät selkeästi eniten?

Tämän avulla halusin tietää, mihin suurin osa ohjelman käytön ajasta menee. Lisäksi halusin selvittää voisiko toiminnon käyttöä helpottaa tai nopeuttaa vai jätetäänkö se ennalleen.

9. Millaisia puutteita ohjelmassa on? Mitä haluaisit lisättävän?

Tämän kysymyksen taustalla oli selvittää, mitä puutteita käyttäjä näkee ohjelmassa. Tällä pystyy kartoittamaan ohjelman tilannetta, vaikka ei saisi käyttöliittymään kohdistuvia vastauksia. Jos haastateltavien oli vaikea keksiä tähän mitään vastauksia, siitä voidaan tehdä johtopäätös, että ohjelmassa on jo kaikki tarvittava, ja mahdolliset muutokset tulevat olemaan pelkästään hyvin esteettisiä.

10. Toimiiko ohjelman navigointi hyvin? Millaisia muutoksia se kaipaisi?

Tällä halusin selvittää, onko ohjelman navigoinnissa ongelmia. Tämä kysymys lisättiin ohjelman läpikäyntien jälkeen, koska ohjelman navigointi ei toimi jananan vaan siinä palataan aina etusivulle, josta siirrytään seuraavaan kohtaan. Halusin tietää näkevätkö käyttäjät tässä ongelmia tai haluavatko he siihen muutoksia.

11. Missä sinun mielestäsi ohjelman käyttöliittymän tulisi parantua?

12. Mitä muutoksia haluaisit käyttöliittymään?

Nämä kysymykset olivat hyvin samanlaisia, mutta kysymys kaksitoista kysyttiin vain, jos koin, että en saanut tarpeeksi tietoa kysymyksestä yksitoista. Kysymys kahdentoista kohdalla saatoinkin antaa esimerkkejä mahdollisille kehityskohteille, jos haastateltavalla oli vaikea miettiä asiaa. Näillä kysymyksillä halusin kartoittaa käyttäjien mieltymyksiä nykyiseen käyttöliittymään ja siihen, mitkä kohdat he näkevät ongelmallisiksi. Nämä kysymykset olivat yksiä tärkeimpiä tiedonlähteitä ensimmäisessä haastattelukierroksessa.

13. Käytätkö vanhaa ristikköä apuna, kun lähdet suunnittelemaan uutta?

Tämä kysymys lisättiin myös ohjelman läpikäyntien jälkeen. Huomasin, että monet ristiköt ovat hyvin samannäköisiä ja halusin tietää, onko heillä jotain hakemistoa tai mallipohjaa, joita he voivat hyödyntää, kun he aloittavat uuden ristikon suunnittelemisen.

14. Haluaisitko ohjelmaan sisäisen laskimen? Koetko sitä tarpeelliseksi?

Tämäkin kysymys muodostui ohjelman läpikäyntien jälkeen. Jokaisella suunnittelijalla on kiinteä taskulaskin työpisteellä, mitä he käyttävät paljon. Halusin selvittää kekevatko he ohjelmaan integroitua laskinta tarpeelliseksi vai ovatko he tyytyväisiä nykytilanteeseen.

4.2.2. Toteutus

Haastatteluiden toteutus tehtiin 14.01.2020 ja 22.01.2020. Haastateltavia oli yhteensä kuusi. Ensimmäisen kahden kanssa käytiin läpikäynti ohjelmasta sekä haastattelu ja seuraavien neljän kanssa käytiin vain haastattelu. Haastattelut pidettiin joko erillisessä työhuoneessa tai haastateltavan työpisteellä. Haastattelujen aikoja ei sovittu erikseen vaan ne työntekijät, jotka kerkesivät omilta töiltään haastateltaviksi, tulivat heille sopivassa kohdassa.

Ensimmäiset kaksi haastattelua tehtiin niin, että haastateltava oli tietokoneella, jossa oli Wode -ohjelma auki. He näyttivät minulle, miten ohjelmaa normaalisti käytetään ja pyysin myös näyttämään joitain erikoistilanteita. Läpikäynnin yhteydessä muodostin kysymyksiä tarpeen mukaan, että ymmärsin mahdollisimman hyvin, miten ohjelmaa käytetään, siihen ei ollut kuitenkaan valmiita kysymyksiä. Ohjelman läpikäynnin jälkeen kävimme vielä haastattelukysymykset läpi. Kaksi ensimmäistä haastattelua kestivät noin tunnin verran kumpikin. Ne nauhoitettiin käyttämällä ActivePresenter -ohjelmaa (<https://atomisystems.com/activepresenter/>), joka tallensi kaiken, mitä tapahtui näytöllä sekä äänen.

Seuraaviin neljään haastatteluun kysymyksiä muutettiin hieman. Muutokset tehtiin ensimmäisten haastattelujen ja ohjelman läpikäyntien perusteella. Näissä haastatteluissa käyttäjä ei ollut koneella, pelkästään tutkijalla oli kannettava tietokone, jossa oli kyselylomake. Näihin haastatteluihin oli varattu puoli tuntia jokaiseen, mutta haastattelut kestivät viidestätoista minuutista vajaaseen puoleen tuntiin. Haastatteluista tallennettiin vain ääni käyttämällä Windows 10:n Voice Recorder -ohjelmaa.

4.2.3. Analyysi/tulokset

Haastatteluista kirjoitettiin heti niiden jälkeen huomiot ylös. Myöhemmin analysointia jatkettiin litteroimalla haastattelut. Tuloksia lähdettiin purkamaan kysymys kerrallaan. Analyysin avulla pyrittiin selvittämään kuinka usein ja miten käyttäjät käyttävät ohjelmaa ja onko eri käyttäjien käyttötavoissa eroja. Vastauksista etsittiin yhtäläisyyksiä sekä eroavaisuuksia.

Ohjelman käytettävyyttä ja käyttöliittymää liittyvien kysymyksien analyysissä pidettiin apuna ISO 9241-210 [2019] standardin mukaisia suunnitteluperiaatteita. Analyysiä tehdessä pidettiin koko ajan mielessä toteutusmahdollisuudet tämän tutkielman puitteissa.

Vastauksista koitettiin paikantaa isoimmat ongelma-alueet etsimällä yhtäläisyyksiä kaikkien haastateltavien vastauksista. Haastatteluista poimittiin ongelmia ohjelmassa ja toiveita muutoksille.

Haastattelutulokset on purettu auki kysymys tai kysymysryhmä kerrallaan. Tuloksien esittelyssä pyritään pitämään haastateltavien anonymiteetti ja lainauksia ei ole otettu. Tuloksissa käydään kaikkien kysymysten vastaukset läpi, mutta eri kysymyksien vastauksia on yhdistetty vastausten ja kysymysten samankaltaisuuksien vuoksi.

Haastateltavat ovat työskennelleet ohjelman parissa jo monia vuosia. Keskiarvolta he ovat käyttäneet ohjelmaa 22 ja puoli vuotta. Haastateltavien useamman vuoden kokemuksen perusteella voi luottamuksella sanoa, että heillä on ohjelman käyttö hallussa. Heitä pyydettiin myös arvioimaan omaa tietoteknistä osaamista asteikolla 1-5 (1 = erittäin huono ... 5 = erittäin hyvä). Vastaukset vaihtelivat kahdesta viiteen, keskiarvolta 3,5. Vaikka ohjelman käyttö näyttää hyvin vaikealta alkuun niin se ei näytä vaativan kovin suurta tietoteknistä osaamista.

Suurin osa haastateltavista käyttää ohjelmaa päivittäin. Jos heillä ei ole muita työtehtäviä kuin suunnittelu niin työpäivästä kuluu noin 80-95% ohjelman parissa eli käytännössä koko päivä.

Ohjelman käytön opetteluun on kulunut huomattava aika. Nopein on oppinut käytön muutamassa kuukaudessa (n. 4-5kk), kun suurimmalla osalla on kulunut aikaa puolesta vuodesta kahteen vuoteen. Puolen vuoden jälkeen on osattu tehdä edes vähän jotain, mutta puolitoista vuotta on sellainen aika, että pystyy jo tekemään omasta mielestä hyvin. Yksi mainitsi, että ohjelman käytön opetteluun on mennyt lähemmäksi 5 vuotta, että pärjää täysin itsenäisesti sen kanssa.

Ohjelman käyttö koetaan hyvin helpoksi. Varsinkin, kun kaikilla on jo niin monen vuoden kokemus takana. Ohjelma on kuitenkin omanlaisensa maailma verrattuna nykypäivän ohjelmiin. Ohjelma ei taivu kaikkeen mihin sitä haluttaisiin ja se tuottaa välillä hankaluuksia. Siinä on paljon ominaisuuksia muistin varassa, kun siitä ei ole tehty manuaalia. Käyttäjät saattavat unohtaa ominaisuuksia, kun niitä ei ole käytetty pariin vuoteen. Ohjelmassa on paljon myös erikoistoimintoja, joista ei tiedetä. Kukaan ei muista kaikkea mitä ohjelma pitää sisällään. Uudelle työntekijälle ohjelman käyttö on varmasti hankalaa.

Käytön tehokkuutta pidetään todella hyvänä. Näppäinkomennot nopeuttavat ohjelman käyttöä ja sen suhteen siitä pidetään, varsinkin kun ne tulevat selkärangasta. Ohjelman tehokkuus on verrannollinen näppäintekniikan suhteen, mitä nopeammin se luonnistuu, niin sen tehokkaammin ohjelman käyttö onnistuu. Kuitenkin ohjelmassa on tiettyjä rajoitteita, mitkä aiheuttavat käytön tehokkuuden laskemista. Muun muassa laskentaelementtien määrää on rajoitettu ja saumoille on määritetty maksimirajat. Tehokkuus laskee myös, kun käyttäjän pitää alkaa itse siirtelemään pisteitä tai syöttää koordinaatteja. Nämä

ongelmat koskevat lähinnä erikoisempia rakennuksia, perustapauksissa ohjelma on äärimmäisen tehokas.

Käyttäjillä oli alkuun todella hankalaa käyttää ohjelmaa. Käyttäjillä oli vaikeuksia muistaa kaikkea, sillä ohjelman sisäiset ohjeet ovat todella huonot. Funktionäppäimet, mitkä ovat suurimmassa roolissa, sisältävät vain yhden kirjaimen selitteen mitä niistä tapahtuu. Nyt kuitenkin heillä on ohjelman käyttö selkärangassa eivätkä he tarvitse ohjeistusta. Kuitenkin välillä he joutuvat kysymään apua jossain toiminnossa, mitä he eivät ole joutuneet käyttämään hetkeen. Varsinkin tulevia työntekijöitä varten ohjelma kaipaisi selkeämmät käyttöohjeet, mutta nykyiset työntekijät eivät niitä tarvitse. Heillä on harvoin vaikeuksia ohjelman kanssa ja silloinkin he voivat tukeutua toisiin työntekijöihin.

Suurin osa ohjelmassa tapahtuvista virheistä tulee käyttäjältä. Ohjelma on luotettava siinä suhteessa. Ohjelma tukee hyvin ongelmatilanteiden ratkaisemisessa. Mutta virheilmoituksia tulee paljon niin ne saattavat hukkuu ohjelman sekaan. Välillä virheiden korjaaminen on hankalaa, sillä ohjelmassa ei ole peruutus -näppäintä, mutta ohjelma antaa kyllä palata taaksepäin korjaamaan. Käyttäjien mielestä osa virheilmoituksista saisi olla selkeämpiä, varsinkin jos on jokin todella vakava virhe niin virheilmoitus saisi olla korostetumpi. Osassa virheilmoituksissa on vain numerosarja ja se ei kerro mitään, saisi olla selkeämpi selite mistä virhe johtuu. Myös kaivattaisiin ilmoitusta, jos ollaan tallentamassa vanhan työn päälle. Ohjelma ei tallenna automaattisesti vaan se pitää käydä itse tekemässä. Välillä käyttäjä joutuu tekemään paljon välitalennuksia, mikä on hieman aikaa vievää ja täysin käyttäjän muistin varassa. Ohjelman virheilmoituksiin on totuttu ja ne osataan etsiä, mutta voisivat olla selkeämpiä ja välillä korostetumpia.

Ohjelman käytössä on miellyttävää se, että ei tarvitse käyttää hiirtä juuri ollenkaan. Muuten ohjelman miellyttävyys jakoi mielipiteitä, osa tykkää sen käyttämisestä, mutta osa on sitä mieltä, että tarvitseeko sen edes olla miellyttävää. Käyttäjien mielestä osa ohjelman toiminnoista on turhauttavia, esimerkiksi kun heidän pitää siirrellä pisteitä koodinaattien avulla.

Ohjelmasta pidetään selkeästi eniten sen tehokkuudesta. Yksinkertaisten rakenteiden suunnittelu on tehokasta. Ohjelma on erittäin tehokas ja tarkka laskentamoottori. Se on myös luotettava, kun käyttäjä tekee suunnitelmat niin siihen voidaan luottaa, että rakenne tulee kestäväksi. Avainsanana korostuu käytön tehokkuus.

Ohjelma ei ole pysynyt ajan tasalla nykyaikaisten ohjelmien kanssa. Ohjelma ei pysty keskustelemaan tai saamaan yhteyttä muihin ohjelmiin. Se ei ole kehittynyt tarpeeksi vuosien aikana. Esimerkiksi mitä nykyaikana halutaan saada rakenteisiin ei helposti saa tehtyä, käyttäjä joutuu tekemään paljon työtä käsin korjaten. Osa toiminnoista on hitaita ja vaikeita. Geometria on vaikea luoda vaikeimmissa tapauksissa. Kun tekee ylä- ja alapaarteiden väliin saumoja niin ne on kiinnitettävä ylä- ja alapaarteisiin, ei voida laittaa muihin osiin. Välillä saumoissa puut saattavat mennä päällekkäin 0,1 millimetriä milloin ohjelma

ei löydä saumoja, vaikka niin pienellä mitalla ei pitäisi olla merkitystä. Käyttöliittymää tulisi saada kehitettyä erikoisrakenteita varten.

Funktionäppäimet ovat isoimmassa roolissa ohjelman käytössä. Ohjelman kaikki vaiheet käydään aina alusta loppuun läpi, kun suunnitellaan rakenteita. Käyttäjien mielestä hiirelle ehkä voisi lisätä jotain toimintoja, jotka auttavat esimerkiksi zoomauksessa tai kapuloiden liikuttelussa. Ei kuitenkaan haluta siirtyä funktionäppäimistä hiiren käyttöön. Käyttöliittymää ajatellen funktionäppäimet ja hiiri voisivat olla rinnakkain.

Ohjelman navigointi toimii siten, että etusivulta pääsee aina etenemään seuraaviin vaiheisiin, mutta sinun täytyy aina palata etusivulle, että voit edetä seuraavaan vaiheeseen. Haastateltavien mielestä siinä ei ole mitään ongelmaa ja sivujen välillä hyppiminen käy erittäin nopeasti funktionäppäinten avulla. Yksi kommentoi, että kyllä se mukavasti toimii, kun ei paremmastakaan tiedä. Vanhoilla työntekijöillä se tulee selkärangasta, mutta he voisivat nähdä muutoksen siinä, että saataisiin uusille tulokkaille parannusta. Ohjelmassa ei ole välilehtiä vaan ne sivut ovat muistin varassa. Jos tekee muutoksen yhdessä ikkunassa, niin se päivittyy kaikissa ikkunoissa. Tässäkin kuitenkin tehokkuus on tärkeä ja heiltä se käy tehokkaasti.

Osa haastateltavista tykkää käyttää vanhoja töitä apuna ja osa tykkää tehdä aina alusta. Olisi hyvä, jos ohjelma pystyisi tarjoamaan valmiita pohjia, joista valita. Haluttaisiin myös ehdotuksia mikä vanhoista rakenteista toimisi parhaiten. Ongelmana on, että toisella ohjelmalla pitää hakea vanha työ ensin, ennen kuin sen saa suunnitteluohjelmaan pohjaksi. Harvoin rakenteet ovat aina samanlaisia, niihin joudutaan aina tekemään jotain muutoksia, mutta vanhoja on hyvä käyttää apuna.

Sisäinen laskin jakaa mielipiteitä. Osa näkee sen hyvänä lisänä ja osa täysin turhana, se olisi myös totuttelu kysymys. Tällä hetkellä kaikilla on oma taskulaskin työpisteellä, mitä he käyttävät. Nuoremmille ja uusille työntekijöille se voisi olla mielekäs. Siinä tarvitsisi kuitenkin olla sama logiikka kuin taskulaskimessa, ei liian monimutkainen eikä tieteislaskin. Ohjelma tietenkin laskee jo vaikeimmat laskut itse.

Ohjelmassa tuntuu olevan paljon hyvin yksityiskohtaisia puutteita. Esimerkiksi:

- Kapuloihin halutaan erilaisia työstö mahdollisuuksia, mitä voisi itse tehdä kuten sahauksia ja reikiä.
- Viisteet ja kolot ovat hitaita ja hankalia.
- Erikoisissa sahauksissa joudutaan siirtelemään pisteillä koordinaatit ja silloin ohjelma saattaa mennä lukkoon ja se on hyvin hidasta.
- Puun valinnoille haluttaisiin eri leveyksiä.
- Ohjelma laskee vain naulalevyliitoksen, mutta haluttaisiin sen laskevan muitakin liitoksia.
- Käyttäjät haluaisivat AutoCAD:in mallista toimintoa, että pystyisi piirtämään riskon ja silti laskentamalli pysyisi taustalla mukana.

- 3D mallinnusta toivottiin myös useamman puolesta.
- Haluttaisiin, että voidaan ottaa usea rakenne rinnakkain niin, että pystyy katsomaan kokonaisuutta.
- Enemmän nykytrendien mukainen.
- Ylä- ja alapaarteiden rajoitukset pitäisi poistaa, esimerkiksi tällä hetkellä alapaarteita voi olla neljä, mutta joskus tarvittaisiin kuusitoista.
- Saumoihin saa vain 3 pistettä, mutta niihinkin tarvitsisi saada enemmän pisteitä välillä.
- Kaivattaisiin myös parempaa muokattavuutta tulosteisiin, niihin esimerkiksi haluttaisiin kopioida tekstiä, mutta tällä hetkellä se ei ole mahdollista.
- Funktionäppäimiä haluttaisiin myös selkeämmäksi. Tällä hetkellä kerralla näkyy aina kahdeksan näppäintä ja, kun yhtä painaa niin aukeaa taas uudet kahdeksan ja taas, kun painaa yhtä niin aukeaa uudet kahdeksan.
- Haluttaisiin, että ohjelma tarjoaa valmiita malleja, kun syöttää tietyt parametrit rakenteelle.
- Geometria vaiheessa kaivattaisiin lisää erilaisia liitoksia, joista valita. Voisi olla kuvia koko ristikon saumoituksesta, mistä valita.

Toiminnoiltaan uuteen ohjelmaan halutaan samat kuin vanhassa. Tämän hetkinen rautalankamalli on helppo käsitellä ja yksinkertainen, ohjelmassa on kuitenkin tärkeimpänä ominaisuutena laskeminen, visuaalinen puoli veisi vain tehoja.

4.2.4. Johtopäätökset

Haastatteluiden avulla saatiin hyvä tilannekartoitus ohjelman tämän hetkisestä tilanteesta, käyttäjien tottumuksista ja mieltymyksistä. Käyttäjiltä sai myös hyvin palautetta, siitä mihin suuntaan ohjelmaa olisi hyvä kehittää tai millaisia ominaisuuksia he haluaisivat. Tämä tieto on erittäin tärkeää käyttäjäkeskeisessä suunnittelussa.

Tulosten perusteella voi sanoa, että kaikki käyttäjät toivovat edes jonkinlaista muutosta nykyiseen ohjelmaan. He olivat avoimia muutoksille, vaikka he ovat tottuneet nykyiseen ohjelmaan monien vuosien käyttökokemuksella. Monet osasivat ajatella muutoksien hyötyjä uusia käyttäjiä varten.

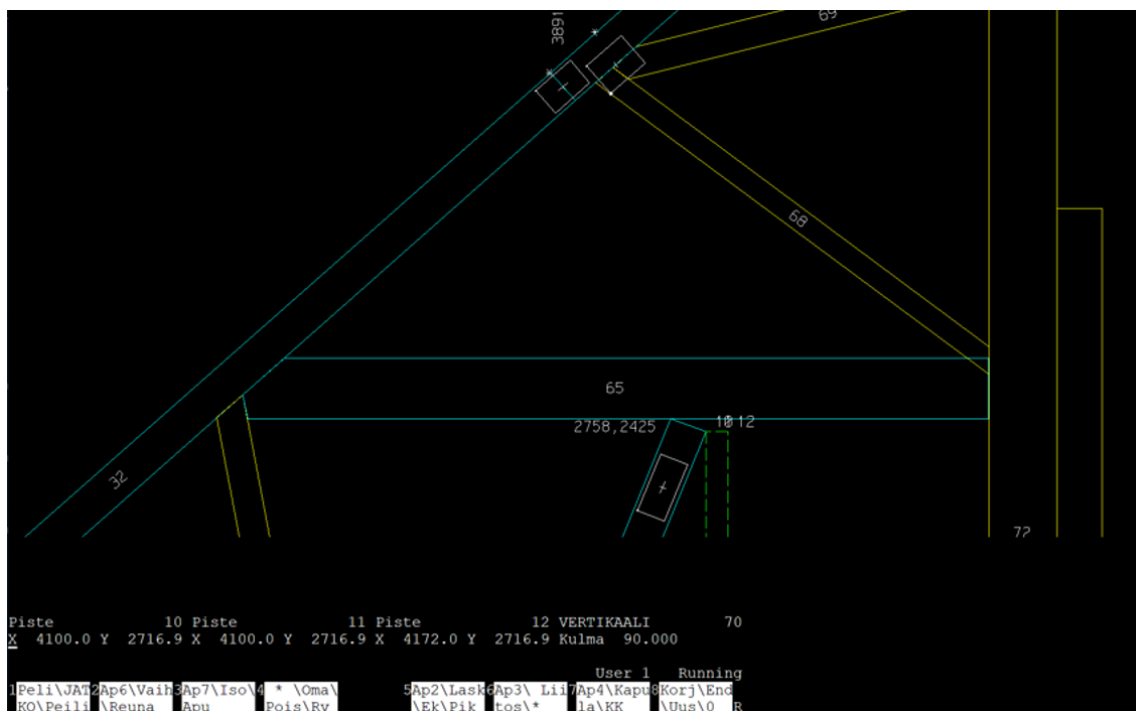
Ohjelmaan haluttiin todella paljon muutoksia, mutta yhden tutkielman puitteissa kaikkia mahdollisuuksia ei ole mahdollista tutkia. Tulosten perusteella täytyy löytää isoimpia ongelma-alueita, joihin pystyy vaikuttamaan käyttöliittymällä, ja jotka ratkaisemalla pystytään lisäämään käytön tehokkuutta ja löytää lisäarvoa ohjelmalle. Käyttäjäkeskeisen suunnittelun mukaan nyt on saatu määriteltyä käyttötilanne sekä käyttäjävaatimukset, joten voidaan aloittaa ideoiminen tulosten perusteella.

Koska käyttäjillä kuluu eniten aikaa erikoisten rakenteiden suunnittelussa, lähden tekemään muutoksia niihin. Ohjelmaan tullaan esittämään muutoksia varovasti ja pienin

askelin, sillä liian suuret muutokset voivat olla vaikea omaksua ja ne tyrmätään helpommin. Kriittisimpiin ongelmiin kuuluu eri osien/kapuloiden siirtely käyttäen koordinaatteja ja uskon, että siihen ongelmaan saadaan käyttöliittymää kehittämällä parempi ratkaisu.

4.3. Prototyypin kehittäminen

Prototyyppiä lähdettiin kehittämään käyttäjakeskeisen suunnittelun periaatteita noudattaen. Prototyyppiä aloitettiin luonnostelevaan haastatteluista löydettyjen ongelmakohtien avulla. Koska ongelma kohtia löytyi todella paljon, niitä ei pystytty ratkaisemaan kaikkia. Prototyyppiä lähdettiin rakentamaan yksi iso ongelma mielessä, kapuloiden siirtely pelkästään koordinaattien avulla. Tällä hetkellä ohjelmassa on jokaisella kapulalla kuusi eri pistettä, kolme pistettä molemmissa päissä, joita voi siirrellä. Vaikka kapulan pää olisi suorakulmainen siinä on silti kolme pistettä, jotka täytyy siirtää yksi kerrallaan. Niissä on kolme pistettä, koska niihin usein tehdään erilaisia sahauksia, joten kulmia syntyy enemmän. Kuvassa 3 näkyy esimerkki koordinaattien syöttämisestä. Kuvassa 3 on valittuna vihreällä katkoviivalla näkyvä kapula ja sen yläpään kolmea pistettä ollaan siirtämässä. Normaalisti kapula kiinnittyisi rakenteen ylimmäiseen paarteeseen automaattisesti, mutta koordinaatteja muuttamalla se on saatu kiinnitettyä poikittain olevaan kapulaan. Käyttäjällä ei ole myöskään apuna mitään koordinaatteja, mitä hän voi syöttää vaan ne syötetään kokeilun ja virheen kautta.



Kuva 3. Kapulan siirtäminen koordinaattien avulla.

Koska tämä ongelma nousi useassa haastattelussa esiin ja tämä hidastaa huomattavasti käyttäjien työskentelyä, lähdin ensimmäiseksi ehdottamaan erilaisia ratkaisuja tämän helppottamiseksi.

Ennen kuin lähdin kehittämään prototyyppiä, minun piti löytää sopiva ohjelma siihen. Nykypäivänä internet on täynnä erilaisia prototyyppi -työkaluja. Suurin osa niistä on kehitetty nettisivujen suunnitteluun, joten voi olla hankalaa löytää itselle sopivaa ohjelmaa. Aluksi koitin muutamia ilmaisia ohjelmia, mutta jokaisesta puuttui aina jotain ominaisuuksia, mikä teki prototyypin tekemisestä liian hankalaa. Siirryin katselemaan maksullisia ohjelmia ja muutaman ohjelman testaamisen jälkeen löysin sopivan ohjelman itselleni. Vaikka sekin on suunniteltu enemmän nettisivujen suunnitteluun, siitä löytyi kaikki ominaisuudet, mitä etsin. Ohjelma, johon päädyin on nimeltään Axure RP 9 (<https://www.axure.com/>).

Aluksi kehitelin parikymmentä kuvaa, miten ongelman voisi korjata. Käytin internetiä apuna ja etsin mahdollisia tapoja ratkaista ongelmaa. Kuvat olivat hyvin alkeellisia, koska niillä vain testailtiin mahdollisia ratkaisuja. Kun mielestäni löysin toimivan ratkaisun, aloitin prototyypin piirtämisen alusta. Aloitin piirtämään kuvia, miten käyttäjä etenisi ristikon suunnittelussa. Prototyyppi pysyi ulkoasultaan hyvin samannäköisenä kuin nykyinen ohjelma. Halusin käyttäjien keskittyvän enemmän toiminnallisiin muutoksiin kuin ohjelman esteettisyyteen. Prototyyppi pidettiin myös hyvin alkeellisena, koska käyttäjät saattavat tyrmätä idean kokonaan, joten se olisi ajan hukkaan heittoa, jos kehittäisi prototyyppiä liian yksityiskohtaiseksi tässä vaiheessa. Prototyypissä edettiin alkuun samalla lailla kuin nykyisessä ohjelmassa, ainoana lisänä siinä oli mahdollisuus valita hiirellä piste mihin jatketaan seuraava kapula, normaalisti se syötetään näppäimistöllä millimetri mittana. Prototyyppiin tehtiin myös muita lisäyksiä, mitä nykyisessä ohjelmassa ei ole kuten kapuloiden niin sanottu mielivaltainen lisääminen minne vain ristikkoon. Näin käyttäjän ei tarvitse koordinaattien siirtelyn avulla saada kapuloita oikeisiin kohtiin vaan ne voidaan lisätä halutessa tarvitsemaan kohtaan. Kapuloille lisättiin myös ominaisuuksia siirrellä ja muokata niitä vapaammin hiiren avulla. Niille lisättiin ominaisuus raaha ja pudota (eng. drag and drop) eli kapulaa pystyy siirtelemään rakenteessa helpommin. Uusia prototyypikuvia haastatteluja varten tehtiin 33. Prototyypin muutoksia käydään tarkemmin läpi seuraavassa luvussa.

4.4. Prototyypin testaaminen

Tässä kohtaa oli saatu valmiiksi prototyypin ensimmäinen versio ja se oli valmis testattavaksi. Tässä luvussa käydään testaamisen suunnitelma, itse toteutus, analyysi, tulokset ja johtopäätökset.

4.4.1. Suunnitelma

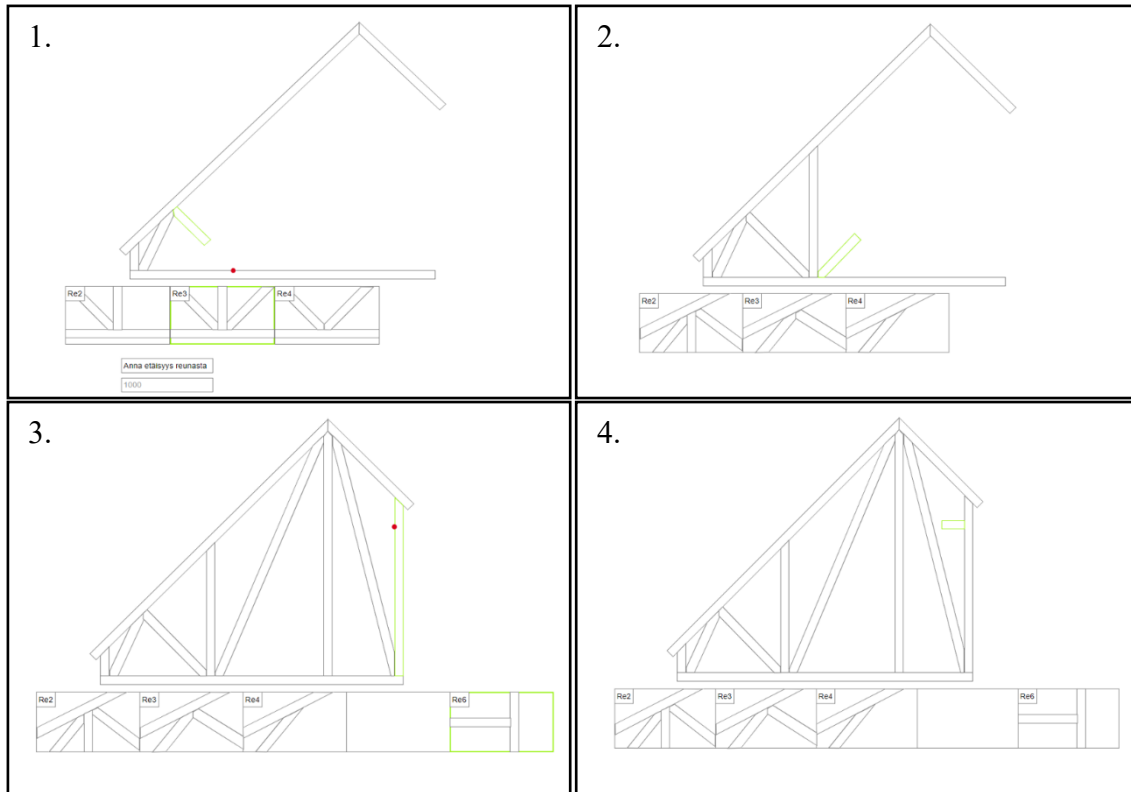
Haastattelut prototyypin pohjalta pidettiin huhtikuun 2020 ensimmäisellä ja toisella viikolla. Koronaviruspandemian vuoksi kaikki Swecon työntekijät oli määrätty etätöihin, jos etätö on vain mahdollista. Tämä poisti mahdollisuuden pitää haastatteluja kasvotusten, joten etähaastattelut olivat ainoa vaihtoehto.

Haastateltaviin otettiin yhteyttä sähköpostin välityksellä ja sovittiin heille sopiva päivä ja aika. Haastatteluihin varattiin aikaa noin puoli tuntia. Mahdollisia haastateltavia oli tässä vaiheessa vain kymmenen, ja koska kaikki ovat etänä töissä heihin oli vaikeampi saada kontaktia. Tähän vaiheeseen sai osallistua, vaikka ei olisi osallistunut ensimmäisiin haastatteluihin. Mitä enemmän saadaan osallistujia, sitä paremmin saadaan taattua käyttäjiä miellyttävä ja heidän tarpeitaan vastaava ohjelma. Suunnitelma ja prototyyppi käytiin läpi Swecon tutkimus- ja kehitysjohtajan kanssa ennen kuin haastatteluja aloitettiin.

Haastatteluiden ohjelmaksi valittiin Microsoft Teams, sillä Sweco käyttää sitä muutenkin, niin jokaiselta tulisi löytyä kyseinen ohjelma ja toivottavasti he osaavat myös käyttää sitä. Microsoft Teamsin avulla pystyy helposti jakamaan oman näytön, joten haastateltava näkee prototyypin koko ajan. Ohjelma tarjoaa myös vaihtoehdon kokouksen talentamiselle, joka on erittäin hyvä lisä.

Haastatteluja varten ei tehty erillistä kyselylomaketta. Prototyypissä on selkeitä kohtia, mistä pyydettiin mielipidettä, mutta en nähnyt tarpeelliseksi tehdä strukturoitua tai puolistrukturoitua haastattelua, avoin haastattelu sopi tähän hyvin. Avoin haastattelu on tavallisen keskustelun omainen, jossa keskustelun etenemistä ei ole lyöty lukkoon, haastattelussa toki pyritään puhumaan tutkijan etukäteen poimimista teemoista, mutta haastattelu etenee haastateltavan ehdoilla [Saaranen-Kauppinen ja Puusniekka 2006].

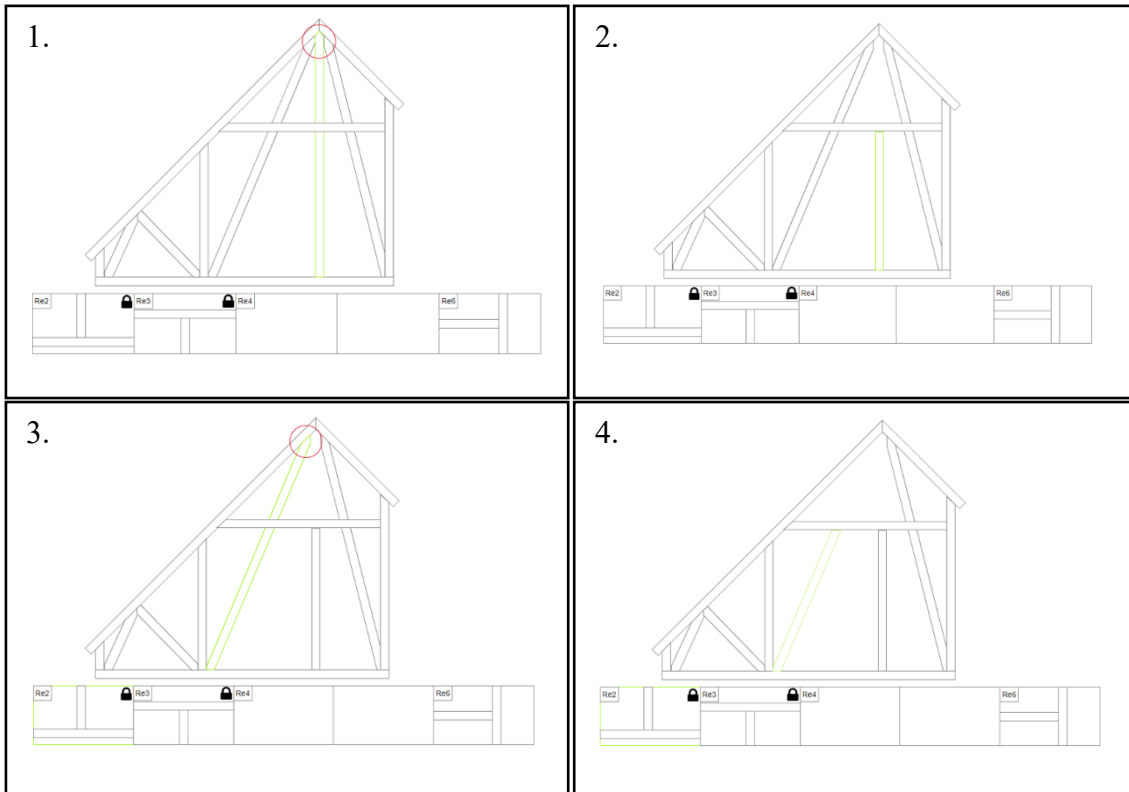
Kuvassa 4 on kuvia prototyypistä. Kuvassa 4 kohdassa 1 on kuvattu uusi tapa jatkaa suunnittelua hiiren avulla. Siinä on valittuna (alareuna), minkälaiset kapulat piirretään seuraavaksi ja punainen piste on indikoimassa, että siihen voi klikata hiirellä. Kuvassa 4 kohdassa 2 näkyy, mitä tapahtuu, kun punaiseen pisteeseen klikataan ja siinä tarjotaan taas uusia vaihtoehtoja, miten jatketaan rakenteen suunnittelua. Kuvassa 4 kohdassa 3 on luotu uusi toiminto eli siinä voi piirtää uuden kapulan mihin kohtaan vain valitussa kapulassa. Vihreällä näkyvä kapula on valittuna ja siihen voi piirtää uuden, punainen piste on taas indikoimassa, että siihen kohtaan voidaan piirtää uusi kapula. Kuvan 4 kohdassa 4 näkee, mitä ohjelma tekee, kun punaiseen pisteeseen klikataan eli ohjelma piirtää kyseiseen kohtaan uuden kapulan. Nämä ovat esimerkkejä, mistä halusin saada tietoa haastateltavilta, että miltä tämä vaikuttaa ja näkevätkö he tätä tarpeellisena.



Kuva 4. Ensimmäisen prototyypiversion kuvia.

Seuraavaksi tein muutoksia kapuloiden liikutteluun. Kuvassa 5 kohdassa 1 on valittuna vihreällä näkyvä kapula. Punainen ympyrä on indikoimassa, että kapulan yläreunasta voi ottaa kiinni. Prototyypin tarkoituksena on demonstroida, että kapulan kokoa voidaan muuttaa ottamalla siitä kiinni ja pystytään siirtämään sen kiinnitys toiseen kapulaan. Kuvassa 5 kohdassa 2 se on siirretty alempana olevaan kapulaan kiinni. Tämän hetkessä ohjelmassa tämä täytyisi tehdä koordinaatteja siirtämällä, mutta mielestäni tähän voisi käyttää apuna raahaa ja pudota toimintoa. Kuvassa 5 kohdassa 3 on kuvattu toisenlainen tapa, siinä on lukittuna kapulan alareuna paikalleen, tällöin kun sitä lähdetään siirtämään, kapulan alapään sahauskset eivät muutu. Kuvassa 5 kohdassa 4 kapula on siirretty alempaan kapulaan kiinni, mutta alapään sahauskset ovat pysyneet ennallaan. Kapulaa ei ole pakko lukita vaan sitä voidaan siirrellä myös vapaammin.

Näistä kahdesta muutoksesta oli myös tärkeä saada palautetta. Mielestäni ratkaisut ovat järkeviä, mutta käyttäjät voivat olla eri mieltä. Tarkoitukseni oli tarjota erilaisia vaihtoehtoja ja ratkaisuja, joita käyttäjät voivat arvioida.



Kuva 5. Prototyypikuvia.

Nämä muutokset prototyypissä ovat ydintehtäviä ja näistä halusin saada eniten palautetta. Halusin saada selville, näkevätkö käyttäjät näitä tarpeellisiksi tai suunnittelua nopeuttaviksi. Heidän täytyi myös miettiä, haluavatko he käyttää hiirtä enemmän, sillä suurin osa muutoksista tehtiisiin hiiren avulla. Haastateltavalle näytettiin ensin koko prototyyppi läpi ja sitten palattiin tarkastelemaan näitä yksittäisiä muutoksia. Koska haastattelu oli avoin, siihen jäi tilaa myös muille kommenteille, mutta nämä ovat poimittuja teemoja, jotka käytiin läpi.

4.4.2. Toteutus

Otin yhteyttä kymmeneen käyttäjään, mutta vain kolme vastasi minulle. Haastattelut saatiin sovitettua heidän aikatauluunsa. Jokainen haastattelu kesti reilu puoli tuntia. Haastateltavilta sai todella paljon mielipiteitä ja ehdotuksia, miten jatkaa prototyypin kehittämisessä, joten päätin kolmen haastattelun riittävän. Kaikki haastattelut tehtiin etänä käyttäen Microsoft Teamsia ja niistä nauhoitettiin ääni sekä kuva.

4.4.3. Analyysi/tulokset

Haastattelujen analyysi aloitettiin heti haastattelun päätyttyä. Haastattelun aikana kirjattiin jo muistiinpanoja. Haastattelujen jälkeen ylös kirjattiin kommentteja haastattelusta ja tärkeimmät nostot haastateltavan osalta. Näitä haastatteluja ei litteroitu vaan niitä käytiin

läpi myöhemmin ja kirjattiin huomiot ylös. Analyyseissä otettiin jo huomioon edellisten haastateltavien kommentteja ja pyrittiin löytämään yhteneväisyyksiä.

Tuloksia käydään läpi aloittaen kuvasta 4 ja etenemällä kuvaan 5. Lopuksi käydään vielä läpi lisähuomioita, mitä käyttäjiltä tuli prototyypin haastattelun aikana.

Kuvan 4 kohdan 1 ja 2 muutosta ei koettu kovin tarpeelliseksi, ainakaan tässä yhteydessä. Käyttäjät haluavat, ja heidän pitääkin, syöttää mitat tarkasti. Todennäköisesti hiirellä valittaessa osien paikkoja, ei saada tarpeeksi tarkkoja mittoja. Kuitenkin tätä ominaisuutta voidaan käyttää hyödyksi suunnittelussa, mutta pitää olla mahdollisuus tarkentaa mitat. Asiakkaat vaativat millin tarkkoja mittoja, joten mahdollisuus siihen täytyy olla. Vaikka ristikon piirtäisi käyttämällä vain hiirtä, siihen täytyy lisätä jokin tapa, millä mitat saadaan tarkasti asetettua tai niitä voi myöhemmin muokata.

Kuvan 4 kohdan 3 ja 4 lisäykset vaikuttivat käyttäjien mielestä hyvältä. Ominaisuutta kuvailtiin todella tärkeäksi ja näppäräksi. Tällä hetkellä, kun ei ole tuollaista ominaisuutta, mutta sellainen tarvittaisiin. Tähän täytyy myös lisätä vaihtoehto, jolla voidaan antaa mitta kapulalle alareunasta, että saadaan se varmasti oikeaan korkeuteen. Toinen ehdotus olisi, että kapulaa voisi siirrellä hiirellä ja mukana näkyisi koko ajan sen korkeus rakenteen alatasosta. Hiiren osoittimen mukana kulkisi siis vierivä mitta siirron yhteydessä. Tämä on hyvä lisä, varsinkin jos se saadaan kaikkiin ristikon osiin ja kapuloita voidaan lisätä mihin vain. Tällaista ominaisuutta ei ole tällä hetkellä ja se on todella tarpeellinen.

Kuvan 5 kohdissa 1 ja 2 demonstroitiin, miten yksittäistä kapulaa pystytään siirtämään. Käyttäjien mielestä tämä vaikutti erittäin hyvältä. Tällä hetkellä se tehdään siirtämällä kapulan toisen pään pisteiden koordinaatteja. Kapuloiden koordinaattien nopeaa muuttoa tarvitaan ja tämä mahdollistaa sen. Tässä nousi myös huomiona, että välillä kahden kapulan väliin voidaan jättää pieni kolo, niin sanottu palkkikolo, joten jos kapulat ”loksahtavat” kohdilleen niin siinä täytyy myös olla mahdollisuus jättää pieni kolo niiden väliin.

Kuvan 5 kohdissa 3 ja 4 annettiin mahdollisuus lukita toinen pää kapulasta ennen kuin sitä aletaan siirtämään. Käyttäjien mielestä tämä on ehdoton lisä mikä pitää saada, jos lisätään ominaisuus hiirellä muuttaa kapuloiden paikkaa ja kokoa. Välillä kapulan toisen pään sauhuksien mitat ovat todella tärkeitä pitää koskemattomina, varsinkin jos ne on asetettu jo valmiiksi vaatimusten mukaan.

Ylipäätään käyttäjät olivat tyytyväisiä kaikkiin muutoksiin. Heidän mielestään prototyypissä oli todella paljon hyviä ideoita, jotka olisi hyvä saada toteutettua uuteen ohjelmaan. Koska käyttäjät olivat niin tyytyväisiä muutoksiin, niihin ei tullut merkittävästi kehitysehdotuksia, joten haastatteluissa jäi aikaa myös muulle palautteelle. Seuraavaksi käydään läpi muutamia lisäyksiä ohjelmaan, joita toivottiin.

Haastatteluissa nousi esille niin sanottu lyhty. Rakenteeseen välillä tarvitsee lisätä lyhdyn näköinen rakennelma rakenteen yläpaarten yläpuolelle. Yläpaarten yläpuolelle tulisi vaakapuu eli lyhty. Myös tämän lyhdyn alapuolelle jäävä yläpaarre pitää saada poistettua, joissain tilanteissa. Toisena uutena toivottuna lisäyksenä nousi esiin viisteet. Eli esimerkiksi yläpaarten toiseen päähän tarvitsee saada viiste. Nykyinen ohjelma ei tarjoa sille nopeaa ja helppoa ratkaisua vaan sen tekeminen vie huomattavasti aikaa tehdä. Kolmantena nousi esiin yläpaarteiden koloukset. Välillä niihin tehdään päästä lähteviä pieniä kolouksia tai sitten keskelle yläpaarretta pieni kolo. Tähänkään ei ole nopeaa ja helppoa tapaa tehdä. Viimeiseksi esiin nousi vielä infolaatikot kapuloista. Toivottiin, että olisi nopea tapa nähdä, että minkä kokoisia kapulat ovat ja minkälaisia sahauksia niissä on.

4.4.4. Johtopäätökset

Prototyypin ehdottamiin muutoksiin ei tullut kovin paljoa kehitysehdotuksia. Yleisesti niihin oltiin todella tyytyväisiä ja ne koettiin tarpeellisena, muutamaa muutosta lukuun ottamatta. Haastatteluissa nousi myös esiin todella paljon lisää erilaisia tarpeita. Prototyyppi kaipasi vielä paljon kehitystä eikä se kattanut vielä kaikkia ongelma-alueita. Tämä oli hyvä ensimmäinen haastattelukierros prototyypin kanssa. Ja kaiken kaikkiaan haastatteluista jäi sellainen kuva, että käyttäjät odottavat uusia muutoksia ja niitä todella kaivataan. Koska alkuperäisiin lisäyksiin oltiin tyytyväisiä, lähdin kehittämään seuraavasta prototyyppiversiosta vielä yksityiskohtaisempaa ja koitin ratkoa esille nousseita uusia ongelma-alueita.

4.5. Prototyypin jatkokehittäminen

Prototyyppiä lähdettiin taas kehittämään edellisten haastattelujen perusteelta. Prototyypin jatkokehityksessä pidettiin mielessä käyttäjäkeskeisen suunnittelun periaatteet. Seuraavaksi kiinnitettiin huomioita prototyyppissä oleviin ominaisuuksiin ja niihin liittyviin korjauksiin. Haastatteluissa tuli hyviä huomioita ja lisäyksiä, mutta mielestäni kaikkia muutoksia ei tarvitse tehdä prototyyppiin, kunhan ongelmat ovat tiedossa. Varsinkin jos muutokset prototyyppiin eivät anna lisäarvoa seuraavalle haastattelukierrokselle, niitä ei kannata välttämättä tehdä. Prototyyppiä lähdettiin kehittämään yksityiskohtaisemmaksi, pitäen mukana edelliset muutokset ja niitä parantaen.

Haastatteluissa nousi muutamia lisäyksiä prototyyppiin ja uuteen ohjelmaan, jotka mielestäni pystyttiin helposti lisäämään ohjelmaan. Kun lähdin kehittämään prototyyppiä lisää, aloitin tekemällä toivottuja lisäyksiä. Muutokset jo oleviin ominaisuuksiin tehtiin, mikäli niitä käytettiin uusissa prototyyppikuvissa. Lisäykset mitä prototyyppiin haluttiin, olivat lovet, viisteet, infolaatikko sekä mahdollisuus tehdä lyhtyjä helpommin.

Uusia kuvia lähdettiin kehittämään ensimmäisen prototyypin kuvista. Kuvista tehtiin yksityiskohtaisempia. Ohjelman käyttöä koitettiin kuvata mahdollisimman selkeästi, että käyttäjät pystyvät sisäistämään ominaisuuksien käytön. Yksittäisistä prototyypikuvista voi olla vaikea hahmottaa, että miten ohjelmaa tullaan oikeasti käyttämään, mutta tekemällä mahdollisimman selkeäksi kaikki välivaiheet, käyttäjät ymmärtävät vaiheet paremmin ja he pystyvät antamaan laadukkaampaa palautetta. Prototyypin ulkonäkö jäljitteli vieläkin vanhaa ohjelmaa, jotta pystyin pitämään huomiot uusissa ominaisuuksissa, eikä niinkään estetiikassa. Myös ohjelman käyttö on helpompi sisäistää, jos se muistuttaa hyvin paljon heidän nykyistä ohjelmaansa. Uusia prototyypikuvia kehitettiin 41, että ohjelman työvaiheet olisivat mahdollisimman selkeitä. Prototyyppeä esitellään tarkemmin seuraavassa luvussa.

4.6. Prototyypin testaaminen

Tässä kohtaa oli saatu valmiiksi prototyypin toinen versio ja se oli valmis testattavaksi. Tässä luvussa käydään testaamisen suunnitelma, toteutus, analyysi, tulokset ja johtopäätökset.

4.6.1. Suunnitelma

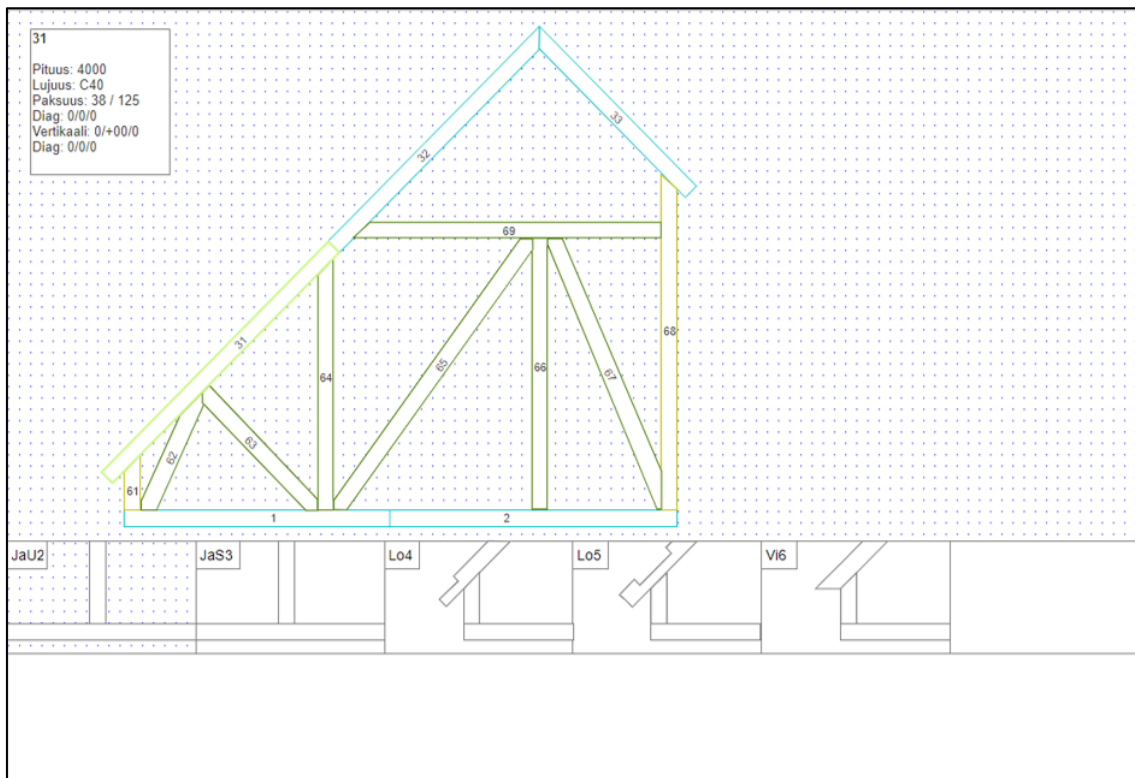
Haastattelut prototyypin toisen version pohjalta pidettiin huhtikuun 2020 neljännellä viikolla. Koronavirusepidemian vuoksi kaikki Swecon tekijät olivat vieläkin etätöissä, joten haastattelutkin pidettiin etäyhteydellä. Haastateltaviin oltiin yhteydessä sähköpostilla ja sovittiin heille sopiva päivä ja aika. Näihinkin haastatteluihin varattiin noin puoli tuntia. Mahdollisten haastateltavien määrä oli vieläkin sama eli kymmenen. Tässäkin vaiheessa oli tärkeä saada mahdollisimman paljon osallistujia, että saadaan heidän tarpeisiinsa mahdollisimman käytännöllinen suunnittelu. Suunnitelma ja prototyyppi käytiin läpi Swecon tutkimus- ja kehitysjohtajan kanssa ennen kuin haastattelut aloitettiin.

Haastatteluihin käytettävää ohjelmaa ei vaihdettu eli nämäkin haastattelut tehtiin käyttäen Microsoft Teamsia. Koska ensimmäiset etähaastattelut sujuivat hyvin sen kanssa, niin ei ollut syytä vaihtaa ohjelmaa.

Näitäkään haastatteluja varten ei tehty erillistä kyselylomaketta. Tässäkin prototyypin versiossa oli selkeitä kohtia, joihin tultiin pyytämään erikseen palautetta, mutta en nähnyt vieläkaan tarpeelliseksi tehdä strukturoitua tai puolistrukturoitua haastattelua. Edellinen kierros sujui hyvin käyttäen avoimia haastatteluja, joten sillä suoritettiin nämäkin.

Tämän ollessa viimeinen versio prototyypistä minkä pohjalta tehtiin haastatteluja niin prototyyppeä kuvataan tässä osiossa tarkemmin. Prototyypikuvia on yhteensä 41, joten niitä kaikkia ei käydä läpi. Tässä luvussa käydään ydintoiminnot läpi ja kuvataan prototyyppeä tarkemmin.

Kuvassa 7 näkyy kehitetty prototyyppi. Kaikki osat on numeroitu käyttäen samaa logiikkaa kuin nykyinen ohjelma. Osiin on lisätty värit myös noudattaen heidän nykyisen ohjelman logiikkaansa. Prototyypissä on tällä hetkellä valittuna yläpaarre 31. Heidän toiveidensa mukaan vasemmalle ylös on lisätty infolaatikko ja sinne on laitettu joitain tietoja, että he saisivat idean mitä sinne voisi laittaa ja miltä se tulisi näyttämään. Rakenteen alapuolella on valmiita malleja, millä voidaan lähteä tekemään muutoksia ristikkoon. Kaikki nämä ovat uusia muutoksia, nykyisestä ohjelmasta ei löydy tällaisia vaihtoehtoja. Rakenteen taustalle on myös lisätty ruudukko, jonka ideana on auttaa hiiritoimintojen kanssa. Ensimmäiseksi lähdemme katsomaan, miten näillä muutoksilla pystytään lisäämään lyhty, joka lähtee yläpaarteesta 31. Ohjelmassa on vaihtoehdot ”JaU2” sekä ”JaS3”. Näillä tarkoitetaan jatkopala ulkopuoli ja jatkopala sisäpuoli, numerot ovat näppäinkomentoja varten. Tämä nimeämislogiikka on sama kuin heidän nykyisessä ohjelmassaan, mutta tästä halusin palautetta. Myös ”JaU2” toiminnon taustana on ruudukko, että sen erottaisi helpommin toisesta vaihtoehdosta. Ideana on, että valittaessa JaU2 käyttäjä voi laittaa jatkopalan yläpaarteesta 31 yläpuolelle eli niin sanotulle ulkopuolelle. JaS3 valinnan tehdessään hän voi tehdä jatkopalan yläpaarteesta 31 alapuolelle eli niin sanotulle sisäpuolelle. Koska tässä tehdään lyhty yläpuolelle, jatkamme prototyypin tarkastelua tekemällä valinnan JaU2.



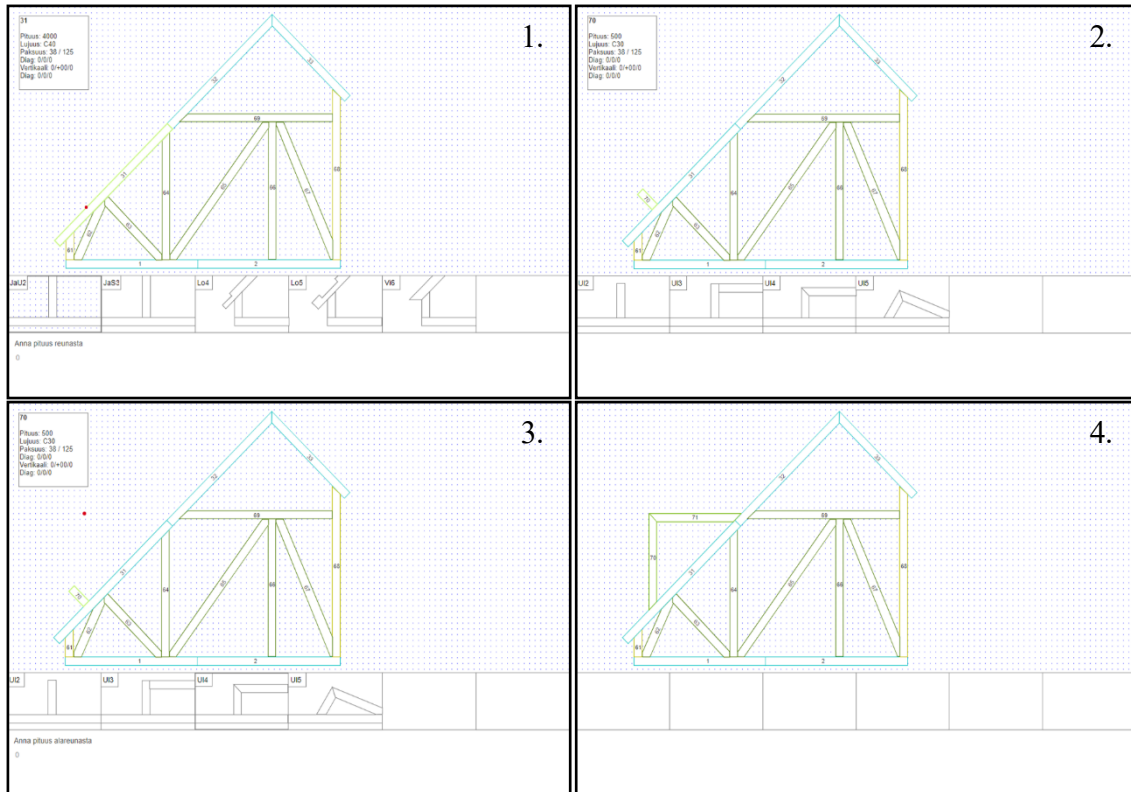
Kuva 7. Prototyypikuva, jossa valittuna yläpaarre 31.

Kuvassa 8 kohdassa 1 on valittuna JaU2. Valittu laatikko näkyy korostettuna, että käyttäjä muistaa, minkä valinnan hän on tehnyt. Alhaalle tulee teksti ”Anna pituus reunasta” tarkoittaen, että käyttäjä voi antaa uuden kapulan etäisyyden rakenteen vasemmasta reunasta. Vaihtoehtona on myös hiirellä klikata yläpaarteen yläreunaa siitä kohdasta, mistä haluaa kapulan lähtevän, punainen piste on merkitsemässä tätä vaihtoehtoa. Tässä on säilytetty vanha tapa syöttää etäisyys rakenteen reunasta ja uutena ominaisuutena on hiirellä klikata oikeaa kohtaa, kuten toiveena oli.

Kuvassa 8 kohdassa 2 käyttäjä on painanut edellisen kohdan punaista pistettä ja ohjelma on piirtänyt kapulan valittuun kohtaan. Seuraavaksi tarkoituksena on lähteä rakentamaan tätä kapulaa pidemmälle. Valikkoon on ilmestynyt uusia vaihtoehtoja lyhdyn rakentamista varten. Sinne on laitettu esimerkkejä, miten lyhtyä voisi lähteä rakentamaan, että käyttäjät saavat idean, minkälaisia vaihtoehtoja siihen voisi saada. Tämän prototyypin osalta käyttäjä valitsee kolmannen vaihtoehdon, eli lyhtyyn tulee 90-asteen kulma.

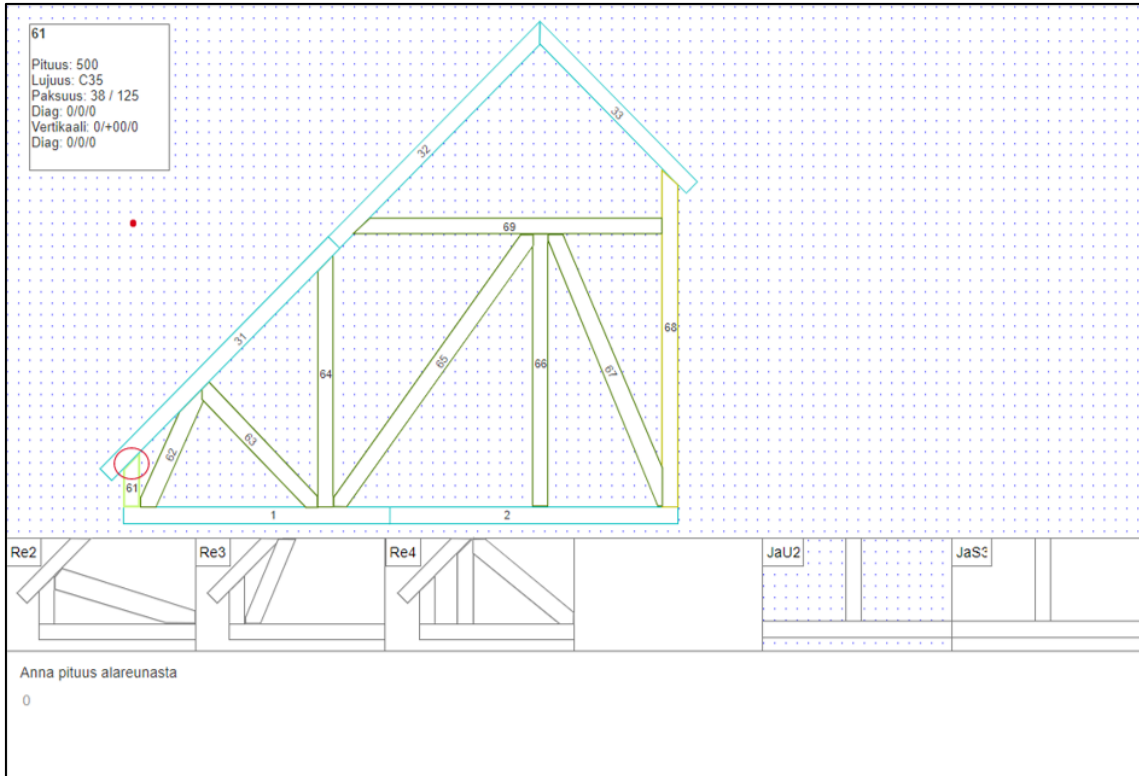
Kuvassa 8 kohdassa 3 näkyy taas valittu laatikko korostettuna, että käyttäjät muistavat valintansa. Alhaalle on tullut teksti ”Anna pituus alareunasta” tarkoittaen, että käyttäjä voi antaa mitan rakenteen alareunasta. Halusin tarkennuksen, halutaanko mitta antaa alapaarteen alareunasta vai alapaarteen yläreunasta. Vaihtoehtona on myös hiirellä klikata ruudukkoa, prototyypikuvaan on lisätty punainen piste selkeyttämään, että sinne voidaan klikata.

Kuvassa 8 kohdassa 4 käyttäjä on klikannut edellisen kohdan punaista pistettä. Koska käyttäjä oli valinnut kapuloille 90-asteen kulman, ohjelma osaa piirtää seuraavan kapulan liittymään yläpaarteeseen. Jos kyseessä ei olisi ollut 90-asteen kulma, ohjelma ei olisi piirtänyt seuraavaa kapulaa valmiiksi vaan käyttäjän olisi täytynyt tehdä se itse. Tämä automaattinen piirto oli yksi asioista, joista halusin myös saada palautetta. Uusien kapuloiden tietoja voi tarkastella viemällä hiiren niiden päälle, tämä avaa kuvissa nähdyn infoaatikon, tämä on ominaisuus, jota kanssa toivottiin ja se demonstroitiin selkeämmin haastatteluissa. Tämä oli ensimmäinen esimerkki lyhdyn suunnittelulle.



Kuva 8. Lyhdyn rakentaminen, tapa yksi.

Kuvassa 9 aloitetaan toinen tapa tehdä lyhtyä. Tässä on valittuna kapula 61, vasemmassa reunassa. Tämän ideana on kasvattaa reunimmaista kapulaa ja käyttää sitä lyhdyn tekemiseen. Punainen ympyrä on indikoimassa, että kapulan yläreunasta voi ottaa kiinni ja sen voi hiirellä kasvattaa punaiseen pisteeseen asti. Tähän toivottiin juoksevaa mitta, kun hiirellä kasvatetaan, mutta prototyyppiin se oli vaikea tehdä, mutta on tärkeä tietää, että sellaista tarvitaan. Toinen tapa on syöttää kapulalle uusi mitta, alhaalla näkyy teksti ”Anna pituus alareunasta”, jonka tarkoituksena on, että käyttäjä voisi syöttää kapulalle uuden mitan. Tähän liittyen pieni selvítettävä asia oli sanoitus tekstille, jotta siitä saadaan mahdollisimman selkeä. Tässä kohtaa ei tehdä vielä mitään valintoja, sillä nyt vain kasvatamme kapulaa. Valintoina on samanlaisia vaihtoehtoja kuin nykyisessä ohjelmassa, mutta sinne on myös lisätty muutama, mitä käytämme seuraavassa vaiheessa.



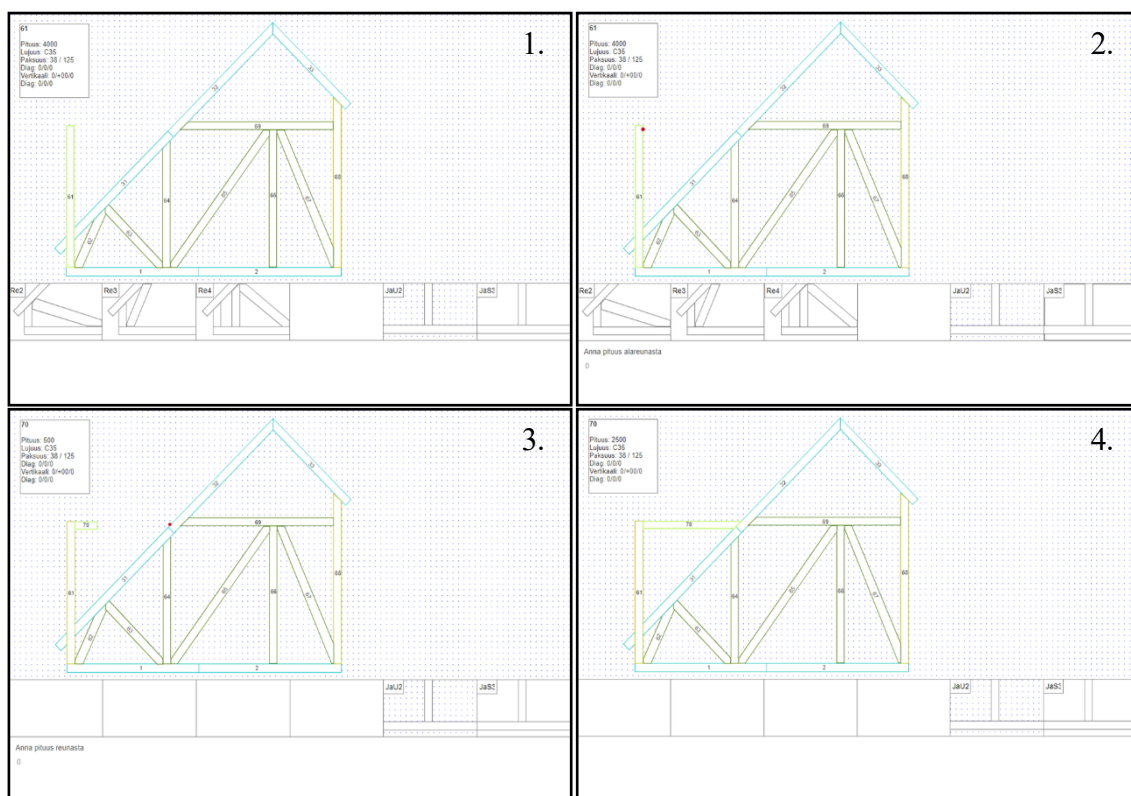
Kuva 9. Lyhdyn rakentaminen, tapa kaksi.

Kuvassa 10 kohdassa 1 käyttäjä on jatkanut kapulaa kuvan 9 punaiseen pisteeseen. Seuraavaksi haluamme lisätä yhden kapulan valittuun reunimmaiseen kapulaan. Tässäkin on samanlaiset vaihtoehdot kuin oli kuvassa 7 eli JaU (jatkopala ulko) ja JaS (jatkopala sisä). Tässä tilanteessa JaU tarkoittaa kapulan vasenta reunaa ja JaS tarkoittaa kapulan oikeaa reunaa. Tämä on yksi asia mihin tarvitsin palautetta käyttäjiltä, sillä tämä voi olla hieman sekava. Varsinkin jos miettii, että valittu kapula olisi yksi ristikon keskellä sijaitseva kapula. Oletuksena on, että tämä tapa merkitä näitä kahta vaihtoehtoa on liian sekava. Mutta tämän haastattelun aikana valitsimme vaihtoehdon JaS.

Kuvassa 10 kohdassa 2 käyttäjä on tehnyt valinnan JaS. Valittu laatikko näkyy korostettuna, että käyttäjä muistaa minkä valinnan hän on tehnyt. Alhaalle tulee teksti ”Anna pituus alareunasta” tarkoittaen tässä tilanteessa alapaarteen yläreunaa. Meillä on tällä hetkellä valittuna reunimmainen kapula ja infolaatikosta me näemme sen korkeuden. Me voimme syöttää tämän saman korkeuden alhaalla olevan tekstin alapuolelle. Tällöin ohjelma piirtää kapulan valitun kapulan yläpäähän. Toinen vaihtoehto on klikata hiirellä kapulan yläpäähän, jossa näkyy punainen piste.

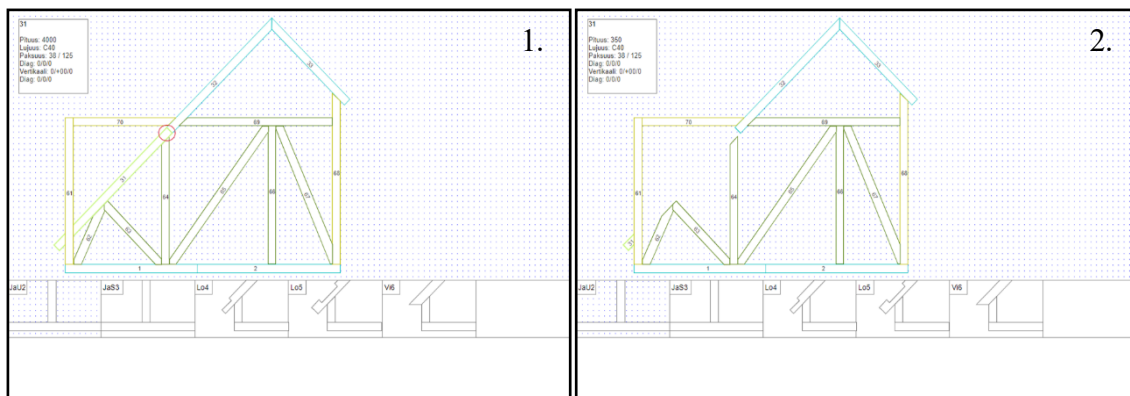
Kuvassa 10 kohdassa 3 käyttäjä on piirtänyt uuden kapulan reunimmaiseen kapulan yläpäähän ja nyt tarkoituksena on jatkaa sitä. Käyttäjälle tulee taas teksti alas ”Anna pituus reunasta”, joka tarkoittaa ristikon vasenta reunaa. Tai käyttäjä voi hiirellä venyttää kapulan haluamaansa kohtaan. Punainen piste on indikoimassa, että sinne voi venyttää hiirellä kapulan.

Kuvassa 10 kohdassa 4 käyttäjä on venyttänyt kapulan yläpaarteeseen kiinni. Ongelmana tässä on, jos käyttäjä tekee tämän kaiken hiirellä syöttämättä tarkkoja mittoja, osat eivät välttämättä ole linjassa toisiinsa vaakasuoraisesti. Tämän korjatakseen käyttäjä voi painaa kapulan keskiosasta, jolloin hän voi hiirellä tai nuolinäppäimillä siirtää valittua kapulaa. Ideana on, että keskiosasta klikkaamalla kapulaa voi siirrellä ja jommastakummasta päästä klikattaessa kapulaa voi kasvattaa. Kun kapuloita siirrellään apuna voisi olla punaiset apuviivat, että kaikki osat saadaan kohdistettua linjaan toistensa kanssa. Tämä oli yksi asia, mistä halusin käyttäjien mielipiteitä.



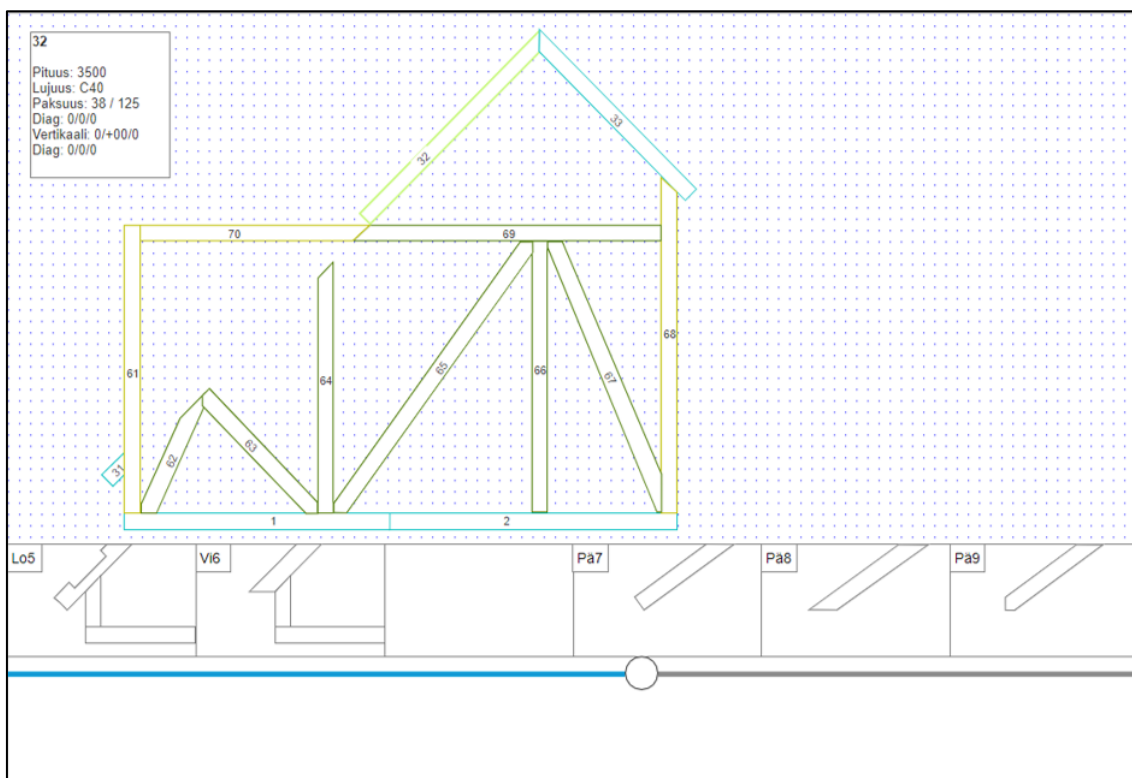
Kuva 10. Lyhdyn rakentaminen, tapa kaksi.

Kuvassa 11 kohdassa 1 on valittuna yläpaarre 31, joka on jäänyt lyhdyn alle. Käyttäjät tarvitsevat myös vaihtoehdon muokata näitä alle jääviä osia. Kapulan toisessa päässä on punainen ympyrä indikoimassa, että siitä voi ottaa kiinni ja muokata kapulaa pienemmäksi. Kuvan 11 kohdassa 2 kapula on muokattu pieneksi ja se jäi todella pieneksi osaksi ristikon vasempaan reunaan. Tästä halusin myös kysyä, että tarvitseeko sinne jättää tuollaista pientä kapulaa vai olisiko se vain parempi poistaa kokonaan.



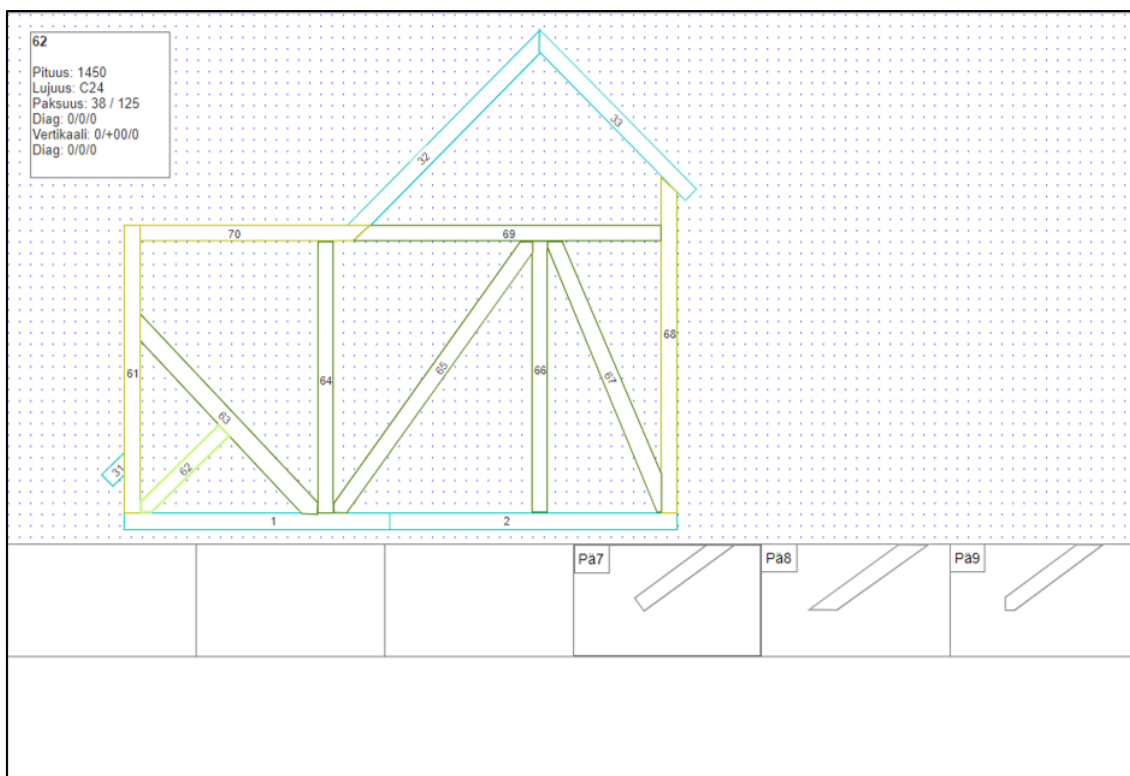
Kuva 11. Lyhdyn alle jäävien osien muokkaus.

Kuvassa 12 kapuloita on muokattu jo hieman pidemmälle käyttäen samaa logiikkaa kuin kuvassa 11. Kuvassa 12 on lisätty taas uusi ominaisuus, valinnoille on lisätty liukusäädin, että saadaan enemmän valmiita vaihtoehtoja. Tällä hetkellä käyttäjillä on vain rajallinen määrä vaihtoehtoja, joista he lähtevät muokkaamaan haluamaansa. Tässä kuvassa on lisätty vaihtoehtoja muokata kapulan päätä, eli minkälainen sahaus siihen halutaan. Ajatuksena on, että kapulasta painetaan sitä päätä, jota halutaan muokata, mutta tämä oli yksi asia mistä halusin saada palautetta, eli miten käyttäjät sen kokevat, tarvitseeko se esimerkiksi tehdä selkeämmin. Tässä kohtaa prototyypin haastattelua on valittuna kapulan 32 alapääty ja tarkoituksena on tehdä valinta”Pä8” eli tarkoittaen päätä ja näppäinkomentoa varten.



Kuva 12. Liukusäädin ja pään muokkaaminen.

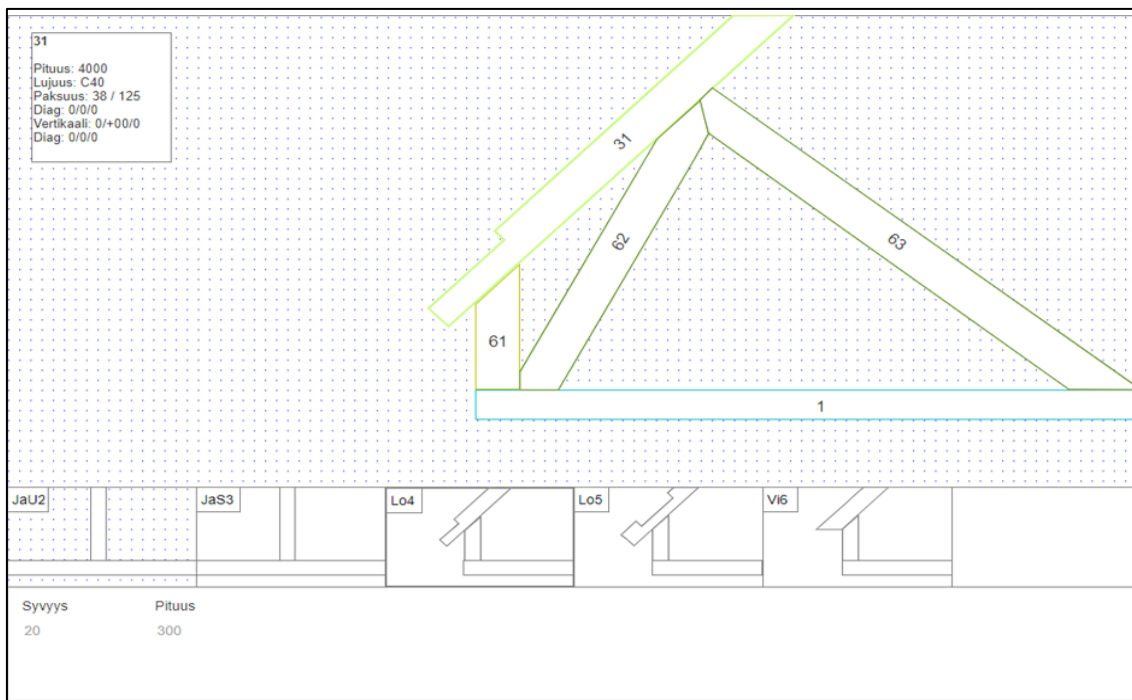
Kuvassa 13 nähdään, miten kapulan 32 pää on muokkaantunut. Tässä kohtaa kuvaa on jo muokattu pidemmälle. Tämä kuva on otettu esimerkiksi, sillä tässä on valittuna kapula 62 ja sen oikea pää on muokattu tasaiseksi. Sen valinta näkyy korostettuna. Tällöin ohjelma on kääntänyt kapulan 62 menemään tasaisesti kapulaan 63. Kapulan 62 vasemman pään sahaukset ovat tällöin muuttuneet. Tämä oli yksi asia, mistä halusin käyttäjiltä palautetta.



Kuva 13. Pään muokkaaminen kääntää kapulaa.

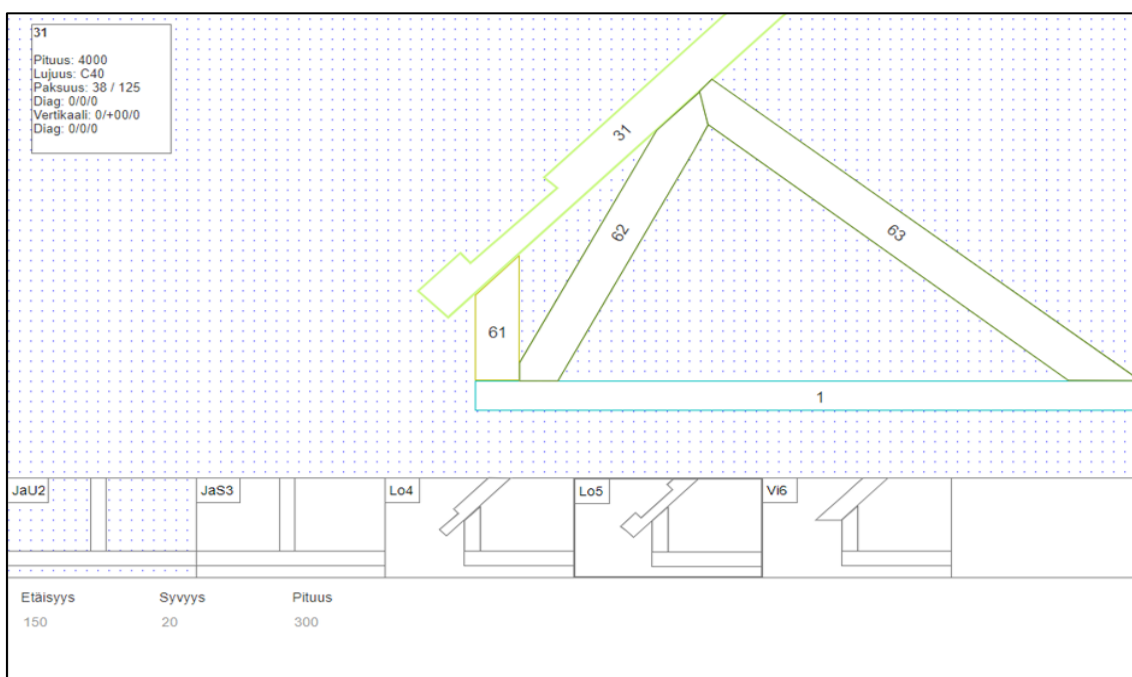
Edellä on esitetty kaksi tapaa tehdä lyhty ja hieman muitakin lisäyksiä. Erityisesti kiinnosti käyttäjien mielipiteet lyhtyjen rakentamisen logiikasta ja siihen liittyvistä muut muutoksista; pystyvätkö he näkemään itsensä käyttämässä ohjelmaa tällä tavalla. Seuraavaksi käydään läpi lovien ja viisteiden tekeminen.

Kuvassa 14 käyttäjällä on valittuna yläpaarre 31 ja hän on valikosta tehnyt valinnan ”Lo4” tarkoittaen lovea ja numero on näppäinkomentoa varten. Kuvassa valinta näkyy korostettuna, että käyttäjä muistaa valintansa. Käyttäjä on syöttänyt jo mitat lovealle. Kuten kuvan alareunassa näkyy, lovealle on annettu mitat: syvyys 20mm ja pituus 300mm. Ohjelma on piirtänyt sen mukaisen loveuksen. Syvyys lasketaan kapulan yläreunasta ja pituus kapulan päästä. Tästä piti tarkistaa käyttäjiltä termien oikeellisuus ja tarvitsiko tähän lisätä mitään vai tehdäänkö loveukset tällä tavalla.



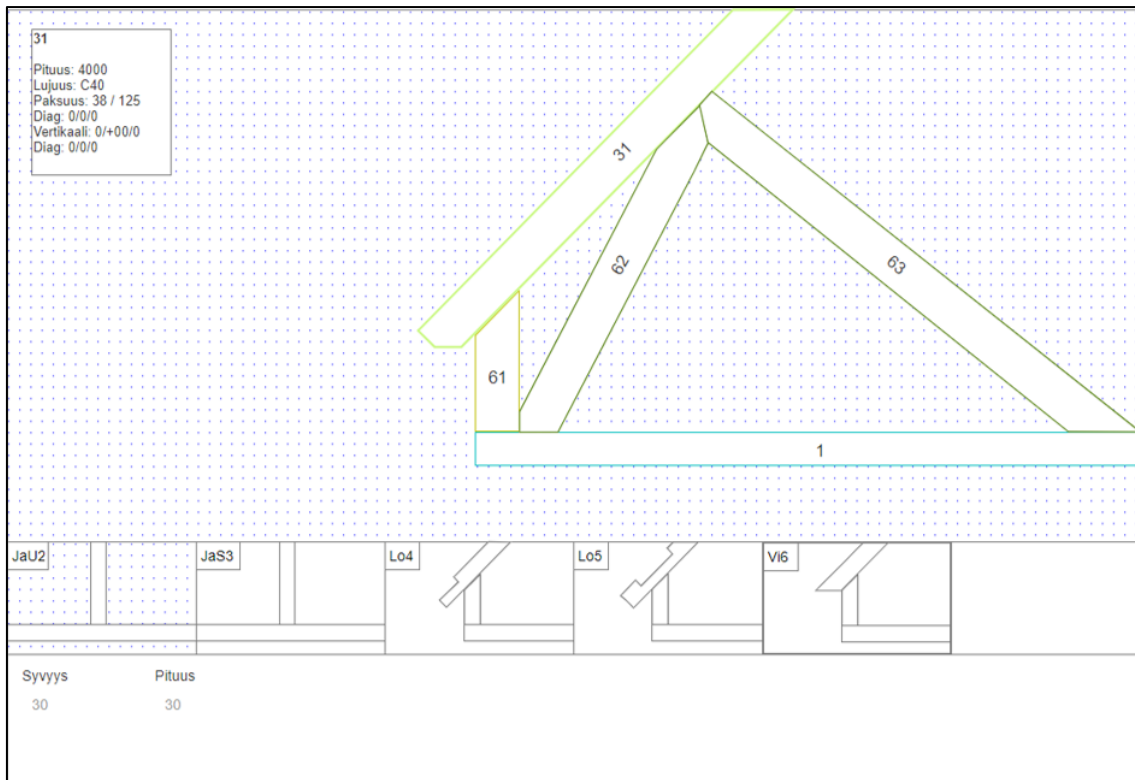
Kuva 14. Loveuksen tekeminen, tapa yksi.

Kuvassa 15 esitellään toinen tapaus tehdä loveuksia. Tässäkin on valittuna yläparre 31, mutta valikosta on valittu "Lo5", jolla tehdään hieman erinäköinen loveus. Tässä tarkoituksena on saada lovi syvemmälle kapulaa eikä aivan päähän asti. Käyttäjä on ensin syöttänyt etäisyyden, mistä kohtaa lovea aletaan tehdä ja sen jälkeen hän on syöttänyt loveuksen mitat. Tämä toimii kuvan 14 kanssa melkein samalla tavalla, mutta tässä syötetään vain etäisyys ensin.



Kuva 15. Loveuksen tekeminen, tapa kaksi.

Kuvassa 16 on esimerkki viisteiden teosta. Tässäkin on valittuna yläparre 31 ja valintana on ”Vi6” tarkoittaen viistettä ja numero on taas näppäinkomentoa varten. Tässä on hyvin samankaltainen logiikka kuin kolouksien teossa. Syvyys annetaan ensin ja se lasketaan kapulan yläreunasta, seuraavaksi käyttäjä antaa pituuden ja se lasketaan kapulan päädyistä. Erona on, että viiste tehdään kapulan alapuolelle. Kuvassa käyttäjä on syöttänyt mitat 30mm ja 30mm, joten ohjelma on piirtänyt hyvin pienen viisteen. Tästäkin oli tärkeä kysyä käyttäjien mielipidettä.



Kuva 16. Viisteen tekeminen.

Edellä on kuvattu prototyypin ydintehtävät ja näistä halusin saada erityisesti palautetta. Nämä kaikki muutokset nousivat esille edellisissä haastatteluissa ja nyt tarkoituksena oli selvittää, olivatko ratkaisut käyttäjiä miellyttäviä. Haastateltaville näytettiin ensin koko prototyyppi läpi ja sen jälkeen palattiin näihin kohtiin katsomaan niitä tarkemmin. Haastattelu oli avoin, joten siihen jäi tilaa myös muille kommenteille, mutta nämä tuli ainakin käydä läpi.

4.6.2. Toteutus

Haastatteluja varten otin taas yhteyttä samaan kymmeneen henkilöön kuin aikaisemmassakin haastattelussa. Tällä kertaa sain vastauksen neljältä ja heidän kanssaan saatiin so-

vittua haastattelut. Kolme haastateltavaa oli myös osallistunut jo aikaisempiin haastatteluihin. Jokainen haastattelu kesti noin puoli tuntia. Kaikki haastattelut tehtiin etänä käyttäen Microsoft Teamsia ja niistä nauhoitettiin ääni sekä kuva.

4.6.3. Analyysi/tulokset

Haastattelujen analyysi aloitettiin heti haastattelun päätyttyä. On tärkeää kirjoittaa heti muistiinpanoja haastatteluista myöhempää käyttöä varten, kun haastattelusta on kulunut aikaa, on vaikea muistaa miten ne ovat menneet. Myös jo haastattelujen aikana kirjoitettiin muistiinpanoja. Näitäkään haastatteluja ei litteroitu vaan niitä käytiin myöhemmin läpi ja kirjattiin huomiot ylös. Haastatteluissa huomioitiin jo edeltävien haastateltavien kommentteja, ja jos haastateltavalla oli vaikea keksiä kommentoitavaa, häntä autettiin edellisten huomioilla. Tämä tehtiin siksi, että saatiin samoista asioista palautetta eri käyttäjiltä.

Tulokset käydään läpi samassa järjestyksessä kuin prototyypin eteneminen kuvattiin luvussa 4.6.1. Lopuksi mainitaan vielä irrallisia kommentteja tuloksista.

Kuvassa 7 näkyi kehittynyt prototyyppi. Taustalle oli lisätty ruudukko, jonka tarkoituksena on auttaa käyttäjää, jos hän käyttää hiirtä enemmän näppäimistön sijaan. Käyttäjillä oli vaikeaa hahmottaa sen käyttämistä itse ohjelmassa. Heidän mielestään se on mahdollisesti hyvä lisä, mutta siihen täytyy olla vaihtoehto laittaa se päälle tai pois päältä haluttaessa. Varsinkin jos käyttää vain näppäimistöä sille ei ole tarvetta. Prototyyppiin lisätty infolaatikko, joka sisältää kapulan tietoja, nähtiin hyvänä ideana. Täytyy kuitenkin miettiä, mitä tietoja siihen halutaan. Ehdotettiin esimerkiksi, että siinä näkyisi vasemman ja oikean pään kulmien mitat. JaU ja JaS merkintöjä pidettiin kelvollisena. Niitä ei pidetty kovin kriittisinä tekijöinä, mutta niihinkin on tärkeä saada selkeät merkinnät, selkeämmät kuin ne nyt ovat, varsinkin jos niitä käytetään ristikon sisällä olevissa osissa enemmän. Tämä ei kuitenkaan oli prioriteetti, sillä ne kyllä saadaan tehtyä näilläkin termeillä yrittämisen ja erehtymisen kautta.

Kuvassa 8 rakennettiin lyhtyä. Siinä pidettiin, että on mahdollisuus antaa kaikki mitat myös näppäimistöllä, että ei olla sidottu pelkästään käyttämään hiirtä, mutta hiiren käyttö on hyvä lisä. Haastateltavien mielestä tämä oli hyvin selkeä tapa tehdä lyhtyä ja ehdottomasti hyvä lisä. Kaikki mitat tulee antaa alapaarteen alareunasta, tämä oli hyvä tarkennus. 90-asteen kulman suora piirtäminen nähtiin erittäin kätevä. On myös hyvä, että hiiren viemällä uusien kapuloiden päälle nähdään heti niiden tiedot ja tiedetään, tarvitseeko niitä muuttaa.

Kuvissa 9 ja 10 kuvattiin toinen tapa tehdä lyhty. Tekstille ”Anna pituus alareunasta” ei tullut vaihtoehtoja, sillä käyttäjät eivät pidä tällaisia asioita niin tärkeänä, enemmän heitä kiinnostaa, että miten nämä toimivat. Myöhemmin voi kiinnittää huomioita enem-

män nimeämisisille ja esteettisyyteen. Tässä vaihtoehdot JaU ja JaS olivat hieman epäselkeämmät käyttäjien mielestä. Tässä nousi esiin, että pitää saada selkeät merkinnät niille. Hyvänä lisänä nähtiin, että reunimmaista kapulaa oli helppo kasvattaa ja siihen oli helppo lisätä toinen kapula. Tässäkin nousi esille, että mitat annetaan aina alapaarteiden alareunasta, eikä koskaan yläreunasta. Kun käyttäjät laittavat kapuloita linjaan, he syöttävät niille tarkat mitat, joten tässä kapulan siirtely hiiren avulla ja apuviivojen kanssa linjaan laittaminen oli vaikea hahmottaa. He ovat niin tottuneita syöttämään vain tarkat mitat ohjelmalle. Varsinkin apuviivat nähtiin tarpeellisena, jos laitetaan hiirellä osia linjaan. Nähtiin selkeänä logiikkana, että kapulan keskiosasta klikkaamalla saadaan se haltuun ja voidaan siirrellä sitä ja päistä klikkaamalla voidaan kasvattaa kokoa.

Kuvissa 11, 12 ja 13 on esitelty lyhdyn alle jäävien kapuloiden muokkausta. Niiden muokkausta pidettiin todella kätevänä. Eri osien valinta ja niiden koon muuttaminen vaikuttaa hyvältä muutokselta, sitä kaivataan nykyiseen ohjelmaan. Liukusäätimestä pidettiin myös paljon, sillä käyttäjistä on tuntunut, että valmiita vaihtoehtoja ei ole aina tarpeeksi. Sen kanssa pitää myös olla tarkkana, ettei lisätä liikaa vaihtoehtoja tai muuten ne hukkuvat. Yksi ehdotus oli, että yhtä valintaa voisi klikata hiiren oikealla näppäimellä ja sen alta löytyisi lisää samankaltaisia vaihtoehtoja, joista valita itselle sopiva. Myös kapulan pään muokkaaminen vaikutti hyvältä lisäykseltä, varsinkin tässä asiayhteydessä; hiirellä siirtämisen yhteydessä helpottaa todella paljon, että voi valita valmiin pään. Kuvassa 13 näkyy kapula 62 toinen pää muokattuna tasaiseksi ja koko kapula on kääntynyt sen mukaan. Tämä vaikutti hyvältä lisäykseltä, mutta tässä on hyvä myös käyttää apuna ensimmäisessä prototyypissä esiteltyä lukko -ominaisuutta, eli voidaan lukita toinen pää paikalleen, etteivät sen mitat muutu.

Kuvissa 14, 15 ja 16 tehtiin loveuksia ja viisteitä. Ensimmäisessä haastattelussa tuli ilmi, että termit olivat aluksi väärät, mutta ne korjattiin heti myöhempiä haastatteluita varten, jotta termeihin ei menisi turhaa aikaa. Loveukset vaikuttivat yksinkertaisilta ja hyviltä lisäyksiltä. Haastatteluissa tuli ilmi, että loveukset saatetaan tehdä välillä myös viistoon ja alapuolelle kapulaa. Myös loveuksia saatetaan tehdä useampi samaan kapulaan. Valintoihin on siis hyvä tehdä lisäys, jolla lovet saadaan myös alapuolelle, niitä voidaan tehdä useita ja loveuksille täytyy pystyä määrittelemään kaksi eri syvyyttä, lähtösyvyys ja loppusyvyys. Viisteestä tuli palautetta, että se olisi hyvä saada lähtemään valmiiksi kahden kapulan leikkauskohdasta, eli ettei tarvitsisi määritellä mitalla muuta kuin alkusyvyys. Viisteissä oli hyvä, että sen syvyys annetaan kapulan yläosasta, sillä välillä heillä on tarkka mitta sitä varten, koska siihen laitetaan niin sanottu otsalauta.

Yleisesti ottaen kaikista lisäyksistä pidettiin, ja jos nämä muutokset saadaan uuteen ohjelmaan niin siitä tulee selkeästi parempi, varsinkin uusille käyttäjille. Nämä muutokset vaikuttavat paljon näppärämmiltä kuin tämän hetkinen ohjelma pitää sisällään. Prototyy-

pissä oli paljon lisäyksiä ohjelmaa varten, joten hirveästi lisäpyyntöjä ei tullut, mutta kuitenkin muutamia. Jokaiselle kapulalle haluttaisiin mahdollisuus lisätä eri kuormia, esimerkiksi voisi olla valintaruutu tyyppinen tapa lisätä kuormia yksittäisille kapuloille. Myös eri kapuloiden muotoa tulisi pystyä muuttamaan helposti, varsinkin kun rakennetaan lyhtyjä. Tällä tarkoitetaan, että sisäpalkista voitaisiin tehdä yläpaarre, tällä on vaikutuksia ohjelman laskentamalliin ja siksi se olisi tärkeä. Välillä myös tarvitsee jatkaa kapulaa, jos puutavara ei ole tarpeeksi pitkää. Tätä varten voisi lisätä ominaisuuden, mutta tapaus on kuitenkin harvinainen. Haastatteluissa myös korostui, että mitään ominaisuuksia ei saa poistaa, mutta lisäominaisuudet ovat plussaa.

4.6.4. Johtopäätökset

Prototyypin oltiin erittäin tyytyväisiä. Siihen ei tullut enää paljoa kehitysehdotuksia, mikä on hyvä asia. Kaikkia muutoksia koettiin tarpeellisina ja edesauttavina. Kuitenkaan kaikki prototyypissä esitetyt muutokset eivät ole lukkoon lyötyjä ja työtä kannattaisi vielä jatkaa. Ominaisuuksien hienosäätämistä tarvittaisiin. Kun ne on tehty, voidaan keskittyä esteettisyyteen ja termeihin. Suunnittelussa nousi paljon ongelmakohtia esille, joihin saatiin hyviä ratkaisuja. Käyttäjät ovat tyytyväisiä tämän hetkisiin ehdotuksiin ja niiden toteuttamisella pystytään nostamaan tehokkuutta.

5. Tutkimuksen johtopäätökset ja keskustelua

Tutkimuksella oli tarkoituksena selvittää käyttäjakeskeisen suunnittelun vaikutuksia kehitysprosessiin ja varsinkin sen lopputulokseen käyttäjien näkökulmasta. Tutkimuksessa pyrittiin selvittämään käyttäjakeskeisen suunnittelun hyötyjä ja haittoja.

Käyttäjakeskeinen suunnittelu valittiin kehitystavaksi jo heti projektin alussa. Koska kehityskohtena oleva ohjelma on hyvin vanha ja sen käytön opettelu vaatii pitkän ajan, niin käyttäjakeskeinen suunnittelu oli paras ratkaisu. Käyttäjakeskeinen suunnittelu kattaa paljon asioita, jotka tulee ottaa huomioon jo projektin suunnitteluvaiheessa. Käyttäjakeskeiselle suunnittelulle on monia eri suunnitteluperiaatteita ja ne ovat kehittyneet vuosien saatossa. Monet suunnittelun periaatteet ovat hyvin samankaltaisia, ja suurimmat erot niissä ovat käytetyissä termeissä. Ennen projektin käynnistämistä suunnitteluperiaatteiksi valittiin standardin ISO 9241-210 [2019] mukaiset periaatteet. Tämä valinta tehtiin, koska ne ovat hyvin tuoreet, ne pitävät käyttäjän keskiössä ja ne korostavat käyttäjän tärkeyttä koko suunnitteluprosessissa. Prosessin on myös oltava iteratiivinen. Kehitysprosessissa oli myös vahvasti mukana käytettävyys, joka on tärkeä osa käyttäjakeskeisyyden onnistumista.

Käyttäjakeskeistä suunnittelua korostettiin koko tutkimuksen ajan. Käyttäjien kanssa tutustuttiin ohjelman käyttöympäristöön, tehtiin haastatteluja, joissa kuultiin heidän tarpeitaan ja vaatimuksia. Prototyyppiä kehitettiin vastaamaan haastatteluissa esille nousseisiin tarpeisiin. Prototyypin kanssa jatkettiin haastatteluja, jotta käyttäjät pääsivät ilmaisemaan mielipiteensä muutoksista.

Kehitysprosessin alkuvaiheissa käyttäjät olivat kiinnostuneita mahdollisuudesta parempaan käyttöliittymään, mutta myös hieman varautuneita. He ovat käyttäneet nykyistä ohjelmaa todella pitkään ja he eivät halua siihen liian radikaaleja muutoksia. Ensimmäiseen haastatteluun käyttäjät osallistuivat mielellään ja varasivat siihen aikaa kiireisistä työpäivistään. Ohjelmaan on kaavailtu muutoksia jo useamman vuoden ajan, mutta prosessi on ollut hidas. Käyttäjät toivovat ja tarvitsevat muutoksia ohjelmaan.

Tämän kehitysprosessin puitteissa voidaan todeta, että käyttäjakeskeinen suunnittelu vaikutti lopputulokseen positiivisesti. Prototyyppi otettiin todella hyvin vastaan ja käyttäjät olivat tyytyväisiä sen ratkaisuihin. Heidän mielestään tutkimuksessa esitetyillä muutoksilla uudesta ohjelmasta saadaan hyvin kätevä ja se tulee varmasti auttamaan heitä ja varsinkin uusia käyttäjiä. Tässä onnistuttiin, koska käyttäjiä kuunneltiin koko prosessin ajan ja heidän mielipiteensä otettiin huomioon. Haastatteluissa käyttäjien oli helppo ilmaista mielipiteensä ja mielestäni he olivat tilanteessa rentoja. Haastattelujen metodit, puolistrukturoidut ja avoimet haastattelut, olivat tehokkaita menetelmiä ja toimivat hyvin tässä tutkimuksessa. Haastatteluissa ilmeni paljon ongelma-alueita, mutta ikävä kyllä tämän tutkimuksen puitteissa niihin kaikkiin ei pystytty kehittämään ratkaisuja. Kaikki

haastateltavat osasivat esittää asiansa niin, että tutkija ymmärsi ongelmat, vaikka ohjelman käyttö ei ole tuttua. Tämä helpotti huomattavasti prototyypin suunnittelua ja se auttoi saamaan siitä enemmän käyttäjien vaatimusten näköisen.

Tutkimukseen olisi saatu enemmän näkökulmia, jos tutkijoina olisi ollut monialainen tiimi. Varsinkin prototyypin ideoinnissa tiimistä on paljon apua. Myös tällaisissa projekteissa saa hyvin vertaistukea kollegoilta ja heidän kanssaan on hyvä keskustella ideoista ja suunnitteluratkaisuista. Tutkimusta olisi myös pystytty laajentamaan. Tämä on hyvä huomioida, kun suunnittelee eri kokoisia projekteja.

Prototyypin haastatteluihin oli hieman vaikea saada osallistujia, mutta siihen varmasti vaikutti myös tämä erikoistilanne johtuen koronaviruspandemiasta. Kuitenkin haastatteluun osallistuvilta käyttäjiltä sai todella laadukasta palautetta. Niissä paljastui erilaisia ongelmia, jota tutkija ei ollut tiedostanut. Prototyypistä ei tullut vielä täydellistä ja siinä on vielä puutteita, jotka tulee huomioida ennen mahdollista käyttöönottoa. Prototyypin kehitystä kannattaa vielä jatkaa, mutta nyt siinä voidaan keskittyä pienempiin asioihin kuten esteettisyyteen ja termistöön.

Koko kehitysprosessia voidaan pitää onnistuneena. Tarkoituksena oli löytää tehokkuutta lisääviä tekijöitä käyttöliittymään ja saada niistä lisäarvoa käyttäjille. Tämän voidaan katsoa onnistuneen. Käyttäjille saatiin suunniteltua muutoksia, jotka tulisivat edistämään tehokkuutta ja lisäämään käytön mielekkyyttä. Käyttäjäkeskeinen suunnittelu toimi erittäin hyvin tässä projektissa. Käyttäjät olivat pieni, rajattu ryhmä, joilta sai laadukasta palautetta.

Käyttäjäkeskeisen suunnittelun haittoina voidaan nähdä tässä projektissa käyttäjien tietämättömyys käyttöliittymäsuunnittelusta. Heille ei välttämättä ollut selvää, mitä kaikkea käyttöliittymäsuunnitteluun kuuluu. Tässä projektissa käyttäjäkeskeisellä suunnittelulla löydettiin kuitenkin konkreettisia ratkaisuja käyttäjien ongelmiin.

Vaikka tutkimus oli onnistunut, uuden ohjelman suunnittelu vaatii vielä jatkokehitystä. Iteraatiokierroksia on vielä hyvä jatkaa, sillä kaikkia ongelmia ei saatu vielä ratkotua. Käyttäjäkeskeisiä suunnitteluperiaatteita kannattaa ehdottomasti jatkaa mahdollisessa jatkokehityksessä. Valitettavasti tämän tutkimuksen puitteissa iteraatiokierroksia ei ollut mahdollista jatkaa.

Käyttäjäkeskeinen suunnittelu ja sen korostaminen oli ehdottomasti oikea ratkaisu tähän tutkimukseen. Käyttäjille saatiin heitä miellyttäviä suunnitteluratkaisuja käyttöliittymään, joista he hyötyvät. Valitettavasti huonona puolena on, että kaikkia ongelmia ei keritty ratkomaan. Mutta jo näillä muutoksilla pystytään lisäämään käytön mielekkyyttä ja tehokkuutta.

6. Yhteenveto

Tutkielman alussa käytiin läpi teoreettista taustaa käyttäjäkeskeisyyden eri näkökulmista. Tutkielman pääpainona oli käyttäjäkeskeinen suunnittelu ja siihen liittyvä prosessi. Pääpainon katsomaan olevan ISO 9241-210 [2019] standardin periaatteissa, näkemystä laajennettiin muiden tieteellisten tutkimuksien avulla.

Tutkielman seuraavana kohtana oli varsinainen tutkimusmenetelmä ja kehityksen kohteena oleva yritys ja ohjelma. Tutkimuksessa selvitettiin, miten käyttäjäkeskeistä suunnittelua korostaen saadaan käyttäjien mielipiteet lopputulokseen näkyviin. Tutkimusmenetelmäksi valittiin toimintatutkimus, joka mahdollistaa käyttäjäkeskeisen suunnittelun suorittamisen tieteellisestä näkökulmasta. Tutkimuskohteena oli Swecon Wode ohjelma, joka on rakennussuunnitteluohjelma.

Tutkimusongelmana oli ohjelman käyttöliittymän kehittäminen, jotta siitä saadaan tehokkaampi käyttäjille, jotka ovat käyttäneet ohjelmaa jo useita vuosia. Käyttäjät otettiin kehittämisprosessin keskiöön, että pystyttiin keskittymään käyttäjän tarpeisiin. Tutkimusongelmaa lähdettiin purkamaan toimintatutkimuksen avulla. Toimintatutkimus suoritettiin kolmella iteraatiokierroksella. Ensimmäisellä kierroksella tutkija tutustui ohjelman käyttöön, käyttäjien kokemuksiin ja mielipiteisiin ohjelmasta. Seuraavat kaksi kierrosta olivat haastatteluja prototyypin pohjalta, joissa keskityttiin konkreettisemmin ohjelman ongelmiin ja niiden ratkaisuihin. Haastattelut olivat puolistrukturoituja ja avoimia haastatteluja. Haastattelut olivat erittäin toimivia metodeja tällaiseen tutkimukseen, jossa käyttäjän mielipide on tärkeä.

Tutkimuksen tuloksina voidaan todeta, että projekti oli onnistunut ja sen avulla käyttäjille saatiin tuotettua lisäarvoa. Lisäarvona käyttäjille saatiin suunniteltua ratkaisuja, jotka helpottavat ja nopeuttavat heidän työskentelyään. Käyttäjät toivoivat, että nämä suunnitellut muutokset saadaan uuteen ohjelmaan. Tärkeää oli, että käyttäjät itse saivat vaikuttaa lopputulokseen.

Jatkotutkimuksena jatkaisin prototyypin kehittämistä ja tekisin siitä vielä yksityiskohtaisemman. Sen jälkeen voisi lähteä etsimään ratkaisuja ohjelman muihin ongelma-alueisiin ja työstämään muita työvaiheita ohjelmassa.

Lähteet

- Ari Alamäki ja Amir Dirin. 2015. The Stakeholders Of A User-Centred Design Process In Mobile Service Development. *International Journal of Digital Information and Wireless Communications*, 5(4), 270–284.
- Abras Chadia, Maloney-Krichmar Diane and Preece, Jenny. 2004. User-Centered Design. In: *Bainbridge, W. Encyclopedia of Human-Computer Interaction*. Thousand Oaks: Sage Publications. (in press)
- Kursat Cagiltay, Eun-Ok Baek,, Elizabeth Boling ja Theodore Frick. 2008. *User-centered design and development*.
- Cristopher Duncan. 2013. *Unite the Tribes Ending Turf Wars for Career and Business Success*. Dordrecht: Springer.
- Francisco-Javier Ferrández-Pastor, Juan-Manuel García-Chamizo, Mario Nieto Hidalgo, ja José Mora-Martínez. 2017. User-Centered Design of Agriculture Automation Systems Using Internet of Things Paradigm. *Ubiquitous Computing and Ambient Intelligence Lecture Notes in Computer Science*, 56–66.
- Andrea K. Graham, Jennifer E. Wildes, Madhu Reddy, Sean A. Munson, C. Barr Taylor ja David C. Mohr. 2019. User-centered design for technology-enabled services for eating disorders. *International Journal of Eating Disorders*, 52(10), 1095–1107.
- Ray Grott. 2019. Assistive technology design and fabrication. *Assistive Technology Service Delivery*, 145–158.
- Jan Gulliksen, Bengt Göransson, Inger Boivie, Jenny Persson, Stefan Blomkvist ja Åsa Cajander. 2005. Key Principles for User-Centred Systems Design. In: *Seffah A., Gulliksen J., Desmarais M.C. (eds) Human-Centered Software Engineering — Integrating Usability in the Software Development Lifecycle*. Human-Computer Interaction Series, vol 8. Springer, Dordrecht
- Gillian R. Hayes. 2011. The relationship of action research to human-computer interaction. *ACM Transactions on Computer-Human Interaction*, 18(3), 1–20.
- Sirkka Hirsjärvi ja Helena Hurme. 2000. *Tutkimushaastattelu: Teemahaastattelun teoria ja käytäntö*. Helsinki: Yliopistopaino.
- Karen Holtzblatt, Jessamyn Burns Wendell ja Shelley Wood. 2005. *Rapid Contextual Design : A How-to Guide to Key Techniques for User-Centered Design*. Morgan Kaufmann.
- SFS-EN ISO 9241-210:2019 Ergonomics of human-system interaction. Part 210: Human-centred design for interactive systems. 2019.
- SFS-EN ISO 9241-11:2018 Ergonomics of human-system interaction. Part 11: Usability: Definitions and concepts. 2018.
- SFS-EN ISO 9241-110:2006 Ergonomics of human-system interaction. Part 110: Dialogue principles. 2006.

- Jane Ledbury. 2018. Design and product development in high-performance apparel. *High-Performance Apparel*, 175–189.
- Dennis List. 2006. Action research cycles for multiple futures perspectives. *Futures*, 38(6), 673–684.
- Mark Jones Lorenzo. 2017. *Endless Loop: The History of the BASIC Programming Language (Beginner's All-purpose Symbolic Instruction Code)*. CreateSpace Independent Publishing Platform, North Charleston, SC, USA.
- Travis Lowdermilk. 2013. *User-Centered Design: A Developer's Guide to Building User-Friendly Applications*
- Jakob Nielsen. 1994. *Usability engineering*. San Francisco, CA: Morgan Kaufmann Publishers.
- Donald A. Norman ja Stephen W. Draper. 1986. *User-Centered System Design: New Perspectives on Human-Computer Interaction*. Lawrence Earlbaum Associates, Hillsdale, NJ.
- Donald Norman. 1988. *The psychology of everyday things*. New York: Doubleday.
- Kent Norman ja Jurek Kirakowski. 2018. *The Wiley handbook of human computer interaction*. Hoboken, NJ: John Wiley & Sons.
- Ji-Ye Mao, Karel Vredenburg, Paul W. Smith ja Tom Carey. 2005. The state of user-centered design practice. *Communications of the ACM*, 48(3), 105–109.
- Niamh Mcnamara ja Jurek Kirakowski. 2006. Functionality, usability, and user experience. *Interactions*, 13(6), 26.
- Anita Saaranen-Kauppinen ja Anna Puusniekka. 2006. KvaliMOTV - Menetelmäopetuksen tietovaranto. Tampere: Yhteiskuntatieteellinen tietoarkisto. <<https://www.fsd.tuni.fi/menetelmaopetus/>>. (Viitattu 22.04.2020.)
- Syed Ghulam Sarwar Shah ja Ian Robinson. 2007. Benefits of and barriers to involving users in medical device technology development and evaluation. *International Journal of Technology Assessment in Health Care*, 23(1), 131–137.
- Ben Shneiderman. 1987. *Designing the user interface strategies for effective human-computer interaction*. Reading, MA.
- Gerald I. Susman ja Roger D. Evered. 1978. An Assessment of the Scientific Merits of Action Research. *Administrative Science Quarterly*, 23(4), 582.
- Tarja Tiainen, Johanna Aittoniemi, Ilkka Haukijärvi ja Tiina Yli-Karhu. 2015. Toimintatutkimus tietojenkäsittelytieteen tutkimuksessa. *Informaatiotieteiden raportteja* 38/2015. Tampereen yliopisto, Tampere, 2015, 1-30
- Karel Vredenburg, Ji-Ye Mao, Paul W. Smith ja Tom Carey. 2002. A survey of user-centered design practice. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems Changing Our World, Changing Ourselves - CHI 02*.

- Ashley Williams. 2009. User-centered design, activity-centered design and goal-directed design: A review of three methods for designing web applications. *Proceedings of the 27th ACM international conference on Design of communication, 2009, Bloomington, Indiana, USA*.
- Chauncey Wilson. 2014. *Interview techniques for Ux practitioners: a user-centered design method*. Amsterdam: Elsevier.

Haastattelukysymykset

Taustatiedot

1. Kuinka kauan olet työskennellyt tämän ohjelman parissa?
2. Arvio omasta tietoteknisestä osaamisestasi? 1 = erittäin huono ... 5 = erittäin hyvä

Ohjelman käyttö

1. Kuinka usein käytät ohjelmaa?
2. Paljonko käytät aikaa ohjelman parissa?
3. Kauanko kesti opetella käyttämään ohjelmaa?

Ohjelman arviointi

1. Koetko ohjelman käytän helpoksi?
2. Millaisena koet käytön vaatiman aikamäärän? Onko käytön tehokkuus hyvä?
3. Onko ohjelman käytön muistettavuus helppoa? Tarvitsetko tukea muistamiseen? Tarvitsetko lisää ohjeita muistamiseen, esim. selitteet funktionäppäimille?
4. Kuinka helposti ohjelmassa tapahtuu virheitä? Haluaisitko selkeämmät virheilmoitukset?
5. Onko käyttö miellyttävää?
6. Mistä tykkäät eniten ohjelmasta?
7. Mistä tykkäät vähiten ohjelmasta?
8. Onko ohjelmassa jotain toimintoa mitä käytät selkeästi eniten?
9. Millaisia puutteita ohjelmassa on? Mitä haluaisit lisättävän?
10. Toimiiko ohjelman navigointi hyvin? Millaisia muutoksia?
11. Missä sinun mielestäsi ohjelman käyttöliittymän tulisi parantua?
12. Mitä muutoksia haluaisit käyttöliittymään? Esimerkiksi funktionäppäinten paikat, fonttien koko, kuvien selkeys...
13. Käytätkö vanhaa ristikköä apuna kun lähdet suunnittelemaan uutta?
14. Haluaisitko ohjelmaan sisäisen laskimen? Koetko sitä tarpeelliseksi?