

„Project: Fingerhythm“ - Konzeption, Entwicklung und Gestaltung
eines Mobile Music Games

Masterarbeit
zur Erlangung des akademischen Grades Master of Arts in Arts and Design
im interuniversitären Masterstudium Sound Design

begutachtet von Univ. Prof. Mag. Ph.D. Marko Ciciliani

Universität für Musik und darstellende Kunst Graz
Institut für Elektronische Musik und Akustik

FH Joanneum Graz
Institut Design & Kommunikation

Peter Zotter CMS15
00873160 / 1510374043
peter.zotter@student.kug.ac.at
September 2017

Eidesstattliche Erklärung

Ich erkläre ehrenwörtlich, dass ich die vorliegende Masterarbeit selbstständig angefertigt und die mit ihr verbundenen Tätigkeiten selbst erbracht habe. Ich erkläre weiters, dass ich keine anderen als die angegebenen Hilfsmittel benutzt habe. Alle aus gedruckten, ungedruckten oder dem Internet im Wortlaut oder im wesentlichen Inhalt übernommenen Formulierungen und Konzepte sind gemäß den Regeln für gutes wissenschaftliches Arbeiten zitiert und durch Fußnoten bzw. durch andere genaue Quellenangaben gekennzeichnet.

Die vorliegende Originalarbeit ist in dieser Form zur Erreichung eines akademischen Grades noch keiner anderen Hochschule vorgelegt worden. Diese Arbeit wurde in gedruckter und elektronischer Form abgegeben. Ich bestätige, dass der Inhalt der digitalen Version vollständig mit dem der gedruckten Version übereinstimmt. Ich bin mir bewusst, dass eine falsche Erklärung rechtliche Folgen haben kann.

Ort, Datum

Unterschrift

Zusammenfassung

Ziel des Projekts war es, eine Anwendung für mobile Endgeräte zu gestalten, die dem Benutzer einen spielerischen Zugang zum Musizieren und Experimentieren mit Klang bietet. Dieser Ansatz führte zu einem Entwicklungsprozess in zwei Teilen. Der theoretische Teil widmete sich dabei einerseits der musikwissenschaftlichen Betrachtung von Klang und Musik im Kontext elektronischer/digitaler Spiele. Andererseits wurden kultur- und medienwissenschaftliche Perspektiven auf das Phänomen des (Video-)Spielens untersucht. Im praktisch-gestalterischen Teil der Arbeit wurde, anhand der Software *MobMuPlat* und *Pure Data*, ein portables Musikspiel für Apples *iPad* entwickelt.

Abstract

The goal of the project was to create an application for mobile devices that offers the user a playful approach to music making and experimenting with sound. This concept resulted in a two-part development process. The theoretical part devoted itself, on the one hand, to the musicological assessment of sound and music in the context of electronic/digital games. On the other hand, the phenomenon of (video) gaming was examined from the perspectives of both cultural and media science. In the practical design part of the work a portable music game for Apple's *iPad* was developed using the software *MobMuPlat* and *Pure Data*.

Inhaltsverzeichnis

1. Einleitung	2
2. Musik und Klang in Videospielen	5
2.1 Chronologie und technologische Aspekte	6
2.2 Funktion und Wirkung	10
2.3 Kategorisierung und Möglichkeiten der Gestaltung	12
3. Zur Theorie des (Video-)Spielens	15
3.1 Game Studies	15
3.2 Gamification	17
3.3 Serious Games	22
4. Musikspiele	24
4.1 Geschichte und Entwicklung	24
4.2 Kategorisierung	28
4.3 Beispiele, Spezialfälle und besondere Eigenschaften	30
4.3.1 Rhythmus-Reaktionsspiele / Rhythm-Games	30
4.3.2 Klangproduzierende Spiele / kreative Musikspiele	31
4.3.3 Musician-Themed Games	32
4.3.4 Audiospiele / klangbasierte Spiele	33
4.4 Musikspiele als Forschungsgegenstand	35
5. Sampling	38
5.1 Geschichte, technische Implikationen und kulturelle Bedeutung	38
5.2 Sample-basierte Performance	41
6. „Project: Fingerhythm“ (Praktischer Teil)	43
6.1 Idee und Konzept	43
6.2 Erklärung und Aufbau des Spiels	44
6.3 Verwendete Software	55
6.4 Umsetzung und Programmierung	56
7. Zusammenfassung	63
8. Bibliografie	64
8.1 Anhang	68

1. Einleitung

Das übergeordnete Thema dieser Arbeit ist eine theoretische sowie praktische Auseinandersetzung mit Klang und Musik in Videospielen. Der theoretische Teil widmet sich dabei einerseits der musikwissenschaftlichen Betrachtung von Klängen im Kontext elektronischer Spiele. Andererseits werden kultur- und medienwissenschaftliche Perspektiven auf das Phänomen des (Video-)Spielens eröffnet. Im praktisch-gestalterischen Teil der Arbeit wurde ein portables Musikspiel für *Apples iPad* entwickelt.

* * *

Elektronische Unterhaltungsmedien im Allgemeinen und Videospiele im Speziellen erfreuen sich heutzutage weitreichender Popularität. Dies zeigt nicht nur die offizielle Eröffnung der diesjährigen *Gamescom*, der weltgrößten Messe für digitale Spiele, durch die deutsche Bundeskanzlerin Angela Merkel. In manchen Ländern ist Computerspielen inzwischen als eigenständige Disziplin (genannt: *E-Sports*) innerhalb der Sportarten anerkannt.

Entgegen der Annahme, dass der durchschnittliche Videospieler ein männlicher Teenager wäre, deuten Zahlen und Fakten (zumindest für den us-amerikanischen Markt) in eine andere Richtung:

According to the Entertainment Software Rating Board (ESRB) almost 41 percent of PC gamers are women. [...] While the market for teen and young adult men is still a strong market, it is the other demographics that are posting significant growth. Apparently, the average American gamer has been playing for 12 years, and is now, on average, 35 years of age. (Horowitz et al., 2014, S. 3)

Das sogenannte *Casual Gaming*, also ein Gelegenheitsspielen, sorgt neuerdings auf Smartphones und Tablets für Zeitvertreib zwischendurch und unterwegs. Klänge und Musik spielen dabei jedoch oftmals eine untergeordnete Rolle oder sind sogar gänzlich unerwünscht.

Auf der anderen Seite bergen diese tragbaren Endgeräte ein enormes Potential für die Entwicklung und Gestaltung klangbasierter Spiele und Anwendungen, wie auch schon vor knapp zehn Jahren prognostiziert wurde: „Audio

technologies and interactive applications on mobiles will change the way music is conceived, designed, created, transmitted and experienced.“ (Guerraz et al. 2008, S. 56)

Wichtige Erkenntnisse für die Arbeit von Sound Designern¹ und Entwicklern liefern u.a. die *Game Studies* - jene Disziplin, die sich der theoriegeleiteten Erforschung von Spielen widmet:

A knowledge of how audio works, why it is important, how musical assets fit into the scheme of a game, and how they are manipulated is crucial to the game development process, although it is rare for sound to be the actual primary feature of a game. (Parker et al., 2008, S. 1)

Zwar bilden Spiele mit Fokus auf Klang nur eine kleine Gruppe innerhalb der stetig wachsenden Industrie der elektronischen Unterhaltungsmedien - aber die große Fangemeinde und der kommerzielle Erfolg von Titeln wie *Guitar Hero* oder *Dance Dance Revolution* zeigen deutlich, dass Videospiele und die Popmusikultur schon seit langem voneinander profitieren.

* * *

Im ersten Kapitel erfolgt die Einleitung - die Kapitel zwei bis fünf bilden den theoretischen Teil dieser Arbeit. Kapitel 2 gibt Einblick in die geschichtliche Entwicklung von Musik und Klang in Videospiele und geht auf mögliche Funktionen und Wirkungen dieser Klänge ein. Weiters werden relevante Begriffe zu diesem Thema erläutert und Gestaltungsmöglichkeiten aufgezeigt.

Kapitel 3 betrachtet das Phänomen des Spielens im wissenschaftlichen Kontext. Neben den Themen *Gamification* und *Serious Games* wird über das Forschungsfeld der *Game Studies* berichtet.

In Kapitel 4 werden die Entwicklungsgeschichte der Musikspiele sowie Möglichkeiten der Kategorisierungen betrachtet. Zusätzlich werden besondere Fälle solcher Spiele und ihre Eigenschaften erläutert. Schließlich folgt ein Abschnitt über Musikspiele als Forschungsgegenstand.

¹ Aus Gründen der leichteren Lesbarkeit beziehen sich geschlechtsspezifische Formulierungen in dieser Arbeit (z.B. Spieler, Designer, Benutzer, etc.) stets in gleicher Weise auf Frauen und Männer.

Aufgrund der Relevanz für den praktischen Teil der Arbeit, berichtet Kapitel 5 von einer speziellen Form von sample-basierter Performance. Außerdem geht das Kapitel auf die Geschichte, die technischen Implikationen und die kulturelle Bedeutung von Sampling ein.

Kapitel 6 dient der Dokumentation des praktischen Teils dieser Arbeit. Dabei werden die Idee und das wesentliche Konzept hinter „Project: Fingerhythm“ sowie der Aufbau des Spiels erläutert. Weiters wird die verwendete Software beschrieben. Das Kapitel schließt mit einem Abschnitt zur konkreten Umsetzung und Programmierung.

* * *

2. Musik und Klang in Videospiele

Videospiele unterschiedlichster Form erfreuen sich heutzutage enormer Popularität in allen Altersgruppen und schon so mancher hat sich wohl dabei ertappt, die Lautstärke an der tragbaren Spielkonsole auf Maximalpegel zu stellen - oder gar Kopfhörer aufzusetzen - um noch tiefer in virtuelle Welten einzutauchen.

Für die meisten Benutzer tragen Soundeffekte und Musik offenbar nicht nur zur Atmosphäre eines Videospiele bei, sondern werden als essentielle Elemente empfunden, ohne die das gesamte Spielerlebnis leidet, oder sogar ganz verloren geht. Diese, sowie weitere Aspekte zur Funktion und Wirkung von Klängen in Videospiele, werden im Abschnitt 2.2 dieses Kapitels erläutert.

Von den ersten „Piepsern“ der Spielautomaten, die zu Beginn der 1970er in den Spielhallen aufgestellt wurden, über den typischen Chip-Klang der Heimkonsolen, die in den 80ern die Wohnzimmer der westlichen Welt eroberten, bis hin zur aufwendigen Produktion orchestraler Soundtracks zu aktuellen *AAA-Spielen*² - Abschnitt 2.1 des Kapitels behandelt die chronologische Entwicklung dieser Klänge und die Technologie hinter der Klangerzeugung.

Um Spielumgebungen akustisch abwechslungsreich zu gestalten, bedienen sich Sound Designer und Komponisten heutzutage spezieller Techniken und Strategien. Dabei entstehen beispielsweise dynamische und adaptive Klänge, die sich den Handlungsweisen eines Spielers, oder dem Verlauf des Spiels selbst anpassen und verändern können. Diese und weitere Möglichkeiten der Gestaltung von Klang und Musik in Videospiele werden im Abschnitt 2.3 beschrieben.

² Sogenannte Triple-A-Spiele sind Videospiele mit äußerst hohen Produktionskosten (bis zu mehreren 100 Mio. US-Dollar pro Titel), die meist auch durch spezielle Werbekampagnen vermarktet werden. Vgl. dazu: [https://de.wikipedia.org/wiki/AAA_\(Videospieleindustrie\)](https://de.wikipedia.org/wiki/AAA_(Videospieleindustrie)).

2.1 Chronologie und technologische Aspekte

Als zu Beginn der 1970er die ersten, kommerziell erfolgreichen Videospieleautomaten entwickelt wurden, war die Speicherkapazität der Chips noch sehr gering, weshalb nur wenige (und simple) akustische Effekte generiert werden konnten:

[...] the early arcade games had only a short introductory and „game over“ music theme, and were limited to sound effects during gameplay. Typically the music only played when there was no game action, since any action required all of the system's available memory. (Collins, 2008a, S. 11)

Die Beschränkung auf simple Piepstöne und perkussionsartige Geräusche der Maschinen sollte die Entwickler nicht davon abhalten, diese von Anfang an sehr geschickt einzusetzen - außerdem sollte durch die Klänge die Neugier potentieller Spieler geweckt werden. (vgl. Collins, 2008a, S. 11)

Trotz der technischen Einschränkungen und des massiven Zeitaufwands für die Programmierung, wies das 1978 erschienene Spiel *Space Invaders* bereits Anzeichen dynamischer Hintergrundmusik auf - dabei steigert sich das Tempo der Musik, abhängig vom Fortschritt des Spielers. (vgl. Collins, 2008b, S. 2)

Zu dieser Zeit wurden auch die ersten Spielkonsolen für den Heimgebrauch veröffentlicht: Sowohl das *Channel F* - System (1976) als auch die *Studio II* - Konsole (1977) waren in der Lage, verschiedene Spielprogramme von externen Datenträgern abzuspielen und einfache Töne zu generieren. (vgl. Fritsch, 2013, S. 12) Die akustische Signalverarbeitung bei diesen Systemen - wie auch bei vielen Spielautomaten - erfolgte mithilfe von programmierbaren Klanggeneratoren, genannt *PSGs*:³

Early PSGs used analogue synthesis, or subtractive synthesis, which starts with a waveform created by an oscillator, and uses a filter to attenuate or subtract specific frequencies. It then passes this through an amplifier to control the envelope and amplitude of the final resulting sound. PSGs offered little control over the timbre of a sound, usually limiting

³ Abkürzung aus dem Englischen für „programmable sound generators“.

sounds to single waveforms (typically square waves) without much ability to manipulate that waveform. (Collins, 2008b, S. 3)

1980 erschien das Spiel *Rally-X*, in dem erstmals geloopte Hintergrundmusik zum Einsatz kam (vgl. Scherer, 2015, S. 17) und noch im selben Jahr eroberte Tohru Iwatanis *Pac-Man* die Spielhallen. *Pac-Man* war nicht nur eines der ersten Spiele, das eine Zwischensequenz beinhaltete - auch der „catchy soundtrack“ sollte dem Titel zu enormer Popularität verhelfen. (vgl. Collins, 2008b, S. 2)

1982 wurde der kommerziell sehr erfolgreiche C64 von Commodore veröffentlicht und von Beginn an als „Spiele-Computer“, inklusive eines speziell entwickelten Soundchips, vermarktet:

The sound chip [...] was a three-voice plus noise generator chip [...]. Unlike other PC chips at the time, each tone on the chip could be selected from a range of waveforms - sawtooth, triangle, pulse and noise. Each tone could also be subjected to a variety of effects and programmable filters including ring modulation. (Collins, 2008b, S. 3)

Aufgrund des technischen Fortschritts konnten also nun bereits mehrstimmige Klänge und unterschiedliche Klangfarben für Videospiele erzeugt werden und allmählich etablierten sich auch japanische Hersteller und Entwickler wie Nintendo auf internationaler Ebene. Mitte der 1980er wurden zwar noch immer Videospielautomaten hergestellt, doch allmählich wurden diese von Heimcomputern und Spielkonsolen verdrängt.

Mit Nintendos Veröffentlichung der 8-bit-Konsole *Famicom* (1985 in den USA erschienen als *Nintendo Entertainment System*, kurz: *NES*), die eine ganze Generation von Spielern prägen sollte, wurde auch der Klang der Videospiele weiterentwickelt: „The NES used a built-in five-channel sound chip with one waveform form each channel - two pulse waves, a triangle wave, a noise and a sample channel.“ (Collins, 2008b, S. 3) Dass für die Gestaltung eines Spiele-Soundtracks - nämlich für *Super Mario Bros.* - ein professioneller Komponist (Koji Kondo) engagiert wurde, war zu dieser Zeit einzigartig. (vgl. Scherer, 2015, S. 19f)

Nach und nach konnten neue ästhetische Entscheidungen verwirklicht und alternative kompositorische Möglichkeiten ausgelotet werden:

Most NES songs consisted of a melody, thickened out with a second channel, a bass line and percussion. [...] but some composers explored other possibilities, including Hirokazu ‚Hip‘ Tanaka’s soundtrack for the *Metroid* game, which intentionally avoided melody-based songs, instead opting for a science-fiction film music style, where sound effects and song blur together to create an eerie atmosphere. (Collins, 2008b, S. 4)

Durch die Entwicklung der ersten FM-Soundkarten für Heimcomputer, sowie die Etablierung des MIDI-Standards Mitte der 1980er, waren der Gestaltung von Klängen für Spiele kaum noch technische Grenzen gesetzt und mit dem Aufkommen der CD-Technologie verbesserte sich zusätzlich die Klangqualität. (vgl. Collins, 2008b, S. 4)

Obwohl das Komponieren via MIDI äußerst praktikabel war - zum einen konnte Musik nun am Keyboard geschrieben werden, zum anderen benötigt MIDI-Code nur sehr wenig kostbaren Speicherplatz - klang das Ergebnis, je nach Soundkarte des Spielers, unterschiedlich. (vgl. Collins, 2008a, S. 50)

Den Start der 16-bit-Ära der Spielkonsolen markierte 1989 die Firma Sega mit ihrer *Genesis* (in Europa: *MegaDrive*). Zu einem PSG-Chip, ähnlich dem des NES, besitzt das System einen Yamaha FM-Synthesizer Chip inklusive sechs Kanälen für Stereoklang, womit bereits Musik in CD-Qualität wiedergegeben werden konnte. (vgl. Collins, 2008b, S. 4-5)

Nintendo brachte zwei Jahre später ebenfalls ein 16-bit-System auf den Markt - nämlich das *Super Nintendo Entertainment System* (SNES). Mit einem von Sony entwickelten Soundchip, bestehend aus mehreren Komponenten, wollte man die *Genesis* technisch übertrumpfen. Collins (2008b) beschreibt das Herzstück des Chips (*SPC-700*), das auch als Koprozessor mit eigenem Speicher fungiert, folgendermaßen:

It was essentially a wavetable synthesizer that supported eight stereo channels at programmable frequency and volume, and effects such as reverb, filters, panning and envelope generators, and with a preset stock

of MIDI instruments. Wavetable synthesis uses preset digital samples of instruments, usually combined with basic analogue waveforms, creating more ‚realistic‘ instrument sounds. (S. 5)

Als Mitte der 1990er Sonys *Playstation* veröffentlicht wurde, konnten mit den Soundchips auch Echtzeit-Audioeffekte realisiert werden - manche Spiele machten es sogar möglich, eigene Audio-CDs als Hintergrundmusik laufen zu lassen. Einige der Spiele zu Nintendos 64-bit Konsole *N64* (1996) unterstützten schließlich Sourround-Klang, was durch einen speziellen Verstärker optimiert werden konnte. (vgl. Collins, 2008b, S. 5)

Collins bringt die Vorteile der CD-ROM-Technologie, wodurch sich in den 1990er Jahren vieles in der Videospiegelindustrie änderte, auf den Punkt: „[...] CD-ROM technology ensured that composers and sound designers could not only know how the audio would sound on most consumer configurations; they could also now record sound effects, live instruments, vocals, and in-game dialogue.“ (2008a, S. 63)

Um die Jahrtausendwende erschienen die ersten 128-bit Spielkonsolen, wie Sonys *Playstation 2*, oder Microsofts *Xbox*. Diese Systeme sind in der Lage, Klang in DVD-Qualität über mehrere Kanäle wiederzugeben, was eine neuerliche Erweiterung der Speicherkapazität sowie höhere klangliche Auflösung und schnellere Abtastgeschwindigkeiten bedeutete.

Wie sich anhand dieser kurzen Zusammenfassung der Entwicklung von Klang und Musik in Videospiele ablesen lässt, stellt der technologische Fortschritt Gestalter und Entwickler immer wieder vor neue Herausforderungen. Heutzutage beschäftigt sich die Forschung vor allem mit 3D-Klangsystemen für virtuelle (oder erweiterte) Realitäten, Konzepten zur Erstellung adaptiver und generativer Musik für unterschiedliche Spielplattformen, sowie den Möglichkeiten der Klanggestaltung für Spiele auf mobilen Endgeräten, wie Smartphones und Tablets.

2.2 Funktion und Wirkung

Die Funktionsweisen und möglichen Wirkungen von Musik und Klängen auf den Spieler können, je nach Genre des Spiels, variieren. Grundsätzlich lässt sich festhalten, dass: „In the context of computer games, audio has clear usability functions in addition to supporting a specific mood and the sense of presence in the game environment.“ (Jørgensen, 2008, S. 163)

Stingel-Voigt präsentiert in ihrer Monografie „Soundtracks virtueller Welten. Musik in Videospiele“ (2014) zahlreiche Erklärungsansätze zu den Funktionen von Musik in virtuellen Umgebungen, stützt sich dabei allerdings auf keine empirischen Studien zum Thema. Dieser Umstand wird auch von Stoppe in einem Review zum genannten Buch kritisiert. (vgl. 2015, S. 287)

Die folgende Liste⁴ soll einen Überblick zu möglichen Wirkungen und Funktionen von Musik in Videospiele geben:

- 1) Ästhetische Funktion: Musik trägt zur gesamten ästhetischen Erscheinung eines Spiels bei.
- 2) Die Musik ist Bedeutungsträgerin und hat eine Signalwirkung.
- 3) In Zwischensequenzen hat Musik vor allem eine begleitende Funktion.
- 4) Musik kann helfen, bestimmte Stimmungen zu erzeugen - dabei spricht man von funktionaler Musik. Es wird Musik eingesetzt, die am besten zu Szenerie und Handlung passt.
- 5) Anhand von Erinnerungs- und Leitmotiven kann Musik zum Verständnis der Handlung und zur Orientierung innerhalb der Spielwelt beitragen.
- 6) Narrative Funktion: Musik kann der Handlung eine Struktur geben und so mithilfe von wiederkehrenden Themen räumlich und zeitlich getrennte Ereignisse miteinander verknüpfen.
- 7) Perzeption: Musik kann die Aufmerksamkeit eines Spielers stärker auf bestimmte Geschehnisse lenken.
- 8) Mithilfe von Musik kann die Motivation des Spielers gefördert werden.

⁴ Diese Übersicht wurden von Stingel-Voigt (2014, S. 66f) übernommen und vom Autor teilweise gekürzt und adaptiert.

9) Die Funktion von Musik als Hilfsmittel / Werkzeug⁵

10) In Musizierspielen wird Musik selbst zum Spielinhalt. Rhythmus und Tempo können dabei den Schwierigkeitsgrad bestimmen und auf die körperlichen Fähigkeiten der Spieler Einfluss nehmen.

Collins (2008a) und andere Forscher weisen darauf hin, dass gewisse Konzepte und Theorien zur Untersuchung von Filmmusik auch bei der wissenschaftlichen Betrachtung von Videospielemusik nützlich sein können. Die Tatsache, dass Handlungsstränge in Spielen aber meist non-linear verlaufen und dass die Unterscheidung von diegetischem⁶ und non-diegetischem Einsatz von Klang nicht immer eindeutig möglich ist, macht die Sache kompliziert. (vgl. S. 125)

Jørgensen (2008) untersuchte in mehreren empirischen Studien die Reaktionen und Gefühle von Probanden, die Videospiele spielten - wobei Klang und Musik während des Spielens plötzlich verstummten und die Anordnung lautete, trotzdem weiter zu spielen:

On the one hand, the usability of the system decreased, since the players did not receive any responses [...] regarding their own actions and commands. Likewise, the players received no warnings or alerts from the system about abnormalities. [...] On the other hand, the mood, sense of presence and the feeling of lifelike world disappeared, and the games revealed themselves as nothing but animated graphics on a screen. (S. 175)

Diese Ergebnisse deuten darauf hin, dass Klang in Videospiele einerseits als Orientierungshilfe - sowohl in Spielmenüs, als auch in der virtuellen Umgebung - dient. Zum anderen wirkt akustisches Feedback als Bestätigung, dass Handlungen ausgeführt wurden oder auch Befehle (innerhalb des Spiels) weitergeleitet wurden. Schließlich tragen Klänge und Musik offenbar zum Empfinden bei, sich tatsächlich in der jeweiligen Spielwelt zu befinden. Im nun folgenden Abschnitt soll auf dieses Thema noch konkreter eingegangen werden.

⁵ Im Spiel *The Legend of Zelda - Ocarina of Time* (Nintendo 64, 1998) muss der Spieler selbst musizieren, um beispielsweise Rätsel zu lösen. Vgl. dazu Scherer (2015, S. 55).

⁶ Eine weiterführende Erklärung folgt im Abschnitt 2.3 dieser Arbeit.

2.3 Kategorisierung und Möglichkeiten der Gestaltung

Zuerst soll dargelegt werden, in welcher Form Klänge und Musik in Videospielen vorkommen. Parker und Heerema (2007) schlagen dazu eine einfache Unterteilung in vier Kategorien vor: Musik, Sprache, Soundeffekte und *Input*. (vgl. S. 217) Die letzte Kategorie bezieht sich auf Klang (vornehmlich Sprache) als Eingabeform. Gemeint ist damit die Eingabe von akustischen Befehlen über ein Mikrofon, um etwas im Spiel auszulösen oder zu steuern. Zwar gibt es Spiele mit Spracherkennungs-Systemen, doch diese sind eher die Ausnahme⁷. Warum diese Kategorie hier aufscheint hat offenbar damit zu tun, dass die Forscher selbst eine Software (*Visual Music*) zur Klangerkennung entwickelt haben. (vgl. 2007, S. 219)

Collins (2013) nennt als klangliche Aspekte von Videospielen: „[...] discrete sound effects, ambient sound beds, dialog, music, and interface sounds.“ (S. 3) Demnach unterteilt die Forscherin Parkers „Soundeffekte“ in „diskrete Soundeffekte“ und „Interface Klänge“ - außerdem unterscheidet sie bei Musik zwischen „Musik“ und „Klangflächen“. Der Aspekt „Dialog“ entspricht Parkers „Sprache“.

Als Sound Designer gilt es, all diese Aspekte (je nach Genre und Atmosphäre des Spiels) zu einem stimmigen Konzept zusammen zu fassen, um das bestmögliche Gesamterlebnis für den Spieler zu gestalten, ohne diesen „akustisch zu überfordern“, oder zu langweilen. Die Klänge und Musik sollten dabei mit der grafischen Umgebung, der Geschichte (falls vorhanden) und den Spielmechaniken zusammenpassen, sofern nicht anders intendiert.

Collins (2008a) weist auf den Umstand hin, dass Videospiele (im Unterschied zu Filmen) non-linear verlaufen und eine dementsprechende Klanggestaltung, sowie konkrete Unterscheidungen von Begriffen notwendig machen. (vgl. S. 125f) Als nächstes werden einige dieser Begriffe erklärt, die für das Verständnis der oftmals komplexen Zusammenhänge zwischen dem Spieler (dessen Aktionen, Reaktionen und Interaktionen) und den klanglichen Elementen im Spiel wichtig sind.

⁷ Im Spiel *The Legend of Zelda - Phantom Hourglass* (Nintendo DS, 2007) beispielsweise, muss man an bestimmten Stellen der Geschichte ins Mikrofon der Konsole rufen bzw. pfeifen, damit die Handlung vorangetrieben wird.

Synchrese: Der Begriff stammt aus der Filmforschung und ist eigentlich ein Kunstwort, das die Begriffe Synthese und Synchronität verbindet. Damit gemeint ist das kognitive Phänomen einem Klang, der von seiner (visuellen) Quelle losgelöst wurde, gewissermaßen automatisch eine neue Bedeutung zu geben, sobald er in Verbindung mit einer neuen Bildquelle auftritt. (vgl. Collins, 2013, S. 20)

Mickey-Mousing: Unter Mickey-Mousing versteht man eine konkrete Technik von Synchrese, die aus frühen Animationsfilmen stammt: „Mickey mousing dates back to the time when pianists were accompanying silent movies. The technique does not involve samples of the physical object depicted in the game, but an imitation or exaggeration of the sound this object would produce.“ (Pichlmair et al., 2007, S. 4) Demnach kann auch der Soundeffekt von Marios⁸ Sprung als Mickey-Mousing bezeichnet werden, wie Whalen analysiert hat. (vgl. 2014)

Diegetische / Non-diegetische Musik: Auch dieser Begriff stammt aus der Filmwissenschaft und kann bis zu einem gewissen Grad auf Klang in Videospielen angewendet werden. Grundsätzlich meint non-diegetisch alle Klänge, die eine Szene nur begleiten, also von außerhalb des jeweils sichtbaren Geschehens stammen. Im Videospiel wäre das beispielsweise die Musik während eines Intros, oder einer Zwischensequenz. Von diegetischem Klang spricht man, wenn die Quelle des Klangs in einer Szene sichtbar ist. Im Film könnte das ein Radiogerät, oder ein gerade gespieltes Musikinstrument sein - bereits hier kann der Übergang zwischen diegetischem und non-diegetischem Einsatz allerdings verschwimmen, wenn man z.B. in einer Szene ein Lied hört, die Kamerafahrt allmählich ein Radio erscheinen lässt und sich herausstellt, dass das Lied offenbar im (zuerst nicht sichtbaren) Radio gespielt wird. In einem Spiel wiederum, könnte man sich eine Szene vorstellen, in der zuerst ein Instrument aus der Ferne (nicht sichtbar, also non-diegetisch) erklingt und man sich der Musik allmählich nähert (jetzt sichtbar, also diegetisch) - beim Instrument angekommen, könnte man dieses nun sogar selbst bedienen - die Musik ändert sich also von non-diegetisch zu diegetisch und gleichsam von non-dynamisch zu interaktiv. (vgl. Collins, 2008a, S. 125)

⁸ Das Spiel *Super Mario Bros.* (Nintendo, 1985) zählt noch heute zu den einflussreichsten Videospielen im Jump'n'Run Genre.

Klänge, die vom Spieler ausgelöst bzw. hervorgerufen werden, die Handlung des Spiels aber nicht beeinflussen, bezeichnet man als non-diegetisch interaktiv. Ein typisches Beispiel dafür sind Klänge, die im Spielmenü ertönen. Ein weiteres Beispiel wäre der Wechsel von einem Musikstück auf ein anderes, sobald der Spielcharakter einen Raum betritt. (vgl. Collins 2008a, S. 125)

Adaptive Klänge: „Adaptive audio [...] occurs in reaction to game-play, in terms of being set to occur in response to timings or other parameters set by the game’s engine, rather than directly in response to the player.“ (Collins, 2008b, S. 8) Man spricht also von adaptiver Musik, wenn sich z.B. zu einer bestimmten Tageszeit innerhalb des Spielverlaufs die Musik ändert.

Dynamische / Non-dynamische Musik: Ein non-dynamisches Musikstück bleibt schlicht immer gleich. Es macht dabei keinen Unterschied, wie oft, oder an welcher Stelle der Handlung eines Spiels das Stück erklingt, oder wie sich der Spieler in der virtuellen Umgebung verhält. Dynamische Musik hingegen ist so programmiert, dass sie sich durch verschiedenste Einflüsse des Spielers auf das Spielgeschehen verändern kann. Heutzutage gibt es zahlreiche Praktiken⁹ um derart klangliche Variabilität zu gewährleisten und aufgrund der Tatsache, dass die virtuellen Umgebungen immer größer, komplexer und offener¹⁰ werden, wächst auch das Forschungsfeld rund um die Möglichkeiten der Gestaltung von dynamischer Musik.

Klassische Praktiken um Klang für Videospiele abwechslungsreicher zu gestalten sind beispielsweise das Variieren von oft erklingenden Samples, wie Schritte. Dafür lassen sich verschiedene, ähnlich klingende Samples einsetzen, die zufällig ausgewählt werden. Möglich für einen solchen Effekt wäre außerdem eine leicht abweichende Filterung in Echtzeit. Komplizierter wird es bei der Erstellung ganzer Musikstücke, doch auch dafür gibt es Lösungsansätze, wie Jesper Kaae in „Theoretical approaches to composing dynamic music for video games“ (2008) beschreibt. (vgl. S. 75f) Collins zeigt ebenfalls zahlreiche Möglichkeiten zur Erstellung dynamischer Musik auf.

⁹ Einige davon werden in Karen Collins Sammelband „From Pac-Man to Pop Music“ (2008b) ausführlich beschrieben. Vgl. dazu die Kapitel 4, 5, 6 und 9.

¹⁰ In sogenannten Open-World-Spielen ist der Spieler nicht an eine vorgegebene Reihenfolge von Aufgaben gebunden und kann den Spielverlauf somit zu großen Teilen selbst bestimmen. Vgl. dazu: <https://de.wikipedia.org/wiki/Open-World-Spiel>

Wichtig hierfür ist die Variation folgender musikalischer Parameter: Tempo, Tonhöhe, Rhythmus bzw. Metrum, Lautstärke und Klangfarbe. Außerdem bieten sich variable Melodien und Harmonien, sowie adaptives Mixing an. (vgl. 2008a, S. 147)

3. Zur Theorie des (Video-)Spielens

Diese Kapitel beinhaltet Vor- und Feststellungen, die sowohl Spiele im Allgemeinen, als auch Videospiele im Speziellen betreffen. Im folgenden Abschnitt (3.1) geht es vor allem um kulturtheoretische Implikationen zu diesen Themen. Abschnitt 3.2 berichtet über Gamification, einem sehr jungen Phänomen, das in der Wissenschaft mitunter heiß diskutiert¹¹ wird. In Abschnitt 3.3 wird schließlich ein kurzer Einblick in *Serious Games* und die Aspekte des spielerischen Lernens gegeben.

3.1 Game Studies

Computerspiele haben sich als ernstzunehmender Forschungsgegenstand in der Medienwissenschaft etabliert. Mit den Game Studies hat sich eine eigene Subdisziplin gebildet, die sich durch eine wachsende Interdisziplinarität von Medienwissenschaft und Informatik, aber auch von anderen Wissenschafts- disziplinen auszeichnet. (Stoppe, 2015, S. 286)

Raessens (2012) verweist in seinem Beitrag „The Ludification of culture“ auf den kulturellen Stellenwert des „Spielens“ und wie sich dieser seit Mitte des vorigen Jahrhunderts gebildet und weiterentwickelt hat:

Since the 1960s, when the word *ludic* became popular to denote playful behaviour and fun objects [...] playfulness has gradually become a central category of our culture. [...] Play is not only characteristic of leisure, but also turns up in those domains that once were considered the opposite of play, such as education (e.g. educational games), politics [...] and even warfare [...]. (S. 94)

Im Kontext der Game Studies ist es wichtig, die Begriffe *gaming* und *playing* als verschiedene Spielweisen zu unterscheiden, die sich aber nicht gegenseitig

¹¹ Vgl. dazu den kürzlich erschienen Artikel „Gamification: What it is, and how to fight it“ von Woodcock & Johnson (2017).

ausschließen. Eine Vielzahl von Autoren weist in diesem Punkt auf zwei essentielle Veröffentlichungen hin: Zum einen, das Buch „Homo Ludens“ (1938) von Johan Huizinga - zum anderen, das Werk „Les jeux et les hommes“ (1958) des Autors Roger Caillois. Huizinga unternimmt den Versuch eines Erklärungsmodells des Menschen, wonach sich dessen Fähigkeiten und entwickelte Eigenschaften vor allem durch das Spielen manifestieren würden. Demnach hätte das Spiel einen entscheidenden Einfluss auf die kulturelle Entwicklung des Menschen. (vgl. Raessens, 2012, S. 100ff) Caillois baut in gewisser Hinsicht auf die Theorien von Huizinga auf, kritisiert jedoch unter anderem dessen zu allgemeine Definition des Spielens.¹² Caillois führt deshalb die Begriffe *Paidia* und *Ludus* als grundsätzliche verschiedene Spielweisen in die Diskussion ein.

Paidia meint dabei das freie, spontane, improvisierte Spiel - angetrieben durch die Fantasie und den Erkundungsgeist der Mitspieler. Der Begriff *Ludus* hingegen bezeichnet die Spielweise nach Regeln - ein strukturiertes Spielen, oftmals geprägt durch den Wettbewerb zwischen den Mitspielenden - dabei benötigen die Teilnehmenden mitunter bestimmte Fähigkeiten und/oder verwenden spezielle Strategien, um Ziele zu erreichen. (vgl. Denk, 2011, S. 9f)

Für die Arbeit von Spielentwicklern und Designern ist diese Unterscheidung sicherlich von Bedeutung. Die zwei Konzepte stehen sich zwar grundsätzlich gegenüber, doch die Grenzen können (insbesondere bei Videospiele) verschwimmen, wie auch Halbeisen (2011) anmerkt und sich dabei u.a. auf die Wissenschaftlerin und Spieldesignerin Jane McGonigal bezieht:

Während in Gesellschaftsspielen die Regeln und Ziele im Voraus geklärt werden, gilt es diese im oft [sic!] Computerspiel erst zu erforschen. Es kann also sein, dass während des Spiels neue Regeln entdeckt oder auch bewusst eingeführt werden. [...] Durch den explorativen Charakter dieser Selbsterforschung von Regeln und Zielen kommt es zu einer Verbindung von *Play* und *Game* im Computerspiel. Da die definierenden Rahmenbedingungen von *Game* noch nicht geklärt sind, müssen diese von dem/der SpielerIn im *Play* erst eruiert werden. Computerspiele brauchen daher keine Instruktionen und sind für jede/n zugänglich. (S.16)

¹² Eine ausführliche Auseinandersetzung mit dem Thema findet sich bei Denk (2011, S. 5ff).

Auch wenn *paidia* und *ludus* noch weiter differenziert werden können, entsprechen die von Halbeisen im vorigen Zitat verwendeten Begriffe *play* und *game* den beiden inhaltlich - das zeigt sich auch in ihrer (von Adamowsky übernommenen) Definition: „*Play* bezeichnet [...] ein nicht-zweckgebundenes und intentionsloses Verhalten, das durch die spontan entstandene Teilnahme an einer lustvollen und kreativen Spielsituation entstanden ist. Im Unterschied dazu ist das *Game* eine klar strukturierte Spielform, die strategisches Handeln hervorbringt.“ (2011, S. 15)

Studien zu Spieldesign im Allgemeinen und Videospieldesign im Speziellen sind noch junge Forschungsthemen, doch ihre Anzahl wächst stetig. An der Universität Utrecht wurde 2005 das Projekt „Playful Identities“¹³ ins Leben gerufen, um den Einfluss digitaler Technologien auf kulturelle Entwicklungen zu untersuchen. Der Forscher Joost Raessens erklärt dazu:

Computer games and other digital technologies such as mobile phones and the Internet seem to stimulate playful goals and to facilitate the construction of playful identities. This transformation advances the ludification of today's culture in the spirit of Johan Huizinga's *homo ludens*. (2006, S. 52)

3.2 Gamification

Die Verwendung des Begriffs *Gamification* tauchte erstmals vor wenigen Jahren im Kontext der Digitalen Medienindustrie auf und breitet sich seither auf zahlreiche Gesellschafts-, Industrie-, und Marketingbereiche aus. (vgl. Deterding et al., 2011a, S. 1) Das Prinzip dahinter kann, wie Sebastian Deterding vorschlägt, folgendermaßen definiert werden: „Gamification is the use of game design elements in non-game contexts.“ (2011a, S. 2) Allgemein handelt es sich also um ein Gestaltungsprinzip, welches bewusst auf die Entwicklung von Produkten, oder Dienstleistungen angewendet wird. Ziel bei diesem Ansatz ist es, die Benutzererfahrung und Motivation auf positive Art zu beeinflussen:

¹³ vgl. den Artikel „Playful Identities or the Ludification of Culture“ (Raessens, 2006).

During the last couple of years, gamification has been a trending topic and a subject to much hype as a means of supporting user engagement and enhancing positive patterns in service use, such as increasing user activity, social interaction, or quality and productivity of actions. These desired use patterns are considered to emerge as a result of positive, intrinsically motivating, „gameful“ experiences brought about by game/motivational affordances implemented into a service. (Hamari et al., 2014, S. 3025)

Die Prinzipien und Ansätze hinter dem Phänomen Gamification werden schon seit über 30 Jahren untersucht, nämlich im Bereich der Human-Computer Interaction (HCI): „Attempts to derive heuristics for enjoyable interfaces from games reach back to the early 1980s [...]. More recently, researchers have tried to identify design patterns that might afford joy of use under the moniker „funology“, explicitly drawing inspiration from game design.“ (Deterding et al., 2011b, S. 2426)

Was hier bereits angedeutet wird, ist der wichtige Zusammenhang zwischen spielerisch-motivierenden Gestaltungsansätzen und Erkenntnissen auf dem Gebiet der Game Studies. Demnach bildet das Wissen über Spielgestaltung ein gut brauchbares Fundament, um Gamification sowohl thematisch einzugrenzen, als auch von Entwicklern gezielt einsetzen zu können.

Deterding gibt zu bedenken, dass die Interpretation des Begriffs Gamification innerhalb der digitalen Medienindustrie noch immer ziemlich kontrovers diskutiert wird und verweist dabei unter anderem auf eine Präsentation von Jane McGonigal im Rahmen der *Game Developers Conference 2011* mit dem Titel: „We don't need no stinkin' badges: How to re-invent reality without gamification. (vgl. 2011a, S. 1)

Damit sich Gamification als eigener Forschungsbereich etablieren kann und bis Konsens bezüglich der Definitionen herrschen kann, muss noch viel an systematischer Forschungsarbeit geleistet werden:

We know extremely little about the actual effect of ‚player types‘, and the effectiveness of designing with player types in mind, let alone individual differences beyond them. And all of that says nothing yet about the relative

impact of person versus situation on the effects of gamification, let alone potential interaction effects of the two. (Nacke et al., 2017, S. 4)

Hier offenbart sich eine wichtige Verbindung zur ebenso jungen Disziplin der Erforschung von Klang in Videospiele, wie Collins (2013) treffend bemerkt: „[...] although academic writing about game sound is growing, we still often miss a fundamental piece of the puzzle that is essential to any theoretical account - the player.“ (S. 2)

Für die Kategorisierung der Begriffe ist zu beachten, dass sich Gamification vom Prinzip der *Serious Games*¹⁴ unterscheidet und auch von *Playful Interaction* zu trennen ist. Zusätzlich soll das Verhältnis zu *Gaming* und *Playing* verdeutlicht werden, wie Abb.1 zeigt. Die Grafik zeigt außerdem an, dass sich Gamification nur Elementen aus Spielen bedient, nicht ganze Spielsysteme übernimmt.

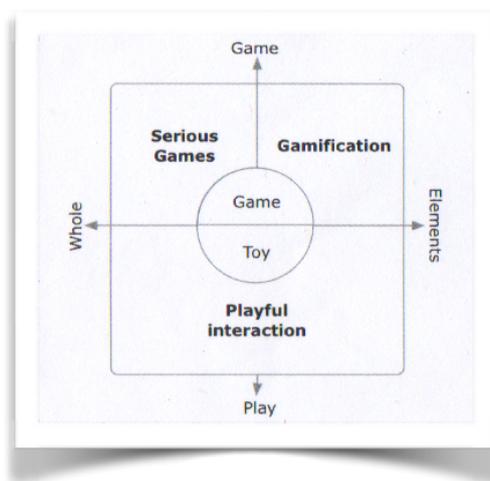


Abb.1: Die Positionierung von Gamification nach Deterding et al. (2011a)

Einige der typischen Elemente, welche die Benutzererfahrung positiv beeinflussen und beispielsweise soziale Interaktionen fördern sollen, finden sich in Abb.2. So werden Benutzer von *Foursquare* z.B. mit Abzeichen (sogen. „badges“) belohnt, wenn sie sich an bestimmten Standorten aufhalten. „Leaderboards“, also Ranglisten, werden gerne verwendet, um das Gefühl eines Wettbewerbs zu vermitteln.

¹⁴ Unter *Serious Games* versteht man (meist digitale) Spiele, die nicht vorrangig zu Unterhaltungszwecken, sondern auch zu Bildungszwecken entwickelt werden. Vgl. dazu auch Raessens (2012, S. 107f), Halbeisen (2011, S. 44f), sowie Abschnitt 3.3 dieser Arbeit.



Abb.2: Typische Gamification-Elemente nach Deterding (2012)

Die Bereiche, in denen Gamification eingesetzt wird, sind inzwischen sehr zahlreich: So findet man Anwendungen im Produktionssektor, dem Finanzmarkt, sowie auf den Gebieten Bildung, Gesundheit/Fitness, Nachhaltigkeit, im Nachrichtenwesen, oder beim Angebot von Onlinetutorials.

Eines der bekanntesten Beispiele ist die Anwendung *Foursquare* - dabei handelt es sich um ein standort-basiertes Onlineservice, welches den Nutzern Restaurants, Bars, Clubs, etc. in Abhängigkeit von deren Aufenthaltsort empfiehlt: „Players are expected to check-in to locations while seeking rewards and virtual points. [...] It is location-based and utilises gamification elements such as leader boards, levels, badges and points to engage users into revisiting locations [...].“ (Bozic, 2014, S. 13)

Um den Bezug zur vorliegenden Arbeit zu verdeutlichen, sollen nun Anwendungen beschrieben werden, welche die Verknüpfung von Sonic Interaction Design mit Gamification-Elementen repräsentieren. Dabei ist es wichtig anzumerken, dass die strikte Definition von Deterding („Gamification is the use of game design elements in non-game contexts“) hier teilweise aufgebrochen werden muss. Ob nämlich ein Produkt/Service/Anwendung als „Spiel“, als „Anwendung mit spielerischen Design-Elementen“, oder beispielsweise als „Virtuelles Musikinstrument mit intuitiv-spielerischem Interface“ empfunden wird, hängt letztlich von den Benutzern ab.

Deterding ist sich dessen bewusst und formuliert dazu: „Of course, the boundary between *game* and *artifact with game elements* can often be blurry: Is foursquare a game or a gamified application? Is the purpose of foursquare primarily for entertainment and fun, or for something else? [...] The addition of one informal rule by a group of users may turn a gamified application into a complete game.“ (2011a, S. 2)

Generell wird Sonic Interaction Design als spezielle Nische des Interaktionsdesigns bzw. UX-Designs verstanden. Bisher lag der Forschungsschwerpunkt meist auf der visuellen Ebene von Interfaces, doch seit einiger Zeit rückt das Auditive mehr und mehr in den Vordergrund. (vgl. Gharibpour, 2015, S1f)

Der Forscher und Entwickler Ge Wang hat im Rahmen der *NIME*-Konferenz¹⁵ 2016 einen Artikel veröffentlicht, welcher „Game Design for Expressive Mobile Music“ in den Fokus rückt. Darin werden verschiedene musikalische Anwendungen für mobile Geräte untersucht und verglichen - es zeigt sich, dass viele der Applikationen auch Gamification-Elemente beinhalten und den musikalischen Ausdruck in den Vordergrund stellen. In der Einleitung formuliert der Autor treffend:

Games are no stranger to music. For example, in the practice and mastery of traditional instruments [...], game-like strategies can be employed to instruct, overcome difficulties and undesirable habits, motivate and provide long-term structure in mastering a particular aspect of an instrument. (2016, S. 1)

In kurz gehaltenen Case Studies werden fünf Anwendungen der Firma *Smule*¹⁶ vorgestellt und deren Gestaltungsansätze erklärt. Grundsätzlich basiert die Entwicklung dieser Applikationen auf drei Beobachtungen bzw. Annahmen, welche im Vorfeld vom Entwickler-Team klar formuliert wurden:

1) Whereas musical instruments are generally viewed to be „*specialized*“, games are viewed as playable by potentially anyone. There is a lower

¹⁵ Die Abkürzung NIME steht für New Interfaces for Musical Expression - weiterführende Informationen finden sich unter: <http://www.nime.org/>.

¹⁶ Vgl. dazu: <https://www.smule.com/about>.

perceived barrier of entry with games (or more relevantly, with things that are *perceived* as games).

2) Games can be designed to provide clear, attractive, and attainable goals [...] for engagement with an activity; reaching these goals can provide a sense of satisfaction and reward, which in turn reinforces the desire to experience elements with increasing difficulty. Similarly, game-like elements can help balance challenge and difficulty [...] to provide a sustained sense of progress. It also can give rise to the possibility of flow.

3) Games do not need to come at the expense of expressiveness. Game-like elements can be added while also leaving open space for expression. The most engaging games offer a balance of well-designed constraints and open creativity [...]. (Wang, 2016, S. 2)

Die fünf vorgestellten Anwendungen sind: *Magic Piano*, *Magic Fiddle*, *Magic Guitar*, *Ocarina & Ocarina 2* und *Leaf Trombone: World Stage* - Alle wurden für den Massenmarkt entwickelt „[...] intended to lower the barrier of entry for novices, non-musicians, and casual players, but have high skill ceilings to leverage more seasoned skill and musicianship.“ (Wang, 2016, S. 2)

Der grundlegende Designansatz ist bei sämtlichen Anwendungen gleich, jedoch unterscheiden sie sich in der Spiel-Mechanik. Während bei *Magic Piano* vor allem das Timing des Finger-Tippens im Vordergrund steht, wird *Magic Fiddle* (ähnlich einer echten Geige) beidhändig zu einer Begleitmusik „gespielt“ bzw. gestrichen. *Ocarina* bedient sich schließlich dem Mikrofon-Sensor eines Smartphones um durch den Luftstrom der menschlichen Atemluft sowie einem bestimmten „Fingersatz“ Töne zu erzeugen. (vgl. Wang, 2016, S. 2-4)

3.3 Serious Games

Serious Games werden speziell dafür entwickelt, Lernprozesse verschiedenster Art voranzutreiben, beinhalten also eine starke pädagogische Komponente. „Bei unterhaltungsorientierten Computerspielen ist das *Gameplay* und das Ziel des Spiels in eine fiktive Geschichte und eine entworfene Problemstellung eingebunden, bei *Serious Games* sollte das Design einem realen Problem und dessen Lösungsfindung unterworfen sein.“ (Halbeisen, 2011, S. 44) - was aber

nicht bedeutet, dass Videospiele nicht auch für bestimmte Zwecke eingesetzt werden können.

Die Gestaltung von Serious Games schließt einen gewissen Unterhaltungswert für die Spieler keinesfalls aus, doch sollte ein Gleichgewicht zum jeweiligen Lernziel bestehen. Halbeisen (2011) formuliert in diesem Zusammenhang:

Bei zweckorientierten Spielen werden [...] Engagement und Lernen gleich hoch bewertet, was jedoch zur Voraussetzung hat, dass sich diese beiden Komponenten während des ganzen Spielverlaufs in einer Balance befinden. Vor allem aufgrund der spielinhärenten Motivation und des immersiven Charakters lernen die Spielende [sic!] informell. (2011, S. 46)

Hoffmann berichtet in ihrem Artikel „Learning Through Games“ (2009) über einige wissenschaftliche Institutionen, die Spiele zum Zweck des Lernens erforschen: „Over the past few years [...] a new body of research has begun to demonstrate how games can have a positive effect on youngsters by stimulating their imaginations, sparking their curiosity, and promoting the exploration of difficult issues and concepts.“ (S. 21)

Am New Yorker *Games for Learning Institute* werden in erster Linie dynamische Prozesse zwischen Spielern und Spielen untersucht, um die Gestaltung von Lernspielen zu optimieren: „The ultimate goal [...] is to develop a comprehensive set of principles and standards that could help people effectively design, build, and use educational games.“ (Hoffmann, 2009, S. 22)

Ebenso wie bei der Erforschung von Gamification und der theoriegeleiteten Betrachtung von Musik und Klang in Videospiele wird hier also versucht, die Wechselwirkungen zwischen den Anwendungen und den Benutzern zu analysieren. Daraus sollen Modelle abgeleitet sowie Elemente isoliert werden, um letztlich Designern und Entwicklern zu helfen, effektiver und besser arbeiten zu können.

4. Musikspiele

Die Geschichte der musikalischen Spiele lässt sich zumindest bis zu Lebzeiten Mozarts zurückverfolgen. Um den Rahmen dieser Arbeit aber nicht zu sprengen, widmet sich Abschnitt 4.1 der Entwicklung dieser Spiele, wobei der Fokus auf elektronische- und vor allem Computerspiele mit Musik gerichtet wird.

Wie sich im Verlauf des Kapitels zeigen wird, fällt eine strikte Unterteilung von Musikspielen in bestimmte Gruppen nicht immer leicht. Abschnitt 4.2 gibt einen Einblick in die derzeit vorherrschenden Meinungen zu diesem Thema und versucht, eine Kategorisierung zu liefern.

Abschnitt 4.3 nennt konkrete Beispiele zu den einzelnen Kategorien von Musikspielen und beleuchtet Spezialfälle sowie deren besondere Eigenschaften.

Der Großteil dieses Kapitels berichtet von musikalischen Spielen, die zu kommerziellen Zwecken entwickelt wurden. In Abschnitt 4.4 wird schließlich auf Anwendungen eingegangen, die in erster Linie der Erforschung und theoriegeleiteten Auseinandersetzung mit Sonic Interaction Design dienen. Außerdem geht es darin um spielerische Zugänge zum Thema Musik und Lernen.

4.1 Geschichte und Entwicklung

Schon lange bevor elektronische Spielzeuge und Computerspiele Einzug in die moderne Gesellschaft hielten, verbreiteten sich (u.a. entwickelt von Mozart) ab Mitte des 18. Jahrhunderts „musikalische Würfelspiele“ für Klavier, die Amateure in ganz Europa abonnieren konnten. (vgl. Noguchi, 1996) Dabei war es möglich, mithilfe von Würfeln und Tabellen, eigene Kompositionen anzufertigen und auszuprobieren: „All the publishers stated that an infinite number of compositions could be written by any amateur, even if he were not familiar with the techniques or rules of composing.“ (Noguchi, 1996)

Frühe elektronische Spielzeuge und Videospiele, die Klang und Musik in den Fokus der Interaktion zwischen Benutzer und Anwendung rückten, erschienen Mitte der 70er Jahre des vorigen Jahrhunderts. Eines der bekanntesten

Beispiele aus dieser Periode ist das Spiel *Simon* (1978), entwickelt von Ralph Baer. (vgl. Pichlmair et al., 2007, S. 3) Das Prinzip hinter *Simon* ist so simpel, wie genial:

The game has four buttons and four lights. Each light and button has an associated sound. The game flashes the light (and plays the sound) in a specific pattern. The player has to repeat the pattern, tapping the buttons in rhythm. (Pichlmair et al., 2007, S. 4)

Ungefähr zehn Jahre später, als bereits die ersten 16-bit Spielkonsolen im Umlauf waren, erschien *Michael Jackson's Moonwalker* (Sega, 1989) - und obwohl es nur mäßigen kommerziellen Erfolg brachte, markiert es einen Meilenstein in der Geschichte der Musik-Videospiele, als „[...] the first major collaboration of a pop artist with a video game's soundtrack.“ (Pidkameny, 2002) Wie auch Holly Tessler in ihrem Artikel „The new MTV? Electronic Arts and ‚playing‘ music“ (2008) anmerkt, sollten noch viele solcher Kollaborationen folgen. (vgl. S. 15f)

Allmählich wurden auch Musiker aus alternativen Genres und Medienkünstler auf das Phänomen Videospiele aufmerksam und begannen, mit synästhetischen Effekten und dynamischen Klängen zu experimentieren. In diesem Zusammenhang ist das 1987 erschienene Spiel *Otocky* von Toshio Iwai zu nennen, bei dem der Spieler einen Roboter durch unterschiedliche Levels steuern muss. Hübner (2011) beschreibt das innovative Gameplay folgendermaßen: „Ungewöhnlich ist, dass der Spieler die Hintergrundmusik durch die Bewegungen seiner Spielfigur erzeugt. Je nachdem, in welche Richtung die Spielfigur schießt, wird eine andere Note abgespielt. Dank einer ausgeklügelten Programmierung ergeben diese Noten stets eine harmonische Melodie.“ (S. 17-18) Dieses interaktive „Musizieren“ in *Otocky* besticht also durch Musik, die bereits in hohem Maße dynamisch und zugleich adaptiv ist.

Die 1990er markieren jene Dekade, in der musikbezogene Videospiele allmählich stark an Popularität gewinnen sollten. Mit dem 1996 in Japan erstmals erschienenen Spiel *PaRappa the Rapper* (Sony Playstation) wurde gewissermaßen das Genre der Rhythmus-Videospiele begründet, bei denen der Spieler Rhythmusgefühl und Timing einsetzen muss, um die Handlung voran zu treiben - oder schlicht, um Punkte zu generieren. Muss man bei

PaRappa the Rapper „lediglich“ die Tasten auf dem Controller in der richtigen Reihenfolge und im jeweiligen Takt der Musik drücken, verlangen Titel wie *Dance Dance Revolution* (Konami, seit 1998) oder *Guitar Hero* (Activision, seit 2005) ganzen Körpereinsatz - ein Umstand, der die Immersion entschieden erhöht.

Eine weitere Innovation boten Spiele, bei denen der Benutzer tatsächlich (oder zumindest innerhalb der virtuellen Welt) selbst musiziert. Zu erwähnen ist in diesem Zusammenhang *The Legend of Zelda: Ocarina of Time*¹⁷ (Nintendo 64, 1998) - im Verlauf des Spiels müssen verschiedene Tonfolgen erlernt werden, die bestimmte Bedeutungen haben. Aktiviert man die Okarina, welche der Spielcharakter *Link* bei sich trägt, können diese Melodien anhand der Tasteneingabe am Controller wiedergegeben werden - dabei entspricht jeder Taste ein bestimmter Ton. (vgl. Scherer 2015, S. 26) Theoretisch kann das virtuelle Instrument also jederzeit (sofern *Link* es dabei hat) aktiviert werden, um darauf zu spielen - doch nur an gewissen Stellen, zu bestimmten Zeiten, oder unter den richtigen Umständen bewirken die Melodien auch etwas innerhalb des Spiels.

Die 2000er Jahre waren u.a. davon geprägt, neue tragbare Spielkonsolen hervorzubringen, die zum Teil schon mit berührungsempfindlichen Bildschirmen ausgestattet waren. Dieser Umstand bot den Entwicklern weitere Möglichkeiten für die Gestaltung von Musikspielen und es entstanden mehr und mehr Spiele, die man vielleicht „virtuelle Instrumente“, oder „interaktive Klanginstallationen“ nennen könnte. Dazu zählt auch *Electroplankton* (2005) für die portable Nintendo DS Konsole, das durch die Berührungen des Bildschirms mittels eines Eingabestifts gesteuert wird:

Es besteht aus zehn verschiedenen Musikspielzeugen, mit deren Hilfe sich unterschiedlich komplexe Klangwelten erschaffen lassen. [...] jede der zehn comicartigen Experimentierumgebungen im Spiel ist nach einer fiktiven Planktonart benannt. Durch Bewegungen auf dem Touchscreen kann der Spieler die dargestellten Figuren manipulieren und dadurch

¹⁷ Vgl. dazu auch die Masterarbeit von Plaschg (2016) mit dem Titel: Kognitives Training durch Computerspielen - am Beispiel „The Legend of Zelda: Ocarina of Time“.

Klänge erzeugen, die sich alle in die Kategorie Ambient-Musik einordnen lassen. (Hübner, 2011, S. 18-19)

Auch Karaoke-Spiele wurden zu Beginn des neuen Millenniums enorm populär und förderten die Expressivität und den Körpereinsatz der Spieler noch weiter - allerdings nur in der „echten Welt“ (vgl. Pichlmair et al., S. 5) Bei diesen Spielen, die meist auch gemeinsam im Duett (oder aber gegeneinander) zu bestreiten sind, müssen die Texte zu Popsongs über ein Mikrofon in der richtigen Tonhöhe und dem entsprechenden Rhythmus eingesungen werden. Die bekanntesten Titel darunter sind *Karaoke Revolution* (Konami, seit 2003) und *SingStar* (Sony, seit 2004).

Das Spiel *Vib Ribbon* (Sony, 2000) für Playstation 2 mit minimalistischer Vektorgrafik, brachte eine weitere Innovation, wie Collins (2008b) berichtet:

[it] allowed the user to put in his or her own music CDs, which would then influence the generation of maps in the game. The game scanned the user's CD and made two obstacle courses for each song (one easy and one difficult), so the game was always as varied as the music the player chose. (S. 5-6)

Mit dem Erscheinen der Software *Korg DS-10* (Nintendo DS, 2005) konnte der Spieler schließlich wirklich zum Musiker bzw. Produzenten werden, und das auf einer tragbaren Spielkonsole in Hosentaschenformat. Hübner (2011) bemerkt dazu treffend: „Wie der Name schon vermuten lässt, handelt es sich hierbei um die detailgetreue Simulation eines sechsspürigen 16-Step-Sequenzers. Traditionelle Merkmale eines Videospieles [...] lassen sich nicht finden, obwohl das Nintendo DS als reine Spielkonsole vermarktet wird.“ (S. 10)

Aktuelle Trends in der Entwicklung und Gestaltung von musik- und klangbezogenen Videospiele spiegeln sich vor allem in der Nutzung von neuartiger Sensortechnik wider, die viele Smartphones und Tablets eingebaut haben. Die bereits in Abschnitt 3.2 dieser Arbeit beschriebenen Anwendungen von *Smule* liefern Beispiele dafür.

4.2 Kategorisierung

Die systematische Einteilung der Musikspiele gestaltet sich mitunter schwierig, da die Grenzen zwischen den Kategorien und Subkategorien zum Teil verschwimmen. Außerdem passen manche Spiele in mehrere Kategorien gleichzeitig.

Pichlmair und Kayali (2007) haben im Zuge einer Studie mehrere Musikspiele untersucht und schlagen eine grundsätzliche Unterteilung in „rhythm games“ und „electronic instrument games“ vor. (vgl. S. 1) Die Hauptmerkmale setzen sich wie folgt zusammen:

[...] a rhythm game is primarily played by comprehending the rhythm the game sets. The player has to smash buttons in a rhythm provided by the game. As progress is made the game successively increases the speed and complexity of the rhythms. (S. 1-2)

Electronic instrument games are different in all regards. The rhythm - as well as in most cases the melody - is generated by the player. She plays the game as an instrument. The game provides - or at least pretends to provide - all the freedom of expression that a musical instrument calls for. (S. 2)

Collins (2008c) schlägt eine ähnliche Kategorisierung vor, indem sie formuliert: „Games in which music is the primary driving motive or narrative element can be roughly divided into three categories: musician-themed games, creative games, and rhythm-action games.“ (S. 36) Dabei lassen sich „rhythm-action games“ mit Pichlmairs „rhythm games“ und „creative games“ mit der Kategorie „electronic instrument games“ gleichsetzen. Mit „musician-themed games“ meint Collins Spiele, in denen Musiker selbst als Charaktere vorkommen und/oder in irgendeiner Form in die Handlung eingebunden sind. (vgl. 2008c, S. 37)

Pichlmair (2007) beschreibt, neben den beiden genannten Hauptkategorien, eine weitere Gruppe von Spielen, nämlich: „Musical puzzles or challenges in games“ (S. 2) Diese musikalischen Rätsel als eigene „Gruppe“ innerhalb der Musikspiele zu deklarieren, würde allerdings zu weit führen, da sie in verschiedensten Spielgenres vorkommen können.

Ähnlich wie Collins, schlägt auch Hübner (2011) eine Unterteilung in drei Hauptkategorien vor und nennt diese „(interaktive) Soundtrack-Spiele“, „Rhythmus-Reaktionsspiele“ und „Musikproduzierende Spiele“. (vgl. S. 8-10) Unter „Soundtrack-Spiele“ fallen Spiele „[...] in denen die Musik von außerhalb der Spielindustrie tätigen Künstlern vorkommt, diese jedoch lediglich zur Untermalung des Spielgeschehens oder dem Aufbau einer besonderen Stimmung dient.“ (S. 8) Als Beispiel dafür werden Titel wie *The Eye* (IBM PC, 1998) oder *Crazy Taxi* (Sega Dreamcast, 1999) genannt, die mit Musik der Bands Queen bzw. Bad Religion untermalt sind. Hat man als Spieler zusätzlich die Möglichkeit, mit der Musik zu interagieren - bei *Xplora1* (Macintosh/PC 1993) kann beispielsweise anhand eines Mischpults ein Mix eines Peter Gabriel Songs angefertigt werden - spricht man von einem „interaktiven Soundtrack-Spiel“. (vgl. Hübner 2011, S. 8)

Einen weiteren Bereich unter den Musikspielen, der zwar relativ klein, aber wegen seiner gestalterischen Herausforderungen interessant für Forschung und Entwicklung ist, bilden Audiospiele. Diese Spiele werden vor allem für blinde und sehbehinderte Menschen entwickelt, um ihnen auch die Möglichkeit zu geben, Computerspiele zu erleben. Gärdenfors (2013) bemerkt zu diesem Thema treffend:

The new possibilities to use sound in interactive media are very welcome to computer users that have difficulties in using graphical displays. [...] Mainstream computer games are generally graphics-oriented, with sounds added mainly as decorative effects. Therefore, they are most often inaccessible to visually impaired computer users. (S. 111)

Rovithis et al. (2014) unterscheiden bei Audiospielen zwischen „audio-based games“ und „audio-only games“. (vgl. S. 21) Die Kategorie setzt sich also aus Spielen zusammen, die keinerlei visuelle Stimuli beinhalten und Spielen „[...] in which some visual elements are used, but only in a supportive way to facilitate the better perception of aural stimuli.“ (Rovithis et al., 2014, S. 21)

Gärdenfors (2003) fasst Audiospiele in seinem Artikel „Designing Sound-Based Computer Games“ unter der Bezeichnung „sound-based games“ bzw. „sound-based interactive entertainment for visually impaired users“ zu einer gesamten Gruppe zusammen. (vgl. S. 111-112)

Bei Parker & Heerema (2007) findet sich eine Übersicht zu Audiospielen (kategorisiert als „audio games“), die aufzeigt, wie vielseitig Computerspiele sein können, die ausschließlich mit Klang arbeiten. (vgl. S. 3) Interessanterweise scheint dort - neben Audio-Rennspielen oder Audio-Shootern¹⁸ - auch die bereits kurz erwähnte Spielreihe *Karaoke Revolution* auf. Hübner (2011) ordnet diese Singspiele in die Kategorie der Rhythmus-Reaktionsspiele ein, da es darum geht: „[...] dass der Spieler im richtigen Moment eine bestimmte Aktion vollzieht: in diesem Fall das Singen einer Textzeile.“ (S. 9) Außerdem wird bei Karaoke-Spielen normalerweise der Text des jeweiligen Liedes am Bildschirm angezeigt. Parker ist sich dessen grundsätzlich bewusst, bemerkt allerdings treffend: „[...] the display is really not needed, since most folks know the words for the song they are going to sing.“ (2008, S. 3)

Wie dieses Beispiel deutlich zeigt, kann es bei gewissen Spielen zu Überschneidungen bezüglich der Kategorisierung kommen. Außerdem tauchen in der weitreichenden Landschaft der Musik- und Audiospiele immer wieder Sonderfälle auf, für die eine eigene Bezeichnung erst erfunden werden müsste.

4.3 Beispiele, Spezialfälle und besondere Eigenschaften

4.3.1 Rhythmus-Reaktionsspiele / Rhythm-Games

Zu den prominentesten Vertretern dieser Kategorie zählen Spielreihen wie *Dance Dance Revolution*, *Guitar Hero*, oder *SingStar* (Sony, seit 2004). Diese Spiele können als Musiksimulationen bezeichnet werden und werden oftmals mit speziellen Eingabegeräten und Controllern ausgeliefert. So gibt es u.a. einen Turntable-Controller für DJ-Simulationen, wie *DJ Hero* (Activision, 2009). (vgl. Hübner, 2011, S. 9) Musiksimulationen beinhalten immer auch eine gewisse soziale Komponente, da man z.B. im Duett singen, oder im Tanz gegeneinander antreten kann. Das Spiel *Donkey Konga* (Nintendo, seit 2003) beinhaltet einen Bongo-Controller, der auch ein Mikrofon besitzt (um Klatschgeräusche zu interpretieren) - und lässt bis zu vier Spieler gleichzeitig am Spielgeschehen teilnehmen. (vgl. Marschner, 2008, S. 20)

¹⁸ Shooter sind im Grunde Videospiele in denen es gilt, eine größtmögliche Anzahl an Gegnern abzuschießen.

4.3.2 Klangproduzierende Spiele / kreative Musikspiele

Neben der kurz erwähnten Anwendung *Korg DS-10*, beschreibt Collins (2008c) weitere Spiele, die es ermöglichen, selbst Musik zu machen und mit Klang zu experimentieren. (vgl. S. 38) Als frühes Beispiel dazu, nennt die Forscherin die von MTV vermarktete *Music Generator* - Serie (Codemasters, 2000):

[it] began as what was essentially sequencing software with sequences provided and became a remixing project of various popular music songs. Users are given an editing bay of riffs, beats, samples, bass lines, and vocal sequences and can assemble music from a variety of genres. In this case, players are given the opportunity to interact with and participate in various popular songs. (2008c, S. 38)

Obwohl kein Videospiele, kann das von Toshio Iwai entwickelte *Tenori-On* (Yamaha) trotzdem unter die Kategorie „Electronic Instrument Game“ eingeordnet werden. (vgl. Pichlmair et al., 2007, S. 2) Hübner beschreibt das Gerät folgendermaßen:

Das äußere Erscheinungsbild [...] wird bestimmt von 256 quadratisch angeordneten Leuchtdioden, die nicht nur die Musik visualisieren, sondern auch als Steuerelemente dienen. Obwohl das Instrument viele Funktionen bietet, die sich am ehesten mit denen eines Sequenzers vergleichen lassen, können auch Laien damit Musik erzeugen. [...] Trotz des einfach gehaltenen Bedienungskonzepts können mit dem Tenori-On komplexe Klangkompositionen erschaffen werden, denn es lassen sich [...] unterschiedliche Ebenen auf dem Gerät öffnen, die simultan abgespielt werden können. (2011, S. 19-20)

Pichlmair (2007) weist darauf hin, dass das grundlegende Prinzip hinter dem *Tenori-On*, nämlich Musik auf spielerische Weise zu erzeugen, auch bei Toshio Iwais Videospiele *Sim Tunes* (Maxis, 1996) zu finden ist:

In *Sim Tunes*, the player - or composer, or musician - lays out a score on a two-dimensional surface. Then she pours so-called musical insects on the surface and these are playing the score. The insects wander across the

surface and once they hit a play element, they act accordingly by altering their movement and/or emitting sound. (S. 2)

Das Spiel ragt also nicht nur heraus, durch die spezielle Art mit Musik zu experimentieren, es bietet dem Anwender auch Platz, seine grafische Kreativität auszuleben.

4.3.3 Musician-Themed Games

Einige Videospiele, in denen reale Musiker oder Bands als Charaktere (und meist auch ihre Songs) vorkommen, erscheinen besonders bizarr und absurd. Gerade ältere Spieltitel legen dabei die Vermutung nahe, dass die Spielfirmen sich allzu oft auf die Kaufkraft der Fans verließen, ohne dabei innovative Gameplay-Ideen, Handlungen, oder Spielmechaniken zu liefern - noch dazu wurde in vielen Fällen kaum Wert auf die Musik selbst gelegt, wie Collins (2008c) über das Spiel *Revolution X* (Midway/Acclaim, 1994) mit der Band Aerosmith berichtet:

The game's plot is that the world has been overtaken by a dominatrix, Mistress Helga, and her followers, the New Order Nation, who kidnap the band. The player must throw CDs at the opponents and rescue the band members. [...] The home game featured five Aerosmith songs [...] three of the songs dated back two decades before the game's release. Complete with dancing girls in cages, the game featured short, distorted and scratchy sounding looped four-track instrumental riffs of about fifteen seconds in duration that repeated over and over. The lack of attention to the quality of the music suggests that the game is more about capitalizing on the persona of the band and its fans, than the actual music. (S. 37)

Zugegeben, die Wechselwirkung zwischen Musikspielen und Popmusik ist nicht zu unterschätzen, doch gibt es auch Fälle, in denen Musiker selbst zu Spielentwicklern werden - und dabei sieht das Ergebnis meist sehr viel anders aus. Das von Peter Gabriel veröffentlichte Projekt *Xplora1* wurde bereits erwähnt und von Hübner (2011) als „interaktives Soundtrack-Spiel“ klassifiziert. In eine ähnliche Richtung gehen die Spiele *Tres Lunas* (2002) und *Maestro* (2004) von Mike Oldfield, wie Hübner formuliert:

Bei beiden Programmen handelt es sich um die Simulation einer virtuellen - optisch an die Gemälde Salvador Dalis angelehnten - Welt, in der der Spieler Soundschnipsel von Mike Oldfield hören und Objekte manipulieren sowie Rätsel lösen kann. Die Spielmechanik ähnelt der eines First-Person-Shooters; im Gegensatz zu diesen fast immer brutalen Spielen verströmen *Tres Lunas* und *Maestro* jedoch eine traumhafte und entspannte New Age-Atmosphäre. [...] Insgesamt betrachtet sind [beide Titel] eher Kunstinstallationen als Computerspiele. (S. 6-7)

4.3.4 Audiospiele / klangbasierte Spiele

Eines der ersten Audiospiele war vermutlich *Bear Hunt*, bei dem der Spieler - bewaffnet mit einer Spielzeugpistole und vor den Lautsprechern stehend - Jagd auf Bären machen sollte. Dabei musste man auf die Geräusche im Stereofeld achten, um zu erkennen, aus welcher Richtung sich das Ziel näherte, um schließlich im richtigen Moment abzurücken. (vgl. Parker et al., 2008, S. 3)

Weitaus weniger brutal, dafür umso innovativer, war das Konzept, ein interaktives Hörspiel für sehbehinderte Kinder zu entwickeln. Im Spiel *Real Sound-Kazeno Regret* (Sega, 1999), erschienen für die Konsolen *Dreamcast* und *Saturn*, können die Benutzer ihr eigenes Audio-Abenteuer erleben. (vgl. Collins et al., 2010, S. 253)

Ein interessantes und noch dazu ziemlich populäres Beispiel eines klangbasierten Spiels ist *Papa Sangre* (2010), erschienen für iOS. (vgl. Collins, 2013, S. 24) Dabei muss der Spieler mithilfe eines grafisch sehr einfach gehaltenen Interfaces aus verschiedenen Verliesen entkommen, indem er genau auf die jeweilige akustische Umgebung achtet. Collins (2013) beschreibt die Stimmung und das Gameplay des Spiels:

[...] each castle has its own sonic identity, and different forms of monsters are represented through sound effects. The main goal of the game is survival as the player wanders through the castles collecting musical keys that commonly hide behind one of the many roaming monsters. Monsters respond to player-generated sound, so if players move too quickly or step on an object that makes sound, they will be chased and killed. (S. 24-25)

Die Spannung bei solchen Spielen kann durch die Abwesenheit von grafischen Anhaltspunkten enorm gesteigert werden, da sich der Spieler die Umgebung nur vorstellen kann und die Klänge gewissermaßen von den Objekten in der virtuellen Welt entkoppelt sind: „This mental imagery is a result of our synesthetic experience of sound as a component of a multisensory integration [...]“ (Collins, 2013, S. 26)

Ein weiteres Beispiel, bei dem das Phänomen der Synästhesie eine entscheidende Rolle spielt, ist der Titel *Rez* (Sega, 2001). Pichlmair et al. (2007) bezeichnen es als „action music game“, dessen Spielprinzip als umgekehrte Form von Rhythmus-Reaktionsspielen gedeutet werden kann. (vgl. S. 4) Das Gameplay von *Rez* gleicht dem eines klassischen Shooters, doch hierbei löst jeder Schuss Klänge aus, die sich dem Rhythmus und der Musik im Spiel anpassen:

The challenge is not to hit the beat but to create a steady flow of sound. [...] *Rez* provides an immersive environment by tightly coupling visual and acoustic sensations. Every shot is a particle of sound; every beat is felt through the vibration of the controller, every enemy encountered is a reason to pump a wave of techno music into the air [...] (Pichlmai et al., 2007, S. 4)

Aufgrund der Verbindung von Gameplay-Elementen klassischer Spielgenres und der adaptiven Gestaltung von Klang, welcher in hochdynamischer Musik resultiert, können Spiele wie *Rez* auch als „hybrid music games“ bezeichnet werden. (vgl. Bernardo, 2014, S. 2)

Als letzter Spezialfall von Computerspielen, die Musik und Klänge als einzigartiges Gestaltungsmittel verwenden, soll das 1990 erschienene Abenteuerspiel *Loom* (Lucasfilm Games) erwähnt werden. Obwohl es normalerweise (und auch zurecht) als klassisches Grafik-Adventure im Point&Click-Stil bezeichnet wird, setzte *Loom* neue Maßstäbe, wie Pidkameny berichtet: „[it] stands out as the first computer game to feature music with varied tempo and dynamics, as well as recurring themes and character leitmotifs; in some ways, *Loom* represents classical music's first real breakthrough into the world of computer games.“ (2002)

Das Spiel beinhaltet u.a. Ausschnitte aus Tschaikowskis „Schwanensee“, die in das MIDI-Format transponiert wurden, um zusätzlich zum Hauptsoundtrack (komponiert von George Sanger) eingebunden zu werden. (vgl. Pidkameny, 2002) Auch die Steuerung von *Loom* war besonders innovativ, da dem Spieler ein Instrument als Interface dient: „In order to interact with the game world, the player had to play tunes that magically affected the universe.“ (Pichlmair et al., 2007, S. 4) Ähnlich wie bei *The Legend of Zelda: Ocarina of Time* lernt man während des Spielverlaufs unterschiedliche Melodien, die man später benötigt, um Rätsel zu lösen, oder Gegner zu bezwingen.

4.4 Musikspiele als Forschungsgegenstand

Es sollen nun musikalische Anwendungen vorgestellt werden, die sich der wissenschaftlichen Untersuchung von Sonic Interaction Design sowie verschiedenen theoretischen und praktischen Implikationen von Klang als Gestaltungsmittel widmen. Um die Relevanz dieser Themen für die vorliegende Arbeit hervorzuheben und allmählich auf den praktischen Teil überzuleiten, wird ein besonderes Augenmerk auf Anwendungen für mobile Endgeräte gelegt und der spielerische Zugang zum Experimentieren mit Musik unterstrichen.

Gillian et al. (2009) weisen im Artikel zu ihrem Projekt „Scratch-Off: A gesture based mobile music game with tactile feedback“ darauf hin, dass die mittlerweile weit verbreiteten Interaktions-Gesten (z.B. Wischen, Schütteln, oder Tippen) bei der Nutzung von Smartphones interessante Möglichkeiten zur Spielgestaltung bieten. (vgl. S. 308) Die Forscher entwickelten die Anwendung „Scratch-Off“, ein DJ-Spiel für Smartphones, das durch unterschiedliche Gesten gesteuert werden muss. Akustisches, sowie haptisches Feedback werden genutzt, um dem Anwender Hinweise auf den Verlauf der Musik und die auszuführenden Bewegungen zu geben. Dabei bedienen sich die Forscher beim klassischen Paradigma des „Scratchens“ eines Discjockeys:

[...] one hand controls the playback speed of the record by pushing and dragging it with fast and short movements, whilst the other hand is used to control the sound level that is sent to the speakers. This is reflected in Scratch-Off by the two different methods of interacting with the game, using the flick or pinch gestures.

Nach der Entwicklungsphase wurde in einer Studie untersucht, welche Formen des multimodalen Feedbacks den Benutzern am meisten zusagten. Obwohl nicht signifikant, zeigte sich, dass die Kombination aus akustischem und haptischem Feedback besser empfunden wurde, als die Kombination aus akustischem und visuellem Feedback. (vgl. Gillian et al., 2009, S. 310-311)

Die Anwendung „TapBeats“ wurde 2011 von Joy Kim und Jonathan Ricaurte an der Universität von Washington entwickelt. Dabei handelt es sich um ein musikalisches Gelegenheitsspiel für das mobile Betriebssystem Android, das auch blinde Menschen spielen können. (vgl. S. 285) TapBeats ist ein gutes Beispiel für die Anwendung von „user centered design“ - einem Verfahren, bei dem der Benutzer von Anfang an in die Entwicklung mit einbezogen wird.

Das Spiel bietet vier Spielmodi zur Auswahl, die im Menü gewählt werden können, nämlich: Memory Mode, Concert Mode, Studio Mode und Career Mode. (vgl. Kim et al., 2011, S. 286) Die Forscher erklären, wie blinde oder sehbehinderte Menschen im Spielmenü navigieren können:

As the user's finger touches a menu item, the phone will speak out the menu item name and generate a small vibration. Once the user hears the desired item, they can select it by double tapping anywhere on the screen. To return to a previous menu screen, the user can swipe with two fingers anywhere on the screen. The user can also use this gesture during gameplay to pause. Finally, if the user wants the game to speak out what menu screen they are currently on, they can double tap with two fingers. (2011, S. 285)

Der Hauptbildschirm während des Spiels ist in vier rechteckige Felder unterteilt, die je einem Schlagzeug-Sample entsprechen: Kickdrum, Snare, Cymbal und Hi-hat. Je nach Spielmodus, müssen diese virtuellen Knöpfe beispielsweise im richtigen Rhythmus zu einem laufenden Song gedrückt werden. In einem anderen Modus müssen kurze Sequenzen nachgespielt werden, die exaktes Timing voraussetzen. Nach Rücksprache mit einem blinden Kollegen wurde entschieden, zusätzlich einen „Free-Play-Modus“ - zum Ausprobieren der Funktionen - einzuführen. (vgl. Kim et al., 2011, S. 286)

Wie Liu et al. (2014) in ihrem Artikel zu „Chorlody: A Music Learning Game“ anmerken, kann ein spielerische Zugang zu Musik die Benutzer motivieren, selbst zu musizieren, oder ihr Wissen zu musikbezogenen Themen zu erweitern:

Giving power to players is one important design principle in combining learning materials and video games. In music games, players are not limited by instrumental techniques or knowledge in music theory. Therefore they would be motivated to learn music or encouraged to start playing an instrument. (S. 278)

Chorlody ist eine Anwendung im Stil eines Rhythmus-Reaktions-Spiels, die dem Benutzer helfen soll, Akkord-Fortschreitungen zu erkennen, diese zu verstehen und zu erlernen. Dabei symbolisiert ein Ring, bestehend aus zwölf bunten Tasten, die chromatische Tonleiter und durch Dreiecke innerhalb des Rings werden Dreiklänge visualisiert. (vgl. Liu et al., S. 279) Bevor ein Spiel gestartet wird, kann das Interface ausprobiert werden. Die zwölf Tasten sind entsprechend ihrer Töne beschriftet und drückt man eine davon, erklingt ein Dreiklang mit dem gedrückten Ton als Grundton. Zusätzlich wird in der Mitte des Dreieck-Symbols der Name des Akkords angezeigt.

Während des Spiels müssen schließlich die richtigen Akkorde im entsprechenden Rhythmus eines im Hintergrund laufenden Songs gedrückt werden, um Punkte zu erhalten - präzises Timing und die richtige harmonische Fortschreitung werden dabei mit Extrapunkten belohnt. (vgl. Liu et al., S. 279)

5. Sampling

Der nun folgende Abschnitt (5.1) zeigt einen kurzen geschichtlichen Abriss des Phänomens Sampling sowie die technischen Aspekte dazu und geht auf die kulturelle Bedeutung des Themas ein.

Um die Relevanz von Sampling für die vorliegende Arbeit zu unterstreichen, gibt Abschnitt 5.2 Einblick in eine besondere Form der Live-Performance.

5.1 Geschichte, technische Implikationen und kulturelle Bedeutung

Die Geschichte des Sampling in der Musik steht in direktem Zusammenhang mit der Entwicklung von Aufnahmetechnologien für Klänge. Sampling manifestiert sich im Grunde in der Speicherung und anschließenden Wiedergabe von akustischen Ereignissen - wobei in einem Zwischenschritt das Klangmaterial oftmals in unterschiedlicher Form manipuliert wird. Das Prinzip hinter dem Sampling lässt jedenfalls aus Bestehendem etwas Neues entstehen, wie Fischer (2013) berichtet: „Sampling ist kurz gesprochen eine Methode, bei der aus vorhandenem Klangmaterial Ausschnitte („Samples“) extrahiert, diese nach eigenen Kriterien verändert und eventuell mit anderen Samples kombiniert werden, um auf diese Weise neue Musikstücke zu kreieren.“ (S. 3)

Im Verlauf des 20. Jahrhunderts sollte dieses Prinzip zahlreiche Ingenieure, Komponisten und Musiker zu innovativen Ideen inspirieren, wie man mit gespeicherten Klängen umgehen kann.¹⁹ Waren es anfangs noch eher technische Experimente, wie etwa in den Klangkompositionen von Walter Ruttmann (um 1930), beschäftigte sich bald die Avantgarde der elektronischen Musik mit den neuen Möglichkeiten. (vgl. Fischer, 2013, S. 80f) Hier waren es vor allem Vertreter der *Musique Concrète* (Pierre Schaeffer), der Neuen Kölner Schule (Karlheinz Stockhausen) und der *Tape Music* (Terry Riley) ab den frühen 1950ern, die die Grenzen des Sampling (das damals aber noch nicht „Sampling“ genannt wurde) als ästhetisches Gestaltungsmittel ausloteten.

Ab den 1970ern begann die eigentliche Ära des Sampling, wie wir es heute kennen und nahm bald darauf entscheidenden Einfluss auf die Welt der Popmusik: „Mit der Entstehung von Hip-Hop in den 1970er und 1980er Jahren

¹⁹ Eine detaillierte Übersicht zum Thema findet sich bei Fischer (2013).

und der anschließenden Entwicklung von Drum'n'Bass in den 1990er Jahren halten die Methoden der Wiederholung und Bearbeitung von klanglichen Fragmenten in die populäre Musik Einzug.“ (Fischer, online)

Das Aufkommen der Dj-Kultur im Allgemeinen und der Hip-Hop-Kultur im Speziellen, machte das Sampling gewissermaßen salonfähig. Anfangs wurden dabei Passagen aus alten Funk- und Soul-Platten mithilfe von zwei oder mehreren Plattenspielern live aneinandergereiht, um den Fluss der Musik aufrecht und das Publikum auf der Tanzfläche zu halten. Bald schon sollte sich die Arbeit des Discjockeys aber mehr und mehr ins Studio verlagern, als nämlich die ersten digitalen Hardware-Sampler für die Masse erschwinglich wurden: „[...] the story of sampling is a tale of technology catching up with the DJ, of equipment being created that could do faster, more accurately and more easily what a DJ had long been able to.“ (Brewster et al., 1999, S. 245)

Die Technologie wurde ab Mitte der 80er ökonomisch verfügbar, wie Rodgers (2003) berichtet: „The first sampler at a price point affordable to a broad market was the Ensoniq Mirage, introduced in 1984 at retail price of under \$2,000 [...]“ (S. 313) Poschardt erklärt die Funktion und technischen Aspekte solcher Geräte in einfachen Worten: „Der Sampler besteht aus einem Analog/Digital-Konverter, einem Speicher und einem Digital/Analog-Konverter, der aus der digitalisierten Form der Geräusche wieder hörbare macht. Dadurch, daß die Schallereignisse in digitale Form verwandelbar sind, lassen sie sich manipulieren und bearbeiten.“ (1995, S. 229)

Mithilfe der Sampler konnten fortan Klangfragmente aus älteren Musikstücken entnommen, weiterverarbeitet und in neue Kontexte eingebettet werden. Dabei wurden beispielsweise die Abspielgeschwindigkeit (und somit die Tonhöhe) oder andere Parameter verändert, um das Material zu verfremden und neu zu interpretieren. Es wurden verschiedene Ausschnitte aus Schlagzeugpassagen entnommen, zeitlich neu arrangiert und schließlich geloopt, um alternative Rhythmen zu erzeugen. Zudem wurden mehrere Schichten von Klängen übereinander gelegt und gleichzeitig abgespielt. Konkrete Klänge aus dem Alltag oder Passagen aus Filmen wurden auf kreative Weise in Musikstücke eingebaut und gaben der Musik mitunter ganz neue Bedeutungen.

Als schließlich leistungsfähige Software-Sampler für den Gebrauch am Heimcomputer entwickelt wurden, waren dem Manipulieren, Re-interpretieren und Umstrukturieren von Klangmaterial praktisch keine Grenzen mehr gesetzt. Bedeutend ist diese Entwicklung u.a. für die Stilrichtungen Techno, Jungle und Drum'n'Bass.

Gerade bei Jungle und Drum'n'Bass - sehr schnellen und rhythmus-betonten Formen von elektronischer Clubmusik - geht es also nicht darum, viele unterschiedliche Elemente in ein Musikstück zu packen, sondern die technischen Möglichkeiten des digitalen Samplings auszureizen. Dabei entstehen z.B. durch Dehnung und Stauchung der Samplelänge, Rückwärtsabspielen oder Schichtung einzelner Schläge zwar ähnliche, aber trotzdem individuelle Rhythmusmuster. (vgl. Fischer, 2013, S. 99)

Sampling wurde im frühen Hip-Hop mitunter sehr gezielt als Stilmittel eingesetzt. Gruppen wie *De La Soul* oder *Public Enemy* wählten besondere Samples aus, um damit bestimmte Identitätsmerkmale der Afro-Amerikanischen Kultur aufzugreifen und dadurch Botschaften zu transportieren. Durch die Verschärfung der Copyright Handhabungen wurde diese reiche Sampling Kultur allerdings ab ca. Mitte der 90er Jahre regelrecht vernichtet. (vgl. Fischer, 2013, S. 107)

Die unzähligen Anwendungsmöglichkeiten sowie der spielerische (zum Teil amateuristische) Umgang mit gespeichertem Audiomaterial, sollten nicht nur Klang und Ästhetik von Hip-Hop oder Drum'n'Bass beeinflussen. So kann der Künstler mitunter selbst - durch seine Performance, oder Produktion - zum Wissensvermittler werden und kulturelles (Klang-)Gut weitergeben (vgl. Jelinek 2002, S. 40), auch wenn dies vielleicht unbewusst geschieht:

Dieser [der samplende Produzent bzw. DJ] sucht geleitet von ästhetischen oder funktionalen Präferenzen und Motiven gewisse Teile aus dem kollektiven Archiv eines bekannten oder fremden Kulturkreises aus und assembliert diese für seine Zwecke. [...] Über Sampling wird tradiert, reinterpretiert, entschlüsselt, verschlüsselt, bewahrt und hinterfragt und die Anforderungen einer Kulturtechnik erfüllt. (Jelinek 2002, S. 41)

5.2 Sample-basierte Performance

Eine besondere Form des sample-basierten Musizierens findet sich in den Auftritten einiger Produzenten und DJs, die für diesen Zweck ein bestimmtes Instrument auserkoren haben: den *MPC* der Firma Akai²⁰. Die Abkürzung MPC stand früher für *Midi Production Center*, heute heißt das Gerät *Music Production Center* bzw. *Music Production Controller*. Die MPC-Serie, welche seit 1988 hergestellt wird, umfasst inzwischen zahlreiche Hardware-Varianten und ist seit einiger Zeit auch als Software-Applikation für mobile Endgeräte erhältlich.

Der MPC vereint im Grunde Sampler, Drumcomputer und Sequencer in einem Gerät und inkludiert 16 druckempfindliche Tasten, die mit den Fingern bedient werden. Vor allem diese Touchpads machen das Gerät zu einem viel verwendeten Live-Instrument, indem damit Samples via MIDI angesteuert und abgespielt werden. So verwenden manche DJs einen MPC um beispielsweise ihren Auftritt mit Soundeffekten zu untermalen, oder auch Bands gebrauchen ihn als Zusatzinstrument.

Künstlern wie *AraabMuzik* (USA), *Comfort Fit* (D) oder *Scarfinger* (F) allerdings dient der MPC als Hauptinstrument - manchmal in mehrfacher Ausführung - und essentielles Werkzeug für ausdrucksstarke Solo-Live-Auftritte:

It was a blistering performance. Araabmuzik took familiar beats [...] and dismantled them, then improvised off the themes, playing furious fusillades of sharp drums, shrieks and all sorts of sounds programmed into the machines. He played them like drums, with force. He played them like a piano, with dignity. And then, as if giving out at the end of a long sprint, he stopped, stepping away from the machines as the room caught its breath [...]. (Caramanica, 2011)

Dieser Eindruck des Journalisten Jon Caramanica aus der *New York Times Online* zeugt von einer energiegeladenen Performance, die das Resultat von viel Übung, wohl überlegter Vorbereitung und Beherrschung des verwendeten Instruments ist.

²⁰ Vgl. dazu: <http://www.akaipro.com/products/mpc-series>

Analysiert man die Videos²¹ von Performances von Künstlern wie *Ben Durazzo* (USA) oder *MorFiy* (RUS) und ist mit den grundlegenden Funktionen eines MPCs vertraut, wird deutlich, nach welchem Muster hier gearbeitet wird. Beim *MPC 1000* ergeben sich beispielsweise bis zu 64 Samples, die gleichzeitig im Speicher liegen und jederzeit über die Pads ausgelöst werden können. Das Umschalten zwischen den vier Pad-Banken kann live erfolgen, der Benutzer muss also wissen, welches Sample gerade mit welchem Pad verknüpft ist. Da jedes Sample exakt zurechtgeschnitten ist und so mit dem Tempo des Stücks harmoniert, ergibt sich eine Performance, die der eines Schlagzeugers ähnelt. Jede Hand bzw. jeder Finger bedient also die 16 Pads in einem bestimmten Rhythmus, während zwischendurch Effekte aktiviert und manipuliert, oder die Pad-Bank gewechselt wird. *Comfort Fit* verwendet gelegentlich zwei MPCs gleichzeitig, wobei jede Hand ein Gerät bedient - daraus ergeben sich bis zu 32 gleichzeitig abrufbare Samples.

Wie sich zeigt, entsprechen die Samples nur selten einzelnen Trommelschlägen, oder anderen kurzen Geräuschen. Vielmehr sind es Dauern von Notenlängen, die sich bei exakter zeitlicher Aneinanderreihung zu einem Rhythmus zusammenfügen. Als Ausgangsmaterial werden oft eigene Produktionen verwendet. Daraus extrahiert man beispielsweise ganze Takte des Beats, halbe Takte einer Vokalpassage oder Vierteltakte der Basslinie und weist diese Samples den Pads zu. Zusätzlich könnten die Ausschnitte mit Effekten belegt oder vielleicht in andere Tonarten transponiert werden. So ergibt sich eine Vielzahl an Möglichkeiten, ein Stück für die Aufführung vorzubereiten. Jeder der Künstler strukturiert die Anordnung der Samples auf eigene Art, was sich in unterschiedlichen Performance-Stilen widerspiegelt. Meist sind die Auftritte sehr körperbetont und virtuos in der Darbietung - dieser Umstand hat mittlerweile dazu geführt, eigene MPC-Wettbewerbe zu organisieren, ähnlich den DJ-Battles.

²¹ Mehrere Links zu Beispielen finden sich im Anhand dieser Arbeit.

6. „Project: Fingerhythm“ (Praktischer Teil)

Der folgende Abschnitt (6.1) berichtet u.a. von Erlebnissen und Erkenntnissen während des Studiums, die bei der Ideensuche geholfen und zur Entscheidung für dieses Projekt geführt haben.

Eine Erklärung zum Aufbau und den wesentlichen Elementen des Spiels wird in Abschnitt 6.2 geliefert. Außerdem wird dabei auf das Spielprinzip, die Spielmechaniken und die Spielmodi eingegangen.

In Abschnitt 6.3 wird auf die Programme eingegangen, die für die Umsetzung verwendet wurden.

Abschnitt 6.4 widmet sich schließlich der konkreten Umsetzung anhand von *MobMuPlat* und *Pure Data*. Hierbei werden die wichtigsten Schritte in der Programmierung erklärt und anhand von Abbildungen veranschaulicht.

6.1 Idee und Konzept

Erste Ideen, ein Spiel mit Fokus auf Klang zu entwickeln, entstanden bereits zu Beginn des 2. Semesters im Rahmen eines Game Jams. Innerhalb von 72 Stunden wurde gemeinsam mit drei KollegInnen das Rhythmus-Reaktionsspiel „HeartBeat“ für Smartphones konzipiert, gestaltet und präsentiert.

Kurze Zeit später habe ich durch einen Workshop die Software *MobMuPlat* kennengelernt, die eine Vielzahl an grafischen Gestaltungsmöglichkeiten für Interfaces auf Touch-Devices bietet - diese Interfaces werden meist dazu verwendet, um *Pure Data* Patches zu steuern.

Anhand einer intensiven Literaturrecherche zu den Themen Sonic Interaction Design, Gamification und Mobile Music Games wurde ich auf den Forscher und Spielentwickler Ge Wang aufmerksam. So lernte ich auch die Apps von *Smule* kennen, die später als wichtige Inspirationsquelle für mein eigenes Projekt dienen sollten.

Ergänzt durch mein persönliches Interesse an sample-basierter Produktion elektronischer Musik sowie der Faszination für Performance-Möglichkeiten mittels MPCs, wurde schließlich die konkrete Idee zu *Project: Fingerhythm* (kurz: ProFi) geboren und die Konzeption konnte beginnen.

Ziel des Projekts war es, eine Anwendung zu entwickeln, die dem Benutzer einen spielerischen Zugang zum Experimentieren mit Klängen und Musik bietet. Außerdem sollte der Benutzer die Möglichkeit bekommen, Wissen und Informationen über einfache musiktheoretische Aspekte sowie die Geschichte von Popmusik zu sammeln. Diese „Lernziele“ sollen aber nicht zwingend, sondern optional sein. Der Benutzer sollte außerdem Fingerfertigkeit, Gehör und Rhythmusgefühl trainieren können und natürlich Spaß dabei haben.

So wurde ProFi als Spiel²² realisiert, dessen Gameplay vorrangig auf Rhythmus und Timing basiert. Diese Spielmechanik ist eine direkte Anlehnung an die sample-basierte Performance mit einem MPC. Sie ähnelt auch jener von klassischen Reaktionsspielen wie *Guitar Hero*, allerdings mit einem wichtigen Unterschied: Meistens läuft bei diesen Spielen ein Song im Hintergrund, zu dem der Benutzer gewissermaßen „dazu“ spielt - bei ProFi löst der Spieler durch das Betätigen virtueller Tasten im richtigen Rhythmus die eigentliche Musik (Audiosamples) erst aus - stimmt dabei das Timing nicht, verstummt die Musik.

6.2 Erklärung und Aufbau des Spiels

Project: Fingerhythm ist in erster Linie ein Rhythmus-Reaktionsspiel, das dem Benutzer ein gewisses Maß an Fingerfertigkeit und Timing abverlangt. Abgesehen davon kann der Spieler sein Wissen über Musiktheorie, die Geschichte der elektronischen Popmusik und die Technologien dahinter erweitern. Zusätzlich kann selbst mit Musik, Samples und Effekten experimentiert und improvisiert werden.

Das grundlegende Spielprinzip besteht darin, bunte virtuelle Tasten in richtiger Reihenfolge und Timing zu drücken, um den Fluss der Musik zu erhalten. Die Richtung verläuft dabei normalerweise von links nach rechts. Jede Taste löst ein Sample bestimmter Länge (Dauer) aus und gibt dem Spieler durch ihre Farbe bzw. Größe einen Hinweis auf diese Länge. Die Herausforderung besteht darin, sich vorrangig auf den Rhythmus der Musik zu konzentrieren, weil dieser die erforderlichen Timings bestimmt, also jene Momente, in denen das nächste

²² Beim ursprünglichen Konzept zu ProFi lag der Fokus auf dem Gamification-Aspekt - bei dieser Herangehensweise wäre es allerdings nötig, eine Art virtuelles Instrument zu entwickeln (wie z.B. die Anwendung *Ocarina* von Smule) und den Benutzer, anhand von Elementen aus Spielen (Punkten, Belohnungen, Ranglisten, etc.), eben zur Benutzung des Instruments zu motivieren. Da es mir aber vorrangig um die spielerische Interaktion ging, wurde mir bewusst, dass die Gamification-Aspekte von diesem nur ablenken würden.

Sample ausgelöst werden muss. Weiters muss man sich als Spieler merken, welche Tasten für welche Dauern stehen und ob die Tasten nur angetippt, oder gehalten werden müssen. Hat man diese Aspekte verinnerlicht, stellt sich, während man die Sequenzen durchläuft, ein gewisser Überraschungseffekt ein, da man ja nicht weiß, was als nächstes erklingen wird. Diese Vorfreude, zu erfahren, wie es musikalisch „weitergeht“, soll den Spieler motivieren, sich die Funktionen der Tasten gut einzuprägen. Da für gute Timings auch Punkte vergeben werden, wird der Benutzer dazu animiert, zeitlich so exakt wie möglich zu „spielen“.

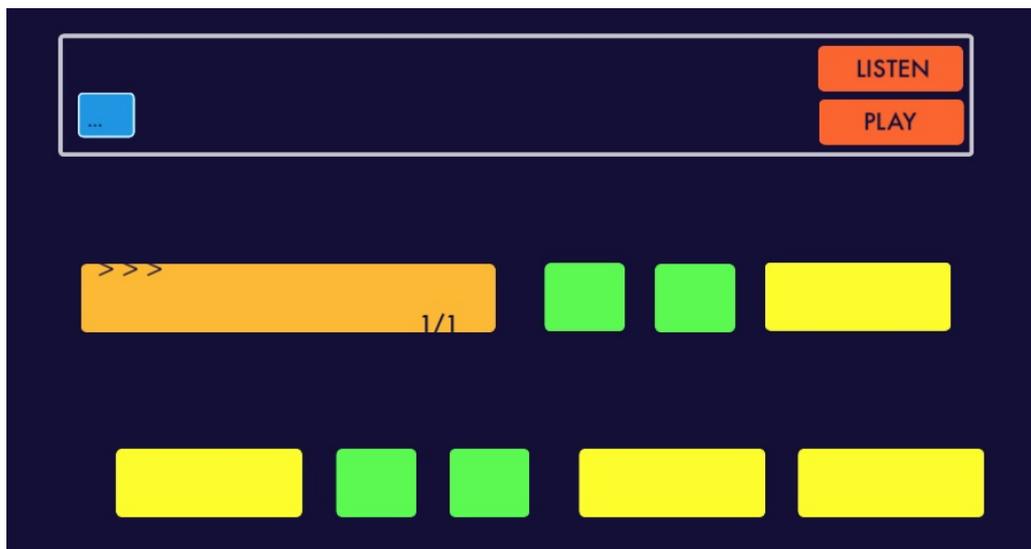


Abb.3: Ausschnitt aus Project: Fingerhythm (Tutorial)

Abb. 3 zeigt ein Beispiel aus dem Tutorial. Die hellorangefarbene Taste startet die Sequenz und löst ein Sample mit der Dauer eines ganzen Taktes aus. Die grünen Tasten entsprechen der Dauer eines Vierteltaktes, die gelben der eines halben Taktes.

Es geht also um das zeitlich exakte Aneinanderreihen von Samples, wodurch die zugrunde liegende Sequenz (beispielsweise 4 Takte eines Hip-Hop Beats) zusammengefügt wird. Je genauer der Spieler die Taste erwischt, desto mehr Punkte bekommt er. Dieses Timing äußert sich natürlich auch akustisch. Drückt man eine Taste zu früh, überlagern sich die Samples für einen Moment - drückt man zu spät, entsteht eine Pause. Der Rhythmus weicht etwas ab und der Beat beginnt zu „holpern“. Da der Benutzer nicht weiß, was als nächstes kommt (erklingt), entsteht eine Art Überraschungseffekt. In gewisser Hinsicht musiziert der Spieler automatisch, indem er die Tasten richtig (zeitlich exakt)

bedient, ohne die Musik „dahinter“ zu kennen. Er „kennt“ nur den Rhythmus bzw. orientiert sich an ihm. Durch den Einsatz von vorprogrammierten Sequenzen baut der Spieler somit die Musik erst auf bzw. „lässt sie erklingen“. Ist zwischendurch das Timing nicht exakt genug, also außerhalb eines programmierten Zeitfensters, verstummt die Musik - somit muss die Sequenz von neuem gestartet werden.

Sobald das Spiel geladen wird, öffnet sich der Startbildschirm (vgl. Abb.4) und es erklingt Musik. Kurz darauf wird der Titel des Spiels eingeblendet und die Starttaste erscheint. Drückt man diese, ertönt ein Soundeffekt und die Musik verstummt - gleichzeitig erscheint über der Starttaste ein blinkendes Pfeilsymbol, das dem Spieler bedeutet, nach links über den Bildschirm zu wischen, um zum Hauptmenü zu gelangen.



Abb.4: Der Startbildschirm von ProFi

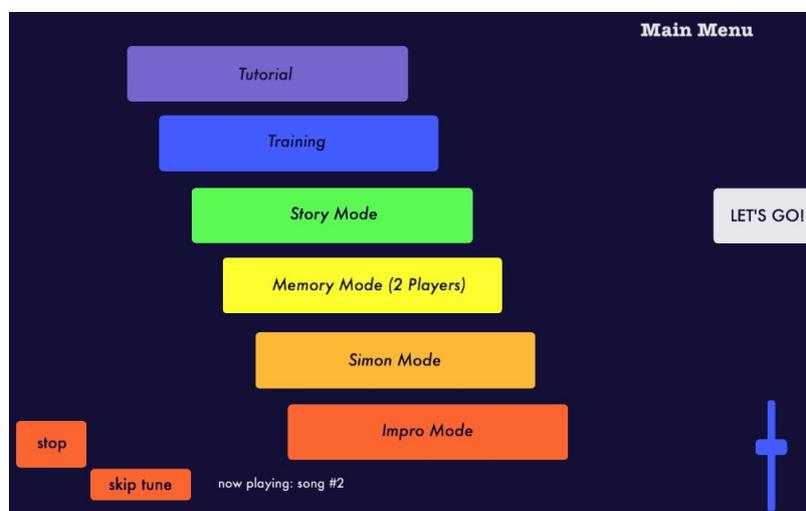


Abb.5: Das Hauptmenü

Im Hauptmenü (Abb.5) angekommen, erklingt wiederum Hintergrundmusik, deren Titel im unteren linken Bereich in weißer Schrift angezeigt wird. Der Spieler kann diese mithilfe der Tasten *stop* und *skip tune* entweder stoppen, oder einen anderen Song auswählen - in letzterem Fall verstummt der aktuelle Song abrupt und der nächste wird allmählich eingeblendet. Mit dem blauen Regler am unteren rechten Rand, kann die Lautstärke justiert werden. Der Spieler kann nun aus den angebotenen Menüpunkten einen auswählen, indem er diesen antippt - dabei leuchtet die entsprechende Taste kurz auf und es erklingt ein Soundeffekt. Zusätzlich erscheint auf der weißen Taste am rechten Bildschirmrand der Schriftzug *LET'S GO!*. Wird diese daraufhin angetippt, gelangt man zum ausgewählten Modus.

Das Tutorial soll den Benutzer mit der Spielmechanik vertraut machen und ihm wichtige Elemente sowie deren Funktionen erklären. Es besteht aus insgesamt fünf Abschnitten („Seiten“), die in aufsteigender Reihenfolge absolviert werden sollten. Grundsätzlich hat der Spieler aber jederzeit die Möglichkeit, eine Seite zu überspringen bzw. zum Hauptmenü zurückzukehren.

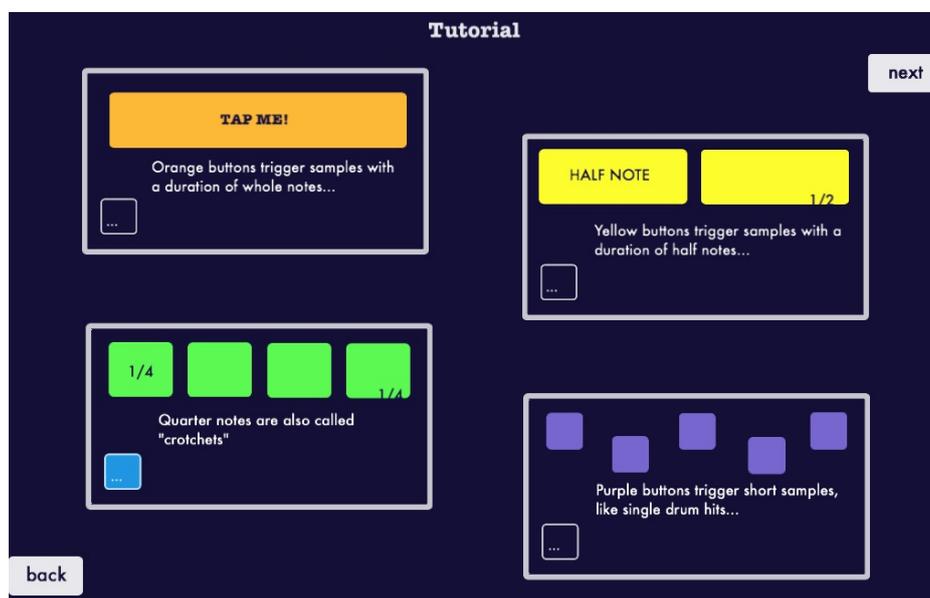


Abb.6: Tutorial - Seite 1

Auf der ersten Seite (vgl. Abb.6) werden die häufigsten Elemente erklärt - jene bunten Tasten, welche die Audiosamples auslösen. Drückt der Spieler eine Taste, erklingt ein Sample mit der entsprechenden Dauer und darunter erscheint ein Text mit Informationen dazu. So wird der Benutzer z.B. darauf hingewiesen, dass die kleinen violetten Tasten meist sehr kurze Samples, wie

einzelne Trommelschläge eines Schlagzeugs auslösen. Tippt man eine der weiß umrahmten Tasten mit den drei Punkten an, erscheinen weiterführende Informationen.

Auf Seite 2 (vgl. Abb.7) des Tutorials können weitere Objekte ausprobiert werden. Der Spieler wird anhand des Textfensters im oberen Bereich des Bildschirms dazu ermutigt, mit allen Objekten zu interagieren. Zusätzlich gibt es den Hinweis auf spezielle Tasten (markiert mit einem Pfeilsymbol), die man länger gedrückt halten muss, um das zugrunde liegende Audiosample zu hören bzw. abzuspielen. Es findet sich wieder ein blauer Lautstärkeregler sowie ein gelber Drehknopf, mit dem ein Filter gesteuert werden kann.

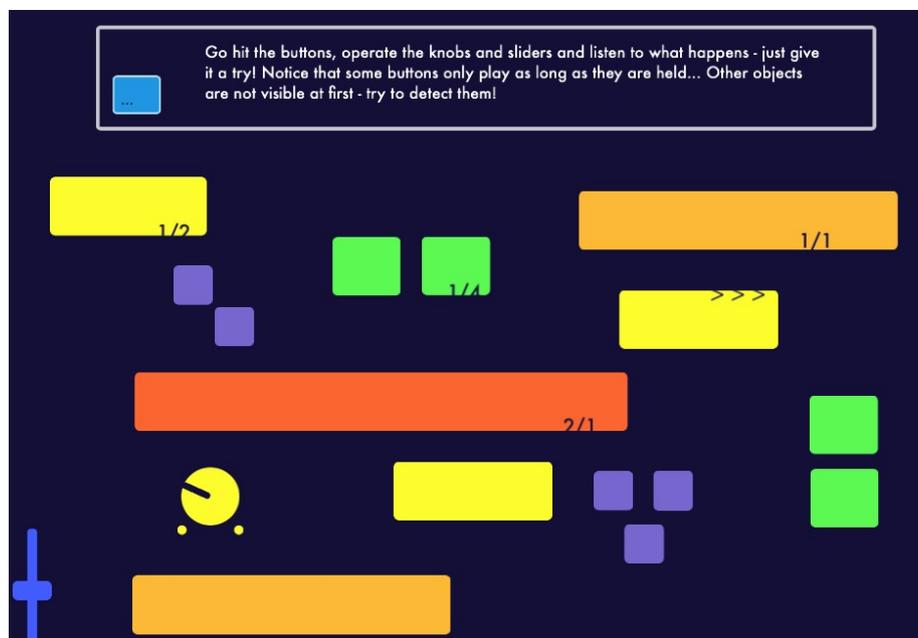


Abb.7: Tutorial - Seite 2

Der Benutzer bekommt auch den Hinweis auf verborgene Objekte, die entdeckt werden können. Tippt man auf das hellblaue Symbol innerhalb des Textfensters, kommen dazu weitere Informationen. Die Tasten auf dieser Seite sind mit Samples unterschiedlicher Musikstile (und unterschiedlicher Tempi) verknüpft - dadurch kann der Spieler ein Gespür für die „Notenlängen“ bekommen. So dauert beispielsweise ein ganzer Takt eines gemäßigten Hip-Hop Beats wesentlich länger als z.B. ein Takt Drum'n'Bass. Auch dazu kann der Benutzer optional mehr Informationen einholen, indem er das Symbol im Textfenster mehrmals betätigt.

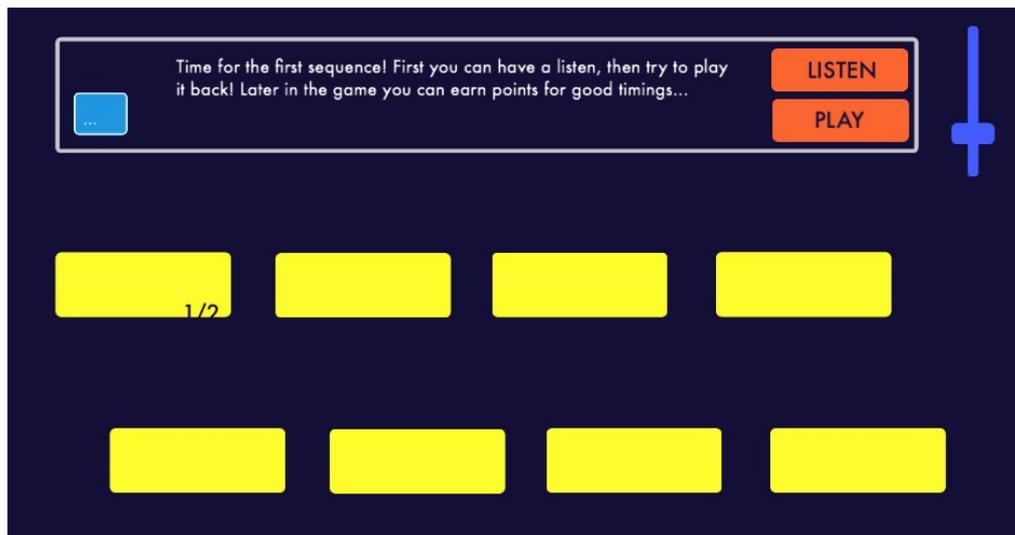


Abb.8: Tutorial - Seite 3

Seite 3 des Tutorials (vgl. Abb.8) enthält die erste zusammenhängende Sequenz, die es zu absolvieren gilt. Ihr zugrunde liegt ein 4-taktiger Ausschnitt aus einem Hip-Hop Song der Gruppe *Jurassic 5*. Das Textfenster enthält, neben den Informationen für den Spieler auch eine *LISTEN*- sowie eine *PLAY*-Taste.

Der Regler für die Lautstärke befindet sich nun in der rechten oberen Ecke des Bildschirms. Der Benutzer kann sich die nachzuspielende Sequenz erst anhören, indem er auf *LISTEN* drückt. Daraufhin wird die gesamte Sequenz abgespielt und jede der gelben Tasten blinkt zu dem Zeitpunkt, an dem sie gedrückt werden muss, kurz auf - dies dient zur visuellen Orientierung für den Spieler. Drückt man schließlich auf *PLAY* verstummt die Musik und ein Vokalsample ertönt: „Kick that fat bag beat, yo!“

Während des Tutorials erfolgt noch keine Punktvergabe für besonderes Timing, allerdings erklingen die Samples nur, wenn sie innerhalb eines definierten Zeitfensters ausgelöst werden. Jede der Tasten entspricht einem halben Takt der Sequenz. Die erste Taste löst den ersten halben Takt aus, die zweite den zweiten halben Takt usw. Der Song hat ein Tempo von ziemlich genau 100bpm - demzufolge dauert ein Sample eines halben Takts daraus ca. 1200ms. Ein Auslösen der Samples in regelmäßigem Abstand von 1200ms würde also einen nahtlosen Übergang ergeben. Für dieses Tutorial-Beispiel wurde ein „großzügiges“ Zeitfenster von +/- 300ms gewählt - was folgendes bedeutet: Nachdem der Spieler die erste Taste gedrückt hat, darf die zweite Taste frühestens 901ms danach gedrückt werden, ansonsten löst die zweite Taste

einen Error-Soundeffekt aus und man muss von Neuem beginnen. Wird die zweite Taste erst 1500ms nach der ersten gedrückt, ertönt ebenfalls der Error-Sound, da man zu lange gebraucht hat. Erwischt der Spieler das Zeitfenster zwischen 900 und 1500ms nach Drücken der ersten Taste, wird das zweite Sample regulär ausgelöst und abgespielt. Die Funktion der dritten Taste „orientiert“ sich wiederum am Zeitpunkt des Drückens der zweiten Taste usw.

Die Informationen im Textfenster weisen den Spieler auch darauf hin, dass nach erfolgreichem Absolvieren der Sequenz - nachdem also das achte Sample ausgelöst wurde - wieder die erste Taste gedrückt werden muss, als würde man von vorne beginnen. Erst wenn auch dies „in time“ passiert, gilt der Abschnitt gewissermaßen als „geschafft“ - es ertönt ein Jubel-Sample und der Spieler kann mit dem nächsten Abschnitt fortfahren.

Die Seiten 4 und 5 des Tutorials laufen im Grunde nach dem selben Schema ab, nur dass die „Notenwerte“ variiert werden, um den Spieler an das abwechslungsreiche Gameplay zu gewöhnen. In Abschnitt 4 (vgl. Abb.9) wurde jeweils die zweite Hälfte des zweiten bzw. vierten Takts durch Viertelnoten-Werte ersetzt, sodass der Benutzer hier schneller reagieren muss.

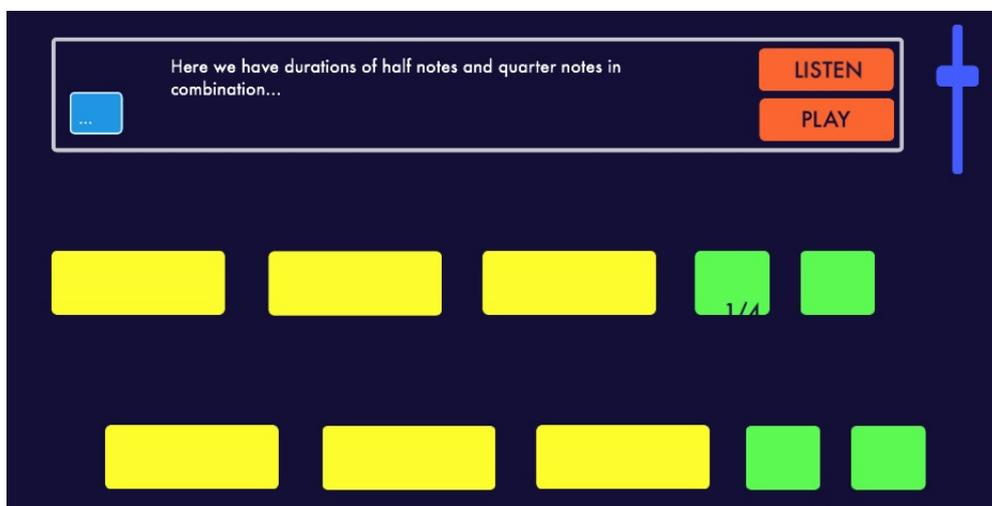


Abb.9: Tutorial - Seite 4

Dementsprechend mussten auch die Zeitfenster in der Programmierung angeglichen werden. Da ein Sample eines Vierteltakts auf 100bpm nur ca 600ms dauert, wäre ein Fenster von +/- 300ms nicht sinnvoll. Ich hab mich bei diesen Samples deshalb für +/- 150ms entschieden, was immer noch relativ großzügig ist und mir für das Tutorial angemessen erscheint.

Abgesehen von den Informationen, die das Spiel selbst betreffen, kann der Benutzer in Abschnitt 4 und 5 Wissenswertes über das Genre Hip-Hop sowie die Praxis des Sampling erfahren, indem er die hellblaue Informationstaste mehrmals drückt. Neben einer weiteren Variation in Anordnung und Dauer der Samples, muss die Sequenz in Abschnitt 5 (vgl. Abb.10) mit einer markierten Taste gestartet werden. Dieser Taste liegt das Sample eines ganzen Taktes (2400ms) zugrunde und sie muss nicht nur angetippt, sondern gehalten werden, damit die Musik auch wirklich erklingt. Es wurde entschieden, dass die Taste zumindest 1800ms lang gehalten werden muss, ohne einen Error-Sound auszulösen - also ungefähr 3/4 der gesamten Dauer. Die zweite (grüne) Taste orientiert sich, betreffend ihres Zeitfensters, am Startzeitpunkt der ersten. Wird dieser Abschnitt erfolgreich absolviert, ist das Tutorial abgeschlossen und man kehrt zum Hauptmenü zurück.

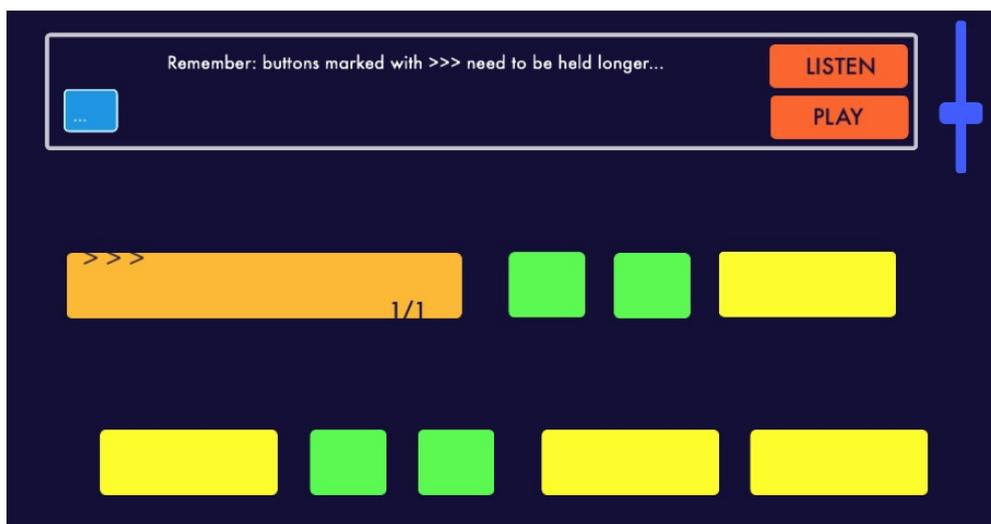


Abb.10: Tutorial - Seite 5

Im Verlauf des Story Modes (für diesen Modus wurde ein Prototyp-Level entwickelt) müssen ebenfalls Sequenzen von Musik „gespielt“ werden, aber für das Auslösen der einzelnen Samples bekommt der Spieler jedes mal Punkte. Die Musik für das Prototyp-Level stammt von der Hip-Hop Formation *Beatnuts*. Es handelt sich wieder um eine 4-taktige Sequenz mit einem Tempo von 100bpm. Gestartet wird hier allerdings mit der weißen Taste am linken Bildschirmrand, die einen Auftakt von der Länge eines ganzen Taktes auslöst. Die vier Samples der Sequenz haben je eine Länge von ca. 2400ms, entsprechen also ebenfalls je einem ganzen Takt Musik. Die Anzahl der Punkte, die der Spieler bekommt, hängt von seinem Timing ab - umso exakter die

Samples ausgelöst werden, desto mehr Punkte gibt es dafür. So wurden für das Prototyp-Level fünf Zeitfenster definiert. Die Punktvergabe setzt sich folgendermaßen zusammen: 1000 Punkte für +/- 20ms, 700 Punkte für +/- 60ms, 500 Punkte für +/- 100ms, 300 Punkte für +/-150ms, 0 Punkte für +/- 175ms. Das Sample wird also im Rahmen des letzten Zeitfensters noch ausgelöst, es werden aber 0 Punkte vergeben. Außerhalb von +/- 175ms erklingt ein Error-Sound und die Sequenz muss von neuem gestartet werden.



Abb.11: Story Mode

Das besondere am Story Mode ist, dass die Sequenz nach einem Durchgang direkt wiederholt werden kann, sofern während des ersten Durchgangs eine gewisse Anzahl an Punkten erreicht wurde. Im Prototyp-Level bedeutet das: Erreicht der Spieler zumindest eine Punktezahl von 2800 im ersten Durchgang, kann die Sequenz erneut gespielt werden - in diesem Fall ändern sich aber die Samples, die durch die Tasten ausgelöst werden.

Im Story Mode gibt es nichtmehr die Möglichkeit, die Sequenz im Vorhinein anzuhören. Allerdings kann der Spieler bei Bedarf ein Metronom aktivieren, das einen Click auf 100bpm spielt - die Lautstärke davon kann separat justiert werden. Während des Spiels blinkt beim Tastendruck für einen Moment die jeweilige Punktezahl auf. Im unteren Bereich des Bildschirms wird die aktuelle Gesamtpunktezahl angezeigt.

Der Menüpunkt „Training“ soll dem Spieler die Möglichkeit bieten, sein Timing zu trainieren und unterschiedliche musikalische Rhythmen kennenzulernen.

Dazu wurde ein Prototyp-Level (vgl. Abb.12) mit einem Loop im Stil des Bossa Nova realisiert. Dem Level liegt nur ein einzelnes Sample von der Dauer eines ganzen Taktes zugrunde - das Tempo beträgt ca 110bpm. Das Prinzip liegt hier darin, im richtigen Rhythmus zur Musik zu spielen. Wie im Tutorial Mode, gibt es eine Vorhör-Funktion. Wird diese aktiviert, startet der Loop und der horizontale Slider läuft von links nach rechts um den zeitlichen Verlauf der Musik anzuzeigen. An fünf Stellen (gekennzeichnet durch die violetten Tasten) des Taktes erklingt im Loop ein markanter Trommelschlag (Rimshot), der den Rhythmus deutlich hervorhebt.

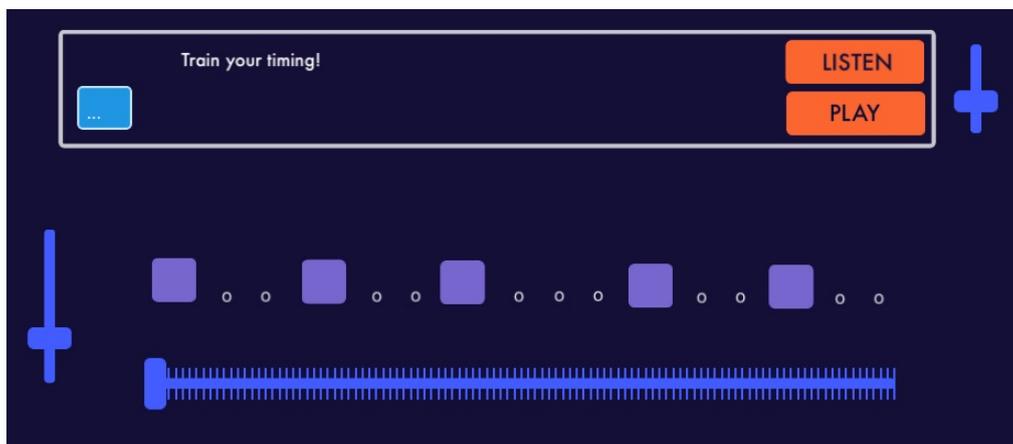


Abb.12: Training

Die violetten Tasten lösen alle ein kurzes Clave(Klanghölzer)-Sample aus. Mithilfe dieser Tasten soll der Spieler so exakt wie möglich im Rhythmus des Rimshots zur Musik spielen. Die weißen Punkte markieren die 16tel-Schläge des Rhythmus - auch daran kann sich der Spieler orientieren. Hat man den Rhythmus verinnerlicht und drückt die *PLAY*-Taste, stoppt der Loop. Sobald man nun die erste violette Taste antippt, startet der Loop erneut und man kann selbst versuchen, zum Rhythmus der Musik zu spielen. Auch hier wurden Zeitfenster programmiert, die mit +/- 30ms relativ eng gewählt wurden, um die Herausforderung zu erhöhen. Der Benutzer bekommt hier keine Punkte, sondern bei jedem Schlag kurze Kommentare, wie „perfect!“ oder „no!“ als Text eingeblendet.

Beim Memory Mode können zwei Spieler in einer Art musikalischem Memory-Spiel gegeneinander antreten. Um dem Programm zu signalisieren, welcher Spieler gerade am Zug ist, tippt man auf den Schalter neben *Player 1* bzw. *Player 2*, sodass dieser gelb leuchtet. Die grünen Felder lösen jeweils Samples

von der Dauer eines ganzen Taktes unterschiedlicher Musikstile aus, wobei immer zwei zusammenpassen. Findet man ein Pärchen, muss man die zwei Samples zeitlich exakt nacheinander antippen, sodass eine zweitaktige Sequenz erklingt. Dabei ist es egal, welches Sample zuerst ausgelöst wird. Erwischt man das richtige Timing, erklingt ein Soundeffekt und die beiden Tasten werden deaktiviert. Der entsprechende Spieler bekommt einen Punkt und darf ein weiteres Mal zwei Tasten drücken.

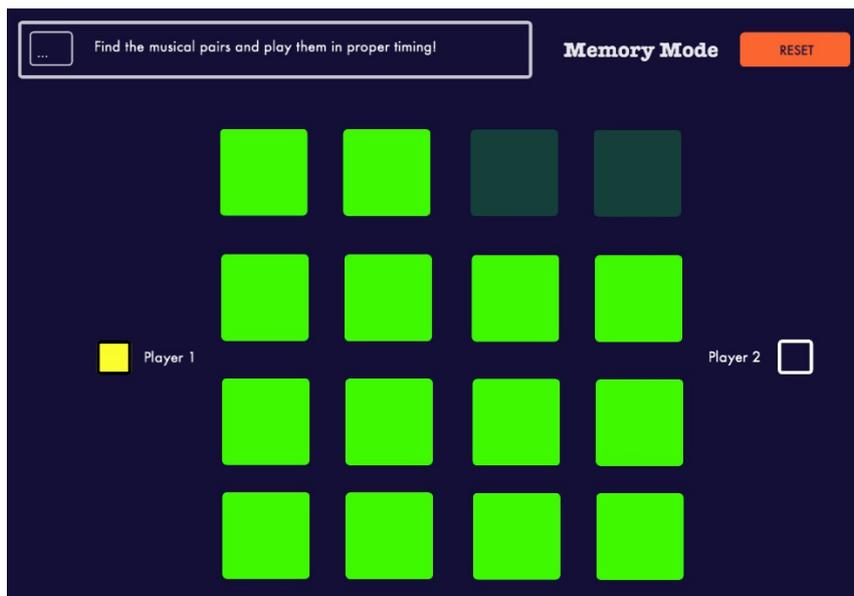


Abb.13: Memory Mode

Wie beim klassischen Memory-Spiel gewinnt derjenige Spieler, der am Ende mehr Pärchen gefunden und durch entsprechendes Timing „aufgedeckt“ hat.

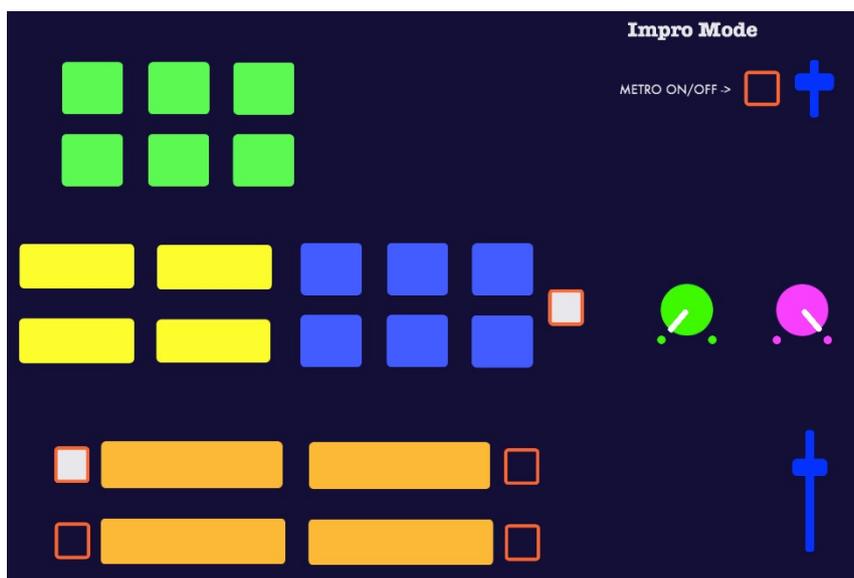


Abb.14: Impro Mode

Eine Besonderheit des Spiels ist der Impro Mode (vgl. Abb.14), in dem der Spieler frei mit Samples, Musik und Effekten experimentieren kann. Die eintaktigen Drum'n'Bass-Sequenzen (174bpm) am unteren Bildschirmrand können anhand der Schalter daneben geloopt werden. Zur Verfügung stehen ein Metronom mit eigenem Lautstärkereglern, ein High- sowie ein Low-Pass-Filter und ein Regler für die allgemeine Lautstärke. Mit den blauen Tasten können langgezogene Synthesizer-Akkorde ausgelöst werden. Aktiviert man allerdings den Schalter daneben, werden die Akkord nur solange gespielt, wie die Taste auch gehalten wird. Die gelben bzw. grünen Tasten sind ebenfalls mit Samples belegt - manche davon im Stil von Hip-Hop/R'n'B, wobei deren Tempo genau der Hälfte des Drum'n'Bass Tempos (87bpm) entspricht.

6.3 Verwendete Software

Für die praktische Umsetzung des Projekts wurden die Programme Audacity, MobMuPlat und Pure Data verwendet, welche alle im Internet frei zur Verfügung stehen (Links dazu findet sich im Anhang dieser Arbeit). Audacity wurde dazu genutzt, um die Audiosamples zu editieren bzw. zu bearbeiten. MobMuPlat diente sowohl als Interface Editor, wie auch als Applikation selbst, mit der Project: Fingerhythm am iPad geöffnet werden kann. Pure Data fungiert gewissermaßen als Game Engine, die in Interaktion mit MobMuPlat sämtliche Rechenoperationen ausführt und die Befehlsketten für die Spielmechanik steuert. Einige der Soundeffekte, wie der Error-Sound und die Bestätigungs-Sounds im Interface, wurden in Ableton Live 9 mithilfe von Software Synthesizern erstellt.

Pure Data ist eine grafikbasierte Musik-Programmierungsumgebung, entwickelt von Miller Puckette. (vgl. Wang, 2007, S. 63) Die Plattform MobMuPlat wurde von Daniel Iglesia speziell dafür entworfen, um Pure Data-Patches auf mobilen Endgeräten zu steuern bzw. mit diesen zu interagieren. (vgl. Iglesia, 2016, S. 1) Im Grunde besteht MobMuPlat aus vier Programmen: „[...] an iOS app, and Android app, an OSX editor application, and a cross-platform (Java Swing) editor application.“ (Iglesia, 2016, S. 1) Für die Erstellung von ProFi wurde davon der OSX Editor (für die Gestaltung des Interfaces) sowie die iOS Applikation (um das Programm auf dem iPad zu öffnen) verwendet.

Die prinzipielle Arbeitsweise mit diesen Programmen sieht vor, zuerst ein grafisches Interface im MobMuPlat Editor zu entwerfen. Die darin erstellten Objekte (z.B. Tasten, Drehknöpfe, Fader, usw.) senden bestimmte Datenströme aus, die von Pure Data empfangen und dort verarbeitet werden können. Die in Pure Data erstellten Objekte wiederum können ihrerseits Datenströme an Objekte in MobMuPlat schicken. Somit können nicht nur Pure Data-Patches anhand von MobMuPlat-Objekten gesteuert werden, sondern (bis zu einem gewissen Grad) auch umgekehrt. Dieser Umstand hat mich dazu bewogen, die Kombination der beiden Programme für die Umsetzung meines Projekts zu verwenden, auch wenn die „typische“ Nutzung von MobMuPlat eher im Bereich der Performance und Steuerung bzw. Erweiterung von Instrumenten liegt. (vgl. Iglesia, 2016, S. 4)

Um ein Programm, das mit MobMuPlat erstellt wurde, auf einem Endgerät zu öffnen, muss darauf lediglich die MobMuPlat-Applikation (nicht der Editor oder Pure Data) installiert sein. In den Dokument-Ordner der Applikation müssen dazu der MobMuPlat-Patch, der Pure Data-Patch sowie sämtliche verwendete Audiodateien gelegt werden.

6.4 Umsetzung und Programmierung

Um zu verstehen, welche gestalterischen Möglichkeiten die Kombination aus MobMuPlat (kurz: Mmp) und Pure Data (kurz: Pd) bietet, war es zuerst notwendig, die Datenströme zu analysieren, die zwischen den Programmen ausgetauscht werden. Im MobMuPlat-Editor gibt es für diesen Zweck ein praktisches Fenster, welches sämtliche ausgehenden und ankommenden Daten anzeigt. Erzeugt man beispielsweise ein Button-Objekt, sendet dieses seinen Status in Form von 0 bzw. 1 (nicht gedrückt bzw. gedrückt) in Richtung Pd. Um diese Daten in Pd zu empfangen, muss dort ein Receive-Objekt erstellt werden, das mit dem Namen des entsprechenden Button-Objekts in Mmp übereinstimmt. Um andererseits von Pd aus Daten an Mmp zu senden, muss ein Send-Objekt erstellt werden, das den Namen jenes Objekts enthält, das man ansprechen möchte.

Der Startbildschirm von ProFi besteht zum größten Teil aus bunten Button-Objekten, die hier rein gestalterische Funktion besitzen. Zusätzlich kommt aber Text vor, der in Mmp anhand von Label-Objekten eingebettet wird. Um den Titel

des Spiels, die Beschriftung der Starttaste sowie das darüber liegende Pfeilsymbol anzuzeigen, gäbe es also die Möglichkeit, die drei Labels zu positionieren und direkt mit Text zu füllen - diesen Textinhalt der Labels könnte man aber danach nicht mehr verändern, weshalb zuerst leere Labels erstellt wurden und der entsprechende Text von Pd aus an die Labels gesendet wird (vgl. Abb.15). Das hat einerseits den Vorteil, dass der Zeitpunkt bestimmt werden kann, wann Text angezeigt wird. Andererseits kann der Text jederzeit wieder gelöscht oder mit anderem Text ausgetauscht werden.

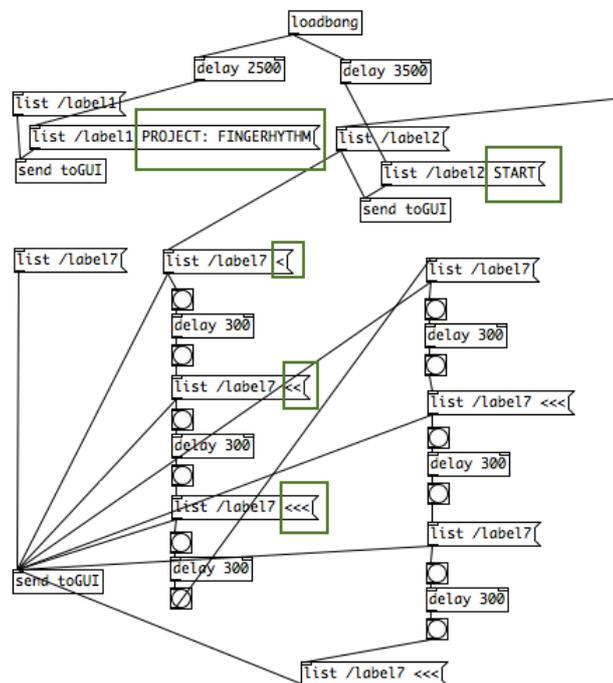


Abb.15: Ausschnitt aus Pd-Patch: Text an Labels schicken

Wie aus Abb.14 ersichtlich, wird 2,5s nachdem das Programm geladen wurde, der Titel im Label „/label1“ angezeigt und 1s später erscheint „START“ im Objekt „/label2“. Das Objekt „/label7“ wird erst mit einem Pfeilsymbol gefüllt, kurz darauf mit zwei Pfeilsymbolen und schließlich mit drei. Daraufhin wird es im Abstand von 300ms zweimal gelöscht und wieder angezeigt, was optisch einem Blinken entspricht. Um also Text zu bestimmten Zeiten an bestimmten Orten in Mmp erscheinen zu lassen, eignet sich diese Methode relativ gut. So wurde sie für ProFi auch mehrmals verwendet, z.B. für die Punktanzeige im Story Mode.

Öffnet sich der Startbildschirm, wird automatisch einer von drei Songs als Hintergrundmusik abgespielt (vgl. Abb.16). Die Auswahl passiert dabei zufällig

und die Musik (ein viertaktiger Ausschnitt) läuft so lange im Loop, bis der Spieler die Starttaste drückt.

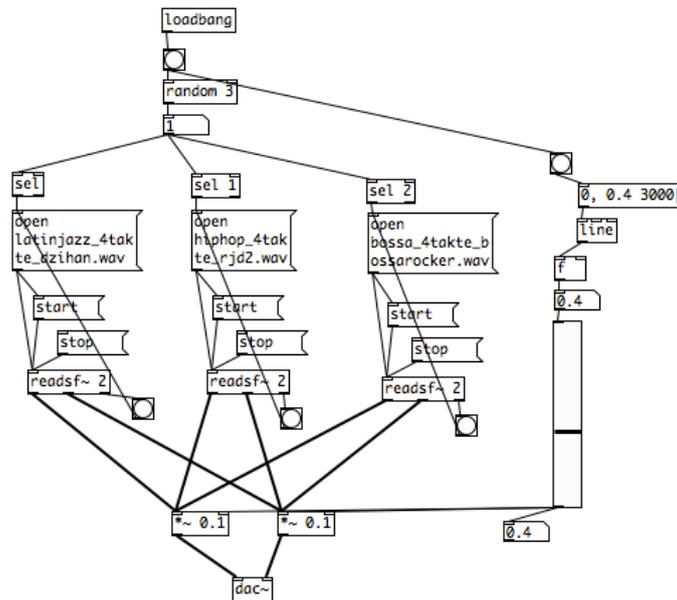


Abb.16: Ausschnitt aus Pd-Patch: Hintergrundmusik im Startbildschirm

Das Modul an der rechten Seite sorgt dafür, dass die Musik langsam (über eine Dauer von drei Sekunden) eingeblendet und nicht zu laut abgespielt wird.

Sobald man im Hauptmenü angekommen ist, wird wiederum einer von drei Songs abgespielt, doch diesmal kann der Spieler selbst eine Auswahl vornehmen, oder die Musik stoppen. Dies funktioniert über ein einfaches Zählermodul bzw. eine Stop-Message. Wechselt man die Musik, wird über ein Fader-Modul dafür gesorgt, dass der Übergang nicht zu abrupt erfolgt, sondern die Musik erst allmählich eingeblendet wird. Der blaue Lautstärkereglern im Startmenü sendet Kommazahlen zwischen 0 und 1 aus. Das entsprechende Receive-Objekt in Pd übernimmt diese Werte und überträgt es auf das dortige Fader-Objekt - somit kann der Benutzer die Lautstärke ggf. selbst nachjustieren. Die Beschriftungen der Menüpunkte wurde wieder mit Label-Objekten realisiert, wobei diese nicht verändert werden müssen und der entsprechende Text somit im Editor direkt eingegeben werden konnte. Ebenso verhält es sich mit der Beschriftung der *stop*- und *skip tune*-Tasten. Für die Anzeige der Titel der Songs allerdings wurde wieder von Pd aus gearbeitet. Je nachdem, welches Sample gerade abläuft, wird (anhand eines Message-Objekts) dessen Titel an das Label in Mmp geschickt und dort als weißer Text angezeigt. Ähnliches passiert beim *LET'S GO!*-Schriftzug (vgl. Abb.5). Dieser

wird, eine Sekunde nachdem ein Menüpunkt ausgewählt wurde, über der weißen Taste am rechten Bildschirmrand angezeigt.

Da ProFi sehr viele Audiosamples enthält, mussten diese zu Beginn des Projekts ausgewählt und so exakt wie möglich editiert werden, was mithilfe von Audacity durchgeführt wurde. Weil später die Zeitfenster für das Auslösen der Samples im Millisekundenbereich liegen sollten, wurde beim Editieren auch in diesem Bereich gearbeitet. Um Zero-Crossings zu treffen, wurden dabei Abweichungen von wenigen Millisekunden in Kauf genommen. Viele der nachzuspielenden Sequenzen wurden mit Elementen aus 4-taktigen Ausschnitten zusammengestellt. So wurde die 4-taktige Basis zuerst in vier ganze Takte, danach in acht halbe und schließlich in 16 Vierteltakte unterteilt.

Um Audio in Pd abzuspielen, gibt es verschiedene Möglichkeiten. Anhand des [tabplay~]-Objekts können Samples relativ schnell abgerufen werden, da sie vorher in einen Speicher gelegt werden. Dieses eignet sich aber nur für Mono-Samples gut. Um ein Stereosample damit abzurufen, müsste man jeden Kanal separat einlesen, im Speicher ablegen und danach die Kanäle parallel abspielen lassen. Ich habe mich dazu entschieden, vorrangig das [readsf~]-Objekt zu verwenden, welches (in Kombination mit einem Message-Objekt) Audiodaten in Stereo direkt von der Festplatte abspielt. Da Pd (und in der Folge auch Mmp) weiß, wo es nach den Daten suchen muss, führt diese Methode nur zu geringen Latenzen.

Um ein Sample in Pd anhand eines Button-Objekts via Mmp abzuspielen, muss ein wichtiges Detail beachtet werden. Da beim Drücken eines Buttons in Mmp die Werte 1 und 0 knapp nacheinander als Daten gesendet werden, muss man dafür sorgen, dass das Sample in Pd nur bei einem Wert von 1 ausgelöst wird. Andernfalls wird es zweimal hintereinander abgespielt.

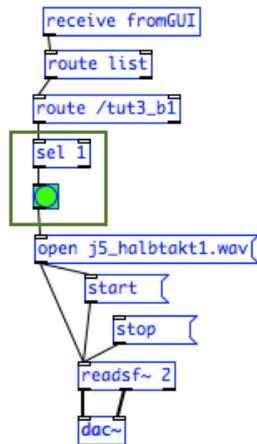


Abb.17: Modul zum Auslösen eines Samples

Abb.17 zeigt den Signalweg, bei dem die ankommenden Daten aus Mmp (und zwar genau jene vom Objekt „tut3_b1“ - einem Button-Objekt) ein Sample (nämlich die Datei „j5_halbtakt1.wav“) auslösen. Der grün eingerahmte Teil ist essentiell dafür, dass das Signal ausschließlich bei einem Wert von 1 weitergeleitet wird, also in dem Moment, wo der Button „niedergedrückt“ wird.

Soll ein Sample nur solange erklingen, wie ein Button-Objekt in Mmp gehalten wird, muss das Sample gestoppt werden, sobald das Objekt den Wert 0 sendet. Dies wird in Abb.18 veranschaulicht.

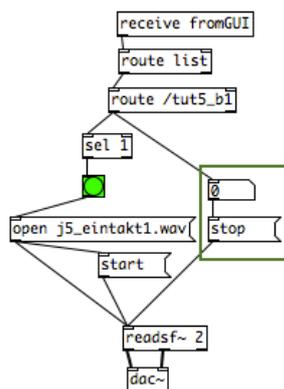


Abb.18: Modul zum Auslösen/Anhalten eines Samples

Für die Programmierung der Zeitfenster benötigt man in Pd ein Objekt, das die genauen Zeitabstände zwischen zwei Ereignissen messen kann - dafür eignet sich das [realtime]-Objekt. Dieses misst den Abstand zwischen zwei „bangs“ (eine Art „Anstoß“ oder „Impuls“ - damit werden in Pd Objekte

angesprochen oder Daten weitergeleitet) und gibt ihn als Wert in Millisekunden aus.

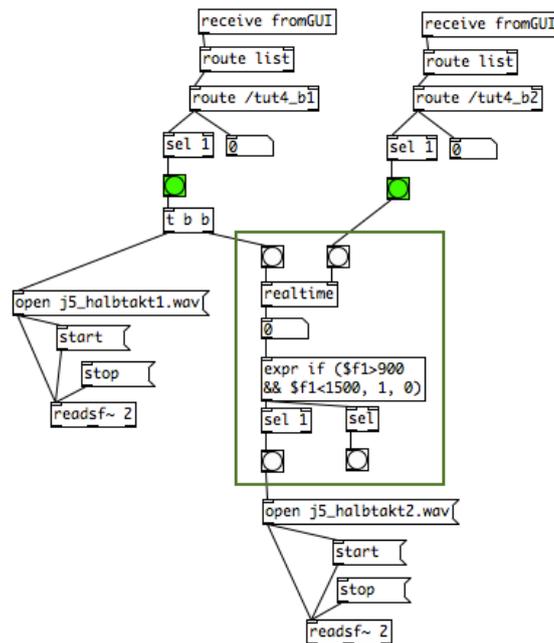


Abb.19: Zeitfenster mittels [realtime]-Objekt

Das Beispiel in Abb.19 zeigt, wie der Datenfluss mithilfe des Zeitfensters „gelenkt“ werden kann. Die erste Taste („tut4_b1“) schickt einen bang an das Realtime-Objekt und startet gleichzeitig das Sample „j5_halbtakt1.wav“. Sobald die zweite Taste („/tut4_b2“) gedrückt wird, bekommt das Realtime-Objekt seinen zweiten Wert und schickt diesen Zeitwert (ms) an ein [expr]-Objekt. Dieses prüft, ob sich der Wert zwischen den zwei Grenzen 900 und 1500 befindet - falls ja, gibt das Objekt den Wert 1 aus, woraufhin das Sample „j5_halbtakt2.wav“ gestartet wird.

Für den Story Mode war es notwendig, die Zeitfenster gewissermaßen stufenweise (rund um einen Mittelwert) zu unterteilen, um die Punktvergabe zu regeln. Dafür wird wiederum der Wert des Realtime-Objekts herangezogen (also der zeitliche Abstand zwischen dem Drücken zweier Tasten) und zuerst vom Mittelwert (dem „optimalen“ Wert) abgezogen. Dabei bekommt man den Abstand zum Mittelwert heraus. Dieser kann sowohl positiv (falls zu früh gedrückt wurde), als auch negativ (falls zu spät gedrückt wurde) sein. Um Punkte zu vergeben, muss der Wert mit festgelegten Bereichen verglichen werden, die sich rund um den optimalen Mittelwert ausdehnen. Dies geschieht anhand mehrerer [expr]-Objekte (vgl. Abb.20).

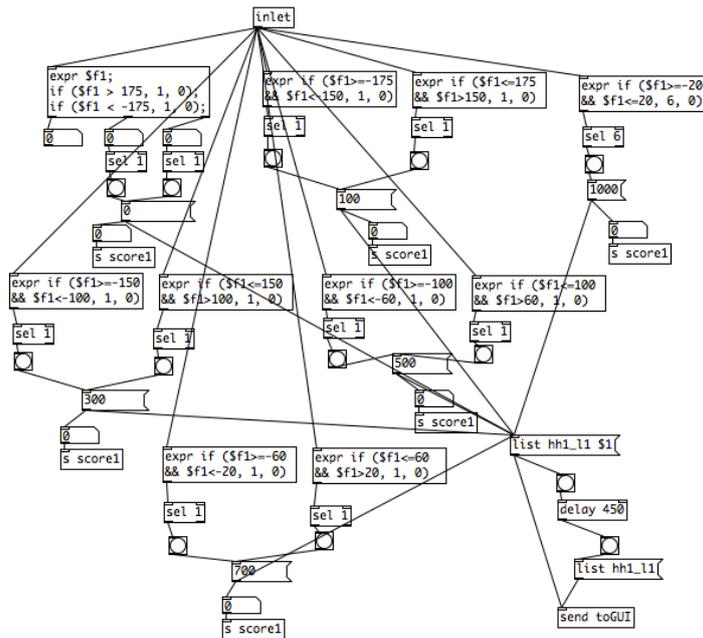


Abb.20: Kalkulation der Punktvergabe

Sobald der Bereich ermittelt wurde, wird die entsprechende Punktzahl als Text an das Label-Objekt in Mmp geschickt und 450ms lang als Hinweis über der gedrückten Taste angezeigt.

Um es im Story Mode zu ermöglichen, dass die Samples nach einem erfolgreichen Durchspielen der ersten Sequenz automatisch wechseln, wurde das [spigot]-Objekt eingesetzt. Dieses fungiert gewissermaßen als Schranke, die nur unter bestimmten Umständen Daten durchfließen lässt. So kann eine Taste in Mmp mit mehreren Samples in Pd verknüpft werden und abhängig von der aktuellen Punktzahl des Spielers wird entschieden, welches der Samples abgespielt werden soll.

7. Zusammenfassung

Wie sich zu Beginn des theoretischen Teils der Arbeit gezeigt hat, beschränkte sich der Wettkampf zwischen den Herstellern von Spielkonsolen nicht nur auf schnellere Prozessorleistung oder bessere Spielgrafik - auch die Leistung der Soundchips war dabei wichtig. Ihre stete Weiterentwicklung und Verbesserung bot den Programmierern und Entwicklern immer wieder neue Möglichkeiten, stellte sie aber auch vor neue Herausforderungen.

Die wissenschaftliche Literatur aus dem Gebiet der Game Studies weist deutlich darauf hin, dass bei der Erforschung von Videospiele der Spieler selbst ein wichtiges Untersuchungsobjekt darstellt. Ebenso verhält es sich im Bereich der Gamification-Forschung und bei der Entwicklung von Lernspielen. Um wesentliche Elemente und Modelle für spätere Gestaltungszwecke zu erkennen und diese nutzen zu können, müssen die komplexen Prozesse, die zwischen Anwendern und Anwendungen passieren, systematisch untersucht werden.

Die Betrachtung der unterschiedlichen Formen von Musikspielen hat gezeigt, dass sie in der umfangreichen Welt der Videospiele eine gewisse Sonderstellung einnehmen. So können sie soziale Interaktionen fördern, die Benutzer zu körperlichen Höchstleistungen motivieren, die Kreativität der Spieler anregen, musikalisches Wissen vermitteln, oder Menschen, trotz visueller Einschränkungen, in virtuelle Welten mitnehmen.

Die Zielsetzung, eine Anwendung für den spielerischen Zugang zum Musizieren bzw. den spielerischen Umgang mit Klängen zu gestalten, wurde mit „Project: Fingerhythm“ im wesentlichen realisiert. Vielleicht können durch diese Herangehensweise auch Menschen erreicht werden, die sich normalerweise nicht besonders für Musik interessieren - was in jedem Fall wünschenswert wäre. Besonders der Impro Mode bietet dabei Platz zum Experimentieren und Ausprobieren. Das Projekt ist außerdem ausbaufähig - so wäre es z.B. möglich, einen speziellen Modus für blinde und sehbehinderte Menschen zu entwickeln.

Es ist schwer vorauszusagen, wohin uns die aktuellen Tendenzen zu erweiterter und virtueller Realität in der Spielentwicklung klanglich führen werden. Eines steht allerdings fest: Das Feld der musikalischen und musikbezogenen Spiele

war von Beginn an ein höchst kreativer Ausgangspunkt für innovative Ansätze und ungewöhnliche gestalterische Ideen, oder, um es mit den Worten von Martin Pichlmair zu formulieren: „Yet the one type of games that opened itself up to experimentation like no other [...] is the music video game“ (2007, S. 1)

8. Bibliografie

Bernardo, F. (2014). Music Video Games in Live Performance: Catachresis or an emergent approach?. In *Videojogos 2014-Conferência de Ciências E Artes Dos Videojogos*.

Bozic, D. (2014). Analysis of Gamification Mechanisms in Mobile Service Applications. Masterarbeit, KFU Graz.

Brewster, B., & Broughton, F. (1999). Last night a dj saved my life: the history of the disc jockey. New York: Grove Press.

Collins, K. (2008a). Game sound: an introduction to the history, theory, and practice of video game music and sound design. MIT Press.

Collins, K. (Hg.) (2008b) From Pac-Man to Pop Music. Ashgate.

Collins, K. (2008c). Grand Theft Audio? Popular Music and Intellectual Property in Video Games. In *Music and the Moving Image*, Band 1, Heft 1. S. 35-48.

Collins, K., Kapralos, B., Hogue, A., & Kanev, K. (2010). An exploration of distributed mobile audio and games. In *Proceedings of the International Academic Conference on the Future of Game Design and Technology* (S. 253-254). ACM.

Collins, K. (2013). Playing with sound: a theory of interacting with sound and music in video games. MIT Press.

Denk, N. (2011). Paidia und ludus im Computerspiel: eine qualitativ-empirische Untersuchung zum Spielerleben der Computerspieler. Doktorarbeit, Uni Wien.

Deterding, S., Khaled, R. N., Nacke, L., & Dixon, D. (2011a). Gamification: Toward a Definition. In *CHI 2011 Gamification Workshop Proceedings* (S. 1-4).

Deterding, S., Sicart, M., Nacke, L., O'Hara, K., & Dixon, D. (2011b). Gamification: using game-design elements in non-gaming contexts. In *CHI'11 Extended Abstracts on Human Factors in Computing Systems* (S. 2425-2428). ACM.

Deterding, S. (2012). *9,5 Theses on the Power and Efficacy of Gamification*. (Präsentation am Hans Bredow Institut für Medienforschung, Universität Hamburg).

Fischer, G. *Von Jägern und Samplern. Eine kurze Geschichte des Remix in der Musik*. Abrufdatum 14.09.2017, von: <http://irights-media.de/webbooks/generationremix/front-matter/von-jagern-und-samplern-eine-kurze-geschichte-des-remix-in-der-musik/>.

Fischer, G. (2013). Jäger und Sampler. Kreativität und Innovation am Beispiel des Samplings. Dipl. Arbeit, TU Berlin.

Fritsch, M. (2013). History of video game music. In Moormann (Hg.). *Music and Game*, S. 11-40.

Gärdenfors, D. (2003). Designing sound-based computer games. *Digital Creativity*, 14(2), S. 111-117.

Gharibpour, S. (2015). Going beyond the Visual Domain: Improving the UX of Sonic Interaction. Umeå, Schweden.

Gillian, N., O'Modhrain, S., & Essl, G. (2009). Scratch-Off: A Gesture Based Mobile Music Game with Tactile Feedback. In *NIME* (S. 308-311).

Guerraz, A. & Lemordant, J. (2008). Indeterminate adaptive digital audio for games on mobiles. In Collins, K. (Hg.). *From Pac-Man to Pop Music*, S. 55-72.

Halbeisen, S. (2011). *Ernste Spiele? Eine Systematisierung von Serious Games*. Dipl.-Arb., Uni Wien.

Hamari, J., Koivisto, J., & Sarsa, H. (2014). Does gamification work? a literature review of empirical studies on gamification. In *2014 47th Hawaii International Conference on System Sciences* (S. 3025-3034). IEEE.

Hoffmann, L. (2009). Learning through games. In *Communications of the ACM*, 52(8), S. 21-22.

Horowitz, S., & Looney, S. R. (2014). The essential guide to game audio: the theory and practice of sound for games. CRC.

Hübner, T. (2011). „Real guitars are for old people“ - Wie die Zusammenarbeit von Pop- und Computerspielindustrie eine Generation prägt. In Jost, C., Klug, D., Schmidt, A., & Neumann-Braun, K. (Hgg.). *Populäre Musik, mediale Musik?* S. 61-81. Short Cuts / Cross Media, Bd. 3.

Iglesia, D. (2016) The Mobility is the Message: the Development and Uses of MobMuPlat. PdCon.

Jelinek, N. (2002). Sampling Culture : DJ-Kultur als Fallbeispiel für den kulturellen Eklektizismus des ausklingenden 20. Jahrhunderts. Dipl.-Arb., Uni Wien.

Jørgensen, K. (2008). Left in the Dark: playing computer games with the sound turned off. In Collins, K. (Hg.). *From Pac-Man to Pop Music*, S. 163-176.

Kaae, J. (2008). Theoretical approaches to composing dynamic music for video games. In Collins, K. (Hg.). *From Pac-Man to Pop Music*, S. 75-91.

Kim, J., & Ricaurte, J. (2011). TapBeats: accessible and mobile casual gaming. In The proceedings of the 13th international ACM SIGACCESS conference on Computers and accessibility (S. 285-286). ACM.

Liu, Y., Yan, N., & Hu, D. (2014). Chorlody: a music learning game. In *CHI'14 Extended Abstracts on Human Factors in Computing Systems* (S. 277-280). ACM.

Marschner, M.-O. (2008). Foundations for Music-Based Games. Masterarbeit, TU Wien.

Nacke, L. E., & Deterding, S. (2017). The maturing of gamification research. In *Computers in Human Behavior*. Band 71, S. 450-454.

Noguchi, H. (1996). *Mozart - Musical Game in C K. 516f*. Abrufdatum 14.09.2017, von <http://www.asahi-net.or.jp/~rb5h-ngc/e/k516f.htm>.

- Parker, J. R., & Heerema, J. (2007). Musical interaction in computer games. In *Proceedings of the 2007 conference on Future Play* (S. 217-220). ACM.
- Parker, J. R., & Heerema, J. (2008). Audio interaction in computer mediated games. In *International Journal of Computer Games Technology, 2008*, 1.
- Pichlmair, M., & Kayali, F. (2007). Levels of Sound: On the Principles of Interactivity in Music Video Games. In *DiGRA Conference*.
- Pidkameny, E. (2002). *Levels of Sound*. Abrufdatum 14.09.2017, von <https://www.vgmusic.com/information/vgpaper2.html>.
- Plaschg, J. (2016). Kognitives Training durch Computerspielen - am Beispiel „The Legend of Zelda: Ocarina of Time“. Masterarbeit, KFU Graz.
- Poschardt, U. (1995). *DJ-Culture*. Hamburg: Rogner & Bernhard bei Zweitausendeins.
- Raessens, J. (2006). Playful identities, or the ludification of culture. In *Games and Culture, 1*(1), S. 52-57.
- Raessens, J. (2012). The ludification of culture. In M. Fuchs, S. Fizek, P. Ruffino, & N. Schrape (Hgg.). *Rethinking gamification* (S. 91-114). Lüneburg, meson press.
- Rodgers, T. (2003). On the process and aesthetics of sampling in electronic music production. In *Organised Sound, 8*(3), S. 313-320.
- Rovithis, E., Mniestris, A., & Floros, A. (2014). Educational audio game design: sonification of the curriculum through a role-playing scenario in the audio game 'Kronos'. In *Proceedings of the 9th Audio Mostly: A Conference on Interaction With Sound* (S. 21-25). ACM.
- Scherer, S. (2015). Musik und Sound in Videospielen: Funktion, Interaktion, Atmosphäre. Masterarbeit, KFU Graz.
- Stingel-Voigt, Y. (2014). Soundtracks virtueller Welten: Musik in Videospielen. Diss., Verlag Werner Hülsbusch, Glückstadt.

Stoppe, S. (2015). Yvonne Stingel-Voigt: Soundtracks virtueller Welten: Musik in Videospiele (Review zur Dissertation). In *Medienwissenschaft Rezensionen/Reviews*. Band 2 (2015), S. 286-287.

Tessler, H. (2008). The new MTV? Electronic Arts and ‚playing‘ music. In Collins, K. (Hg.). *From Pac-Man to Pop Music*, S. 13-25.

Wang, G. (2016). Game Design for Expressive Mobile Music. In *NIME*.

Wang, G. (2007). A history of programming and music. In *The Cambridge Companion to Electronic Music*. S. 55-71.

Whalen, Z. (2014). Play Along - An Approach to Videogame Music. In *International Journal of Computer Game Research*. Band 4, Ausgabe 1. Abrufdatum 14.09.2017, von <http://www.gamestudies.org/0401/whalen/>.

8.1 Anhang

Website zu Audio-Only-Games:

<http://audiogames.net/>
(Abrufdatum 14.09.2017)

Wikipedia-Liste zu Music Video Games:

https://en.wikipedia.org/wiki/List_of_music_video_games
(Abrufdatum 14.09.2017)

MPC Performance:

<https://www.youtube.com/watch?v=QXX-4DiGi8w>
<https://www.youtube.com/watch?v=U-uKfQQACg4>
<https://www.youtube.com/watch?v=aE3FrVbqqm0>
<https://www.youtube.com/watch?v=s0a6sKCHhus>
(Abrufdatum 14.09.2017)