

Niko Haapakoski

**ADAPTIIVISEN ÄÄNIMAILMAN SUUNNITTELU JA TOTEUTUS  
UNITY-PELIMOOTTORILLA**

# **ADAPTIIVISEN ÄÄNIMAILMAN SUUNNITTELU JA TOTEUTUS UNITY-PELIMOOTTORILLA**

Niko Haapakoski  
Adaptiivisen äänimaailman suunnittelu  
ja toteutus Unity-pelimoottorilla  
Kevät 2021  
Tietojenkäsittelyn koulutusohjelma  
Oulun ammattikorkeakoulu

## TIIVISTELMÄ

Oulun ammattikorkeakoulu  
Tietojenkäsittelyn tutkinto-ohjelma

---

Tekijä(t): Niko Haapakoski

Opinnäytetyön nimi: Adaptiivisen äänimaailman suunnittelu ja toteutus Unity-pelimootorilla

Työn ohjaaja(t): Matti Viitala

Työn valmistusluku- ja -vuosi: Kevät 2021

Sivumäärä: 44

---

Tämän opinnäytetyön tarkoituksena oli suunnitella ja toteuttaa Unity-pelimootorilla peli, joka käyttää adaptiivista musiikkia niin perinteisiksi muodostuneilla menetelmillä, kuin myös hieman luovempia ratkaisuja hyödyntäen. Peli on Bullet Hell -henkinen kaksiulotteinen räiskintä. Pelin musiikki toteutettiin FL Studio -ohjelmistossa. Raportin teoriaosuudessa käydään läpi pelimusiikin historiaa, säveltäjän työkaluja, Unityn audiokomponentteja ja äänimaailman tekniseen suunnitteluun liittyviä asioita. Käytännön osuus on jaettu kahteen osaan, jossa ensimmäinen koostuu pelin äänimaailman suunnittelu- ja sävellystyöstä, ja jälkimmäinen sen toteuttamisesta Unityn sisällä. Työ painottuu musiikin ja äänien toteutukseen.

Lopputuloksena saatiin aikaiseksi toimiva peli, joka demonstroi adaptiivisen musiikin mahdollisuuksia. Opinnäytetyön tekemisessä kehittyttiin äänen käsittelyssä Unityssa ja äänien suunnittelutyössä varsinkin pelien saralla.

---

Asiasanat: pelimusiikki, pelisuunnittelu, säveltäminen

## ABSTRACT

Oulu University of Applied Sciences  
Degree Programme in Business Information Systems

---

Author(s): Niko Haapakoski

Title of thesis: Designing and Implementing Adaptive Sounds in Unity Game Engine

Supervisor(s): Matti Viitala

Term and year when the thesis was submitted: Spring 2021

Number of pages: 44

---

The purpose of this thesis is to design and develop a game in Unity game engine, that utilizes adaptive music through conventional and more artistic methods. The game is a Bullet Hell-esque 2D Shooter. The music was composed in FL Studio. In the theoretical part we go through history of video game music, composer's tools, Unity audio components and technical aspects of designing sound and music for a videogame. The practical part of this thesis is divided in to two parts. In the first part we design and compose a soundtrack and the latter part we implement the sounds into our Unity project. This thesis emphasizes the development of the audio elements of the game.

The end result was a working game which demonstrates possibilities of adaptive audio. During the making of this thesis I advanced my skills in sound design and handling audio inside Unity.

---

Keywords: game music, game design, composing

# SISÄLLYS

1	JOHDANTO .....	6
2	PELIMUSIIKKI .....	7
3	ADAPTIIVINEN PELIMUSIIKKI .....	8
3.1	Yleisimmät tekniikat.....	8
3.2	Esimerkkejä adaptiivisesta musiikista.....	8
4	PELIMUSIIKIN HISTORIA .....	10
4.1	Kolikkopelit ja varhaiset konsolit.....	10
4.2	FM-synteesi.....	11
4.3	MIDI.....	12
4.4	Pelien CD-jakelu.....	13
5	SÄVELTÄJÄN TYÖKALUT.....	15
5.1	Kaiuttimet .....	15
5.2	Kuulokkeet.....	17
5.3	DAW-ohjelmistot.....	18
5.4	MIDI-ohjaimet.....	21
5.5	Audio middleware.....	22
6	UNITY JA SEN AUDIOKOMONENTIT .....	23
6.1	Audio Source.....	23
6.2	Audio Listener .....	25
6.3	Audio Mixer .....	25
7	ÄÄNIMAAILMAN SUUNNITTELU.....	27
8	ADAPTIIVISEN ÄÄNIMAAILMAN TOTEUTUS.....	30
8.1	Suunnittelu ja sävellys.....	30
8.2	Toteutus Unityssa.....	35
9	POHDINTA.....	39
	LÄHTEET.....	40

# 1 JOHDANTO

Eksyin videopelien maailmaan serkuilta lainatun NES-konsolin avulla samoihin aikoihin, kun opetelin kävelemään. Super Mario Bros. 3 on edelleen yksi lempipeleistäni, ja osa sen viehätystä oli pelin taustamusiikki, jonka melodiat ovat säilyneet muistissani vuosikymmenet. Videopeleistä muodostui nopeasti luovalle pojalle harrastus, joka kehittyi myöhemmin pelien modaamiseksi, joka puolestaan syytti palon luoda omia pelejä.

Peruskoulussa löysin rock musiikin ja sain joululahjaksi kitaran. Vuosien saatossa musiikkiharrastukseni paisui ja minusta tuli laulaja-lauluntekijä, kotistudioharrastaja ja instrumenttien hamstraaja. Tiedonjanoni musiikin suhteen kasvoi, ja varsinkin säveltämisestä ja studiotyöskentelystä kehkeytyi minulle rakas harrastus.

Opinnäytetyössäni halusin yhdistää musiikki- ja peliharrastukseni. Pelien musiikki ja äänimaailma on aihe, joka usein keskusteluissa jää hieman pelin muiden osa-alueiden varjoon, tai ainakin silloin, kun peli ei satu sisältämään ikoniseksi muodostuvaa teemamusiikkia. Onnistunut äänimaailma lielee sellainen, joka ei kiinnitä huomiota itseensä silloin, kun sen ei ole tarkoitus. Tässä työssä halusin parantaa myös omaa ymmärrystäni pelien äänimaailmoista ja siitä, kuinka ääni toimii ja kuinka sitä käsitellään etenkin Unity-pelimoottorissa.

Raportissa paneudutaan pelimusiikkiin ja sen historiaan, sen käyttötapoihin, säveltäjän työkaluihin ja sävellystä tukevaan työhön, äänen käsittelyyn Unity-pelimoottorissa. Lopuksi käydään läpi, kuinka suunnittelin ja tein äänimaailman osana tätä opinnäytetyötä toteutettuun pieneen Bullet Hell-peliin, jonka tarkoituksena oli esitellä sekä perinteisiä että luovempia tapoja käyttää adaptiivista musiikkia ja samalla hioa ja kehittää omaa tietotaitoa.

## 2 PELIMUSIIKKI

Vaikka videopeli-termillä on haluttu korostaa juuri pelien visuaalisuutta, ei äänen roolia missään vaiheessa videopelien historiaa tule vähätellä. Pelimusiikilla ja pelien äänimaailmalla viitataan kaikkien musiikkiin ja ääniefekteihin videopeleissä. Musiikki ja efektit saattavat olla joko pelin aikana digitaalisesti tuotettuja tai ennalta äänitettyjä. (Campbell 2016.)

Musiikkia ja ääntä käytetään peleissä oikean tunnelman ja tilanteen välittämiseen pelaajalle ja rikastamaan pelikokemusta. Musiikilla voidaan tehostaa pelaajan immersiota, korostaa kertomusten ja tarinoiden emotionaalista painoa, pitää yllä pelin temaattista kokonaisuutta ja korostaa pelin estetiikkaa grafiikan jatkumona. (Zehnder & Lipscomb 2006, 241.)

Nykyisin pelimusiikki elää vahvasti myös pelien ulkopuolella. Pelien musiikkia julkaistaan fyysisinä painoksina, pelimusiikkia varten järjestetään konsertteja sekä festivaaleja, ja striimauspalvelut, kuten YouTube ja Spotify, tuovat peleistä tutut kappaleet miljoonayleisön eteen.

### 3 ADAPTIIVINEN PELIMUSIIKKI

Adaptiivinen tai dynaaminen musiikki on taustamusiikkia, joka reagoi pelin sisäisiin tapahtumiin muuttamalla esimerkiksi tempoa, äänenvoimakkuutta, rytmia, melodiaa tai jotain muuta osaa musiikista. Tällä tavoin musiikki voidaan saada vastaamaan sitä, mitä ruudulla näkyy ja tapahtuu. Toisin kuin elokuvissa, pelien tapahtumat eivät yleensä etene lineaarisesti, mutta musiikin tulisi silti pystyä kuvastamaan pelin tapahtumia. Tänä päivänä käytännössä jokainen videopeli hyödyntää adaptiivista musiikkia ainakin jossain määrin. Yleisimpiä tapoja ovat taustamusiikin vaihdokset pelin eri tilanteissa, kuten pelaajan taistellessa tai siirtyessä alueelta toiseen.

#### 3.1 Yleisimmät tekniikat

Musiikkiin vaikutetaan pääsääntöisesti käyttäen kahta eri tekniikkaa. Vaakasuoraa sekvensointia (horizontal re-sequencing) ja vertikaalista uudelleenjärjestelyä (vertical re-orchestration tai reorkestraatio.) (Varney 2018.)

Vaakasuorassa tekniikassa sävelletään osia, jotka siirtyvät toisiinsa saumattomasti ja joita voidaan tarvittaessa joko toistaa tai niiden järjestystä voidaan sekoittaa. Tämä saattaa tarkoittaa lukuisten lyhyiden musikaalisten palasten säveltämistä, jolloin taustamusiikki voi olla hyvinkin erilainen eri pelikerroilla. Vertikaalisessa reorkestraatiossa muutetaan parhaillaan soivan musiikin miksausta tuomalla musiikkiin lisää raitoja tai ottamalla niitä pois. (Varney 2018.)

Esimerkiksi pelissä voidaan toistaa taustamusiikkia, johon vertikaalisella reorkestraatiolla lisätään jokin instrumentti tilanteen sitä vaatiessa vaihtamatta kappaletta kokonaan. Vertikaalinen reorkestraatio on lähtökohtaisesti hienovaraisempi tapa vaikuttaa jo soivan musiikin tunnelmaan ja sävyyn, siinä missä sekvensointi sopii paremmin musiikin viemiseen uuteen suuntaan tai tarvittaessa musikaalisen kokemuksen satunnaistamiseksi.

#### 3.2 Esimerkkejä adaptiivisesta musiikista

Vuonna 1991 *Monkey Island 2: LeChuck's Revenge* -pelissä käytettiin ensimmäistä kertaa LucasArtsin kehittämää iMUSE-järjestelmää. Pelaajan liikkuessa pelissä WoodTick-alueella huoneesta



toiseen järjestelmä tuottaa sulavia muutoksia taustamusiikin instrumentaatioissa ja melodioissa. Järjestelmä valitsee satunnaisen siirtymän, jonka jälkeen se aloittaa huonekohtaisen musiikin toistamisen. Poistuessaan huoneista järjestelmä soittaa huonekohtaisen melodian ja siirtyy tasaisesti ilman havaittavaa taukoa tai loppua takaisin alueen pääteemaan. Alueen pääteeman toisto voidaan aloittaa seitsemästä eri kohdasta. (Zaarin 2021.)

The Legend of Zelda: Breath of the Wild -pelissä vertikaalista reorkestraatiota on käytetty luovasti sivutehtävässä ”From the Ground Up”. Tehtävässä pelaaja värvää asukkaita uuteen kaupunkiin, jonka taustamusiikki on aluksi hyvin pelkistetty, hiljainen mutta toiveikas. Pelissä jokaisella kaupungilla ja alueella on oma, tunnistettava taustamusiikkinsa. Jokainen uusi kaupunkiin muuttanut asukas tuo uuden kaupungin taustamusiikkiin jotain uutta. Lisäksi jokaista uutta asukasta vastaava soitin tai melodia viittaa siihen kaupunkiin tai alueeseen, josta asukas on alun perin kotoisin. Lopputuloksena on täyteläinen sävellys, joka symboloi uutta kaupunkia kulttuurien ja rotujen kohtaustapaikkana. (Game Maker’s Toolkit 2014.)

## 4 PELIMUSIIKIN HISTORIA

Pelimusiikki on muuttunut paljon sen alkuajoista. Varhaisten pelien äänimaailman määräsivät pitkälti tekniset rajoitukset. Ensimmäiset ääntä sisältäneet pelit kykenivät vain yksinkertaisiin piip-pauksiin, mutta nykyisin pelejä varten voidaan äänittää kokonaisia sinfoniaorkestereita. Tässä luvussa käydään läpi pelien äänimaailmojen vaiheita ja niitä edistäneitä teknologioita.

### 4.1 Kolikkopelit ja varhaiset konsolit

Varhaisille videopeleille ominaiset äänet ja musiikki syntyivät teknisten rajoitusten pohjalta. Aikaa 1970-luvun loppupuolelta noin vuoteen 1983 kutsuttiin kolikkopelien kulta-ajaksi (Northfield 2018). Tuolloin musiikki oli vielä laajalti analogista ja joko nauhalle tai vinyylille taltioitua. Näiden tallennusmuotojen käyttäminen kolikkopeleissä ja kotikonsoleissa olisi ollut hyvin epäkäytännöllistä, joten laitteet varustettiin siruilla, jotka tuottivat äänet ja musiikin pelin aikana. Ennalta äänitettyä musiikkia ja efektejä ei vielä voitu peleissä käyttää, sillä tallennustila oli hyvin rajallista ja myös äänen digitaalinen tallentaminen oli vielä lapsen kengissä.

Vuonna 1977 ilmestynyt Atari 2600 -konsoli oli varustettu kaksikanavaisella sirulla, eli se pystyi tuottamaan kahta eri ääntä samaan aikaan. Toinen kanava kykeni tuottamaan ääntä, jota voitiin käyttää melodisesti ja toinen "noise"-kanava kykeni tuottamaan melua. Noise-kanavia käytettiin lähinnä ääniefektien luomisessa, sekä perkussiivisen, eli lyömä- ja rytmisoitinten äänen simuloimisessa. 70-luvun loppupuolella ensin kolikkopelinä julkaistu Space Invaders julkaistiin vuonna 1980 myös Atari 2600 -konsolille. Pelin taustalla toistui neljän nuotin laskeva melodia, joka kiihtyi pelaajan tuhotessa lähestyvä vastustaja. (Library of Congress 2019.)

Vaikka ääni olikin tähän mennessä muodostunut olennaiseksi osaksi pelejä, niin varsinaisella musiikilla oli ollut vielä pieni rooli. Tekniikka ei vielä mahdollistanut suuria sovituksia, joissa useat soittimet soivat yhtä aikaa. Muita syitä musiikin puuttumiseen kolikkopeleissä saattoi olla esimerkiksi ympäristö, jossa pelejä pelattiin. Kolikkopelejä pelattiin julkisella paikalla, jossa esiintyi luonnollisesti taustamelua ja hyvin mahdollisesti myös muuta musiikkia (Driscoll & Diaz 2009) Kehittäjät saattoivat mieltää musiikin lisäämisen peliin matalaksi prioriteetiksi, ja mahdollisesti käytettävien äänikanavien rajattu määrä sai kehittäjät suosimaan ääniefektejä musiikin sijaan.

Ensimmäinen peli, joka sisälsi jatkuvaa taustamusiikkia, oli vuonna 1980 kolikkopelinä julkaistu Rally-X. Tekniikka ei kuitenkaan vielä mahdollistanut muita ääniä musiikin soidessa. Seuraavana vuonna julkaistu kolikkopeli Frogger oli ensimmäinen peli, joka vaihtoi taustamusiikkia pelaajan siirtyessä tasosta toiseen. (Library of Congress 2019.)

1980-luvulla kolikkopelien ja konsolien äänisirut kehittyivät mahdollistaen useampia yhtäaikaisesti soivia ääniakanavia, ja kehittäjät oppivat tuntemaan sirujen mahdollisuudet sekä toimimaan tehokkaammin rajoitusten sisällä, mikä johti äänimaailman osalta luultavasti rikkaampaan kokemukseen. Siinä missä varhaisempien konsolien äänimaailmat koostuivat enimmäkseen erilaisista piippauksista, vuonna 1982 julkaistu Commodore 64 sisälsi SID (Sound Interface Device) -siru, joka oli jo käytännössä ammattilaistason syntetisaattori. Tämä siru pystyi toistamaan yhtä aikaa kolmea eri ääntä, ja sen ääniala oli jopa yhdeksän oktaavia. Vertauksen vuoksi kerrottakoon, että perinteisessä pianossa on 88 näppäintä eli vähän yli seitsemän oktaavia. SID-siru pystyi antamaan äänelle neljä eri aaltomuotoa, mikä käytännössä määrittää sen, miltä ääni kuulostaa. Lisäksi laite pystyi säätämään äänen sointia suotimilla. Suotimet kuuluvatkin syntetisaattoreiden perusominaisuuksiin. (Barton & Loguidice 2007.) Vuotta myöhemmin julkaistu Nintendo Entertainment Systemin sisälsi jo viisi ääniakanavaa verrattuna Commodore 64:n kolmeen. Kaksi sen kanavaa oli varattu pulssi-aallolle eli kantiaallon variaatiolle ja yksi kolmioaallolle, jota käytettiin usein basson roolissa. Neljäs kanava oli noise-kanava. Viimeinen kanava oli sample-kanava, jolle voitiin äänittää lyhyitä ”sampleja” eli äänitteitä. Tältä kanavalta voitiin esimerkiksi soittaa ääniä, jotka olisivat olleet käytännössä mahdottomia tuottaa muita äänikanavia käyttäen. (explod2A03 2012.)

## 4.2 FM-synteesi

Seuraava digitaaliselle musiikille merkittävä teknologia oli taajuusmodulaatiosynteesi eli FM-synteesi. Teknologia itsessään oli keksitty jo paljon aikaisemmin, mutta Yamahan vuonna 1983 julkaiseman DX7-kosketinsoittimen avulla se mullisti musiikkimaailmaa. (Yamaha Design Laboratory 2021.) Yksinkertaistettuna FM-synteesissä käytetään ääniaaltoja A ääniaallon B modulointiin. FM-synteesiä hyödyntämä syntetisaattori siis tuottaa ääntä erilaisten algoritmien kautta, jotka saattavat olla hyvinkin monimutkaisia, ja tuloksena olevien erilaisten äänien määrä on käytännössä ääretön. FM-synteesillä onkin mahdollista jäljentää perinteisten instrumenttien kuten pianon, kitaran ja jousitai puhallinsoittinten ääntä ainakin jossain määrin. FM-synteesin käyttö ei rajoitu vain musikaalisten

äänien tuottamiseen, vaan sillä pystytään tuottamaan myös monipuolisesti erilaisia ääniefektejä, kuten esimerkiksi räjähdyksiä. (Crute 2019.)

### 4.3 MIDI

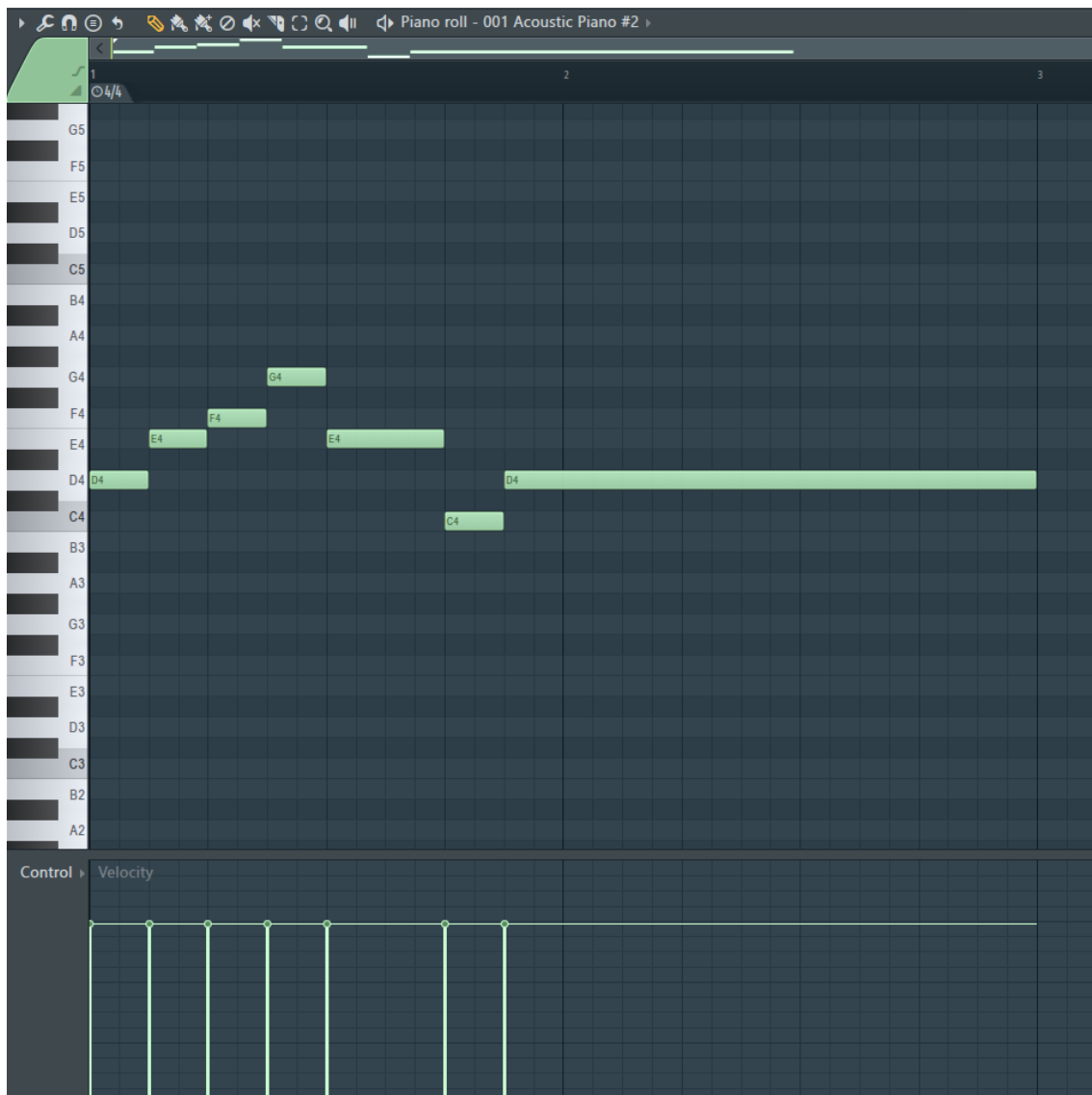
MIDI eli Musical Instrument Digital Interface on tiedonsiirtojärjestelmä, jota käytetään esimerkiksi tietokoneiden, soittimien ja ohjainten väliseen kommunikointiin. Se kehitettiin 80-luvun alkupuolella tarkoituksena standardisoida määrällisesti nopeasti kasvavan musiikkilaitteiden välinen tiedonsiirto, ja se on käytössä vielä tänäkin päivänä. (Hahn 2020.)

Havainnollistamiseksi mitä MIDI-data on, sitä voidaan verrata perinteisiin nuotteihin. MIDI-data on siis informaatiota siitä, mitä ja miten soitetaan, mutta itsessään se ei sisällä mitään ääntä. Tämä tarkoittaa sitä, että esimerkiksi MIDI-koskettimisto on vain MIDI-ohjain, eikä se tuota itsessään ääntä, jos se ei sisällä esimerkiksi syntetisaattoria. Sitä voidaan kuitenkin käyttää niin hardware- kuin software-syntetisaattorien tai virtuaalisten instrumenttien ohjaamiseen, jotka vuorostaan tuottavat äänen niille annetun datan pohjalta. (Hahn 2020.)

Tietokoneella MIDI-dattaa voidaan syöttää DAW-ohjelmistoon myös manuaalisesti ilman erillisiä MIDI-kontrollereita. Kontrollereita tai koskettimia käyttämällä MIDI-dattaa pystytään kuitenkin syöttämään reaaliajassa. MIDI-dattaa voidaan myös tallentaa esimerkiksi kosketinsoittajan soitosta. Tällöin jokainen nuotti tallentuu niin kuin se soitettiin MIDI-ohjaimen ominaisuuksien mukaan. Jos soitossa sattuu tulemaan virheitä, niiden korjaaminen sekvenssissä DAW-ohjelmistosta riippumatta pitäisi olla helppoa, sillä jokaisen soitetun nuotin ominaisuuksia voidaan säätää erikseen. (Hahn 2020.) Kuvioissa 1 ja 2 näkyy molemmissa sama melodia. Kuviossa 1 melodia on esitetty perinteisinä nuotteina, ja kuviossa 2 se on syötetty MIDI-datana DAW-ohjelmistoon.



KUVIO 1. Esimerkki melodiasta perinteisiä nuotteja käyttäen (Haapakoski 2021a.)



KUVIO 2. Esimerkki melodiasta DAW-ohjelmiston piano roll -näkyvässä (Haapakoski 2021b.)

#### 4.4 Pelien CD-jakelu

CD-levyt saapuivat markkinoille 80-luvun alkupuolella. Ensimmäiset CD-levyt ja soittimet olivat käytännössä varattu musiikille (Kivelä 2020). Noin vuosikymmen myöhemmin valmistajat alkoivat lisätä CD-asemia konsoleihin ja tietokoneisiin (Cohen 2021).

Ilmestyessään CD oli aikansa laajin tallennusformaatti ja myös halvempi valmistaa kuin aikaisemmin pelien jakelussa käytetyt kasetit ja levykkeet. Tallennustilan moninkertaistuesssa kehittäjät pysyivät käyttämään enemmän ja korkealaatuisempia ja enemmän tallennustilaa vieviä asset-tiedostoja, eli pelissä käytettäviä tiedostoja, jotka sisältävät yleensä esimerkiksi ääntä tai grafiikkaa. Tämä johti paremman näköisiin ja kuuloisiin peleihin, eikä pelejä enää tarvinnut jakaa usealle levykkeelle.

Runsaampi tallennustila mahdollisti musiikin suoratoistamisen (streaming), eli ääniä ei enää tarvinnut generoida pelin aikana, vaan ne oli mahdollista äänittää ennakkoon ja toistaa tarvittaessa. (Cohen 2021.)

## 5 SÄVELTÄJÄN TYÖKALUT

Tässä luvussa käsitellään yleisimpiä työkaluja, joita säveltäjät ja muut äänentuottajat hyödyntävät työssään. Käydään läpi niin laitteisto-, kuin ohjelmistopuolta.

### 5.1 Kaiuttimet

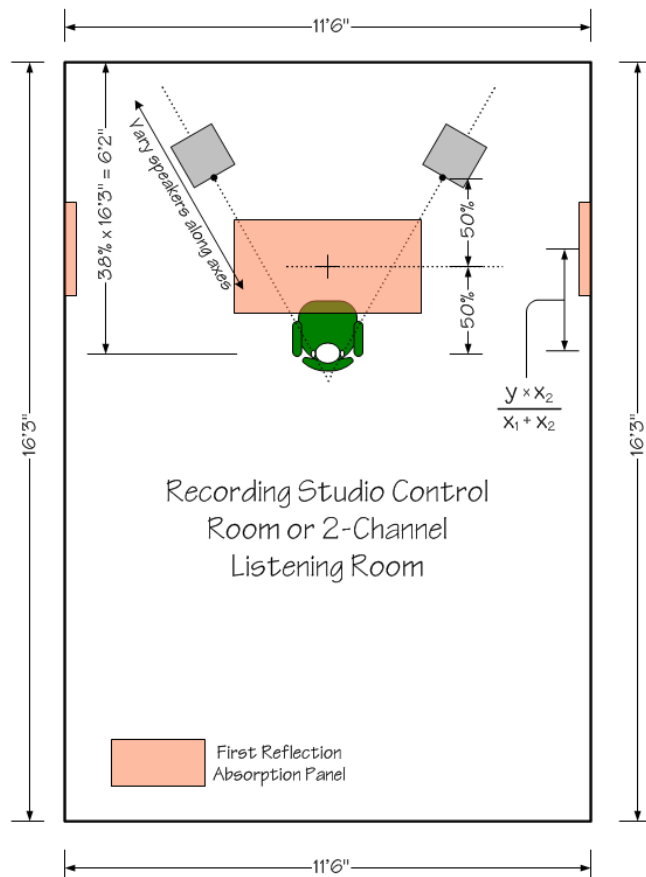
Ääntä tuottaessa tärkeintä on tietenkin kuulla, mitä ollaan tekemässä. Äänenulostuloa varten on olemassa käytännössä kaiuttimia ja kuulokkeita, ja niillä molemmilla on omat heikkoutensa ja vahvuutensa.

Kun halutaan tuottaa mahdollisimman hyvälaatuista ääntä, on tärkeää, että säveltäjä tai tekniikko kuulee työn jälkensä tarkasti. Tätä varten on olemassa studiomonitorikaiuttimia, jotka ovat suunniteltu toistamaan ääntä mahdollisimman neutraalisti eli *rehellisesti* mitään taajuusaluetta, kuten esimerkiksi matalia bassotaajuuksia, lisäämättä tai leikkaamatta. Kuluttajille suunnatut äänentoistojärjestelmät usein muokkaavat ääntä *paremmaksi* esimerkiksi lisäämällä jotain taajuusaluetta, jotta ääni olisi kirkaampi tai jyrkempi. Tällaiset kaiuttimet ovatkin suunniteltu valmiin musiikin kuunteluun ja toistavatkin musiikin hyvin esimerkiksi myös pidemmällä matkalla, toisin kuin studiomonitorikaiuttimet, etenkin lähikenttämonitorit, jotka ovat tarkoitettu käytettäväksi hyvin lähellä miksaajaa mahdollisimman tarkan kuuntelun mahdollistamiseksi. (Wreglesworth 2021.)

Monitorikaiuttimista kaiken hyödyn irti saaminen ei välttämättä ole helppoa. Kuunteluun vaikuttavia seikkoja ovat esimerkiksi kaiuttimien sijoittelu huoneessa, niiden etäisyydet seiniin, lattiaan, kattoon, toisiinsa ja kuunteliijaan, ovatko kaiuttimet telineiden vai pöydän päällä sekä huoneen akustiikka. Kaiuttimien oikeaoppiseen sijoitteluun kannattaa käyttää aikaa ja tarkkuutta. Sijoitteluun löytyy paljon oppaita ja kaavoja erimallisia huoneita varten. Silmäilyn sijaan kannattaa käyttää mittanauhaa. Akustisesti hoitamattomassa huoneessa miksaaja ei saa rehellistä ja tarkkaa kuvaa kaiuttimista tulevasta äänestä. Esimerkiksi jos huoneessa pahimmassa tapauksessa on selvästi kuultava kaiku, vaikeuttaa se äänen kanssa työskentelemistä huomattavasti varsinkin silloin, kun yritetään itse lisätä ääneen kaikua. Tällöin miksaaja saattaa lisätä sitä huomattavasti liikaa tai liian vähän. (Arqen 2021.)

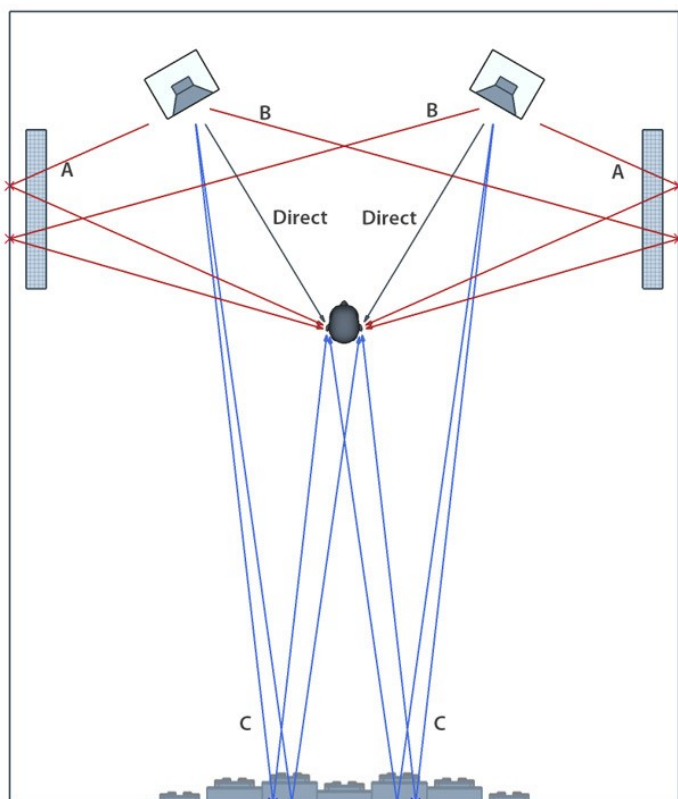
Toinen akustinen haaste kuuntelutilalle ovat bassotaajuudet. Toisinkin kuin korkeat äänet, bassotaajuudet ovat vähemmän suuntaavia ja kulkevat seiniä ja lattiaa pitkin. Mitä lähempänä kuuntelija on seinää, lattiaa tai kattoa, sitä enemmän hän kuulee bassotaajuuksia. Puolivälimatassa vastakaisten pintojen välissä on niin sanottu null-alue, joissa basso toistuu huoneessa vähiten. Kuuntelupiste ei siis saisi olla liian keskellä huonetta eikä liian lähellä seinää. (Winer 2021.) Bassotaajuk-sien kuuleminen saattaakin olla tilassa vaikeaa. Subwoofer-kaiuttimen hankkiminen saattaa tällöin auttaa asiassa. (Arqen 2021.)

Kaiuttimien asettelussa on tärkeää asettaa ne huoneeseen symmetrisesti, mielellään osoittamaan huoneen pitempien seinien mukaisesti, jos vain mahdollista. Kun kaiuttimien paikat on löydetty, seuraavaksi tulee etsiä first reflection -pisteet eli kohdat huoneen sienien pinnoilla, joista ääni ensimmäisenä heijastuu kuuntelijan korviin. Tavoitteena on siis se, että ääni tulee kaiuttimista suoraan kuuntelijan korviin, eikä heijastettuna seinistä, katosta tai lattiasta. Nämä kohdat peitellään akustiikkaelementeillä, kuten tarkoitukseen valmistetuilla paneeleilla. (Arqen 2021.) Kuvio 3 esittää kaiuttimien sijoitteluun huoneessa liittyvää tietoa. Kuvio 4 esittää first reflection -pisteiden sijainnin ja äänen heijastumisen huoneessa.



KUVIO 3. Kaiuttimien sijoittelu huoneessa (Winer 2021.)





KUVIO 4. Äänen heijastuminen huoneessa ja first reflection -pisteet (Arqen 2021.)

## 5.2 Kuulokkeet

Kuten kaiuttimia, myös kuulokkeita on valmistettu sekä kuluttaja- että studiokäyttöön. Yleisesti vaihtaisi siltä, että kuulokkeiden paikka studiossa ei ole olla niinkään osana miksausta, vaan äänittäessä monitorointia varten silloin, kun kaiuttimista tuleva ääni olisi vaarassa vuotaa äänittävään mikkiin. Kuulokkeita ei kannata sulkea kokonaan poiskaan, sillä paras lopputulos lienee sellainen, joka kuulostaa hyvältä niin kaiuttimilla kuin kuulokkeilla. Kuulokkeilla kuunnellessa saatat kuulla jotain huomaamisen arvoista, joka olisi mennyt pelkästään kaiuttimilla kuunnellessa ohi (Robjohns 2006).

Kuulokkeet eroavat kaiuttimista eniten siinä, minkälaisen stereokuvan ne kuuntelijalla antavat. Stereoääni muodostuu aina kahdesta kanavasta, oikeasta ja vasemmasta. Koska kuulokkeissa kaiuttimet ovat kuuntelijan pään vastakkaisilla puolilla, muodostuva stereokuva on laajempi, kun kaiuttimilla kuunnellessa, koska silloin ääni kimpoilee huoneessa ja tulee kuuntelijan korvaan useammasta suunnasta. Tätä ilmiötä kutsutaan ristiinsyötöksi. Silloin molemmat korvat kuulevat jonkin verran stereokuvan vasenta ja oikeaa kanavaa. Kuulokkeilla vasen ja oikea korva kuulevat vain

oman puoleisensa kanavan, jolloin ristiinsyöttö puuttuu ja stereokuva vaikuttaa laajemmalta. (McAlister 2018.)

Vaikka ristiinsyöttö on yleensä toivottua miksatessa, sen puutteesta voi myös olla hyötyä varsinkin panoroitaessa pelin äänisuunnittelun kannalta. Pelille säveltäessä ja äänimaailmoja suunniteltaessa ei ole järjetöntä olettaa, että suuri määrä, ehkä jopa enemmistö pelaajista käyttäisi kuulokkeita.

### 5.3 DAW-ohjelmistot

DAW eli Digital Audio Workstation on sovellus äänen tallentamiseen, muokkaamiseen ja miksaamiseen. Ensimmäiset DAW-ohjelmistot kehitettiin 1970–80 lukujen vaihteessa. Ennen tätä äänen nauhoittaminen oli alusta loppuun täysin analoginen prosessi. Tietokoneiden kehitys mahdollisti äänityön siirtymisen digitaalisen muotoon. (Recording Connection 2021.) DAW-ohjelmistoja on useita erilaisia eri valmistajilta, ja vaikka ne sisältävätkin joitakin eroja, suurin osa markkinoiden suosituimmista ovat varsin kykeneviä ja taipuvaisia eritasoisille tekijöille ja erilaisille musiikkityyleille. Yleisimpiin DAW-ohjelmistojen ominaisuuksiin kuuluu:

- Äänen digitaalinen käsittely, eli yhden tai useamman raidan äänittäminen ja näiden muovaaminen erilaisia työkaluja kuten taajuuskorjaimia, kompressoreja tai kaikuja käyttäen.
- MIDI sekvensseri eli työkalu, johon syötetään aikajanelle tavalla tai toisella, esimerkiksi koskettimia soittamalla MIDI-dattaa, jolla puolestaan ohjataan erilaisia virtuaalisia instrumentteja. (Murphy 2020.)
- VST-tuki. VST on lyhenne sanoista Virtual Studio Technology. Se julkaistiin vuonna 1996 tarkoituksena tuoda aiemmin analogiset efektit ja työkalut tietokoneen digitaalisen ympäristöön reaaliaikaista käyttöä varten. (Vincent 2019.) Ennen VST-laajennuksien keksimistä, jos jo äänitettyyn ääniraitaan tahdottiin lisätä esimerkiksi kaikua, tuli signaali viedä DAW-ohjelmistosta ulos johtoa pitkin esimerkiksi analogiseen kaikulaitteeseen, jonka läpi se johdettiin takaisin DAW-ohjelmistoon. Muutama vuosikymmen julkaisunsa jälkeen erilaisia VST-laajennuksia on joka tarpeeseen. Osa niistä mallintaa olemassa olevia hardware-efektejä tai työkaluja ja osa hyödyntää puhtaasti digitaalista ympäristöä tavalla, joka mahdollistaa äänien tai efektien luomisen, jotka olisivat aiemmin analogisessa ympäristössä olleet käytännössä mahdottomia toteuttaa.

VSTi eli Virtual Studio Technology Instrument on nimensä mukaisesti virtuaalinen instrumentti, eli VST-laajennus, jonka tarkoitus on tuottaa ääntä. VST-instrumentteja on lukemattomia erilaisia. Osa niistäkin saattaa mallintaa perinteisiä soittimia tai hardware-syntetisaattoreita. Laadukkaiden virtuaalisten instrumenttien valmistuksessa on saatettu esimerkiksi äänittää muusikoita soittamassa oikeita soittimia autenttisen lopputuloksen takaimiseksi. (Vincent 2019.) Jos DAW-ohjelmiston käyttäjä tarvitsee esimerkiksi pianon ääntä, saattaa hän ostaa VSTi-instrumentin joltain lukuisista VST-laajennuksista kehittävilta yrityksiltä. Tarjontaa on paljon ja hintahaarukka vaihtelee ilmaisesta satoihin euroihin ja hinta heijastuu usein laatuun. Virtuaalisia instrumentteja käytetään DAW-ohjelmistoissa ohjaamalla niitä MIDI-datalla sekvensserissa. Niitä voidaan myös toistaa reaaliajassa MIDI-ohjaimia käyttäen. VST ei ole ainoa efektejä ja virtuaalisia instrumentteja tarjoava laajennusmuoto. Vaihtoehtoisina formaatteja ovat esimerkiksi AAX ja Mac-käyttäjille tarkoitettu AU. (O'Connor 2019.)

- Nuotitus eli MIDI-datan muuttaminen perinteisiksi tulostettaviksi nuoteiksi (Murphy 2020).

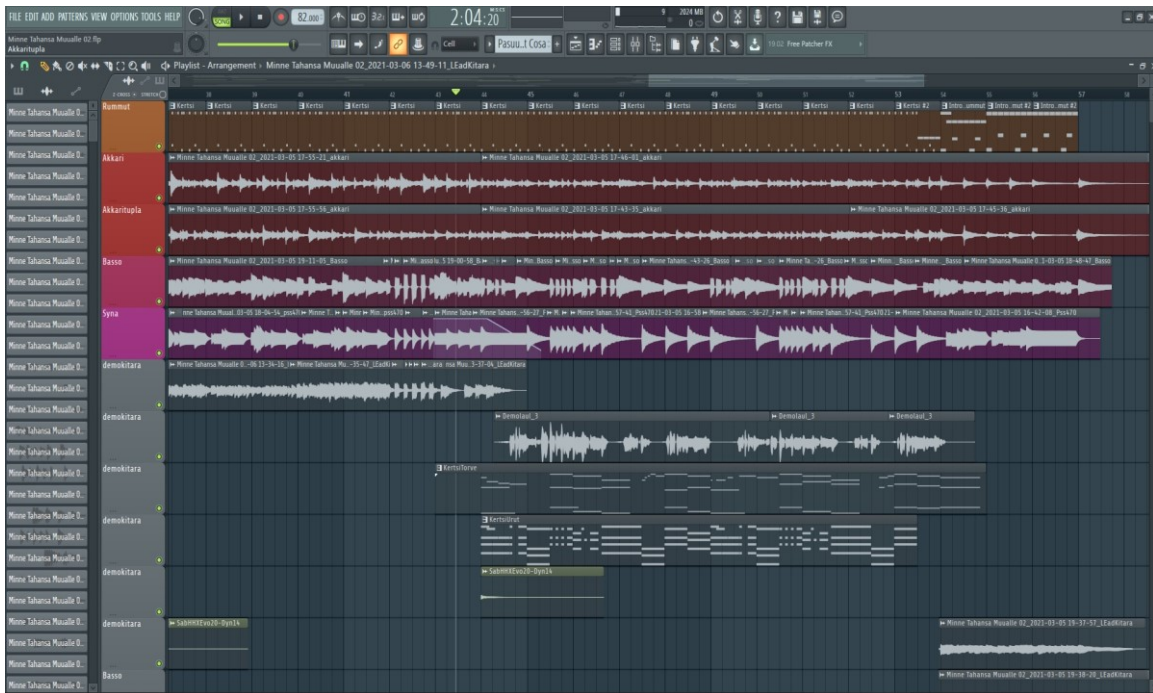
DAW-ohjelmistoissa pystytään nauhoittamaan ääntä yksi tai useampi raita kerrallaan. Äänittämistä varten tietokoneeseen kytketään yleensä äänikortti tai mikseri, johon voidaan kytkeä esimerkiksi mikrofoneja tai soittimia. (Murphy 2020.) Erilaiset äänikortit ja mikserit sisältävät erilaisia ominaisuuksia, ja osa varsinkin halvimmissa mikserimalleista ei välttämättä tue moniraitaäänitystä USB-yhteydestä huolimatta. Korttien ja mikserien hinnat nousevat sitä mukaa, mitä enemmän kytkentöjä ja ominaisuuksia ne sisältävät. Perinteisten instrumenttien äänittäminen on oma taiteenlajinsa mutta ei suinkaan ainoa tapa tehdä musiikkia DAW-ohjelmistoissa.

Virtuaalisten instrumenttien ohjaaminen on tehty käyttöliittymän puolesta hyvin helpoksi, ja nuotteja pystyykin yleensä lisäämään haluamaansa paikkaan vain hiirellä klikkaamalla. Lisäksi suurin osa DAW-ohjelmistoista sisältää jo valmiina virtuaalisia instrumentteja ja efektejä, joilla pääsee heti alkuun.

DAW-ohjelmistot sisältävät aina aikajanan, jolle työn alla oleva musiikkikappale rakennetaan. Se etenee aina vasemmalta oikealle ja siihen on aseteltu vertikaalisesti raidat, jolle lisätään joko ääntä tai MIDI-dataa. Hyvässä DAW-ohjelmistossa asioiden siirtely raidoilla on vaivatonta. Työtä helpottavat perinteiset tietokoneen leikepöytätoiminnot, jotka toimivat myös DAW-ohjelmistoissa (Recording Connection 2021.) Esimerkiksi jotain kohtaa kappaleesta ei välttämättä tarvitse äänittää kahdesti, vaan se voidaan kopioida sinne missä sitä tarvitaan niin monesti kuin tarve on.

DAW-ohjelmistoissa suoritetaan myös miksaus, jossa siis säädetään raitojen äänenvoimakkuudet, katsotaan niiden paikat stereokuvassa ja ikään kuin kiillotetaan työn alla olevan kappaleen äänet kuntoon niin, että ne kuulostavat ennen kaikkea hyvältä yhdessä. Tässä vaiheessa kuvaan astuvat VST-efektit kuten taajuuskorjaimet, joilla saatetaan leikata tai lisätä taajuuksia eri äänistä, jotta tehdään tilaa toisille äänille tai tuodaan jotain ääntä tietyllä äänialueella enemmän esille tai äänen muotoiluun puhtaasti taiteellisessa tarkoituksessa. Myös esimerkiksi kaikujen ja muiden efektien lisääminen ja muokkaaminen kuuluvat miksauseseen. (Murphy 2020.)

Mainitsemisen arvoisia DAW-ohjelmia ovat esimerkiksi Pro Tools ja Ableton Live, jotka ovat mahdollisesti markkinoiden suosituimmat vaihtoehdot (Recording Connection 2021). Varsinkin Pro Toolsiin törmää usein ammattistudioissa. Muitakin vaihtoehtoja toki löytyy, ja vaikka täysversioiden lisenssit liikkuvat sadoissa euroissa, myös kokeiluversioita ja ilmaisia vaihtoehtoja on useampi. Reaper sisältää käyttäjän tuloihin perustuvan lisenssin, ja harrastelijapiireissä törmää usein sitä suositteleviin ihmisiin. Täysin ilmaisista vaihtoehdoista Mac-käyttöjärjestelmälle on tarjolla GarageBand. Windowsille on saatavilla ilmainen avoimen lähdekoodin Audacity. Se lienee käyttöliittymältään kankein ja myös vaikein käyttää ja tästä syystä se usein sivuutetaan vaihtoehtona kokonaan pois. Kuvio 5 on kuvakaappaus FL Studio DAW-ohjelmiston aikajanasta. Aikajanalla näkyy äänien aaltomuotoja ja MIDI-dataa sisältäviä kuviota.



KUVIO 5. FL Studion Playlist-näkymä (Haapakoski 2021c.)

## 5.4 MIDI-ohjaimet

MIDI-ohjaimet mallintavat usein pianon koskettimia. Tästä huolimatta kaikki MIDI-ohjaimet eivät ole kosketinsoittimia, eivätkä kaikki kosketinsoittimet ole MIDI-ohjaimia, vaikkakin useat kotikäyttöön tarkoitetut kosketinsoittimet sisältävät mahdollisuuden syöttää MIDI-dataa USB-yhteyden kautta tietokoneelle. MIDI-ohjaimet usein sisältävät erilaisia nappeja ja säätimiä MIDI-datan hallintaa varten. (Computer Music Specials 2009.)

MIDI-ohjain ei ole säveltäjälle välttämätön työkalu, mutta se nopeuttaa ja helpottaa työtä huomattavasti. MIDI-ohjaimen koskettimistolla säveltäjä pystyy ohjaamaan VST-instrumentteja reaaliajassa tai tallentamaan syöttämänsä MIDI-datan. Ilman koskettimia jokainen nuotti syötettäisiin joko hiirellä tai näppäimistöllä. Tyypillinen MIDI-ohjain liitetään tietokoneeseen USB-yhteydellä, mutta on olemassa myös varsinaisia MIDI-liitäntöjä, joilla voidaan esimerkiksi syöttää MIDI-dataa hardwaresyntetisaattorille. Myös useat kotikäyttöön tarkoitetut kosketinsoittimet sisältävät mahdollisuuden syöttää MIDI-dataa USB-yhteyden kautta tietokoneelle. (Computer Music Specials 2009.) Kuvio 6 on valokuva erilaisilla ohjelmoitavilla napeilla ja säätimillä varustetusta MIDI-ohjaimesta.



KUVIO 6. Pianon koskettimia mallintava MIDI-ohjain (Haapakoski 2021d.)

## 5.5 Audio middleware

Usein peleissä tarvitaan tai niiltä odotetaan hyvin samankaltaisia ratkaisuja ja toimintatapoja äänen suhteen. Audio middleware on ohjelmisto, jonka tarkoitus on helpottaa ääneen liittyvien toimintojen toteuttamista peleissä. Se sijoittuu pelimoottorin ja äänipiirin väliin ja tarjoaa kehittäjille yleisiä toimintoja kuten musiikin ja äänten häivyttämistä tai satunnaistoittoa. Ilman audio middlewarea nämä toiminnot toteutettaisiin pelimoottorissa. Audio middlewaren ollessa kolmannen osapuolen ulkoinen työkalu se mahdollistaa helposti saman koodin ja toimintojen käyttämisen eri peliprojekteissa, jopa eri alustoilla ja konsoleilla. Tämä keventää myös ohjelmoijan taakkaa ja antaa mahdollisuuden ääniteknikolle tai säveltäjälle suunnitella ja luoda prototyyppejä mutkikkaistakin äänijärjestelmistä, jotka ilman Audio middlewarea vaatisivat ohjelmointitaustaa. (Brown 2021.)

Audio middlewaren käyttäminen pelinkehityksessä on yleistynyt, ja sen hallintaa usein odotetaan äänityöntekijältä pelialla. Kyky hallita ja työskennellä sujuvasti ohjelmistossa mahdollistaa sujuvan työskentelin eri projektien tai töiden välillä. Siinä missä 3D-artistin pystyy siirtymään projektista toiseen käyttäen samaa mallinnusohjelmistoa, Audio middleware mahdollistaa saman äänen parissa työskenteleville. (Brown 2021.) Suosittuja Audio middleware ohjelmistoja on esimerkiksi WWise, FMOD Studio ja Fabric.

## 6 UNITY JA SEN AUDIOKOMPONENTIT

Tässä luvussa käsitellään Unity-pelimoottorin sisäisiä ääniominaisuuksia. Unityssa peliobjektit koostuvat erilaisista komponenteista. Äänen kannalta tärkeimmät komponentit ovat Audio Source ja Audio Listener.

Unity tukee 3D-tasoista ääntä ja sisältää työkaluja äänien ja efektien reaaliaikaiseen miksaamiseen ja hallintaan. Unity pyrkii mallintamaan äänen käyttäytymistä oikeassa elämässä. Äänellä on aina lähde ja sen ottaa vastaan kuuntelija. Unityssä lähteitä ja vastaanottajia edustavat komponentit Audio Source ja Audio Listener. Unity tukee yleisimpiä äänitiedostotyyppjejä, kuten AIFF, WAV, MP3 ja Ogg. (Unity 2021a.)

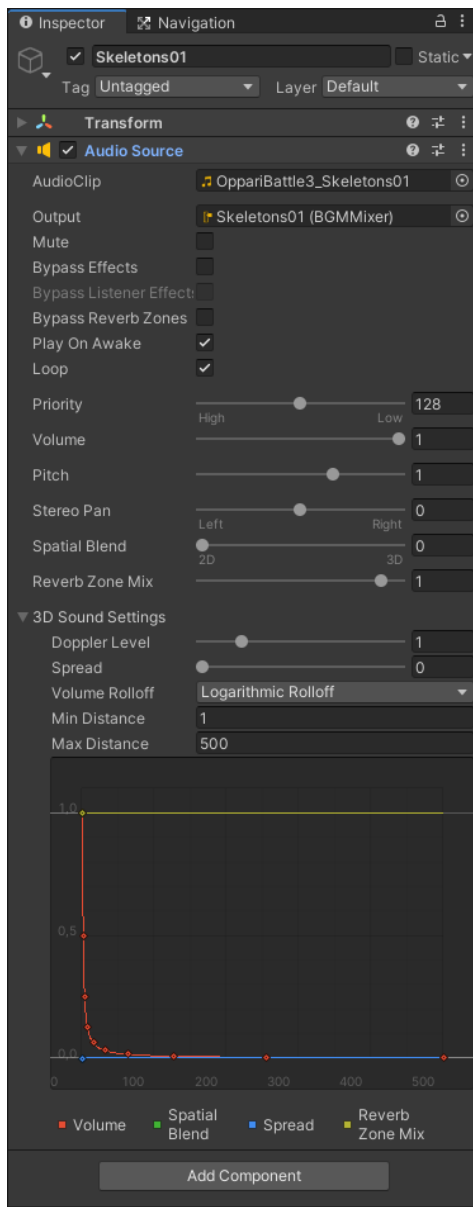
### 6.1 Audio Source

Unityssa ääni lähtee aina Audio Source -komponentista. Audio Source ei itsessään tuota ääntä, vaan se ohjaa sille osoitetun äänitiedoston (Audio Clip) toistoa. Ääni toistetaan pelaajalle joko Audio Mixerin kautta Audio Listener -komponenttiin tai ilman mikseriä suoraan Audio Listener -komponenttiin. (Unity 2021a.)

Audio Source -komponentti kiinnitetään siis peliobjektiin. Komponentti voidaan säätää toistamaan ääntä heti aktivoituttuaan tai sille voidaan antaa koodin kautta komento toistaa ääni myöhemmin tilanteen sitä vaatiessa. Audio Source -komponentin sisältävä peliobjektin täytyy olla aktiivisena pelimaailmassa toistaakseen ääntä. (Unity 2021a.)

Ulostulon lisäksi Audio Source -komponentin kautta voidaan säätää äänen avaruudellisuuteen liittyviä asetuksia, kuten toistetaanko ääni kaksi- vai kolmiulotteisesti vai sekoituksena molempia. Näillä asetuksilla ääni voidaan siis panoroida kuulumaan oikeasta suunnasta suhteessa Audio Sourcen ja Audio Listenerin sijainnista riippuen. Samoin muutkin äänen käyttäytymistä tilassa säätävät asetukset, kuten Doppler-ilmiö ja äänen kantama ja sen voimakkuuden heikentyminen välimatkan kasvaessa, löytyvät Audio Source -komponentista. Myös äänen voimakkuutta, prioriteettia sekä korkeutta voidaan säätää komponentin asetuksista ja niitä voidaan myös ohjata koodilla viit-

taamalla kyseiseen komponenttiin. Näihin asetuksiin koodataan usein pientä satunnaista variaatiota, jos ääntä toistetaan usein. Näin ääni kuulostaa hieman luonnollisemmalta, eikä siltä, että tietokone toistaa sen joka kerta täysin samanlaisena. Tätä menetelmää saatetaan käyttää esimerkiksi erilaisten kolahdusten, räjähdysten, särkymisäänien ja aseiden laukausten kanssa. (Unity 2021a.)



KUVIO 7. Audio Source -komponentti (Haapakoski 2021e.)

Kuviossa 7 on esitetty Audio Source -komponentti inspector-näkymässä kaikkine asetuksineen. Tässä kuviossa komponentille on annettu toistettavaksi luurankoa vastaava ääniraita ja sen ulostulo on asetettu omalle mikserikanavalleen. Loop-asetus on päällä ja ääni on asetettu toistettavaksi heti peliohjelman aktivoiduttua.



## 6.2 Audio Listener

Audio Listener on ikään kuin mikrofoni tai pelaajan korvat. Se poimii Audio Source -komponenteista lähtevät äänet ja toistaa ne tietokoneen kaiuttimilla. Se kiinnitetään yleensä joko tärkeimpään kameraan tai pelaajaa edustavaan peliobjektiin. Tällöin se poimii äänet pelaajan ympäriltä ja tarvittaessa emuloi niiden avaruudellista sijaintia pelimaailmassa.

Audio Listener -komponentti ei sisällä minkäänlaisia säätöjä. Se täytyy vain olla olemassa toimiakseen. Jotta Audio Listener -komponentti toimisi oikein, tulee pelimaailmassa olla niitä vain yksi kappale. (Unity 2021b.)

## 6.3 Audio Mixer

Audio Mixer on mikseri, joka jakautuu Audio Group -ryhmiin. Group on yksi kanava mikserissä, jolle voidaan osoittaa useampia Audio Source -komponentteja. Tästä eteenpäin raportissa Audio Mixer Group -ryhmiä kutsutaan kanaviksi. Mikserissä on aina master-kanava, jonka kautta kaikki mikserin muut kanavat kulkevat matkalla ulostuloon, kuten tietokoneen kaiuttimiin. Audio Source -komponentissa voidaan siis osoittaa mikserin kanava, johon ääni ohjataan. Unityn Audio Mixer toimii hyvin samalla tavalla kuin perinteinen mikseri, ja jokaiselle kanavalle voidaan asettaa erilaisia sisään rakennettuja efektejä ja kanavien ominaisuuksia, kuten äänenvoimakkuutta voidaan säätää erikseen ja jopa pelitilassa reaaliajassa. (Unity 2021c.)

Mikserin kanavat muodostavat hierarkian, jossa Master-kanava on aina ylimpänä. Uusia kanavia voidaan luoda toisten kanavien alle, jolloin niiden ulostulo kulkee hierarkiassa korkeammalla olevan kanavan kautta Masteriin. Näin voidaan esimerkiksi luoda kanava nimeltään BGM, jonka alla oleville kanaville osoitetaan pelin taustamusiikki. Tällä tavoin voidaan hallita kaikkia BGM-kanavan alla olevia kanavia säätämällä vain hierarkian ylintä kanavaa. Saman kaltaista järjestelyä on hyvä käyttää muillekin äänille, jotka voidaan niputtaa yhteen, esimerkiksi efekteille. Kun musiikki ja efektit ovat jo eroteltu toisistaan näin, voidaan toteuttaa esimerkiksi ääniasetusvalikko, jossa pelaaja voi säätää erikseen musiikin ja efektien äänenvoimakkuutta. (Unity 2021c.)

Audio Mixerissä Snapshot on Audio Mixerin tallennettu tila. Kun uusi Snapshot luodaan, Unity tallentaa mikserin sen hetkisen tilan sellaisenaan ottaen mukaan kanavien parametrit. Sen lisäksi että Snapshottien käytöstä on hyötyä pelkästään eri miksausten kokeilemisissa, niin käytössä oleva Snapshot voidaan vaihtaa toiseen myös pelin aikana. Tällä tavoin voidaan esimerkiksi häivyttää taustamusiikki toiseen. (Unity 2021c.)

Unityssä on mahdollista hallita Audio Mixerin parametrejä koodin kautta käyttämällä `expose to code`-toimintoa. Tämä toiminto löytyy klikkaamalla inspector-näkymässä jotain mikserin parametria, kuten äänenvoimakkuutta hiiren oikealla näppäimellä. Tämän jälkeen parametri nimetään ja sen arvo voidaan säätää skriptillä käyttämällä `SetFloat`-funktiota. (Unity 2021d.)

## 7 ÄÄNIMAAILMAN SUUNNITTELU

Tässä luvussa ei pohjauduta varsinaiseen sävellystyöhön tai musiikillisiin asioihin, vaan säveltäjän työtä helpottavaan suunnitteluun. Valmistautuessaan uuteen projektiin, säveltäjälle on hyvin tärkeää saada riittävä ymmärrys työn alla olevasta pelistä. Jos säveltäjä on yrityksen ulkopuolelta palkattu urakoitsija, tulee hän yleensä mukaan projektiin vasta myöhemmässä vaiheessa. Talon sisäisellä säveltäjällä on ollut mahdollisuus olla mukana design-vaiheessa muiden tiimiläisten kanssa ja parempi käsitys työn alla olevasta projektista ja sen vaatimuksista. Säveltäjä saattaa työskennellä fyysisesti eri kohteessa kuin muut työntekijät, ja tällöin hänelle tulisi mahdollistaa pääsy projektiin liittyvään materiaaliin, jota käyttää lähteenä työnsä pohjalle. Säveltäjälle hyödyllistä materiaalia on esimerkiksi pelisuunnitteludokumentti (game design document tai GDD), konseptitaide, pelinsisäinen videokuva, peliin tuleva dialogi sekä asset-lista tarvittavasta musiikista. (Philips 2014, 117–128.)

Videokuva antaa säveltäjälle käsityksen, miten peli liikkuu ja toimii selkeämmin kuin pelkät kuvat ja teksti. Rytmii on yksi säveltäjän tärkeimmistä työkaluista, ja liikkuvasta kuvasta on hyötyä sen määrittämisessä. (Philips 2014, 118.)

Pelisuunnitteludokumentti on suunnitelma, jota työntekijät aikovat noudattaa peliä tehdessään. Sen tarkoitus on ilmaista visiota pelistä, sen sisällöstä ja toimia ohjenuorana pelin toteuttamiselle. Se toimii oppaana pelin eri osa-alueilla työskenteleville, jotta jokainen työntekijä olisi perillä siitä mitä, tehdään ja työskentelisi kohti yhteistä päämäärää. (Ryan 1999.) Koska pelisuunnitteludokumentille ei ole olemassa standardeja, jokainen suunnitelma on erilainen ja sisältää hieman eri asioita.

Yleisiä pelisuunnitteludokumentissa mainittavia asioita: (Mozolevskaya 2021.)

- Kuvaus pelistä. Pelimekaniikat ja säännöt. Muita samankaltaisia pelejä.
- Yleistä tietoa, kuten genre, kohdeyleisö ja alusta.
- Pelattavuus, esimerkiksi pelin päämäärä, kuinka pelin voittaa tai häviää ja kuinka peliä ohjataan.
- Pelin aputiedostot eli assetit. Mikä on pelin graafinen ilme? Asset-lista eli tarvittavat grafiikat, kuvat, äänet ja musiikki.

Pelin dialogi eli käsikirjoitus on säveltäjälle avuksi ymmärtämään pelin tarinaa ja hahmoja. Hahmojen ymmärtäminen syvemmin mahdollistaa myös hahmokohtaisen säveltämisen, jossa hahmolle luodaan oma motiivi eli musiikillinen idea tai teema eli musiikillinen kokonaisuus (Jyväskylän Lyseon Lukio 2021.) Dialogin avulla ymmärrys hahmosta ja hahmon roolista osana kokonaisuutta kasvaa ja sävellys osuu kohdalleen. (Philips 2014, 121–122).

Konseptitaide on pala projektin visuaalisesta kokonaisuudesta. Se kertoo säveltäjälle pelin tunnelmasta ja ilmeestä. Tuotantovaiheessa pelinsisäinen video saattaa vielä sisältää asetteja, jotka eivät päädy viimeiseen tuotteeseen tai visuaalinen ilme saattaa olla vielä vahvasti työn alla. Silloin konseptitaide on tarkoin kuva siitä, millaista maailmaa pelissä maalataan. Se esittää pelin hahmot viimeistä yksityiskohtaa myöten. Myös konseptitaide on erinomainen inspiraation lähde. (Philips 2014, 122–123.)

Asset-lista musiikista tai äänistä on oletus siitä, kuinka paljon erilaista ääntä peliin tarvitaan. Yleensä se sisältää kuvauksen tarvittavasta musiikista, musiikin tarvittavan pituuden ja tiedon siitä, mihin tilanteeseen musiikki tulee. Musiikin asset-lista on sitä hyödyllisempi, mitä enemmän tietoa se tarvittavista äänistä sisältää. Kuviossa 8 kuvataan esimerkki asset-listassa, joka on varusteltu kuvauksilla tarvittavista äänistä.

	A	B	C	D	E	F	G	H
1								
2	<i>for example Environment or cutscene, or UI</i>	<i>A title/brief description of the sound ie. Large snowball rolling</i>	<i>A more detailed description of the sound, for example: A snowball roughly twice as high as the main character. Should sound dangerous yet soft</i>	<i>Number of variation, footsteps for example would need a high number where as a UI button click may only need 1</i>	<i>Should the sound loop or not. Music for example often loops where as sfx often do not</i>	<i>Who is this particular sound asset assigned to</i>	<i>The due date of the sound</i>	<i>The exact file name of this sound for reference later in the development or for easy location by the supervisor</i>
3	Narrator	Narration	Audio recorded afterwards	1	No	Narrator	27/05/21	
4	SXF	Cigarettes burning	Soft sound	2-10	Y	Girl	27/05/21	
5	SFX	Stamp	A shoe crushing ends of a cigarette	1	N	Man	27/05/21	
6	Ambience	Market ambience	One shot, recorded on location, can be repeated	1	N		27/05/21	
7	SFX	Slapping skin	short bullet sound that can be rapidly repeated	1-2	N	Girl	27/05/21	
8	SFX	Newspaper	One shot, recorded afterwards, soft but attention calling	1-3	N	Man	27/05/21	
9		Rienzi : abertura em re major	Turns from diagetic to non diagetic	1	Yes	Radio	27/05/21	

KUVIO 8. Asset-lista (Haapakoski 2021f.)

Jos pelistä on jo olemassa pelattava versio, ajaa se videokuvan roolin paremmin. Jos pelattavaa versiota ei vielä ole, tulisi säveltäjän perehtyä siihen heti kun se on saatavilla. Pelin pelaaminen

itse varhaisessa vaiheessa on äärimmäisen tärkeää kokonaisuuden hahmottamiseksi ja spottaamiseksi, eli paikkojen löytämiseksi, jotka vaativat musiikkia. Sekä sen varmistamiseksi, että musiikki ei sodi muuta äänisuunnittelua vastaan. (Philips 2014, 126–128.)

## 8 ADAPTIIVISEN ÄÄNIMAAILMAN TOTEUTUS

Osana opinnäytetyötä toteutettiin pieni peli, jonka tarkoituksena oli esitellä sekä perinteisiä että luovempia tapoja käyttää adaptiivista musiikkia ja samalla hioa ja kehittää omaa tietotaitoa. Tässä luvussa on kuvattu pelimoottorin ulkopuolella tehty työ.

### 8.1 Suunnittelu ja sävellys

Pelin suunnittelu aloitettiin kirjoittamalla lyhyt pelisuunnitteludokumentti. Pelin mekaniikoita suunniteltaessa pohdittiin millaisia mahdollisuuksia ne soisivat adaptiivisen musiikin käyttöön. Päätettiin toteuttaa kaksiulotteinen ylhäältä päin kuvattu retrohenkisen bullet hell -peli velhoteemalla. Suunniteltiin, että adaptiivista musiikkia voitaisiin hyödyntää seuraavissa tilanteissa:

- Pelaajan tuhotessa monta vihollista lyhyen ajan sisään.
- Jos huoneessa on vihollisia, toistetaan taistelumusiikkia ja vaihdetaan se rauhallisempaan pelaajan ollessa yksin.
- Pelaajan käyttäessä erilaisia loitsuja.
- Vihollisten ilmestyessä jokainen toisi taustamusiikkiin oman instrumenttinsa.

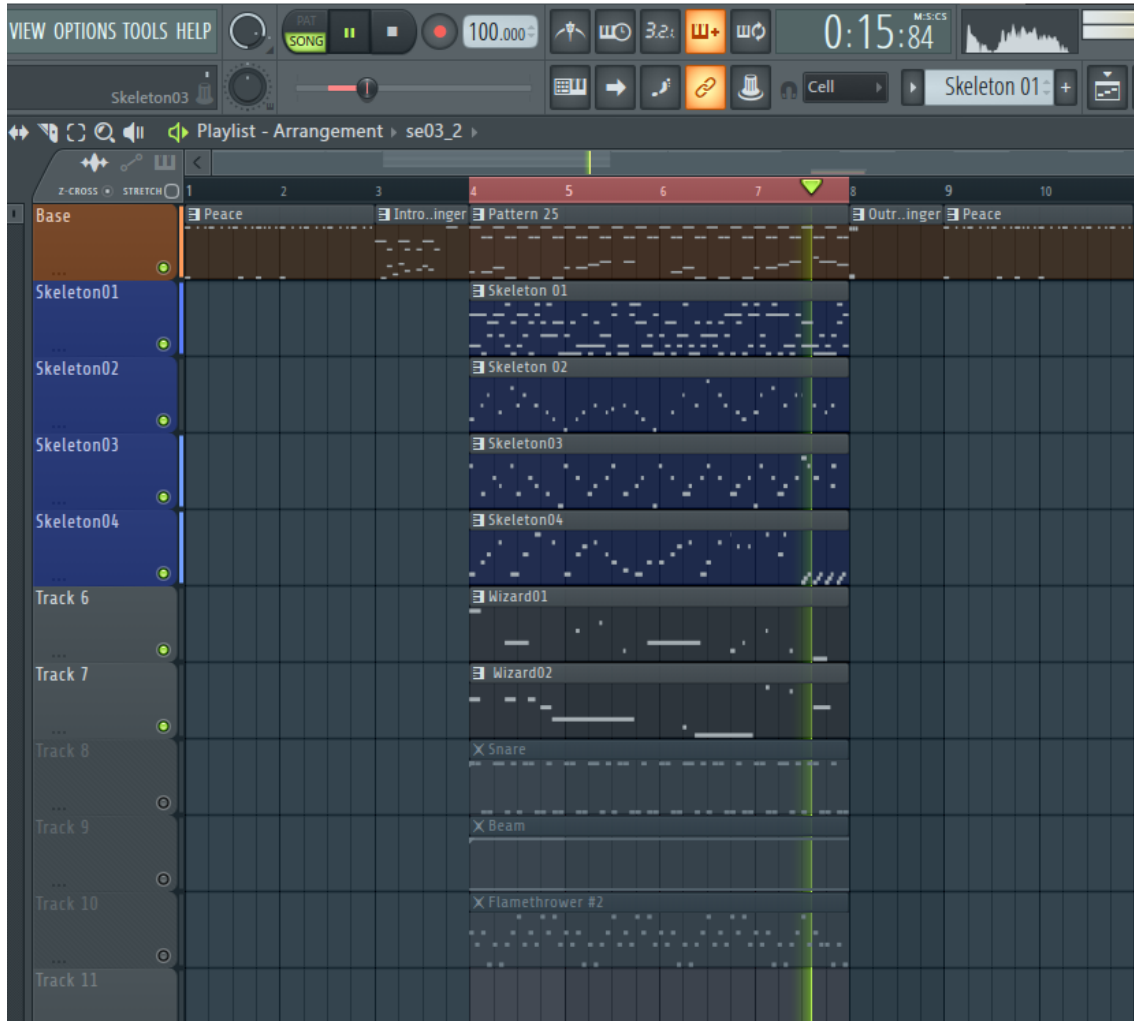
Lisäksi suunniteltiin Unityn sisäänrakennettuja ääniefektejä käytettäväksi dynaamisesti.

Aluksi laadittiin suunnitelma äänimaailmasta ja instrumentaatiosta. Koska pelin tyyliksi oli valittu retrohenkinen pikseligrafiikka, päätettiin että myös musiikki saisi olla kuin tuulahdus menneiltä vuosikymmeniltä, mutta ilman teknisiä rajoitteita. Niinpä yleistyökaluksi valikoitui FB-3100 virtuaalinen VSTI-syntetisaattori, joka mallintaa Korgin valmistamaa PS-3100-laitetta vuodelta 1977. Tällä syntetisaattorilla toteutettiin musiikin perkussiiviset eli rytmilliset osuudet, jotka tässä yhteydessä tarkoittavat rumpusetin osia, kuten basso- ja virvelirumpua sekä hi-hattia mallintavat äänet sekä bassokuvion ja velhojen melodiat.

Päätettiin, että pelin musiikki tulisi muodostumaan kahdesta osasta. Niin sanotusta taikomusiikista, jota toistetaan pelaajan ollessa yksin huoneessa sekä taistelumusiikista, jota toistetaan, kun huoneessa on vihollisia. Säveltäminen aloitettiin taistelumusiikista, sillä sen koettiin olevan tärkein osa peliä. Aloitettiin kirjoittamalla bassokuvio ja sen taustalle rytmiraita. Kuvion pituudeksi tuli neljä tahtia ja tempoksi valittiin sata iskua minuutissa, jota voidaan pitää keskinopeana. Tarkoituksena

oli, että musiikki ei olisi liian hätäisen kuuloista, mutta se ei saisi myöskään kuulostaa laahaavalta taistellessa. Bassokuvio implikoi, että sävellajiksi olisi tullut F-molli, mutta myöhemmässä vaiheessa soitettiin myös nuotteja sävellajin ulkopuolelta kaoottisuuden korostamiseksi. Molli mielletään usein surumieliseksi tai synkäksi. Tätä neljää tahtia tulitaisiin toistamaan uudestaan ja uudestaan niin kauan, kun pelaajalla olisi vastassa vihollisia.

Ensimmäinen peliin toteutettu vihollinen oli pelaajaa kohti juokseva luuranko. Peli toteutettiin siten, että huoneessa on yhtä aikaa maksimissaan neljä luurankoa, ja jokaista luurankoa varten luotiin oma ääniraita, jota toistettaisiin rumpujen ja basson muodostaman pohjaraidan kanssa niin kauan kuin luuranko säilyisi hengissä. Luurangot ovat usein yhdistetty populaarikulttuurissa ksylofoneihin, ja niinpä ksylofonin ääntä päätettiin käyttää myös tässä yhteydessä luurankojen ääniraidoissa. Toteutettiin siis yhteensä neljä hieman erilaista ääniraitaa ksylofonin ”kolistelua.” Raidat soitettiin talteen Midi-koskettimillam ja niitä toistamaan valittiin BBC Symphony Orchestra VSTI -instrumentti, jonka ksylofonin ääni kuulostaa ihan oikealta ksylofonilta luoden mielenkiintoisen kontrastin syntesaattoreiden hallitsemaan äänimaisemaan. Nuoteissa pysyttiin suurimmaksi osakseen sävellajissa mutta ei täysin. Lisäksi jokainen raita soitettiin eri rytmityksellä. Kun kaikki raidat soitetaan yhtä aikaa, on tuloksena hieman mielenvikainen ja kaoottinen ksylofonien sekamelska, joka sopii oivasti kuvastamaan pelaajaa kohti hoipertelevaa luurankolaumaa. Kokonaisuuden hahmottamiseksi luurankojen ääniraidat ja myös myöhemmin lisätyt raidat asetettiin DAW-ohjelmistossa aikajanelle päällekkäin. Näin niitä pystyttiin kuuntelemaan yhtä aikaa ja tarvittaessa vaimentamaan osa ääniraidoista tiettyjen yhdistelmien kuuntelemiseksi.



KUVIO 9. Näkymä projektin aikajanasta DAW-ohjelmistossa (Haapakoski 2021g.)

Kuviossa 9 on aseteltu mididataa sisältäviä kuvioita päällekkäin projektin aikajanalle. Ruskean väriselle raidalla "Base" on asetettu rummut ja basson sisältävä pohjaraita. Tahdit yksi ja kaksi kattava kuvio "Peace" sisältää taukomusiikin. Kolmas ja kahdeksas tahti ovat siirtymiä. Tahdistä neljä tahdin seitsemän loppuun asti on taistelumusiiikkia. Aikajanalle on asetettu päällekkäin jokaista vihollista vastaavat kuviot. Sinisellä värillä merkityt raidat sisältävät mididataa luurankojen raitojen toistamiseen ja niiden alla olevat raidat (Track) kuusi ja seitsemän sisältävät velhojen syntetisaattorimelodiat. Raidat kahdeksan, yhdeksän ja kymmenen ovat vaimennettuja. Ne sisältävät dataa äänistä, joita toistetaan pelaajan käyttäessä liekinheitinloitsua.

Varhaisessa vaiheessa huomattiin, että jos jokainen luuranko generoitiin huoneeseen yhtä aikaa, niin pelaajalle jää epäselväksi, että jokainen huoneessa oleva vihollinen vastaa yhtä taustalla soivaa soitinta. Yhtä aikaa generoitaessa ksylofonien kolina alkoi heti ja harveni sitä mukaa, kun pe-



laaja tuhosi luurankoja. Tämän kiertämiseksi luurangot koodattiin ilmestymään viiveellä. Luurankojen generoiminen annettiin tehtäväksi toiselle vihollistyyppille, ja ensimmäiset luurangot asetettiin ilmestymään satunnaisesti valittuun aikaan tietyn ikkunan sisällä. Näin luurangot ilmestyisivät hieman eri aikaan ja ksylofonien liittyminen taustamusiikkiin olisi pelaajalle hieman selvempi efekti.

Pelin toisen vihollistyyppin, Velhon, ääneksi päätettiin valita taas jo aikaisemmin pohjaraitaa tehdessä käytetty FB-3100-syntetisaattori. Velhoja pelissä on huoneessa kerrallaan maksimissaan kaksi, joten tarvittiin kaksi eri ääniraitaa. Velhoille sävellettiin omat melodiat, jotka kierron alussa ja lopussa harmonisoivat toistensa kanssa ja kierron keskellä käyvät ikään kuin vuoropuhelua. Luurankojen ääniraitoihin verrattuna velhojen raidat tehtiin huomattavasti rauhallisemmiksi ja kevyemmiksi. Kuviossa 10 on nuotitettuna velhojen melodiat. Huomaa etenkin harmonia ja vuoropuhelu.

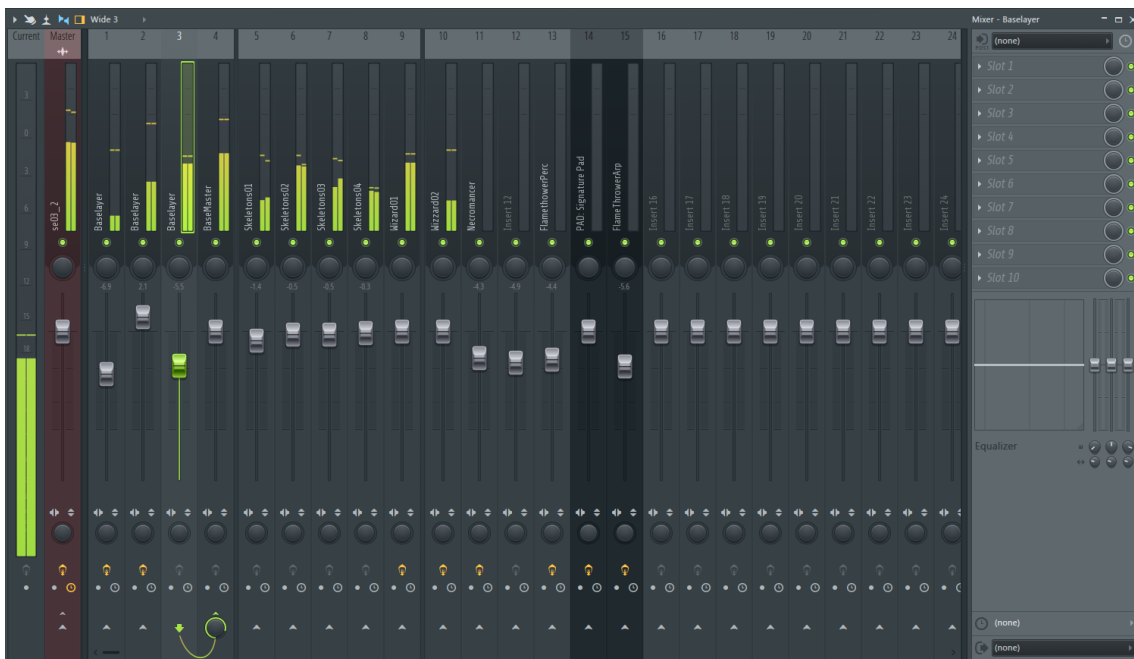


KUVIO 10. Velhojen melodiat (Haapakoski 2021h.)

Peliin on toteutettu erilaisia loitsuja, joita pelaaja voi käyttää vihollisia vastaan. Yksi näistä loitsuista toimii kuin liekinheitin, ja sitä käytetään pitämällä hiiren nappia pohjassa. Tätä loitsua varten sävellettiin syntetisaattorilla aggressiivisen kuuloinen kuudestoistaosanuotteja soittava murtosointu, jota toistetaan niin kauan kuin pelaaja hyökkäykseen käytettävää nappia pohjassa. Lisäksi loitsua varten toteutettiin syntetisaattorilla ääniefekti, joka kuulosti yskien käyntiin kiihtyvältä moottorilta. Tämä efekti toistettaisiin aina kerran, kun pelaaja aloittaa loitsun käytön. Murtosointu tarkoittaa sitä, että soinnun ääniä ei soiteta yhtä aikaa, vaan peräkkäin. Lopuksi päätettiin, että loitsu tarvitsisi vielä *voimaa*, ja murtosoinnun lisäksi taustalle lisättiin ääniraita virvelirummun pauketta. Loitsua käytettäessä soi nyt kaksi uutta ääniraitaa. Jotta musiikki ei menisi liian sekamelskaiseksi, päätettiin miksausta säätää loitsun aikana niin, että vihollisia vastaavat ääniraidat vaimenisivat ja hukkuisivat liekinheittimen ääniraitojen alle.

Taistelumusiikin lisäksi peliin toteutettiin kahden tahdin mittainen pätkä taukomusiikkia, jota toistetaan pelaajan ollessa huoneessa yksin ilman vihollisia. Taukomusiikki pidettiin hyvin sisällöllisesti yksinkertaisena ja kevyenä, jotta kontrasti taistelumusiikkiin olisi tarpeeksi selvä. Taukomusiikista on myös helpompi ja saumattomampi siirtyä taistelumusiikkiin missä tahansa vaiheessa, jos taukomusiikissa ei tapahdu liikaa. Taukomusiikista taistelumusiikkiin ja siitä takaisin taukomusiikkiin siirtymistä varten luotiin kaksi erilaista stinger-siirtymää. Molemmat siirtymät tehtiin yhden tahdin mittaisiksi. Taistelumusiikkia edeltävää siirtymää päätettiin kutsua introksi ja taistelumusiikin jälkeen toistettavaa siirtymää outroksi. Koska intro-siirtymän tehtävä on johdattaa taistelumusiikkiin, sävellettiin se tarkoituksena nostattamaan tunnelmaa. Tämä päätettiin toteuttaa yksinkertaisella rumpufillillä ja bassoriffillä. Taistelun päättyessä toistettavan outro-siirtymän tehtävä on rauhoittaa tilanne. Siksi se koostuikin vain yhdestä bassorummun iskusta ja ääniefektistä ja kolmen neljäs-osanuotin mittaisesta tauosta. Tämän jälkeen siirrytään takaisin taukomusiikkiin.

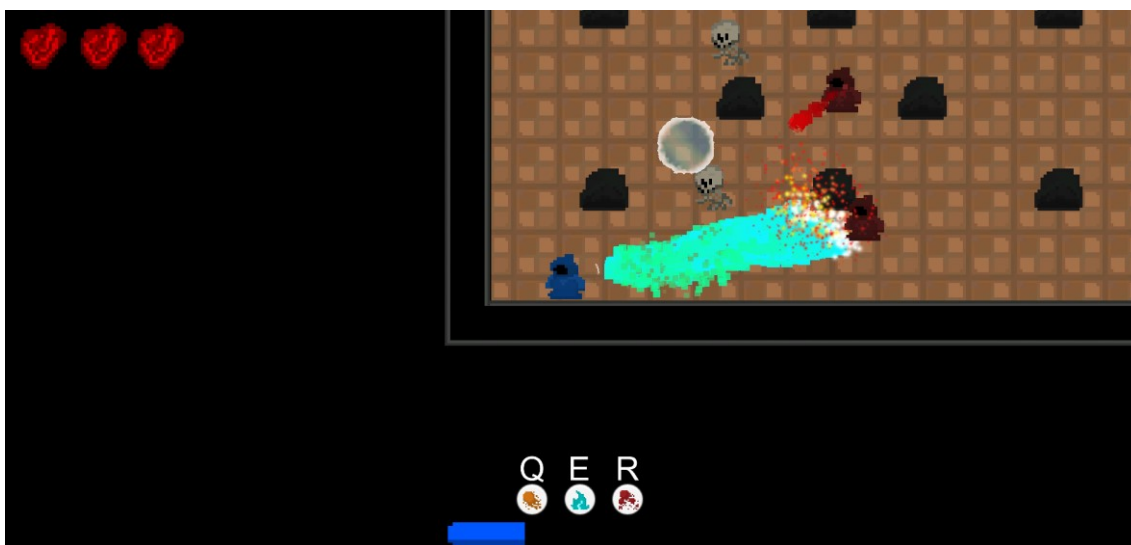
Pelaajan tuhotessa vastustajia osa niistä tiputtaa erilaisia esineitä pelaajan poimittavaksi. Yksi näistä esineistä on power-up, joka antaa pelaajalle muutamaksi sekunniksi suojakilven. Tehostaakseen vaikutelmaa että pelaaja on fyysisesti kuplan sisällä, päätettiin mikserin master-kanavalle asettaa low-pass filter eli alipäästösuodatin, joka päästää tietyn ennalta määritetyn taajuuden alle jäävät äänet läpi ja vaimentaa sen yläpuolelle jääviä taajuuksia. Näin luotiin efekti, joka kuulostaa siltä, kuin musiikki kuuluisi seinän takaa.



KUVIO 11. Projektin mikserinäkömä DAW-ohjelmistossa (Haapakoski 2021i.)

Kuviossa 11 jokainen ääntä tuottava soitin tai raita on jaettu omalle mikserikanavalle. Nämä kana-  
vat vietiin DAW-ohjelmistosta ulos jokainen omaksi WAV-äänitiedostoksensa Unitya varten.

Kun pelin äänimaailma alkoi muodostua, pelin visuaalinen ilme, joka tässä vaiheessa koostui vain  
placeholder-grafiikasta, kaipasi kipeästi päivitystä. Grafiikka päivitettiin pelisuunnitteludokumentin  
mukaan koostumaan 16 x 16 pikselin kuvista, jotka suunniteltiin ja toteutettiin itse. Kuviossa 12  
pelaaja (sininen hahmo) käyttää loitsua vihollisia vastaan.



KUVIO 12. Kuvankaappaus pelitilasta (Haapakoski 2021j.)

## 8.2 Toteutus Unityssa

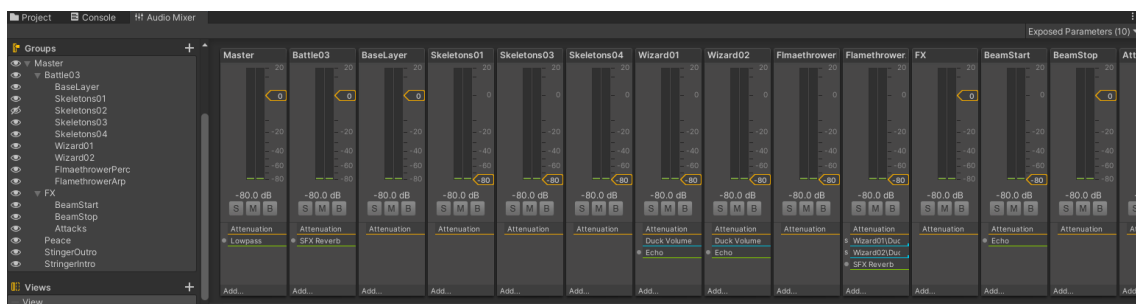
Aluksi luotiin jokaiselle toistettavalle ääniraidalle hierarkiaan omat peliobjektinsa. Kuviossa 13 nä-  
kyy, kuinka järjestyksen ylläpitämiseksi musiikista vastaavat peliobjektit järjestettiin AudioManager-  
peliobjektin alle ja taistelumusiikin ääniraidat asetettiin Battle03-nimisen peliobjektin alle. Peliob-  
jektit nimettiin ääniraitojen mukaan, ja niille annettiin Audio Source -komponentit, joihin asetettiin  
vastaavat äänitiedostot. Koska taukomusiikki ja siirtymät koostuivat vain yhdestä ääniraidasta, ne  
tarvitsivat jokainen vain yhden oman peliobjektin. Jokaiselle Audio Sourcelle asetettiin inspectorissa  
komponentin asetuksista totuusarvomuuttuja Play On Awake todeksi, jolloin ne toistavat niille osoi-  
tetun äänen peliobjektin aktivoitua. Ääniraidat asetettiin siirtymiä lukuun ottamatta loop-asetukselle.  
Pelin alussa vain taukomusiikista vastaava peliobjekti asetettiin aktiiviseksi. Näin taukomusiikki  
toistuu ensin pelin käynnistyessä.



KUVIO 13. Kuvankaappaus Unityn scene-näkymän hierarkiasta (Haapakoski 2021k.)

Pelin kulkua ohjaamaan tehtiin GameManager-skripti, jonka tehtäväksi osoitettiin myös musiikin ohjaus. Taukomusiikista taistelumusiikin ja takaisin siirtymisessä käytettiin vaakasuoraa sekvensointia. Siirtyminen tauko- ja taistelumusiikin välillä toteutettiin koodissa vuorottaisrutiineja käyttäen. Pelaajan astuessa pelissä portaaliin, joka aloittaa uuden erän, kutsuttiin vuorottaisrutiini, joka lopettaa taukomusiikin toistamisen asettamalla taukomusiikista vastaavan peliobjektin epäaktiiviseksi ja aktivoi samalla siirtymästä vastaavan peliobjektin. 2,4 sekunnin päästä siirtymä on soinnut loppuun ja vuorottaisrutiini asettaa aktiiviseksi Battle03-peliobjektin, jolloin kaikki sen alaisuudessa olevat peliobjektit alkavat toistamaan ääntä. Samalla tavoin asettamalla peliobjekteja järjestyksessä aktiiviseksi ja epäaktiiviseksi siirrytään myös taistelumusiikista taukomusiikkiin.

Ääniraitojen hallintaan luotiin Audio Mixer -asetti. Jokaiselle aiemmin luodulle Audio Source -komponentille osoitettiin ulostuloksi mikseristä oma kanava. Musiikin lisäksi mikseriin luotiin kanava, jonka kautta toistettiin pelin ääniefektit. Alla olevasta kuvioista 14 näkyy pelissä käytetyn mikserin hierarkia.



KUVIO 14. Näkymä Unityn sisältä projektissa käytetystä Audio Mixeristä (Haapakoski 2021l.)

Pelin adaptiivinen taistelumusiikki toteutettiin vertikaalista reorkestraatiota käyttäen. Taistelun alkaessa jokaista taistelumusiikin kanavaa aletaan toistamaan, mutta vain pohjaraidan äänenvoimakkuus on säädetty niin, että raita on kuultavissa. Vaikka muutkin taisteluun liittyvät kanavat soivat, niiden äänenvoimakkuus on asetettu minimiin. Pelinaikaista dynaamista miksaamista varten käytettiin skriptiä, jolle syötettiin argumentteina mikseri, säädettävä exposed parameter, haluttu äänenvoimakkuus ja häivytyksen kesto. Skripti asetettiin julkiseen staattiseen luokkaan, jotta sitä voitaisiin kutsua koodilla helposti muista skripteistä. Jokaisen taistelumusiikin kanavan äänenvoimakkuuden säätäminen koodilla mahdollistettiin käyttämällä expose to code -toimintoa. Myös masterkanavalle asetettua alipäästösuodatinta, jota käytettiin pelaajan ollessa suojakuplan sisällä, ohjattiin exposed parametrien avulla. Kirjoitettiin skripti, jonka avulla jokaisella viholliselle osoitettiin oma kanava mikseristä, kiinnitettiin se komponenttina jokaiseen vihollis-prefabiin. Tämä skripti kutsui yllä mainittua julkista skriptiä vihollisen generoituessa huoneeseen ja asetti viholliselle osoitetun kanavan äänenvoimakkuuden nopealla häivytyksellä kuultavalle tasolle ja vihollisen kuollessa tiiputti äänenvoimakkuuden takaisin alas kuulemattomiin.

```
6 public static class FadeMixerGroup {
7
8     public static IEnumerator StartFade(AudioMixer audioMixer, string exposedParam, float duration, float targetVolume)
9     {
10         float currentTime = 0;
11         float currentVol;
12         audioMixer.GetFloat(exposedParam, out currentVol);
13         currentVol = Mathf.Pow(10, currentVol / 20);
14         float targetValue = Mathf.Clamp(targetVolume, 0.0001f, 1);
15
16         while (currentTime < duration)
17         {
18             currentTime += Time.deltaTime;
19             float newVol = Mathf.Lerp(currentVol, targetValue, currentTime / duration);
20             audioMixer.SetFloat(exposedParam, Mathf.Log10(newVol) * 20);
21             yield return null;
22         }
23         yield break;
24     }
25 }
```

*KUVIO 15. Skripti, jolla muutettiin kanavien parametrejä, kuten äänenvoimakkuutta pelin aikana (Haapakoski 2021m.)*

Vertikaalia reharmonisaatiota käytettiin myös pelaajan käyttäessä liekinheitinloitsua. Loitsun käyttäminen nosti loitsulle osoitetun kanavan äänenvoimakkuutta. Jotta liekinheittimen ääni korostuisi,

asetettiin liekinheittimen äänen kanavalle Send-efekti, jonka kautta signaali lähetettiin velhojen äänestä vastaaville kanaville, joille asetettiin Duck Volume-efekti, joka tiputti niiden äänenvoimakkuuden melkein kuulemattomiin liekinheitinkanavan soidessa.

Koska DAW-ohjelmistossa ääniraitoja luodessa äänet oli jätetty tarkoituksellisesti kuivaksi, eli ne olivat vailla efektejä, päätettiin niitä lisätä tässä vaiheessa Unityn sisäisesti. Erityisesti liekinheittimen kanava kaipasi hieman huonekaikua, jotta sen ääni saisi hieman ”hääntää” kun liekinheittimen ääni itsessään sammuu. Velhojen melodioista vastaaville kanaville asetettiin myös hyvin pienet Echo-efektit, jotta ne eivät kuulostaisi niin kaksiulotteiselta. Lopuksi taistelumusiikin master-kanavalle, joka kuviossa 4 on nimetty Battle03:ksi, asetettiin huonekaikua mallintava reverb-efekti, jota miksatessa usein käytetään ikään kuin ”liimaamaan” kokonaisuus kasaan, jotta äänimaailma kuulostaisi yhtenäisemmältä.

## 9 POHDINTA

Adaptiivisten äänimaailmojen suunnittelu ja toteutus osoittautui hyvinkin suureksi kokonaisuudeksi ja koen, että tässä opinnäytetyössä vasta raapaistiin pintaa. Työ oli hyvinkin opettava ja hieman uutena asiana itselleni tuli varsinkin sävellystyötä edeltävät design-vaiheet ja projektin hallintaan liittyvät asiat, kuten pelisuunnitteludokumentit ja asset-listat. Vaikka olin jo aiemmin toteuttanut adaptiivista ääntä Unityssa, en koskaan aikaisemmin näin paneutuvasti tai projektissa, jossa ääni oli näin suuressa roolissa.

Peli saatiin valmiiksi ja toimivaksi ja suurilta bugeilta vältyttiin. Vaikka pelillisesti ja sisällön kannalta sitä voisikin ehkä kuvailla hieman köyhäksi, koen että se demonstroi adaptiivisen musiikin käyttöä onnistuneesti. Vaikka peli ei myöskään musiikillisesti ole mikään mestariteos ja ääntä ei yhteen laskettuna ole kovinkaan montaa sekuntia, miellän äänimaailman onnistuneeksi ja peliin sopivaksi.

Koen, että taitojeni kehittymisestä äänen hallinnasta pelimoottorin tasolla on hyötyä, kun aion seuraavaksi perehtyä tarkemmin audio middleware -ohjelmistoihin, joiden koen olevan seuraava looginen askel kehittäessäni itseäni pelien äänimaailmojen saralla.

## LÄHTEET

Arqen 2021. How to Create a Killer Sweet Spot in Your Room. Hakupäivä 23.5.2021. <http://arqen.com/acoustics-101/reflection-free-zone/>

Barton, Matt & Loquidice, Bill 2007. A History of Gaming Platforms: The Commodore 64. Hakupäivä 20.5.2021. [https://www.gamasutra.com/view/feature/130406/a\\_history\\_of\\_gaming\\_platforms\\_the\\_.php](https://www.gamasutra.com/view/feature/130406/a_history_of_gaming_platforms_the_.php)

Brown, Yannis 2021. About Audio Middleware. Hakupäivä 23.5.2021. <https://www.yannisbrown.com/about-audio-middleware/>

Campbell, Bruce Donald 2016. History of Video Game Music. Hakupäivä 19.5.2021. <http://bdcampbell.net/music/>

Cohen, D.S 2021. The History of Classic Video Games. Hakupäivä 23.5.2021. <https://www.lifewire.com/the-cd-rom-revolution-729750>

Computer Music Specials 2009. The beginner's guide to: MIDI controllers. Hakupäivä 23.5.2021. <https://www.musicradar.com/tuition/tech/the-beginners-guide-to-midi-controllers-179018>

Crute, Adam 2019. Learning the basics of FM synthesis and how it works. Hakupäivä 22.5.2021. <https://www.musictech.net/guides/essential-guide/how-fm-synthesis-works/>

Driscoll, Kevin & Diaz, Joshua 2009. Endless loop: A brief history of chiptunes. Hakupäivä 19.5.2021. <https://journal.transformativeworks.org/index.php/twc/article/view/96/94>

explod2A03 2012. NES Audio: Brief Explanation of Sound Channels. Hakupäivä 22.5.2021. <https://www.youtube.com/watch?v=la3coK5pq5w>

Game Maker's Toolkit 2014. Adaptive Soundtracks in Games. Hakupäivä 23.5.2021. [https://www.youtube.com/watch?v=b0gvM4q2hdl&t=42s&ab\\_channel=GameMaker%27sToolkit](https://www.youtube.com/watch?v=b0gvM4q2hdl&t=42s&ab_channel=GameMaker%27sToolkit)



Haapakoski, Niko 2021a. Kuvankaappaus. Tux Guitar -sovellus. Hakupäivä 10.6.2021.

Haapakoski, Niko 2021b. Kuvankaappaus. FL Studio -sovellus. Piano roll -näkyvä. Hakupäivä 10.6.2021.

Haapakoski, Niko 2021c. Kuvankaappaus. FL Studio -sovellus. Playlist-näkyvä. Hakupäivä 10.6.2021.

Haapakoski, Niko 2021d. Valokuva. MIDI-ohjain. Hakupäivä 10.6.2021.

Haapakoski, Niko 2021e. Kuvankaappaus. Audio Source -komponentti. Unity. Hakupäivä 10.6.2021.

Haapakoski, Niko 2021f. Kuvankaappaus. Bleeps 'n' Bloops Audio Asset List. Google Sheets -dokumentti. Hakupäivä 10.6.2021.

Haapakoski, Niko 2021g. Kuvankaappaus. FL Studio -sovellus. Opinnäytetyöhön liittyvää sävellystä. Hakupäivä 10.6.2021.

Haapakoski, Niko 2021h. Kuvankaappaus. MuseScore3 -sovellus. Opinnäytetyöhön sävellettyä musiikkia. Hakupäivä 10.6.2021.

Haapakoski, Niko 2021i. Kuvankaappaus. FL Studio -sovellus. Mikseri. Hakupäivä 10.6.2021.

Haapakoski, Niko 2021j. Kuvankaappaus. Opinnäytetyötä varten kehitetty peli. Hakupäivä 10.6.2021.

Haapakoski, Niko 2021k. Kuvankaappaus. Unityn käyttöliittymä. Hakupäivä 10.6.2021.

Haapakoski, Niko 2021l. Kuvankaappaus. Audio Mixer -asetti. Unity Hakupäivä 10.6.2021.

Haapakoski, Niko 2021m. Kuvankaappaus. Opinnäytetyön pelissä käytettyä koodia. Visual Studio Code. Hakupäivä 10.6.2021.

Hahn, Michael 2020. What is MIDI? How To Use the Most Powerful Tool in Music. Hakupäivä 23.5.2021. <https://blog.landr.com/what-is-midi/>

Jyväskylän Lyseon Lukio 2021. Mitä on melodia? Hakupäivä 25.5.2021. <https://peda.net/jao/lyseo/opiskelu2/ojkuo/tjt/musiikki/mu1-luonnos/mu1-kale/tuntien-aiheet/mp/mlk/mit%C3%A4-on-melodia>

Kivelä, Heikki 2020. Opas: Cd-levyn historia ja tekniikka – Täydellistä ääntä, ikuisesti. Hakupäivä 23.5.2021. <https://avplus.fi/opas-cd-levyn-historia-ja-tekniikka-taydellista-aanta-ikuisesti/>

Library of Congress 2019. A Brief History of Video Game Music. Hakupäivä 19.5.2021. <https://www.youtube.com/watch?v=eSSqgeMkuJM>

McAllister, Max 2018. Considerations for Mixing with Headphones vs Monitors. Hakupäivä 23.5.2021. <https://producelikeapro.com/blog/mixing-with-headphones-vs-monitors/>

Mozolevskaya, Victoria 2021. What is game design or welcome to the game designers backstage: rates, mistakes and victory conditions [+ detailed game design document template]. Hakupäivä 25.5.2021. <https://kevurugames.com/blog/what-is-game-design-or-welcome-to-the-game-designers-backstage-rates-mistakes-and-victory-conditions-detailed-game-design-document-template/>

Murphy, Caleb J. 2020. What Is a DAW (And What Can You Do With It)? Hakupäivä 23.5.2021. <https://www.careersinmusic.com/what-is-a-daw/>

Northfield, Rebecca 2018. Gaming's golden age: top 10 retro-vintage arcade classics. Hakupäivä 19.5.2021. <https://eandt.theiet.org/content/articles/2018/07/gaming-s-golden-age-top-10-retro-vintage-arcade-classics/>

O'Connor, Mike 2019. Plugin Formats: Differences between VST, VST3, AU, AAX, RTS, TDM. Hakupäivä 31.5.2021. <https://www.electronicdrumadvisor.com/plugin-formats-differences-between-vst-vst3-au-aax-rtas-tdm/>

Phillips, Winifred 2014. A Composer's Guide to Game Music. Cambridge, MA: MIT Press. Hakupäivä 25.5.2021. Kindle. Vaatii käyttöoikeuden.

Recording Connection 2021. What Are Digital Audio Workstations (DAW)?. Hakupäivä 23.5.2021.  
<https://www.recordingconnection.com/what-are-digital-audio-workstations-daw/>

Robjohns, Hugh 2006. Q. Why can't I do all my mixing on headphones? Hakupäivä 23.5.2021.  
<https://www.soundonsound.com/sound-advice/q-why-cant-do-all-my-mixing-headphones>

Ryan, Tim 1999. The Anatomy of a Design Document, Part 1. Documentation Guidelines for the Game Concept and Proposal. Hakupäivä 25.5.2021. [https://www.gamasutra.com/view/feature/131791/the\\_anatomy\\_of\\_a\\_design\\_document\\_.php](https://www.gamasutra.com/view/feature/131791/the_anatomy_of_a_design_document_.php)

Unity 2021a. Unity Documentation. Audio Source. Hakupäivä 29.5.2021.  
<https://docs.unity3d.com/Manual/class-AudioSource.html>

Unity 2021b. Unity Documentation. Audio Listener. Hakupäivä 29.5.2021.  
<https://docs.unity3d.com/Manual/class-AudioListener.html>

Unity 2021c. Unity Documentation. Audio Mixer. Hakupäivä 29.5.2021.  
<https://docs.unity3d.com/Manual/AudioMixer.html>

Unity 2021d. Unity Documentation. AudioGroup Inspector. Hakupäivä 29.5.2021.  
<https://docs.unity3d.com/Manual/AudioMixerInspectors.html>

Varney, Kit 2018. Non-Linear Soundtracks: Horizontal and Vertical Techniques. Hakupäivä 23.5.2021. <https://kitvarneycreativeblog.wordpress.com/2018/11/29/none-linear-soundtracks-horizontal-and-vertical-techniques/>

Vincent, Robin 2019. What Is a VST? All Your Questions Answered. Hakupäivä 23.5.2021.  
<https://www.careersinmusic.com/what-is-a-vst/>

Winer, Ethan 2021. RealTraps – How To Set Up a Listening Room. Optimizing placements. Hakupäivä 23.5.2021. [https://realtraps.com/art\\_room-setup.htm](https://realtraps.com/art_room-setup.htm)

Wreglesworth, Rob 2021. Studio monitors Vs Regular Speakers: There's a Big Difference. Hakupäivä 23.5.2021. <https://musicianshq.com/studio-monitors-vs-regular-speakers/>

Yamaha Design Laboratory 2021. DX7. Hakupäivä 23.5.2021. [https://www.yamaha.com/en/about/design/synapses/id\\_009/](https://www.yamaha.com/en/about/design/synapses/id_009/)

Zaarin 2021. LucasArt's Secret History #6: Monkey Island 2: LeChuck's Revenge. The Music of LeChuck's Revenge. Hakupäivä 23.5.2021. <https://mixnmojo.com/features/sitefeatures/LucasArts-Secret-History-Monkey-Island-2-LeChucks-Revenge/9>

Zehnder & Lipscomb 2006, 241. Playing Video Games. Motives, Responses, Consequences. New York: Routledge.