BACHELORARBEIT

zur Erlangung des akademischen Grades

„Bachelor of Science in Engineering“ im Studiengang Informatik/Computer Science

Wie Lootboxen in Videospielen das Suchtverhalten von Spielern beeinflussen können

Ausgeführt von: Oliver Müller

Personenkennzeichen: 2110257017

1. BegutachterIn: Titel Vorname Name, Titel

Wien, 30.April 2024

Eidesstattliche Erklärung

„Ich, als Autor / als Autorin und Urheber / Urheberin der vorliegenden Arbeit, bestätige mit meiner Unterschrift die Kenntnisnahme der einschlägigen urheber- und hochschulrechtlichen Bestimmungen (vgl. Urheberrechtsgesetz idgF sowie Satzungsteil Studienrechtliche Bestimmungen / Prüfungsordnung der FH Technikum Wien idgF).

Ich erkläre hiermit, dass ich die vorliegende Arbeit selbständig und nach den aktuell geltenden Regeln der FH Technikum Wien angefertigt und dass ich Gedankengut jeglicher Art aus fremden sowie selbst verfassten Quellen zur Gänze zitiert habe. Ich bin mir bei Nachweis fehlender Eigen- und Selbstständigkeit sowie dem Nachweis eines Vorsatzes zur Erschleichung einer positiven Beurteilung dieser Arbeit der Konsequenzen bewusst, die von der Studiengangsleitung ausgesprochen werden können (vgl. Satzungsteil Studienrechtliche Bestimmungen / Prüfungsordnung der FH Technikum Wien idgF).

Weiters bestätige ich, dass ich die vorliegende Arbeit bis dato nicht veröffentlicht und weder in gleicher noch in ähnlicher Form einer anderen Prüfungsbehörde vorgelegt habe. Ich versichere, dass die abgegebene Version jener im Uploadtool entspricht.“

|  |  |  |
| --- | --- | --- |
|  |  |  |
| Ort, Datum |  | Digitale Unterschrift |

Kurzfassung

Text

Abstract

**Schlagwörter:** Schlagwort1, Schlagwort2, Schlagwort3, Schlagwort4, Schlagwort5

**Keywords:** Keyword1, Keyword2, Keyword3, Keyword4, Keyword5

Inhaltsverzeichnis

[1 Einleitung 7](#_Toc165560638)

[1.1 Motivation 9](#_Toc165560639)

[1.2 Aufgaben 10](#_Toc165560640)

[2 Methodik 11](#_Toc165560641)

[2.1 Entwicklung der Spielbasis? 11](#_Toc165560642)

[2.2 Entwicklungsumgebung und Programmiersprache? 12](#_Toc165560643)

[2.3 Auswahl passender Mechaniken/Game Patterns 13](#_Toc165560644)

[2.3.1 Balancing des Spiels 13](#_Toc165560645)

[2.3.2 Dark Patterns 14](#_Toc165560646)

[2.3.3 Existierendes Spiel als Referenz 15](#_Toc165560647)

[2.3.4 Blog über das Design von Free2Play Spielen als Referenz 16](#_Toc165560648)

[2.3.5 Gewählte Patterns 16](#_Toc165560649)

[2.4 Datensammlung 17](#_Toc165560650)

[2.5 Gewählte Tools 17](#_Toc165560651)

[3 Lösung 18](#_Toc165560652)

[3.1 Anforderungen 18](#_Toc165560653)

[3.1.1 Basisspiel 18](#_Toc165560654)

[3.1.2 Lootbox-Mechanik 18](#_Toc165560655)

[3.1.3 Premium-Shop 18](#_Toc165560656)

[3.1.4 Rangliste 19](#_Toc165560657)

[3.2 Funktionalität 19](#_Toc165560658)

[3.2.1 Basisspiel / User Interface 19](#_Toc165560659)

[3.2.2 Lootbox-Mechanik 20](#_Toc165560660)

[3.2.3 Premium Shop 22](#_Toc165560661)

[3.2.4 Rangliste 24](#_Toc165560662)

[3.3 Benutzeroberfläche 25](#_Toc165560663)

[3.3.1 Basisspiel / User Interface 25](#_Toc165560664)

[3.3.2 Lootbox-Mechanik 28](#_Toc165560665)

[3.3.3 Premium-Shop 29](#_Toc165560666)

[3.3.4 Rangliste 31](#_Toc165560667)

[3.4 Softwarearchitektur 32](#_Toc165560668)

[3.4.1 Grundgerüst 32](#_Toc165560669)

[3.4.2 Lootbox-Mechanik 32](#_Toc165560670)

[3.4.3 Umgesetzte Patterns 35](#_Toc165560671)

[4 Diskussion (900 W.) 40](#_Toc165560672)

[Literaturverzeichnis 41](#_Toc165560673)

[Abbildungsverzeichnis 43](#_Toc165560674)

[Tabellenverzeichnis 44](#_Toc165560675)

[Abkürzungsverzeichnis 45](#_Toc165560676)

[Anhang A: Überschrift des ersten Anhangs 46](#_Toc165560677)

[Anhang B: Überschrift des zweiten Anhangs 47](#_Toc165560678)

# Einleitung

Im letzten Jahrzehnt wurde die Implementierung von Lootboxen (deutsch: Beutekiste) zur vielfach angewandten Praxis, da diese eine zusätzliche nicht unerhebliche Einnahmequelle bedeutet. Activision Blizzard erzielte laut Statista im Jahr 2022 5,886 Milliarden U.S. Dollar über In-Game Käufe, Abonnements und andere Einnahmen verglichen mit 1,642 Milliarden in Produktverkäufen [1]. Electronic Arts wiederum generierte 2022 4,277 Milliarden U.S. Dollar allein durch den Verkauf von Zusatzinhalten [2].

Vergleicht man Daten von Computer- und Videospiel-Verkäufen mit In-Game- und In-App-Käufen aus Deutschland, zeichnet sich ein ähnliches Bild. Gegenüber dem Jahr 2022 ist im Jahr 2023 der Umsatz mit den zuletzt genannten Käufen um 6% von 4,458 Millionen € auf 4,742 Millionen € gestiegen. Der Umsatzanteil mit In-Game- und In-App-Käufen liegt damit bei über 80% des Gesamtumsatzes (1) [3].

Ein Bild, das Text, Screenshot, Schrift, Design enthält.

Automatisch generierte Beschreibung

Abbildung 1: Umsatz mit Computer- und Videospielen in Deutschland

Die Lootbox-Mechanik, die oft als Form von Glücksspiel bezeichnet wird, ist aber nicht ohne Kontroversen. Spielende können mit dem Einsatz von Echtgeld Lootboxen kaufen, die eine definierte Anzahl zufälliger virtueller Gebrauchsgegenstände mit unterschiedlichen Seltenheitsgraden enthalten. Erst nach dem Kauf und dem Öffnen wird bekannt, was darin enthalten war. Anfangs war es nicht üblich die Wahrscheinlichkeiten der verschiedenen Gegenstände anzugeben (zum Beispiel: 60% Normal, 30% Selten, 9% Episch, 1% Legendär).

Im Jahr 2019 forderten drei große Hersteller von Videospielkonsolen, dass alle Spiele auf ihren Plattformen die „Drop-Rate“ (Fund Rate) der Lootboxen offenlegen müssen [3].

Seitdem ist dies gängige Praxis in fast allen Videospielen mit Lootbox-Mechaniken. Diese willkommene Veränderung in der Transparenz hat aber leider keinen Einfluss auf die potenzielle Suchtgefahr, denn die Spannung, vielleicht aus der nächsten Box einen legendären Gegenstand zu erhalten, bleibt.

Denn gerade die Seltenheit macht Gegenstände erst wertvoll und begehrenswert.

Selbst wenn ein Spiel keine besonders interessante Geschichte hat oder aufwändig programmiert ist, kann eine Implementierung von Lootboxen zu einem Erfolg führen. Vorausgesetzt, sie sind gut umgesetzt und motivierend [4].

In manchen Spielen sind die Inhalte von Lootboxen nicht spielentscheidend. Das heißt, dass Spielende sich durch deren Kauf keinen unfairen Vorteil verschaffen können (zum Beispiel kosmetische Gegenstände). Bei anderen bekannten Spielen, wie beispielsweise FIFA Ultimate Team, ist es durch den Einsatz von Echtgeld sehr wohl möglich sich einen Vorteil zu erkaufen, da in den Lootboxen unterschiedlich starke und seltene Fußballspieler\*innen enthalten sind. Letzteres wird in der Branche und Gamingwelt auch als „pay to win“ bezeichnet.

Neben Lootboxen finden auch andere Arten von „Mikrotransaktionen“ in Videospielen Verwendung, die es ermöglichen sollen, im Spiel selbst weitere Inhalte wie DLCs (Downloadable Content), Premiumwährung, Beschleunigungen oder Kosmetische Gegenstände zu kaufen. DLCs sind beispielsweise meist größere Erweiterungen des Basisspiels mit einer neuen Geschichte und neuen Orten, die erkundet werden können.

Das Thema Lootboxen, aber auch andere Monetarisierungsmethoden, sind in Videospielen ist in den letzten Jahren häufiger in den Schlagzeilen. Diese werden mit Glücksspielmechanismen verglichen, da immer mehr Menschen, vor allem Jugendliche und junge Erwachsene, Verhaltensweisen entwickeln, die einer Sucht gleichkommen. Verglichen werden kann hier die Ungewissheit. Auf der einen Seite im Glückspiel, ob bei der nächsten Runde endlich der große Geldgewinn dabei ist und auf der anderen die Lootbox, bei der hoffentlich der lang ersehnte seltene und wertvolle Gegenstand enthalten ist. Auch die Inszenierung lässt sich vergleichen, denn hier wird auf beiden Seiten mit Musik, bunten Lichtern und Effekten gearbeitet, um das Öffnen möglichst spannend zu gestalten [5].

Diese strukturellen und psychologischen Ähnlichkeiten führen potenziell zu problematischem Glücksspielverhalten. Während die Spieleindustrie solche Kritiken zurückweist, Lootboxen als „Überraschungsmechanik“ bezeichnet und mit dem Öffnen von Sammelkartenspielpackungen vergleicht, bleibt das Potenzial dieser Mechanismen, dafür anfällige Spielende anzuziehen und ernsthafte Probleme zu verursachen, ein umstrittenes und moralisch komplexes Thema [6].

## Motivation

Werden Lootboxen in ein Spiel integriert, sind das meistens Free2Play-Spiele. Solche Spiele können grundsätzlich gratis gespielt werden, allerdings werden diese dann anderweitig monetarisiert. Eben entweder durch Lootboxen oder andere Angebote für Mikrotransaktionen. Diese werden, wie in „Monetarisierung von Computerspielen“ von Koubek beschrieben, Teil einer sogenannten Game Loop (Spielkreislauf), welche immer wieder durchlaufen wird, damit Spielende motiviert bleiben [8]. Beispielsweise müssen produzierte Ressourcen täglich eingesammelt werden, mit diesen dann Gebäude immer weiter aufgewertet oder Krieger trainiert und verbessert werden. Diese Krieger werden dann für Kämpfe gegen andere Spielende verwendet, um deren Ressourcen zu stehlen und Punkte zu bekommen. Und die Implementation von Premiumwährungen wie Juwelen, welche hauptsächlich durch Echtgeldeinsatz erhältlich sind, werden dann für die Verkürzung von Wartezeiten verwendet.

Dadurch kann nicht eindeutig gesagt werden, ob bei dieser Art Spiel allein Lootboxen eine Sucht maßgeblich begünstigen oder ob andere Mechaniken der Monetarisierung ebenfalls eine tragende Rolle spielen.

Während der Recherche nach geeigneten Studien und Artikeln, wurden ausschließlich Auswertungen von Umfragen gefunden, die unter Personen durchgeführt wurden, die allgemein Videospiele spielen. Als Beispiel sei hier die Metastudie „Characteristics of Gamers who Purchase Loot Box: a Systematic Literature Review” genannt, die 20 Studien auf die Zusammenhänge zwischen Lootboxen, Gaming, Glücksspiel und anderen Merkmalen untersucht hat [7]. Der Artikel befasst sich mit den psychosozialen Merkmalen von Personen, die Echtgeld für Lootboxen in Videospielen ausgeben. Es wird ein positiver Zusammenhang zwischen dem Kauf von Lootboxen und Merkmalen der Internetspielsucht sowie Glücksspielstörungen festgestellt. Die Autoren untersuchen auch, wie Stimmung, Geschlecht, physiologischer Zustand und Motivation mit dem Kauf von Lootboxen zusammenhängen.

Ein limitierender Faktor war, dass die Qualität der einbezogenen Studien generell niedrig war, was größtenteils auf die Selbstberichte der Befragten zurückzuführen ist. Um die methodische Qualität zu erhöhen, könnten beispielsweise Protokolle von Käufen und Spielzeiten verwendet werden, um objektivere und genauere Messungen durchführen zu können [7].

Aus diesem Grund wäre eine Testung und Umfrage mit einem eigens entwickelten Spiel, welches genauere Daten liefern kann (wie Spielzeit, ausgegebenes Echtgeld), ein neuartiger und praxisorientierterer Zugang zu dieser Thematik. Dadurch könnten potenziell neue Lösungen entwickelt werden, um solche Mechaniken zu entschärfen und um Personen, die anfällig für eine Spielsucht sind, besser helfen zu können.

## Aufgaben

Das primäre Ziel dieser Arbeit ist es zu untersuchen, welchen Einfluss Lootboxen in Bezug auf das Spielverhalten von Spielenden haben. Im speziellen wird analysiert, ob und in welchem Ausmaß Lootboxen die Spieldauer und die Spielintensität beeinflussen. Außerdem soll erforscht werden, wie die Integration von Lootboxen die Bereitschaft der Spielenden beeinflusst, Geld für In-game-Transaktionen auszugeben. Ziel ist es auch zu verstehen, ob die Anziehungskraft der Lootbox-Mechanik möglicherweise stärker ist als das eigentliche Spielerlebnis selbst.

1. Welche Rolle spielen Lootboxen bei der Veränderung des Spielverhaltens hinsichtlich Spieldauer und der Spielintensität der Spieler?
2. Wie beeinflusst die Integration von Lootboxen in Videospielen die Bereitschaft der Spielenden, Geld für In-Game-Transaktionen auszugeben?

Das Ziel ist, ein einfaches und leicht zugängliches Spiel mit einer simplen Lootbox-Mechanik und Möglichkeiten von Mikrotransaktionen, sowie kompetitiven Elementen auszustatten. Dieses Spiel soll dann von verschiedenen Personen getestet werden. Im Anschluss beantworten die Testpersonen einen Fragebogen zu den Erfahrungen und Eindrücken des Spiels sowie generell zu deren Erfahrung mit Lootboxen und Glücksspielen in Videospielen.

# Methodik

Die fertige Lösung soll ein einfaches Spiel mit simpler Steuerung und moderatem Inhalt mit einer Lootbox Mechanik sein. Da so eine Mechanik nicht für sich allein steht soll das Spiel auch einen Shop beinhalten der Premiumwährung für „Echt“-Geld anbietet. Zusätzlich sollen auch andere relevante Inhalte im Spiel um diese Währung gekauft werden können, seien es Beschleunigungen oder Extraenergie. Der oder die Testpersonen sollen das Spiel schnell starten können und die Steuerung und das Spielziel möglichst sofort erfasst haben.

## Entwicklung der Spielbasis?

Zu Beginn stand die Frage im Raum, auf welcher Basis die Lootbox-Mechanik entwickelt werden soll. Ein eigenes Spiel von Grund auf zu Programmieren bedeutet sehr viel Aufwand, hier wären aber keine Lizenzprobleme vorhanden. Ebenfalls in Betracht gezogen wurde die Möglichkeit eine Modifikation für ein vorhandenes Spiel zu entwickeln wie beispielsweise Minecraft. Hier eröffnet sich allerdings das Problem, dass dieses Spiel frei zugänglich sein muss, damit die Testpersonen sich nicht erst ein Spiel kaufen müssen. Zuletzt wäre es auch möglich, vorhandene frei zugängliche Projekt-Samples zu nehmen, die beispielsweise im Marktplatz von Spiele-Engines wie Unity und Unreal zu finden sind.

Alle drei Vorgehensweisen bedeuten einen recht großen Aufwand. Eine längeren Recherche nach einen passenden open source Spiel, zu dem eine Lootbox-Mechanik passt und zusätzlich leicht modifizierbar ist, blieb erfolglos. Auch die Suche nach geeigneten Projekt-Samples stellte sich als schwierig heraus. Es wurde zwar ein passendes RPG-Toolkit gefunden, jedoch hätte die Einarbeitung in dieses Toolkit und zusätzlich in die Engine viel Zeitbeansprucht.

Übrig geblieben ist die Entwicklung eines eigenen Spiels von Grund auf. Um den Arbeitsaufwand zu minimieren und auf die Einarbeitung in die Spiele-Engine und das Programmieren der Funktionen zu beschränken soll auf lizenzfreie grafische Assets und bei Bedarf Musik und Geräusche zurückgegriffen werden.

## Entwicklungsumgebung und Programmiersprache?

Der nächste Schritt war, eine geeignete Plattform für die Entwicklung auszuwählen. In der engeren Auswahl standen Unity, Unreal, Godot und Pygame.

Unity und Unreal sind mächtige Werkezuge die auch in der Spieleindustrie die häufigste Anwendung finden. Dementsprechend hoch kann die Einstiegshürde sein. In Unity wird mit C# während in Unreal mit C++ programmiert wird. Unreal bietet jedoch Blueprints an, die den Einstieg gerade für Anfänger stark vereinfachen sollen. Dabei ist es möglich, Gameplay-Elemente über ein visuelles Skriptsystem zu erstellen und zu modifizieren. Verschiedene Knoten können mit Funktionen und Aktionen ausgestattet werden, welche dann wiederum miteinander verbunden werden können.

Diese Funktion erleichtert zwar die Implementierung einzelner Mechaniken, der Rest des Spiels muss aber trotzdem programmiert werden.

Pygame ist eine Python Bibliothek, welche bereits weit entwickelt wurde und unter Hobby- und Indie-Entwicklern beliebt ist. Sie ist open source und nicht die einzige Python Bibliothek, die für die Spieleentwicklung geeignet ist. Der Einstieg ist hier zwar denkbar schnell, da in jeder Entwicklungsumgebung, in der Python funktioniert, auch programmiert werden kann, jedoch ist die Handhabung nicht unbedingt Einsteigerfreundlich.

Letztendlich fiel die Wahl auf die open source Spiel-Engine Godot. Godot wird stetig weiterentwickelt und ist mittlerweile sehr stabil und benutzerfreundlich geworden. Die aktuelle Version ist 4.2.1, in der auch das Spiel für die Bachelorarbeit programmiert werden soll. Programmiert wird in der hauseigenen Skriptsprache GDScript, wobei aber auch C# verwendet werden kann. Zu GDScript und Godot werden umfangreiche Dokumentationen bereitgestellt und auf diversen Plattformen wie Github und Youtube finden sich viele Inhalte, um schnell an Informationen zu kommen.

## Auswahl passender Mechaniken/Game Patterns

In diesem Kapitel wird besprochen, welche Mechaniken und Game Patterns später im Spiel implementiert werden sollen. Anhand von Beispielen und Erklärungen soll an die Auswahl herangeführt werden.

### Balancing des Spiels

Damit das fertige Spiel auch motivierend ist, ist es wichtig auf das richtige Balancing in der Spielerfahrung zu achten, wie in dem Buch „Monetarisierung in Computerspielen“ beschrieben [8].

Dabei ist unter anderem auf Fairness zu achten, damit alle Spielenden für die Herausforderungen des Spiels genügend Ressourcen zur Verfügung haben. Auch soll es weder zu schwer noch zu leicht sein. Die Balance zwischen Können vs. Glück, Verstand (Rätsel) vs. Geschicklichkeit (Reaktionsvermögen), Kooperation vs. Konkurrenz, Kurze vs. Lange Spieldauer, Belohnung und Strafen ist ebenfalls wichtig zu beachten.

Solche Methoden werden für die Verbesserung der Spielerfahrung herangezogen und immer weiter verfeinert. Gerade bei Free2Play Spielen ist es üblich, regelmäßige Updates mit Änderungen am Balancing durchzuführen, um die Spielerfahrung möglichst vieler Spielenden zu verbessern. Dabei ist aber auch eine gewisse Zusammenarbeit und ein Vertrauen zwischen Entwickler bzw. Game Designer und den Spielenden nötig.

Sämtliche Methoden des Balancings werden zusammengenommen auch Design Patterns genannt.

### Dark Patterns

Zu den genannten Design Patterns haben sich in den letzten Jahren auch Dark Patterns oder Dark Game Design Patterns einen Namen gemacht. Hierbei geht es um gezieltes Herbeiführen von negativen Erlebnissen im Spiel, um Spielende dazu zu bringen, Geld auszugeben, um diese in positive zu wandeln. Einige dieser Muster werden ebenfalls im Buch „Monetarisierung in Computerspielen“ genannt [8].

Beispiele:

* **Wait to Play:** Spielende werden willkürlich warten gelassen. (Mit steigendem Level und Spielfortschritt werden Wartezeiten länger.)
* **Playing by Appointment:** Die Spielzeit wird vom Spiel definiert und entscheidet nicht mehr der Spielende. (Regelmäßiges anmelden, um den Spielfortschritt zu sichern.)
* **Daily Rewards:** Das tägliche Anmelden wird belohnt und das Fernbleiben bestraft.
* **Grinding:** Nur das regelmäßige aktive Spielen bringt spürbaren Fortschritt.
* **Can’t Pause or Save:** Spiele werden auf zentralen Servern verwaltet und Spielende haben keine Chance ihr Spiel zu speichern oder zu pausieren
* **Infinite Treadmill:** Es gibt keinen Endzustand, Spielfortschritt und das Ranking in der Rangliste können endlos weitergeführt werden.
* **Premium Currency:** Das Einführen einer Premium Spielwährung beispielsweise als Juwelen, welche zwar auch in kleinen Mengen erspielbar ist, allerdings hauptsächlich gegen Echtgeld angeboten wird
* **Pay to Skip:** Meist in Kombination mit einer Premiumwährung lässt sich damit Wartezeiten verkürzen oder ganz auflösen
* **Pay to Win:** Spielerische Vorteile durch den Kauf von stärkerer Ausrüstung oder mächtigeren Einheiten durch Premiumwährung.
* **Artifical Scarcity:** Eine künstliche Verknappung von Inhalten die nur kurzzeitig oder in Sonderangeboten erhältlich sind. (Fear of missing out – FOMO)

Beispiele zu Dark Social Patterns:

* **Competition:** Der Wettbewerb mit anderen Spielern wird durch Einsatz von Zeit und Geld erleichtert.
* **Social Obligation:** Spielende fühlen sich ihrer Community/ihrem Clan verpflichtet und wollen nicht enttäuschen.
* **Fear of Missing Out:** Die Angst, etwas zu verpassen, seien es Events oder geplante Unternehmungen des Clans.
* **Endowed Value:** Der erarbeitete Spielfortschritt, eventuell auch durch bereits ausgegebenes Echtgeld unterstützt, wird nicht einfach so aufgegeben.

Oft werden manche der genannten Dark Patterns gar nicht als negativ empfunden. Gerade die sozialen Mechaniken wie der Wettbewerb zwischen Spielenden und Clans wird oft als positive intensive Erfahrung gesehen.

Dark Patterns finden allerdings nicht nur in der Spieleentwicklung ihren Platz, sondern auch in sozialen Netzwerken oder im Marketing.

### Existierendes Spiel als Referenz

Best Gaem Studio hat am 15.12. 2015 einen Pay2Win Simulator mit dem Namen „Pay 2 Win: The Tricks Exposed“ veröffentlicht, der die Tricks von Monetarisierung in Videospielen anhand der Spielerfahrung offen anspricht und erklärt [8].

Ein Bild, das Text, Screenshot, Website, Onlinewerbung enthält.

Automatisch generierte Beschreibung

Abbildung 2: Screenshot des Spiels aus der Vertriebsplattform Steam

Die Inhalte dieses Simulators sollen als Idee und Referenz für das eigens entwickelte Spiel herangezogen werden. Unter anderem die Möglichkeit durch mehrmaliges Klicken auf einen Button je 40 Cent zu erhalten. Die Testpersonen geben also kein echtes Geld aus, sondern können fiktives Echtgeld selbst generieren, welches wiederum im Spiel für diverse Inhalte ausgeben werden kann.

### Blog über das Design von Free2Play Spielen als Referenz

Ein Blog mit dem Titel „The design of Free-to-play Games“ wird ebenfalls als Referenz für einige der Mechaniken verwendet. In diesem Blog-Eintrag geht es hauptsächlich um grundsätzliche Eigenschaften, die Free to Play Spiele besitzen, wie beispielsweise ein leichter und schneller Einstieg, eine leicht zu verwendende und verstehende Benutzeroberfläche mit wenigen Untermenüs und die Implementierung von Abläufen, die immer nur kurze Zeit gespielt werden, aber dafür sehr oft über einen langen Zeitraum. Sehr wichtig sind auch offene Schleifen, die Spieler immer wieder zurückholen, da sie das Gefühl bekommen sollen, immer noch etwas erledigen zu müssen [9].

Dabei bleibt aber das Ende offen, sodass sich Spielende in einer Endlosschleife befinden und das Spiel nicht einfach so beenden können. Sogenannte Endlosschliefen werden auch Game Loops genannt.

### Gewählte Patterns

Folgende Patterns wurden ausgewählt, um in das Spiel integriert zu werden. Dazu jeweils eine kurze Erklärung, wie diese ungefähr umgesetzt werden könnten.

* **Wait to Play:** Eine Komponente soll so umgesetzt werden, dass Testpersonen warten müssen. Eine in Auftrag gegebene Herstellung eines Gegenstandes dauert eine gewisse Anzahl Sekunden oder Minuten, ehe sie eingesammelt werden können.
* **Grinding:** Die Testpersonen sollen das Spiel regelmäßig oder über längere Zeiträume spielen müssen, um einen gewissen Fortschritt erreichen zu können.
* **Can’t Pause or Save:** Die Testpersonen sollen keine Möglichkeit haben, ihren Spielstand selbst zu speichern, wobei dieser aber im Hintergrund regelmäßig gespeichert werden soll.
* **Infinite Treadmill:** Das Spiel soll so aufgebaut sein, dass die Kernmechanik immer wieder wiederholt werden kann. Punkte, die man durch diese Mechanik sammelt, sollen in einer Rangliste mit simulierten Gegnern aufscheinen.
* **Premium Currency:**  Die Verwendung einer Premiumwährung soll ein zentraler Bestandteil sein, mit dem die Lootboxen aber auch andere Inhalte bezahlt werden können.
* **Pay to Skip:** Kommt ganz auf das Spiel an, das entwickelt werden soll. Eventuell passt hier auch stattdessen eine **Pay to Win** Mechanik besser.
* **Artifical Scarcity:** Ganz trivial als ein Sonderangebot für eine gewisse Zeit am Anfang des Spiels. Zum Beispiel Premiumwährung um den halben Preis.
* **Competition:** Die meisten Spiele mit einer Lootbox Mechanik sind kostenlos spielbar und auf Mehrspieler ausgelegt. Da die Entwicklung so eines Spiels aber zu aufwändig werden würde und auch das Testen erschwert, soll jeglicher Mehrspieleranteil nur simuliert werden, beispielsweise eine Rangliste, bei der durch die Darstellung anderer fiktiver Spielende eine Art Wettbewerb suggeriert wird. Die Testpersonen sollen so motiviert werden Zeit, Aufwand und im Idealfall auch fiktives Echtgeld zu investieren, um auf dieser Rangliste möglichst weit oben zu stehen, da dies auch mit besseren Gewinnen belohnt werden soll.

Die Implementierung verschiedener Mechaniken wie ein beschleunigter und leichter Start, der recht bald zu einer hohen Paywall führt, Vergünstigungen aber auch Bundles (günstigere Kombiangebote) sollen Testpersonen zu einem Kauf von Lootboxen verführen.

## Datensammlung

Während des Spiels sollen Daten gesammelt werden, unter anderem wie viel fiktives Echtgeld die Testpersonen generiert, wie viel sie davon ausgegeben und wie viel Zeit sie mit dem Spiel verbracht haben. Diese Daten sollen die Testpersonen am Ende bei einer kleinen Umfrage eingeben, damit diese ausgewertet werden können.

## Gewählte Tools

Als Entwicklungsumgebung wurde die Spiel-Engine Godot [10] [11] gewählt, da diese open source ist und die damit entwickelten Spiele nicht durch Lizenzen beschränkt sind. Das Spiel selbst wurde in der dazu gehörenden Programmiersprache GDScript geschrieben, da diese Sprache von den Entwicklern von Godot erstellt wurde, sehr gut dokumentiert ist und sich nahtlos in die Engine einfügt.

Es werden keine externen Bibliotheken und Frameworks benötigt, da die Engine bereits alles Benötigte beinhaltet.

Als Grafikprogramme wurden MS Paint und Affinity Designer verwendet. Da auf ein vorhandenes frei verfügbares Asset Pack für sämtliche Grafiken im Spiel zurückgegriffen wurde, wurden die genannten Programme nur für Anpassungen verwendet.

# Lösung

Daten zu Spiel:

Titel: Farm Island – A Lootbox Experience

Ansicht: 2D – Frontal

Grafikdesign: Pixel

Genre: Farmsimulation

Für die Grafiken wurde online nach Assets recherchiert und folgende ausgewählt:

Sprout Lands Asset Pack by Cup Nooble [12]

Sprout Lands UI Expansion by Cup Nooble [12]

In den folgenden Unterkapiteln werden zuerst die Anforderungen (3.1) für jeden relevanten Teil erklärt. Anschließend wird die genaue Ausführung textuell besprochen (3.2) und es werden Screenshots dieser Teile gezeigt und beschrieben (3.3).

Zuletzt wird in 3.4 die Softwarearchitektur, der für die Lootbox-Mechanik und Monetarisierung relevanten Teile, in Codeausschnitten veranschaulicht.

## Anforderungen

### Basisspiel

Mittels der „WASD“-Tasten oder den Pfeiltasten kann die Spielfigur in der Spielwelt bewegt werden. Mit der Maus und der „E“-Taste kann mit der Spielwelt interagiert werden. Die Spielwelt ist eine kleine Insel auf der eine eigene Hütte, Farmland, Vegetation und Nutztiere sowie der Shop eines NPC-Händlers zu finden sind. Die Samen aus Lootboxen werden auf Farmland angepflanzt. Diese wachsen über einen gewissen Zeitraum und können dann geerntet und im Shop verkauft werden.

### Lootbox-Mechanik

Mit der Spielwährung „Coins“ können unter anderem Lootboxen geöffnet werden, welche drei Exemplare von zufällig gewählten unterschiedlich seltenen Samen enthalten. Eine Premium-Lootbox mit doppelter Chance auf seltenere Samen kann mit der Premiumwährung gekauft werden.

### Premium-Shop

Im Premium-Shop können Premiumwährung, größere Farmflächen oder strukturelle Erweiterungen wie einen Brunnen (schnelleres Pflanzenwachstum) gekauft werden.

### Rangliste

Die Rangliste soll Spielende motivieren, mehr Zeit und bestenfalls Geld in das Spiel zu investieren, um auf den vordersten Plätzen zu sein. Sie zeigt den aktuellen Punktestand aller Spielenden. Punkte werden gesammelt durch den Verkauf der Feldfrüchte oder dem Kauf von Farmland und strukturellen Erweiterungen. Um den Mehrspieleraspekt und einen gewissen Wettbewerb zu simulieren, werden fiktive Namen und deren Punktestände hinzugefügt. Die Spielenden werden nur sehr schwer den ersten Platz erreichen und halten können. Wobei der Kauf von Premiumwährung mit Geld eine höhere Platzierung erleichtern soll.

## Funktionalität

Die Funktionalität wird anhand der Untermenüs in 3.1 erklärt.

### Basisspiel / User Interface

Für eine angenehmere Einführung in das Spiel wurde ein Startmenü designt (2). Nach drücken des Play-Buttons öffnet sich eine Textbox mit einer kurzen Begrüßung und der Möglichkeit, einen Usernamen einzugeben (3).

Bevor das Spiel selbst gespielt werden kann, wird noch die Steuerung eingeblendet (4) sowie eine kurze Spielanleitung „How to play“ als Bilderreihe (Screenshots des Spiels mit Beschriftung) (5). Diese Erklärung kann jederzeit im Pause-Menü unter „Controls“ eingesehen werden.

Direkt im Anschluss beginnt das eigentliche Spiel vor dem eigenen Haus. Davor befindet sich ein kleiner Fleck fruchtbare Erde, der zum Anpflanzen der Samen genutzt wird. In der linken oberen Ecke befindet sich die Übersicht über den derzeitigen Kontostand des fiktiven Echtgeldes, den bereits ausgegebenen Echtgeldbetrag und der „+“-Button, mit dem der Kontostand pro Mausklick um den Betrag 0,40€ erhöht werden kann.

Rechts oben informiert eine Übersicht über den aktuellen Stand der Spielwährung „Coins“ und Premium-Spielwährung „Silver“.

Am unteren Rand befindet sich die sogenannte „Hotbar“, zu der Items hinzugefügt werden können, um sie im Spiel zu verwenden (6).

Zu den Basisfunktionen zählt noch das Inventar, dass über die Taste „I“ aufgerufen werden kann. Anfangs zählt das Inventar noch 8 Plätze, die jedoch mit Premium-Spielwährung erweitert werden können. Über das Inventar werden Items wie Samen oder Gemüse der Hotbar zugewiesen. Ein kleines Fenster gibt auch Informationen über das Item selbst wie Name, Gruppe und Seltenheit (7). Eine Möglichkeit, Items fallen zu lassen ist ebenfalls vorhanden.

Wurden Samen der Hotbar zugewiesen, können diese mit den Tasten „1, 2, 3, 4 und 5“ ausgewählt werden und per linkem Mausklick auf fruchtbarer Erde gepflanzt werden (8).

### Lootbox-Mechanik

Für die Lootbox-Mechanik wurde ein eigener Premium-Shop entwickelt, der während des Spiels wie ein Menü mit der Taste „Q“ aufrufbar ist.

Es wurden zwei verschiedene Lootboxen erstellt, eine als Basisvariante die mit der normalen Spielwährung „Coins“ und eine Premiumvariante, die nur mit der Premium-Spielwährung „Silver“ käuflich ist. Der Unterschied der beiden Boxen ist, dass die Premiumvariante eine höhere Fundrate (Droprate) bei den selteneren Samen hat.

|  |  |
| --- | --- |
| Droprates (Fundraten) der  verschiedenen Seltenheitskategorien beider Lootboxen | |
| Seltenheitsstufe | Basis Lootbox | Premium Lootbox |
| 1 | 35% | 30% |
| 2 | 55% | 50% |
| 3 | 9.5% | 19% |
| 4 | 0.5% | 1% |

Tabelle 1: Lootboxen – Droprates/Fundraten

Die Unterteilungen wurden aufgrund eigener Erfahrungswerte in Free2Play Spielen mit Lootboxen inspiriert und gewählt. Übliche Raten sind gerade für die Kategorie der seltensten Gegenstände im Bereich von 0,2% bis 0,6%.

Dass die niedrigste Seltenheitsstufe nicht die höchste Rate aufweist, ist ebenfalls Absicht. Erhält man mit höherer Wahrscheinlichkeit einen selteneren, aber im Vergleich immer noch relativ wertlosen Gegenstand, verringert sich die Motivation weniger, neue Lootboxen zu öffnen.

Die Kosten, um eine Basis Lootbox zu öffnen beträgt 5 „Coins“ und bei der Premium Lootbox 5 „Silver“. Ist der Betrag verfügbar, wird nach einem Klick auf die entsprechende Lootbox der Mechanismus gestartet. Während sich die Box öffnet, leuchtet das Innere und hinter der Box fliegen weiße Partikel Sternförmig nach außen. Nach kurzer Zeit erscheinen die drei Items, welche per „Linksklick“ ausgehoben werden können. Verlässt der Spieler vorzeitig den Shop, werden die Items automatisch dem Inventar hinzugefügt.

Es wurden ebenfalls Lootboxen für die Spielwelt erstellt. Diese hat die gleiche Droprate wie die Premium Lootbox, der einzige Unterschied ist, dass die Items mit der Taste „E“ aufgehoben werden.

Insgesamt existieren im Spiel 14 verschiedene Samen, die in den Lootboxen enthalten sind:

|  |  |  |  |
| --- | --- | --- | --- |
| Samen | Seltenheit | Wert „Coins“ / „Silver“ | Punktewert |
| Cabbage | 1 | 1 „Coins“ | 1 |
| Wheat | 1 | 1 „Coins“ | 1 |
| Aubergine | 2 | 2 „Coins“ | 4 |
| Carrot | 2 | 2 „Coins“ | 4 |
| Corn | 2 | 2 „Coins“ | 4 |
| Cucumber | 2 | 2 „Coins“ | 4 |
| Tomato | 2 | 2 „Coins“ | 4 |
| White Radish | 2 | 2 „Coins“ | 4 |
| Artichocke | 3 | 4 „Coins“ | 20 |
| Cauliflower | 3 | 4 „Coins“ | 20 |
| Pumpkin | 3 | 4 „Coins“ | 20 |
| Purple Radish | 3 | 4 „Coins“ | 20 |
| Blue Poppy | 4 | 20 „Coins“ / 1 “Silver“ | 100 |
| Starfruit | 4 | 20 „Coins“ / 1 “Silver“ | 100 |

Tabelle 2: Lootbox-Mechanik Seltenheit und Wert der Samen

Die Seltenheitskategorie 2 beinhaltet die meisten Samen, damit die höchstmögliche Diversität in der Belohnung erreicht wird.

### Premium Shop

Der Premium Shop beinhaltet neben den Lootboxen auch andere kaufbare Inhalte wie Erweiterungen für Inventar und Farmland, Strukturen, eine Umtauschfunktion und den Kauf von Premium-Spielwährung anbietet.

Alle genannten Bereiche sind in einem Tab-Menü untergebracht. Das heißt, dass immer nur das aktive Tab angezeigt wird.

Je nachdem, welche Währung für die Erweiterungen benötigt werden wurde danach entschieden, welche Relevanz diese für das Spiel hat. Beispielsweise muss zwangsläufig mehr Farmland gekauft werden, um mithalten zu können, daher werden hier „Coins“ verlangt. Eine Inventarerweiterung oder der Kauf einer Struktur sind hingegen nur Erleichterungen für den Spielfluss, daher werden diese für „Silver“ angeboten.

**Inventory:**

Im Tab „Inventory“ kann das Inventar um weitere 4 Plätze erweitert werden. Anfangs bieten das Inventar 8 Plätze. Für jede weitere Stufe verdoppeln sich die Kosten.

|  |  |
| --- | --- |
| Plätze (maximal) | Kosten („Silver“) |
| 12 | 5 |
| 16 | 10 |
| 20 | 20 |
| 24 | 40 |
| 28 | 80 |
| 32 | 160 |

Tabelle 3: Premium Shop – Kosten der Inventarerweiterung

**Farmland:**

Im Tab „Farmland“ kann das verfügbare fruchtbare Farmland für vergrößert werden. Für jede gekaufte Stufe erhöht sich der Preis um das Zweifache.

|  |  |
| --- | --- |
| Stufe | Kosten („Coins“) |
| 1 | 25 |
| 2 | 50 |
| 3 | 100 |
| 4 | 200 |
| 5 | 400 |
| 6 | 800 |
| 7 | 1600 |

Tabelle 4: Premium Shop - Kosten der Farmlanderweiterung

Für jede Stufe bis einschließlich der 6. Stufe werden neue Farmland-Flächen in der Spielwelt generiert. Die letzte Stufe entfernt eine Absperrung für einen vordesignten Bereich, in dem sich eine größere Fläche Farmland befindet.

**Structure:**

Im Tab „Structure“ kann ein Brunnen um 50 „Silver“ gekauft werden, welcher in der Spielwelt generiert wird und das Pflanzenwachstum um 20% beschleunigt.

**Exchange:**

Im Tab „Exchange“ kann die Premiumwährung „Silver“ jederzeit gegen “Coins“ 1:5 getauscht werden.

**Silver Ingots:**

Im Tab „Silver Ingots“ kann das generierte fiktive Echtgeld für „Silver“-Pakete ausgegeben werden. Dies ist eine in der Spieleindustrie übliche Praktik, um Free2Play-Spiele zu monetarisieren. Die Preise variieren sehr stark zwischen den Spielen und reichen manchmal für einzelne Pakete von 0,99€ bis zu 999€.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Paket | Kosten | Grundmenge | Multiplikator | Bonus | Finale Menge |
| Tiny Pack | 1,99€ | 5 | - | - | 5 „Silver“ |
| Small Pack | 7,99€ | 20 | 4 | 10% | 22 „Silver“ |
| Medium Pack | 19,99€ | 50 | 2,5 | 10% | 55 „Silver“ |
| Large Pack | 49,99€ | 125 | 2,5 | 10% | 138 „Silver“ |

Tabelle 5: Premium Shop - Kosten der "Silver" Pakete

Erklärung der Berechnung:

Für das „Tiny Pack“ erhält man für 1,99€ 5 „Silver“. Für das nächsthöhere Paket „Small Pack“ wurde mit dem Multiplikator 4 gerechnet. 1,99€ \* 4 ergibt ungefähr die 7,99€ und 5 \* 4 ergibt 20 „Silver“. Diese Berechnung setzt sich für die beiden letzten Pakete mit dem Multiplikator 2,5 fort. Für alle Pakete außer dem „Tiny Pack“ erhält man zusätzlich 10% Bonus auf das „Silver“, das wären beim „Small Pack“ also 2 für insgesamt 22 Stück.

Diese Art von Bonus hat sich in der Spieleindustrie durchgesetzt, da sie Spielende dazu verleitet, mehr Geld auszugeben, um einen besseren Deal zu bekommen.

### Rangliste

Zu Beginn des Spiels ist der eigene Punktestand bei 0, wobei bereits fiktive NPCs (Non Player Character) mit einem höheren Spielstand in der Liste sind. Punkte werden beim Verkauf von Gemüse im Shop vergeben, diese Tabelle erläutert den Punktewert für jedes einzelne Gemüse (2).

Anfangs erhalten die NPCs unterschiedlichen Punktefortschritt zwischen 0 und 126 Punkten. Erhalten Spielende aus jeglicher Quelle Punkte, erhöhen sich die der NPCs mit 50-prozentiger Wahrscheinlichkeit ebenfalls um einen zufälligen Wert (1, 4, 20 oder 100 Punkte). Die Chance wird für jeden NPC extra berechnet. Erhalten Spielende aber über 50 Punkte auf einmal, bekommen die NPCs eine zweite Chance ihre Punkte zu erhöhen, damit der/die Spieler\*in nicht zu schnell auf dem ersten Platz landet.

## Benutzeroberfläche

### Basisspiel / User Interface

Hier aufgelistet sind einige Screenshots der UI. Beginnend mit dem Startmenü (2) und den Textfenstern (3) und Anleitungen (4)(5), die den Spielenden beim ersten Start angezeigt bekommen.

Ein Bild, das Text, Screenshot, Plan, Karte enthält.

Automatisch generierte Beschreibung

Abbildung 3: Benutzeroberfläche - Startmenü

Ein Bild, das Text, Screenshot, Schrift enthält.

Automatisch generierte Beschreibung

Abbildung 4: Benutzeroberfläche - Begrüßung

Ein Bild, das Text, Screenshot, Schrift, Rechteck enthält.

Automatisch generierte Beschreibung

Abbildung 5: Benutzeroberfläche - Steuerung

Ein Bild, das Text, Screenshot, Karte enthält.

Automatisch generierte Beschreibung

Abbildung 6: Benutzeroberfläche - Anleitung

Die folgenden Screenshots zeigen und erklären das User Interface im Spiel (6) und welche Informationen den Spielenden über Items erhalten kann und über welche Optionen diese verfügen (7)(8).

Ein Bild, das Text, Screenshot, Karte enthält.

Automatisch generierte Beschreibung

Abbildung 7: Benutzeroberfläche - User Interface

Ein Bild, das Screenshot, Text enthält.

Automatisch generierte Beschreibung

Abbildung 8: Benutzeroberfläche –   
Item Information

Ein Bild, das Screenshot enthält.

Automatisch generierte Beschreibung

Abbildung 9: Benutzeroberfläche -   
Item Optionen

### Lootbox-Mechanik

Die Lootbox-Mechanik wurde in einen Premium Shop integriert. Auf dieser Oberfläche sind der derzeitige Stand der Spielwährung „Coins“ und Premiumwährung „Silver“ ersichtlich (9). Die Basis Lootbox und Premium Lootbox stehen nebeneinander und können direkt hier per linkem Mausklick geöffnet werden, solange ausreichend der entsprechenden Währung verfügbar ist. Die Animation beginnt und nach kurzer Zeit erscheinen die drei zufällig gewählten Samen.

Die Fragezeichen neben den Kosten sind Buttons, die nach Betätigung ein Info-Fenster mit den entsprechenden Fundraten anzeigen.

Ein Bild, das Text, Screenshot, Software, Multimedia-Software enthält.

Automatisch generierte Beschreibung

Abbildung 10: Benutzeroberfläche - Lootbox-Mechanik

### Premium-Shop

Der wichtigste Teil des Premium Shops ist der Kauf von Premiumwährung mit Echtgeld (10). Wie bereits erwähnt, können Spielende im User Interface Geld mit einem Button generieren. Dieses Geld kann ausschließlich hier ausgegeben werden.

Die Flags „!!!Most Popular!!!“ und „!!!Best Value!!!“ wurden an ähnliche Praktiken aus der Spieleindustrie aber auch aus anderen Bereichen inspiriert. Oft wird bei manchen Angeboten hinzugefügt, dass diese die beliebtesten oder meistverkauften seien.

Ein Bild, das Text, Screenshot, Schrift enthält.

Automatisch generierte Beschreibung

Abbildung 11: Benutzeroberfläche - Premiumwährung Shop

Folgende Teile des Premiumshops erweitern die Möglichkeiten „Coins“ als auch „Silver“ auszugeben:

Die Erweiterung des Inventars um 4 Slots für „Silver“ (11).

Der Kauf von Farmlanderweiterungen für „Coins“ (weil Spielrelevant) (12).

Der Kauf von Strukturen (hier der Brunnen) für „Silver“ (13).

Der Umtausch von „Silver“ in „Coins“, für höheren Anreiz Geld auszugeben (14).

Ein Bild, das Text, Elektronik, Screenshot, Software enthält.

Automatisch generierte Beschreibung

Abbildung 12: Benutzeroberfläche - Inventarerweiterung

Ein Bild, das Text, Screenshot, Software, Betriebssystem enthält.

Automatisch generierte Beschreibung

Abbildung 13: Benutzeroberfläche - Farmlanderweiterung

Ein Bild, das Text, Screenshot, Software, Multimedia-Software enthält.

Automatisch generierte Beschreibung

Abbildung 14: Benutzeroberfläche - Strukturerweiterung

Ein Bild, das Text, Screenshot, Software, Multimedia enthält.

Automatisch generierte Beschreibung

Abbildung 15: Benutzeroberfläche - Umtausch

### Rangliste

Hier abgebildet ist die Rangliste, die Wettbewerb simulieren soll. Dadurch bekommen Spielende den Anreiz, mehr Zeit zu investieren da es auch um etwas geht und es kein entspanntes Einzelspielerspiel ist.

Ein Bild, das Text, Screenshot enthält.

Automatisch generierte Beschreibung

Abbildung 16: Benutzeroberfläche - Rangliste

## Softwarearchitektur

Anhand der in der Methodik im Kapitel 2.3.5 beschriebenen und gewählten Patterns wird der für die Monetarisierung relevante Teil der Softwarearchitektur erklärt.

Die Lootbox-Mechanik wird ebenfalls beschrieben.

### Grundgerüst

Die zentrale Logik, in der die meisten Funktionen zusammengeführt werden, passiert in einem globalen Skript mit dem Namen „Global.gd“. Dieses wird als Autoload Skript geführt, dadurch wird es als allererstes Skript im Programm geladen und wird von allen anderen Skripts gesehen und kann sehr einfach referenziert werden. Es braucht keine eigene Instanz mehr erstellt werden, um auf Variablen und Funktionen zuzugreifen.

Um verschiedene Skripts miteinander zu verbinden, wurden auch Signale verwendet. Wird ein Signal von einem Skript ausgelöst, kann dieses in anderen empfangen werden, wenn ein Empfänger definiert wurde. So können zum Beispiel bei Datenänderungen im Hintergrund alle verbundenen Elemente informiert werden, um beispielsweise einen Wert an mehreren Stellen in der UI gleichzeitig zu aktualisieren.

### Lootbox-Mechanik

Da die Lootbox-Mechanik der zentrale Bestandteil dieser Arbeit ist, wird diese auch detaillierter beschrieben. In Godot wird jeder Gegenstand, jedes Level oder Menü als eine eigene Szene erstellt.

Ein Bild, das Text, Screenshot, Software, Multimedia-Software enthält.

Automatisch generierte Beschreibung

Jede Szene enthält eine sogenannte Root-Node (zu Deutsch: Knotenwurzel) hier „*Chest*“ (17). Diese Node ist ein Parent aller darunter liegenden Child-Nodes (hier zum Beispiel „*Chest*\_*Body*“, welche wiederum Parent von „CollisionShape2D“ ist).

Dadurch kann jedes dargestellte Objekt sehr individuell aufgebaut werden. Wird nun so eine Szene wie diese Lootbox in einem Level benötigt, muss nur die Szene selbst als neue Child-Node in das Level integriert werden. Sie ist dadurch sehr flexibel einsetzbar und kann leicht an mehreren Stellen gleichzeitig verwendet werden, ohne jedes Mal eine ganz neue Kiste programmieren zu müssen.

Abbildung 17: Lootbox Mechanik - Node Übersicht

#### Erläuterung der einzelnen Nodes

* **Chest**

Art: Sprite2D

Aufgabe: Beinhaltet die Sprites (Grafiken) für das Aussehen der Lootbox und der Animation.

Außerdem wurde dieser Node das zugehörige Skript angehängt, welches sämtliche Logik beinhaltet.

* **Chest\_Body**

Art: StaticBody2D

Aufgabe: Definiert mithilfe der Child Node **CollisionShape2D** einen Bereich direkt über der Lootbox, der von Spielfiguren (auch NPCs) nicht betreten werden kann.

* **Area2D**

Art: Area2D

Aufgabe: Definiert mithilfe der Child Node **CollisionShape2D** einen Bereich vor der Lootbox. Befindet sich eine Spielfigur in diesem Bereich, kann die Lootbox geöffnet werden.

* **AnimationPlayer**

Art: AnimationPlayer

Aufgabe: Beinhaltet alle Animationen der Lootbox. Im Fall der Lootbox nur für das Öffnen. Hierfür werden die Sprites verwendet, die in Sprite2D hinzugefügt wurden.

* **AnimationTree**

Art: AnimationTree

Aufgabe: Enthält die erstellten Animationen als Baumstruktur mit einem Startknoten, der zur „Öffnen“-Animation führt.

* **CPUParticles2D**

Art: CPUParticles2D

Aufgabe: Enthält die Einstellungen des Partikeleffekts der Lootbox, der über die CPU berechnet wird. Dieser Partikeleffekt wird vom Skript gesteuert.

* **PointLight2D**

Art: PointLight2D

Aufgabe: Enthält die Einstellungen des Lichteffekts der Lootbox. Dieser Effekt wird vom Skript gesteuert.

#### Lootbox-Mechanik Skript

Zu Programmstart werden vor dem ersten Durchlauf des Skripts alle Variablen initialisiert und vorbereitet, die etwas von außerhalb des Skripts referenzieren oder holen (18).

Alle *„@onready*“ Variablen referenzieren alle Nodes der „*Chest*“ Szene, auf die während des Skripts zugegriffen wird. Außerdem wird eine neue Instanz der *ItemData*-Klasse erstellt, welche sämtliche Informationen über Samen und Gemüse enthalten.

Die „*items*“ Variable referenziert eine Node von außerhalb der Szene, an die die nach dem Öffnen der Lootbox erzeugten Samen-Szenen angehängt werden, damit diese in der Spielwelt erscheinen.

Die Variable „*seeds*“ enthält später die Daten für die Samen.

„*player\_in\_range*“ wird immer dann auf „true“ gesetzt, wenn sich die Spielfigur in der **CollisionShape2D** der **Area2D** befindet.

Ein Bild, das Text, Screenshot, Schrift enthält.

Automatisch generierte Beschreibung

Abbildung 18: Lootbox Mechanik – Skript Variableninitialisierung

Die *„\_ready()*“ Funktion wird einmal am Anfang ausgeführt und weist der Variable „seeds“ den Rückgabewert der „*get\_seeds()*“ Methode zu, welche die Daten der Samen enthält.

„\_*process()*“ prüft, ob „*player\_in\_range*“ auf „true“ gesetzt ist und gleichzeitig die Interaktionstaste von der Testperson gedrückt wird. Ist dies der Fall, wird die Methode „*spawn\_items()*“ ausgeführt.

Ein Bild, das Text, Screenshot enthält.

Automatisch generierte Beschreibung

Abbildung 19: Lootbox Mechanik - Ready und Process Funktionen

Die beiden Funktionen “\_*on\_area\_2d\_body\_entered()*” und “*\_on\_area\_2d\_body\_entered()*” ändern die Variable “*player\_in\_range*” entweder zu „true“ oder „false“, je nachdem ob die Spielfigur den Bereich „**Area2D**“ betreten oder verlassen hat. Zusätzlich wird noch die UI aus- oder eingeblendet (20).

Ein Bild, das Text, Screenshot, Schrift enthält.

Automatisch generierte Beschreibung

Abbildung 20: Lootbox Mechanik - Spielfigur im Bereich

„*spawn\_items()*“ ist die zentrale Funktion, die den Ablauf des gesamten Öffnungsmechanismus abarbeitet. Zuerst wird „*open\_chest\_animation()*“ aufgerufen.

Als nächstes wird die Position der Items in der Spielwelt über die Position der Lootbox selbst definiert, damit die Saatgutpakete neben dieser erscheinen.

Dann werden einzeln die drei Saatgutpakete von der Funktion „*random\_seed\_generator()*“ erzeugt und mit den Positionsdaten der Funktion „*spawn\_items()* übergeben, welche nun die Instanzierung übernimmt und die Items in der Spielwelt platziert.

Im Anschluss wird noch eine Logik ausgeführt, die die Lootbox graduell unsichtbar macht und dann aus der Spielwelt löscht (21).

Ein Bild, das Text, Screenshot, Software enthält.

Automatisch generierte Beschreibung

Abbildung 21: Lootbox Mechanik – Zentrale Logik des Lootbox Mechanismus

„*open\_chest()*“ übernimmt die Animation und die Aktivierung der Licht und Partikeleffekte (22).

Ein Bild, das Text, Screenshot, Schrift enthält.

Automatisch generierte Beschreibung

Abbildung 22: Lootbox Mechanik - Lootbox Animation

“*random\_seed\_generator()*” erzeugt eine zufällige Zahl zwischen 1 und 100 und ruft die Funktion „*get\_seeds\_by\_rarity()*“ mit der zugehörigen Rarität auf und gibt den Rückgabewert dieser (ein zufälliges Saatgutpaket der übergebenen Rarität) zurück.

Ein Bild, das Text, Screenshot, Schrift enthält.

Automatisch generierte Beschreibung

Abbildung 23: Lootbox Mechanik – Zufallsgenerator Saatgutrarität

Ein Bild, das Text, Screenshot, Software, Schrift enthält.

Automatisch generierte Beschreibung

Abbildung 24: Lootbox Mechanik – Zufällige Saatgutauswahl

Ein Bild, das Text, Schrift, Screenshot enthält.

Automatisch generierte Beschreibung

Abbildung 25: Lootbox Mechanik - Instanzierung der einzelnen Saatgutszenen

### Umgesetzte Patterns

* **Wait to Play:** Die Wartezeiten wurden beim Pflanzenwachstum implementiert. Jede Instanz eines gepflanzten Samens erhält je nach Rarität einen eigenen Zeitwert als Float übergeben (10s für Rarität 1, 15s für Rarität 2, 30s für Rarität 3 und 45s für Rarität 4). Bis auf eine Ausnahme hat jede Pflanze 4 Stufen und für jede Stufe wird ein Timer mit der übergebenen Zeitspanne gestartet. Somit wartet die Testperson bei einer Pflanze der Rarität 1 insgesamt 30s bis diese geerntet werden kann.

Ein Bild, das Text, Screenshot, Schrift enthält.

Automatisch generierte Beschreibung

* **Grinding:**

Das Grinding konnte nur in sehr geringem Ausmaß realisiert werden. Bedingt durch die fehlende Möglichkeit das Spiel zu speichern, konnten nur niedrige Wartezeiten beim Pflanzenwachstum gewählt werden, damit die Testpersonen das Spiel auch in einer Sitzung ordentlich testen können. Durch die Game Loop (17) kann aber trotzdem ein länger andauernder Test ermöglicht werden.

* **Can’t Pause or Save:**

Aufgrund zu hoher Komplexität und zu hohem damit verbundenen Zeitaufwand einer Integration einer Speicherfunktion wurde darauf verzichtet. Dadurch müssen die Testpersonen das Spiel in einer einzigen Sitzung testen und können zwischendurch das Spiel nicht beenden.

* **Infinite Treadmill:**

Eine Game Loop wurde realisiert, indem immer wieder von neuem Samen gekauft und angepflanzt werden müssen. Anfangs erhalten Testpersonen eine Grundmenge an der Spielwährung „Coins“ um Basis-Lootboxen zu kaufen und die ersten Samen anzupflanzen. Nach der Ernte wird das Gemüse verkauft, die Testpersonen erhalten „Coins“ und Punkte für die Rangliste und um die „Coins“ können wieder neue Lootboxen oder andere Inhalte im Premiumshop gekauft werden (17).

Ein Bild, das Text, Diagramm, Entwurf, Kreis enthält.

Automatisch generierte Beschreibung

Abbildung 26: Software Architektur - Grinding Cycle

* **Premium Currency:**

Eine Premiumwährung „Silver“ wurde in das Spiel implementiert (11). Am Beispiel der beiden kleinsten kaufbaren Packs ist ersichtlich, dass zuerst der Kontostand, welcher im „Global.gd“ Skript zentral gespeichert wird, mit dem zum Kauf nötigen Betrag verglichen wird. Anschließend werden je ein Signal mit dem neuen Echtgeld-Kontostand und dem Bestand an der Premiumwährung „Silver“ ausgesendet. Somit erhalten alle notwenigen Empfänger ein Update. Zuletzt wird noch der „Spent“-Betrag erhöht, welcher den Gesamtwert des ausgegebenen Echtgeldes anzeigt.

Ein Bild, das Text, Screenshot, Software, Schrift enthält.

Automatisch generierte Beschreibung

Abbildung 27: Software Architektur - Premiumwährung

* **Pay to Skip** oder **Pay to Win:**

Für das Spiel wurde eine Pay to Win Mechanik ausgewählt, da Pay to Skip einen höheren Aufwand bedeutet hätte.

Es wurden zwei Varianten der Lootbox implementiert, eine Basisvariante kaufbar durch die Spielwährung „Coins“ und eine Premiumvariante für die Premiumwährung „Silver“ (10). Die Premiumvariante hat eine doppelt so hohe Chance auf Rarität 3 und 4 Samen wie die Basis-Lootbox.

* **Artifical Scarcity:**

Eine Form von künstlicher Verknappung wurde nicht implementiert da dies ebenfalls zu viel weitere Zeit benötigt hätte. Geplant gewesen wäre aber ein einmaliges Sonderangebot, welches nach ungefähr 5 Minuten im Spiel getriggert worden wäre und der Testperson ein besonders günstiges Angebot unterbreitet hätte wie 5 Premium-Lootboxen zum Preis von einer oder einfach doppelte oder dreifache Premiumwährung auf den nächsten Einkauf.

* **Competition:**

Eine sinnvolle Komponente, die eine Art Wettbewerb ausreichend simuliert, ist ein Leaderboard (zu Deutsch: Rangliste). Um diese sinnvoll umzusetzen, ist es wichtig darauf zu achten, dass sich diese Rangliste so organisch wie möglich anfühlt. Startet die Testperson zum Ersten Mal das Spiel und öffnet das Leaderboard Menü, zeigt der eigene Punktestand 0 und 15 andere simulierte NPCs haben zum Großteil bereits Punkte gesammelt und auch schon einen beachtlichen Punktestand (16).

Sämtliche Daten bezüglich der Punkte werden in einem Dictionary Array gespeichert. Jeder Eintrag beinhaltet den Namen und die dazugehörigen Punkte (Abbildung 19).

Ein Bild, das Text, Screenshot, Schrift, Karte Menü enthält.

Automatisch generierte Beschreibung

Abbildung 28: Software Architektur - Leaderboard Date

Videospiele werden in Schleifen ausgeführt, anders gesagt immer ein Frame (Bild) nach dem anderen. Die *\_process(\_delta*) Funktion wird bei jedem einzelnen Frame einmal aufgerufen (20). Hier wird in jedem Frame bei *testLB()* zuerst geprüft, ob die Taste für das Leaderboard gedrückt wurde und zwischen den Zuständen „Anzeigen“ und „Nicht anzeigen“ wechselt. Der Code darunter überprüft bei jedem Durchlauf, ob sich die Spielerpunkte geändert haben. Falls ja, wird den Punktestand jedes NPCs mit einer Chance von 50% um einen zufälligen Wert erhöht. Erhält ein Spieler über 50 Punkte auf einmal, wird *update\_leaderboard\_data()* ein zweites Mal aufgerufen.

Am Ende wird das Array mit den Spielerdaten geordnet, damit die höchste Punktezahl am Anfang und die niedrigste am Ende steht.

Ein Bild, das Text, Screenshot, Schrift, Software enthält.

Automatisch generierte Beschreibung

Abbildung 29: Software Architektur - Update Leaderboard Data

Von *show\_leaderboard()* und *hide\_leaderboard()* wird zentral die Erzeugung und Darstellung beziehungsweise das korrekte Auflösen und Verbergen des leaderboards gesteuert.

*clear\_leaderboard()* ist dafür verantwortlich, dass alle Zeilen im leaderboard gelöscht werden, damit das Leaderboard beim nächsten Aufruf neu befüllt werden kann.

Ein Bild, das Text, Screenshot, Software enthält.

Automatisch generierte Beschreibung

Abbildung 30: Software Architektur - Show and Hide Leaderboard

Ein weiterer wichtiger Teil der Leaderboard-Logik ist das Erstellen der neuen Zeilen mit den aktuellen Daten. Das Ordnen der Daten passiert beim Prüfen des Punktestandes, sodass in *add\_label()* bei jedem Aufruf des Leaderboards nur jede Zeile neu eingefügt wird (22).

Ein Bild, das Text, Screenshot enthält.

Automatisch generierte Beschreibung

Abbildung 31: Software Architektur - Labels hinzufügen

# Diskussion (900 W.)

Literaturverzeichnis

|  |  |
| --- | --- |
| [1] | J. Clement, „Activision Blizzard Revenue by Composition 2020,“ Statista, 05 2023. [Online]. Available: https://www.statista.com/statistics/274761/electronic-arts-ea-extra-content-revenues/. [Zugriff am 01 12 2023]. |
| [2] | J. Clement, „Activision Blizzard Revenue by Composition 2020,“ Statista, 02 2023. [Online]. Available: https://www.statista.com/statistics/1208560/activision-blizzards-revenue-by-composition/. [Zugriff am 01 12 2023]. |
| [3] | M. Wille, „game,“ game Verband der deutschen Games-Branche, 09 04 2024. [Online]. Available: https://www.game.de/marktdaten/umsatz-mit-computer-und-videospielen-steigt-weiter/. [Zugriff am 29 04 2024]. |
| [4] | A. Castro und J. Alexander, „Nintendo, Microsoft, and Sony commit to cisclose Drop Rates for Loot Boxes,“ The Verge, 08 2019. [Online]. Available: https://www.theverge.com/2019/8/7/20758626/nintendo-microsoft-sony-loot-box-drop-rate-disclosure-video-games. [Zugriff am 01 12 2023]. |
| [5] | C. J. Larche, K. Chini, C. Lee, M. J. Dixon und M. Fernandes, „Rare Loot Box rewards trigger larger arousal and reward responses, and greater urge to open more Loot Boxes,“ *Journal of Gambling Studies,* Bd. 37, pp. 141-163, 11.2019. |
| [6] | B. Abarbanel, „Gambling vs. gaming: a Commentary on the Role of regulatory, industry, and Community Stakeholdersin the Loot Box Debate,“ *Gaming Law Review,* Bd. 22, pp. 231-234, 05.2018. |
| [7] | D. Zendle und P. Cairns, „Video Game Loot Boxes are Linked to Problem gambling: Results of a large-scale Survey,“ *PLOS ONE,* Bd. 13, Nr. 11, pp. 1-12, 11.2018. |
| [8] | J. Koubek, Monetarisierung von Computerspielen, Berlin: Nomos, 2021. |
| [9] | K. Yokomitsu, T. Irie, H. Shinkawa und M. Tanaka, „Characteristics of Gamers who Purchase Loot Box: a Systematic Literature Review,“ *Current Addiction Reports,* Bd. 8, Nr. 4, pp. 481-493, 01 12 2021. |
| [10] | Best-Gaem-Studio, „Steam,“ Pay2Win: The Tricks Exposed, 2015. [Online]. Available: https://store.steampowered.com/app/416760/Pay2Win\_The\_Tricks\_Exposed/. [Zugriff am 30 01 2024]. |
| [11] | P. Luban, „Game Developer,“ 22 11 2011. [Online]. Available: https://www.gamedeveloper.com/design/the-design-of-free-to-play-games-part-1. [Zugriff am 17 04 2024]. |
| [12] | J. Linietsky, A. Manzur und Contributors, „Godot Engine,“ Godot Foundation, [Online]. Available: https://godotengine.org/. [Zugriff am 17 04 2024]. |
| [13] | J. Linietsky, A. Manzur und Community, „Godot Documentation,“ Godot Foundation, [Online]. Available: https://docs.godotengine.org/de/4.x/index.html. [Zugriff am 17 04 2024]. |
| [14] | C. Nooble, „Sprout Lands Asset Pack,“ [Online]. Available: https://cupnooble.itch.io/sprout-lands-asset-pack. [Zugriff am 17 04 2024]. |
| [15] | „Games Wirtschaft,“ 21 02 2024. [Online]. Available: https://www.gameswirtschaft.de/wirtschaft/usk-statistik-2023-altersfreigaben-lootboxen/. [Zugriff am 16 04 2024]. |

Abbildungsverzeichnis

[Abbildung 1: Umsatz mit Computer- und Videospielen in Deutschland 6](#_Toc165560526)

[Abbildung 2: Screenshot des Spiels aus der Vertriebsplattform Steam 14](#_Toc165560527)

[Abbildung 3: Benutzeroberfläche - Startmenü 24](#_Toc165560528)

[Abbildung 4: Benutzeroberfläche - Begrüßung 24](#_Toc165560529)

[Abbildung 5: Benutzeroberfläche - Steuerung 25](#_Toc165560530)

[Abbildung 6: Benutzeroberfläche - Anleitung 25](#_Toc165560531)

[Abbildung 7: Benutzeroberfläche - User Interface 26](#_Toc165560532)

[Abbildung 8: Benutzeroberfläche – Item Information 26](#_Toc165560533)

[Abbildung 9: Benutzeroberfläche - Item Optionen 26](#_Toc165560534)

[Abbildung 10: Benutzeroberfläche - Lootbox-Mechanik 27](#_Toc165560535)

[Abbildung 11: Benutzeroberfläche - Premiumwährung Shop 28](#_Toc165560536)

[Abbildung 12: Benutzeroberfläche - Inventarerweiterung 29](#_Toc165560537)

[Abbildung 13: Benutzeroberfläche - Farmlanderweiterung 29](#_Toc165560538)

[Abbildung 14: Benutzeroberfläche - Strukturerweiterung 29](#_Toc165560539)

[Abbildung 15: Benutzeroberfläche - Umtausch 29](#_Toc165560540)

[Abbildung 16: Benutzeroberfläche - Rangliste 30](#_Toc165560541)

[Abbildung 17: Software Architektur - Lootbox Node Übersicht 31](#_Toc165560542)

[Abbildung 18: Software Architektur - Grinding Cycle 35](#_Toc165560543)

[Abbildung 19: Software Architektur - Premiumwährung 35](#_Toc165560544)

[Abbildung 20: Software Architektur - Leaderboard Date 36](#_Toc165560545)

[Abbildung 21: Software Architektur - Update Leaderboard Data 37](#_Toc165560546)

[Abbildung 22: Software Architektur - Show and Hide Leaderboard 37](#_Toc165560547)

[Abbildung 23: Software Architektur - Labels hinzufügen 38](#_Toc165560548)

Tabellenverzeichnis

[Tabelle 1: Lootboxen – Droprates/Fundraten 16](#_Toc164974360)

[Tabelle 2: Lootbox-Mechanik Seltenheit und Wert der Samen 17](#_Toc164974361)

[Tabelle 3: Premium Shop – Kosten der Inventarerweiterung 18](#_Toc164974362)

[Tabelle 4: Premium Shop - Kosten der Farmlanderweiterung 18](#_Toc164974363)

[Tabelle 5: Premium Shop - Kosten der "Silver" Pakete 19](#_Toc164974364)

Abkürzungsverzeichnis

|  |  |
| --- | --- |
| NPC | Non Player Character |
| DLC | Downloadable Content |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

Anhang A: Überschrift des ersten Anhangs

Anhang B: Überschrift des zweiten Anhangs