



Jonas Böhm

Waterside

Die Entwicklung eines Audio Games

**Bachelorarbeit der Studienrichtung Elektrotechnik-Toningenieur
zum Seminar Computermusik und Medienkunst**

Supervisor: Johannes Zmöllnig, Dipl.-Ing.
Institut für Elektronische Musik und Akustik
Universität für Musik und darstellende Kunst Graz

Graz, November 2018

Abstract

Diese Bachelorarbeit ist im Rahmen des Seminars „Computermusik und Medienkunst“ zum Thema „Audio Games“ entstanden. Im Unterschied zu traditionell eher visuell orientierten Computerspielen spielt in Audio Games der Hörsinn eine zentrale Rolle. Im Zuge des Seminars wurde von jedem teilnehmenden Studierenden ein eigenes Audio Game konzipiert und als Prototyp implementiert.

In dieser Arbeit werden zunächst Spiele im Allgemeinen – bis hin zu Audio Games im Speziellen – aus unterschiedlichen Perspektiven beleuchtet. Kern dieser Arbeit ist die Dokumentation des Designs und der Implementation eines Jump’n’Run Gehörbildungs-Lernspiels. Im Spiel – *Waterside*¹ – werden auditiv vermittelte Aufgabenstellungen innerhalb einer zweidimensionalen Spielwelt durch „physikalische“ Interaktion mit visualisierten Lösungsmöglichkeiten beantwortet. Die zu lösenden Aufgaben sind vom Seashore Test inspiriert und sollen eine spielerische Vorbereitung für diesen ermöglichen.

¹<https://git.iem.at/lehre/CMMM/LU18/waterside>

Inhaltsverzeichnis

Abstract	ii
1 Hintergrund und Motivation	1
1.1 Was ist ein Spiel?	1
1.2 Sinnvolles Spielen	2
1.3 Game Design	3
1.4 Digitale Spiele	3
1.5 Audio Games	4
1.6 Waterside - Der Seashore Test als Lernspiel	6
2 Design	9
2.1 Idee	9
2.2 Spielwelt	10
2.3 Spielfigur	10
2.4 Aufgaben	11
2.5 Ziel	12
2.6 Waterside als kybernetisches System	13
2.7 Spielregeln	14
3 Implementation	17
3.1 Plattform	17
3.2 Processing als Entwicklungsumgebung	17
3.3 Spielmenü	18
3.4 Bewegung	21
3.5 Aufgaben zur Gehörbildung	23
3.6 Sound Effekte	27

Inhaltsverzeichnis

3.7 Statistik	28
4 Konklusion	30
4.1 Rekapitulation	30
4.2 Ausblick	31
Literatur	33

1 Hintergrund und Motivation

1.1 Was ist ein Spiel?

Das Wort *Spiel* ist ein vielfältig definierbarer Begriff. Salen und Zimmerman teilen das Wort *Game* in die Aspekte *Rules*, *Play* und *Culture* ein, und fassen Definitionen aus verschiedenen Perspektiven zu einem eigenen Ergebnis zusammen:

A game is a system in which players engage in an artificial conflict, defined by rules, that results in a quantifiable outcome. [17]

- Jedes Spiel kann als ein durch Regeln definiertes *System* verstanden werden, welches seine Komponenten, deren Eigenschaften und die Zusammenhänge der Komponenten untereinander beschreibt. Auch der kulturelle Kontext und die individuelle Spielerfahrung ist Teil des Spielsystems.
- Ein oder mehrere *Spieler/Spielerinnen* interagieren mit dem Spielsystem, das Spiel wird gespielt.
- Jedes Spiel beinhaltet eine Form des *künstlichen Konflikts*. Damit besteht immer ein Wettbewerb mit dem Spielsystem oder den Mitspielenden.
- *Regeln* definieren Spielsysteme und legen daher auch den Zusammenhang zwischen Aktionen der Spielenden und den resultierenden Systemantworten fest. Es wird zwischen operativen, konstituierenden und impliziten Regeln unterschieden, wobei den Spielenden nur operative Regeln offen als „Spielregeln“ präsentiert werden können. Konstituierende Regeln ergeben sich hingegen aus

der formalen Struktur des Systems, beschreiben also die systeminterne Funktionsweise und sind für die Spielenden daher nicht zwangsläufig offensichtlich. Implizite Regeln können schließlich als „ungeschriebene Gesetze“ – zum Beispiel aufgrund ihrer Selbstverständlichkeit – zusammengefasst werden.

- Zum Ende jeden Spiels wird den Spielenden ein *Ergebnis* vermittelt. Ist keine Siegesbedingung vorhanden, wird das Ergebnis zum Beispiel in Form eines erreichten Punktestands *quantifiziert*.

1.2 Sinnvolles Spielen

Ein intuitiv sehr wichtiger Punkt bleibt in der bisherigen Definition unerwähnt: Der Spaß am Spiel. So bedeutet *Play* für Huizinga eine vorübergehende, freiwillige, bewusste, von einem Gefühl der Spannung und Freude begleitete, sich von der realen Welt abgrenzende und bindenden Regeln folgende Aktivität. Laut ihm ist *Play* einerseits eine völlig einnehmende, andererseits jedoch auch „nicht ernste“ Handlung [7].

Doch in welchem Zusammenhang stehen nun die Wörter *Spiel* und *spielen*? In der englischen Sprache lassen sich hierzu die Wörter *Game* und *Play* gegenüberstellen. Nach obiger Definition stellt *Play* neben *Rules* und *Culture* nur einen Teil des komplexen Systems *Game* dar, *Spiele* können *gespielt* werden. Andererseits könnte *Game* auch als Teil von *Play* betrachtet werden, wenn das *Spielen* keinen formalen Regeln folgt. Um allen Bedeutungen des Wortes *Play* gerecht zu werden, haben Salen und Zimmerman folgende einfache Definition erarbeitet:

Play is free movement within a more rigid structure. [17]

Das *Spielen* eines *Spiels* kann als die Interaktion zwischen Spielenden und Spielsystem, also der Beziehung zwischen Handlung und Auswirkung verstanden werden. *Sinnvolles Spielen* entsteht, wenn die Beziehung zwischen Handlungen und Auswirkungen sowohl erkennbar als auch in den größeren Kontext des Spiels integriert ist.

1.3 Game Design

The goal of successful game design is the creation of meaningful play. [17]

Um *Sinnvolles Spielen* durch *Game Design* zu ermöglichen, muss also ein geeigneter Handlungsrahmen entworfen werden, welcher den Spielenden erlaubt Entscheidungen zu treffen, welche sich systemseitig auf den weiteren Spielverlauf auswirken. Eine von dem/der Spielenden gesetzte Handlung soll sowohl kurz-, als auch langfristig neue Bedeutungen innerhalb des Spielsystems erzeugen. Bedeutung kann jedoch nur zustande kommen, wenn die Systemauswirkung von dem/der Spielenden auch als solche erkannt wird. Eine sinnhafte Interaktion zwischen Spielenden und System setzt also ein erkennbares Feedback voraus.

Ein gut entworfenes Spielsystem ist auf verschiedenen Ebenen interaktiv, so soll den Spielenden nicht nur die Teilnahme durch direkt entworfene Wahlmöglichkeiten gegeben sein, wichtiger ist, *wie* die Interaktion von den Spielenden empfunden und gedeutet wird. Darüber hinaus lässt sich auch der kulturelle Kontext des gespielten Spiels als Form der Interaktivität betrachten.

Design kann als das Schaffen eines Kontextes verstanden werden, welcher durch die Interpretation durch seine Teilnehmer Bedeutung erhält [17]. Auch aus etymologischer Sicht weist der Begriff *Design* (lateinisch: designare – deutsch: bezeichnen) auf die Herstellung von Sinn und Bedeutung hin. Als *Game Design* lässt sich also das Erschaffen interaktiver und komplexer Systeme durch Herstellung einer interpretierbaren Bedeutung verstehen. Erfolgreiches *Game Design* beschränkt sich daher nicht auf den direkten Entwurf formaler Regeln, sondern setzt auch das indirekte Design der Spielerfahrung voraus.

1.4 Digitale Spiele

Digitale Spiele bilden neben analogen elektronischen Spielen eine Unterkategorie der *elektronischen Spiele*. Zusätzlich zu den bisher beschriebenen Merkmalen eines

1 Hintergrund und Motivation

Spiele weisen *digitale Spiele* weitere besondere Eigenschaften auf.

Die automatisierte Anwendung von fest implementierten Spielregeln digitaler Spiele kann Spielenden die Notwendigkeit ersparen, komplizierte Regeln selbst anwenden zu müssen. Damit einhergehend ist es den Spielenden nicht möglich, spontan die Regeln zu brechen.

Da die softwareseitige Datenverarbeitung den Spielenden nicht direkt präsentiert wird, lässt sich die Zugänglichkeit bestimmter spielrelevanter Informationen gezielt steuern.

Systemseitiges Feedback kann durch kurze Rechenzeiten heutiger Computer in Echtzeit vermittelt werden. Die Interaktion mit digitalen Spielen wird jedoch grundsätzlich durch limitierte Steuerungsmöglichkeiten beziehungsweise limitierte Input und Output Hardware eingeschränkt.

Die Automatisierung eines komplexen Spielsystems und der limitierte Handlungsrahmen der Spielenden tragen maßgeblich dazu bei, dass die Regeln digitaler Spielsysteme oft *spielerisch* erlernt werden können, und so einen intuitiven Zugang ermöglichen.

1.5 Audio Games

Definitionen

Während *Video Games* meist einen Bildschirm als visuelles Ausgabemedium verwenden, spielt bei *Audio Games* der Klang eine zentrale Rolle. Die beiden Kategorisierungen schließen einander jedoch nicht aus. Bezogen auf *Video Games* – verallgemeinert bezeichnet als *digitale Spiele* – können Audio Games als Unterkategorie dieser verstanden werden. Die Klassifizierung eines Spiels als *Audio Game* – oder einer Unterkategorie dessen [16] – kann jedoch auch parallel zu üblichen *Video Game* Klassifizierungen wie „Platform“ oder „Strategy“ erfolgen.

Friberg und Gärdenfors beziehen sich in ihrer Definition eines *Audio Games* auf *Accessible Games*, welche auch ohne visuelles Interface spielbar sein sollen [5]. Spiele,

1 Hintergrund und Motivation

deren auditives Interface einen unentbehrlichen Teil der Spielmechanik darstellt, werden von Rovithis als *Audio Based Games* bezeichnet, der Fokus liegt bei diesen Spielen also nicht notwendigerweise auf der Barrierefreiheit [16]. Nach Definition des Seminars und in weiterer Folge dieser Arbeit werden ebendiese Spiele als *Audio Games* zusammengefasst. Das hier entwickelte Spiel – *Waterside* – erfordert ein sowohl visuelles als auch auditives Interface.

Waterside lässt sich aus Sicht der klassischen *Videospiele* als *Platform Game* (siehe auch Kapitel 1.6) und aus Sicht der *Audio Games* als *Puzzle* [16]. bezeichnen, wobei gleichzeitig auch das Prinzip eines *Lernspiels* im Vordergrund steht.

Geschichte und heutige Perspektive

Das erste kommerzielle Audio Game, *Touch Me*, wurde 1974 von Atari veröffentlicht [16]. Verpackt in einem Arcade-Gehäuse können Spielende damit ihr Gedächtnis trainieren, indem visuell und akustisch vermittelte Klangsequenzen nachgespielt werden.

Das Potential auditiver Interfaces wurde bereits sehr früh für die Entwicklung von Spielen genutzt, die auch von sehbehinderten Personen spielbar sein sollten. Dazu wurde zunächst die visuelle Repräsentation bereits existenter, jedoch nicht intrinsisch visueller Spielprinzipien akustisch dargestellt. Auch heute bilden solche „nur übersetzten“ Spielprinzipien noch die Grundlage für einen Großteil der bisher entwickelten Audio Games [13].

Mit steigender Anzahl interessierter Entwickelnder und fortschreitenden Technologien werden heute auch zunehmend eigenständigere Spielkonzepte entworfen. Beispielsweise bietet moderne 3D-Audiotechnik zahlreiche Grundlagen zum Entwurf innovativer Spielsysteme [5][9][16]. Auch die technischen Fortschritte in Analyse und Synthese von Sprache tragen zur ständigen Weiterentwicklung audiobasierter Interfaces bei [13][16]. Die Psychoakustik bietet bezüglich des Audio Game Designs einige interessante Ansätze, welche eine erhöhte Aufmerksamkeit durch Spieleentwickelnde verdienen. Beispielsweise untersucht Michel Chion drei Arten des Hörens [3], welche durch auditive Interfaces auf verschiedene Weise gezielt

angesprochen werden können [5][11]. All diese Gegebenheiten lassen zahlreiche Innovationen auf dem Feld der Audio Games erhoffen.

Die vielfältigen Möglichkeiten Klang als zentrales Element der Spielmechanik einzusetzen werden bis heute jedoch nur spärlich genutzt. Ein Grund warum Klang in nur relativ wenigen Mainstream Spielen eine bedeutende Rolle einnimmt, ist die begrenzte Nachfrage seitens Spielender beziehungsweise das daraus resultierend begrenzte Budget der Spielentwickelnden [16]. Zudem scheint es sehenden Personen aufgrund ihrer stark visuell orientierten Natur schwer zu fallen, sich auf die Beschränkung gegebener Sinne einzulassen, um innovative Ideen überhaupt erst zu ermöglichen. Als Maß für innovatives Design eines *Audio Games* könnte also die Erweckung größeren Interesses uneingeschränkter Spielender verstanden werden [13].

1.6 Waterside - Der Seashore Test als Lernspiel

Platform Games

Als *Platform Games* werden Computerspiele bezeichnet, in denen eine Spielfigur durch Laufen und Springen in der Spielwelt bewegt wird. Das Sichtfeld des/der Spielenden ist dabei typischerweise kleiner als die vollständige Spielwelt, wodurch das Sichtfeld im Laufe des Spiels verschoben wird, um mehr von der Spielwelt erkundbar zu machen [8].

Im deutschen Sprachgebrauch wird oft synonym die Bezeichnung *Jump'n'Run* verwendet. Dieser Begriff könnte auch als Unterkategorie der *Platform Games* verstanden werden, falls die Sprungbewegung als primäre Bewegungs- oder Interaktionsform vorausgesetzt wird. Im Falle des hier entwickelten Spiels treffen beide Begrifflichkeiten jedoch klar auf das Ergebnis zu: Sprungbewegungen sollen ein essentieller Teil des Spiels sein.

Seashore Test

Der Ursprung des heute zur Zulassungsprüfung verschiedener Studienrichtungen am *IEM* verwendeten Seashore Tests wurde erstmals 1915 von Carl Emil Seashore als „The Measurement of Musical Talent“ [18] veröffentlicht. Seine psychoakustisch orientierten Veröffentlichungen behandelten unter anderem die Erforschung der Grenzen menschlicher auditiver Wahrnehmung sowie die Beurteilung musikalischen Talents individueller Einzelpersonen. Der Seashore Test versucht musikalische Fähigkeiten auf einzelne messbare Größen zu reduzieren. Beim Testen einzelner Größen werden im Sinne der experimentellen Psychologie also alle anderen Klangeigenschaften bestmöglich konstant und reproduzierbar gehalten [18][19][20].

Ziel dieser Arbeit ist es, ein Lernspiel zu schaffen, welches neben dem Zweck der *spielerischen* Gehörbildung auch eine Vorbereitungsmethode für den am *IEM* verwendeten Seashore Test darstellt.

EarMaster

Hinsichtlich des Zwecks der reinen Gehörbildung wurden im Laufe der letzten Jahrzehnte bereits zahlreiche Arten von Gehörbildungs-Software veröffentlicht. So steht beispielsweise bei *EarMaster* die benutzerfreundliche, effiziente und flexible Aneignung und Optimierung von musikalischen Fähigkeiten durch verschiedenartige Interfaces im Vordergrund. Benutzer können sich in verschiedenen Disziplinen fortbilden, und ihren Lernfortschritt anhand quantifizierbarer Ergebnisse überprüfen. Trotzdem ist *EarMaster* nicht als echtes Spiel zu verstehen. In Standardkursen sind zwar Siegesbedingungen vorhanden, jedoch kann dabei ein großer Teil der „Spielregeln“ selbst festgelegt werden. Huizinga würde außerdem anmerken, dass die „Spielenden“ nicht mit Spannung und Freude erfüllt von ihrer Beschäftigung absorbiert werden.

Waterside

Die Entwicklung von *Waterside* ist ein Versuch, ebendiese Lücke zu schließen. Es soll ein Spielsystem geschaffen werden, welches ein *spielerisches Trainieren* des

1 Hintergrund und Motivation

Gehörs ermöglicht. Huizinga schreibt, dass dem Spielen eine wichtige Bedeutung für die psychische und körperliche *Entwicklung* des Menschen zukommt [7]. Miller glaubt sogar, dass die ersten Spiele der Menschheitsgeschichte auf das *Erlernen* überlebensnotwendiger Fähigkeiten zurückgehen, und erklärt bei der Auseinandersetzung mit dem Begriff *Edutainment*, wie interaktive *Lernprojekte* effektiver funktionieren können, wenn diese als Spiele dargeboten werden [10].

Die in *Waterside* spielbaren Gehörbildungs Disziplinen beschränken sich auf die sechs einfachen Kategorien der am *IEM* verwendeten Version des Seashore Tests: Tonhöhe, Lautstärke, Tonlänge, Klangfarbe, Tongedächtnis und Rhythmus. Diese werden in Form von Single Choice Aufgaben vermittelt. Es wird erwartet, dass größtenteils Bewerber einer Zulassungsprüfung bestimmter Studienrichtungen am *IEM* Interesse an *Waterside* zeigen werden, womit auch der resultierende kulturelle Kontext in deren Umfeld verortet sein wird.

2 Design

2.1 Idee

Waterside wird als vertikal verlaufendes, zweidimensionales Platform Game für einen/eine Spieler/Spielerin entworfen. Die Spielfigur soll nach oben durch die Spielwelt bewegt werden. Die akustisch vermittelten gehörbildenden Aufgabenstellungen sind durch „physikalische“ Interaktion mit den visualisierten Antwortmöglichkeiten lösbar.

Das Lösen der Aufgabenstellungen wird dem/der deutenden Spielenden als das Fangen von Schmetterlingen präsentiert. So sieht der/die Spieler/Spielerin sich zu jeder Aufgabe mit mehreren Fluginsekten konfrontiert, und kann nach Gehör eines der Insekten als Schmetterling identifizieren und „einfangen“.

Die grundlegende Idee in *Waterside* ist das spielerische Trainieren des Gehörs. Obwohl das auditive Training hintergründig eine große Motivation für dieses Projekt darstellt, soll am Ende der Spielspaß im Vordergrund stehen. Im Idealfall sind die Spielenden also in der Lage, den Zweck des Lernens völlig auszublenden, und sich nur aus Lust am Fangen von Schmetterlingen mit *Waterside* zu beschäftigen.

Der Name *Waterside* ergibt sich einerseits aus der Anlehnung an den nach seinem Entwickler benannten *Seashore* Test, andererseits durch die drohende Gefahr im unteren Teil des Sichtfelds: ein ständig steigender Wasserpegel, welcher die Spielfigur nach oben zwingt.

2.2 Spielwelt

Hinsichtlich des Bewegungsraums der Spielfigur präsentiert sich die Spielwelt durch Plattformen, auf welchen die Spielfigur landen kann. Die Plattformen bewegen sich dabei von oben nach unten durch den sichtbaren Teil der Spielwelt. Aus der Perspektive des/der Spielenden scheint das Sichtfeld also mit dem Wasserpegel nach oben zu wandern. Das Spiel endet, sobald die Spielfigur aus dem Sichtfeld heraus ins Wasser fällt. Die Geschwindigkeit der Erhöhung des Wasserstandes stellt daher als Spielgeschwindigkeit den Schwierigkeitsgrad der Plattform Game Komponente von *Waterside* dar und wird indirekt durch das Antwortverhalten des/der Spielenden beeinflusst (siehe auch Kapitel [2.6](#)).

2.3 Spielfigur

Die Bewegung der Spielfigur ist stark von der klassischen Mechanik inspiriert. Als Teilgebiet der Physik beschreibt diese die Bewegung von Körpern unter dem Einfluss von Kräften. Die Effekte jener Beschreibungen sind durch real getätigte oder beobachtete Bewegungen von jedem Menschen alltäglich erfahrbar. So vermittelt beispielsweise das Werfen oder Rollen von Murmeln ein intuitives Verständnis über Phänomene wie Masseträgheit, Gravitation und Reibung, selbst wenn die zugrundeliegenden physikalischen Gesetzmäßigkeiten dadurch nicht zwangsläufig wissenschaftlich erklärbar werden. Durch Anpassung des Systemverhaltens auf die Erwartungshaltung der Spielenden wird also eine intuitive Steuerung ermöglicht, welche zur Interaktivität des Spielsystems beiträgt.

Die Spielfigur wird durch Bewegungen nach links und rechts, sowie durch Sprünge in der Spielwelt fortbewegt. Eine Sprungbewegung soll nur möglich sein, wenn sich die Spielfigur auf einer Plattform oder an einer der Seitenwände befindet, wenn also eine feste Oberfläche zum Absprung vorhanden ist. Die Steuerbefehle links und rechts sollen die Spielfigur seitwärts beschleunigen, solange die entsprechenden Bedienelemente gedrückt gehalten werden. Während die Spielfigur sich nicht auf einer Plattform befindet, sich also in der freien Luft bewegt, soll horizontal nur

leichter beschleunigt werden können. Die kreisförmig dargestellte Spielfigur erreicht auf den Plattformen „rollend“ also ihre höchste Agilität.

2.4 Aufgaben

Während des Spielverlaufs werden nacheinander zufällig ausgewählte Gehörbildungsaufgaben akustisch vermittelt. Jede Frage besitzt nur eine einzige korrekte Antwort, diese soll von dem/der Spielenden aus den angezeigten Antwortmöglichkeiten ausgewählt und angesprungen werden. Die als herumschwebende Fluginsekten dargestellten Antworten tragen einfache Symbole auf ihren Flügelflächen, welche auf zugehörige Aufgabentypen hinweisen. Der/Die Spielende entscheidet also mithilfe des Gehörs und der dargebotenen Antwortmöglichkeiten, hinter welchem Fluginsekt sich der Schmetterling verbirgt. Die Antwortsymbole sollen von den Spielenden intuitiv interpretierbar sein. dazu werden sehr einfache und selbsterklärende Zeichen verwendet. Trotzdem ist es notwendig, dem/der Spielenden noch vor der auditiv erfassbaren Aufgabenstellung zu vermitteln, um welche Disziplin es sich handeln wird. Der/Die Spielende soll in der Lage sein, sich auf die geforderten Eigenschaften des Klangbeispiels vorzubereiten. Zu diesem Zweck erscheinen die möglichen Antworten in leicht transparenter Form noch bevor die zugrundeliegenden Klänge abgespielt werden. Erst zum Ende des Klangbeispiels werden die Fluginsekten vollständig sichtbar – und damit fangbar – vorher ist also kein versehentliches Lösen der Aufgabenstellung möglich.

Alle sechs Aufgabentypen (siehe auch Kapitel 3.5) bestehen wie beim Seashore Test am *IEM* jeweils aus zwei Klängen beziehungsweise zwei Sequenzen. Bei den Aufgaben zu Tonhöhe, Lautstärke und Tonlänge bezieht sich die Antwort auf den zweiten Einzelklang. Hier soll beurteilt werden, ob dieser im Vergleich zum ersten Einzelklang tiefer/höher, leiser/lauter beziehungsweise kürzer/länger ist. Zu den Disziplinen Klangfarbe und Rhythmus wird nur die Gleichheit der beiden Klänge beziehungsweise Sequenzen als ungleich/gleich ausgewählt. Bei den Aufgaben zum Tongedächtnis verändert sich von der ersten zur zweiten Sequenz eine einzelne

Tonhöhe. Der/Die Spielende muss entscheiden, der wievielte Ton der Sequenz sich geändert hat.

Die Antwortmöglichkeiten jeder Disziplin schweben jeweils in gleicher Reihenfolge im Sichtfeld umher. Damit kann der/die Spielende nach kurzer Eingewöhnungszeit nicht nur das Antwortsymbol, sondern auch intuitiv die Richtung der Antwort dem Klangbeispiel zuordnen.

Das Klang- beziehungsweise Sequenzpaar jeder Aufgabe kann genau einmal wiederholt werden. Die Wiederholbarkeit wird nach Abspielen der Frage und einigen Sekunden Verzögerung durch farblich hervorgehobene Seitenwände gekennzeichnet. Wenn die Spielfigur nun einen „Walljump“ tätigt, sich also mit einem Sprung von einer Seitenwand abstößt, wird die aktuelle Aufgabe akustisch wiederholt. Die zugehörigen Antworten bleiben dabei unverändert.

Im Sinne eines interaktiven Spielsystems soll dem/der Spielenden deutlich mitgeteilt werden, welches Insekt gefangen wurde, ob die Aufgabenstellung also richtig gelöst wurde. Dazu färbt sich das gefangene Fluginsekt entweder grün oder rot und bleibt noch eine weitere kurze Zeit bestehen bevor es endgültig verschwindet. Zusätzlich löst die gegebene Antwort – je nachdem ob richtig oder falsch – einen von zwei verschiedenen Sound Effekten (siehe Kapitel 3.6) aus. Falsches Antworten hat außerdem eine Kollision mit dem entsprechenden Fluginsekt zur Folge, welche die Spielfigur zurückstößt.

2.5 Ziel

Das Spiel endet, wenn die Spielfigur aus der sichtbaren Spielwelt heraus ins Wasser fällt, es also nicht rechtzeitig schafft, sich auf eine Plattform zu retten und den Weg nach oben zu finden. Das Ende des Spiel soll unmissverständlich mit einem rötlich transparenten Bildschirm, einem „Game Over“ Sound Effekt, und der Möglichkeit des Neustarts gekennzeichnet werden.

Das Ergebnis des Spiels ist die Anzahl gefangener Schmetterlinge. Der Punktestand wird während des Spiels in der oberen Menüleiste angezeigt, zum Ende des Spiels wird das Ergebnis zusätzlich in der Bildschirmmitte verkündet.

2.6 Waterside als kybernetisches System

Das Ende des Spiels wird zwangsläufig durch das Feedbacksystem herbeigeführt. Der/Die Spielende soll einerseits durch auditive Fähigkeiten, andererseits auch durch geschicktes Manövrieren der Spielfigur in der Lage sein das Spiel zu verlängern und dementsprechend das Fangen vieler Schmetterlinge zu ermöglichen. Es ergeben sich zwei verschiedene Schwierigkeitsgrade, da die Antworten sowohl auditiv *erkannt* als auch „physikalisch“ *beantwortet* werden müssen.

Auditiver Schwierigkeitsgrad

Die Schwierigkeit der auditiven Komponente des Spiels erhöht sich wie beim Seashore Test mit fortschreitender Dauer. Dazu wird die Anzahl der in einem Spiel gefangenen Schmetterlinge auf den auditiven Schwierigkeitsgrad abgebildet (siehe auch Kapitel 3.5). Die maximale Schwierigkeit der Aufgaben wird unabhängig vom Aufgabentyp nach 30 richtigen Antworten erreicht. Das Erhöhen des auditiven Schwierigkeitsgrads wird dem/der Spielenden nicht als Spielregel präsentiert, ist für Teilnehmer des Seashore Tests allerdings selbstverständlich.

Spielgeschwindigkeit

Jede falsche Antwort hat eine Steigerung der Spielgeschwindigkeit zur Folge. Die erhöhte Geschwindigkeit erschwert ein *Beantworten* der Aufgaben, selbst wenn eine Antwort durch den Einsatz auditiver Fähigkeiten richtig *erkannt* wurde. Dieses Feedback kann jedoch teilweise durch das richtige Lösen mehrerer Aufgaben in Folge, beziehungsweise der daraus resultierenden temporär erhöhten Sprungkraft, ausgeglichen werden. Dieser Bonus wird durch einen leuchtenden Umriss der Spielfigur symbolisiert.

Trotz der erzeugten Spannung und der Möglichkeit des Ausgleichs tragen beide dieser positiven Feedbackelemente zur Instabilität des Spiels bei; richtiges Antworten erleichtert das *Beantworten* weiterer Aufgaben, falsches Antworten erschwert das *Beantworten* weiterer Aufgaben. Der Konflikt zwischen dem/der Spielenden und dem Spielsystem kann daher schnell aus der Balance geraten.

Zur Stabilisierung wird das Feedbacksystem um ein negatives Feedbackelement erweitert. Droht die Spielfigur ins Wasser zu fallen, wird die Spielgeschwindigkeit verringert. Hält sich die Spielfigur hingegen in Antwortnähe im oberen Teil des Sichtfelds auf, wird die Spielgeschwindigkeit temporär erhöht (siehe auch Kapitel 3.4).

Die effektive Spielgeschwindigkeit ergibt sich durch Multiplikation der destabilisierenden Geschwindigkeit mit dem aus der Position der Spielfigur ermittelten Grad der Stabilisierung.

2.7 Spielregeln

Der Startbildschirm von *Waterside* (Abbildung 2.1) soll dem/der Spielenden die vier wichtigsten Spielregeln präsentieren.

- Fange soviele Schmetterlinge wie möglich, ohne ins Wasser zu fallen.
- Wiederhole einen Hinweis auf einen gesuchten Schmetterling durch einen „Walljump“.
- Das Fangen anderer Insekten lässt den Wasserspiegel schneller steigen.
- Solange mindestens drei Schmetterlinge in Folge gefangen wurden, sind höhere Sprünge möglich.

Die Spielregeln sind bewusst einfach formuliert. Es ließen sich natürlich viele weitere operative Regeln als Spielregeln darstellen, viele davon sind allerdings auch auf spielerische Weise schnell erlernbar.

2 Design

Die möglichen Steuerungseingaben des Spiels sind für die Verwendung mit Maus und Tastatur konzipiert und werden ebenfalls im Startbildschirm gezeigt. Diese können nach der Definition aus Kapitel 1.1 zwar ebenfalls als Regeln verstanden werden, allerdings spielt der deskriptive Zusammenhang zwischen Regel und Steuerungsmöglichkeit aus Spielendenperspektive keine bedeutende Rolle. Zur Übersichtlichkeit und Verständlichkeit werden die beiden Kategorien daher in verschiedenen Textblocks erklärt.

Zuletzt wird im Startmenü der Zusammenhang zwischen Höreindrücken und Antwortsymbolen gezeigt. Es wird nicht weiter erklärend auf die gehörbildenden Aufgabenstellungen eingegangen, da zu erwarten ist, dass die Spielenden bereits über die Funktionsweise des Seashore Tests informiert sind. Die visuelle Darstellung der Aufgabentypen Klangfarbe und Rhythmus durch identische Antwortsymbole wird aufgrund der gut unterscheidbaren Klangeigenschaften als eher unproblematisch angesehen.

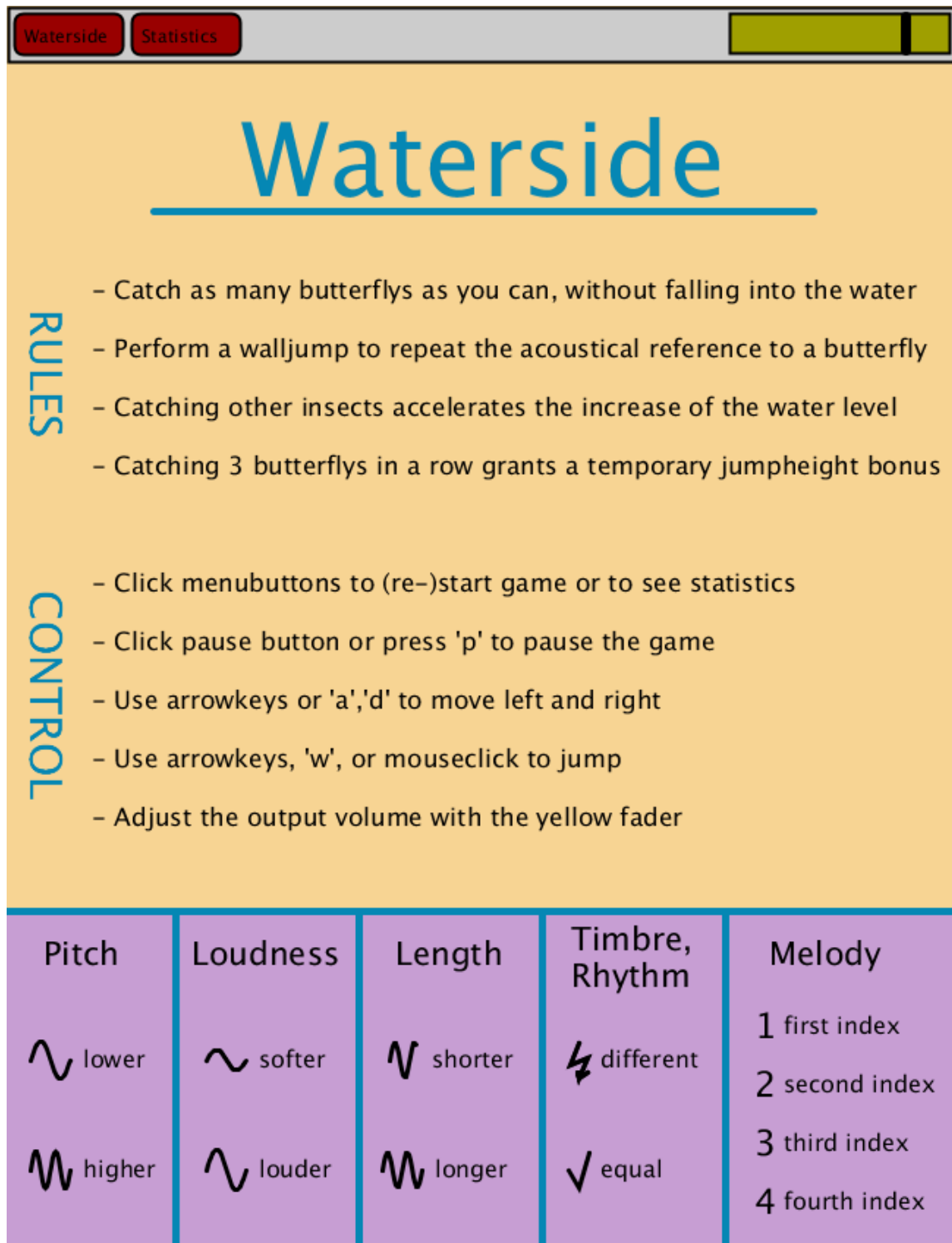


Abbildung 2.1: Waterside Startbildschirm

3 Implementation

3.1 Plattform

Waterside setzt als digitales Audio Game gewisse Ein- und Ausgabehardware zur Interaktion voraus. Inputseitig soll der/die Spielende die Spielfigur steuern können, outputseitig muss das System sowohl durch auditive, als auch durch visuelle Informationen reagieren. Aufgrund der flexibel kombinierbaren Hardware und den zum Entwicklungsbeginn schwer abschätzbaren technischen Anforderungen wird der Prototyp von *Waterside* als Offline-Spiel zunächst für *Windows* entwickelt, später auch für die Desktop-Betriebssysteme *macOS* und *Linux* veröffentlicht.

3.2 Processing als Entwicklungsumgebung

Processing wurde am Massachusetts Institute of Technology als textbasierte Programmiersprache mit eigener integrierter Entwicklungsumgebung (IDE) geschaffen. Ziel des Projekts war das Erreichen eines hohen Maßes an Benutzerfreundlichkeit, ohne auf fortschrittliche Features verzichten zu müssen [15]. Neben dem Hauptanwendungsbereich der Bilddarstellung ist zum Beispiel das Verarbeiten von Maus- und Tastatureingaben, sowie die Analyse und Synthese von Audiodaten in Echtzeit möglich.

Die IDE bietet neben Texteditor, Compiler, Ausgabekonsole und einem Darstellungsfenster für den grafischen Output viele weitere praktische Features. Die Syntax der Sprache ist stark an Java angelehnt. Durch Objektorientierung und die Möglichkeit

3 Implementation

der Einbindung mehrerer Files durch die IDE sind auch größere Projekte realisierbar. Mittlerweile ist die IDE sogar mit JavaScript, Python und Android kompatibel. Durch die Anordnung der Pixel auf einem Koordinatensystem können Programmierende sich intuitiv im Darstellungsfenster zurechtfinden. *Processing* Codes bestehen im Kern aus zwei elementaren Funktionen: In der „setup“ Funktion werden zu Anfang des Programmaufrufs einmalig Variablen initialisiert, dannach wird der Code innerhalb der „draw“ Funktion bis zum Beenden des Programms als Dauerschleife ausgeführt.

Für die Entwicklung von *Waterside* wurde die ursprüngliche, auf Java basierende Sprachversion verwendet. Die IDE ermöglicht den direkten Export von Projekten als ausführbare Java Applikationen für verschiedene Betriebssysteme. *Waterside* wird als freie Software, wie auch *Processing*, unter der *GNU General Public License Version 3* veröffentlicht. Diese Lizenz gewährt auf *Waterside* aufbauenden Projekten nahezu alle denkbaren Freiheiten, solange die selbe quelloffene Lizenz weiterverwendet wird.

3.3 Spielmenü

Jede Ansicht von *Waterside* zeigt eine Menüleiste am oberen Bildschirmrand. Hier wird über die Menü-Schalter entweder ein neues Spiel gestartet (Abbildung 3.1) oder der Statistikbildschirm (siehe auch Kapitel 3.7 und Abbildung 3.2) aufgerufen.

Im Spielbildschirm wird zusätzlich ein Schalter zum Pausieren des Spiels eingeblendet. Zudem werden hier die wichtigsten Daten des aktuellen Spiels angezeigt; die Anzahl der richtigen und falschen Antworten, sowie die Anzahl der aktuell in Folge richtig beantworteten Aufgaben.

Durch einen Schieberegler in der Menüleiste ist die Lautstärke der Aufgabenstellungen durch den/die Spieler/Spielerin frei anpassbar. Inspiriert von Puckette [14] wird dieser Lautstärkeregler mit linear skalierten Dezibelschritten implementiert. Der Regler bildet eine Reichweite von -100 dB bis 0 dB auf Amplitudenwerte zwischen

3 Implementation

$1 \cdot 10^{-5}$ und 1 ab. Intern kann bei der Initialisierung des Schieberegler Objekts ein beliebiges Minimum und Maximum in Dezibel gewählt werden. Jedes gewählte Maximum bezieht sich aufgrund der internen Weiterverarbeitung auf eine ausgegebene Maximalamplitude von 1. Der kleinstmögliche ausgegebene Amplitudenwert ergibt sich durch die indirekt eingestellte dynamische Reichweite im Bezug auf die fixierte Maximalamplitude. Ob eine Dynamik von 100 dB hardwareseitig erreicht und auditiv wahrgenommen wird, ist jedoch zu bezweifeln.

Die objektorientierte Programmierung der interaktiven GUI-Elemente ermöglicht eine komfortable Handhabung und eine hohe Arbeitersparnis bei der Entwicklung. So werden zum Beispiel die Klassen aller Schalter Objekte – Menü-Schalter, Pause-Schalter, Neustart-Schalter, Schalter zum Zurücksetzen der Statistik, und für die Entwicklung hilfreiche Debugging-Schalter – von der selben Basisklasse abgeleitet. Jedes dieser Schalter-Objekte enthält sowohl einen Zustand, als auch eine Information zur Änderung dessen. Auf diese Weise kann ein Schalter zum Umschaltzeitpunkt auch einmalige Ereignisse auslösen.

Jedes GUI-Element reagiert interaktiv auf die Präsenz des durch die Maus gesteuerten Cursors, indem dessen aktuelle Grundfarbe leuchtend hervorgehoben wird. Der Schieberegler zeigt in diesem Fall zusätzlich seinen momentan gehaltenen Wert.

3 Implementation

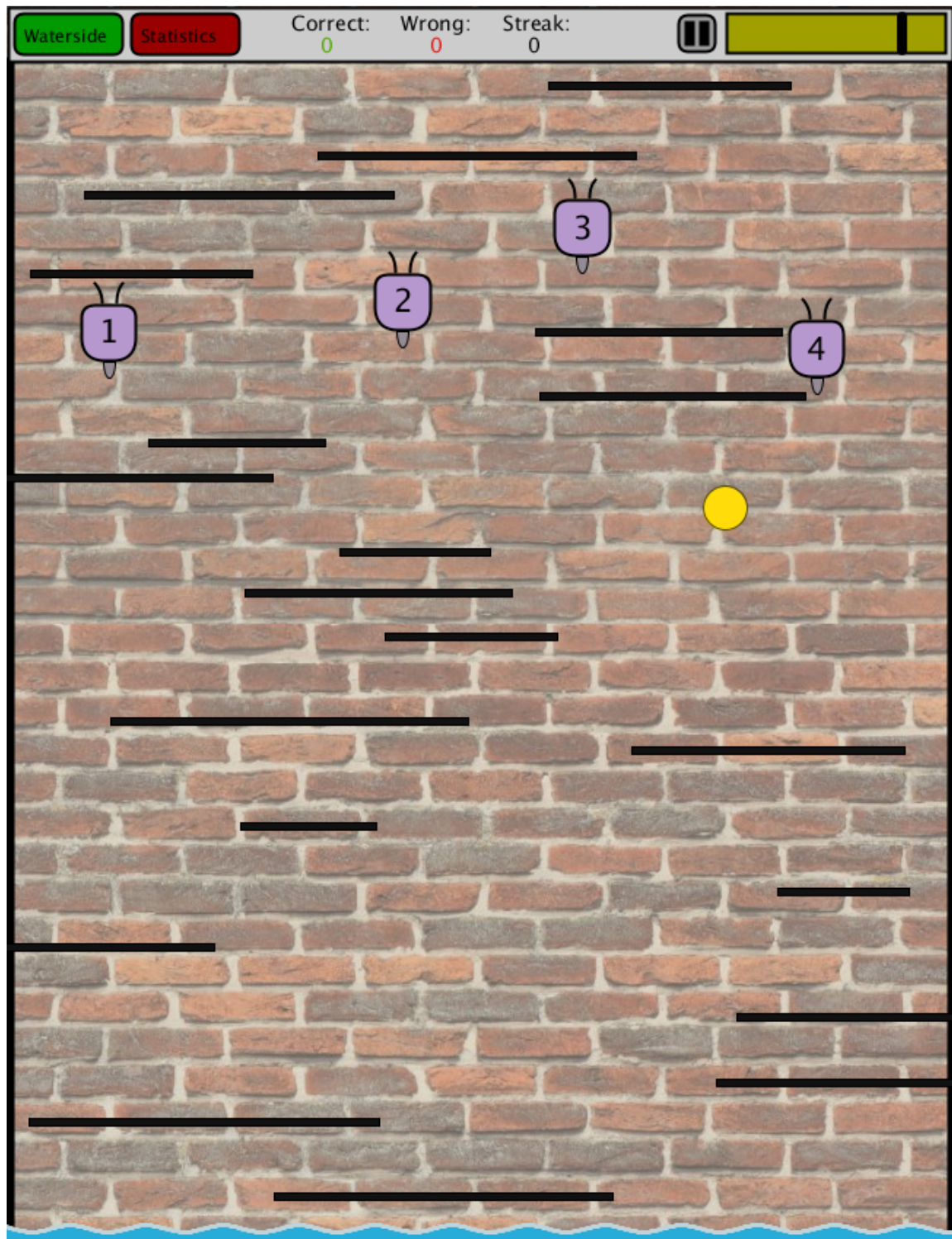


Abbildung 3.1: Waterside Spielbildschirm

3.4 Bewegung

Spielgeschwindigkeit

Neben den mit Spielgeschwindigkeit durch das Sichtfeld wandernden Plattformen wird im Hintergrund der Spielwelt auch das Bild einer Ziegelwand mitbewegt. Um die Illusion räumlicher Tiefe zu gestalten, bewegt sich der Hintergrund nur mit einem konstanten Anteil der Spielgeschwindigkeit. Die so entstehende Bewegungsparallaxe [8][21] wirkt durch die räumlichere Darstellung auch einer Ermüdung der Augen des/der Spielenden entgegen. Sowohl das zufällig generierte Plattformgefüge als auch der Hintergrund werden aufgrund ihrer begrenzten Höhe in voneinander unabhängigen Dauerschleifen durch das Sichtfeld bewegt.

Das in Kapitel 2.6 beschriebene negative Feedbackelement zur Stabilisierung der Spielgeschwindigkeit kann durch die Vertikalkomponente der Spielfigurposition gesteuert werden. Um ein „Springen“ des Sichtfelds zu verhindern, werden zu jedem Frame die 10 aktuellsten Vertikalkomponenten arithmetisch gemittelt, bevor das Ergebnis linear auf die Spielgeschwindigkeit abgebildet wird. Aus signaltheoretischer Sicht kann diese Mittelung auch als Tiefpassfilterung durch ein kausales *Moving Average System* verstanden werden [12].

Bewegung der Spielfigur

Inspiziert von den Gesetzen der klassischen Mechanik wird die zeitdiskrete zweidimensionale „Bewegung“ der Spielfigur im konstanten Sichtfeld nach den Gleichungen 3.1 und 3.2 implementiert. Diese Vorgehensweise lässt sich auf das explizite Euler-Verfahren zur schrittweisen, numerischen Lösung von Anfangswertproblemen zurückführen [1][22]. Vor jedem Darstellungszeitpunkt n wird aus der Beschleunigung $\mathbf{a}[n]$ die Geschwindigkeit $\mathbf{v}[n]$ und der zur grafischen Darstellung benötigte Ort $\mathbf{x}[n]$ berechnet. Zur zeitlich gleichmäßigen Darstellung der Einzelbilder setzt diese Implementation eine konstante Framerate voraus.

$$\mathbf{v}[n] = \mathbf{v}[n - 1] + \mathbf{a}[n] \quad (3.1)$$

$$\mathbf{x}[n] = \mathbf{x}[n - 1] + \mathbf{v}[n] \quad (3.2)$$

3 Implementation

Der Beschleunigungsvektor $\mathbf{a}$ wird zu jedem Frame auf Null zurückgesetzt und ergibt sich im Anschluss durch Akkumulation der Steuerungseingaben des/der Spielenden und den äußerlichen Krafteinwirkungen zu $\mathbf{a}[n]$. Erst dannach wird die aktuelle Position $\mathbf{x}[n]$ nach 3.1 und 3.2 berechnet. Da die Masse der Spielfigur konstant bleibt, können auf diese einwirkende Kräfte intern auch als Beschleunigungen implementiert werden. Zur qualitativen Beschreibung des Bewegungssystems sind also nur die kinematischen Größen $\mathbf{a}$, $\mathbf{v}$ und $\mathbf{x}$ notwendig.

Da intern nicht das Sichtfeld nach oben, sondern nur der Hintergrund und die Plattformen mit der Spielgeschwindigkeit nach unten wandern, wird der Ort $\mathbf{x}$ vor allen anderen bewegungsrelevanten Berechnungen durch die selbe Geschwindigkeit angepasst. Damit ergeben sich vertikale Bewegungen der Spielfigur bezogen auf deren wahrgenommene Position als unabhängig von der Spielgeschwindigkeit.

Die Gravitation wird als vertikal gerichtete konstante Beschleunigung implementiert. Die Spielfigur besitzt eine vom Feedbacksystem abhängige Sprungkraft (siehe auch Kapitel 2.6 und 2.7). Zum Ausführen der Sprungbewegung wird zunächst die Geschwindigkeit der Spielfigur auf Null gesetzt. Im Falle eines gewöhnlichen Sprungs von einer Plattform wird der Wert der Sprungkraft anschließend zur vertikalen Beschleunigung addiert. Für den „Walljump“ stellt selbiger Wert den vektoriellen Betrag einer schief gerichteten Beschleunigungskomponente dar. In beiden Fällen könnte stattdessen auch direkt der Geschwindigkeitsvektor modifiziert werden. Die Vorstellung einer Kraft als Ursache für den Sprung macht die erstere Lösung lediglich intuitiver verständlich.

Die horizontalen Steuerungseingaben können gehalten werden, um die Spielfigur über mehrere Frames hinweg zu beschleunigen. Die Horizontalkomponente des daraus resultierenden Geschwindigkeitsbetrags wird auf ein Maximum beschränkt, womit die Spielfigur nach sehr kurzer Zeit der Beschleunigung eine konstante Höchstgeschwindigkeit erreicht. Durch verschieden große Beschleunigungsbeträge kann die Spielfigur die horizontale Höchstgeschwindigkeit schneller erreichen, wenn sie sich auf einer Plattform befindet. Die horizontalen Bewegungen der Spielfigur werden durch eine bremsende Kraft eingeschränkt. Diese wird aus einem Anteil der horizontalen Komponente der Geschwindigkeit $\mathbf{v}[n - 1]$ berechnet und wirkt in entgegengesetzter Richtung. Diese Implementation ist ein experimenteller

3 Implementation

Kompromiss zur Simulation von Reibung und Strömungswiderstand, und soll dem /der Spielenden eine leichte Trägheit der Spielfigur vermitteln.

Kollisionen mit falschen Antworten erfolgen durch eine einmalig auf die Spielfigur einwirkende Kraft. Diese ist abhängig von der Einfallsrichtung implementiert, wirkt nach deren vektorielltem Betrag aber immer gleich stark.

Modulation und Rauschen

Die zur Darstellung des Antwortsymbols dienende Flügelfläche eines Insekts wird als Rechteck mit abgerundeten Ecken implementiert. Zur Simulation des Flügelschlags werden die Abmessungen und die Radien der Abrundungen des Rechtecks durch verschiedene Sinusfunktionen moduliert.

Zur Umsetzung der natürlichen Bewegungen der Insekten und des Wassers, wird eindimensionales Perlin-Rauschen verwendet; eine Rauschfunktion auf Basis pseudo-zufälliger Gradientenwerte, deren Algorithmus in der Standardversion von Processing bereits in konfigurierbarer Form implementiert ist [2][22].

3.5 Aufgaben zur Gehörbildung

Die gehörbildenden Aufgabenstellungen (siehe auch Kapitel 2.4) bilden den Kern von *Waterside* und werden mithilfe der von der Processing Foundation entwickelten Sound Bibliothek realisiert. Da alle Klänge in Echtzeit generiert werden, müssen keine Audiodateien eingebunden werden. Somit ist für die exportierte Applikation mit einem sehr geringen Speicherbedarf zu rechnen.

Zum gezielten Testen nur einer bestimmten Eigenschaft pro Aufgabentyp werden sehr minimalistische Klänge verwendet. Es ist daher keine komplizierte Klangerzeugung notwendig: Abgesehen von den Aufgaben zur Klangfarbe wird lediglich auf ein Sinusoszillator-Objekt und ein zugehöriges Hüllkurvengenerator- Objekt zugegriffen.

3 Implementation

Waterside soll als Vorbereitung für den Seashore Test dienen und kann deshalb als Inspiration für die im Folgenden beschriebenen Aufgaben verstanden werden. Allerdings ist im Endergebnis der Spielspaß an *Waterside* höher zu bewerten als dessen exakte Reproduktion der originalen Testaufgaben. Daher wird zugunsten einer abwechslungsreicheren Aufgabenvielfalt bewusst auf das Übernehmen bestimmter Eigenheiten des Seashore Tests verzichtet.

Der Seashore Test verwendet für die aus je zwei Einzelklängen bestehenden Aufgabentypen (Tonhöhe, Lautstärke, Tonlänge und Klangfarbe) jeweils Grundfrequenzen aus einer bestimmten möglichst schmalen Bandbreite, um die frequenzabhängigen *Just-Noticeable Differences (JNDs)* [4][23] möglichst konstant zu halten und eine Reproduktion von Messergebnissen zu ermöglichen. In *Waterside* wird hingegen zu jeder Aufgabe dieser Disziplinen eine zufällige Referenzfrequenz aus der Reichweite einer Oktave ausgewählt. Damit einhergehend wird die frequenzunabhängig eingestellte Schwierigkeit durch die Frequenzabhängigkeit der *JNDs* beeinflusst.

Für die Aufgabentypen Tonhöhe, Lautstärke und Tonlänge wird die Reihenfolge von Referenzklang und modifiziertem Klang zufällig festgelegt. Zusätzlich trägt die zufällige Reihenfolge aufeinanderfolgender Aufgabentypen zur Verhinderung einer Mustererkennung durch Spielende bei.

Tonhöhe

Die empfindungsbasierte Frequenzunterschiedsschwelle beim Vergleich aufeinanderfolgender Sinustöne (*JNDF*) hängt von der Frequenz, der Tondauer, der Pausedauer, und nur unwesentlich vom Schalldruckpegel ab [4]. Fastl und Zwicker schätzen die Unterschiedsschwelle unterhalb von 500 Hz mit 1 Hz ab; oberhalb von 500 Hz mit dem 0,002-fachen Wert der Referenzfrequenz (entspricht circa 3,5 Cent) [4]. Die in *Waterside* gewählte Tondauer einer halben Sekunde liegt dabei innerhalb des Gültigkeitsbereichs bisherig genannter Abschätzungen [4]. In musikalischen Intervallen ausgedrückt, ergeben sich aus der Abschätzung des *JNDF* Werts mit 1 Hz für die in *Waterside* gewählten Frequenzen im Bereich von c bis c^1 (261.626 Hz bis 523.251 Hz) Unterschiedsschwellen zwischen 3,3 Cent und 6,6 Cent. Um Menschen mit absolutem Gehör keinen zusätzlichen Vorteil zu verschaffen, wird die zufällige

3 Implementation

Auswahl der Referenzfrequenz aus oben genanntem Spektrum für diesen Aufgabentyp nicht auf diskrete Tonhöhen beschränkt. Als maximaler Schwierigkeitsgrad wird frequenzunabhängig eine Abweichung von 1 Hz eingestellt.

Lautstärke

Die empfindungsbasierte Intensitätsunterschiedsschwelle beim Vergleich aufeinanderfolgender Sinustöne (*JNDL*) hängt vom Schalldruckpegel, der Tondauer, der Pausedauer, und nur unwesentlich von der Frequenz ab [4][23]. Terhardt fasst verschiedene Messungen dieser Methode zusammen und nennt zu Schalldruckpegeln von 40 dB und 80 dB *JNDL* Messergebnisse zwischen 0,4 dB und 1 dB [23]. Fastl und Zwicker geben abhängig vom Pegel sogar geringfügig kleinere Schwellenwerte an; laut ihnen liegen die für *Waterside* gewählten Ton- und Pausedauern von jeweils einer halben Sekunde bezüglich der Messung möglichst kleiner *JNDLs* in einem repräsentativen Bereich [4]. Der Schalldruckpegel ist in *Waterside* jedoch selbstständig von den Spielenden festzulegen, wodurch keine saubere Anpassung des Schwierigkeitsgrads möglich ist. Als Kompromiss wird schlussendlich 0,5 dB als Maximalschwierigkeit gewählt, da von den Spielenden zwar überdurchschnittliche auditive Fähigkeiten, andererseits jedoch keine Einstellungen überhöhter Pegel erwartet werden.

Tonlänge

Die wahrgenommene Dauer eines Sinustons hängt neben dessen physikalischer Dauer vom Schalldruckpegel und der Frequenz ab [4][23]. Die hier gewählte Dauer des Referenztons von einer halben Sekunde liegt bezogen auf dessen wahrgenommene Dauer in einem relativ frequenzunabhängigen Bereich [23]. Der Längenunterschied zwischen den zu vergleichenden Tönen gleicher Frequenz wird in *Waterside* durch ein Verhältnis bestimmt; die „Attack“- und „Release“- Zeiten werden dabei vernachlässigbar kurz gewählt. Seashore verwendete in seinen ersten Veröffentlichungen zur Untersuchung der Unterscheidbarkeit von Zeitintervallen eine abweichende Messmethode, und bezog sich mit seinen Ergebnissen auf absolute Zeitunterschiede [19][20]. Nach einer technischen Analyse des *IEM* Seashore

3 Implementation

Tests, welcher ähnliche Tondauern wie *Waterside* verwendet, wird die maximale Schwierigkeit auf ein Längenverhältnis von 1,07 zu 1 eingestellt.

Klangfarbe

Zur empfindungsbasierten Charakterisierung der Klangfarbe kommen nach Terhardt das Volumen, die Schärfe und die Klanghaftigkeit in Betracht, wobei das Volumen größtenteils von Bandbreite, Frequenzlage und Schalldruckpegel abhängt [23]. Zur Reduzierung des Klangunterschieds auf die Klangfarbe ist es sinnvoll, wenn das Volumen – für Fastl und Zwicker die in diesem Zusammenhang gleichbedeutende Lautheit [4] – zwischen beiden zu vergleichenden Klängen gleich bleibt. Der spektrale Einfluss auf das Volumen beziehungsweise die Lautheit des gefärbten Klangs bleibt hier unberücksichtigt. Es wird lediglich der Pegel des gefärbten Signals an den des sinusförmigen Referenzsignals angepasst. Der gefärbte Klang wird durch additive Kombination eines Sinussignals gleicher Frequenz mit dessen ersten beiden sinusförmigen Obertönen generiert. Mit ausreichend großer Signallänge werden die Kovarianzen zwischen den drei Einzelsignalen vernachlässigbar klein, womit die Amplituden der somit unkorrelierten Einzelsignale nach der vertrauten pythagoräischen Beziehung zusammenhängen [6][14]. Als maximaler Schwierigkeitsgrad wird die gleichmäßige Absenkung der Obertöne gegenüber dem Referenzsignalpegel auf 35 dB eingestellt.

Tongedächtnis

Für die Einzeltöne der wiedergegebenen Sequenzen werden ausschließlich diskrete Frequenzen der gleichstufigen Stimmung verwendet. Statt die Länge der zu vergleichenden Sequenzen zu erhöhen, soll mit steigendem Schwierigkeitsgrad lediglich die Größe des Intervalls des in zufälliger Richtung verschobenen Tons verringert werden. Auf höchster Schwierigkeit verändert sich der zufällig bestimmte Ton der Sequenz somit um nur einen Halbtonschritt.

Rhythmus

Ein Schlag einer Rhythmus-Sequenz wird durch eine hohe Frequenz mit einer sehr kurzen und steilen Amplitudenhüllkurve realisiert. Der Unterschied zwischen zwei zufällig generierten und zeitlich quantisierten Sequenzen ergibt sich durch die Verschiebung eines einzelnen zufällig ausgewählten Schlags auf eine benachbarte Quantisierungsstufe. Der Schwierigkeitsgrad bestimmt hier die Länge der Sequenz, und lässt bis zu sechs Schläge pro Sequenz zu.

3.6 Sound Effekte

Zum Zweck der eindeutigen Repräsentation bestimmter Ereignisse werden verschiedene Sound Effekte verwendet. Dazu werden insgesamt drei Sinusoszillatoren und drei zugehörigen Hüllkurvengeneratoren aus der Processing Sound Bibliothek verwendet. Im Vergleich zu den gehörbildenden Klängen werden die hier beschriebenen Klänge mit leicht reduzierter Lautstärke abgespielt.

Das Fangen eines Schmetterlings – also eine richtige Antwort – wird mit einem Durdreiklang unterlegt. Dazu werden die drei Einzeltöne gleichzeitig aktiviert und deren zeitlicher Amplitudenverlauf durch die zugehörige Hüllkurvengeneratoren angepasst. Vom tiefsten bis zum höchsten Ton werden schrittweise die „Attack“ Zeiten verlängert, und die „Sustain“ Zeiten entsprechend verkürzt. Das Ergebnis ist ein erquicklicher, weich anschwellender Klang.

Eine falsche Antwort hat das Auslösen eines verminderten Dreiklangs zur Folge. Im Vergleich zum Sound Effekt der richtigen Antwort werden hier besonders kurze „Attack“ und „Release“ Zeiten gewählt. Die Hüllkurvengeneratoren aller Einzeltöne erhalten hier identische Parameter. Der Klang soll damit eine empörte und schroffe Stimmung vermitteln.

Der „Game Over“ Sound Effekt wird schließlich aus einer kurzen Sequenz fallender Tritonusintervalle zusammengesetzt. Durch Verlängerung der „Sustain“ und „Release“ Zeiten des letzten Tons soll dieser Sound Effekt ein Gefühl der Enttäuschung erzeugen.

3.7 Statistik

Im Statistik Bildschirm (Abbildung 3.2) kann die Zusammenfassung der Ergebnisse bisheriger Spiele tabellarisch eingesehen werden. Zu jeder Disziplin werden hier zusätzlich zur Darstellung der Anzahl richtiger und falscher Antworten auch die prozentualen Erfolgsquoten berechnet. Der „Absolute Highscore“ stellt die höchste Anzahl innerhalb eines Spiels gefangener Schmetterlinge dar. Mithilfe eines unten eingeblendeten Schalters lassen sich alle gespeicherten Werte zurücksetzen.

Die zugrundeliegenden Werte sind in der Datei <statistics.txt> innerhalb des <data> Ordners der Applikation gespeichert und werden während des Spiels aktualisiert. In der Statistikansicht werden diese ausgelesen und visualisiert.

Die Statistik dient der Information über den eigenen Lernfortschritt, soll aber auch den Wettbewerbsgedanken innerhalb der Spielkultur anregen und somit die Interaktivität des Spielsystems erhöhen.

3 Implementation

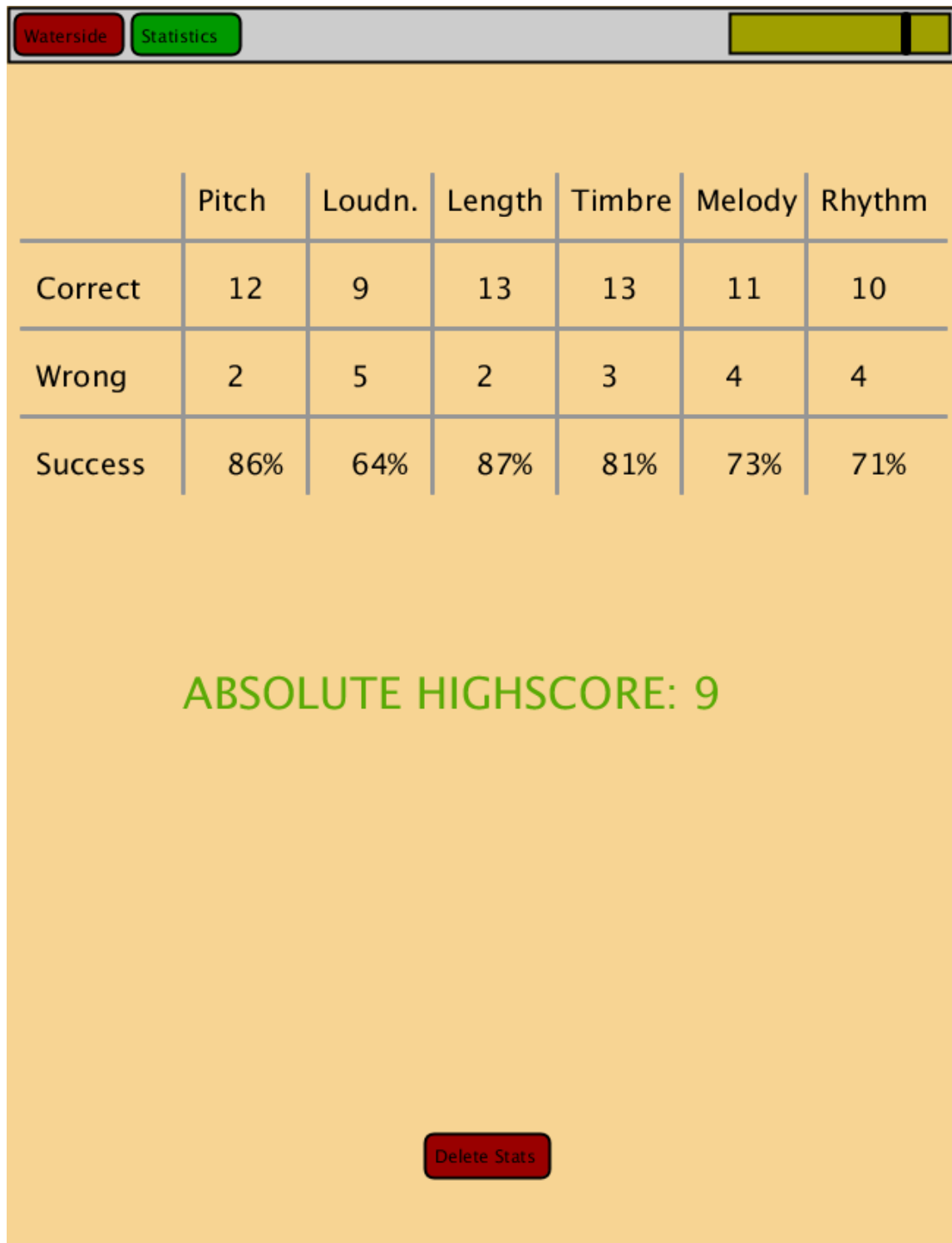


Abbildung 3.2: Waterside Statistikbildschirm

4 Konklusion

4.1 Rekapitulation

Der iterative Entwicklungsprozess des Spiels hatte ein ständiges Wiederholen von Design-, Implementations- und Testphasen zur Folge. Viele Ideen zum Design von *Waterside* kamen dabei jedoch erst während Implementations- und Testphasen auf. Zugunsten einer effizienteren Spieleentwicklung hätte der ersten experimentellen Implementationsphase ein konkreteres Design zugrundeliegen können.

Auch auf Implementationsebene ist eine wohldurchdachte Planung vorteilhaft; so kann bei objektorientierten Projekten ein strukturierter Entwurf der Klassen viel Arbeit ersparen und die Wartbarkeit der Software erhöhen – selbst wenn das Projekt nur von einer einzelnen Person entwickelt wird.

Im Endergebnis konnte die ursprüngliche Idee zufriedenstellend realisiert werden. Der entstandene Prototyp, *Waterside Version 1.0*, ermöglicht ein spielerisches Trainieren bestimmter auditiver Fähigkeiten. Dennoch bestehen zahlreiche Möglichkeiten der Optimierung und Erweiterung (siehe Kapitel [4.2](#)).

Im Idealfall wird der Sinn von *Waterside* von den Spielenden als das Fangen von Schmetterlingen interpretiert, und der Zweck der Gehörbildung völlig ausgeblendet. Ebenfalls erfreulich wäre die Erweckung von Interesse an Gehörbildung – oder gar einem Studium am *IEM* – seitens Spielender, welche ursprünglich nur durch die Jump’n’Run Komponente des Spiels angelockt wurden.

4.2 Ausblick

Waterside Version 1.0 wurde mithilfe von Processing als lokal speicherbares Spiel entwickelt. Diese Implementation ermöglicht eine Erfassung statistischer Daten innerhalb der lokalen Spielkopie, ohne eine Registrierung des/der individuellen Spielenden vorauszusetzen. Stattdessen ist für das Projekt auch eine auf JavaScript basierende Online-Version denkbar, in welcher die Statistik beliebiger Spielender global gespeichert und verarbeitet werden kann. Eine browserbasierte Online-Implementation bietet im Allgemeinen weitere Vorteile hinsichtlich der Verfügbarkeit, Wartbarkeit und Datensicherheit.

Eine Schwachstelle von *Waterside* ist die Voraussetzung einer stabilen Framerate von 60 Bildern – beziehungsweise 60 Berechnungsdurchläufen – pro Sekunde. Die Korrektheit aller Bewegungsabläufe und Auslösezeitpunkte ist von der Gleichmäßigkeit des zeitlichen Rasters der Berechnungsdurchläufe abhängig. Um einen von der geleisteten Framerate unabhängigen Ablauf zu gewährleisten, könnten bewegungsrelevante Berechnungen und Auslösezeitpunkte auf gemessene Zeitunterschiede bezogen werden. Da die eingestellte Framerate auf den meisten getesteten Systemen stabil gehalten werden konnte, ist diese Schwachstelle eher ideologischer Natur, trotzdem wäre eine systemunabhängigere Lösung zu bevorzugen.

Neben den unzähligen Möglichkeiten einzelne Parameter der Jump'n'Run Komponente des Spiels zu optimieren, bietet auch das Feedbacksystem noch viel Spielraum zur Weiterentwicklung oder Änderung. So wäre es beispielsweise denkbar, die Spielgeschwindigkeit gegenläufig zum auditiven Schwierigkeitsgrad zu implementieren, um die Gesamtschwierigkeit über die Dauer eines Spiels zu vergleichmäßigen.

Auf auditiver Seite ließen sich die gehörbildenden Aufgabenstellungen und deren Schwierigkeitsgrade basierend auf intensiverer Recherche oder einer statistisch gemittelten Vielzahl an Spielergebnissen besser an die Vorlage des Seashore Tests beziehungsweise die menschliche auditive Wahrnehmung anpassen. Programmintern könnten außerdem die Ergebnisse der Statistik Datei benutzt werden, um auf das Antwortverhalten des/der individuellen Spielenden zu reagieren.

4 Konklusion

Im Vergleich zum Seashore Test verwendet *Waterside* ein breiteres Spektrum an Tonhöhen. Dies hat frequenzabhängige Schwankungen des wahrgenommenen Schwierigkeitsgrads zur Folge (siehe auch Kapitel 3.5). Daher könnte die Schwierigkeit individueller Aufgabentypen an die frequenzabhängige Wahrnehmbarkeit derer zugrundeliegender Klangeigenschaften angepasst werden. Unter Beibehaltung der höheren Bandbreite ist außerdem eine Implementation einer – oder sogar mehrerer – der Kurven gleicher Lautheit [4][23] zur Bewertung der Einzelfrequenzen aller Aufgabentypen denkbar.

Waterside's gehörbildende Aufgabenstellungen sind unverkennbar an den Seashore Test angelehnt, jedoch wurde bei der Implementation mehrfach von dessen exakter Reproduktion abgewichen. Inwieweit *Waterside* sich tatsächlich zum Trainieren des Gehörs – insbesondere zur Vorbereitung auf den Seashore Test – eignet, ist mit Version 1.0 nicht untersucht worden. Zur Optimierung des Testaufgaben und zur schlussendlichen Überprüfung der Eignung als Gehörbildungsmaßnahme wäre daher eine Evaluation des Projekts sinnvoll.

Literatur

- [1] Arens, Tilo u. a. *Mathematik*. Springer, 2015 (siehe S. 21).
- [2] Burger, Wilhelm. *Gradientenbasierte Rauschfunktionen und Perlin Noise*. School of Informatics, Communications und Media, Upper Austria University of Applied Sciences, 2008 (siehe S. 23).
- [3] Chion, Michel. *Audio-Vision*. Columbia University Press, 1994 (siehe S. 5).
- [4] Fastl, Hugo und Zwicker, Eberhard. *Psychoacoustics: Facts and Models*. Springer, 2007 (siehe S. 24–26, 32).
- [5] Friberg, Johnny und Gärdenfors, Dan. *Audio games: New Perspectives on Game Audio*. Stockholm International Toy Research Centre, KTH, Royal Institute of Technology, 2004 (siehe S. 4–6).
- [6] Howard, David M. und Angus, Jamie A. S. *Acoustics and Psychoacoustics, Fourth Edition*. Focal Press, 2009 (siehe S. 26).
- [7] Huizinga, Johan. *Homo Ludens: A Study of the Play-Element in Culture*. Routledge & Kegan Paul Ltd, 1949 (siehe S. 2, 8).
- [8] Koopman, Pieter. *Implementation of Functional Languages: 11th International Workshop, IFL'99 Lochem, The Netherlands, September 7-10, 1999 Selected Papers*. Springer, 2000 (siehe S. 6, 21).
- [9] Menshikov, Aleksei. *Modern Audio Technologies in Games*. Game Developers Conference, Moscow, 2003 (siehe S. 5).
- [10] Miller, Carolyn H. *Digital Storytelling: A Creator's Guide to Interactive Entertainment*. Focal Press, 2004 (siehe S. 8).
- [11] Oldenburg, Aaron. *Sonic Mechanics: Audio as Gameplay*. Game Studies: The International Journal of Computer Game Research, 2013 (siehe S. 6).

- [12] Oppenheim, Alan V. und Schafer, Ronald W. *Discrete-Time Signal Processing, Third Edition*. Pearson Education, 2014 (siehe S. 21).
- [13] Parker, Jim R. und Heerema, John. *Audio Interaction in Computer Mediated Games*. International Journal of Computer Games Technology, Hindawi Publishing Corporation, 2008 (siehe S. 5, 6).
- [14] Puckette, Miller. *The Theory and Technique of Electronic Music*. World Scientific Publishing Company, 2007 (siehe S. 18, 26).
- [15] Reas, Casey und Fry, Ben. *Processing: A Programming Handbook for Visual Designers and Artists, Second Edition*. MIT Press, 2014 (siehe S. 17).
- [16] Rovithis, Emmanouel. *A Classification of Audio-Based Games in Terms of Sonic Gameplay and the Introduction of the Audio-Role-Playing-Game: Kronos*. Music Department, Ionian University, Corfu, Greece, 2012 (siehe S. 4–6).
- [17] Salen, Katie und Zimmerman, Eric. *Rules of Play: Game Design Fundamentals*. MIT Press, 2004 (siehe S. 1–3).
- [18] Seashore, Carl E. *The Measurement of Musical Talent*. The Musical Quarterly, Oxford University Press, 1915 (siehe S. 7).
- [19] Seashore, Carl E. *The Psychology of Music*. McGraw-Hill Book Company Inc., 1938 (siehe S. 7, 25).
- [20] Seashore, Carl E. *The Psychology of Musical Talent*. Silver, Burdett & Company, 1919 (siehe S. 7, 25).
- [21] Shekar, Siddharth. *Cocos2d Cross-Platform Game Development Cookbook, Second Edition*. Packt Publishing, 2016 (siehe S. 21).
- [22] Shiffman, Daniel. *The Nature of Code: Simulating Natural Systems with Processing*. The Magic Book Project, 2012 (siehe S. 21, 23).
- [23] Terhardt, Ernst. *Akustische Kommunikation: Grundlagen mit Hörbeispielen*. Springer, 1998 (siehe S. 24–26, 32).