

Konzept:	1
Musik:	1
Instrumentalisierung:	3
Wwise integration der Musik:	3
Sound-Effekte:	4
Soundeffekte aus der DAW:	5
Soundeffekte mit Foley-Techniken:	5
Stock-Soundeffekte:	6
Wwise integration:	6
Mixing und Finalisierung:	7
Verwendete Plug-Ins und Instrumentenaufbau:	8
Soundsheet:	8

Konzept:

Das Konzept des Projektes ist es, sehr nah an einem Oldschool-Shooter zu sein. Dies spiegelt sich sowohl in der Musik, als auch in den Soundeffekten wieder, was in den folgenden Abschnitten erläutert werden wird. Alle Sounds im Spiel sind nahe an der Realität, aber gerade noch so weit von ihr entfernt, dass es nicht überzeugend Real wirkt. Die Musik, wie auch die Soundeffekte, wurden bestmöglich realisiert, später aber teilweise so bearbeitet, das kleine Unreinheiten in Soundeffekten und der Musik sich natürlich anhören. So kommt es bei manchen Tracks der Musik zu leichten Übersteuerungen, was zunächst als störend empfunden wurde, allerdings nach dem 'Verunreinigen' einiger anderer Teile des Projektes gut in den Stil passte. Wenn die Sounds von Oldschool-Shootern von früher natürlich schon die Absicht hatten, möglichst realistisch zu sein, so lebt der Stil heute von seiner gewissen Unsauberkeit, die sich auch in den 3D-Objekten und der Spielwelt widerspiegeln. Der folgende Abschnitt wird sich zunächst mit der entstehung der Musik beschäftigen und welche Gedanken sich hinter den getroffenen Entscheidungen befinden.

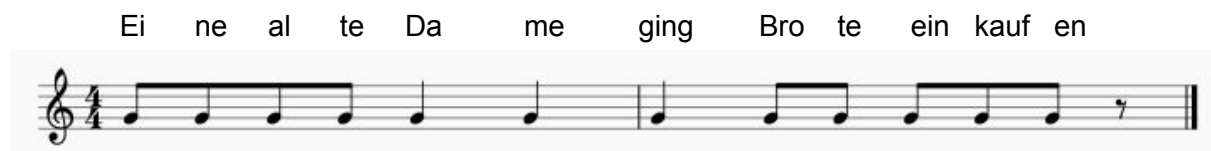
Musik:

Der Soundtrack eines Spiels hat viele wichtige Aufgaben. Er setzt die Stimmung und Atmosphäre des Spiels und kann damit das Spielerlebnis erheblich verbessern. Der Soundtrack sollte dabei natürlich individuell für das jeweilige Spiel erstellt werden, um ein passendes Spielgefühl zu erzeugen. Das Spiel, dass in der Veranstaltung musikalisch untermalt werden sollte, ist das Spiel *Cube*. *Cube* ist ein Ego-Shooter aus dem Jahr 2001 Entwickelt von Wouter van Oormerssen. Das Spiel ist stilistisch, wie viele Ego-Shooter der

späten 90er und frühen 2000er, an Spiele wie *Quake* (1996) und *Doom* (1993) angelehnt, hat aber dennoch einige alleinstellungsmerkmale. So ist *Cube* beispielsweise weniger düster als die beiden angesprochenen Vorreiter des Genres. Dennoch steht es beinahe als ungeschriebene Regel für Ego-Shooter dieser Zeit, dass der Soundtrack sich eher in Richtung Rock bis Metal mit viel Synthesizer-Sounds und Klassischen Einflüssen bewegt. Dieser Fakt wurde beim erstellen des Soundtracks für diese Veranstaltung miteinbezogen.

Die erste Übung, die für die Erstellung des Soundtracks hilfreich war, beschäftigte sich mit Rhythmus-, Tempo- und Melodie-Entwicklung. Dazu wurden zunächst einige kurze Sätze notiert. Anschließend wurden diese abgeklatscht und die länge der Noten und Pausen notiert. Der gewählte Satz, der von mir gewählte Satz für die Veranstaltung war:

“Eine alte Dame ging Bro te ein kauf en.”



Diese zwei Takte sollten über die Zeit und über weitere Übungen hinweg weiter bearbeitet und schlussendlich zu einer Vollständigen Melodie ausgearbeitet werden. Zunächst sollten diese Takte aber in eine DAW (Digital Audio Workstation) übertragen werden, um dort die nächsten Schritte weiter zu bearbeiten.



Nun ging es daran, die Noten in ein Motiv zu verwandeln.



Dieses Motiv wurde nun durch das Anfügen eines weiteren Motivs erweitert, um eine musikalische Phrase zu bilden. Diese Phrase wurde nun in abgewandelter Form wiederholt, um die Phrase zu verlängern und gleichzeitig etwas Abwechslung in die Musik zu bringen. Durch das Wiederholen der aneinander gefügten Phrasen entstand damit eine Periode. Zwischen die Wiederholung wurde zum Schluss eine weitere kurze Phrase, die an das Grundmotiv angelehnt ist, eingeschoben, um die Wiederholung weiter aufzubrechen.



Das Tempo ist mit 155-bpm verhältnismäßig schnell gewählt, um die Hektik der Kämpfe im Spiel passend zu untermalen.

Da es sich bei der Periode um ein hektisches und schnelles Stück handelt, wurde dies als Grundlage für den Battle-Track gewählt. Die nächste Aufgabe war also, auf Grund der Periode einen Mood-Track zu erstellen, der während den ruhigen Phasen des Spiels spielt,

in welchen kein Kampf stattfindet. Um in die Melodie etwas Ruhe zu bringen, wurden viele Noten verlängert. Dadurch entsteht zusätzlich eine Art 'Timewarp'-Effekt. Zusätzlich wurde die Tonfolge abgeändert, um besser mit der Entschleunigung durch die veränderte Notenlänge zu harmonisieren. Dabei wurde dennoch darauf geachtet nicht zu weit vom ursprünglichen Motiv abzuweichen, um die Ähnlichkeit zwischen Mood-Track und Battle-Track nicht zu verlieren. Damit wird die Musik zugleich unheilvoller und bedrückender.



Instrumentalisierung:

Die Instrumentalisierung bietet eine weitere Möglichkeit, um die Atmosphäre, die der Soundtrack vermitteln soll, zu unterstützen. Da der Soundtrack, wie das Spiel selbst, an Ego-Shooter wie *Quake* und *Doom* angelehnt sein soll, wurden Instrumente gewählt, die häufig in den Musikrichtungen Rock und Metal zu finden sind. E-Gitarre, E-Bassgitarre und Schlagzeug. Um der Musik einen etwas düsteren Ton zu verleihen und Abwechslung bei der Instrumentalisierung zu bekommen, wurden noch andere Instrumente, wie Orgel und eine stark verzerrte Drehleier, hinzugefügt. Diese spielen eigene Perioden mit längeren Tönen über die Melodie, welche von den Gitarren gespielt wird. Dies spielt für das *vertical remixing* eine große Rolle, da die Tracks dieser Instrumente jederzeit austauschbar sind oder ganz weggelassen werden können. Zusätzlich füllen Drehleier und Orgel auch den höheren Klangbereich, da sowohl E-Bassgitarre als auch die E-Gitarre verhältnismäßig tiefe Töne spielen. Um einen ähnlichen und doch sehr unterschiedlichen Mood- und Battletrack zu erhalten, wurde hier wieder primär auf das Abändern der Notenlänge zurückgegriffen. Der Moodtrack wirkt durch die langgezogenen Noten damit insgesamt eher etwas unheimlich, während der Battletrack durch die reine Geschwindigkeit bereits ein Gefühl der Hektik und Anspannung auslöst. Das Schlagzeug spielt einen einfachen Rhythmus, um die restliche Musik mit Struktur zu unterfüttern. Außerdem wurde ein zweiter Drum-Track erstellt, welcher gegen den ursprünglichen Drum-Track getauscht werden kann. Alternativ können auch beide Drum-Tracks gleichzeitig gespielt werden, wie es im Boss-Track beispielsweise der Fall ist. Für den Mood-track wurde der Einfachheit halber ein langsamerer Rhythmus gewählt.

Der Chord-Track ist in beiden Tracks der Gleiche und begleitet die restlichen Instrumente mit Akkorden in der Tonart Moll. Diese werden von zwei verschiedenen Gitarren gespielt, welche ebenfalls getrennt voneinander oder zeitgleich gespielt werden können. Für den Battle- und Boss-Track wurde zusätzlich ein Motiv für eine E-Gitarre geschrieben, das mit kurzen, hohen Tönen den Stressfaktor weiter erhöht.

Wweise Integration der Musik:

Nachdem die einzelnen Tracks aus der DAW gerendert wurden, wurde die Musik in Wwise integriert und zusammengestellt. Insgesamt wurden vier Battle-Tracks und vier Mood-Tracks erstellt, welche durch Randomisierung der einzelnen Instrumente weitere Variationen erlauben. Die jeweiligen Tracks wurden dabei nach unterschiedlichen Regeln erstellt. So gibt es Tracks in denen beispielsweise weder die Drehleier noch die Orgel gespielt wird. In anderen Versionen des Tracks wird stattdessen ein anderes Instrument nicht gespielt. Dudelsack und Orgel sind dabei beispielsweise als "Random Step" zusammengefasst. Dies verhindert, dass beide Instrumente gleichzeitig spielen. Zugleich kann damit ebenfalls die Abwechslung der Tracks erhöht werden.

Die verschiedenen Tracks wurden in einen "Random Step" Container angeordnet. Dies bewirkt, dass die verschiedenen Tracks zufällig ausgewählt werden. Dieser Schritt wurde sowohl für den Combat-Track als auch den Explore-Track durchgeführt. Da der Boss-Track lediglich eine Version hat, der aus allen möglichen Instrumenten besteht, welche mit Hoch- und Tiefpassfiltern modifiziert wurden, wurde hier darauf verzichtet.

Der Boss-track wurde dadurch realisiert, dass sämtliche Tracks, die in den Battle-Tracks vorhanden sind, gleichzeitig gespielt werden. Nur die Drehleier und die Orgel spielen nicht direkt gleichzeitig. Der Track startet mit der Drehleier und faded dann über die Periode hinweg in die Orgel. Wenn sich der Track wiederholt beginnt er wieder mit der Drehleier.

Nachdem sowohl Battle-, Boss-, als auch Mood-Track integriert waren, wurden noch weitere kurze Musikstücke in der DAW erstellt, um die restlichen, kurzen Segmente im Spiel zu füllen. Dies sind der Story-Track, der Death-Track und der Victory-Track. Außerdem wurde ein kurzes Übergangsstück erstellt, welches von einem beliebigen Track in den Battle-Track überleitet.

Der Victory-Track spielt bei Abschluss des Levels und ist ein fixes, kurzes Musikstück, das lediglich aus einem Schlagzeug des Battle-Tracks und einer E-Gitarre besteht. Die einzelnen Noten werden dabei stetig höher und, wie im Boss-Track, schnell aufeinanderfolgend gespielt, was einen erfolgreichen Abschluss des Levels unterstützt. Im Gegensatz dazu werden die Noten des Death-Tracks stetig tiefer und länger.

Der Story-Track ist eine leichte Abwandlung des Mood-Track. Die Instrumentalisierung des Story-Tracks kann im Mood-Track nicht auftreten, um einen klaren Unterschied zwischen den beiden Tracks zu erzeugen. Dennoch sollte er sehr nah an dem Mood-Track bleiben, weswegen lediglich die Instrumentalisierung geändert wurde.

Sound-Effekte:

Für die Soundeffekte wurde sich ebenfalls an Ego-Shootern der späten 90er und frühen 2000er orientiert. In Ego-Shootern dieser Zeit wurden annähernd realistische Soundeffekte mit Sounds vermischt, die eher aus dem 8-Bit Bereich kommen. Eine genaue Auflistung der Soundeffekte und wie diese erdacht sind, kann aus dem Sound-Sheet am Ende entnommen werden. Es wurde versucht möglichst viele Soundeffekte selbst zu erstellen und nur im Notfall auf Stock-Sounds zurückgreifen, was in erster Linie für die Waffensounds nötig war. In diesem Abschnitt werden einige Sounds beispielhaft vorgestellt, wie sie entstanden sind und was der Gedankengang hinter dem Soundeffekt war.

Soundeffekte aus der DAW:

Die meisten Soundeffekte die in der DAW erstellt wurden, wurden mit Synthesizern erstellt und haben so einen gewissen 8-Bit Charme. Sounds wie der Teleport-Emitter, Teleport, Health-Pickup oder Rumble, wurde alle mit Synthesizern stellt. Der Teleport-Emitter besteht aus einer einzelnen Note, über die ein Flanger-Effekt gelegt wurde, um einen pulsierenden Sound zu erstellen. Der Sound wurde zusätzlich mit einem Distortion-Effekt verzerrt. Mit Effekten wie diesen wurden im Grunde alle restlichen Sounds erstellt, die in der DAW erstellt wurden. Da der Health-Pickup Sound sehr wichtig für das auditive Feedback ist, ist dieser im vergleich zu anderen Sound-Effekten sehr klar und bedient eher das höhere Tonspektrum, damit er sich von den restlichen Sounds klar abhebt. Der Antrieb der Rakete des Raketenwerfers wurde mit einem Noise-Generator erstellt.

Soundeffekte mit Foley-Techniken:

Der größte Teil der Soundeffekte wurde mit Foley-Techniken erstellt. So zum Beispiel die Schrittgeräusche für den Spielcharakter und die Gegner im Spiel. Die Böden im zu vertonenden Level sind Metall und Sand. Für den Metall-Sound wurde mit Kampfstiefeln auf einer abmontierten Seitenwand eines PC-Gehäuses getreten. Für die Schritte auf Sand wurde auf Kies zurückgegriffen, da Schritte auf Sand zwar offensichtlich richtiger wären, jedoch nicht präsent genug.

Die Sounds für die Monster im Spiel wurden alle durch verschiedene "Grunz"- und "Quietsch"- Geräusche mit der eigenen Stimme realisiert. Für Bauul beispielsweise wurden verschiedene "Pigsqueal"-Sounds aufgenommen. Der unterschied zwischen den Soundeffekten für Grunt, Pain und Death für die einzelnen Monster besteht in erster Linie daraus, dass die Painsounds etwas gequälter sind und die Deathsounds länger und noch gequälter. Die Sounds wurden mit einem Mikrofon aufgenommen und anschließend mit pitch-shifting und ggf. distortion überarbeitet. Ähnliches gilt für die Sounds die vom Spielercharakter selbst kommen.

Um die Referenzen zu spielen wie *Quake* weiter zu festigen, wurde das aufheben des Quaddamage Powerups ähnlich vertont, wie es in diesen Spielen der Fall ist. Der Kommentator dieser Spiele wird häufig mit einer tiefen, verzerrten Stimme dargestellt. Dies wurde bei der Vertonung von *Cube* wieder aufgegriffen. Die aufgenommene Voiceline wurde stark gepitchshiftet. Dabei wurden verschiedene Pitchshifter auf den linken und rechten Tonkanal gelegt, was die Stimme Omnipräsenter wirken lässt. Selbiges wurde für den eigentlich unvertonten Satz "To move beyond this empty room walk the ledge or meet thy doom" gemacht, der im Story-abschnitt des Levels angezeigt wird.

Sounds, wie das Geräusch welches abgespielt wird wenn versucht wird zu schießen obwohl keine Munition mehr vorhanden ist, wurden durch das abfeuern eines ungeladenen Soft-Air Revolvers realisiert. Gleiches gilt für den Munition-Pickup-Sound. Dieser wurde allerdings weiter durch pitch-shifting bearbeitet.

Die Sounds der Schleimbälle, welche die Goblins abfeuern, wurden mittels einem in Öl getränkten Backpinsel, dessen Borsten mit der Hand zusammengedrückt wurden, erstellt.

Stock-Soundeffekte:

Die meisten Soundeffekte, die aus Stockbibliotheken entstammen, sind Sounds, die mit den Waffen der Spieler und Monster zu tun haben. Beispielsweise die Schussgeräusche, die Munitionshülsen, die auf den Boden fallen, Explosionen, etc. Diese wurden allerdings vor der Integration in Wwise noch in einer DAW mit verschiedenen Effekten, wie Pitchshiftern und Distortion, bearbeitet und auf Länge zugeschnitten.

Wwise integration:

Nachdem sowohl die Musik und die Soundeffekte in Wwise integriert und auf die jeweiligen Events gemappt wurden, wurden noch einige Kleinigkeiten in Wwise selbst erstellt und bearbeitet. Besonders die Erzeugung von Raum war in diesem Bereich wichtig. Dafür wurden verschiedene Attenuations erstellt. Beispielsweise wurden für den Teleporter und das Antriebsgeräusch der Raketen verschiedene Attenuations erstellt. Der Teleportersound sollte früher abfallen als das Antriebsgeräusch.

Desweiteren wurden einige Soundeffekte mit einer 3D-Position bestückt. Die Geräusche der auf den Boden fallenden Munitionshülsen folgen verschiedenen Pfaden, die im 3D-Position->Automation Tab erstellt wurden. Die Funktion "Hold Emitter Position and Orientation" wurde für die fallenden Hülsen ebenfalls aktiviert, damit die Soundquelle sich nicht mit dem Spieler mitbewegt, sondern ab dem Zeitpunkt des Auftreffens an einer festen Position in der Spielwelt bleibt und sich dessen Position lediglich über den definierten Pfad verändert. Verwirrender Weise funktioniert diese Behandlung nicht für den Raketenwerfer. Hier wurde der gleiche Switch integriert, wie für die anderen Waffen. Dennoch wird hier lediglich das Auftreffen einer Hülse auf Stein abgespielt und nie das Auftreffen auf Sand. Im 3D-Position Tab von Wwise muss der Soundeffekt auch auf "Listener with Automation" statt "Emitter with Automation" gestellt werden. Vermutlich hängt dies an der Art und Weise, wie das Event funktioniert. Die Sounds die über das Event abgespielt werden sind scheinbar an das Projektil gekoppelt. Stellt man die 3D-Position auf "Emitter with Automation", so wird der Sound der auf den Boden fallenden Hülse am Einschlagspunkt der Rakete gespielt.

Dementsprechend wurde sich mit dem besten, das erreicht werden konnte, zufrieden gegeben, was bedeutet, dass die Hülsen des Raketenwerfers immer mit dem Sound für harte Oberflächen verbunden ist.

Als kleines Detail wurde ebenfalls darauf geachtet, dass es sich bei der Shotgun um eine doppelläufige Schrotflinte handelt, weswegen der Switch für die fallenden Hülsen bei der Shotgun zweimal integriert worden ist, um zwei verschieden fallende Hülsen zu simulieren. Für die Fußstapfen des Spielers wurde ebenfalls mit 3D-Positionen gearbeitet, damit die Sounds abwechselnd links und rechts abgespielt werden, um das Laufen besser zu simulieren.

Für jeden Gegnertypen wurden ebenfalls Switches für die Laufgeräusche erstellt. Für kleinere Gegner wurden die Schritte höher gepitcht, für größere tiefer.

Da die Sounds für das Auftreffen der Munitionshülsen auf den Boden sehr metallisch klingen und, sollte der Spieler auf Sand stehen, nicht wirklich Sinn ergibt, wurde für diese ebenfalls ein Switch erstellt. Steht der Spieler auf Metall, Beton oder einer anderen festen Oberfläche, spielen die dafür vorgesehenen Soundeffekte. Auf Sand, Kies oder ähnlichem wird ein anderer Soundeffekt für das Fallen der Hülsen gespielt, der nur einen dumpfen Aufprall der

Hülsen simuliert. Dieser Soundeffekt wurde wieder selbst erstellt, indem ein Stein auf eine Zeitung fallen gelassen wurde und anschließend in einer DAW bearbeitet.

Die Musik ändert sich in erster Linie über die bereits erwähnten Tracks. Allerdings wurde ebenfalls eine Synchronisation mit dem Spielerleben integriert. Fällt das Spielerleben unter 35 wird sowohl ein Highpass-Filter angewendet, der stetig stärker filtert, je weniger Leben der Spieler besitzt. Zusätzlich wird die Wiedergabegeschwindigkeit stetig verringert. Dies bewirkt, dass der Spieler auch über die Musik erfährt, dass er nur noch wenig Leben besitzt. Außerdem wird die Musik damit auf eine Art und Weise verzerrt, die Gefahr suggeriert. Es wurde auch ein neuer Herzschlag-Sound eingefügt, dessen Lautstärke ebenfalls an das Spielerleben gekoppelt ist. Der Pitch des Herzschlages wird immer höher, je tiefer das Spielerleben fällt, was den Nebeneffekt hat, dass der Herzschlag auch immer schneller wird.

Ein Problem, das leider nach wie vor besteht, ist, dass beim erneuten Start des Levels manchmal noch kurz der Battle-Track angestimmt wird, statt dem Mood-Track. Dies geschieht in der Regel, wenn der Spieler sich in einem Kampf befindet und sich dann dazu entschließt das Level neu zu starten. Um dies zu beheben, wurde im Profiler-Tab von Wwise versucht, die Quelle dieses Problems zu finden und es anschließend zu beheben. Dabei wurde klar, dass der Parameter "EnemyAware" bei Neustart des Levels falsch gesetzt wird und deswegen der Music-Switch-Container "Alive.Gameplay.Combat" statt "Alive.Gameplay.Explore" ausgewählt wird. Das Event, das die Musik auslöst, Music, wurde daraufhin mit verschiedenen Mitteln bearbeitet, doch kein Versuch führte zum gewünschten Ergebnis. So wurden direkt vor dem Event "Play Wwise 201 Music" Events gesetzt, die den Status auf "Alive.Gameplay.Explore" stellen sollten. Das Ergebnis blieb allerdings das gleiche. Auch wenn das Event "Enemys_Unaware" gepostet wurde, änderte sich nichts. Auch bei gleichem Delay von Musikstart und Parameteränderung wurde der Parameter "EnemyAware" gesetzt. Diese Möglichkeiten wurden auch an anderer Stelle versucht, wie beim Ableben des Spielers. Vermutlich ändert *Cube* den Parameter zu spät, weswegen der Gamecall bei Wwise 'fehlerhaft' ankommt. Ein Workaround für dieses Problem wäre gewesen, bei Aktivierung der Musik stets zuerst den Mood-Track zu spielen. Ein zufriedenstellendes Ergebnis im Übergang konnte aber nicht erzeugt werden.

Mixing und Finalisierung:

Beim Mixing wurde in erster Linie darauf geachtet, dass die wichtigen Sounds stets gut hörbar sind und unwichtige Sounds oder die Musik im schlimmsten Fall in den Hintergrund gedrängt werden. Das Aufheben von Lebens-Pickups beispielsweise ist ein Sound, der vom Spieler unter allen Umständen gut gehört werden sollte. Auch Sounds von Gegnern sollten im Mix jederzeit gut hörbar sein, um dem Spieler die Position des Gegners zu verraten. Sounds die eher der Atmosphäre dienen, wie Schrittgeräusche, Sprunggeräusche, etc. spielen im Kampfgeschehen keine große Rolle, weswegen sie im Mix sowohl im Battle-Track als auch im Boss-Track ruhig untergehen können. Für die Stimmung wurde entschieden, dass die Musik hier mehr zum Spielerlebnis beiträgt, als einige atmosphärische Soundeffekte. Damit es unmöglich wird, dass manche wichtige Soundeffekte von den unwichtigen überlagert werden, wurde für einige dieser Sounds ein eigener Audiobus erstellt. Dieser regelt die Soundeffekte und auch die Musik mittels auto-ducking herunter, damit die Sounds auf diesem Audiobus immer gut zu hören sind, egal wie viele andere Soundeffekte das Geschehen überlagern würden. Soundeffekte, die auf diesem Audiobus

liegen, sind unter anderem auch die Sounds für das aufheben, des Quad-Powerups und der Voiceline im Storyabschnitt. Das Aufheben des Quad-Powerups sollte für den Spieler immer gut zu hören sein, damit er weiß, dass der Schaden des Spielers erhöht ist. Die Voiceline liegt auf diesem Audiobus, damit Schussgeräusche o.ä. nicht von der Voiceline nehmen. Das restliche Mixing bestand in erster Linie daraus, die einzelnen Sounds aneinander anzupassen, bis keiner der Sounds mehr übermäßig laut oder zu leise war.

Verwendete Plug-Ins und Instrumentenaufbau:

Verwendete DAW: Reaper

Reaper-Externe Instrumente und Effekte:

Drehleiher: <https://music-society.de/showthread.php?tid=3931>

(Weiter bearbeitet mit: JS: Distortion, VST: ReaEQ, easyReverb)

easyReverb: https://vst.saschart.com/easyreverb_digital_vst_low_resources

DVS Guitar: <https://vst4free.com/plugin/192/>

Church Organ: <https://vst4free.com/plugin/622/>

4font Bass: <https://vst4free.com/plugin/190/>

Sonatina Orchestra: <https://vst4free.com/plugin/2541/>

Hell Drums: <https://vst4free.com/plugin/2101/>

Drum Dream: <https://vst4free.com/plugin/1549/>

Haunted Guitar: <https://vst4free.com/plugin/3189/>

StingerMax: <https://vst4free.com/plugin/2947/>

Reaper-Interne Effekte und Instrumente:

JS: Distortion

JS: Gaussian Noise Generator

JS: Flanger

VST: ReaPitch

VST: ReaEQ

VSTi: ReaSynth

Soundsheet:

Spieler:

Event_Name/Sound_Name	Soundbeschreibung	verwendete Methode	ggf. Quelle
Jump	Grunzen für Sprung ~x4	Innen, Selbst	
Foot_Player	Metal ~x4, Sand ~x4,	Innen, Selbst	

	Tile ~x4		
Land	Landen aus Höhe ~x2	Innen, Selbst	
Pain	Schmerzensschrei/ Stöhnen ~x4	Innen, Selbst	
NoAmmo	Metallisches klacken, keine Munition mehr	Innen, Selbst	
WeaponLoad	Waffenwechsel	Stock	https://www.fesliyanstudios.com/sound-effects-search.php?q=weapon+cock
Death_Player	Spieler Todesschrei ~x2-3	Selbst, Innen	
BodyImpact	Körpertreffer auf Spieler	Selbst, Innen	

Waffen-Spieler:

Fire_Fist_Player	Schlaggeräusch ~x4	Innen, Selbst	
Fire_Rifle_Player	Peitschendes Schussgeräusch	Stock	https://www.fesliyanstudios.com/sound-effects-search.php?q=mashine+gun
Fire_Shotgun_Player	Shotgun-sound	Stock	https://soundbible.com/tags-shotgun.html
Shotgun_Pump_in	Repetier heranziehen	Stock	https://www.fesliyanstudios.com/sound-effects-search.php?q=Shotgun+pump
Shotgun_Pump_out	Repetier verschieben	Stock	https://www.fesliyanstudios.com/sound-effects-search.php?q=Shotgun+pump
Fire_Chaingun_Player	MG-Schüsse	Stock	https://www.fesliyanstudios.com/sound-effects-search

			h.php?q=machine+gun
Fire_Rocket_Player	Schuss Antriebsgeräusch	Stock DAW	https://www.soundsnap.com/tags/grenade_launcher
End_Rocket_Player	Einschlag/Explosion	Stock	https://www.youtube.com/watch?v=f0gB369NYFA
Shotgun Shells, Metal/Concrete	Auftreffen der Shotgun Hülsen auf Boden ~x3	Stock	https://www.fesliyanstudios.com/sound-effects-search.php?q=shotgun+shells
Shotgun Shells, Sand/Gravel	Auftreffen der Shotgunhülsen auf Boden ~x3	Selbst, Innen	
Rifle-Rounds, Metal/Concrete	Auftreffen der Gewehr Hülsen auf Boden ~x3	Stock	https://www.youtube.com/watch?v=QCpVjewi3QM
Rifle-Rounds, Sand/Gravel	Auftreffen der Gewehr Hülsen auf Boden ~x3	Selbst, Innen	
MG-Rounds, Metal/Concrete	Auftreffen der Chaingun-Hülsen auf Boden ~x3	Stock	https://www.youtube.com/watch?v=QCpVjewi3QM
MG-Rounds, Sand/Gravel	Auftreffen der Chaingun-Hülsen auf Boden ~x3	Selbst, Innen	
Rocketlauncher-Rounds, Metal/Concrete	Auftreffen der Raketen-Hülsen auf Boden	Stock	https://www.fesliyanstudios.com/sound-effects-search.php?q=shotgun+shells
Rocketlauncher-Rounds, Sand/Gravel (Sound wird nicht abgespielt, Switch hat keine	Auftreffen der Raketen-Hülsen auf Boden	Selbst, Innen	

auswirkung auf Raketenwerfer)			
Rifle-Chambering-In	Zurückziehen des Repetiers	Stock	https://www.fesliyanstudios.com/sound-effects-search.php?q=rifle+cocking
Rifle-Chambering-Out	Vorschieben des Repetiers	Stock	https://www.fesliyanstudios.com/sound-effects-search.php?q=rifle+cocking

Pickups:

ItemPup	Quad, Sound signalisiert Buff (Acid)	Selbst, Innen	
PupFire	Soundlayer über Schuss, Signalisiert Quad	DAW	VSTi: ReaSynth
pupOut	Quad, Sound signalisiert Buff ausgelaufen	DAW	VSTi: ReaSynth
ItemSpawn (In Level nicht enthalten, dennoch implementiert)	Item ist nachgespawnt signalisierung	DAW	VSTi: ReaSynth
ItemAmmo	Munition aufnehmen	Selbst, Innen	
ItemHealth100	Leben aufnehmen	DAW	VSTi: ReaSynth
ItemArmor (nicht im Level, dennoch implementiert)	Rüstung aufheben (Zipper)	Selbst, Innen	

Monster:

Grunt_Rhino	Grunzen von Rhinogegner ~x2	Selbst, Innen	
Foot_Rhino	FootstepsPlayer - heller	-	

Fire_Chaingun_Monster	Kurze Feuersalve der Chaingun des Spielers	Stock	https://www.fesliyanstudios.com/sound-effects-search.php?q=machine+gun
Death_Rhino	Rhinogegner stirbt ~x2	Selbst, Innen	
Grunt_Ogre	Grunezen von Ogergegner ~x3	Selbst, Innen	
Foot_Ogre	FootstepsPlayer	-	
Death_Ogre	Ogergegner stirbt ~x2	Selbst, Innen	
Grunt_Rat	Rat (shotgun-Typ) grunzt ~x3	Selbst, Innen	
Foot_Rat	Footsteps	-	
Death_Rat	Rat (shotgun-Typ) stirbt ~x2	Selbst, Innen	
Grunt_Goblin	Grunezen von Goblinglegner ~x3	Selbst, Innen	
Foot_Goblin	Footsteps - etwas heller	-	
Death_Goblin	Goblin-gegner stirbt ~x2	Selbst, Innen	
Fire_SlimeBall_Monster	Schleimball schießen	Selbst, Innen	
Fire_Iceball_Monster	Eisballschießen	Stock	https://www.soundsnap.com/search/audio/Ice%20cracking/score
End_Iceball_Monster	Eisballende	Stock	https://www.soundsnap.com/search/audio/Ice%20cracking/score
Hit_Iceball_Monster	Eisball trifft	Stock	https://www.soundsnap.com/search/audio/Ice%20cracking/score
Fire_Fireball_Monster	Feuerball-schießen, (FLaunch)	Stock	https://www.soundsnap.com/tags/fireball

Hit_Slimeball_Monster	von Schleimball getroffen	Selbst, Innen	
Hit_Fireball_Monster	von Feuerball getroffen (explodeMini) ~x3	Stock	https://www.soundsnap.com/tags/fireball
End_Fireball_Monster	Feuerball-ende (explodeMini)	Stock	https://www.soundsnap.com/tags/fireball
Pain_Rat	Ratte getroffen, nicht tot ~x3	Selbst, Innen	
Pain_Rhino	Rhino getroffen, nicht tot ~x3	Selbst, Innen	
Pain_Ogre	Oger getroffen, nicht tot ~x3	Selbst, Innen	
Pain_Goblin	Goblin getroffen, nicht tot ~x3	Selbst, Innen	
Grunt_Knight	Grunzen von Knightgegner ~x4	Selbst, Innen	
Foot_Knight	Footsteps - bissl tiefer	-	
Death_Knight	Knightgegner stirbt ~x2	Selbst, Innen	
Pain_Knight	Knight getroffen, nicht tot ~x3	Selbst, Innen	
Grunt_Bauul	Grunezen von Bauulgegner ~x3	Selbst, Innen	
Foot_Bauul	Footsteps - tiefer	-	
Death_Bauul	Bauulgegner stirbt	Selbst, Innen	
Pain_Bauul	Bauul getroffen, nicht tot ~x3	Selbst, Innen	
Fire_Rocket_Monster	Raketensabschuss Antriebsgeräusch	Rocketlauncher-Player	
End_Rocket_Monster	Rackete Endet	Rocketlauncher-Player	
Fire_Shotgun_Monster	Shotgunschuss von Monster	Shotgunsound-Player	

Sonstiges:

Emitter_teleport	Konstantes Geräusch des Teleporters	DAW	StingerMax: https://vst4free.com/plugin/2947/
Teleport	Teleportiert werden	DAW	VSTi: ReaSynth
Rumble	Levelstart, Türschalter	DAW	VSTi: ReaSynth
Heartbeat	Herzschlag	selbst,Innen	
Meet_Thy_Doom	Story-Voiceline	Selbst, Innen	