

Jari Vähkyrä

**Käyttöliittymäkehityksen periaatteet ja niiden
soveltaminen 3D-suunnitteluohjelmistossa**

Tietotekniikan pro gradu -tutkielma

26. syyskuuta 2024

Jyväskylän yliopisto

Informaatioteknologian tiedekunta

Tekijä: Jari Vähkyrä

Yhteystiedot: jari.a.vahkyra@student.jyu.fi

Ohjaaja: Raino Mäkinen

Työn nimi: Käyttöliittymäkehityksen periaatteet ja niiden soveltaminen 3D-suunnitteluohjelmistossa

Title in English: UI design principles and their application in 3D design software

Työ: Pro gradu -tutkielma

Opintosuunta: Ohjelmisto- ja tietoliikennetekniikka

Sivumäärä: 40+0

Tiivistelmä: Käyttöliittymäkehityksen tueksi on jo pitkään tarjottu erilaisia periaatelistauksia, joiden tarkoituksena on ollut toimia yleispätevänä ohjenuorana toimivien käyttöliittymien suunnittelemiseksi ja kehittämiseksi. Periaatteita on kritisoitu vuoroin liian laveiksi, vuoroin liian suppeiksi. Tutkielmassa on tarkoituksena selvittää, mitkä ovat kirjallisuudessa esitetyt käyttöliittymien kehityisperiaatteet ja -käytänteet ja miten niitä sovelletaan käytännön kehitystyössä. Tätä tarkoitusta varten tehdään kirjallisuuskatsaus yleisistä käyttöliittymäperiaatteista ja käytänteistä, joita soveltaen ja hyödyntäen toteutetaan käyttöliittymäsuunnitelma- artefakti 3D-suunnitteluohjelmistoon.

Avainsanat: Suunnittelututkimus, käyttöliittymäkehitys, kehityisperiaatteet, 3D-suunnittelu, pro gradu -tutkielmat

Abstract: Several different design principle suggestions for user interface design have already been introduced since the 1970s. The aim of these principles and guidelines is to improve the quality of the design and inform the developer of good design. The principles have been criticized being both too general and overly specialized. This thesis sets out to clarify what general UI design principles and practices are found in the research literature and how they can be applied in the design of user interfaces. This is accomplished via a systematic literature review after which a UI design artefact is constructed by applying the collected design principles.

Keywords: Design science, UI design principles, GUI, 3D design applications

Kuviot

| | |
|---|----|
| Kuvio 1. Ensimmäinen iteraatio | 21 |
| Kuvio 2. Toinen iteraatio: kiertonäkymä | 23 |
| Kuvio 3. Toinen iteraatio: leveysnäkö | 25 |
| Kuvio 4. Toinen iteraatio: videovälilehti | 27 |
| Kuvio 5. Kolmas iteraatio | 30 |

Taulukot

| | |
|--|----|
| Taulukko 1. Tietokantahauissa käytetyt hakulausekkeet | 8 |
| Taulukko 2. Suunnitteluperiaatteita sisältäneet artikkelit. | 9 |
| Taulukko 3. Katsauksella kootut suunnitteluperiaatteet..... | 11 |
| Taulukko 4. 3D-suunnitteluun erikoistuneet suunnitteluperiaatteet. | 12 |
| Taulukko 5. Käyttöliittymäkehityksen käytänteet. | 12 |
| Taulukko 6. Käyttöliittymäsuunnitteluprosessin käytänteet. | 13 |
| Taulukko 7. 3D-suunnitteluun erikoistuneet käyttöliittymäkehityksen käytänteet. | 13 |

Sisällys

| | | |
|---|--|----|
| 1 | JOHDANTO | 1 |
| 2 | TEORIAOSA | 3 |
| | 2.1 Kirjallisuuskatsaus | 3 |
| | 2.2 Suunnittelututkimusteoriaosuus | 4 |
| 3 | KIRJALLISUUSKATSAUS..... | 7 |
| | 3.1 Kirjallisuuskatsauksen toteutus ja tulokset..... | 7 |
| | 3.2 Yhteenveto kirjallisuudessa mainituista periaatteista..... | 14 |
| 4 | SANOISTA TEOIKSI – SUUNNITTELUTUTKIMUSOSUUS..... | 17 |
| | 4.1 Artefaktin suunnittelu ja toteutus | 17 |
| | 4.2 Vaatimusmäärittely | 18 |
| | 4.3 Artefaktin toteutus..... | 19 |
| 5 | YHTEENVETO JA POHDINTA | 31 |
| | LÄHTEET | 35 |

1 Johdanto

Tietotekniikka on läsnä ihmisten arjessa tiiviimmin kuin koskaan aikaisemmin. Käytämme tietokoneita ja älylaitteita ja olemme vuorovaikutuksessa ohjelmien ja ohjelmistojen kanssa päivittäin niin työssä, koulussa kuin vapaa-ajalla. Erilaiset digitaaliset työskentelyympäristöt ja ohjelmistot ovat muuttuneet viime vuosikymmenien aikana itsestäänselvyydeksi ja niiden käytön varaan on muotoutunut suuri osa työelämän toiminnoista. Vuorovaikutukseen teknologian kanssa liittyy kuitenkin paljon sellaista, joka täytyy ottaa huomioon, jotta tietoteknisten työkalujen käyttö olisi sujuvaa ja tehokasta. Tähän seikkaan ovat heränneet niin laitevalmistajat kuin ohjelmistokehittäjät ja käyttökokemuksen suunnittelu onkin olennainen osa nykyistä tuotekehitystä.

Käyttöliittymien arviointiin käytetyt kriteerit ovat helppo tapa kuvailla, millainen käyttöliittymän tulisi olla. Tällaiset määreet eivät kuitenkaan kerro, miten niiden mukainen käyttöliittymä saavutetaan. On esimerkiksi helpompi sanoa, että käyttöliittymän tulisi olla helpokäyttöinen, kuin kertoa miten kehitetään helpokäyttöinen käyttöliittymä. Vaikka eri järjestelmillä on eri käyttäjät, ihmisten tavoissa käyttää ohjelmistoja on tiettyjä yhtäläisyyksiä, joiden tutkiminen on johtanut kirjallisuudessa esitettyihin yleisiin kehityisperiaatteisiin käyttöliittymien suunnittelemiseksi. Nämä yleisemmän tason periaatteet kattavat parhaat käytännöt käyttöliittymäkehityksen alalla ja parantavat lopputuotteiden käytettävyyttä. (s. 47 Ruiz, Serral ja Snoeck 2021) Ne voivat toimia kehityksessä ohjenuorina, joita noudattamalla ohjelman käytettävyyteen voidaan vaikuttaa kustannustehokkaasti varhaisessa vaiheessa kehitysprosessia.

Periaatteet (principle) ja käytänteet (practice) ovat osin limittyviä käsitteitä, joiden täsmällinen määrittely on haastavaa. Vaikka ne olisikin tutkimuskirjallisuudessa määriteltä, periaatteiden yhteydessä usein mainitaan toimintatapoja tai ohjenuoria, jotka muistuttavat enemmän konkreettisia käytäntöjä. On siis jossain määrin tulkinnanvaraista, määritelläänkö jokin ohjeistus periaatteeksi vai käytänteeksi. Tässä tutkielmassa periaatteella tarkoitetaan yleisempää ohjenuoraa kehityksen ohjaamiseksi etenkin suunnitteluvaiheessa ja käytänteillä viitataan konkreettisempiin kehitystehtäviin ja -prosesseihin liittyviin suosituksiin. Käsitteiden yhteys on nähtävissä myös siinä, että periaatteista voidaan usein johtaa yksityiskohtaisempia

käytänteitä ja käytänteistä on monesti hahmotettavissa periaate, johon se pohjaa.

Tämän tutkielman tavoitteena on selvittää, mitä tulee ottaa huomioon käyttöliittymää suunniteltaessa ja miten toteuttaa toimiva käyttöliittymä. Tutkimuskysymykset ovat: mitkä ovat pääperiaatteet ja parhaat toimintatavat hyvien käyttöliittymien kehittämiseksi tutkimuskirjallisuudessa ja miten näitä periaatteita voidaan soveltaa käyttöliittymän kehittämisessä. Tutkielmassa kehitetään suunnitelma-artefakti yrityksen 3D-suunnitteluohjelmistossa näytettävien animaatioiden ja niiden parametrien määrittämisen käyttöliittymäksi. Kehityksen tueksi ja teoriaksi toteutetaan kirjallisuuskatsaus, josta löydettyjä periaatteita ja parhaita käytänteitä hyödynnetään käyttöliittymäsuunnitelman toteutuksessa. Käyttöliittymän tulisi olla yrityksen vaatimusmäärittelyn mukaan moderni ja helppokäyttöinen, jotta se paitsi tukisi yrityksen brändiä nykyaikaisena ohjelmistovalmistajana myös palvelisi käyttäjien pyrkimyksiä joustavasti.

2 Teoriaosa

2.1 Kirjallisuuskatsaus

Kitchenhamin ja Chartersin mukaan systemaattinen kirjallisuuskatsaus on hyvä tapa rakentaa teoreettista viitekehystä tai taustaa tutkimuksen pohjaksi. Useimmat tutkimukset alkavat kirjallisuuden esittelemisellä, mutta mikäli kirjallisuuskatsaus on vajaa tai vääristynyt, sen tutkimuksellinen arvo jää kyseenalaiseksi. Hyvin suoritettuna systemaattisen kirjallisuuskatsauksen tekeminen tuo tutkimukseen kattavan tietopohjan lisäksi myös menetelmällistä läpinäkyvyyttä. Huonoiksi puoliksi Kitchenham ja Charters taas listaavat katsauksen työläyden, sillä katsauksen tekeminen edellyttää kattavaa dokumentaatiota toistettavuuden saavuttamiseksi ja arvioinnin mahdollistamiseksi. (Kitchenham ja Charters 2007, s. 3, s. 10)

Teoreettisen pohjan kokoamisen lisäksi tässä tutkielmassa tehtävän systemaattisen kirjallisuudenkatsauksen tarkoitus on tutkimuskysymysten mukaisesti etsiä tutkimuksia, jotka esittelevät käyttöliittymien kehittämisen periaatteita ja parhaita toimintatapoja, joita hyödynnetään animaatiokäyttöliittymän suunnitelma-artefaktin kehittämisessä. Katsauksesta rajataan pois käyttäjäkokemukseen keskittyvät tutkimukset. Koska käyttöliittymät ja käyttäjäkokemus ovat hyvin läheisiä teemoja, joitakin hakuja tehdään myös hakusanoilla käyttäjäkokemus ja user experience (ux). Tämän lisäksi tehdään kohdennettuja lisähakuja sekä jo löydettyjen tutkimusten lähdeluetteloiden katsauksia aihealueeseen liittyvien lähteiden löytämiseksi. Tällä tavoin pyritään varmistamaan kaikkien relevanttien tutkimusten löytyminen.

Koska tarkoituksena on kehittää käyttöliittymäartefakti työpöytäohjelmistoon, kirjallisuushakuprosessissa rajataan pois mobiilikehitykseen keskittyvät julkaisut. Tämän vuoksi katsauksessa ei käsitellä myöskään web-suunnittelua käsitteleviä tuloksia. Katsauksesta on rajattu pois myös erikoistuneisiin alakohtaisiin käyttöliittymiin keskittyvät julkaisut, paitsi silloin, kun ne ovat relevantteja suunnittelututkimuksen aihealueeseen. Tämä tarkoittaa, että esimerkiksi tekoälyyn, ääniohjaukseen tai peleihin keskittyviä käyttöliittymäratkaisuja käsitteleviä julkaisuja ei oteta mukaan katsaukseen. Myös verkko- ja mobiilikäyttöliittymille laaditut periaatteet on rajattu pois katsauksesta, samoin kuin puhtaasti visuaalisiin periaatteisiin keskittyvät julkaisut.

Katsausprotokolla erittelee ne menet, joilla tietty systemaattinen kirjallisuuskatsaus toteutetaan. Ennalta määrätty protokolla on välttämätön tutkijasta johtuvien vääristymien ehkäisemiseksi. Ilman protokollaa esimerkiksi tutkijan odotukset voivat vaikuttaa tiettyjen tutkimusten valintaan. Tässä tutkielmassa noudatettiin hieman soveltaen Kitchenhamin ja Chartersin protokollaa, (Kitchenham ja Charters 2007, s. 12) joka koostuu seuraavista kohdista:

1. Tausta, katsauksen syy.
2. Tutkimuskysymykset, joihin katsauksella haetaan vastausta.
3. Strategia, jolla primääritutkimuksia etsitään, mukaan lukien hakusanat ja etsittävät resurssit (hakukone, tietokannat, spesifit artikkelit ja konferenssijulkaisut)
4. Tutkimusten valinnan kriteerit. Niillä määritetään, mitkä tutkimukset valitaan ja mitkä suljetaan pois katsauksesta. Kriteereitä voi kokeilla aluksi pieneen joukkoon tutkimuksia.
5. Tutkimusten valinnan proseduuri, tulisi kuvailla kuinka valitsemiskriteereitä sovelletaan, eli kuinka moni arvioija tarkastelee jokaista mahdollista sisällytettävää tutkimusta ja miten erimielisyydet ratkaistaan.
6. Tutkimusten laadun arvioinnin tarkistuslistat ja proseduurit. Tutkijoiden tulisi laatia laadun tarkistuslista, jonka perusteella yksittäisiä tutkimuksia arvioidaan.
7. Datan johtamisen strategia määrittelee, kuinka yksittäisestä tutkimuksesta haetaan tietoa. Jos data vaatii manipulaatiota tai oletuksia ja päätelmiä, protokollan tulisi yksilöidä soveltuva validointiprosessi.
8. Johdetun datan synteesi, jolla määritellään synteesisstrategia. Sen tulisi selventää aiotanko tehdä muodollinen meta-analyysi ja millä tekniikoilla.

2.2 Suunnittelututkimusteoriaisuus

Tutkimuskysymyksiin vastaamiseksi suoritetaan systemaattinen kirjallisuuskatsaus, jossa etsitään tutkimuskirjallisuudessa määritettyjä käyttöliittymäsuunnittelun periaatteita ja parhaita toimintatapoja. Lisäksi kehitetään käyttöliittymän suunnitelma-artefakti, jonka toteutuksen pohjalta pyritään löytämään vastauksia siihen, miten löydettyjä periaatteita voidaan soveltaa toimivan käyttöliittymän suunnittelemiseksi. Koska työn tavoitteena on kehittää konkreettinen artefakti, on perusteltua valita tutkielman lähestymistavaksi design science -

tutkimussuuntaus, jossa pyritään luomaan asioita, jotka edistävät ihmisten tarkoitusperiä ja tuottavat hyötyä ratkaisemalla ”käytännöllisesti relevantteja ongelmia”. (Weber 2010, s. 2)

Kehitystyön pohjaksi tehdään kirjallisuuskatsaus käyttöliittymäkehitystä käsittelevästä kirjallisuudesta. Usein uusien artefaktien kehitys onkin haastavaa sen vuoksi, että niiden tarvealueilla teoreettinen tutkimustieto on monesti riittämätöntä. (Hevner ym. 2004, s. 76) Tässä on paitsi suunnittelututkimuksen mahdollisuus osoittaa merkityksellisyytensä menetelmällisenä lähtökohtana, myös se seikka, joka erottaa suunnittelututkimuksen rutiininomaisesta kehityksestä: suunnittelututkimuksessa tulee osoittaa selvästi kehityksen kontribuutio alan yhteisiin tietoperusteisiin ja menetelmiin. (Hevner ym. 2004, s. 81)

Weberin mukaan IT-artefaktit kytkeytyvät tiettyyn paikkaan, aikaan ja yhteisöön ja ovat merkittävässä vuorovaikutuksessa ympäristönsä kanssa. Artefaktin fokus on ongelmassa itsessään ja se on valmis, kun se tyydyttää kaikkien sidosryhmien tarpeen ja ratkaisee relevantin ongelman. Artefaktin kehitys voidaan jakaa kahteen osaan: kehitykseen ja arviointiin. Kehitysprosessi on sarja toimintoja, joiden seurauksena artefakti syntyy. Arvioinnissa taas pyritään saamaan tietoa, jota voidaan käyttää sekä itse artefaktin että sen syntymiseen johtaneen prosessin laadun parantamiseksi. (Weber 2010, s. 2)

Tutkimus- ja kehitystyötä tehdään hyödyntäen sovelletusti Peffersin ja muiden (Peffers ym. 2006, s. 88–92) kehittämää suunnittelututkimuksen prosessia, johon kuuluu kuusi askelta:

1. Ongelman tunnistaminen ja motivointi: tunnista tutkimusongelma ja perustele ratkaisun arvo.
2. Johda ongelman määrittelystä ratkaisun tavoitteet: ratkaisun tulee esimerkiksi olla parempi kuin nykyiset.
3. Suunnittele ja kehitä artefakti. Tähän sisältyy sekä artefaktin toiminnallisuuden ja arkkitehtuurin suunnittelu että myös sen toteutus.
4. Demonstroi artefaktin vaikutus ongelman ratkaisemiseen.
5. Arviointi: tarkkaile ja mittaa miten hyvin artefakti ratkaisee ongelman. Ratkaisun tavoitteita verrataan artefaktin tuloksiin ongelmanratkaisussa. Tässä kohtaa on mahdollista palata kohtaan 3 vaikuttavuuden parantamiseksi tai se voidaan jättää tehtäväksi

myöhemmin.

6. Ongelman tärkeyden, artefaktin käytettävyyden, uutuuden ja vaikuttavuuden sekä sen suunnittelun eksaktiuden kommunikointi relevanteille yleisöille.

Pefferin ja muiden mukaan prosessin järjestys voi myös poiketa edellä mainitusta. Esimerkiksi voidaan aloittaa suoraan kohdasta 3, eli artefaktin suunnittelusta ja kehityksestä ja edetä taaksepäin prosessin täsmällisyyttä jälkikäteen lisäten. (Peffer ym. 2006, s. 92–93)

Edellä esitetyn etenemismallin toimiessa yleisempänä ohjenuorana tutkielman suunnitellututkimusosioon hyödynnetään artefaktin kehitystyössä toisteista ja inkrementaalista prosessia, jossa vuorottelevat kehitystyö ja sen tulosten arviointi. Tällöin arviointivaihe tuottaa olennaista palautetta kehitystyöhön suunnitteluprosessin ja kehitettävän tuotteen laadun osalta. Kehitysprosessi etenee toisteisesti siten, että kehitystyön tulosten arvioinnin jälkeen artefaktia kehitetään inkrementaalisesti edelleen vastaamaan arvioinnissa havaittuihin heikkouksiin tai puutteisiin. (Weber 2010, s. 2)

Artefaktin arvioinnissa tulee ottaa huomioon myös yrityksen tarpeet jolloin myös liiketoimintaympäristö tuottaa vaatimuksia, joiden pohjalta artefaktia arvioidaan. Arvioinnissa tulee ottaa huomioon, että kehitettävän käyttöliittymän tulee myös integroitua saumattomasti yrityksen brändiin ja ohjelmistoympäristöön. (Hevner ym. 2004, s. 85) Kuten niin monella muulla alalla, myös suunnittelu- ja insinööriyöstä suurin osa on viimeisen kahdenkymmenen vuoden aikana siirtynyt tehtäväksi tietokoneella.

Suunnittelukäyttöön tehdyt tietokoneohjelmat ovat kehittyneet 2D-piirtotyökaluista alakohdasta tietoa mallintaviksi ja yhä useampia spesifejä toiminnallisuuksia tarjoaviksi 3D-suunnitteluohjelmistoiksi. Tämän myötä myös niiden käyttö on muuttunut entistä monimutkaisemmaksi: eri suunnittelualoilla käytettävät sovellukset voivat sisältää useita satoja erilaisia toimintoja erilaisine alavalikkoineen ja asetuksineen. (s. 90–91 Lee ym. 2010) Tämä asettaa omat haasteensa käyttöliittymäkehitykselle.

3 Kirjallisuuskatsaus

3.1 Kirjallisuuskatsauksen toteutus ja tulokset

Kirjallisuuskatsauksessa tehtiin hakuja vuoden 2024 tammi- ja helmikuun aikana tietokannoista IEEE Xplore Digital Library, Web of Science, ACM Digital Library ja Scopus, jotka Jyväskylän yliopiston kirjaston tiedonhankintasivuilla mainittiin keskeisiksi informaatioteknologian lähteiden hakemiseen. Hakutuloksiin rajattiin aikavälillä 2000–2024 julkaistut vertaisarvioidut artikkelit ja konferenssijulkaisut, jotta tulokset rajautuisivat suhteellisen tuoreeseen tutkimustietoon.

Hakulausekkeet muodostettiin siten, että artikkelien otsikoista avainsanoja, kuten ”ui”, ”development” ja ”guidelines” eri vaihtoehtoilla, kuten ”graphical user interface” ja ”gui”. Lisäksi joissakin tietokannoissa eriteltiin tulosten rajaamiseksi vielä poissuljettavia termejä, kuten ”mobile”, ”web” tai ”ai”, koska näiden hakusanojen kohteiden katsottiin olevan erikoistuneita periaatteita sisältäviä aihealoja. Tietokantakohtaiset hakulausekkeet on kirjattu tarkemmin taulukkoon 1, jossa on eritelty myös tietokannasta kyseisellä hakulausekkeella löytyneiden tulosten määrä.

Tietokannoista tehtävien hakulausekehakujen lisäksi tehtiin lisä- ja tarkentavia hakuja kirjaston omasta kansainvälisten artikkeleiden JYKDOK-palvelusta sekä Google Scholarista, millä pyrittiin varmentamaan hakulauseiden kattavuus aihealueen kirjallisuuden osalta. Hakulausekkeitä muokattiin näiden lisähakujen perusteella siten, että täydentävien hakujen kautta löytynyt tutkimuskysymyksiin vastaava kirjallisuus löydettiin myös tietokantahakujen kautta. Tämän avulla hakulausekkeiden kattavuutta saatiin parannettua jonkin verran.

| Tietokanta | Käytetty hakulause | Lisärajoituksia | Tulosten määrä |
|----------------------------------|---|---|----------------|
| Web of Science | ((TI=("ui" OR "user interface" OR "gui" OR "graphical user interface" OR "interaction" OR "hci")) AND TI=("design" OR "development")) AND TI=("principle*" OR "practice*" OR "guideline*" OR "heuristic*") | Julkaisut 2000–2024, ala: Computer Science | 36 |
| JYKDOK kansainväliset artikkelit | (Otsikko:"ui" "user interface" "gui" "graphical user interface" "interaction" "hci" AND Otsikko:"design" "development" AND Otsikko:"guideline" "principle" "heuristic" "practice") | Kokoteksti, vertaisarvioitu, 2000–2024, englanninkieliset, aihe: Computer Science | 171 |
| Scopus | TITLE ((ui OR user AND interface OR gui OR graphical AND user AND interface OR interaction OR hci) AND (design OR development) AND (principle OR guideline OR practice OR heuristic) AND NOT (mobile OR ai OR voice)) AND PUBYEAR > 1999 AND PUBYEAR < 2025 | | 99 |
| IEEE Xplore | (((((("Document Title":ui) OR ("Document Title":user interface) OR ("Document Title":graphical user interface) OR ("Document Title":interaction) OR ("Document Title":gui) OR ("Document Title":hci)) AND (("Document Title":design) OR "Document Title":development) AND (("Document Title":guideline*) OR ("Document Title":principle*) OR ("Document Title":heuristic*) OR ("Document Title":practice*)))))) | 2000–2024 | 70 |
| ACM Digital Library | [[Title: "ui"] OR [Title: "user interface"] OR [Title: "graphical user interface"] OR [Title: "gui"] OR [Title: "interaction"] OR [Title: "hci"]] AND [[Title: "principles"] OR [Title: "guidelines"] OR [Title: "practice"] OR [Title: "heuristic"]] AND NOT [Title: "mobile"] AND NOT [Title: "ai"] AND [[Title: "design"] OR [Title: "development"]] AND NOT [Title: "web"] AND [E-Publication Date: (01/01/2000 TO 01/31/2024)] | | 118 |

Taulukko 1. Tietokantahauissa käytetyt hakulausekkeet

Tietokantahakujen tulokset käytiin läpi yksitellen arvioiden artikkelin otsikon relevanttiutta, eli vastasiko artikkeli otsikon perusteella tutkimuskysymyksiin. Mikäli artikkeli vaikutti aiheeseen soveltuvalta, luettiin tarkemmin abstraktia ja itse tekstiä. Mikäli artikkelissa esiteltiin tai käsiteltiin käyttöliittymäsuunnittelun periaatteita tai käytänteitä, ne kirjattiin ylös ja arvioitiin samalla olivatko artikkelissa käytetyt menetelmät ja niiden esittely oman arvioni mukaan pätevää. Artikkelit rajattiin katsauksen ulkopuolelle esimerkiksi siinä tapauksessa, mikäli menetelmää ei eritelty lainkaan.

Sisällyttämisarvioinnissa katsottiin myös millaisiin lähteisiin artikkelissa oli tukeuduttu ja millä laajuudella. Mikäli lähdeluettelo oli lyhyt tai sisälsi ainoastaan ennen vuotta 2000 julkaistuja teoksia, sitä ei sisällytetty kirjallisuuskatsauksen tuloksiin. Rajasin pois tuloksista

myös sellaisia artikkeleita, joissa periaatteiden johtamistapaa ei ollut eritelty tai se ei ollut pääteltävissä tekstistä. Taulukossa 2 on eritelty näiden rajausten jälkeen kirjallisuuskatsauksen tuloksena löydetyt 10 artikkelia, joista varhaisin on vuodelta 2002 ja viimeisin viime vuodelta.

| Tekijät | Otsikko | Julkaisu vuosi |
|--|--|----------------|
| Benaida | Developing and extending usability heuristics evaluation for user interface design via AHP | 2023 |
| Ruiz, Serral & Snoeck | Unifying Functional User Interface Design Principles | 2021 |
| Pitale & Bhungara | Human Computer Interaction Strategies — Designing the User Interface | 2019 |
| Silva, Merino, Merino, & Figueiredo | Usability principles and measures for the design of a product applied to the design management | 2015 |
| Gómez Reynoso & Olfman | The impact of combining gestalt theories with interface design guidelines in designing user interfaces | 2012 |
| Torres, Heck, Rudd & Kelley | Usability Engineering: “Best of the Best” Best Practices | 2012 |
| Lee, Eastman, Taunk & Ho | Usability principles and best practices for the user interface design of complex 3D architectural design and engineering tools | 2010 |
| Blair-Early & Zender | User Interface Design Principles for Interaction Design | 2008 |
| Badashian, Mahdavi, Pourshirmohammadi, & Nejad | Fundamental Usability Guidelines for User Interface Design | 2008 |
| Kamper | Extending the Usability of Heuristics for Design and Evaluation: Lead, Follow, and Get Out of the Way | 2002 |

Taulukko 2. Suunnitteluperiaatteita sisältäneet artikkelit.

Löydetyjä periaatteita oli 28 kappaletta ja ne on eritelty taulukossa 3. Yksi katsauksessa löydetty artikkeli (Lee ym. 2010) sisälsi 3D-suunnitteluun ja mallinnukseen erikoistuneita periaatteita ja käytänteitä, jotka sisällyttiin katsaukseen, koska ne sopivat kehitettävän käyttöliittymäsuunnitelman kohdealalle. Kyseiset periaatteet on esitelty tarkemmin taulukossa 4. Katsauksessa löydetyt käyttöliittymäsuunnittelun käytänteet on jaoteltu erikseen taulukoihin 5 ja 6 sen mukaan, koskevatko ne enemmän koko kehitysprosessia vai yksittäisiä kehitystehtäviä. Kehitystehtävien käytänteet on listattu taulukossa 5 ja kehitysprosessin käytänteet taulukossa 6. Taulukkoon 7 on koottu erikseen 3D-suunnitteluovelluksiin erikoistuneet käytänteet, jotka ovat taulukon 5 käytänteiden kanssa kehitystehtäviä koskevia käytänteitä.

| Periaate | Artikkelit, joissa esiintyy | Samankaltaisia muotoiluja / selite |
|--|---|---|
| Johdonmukaisuus | Pitale & Bhungara, Benaida, Lee ym., Ruiz ym., Blair-Early & Zender, Badashian ym., Kamper, Silva ym., Gómez Reynoso & Olfman | Yhdenmukaiset toimintoketjut |
| Standardit | Pitale & Bhungara, Benaida, Lee ym., Ruiz ym., Blair-Early & Zender, Badashian ym., Kamper, Silva ym., Gómez Reynoso & Olfman | Mukaudu alustakohtaisiin konventioihin ja UI-standardeihin |
| Joustavuus | Pitale & Bhungara, Benaida, Lee ym., Ruiz ym., Blair-Early & Zender, Badashian ym., Kamper, Silva ym. | Erilaisten käyttäjien huomioiminen, huomioi käyttäjien taitotasot, kustomoitavuus, mukautuminen |
| Käyttäjän kontrolli ja vapaus | Benaida, Ruiz ym., Badashian ym., Kamper, Silva ym., Gómez Reynoso & Olfman, Blair-Early & Zender | Undo ja redo, selvä peruuttaminen ja poistuminen |
| Apu ja dokumentaatio | Benaida, Lee ym., Ruiz ym., Blair-Early & Zender, Badashian ym., Kamper, Gómez Reynoso & Olfman | |
| Informatiivinen palaute | Pitale & Bhungara, Lee ym., Ruiz ym., Blair-Early & Zender, Kamper, Silva ym., Gómez Reynoso & Olfman | Anna selkeitä ja tiiviitä kehoitteita käyttäjän omalla terminologialla ja kielellä |
| Käyttäjien tunteminen / huomioiminen | Pitale & Bhungara, Benaida, Lee ym., Ruiz ym., Blair-Early & Zender, Badashian ym., Kamper | Käyttäjien huomioiminen suunnittelussa, mihin tehtävään ohjelmaa on tarkoitettu |
| Minimalismi | Pitale & Bhungara, Benaida, Lee ym., Badashian ym., Kamper, Silva ym., Ruiz ym. | Esteettisyys ja minimalistinen design, suojele käyttäjää yksityiskohdilta ellei hän itse hae niitä, visuaalinen selkeys |
| Tunnistaminen muistamisen sijaan | Pitale & Bhungara, Benaida, Ruiz ym., Badashian ym., Kamper, Gómez Reynoso & Olfman | Käyttäjän muistia ei tule kuormittaa |
| Virheiden tunnistus, diagnosointi ja niistä palautuminen | Benaida, Lee ym., Ruiz ym., Badashian ym., Kamper, Silva ym. | Anna virheviestejä, jotka tarjoavat ratkaisuja |
| Organisoi tieto sijainneittain | Pitale & Bhungara, Ruiz ym., Blair-Early & Zender, Badashian ym., Gómez Reynoso & Olfman | Laita samaan liittyvät asiat yhteen, toisiinsa liittyvien käyttöliittymäelementtien läheisyys, käsitteellinen läheisyys |
| Järjestelmän tilan näkyvyys | Benaida, Lee ym., Ruiz ym., Badashian ym., Kamper | Esimerkiksi toiminnan suorittamisen näyttäminen |
| Virheiden ennaltaehkäisy | Benaida, Ruiz ym., Badashian ym., Kamper, Silva ym. | Hyödylliset rajoitteet |
| Järjestelmän ja maailman välinen yhteensopivuus | Benaida, Lee ym., Badashian ym., Kamper, Silva ym. | Tue käyttäjän luonnollista työjärjestystä |
| Minimoi käyttöliittymä, maksimoi sisältö | Blair-Early & Zender, Silva ym., Gómez Reynoso & Olfman | Informaation priorisointi, käyttöliittymä on sisältöä |
| Oppimisen helppous | Benaida, Badashian ym., Kamper | Tee käyttöliittymän toiminnallisuuksista ilmiselviä ja saavutettavia – yksilölliset nimet |
| Metaforat | Ruiz ym., Blair-Early & Zender, Badashian ym. | Auttaa käyttäjiä ymmärtämään monimutkaisempia toimintoja |
| Tehokkuus | Badashian ym., Kamper, Silva ym. | Minimoi tehtävän suorittamiseen vaaditut kohdat, funktionaalisuuden priorisointi |

| | | |
|---|--|---|
| Maamerkit | Blair-Early & Zender, Gómez Reynoso & Olfman | Käyttäjillä tulisi olla tietoa heidän senhetkisestä sijainnistaan käyttöliittymän käsitteellisessä tilassa |
| Selkeä aloituspiste | Blair-Early & Zender, Gómez Reynoso & Olfman | |
| Minimoi latenssi | Badashian ym., Gómez Reynoso & Olfman | Ajallinen läheisyys |
| Rohkaise tutkimiseen | Ruiz ym., Badashian ym. | Tee tutkittavia käyttöliittymiä |
| Hyödyllisyys | Benaïda, Ruiz ym. | Auta käyttäjää olemaan tuottavampi |
| Helppokäyttöisyys | Benaïda | |
| Tyytyväisyys | Benaïda | |
| Kontekstin tunnistaminen | Lee ym. | Käyttöliittymän tai järjestelmän käytön automaattinen säätäminen ohjelman käyttömoodin tai kontekstin perusteella, esim. tilannekohtainen apu |
| Tarjoa järjestelmästä selkeä käsitteellinen malli | Ruiz ym. | |
| Toimintojen jakaminen pienempiin osiin | Pitale & Bhungara | |

Taulukko 3. Katsauksella kootut suunnitteluperiaatteet.

Löydetyistä periaatteista suurin osa oli sellaisia, että ne esiintyivät useammalla kuin yhdellä kirjoittajalla. Taulukossa on esitelty myös vaihtoehtoisia muotoiluja periaatteille sekä osan kohdalla tarkempia selityksiä esitellyille periaatteille. Joissakin lähteissä samaan periaatteen viitattiin eri nimityksillä, joissakin samat ilmaisut oli listattu yhden periaatteen alle ja toisissa ne oli eroteltu omiksi periaatteikseen. Tässä tutkielmassa kyseiset periaatteet yhdistettiin samankaltaisuuden vuoksi taulukkoon yhdeksi yleisemmäksi periaatteeksi. Periaatteita olisi voinut kertyä siis enemmänkin, mutta katsoin hyödyllisemmäksi karsia yksittäisten periaatteiden määrää luokittelemalla niitä omiksi kattavimmiksi periaatteikseen.

Pelkällä mekaanisella listauksella periaatteiden määrä olisi kasvanut suureksi ja useammat samankaltaiset periaatteet olisivat mielestäni olleet tarpeetonta toistoa. Esimerkiksi periaatteen *Käyttäjän kontrolli ja vapaus* kanssa vaihtoehtoiseksi muotoiluksi on kirjattu *undo ja redo* sekä *selvä peruuttamis- ja poistumismahdollisuus*, jotka esiintyivät kaikki yksittäisinä periaatteina kirjallisuudessa. Muun muassa Ruiz ja muut toivat periaatekatsauksessaan esille erillisinä periaatteina sekä toimintojen peruuttamismahdollisuuden että käyttäjän kontrollin mahdollistamisen (Ruiz, Serral ja Snoeck 2021, s. 56). Reynosolla ja Olfmanilla taas selkeä peruuttaminen sekä istunnosta poistuminen oli mainittu samassa periaatteessa (Gómez Reynoso ja Olfman 2012, s. 3), mikä osoittaa, että kirjallisuudessakin luokittelu tapahtuu

jotakuinkin tulkinnanvaraisesti.

| 3D-parametriseen suunnitteluun erikoistuneet periaatteet: | Artikkelit, joissa esiintyy | Tarkempi kuvaus |
|---|-----------------------------|--|
| Työtilan maksimoiminen | Lee ym. | Mallintamiseen tai piirtämiseen tarkoitetun tilan maksimoiminen |
| Graafinen rikkaus | Lee ym. | Teksti-informaation korvaaminen graafisella informaatiolla, kuten kuvilla ja animaatioilla. Esim. esikatselun tarjoaminen tietyille toiminnoille, värikoodaus valituille objekteille tai särmille jne. |
| Suora manipulointi | Lee ym. | Suoran vuorovaikutuksen tarjoaminen järjestelmässä näkyvän objektin tai kokonaisuuden kanssa. |

Taulukko 4. 3D-suunnitteluun erikoistuneet suunnitteluperiaatteet.

Taulukossa 4 on Leen ja muiden artikkelista kirjatut 3D-suunnitteluun erikoistuneet periaatteet. Kirjoittajat kävivät läpi yleisiä 3D-suunnittelun ja -mallinnuksen alalla käytettäviä ohjelmistoja, joiden perusteella he esittivät periaatteita alalla käytettävien ohjelmistojen käyttöliittymien suunnitteluun. Lisäksi he esittelivät toimiviksi kokemiaan ratkaisuja ja käytänteitä kyseisissä ohjelmistoissa. (Lee ym. 2010, s. 10) Artikkelissa esiteltyt käytänteet on eritelty taulukossa 5.

| Parhaat käytänteet kehitystyössä | Artikkelit, joissa esiintyy | Tarkempi kuvaus |
|---|----------------------------------|---|
| Otsikoiden ja linkkitekstien tulisi olla kuvaavia | Pitale & Bhungara, Badashian ym. | |
| Värien käyttö ja synteesi | Badashian ym., Silva ym. | Värejä tulisi käyttää kontrastin parantamiseen, mutta ei liikaa. Lisäksi informaation esittäminen värien avulla |
| Valintaruutujen käyttäminen binäärisiin valintoihin | Pitale & Bhungara | |
| Tekstin näkyvyyden tulisi olla käyttäjän säädettävissä | Pitale & Bhungara | |
| Tee aihealue selväksi heti kättelyssä | Blair-Early & Zender | Käyttöliittymän tulisi ilmentää ohjelman sisältöä |
| Käytä käyttöliittymässä sisällön esittämiseen soveltuvaa asetelua | Blair-Early & Zender | Datan visuaalioidinnissa tulisi ottaa huomioon sen ymmärrettävyys |
| Suunnittele käyttöliittymästä visuaalisesti miellyttävä | Blair-Early & Zender | Paras muoto ilmentää sisällön ominaisuuksia stimuloivalla tavalla |
| Käytä metaforia, kun sisältö on uutta tai epäselvää | Blair-Early & Zender | Metaforat luovat muistoja ja assosiaatioita ja mahdollistavat aikaisemman tiedon hyödyntämisen |
| Valikoiden käyttäminen dokumentaationa | Badashian ym. | Valikoista tulisi käydä ilmi, mitä ohjelmalla voi tehdä |
| Monikielinen käyttöliittymä | Badashian ym. | Samaa käyttöliittymää tulisi voida käyttää eri kielillä |
| Käyttäjäoikeuksien keskitetty hallinta | Badashian ym. | Käyttäjäoikeuksien hallinnalla voidaan määrittää käyttöliittymässä kullekin käyttäjälle näkyvät elementit |

Taulukko 5. Käyttöliittymäkehityksen käytänteet.

| Parhaat käytänteet kehitysprosessissa | Artikkelit, joissa esiintyy | Tarkempi kuvaus |
|--|---|---|
| Käyttäjien sisällyttäminen kehitykseen | Torres ym., Ruiz ym., Badashian ym., Benaïda | Palautetta sidosryhmiltä ja käyttäjätestaus, käyttäjätutkimus |
| Iteraatiot | Torres ym., Ruiz ym. | Projektivaatimusten toteuttamisen mittaaminen ja vertaaminen aiempiin tuloksiin |
| Ymmärrä toiminnot | Torres ym., Ruiz ym. | Esim. käyttötapausmallien kautta |
| Varhainen visualisointi | Torres ym. | Interaktiivinen prototyyppi 2-3 iteraation aikana ja sen yhdistäminen MVP:en seuraavien 2-3 aikana |
| Käyttöjärjestelmäriippumattomuus | Silva ym. | Käytä alustariippumatonta ohjelmointikieltä |
| Empiirinen mittaaminen | Ruiz ym. | |

Taulukko 6. Käyttöliittymäsuunnitteluprosessin käytänteet.

| Parhaat käytänteet kehitystyössä (3D-suunnitteluun erikoistuneet) | Artikkelit, joissa esiintyy | Tarkempi kuvaus |
|--|-----------------------------|--|
| Tärkeimmät dialogit näkyvillä | Lee ym. | Esimerkiksi ominaisuudet-ikkuna pääkkunan vasemmassa reunassa |
| Dialogin tiedon levittäminen | Lee ym. | Dialogin skrollattavuus alaspäin tai tiedon levittäminen useammille välilehdille |
| Esikatselunäkymä | Lee ym. | Aikaa vievien toimintojen seurausten esikatselu välittömästi |
| Mallin ja piirrosten yhdenmukaisuus | Lee ym. | Kaksisuuntainen tai yksisuuntainen (mallista piirroksen) |
| Erilliset mallinnus- ja piirustustilat | Lee ym. | Piirustusasetusten rajoittuminen vain piirustuksiin |
| Hierarkkinen apu vs. informaatio- ja tutoriaaliosiot | Lee ym. | Molemmat kirjoittajien mielestä tehokkaita |
| Kontekstisidonnainen opastus | Lee ym. | Vihjetekstit vietäessä hiiri painikkeen ylle, kehotteet komentorivillä tai erillinen ohjeruutu kursorin vieressä |
| Kustomointi | Lee ym. | Omat asetukset valintojen mukauttamiselle tai esim. piilottaminen tai järjestäminen valikoissa saatavilla |
| Valikoiden rakenne työvaiheiden mukaan | Lee ym. | |
| Valikkojen ja toimintojen näkyvyyden rajoittaminen | Lee ym. | Koska vaihtoehtoisia komentoja ja toimintoja on paljon, niiden näkyvyyttä voidaan rajoittaa ryhmittelyllä, esiin klikattavilla työkaluriveillä tai eri ohjelman työskentelymooideilla |
| Objektien valitseminen suoralla interaktiolla | Lee ym. | Objektin valitseminen hiirellä eri kohtiin osoittamalla |
| Valittujen objektien värikoodaus | Lee ym. | Valitut objektit erivärisiä kuin muut |
| Automaattinen tarkistus | Lee ym. | Esim. piirretyn moniviivan sulkemisen jälkeinen värimuutos, fyysiset konfliktit tai muut potentiaaliset virhetilanteet |
| Objektien suora muokkaaminen | Lee ym. | Esimerkiksi muodon muokkaaminen sen reunoista raahaamalla |
| Toimintojen suorittaminen mallinnustilasta | Lee ym. | Esim. kontekstivalikosta, jossa näkyy vain tilanteeseen sopivat toiminnot tai näppäinyhdistelmillä |
| Näkymän muuttaminen | Lee ym. | Esim. näkymäkuutiolla tai tallennetuilla omilla kuvakulmilla |
| Näkymän hallitseminen tasoilla | Lee ym. | Objektien näyttäminen eri tasoilla, tasojen syyttämisen tai sammuttaminen |

Taulukko 7. 3D-suunnitteluun erikoistuneet käyttöliittymäkehityksen käytänteet.

Taulukoihin 5–7 on koottu katsauksessa löydettyjä käyttöliittymäsuunnittelun käytänteitä, joiden jokaisen kohdalla on löydettävissä sitä vastaava periaate taulukoista 3–4. Tämä kuvastaa periaatteiden ja käytänteiden välistä yhteyttä, jossa periaatteiden voidaan nähdä olevan ylätasoa käsitteitä ja ohjenuoria, joista on mahdollista johtaa kehitystyön kannalta konkreettisia käytännön ohjeita.

Taulukoista on myös nähtävissä, kuinka yleiset käyttöliittymäkehityksen käytänteet sisältävät potentiaalin muodostaa tiettyyn kontekstiin sidottuja erikoistuneita ohjeistuksia: esimerkiksi 3D-suunnittelussa valikoiden rakenteen muodostaminen työvaiheiden mukaan voidaan johtaa yleisestä periaatteesta, jonka mukaan järjestelmän ja sen ulkopuolisen maailman tulisi vastata toisiaan. Näkisin, että tällainen johtaminen ei kuitenkaan ole triviaali toimenpide, vaan se vaatii kehittäjältä tietoa ja näkemystä kohdealasta, mutta osoittaa kuitenkin periaatteiden potentiaalin käyttöliittymäkehityksessä.

3.2 Yhteenveto kirjallisuudessa mainituista periaatteista

Monet kirjallisuudessa esitetyistä periaatteista ovat peräisin 1970–1990-luvuilta, joskin niitä käytetään edelleen tutkimuksessa ja tietojärjestelmien kehityksessä. Niitä on myös arvioitu ja kehitetty edelleen useaan otteeseen. Esimerkiksi Nielsenin kymmenen heuristiikkaa ja Shneidermanin ”kahdeksan kultaista sääntöä” ovat hyvin tunnettuja ja käytettyjä käyttöjärjestelmien alalla. Ruizin ja muiden mukaan käyttöliittymäsuunnittelun koulutuksessa erilaiset periaatteet voivat olla avuksi, mutta he toteavat, että periaatteiden monenkirjava joukko voi kuitenkin vaikeuttaa olennaisten sääntöjen löytämistä. (Ruiz, Serral ja Snoeck 2021, s. 48, 51–52) Osa katsauksessa löytyneistä periaatteista on myös keskenään ristiriidassa, jolloin voi olla vaikea valita, mitä ja miten periaatteita kulloinkin käytetään. Myös periaatteiden suuri määrä tuottaa haasteita niiden soveltamiselle.

Kun listataan periaatteiden yleisyyttä katsauksen tuloksena saaduissa artikkeleissa, voidaan havaita että kehitysperiaatteista *johdonmukaisuus* sekä *yhdenmukaisuus eri standardien kanssa* oli mainittu kaikissa periaatteita listanneissa artikkeleissa. Myös *joustavuus* oli listattu kaikkiaan kahdeksassa artikkelissa kymmenestä, minkä lisäksi seitsemässä löytyneistä teksteistä mainittiin seuraavat periaatteet: *käyttäjän kontrolli ja vapaus*, *minimalistinen design*, *käyt-*

täjien tunteminen, informatiivinen palaute sekä apu ja dokumentaatio. Edelleen kuudessa artikkelissa kymmenestä mainittiin periaatteet *tunnistaminen muistamisen sijaan ja virheiden käsittely*. Loput periaatteista oli mainittu ainoastaan puolessa tai sitä pienemmässä osuudessa katsauksen tuloksista, joten edellä mainitut kymmenen kehitysperiaatetta ovat katsauksen kirjallisuudessa eniten mainintoja saaneita periaatteita. Tästä perusteella niitä voidaan pitää myös suositeltavina ohjeistuksina käyttöliittymäkehityksessä.

Osa periaatteista oli mainittu ainoastaan yhdessä artikkelissa, kuten esimerkiksi *helppokäyttöisyys, tyytyväisyys tai kontekstin tunnistaminen*. Näistä helppokäyttöisyys tai tyytyväisyys ovat melko abstrakteja periaatteita, joiden voidaan oikeastaan nähdä olevan tavoitteita, joihin periaatteilla pyritään tähtäämään. Esimerkiksi usein viitatuista periaatteista *minimalismin* voidaan ajatella olevan sellaisen visuaalisen selkeyden tarjoamista, jossa etsittävät käyttöliittymäelementit löytyvät helposti, mikä itsessään jouduttaa ja helpottaa ohjelman käyttöä. Voi olla, että tämän vuoksi sitä ei ole listattu kovinkaan monessa artikkelissa: Blair-Early ja Zender (2008, s. 86) jopa toteavat, että ”helppokäyttöisyyttä” ja ”intuitiivisuutta” on aiheestakin kritisoitu epämääräisiksi käsitteiksi, eikä niiden tarkempi määrittelykään varsinaisesti edesauta käyttöliittymän suunnittelussa niiden mukaisiksi.

Useimmiten mainittujen periaatteiden *johdonmukaisuuden ja yhdenmukaisuuden eri standardien* kanssa voidaan niin ikään ajatella olevan ohjelman helppokäyttöisyyttä ja intuitiivisuutta tukevia kehitysperiaatteita. Mikäli käyttäjän tavat vuorovaikuttaa ohjelman kanssa ovat samanlaisia ohjelman eri toiminnoissa ja osissa, tämä tukee käytettävyyttä, koska käyttäjän ei tarvitse opetella uusia toimintatapoja toiminnallisuuksien käyttämiseksi ja käyttäjä voi toimia ennako-olettamustensa varassa myös uusia toiminnallisuuksia käyttäessään. Lisäksi sama koskee yhdenmukaisuutta eri standardien kanssa: käyttäjä todennäköisesti olettaa Windows-käyttöjärjestelmässä toimiessaan, että näppäinyhdistelmä ctrl-c toimii myös alustalla pyörivissä ohjelmissa kopioimisen pikatoimintona, mikä toimiessaan nopeuttaa ja helpottaa työntekoa.

Edellä eriteltyt katsauksen kymmenen viitatuinta kehitysperiaatetta ovat miltei samat kuin Nielsenin kymmenen heuristiikkaa vuodelta 1994. Kaikki Nielsenin heuristiikat löytyvät listattujen periaatteiden joukosta, mutta kymmenestä viitatuimmasta jäävät puuttumaan Nielsenin mainitsemat ”järjestelmän tilan näkyvyys”, ”maailman ja järjestelmän vastaavuus” ja

”virheiden ehkäisy”. (Benaida 2023, s. 9694–9695) Näiden tilalla katsauksen viitatuimmissa kymmenessä periaatteessa ovat *johdonmukaisuus, käyttäjien tunteminen / huomioiminen* ja *virheiden käsittely*. Shneidermanin vuoden 1998 ”kahdeksasta kultaisesta säännöstä” kaikki sisältyvät myös katsauksella löydettyihin periaatteisiin, mutta niistä ”dialogien selkeä lopetus” jää katsauksen kymmenen viitatuimman ulkopuolelle. (Benaida 2023, s. 9694)

Sen perusteella, että tämän tutkielman kirjallisuuskatsauksen tuloksiin sisältyy myös edellä mainitut tunnetut ja tunnustetut periaatelistaukset, voi katsausten tulosten nähdä kattavan ainakin tunnetuimmat käyttöliittymäkehitysperiaatteet. Toisaalta tässä esitetty listaus ei ole identtinen niiden kanssa, vaan sisältävät hieman erilaisen painotuksen, kuin Shneidermanin ja Nielsenin periaatelistaukset. Tämän voi ajatella heijastavan tämänhetkisen kirjallisuuden näkemystä eri periaatteiden merkittävyydestä, mutta toisaalta se voi olla myös merkki siitä, että yleisten suunnitteluperiaatteiden sijaan viime aikoina olisi keskitytty enemmän erikoistuneisiin suunnitteluperiaatteisiin.

4 Sanoista teoksi – suunnittelututkimusosuus

4.1 Artefaktin suunnittelu ja toteutus

Tässä tutkielmassa kehitetään suunnitelma-artefakti 3D-suunnitteluohjelmiston animaatioiden määrittämisen käyttöliittymästä. Käyttöliittymäsuunnitelma kehitetään yrityksen toimintasuunnitelmasta ja sen kehitys tapahtuu yrityksen vaatimusmäärittelyn pohjalta hyväksikäyttäen kirjallisuuskatsauksessa koottuja suunnitteluperiaatteita ja -käytänteitä. Kehitystyön pohjalta pyritään kokoamaan havaintoja yleisten periaatteiden soveltamisesta suunnittelutehtävään.

Kehitystyö tapahtuu mukailien jo aikaisemmin esiteltyä Peffersin ja muiden (Peffers ym. 2006, s. 88–92) suunnittelututkimuksen prosessia. Prosessissa tunnistetaan tutkimusongelma, johdetaan tutkimusongelmasta tavoitteet, suunnitellaan artefakti, demonstroidaan sen vaikutus ongelman ratkaisemiseen, arvioidaan aikaansaatu ratkaisu ja kommunikoidaan tulokset.

Tässä tapauksessa tutkimusongelmassa yhdistyvät yrityksen tarpeet ja teoreettinen kehys: yrityksellä on olemassa käyttöliittymän tarve, johon suunnitelma-artefaktin kehittäminen pyrkii vastaamaan. Artefaktin toteutuksella tavoitellaan kuitenkin myös tutkimuksellista lisäarvoa ja lisätietoa suunnitteluperiaatteiden ja -käytänteiden soveltamisesta käyttöliittymäkehityksessä. Tämän johdosta ratkaisun tavoitteiksi voidaan määritellä, että sen tulee vastata yrityksen toiminnallista vaatimusmäärittelyä ja sen suunnittelussa tulee toteuttaa tutkielmassa käsitellyjä periaatteita ja käytänteitä.

Lisäksi artefaktin tulee vastata yrityksen yleisiä käyttöliittymätavoitteita, kuten helppokäyttöisyyttä ja intuitiivisuutta. Näihin tavoitteisiin pyritään katsauksella koottujen suunnitteluperiaatteiden noudattamisella. Artefaktia arvioidaan toteutuksen jälkeen ja sen aikana toistuvasti ratkaisun vaikuttavuuden parantamiseksi ja periaatteiden soveltamisen arvioimiseksi. Tulosten kommunikointi tapahtuu tämän tutkielman kautta.

Suunnittelututkimusprosessin noudattamisen lisäksi kehitystyössä hyödynnetään kirjallisuuskatsauksen tuloksena saatuja käyttöliittymäkehityksen käytänteitä, joista taulukosta 6 itse suunnitteluprosessissa toteutettiin *iteraatioita, varhaista visualisointia* ja yrityksen vaatimusmäärittelyä saatuja käytötapausmallien kautta käytännettä *ymmärrä toiminnot*. Itera-

tiivinen kehitysmuoto mainittiin myös suunnittelututkimuksen viitekehyksessä, joka toimii teoreettisena perustana itse tutkimusmenetelmälle (Hevner ym. 2004, s. 78).

4.2 Vaatimusmäärittely

Vaatimusmäärittelyssä todetaan, että animoinnin käyttöliittymän tulee mahdollistaa seuraavat toiminnallisuudet:

1. Polun määrittely
2. Kohteen kokoonpanon muuttuminen polun matkalla
3. Kohteen transformaatio polun matkalla
4. Kunkin yksittäisen objektin parametrien muuttuminen polun matkalla
5. Edellä mainittujen toimintojen ajoitukset
6. Videon tuottaminen

Vaatimusmäärittelyssä on määritelty käyttöliittymään sisällytettävät parametrit sekä niiden muokkaamisen mahdollisuudet. Yrityksen puolesta päätettiin käyttöliittymän ensisijaiseksi muodoksi käyttöliittymälle pitkittäinen dialogi suunnittelutilan sivustalle, mikä on tyypillinen käyttöliittymä ohjelmiston sisällä. Tämä noudattaa myös Leen ym. (Lee ym. 2010, s. 102) periaatetta *työtilan maksimoimisesta*. Animaatiossa objektin kulkema reitti määritetään suunnittelunäkymään moniviivana, jonka määrittämisessä hyödynnetään ohjelmiston olemassa olevaa piirtotyökalua, mikä on perusteltua ohjelmiston sisäisten vuorovaikutustapojen johdonmukaisuuden säilyttämiseksi. Tarpeen vaatiessa piirtoa voidaan myöhemmin muokata mahdollistamaan myös monimuotoisemmat polut, kuten erilaiset käyrät, jotka ovat ohjelmassa mahdollisia.

Vaatimusmäärittely on rajoitettu siinä mielessä, että siinä ei oteta kantaa mahdollisesti myöhemmin mahdollistettaviin määrittelyihin. Näihin voivat sisältyä esimerkiksi animaation kameran polun ja toiminnan määrittäminen sekä eri objektien, transformaatioiden ja animaation ajoituksen määrittäminen. Vaatimusmäärittelyn rajoittamisella pyrittiin pitämään tämän tutkielman toteutus hallittavana.

4.3 Artefaktin toteutus

Käyttöliittymäsuunnitelman toteutus lähti liikkeelle vaadittavien toiminnallisuuksien sijoittamisesta sivupaneelimuotoon. Kuviossa 1 näkyvässä ensimmäisessä iteraatiossa ajatuksena oli, että ohjelman työkaluriviin sijoitetaan toiminnon painike, josta painamalla ohjelma siirtyy animoitavien objektien valintaan ja siitä automaattisesti polun piirtoon, joka toimii toiminnossa *selkeänä aloituspisteenä*. Käyttöliittymänäkymään päivittyy tämän myötä polun eri osat eli segmentit, joita erottavat solmupisteet. Tämä tapahtuu periaatteen *järjestelmän tilan näkyvyys* mukaisesti, jolloin käyttäjä näkee suoraan oman vuorovaikutuksensa tulokset käyttöliittymässä.

Ensimmäisessä iteraatiossa segmentit ja solmupisteet ovat samassa käyttöliittymäosiossa ja niistä näytetään suunnittelutilassa valittu osio tai solmu sinisellä korostettuna samoin periaatteen *järjestelmän tilan näkyvyys* mukaisesti. Suunnittelunäkymästä kursorilla segmentin tai solmun valitsemalla näkymä muuttuu valinnan mukaiseksi, mikä noudattaa myös edellä mainittua periaatetta. Ohjelmassa toimiva valinta on myös linjassa 3D-suunnitteluun erikoistuneen periaatteen *suora manipulointi* kanssa toteuttaen käytännettä *objektien valitseminen suoralla interaktiolla* taulukosta 7.

Periaatetta *organisoii tieto sijainneittain* soveltaen käyttöliittymä on jaettu pienempiin osioihin, joissa on koottu toisiinsa läheisesti liittyvät asiat yhteen. Yllä mainitussa segmentit ja solmut -osiossa on kootusti polun osatekijät ja niiden parametrit, toisena olevassa transformaatio-osiossa on näkyvissä ja muokattavissa ensimmäisen segmentin transformaatiot ja niiden parametrit ja kolmannessa osiossa on yhdessä segmentillä liikkuvat objektit ja niiden tiedot sekä mahdollisuus lisätä objekteja animoitavaan joukkoon ohjelman valintatyökaluilla.

Ohjelman valmiiden valintatyökalujen käyttö edistää *johdonmukaisuutta* ohjelmistossa, mikä oli myös yksi kirjallisuuskatsauksessa löydetyistä periaatteista. Saman periaatteen mukaisesti transformaatio-osiossa on määritelty transformaation tavoiksi ”Along path”, ”Directed” ja ”Sweep”, jotka esiintyvät muuallakin ohjelmassa vaihtoehtoina esimerkiksi objektin kierron toiminnalle. Objekti voi siis parametreista riippuen kiertyä polun mukaisesti, pysyä ennallaan tai muuttua tasaisesti annettujen arvojen välillä.

Käyttöliittymän jakaminen pienempiin osiin edustaa myös periaatetta *minimalismi*, jonka

vaihtoehtoisena muotoiluna on kirjallisuuskatsauksessa nähty *visuaalinen selkeys*. Tätä edustaa myös mahdollisuus pienentää ja laajentaa esimerkiksi animaatio-objektijoukon yksittäisen objektin tietoja käyttäjän niin halutessa plus- ja miinuspainikkeista. Tämä mahdollisuus liittyy myös periaatteen *käyttäjän kontrolli ja vapaus* kanssa, mistä tosin selkeämpi esimerkki on solmut ja segmentit -osion mahdollisuus poistaa polun osasia suoraan käyttöliittymän punaisten painikkeiden kautta. Tällaiset poistopainikkeet ovat myös yhdenmukaisia työpöytäohjelmistojen käyttöliittymästandardien kanssa periaatteen *standardit* mukaisesti.

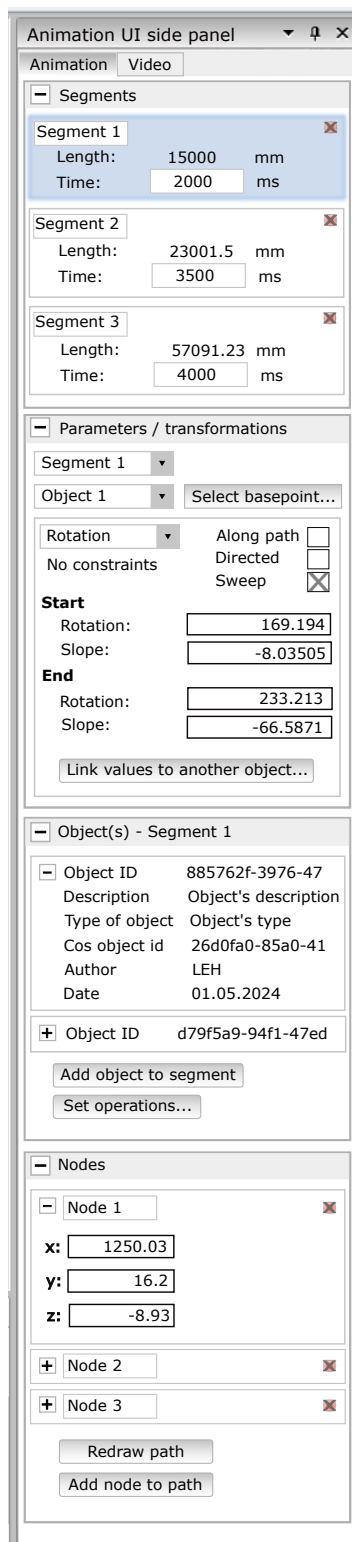
Standardien mukaisia ovat myös arvojen näyttäminen muokattavissa kentissä, joita klikkaamalla käyttäjä pääsee muokkaamaan esimerkiksi segmentteihin kuluvaan aikaan suoraan. Tämä on paitsi 3D-suunnitteluun erikoistuneen *suora manipulointi* periaatteen mukaista, myös yhdenmukaista yleisen käyttöliittymäperiaatteen *järjestelmän tilan näkyvyys* kanssa. Mikäli käyttäjä muokkaa polun osan arvoja käyttöliittymässä, se heijastuu myös suunnittelunäkymään ja toisinpäin. Arvokenttiin on mahdollista yhdistää myös 3D-suunnittelukäytännön *automaattinen tarkistus* mukainen soveltuvien arvojen tarkistus. Esimerkiksi pituuden kenttään ei ole tarkoituksenmukaista mahdollistaa negatiivista arvoa. Tämä on yhtenevää myös yleisen periaatteen *virheiden ennaltaehkäisy* kanssa.

Ensimmäisen iteraation jälkeen oli nähtävissä, että vaikka vaadittu perustoiminnallisuus oli toteutettu, kaikkia vaatimusmäärittelyn kohtia ei ollut täytetty. Kuviossa 2 on nähtävillä käyttöliittymäsuunnitelman toinen iteraatio, johon lisäksi täydentäviä toiminnallisuuksia. Toisessa iteraatiossa sovelsin myös edelleen periaatetta *organisoii tieto sijainneittain* ja erotin segmentit ja solmut omiksi osioikseen, sillä ajattelin sen selkeyttävän käyttöliittymää. Ajatuksena oli myös, että se olisi periaatteiden *käyttäjän kontrolli ja vapaus* ja *toimintojen jakaminen pienempiin osiin* sekä 3D-suunnittelukäytäntöön *dialogin tiedon levittäminen* mukaista, sillä yksittäisestä solmupisteestä olisi käyttäjän niin halutessa mahdollisuus näyttää enemmän tietoja ja solmujen parametreja voisi muuttaa laajemmassa näkymässä.

Käyttäjän hallinnan ja vapauden lisäämiseksi käyttöliittymän objektit-osioon lisättiin erilliset painikkeet objektien lisäämiseksi valittujen objektien joukkoon ja ohjelman joukkooperaatiotyökalun käynnistävä painike. Lisäksi lisättiin painikkeet koko polun uudelleen piirtämiseksi sekä solmupisteen lisäämiseksi animaatiopolkuun. Tämän nähtiin laajentavan käyttäjän mahdollisuuksia muokata animaatiopolkua ja lisäävän myös toiminnallisuuden *tehokkuutta*, kun polun määrittelyn saattoi aloittaa uudelleen erillisestä painikkeesta käyttöliittymässä.

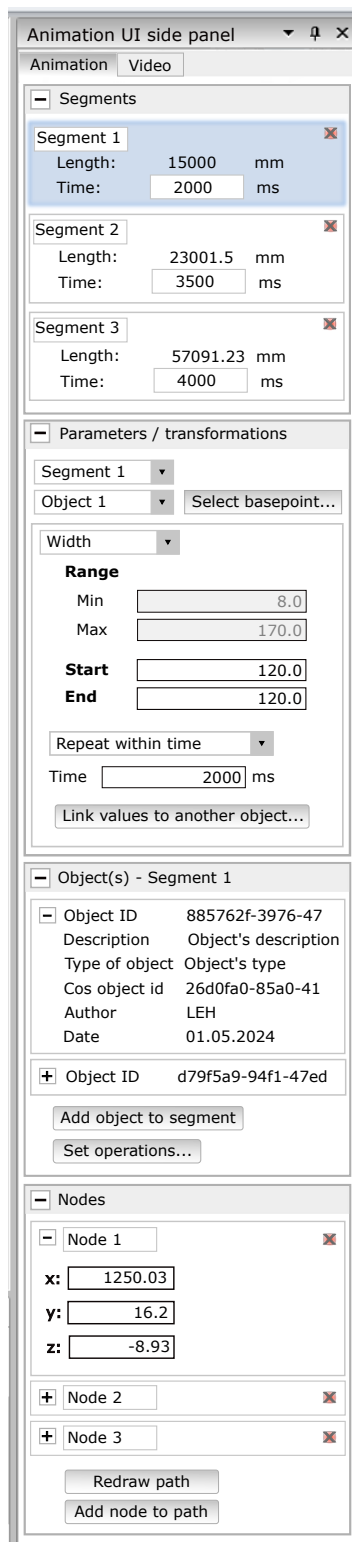
Toisessa iteraatiossa on myös muokattu transformaatiot-osiota sisällyttäen siihen parametrien muokkaus vaatimusmäärittelyn mukaisesti. Lisäksi pudotusvalinnoilla on mahdollisuus valita halutun segmentin haluttu objekti, jonka parametreja tai transformaatioita määritetään. Lisäksi vaatimusmäärittelyn mukaisesti lisättiin painike, josta voidaan määrittää objektin peruspiste, jonka suhteen transformaatiot, kuten esimerkiksi kierto, tapahtuvat. Samalla käyttöliittymään sisällytettiin painike, jolla transformaatioiden tai parametrien arvot voidaan kiinnittää seuraamaan toisen objektin vastaavia arvoja.

Eri parametrien ja painikkeiden lisäämisen taustalla oli pyrkimys lisätä *käyttäjän kontrollin* lisäksi myös toiminnon käytön *joustavuutta*. Tämä vastaa taulukon 6 käytännettä *ymmärrä toiminnot*, sillä kyseinen toiminnallisuus on käytännöllinen animaatiotoiminnon käyttötapausta mietittäessä: usein tarpeena on liikuttaa objektijoukkoa esimerkiksi yhdessä tai tietyllä tavoin suhteessa toisiinsa.



Kuvio 2. Toinen iteraatio: kiertonäkymä

Toiseen iteraatioon kuuluu myös kuvio 3, jossa on näkyvissä objektin parametrien muokausmahdollisuus polun segmentillä. Työpöytäohjelmien *standardien* mukaisesti arvokentät ovat harmaana, mikäli parametrin muokkaus ei kyseissä tilanteissa ole sallittua. Tämä *värien käyttökäytäntö* on varsin suora tapa ilmentää *järjestelmän tilan näkyvyyttä*, mikäli esimerkiksi objektin tietyn parametrin muokkaaminen ei ole sallittua. Vaatimusmäärittelyn mukaisesti parametrin muokkausnäkömään lisättiin myös mahdollisuus valita parametrin muutokselle tyyli, esimerkiksi muutosta toistaen tietyn ajan sisällä.

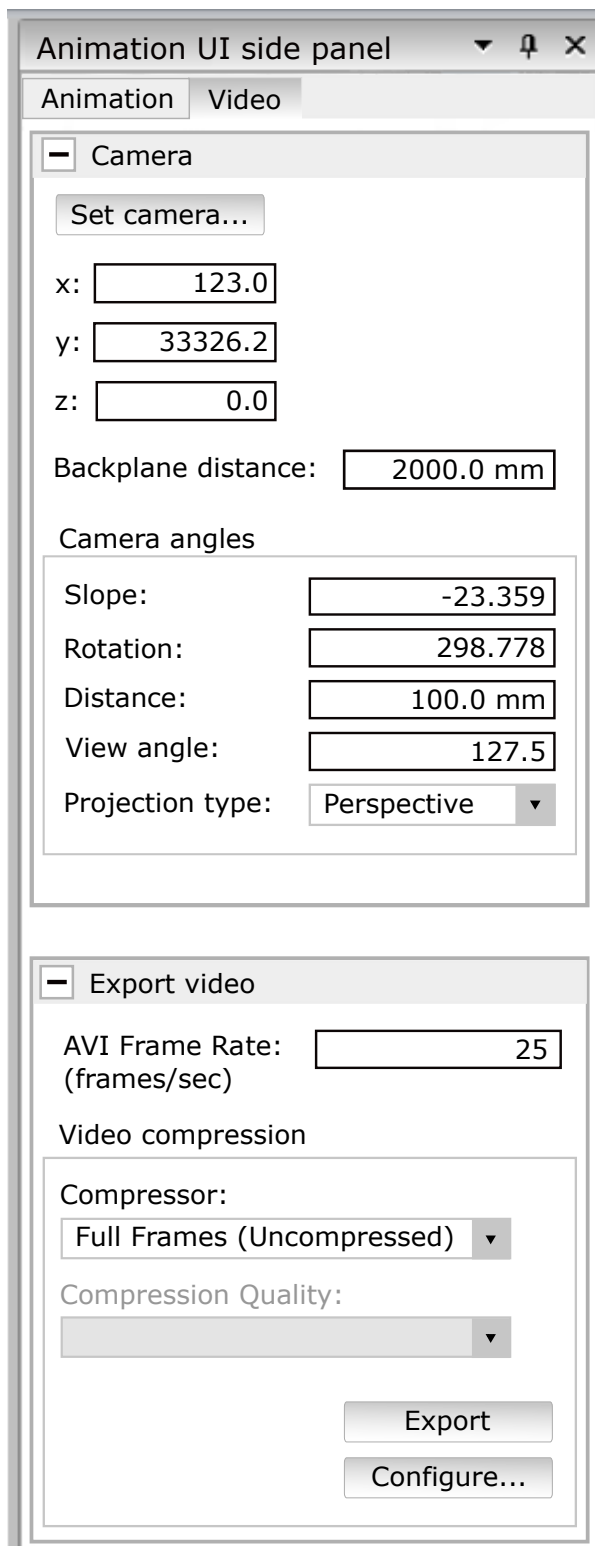


Kuvio 3. Toinen iteraatio: leveysnäky

Toisessa iteraatiossa käyttöliittymään lisättiin vaatimusmäärittelyn mukaisesti myös videon tuottamisen asetukset kuviossa 4 nähtävällä tavalla. Videon tuottaminen nähtiin erilliseksi kokonaisuudeksi animaation objektien asetuksista, jolloin periaatteita *organisoi tieto sijainneittain ja toimintojen jakaminen pienempiin osiin* soveltaen kyseiset asetukset sijoitettiin omalle välilehdelleen.

Näkymän jakaminen eri välilehdille mahdollistaa suuremman määrän erilaisia asetuksia näkyvissä dialogeissa vähentämättä suunnittelutilan määrää, jolloin toimitaan 3D-suunnitteluun erikoistuneen käyttöliittymäperiaatteen *työtilan maksimoiminen* mukaisesti. 3D-suunnittelun käytänteiden osalta kyseisen periaatteen alle tässä suunnittelutapauksessa voidaan nähdä asettuvan paitsi *tärkeimpien dialogien näkyvyys* myös *dialogin tiedon levittäminen*, sillä näkyvillä olevan animaatiokäyttöliittymän eri dialogien määrittämissä mahdollisuuksissa ovat välilehdille jakamisen edelleen helposti saavutettavissa ja suunnittelutila pysyy näkyvillä.

Suunnittelutilan maksimoimisen periaatetta noudattamalla olisi voitu päätyä myös käytänteeseen *valikkojen ja toimintojen näkyvyyden rajoittaminen*. Koska tässä tapauksessa asetusten jakaminen välilehdille oli mahdollista ja tuotettavan videon parametrit ovat kiinteästi yhteydessä animaatiotoimintoon, olisi tuntunut irralliselta laittaa videoasetukset erilliseen ikkunaan tai oman valikkovalinnan taakse. Jossakin toisessa tilanteessa toisenlainen valinta olisi voinut olla perustellumpi, mistä päästään siihen, ettei tällaisten periaatteiden avulla yksinään mekaanisesti kehitetä käyttöliittymiä, vaan vasta suunnittelijan omat valinnat luovat lopullisen käyttöliittymäartefaktin.



Kuvio 4. Toinen iteraatio: videovälilehti

Käyttöliittymäkehityksen suorittaminen toisteisesti jakaen kehityksen tässä esiteltyihin erillisiin *iteraatioihin* noudattaa paitsi tutkielman teoriaosuudessa kuvattua suunnittelututkimuksen menetelmää (Weber 2010, s. 2) myös taulukossa 6 esiteltyjä kirjallisuuskatsauksessa kerättyjä käyttöliittymäkehityksen parhaita käytänteitä. Näistä kohdista kehitysprosessi seurasi myös *varhaisen visualisoinnin* käytännettä, koska kehitysprosessi lähti liikkeelle käyttöliittymän ulkoasun suunnittelusta.

Iteraatioita arvioimalla suunnitelma-artefaktia kehitettiin edelleen. Tässä tutkielmassa iteraatioiden arviointi tapahtui pääasiassa kirjoittajan toimesta, mutta toisen kehitysiteraation jälkeen palautetta pyydettiin lisäksi yrityksen sisältä suunnitelman vaikuttavuuden arvioimiseksi ja kehityskohteiden tunnistamiseksi. Saadun palautteen pohjalta kehitettiin käyttöliittymän kolmas ja tutkielman puitteissa viimeinen versio, joka on esitetty kuviossa 5.

Kolmannessa iteraatiossa on avattu alavetovalikoiden vaihtoehdot, jotka suunniteltiin vaatimusmäärittelyn pohjalta kattamaan siinä mainitut tarpeet. Käyttöliittymäsuunnitelman tässä vaiheessa suoritettiin palautteen pohjalta yhdistämistä ja karsintaa, pitkälti periaatteiden *johdonmukaisuus* ja *minimalismi* pohjalta. Esimerkiksi animaatiopolun solmut ja segmentit yhdistettiin samaan näkymään yksinkertaistaen niiden näyttämistä ja niiden muokkaamiseksi tarkoitettuja painikkeita, jolloin yhdestä lisää-painikkeesta lisätään sekä solmu että segmentti.

Solmujen ja segmenttien näyttäminen samassa käyttöliittymäelementissä vähentää ylimääräistä hierarkiaa käyttöliittymässä ja selkeyttää käyttöliittymää visuaalisesti. Tämän lisäksi sen voidaan ajatella ilmentävän paremmin animaatiopolun rakentumista vuorotteleviin solmukohtiin ja niiden välisiin osioihin. Sen kautta tämän muutoksen voi nähdä myös edustavan periaatetta *minimoi käyttöliittymä, maksimoi sisältö* tukien toiminnon ymmärtämistä ja antaen käyttäjälle enemmän informaatiota vähentäen samalla ylimääräisten käyttöliittymäelementtien määrää. Tämän vuoksi muokkauksissa noudatettiin siis käyttöliittymäkehityksen parhaista käytänteistä käytäntöä *käytä käyttöliittymässä sisällön esittämiseen soveltuvaa asettelua*.

Johdonmukaisuuden periaatetta kolmannessa iteraatiossa ilmentää esimerkiksi objektien listauksen muokkaaminen samanlaiseksi kuin muut ohjelmistossa käytettävät vastaavat listauk-

set. Lisäksi joukko-operaatioiden painikkeet muokattiin vastaamaan paremmin ohjelmiston yleistä linjaa. Parametrien ja transformaatioiden osalta käyttöliittymää muokattiin vastaamaan paremmin käyttäjän työjärjestystä periaatetta *järjestelmän ja maailman välinen yhteensopivuus* noudattaen ja taulukon 7 valikoiden rakenne työvaiheiden mukaan -käytänteen mukaisesti.

| Objects |
|---------|
| 2 |
| 2 |
| 2 |
| 1 |

| Set |
|---------|
| Object1 |
| Object2 |

| Width |
|---------------|
| Rotation |
| Speed |
| Add params... |

| Linear |
|--------------------|
| Accelerating |
| Slowing down |
| Repeat within time |

✕
Animation UI side panel

Animation Video

Path

Route Add Delete

| Path elem | Coord / Length | Duration | |
|-----------|----------------|----------|---|
| Node1 | 200,5000,100 | 0ms | ▲ |
| Segment1 | 4000mm | 4000ms | |
| Node2 | 200,5000,300 | 300ms | |
| Segment2 | 2535mm | 4400ms | |

Object(s)

Objects in set

| | |
|--------------------|---|
| Standard Component | 2 |
| Total | 2 |

Add to set
Remove from set
Preview

Parameters / transformations

Set Select basepoint...

Width

Range

| | |
|--------------|-------|
| Min | 8.0 |
| Max | 170.0 |
| Start | 120.0 |
| End | 120.0 |

Repeat within time

| | | |
|------|------|----|
| Time | 2000 | ms |
|------|------|----|

Link values to another object...

Kuvio 5. Kolmas iteraatio

5 Yhteenveto ja pohdinta

Tässä tutkielmassa haasteita tuottivat katsauksessa löytyneen kirjallisuuden alustava määrä sekä niissä esiintyneiden erilaisten käyttöliittymäperiaatteiden suuri määrä ja kirjavuus. Näitä haasteita pyrittiin rajaamaan hakumenetelmien kautta lähdekirjallisuuden ajanjaksollisen rajaamisen sekä laadullisen rajaamisen avulla. Haussa rajattiin pois erikoistuneet käyttöliittymäkehityspaneilla lukuunottamatta suunnittelututkimuksessa kehitetyn artefaktin kanalta relevantteja alakohtaisia periaatteita ja käytänteitä.

Muut yleiset periaatteet ja käytänteet sisällytettiin sillä perusteella, etteivät ne olleet mihinkään muuhun erikoisalaan tai tiettyyn kontekstiin tai teknologiaan viittaavia. Tällä perusteella katsauksen ulkopuolelle jääneet artikkelit käsittelivät esimerkiksi tekoälyn käytössä hyödynnettäviä periaatteita tai mobiilikehitykseen erikoistuneita suunnitteluperiaatteita. Lisäksi kirjallisuutta rajattiin koskemaan vain vuodesta 2000 alkaen julkaistua kirjallisuutta ja edelleen laadullisin kriteerein kattamaan vain monipuolista lähdemateriaalia käyttäviä tai dokumentoituin menetelmin kehitettyjä periaatteita sisältäviä artikkeleita.

Katsauksen rajoitusten perusteella tutkielmasta saadut tulokset rajoittuvat yleisiin käyttöliittymäkehityksen periaatteisiin, minkä myötä jollekin erikoistuneelle osa-alueelle kehitetyt periaatteet voivat erota tässä esitetyistä. Erikoistuneemmat periaatteet voisivat olla hedelmällinen lähtökohta jatkotutkimusta ajatellen. Yleisyyden lisäksi tutkielmassa tehdyssä periaatteiden yhdistämisessä ja luokittelussa tehtiin tulkintoja, jotka joku toinen olisi saattanut tehdä eri tavalla johtuen katsauksessa löytyneiden periaatteiden erilaisista ilmaisutavoista ja toisaalta samojenkin käsitteiden monitulkintaisuudesta.

Läpinäkyvyyden ja toistettavuuden lisäämiseksi luokittelun pohjaksi pyrittiin kartoittamaan laajemmin paitsi yksittäisiä käsitteitä myös kirjoittajien periaatteille antamia selittäviä tarkennuksia. Toin esille luokittelun selventämiseksi taulukossa 3 kolmannessa sarakkeessa joko selityksen periaatteelle tai kirjallisuudessa käytettyjä muita muotoiluja, joilla tulkitsein tarkoitettavan samaa asiaa.

Tutkimusmenetelmänä kirjallisuuskatsaus oli työläs, mutta tuotti hyvin aineistoa kehityksen perustaksi. Lisäksi kirjallisuuskatsauksen kautta suunnittelututkimusosuuden teoreetti-

nen pohjustus oli tuloksekasta. Saadut kymmenen yleisintä suunnitteluperiaatetta ovat miltei samat kuin Nielsenin kymmenen heuristiikkaa ja Shneidermanin ”kahdeksan kultaista sääntöä”, mikä osaltaan kertonee periaatelistauksien vakiintuneesta asemasta alan kirjallisuudessa ja tämän tutkielman tulosten voi nähdä vahvistavan niiden asemaa alan teoreettisessa viitekehysessä. Benaida (2023, s. 9694)

Vaikka jotkin kirjoittajat ovat kritisoineet esimerkiksi Nielsenin heuristiikoita muun muassa ohjelmistokeskeisyydestä, esimerkiksi Benaida (2023, s. 9693–9695) toteaa, että ne ovat edelleen käyttökelpoisia ja alalla hyödynnettyjä. Koska tutkimuskysymyksenä oli selvittää kirjallisuudesta löytyvät kehityspeiraatteet, kirjallisuuskatsaus oli menetelmänä nähdäkseni tutkielmaan tarpeisiin soveltuva. Mikäli olisi haluttu löytää uusia ja vielä vakiintumattomia kehityspeiraatteita, niiden löytämiseen olisi soveltunut paremmin jokin toinen lähestymistapa.

Menetelmänä suunnittelututkimus soveltui hyvin työpaikalta saadun tehtävän suorittamiseen ja käyttöliittymäartefaktin kehitys antoi arvokasta lisätietoa suunnitteluperiaatteiden ja -käytänteiden soveltamisesta. Haasteena menetelmän käytössä oli ehkä yleisestikin suunnittelututkimuksen paikanhaku tieteen kentällä: miten käytännön kehitystyön kautta voi saavuttaa uutta tieteellistä tietoa ja miten se eroaa jokapäiväisestä kehitystyöstä (Hevner ym. 2004, s. 81)? Tässä tutkielmassa haasteeseen vastattiin pohjaamalla kehitystyö kirjallisuuskatsauksessa löydettyihin periaatteisiin ja käytänteisiin, minkä lisäksi tarkasteltiin näiden periaatteiden ja käytänteiden soveltamista käyttöliittymäsuunnitelma-artefaktin kehittämässä, mihin suunnittelututkimus antoi hyvän lähtökohdan.

Artefaktin kehitystyön pohjaaminen kirjallisuuskatsauksessa koottuihin periaatteisiin ja käytänteisiin sai aikaan myös sen, että kehitystyön valinnat tuntuivat perustelluilta. Vaikka jotkin kehitysratkaisut muuttuivat suunnitelma-artefaktin eri iteraatiovaiheissa, ratkaisujen pohjina käytettyjen periaatteiden asettaminen vastakkain ja niiden vaikuttavuuden vertaaminen opasti kehitysratkaisujen teossa ja sai niihin päätyminen tuntumaan oikeutetummalta. Lisäksi kun ratkaisun taustalla oli jokin selkeästi sanoitettava periaate tai käytäntö, myös sen painoarvoa oli helpompi arvioida. Mikäli eri ratkaisuilla oli eriävät periaatteet perusteluinaan, oli helpompi arvioida ikään kuin abstraktimmalla tasolla, kumpaa periaatetta pidettiin kulloisessakin tilanteessa tärkeämpänä ja miten se vaikutti artefaktin kokonaisuuden kannalta.

Tutkielmassa toteutettu artefakti täyttää sille asetetut toiminnalliset vaatimukset ja sen kehittämisesä käytetty toisteinen ja inkrementaalinen toteutustapa arviointineen vaikuttivat suotuisasti sen ulkomuotoon. Erilaiset pidemmälle menevät arviointi- sekä testausmenetelmät, kuten yksikkö- ja käyttäjättestaus olisivat antaneet tarkempaa tietoa artefaktin vaikuttavuudesta, mutta mielestäni kehitetty käyttöliittymäsuunnitelma antaa hyvän pohjan lähteä toteuttamaan animaatioiden määrityksen käyttöliittymää.

Kaiken kaikkiaan periaatteet ovat mielestäni hyödynnettävissä käytännön käyttöliittymäkehityksen pohjana tai vähintäänkin tukena, joskin niiden käyttökelpoisuus riippunee myös UI-kehittäjän kokemuksesta ja kehitysprosessin systemaattisuudesta: jos on alusta asti selvää, että käyttöliittymä tulee kehittää tietyn prosessin mukaisesti, omaperäisempää ratkaisua ei tarvitse tehdä vaan voi seurata prosessin vaiheita ja sen mukaisia käytänteitä.

Mikäli haluaa hyödyntää käyttöliittymäkehityksen yleisiä periaatteita, kehityksessä tulisi niiden kohdalla tulisi pysähtyä miettimään, mitä kunkin periaatteen noudattaminen tarkoittaa kehitettävän ohjelmiston kannalta. Aloittelevalle kehittäjälle siitä voi olla hyötyä turhan työn vähentämisessä, käyttöliittymän jäsentämisessä ja kehitysratkaisujen perustelussa. Kokeneemmalle suunnittelijalle sen arvo voi ilmetä ehkäpä tarkistuslistamaisen käytön myötä esimerkiksi käyttöliittymän arvioinnissa.

Periaatteiden hyödyntäminen kehitystyössä voi helpottaa kehittäjän työtä etenkin jos periaatteiden merkitys on selvitetty. Mikäli periaatteita on verrattu keskenään ja ne on priorisoitu etukäteen esimerkiksi vaatimusmäärittelyn kautta tai mikäli kehitysprosessi on selkeästi muotoiltu joidenkin tiettyjen periaatteiden pohjalta tapahtuvaksi, itse kehityksen aikana tehdyt ratkaisut ja tulkinnat helpottuvat, koska niitä ei tarvitse tehdä tyhjiössä, vaan niissä voi tukeutua valittuihin periaatteisiin.

Jatkotutkimusta aiheesta voisikin tehdä esimerkiksi kehitysperiaatteiden käytännön soveltamisesta: voisiko siihen kehittää etenemistavan, joka auttaisi kehitysprosessin nopeuttamisessa ja laadun parantamisessa. Systemaattisempi soveltaminen voisi auttaa myös aloittelevia käyttöliittymäkehittäjiä ja toimia myös koulutuksellisenä välineenä. Edelleen tutkimusta voisi tehdä löydettyjen periaatteiden arvioimiseksi ja parantamiseksi sekä mahdollisesti kokonaan uusien kehitysperiaatteiden kehittämiseksi.

Yksi näkökulma jatkotutkimuksen pohjaksi voisi olla myös alakohtaisten ja erikoistuneiden periaatteiden sekä yleisten kehitysperiaatteiden välinen suhde: onko mahdollista kehittää jokin menetelmä erikoistuneiden käytänteiden ja periaatteiden johtamiseksi yleisistä periaatteista. Toisaalta asetelman voisi kääntää myös toisin päin, eli onko erikoistuneista periaatteista mahdollista laajentaa yleisemmät periaatteet, joista ne on johdettu? Tämä olisi myös mielenkiintoinen tapa arvioida periaatteita ja analysoida tarkemmin niiden käyttötapoja ja -mahdollisuuksia.

Lähteet

- Badashian, Ali Sajedi, Mehregan Mahdavi, Amir Pourshirmohammadi ja Minoos Monajjemi Nejad. 2008. "Fundamental Usability Guidelines for User Interface Design". Teoksessa *2008 International Conference on Computational Sciences and Its Applications*, 106–113. <https://doi.org/10.1109/ICCSA.2008.45>.
- Benaida, Mohamed. 2023. "Developing and extending usability heuristics evaluation for user interface design via AHP". *Soft Computing*, 9693–9707.
- Blair-Early, Adream, ja Mike Zender. 2008. "User interface design principles for interaction design". *Design Issues* 24 (3): 85–107.
- Forlizzi, Jodi, ja Katja Battarbee. 2004. "Understanding experience in interactive systems". Teoksessa *Proceedings of the 5th conference on Designing interactive systems: processes, practices, methods, and techniques*, 261–268. ACM.
- Gómez Reynoso, J. M., ja L. Olfman. 2012. "The impact of combining gestalt theories with interface design guidelines in designing user interfaces", 2367–2374. 3. AMCIS.
- Hassenzahl, Marc, ja Noam Tractinsky. 2006. "User experience – a research agenda". *Behaviour & information technology* 25 (2): 91–97.
- Hevner, Alan R., Salvatore T. March, Jinsoo Park ja Sudha Ram. 2004. "Design Science in Information Systems Research". *MIS Quarterly* 28, numero 1 (maaliskuu): 75–105.
- Kamper, Robert J. 2002. "Extending the usability of heuristics for design and evaluation: Lead, follow, and get out of the way". *International journal of human-computer interaction* 14 (3-4): 447–462.
- Kitchenham, Barbara, ja Stuart Charters. 2007. *Guidelines for performing Systematic Literature Reviews in Software Engineering*. Tekninen raportti. Keele, Staffs ST5 5BG, UK: Keele University, tammikuu.
- Kuniavsky, Mike. 2007. "User experience and HCI". Teoksessa *The Human-Computer Interaction Handbook*, 923–942. CRC Press.

- Lee, Ghang, Charles M. Eastman, Tarang Taunk ja Chun-Heng Ho. 2010. “Usability principles and best practices for the user interface design of complex 3D architectural design and engineering tools”. *International Journal of Human-Computer Studies* 68 (1): 90–104. ISSN: 1071-5819. <https://doi.org/https://doi.org/10.1016/j.ijhcs.2009.10.001>.
- Peppers, Ken, Tuure Tuunanen, Charles E. Gengler, Matti Rossi, Wendy Hui, Ville Virtanen ja Johanna Bragge. 2006. *The Design Science Research Process : A Model for Producing and Presenting Information Systems Research*, 83–106.
- Pitale, Anand, ja Afreen Bhungara. 2019. “Human Computer Interaction Strategies — Designing the User Interface”. Teoksessa *International Conference on Smart Systems and Inventive Technology (ICSSIT)*, 752–758. <https://doi.org/10.1109/ICSSIT46314.2019.8987819>.
- Ruiz, Jenny, Estefanía Serral ja Monique Snoeck. 2021. “Unifying Functional User Interface Design Principles”. *International Journal of Human–Computer Interaction* 37 (1): 47–67. <https://doi.org/10.1080/10447318.2020.1805876>.
- Silva, Carina da, Eugenio Merina, Giselle Merino ja Luiz Fernando de Figueiredo. 2015. “Usability principles and measures for the design of a product applied to the design management”. *Ergodesign & HCI* 3 (2): 28–37.
- Torres, R. J. (Bob), Michael P. Heck, James R. Rudd ja John F. (Jeff) Kelley. 2009. “Usability Engineering: “Best of the Best” Best Practices”. *Ergonomics in Design* 17 (3): 20–25. <https://doi.org/10.1518/106480409X12487281637658>. <https://doi.org/10.1518/106480409X12487281637658>.
- Weber, Sven. 2010. “Design science research: Paradigm or approach?” Teoksessa *Proceedings of the Sixteenth Americas Conference on Information Systems*, 214–222. AMCIS, elokuu.
- Venable, John, Jan Pries-Heje ja Richard Baskerville. 2012. “A Comprehensive Framework for Evaluation in Design Science Research”. *LNCS 7286 (toukokuu)*: 423–438.