



**HOCHSCHULE
MITTWEIDA**
University of Applied Sciences

Fakultät Angewandte Computer- und Biowissenschaften

Professur Medieninformatik

Bachelorarbeit

Untersuchung der Wirkung reduzierter Kernmechaniken eines
Rollenspiels auf die Motivation des Spielers

Nico Daniel Noé

Mittweida, den 4. Dezember 2020

Erstprüfer: Prof. Dr. rer. nat. Marc Ritter

Zweitprüfer: Manuel Heinzig M.Sc.

Noé, Nico Daniel

Untersuchung der Wirkung reduzierter Kernmechaniken eines Rollenspiels auf die
Motivation des Spielers

Bachelorarbeit, Fakultät Angewandte Computer- und Biowissenschaften
Hochschule Mittweida — University of Applied Sciences, Dezember 2020

Referat

In dieser Arbeit geht es um die Untersuchung eines auf wenige Kernmechaniken reduziertes Rollenspiel auf die Motivation des Spielers. Hierfür wurde anhand einer angefertigten Applikation eine quantitative Nutzerstudie mit sechs Probanden durchgeführt. Die Ergebnisse sind dabei gemischt ausgefallen und es konnte kein eindeutiger Beweis auf die Beeinflussung der Motivation durch das Testszenario festgestellt werden.

Name: Noé, Nico Daniel

Studiengang: Medieninformatik und Interaktives Entertainment

Seminargruppe: MI15w1-B

English Title: Investigation of the effect that reduced core mechanics in a Role Playing Game have on the players motivation

Inhaltsverzeichnis

1	Einführung und Motivation	1
1.1	Aufgabenbeschreibung	2
1.2	Zielstellung und Aufbau der Arbeit	3
2	Theoretische Grundlagen	5
2.1	Motivation	5
2.1.1	Eigenschaftstheorien	7
2.2	Minimalismus	12
2.3	Elemente eines Computer-Rollenspiels	17
3	Anforderungen und Konzeption	23
3.1	Anforderungsanalyse der Software	23
3.1.1	Zielbestimmung	23
3.1.2	Anforderung an das Produkt	23
3.1.3	Produkteinsatz	25
3.1.4	Produktübersicht	25
3.1.5	Produktdaten und -leistungen	28
3.1.6	Qualitätsanforderungen nach ISO / IEC25010:2011	29
3.1.7	Minimalismus	29
3.1.8	Quest- und Leveldesign	29
3.1.9	Technische Umgebung	31
3.1.10	Systementwurf	32
3.2	Konzeption der Nutzerstudie	33
4	Implementierung	35
4.1	Programmierung	35

4.2	Objekt- und Levelerstellung	42
4.3	Nutzerstudie	44
5	Evaluation	47
5.1	Ablauf	47
5.2	Auswertung	47
6	Zusammenfassung und Ausblick	55
	Literaturverzeichnis	I

1 Einführung und Motivation

In unserer heutigen Gesellschaft sind Videospiele zu einer der primären Unterhaltungsmedien aufgestiegen. Allein in Deutschland spielen 34,3 Millionen Menschen [tec20, S. 10 f.], das entspricht ca. 41 % der Gesamtbevölkerung. [Bev20] Angefangen hat diese Entwicklung im 20. Jahrhundert, mit der langsamen Ersetzung der analogen Freizeitaktivitäten wie Theater, Literatur oder Sport durch digitale Medien. Anfangs in Form von Radio, Fernsehen und Kino werden diese heutzutage durch Streams und Videospiele ersetzt. Dies beweist der steigende Umsatz von Spiele-Hard- und Software die allein in Deutschland (2019) 6,2 Milliarden Euro betragen, was einer Steigerung von 6% zum Vorjahr entspricht. [tec20, S. 14]

Mit dem Erfolg dieses Mediums kommt die Gier ins Spiel, bei dem von den großen Publishern versucht wird minderwertige Spiele für den gängigen Vollpreis zu verkaufen. Hierbei ist das Sport-Genre am stärksten betroffen bei der durch jährliche Serien, wie die Fußball Simulation *FIFA*, versucht wird alte Mechaniken und Funktionen als neue Features zu verkaufen. Neben dem kurzen Veröffentlichungstakt ist die Schuld bei den Konsumenten zu suchen, da die Applikation sich immer noch zu gut verkauft um Reformen zu erzwingen. [Fif20] Aber auch in den meisten anderen etablierten Spiele-Serien herrscht der selbe Modus Operandi bei den Publishern. Es wird das selbe Spiel mit anderer Aufmachung verkauft und dabei nur Meta-Mechaniken wie Open World oder Rollenspiel-Elemente hinzugefügt. Durch diese Handlungsweise wird häufig die Story und die allgemeine Qualität der Mechaniken vernachlässigt.

Im Gegensatz hierzu steht die Entwicklung während der Anfängen der Videospiele. Bei dem durch die damaligen Hardware-Limitierungen das Design nur mit Kompromissen umgesetzt werden konnte. Dadurch musste man sich auf wenige Mechaniken und simple Grafik-Darstellungen einstellen. Angesichts dieser Beschränkungen wurden Spiele mehr durchdacht und dabei nur qualitative Elemente implementiert, um das Design-Ziel zu erreichen. Ein gutes Beispiel hierzu ist das Spiel *Pong* (Siehe Abbildung 1.1) ein abstrakter Tennis-Simulator, in welcher ein Spieler gegen einen anderen oder dem Computer spielen kann. Hierbei ist die gesamte grafische Oberfläche

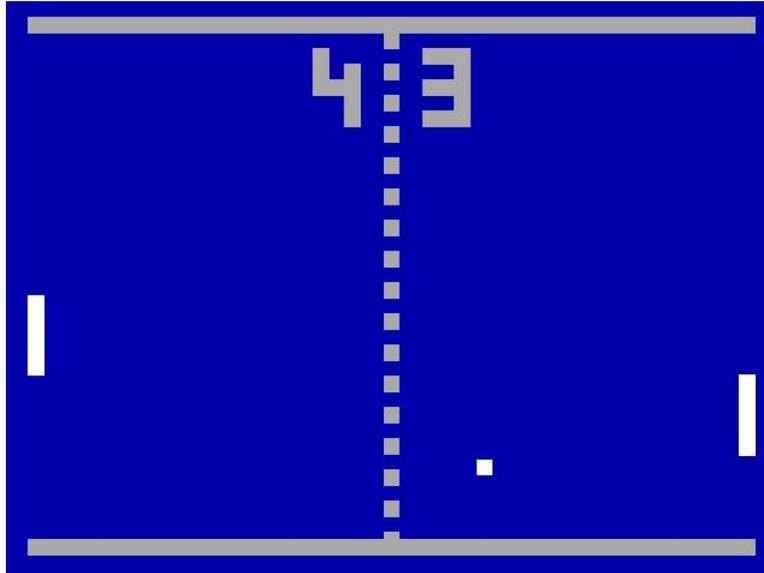


Abbildung 1.1: Screenshot des Spiels Emeritus Pong, eines Pong-Klons

mit geometrischen Formen dargestellt und die Steuerung der Spieler¹ funktioniert durch zwei Tasten, welche diese vertikal nach oben oder unten verschiebt. Ziel ist es den Ball² am Gegenspieler vorbei zu schießen, um Punkte zu ergattern.

Im Angesicht der Gesamtentwicklung von Videospiele stellt sich dadurch die Frage, müssen diese viele Mechaniken und Grafiken zur Unterhaltung verwenden, oder reicht eine Reduktion auf die Essenz durch qualitative Elemente aus?

1.1 Aufgabenbeschreibung

Im Rahmen dieser Arbeit soll ein Rollenspiel, anhand des Reduktions-Gedanken des Minimalismus-Stiles, in der Unity-Game-Engine erstellt werden. Dieses Genre wurde gewählt weil es eine Vielzahl an unterschiedlichen Mechaniken in sich vereint und vom Autor selbst konsumiert wird. Dabei soll untersucht werden inwieweit die Motivation des Spielers durch die Reduktion der Mechaniken beeinträchtigt wird.

¹Die Spieler werden als zwei weiße Balken an dem rechten und linken Bildschirmrand dargestellt

²Der Ball ist als weißes Quadrat dargestellt und pendelt im Spielfeld von einer Seite zu der anderen

1.2 Zielstellung und Aufbau der Arbeit

Ziel dieser Arbeit ist es festzustellen ob in Rollenspiele, eine große Anzahl und Vielfalt an Mechaniken und grafischen Elementen benötigt werden, um den Spieler zu motivieren.

Als Forschungs-Medium soll eine quantitative Nutzerstudie anhand der angefertigten Applikation dienen. Diese soll generelle Meta-Daten und die Meinung der Probanden gegenüber der Applikation sowie den nicht verwendeten Elementen erfassen, welche anschließend ausgewertet wird.

Der Aufbau dieser Arbeit unterteilt sich in sechs Kapitel:

1. In der Einleitung dieser Arbeit soll der generelle Stand und Gedankengang des Autors vermitteln und den damit verknüpften Reduktions-Gedanken. Daneben ist der Rahmen der empirischen Arbeit und dem verfolgten Ziel deklariert.
2. Das zweite Kapitel setzt sich mit den für die Arbeit benötigten theoretischen Grundlagen auseinander. Hier wird klargestellt was Motivation und Minimalismus ist und welche Elemente ein normales Rollenspiel besitzt.
3. Im dritten Kapitel wird die Applikation anhand eines Pflichtenhefts geplant und anschließend darauf aufbauend die Nutzerstudie.
4. Das vierte Kapitel beschäftigt sich mit der generellen Umsetzung und Funktion der Applikation. Hier wird auf die Programmierung, der Level- und Objekt-Erstellung und die Umsetzung der Nutzerstudie eingegangen.
5. Das fünfte Kapitel beschäftigt sich mit dem Ablauf der Studie und die anschließende Evaluation der Ergebnisse.
6. Im letzten Kapitel werden die Ergebnisse zusammengefasst und ein Ausblick auf mögliche zukünftige Bearbeitungen, auf Grundlage dieser Arbeit, gegeben.

2 Theoretische Grundlagen

In diesem Kapitel werden für diese Arbeit benötigten Grundlagen erarbeitet und dargestellt. Dabei wird dargelegt was Motivation ist und aus welchen Komponenten sie besteht, anschließend wird auf die Eigenschaftstheorien der Motivationsforschung eingegangen. Im Anschluss wird die Herkunft und generelle Definition des Minimalismus-Stiles ergründet und eine Verbindung zur Videospiele-Entwicklung hergestellt. Das letzte Unterkapitel beschäftigt sich mit den Elementen eines Computer-Rollenspiels, hier soll für den Leser die generellen Mechaniken und Funktionen eines solchen Spieles deklariert werden.

2.1 Motivation

Das Wort Motivation ist abgeleitet aus dem lateinischen Verb *movere*. Sie reflektiert die Idee, eines inneren Antriebs für Aktivitäten, arbeiten oder das erledigen von Aufgaben. In seiner Essenz ist Motivation ein psychologischer Vorgang, zur Etablierung und Erhaltung von zielgerichteten Handlungsweisen. [DS14, S. 4 f.] Somit liegt die Forschungsaufgabe der Motivationspsychologie darin, die Ausrichtung, Beharrlichkeit, Persistenz und Intensität von diesem systematischen Verhaltens zu ergründen. [JH18, Kap. 1, S. 4]

Das Verhalten einer Person lässt sich prinzipiell in drei verschiedene Faktoren unterteilen:

- Eine grundsätzliche Verhaltenstendenz mit dementsprechenden Bedürfnissen. [JH18, Kap. 1, S. 4]
- Eine Motivdisposition (implizierte Motive), welche von Person zu Person unterschiedlich ist. [JH18, Kap. 1, S. 4]

- Eine Zielsetzung (explizite Motive), welche sich von der jeweiligen Personen ausgesucht und nachgegangen wird. [JH18, Kap. 1, S. 4]

Erstere sind Bedürfnisse und Tendenzen im Sinne der Organismus-Erhaltung und Homöostase¹. Gefolgt von den implizierten oder intrinsischen Motivation bei dem Handlungen aus eigenen Antrieb erfolgt, [JH18, Kap. 1, S. 4 f.] oft aus Freude, Interesse oder Kompetenzerleben. [JH18, Kap. 14, S. 424] Sowie explizite oder extrinsische Motivation bei dem Handlungen durch gewählte Ziele erfolgt. Implizierte und explizite Motivation schließen sich allerdings nicht aus. So können explizite Ziele durch impliziert motivierte Handlungen erfüllt werden, jedoch ist dies nicht der Normalfall. Generell ist die explizite Motivation zur gesamten Handlungssteuerung zuständig, da diese durch Langzeitziele eine Richtung vorgibt und vor Ablenkung schützt. [JH18, Kap. 1, S. 4 f.]

Daneben existiert ein Spezialfall der implizierten Motivation, die Leistungsmotivation oder auch Flow genannt. Bei dem während eine Person, eine herausfordernde Aktivität bewältigt, Motivation entsteht und direkt nach erfüllen des Zieles wieder verschwindet. Der Zustand selber ist ein Balanceakt zwischen Schwierigkeit der ausgeübten Aktivität und den Fähigkeiten der Person, welcher in diesem Zustand vollkommen ausgelastet ist. Neben Verlust des Zeitgefühls und unter Umständen des Selbstbewusstseins, hat es einen positiven Einfluss auf die Konzentration sowie Leistung, da intuitiv auf Probleme reagiert wird. [JH18, Kap. 1, S. 437 ff.]

Neben der rein personenbezogenen Motivation kommt ein Situations-Aspekt hinzu, der Einfluss auf die Handlung hat. Nur durch diesen Schritt lassen sich Verhaltens-Variationen von Personen in verschiedenen Szenarien erforschen und erklären. Nach *Heinz Heckhausen* kommen hierzu noch Anreize in das Motivationsmodell. Diese lassen sich in ein Handlungsablauf-System mit Handlungs-Ergebnis-Erwartung, Situations-Ergebnis-Erwartung und Ergebnis-Folgen-Erwartung einordnen (siehe Abbildung 2.1). Wird ein Bedürfnis allein durch die Situation eines Szenarios erfüllt, hat die Testperson nur ein geringen Anreiz zur Handlung und hat somit eine hohe Situations-Ergebnis-Erwartung vorliegen. Ist diese aber niedrig und das Testsubjekt hat eine hohe Handlungs-Folgen-Erwartung mit einer guten Ergebnis-Folgen-Erwartung wird eine Handlung ausgeführt. [JH18, Kap. 1, S. 4 ff.]

¹Homöostase ist ein offenes dynamisches System in einem Gleichgewichtszustand, welches sich selbst reguliert

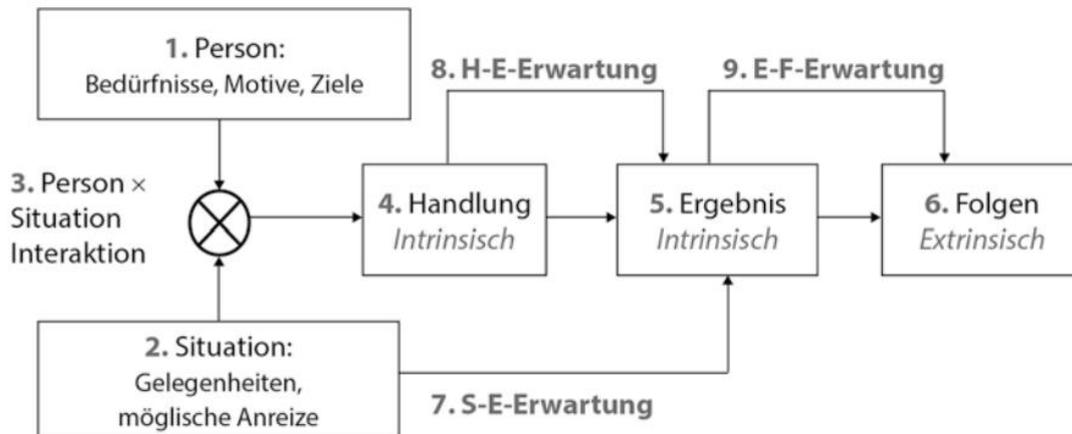


Abbildung 2.1: Determinanten und Verlauf motivierten Handelns [JH18, Kap. 1, S. 5]

Zusammenfassend ist die Motivation ein komplexes Konstrukt von Faktoren, welche zusammenarbeiten um ein gewähltes Ziel zu verfolgen. Neben Anreizen der Situation, der verfolgten Tätigkeit, persönlicher Neigungen und deren Wechselwirkung, werden Ergebnis und Folgen der Handlung miteinbezogen um überhaupt erst eine Handlung auszuführen. [JH18, Kap. 1, S. 7]

2.1.1 Eigenschaftstheorien

In den letzten Jahrzehnten wurden mehrere unterschiedliche Theorien zur Motivation aufgestellt. [Urh08, S. 150] Diese schließen sich jedoch nicht gegenseitig aus, da sie sich auf verschiedene Perspektiven und Aspekte konzentrieren und sich somit ergänzen können. [Kra93, S. 187] Darunter der Ansatz der Eigenschaftstheorie. In dieser wird die Entstehung von Motivation durch situationelle Anreize in Verbindung mit den Eigenschaften, wie etwa Entwicklung und Persönlichkeit einer Person, erklärt. Die Forschung in diesem Bereich ist darauf aus festzustellen warum Menschen in verschiedenen Situationen unterschiedlich agieren. [JH18, Kap. 3, S. 50] Einer dieser Forschungsstränge ist die Bedürfnistheorie mit deren drei Hauptvertreter *McDougall*, *Murray* und *Maslow*. [JH18, Kap. 3, S. 59] In diesen Ansätzen sind Bedürfnisse als Missverhältniss zwischen einem Ist- und Soll-Zustand definiert. [DCM53] Welche sich grundsätzlich in zwei Motivations-Arten unterteilen lassen. Zum einen das anstreben

von Behebung oder Vermeidung eines Defizits und zum anderen das Verwirklichen einer Erwartung oder eines Ziels. [TES52, S. 354]

Motivklassifikation nach Instinkten

Im Jahre 1908 hat *McDougall* den ersten großen Versuch gestartet, Verhalten durch Motivdisposition zu erklären. Seiner Auffassung nach ist jede Handlung auf ein Zweck oder idealen Endzustand ausgerichtet und somit teleologisch. Für ihn sind dafür sieben Merkmale ausschlaggebend. [JH18, Kap. 3, S. 59 f.]

1. Impulsivität der Bewegung. [JH18, Kap. 3, S. 60]
2. Kontinuierliche Intensität einer Bewegung, ohne Verknüpfung des daraus resultierenden Ergebnisses und dessen Kontinuität. [JH18, Kap. 3, S. 60]
3. Richtungswechsel bei einer intentionalen Bewegung. [JH18, Kap. 3, S. 60 f.]
4. Das stoppen der Bewegung, bei Erreichen der gewünschten Änderung. [JH18, Kap. 3, S. 60]
5. Planung der nächsten Umstände, während eine Handlung vollzogen wird. [JH18, Kap. 3, S. 60]
6. Geringe Optimierung von Verhalten, bei Wiederholung unter ähnlichen Szenarien. [JH18, Kap. 3, S. 60]
7. Einheitlichkeit eines Organismus, in seinem Reaktionsverhalten. [JH18, Kap. 3, S. 60]

Diese Eigenschaften von intentionalem Handeln auf spezifische Umstände führt dieser auf *Instinkte* zurück. In seiner Theorie definiert er hierzu *Instinkt* als dreigliedriger Prozess von selektiver Wahrnehmung in Bezug auf verschiedene Zustände des Körpers, einem emotionalen Impuls und eine darauf folgende Aktion der Person. Bei dem nur die emotionale Komponente unverändert bleibt und nicht wie die kognitive oder motorische Verhaltensweisen durch gesammelte Erfahrung verändert werden kann. [JH18, Kap. 3, S. 59 f.]

Person-Umwelt-Bezug

In *Murray's* veröffentlichten Theorie, steht Bedürfnis als *need* im Mittelpunkt. Welche er anhand einer komplexen Studie von 50 Personen in der *Harvard Psychological Clinic* erstellt hat. Mithilfe von den ansässigen Mitarbeitern, Psychologen und Psychater wurden wiederholt die Testsubjekte in unterschiedlichen Situationen gebracht bei dem man ihre Motivation erfragt und beobachtet hat. Unter anderem verwendete er dabei den von ihm entwickelten *Thematischen Auffassungstest* (TAT), der bis heute als einer der wichtigsten Methoden in der Motivationsforschung gilt. Mit einer Großzahl an unterschiedlichen Bedürfnis-Begriffen ist diese neben Verhaltens- und Reaktionsunterschiede in Testszenarien, für die Langzeitforschung gedacht. Durch die Beobachtung einer Person über eine Zeit hinweg wird in diesem System, das Testsubjekt zu einem Organismus, welcher nicht nur auf Situationen reagiert sondern auch aktiv diese ersuchen und sie beeinflussen kann. Durch diesen Schritt hat Murray die einseitige Verhaltens-Erklärung der Eigenschaftstheorien, durch situationelle Anreize in Verbindung mit den Eigenschaften einer Person überwunden. Zum Zweck der Analyse verwendet Murray ein zyklisches System mit einem Organismus *need* und der Situation *press* das als Thema deklariert ist. In diesem kann ein *need*, ein *press* auslösen und umgekehrt. Daneben teilt er die Bedürfnisse *need* in primär (viszerogene) und sekundär (psychogene) ein. Die primären *need* beruhen auf die körperlichen auf Homöostase basierten Bedürfnisse², die psychogene dagegen auf die psychischen. Zudem kann diese in positive, negative, latente oder manifeste Bedürfnisse unterteilt werden. In manchen Fällen kann auch eine Verschmelzung oder Konflikt zwischen Bedürfnissen in einer Situation auftreten. [JH18, Kap. 3, S.61 f.]

Hierarchie-Modell

Im Gegensatz zu *Murray* und *McDougall* fasst *Maslow* in seiner Theorie einzelne Bedürfnisse zu fünf verschiedenen Hierarchien zusammen. [JH18, Kap. 3, S. 65]

1. **Physiologische Bedürfnisse:** Bedürfnisse die zum Erhalt des Organismus dienen und somit der Homöostase. Im Falle eines Organismus sind diese unter anderem Hunger, Durst oder Müdigkeit. [Mas43, S. 372]

²need Trinken, nSchlafen, nEssen etc.

2. **Sicherheit:** Bedürfnisse zur Absicherung der Person, der Umwelt und vor unbekanntem Situationen. Unter anderem wird dies befriedigt durch Besitz einer Unterkunft, eine gesicherte Essens- und Wasserversorgung [Mas43, S. 377 f.] oder in der heutigen Zeit eine berufliche Sicherheit. [MaM17, S. 3]
3. **Soziale Bindung:** Das Bedürfnis für die Beziehung zu sozialen Kontakten, Zugehörigkeit zu Gruppierungen sowie Liebe und Zuneigung. [Mas43, S. 380 f.]
4. **Selbstachtung:** Das Bedürfnis zur Zufriedenstellung des Selbstwertgefühls, durch eigene Leistung, Erfolg oder Respekt von anderen Personen. Diese setzt sich in eigene Faktoren wie Freiheit oder Unabhängigkeit und externe Faktoren wie Ansehen oder Reputation gegenüber anderen Individuen zusammen. [Mas43, S. 381 f.]
5. **Selbstverwirklichung:** Dieses Bedürfnis ist komplett subjektiv und beruht meistens auf zuvor befriedigte Bedürfnisse und die im Leben gesammelte Erfahrung. Es entspricht der Suche nach Glück und Zufriedenheit. Wie etwa ein hauptberuflicher Maler der seine Kreativität mittels erstellen von Bildern auslebt. [Mas43, S. 382 f.]

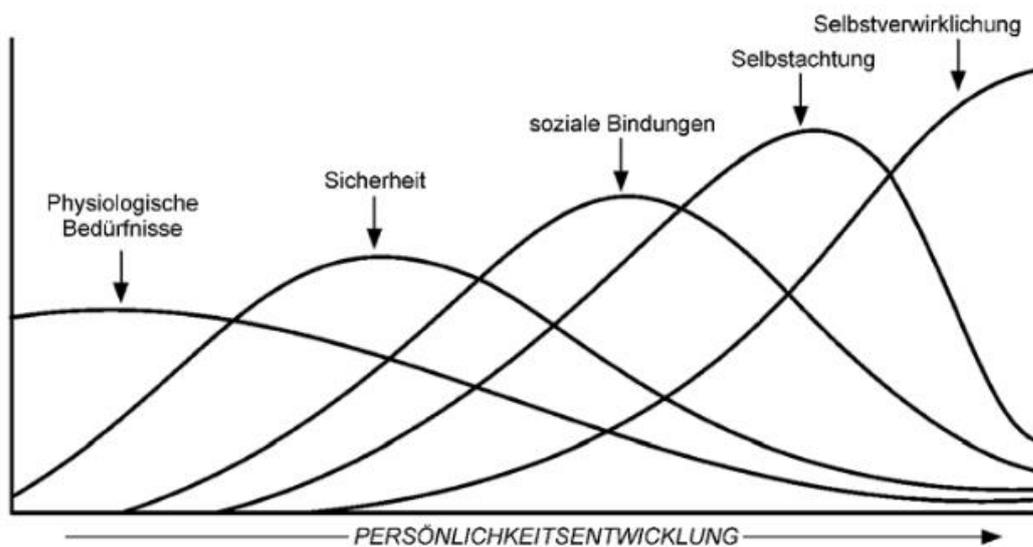


Abbildung 2.2: Hierarchie der Motivgruppen und deren Priorität in der Persönlichkeitsentwicklung [JH18, Kap. 3, S. 65]

Die Bedürfnis-Hierarchien sind nach ihrer Priorität in der Persönlichkeitsentwicklung geordnet (siehe Abbildung 2.2). Bei dem höhere Bedürfnisse durch die Weiterentwicklung einer Spezies hinzu kommen. Daher je höher das Bedürfnis in der Hierarchie, desto weniger hat es mit dem reinen Überleben und Erhalt des Organismus zu tun. Dabei wird nicht zwischen Organismus erhaltenden, instinktuellen oder psychologischen Bedürfnissen priorisiert. Das Handeln wird nur beeinflusst wenn ein Bedürfnis nicht befriedigt ist. Somit ist im Gegensatz zu Murray's und McDouglas Ansätzen das Individuum nicht zur Handlung getrieben (*pushed*), als viel mehr zu einer Tilgung des Bedürfnisses angezogen oder bewegt (*pulled*). Jedoch müssen zuerst die niederen Bedürfnisse getilgt werden, bevor die nächst höhere Bedürfnis-Hierarchie aktiviert und das Handeln beeinflusst werden kann. Im Umkehrschluss, wenn ein Bedürfnis der unteren Hierarchien aktiviert wird muss erst dieses behoben werden, bevor das davor verfolgte Bedürfnis einer höheren Hierarchie, das Handeln beeinflussen kann. Daher kann die Selbstverwirklichungs-Hierarchie erst in Kraft treten und das Verhalten einer Person bestimmen wenn alle anderen Bedürfnisse getilgt sind, was einem teleologischen Aufbau entgegen dieses Wertes an der Spitze entspricht (siehe Abbildung 2.3). [JH18, Kap. 3, S.65 f.]



Abbildung 2.3: Maslows Hierarchien-Pyramide [MaM17]

Infolge kann angenommen werden das Unterhaltungsmedien, insbesondere in den weiterentwickelten Formen wie Fernsehen, Videospiele oder Streams, der Selbstverwirklichungs Hierarchie zuzuordnen sind. Da eine Entwicklung dieser nur in einer Gesellschaft stattfinden kann, welche die niederen Hierarchien weitgehend befriedigt hat und dementsprechend hoch entwickelt ist. Des Weiteren dient der generelle Konsum der Unterhaltung oder Zeitvertreibe und ist somit intrinsischer Natur.

2.2 Minimalismus

Der Term Minimalismus und der damit verbundene Reduktions-Gedanke wurde schon in den 1950ern in den Medien verwendet, unter anderem in Kunst, Musik und Literatur. Als Kunststil entwickelte sich diese aber erst in den 1960ern, [Mar14, S.6 f.] als direkte Gegenbewegung zur Pop- und Op-Art in New York (USA) [Str00, Kap. A, S. 4]. Durch den Kunstkritiker *Richard Wollheim*, der junge Künstler und ihre Kunstform als repetitiv und monoton bezeichnete wurde der Begriff *Minimal Art* für diesen Stil geprägt. [Per14, Kap. Barbara Rose, S.1555f] Als Definition kann Minimalismus als Reduktion auf das wesentliche, in Verbindung mit der Klarheit und Eindeutigkeit der gewählten Form, angesehen werden. In der Kunst kommt hierzu noch die Einfachheit der Struktur und Textur der angefertigten Skulpturen und Gemälden. [Str00, Kap. A, S. 4] Bei denen die Künstler auf abstrakte Weise, geometrische Figuren wie Dreiecke, Quadrate, Kreise oder Würfel als Ausgangspunkt ihrer Arbeit verwenden (siehe Abbildung 2.4). Daneben ist stilistisch auf schlichte und konservative Verwendung von Farben in der Komposition zu achten. [Str00, Kap. A, S. 5]

Selbst in unserer modernen Gesellschaft ist Minimalismus in verschiedenen Bereichen anzutreffen. Sei es in der Werbung bei dem durch Reduktion auf den Inhalt, zum Nachdenken angeregt wird, [Str00, Kap. A, S. 2] oder sogar einer Lebensphilosophie in der im Verhältnis zu "normalen" Menschen diese fast besitzlos sind. [JFM12, S. 7 f.] Aber auch in Videospiele ist dieser Kunststil vertreten. Generell muss in diesem Fall zwischen Limitierung und Intention unterschieden werden. Ältere Spiele, insbesondere von den Anfängen der Videospiele-Entwicklung, wie etwa *Pong* (1972) (siehe Abbildung 1.1), einen abstrakten Tennis-Simulator der mit einfachen geometrischen Formen als grafische Oberfläche und einer Zwei-Tasten-Steuerung zwar der Definition eines Mediums der Minimalistik entspricht, jedoch sein Aussehen und

seine Mechaniken der damaligen Hardware-Limitierung geschuldet ist. Somit wurde dieser nicht intentional auf Minimalismus entwickelt, sondern ist viel mehr als ein Produkt der damaligen Zeit anzusehen. Die zielgerichtete Entwicklung von Spielen mit Fokus auf Minimalismus, kann dagegen in drei Entwicklungsstränge unterteilt werden.

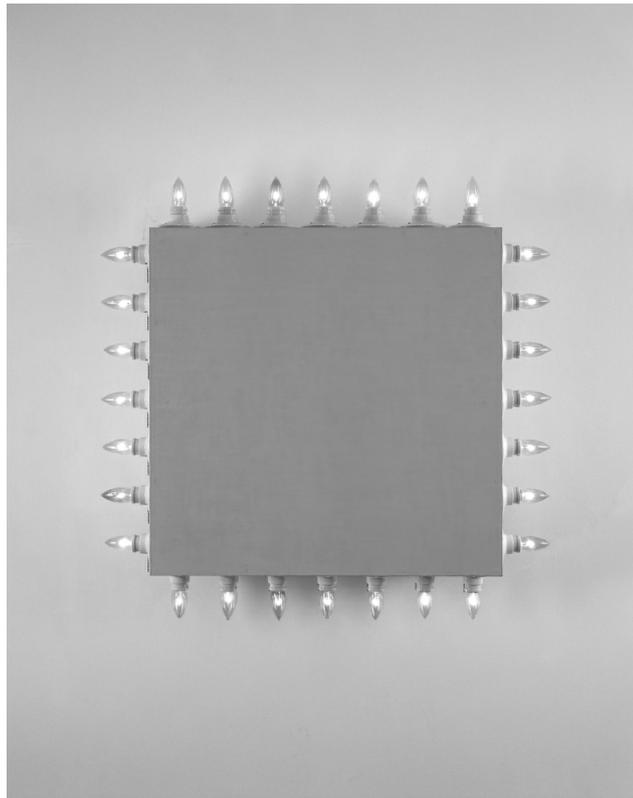


Abbildung 2.4: Minimalistische Kunst, Dan Flavin: Icon V (Coran's Broadway Flesh) 1962. Oil and mixed media, 415/8 * 4158 * 97/8 in. [Per14, Kap. Barbara Rose, S.1560]

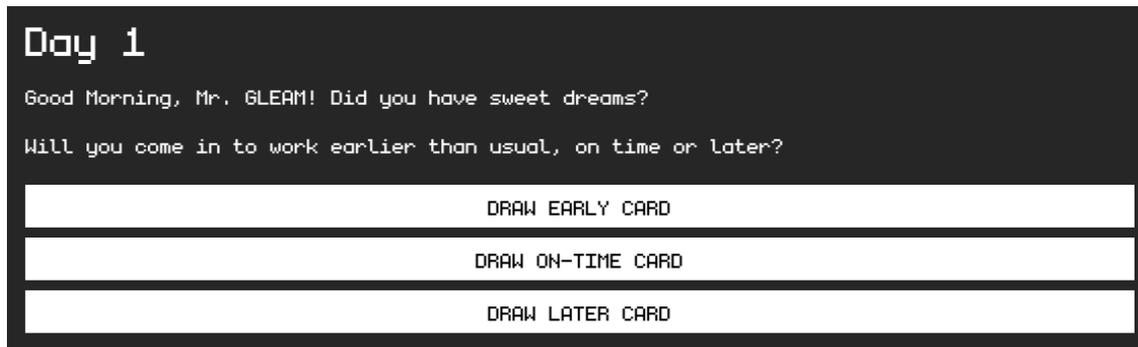


Abbildung 2.5: Screenshot des Textabenteuers Dungeons & Deadlines

Narrative

In einem auf die Narrative entwickeltes Spiel dienen ein Großteil aller Spielelemente zur Erhaltung, Progression und Lenkung der Story. Ähnlich wie in schriftlich verfassten Geschichten sind die Regelungen *Show, don't tell* und *Less is more* im Gamedesign anzuwenden. Hierbei sollte vermieden werden den Spieler mit zu viel Informationen zu überhäufen und stattdessen Fragen nicht beantwortet zu lassen. [Bon15]

Das beste Beispiel einer solchen Entwicklungsweise sind Textabenteuer, bei dem alle Elemente auf das erzählen einer Geschichte abgezielt sind. Der Spieler muss in vorgegebenen Szenarien zwischen mindestens zwei verschiedenen Aktionen entscheiden (siehe Abbildung 2.5), welche direkte oder indirekte Auswirkungen auf das Ende haben. Je nach verwendeten Gamedesign können Spieler, bei Fehlentscheidungen direkt ausscheiden und müssen von vorne anfangen. Wie etwa in *Dungeons & Deadlines* bei dem man bei zu vielen Fehlritten gefeuert wird. In diesem System der Multiple-Choice-Interaktion existiert mindestens ein positives und negatives Ende, welches durch die korrekte beziehungsweise inkorrekte Wahl des Spielers getriggert wird.

System

In einem auf das System entwickeltes Spiel ist die Interaktion zwischen Spiel und Spieler so gering wie möglich zu halten. Im Sinne von *More is less* ist das implementieren von qualitativen Funktionen zu bevorzugen. Generell sind solche Systeme

einfach zu lernen aber hart zu meistern. Hierbei kann das Drücken der Funktionen in Variation, unter anderem Timing, Rhythmus, Festhaltung oder mehrfach Druck der Tasten verschiedene Aktionen auslösen. Bei denen nur mit Benutzung der wenigen Funktionen, der Spieler immer komplexere Puzzle und Level meistern muss. [Bon15]

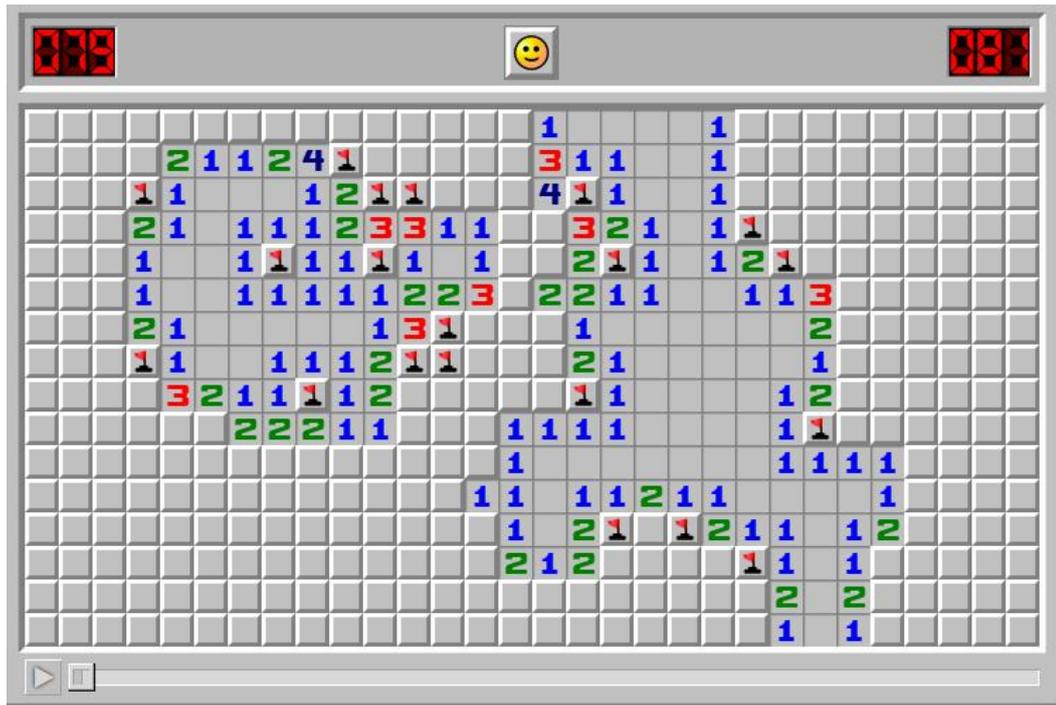


Abbildung 2.6: Screenshot des Spiels Minesweeper

Eines dieser Spiele ist *Minesweeper*, ein 2D-Spiel bei dem der Spieler in einem Bombenfeld, alle Bomben freilegen und markieren muss. In dieser erfolgt die Interaktion, mit den Menü und dem Spiel, mit der linken und rechten Maustaste. Im Spiel selbst legt der Spieler ein Feld frei oder löst eine Bombe mit der linken Maustaste aus. Dagegen kann mit der rechten ein Feld, auch ohne Bombe, als solche deklariert werden. Das Erkennen der Bomben erfolgt mit Zahlen welche freigelegt werden (siehe Abbildung 2.6), diese geben an wie viele Bomben im Umfeld des Feldes liegen³. Mit dieser Information muss sich der Spieler Feld für Feld durcharbeiten bis er alle Felder

³Es können maximal acht Bomben im Umfeld eines Feldes liegen, ähnlich wie die fünf im Ziffernblock / Num-Pad

freigelegt sowie alle Bomben deklariert hat oder die selbige auslöst und verliert. Die Schwere des Spiels ergibt sich durch die maximale Anzahl an Feldern und Bomben.

Ästhetik

In einem auf die Ästhetik entwickeltes Spiel ist der visuelle Stil das wichtigste Element. In dem Farben nur konservativ und mit Kontrast zur Hervorhebung verschiedener Ebenen verwendet werden. Auch Verwendung von grafischen Effekten oder Modellen anstatt einiger User-Interface-Elemente unterstützt den Gesamteindruck. Daneben sollte der Hauptaugenmerk auf die Belichtung und Animationen gesetzt werden, die unabhängig zur Darstellungsqualität des Spiels für Eindruck sorgen kann. [Bon15]

Das perfekte Beispiel hierzu ist *Limbo* ein 2D-Sidescroller welcher nur mittels Schwarz- und Grautönen eine bedrückende Atmosphäre erzeugt, welche die brutale Welt im Spiel perfekt widerspiegelt (siehe Abbildung 2.7). Durch geschickten Einsatz der Belichtung und Anordnung der Objekte im Spiel wird zusätzlich eine Illusion von tiefe in die 2D-Welt projiziert. Daneben besitzt diese Applikation auch eine minimalistische Steuerung, bei dem der Spieler nur springen, greifen und nach rechts oder links gehen kann.



Abbildung 2.7: Screenshot des Spiels Limbo

2.3 Elemente eines Computer-Rollenspiels

In einem Computer-Role-Play-Game oder kurz CRPG sind Spiele bei dem der Spieler die Kontrolle von einem oder mehreren Charaktere übernimmt und diese durch eine Serie von dem Computer gesteuerten Quests beziehungsweise Aufgaben führt, in der die Gewinnbedingung das erledigen aller Hauptquest ist. Währenddessen muss der Spieler, für das Genre charakteristische Herausforderungen erledigen. Diese sind neben Logistik, Ökonomie, Erkundung und Rätsel durch taktische Kämpfe gegeben. Über das gesamte Spiel hinweg wächst dadurch der Charakter in Stärke und Fähigkeiten, welches ein Hauptmerkmal des Genres ist. [Ada14, S. 5] Um einige Hauptmechaniken der Computer-Rollspiele nachzuvollziehen muss zunächst auf die Entstehung hingewiesen werden. Zu dieser kam es als Spieler von analogen Kriegsspielen, fantasy Elemente in diese integrierten. Unter anderem auch *Gary Gygax* der 1971 in das analoge Kriegspiel *Chainmail*, Elemente der *Herr der Ringe* Serie von *J.R.R. Tolkien* als Basis für seine fantasy Kriegsführung verwendete. [Tre11, S. 51 f.] Dies endete damit das er 1974 das Genre Rollenspiel mit Veröffentlichung seines ersten Regelwerks und Spiels der *Dungeon & Dragons*⁴ Serie erschaffen hat. [Bow10, S. 11]

Kernmechaniken

Die Mechaniken und Kernmechaniken⁵ stellen die verfügbaren Funktionen im Spiel dar. [Ada14, S. 26 f.] Je nach Spieletyp und Genre können sich diese unterscheiden. Bei Shootern ist generell eine Ressourcen-Kernmechanik für Munition und eine Ziel- und Bewegungs-Kernmechanik vorhanden. In Computer-Rollenspielen jedoch kommt es auf den Reinheitsgrad des Rollenspieles an. Da die ursprünglichen CRPG's eine direkte Ableitung der Pen-&-Paper-Rollenspiele sind und ein Großteil ihrer Mechaniken übernehmen. [Tre11, S. 369 f.] Die heutigen Hybrid-Rollenspiele, wie etwa Action-RPG's, vereinfachen das System und fordern vom Spieler mehr Koordination und physischen Einsatz. [Ada14, S. 12 f.]

Zufall

Der Zufall in CRPG's spielt eine entscheidende Rolle in den meisten Bereichen des Spiels. Neben direkten Einflüssen der Charakter-Attribute und Skillwerte, entschei-

⁴Dungeon & Dragons ist ein Pen-&-Paper-Rollenspiel, in der Spieler ihr Abenteuer mit Stift und Papier anhand des Regelwerks ausspielen.

⁵Kernmechaniken sind für die Hauptfunktion des Spiels zuständig

det sie ob eine Aktion fehlschlägt oder gelingt [Ada14, S. 26 f.] In den meisten Fällen wird der Zufall durch einen Pseudo-Zufallsgenerator imitiert. Dieser erzeugt mittels einem Algorithmus und einer Startzahl, auch Seed genannt, eine Zahlenreihe welche anschließend zum ermitteln der Werte verwendet wird. Im Gegensatz zu echten Zufallszahlen erspart dies Rechenleistung und somit Performance. [SB11] Diese Mechanik ist direkt abgeleitet aus dem Pen-&-Paper-Rollenspiels und ersetzt das erwürfeln des Zufalls durch den Spieler. [Ada14, S. 26 f.]

Charakter-Attribute

In einem Rollenspiel sind die Attribute als Kernstück der Individualisierung des Charakters anzusehen. Sie entscheiden welche Art von Fähigkeiten, Rüstung oder Aktionen der Spieler verwenden kann. [Ada14, S. 28 f.]

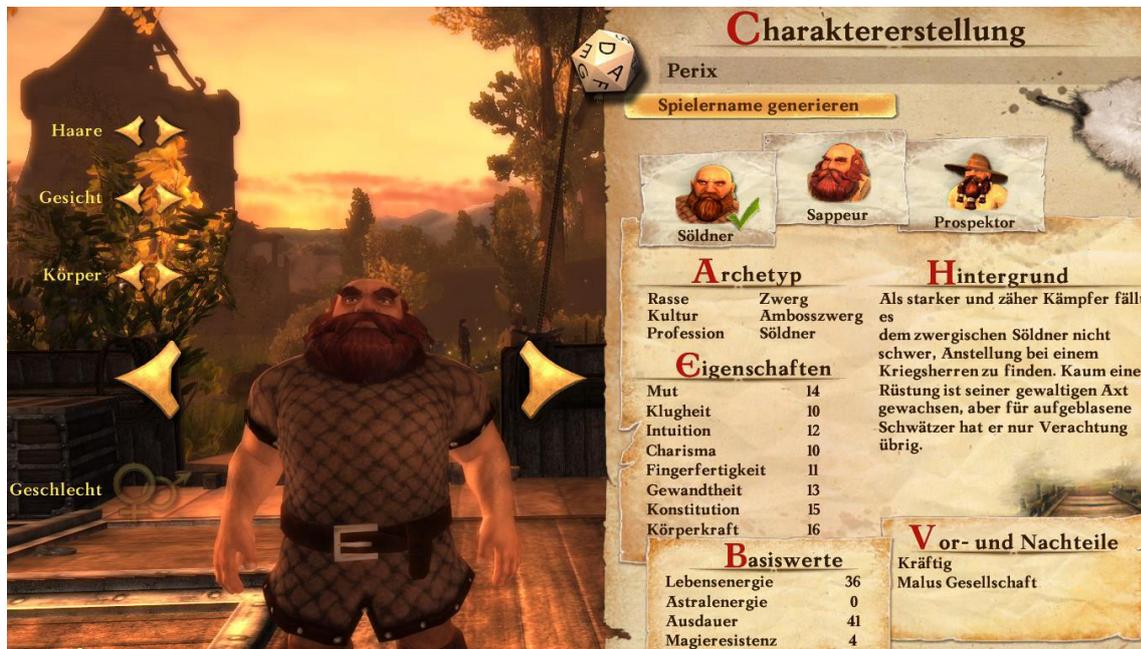


Abbildung 2.8: Screenshot vom Spiel Drakensang am Fluss der Zeit, im Charakter-Erstellungs-Bildschirm

- **Charakterisierende Attribute** werden meistens zu Beginn eines Spiels ausgewählt (siehe Abbildung 2.8). [Ada14, S. 29] Beginnend mit Auswahl des Aussehens, Geschlechts und Rasse. Erstere sind rein kosmetischer Natur, die Rasse jedoch kann je nach verwendetem System spezifische Mali und Boni für Skills oder Attribute bieten. [Ada14, S. 26 ff.] Anschließend kommt die

Klassenwahl, welche über die grundlegende Spielweise des Charakters entscheidet. Diese beeinflusst ob Nahkampf, Fernkampf oder Magie als Hauptangriff beziehungsweise überhaupt verwendet werden kann. [Ada14, S. 32 f.]

- **Physische Attribute** die die Stärke, Beweglichkeit und Belastbarkeit des Charakters darstellen. Diese beeinflussen die Einschränkungen von ausgerüsteter Rüstung, die maximale Gesundheit und Gewichts- / Trage-Kapazität. Daneben sind sie im Kampf für die Ausweichchance zuständig. [Ada14, S. 33 f.]
- **Mentale Attribute** die die Intelligenz und Weisheit des Charakters darstellen. Diese beeinflussen die mentale Widerstandskraft, Auffassungsgabe und Lernkapazität. Daneben kann bei zu niederem Intelligenz-Wert der Magie-Einsatz limitiert sein. [Ada14, S. 33 f.]
- **Moralische Attribute** die die Einstellung des Charakters gegenüber der Welt darstellen. [Ada14, S. 33 f.]
- **Soziale Attribute** die die soziale Kompetenz des Charakters darstellt. Diese beeinflussen Gespräche und deren Verlauf. [Ada14, S. 33 f.]
- **Status Attribute** zeigen den derzeitigen Zustand des Charakters an. Neben derzeitiger Kartenposition, dem Gesundheitszustand und Reputation ist das Inventarsystem mit den ausgerüsteten Equipment dem Status zuzuordnen. Ebenfalls beherbergt dieser auch die Attribute die die Progression widerspiegelt. Unter anderem Erfahrungspunkte und die damit verknüpften Charakterlevel. Erstere werden bei Lösung von Herausforderungen dem Spieler gutgeschrieben. Sie dienen zur Darstellung des Fortschritts und bei überschreiten bestimmter Punktestände als Trigger zur Erhöhung des Levels. Je nach System kann anschließend ein oder mehrere Punkte verwendet werden, um die Attribute zu erhöhen. Überdies kann das Level ausschlaggebend sein verschiedene Gebiete und Funktionen im Spiel freizuschalten. [Ada14, S. 35 f.]

Magie

Ein charakteristisches Element von Rollenspielen ist Magie oder vergleichbare Kräfte die Einfluss auf die Welt ausüben können, wie etwa fortgeschrittene Technologie im Science-Fiction-Genre. Meistens ist diese Mechanik bestimmter Klassen wie Magier

vorbehalten. Grund hierfür ist, um eine Variation von verschiedenen Klassen und damit neben Wiederspielwert, verschiedene Gruppenzusammensetzungen zu erreichen. Die Schwierigkeit hierbei ist der Balanceakt zwischen wie Magie gewirkt wird, deren Auswirkungen und Limitierungen. Neben den Typen, wie Angriffs-, Unterstützungs- oder Schwächungszauber, muss die Zauberzeit, die Kosten und das Resultat definiert werden. Die Ressource hierzu ist in den meisten Fällen Mana, welches an ein Mentales Attribut wie Intelligenz für die Gesamtmenge und an ein Status Attribut für die derzeitige Menge gebunden ist. Je nach System kann diese sich automatisch oder durch Verbrauchsgüter regeneriert werden. [Ada14, S. 41 ff.]

Skills

Neben den Attributen, sind Skills oder Fähigkeiten ein Hauptelement der Individualisierung in Rollenspielen.



Abbildung 2.9: Screenshot vom Spiel Drakensang am Fluss der Zeit, im Skillbaum-Fenster

Diese sind in der Regel durch die gewählte Klasse, Attribute oder auch Spezialisierung limitiert, wie etwa Magie für Magieklassen. Die Fähigkeiten an sich, kommen als Lernmechanik im Spiel vor und ermöglichen es dem Spieler Angriffe zu verbessern oder neue zu erwerben. Je nach System sind sie in Verbindung durch verschiedene Mentoren gegen Ressourcen wie Gold zu kaufen oder mit Fähigkeitspunkte und deren Ausgabe in einem Skill- / Fähigkeitenbaum. Diese können unter anderem durch ein Levelaufstieg erspielt werden. In der Regel sind Fähigkeiten in einen solchen integriert und können bei Erlernung bessere freischalten (siehe Abbildung 2.9). [Ada14, S. 44 ff.]

Quests

Quests oder Aufgaben, welche meist in Verbindung eines Questsystems vorhanden sind, dienen als Bindeglied zwischen der Narrative des Spiels und deren Mechaniken sowie Regeln. [How08, S. 1] Im Spiel selbst sind die Quests als eine predefinede Sequenz von Situationen vorzufinden. Bei dem der Spieler, je nach System bei Aktivierung beziehungsweise Annahme, diese anschließend bewältigen oder beeinflussen kann. [Jes05, S. 43 ff.] In einem durch Quest verfolgte Narrative ist generell als Inszenierung anzusehen. Bei dem der Spieler nur vom Entwickler definierte Interaktion vornehmen kann und somit nicht komplett interaktiv ist. [How08, S. 1] Daher kann eine Quest im gesamten, als zielorientierte Herausforderungs-Bewältigung des Spielers mittels eines Avatars angesehen werden. [Aar04, S. 368] Der Avatar hierbei ist die vom Spiel zur Verfügung gestellte Figur, welcher der Spieler Steuern kann.

3 Anforderungen und Konzeption

In diesem Kapitel werden die, für die empirische Studie benötigten Elemente Konzipiert. Daher wird Anhand eines Pflichtenheftes die Applikation geplant und anschließend darauf aufbauend die Nutzerstudie. Hierzu sei gesagt, dass von einem zuvor bearbeiteten Projekt Code-Abschnitte in diesem Projekt Verwendung finden sollen.

3.1 Anforderungsanalyse der Software

Der Prozess der Softwareentwicklung ist hochkomplex und soll hier nur vereinfacht dargestellt werden, mittels Ableitung auf ein Pflichtenheft. Bei dem genau analysiert wird welche Komponenten und Funktionen diese integriert haben muss, erstrebenswert sind und welche Aspekte durch das Endprodukt nicht zu vermitteln sind.

3.1.1 Zielbestimmung

Die Applikation soll als Teil einer Nutzerstudie, als Rollenspiel mit reduzierten (Kern-)Mechaniken erstellt werden. Bei dem der Spieler Rollenspiel typische Quests mit dementsprechenden Funktionen ausführt und mithilfe eines Questsystems durch das Level geführt wird. Dabei ist diese in der Unity-Engine zu erstellen und soll das Questsystem eines vorherigen Projektes verwenden.

3.1.2 Anforderung an das Produkt

Folgende Kriterien muss die Applikation vorweisen:

- Die Applikation ist mit der Unity-Engine erstellt.

- Es wird die Programmiersprache C# verwendet.
- Die Handlung findet in einer 3D-Umgebung statt.
- Die Applikation hat mindestens ein Level.
- Der Spieler muss sich frei bewegen können.
- Der Spieler wird anhand von Quests durch das Level geführt.
- Es gibt mindestens drei verschiedene Quests, mit unterschiedlichen Funktionen.
- In der Arbeit wird sich auf das Gameplay und die damit erzeugte Motivation fokussiert.
- Es sind feindliche Einheiten / Gegner implementiert.
- Der Spieler kann angreifen.
- Der Spieler kann mit der Welt interagieren.
- Alle grafischen 3D-Objekte müssen aus geometrischen Grundkörpern bestehen.
- Alle Texturen müssen schlicht gehalten werden.

Folgende Kriterien sind für die Applikation wünschenswert:

- Die Quests und deren Funktionen machen Spaß und motivieren den Spieler.
- Quests können leicht neu erstellt oder editiert werden.
- Quests können leicht getestet werden.
- Die Steuerung ist unkompliziert und leicht erlernbar (intuitiv).
- Das Spiel ist mit Tastatur oder Maus steuerbar.

Folgende Abgrenzungskriterien sollen gelten:

- Es ist kein Sound oder Musik implementiert.
- Bis auf das Questsystem, ist kein User-Interface-Element (UI) implementiert.
- Es gibt kein Ingame- oder Hauptmenü.
- Es gibt kein Tutorial.

- Es wird nur das nötigste an Funktionen implementiert.
- Es wird nur auf eine Windows Kompatibilität geachtet, daher keine Portierung auf andere Plattformen.
- Es gibt kein Feedback für den Spieler (Beendigung des Spiels, wenig Gesundheit etc.).

3.1.3 Produkteinsatz

Der Anwendungsbereich dieser Applikation ist als Testszenario für eine Nutzerstudie mit dem Thema Untersuchung der Wirkung reduzierter Kernmechaniken eines Rollenspiels auf die Motivation des Spielers.

Das Spiel ist ab der Altersgruppe der fortgeschrittenen Mittelschule (7+ Klasse), beziehungsweise des sekundären Bildungsbereichs, ab 12 Jahren geeignet. Da Gewalt durch das schwingen eines abstrakten Schwertes gegen mehrere Gegner in Kugelform vorkommt.

Betriebsbedingung:

- Verwendung eines Windows-PC.
- Besitz und Verwendung einer Maus und Tastatur.

3.1.4 Produktübersicht

Ingame-Produkt-Funktionen

/PF010/	Funktion	Durch die Spielwelt bewegen
	Akteur	Spieler
	Beschreibung	Der Spieler kann sich mittels der Tastatur durch die Spielwelt bewegen, dabei folgt ihm die Kamera.

/PF020/	Funktion	Rotieren der Spielfigur
	Akteur	Spieler
	Beschreibung	Der Spieler kann sich mittels der Tastatur nach rechts oder links rotieren, dabei folgt ihm die Kamera.
/PF030/	Funktion	Angreifen
	Akteur	Spieler
	Beschreibung	Der Spieler kann durch Tastendruck eine Angriffs-Animation auslösen.
/PF040/	Funktion	Interagieren
	Akteur	Spieler
	Beschreibung	Der Spieler kann durch Tastendruck mit der Umwelt interagieren.
/PF050/	Funktion	Quest-Interface Interaktion
	Akteur	Spieler
	Beschreibung	Der Spieler kann durch die Maus mit dem Quest-Interface interagieren
/PF060/	Funktion	Questlog
	Akteur	Spieler
	Beschreibung	Der Spieler kann durch Tastendruck das Questlog öffnen.
/PF070/	Funktion	Positions-Reset
	Akteur	Spieler
	Beschreibung	Der Spieler kann per Tastendruck zum Anfang des Levels zurückkehren.
/PF080/	Funktion	Questsystem
	Akteur	Unity
	Beschreibung	Das Questsystem schaltet und aktualisiert sich automatisch, umso an und Abgabe von Quests zu ermöglichen.

/PF090/	Funktion	Questobjekt
	Akteur	Unity
	Beschreibung	Unity registriert durch ein Skript, die (Teil-)Erfüllung einer Quest.
/PF100/	Funktion	Gegner
	Akteur	Unity
	Beschreibung	Unity lenkt durch einen Navmesh-Agenten den Gegner in den Spieler (Bodyslam).

Entwickler-Funktionen

/PF110/	Funktion	Quest-Erstellung
	Akteur	Entwickler
	Beschreibung	Der Entwickler kann innerhalb des QuestManagers neue Quests in einer Liste erstellen und die damit verbundenen Texte, Quest-Bedingung, Folgequest und IDs.
/PF120/	Funktion	Quest-Verwaltung
	Akteur	Entwickler
	Beschreibung	Der Entwickler kann innerhalb des QuestManagers direkt Quests editieren.
/PF0130/	Funktion	Quest-Zuweisung
	Akteur	Entwickler
	Beschreibung	Der Entwickler kann einem beliebigen QuestGiver-Objekt in der Spielwelt, entweder eine Annahme oder Abgabe einer Quest zuweisen (oder beides).
/PF0140/	Funktion	Gegner und Spieler Verwaltung
	Akteur	Entwickler
	Beschreibung	Der Entwickler kann die Geschwindigkeit des Spielers und Gegners, sowie den Aufmerksamkeits-Radius des Gegners einstellen..

3.1.5 Produktdaten und -leistungen

Produktdaten

- /PD010/ Datengruppe 3D-Objekte (max. 600)
 Beschreibung Alle erstellte Objekte im Level, Bäume, Pilze, Steine, Gegner und Spielfigur.
- /PD020/ Datengruppe Texturen (max. 20)
 Beschreibung Alle Unity Materialien / Texturen für die 3D-Objekte.
- /PD030/ Datengruppe Skripte (max. 20)
 Beschreibung Der selbst erstellte Programmcode.
- /PD040/ Datengruppe Level (max. 5)
 Beschreibung Die selbst erstellten Umgebungen.

Produktleistung

- /PL010/ Das Spiel muss flüssig laufen.
- /PL020/ Das Spiel soll trotz des minimalistischen Stils detailliert und immersiv sein.
- /PL030/ Das Questsystem läuft fehlerlos.
- /PL040/ Die Funktionen / Mechaniken der Quests beeinträchtigen nicht den Spielfluss.

3.1.6 Qualitätsanforderungen nach ISO / IEC25010:2011

	Sehr Gut	Gut	Normal	Nicht relevant
Funktionale Eignung	X			
Kompatibilität				X
Sicherheit				X
Zuverlässigkeit	X			
Benutzbarkeit	X			
Leistungseffizienz			X	
Wartbarkeit				X
Portabilität				X

3.1.7 Minimalismus

Die Applikation und die darauf folgende Nutzerstudie basieren in ihrer Gesamtheit auf Minimalismus beziehungsweise den Reduktions-Gedanken. Hierbei soll anhand der Recherche des Minimalismus-Stils das TestszENARIO gestaltet werden. Auf Systemseite soll Spieler-Interaktion auf das nötigste reduziert und versucht werden ob mehrere Funktionen mit einer Taste bedienbar sind. Wie etwa Interaktion mit der Angriffstaste. Daneben ist bei der Ästhetik darauf zu achten nur geometrische Grundkörper als Ausgangsmaterial sowie nur wenige und einfarbige Texturen für das Leveldesign zu verwenden, um somit einen low-poly Stil zu erreichen.

3.1.8 Quest- und Leveldesign

Um das Spiel abzuschließen muss der Spieler drei Quests, welche für das Genre typisch sind, durch das Questsystems bewältigen. Hierbei soll das Level anhand dieser gestaltet werden. Die Aufgaben haben hierzu einen komplett linearen Verlauf und es kommt zu keiner möglichen freien Erkundung des Levels.

Questdesign 1

Mit Beginn des Spieles findet sich der Spieler in einer kleinen Lichtung im Wald. Vor ihm ist eine Ruine und in dessen Mitte steht ein NPC¹ in einem Pilzkreis, mit einem Ausrufezeichen über seinem Kopf. Der Spieler kann anschließend zu diesem hinlaufen und mit seinem getreuen Schwert mit den NPC interagieren, dabei kann die Quest angenommen werden. Die erste Quest ist das finden des nächsten Questgebers. Hierbei muss der Spieler, wie in einem Platformer, über Hindernisse springen. Auf der anderen Seite der Ruine hinter einer Mauer kann anschließend die Quest abgegeben werden.

Questdesign 2

Die zweite Quest entspricht der RPG-typischen Sammelquest bei dem man eine Menge x von einem Gegenstand dem Questgeber bringt. In diesem Fall soll man fünf braune Pilze dem nächsten Questgeber überreichen. Die Quest findet auf einer baumreichen Lichtung statt, bei dem der Spieler die braunen Pilze hinter den Bäumen und auf einer Plattform finden muss. Hierbei sind die Pilze in Schatten zu platzieren. Wenn der Spieler einen Pilz findet muss dieser nah genug heran gehen um diesen zu ernten. Beim Erhalt aller Pilze kann die Quest bei dem Quest-NPC vor dem gefallenen Baum abgegeben werden.

Questdesign 3

Die letzte Quest findet in der großen Ruine hinter dem gefallenen Baum statt und entspricht einem Arenakampf. Die Gegner beschleunigen auf den Spieler zu wenn dieser zu nahe heran geht, dabei muss dieser in eine Art Geschicklichkeitsspiel den richtigen Moment abwarten um den Gegner anzugreifen und zu töten. Je nach Gegnertyp, bestimmt durch Farbe und Größe, haben sie verschiedene Geschwindigkeiten, Aufmerksamkeits-Radien und Gesundheit. Währenddessen muss der Spieler aufpassen das er nicht selbst stirbt, im Falle er tut dies wird er mit vollen Lebenspunkten vor das Areal mit den Gegnern zurück teleportiert. Bei Erledigung kann die Quest im Anschluss abgegeben werden und das Spiel ist zu Ende.

¹NPC oder auch Non Player Charakter ist ein von dem Computer gesteuerter Charakter in Computerspielen

3.1.9 Technische Umgebung

Die Erstellung der Applikation und spätere Nutzerstudie wird auf einem Windows Rechner erstellt. Da somit gewährleistet ist, dass die Nutzerstudie und das Test-szenario auch an den Hochschulrechnern der *Hochschule Mittweida* ablaufen kann. Daneben ist die Entscheidung, die Entwicklungsumgebung Unity zu verwenden, aufgrund der Benutzung dieser im Studium zurückzuführen.

3.1.10 Systementwurf

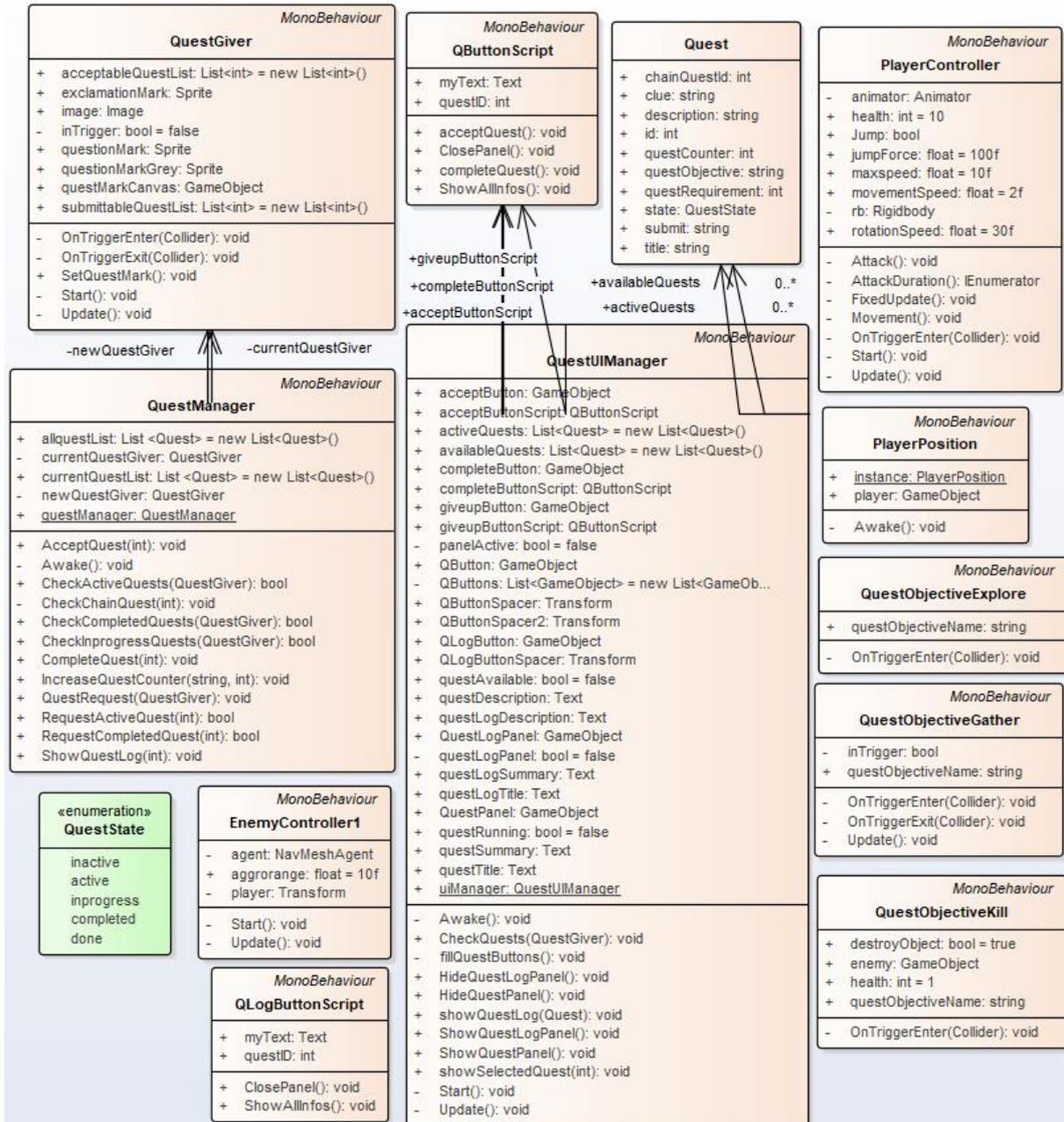


Abbildung 3.1: Der Systementwurf anhand der Klassendiagrammen

3.2 Konzeption der Nutzerstudie

Das Hauptforschungs-Medium dieser empirischen Arbeit ist eine quantitative Nutzerstudie, bei der mit einer Anzahl von Testpersonen deduktiv eine Theorie oder Fragestellung, anhand der zuvor konzipiert und erstellten Software Anwendung, untersucht wird. Dabei wird die allgemeine Aussagekraft aufgrund der Anzahl der Versuchspersonen und der *Objektivität*², *Reliabilität*³ sowie *Validität*⁴ der Umfrage gemessen. [VR16, Kap. 7, S. 138]

In diesem Fall wird die subjektive Motivation der Probanden gegenüber des erstellten Testszenarios. eines auf Minimalismus fokuzierten Rollenspiels, erfasst. Hierzu ist die Nutzerstudie in mehrere Abschnitte mittels Fragen mit Ein- und Mehrfachantworten unterteilt.

Im ersten Abschnitt sind zunächst anonym die Metadaten der Probanden zu erfassen. Beginnend mit den generellen Angaben wie Alter und Geschlecht, um Tendenz und Präferenz verschiedener Altersgruppen analysieren zu können. Sowie mögliche Korrelationen auf simple Mechaniken zwischen jungen und älteren Spielern, welche mit durch die Hardware limitierte Spiele aufgewachsen sind. Daneben ist der wöchentliche Konsum von Videospiele und seit wann dieser stattgefunden hat anzugeben. Dieser zeigt an wie viel Erfahrung mit der Thematik gesammelt wurde und ob bei Langzeit-Konsumenten im Vergleich Unterschiede auftreten. Anschließend sind die damit verbundenen bevorzugten Spielgenre zu deklarieren, um mögliche Rückschlüsse, unter anderem eines Rollenspielers auf die Test-Applikation, festzuhalten. In den letzten beiden Fragen der Metadaten gibt der Proband an, auf welche zwei Aspekte eines Spiels er am meisten wert legt. Also entweder auf das System mit der Funktionalität und Spielart (Genre), der Narrative mit einer gut geschriebenen Geschichte und deren Lenkung im Spiel oder der Ästhetik mit einem beeindruckenden Audio- oder Grafikstil.

Im zweiten Abschnitt wird der generelle Eindruck des Testszenarios sowie der einzelnen Quests und deren vorgestellte Mechaniken auf ihren Motivationswert abgefragt, daher wie viel Spaß die Interaktion erzeugt hat. In einem Punktesystem von eins bis

²Objektivität im Sinne von neutraler und uneingenommener Bewertung der Ergebnissen

³Reliabilität im Sinne von Zuverlässigkeit der Applikation daher jeder Proband erfährt das gleiche Szenario, das erfahrene Erlebnis ist hierbei subjektiv

⁴Validität im Sinne von einem Aussagekräftigen Ergebnis bei dem das zu Messende mit der Methodik auch gemessen wird

vier, bei dem eins für gefällt nicht und vier für gefällt steht, werden alle Elemente des reduzierten Rollenspiels bewertet. Diese sind wiederum in drei Abschnitte eingeteilt.

- **Mechanik:** Hier soll die generelle Funktionalität des Testszenarios bewertet werden.
 - Die Bewegung und generelle Interaktions-Funktion mit der Umwelt.
 - Die Interaktion mit dem Questsystem und dessen User Interface.
 - Die Interaktion mit den Gegner und dessen Verhalten
- **Ästhetik:** Hier soll der Visuelle-Stil und die Gestaltung bewertet werden.
 - Der Low-Poly Stil bei dem im Spiel alle 3D-Elemente aus geometrischen Grundkörpern bestehen.
 - Die einfache auf Minimalismus basierte Farbwahl der Umwelt (Einfarbigkeit).
 - Das Leveldesign mit seiner generellen Gestaltung.
- **Narrative:** Hier soll die Narrative der Quest bewertet werden.

Der letzte Abschnitt dieser Nutzerstudie bezieht sich auf die Untersuchung der Unterschiede von kommerziellen Spielen, beziehungsweise wie die vorherrschenden Elemente in Bezug auf komplexere Systeme abschneiden. Hier sind alle nicht implementierten Elemente aufzuzählen die durch den Minimalismus-Aspekt des Testszenarios nicht verwendet worden sind. Um somit einen möglichen Defizit zwischen einer Applikation mit reduzierten Mechaniken und einem vollwertigen Spiel aufzuzeigen.

Zusammenfassend kann gesagt werden, dass die angefertigte Nutzerstudie die Motivation der Probanden hinsichtlich des Minimalismus-Stiles analysiert sowie alle implementierten und nicht verwendeten Aspekte bewertet. Anhand dieser soll die Schlussfolgerung der allgemeinen Meinung hinsichtlich der Reduktion von Rollenspielen erzielt werden.

Zielstellung des Autors dieser Arbeit ist es mittels dieser Nutzerstudie festzustellen, ob auch andere Spieler das über-implementieren von Mechaniken und unnötiger Grafik-Elemente negativ sehen.

4 Implementierung

In diesem Kapitel wird auf die Umsetzung und Funktionen aller für die empirische Studie benötigten Elementen eingegangen. Beginnend mit der Programmierung mit den wichtigsten Klassen und deren Funktion. Gefolgt von der Erstellung des Levels und der Objekte sowie die Herangehensweise des Autors. Im letzten Schritt der Implementierung wird auf die zuvor erstellte Konzeption und Applikation die Nutzerstudie anhand dem Google-Tool *Google-Formular* erstellt.

4.1 Programmierung

Im Rahmen der Erstellung der Applikation ist das Questsystem und die damit verknüpften Skripte als Hauptelement anzusehen. Diese beruhen wie in der Konzeption beschrieben auf ein zuvor bearbeitetes Projekt. Hierbei sei gesagt, da es bei der Entwicklung zu Schwierigkeiten bei der Implementierung des User Interfaces gekommen ist. Daher wurde anhand eines Tutorials¹, basierend auf dem gleichen System, die User-Interface-Elemente übernommen.

Das Questsystem selbst besteht aus neun Klassen, bei dem die drei Klassen QuestManager, QuestUIManager und QuestGiver die Lenkung des Systems übernehmen. Die Quest an sich wird durch eine serialisierbare Klasse des selben Namens verkörpert, welche nur aus Variablen besteht. Die wichtigsten Variablen für die Schaltung des Questsystems sind hierbei:

- **enum QuestState:** Gibt die Zustände an, die eine Quest annehmen kann.
- **QuestState:** Deklariert den derzeitigen Zustand der Quest und wie mit dieser in Funktionen und Schleifen umgegangen wird, standardmäßig inactive.
- **int id:** Dient als Kennmarke im System, um diese zu lenken.

¹<https://www.youtube.com/watch?v=a8y6Ul-nX9o>

- **int chainQuestId:** Deklariert die nächste Quest, welche durch die Erledigung der gerade verfolgten, freigeschaltet wird.
- **string questObjective:** Dient als Kennmarke für das Questziel, daher nur das erledigen der Aufgabe / Handlung mit übereinstimmenden Inhalt zählt zur Erledigung.
- **int questCounter:** Gibt an wie viele in der Quest angegebene Handlungen bewältigt worden sind.
- **int questRequirement:** Deklariert wie viele in der Quest angegebene Handlungen bewältigt werden müssen.

Die Hauptklasse des Systems ist der QuestManager, der mittels einer Prefab² in jeder Szene³ vorhanden sein muss. In dieser wird mittels zwei Quest-Listen, eine mit allen Quests und eine mit der derzeitig von Spieler verfolgten, das System anhand von Bool- und Schleifen-Funktionen gesteuert. Im Inspektor kann hierbei, (siehe Abbildung 4.1) durch die Erhöhung der allquestList<> Listenlänge neue Quests hinzugefügt werden. Daneben sind die anderen beiden Klassen QuestUIManager, welche auch in jeder Szene per Prefab vorkommen muss, sowie QuestGiver für die Kommunikation zwischen Spieler und der QuestManager-Klasse zuständig. Hierbei hat der QuestUIManager zwei Quest-Listen und der QuestGiver zwei integer-Listen für Quest-ID's, wobei erstere für aktive daher annehmbare Quests verantwortlich sind und letztere für angenommene oder abgebbare. Dadurch kann bei einem Objekt mit der QuestGiver-Klasse in der Szene entschieden werden wo Quests angenommen oder abgegeben werden können.

²Eine Prefab ist eine zuvor zusammengestellte Vorlage, kann aus verschiedenen Elementen bestehen

³Eine Szene, in der Spieleentwicklung, ist ein Level mit all seinen Elementen (Terrain, Belichtung, Spieler, Objekte etc.)

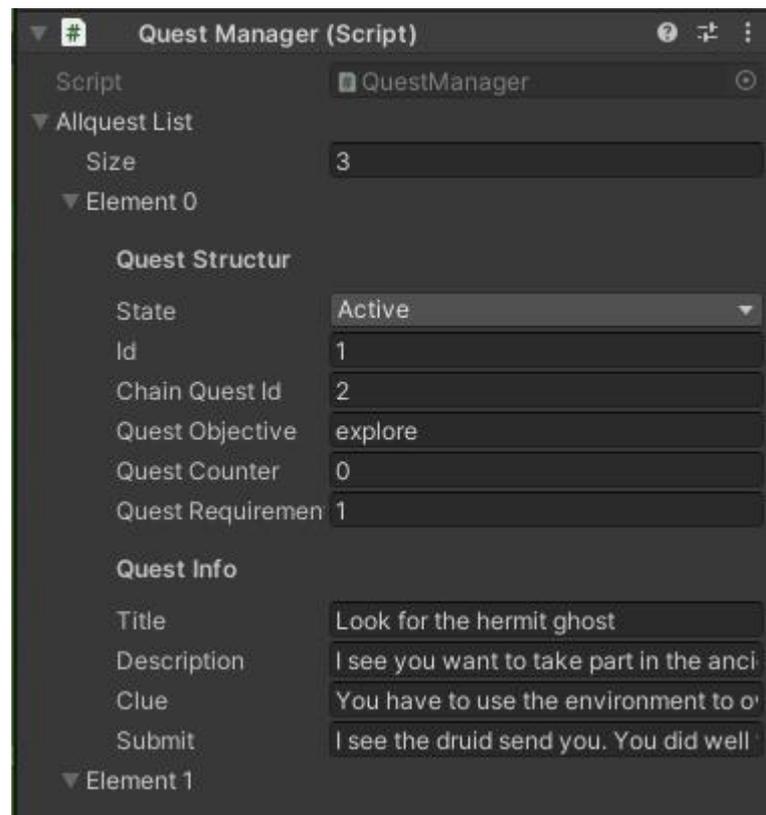


Abbildung 4.1: Ausschnitt des QuestManagers im Inspector-Fenster von Unity

Hierzu die Funktion des Questsystems anhand einer Quest im Spiel.

Bei Start der Szene wird bei allen QuestGiver-Objekten die Funktion SetQuestMark() gestartet (siehe Listing 4.1).

Listing 4.1: Ausschnitt der QuestGiver-Klasse mit der SetQuestMark()-Funktion

```

1 public void SetQuestMark()
2 {
3     if (QuestManager.questManager.CheckActiveQuests(this))
4     {
5         questMarkCanvas.SetActive(true);
6         image.sprite = exclamationMark;
7     }
8     else if (QuestManager.questManager.
9     CheckInProgressQuests(this))

```

```
10     {
11         questMarkCanvas.SetActive(true);
12         image.sprite = questionMarkGrey;
13     }
14     else if (QuestManager.questManager.
15     CheckCompletedQuests(this))
16     {
17         questMarkCanvas.SetActive(true);
18         image.sprite = questionMark;
19     }
20     else
21     {
22         questMarkCanvas.SetActive(false);
23     }
24 }
```

Dabei wird anhand der angegebenen Quest-ID's in der `acceptableQuestList<>` des jeweiligen QuestGiver-Objekts, im QuestManager drei Bool-Funktionen ausgeführt. Wenn eine der referenzierten Quests den Zustand *active*, *inprogress* oder *completed* hat, wird anschließend dementsprechend ein Rollenspiel typisches Questsymbol über einen lokalen Canvas⁴ angezeigt.

Im Falle der Spieler interagiert mit dem QuestGiver-Objekt in seinem Collider-Bereich und dieser besitzt eine Quest, wird eine Reihe von Funktionen im QuestUIManager ausgelöst. Beginnend mit `CheckQuests()` bei dem die Quest-ID an die `QuestRequest()`-Funktion im QuestManager weitergeleitet wird. In dieser wird bei einer Quest im *active* Zustand im QuestUIManager die `questAvailable` Bool auf wahr gesetzt und daraufhin die Quest in die `availableQuests<>` Liste aufgenommen. Anschließend wird die Funktion `ShowQuestPanel()` aktiviert, in dieser wird ein zuvor zusammengestellter Prefab-Canvas geöffnet und mit Buttons der `fillQuestButtons()`-Funktion gefüllt. Diese referenziert die anzunehmende Quest. Nun muss der Spieler per linken Mausklick den Button auf der linken Seite des User Interface anklicken um die Quest anzuzeigen. Dadurch wird durch den Button der `QButtonScript`-Klasse die Funktion `ShowAllInfos()` ausgeführt. Diese zeigt durch die `showSelectedQuest()`-Funktion des QuestUIManagers alle Informationen der Quest an. Daraufhin wird

⁴Ein Canvas ist das Mutter-Element von allen User-Interface-Elementen

bei *active* Zustand der Quest ein Akzeptieren-Button im User Interface erzeugt. Bei dem der Spieler die Quest annehmen kann. Bei Annahme wird in der QButtonScript-Klasse die Funktion `acceptQuest()` im QuestManager aktiviert. Diese kopiert die Quest in die `currentQuestList<>` und setzt deren Zustand auf *inprogress*. Daneben wird die QuestUIManager Funktion `HideQuestPanel()` ausgeführt, welche alle Informationen und Buttons vom Canvas des User Interfaces entfernt. Als letzter Schritt werden alle QuestGiver-Objekte mit `SetQuestMark()` in der Szene aktualisiert.

Im Anschluss kann nun der Spieler die in der Quest geforderte Handlung ausführen. Die je nach Art der Quest in diesem Projekt, wird eine von drei verschiedene Skripte verwendet. In diesem Fall soll die erste Quest-Art Verwendung finden mit der QuestObjectiveExplore-Klasse, bei dem der Spieler bei Eintritt in einen meshlosen Collider⁵ die Funktion `IncreaseQuestCounter()` des QuestManagers auslöst. Dieser erhöht den Integer-Wert der `questCounter`-Variable der Quest und setzt den Zustand anschließend auf *completed*, wenn das Questziel erfüllt wurde. Nach dieser Funktion wird abermals alle QuestGiver-Objekte durch `SetQuestMark()` in der Szene aktualisiert (siehe Listing 4.2).

Listing 4.2: Ausschnitt der QuestManager-Klasse mit der `IncreaseQuestCounter()`-Funktion

```

1 public void IncreaseQuestCounter
2 (string questObjective ,int itemAmount)
3 {
4 for (int i = 0; i < currentQuestList.Count; i++)
5     {
6         if (currentQuestList[i].questObjective ==
7             questObjective && currentQuestList[i].state ==
8             Quest.QuestState.inprogress)
9             {
10            currentQuestList[i].questCounter += itemAmount;
11            }
12            if (currentQuestList[i].questCounter >=
13                currentQuestList[i].questRequirement &&
14                currentQuestList[i].state ==
15                Quest.QuestState.inprogress)
16            {

```

⁵Daher nicht sichtbar

```
17         currentQuestList [ i ]. state =  
18         Quest . QuestState . completed ;  
19     }  
20 }  
21 }
```

Als letzter Schritt kann nun zu dem QuestGiver-Objekt, bei dem man die Quest abgeben kann mit dem gelben Fragezeichen über den Kopf, herangetreten und interagiert werden. Ähnlich wie im Fall der Quest Annahme wird CheckQuests() ausgeführt und anschließend QuestRequest(). Hier wird jedoch die Quest in die activeQuests<> Liste des QuestUIManagers hinzugefügt und die Bool questRunning auf Wahr gesetzt. Daraufhin wird wieder die Funktionen ShowQuestPanel() und FillQuestButtons() ausgeführt sowie die anschließende Interaktion des Spielers mit dem User Interface benötigt. Infolge wird jedoch durch den *completed* Zustand der Quest mit der Funktion ShowAllinfos() ein Abgeben Button generiert. Dieser löst bei Abgabe der Quest die completeQuest()-Funktion der QButtonScript Klasse aus. Welcher im QuestManager die CompleteQuest()-Funktion aktiviert, die die Quest aus der currentQuestlist<> entfernt und die CheckChainQuest()-Funktion auslöst. Diese schaltet wenn angegeben eine neue Quest vom Zustand *inactive* zu *active*, die darauf vom Spieler angenommen werden kann. Zuletzt wird der Canvas mit HideQuestPanel() von allen Infos und Buttons befreit und die QuestGiver-Objekte durch SetQuestMark() in der Szene aktualisiert.

Steuerung

Die generelle Steuerung über die PlayerController-Klasse wurde mithilfe eines auf Kräfte basierten Bewegung erzielt. Anhand einer lokalen Krafteinwirkung mit *AddForce transform.forward* und *AddForce transform.right* bei den die jeweils gegenüberliegende Seite mit einem negativen Wert angesteuert wird. Die Springen Funktion greift hier jedoch auf die Vector3-Variable des Worldspaces zurück um den Spieler aufhüpfen zu lassen. Daneben hat dieser eine Gesundheitsvariable für die Kampfquest, welche bei Berührung mit den Gegner (selbst mit dem Schwert), den Spieler schadet und ihn bei tot zum Anfang des Areal mit vollen Leben teleportiert. Dabei wird der Angriff gegen den Gegner durch eine Animation simuliert, die auf Tastendruck eine zuvor erstellte Int-Variable auf einen Wert setzt der der

Angriffs-Animation entspricht und anschließend eine Coroutine⁶ startet um diese nach Ausführung wieder zurückzusetzen (siehe Listing 4.3).

Listing 4.3: Ausschnitt der PlayerController-Klasse mit der Attack() Funktion

```

1 void Attack()
2     {
3         StartCoroutine (AttackDuration());
4     }
5     IEnumerator AttackDuration()
6     {
7         animator.SetBool("Attack", true);
8         animator.SetInteger("condition", 1);
9         yield return new WaitForSeconds (0.5f);
10        animator.SetInteger("condition", 0);
11        animator.SetBool("Attack", false);
12    }

```

Bei der Entwicklung dieser Klasse kam es zu Problemen mit der Springen- und Angriffs-Funktion. Diese haben in beiden Fällen zunächst in der Unity-internen FixedUpdate()-Funktion teilweise die Aktivierung versäumt. Daraufhin sind beide in Update() verschoben worden, welche das Problem gelöst hat.

Gegner

Die implementierten Gegner, in diesem Fall Bälle, werden durch einen NavMesh-Agenten gesteuert. Durch die EnemyController-Klasse reagiert hierbei ein Gegner wenn der Spieler ihm zu nahe kommt. Dies wird erreicht mit der Klasse PlayerPosition welche auf das Spieler-Objekt hinzugefügt wird. Bei dem anschließend der Gegner anhand des eingestellten Aufmerksamkeits-Radiuses den Abstand zwischen ihm und den Spieler ausrechnet. Bei Unterschreitung des Wertes wird der Spieler verfolgt bis er aus diesem Radius austritt oder stirbt. Die Questsystem-Komponente mit Gesundheit ist hierbei auf die Klasse QuestObjectiveKill ausgelagert, um somit ein modulares System für die Gegner Erstellung zu besitzen (siehe Listing 4.4).

Listing 4.4: Ausschnitt der QuestObectiveKill-Klasse mit der Gegner-Mechanik des sterbens und updaten des Questsystems

⁶Eine Coroutine unterbricht den Ablauf einer Funktion anhand eines angegeben Zeitraums / Wertes, die Funktion läuft anschließend weiter

```
1 void OnTriggerEnter(Collider other)
2 {
3     if (other.tag == "Weapon")
4     {
5         health = health - 1;
6         if (health <= 0)
7         {
8             QuestManager.questManager.
9             IncreaseQuestCounter(questObjectiveName, 1);
10            QuestGiver [] NPCs = GameObject.
11            FindObjectsOfType(typeof(QuestGiver))
12            as QuestGiver [];
13            foreach (QuestGiver obj in NPCs)
14            {
15                obj.SetQuestMark();
16            }
17            if (destroyObject)
18            {
19                Destroy(enemy);
20            }
21        }
22    }
23 }
```

4.2 Objekt- und Levelerstellung

Nach Erstellung aller Skripte, deren Einpflegung und Testung kann das Level erstellt werden. Insgesamt soll hierbei nur ein Level entstehen. Anhand der Quest und Questmechaniken ist zuerst der generelle Rahmen des Leveldesigns festzustellen. Für die Untersuchung des Reduktions-Aspektes muss das Level nur so groß werden wie die drei zu erstellenden Quest-Areale. Basis hierfür soll eine Terrain-Plane dienen, welche für den Low-Poly-Stil als Fläche belassen wird. Infolge der Kettenquest-Mechanik sind die Quest-Areale Linear hintereinander anzuordnen. Diese sind im

Level als eine Art Lichtung in einem Wald darzustellen, bei dem offene Erkundung unterbunden werden soll. Daneben ist eine steinerne Ruine und Pilze als Kontrast zu den Bäumen zu erstellen. Durch die Sprungfunktion des Spielers und der Plattform-Mechanik der ersten Quest, müssen hohe 3D-Objekte als Levelbeschränkung eingesetzt werden. Um so eine natürliche Grenze zu erzeugen und keine immersionszerstörenden Alphawände⁷. Die Gestaltung der Areale sind anhand dieser Parameter anzulegen:

- **Quest 1:** Mindestens drei Hindernisse die Übersprungen werden müssen, dabei sollen zwei nur durch das Springen auf mehrere Plattformen überwunden werden können.
- **Quest 2:** Mindestens fünf Pilze die eingesammelt werden müssen, diese sollen generell im Schatten platziert werden.
- **Quest 3:** Ein Arena Bereich den die Gegner nicht verlassen können, bei dem mindestens sechs Gegner platziert werden. Dabei sollen unterschiedliche Gegnertypen durch Größe, Farbe, Geschwindigkeit und Lebenspunkte differenziert werden.

Im nächsten Schritt wird das Level in Unity gebaut. Hierzu ist das interne GameObject-Tool zur Erstellung der Plane und der 3D-Objekte zu verwenden. Unter den Punkt 3D-Object (siehe Abbildung 4.2) sind diese in der Szene erstellbar.

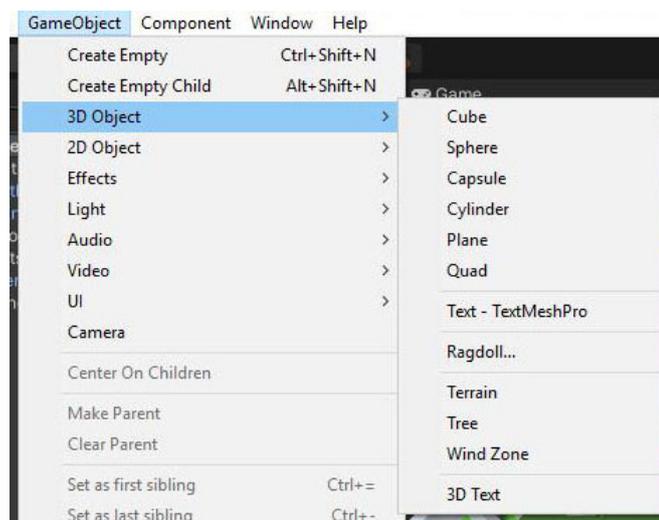


Abbildung 4.2: Screenshot von dem Unity internen GameObject-Tool

⁷Alphawände sind unsichtbare Collider die zur Levelbegrenzung verwendet werden

Dabei sind alle Gestaltungselemente aus Würfeln mit unterschiedlichen Skalierungen zusammengesetzt. Unter anderem der Baum mit einem kleinen braunfarbigen Würfel als Stamm und einem großen grünen als Blattwerk. Mehrere dieser Bäume wurden für den Levelbau hierfür unterschiedlich skaliert, gruppiert und per Kopieren- und Einfügen-Methodik grob ein Levelverlauf erzeugt. Anschließend wird das selbige anhand der Deko- und Quest-Elemente angewandt. Als letzter Schritt der Gestaltung des Levels, werden die einzelnen Elemente unterschiedlich skaliert und rotiert, um einen natürlichen Eindruck zu erzeugen, unter anderem unterschiedliche Baumgrößen. (siehe Abbildung 4.3)

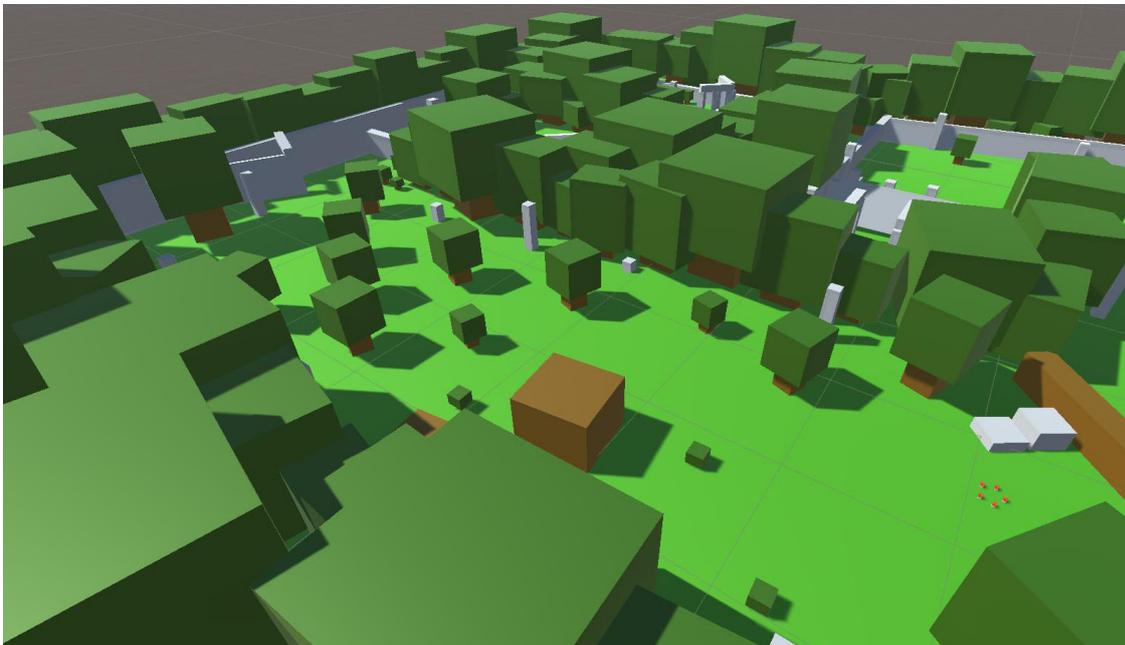


Abbildung 4.3: Screenshot des erstellten Levels

4.3 Nutzerstudie

Die Umsetzung der Nutzerstudie dieser Arbeit wurde mittels dem Google-Tool *Google-Formular* erzeugt. Bei dem man als Besitzer eines Google Accounts schnell Umfragen oder Formulare erzeugen und für Personen bereitstellen kann. Diese Applikation wurde gewählt weil sie ein übersichtliches und intuitives System verwendet, bei dem man leicht Fragen kopieren, in Abschnitte einteilen oder Medien wie Bilder

und Videos hinzufügen kann. Daneben war es wichtig, dass alle Fragen als Pflichtfragen deklariert werden können, um somit ein auswertbares Ergebnis zu erzielen. Auf die angegebenen Daten kann anschließend direkt zugegriffen werden und diese sind automatisch von Google als Diagramm angezeigt. Die Nutzerstudie ist wie in der Konzeption beschrieben in die drei Abschnitte Allgemeine Angaben, Bewertung der Applikation und was hat den Proband an der Applikation gegenüber eines normalen Rollenspiels gestört, eingeteilt worden. Bei denen zwei verschiedene Typen von Fragen mit dementsprechender Formatierung vorkommen. Die erste Art von Fragen, die nur im ersten Abschnitt für die Metadaten auftreten, sind auf Multiple-Choice basis, daher eine Antwort aus mehreren Möglichkeiten. Die zweite basiert auf einem Multiple-Choice-Raster, in der eine Vielzahl an Multiple-Choice-Fragen untereinander stehen und der Proband seine Präferenz oder Meinung gegenüber verschiedener Aussagen und Angaben äußert.

5 Evaluation

Nach der Konzeption und Implementation der Forschungsmedien ist in diesem Schritt die erhobenen Daten der Nutzerstudie zu analysieren. Der Teilnahmezeitraum dieser Untersuchung betrug eine Woche vom 13.11.2020 bis einschließlich 20.11.2020, bei dem sechs Probanden teilgenommen haben.

5.1 Ablauf

Der Ablauf dieser Studie wurde anhand der derzeitigen Corona Situation komplett online vorgenommen. Die Probanden haben einen Google-Drive Download-Link erhalten, bei dem sie eine zip-Datei namens Spieletest herunterladen und diesen entpacken mussten. In dieser enthalten war eine Hyperlink-Verknüpfung zur Umfrage, eine Text Datei mit der Aufschrift *Bitte Lesen!* sowie ein Ordner namens *Applikation*, der das Testszenario enthalten hat. Anschließend konnten die Probanden anhand der Textdatei den generellen Ablauf sowie die Steuerung für das Spiel entnehmen. Diese haben im Anschluss die Applikation¹ und daraufhin die Nutzerstudie absolviert.

5.2 Auswertung

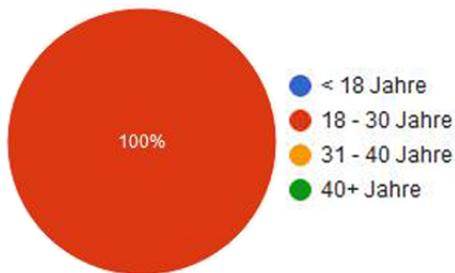
Durch die Auswertung der Nutzerstudie soll festgestellt werden inwiefern die Motivation der Probanden auf den Reduktions- beziehungsweise Minimalismus-Gedanken gegenüber Rollenspielen beeinflusst wird.

Der erste Abschnitt Namens *Allgemeine Angaben* befasste sich mit den Metadaten der Testpersonen (siehe Abbildung 5.1). Bei dieser wurde ersichtlich das alle Teilnehmer zwischen 18 und 30 Jahre alt sind, mit einem Geschlechtsverhältnis von

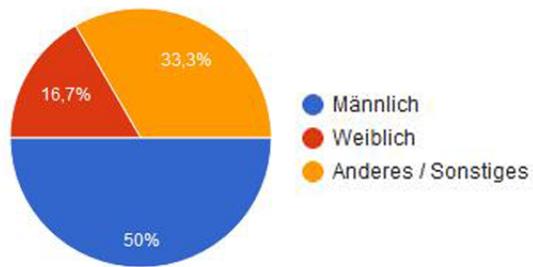
¹Entspricht dem im Unterkapitel 3.1.8 Quest- und Leveldesign konzipierten Quest-ablauf.

50% männlich, 33,3% anderes / sonstiges und 16,7% weiblich. Bei den 50% 8 - 14 Stunden, 33,3% 28 Stunden oder mehr sowie 16,7% nur 1 - 7 Stunden Videospiele in der Woche konsumieren. Hierbei nutzen alle Befragten bis auf einer, mit 1 - 5 Jahren, seit mindestens 11 Jahren dieses Unterhaltungs-Medium. Ihre bevorzugten Spiele-Genres sind hierbei Rollenspiele, Adventure, Action und Indie Spiele (siehe Tabelle 5.1). Für diese Studie passend ist das Rollenspiel-Genre am beliebtesten, es wird von allen Teilnehmern gespielt und von 83,3% mindestens oft konsumiert. Somit ist die Validität des Ergebnisses der Nutzerstudie gegeben. Daneben ist das Adventure- und Action-Genre teil von modernen Hybrid-Rollenspielen. [Ada14, S. 12 f.]. Die beiden letzten Fragen dieses Abschnittes beziehen sich auf die Präferenzen der Probanden hinsichtlich der Elemente im Spiel. Für 66,7% ist das System mit deren Mechaniken und Spielart (Genre) der wichtigste Aspekt, die restlichen 33,3% bevorzugen die Narrative mit der erzählten Geschichte und deren Lenkung. Das zweitwichtigste Elemente dieser Testgruppe ist mit 66,7% die Ästhetik mit dem Visuellen Stil und Sound, gefolgt von jeweils einer Stimme für das System und der Narrative. Daraus kann man schließen das bei dem Großteil der Testgruppe anhand des Genres gefolgt von der Ästhetik ihr Kauf- und Konsumverhalten entschieden wird. Bei dem eher auf die Funktionalität mit den verbundenen Mechaniken geachtet wird, wenn es um den Konsum eines Computer-Videospieles geht.

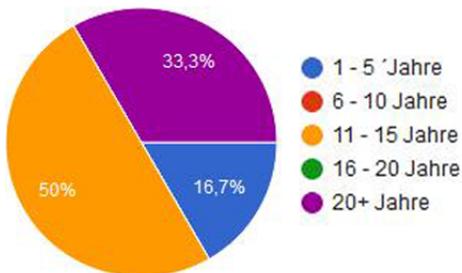
Bitte geben Sie ihre Altersgruppe an.



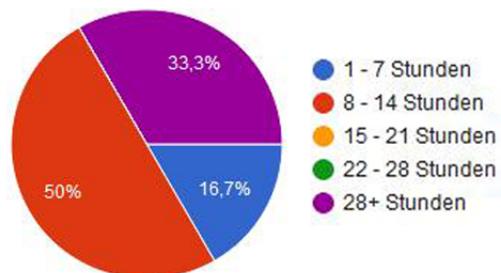
Bitte geben Sie ihr Geschlecht an.



Seit wie viel Jahren spielen Sie Videospiele?



Wie viele Stunden pro Woche spielen Sie Videospiele?



Was ist der wichtigste Aspekt in Spielen für Sie? Was ist der zweitwichtigste Aspekt in Spielen für Sie?
(Bitte wählen Sie eine andere Antwort als in 1.6)

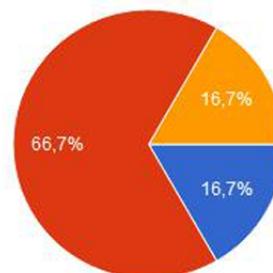
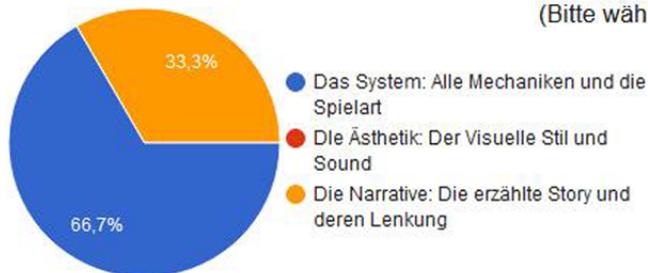


Abbildung 5.1: Diagramme zu den allgemeinen Angaben der Probanden

Spiele ich....	Nicht	Selten	Gelegentlich	Oft	Häufig
Action		1	1	3	1
Adventure		1	1	1	3
Aufbau- und Strategie	1	1	2	1	1
First-Person-Shooter		4	1	1	
Gelegenheitsspiele	1	1	2	1	1
Hack and Slash	2	3	1		
Horror	4	1	1		
Indie		2		2	2
MMORPG	2		2	1	1
MOBA	3			2	1
Multiplayer	1	1	1	1	2
Platformer (Jump'n'run)	2	1	3		
Rennspiele	4	1	1		
Rollenspiele			1	2	3
Simulation	1	4	1		
Sport	6				
Stealth	2	2		2	

Tabelle 5.1: Spiele-Genre die die Probanden der Nutzerstudie nach eigenen Angaben konsumieren

Im zweiten Abschnitt wurde die Motivation und der Eindruck gegenüber der erstellten Applikation abgefragt (siehe Tabelle 5.2). Bei dem eine Pattsituation zwischen den Probanden entstanden ist, 50% sind positiv und 50% negativ gegenüber des gesamten Testszenarios und der bei Bewältigung entstandenen Motivation eingestellt. Daher kann generell keine eindeutige Aussage über den positiven oder negativen Einfluss von Reduktion der Mechaniken eines Rollenspieles auf die Motivation geschlossen werden. Somit sollte die Frage offen sein ob Reduktion oder Minimalismus, ähnlich wie in der Kunstszene als Stil, in der Spielebranche als Genre deklariert werden sollte. Die Bewertung der einzelnen Komponenten des Spiels sind bei Steuerung, Ästhetik (Stil und Farbwahl) und dem Questsystem eher negativ ausgefallen. Bei der Narrative neutral und dem Leveldesign jedoch positiv bei dem 66,7% der Probanden diese mindestens eher gefallen hat. Die Meinung zu den implementierten Quests ist hierbei zum Großteil positiv ausgefallen, bei dem die Platformer(1)- und

Sammelquest(2) positiv und die Kampfquest(3) negativ bewertet wurde. Hierbei sticht die Sammelquest heraus bei dem 3 von 6 Personen das erledigen dieser gefallen und motiviert hat sowie einer bei dem dies eher zutrifft. Dabei kann man ableiten, dass die positive Bewertung des Leveldesigns anhand der Sammelquest beeinflusst wurde. Bei der Auswertung dieses Abschnittes ist aufgefallen das bei 6 von 10 Fragen immer 2 Personen den Aussagen nicht zugestimmt haben, somit kann angenommen werden dass diese negativ gegenüber dem Minimalismus-Design eingestellt sind.

Ich stimme	Nicht zu	Eher nicht zu	Eher zu	Zu
Das spielen des Spiels hat mir gefallen und mich motiviert.	2	1	1	2
Die Bewegungssteuerung hat mir gefallen.	3	2	1	
Der Low-Poly Grafikstil hat mir gefallen.	3	1	1	1
Die Einfarbigkeit hat mir gefallen.	3	1	1	1
Die Narrative des Spiels hat mir gefallen.	2	1	3	
Das Leveldesign hat mir gefallen.	2		2	2
Das Questsystem und die Interaktion mit dieser hat mir gefallen.	2	3	1	
Das bewältigen der ersten Quest mit deren Mechaniken hat mir gefallen und mich motiviert. (Jump 'n' run)	2		2	2
Das bewältigen der zweiten Quest mit deren Mechaniken hat mir gefallen und mich motiviert. (Sammelquest)	2		1	3
Das bewältigen der dritten Quest mit deren Mechaniken hat mir gefallen und mich motiviert. (Kampf)	4		1	1

Tabelle 5.2: Bewertung der Applikation anhand der Probanden der Nutzerstudie

Im letzten Abschnitt *Was hat Sie gestört?* wurde die Meinung über alle nicht implementierte Funktionen eines typischen Rollenspiels abgefragt (siehe Tabelle 5.3). Ähnlich wie in Abschnitt 2 ist auf die Aussage *Mir war das Spiel zu simpel* eine Pattsituation entstanden, welche eher auf zu simpel tendiert. Den Probanden hat

generell in diesem Testszenario, geordnet an der Priorität, das Feedback (Audio und Grafik) zu 83,3%, der Sound und die Musik zu 66,7% und das User Interface zu 50% gefehlt. Somit ist anzunehmen das die Testpersonen mit dem Informations-Defizit ein Problem haben, um dies zu bestätigen müsste jedoch diese fehlenden Elemente implementiert und eine neue Studie angefertigt werden. Daneben wird nach Meinung der Probanden das Attributsystem, die Cutscenes und Charakterstellung zu 100% in dieser Applikation nicht benötigt. Alle anderen aufgezählten Elemente tendieren ebenfalls auf, wird im System nicht benötigt, mit dem Sonderfall *Mir hat ein Skill-system gefehlt* bei dem 5 von 6 Personen nicht zustimmen und eine zustimmt. An diesem Abschnitt kann angenommen werden das reguläre Rollenspiel-Mechaniken in diesem Szenario nicht gefehlt haben.

Ausgehend am Ergebnis des Testszenarios und die damit verknüpften Nutzerstudie lässt sich ableiten, dass kein eindeutiger Einfluss auf die Motivation durch die Reduzierung von Kernmechaniken eines Rollenspiels, durch diese Applikation erzielt werden konnte.

Ich stimme	Nicht zu	Eher nicht zu	Eher zu	Zu
Das Spiel war zu simpel.	1	2	1	2
Der Sound und die Musik hat gefehlt.	1	1	3	1
Das Feedback hat gefehlt. (Audio und Grafik)	1		4	1
Das User Interface hat gefehlt. (Gesundheitspunkte etc.)		3	2	1
Ein Attributsystem hat gefehlt.	6			
Ein Skillssystem hat gefehlt.	5			1
Ein Inventarsystem hat gefehlt.	4	2		
Eine andere Waffe oder Magie hat gefehlt.	3	1	1	1
Die Belohnungen der Questgeber haben gefehlt.	3	1	1	1
Ein Haupt- und Ingame-Menü hat gefehlt.	4	1	1	
Ein Speichersystem hat gefehlt.	2	2	1	1
Cutscenes haben gefehlt.	6			
Zufalls-Elemente haben gefehlt.	4	2		
Eine Charakter-Erstellung hat gefehlt.	6			

Tabelle 5.3: Bewertung der nicht implementierten Elemente eines Rollenspiels anhand der Probanden der Nutzerstudie

6 Zusammenfassung und Ausblick

Ziel dieser Arbeit war die Untersuchung der Wirkung reduzierter Kernmechaniken eines Rollenspiels auf die Motivation des Spielers, die anhand einer Applikation und anschließender Nutzerstudie evaluiert wurde.

Die Untersuchung der Umfrage hat ergeben, dass sich anhand der erstellten Applikation kein eindeutiger Einfluss auf die Motivation durch die Reduzierung von Kernmechaniken eines Rollenspiels erzielt werden konnte. Dieses Ergebnis kommt für den Autor überraschend, der zu viele Mechaniken als negativer Aspekt sieht. Hier kann, wie in der Auswertung des zweiten Abschnitts angemerkt, eine Verzerrung des Ergebnisses anhand von zwei Personen aufgetreten sein. Daher müsste an diesem Punkt eine neue Studie, mit dem von den Probanden angemerkten fehlenden Elementen, angesetzt werden. Um somit ein aussagekräftiges Ergebnis zu erhalten. Hierzu wären Audio- und User-Interface-Elemente bei den Ausbesserungen im Fokus zu stellen. Unter anderem eine neue Bool-Variable in der serialisierten Questklasse, mit einer modifizierten `completeQuest()` Funktion im `QuestManager`, um nach einer ausgewählten Quest eine Gewinn-Funktion zu triggern. Daneben sind allgemeine Interface Elemente für den Gegner und Spieler zu implementieren. Dabei soll durch einen lokalen Canvas, wie bei den `QuestGiver`-Objekten, über dem jeweiligen Gameobjekt ein roter Balken für das Status-Attribut Gesundheit verwendet werden. Als letzter Schritt sind Audio-Elemente an jede Funktion und Interaktion zwischen Spieler und Spielwelt hinzuzufügen. Zum Beispiel eine generelle Hintergrundmusik, die Kollision zwischen Spieler und Gegner oder ein markantes gewonnen / verloren Jingle¹.

Neben einer Erweiterung hinsichtlich einer neu erstellten Nutzerstudie, kann diese Applikation als Entwicklung eines modularen Rollenspiel-Systems dienen. Im jetzigen Zustand ist die Modularität auf natürliche Weise bei der Kreierung der Software entstanden. Bei der Weiterentwicklung müssten sich hierzu die allgemeine

¹Ein Jingle ist eine kurze einprägsame Erkennungsmelodie

Konzeption hinsichtlich der Art der Module überlegt werden. Entweder, wie in der jetzigen Applikation bei den Questobjekten(Questobjective-/Kill/Gather/Explore) im Level, sind einzelne Klassen mit limitierten Funktionen auf die Objekte hinzuzufügen. Dabei wird für eine Mechanik im Spiel mehrere kleine variierende Klassen erstellt, um unterschiedliche Interaktions-Möglichkeiten bei gleich bleibender Mechanik zu ermöglichen. Oder es wird für jede Mechanik eine Sammel-Klasse mit allen Funktionen erstellt, bei der durch Bool-Variablen die Funktionen, in Unity durch das Inspektor-Fenster, ausgewählt werden können. Beide Varianten haben Vor- und Nachteile die dabei zu berücksichtigen sind.

Zusammenfassend kann gesagt werden, dass durch das Ergebnis dieser empirischen Arbeit eine bessere Grundlage für die Erforschung des Minimalismus-Stiles, anhand eines Computer-Rollenspiels, erfolgen kann. Da alle Entwicklungsfehler, bezüglich der zu starken Reduzierung der Mechaniken, bei zukünftigen Studien vermieden werden können.

Literaturverzeichnis

- [Aar04] E. J. Aarseth: *Quest Games as Post-Narrative Discourse, Narrative across Media : The Languages of Storytelling*, S. 361–376, 2004, isbn: 978-0-8032-8993-2.
- [Ada14] E. Adams: *Fundamentals of Role-Playing Game Design*, Pearson Education, Inc., 2014, ISBN 978-0-133-81192-6.
- [Bev20] *Bevölkerungszahl im 1. Halbjahr 2020 leicht zurückgegangen*, Okt. 2020, URL:https://www.destatis.de/DE/Presse/Pressemitteilungen/2020/10/PD20_404_12411.html, besucht am 01.12.2020.
- [Bon15] S. Bone: *Minimalism in Game Design: Examples, Tips, and Ideas*, Nov. 2015, URL:<https://gamedevelopment.tutsplus.com/articles/minimalism-in-game-design-examples-tips-and-ideas--cms-23446>, besucht am 01.12.2020.
- [Bow10] S. L. Bowman: *The Functions of Role-Playing Games*, McFarland Company, inc., 2010, ISBN 978-0-7864-4710-7.
- [DCM53] R. A. Clark E. L. Lowell D. C. McClelland, J. W. Atkinson: *The achievement motive*, New York, Appleton-Century-Crofts, 1 Aufl., 1953, (OCoLC)609111964.
- [DS14] P. Pintrich D. Schunk, J. Meece: *Motivation in Education Theory, Research and Applications*, Pearson Education Limited, 4 Aufl., 2014, ISBN 978-1-292-04147-6.
- [Fif20] *Illusion Fortschritt: Wie FIFA uns jedes Jahr für dumm verkauft*, Nov. 2020, URL:<https://www.pcgames.de/FIFA-21-Spiel-72823/Specials/FIFA-21-EA-Sports-Fussball-Simulation-Gameplay...-Features-Meinung-1361152/>, besucht am 01.12.2020.

- [How08] J. Howard: *Quests Design, Theory, and History in Games and Narratives*, A K Peters/CRC Press, 2008, ISBN 978-1-56881-347-9.
- [Jes05] J. Jesper: *Half-Real: Video Games between Real Rules and Fictional Worlds*, The MIT Press, 2005, ISBN 978-0-262-10110-3, ePUB.
- [JFM12] R. Nicodemus J. F. Millburn: *Minimalism: Live a Meaningful Life*, Asymmetrical Press, 2012, ISBN 978-1936539673, ePUB.
- [JH18] H. Heckhausen J. Heckhausen: *Motivation und Handeln*, Springer, 5 Aufl., 2018, ISBN 978-3-662-53926-2.
- [Kra93] A. Krapp: *Die Psychologie der Lernmotivation. Perspektiven der Forschung und Probleme ihrer pädagogischen Rezeption*, *Zeitschrift für Pädagogik*, Bd. 39(2):S. 187–206, 1993, uRN: urn:nbn:de:0111-pedocs-111711.
- [MaM17] *Die Maslowsche Bedürfnispyramide Motivation und Bedürfnisse des Menschen*, 2017, URL: https://www.lpb-bw.de/fileadmin/Abteilung_III/jugend/pdf/ws_beteiligung_dings/2017/ws6_17/maslowsche_beduernispyramide.pdf, besucht am 01.12.2020.
- [Mar14] Da. Marzona: *Minimal Art*, TASCHEN, 2014, ISBN 9783822830604.
- [Mas43] A. H. Maslow: *A Theory of Human Motivation*, *Psychological Review*, Bd. 50:S. 370–396, 1943.
- [Per14] J. Perl: *Art in America 1945-1970: WRITINGS FROM THE AGE OF ABSTRACT EXPRESSIONISM, POP ART, AND MINIMALISM*, Library of America, 2014, ISBN 978-1-59853-367-5, ePUB.
- [SB11] A. Akhshani A. Samsudin S. Behnia, A. Akhavan: *A novel dynamic model of pseudo random number generator*, *Journal of Computational and Applied Mathematics*, Bd. 235:S. 3455–3463, 8 2011.
- [Str00] E. Strickland: *Minimalism:Origins*, Indiana University Press, 2 Aufl., 2000, ISBN 978-0-253-21388-4.
- [tec20] *Jahresreport der deutschen Games-Branche 2020*, Techn. Ber., game – Verband der deutschen Games-Branche e. V., 2020, URL:<https://www.game.de/wp-content/uploads/2020/08/game-Jahresreport-2020.pdf>, besucht am 01.12.2020.

- [TES52] J. Veroff T. E. Shipley: *A projective measure of need for affiliation*, *Journal of Experimental Psychology*, Bd. 43(5):S. 349–356, 1952.
- [Tre11] M. J. Tresca: *The Evolution of Fantasy Role-Playing Games*, McFarland Company, inc., 2011, ISBN 978-0-7864-5895-0, ePUB.
- [Urh08] D. Urhahne: *Sieben Arten der Lernmotivation Ein Überblick über zentrale Forschungskonzepte*, *Psychologische Rundschau*, Bd. 59(3):S. 150–166, 2008.
- [VR16] T. Stamm V. Ritschl, R. Weigl: *Wissenschaftliches Arbeiten und Schreiben Verstehen, Anwenden, Nutzen für die Praxis*, Springer, 2016, ISBN 978-3-662-49908-5, ebook.

Anhang

Selbstständigkeitserklärung

Hiermit erkläre ich, daß ich die vorliegende Arbeit selbstständig angefertigt, nicht anderweitig zu Prüfungszwecken vorgelegt und keine anderen als die angegebenen Hilfsmittel verwendet habe. Sämtliche wissentlich verwendete Textausschnitte, Zitate oder Inhalte anderer Verfasser wurden ausdrücklich als solche gekennzeichnet.

Mittweida, den 4. Dezember 2020

Nico Daniel Noé