

**Eerik Kukkonen**

**Luovien musiikkipelien parantaminen: musiikinluontipelin  
kehitysprosessi ja evaluointi**

Tietotekniikan pro gradu -tutkielma

29. toukokuuta 2018

Jyväskylän yliopisto

Informaatioteknologian tiedekunta

**Tekijä:** Eerik Kukkonen

**Yhteystiedot:** eerik.kukkonen@outlook.com

**Ohjaajat:** Ville Isomöttönen ja Jukka Varsaluoma

**Työn nimi:** Luovien musiikkipelien parantaminen: musiikinluontipelin kehitysprosessi ja evaluointi

**Title in English:** The improvement of creative music games: the development process and evaluation of a music creation game

**Työ:** Pro gradu -tutkielma

**Suuntautumisvaihtoehto:** Pelit ja pelillisuus

**Sivumäärä:** 77+3

**Tiivistelmä:** Musiikkipeleistä rytmipelit ovat nousseet selkeästi suurempaan suosioon kuin luovat musiikkipelit, joissa pelaaja voi soittaa ja säveltää musiikkia. Tässä tutkimuksessa selvitettiin musiikinluontipelin (ML-peli) kehitysprosessia ja kuinka evaluoida kehitettyä ML-peliä. Esitän ML-pelin uutena käsitteenä musiikkipelien olemassa olevaan teoriaan. Kehitin mobiilialustalle ML-pelin prototyypin, jota verrattiin julkaistuihin mobiilipeleihin. Tulosten mukaan ML-peli voi tarjota miellyttävän ja opettavan pelikokemuksen. ML-pelin tulisi tarjota pelaajalle selkeitä pitkäaikaisia tavoitteita, jotta peli vetäisi paremmin vertoja julkaistuihin mobiilipeleihin. Hyvän pelisuunnittelun omaava ML-peli voi avata musiikinluonnin yleisölle, joka ei ole aiemmin harrastanut musiikinluontia. Tutkimuksesta saadun suunnittelutietämyksen huomioinnilla voidaan parantaa luovien musiikkipelien tunnetuksi tulemisen mahdollisuutta.

**Avainsanat:** musiikkipeli, musiikinluontipeli, instrumenttipeli, sävellyspeli, pelinkehitys, pelit

**Abstract:** In the genre of music games rhythm games have become a lot more popular than creative music games which provide the opportunity to play and compose music. In this research I looked into the development process of music creation game (MC game) and how

to evaluate the created MC game. I present the MC game as a new concept to the music game literature. I developed an MC game prototype for the mobile platform and the prototype was compared to published mobile games. According to the results MC game can provide a pleasant and educational game experience. The MC game should provide clear long-term goals in order to compare better against published mobile games. An MC game with good game design can bring music creation to an audience that haven't had previous experience of music creation. When taking into account the design knowledge acquired through this research the possibility of creative music games to get more awareness can be increased.

**Keywords:** music game, music creation game, instrument game, composition game, game development, games

## **Esipuhe**

Nyt se on valmis. Kiitos Jeesus, kiitos ystävät, kiitos vanhemmat. Kiitos Jukka Varsaluoma ja Ville Isomöttönen graduohjauksesta. Kiitos pelitestaajille, jotka vielä gradun loppuvaiheessa pääsivät auttamaan työssäni.

Jyväskylässä 29. toukokuuta 2018

Eerik Kukkonen

## Kuviot

Kuvio 1. Sentris ja ympyränmuotoinen musiikkikone. Lähde: <a href="https://timbreinteractive.squarespace.com/sentris/">https://timbreinteractive.squarespace.com/sentris/</a> .....	9
Kuvio 2. Moondust. Lähde: <a href="http://www.lemon64.com/?mainurl=http%3A//www.lemon64.com/games/details.php%3FID%3D1709">http://www.lemon64.com/?mainurl=http%3A//www.lemon64.com/games/details.php%3FID%3D1709</a> .....	11
Kuvio 3. Animal Crossingin sävellystila ja sekvensseri .....	13
Kuvio 4. Starship Titanicin musiikkihuone .....	13
Kuvio 5. Wario Land 4 ja musiikin keräily. Lähde: <a href="https://www.mariowiki.com/File:Soundroom.PNG">https://www.mariowiki.com/File:Soundroom.PNG</a> .....	14
Kuvio 6. Tenori-On .....	18
Kuvio 7. Electroplanktonin Hanenbow-pelimuoto. Lähde: <a href="https://www.nintendo.com/games/detail/B4QgT8uckp1VMV3w1DQwPSWeFA11Bbr7">https://www.nintendo.com/games/detail/B4QgT8uckp1VMV3w1DQwPSWeFA11Bbr7</a> .....	24
Kuvio 8. One Man Band ja free play -pelimuoto .....	25
Kuvio 9. Birnbaumin ym. kokemusperäinen malli .....	26
Kuvio 10. Magnussonin episteminen malli .....	28
Kuvio 11. Suunnittelututkimuksen kontribuutioviitekehys .....	35
Kuvio 12. Kehitysprosessin merkittävät vaiheet .....	36
Kuvio 13. Äänensoittoluuppi tahti kerrallaan .....	39
Kuvio 14. Äänensoittoluuppi kahdeksasosanuotin välein .....	41
Kuvio 15. Pelinäkymä. ....	45
Kuvio 16. Musiikkinäkymä. ....	45
Kuvio 17. Musiikinluontipelin arkkitehtuurikuvaus .....	47
Kuvio 18. Musiikkipelien ominaisuudet .....	51
Kuvio 19. Uusittu ominaisuustaulukko .....	53
Kuvio 20. Prototyypin arviointi Birnbaumin ym. mallilla. ....	55
Kuvio 21. Prototyypin arviointi Magnussonin mallilla. ....	55
Kuvio 22. Flow: haasteen ja taitojen tasapaino .....	57
Kuvio 23. Arvosanoja ja kehityksen kohteita .....	62
Kuvio 24. Siirtyminen tehtävänäkymään .....	63
Kuvio 25. Jäljellä olevien nuottien määrä ilmaistuna vasemmassa alakulmassa. ....	63

# Sisältö

1	JOHDANTO .....	1
2	PELI JA MUSIIKKI .....	2
2.1	Peli .....	2
2.2	Pelisuunnittelu .....	4
2.3	Musiikki, improvisaatio ja säveltäminen .....	5
3	MUSIIKKIPELIT: RYTMII- JA INSTRUMENTTIPELIT .....	7
3.1	Rytmipelit .....	7
3.2	Instrumenttipelit .....	8
3.3	Musiikinluontipelit .....	9
3.4	Muita musiikillisia ja äänen ilmentymiä .....	11
3.4.1	Sekvensserit pelien sisällä ja hiekkalaatikkopelit .....	12
3.4.2	Instrumentti- ja sekvensseriohjelmistot pelialustoilla .....	12
3.4.3	Musiikilliset puzzlet peleissä .....	13
3.4.4	Musiikin keräily peleissä .....	13
3.4.5	Audiopelit .....	14
4	LUOVIEN MUSIIKKIPELIIEN TEOREETTINEN POHJA .....	16
4.1	Pichlmairin ja Kayalin instrumenttipelit ja niiden ominaisuudet .....	16
4.1.1	Kvantisointi .....	16
4.1.2	Aktiivinen nuotti .....	17
4.1.3	Ääniagentit .....	19
4.1.4	Leikki .....	19
4.2	Lieben interaktiivisuus ja musiikki peleissä .....	20
4.2.1	Pelaajan tai tietokoneen toiminta .....	20
4.2.2	Käyttöliittymän ja pelitapahtumien suhde .....	20
4.2.3	Ääniraidan tyyppi .....	21
4.3	Herberin sävellysinstrumentti .....	22
4.3.1	Asynkronisuus .....	22
4.3.2	Generatiivisuus .....	23
4.3.3	Emergenssi .....	23
4.4	Bott ym. musiikin luomisesta peleissä eleillä .....	23
4.5	Digitaalisten instrumenttien analysoimisesta .....	25
4.5.1	Birnbaumin ym. kokemuseräinen malli .....	26
4.5.2	Magnussonin tietoperäinen malli .....	27
4.5.3	Viitekehyksiä instrumentin suunnitteluun .....	29
4.6	Muuta pelimusiikkikirjallisuutta .....	30
5	KONSTRUKTIO: MUSIIKINLUONTIPELI .....	32
5.1	Vaatimusmäärittelyä .....	33
5.2	Suunnittelututkimuksesta .....	34
5.3	Suunnittelututkimuksesta tässä työssä .....	34

5.4	Unity, käytänteitä ja suunnittelumalleja .....	37
5.5	Musiikkikomponentti .....	38
5.5.1	Säveltämistä tukevat ominaisuudet .....	38
5.5.2	Äänensoittoluuppi .....	39
5.5.3	Äänen sävelkorkeuden muuttaminen .....	40
5.5.4	Sävelasteikot .....	42
5.5.5	Unityn Audio Mixer .....	42
5.6	Pelikomponentti .....	43
5.7	Peli- ja musiikkikomponentin yhteenliittäminen .....	44
5.8	Musiikinluontipelin arkkitehtuurista .....	46
6	MUSIIKINLUONTIPELIN ARVIOINTI .....	49
6.1	Musiikinluontipelin määritelmän toteutuminen.....	49
6.2	Reflektio luovien musiikkipelien kirjallisuuteen.....	49
6.3	Prototyypin arviointi instrumenttina.....	54
6.4	Prototyypin arviointi pelisuunnittelullisesti .....	56
6.5	Systemaattinen pelitestaus ja RGT-menetelmä .....	59
6.6	Prototyypin parantelu .....	61
7	POHDINTA JA JOHTOPÄÄTÖKSET .....	64
	LÄHTEET .....	67
	LIITTEET.....	71
A	RGT-menetelmän vertailupelit.....	71
B	RGT-menetelmän konstruktiot.....	72
C	Lähdekoodi soveltuvin osin.....	73

# 1 Johdanto

Musiikkipelit ovat tulleet tutuksi suurelle yleisölle erityisesti erilaisten rytmipelien muodossa kuten Guitar Hero ja Just Dance. Pelit, joissa pelaajalle annetaan mahdollisuus luoda musiikkia pelaamisen aikana, ovat melko harvinaisia ja suurelle yleisölle tuntemattomia. Musiikin soittamista tai säveltämistä tukevia pelejä kutsutaan luoviksi musiikkipeleiksi tai instrumenttipeleiksi. Luovat musiikkipelit poikkeavat usein perinteisistä peleistä siinä, että luovat musiikkipelit ovat tavoitteettomia. Näin niitä voidaan sanoa leluiksi. Tämä lelumainen luonne voi olla eräs suurimmista syistä, miksi luovat musiikkipelit ovat niin vähän tunnettuja. Tavoitteellisen luovan musiikkipelin eli musiikinluontipelin (ML-peli) kehittäminen on erityisesti vaikeaa pelisuunnittelun näkökulmasta. ML-pelin tulee olla yhtäältä peli, toisaalta helpokäyttöisenä musiikinluontityökaluna tukea säveltämistä ja samalla tarjota miellyttävä pelikokemus, mikä tekee pelisuunnittelusta haastavaa. Kun pelisuunnittelu on kohdallaan, ML-peli voi tarjota opettavaisen pelikokemuksen ja avata väylän musiikinluontiin yleisölle, joka ei ole aiemmin harrastanut musiikinluontia.

Tässä tutkimuksessa selvitetään musiikinluontipelin kehitysprosessia mobiilialustalla ja kuinka evaluoida kehitettyä ML-peliä kvalitatiivisesti. Konstruktiivisella tutkimuksella parannetaan ML-pelin ominaisuuksia kohti julkaistujen mobiilipelien ominaisuuksia. Kehitin Android-mobiilialustalle ML-pelin prototyypin. Kehitysprosessin viimeinen evaluointivaihe perustui Repertory Grid Technique -menetelmään (RGT). RGT-menetelmällä verrattiin kehitetyn prototyypin mobiilipelikokemusta julkaistujen mobiilipelien pelikokemukseen. Tulosten perusteella saatiin monia jatkokehityksen kohteita, joista yksi tärkeimmistä oli progressio eli pelin selkeät pitkäaikaiset tavoitteet. Progressio on usein piirre, joka jää vaillinaiseksi luovissa musiikkipeleissä.

Saadun suunnittelutietämyksen perusteella voidaan parantaa luovien musiikkipelien ominaisuuksia. Jotta pelikokemus saadaan julkaistujen mobiilipelien tasolle, tulee pelinkehittäjän huolehtia luovan musiikkipelin riittävästä ja mielenkiintoisesta progressiosta. Tutkielman tulosten yleistettävyyttä voitaisiin parantaa jatkotutkimuksella ja laajemmalla pelitestauksella. Progression ja muiden pelitestauksessa nousseiden ominaisuuksien huomioinnilla voidaan parantaa luovien musiikkipelien tunnetuksi tulemisen mahdollisuutta.



## 2 Peli ja musiikki

Tässä kappaleessa käsittelen tutkimukseni peruskäsitteitä, joiden pohjalle myöhemmät aiheet ja käsitteet rakentuvat. Pelin ja musiikin käsitteitä on olemassa monia. Kirjallisuudesta löytyneet määritelmät voivat juontua määrittelijän henkilökohtaisesta taustasta, näkökulmasta ja kulttuurista. Määrittelyä voidaan tehdä erilaisista lähtökohdista. Kaikkia määritelmiä tehdessä ei ole pyritty olemaan tieteellisen tarkkoja, vaan ollaan saatettu haluta korostaa jotain tiettyä havaintoa.

Pelejä katsotaan tässä työssä erityisesti pelisuunnittelun näkökulmasta, joten olen valinnut pelien käsitteet tältä kannalta. Tässä kohdin on omalla henkilökohtaisella näkemykselläni myös vaikutusta käytettävien käsitteiden valintaan. Musiikin käsitteitä olen etsinyt musiikkitieteen ja muusta musiikin alan kirjallisuudesta.

### 2.1 Peli

Voiko olla peliä ilman vapaaehtoisuutta? Entä onko työ enää työtä, jos sitä tehdään leikki-mielisesti? Näiden teemojen ympärillä ovat pyörineet henkilöt, jotka ovat pyrkineet antamaan määritelmiä leikille ja peleille. Englannin kielen sanat “play” ja “game” ovat niin monimuotoisia, että ne luovat määrittelylle oman kimuranttiutensa (Salen ja Zimmerman 2004, s.73, s.302). Toisaalta itse pidän ajatuksesta, että leikin ja pelaamisen määritelmään liittyy vapautta ja avaruutta, kuten leikkiin itseensä.

Schell (2008) käy työssään läpi eri henkilöiden muodostamia määritelmiä leikistä ja peleistä. Lopulta hän päätyy kymmeneen ominaisuuteen, jotka määrittelevät pelin.

- O1. Peleihin osallistutaan vapaaehtoisesti.
- O2. Peleihin sisältyy tavoitteet.
- O3. Peleissä on konfliktitilanne.
- O4. Peleillä on säännöt.
- O5. Pelit voidaan voittaa ja hävitä.
- O6. Pelit ovat interaktiivisia.

- O7. Peleissä on haastetta.
- O8. Pelit voivat luoda oman sisäisen arvonsa.
- O9. Pelit sitouttavat (engage) pelaajia.
- O10. Pelit ovat suljettuja, formaaleja järjestelmiä.

Tämän jälkeen Schell pohtii näkökulman muutosta määrittelemisen suhteen (Schell 2008, s.34). Edelliset kymmenen ominaisuutta saatiin katsomalla peliä pelin sisältä ulospäin. Hän päättää vaihtaa näkökulmaa ja katsoa peliä ulkoapäin pelaajan näkökulmasta. Seurauksena tälle Schell antaa määritelmän, joka hänen mielestään täyttää kaikki kymmenen ominaisuutta. Oma käänökseni määritelmästä on seuraavanlainen.

**Määritelmä 1.** Peli on ongelmanratkaisuvälineitä leikkimielisellä asenteella.

Kun puhun työssäni peleistä, tarkoitan Schellin määritelmän mukaista peliä. Käytän Schellin määritelmää myös, kun määrittelen tutkimukseni konstruktion eli pelin vaadittuja ominaisuuksia. Antaessaan määritelmän Schell toteaa, että määritelmät itsessään eivät ole työn varsinainen hedelmä (Schell 2008, s.38). Hänen mielestään määritelmistä on hyötyä vain silloin, jos ne antavat näkemystä ja oivalluksia, jotka auttavat pelisuunnittelussa. Haluan itsekin korostaa tätä mielipidettä valitsemalla Schellin määritelmän työssäni käytettäväksi määritelmäksi.

Kun Salen ja Zimmerman (2004) pyrkivät antamaan peleille määritelmän, he tarkastelevat kahdeksaa erilaista pelien ja pelaamisen määritelmää. Määritelmien tekijät tulevat hyvin erilaisista taustoista. Mukana ovat pelihistorioitsija, antropologi, pedagogi, sosiologi, filosofi, kaksi pelisuunnittelijaa ja pelitutkija. Mielestäni määritelmissä näkyy myös määrittelijöiden henkilökohtaista näkökulmaa. Jotkin määritelmät sisältävät piirteitä, joita ei näy yhdessäkään muussa määritelmässä (Salen ja Zimmerman 2004, s.79). Salen ja Zimmerman (2004) toteavat, että jotkin näistä piirteistä eivät päde kaikkiin peleihin. Jotkin piirteet selittävät enemmänkin pelien vaikutuksia eivätkä välttämättä pelejä itsessään. Lopulta näiden kahdeksan määritelmän pohjalta Salen ja Zimmerman muodostavat seuraavan määritelmän. Tässä annettu määritelmä on oma käänökseni.

**Määritelmä 2.** Peli on järjestelmä, jossa pelaajat ottavat osaa sääntöjen määrittelemään keinotekoiseen konfliktiin, joka päättyy kvantifioitavaan (quantifiable) lopputulokseen.

Tämä määritelmä sisältää samoja ominaisuuksia kuin Schellin luettelemat kymmenen ominaisuutta. Määritelmässä esiintyvät järjestelmä (O10), osaaottavat pelaajat (O9), säännöt (O4), keinotekoinen konflikti (O8, O3) ja kvantifioitava lopputulos (O2, O5). Mielestäni Schellin sekä Salenin ja Zimmermanin määritelmissä ei selkeästi näy vapaaehtoinen osallistuminen peleihin (O1). Muutamassa Salenin ja Zimmermanin esittämässä määritelmässä tulee esiin vapaaehtoisuuden ominaisuus (Salen ja Zimmerman 2004, s.79). Kuitenkin Salen ja Zimmerman toteavat jättäneensä omasta määritelmästäan pois epäolennaiset asiat (Salen ja Zimmerman 2004, s.80).

## 2.2 Pelisuunnittelu

Schell (2008, s.10) esittää, että pelisuunnittelija välittää kokemuksesta, jonka pelin pelaaminen synnyttää. Peli on vain keino saada ihmiset kokemaan kokemuksia. Ihmiskokemuksen tunteminen ja ymmärtäminen on pelisuunnittelijalle tärkeää. Toisaalta Schell mainitsee, että kokemuksista puhuminen on vaikeaa, koska kaikki mitä koemme ovat kokemuksia. Kokeemukset ovat niin arkipäiväisiä, että niitä on vaikea määritellä. On siis paradoksaalista, että kokeemukset ovat niin vaikeasti määriteltävissä, mutta toisaalta ne ovat kaikki mitä tiedämme. Schell toteaa pelisuunnittelijan tärkeimmäksi ominaisuudeksi kuuntelemisen (Schell 2008, s.4). Pelisuunnittelijan tulee osata kuunnella tiimiä, kohdeyleisöä, peliä, asiakasta ja itseään (Schell 2008, s.5). Kirjassaan Schell ei anna tarkkaa määritelmää pelisuunnittelulle.

Salen ja Zimmerman (2004, s.32) puhuvat mielekkään pelaamisen käsitteestä (meaningful play). Mielekkästä pelaamisesta syntyy, kun pelissä tehtävien toimintojen seuraukset ovat **selviä** (discernable) ja **integroitua** laajempaan kontekstiin pelissä. Selvillä seurauksilla tarkoitetaan selkeän palautteen antamista pelaajalle hänen toiminnoistaan (Salen ja Zimmerman 2004, s.35). Vaikka järjestelmä rekisteröisi pelaajan toiminnot, mutta tätä ei kommunikoida pelaajalle, pelaaja ei välttämättä ymmärrä tekemiensä toimintojen tarkoitusta. Integroidut toiminnot tarkoittavat, että toiminnoilla ei ole ainoastaan välittömiä seurauksia pelissä, vaan että toiminnot vaikuttavat myös myöhempään pelikokemukseen. Toiminnoista tulee turhia, jos niillä on vain välittömiä seurauksia ja jos ne eivät vaikuta myöhemmin pelin kulkuun. Kuitenkin Salen ja Zimmerman (2004, s.35) mainitsevat, että mielekäs pelaaminen voi tulla esiin eri peleissä eri tavoin. Jotkin mobiilipelit voivat esimerkiksi antaa hyvin visuaalista

ja suoraa palautetta pienistäkin pelaajan toiminnoista. Puolestaan shakin alkusiirtojen merkittävämmät seuraukset voivat näkyä vasta paljon myöhemmin pelin edetessä. Hyvän pelisuunnittelun päämääränä on luoda mielekästä pelaamista (Salen ja Zimmerman 2004, s.34). Salen ja Zimmerman (2004, s.80) esittävät pelisuunnittelulle seuraavan määritelmän.

**Määritelmä 3.** Pelisuunnittelu on prosessi, jonka kautta pelisuunnittelija luo pelaajalle kohdattavaksi pelin, josta syntyy mielekästä pelaamista.

Kun Schellin sekä Salenin ja Zimmermanin pelisuunnittelun päämääriä vertailee, huomaa heidän lähestyvän asiaa selvästi eri näkökulmista. Schell puhuu tunteiden ja kokemusten herättämisestä pelaajassa. Schell keskittyy suurempaan kokonaisuuteen ja lähtökohta on käytännöllinen. Voi myös ajatella, että Schellin näkemyksessä on kyse enemmän pelaajan näkökulmasta peliin. Salenin ja Zimmermanin määritelmässä puretaan peli pienempiin osiin ja näkökulma on astetta tieteellisempi. Tosin Salen ja Zimmerman (2004, s.36) mainitsevat mielekästä pelaamista tapahtuvan eri tasoilla, niin yhden shakin siirron merkityksistä aina kulttuurillisiin vaikutuksiin keskusteluissa ihmismielen ja tekoälyn voimien suhteista. Voidaan sanoa, että Salen ja Zimmerman toimivat työssään enemmän tutkimuksellisella tasolla.

### **2.3 Musiikki, improvisaatio ja säveltäminen**

Musiikki on taiteen muoto, jonka ilmaisumuotona on ajan mukaan organisoitu ääni (Ivanova 2010). Musiikki koostuu sävelkorkeudesta, rytmistä, dynamiikasta eli äänenvoimakkuuden vaihteluista, äänenväristä ja teoksen rakenteesta. Musiikin käsitteellä on eroja eri kulttuurien välillä.

Improvisointi ja säveltäminen kulkevat musiikillisina käsitteinä käsi kädessä. Improvisoinnin voidaan käsittää olevan säveltämistä ja säveltämisen improvisointia (Partti 2016). Tosin improvisointi voidaan käsittää paljon laajemmin ja kattavan erilaisia yksilön improvisoituja tekoja tai kollektiivin luovia ilmiöitä. Toisaalta improvisointi voidaan nähdä jokapäiväisenä ihmisen toimintana. Tässä mielessä jokainen ihminen on improvisoija, vaikkei sellaiseksi itseään ajattelisikaan (Santi 2010). Tämä ajatus on mielestäni siinä mielessä tärkeä, että improvisointi nähdään asiana, jota kuka tahansa voi tehdä.

Säveltämisen tai artistin luovan työn on voitu nähdä vaativan jonkinlaisia erityiskykyjä tai neroutta (Partti 2016, s.10). Partti (2016) mainitsee, että suomen kielen sana säveltäminen voi kuulostaa turhankin juhlalliselta. Hän mainitsee, että monissa muissa kielissä vastaava sana on paljon arkisempi ja viittaa sanoihin kuten järjestäminen, sommittelu ja muokkaaminen.

Marjamäki vertaili pro gradu -tutkielmassaan yhdentoista eri oppaan näkemyksiä sävellysprosessista. Oppaat edustivat yhdysvaltalaisista, kanadalaisista, englantilaisista, saksalaisista ja suomalaisista musiikkikasvatusta. Oppaista löytyi yhteneväisyyksiä, joiden pohjalta Marjamäki teki mallin sävellysprosessin vaiheista. Luukkanen (2016) luettelee pro gradu -tutkielmassaan mallin vaiheet, mutta toteaa, ettei vaiheiden järjestystä voida pitää ehdottomana.

1. valmistelu, inspiraatio,
2. itämisvaihe, kehittäminen, laajentaminen, uusi päämäärä.
3. oivallus, lopullinen päämäärä,
4. lopullinen versio, testaus, vahvistus.

### 3 Musiikkipelit: rytmii- ja instrumenttipelit

Käsittelen tässä kappaleessa erilaisia musiikillisia pelejä. Musiikkipelit voidaan jakaa rytmii- ja instrumenttipeleihin (Pichlmair ja Kayali 2007). Nämä kaksi kategoriaa voivat olla limitteisiä, sillä joissain peleissä voi olla ominaisuuksia molemmista kategorioista. Rytmipelejä voidaan pitää selvästi tunnetumpina verrattuna instrumenttipeleihin. Kaupallisesti julkaistuja instrumenttipelejä on huomattavasti vähemmän kuin rytmipelejä. Tästä huolimatta instrumenttipeleistä on olemassa tutkimusta ja kirjallisuutta. Käsittelen myös muita pelejä ja ohjelmistoja, joissa musiikki esiintyy ainakin hetkittäin merkittävässä roolissa pelaajan interaktion kannalta. Olen listannut digitaalisia luovia musiikkipelejä <sup>1</sup>. Lista painottuu erityisesti kaupallisesti julkaistuihin instrumenttipeleihin. Kaupallisten julkaisujen lisäksi on olemassa paljon erilaisia peliprototyyppisiä ja taideinstallaatioita, jotka antavat mahdollisuuden musiikilliseen ilmaisuun, kokeiluun ja leikkimiseen. Tässä luvussa esitän asioita omaan asiantuntijuuteeni perustuen.

#### 3.1 Rytmipelit

Rytmipeleissä pelaajan tulee toimia pelin asettaman rytmin mukaan (Pichlmair ja Kayali 2007, s.424). Pelaajan suoriutumista voidaan arvioida sen mukaan, kuinka hyvin pelaaja on pysynyt rytmisissä. Tämä mittari antaa mahdollisuuden kilpailulliseen pelaamiseen ja tulosten vertailuun. Rytmipeleissä ääniraita on proaktiivinen, sillä peli ja musiikki antavat pelaajalle ennakkoon tietoa ja vihjeitä siitä, kuinka hänen tulee toimia (Liebe 2013).

Kaupallisesta näkökulmasta katsottuna rytmipelien myyntimäärät ja julkaistujen rytmipelien määrä ylittää selvästi instrumenttipelien vastaavat määrät. Esimerkiksi rytmipeleistä Just Dance -sarjan pelit ylittävät kaikkien aikojen 100 myydyimpien pelien joukkoon (Limited). Tällöin puhutaan miljoonien kappaleiden myyntimääristä. Monessa tieteellisessä artikkelissa käsitelty ja yleisesti tunnetuimpiin instrumenttipeleihin kuuluva Electroplankton on VGChartz-sivuston mukaan myynyt vain 30,000 kappaletta.

---

1. Lista luovista musiikkipeleistä <http://www.mintshire.com/2017/04/20/list-of-music-creation-games/>

## 3.2 Instrumenttipelit

Instrumenttipelit antavat pelaajalle vallan musiikin rytmiin ja melodiaan. Peliä pelataan tai soitetaan kuten instrumenttia (Pichlmair ja Kayali 2007, s.425). Collins (2008, s.112) puhuu työssään luovista musiikkipeleistä (creative music games) tai sävellyspeleistä (composition games). Instrumenttipeli siis antaa pelaajalle mahdollisuuden musiikilliseen ilmaisuun. Instrumenttipelien musiikillisen ilmaisun laajuus ei välttämättä yllä perinteisten instrumenttien tasolle, mutta nämä pelit tuovat omaperäisen lähestymistavan musiikin luomiseen. Instrumenttipelien ääniraita on reaktiivinen, sillä ääniraita muodostuu pelaajan käskyjen mukaan (Liebe 2013). Usein instrumenttipelejä leimaa leikkilinen ote ja niistä saattaa puuttua kokonaan pelin määritelmään kuuluvat tavoitteet. Joskus instrumenttipelejä kutsutaankin digitaalisiksi leluiksi.

Tavoitteiden lisäämiseen ja instrumenttipelien muokkaamiseen leluista peleiksi liittyy erilaisia ongelmia. Nämä ongelmat voivat osaltaan heijastella niitä syitä, miksi instrumenttipelit ovat jääneet melko tuntemattomiksi suurelle yleisölle. Instrumenttipelit nähdään ja toisaalta tarkoituksellakin näyttäytyvät taidepeleinä, jotka vetoavat tiettyyn ja usein kapeaan yleisöön. Instrumenttipeleissä progressio saattaa jäädä vaillinaiseksi. Nykyisin niin pienet mobiilipelit kuin laajat AAA-pelit<sup>2</sup> sisältävät progressiota. Progression puute vaikuttaa asiantuntijuuteni mukaan myös pelin retentioon<sup>3</sup> toisin sanoen pelaajalta puuttuu selkeät etenemis- tai kehittymismahdollisuudet, joiden vuoksi palata takaisin pelin pariin. Perinteisten pelien etenemisen tuntu (Schell 2008, s.214), tavoitteiden hierarkisuus ja pitkäaikaiset tavoitteet (Schell 2008, s.217) usein puuttuvat instrumenttipeleistä. Instrumenttipelejä ei ehkä suunnitella pitkäkkestoiseen pelaamiseen vaan suunnittelun lähtökohtana on lelu (Schell 2008, s.90). Kuitenkaan en näe mahdollisena suunnitella instrumenttipelistä pelin määritelmän täyttävää peliä jopa lyhyt- ja pitkäaikaisine tavoitteineen. Progression puute saattaa johtua myös siitä, että instrumenttipelit ovat usein yksittäisen kehittäjän tai pienen tiimin teoksia, jolloin jo musikaalisen järjestelmän tai komponentin kehittämiseen kuluu suhteellisen paljon resursseja. Musikaalisen järjestelmän lisäksi tulee kehittää perinteisten pelien mekaniikat ja tavoitteet, joten työmäärä saattaa kasvaa tavanomaista peliä suuremmaksi. Progression lisääminen in-

---

2. AAA-pelit tarkoittavat tyypillisesti suuren budjetin pelejä, jotka on julkaissut kohtuullisen suuri julkaisija

3. Retentio tarkoittaa ominaisuutta, kuinka hyvin peli kykenee saamaan pelaajat palaamaan pelin pariin pelisession jälkeen. Retentiota pidetään tärkeänä ominaisuutena erityisesti mobiilipeleissä.

strumenttipeliin on myös ongelma suunnittelulle, sillä voi olla vaikea suunnitella tai keksiä instrumenttipelille luontevia pitkän ajan tavoitteita, jotka eivät tuntuisi keinotekoisilta. Tämä saattaa liittyä siihen riskiin, joka sisältyy valintaan suunnitella peli perustuen leluun (Schell 2008, s.91).

Instrumenttipelit toimivat usein pieninä annoksina eikä niiden suunnittelu välttämättä taivu pidempiin pelisessioihin. Toisaalta instrumenttipelit voivat tarjota elämyksiä kohdeyleisölleen vuosienkin jälkeen. Kohdeyleisö saattaa lähestyä peliä juuri musiikin luomisen näkökulmasta. Kohdeyleisön pelaajat eivät ehkä odotakaan instrumenttipelin sisältävän progressiota. Tärkeintä ehkä onkin tarjota mekaniikat, jotka tukevat rikasta musiikinluontia. Tällöin toistuvilla pelikerroilla luotu musiikki voi poiketa aiemmin luodusta musiikista ilahduttavalla tavalla, vaikka ensimmäisestä pelikerrasta olisi jo vuosia.

Instrumenttipelejä ovat muun muassa Electroplankton (Nintendo DS), Songbirds (iOS) , Sentris (PC) (kuvio 1) ja Otocky (NES).



Kuvio 1. Sentris ja ympyränmuotoinen musiikkikone. Lähde: <https://timbreinteractive.squarespace.com/sentris/>

### 3.3 Musiikinluontipelit

Käytän tässä työssä tutkimuksen kohteena olevista peleistä termiä musiikinluontipelit (ML-pelit). Musiikinluontipelin tulee tukea musiikin luomista ja olla aito määritelmän mukainen peli. Musiikinluontipelit kuuluvat instrumenttipeleihin, mutta ML-pelin määritelmä on tarkempi ja poissulkevampi. Terminä musiikinluontipelillä haluan korostaa, että pelillä voi sekä



soittaa sekä säveltää musiikkia pelaamisen aikana. Uudella termillä haluan myös erottaa ML-pelit muista musiikkipeleistä. Terminä sävellyspeli saattaisi kuulostaa helpommalta arkikielessä, mutta voi mielestäni johtaa mielikuvaan säveltämisen pelillistamisestä tai pelillisestä suunnittelusta, joka poikkeaa tämän työn päänäkökulmasta ja työn pelinkehitysprosessin tavoitteesta (Deterding ym. 2011).

**Määritelmä 4.** Ohjelmisto on musiikinluontipeli, jos seuraavat kohdat täyttyvät.

1. Ohjelmisto täyttää pelin määritelmän (Schell 2008, s.37).
2. Ohjelmisto tukee aktiivista nuottia (Pichlmair ja Kayali 2007, s.426).
3. Pelaamisen aikana kohdat 1 ja 2 täyttyvät samanaikaisesti.

Määritelmän kolmannella kohdalla pyrin selventämään sitä, ettei ohjelmisto olisi vain yhdessä näkymässä peli ja toisessa näkymässä sävellysovellus. Musiikinluontipelin mekaniikojen tulee siis täyttää pelin määritelmä ja samanaikaisesti tarjota mahdollisuus musiikin säveltämiseen. Pelin tavoitteiden saavuttamisen ja säveltämisen on hyvä tukea toinen toistaan.

Peliä *Moondust* (kuvio 2), jonka on suunnitellut Jaron Lanier vuonna 1983, pidetään maailman ensimmäisenä taidepelinä (Silfer; Pease). Sitä voidaan myös pitää ensimmäisenä musiikinluontipelinä, sillä peli täyttää musiikinluontipelin määritelmän. Onkin mielenkiintoista huomata, että jo ensimmäinen taidepeli sisälsi myös tavoitteita eikä ollut vain "leikkiä" (free-form play). Toisaalta voidaan ajatella, että on ollut vain luontevaa lisätä pelimäinen pistelas-kuri Commodore 64:lle julkaistuun ohjelmistoon. Pelaajalle *Moondust*issa ei juuri ole pitkän aikavälin tavoitteita, mikä on leimaavaa monelle taidepelinä pidetylle luomukselle. Ehkäpä *Moondust*in pelimäisiä tavoitteita ja musiikin luontia yhdistelevä ja kokeileva kokonaisuus on ollut ajan hengen mukainen tuotos. Onkin niin, ettei pelin ja musiikin luonnin yhdistelmästä ole saatu tehtyä tähän päivään mennessä tuotetta, jota voitaisiin kuvailla valtavirtatuotteeksi.



Kuvio 2. Moondust. Lähde: <http://www.lemon64.com/?mainurl=http%3A//www.lemon64.com/games/details.php%3FID%3D1709>

### 3.4 Muita musiikillisia ja äänen ilmentymiä

On myös olemassa pelejä, jotka voivat sisältää syvällisempiä musiikillisia ominaisuuksia kuin musiikkipelit itse. Tarkoitan tällä sitä, että jotkin pelit voivat sisältää esimerkiksi musiikin tekemiseen tarkoitettua sekvensseriä, jollaista harvoin sisällytetään vaikkapa rytmipeleihin. Peliä, joka sisältää sekvensserin, ei välttämättä kuitenkaan luokitella musiikkipeleiksi, jos pelin päämekaniikat eivät perustu musikaaliseen interaktioon. Tällöin sekvensseri on kuin erillinen musiikillinen työkalu eikä ole keskeisessä asemassa pelaajan etenemisen kannalta.

Olen jaotellut alalukuihin erilaisia musiikin, musiikin luomisen ja äänen ilmentymiä peleissä ja pelialustoilla. Annan alaluvuissa muutamia sopivia pelejä ja ohjelmistoja esimerkkeinä. Valittujen esimerkkien ei ole tarkoitus antaa kaikenkattavaa kuvaa erilaisista otsikkoihin sopivista ohjelmistoista. Alaotsikotkaan eivät anna koko kuvaa pelimusiikin kentästä. Musiikin ilmiöitä peleissä on mahdollista tutkia huomattavasti hienojakoisemmin. Collins (2008) kirjoittaa kirjassaan yleisesti pelimusiikin ja äänitekniikan kehittymisestä ajan myötä, musiikintekoprosessien kehittymisestä pelialalla ja pelimusiikin erilaisista ilmiöistä. Tähän lukuun pyrin valitsemaan esimerkkejä, jotka kuvaavat musiikin ja musiikin luomisen erityistä roolia peleissä ja muissa ohjelmistoissa.

### 3.4.1 Sekvensserit pelien sisällä ja hiekkalaatikkopelit

Jotkin pelit antavat mahdollisuuden musiikin luomiseen erilaisten työkalujen avulla. Tällaisten pelien varsinainen sisältö ei välttämättä ole musiikillisissa mekaniikoissa tai pulmissa, mutta nämä pelit mahdollistavat silti musiikin luomisen. Tarkoituksellisesti musiikin säveltämiseen tehdyt **sekvensserit** on eräs tapa antaa pelaajalle mahdollisuus tehdä musiikkia. Sekvensseri löytyy peleistä kuten Mario Paint, WarioWare D.I.Y., Mabinogi ja Animal Crossingin konsoliversiot (kuvio 3).

Super Mario Makerin kenttäeditorilla pelaaja voi luoda omia kenttiä. Kenttäeditorin musikaaliset ominaisuudet ovat yhdistelmä emergenssiä ja musiikin säveltämisen mahdollistavia elementtejä. Kun yhdistetään oikein alunperin tasoloikkapeliin tarkoitettuja elementtejä, saadaan rakennettua sekvensseri ja soitettua musiikkia. Tällaisia pelejä voidaan kutsua **hiekkalaatikkopeleiksi**. Hiekkalaatikkopelit avaavat mahdollisuuden yhteisön tekemälle sisällölle, jolloin usein enemmän tai myöhemmin esiin pulpahtavat myös musiikilliset luomukset (Schell 2008, s.364). Muita hiekkalaatikkopelejä ovat esim. Minecraft ja LittleBigPlanet.

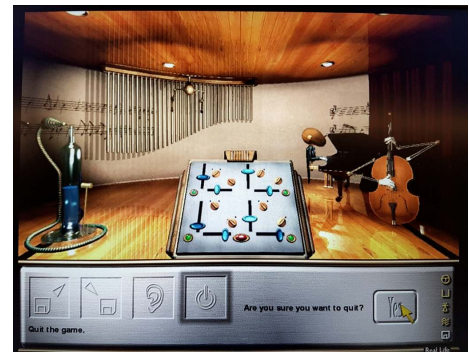
### 3.4.2 Instrumentti- ja sekvensseriohjelmistot pelialustoilla

Ensisijaisesti pelaamiseen tarkoitetuille alustoille on myös julkaistu pelkästään musiikin soittamiseen tai tekemiseen tarkoitettuja ohjelmia. Eräs tunnetuimmista lienee Music-sarja, jonka esikoinen ilmestyi ensimmäisen kerran alkuperäiselle PlayStationille vuonna 1998. Hieman tätä ennen ilmestyi PC:lle musiikkiohjelma eJay. Myöhemmin eJay:sta on julkaistu versio myös PlayStation 2:lle. Jopa Famicomille (NES) on Japanissa ilmestynyt ainakin Ikinari Musician -syntetisaattorihjelma.

Animal Crossingin konsoliversioissa voi pelaaja soittaa melodioita pelin ääniraidan päälle. Huomattavaa tässä ominaisuudessa on, että pelin ääniraidat on ohjelmoitu niin, että pelaaja soittaa aina ääniraidan sointuihin sopivia säveliä! Tämä on mielestäni yllättävä ominaisuus ja yksityiskohta, sillä musiikin soittaminen Animal Crossingissa ei ole pelaajan etenemisen kannalta olennaista.



Kuvio 3. Animal Crossingin sävellystila ja sekvensseri



Kuvio 4. Starship Titanicin musiikkihuone

### 3.4.3 Musiikilliset puzzlet peleissä

Musiikillisia pulmia näkee peleissä kuin peleissä aika ajoin. Tässä tarkoitan pelejä, joiden päämekaniikat eivät perustu musiikkiin. Ehkäpä pelit, jotka koostuvat minipeleistä tai yksittäisistä ratkaistavista pulmista, ovat otollisia musiikilliselle ongelmanratkaisulle. Esimerkiksi Sierran seikkailupeli *Torin's Passage* (1995) sisälsi pulman, jossa täytyi järjestää pienet otuskuorolaiset laulamaan haluttu melodia. Tasoloikkapeli *Super Mario 3D World* (2013) sisälsi kenttiä, jossa tasot vaihtoivat paikkaa pelin ääniraidan tahdin mukaan. Eräänä melko pitkälle vietyinä esimerkkinä voi mainita seikkailupeli *Starship Titanic* (1998) (kuvio 4). Pelin pulmassa pelaajan tulee järjestää robottibändin soittajat soittamaan sopivalta korkeudelta toistensa kanssa ja oikeaan tahtiin. Pelaajan on mahdollista äänittää ja toistaa robottien esitys missä vaiheessa tahansa. Robottibändin lopullinen tuotos on melkoista avantgarde jazzia, joten ainakin minun oli pelaajana vaikea tietää, olinko lopulta saanut ratkaistuksi pelin pulman. *Super Mario Odyssey* (2017) sisältää pulmia, jossa Toad-hahmolle tulee valita hänen teemakuvaukseensa sopivaa musiikkia.

### 3.4.4 Musiikin keräily peleissä

Pelien ääniraitaa ja -efektejä ollaan käytetty pääsiäismunana tai myös keräiltävinä saavutuksina. *Wario Land 4* (2001) (kuvio 5) sisälsi kerättäviä CD:itä pelin säveltäjältä Ryoji Yos-

hitomilta.<sup>4</sup> Nämä CD:t sisälsivät erikoisia äänimaisemia tai -tarinoita ja musiikkia, jota itse pelistä ei välttämättä löytynyt. Erikoiset CD:t toisaalta sopivat Wario-maailman ajoittain eriskummalliseen teemaan (Schell 2008, s.279). Peleissä, kuten Final Fight(1989)<sup>5</sup> ja Me-teos (2005), on mahdollista kuunnella pelien ääniä ja musiikkia pelin menujen kautta. Super Mario Odysseysssa (2017) on mahdollista kuunnella ja vaihtaa taustamusiikki mihin tahansa pelaajan aiemmin pelissä kohtaamaan musiikkikappaleeseen. Kappaleet esitellään listassa, josta on mahdollista nähdä, onko kappaleita vielä keräämättä.



Kuvio 5. Wario Land 4 ja musiikin keräily. Lähde: <https://www.mariowiki.com/File:Soundroom.PNG>

### 3.4.5 Audiopelit

Pelejä, joita voidaan pelata pelkästään äänien avulla eli ilman kuvaa, kutsutaan audiopeliksi (audio games). Tällaiset pelit ovat saaneet alkunsa näkörajoitteisille henkilöille suunnitelluista peleistä ja valtavirran musiikkipeleistä (Friberg ja Gärdenfors 2004). Kirjallisuushakua tehdessäni törmäsin tämän aiheen kirjallisuuteen, jota tuntui löytyvän enemmän ja helpommin kuin omasta aiheestani. Friberg ja Gärdenfors (2004) luettelevat työssään kolme audiopeliä, joista yksi on musiikin tekemiseen tarkoitettu X-tune. Käyttäjä pystyy navigoimaan ohjelmiston menuissa puhuttujen äänivihjeiden avulla. X-tunea luonnehditaan enemmän leluksi kuin peliksi, koska ohjelmistossa ei ole kilpailua tai tavoitteita. Ohjelmisto sisäl-

4. Wario Land 4 CD:t [https://www.youtube.com/watch?v=7sQSm5VWwOI&list=PLUoKNI6RW1SNC6RGAawlKdeOyz\\_1vaJgD](https://www.youtube.com/watch?v=7sQSm5VWwOI&list=PLUoKNI6RW1SNC6RGAawlKdeOyz_1vaJgD)

5. Final Fight -ääniefektejä <https://youtu.be/HDIW5TIW-cY>

tää erilaisia teemoitettuja ääniä musiikin tekemistä varten. X-tunella käyttäjät voivat tehdä, nauhoittaa ja jakaa musiikkia.

Teimme Paavo Kypön kanssa Global Game Jameissa (2017) Listen! nimisen audiopelin.<sup>6</sup> Pelissä pelaajan tulee navigoida sokkelossa pelkästään äänivihjeiden avulla. Listen! sisältää pienen seikkailun, tarinan ja satunnaisgeneroituja kenttiä. Peli on valtavirrasta poikkeava ja virkistävä kokemus kaikenlaisille pelaajille.

---

6. Listen! -audiopeli <https://globalgamejam.org/2017/games/listen>

## **4 Luovien musiikkipelien teorettinen pohja**

Esittelen tässä kappaleessa kolmea erilaista näkökulmaa yleisesti luoviin musiikkipeleihin. Näitä näkökulmia voidaan pitää myös musiikinluontipelien pohjana. Näiden työni pääartikkelien lisäksi kerron yleisemmin digitaalisten instrumenttien arviointiin ja suunnitteluun keskittyvästä kirjallisuudesta. Pichlmair ja Kayali (2007, s.428) puhuvat rytmi- ja instrumenttipeleistä ja niiden ominaisuuksista. Liebe (2013) jaottelee yleisesti pelejä ja näyttää, että tietyillä peleillä on muista poikkeavanlaiset ominaisuudet. Näiden pelien voidaan katsoa olevan Pichlmairin ja Kayalin määrittelemiä instrumenttipelejä. Omassa työssään Herber (2006) puhuu sävellysinstrumenteista ja korostaa yhtäaikaisen soittamisen ja säveltämisen ominaisuutta. Nämä kolme artikkelia ovat parhaat työni aiheeseen suoranaisesti liittyvät artikkelit, jotka olen löytänyt.

### **4.1 Pichlmairin ja Kayalin instrumenttipelit ja niiden ominaisuudet**

Pichlmair ja Kayali (2007, s.428) ovat esittäneet ominaisuuksia, jotka kuvaavat rytmi- ja instrumenttipelejä. Nämä termit ovat kvantisointi (quantisation), aktiivinen nuotti (Active Score Music), ääniagentit (sound agents), leikki (free-form play), rytmitoiminta (rhythm action), synestesia ja esiintyminen (play as performance). Näistä termeistä en keskity tässä työssä rytmitoimintaan, koska tämä termi liittyy usein vain rytmipeleihin. Jätin pois käsittelystä myös synestesian ja esiintymisen, koska on enemmän tai vähemmän tulkinnanvaraista, sisältääkö jokin peli näitä ominaisuuksia (Liebe 2013, s.52). Pichlmair ja Kayali eivät ole työssään esittäneet tarkkaa tapaa sen arvioimiseen, löytyykö jokin ominaisuus pelistä vai ei. Näitä käsittelystä pois jätettyjä ominaisuuksia ei ole välttämätöntä löytyä musiikinluontipelistä eikä sen musikaalisesta komponentista, jotta musiikinluontipelin määritelmä täyttyisi.

#### **4.1.1 Kvantisointi**

Kvantisoinnissa (quantisation) nuotit asetetaan tiettyyn pohjalla olevaan rytmiin (Pichlmair ja Kayali 2007, s.427). Luovissa musiikkipeleissä kvantisointi auttaa pelaajaa tekemään helpommin hyvän kuuloista musiikkia, vaikka toisaalta musikaalista vapautta rajataan. Pelaaj-

jaa siis estetään soittamasta säveliä mielivaltaisessa rytmissä. Esimerkiksi Electroplanktonin Beatnes-osiossa pelaaja voi painella ruudun nappuloita minkälaisessa rytmissä tahansa, mutta tästä syntyvät nuotit kuitenkin soitetaan ja tallennetaan sopivassa rytmissä taustalla soivan musiikin suhteen. Tällaisessa tapauksessa kvantisointia tehdään siis jatkuvasti yksittäisille nuoteille, kun pelaaja antaa syötteitä. Joissain tapauksissa kvantisointi tehdään jälkikäteen tallennettujen nuottien joukolle.

Kvantisointi tunnetaan musiikin tuottamisessa etenkin MIDI:n yhteydessä. Tällöin soittaja voi soittaa MIDI-soittimella nuotteja, jotka jälkikäteen voidaan asettaa tarkempaan rytmiin. Musiikin nauhoittamisen yhteydessä kvantisointia käytetään enemmänkin inhimillisen epätarkkuuden korjaamiseen kuin suoranaisesti tavoiteltaisiin leikinomaista tai helpompaa improvisoidun musiikin tuottamista.

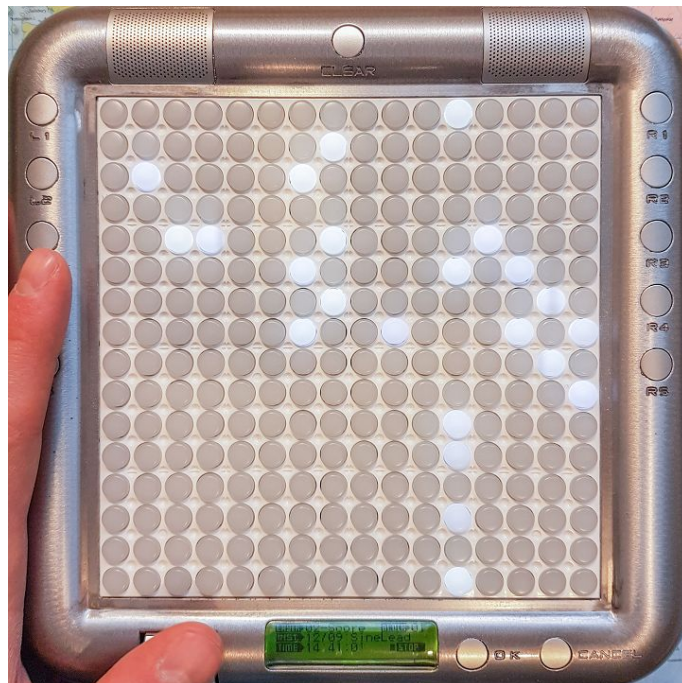
Kvantisoinnista johtuvaa musikaalisen vapauden rajoittamista voi myös miettiä kriittisesti pelisuunnittelun kannalta. Pelaaja voi kokea täsmällisen rytmien pidemmän päälle itseään toistavan kuuloiseksi. Teokset voivat kuulostaa myös liian kliinisiltä, kun inhimillisesti tuotetun musiikin svengiä ei voida kuulla. Toisaalta kvantisointi voi rajata myös musikaalista ilmauksellista vapautta. Voidaan nähdä, että Electroplanktonissa näitä ongelmia ollaan yritetty ratkaista niin, että pelaajalle on annettu mahdollisuus soittaa erilaista musiikkia eri osioissa. Toisissa osioissa nuotit putoavat täydellisesti rytmiin ja toisissa osioissa on pelaajalle annettu enemmän vapautta. Tasapainon hakemista kvantisoinnin ja vapaamman ilmaisun välillä voi olla hyvä pitää mielessä musiikinluontipeliä suunnitellessa.

#### **4.1.2 Aktiivinen nuotti**

Aktiivisen nuotin (Active Score Music) juuret ovat klassisen musiikin puolella (Pichlmair ja Kayali 2007, s.426) (Collins 2008, s.155). Aktiivisessa nuottimusiikissa nuottia eli partituuria voidaan muuttaa jokaiselle esityskerralle (Liebe 2013, s.51). Esitettävän kappaleen nuottia voidaan kutsua tällöin aktiiviseksi nuotiksi. Digitaalisissa peleissä tai installaatioissa aktiivinen nuotti tarkoittaa usein sitä, että pelaaja tai käyttäjä voi säveltää ja soittaa musiikkia samanaikaisesti. Toisaalta voidaan sanoa, että pelin ääniraita muodostuu dynaamisesti peliä pelatessa ja pelaajan toimintojen kautta.



Tällaisesta sävellyksen tavasta antaa hyvän esimerkin Toshio Iwain suunnittelema Tenori-On -laite.<sup>1</sup> Tenori-Onissa on 16x16 nappuloiden matriisi, jossa kukin nappula vastaa yhtä paikkaa nuotille (kuvio 6). Vasemmalta oikealle kulkeva aikapalkki soittaa käyttäjän asettamat nuotit tempon mukaisessa rytmissä. Käyttäjä voi jatkuvasti antaa aikapalkin soittaa säveliä ja samalla käyttäjä voi asettaa uusia nuotteja haluamiinsa kohtiin. Laite siis pitää muistissa käyttäjän määrittämää partituuria ja samalla soittaa sen mukaan. Tenori-On hyödyntää myös kvantisaatiota ja ennalta-asetettuja sävelasteikkoja, mitkä ovat asiantuntijuuteni mukaan perustavanlaatuisia ominaisuuksia, kun halutaan helpokäyttöinen työkalu miellyttävän musiikin tekemiseen.



Kuvio 6. Tenori-On

Aktiivinen nuotti voi myös toimia brainstorming-työkaluna säveltämisessä. Uusia ideoita voi syntyä kuunnellessa juuri sävellettyä musiikkia. Toisaalta nuotteja voidaan lisätä mielivaltaisesti satunnaisiin paikkoihin ja kuunnella lopputulosta lennosta. Tämä voi myös tehdä säveltämisestä leikinomaista.

1. Tenori-On demovideo <https://youtu.be/hzcpTMO0CrI>

### 4.1.3 Ääniagentit

Ääniagentit (sound agents) ovat visuaalisia elementtejä tai olioita, joiden ensisijainen tarkoitus on päästää ääntä tai vaikuttaa pelin ääniin (Pichlmair ja Kayali 2007, s.428). Ääniagentit palvelevat luovissa musiikkipeleissä nimenomaan musikaalista ulosantia, kun taas perinteisissä peleissä monilla hahmoilla on jokin pelaajan tavoitteeseen liittyvä tarkoitus. Ääniagentit eivät siis usein liity pelin pelaajalle asettamiin tavoitteisiin.

Esimerkkinä ääniagenteista Pichlmair ja Kayali (2007) mainitsevat pelin Rez, jossa pelin viholliset lakkaavat olemasta vihollisia ja muuttuvat ääniagenteiksi "travel mode-pelimuodossa. Electroplanktonin hahmot ovat lähes kauttaaltaan ääniagenteja, koska hahmoihin ei liity pelillisiä tavoitteita vaan ne ovat olemassa lähinnä niiden kautta tapahtuvan äänellisen interaktion tähden.

### 4.1.4 Leikki

Monissa luovissa musiikkipeleissä on mahdollista pelata ilman tavoitteita. Tällöin pelaamisesta tulee leikkiä (free-form play) (Pichlmair ja Kayali 2007, s.428). Tämä tosin voi itsessään olla pelin määritelmän vastaista, koska tavoitteet kuuluvat usein pelin määrittelymään (Schell 2008, s.37). Jotkin pelit voivat sisältää erillisen leikillisen pelimuodon. Leikkiä voidaan pitää digitaalisten lelujen ominaisuutena eikä niinkään pelien (Liebe 2013, s.51). Pichlmair ja Kayali (2007, s.428) sanovat, että leikki saa musiikkipelin tuntumaan enemmän instrumentilta, koska tällöin instrumenttimaiset piirteet korostuvat pelimäisten piirteiden sijaan.

Electroplanktonissa lähes kaikki pelimuodot ovat leikkiä. Fingerlabin Musyc iOS-sovellus on fysiikan mallinnukseen pohjautuva musiikkisovellus. Musyc sisältää leikillistä interaktiota, kuten pelit usein tekevät, mutta Musyc ei sisällä tavoitteita. Näin Musyc voidaan luokitella leikiksi.

## **4.2 Lieben interaktiivisuus ja musiikki peleissä**

Liebe (2013) rakentaa peleille luokittelun tarkastelemalla sitä, kuinka pelaaja on interaktiossa pelin ja musiikin kanssa. Luokittelussa kolme peliä Electroplankton, Mr.Bounce ja Radio Flare Redux poikkeavat muista luokitelluista peleistä. Kerron seuraavaksi Lieben luokittelun kolme yläkohtaa.

### **4.2.1 Pelaajan tai tietokoneen toiminta**

Ensimmäiseksi Liebe (2013, s.53) tekee peleille yleisen jaottelun sen mukaan, onko pelikokemuksen keskeisenä tekijänä pelaajan toiminta vai tietokoneen toiminta. Tietokoneen toimintaan pohjautuvissa peleissä tietokoneen suoritus ja algoritmit luovat pelin äänimaailman. Pelaaja antaa pelille syötteitä, mutta on enemmän pelin sisäisestä toiminnasta kiinni, millaiseksi pelikokemus ja äänimaailma muodostuu. Useimmat Lieben listaamista peleistä perustuvat tietokoneen toimintaan. Usein myös musiikinluontipelit kuuluvat tähän kategoriaan, koska äänimaailma muodostuu pelin sisäisen toiminnan pohjalta eivätkä musiikinluontipelit välttämättä vaadi pelaajalta esiintymistä.

Toisenlaisissa peleissä pelaajan toiminta on enemmän fyysistä kuin virtuaalista (Liebe 2013, s.54). Tanssimattopeleissä pelaajan täytyy tehdä työtä fyysisen suoriutumisen eteen sen sijaan, että pelaaja esimerkiksi painelisi vain perinteisen käsiohjaimen näppäimiä. Lieben näkökulmasta fyysistä suoriutumista vaativat myös karaokepelit, joissa pelaajan tulee oikeasti laulaa ja pelit kuten Guitar Hero ja Donkey Konga, joissa pelaajalle on annettu erilliset instrumentteja muistuttavat ohjaimet interaktiota varten. Liebe mainitsee oman käsityksensä esiintymisestä poikkeavan Pichlmairin ja Kayalin käsityksestä. Liebe katsoo esiintymiseksi vain sen, jos pelaaja tekee fyysistä suoritusta. Pichlmair ja Kayal sen sijaan hyväksyvät esiintymiseksi sekä fyysisen suorituksen että musiikillisen esiintymisen (Pichlmair ja Kayali 2007, s.429).

### **4.2.2 Käyttöliittymän ja pelitapahtumien suhde**

Liebe (2013, s.56) luokittelee pelien käyttöliittymät joko symbolisiksi tai objektiivisiksi eli edustaviksi (representational). Käyttöliittymä on objektiivinen, jos pelitapahtumat kuvaavat

pelaajan antamia syötteitä konkreettisella tavalla. Esimerkiksi Guitar Herossa pelaaja käsittelee kitaraohjainta, jolloin puolestaan itse pelissä näytetään artistia, joka soittaa kitaraa. Peleissä, joissa käytetään liikettä tunnistavia ohjaimia tai kameroita, on usein objektiivinen vastaavuus pelihahmojen liikkeinä pelitapahtumissa. Liebe luokittelee myös Electroplanktonin käyttöliittymän objektiiviseksi, koska hän on varmaankin ajatellut hahmojen kuvaavan pelaajalle, miten pelaaja voi olla niiden kanssa interaktiossa.

Symbolinen käyttöliittymä on ikään kuin objektiivisen vastakohta. Perinteisissä peleissä on usein käyttöliittymä, jolle annetut syötteet eivät vastaa pelitapahtumia konkreettisella tavalla. Pelaaja naputtelee näppäimistöä tai kosketusnäyttöä, mutta pelitapahtumat voivat olla pelihahmon hyppyjä, pelin esineiden käyttämistä tai menussa navigointia. Pelaajan antamalla käskyillä ei siis ole intuitiivista suhdetta pelin tapahtumiin (Liebe 2013, s.56). Tällöin käyttöliittymän ja pelitapahtumien välinen suhde on symbolinen.

#### **4.2.3 Ääniraidan tyyppi**

Kolmanneksi Liebe (2013) jaottelee pelejä sen mukaan, kuinka pelin ääniraita muodostuu. Ääniraita voi muodostua joko lineaarisesti, reaktiivisesti tai proaktiivisesti.

a. *Lineaarinen musiikki* viittaa perinteiseen pelin taustalla soivaan ääniraitaan. Tällöin ääniraita ei vaihdu tai muutu pelitilanteiden mukaan. Termi sisältää myös joissain peleissä olevat virtuaaliset radiokanavat, jotka kyllä vaihtuvat pelaajan käskyistä, mutta jotka jatkavat soimistaan pelitilanteista riippumatta.

b. *Reaktiivisella musiikilla* tarkoitetaan musiikkia, joka reagoi tai muuttuu pelaajan antamalla käskyillä. Tämä voi tarkoittaa myös musiikin vaihtumista pelitilanteen mukaan esim. taistelun alkaessa. Instrumenttipelien ääniraita muodostuu reaktiivisesti.

c. Soidessaan *proaktiivinen musiikki* kehottaa pelaajaa tekemään tiettyjä toimintoja. Musiikki ei siis reagoi pelaajan antamiin syötteisiin vaan vihjaa pelaajaa reagoimaan pelitilanteeseen. Proaktiivinen musiikki on ominaista rytmipeleille.

### **4.3 Herberin sävellysinstrumentti**

Instrumenttipelien luonne voidaan nähdä kahtalaisena. Instrumenttipeleissä yhdistyy samanaikaisesti sekä instrumentti että sävellys, kuten Herber (2006) mainitsee. Herber puhuu tällöin sävellysinstrumenteista (composition-instrument). Herber ei rajaa sävellysinstrumentteja ainoastaan peleihin vaan puhuu laajemin erilaisista töistä ja installaatioista, jotka sisältävät sävellysinstrumentin idean. Pääesimerkkinään Herber käyttää Electroplanktonia (Herber 2006, s.33). Tämä lisäksi hän mainitsee lisäesimerkkejä, kuten pelin Rez (2001) ja taideinstallaation Eden.

Herber esittää kolme ominaisuutta, jotka löytyvät kaikista hänen käsittelemistään sävellysinstrumenteista (Herber 2006, s.33). Nämä ominaisuudet ovat asynkronisuus, emergenssi ja generatiivisuus (generative-ness).

#### **4.3.1 Asynkronisuus**

Asynkronisuudella tarkoitetaan, että käyttäjän antaman syötteen ja järjestelmän vasteen välillä on jonkinlainen viive. Herber sanoo, että vaikka tällainen toteutus saattaa aluksi vaikuttaa epäresponsiiviselta, ominaisuus on tärkeä musiikin muodostumisen ja kuuntelukokemuksen kannalta. Välitön vaste saattaisi kiinnittää käyttäjän huomion järjestelmän ominaisuuksiin. Viive auttaa pitämään käyttäjien huomion kuuntelemisessa, luo tilaa asteittaiselle löytämisen prosessille ja musikaalisille yllätyksille.

Asynkronisuus näkyy Electroplanktonissa eri pelimuodoissa. Hanenbow-pelimuodossa pelaaja voi kääntää etukäteen kentän lehtien kulmaa, mutta vasta kun pelihahmot osuvat lehtiin, pelaaja kuulee lehden asennon kulmasta riippuvan sävelen (kuvio 7). Luminarrow-pelimuodossa pelaaja voi jälleen etukäteen kääntää nuolia, jotka määräävät pelihahmojen kulkusuunnan kentällä. Kuitenkin vasta kun pelihahmot tulevat nuolien kohdalle, pelaaja kuulee nuolen kohdalta kuuluvan sävelen, ja pelihahmo jatkaa matkaa nuolen osoittamaan suuntaan.

### **4.3.2 Generatiivisuus**

Generatiivisuus tarkoittaa erilaisia prosesseja, jotka vaikuttavat ääneen, grafiikkaan ja työn yleiseen kokemukseen (Herber 2006, s.33). Järjestelmän toiminta muuttuu käyttäjän interaktion kautta, mutta myös generatiivisten prosessien kautta. Herberin mukaan generatiivisuus näkyy eri tavoin eri töissä. Generatiivisuus voi syntyä langattomasta viestinnästä, satunnaisesta järjestämisestä ja valinnasta sekä algoritmeista. Generatiivisia prosesseja voi olla useita, mikä luo käyttäjälle tai kuuntelijalle uniikin kokemuksen.

Generatiivisuus näkyy Electroplanktonissa selkeiten Audience-pelimuodossa, jolloin peli itse pelaa itseään. Tällöin algoritmit generoivat musiikkia ja äänimaailmaa Electroplanktonin pelimuotojen sääntöjen puitteissa.

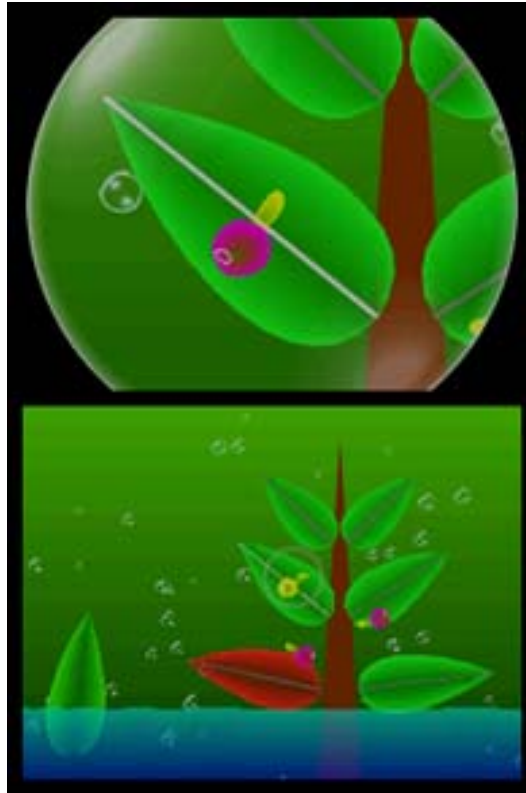
### **4.3.3 Emergenssi**

Emergenssiä tapahtuu, kun vähäistä sääntöjen joukkoa sovelletaan rajattuun joukkoon objekteja, ja seurauksena järjestelmässä tapahtuu jotain odottamatonta (Salen ja Zimmerman 2004, s.158). Emergenssiä voidaan pitää hyvän pelin tunnusmerkkinä (Schell 2008, s.141). Herberin mukaan emergenssi on tärkeä ominaisuus kokeilevassa ja improvisoidussa musiikissa (Herber 2006, s.33). Tämä pätee myös digitaalisiin taidetöihin. Emergenssi voi syntyä monista eri lähteistä. Emergenssiin voi johtaa esimerkiksi äänen ja kuvan kerroksittaisuus ja luovat algoritmit.

Emergenssiä tapahtuu ehkäpä selkeimmin Electroplanktonin fysiikan mallinnukseen perustuvassa Hanenbow-pelimuodossa (kuvio 7). Pelihahmot laukaistaan liikkeelle pelaajan haluamaan suuntaan. Kun pelihahmot osuvat kentässä oleviin lehtiin, lehdet soittavat säveliä. Näillä yksinkertaisilla säännöillä saadaan aikaan yllättäviä ja vaihtelevia melodioita ja säveliä.

## **4.4 Bott ym. musiikin luomisesta peleissä eleillä**

Bott, Crowley ja LaViola (2009) tutkivat eleiden kautta luotavaa musiikkia peleissä. He haluavat näyttää One Man Band -peliprototyypillään, että suosittuun rytmipeligenreen on mah-

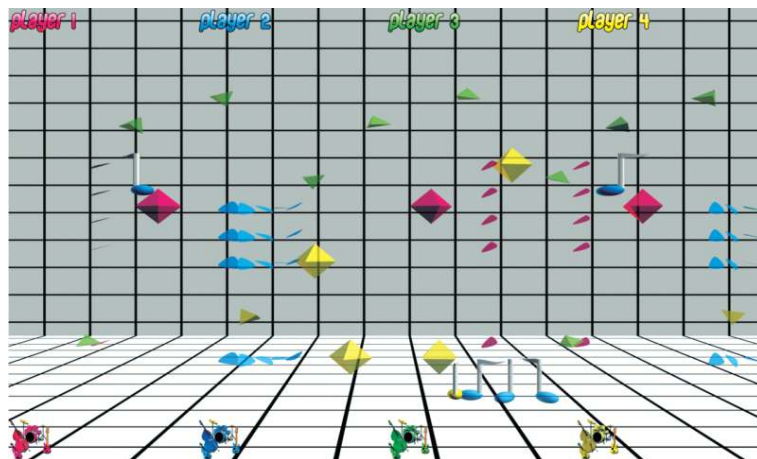


Kuvio 7. Electroplanktonin Hanenbow-pelimuoto. Lähde: <https://www.nintendo.com/games/detail/B4QgT8uckp1VMV3w1DQwPSWeFA11Bbr7>

dollista tuoda enemmän musiikillista ilmaisua ja musiikin luomista. Bott ym. vertailevat omaa luomaansa prototyyppiä Wii Music -peliin. Molemmissa peleissä ohjaamiseen käytetään Nintendo Wiin ohjaimia, joilla pelaajat voivat tehdä kiihtyvyyssanturien havainnoimia eleitä. One Man Bandissa soittimia soitettiin erilaisilla eleillä kuin Wii Musicissa. One Man Bandissa pelaaja voi valita eksplisiittisesti soittamansa sävelet, mitä Wii Musicissa ei pysty tekemään. One Man Bandissa on erilaisia pelimuotoja, mutta tämän työn kannalta kiinnostavin on Free play.

Free play -pelimuodossa pelaajat voivat soittaa improvisoiden ja halutessaan nauhoittaa musiikkiaan (kuvio 8) (Bott, Crowley ja LaViola 2009, s.20). Pelaajalla on käytössään eri soittimia ja useampia ääniraitoja eri soittimille. Pelimuodossa musiikkia soitetaan luupissa, joten päällekkäiset raidat soivat samanaikaisesti pelaajan soiton kanssa. Pelaajalla on mahdollista kvantisoida nauhoitetut sävelet. Tämä auttaa esimerkiksi tekemään täsmällisessä rytmissä pysyvän taustaraidan, jonka päälle pelaaja voi alkaa improvisoida. Bott, Crowley ja LaVio-

la (2009, s.21) esittävät, että rytmisesti miellyttävä soitto antaa pelaajille rohkaisua jatkaa omien musikaalisten ideoiden kehittämistä eteenpäin. Käyttöliittymä tarjoaa undo-toiminnon ja sävelien tyhjentämisen instrumenttikohteisesti. Luodut kappaleet voidaan tallentaa myöhempää toistoa tai muokkausta varten.



Kuvio 8. One Man Band ja free play -pelimuoto

Bott, Crowley ja LaViola (2009, s.25) sanovat havaintojensa perusteella, että yleisesti käyttäjät haluavat itse päättää soittamansa sävelet. He mainitsevat, että tämä havainto on ristiriidassa rytmipelien suosion kanssa, koska ne eivät anna pelaajalle mahdollisuutta valita soittamaansa musiikkia tai säveliä. Bott ym. sanovatkin, että rytmipelit antavat pelikokemuksen kautta *tunteen* siitä, että pelaajalla on valtaa soittamaansa musiikkiin. Jos sitten peli lupaa pelaajalle musiikinluontia, pelaajat haluavat kontrolloida sävelkorkeutta ja sävelien ajoitusta. Käsitin, että tässä on siis kyse myös pelaajien odotusten hallinnasta ja täyttämisestä. Bott ym. mielestä yksinkertainen mutta tehokas käyttöliittymä voi antaa pelaajalle mahdollisuuden luoda monimutkaisiakin sävellyksiä.

#### 4.5 Digitaalisten instrumenttien analysoimisesta

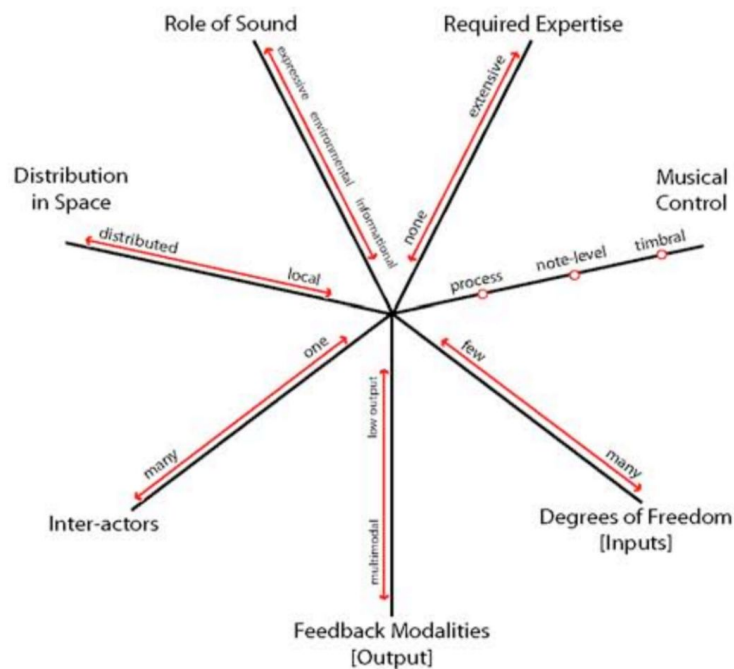
Aiemmissa mainitsemisani artikkeleissa on suoraan tutkittu luovia musiikkipelejä. Tällaiset artikkelit eivät ole oman kirjallisuuskatsaukseni mukaan kovin yleisiä. Sen sijaan, jos hakua laajentaa digitaalisiin instrumentteihin, tutkimuskirjallisuutta on helppo löytää runsaasti. Tällaisista artikkeleista voidaan löytää ominaisuuksia, joita digitaalisiin instrumentteihin



on sisällytetty ja joiden mukaan instrumentteja on luokiteltu. Näitä ominaisuuksia voidaan rakentaa myös musiikinluontipeleihin, mutta luulen että jotkin ominaisuudet saattavat olla pitkälle vietyjä ja vaatia pelaajalta musikaalista perehtyneisyyttä. Toisaalta voidaan ajatella kehityshaasteena sitä, kuinka edistyneitä musikaalisia ominaisuuksia saadaan rakennettua peliin ja samalla pitää peli eleganttina ja helposti lähestyttävänä. Joka tapauksessa on hyvä pitää mielessä musiikinluontipelin sopiva laajuus kehitysresurssien suhteen.

#### 4.5.1 Birnbaumin ym. kokemuseräinen malli

Birnbaum ym. (2005) esittelivät seitsemän dimension avaruuden, jolla voidaan arvioida digitaalisen instrumentin fenomenaalista eli kokemuseräistä luonnetta ja ominaisuuksia (kuvio 9). Malli katsoo siis instrumenttia käyttäjän näkökulmasta. Nimitän tätä mallia kokemuseräiseksi malliksi, kuten Magnusson (2010) tekee artikkelissaan. Akselien suomennokset ovat omia ja osittain tähän työhön sovitettuja:



Kuvio 9. Birnbaumin ym. kokemuseräinen malli

- Vaadittu asiantuntijuus. Kuinka paljon harjoittelua instrumentti vaatii käyttäjältään?
- Musiikillisen hallinnan taso. Käyttäjä voi hallita musiikkia äänensoinnin, yksittäisen

nuotin tai prosessin tasolla.

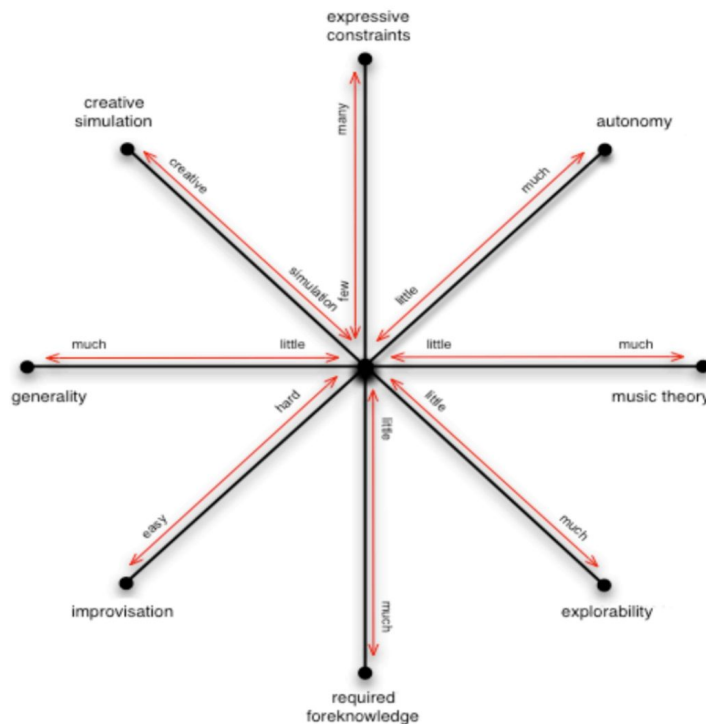
- Palautteen laatu. Tyypillisesti instrumentti voi antaa visuaalista, auditiivista, tuntoaistiin perustuvaa tai kinesteettistä palautetta.
- Syötemahdollisuudet. Kuinka monella tavalla käyttäjä voi antaa syötteitä instrumentille?
- Käyttäjien määrä. Kuinka monta henkilöä voi käyttää instrumenttia yhtäaikaisesti?
- Tilan distribuutio. Toimiiko instrumentti paikallisesti yhdessä paikassa tai monessa paikassa vaikkapa pilvipalvelun yli?
- Äänen rooli. Ääni on joko ilmauksellisessa, ääniympäristöllisessä tai tietoa välittäväsä roolissa.

Musiikinluontipeleissä nämä ominaisuudet liittyvät suorimmin musiikkikomponentin ominaisuuksiin. Omaan perehtyneisyyteeni perustuen mainitsen, että useimmissa pelialustoilla ilmestyneissä musiikintekoon tarkoitetuissa työkaluissa vaadittu asiantuntijuus on pyritty pitämään matalalla. Tähän pyrin myös omassa prototyypissäni. Musiikillisen hallinnan tasoa viedään harvoin äänensoinnin tasolle pelialustoilla esiintyneissä sekvenssereissä. Sen sijaan esimerkiksi Electroplanktonin Lumiloop tarjoaa mahdollisuuden myös äänensoinnin hallintaan. Äänen rooli vaihtelee luovissa musiikkipeleissä ääniympäristöllisen ja ilmauksellisen välimaastossa.

#### **4.5.2 Magnussonin tietoperäinen malli**

Magnusson (2010) mainitsee työssään perinteisten instrumentteja analysoivien mallien olevan enemmänkin kokemuseräisiä ja keskittyvän enemmän ihmisen ja laitteen interaktioon. Magnusson esittääkin rinnalle oman epistemisen eli tietoon perustuvan mallin (kuvio 10). Mallin näkökulma keskittyy instrumentin teoreettiseen luonteeseen, konseptiin ja instrumentin sisäisiin ominaisuuksiin, kuten musiikin teorian sisältymiseen instrumenttiin itseensä. Magnusson mainitsee oman mallinsa täydentävänä ja osittain limittäisenä Birnbaumin ym. mallille. Mallissa on kahdeksan akselia.

- Ilmaukselliset rajoitteet. Kuinka paljon instrumentti asettaa rajoitteita musikaaliseen ilmaisuun?



Kuvio 10. Magnussonin episteminen malli

- Autonomia. Kuinka paljon instrumentti tarjoaa automatisoitua toiminnallisuutta? Joi-tain toimintoja pystytään delegoimaan instrumentille.
- Musiikin teorian taso. Kuinka paljon instrumenttiin on sisällytetty kulttuurikohtaista musiikin teoriaa, kuten sävelasteikkoja, sointuja tai rytmisiä rakenteita.
- Tutkittavuus ja syvyys (explorability). Kuinka paljon instrumentista löytyy tutkittavaa? Tämä vaikuttaa instrumentin osallistavuuteen, oppimiskynnykseen ja flow-kokemuksen mahdollistumiseen.
- Vaatimukset ennakkotiedoista. Monien instrumenttien käyttö ei vaadi musiikillista tie-tämystä, koska se on sisällytetty itse instrumenttiin.
- Improvisointi. Kuinka paljon instrumentilla on mahdollista tehdä vapaata improvisoin-tia?
- Yleisluonteisuus. Soveltuuko instrumentin ilmaisu erilaisiin musiikillisiin tilanteisiin ja yhteyksiin?
- Luova-simulaatio -akseli. Onko instrumentti uusi interaktioltaan, ääneltään ja toimin-naltaan vai simuloidaanko vakiintuneita työkaluja ja toimintatapoja?

Magnussonin antamien esimerkkikuvien mukaan, sekvensserityökalut usein rajoittavat musiikallista ilmaisuja paljon Magnusson (2010, s.45). Toisin sanoen sekvenssereillä ei juuri pysty soittamaan improvisoitua musiikkia. Luovat musiikkipelit sisältävät joskus instrumenttimaisia käyttöliittymiä, jotka asettavat vähemmän rajoitteita ilmaisuun kuin sekvensserit. Asiantuntijuuteni mukaan luoviin musiikkipeleihin halutaan sisällyttää autonomiaa ja musiikin teoriaa, jotta peli soveltuisi paremmin myös musiikintekoa harrastamattomille. Samalla pelikokemuksesta tulee enemmänkin leikkisää ja vähemmän vakavaa ja työkalumainen.

### **4.5.3 Viitekehyksiä instrumentin suunnitteluun**

Overholt (2009) esittää artikkelissaan viitekehyksen musikaalisten instrumenttien suunnitteluun (Musical Interface Technology Design Space, MITDS). Overholtin artikkeli sisältää seitsemän kohtaa instrumenttien interaktion arvioimiseen ja iteratiiviseen kehittämiseen. Overholt mainitsee jokaisen MITDS-viitekehyksen avulla suunniteltavan instrumentin kolmeksi pääaiheeksi ihmiskäyttöliittymän, äänen syntetisoinnin ja tiedon mappauksen syöte- ja tulostejärjestelmien välillä. Seitsemän arviointikohdan lisäksi Overholt antaa suunnittelumalleja instrumentin kehittämistä varten.

Armitage, Morreale, McPherson ym. (2017) mainitsevat, että digitaalisten instrumenttien arviointiin ja vertailuun on esitetty monia sopivia viitekehyksiä. Armitagen ym. mukaan nämä viitekehykset eivät kuitenkaan anna vastauksia hienovaraisiin ja tärkeisiin yksityiskohtaisiin suunnittelukysymyksiin. Armitage ym. esittävätkin uuden termin "NIMEcraft". NIMEcraft tarkoittaa kahden identtisen instrumentin välillä olevia mikrotason eroja ja eroihin johtaneita suunnitteluprosesseja. Armitage ym. myös mainitsevat instrumentin testauksen erilaiseksi toiminnaksi kuin soittamisen. Instrumentin rakentajan ei tarvitse olla huippusoittaja voidakseen rakentaa ja testata rakentamiaan soittimia. Toisaalta rakentajat ovat kokeneempia juuri instrumenttien testaamisessa kuin soittajat.

O'modhrain (2011) esittää digitaalisten instrumenttien arvioinnin viitekehysessään, että eri sidosryhmät haluavat instrumentilta erilaisia ominaisuuksia. O'modhrain sanoo, että eri viitekehysten antamiin vastauksiin ei vaikuta ainoastaan kysytty kysymys vaan viitekehyksen metodologinen lähestymistapa ja arviointia tekevän sidosryhmän omat intressit. O'modhrain

tekee muutamia viittauksia pelitutkimukseen. Hänen mielestään “pitkien pelien” pelaaminen on samankaltaista kuin uuden instrumentin soittamisen opettelu tai uuden kappaleen opettelu tutulla soittimella. Arvioitaessa soittamisen kokemusta O’modhrain (2011, s.37) viittaa tutkimukseen, jossa testattiin kahdeksaa “pitkää peliä” (long games). Tutkimuksen mukaan pelisessioiden aikana havaittujen puutteiden korkea tiheys vaikuttaa pelikokemukseen selvästi. Sen sijaan jos jokin puute ilmenee harvoin pelisession aikana, vaikutus pelikokemukseen on vain väliaikaista. O’modhrain esittää, että digitaalisten instrumenttien sisältämien puutteiden ja soittokokemuksen välillä saattaa olla samanlainen yhteys.

O’modhrain (2011, s.38) luettelee digitaalisen instrumentin arvioitavina asioina ja suunnitellutavoitteina nautittavuuden, soitettavuuden, kestävyuden (robustness) ja suunnittelumäärittelyjen toteutuvuuden. Puhuessaan nautittavuuden arvioimisesta O’modhrain viittaa jälleen pelitutkimukseen. Pelaamisen nautittavuutta voidaan arvioida pelin flow-kokemuksella. Pelaajan flow-kokemukseen vaikuttaa kahdeksan asiaa, jotka ovat keskittyminen, haaste, taidot, pelin hallitseminen, selkeät tavoitteet, palaute, immersio ja sosiaalinen interaktio (Sweetser ja Wyeth 2005). O’modhrain sanoo, että monet näistä elementeistä on rinnastettavissa myös instrumentin soittamiseen. O’modhrain esittää siis, että samankaltaista arviointimallia olisi mahdollista käyttää myös instrumenttien kohdalla.

## **4.6 Muuta pelimusiikkikirjallisuutta**

Muita työni aihetta sivuavia artikkeleita olen löytänyt muutamia. Collins puhuu pelimusiikista laajasti mutta enemmän kulttuurillisesta näkökulmasta. Collins mainitsee lyhyesti myös luovat musiikkipelit (Collins 2008, s.112). Musiikkipelejä kuvaillessaan Collins sanoo, että on olemassa pelejä, joissa musiikki on keskeisessä roolissa, kuten luovat musiikkipelit ja rytmitoimintapelit. Tämä kahtiajako on hyvin samankaltainen kuin Pichlmairilla ja Kayalilla. Collins käyttää lisäksi termiä sävellyspelit (composition games) luovista musiikkipeleistä. Collins mainitsee uudelleen miksaamisen, tuottamisen ja uusien kappaleiden säveltämisen olevan tärkeässä asemassa luovissa musiikkipeleissä. Amplitude (2003) sekoittaa ominaisuuksia sävellyspeleistä ja rytmitoimintapeleistä.<sup>2</sup> Amplitudessa pelaajien on mahdollista miksatua uudelleen populaarimusiikin kappaleita. Varsinainen pelaaminen koostuu toiminto-

---

2. Amplitude Remix Mode <https://youtu.be/j2avIpaYkYY>

jen ajoittamisesta rytmin ja melodian mukaan kuten rytmipeleissä. Pelissä uudelleenmiksatuista kappaleista muodostuu uusia kenttiä. Uudelleenmiksattuja teoksia voidaan myös tallentaa.

Collinsin kokoomateoksessa Schultz (2008) puhuu, millä tavalla musiikkipelit opettavat musiikin teoriaa. Schultz ei tarkoita musiikin teoriolla tiukkaa akateemista teoriaa vaan musiikin rakenteiden ja ilmiöiden ymmärtämistä. Schultz käyttää esimerkkinä Dance Dance Revolutionia ja toteaa, kuinka peliä on käytetty rytmidiktaatin opettamiseen. Schultzin mukaan peleissä käytetään vain vähän länsimaista musiikin nuotinnusta. Hänen mukaansa lähimmäksi tätä ovat tulleet Mario Paint ja Animal Crossing (kuvio 3). Schultz mainitsee nuotinnuksen harvan käytön syyksi sen, että perinteisen nuotinnuksen monimutkaisuus voisi olla turhauttavaa aloittelevalle pelaajalle. Tämä voisi häiritä flow-kokemusta. Hän käsittelee sitten pelisuunnittelijoiden eri tapoja kommunikoida musikaalisia käsitteitä, kuten sävelkorkeus ja ajan kuluminen, pelaajalle intuitiivisesti.

## 5 Konstruktio: Musiikinluontipeli

Luotava ohjelmisto koostuu peliohjelmistosta ja musiikkikomponentista. Ohjelmiston korkean tason vaatimukset tulevat musiikinluontipelin määritelmästä 4. Tällöin peliohjelmiston tulee olla määritelmän 1 mukainen peli, ja musiikkikomponentin tulee tukea musiikin säveltämistä. Tarkempuna vaatimuksena musiikkikomponentille on, että se tukee aktiivista nuottia (luku 4.1.2). Muita vaatimuksia ominaisuuksille saadaan aiheeseen liittyvästä kirjallisuudesta.

Pichlmairin ja Kayalin sekä Herberin artikkeleista saadaan vaatimuksia konstruktion ominaisuuksille suunnitteluvaiheessa. Myös Bott ym. mainitsevat asioita, joita voidaan ottaa huomioon suunnittelussa. Lieben artikkelissa mainitut asiat sopivat mielestäni paremmin pelin arviointiin sitten, kun pelinkehitys on päättynyt. Arvioin peliä Lieben mainitsemilla kohdilla, kun konstruktio on tämän työn puitteissa valmis. Valmista konstruktiota voidaan myös arvioida, onko se Pichlmairin ja Kayalin nimeämän instrumenttipelin mukainen. Edelleen peliä voidaan arvioida, onko siinä Herberin mainitsemia sävellys-instrumentin piirteitä. Kehitysprosessin jälkeen voidaan myös katsoa, oliko Bott ym. mainitsemia suunnittelullisia asioita otettu huomioon. Peliä arvioidaan myös, täyttääkö se musiikinluontipelin määritelmän. Näistä saadaan mielestäni varsin kattava patteristo valmiin konstruktion teoreettiseen arviointiin. Tämä on suunnittelututkimuksen loppuvaiheen teoreettisen yhteyden näyttämistä.

Peli ei vaadi käyttäjältä ennakkotietoja musiikin säveltämisestä tai tuottamisesta. Peli on tarkoitettu kaikille käyttäjille, jotka ovat kiinnostuneet musiikin luomisesta. Pelin tulisi rohkaista pelaaja leikkimielisyyteen ja kokeiluun, vaikka peli sisältääkin tavoitteita. Leikillisen asenteen tulisi näkyä erityisesti musiikinluontiprosessissa ja luodussa musiikissa. Toisaalta peliohjelmiston tulee olla tarpeeksi joustava kokeilemiseen. Tarkoitin tällä sitä, että pelimekaniikkojen ei tulisi tuomita pelaajaa kokeilemisesta liian ankarasti. Jos halutaan, että peli sisältää tavoitteita ja samalla säilyttää mahdollisuuden kokeilulle, tulee hakea tasapainoa leikin ja tavoitteiden välillä. Pelin antaman arvostelun tai tuomion tulisi toimia niin, että se rohkaisee pelaajaa kehittymään pelaamisessaan (Schell 2008, s.128).

Vaikka ohjelmistolle asetetaan vaatimuksia, jäävät jotkin pelisuunnittelulliset osat väistämät-

tä aavistusten ja parhaiden arvausten varaan. Näin on etenkin laadullisten vaatimusten kohdalla, joiden vaikutukset voivat näkyä monessa pelin pienessä yksityiskohdassa. Pelisuunnittelu on monisäikeinen asia eikä sitä ole helppoa tehdä täysin formaalisti. Pelisuunnittelun tarkoituksena on rakentaa ja muodostaa pelaajalle kokemus, joka rakentuu hyvin erilaisista osatekijöistä (Schell 2008, s.21).

## **5.1 Vaatimusmäärittelyä**

Kerron tässä pelin suunnitteluvaiheen vaatimusmäärittelyä. Yleisesti sovelluksen tulee olla helppokäyttöinen, kohtuullisen vakaa ja pelissä tulee olla yhtenäinen graafinen tyyli. Helppokäyttöisyydellä tarkoitan muista mobiilisovelluksista tuttuja käyttöliittymäratkaisuja ja matalaa oppimiskynnystä. Kohtuullisella vakaudella tarkoitan sitä, että sovellus tuntuu tarpeeksi suorituskykyiseltä eikä siitä löydy liikaa sovelluksen käyttöä keskeyttäviä puutteita. Yhtenäisellä graafisella tyylillä tarkoitan värimaailmaa ja yleistä estetiikkaa.

Pelille vaatimuksia ovat helppokäyttöisyyden lisäksi hauskat pelimekaniikat, mielenkiintoinen progressio ja retentio. Hauskoilla pelimekaniikoilla tarkoitan, että pelaajan tulee kokea mekaniikat uudennlaisiksi tai yllättäviksi ja miellyttäväksi (Schell 2008, s.26). Mielenkiintoisella progressiolla tarkoitan, että pelin tulee sisältää riittävästi yhden pelisession keston ylittäviä etenemismahdollisuuksia. Retentio menee hieman käsi kädessä progression kanssa, sillä pelaajan tulisi mielenkiintoisten tavoitteiden tähden haluta palata pelin pariin pelisession jälkeen.

Musiikkikomponentin vaatimuksia ovat musiikillinen tarkkuus, mielenkiintoiset ja samalla rajatut musiikilliset mahdollisuudet ja leikillinen lähestymistapa. Musiikillisella tarkkuudella tarkoitan sitä, että komponentti pystyy soittamaan musiikkia rytmillisesti tarkasti viiveettä. Mielenkiintoisilla musiikillisilla mahdollisuuksilla tarkoitan, että instrumentin tulisi tarjota riittävästi luovaa tilaa käyttäjälle. Samalla mahdollisuuksia tulee rajata, jotta interaktion kautta on mahdollista syntyä mielenkiintoista musiikkia ilman erityistä musiikin teorian tuntemusta. Leikillisellä lähestymistavalla tarkoitan, että instrumentilla tulee voida leikkiä ja kokeilla. Leikkiminen voi johtaa uusiin ja odottamattomiin melodisiin tai rytmisiin löytöihin.



## 5.2 Suunnittelututkimuksesta

Järvinen (2004) kokoaa yhteen suunnittelututkimuksen metodikirjallisuutta. March ja Smith (1995) sanovat, että suunnittelututkimuksen konstruktion tulee vastata tosimaailman ongelmaan. Van Aken (2004, s.225) sanoo suunnittelututkimuksen tuottavan tietämystä suunnittelu- ja konstruointiongelmien ratkaisemiseen alan ammattilaisille eikä niinkään maallikoille. Van Akenin mielestä suunnittelututkimuksen keskiössä ei ole artefakti itsessään vaan tietämyksen kehittäminen ratkaisujen suunnitteluun (Aken 2004; Järvinen 2004).

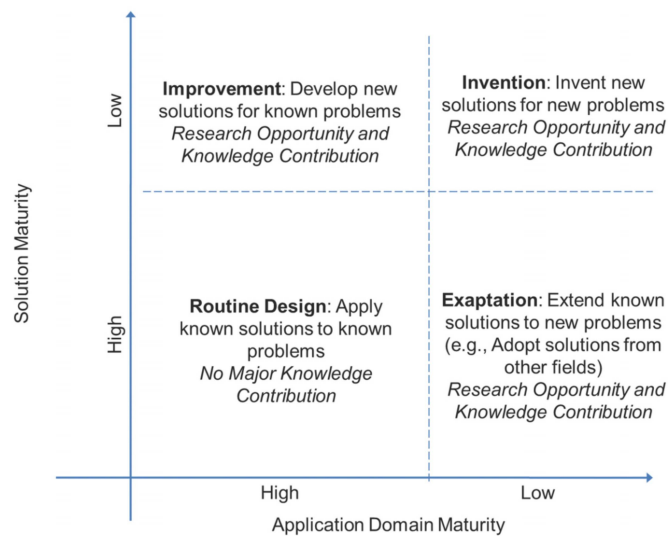
Rautiainen, Sippola ja Mättö (2017) esittävät suunnittelututkimukselle seitsemän vaihetta:

1. Etsitään käytännöllisesti relevantti ongelma, jossa on myös aihetta tutkimukselle;
2. Tarkastellaan potentiaalia pitkän aikavälin tutkimukselle;
3. Yleisen ja kokonaisvaltaisen ymmärryksen kokoaminen aihealueesta;
4. Teoriaan perustuvan ratkaisun innovointi ja konstruointi;
5. Ratkaisun implementointi ja testaaminen käytännön ympäristössä;
6. Ratkaisun sovellettavuuden laajuuden (scope) tarkastelu;
7. Teoreettisen yhteyden ja tieteellisen kontribuution näyttäminen.

Labro ja Tuomela (2003) sanovat, että vaiheilla 3, 4 ja 5 varmistetaan sisäinen validiteetti ja vaihe 6 esittää tarpeen myös ulkoisesta validoinnista. Useimmat vaiheista ovat päällekkäisiä aiempien ja seuraavien vaiheiden kanssa. Vaiheen 3 kokonaisvaltaisen ymmärryksen kokoaminen jatkuu koko tutkimusprosessin ajan. Teoreettisen ymmärryksen kehittäminen on todennäköisesti alkanut jo huomattavasti ennen varsinaista konstruktivistista tutkimusprosessia. Myös vaiheen 7 teoreettisen yhteyden näyttäminen tulisi jatkua koko tutkimusprosessin ajan. Kuitenkin varsinaista teoreettista kontribuutiota voidaan käsitellä yksityiskohtaisemmin vasta tutkimuksen loppuvaiheessa.

## 5.3 Suunnittelututkimuksesta tässä työssä

Gregor ja Hevner (2013) esittävät ohjeita mahdollisimman vaikuttavaan suunnittelututkimukseen. He jaottelevat nelikentässä erilaisia suunnitteluratkaisuja uutuusarvonsa mukaan (kuvio 11). Tämä työ sijoittuu *parantelun* (improvement) kenttään.



Kuvio 11. Suunnittelututkimuksen kontribuutiioviitekehys

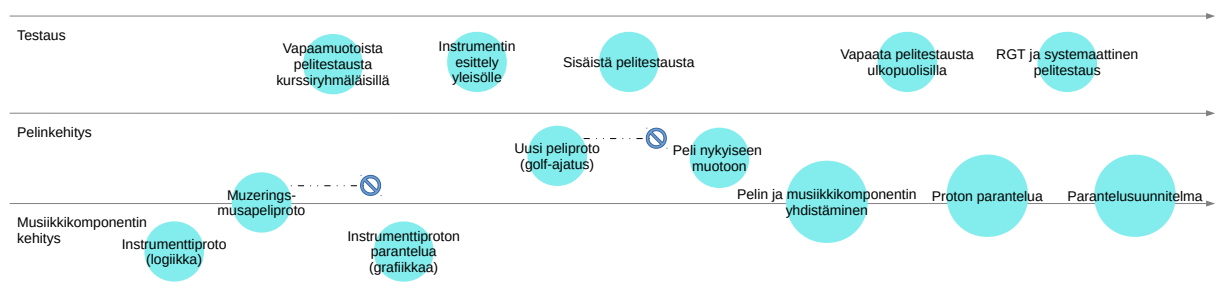
Avaan tässä prototyypin kehitysprosessia, joka on esitetty kuviossa 12. Kuviossa vaiheet on esitetty ajan suhteen vasemmalta oikealle. Yleisesti kehitys eteni vaatimusten ja testauksesta saadun palautteen ohjaamina. Kuitenkin ennen kuin pelin mekaniikat saavuttivat lopullisen muotonsa, oli kehitys hieman poukkoilevaa.

Toteutusvaiheessa aloitin musiikkikomponentin ohjelmoinnista, koska siihen liittyi enemmän riskejä kuin pelinkehitykseen. Riskit liittyivät lähinnä oman kokemukseni puutteeseen musiikkisekvenssereiden rakentamisesta. Sain valmistettua jonkinlaisen prototyypin instrumentista, joka sisälsi lähinnä sekvensserin logiikkaa ilman grafiikkaa. Tämän pohjalta kehitin Unitylla peliprototyypin, jota testautin vapaamuotoisesti yliopistokurssin ryhmäkave-reillani. Kurssilaisilta saamastani palautteesta (tai sen puutteesta) sain käsityksen, ettei peliprototyyppi kommunikoinut tarpeeksi hyvin sekvensserin olemassaoloa. Näin pelikokemus jäi vaillinaiseksi. Päätin hylätä tämän pelin kehittämisen. Seurauksena halusin tehdä puhtaana instrumentin, joka näyttää selvästi rumpukoneelta ja jonka käyttöliittymä olisi intuitiivinen. Jo käyttöliittymää katsoessaan käyttäjä siis ymmärtäisi paremmin, mitä sovelluksella on mahdollista tehdä. Esittelin instrumenttiprototyyppejä yliopistokurssilla ja siitä pidettiin. Vielä tässä vaiheessa sekvensserin toiminta perustui kuitenkin tahdin kerrallaan soittavaan soittoluuppiin eli viiveellä käyttäjän syötteisiin reagoivaan versioon (kuvio 13).

Tämän jälkeen aloitin uuden pelin rakentamisen erillään musiikkikomponentista. Peli-idea

perustui tuolloin fysiikan mallinnukseen perustuvaan golf-tyyliseen ajatukseen. Pelaajan tarkoituksena oli saada pallo pomppimaan siirreltävien objektien kautta reikään tai maaliin. Maali ilmestyi joka kerta satunnaiseen kohtaan kentän alareunassa. Testasin tätä prototyyppiä itse ja kehittäjänä minusta tuntui, että palloa oli liian vaikeaa saada osumaan maaliin. Sen sijaan pallon kimmoittelu liikuteltavien objektien välillä tuntui helpolta. Tästä seurasi ajatus, jonka seurauksena pelimekaniikat saivat nykyisen muotonsa. Hylkäsin ajatuksen paikallaan olevasta maalista. Sitten muutin pelaajan tavoitteeksi saada kimmoiteltua pallo tiettyssä järjestyksessä liikuteltavien objektien välillä. Tämä mekaniikka tuntui pelinkehittäjänä hauskalta ja koukuttavalta kaikessa yksinkertaisuudessaan.

Tämän jälkeen yhdistin pelin ja aiemmin rakentamani instrumentin. Lisäisin ominaisuuden, jossa pallon kimpoaminen lisäsi yhden nuotin musiikkikomponentille. Tässä vaiheessa musiikinluontipelin määritelmä 4 oli täyttynyt. Kun prototyypistä oli syntynyt musiikinluontipeli, testautin prototyyppiä vapaamuotoisesti useammalla MUPS-kurssilaisella ja kurssin luennoitsijalla. <sup>1</sup> Sain parannusehdotuksia ja myös hyvää palautetta pelin toimivuudesta mobiilipelinä. Sain tärkeää palautetta käyttöliittymään liittyvistä asioista, joille olin itse ehtinyt sokeutua kehitysprosessin aikana. Eräs merkittävä palaute liittyi pelin ja instrumentin välisen mappauksen parantamiseen. Prototyyppi siis kaipasi selkeyttä siihen, kuinka pelin tapahtumat vaikuttavat instrumentin toimintaan ja musiikin soittamiseen. Pelinkehittäjänä tulkitsin tämän tarpeena parantaa erityisesti pelin graafista palautetta nuottien soiton yhteydessä.



Kuvio 12. Kehitysprosessin merkittävät vaiheet

1. Musiikkiteknologiaprojekti eli MUPS oli kurssi, jossa kehitettiin musiikillisia ja pelillisiä sovelluksia

## 5.4 Unity, käytänteitä ja suunnittelumalleja

Unity on pelimoottori 2D- ja 3D-pelien tekemiseen.<sup>2</sup> Käytän tässä työssä Unity-pelimoottoria. Pelimoottori on itselleni tuttu ja yksi käytetyimmistä pelimoottoreista pelinkehittäjien keskuudessa. Unityn käyttämistä tutkimuksessa puoltaa myös se, että tällöin voidaan saada usealle kehittäjälle hyödyllistä tietoa pelinkehitysprosessista. Unity on ladattavissa ilmaiseksi, mutta siitä on olemassa myös maksullisia versioita.

Musiikinluontipelin sekvensserin kehittämisessä Unityn AudioSource.PlayScheduled -komento on oleellinen (Unity Technologies 2018). PlayScheduled-komennolla voidaan pelimoottorille asettaa etukäteen täsmällinen aika äänen soittamiseen. Soittoaika annetaan AudioSource.dspTime -ajan mukaan<sup>3</sup> DspTime antaa tiedon ajasta sekunneissa ja perustuu samalla äänijärjestelmän käsittelemien äänten lukumäärään. DspTimen aika on huomattavasti tarkempi kuin Time.time-komennon antama aika. PlayScheduled ei ole riippuvainen pelimoottorin kuvanpäivitysnopeudesta, kuten tavalliset äänikomennot. Juuri tämä seikka tekee PlayScheduled-komennosta tärkeän. Oman kokemukseni mukaan riippumattomuus kuvanpäivitysnopeudesta auttaa huomattavasti äänen soittamista tasaisessa rytmissä, mikä on oleellista musikaalisissa toiminnoissa. Kuten nimi PlayScheduled viittaa, tulee soittokomento antaa etukäteen, mikä vaatii ohjelmistolta hieman erilaista rakennetta äänien soittamiseen kuin perinteisesti reaaliajassa soitettavat peliäänet. Tämä tekninen seikka johtaa tahattomasti myös Herberin mainitsemaan asynkronisuuteen (luku 4.3.1).

Käytän ohjelmistossa Object Pool -nimistä suunnittelumallia.<sup>4</sup> Object Poolin tarkoituksena on optimoida muistinkäyttöä ja muistin roskienkeruuta niin, että usein käytettäviä asioita luodaan etukäteen varastoon. Asioita voidaan sitten ottaa varastosta käyttöön ja laittaa takaisin käytön jälkeen "uusiokäyttöä" varten. Tällöin olioita ei tarvitse jatkuvasti luoda ja tuhota, mikä ennen pitkää aiheuttaa roskienkeruuta. Tällaista mallia voidaan hyödyntää esim. äänien soittamisessa, useiden samanlaisten vihollishahmojen kanssa tai visuaalisissa efekteissä.

---

2. Unity-manuaali <https://docs.unity3d.com/Manual/UnityManual.html>

3. AudioSource.dspTime. <https://docs.unity3d.com/ScriptReference/AudioSettings-dspTime.html>

4. Object Pool -malli <http://gameprogrammingpatterns.com/object-pool.html>

## 5.5 Musiikkikomponentti

Kehitetty musiikkikomponentti toimii oleellisena musikaalisena osana musiikinluontipelissä. Pelin tulee pystyä antamaan komponentille syötteitä reaaliajassa. Syötteet nauhoitetaan nuotteina, ja nuotit tulee olla mahdollista soittaa heti tai pienellä viiveellä nauhoittamisen jälkeen. Tällöin toteutuu aktiivisen nuotin ominaisuus. Erilaiset pelitapahtumat tai pelaajan antamat komennot ovat laukaisimina komponentille annetuille syötteille. Komponentin tulee voida kirjoittaa nuotit niin, että ne pysyvät rytmissä tietyn tempon mukaisesti. Tietyissä mielessä tämä rajaa pelaajan ilmauksellista vapautta, mutta toisaalta auttaa pelaajaa olemaan säveltämättä mielivaltaisen kuuloista teosta (Pichlmair ja Kayali 2007, s.427). Komponenttia voi ajatella sekvensserinä, joka nauhoittaa nuotteja pelitapahtumien perusteella.

### 5.5.1 Säveltämistä tukevat ominaisuudet

Instrumenttipelien oleellisina ominaisuuksina voidaan pitää kvantisointia, ääniagenteja, aktiivista nuottia ja leikkiä (Pichlmair ja Kayali 2007, s.428). Esimerkiksi Pichlmair ja Kayali (2007, s.428) luettelemat kaikki instrumenttipelit sisälsivät 3-4 edellä mainituista ominaisuuksista. Komponentin on siis oleellista tukea kvantisointia, ääniagenteja ja aktiivista nuottia. Sen sijaan leikin sisältyminen riippuu olennaisesti pelin mekaniikoista eikä niinkään musikaalisen komponentin toteutuksesta. Tässä tutkimuksessa ohjelmistosta rakennetaan peli, joka ei siis ole vain leikkiä. Työn konstruktioita suunniteltaessa leikkiä pyritään vaalimaan pelikokemuksellisena asiana sen sijaan, että pelin tavoitteista luovuttaisiin kokonaan.

Komponentin tulee voida nauhoittaa nuotit tiettyyn rytmiin. Tämä tarkoittaa komponentin tukevan kvantisointia (Pichlmair ja Kayali 2007, s.427). Komponentti mahdollistaa nuottien soittamisen pelitapahtumien perusteella. Äänen soittaminen tulee kommunikoida pelaajalle myös visuaalisesti. Äänen soittotapahtuma tulee siis olla mahdollista kytkeä johonkin olioön, joka soittamishetkellä näyttää visuaalisen animaation. Tällöin komponentti tukee ääniagenttien käyttöä. Tämä tosin vaatii, että peliosa on mekaniikoiltaan kehitetty niin, että peli sisältää ääniä päästeleviä visuaalisia agenteja tai otuksia. Ääniagenttien sisältyminen peliin riippuu siis yhtäältä pelin visuaalisen käyttöliittymän toteutuksesta ja toisaalta musiikkikomponentin tuesta. Kun komponentti mahdollistaa nuottien nauhoittamisen sekä soittamisen, voidaan sa-

noa komponentin tukevan aktiivista nuottia (Pichlmair ja Kayali 2007, s.426). Jos nämä ominaisuudet löytyvät komponentista, se sisältää instrumenttipelien oleellimmat vaatimukset.

Musiikkikomponentti on yleiskäyttöinen eli sitä voidaan käyttää useiden erilaisten musiikinluontipelien kehityksessä. Tutkielmassa kehitetään musiikinluontipelin prototyyppi, jonka tukena käytetään kehitettyä musikaalista komponenttia.

### 5.5.2 Äänensoittoaluoppi

Unity tarjoaa siis PlayScheduled-komennon, jolla voidaan antaa äänelle täsmällinen soittoaika. Jotta komentoa voidaan hyödyntää kunnolla, tulee ensin kehittää järjestelmä, jolla äänille voidaan antaa soittoajat etukäteen. Musiikkikomponentissa eräs toteutusvaihtoehto tästä järjestelmästä on luoppi (kuvio 13), jossa äänien soittoajat määritetään aina tahdin alussa. Tässä toteutuksessa tulee huomioida, että käyttäjän tai pelin musiikkikappaleeseen tekemät muutokset huomioidaan vain tahdin alussa. Jos sitten tahdin soittaminen on jo alkanut, ja käyttäjä tekee muutoksia kyseiseen tahtiin, soitetaan muutokset vasta seuraavalla kerralla, kun tahti soitetaan uudelleen.

```
void Update () {
    if (!running) return;

    double time = AudioSettings.dspTime;
    // Tahdin alussa määritetään tahdin kaikille nuoteille soittoajat
    if (time + oneBar > nextEventTime) {
        if (tempoClick) PlayClick(time);
        // Kaydaan lapi jokainen instrumenttiraita
        for (int i = 0; i < currentSong.tracks.Count; i++)
        {
            Track track = currentSong.tracks[i];
            // Haetaan tamanhetkinen tahti
            Bar bar = track.giveBar(barIndex);
            // Kaydaan lapi tahdin kaikki nuotit
            for (int k = 0; k < bar.DividedInto; k++)
            {
                Note note = bar.GetNote(k);
                if (note != null)
                {
                    double nextTime = nextEventTime + eightBeat * k;
                    track.instrument.PlayNote(note, nextTime);
                }
            }
        }
        nextEventTime += oneBar;
    }
}
```

Kuvio 13. Äänensoittoaluoppi tahti kerrallaan

Selitän tässä hieman luoppikoodin toimintaa. Luoppi pyörii Update()-funktiossa, joka käydään läpi jokaisella kuvanpäivityksellä eli framella. Alkupuolella kysytään ohjelman tämänhetkinen suoritus aika `AudioSettings.dspTime`-komennolla. Jos sitten suoritusajan ja yhden tahdin keston summa on juuri ylittänyt seuraavan `nextEventTime`-hetken, aletaan määrittää seuraavan tahdin sävelten soittoaajat. For-silmukoissa käydään läpi jokainen instrumenttiraita, jokaisen raidan tämänhetkinen tahti ja tahdin kaikki nuotit. Jos tahdistä löytyy soitettavia nuotteja, annetaan niille soittoaika kahdeksasosanuotin tarkkuudella.

Toinen tapa rakentaa luoppi on se, että käydään nuotit läpi aina pienimmän määritetyn aikaarvon välein (kuvio 14). Prototyypissä olen käyttänyt kahdeksasosanuottia pienimpänä aikaarvona. Tällöin soittoaajat määritetään joka kahdeksasosanuotilla eikä pelkästään tahdin välein. Tällöin on mahdollista soittaa käyttäjän kappaleeseen tekemät muutokset pienemmällä viiveellä. Tässä ratkaisussa määritetään soittoaajat vain juuri seuraaville nuoteille eikä koko tahdin nuoteille kuten aiemmassa vaihtoehdossa.

`PlayScheduled`-komennon hyöty tulee juuri siinä, että Unityn äänijärjestelmä pystyy valmistautumaan äänen soittamiseen etukäteen, joten ääni on mahdollista soittaa täsmällisesti oikeassa ajassa. Vaatisi kuitenkin tarkempaa testausta määrittää aika, milloin tämä hyöty ei enää päde. Tarkoitan tilannetta, jossa etukäteen määritetty soittoaika on liian lähellä ohjelman senhetkistä suoritus aikaa eikä järjestelmä ehdi valmistautua soittamiseen. Tällöin äänet alkaisivat soida enemmän tai vähemmän väärinä hetkinä. Varmastikin tämä riippuu myös laitteen suorituskyvystä. Annetussa esimerkissä (kuvio 14) kutsutaan myös `UI.TimeTickMessage()` -komentoa kahdeksasosanuotin välein, millä siirretään käyttäjälle näytettävää aikapalkkia. Aikapalkki näyttää käyttäjälle missä kohdassa kappaleensoitto on menossa.

### **5.5.3 Äänen sävelkorkeuden muuttaminen**

Usein musiikintuotantoon käytetyt samplerit sisältävät eri instrumenteista muutaman nauhoitetun sävelen, joista muodostetaan matemaattisella kaavalla muut sävelet. Tällä tavalla säästetään muistia ja toisaalta vaivaa nauhoitussessioissa. Olen itse käyttänyt samanlaista mallia musiikkikomponentissa. Sävel A on standardisoitu vastaamaan 440 hertsin taajuutta.

```

void Update () {
    double time = AudioSettings.dspTime;
    // Maaritetaan soittoajat kahdeksasosanuotin valein
    if (time > nextTinyEventTime) {
        myUI.TimeTickMessage(eightBeatIndex, barIndex, -1);

        // Kaydaan lapi vain yksi nuotti kustakin instrumenttiraidasta
        for (int i = 0; i < currentSong.tracks.Count; i++)
        {
            Track curTrack = currentSong.tracks[i];
            Bar bar = curTrack.giveBar(barIndex);
            Note note = bar.GetNote(eightBeatIndex);

            if (note != null)
            {
                double nextTime = nextTinyEventTime + eightBeat;
                List<Note> curChord =
                    currentSong.chordTrack.GetChordNotes(barIndex, eightBeatIndex);
                curTrack.instrument.PlayNote(note, curChord, nextTime);
            }
        }
    }
}

```

Kuvio 14. Äänensoittoluuppi kahdeksasosanuotin välein

Tätä käytetään laajasti musiikissa ja on tärkeä seikka, mikäli halutaan oman soittimen olevan samassa vireessä muiden soittimien kanssa. Käytän musiikkikomponentissa kullekin instrumentille säveltä A, jonka muuttamiseen muiksi säveliksi löytyy kaava. Kaavan esitti tutkija Simon Stevin jo 1600-luvulla (Cohen 1987, s.473). Kaava perustuu siihen, että puolissävelaskel vastaa irrationaaliluvun  $\sqrt[12]{2}$  suuruista muutosta äänentaajuudessa. Sävelkorkeuksien kertoimet saadaan laskettua kaavalla:

$$2^{(n-1)/12}$$

Edelleen sävelkorkeuksien taajuudet saadaan laskettua kaavalla:

$$440 \times 2^{(n-1)/12}$$

, missä n on halutun sävelen indeksi. Sävelet ovat järjestyksessä a = 1, ais, h, c, cis, d, dis ja jne. Kaavalla voidaan siis laskea sävelet puolissävelaskeleen tarkkuudella, mikä on usein tapana länsimaisessa musiikissa. Unityn sisällä voidaan sitten käyttää komentoa AudioSource.pitch, jolle voidaan antaa kaavalla laskettu kerroin. Tällöin tosin myös äänen soit-



tonopeus muuttuu. Tämä ei kuitenkaan ole vakavaa, jos soitt nopeuden muutos pysyy kohtuullisissa rajoissa, tai muutos kuulostaa sopivalta ohjelmiston äänimaisemassa. Musiikin tuotannossa käytetään usein useampia säveliä, joista muut sävelet muodostetaan, pelkän yhden sävelen sijaan. Päädyin tässä työssä yhteen säveleen perustuvaan samplaamiseen, joka on riittävä ja samalla yksinkertainen toteutukseltaan.

#### 5.5.4 Sävelasteikot

Sävelkorkeuksien lisäksi on hyvä käyttää sävelasteikoita, jotta musiikki olisi miellyttävämmän kuuloista. Sävelasteikon perussävel on asteikon ensimmäinen sävel. Sävelasteikon loput sävelet ovat aina tietyn intervallin päässä toisistaan. Duurisävelasteikossa intervallit perussävelestä laskien ovat K-K-P-K-K-K-P, missä K tarkoittaa kokosävelaskelta ja P puolissävelaskelta. Tällöin C-duuriasteikko sisältää sävelet c, d, e, f, g, a, h, c. Luonnollinen molliasteikko sisältää intervallit K-P-K-K-P-K-K. Tällöin c-molli asteikko sisältää sävelet c, d, dis, f, g, gis, ais, c.

Sävelasteikon intervallit on helppoa syöttää tietokoneelle, jolta sitten voidaan pyytää sävelasteikkoon sopivia säveliä. Musiikkikomponentissa olen käyttänyt kokonaislukutaulukkoa ilmaisemaan sävelasteikon intervaleja. Esimerkiksi duuriasteikkoa kuvaa taulukko [0, 2, 4, 5, 7, 9, 11, 12], missä yksi kokonaisluku vastaa puolissävelaskeleen hyppyä. Tämänkaltaisen esitys on hyvin samankaltainen kuin Kaliakatsos-Papakostas ym. (2017) ovat käyttäneet omassa työssään.

#### 5.5.5 Unityn Audio Mixer

Unity tarjoaa valmiina äänimikserin, jolla voidaan ohjata äänet eri kanaviin, säätää äänentasoja ja lisätä efektejä.<sup>5</sup> Unity nimittää yksittäistä mikserikanavaa nimellä group. Prototyypissä olen luonut groupeja eri soittimia varten kuten bassolle, melodiasoittimelle, rumuille ja virvelille. Näin eri soittimien äänentasot saadaan miksattua toisiinsa sopiviksi. Groupit ovat hyödyllisiä myös efektien lisäämiseen. Esimerkiksi basso-groupille olen lisännyt PitchShifter -efektin, jolla olen pudottanut basson äänentaajuutta, mikä saa basson kuulostamaan en-

---

5. Unity Audio Mixer <https://docs.unity3d.com/Manual/AudioMixer.html>

tistä matalammalta ja voimallisemmalta. Unity tarjoaa valmiina useita musiikkituotannon puolelta tuttuja efektejä.

Prototyypissä AudioPlay -objektit vastaavat varsinaisesta äänensoittamisesta. Kullakin AudioPlaylla on AudioSource -komponentti. AudioPlayn AudioSource -komponenttiin olen määrännyt tietyn ulostulo-groupin. Olen tehnyt kutakin ulostulo-groupia vastaamaan yhden AudioPlay -objektin. Näin jokaiselle instrumentille voidaan antaa haluttu AudioPlay -objekti, joka ohjaa instrumentin äänet haluttuun kanavaan eli groupiin. Tämä mahdollistaa äänimikserin eri kanavien käytön.

Mikserin äänentasot ja efektien parametrit voidaan tallentaa snapshot-tilannevedokseen. Useampien tilannevedosten välillä voidaan tehdä siirtymiä, joilla voidaan luoda erilaisia tunnelmia. Esimerkiksi kappale voisi alkaa niin, että vain bassoraita olisi kuuluvissa ja myöhemmin toiseen tilannevedokseen siirryttäessä tulisivat mukaan muutkin soittimet. Jos mikserin parametreja haluaa säätää ohjelmakoodissa, täytyy parametrit ensin saattaa näkyviin (expose) Unityn editorin puolella. <sup>6</sup>

## 5.6 Pelikomponentti

Päätin lähteä suunnittelemaan pelikomponenttia kasuaalin mobiilipelin suuntaan. Mielessäni oli ajatus fysiikan mallinnukseen perustuvasta kaksiulotteisesta pelistä. Asiantuntijuuteni mukaan fysiikan mallinnus altistaa pelin erilaisille tapahtumille, yllätyksille ja emergenssille. Tässä mielessä fysiikan mallinnus on pelille hyvinkin otollinen lähtökohta pelisuunnittelullisesti. Schell (2008, s.408) puhuu teknologiasta, jonka hyödyntäminen pelissä voi olla perustavanlaatuisista tai dekoratiivista. Tässä pelissä fysiikan mallinnus tulisi olemaan perustavanlaatuisessa asemassa eikä vain dekoratiivisessa roolissa. Enemmän voikin miettiä, tuleeko musiikinluontiosuus olemaan pelissä vain dekoratiivisessa roolissa. Ehkäpä juuri se, että pelaaja pääsee myös itse luomaan musiikkia ja vaikuttamaan yksittäisiin nuotteihin, tekee musiikinluonnista astetta enemmän perustavanlaatuisista eikä vain proseduraalista taustamusiikkia. Avoimeksi kysymykseksi kuitenkin jää, onko musiikinluontia mahdollista sitoa

---

6. Exposed AudioMixer Parameters <https://unity3d.com/learn/tutorials/topics/audio/exposed-audiomixer-parameters>

pelillisiin tavoitteisiin mielekkäällä tavalla. Joka tapauksessa lähtökohta tämän pelin suunnittelulle on muodostaa pelaajalle selkeä tavoite.

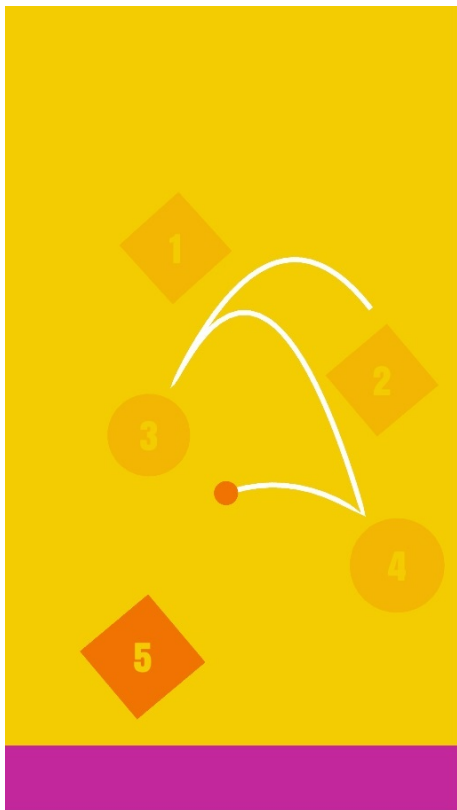
Pelaajan tavoite on saada pallo kimpoilemaan liikuteltavien Movable-objektien välillä oikeassa järjestyksessä (kuvio 15). Liikuteltavat objektit on numeroitu, mikä näyttää pelin vaatiman kimpoilujärjestyksen. Pelaaja läpäisee kentän osuessaan kaikkiin objekteihin oikeassa järjestyksessä. Tämä on yhden kentän tavoite kaikessa yksinkertaisuudessaan. Tavoite on oleellinen asia, jotta peliä voidaan kutsua määritelmän 1 mukaiseksi peliksi.

Kenttien haaste tulee osaltaan fysiikan mallinnuksesta ja pallon yllättävistä kimpoamisista. Haastetta helpottaa pallon jättämä vana, joka näyttää mitä reittiä pallo on lentänyt. Pelaaja voi vanan perusteella liikutella objekteja pallon seuraavalle todennäköiselle kulkuradalle. Sisäisessä pelitestauksessa vana osoittautui oleelliseksi asiaksi, joka sujuvoittaa pelin pelaamista. Vana sai myös kiittäviä kommentteja ulkoisessa epäformaalissa pelitestauksessa. Vanan poisjättäminen tuntuisi mielestäni enemmänkin pelaajaa ärsyttävältä seikalta eikä toisi peliin haluttua mielekästä lisähaastetta. Lisähaastetta tuovat Obstacles-esteet, joita alkaa esiintyä lisääntyvässä määrin sitä mukaa, kun pelaaja etenee pelissä.

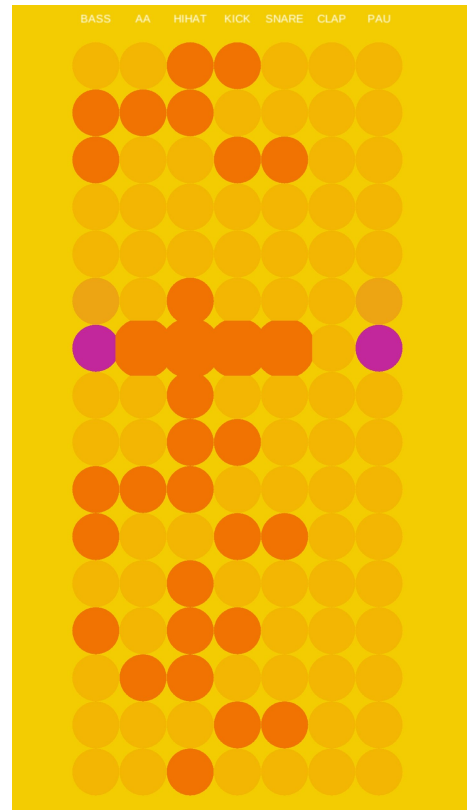
Pelin yksinkertaista ideaa on mahdollista laajentaa todella monilla eri tavoilla. Peliin on mahdollista suunnitella erilaisia liikuteltavia objekteja, jotka käyttäytyvät uusin tavoin ja jotka luovat vaihtelua ja uutta haastetta.

## **5.7 Peli- ja musiikkikomponentin yhteenliittäminen**

Ohjelmistokehitysprosessin alussa ohjelmoin peli- ja musiikkikomponentteja erillään toisistaan. Käytännössä molemmat olivat eri projekteja Unityssa. Paras löytämäni keino, yhdistää projektit, on käyttää Unityn sisältämää Export Package -komentoa. Tällä voidaan tehdä projektin halutuista osista ja skripteistä paketti, joka voidaan Import Package -komennolla integroida toiseen projektiin. Paketinluontivaiheessa voidaan myös jättää ottamatta pakettiin mukaan turhia tai projekteille päällekkäisiä tiedostoja. Paketin luominen ja tuominen toimii paremmin kuin raaka tiedostojen siirtely projektikansioista toiseen.



Kuvio 15. Pelinäkömä.



Kuvio 16. Musiikinäkömä.

Ennen paketin luomista tein musiikkikomponentin osista yhden ison prefabin<sup>7</sup>, joka sisälsi koko musiikkikomponentin toiminnallisuuden. Sitten tein musiikkikomponentista paketin, jonka toin sisälle peliprojektiin. Tällä tavalla integrointiprosessi oli melko kivutonta toteuttaa.

Olin suunnitellut pelin toimimaan niin, että mobiililaitetta pidetään pystyasennossa. Puolestaan musiikkikomponentin olin alun perin suunnitellut katseltavaksi vaakaa-asennossa. Kuitenkin pelin osalta oli oleellisempaa, että pelaaminen tapahtuu pystyasennossa. Tämän vuoksi päätin kääntää myös musiikkikomponentin käyttöliittymän toimimaan pystyasennossa (kuvio 16). Tähän meni hieman aikaa, mutta prosessi ei ollut kuitenkaan turhan vaikea. Tältä vaiheelta olisi voinut välttyä, jos molemmat projektit olisi suunnitellut toimimaan samassa orientaatioissa alusta lähtien. Muuten tämän mittakaavan projektien yhdistäminen sujui melko lailla ongelmitta. Tästä eteenpäin jatkoin molempien komponenttien kehittämistä saman projektin sisällä.

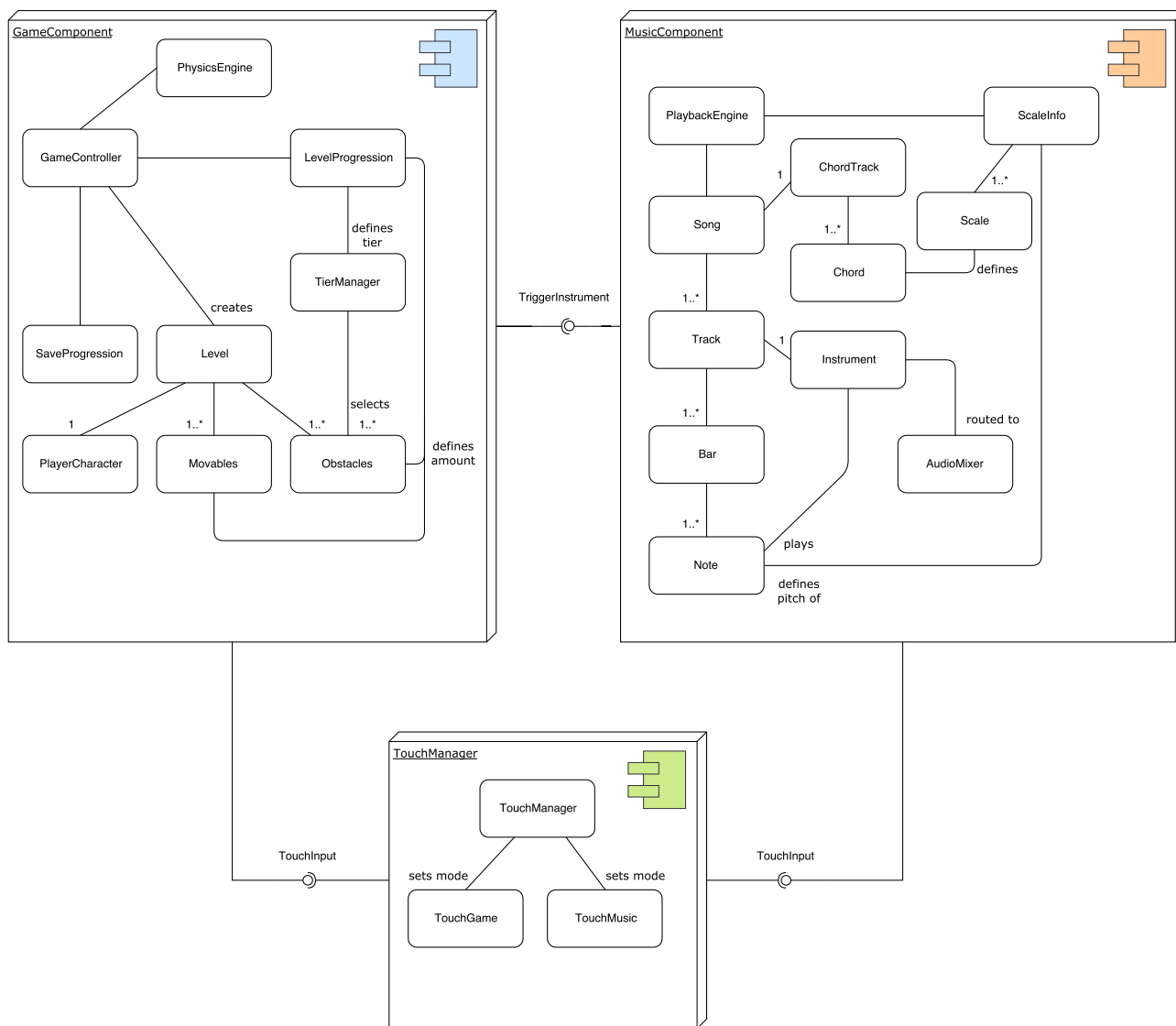
## 5.8 Musiikinluontipelin arkkitehtuurista

Annan tässä erään kuvauksen prototyypin arkkitehtuurista (kuvio 17). Tässä kuvauksessa olen jakanut ohjelmiston peli- ja musiikkikomponenttiin sekä käyttäjän syötteitä kuuntelemaan komponenttiin. Olen keskittynyt kuvauksessa työn aikana kehitettyjen osien toimintaan enkä niinkään esimerkiksi pohjalla olevaan pelimoottoriin.

Pelikomponentissa pääosassa on GameController. GameController hallinnoi lähes kaikkea, mitä pelissä tapahtuu. GameController luo kentät LevelProgression-osan antaman tason mukaan. Taso vaikuttaa pelin Movable- ja Obstacles-objektien määrään sekä siihen, millaisia objekteja kenttään luodaan. Tarkoitus olisi, että erilaiset kenttien objektit loisivat uudenlaista ja lisääntyvää haastetta pelaajalle. Erilaisia objekteja ei ole toteutettu nykyiseen prototyyppiin monia. Pelikomponentti käyttää musiikkikomponentin tarjoamaa TriggerInstrument rajapintaa, jolla voidaan syöttää yksittäisiä nuotteja m-komponentille pelitapahtumien perusteella.

---

7. Prefab on Unityn tapa tallentaa peliobjektista malli, josta voidaan luoda useita instansseja. Prefabiin tehdyt muutokset heijastuvat kaikkiin luotuihin instansseihin.



Kuvio 17. Musiikinluontipelin arkkitehtuurikuvaus

Musiikkikomponentin sydän on PlaybackEngine, jossa pyörii äänentoistoluoppi (kuviot 13 ja 14). PlaybackEnginellä on tieto soitettavista Song-kappaleista. Kullakin kappaleella on useampia soittoraitoja ja yksi sointuraita. Kukin soittoraita sisältää nuotit yhdelle instrumentille. Instrumenttien soitto ohjataan mikseriin, jolla voidaan säätää äänenvoimakkuutta ja lisätä efektejä. Sointuraita määrittää kappaleessa käytetyt soinnut, joiden mukaan ScaleInfo osaa antaa sävelasteikkoon sopivia säveliä. Sointuraita sisältää myös tiedon siitä, millä ajankohdalla soitto vaihtuu. Tässä kohdin järjestelmän olisi voinut tehdä myös pelkästään sävelasteikkoihin perustuvaksi. Tämä mahdollistaisi käsitykseni mukaan laajemmin muun kuin länsimaisen musiikin soittamisen. Halusin kuitenkin pohjata musiikin sointuraitaan, koska pelitestauksessa muodostunut musiikki kuulosti miellyttävämmältä ja runsaammalta sointuraidan kanssa kuin ilman sointuraitaa.

TouchManager kuuntelee käyttäjän syötteitä, joilla käyttäjä voi ohjata peli- ja musiikkinäkymää. TouchManagerille määritetään jokin ele tai komento, jonka perusteella TouchManager valitsee joko peli- tai musiikkinäkymän ohjauksen.

## **6 Musiikinluontipelin arviointi**

Tässä kappaleessa reflektoin valmistunutta prototyyppiä kirjallisuuteen luovista musiikkipeleistä, instrumenttien rakentamisesta ja pelisuunnittelusta. Viitataan valmistuneeseen konstruktion termillä prototyyppi.

### **6.1 Musiikinluontipelin määritelmän toteutuminen**

Valmiissa prototyypissä täyttyy musiikinluontipelin määritelmän 4 kohdat 1 ja 2. Valmiissa pelikomponentissa on ongelmanratkaisuvälineitä, mikä on Schellin pelin määritelmän 1 mukaista. Peli siis sisältää tavoitteen eikä ole vain leikkiä. Musiikkikomponentti tukee aktiivista nuottia, koska komponentilla on mahdollista muokata soitettavaa nuottia tai partituuria eri esityskerroille. Myös ML-pelin määritelmän 3. kohta täyttyy, sillä pelaamisen aikana musiikkia voidaan muokata ja soittaa.

Peli- ja musiikkikomponentin käyttöliittymät ovat erilliset. Tämä ei kuitenkaan estä sitä, että musiikinluontipelin määritelmän 4 kohta 3 täytyisi. Pelin käyttöliittymän kautta musiikkia muodostuu paremminkin generatiivisesti (luku 4.3.2), kuten Herber (2006) mainitsee. Puolestaan musiikkikomponentin käyttöliittymän kautta käyttäjä pääsee muokkaamaan musiikkia yksityiskohtaisemmin ja helpommin haluamaansa suuntaan. Peli voi auttaa sävellyksen muodostumista alkuvaiheessa. Toisaalta pelin satunnaiset syötteet voivat saada musiikin kuulostamaan liian satunnaiselta.

### **6.2 Reflektio luovien musiikkipelien kirjallisuuteen**

Prototyypin musiikin muodostuminen on asynkronista (luku 4.3.1). Tätä Herber (2006) kuvailee tutkimuksessaan. Tämän työn kaltaisessa ohjelmistossa asynkronisuuden suuruus riippuu soittoluupin toteutustavoista. Soittoluupin toteutus vaikuttaa siis siihen, kuinka paljon ruudulla nähdystä tapahtumista kuluu aikaa ennen kuin luotu musiikki soitetaan ja kuullaan. Asynkronisuuden ja responsiivisuuden suhde on mielenkiintoinen ongelma. Usein peleissä halutaan rakentaa käyttäjän syötteiden ja pelimekaniikkojen suhteesta mahdollisimman res-



ponsiivinen. Herber näkee asynkronisuuden positiivisena asiana, joten pelinkehittäjän tulee päättää, haluaako hän rakentaa ML-peliä responsiivisuus vai asynkronisuus edellä. Tosin tämän työn prototyypissä voidaan valita enemmän tai vähemmän asynkroninen vaihtoehto, koska PlayScheduled-komennon käyttö johtaa väistämättä asynkronisuuteen.

Musiikillista emergenssiä (luku 4.3.3) tapahtuu prototyypissä ehkäpä eniten sävelasteikkoihin ja rytmisiin sääntöihin perustuen. Herber (2006, s.33) mainitsee emergenssin tärkeänä piirteenä kokeilevassa ja generatiivisessa musiikissa. Näin on tehdyn prototyypinkin kohdalla. Kun ohjelmoidaan instrumentti soittamaan vain halutun sävelasteikon mukaisia säveliä, voidaan soittaa aina suhteellisen hyvän kuuloista musiikkia. Kun samalla käytetään kvantisointia, pysyy musiikki aina halutussa rytmissä. Näistä rajoituksista huolimatta erilaisia sävelten ja rytmien kombinaatioita voidaan muodostaa erittäin paljon. Kun melodian seuraksi lisätään bassoraita ja soinnut, on erilaisia musiikillisia kombinaatioita sitäkin enemmän. Prototyyppi siis sisältää emergenssiä, joka pohjautuu suhteellisen yksinkertaisiin musiikillisiin sääntöihin. Kaikki kombinaatiot eivät varmastikaan kuulosta yhtä hyviltä. Musiikillisen tuotoksen arviointi on toki myös subjektiivista ja riippuu myös siitä, minkälaista musiikkia halutaan kulloinkin kuulla. Joka tapauksessa järjestelmän sisäänrakennetut säännöt tuottavat odottamattomia seurauksia eri pelikerroilla.

Prototyyppi kuuluu Lieben artikkelin mukaan poikkeavaan ryhmään (Liebe 2013). Prototyypin muodostama pelikokemus perustuu enemmän tietokoneen kuin pelaajan toimintaan. Interaktio prototyypin pelin ja pelaajan välillä ei ole kovin fyysistä tai esittävää. Samalla musiikki muodostuu pelatessa ohjelmoituihin sääntöihin perustuen. Jouduin hieman miettimään, onko prototyypin käyttöliittymä objektiivinen vai symbolinen. Lopulta päädyin objektiiviseen, koska käyttöliittymä perustuu paljon asioiden raahailuun kosketuksella. Tällöin pelaajan liikkeiden ja pelin tapahtumien välillä on tietty objektiivinen yhteys. Pohjaan valinnan myös siihen, että Liebe on artikkelissaan luokitellut monet kosketuseleillä toimivat pelit objektiivisiksi. Viimeiseksi pelin ääniraita on reaktiivista, sillä musiikki muodostuu pelaajan toimintojen kautta.

Löytyykö prototyypistä instrumenttipeleille ominaisia piirteitä (Pichlmair ja Kayali 2007)? Tein vertailua Pichlmairin ym. ominaisuustaulukolla prototyypin ja muutaman musiikkipeilin välillä (kuvio 18). Prototyyppi tukee kvantisointia ja aktiivista nuottia. Vaikka ääniagentit

ja leikki yhdistetään luoviin musiikkipeleihin, ne eivät päde samalla tavalla prototyypin kohdalla, koska pelistä haluttiin tehdä tavoitteellinen peli. Pichlmair ja Kayali (2007) esittävät kuvauksen ääniagenteista, jotka alkavat olla ääniagenteja vasta, kun peli ei enää sisällä tavoitteita ja muuttuu leikiksi. Tällöin ääniagenteilla ei siis ole muuta roolia kuin interaktio äänen kanssa. Tämän seikan mukaan prototyypin peli ei sisällä ääniagenteja, koska prototyypin ääniinkin liittyvillä objekteilla on pelin tavoitteeseen vaikuttava rooli. Sen sijaan prototyypin musiikkinäkymä sisältää ääniagenteja. Prototyypin peli ei sisällä leikkiä juuri siitä syystä, että tavoitteellinen peli ei ole leikkiä. Toisaalta taas musiikkinäkymä ei sisällä tavoitetta ja tältä kannalta on leikkiä. Mielestäni Pichlmair ja Kayal ottavat tällaisissa tapauksissa inklusiivisen näkökulman. Esimerkiksi Rezin kohdalla he näkevät pelin sisältävän leikkiä, vaikka pelin tavallinen pelitila on tavoitteellinen. Rez on mahdollista vaihtaa “travel mode”-tilaan, jolloin pelistä tulee vain lentelyä ja soittamista ilman tavoitteita. Kuitenkin prototyypin kohdalla tilanne on hieman erilainen, koska prototyyppi sisältää leikin ja leikittömyyden sekä ääniagentit ja ääniagenttisuuden yhtäaikaaisesti.

	1	2	3	4	5	6	7*
Rez	√	√	-	-	√	√	√
Electroplankton	√	√	-	√	√	√	√
Sim Tunes	√	√	-	√	√	-	-
Vib Ribbon	-	-	√	-	-	√	√
Guitar Hero	-	-	√	-	-	-	√
Prototyyppi	√	-	-	√	-	-	√

\* kvantisointi (1), ääniagentit (2), rytmitoiminta (3), aktiivinen nuotti (4), leikki (5), synestesia (6), esiintyminen (7)

### Kuvio 18. Musiikkipelien ominaisuudet

Tilannetta voisi selventää, jos ääniagenttien kohdalla voitaisiin puhua erilaisista ääniagenteista. Ääniagenteja voisi olla **tavoitteellisia** ja **tavoitteettomia**. Ensimmäiset liittyvät pelaajan ongelmanratkaisuun ja toiset eivät. Tällöin voitaisiin tarkemmin jakaa ääniagenttien roolitusta. Samalla helpotetaan ääniagenteista puhumista, kun voidaan käyttää termiä myös tavoitteellisten luovien musiikkipelien yhteydessä. Pichlmair ja Kayali (2007) mainitsevat ääniagenttien olevan tulevan tutkimuksen kohteena. Ääniagenttien jatkotutkimusta en kuitenkaan löytänyt omassa kirjallisuuskatsauksessani.

Pichlmairin ja Kayalin käsitys esiintymisestä ei ole yhtä rajaava kuin Liebellä. Pichlmair ja Kayal ottavat huomioon niin fyysisen suorittamisen kuin musiikillisenkin esiintymisen pelaamisen aikana. Mielestäni tarvittaisiin kuitenkin selkeyttä, milloin voidaan sanoa pelin sisältävän esiintymistä ja milloin ei. Itse haluaisin kuulla tarkempia perusteluja sille, miksi Pichlmair ja Kayal eivät katsoneet Sim Tunesin sisältävän esiintymistä. Taas Electroplankton sisältää esiintymistä heidän mielestään. Seuraavassa hieman pohdintaa siitä, miksi tähän johtopäätökseen ollaan ehkä päädytty.

Sim Tunesin mekaniikat ovat ehkäpä hieman hitaat esiintymiseen. Sim Tunesilla tehtävän teoksen valmistumisessa on ehkäpä jotain samaa kuin maalaamisessa tai piirtämisessä. Tarkoitan että teos on esityskelpoinen vasta valmistuneena. Tällöin prosessin näyttäminen käy enemmän opastevideosta kuin esiintymisestä. Puolestaan Electroplanktonissa tapahtuvaa toimintaa voi olla hauskeempaa seurata alusta lähtien, vaikkei varsinainen musiikillinen teos olisi vielä valmis. Tässä on jotain samaa kuin Schellin mainitsemassa seurattavuudessa.<sup>1</sup> Toisaalta Sim Tunesin alusta on PC, kun taas Electroplankton on kannettavana Nintendo DS-pelinä soveltuvampi esiintymisympäristöön. Mielestäni Electroplankton sisältää myös audiovisuaalisesti miellyttävämmän kokemuksen kuin Sim Tunes. Jos musiikillisen esiintymisen ominaisuuteen liittyy kaikki tämä, arviointi ei ole aivan yksinkertaista.

Pichlmair ja Kayali (2007, s.429) puhuvat rytmipelien fyysisestä suorituksesta ja instrumenttien musiikillisesta esiintymisestä. He eivät kuitenkaan jaottele esiintymistä (performance) ominaisuustaulukossaan (kuvio 18). Toki tässä on se hyöty, että taulukko pysyy yksinkertaisena. Kuitenkin voisi olla selkeämpää tehdä jaottelu **fyysiseen** ja **musiikilliseen** esiintymiseen. Tätä puoltaa myös se, että Liebe (2013) puhuu vain fyysisestä performanssista. Tällöin voidaan sanoa aiempaa tarkemmin, minkälaista esiintymistä mikäkin peli tukee.

Näiden ehdotusten pohjalta voidaan tehdä ominaisuustaulukko, missä on otettu huomioon erilaiset jaottelut (kuvio 19). Jaotelluissa kohdissa voidaan valita vain jompikumpi vaihtoehtoista.

---

1. Seurattavuuden (spectation) linssi numero 95 sovelluksessa Art Of Game Design: Lenses [https://play.google.com/store/apps/details?id=com.schellgames.deckoflenses&hl=en\\_US](https://play.google.com/store/apps/details?id=com.schellgames.deckoflenses&hl=en_US)

	1	2a	2b	3	4	5	6	7a	7b*
Rez	√	√	-	-	-	√	√	-	√
Electroplankton	√	√	-	-	√	√	√	-	√
Sim Tunes	√	√	-	-	√	√	-	-	-
Vib Ribbon	-	-	-	√	-	-	√	-	√
Guitar Hero	-	-	-	√	-	-	-	√	-
Prototyyppi	√	-	√	-	√	-	-	-	√

\* kvantisointi (1), ääniagentit: tavoitteettomat (2a) tavoitteelliset (2b), rytmitoiminta (3), aktiivinen nuotti (4), leikki (5), synestesia (6), esiintyminen: fyysinen (7a), musiikillinen (7b)

### Kuvio 19. Uusittu ominaisuustaulukko

Vertailen seuraavaksi prototyypin ominaisuuksia Bottin ym. One Man Band -sovelluksen (OMB) ominaisuuksiin (Bott, Crowley ja LaViola 2009). Koska prototyypissä pelataan peliä, eikä prototyyppi ole pelkästään instrumentti, päätin ettei musiikin nauhoitusta voi laittaa pois päältä. Prototyypissä pallon kimpoilun aikana eräällä tavalla nauhoitetaan musiikkia, kun pallon kimpoiluiden seurauksena asetetaan nuotteja musiikkikomponentille. Mielestäni nauhoituksen säätämiseen keskittyminen veisi pelaajan huomiota pois pelaamisesta. Koska nauhoituksen säätämistä ei ole annettu pelaajan vastuulle, on järkevää, että peli jollain tapaa säätelee musiikin muodostumista. Mielellään musiikista ei saisi muodostua kakofoniaa tai kaoottista ainakaan pelaamisen aikana. Tämä voisi johtaa epämiellyttävään pelikokemukseen. Tätä olen pyrkinyt hallitsemaan esimerkiksi sillä, että yhdelle instrumentille voi asettaa nuotteja vain rajatun määrän. Kun rajattu määrä on saavutettu, nuotteja joko poistetaan tai olemassa olevien sävelkorkeutta muutetaan.

Myöskään musiikin toistoa ei voi keskeyttää. Tämä on yksi syy siihen, että prototyypin käyttäjä voi kyllästyä musiikin jatkuvaan silmukkamaiseen toistumiseen. Kyllästymiseen mielestäni vaikuttaa myös lyhyet ja hetken kestävät samplet. Prototyypin jatkokehityksessä tulisi keskittyä musiikin tauottamiseen ja erilaisten vaiheiden ja siirtymien mahdollistamiseen ja hallintaan. Näin voitaisiin muodostaa vähemmän käyttäjää kuluttava äänikokemus.

Undo-ominaisuutta en ole toteuttanut prototyyppiin, mutta OMB sisältää undo-ominaisuuden. Päätin jättää undo-ominaisuuden pois, koska prototyypissä pelaaja ei samalla tavalla soita reaaliaikaisesti yksittäisiä nuotteja kuin OMB:ssa. Tosin välillä prototyypissä voi muodostua

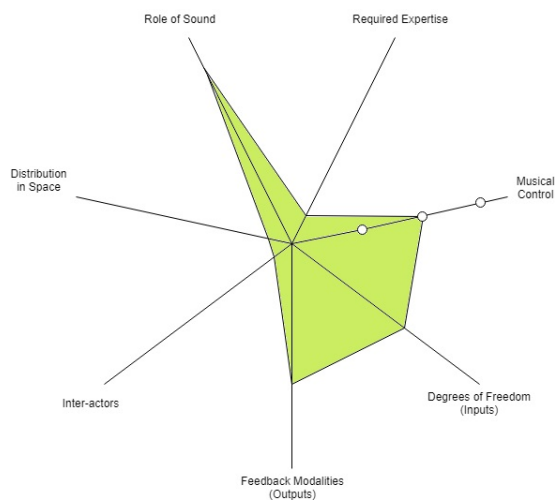
mielenkiintoisia melodioita, jotka saattavat vahingossa poistua hetken kuluttua prototyypin generatiivisen luonteen takia. Tällaisiin tilanteisiin undo-toiminto voisi olla tarpeellinen lisäys. Tällaiset ratkaisut ovat tasapainottelua työkalumaisuuden ja lelumaisuuden välillä. Esimerkiksi Electroplankton on hyvin lelumainen eikä sisällä undo-toimintoa.

### **6.3 Prototyypin arviointi instrumenttina**

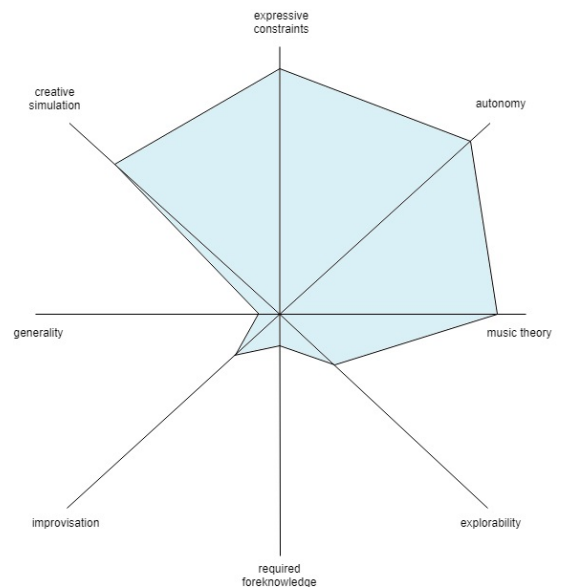
Tässä osiossa arvioin prototyyppiä Birnbaumin ym. ja Magnussonin malleilla. Birnbaumin mallilla arvioin prototyyppiä kokonaisuutena musiikillisena instrumenttina. Näin ollen en jaottele prototyyppiä peli- ja musiikkikäyttöliittymiin. Tehty arviointi on subjektiivinen näkemys, ja arviointia kannattaisi tehdä useammalla henkilöllä (Magnusson 2010, s.45). Joillain mallin akseleilla olen valinnut haluamani näkökulman arviointia varten. Esimerkiksi äänen roolia arvioitaessa riippuu henkilön omasta näkökulmasta, näkeekö musiikin ilmauksellisenä vai ympäristöllisenä eli taustamusiikkina. Tämä riippuu myös siitä, onko käyttäjä kiinnostunut ohjelmistosta pelinä, musiikkityökaluna vai luovana musiikkipelinä. Omassa arviossani sijoitin äänen roolin jonnekin ilmaisun ja ympäristöllisen välimaastoon. Arviointiin voi vaikuttaa myös se, että itselläni ei ole aiempaa kokemusta digitaalisten instrumenttien arvioimisesta. Pyrin kuitenkin suhteuttamaan prototyypin arvioinnin Birnbaumin ym. ja Magnussonin instrumenttien arviointeihin.

Birnbaum ym. (2005) mainitsevat, että heidän mallissaan instrumentit kallistuvat oikealle ja installaatiot vasemmalle puolelle kuvaa. Edelleen sekvensseriohjelmat, pelit ja lelut tyypillisesti kallistuvat oikealle puolelle, joskin niiden kattama ala on pienempi.

Arvioin prototyyppiä Birnbaumin mallin seitsemän akselin mukaan (kuvio 20). Prototyypissä vaadittu asiantuntijuus on vähäistä, musiikillista hallintaa voi tehdä nuottitasolla ja syötemahdollisuuksia on melko paljon. Syötemahdollisuuksissa arvioin pelin generatiivisten syötteiden ja musiikkinäkymän nappuloiden kokonaisuutta. Palautteen laatu on melko rikasta, sillä peli antaa visuaalista sekä auditiivista palautetta yksittäisistä painalluksista niin peli- kuin musiikkinäkymässäkin. Toisaalta kosketusnäyttö itsessään antaa jonkinlaisen palautteen käyttäjän sormenpäille (vrt. installaatiot). Käyttäjiä on vain yksi, ja tila on paikallinen. Arvioin äänen roolin olevan enemmän ilmauksellinen kuin ympäristöllinen.



Kuvio 20. Prototyypin arviointi Birnbaumin ym. mallilla.



Kuvio 21. Prototyypin arviointi Magnussonin mallilla.

Magnussonin mallilla analysoin prototyypin epistemisiä ominaisuuksia (Magnusson 2010). Mallin akselit tuovat mielenkiintoisia näkökulmia instrumentin analysointiin. Vaikka malli onkin suunnattu digitaalisten instrumenttien arviointiin, olisin mielelläni halunnut nähdä Magnussonilta esimerkkinä vaikkapa viulun arvioinnin mallilla. Tämä olisi helpottanut akselien ominaisuuksien ymmärtämistä. Ilmauksellisten rajoitteiden akselin kohdalla mietin, onko esimerkiksi viulu perinteisenä soittimena sellainen, joka sisältää mahdollisimman vähän ilmauksellisia rajoitteita. Tarkoitin että jos halutaan rakentaa mahdollisimman ilmauksellisista rajoitteista vapaa soitin, onko ihanteena rakentaa soitin, joka on jollain tapaa mahdollisimman lähellä perinteisiä soittimia. Ilmauksellisten rajoitteiden kohdalla voisi ajatella myös, kuinka paljon instrumentti rajoittaa itse itseään tai toisin päin, kuinka hyvin instrumentti tarjoaa omat ominaisuutensa käyttäjän hyödynnettäväksi.

Prototyyppi sisältää paljon ilmauksellisia rajoitteita, on hyvin autonominen ja sisältää melko paljon musiikin teoriaa (kuvio 21). Prototyyppi tarjoaa jonkin verran syvyyttä, ei vaadi juurikaan ennakkotietoja ja tarjoaa vähän mahdollisuutta improvisointiin. Prototyyppi ei ole yleisluonteinen. Luova-simulaatio -akselilla prototyyppi on uudenlainen pelillisellä käyttöliittymällään.

Jatkokehityksen kannalta olisi mielenkiintoista laajentaa prototyypin tutkittavuutta. Pidän mielenkiintoisena kysymyksenä sitä, kuinka instrumentin ominaisuuksista on mahdollista saada mahdollisimman tutkittavia tai syviä, mutta silti pitää vaatimusta esitiedoista matalana. Asiantuntijuuteni mukaan prototyypin luonnetta tulisi muuttaa radikaalisti, jos siitä haluaisi yleiskäyttöisemmän erilaisiin musiikillisiin tilanteisiin.

## **6.4 Prototyypin arviointi pelisuunnittelullisesti**

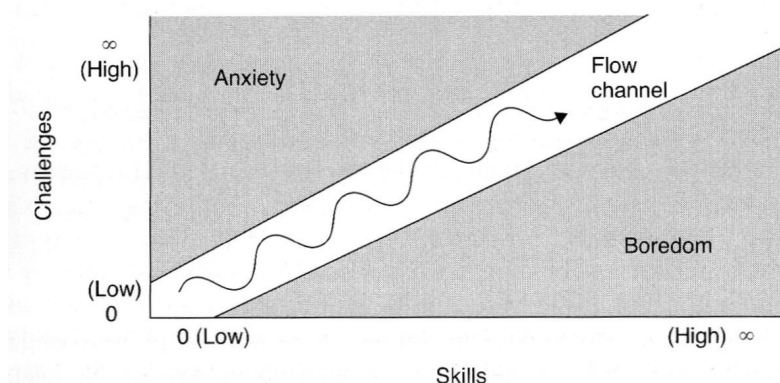
Salen ja Zimmerman (2004, s.32) sanovat, että mielekästä pelaamista syntyy, kun pelaajan toimintojen seuraukset ovat selkeitä ja integroitu laajempaan kontekstiin. Käytännössä tämä tarkoittaa sitä, että seuraukset ja integroituminen tulee kommunikoida pelaajalle selkeällä tavalla (Salen ja Zimmerman 2004, s.35). Pelisuunnittelussa puhutaan usein palautesilmukasta (feedback loop), joka liittyy läheisesti selkeisiin seurauksiin (Schell 2008, s.230). Schell mainitsee, että peleissä pelaajan tulisi nähdä seuraus annetulle syötteelle vähintään 0,1 sekunnissa (Schell 2008, s.231).

Prototyypissä pelaajan on helppo huomata tekemiensä toimintojen seuraukset. Pelaajan kosketus johtaa todennäköisesti jonkin asian liikkumiseen alle 0,1 sekunnissa. Mielestäni pelaaja oppii nopeasti tästä palautteesta, että voin liikutella asioita kosketuksella. Asioiden liikuttamisen integroituminen nähdään siitä, että pelaajan pallo kimpoaa liikuteltavista asioista. Monien asioiden liikuttelu on edelleen integroitu kentän läpäisemiseen. Pysin tekemään pelin mekaniikoista siinä määrin selkeitä, ettei pelin tarvitsisi antaa pelaajalle sanallisia ohjeita. Pallon jättämä vana ei ole vain visuaalinen efekti, vaan myös tärkeä osa pelaajalle annettua palautetta. Tämä liittyy myös Schellin mainitsemaan seikkaan teknologian käytöstä joko dekoratiivisesti tai perustavanlaatuisesti (Schell 2008, s.408).

Schell (2008, s.233) puhuu palautteen yhteydessä myös käyttöliittymän mehevyydestä (juiciness). Tämä perustuu järjestelmän näyttämään liikkeeseen, joka aiheutuu pelaajan toiminnasta (second-order motion). Jos tämän liike on helppoa hallita, tarjoaa vallan tunnetta ja on palkitsevaa, järjestelmää voidaan sanoa meheväksi. Asiantuntijuuteni mukaan fysiikan mallinnukseen perustuva peli antaa perustan mehevyydelle juuri fysiikan mallinnuksen tähden. Tämä toki vaatii sen, että pelaajalle myös näytetään ruudulla fysikaalista liikettä, toisin

sanoen ettei fyysiikkamoottori toimi vain näkymättömänä taustalla. Pallon kimpoilun hallitseminen liikuteltavilla asioilla ja siitä seuraava palaute tarjoaa mielestäni mehevyyttä. Pelaajaa myös palkitaan soitinten äänillä, kun pallo osuu asioihin. Mehevyyttä voi lisätä animaatioiden kautta annettavalla palautteella. Tällöin animaatiot itsessään palkitsevat pelaajaa. Esimerkiksi musiikkinäkymässä päällä oleva nappula pomppaa esiin, kun nappulaan liitetty ääni soitetaan. Laajemmassa kontekstissa musiikkikomponentti palkitsee pelaajaa mielenkiintoiselta kuulostavalla musiikilla. Tällöin yksittäiset pelaajan etenemisestä seuranneet nuotit tai rytmisoitinten iskut muodostavat kokonaisuutena musiikkia.

Pelin etenemisessä kentästä toiseen olen pyrkinyt noudattamaan Schellin mainitsemaa flow-kanavassa pysymistä (Schell 2008, s.121). Flow-tilaan pääsy vaatii tasapainoa pelaajan taitojen ja pelin tarjoaman haasteen välillä. Schell esittää, että pelaajan kokemuksen kannalta on todennäköisesti mielenkiintoisempaa, jos taitojen ja haasteen tasapainoa kiristetään ja löysätään aika ajoin kuvion 22 mukaan kuin että flow-kanavassa edettäisiin lineaarisesti. Toki lineaarinen eteneminen on parempi kuin pelaajan tylsistyminen tai turhautuminen. Mielestäni flow-tilassa pysyttelevän pelikokemuksen muodostaminen vaatii runsaasti pelitestaamista ja pelin tasapainottamista iteratiivisesti. Prototyypin kenttien muodostaminen on satunnaisuuteen perustuvaa, mikä luo haasteen optimaalisen vaikeustason tarjoamiseen. Tästä johtuen myös miellyttävän pelikokemuksen muodostamiseen tulee haastetta.



Kuvio 22. Flow: haasteen ja taitojen tasapaino

Prototyyppiin tein juurikin aaltoilevan ylöspäin nousevan kaaren (kuvio 22), joka palauttaa x-arvolla pelin nykyisen haastearvon. X-arvoa nostetaan aina, kun pelaaja läpäisee kentän (Schell 2008, s.177). Käyrän haastearvon perusteella voidaan optimitilanteessa muodostaa



kenttiä, jotka muuttuvat haastavammiksi pitkän ajan kuluessa samalla kun haastetta löysätään tai tiukennetaan lyhyellä aikavälillä. Tässä kuitenkin on tärkeintä se, kokeeko pelaaja kenttien etenemisen tällä tavalla. Pelikokemusta on hyvin vaikea arvioida muuten kuin testaamisen, pelaajien antaman palautteen avulla tai pelimetriikoiden analysoinnilla (Schell 2008, s.178). Prototyypissä haastearvo vaikuttaa liikuteltavien asioiden ja esteiden määrään. Lähtökohtaisesti pelin haastetaso on melko alhainen. Tämä on mielestäni otollisempi lähtökohta kuin että peli olisi alusta alkaen liian vaikea tai turhauttava. Asiantuntijuuteni mukaan kasuaalipelille on tärkeää, että peli vaatii oikeanlaista ja riittävän matalaa keskittymistasoa. Ennen pelitestausta keskittymistason arviointi on kuitenkin itselleni kokemukseen ja intuitioon perustuvaa.

Prototyyppiä kehitettäessä pohdin oikeaa tasapainoa pulmapelillisten ja luovan musiikkipelin elementtien välillä. Tämä on osaltaan myös markkinoinnillinenkin asia. Tulisiko pelin olla enemmän lelumainen, jolloin pelaaja voisi keskittyä musiikin tekemiseen? Luovat musiikkipelit ovat tyypillisesti tällaisia. Vai tulisiko pelin olla täysiverinen pulmapeli kehittyvine haasteineen? Johtavaksi ajatukseksi nousi se, että peli vetoaisi mahdollisimman laajaan yleisöön. Tällöin musiikin luominen olisi ikään kuin löydettävä tai paljastuva asia pelissä. Mietin tässä Schellin ajatusta transformaalisista peleistä (Schell 2008, s.452). Schell uskoo, että pelit muuttavat pelaajiaan joko parempaan tai huonompaan suuntaan. Kuitenkin jos pelillä halutaan olevan laajaa vaikutusta, tulee suuren joukon ihmisiä haluta pelata peliä (Schell 2008, s.456). Tällöin ei auta, vaikka kehittäjällä olisi hyvät tarkoitusperät, jos kehitetty peli ei vedä ihmisiä puoleensa.

Säveltämistä voidaan pitää jonakin, joka kuuluu vain muusikoille ja tuottajille. Musiikinluomiselementtiä voidaan ehkäpä pitää hieman piilossa (Schell 2008, s.456). Tämä siis siksi että monet eivät ehkä kajoa peliin, jos sitä markkinoidaan sävellyspelinä. Mutta jos peliä markkinoidaankin kasuaalina pulmapelinä, saattaa useammat pelaajat haluta pelata peliä, jos peli on koukuttava itsessään. Musiikinluonti voi sitten sivussa tulla mielenkiintoisena yllätyksenä pelaajille, mikä olikin pelinkehittäjän salainen agenda (Schell 2008, s.456).

## 6.5 Systemaattinen pelitestausta ja RGT-menetelmä

Viimeisenä testausvaiheena toteutin Repertory Grid Technique -menetelmään (RGT) perustuvan käyttäjätestin viidelle koehenkilölle. RGT-menetelmän vahvuutena on, että tutkittavat henkilöt pääsevät käyttämään arviointiprosessissa omia henkilökohtaisia konstruktioita (Hassenzahl ja Wessler 2000). Näistä konstruktioista muodostetaan kaksinapaisia pareja, jotta konstruktioilla voidaan tehdä arviointia tietyllä asteikolla. Bellotti ym. (2009) käyttivät RGT-menetelmää digitaalisten pelien arviointiin. Heidän tavoitteenaan oli parantaa pelien opetuksellisuutta. Bellotti ym. osoittivat, että heidän kehittämä opetuksellisia elementtejä sisältävä peli oli pelaajien mielestä miellyttävä samalla tapaa kuin kaupalliset videopelit (Bellotti ym. 2009, s.16).

Tässä työssä tehdyssä käyttäjätestissä verrattiin kehitettyä prototyyppiä julkaistuihin mobiilipeleihin, joita koehenkilöt olivat aiemmin tykänneet pelata. Tavoitteena oli löytää prototyypistä paranneltavia kohtia jatkokehitystä varten. Toisena tavoitteena oli selvittää, pitävätkö käyttäjät musiikinluontipeliä luontevana kokonaisuutena verrattuna julkaistuihin mobiilipeleihin.

Käytin tässä työssä samankaltaista tutkimustilanteen rakennetta kuin Bellotti ym. käyttivät omassa työssään (Bellotti ym. 2009, s.10). Tutkimustilanteen vaiheet olivat seuraavanlaiset:

1. *Pelien valinta*: Käyttäjää pyydettiin mainitsemaan kuusi mobiilipeliä, joita he olivat tykänneet pelata. Näistä muodostui kaksi kolmikkoa pelejä.
2. *Konstruktioiden muodostaminen*: Käyttäjä kertoi kustakin kolmikosta piirteitä, jotka yhdistivät kahta peliä, mutta erottivat ne kolmannesta. Tästä syntyi kaksisuuntaisia konstruktioita (esim. aivojen käyttöä vaativa-sattumanvarainen, siisti äänimaailmatylsät äänet, yhteistoiminta-yksinpeli).
3. *Pelien arviointi*: Käyttäjä arvioi kaikki kuusi peliä vaiheessa 2 saaduilla konstruktioilla asteikolla 1-7.
4. *Musiikinluontipelin testaus*: Käyttäjä testasi tässä työssä kehitettyä musiikinluontipeliä Free Golf.
5. *Musiikinluontipelin arviointi*: Käyttäjä arvioi musiikinluontipelin vaiheessa 2 saaduilla konstruktioilla asteikolla 1-7.

Kokeeseen osallistuneet viisi henkilöä olivat 20–30-vuotiaita miehiä. Kaikilla osallistujilla oli asiantuntemusta musiikin soittamisesta tai tekemisestä, ohjelmisto- tai pelinkehityksestä tai yrittäjyydestä. Kukin koehenkilö mainitsi kahdeksan erilaista konstruktiota, joilla koehenkilö arvioi pelit. Kokeen jälkeen koehenkilöille annettiin mahdollisuus myös vapaan palautteen antamiseen.

Tutkimuksesta saadut tulokset tarjosivat prototyypille runsaasti kehityskohteita ja pohdittavaa jatkokehitystä varten. RGT-menetelmän tuloksia on esitetty kuviossa 23. Kuvioon kaikki arvosanat on järjestetty niin, että korkea palkki tarkoittaa korkeaa arvosanaa tietyssä ominaisuudessa tai piirteessä. Kukin kuvion ominaisuus on yksittäisen koehenkilön omin sanoin kuvailema asia, joka tuli mainituksi RGT-menetelmän aikana. Keskiarvo tarkoittaa yhden henkilön mainitseman kuuden pelin saamaa keskiarvoa. Kuviossa vasemmalla puolella on esitetty piirteitä, jotka ovat selkeimpiä jatkokehityksen kohteita. Oikealla puolella kuviota on mainittu piirteitä, joista prototyyppi sai keskiarvoa parempia arvosanoja. Tulokset ovat suuntaa-antavia ja yleistettävämpiin tuloksiin tarvittaisiin suurempi testaajamäärä.

Koehenkilöillä oli paljon omia näkökulmia peleihin ja pelaamiseen. Eräs piti fysiikan mallinnukseen pohjautuvista peleistä, toinen korosti yhteistoimintaa. Eräs koehenkilö kiinnitti peliin yrittäjämäisesti huomiota ja mietti, kuinka peliä tulisi parantaa jotta siitä saisi koukuttavamman ja kuinka monetisaatio tulisi tehdä. Muutamit koehenkilöt kiinnittivät samankaltaisiin asioihin huomiota, kuten graafiseen tyyliin, kontrolleihin ja äänimaailmaan. Äänimaailmasta saadut tulokset olivat eniten samansuuntaisia ja kiittäviä. Kontrolleista ja graafisen tyylin vertautumisesta julkaistuihin peleihin oli enemmän erimielisyyttä. Toisistaan poikkeavat arviot saattoivat johtua koehenkilön taustasta. Esimerkiksi pelinkehitystaustainen saattoi kiinnittää kriittisemmin huomiota grafiikkaan ja kontrolleihin.

Selkeimpiä kehityksen kohteita pelissä on progressio, joka toistui eri koehenkilöillä eri muodoissa. Progressio on myös asiantuntijuuteni mukaan resursseja ja vähittäin etenevää kehitystä ja testausta vaativa prosessi (luku 3.2). Pelinkehitystaustainen koehenkilö mainitsi avoimessa palautteessa musiikintekemisen ja pelin yhdistämisen vaikeaksi suunnitteluongelmaksi. Hän viittasi samalla siihen, että vaikeus liittyy nimenomaan pelin ja musiikintekemisen yhdistämisen yhteen ja samaan käyttöliittymään. Tällä hetkellä prototyypissä on kaksi erillistä käyttöliittymää, joilla on yhteys keskenään. Koehenkilö mainitsi, että peli ja musiikki-

komponentti voisivat toimia erillisinä sovelluksina, mutta mielti, ovatko ne yhdessä järkevä yhdistelmä. Koehenkilö mainitsi pelin ja musiikin luomisen yhdistämisen yhdeksi kokonaisuudeksi ja käyttöliittymäksi vaikeaksi pelisuunnitteluongelmaksi.

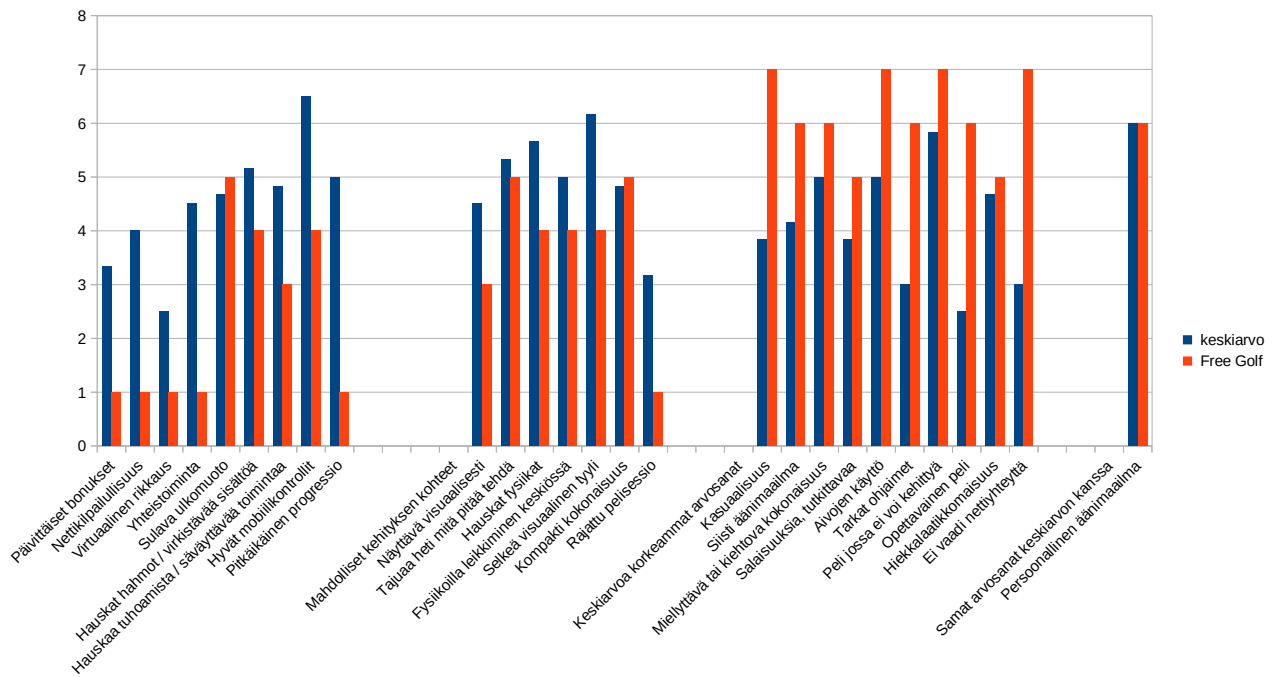
Toinen selkeä parannuskohde on edelleen pelin ja musiikkikomponentin välisen mappauksen selkeyttäminen. Jotkin koehenkilöt kaipasivat selvennystä siihen, kuinka musiikki muodostuu pelatessa. Kahdella koehenkilöllä nousi myös tarve vaikuttaa musiikinmuodostumiseen heidän haluamallaan tavalla pelin kautta. Tämä ei heidän mielestään onnistunut. Pelin kautta muodostuva musiikki onkin hyvin satunnaista, kuten mainitsin luvussa 6.1. Tämän seikan parantaminen saattaisi vaatia kokonaan pelimekaniikkojen vaihtamista.

Prototyyppi sai tietyistä piirteistä myös keskiarvoa parempia arvioita. Prototyyppi sai hyviä arvosanoja muun muassa “miellyttävästä tai kiehtovasta kokonaisuudesta”, tarkoista ohjaimista, “siististä” äänimaailmasta ja tutkittavuudesta. Nämä arvosanat viittaavat siihen, että ensinnäkin musiikinluontipeli voi vetää vertoja julkaistuille mobiilipeleille. Musiikilliset piirteet voivat tuoda uutuusarvoa pelikokemukseen muun muassa instrumentin mahdollisuuksien tutkimisella. Prototyyppiä voidaan pitää kiehtovana kokonaisuutena vähintään joidenkin pelaajien mielestä. Hiekkalaatikkomaiseen interaktioon mieltyneet pelaajat voivat pitää musiikin luomisen tarjoamasta kokeilusta ja leikittelystä.

## **6.6 Prototyypin parantelu**

Tässä luvussa esittelen prototyypin parannussuunnitelmia RGT-menetelmällä saatuihin tuloksiin ja avoimeen palautteeseen perustuen. Palautteen perusteella prototyyppiin tulisi saada konkreettisia tavoitteita eli laajemmin progressiota. Tavoitteita olisi mahdollista saada lisäämällä peliin tehtäviä ja keräiltävää. Pelin olisi mahdollista esitellä tehtävät heti aloitusruudun jälkeen kuten kuviossa 24.

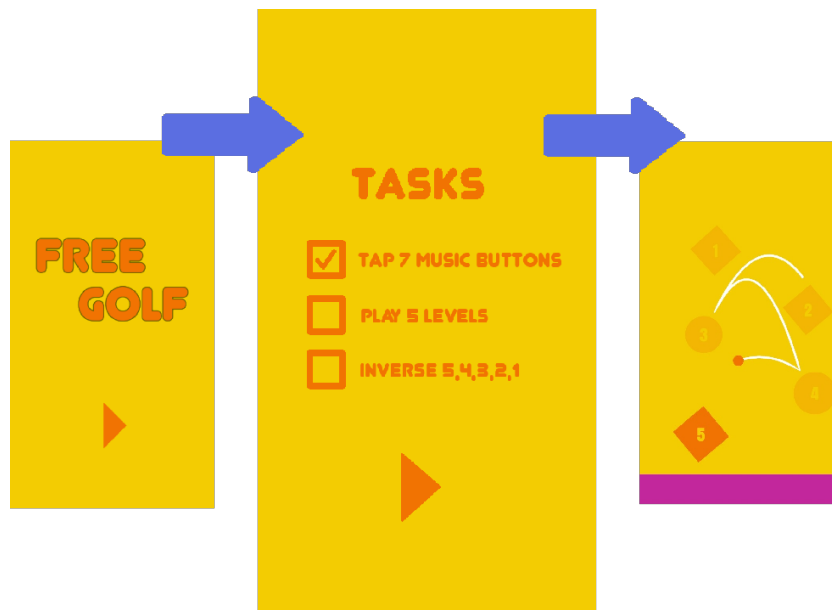
Pienten tehtävien lisäksi eräs testaajista ehdotti, että instrumentin nuoteista tulisi tehdä keräiltäviä. Tällöin pelaajalla ei olisi heti alussa mahdollista lisätä nuotteja rajattomasti vaan hänellä olisi käytössä vain muutama nuotti. Sitten tavoitteiden suorittamisella pystyttäisiin avaamaan enemmän nuotteja instrumenttiin. Nuottien senhetkinen määrä voitaisiin ilmaista kuten kuviossa 25.



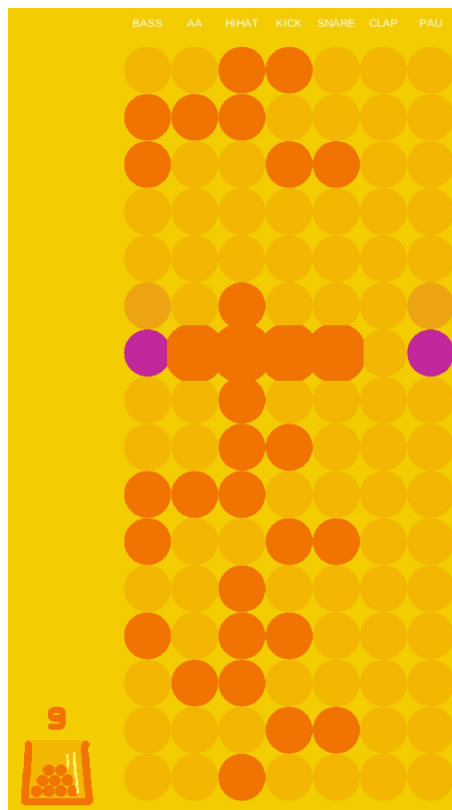
Kuvio 23. Arvosanoja ja kehityksen kohteita

Tehtävistä ja keräiltävistä asioista olisi näin mahdollista muodostaa peliluoppi. Pelaaja suorittaa tehtäviä, mikä avaa instrumenttia enemmän käyttöön. Pelaaja siirtyy tekemään musiikkia, mutta hetken päästä nuotit loppuvat kesken. Nuottien loppuminen motivoi siirtymään takaisin suorittamaan tehtäviä.

Tehtävien suorittamista voidaan rajata niin, että kaikkia pelissä olevia tehtäviä ei ole mahdollista läpäistä yhdellä pitkällä pelikerralla. Uusia tehtäviä voitaisiin avata vaikkapa muutaman tunnin kuluttua pelisessiosta. Tämä toisi peliin myös muista mobiilipeleistä tuttua retentiota.



Kuvio 24. Siirtyminen tehtävänäkymään



Kuvio 25. Jäljellä olevien nuottien määrä ilmaistuna vasemmassa alakulmassa.

## 7 Pohdinta ja johtopäätökset

Luovat musiikkipelit antavat pelaajalle mahdollisuuden musiikin soittamiseen ja luomiseen. Kuitenkin rytmipelit ovat saaneet paljon enemmän näkyvyyttä ja tulleet tunnetuksi suurelle yleisölle. Rytmipelit yleensä tarjoavat selkeät lyhyen ja pitkän aikavälin tavoitteet ja mahdollisuuden pelaajalle kartuttaa taitojaan. Näitä ominaisuuksia ei useinkaan löydy luovista musiikkipeleistä tai ominaisuudet jäävät vaillinaisiksi.

Tutkin ML-pelin kehitysprosessia konstruktivisen tutkimuksen keinoin. Konstruktiona syntyi peliprototyyppi, jolla voidaan luoda musiikkia pelaamisen aikana.<sup>1</sup> Saadulla suunnittelutietämyksellä voidaan parantaa olemassa olevia luovia musiikkipelejä julkaistujen mobiilipelien suuntaan. Suunnittelutietämys antaa tietoa uusille luovien musiikkipelien projekteille varsinkin, jos niitä halutaan kehittää tavoitteellisen pelin suuntaan. Tutkimus esittää ML-pelin uutena käsitteenä musiikkipelien olemassa olevaan teoriaan.

Musiikinluontipelinkehitys on monialainen prosessi, joka liittyy muun muassa musiikin teoriaan, digitaalisen instrumentin suunnitteluun, pelisuunnitteluun ja -kehitykseen, ohjelmistoarkkitehtuurin suunnitteluun ja käyttäjäkokemukseen. Usein luovat musiikkipelit ovat leluja, mutta tässä tutkielmassa konstruktioista haluttiin rakentaa tavoitteellinen peli (määritelmä 1). Tämä luo lisähaastetta siihen, kuinka käyttäjäkokemuksesta saadaan luonteva kokonaisuus pelin ja musiikin luomisen välillä. Luovat musiikkipelit eivät ole juurikaan nousset suuren yleisön tietoisuuteen ehkäpä taiteellisen ja eriskummallisen luonteensa vuoksi. Tämä kuitenkin antaa lähtökohdan tavoitteelle nivoa musiikin luominen yhteen kasuaaliin ja tavoitteelliseen mobiilipeliin. Kouvuttava mobiilipeli voi kiinnostaa monia, ja samalla peli houkuttelisi ihmisiä myös kokeilemaan helppoa musiikin luomista. Jatkokehityksessä olisi mahdollista ottaa mukaan musiikin teoreettisia asioita kuten sointumerkintöjä. Käyttöliittymä voitaisiin edelleen säilyttää helppokäyttöisenä, jolloin käyttäjältä ei vaadita aiempaa tuntemusta musiikin teoriasta. Käyttäjä voisi tällöin itse valita, haluaako hän syventyä pidemmälle meneviin ominaisuuksiin. Säveltämisen pelillistäminen on mahdollinen kohde jatkotutkimukselle esimerkiksi säveltämisen opettamista ajatellen.

---

1. Prototyypin latauslinkki: <https://play.google.com/apps/testing/mintshire.freegolf>

Prototyypissä musiikin kehittyminen on generatiivista pelaamisen aikana, mutta musiikki on täysin käyttäjän muokattavissa musiikkityökalun puolelta. Jatkokehityksessä tätä voitaisiin rajata keräiltävillä nuoteilla, joita pelaaja voisi saada suoritettuaan tehtäviä. Tässä tutkielmassa en keskittynyt musiikin algoritmiseen generointiin vaan enemmän käyttäjäkokemukselliseen ja osallistavaan musiikin luomiseen. Kuitenkin generatiivisen luonteensa vuoksi musiikista tulee helposti satunnaisen kuuloista. Satunnaisuutta estin sillä, että tein vain melodian generoituvaksi pelin kautta. Puolestaan rytmin luominen on kokonaan käyttäjän vastuulla. Vaihtoehtoisesti peli voi ohjata pelaajaa tekemään tietyn kehittäjän tallentaman rytmin.

Harvinaisempana prototyypin ominaisuutena voidaan pitää helppokäyttöistä sointuraitaa. Sointuraita soittaa länsimaiselle musiikille tyypillistä sointutaustaa, jonka mukaan valitaan sopivia melodian ja basson säveliä. Toki sävelletäessä musiikkia melodia voidaan tehdä ensin ja vasta tämän jälkeen soinnuttaa halutunlaisesti, mutta prototyypissä soinnut ratkaisevat melodian. Sointuraita antaa rakennetta luotuun musiikkiin. Joskin voidaan sanoa, että soinnut rajoittavat instrumentin musiikillista antia länsimaiseen suuntaan. Helppokäyttöinen sointuraita on asiantuntijuuteni mukaan vähemmän nähty ominaisuus ottaen huomioon myös varsinaiset musiikintuottamiseen tehdyt ohjelmistot. Tässä kohdin mielenkiintoiseksi muodostuu sointukiertoihin liittyvä ominaisuus, joka osaisi ehdottaa sopivia tai erikoisempiakin sointukiertoja käyttäjälle.

Musiikkiteknologiaprojekti-kurssilla teimme yhteistoiminnallisen ryhmäsävellyspelin Compose It (CI). CI:n sävellysosuus pohjautui tässä työssä tehtyyn musiikkikomponenttiin. CI:ssä pelaajat säveltävät lyhyitä luppeja ja äänestävät parhaita luppeja jatkoon. Äänestyksissä menestyneet pelaajat saavat pisteitä ja eniten pisteitä kerännyt voittaa pelin. Tällainen ryhmässä tehtävä pelillinen säveltäminen on toiminut hyvin omien pelitestaustemme mukaan varsin eritaustaisilla ja eri-ikäisillä pelaajilla. CI:ssä itse säveltämisestä ja instrumentista tulee tärkeä osa peliä. Samalla peli ja säveltäminen sulautuvat hyvin yhdeksi pelikokemukseksi. Tällainen peli vie pelikokemusta enemmän yhteistoiminnalliseen suuntaan yksinpelin sijaan. Tämän kokemuksen mukaan yksinpeli vaatii peliin selkeitä pitkäaikaisia tavoitteita, mutta moninpelissä sosiaalisuus korvaa osittain progression puutetta.

Tässä tutkimuksessa kehitettiin musiikinluontipelin prototyyppi, joka osoitti että musiikin



luomiseen nivoutuva pelikokemus voi olla pelaajien mielestä mielenkiintoinen ja opettava. ML-pelissä oleva instrumentti voi sisältää samanlaisia piirteitä kuin muut digitaaliset instrumentit (kuviot 20 ja 21). Tämän työn tulosten yleistettävyyttä voidaan parantaa jatkotutkimuksella ja laajemmalla pelitestauksella. Tutkielman rajallisuus näkyy myös kehitetyssä ML-pelissä, joka jäi prototyypivaiheeseen. ML-pelin tulee tarjota tarpeeksi sävyttävää toimintaa ja virkistävää sisältöä esimerkiksi hauskojen hahmojen muodossa. ML-pelin tulee sisältää progressiota eli selkeitä pitkän aikavälin tavoitteita, että ML-peli vertautuisi paremmin julkaistuihin mobiilipeleihin.

RGT-menetelmän tuloksiin perustuen ehdotin musiikinluontipeliin progression parantelua. Monesti progressio on juuri se osa, joka puuttuu luovista musiikkipeleistä (luku 3.2). Pelitestauksen kautta saadun palautteen perusteella tein suunnitelman, jolla prototyyppiin voidaan rakentaa progressiota (luku 6.6). Tämän työn suunnittelutietämyksen ehdottamaa progressiota on mahdollista tuoda myös olemassa oleviin luoviin musiikkipeleihin ainakin yleisellä tasolla. Progression parantelun käytännön ratkaisut voivat vaihdella pelistä toiseen. Yleisempien ratkaisumallien kehittämiseen tarvittaisiin lisätutkimusta ja pelitestausta. Progression parantelu ja retentio voivat johtaa pidempiin pelikokemuksiin, joita julkaistut mobiilipeleillä usein tarjoavat. Tällöin luovat musiikkipelit voivat saada tunnettuutta myös suurten yleisöjen joukossa. Musiikinluontipelin kautta musiikinluontia on mahdollista tuoda yleisöille, jotka eivät ole aiemmin harrastaneet säveltämistä.

## Lähteet

- Aken, Joan E van. 2004. "Management research based on the paradigm of the design sciences: the quest for field-tested and grounded technological rules". *Journal of management studies* 41 (2): 219–246.
- Armitage, Jack, Fabio Morreale, Andrew McPherson ym. 2017. "'The finer the musician, the smaller the details": NIMEcraft under the microscope".
- Bellotti, F., R. Berta, A. De Gloria ja L. Primavera. 2009. "Enhancing the Educational Value of Video Games". *Comput. Entertain.* (New York, NY, USA) 7, numero 2 (kesäkuu): 23:1–23:18. ISSN: 1544-3574. doi:10.1145/1541895.1541903.
- Birnbaum, David, Rebecca Fiebrink, Joseph Malloch ja Marcelo M Wanderley. 2005. "Towards a dimension space for musical devices". Teoksessa *Proceedings of the 2005 conference on New interfaces for musical expression*, 192–195. National University of Singapore.
- Bott, Jared N., James G. Crowley ja Joseph J. LaViola Jr. 2009. "Exploring 3D Gestural Interfaces for Music Creation in Video Games". Teoksessa *Proceedings of the 4th International Conference on Foundations of Digital Games*, 18–25. FDG '09. Orlando, Florida: ACM. ISBN: 978-1-60558-437-9. doi:10.1145/1536513.1536527.
- Cohen, H Floris. 1987. "Simon Stevin's equal division of the octave". *Annals of science* 44 (5): 471–488.
- Collins, Karen. 2008. *Game sound : an introduction to the history, theory, and practice of video game music and sound design*. Cambridge, MA: MIT Press.
- Deterding, Sebastian, Dan Dixon, Rilla Khaled ja Lennart Nacke. 2011. "From Game Design Elements to Gamefulness: Defining "Gamification"". Teoksessa *Proceedings of the 15th International Academic MindTrek Conference: Envisioning Future Media Environments*, 9–15. MindTrek '11. Tampere, Finland: ACM. ISBN: 978-1-4503-0816-8. doi:10.1145/2181037.2181040.

- Friberg, Johnny, ja Dan Gärdenfors. 2004. "Audio Games: New Perspectives on Game Audio". Teoksessa *Proceedings of the 2004 ACM SIGCHI International Conference on Advances in Computer Entertainment Technology*, 148–154. ACE '04. Singapore: ACM. ISBN: 1-58113-882-2. doi:10.1145/1067343.1067361.
- Gregor, Shirley, ja Alan R Hevner. 2013. "Positioning and presenting design science research for maximum impact." *MIS quarterly* 37 (2).
- Hassenzahl, Marc, ja Rainer Wessler. 2000. "Capturing Design Space From a User Perspective: The Repertory Grid Technique Revisited." *International Journal of Human-Computer Interaction* 12 (3/4): 441–459. ISSN: 10447318.
- Herber, Norbert. 2006. "The Composition-Instrument: musical emergence and interaction". Teoksessa *Proceedings of the Audio Mostly Conference 2006*, 31–36.
- Ivanova, Tamara E. 2010. *Music : Composition, Interpretation and Effects*. Focus on Civilizations and Cultures– Music. Nova Science Publishers, Inc. ISBN: 9781608761708.
- Järvinen, Pertti. 2004. *Tutkimustyön metodeista*. [Uud. p.] Toimittanut Annikki Järvinen. Tampere: Opinpajan kirja.
- Kaliakatsos-Papakostas, Maximos, Marcelo Queiroz, Costas Tsougras ja Emilios Cambouropoulos. 2017. "Conceptual Blending of Harmonic Spaces for Creative Melodic Harmonisation". *Journal of New Music Research* 46 (4): 305–328.
- Labro, Eva, ja Tero-Seppo Tuomela. 2003. "On bringing more action into management accounting research: process considerations based on two constructive case studies". *European Accounting Review* 12 (3): 409–442.
- Liebe, Michael. 2013. "Interactivity and music in computer games". *Music and game*: 41–62.
- Limited, VGChartz. *Game Database*. <http://www.vgchartz.com/gamedb/>. Haettu 28.5.2018.
- Luukkanen, Timo. 2016. "Sattumallako sijansa äänitetuotannossa?" Tutkielma, Helsingin yliopisto, Humanistinen tiedekunta, Filosofian, historian, kulttuurin ja taiteiden tutkimuksen laitos.

- Magnusson, Thor. 2010. "An epistemic dimension space for musical devices". Teoksessa *Proceedings of the 2010 conference on New interfaces for musical expression*, 2010:43–46. University of Technology Sydney.
- March, Salvatore T, ja Gerald F Smith. 1995. "Design and natural science research on information technology". *Decision support systems* 15 (4): 251–266.
- O'modhrain, Sile. 2011. "A framework for the evaluation of digital musical instruments". *Computer Music Journal* 35 (1): 28–42.
- Overholt, Dan. 2009. "The musical interface technology design space". *Organised Sound* 14 (2): 217–226.
- Partti, kirjoittaja, Heidi. 2016. *Säveltäjäyden jäljillä : musiikintekijät tulevaisuuden koulussa*. Sibelius-akatemia julkaisuja. Kirjan pohjana on Säveltäjän Tekijänoikeustoimisto Teoston lukuvuonna 2013-14 toteuttama Koulujen BiisiPumppu-projekti. Helsinki: Taideyliopiston Sibelius-Akatemia. <http://urn.fi/URN:ISBN:978-952-329-057-0>.
- Pease, Emma. *CSLI Calendar Of Public Events. Stanford Center for the Study of Language and Information. 14 May 1997*. <https://web.archive.org/web/20110716051829/http://www-csli.stanford.edu/Archive/calendar/1996-97/msg00029.html>. Haettu 28.5.2018.
- Pichlmair, Martin, ja Fares Kayali. 2007. "Levels of sound: On the principles of interactivity in music video games". Teoksessa *Proceedings of the Digital Games Research Association 2007 Conference "Situating play"*. Citeseer.
- Rautiainen, Antti, Kari Sippola ja Toni Mättö. 2017. "Perspectives on relevance: The relevance test in the constructive research approach". *Management Accounting Research* 34 (Supplement C): 19–29. ISSN: 1044-5005. doi:<https://doi.org/10.1016/j.mar.2016.07.001>. <http://www.sciencedirect.com/science/article/pii/S1044500516300233>.
- Salen, K., ja E. Zimmerman. 2004. *Rules of Play: Game Design Fundamentals*. Books 24x7 IT PRO. Books24x7.com. ISBN: 9780262240451.

Santi, Marina. 2010. *Improvisation : Between Technique and Spontaneity*. Cambridge Scholars Publishing. ISBN: 9781443818544.

Schell, Jesse. 2008. *The Art of Game Design: A Book of Lenses*. Morgan Kaufmann. Elsevier/Morgan Kaufmann. ISBN: 9780123694966.

Schultz, Peter. 2008. *Music theory in music games*. na.

Silfer, Kyle. *Applied Ludology. Art games and game art. ALIBI V.16 NO.28. JULY 12-18, 2007*. <http://alibi.com/feature/19721/Applied-Ludology.html>. Haettu 28.5.2018.

Sweetser, Penelope, ja Peta Wyeth. 2005. "GameFlow: A Model for Evaluating Player Enjoyment in Games". *Comput. Entertain.* (New York, NY, USA) 3, numero 3 (heinäkuu): 3–3. ISSN: 1544-3574. doi:10.1145/1077246.1077253.

Unity Technologies. 2018. *AudioSource.PlayScheduled*. <https://docs.unity3d.com/ScriptReference/AudioSource.PlayScheduled.html>. Haettu 28.5.2018.

# Liitteet

## A RGT-menetelmän vertailupelit

- 2048
- Alto's Adventure
- Angry Birds
- Bad Lands Brawl
- Bottle Flip Challenge
- Brain It On! - Physics Puzzles
- Burning Sand 3 (muita pelattuja: Sand! ja The Elements: Sand Game)
- Cell Lab: Evolution Sandbox
- Clash of Clans
- Clash Royale
- Crossy Road
- Fire Emblem Heroes
- Geometry Dash
- Hill Climb Racing
- Orbit - Playing with Gravity
- Pokemon Go
- Polytopia
- PUBG
- Rider
- Sanajahti
- Sanapala
- slither.io
- Super Mario Run
- Vector Pinball
- Where's My Water?

## B RGT-menetelmän konstruktiot

Käyttäjien mielestä positiiviset asiat on lihavoitu. Jotkin sanaparit olivat neutraaleja toisiinsa nähden.

- **aivojen käyttö** - sattumanvarainen
- alunperin raskas suorituskyvyllle - kevyt
- fysiikoita ei hyödynnetä, tylsät fysiikat - **fysiikoilla leikkiminen keskiössä**
- haastavuus - casual
- **hauskaa tuhoamista** - ei sävyyttävää toimintaa
- **hauskat hahmot** - ei virkistävää sisältöä
- **hiekkalaatikkomaisuus** - rajattu toiminta, näkymättömiä seiniä
- karkkigrafiikka - realistinen grafiikka
- kevyt tuotteistus - vahva tuotteistus
- **kompakti kokonaisuus** - sekava kokonaisuus
- **liikutaan fyysisesti** - ei liikuta
- **miellyttävä/kiehtova kokonaisuus** - yksitoikkoinen, tylsä
- **missä vaan pelattava** - vaatii nettiyhteyden
- monimutkainen - yksinkertainen
- **nettikilpailullisuus** - ei netti-high scorea
- **näyttävä visuaalisesti** - pelkistetty
- peli, jossa ei voi kehittyä - **peli, jossa on kehittymismahdollisuuksia**
- peli, josta ei ole käytännönläheistä hyötyä - **opettava peli**
- **persoonallinen äänimaailma, istuu kokonaisuuteen** - ei persoonallista äänimaailmaa
- perustuu yhteen näkymään - monipuolisuus
- pisteiden kerääminen - ongelmanratkaisu
- **pitkäikäinen progressio** - yksinkertainen high score
- pulmanratkaisu - strategia
- **päivittäiset bonukset** - ei päivittäisiä bonuksia
- **rajattu pelisessio** - peli ei rajaa pelisessiota
- **selkeä visuaalinen tyyli** - sillisalaatti, ei persoonallisuutta

- sota - rauha
- staattiset fysiikat, luonnon käyttäytyminen - **hauskat fysiikat**
- tajuaa heti, mitä pitää tehdä - vaikea, kärsivällisyyttä vaativa
- **tarkat ohjaimet** - huono käyttöliittymä
- todella paljon yksityiskohtia - yksinkertainen grafiikka
- tylsät äänet - **siisti äänimaailma**
- tönkkö - **sulava ulkomuoto**
- uhkapelimäinen gatcha - **pelaaja tietää, mitä saa käyttäessään rahaa**
- **vastaa realismia** - ei vastaa realismia
- virtuaalinapit, monen sormen kontrollit - **hyvät mobiilikontrollit**
- **virtuaalinen rikkaus** - ei keräiltävää
- yhteistoiminta - yksinpeli
- yhteys reaali maailmaan vahva - yhteys reaali maailmaan heikko
- yllätyksättömyys - **salaisuuksia, piilotettuja asioita, tutkittavaa**

## C Lähdekoodi soveltuvin osin

Linkki versionhallintaan:

[git@bitbucket.org:speci/musprotoforgradu.git](https://git@bitbucket.org:speci/musprotoforgradu.git)