Inhaltsverzeichnis

[Abbildungsverzeichnis 4](#_Toc48575845)

[1. Einleitung 5](#_Toc48575846)

[1.1 Idee des Medienprojekts 5](#_Toc48575847)

[1.2 Zielsetzung 5](#_Toc48575848)

[1.3 Verortung in der Industrie 5](#_Toc48575849)

[2. Kontext 6](#_Toc48575850)

[2.1 Referenzen 6](#_Toc48575851)

[2.1.1 Technische Referenzen 6](#_Toc48575852)

[2.1.1.1 Kingdom: Two Crowns 6](#_Toc48575853)

[2.1.1.2 Minecraft 8](#_Toc48575854)

[2.1.1.3 Terraria 8](#_Toc48575855)

[2.1.1.4 World of Warcraft 9](#_Toc48575856)

[2.1.1.5 Clonk Rage 10](#_Toc48575857)

[2.1.2 Stilistische Referenzen 12](#_Toc48575858)

[2.1.2.1 Kingdom: Two Crowns 12](#_Toc48575859)

[2.1.2.2 斩妖行 Eastern Exorcist 13](#_Toc48575860)

[2.1.2.3 Ori and the Will of the Wisps 14](#_Toc48575861)

[2.2 Vorstellung Spezieller Techniken 14](#_Toc48575862)

[2.2.1 Crafting 14](#_Toc48575863)

[2.2.1.1 Crafting Grundlagen 14](#_Toc48575864)

[2.2.1.2 Minecraft 14](#_Toc48575865)

[2.2.1.3 Terraria 16](#_Toc48575866)

[2.2.1.4 World of Warcraft 17](#_Toc48575867)

[18](#_Toc48575868)

[2.2.1.5 Clonk Rage 18](#_Toc48575869)

[3. Methodik 20](#_Toc48575870)

[3.1 Projektumsetzung 20](#_Toc48575871)

[3.1.1 Crafting System 21](#_Toc48575872)

[3.1.2 Engine 21](#_Toc48575873)

[3.2 Meilensteinplanung 21](#_Toc48575874)

[3.2.1 Meilensteindefinierung 21](#_Toc48575875)

[3.2.2 Meilenstein 1 21](#_Toc48575876)

[3.2.3 Meilenstein 2 21](#_Toc48575877)

[3.2.4 Meilenstein 3 22](#_Toc48575878)

[3.2.5 Meilenstein 4 22](#_Toc48575879)

[3.2.6 Meilenstein 5 22](#_Toc48575880)

[4. Durchführung 23](#_Toc48575881)

[4.1 Grundlegende Mechaniken 23](#_Toc48575882)

[4.1.1 Player Character 23](#_Toc48575883)

[4.1.2 NPCs 23](#_Toc48575884)

[4.1.2.1 Freundlich 23](#_Toc48575885)

[4.1.2.2 Feindlich 25](#_Toc48575886)

[4.2 Crafting System 26](#_Toc48575887)

[4.3 Inventar System 26](#_Toc48575888)

[5. Ergebnisse und Zusammenfassung 27](#_Toc48575889)

[5.1 Ergebnisse 27](#_Toc48575890)

[5.2 Zusammenfassung 27](#_Toc48575891)

[Quellenverzeichnis 27](#_Toc48575892)

# Abbildungsverzeichnis

[Abbildung 1 Kingdom: Two Crowns Multiplayer () 7](#_Toc48581311)

[Abbildung 2 Kingdom: Two Crowns Angriff auf die Greed () 8](#_Toc48581312)

[Abbildung 3 Neuer Spiel Start in Terraria () 9](#_Toc48581313)

[Abbildung 4 World of Warcraft Charakter Editor () 10](#_Toc48581314)

[Abbildung 5 Beispielziel in Clonk Rage () 11](#_Toc48581315)

[Abbildung 6 4 Spieler Split-Screen in Clonk Rage () 12](#_Toc48581316)

[Abbildung 7 Kingdom: Two Crowns Pixelart () 13](#_Toc48581317)

[Abbildung 8 斩妖行 Eastern Exorcist Grafikstil () 13](#_Toc48581318)

[Abbildung 9 Ori and the Will of the Wisps Grafikstil () 14](#_Toc48581319)

[Abbildung 10 Minecraft Crafting im Inventar () 15](#_Toc48581320)

[Abbildung 11 Minecraft Crafting in der Werkbank () 15](#_Toc48581321)

[Abbildung 12 Minecraft das neue Rezeptbuch () 16](#_Toc48581322)

[Abbildung 13 Crafting in Terraria () 17](#_Toc48581323)

[Abbildung 14 Profession in World of Warcraft und sein Fortschritt 17](file:///F:\Bachelor\Bachelor\Schriftlich\Major%20Project.docx#_Toc48581324)

[Abbildung 15 Neue Profession Rezepte lernen in World of Warcraft () 18](file:///F:\Bachelor\Bachelor\Schriftlich\Major%20Project.docx#_Toc48581325)

[Abbildung 16 Craften von Gebäuden in Clonk Rage () 18](#_Toc48581326)

[Abbildung 17 Craften in der Werkstatt () 19](#_Toc48581327)

[Abbildung 18 Flowchart NPC Logik () 24](file:///F:\Bachelor\Bachelor\Schriftlich\Major%20Project.docx#_Toc48581328)

[Abbildung 19 Flowchart Enemy Logik () 25](file:///F:\Bachelor\Bachelor\Schriftlich\Major%20Project.docx#_Toc48581329)

[Abbildung 20 Inventar Taskleiste () 27](#_Toc48581330)

# 1. Einleitung

## Idee des Medienprojekts

Das Projekt soll eine Mischung aus Spielen sein, um ein neues einzelnes Spiel zu erschaffen. Dabei wird das meiste aus dem Spiel Kingdom: Two Crowns genommen. Die Idee ist es, die Mechaniken aus diesem Spiel zu nehmen und mit weiteren zu erweitern. Das heißt, es wird ein 2D Strategiespiel, in dem man als König seines Landes freie Bürger anwerben kann, um sein Königreich zu verteidigen. Dabei soll das Spiel noch mit einem Crafting System erweitert werden. Dies soll dafür sorgen, dass man die Ausrüstung seiner eigenen Truppen aufwerten und sich selbst eine Ausrüstung craften kann, um an der Seite der Truppen das Königreich zu verteidigen.

## Zielsetzung

Das Projekt soll ein 2D Crafting und Strategieplattformer sein. Die Qualität des Ergebnisses soll ein spielbares fertiges Produkt darstellen, auf gleicher oder ähnlicher Qualitätsstufe wie Kingdom: New Lands oder Kingdom: Two Crowns. Anders als bei Kingdom: Two Crowns ist kein Multiplayer Modus zu dem Zeitpunkt eingeplant. Das Projekt soll sich stark an der Kingdom-Reihe anlehnen. Das heißt, es werden Core-Features, wie das Erwerben von freien Bürgern, mithilfe einer Währung ermöglicht. Mit diesen wird dann ein Königreich gebaut und sich gegen die gegnerischen Streitmächte verteidigt. Das Projekt soll aber auch eigene Features beinhalten, wie zum Beispiel: Man kann von dem Reittier absteigen und selbst mit seinen Untertanen gegen die gegnerischen Streitmächte kämpfen. Man kann auch mithilfe eines Crafting-Systems sich die dafür benötigte Ausrüstung erstellen und weiter verbessern, um so den feindlichen Kräften Einhalt zu gebieten. Auch soll man seinen derzeitigen Spielstand speichern und wieder Laden können, so dass es zu keinem Verlust des derzeitigen Spielstandes kommt. Anders als bei der Kingdom Reihe soll kein Pixel Art Stil genutzt werden, sondern soll in dem Stil ähnlich wie „斩妖行 Eastern Exorcist“( Wildfire Game（无锡野火数字科技有限公司）2020: o.S.). Zu dem derzeitigen Stand ist keine Veröffentlichung des praktischen Projekts auf Seiten wie Steam oder dem Epic Games Store geplant.

## Verortung in der Industrie

Das Projekt könnte auf der Gamescom gezeigt werden, um Spieler auf dieses aufmerksam zu machen. Diese könnten dann auf der Gamescom eine spielbare Version des Projektes spielen und mögliche Interessenten begeistern. Man könnte mit dem Projekt auf der SAE Alumni Convention und dem Deutschen Entwicklerpreis teilnehmen. Da das Projekt primär die Zielgruppe von Kingdom: Two Crowns ist, das Projekt sich sehr stark an diesem Spiel orientiert und Kernaspekte und Elemente aus Kingdom: Two Crowns übernimmt, sollten die Erwartungen der Zielgruppe und Spielern von Kingdom: Two Crowns entsprechen. Die Zielgruppe von Kingdom: Two Crowns sind Spieler im Alter von 6 Jahren bis 30 Jahren. Konkret sind es Spieler, die strategische und Pixel 2D Spiele spielen. Bezüglich der Nische des geplanten Spiels lassen sich wenige vergleichbare Spiele finden, unter anderem „Regions of Ruin“(GameClaw 2018: o.S.), „Life is Hard“(Priozhok Studio 2015: o.S.) und „Forsaken Realm“(Forester Games 2020: o.S.). Die genannten Spiele haben mit dem geplanten folgende gemeinsam: Alle basieren auf der Kingdom Reihe und haben Aspekte wie: Das Verteidigen des Königreichs vor feindlichen Einheiten, das Erwerben von Truppen, das Aufbauen des Königreichs und sind auch im 2D und Pixel Art Stil gehalten. Jedoch unterscheidet sich das geplante Projekt in der Kombination aus einem Crafting-System zur Verbesserung der eigenen Ausrüstung und dass der Spieler in der Lage ist die Truppen direkt im Gefecht zu unterstützen.

# 2. Kontext

## 2.1 Referenzen

## 2.1.1 Technische Referenzen

### 2.1.1.1 Kingdom: Two Crowns

Kingdom: Two Crowns ist das dritte Spiel in der Kingdom Reihe vom Entwickler Fury Studios. Das erste Spiel der Reihe war Kingdom: Classic zuerst Veröffentlicht am 21 Oktober 2015. Kingdom: Two Crowns wurde am 11 Dezember 2018 auf Steam veröffentlicht. Kingdom: Two Crowns ist wie seine vorherigen Ableger ein 2D Strategie Plattformer in dem man sein Königreich vor den gegnerischen Streitmächten Verteidigen. Anders in Kingdom: Two Crowns ist allerdings, dass dieses Spiel auch im Multiplayer mit einem weiteren Spieler/in gespielt werden kann. Der Multiplayer ist als lokaler Multiplayer an einem Computer, als auch als online Multiplayer verfügbar. Anders ist auch das Kingdom: Two Crowns das einzige Spiel der Reihe ist, für den zwei DLCs (Downloadable Content) verfügbar sind (vgl. Fury Studios 2018: o.S.). Downloadable Content sind Erweiterungen für das Spiel, die mehr Spielinhalte für das Spiel bringen sollen nach dessen Veröffentlichung.

Abbildung 1 Kingdom: Two Crowns Multiplayer ()

Das Ziel des Spiels ist es die gegnerischen Streitkräfte, die Greed, aus dem eigenen Reich zu vertreiben, um Friedlich in diesem leben zu können. Dabei muss man als Spieler auf mehreren Inseln diese Vertreiben, um Endgültig von ihnen befreit zu sein. Dabei muss man Mauern und eine Armee aufbauen um sich vor den Greed zu schützen. Um sich eine Armee aufzubauen muss man aus Zeltlagern *Herumstreicher* mithilfe von Gold zu Bürgern anwerben. Danach muss man dann für diese Bögen kaufen, damit diese dann zu Jägern werden. Am Tag jagen sie Tiere um Gold zu generieren für den Spieler, in der Nacht gehen sie hinter die Mauer und Verteidigen diese vor den anstürmenden Greed. Jäger können auch Türme besetzen welche man an bestimmten Stellen im Spiel bauen kann. Um Mauern oder auch Türme bauen zu können muss man Hammer, anstatt Bögen für die Bauern kaufen. Diese kümmern sich dann um den Aufbau von Türmen und Wälle. Wälle die von den Greed beschädigt werden wieder von diesen Repariert. Wird ein Wall allerdings niedergerissen muss der Spieler wieder Gold Investieren um den Wall wieder aufbauen zu können. Im weiteren Spielverlauf können die Bauern dann auch Katapulte bedienen um in der Nacht, mit diesen die Greed zu bekämpfen. Um die Greed aus der Insel zu vertreiben, muss der Spieler eine Bombe kaufen und einen Gegenangriff gegen die Greed starten. (vgl. Fury Studios 2018: o.S.)

Abbildung 2 Kingdom: Two Crowns Angriff auf die Greed ()

### 2.1.1.2 Minecraft

Minecraft ist ein 3D Open-World, Survival und Aufbauspiel von Mojang. Minecraft ist für die Plattformen Linux, Mac OS, Andorid, IOS, Windows Phone, XBOX, Playstation und die Switch verfügbar (vgl. Mojang o.J.: o.S.).

Minecraft kann sowohl im Single- als auch Multiplayer mit mehreren Spielern gespielt werden. Dabei hat Minecraft kein richtiges Ziel. Jeder Spieler setzt sich sein eigenes Ziel. Auf dieses Ziel arbeitet dann jeder Spieler hin. So kann ein Ziel des Spielers sein eine sehr große Stadt aufzubauen, oder auch einfach alle Ressourcen zu sammeln und ein großes Minensystem aufzubauen. Minecraft hat keine Einschränkung in die Zielsetzung eines jeden einzelnen in dem Spiel (vgl. Mojang o.J.: o.S.).

### 2.1.1.3 Terraria

Terraria ist ein 2D Open World, Survival und Crafting Spiel, das am 16 Mai 2011 von dem Entwickler Re-Logic für Windows PCs erschien. Terraria ist sowohl als Single- und auch als Multiplayer Spiel spielbar. Das Ziel in Terraria ist gleich wie in Minecraft, der Spieler setzt sich selbst die Ziele. Terraria gibt aber auch Ziel Anstöße in dem man in dem Spiel Bosse besiegen kann. Dafür braucht man bestimmte Ausrüstung um das zu schaffen. Je weiter man kommt desto schwerer werden die Bosse und umso bessere Ausrüstung wird benötigt um diese zu besiegen (vgl. Re-Logic o.J.: o.S.).

Abbildung 3 Neuer Spiel Start in Terraria ()

### 2.1.1.4 World of Warcraft

World of Warcraft ist ein MMROPG (Massive Multiplayer Online Role-Playing Game), bedeutet das eine Spieler Anzahl von 3500 bis 4000gleichzeitig in einem Spiel spielen (vgl. Scholz 2019: o.S.), welches 2004 für den Amerikanischen und 2005 für den Europäischen Markt erschien. Für World of Warcraft sind bereits 7 Erweiterungen erschienen. In diesen Erweiterungen wurden jeweils neue Klassen, Rassen und Gebiete dem Spiel hinzugefügt. Auch wurde das Höchstlevel, das man erreichen kann zu jeder Erweiterung erhöht (vgl. Blizzard Entertainment 2016: o.S.). In World of Warcraft kann man sich einen Charakter erstellen. Dabei wählt man aus verschiedenen Rassen und dem gewünschten Geschlecht eine Rasse aus (vgl. Blizzard Entertainment o.J.: o.S.). Danach wählt man eine Klasse aus welche der Charakter haben soll. Wichtig dabei ist allerding, dass nicht jede Rasse jede Klasse nehmen kann. Das liegt unteranderem an der Herkunft der jeweiligen Rassen und ihrer Geschichte in der Spielwelt zusammen (vgl. Blizzard Entertainment o.J.: o.S.).

Abbildung 4 World of Warcraft Charakter Editor ()

Nachdem man seine Rasse, Klasse und Geschlecht ausgewählt hat, kann man seinen Character noch Individuell anpassen. Man kann das Gesicht, die Gesichtsbehaarung, die Frisur, die Frisur Farbe und die Hautfarbe zwischen mehreren vorgefertigten auswählen und so seinen Character Individuell anpassen wie man in Abbildung 5 sehen kann.

### 2.1.1.5 Clonk Rage

Clonk Rage ist der dritte Ableger der Clonk Reihe. Clonk Rage erschien erstmals im Jahr 2008 von dem Entwickler RedWolf Design für Windows, Linux und Mac (vgl. RedWolf Design o.J.: o.S.).

Clonk Rage ist ein 2D Strategie, Aufbau, Wirtschaft, Kampf und Geschicklichkeit Plattformer. Das Ziel in Clonk Rage kann nicht so einfach definiert werden, da die jeweiligen Ziele an dem geraden gespielten Level abhängen. So kann es sein das man eine bestimmte Anzahl an Wirtschaftspunkten sammeln muss, oder man muss als einziges Team noch in dem Level überleben um dies zu Gewinnen. Manche Level und Karten haben aber auch kein vordefiniertes Ziel, so kann sich der Spieler dann selbst überlegen welches Ziel er erreichen möchte. Ein Beispiel Ziel für ein Level in Clonk Rage ist in Abbildung 6 zu sehen. 

Abbildung 5 Beispielziel in Clonk Rage ()

Clonk Rage kann sowohl im Single- und Multiplayer spielbar, unteranderem ist Clonk Rage auch an einem Computer im Split-Screen mit bis zu vier Personen spielbar. Ein Split-Screen teilt den gesamten Monitor anhand der Spieleranzahl an einem Computer auf. Und gibt jedem Spieler dann seinen eigenen Bereich in dem er spielen kann wie zu sehen in Abbildung 7.

Abbildung 6 4 Spieler Split-Screen in Clonk Rage ()

## 2.1.2 Stilistische Referenzen

### 2.1.2.1 Kingdom: Two Crowns

Die Grafik in Kingdom: Two Crowns ist im Pixelart Stil gehalten (vgl. Fury Studios 2018: o.S.). Das bedeutet, dass die Struktur von Charakter, Umgebung, Hintergrund und Details dargestellt werden in Pixelform. Dieser Grafikstil wird zahlreich bei 2D Plattformern genutzt, da größtenteils 2D Plattformern meistens durch Indie Studios entwickelt werden. Indie Studios sind Entwicklerstudios, die eine kleinere Anzahl an Personen im Team haben und nicht ein sehr hohes Budget zur Verfügung haben. Der Pixel Art Stil hat für gewöhnlich einen nicht so hohen Aufwand wie ein gezeichneter Stil.

Abbildung 7 Kingdom: Two Crowns Pixelart ()

### 2.1.2.2 斩妖行 Eastern Exorcist

Eastern Exorcist ist ein 2D RPG Plattformer der 2020 für den PC und die Playstation 4 veröffentlicht werden soll. Eastern Exorcist ist nicht wie andere 2D Titel im Pixelart Stil gehalten, sondern in einem eher gezeichneten Stil, welche das Spiel von der Mehrheit differenziert (vgl. Wildfire Game（无锡野火数字科技有限公司）2020: o.S.).

Abbildung 8 斩妖行 Eastern Exorcist Grafikstil ()

### 2.1.2.3 Ori and the Will of the Wisps

Ori and the Will of the Wisps hat ähnlich wie Eastern Exorcist einen gezeichneten Grafikstil, welches auch diesen Spieltitel von der Mehrheit der 2D Plattformer differenziert und so ein Alleinstellungsmerkmal bildet (vgl. Moon Studios 2020: o.S.). 

Abbildung 9 Ori and the Will of the Wisps Grafikstil ()

## 2.2 Vorstellung Spezieller Techniken

## 2.2.1 Crafting

### 2.2.1.1 Crafting Grundlagen

Crafting ins Deutsche Übersetzt Basteln, ist eine Spielfunktion die es dem Spieler ermöglicht das Herstellen oder Basteln von Gegenständen, oder bauen von Gebäuden, Häusern oder ähnlich Vergleichbaren. So kann der Spieler mithilfe dieser Spielfunktion Werkzeuge, Gebäude, Maschinen und viel mehr Herstellen, ohne Kosten von Materialien aus der realen Welt (vgl. Dickinson et aliter 2017: o.S.).

### 2.2.1.2 Minecraft

Crafting in Minecraft ist in zwei Bereichen möglich. Einmal kann man im eigenen Inventar in einem 2x2 großen Feld Craften. In diesem kann man allerdings nur kleinere Gegenstände herstellen, wie zum Beispiel: Aus Holz werden Planken gemacht, Planken können weiter zu Stöcken weiterverarbeitet werden, die Herstellung von Fackeln und viel mehr.



Abbildung 10 Minecraft Crafting im Inventar ()

Um Größere Werkzeuge oder Gegenstände craften zu können braucht man eine Werkbank, welche man in dem eigenen Crafting Feld im Inventar herstellen kann. Mit der Werkbank ist man dann in der Lage in einem 3x3 Feld Werkzeuge, Rüstung, Türen, Boote und viel mehr zu craften. Für verschiedene Werkzeug Stufen werden verschiedene Materialien benötigt um diese herzustellen wie zum Beispiel für eine Eisen Spitzhacke werden drei Eisenbarren benötigt.



Abbildung 11 Minecraft Crafting in der Werkbank ()

Um diese zu bekommen muss man diese in einer Höhle oder einer eigen angelegten Miene fördern. Erze müssen dabei dann in einem Ofen zu Barren geschmolzen werden.

Vor der Minecraft Version 1.12 musste man um neue Rezepte herauszufinden, diese mit den verschiedenen Materialien ausprobieren oder diese im Internet heraussuchen. Mit der Minecraft Version 1.12 wurde das Rezeptbuch hinzugefügt. Eine neue Spielfunktion die es dem Spieler ermöglicht, nach erlangen von neuen Materialen, werden die diesbezüglichen Rezepte in dieses Buch hinzugefügt (vgl. Bergenstein 2017: o.S.). Das Rezeptbuch mitsamt verfügbaren Rezepten, ist in Abbildung 13 zu sehen.

Abbildung 12 Minecraft das neue Rezeptbuch ()

### 2.2.1.3 Terraria

Das Craften in Terraria ist ähnlich zu dem Crafting in Minecraft. In Terraria ist es möglich in seinem Inventar Gegenstände mit den nötigen Ressourcen zu craften. Wenn man größere Sachen Craften möchte, muss man sich eine Werkbank herstellen. Um allerdings sich Waffen und Rüstung zu Craften ist ein Amboss notwendig. Dort werden dann die Metalle zu dem beliebigen Ausrüstungsgegenstand geschmiedet. Um Erze zu Barren zu craften wird auch ein Ofen benötigt, in diesem werden dann Barren zu Erzen geschmolzen. Um im späteren Spielverlauf bessere Erze zu Barren zu schmelzen, muss man den Ofen zu einer besseren Stufe Aufrüsten müssen. Sobald man eine neue Ressource erlangt wird dem Spieler Crafting Rezepte angezeigt die er, sofern alle verfügbaren Ressourcen vorhanden sind craften kann. Es ist also nicht notwendig diese Ressourcen in der Werkbank in ein bestimmtes Muster legen zu müssen (vgl. Re-Logic o.J.: o.S.).



Abbildung 13 Crafting in Terraria ()

### 2.2.1.4 World of Warcraft

In World of Warcraft wird das Crafting System Profession genannt. Diese Professionen werden als Jobs im Spiel gesehen, die der Spieler ausüben kann. Dabei gibt es verschiedene Profession Stufen, die wichtigsten dabei sind die Primäre- und Sekundäre Profession. Der Spieler kann in diesen zwischen verschiedenen Jobs wählen die er ausüben möchte. Dabei kann man nur zwei Primäre Professionen lernen, da diese erlernte auch wieder verlernt werden können. Dies sorgt dafür, dass man noch einmal eine andere Profession nimmt als für die man sich anfangs entschied. Wenn man nun eine Primäre Profession verlernt geht der Gesamt gelernte Fortschritt verloren. Wenn man diese also später nochmal lernen möchte, muss man also den Fortschritt in der Profession wieder von vorne beginnen. Sekundäre Professionen können nicht verlernt werden, da es dort kein Limit gibt wie viele man von ihnen lernen kann. Fortschritt in der Profession sammelt man, indem man Gegenstände in dieser herstellt. Je mehr Gegenstände in dieser Hergestellt werden umso höher wird nach einer gewissen Zeit die Fähigkeitsstufe der Profession. Um eine Profession zu lernen oder neue Gegenstände in dieser zu erlangen, muss man zu einem Profession Lehrer gehen. Dort kann man dann mithilfe der im Spiel vorhandenen Währung neue Rezepte Lernen.

Abbildung 14 Profession in World of Warcraft und sein Fortschritt

### 

Abbildung 15 Neue Profession Rezepte lernen in World of Warcraft ()

### 2.2.1.5 Clonk Rage

Clonk Rage hat zwei verschiedene Möglichkeiten des Crafting. Einmal das Craften von Gegenständen und das Craften von vorgefertigten Gebäude Bauplänen. Für diese Gebäude Baupläne muss man einen Baukasten nutzen um dieses Craften zu können. Diesen Baukasten kann man mithilfe der in dem Spiel vorhandenen Währung kaufen oder auch an einem Amboss craften. 

Abbildung 16 Craften von Gebäuden in Clonk Rage ()

In diesen Gebäuden kann man die Rohstoffe dann zu anderen Gegenständen weiterverarbeiten. So wir für Holz zum Beispiel ein Sägewerk benötigt um aus Bäumen Holz craften zu können. Eine wichtige Ressource die ebenfalls benötigt wird für viele Gebäude ist Strom. Gebäude wie das Sägewerk oder die Werkstatt benötigen Strom um arbeiten zu können. Strom kann zum Beispiel aus einem Windrad gewonnen werden. Um Strom von dem Windrad in ein Gebäude zu bekommen werden Leitungen benötigt. Diesen kann sowie auch den Baukasten entweder kaufen oder an einem Amboss herstellen. In einer Werkstatt der Strom hat kann man dann verschiedene Gegenstände herstellen zum Beispiel: Brücken, Luftschiffe, Züge, etc.



Abbildung 17 Craften in der Werkstatt ()

Zu Beginn des Spiels hat man eine bestimmte Auswahl an Gebäuden und Gegenstände die man herstellen kann. Um weitere Baupläne für Gebäude und Gegenstände zu erhalten muss man in einem Forschungszentrum diese erforschen. In Clonk Rage ist es nicht erforderlich die Ressourcen in ein bestimmtes Muster zu legen, dafür werden nur Ressourcen benötigt. Danach wird der Spieler dann diesen Gegenstand oder auch das Gebäude anfangen zu craften.

# 3. Methodik

## 3.1 Projektumsetzung

Das Projekt soll wie im Kapitel 1.2 Zielsetzung beschrieben primär die Mechaniken von dem Spiel Kingdom: Two Crowns übernehmen. Anders als Kingdom: Two Crowns soll allerdings sein, dass der Spieler nicht auf einem Reittier sitzt. Der Spieler soll stattdessen normal rumlaufen können. Nichtsdestotrotz soll der Spieler in der Lage sein, die gegnerischen Truppen selber angreifen zu können. Dies soll der Spieler entweder durch eine Nahkampf oder Fernkampf Waffe machen. Das Projekt soll außerdem einen vorgefertigten Character beinhalten. Es soll nicht möglich sein, dass der Spieler seinen eigenen Character erstellen kann wie in dem Spiel World of Warcraft. Der Spieler soll nicht die Möglichkeit haben einen eigenen Character erstellen zu können, weil es den Aufwand des Projektes dramatisch erhöhen würde, außerdem liegt der Fokus in dem Projekt mehr auf die Umsetzung eines Crafting Systems als die Umsetzung eines Character Creator. Anders als in Minecraft soll der Spieler nicht frei Bauen können. Der Primäre Fokus des Spiels liegt auf der Verteidigung des Königreichs vor den feindlichen Einheiten, daher würde es eher weniger Sinn machen dem Spieler die Möglichkeit zu geben frei bauen zu können. Wenn der Spieler eine Mauer aufbauen würde, müsste man erstmal wissen mit welchen Materialen er diese aufbaut. Danach müsste man überlegen wie man die Stärke der Mauer berechnet, würde man das anhand der Anzahl der Gesetzten Blöcke machen oder wie hoch die Mauer ist und wie breit. Diese Freiheit die man dem Spieler geben würde, würde allerdings auch die Berechnung erschweren wie Stark eine Mauer ist und wie viele Treffer diese braucht um zerstört zu werden. Außerdem würde es das Upgraden der Mauer erschweren, dafür müsste der Spieler erstmal diese wieder abbauen und durch neue Blöcke ersetzen. Auch könnte durch die breiteren oder höheren Mauern die Berechnung der Flugbahn etwas erschwert werden. So könnten die Truppen die hinter der Mauer stehen gar nicht von feindlichen Pfeilen getroffen werden, da diese zu hoch oder zu breit ist. Das Projekt soll ein Level mit einem Ziel haben. Anders als in Clonk Rage wo es mehrere Level gibt mit verschiedenen Zielen. Verschiedene Ziele würden dem Grundprinzip von Kingdom: Two Crowns nicht entsprechen, wäre aber eine gute Möglichkeit sich von dem Grundspiel weiter zu differenzieren und ein weiteres Alleinstellungsmerkmal zu erlangen.

### 3.1.1 Crafting System

Das Crafting System worum es sich im Projekt primär handelt, soll ein simples und sehr übersichtliches Design aufweisen. Dies soll dadurch ermöglicht werden, dass der Spieler die Ressourcen nicht in ein Rezept legen soll wie in Minecraft. Hier ist eher die Terraria Variante angestrebt, der Spieler soll also vordefinierte Rezepte haben. Anders als in dem Spiel World of Warcraft soll der Spieler nicht mehrere Trainer haben bei dem er seine Rezepte durch eine Währung lernen kann. Dieser soll eher durch craften das neue Rezept lernen. Wenn er dann genug Ressourcen hat, soll er in der Lage sein das neue Rezept craften zu können. Auch soll der Spieler keine expliziten Gebäude craften können und aufstellen können wie in Clonk Rage.

### 3.1.2 Engine

Das Projekt Kingdom and Glory wird in der Spiel Engine Unity umgesetzt. Unity ist eine von Unity Technologies entwickelte Spiel Engine, die es Nutzern ermöglichen soll, mit dieser eigene Spiele entwickeln zu können. Für das Projekt wird die Unity Personal Edition gewählt, da kein kommerzieller Vertrieb geplant ist und diese Edition das erstellen von 2D spielen ermöglicht ist diese ausreichend.Eine Alternative zu Unity wäre die Spiel Engine Godot Engine gewesen. Die Godot Engine nutzt die Programmiersprache C++ im Gegensatz zu Unity die auf C# setzt. Die Godot Engine könnte eingesetzt werden, da dies allerdings eine komplett neue Engine wäre in der man noch nicht gearbeitet hat und sich in diese intensiv einarbeiten müsste um ein solches Projekt umsetzen zu können.

## 3.2 Meilensteinplanung

### 3.2.1 Meilensteindefinierung

Um die Features im Projekt umzusetzen werden diese in verschiedenen Milestones gesetzt. Die Features wurden in fünf Meilensteine eingeteilt. Die ersten Meilensteine sollen Grundfunktionen des Spiels beinhalten. Nachdem diese dann Implementiert sind und funktionieren soll das Crafting-System in das Projekt implementiert werden. Ich habe mich für diese Aufteilung entschieden, da erst die Grundlegenden Mechaniken Implementiert werden sollen und danach Spezifische Funktionen die für das Projekt wichtig sind.

### 3.2.2 Meilenstein 1

In diesem Meilenstein soll die Grundlegende Bewegung fertiggestellt werden. Des Weiteren soll in diesem Meilenstein der Spieler in der Lage sein, Münzen aufzusammeln. Auch sollen Vagrant an bestimmten Stellen und bestimmter Anzahl spawnen.

### 3.2.3 Meilenstein 2

Im zweiten Meilenstein soll die Funktion hinzukommen, die Vagrant zu Bauern zu rekrutieren. Auch soll dann die Funktion hinzukommen den Bauern einen Job zuweisen zu können, sowie grundlegende Funktionen für ihre Jobs. So sollen die Jäger jagen. Dafür wird eine Funktion sorgen die Tiere spawnen lässt, welche die Jäger dann jagen können um ein Einkommen zu generieren. Die Funktion für die Handwerker soll erst in dem nächsten Meilenstein implementiert werden.

### 3.2.4 Meilenstein 3

Im dritten Meilenstein sollen die Funktionen hinzukommen das man Mauern aufbauen, reparieren, sowie upgraden kann. Das fügt dann auch die Job Funktion für die Handwerker hinzu, welche für die Jäger schon im zweiten Meilenstein hinzugekommen ist. Diese Aufteilung soll gemacht werden, um in einem Meilenstein nicht zu viele Aufgaben zu haben und, nicht den Fokus an diesen zu verlieren. Es soll damit ermöglicht werden die Zeitplanung einzuhalten um besser an den Aufgaben im Meilenstein zu arbeiten ohne in einen Zeitdruck zu kommen, weil zu viele Aufgaben in einen Meilenstein gepackt worden sind.

### 3.2.5 Meilenstein 4

In dem vorletzten Meilenstein soll das Crafting System für das Projekt erstellt werden. Das Crafting System ist die einzige Aufgabe für diesen Meilenstein, da dieses für das Projekt sehr wichtig ist und somit der gesamte Fokus auf diesem liegen soll.

### 3.2.6 Meilenstein 5

Im letzten Meilenstein soll das Inventar erstellt werden, in diesem sollen dann Rohstoffe die für das Crafting-System benötigt werden gesammelt werden. Des Weiteren sollen in diesem Meilenstein weitere Kleinigkeiten ausgebessert und Poliert. Das polieren soll erst in dem fünften Meilenstein erfolgen, damit vorher der Fokus auf die eigentlichen Umsetzungen der Funktionen liegt und sich darauf konzentriert werden kann.

# 4. Durchführung

## 4.1 Grundlegende Mechaniken

### 4.1.1 Player Character

Als erstes wurde die Bewegung des Spielers implementiert. Hierbei traten keine größeren Fehler auf. Nachdem die Bewegung des Spielers implementiert wurde, wurde die Funktion implementiert das der Spieler münzen hinlegen und einsammeln kann. Nachdem dies implementiert war, wurde die Mechanik implementiert, dass der Spieler mithilfe der Münzen Wälle bezahlen kann um diese aufzubauen. Dabei trat allerdings der Fehler auf, dass die Münzen die der Spieler für das Gebäude bezahlte, trotzdem fallen gelassen worden sind. Der Spieler bezahlte zwar das Gebäude für 3 Münzen, konnte allerdings eine Münze wieder aufsammeln und so ein Teil des Preises wieder zu erhalten. Um diesen Fehler richtig zu beheben, musste erst einmal die Interaktion mit Gebäuden implementiert werden. Danach konnte der Fehler behoben werden, indem der Spieler nun ein UI für das Gebäude aufruft, wenn er mit diesem Interagiert. Dort kann dieser den Preis für das Gebäude bezahlen. Dies ermöglicht auch, dass der Spieler in der Lage ist nicht nur Münzen, sondern auch andere Rohstoffe für das Gebäude zu bezahlen. Somit ist der Spieler nun auch in der Lage, für den freundlichen NPC einen Hammer oder Bogen zu kaufen. Danach wurde die Mechanik implementiert das der Spieler mit einem Borgen schießen kann. Hierfür wurde sich die „Wurfparabel“ und der „Schräger Wurf“ angeschaut, mithilfe dieser kann die Flugbahn des Pfeils berechnet werden.

### 4.1.2 NPCs

### 4.1.2.1 Freundlich

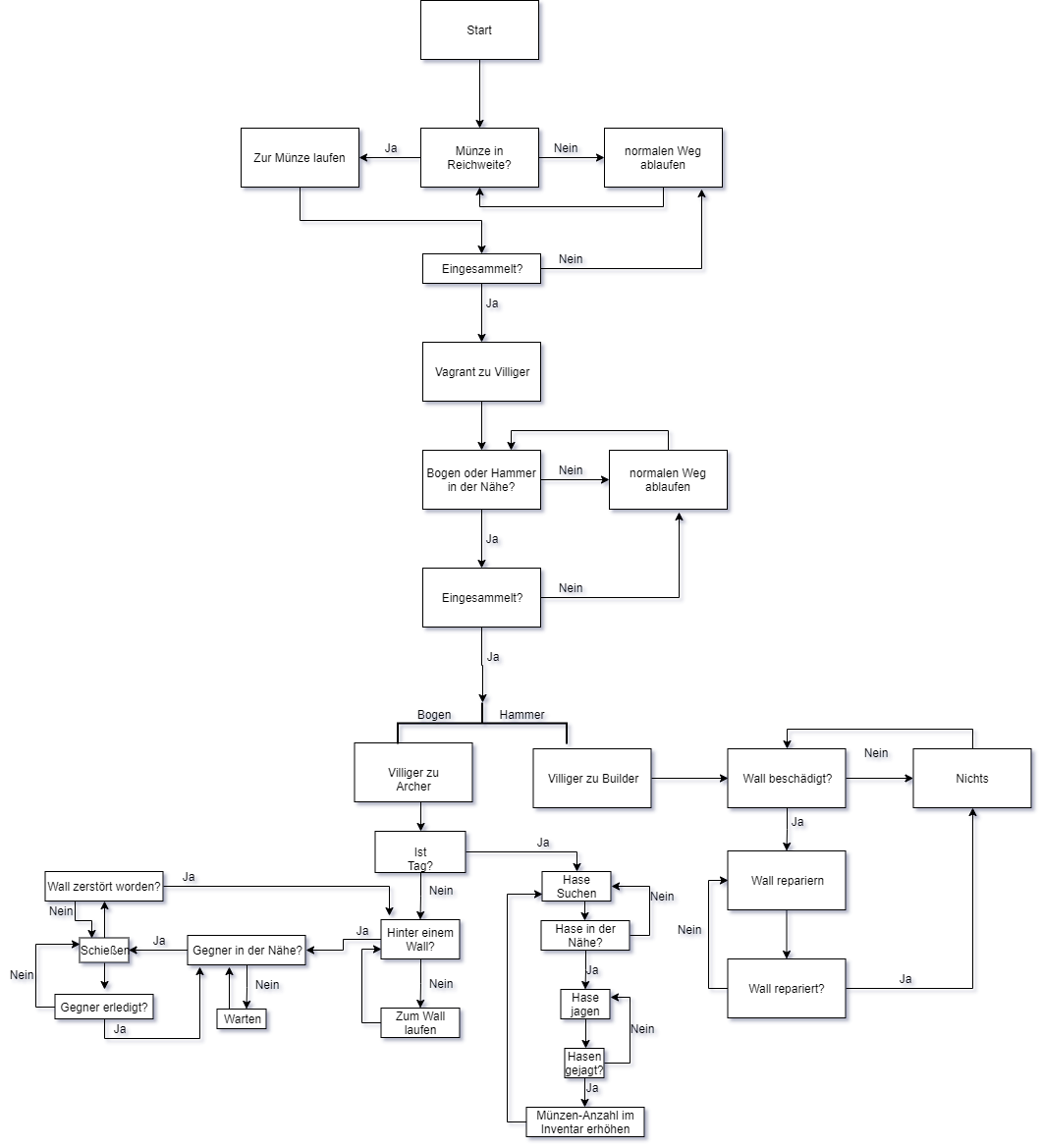
Für den verbündeten NPC musste als aller erstes die Bewegung implementiert werden. Als erstes wurde sich dafür ein A\* Skript genommen. Mithilfe dieses Skripts konnte die Bewegung ausgelagert werden und musste nicht weiter zeit darauf verwendet werden. Dieses A\* Skript erstellte ein Grid und Analysierte wo der NPC langlaufen konnte. Danach berechnet er einen Weg zu seinem Ziel und diesen läuft der NPC dann entlang. Bei dieser Implementierung kam allerdings der Fehler auf, dass der NPC sobald er am Ende eines Weges angekommen war eine Fehlermeldung Endstand. Dies und ein weiteres Hindernis im Zusammenhang mit dem A\* Skript (das im Kapitel 4.1.2.2 beschrieben wird) wurde dieses wieder verworfen. Daraufhin wurde ein eigenes System entwickelt, mithilfe dieses Systems konnte die Bewegung des NPCs implementiert werden. Nach dieser Implementierung wurde das generelle Verhalten des verbündeten NPCs entwickelt. Im ersten Schritt wurde das Verhalten implementiert, wenn der NPC noch ein „Vagrant“ ist. Das „Vagrant“ verhalten soll solange passieren bis die Spieler diesen mithilfe einer Münze rekrutiert, danach soll der NPC sein Verhalten ändern. Die erste Überlegung das Verhalten zu wechseln war es, dass der NPC wieder gelöscht und ein neuer an der gleichen Stelle erstellt wird. Diese Herangehensweise würde sich allerdings auf Dauer als zu Rechenaufwendig erweisen. Stattdessen sollte der NPC einfach seinen State wechseln, hierfür wurde eine State Maschine implementiert. Diese beinhaltet die Verschiedenen States die der NPC annehmen kann. Diese Wären: der „Vagrant“ (State bevor der NPC rekrutiert wird), der „Villiger“ (State, wenn der NPC eine Münze einsammelt), der „Archer“ (State, wenn der NPC einen Bogen einsammelt) und der „Builder“ (State, wenn der NPC einen Hammer einsammelt). In diesen States sollen der NPC verschiedene Aufgaben erledigen. Im „Vagrant“ State soll dieser zwischen bestimmten Wegpunkten herlaufen. Wenn sich in der Nähe eine Münze befindet, soll dieser sich auf die Münze zubewegen und diese einsammeln. Im „Archer“ State soll dieser nach Hasen suchen. Wenn der NPC in der Nähe eines Hasen steht, soll dieser auf den Hasen schießen. Solange es Tag ist soll dieser Hasen suchen und jagen. Während der Nacht soll der NPC hinter den Mauern stehen und diese vor den gegnerischen NPCs Verteidigen. Eine genauere Theoretische Logik des NPC ist in Abbildung 18 zu sehen.

Abbildung Flowchart NPC Logik ()

### 4.1.2.2 Feindlich

Um den gegnerischen NPC zum Laufen zu bringen, wurde wie vorher bei dem freundlichen NPC die Bewegung mithilfe eines A\* Skripts implementiert. Der gegnerische NPC sollte dabei ein eigenes Grid nutzen. Dies sollte dem gegnerischen NPC die Bewegung durch Wälle nicht ermöglichen. Dabei trat aber der Fehler auf das neue Grid dem NPC zuzuweisen. Um allerdings an diesem Problem nicht zu viel Zeit zu investieren, wurde dies wieder verworfen und eine andere Herangehensweise wurde gewählt. Die neue Herangehensweise war das dieser zu Beginn nach allen Wällen, NPCs und dem Spieler selber sucht und zu Listen hinzufügt. Danach wird aus den Listen, den am nähest liegenden Wall und NPC gesucht. Im nächsten Schritt wird aus dem nähest liegenden Wall, NPC und dem Spieler das generell am nähest liegende Ziel ausgesucht. Auf dieses Ziel bewegt der NPC dann zu. Wenn er dann in einer Nähe von 1 Meter bis 4 Meter zum Ziel steht, fängt dieser an auf das Ziel zu schießen. Dieser Abstandsdifferenz wurde gewählt, damit der Spieler besser erahnen kann wie viele Gegner angreifen. Sollte das angegriffene Ziel in der Zwischenzweit zerstört worden sein so wird das Ziel aus der Liste der Wälle oder NPCs gelöscht werden, danach sucht der gegnerische NPC sich ein neues Ziel. Sollte allerdings in der Zwischenzeit die Nacht vorbei gegangen und der Tag herangebrochen, so werden die gegnerischen NPCs zerstört. Zur verbesserten Veranschaulichung dient die Abbildung 19, in dieser sieht man den Flowchart zu dem gegnerischen NPC und dessen Theoretischen Logik.

Abbildung Flowchart Enemy Logik ()

## 4.2 Crafting System

Für das Crafting System musste als aller erstes überlegen werden, wie der Spieler das ganze Crafting System aufrufen sollte. Dabei wurde überlegt ob der Spieler am beliebigen Ort mithilfe des Inventar craften kann oder hierfür eine Werkbank aufrufen muss. Gewählt wurde letzteres. Nachdem diese gemacht worden ist, musste das Crafting System ausgearbeitet werden. Der erste Fokus lag auf dem UI des Crafting Systems. Die erste Überlegung war es, dass der Spieler mehrere Auswahlmöglichkeiten haben soll was genau er craften möchte. Dabei soll er auf einen Button für ein bestimmtes Item drücken um dieses auszuwählen. *Danach soll der Spieler dann auf einen weiteren Button drücken um den ausgewählten Gegenstand zu craften*. Da dieses UI aber nicht als Intuitiv erachtet worden ist, wurde diese Idee wieder verworfen. Die zweite und finale Überlegung war ähnlich zu der ersten. *Der Spieler hat mehrere Auswahlmöglichkeiten zwischen verschiedenen Gegenständen die er craften kann*. Anders dabei war, dass der Spieler nun über den Button des Gegenstandes craftet. *Ob der Spieler den Gegenstand craften kann, wird diesem Farblich angezeigt mit der Farbe Rot (nicht genug Ressourcen), Grün (genug Ressourcen) und Grau (komplett Aufgewertet)*. Nach dem UI wurde die Interaktion mit der Schmiede implementiert. Dabei trat der Fehler auf, dass der Spieler das UI nicht mehr mit der gleichen Taste zum Aufrufen dies schließen konnte. *Dies wurde aber behoben so dass der Spieler mit der gleichen Taste das UI auch wieder schließen kann*. Danach wurde dann die eigentliche Logik des Crafting Systems umgesetzt. Hierfür wurde als erstes überlegt wie viele Ressourcen man für bestimmte Ausrüstungen benötigt um diese craften zu können. Daraufhin folgten die Überlegungen der verschiedenen Wertigkeitsstufen der Ausrüstungen und dessen benötigte Werte um diese craften zu können. Als dies dann feststand, wurden im Skript die Abfragen des aktuellen Ressourcen Standes und der benötigten Ressourcen implementiert. Dabei sollte der Ressourcen-Stand durchgehend aktualisiert werden, sprich der Spieler sollte nicht neu mit der Schmiede interagieren müssen damit die neuen Werte der Ressourcen im Crafting System übernommen wurden. Danach folgte dann die Implementierung der Buttons und der „CraftItem“ Funktion. Die „CraftItem“ Funktion sollte dann aufgerufen werden, wenn der Spieler auf den Button drückt. Danach sollte der aktuelle Gegenstand übergeben werden den man gecraftet hat. Wenn der Spieler genügend Ressourcen für den Gegenstand besaß, wurde dieser entweder neu gecraftet oder aufgewertet. Dabei wurden dann die benötigten Ressourcen von dem aktuellen Ressourcen-Stand abgezogen und der Wert für die nächste Gegenstandsstufe wurde erhöht.

## 4.3 Inventar System

Zu Beginn der Arbeit an dem Projekt, wurde das Inventar in das Player Skript implementiert. Anfangs gab es auch kein großes Problem damit. Als dies dann aber anfing komplexer zu werden durch die verschiedenen Rohstoff Arten, wurde das Inventar ausgelagert und als eigenständiges Skript im Projekt implementiert. In diesem wurden dann die Rohstoffe und die Ausrüstung gespeichert. Diese Aufteilung sorgte auch dafür, das andere Skripts nicht mehr so oft auf das Player Skript zugreifen mussten. *So wird nun das Inventar Skript aufgerufen und nicht mehr das Player Skript, wenn das Crafting System zum Beispiel den aktuellen Rohstoff stand wissen möchte*. Um den Spieler anzuzeigen wie viel er am Rohstoffe im Inventar hat, wurde an einer Grafischen Darstellung im Spiel gearbeitet. Dabei kam die Idee das Inventar in einer Taskleiste anzuzeigen. In dieser Taskleiste wird die aktuelle Menge eines Rohstoffes angezeigt. Um das ganze übersichtlich zu machen wird der Wert neben dem entsprechenden Rohstoff angezeigt. Wie das Ganze im Spiel aussieht sieht man in der Abbildung 20.

Abbildung 20 Inventar Taskleiste ()

# 5. Ergebnisse und Zusammenfassung

## 5.1 Ergebnisse

-

## 5.2 Zusammenfassung

Fazit:

Ausblick:

- Verwendung von mehr Ressourcen und somit mehr Upgrade Stufen.

- Mehr NPCs -> NPCs Ausrüstung auch Upgraden.

- Mehr Gegner und stärkere Gegner je länger das Spiel geht

- Verschiedene Gebiete mit verschiedenen Bedingungen

- Wettereinflüsse und Jahreszeiten

- Andere Szenarien wie: So lange überleben wie es geht

- Multiplayer -> Zusammen oder Gegeneinander

# Quellenverzeichnis