



**HFU - Projektsemester
2020/2021**

Initiatoren:
Geyer, Fabian - MIB
Willbold, Lisa - MIB

Gruppe:
Knecht, Leandro - MIB
Wolf, Vivian - MIB
Kölle, Felix - MKB
Böttcher, Daniel - OMB

Inhalt

Prozess der Ideenfindung	5
Grundidee für die Geschichte	7
Details zum Hauptcharakter	7
Details zur Spielumgebung/des Settings	7
Gegner	8
Eigentlicher Spielstart	8
Ingame / Spielablauf / Hauptelemente	9
Hintergrund Geschichte	9
Beispiel einer „Halluzination“/ Rätsel im Wald	10
Beispiel Mühle	11
Game Design Dokument	15
Zusammenfassung des Spiels	15
Genre	15
Charaktere	15
Story	15
Thema	16
Setting	16
Gameplay Mechaniken	16
Story - Ergebnisse	20
Steckbrief des Hauptcharakters - Charakterbogen	20
Steckbrief des Hauptcharakters - Personalitätsskala	25
Story - Texte	26
Abschiedsbrief Pastor	26
Notiz an Pastorentür	26
Notizen - Pastorenzimmer	26
Brief der Schwester	29
Kirchenschiffrätsel	30
Story - Antagonist	31
Game Design - Ergebnisse	35
Zusammenschrieb der Rätsel im Kirchenschiff	35
Game Design - Blueprints	39
Verwendung von Blueprints vs C++	39
Das Lesen von Briefen und Notizen	40
Scanner - Markieren von Objekten	42
Inventarsystem	46
Inspizieren von Gegenständen	53
Öffnen und Schließen von Türen	57

3D Ergebnisse	59
Kirche	59
Kirchenorgel	59
User Interface - Ergebnisse	65
Entwicklung eines UI-Elements	65
Weiterentwicklung der Idee UI-Element	66
Ausarbeitung von 3D Widgets	69
Designänderung und aktueller Stand	70
Dokumentationen	73
Erstellung von 3D-Inhalten für UE4	73
Namenskonvention innerhalb von Unreal Engine 4	78
Namenskonvention im Erstellungsprozess	79
Source Control	80
Installation der Helix Core Clientanwendungen	84
Fazit	92
Ausblick	92

IDEENFINDUNG

Prozess der Ideenfindung

Einer der ersten Schritte im Prozess der Ideenfindung für das Projekt ist die Festlegung auf ein Genre, welches mit seinen typischen Kernfunktionen in eine passende Umgebung eingebracht werden soll. Relativ früh wurde deshalb entschieden, dass zum Beispiel Rollenspiel und Shooter nicht das gewünschte Genre darstellen. Dem gegenüber war es eine einfachere Entscheidung für ein Genre aus dem Bereich Adventure, Mystery und Puzzle.

Als Nächstes folgt eine Überlegung, in welchem Setting, also der Spielwelt sich das Abenteuer unserer Spielfigur abspielen soll. Aus vielen Überlegungen gingen zwei besonders passende Umgebungen hervor. Zum einen der Ansatz einen dichten Dschungel mit Ruinen, Fallen und Rätseln, deren Optik angelehnt an die Kulturen der Maya und Azteken. Der andere Ansatz war, das Geschehen in die Stadt oder eine größere Dorfgemeinde zu platzieren, welches sich optisch im Zeitraum des ca. 18. Jahrhunderts in Zentraleuropa befindet. Aus den genannten Ansätzen sind mehrere Ideen zur Geschichte und Hintergrundinformationen zur Spielwelt entstanden.

1. Unser Hauptcharakter ist ein Schatzjäger, welcher sich während seines Abenteuers dort in einer Zeitschleife befindet. Um den Grund des Fluchs der Zeitschleife zu erfahren, müssen Puzzles gelöst und geheime Orte erkundet werden. Als zentraler Punkt ist ein Schiff vorgesehen, welches dem Charakter gehört und sich auf dem Fluss Befindet, dem der Spieler folgt. Weitere Themen sind die Verwendung von mystischen und okkulten Themen, sowie dem Wiederspielwert durch unterschiedliche Herangehensweisen bei den Rätseln.

Aus dieser Idee entnommen sind die Grundidee der Zeitschleife, sowie ein zentraler Punkt, an dem sich der Spieler wiederfinden kann.

2. Der Hauptcharakter ist ein Ermittler, der beim Verfolgen seiner Spuren in ein mysteriöses, verlassenes Haus gerät. Das Haus selbst erweist sich mehr und mehr als lebendig, bis sich herausstellt, dass es selbst der Gegner ist. Durch das Erkunden und Lösen von Geheimnissen und Rätseln kann jedoch Gegenwehr geleistet werden, um zu entkommen. Als weitere Waffe kann das Haus Risse zu einer Parallelwelt öffnen oder von dort Kreaturen beschwören. Die Realitäten können aufeinander einwirken und sich beeinflussen.

Auch hier wieder das Thema eines Zentrums, in dem sich die Geschichte abspielt, sowie die Elemente Puzzle, Rätsel und übernatürliche Elemente.

3. In dieser Idee ist der Hauptcharakter im Beruf eines Detektivs unterwegs. In einem kleinen Dorf geschehen Morde, welche es zu verhindern und aufzuklären gilt. Dabei hilft ein magisches Amulett, welches eine Reise in die eine parallele Vergangenheit ermöglicht, in der die Morde noch nicht geschehen sind. In dieser Zeit können Hinweise gesammelt und sogar Zeugen befragt werden, um den Mörder zu fassen.

4. Die Spielumgebung der Idee ist in einem Dorf am Rande Wiens angesiedelt. Das Dorf ist komplett menschenleer. Durch Erkundung werden Rückblenden ausgelöst, die die Geschichte eines Anschlags erzählen. Plot Twist der Geschichte ist, dass der Hauptcharakter selbst der Attentäter ist.

Hier kommt die Idee des Dorfes mehr ins Spiel. Das Dorf ist wie auch im finalen Projekt zuerst menschenleer und der Protagonist weiß nicht wieso.

5. Eine Idee, die das Konzept Wald, insbesondere Dschungel in den asiatischen Raum bewegt. Ein junger Fischer, dem ein Boot gehört (welches später als Hub verwendet wird), möchte sein Glück versuchen und dem Ruf des Kaisers folgen, eine unbekannte Insel zu erforschen und belohnt zu werden. Bei der Erkundung gerät er in eine Zeitschleife und kann sich nach und nach durch das Lösen von Rätseln einen Weg in die Freiheit erschließen.

Hier ist wieder das Konzept der Zeitschleife und der Rätsel vorhanden.

6. Als Ausgangspunkt der Idee ist ein kleines Städtchen im 18./19. Jahrhundert gedacht gewesen, in dem der Spieler in einem Gasthaus startet. Die Stadt ist komplett verlassen. Das Setting ist neblig, dunkel, bedrückend. Puzzles und mysteriöse Elemente lösen Flashbacks aus, die die Geschichte eines Anschlags erzählen. Der Täter ist noch auf freiem Fuß. Es stellt sich heraus, dass der Hauptcharakter der Täter mit zwei Persönlichkeiten ist. Hier kämpft die Persönlichkeit des Detektivs gegen die von einem religiösen Kult kontrollierte Persönlichkeit an.

Eine weitere Idee mit einer kleinen Stadt/größerem Dorf als Umgebung. Verworfen wurden die Idee mit der doppelten Persönlichkeit, sowie auch schon die zuvor genannte Idee mit dem Anschlag auf die Bevölkerung und der religiöse Kult als Gegner.

7. Der Hauptcharakter wacht auf einem außer Kontrolle geratenen Schiff auf und muss es erst wieder unter Kontrolle bringen, bevor er auf einer verlassenen Insel strandet, auf welcher sich ein dichter verwirrender Dschungel befindet. Bei Nacht erreicht er darin einen Tempel, bricht aber durch den Boden ein und muss sich aus einem Labyrinth voller Rätsel befreien.

Diese Idee orientiert sich am Dschungel Setting. Entnommen und deckend mit anderen Ideen sind die Ruinen, welche den Weg in die finale Idee gefunden haben.

8. Die Idee ist im asiatischen Raum mit dem Sohn des Kaisers als Protagonisten erdacht worden. Dieser wird auf eine Expedition geschickt und möchte sich auf diesem Wege seinem Vater als würdig erweisen. Die Idee greift bekannte Themen Wald, Tempelanlage (diesmal asiatisch), Rätsel und Puzzle auf, ergänzt aber durch vermeintliche Halluzinationen. Diese sollen beim Spieler für ein Zweifeln und Hinterfragen der Realität sorgen.

Eine Ansiedlung im asiatischen Raum wurde zu Gunsten der Umgebung in zentral Europa/Wiener Gegend verworfen.

Grundidee für die Geschichte

Der Hauptcharakter erwacht in einem verlassenen Dorf. Er weiß nicht, wer er ist oder warum er hier ist. Trotz seiner Amnesie folgt er den Hinweisen, die er um sich herum finden kann und löst dabei Rätsel auf seinem Weg. Auf diesem Weg begegnen ihm Erscheinungen, die nicht von dieser Welt sind und er stellt sich die Frage, ob alles nur eine Illusion ist oder okkulte Mächte am Werk sind.

Diese mysteriösen Mächte erscheinen dem Hauptcharakter des Spiels zum Teil als übernatürliche Kreaturen, die es zu bekämpfen gilt. Manchmal ist es jedoch schlauer, die von den Kreaturen ausgehende Gefahr zu umgehen und die Umgebung zum eigenen Vorteil zu nutzen, um einem direkten Kampf aus dem Weg zu gehen.

Was zu Beginn der Geschichte noch nicht bekannt ist: der Hauptcharakter befindet sich seit seiner Ankunft im Dorf in einer Zeitschleife. Der Moment des Spielbeginns ist bereits der „zweite“ Durchgang für die Spielfigur, welche sich daran aber nicht erinnern kann.

Das Dorf dient in jeder Iteration der Zeitschleife als zentraler Ausgangspunkt, quasi als Hub.

Details zum Hauptcharakter

Der Hauptcharakter ist männlich. Sein Alter ist im Bereich von Mitte 30 bis Anfang 40 angesetzt. Sein Beruf ist Pastor/Priester (christlich) und er besitzt einen dementsprechenden Kleidungsstil.

Details zur Spielumgebung/des Settings

Die Geschichte des Spiels soll sich zu einem großen Teil innerhalb einer kleinen Stadt bzw. eines etwas größeren Dorfs abspielen, welches inmitten der Berge von einem dichten Wald umgeben den zentralen Anlaufpunkt (Hub) darstellt.

Optisch orientiert sich die Spielwelt an der ländlicheren Region von Wien und Umgebung.

Auf Grund der vorherig eingegangenen Ideensammlung für das gesamte Setting, war neben der Umgebung als reiner Wald auch eine Ansiedlung am Meer im Gespräch. Dies hätte als weitere Elemente ein großes offenes Gewässer, sowie Klippen im nordischen Stile und einen Leuchtturm als Erweiterung des Dorfes eingeführt. Zu Gunsten der visualisierten Stimmung innerhalb des Dorfes für den Spieler wurde für einen reinen Wald abgestimmt, der bedrückend, einsperrend und als Heimat des Unbekannten beiträgt.

Neben dem eher ländlich gehaltenen Stil der Ortschaft existiert ein vom Nordischen inspiriertes Thema. Dies beinhaltet Runen, Symbole, einen „Tribal“ Style der Bewohner des Waldes, sowie deren Rituale. In jenem Stil gehalten sind auch die auffindbaren Ruinen, sowie ganze Rätsel oder Teile davon innerhalb der Spielgeschichte, die es zu lösen gilt.

Gegner

Neben Runen und Rätseln, die es zu erkunden gibt, stellen sich dem Hauptcharakter auch Gefahren in den Weg. Zu diesen zählen übernatürliche Kreaturen, die vom Erscheinungsbild her an mythischen Wesen, wie Nagas und Wendigos angelehnt sind. Begegnet der Spieler diesen Kreaturen, so muss er sich diesen im Kampf stellen oder einen alternativen Weg finden, diese lebendigen Hindernisse zu Überwinden.

Eigentlicher Spielstart

(Originaltext von vor weiteren Besprechungen/Treffen, die den aktuellen Startpunkt in die Kirche des Dorfes verlegt haben.)

Der Hauptcharakter erwacht in der Mitte des Dorfes. Er weiß nicht, wo oder wer er ist. Das Dorf ist auf den ersten Anschein hin menschenleer. Sein einziger Hinweis ist ein Brief mit einer Adresse und einem Absender in seiner Tasche. Der Spieler kann sich entweder zuerst frei umsehen und die Ortschaft und die Gegend (soweit bis dahin zugänglich) erkunden oder anhand der Adresse das erste Haus und somit den ersten Hinweis finden. Zu diesem Zeitpunkt ist noch nicht klar, dass dies bereits das zweite Mal ist, dass der Charakter die Ortschaft erkundet und nach Hinweisen sucht, um aus der Schleife und damit dem Dorf zu entkommen und dabei vielleicht auch Überlebende zu finden. Er hat sein Gedächtnis nach einem Vorfall in seinem ersten Durchgang verloren. Durch Flashbacks an bestimmten Stellen bzw. nach dem Finden eines Gegenstandes oder dem Lösen eines Rätsels zum Finden eines Hinweises wird die vorherige Geschichte erklärt.

Ingame / Spielablauf / Hauptelemente

Auf der Suche nach neuen Hinweisen erkundet der Spieler durch den Hauptcharakter das Dorf/Stadt und die Häuser. Neben dem reicheren Zentrum gibt es nach außen hin auch die ärmlicheren Häuser und Gebiete. Die gesamte Gegend ist von einem dichten, verwirrenden Tannenwald umgeben in den sich der Spieler begeben um muss, um den Verlauf der Geschichte voran zu treiben. Auf seinem Weg findet der Spieler Rätsel, die es zu lösen gilt. Jedes dieser Rätsel kann auf unterschiedliche Arten gelöst werden, je nachdem welche Gegenstände bereits vorher gefunden wurden. Neben Rätseln und Ruinen im nordischen / Tribal Stil verbergen sich auch Gefahren im Wald. Diese treten in Form der Ureinwohner und deren Monster und Kreaturen auf. Es gilt auch diese zu überwinden. Dabei stehen dem Spieler teils Waffen zur Verteidigung gegen manche Waldbewohner zur Verfügung. Manche dieser übernatürlichen Kreaturen lassen sich wiederum nur unter Zuhilfenahme der Umgebung besiegen. Dies beinhaltet natürliche Fallen oder eine sich bietende Möglichkeit zur Flucht.

Hintergrund Geschichte

Version A (family version)

Der HC kommt ursprünglich aus dem Dorf/der Stadt in dem unser Spiel angesiedelt ist. Seine Familie hat dort schon seit vielen Generationen vor ihm gelebt. Bereits in seiner Kindheit wurden Geschichten über Urvölker und deren Rituale von einer Generation an die nächste weitergegeben. Die Kinder, inklusive des Hauptcharakters nehmen die Geschichten nicht ernst und machen eine Mutprobe beim Spielen in einem abgelegenen Teil des Waldes daraus (z.B. weiter in den Wald zu einer alten „Ritualstelle“ gehen oder sie treffen tatsächlich auf die vergessen geglaubten Ureinwohner, welche gerade ein Ritual durchführen). Der HC wird dabei (versehentlich) ein Teil des Rituals. Er irrt im Wald umher, findet spät abends wieder heraus, aber erinnert sich im Dorf nicht mehr daran, was passiert ist.

(Alternative mit weniger Backstory: Da seine Familie schon immer dort wohnt wird der „Schutz“ von Generation zu Generation weitergegeben, sowie die Geschichten über die Urvölker. Das Wissen geht über die Jahre verloren bzw. wird als Unfug abgetan)

Jahre später, der HC ist Pastor seiner Gemeinde, glücklich verheiratet und hat ein Kind. Er erhält Nachricht und wird in die nächst größere Stadt (2 Wochen Kutschfahrt) einberufen, um neue Regeln etc. zu empfangen bzw. Report abzulegen.

Als er zurück kommt und sich auf seine Familie freut, ist niemand mehr zu finden.

Start von Durchlauf 0, welchen der Spieler durch Flashbacks miterlebt. Der Spieler selbst startet bei Durchgang 1, nachdem der HC seine Erinnerungen bereits einmal verloren hat und quasi wieder am Anfang seiner Suche steht.

Version B (sister version)

Hauptcharakter und Familie leben wie in Version A seit Generationen in der Ortschaft und Umgebung. In dieser Version verlässt unser HC das Dorf, um seiner Berufung zu folgen und sein Amt in einer größeren Stadt zu erlernen. Wie bereits in Version A ist er auch hier in seiner Kindheit mit den Ritualen in Berührung gekommen und so ein Teil des Ganzen geworden (Alternative mit Schutz über Generationen weitergegeben kann hier auch passen).

Nach einiger Zeit erhält er von seiner Schwester einen Brief, die ihn wieder zurück im Dorf haben möchte.

(good version: sie will heiraten und möchte, dass er die Trauung vornimmt / evil version: sie hat Angst, weil etwas Komisches im Dorf vor sich geht und sonst keine Familienangehörigen mehr da sind)

Der Hauptcharakter fährt daraufhin zurück und findet das Dorf leer vor.

Iteration 0 als Start wie bei Version A...

Version C (work edition)

Der Hauptcharakter kommt nicht wie in den beiden zuvor genannten Versionen ursprünglich aus der Ortschaft, sondern von außerhalb. Er ist Priester/Pastor/Ermittler und wird durch den Brief eines ansässigen Pastors oder Dorfoberhaupts um Hilfe gebeten, weil etwas Merkwürdiges in der Umgebung vor sich geht.

Das Merkwürdige sind die Rituale der Ansässigen und Ureinwohner, welche die Rituale nur alle 200 Jahre genau da abhalten können.

Er folgt dem Hilferuf bzw. wird von seinem Vorgesetzten dorthin beordert. Anfang der Geschichte / Iteration 0, wie in Version A und B.

Beispiel einer „Halluzination“/ Rätsel im Wald

Der Spielcharakter befindet sich im Dorf und hört aus der Ferne das Läuten einer Hafenglocke. Er folgt dem Geräusch und wird durch das Umfallen eines großen, alten Ankers ein Stück entfernt erschreckt. Der Anker wird von einer unsichtbaren Kraft um die nächste Hausecke und somit außer Sicht gezogen. Der Charakter folgt dem Anker und findet ihn ein Stück Richtung Waldrand gezogen, jedoch mit abgerissenem Tau. Er bemerkt eine Schneise im Wald.

Je weiter der Charakter dem Waldweg folgt, desto mehr verändert sich der Himmel und die Baumkronen. Das Wiegen der Baumspitzen wird mehr und mehr zu einer Wellenbewegung. Das Licht, das durch die Wipfel fällt, ändert seine Farbe immer weiter ins bläuliche. Der Charakter bemerkt im Gehen einen großen Schatten am Boden, der sich über ihn hinweg bewegt. Er blickt nach oben. Dort fährt ein Schiff über ihm vorbei und verschwindet in der Ferne. Der Charakter folgt der Richtung des Schiffes. Der Wald lichtet sich. Auf der neu entdeckten Lichtung steckt der Bug eines alten Schiffes (Wikingerschiff oder Segelschiff?) abgebrochen mit der Spitze zum Himmel im „Waldboden“. Mittlerweile schimmert das Licht wie durch die Meeresoberfläche auf die Lichtung. Am Schiff selbst wachsen Korallen und Tang, der Boden wird sandig. Das Schiff muss erkundet werden, wofür der Spieler Rätsel lösen muss, um ans Ziel zu gelangen (z.B. man hat vorher einen Schlüssel gefunden und kann hier eine Truhe öffnen). Nachdem der Schatz bzw. der nächste Hinweis geborgen ist, geht ein Beben durch das Schiff. Mahlende Geräusche werden lauter, das Knirschen von Holz ist zu hören. Der Spieler muss schnellstmöglich den Weg aus dem Wrack herausfinden und dabei brechenden oder herunterfallenden Gefahren ausweichen. Ist dies erfolgreich geschafft beginnt eine Filmsequenz, in der mitverfolgt werden kann, wie das Schiff im Boden, wie Treibsand verschwindet.

Beispiel Mühle

Am ärmeren, ländlicheren Rande des Dorfes, etwas versteckt hinter Bäumen ist eine Windmühle. Die Flügel der Windmühle drehen sich im bisherigen Verlauf nicht, weswegen der Eingang versperrt ist, auch wenn man den Ort vorher findet/aufsucht.

Nach einem zuvor gelösten Rätsel wird mittels einer kurzen Filmsequenz (z.B. Wind frischt auf, Raben fliegen tief am HC vorbei in Richtung Mühle) der Blick auf die Mühle gerichtet. Der Spieler folgt dem Hinweis und stellt fest, dass es nun einen möglichen Zugang zur alten Mühle gibt. Innerhalb der Mühle müssen Kisten verschoben, Zahnräder zusammengebaut und Gewichte mittels Mehlsäcken richtig verteilt werden, um zur Spitze zu gelangen. Hier kann es mehrere Lösungswege nach oben geben, in Abhängigkeit dessen, was der Spieler zuvor an Gegenständen gefunden hat (z.B. eine Axt, um ein Seil zu zerschneiden). Wenn die Spitze erreicht ist, wird eine weitere Sequenz freigeschaltet, die den Blick wieder in eine Richtung lenkt, in der er weiter gehen soll.

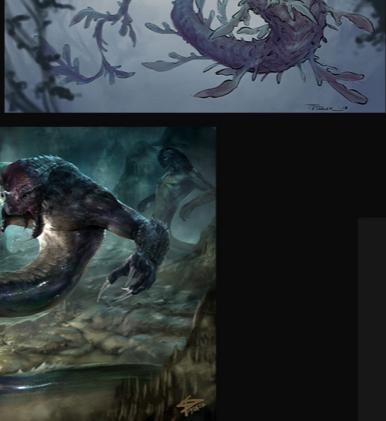
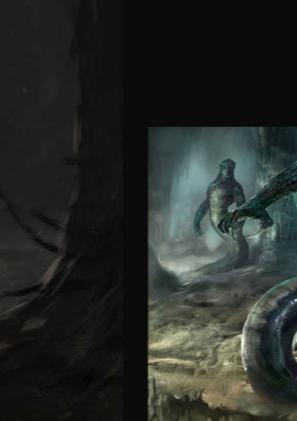
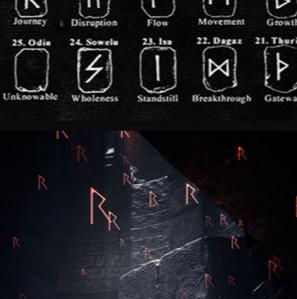
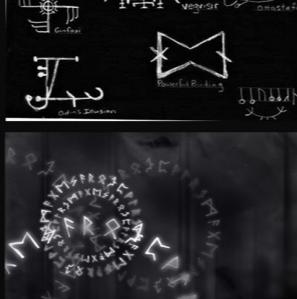
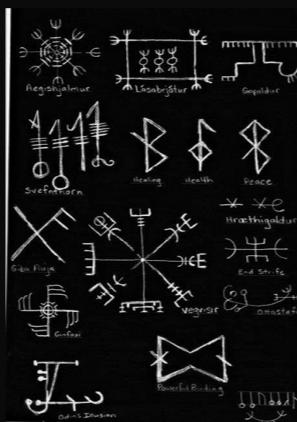
(Beispiel: Im Dorf oder am Rande des Dorfes sind plötzlich Fackeln zu sehen, die in Richtung eines neuen Weges verschwinden, der vorher nicht zugänglich war. Oder der Spieler/der Charakter bekommt zum ersten Mal eine Übersicht von oben und erkennt ein neues Zeichen, z.B. eine Form oder Rune die das gesamte Dorf bildet oder ein großes Zeichen in der Dorfmitte.)

Location & Mood



Story & Creatures

Character



GAME DESIGN DOKUMENT

Game Design Dokument

Zusammenfassung des Spiels

In Ouroboros erlebt der Spieler die Geschichte des Pastors Jakob. Dieser befindet sich zu Beginn des Spiels in einer menschenleeren Kirche – der seines Heimatdorfes. Schnell bemerkt er, dass das komplette Dorf verlassen zu sein scheint und erkennt die alten, übernatürlichen Kräfte, die dafür verantwortlich sind.

Der Spieler löst hierfür Rätsel und Puzzles, um dem Geheimnis auf die Spur zu kommen. Er interagiert mit seiner Umgebung und findet so nach und nach Hinweise, welche ihn mit den nötigen Informationen versorgen, die er braucht, um das große Rätsel zu lösen und das Dorf zu retten.

Genre

Adventure, Puzzle, Mystery

Bei Ouroboros handelt es sich um ein Third-Person Mysteryadventure mit Rätselementen.

Charaktere

Der Hauptcharakter: Pastor Jakob

Jakob ist ein 38 Jahre alter, zurückhaltender und freundlicher Pastor, der sein Heimatdorf verließ, um in der großen Stadt als Pastor der Kirche zu dienen. Eher unscheinbar in der Menschenmenge, hat seine Anwesenheit bei anderen immer eine leichte Nervosität ausgelöst. Nach dem Erhalt eines Briefes seiner Schwester kehrt er in sein Heimatdorf zurück.

Story

Die Aufgabe des Spielers liegt darin, durch das Lösen von Rätseln und Puzzles durch die Umgebung zu navigieren. Es sollen Informationen gesammelt werden, was hinter dem Verschwinden der Dorfbewohner steckt und wie es rückgängig gemacht werden kann. Jakob wird auf gefährliche Kreaturen und mysteriöse Waldbewohner stoßen, welche sich ihm entgegenstellen und welche es zu überwinden gilt.

Der Spieler tritt gegen einen uralten, keltischen Dämon an, der das Dorf und seine Bewohner fest in der Gewalt hat. Um dies zu bewältigen geht er ein unerwartetes Bündnis mit einer geheimnisvollen Schamanin ein. Auch sie ist auf der Suche nach einer Möglichkeit den Dämon zurückzudrängen, denn auch die Bewohner ihres Volkes sind vom Verschwinden betroffen. Er erfährt von einer Prophezeiung, welche unmittelbar mit dem Auftauchen des Dämons zu tun hat. Diese besagt, dass der dieser seinem zwischen-

weltlichen Gefängnis entkommen und seine ursprüngliche Macht wiederherstellen will. Trotz fehlender Kräfte ist er in der Lage die gefangenen Menschen in seine Diener zu verwandeln und sich so von deren Energie und der Lebenskraft der entführten Bewohner zu ernähren. Wird er nicht aufgehalten, so stellt der Dämon nach und nach seine Kraft wieder her und versammelt eine Armee, um die alten Götter zu besiegen –und die Welt in eine ewige Nacht zu hüllen. So heißt es zumindest.

Mit seiner Verbündeten sucht Jakob eine Lösung, dies zu verhindern und das Dorf und seine Bewohner zu retten.

Thema

Die Geschichte soll düster und mysteriös sein, aber dennoch positive und schöne Momente durchscheinen lassen.

Setting

Das Spiel ist angesetzt im 19. Jahrhundert in einem Dorf in Zentraleuropa. Das Dorf ist umgeben von einem dichten, mysteriösen und uralten Wald und generell sehr abgelegen.

Umgeben von Bergen gibt es nur einen wirklichen Weg in das Dorf – ein Grund warum viele Bewohner ihre Heimat für größere Städte verlassen. So auch Jakob, der in einer größeren Stadt die Stelle eines Pastors angetreten hat. Er wird jedoch zurückgerufen von einem angstefüllten Brief seiner Schwester – und so nehmen die seltsamen Geschehnisse ihren Lauf.

Bei seiner Rückkehr findet er ein menschenleeres und gänzlich in Nebel gehülltes Dorf vor.

Gameplay Mechaniken

Ziele

- Primär: Das Retten des Dorfes und seiner Bewohner
- Sekundär: Rätsel und Puzzle lösen, Informationen sammeln, Gegner umgehen, Zwischenabschnitte meistern

Mechaniken

Die verwendeten Mechaniken lassen sich in mehrere Kategorien unterteilen:

- Die Interaktion mit der Umgebung:
Die Interaktion mit der Umgebung soll für den Spieler ein zentrales Gameplayelement sein. Durch

die Möglichkeit, mit verschiedensten Objekten in seiner Umgebung zu interagieren und so Reaktionsketten auszulösen, kann der Spieler in der Geschichte voranschreiten und die ihm gestellten Rätsel überwinden. Der Spieler ist in der Lage bspw. durch das Anzünden von Kerzen für Licht zu sorgen, durch das Umstoßen von Fässern Kreaturen zu entkommen oder gar neue Wege aufzudecken. Zudem ist er in der Lage Gegenstände aufzunehmen und zu transportieren. Bei Bedarf sollen diese Gegenstände abrufbar sein.

Die Interaktion mit der Umgebung stellt für die weiteren Gameplayelemente die Grundlage und ist daher essenziell.

- Rätsel und Puzzle

Durch das Lösen von Rätseln und Puzzles schreitet der Spieler in der Geschichte und dem Spiel voran. Mit diesen Rätseln und Puzzles sollen die Kombinatorik und die Problemlösefähigkeiten des Spielers unter Beweis gestellt werden. Durch das erfolgreiche Lösen einer Aufgabe wird der Spieler mit Informationen zu den Geschehnissen, hilfreichen Items wie Schlüsseln oder das Öffnen von neuen Passagen und Gebieten belohnt.

Diese Rätsel selbst können sich auf ein Objekt beziehen oder auch Teile der Umgebung beinhalten.

- Die Umgebung analysieren

Für die bessere Erkennbarkeit von Objekten, welche dem Spieler eine Interaktion erlauben, kann der Spieler eine „Scan-Funktion“ nutzen, die ihm solche Objekte markiert und diese von der Umgebung hervorhebt.

- Items inspizieren

Gewisse Items und Objekte erlauben dem Spieler, sich diese innerhalb des Inventars genau anzuschauen und zu inspizieren. Hierfür wird das Modell des Objekts zentral und großflächig auf dem Bildschirm angezeigt. Der Spieler hat in diesem Untersuchungsmodus die Möglichkeit, das Objekt frei zu bewegen und zu rotieren. So kann der Spieler gegebenenfalls versteckte Details finden, welche ihm hilfreiche Informationen für den Fortschritt im Spiel liefern können.

- Gegner bekämpfen oder umgehen

Für den Pastor stellen Gegner eine direkte Gefahr dar, die der Spieler überwinden muss. Ein direkter Kampf ist aus Sicht des Pastors keine gute Idee, daher stehen dem Spieler verschiedene alternative Möglichkeiten zur Verfügung. Beispiele hierfür sind das Nutzen von Items oder die Interaktion mit der Umgebung, mit denen er solche Kämpfe umgehen oder für sich entscheiden kann.

Fortschritt und Herausforderung

Die Rätsel und Puzzles sollen mit dem Fortschritt des Spiels schwerer werden. Auch die Gegner sollen immer schwerer zu umgehen sein, so dass der Spieler mit äußerster Vorsicht und Gerissenheit agieren

muss, um zu seinem Ziel zu kommen.

Versagen

Der Spieler kann nur im Kampf versagen. Wird er mehr als drei Mal getroffen, stirbt die Spielfigur und wird an den letzten Speicherpunkt zurückgesetzt.

User Interface

Das User Interface wird auf das Nötigste beschränkt, sodass das Spiel seine volle visuelle Wirkung entfalten kann und der Spieler so immersiv wie möglich die Geschichte erlebt.

- Objekte, mit welchen interagiert werden kann, werden bei direktem Stand daneben markiert
- Die Gesundheit des Spielers wird durch rötliches Blinken des Bildschirms indiziert, sollte er getroffen worden sein. Ansonsten wird auf eine direkte Lebensanzeige verzichtet
- Menüs und das Inventar werden innerhalb eines Notizbuches angezeigt, ebenso wie das tatsächliche Tagebuch, welches Jakob während des Spiels mit sich führt
- Speichern des Spielstandes ist an Altären möglich, an welchen sich Jakob niederlässt um zu beten und sein Tagebuch zu führen

Art Style

Bei Ouroboros handelt es sich um ein düsteres und realistisches 3D Mysteryadventure. Es soll alles grau und trist wirken, aber dennoch eine eigene Schönheit dadurch ausstrahlen. Das Dorf ist in Nebel gehüllt und die Gebäude wirken, vor allem von innen, verlassen.

Der Wald soll märchenhaft, düster und angsteinflößend wirken – mit dem Hintergedanken der andauernd lauernden Gefahr, die sowohl vom Antagonist, als auch von den Befallenen ausgeht. Alles wirkt sehr verwachsen und dicht und auch hier dominiert der Nebel.

Technische Merkmale

- Verwendete Technologien
 - o Unreal Engine 4
 - o Blender
 - o Substance Suite (Painter, Designer)
 - o Quixel Suite (Megascans, Mixer, Bridge)
 - o Photoshop
- Hardwarevoraussetzungen
 - o Plattform: PC
 - o Eingabegerät: Maus/Tastatur sowie Controller
 - o Aufgrund des angestrebten hohen visuellen Detailgrads sollen moderate Hardwareanforderungen gestellt werden.

STORY

Story - Ergebnisse

Steckbrief des Hauptcharakters - Charakterbogen



ALLGEMEIN

Name: Jakob („Gott beschützend“)

Alter: 38

Herkunft: Zurzeit wohnt der Charakter in der Stadt in einer kleinen Wohnung nicht weit von der Kirche entfernt, in der er dient.

Er wurde in einem kleinen Dorf geboren. In diesem ist er bis zu seinem 10. Lebensjahr aufgewachsen. Seine Pubertät verbrachte er im Kloster. Nach seinem 18. Lebensjahr zog er jedoch wieder zurück in das Dorf, bis er mit 33 Jahren in die Stadt zog.

PHYSISCH

Haare: Kurz bis mittellanges, leicht ungepflegtes Haar (bisschen länger als im Bild)
Backenbart

Narben &

Male: Brandmal am linken Unterarm, entstanden durch ein Ritual in der Kindheit

Kleidungsstil: Schlichte Farben

Knielanger Mantel

Leichtes Hemd und Weste

Einfache Stoffhose

Melonenhut

Kreuzkette (entweder an der Weste befestigt oder als Kette)

Einfache, über den Knöchel ragende Lederschuhe mit kleinem Absatz

Aura & Wirkung: Wirkt nach außen als ein guter, freundlicher Mensch. Er fällt in der Menge jedoch nicht sehr auf und ist eher zurückhaltend. Seine Anwesenheit löst bei manchen Menschen ein leicht nervöses Gefühl aus. Er wirkt als könnte ihn emotional nicht

viel zu Fall bringen, jedoch ist er innerlich sehr verletzlich.

MENTAL

Persönlichkeit:

- o Leicht verbittert
- o Perfektionistisch
- o Gibt sich nach außen hin sehr stark, ist aber durchaus verletzlich
- o Möchte immer das Beste für die Menschen
- o Leicht eingebildet
- o Sensibel
- o Handelt wenn es um seine Familie geht sehr emotional und nicht rational
- o Handelt in Dingen, die die Kirche betreffen sehr rational auf die Tugenden der Kirche bezogen
- o Ist leicht gelangweilt vom eigenen Leben (sieht Brief seiner Schwester vielleicht als Chance sich zu beweisen?)
- o Immer freundlich
- o Hat starke Überzeugungen
- o Heldenkomplex
- o Ehrlich
- o Sehr sicher in allen Handlungen verbunden mit Gott, hat jedoch keinen Plan vom richtigen Leben und wirkt auf der Straße sehr unsicher
- o Versteht keinen Humor
- o Skeptisch gegenüber anderen Menschen
- o Loyal

Tugenden:

- o Wohlwollen
- o Keuschheit
- o Freundlichkeit
- o Gutherzigkeit
- o Hingabe (für seinen Beruf)
- o Demut

Laster:

- o Stolz (auf seine Stellung und seinen Ruf, leicht eingebildet)
- o Neid/Eifersucht (auf Ehemann seiner Geliebten)
- o Begehrten (einer Frau, obwohl er ein Diener Gottes ist)

Hobbies: Lesen (wissenschaftliche Bücher, worin das Wesen Gottes gesucht wird), Spazieren gehen

Motivation: Der einzige Sinn des Lebens besteht darin auf Jesus Christus zu warten, das Wort Gottes zu verkünden und dadurch ewiges Leben im Paradies zu erlangen.

Ängste:

- o Große Verlustängste
- o Angst vor der Hölle
- o Angst Menschen, die er liebt, zu enttäuschen
- o Den Glauben zu verlieren in schweren Zeiten

Schwächen:

- o Unverständnis für andere Glaubensrichtungen
- o Emotional verbunden mit seiner Familie, lässt sich dabei von Emotionen lenken

Abneigungen:

- o Ekel vor Käfern
- o Abneigung gegenüber anderen Glaubensrichtungen

KULTUR & SOZIAL**Arbeit/Schule:**

- o Pastor in der Stadt in kleinerer Zweitkirche
- o Ausbildung in Domschule und Kloster absolviert
- o Träumt davon einmal in der Stadtkirche dienen zu dürfen

Sozialer Status: Er ist ein angesehener Pastor in seinem Kirchenbezirk, der sehr geschätzt wird von den Schwestern und Brüdern der Gemeinde. Von ihnen wird er als Vorbild angesehen. Besitzt kleine Wohnung in der Stadt, die nur spärlich eingerichtet ist.

Kostbarster Besitz: Kreuzkette

Häufige Orte: Kirche, Park?, Bibliothek

FAMILIE & BEZIEHUNG**Eltern:**

Vater: gestorben als er 20 war, schlechtes Verhältnis, hat Kinder körperlich misshandelt, Alkoholiker

Mutter: bei der Geburt der jüngsten Schwester gestorben, sehr enges Verhältnis

Geschwister: 4 Brüder und 2 Schwestern: Bruder, Bruder, Schwester, Bruder, MC, Bruder, Schwester
1.Bruder gestorben an Lungenentzündung
3. ältester Bruder in der Armee gestorben
Sehr innige Beziehung zu jüngster Schwester

Freunde: Priester aus der Stadt ist sein bester Freund. Die beiden haben sich auf der Domschule kennengelernt.
Ansonsten wenig Freunde.

Beziehung: Trotz seines Berufes war/ist der Hauptcharakter verliebt in eine Freundin der jüngsten Schwester. Er schreibt ihr regelmäßige Briefe, jedoch fand nie eine körperliche Verbindung

untereinander statt. Seine Geliebte ist bereits verheiratet und hat mehrere Kinder.

Storyline

Motivation: Charakter möchte seine Schwester retten, da er zu große Angst hat sie zu verlieren. Er möchte etwas gegen den Schwund des Dorfes unternehmen, weil es seine Heimat ist, mit der er sehr viel verbindet. Er sieht dies als eine Art Chance aus seinem kleinen, langweiligen Leben zu entkommen und tatsächlich etwas zu erleben, nimmt als Vorwand jedoch, dass Gott ihn geschickt hat, um die Menschen zu retten.

Ziel: ursprünglich -> Schwester und deren Familie retten;

als er sieht, dass das ganze Dorf verschwunden ist -> herausfinden was geschehen ist, um etwas dagegen zu unternehmen

Konflikt: Er weiß nicht was passiert ist und warum es passiert ist. Die Waldbewohner verfolgen eine andere Glaubensrichtung, die die Lösung zum Rätsel zu sein scheint. Dies ergibt für Jakob jedoch keinen Sinn, da andere Glaubensrichtungen keine Wahrheiten beinhalten. Zudem müsste er durch diese Annahme mit den „bösen“ Waldbewohnern zusammen arbeiten. Dies spreche gegen seine Prinzipien.

Vergangenheit

4 Jahre: Tot der Mutter. Dieser löste bei Jakob Verlustängste aus.

Während der gesamten Kindheit starke Misshandlungen durch den Vater -> Trost und Schutz in der Kirche gefunden -> Daher sehr starke Bindung zum Glauben.

7 Jahre: Kinder spielen im Dorf Fangen und geraten dadurch immer tiefer in den Wald. Als sie an eine Ritualstelle der Waldbewohner kommen, soll der junge Hauptcharakter sich in die Mitte der Ritualstelle begeben. Als er den Stein in der Mitte betritt kommt ein Wind auf, der die Sicht zu den restlichen Kindern verhindert. Die Kinder bekommen Angst und rennen davon, wodurch Jakob den Waldbewohnern allein gegenüber steht. Diese haben bereits durch Prophezeiungen auf diesen Moment gewartet und schließen ihn damit in ihr Ritual ein. Er wird zum Schutzengel der Waldbewohner und des Dorfes, damit die Dämonen nicht austreten können und die Welt ihr normales Leben weiterleben kann. Jakob taumelt nach diesem Ereignis allein durch den Wald zurück ins Dorf und wird direkt auf dem Marktplatz bewusstlos. Seine Erinnerungen hören in dem Moment auf wo der Wind angefangen hat die Sicht zu versperren. Seitdem hat er ein Brandmahl auf seinem linken Unterarm, welches dasselbe Symbol trägt wie der Stein in der Mitte der Ritualstelle auf den er sich gestellt hat.

10 Jahre: Besuch der Domschule, anschließend Diener im Kloster

14 Jahre: Ältester Bruder stirbt an Lungenentzündung.

18 Jahre: Messediener der Dorfgemeinde, in der er aufgewachsen ist.

Geliebte heiratet. Er ist Teil der Zeremonie.

20 Jahre: Tot des Vaters.

23 Jahre: Pastor der Dorfgemeinde

26 Jahre: 3. Ältester Bruder stirbt in der Armee.

33 Jahre: Brief vom Kirchenoberhaupt: Er wird berufen in der Stadt als Pastor zu dienen, da sein Ruf überall bekannt ist und er stets ein treuer Diener der Kirche war.

36 Jahre: Erster Brief der Schwester, dass immer mehr Leute plötzlich verschwinden. Die Polizei geht von einem Serienmörder aus.

38 Jahre: Brief der Schwester, dass dringend Hilfe benötigt werde. Sie weiß nicht wie lange sie noch durchhält (letztes Lebenszeichen der Schwester).

Jakob macht sich auf den Weg in das Dorf, um seine Schwester zu retten.

Auto-Biografie:

Wenn der Charakter eine kurze Biografie über sich selbst schreiben müsste, was würde darin stehen?

„Es gibt nicht viel zu meinem Leben zu sagen. Ich wuchs in einem kleinen Dorf in Mitten eines Waldes auf. Schon in meiner Kindheit verbrachte ich viel Zeit in der Kirche. Sie schenkte mir Trost in düsteren Zeiten. Der Glauben war alles, was von Anfang an ein Teil meines Lebens war und nie fort gegangen ist. Aus diesem Grund ist alles in meinem Leben auf ihn ausgerichtet. Die Zeit in der Domschule und im Kloster stärkten meine Beziehung zu Gott und Jesus Christus, welche ich nun an die Brüder und Schwestern meiner Gemeinde weitergeben kann, damit auch sie von den Sünden erlöst werden können.“

Steckbrief des Hauptcharakters - Personalitätsskala

Character Reference Sheet: Expanding on the Basics Personality Traits

Name: _____

Extrovert	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	Introvert
Honest	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Deceptive
Sporadic	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Predictable
Confident	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Meek
Humble	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Conceited
Selfless	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Selfish
Organized	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Messy
Humorous	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	Serious
Critical	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Creative
Logical	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Irrational
Optimist	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Pessimist
Patient	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	Short-tempered
Trusting	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	Suspecting
Lazy	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Productive
Book smart	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Street smart
Health nut	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Sweet tooth
Sexual	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	Chaste
Loyal	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Backstabbing
Reserved	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Reckless

If they had to write a short bio about themselves, what would they say? (Write this in first person as the character.)

Crystalmmburton.com

Story - Texte

Abschiedsbrief Pastor

Vater du hast mich verlassen. Die Sünde, der Teufel, es wütet in mir. Lass dies Gift sowohl meinen sterblichen als auch meinen unsterblichen Körper befreien. Erlöse mich und vergib mir diese letzte Sünde. Sie scheint mein letzter Ausweg zu sein, bevor mich die Hölle auch unter den Lebenden einnimmt.

Notiz an Pastorentür

Achtung! Diese Tür ist auf Wunsch des Pastors verschlossen. Ich habe ihn eingesperrt, damit er wieder zu sich finden kann.

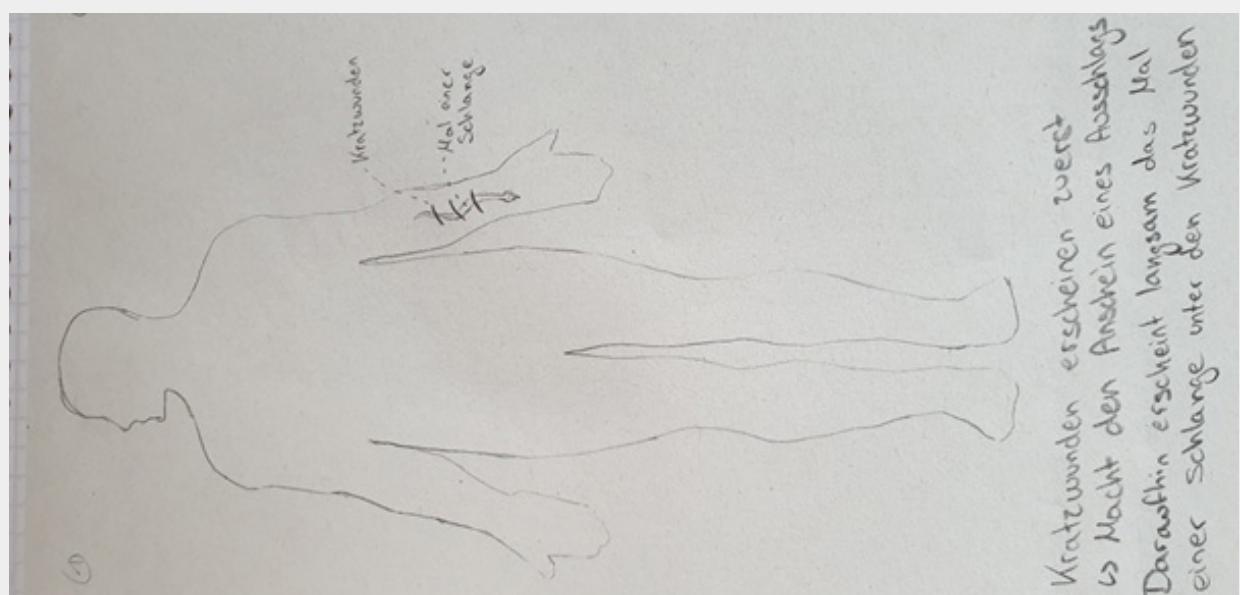
Bruder Lukas, Orgelspieler

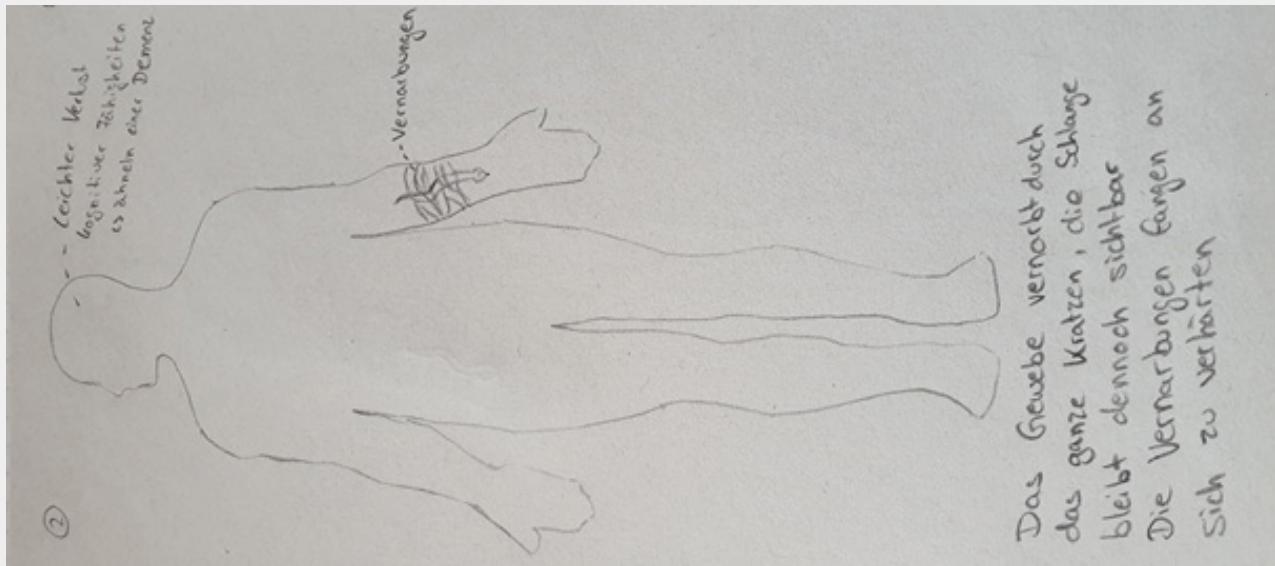
Notizen - Pastorenzimmer

Mein Körper, er verändert sich. An meinem Arm zeichnet sich das Bild einer Schlange. Was ist das? Wo kommt es her? Ich werde Doktor Schönwald aufsuchen. Er wird mir sicher helfen.

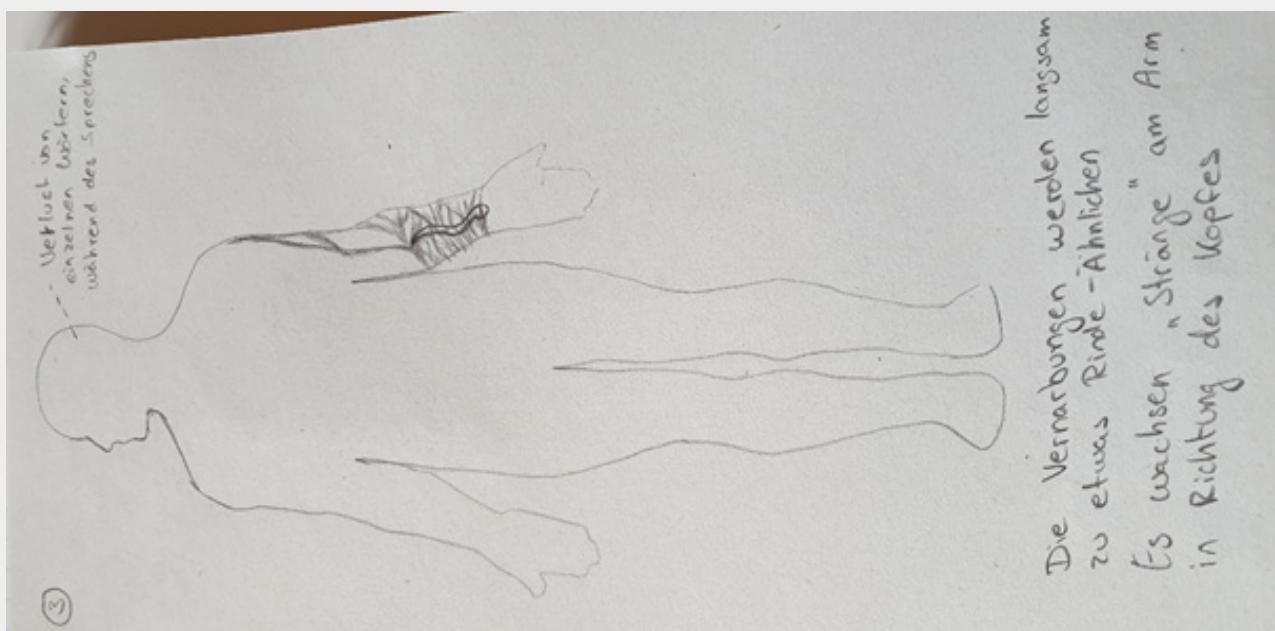
Immer öfter kommen Brüder und Schwestern zu mir, die verzweifelt um Hilfe bitten. Ihre Kinder verschwinden. Die Polizei geht von einem Mörder aus, der es auf die Kinder abgesehen hat. Ich werde für sie beten. Mögen ihre Seelen beim Herrn in Frieden ruhen.

Zeichnungen des Pastors über Verlauf der „Krankheit“:





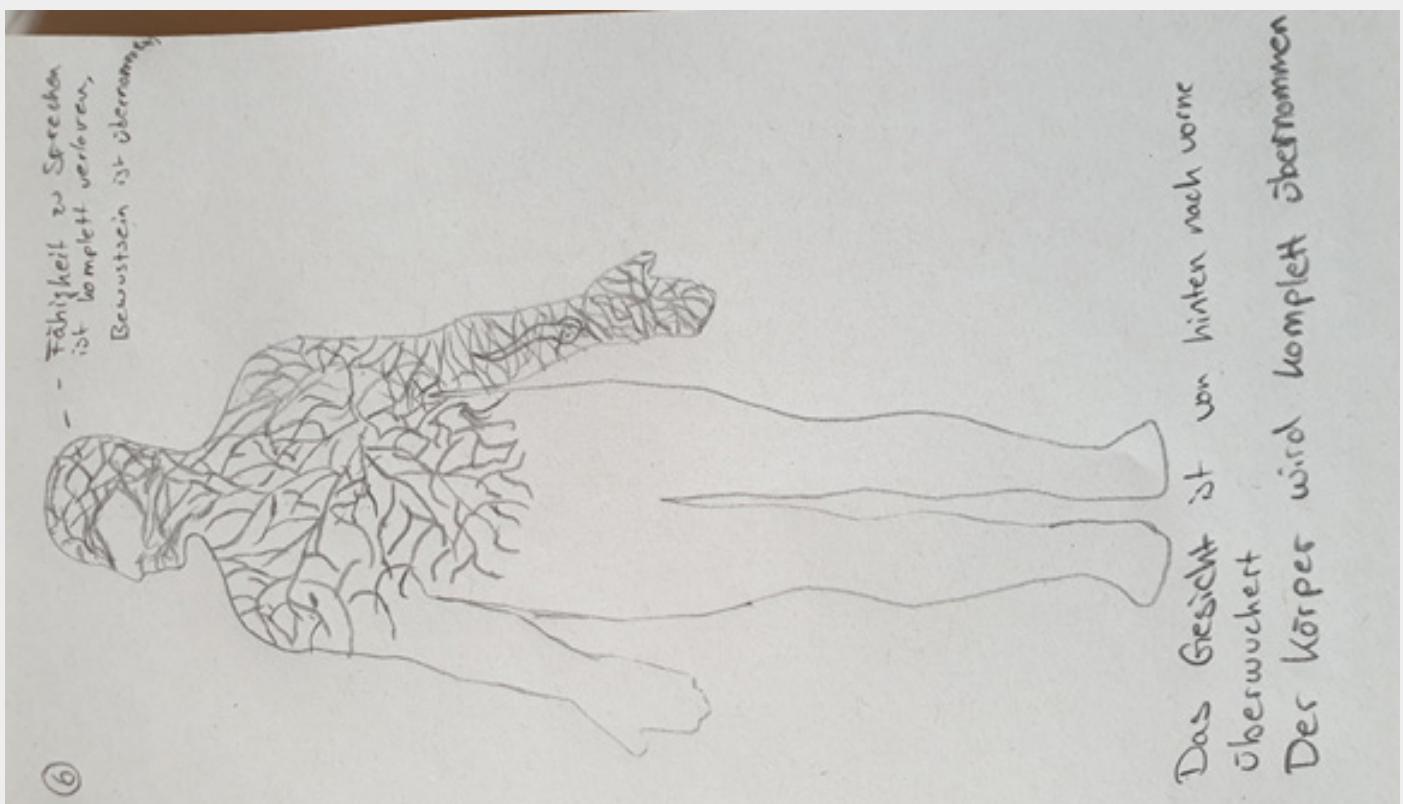
Das Gewebe vernichtet durch das ganze Körperteil, die Schäde bleibt dennoch sichtbar
Die Vernarbungen fangen an sich zu verhärten



Die Vernarbungen werden langsam zu etwas Rinde - Ähnlichen Es wachsen "Stränge" am Arm in Richtung des Kopfes



"Stränge" richten Knochen vernarben und verhärten sich ebenfalls Übergang in Richtung Herz, vom Herz ausgehend breite sich die Krankheit auch in die Organe und damit in den ganzen Körper aus.



Ich werde in die heutige Predigt das Thema Hoffnung mit einbringen. Die Bürger des Dorfes scheinen jegliche Hoffnung zu verlieren. Die Kirche wird leerer an den Sonntagen, doch immer mehr Menschen kommen zur Beichte. Ihrer Meinung nach würde sie das verschonen. Diesen Glauben teile ich nicht. Nichts als Hirngespinst.

Die Stimmen in meinem Kopf werden lauter. Sie schreien und doch erreichen sie niemanden... Verliere ich meinen Verstand?

Schwester Erika bat mich heute um ein Wort. Ich schätze ihren Bruder Pastor Jakob sehr. Sie hat große Angst ihre Kinder zu verlieren. Ich kann von Glück sprechen, dass sie ihrem Glauben treu bleibt, dennoch ist sie sehr besorgt. Ich möchte den Kontakt zu ihr erhalten. Vielleicht kann ich durch einen intensiveren Kontakt zu den Gemeindemitgliedern herausfinden was hier geschieht.

Mir sind heute zum ersten Mal in meiner Lebenszeit Wörter entfallen. Wörter. Wie ist das möglich? Es scheint, als wären sie verflogen.

Brief der Schwester

Liebster Jakob,

ich hoffe dieser Brief erreicht dich noch bevor wir vom Teufel verschlungen wurden. In meiner Einschätzung ist dies der letzte Brief, den ich an dich adressieren werde. Pastor Benedikt ist dem Wahnsinn verfallen, es laufen beinahe keine Kinder mehr im Dorf herum und selbst die Kirche gleicht einem Friedhof. Der Teufel hat uns übernommen, die Säulen des Glaubens drohen über uns einzustürzen.

Jakob, bitte ersuche uns. Hol uns zu dir, lass eine Kutsche schicken, die uns hier weg bringt. Ich weiß, dies ist unsere Heimat, doch dieses Dorf fühlt sich nicht mehr nach einem Heim an. Ich brauche dich. Wir brauchen dich.

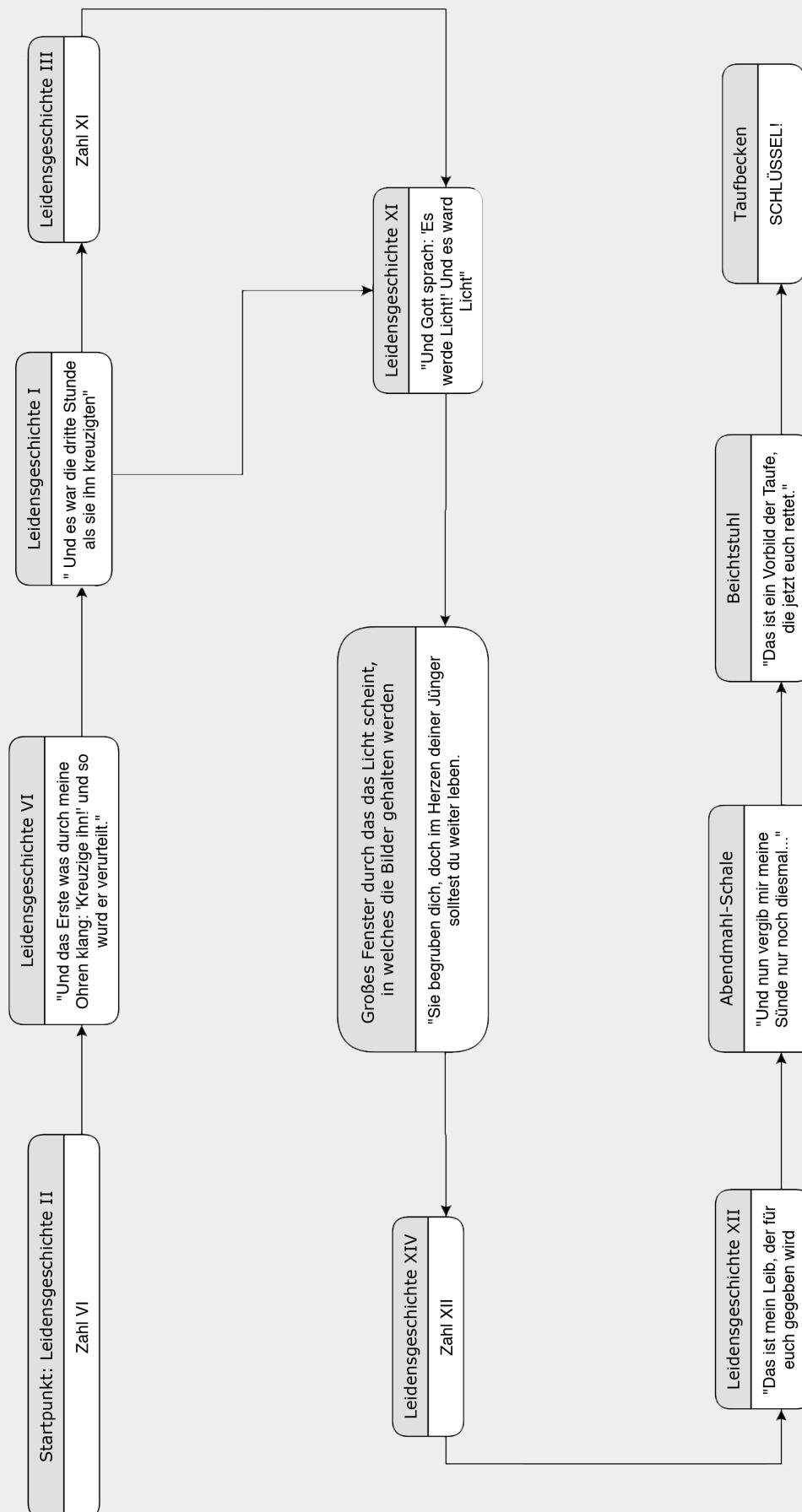
Meine Vorstellungskraft sagt mir, dass du der Einzige seist, der uns jetzt noch erlösen kann. Du bist ein Diener Gottes, Jakob. Wer sonst, könnte uns nun retten?

Ich flehe dich an. Die Verzweiflung übersteigt meinen Verstand.

Falls dies unser letzter Brief sein sollte, sollst du wissen, dass ich stets an dich denke. Ich danke dir für deine Herzensgüte mir und den Kindern gegenüber. Mutter wäre stolz, könnte sie sehen, was aus dir geworden ist.

Erika

Kirchenschiffrätsel



Story - Antagonist

Unser Antagonist ist ein uralter Dämon, längst vergessen und ignoriert. Es handelt sich um den Dämon NAME. Sie ist Gestaltwandlerin, meist wandelt sie zwischen einer hübschen Frau, einer Schlange und einem Raben.

NAME war vor langer Zeit als finstere Göttin der schwarzen Magie, aber auch als Göttin der Jagd bekannt und gefürchtet. Es wurde überliefert, dass sie Jäger, die die Regeln des Waldes nicht respektierten, in Gestalt eines Raben tief in die Wälder gelockt haben soll, um sie dann zu verführen und in Form der Schlange dann zu verspeisen. Es heißt des Weiteren, dass sie aber auch gewöhnliche Menschen, Pilzesammler bspw., ebenfalls verführt und verspeist haben soll. Aber all das scheinen nur Märchen zu sein, um die Kinder aus den Wäldern fern zu halten.

All das soll geschehen sein, bevor die alten Götter die Welt von ihrer tyrannischen Herrschaft befreiten und sie auf ewig in die Zwischenwelt verbannten. In den Überlieferungen wird sie meist von ihren 3 Söhnen begleitet, ebenfalls Gestaltwandler. Jedoch traten sie ausschließlich in Form von Wargen, mit dämonischen Hörnern, auf. Die Götter hatten Mitleid mit den 3 Brüdern und kamen auf eine andere Bestrafung.

NAME's Söhne sollten auf ewig den Mond ziehen und so für den Tageszyklus verantwortlich sein. Sollten sie sich wehren – was sie regelmäßig versuchten – wurden sie von göttlichen Wachen ausgepeitscht und gemaßregelt. Im Volksmund wurde dies Blutmond genannt (es hieß, das Blut der 3 Brüder färbe den Mond rot) und war regelmäßig Anlass großer spontaner Feste.

Der letzte Blutmond ist schon sehr lange her und laut Überlieferungen kann das nur eins bedeuten: Eine uralte Prophezeiung scheint sich zu verwirklichen und die drei Brüder NAME3, NAME4 und NAME5 konnten der strengen Aufsicht und der Gewalt der alten Götter entfliehen und werden versuchen, ihre Mutter zu befreien.

Die Waldmenschen wurden von den Göttern vor langer Zeit damit beauftragt, NAME in der Zwischenwelt zu halten. Dafür waren regelmäßige Rituale notwendig, um das Portalschloss weiterhin mit Energie zu versorgen. All das haben sie Jahrhunderte lang im Verborgenen getan. Bis MC als Jungspund in eines der Rituale gestolpert ist. Seither scheint etwas nicht zu stimmen.

Im Laufe der Jahre, die zwischen dem Ritualvorfall und der Rückkehr des MC liegen sind nach und nach immer mehr Menschen verschwunden, in unregelmäßigen Abständen. Erst waren es vereinzelte Vorfälle und meist traf es auch eher die Einsamen und Ausschüssigen des Dorfs – Obdachlose, verrückte Alte oder Witwer/Witwen. Mit der Zeit wurden es jedoch immer mehr und so verschwanden auch Kinder. Erst wurde vermutet, sie hätten sich im Wald verlaufen und wurden gefressen – so traurig es auch ist. Aber es verschwanden immer mehr und der Nebel wurde immer dichter. Es wurde dunkler, es wurde heller, aber

alles in allem wurde es vor allem nebliger. Erklären konnte sich das jedoch niemand, zumindest niemand innerhalb des Dorfes. Es gab die alten Dorfverrückten, die etwas von einer Prophezeiung redeten, aber wer hört schon auf alte, senile und offensichtlich verrückte alte Menschen.

Das Dorf nahe der Ritualstelle wirkt für NAME als Nährboden. Sie braucht eine Armee, sie braucht Scherben um die weltlichen Jagdgründe zu übernehmen und es mit den Göttern aufzunehmen. Sie übernimmt nach und nach die Bewohner, welche in den Wald verschwinden und dort praktisch unsichtbar werden – denn deren Haut wird schwarz, die Augen leer und weiß und die Haare werden mit der Zeit zu Ästen. Auf den Schultern und den Armen beginnt Grünzeug zu wachsen und die Befallenen werden nach und nach zu einer Art Baummensch – ursprünglich NAME2 genannt. Dieser Effekt scheint vom Biss einer Schlange ausgelöst zu werden, welche dann aber mit den Pflanzen verwächst. Zudem scheint das alles Auswirkungen auf den geistigen Zustand der NAME2 zu haben und sie unter eine Art Gedankenkontrolle zu stellen. Mit den NAME2 kann sie zu ihrer alten Stärke zurückkehren, in dem sie deren Kraft absorbiert – wie eine Art Photosynthese. Die NAME2 sind mit ihr verbunden und sie nährt sich von deren Lebenskraft als Mensch, so lange, bis sie nur noch leere Hüllen sind. Perfekt für einen Angriff.

Die NAME2 sind im Wald praktisch unsichtbar und können sich lautlos bewegen. Sie kommunizieren über Telepathie mit NAME und untereinander – unhörbar für normale Menschen. Die NAME2 bewegen sich allerdings nur im Schutze des Waldes und der dichten Baumkronen, denn das Sonnenlicht schwächt sie. Ein Glück – für die NAME2 – dass es in DORFNAME selten sonnig ist, zumindest seit der MC zurückgekehrt ist. Doch wie wurde der Erste verwandelt? Entgegen der Vermutung der Schamanin ist NAME bereits längst der Zwischenwelt entwichen, jedoch fehlt ihr die Kraft eine andere Form als die der Schlange anzunehmen. Sie hatte genug Kraft einen Übernahmzauber in Schlangenform zu wirken. So hat sie über die Jahrzehnte nach und nach immer mehr Dorfbewohner, die töricht genug waren den Wald zu betreten, verwandelt und zu NAME2 gemacht. Doch ihre menschliche Form kann sie nach wie vor nicht annehmen.

NAME wurde, laut Überlieferungen, schon immer mit dem Mond in Verbindung gebracht, nicht zuletzt wegen ihrer Söhne. Dies erklärt auch, warum die Findlinge, rund um die Ritualstelle mit dem Portal, und auch die anderen Findlinge, die verteilt im Wald sind, mit Mondlicht aktiviert und gesteuert werden können.

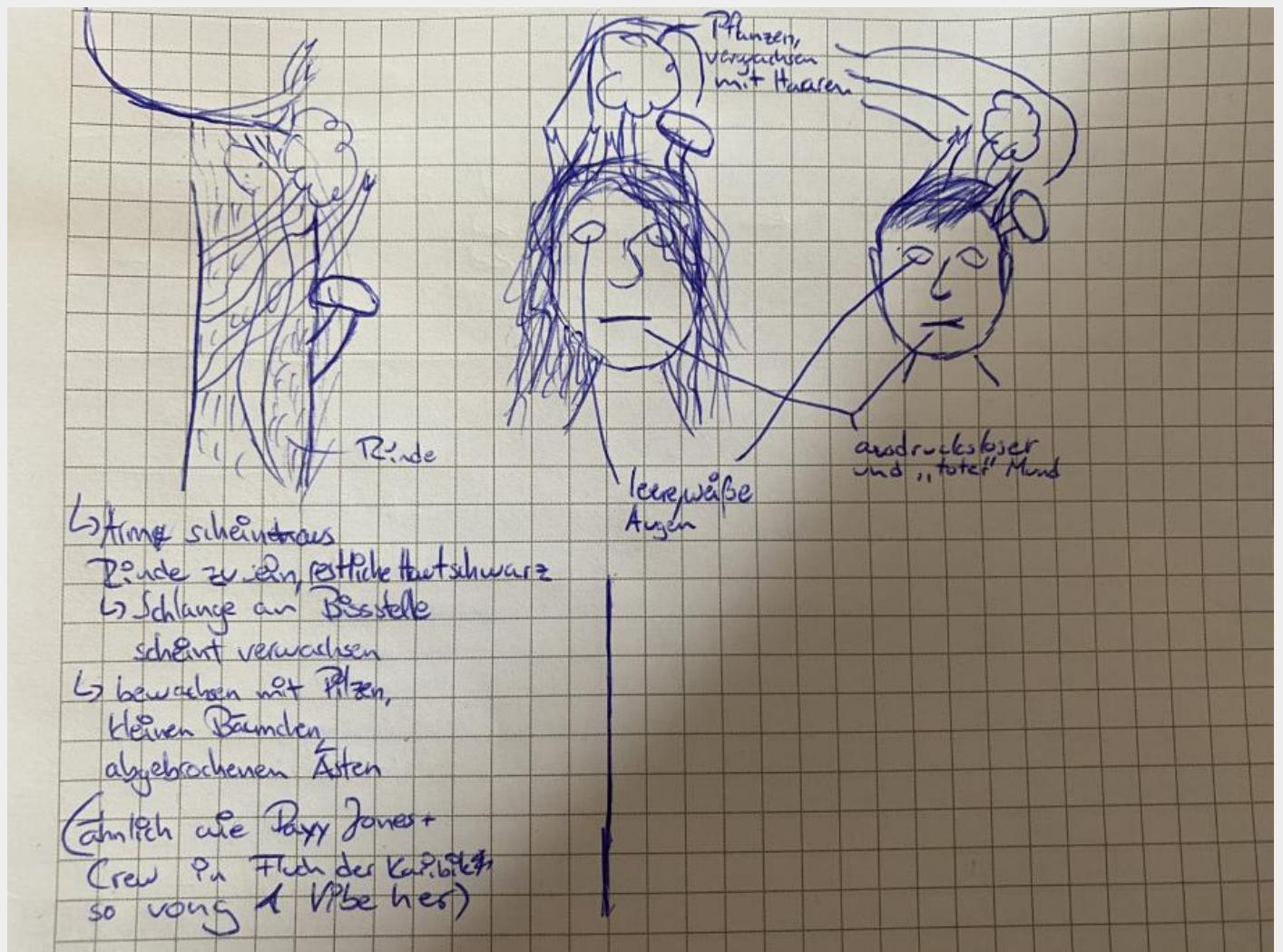
Des Weiteren scheinen die Rätsel, die mit den Findlingen verbunden sind, wie Schlüssel für das Portalsiegel zu fungieren. Um NAME2 von deren Fluch zu befreien muss NAME freigesetzt werden und überzeugt werden, den Bann zu lösen. Doch ist dies der einzige Weg? Es muss einen anderen Weg geben. Die Schamanin hat eine Idee, doch wer weiß ob das funktioniert.

Beschreibung NAME2: (sehr grobe Skizze mitgeschickt)

Werden von Schlange gebissen, Bissarm wird zu Rinde. Restliche Haut wird schwarz. Die Schlange wird zu Ast und „verwächst“ mit dem restlichen Arm. Es sprießen allerlei Pflanzen, Pilze und weitere Äste aus dem Arm.

Die Augen werden leer und weiß. Die Haare verwachsen mit Ästen und Pflanzen auf dem Kopf -> im Wald unsichtbar.

Mund wird „tot“ und wächst zu -> Kommunikation untereinander und mit dem Wirt über Telepathie.



GAME DESIGN

Game Design - Ergebnisse

Zusammenschrieb der Rätsel im Kirchenschiff

Kirchenlevel - Tutorial

Abfolge - Kurz:

1. Haupttür abgeschlossen. MC muss Schlüssel finden.
2. MC findet Brief an Pastorenzimmer der Hinweis auf Leidenstafel-Rätsel gibt.
3. MC muss jetzt Leidensportraits in richtige Reihenfolge bringen und im Kirchenschiff versteckte Hinweise finden, die schlussendlich zum Schlüssel für das Pastorenzimmer führen.
4. Im Pastorenzimmer findet MC die Leiche des Priesters. Hier findet er viele Hinweise auf die grobe Storyline und den Hinweis für das Vers- und Orgelrätsel.
5. MC bekommt durch Hinweise heraus, dass er eine bestimmte Tonabfolge auf der Orgel spielen muss, um so an den Schlüssel des Mausoleums zu kommen.
6. Sobald er den Schlüssel hat, geht er ins Mausoleum. Dort findet er sich als erstes in den Katakomben wieder. Hier kommen zwei/drei Rätsel.
7. Erstes Rätsel: Kerzenhalter an Säulen in die richtige Reihenfolge bringen.
Zweites Rätsel: Kerzen richtig anordnen
Drittes Rätsel: Zahlenrätsel mit den Todesdaten der Verstorbenen.
8. Nach dem zweiten oder dritten Rätsel geht es für den MC in die Freiheit. Man sieht Dorf- und Waldkulisse.

Abfolge - Ausgeschrieben:

Der Protagonist wacht auf dem Altar auf und steigt davon herunter. Er ist sichtlich verwirrt und hat für ein paar Sekunden leichte Flashbacks von seinem vorherigem Loop (leichter Glitch Effekt) in dem er vor einem der Monster geflüchtet ist.

Er bemerkt, dass die Kirche komplett verwüstet ist. Weil er verunsichert ist, ruft er erstmal in die Gewölbe hinein um zu wissen, ob sich außer ihm noch jemand in der Kirche befindet. Als er keine Rückmeldung bekommt, geht er in Richtung Tür.

Nun kann der Spieler sich frei bewegen und es werden unten die Steuerungsoptionen für den Charakter eingeblendet. Der Spieler kann sich nun im Kirchenschiff umschauen und bekommt so die ersten Steuerungsknöpfe beigebracht. (kann sich den Altar ansehen und dort speichern, kann sich das Kreuz ansehen und Informationen darüber erhalten usw.) Sobald der Protagonist die Haupttür erreicht hat, bemerkt er, dass die Tür verschlossen ist.

Der MC ist verwirrt, dass die Haupttür nicht offen ist und geht zurück ins Haupthaus.

Der Charakter sagt in Gedanken, dass er vielleicht in das Pastorenzimmer schauen sollte, da dort die Priester ihre Schlüssel und Wertgegenstände aufbewahren.

Der Charakter kann sich frei bewegen. Es wird das Inventarsymbol angezeigt und der Spieler kann sich im Inventar den Brief und sonstige Gegenstände, die der Charakter bei sich trägt noch einmal genauer anschauen und durchlesen.

Dort ist auch der Brief seiner Schwester, die ihn bittet in dieses Dorf zu kommen.

An der Tür des Pastorenzimmers hängt ein Brief des Orgelspielers der erzählt, dass er den Pastor auf seinen Wunsch in dem Zimmer eingesperrt hat, da der Priester vom „Bösen übermannt wurde“. Den Schlüssel zum Zimmer habe er gut versteckt. Hier gibt es die ersten Hinweise auf das Orgel- bzw. Kirchenschiff-Rätsel evtl. als Bibelvers oder Ähnliches.

„Da redete Jesus abermals zu ihnen und sprach: Ich bin das Licht der Welt. Wer mir nachfolgt, der wird nicht wandeln in der Finsternis, sondern wird das Licht des Lebens haben.“

Johannes 8:12 | LUT |

Um es noch mehr zu verdeutlichen kann der MC nach dem Lesen des Zettels auf eine der Leidenstafeln (an der Wand und am Altar) blicken, um noch deutlicher zu symbolisieren, dass das nächste Rätsel damit zu tun hat.

Der MC muss jetzt die einzelnen Leidensportraits gegen das Licht beim Altar halten und bekommt so eine Zahl (Station der Leidensgeschichte) und einen Vers (der den Ort beschreibt bei dem ein Hinweis versteckt ist). So befinden sich am Ende immer entweder eine Zahl (das nächste Leidensportrait der Reihenfolge) oder einen Vers, der beschreibt wo sich die nächste Zahl befindet. Hat man alle Portraits in die richtige Reihenfolge gebracht, findet man bei dem letzten Rätsel des letzten Bildes den finalen Schlüssel für das Pastorenzimmer.

Vivian - Discord

„(...) die Gemälde der Leidensgeschichte zwar in der richtigen Reihenfolge sind, jedoch abgehängen werden können. Hält man sie ins Licht beim Altar ist auf der Rückseite eine Zahl zu sehen, welche den Hinweis auf eine bestimmte Station der Leidensgeschichte gibt. Es sind jedoch nicht auf allen Bildern Zahlen, das wäre ja zu langweilig. Manchmal sind auch Ausschnitte aus Versen oder nur ein einzelnes Wort zu sehen, die auf bestimmte Dinge in dem Kirchenschiff verweisen, damit der Charakter auch im Kirchenschiff rumläuft. Dort ist dann jeweils der nächste Hinweis. So könnte der Spieler letztendlich auf den Beichtstuhl gelenkt werden, beispielsweise mit dem Vers „Und nun, vergib mir meine Sünde nur noch diesmal ...“ . Im Beichtstuhl könnte dann eine Notiz gefunden werden, die auf die Taufe verweist, beispielsweise „Das ist ein Vorbild der Taufe, die jetzt auch euch rettet“ (retten verweist in dem Fall auf den Ausweg) oder „... und predigte die Taufe der Buße zur Vergebung der Sünden“ (weil wir ja gerade im Beichtstuhl sind). Der Schlüssel

zum Pastorzimmer befindet sich also im Taufbecken.“

Wenn der MC den Pastorenraum betritt, sieht er die halb verwandelte Leiche des Priesters. Natürlich erschreckt sich der MC und möchte den Schlüssel zur Haupttür finden, um aus der Kirche zu kommen. Natürlich will er auch herausfinden, was dem Priester widerfahren ist und womit er es hier zu tun hat.

Nun kann der Spieler den Raum erforschen und kann viele Notizen und Hinweise finden, die weiter über die Story aufklären und Hinweise auf das nächste Rätsel geben. So erfährt der MC über Notizen des Orgelspielers, dass er auch die Luke/Tor/Durchgang zu den Katakomben/Mausoleum auf Befehl des Priesters abgesperrt hat und dass die Haupttür von außen versperrt ist und damit für eine Flucht aus der Kirche für den MC ungeeignet ist. Durch die Aufzeichnungen des Priesters erfährt man, dass er sicher stellen wollte, dass das Böse in ihm nicht aus dieser Kirche entkommen konnte, denn auch das Mausoleum hat einen zweiten Ausgang. Ein Hinweis, wo der Schlüssel zum Mausoleums des Orgelspielers ist, ist ein Bibelvers der im Raum versteckt wurde. Dort werden Buchstaben (Noten) groß geschrieben und zeigen so die Melodie an. Auf dem Zettel neben dem Gedicht/Vers/Psalm ist eine Orgel gezeichnet.

Fabian - Discord

„Was mir noch zur Story eingefallen ist: Wir wollen dem Spieler und auch unserem MC ja auch nach und nach vermitteln was in dem Dorf und mit den Menschen passiert ist. Gerade Notizen, Tagebucheinträge oder hinterlassene Nachrichten eignen sich da ja gut. Diese lassen sich wie wir teilweise eh schon vor haben mit Rätselinformationen verknüpfen. Für das Kirchenbeispiel hätte ich mir z.B. gedacht das man etwas über den Orgel-dude und den Pastor erfährt, die wir ja schon angesprochen hatten. Ein Szenario hier könnte sein, da unsere Opfer von den Kreaturen ja Zombiestyle irgendwie infiziert werden per Biss/Kratzen oder wie auch immer, hab ich mir gedacht das es den Pastor erwischt hat und zumindest er und der Orgel-dude der Folgen bewusst waren. Pastor wird auf Wunsch oder gegen seinen Willen im Pastorzimmer eingesperrt. Hier kann man eine Notiz vom Orgel-dude zu finden sein wo er den Schlüssel versteckt (so kommt man auch auf unsere Rätsel). Im Pastorzimmer findet man dann schließlich den Pastor der sich umgebracht hat und eine Notiz zurücklässt nach dem Motto: „Herr vergib mir diese schwerwiegende Sünde, doch mein ableben auf diese Weise scheint der einzige Weg nicht als Diener des Teufels zu enden.“ Das lässt den Spieler und MC rätseln was passiert es und über den Spielverlauf langsam zusammenstückeln was passiert ist.“

Die Melodie muss auf der Orgel eingespielt werden. Die Buchstaben sind in die Tasten eingraviert, um es Musikun erfahrenen so einfacher zu machen.

Es öffnet sich ein geheimes Fach und man bekommt den Schlüssel.

Der MC geht in die Katakomben und steht hier wieder vor einem Rätsel.

Hier gibt es mehrere Rätselvorschläge:

Daniel: Der Keller hat drei Säulen in der Mitte, an jeder Säule ist ein kurioser Kerzenhalter angebracht -> Der Spieler muss diese richtig nach mehreren christlichen Symbolen anordnen und kann so den Tunnel in die Freiheit/ oder Tunnel in den nächsten Rätselraum öffnen.

„Niemand aber zündet ein Licht an und bedeckt es mit einem Gefäß oder setzt es unter eine Bank; sondern er setzt es auf einen Leuchter, auf dass, wer hineingeht, das Licht sehe.“

Lukas 8:16 | LUT |

Felix: In dem Raum stehen 12 Kerzen, auf deren Halterungen verschiedene christliche Symbole abgebildet sind. Aufgabe ist es, die Kerzen in die zentrale Halterung zu stecken und anzuzünden. (Halterung in Form der Trinität.) Ein Werkzeug zum Entzünden der Kerzen ist oben in der Kirche. Sind die Kerzen erleuchtet, öffnet sich ein Durchgang und es geht in den nächsten Rätselraum oder in die Freiheit.

Vivian: Durch ein Zahlenrätsel mit den Todesdaten auf den Grabsteinen/Särgen etc. lässt sich ein Mechanismus aktivieren, der den Durchgang in den nächsten Rätselraum oder in die Freiheit öffnet.

Game Design - Blueprints

Für die Rätsel und Puzzles werden natürlich verschiedenste Mechaniken benötigt. Diese werden hier vorgestellt und näher erklärt.

Verwendung von Blueprints vs C++

Das mit der Unreal Engine bereitgestellte System der Blueprints ist eine Art des sogenannten „Visual Scripting“. Parallel dazu ist es möglich mit C++, einer textbasierten Programmiersprache, unter Verwendung eines Editors Mechaniken zu programmieren.

Je nach Komplexität und Fähigkeiten haben beide Systeme ihre speziellen Vor-und Nachteile. So steht dem Projekt mit C++ der komplette Code der Engine zur Verfügung, anstatt diesen in Teilen als Blueprints bereit zu stellen. Für größeren Funktionsumfang ist mit C++ auch eine höhere Performanz zu erwarten.

Mit Hilfe von Blueprints ist die „Visualisierung“ und das Nachvollziehen der Vorgänge einfacher zu erschließen, da oftmals verwendete Bausteine bereits zu Verfügung stehen. Ebenso ist keine vorherige Programmierkenntnis nötig. Prototypen können schneller technisch umgesetzt werden, da das System selbst Verbindungen setzt oder Fehler in der Verknüpfung direkt hervorhebt.

Für das Projekt selbst fiel die Auswahl des zu verwendeten „Programmiersystems“ eindeutig auf Blueprints. Die zuvor konzipierten Funktionen, Ideen und Rätsel können hiermit sehr schnell in Reihenfolge gebracht und als Prototyp implementiert werden. Auf Grund der maximalen Komplexität der Prototypen des Projekts gibt es nichts, was den Aufwand der Programmierung in C++ rechtfertigen würde.

Das Lesen von Briefen und Notizen

Für das Lesen von Briefen und Notizen muss eine Funktion bestehen, diese großflächig auf dem Bildschirm anzuzeigen und lesbar zu machen, denn in der Third-Person-Perspektive sind gerade solche kleinen Objekte schwer bis gar nicht lesbar.

Hierfür wurde eine simple Mechanik gebaut, die auf den Mausklick des Spielers reagiert und das Objekt großflächig als Widget anzeigt. Zudem werden Objekt bei Hover über dem Objekt grün markiert. Um das Widget zu schließen wird ein „Close“-Button eingebaut.

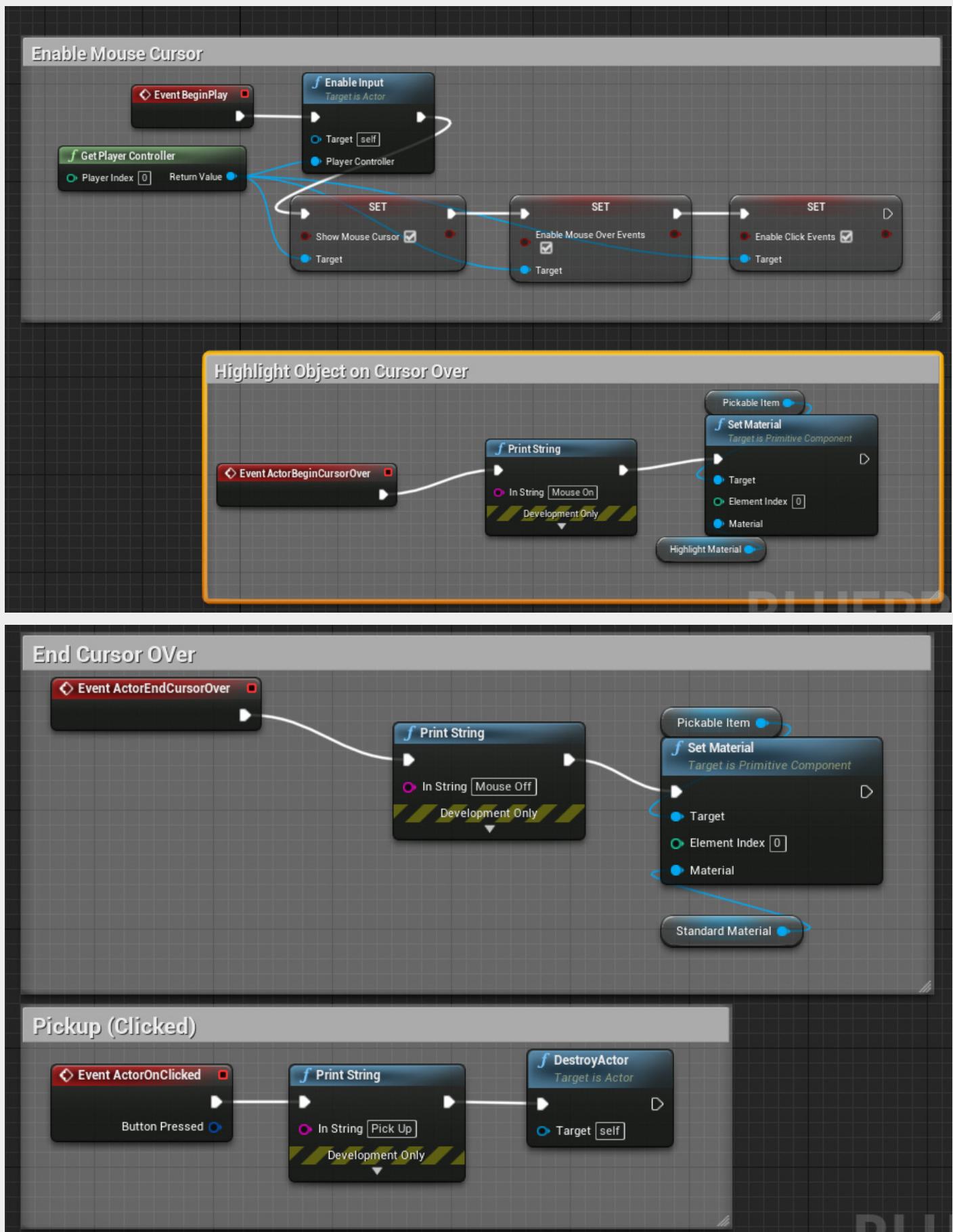


Position und Aussehen der Funktion im Spiel.

In den nachfolgenden Screenshots ist zum einen der Code für das Markieren und zum anderen der Code für das Anzeigen des Widgets zu sehen.

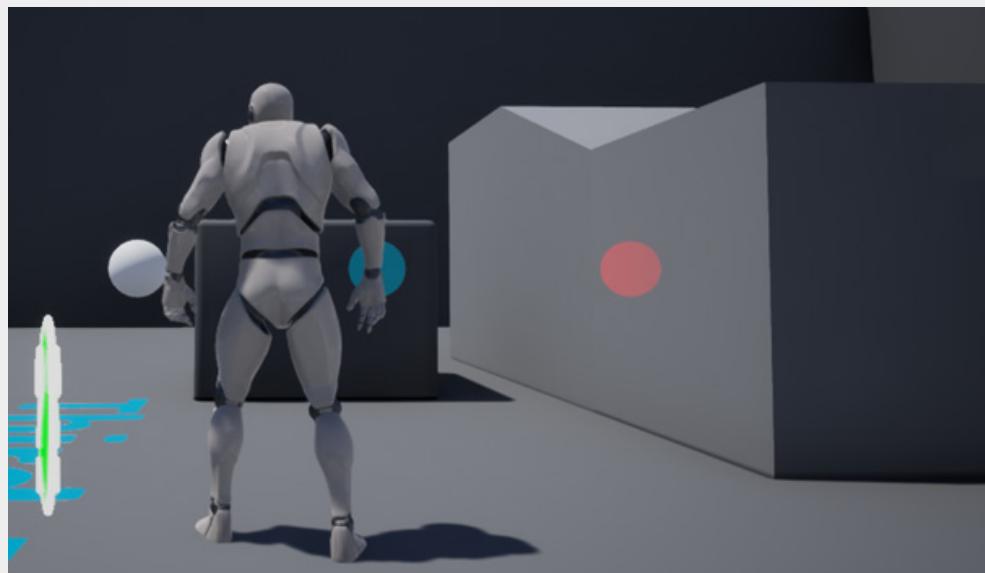
Der Code ist simpel und beeinträchtigt die Performanz nicht ausschlaggebend negativ.

Für jeden Brief und für jede Notiz muss ein eigenes Widget eingerichtet werden. Die Objekte, die im Spiel platziert werden, basieren auf einem Eltern-Element, welches den Code beinhaltet. Die Kind-Elemente müssen dann nur noch optisch angepasst werden.

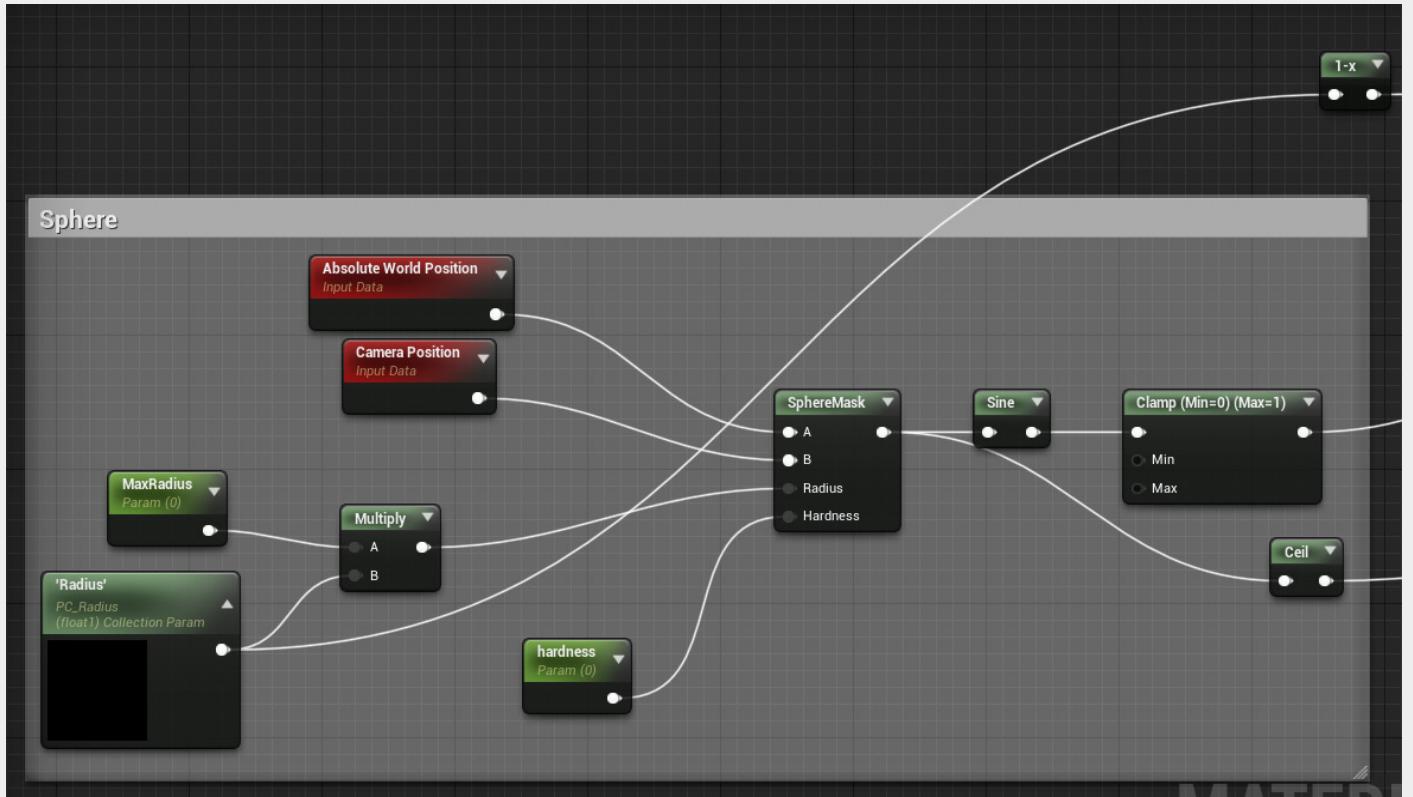


Scanner - Markieren von Objekten

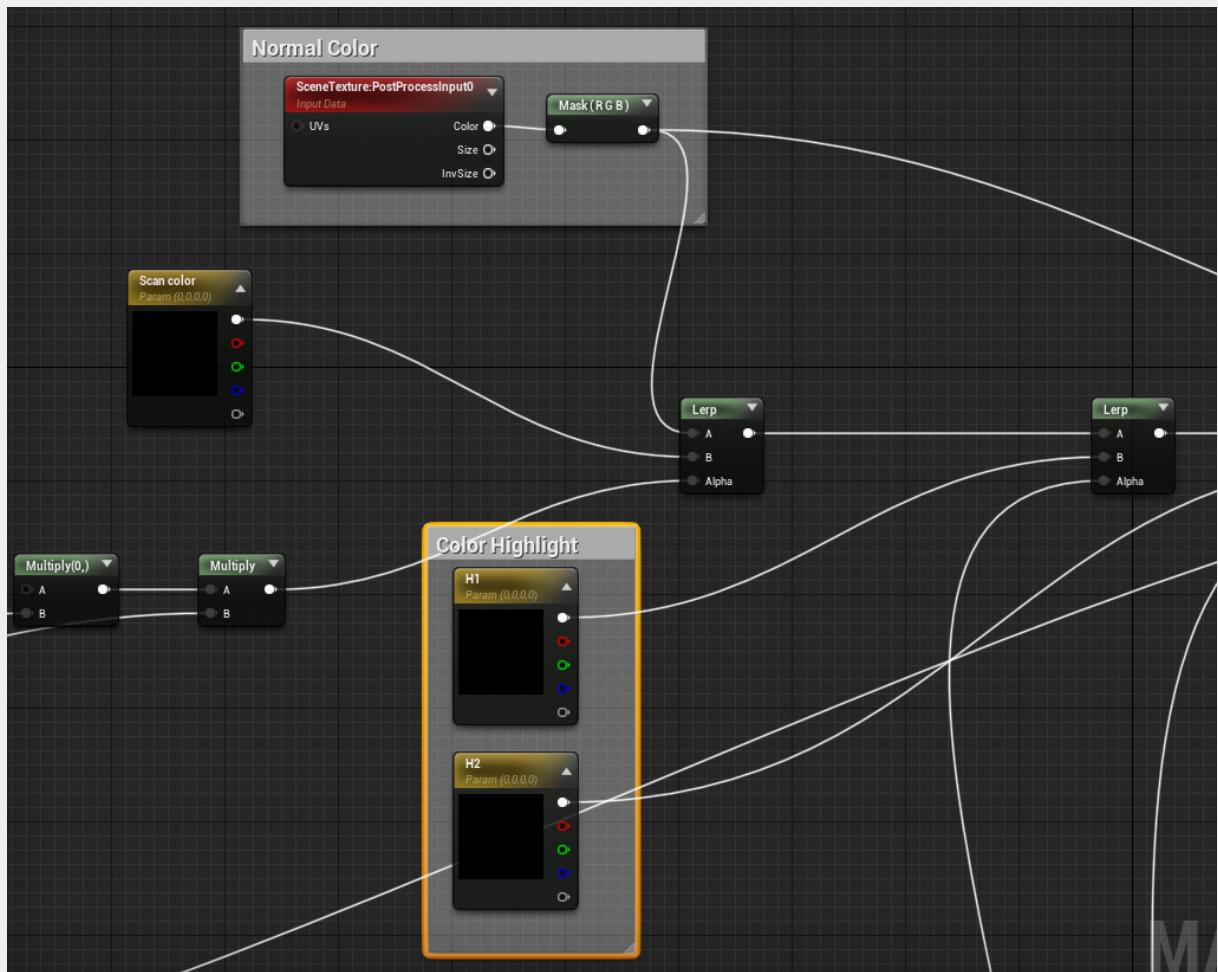
Für die Nutzerfreundlichkeit und als Hilfestellung soll eine Art Scanner eingebaut werden, welcher bestimmte Objekte für einen kurzen Zeitraum markiert und auch durch Wände anzeigen, wie in folgenden Screenshot zu sehen ist.

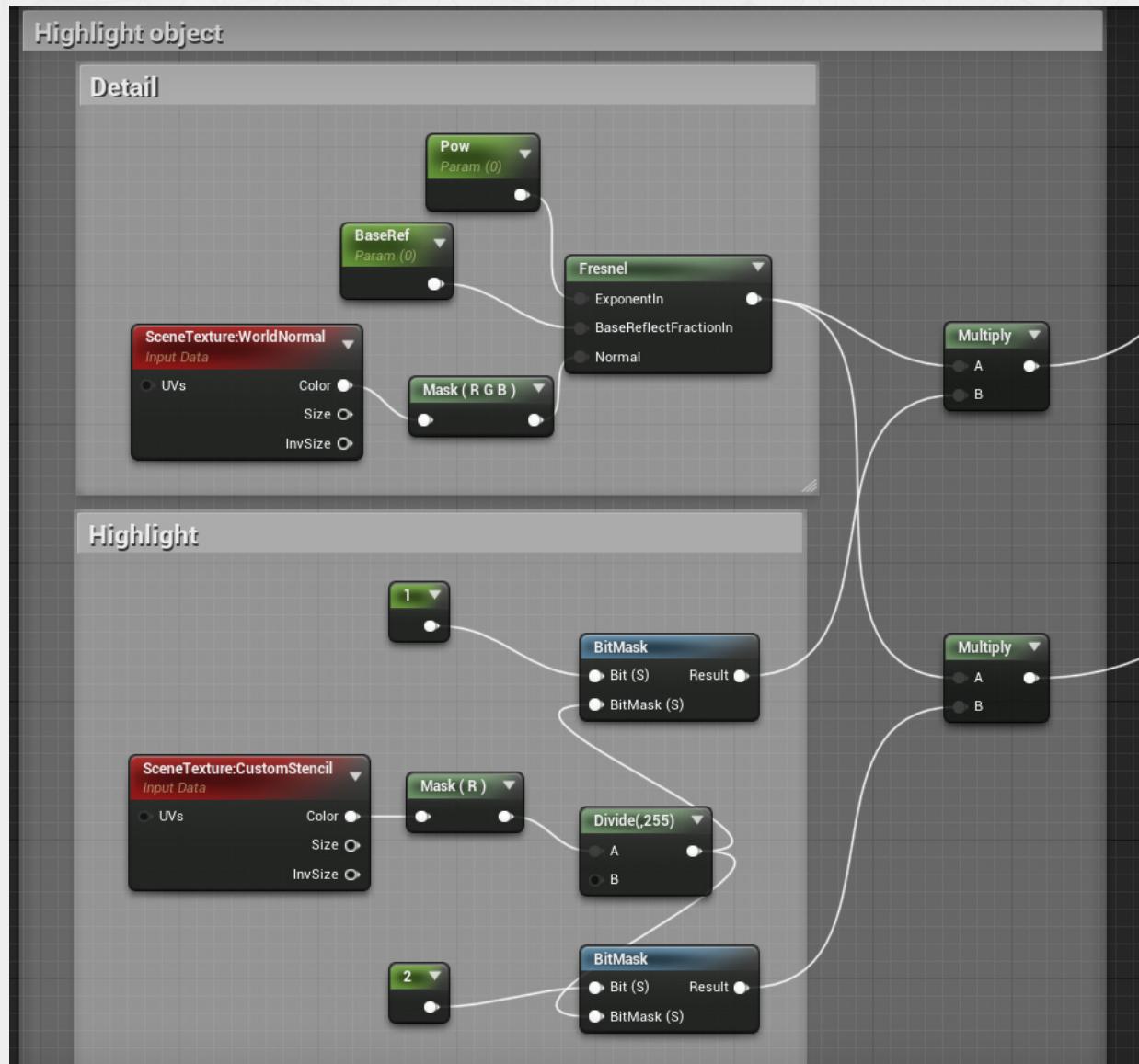


Der Scanner sendet eine Welle von der Kamera aus und markiert alles, was im Vorfeld innerhalb eines Post-Process-Volumes festgelegt wurde. Hier sind verschiedene Farben möglich, für verschiedene Objektarten bspw. theoretisch ist es auch möglich, den Impuls mit einer Textur zu belegen und so auch ein optisches Feedback zu erzeugen. Das Setup ist in den folgenden Screenshots zu sehen.

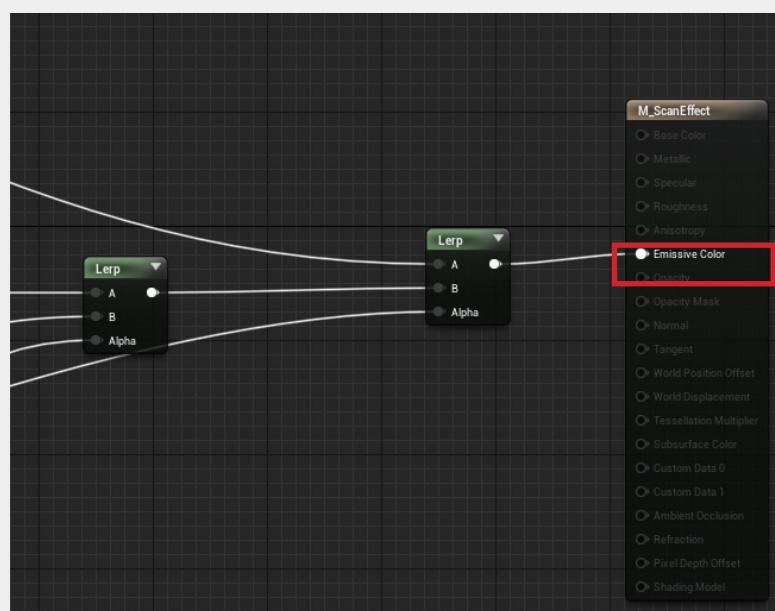


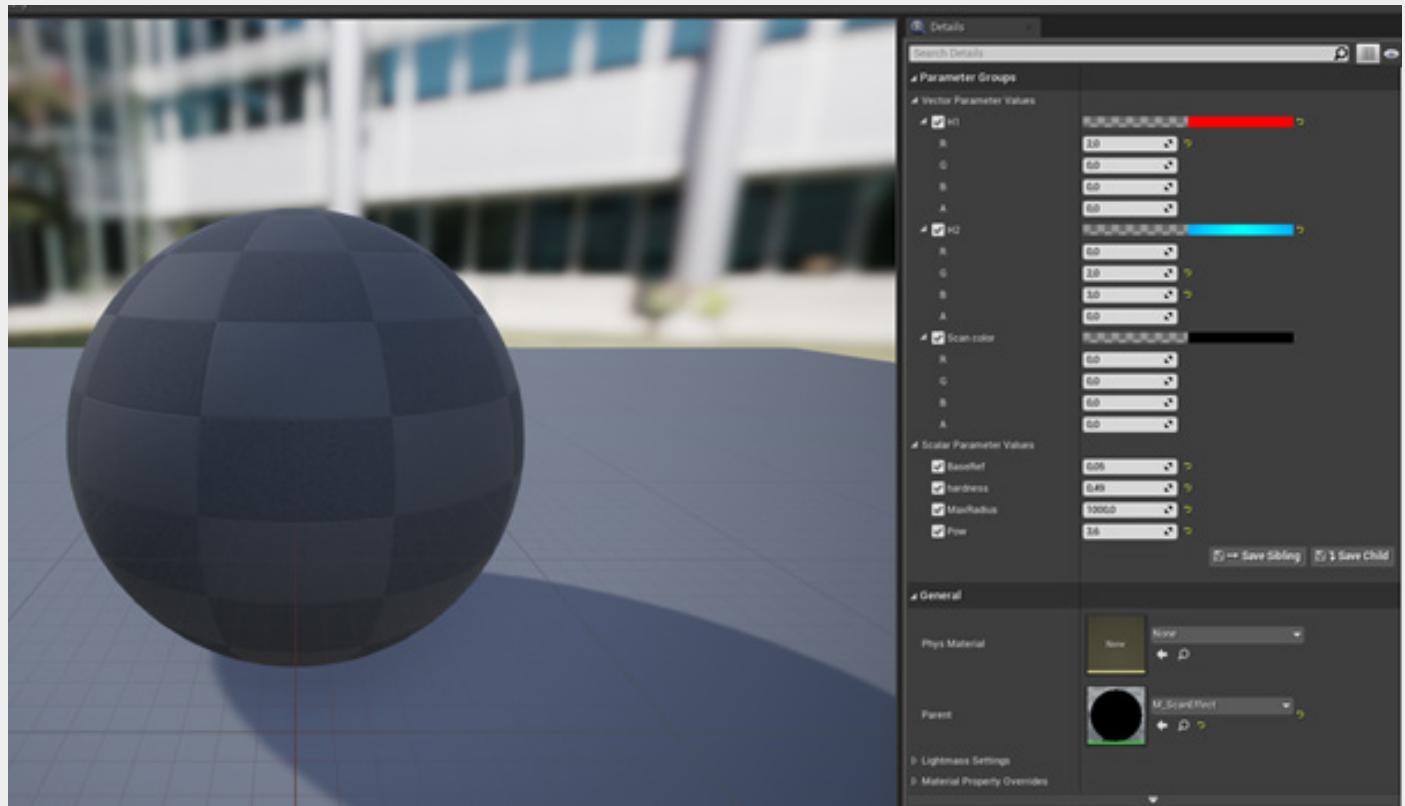
Zuerst wird hier die Sphere aufgesetzt, welche von der Kameraposition ausgeht und durch einen immer größer werdenden Radius einmal das Level durchläuft. Über diverse Multiplys und Lerps - zur Verbindung und Bündelung - werden dann die Farbbasen aufgesetzt. Sollten mehr Farben benötigt werden, muss dies hier ergänzt werden.





Der Highlighteffekt wird hier aufgesetzt und auch hier muss - falls benötigt - erweitert werden, sollten weitere Farben hinzugefügt werden. Die eigentliche Farbe wird in einer Instanz festgelegt, welche auf der folgenden Seite gezeigt wird. Das Setup allerdings erfolgt hier. Zusammen mit dem Farbssetup aus dem vorherigen Screenshot läuft alles - über Lerps - in die Emissive Color des Materials, was bedeutet, dass die Farbe einen scheinenden Effekt hat.





In dieser Instanz des Materials werden die Rahmendaten festgelegt. Es können die Farben, die Größe des Impulses (der Radius) und die „Dauer“ festgelegt werden. So kann hier bestimmt werden, dass bspw. Truhen rot und Schalter oder Knöpfe grün markiert werden - ganz nach Bedarf.

Inventarsystem

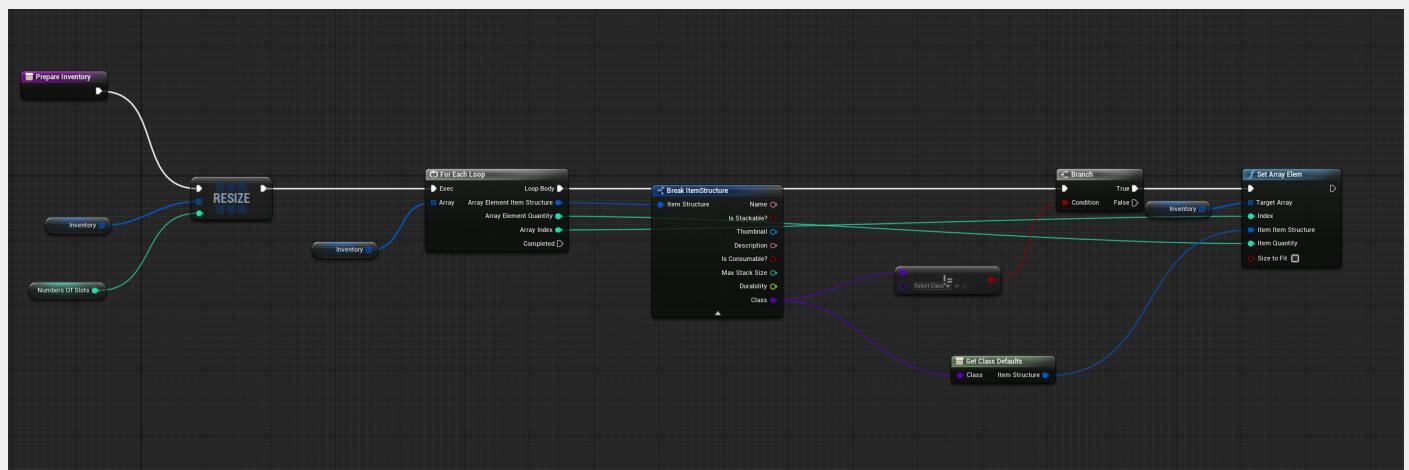
Ein funktionierendes Inventarsystem gehört zu jedem Spiel dazu, in welchem mit Gegenständen interagiert wird, die benutzt werden können oder sogar für einen Story Fortschritt von Nöten sind. Das System sollte dabei so flexibel wie möglich sein, damit es auf die verschiedenen Szenarien des Spieles einfach angewendet werden kann, ohne Sonderfunktionen oder zusätzlich mehrere Klassen zu entwickeln. Das im weiteren Verlauf des Dokuments beschriebene System wurde zwar mittlerweile durch ein besser passendes und funktionierendes ersetzt, nichtsdestotrotz ist in das System Arbeit hineingeflossen und half dabei mit der Engine vertraut zu werden.

Das Inventarsystem hatte dabei folgende grundsätzlichen Voraussetzungen bzw. Anforderungen:

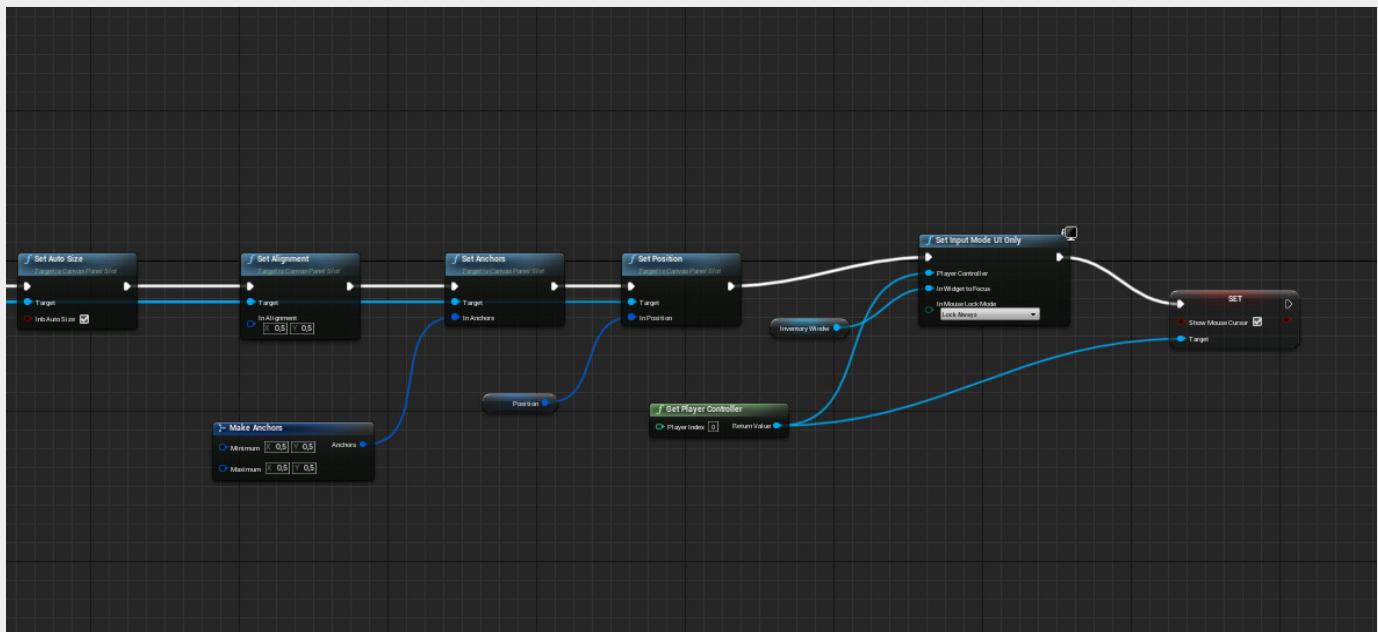
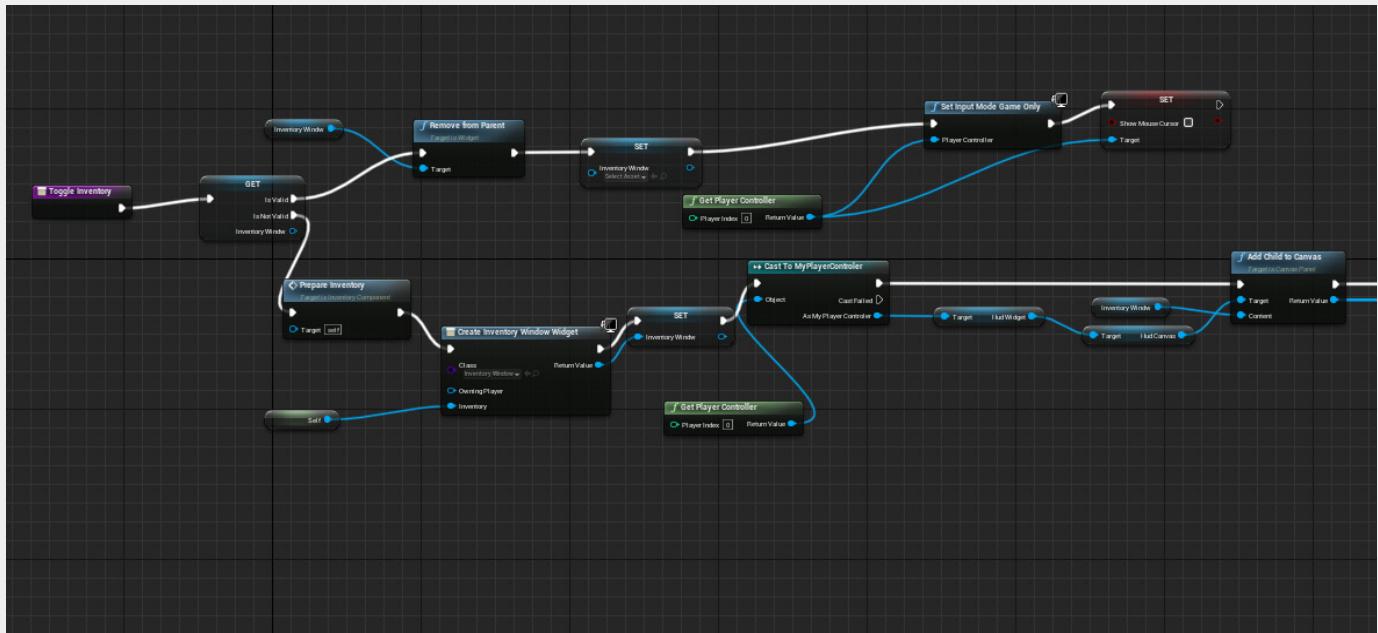
- Inventar des Spielers jederzeit aufrufbar
- Items sammeln, benutzen und abwerfen
- Items einfach zu erstellen und zu definieren
- Kisten und Container einfach erstellen und befüllen
- direkter Anschluss an folgende Quest Systeme (Überprüfung ob Item in Inventar)

Im folgenden Absatz wird versucht die Abläufe des Inventars näher zu beschreiben :

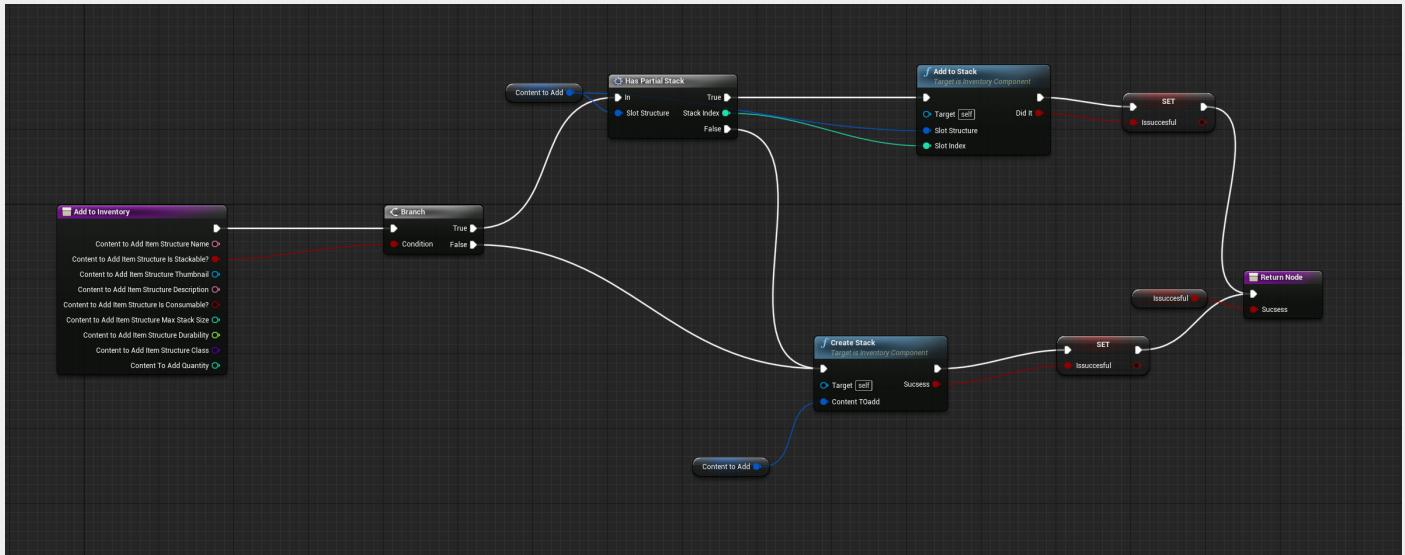
Nach jedem Start des Spieles wird das Inventar automatisch vorbereitet. Das bedeutet, das System überprüft das Save File, ob bereits Gegenstände im Inventar liegen und generiert diese erneut.



Des Weiteren wird die Grundstruktur für Items angelegt und das Inventar des Spielers generiert, welches mit leeren Slots befüllt wird. Die Größe und Startitems können dabei variabel angelegt und spezifiziert werden.

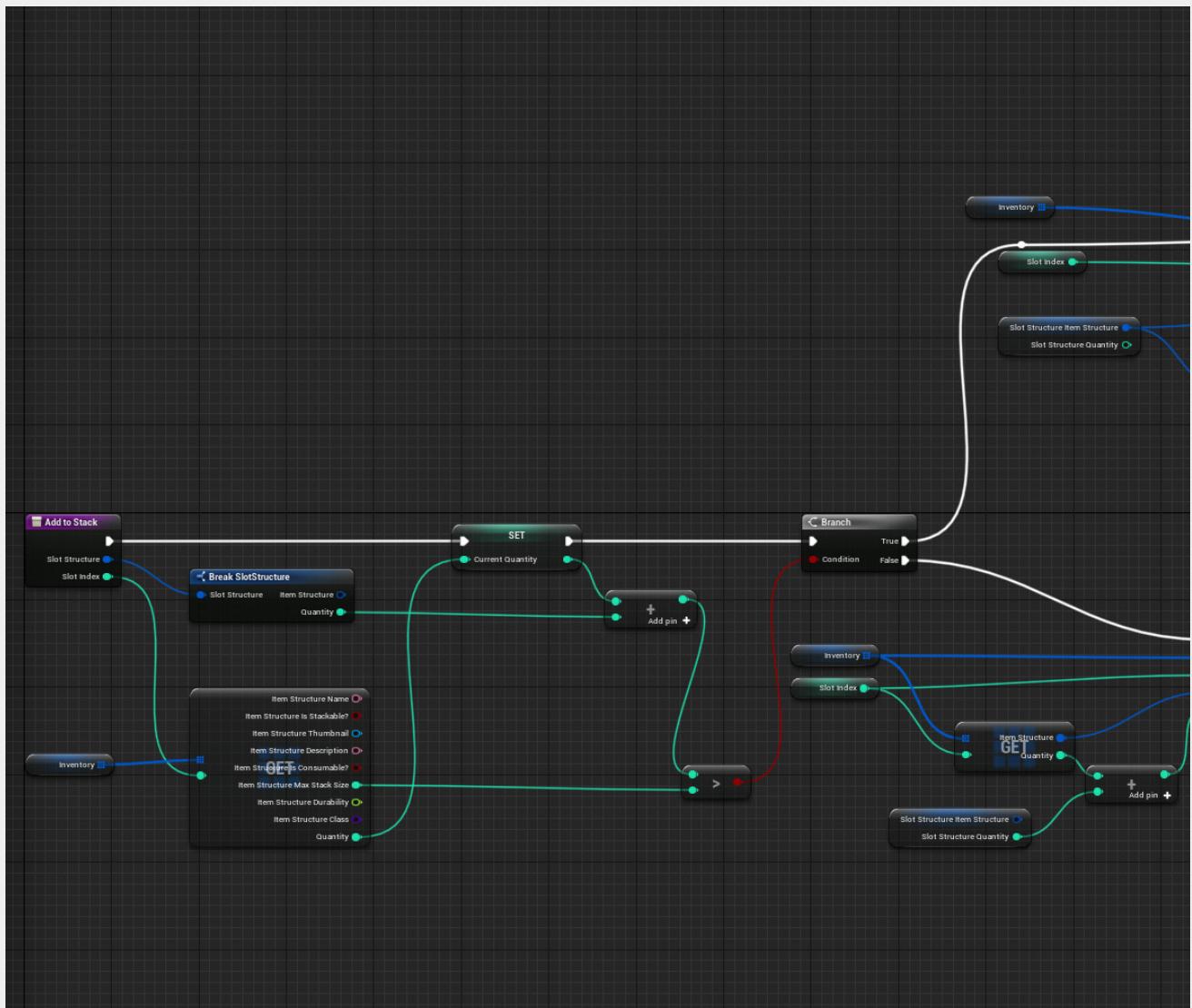


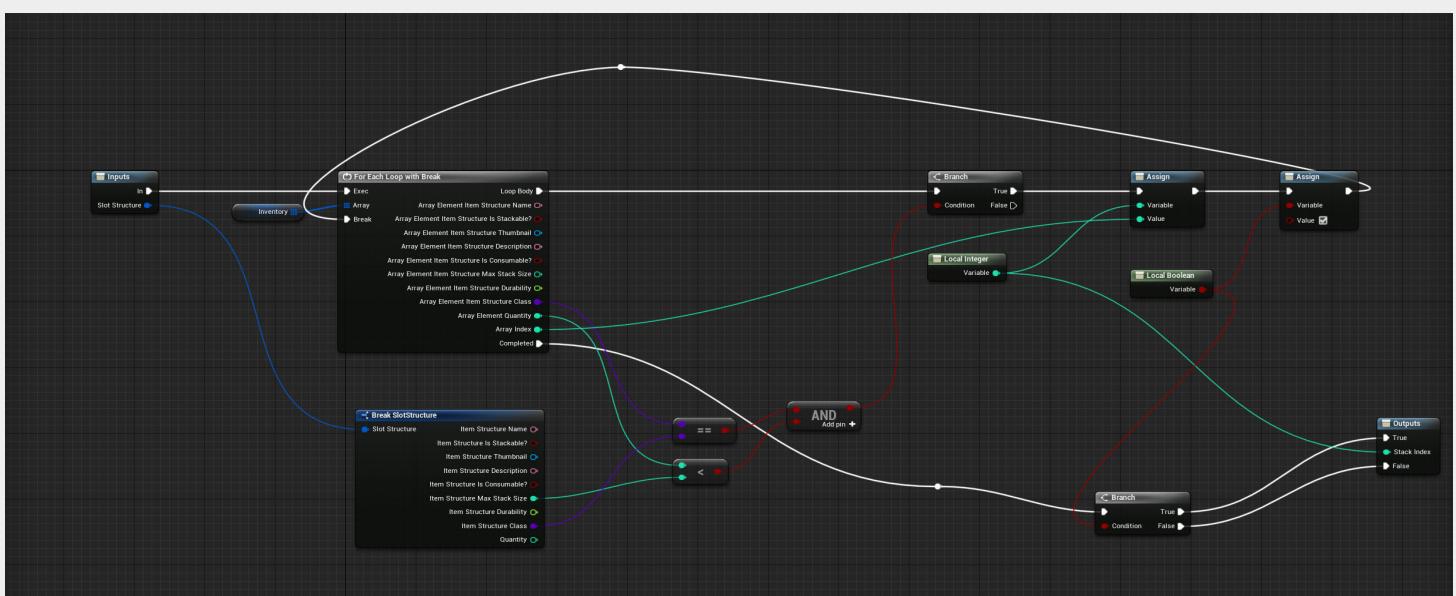
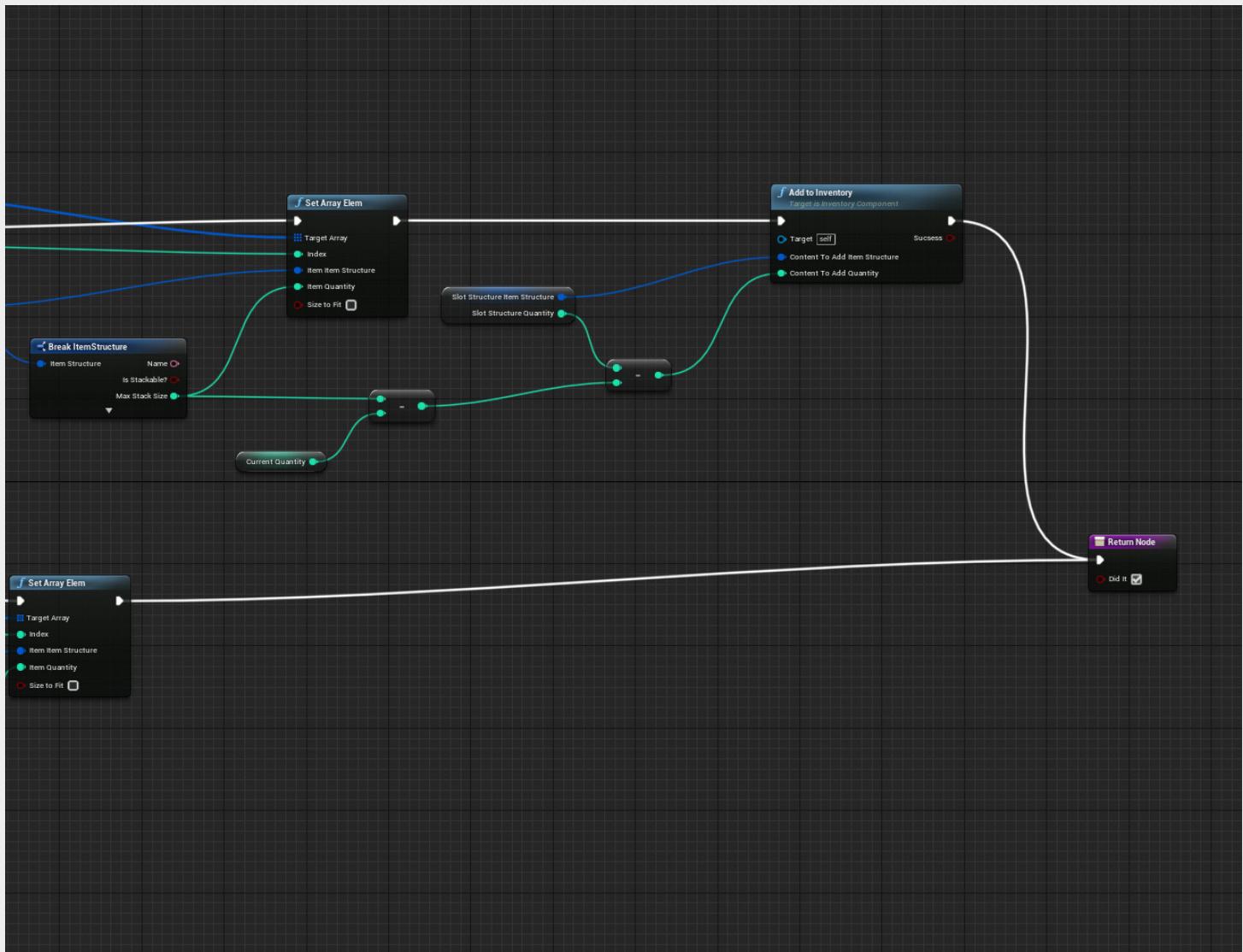
Ein weiterer Schritt, der direkt nach Starten des Spiels durchlaufen wird, ist die Sichtbarkeit des Inventars. Das Inventar wird dabei als HUD-Widget dargestellt, da dieses System optische Anpassungen sehr einfach hält. Das Widget wird generiert und gesetzt, bleibt dabei aber vorerst unsichtbar für den Spieler. Sobald dieser die für das Inventar zuständige Taste betätigt, wird die Funktion aktiviert und blendet das aktuelle Inventar des Spielers ein. Wird die Inventar-Taste erneut gedrückt, wird das HUD-Element einfach zerstört und im Hintergrund direkt neu generiert. Sobald eine Änderung im Inventar stattfindet, wird auch das Widget erneuert.



Ein weiterer wichtiger Punkt eines Inventars ist es, Items hinzuzufügen. Da dieses System mit stapelbaren Items arbeitet, muss geprüft werden, ob ein Item dieses Typs bereits im Inventar vorhanden ist sobald ein neues Element hinzugefügt wird.

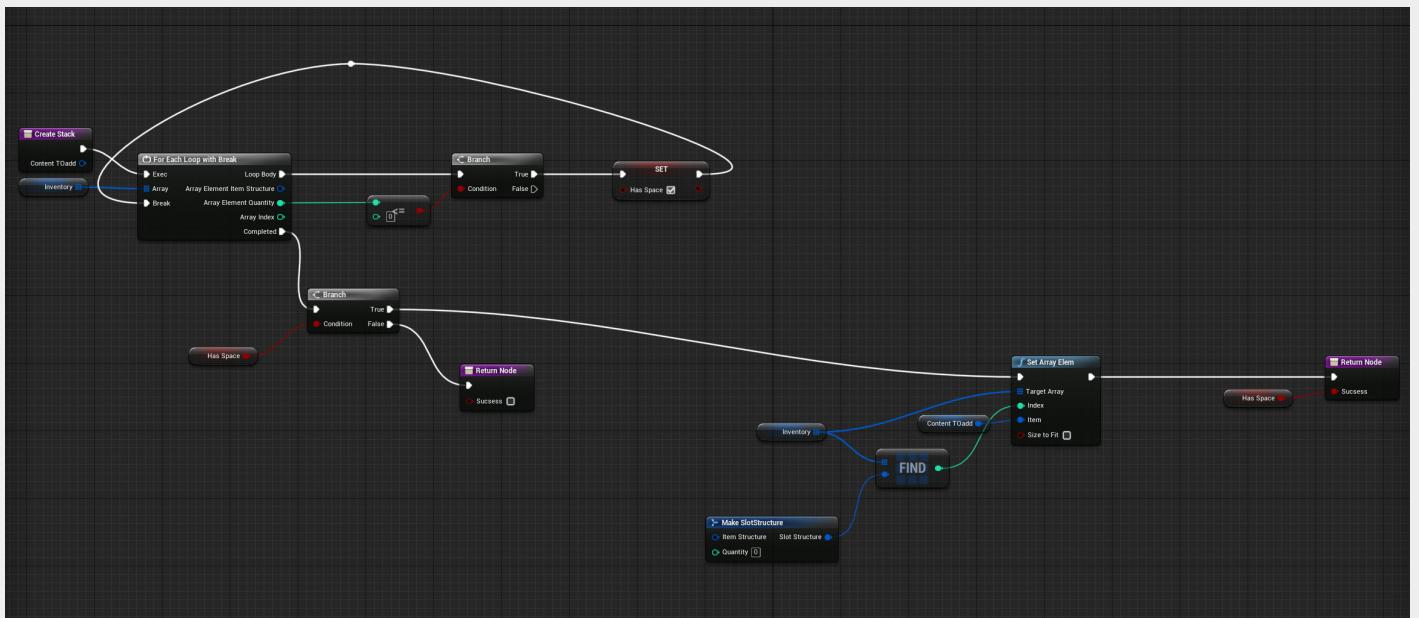
Dies erledigen die nachfolgenden Funktionen:



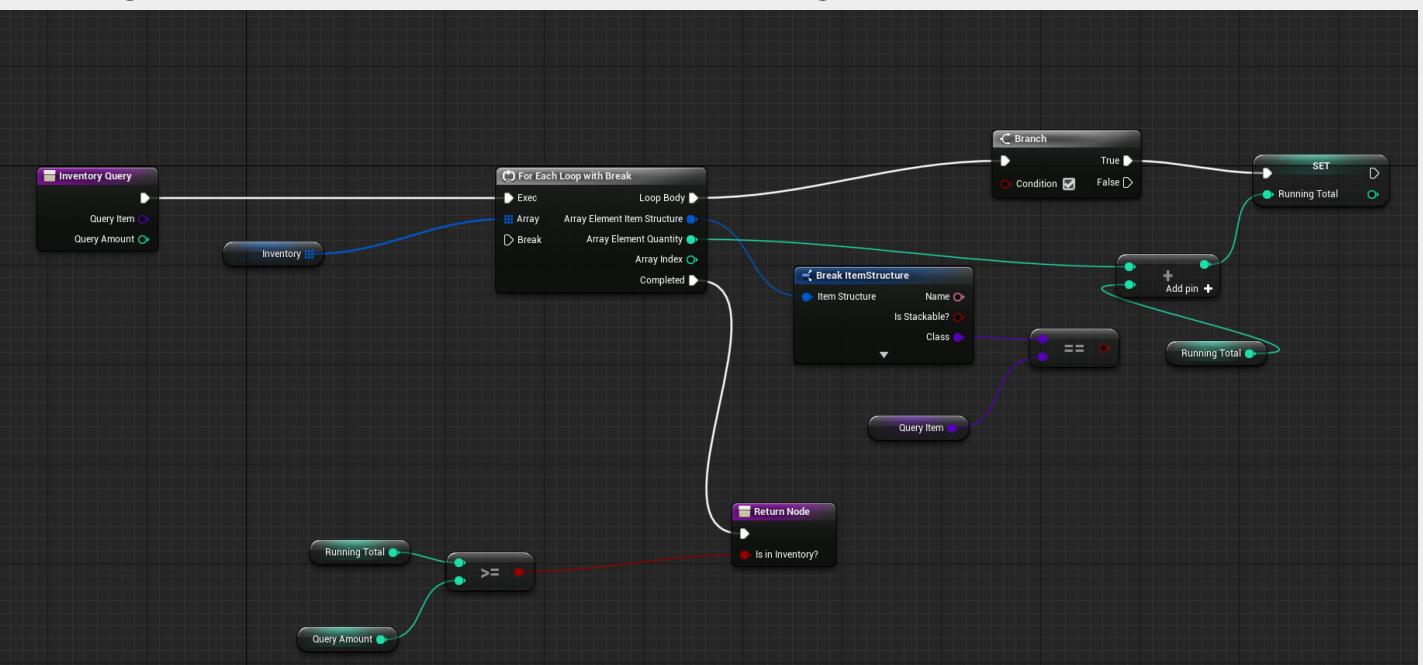


Abgebildet wird die Funktion schlussendlich durch eine Schleife über das gesamte Inventar des Spielers. Wird dabei ein Stapel des Items gefunden, welches hinzugefügt werden soll, unterbricht die Schleife und erhöht den Zähler des Stapels um eins. Das System erhält die Rückmeldung, dass das Item erfolgreich hinzugefügt wurde. Findet die Schleife keinen Stapel, wird ein neuer Stapel dieses Items erstellt und der Zähler auf eins gesetzt.

Folgende Abbildung zeigt dies:



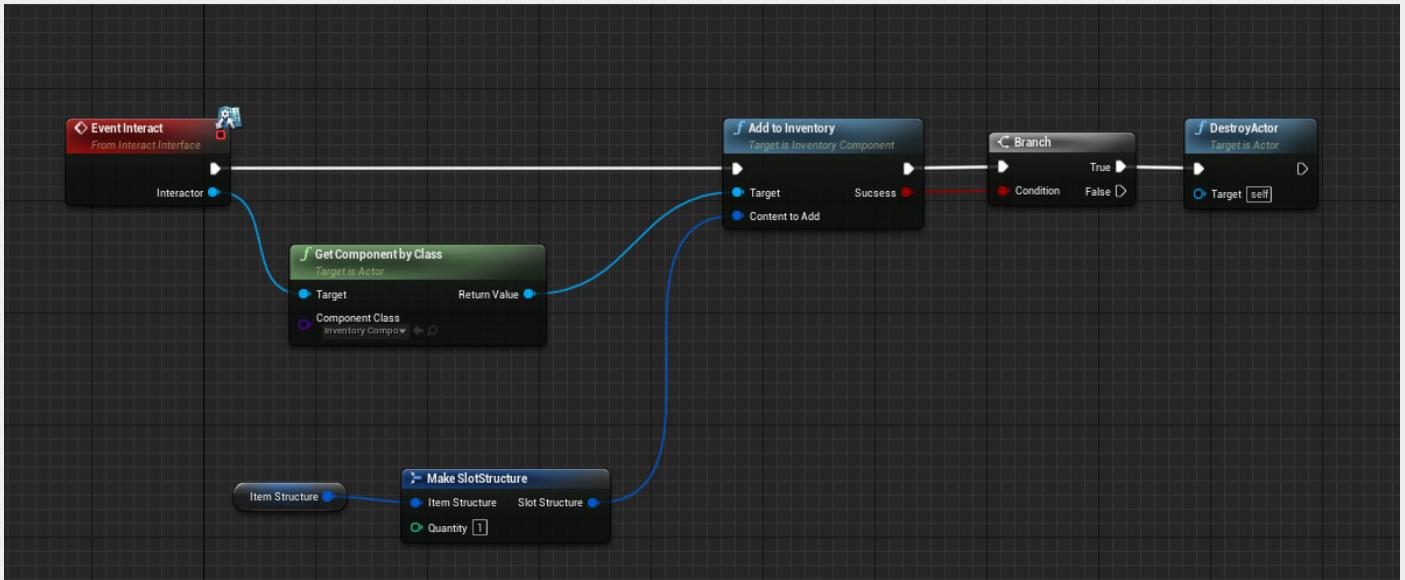
Nach erfolgreichem Durchlauf gibt es ebenfalls eine Rückmeldung an das System, dass das Item erfolgreich dem Inventar hinzugefügt ist. Schlägt die Aktion im Allgemeinen fehl, wird dies als Boolean in eine Variable geschrieben, um im weiteren Verlauf der Entwicklung diesen Fall einfach bearbeiten zu können.



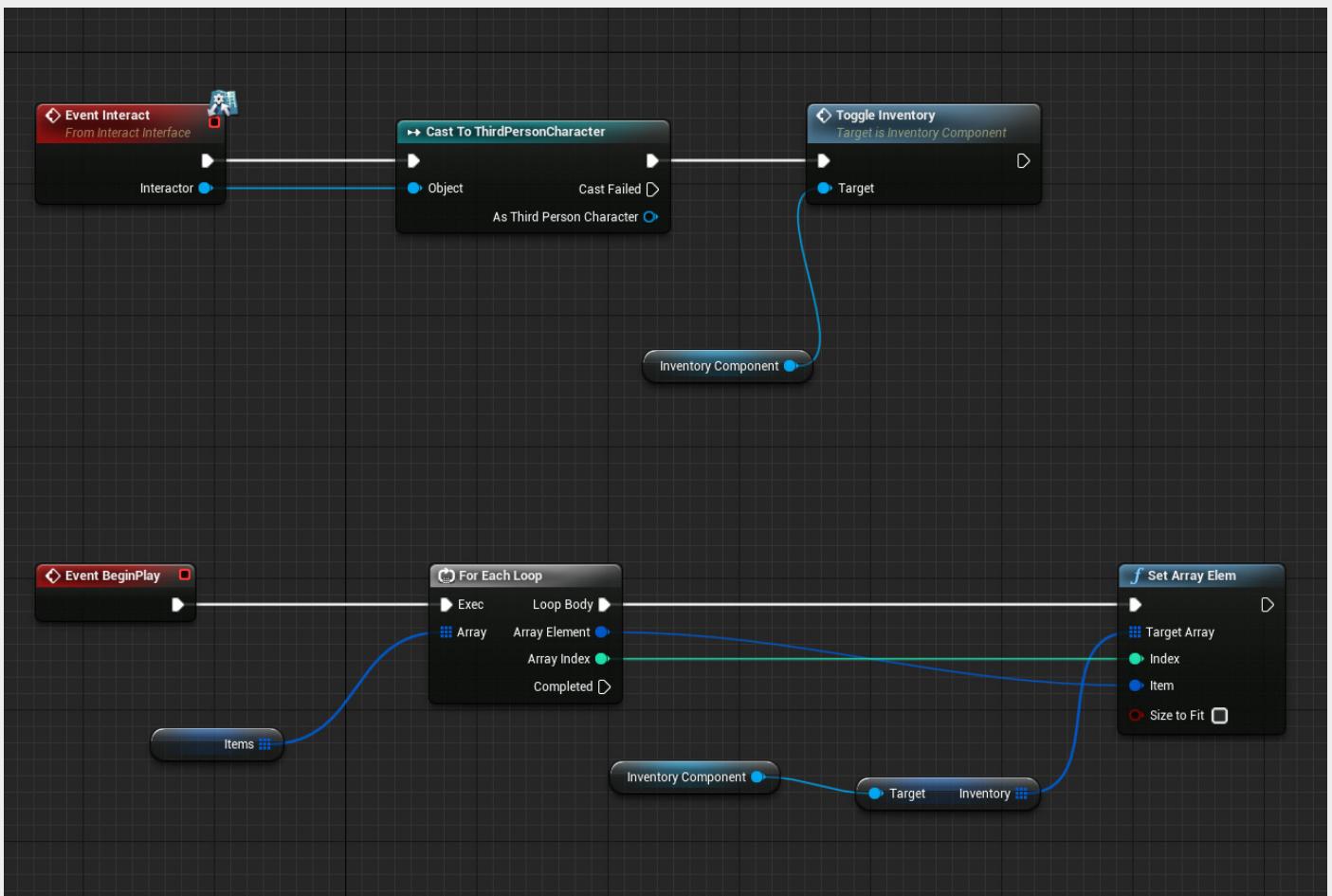
Die oben abgebildete Funktion ist eine einfache Schleife, die überprüft, ob sich eine vorher definierte Menge eines bestimmten Gegenstandes im Inventar des Spielers befindet. Diese Funktion überprüft beispielsweise, ob sich im Inventar des Spielers der richtigen Schlüssel für die Tür befindet, die er öffnen möchte. So können verschiedene Quest Systeme direkt in das Spiel integriert werden.

Zur vollen Funktionalität des Inventars fehlen die Items und Container selbst. Für beides wird eine abstrakte Klasse mit den Grundfunktionen und Attributen geschrieben, so dass für jedes Item oder Kiste nur noch Kind-Klassen erstellt werden müssen, die benötigt werden.

Als Darstellung in Blueprints:



Item



Container

Das Item hört nur auf den Befehl der Interaktion mit dem Spieler. Anderweitige Events werden ignoriert. Sobald dies geschieht, wird der Spieler vom System identifiziert und die Funktion zum Hinzufügen ins Inventar ausgeführt. Anders verhält es sich beim Container . Sobald das Spiel gestartet wird, bekommt der Container sein eigenes Inventar - wie oben beschrieben - zugewiesen. Dafür wird einmal eine Schleife ausgeführt, die jeden Slot im Container-Inventar abfragt, ob dieser bereits mit einem vordefinierten Item gefüllt ist. Ist dies der Fall, wird das Item hineingeneriert. Auch der Container reagiert auf den Interaktionsbefehl des Spielers, doch im Gegensatz zu den Gegenständen wird hierbei das Inventar des Containers neben dem Spielerinventar aufgerufen.

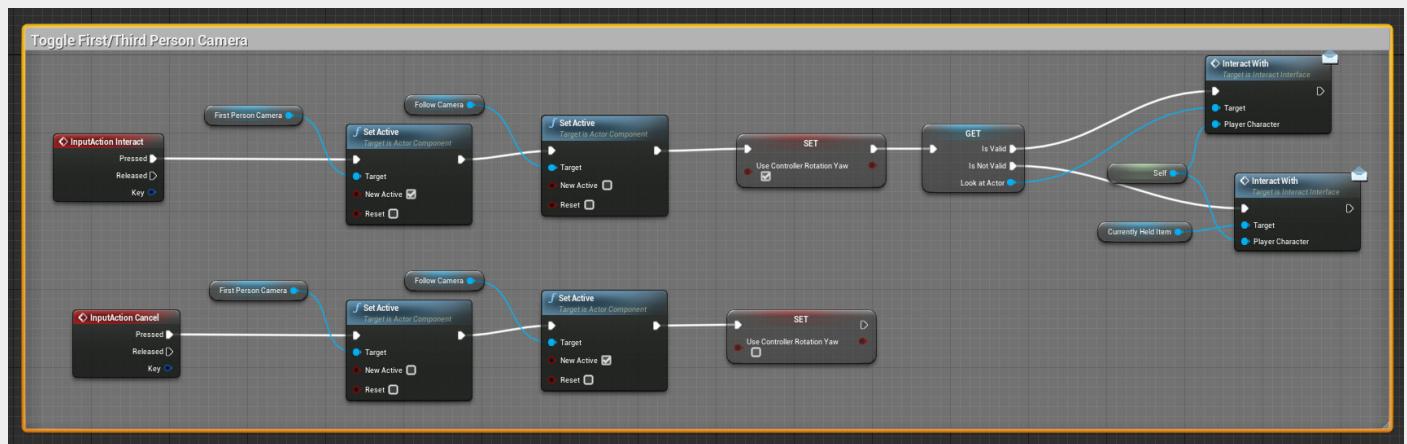
Damit wurden alle Grundanforderungen an das Inventar, Gegenstände und Container erfüllt. Der Nachteil dieses Systems jedoch ist, dass es mit dem geplanten Inspektionssystem nicht vollständig kompatibel ist . Es kam zu vielen Fehlermeldungen und teilweise ließen sich Funktionen nicht mehr ausführen. Zwischenzeitlich wurde auf ein fertiges Kombisystem aus dem Marktplatz gesetzt, da dieses in sich funktionsfähig ist und problemlos in das bestehende System eingefügt werden konnte .

Inspizieren von Gegenständen

Das Inspizieren von Gegenständen ist als Kernmechanik im Spiel geplant. Daher wurden verschiedene Ansätze entwickelt, wie ein solches System nutzerfreundlich umgesetzt werden kann. Hintergrundgedanke hierbei war immer, diese Mechanik sinnvoll in ein Third-Person-Spiel zu implementieren. Denn ein schwebender Gegenstand zwischen Spielerfigur und Kamera ist zwar möglich, allerdings optisch nicht wirklich ansprechend.

Ein Lösungsansatz war der Folgende:

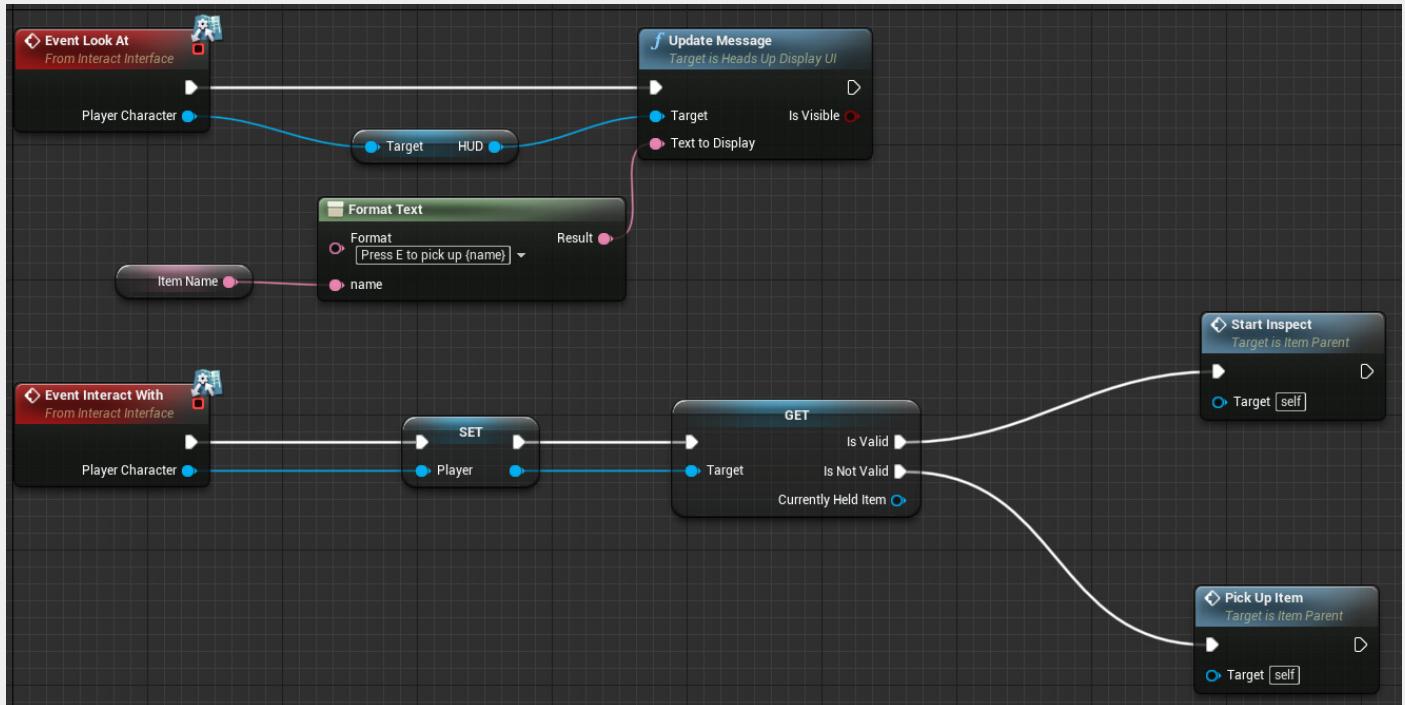
Die Kamera wechselt beim Drücken der Interagieren-Taste und im Rahmen der Collision Box des Objekts in die First-Person-Sicht, in der dann mit dem Objekt interagiert werden kann. Mit dem Verlassen des „Inspizieren-Modus“ wird dann wieder in die Third-Person-Sicht gewechselt.



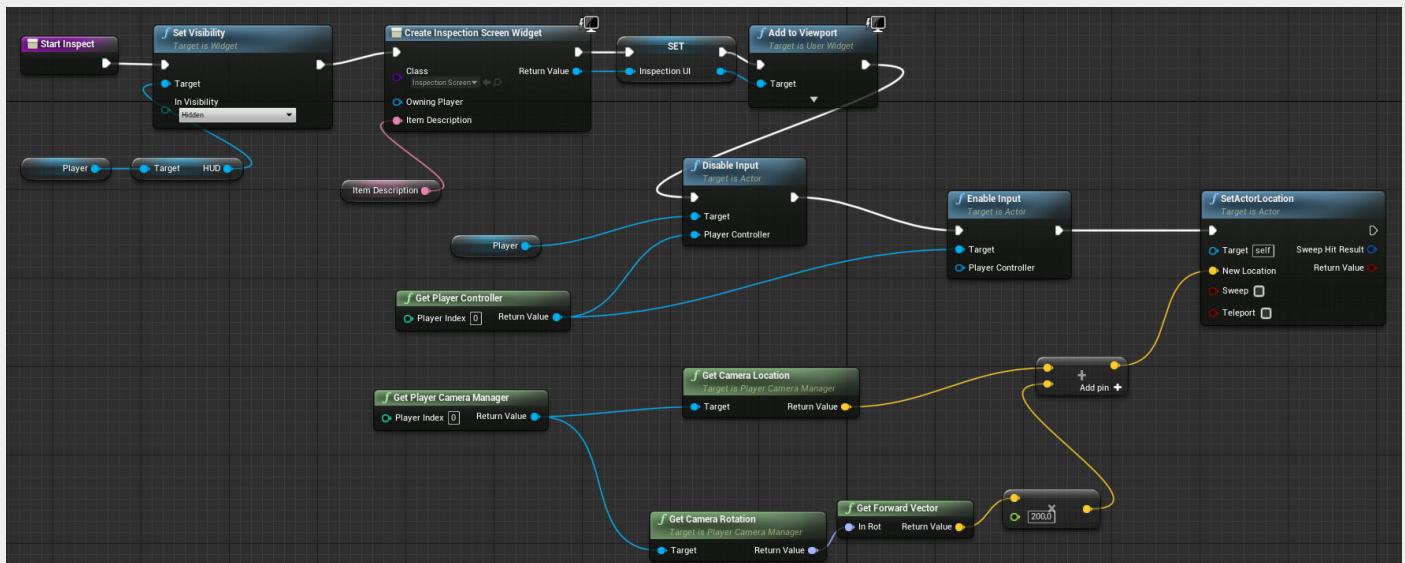
Mit diesem Code wird festgelegt, dass die Kamera beim Drücken der „Interact“-Taste (hier: E) die Kamera wechselt und als aktiv setzt. Zudem wird die Mouse/Controller Rotation für den Moment der Inspektion als aktiv gesetzt. Wird ein Item gehalten, gilt dies für das Item. Wenn nicht, dann für die Spielfigur an sich. Mit erneutem Drücken der „Interact-Taste“ wird wieder in die Third-Person-Sicht gewechselt.

Für das korrekte Aufheben von Objekten war es nötig, einen Line Trace zu führen – sprich eine von der Kamera ausgehende Linie, die überprüft, ob man auf ein Objekt blickt. Ist dies der Fall, ermöglicht es die weitere Interaktion mit diesem Objekt (durch das Aktivieren eines Interfaces innerhalb der Engine). Wenn der Line Trace fehlschlägt, ist keine Interaktion möglich.

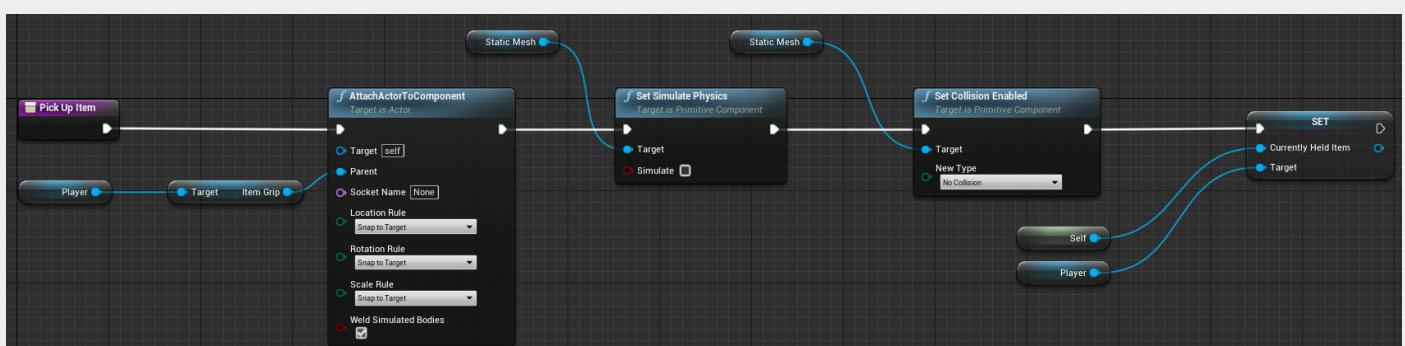
Dieses Interface trägt sämtliche Funktionen, welche innerhalb der benötigten Blueprints aufgegriffen werden. Konkret bedeutet das: wird das Interface aktiviert, ermöglicht es dem Code innerhalb des Blueprints für die tatsächliche Interaktion mit dem Objekt, auf diese Funktionen – in diesem Fall „LookAt“ und „InteractWith“ - zuzugreifen.



Innerhalb des Blueprints, der als Parent für alle Items dienen sollte, werden diese Funktionen verarbeitet und mit neuen Aktionen (hier: Start Inspect und Pick Up Item) belegt, welche dann die tatsächliche Mechanik beschreiben und aktivieren.

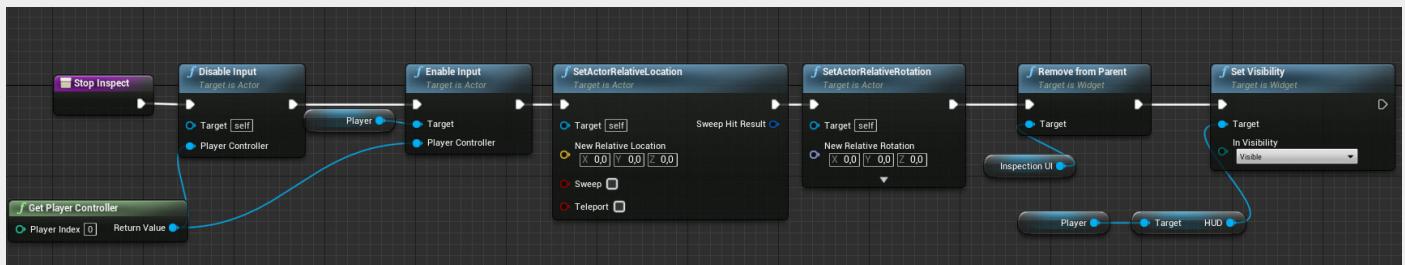


Start Inspect: Das Widget (UI-Element) für die Inspection wird aktiviert und auf den Bildschirm projiziert. Zudem wird der Actor (das Objekt) an eine festgelegte Position – näher an der Kamera für die Inspektion - gesetzt.

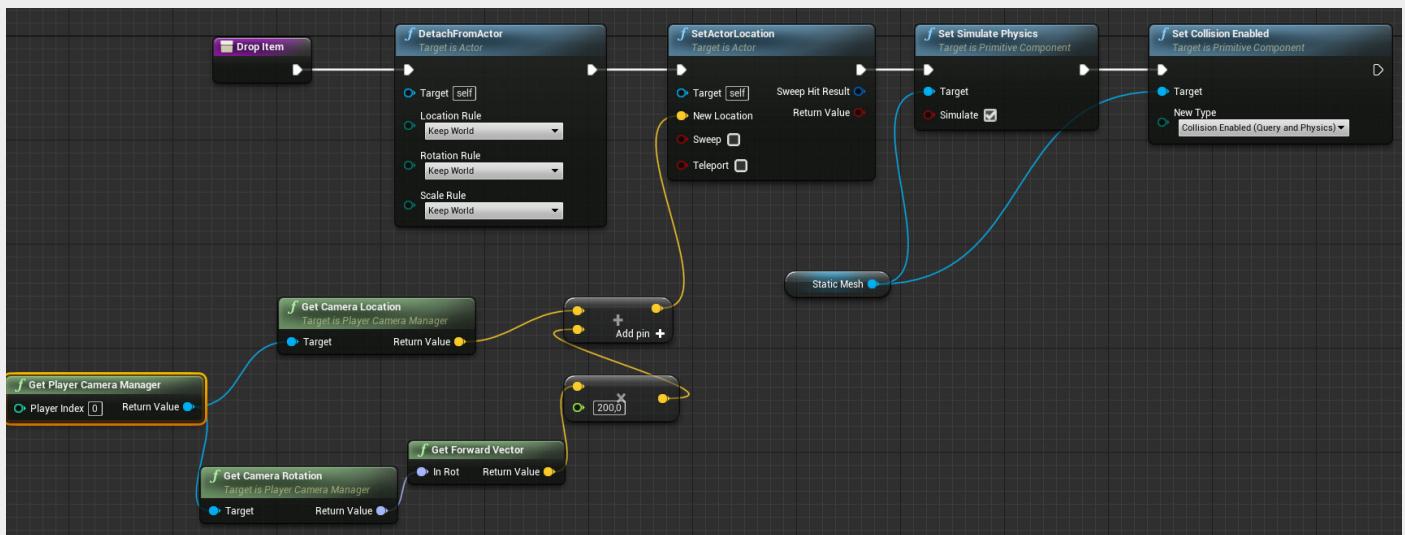


Pick Up Item: Das Objekt wird aufgenommen und an den Spieler angehängt, der damit interagiert.

Um beide Interaktionen zu beenden, wurden die hier gezeigten Funktionen wieder umgekehrt.

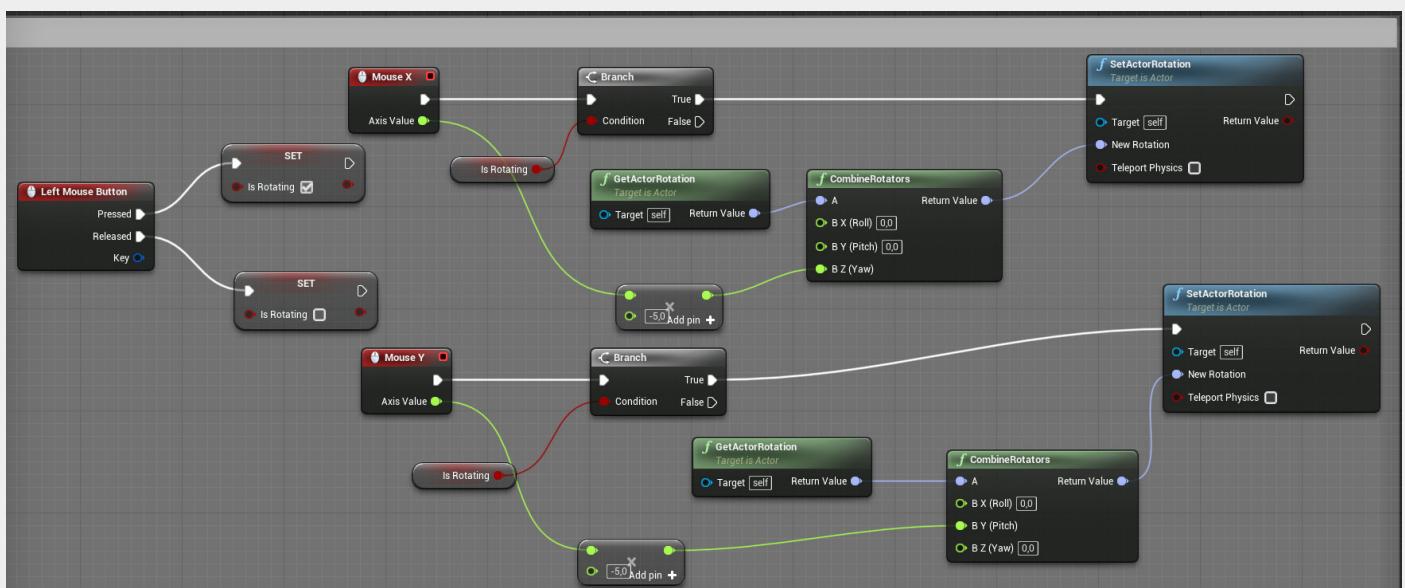


Stop Inspect: Dem Spieler wird die Kontrolle über das Item wieder entzogen und das UI wird deaktiviert.

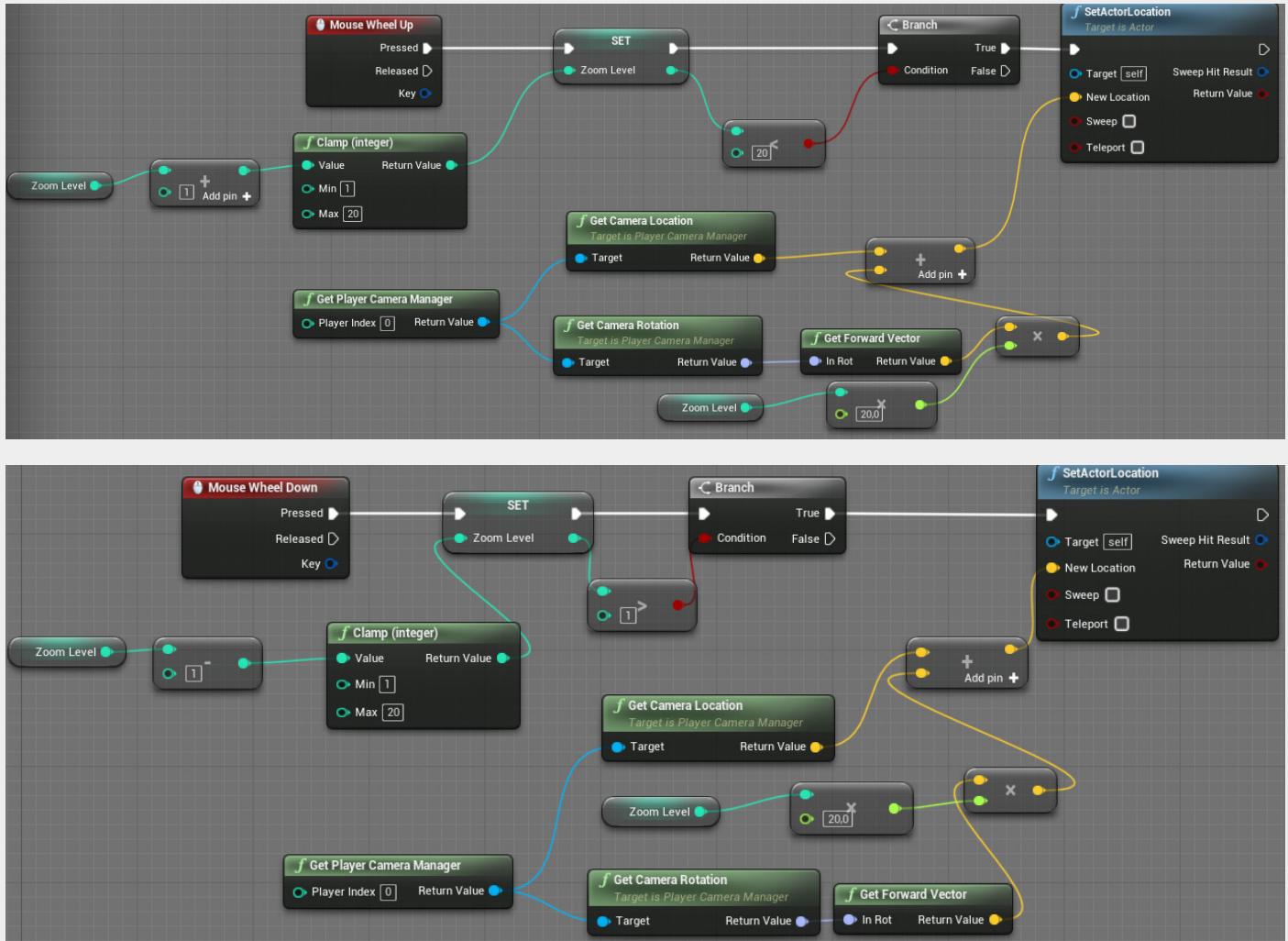


Drop Item: Das Item wird einige Längeneinheiten vor dem Spieler fallen gelassen – mit aktivierter Physiksimulation und Kollision.

Die Rotation und das Zoom auf das Objekt ist innerhalb des Eltern-Blueprints geregelt.



Rotation: Die Rotation wird anhand der X- und Y-Achse der Maus und mit gedrückter linker Maustaste festgelegt und bestimmt. Das Objekt rotiert demnach nur, wenn die linke Maustaste gedrückt ist.



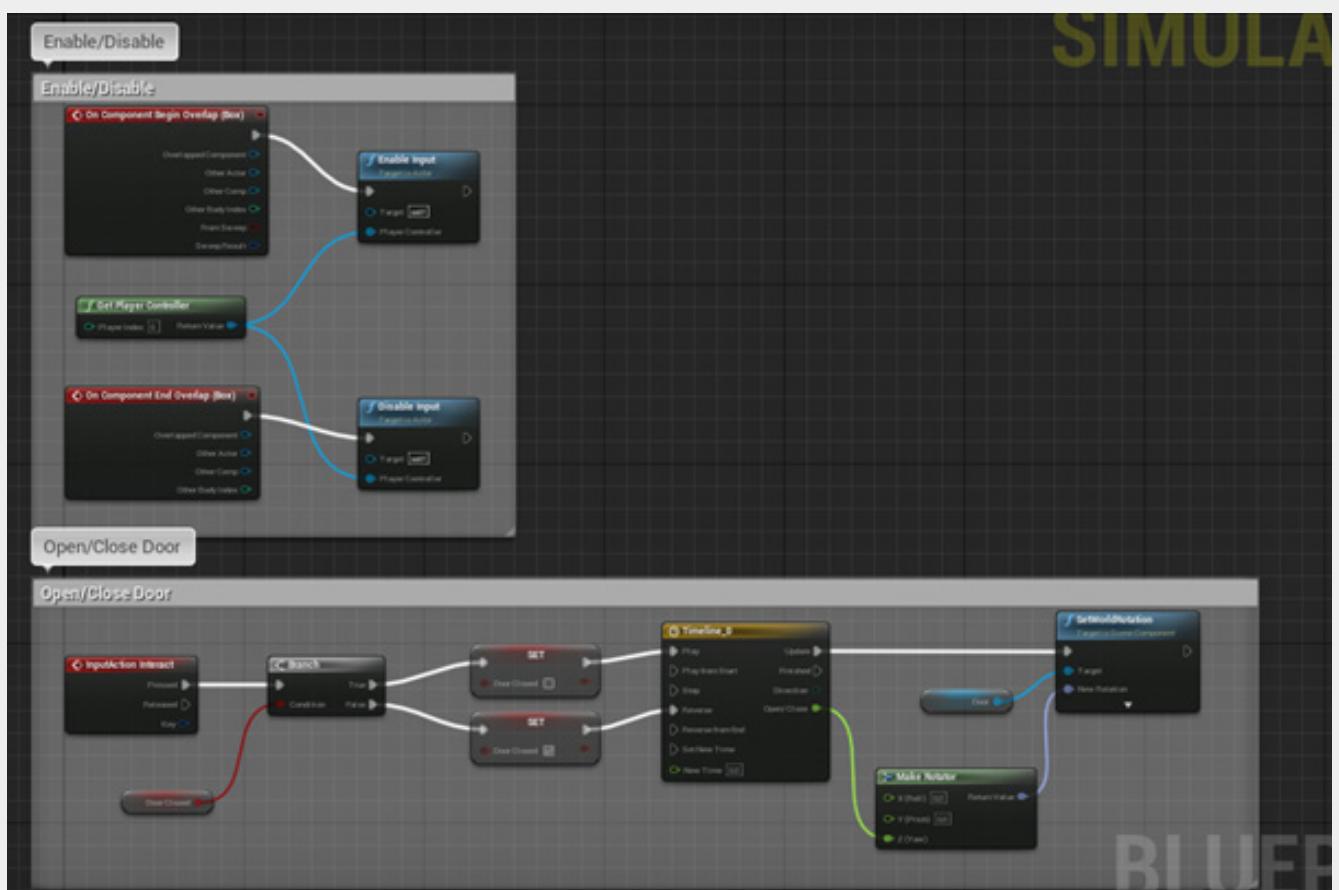
Zoom in/out: Das Zoomen auf das Objekt wird mit dem Mausrad gesteuert – nach oben entspricht dem Heranzoomen und nach unten dem Herauszoomen. Hierfür wird die Kameraposition und Rotation verwendet und anhand dessen ein Abstand berechnet, der mit dem Betätigen des Mausrads größer oder kleiner wird.

Leider gab es innerhalb dieses ganzen Konstruks einige Probleme.

Zum einen hat der Kamerawechsel zwar funktioniert, jedoch alle anderen Mechaniken deaktiviert. Zum anderen konnte der Line Trace, aufgrund der zwei aktiven Kameras, nicht ordentlich berechnet werden. Dies führte dazu, dass sämtliche Mechaniken gar nicht erst angewendet werden konnten. Außerdem war eine Kombination mit dem Inventarsystem bis zu diesem Zeitpunkt nicht funktionsfähig umsetzbar.

Öffnen und Schließen von Türen

Eine simple, aber wichtige Funktion ist das Öffnen und Schließen von Türen. Mit dem Drücken der Interagieren-Taste öffnet sich die Tür, wenn der Spieler sich innerhalb der Box Collision befindet. Mit erneutem Drücken der Taste schließt sich die Tür wieder.



Der Code für das Öffnen und Schließen der Türe.

3D

3D Ergebnisse

Für die Gestaltung des ersten Levels wird eine umfangreiche Auswahl von 3D Assets benötigt. Einige Objekte können durch bereits bestehende Modelle aus Asset-Bibliotheken bereitgestellt werden. Andere Objekte, die wiederum einen spezifischen Zweck für den Spielablauf oder das gewünschte Aussehen erfüllen sollen, müssen im Rahmen der Projektarbeit erstellt werden.

Kirche

Das zentrale und aufwendigste Modell des ersten Levels ist die Kirche. In ihr beginnt der Spieler sein Abenteuer und wird dort voraussichtlich eine gewisse Spielzeit verbringen. Das Asset soll zudem eine Anzahl von Anforderungen erfüllen:

Da die Kirche sowohl von innen als auch außen zu sehen sein wird, mussten beide Bereiche modelliert werden. Dies in zwei separate Modelle aufzuteilen ist hierbei keine Option gewesen, da es aufgrund des Umfangs einen zu großen Ladeaufwand in der Engine darstellen würde.

Für die Generierung korrekter Lightmaps war es für ein Modell dieser Größe notwendig es in mehrere Meshes zu unterteilen, welche eigene Lightmap UVs erhalten haben.

Wände und Böden sollen zudem die Möglichkeit für Vertex-Painting in der Engine bieten, wofür eine gewisse Anzahl von Geometrie vorhanden sein muss, um in der Lage zu sein glaubwürdige Übergänge zwischen Materialien zu erzeugen.

Kirchenorgel

Die Kirchenorgel ist eines der größten Modelle, welches für die Inneneinrichtung der Kirche benötigt wird. Zu ihrem Umfang gehören der eigentliche Orgelkörper, die Orgelpfeifen, das zentrale Element, welches die Tasten zum Spielen der Orgel besitzt, sowie ein dazu passender Hocker. Das hier vorgestellte Modell ist bereits die dritte modellierte Version. Nach einer grundlegenden Änderung der Silhouette, beinhaltet es nun auch an die Umgebung und andere Modelle angepasste Deko-Elemente in Form von Ornamenten. Mit der Orgel wird im Verlauf des Levels innerhalb der Kirche mindestens einmal interagiert. Im geplanten Rätselablauf ist sie ein zentrales Objekt, welches richtig bedient werden muss, um weiter voranzuschreiten.









USER INTERFACE

User Interface - Ergebnisse

Entwicklung eines UI-Elements

a. Grundlage Vorüberlegung

Um den Spieler nicht vom Gesamteindruck der Szenen abzuhalten, sollen UI-Elemente nicht dauerhaft angezeigt werden. Dies verhindert auch unnötige Ablenkungen von Abläufen, auf die der eigentliche Fokus zu einem bestimmten Zeitpunkt gerichtet sein soll. Das UI-Element soll wirklich erst dann auftauchen, wenn es relevant für eine mögliche Interaktion wird, die der Spieler ausführen kann.

b. Grundlage Lösung

Technischen Ansatz zur Lösung der Vorüberlegung ist es, die Sichtbarkeit über das Betreten und Verlassen einer Kollisionsbox als Auslöser zu steuern. Beim Annähern an ein Objekt, welche eine Interaktionsmöglichkeit besitzt, gelangt die Spielfigur in einen definierten Bereich um das Objekt selbst. Das Betreten und Aufhalten innerhalb dieser Zone (Kollisionsbox) aktiviert das Anzeigen des vorher bestimmten UI-Elements. Wird die Zone anschließend wieder verlassen, so wird die Anzeige folglich ebenso deaktiviert.



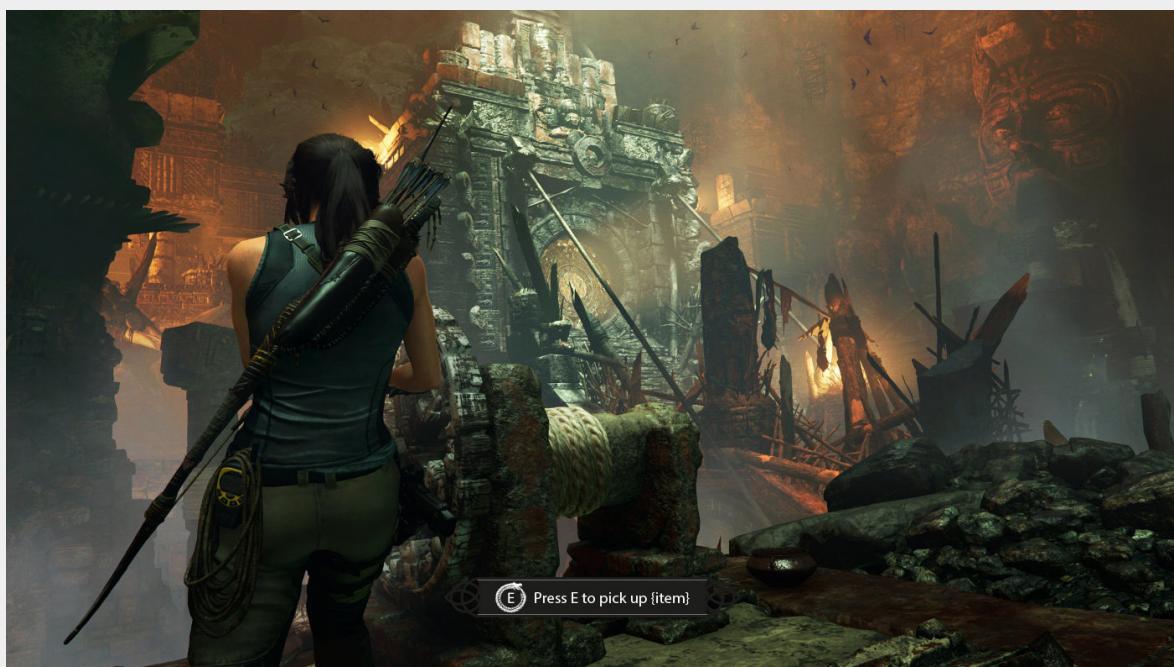
Weiterentwicklung der Idee UI-Element

a. Design

Das Design der im Spiel vorhanden UI-Elemente soll der Überlegung nach in einem passenden Stil entworfen werden. Zur Thematik der Geschichte gehören Merkmale aus dem mitteleuropäischen Raum, sowie Elemente aus der nordischen Mythologie. Dementsprechend muss dies bei der Auswahl der Symbolik und der Farbgebung beachtet werden. Ebenso wichtig ist die konsequente Bestimmung einer durchgehend einheitlichen Schriftart, sowie deren Farbe und Größe innerhalb der Elemente. Ein Faktor, der nicht außer Acht zu lassen ist, ist die gute Sicht- und Lesbarkeit für den Spieler.

b. Beispielfindung

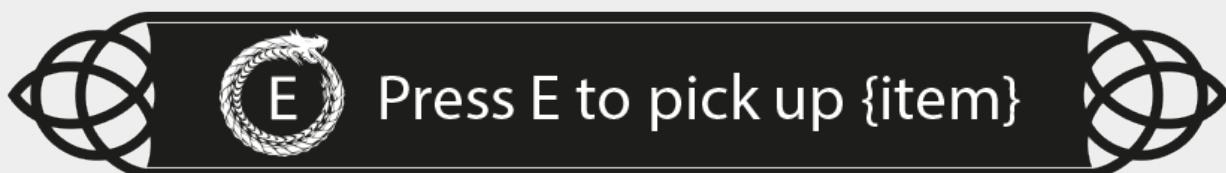
Beispiele für passend konzipierte und gut integrierte UI-Elemente sind in vielen Titeln zu finden, welche ähnliche Genres und Grundstimmungen mit unserem Projekt teilen. Eine Suche nach Bildern führt daher schnell zum Erfolg. Nach Auswahl einer passenden Designvorlage und dem Vergleich mit dem bisher Entworfenen, fallen Punkte auf, die positiv verbessert werden können und sollten.



Quelle: <https://www.playstationlifestyle.net/2018/09/10/shadow-of-the-tomb-raider-launch-trailer/#/slide/1>

c. Erstes Ergebnis

Im ersten Zwischenschritt wurde auf einen hohen Kontrast zwischen hell und dunkel gesetzt, um für eine besonders gute Lesbarkeit für den Spieler zu sorgen. Des Weiteren wurde die für das Design gewählte Symbolik zum Element hinzugefügt. Für den eigentlichen Text innerhalb des Elements wird eine vorher bestimmte, einfache und verständliche Interaktion für den Spieler aufgezeigt.



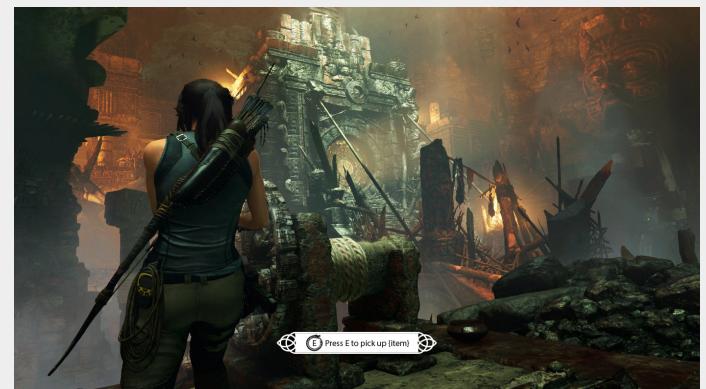
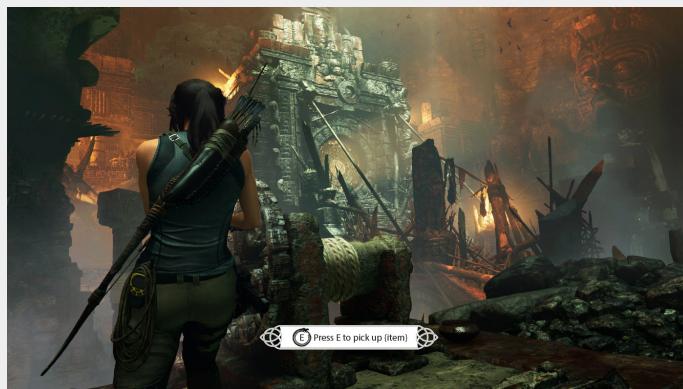
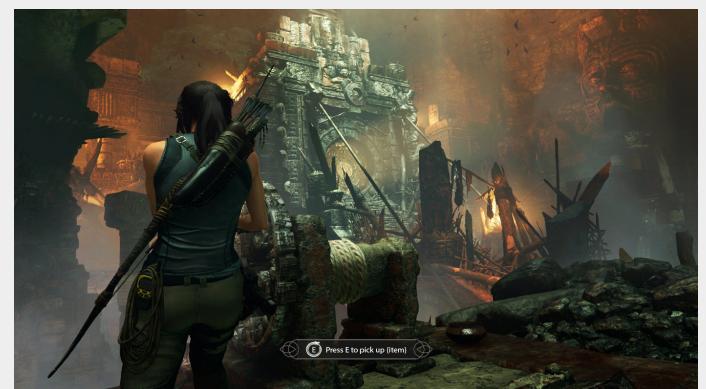
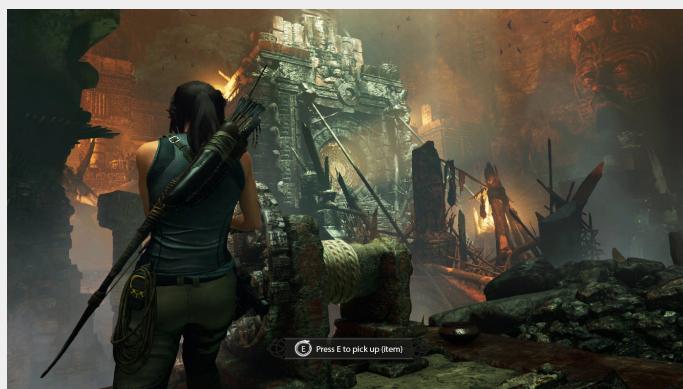
d. Weiterentwicklung

Nach Tests innerhalb einer Spielszene und eines Beispielbilds wurde entschieden, dass das UI-Element für ein optimales Ergebnis noch weiter verbessert werden müsse. Der Kontrast sollte erhalten bleiben, die Ornamente in dunkler Umgebung jedoch deutlich besser erkennbar sein. Eine Herangehensweise zur Verbesserung des ursprünglichen Designs in diese Richtung wurde beschlossen und umgesetzt. Eine Auflockerung und bessere Sichtbarkeit der Ornamente entsteht hier durch das Hinzufügen von dünnen weißen Linien, die direkt über die dunklen Originale darunter gelegt werden. Ebenso kommt ein weißer Rahmen in verschiedenen Abstufungen hinzu.



e. Test des UI Elements auf Spielhintergrund

Das überarbeitete Design des UI-Elements wurde erneut auf dem bisher als Referenz verwendeten Beispielbild eingefügt. Abbildung 1 und 2 besitzen als Grundlage das Element mit dunkler Basis und hellen Inhalten. Die Abbildung 1 arbeitet im Vergleich zu Abbildung zwei zusätzlich mit den weißen Zusatzlinien auf den äußeren Ornamenten. Hingegen besitzt Abbildung 2 als Kontrast nur den weißen Rahmen innerhalb. Die beiden unteren Abbildungen 3 und 4 zeigen das UI-Element mit invertierten Farben. Die Basis ist nun hell und die Inhalte viel dunkler gehalten.



f. Ergebnisse zum Empfinden in der Gruppe

Nach Auswertung und Rücksprache innerhalb der Gruppe wurde relativ eindeutig gegen die Verwendung des Elements mit der hellen/weißen Basis entschieden. Argumente gegen diese Basis: mit einer hellen Basis wirkt das Element innerhalb der Szene viel zu grell, da die Umgebung sonst überwiegend düster gehalten ist. Es erschien unpassend den Spieler durch ein solches Design aus der Stimmung zu reißen. In der hellen Färbung wurde das Element auch eher als unnötig störend beschrieben. Ein Test nach den vorgenommenen Änderungen erfolgt in einer Unreal Szene.

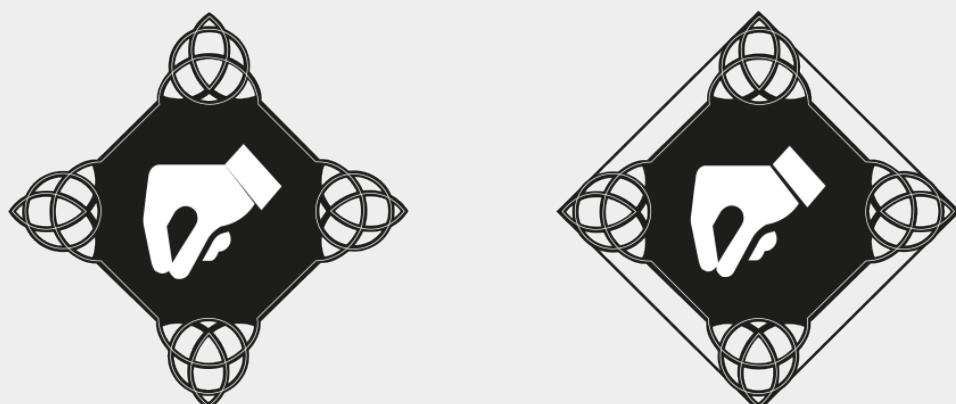


Ausarbeitung von 3D Widgets

a. Überlegung

Es soll eine optische Möglichkeit geschaffen werden, wie dem Spieler ohne Worte mitgeteilt werden kann, mit welchem Objekt eine Interaktion stattfinden kann. Wie bereits zuvor im Abschnitt des UI-Elements beschrieben, mussten auch hier Überlegungen und Entscheidungen für das Design getroffen werden. Dieses Design soll sich nahtlos und passend zu anderen Elementen integrieren lassen. Dafür spielen auch hier Farbe, Kontrast, Highlights und Symbolik eine Rolle.

Die ersten Vorschläge orientieren sich sehr stark am UI-Element. Ersterer Vorschlag besitzt die bekannten Ornamente mit weißen Hilfslinien, sowie den Rahmen innerhalb der Box. Des Weiteren ist ein Symbol vorhanden, welches die Interaktion stilistisch und ohne Worte beschreibt. Der zweite Vorschlag wird durch einen zusätzlichen Rahmen ergänzt, welcher die Form zu einem Quadrat verändert.



b. Erster Test innerhalb einer Unreal Szene

Die erste Implementierung von 3D-Widgets wurde als Prototyp mit Platzhaltern für Objekte, welche eine Interaktionsmöglichkeit besitzen und den ersten beiden zuvor vorgestellten Designversuchen innerhalb einer Unreal Szene getestet.

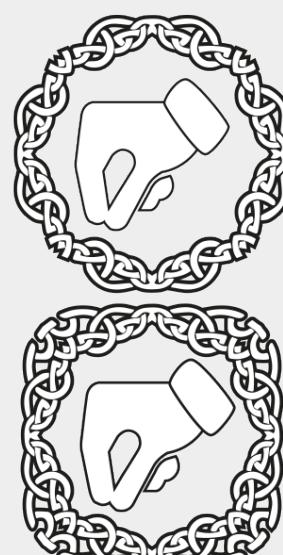


Designänderung und aktueller Stand

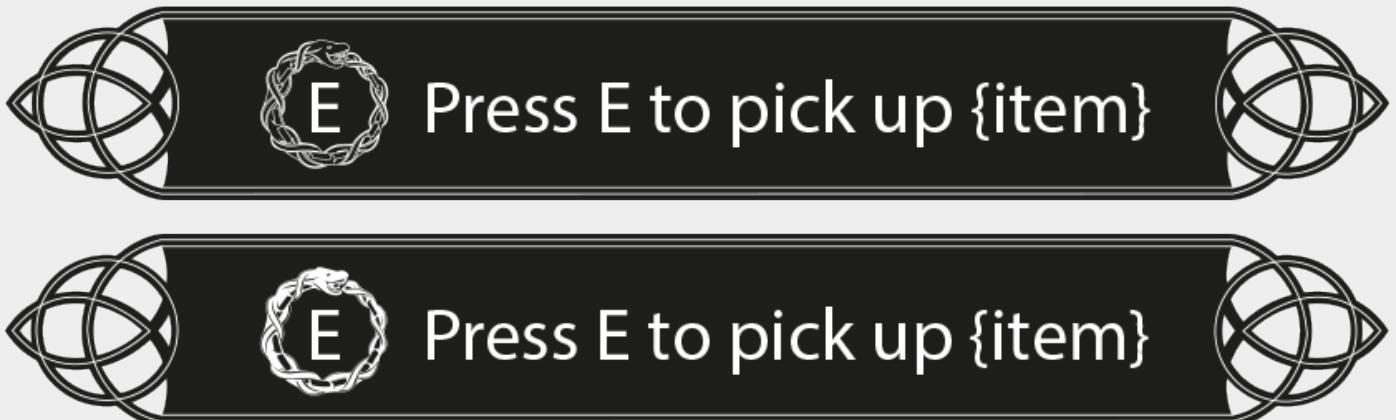
a. Feedback und Anpassung

Nach den zuvor ausgeführten Test innerhalb der Unreal Szene kam es zu weiteren farblichen und formtechnischen Anpassungen des Widgets.

Eine grundsätzlich hellere Variante sollte zu einer weniger überladenen Spielwelt und Umgebung beitragen.



Nach weiteren eingegangenen Vorschlägen und gesammelter Erfahrung wurde auch das UI-Element weiteren kleinen Änderungen unterzogen. So zum Beispiel die Anpassung der Schlange innerhalb des Banners. Neben der grundsätzlichen Optimierung des Schlangensymbols wurde auch die Linienstärke verringert, was zu den beiden folgenden Versionen führt.



Eine Einigung der Gruppe für das obere der beiden Designvorschläge wurde schnell erreicht. Begründung dafür ist das optisch ansprechender Design, sowie ein gleichzeitig besser erkennbares, aber deutlich weniger aufdringliches Schlangensymbol innerhalb des Banners.



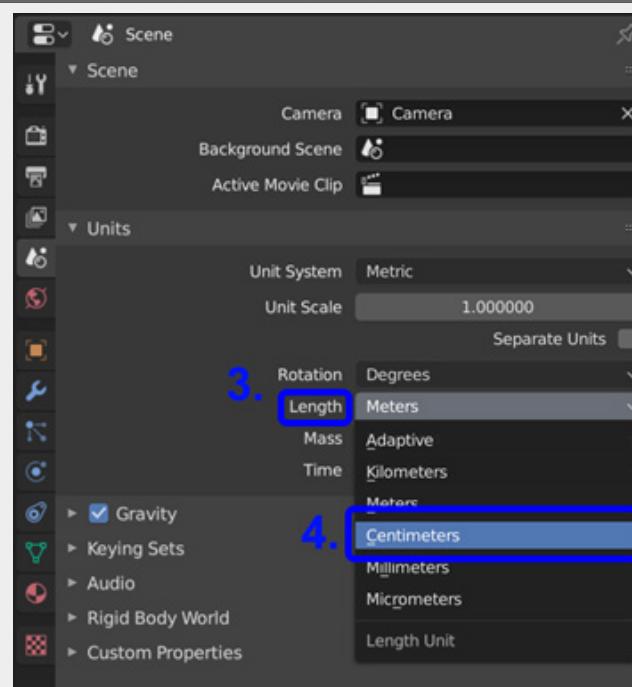
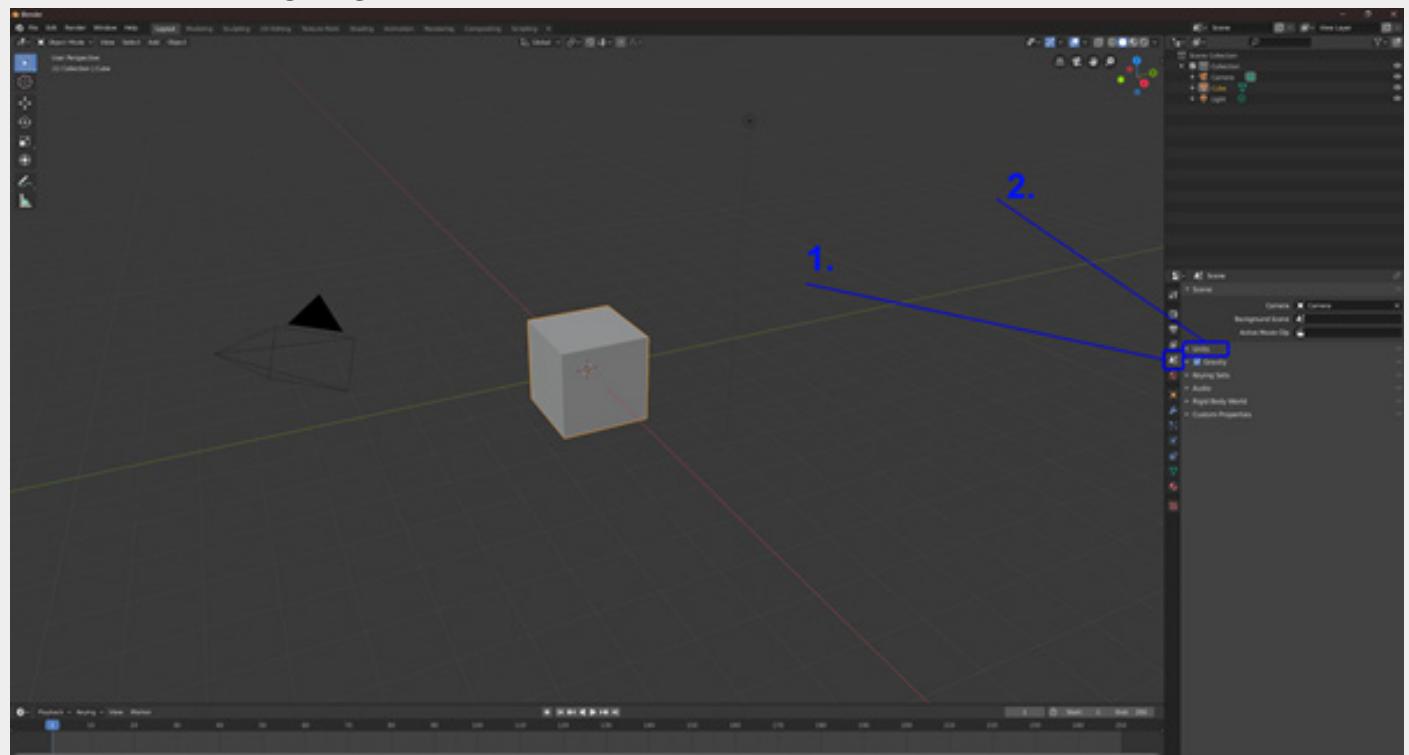
DOKUMENTATIONEN

Dokumentationen

Erstellung von 3D-Inhalten für UE4

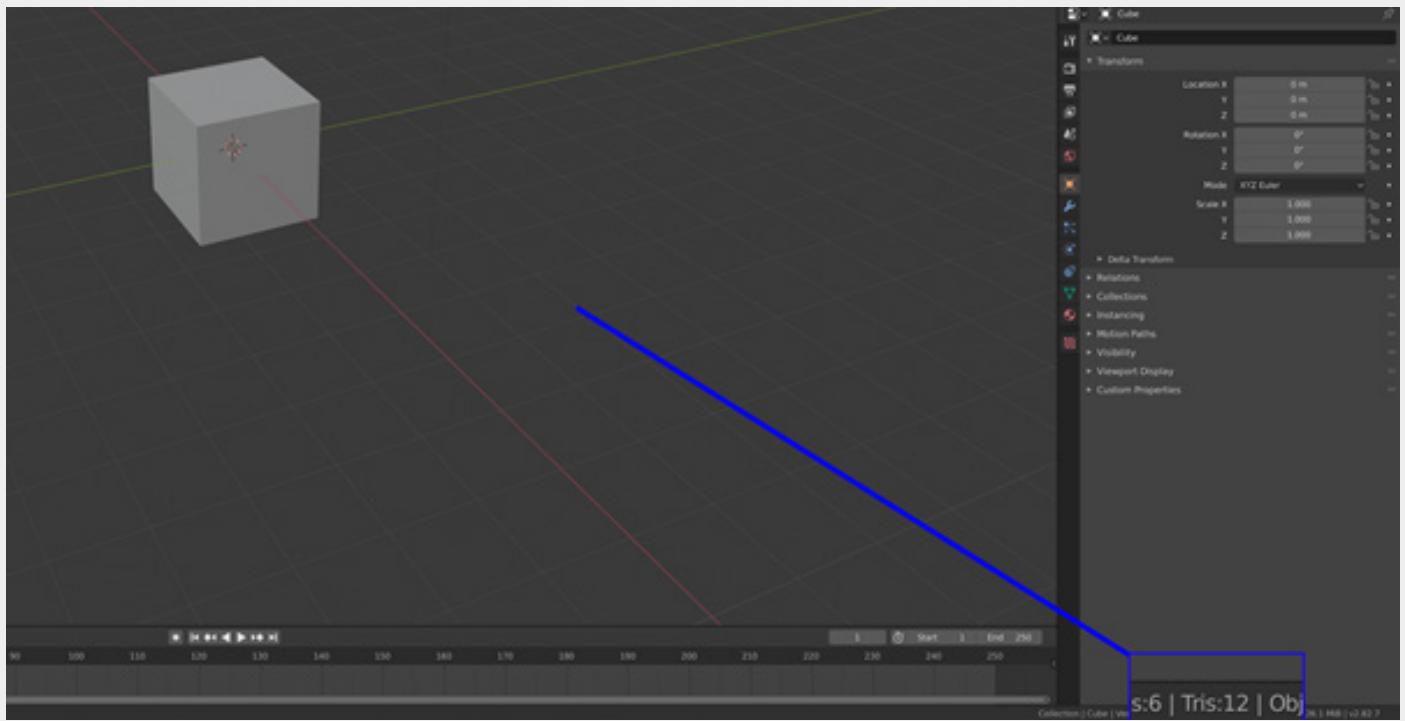
Blender Mess-System

Zuallererst sollte das 3D Tool der Wahl, in unserem Fall Blender, auf das gleiche Mess-System ("cm") wie die Unreal Engine 4 eingestellt werden - sofern nicht bereits geschehen. Modelle sollten dann stets in der Größe ihres realen Gegenstands modelliert werden. Referenzen sind hier unverzichtbar, Augenmaß oder Abschätzen ist nicht genug!



Polygonanzahl

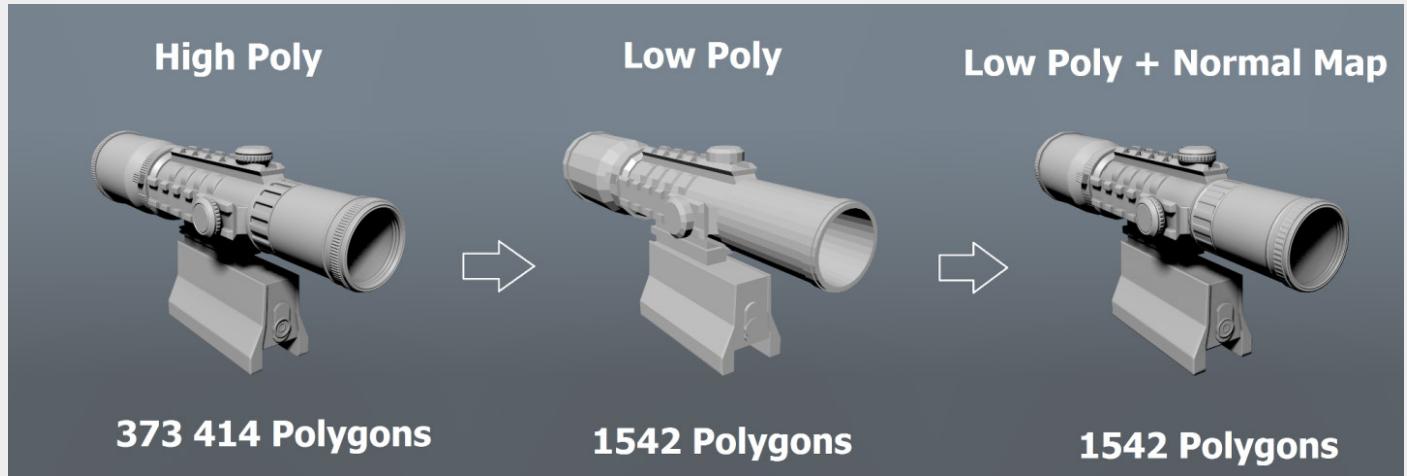
Die Polygonanzahl von 3D Modellen möglichst gering zu halten, ist ein entscheidender Aspekt für deren Performanz in der Engine. Es ist schwierig ein festes Maß anzulegen, wie viele Polygone ein Modell haben sollte. Vielmehr sollte man sich während dem Modellieren stets überlegen, ob manche Details später überhaupt zu sehen sind oder wie nah das Modell zu sehen sein wird. Am ehesten sollte versucht werden die Silhouette des Objektes zu erhalten. Hilfreich, um die Zahl der Polygone im Auge zu behalten, ist regelmäßig einen Blick auf die Anzeige von Blender zu werfen.



High Poly - Low Poly Workflow

Um visuell hochwertige Modelle zu erstellen, möchte man in der Regel Details hinzufügen welche, die zu reduzieren versuchte Polygonanzahl jedoch weit übersteigen lässt. Dies lässt sich lösen, indem man versucht diese Detailinformationen über Texturen ("Normal Maps") wiederzugewinnen. Dafür gilt es neben dem eigentlichen Modell für die Engine ("Low Poly Version") ein weiteres detailliertes Modell ("High Poly Version") zu erstellen. Das High Poly Modell kann über so viele Polygone wie nötig verfügen, um die gewünschten Details darzustellen.

Um diese Details als Normal Map für das Low Poly Modell zu erhalten, müssen diese Texturen "gebaked" werden. Dies geschieht in unserem Fall mit der Software Substance Painter, welche neben Normal Maps noch weitere nützliche Texturen wie Ambient Occlusion oder Thickness erstellt.



Materialien

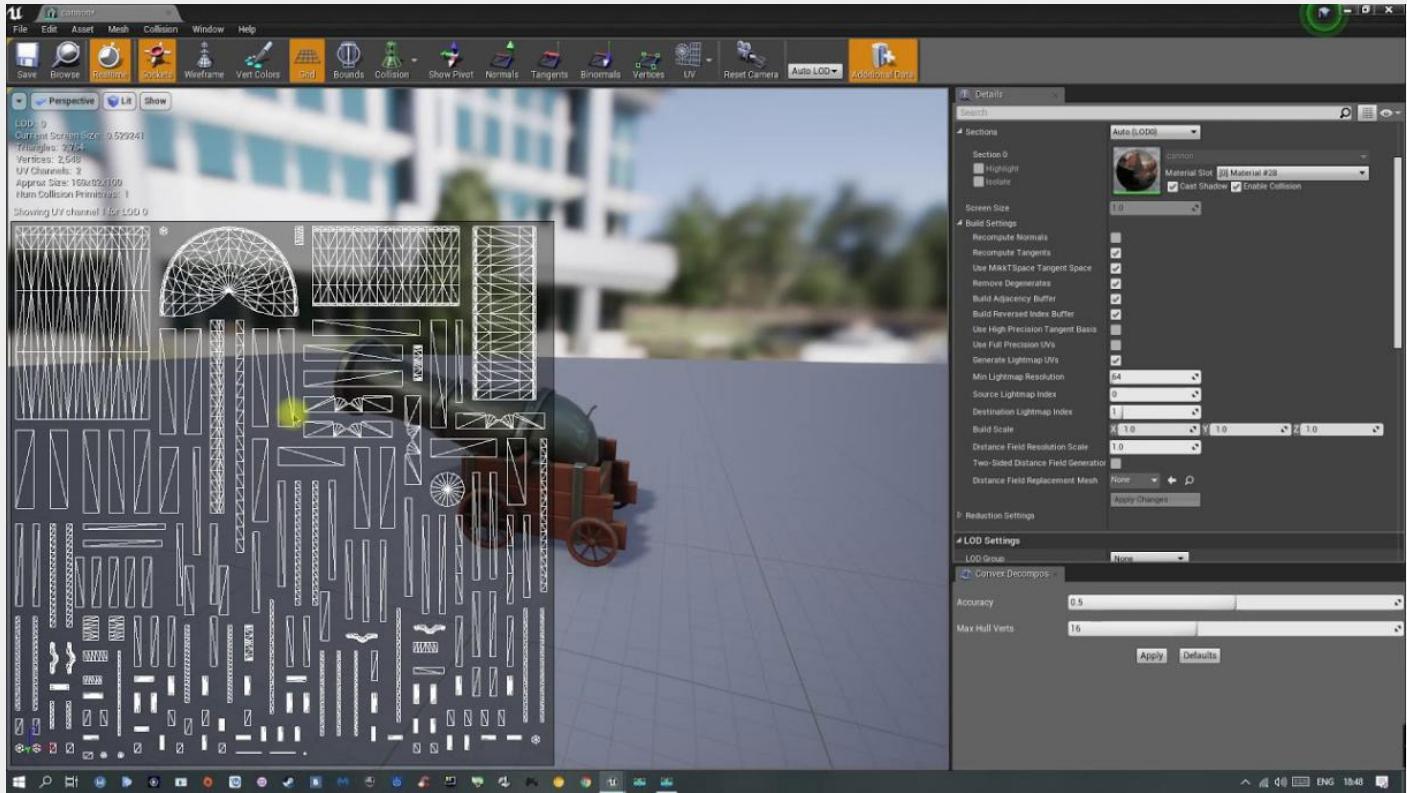
Die Materialien der 3D-Modelle sollten in Blender unbedingt nach den festgelegten Namenskonventionen benannt werden, um die weitere Arbeit in Substance und Unreal Engine zu erleichtern. Generell sollte jedes 3D Objekt nur ein Material erhalten, da jedes Material in der Engine neu gerendert werden muss und so viel Performanz kosten kann. Lediglich bestimmte Ausnahmen wie Charaktere oder große Objekte sollen über mehrere verfügen dürfen. Die für die Engine nötigen PBR Materialien werden nicht in Blender erstellt, sondern in der umfangreichen Software von Allegorithmic, Substance Designer und Painter.

Pivot Points

Um Modelle richtig in einer Szene zu platzieren oder zu animieren ist ein richtig gesetzter Pivot Point notwendig. Daher sollte dieser vor dem Abspeichern unbedingt bedacht werden und an eine für das Mesh sinnvolle Position gebracht werden.

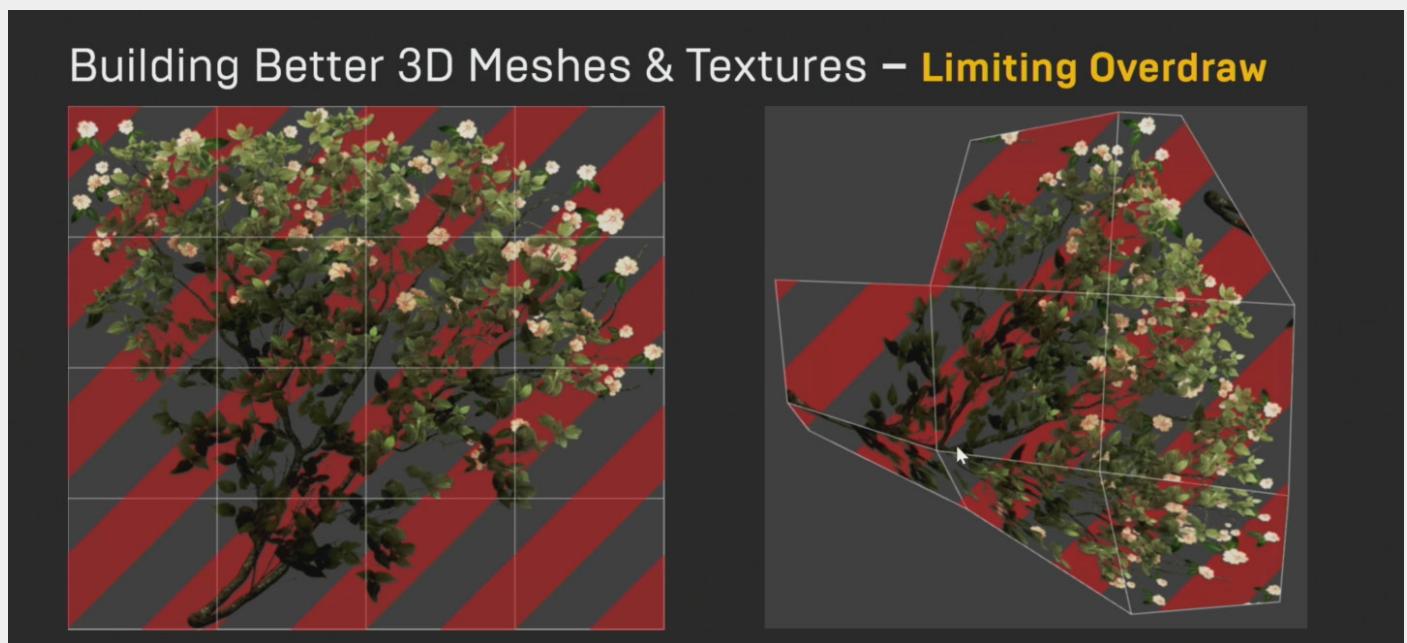
Lightmaps

Unreal Engine ist in der Lage Lightmaps für Static Meshes zu erstellen. Dadurch werden Licht Informationen als Textur gespeichert und müssen nicht in Echtzeit neu berechnet werden. Dies erfordert jedoch die Erstellung separater UV Maps. Während die Engine diese automatisch generieren kann, empfiehlt es sich diese Maps für bessere Ergebnisse selbst in Blender zu erstellen.. Dabei ist darauf zu achten das sich keine Polygone überlappen und genug Platz zwischen den UV Islands gelassen wird.

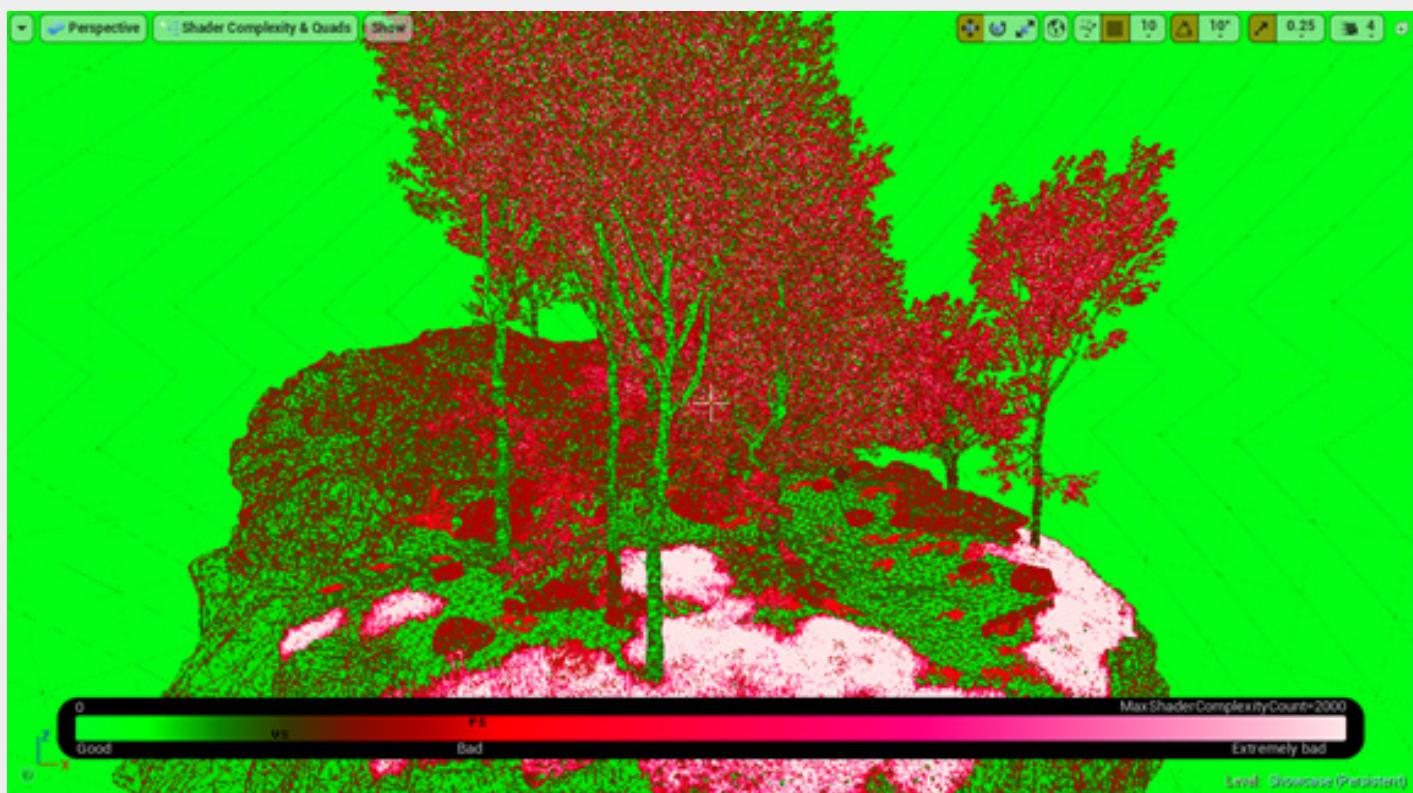
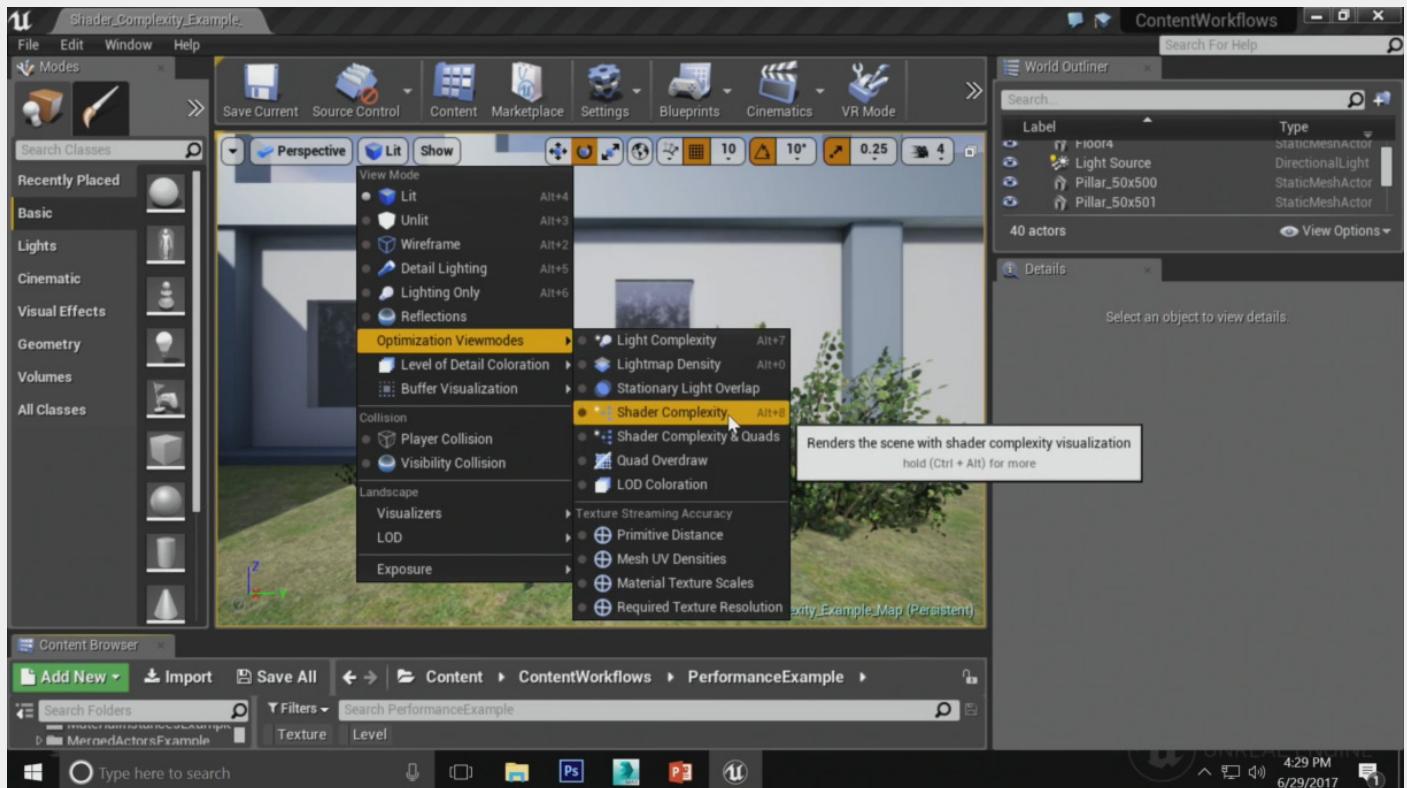


Transparente Materialien

Besonders bei der Erstellung von Pflanzen, aber auch in anderen Bereichen werden transparente Materialien mithilfe von Alpha Masken verwendet. Bei der Verwendung von Transparenz gilt es jedoch gewisse Aspekte im Auge zu behalten, da diese Materialien sehr aufwendig für die Engine zu berechnen sind. Der Teil, welcher keine direkten Farben enthält, also durchsichtig ist, nennt man "Overdraw". Es empfiehlt sich diesen zu reduzieren so gut es geht, indem man das Modell anpasst, um die Form der eigentlichen Textur so gut es geht nachzuahmen. Es ist eine durchaus valide Option hierfür, mehr Polygone in Kauf zu nehmen und dadurch eine performantere Lösung zu erhalten.



Die Unreal Engine bietet zudem ein Tool zur Selbstkontrolle im Bezug auf Overdraw.

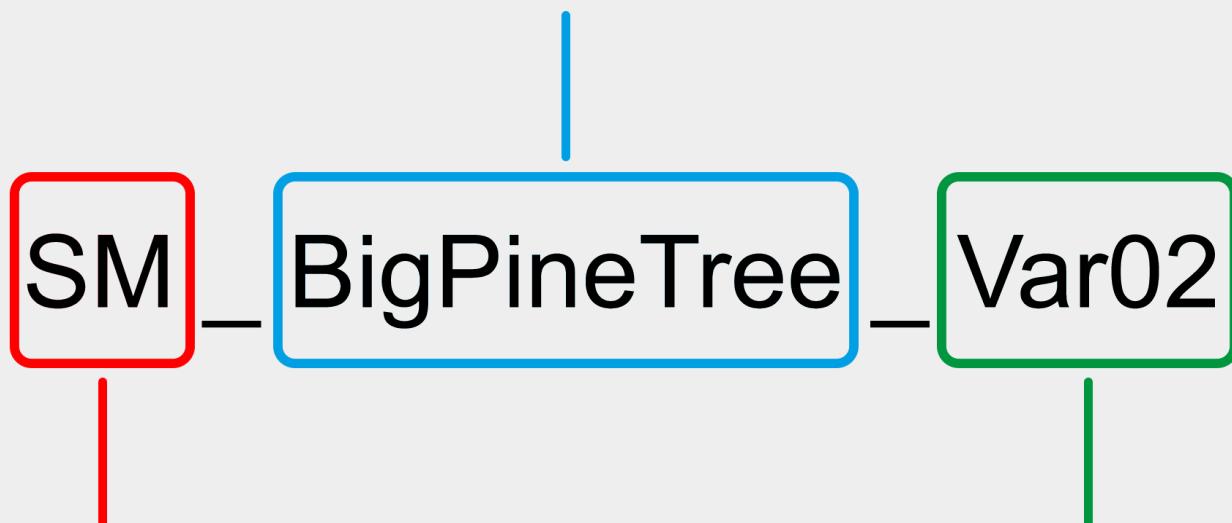


Namenskonvention innerhalb von Unreal Engine 4

Um eine effiziente Arbeit innerhalb der Unreal Engine 4 zu gewährleisten, müssen Dateien während des Entwicklungsprozesses schnell und eindeutig auffindbar sein. Da mehrere Personen an einem Projekt arbeiten, ist die Benennung von Dateien nach einem prägnanten und einheitlichen Schema unverzichtbar. Somit möchten wir im Folgenden Regeln für eine einheitliche Namenskonvention aufstellen:

Alle Dateibenennungen sollen an erster Stelle eine Abkürzung für dessen Objekttyp in Großbuchstaben haben. Darauf folgt eine prägnante und einmalig vorhandene Beschreibung des Inhalts in CamelCase-Notation. Zusätzlich besteht die Option, Zusatzinformationen für Nummerierung von Varianten oder genauerer Spezifizierung der Datei anzuhängen. Somit bestehen Dateinamen aus zwei oder drei Inhaltsblöcken, welche durch Verwendung von Unterstrichen ("_") voneinander zu trennen sind.

Es handelt sich um einen großen
Kiefernbaum.



Das Objekt ist vom Typ
„Static Mesh“ (SM).

Bei dem Mesh handelt es sich
um die Variante 2.

Beispiele

TEX_Doorframe_Roughness

-> Texturdatei für einen Türrahmen in Form einer Roughness Map.

BP_MainCharacter

-> Die Blueprint für den Hauptcharakter im Spiel.

Liste der zu verwendeten Abkürzungen für dessen Objekttypen

SM	Static Mesh
SKM	Skeletal Mesh
BP	Blueprint
MAT	Material
SFX	Sound
AN	Animation
TEX	Textur

Für die Spezifizierung der unterschiedlichen Texture Maps von Materialien soll ebenfalls eine einheitliche Bezeichnung genutzt werden. Diese ist als Zusatzinformation anzuhängen (siehe Beispiele oben).

..._Diffuse
..._Roughness
..._Metalness
..._Normal
..._AO (Ambient Occlusion)
..._Alpha

Namenskonvention im Erstellungsprozess

Die Dateibenennungen außerhalb der Unreal Engine 4 während der Erstellung der Inhalte soll denen innerhalb der Spieleengine gleichen. Da im Laufe des Prozesses mehrere Kopien dieser Dateien entstehen, sollen diese bei der Benennung um einen weiteren Informationsblock erweitert werden, welcher dessen Version kennzeichnet.

Beispiele

SM_BigPineTree_Var02_03

SKM_Zombie_Var01_29

Source Control

Source Control auch bekannt als Versionskontrolle oder Quellcode Verwaltung, wird über Softwaresysteme bereitgestellt. Diese Systeme bieten zahlreiche nützliche Funktionen für Entwickler. In einer speziellen Datenbank verfolgt das System die Änderungen, welche an einem Projekt vorgenommen werden. Dies kann der Austausch eines 3D-Modells oder auch die Ergänzung von Quellcode sein. Jede erfasste Änderung wird als Version angelegt und gespeichert, was es möglich macht, diese nachzuvollziehen und gegebenenfalls rückgängig zu machen, sollte ein Fehler unterlaufen sein.

Ein weiterer Vorteil ist die (dezentrale) Zusammenarbeit mehrerer Personen an einem Projekt. Von einem definierten Server auf dem das eigentliche Projekt abgelegt ist, können sich die Entwicklungsteilnehmer eine aktuelle lokale Kopie anlegen. Wird diese bearbeitet und verändert, kann der Teilnehmer sie wieder in das Repository, also den eigentlichen Speicherort des Projekts, einpflegen.

Weitere Personen, die danach arbeiten möchten, werden über eine neue Version benachrichtigt und können diese herunterladen, um so selbst auf dem aktuellen Stand zu sein.

Zu einem guten Versionskontrollsysteem gehört auch das Warnen vor versehentlichen Überschreibungen beim Zusammenführen zweier Versionen. Neben einem einfach einsehbaren Verlauf einer jeden Datei im System und den dementsprechenden Sicherungskopien dieser Versionen, stellt die Arbeit mit einem Versionskontrollsysteem also eine effiziente Möglichkeit bereit, mit verteilten Teams zeitgleich an einem zentralen Projekt zu arbeiten, ohne dass Änderungen verloren gehen oder versehentlich überschrieben werden.

Erwägung von Versionskontrolle für das Projekt

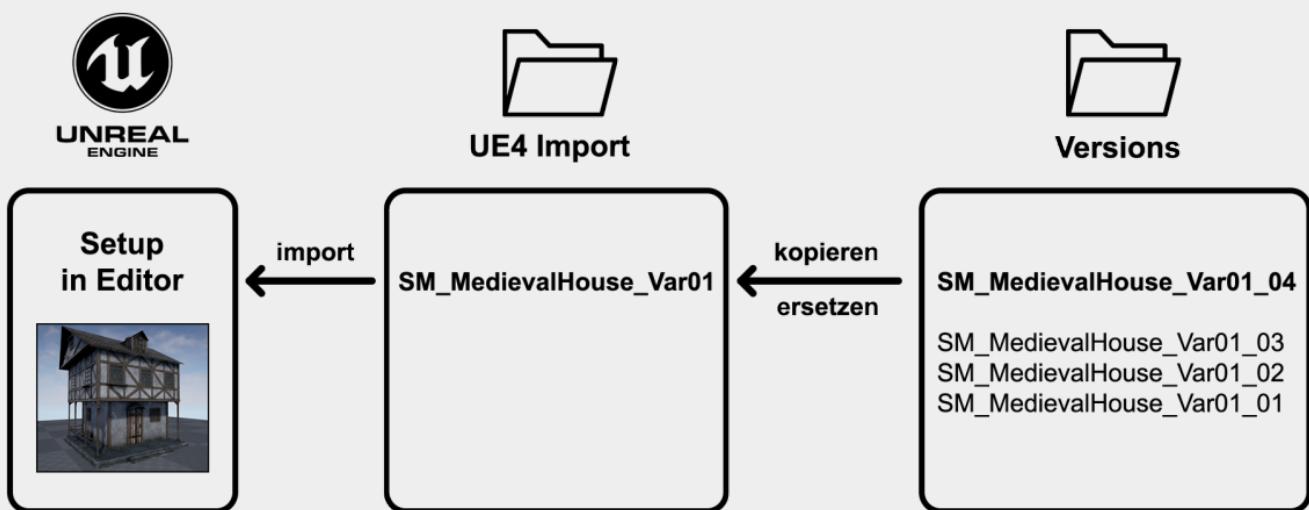
Ausschlaggebender Punkt für die Einführung von Versionskontrolle sind die Vorteile der Datensicherung auf einem extern gemieteten System und die dadurch erhöhte Sicherheit in diesem Bereich. Des Weiteren ist es von Vorteil einen zentralen Punkt zu definieren, worauf die Mitglieder des Projektteams zugreifen können. Daraus folgt auch eine reduzierte Ansammlung unkontrollierter lokaler Versionen, die gegebenenfalls Probleme bei der manuell notwendigen Zusammenführung erzeugen könnten.

Ausgehend von der zuvor definierten Entwicklungsumgebung in Unreal Engine 4 und deren Dokumentation ist dieser zu entnehmen, dass standardmäßig die beiden Anbieter von Versionskontrollsystemen, SVN und Perforce, unterstützt werden. Nach ausgiebigem Vergleich mit weiteren Anbietern (GitHub, GitLab, Beanstalk, AWS, ...), ergibt sich, dass Perforce den besten Umgang mit Dateien vorzuweisen hat, die keinen reinen Quellcode beinhalten. Beispielsweise Modelle im FBX oder OBJ Format, sowie Unreal eigene Dateiformate.

Für das Aufsetzen des Projektserver wird ein zentrales System benötigt. Der Server, konfiguriert nach angemessenen Anforderungen des Projekts ist bei einem erfahrenen deutschen Anbieter für Serverhosting eingerichtet. Die zugehörige Perforce Software mit dem Namen „Helix Core“ ist manuell darauf installiert und ebenfalls für das Projekt passend konfiguriert worden.

Implementierung der Versionskontrolle

Auf dem Server befindet sich das gesamte Unreal Projekt. Parallel in diesem Repository angelegt ist ein eigens für Modelle gedachter Ordner. Jeder Rechner kann über den Helix Core Client eine lokale Kopie des Repository herunterladen. Die Modelle werden nach einer einheitlichen Namenskonvention erstellt und gespeichert und so im Projekt eingepflegt. Die aktuellste Version wird als Unreal Import definiert. Aus diesem Verzeichnis werden immer aktuelle Versionen eines Objekts direkt im richtigen Pfad erkannt und eingebunden. Wird das gesamte Projekt nun von einem Nutzer heruntergeladen, erhält er sowohl den aktuell gehaltenen Ordner mitsamt den verwendeten Versionen eines Objekts, sowie den momentanen Stand des kompletten Projekts, welches Strukturbedingt ebenfalls Zugriff aus den zuvor benannten Ordner hat.



Source Control mit Perforce Helix Core Server gehosted bei Strato.de

Logindaten Strato:

- Username: willbold_l
- Passwort:

Serverdaten:

- Betriebssystem: Ubuntu 18.04 LTS 64bit
- CPU vCores: 6
- RAM: 8 GB
- Speicherplatz: 600GB
- Hostname: h2875935.stratoserver.net
- IP-Adresse: 85.214.187.250
- Root:

Installationsanleitung Helix Core

<https://www.perforce.com/manuals/p4sag/Content/P4SAG/install.linux.packages.install.html>

1. Download the public key at <https://package.perforce.com/perforce.pubkey>
 - a. <https://vitux.com/how-to-download-a-file-on-ubuntu-linux-using-the-command-line/>
 - b. apt-get install wget
 - c. wget <https://package.perforce.com/perforce.pubkey>
2. To obtain the fingerprint of the public key, run:

```
gpg --with-fingerprint perforce.pubkey
```
3. Verify that it matches this fingerprint:
E581 31C0 AEA7 B082 C6DC 4C93 7123 CB76 0FF1 8869
 - a. ?

```
Wird in »perforce.pubkey« gespeichert.

perforce.pubkey          100%[=====] 1,67K --.-KB/s in 0s
2020-04-23 23:14:33 (104 MB/s) - »perforce.pubkey« gespeichert [1707/1707]

root@h2875935:~# gpg --with-fingerprint perforce.pubkey
gpg: Verzeichnis `/root/.gnupg' erzeugt
gpg: Die "Keybox" `/root/.gnupg/pubring.kbx' wurde erstellt
gpg: WARNING: Kein Kommando angegeben. Versuche zu raten was gemeint ist ...
pub    rsa4096 2013-08-16 [SC] [verfällt: 2023-08-14]
uid          Perforce Software (Package Signing) <support+packaging@perforce.com>
root@h2875935:~# df -h
Dateisystem      Größe Benutzt Verf. Verw% Eingehängt auf
/dev/ploop63444pl 591G   1,4G  566G   1% /
none            4,0G     0  4,0G   0% /sys/fs/cgroup
none            4,0G     0  4,0G   0% /dev
tmpfs           4,0G     0  4,0G   0% /dev/shm
tmpfs           4,0G   1,1M  4,0G   1% /run
tmpfs           5,0M     0  5,0M   0% /run/lock
none            4,0G     0  4,0G   0% /run/shm
```

4. Add the Perforce packaging key to your APT keyring

```
wget -qO - https://package.perforce.com/perforce.pubkey | sudo apt-key add -
```
5. Add the Perforce repository to your APT configuration.
 - a. Create a file called /etc/apt/sources.list.d/perforce.list with the following line:

```
deb http://package.perforce.com/apt/ubuntu {distro} release
```

Where {distro} is replaced by one of the following:
precise, trusty, xenial or bionic (= 18.04).
6. Run apt-get update
7. Install the package by running sudo apt-get install helix-p4d

```
:::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::  
::  
:: Thank you for choosing Perforce Helix  
:: The following has been installed by the 'helix-p4d' package:  
::  
:: - The Helix Version Engine (p4d)  
:: - A 'perforce' system user  
:: - p4dctl, a tool for managing Perforce service instances  
:: - The Helix Command-Line Client (p4)  
::  
:: The Helix Version Engine is now installed, but not yet configured.  
:: You must run the following to configure p4d (as root):  
::  
::     sudo /opt/perforce/sbin/configure-helix-p4d.sh  
::  
:::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
```

Konfigurationsanleitung Helix Core

<https://www.perforce.com/manuals/p4sag/Content/P4SAG/install.linux.packages.configure.html>

1. Perforce Service name [master]: helix
2. Perforce Server root (P4ROOT): /opt/perforce/servers/helix (= Standard)
3. Create directory? (y/n) [y]: y
4. Perforce Server unicode-mode (y/n) [n]: n
5. Perforce Server case-sensitive (y/n) [y]: y
6. Perforce Server address (P4PORT) [ssl:1666]: 1666 (= Standart)
7. Perforce super-user login [super]: lisa
8. Perforce super-user password:

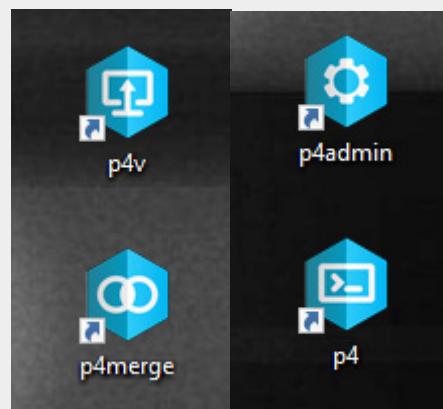
Here is what has been done so far:

- Your p4d service settings have been written to the following p4dctl configuration file:
`/etc/perforce/p4dctl.conf.d/helix.conf`
- The p4d service has been initialized with the P4ROOT:
`/opt/perforce/servers/helix/root`
- The p4d service has been started with the P4PORT: `ssl:1666`
- The p4d service has been set to Security Level 3.
- The new Perforce super-user '`lisa`' has been created and the password has been set to the one specified.

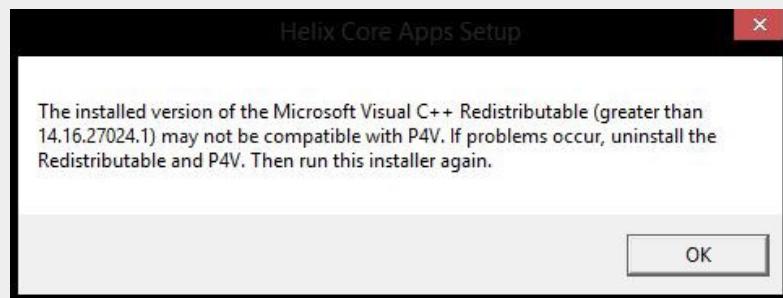
To connect to this p4d service from another machine, include this machine's name or IP address in the P4PORT. P4PORT=ssl:85.214.187.250:1666

Installation der Helix Core Clientanwendungen

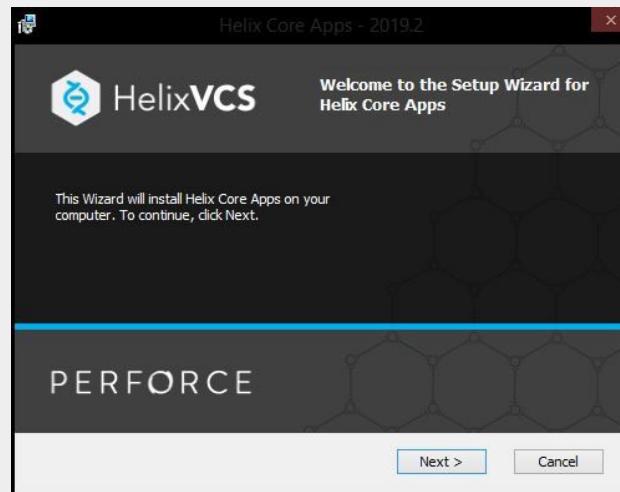
1. Download der Datei P4inst64 (für Windows 64 Bit OS) von der Perforce Homepage
<https://www.perforce.com/downloads/helix-visual-client-p4v>
2. Installation der Client Tools:
 - a. P4V ist der eigentliche Source Control Client für Commit, Pull und Push etc.
 - b. P4Merge zum Vergleich der Versionen und Merging von Branches (Streams)
 - c. P4Admin (Tool für den Server Administrator)
 - d. P4 - Command Line Tool (Tool für den Server Administrator)



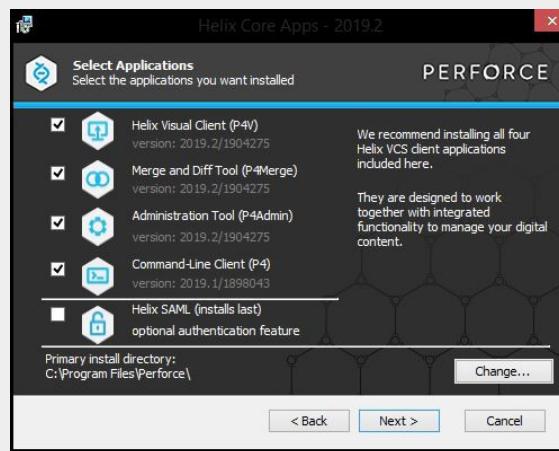
3. Ablauf:
 - a. Ausführen der Datei „p4vinst64.exe“
 - b. Abfrage für die installierte Version von Microsoft Visual C++ Redistributable auf dem Rechner
 - i. falls nicht vorhanden wird sie von Perforce nachinstalliert, dies führt zu einem Neustart des Rechners
 - ii. falls bereits installiert, aber in höherer Version als 14.16.27024.1 vorhanden, erscheint einmalig eine Warnmeldung, die mit OK anerkannt werden kann



- c. Next.



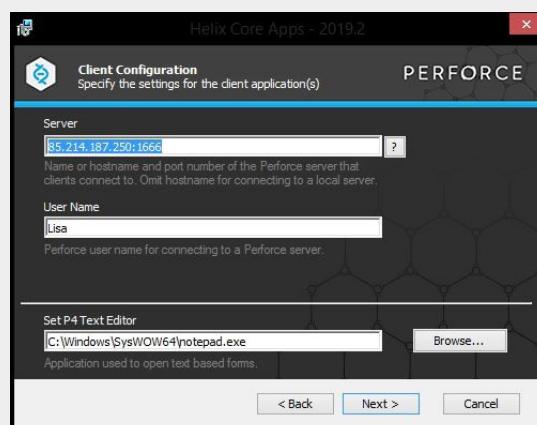
- d. Auswahl der zu installierenden Softwarekomponenten. Für nichtadministrative Nutzer reichen vorläufig der „Helix Visual Client (P4V)“ und für später das „Merge and Diff Tool (P4Merge)“ Der Standard Installationspfad ist „C:\Programme\Perforce“. Dort liegen falls benötigt die *.exe der Programme, da von Perforce selbst keine Desktopverknüpfung erzeugt wird.



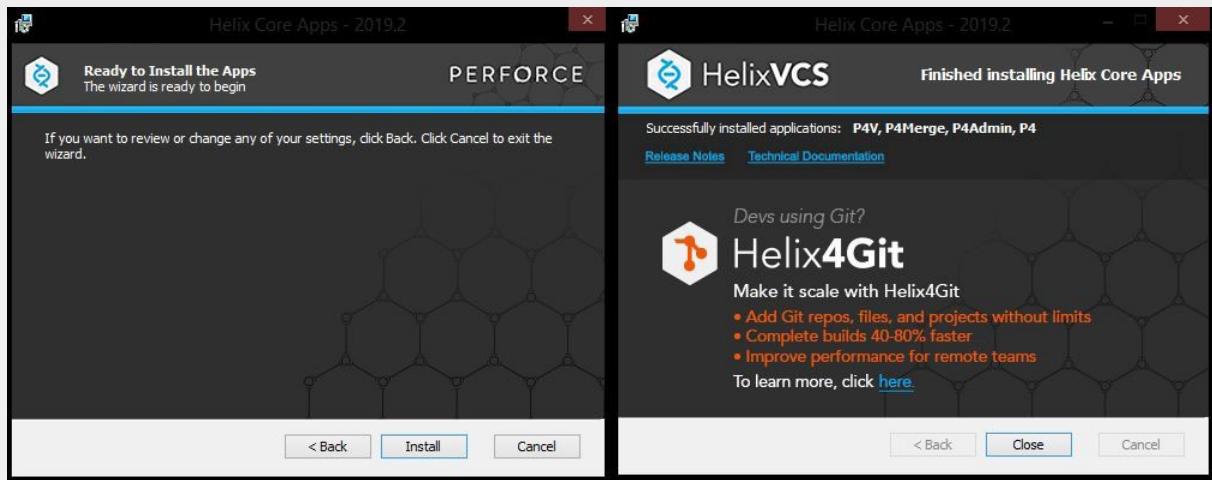
- e. Konfiguration der Verbindung zum Server.

Server = ssl:85.214.187.250:1666

User = jeweiliger Vorname klein geschrieben; Achtung: Case Sensitive

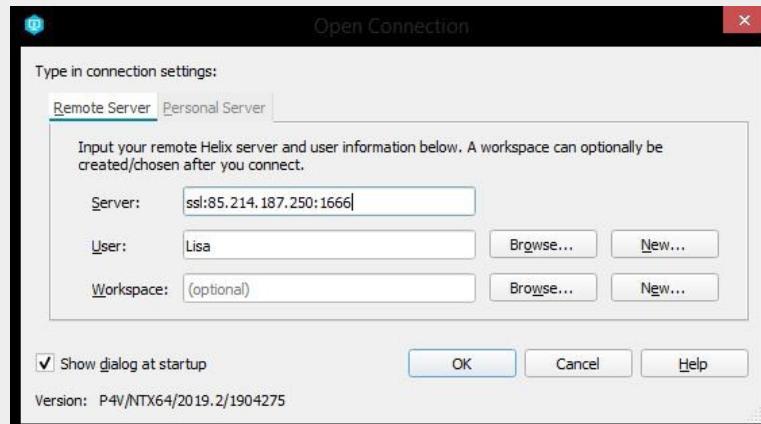


- f. Alle Angaben bestätigen und nach Beenden der Installation schließen.



Verbindung der Clients mit dem Server

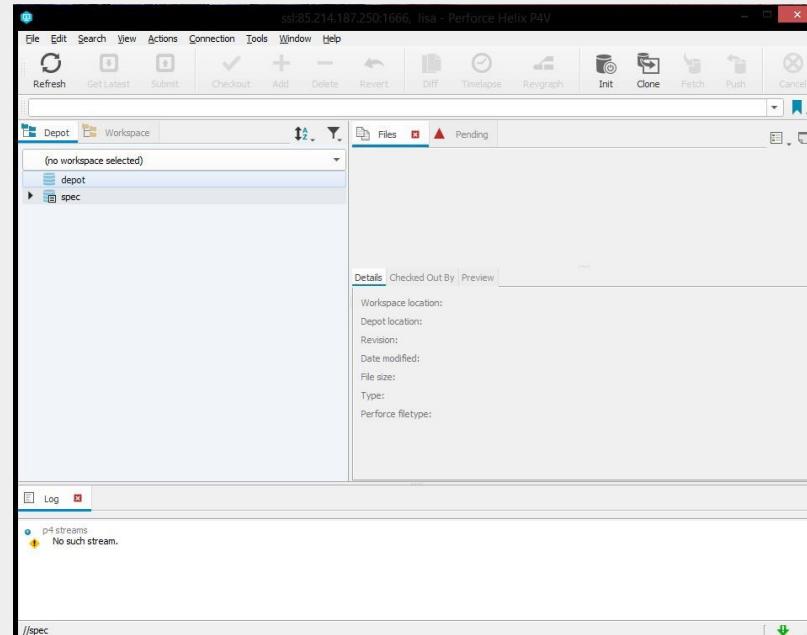
1. Starten des P4V Clients (über Startmenü, Desktopverknüpfung, direkte *.exe, ...)
2. Connection Settings für den Remote Server eintragen (Daten, die bei der Installation verwendet wurden sollten bereits vorab eingetragen sein)



3. Fingerprint des Servers für den Rechner vertrauen und bestätigen



4. Zugewiesenes Passwort für den User eingeben. Passwörter sind nach dem Schema „Nachname1234!” angelegt. Der jeweilige Nachname beginnt mit Großbuchstaben.
5. Startbild.



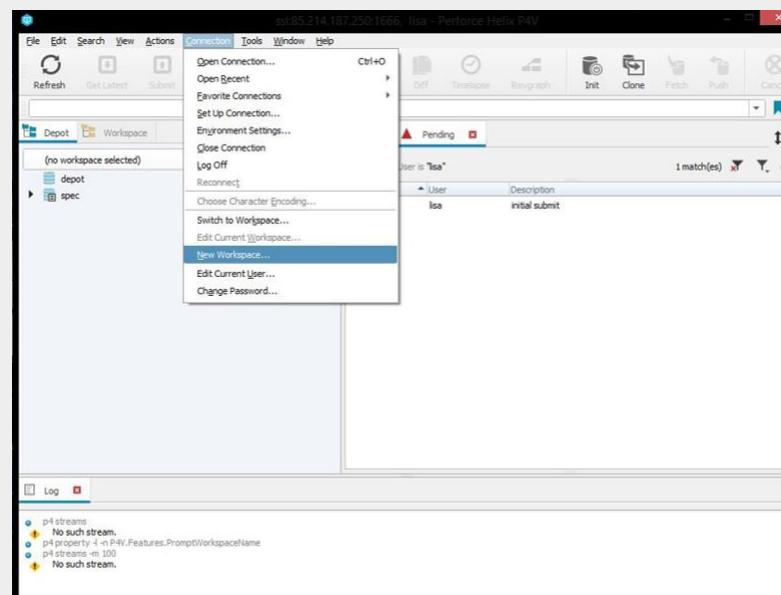
Anlegen eines lokalen Workspaces

Erklärung: Der Workspace ist das Verzeichnis auf der lokalen Maschine in welche der Server die Dateien aus dem Depo (=Repository bei Perforce) ablegen soll bzw. welches Verzeichnis auf Änderungen überwacht werden soll.

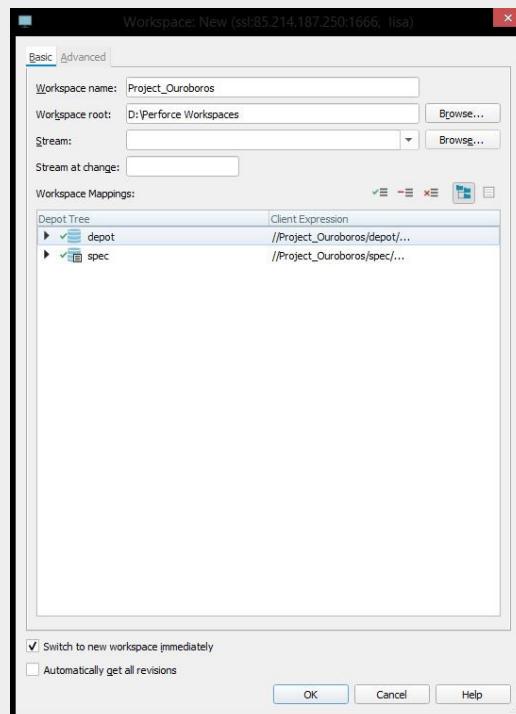
Ein User kann mehrere Workspaces pro Gerät haben, jedoch sollten diese auf jedem Gerät neu eingerichtet werden.

Beispiel: ein User hat einen Rechner mit Workspace für unser Projekt mit dem Namen Workspace 1, ebenso besitzt er einen Laptop. Durch den Login kann der Laptop „sehen“, dass Workspace 1 existiert, wird dieser jedoch ausgewählt kommt es zu Problemen. Deshalb auf dem Laptop Workspace 2 anlegen.

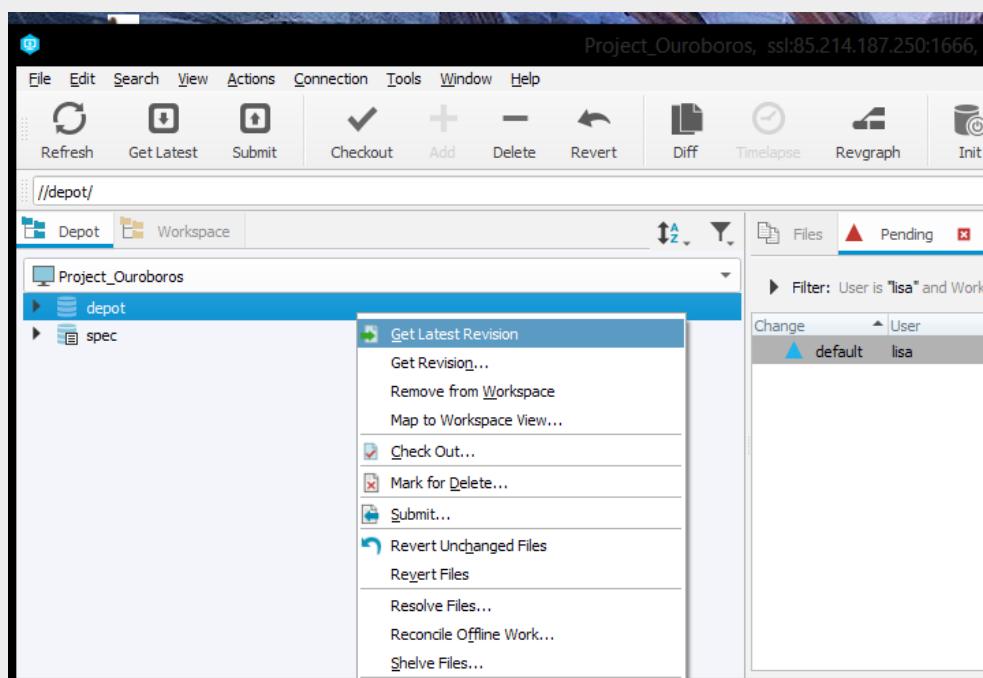
1. In der Menüleiste oben auf „Connections“ --> „New Workspace“



2. Im folgenden Fenster einen sinnvollen Namen und ein Verzeichnis für die Ablage wählen. Standardmäßig ist das Depot auf dem Server schon ausgewählt.
3. Mit OK bestätigen.

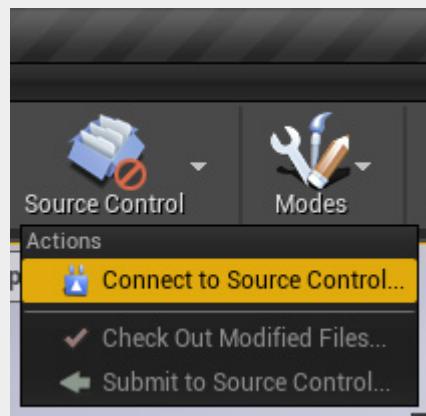


4. Im Tab „Workspace“ sollte der Pfad nun eingepflegt sein.
5. Den initialen Pull des gesamten soweit verfügbaren Projekts starten. Dazu im Tab „Depot“ links neben „Workspace“ den Ordner „depot“ auswählen und entweder mit Rechtsklick --> „Get Latest Revision“ oder dem Button „Get Latest“ im Menü Band den Download beginnen.



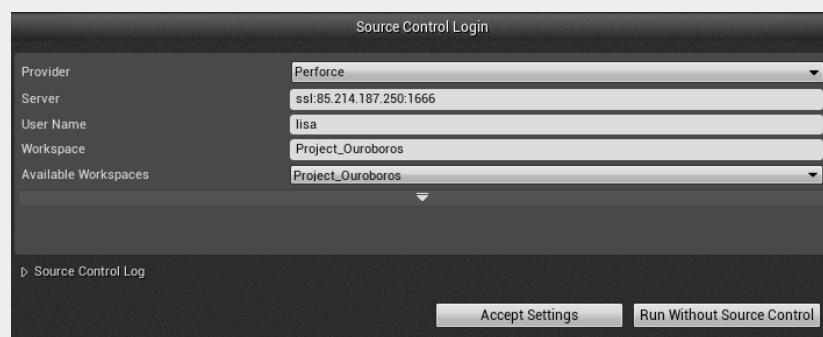
Einbinden von Perforce Source Control im UE Projekt

1. Das Projekt lokal öffnen.
2. In der Menüleiste den Button „Source Control“ aktivieren und „Connect to Source Control“ auswählen.
 - a. beim ersten Mal Einrichten kann es sein, dass die alten Informationen des letzten pushenden Users eingetragen sind. Sobald man auf „Change Setting“ geht, springt die Anzeige auf „nicht Verbunden“ (= durchgestrichen) bis die Login Daten wieder stimmen.



3. Auswählen des Providers im Dropdown, in unserem Fall Perforce.
 4. Eingeben der Server und Login Daten.
 5. Auswählen der lokalen Workspaces.
 6. „Accept Settings“.
- Rechts unten sollte eine kurze Bestätigungsnachricht auftauchen und das Verbindungssymbol vom Verbotsschild auf den grün hinterlegten Doppelpfeil wechseln.

Beispiel auf Rechner 1:



Helix Core Workflow mit P4V und Unreal Engine V1

7. P4V Client öffnen und am Server anmelden.
8. Prüfen, ob der richtige Workspace für den jeweiligen Rechner ausgewählt ist.
9. Ordner „depot“ im Tab „Depot“ anwählen und den aktuellsten Stand beziehen.
10. Projekt lokal öffnen / Unreal Editor starten.
11. Prüfen, ob die Verbindung zu Perforce unter Source Control aktiv ist. Falls nicht: aktivieren!
12. Checkout der Datei, an der gearbeitet wird mit Rechtsklick --> Source Control --> Checkout (somit ist die Datei gesichert gegen versehentliches Überschreiben durch einen anderen Nutzer)
 - a. ein roter Haken an der Datei steht für eigenen Checkout
 - b. ein blauer Haken zeigt an, dass ein anderer User die Datei derzeit bearbeitet
 - c. ein Ausrufezeichen sagt aus, dass die Datei nicht auf dem aktuellsten Stand/Revision ist, die lässt sich mit Sync, welche anstelle von Checkout erschien, beheben
13. Checkin der Datei auf demselben Weg. Dies stellt gleichzeitig den Befehl Commit und Push dar, weswegen ein Änderungskommentar gefordert wird. Die Datei kann nun von einem anderen Benutzer bearbeitet werden.

FAZIT UND AUSBLICK

Fazit

Das Semester hat mit einer sehr motivierten Projektgruppe angefangen. Kreative Ideen haben eine sehr positive Stimmung ausgelöst. In den ersten Wochen wurde stark an der Konzeption gearbeitet. Jedoch hat diese Motivation mit dem Beginn der Vorlesungszeit sehr nachgelassen. Durch Corona und dem damit verbundenen zusätzlichen Druck auf die restlichen Module innerhalb des Studiums, flachte die Stimmung und Motivation immer weiter ab. Hierdurch wurden gesetzte Deadlines nicht mehr ernst genommen und Wochenbesprechungen wurden nur noch abgehalten, damit sie gehalten wurden. Aus diesem Grund ist es nicht verwunderlich, dass die Projektgruppe ihr Ziel nicht erreicht hat.

Dennoch hat die Gruppendynamik innerhalb der Gruppe von Anfang an gestimmt. Es konnte offen ausgesprochen werden, wenn etwas nicht funktioniert. Zusätzlich gab es auch trotz der Umstände Phasen, in denen das Projekt gut voran gekommen ist.

Insgesamt lässt sich sagen, dass es viel Potential nach oben gibt. Die Projektgruppe hat eingesehen, dass sie sich mit ihrem Ziel zu viel vorgenommen hat. Es konnte zu Beginn des Semesters schwer eingeschätzt werden, inwieweit die Corona-Situation das Team beeinflussen wird. Aus dieser Erfahrung soll nun gelehrt werden.

In einer Nachbesprechung des Semesters wurden alle Aspekte, die verbessert werden können, und direkte Verbesserungsvorschläge gesammelt. Diese werden nun konsequent umgesetzt.

Ausblick

Die Semesterbesprechung hat dazu geführt, dass das Team nun einen Projektleiter eingeführt hat. Wochenbesprechungen sollen strukturierter und geplanter ablaufen, um Zeitverschwendungen zu vermeiden und die Qualität der Besprechungen zu verbessern.

Gruppenmeetings finden nun nur noch alle zwei Wochen statt. Zusätzlich gibt es jedoch in kleineren Gruppen selbstorganisierte Briefings, welche dafür sorgen sollen, dass die Teammitglieder einzelner Untergruppen immer auf dem neusten Stand sind.

Damit sollen Deadlines ernster genommen werden, wodurch das Projekt wieder stärker voran kommt. Das Ziel für die kommenden Wochen soll sein das Kirchenschiff, sowie das Pastorenzimmer fertig zu stellen, sodass das Kirchenlevel als Prototyp gespielt werden kann. Daraufhin folgt der Untergrund mit den Katakomben.

Für das nächste Semester ist nun geplant den Feinschliff am Kirchenlevel und den Katakomben vorzunehmen, sowie am Ende des Kirchenlevels einen kleinen Filmschnitt einzubauen. Je nach übriger Zeit und Puffer, kann dieser beliebig ausgebaut werden. Der Zeitplan wurde dementsprechend angepasst.