



TAMPEREEN
AMMATTIKORKEAKOULU

MAX VERKKO-OPPIMATERIAALIN LUONTI

Severi Kärki

Opinnäytetyö
Toukokuu 2018
Musiikin koulutusohjelma
Musiikkiteknologiapedagogi



TIIVISTELMÄ

Tampereen ammattikorkeakoulu
Musiikin koulutusohjelma
Musiikkiteknologiapedagogi

KÄRKI SEVERI:

Max verkko-oppimateriaalin luonti

Opinnäytetyö 42 sivua, joista liitteitä 7 sivua
Toukokuu 2018

Tämä opinnäytetyö käsittelee prosessia, jonka lopputuloksena syntyi verkko-oppimateriaali ohjelmoinnista Cycling '74:n Max-ohjelmalla. Opinnäytetyön tavoitteena oli täydentää Max-ohjelmoinnin vähäistä verkko-oppimateriaalia suomenkielisellä materiaalilla. Opinnäytetyö koostuu kirjallisesta osiosta ja verkko-oppimateriaalista.

Opinnäytetyön kirjallisen osion alussa käsitellään Max-ohjelmoinnin perusteita ja oppimateriaalin formaattia. Seuraavaksi käydään läpi musiikin ohjelmoinnin historiaa ohjelmointikielten näkökulmasta. Sen jälkeen paneudutaan verkkosivujen luontiin ja yhden oppimateriaaliin sisältyvän Max-ohjelmointiprojektin tekemiseen. Lopuksi käydään läpi testiryhmältä kerätty palaute ja sen pohjalta oppimateriaaliin tehdyt korjaukset.

Opinnäytetyön toinen osa, varsinainen oppimateriaali, löytyy verkkosivuilta osoitteesta <http://max-ja-msp.webflow.io>. Oppimateriaali koostuu Max-ohjelmointiympäristön perusteista sekä kuudesta ohjelmointia opettavasta projektista. Oppimateriaali on koottu siten, että kuka tahansa aiheesta kiinnostunut voi itsenäisesti käyttää oppimateriaalia hyödykseen Max-ohjelmointia opiskellessaan. Opiskelu vaatii vain tietokoneen ja Max-ohjelman joko kokonaisena tai ilmaisena demo-versiona. Materiaali itsessään on täysin suomenkielinen ja ilmainen, sekä suunniteltu käytettäväksi myös opetuksen tukena.

Asiasanat: Max, verkko-oppimateriaali, www-sivut, musiikin ohjelmointi, ohjelmointikieliset

ABSTRACT

Tampereen ammattikorkeakoulu
Tampere University of Applied Sciences
Degree Programme in Music
Pedagogue in Music Technology

KÄRKI SEVERI:

Creation of Max instructional web pages

Bachelor's thesis 42 pages, appendices 7 pages
May 2018

This bachelor's thesis reviews a process of creating instructional web pages of programming with Cycling '74's Max software. The main objective of this thesis was to response to the shortage of Max learning material in Finnish. Thesis consists of textual part and web pages.

The beginning of the textual part covers the basics of Max programming and the format of learning material. The next section covers music programming history based on programming languages. The third section covers the creation of web pages and the creation of one programming project included in the learning material. Finally, feedback and improvements based on it are processed.

The second part of this thesis is the actual learning material, which can be found under the link <http://max-ja-msp.webflow.io>. Web pages consists of the basics of Max programming and 6 programming projects. The material is made for self-learning for everybody, who wants to learn Max programming. Studying this needs only a computer and Max software (demo or full version). The learning material is completely in Finnish and free of charge, and also suitable for teachers to aid Max teaching.

Key words: Max, e-learning material, web pages, music programming, programming languages

SISÄLLYS

1	JOHDANTO.....	5
2	TAUSTOITUS	6
2.1	Mikä on Max?.....	6
2.2	Miksi juuri Max?	7
2.2.1	Ohjelmointi Maxilla.....	8
2.2.2	Oma historiani Maxin parissa	11
2.3	Miksi verkkosivut?	11
2.4	Esikuvat ja vertailu	13
2.5	Palautteen keruu	14
3	MUSIIKIN OHJELMOINTIKIELTEN HISTORIA	17
3.1	Maxin historia.....	17
3.2	Muita ohjelmointiympäristöjä	18
4	TOTEUTUS	21
4.1	Verkkosivuston koonti.....	22
4.1.1	WebFlow.....	22
4.1.2	Ulkoasu	23
4.2	Sekvensseri-projektin työstö vaiheittain.....	23
4.2.1	Materiaalin laatiminen	23
4.2.2	Oppimateriaali verkkosivuille.....	27
4.2.3	Tehtävän laatiminen ja viimeistely	28
5	PALAUTE JA KEHITTÄMINEN.....	30
6	POHDINTA	33
	LÄHTEET.....	34
	LIITTEET	36
	Liite 1. Max-oppimateriaalin testiryhmän kyselyn tulokset.....	36

1 JOHDANTO

Opinnäytetyöni tarkoitus on kertoa musiikin ohjelmointiin suunnatun Max-ohjelman oppimateriaalin tekoprosessista. Koen Max-ohjelmasta tehdyn oppimateriaalin olevan kovin vähäistä, joten haluan oppimateriaalini avulla auttaa asiasta kiinnostuneita päästäkseen Max-ohjelmoinnissa alkuun.

Max valikoitui oppimateriaalini aiheeksi, koska se on mielestäni musiikin ohjelmoinnissa selkein, laajimmin käytössä oleva ja parhaiten ylläpidetty ohjelmointialusta. Keskityn opinnäytetyössäni Maxin äänisignaalin sekä numerodatan prosessointiin, sillä työssäni ei ole tarkoitus käsitellä kaikkia Maxin ominaisuuksia. Käsittelemieni ominaisuuksien kautta oppija saa kattavan käsityksen Maxin toiminnasta.

Opinnäytetyöni koostuu verkkosivuista, jossa varsinainen oppimateriaali on, ja kirjallisesta osiosta. Kirjallisen osuuden tavoitteena on perehdyttää lukija oppimateriaalin tekemisen vaiheisiin. Kirjallisessa osuudessa käsitellään hieman musiikin ohjelmoinnin historiaa, prosessissa käytettyjä työvälineitä ja yhden projektin teko yksityiskohtaisesti. Lisäksi lopussa kerron testiryhmältä keräämästäni palautteesta ja sen myötä tehdyistä parannuksista.

En ole ennen opinnäytetyötäni tehnyt verkkosivuja tai oppimateriaalia näin laajassa mittakaavassa, mutta koen molempien taitojen olevan tärkeitä musiikkiteknologian opettajalle. Oppimateriaali sijaitsee osoitteessa <http://max-ja-msp.webflow.io>.

Oppimateriaalin tarkoitus on myös toimia Max-ohjelmoinnin opetuksessa apuvälineenä. Itse Max-ohjelmointia yhden lukuvuoden opettaneena olen huomannut aiheesta löytyvän hyvin niukasti oppimateriaalia. Tehtäviä ei ole juurikaan saatavilla, joten korjaan tämän ongelmatilanteen opinnäytetyössäni.

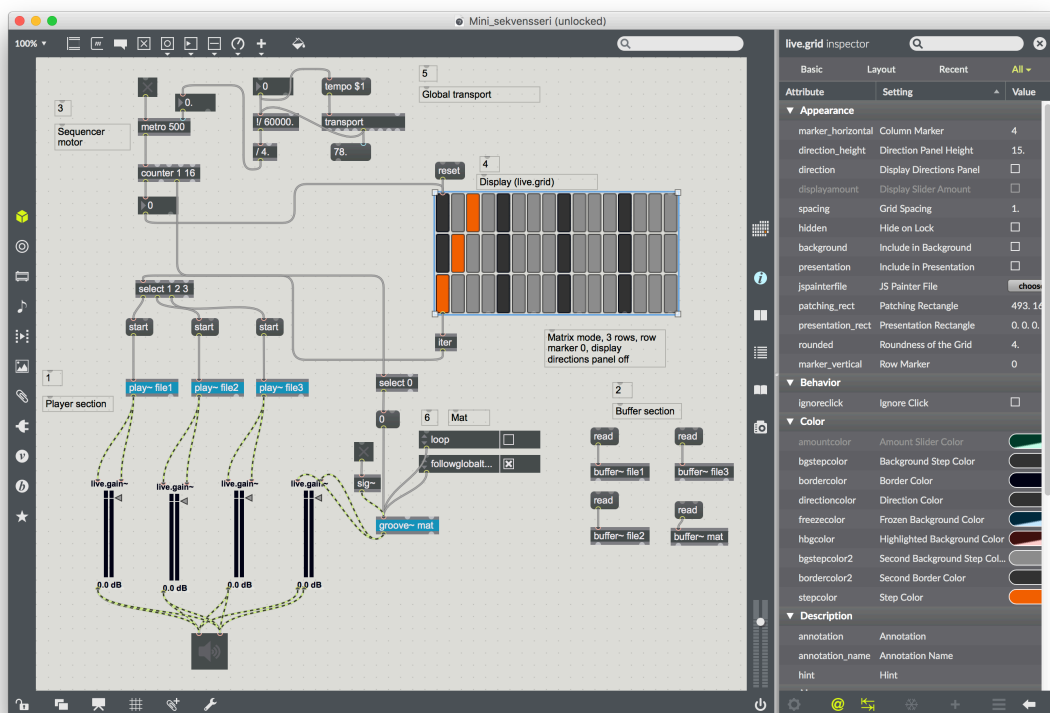
Olen ollut jo pitkään kiinnostunut musiikin ohjelmoinnista, koska se rohkaisee ajattelemaan laatikon ulkopuolelta. Max-ohjelma on ollut minulla käytössä vuodesta 2014 saakka, jona aikana olen ehtinyt vasta raapaista pintaa. On hyvin vapauttavaa, kun voi rakentaa itse efektin tai soittimen juuri sellaiseksi, kuin sen itse haluaa.

2 TAUSTOITUS

2.1 Mikä on Max?

Max on musiikkiteknologian saralla todella merkittävä ohjelma, jolla on historiaa jo vuosikymmenien verran. Alun perin Pariisissa 1980-luvulla kehitetty Max-ohjelma on laajentunut nykyiseen muotoonsa monimuotoiseksi äänen ja multimedian ohjelmointiympäristöksi. (Cycling '74 2007.) Max on monen musiikin ja multimedian ohjelmoijan tärkeä työkalu, jolla voi toteuttaa monimutkaisiakin interaktiivisia multimediaesityksiä.

Max on omissa luokissaan multimedian ohjelmointiympäristönä yksi suosituimmista ohjelmista. Muita samantapaisia ohjelmia ovat mm. PureData ja SonicPi. Maxin vahvuuksia ovat täysin graafinen ulkoasu, katso kuva 1, ja laaja objektikirjasto. Näiden lisäksi Maxilla on aktiivinen käyttäjäkunta sekä kehitystiimi.



Kuva 1. Max 7:n työskentelynäkömää.

Maxilla ohjelmointi perustuu ohjelmoinnin käsitteeseen olio-ohjelmointi. Olio tarkoittaa ohjelmoinnissa jotakin asiaa, jolla on ominaisuuksia (*attributes*), esimerkiksi ihminen. Ihmisellä voi olla ominaisuuksia kuten etunimi, sukunimi ja ikä. Ominaisuuksilla voidaan

olio-ohjelmoinnissa muokata oliota. Oliolla voi olla myös jokin tehtävä (*function*). (MikaBug 2007.) Maxissa oliot ovat objekteja, joilla jokaisella on oma tehtävänsä kuten signaalin viivästys ja muokattavat ominaisuudet kuten viiveen pituus. Olio-ohjelmoinnin perustana on oliopohjainen ohjelmointikieli, joka tarkoittaa kielen ymmärtävän olioita.

Musiikin ja multimedian ohjelmoinnissa olio-ohjelmointi on huomattava helpottava tekijä, sillä ohjelmoijan ei tällöin tarvitse kirjoittaa jokaista tehtävää erikseen. Maxilla ohjelmoimissa riittää, kun asettaa tarvittaville objekteille oikeat ominaisuudet ja yhdistää ne toisiinsa. Kuva 1.

2.2 Miksi juuri Max?

Ensikosketus Maxin kanssa minulla tapahtui toisena opiskeluvuoteni Max-aiheisen kurssin parissa. Kurssin tavoitteena oli oppia käyttämään Maxia siten, että pystyimme ryhmän kesken toteuttamaan Maxilla esityksen elektronisen musiikin tapahtumaan. Minun rooliksi valikoitui esityksen miksaus, sekin totta kai Maxin avulla. Silloin tajusin, että Max on kaipaamani työkalu ja aloin opiskella sen käyttöä itsenäisesti.

Olen saanut Max-ohjelmaan liittyvää tietoa YouTuben videotarjonnasta, aihetta käsittelevien sivujen keskustelupalstoilta, vähäisestä kirjallisuudesta ja yritys-erehdysperiaatteella virheitä korjaten. Max-käyttäjät ovat monesti asialle omistautuneita ja myös auttavat muita ongelmatilanteissa esimerkiksi keskustelupalstoilla. Se auttoi myös minua omalla tielläni Max-noviisista eteenpäin.

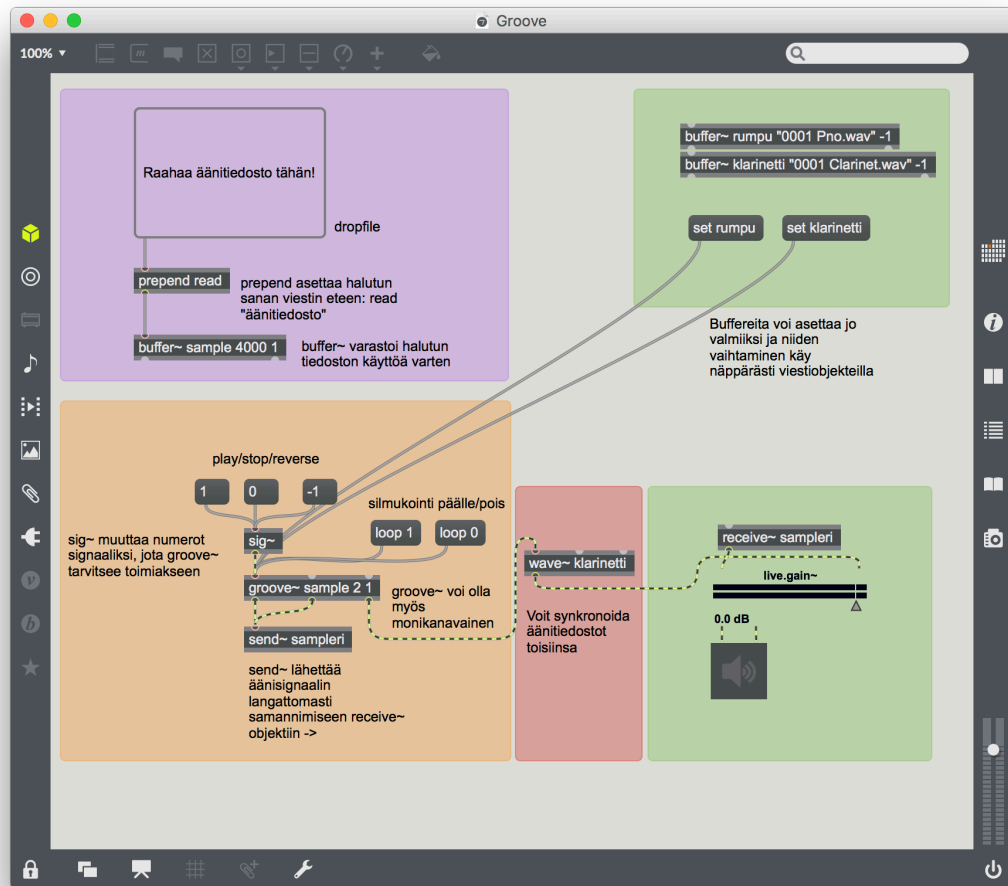
Internet on pullollaan tietoa Maxin kiemuroista ja objekteista, mutta toiminnallista oppimateriaalia on verrattain vähän. Muutama yliopisto pitää verkkokursseja yleisesti Maxin käyttöön liittyen tai jostakin tietystä Maxin osa-alueesta englanniksi. Kurssit maksavat melkein aina paljon, mutta ovat toki hyvin suunniteltuja ja opettajat ovat pitkään olleet Maxin kanssa tekemisissä. Esimerkiksi Stanfordin yliopiston *kadenze.com* -sivustolla on kurssi nimeltä *Programming Max: Structuring Interactive Software for Digital Arts*, jonka vieraileva opettaja David Zicarelli on Maxia ylläpitävän Cycling '74 -yhtiön perustaja (Kadenze 2018). Kyseiselle kurssille pääsee seuraajana ilmaiseksi.

Haluan nyt vuorostani auttaa muita Max-ohjelmasta kiinnostuneita heidän oppipolkunsa alussa. Maxin verkkokursseista ei yksikään ole suomeksi. Tartuin tähän epäkohtaan ja suuntasin oppimateriaalini suomenkielisille oppijoille.

2.2.1 Ohjelmointi Maxilla

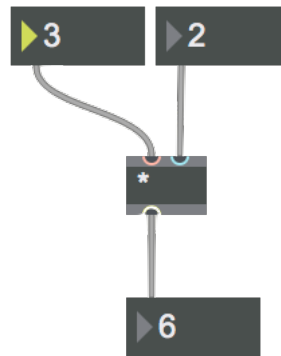
Max jakautuu kuuteen eri osioon, joista jokainen keskittyy erilaisen datan kanssa työskentelyyn. Osiot ovat nimeltään Max, MSP, Jitter, Gen, Mira ja Max for Live. Keskityn opinnäytetyössäni kahteen ensimmäiseen, joista Max kattaa numerodatan käsittelyn, esimerkiksi MIDI-data, ja MSP keskittyy äänisignaalien käsittelyyn. Muista osioista Jitter keskittyy videosignaaleihin sekä 3D-grafiikkaan, Gen on alusta käyttäjän omille algoritmeille, Mira mahdollistaa Maxin ohjaamisen iPadilla ja Max for Live on rajapinta Maxin ja Ableton Liven välillä. Lisäksi Max 7 -versioon on koottu kaksi moduulikirjastoa: moduuleja analogisista syntetisaattoreista (Beap) ja visuaalisia prosessointimoduuleja (Vizzie). (Cycling '74 B 2018.) Kaikki edellä mainitut sisältyvät Max-ohjelmaan ilman erillistä lisäosien latausta, paitsi Mira-appi iPadille maksaa 9.99 Sdollaria (Cycling '74 A 2018).

Maxilla ohjelmointi on graafista ja modulaarista. Graafisuudella tarkoitan Maxin käyttäjäystävällistä, graafista ulkoasua, joka tekee ohjelmoinnin aikana datan liikkumisen hahmottamisen helpoksi verrattuna perinteiseen tekstipohjaiseen ohjelmointiin. Modulaarisuus tarkoittaa sitä, että Maxilla tehdyt *patchit* koostuvat objekteista tai objektikirjastoista, eli moduuleista, jotka ovat kytkettynä toisiinsa. Moduulien avulla on helpompi hahmottaa monimutkaisia tehtäväprosesseja ja etsiä mahdollisia ongelmia kokonaisuudesta *patchista*. Katso Kuva 2. Maxilla voi myös luoda omia objektikirjastoja, jotka pitävät sisällään objekteja ja kytkentöjä. Näitä objektikirjastoja voi jakaa muiden käyttäjien kanssa ja vastaavasti myös käyttää muiden tekemiä objektikirjastoja omissa *patcheissa*. (Blum 2007, 20.)



Kuva 2. Groove-patchin toiminta jaoteltu moduuleihin eri värein.

Maxin toiminta perustuu objekteihin, kuva 3. Objektit ovat rakennuspalikoita, jotka on ohjelmoitu valmiiksi käsittelemään dataa tietyllä tavalla. Max-käyttäjä ei objektin sisälle pääse muokkaamaan objektin toimintaa, vaan hänen tehtävä on valita sopivat objektit ja syöttää niihin ja niistä ulos sopivaa dataa. Max-ohjelmointia voi verrata talon rakentamiseen: kirvesmies ei voi muuttaa rakennusmateriaalien koostumusta esimerkiksi puusta metalliksi, vaan päättää mihin kohtaan kukin materiaali tulee ja millä tavalla se kiinnitetään. Max-ohjelmoinnissa käyttäjälle ei olekaan olennaista se millainen koodi objektin sisällä on, vaan se mitä objekti syöttää ulos dataa saadessaan (Blum 2007, 20).



Kuva 3. Yksinkertainen kertolasku toteutettuna objekteilla.

Max on luonteva valinta audiovisuaalista ohjelmointia harjoittavalle selkeän ulkoasunsa lisäksi myös reaaliaikaisuutensa takia. Nykyään tietokoneiden laskentateho on jo niin huima, että videotakin voidaan muokata reaaliajassa. Äänen muokkaus on niin ikään mahdollista monimutkaisissakin *patcheissa* reaaliaikaisesti. Näitä ominaisuuksia kutsutaan nimellä reaaliaikainen äänen tai videon prosessointi (*real-time audio/video processing*).

Maxin kehityksessä on otettu käyttäjä huomioon koko ajan, minkä ansiosta nykyinen Maxin versio ei ole sotkuinen läjä koodia. Maxin käyttäjäystävällisiä ominaisuuksia voidaan luokitella neljännen sukupolven ohjelmointikielen ominaisuuksiksi, mikä tarkoittaa niiden helpottavan koodin ymmärtämistä (Blum 2007, 20). Mitä korkeammasta sukupolvesta puhutaan, sitä tiivistetympiä ja selkeämpiä tietokoneen suorittamat laskutoimitukset ovat ohjelmoijan näkökulmasta. Se mahdollistaa sen, että ohjelmoijan ei tarvitse enää miettiä jokaista bittiä erikseen. Kolmannen sukupolven ohjelmointikieliä ovat mm. C ja C++, jotka ovat edelleen suosittuja. Max on kirjoitettu C-kielellä (Blum 2007, 21).

Aiemmin mainitsin, että Maxin objektien koodia ei käyttäjä voi itse muokata. C-kieltä käyttämällä on kuitenkin mahdollista tehdä kokonaan omia objekteja ja käyttää niitä *patchissa*. C-kieli toimii nopeammin kuin Max tarkoittaen sitä, että tietokone prosessoi suoraan C-kielellä kirjoitetun materiaalin nopeammin kuin Maxilla kirjoitetun. Omien objektien tekeminen saattaa ratkaista joitakin ongelmia joita Maxilla ei muuten voisi ratkaista (Blum 2007, 21).

2.2.2 Oma historiani Maxin parissa

Max on ollut minulle luovan ajattelun mahdollistava työkalu. Maxin ansiosta minun ei tarvitse mukautua olemassa olevien työkalujen mukaan, vaan voin työkaluja omiin tarpeisiini sopiviksi. Olen Maxin avulla luonut apuvälineitä esimerkiksi Ableton Liven käyttöliittymään, kuten esimerkiksi hiiren liikkeitä tunnistavan ohjaimen ja monikanavamikserin. Livessä ei itsessään ole esimerkiksi 6.0-surroundjärjestelmää ohjaavaa mikseriä, vaan se pitäisi toteuttaa hankalasti *Return*-raitoja käyttämällä. Pystyn Maxilla tehdyn monikanavamikserin avulla ohjaamaan monia kaiuttimia eri tavoin, joka mahdollistaa esityksen toteuttamisen ilman Liven monikanavarajoituksia. Tällöin on myös enemmän vapauksia ajatella luovasti esityksen sisältöä.

Olen erittäin tarkka siitä, millainen työnkulku esimerkiksi miksatessa minulla on. Haluan tehdä miksaamisen teknisesti mahdollisimman mukavaksi ja helpoksi, jotta ajatustyö sujuisi luontevasti. En omista miksauspöytää tai potentiometreillä ja liukusäätimillä varustettua kontrolleria, joten teen kaiken tietokoneella digitaalisesti. Olen tehnyt työskentelyäni helpottamaan kaksiulotteisen *pad*-ohjaimen Max for Livellä, jotta voin ohjata Ableton Liven parametreja kätevästi hiirellä. Toinen vastaava toteuttamani ohjausmekanismi avaa videoikkunan kokoruudun kokoiseksi Esc-näppäintä painamalla tuplaklikkauksen sijaan.

Tällä hetkellä Maxilla tekemäni ohjelmat keskittyvät työskentelyni helpottamiseen ja erilaisten ohjausmekanismien kehittämiseen. Olen jonkin verran kokeillut ja tutkinut äänen muodostusta Maxilla, mutta tähän asti en ole vielä ollut tyytyväinen lopputulokseen. En ole siksi käyttänyt vielä Maxilla tehtyjä ääni-*patcheja* omassa musiikissani.

2.3 Miksi verkkosivut?

Halusin tehdä oppimateriaalin verkkosivujen muotoon sen saavutettavuuden vuoksi. Nykyään jokaisella on pääsy internetiin joko yhdellä tai useammalla laitteella, joten verkkosivujen avulla käyttäjällä on vapaus valita laitteiden välillä. Olen itse etsinyt tietoa tabletilla samaan aikaan, kun olen käyttänyt jotain toista ohjelmaa tietokoneella. Olen havainnut sen olevan erinomainen tapa opiskella ilman, että tarvitsee koko ajan hyppiä samalla ruudulla ohjelman ja verkkosivujen väliä.

Nykyään on paljon mobiilisovelluksia eri asioille, ja minulla kävikin mielessä sovelluksen tekeminen mobiililaitteille. Se ei ollut kuitenkaan järkevä ratkaisu sen rajoittuneisuuden vuoksi, sillä tietokoneella ei ole silloin mahdollista käyttää ko. sovellusta. Nykyiset verkkosivut on optimoitu tietokoneelle.

Tärkeä päämäärä opinnäytetyössäni oli se, että opin jotain uutta prosessin aikana. Olen oppinut hyvin monta uutta asiaa opinnäytetyötä tehdessäni, joista yksi on verkkosivujen tekeminen. Se oli myös yksi syy miksi valitsin verkkosivut esimerkiksi videotutoriaalini sijaan. Taidosta on hyötyä tulevaisuudessa, kun teen itselleni verkkosivut.

Pidän itse enemmän teksti- ja kuvapohjaisista oppimateriaaleista kuin videoista. Monesti oppimateriaalin tekijällä on erilainen käsitys asian omaksumisesta kuin oppijalla, jolloin oppijan on hyvä palata asiassa taaksepäin tai hypätä hieman eteenpäin pysyäkseen tahdissa mukana. Mielestäni tällainen liikkuminen oppimateriaalin sisällä on helpompaa tekstipohjaisessa materiaalissa kuin videoissa.

Videotutoriaaleissa on se haaste, että ne pitäisi pystyä tekemään tarpeeksi lyhyiksi ja selkeiksi. Oppija turhautuu helposti videon ollessa liian pitkä tai vaikeaselkoinen. Harva jättää videon kesken ja jatkaa siitä seuraavana päivänä. Wistian tekemä tutkimus ihmisten videoiden katselusta kertoo, että yleisesti noin 70 % katsojista katsoo videon loppuun asti, kun videon kesto on 1–2 minuuttia, kun 6–12 minuutin mittaisen videon katsoo loppuun asti enää vain puolet (Fishman 2016).

Tekstipohjaisessa oppimateriaalissa on oppijan helpompi jatkaa asian opiskelua myöhemmin, ja asia on mielestäni helpompi jaotella. Oppija turhautuu myös tekstipohjaisen materiaalin ollessa liian pitkä ja vaikeaselkoinen, mutta silloin tulevat kuvat apuun.

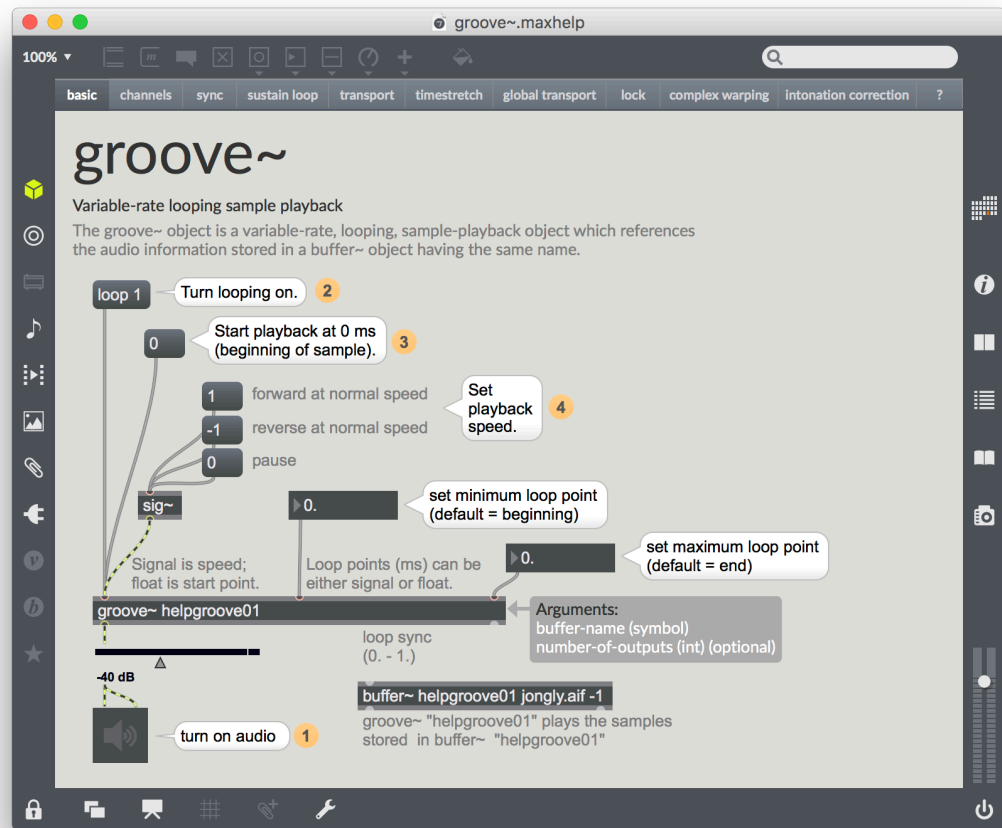
Tekstipohjainen oppimateriaali on tosin haastava sen esille saamisen vuoksi, sillä videot ovat yhä enenevässä määrin kasvussa suosionsa ja tarjonnan ansiosta. Max-aiheiset videot löytyvät vielä helpommin YouTubesta kuin Max-aiheiset verkkosivut Googlen hakutulostulostakosta.

2.4 Esikuvat ja vertailu

Kuten jo aiemmassa luvussa totesin, olen itse opiskellut Maxin käyttöä paljon itsenäisesti. Olen ottanut paljon mallia internet-lähteistä, sillä kirjallisuutta ei aiheesta ole kovinkaan paljon helposti saatavilla.

Otin esimerkkiä dude837:n YouTube-kanavasta, josta löytyy kattavasti videotutoriaaleja monimutkaisiinkin projekteihin. dude837 selventää videoillaan hyvin yksityiskohtaisesti ja johdonmukaisesti projektien etenemistä. Pysin tuomaan omassa oppimateriaalissani asiat myös selkeästi ja yksityiskohtaisesti esille. Haasteena oli saada oppija ymmärtämään asiat kuvien ja tekstien avulla, koska en itse käyttänyt kirjallista oppimateriaalia Max-ohjelmointia opiskellessani.

Toinen esikuva Maxin oppimiseen oli Maxin oma tutoriaali, joka tulee ohjelman mukana. Maxin tutoriaalista sain tietoa perusasioista, joita dude837 ei videoillaan tuo esille. Tutoriaalista löytyy oma osio jokaiselle Maxin ominaisuudelle, kuten MIDI:n käsittely, datatyypit ja ulkoisten ohjainten käyttö. Lisäksi Maxista löytyy myös ohjeikkuna (*Help Window*) jokaiselle objektille, sekä objektin sovellusideoita. Katso kuva 4. Ohjeikkunan käyttöä painotan oppimateriaalissani yhtenä parhaista tiedonlähteistä. Poimin ohjeikkunoista ja tutoriaaleista oppimateriaalin aiheita käsittelevät osiot ja hyödynsin niiden ajattelutapaa *patcheja* rakentaessani.



Kuva 4. Maxin *groove~*-objektin ohjeikkuna.

Verkkosivujen ulkoasuun otin vaikutteita Ableton-yhtiön Live-ohjelman käyttöoppaasta. Liven ohjesivut ovat hyvin kattavat ja mielestäni myös hyvin jäsennellyt. Kyseessä on käyttöopas, joten se ei sellaisenaan ole hyvä pohja oppimateriaalille, mutta sen pohjalta muokkasin esimerkiksi tekstin asettelua. Oppija voi käyttää sivustoa millä tahansa laitteella, jossa on internet-selain, joten tekstiä täytyy pystyä lukemaan sekä isoilta, että pieniltä ruuduilta. Ableton on sivuillaan ratkaissut ongelman rajaamalla tekstin vasempaan laitaan, jota itse myös sovelsin.

2.5 Palautteen keruu

Oppimateriaalini testiryhmänä toimi musiikkiteknologian opiskelijoiden ryhmä Tampereen konservatoriolla. Testiryhmän tehtävänä oli opiskella Maxin käyttöä oppimateriaalini avulla.

Valitsin palautteenkeruumenetelmäksi internet-kyselyn, sillä kohderyhmä oli jo tiedossa ja se oli helpoin ratkaisu välimatkan vuoksi. Kyselyni oli sekä kvantitatiivinen, että kvalitatiivinen, mikä näkyy kyselyn erityyppisissä kysymyksissä. Halusin kyselylläni saada selville käyttäjien kokemuksia (kvalitatiivinen, kuvio 2) ja oppimateriaalini teknistä toimivuutta (kvalitatiivinen, kuvio 1).

Ryhmällä oli hieman reilu kuukausi aikaa opiskella Maxia, minkä jälkeen he vastasivat teettämäni kyselyyn. Kyselyn tarkoitus oli kerätä opiskelijoiden kokemuksia oppimateriaalin käytettävyydestä, vaikeustasosta, laajuudesta ja mielenkiintoisuudesta. Kaikki viisi testiryhmän opiskelijaa vastasivat kyselyyni. Ryhmän opettaja antoi minulle vapaamuotoisesti palautetta sähköpostilla.

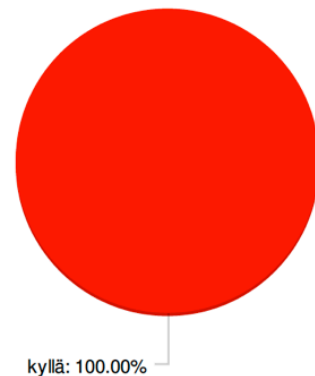
Kysymykset koostuivat avoimista, monivalinta- ja kyllä–ei-kysymyksistä. Kyselyn alussa kartoitin oppimateriaaliin pääsyä ja luonnetta suurpiirteisesti. Katso Kuvio 1. Seuraavaksi kysymyksissä oli monta adjektiivia, joista vastaajan piti valita mielestään sopivimmat. Katso Kuvio 2. Kyselyn edetessä kysymykset tarkentuivat ja avoimet kysymykset tulivat eteen. Kysymyksiä oli yhteensä 15, joten kyselyn täyttäminen vei noin 10–15 minuuttia. Tein kyselyn kyselynetti.com-palvelun avulla (Liite 1).

1|. Pääsitkö sivuille aina kun halusit? *

Osallistujamäärä: 5

5 (100.0%): **kyllä**

- (0.0%): **ei**



Kuvio 1. Kyselyn ensimmäinen kysymys.

4. Valitse kaikki Max ja MSP -verkkosivuja mielestäsi kuvaavat adjektiivit. *

Osallistujamäärä: 5

- (0.0%): mielenkiintoinen

- (0.0%): tylsä

1 (20.0%): rento

- (0.0%): kireä

3 (60.0%): opettavainen

- (0.0%): helposti seurattava

3 (60.0%): epäselkeä

1 (20.0%): moderni

1 (20.0%): vanhanaikainen

- (0.0%): innostava

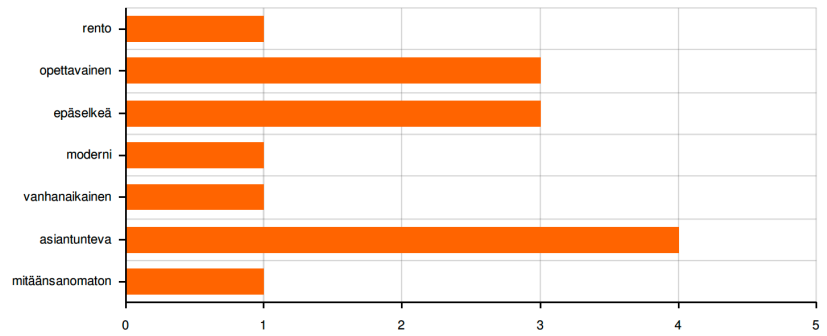
- (0.0%): inspiroiva

4 (80.0%): asiantunteva

1 (20.0%):

mitäänsanomaton

- (0.0%): Muuta



Kuvio 2. Palautekyselyn neljäs kysymys. Vaihtoehdot vasemmalla, tulokset taulukossa.

3 MUSIIKIN OHJELMOINTIKIELTEN HISTORIA

Max Mathews kirjoittaa *The Audio Programming Book* -teoksen esipuheessa musiikin ohjelmoinnin olleen noin 50 vuotta sitten pitkäväteistä ja lopputuloksen olleen kovin yksinkertaista. Silloin ohjelmoinnissa oli käytössä ns. Assembly-kieli, eli symbolinen konekieli (Computer Hope 2017). Tänä päivänä käytössä olevat menetelmät ovat mm. MUSIC V, smusic ja Csound -ohjelmointikielten mahdollistamia, sillä ne laajensivat musiikin ohjelmoinnin mahdollisuuksia. Nykyään käytetyin musiikin ohjelmointikieli on C. (Boulanger & Lazzarini 2011, xi.)

MUSIC N on termi, jolla viitataan useaan eri musiikin ohjelmointikieleen. Termin alle sisältyy mm. ohjelmointikielet MUSIC I ja MUSIC IV. (Di Nunzio, 2011.) MUSIC I on Max Mathewsin vuonna 1957 kehittämä digitaalinen synteessiohjelma, joka oli uranuurtaja seuraajiensa kanssa kohti modernia musiikin ohjelmointia (Lazzarini 2013, 97). Se muodosti kolmioaalto-signaalia, jonka sävelkorkeutta, pituutta ja aaltomuotoa pystyi muokkaamaan rajallisesti (Roads 1996, 87). MUSIC II toi mukanaan *table-lookup* -oskillaattorin, joka on edelleen digitaalisten syntetisaattoreiden ydin. MUSIC III:n myötä tuli digitaalinen synteessimoduuli sekä muita tärkeitä ominaisuuksia digitaalisynteesin maailmaan. (Lazzarini 2013, 97.)

MUSIC IV oli IBM7094-tietokoneella toimiva musiikin ohjelmointiin suunniteltu ohjelma, joka pystyi muokkaamaan äänisignaalia, muttei toistamaan sitä. MUSIC IV kehittyi myöhemmin MUSIC 4C -ohjelmaksi (M4C), joka käyttää hyväkseen C-ohjelmointikieltä. (Lazzarini 2013, 98). Aikaisemmat MUSIC:n versiot käyttivät Assembler-ohjelmointikieltä (Di Nunzio 2011).

3.1 Maxin historia

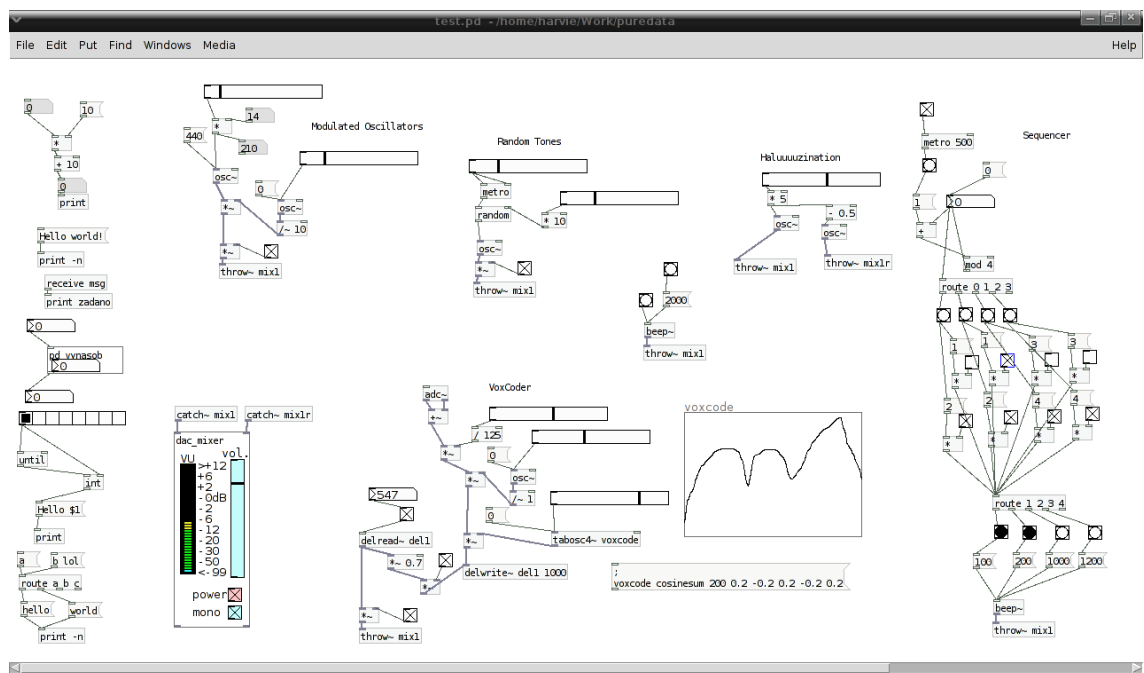
Alun perin Miller Puckette kehitti Maxin vuonna 1986 *Institute de Recherche et Coordination Acoustique/Musiquessa* Pariisissa (IRCAM) ohjaamaan IRCAM 4x -syntetisaattoria. Siitä lähti Maxin kehitys kohti reaaliaikaista musiikinohjelmointiohjelmaa, jonka seuraava askel oli vuonna 1991 kuluttajamarkkinoille tuotu versio, jossa oli noin 150 objektia numero- ja MIDI-dataa varten. Sen kehitti vielä nykyäänkin Maxin kehittäjänä toimiva David Zicarelli yhdessä Opcode Systemsin kanssa. (Blum 2007, 18.) Vuonna 1999 Zicarellin johtama Cycling

'74 julkaisi Max-ohjelman omissa nimissään MSP-lisäosan kanssa ja on siitä lähtien kehittänyt Maxia (Cycling '74 2007).

Maxin lisäosia on tullut matkan varrella melkoinen määrä MSP:n lisäksi. Video-, 3D- ja matriisiprosessointiin suunniteltu Jitter-lisäosa julkaistiin vuonna 2003 (Holmes 2012, 487). Gen julkaistiin vuonna 2011 yhdessä Max 6 -version kanssa (Cycling '74 2011). Max for Liven ensimmäinen versio julkaistiin vuonna 2009. Pitkään jatkuneen yhteistyön tuloksena Ableton osti Cycling '74:n itselleen kesäkuussa 2017. (Zicarelli 2017.) Kirjoitushetkellä Maxin uusin versio on 7.3.5.

3.2 Muita ohjelmointiympäristöjä

Muita Maxin kaltaisia musiikin ja multimedian ohjelmointiympäristöjä ovat mm. Pure Data, SonicPi ja Csound. Pure Datan on kehittänyt myös Maxin kehittäjä Miller Puckette vuonna 1996 (Cycling '74 2007). Pure Data on kaikille ilmainen C-kielellä kirjoitettu ohjelma, joka on saatavilla myös Linuxille, mobiililaitteille, sekä pienempitehoisille tietokoneille kuten Raspberry Pi:lle. (IEM A 2018). Pure Datalla pystyy tekemään käytännössä kaiken sen mitä Maxillakin, mutta käyttöliittymä on hieman erilainen, Kuva 5. Pure Datan kehitystiimi perustuu vapaaehtoistoimintaan, jonka vuoksi ohjelma ei kehity yhtä tasaisesti kuin Max (IEM B 2018).



Kuva 5. Pure Datan käyttöliittymä.

(https://uk.wikipedia.org/wiki/Pure_Data#/media/File:PureData-Harvie-Examples.png)

Sonic Pi on Ruby-ohjelmointikieleen perustuva musiikin reaaliaikainen ohjelmointiympäristö, jonka Sam Aaron kehitti Cambridge yliopistossa vuonna 2012 (Aaron 2018). Se on saatavilla Windowsille, Mac OS:lle, Linuxille ja Raspberry Pi:lle. Sonic Pi on käytössä nuorten ja lasten musiikin ohjelmointikoulutuksessa ja on saanut tunnustuksia mm. Googelta ja Music Teacher Awards -organisaatiolta. (Sonic Pi 2018.) Sonic Pi perustuu koodin kirjoittamiseen, joka muuttuu reaaliaikaisesti musiikiksi, Kuva 6.



Kuva 6. Sonic Pi:n käyttöliittymä. (<https://liliputing.com/2016/02/sonic-pi-make-music-by-code-while-learning-to-code.html>).

Csound on Sonic Pi:n tavoin ilmainen, tekstipohjainen musiikin ohjelmointiympäristö. Csoundia luonnehditaan äänen renderoijaksi, joka muuttaa tekstipohjaiset instrumentit toistettaviksi soundeiksi. Ohjelmoija voi soittaa näitä instrumentteja määrittämällä nuottiarvoilla ja parametreilla. (Boulanger 2018). Csoundin koodia voidaan kirjoittaa millä tahansa lähdekoodieditorilla. Csoundin mukana tuleva editori on nimeltään CsoundQT. (Csound 2018.) Csoundia käytetään paljon mobiililaitteilla toteutetuissa musiikkiprojekteissa ja installaatioissa, sillä Csoundin avulla on helppo ohjelmoida myös mobiililaitteille. Lisäksi Csoundin

The screenshot displays the CsoundQt application interface. The main window is a code editor showing Pure Data patch code for a grain-based sound synthesis. The code includes various signal processing and control structures. A yellow box highlights a comment: `file_old` [Mixes the output of any number of oscillators.].

On the left, the **Interactive Code Pad** shows a list of objects and their parameters:

Object	Parameter 1	Parameter 2	Parameter 3	Parameter 4
1 gisine	ftgen	0,	0,	
4096,	10,	1		
2 gibuzz	ftgen	0,	0,	
2048,	11,	30, 1, 0.8		
3 giratio	ftgen	0,	0,	
4096,	-16,	0.125, 1024,		
4, 4				
5 gicps	ftgen	0,	0,	
4096,	-16,	50, 1024,		
4, 2000				
6				

On the right, the **Csound Debugger** is visible, showing the current stack and breakpoints.

Kuva 7. CsoundQT:n käyttöliittymä. (<https://csoundqt.github.io>)

4 TOTEUTUS

Oppimateriaalin suunnitelma oli alkujaan hyvin suurpiirteinen, koska työtapaani kuuluu suunnitelman hiominen matkan varrella. Ensimmäisessä suunnitelmassani olin miettinyt aikataulun jo pitkälle, enkä arvannut, kuinka yllättäviä käännteitä matkan varrella voi tulla vastaan.

Ensimmäinen askel kohti valmista oppimateriaalia oli luoda kaikkiin suunnittelemiini projekteihin Max-patchit valmiiksi. Alkuvaiheessa aikeissani oli käsitellä yli kymmentä eri projektia, mutta suuren laajuuden vuoksi oli projektit rajattava viiteen:

- Delay
- Tiedostosoitin
- Syntetisaattori (mono)
- Syntetisaattori (poly)
- Sekvensseri

Projektien karsintaan vaikutti oma kokemus Maxin opiskelusta eräällä kurssilla TAMKissa, jäljelle jääneiden projektien tarpeellisuus ja helppous aiheeseen perhetymättömälle, sekä ennen kaikkea englantilaisen muusikkokaverini Nick Rothwellin (2017) kommentti, jossa hän kehotti minua opettamaan Maxia projektien kautta. Oppimateriaalini muoto muuttui tuolla hetkellä teoriapainotteisesta projektipainotteiseksi.

Alkuvaiheessa minun täytyi rakentaa verkkosivuista luonnos, jotta pystyin hahmottamaan mahdolliset tarpeet ja puutteet oppimateriaalin sisältöä luodessani. Verkkosivujen tekemiseen uppoutuikin yllättävän paljon aikaa, sillä vaikka WebFlow-editori olikin helppo ja nopea opetella, oli visuaalisen ilmeen hiominen pitkä prosessi. Kun sivut olivat lähes julkaisukelpoiset, oli minun helppo lisätä sinne asiasisältöä.

Oppimateriaalin sisältö rakentuu Maxin perusteista, ohjatusti tehtävistä Max-patcheista eli projekteista ja niihin liittyvistä tehtävistä, sekä tehtävien ratkaisuista. Luvussa 4.2 käyn läpi yhden projektin työvaiheet, sillä kaikki projektit toteutuivat samaa työprosessia noudattaen.

4.1 Verkkosivuston koonti

4.1.1 WebFlow

Pitkän vertailun ja harkinnan jälkeen HTML-editoriksi valikoitui WebFlow, joka osoittautuikin erittäin hyväksi valinnaksi. WebFlow oli minulle entuudestaan tuntematon, koska se on todella uusi editori, enkä ole tehnyt verkkosivuja aiemmin tässä laajuudessa. Muita potentiaalisia vaihtoehtoja olivat Wix, RapidWeaver ja WordPress. Niissä kaikissa oli kuitenkin joko paljon vaadittavia maksullisia ominaisuuksia tai monimutkainen käyttöliittymä. Olen näistä käyttänyt aiemmin RapidWeaveria ja WordPressia, mutta nähdessäni WebFlow'n mainoksen Facebookissa ja tutustuessani siihen paremmin, oli päätökseni ehdottomasti WebFlow. Sen erikoisuus on sen täysin graafinen käyttöliittymä, jossa käyttäjä näkee sivut koko ajan reaaliajassa, eikä HTML-koodia tarvitse kirjoittaa välttämättä merkkiäkään.

WebFlow:lla editointi on ilmaista, mutta paremmasta editorista ja sivuston julkaisemisesta täytyy maksaa. Ilmainen editointi sisältää kaksi sivustoa ja melkein kaikki editointityökalut. Pois rajattuja ominaisuuksia ovat mm. oma sivuston sisäinen hakukone ja CMS-työkalut (*Content Management System*). Maksamalla 192 dollaria vuodessa saa käyttöönsä hieman enemmän työkaluja ja 10 eri sivustoa kerralla. Paras lisenssi maksaa 420 dollaria/vuosi, jolloin sivustojen määrää ei ole rajattu ja käytössä on kaikki mahdolliset editointiominaisuudet, kuten sivujen salasanasuojaus ja sivustojen siirto käyttäjältä toiselle. Molempien lisenssien hinta on hieman korkeampi kuukausittain maksettuna. (WebFlow A 2018.)

Sivuston julkaiseminen maksaa melkein aina verkkosivuja tehdessä, ellei satu löytämään jostain harvinaista, ilmaista web-hotellia. Web-hotelli on palvelu, joka mahdollistaa verkkosivujen julkaisun. WebFlow'n sivujen julkaisu maksaa halutuista ominaisuuksista riippuen joko 144, 192 tai 432 dollaria vuodessa, tai kuukausittain maksettuna 15, 20 tai 45 dollaria kuukaudessa (WebFlow B 2018). Valitsin itselleni keskimmäisen julkaisulisenssin, lähinnä CMS-ominaisuuksien vuoksi. Tiesin myös, että tulen tarvitsemaan sitä vuoden verran, jolloin oli järkevämpää maksaa koko vuosi kerralla.

Sopivat lisenssit valittuani tutustuin WebFlow'n editointiominaisuuksiin testisivustoa muokkaamalla. Löysin paljon ohjeita työkalujen käytöstä WebFlow Universityn sivuilta,

joka on varta vasten koottu tietopankki palvelua käyttäville. Positiiviseksi yllätyksekseni WebFlowlle on muodostunut todella aktiivinen käyttäjäyhteisö, josta sain myös tietoa mm. keskustelupalstoilta.

4.1.2 Ulkoasu

Prosessin alkuvaiheessa halusin saada visuaalisen kehyksen oppimateriaalilleni, jotta näin miten teksti ja kuvat sopivat sivustoon. Ensiksi valitsin värimaailman, joka oli helppoa sopivan taustakuvan löydettyäni Pixabay-palvelusta. Valitsin rauhallisen, pastellisävyyhin painottuneen värimaailman, joka helpottaa asiaan keskittymistä. Valitsin projektisivujen taustavärit siten, että ensin Delay-projektista alkaen on vihreän sävyjä ja Sekvensseri-projektissa violettia. Vihreä on tunnetusti turvallinen väri (Trötschkes 2012.), joka viestii ensimmäisten projektien kohdalla matalaa aloituskynnystä. Taustavärien vaihtuessa siniseen ja punaiseen projektien vaatimustaso nousee myös. Tekstin ja kuvien taustalle otin kuvankaappauksen Maxin työpohjan taustasta, joka luo yhteyden Maxin ja oppimateriaalin välille.

Käytin sivuston pohjana WebFlown mukana tullutta teemaa, joka sisälsi mm. valikkopalkin, fontit ja yhteystietosivun. Teema oli lähinnä esikuva visuaalisesta ilmeestä, jota pystyin helposti muokkaamaan nykyiseen, jopa alkuperäisestä tunnistamattomaan muotoon.

Tärkeää sivuston ulkoasussa on myös se, että kaikki on tekijänoikeussuojan mukaisesti laillista materiaalia kuvia ja logoja myöten. Olen käyttänyt taustakuvia lukuun ottamatta vain omia kuvamateriaalejani. Taustakuva on haettu Pixabay-palvelusta, jota muokkasin ja monistin Gimp 2.8 -kuvankäsittelyohjelmalla. Pixabay on palvelu, josta saa kuvia ilmaiseksi myös julkiseen käyttöön.

4.2 Sekvensseri-projektin työstö vaiheittain

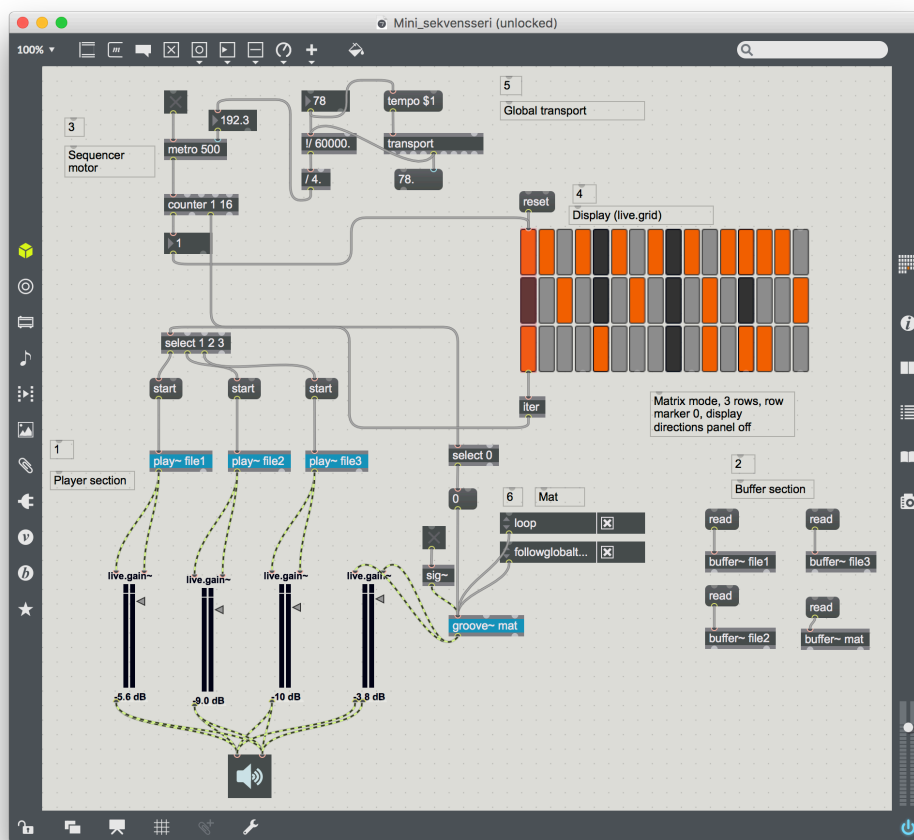
4.2.1 Materiaalin laatiminen

Projektien työstäminen osoittautui suurimmaksi työksi oppimateriaalin teossa. Projektien aiheet valittuani aloin työstää niitä yksi kerrallaan, aina alusta loppuun saakka samalla

dokumentoiden omaa tekemistäni. Käyn seuraavaksi läpi Sekvensseri-projektin työvaiheet.

Ensiksi tein Sekvensseri-patchin Maxilla kokonaan testaten sen toimivuutta ja työvaiheiden selkeyttä. Käytin lähdemateriaalina ko. projektin kohdalla käyttäjä MUST1002:n YouTube-videota ”Max 7 Tutorials: 7.2 Basic Drum Sequencer – Introducing [live.grid]”. Videolla selitetään rumpusekvensserin toimintaperiaate ja sen yksinkertainen toteuttaminen Maxilla. Lisäksi etsin tietoa Cycling ’74 -yhtiön verkkosivujen keskustelupalstalta, sillä en heti löytänyt tarpeisiini sopivia objekteja. Kokemuksen kautta olen oppinut myös sen, että Maxin ohjeikkuna (*Help Window*) auttaa pulmatilanteissa patcheja rakentaessa, joten se oli vahvasti läsnä myös Sekvensseri-projektissa. Sain sieltä apua mm. siihen, millaista dataa *play~*-objekti ottaa vastaan.

Kirjasin muistiin työvaiheet valmiiseen patchiin helpottaakseni patchin uudelleenrakentamista vaiheittain. Valmis ensimmäinen versio Sekvensseri-patchista olikin lähinnä muistiinpano itselleni. Katso Kuva 8.



KUVA 8. Sekvensseri-projektin luonnos, työvaiheet numeroitu.

Kun työvaiheet olivat tiedossa, patch toimi ja oli siisti, olin valmis aloittamaan toisen työvaiheen, joka oli projektin ohjeen kirjoittaminen ja samalla patchin uudelleenrakentaminen vaihe vaiheelta samalla kuvankaappauksia ottaen. Tein kaiken edellä mainitun yhtäaikaisesti pysyäkseen mukana projektin etenemisessä tekstiä kirjoittaessani.

Vältyin myös turhalta oman työskentelytavan painottamiselta, sillä haluan oppijan saavan tarvitsemansa tiedon, muttei valmiita työtapoja. Patchien kuvankaappaukset ja teksti kertovatkin yksinomaan patchin sisällöstä ja toiminnasta.

Sekvensseri-projektissa on paljon asioita, kuten *live.grid*-objektin toiminta, tempon määrittely ja *play~*-objektien ohjaaminen, jotka vaativat minulta tarkkaavaisuutta oppimateriaalin laatijana. Kiinnitin paljon huomiota siihen, että kaikki termit ja käsitteet olisivat mahdollisimman selkeitä ja yhteneviä muiden projektien termien ja käsitteiden kanssa. Esimerkkinä termi sisääntulo, joka monesti alan sanastossa taipuu englanninkieliseen sanaan *input*, mutta oppimateriaalissani se on suomeksi.

Kirjoitin projektien ohjeet Microsoft Wordilla, sillä WebFlow:n tekstinkäsittelyominaisuudet eivät ole läheskään yhtä kattavat. Wordissa oli apuna myös kommenttityökalu, jolla sain kirjattua muistiin asioita myöhempää tarkastelua varten. Tällaisia asioita olivat mm. kappalejaot, ja linkit muille sivuille tai sivun osiin.

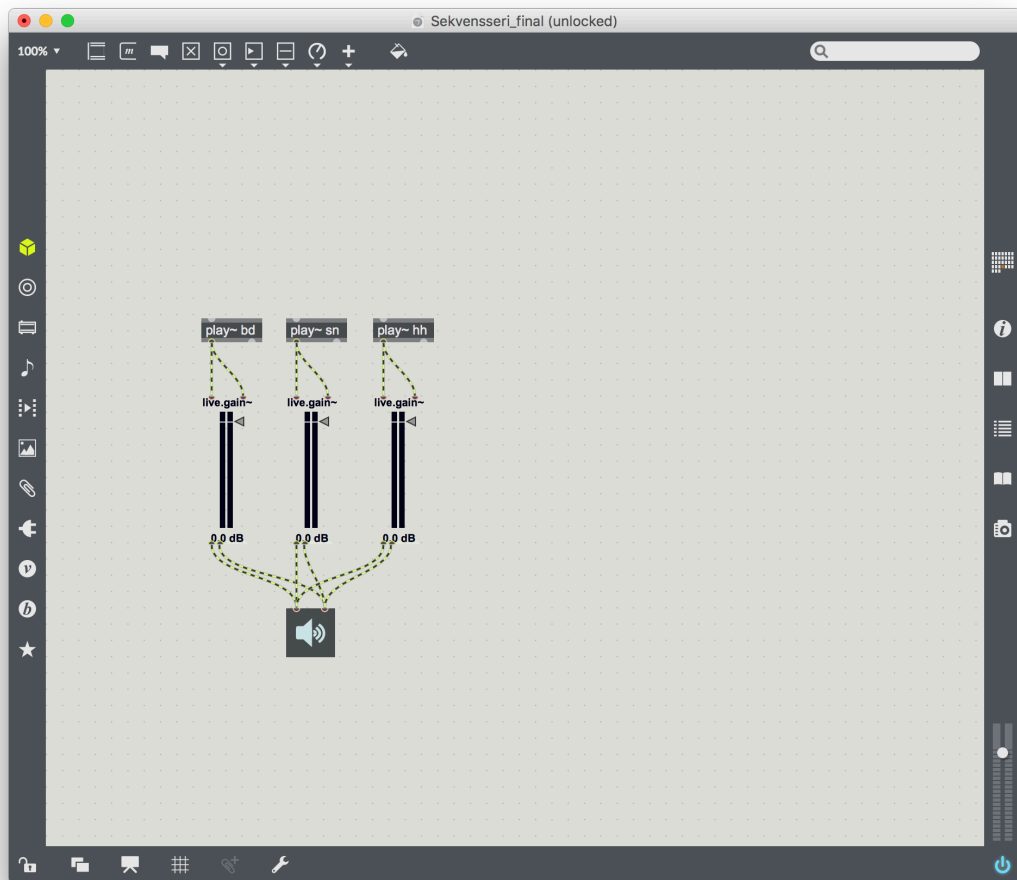
Itse tekstin lähtökohtana oli selostaa oppijalle projektin eteneminen kuvia apuna käyttäen. Ohjeen rakenne koostuu erityövaiheiden selittämisestä sanoin, jonka jälkeen seuraa kuvankaappaus selitetystä asiasta, ja lisäksi olen avannut tarvittavia käsitteitä matkan varrella.

Selitän tekstissä sellaiset asiat, jotka ovat oppijalle mahdollisesti entuudestaan tuntemattomia, tai tärkeitä ymmärtää projektin etenemisen kannalta. Esimerkkinä näyte oppimateriaalistani Sekvensseri-projektin alusta. Tekstiosion jälkeen verkkosivuilla on aina kuvankaappaus juuri selitetystä asiasta. Alla ote Sekvensseri-projektin muistiinpanoista ja sen jälkeen Kuva 9, jossa teksti visualisoituu.

Tarkoituksena on rakentaa rumpusampleja soittava sekvensseri, joten käytämme tässä projektissa tiedostosoitinta, `play~`. `Play~` onkin tuttu jo aiemmasta projektista (#4 Tiedostosoitin), ja se on rumpusekvensseriin sopiva soitin, sillä siinä on vähemmän viivettä kuin esimerkiksi `playlist~`-soittimessa, ja se on helppokäyttöinen.

Otetaan uusi työpohja ja lisätään sinne 3 kpl `play~`-soittimia (bassorumpu, virveli ja hi-hat).

Voit nimetä objektit haluamaasi muotoon, kunhan pysyt itse kartalla mikä objekti soittaa mitäkin samplea. Lisää sen jälkeen kaikille oma volumesäädin (`live.gain~`) ja kytke kaikki säätimet `ezdac~`-objektiin.



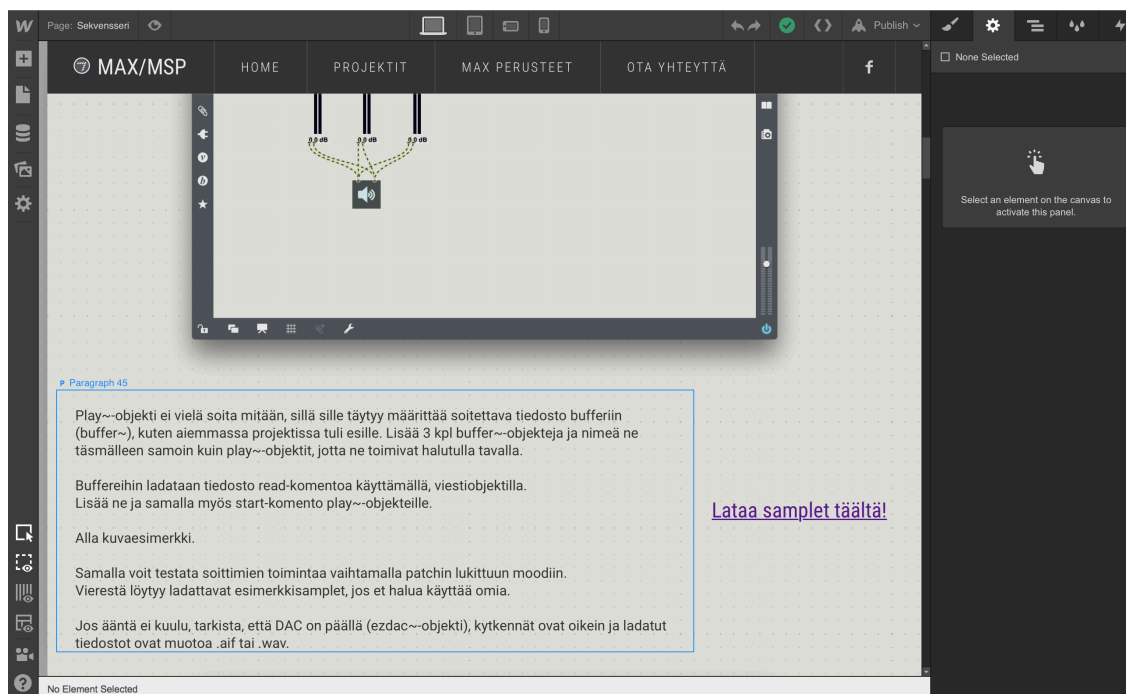
KUVA 9. Sekvensseri-projektin ensimmäinen vaihe.

Kuvankaappauksia en lisännyt Wordiin, vaan merkitsin niiden paikat aputekstillä hakasulkeisiin säästäakseni kovalevytilaa ja turhaa kuvien prosessointia. Kuvankaappaukset on otettu pääosin pelkästään aktiivisesta ikkunasta Macin näppäinkomennolla `Shift + Cmd + 4` ja välilyönti, poikkeuksena koko ruudun kuvankaappaukset työskentelynäkymästä ja ponnahdusikkunoita sisältävät tilanteet.

Rajasin tarkoituksella Sekvensseri-projektissa matriisin ulostuloa kuvaavan kuvankaappauksen eri tavalla, sillä ponnahdusikkunat näkyvät pelkästään koko ruudun kuvankaappauksissa. Verkkosivuilla kuvissa ei ole kuvatekstiä, sillä jokaisen kuvan sisältö on jo selitetty kertaalleen tekstissä.

4.2.2 Oppimateriaali verkkosivuille

Seuraavaksi siirsin valmiin tekstimateriaalin verkkosivuille yhtenä kokonaisuutena, jota muokkasin WebFlown antamilla työkaluilla haluamakseni. Kuten jo aiemmin mainitsin, WebFlown tekstinkäsittely ei ole yhtä sujuvaa kuin Wordilla, mutta huomasin myös, etteivät nämä kaksi ohjelmistoa ymmärrä toisiaan täysin. Tekstin liitettyäni sivuille se sisälsi paljon ihmeellisiä rivinvaihtoja ja välilyönnit puuttuivat aika ajoin kokonaan. Sen lisäksi jaoin tekstin kuvien kohdalta eri osiin, sillä WebFlow rakentaa sivut osioista (*section*) ja niiden sisällä olevista palstoista (*column*). Jokaiseen osioon tai palstaan voi lisätä tekstiä kappaleina (*paragraph*), mutta kappaleiden sisälle ei voi lisätä muuta kuin tekstiä ja linkkejä. Näin ollen sijoitin kuvat kappaleiden väliin ja tekstin muokkaus tapahtui kappale kerrallaan. Kuva 10.



KUVA 10. Tekstin ja kuvien asettelu verkkosivuille. (<http://max-ja-msp.webflow.io/projektit/sekvensseri>)

Fontiksi valikoitui sivuston teeman oletusfontti Roboto, ja fonttikoko vaihtui kokoon 16. Fonttikoon lisäksi sivuston asettelu on tässä vaiheessa vielä optimoitu tietokoneen ruudulle, mutta lähitulevaisuudessa optimoin ulkoasun myös mobiililaitteille. Tekstin asetteluna käytin seuraavia parametreja: vasen reuna on 3 % irti ruudun vasemmasta reunasta, oikea reuna 250 pikseliä irti oikeasta reunasta. Yllä olevassa kuvaesimerkissä teksti on kaksiosaisen taulukon vasemmassa solussa, jotta linkki *sample*-kirjastoon mahtuu sivun oikeaan laitaan.

Lisäsin myös otsikot ja väliotsikot, joita ei Wordissa ollut, sekä oikoluin tekstin. Lisäsin kuvankaappaukset ja pienensin niiden kokoa 75 prosenttiin alkuperäisestä. WebFlow antaa määrittää kuvien koon joko prosentteina tai pikseleinä, mutta pikseleinä kokoa muuttaessa se ei skaalaudu muille laitteille sopivalla tavalla. Kuvien keskittäminen antoi mielestäni ryhtiä muuten vasemmalle painottuneeseen asetteluun. Kuvien taustalla näkyvä harmaa varjo on Macin aktiivisen ikkunan kuvankaappauksesta johtuva efekti, joka tuo kuvan hyvin esille taustastaan. Ainut huono puoli on se, ettei samaa efektiä saa aikaiseksi rajatuissa tai koko ruudun kuvankaappauksissa.

4.2.3 Tehtävän laatiminen ja viimeistely

Jokaiseen projektiin sisältyy pieni tehtävä, joka yhdistää jo aiemmin opittuja asioita ko. projektiin. Niin kuin jokaisessa projektissa, myös Sekvensseri-projektissa keksin tehtävän vasta projektin valmistuttua. Sekvensseri-tehtävä ei ole kovin haastava, mutta se ohjaa oppijan etsimään tietoa objektien toiminnasta Maxin sisällä. Tehtävänanto kuuluu näin: ”Selvitä, kuinka sekvensserin kulkusuuntaa voidaan muuttaa lennosta etuperin, takaperin tai edestakaisin kulkevaksi. Vihje: counter.”

Valmiissa Sekvensseri-patchissa on kaikki tarvittava jo valmiina tehtävän tekemistä varten. *counter*-objekti ja sekvensserin toiminta ovat melko tuttuja käsitteitä jo tässä vaiheessa, joten tehtävässä oppijalla on valmiudet tutustua *counter*-objektin toimintaan hieman paremmin. Tehtävän tarkoitus on vahvistaa oppijan käsitystä Maxin loogisesta toiminnasta ja antaa positiivisia onnistumisen hetkiä objektiorientoituneen ohjelmoinnin parissa.

Tehtävän ratkaisu löytyy tehtävänannon alapuolelta linkin takaa tekstitiedostona, jonka avaaminen Maxissa on selostettu jokaisen tehtävänannon ohessa. Ratkaisut ovat Max-

patcheja, jotka tein itse tehtäviä suunnitellessani, ja ovat tekstitiedostoina (txt), koska WebFlow ei tue maxpat-tiedostoja, tekstitiedostot ovat pienikokoisia ja ne toimivat sekä Windows- että Mac-käyttäjien Max-ohjelmissa. Lisäksi esimerkiksi Maxin valmistajan, Cycling '74:n, keskustelupalstalla on paljon Max-patcheja jaossa tekstitiedostoina, joten kun oppija osaa aukaista tekstitiedoston Maxilla, on hänen helpompi jatkossa käyttää keskustelupalstaa hyväksi.

Sekvensseri-projektin viimeistelyvaiheessa lisäsin tekstiin lähdeviitteet, tarkistin asettelun vielä kertaalleen, lisäsin linkin Tiedostosoitin-projektin sivulle alkutekstiin, sekä muutin alapalkin vuosiluvun ja taustakuvan oikeaksi. Sen jälkeen tarkastin koko sivuston ulkoasun ja lähetin linkin testiryhmälle.

5 PALAUTE JA KEHITTÄMINEN

Testiryhmältäni tuli palautekyselyssä erittäin hyviä kehitysehdotuksia. Ryhmä oli selkeästi paneutunut asiaan ja heidän ollessaan oppimateriaalini kohderyhmää on palaute erittäin antoisaa. Verkkokyselyn avoimissa kysymyksissä opiskelijat saivat esittää kehitysehdotuksia ja kertoa mikä heidän mielestään toimi hyvin. Vastaukset näissä kysymyksissä olivat enimmäkseen painottuneet kehitysehdotuksiin, joka oli mielestäni positiivinen yllätys.

Suurin epäkohta oppimateriaalissani testiryhmän mukaan oli ”mutkien suoristaminen”, eli oppijan taitojen yliarviointi. Opiskelijat kertoivat hukanneensa ajatuksen tai etsineensä tietoa muualta selvittääkseen mitä jossakin projektin työvaiheessa oli tapahtunut. Seuraavassa hieman otteita opiskelijoiden palautekyselyn avoimista kysymyksistä:

”Sivustolla on tehty mielestäni hiukan liikaa oletuksia koskien käyttäjän tietoja Maxista, tai osaamista musiikin tietotekniikan suhteen.

Vähän sellaista "mutkat suoriksi" -meininkiä.

Eli enemmän Maxin toimintaperusteiden selitystä olisin kaivannut.

Tehtävissä olisi saanut olla enemmän välivaiheiden kuvauksia.”

”Yleisellä tasolla hyvä kokonaisuus. Parannuksia kannattaa miettiä vielä enemmän sellaisen tekijän suunnasta, joka ei ymmärrä asiasta ennen sivustolle tulemistakaan. Vaikka kyseessä olisikin pidemmälle ehtinyt tekijä, niin häntä täytyy johdatella. Nyt tuntui siltä, että on mahdollista hukata tekijä esimerkiksi etsimään tietoa toisilta sivustoilta.”

Tein palautteen perusteella muutoksia sivustolle lisäämällä välivaiheista enemmän kuvia ja jaotteleamalla tekstiä selkeämmin. Alun perin esimerkiksi Käyttöliittymä-sivulla oli ensin tyhjä kuva työpohjasta ja heti seuraavassa kappaleessa kerroin jo omasta työskentelytavastani hieman ylitsepuksuvan kuvan kera. Otin sen kappaleen kokonaan pois ja jäsentelin Käyttöliittymä-sivun siten, että oppija käy vaiheittain kaikki työskentelyikkunat läpi ja kokeilee niitä. Koin oman työtapani esittelemisen olevan sittenkin turha, joten se ei ole enää esillä.

Projektit alkavat Delay-efektistä ja jatkuvat Tiedostosoittimeen ja sitä kautta syntetisaattoreihin. Tämä herätti hämmennystä, sillä vaikka Max perusteet -osiossa

kerron äänisignaalin reitityksestä mikrofonista Maxiin ja kaiuttimiin, ei testiryhmä pystynyt sitä sisäistämään tarpeeksi hyvin. Muutin rakennetta siten, että ensimmäinen projekti onkin Äänen reititys. Siinä perehdytään hieman tarkemmin äänisignaalin reitittämiseen Maxissa, jonka jälkeen on loogisempaa edetä kohti Delay-efektin rakentamista.

Palautteessa mainittiin myös patchien graafisen ulkoasun muokkaus ja Maxin toimintaperiaatteet, joita oli ryhmän mukaan esitelty liian vähän. Eräs oppija olisi halunnut saada paremman käsityksen Maxin mahdollisuuksista ja käytössä olevista työkaluista, jotta voisi tulevaisuudessa käyttää sitä paremmin hyödyksi. Otin tämän asian esille, koska koen ko. asian olevan olennainen osa oppijan motivaatiota jatkaa Maxin käyttöä oppimateriaalin jälkeenkin. Haluan oppijan innostuvan Maxin käytöstä ja sen mahdollisuuksista, jotka palautteen mukaan eivät toteutuneet oppimateriaaliani käyttäessä. Asian korjaaminen on seuraava askel oppimateriaalia parantaessa, sillä en pystynyt sitä opinnäytetyön aikataulun puitteissa toteuttamaan. Graafisen ulkoasun muokkaaminen selkeämmäksi oli myös esille tullut epäkohta, joka on myös kehitysehdotuksien listalla.

Sain myös ryhmän opettajalta tekstimuotoisen palautteen sivuston toiminnasta, jossa tuli hyvin esille opettajan näkökulma oppimateriaalin toimivuudesta. Oppimateriaalini on myös suunnattu opettajan apuvälineeksi Max-aiheisia kursseja pidettäessä, joten opettajan näkökulma on todella arvokas kehittäessäni oppimateriaalia yhä paremmaksi.

Opettajan palautteen perusteella tein muutoksia tekstin seuraamisen ja selkeyttämiseen liittyen. Muutin kuvakoon 50 % aiemman 80 %:n sijaan ja lisäsin numeroinnin tekstiin niihin kohtiin, joissa oppijan on tarkoitus tehdä jotakin itse. Tällöin oppija erottaa toiminnalliset ohjeet muun ohjeistuksen seasta paremmin.

Opettaja ei lähtökohtaisesti opettanut ryhmää testitilanteessa, joten myös hänen mukaansa asiat voisivat olla vielä enemmän vaiheittain selitettyjä. Oppija pysyy tällöin paremmin mukana asiassa edetessään ja opettajan rooliksi opetustilanteessa jää valvoa projektien edistymistä ja vastata kysymyksiin tarvittaessa. Tällainen oppiminen olikin yksi tavoitteistani oppimateriaalia tehdessäni.

Pääosin palautteessa muistuteltiin minua siitä, että Maxista täysin tietämätön tarvitsee enemmän välivaiheita, sekä selkeämmän ja johdonmukaisemman ohjeistuksen. Oli erittäin tärkeää saada palautetta juuri näistä asioista, sillä itse tekijänä ei aina pysty näkemään tekemistään kohderyhmän silmin.

Muita kehitystä vaativia asioita oppimateriaalissani opinnäytetyön jälkeiselle ajalle ovat mm. sivuston optimointi mobiililaitteille, projektikirjaston kasvattaminen eri efektien rakennusohjeilla ja mahdolliset video-oppaat tekstimateriaalin rinnalle. Kaikki edellä mainitut kehitysehdotukset ovat olleet mielessäni jo projektin alusta asti, mutta aikataulun vuoksi en pystynyt niitä toteuttamaan opinnäytetyöni aikana.

6 POHDINTA

Tavoitteet itselleni opinnäytetyöni suhteen oli oppia luomaan toimivat verkkosivut, laadukas oppimateriaali, vastata Max-opetuksessa havaitsemiini puutteisiin ja havainnoida omaa pedagogista osaamistani. Opin paljon asioita kaikista edellä mainituista alueista, mutta ennen kaikkea itsestäni. Koen olevani sekä musiikkiteknologian osaajana, että opettajana enemmän perillä omasta osaamisestani opinnäytetyöni myötä.

Haasteita opinnäytetyössäni olivat varsinkin aiheen rajausta ja oppimateriaalin työstäminen selkeään muotoon. Monta kertaa prosessin aikana kuulin sanottavan eri yhteyksissä, että asiaan perehtyneen on joskus vaikeaa selittää asia siitä mitään tietämättömälle selkeästi. Näin on käynyt itselleni varsinkin musiikkiteknologian parissa, sillä jo pelkästään ammattisanasto voi olla tottumattomalle hyvin haasteellista ymmärtää. Siitä huolimatta koen onnistuneeni, varsinkin palautteen myötä saamieni kehitysehdotuksien myötä, muokkaamaan oppimateriaalini selkeästi ymmärrettävään muotoon.

Nyt valmista oppimateriaalia tarkastellessani mietin jo edellisessä luvussakin esille tulleita kehitysmahdollisuuksia. Voisin hyvin tehdä yhteistyötä jonkun toisen aiheesta kiinnostuneen opettajan kanssa ja työstää oppimateriaalista hieman laajempi kokonaisuus, joka käsittäisi mahdollisesti myös videoprosessointia. Koen, että yksin sellaisen oppimateriaalikokonaisuuden tekeminen vaatisi aivan liikaa resursseja ja voi kaatua yksipuolisuuteen.

Prosessin aikana tärkein opetus itselleni on ollut omien rajojeni löytäminen. Olen oppinut löytämään itselleni mielekkäitä työtapoja ja olen oppinut paljon uutta sellaisista asioista, joista en ennen tiennyt paljoakaan. Olen mielestäni täyttänyt omat tavoitteeni opinnäytetyön suhteen ja aion hyödyntää tuotostani jatkossa itseni markkinoinnissa sekä opetuksessa. Toivon, että kaikki ne, jotka haluavat opiskella Max-ohjelman käyttöä suomeksi, löytävät sivustoni ja hyötyvät siitä.

LÄHTEET

Aaron, S. 2018. Samuel Aaron. Curriculum vitae. [Ansioluettelo]. Luettu 24.3.2018. <http://sam.aaron.name/files/cv.pdf>

Blum, F. 2007. Digital Interactive Installations. Programming interactive installations using the software package Max/MSP/Jitter. Diplomica Verlag GmbH.

Boulanger, R. 2018. Introduction to Sound Design in Csound. WWW-dokumentti. Luettu 2.4.2018. <http://www.csounds.com/chapter1/index.html>

Csound. 2018. Get Started. Csound editors. Luettu 2.4.2018. <http://csound.com/get-started.html>

Cycling '74. 2007. History and Background. Luettu 16.3.2018. <http://web.archive.org/web/20090609205550/http://www.cycling74.com/twiki/bin/view/FAQs/MaxMSPHistory>

Cycling '74. 2011. Gen - Extend the Power of Max. Luettu 16.3.2018. <http://cycling74.com/products/gen/>

Cycling '74 A: Cycling '74. 2018. MIRA. An iPad app that automatically connects to Max and mirrors your interface. Luettu 16.3.2018. <https://cycling74.com/products/mira>

Cycling '74 B: Cycling '74. 2018. Patch Inside Max. Luettu 7.3.2018. <https://cycling74.com/products/max>

Computer Hope. 2017. Assembly language. Päivitetty 11.4.2017. Luettu 16.3.2018. <https://www.computerhope.com/jargon/a/al.htm>

Di Nunzio, Alex. 2011. Music N. Artikkel. Päivitetty 13.10.2013. Luettu 16.3.2018. <http://www.musicainformatica.org/topics/music-n.php>

Fishman, E. 2016. How long should your next video be?. Blogikirjoitus. Julkaistu 5.7.2016. Luettu 2.4.2018. <https://wistia.com/blog/optimal-video-length>

Ali Jamieson. 2018 <http://alijamieson.co.uk/2014/08/getting-started-maxmsp-maxforlive-best-resources-tutorials/>

Holmes, T. 2012. Electronic and Experimental Music. Technology, Music, and Culture. Routledge.

IEM A: IEM. 2018. Pure Data. About Pure Data (aka Pd). Luettu 24.3.2018. <http://puredata.info>

IEM B: IEM. 2018. Pure Data. Development Wiki. Luettu 24.3.2018. <https://puredata.info/dev>

Kadenze Inc. 2018. Programming Max: Structuring Interactive Software for Digital Arts. Luettu 7.3. 2018. [https://www.kadenze.com/courses/programming-max-structuring-interactive-software-for-digital-arts-i/info?signedin=true#_ =](https://www.kadenze.com/courses/programming-max-structuring-interactive-software-for-digital-arts-i/info?signedin=true#_=)

Lazzarini, V. 2013. The Development of Computer Music Programming Systems. Journal of New Music Research. 1/2013, 97-110.
<http://dx.doi.org/10.1080/09298215.2013.778890> (vaatii käyttöoikeuden)

MikaBug. 2007. Opasarkisto. JavaScript-perusopas. Osa 10 - Olio-ohjelmointi. Luettu 24.3.2018. https://www.ohjelmointiputka.net/opaat/opas.php?tunnus=js_10

Boulanger, R & Lazzarini, V. 2011. The Audio Programming Book. Massachusetts Institute of Technology.

Roads, C. 1996. Introduction to Digital Sound Synthesis. MIT Press.

Sonic Pi. 2018. Sonic Pi. Luettu 24.3.2018. <https://sonic-pi.net>

WebFlow A: WebFlow. 2018. Designer Plans. Luettu 6.2.2018.
<https://webflow.com/pricing/designer-plans>

WebFlow B: WebFlow. 2018. Hosting Plans. Luettu 6.2.2018.
<https://webflow.com/pricing/one-site>

Rothwell, N. Säveltäjä, muusikko. 2017. Haastattelu 12.7.2017. Haastattelija Kärki, S. Nurmes.

Trötschke, R. 2012. Värit ovat ikivanha visuaalinen kieli. Yle. Päivitetty 8.11.2016. Luettu 23.2.2018. <https://yle.fi/aihe/artikkeli/2012/12/10/varit-ovat-ikivanha-visuaalinen-kieli>

Zicarelli, D. 2017. Cycling '74 + Ableton. Keskustelupalstakirjoitus. Julkaistu 6.6.2017. Luettu 16.3.2018. <https://cycling74.com/articles/cycling-'74--ableton>

LIITTEET

Liite 1. Max-oppimateriaalin testiryhmän kyselyn tulokset.

Kysely luotu kyselynetti.com-palvelun kautta.

1 (7)

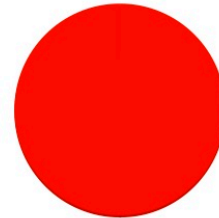
Max-ohjelman oppimateriaali

1. Pääsitkö sivuille aina kun halusit? *

Osallistujamäärä: 5

5 (100.0%): kyllä

- (0.0%): ei



kyllä: 100.00%

2. Mielestäni sivuilla on tietoa Max-ohjelmasta... *

Osallistujamäärä: 5

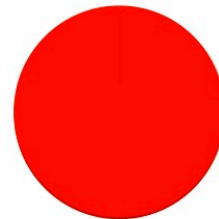
- (0.0%): Aivan liian vähän

5 (100.0%): Hieman liian vähän

- (0.0%): Juuri sopivasti

- (0.0%): Hieman liikaa

- (0.0%): Aivan liikaa



Hieman liian vähän: 100.00%

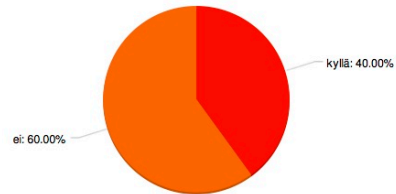
2 (7)

3. Löysitkö sivuilta tietoa helposti? *

Osallistujamäärä: 5

2 (40.0%): kyllä

3 (60.0%): ei



4. Valitse kaikki Max ja MSP -verkkosivuja mielestäsi kuvaavat adjektiivit. *

Osallistujamäärä: 5

- (0.0%): mielenkiintoinen

- (0.0%): tylsä

1 (20.0%): rento

- (0.0%): kireä

3 (60.0%): opettavainen

- (0.0%): helposti seurattava

3 (60.0%): epäselkeä

1 (20.0%): moderni

1 (20.0%): vanhanaikainen

- (0.0%): innostava

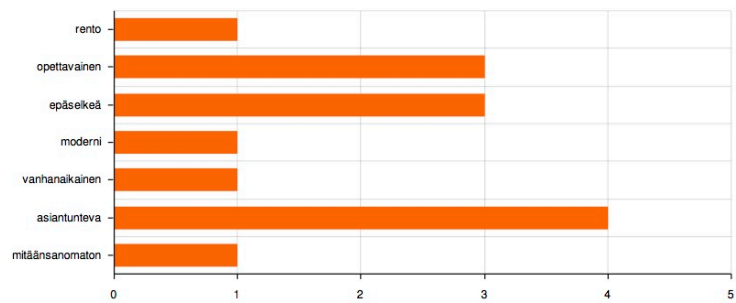
- (0.0%): inspiroiva

4 (80.0%): asiantunteva

1 (20.0%):

mitäänsanomaton

- (0.0%): Muuta



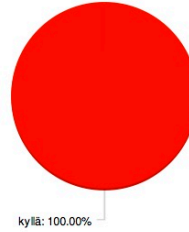
3 (7)

5. Oliko oppimateriaalin Perusteet-osio mielestäsi tarpeeksi kattava? *

Osallistujamäärä: 5

5 (100.0%): kyllä

- (0.0%): ei



6. Jos vastasit edelliseen kysymykseen "ei", kerro mitä asioita haluaisit Perusteet-osiossa olevan.

Osallistujamäärä: 0

7. Mielestäni Perusteet-osiota oli ... seurata. *

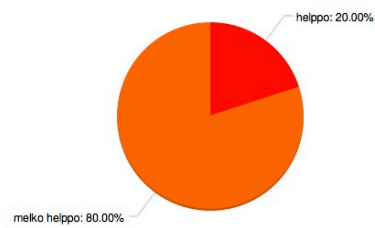
Osallistujamäärä: 5

1 (20.0%): helppo

4 (80.0%): melko helppo

- (0.0%): melko vaikea

- (0.0%): vaikea



4 (7)

8. Mielenkiintoisin projekti *

Osallistujamäärä: 5

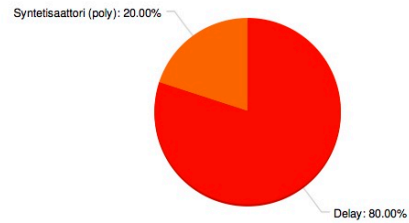
4 (80.0%): Delay

- (0.0%): Tiedostosoitin

- (0.0%): Syntetisaattori (mono)

1 (20.0%): Syntetisaattori (poly)

- (0.0%): Sekvensseri



9. Valitse mielestäsi helpoin ja vaikein projekti. *

Osallistujamäärä: 5

	Delay		Tiedostosoitin		Syntetisaattori (mono)		Syntetisaattori (poly)		Sekvensseri	
	Σ	%	Σ	%	Σ	%	Σ	%	Σ	%
Helpoin	-	-	3x	60,00	1x	20,00	1x	20,00	-	-
Vaikein	2x	40,00	-	-	1x	20,00	1x	20,00	1x	20,00

5 (7)

10. Kerro mitä olisit halunnut oppia lisää projektien avulla? Vastaa omin sanoin.

Osallistujamäärä: 4

- Jossain kohden tiettyjen moduulien toiminta ja se, miksi kyseisiä moduleita kytkiskellään yhteen tuntui olleen ilmaistu siten, että asiaa tuntematon tekijä ei välttämättä pysy mukana tai saa käsitystä, miksi mitään tapahtuu.

Osin tämä viittaisi siihen, että johdatusta voisi olla enemmän.

(Kuvia kannattaa miettiä työvaiheista. Nyt tuntuma on, että löytyy kuva isosta lopputuloksesta, mutta työvaiheet eivät välttämättä hahmotu halutulla tavalla)

- Olisin halunnut saada paremman käsityksen Maxin toimintaperiaatteista ylipäätään sen sijaan, että kasaan plugarin ohjeiden mukaan. Nyt katsoin materiaalin lisäksi Youtubesta opetusvideoita, jotta ymmärtäisin, miksi jokin asia pitää tehdä jollain tavalla. Haluan tietää, mitä työkaluja on käytettävissä, jotta voin hyödyntää niitä myös muussa ympäristössä.

- Sivustolla on tehty mielestäni hiukan liikaa oletuksia koskien käyttäjän tietoja Maxista, tai osaamista musiikin tietotekniikan suhteen.

Vähän sellaista "mutkat suoriksi" -meininkiä.

Eli enemmän Maxin toimintaperusteiden selitystä olisin kaivannut.

Tehtävissä olisi saanut olla enemmän välivaiheiden kuvauksia.

- Projektit ei olleet aivan idioottivarmasti ohjeistettuja. Syy-/seuraussuhteiden hahmottaminen jäi välillä puutteelliseksi.

11. Mitä mieltä olet projektien yhteydessä olevista tehtävistä? *

Osallistujamäärä: 5

3 (60.0%): looginen

1 (20.0%): tärkeä

- (0.0%): turha

- (0.0%): irrallinen

1 (20.0%): kekseliäs

1 (20.0%): helppo

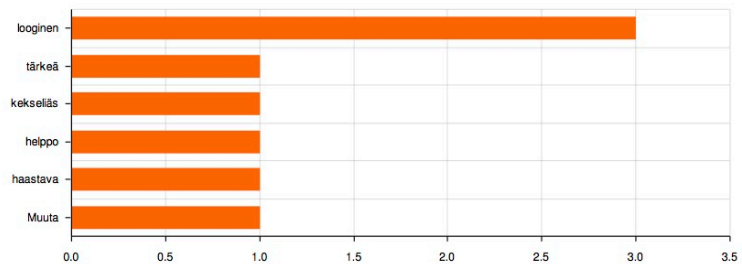
1 (20.0%): haastava

- (0.0%): mielenkiintoinen

1 (20.0%): Muuta

Lisäkentän vastaukset:

- hyvä



6 (7)

12. Uskotko käyttäväsi Max-ohjelmaa äänen tai musiikin tekemiseen? Miksi? *

Osallistujamäärä: 5

- Käytän kyllä. Se auttaa hahmottamaan signaalireittejä.
- Uskon. Omien soittimien, efektien yms tekeminen on aina mielenkiintoista.
- Kyllä, sillä olen huomannut, että se mahdollistaa vaikka mitä.
- Mahdollisesti.
- En voi varmuudella luvata. Vaikuttaa aika vaivalloiselta hyötyyn nähden.

13. Mitä muuta haluaisit oppia Max-ohjelmasta? *

Osallistujamäärä: 5

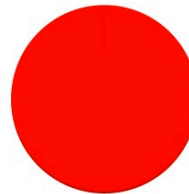
- Graafisen ulkoasun muokkaamista järkeväksi
- Toivoisin lisää efektien rakenteluohjeita.
- Keinoja yhdistää sitä muihin ohjelmiin ja käyttöliittymiin, kuten modulaarisyntetisaattoreihin.
- Aikamatkustuksen.
- Mikäli inspiroidun - kaiken olennaisen.

14. Osaatko etsiä lisää tietoa Max-ohjelmasta? *

Osallistujamäärä: 5

5 (100.0%): kyllä

- (0.0%): ei



kyllä: 100.00%

7 (7)

15. Tähän voit kirjoittaa mietteitäsi, parannusehdotuksia, kokemuksia, yms.

Osallistujamäärä: 2

- Yleisellä tasolla hyvä kokonaisuus. Parannuksia kannattaa miettiä vielä enemmän sellaisen tekijän suunnasta, joka ei ymmärrä asiasta ennen sivustolle tulemistakaan. Vaikka kyseessä olisikin pidemmälle ehtinyt tekijä, niin häntä täytyy johdatella. Nyt tuntui siltä, että on mahdollista hukata tekijä esimerkiksi etsimään tietoa toisilta sivustoilta.

Tehtävien järjestys olisi suositeltavaa aloittaa esim. tiedostosoittimesta ja vaikka lisätä siihen delay. Itselleni on erikoista aloittaa delaystä, ennen kuin on mitään käsitystä, miten ääntä saadaan ohjelmassa aikaan.

Eli pientä hiomista ja nimenomaan siitä näkökulmasta, että olet vähän enemmän asiasta pihalla, kun saavut sivustolle. Lyhyempia tekstikappaleita. Osa asioista voi olla sellaisia, että ne kannattaa vain jättää pois.

-Lähinnä kaipasin selkeyttä projektien eri työvaiheiden esittelyyn.