

HOCHSCHULE FÜR ANGEWANDTE WISSENSCHAFTEN HAMBURG

Experimentelle Analyse von Spielfeldern in Strategic Defense Games

Masterarbeit - Zeitabhängige Medien
Sound/Vision/Games

Erster Gutachter: Prof. Ralf Hebecker
Zweiter Gutachter: Prof. Gunther Rehfeld

Vorgelegt von
Christopher Klose
Matrikelnummer: 1938603

Wichmannstraße 40a
22607 Hamburg

Mai 2015

Masterarbeit am	Department Medientechnik Fakultät Design, Medien & Information (DMI) Hochschule für angewandte Wissenschaften Hamburg Finkenau 35 22081 Hamburg
Vorgelegt von	Christopher Klose Matrikelnummer: 1938603
Betreut durch	Prof. Ralf Hebecker Hochschule für angewandte Wissenschaften Hamburg Department Medientechnik Fakultät Design, Medien & Information Prof. Gunther Rehfeld Hochschule für angewandte Wissenschaften Hamburg Department Medientechnik Fakultät Design, Medien & Information
Zeitraum	21.01.2015 – 21.05.2015

EIGENSTÄNDIGKEITSERKLÄRUNG

Hiermit versichere ich, die vorliegende Arbeit selbständig ohne fremde Hilfe verfasst und keine anderen Quellen und Hilfsmittel als die angegebenen benutzt zu haben. Die aus anderen Werken wörtlich entnommenen Stellen oder dem Sinn nach entlehnten Passagen sind durch Quellenangaben kenntlich gemacht.

(Unterschrift)

INHALTSVERZEICHNIS

Abbildungsverzeichnis

Tabellenverzeichnis

1	Einleitung.....	1
1.1	Gliederung und Herangehensweise	2
1.2	Verwandte Arbeiten	3
1.3	Hypothese	4
2	Hauptteil.....	5
2.1	Strategic Defense Games	5
2.1.1	Überblick.....	5
2.1.2	Kernelemente	6
2.1.3	Spielfelddesign	8
2.1.4	Beispiele	9
2.2	Praktische Umsetzung	14
2.2.1	Konzept.....	14
2.2.2	Zielgruppenabgrenzungskriterien	14
2.2.3	Plattformabgrenzungskriterien	14
2.2.4	Prototyp.....	15
2.2.5	Analysemethoden.....	23
2.2.6	Entwicklungsprobleme	24
2.3	Auswertung der Analyseergebnisse	25
2.3.1	Ergebnisse.....	25
2.3.2	Auswertung der Ergebnisse.....	29
3	Schluss	30
3.1	Abschließende Betrachtung	30
3.2	Ausblick.....	31

Anhang

Quellenverzeichnis

ABBILDUNGSVERZEICHNIS

Abbildung 1 - Strategy Defense Game Gameloop.....	5
Abbildung 2 - Spielfelder Kingdom Rush IP (Screenshots)	10
Abbildung 3 - Spielfeld Flash Element TD (Screenshot)	10
Abbildung 4 - Vector TD Spielfelder (Screenshots)	11
Abbildung 5 - Spielfelder Bloons TD IP (Screenshots)	12
Abbildung 6 - Spielfelder Gemcraft IP (Screenshots)	12
Abbildung 7 - Spielfelder Cursed Treasure IP (Screenshots)	13
Abbildung 8 – Release Version Quadfender (Screenshots).....	15
Abbildung 9 - Unique User Tracking Diagramm	18
Abbildung 10 - Eventtracking Diagramm.....	19
Abbildung 11 - Leserichtung (schematisch dargestellt)	20
Abbildung 12 - Spielfeldauswahl: Rotierende Darstellung.....	21
Abbildung 13 - Spielfeldauswahl: Cover Flow Darstellung.....	22
Abbildung 14 - Spielfeldauswahl: Table Sheet Darstellung.....	22
Abbildung 15 - Spielfeldauswahl: Release Version von Quadfender	23
Abbildung 16 - Google Analytics, Kongregate, Übersicht	25
Abbildung 17 - Google Analytics, Kongregate, Spielfeldauswahl	26
Abbildung 18 - Google Analytics, GameJolt, Übersicht	26
Abbildung 19 - Google Analytics, GameJolt, Spielfeldauswahl	27
Abbildung 20 - Google Analytics, Newgrounds, Übersicht.....	27
Abbildung 21 - Google Analytics, Newgrounds, Spielfeldauswahl	28

TABELLENVERZEICHNIS

Tabelle 1 - Strategic Defense Games & Spieleportale.....	9
Tabelle 2 - Ergebnisse der Distributionsplattformen	28
Tabelle 3 - Spielfeldauswahl (Anzahl und Häufigkeit)	29

1 EINLEITUNG

Allgemein kann man sagen, dass Tower Defense Spiele leicht zu erlernen sind, da ihre Komplexität überschaubar ist, aber dass es trotzdem schwer ist, sie zu meistern. Sie bieten eine hohe taktische Tiefe, einen raffinierten Spielaufbau sowie eine ständige Herausforderung, sowohl für *Casualspieler*¹ als auch für Langzeitspieler bzw. *Hardcorespieler*². Auf Grund dieser durchaus gegensätzlichen Zielgruppe müssen Tower Defense Spiele bzw. Strategic Defense Games, wie der wissenschaftliche Oberbegriff lautet, sehr ausgereift und gut ausbalanciert sein. Das Interesse dieser beiden Spielergruppen kann man mit einem simplen Spielprinzip und schnell zu verstehenden Mechaniken wecken, aber durch den ansteigenden Schwierigkeitsgrad auch eine kompetitive Atmosphäre schaffen, in der nicht jeder jedes Ziel auf Anhieb - oder generell - erreichen kann. Die meisten Strategic Defense Games (nachfolgend mit SDG abgekürzt) unterscheiden sich mehr durch die Kartenauswahl, das teils komplexe Layout der Spielfelder und weniger durch die grundlegenden Spielmechaniken. Andere Genrevertreter setzen auf wenige, simpel aufgebaute Spielfelder und legen den Fokus mehr auf komplexe Spielmechaniken und ausgefallene Features. Da der Hauptfokus bei den meisten Gamedesignern auf der Entwicklung neuer, interessanter und innovativer Spielmechaniken liegt, möchte ich mich in dieser Arbeit mit dem Layout der Spielfelder auseinandersetzen.

Die Hypothese dieser Arbeit lautet, dass die Spielfelder und das Layout hinter den Mechaniken eine untergeordnete Rolle spielen, da sowohl auf einfachen als auch auf komplexen Spielfeldern eine hohe taktische Tiefe und Komplexität erreicht werden kann, obwohl die Spielmechaniken unverändert bleiben.

Um diese Hypothese zu untersuchen, wird ein Prototyp eines SDGs entwickelt. Die Spielmechanik wird simpel ausfallen und die grundlegenden Features eines SDGs enthalten, wie z. B. erweiterbare, platzierbare Verteidigungsanlagen, Einheiten mit verschiedenen Eigenschaften und einen ansteigenden Schwierigkeitsgrad. Des Weiteren wird ein Wellenmodus angestrebt, der im Zweifelsfall jedoch durch einen simpleren Spielmodus ersetzt werden kann. Es werden sechs Spielfelder bereitgestellt, welche durch bereits veröffentlichte, erfolgreiche Spiele inspiriert wurden und an die Mechanik von *Construct2*³ adaptiert werden. Hierdurch soll ein indirekter Bezug zu den auf dem Spielmarkt positionierten Titeln hergestellt werden können.

Dieser Spielprototyp wird über die API mit einer Analyseplattform verbunden, so dass die Spieldaten erfasst und ausgewertet werden können. Die innerhalb eines vordefinierten Zeitraums erhaltenen Daten dienen der Auswertung der Hypothese und überprüfen, ob eine fundierte Aussage getroffen werden kann oder ob die getroffenen Annahmen haltlos sind bzw. sich neue Erkenntnisse gewinnen lassen.

¹ Spieler mit mehreren kurzen Sessions

² Spieler, der kompetitive und komplexe Spiele bevorzugt

³ HTML5-Entwicklungsumgebung für Computerspiele; <https://www.scirra.com/construct2>

1.1 GLIEDERUNG UND HERANGEHENSWEISE

Während der Thesis wird zunächst das Genre der Strategic Defense Games näher beleuchtet und nachfolgend die Spielmechaniken und grundlegenden Prinzipien genauer dargestellt. Hierbei wird auf die einzelnen Elemente der Strategic Defense Games eingegangen und insbesondere die Spielfelder und deren Layout im Detail betrachtet und erläutert.

Im Anschluss werden die vier bekanntesten bzw. erfolgreichsten Genrepräsentanten sowie zwei weitere Titel, die unter anderem zu dieser Thesis inspirierten, genauer betrachtet und ihre Kernmechaniken erläutert. Der Fokus hierbei liegt auf der Spielfelddarstellung der verschiedenen Spiele, damit anschließend die Anforderungen an den eigenen Prototypen bestimmt werden können. Hierzu gehört auch die Wahl der Schnittstelle zur Datenerfassung und die Auswahl der Distributionskanäle bzw. -plattformen.

Sobald alle Anforderungen an den Prototypen definiert sind, wird dieser in Construct2 umgesetzt. Hierbei muss vor allem die Datenerfassung frühzeitig getestet und die einwandfreie Funktionsweise sichergestellt werden, da dies die Basis der späteren Analyse und somit den Kern der Thesis darstellt.

Nach Abschluss des Prototypenbaus wird das Spiel auf den vorher festgelegten Distributionsplattformen veröffentlicht, so dass die Datenerfassung beginnen kann. Es wird sehr genau darauf geachtet, dass keine Informationen zur Veröffentlichung in den privaten Kommunikationskanälen erscheinen und dass niemand, der eine persönliche Beziehung zum Autor hat, von der Veröffentlichung erfährt, solange die Datenerfassung nicht abgeschlossen ist. Auf diesem Wege soll sichergestellt werden, dass die Zielgruppe und die gewonnenen Daten möglichst unverfälscht vorliegen und nicht durch persönliche Anliegen verwässert werden.

Abschließend werden die gesammelten Daten analysiert und in Hinsicht auf die Überprüfung der getroffenen Hypothese ausgewertet. Nach der Auswertung werden die gewonnenen Erfahrungen in einem Fazit zusammengefasst.

1.2 VERWANDTE ARBEITEN

Das Forschungsgebiet der Strategic Defense Games ist, wie die Recherchen zeigen, durchaus ergiebig und bietet vor allem bei der Überprüfung und Erarbeitung von Algorithmen zur *Computational Intelligence* viele Ansatzmöglichkeiten [04]. Laut *Avery et al.* [04] liegt der Nutzen von Tower Defense Spielen darin, gut zugängliche, einfache Benchmarks zu entwickeln, um die Performance von *CI*⁴ Algorithmen zu überprüfen und um im Anschluss rückwirkend das Tower Defense Gamedesign und die Entwicklungspipeline zu verbessern, indem neue Spielformen und angepasste Entwicklungsprozesse getestet und etabliert werden.

Bei *Rummel* [03] wird der Ansatz eines SDGs, dass der Spieler die Verteidigungsanlagen positioniert und gegen eine vordefinierte Abfolge von Gegnern trifft, umgedreht. Der Spieler wird durch eine künstliche Intelligenz (K.I.) ersetzt, um die Stärken und die Lernstrategien von realen Personen im Gegensatz zu einer künstlichen Intelligenz vergleichen zu können. Hierbei geht es um die optimale Platzierung von Verteidigungsanlagen, wobei die Auswahl auf einen einzelnen Turm ohne Upgrades beschränkt wurde. Die K.I. platziert pro Versuchsdurchlauf an mehreren Stellen einen Turm und misst die erreichten Gegnerabschüsse bis zum Spielende. Jeder Turm wird nach einigen Spielrunden auf seine Effizienz und damit das Kosten-Nutzen-Verhältnis hin untersucht und anhand seiner Performance in einem internen Ranking eingeordnet. Obwohl die Forschungen zu dem Ergebnis führten, dass eine optimale Platzierung der Türme nach 55-60 Durchläufen erreicht werden konnte, kommt der Autor zu der Feststellung, dass es noch zu große Unterschiede zwischen der K.I. und einem realen Spieler gibt, die auf die vielen Einschränkungen bei der K.I.-Entwicklung zurückzuführen sind.

In Bezug auf die Analyse von Spielfeldern bzw. deren Einfluss auf die Spielweise und Aktivität von Spielern konnten bisher keine Forschungen oder Thesen ermittelt werden. Aus diesem Grunde muss davon ausgegangen werden, dass kaum Literatur zu diesem speziellen Themenkomplex vorhanden ist.

⁴ Computational Intelligence

1.3 HYPOTHESE

In *Avery et al.* [04] stellen die Autoren unter anderem die These auf, dass die Spielfelder zwar die Kernelemente eines SDGs sind, jedoch der Spieler nicht aufgrund ihrer Variationen oder den eigentlichen Spielmechaniken das Spiel verlässt, sondern dass die Auswahl der Spielfelder in einem SDG zu beschränkt sei und dem Spieler weitere Entdeckungsmöglichkeiten fehlen, die sein Interesse steigern könnten. Eine automatische Spielfeldgenerierung ist somit für die Autoren die Lösung für das Problem der schwindenden Motivation und einer damit verbundenen hohen *Churnrate*⁵.

Im Gegensatz zu *Avery et al.* [04] wird in dieser Thesis die Hypothese aufgestellt, dass nicht die Spielfelder die Kernelemente sind, die den Spieler zum Weiterspielen oder Aufhören bewegen, sondern dass eine hohe Anzahl an Spielfeldvariationen sogar unnötig ist, da die Mehrheit der Spieler einen bestimmten Spielfeldtyp bevorzugen und viele Auswahlmöglichkeiten nicht relevant für die Spielmotivation sind.

Um diese Hypothese zu überprüfen müssen entweder genug fremde, nachvollziehbare und vertrauenswürdige Datensätze bzw. Quellen vorliegen oder eine eigene Analyse durchgeführt werden. Wie im vorherigen Abschnitt bereits erwähnt, liegt kaum Literatur für den Themenkomplex der SDGs vor und bezüglich einer Spielfeldhäufigkeitsanalyse konnten bisher keine Datensätze und wissenschaftliche Arbeiten gefunden werden. Aus diesem Grund wird sich diese Master-Thesis mit der experimentellen Analyse von Spielfeldern beschäftigen. Es wird zunächst ein Prototyp entwickelt, welcher über Tools zur Datenanalyse die Häufigkeit der gewählten Spielfelder erfassen und auswerten kann. Nachdem über diverse Distributionskanäle ausreichend Datensätze zur Verfügung stehen, werden die Daten ausgewertet und auf ihre Häufigkeitsverteilung hin untersucht.

Diese Datensammlung mit anschließender Analyse der gewonnenen Datensätze soll eine Möglichkeit bieten, die getroffene Hypothese klar und eindeutig zu widerlegen oder sie im Gegenzug zu bestätigen. In jedem Fall kann eine hoffentlich eindeutige Aussage über Sinnhaftigkeit und Aussagekraft einer solchen Datenanalyse getroffen werden, mit einem Verweis auf mögliche spätere Forschungsgebiete bzw. andere Forschungsansätze.

⁵ Abwanderungsquote; Anteil der Spieler, die das Spiel verlassen und nicht für eine weitere Spielsitzung wiederkommen

2 HAUPTTEIL

2.1 STRATEGIC DEFENSE GAMES

In diesem ersten Abschnitt werden zunächst grundlegende Eigenschaften von Strategic Defense Games vorgestellt, näher erläutert und auf ihre Relevanz für die praktische Untersuchung hin betrachtet. Neben den Grundlagen wird detailliert auf die Kernmechaniken und vor allem das Spielfelddesign eingegangen. Nachdem die Grundlagen auf diesem Wege geklärt wurden, werden die Features des Genres anhand einiger Beispiele verdeutlicht.

2.1.1 Überblick

Ein Strategic Defense Game ist ein Strategiespiel, welches sich auf das Sammeln einer Ressource und das taktische Platzieren von Verteidigungsanlagen (nachfolgend umgangssprachlich auch Türme genannt) fokussiert. Der grundlegende Spielablauf besteht hauptsächlich aus einem Spieler, der die Türme kauft, platziert, erweitert oder verkauft, und den Gegnern, die versuchen von einem Startpunkt an einen Endpunkt zu gelangen, während sie von den Türmen beschossen werden. Sollte ein Turm einen Gegner zerstören, so erhält der Spieler eine Ressource, die dem Schwierigkeitsgrad des Gegners entspricht. Mit Hilfe dieser Ressource kann der Spieler neue Türme kaufen. Die Türme übernehmen die Gegnerbekämpfung automatisch und benötigen wenige bis gar keine zusätzliche Aktion vom Spieler. Hat der Spieler eine vorgegebene Anzahl von Gegnern zerstört oder erreichen zu viele Gegner ihren Zielort, ist das Spiel zu Ende. Strategic Defense Games haben sich als ein herausfordernder, süchtig machender und spaßiger Zeitvertreib im Bereich der Web- und Mobilespiele etabliert. [04]

Nachfolgend ist der grundlegende Gameloop eines SDGs schematisch dargestellt:

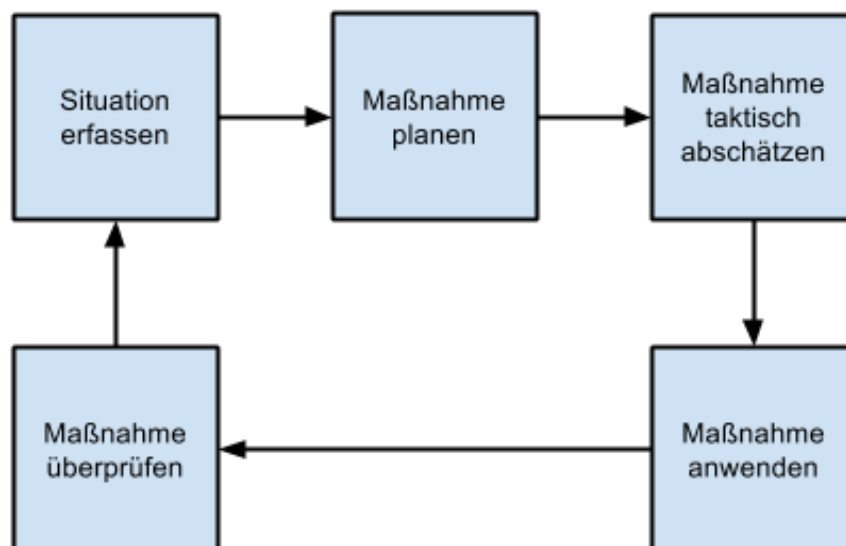


Abbildung 1 - Strategy Defense Game Gameloop

2.1.2 Kernelemente

Während die speziellen Elemente sich bei jedem Spiel stark unterscheiden können, gibt es Standardfeatures, die in fast jedem SDG vorhanden sind. Dies sind z. B. Wellen- und/oder Endlosmodus, Erweiterbarkeit der zu setzenden Einheiten, globale Upgrades, Freischaltung von *Content*⁶ und mittlerweile häufig auch der Storymodus.

Im folgenden Abschnitt wird zunächst genauer auf die einzelnen Elemente eines SDGs eingegangen, um anschließend einige aktuelle Beispiele von SDGs, hauptsächlich im Hinblick auf das Spielfeldlayout, zu betrachten.

Platzierbare Objekte

Das eigentliche Hauptfeature eines SDGs sind die vom Spieler platzierbaren Einheiten, oftmals repräsentiert durch Türme. Diese Objekte bzw. Türme unterscheiden sich in diversen Eigenschaften, wie z. B. Schaden, Reichweite, Schussfrequenz, Baukosten, Rüstungsdurchbrechung, Areal Schaden und durch eventuelle Stärken und Schwächen gegenüber bestimmten Gegnertypen.

Die Türme werden vom Spieler möglichst taktisch auf dem Spielfeld platziert, so dass eine maximale Effektivität pro Einheit erreicht wird und ein strategischer Einsatz garantiert ist. Bei der Platzierung muss allerdings nicht nur das Spielfeld und die Gegnerbewegung berücksichtigt werden, sondern auch eine eventuelle Synergie zu den benachbarten Türmen. So kann z. B. eine Reihe von Türmen die gegnerischen Einheiten verlangsamen und diese so in einem Bereich konzentrieren, damit ein anderer Turm mit Gebietsschaden, durch die geschickte Ausnutzung der Synergieeffekte, möglichst viele gegnerische Einheiten pro Angriff treffen kann.

In den meisten Fällen können einmal platzierte Verteidigungseinheiten nicht umplatziert werden, allerdings verfügt fast jedes SDG über die Möglichkeit, die gebauten Einheiten auch wieder zu einem geringeren Preis zu verkaufen. Somit hat der Spieler stets die Möglichkeit eventuelle Spielfehler auszugleichen oder seine Strategie an die bevorstehenden Gegner anzupassen.

Upgrades

Als Zusatz zum Platzieren und Verkaufen der Verteidigungseinheiten hat der Spieler in den meisten Fällen die Möglichkeit, diese Einheiten im Spielverlauf zu verbessern. Hierbei kann der Spieler die positionierten Einheiten durch Zugabe von Spielwährung im Funktionsumfang aufsteigen lassen und die Effektivität der einzelnen Einheiten deutlich steigern. Oftmals sind diese Erweiterungen keine einmalige Option, sondern jede Einheit kann mehrere Male erweitert werden. Je nach Spieletitel erhalten diese Einheiten auch neue Fähigkeiten oder Angriffsmuster. Teilweise können auch mehrere Einheiten in einer Art Symbiose zusammengeführt werden und vereinen die Eigenschaften und Angriffsmuster der einzelnen Einheiten in einer Einheit. Die Kosten für die Erweiterungen steigen bei den meisten Titeln exponentiell, da das Erweitern der Einheiten im Gegensatz zum stark limitierten Platzangebot auf dem Spielfeld keine zusätzlichen Flächen benötigt. Sollte die komplette Spielfeldfläche verbaut sein, ist ein Fortschritt demnach nur noch über die Erweiterung der einzelnen Einheiten möglich.

⁶ Inhalt; Häufig gebrauchter Begriff in Computerspielen

Gegner

Die gegnerischen Entitäten bilden die unmittelbare Bedrohung für den Spieler. Sollten zu viele Gegner ihren Zielort erreichen, verliert der Spieler oftmals direkt das Spiel und muss erneut beginnen. Die höchste Priorität des Spielers besteht folglich in der möglichst effizienten und raschen Vernichtung der aktuell präsenten Gegner. Sollte ein Gegner durch die Türme des Spielers vernichtet worden sein, so erhält der Spieler in den meisten Fällen eine bestimmte Anzahl an Ressourcen. Je nach Schwierigkeitsgrad des Gegners werden zudem unterschiedlich viele Ressourcen ausgeschüttet. Viele SDGs verfügen zusätzlich über Bossgegner, Gegner mit mehr Lebenspunkten oder sogar Spezialfähigkeiten, die im Falle ihres Ablebens neben vielen Ressourcen auch spezielle Ressourcen oder Gegenstände für Upgrades für den Spieler bereithalten.

Gegner verfügen entsprechend ihrer Schwierigkeit über verschiedene Eigenschaften und unterscheiden sich maßgeblich in Bezug auf Trefferpunkte, Geschwindigkeit, Rüstung und Resistenzen gegenüber den platzierten Türmen. Allerdings verfügen die Gegner in den wenigsten Spielen über die Möglichkeit, die Türme des Spielers zu beschädigen.

Spielmodi

Bei den Spielmodi gibt es grundsätzlich drei Hauptaspekte bzw. -modi, die immer wieder ihren Weg in die Spieletitel finden. Zum einen ist es der Storymodus, bei dem der Spieler einen vorgefertigten Ablauf „nachspielt“, welcher stets gleich ist und beim erneuten Durchspielen oftmals keinerlei Neuerungen bereithält. Die einzige Variante hierzu ist das *New Game+*, bei dem zwar der bekannte Storymodus erneut gespielt wird, sich jedoch die bekannten Gegner, Angriffsmuster und Rahmenbedingungen, wie z. B. zur Verfügung stehende Ressourcen oder Einheiten, zum ersten Durchgang wenig bis stark unterscheiden.

Der zweite häufig vertretene Modus ist der Wellenmodus, bei dem der Spieler Wellen von Gegnern standhalten muss. Die Wellen sind meistens zeitgesteuert und die nachfolgenden Wellen stehen nicht in Zusammenhang zum Überleben der aktuellen Welle. Sollte die Verteidigung des Spielers nicht standhalten und Gegner das Ziel erreichen, werden diese erneut an den Startpunkt gesetzt. Die zurückgesetzten Einheiten überlappen sich in diesem Fall mit der nachfolgenden Welle. Die Gegnertypen in den einzelnen Wellen haben stets die gleichen Eigenschaften, so dass sich der Spieler mit jeder neuen Welle einer anderen Herausforderung gegenüber sieht.

Oftmals gibt es zusätzlich auch einen Endlosmodus, der auf dem Wellenmodus basiert, bei dem die Wellen in der Schwierigkeit oder der Masse der Einheiten einfach stetig erhöht werden. Da jedoch die Verteidigungsanlagen sowohl durch die Erweiterungen als auch durch den verfügbaren Baulandplatz begrenzt sind, ist der Endlosmodus nur in der Theorie endlos, da die Möglichkeiten des Spielers beschränkt sind.

Globale Upgrades und Fähigkeiten

Ein häufiges, aber nicht grundlegendes Feature in vielen SDGs sind die globalen Upgrades und Fähigkeiten, welche dem Spieler während des gesamten Spiels zur Verfügung stehen. Die globalen Upgrades beeinflussen oftmals die zu platzierenden Verteidigungsanlagen, indem die Reichweite, die Durchschlagskraft, der Gebietsschaden oder die Kosten beeinflusst werden. Daneben gibt es noch Upgrades, welche sich z. B. auf die Lebenspunkte des Spielers oder die erhaltenen Belohnungen für eine Gegnereliminierung beziehen.

Bei den Fähigkeiten gibt es zum Teil gravierende Unterschiede zwischen den Spielarten. Manche Spiele, wie z. B. *Kingdom Rush*, setzen sehr stark auf die Interaktion zwischen Spieler und Umwelt mit Hilfe der Fähigkeiten, während bei anderen Spielen die Zusatzfähigkeiten eine untergeordnete Rolle einnehmen.

Contenterweiterung

Im Allgemeinen ist die Contentfreischaltung ein *Gating*⁷-Prozess, der den Spieler mit kontinuierlicher Spielteilnahme über die Bereitstellung von neuen Inhalten extrinsisch belohnt. Diese Belohnungen beinhalten z. B. neue Gebiete und Spielfelder, bessere und neue Fähigkeiten sowie *Customizations*⁸, wie beispielsweise neue Skins, Farbschemata oder Ähnliches. Diese Art der Belohnung findet der Spieler hauptsächlich im Mobile Games Bereich, da hier die Grundinhalte in den meisten Fällen kostenlos sind und die Erweiterungen entweder über ein *Timecommitment*⁹ oder über eine monetäre Vergütung für den Spieler erreichbar sind.

2.1.3 Spielfelddesign

Die Spielfelder, auch als Terrain, Karte oder Level bezeichnet, bilden die Grundlage für den Spieler, um Entscheidungen zum strategischen Positionieren der Verteidigungsanlagen zu treffen. Im Bereich der klassischen SDGs besteht das Spielfeld aus einer unbebaubaren Fläche, dem Weg, dem die Gegner vom Start zum Ziel folgen, und einer bebaubaren Fläche, auf der der Spieler seine Verteidigung errichten kann. Je nach Spiel unterliegt diese Fläche den Einschränkungen, dass bestimmte Areale nur durch spezielle Verteidigungsanlagen bebaut werden können. Durch diese Art der Restriktion kann eine ansonsten sehr einfach zu bewältigende Karte in der strategischen Tiefe deutlich erschwert werden und den Spieler vor neue Herausforderungen stellen.

In dieser Master-Thesis werden vor allem die klassischen Strategic Defense Games zur Analyse der Spielfeldstrukturen herangezogen. Der Unterschied zwischen den klassischen und den neueren Varianten ist der gewählte Schwerpunkt bzw. spielmechanische Fokus. Klassische Spiele dieses Genres überlassen dem Spieler die freie Wahl, an welcher Stelle, welche Verteidigungsanlage platziert wird, während die neueren Genrevertreter oftmals festgelegte Spielfeldpositionen zur Auswahl stellen und der Spieler nur die Art der Verteidigungsanlage wählen kann. Hierbei wird der Fokus stärker auf Fähigkeiten, zusätzliche Aktionsmöglichkeiten und die geschickte Ausnutzung der gegnerischen Einheitenaufstellung gelegt.

Bei den klassischen Strategic Defense Games spielt aufgrund der frei wählbaren Positionierung der Verteidigungsanlage der Spielfeldaufbau und die Wegführung eine besondere Rolle. Die vordefinierten Pfade für die gegnerischen Einheiten unterscheiden sich in ihrer Komplexität und Länge und definieren hierdurch die Komplexität und Schwierigkeitsstufe des Spielfeldes. Da jede Verteidigungsanlage nur über einen bestimmten Aktionsradius verfügt, ist die Aktivitätszeit zur Gegnerbekämpfung durch die maximale Weglänge, die innerhalb des Aktionsradius liegt, definiert. Demzufolge ist ein Spielfeld, welches über nur einen geraden Weg verfügt, mit die größte Herausforderung, da jede Verteidigungsanlage nur eine Möglichkeit zur Gegnerbekämpfung bekommt und die Gegner nicht erneut vorbeigeführt werden. Im Gegensatz hierzu sind verschlungene Pfade mit vielen Kurven leichter

⁷ Von engl. *Gate*: Gatter, Tor; Art der Regulierung des Spielerfortschritts durch freischaltbare Inhalte

⁸ Anpassungen, die oftmals nur kosmetischen Nutzen haben

⁹ Zeitliche Verpflichtung oder Aufwendung

zu bewältigen, da unter Umständen eine einzelne Verteidigungsanlage mehrere Wegstrecken mit ihrem Aktionsradius abdecken kann.

2.1.4 Beispiele

Im Bereich der Strategic Defense Games gibt es unzählige Genrevertreter. Um die Liste mit Beispielen zu verknappen, wurde eine Auswahl an Webseiten zu diesem Thema herangezogen und die jeweiligen vier meistgespielten und höchstbewerteten Spiele ausgewählt.

Tabella 1 - Strategic Defense Games & Spieleportale

	<i>Defense-Tower</i> ¹⁰	<i>Addictinggames</i> ¹¹	<i>Armorgames</i> ¹²	<i>Kongregate</i> ¹³
<i>Cursed Treasure (IP)</i>	x		x	x
<i>Gemcraft (IP)</i>	x		x	x
<i>Bloons TD (IP)</i>	x	x		x
<i>Incursion 2</i>			x	
<i>Defend your Nuts</i>		x		
<i>Ancient Planet</i>		x		
<i>Kingdom Rush (IP)</i>		x	x	x
<i>Demon Rift</i>	x			

Aus den Übereinstimmungen der Hitlisten ist die endgültige Auswahl der Beispiele entstanden. Auch wenn die Auswahl nicht die Spiele *Flash Element TD*¹⁴ und *Vector TD*¹⁵ enthält, so werden diese in der folgenden Übersicht dennoch erscheinen, da diese Spiele die Inspiration für die experimentelle Analyse sind und erwähnt werden sollten, da die Spielfelder besonders ins Auge stechen.

Kingdom Rush IP

Das *Kingdom Rush*¹⁶ Franchise beinhaltet *Kingdom Rush*, *Kingdom Rush Frontiers* und *Kingdom Rush Origins* und ist momentan das am häufigsten gespielte Tower Defense Franchise. Der erste Teil der Reihe, *Kingdom Rush*, wurde 2011 von den Ironhide Studios auf Armorgames herausgegeben. Eine Umsetzung für iOS folgte kurz darauf und die Umsetzung für Android und Steam 2013. Die Fortsetzung *Kingdom Rush Frontiers* erschien 2013 auf iOS und Android und der momentan letzte Teil der Serie, *Kingdom Rush Origins*, 2014.

Bei dieser Variante des Strategy Defense Games kann der Spieler verschiedene Türme in vordefinierten Bauplätzen entlang einer Route bauen. Auf dieser Route versuchen verschiedene Gegner, in unterschiedlichen Variationen und Gruppenkonstellationen, von ihrem Startpunkt zum Ziel zu gelangen. Dieses muss der Spieler durch die taktische Nutzung der Bauplätze zu verhindern versuchen. Zusätzlich hat der Spieler die Möglichkeit, auf Spezialfertigkeiten in Form von Zaubersprüchen und

¹⁰ Top 40 Tower Defense Games (<http://www.defense-tower.de/charts.html>)

¹¹ Tower Defense Games – last week (<http://www.addictinggames.com/tag/tower-defense-games.jsp>)

¹² Top 10 Tower Defense Games (<http://armorgames.com/news/top-10-tower-defense-games>)

¹³ Tower Defense Spiele (<http://www.kongregate.com/de/tower-defense-games>)

¹⁴ Flash Element TD: <http://www.newgrounds.com/portal/view/362047>

¹⁵ Vector TD: <http://www.candystand.com/play/vector-td>

¹⁶ Kingdom Rush Webseite der Ironhide Studios: <http://www.kingdomrush.com/>

Verstärkungstruppen zurückzugreifen. Im dritten Teil der Serie, Kingdom Rush Origins, kann der Spieler nicht nur die Heldeneinheit kontrollieren, sondern auch über dessen Spezialfähigkeiten verfügen.

Die Spielfelder sind durch den vordefinierten Pfad und die festgelegten Bauplätze stets ähnlich vom Spielablauf und bieten nur durch die Verteilung der Bauplätze und die Anzahl der Pfade zum Ziel eine abwechselnde Herausforderung. Da das Spiel in Form einer Kampagne durch die diversen Level führt, ist ein Vergleich in Hinblick auf die Spielfeldauswahl nicht möglich.



Abbildung 2 - Spielfelder Kingdom Rush IP (Screenshots)

Flash Element TD

Flash Element TD ist das erste Tower Defense Game in Flash und wurde von David Scott Anfang 2007 entwickelt. Es gibt bereits sieben verschiedene Türme und freischaltbare, globale Upgrades. Allerdings verfügt die aktuelle Version über nur ein Spielfeld.

Das Layout stellt eine vereinfachte Karte der *Warcraft III* Modifikation *Element TD* dar. Die Gegner bewegen sich von dem linken oberen Kartenrand auf dem braunen Weg bis zum rechten oberen Kartenrand. Pro Gegner, der die Karte wieder verlässt, bekommt der Spieler einen Lebenspunkt abgezogen. Die Verteidigungstürme kann der Spieler auf der grünen Fläche platzieren, wobei die Möglichkeiten durch die zwingend rasterförmige Anordnung beschränkt sind.

Insgesamt gibt es auf dieser Karte zwölf 90°-Kurven, wodurch der Spieler viele Möglichkeiten hat, die Türme taktisch zu platzieren, und so immer neue Konzepte und Ideen ausprobieren kann, um möglichst weit im Spiel voranzukommen.



Abbildung 3 - Spielfeld Flash Element TD (Screenshot)

Vector TD

Vector TD ist ein weiteres, von David Scott entwickeltes, Tower Defense Game in Flash und wurde 2007 auf CandyStand veröffentlicht. Wie Flash Element TD ist auch dieses Spiel ein Ableger der klassischen Tower Defense Spielmechanik. Der Spieler hat die Möglichkeit, vier verschiedene Türme zur Abwehr der sechs unterschiedlichen Gegnertypen zu bauen. Die Türme können aufgewertet werden und es stehen noch verschiedene globale Upgrades bereit, die nach der erfolgreichen Abwehr einer Boss-Welle freigeschaltet werden.

Die Spielfelder sind in drei verschiedene Schwierigkeitsgruppen unterteilt und lassen sich zu Beginn jedes Spieldurchlaufs auswählen. Es stehen insgesamt zwei *Beginner Maps*, drei *Normal Maps* und drei *Extreme Maps* zur Verfügung. Die Schwierigkeitsgrade unterscheiden sich in der Anzahl der Ein- und Ausgänge, in der Anzahl der Kurven und dem Platz, der zur Bebauung durch Türme zur Verfügung steht.

Vector TD ist, was die Spielfeldvielfalt und die Gestaltung der Level betrifft, das Spiel, welches am meisten Einfluss auf die Entstehung der Master-Thesis und der gestellten Hypothese ausgeübt hat.

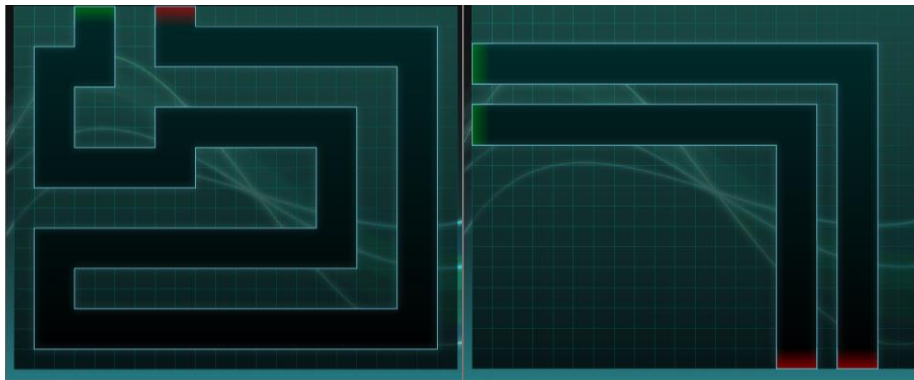


Abbildung 4 - Vector TD Spielfelder (Screenshots)

Bloons Tower Defense IP

Das *Bloons Tower Defense*¹⁷ Franchise umfasst mittlerweile fünf Versionen und wurde vom Studio NinjaKiwi entwickelt. Seit Version 1, erschienen im Jahre 2007, werden die Bloons TD Spiele kontinuierlich weiterentwickelt, angepasst und auf diversen Distributionsplattformen herausgegeben. Die Publishingkanäle umfassen nicht nur eine Web-Version und die Releases für iOS und Android, sondern auch das Playstation Portable System und den Nintendo DSi.

Bei Bloons Tower Defense muss der Spieler mit Hilfe von dartpfeilwerfenden Affen, Pfeilwurfmaschinen und anderen Gadgets versuchen Wellen von diversen Luftballons daran zu hindern, den vordefinierten Kurs zurückzulegen.

Die verschiedenen Spielfelder sind auch hier in verschiedene Schwierigkeitsgrade unterteilt: *Beginner*, *Intermediate*, *Advanced* und *Expert*. Je nach Version des Spiels unterscheiden sich sowohl die Spielfelder als auch die Anzahl der Spielfelder in den unterschiedlichen Schwierigkeitsstufen. Während bei vielen anderen Strategy Defense Games der Fokus auf 90°-Kurven liegt, gibt es bei Bloons TD viele kreisförmige Kurven und ovalgeformte Streckenabschnitte. Damit die Positionierung der Abwehrmechanismen an das Spielfeldlayout angepasst werden kann, sind die Abwehreinheiten frei platzierbar und unterliegen keinem Raster bei der Positionierung. Hieraus ergibt sich eine höhere

¹⁷ NinjaKiwi Webseite mit den Bloons TD Spielen: <http://ninjakiwi.com/Games/Tower-Defense>

Schwierigkeitsstufe, aber bietet auch die Möglichkeit zur Perfektion, da versucht werden kann, die zur Verfügung stehende Fläche möglichst platzsparend und effizient auszuschöpfen.



Abbildung 5 - Spielfelder Bloons TD IP (Screenshots)

Gemcraft IP

Das von Game in a Bottle 2008 veröffentlichte *Gemcraft Chapter 1: The forgotten* bildet den Grundstein der mittlerweile aus vier Teilen bestehenden *Gemcraft*¹⁸ IP. Bei *Gemcraft* bekämpft der Spieler herannahende Gegenschwärme durch das taktische Platzieren von Türmen und verschiedenen kombinierbaren Edelsteinen. Die Türme in *Gemcraft* sind grundsätzlich nicht mit der Möglichkeit ausgestattet, Gegner direkt zu bekämpfen, sondern müssen zunächst mit Edelsteinen bestückt werden. Die Edelsteine erschafft der Spieler über die Hauptressource und kontrolliert so die Bestückung der jeweiligen Türme, wobei jeder Edelstein über andere Angriffsformen und -stärken verfügt. Sollte der Spieler über genügend Edelsteine im Lager verfügen, so können die Edelsteine kombiniert werden, um noch mächtigere Werte zu erhalten.

Die Spielfelder orientieren sich stark an den klassischen SDGs mit rechtwinkligen Abzweigungen, meistens einer einzelnen Wegstrecke, wenn auch mit mehreren Startpunkten und freien Bebauungsflächen für die Türme. Auf den meisten Spielfeldern sind von Beginn an einige Türme gesetzt, die der Spieler nutzen kann, während neue Türme nicht frei platziert werden können, sondern sich an die rasterförmige Platzierung halten müssen.



Abbildung 6 - Spielfelder Gemcraft IP (Screenshots)

¹⁸ Game in a Bottle Webseite mit allen Gemcraft-Spielen: <http://gameinabottle.com/>

Cursed Treasure IP

Die *Cursed Treasure*¹⁹ IP basiert auf den zwei Titeln *Cursed Treasure* und *Cursed Treasure 2* des russischen Entwicklerstudios Iriysoft und wurde 2010 mit der ersten Version auf Armorgames veröffentlicht. In den Grundsätzen ist es ein Strategic Defense Game, welches allerdings ein paar Auflagen an den Spieler stellt. Sobald ein Gegner das Ziel erreicht, wird dieser einen der Edelsteine des Spielers stehlen und versuchen, diesen erneut den Weg zurückzutransportieren. Sollte der Gegner vernichtet werden, so fällt der Edelstein zu Boden und der nächste Gegner kann versuchen, ihn von diesem Punkt aus in Sicherheit zu bringen. Da der Spieler bereits Minuspunkte bekommt, sobald ein Gegner den Edelstein berührt, muss sich auf die Verteidigung des Edelsteinlagers konzentriert werden, ohne dabei die übrigen Wege außer Acht zu lassen, sollte doch ein Gegner durch die Verteidigung gelangen. Zudem gibt es die Restriktion, dass die Verteidigungsanlagen nur auf bestimmtem Terrain gebaut werden können, so dass der Spieler auf jedem Spielfeld seine Strategie an die Umgebung bzw. das Terrain anpassen muss.

Das Spielfeldlayout ist bis auf die Terrainunterschiede recht klassisch mit 90°-Kurven und einer gerasterten Bebauungsfläche. Davon abgesehen sind die Spielfelder sehr viel offener und bieten den gegnerischen Einheiten einen erhöhten Aktionsradius, welches wiederum den für den Verteidigungsbau zur Verfügung stehenden Platz stark einschränkt. Allerdings bleibt *Cursed Treasure* damit in der Komfortzone der klassischen Tower Defense Games.



Abbildung 7 - Spielfelder *Cursed Treasure IP* (Screenshots)

¹⁹ Newgrounds-Seite von Iriysoft: <http://iriysoft.newgrounds.com/games/>

2.2 PRAKTISCHE UMSETZUNG

2.2.1 Konzept

Damit die aufgestellte Hypothese möglichst zutreffend überprüft werden kann, wird auf externe Quellen und Datensätze verzichtet. Zudem ist es leider nicht möglich, auf die eventuell vorhandenen Datensätze der einzelnen Spieleentwickler bzw. Publisher zuzugreifen, sofern diese überhaupt die benötigten Daten erfassen. Es ist daher nötig, einen Spieleprototypen zu erstellen, der alle Anforderungen eines Strategic Defense Games erfüllt und zudem mit einer Schnittstelle ausgestattet ist, die es ermöglicht, Daten für die Analyse zu gewinnen. Die gewonnenen Daten werden anschließend analysiert und ausgewertet. Hierbei steht die Überprüfung der gestellten Hypothese im Vordergrund.

2.2.2 Zielgruppenabgrenzungskriterien

Zu Beginn der Master-Thesis ging aus der Planung eindeutig hervor, dass eine Einschränkung der Zielgruppe dringend benötigt wird. Es wurde davon ausgegangen, dass nur Coregamer, also Spieler, die sich bereits intensiv mit SDGs beschäftigen, diesen Prototyp spielen werden, da für alle anderen Spieler die Thematik und die Umsetzung zu abstrakt wären. Durch die Veröffentlichung auf diversen Plattformen und Portalseiten kann eine Einschränkung der Zielgruppe jedoch nicht gewährleistet werden und erscheint daher wenig sinnvoll.

Folglich wird bezüglich der Zielgruppen keine Unterscheidung getroffen, da diese nicht eindeutig überprüft werden können. Selbst eine an den Spieldurchlauf angeschlossene Nutzerumfrage mit dem Schwerpunkt auf den favorisierten Spielgenres würde zu keinem verlässlichen Ergebnis führen, da es keine Möglichkeit zur Überprüfung des Wahrheitsgehaltes gibt.

2.2.3 Plattformabgrenzungskriterien

Damit der Prototyp in der kurzen Laufzeit von circa einer Woche eine möglichst breite Masse erreichen kann und hierbei keine spezielle Zielgruppe, sondern Spieler vieler unterschiedlicher Genres angesprochen werden, muss der Prototyp auf unterschiedlichen Plattformen zeitgleich online zur Verfügung stehen.

Bei der Auswahl der Plattformen muss sowohl auf die regionale, die nationale und die internationale Erreichbarkeit geachtet werden als auch auf die technische Umsetzbarkeit von HTML5-Spielen. Aufgrund früherer Erfahrungen mit der Plattform Kongregate und der Unterstützung von HTML5-Spielen wurde diese Portalseite als eine Zielplattform festgelegt. Eine hohe internationale Erreichbarkeit wurde bereits über ein früheres Masterprojekt evaluiert und als ausreichend für diese Masterarbeit befunden. Wegen möglicher technischer Probleme und der Risikostreuung wurde eine zweite Plattform gesucht, welche ähnliche Eigenschaften bietet wie Kongregate. Durch die starke Limitierung auf die HTML5-Unterstützung und die geringe Verbreitung der Technologie auf den Spieleportalen gab es zum damaligen Zeitpunkt nur wenige potentiell vielversprechende Alternativen für Kongregate. Somit wurde GameJolt als zweite Distributionsplattform ausgewählt, die sowohl international frequentiert wird, als auch HTML5-Spiele von Indie-Entwicklern unterstützt. Um eine eventuell schwache Resonanz auf die Veröffentlichung des Spiels abzufangen, wurde während der Releasephase des Prototypen zusätzlich eine Version auf Newgrounds veröffentlicht. Dieses Spieleportal ist vor allem bei einer relativ jungen Zielgruppe bekannt und dient nicht nur zur Verbreitung und Veröffentlichung von Spieleprojekten, sondern beinhaltet auch Musik-, Video- und Kunstprojekte.

Weitere Distributionswege sind nicht ausgeschlossen, jedoch durch die technischen Voraussetzungen stark eingeschränkt. Eine mobile Verbreitung über den Playstore von Google oder den Appstore von Apple für Smartphones ist nicht vorgesehen. Dieser Distributionskanal würde starke Anpassungsanforderungen an das Spiel stellen, die in keinem Verhältnis zum Nutzen stehen würden.

2.2.4 Prototyp

Um in einem der Master-Thesis entsprechenden, kurzen zeitlichen Rahmen den Prototypen zu entwickeln, wurde bereits zu Beginn der Arbeit die Entscheidung zugunsten eines vorher erprobten Entwicklungsframeworks getroffen. Der Prototyp basiert auf HTML5 und wird mit Hilfe des Entwicklungsframeworks Construct2 realisiert. Alle Assets und jeder Code werden selbst entwickelt und um geeignete Schnittstellen für die Datenerfassung erweitert. Diese Schnittstellen kommunizieren mit dem Webdienst *Google Analytics*²⁰, der die Daten über sogenannte Ereignisse bzw. Events erfasst und anschließend tabellarisch und optisch aufbereitet und bereitstellt.

Der Prototyp umfasst die meisten Kernelemente eines Strategic Defense Games, die im Abschnitt 2.1.2 genauer erläutert wurden. Als Distributionsplattform wurden *Kongregate*²¹, *GameJolt*²² und *Newgrounds*²³ gewählt, da diese Plattformen einerseits eine etablierte Masse an Konsumenten aufweisen und andererseits die Entwicklung und Veröffentlichung von HTML5-Spielen ermöglichen und unterstützen.

Der Prototyp ist unter folgenden Adressen erreich- und spielbar:

Kongregate: <http://www.kongregate.com/games/nescgames/quadfender>
GameJolt: <http://gamejolt.com/games/strategy-sim/quadfender/62282/>
Newgrounds: <http://www.newgrounds.com/portal/view/656882>

Im Anhang befinden sich weitere Referenzbilder der Release Version und Codebeispiele aus Construct2.

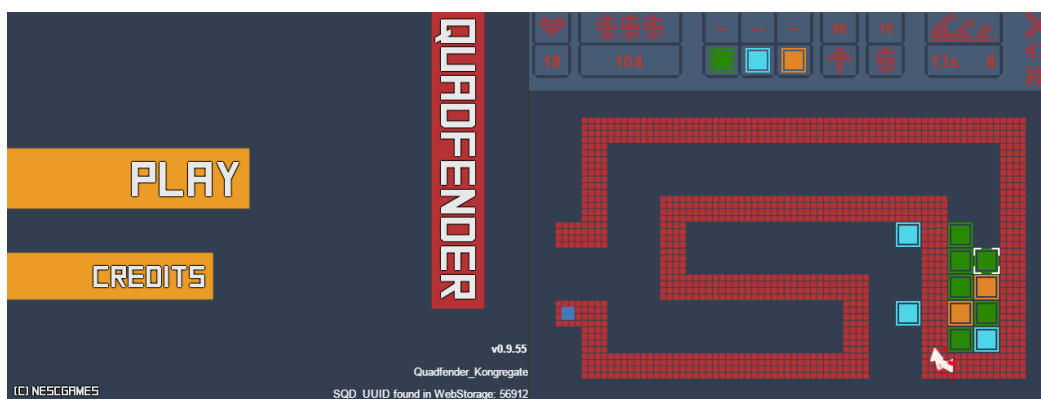


Abbildung 8 – Release Version Quadfender (Screenshots)

²⁰ Webanalyseplattform von Google: https://www.google.com/intl/de_DE/analytics/

²¹ Amerikanisches Spieleportal: <http://www.kongregate.com/>

²² Britisches Indie-Spieleportal: <http://www.gamejolt.com/>

²³ Portal für digitale Eigenkreationen: <http://www.newgrounds.com/>

Features

Der entstandene Prototyp verfügt über die Basisfunktionen eines Strategic Defense Games. Im Detail bestehen diese Funktionen aus:

1. Verteidigungsanlagen (3 Varianten)

Die Verteidigungstürme werden über die Mauseingabe platziert. Um eine gleichmäßige und möglichst einfache Platzierung zu erlauben, wurde das gesamte Spielfeld in 32x32 Pixel große Felder gerastert. Um einen Turm zu bauen, muss der Spieler zunächst ein Feld anwählen, welches dann mit einem Cursor hervorgehoben wird. Über das UI²⁴ am oberen Fensterrand können anschließend durch einen Mausklick die drei zur Verfügung stehenden Türme ausgewählt und platziert werden. Hierbei wird im Hintergrund überprüft, ob die Fläche noch bebaubar ist und ob genügend Ressourcen vorhanden sind. Sind die Bedingungen erfüllt, wird der Turm gesetzt und beginnt sofort die Gegner unter Beschuss zu nehmen.

2. Verkaufen und Erweitern der Verteidigungsanlagen

Damit der Spieler die Türme an seine taktischen Überlegungen anpassen und Änderungen vornehmen kann, ist es möglich, bereits platzierte und gekaufte Türme wieder zu verkaufen oder auch zu erweitern. Der Verkauf eines Turmes läuft nach einem ähnlichen Schema ab wie der Kauf. Zunächst wählt der Spieler mittels Mausklick das Feld mit dem entsprechenden Turm aus und danach im UI den Verkaufsbutton. Hierbei wird noch vor dem Verkauf der Verkaufserlös für den angewählten Turm im UI angezeigt, wobei sich dieser Verkaufspreis an der Ausbaustufe des Turms orientiert. Das Erweitern eines Turmes geschieht auf dieselbe Art, wobei jeder Turm bei der Aufwertung unterschiedlich viel kostet und je nach Turm auch eine andere Wertesteigerung erfährt. Die aktuelle Ausbaustufe eines Turmes wird farblich auf dem Spielfeld dargestellt.

3. Gegnerische Einheiten (10 Abstufungen)

Es gibt insgesamt 10 unterschiedliche Gegner in dem Prototyp, wobei sich acht Gegnertypen grundsätzlich ähnlich verhalten, während die übrigen zwei Gegnertypen sich vor allem in Anzahl und Schwierigkeitsgrad zu den ersten acht Gegnern unterscheiden. Die Gegner kommen in einer festen Abfolge auf das Spielfeld. Die ersten acht Gegnertypen treten als verschiedenfarbige Dreiecke auf, die sich nur in Geschwindigkeit, Lebenspunkten und Ressourcengewinn unterscheiden. Der neunte Gegner bildet die Vorstufe zum Bossgegner und besteht aus Vierecken, die sich durch Anzahl und Schwierigkeit von den übrigen Gegnern abheben. Der zehnte Gegner ist der Bossgegner, welcher alleine auftritt und sich sehr langsam fortbewegt, dafür aber auch über die meisten Lebenspunkte verfügt und bei seinem Ableben dem Spieler sehr viele Ressourcen hinterlässt. Ab der zehnten Gegnerwelle wiederholt sich das Muster und es erscheint erneut der erste Gegnertyp.

²⁴ User Interface; Benutzungsschnittstelle: Interaktionsmöglichkeit des Nutzers mit dem Spiel

4. Wellenmodus mit Bossmechanik

In Anlehnung an die typischen Genrevertreter wurde bewusst auf einen simplen Endlosmodus verzichtet und ein komplexerer Wellenmodus implementiert. Alle 30 Sekunden erscheint eine neue Gegnerwelle, die nach der anschließenden Pfadberechnung versucht ihren Zielpunkt zu erreichen. Die letzten beiden Gegnerwellen (Nummer 9 & 10) stellen die größte Herausforderung für den Spieler dar. Nachdem entweder der Boss aus Welle 10 erfolgreich besiegt wurde oder die 30 Sekunden für das Erscheinen der Welle 11 abgelaufen sind, werden die Schwierigkeitsmodifikatoren für die nächsten Wellen angepasst. Anschließend wiederholt sich das Muster der Wellengenerierung, wobei alle zehn Wellen der Schwierigkeitsgrad erneut angepasst wird, um dem Spieler stets eine möglichst adäquate Herausforderung zu bieten.

5. Vorgenerierte Spielfelder (6 Varianten)

Um die gestellte Hypothese überprüfen zu können, war es notwendig, mehrere auswählbare Spielfelder bereitzustellen, die sich in Bezug auf die Komplexität und den Schwierigkeitsgrad unterscheiden. Wie bereits in einem vorherigen Abschnitt erwähnt, ergeben sich Schwierigkeit und Komplexität über den Weg, den die Gegner zurücklegen müssen. Mehr Kurven, eng aneinander vorbeiführende Streckenabschnitte und längere Wege verringern den Schwierigkeitsgrad enorm und tragen so zu einer höheren Erfolgswahrscheinlichkeit des Spielverlaufs bei. Aus diesem Grund wurde darauf geachtet, sowohl Spielfelder mit geradlinigem Streckenverlauf als auch Spielfelder mit sehr vielen Abbiegungen zu erstellen, aus denen der Spieler auswählen kann. Hierzu wurden Spielfelder aus bekannten Genrevertretern als Referenz herangezogen und teilweise möglichst originalgetreu im Prototypen nachgebildet. Die Spielfelder wurden anschließend noch für die Levelauswahl im Miniaturformat dargestellt. Damit das Ergebnis der Datenerfassung möglichst gut zu interpretieren ist, wurde versucht, die Spielfelder zwar im Grundsatz ähnlich, aber vom Aufbau und der Spielweise unterschiedlich zu gestalten.

Bei der Bereitstellung von Spielfeldern wurden bewusst nur sechs Spielfelder gewählt, denn wie Studien zeigen, verringern sich Hingabe und Zufriedenheit mit der Entscheidung und vor allem die Wahrscheinlichkeit zur Wiederholung des Auswahlprozesses, wenn zu viele Auswahlmöglichkeiten zur Verfügung stehen [01].

Um alle benötigten Daten für die Analyse und die Auswertung des Spielverhaltens der Nutzer zu ermöglichen, wurden zusätzlich noch folgende technische Features integriert:

1. Unique User Tracking auf Basis von WebSessionStorages

Da im Zuge der Datenerfassung sichergestellt werden musste, wie viele Unique User sich im Spiel befinden bzw. wie viele Unique Sessions gestartet wurden, ist es unumgänglich, ein System bzw. eine Technik zu finden und zu implementieren, mit der eine solche Erfassung möglich ist. Des Weiteren sollte die Möglichkeit bestehen, einen Unique User auch über mehrere Unique Sessions hinweg eindeutig zu identifizieren, so dass wenig Spielraum für Fehlinterpretationen der Datensätze besteht.

Die momentane Lösung über den WebSessionStorage von Construct2 bietet hierbei eine gute Grundlage, um eine User ID mit dazugehörigen Datensätzen in dem WebStorage des Browsercaches

zu lagern. Allerdings besteht trotzdem die Möglichkeit, dass ein Spieler die Browserdaten automatisch oder manuell zurücksetzt bzw. löscht, so dass ein zweiter Unique User mit passender User ID generiert wird, obwohl es der gleiche Spieler ist. Diese Schwachstelle in der Datenerfassung könnte nur über dedizierte Loginmechanismen mit einer Nutzerverwaltung beseitigt werden. Allerdings würde eine Loginpflicht (über Facebook-Login, Google-Login, Kongregate-Login o. Ä.) eine zusätzliche Hürde für den Spieler darstellen und potenzielle Spieler vom Spielerlebnis fernhalten.

Aus diesem Grund wurde bei der Auswertung die Möglichkeit der Doubletten als geringfügige Abweichung erkannt und akzeptiert. Die hierdurch entstehende Fehlerquote wird in der Auswertung vernachlässigt.

Eine genauere Übersicht über den Prozess hinter dem Unique User Tracking liefert das folgende Diagramm:

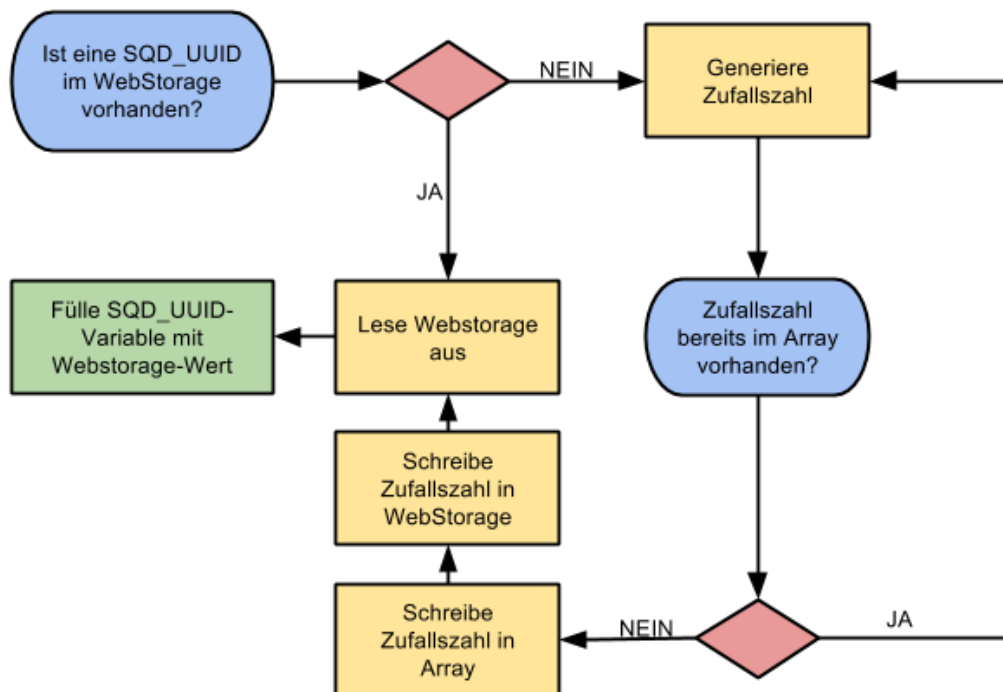


Abbildung 9 - Unique User Tracking Diagramm

2. User Action Tracking mit der Google Analytics API

Für eine möglichst genaue, verlässliche und leicht anpassbare Trackingmethode wurde die API von Google Analytics eingesetzt, welche es möglich macht, angepasste Events mit entsprechenden Datensätzen zu erfassen und zu verarbeiten. Hierbei gibt es keine Einschränkungen, die durch die Google Analytics API entstehen und die nicht schon durch Construct2 auftreten würden. Die Events werden über die Logik-Blöcke direkt mit angesprochen und wie jeder anderer Event-Block bei Construct2 behandelt. Sobald ein entsprechendes Event eintritt oder über ein zweites Event ausgelöst wird, schickt das Spiel ein Informationspaket an den Analytics-Dienst. Die Datensätze sind in der Regel 5-30 Minuten später über die entsprechende Webseite aufrufbar und können von dort in diversen Formaten gespeichert oder über die Webseite auch grafisch dargestellt werden.

Bei Bedarf kann mittels der „Echtzeit-Ansicht“ direkt nachverfolgt werden, wie viele Nutzer sich gerade im Spiel befinden und welche Events ausgelöst werden. Auch wenn der Dienstleistungsumfang von Google Analytics enorm ist, wird nur die Datenerfassung über Events berücksichtigt und ausgewertet.

Im nachfolgenden Diagramm ist der Ablauf schematisch dargestellt:

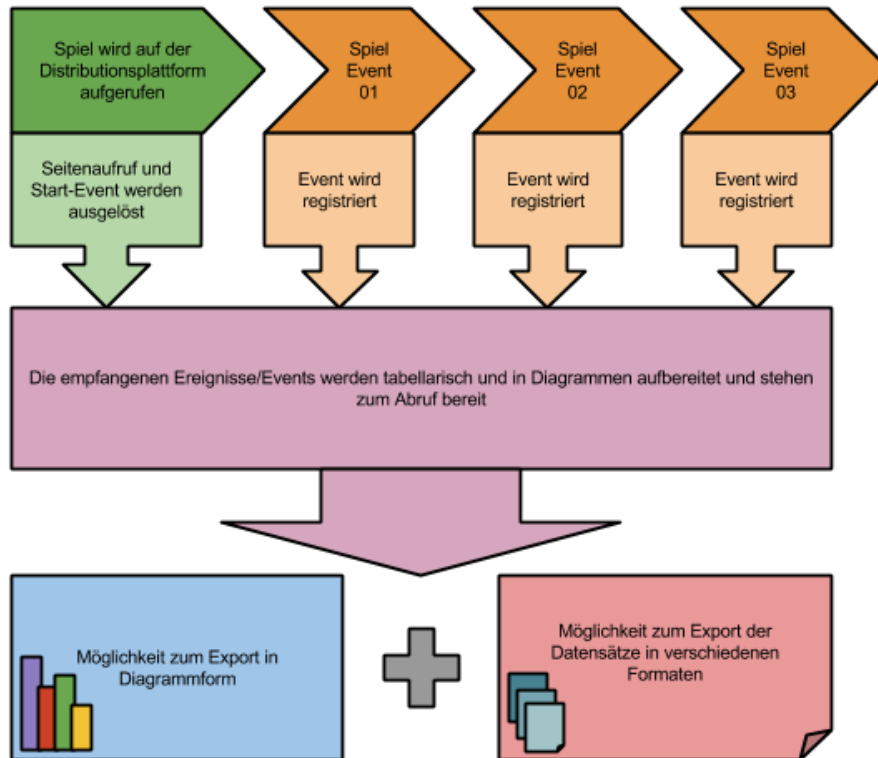


Abbildung 10 - Eventtracking Diagramm

Spielfeldauswahl

Ein grundsätzliches Problem bei statistischen Erhebungen mit mehreren Auswahlmöglichkeiten besteht in der vordefinierten Anordnung der auswählbaren Objekte. Für dieses konkrete Projekt bezieht sich die Aussage auf die Auswahl der Spielfelder bzw. Level durch den Spieler. Beim vorliegenden Prototyp besteht die Herausforderung darin, die bestmögliche Anordnung der Vorschaubilder für die Levelauswahl zu finden, ohne dass der Spieler in irgendeiner Form beeinflusst wird.

Den wohl größten Einfluss auf die Entscheidung des zu wählenden Spielfeldes hat die Leserichtung des Betrachters. Im westeuropäischen und amerikanischen Raum ist die Leserichtung von links nach rechts und von oben nach unten. Am Zeilenende oder wenn in der jeweiligen Zeile keine Objekte mehr vorhanden sind, wird in die darunter liegende Zeile gewechselt, wobei der Blick nicht unmittelbar zum Zeilenanfang springt, sondern diagonal vom Ende zum Anfang wandert.

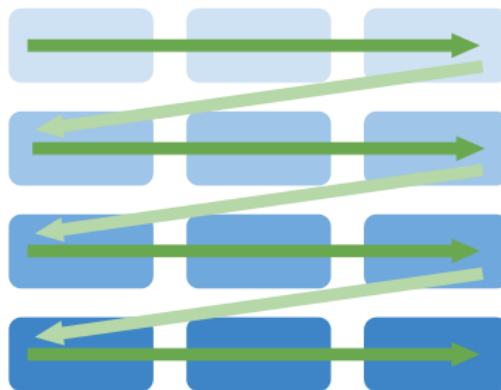


Abbildung 11 - Leserichtung (schematisch dargestellt)

Bei einer Matrix aus sechs Objekten mit einer Aufteilung von drei Objekten pro Zeile und dementsprechend zwei Zeilen erfahren die Objekte an erster und letzter Stelle die geringste Betrachtungsdauer und es erscheint somit unwahrscheinlicher, dass diese ausgewählt werden. Sollten sich also bei dieser Leserichtung stets dieselben Objekte an Position 1 und 6 befinden, so wird die Auswertung allein durch die typische Leserichtung verfälscht.

Eine weitere, leider unvermeidbare Hürde bei der Erfassung der Spielfeldauswahl ist in der menschlichen Psyche durch das Effektgesetz²⁵ begründet [02]. Hierbei geht die Verhaltensforschung davon aus, dass eine in der Vergangenheit liegende Entscheidung mit positivem Ausgang erneut zu einer ähnlichen, wenn nicht sogar gleichen Entscheidung führt [02]. Hat ein Spieler dementsprechend in anderen SDGs bereits die Erfahrung mit einem speziellen Kartentyp gesammelt, so wird dieser Spieler unbewusst dazu neigen, auch in dem vorliegenden Prototyp das mit positiven Gefühlen verknüpfte Spielfeldlayout zu wählen, anstatt komplett unbeeinflusst eine Entscheidung zu treffen. Die vorzeitige Evaluation des Spielfeldes während der Spielfeldauswahl anhand der Vorschaubilder trägt somit bereits zu einem inneren Vergleich mit bereits gespielten Spielen und gewählten Spielfeldern bei und könnte daher das Ergebnis verfälschen.

Im Folgenden werden einige mögliche Darstellungsweisen mit ihren Vor- und Nachteilen genauer erläutert. Hierbei ist anzumerken, dass alle Darstellungsvarianten in ihrem Grundaufbau minimalistisch und simpel gehalten werden, da nach „Ockham’s Razor“ bei unterschiedlichen

²⁵ Law of Effect

Designansätzen stets das einfachste bzw. simpelste gewählt werden sollte [05]. Anschließend wird die letztendlich umgesetzte Variante genannt und die Entscheidung begründet.

Rotierende Darstellung

Bei dieser Darstellungsweise werden alle wählbaren Vorschaubilder der Spielfelder gleichzeitig auf dem Bildschirm angezeigt. Die Vorschaubilder sind entweder kreisförmig oder im Quadrat angeordnet und rotieren mit einer festgelegten Geschwindigkeit um einen Mittelpunkt. Hiermit soll verhindert werden, dass die Anordnung der Elemente den Spieler bei der Entscheidung beeinflusst, da sich die relative Position auf dem Bildschirm ständig ändert.

Zu bedenken war allerdings, dass die Reihenfolge der Elemente in der rotierenden Form stets dieselbe sein würde. Dieses Problem kann allerdings über die zufallsgenerierte Zuteilung der Positionen mit Hilfe eines Arrays der Elemente ausgeglichen werden, erfordert jedoch einen erhöhten Programmieraufwand.

Ein weiteres Problem besteht in der grafischen Umsetzung der zirkularen Bewegung. Damit der Spieler die Miniaturbilder stets korrekt ausgerichtet sehen kann, muss bis zu 60 mal pro Sekunde (um bei 60 FPS²⁶ verzögerungsfrei darstellen zu können) die Position der einzelnen Grafiken angepasst werden, indem diese horizontal und vertikal korrekt ausgerichtet werden.

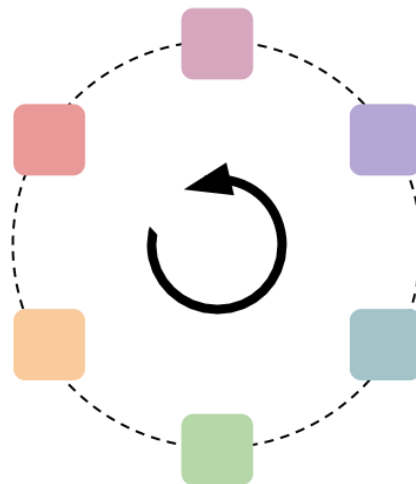


Abbildung 12 - Spielfeldauswahl: Rotierende Darstellung

Cover Flow Darstellung

Bei der Cover Flow Darstellung wird zunächst ein Vorschaubild zentral angezeigt, während die übrigen Vorschaubilder gekippt und verdeckt dargestellt werden, wie Schallplatten in einem Regal. Anschließend kann der Spieler nach links und rechts durch die Sammlung aus Miniaturbildern blättern, wobei nur das aktuelle Bild vollständig gezeigt wird. Mit Hilfe dieser Animation ist es möglich, die Neugier des Spielers zu wecken, da die restlichen Vorschaubilder nicht sofort sichtbar sind. Der Spieler muss erst durch das Herumblättern herausfinden, welche anderen Spielfelder noch zur Auswahl stehen.

²⁶ Frames per second; Bilder pro Sekunde

Auch hier gibt es ein Problem zu berücksichtigen, da der Spieler durch die Auswahl des ersten Miniaturbilds bereits beeinflusst werden kann. Wird die Neugier nicht geweckt, wird der Spieler höchstwahrscheinlich die erste vorgestellte Variante auswählen und das Spiel starten, ohne die Auswahlmöglichkeiten vollständig erfasst zu haben.

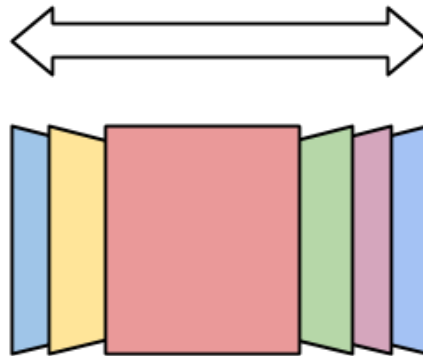


Abbildung 13 - Spielfeldauswahl: Cover Flow Darstellung

Table Sheet Darstellung

Die Table Sheet Darstellung basiert auf der festen Anordnung in einer tabellarischen Ansicht. Hierbei werden die anzuzeigenden Objekte in Spalten und Zeilen angeordnet. Für die Levelauswahl mit sechs unterschiedlichen Karten wurde eine 3x2 Matrix verwendet, also zwei Zeilen mit jeweils drei Spalten. Dieses ist die einfachste und gewöhnlichste Anordnung, wie sie in vielen Spielen bei der Auswahl des Levels oder auch der Spielschwierigkeit zu finden ist. Gleichzeitig unterliegt diese Ansicht aber auch sehr stark den Einflüssen der Leserichtung des Users. Um diesen Faktoren entgegenzuwirken, werden die sechs Vorschaubilder der Spiellevel zufällig auf die sechs verfügbaren Flächen verteilt, so dass z. B. stets ein anderes Vorschaubild auf Position 1 liegt. Allerdings kann nicht sichergestellt werden, dass die Anordnung über alle auszuwertenden Ergebnisse gleich verteilt ist.

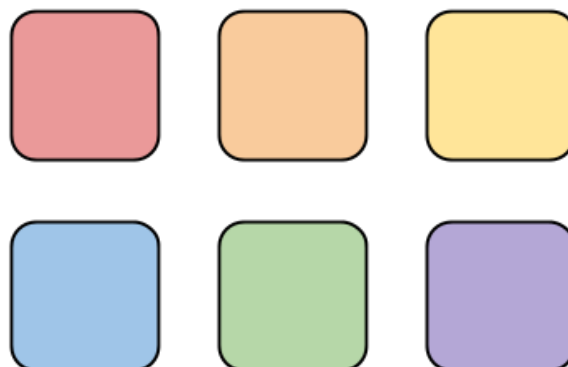


Abbildung 14 - Spielfeldauswahl: Table Sheet Darstellung

Entscheidung

Nach reiflicher Abwägung der einzelnen Darstellungsweisen und den damit verbundenen Vor- und Nachteilen wurde die Table Sheet Darstellung für die Spielfeldauswahl umgesetzt. Bei der Entscheidung zugunsten dieser Darstellung spielte neben der relativ einfachen Umsetzung mittels Zufallspositionierung auch die eigene Erfahrung mit der Spielfeldauswahl in anderen Spielen eine große Rolle. In vielen zeitgemäßen Spielen werden dem Spieler die auszuwählenden Spielfelder, Spielfiguren, Schwierigkeitsgrade oder auch Gegenstände in einem Spiel in Form eines Rasters oder Gitters präsentiert.

Des Weiteren war eine Umsetzung der rotierenden Darstellung mit ständig korrekter Ausrichtung der Vorschaubilder in der horizontalen Achse nicht mit den eigenen Qualitätsansprüchen vereinbar.

Letztendlich wurde die Entscheidung aufgrund der vorhandenen Erfahrung und der technischen Umsetzbarkeit getroffen.

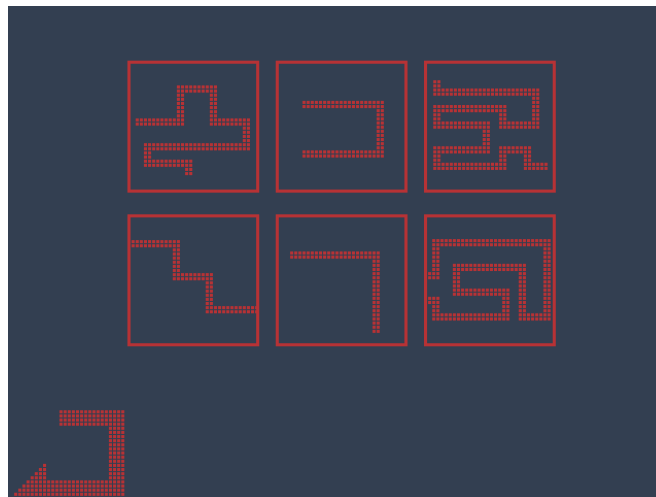


Abbildung 15 - Spielfeldauswahl: Release Version von Quadfender

2.2.5 Analysemethoden

Die Datenanalyse erfolgt, wie bereits im vorherigen Abschnitt erwähnt, mit Hilfe des Analytics-Dienstes von Google. Ursprünglich war hierfür die Verwendung von Flurry²⁷ angedacht, da hiermit bereits in anderen Projekten erste Erfahrungen gewonnen werden konnten. Leider hat sich herausgestellt, dass das Flurry-Plugin für Construct2 seit geraumer Zeit nicht weiterentwickelt wird und nicht nur eine stabile Implementierung gefährdet, sondern teilweise die Daten unvollständig oder gar nicht übertragen werden. Um ein sicheres und zuverlässiges Datentracking zu ermöglichen, wurde deshalb auf Google Analytics als Alternative zurückgegriffen, da diese Plattform über dieselben Features und Möglichkeiten verfügt. Die Einbindung der API bzw. des Google Analytics Plugins unterscheidet sich nur marginal von der Handhabung des Flurry-Plugins.

Der Google Analytics Webdienst kann über die Plugin-Schnittstelle von Construct2 direkt mit dem Spiel verbunden werden, so dass alle Events weitergeleitet werden können. Anschließend müssen im Spiel bestimmte Vorgänge als Ereignis bzw. Event gekennzeichnet werden. Diese Events werden zusammen mit einem String - bestehend aus dem Eventnamen und etwaigen Parametern - über die Schnittstelle

²⁷ Flurry from Yahoo: <http://www.flurry.com/>

an Google Analytics übertragen. So können unterschiedlichste Vorgänge, wie z. B. der Spielstart, das Betätigen einer Schaltfläche oder das Überschreiten eines vorher festgelegten Punktestandes, ein Event auslösen und somit von Google Analytics erfasst werden.

Um vergleichbare Datensätze auf allen Distributionsplattformen zu erhalten, wird der Erfassungszeitraum auf sieben zusammenhängende Tage beschränkt. Es wird angenommen, dass innerhalb dieses Zeitraums genügend Datensätze zur Verfügung stehen, um eine aussagekräftige Schlussfolgerung zu ermöglichen.

2.2.6 Entwicklungsprobleme

Unique User Tracking

Das Unique User Tracking soll sicherstellen, dass jedem User, der das Spiel zum ersten Mal aufruft, eine einzigartige zufallsgenerierte Kennziffer (UUID - Unique User ID) zugeteilt wird. Diese UUID identifiziert den User über den genutzten Browser. Leider ist es nicht möglich mittels der UUID über verschiedene Browser hinweg einen User eindeutig zu ermitteln. Sollte ein User zunächst mit Browser A das Spiel aufrufen, so wird die UUID im Cache von Browser A gespeichert. Sollte der User später mit Browser B erneut auf das Spiel zugreifen, so würde für diesen User eine neue UUID generiert werden und im Cache von Browser B abgelegt werden. Hierbei unterscheiden sich die beiden UUIDs aus Browser A und B. Durch die Art der Speicherung im Browsercache ist es nicht möglich, dieses Szenario zu verhindern und eventuell zu bemerken, dass derselbe User durch die Verwendung von zwei Browsern doppelt in den Statistiken auftaucht. Da es jedoch relativ unwahrscheinlich ist, dass ein User über zwei verschiedene Browser das Spiel aufruft, wird diese Möglichkeit bei der Auswertung der Daten nicht berücksichtigt.

Um sicherzustellen, dass jede zufallsgenerierte UUID einzigartig ist, werden alle UUIDs in einem Array im Spiel gespeichert. Da die Spieldaten nicht bei den jeweiligen Distributionsplattformen direkt verfügbar sind, sondern die Anfragen auf den persönlichen Webspace umgeleitet werden, kann ausgeschlossen werden, dass es Dubletten in der UUID gibt.

Construct2 Plugins

Während der Entwicklung des Prototypen musste aufgrund der Nutzung von Construct2 auf Plugins bzw. Erweiterungen zurückgegriffen werden, die die Datenerfassung ermöglichen. Hierbei wurde zunächst ein Plugin zur Nutzung des Flurry-Dienstes verwendet. Zu Beginn der Prototypenentwicklung wurde die Verwendung ausgiebig getestet und funktionierte wie erwartet. Anschließend wurde die meiste Zeit dafür aufgewendet, die Spielmechaniken zu realisieren und das Erlebnis für den Spieler möglichst „rund“ zu gestalten. Ziel sollte sein, die Absprungrate möglichst gering zu halten und eventuell sogar Spieler zu einer zweiten Runde zu motivieren.

Während der längeren Entwicklungsphase wurde die Datenerfassung nicht oft genug überprüft, so dass in der finalen Testphase festgestellt wurde, dass durch diverse Updates von Construct2, seitens des Herstellers, die Nutzung des Flurry-Plugins offensichtlich nicht mehr sichergestellt werden konnte. Hierbei war es weder möglich das Plugin selbst anzupassen noch auf eine Korrektur des Autors zu warten, da die letzten Konversationen im Construct2-Forum schon einige Zeit zurücklagen. Aufgrund dessen wurde kurzfristig entschieden die gesamte Datenerfassung mit Google Analytics umzusetzen. Auch hier musste erneut auf ein Plugin für Construct2 zurückgegriffen werden. Glücklicherweise stellte sich heraus, dass die Schnittstelle des Plugins noch einwandfrei funktionierte und nur geringe Anpassungen nötig waren.

2.3 AUSWERTUNG DER ANALYSEERGEBNISSE

Im folgenden Abschnitt werden die Ergebnisse der einwöchigen Datenerhebung tabellarisch in Zahlen, grafisch in Diagrammen und in Form einer schriftlichen Auswertung dargereicht, um einen möglichst breiten Einblick in die Daten zu ermöglichen. Nachfolgend werden die Datensätze interpretiert und die gewonnenen Erkenntnisse dargestellt und begründet. Zusammenführend folgt die Auswertung mit einer Gegenüberstellung der interpretierten Ergebnisse und der eingangs gestellten Hypothese.

Eine Darstellung aller ausgelösten Ereignisse mit Anzahl und Häufigkeiten befindet sich im Anhang.

2.3.1 Ergebnisse

Kongregate

Betrachtet man Abbildung 16, lässt sich erkennen, dass Kongregate als primäre Distributionsplattform mit 123 Ereignissen für die Spielfeldauswahl eine solide Basis für eine anschließende Betrachtung der Ergebnisse und für eine mögliche Überprüfung der gestellten Hypothese liefert.

Insgesamt wurden auf Kongregate 235 Sitzungen registriert - mit einer durchschnittlichen Sitzungsdauer von 03:57 Minuten und 214 einzigartigen Nutzern. Die Differenz zwischen Sitzungsanzahl und einzigartigen Nutzern ergibt sich aus den insgesamt 21 Spielwiederholungen einzelner Nutzer.

Abbildung 17 liefert die Aufschlüsselung der Ereignisse gefiltert nach den Spielfeldauswahl-Ereignissen, die zur weiteren Analyse herangezogen werden.

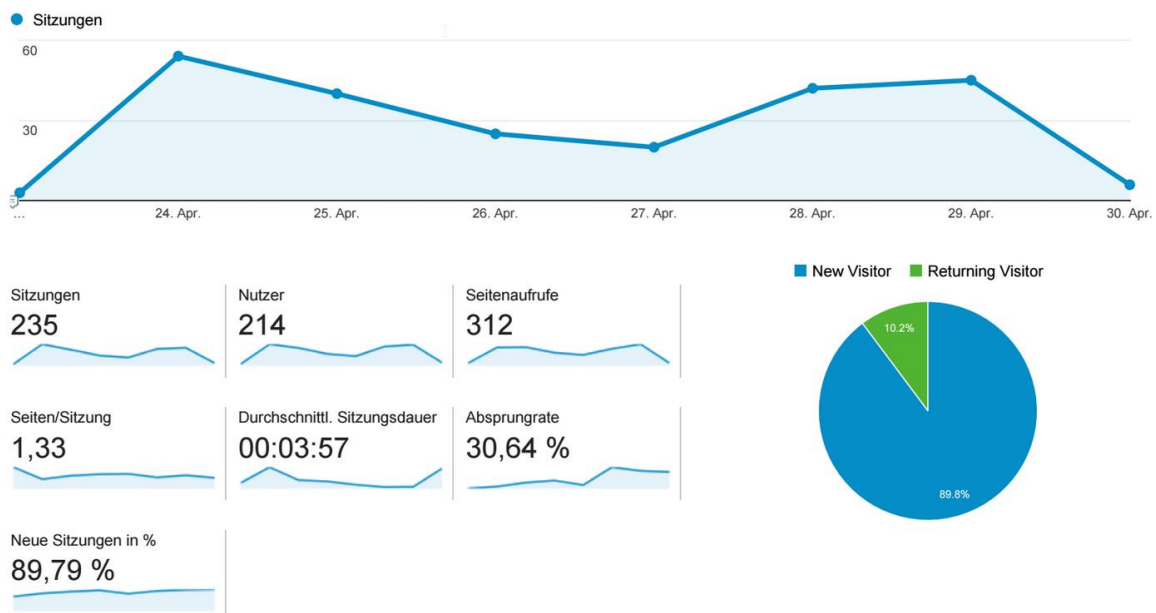


Abbildung 16 - Google Analytics, Kongregate, Übersicht

Ereignisaktion	Ereignisse gesamt	Ereignisse gesamt
	123 % des Gesamtwerts: 2,11 % (5.836)	123 % des Gesamtwerts: 2,11 % (5.836)
1. Select_Level_05	39	31,71 %
2. Select_Level_03	22	17,89 %
3. Select_Level_01	19	15,45 %
4. Select_Level_02	16	13,01 %
5. Select_Level_06	15	12,20 %
6. Select_Level_04	12	9,76 %

Abbildung 17 - Google Analytics, Kongregate, Spielfeldauswahl

GameJolt

GameJolt wurde bewusst als zweite Distributionsplattform gewählt, da hier vor allem frühes Feedback von anderen Indie-Entwicklern eingeholt werden kann. Leider lassen die endgültigen Trackingdaten nach einem einwöchigen Erfassungszeitraum keine fundierten Schlüsse zu, so dass diese Datensätze nicht für die Überprüfung der Hypothese und für nachfolgende Schlussfolgerungen genutzt werden können.

Insgesamt verzeichnet das Prototypspiel auf GameJolt sieben Seitenaufrufe die zu sechs Spielsitzungen von sechs einzigartigen Nutzern geführt haben. Die durchschnittliche Sitzungsdauer liegt bei 05:00 Minuten und es wurden insgesamt drei Ereignisse bei der Spielfeldauswahl registriert.

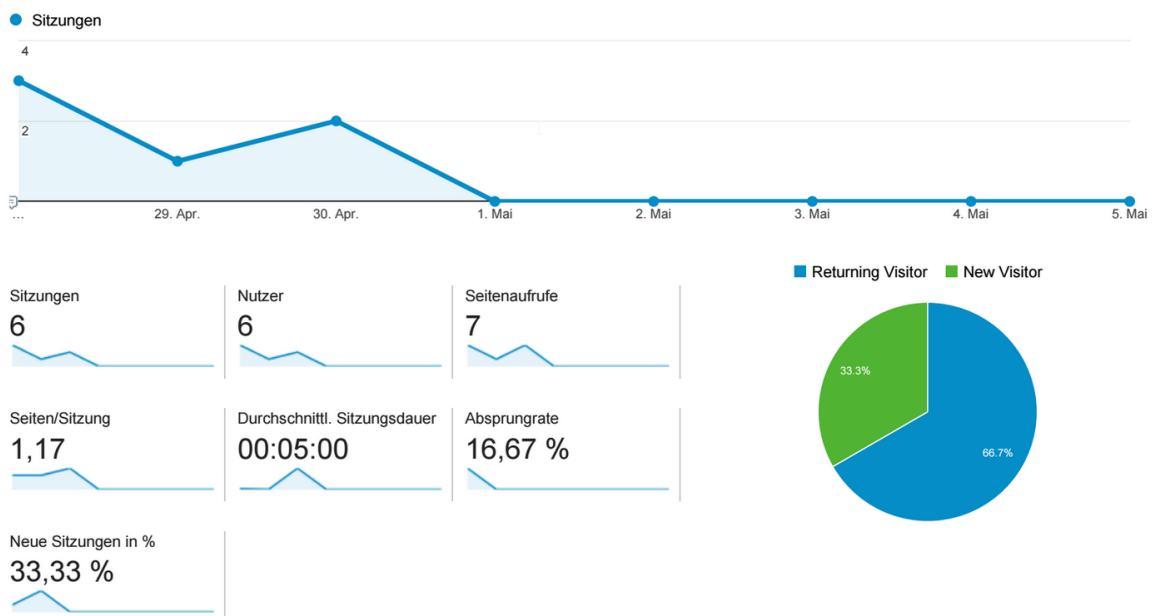


Abbildung 18 - Google Analytics, GameJolt, Übersicht

Ereignisaktion	Ereignisse gesamt	Ereignisse gesamt
	3 % des Gesamtwerts: 0,91 % (331)	3 % des Gesamtwerts: 0,91 % (331)
1. Select_Level_01	2	66,67 %
2. Select_Level_05	1	33,33 %

Abbildung 19 - Google Analytics, GameJolt, Spielfeldauswahl

Newgrounds

Obwohl Newgrounds in Hinblick auf die Spieleperformance bei GameJolt herangezogen wurde, übertreffen die Seitenaufrufe und Spielesitzungen selbst die Primärplattform Kongregate.

Wie Abbildung 20 erkennen lässt, haben 334 Seitenaufrufe zu insgesamt 304 Sitzungen von 282 einzigartigen Nutzern geführt, mit einer durchschnittlichen Sitzungsdauer von 09:38 Minuten. Somit können 22 Spielesitzungen als wiederholte Aufrufe einzigartiger Nutzer registriert werden.

Wie bei Kongregate auch, stellt die Abbildung 21 die Spielfeldauswahl mit der Häufigkeit und Anzahl der einzelnen Spielfelder dar. Eine genaue Analyse mit anschließender Schlussfolgerung folgt nach der zusammenfassenden Betrachtung.

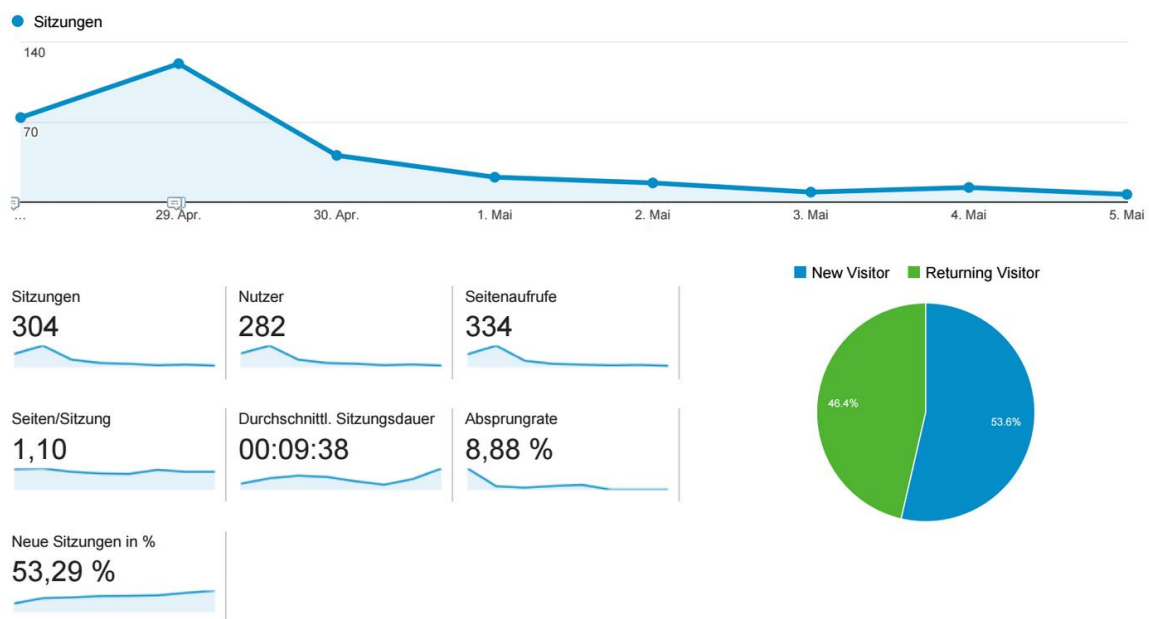


Abbildung 20 - Google Analytics, Newgrounds, Übersicht



Abbildung 21 - Google Analytics, Newgrounds, Spielfeldauswahl

Zusammenfassung

Da über GameJolt keine ausreichende Anzahl an Datensätzen zur Analyse bereitgestellt wurde, wird GameJolt in der Zusammenfassung und der Interpretation der Ergebnisse nicht bewertet und die Datensätze nicht weiter in Betracht gezogen.

Mit dem Spieleprototypen wurden innerhalb eines Zeitraums von einer Woche (7 Tage, inkl. Wochenende) auf den Plattformen Kongregate und Newgrounds insgesamt 539 Sitzungen mit 496 einzigartigen Nutzern über Google Analytics registriert. Die durchschnittliche Sitzungsdauer betrug, nach Tabelle 2, 06:47 Minuten.

Tabelle 2 - Ergebnisse der Distributionsplattformen

	Kongregate	Newgrounds	Gesamt
<i>Ereignisse</i>	123	387	510
<i>Unique User</i>	214	282	496
<i>Durchschnittliche Sitzungsdauer</i>	00:03:57	00:09:38	~00:06:47
<i>Select_Level_01</i>	19	67	86
<i>Select_Level_02</i>	16	42	58
<i>Select_Level_03</i>	22	50	72
<i>Select_Level_04</i>	12	68	80
<i>Select_Level_05</i>	39	97	136
<i>Select_Level_06</i>	15	63	78

2.3.2 Auswertung der Ergebnisse

In diesem Abschnitt soll die anfänglich gestellte Hypothese den interpretierten Ergebnissen gegenübergestellt werden, um zu überprüfen, ob sich damit die Hypothese widerlegen oder bestätigen lässt.

Die eingangs definierte Hypothese lautet:

Eine hohe Anzahl an Spielfeldvariationen ist unnötig, da die Mehrheit der Spieler einen bestimmten Spielfeldtyp bevorzugt und viele Auswahlmöglichkeiten nicht relevant für die Spielmotivation sind.

Stellt man dieser Hypothese nun die Ergebnisse aus Tabelle 3 gegenüber, so lässt sich zunächst erkennen, dass bei Spielfeld Nr. 5 die höchste Anzahl an Spieldurchgängen ermittelt werden konnte. Dieses Spielfeld wurde insgesamt 136 mal ausgewählt und repräsentiert damit mit 26,66% knapp über ¼ aller gewählten Spielfelder.

Tabelle 3 - Spielfeldauswahl (Anzahl und Häufigkeit)

	Anzahl	in %
Select_Level_01	86	17,45
Select_Level_02	58	11,37
Select_Level_03	72	14,12
Select_Level_04	80	15,68
Select_Level_05	136	26,66
Select_Level_06	78	15,29

Ein Rückschluss über die Anzahl der einzigartigen Nutzer, die Spielfeld Nr. 5 ausgewählt haben, lässt sich hieraus allerdings nicht ableiten.

Die Wahrscheinlichkeit, dass ein bestimmtes Spielfeld zufällig vom Nutzer ausgewählt wird, liegt bei

$$P(E) = \frac{\text{Bestimmtes Ereignis}}{\text{Alle Ereignisse}} = \frac{1}{6} = 16,6667\%$$

Mit dem Ergebnis von 26,66% für das meistvertretene Spielfeld und damit einer Differenz von nur 10% zu P(E), lässt sich die gestellte Hypothese weder eindeutig belegen noch kann die Nullhypothese verworfen werden. Somit kann keine statistisch signifikante Aussage bezüglich der anfänglich gestellten Hypothese getroffen werden.

3 SCHLUSS

In den folgenden beiden Abschnitten werden zunächst die Arbeitsergebnisse, die gewählte Methode, die Hypothese und die Herangehensweise kritisch begutachtet, um im Anschluss auf mögliche zukünftige Forschungen oder Thematiken einzugehen.

3.1 ABSCHLIEßENDE BETRACHTUNG

Ziel der vorliegenden Master-Thesis war der Erhalt einer eindeutig belegbaren Aussage über die Notwendigkeit multipler Spielfelder im Bereich der Strategic Defense Games. Zu diesem Zweck wurde ein Spielprototyp entwickelt, der mittels Datentracking alle benötigten Datensätze liefern sollte, um eine fundierte Aussage treffen zu können. Bei der Entwicklung musste vor allem darauf geachtet werden, keine Methoden zu verwenden, die die Ergebnisse zu stark verfälschen könnten und die Studie gefährden würden. Nachdem verschiedene Konzepte evaluiert wurden, konnten bereits erste Probleme bei der gestellten Hypothese und den gewählten Methoden benannt werden. Dabei ergab sich, dass sowohl die Tiefe des Datentrackings als auch die Auswahl der Zielgruppe und der Distributionsplattformen bereits im Vorfeld einen nicht unerheblichen Einfluss auf die Ergebnisse nehmen könnten. Anhand verschiedener Maßnahmen wurde versucht diesen Faktoren entgegenzuwirken, jedoch können weder Auswirkungen dieser Maßnahmen noch der Einfluss der Risikofaktoren eindeutig aus den Trackingergebnissen abgelesen werden.

Was die gestellte Hypothese betrifft, so konnte anhand der erhaltenen Werte des Trackingzeitraums kein signifikantes Ergebnis erzielt werden. Eine Vergrößerung der erfassten Datensätze, die Nutzung von zusätzlichen Distributionsplattformen und die Verfeinerung des Auswahlprozesses der Spielfelder sollte für eine ähnliche Studie unbedingt in Betracht gezogen werden.

Zwar kann anhand der erhaltenen Datensätze darauf geschlossen werden, dass es eine eventuelle Tendenz zu einem bestimmten Typ von Spielfeld gibt, doch kann dies nicht mit wissenschaftlich fundierten statistischen Werten und Schlussfolgerungen untermauert werden und basiert lediglich auf der Häufigkeitsverteilung der 510 Spieldurchgänge im gewählten Testzeitraum.

Insgesamt kann festgestellt werden, dass die Frage, ob keine große Auswahl an Spielfeldern benötigt wird und stattdessen die Spielmechaniken im Vordergrund stehen sollten, nicht zufriedenstellend geklärt werden konnte und in einer weiteren Studie geklärt werden müsste.

3.2 AUSBLICK

Obwohl die Master-Thesis in Hinblick auf die gestellte Hypothese nicht zur vollen Zufriedenheit zu einem eindeutigen Ergebnis führte, bietet sie viele Ansatzmöglichkeiten für eine Weiterentwicklung oder den Wissenstransfer in ein neues Projekt.

Zunächst ist eine Erweiterung des Prototypen um einige weitere Features geplant, die allerdings nicht im direkten Zusammenhang mit dieser Thesis stehen. Während der Arbeit am Prototypen musste im Hinblick auf den engen zeitlichen Rahmen der Master-Thesis auf einige Spielelemente verzichtet und der Rest minimalistisch gehalten werden. Für die weitere Entwicklung ist neben einer besseren Gegnerbalance, mehr bzw. besser definierten Türmen und einem Achievementsystem vor allem die grafische Aufbesserung geplant. Für den Prototypen wurden sämtliche Grafiken in Eigenproduktion erstellt, worauf in einer folgenden Version nach Möglichkeit verzichtet werden soll, da die fachliche Expertise auf dem Gebiet des Art-Designs fehlt.

Das Tracking in seiner jetzigen Form war für die anfängliche Überprüfung ausreichend und detailliert genug. Sollte jedoch eine weitere genauere Untersuchung folgen, muss eventuell auch über eine Änderung der Entwicklungsumgebung nachgedacht werden, um zukünftige Fehler mit benötigten Plugins zu vermeiden und eine größere Tiefe der Datensätze zu erreichen. Diesbezüglich ist bereits eine Umsetzung in Unity angedacht und wird momentan evaluiert. Für die Realisierung des Datentracking ist die Umsetzung der Datenspeicherung in der Dropbox nicht ausreichend bzw. wenig praktikabel und muss in dem Fall auf eine externe Datenbank ausgelagert werden. Mit Hilfe dieser Datenbank können dann auch alle benötigten Daten für eine zukünftige Analyse gespeichert werden, um nicht nur Spieldurchläufe und bestimmte Ereignisse zu tracken, sondern auch, um Heatmaps und vor allem Bewegungsprofile innerhalb des Spiels zu erfassen und darzustellen. Des Weiteren kann über eine geeignete Schnittstelle und die Anbindung an eine eigene Datenbank das Timetracking funktionsfähig realisiert werden, so dass eine zeitliche Erfassung der Spielerbewegung möglich ist, um neue Erkenntnisse für die zukünftige Entwicklung zu gewinnen.

Abschließend betrachtet war diese Master-Thesis in vielerlei Hinsicht eine Erkundung von neuen Wissensgebieten, nicht nur im Hinblick auf die Überprüfung der Hypothese, sondern vor allem auch in der persönlichen Entwicklung von Programmier- und Analysefähigkeiten. Ohne diese Master-Thesis wäre eine ambitionierte Umsetzung eines Strategic Defense Games zum gegenwärtigen Zeitpunkt nicht denkbar gewesen und durch die Einbindung von Analysetools wurde der Blick auf die spielentscheidenden Zusammenhänge geschärft und kann für zukünftige Projekte genutzt werden.

QUELLEN

- [01] When choice is demotivating: Can one desire too much of a good thing?
Sheena S. Iyengar, Mark R. Lepper
Journal of Personality and Social Psychology Vol. 79(6), 2000
- [02] Denken – Urteilen, Entscheiden, Problemlösen
Prof. Dr. Tilmann Betsch, Prof. Dr. Joachim Funke, Prof. Dr. Henning Plessner
Springer-Lehrbuch, 2011
ISBN: 978-3-642-12473-0
- [03] Adaptive AI to Play Tower Defence Game
Paul A. Rummell
IEEE, The 16th International Conference on Computer Games (CGAMES), 2011
ISBN: 978-1-4577-1451-1
- [04] Computational Intelligence and Tower Defence Games
Phillipa Avery, Julian Togelius, Elvis Alistar, Robert Pieter van Leeuwen
IEEE, Congress on Evolutionary Computation (CEC), 2011
ISBN: 978-1-4244-7834-7
- [05] Universal Principles of Design
William Lidwell, Kritina Holden, Jill Butler
Rockport Publishers Inc., 2003
ISBN: 1-59253-007-9

Die nachfolgenden Quellen haben vor allem bei der praktischen Umsetzung des Prototypen Verwendung gefunden und wurden in der schriftlichen Ausarbeitung nicht weiter genutzt.

- [06] Heuristic Evaluation for Games: Usability Principles for Video Game Design
David Pinelle, Nelson Wong, Tadeusz Stach
ACM, Proceedings of the SIGCHI Conf. on Human Factors in Computing Systems, 2008
ISBN: 978-1-60558-011-1
- [07] Simple Gamer Interaction Analysis through Tower Defense Games
Fernando Palero, Antonio Gonzalez-Pardo, David Camacho
Springer Int. Publishing, New Trends in Computational Collective Intelligence, 2015
ISBN: 978-3-319-10774-5
- [08] Towards Gameplay Analysis via Gameplay Metrics
Anders Drachen, Alessandro Canossa
Proceedings of the 13th Int. MindTrek Conf.: Everyday Life in the Ubiquitous Era, 2009
ISBN: 978-1-60558-633-5

ANHANG

Quadfender Main Menu.....	ii
Quadfender Credits	ii
Quadfender Levelselect.....	iii
Quadfender Game_Scene 01	iii
Quadfender Game_Scene 02	iv
Quadfender Game_Scene 03	iv
Quadfender Game_Scene 04	v
Quadfender Game_Scene QuitDialog	v
Construct2 - Game_Sheet 01	vi
Construct2 - Game_Sheet 02	vi
Construct2 - Game_Sheet 03	vii
Construct2 - Game_Sheet 04	viii
Construct2 - Game_Sheet 05	ix
Construct2 - Game_Sheet 06	x
Construct2 - Game_Sheet 07	xi
Construct2 - Game_Sheet 08	xii
Construct2 - Main_Menu_Sheet 01	xiii
Construct2 - Main_Menu_Sheet 02	xiii
Kongregate, Google Analytics, Alle Ereignisse	xiv
GameJolt, Google Analytics, Alle Ereignisse.....	xv
Newgrounds, Google Analytics, Alle Ereignisse	xvi

ANHANG A



Abbildung 22 - Quadfender Main Menu

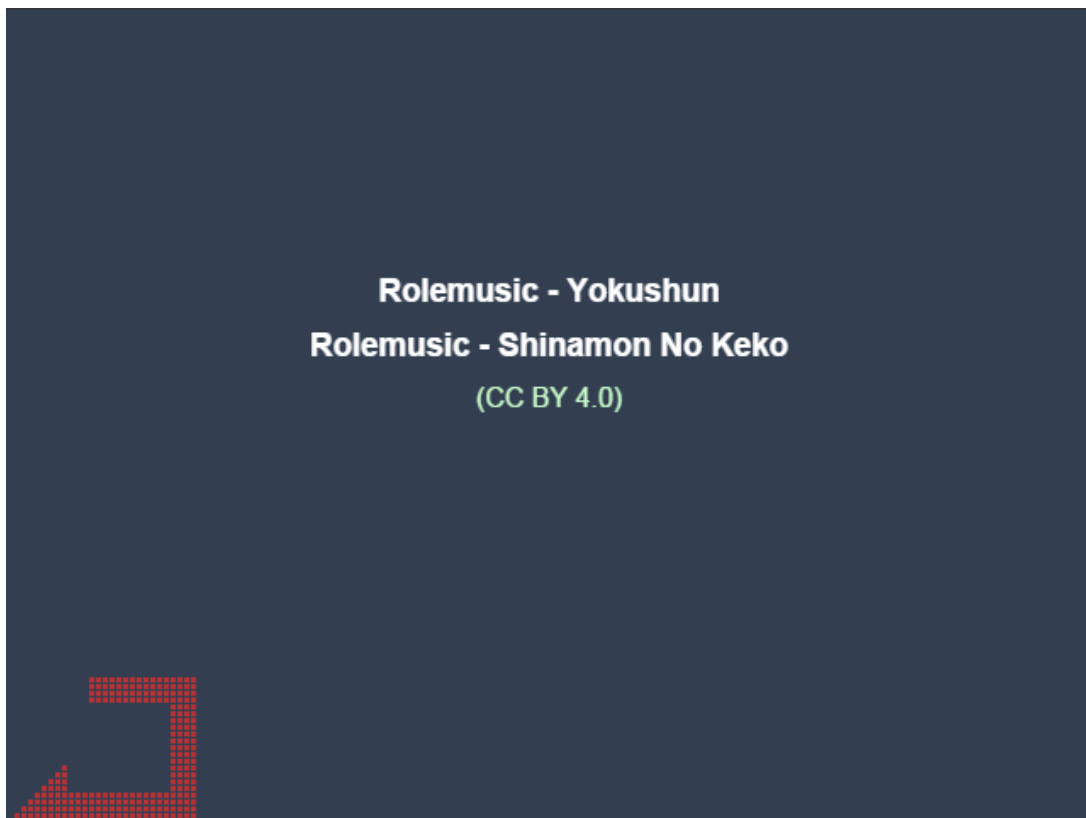


Abbildung 23 - Quadfender Credits

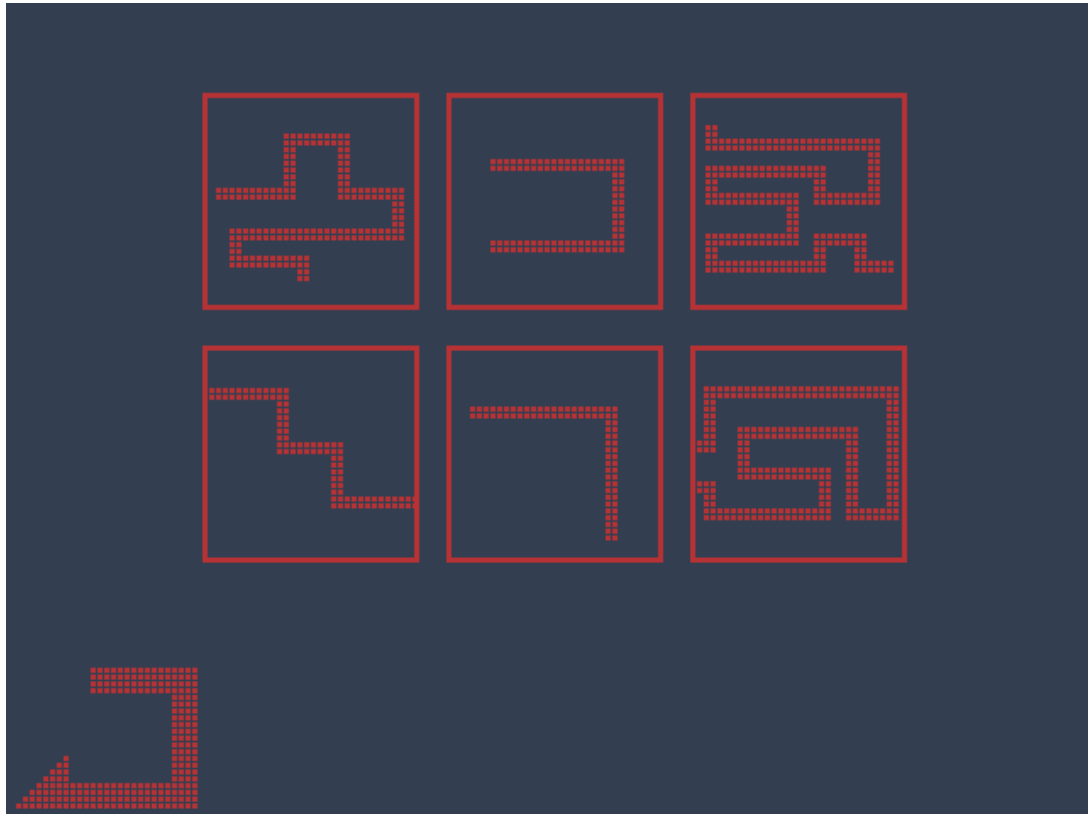


Abbildung 24 - Quadfender Levelselect

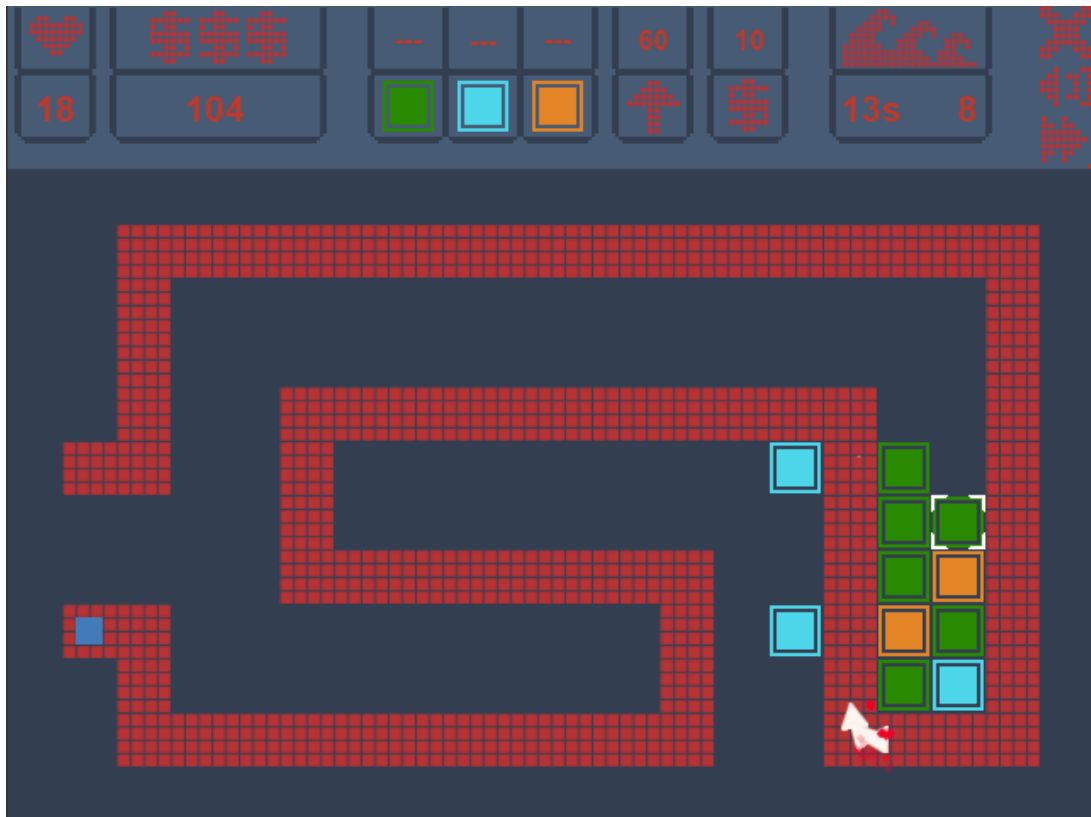


Abbildung 25 - Quadfender Game_Scene 01

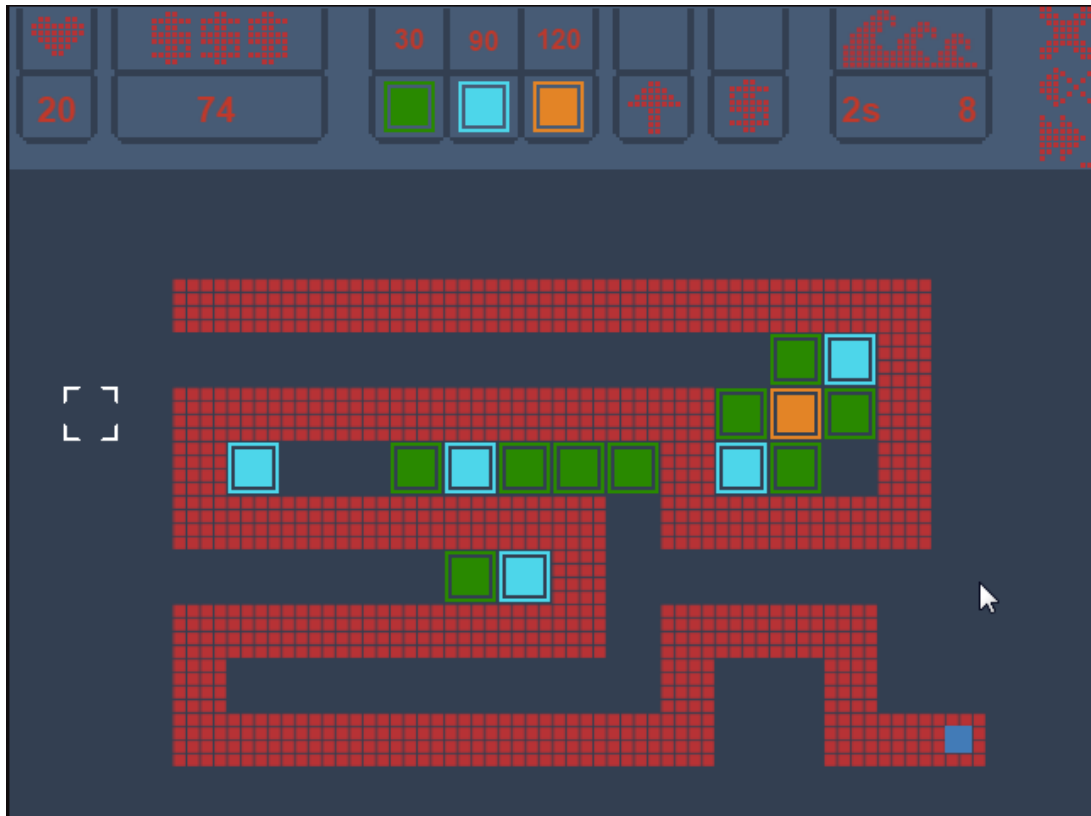


Abbildung 26 - Quadfender Game_Scene 02

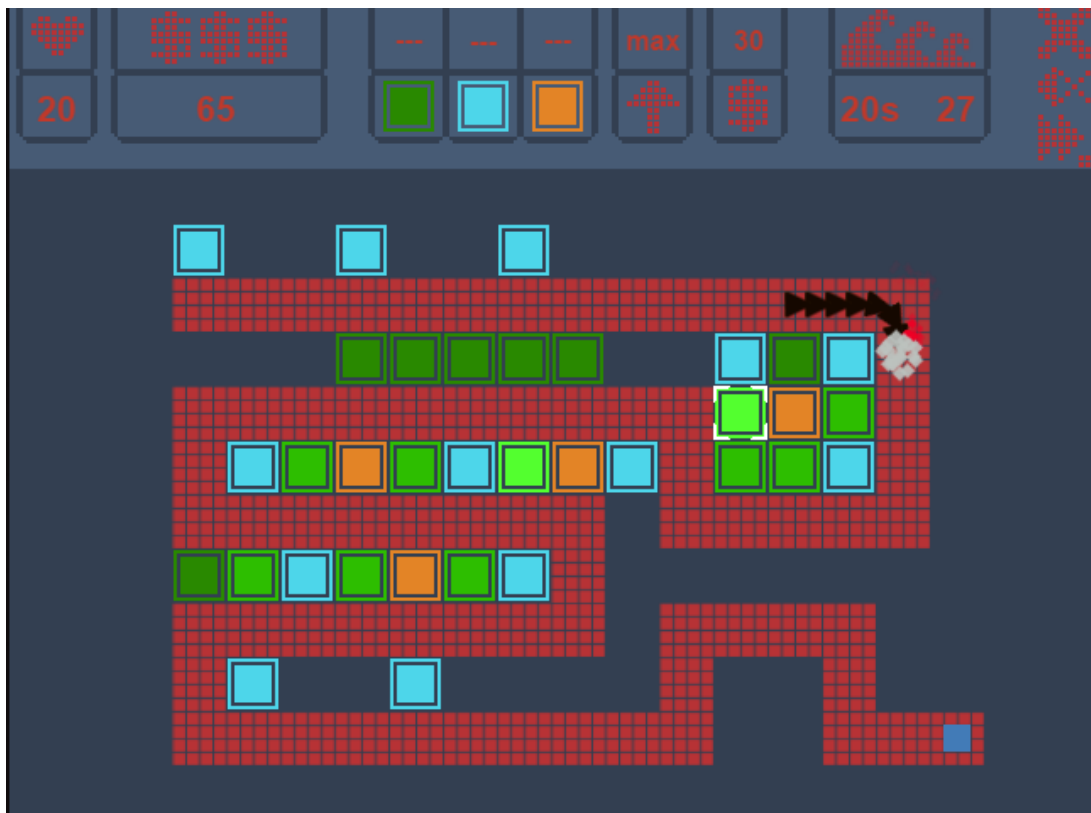


Abbildung 27 - Quadfender Game_Scene 03

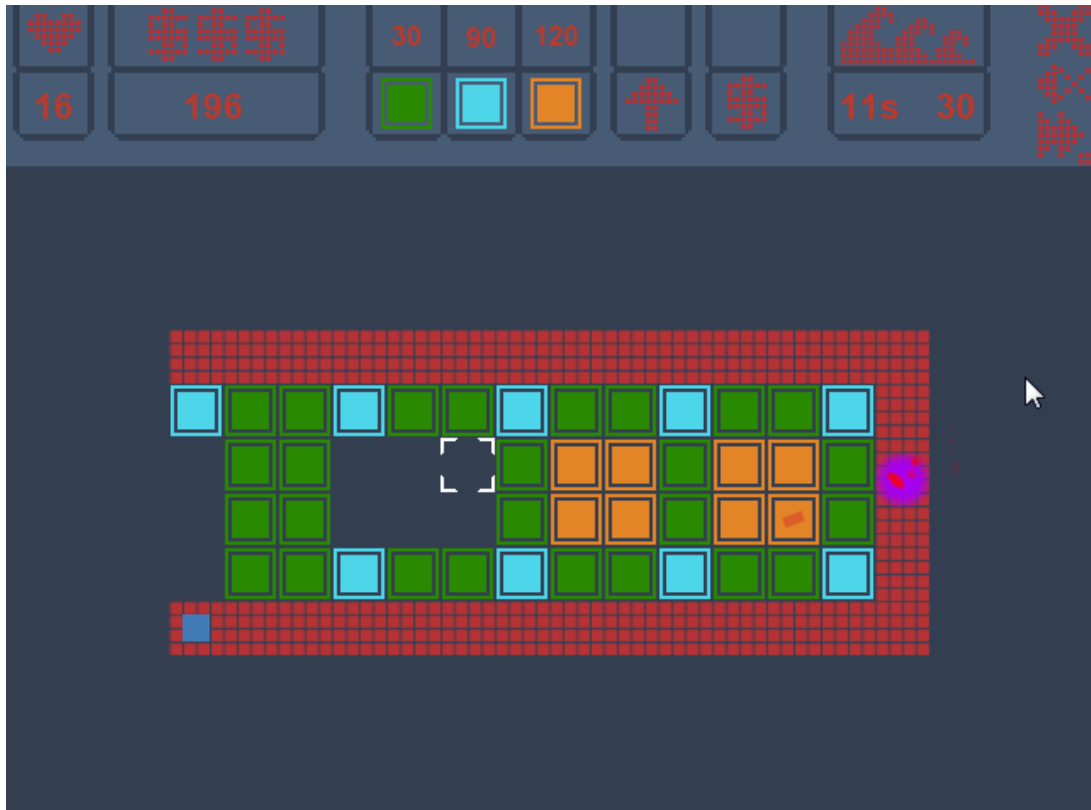


Abbildung 28 - Quadfender Game_Scene 04

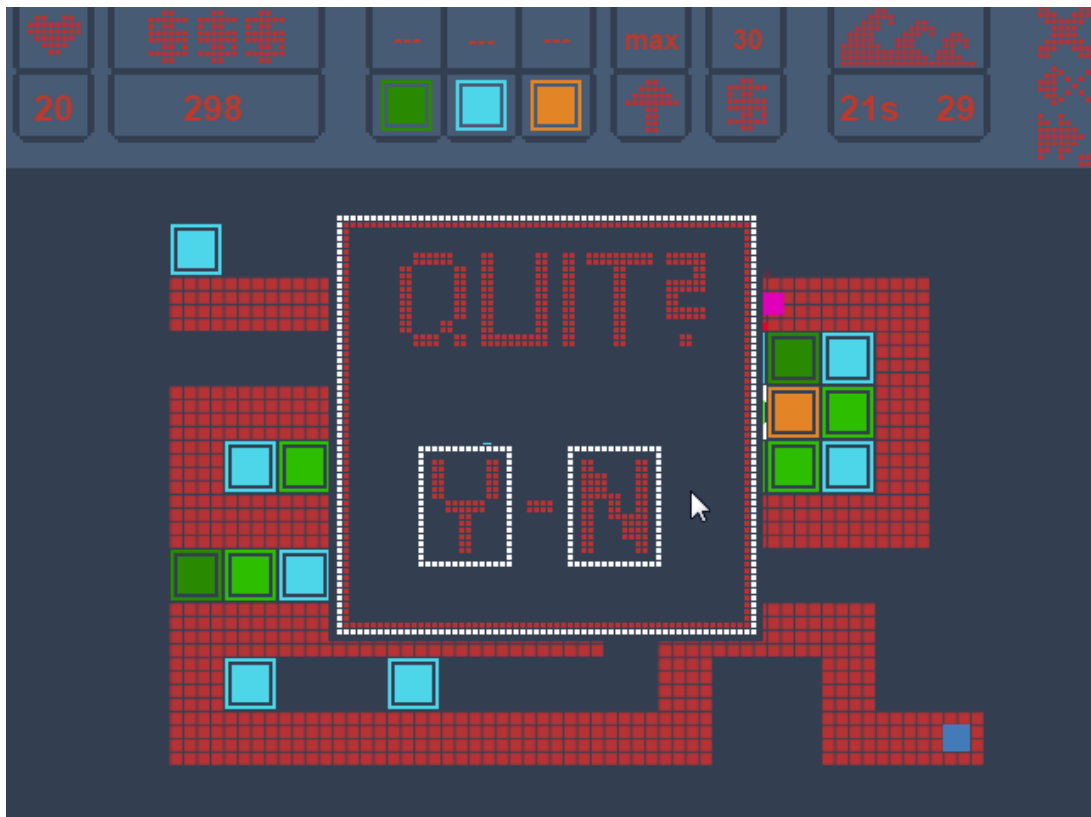


Abbildung 29 - Quadfender Game_Scene QuitDialog

ANHANG B

+++ Infinite Mode Globals +++	
Global number	InfiniteMode_EnemySpawnTime = 2 <small>Seconds interval between spawning a new enemy.</small>
+++ Wave Mode Globals +++	
Global number	WaveMode_TotalWaves = 0
Global number	WaveMode_WaveCounter = 0
Global number	WaveTimer = 0
Global number	WaveMode_PauseTime = 30 <small>Time between waves in seconds</small>
Global number	WaveMode_EnemySpawnTime = 0.25 <small>Seconds interval between spawning a new enemy.</small>
Global number	WaveMode_SpawnBossCounter = 1
Global number	WaveMode_BossDifficulty_Modifier = 0.75
Global number	WaveMode_WaveDifficulty_Modifier = 1 <small>base = 1</small>
Global number	WaveMode_SpawnCreepCounter = 10
Enemy Stats	
Global number	Enemy_01_BaseHealth = 2
Global number	Enemy_02_BaseHealth = 3
Global number	Enemy_03_BaseHealth = 4
Global number	Enemy_04_BaseHealth = 6
Global number	Enemy_05_BaseHealth = 8
Global number	Enemy_06_BaseHealth = 10
Global number	Enemy_07_BaseHealth = 12
Global number	Enemy_08_BaseHealth = 14
Global number	Enemy_09_BaseHealth = 35
Global number	Enemy_10_BaseHealth = 200
+++ All Modes Globals +++	
Global number	TESTMODE = 0
Global number	GameMode = 0 <small>0=undecided, 1=endless, 2=waves</small>
Global number	Lives = 20 <small>Player Lives</small>
Global number	Credits = 150 <small>Credits start amount</small>
Global number	Music_Toggle = 1 <small>0=off, 1=on</small>
Global number	PlayerDeath = 0
Global number	Upgrade_Increase = 0.1
Global number	Sound_Toggle = 1 <small>0=off, 1=on</small>
+ Start of Layout	
+ EVERY TICK	
+ ENEMIES	
+ TURRETS	
+ UI_MENU	

Abbildung 30 - Construct2 - Game_Sheet 01

Start of Layout	
System	On start of layout
Lives	Set text to "&Lives"
Credits	Set text to "&Credits"
GoogleAnalytics	Track duration of "Game": "Duration"
GoogleAnalytics	Track event "Game": "Start"
Audio	Play choose("Shinamon_No_Neko", "Yokushun") looping from Music at -15 dB (tag "music")
UI_Select_Center	Pin Pin to UI_Select (Position & angle)
Dialog_Quit_Yes	Pin Pin to Dialog_Quit_Background (Position & angle)
Dialog_Quit_No	Pin Pin to Dialog_Quit_Background (Position & angle)
Sell_Value	Set text to ""
Upgrade_Value	Set text to ""
Build_Value_Normal	Set text to "---"
Build_Value_Splash	Set text to "---"
Build_Value_Slow	Set text to "---"
WaveCounter	Set text to WaveMode_TotalWaves
System	Set WaveTimer to WaveMode_PauseTime
Timer_NextWave	Set text to WaveTimer&'s"

Abbildung 31 - Construct2 - Game_Sheet 02

EVERY TICK			
+++ Check if player lives are zero -> death +++			
System	Every tick	LocalStorage	Set local key "WaveMode_TotalWave" to WaveMode_TotalWaves
System	Lives = 0	System	Set EnemyFamily time scale to 0
		System	Set BulletFamily time scale to 0
		System	Wait 2 seconds
		GoogleAnalytics	Stop tracking duration of "Game": "Duration"
		GoogleAnalytics	Track event "Game": "Total_Waves" with label "" and value WaveMode_TotalWaves
		GoogleAnalytics	Track event "Game": "GameOver"
		Audio	Stop "music"
		System	Set time scale to 1
		System	Reset global variables to default
		System	Go to GameOver
Add action			
UPDATE ALL TEXT FIELDS			
System	Every tick	Timer_NextWave	Set text to WaveTimer&'s"
Add action			
Turret_Slow_Overlay	On created	System	Wait 1 seconds
		Turret_Slow_Overlay	Set opacity to 25
		System	Wait 1 seconds
		Turret_Slow_Overlay	Set Invisible
Add action			
Turret_Splash1_Overlay	On created	System	Wait 0.1 seconds
		Turret_Splash1_Overlay	Destroy
Add action			
Turret_Splash2_Overlay	On created	System	Wait 0.1 seconds
		Turret_Splash2_Overlay	Destroy
Add action			
Turret_Splash3_Overlay	On created	System	Wait 0.1 seconds
		Turret_Splash3_Overlay	Destroy
Add action			

Abbildung 32 - Construct2 - Game_Sheet 03

ENEMIES			
DEFAULT !!!! +++ WAVE MODE -> Spawn "WaveMode_SpawnCreepsCounter" enemies with short interval then pause spawning			
System	Lives > 0		Add action
System	Every 1.0 seconds	System	Subtract 1 from WaveTimer
			Add action
System	Every WaveMode_PauseTime seconds	System	Set WaveMode_SpawnCreepCounter to 10
		System	Add 1 to WaveMode_WaveCounter
		System	Add 1 to WaveMode_TotalWaves
		WaveCounter	Set text to WaveMode_TotalWaves
		System	Set WaveTimer to WaveMode_PauseTime
			Add action
+	System	WaveMode_WaveCounter = 1	Add action
	System	WaveMode_SpawnCreepCounter > 0	
+	System	WaveMode_WaveCounter = 2	Add action
	System	WaveMode_SpawnCreepCounter > 0	
+	System	WaveMode_WaveCounter = 3	Add action
	System	WaveMode_SpawnCreepCounter > 0	
+	System	WaveMode_WaveCounter = 4	Add action
	System	WaveMode_SpawnCreepCounter > 0	
+	System	WaveMode_WaveCounter = 5	Add action
	System	WaveMode_SpawnCreepCounter > 0	
+	System	WaveMode_WaveCounter = 6	Add action
	System	WaveMode_SpawnCreepCounter > 0	
+	System	WaveMode_WaveCounter = 7	Add action
	System	WaveMode_SpawnCreepCounter > 0	
+	System	WaveMode_WaveCounter = 8	Add action
	System	WaveMode_SpawnCreepCounter > 0	
+	System	WaveMode_WaveCounter = 9	Add action
	System	WaveMode_SpawnCreepCounter > 0	
	System	WaveMode_WaveCounter = 10	EnemySpawner
			Spawn Enemy_10 on layer "Game" (image point 0)
		Enemy_10	Find path to (EnemyTarget.X, EnemyTarget.Y)
		System	Set WaveMode_WaveCounter to 0
		System	Add 1 to WaveMode_WaveDifficulty_Modifier
		System	Add 0.5 to WaveMode_BossDifficulty_Modifier
			Add action
DEPRECATED !!!! +++ INFINITE MODE -> spawn new enemy every EnemySpawnTime seconds and find path to target +++			
+	System	GameMode = 1	Add action
When an enemy reaches the target, destroy it and reduce the player health by 1			
EnemyFamily	On collision with TargetFamily	EnemyFamily	Destroy
		System	Subtract 1 from Lives
		Lives	Set text to Lives
			Add action

Abbildung 33 - Construct2 - Game_Sheet 04

TURRETS			
Tell the turrets to target the enemies.			
System	On start of layout	Turret_Normal	Add Turret target EnemyFamily
		Turret_Splash	Add Turret target EnemyFamily
Add action			
When shooting, spawn a bullet.			
Turret_Normal	On Turret shoot	Add action	
Turret_Normal	Level = 1	Turret_Normal	Spawn Bullet_01 on layer "Game" (image point 0)
		Audio	Play choose("Laser_Shoot12","Laser_Shoot16") not looping from Sounds at -10 dB (tag "shoot")
Add action			
Turret_Normal	Level = 2	Turret_Normal	Spawn Bullet_02 on layer "Game" (image point 0)
		Audio	Play choose("Laser_Shoot12","Laser_Shoot16") not looping from Sounds at -10 dB (tag "shoot")
Add action			
Turret_Normal	Level = 3	Turret_Normal	Spawn Bullet_03 on layer "Game" (image point 0)
		Audio	Play choose("Laser_Shoot12","Laser_Shoot16") not looping from Sounds at -10 dB (tag "shoot")
Add action			
Turret_Splash	On Turret shoot	Add action	
To prevent bullets that miss flying across the whole screen, destroy them when they've travelled out of range.			
Bullet_01	Bullet distance travelled \geq 100	Bullet_01	Destroy
Add action			
Bullet_02	Bullet distance travelled \geq 100	Bullet_02	Destroy
Add action			
Bullet_03	Bullet distance travelled \geq 100	Bullet_03	Destroy
Add action			
Bullet_Splash_01	Bullet distance travelled \geq 100	Bullet_Splash_01	Destroy
Add action			
Bullet_Splash_02	Bullet distance travelled \geq 100	Bullet_Splash_02	Destroy
Add action			
Bullet_Splash_03	Bullet distance travelled \geq 100	Bullet_Splash_03	Destroy
Add action			

Abbildung 34 - Construct2 - Game_Sheet 05

UI_MENU			
Mouse	On Left button Clicked on Wall	UI_Select	Set position to $(\text{round}((\text{Mouse.X} - 16) / 32) * 32 + 16, \text{round}((\text{Mouse.Y} - 16) / 32) * 32 + 16)$
Add action			
Mouse	On Left button Clicked on TurretFamily	UI_Select	Set position to $(\text{round}((\text{Mouse.X} - 16) / 32) * 32 + 16, \text{round}((\text{Mouse.Y} - 16) / 32) * 32 + 16)$
Add action			
Sound			
Mouse	On Left button Clicked on Button_Sound	Audio	Set toggle silent
		GoogleAnalytics	Track event "UI":"Sound_Toggle"
Add action			
Audio	Is silent	Button_Sound	Set animation to "off" (play from beginning)
		Audio	Pause tag "music"
Add action			
Audio	Is silent	Button_Sound	Set animation to "on" (play from beginning)
		Audio	Resume tag "music"
Add action			
Menu			
Speed Toggle			
Mouse	On Left button Clicked on Button_FastForward	GoogleAnalytics	Track event "UI":"Speed"
		System	Set time scale to $\text{timescale} + 1$
		Audio	Play Blip_Select2 not looping at volume -20 dB (tag "click")
Add action			
System	timescale = 2	Button_FastForw...	Set animation to "x2" (play from beginning)
Add action			
System	timescale = 3	Button_FastForw...	Set animation to "x3" (play from beginning)
Add action			
System	timescale = 4	Button_FastForw...	Set animation to "x4" (play from beginning)
Add action			
System	timescale = 5	System	Set time scale to 1.0
		Button_FastForw...	Set animation to "x1" (play from beginning)
Add action			
Menu Button			
Mouse	On Left button Clicked on Button_Menu	GoogleAnalytics	Track event "UI":"Quit_Dialog"
		Audio	Play Blip_Select2 not looping at volume -20 dB (tag "click")
		Dialog_Quit_Bac...	Set position to (320, 250)
		Dialog_Quit_Bac...	Set Visible
		Dialog_Quit_No	Set Visible
		Dialog_Quit_Yes	Set Visible
		System	Set time scale to 0
Add action			

Abbildung 35 - Construct2 - Game_Sheet 06








Turret_Interaction			
Buy Turret			
Mouse	On Left button Clicked on 	Build_Tower_Normal	Add action
- or -			
Keyboard	On 1 pressed		
UI_Select_Center	Is overlapping 	TurretFamily	Add action
System	Credits ≥ Turret_Normal.BuyValue		
System		Create object Turret_Normal on layer "Game" at $(\text{round}((\text{UI_Select.X} - 16) / 32) * 32 + 16, \text{round}((\text{UI_Select.Y} - 16) / 32) * 32 + 16)$	
System		Create object  Tower_Normal on layer "Game" at $(\text{round}(\text{UI_Select.X}), \text{round}(\text{UI_Select.Y}))$	
System		Subtract Turret_Normal.BuyValue from Credits	
Credits		Set text to "&Credits"	
GoogleAnalytics		Track event "UI": "Buy_Normal"	
Slow Turret			
Mouse	On Left button Clicked on 	Build_Tower_Slow	Add action
- or -			
Keyboard	On 2 pressed		
Splash Turret			
Mouse	On Left button Clicked on 	Build_Tower_Splash	Add action
- or -			
Keyboard	On 3 pressed		
Upgrade Procedure			
Keyboard	On U pressed		Add action
- or -			
Mouse	On Left button Clicked on 	Button_Upgrade	Add action
Turret_Normal UPGRADE			
UI_Select_Center	Is overlapping Turret_Normal		Add action
System	Credits ≥ Turret_Normal.BuyValue*(Turret_Normal.Level+1)		
System		Subtract Turret_Normal.BuyValue*(Turret_Normal.Level+1) from Credits	
Credits		Set text to "&Credits"	
Turret_Normal	Level < 3	Add 1 to Level	
GoogleAnalytics		Track event "UI": "Upgrade_Normal"	
+++ CURRENTLY NO UPGRADE FOR SLOW TOWER AVAILABLE +++			
UI_Select_Center	Is overlapping Turret_Slow		Add action
Turret_Splash UPGRADE			
UI_Select_Center	Is overlapping Turret_Splash		Add action
Sell Turret			
Keyboard	On S pressed		Add action
- or -			
Mouse	On Left button Clicked on 	Button_Sell	Add action

Abbildung 36 - Construct2 - Game_Sheet 07

On Start of Level_Select		
System	On start of layout	Clear Track duration of "Menu": "Select_Duration" Track event "Menu": "Select_Open" Add action
SETTING THE RANDOMS		
LevelSelect_Array	Contains value 6	Add action
Local number randomNumber = 0		
Local number i = 0		
System	While	Set randomNumber to floor(random(1,7))
System	i < LevelSelect_Array.Width	Add action
LevelSelect_Array	Contains value randomNumber	Set value at i to randomNumber
Add action		
System	i = 0	Add action
System	randomNumber = 1	Set position to (180, 117)
Add action		
System	randomNumber = 2	Set position to (324, 117)
Add action		
System	randomNumber = 3	Set position to (468, 117)
Add action		
System	randomNumber = 4	Set position to (180, 266)
Add action		
System	randomNumber = 5	Set position to (324, 266)
Add action		
System	randomNumber = 6	Set position to (468, 266)
Add action		
System	i = 1	Add action
System	i = 2	Add action
System	i = 3	Add action
System	i = 4	Add action
System	i = 5	Add action
Add 1 to i		
Add action		
Buttons		
Mouse	On Left button Clicked on Level_Select_Level_01	Play Blip_Select2 not looping at volume -20 dB (tag "click") Track event "Menu": "Select_Level_01" Stop tracking duration of "Menu": "Select_Duration" Go to Level_01 Add action

Abbildung 37 - Construct2 - Game_Sheet 08

On Start of Menu			
System	On start of layout	Add action	
SETTING THE RANDOMS			
WebStorage	Local key "SQD_UUID" exists	Add action	
Local number	SQD_UUID_RandomInt = 0		
Local number	found = 0		
Local number	q = 0		
System	While	System	Set SQD_UUID_RandomInt to floor(random(1,65536))+floor(random(1,65536))
System	q < SQD_UUID_Array.Width	Add action	
System	found = 0		
SQD_UUID_...	Contains value SQD_UUID_RandomInt	SQD_UUID_Array	Set value at q to SQD_UUID_RandomInt
WebStorage	Set local key "SQD_UUID" to SQD_UUID_RandomInt	System	Add 1 to q
System	Set found to 1	System	Set found to 1
Dev_Info_02	Set text to "SQD_UUID: "&SQD_UUID_RandomInt	Add action	
WebStorage	Local key "SQD_UUID" exists	GoogleAnalytics	Track event "Menu": "Main_Open"
Dev_Info_02	Set text to "SQD_UUID found in WebStorage: "&WebStorage.LocalValue("SQD_UUID")	Add action	

Abbildung 38 - Construct2 - Main_Menu_Sheet 01

On Start of Menu			
Main_Menu			
Mouse	On Left button Clicked on	Menu_Credits_Button	Audio: Play Blip_Select2 not looping at volume -20 dB (tag 'click')
			System: Go to Credits
Mouse	On Left button Clicked on	Menu_Play_Button	Audio: Play Blip_Select2 not looping at volume -20 dB (tag 'click')
			System: Go to LevelSelect
Mouse	On Left button Clicked on	Developer_Log	Dev_Info_00: Set Visible
			Dev_Info_01: Set Visible
			Dev_Info_02: Set Visible
Credits			
Mouse	On Left button Clicked on	Credits_ReturnButton	Audio: Play Blip_Select2 not looping at volume -20 dB (tag 'click')
			System: Go to Menu
MODE SELECT NOW DEPRECATED -> WaveMode is far superior gameplay-wise than InfiniteMode, so the ModeSelect screen won't be used anymore			
Mode_Select			
Game_Over			
NEW GAME -> RESET GLOBALS			
Mouse	On Left button Clicked on	GameOver_Yes	GoogleAnalytics: Track event "Game": "Replay_Yes"
			Audio: Play Blip_Select2 not looping at volume -20 dB (tag 'click')
			System: Reset global variables to default
			System: Go to LevelSelect
QUIT GAME -> GO TO CREDITS			
Mouse	On Left button Clicked on	GameOver_No	GoogleAnalytics: Track event "Game": "Replay_No"
			Audio: Play Blip_Select2 not looping at volume -20 dB (tag 'click')
			System: Go to Credits

Abbildung 39 - Construct2 - Main_Menu_Sheet 02

ANHANG C

Ereignisaktion	Ereignisse gesamt ▼	Ereignisse gesamt ▼
	5.836 % des Gesamtwerts: 100,00 % (5.836)	5.836 % des Gesamtwerts: 100,00 % (5.836)
1. Upgrade_Normal	1.408	24,09 %
2. Buy_Normal	1.313	22,50 %
3. Speed	702	12,03 %
4. Upgrade_Splash	506	8,67 %
5. Buy_Splash	404	6,92 %
6. Buy_Slow	391	6,70 %
7. Sell_Normal	179	3,07 %
8. Main_Open	165	2,83 %
9. Select_Open	130	2,23 %
10. Start	123	2,11 %
11. Quit_Dialog	78	1,34 %
12. Sound_Toggle	71	1,22 %
13. Quit_Dialog_Yes	58	0,99 %
14. Sell_Splash	53	0,91 %
15. Sell_Slow	50	0,86 %
16. Select_Level_05	39	0,67 %
17. GameOver	24	0,41 %
18. Total_Waves	24	0,41 %
19. Select_Level_03	22	0,38 %
20. Quit_Dialog_No	19	0,33 %
21. Select_Level_01	19	0,33 %
22. Select_Level_02	16	0,27 %
23. Select_Level_06	15	0,26 %
24. Select_Level_04	12	0,21 %
25. Replay_Yes	9	0,15 %
26. Select_Abort	5	0,09 %
27. Replay_No	3	0,05 %

Abbildung 40 - Kongregate, Google Analytics, Alle Ereignisse

Ereignisaktion	Ereignisse gesamt ▼	Ereignisse gesamt ▼
	331 % des Gesamtwerts: 100,00 % (331)	331 % des Gesamtwerts: 100,00 % (331)
1. Upgrade_Normal	93	28,10 %
2. Buy_Normal	77	23,26 %
3. Buy_Splash	49	14,80 %
4. Sell_Normal	30	9,06 %
5. Buy_Slow	22	6,65 %
6. Upgrade_Splash	18	5,44 %
7. Speed	16	4,83 %
8. Main_Open	7	2,11 %
9. Select_Open	3	0,91 %
10. Sell_Slow	3	0,91 %
11. Sell_Splash	3	0,91 %
12. Start	3	0,91 %
13. Select_Level_01	2	0,60 %
14. Sound_Toggle	2	0,60 %
15. GameOver	1	0,30 %
16. Select_Level_05	1	0,30 %
17. Total_Waves	1	0,30 %

Abbildung 41 - GameJolt, Google Analytics, Alle Ereignisse

Ereignisaktion	Ereignisse gesamt ▼	Ereignisse gesamt ▼
	18.080 % des Gesamtwerts: 100,00 % (18.080)	18.080 % des Gesamtwerts: 100,00 % (18.080)
1. Upgrade_Normal	4.395	24,31 %
2. Buy_Normal	3.833	21,20 %
3. Speed	3.195	17,67 %
4. Upgrade_Splash	1.383	7,65 %
5. Buy_Splash	1.215	6,72 %
6. Buy_Slow	1.057	5,85 %
7. Main_Open	554	3,06 %
8. Select_Open	408	2,25 %
9. Start	388	2,13 %
10. Sell_Normal	294	1,63 %
11. Sound_Toggle	285	1,47 %
12. Quit_Dialog	180	1,00 %
13. Quit_Dialog_Yes	148	0,82 %
14. Select_Level_05	97	0,54 %
15. Sell_Slow	88	0,48 %
16. GameOver	75	0,41 %
17. Total_Waves	75	0,41 %
18. Select_Level_04	68	0,38 %
19. Select_Level_01	67	0,37 %
20. Select_Level_06	63	0,35 %
21. Sell_Splash	58	0,32 %
22. Select_Level_03	50	0,28 %
23. Select_Level_02	42	0,23 %
24. Replay_Yes	37	0,20 %
25. Quit_Dialog_No	28	0,14 %
26. Select_Abort	14	0,08 %
27. Replay_No	11	0,06 %

Abbildung 42 - Newgrounds, Google Analytics, Alle Ereignisse