

PILI: Push-It-Loop-It – ein interaktiver Multi-User-Sequenzer

Martin Würflein, Cornelius Pöpel, Florian Beck

Hochschule Ansbach

{martin.wuerflein, cornelius.poepel, florian.beck}@hs-ansbach.de

Zusammenfassung

Der interaktive Multi-User-Sequenzer *Pili* ermöglicht das gemeinsame parallele Komponieren und Arrangieren von Songs auf mehreren Android-Devices in lokalen Netzwerken via WLAN. Dabei können die User durch Touchscreeneingaben oder Schlagbewegungen Noten einspielen und als Loops ins Arrangement einfügen. Diese Loops können dann von jedem Mitspieler kopiert, editiert, verschoben oder gelöscht werden. Die Sounds wurden größtenteils synthetisch hergestellt. Durch dieses Arrangieren der einzelnen Loops entstehen Kompositionen. Die Bedienung integriert Sensorik und Gestensteuerung wie Schütteln, Touchgesten und Schlagbewegungen des Gerätes. Die Datensynchronisation im Mehrspielermodus folgt dem Prinzip der Client-Server-Architektur. Das Servergerät ist die zentrale Datenschnittstelle und versorgt die Clients mit den aktuellen Daten. Änderungen werden an den Server weitergegeben, der die anderen Clients updatet.

1 Einleitung

Der kommerzielle Erfolg mobiler Endgeräte wie Smartphones und Tablets ist ungebrochen. Die Absatzzahlen übertreffen sich von Jahr zu Jahr. So wurden 2013 weltweit erstmals mehr als eine Milliarde Smartphones verkauft (vgl. International Data Corporation 2014).

Dabei rückt das Themenfeld *soziale Medien* immer weiter in den Vordergrund. Chatmöglichkeiten und Plattformen wie Twitter, Flickr, Facebook und die Videoplattform YouTube beeinflussen den Alltag vieler Nutzer. Das Smartphone passt durch seine Konnektivität in das Anforderungsprofil des kleinen und intelligenten Allzeit-Begleiter, mit dem man neben dem Telefo-

nieren auch surfen, spielen, Videos ansehen, einkaufen und chatten kann. Es ist vor allem für jüngere Menschen zum Angelpunkt ihrer sozialen Aktivität geworden (vgl. Medienpädagogischer Forschungsverbund Südwest 2013).

Auch die Musik ist ein soziales Medium. Miteinander Musik hören, gemeinsames Musizieren, Anderen seine Emotionen oder Geschichten mitteilen, das alles gehört zum Anwendungsfeld der Musik. In der Nutzung von Smartphones und Tablets zur aktiven Musikerzeugung und -bearbeitung kommen beide oben genannte Bereiche zusammen. Wir gehen davon aus, dass es sich um ein vielversprechendes Forschungsfeld handelt. Die Fülle an unterschiedlicher Sensorik wie Touchscreens, Bewegungs- oder Helligkeitssensoren ermöglicht darüber hinaus die Entwicklung neuer Steuerungsformen von Musik und Klängen.

Smart Devices sind sowohl netzwerkfähig als auch mobil. Das ist auch für den Einsatz in Multi-User-Szenarien interessant. Daher erläutert diese Arbeit die Entwicklung eines interaktiven Multi-User-Musikspiels auf mobilen Endgeräten. In Kapitel 2 wird die grundlegende Idee zum vorliegenden Projekt beschrieben. Kapitel 3 erläutert die Implementierung der Software. Hierbei wird – neben den vier Softwarekomponenten Datenverwaltung, Netzwerkschnittstelle, Audio-Engine und Interface – auch auf Funktionen und Mechanismen bei der Bedienung eingegangen. Im vierten Kapitel werden die Evaluationsergebnisse, die durch eine Userstudie generiert wurden, vorgestellt. Diese Ergebnisse werden anschließend im fünften Kapitel diskutiert. Anschließend werden im sechsten Kapitel zukünftige Weiterentwicklungen diskutiert.

2 Idee

Um gemeinsames paralleles Musizieren zu ermöglichen, wurde die Plattform eines Multi-User-Sequenzers ausgewählt. Eine Gruppe von Spielern trifft in einem physischen Raum aufeinander. Dabei hat jeder User ein Smartphone bzw. mobiles Endgerät. Eines der Geräte wird nun als Host-Gerät ausgewählt, alle anderen Smartphones/mobilen Endgeräte verbinden sich mit diesem als Client. Nur von dem Hostgerät wird das aktuelle Arrangement in einer Schleife abgespielt und per Lautsprecher ausgegeben. Ein Arrangement kann bis zu sechs verschiedenen Instrumentenspuren enthalten, jede Spur besteht aus kleinen Loops.

Auf jedem der Endgeräte können nun diese kleinen Loops (1–4 Takte) für die einzelnen Instrumentengruppen – wie Schlagzeug, Pads oder Melodieinstrumente – erstellt und editiert werden. Über Berührungen auf dem Touchscreen oder durch Gesten wie z. B. Schlagbewegungen können Noten eingespielt werden. Diese Loops werden zuerst nur von dem erzeugenden Gerät synchron zum Hostgerät abgespielt. Über den „Push“-Button kann der Loop dann in das Arrangement eingefügt werden. Durch das Zusammenfügen verschiedenster Loops entsteht sukzessive eine Komposition, wobei dieser Prozess keineswegs linear ist.

Jeder User kann jeden – auch von ihm nicht erstellten – Loop immer wieder editieren oder im Arrangement verschieben. Auch die bestehenden Instrumente können während des Musizierens verändert werden oder neue Instrumente hinzugefügt werden. Diese Form der parallelen und nicht linearen Komposition soll kollaboratives Musizieren ermöglichen und zur Kreativität in der Gruppe anregen.

Ein weiteres Ziel war es, eine Applikation zu schaffen, die auf einer möglichst breiten Basis von mobilen Endgeräten lauffähig ist. Zur Umsetzung der Projektidee wurde ein breites Spektrum an Technologien und Wissenschaften benötigt. Neben musiktheoretischen Grundlagen spielen besonders Netzwerktechnologie, Sensorik, Programmierung auf mobilen Endgeräten und elektronische Musikerzeugung eine Rolle. Im Folgenden werden diese Bereiche vorgestellt.

3 Implementation

Die Applikation kann in insgesamt vier Bausteine aufgeteilt werden: die Netzwerkschnittstelle, das Datenmanagement, das Interface zum User und die Audio-/Sequenzereinheit. Diese vier Elemente werden nun genauer erläutert. Außerdem werden in diesem Kapitel Funktionen und Mechanismen bei der Bedienung der Software erklärt.

Programmiert wurde *Pili* – kurzform für *Push-it-loop-it* – mit dem Android-SDK in der objektorientierten Programmiersprache Java für die Bausteine Interface, Netzwerkschnittstelle und Datenmanagement. Die Soundengine wurde über den *Pure Data*-Port *libpd* umgesetzt.

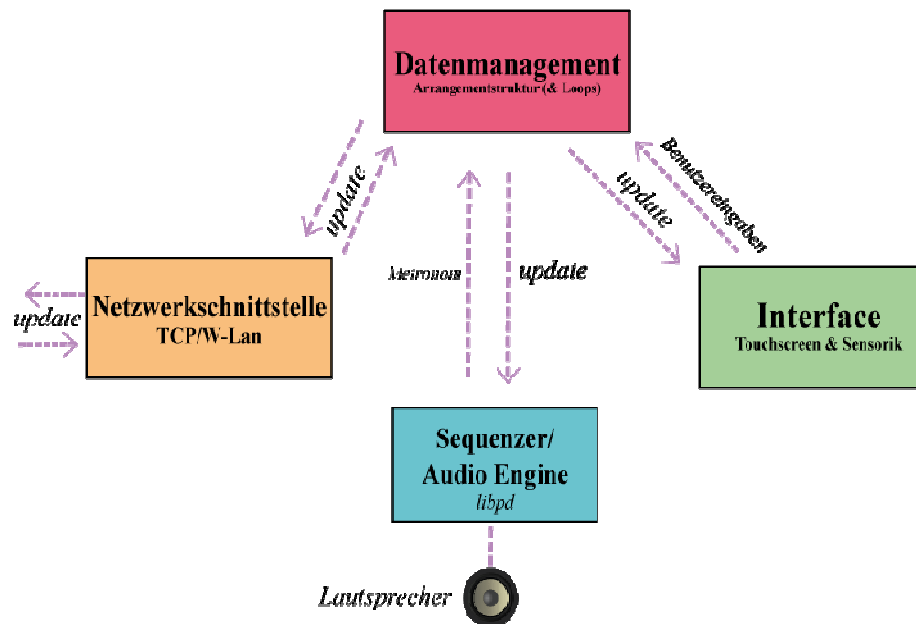


Abb. 1 Systemübersicht

3.1 Datenverwaltung

Im Zentrum der kompletten Applikation steht die Datenverwaltung, die als Grundlage für den Sequenzer, das Interface und die Netzwerkschnittstelle dient. Das Interface übergibt die Benutzereingaben, wie Noten oder Steuerbefehle, an die Datenverwaltung, die sich um die Weiterverarbeitungen kümmert. Die Datenverwaltung sorgt dann dafür, dass der Sequenzer mit den aktuellen Sequenzerdaten gespeist und aktualisiert wird. Dem Interface wird wiederum die Sequenzerstruktur zur Verfügung gestellt, um das visuelle Feedback zu erzeugen. Außerdem kümmert sich die Datenverwaltung um die Aktualisierung der anderen Teilnehmer über die Netzwerkschnittstelle oder wird über diese selbst aktualisiert.

Dabei werden auf den Client-Devices nur Strukturinformationen des Arrangements zwischengespeichert, die Sammlung und Verwaltung der einzelnen Loop-Informationen wird nur auf dem Servergerät ausgeführt. Dies reduziert die Datenkommunikation und sorgt für eine bessere Datenkonsistenz.

Möchte ein Client-Device einen existierenden Loop bearbeiten, muss dieser vom Server angefordert werden.

3.2 Netzwerkschnittstelle

Zur Kommunikation der Devices untereinander wurde als Netzwerkschnittstelle die WLAN-Technologie gewählt, da diese sehr verbreitet ist und für die Integrierung neuer Devices eine hohe Kompatibilität garantiert. Sind keine lokalen WLAN-Netzwerke vorhanden, kann über eines der Geräte ein eigenes lokales Netzwerk in Form eines Hotspots erstellt werden, über den sich bis zu neun andere Teilnehmer einwählen können.

Der Datenaustausch erfolgt über die Transportprotokolle TCP und UDP. Die Daten werden dabei jeweils über das Servergerät synchronisiert. Dabei werden UDP-Multicast-Nachrichten zur Verteilung von häufigen und dadurch unkritischen Informationen verwendet (aktueller Takt, Taktschlag). Kritische Daten – wie die Songstruktur und der Inhalt einzelner Taktinformationen – werden über das TCP-Protokoll versendet. Ändert also ein Client-Device die Struktur des gemeinsamen Songs, schickt dieser die Änderung an den Server, der die Änderung in der Struktur des Arrangements an die anderen Clients über ein TCP-Paket weitergibt.

3.3 Die Sequenzer-Einheit und Audio-Engine

Die Audio-Engine und der Sequenzer wurden mit dem *Pure Data*-Port *libpd* umgesetzt. *Pure Data* ist eine leistungsfähige und plattformunabhängige Audio-Programmiersprache, welche die Erzeugung und Echtzeitverarbeitung von Audio-Streams ermöglicht. Die *libpd* wird dabei über die Android-Applikation angesteuert, besitzt jedoch selbst direkten Zugriff auf die Audiohardware und gibt den Sound-Output somit direkt aus.

Die Android-Applikation initialisiert bei Programmstart die *libpd* und ruft sogenannte *Pure Data Patches* auf, in welchen das Audioprocessing stattfindet. So wird beim Start der App automatisch das Hauptmodul mit der Sequenzer-Einheit, dem Metronom und der Mixereinheit geladen. Die einzelnen Synthese-Einheiten/Instrumente werden erst bei Bedarf dazu geladen, um Ressourcen zu sparen. Momentan gibt es sieben Instrumente und drei verschiedene Drumkits. Die Instrumente bieten vierstimmige Klangerzeugung. Alle Sounds – bis auf zwei Drumkits – wurden über Syntheseformen wie FM-Synthese, additiver Synthese und subtraktive Synthese umgesetzt.

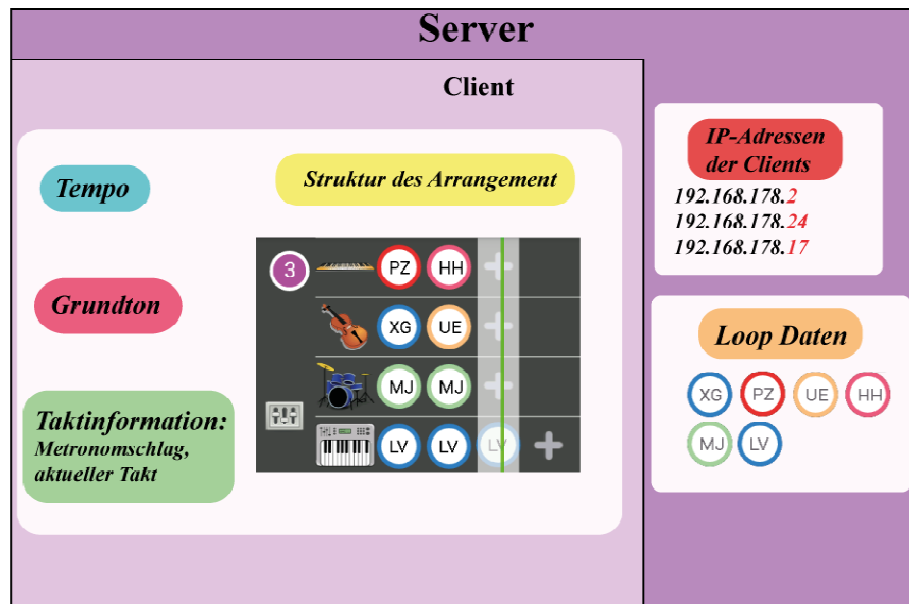


Abb. 2 Datenmanagement

Die Datensätze für die Songstruktur und für die einzelnen Loop-Informationen werden von der Sequenzer-Einheit über Textdateien eingelesen, die von der Applikation erstellt und aktualisiert werden. Über einen Steuerbefehl teilt die Android-App der Sequenzer-Einheit mit, welche Loops und welcher Takt aktualisiert werden müssen. Durch Steuerbefehle können außerdem Werte wie Tempo, Tonart und aktueller Takt angesteuert werden.

3.4 Interfacedesign

Ziel war es, ein visuelles Interface zu erzeugen, das ansprechend, einfach und übersichtlich ist. Deswegen konzentrieren sich die einzelnen Applikationsfenster jeweils auf wenige Kernaufgaben. Eine farbenreiche Gestaltung des Logos und des Interfaces soll den spielerischen Charakter der App unterstreichen. Bedienelemente wurden durch visuelle und metaphorische Repräsentationen der Funktion dargestellt. So werden beispielsweise das Metronom durch ein pulsierendes Auge, die Löschfunktion durch einen Radiergummi und die Loops durch kleine Blasen dargestellt.

Weiterhin spielt die Responsiveness, also die Anpassung der Benutzeroberfläche an unterschiedliche Displaygrößen, bei Smartphone-Devices eine wichtige Rolle. Außerdem sind visuelle und sensorische Rückmeldungen an

den Anwender ein wichtiger Aspekt des Interaktionsdesigns. Die richtige Dosierung an Feedback zu geben, ohne den User zu überwältigen, ist eine schwierige Aufgabe (vgl. Mauney & Masterton 2008).

Der Nutzer erhält darum ständige Rückmeldung in Form von dezenten visuellen Cues und Sounds. Wird beispielsweise ein Loop markiert, wird dieser in seiner Darstellung vergrößert und dies mit einem Ploppgeräusch unterstrichen. Wird das Gerät geschüttelt, gibt es Feedback in Form eines Schüttel-sounds.

Zur Steuerung des Interfaces werden Touchscreen-Eingaben, Accelerometer- sowie Magnetometerdaten ausgewertet. So werden auf dem Display einzelne Klick-Events, Wisch- und Zeichenbewegungen und Zweifingergesten wie Zoombewegungen erkannt. Dazu gibt es noch ein Bewegungserkennung mithilfe der Daten von Accelerometer und Magnetometer. Schütteln wird erkannt, wenn sich über einen Zeitraum von über 200 Millisekunden die Summe aus den Werten der x -Achse, y -Achse und z -Achse kontinuierlich verändert.



Abb. 3 Schlagbewegung

Eine Schlagbewegung kann erkannt werden, wenn die Orientierung des Gerätes bekannt ist. In diesem Fall wird davon ausgegangen, dass der Bildschirm Richtung Himmel zeigt und bei ruhiger Haltung eine Kraft von ca. $+9,81 \text{ m/s}^2$ auf der z -Achse und jeweils eine Kraft von ungefähr 0 m/s^2 auf x - und y -Achse wirkt. Im freien Fall würde der z -Wert 0 m/s^2 betragen. Verkleinert sich nun eindeutig der Beschleunigungswert in der z -Achse, ohne dass sich die Beschleunigung in der x - und y -Achse signifikant verändert,

kann davon ausgegangen werden, dass eine Bewegung nach unten ausgeführt wurde.

Abbildung 4 zeigt den Loopeditor, in dem innerhalb der Ansicht gescrollt werden kann. Der Nutzer kann das Scrolling durch die Rotation um die horizontale Bildachse steuern. Dafür wird eine in Android implementierte Funktion genutzt, welche mittels *Sensor Fusion* (Kombination) die Magnetometer- und Accelerometer-Daten ausliest und die Neigung des Gerätes in Relation zum Weltkoordinatensystem berechnet.



Abb. 4
(links) Schütteln löscht Loop (rechts); Mehrfingergeste zum Kopieren von Loops

3.5 Musizieren mit Pili – Funktionen und Mechanismen

Notenwerte können auf drei verschiedene Arten eingespielt werden. Es kann zwischen den drei Einspielvarianten Zeichenmodus (Pen), Schlagmodus (Strike) und dem Touchmode gewählt werden.



Abb. 5 Einspielmöglichkeiten Pen-, Strike- und Touchmodus

Zuerst muss der Notenwert durch Berühren des entsprechenden Symbols gewählt werden. Die rhythmische Position und Länge wird durch Einzeichnen im Taktfenster (Pen-Modus), rhythmisches Schlagen des mobilen Endgerätes (Strike-Modus) oder durch rhythmisches Anklicken des Touchscreens

(Touch-Modus) erzeugt. Die Eingabe erzeugt dabei jeweils ein direktes Audio-Feedback in Form des jeweiligen Tones.

Die Touchscreens und die Sensorik von mobilen Endgeräten weisen noch erhebliche Latenzzeiten auf. Delays bei Touchscreen-Eingaben sind momentan noch bei allen Geräteherstellern und Plattformen ein Thema. Die niedrigsten Latenzzeiten weisen nach einem Test der Firma Agawi die aktuellen Apple-Geräte mit Delays von ca. 70–90 ms auf. Die Auswahl der getesteten Android-Geräte zeigten Latenzzeiten von 114 bis ca. 170 ms (vgl. Agawi 2013). Auch die in Smartphones verwendeten Sensoren wie Magnetometer oder Accelerometer sind ungenau und langsam. Die hohe Dichte an Elektronik in den mobilen Endgeräten führt zu Rauschen in den Sensorkomponenten.

Aus den eben angeführten Gründen ist eine Quantisierung der Eingaben unumgänglich. Grundsätzlich liegt jedem rhythmischen Ablauf eine gleichmäßig pulsierende Schlagfolge zugrunde (vgl. Sikora 2003). In *Pili* wird mit einem 4/4 Takt gearbeitet. Die Quantisierung erfolgt in Schritten von bis zu 1/16 Notenschlägen. Bevorzugt wird jedoch auf ganze und dazu auf halbe Zählzeiten quantisiert. Die Eingabe von Triolen ist momentan noch nicht möglich.

Ein weiterer wichtiger Aspekt für die Melodiebildung ergibt sich aus dem musikalischen Material, aus dem die Melodie erstellt werden kann. Die Darstellung im Loop-Fenster ist auf gleichzeitig sechs Tonschritte begrenzt, denn Anfänger sollten nicht überfordert werden und schnell brauchbare Resultate schaffen können. Erfahrenere Nutzer sollten bei Bedarf mehr Möglichkeiten zur Verfügung haben. Darum kann der Nutzer zwischen verschiedenen Ansichten wählen: den zwölf Halbtonschritten, der Dur-Tonleiter, der Moll-Tonleiter und der Pentatonik.

Gerade Anfänger können mit der Pentatonik gut klingende Melodien erschaffen. Die Pentatonik lässt sich problemlos zu allen Dur-Akkorden kombinieren. Häufig findet man die Pentatonik in moderner Rockmusik sowie ostasiatischer Musik. Die Pentatonik ist das älteste Tonsystem mit einem Alter von 3700 Jahren und wurde in den unterschiedlichsten Kulturkreisen unabhängig voneinander entdeckt. Die pentatonische Skala besteht aus fünf Tönen, wobei jeweils nur Ganztonschritte oder kleine Terzen vorkommen und womit dadurch Dissonanzen vermieden werden.

Um den Einstieg in die Komposition noch weiter zu erleichtern, wurde eine Funktion entwickelt, mit der fertige Loops ins Arrangement eingefügt werden können. Mit der *Random4*-Funktion werden beispielsweise vier zu-

einander passende Loops eines Instruments ins Arrangement eingefügt. Für jeden der Instrumententypen existiert ein Loop-Pool, aus dem die Loops gewählt werden. Bei Akkordinstrumenten werden jeweils Akkorde einer Kadenz gebildet.

4 Ergebnisse

Die ersten Reaktionen von Personen, die das Produkt getestet haben, waren positiv. Vor allem Kinder waren sehr neugierig und hatten Spaß an der Bedienung des Interfaces, wobei für sie die Erzeugung eines „klingenden“ musikalischen Outputs weniger relevant erschien. Ältere Benutzer reagierten etwas zurückhaltender bei der Bedienung, dabei aber fokussierter im Prozess der Notenerstellung und des Komponierens.

Zur genaueren Analyse wurde eine erste Untersuchung mit einer Probandengruppe von 19- bis 35-jährigen Studenten durchgeführt. Die Probandengruppe war gemischten Geschlechts und wies teilweise Erfahrungen im Bereich Musikproduktion und Musiksoftware auf. Alle User hatten Vorerfahrungen mit Smartphones. Zur Auswertung wurde das AttrakDiff™-Verfahren¹ angewandt. Dieses Verfahren stellt ein Instrument zur Messung der Attraktivität interaktiver Produkte dar und bietet eine standardisierte automatisierte Auswertung.

Insgesamt elf Probanden – die jeweils erst im Single-User-Modus und danach in Kleingruppen im Multi-User-Modus das Produkt zwischen zehn und 20 Minuten getestet haben – gaben mithilfe gegensätzlicher Adjektivpaare an, wie sie das Produkt wahrgenommen haben. Daraus wurden Ergebnisse für die vier Bereiche pragmatische Qualität (PQ – Benutzbarkeit eines Produktes), hedonische Qualität in Bezug auf Stimulation (HQ-S) und Identität (HQ-I) und Attraktivität (ATT) errechnet.

Die pragmatische Qualität (PQ) beschreibt die Benutzbarkeit eines Produktes, und verdeutlicht, wie gut der Nutzer seine Ziele erreichen kann. Die Dimension hedonische Qualität – Stimulation (HQ-S) bildet ab, inwieweit ein Produkt das menschliche Bedürfnis unterstützt, sich weiterzuentwickeln, indem es neuartige und fesselnde Funktionalitäten oder Inhalte bietet. Die hedonische Qualität – Identität (HQ-I) impliziert, wie weit sich der Nutzer mit dem Produkt identifizieren kann. Die Dimension Attraktivität (ATT)

¹ <http://attrakdiff.de> <2014-10-07>

beschreibt eine globale Bewertung des Produkts auf der Basis der wahrgenommenen Qualität (vgl. AttrakDiff 2014).

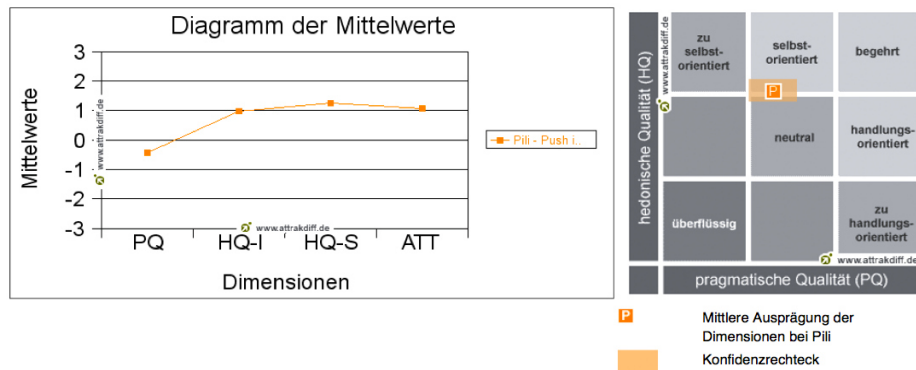


Abb. 6 Mittlere Ausprägung der vier Dimensionen des AttrakDiff beim Produkt *Pili* (Quelle: AttrakDiff)

Die Testauswertung (vgl. Abb. 6) ergab, dass „das Produkt auf die Nutzer insgesamt sehr attraktiv“ wirkt. Die Nutzer identifizieren sich überdurchschnittlich damit, sind gefesselt und werden davon motiviert, es wird auch als besonders kreativ bewertet.

Andererseits ergab die Auswertung, dass in den Bereichen Bedienung noch Verbesserungspotenzial besteht. Die Ergebnisse in den Wertepaaren „technisch – menschlich“, „kompliziert – einfach“, „verwirrend – direkt“, „umständlich – direkt“ fallen mit einer Wertung Richtung „technisch“, „kompliziert“, „verwirrend“ und „umständlich“ aus und sprechen dafür, dass noch Verbesserungspotenzial in der Interfacegestaltung oder in dessen Erklärung besteht.

5 Diskussion

Durch die Veröffentlichung im Google Play Store steht *Pili* nun über einer Milliarde Nutzern kostenlos zur Verfügung. Um eine leistungsfähige Android-Applikation für den Massenmarkt zu entwickeln, müssen die Skalierung von visuellen Inhalten, die unterschiedlichen Prozessorleistungen und das Ansprechverhalten der Sensorik und der Touchscreens beachtet werden.

Vor allem das langsame Ansprechverhalten einiger Touchscreens mit bis zu 200 ms Latenz ist für Musikapplikationen schwierig. Im Datenblatt eines Smartphones werden solche Daten nicht erwähnt. Der User kauft nach Dis-

playgröße, Arbeitsspeicher und Prozessorgeschwindigkeit. Von der Qualität der Sensorik, die das Ansprechverhalten ihres mobilen Endgeräts wohl bedeutend mitgestaltet, weiß der User beim Kauf nichts. *Pili* löst einige der entstandenen Probleme durch Quantisierung. Durch schnellere und zuverlässigere Sensordaten wäre jedoch eine Reihe neuer Interaktionsmöglichkeiten, auch für andere Anwendungen, möglich.

6 Zukünftige Weiterentwicklungen

Für die Weiterentwicklung des Projekts ist es wünschenswert, das User-Interface übersichtlicher zu gestalten und den Usern durch bessere Hilfefunktionen einen einfacheren Einstieg zu bieten. Zusätzlich könnte die Qualität und Quantität der verfügbaren Instrumente gesteigert werden. Hier bietet es sich auch an, die Klangqualität der Instrumente zu steigern und mehr Instrumente hinzuzufügen, die akustischen Instrumenten nachempfunden sind. Es könnte außerdem eine Schnittstelle definiert werden, wodurch User eigene Instrumente – die mit *Pure Data* erstellt wurden – einbinden und verwenden können.

Zudem sind Exportfunktionen geplant. Die Arrangements sollen in Waveform als .wav, .mp3 und auch als Midi-Dateien exportiert werden können. Weiterhin könnte die Bluetooth-Technologie in die Netzwerkschnittstelle integriert werden, um nicht auf WLAN angewiesen zu sein.

Denkbar sind auch Erweiterungen, die auf einer Metaebene *Pili*-Arrangements zur Verfügung stellen, beispielsweise auf einer Website. Hier könnten auch *Pili*-bezogene Updates, News, *physical playgrounds* etc. angegeben werden. Die Weiterentwicklungsmöglichkeiten sind aufgrund des offenen Systems sehr vielfältig und wir hoffen mit *Pili* ein musikalisches Objekt geschaffen zu haben, das neben den Autoren auch noch viele andere Personen inspirieren wird.

Literatur

Agawi (Hrsg.) (2013): TouchMarks I: Smartphone Touchscreen Latencies: <http://appglimpse.com/blog/touchmarks-i-smart-phone-touch-screen-latencies/> <03.07.2014>.

- AttrakDiff (2014): Untersuchungsbericht zum Produkt Pili – Push it loop it: http://config.attrakdiff.de/results/4517/0bd6d8555203a6920290e069d67e690d/pili__push_it_loop_it_ergebnisse.pdf <04.10.2014>.
- Brinkmann, P. (2012): *Making Musical Apps: Real-time audio synthesis on Android and iOS*. Sebastopol: O'Reilly Media.
- Essl, G. & Rohs, M. (2007): ShaMus – A Sensor-Based Integrated Mobile Phone Instrument. In: *Proceedings of the International Computer Music Conference (ICMC 2007)*, Copenhagen. S. 27–31.
- Geiger, G. (2006): Using the Touch Screen as a Controller for Portable Computer Music Instruments. In: *Proceedings of the 2006 International Conference on New Interfaces for Musical Expression (NIME06)*, Paris. S. 61–64
- Gopinath, S. & Stanyek, J. (2014): *The Oxford Handbook of Mobile Music Studies*. New York: Oxford University Press.
- International Data Corporation (2014): Press Release „Worldwide Smartphone Shipments Top One Billion Units for the First Time, According to IDC“: <http://www.idc.com/getdoc.jsp?containerId=prUS24645514> <20.07.2014>.
- Mauney, W. & Masterton, C. (2008): Small-Screen Interfaces. In: P. Kortum (Hrsg.): *HCI Beyond the GUI – Design for Haptic, Speech, Olfactory, and Other Nontraditional Interfaces*. San Francisco: Morgan Kaufmann, S. 343–351.
- Medienpädagogischer Forschungsverbund Südwest (2013): *JIM-Studie 2013 : Jugend, Information, (Multi-) Media*: <http://www.mpfs.de/fileadmin/JIM-pdf13/JIMStudie2013.pdf> <12.07.2014>.
- Sikora, F. (2003): *Neue Jazz-Harmonielehre*. Weihergarten: Schott.
- Tanaka, A. (2004): Mobile Music Making. In: *Proceedings of the International Conference on New Interfaces for Musical Expression (NIME 2004)*, Hamamatsu. S. 154–156.