HOCHSCHULE
MITTWEIDA
UNIVERSITY OF
APPLIED SCIENCES

# MASTERARBEIT

Frau B. Sc.
**Marianne Poser**

## Decentralized Autonomous Organization als Instrument für Prozesse der Inhaltserstellung auf einer Lernplattform

Mittweida, 2023

Fakultät Angewandte Computer- und Biowissenschaften

**MASTERARBEIT**

# Decentralized Autonomous Organization als Instrument für Prozesse der Inhaltserstellung auf einer Lernplattform

Autor:
**Frau B.Sc.**

**Marianne Poser**

Studiengang:
**Blockchain & Distributed Ledger Technologies**

Seminargruppe:
**BC19w1-M**

Erstprüfer:
**Prof. Dr.-Ing. Andreas Ittner**

Zweitprüfer:
**Dipl.-Volkswirt Mario Oettler**

Einreichung:
**Mittweida, 08.06.2023**

Faculty of Applied Computer Sciences and Biosciences

# MASTER THESIS

# Decentralized Autonomous Organization as a tool for content creation processes on a learning platform

author:
**Ms. B.Sc.**

**Marianne Poser**

course of studies:
**Blockchain & Distributed Ledger Technologies**

seminar group:
**BC19w1-M**

first examiner:
**Prof. Dr.-Ing. Andreas Ittner**

second examiner:
**Dipl.-Volkswirt Mario Oettler**

submission:
**Mittweida, 08.06.2023**

## Bibliografische Beschreibung:

Poser, Marianne:
Decentralized Autonomous Organization als Instrument für Prozesse der Inhaltserstellung auf einer Lernplattform - 2023. - 68 S.
Mittweida, Hochschule Mittweida, Fakultät Angewandte Computer- und Biowissenschaften, Masterarbeit, 2023

## Referat:

Diese Arbeit untersucht die Integration einer dezentralen autonomen Organisation (DAO) in eine bestehende Lernplattform. Dabei werden mögliche Schnittpunkte zwischen den beiden Konzepten gesucht und deren Potenziale und Herausforderungen analysiert. Zunächst werden die theoretischen Grundlagen von Blockchain-Technologie, DAO und Lernplattformen vorgestellt. Darauf basierend werden die Prozesse auf einer Lernplattform auf eine mögliche Integration einer DAO untersucht. Besonderer Fokus liegt dabei auf den Prozessen Bezahlung und Bewertung. Es stellt sich heraus, dass der Reviewprozess zum einen mehr Überschneidungspunkte hat und zum anderen viele weitere Prozesse auf diesen aufbauen. Im Hauptteil der Arbeit wird daher ein Konzept für die Integration einer DAO im Bewertungsprozess entwickelt und erste Grundlagen für eine Umsetzung gelegt. Dabei werden verschiedene Ansätze, die sich auch aus den Fallstudien ergeben, betrachtet und verglichen, um den Anforderungen der Lernplattform und der Blockchain gerecht zu werden. Die Arbeit zeigt die Potenziale einer DAO in einer Lernplattform auf, wie bspw. transparenten Reviewprozess für neue Inhalte. Gleichzeitig werden mögliche Herausforderungen bei der Integration identifiziert und geeignete Lösungsansätze entwickelt. Ein Ausblick auf zukünftige Entwicklungen beinhaltet den Einsatz von Künstlicher Intelligenz bei der Entwicklung von Lerninhalten und die weitere Integration der DAO in die Lernplattform.

# Inhalt

**Inhalt I**

**Abbildungsverzeichnis ..... III**

**Tabellenverzeichnis ..... IV**

**Abkürzungsverzeichnis ..... V**

**1 Einleitung ..... 1**

**2 Grundlagen ..... 3**

*2.1 Blockchain und DAO ..... 3*

*2.2 Lernplattformen ..... 6*

**3 Integration einer DAO in eine Lernplattform ..... 7**

*3.1 DAO als Grundlage einer Lernplattform ..... 7*

*3.2 Integration in Prozesse einer Lernplattform ..... 9*

3.2.1 Prozess Vergütung ..... 9

3.2.2 Prozess Review ..... 10

3.2.3 Bewertung einer möglichen Integration ..... 12

*3.3 Fallstudien ..... 14*

3.3.1 Ed3 DAO ..... 14

3.3.2 ODEM ..... 15

**4 Entwicklung eines Konzepts ..... 18**

*4.1 Vorbereitende Prozesse ..... 18*

4.1.1 Vorbereitende Prüfung ..... 18

4.1.2 Spamming durch Inhaltersteller ..... 19

4.1.3 Vergabe von Schlagwörtern ..... 21

4.1.4 Einsatz von KI ..... 22

*4.2 Bewertungsprozess Nutzer ..... 24*

4.2.1 Qualifizierte Nutzer ..... 24

4.2.2 Auswahlverfahren ..... 25

4.2.3 Bewertung ..... 28

*4.3 Bewertungsprozess Experte ..... 30*

*4.4 Auswertung ..... 33*
4.4.1 Statistische Methoden ..... 33
4.4.2 Entwicklung eigener Verfahren ..... 34
4.4.3 Bewertung der Verfahren ..... 35
4.4.4 Mögliche Herausforderungen ..... 40
*4.5 Konzepte für Nutzer ..... 42*
4.5.1 Rankingsystem ..... 42
4.5.2 Anreizsystem ..... 45
**5 Entwicklung eines Prototyps ..... 47**
*5.1 Aufbau ..... 47*
*5.2 Prozess der Nutzerbewertung ..... 50*
5.2.1 Verdeckte Bewertung durch Nutzer ..... 51
5.2.2 Ablauf für den Bewertungsprozess der Nutzer ..... 52
5.2.2.1 Anmeldung Nutzer ..... 53
5.2.2.2 Erzeugung Zufallszahl ..... 53
5.2.2.3 Auswahl Nutzer ..... 55
5.2.2.4 Abgabe der Bewertung ..... 58
5.2.2.5 Beenden eines Bewertungsprozess ..... 59
5.2.3 Ausblick ..... 62
**6 Fazit ..... 64**
**Literaturverzeichnis ..... 66**
**Anhang A: Code Smart Contract ..... I**
**Selbstständigkeitserklärung**

# Abbildungsverzeichnis

Abbildung 1: Diagramm Vorbereitungsprozess ..... 20

Abbildung 2: Qualifizierungsprozess ..... 24

Abbildung 3: Teilnehmer am Bewertungsprozess ..... 32

Abbildung 4: Smart Contract Datenstruktur Operator ..... 47

Abbildung 5: Smart Contract Datenstruktur User ..... 48

Abbildung 6: Smart Contract Datenstruktur VotingRound ..... 50

Abbildung 7: Interaktionen Nutzer - Smart Contract ..... 53

Abbildung 8: Smart Contract Function chooseRandomUser ..... 56

Abbildung 9: Smart Contract For-Schleife chooseNewUsers ..... 57

Abbildung 10: Smart Contract Function submitEvaluation ..... 58

Abbildung 11: Smart Contract Function endVotingRound ..... 60

Abbildung 12: Smart Contract Function decryptEvaluation ..... 60

Abbildung 13: Smart Contract Function rankUser ..... 61

Abbildung 14: Smart Contract Zusatzfunktionen rankUser ..... 62

# Tabellenverzeichnis

Tabelle 1: Auswertung Beispiel 1 .....36

Tabelle 2: geordnete Werte .....37

Tabelle 3: Auswertung Beispiel 2 .....38

Tabelle 4: Auswertung Beispiel 3 .....39

Tabelle 5: Anpassungen bei verschiedenen Szenarien.....41

# Abkürzungsverzeichnis

| | |
|---|---|
| **BCAM** | Blockchain Academy Mittweida |
| **DAO** | Dezentrale Autonome Organisation |
| **EIP** | Ethereum Improvement Proposal |
| **ERC** | Ethereum Request for Comments |
| **EVM** | Ethereum Virtual Machine |
| **IPFS** | Interplanetary File System |
| **IQR** | Interquartile Range |
| **VRF** | Verifiable Random Function |

# 1 Einleitung

Die Entwicklung und der fortschreitende Einsatz von Blockchain-Technologie haben auf viele Bereiche einen positiven Einfluss. Besonders hervorzuheben ist, dass daraus ein System gewachsen ist, in welchem unabhängig von zentralen Instanzen monetäre Transaktionen getätigt werden können. Dieses System ist zudem für jeden zugänglich und agiert über den Globus hinweg gleich. Betrachtet man die aktuelle Bildungswelt, so sind viele dieser Errungenschaften auch für diesen Bereich erstrebenswert.

Bildung sollte für alle Menschen gleichermaßen zugänglich sein, schließlich gehört das Recht auf Bildung zu den Menschenrechten. Doch sieht die Realität oft anders aus. Weltweit gehen 260 Millionen Kinder und Jugendliche nicht zur Schule und verpassen damit den Start in ein Leben mit Bildung. (Statistisches Bundesamt 2021) Doch nicht nur Kinder und Jugendliche sind betroffen, auch Erwachsenen fehlt der Zugang zu neuem Wissen und einer besseren Ausbildung. Viele junge Erwachsene haben keine Möglichkeit auf eine berufliche Ausbildung oder ein Studium. Fest steht, dass die Bildungschancen vor allem von Wohnort und Einkommen abhängig sind, Bildung aber die Grundlage für eine individuelle und gesellschaftliche Weiterentwicklung ist. (Bundesministerium für wirtschaftliche Zusammenarbeit und Entwicklung)

Auch die Entscheidungshoheit über Bildungsinhalte durch zentrale Institutionen sollte überdacht und Alternativen dazu entwickelt werden. Um auf die Möglichkeiten und Bedürfnisse von Lernenden eingehen zu können, fordern auch Lehrkräfte und Schulleitungen immer mehr, auch in die Gestaltung des Lehrplans eingreifen zu können. Eine Verbindung aus Lernenden, Eltern und Lehrenden sollte in die Entscheidungsprozesse eingebunden werden, da sie die Gegebenheiten und Bedürfnisse vor Ort am besten kennen. (Warnke 2023)

Viele Merkmale, welche zur Charakteristik der Blockchain gehören, würden also auch bei der Weiterentwicklung des Bildungswesens ungemeine Vorteile bringen. Die Verbindung zwischen Blockchain und Bildung kann richtig umgesetzt, das Lernen weltweit revolutionieren. Beispielsweise kann die Blockchain als dezentrales, transparentes System auch die Ablage von Zertifikaten effizienter gestalten. Zusätzlich kann die Sicherheit und Integrität von Abschlüssen in Verbindung mit guten Identitätslösungen erhöht werden. Ein auf einer Blockchain abgebildet demokratischer Entscheidungsprozess über Lerninhalte kann Lehrende und Lernende beim Erstellen von Lernplänen sinnvoll einbinden. So kann die realitätsnahe Expertise von Lehrenden direkt einfließen und die Interessen von Lernenden mitbeachtet werden. Dies wiederum kann die Motivation bei allen Beteiligten steigern.

Zuletzt ermöglicht insbesondere eine dezentrale autonome Organisation auf der Blockchain das Organisieren einer Community, in der Wissen frei geteilt und miteinander ausgetauscht werden kann. Dabei kann demokratisch über Inhalte und Vermittlungsarten abgestimmt werden und auch individuell auf Bedürfnisse eingegangen werden. Dies kann unabhängig von Ort und lokalen Gegebenheiten, von Alter und Bildungsstand geschehen und wäre damit ein wichtiger Schritt hin zur Bekämpfung von Armut und ein Erhöhen von Chancengleichheit.

Diese Punkte zeigen, wie viel Potenzial die Verbindung von Blockchain und Bildung trägt. In dieser Arbeit soll ein möglicher Verbindungspunkt zwischen beidem betrachtet werden, nämlich die Schnittmenge von der Funktionalität einer dezentralen autonomen Organisation (DAO) und den Prozessen sowie Anforderungen einer Lernplattform. Dabei soll die Frage, inwieweit eine dezentrale autonome Organisation als Instrument für die Prozesse der Inhaltserstellung auf einer Lernplattform, insbesondere in Bezug auf die Prozesse Bezahlung und Review dienen kann, bearbeitet werden. Die Fragestellung zielt darauf ab, Potenziale und Herausforderungen in der Verbindung von DAO und Lernplattform zu untersuchen. Der Fokus soll auf den Prozessen Bezahlung und Review liegen, da diese für die Qualitätssicherung und das Einbinden externer Inhaltsersteller in eine bestehende Plattform entscheidend sind.

Neben den theoretischen Betrachtungen dieses Sachverhalts sollen in Form eines Konzepts und der Entwicklung eines einfachen Prototyps auch die Möglichkeit einer realen Integration geprüft werden. Besonders bei der Entwicklung des Konzepts können die zuvor gewonnen Erkenntnisse mit eigenen Ansätzen verbunden werden. Ziel ist dabei ein umfassendes Konzept zu entwickeln, welches Lösungen für verschiedene Szenarien beinhaltet. Bei der Entwicklung des Prototypens können Herausforderungen bei der Umsetzung einer Lernplattform mit Blockchain-Technologie herausgearbeitet und mögliche Lösungen vorgestellt werden. Dabei wird auf die Besonderheiten der Entwicklung mit Blockchain eingegangen. Diese Arbeit kann also sowohl als Diskussionsgrundlage für die Verbindung von DAO und Lernplattform dienen, als auch als Ausgangspunkt für die Integration von Blockchain in einer Lernplattform genutzt werden.

Auf dieses einleitende Kapitel folgt im nächsten Kapitel die Vermittlung benötigter theoretischer Grundlagen. Darin wird sowohl auf die Blockchain-Technologie und die Arbeitsweise von DAOs eingegangen als auch auf die Grundlagen von Lernplattformen. Im dritten Kapitel werden die Konzepte Lernplattform und DAO auf ihre Schnittmenge geprüft und dabei besonders auf die Prozesse Vergütung und Review geachtet. Darauf folgt die Entwicklung eines Konzepts für den Review-Prozess auf einer Lernplattform mit Einsatz von Blockchain-Technologie im vierten Kapitel. Daran schließt sich ein Kapitel über die Möglichkeit der Umsetzung dieses Konzepts in Form eines Prototypen an. Den Abschluss der Arbeit bildet ein Fazit, indem die Erkenntnisse zusammengefasst und abschließend ausgewertet werden.

# 2 Grundlagen

In diesem Kapitel sollen die theoretischen Grundlagen, die für das Verständnis dieser Arbeit bedeutend sind, behandelt werden. Zunächst wird dafür der Begriff DAO definiert und ihre Funktionsweise sowie ihr Aufbau erklärt. Dafür wird auch auf die Begriffe Blockchain und Smart Contract eingegangen. Zuletzt werden außerdem die Grundlagen von Lernplattformen behandelt. Dadurch soll ein Einblick in die Prozesse und Funktionen gewonnen werden.

## 2.1 Blockchain und DAO

Eine dezentrale autonome Organisation (DAO) kann als ein System definiert werden, welches einer Gruppe von Menschen ermöglicht, sich selbst unabhängig von einer zentralen Instanz zu organisieren. Dabei wird sowohl auf eine zentrale Autorität als auch auf ein zentrales Vermittlungssystem verzichtet. Stattdessen basiert sie auf einer dezentralen Technologie mit deren Hilfe transparent und demokratisch Entscheidungen getroffen werden. Die Grundlage ist eine Blockchain. (Hassan und Filippi 2021)

Blockchain basiert auf einem dezentralen Netzwerk, bei welchem Verarbeitung und Speicherung von Daten auf mehrere Computern (Nodes) verteilt werden. Die Nodes, welche in Form eines Peer-to-Peer-Netzwerks miteinander verbunden sind, arbeiten zusammen, um sich auf einen gemeinsamen Datensatz zu einigen. Das bedeutet, ihr Ziel ist es, ohne eine zentrale Instanz einen Konsens über den Zustand der Blockchain zu erreichen. Die Kommunikation, die Ablage und die Veränderung von Daten erfolgt dabei in Datenblöcken, welche miteinander verbunden sind. Die Verknüpfung entsteht durch das Einbinden eines Hashs über den vorhergehenden Block. Dadurch sind Blöcke, welche einmal aufgenommen wurden, also auf die weitere Blöcke folgen, im Nachhinein nicht mehr änderbar. Die Blöcke enthalten Transaktionen, durch die Daten in die Blockchain aufgenommen werden oder durch eine bestimmte Aktion verändert werden können. Die Transaktionen werden zunächst von den Nodes verifiziert und validiert und dann in einem neuen Block zusammengefasst. Dieser Block wird anschließend der bestehenden Blockchain hinzugefügt, indem der Hash des vorherigen Blocks eingebunden wird und der Block dann im Netzwerk verteilt wird. Durch diese Verknüpfung entsteht eine Kette von Blöcken, welche nicht mehr änderbar ist. Damit wird Sicherheit und Zuverlässigkeit gewährleistet. (Zhang und Anand 2022)

Die erste Umsetzung dieses Konzepts erfolgte mit der Bitcoin-Blockchain, deren erster Block im Jahr 2009 generiert wurde. Nach und nach folgten weitere Implementierungen dieses Konzepts. Eine der Weiterentwicklungen ist die Ethereum-Blockchain mit dem Ziel,

leichter eigene Anwendungen auf der Blockchain implementieren zu können. Dafür wurde die Programmiersprache Solidity entwickelt, welche eine Turing-vollständige Sprache ist. Sie ermöglicht die Entwicklung von Programmen, sogenannten Smart Contracts, welche die reale Welt mit Transaktionen auf der Blockchain verknüpfen können. Für die Ausführung der Smart Contracts auf der Ethereum-Blockchain wurde eine virtuelle Maschine entwickelt. Die Ethereum Virtual Machine (EVM) baut auf die eigentliche Blockchain auf und kompiliert die Smart Contracts ausgehend Solidity zu EVM-Bytecode. Dieser wird dann von allen Netzwerkknoten der Ethereum-Blockchain ausgeführt. Die Rechenleistung muss von dem Sender der Transaktion bezahlt werden. Dies geschieht mit der Währung Ether, basierend auf der internen Recheneinheit Gas. Damit soll ein effizienter Umgang mit den Ressourcen der Blockchain erreicht werden. (Zhang und Anand 2022)

Sowohl die Bitcoin- als auch die Ethereum-Blockchain sind öffentlich. Das bedeutet, die Daten, welche in den Blöcken abgelegt sind, sind für alle einsehbar. Das hat zum einen den Vorteil der Transparenz, dass alle die Gültigkeit von Transaktionen prüfen und nachvollziehen können. Das hat aber zum anderen den Nachteil, dass an sich keine Privatsphäre gegeben ist. Sender und Empfänger von Transaktionen sowie der Inhalt und die Höhe von gesendeten Beträgen sind öffentlich einsehbar. Das kann bei der Umsetzung von Anwendungen auf der Blockchain problematisch sein. Eine Lösung ist der Einsatz von Hashfunktionen. Dabei werden Verlinkungen von Daten auf der Blockchain abgelegt, wobei der Inhalt verborgen bleibt. Dies kann sowohl für Nachweise genutzt werden als auch für Abstimmungen, bei denen erst nach Abschluss der Abstimmung die Auswertung erfolgen soll. (Zhang und Anand 2022)

Aufbauend auf Smart Contracts haben sich inzwischen dezentrale Anwendungen (dApps) durchgesetzt. Sie bestehen aus einer Benutzeroberfläche, auf welche über den Browser sichtbar ist, welche wiederum mit einem oder mehrere Smart Contracts auf einer Blockchain verbunden ist. So können Daten transparent und dezentral in der Blockchain abgelegt werden, verbunden mit der Annehmlichkeit einer benutzerfreundlichen Oberfläche. Diese kann die Grundfunktionalität der öffentlich einsehbaren Smart Contracts erweitern und die Daten aufbereiten. Als Verbindung zwischen der Anwendung im Browser und der Ethereum-Blockchain kann beispielsweise MetaMask[1] genutzt werden. (Quiniou 2019)

Smart Contracts werden auch bei DAOs eingesetzt und übernehmen dabei verschiedene Funktionen. Vorrangig ermöglichen sie die transparente Festlegung von Regeln für die Interkationen innerhalb der DAO. Zusätzlich sorgt die Programmierung dieser Regeln als Smart Contract dafür, dass diese Regeln immer entsprechend der Festlegung ausgeführt werden. So ist es ohne eine zentrale Instanz möglich, dass getroffene Entscheidungen automatisch und zuverlässig umgesetzt werden. Wie die Entscheidungen getroffen

[1] https://metamask.io/

werden und wie die Verwaltung der Gemeinschaft einer DAO gestaltet wird, kann ebenfalls in Smart Contracts festgelegt werden. Auf diese Art und Weise kann sich die DAO selbst regieren und demokratisch handeln, ohne auf eine zentrale Instanz vertrauen zu müssen. Die Verwendung von Blockchain als Grundlage gewährleistet Transparenz, Sicherheit und Dezentralität. (Hassan und Filippi 2021)

Um die erläuterte Charakteristik einer DAO zu erreichen, werden häufig die gleichen Komponenten verwendet. Einige dieser Bestandteile sollen im Folgenden genannt und erläutert werden.

1. Konsensmechanismus – Ähnlich wie bei der Blockchain braucht es einen Mechanismus, wie innerhalb der DAO Einigung gefunden wird. Dabei gibt es verschiedene Ansätze, wie dies ohne zentrale Instanz gelingen kann. Dazu zählen komplexe Konsensmechanismen ebenso wie simple Mehrheitsentscheidungen.
2. Abstimmungen: Ein wichtiger Bestandteil eines Konsensmechanismus ist ein Abstimmungsprozess. Dies gibt den Mitgliedern der DAO die Möglichkeit, sich an Entscheidungen zu beteiligen und die Zukunft ihrer DAO mitzugestalten. Die Regeln der Abstimmung können in einem Smart Contract abgelegt werden und werden über die Blockchain durchgeführt.
3. Proposal: Ein Proposal geht einer Abstimmung voraus. Es ist ein Vorschlag, der von einem DAO-Mitglied eingebracht wird. Dabei können Änderungen der Regeln vorgeschlagen werden oder auch welche Investitionen als DAO getätigt werden sollen.
4. Token: Token sind digitale Vermögenswerte, welche unter anderem auch innerhalb von DAOs genutzt werden können. Ihr Einsatzmöglichkeiten sind zahlreich und reichen von der Repräsentation von Stimmrechten bis zum finanziellen Anreiz innerhalb der DAO.
5. Governance: Die Gesamtheit aller Regeln und Festlegungen innerhalb der DAO nennt sich Governance. Das umfasst die Ausgestaltung der verschiedenen Prozesse wie Proposal-Einreichungen, Abstimmungen und Mechanismen zur Änderung der Regeln.

Die vorgestellten Bestandteile sind erweiterbar durch verschiedene Konzepte. Beispielsweise können den Mitgliedern einer DAO Rollen zugeordnet werden, welche mit bestimmten Rechten und Pflichten verbunden sind. Auch ein Reputationssystem kann innerhalb der DAO aufgebaut werden. Ein weiteres übliches Konzept ist es, dass die DAO über ein gemeinsames Gut verfügt. Dieses sogenannte Treasury kann beispielsweise Token beinhalten, welche in dem Smart Contract der DAO abgelegt sind und über deren Verwendung die Mitglieder der DAO gemeinsam entscheiden können. (Wang et al. 2019)

## 2.2 Lernplattformen

Eine Lernplattform ist ein virtueller Raum, auf welchem Bildungsinhalte online angeboten werden. Die digitale Plattform ist damit ein wichtiger Grundbaustein im Bereich E-Learning. Denn auf der Plattform können die elektronischen Inhalte präsentiert werden und bieten einen virtuellen Lernraum, in welchem die Lernmedien angeboten und bearbeitet werden. Neben der Bereitstellung von Lernmaterialien wie Videos, E-Books und Präsentationen gehört auch die Strukturierung in Kurse oder Lektionen zu den Funktionen einer Lernplattform. Aus der Möglichkeit, dass Nutzer die Inhalte bearbeiten können, ergibt sich eine weitere Funktionalität der Lernplattform – die Lernüberwachung. Dazu zählt sowohl die Anzeige des Fortschritts beim Bearbeiten des Inhalts als auch das Bereitstellen und Bewerten von Wissensüberprüfungen wie Übungen, Tests und Quizfragen. (Arnold 2018)

Das Anbieten von Lerninhalten im virtuellen Raum ermöglicht dem Lernenden, Autonomie und Selbstwirksamkeit zu erfahren, sowohl individuell als auch in Kooperation mit anderen. Dies geschieht dadurch, dass Zeit und Ort sowie Lerntempo frei bestimmbar sind und auch Inhalte nach Belieben bearbeitet werden können. Um eine Zusammenarbeit mit anderen Lernenden und Lehrenden zu ermöglichen, benötigt es Werkzeuge zur interaktiven Kommunikation auf der Plattform. Zu diesen Tools zählen sowohl asynchrone Formen wie ein Forum oder die Möglichkeit, Nachrichten zu versenden, genauso wie synchrone Kommunikationsformen bspw. eine Videokonferenz. (Arnold 2018; Picciano 2019)

Aus den Anforderungen und der vorgestellten Funktionalität von Lernplattformen ergeben sich Prozesse, welche in einer Lernplattform umgesetzt werden müssen. Dazu zählt vor allem die Verwaltung von Nutzern und Inhalten. Bezogen auf die Nutzer müssen die Modalitäten für den Zugriff auf die Lernplattform bzw. deren Inhalte geregelt werden. Die Prozesse Registrierung und Anmeldung müssen definiert werden. Wenn der Zugriff auf die Inhalte nicht kostenfrei sein soll, muss ein Bezahlungssystem eingerichtet werden und auch dafür ein gutes Konzept gefunden werden. Zusätzlich braucht es ggf. Ansätze für eine mögliche Zusammenarbeit innerhalb eines definierten Lernraums sowie für die interaktiven Teile der Plattform. Dazu zählen auch die Prozesse zur Überwachung des Lernfortschritts. Für die Verwaltung von Lerninhalten müssen Abläufe für die Inhalterstellung, Strukturierung der Angebote und Qualitätskontrolle entwickelt werden. (Picciano 2019; Arnold 2018)

Eine Lernplattform ist also ein virtueller Raum, indem Lehrende ihre Inhalte medialisiert und interaktiv, aber meist in asynchroner Form präsentieren und Lernende diese zeit- und ortsunabhängig bearbeiten können. Das ermöglicht Pädagogen, Experten und Lehrkörpern neben dem Einsatz von vielfältigen Lernressourcen vor allem auch Skalierbarkeit, sodass ihr Wissen effizient mit einer Vielzahl von Menschen geteilt werden kann. Für Lernende kann eine Lernplattform ein Ort für lebenslanges Lernen mit einem hohen Maß an Zugänglichkeit und Flexibilität sein.

# 3 Integration einer DAO in eine Lernplattform

## 3.1 DAO als Grundlage einer Lernplattform

Ausgehend von dem Ansatz einer DAO, dass eine Gruppe von Mitgliedern ein Gut gemeinschaftlich verwaltet und für ein gemeinsames Ziel einsetzt, soll in diesem Unterkapitel ein Aufbau einer DAO grob skizziert werden, deren Gut die Lernplattform selbst ist. Dies soll als Ausgangslage dienen, um im weiteren Verlauf auf die Konzepte einer DAO in Verbindung mit einer Lernplattform zurückgreifen zu können und um besonders sinnvolle Einsatzmöglichkeiten einer DAO zu erkennen. Zusätzlich können auch Fragestellungen und Herausforderungen entdeckt werden, welche im Weiteren besondere Aufmerksamkeit benötigen.

Zunächst kann eine DAO als Verwaltungsinstanz einer Lernplattform betrachtet werden. Alle Entscheidungen, die innerhalb einer Plattform getroffen werden müssen, werden von den Mitgliedern der DAO beschlossen. Dabei kann es sowohl um den grundsätzlichen Aufbau der Lernplattform als auch um den darauf verfügbaren Inhalt gehen. Um Mitglied einer DAO zu werden, muss man klassischerweise deren Token kaufen oder sich registrieren. Die Mitglieder der DAO können dann verschiedene Rollen einnehmen und damit verbundene Rechte wahrnehmen. Beispielsweise kann es die folgenden Rollen für die Mitglieder einer DAO mit einer Lernplattform geben:

1. Konsument – Rolle mit den wenigsten Rechten. Er kann an Abstimmungen teilnehmen.
2. Inhaltersteller – Kann neue Inhalte vorschlagen und eine Abstimmung darüber fordern.
3. Verwalter – Rolle mit den meisten Rechten, die bspw. Abstimmungen zu Plattformthemen (bspw. Benennen von möglichen Inhalterstellern) fordern kann.

Der Aufbau der Gemeinschaft innerhalb der DAO sollte dabei von Beginn an gut durchdacht sein. Es sollten die Fragen geklärt werden, wer wie die verschiedenen Rollen einnehmen kann, welche Berechtigung die einzelnen Rollen tatsächlich mit sich bringen und welche Sicherheitsmechanismen gibt. Zu den Sicherheitsmechanismen zählen zum Beispiel auch, dass man in seiner Rolle zurückgestuft wird oder ggf. komplett ausgeschlossen werden kann. Aber auch viele andere Szenarien, welche für die DAO gefährlich werden können, sollten bedacht werden. Dazu zählen unter anderem das Zusammenschließen von Mitgliedern gegen andere oder sogar gegen die DAO und das Angreifen der DAO mit und ohne wirtschaftlichem Interesse.

Auch für die Abstimmungen muss ein gutes Konzept gefunden werden. Dabei können Token eingesetzt werden oder jedes Mitglied hat automatisch eine Stimme, die es abgeben

kann. Ausgelöst wird eine Abstimmung durch das Einreichen eines Vorschlags (Proposals), welches sich auf Veränderungen der Plattform oder auch auf Inhalte beziehen kann. Wichtige Fragestellungen sind dabei, wer darf Proposals einbringen, sollte das Einbringen mit Kosten oder mit dem Hinterlegen von Token verbunden sein und mit welchen Mechanismen kann sonst noch Spam vermieden werden. Beispielsweise könnten Inhaltersteller, welche einen neuen Kurs auf der Plattform veröffentlichen wollen, diesen als Proposal einbringen und in einer Abstimmung wird entschieden, ob der Kurs Teil der Plattform werden soll. Auch für die Abstimmung müssen verschieden Fragen und Szenarien bedacht werden:

1. Werden fertige Kurse eingereicht oder nur Themenvorschläge? Oder zunächst der Vorschlag und anschließend der Inhalt?
2. Wie werden die Preisgestaltung und die Bezahlung gehandhabt?
3. Ist das Stimmrecht tokenbasiert und somit käuflich oder hat jedes DAO-Mitglied genau eine Stimme?
4. Wie sind die Abstimmungsfaktoren definiert – abgeschlossene Abstimmung, erfolgreiche Abstimmung und Mehrheit?

Zu beachten ist dabei auch immer wieder die Herausforderung, dass die Mechanismen nicht ausnutzbar sind. Wenn beispielsweise ein Inhaltersteller seinen kompletten Inhalt in einem Proposal veröffentlichen muss, damit der Inhalt auf die Plattform aufgenommen wird und erst nach erfolgreichem Proposal bezahlt wird, kann dies von den DAO-Mitgliedern ausgenutzt werden. Sie haben dann bereits den Inhalt bekommen und könnten ihn aber im Proposal immer ablehnen, um eine Auszahlung an den Inhaltersteller zu vermeiden. Auch bei der Frage, ob DAO-Mitglieder automatisch Zugriff auf alle Inhalte erhalten, sollte auf verschiedene Angriffsmöglichkeiten geachtet werden. An sich könnte es ein Ansatz sein, dass alle Mitglieder auch Zugriff auf die Inhalte der gemeinsamen Plattform erhalten. Auf der anderen Seite könnte man nur kurz Mitglied sein, die Inhalte konsumieren und sich dann aus der DAO wieder zurückziehen.

Die wichtigste und auch schwierigste Herausforderung ist das Finden eines Kompromisses zwischen dem gemeinschaftlichen Ansatz einer DAO, dem Absichern gegen verschiedene Angriffsmöglichkeiten und der Wirtschaftlichkeit einer Lernplattform. Zusätzlich kommt noch der Faktor der Umsetzbarkeit hinzu. Die vielen aufgezählten Fragen, welche großteils komplexe Antworten benötigen und auch nur einen Bruchteil eines vollständigen Konzepts darstellen, zeigen, wie umfangreich die Verbindung DAO und Lernplattform beleuchtet werden muss. Vor allem mit Blick auf die Umsetzbarkeit, aber auch auf die Wirtschaftlichkeit und Sicherheit soll im Folgenden der Fokus auf die Vorgänge einer Lernplattform gerichtet werden und wie da eine DAO eingesetzt werden kann, anstelle dass eine DAO als Grundlage für eine Lernplattform eingesetzt werden soll.

## 3.2 Integration in Prozesse einer Lernplattform

In diesem Unterkapitel sollen erste Ansätze für die Vorgänge Vergütung und Review vorgestellt werden. Die Betrachtung basiert zunächst auf den Prozessen von Lernplattformen ohne einer Verknüpfung zur Blockchain-Technologie. Das soll im Weiteren dazu dienen, bewerten zu können, welcher Vorgang sich am besten mit einer DAO verknüpfen lässt, ohne den Blick für die eigentlich wichtigen Abläufe einer Lernplattform zu verlieren.

### 3.2.1 Prozess Vergütung

Der erste Vorgang, welcher näher betrachtet werden soll, ist die Bezahlung von Inhaltersstellern. Es gibt also eine Lernplattform, auf der externe Inhaltersteller ihren Inhalt veröffentlichen können und dafür bezahlt werden. Für diese Bezahlung gibt es unterschiedliche Optionen, wie und wann sie erfolgen soll. Einige davon sollen nun mit ihren Vorteilen und Nachteilen betrachtet werden.

**Festpreis** Nach Erstellung und Veröffentlichung des Inhalts erhält der Ersteller eine zuvor festgelegte Summe für seinen Aufwand. Vorteil daran ist, dass der Ersteller eine sichere Bezahlung erhält und es einfach umsetzbar ist. Allerdings gehen auch viele Nachteile mit diesem simplen Ansatz einher. Zum einen besteht weder ein Anreiz für eine besonders gute Inhalterstellung noch dafür, den Inhalt mit der Zeit zu aktualisieren. Zusätzlich kann es schnell ausgenutzt werden, indem viele mittelmäßige und ähnliche Inhalte veröffentlicht werden. Eine Variante dieses Ansatzes wäre, dass es eine Ausschreibung zu einem festen Thema mit fester Bezahlung gibt, auf die dann Inhalt eingereicht werden kann. Das wäre allerdings wieder mit mehr Aufwand verbunden und auch mit einem gewissen Risiko für Ersteller, welche die Ausschreibung nicht gewinnen, aber trotzdem Inhalt dafür erstellt haben.

**Beteiligung** Bei dem Modell wird ein Inhaltersteller an den Einnahmen durch den Verkauf seines Kurses beteiligt. Er erhält also erst dann eine Bezahlung, wenn sein Kurs auch von den Nutzern gekauft wird. Mit jedem Kauf erhält er einen prozentualen Anteil an der Kursgebühr. Auf diese Weise wird für ihn ein Anreiz geschaffen, gute Qualität zu liefern, den Kurs aktuell zu halten und sich ggf. sogar an der Vermarktung seiner Kurse zu beteiligen. Nachteil ist, dass es weiterhin für ihn sinnvoll ist, viele Kurse mit ähnlichem Inhalt zu erstellen. Außerdem geht ein Ersteller ein gewisses Risiko ein, dadurch, dass er nicht weiß, wie gut seine Kurse ankommen und wie gut auch die Plattform selbst ist. Vor allem bei einer neuen Plattform wird es schwierig sein, Inhaltersteller für dieses Modell zu begeistern.

**Reviewbasierend** Eine erste Abwandlung des Festpreis-Modells kann das Einführen einer Review sein. Der Inhaltersteller reicht also seinen Kurs ein, dieser wird in einem Reviewprozess bewertet und er erhält anschließend basierend auf dem Reviewergebnis eine einmalige Bezahlung. Erst dann wird der Inhalt veröffentlicht. Dies schafft einen Anreiz, qualitativ hochwertige Kurse zu erstellen und verbessert außerdem auch die Plattform

insgesamt. Je nach Review-Modell kann der Ansatz durch die Reviewer allerdings ausgenutzt werden. Wenn diese ein Interesse daran haben, dass die Plattform möglichst wenig zahlt (bspw., weil sie selbst über eine DAO in die Plattform investiert haben), könnten sie auch guten Inhalt schlecht bewerten. Der Inhaltersteller geht also das Risiko ein, Inhalte unter Wert an die Plattform zu verkaufen.

Diese Ansätze können zusätzlich zu ihrer vorgestellten Form auch miteinander kombiniert werden, um ihre Vorteile zu verstärken und Nachteile möglichst auszugleichen. Ziel ist es dabei einen Kompromiss zu finden, bei dem Inhaltersteller motiviert sind, ausschließlich gute Qualität zu liefern (und eventuell sogar aktuell zu halten) und dennoch kein zu hohes Risiko eingehen zu müssen. Je nach dem, worauf eine Plattform den Fokus setzen will – bspw. viele gute Kurse oder wenige sehr gute Kurse, können die Bezahlungsvarianten kombiniert werden.

Bei der Kombination von einer festen, einmaligen Zahlung und einer Beteiligung an den Einnahmen des Kurses ist es für die Ersteller sehr attraktiv, viele (ähnliche) Kurse zu erstellen. Sie müssen dabei nicht so stark auf die Qualität Wert legen, sondern können über Quantität gut Einnahmen generieren. Beim Kombinieren von Beteiligung und Review hingegen wird gute Qualität besonders belohnt, allerdings ist der Ersteller dabei auch am meisten von Review und Plattform abhängig. Die Kombination von allen drei Varianten kann schnell zu komplex und unübersichtlich werden, auch wenn sie die beste Kombination für Anreiz und Sicherheit bietet.

Es zeigt sich bei dieser Betrachtung, dass vor allem der Einsatz einer Review für die Plattform hilfreich ist. Auf diese Weise kann sie die Qualität der angebotenen Kurse sicherstellen. Eine Bezahlung basierend auf ein Reviewverfahren bedarf eines guten Konzepts, um ein ehrliches Feedback zu erhalten. Doch auch beim Modell, dass es einen Festpreis gibt, ist der Einsatz eines Kontrollmechanismus auf die Qualität des Kurses sinnvoll, welcher aber nicht direkt mit der Bezahlung einhergeht. Aus diesem Grund wird im folgenden Kapitel näher auf die Möglichkeiten und Abläufe einer Review eingegangen.

### 3.2.2 Prozess Review

In einem Reviewprozess soll die Qualität eines Kurses bewertet werden. Der Zeitpunkt für diesen Prozess ist sinnvollerweise vor der Freigabe des Inhalts auf der Plattform für die Öffentlichkeit. Aber auch eine nachträgliche Bewertung der Kursqualität kann sinnvoll sein. Grundsätzlich kann die Bewertung in verschiedenen Stufen und Kategorien erfolgen, von denen im Folgenden einige vorgestellt werden sollen. Dabei wird danach unterschieden, wer die Bewertung durchführt.

**Automatisiert** Eine automatisierte Überprüfung von neuem Inhalt kann verschiedene Aspekte prüfen. Zum einen kann der Umfang des eingereichten Inhalts geprüft werden, aber auch erste Aussagen über die Qualität können getroffen werden. Es existieren verschiedene Tools, mit denen Rechtschreibung, Grammatik und auch Verständlichkeit

geprüft werden können. Vor allem die KI-Technologien können hier in Zukunft auch noch große Verbesserungen bringen (siehe Kapitel Einsatz von KI). Eine automatisierte Prüfung sollte aber immer nur der erste Schritt vor einer anschließenden manuellen Prüfung sein und nicht als alleinige Instanz eine Bewertung durchführen.

**Plattformnutzer** Auch Nutzer der Plattform können neuen Inhalt prüfen. Sie können beispielsweise den Aufbau des Kurses bewerten oder auch beurteilen, wie verständlich der neue Inhalt für sie ist. Die Qualität von Nutzerbewertungen kann allerdings schwanken, da man grundsätzlich nicht weiß, wie viel Erfahrung und Wissen die Nutzer haben. Dem kann entgegengewirkt werden, indem nur Nutzer mit Erfahrung auf der Plattform Bewertungen abgeben können oder auch nur Nutzer ein Feedback zu einem Kurs abgeben können, wenn sie diesen bestanden haben. Bei der Bewertung durch Plattformnutzer sollte darauf geachtet werden, dass diese durch ihre Art der Bewertung keine Vor- oder Nachteile erhalten können.

**Experten** Um das Vertrauen in eine Review weiter zu erhöhen, sollten Experten eingesetzt werden. Sie können vor allem die Qualität des Inhalts bewerten und fachliche Fehler erkennen. Dies ist die höchste Stufe einer Review, welche die beste Qualität erzielt. Allerdings ist sie vermutlich auch die schwierigste Stufe, da es weniger Experten als Nutzer gibt und dieses ggf. auch eine Bezahlung für ihre Fachexpertise verlangen.

Das Ergebnis einer Review kann für verschiedene Zwecke verwendet werden. Zum einen kann es, wie bereits im vorherigen Unterkapitel vorgestellt, für die Bestimmung der Höhe der Bezahlung genutzt werden. Es kann aber auch eine allgemeine Voraussetzung sein, um einen Inhalt überhaupt auf die Plattform aufzunehmen und eine Bezahlung (unabhängig der Form) zu ermöglichen. Auch für die Nutzer kann das Ergebnis einer Review zugänglich gemacht werden und ihnen bei der Wahl eines Kurses helfen. Für den Inhaltersteller kann das Feedback ebenfalls lehrreich sein. Eine weitere Möglichkeit wäre, dass die Review auch nach Veröffentlichung auf Verlangen des Erstellers durchgeführt wird. Das Ergebnis dieser Bewertung kann der Ersteller dann als eine Art Gütesiegel für die Bewerbung seiner Kurse nutzen. In diesem Fall kann von dem Ersteller eine Bezahlung verlangt werden. Grundsätzlich ist eine Prüfung von neuem Inhalt allerdings vor Veröffentlichung sinnvoller. Mit dem Ergebnis kann dann bspw. entschieden werden, ob der Inhalt veröffentlicht wird, wie gut er vergütet wird und es kann auf der Plattform selbst veröffentlicht werden, als zusätzliche Information für Nutzer.

Im vorherigen Unterkapitel wurde bereits auf die Herausforderung eingegangen, dass solche Prozesse abgesichert werden müssen. Es muss dabei sowohl vermieden werden, dass der Nutzer nicht die Plattform oder den Inhaltersteller ausnutzt als auch, dass der Inhaltersteller nicht die Plattform missbraucht. Angriffe jeder Form sollten bestmöglich bereits im Konzept verhindert werden. Es kann beispielsweise problematisch sein, den neuen Inhalt zum Bewerten für alle Teilnehmer zugänglich zu machen, da für sie dann später kein Grund mehr besteht, den Kurs tatsächlich auch zu kaufen. Das kann je nach Modell für Ersteller und Plattform ein Problem werden. Aber auch ein Ersteller kann durch

Spamming die Prozesse der Plattform verlangsamen, wenn jedes Erstellen von Inhalten eine manuelle Prüfung erfordert. Daher ist eine vorherige automatische Prüfung sinnvoll und hilfreich.

Bei einer Bewertung durch Nutzer kommt außerdem hinzu, dass die Motivation dafür unklar sein kann. Ein Nutzer kann beispielsweise der Plattform schaden, indem er schlechten Inhalt (absichtlich) sehr gut bewertet oder er kann dem Ersteller zusätzlich schaden, indem er guten Inhalt schlecht bewertet. Vor allem wenn Nutzer das Preis-Leistungs-Verhältnis bewerten müssen, kann es zu einer Verzerrung kommen, da Nutzer meist daran interessiert sind, gute Inhalte für möglichst wenig Geld zu bekommen. Unabhängig davon braucht es auch grundlegend einen Anreiz für die Nutzer, an einer Bewertung teilzunehmen. Bei der Einführung eines Belohnungssystems muss allerdings auch wieder verstärkt auf die Sicherheitsaspekte geachtet werden, da es zu höheren Verlusten für die Plattform kommen kann und für die Nutzer ein Angriff mehr Gewinn bedeuten kann.

Diese Gründe sprechen dafür, dass nicht nur die Nutzer allein an einem Bewertungsprozess sich beteiligen, sondern zusätzlich auch Experten teilnehmen. Auf diese Weise kann nicht nur die Qualität der Review verbessert werden, sondern es kann auch eine Aussage über die Bewertungen der Nutzer getroffen werden. Mit diesem Mechanismus kann den vorherigen Szenarien, dass Nutzer absichtlich falsch bewerten, entgegengewirkt werden. Außerdem können Experten besser die fachliche Korrektheit von Inhalten bewerten, während Nutzer eher die Verständlichkeit und die ansprechende Darstellungsform bewerten können.

Letztendlich zeigt sich, dass ein Reviewprozess für eine Lernplattform von großer Bedeutung ist, wenn sie sowohl die Inhalterstellung externen Lehrenden ermöglichen als auch ein hohes Niveau der Inhalte gewährleisten möchte. Zudem kann er als ein wichtiger Bestandteil des Vergütungsprozesses angesehen werden. Im folgenden Unterkapitel werden beide Prozesse auf ihre Umsetzbarkeit mit einer DAO geprüft.

### 3.2.3 Bewertung einer möglichen Integration

Die Schnittmenge zwischen dem Bezahlungsprozess und einer DAO liegt vor allem in den Token einer DAO. Diese können dafür genutzt werden, einen Ersteller von Inhalten für die Plattform der DAO zu bezahlen. Dieser Vorgang kann in einem Smart Contract abgebildet werden. Darin können die Bedingungen und die Konditionen für die Bezahlung transparent festgelegt werden. Die Vergütung erfolgt dann aus den Token-Beständen der DAO. Über diese Bestände sollten die Mitglieder der DAO gemeinsam entscheiden. Entsprechend muss vor der Bezahlung eine Einigung gefunden werden.

Eine Abstimmung über die Voraussetzungen und die Höhe der Bezahlung kann entweder vor oder nach Inhalterstellung erfolgen. Beide Ansätze haben ihre Vorteile, sind aber auch mit gewissen Risiken verbunden. Eine Herausforderung, die bei einer Abstimmung über die Bezahlung nach Inhalterstellung erfolgt, ist, dass der Inhaltersteller ausgenutzt werden

könnte. Denn er hat den Inhalt bereits erstellt und diesen gegenüber der DAO-Mitglieder veröffentlich, welche dann bspw. über die Höhe der Bezahlung abstimmen. Allerdings haben sie dann bereits den Inhalt gesehen und könnten eine geringe Bezahlung festlegen, ohne dass sie den Inhalt verlieren könnten. Auf diese Weise ist das Risiko hoch, dass der Inhaltersteller unfair bezahlt wird. Erfolgt hingegen die Abstimmung über die Bezahlung vor der Inhalterstellung kann das verhindert werden. Dafür ist es dabei schwerer, die Qualität des Inhalts einzuschätzen und entsprechend die Höhe der Bezahlung anzupassen. Hierfür müsste ein Mittelweg gefunden werden, bei dem beispielsweise nur einzelne Kapitel des Inhalts mit einer Gruppe der DAO-Mitglieder geteilt wird, welche dann über die Bezahlung abstimmen.

Beim Reviewprozess liegt die Schnittmenge zwischen dem Prozess und einer DAO im Voting bzw. in der gemeinsamen Bewertung von neuem Inhalt in Form einer Abstimmung. Hierbei würde die Review also von den Mitgliedern der DAO übernommen werden, welche damit auch entscheiden können, ob ein Inhalt eine ausreichende Qualität für die gemeinsame Plattform hat. Dabei können innerhalb der DAO auch die Unterscheidung der Rollen Nutzer und Experte abgebildet werden. Mit den Rollen können auch in den Smart Contracts der DAO die Berechtigungen festgelegt werden und dabei könnte bspw. den Stimmen von Experten mehr Gewicht gegeben werden.

Die Abgabe der Bewertung würde also On-Chain erfolgen und ist so für alle transparent. Das steigert auch für Plattformnutzer das Vertrauen in die Bewertungen, da sie nachvollziehen können, wie die Kursbewertung zustande kommt. Dabei sollte für jeden Bewertungsprozess nur eine Gruppe von Nutzern und Experten ausgewählt werden, sodass nicht direkt alle Mitglieder der DAO Zugriff darauf erhalten. Zusätzlich können die Mitglieder der DAO so entlastet werden, wenn nicht erwartet wird, dass sie an jedem Bewertungsprozess teilnehmen müssen. In der Auswertung der abgegebenen Bewertungen können die Reviews der Nutzer, mit denen der Experten verglichen werden. Bei großen Abweichungen kann es sinnvoll sein, diese Bewertung nicht mit einzubeziehen und ggf. auch den Nutzer zu bestrafen/auszuschließen.

Die Token der DAO können auch hierbei eine Rolle spielen. Zum Beispiel könnten Nutzer, welche für einen Bewertungsprozess ausgewählt werden, einen gewissen Betrag an Token hinterlegen. Weicht ihre Bewertung stark von denen der Experten ab, könnten diese Token dann als Strafe einbehalten werden. Auf der anderen Seite könnte der Besitz von Token auch als Parameter für die Auswahl der Nutzer für einen Bewertungsprozess dienen. Je mehr Token ein Mitglied hat, desto höher ist die Wahrscheinlichkeit, dass er an einer guten Qualität der Plattform interessiert ist. Schließlich will er nicht, dass seine Token an Wert verlieren. Entsprechend kann auf die Bewertung von Mitgliedern, die mehr in die DAO investiert haben, auch eher vertraut werden. Es kann allerdings nicht ausschließlich auf diese Annahme vertraut werden, da es auch Angreifer geben kann, die Verluste in Kauf nehmen, um der DAO zu schaden.

## 3.3 Fallstudien

In diesem Unterkapitel soll das Ergebnis der Recherche über existierende Projekte vorgestellt werden. Dabei wurden Projekte betrachtet, welche Blockchain im Allgemeinen bzw. DAO im Spezifischen auf einer Lernplattform einsetzen. Ziel ist es, einen ersten Eindruck für die Komplexität der Verknüpfung von E-Learning und DAO zu bekommen. Außerdem können die Projekte als Inspiration im Positiven, welche Ansätze und Mechanismen könnten übernommen werden, wie im Negativen, welche Herausforderungen gibt es, dienen.

### 3.3.1 Ed3 DAO

Im Projekt Ed3 DAO[2] sollen die Lehrenden (Educators) im Vordergrund der Plattform stehen. Sie sollen in der Lage sein dort ihre Projekte und Kurse bekannt zu machen und schließlich auch zu veröffentlichen. Dadurch sollen sie mehr Einfluss auf die Bildungswelt erhalten und deren Zukunft durch Zusammenarbeit mit anderen Pädagogen mitgestalten. (Saraf und Peck) Die DAO wird eingesetzt, um Entscheidungen zu treffen und zur Verwaltung des Projektes. (Meyer und Saraf 2022)

Die Lehrenden können in Abstimmungen darüber entscheiden, wie Budgets verteilt werden, wie Stipendien vergeben werden, aber auch Einsatz neuer Technologien. Die Abstimmungen werden mit einem Governance-Token durchgeführt, welcher keinen monetären Wert besitzt und jedes DAO-Mitglied nur genau einen besitzt. (Saraf und Peck 2022a) zusätzlich gibt es auf der Plattform den Social-Token, welcher wiederum einen monetären Wert besitzt. Mit dem Kauf dieser wird man Mitglied der DAO. Über die Anzahl der gekauften Ed3 DAO Social-Token kann man verschiedene Level der Mitgliedschaft in der DAO erreichen. (Saraf und Peck 2022a)

Als Tool für das Voting wird SafeSnap[3] genannt. Dies ist eine Erweiterung zu Snapshot - einer Plattform, welche Abstimmungsprozesse für Blockchain-Anwendungen anbietet. (SafeSnap: Reality) Zusätzlich zu Abstimmungen, an denen alle Mitglieder teilnehmen können, finden vor allem zwischen gewählten Mitgliedern der DAO (sog. Nodes) Off-Chain Abstimmungen statt. Nodes können während ihrer „Amtszeit“ zusätzliche Entscheidungen treffen– bspw. über Eventplanung, Webseitengestaltung, aber auch Blockchain Protokolle. (Saraf und Peck 2022c) Wie groß ihre Entscheidungsmacht dadurch tatsächlich ist, geht aus den Projektinformationen nicht direkt hervor, es ist allerdings möglich, dass dadurch der dezentrale Charakter eingeschränkt wird. Zusammengefasst können Lehrende sich auf Ed3 DAO vernetzen, eigene Projekte vorschlagen und entwickeln

[2] https://www.ed3dao.com/

[3] https://docs.snapshot.org/user-guides/plugins/safesnap-reality

sowie dafür Unterstützung bekommen (in Form von Ressourcen oder finanziell) und haben dabei durch den Einsatz der DAO ein Mitbestimmungsrecht.

Neben der Frage, wie dezentral diese DAO tatsächlich ist, bleibt außerdem unklar, wie eine gemeinschaftliche Entscheidung über zu unterstützende Projekte sinnvoll getroffen werden soll. Schließlich scheint es als Pädagoge mit eigenem Projekt in der Ed3 DAO kontraproduktiv ein anderes Projekt mit den finanziellen Mitteln der DAO unterstützen zu wollen. Zum einen, weil dadurch ein eventuelles Konkurrenzprojekt gestärkt wird und zum anderen, weil Mittel genutzt werden, die man selbst dem eigenen Projekt zuschreiben wollen würde. Daraus wiederum könnte sich ergeben, dass entweder keine Entscheidungen getroffen werden können innerhalb einer Abstimmung oder die Entscheidung durch Nodes gefällt wird, die nicht Lehrende sind. Dadurch würde wiederum ein Teil der Dezentralität verloren gehen und auch die Vision einer von Educators gestalteten Welt kann infrage gestellt werden.

Als Inspiration für eine Lernplattform im Allgemeinen kann die Idee der sogenannten „Unconference“ von Ed3 DAO dienen. Hierbei wurde eine Austausch- und Vernetzungsmöglichkeit speziell für Bildungsexperten, Innovatoren und Pädagogen geschaffen, welche Lernlabore, Vorträge und Networking gamifiziert im Stile eines Metaverse. (Saraf und Peck 2022b)

### 3.3.2 ODEM

ODEM[4] (On-Demand Education Marketplace) ist ein Schweizer Unternehmen, welches eine online Lernplattform anbietet. Dabei soll eine Verbindung geschaffen werden zwischen der Nachfrage durch Studierende und dem Wissen von Lehrkräften, welche dann entsprechende Kurse erstellen. Einige Prozesse der Plattform basieren auf der Ethereum-Blockchain in Verbindung mit dem IPFS. (Narma 2018)

Im Konzept des Projekts wird nicht direkt von einer DAO gesprochen, dennoch kann ihr Ansatz „Education Program Staking“ in Verbindung mit ERC20 Token (ODE) als Inspiration für eine Entscheidungsfindung in einer dezentralen Gemeinschaft dienen. Das Staking dient dabei der Bewertung der Interessen von Plattformnutzern, also Studierenden und auch der von Inhalterstellern (Educators). Zunächst bringt ein Educator einen Vorschlag für seinen Kurs oder sein Event auf die Plattform und hinterlegt dabei einen Prozentsatz des vermuteten Einkommens (basierend auf Kursplätze und Kurspreis). Daraufhin können Nutzer ihr Interesse an dem Inhalt bekunden, indem sie ebenfalls etwa zehn Prozent des Kurspreises anzahlen. Dabei ist zu beachten, dass Nutzer diese Anzahlung nicht zurückerhalten können (nach 24 Stunden Frist), während Inhaltersteller ihren Vorschlag noch zurückziehen können. Erst nachdem das Staking-Ziel durch Plattformnutzer

[4] https://odem.cloud/

erreicht wurde, muss sich der Ersteller final entscheiden. Erstellt und veröffentlicht er den Kurs bzw. hält das Event, wird die restliche Bezahlung von den Nutzern eingefordert. Nur wenn sie dann die restlichen neunzig Prozent zahlen, erhalten sie auch Zugriff auf den Inhalt. Andernfalls verfallen die angezahlten zehn Prozent. Entscheidet sich der Ersteller gegen die Umsetzung seines Kurses erhält der Nutzer die Anzahlung zurück. Nach Abschluss des Lehrinhalts erhält der Nutzer schließlich ein Blockchain-basiertes Zertifikat und der Inhaltersteller erhält seine hinterlegten ODE zurück. (ODEM.IO 2018)

Neben diesem interessanten Ansatz basiert das Projekt auf einer Online-Plattform mit akademischen Inhalten, welche mit der Ethereum-Blockchain verbunden ist. Auf der Blockchain werden unter anderem mithilfe von IPFS die Zertifikate abgelegt. Ziel bei der Plattform ist es, verschiedene akademische Inhalte kombinieren zu können und so zielgerichtet Inhalte verschiedener Universitäten konsumieren zu können. (The Analyst Team 2019)

Insgesamt sticht vor allem die Idee des Stakings aus dem Konzept hervor. Wobei hier Vorteile und Nachteile betrachtet werden. Vorteilhaft ist die Bindung der Nutzer sowohl an die Plattform als auch an die Interessenbekundung. Ein Nutzer müsste viel investieren, um alle Inhalte als interessant zu markieren. Daher steigt die Aussagekraft der Interessensbekundungen durch die Nutzer, vor allem da sie ihren Einsatz nicht zurückziehen können. Ein Nachteil hingegen ist, dass sich Interessen durchaus ändern können und der Prozess des Stakings dafür zu lange dauern kann. Vor allem dadurch, dass ein Inhaltersteller eventuell erst nach Ende des Stakings seine Inhalte auch wirklich umsetzt und danach erst zur Verfügung stellt. Außerdem ist die Gefahr da, dass trotz einer ernsthaften Interessenbekundung und dem Abwarten bis zum Erfüllen des Stakingziels der Educator seinen Vorschlag zurückzieht. Dabei würde er nicht einmal Konsequenzen fürchten müssen.

Zusammengefasst ist das Hinterlegen einer Anzahlung auf einen Kurs als Interessensbekundung eine reizvolle Idee für Inhaltersteller und Plattformbetreiber, allerdings auch mit vielen Risiken und Nachteilen für die Nutzer verbunden. Inklusive der Hürde, dass sie, bevor sie Inhalte konsumieren können, zumindest in der Anfangsphase erst Token kaufen müssen, ohne zu wissen, welche Qualität der Inhalt haben wird.

Daher ist es wenig verwunderlich, dass sich genau dieses Konzept zum jetzigen Stand nicht auf der Plattform von ODEM wieder findet. Stattdessen wird zum Matching von Interessen und Kursinhalten ein sogenanntes SkillPrint Feature eingesetzt, welches auf KI aufbaut. Die einzige offensichtliche Verbindung zur Blockchain scheint die Ablage der Zertifikate zu sein. (ODEM.Cloud 2023) Die Stakingmöglichkeit wird also nicht mehr von der Plattform angeboten, obwohl sie laut eigenem Feature Release 2019 entwickelt wurde. (Slavin 2019)

Ein zusätzlicher Punkt, welcher bei der Betrachtung des aktuellen Kursangebots der ODEM-Plattform[5] auffällt, ist die Bewertungsform der Kursinhalte. Es wird mit einer 5-Sterne-Skala gearbeitet und zusätzlich die Anzahl der Bewertungen angegeben. Mehr Informationen erhält man jedoch nicht, weder schriftliche Bewertungen noch Kategorien werden angegeben. Außerdem fällt auf, dass etwa die Hälfte der Kurse keine Bewertung hat. Dies wird entsprechend als Null-Sterne-Bewertung gerankt und wirkt eher abschreckend. Entsprechend schwerer haben es also neue Kurse auf der Plattform und auch als Nutzer ist ein Vergleichen der Qualität der Kursinhalte nur schwer möglich.

Letztendlich zeigt das Ergebnis auch, wie viel Wandel ein Projekt über wenige Jahre erfährt. Mit seinem Konzept konnten es während des Presales und des Main Token Events 2018 fast 10 Mio. Dollar einnehmen. Von den beschriebenen Konzepten fehlen allerdings in der heutigen Umsetzung viele und entsprechend unterscheidet sich die ODEM-Plattform nur wenig von anderen Lernplattformen, welche Kurse mit Zertifizierung anbieten. (ODEM.IO 2018)

[5] https://odem.cloud/program-catalog

# 4 Entwicklung eines Konzepts

Für den Ablauf der Bewertung wurde ein Konzept entwickelt, welches in diesem Kapitel vorgestellt wird und in die Abschnitte Vorbereitung, Bewertungsprozess Nutzer und Experte und Auswertung gegliedert werden kann. Es sollen für jeden Prozess verschiedene Möglichkeiten beleuchtet und miteinander verglichen werden, um die Entwicklung des Konzepts und dessen Herausforderungen sichtbar zu machen. Dabei wurden die Erkenntnisse aus den vorhergehenden Kapiteln genutzt und Ansätze übernommen und weiterentwickelt.

## 4.1 Vorbereitende Prozesse

Bevor der eigentliche Bewertungsprozess beginnt, gibt es eine Vorbereitungsphase. Sie startet damit, dass der Inhaltersteller neue Inhalt zur Bewertung freigibt. Der Inhalt ist dabei bereits als Kurs gegliedert und sollte Lessons mit Topics beinhalten. Diese Annahme basiert auf dem Konzept der Lernplattform der BCAM. Zusätzlich zu dem Inhalt wird durch den Inhaltersteller eine Angabe von Schlagwörtern bezüglich des Themas des neuen Kurses gefordert. Zusätzlich wird auf die Einsatzmöglichkeiten von KI und maschinellem Lernen eingegangen.

### 4.1.1 Vorbereitende Prüfung

Ein erstes Problem ist, dass der Ersteller Inhalte für den Prozess freigibt, welcher unvollständig oder gar leer ist. Um den Bewertungsprozess und alle daran Beteiligten nicht unnötig mit leeren Inhalten zu belasten, sollte die erste Prüfung des Inhalts bereits vor dem eigentlichen Bewertungsprozess stattfinden. Diese Prüfung kann idealerweise automatisiert erfolgen. Dabei können verschiedene Faktoren bereits automatisiert geprüft werden:

1. Umfang des eingereichten Inhalts – Es ist ausreichend Inhalt (basierend auf Wortanzahl, Lesson- oder Topicanzahl oder auch Videolänge) eingereicht wurden.
2. Strukturierung des Inhalts – Der Inhalt wurde in Lessons und Topics gegliedert. Zusätzlich könnte auch auf Formatierungsrichtlinien geprüft werden.
3. Grammatik und Rechtschreibung – Hier kann vor allem auch ein erster Eindruck über die Korrektheit des Textes erhalten werden und wie gewissenhaft der Inhalt erstellt wurde.
4. Verständlichkeit – Zusätzlich kann der Text auf seine Lesbarkeit geprüft werden, basierend auf Satzlänge, Satzaufbau und verwendeten Wörtern.

Wie umfangreich diese Prüfung sein kann, hängt zum einen vom Projekt und den verfügbaren Mitteln ab, aber auch davon, wie viele neu eingereichte Inhalte zu erwarten sind. Ist nur eine überschaubare Menge an Inhalten zu erwarten, kann die automatisierte Prüfung weniger Inhalt aussortieren und mehr dem manuellen Bewertungsprozess überlassen. Dann kann beispielsweise auch auf die Entwicklung einer automatisierten Prüfung auf Verständlichkeit verzichtet werden, wenn dies eh im anschließenden Bewertungsprozess betrachtet wird. Auch die Prüfung auf die Strukturierung des Inhalts und der Formatierungsrichtlinien kann vereinfacht werden, indem der Inhaltersteller seinen Inhalt in eine vorgegebene Maske eintragen muss. Dabei können bereits während der Inhalterstellung Überprüfungen stattfinden, ob beispielsweise ein Topic ausreichend Inhalt hat. Entsprechend ist dann auch ein direktes Feedback an den Ersteller möglich.

Dieser Schritt kann je nach Anforderung sehr komplex werden, wenn beispielsweise auch auf Plagiate und Ähnlichkeit zu anderen Inhalten der Plattform getestet wird oder auch die Korrektheit von den genannten Fakten. In diesem Fall kann der Einsatz von komplexeren Algorithmen bis hin zu künstlicher Intelligenz nötig sein. Auf diese Möglichkeit wird im Kapitel Einsatz von KI näher eingegangen.

Am wichtigsten bei diesem Schritt ist allerdings eine grundlegende erste Prüfung, dass Inhalt überhaupt vorhanden ist, bevor der Prozess der eigentlichen Bewertung gestartet wird. Schlägt die automatisierte Prüfung fehl, wird der Inhalt in den Bearbeitungsstatus zurückgesetzt. Weitere mögliche Folgen einer fehlgeschlagenen Prüfung werden im nächsten Kapitel betrachtet.

### 4.1.2 Spamming durch Inhaltersteller

Die Herausforderung von leeren und unzureichenden Inhalten lässt sich noch weiter fassen auf Inhaltersteller, welche das System mit Spamming belasten wollen. Beispielsweise könnte ein Ersteller auch immer wieder leeren Inhalt in Prüfung geben und damit zumindest die Vorprüfung immer wieder starten, wenn es keine Konsequenzen für ihn daraus gibt. Schlägt also die automatisierte Prüfung bei einem Inhaltersteller wiederholt immer wieder fehl, sollte das zu einer Bestrafung führen. Das kann eine Sperre über eine bestimmte Zeit sein oder auch der komplette Ausschluss vom Prozess zum Veröffentlichen neuer Inhalte. Auch bereits vorhandene Inhalte des Erstellers könnten als Bestrafung von der Plattform entfernt werden, allerdings würde dadurch auch der Plattform geschadet werden. Wichtiger ist es, dass das Spamming unterbrochen wird, anstatt einer möglichst hohen Strafe.

Wie oft ein Inhaltersteller bei der Prüfung fehlschlagen darf, bevor er eine Sperre erhält, kann verschieden festgelegt werden. Zum einen könnte es eine Rolle spielen, wie viele Inhalte er bereits auf der Plattform veröffentlicht hat oder auch, wie viel Zeit zwischen fehlgeschlagenen Versuchen vergangen ist. Aber auch ein simpler Ansatz kann bereits ausreichend wirkungsvoll sein - also, dass nach drei fehlgeschlagenen Prüfungen eine Sperre

von sieben Tagen verhängt wird und nach drei weiteren fehlgeschlagenen Prüfungen der Inhaltersteller gesperrt wird. Mit diesem strengen Konzept kann vor allem auf einer neuen Plattform mit vielen neuen Inhalterstellern, schnell zwischen Angreifern und gewissenhaften Inhalterstellern differenziert werden. Eine Anpassung dieses Konzepts kann bei Wachstum der Plattform sinnvoll sein.

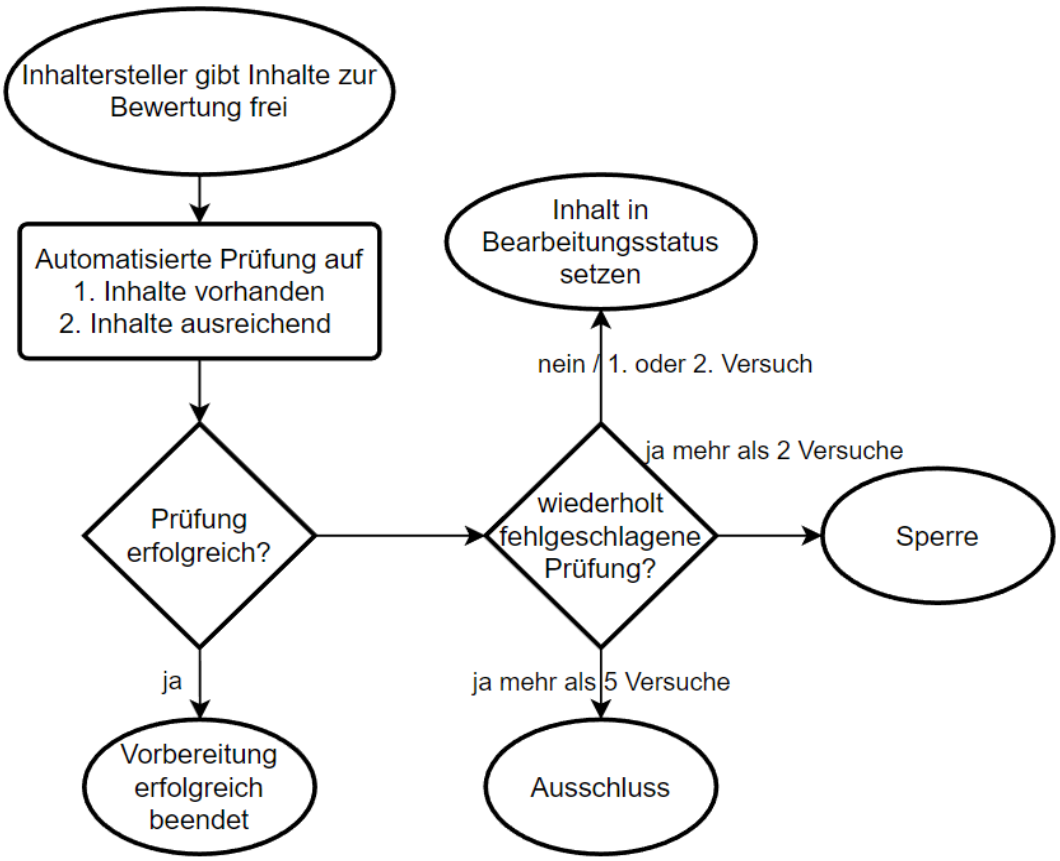



**Abbildung 1: Diagramm Vorbereitungsprozess**

Grundlegend ist also die wichtigste Maßnahme gegen Spammer, dass sie den Zugriff auf das System (vorrübergehend) verlieren. So kann eine Wartepause zwischen zwei Inhaltsfreigaben bereits sehr wirksam sein. Voraussetzung dafür, sowie für die vorgestellten Konsequenzen ist eine vorherige Authentifizierung des Inhaltserstellers. Auch hierfür gibt es verschiedene Herangehensweisen.

Zum einen kann ein einfacher Ansatz in Form einer Registrierung mit E-Mail-Adresse und Passwort genutzt werden. Dabei sollte allerdings ein Captcha eingesetzt werden, um automatisierten Anmeldungen mit automatisch generierten E-Mail-Adressen entgegenzuwirken. Zum anderen könne auch komplexere Ansätze genutzt werden, bis hin zu Blockchain basierten ID-Lösungen. Dabei könnte beispielsweise auch das Projekt ID-Ideal[6] zum

[6] https://id-ideal.de/

Einsatz kommen. Wichtig bei der Entscheidung für oder gegen eine Vorgehensweise ist das Beachten der Plattformkompatibilität und der damit verbundene Aufwand von Ressourcen bei der Integration. Insofern kann der erste Ansatz vor allem zu Beginn eine gute erste Lösung sein und erst bei einer Weiterentwicklung auf komplexe Ansätze gesetzt werden.

### 4.1.3 Vergabe von Schlagwörtern

Der Einsatz von Kategorien oder Zuordnung von Schlagwörtern zu Inhalten auf einer Lernplattform macht aus vielen Gründen Sinn. Zum einen können die Lerninhalte so besser organisiert werden und auf der Plattform strukturiert dargestellt werden. Zusätzlich kann eine Such- und Filterfunktion darauf aufbauen und es den Nutzern erleichtern, bestimmte Themen zu finden und zugehörige Inhalte zu vergleichen. Außerdem können darauf aufbauend personalisierte Empfehlungen ausgegeben werden, basierend auf bspw. angegeben Interessen oder bereits absolvierten Kursen.

Letztendlich können sie auch bereits für den Bewertungsprozess eine Rolle spielen. Sie können dafür eingesetzt werden, dass Nutzer oder Experten basierend auf ihrer Expertise Kurse mit ähnlichen Inhalten bewerten. Bei Nutzern könnten dafür bereits abgeschlossene Kurse verwendet werden und bei Experten eine Selbstauskunft genutzt werden. Insbesondere bei den Experten ist der Einsatz sinnvoll, da sie vor allem für die Prüfung auf korrekten Inhalt zuständig sind. Währenddessen ist es bei Nutzern auch von Vorteil, wenn sie im zu prüfenden Kurs noch nicht viel Vorwissen haben, um bei der Beurteilung der Verständlichkeit unvoreingenommen zu sein.

Zu beachten ist allerdings, dass nicht ausschließlich über die Schlagwörter die Wahl der Experten geschieht. Denn es soll verhindert werden, dass der Ersteller sich seine Experten dadurch selbst aussucht. Um dem entgegenzuwirken, müssen weitere Faktoren bei der Auswahl der Experten einbezogen werden, welche der Inhaltersteller bestenfalls nicht kennt oder zumindest nicht beeinflussen kann. Auf diese Herausforderung wird im Kapitel Bewertungsprozess Experte näher eingegangen werden.

Für den Einsatz beim Bewertungsprozess muss die Angabe der Schlagwörter für einen neuen Inhalt entsprechend direkt mit der Freigabe des Inhalts für den Bewertungsprozess erfolgen. Und demzufolge ist der Inhaltersteller dafür zuständig. Dabei ist zu entscheiden, ob die Plattform Schlagwörter vorgibt oder die Ersteller diese frei eingeben können. Eine bessere Übersicht und klarere Kategorisierung spricht für die erste Option, eine höhere Flexibilität auch bei neuen Inhalten spricht die zweite Option. Es kann also sinnvoll sein, eine erste Auswahl an Schlagwörtern vorzugeben und diese kontinuierlich zu erweitern oder eine hybride Methode zu wählen, bei welcher die Ersteller sowohl vorgegebene Kategorien wählen können als auch eigene Schlagwörter einbringen können.

Während eine falsche Wahl dem Ersteller wenig Vorteile bringen wird, kann die Auswahl möglichst vieler Schlagwörter (die mehr oder weniger passen) durchaus sinnvoll für ihn sein. Denn so wird der Kurs bei möglichst vielen Suchen und in vielen Kategorien angezeigt. Entsprechend sollte es zum einen eine Einschränkung bei der Anzahl von Schlagwörtern geben und zum anderen sollte im Bewertungsprozess auch die Übereinstimmung von Inhalt und Schlagwörtern eine Rolle spielen.

Zusammengefasst ist also der Inhaltersteller für die Auswahl von passenden Schlagwörtern zu seinem Inhalt zuständig und gibt diese gemeinsam mit seinem Inhalt für die Bewertung frei. Diese Schlagwörter werden sowohl zum Teil bei der Auswahl der Experten eingesetzt als auch beim Organisieren und Anzeigen der Inhalte auf der Plattform. Ob die Auswahl passend war, wird während des Bewertungsprozesses mit beachtet, um einer (absichtliche) Fehlauswahl des Inhalterstellers entgegenzuwirken.

### 4.1.4 Einsatz von KI

Eine wichtige Technologie, welche an dieser Stelle sinnvoll durch die Plattform eingesetzt werden könnte, aber sie genauso auch vor neue Herausforderungen stellt, ist künstliche Intelligenz bzw. maschinelles Lernen. Als grobe Einordnung kann gesagt werden, dass beim maschinellen Lernen ein Computer selbstständig lernt, wie er Daten verarbeiten soll, anstelle dass ein Modell vorgegeben wird. Stattdessen werden gewisse Kriterien sowie Datensätze vorgegeben. Durch wiederholtes Ausprobieren und dem Erreichen oder Verfehlen von vorgegebenen Zielen wird von dem Computer selbst ein Vorgehen entwickelt. Mit Projekten wie ChatGPT[7] wird diese Technologie immer mehr Menschen zugänglich gemacht und Einsatzmöglichkeiten werden greifbarer. (Fraunhofer-Institut für Kognitive Systeme IKS 2023)

In den vorangegangenen Betrachtungen hat sich gezeigt, wie wichtig eine automatisierte Überprüfung vor der eigentlichen Bewertung sein kann. Vor allem sollen dadurch die manuellen Prozesse entlastet werden. Durch den Einsatz von maschinellem Lernen kann die Überprüfung wesentlich umfassender werden als durch herkömmlicher Algorithmen. Folgende Aspekte können vor allem durch Verwendung von maschinellem Lernen (besser) umgesetzt werden.

1. Vorprüfung des Inhalts auf Sinnhaftigkeit
2. Prüfung des Inhalts auf Ähnlichkeit zu anderen Inhalten
3. Prüfung, ob Inhalt mit KI generiert wurde
4. Automatisiertes Festlegen von Schlagwörtern und Schlagwortorganisation
5. Neue Inhalte mit KI generieren

[7] https://chat.openai.com/

Die Übersicht zeigt, dass vor allem bei der Prüfung von neuen Inhalt der Einsatz von KI hilfreich sein kann. So können Kurse, welche zwar Inhalt in Form von Wörtern und Sätzen haben und somit die Prüfung auf ausreichenden Inhalt bestehen würden, welche aber auf den ersten (menschlichen) Blick keinen Sinn ergeben, bereits in der Vorprüfung herausgefiltert werden. Auch für die Prüfung auf ähnlichen Inhalt bzw. Plagiate kann durch maschinelles Lernen leichter durchgeführt werden. So kann auch sichergestellt werden, dass auf umfangreichen Lernplattformen, auf denen vielleicht selbst Nutzer keinen kompletten Überblick haben, zu viele ähnliche Kurse vermieden werden. Zusätzlich kann neuem Inhalt auch von einer KI, Schlagwörter direkt zugewiesen werden. Auf diese Weise muss nicht auf den Inhaltersteller vertraut werden, sondern in der Vorprüfung kann abgeglichen werden, welche Schlagwörter es auf der Plattform bereits gibt und welche zu dem neuen Inhalt passen können.

Bezüglich des Erstellens von Inhalten mit KI gibt es zwei Blickpunkte. Zum einen kann man als Plattform vermeiden wollen, dass andere Inhaltersteller KI-generierte Inhalte als neue Kurse veröffentlichen. Vor allem, wenn Inhaltersteller für ihre Arbeit entlohnt werden, kann der Einsatz von KI bei der Inhalterstellung als nicht fairer Vorteil gesehen werden. Hierfür können dann in der Vorprüfung Tools eingesetzt werden, welche KI-Texte erkennt. Zum anderen kann es als Plattform selbst aber auch spannend sein, selbst Inhalte auf die Plattform bringen zu können, indem KI verwendet wird. Vor allem die anschließende manuelle Bewertung des Kurses sichert dabei ab, dass der erstellte Inhalt qualitativ hochwertig und vor allem auch korrekt ist. Dieser Punkt ist bei allen Einsatzmöglichkeiten wichtig zu beachten. Eine KI soll die manuelle Prüfung des Inhalts nicht ersetzen, sondern als Filter oder Hilfestellung fungieren.

Vor allem die Kombination von KI zum Erstellen von Inhalten und einer DAO hat viel Potenzial. Innerhalb der DAO kann abgestimmt werden, welche Inhalte zu welchen Themen erstellt werden sollen und der Inhalt kann dann durch eine KI erstellt werden. Das Ergebnis wiederum kann mit einem entsprechenden Bewertungsverfahren in der DAO auf die Plattform aufgenommen oder abgelehnt werden. Durch dieses Vorgehen wäre die DAO unabhängiger von externen Inhalterstellern und könnte noch autonomer agieren. Neben der Erzeugung von neuen Lerninhalten könnte eine KI auch dafür eingesetzt werden, zusätzlichen Inhalt für die Plattform zu erzeugen. Beispielsweise kann sie nach einem vorgegebenen Muster die Kurs-Startseiten und auch Zusammenfassungen schreiben. Dadurch kann sie die Plattform durch bei der Bewältigung solcher Hintergrundprozesse unterstützen.

Alles in allem birgt der Einsatz von maschinellem Lernen und KI viele neue Optionen vor allem in der automatisierten Vorprüfung neuer Inhalte. Aber auch andere Szenarien sind spannend, wie eine KI basierende Lernplattform. Letztendlich bleibt abzuwarten, wie sich die Technologie weiterentwickelt und trotz ihres Potenzials weiterhin auch Überprüfungen durch Menschen durchzuführen.

## 4.2 Bewertungsprozess Nutzer

Nach der Vorbereitungsphase folgt die erste Bewertungsrunde. In dieser Runde sollen Nutzer der Plattform (Lernende), welche sich ausreichend mit der Lernplattform auseinandergesetzt haben, in den Prozess integriert werden.

### 4.2.1 Qualifizierte Nutzer

Für die Teilnahme am Bewertungsprozess muss sich ein Nutzer der Plattform qualifizieren. Um grundlegend als Nutzer der Plattform zu gelten, muss man sich auf der Plattform registrieren. Durch das Besuchen von Kursen und das erfolgreiche Beenden dieser sammelt ein Nutzer auf der Plattform Learntoken. Hat ein Nutzer ausreichend Learntoken auf der Plattform gesammelt, kann er qualifizierter Nutzer werden. Die Anzahl der benötigten Learntoken hängt von der Vergabe dieser auf der Plattform ab. Bei der Plattform der BCAM können etwa fünfzig bis hundert Token pro Kurs gesammelt werden. Ein möglicher Ansatz ist, dass ein Nutzer sich mit mindestens drei Kursen der Plattform auseinandergesetzt hat. Das würde zum aktuellen Stand etwa der Hälfte der verfügbaren Kurse der Plattform entsprechen. Ein Nutzer hätte also nach erfolgreichem Besuch von drei Kursen etwa 150 bis 300 Token. Zu Beginn der Plattform sollte eine niedrige Schwelle gewählt werden, daher könnten zum aktuellen Zeitpunkt auf der BCAM-Plattform 150 Token die Voraussetzung sein, um qualifizierter Nutzer zu werden. Mit Wachstum der Plattform und bei ausreichend bereits qualifizierten Nutzern könnte die Zahl auch erhöht werden.

Hat ein Nutzer ausreichend Learntoken gesammelt, muss er sich zusätzlich in seinem Profil dafür eintragen, dass er an dem Reviewprozess teilnehmen möchte. So soll verhindert werden, dass gute Nutzer mit vielen Learntoken im Reviewprozess ausgewählt werden, obwohl sie grundlegend daran gar kein Interesse haben. Nur Nutzer, welche beide Bedingungen erfüllen, sind qualifizierte Nutzer.

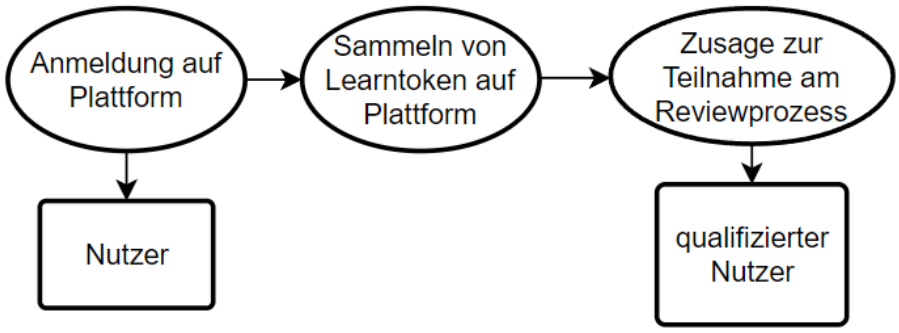



**Abbildung 2: Qualifizierungsprozess**

Innerhalb der Gruppe von qualifizierten Nutzer gibt es eine Rangliste. Diese soll abbilden, wie gut die bisher abgegebenen Bewertungen eines qualifizierten Nutzers waren. Die Qualität einer Bewertung ergibt sich während der Auswertung der Bewertungen von Nutzern und Experten. Grundlegend steigt das Ranking eines qualifizierten Nutzers durch die

Abgabe einer guten Bewertung und sinkt, wenn keine oder eine schlechte Bewertung abgegeben wurde. Das genau Vorgehen wird im Kapitel Rankingsystem erläutert werden. Dieses Ranking soll auch beim Auswahlverfahren der Nutzer für einen Bewertungsprozess eingesetzt werden.

### 4.2.2 Auswahlverfahren

Die Anzahl an qualifizierten Nutzern, die an einem Bewertungsprozess teilnehmen sollen $(n)$, hängt von den verfügbaren Ressourcen der Lernplattform ab. Wichtig ist vor allem die Gesamtzahl der qualifizierten Nutzer auf der Plattform $(x)$. Grundlegend sollte folgendes gelten:

$$x > n$$

So kann die erforderliche Anzahl qualifizierter Nutzer für einen Bewertungsprozess überhaupt erreicht werden. Unter Beachtung, dass eine gewisse Rotation innerhalb der ausgewählten qualifizierten Nutzer stattfinden sollte – zum einen, um Absprachen entgegenzuwirken, zum anderen, um den Nutzern Pausen zu ermöglichen, sollte ein Vielfaches der erforderlichen Anzahl an qualifizierten Nutzern zur Verfügung stehen:

$$x > k * n$$

Für den Faktor $k$ gilt, dass er größer gleich zwei sein sollte. Je höher der Faktor gewählt wird, umso geringer ist die Wahrscheinlichkeit ähnlicher Gruppen. Es sollte also die verfügbare Menge an qualifizierten Nutzern genutzt oder prognostiziert werden, um von ihr auf die Gruppengröße von qualifizierten Nutzern in einem Bewertungsprozess schließen zu können.

Auf einer Plattform, welche das Prinzip von Learntoken und qualifizierten Nutzern neu einführt, kann die Anzahl der Besucher von Kursen als Anhaltspunkt dienen. Gibt es bereits ein Konzept für Learntoken, allerdings noch keine Umsetzung für qualifizierten Nutzer, kann eine erste Prognose gestellt werden. Dafür wird betrachtet, wie viele Nutzer ausreichend Learntoken hätten und bei Interesse qualifizierte Nutzer werden können.

Auf der BCAM-Plattform ist noch kein Learntoken-Konzept vollständig integriert, daher müssen hierfür die Besucherzahlen zurate gezogen werden. Sinnvoll ist vor allem die Betrachtung, wie viele Besucher auch Kurse abschließen. Dafür wurde der Aufruf der jeweils letzten Kursseiten im ersten Quartal 2023 betrachtet, sodass nur aktuell aktive Besucher einbezogen werden. Diese Zahl schwankt zwischen den verschiedenen Kursen zwischen zwanzig und hundert. Es ist also davon auszugehen, dass zum aktuellen Zeitpunkt nur wenige Nutzer tatsächlich mindestens drei Kurse besucht haben, um ausreichend Learntoken gesammelt zu haben, um qualifizierter Nutzer zu werden. Mit dieser Grundlage könnte man zu Beginn von etwa zehn bis zwanzig qualifizierten Nutzern ausgehen.

Je nach Wahl des Faktors $k$ ergibt sich daraus, wie viele qualifizierte Nutzer an einem Bewertungsprozess teilnehmen sollten:

$$n < \frac{x}{k}$$

Ausgehend von $k = 2$ und $x = 10$ als Minimum würden also maximal vier qualifizierte Nutzer pro Bewertungsprozess teilnehmen können. Mit mehr qualifizierten Nutzern kann auch ein höherer Faktor genutzt werden, was erstrebenswert ist. Ausgehend von $k = 3$ und $x = 20$ würden also maximal sechs qualifizierte Nutzer pro Bewertungsprozess teilnehmen können. Als Kompromiss zwischen diesen beiden Werten soll für die Weiterentwicklung des Konzepts von $n = 5$ ausgegangen werden. Damit soll eine gute Genauigkeit im Bewertungsprozess, ausgehend von einer realistischen Betrachtung an Ressourcen erreicht werden. Mit einem guten Wachstum der Plattform in der Zukunft könnte diese Zahl auch erhöht werden, um eine höhere Genauigkeit zu erreichen.

Die Auswahl aus den qualifizierten Nutzern für einen Bewertungsprozess soll möglichst nicht vorhersagbar sein, um Abstimmungen zwischen Inhaltersteller und Bewerten zu erschweren. Daher benötigt das Auswahlverfahren eine Zufallskomponente. Die Wahrscheinlichkeit, ausgewählt zu werden, wird durch ein besseres Ranking erhöht werden. Allerdings nur in dem Maße, dass nicht vorhersagbar ist, dass genau die fünf Besten am nächsten Bewertungsprozess teilnehmen werden.

Zusätzlich zum Stand im Ranking sollte noch ein Zeitfaktor mit eingebracht werden. Andernfalls könnte es passieren, dass die Nutzer, welche in den ersten Bewertungsprozessen teilgenommen haben, immer wieder ausgewählt werden und nur selten neue Kombinationen an Bewertern entstehen. Dem beschriebenen Szenario kann entgegengewirkt werden, indem eine Abklingzeit genutzt wird. Das bedeutet, dass nach der Teilnahme an einem Bewertungsprozess eine gewisse Zeitspanne vergehen muss, bevor man wieder für eine Teilnahme ausgewählt werden kann. Diese Zeitspanne kann auf verschiedene Arten festgelegt werden

1. Zeitspanne entspricht einer Anzahl von Bewertungsprozessen, die abgewertet werden müssen.
2. Zeitspanne entspricht tatsächlich einer festgelegten Zeit, beispielsweise einem Monat.
3. Statt einer starren Zeitspanne wird die Wahrscheinlichkeit ausgewählt zu werden, auch daran angepasst, wann die letzte Teilnahme war.

Der Vorteil der dritten Variante ist, dass sie eine Vorhersage weiter erschwert. Die Wahrscheinlichkeit, ausgewählt zu werden, muss also direkt nach einer Teilnahme an einem Bewertungsprozess sinken. Anschließend kann sie dann mit jedem Prozess, an dem ein Teilnehmer nicht mitgewirkt hat, erhöht werden. Zu bevorzugen ist diese Variante gegenüber der zweiten Variante, da nicht vorhergesagt werden kann, in welchen Zeitabständen neue Inhalte bewertet werden müssen. Dabei ist die erste Variante allerdings auch eine valide Option, welche sich leichter umsetzen ließe als die dritte.

Zu beachten ist, dass sich auf die Periode seit der letzten Auswahl und nicht der letzten Teilnahme bezieht. Ein inaktiver Nutzer, welcher trotz Auswahl lange nicht mehr an einer Bewertung teilgenommen hat, sollte vom Prozess ausgeschlossen werden. Die genaue Auswirkung von Nichtteilnahme oder schlechten Bewertungen wird im Kapitel Rankingsystem behandelt.

Außerdem sollen auch Nutzer, welche sich neu qualifiziert haben, eine reelle Chance haben, an einem Bewertungsprozess teilzunehmen, um ebenfalls wiederum die Möglichkeit zu haben, im Ranking zu steigen. Das kann umgesetzt werden, indem von den fünf ausgewählten Nutzern mindestens einer aus der Gruppe von Neuzugängen stammt. Alternativ kann auch die Wahrscheinlichkeit, ausgewählt zu werden, erhöht werden, wenn ein Nutzer noch keine Bewertung abgegeben haben. Allerdings gäbe es dann die Möglichkeit, dass in einem Bewertungsprozess nur neue qualifizierte Nutzer teilnehmen. Vorteilhafter wäre eine gemischte Gruppe aus erfahrenen und neuen qualifizierten Nutzern, weshalb der erste Ansatz zu bevorzugen ist. Es wird also mindestens ein Nutzer unabhängig vom Ranking zufällig aus der Gruppe von neuen qualifizierten Nutzern ausgewählt. Ab einer Gruppengröße von mindestens fünf Neuzugängen werden aus der Gruppe zwei ausgewählt, um eine schnellere Durchmischung der qualifizierten Nutzer im Ranking zu ermöglichen.

Sollte die Anzahl qualifizierte Nutzer kleiner als fünf sein, muss mit dieser Gruppe gearbeitet werden. Auch wenn es dann vorhersagbar ist, wer teilnehmen wird, ist es dann allerdings wichtiger, überhaupt Feedback zu erhalten. In der Auswertung der Bewertung entspricht es dem Fall, wenn nicht alle Nutzer ihr Feedback abgegeben haben. Wie dann die Handhabung in der Auswertung erfolgt, wird im Kapitel Auswertung geschildert werden. Da wird außerdem beschrieben, wie die Bewertungen aggregiert werden und in welcher Form eine Ausreißerkontrolle erfolgt.

Aus den bisher geschilderten Überlegungen ergibt sich folgendes Verfahren. Jedem qualifizierten Nutzer wird eine Wahrscheinlichkeit zugeordnet, mit welcher er für die nächste Bewertungsrunde ausgewählt wird. Diese Wahrscheinlichkeit ist abhängig von

1. Der Platzierung im Ranking – je besser das Ranking, desto höher die Wahrscheinlichkeit. (Eine genaue Betrachtung davon, findet sich im Kapitel Rankingsystem.)
2. Der Anzahl der Bewertungsprozesse, die ohne Auswahl des Nutzers stattgefunden hat – je höher diese Anzahl ist, desto höher die Wahrscheinlichkeit.
3. Neue qualifizierte Nutzer haben einen eigenen Pool, aus denen mindestens einer der Bewertungsrunde hinzugefügt wird. Ihre Wahrscheinlichkeit ist also abhängig von der Poolgröße. Ist diese größer als fünf, werden zwei Nutzer der Bewertungsrunde hinzugefügt. Die Gesamtanzahl der Teilnehmer in einem Bewertungsprozess bleibt jedoch fünf.

### 4.2.3 Bewertung

Die Bewertung erfolgt innerhalb von verschiedenen vorgegebenen Kriterien an einer Punkteskala von eins bis zehn. Dabei gilt, dass je höher die Punktzahl ist, desto besser wurde das Kriterium erfüllt. Im Folgenden sollen die einzelnen Bewertungskriterien kurz vorgestellt werden.

**Strukturiert** Im ersten Kriterium soll der Aufbau des Kurses bewertet werden. Die Struktur von Kurs, Lessons und Topics soll ersichtlich, aufeinander aufbauend und nachvollziehbar sein. Die Überschriften passen zu den Inhalten und Topics und Lessons haben einleitende und oder abschließende Sätze, die bei der Orientierung im Kurs helfen. Zusätzlich werden zu Beginn des Kurses und an weiteren passenden Stellen Lernziele genannt. In diese Kategorie fällt außerdem, ob die vergebenen Schlagwörter korrekt sind.

**Vollständig** Das zweite Kriterium gibt an, ob alle wichtigen Aspekte des Themas ausreichend betrachtet wurden. Das bedeutet, dass auch Unterthemen behandelt wurden, Alternativen zumindest benannt werden und Themen bestenfalls von verschiedenen Seiten betrachtet werden. Die Bewertung der Vollständigkeit hängt dabei auch von den genannten Lernzielen ab. Sie dienen als Anhaltspunkte, wie weit und wie tiefgreifend ein Thema behandelt werden wird. Andernfalls wäre es schwierig, die Vollständigkeit eines Anfängerkurses mit der eines Expertenkurses zu vergleichen, da im Expertenkurs wesentlich mehr Details nötig sind, um als vollständig zu gelten. An dieser Stelle wird also indirekt auch bewertet, dass die Kursbeschreibung und der Inhalt stimmig sind.

**Verständlich** Für die Verständlichkeit des Kurses soll es ein Extrakriterium geben, auch wenn es in die vorherige Kategorie eingebunden werden könnte. Denn nur wenn der Inhalt auch verstanden werden kann, bringt der Kurs den Nutzern einen Mehrwert. Um diesen wichtigen Punkt auch in der Bewertung ausreichend relevant zu betrachten, soll er unabhängig von anderen Faktoren bewertet werden. In der Kategorie geht es also darum, dass die Inhalte verständlich und schlüssig sind. Das kann durch das Einbringen von Beispielen oder auch Abbildungen erfolgen, zeigt sich aber auch darin, dass eventuell unbekannte Begriffe erklärt werden. Entscheidend ist, dass man durch den Kurs ein Thema (besser) verstehen kann. Dafür kann auch die Einordnung in Anfänger / Fortgeschrittene / Experte ein Anhaltspunkt sein.

**Relevant** In dieser Kategorie soll die Aktualität und Abgrenzung des Kurses bewertet werden. Der Inhalt des Kurses soll auf dem aktuellen Wissenstand sein und relevant sein. Dabei muss nicht das Thema selbst top-aktuell sein, sondern es können auch beispielsweise Grundlagen auf neuem Wissenstand gut bewertet werden. Hierbei sollte dann aber auf die Abgrenzung zu anderen Kursen geachtet werden. Denn es soll vermieden werden, dass sehr viele ähnliche Kurse auf der Plattform angeboten werden. Um die Ähnlichkeit zu anderen Kursen zu bewerten, muss eine gute Kenntnis über das Angebot der Plattform gegeben sein. Das kann für Nutzer eine Herausforderung sein. Eine Unterstützung könnte dabei eine kurze Übersicht über die bestehenden Kurse sein, sodass ein Überblick

gewonnen werden kann, welche Themen schon auf der Plattform behandelt wurden. Schwerpunkt in der Kategorie soll aber die Prüfung auf Aktualität des Inhalts sein.

**Interessant** Damit ein Lerninhalt interessant wird, sollte er abwechslungsreich gestaltet werden. Es sollen also nicht nur Textabschnitte aneinandergereiht sein, sondern der Inhalt ansprechend vermittelt werden. Dafür können zusätzliche Medien wie Bilder, GIFs und Videos in die Inhalte eingebunden werden. Aber auch die Darstellung von Texten und Informationen kann variiert werden (ausklappbar, Listen, Tabellen). Die Kategorie ist allerdings abzugrenzen vom nächsten Kriterium.

**Interaktiv** In diesem Kriterium geht es an sich auch noch darum, dass Inhalte für den Nutzer interessanter werden, allerdings geht es hier ausschließlich um interaktive Elemente. Bewertet wird also, inwieweit der Nutzer angeregt wird, selbst aktiv zu werden. Dazu zählen zum Beispiel das Einbinden von Quizzen, Aufgaben (Assignments) und Tools. Aber auch wenn der Kurs Anleitungen für (plattform-)externe Aktionen enthält, bspw. eine Schritt-für-Schritt Anleitung für eine Programminstallation und dessen Einrichtung, kann dies positiv in die Bewertung dieser Kategorie einfließen.

**Wertvoll** In diesem letzten Kriterium soll das Preis-Leistungs-Verhältnis des Kurses betrachtet werden. Dabei spielt sowohl die Einzigartigkeit des Inhalts als auch die Höhe des Preises eine Rolle. Wird beispielsweise Thema behandelt, zu dem es allgemein noch nicht viele (gut aufbereiteten) Inhalte im Internet gibt, so wirkt dies positiv auf die Wertigkeit des Kurses und ein höherer Preis kann angemessen sein. Eine schlechte Bewertung in diesem Kriterium kann entstehen, wenn triviale Inhalte zu einem hohen Preis angeboten werden. Zu beachten ist bei dieser Kategorie die Möglichkeit, dass Nutzer immer den Preis als zu hoch bewerten, um die Preise niedrig zu halten. Sollte also immer wieder auffallen, dass die Bewertung in dieser Kategorie zwischen Nutzer und Experte weit auseinandergehen, dann sollte diese Kategorie ggf. angepasst werden. Diese Anpassung kann sein, dass die Nutzer dies zwar bewerten können, es allerdings nicht auf die Gesamtbewertung einwirkt.

Mit diesen sieben Kategorie sollen die wichtigsten Bewertungspunkte für einen Kurs abgedeckt werden, ohne dass der Bewertungsprozess zu umfangreich wird. Eine zu detaillierte Bewertung kann Nutzer abschrecken und kann zu Abbrüchen während der Bewertung und dadurch unvollständigen Datensätzen führen. Eine zu grobe Einteilung der Kategorien kann den Bewertungsprozess wiederum unverständlich für die Nutzer machen. Wichtig ist in jedem Fall eine klare Definition der Kategorien und ihren Bewertungspunkten für die Beteiligten des Bewertungsprozesses.

Zusätzlich zu der Vergabe der Punkte in den jeweiligen Kategorien, soll dem Nutzer auch ein Möglichkeit für individuelles Feedback gegeben werden. Dies kann sowohl den Experten als auch dem Inhaltersteller dabei helfen, die abgegebene Punktbewertung einschätzen zu können. Zusätzlich kann der bewertende Nutzer spezielle Probleme des Kursinhalts nennen und so differenzierte seine Meinung zu dem Kurs abgeben. Für die

Aggregierung der Bewertung werden allerdings nur die Punkte verwendet. Das freie Feedback dient nur als Hilfestellung für Experten und Ersteller. Anschließend an die Bewertung durch die qualifizierten Nutzer folgt der Bewertungsprozess durch die Experten.

## 4.3 Bewertungsprozess Experte

Experten sind Menschen aus dem Umfeld der Lernplattform. Sie sind den Betreibern der Plattform bekannt, ebenso wie ihre Bereiche, in denen sie Expertise haben. Ihnen kann bis zu einem gewissen Grad vertraut werden. Sie haben kein eigenes Interesse daran, der Plattform Schaden hinzuzufügen, sind aber auch nicht komplett immun gegenüber Bestechungen.

Die Bewertung durch die Experten erfolgt erst nach der Abgabe der Bewertung durch die Nutzer. Dies hat, auch wenn es den Prozess verlängert, verschiedene Vorteile:

1. Zum einen kann der Experte so das Feedback der Nutzer mit für seine Auswertung verwenden. Das bedeutet nicht, dass er diese übernimmt, sondern dass er die Hinweise mitverwenden kann, um sie auch aus seiner Sicht bewerten zu können.
2. Auf der anderen Seite kann der Experte die Hinweise des Nutzers auf ihre Qualität bewerten. Dies kann mit in die Bewertung des Nutzers einfließen.

Durch die versetzte Zeit kann also der Experte sowohl das Feedback der Nutzer verwenden als auch diese wiederum bewerten.

Im Gegensatz zu den Teilnehmern gibt es bei den Experten keine Rangliste. Stattdessen werden den Experten Fachgebiete zugeordnet, zu denen sie Expertise besitzen. Durch die Bewertung der Inhalte durch Fachexperten soll vor allem die inhaltliche Korrektheit sichergestellt werden. Dennoch ist es auch sinnvoll, fachfremde Experten in die Bewertung einzubeziehen. Diese können vor allem unter dem Gesichtspunkt der Verständlichkeit unvoreingenommener bewerten.

Wichtig ist, dass die Wahl der Experten nicht (komplett) voraussagbar für den Inhalterstellen ist und auch nicht beeinflussbar, bspw. durch den Zeitpunkt der Veröffentlichung. Andernfalls könnte das Bestechung durch den Inhaltersteller vereinfachen. Bei der Auswahl der Experten sollte also bestenfalls eine Zufallskomponente mit Einfluss haben. Als ein weiteres Kriterium sollte die Auslastung der Experten mit einbezogen werden, die Auswahl sollte also möglich gleichverteilt stattfinden und nicht immer die gleichen Experten gewählt werden. Das kann bei den Fachexperten nicht immer gegeben sein, aber sollte dennoch beachtet werden.

Die drei Kriterien sind also Fachexperten, Zufallskomponente und Abwechslung. Die Zufallskomponente kann am besten bei der Wahl aus der Gruppe der Experten ohne Fachexpertise zum Einsatz kommen, da diese Gruppe vermutlich größer ist. Die

Zufallskomponente selbst kann bereits für Abwechslung sorgen, zusätzlich könnte auch eine Art Abklingzeit eingesetzt werden, bis ein Experte erneut ausgewählt wird.

Eine Lösung ist, dass zuerst aus der Gruppe der zum Thema passenden Experten ein bis zwei Experten ausgewählt werden. Dabei werden diejenigen ausgewählt, deren letzte Beteiligung am längsten zurückliegt. Für eine fachlich bessere Bewertung sollten zwei Experten aus dieser Gruppe gewählt werden. Ist die Gruppe der Fachexperten dafür zu klein, dann sollte nur ein Fachexperte gewählt werden. Um den Fachexperten mindestens einen Bewertungsprozess Pause zu ermöglichen, müssen bei einer Wahl von zwei Experten mindestens vier Fachexperten in der Gruppe sein.

Damit zusätzlich auch eine Abwechslung innerhalb der Expertenpaare stattfindet, sollte bis zu einer Gruppengröße von fünf nur ein einzelner Fachexperte gewählt werden, statt zwei. Die ein oder zwei fehlenden Experten werden per Zufall aus den übrigen Experten gewählt, welche nicht am letzten Bewertungsprozess beteiligt waren. So wird eine feste Abklingzeit eingebaut, allerdings die Gruppe, aus der zufällig gewählt wird, nicht unnötig verkleinert. Bei einer größeren Einschränkung, wenn beispielsweise zwei Bewertungsprozesse pausiert werden müssen, steigt die Möglichkeit, dass die Auswahl der Experten vorhersagbar ist.

Eine weitere Herausforderung ist, dass Schlagwörter durch den Inhaltersteller falsch angegeben wurden. Dem wird zwar durch eine schlechtere Bewertung in der Kategorie Vollständigkeit entgegengewirkt, kann aber nicht vermieden werden. Zumal der Anreiz für den Inhaltersteller ist, dass er dadurch die Wahrscheinlichkeit für bestimmte Experten beeinflussen kann. Sollten die gewählten Schlagwörter nicht zum Inhalt passen und daher Fachexperten mit der falschen Expertise gewählt werden, gibt es zwei Optionen.

1. Die Wahl der Experten wird wiederholt mit korrekten Schlagwörtern.
2. Der Bewertungsprozess wird mit den gewählten Experten dennoch beendet, allerdings werden im Nachhinein die Schlagwörter korrigiert.

Beide Optionen haben Vor- und Nachteile. Der Vorteil bei der ersten Option wäre, dass in der Expertengruppe tatsächlich auch Experten für das Thema des Kurses sind, während bei der zweiten Option das nicht zwingend der Fall sein muss. Nachteil bei der ersten Option hingegen ist, dass sich der Prozess verlängert, da erst die korrekten Schlagwörter identifiziert werden müssen und anschließend neu Experten gewählt werden müssen. Während bei der zweiten Option ggf. auch darauf vertraut werden könnte, dass die Experten genügend Verständnis im Allgemeinen haben, um auch einen fachfremden Kurs zu bewerten. Wichtig ist, dass dennoch im Anschluss die Schlagwörter korrigiert werden, um eine korrekte Einordnung innerhalb der Plattform zu haben. Da auch eine leichtere Implementierung für die zweite Option spricht, soll diese in diesem Konzept gewählt werden.

Um möglichst wenig Ausfälle bei den ausgewählten Experten zu haben, sollten sie die Möglichkeit haben, sich vorher für gewisse Zeiträume abmelden zu können. Abgemeldete Experten, können nicht ausgewählt werden. Auf diese Weise soll die Wahrscheinlichkeit verringert werden, dass Experten ausgewählt werden, welche dann keine Bewertung abgeben, da sie verhindert sind.

Nach der Wahl der Experten wird für diese der Inhalt des neuen Kurses freigeschaltet und sie können den Kurs bewerten. Zusätzlich erhalten sie auch Zugriff auf die Bewertungen der Nutzer inclusive deren Hinweise. Die eigentliche Abgabe der Bewertung unterscheidet sich wenig vom Vorgang bei den qualifizierten Nutzern. Auch die Experten geben eine Punktzahl auf einer Skala von eins bis zehn für die vorgestellten Bewertungskriterien ab. Die Experten können außerdem Feedback an den Inhaltersteller formulieren, um diesem zusätzlichen Input zu Verbesserungsmöglichkeiten zu geben.

Sind beide Bewertungsprozesse beendet, sollen die Ergebnisse der Bewertung durch qualifizierte Nutzer und die der Experten aggregiert werden. Dies geschieht während des Auswertungsprozesses.

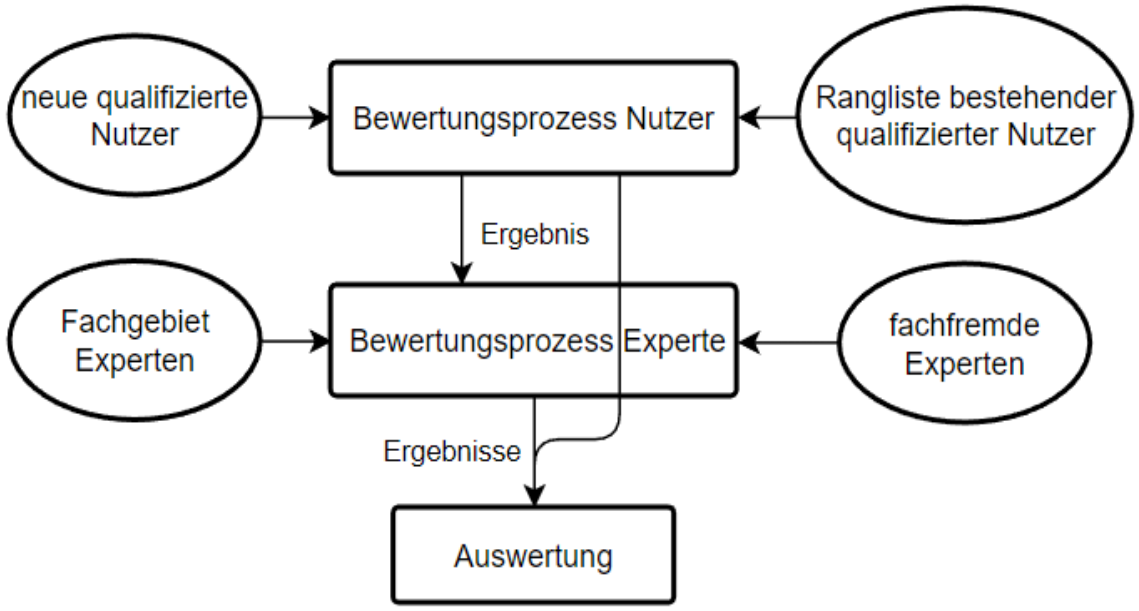



**Abbildung 3: Teilnehmer am Bewertungsprozess**

## 4.4 Auswertung

Für das Bestimmen einer passenden Methode für die Auswertung und Aggregation der Bewertungen werden verschiedene statistische Methoden betrachtet und gegeneinander abgewogen. Zusätzlich werden eigene Ansätze entwickelt und mithilfe von Beispielen eine passende Methode gefunden. Zuletzt werden mögliche Herausforderungen identifiziert und Lösungen dafür betrachtet.

### 4.4.1 Statistische Methoden

Die Ausgangslage bei einer Abgabe von Bewertungen ohne Ausfällen ist, dass fünf Bewertungen pro Bewertungskriterium durch Nutzer vorliegen und jeweils drei von Experten. Die Bewertungen sind ordinale Daten auf einer Skala von eins bis zehn. Und die Auswertung soll pro Bewertungskriterium neu stattfinden, sodass die Stichprobenmenge insgesamt $n = 8$ ist und aufgeteilt auf die Gruppen Experten $n_E = 3$ und Teilnehmer $n_T = 5$.

Ziel bei der Auswertung der Daten ist es, die abgegebenen Bewertungen in einem Wert zusammenzufassen. Dieser Wert soll die Meinung von Nutzern und Experten widerspiegeln. Dafür kommen grundlegend verschiedene Lagemaße infrage – zum einen das arithmetische Mittel, zum anderen der Median. Der Median als Alternative zum arithmetischen Mittel kommt in Betrachtung, da die auszuwertenden Daten Bewertungen enthalten können, welche weit vom Rest der Daten entfernt sind. Diese Ausreißer aus der Reihe der Werte können die Qualität von statistischen Auswertungen beeinträchtigen. (Hornsteiner 2012) Sie entstehen durch Nutzer, welche entweder keine passende Bewertung abgeben können, da das Thema ihr Wissen übersteigt oder aktiv mit unpassenden Bewertungen dem System schaden wollen.

Für den Umgang mit Ausreißern gibt es verschiedene Vorgehensweisen in der Statistik, dazu zählen u.a. die Wahl von robuste Lagemaße und Ausreißerverfahren. (Hedderich und Sachs 2012) Zu den robusten Lagemaßen zählt zum einen der Median oder ein angepasstes arithmetisches Mittel. Nachteil bei der Wahl auf den Median als Lagemaß ist, dass hier nur ein Wert in das Maß eingeht, während beim arithmetischen Mittel alle Daten mit Einwirken. Ein besserer Ansatz ist es daher, mit einem Ausreißerverfahren die Ausreißer zu erkennen und sie von der Berechnung des Durchschnitts auszuschließen. (Hornsteiner 2012) Wobei dies mit einem gewissen Risiko einhergeht, da eine exakte statistische Entscheidung, ob ein Messwert als Ausreißer gilt, schwierig ist und bei einer Fehleinschätzung Daten verloren gehen oder das Ergebnis verzerrt wird. (Hedderich und Sachs 2012)

Bei den Ausreißerverfahren gibt es sowohl komplexe Vorgehensweisen, bei welchen mehrere Berechnungsschritte notwendig sind, als auch Verfahren, wo nur wenige Schritte zum Ziel führen sollen. Ein sehr einfaches Verfahren ist das Trimmen. Dabei werden die Werte, die am äußersten Rand liegen, also der kleinste und der größte Wert, gestrichen. Der Nachteil dieses einfache Verfahrens ist, dass immer zwei Werte verloren gehen, ohne

zu prüfen, ob sie tatsächlich vom Rest der Daten weit entfernt liegen und eine statistische Auswertung beeinträchtigen. (Sachs 1992) Das ist besonders bei der vorliegenden Stichprobengröße sehr bedenklich. Da Ausreißerwerte nur bei den Bewertungen durch den Nutzer vermutet werden, ist die Stichprobenmenge fünf. Durch das Trimmen wäre sie nur noch drei und würde dadurch die Aussagekraft der Bewertungen stark schmälern.

Ein Kennwert zur Entscheidung über Ausreißer mit mehr Berechnungsschritten ist der Interquartilsabstand (IQR). Quartile sind die Werte, welche eine geordnete Reihe von Datenwerten in Viertel einteilt. Der Interquartilsabstand ist der Abstand zwischen dem dritten und dem ersten Quartil und schließt damit die mittleren 50 Prozent der Beobachtungswerte ein. Dieser Wert kann als Ersatz für die Standardabweichung genutzt werden, wenn ein robuster Wert benötigt wird. (Hornsteiner 2012) Zusätzlich kann der Wert genutzt werden, um Ausreißer zu erkennen. Werte, die kleiner sind als das erste Quartil abzüglich des IQR multipliziert mit einem Faktor k, sind Ausreißer nach unten. Ausreißer nach oben sind Werte, welche größer sind als das dritte Quartil plus Faktor k multipliziert mit dem IQR. Die Größe des Faktors k kann zwischen 1,5 und 3 gewählt werden, je nach Art der Studie. (Hedderich und Sachs 2012)

$$Q_1 - \mathrm{IQR} * k \leq \text{kein Ausreißer} \leq Q_3 + \mathrm{IQR} * k$$

Der Nachteil dieses Verfahrens ist, dass dafür eine Stichprobenmenge von mindestens zehn sinnvoll ist. (Hedderich und Sachs 2012) Das ist bei der Betrachtung der Nutzerbewertungen nicht gegeben, stattdessen ist sie nur halb so groß. Zusätzlich könnte auch folgende Betrachtung ein Problem darstellen: Für die Bewertung der Ausreißer mit dem Interquartil sind mehrere Schritte notwendig – Ordnen der Werte, Berechnung der Quartile 1 und 3, Berechnung des Interquartilsabstands und schließlich die Berechnung der robusten Grenzen, welche die Ausreißer ausschließen. Für klassische Computeranwendungen sollte das keine Herausforderung darstellen. Soll die Auswertung allerdings auf der Blockchain ausgeführt werden, so verursacht jeder Berechnungsschritt Kosten und es sollte gut abgewogen werden, ob es eine effizientere Möglichkeit gibt.

### 4.4.2 Entwicklung eigener Verfahren

Zusätzlich wird bei diesen Verfahren aus der Statistik ein Vorteil, der in diesem Szenario gegeben ist, außen vorgelassen – die Bewertung der Experten. Durch das Einbeziehen der Expertenbewertungen wird außerdem der Herausforderung der sehr kleinen Stichprobenmenge von fünf entgegengewirkt. Die Bewertungen der Nutzer können nicht nur innerhalb ihrer Stichproben betrachtet werden, sondern auch mit der Bewertung der Experten abgeglichen werden. Dabei gilt Annahme, dass es nur bei den Nutzern zu Ausreißern kommt, da den Experten sowohl das Wissen als auch der Wille zugesprochen wird, eine korrekte Bewertung abzugeben.

In einem ersten eigenen Verfahren, um Ausreißer festzustellen, soll der Durchschnitt der Expertenbewertung als Referenzwert betrachtet werden. Von ihm ausgehend werden Grenzen definiert, in denen sich die Bewertungen der Nutzer befinden können. Werte außerhalb der Grenzen werden als Ausreißer gekennzeichnet. Wie weit diese Grenze gesetzt wird, ist davon abhängig, wie sehr man die Bewertung der Nutzer an die Bewertung der Experten anpassen will. Eine kleine Grenze schließt schnell Nutzerbewertungen aus, welche zwar von der Expertenmeinung abweichen, aber durchaus valide Wertungen sein können. Schließlich ist auch davon auszugehen, dass Nutzer beispielsweise die Verständlichkeit in einem Kurs schlechter bewerten als Experten, die bereits mit dem Thema vertrauter sind. Daher sollte die Grenze groß genug gewählt werden, um möglichst nur Ausreißer auszuschließen, die auf einer Fehleinschätzung beruhen. Als Abstand wurde daher 3,5 gewählt, sodass um den Durchschnitt der Experten eine Spanne von sieben Punkten als akzeptabel für die Nutzerbewertung festgelegt wird. Damit ist bereits ein Großteil der Skala abgedeckt und es werden wirklich nur sehr schlechte oder sehr gute Bewertungen, die stark von der Expertenmeinung abweichen betrachtet.

Eine Weiterentwicklung dieses ersten eigenen Verfahrens ist, dass auch die Streuung der Expertenbewertung mit beachtet wird. Das bedeutet, dass, wenn sich die Experten uneinig sind und es da zu einer höheren Streuung kommt, auch eine größere Streuung bei den Nutzern erlaubt wird. Um auf die Berechnung von statistischen Streuungsmaßen verzichten zu können, könnte auch der Ansatz des IQR angepasst werden. Vereinfacht man diesen, kann man die Grenzen für akzeptierte Werte der Nutzer ausgehend von der niedrigsten und höchsten Expertenbewertung setzen. Anstatt also vom Durchschnitt der Bewertung ausgehend einen Abstand für die Grenzen festzulegen, werden der kleinste Wert der Expertenbewertung und der höchste Wert als Ausgangspunkte genutzt. Von da aus kann dann ein Abstand von 2,5 als sinnvolle Grenze erachtet werden. Der Abstand kann kleiner als beim Durchschnitt gewählt werden, da der Abstand zwischen höchster und niedrigster Expertenbewertung zur Spanne dazu kommt. Der Vorteil dieses Ansatzes ist, dass die Spanne für Nutzerbewertungen von der Streuung der Expertenbewertungen abhängig angepasst wird und weniger starr als im ersten Verfahren ist. Nachteil ist, dass einzelne Experten mehr Macht haben und dadurch auch anfälliger werden können für Absprachen und Bestechungen zwischen Nutzer und Experten. Außerdem werden bei diesem Verfahren wieder mehr Berechnungsschritte benötigt.

### 4.4.3 Bewertung der Verfahren

Die vorgestellten und selbst entwickelten Verfahren sollen nun an drei Beispielen berechnet und die Ergebnisse verglichen werden. Zu den Verfahren zählen das Trimmen, das Berechnen robuster Grenzen mithilfe des IQR mit dem Faktor $k = 1{,}5$ und die beiden eigenen Verfahren mit den genannten Abständen. Der Gesamtdurchschnitt wird immer über die Expertenwerte und die Nutzerbewertungen, die nach der Ausreißerbewertung bestehen bleiben, berechnet. Dabei werden jeweils die Durchschnitte betrachtet und darüber das arithmetische Mittel gebildet, sodass die beiden Gruppen Experten und Nutzer gleich

viel Anteil am Gesamtdurchschnitt haben. Für das erste Beispiel werden die Berechnungsschritte kurz erläutert, auch um eine Übersicht für die nötigen Berechnungsschritte zu bekommen.

Die folgende Tabelle zeigt in den Spalten zwei und drei die abgegebenen Bewertungen durch die drei Experten und die fünf Nutzer. In den folgenden Spalten zeigt sich, welche Auswirkung auf die Daten der Nutzerbewertung das jeweilige Verfahren hat. Wird ein Wert als Ausreißer von dem Verfahren erkannt, wird der Wert in dieser Spalte durch ein „A“ ersetzt. In den letzten beiden Zeilen wird zunächst der Durchschnitt der Spaltenwerte angegeben und beim Gesamtdurchschnitt mit dem Durchschnitt der Experten verrechnet.

**Tabelle 1: Auswertung Beispiel 1**

| Verfahren | Experten | Nutzer | Trimmen | IQR k=1,5 | Eigenes 1 | Eigenes 2 |
|---|---|---|---|---|---|---|
| Bewertungen | 6 | 8 | A | 8 | 8 | 8 |
| | 8 | 7 | 7 | 7 | 7 | 7 |
| | 6 | 2 | A | 2 | A | A |
| | | 5 | 5 | 5 | 5 | 5 |
| | | 8 | 8 | 8 | 8 | 8 |
| | | | | | | |
| Durchschnitt | 6,67 | 6,00 | 6,67 | 6,00 | 7,00 | 7,00 |
| Gesamtdurchschnitt | | 6,33 | 6,67 | 6,33 | 6,83 | 6,83 |

Beim Betrachten der Nutzerwerte sticht vor allem die Zwei (Zeile 3) sowohl im Vergleich zum Rest der Nutzerbewertungen als auch im Vergleich zur Expertenbewertung hervor. Beim Trimmen wird einmal der höchste Wert 8 und der niedrigste Wert 2 entfernt. Auch wenn die 8 zweimal vorkommt, so wird sie in der Aggregierung nur einmal entfernt. Allerdings sollte später bei der Bewertung der Nutzer beide Nutzer, welche einen Ausreißerwert haben, bestraft werden. Durch diesen Schritt hebt sich der Durchschnitt der Nutzerbewertung von 6 auf 6,67. An diesem Beispiel zeigt sich bereits, dass Trimmen nicht dazu geeignet ist, Ausreißer verlässlich zu erkennen. Denn die eliminierte acht kommt als Wert ein weiteres Mal in der Wertereihe vor, sowie ihr direkter Nachbar sieben. Sie entspricht damit nicht der Definition eines Ausreißers, sondern gehört zu den auszuwertenden Daten dazu. Durch das Trimmen ist diese Information verloren gegangen.

Das Ausreißerverfahren basierend auf den Interquartilsabstand (IQR) hat keinen der Werte als Ausreißer erkannt, damit ändert sich auch der Gesamtdurchschnitt nicht. Für die Berechnung für die Grenzen mit dem Interquartilsabstand mussten zunächst die beiden Quartile 1 und 3 kalkuliert werden. Dafür müssen die Werte geordnet werden:

**Tabelle 2: geordnete Werte**

| Stelle | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|
| X | 2 | 5 | 7 | 8 | 8 |

Anschließend werden die Quartile mit folgender Formel berechnet:

$$Q_p = \begin{cases} \frac{x_{n*p} + x_{n*p+1}}{2}, \text{wenn } n*p \in \mathbb{Z} \\ x_{\lceil n*p \rceil}, \quad \text{wenn } n*p \notin \mathbb{Z} \end{cases}$$

Für die Variable p entspricht für das Quartil Drei 0,75 und für Quartil Eins 0,25. Für beide Fälle gilt bei n=5 der untere Teil der Formel. Daraus ergeben sich folgende Werte:

$$Q_3 = x_{\lceil 5*0{,}75 \rceil} = x_{\lceil 3{,}75 \rceil} = x_4 = 8$$

$$Q_1 = x_{\lceil 5*0{,}25 \rceil} = x_{\lceil 1{,}25 \rceil} = x_2 = 5$$

Der Interquartilsabstand (IQR) ergibt sich aus der Differenz von Quartil Drei und Quartil Eins und ist in diesem Fall 3. Die Berechnung der Grenzen ergibt sich wie folgt:

$$\text{obere Grenze} = Q_3 + 1{,}5 * \text{IQR} = 8 + 1{,}5 * 3 = 12{,}5$$

$$\text{untere Grenze} = Q_1 - 1{,}5 * \text{IQR} = 5 - 1{,}5 * 3 = 0{,}5$$

Nach dieser Berechnung fällt auf, dass die Grenzen außerhalb der vorgegebenen Skala von 1 – 10 liegen. Entsprechend erklärt sich auch, warum mit diesem Verfahren keine Ausreißer erkannt wurden. Wie bereits bei der Erklärung des Verfahrens angemerkt wurde, müsste für dieses Verfahren die Stichprobemenge größer sein, damit gute Aussagen damit getroffen werden können.

Die beiden eigenen Verfahren schließen den Wert zwei als Ausreißer aus der Berechnung aus und entsprechend hebt sich der Gesamtdurchschnitt. Auch für diese beiden Verfahren werden folgend die Berechnungsschritte dargestellt.

Für das erste Verfahren wird zu dem bereits berechneten Durchschnitt der Expertenbewertung ($\emptyset_E$) 3,5 addiert bzw. subtrahiert, um die obere und untere Grenze zu definierten.

$$\text{obere Grenze} = \emptyset_E + 3{,}5 = 6{,}67 + 3{,}5 = 10{,}17$$

$$\text{untere Grenze} = \emptyset_E - 3{,}5 = 6{,}67 - 3{,}5 = 3{,}17$$

Beim zweiten Verfahren werden die Grenzen ausgehend vom Maximal- und Minimalwert der Expertenbewertung berechnet, indem 2,5 addiert bzw. subtrahiert wird.

$$\text{obere Grenze} = 8 + 2{,}5 = 10{,}5$$

$$\text{untere Grenze} = 6 - 2{,}5 = 3{,}5$$

Bei beiden Verfahren liegt die obere Grenze außerhalb der Skala und die untere Grenze schließt den Wert 2 aus der Nutzerbewertung aus, indem sie ihn als Ausreißer definiert.

Zusammengefasst zeigt das Beispiel,

1. dass Trimmen ein zwar simples, aber ggf. schädliches Verfahren für die weitere Auswertung ist.
2. dass die Interquartilsberechnung am meisten Berechnungen benötigen, ohne dass ein Ausreißer in diesem Beispiel erkannt wurde
3. dass die beiden eigenen Verfahren in diesem Beispiel zum gleichen Ergebnis kommen, welches am ehesten der ersten Einschätzung entspricht, nämlich dass der Wert 2 aus der Wertereihe sticht.

Im nächsten Beispiel wird auf die genaue Vorgehensweise der Berechnungen verzichtet, stattdessen werden nur die Ergebnisse und die oberen und unteren Grenzen angegeben. Dies soll als Grundlage für ein weiteres Vergleichen und Abwägen der Verfahren dienen.

**Tabelle 3: Auswertung Beispiel 2**

| Verfahren | Experten | Nutzer | Trimmen | IQR k=1,5 | Eigenes 1 | Eigenes 2 |
|---|---|---|---|---|---|---|
| Bewertungen | 4 | 8 | A | A | A | 8 |
| | 3 | 5 | 5 | 5 | 5 | 5 |
| | 6 | 2 | A | A | 2 | 2 |
| | | 5 | 5 | 5 | 5 | 5 |
| | | 4 | 4 | 4 | 4 | 4 |
| | | | | | | |
| Durchschnitt | 4,33 | 4,80 | 4,67 | 4,67 | 4,00 | 4,80 |
| Gesamtdurchschnitt | | 4,57 | 4,50 | 4,5 | 4,17 | 4,57 |

Bei diesem zweiten Beispiel fällt auf den ersten Blick die Nutzerbewertung mit dem Wert 8 auf, welche am ehesten noch in der Nähe der Expertenbewertung mit dem Wert 6 ist. Sie ist dennoch weit entfernt sowohl vom Durchschnitt der Experten als auch dem von Nutzern. Beim Verfahren des Trimmens werden sowohl die 8 als auch die 2 entfernt, was in diesem Beispiel passender ist als beim ersten Beispiel. Für die weiteren Verfahren wurden folgende Grenzen berechnet:

1. IQR: untere Grenze = 2,5; obere Grenze = 6,5
2. Eigenes Verfahren 1: untere Grenze = 0,83; obere Grenze = 7,83
3. Eigenes Verfahren 2: untere Grenze = 0,50; obere Grenze = 8,50

Beim Betrachten der Grenzen fällt diesmal auf, dass bei den beiden eigenen Verfahren die untere Grenze außerhalb der Skala liegt, während bei Verfahren mit dem Interquartilsabstand beide Grenzen innerhalb der Skala liegen. Zusätzlich erkennt das Interquartilsverfahren die gleichen Werte als Ausreißer wie das Trimmen.

In diesem Beispiel zeigt sich die Änderung der Ausreißeranalyse beim Betrachten der Expertenbewertung als Vergleich anstelle der Nutzerbewertungen. Die beiden eigenen Verfahren beziehen sich auf die Expertenbewertung bei der Analyse der Nutzerbewertung. Beim Verfahren, welches sich auf den Durchschnitt der Experten bezieht, wird die Acht als Ausreißer erkannt. Die Zwei hingegen bleibt in der Wertereihe bestehen, da sie nah genug am Durchschnitt von 4,33 liegt. Das zweite Verfahren bezieht sich auf die Werte drei und sechs bei den Experten und da diese beiden recht weit auseinanderliegen, ergibt sich auch die große Spanne zwischen den Grenzen. Daher wird hier kein Ausreißerwert erkannt.

Das kann bereits ein Indiz auf die Herausforderung bei diesem Verfahren sein. Sobald in den Expertenbewertungen ein größerer Abstand als drei herrscht, ist die Spanne der Grenzen bei 8 und schließt damit einen Großteil der Skala ein – abhängig davon, wo der größere bzw. kleinere Wert liegt. Außerdem wird bei dem Verfahren die dritte Expertenmeinung nicht beachtet, während sie beim ersten Verfahren mit auf den Durchschnitt einwirkt. Hier spielt also eher die Tendenz der gesamten Expertenbewertung eine Rolle bei dem Setzen der Rolle und nicht zwei Einzelmeinungen. Das Ergebnis ist, dass ein einzelner Experte wesentlich mehr Macht bei dem zweiten Verfahren hat als beim durchschnittbasierten.

**Tabelle 4: Auswertung Beispiel 3**

| Verfahren | Experten | Nutzer | Trimmen | IQR k=1,5 | Eigenes 1 | Eigenes 2 |
|---|---|---|---|---|---|---|
| Bewertungen | 7 | 9 | A | 9 | A | 9 |
| | 5 | 6 | 6 | 6 | 6 | 6 |
| | 4 | 3 | A | 3 | 3 | 6 |
| | | 4 | 4 | 4 | 4 | 4 |
| | | 6 | 6 | 6 | 6 | 6 |
| | | | | | | |
| Durchschnitt | 5,33 | 5,60 | 5,33 | 5,60 | 4,75 | 5,60 |
| Gesamtdurchschnitt | | 5,47 | 5,33 | 5,47 | 5,04 | 5,47 |

Gleiches zeigt sich auch in diesem letzten Beispiel, wo die Sieben der Experten im zweiten Verfahren dazu führt, dass die Bewertung von Neun bei den Nutzern nicht als Ausreißer erkannt wird. Während die Gesamtbetrachtung der drei Expertenbewertungen einen Durchschnitt ergeben, durch den die Neun als Ausreißer erkannt wird. Der Gesamtdurchschnitt bei den beiden eigenen Verfahren zeigt, wie viel Unterschied die Neun in den Nutzerbewertungen macht. Eine Nutzerbewertung sorgt in diesem Fall für eine Verbesserung um einen halben Punkt.

Die Betrachtung der verschiedenen Beispiele und der einzelnen Berechnungsschritte hat Aufschluss über die Verfahren im Normalfall gegeben. Das heißt, wenn alle, die eine Bewertung abgeben sollen, auch eine abgegeben haben und diese verarbeitet werden kann. Da dies aber nicht immer der Fall sein muss, werden im nächsten Kapitel andere Fälle betrachtet und welche Auswirkungen das auf die Verfahren haben kann.

### 4.4.4 Mögliche Herausforderungen

Bisher wurde immer von vollständigen Datensätzen ausgegangen. Dies muss aber in der Realität nicht immer gegeben sein. Menschliche oder technische Fehler verschiedener Art können dazu führen, dass Werte in den Datensätzen fehlen. Auch in diesen Fällen muss eine Lösung gefunden werden.

Prinzipiell gibt es zwei Möglichkeiten, wenn Daten in der Bewertung fehlen. Zum einen kann der Bewertungsprozess abgebrochen und neu gestartet werden. Zum anderen kann eine Auswertung der vorliegenden Bewertungen (angepasst) durchgeführt werden. Nachteil bei einer Wiederholung ist, dass es den Reviewprozess verlängert und im schlimmsten Fall es immer wieder zu einer Wiederholung kommt, weil erneut Daten fehlen. Dies kann sogar als eine Angriffsmöglichkeit genutzt werden. Nachteile bei der Auswertung der unvollständigen Daten sind eine erschwerte Auswertung, eine Verschlechterung der Aussagekraft der Aggregierung und ein mögliches Verschieben von Macht. Letzteres ergibt sich dadurch, dass der Einfluss der abgegebenen Bewertungen steigt, je mehr Bewertungen fehlen. Letztendlich muss für einzelne Szenarien abgewogen werden, welches Vorgehen sich besser anbietet.

Es wird unterschieden danach, ob Nutzerbewertungen fehlen oder ob Expertenbewertungen fehlen. Fehlende Expertenbewertungen sind grundsätzlich ein größeres Problem, da zum einen schon nur drei Bewertungen geplant sind und zum anderen sie benötigt werden, um auch die Nutzerbewertung einschätzen zu können. Sollten alle Expertenmeinungen fehlen, so muss der Reviewprozess zumindest für die Experten neu gestartet werden. Die Bewertungen der Nutzer können bestehen bleiben, aber es muss eine Neuwahl der Experten getroffen werden, um eine Expertenbewertung zu erhalten. Andernfalls würde der Kontrollmechanismus für die Nutzerbewertung wegfallen und auch eine Einordnung der Nutzerbewertung wäre nicht möglich. Aus diesem Grund muss in diesem Fall der Reviewprozess verlängert werden, um ein aussagekräftiges Ergebnis zu erhalten.

In dem Fall, dass nur ein oder zwei Expertenmeinungen vorliegen, wird hingegen auf eine Wiederholung der Expertenauswahl verzichtet und stattdessen wird mit den vorliegenden Bewertungen fortgefahren werden. Auf diese Weise sollen zu viele Verzögerungen vermieden werden. Auch wenn die Nachteile einer verschlechterten Qualität in der Bewertung in Kauf genommen werden müssen.

Das Fehlen von Nutzerbewertungen ist zwar weniger problematisch als fehlende Expertenbewertungen, aber verschlechtert dennoch die Qualität der Bewertung, da weniger Meinungen aggregiert werden können. Die Möglichkeit, dass der Bewertungsprozess abgebrochen und mit einer neuen Auswahl von Nutzern wiederholt wird, wenn Bewertungen fehlen, birgt ein Angriffsszenario. Hierfür könnten sich Nutzer absprechen, keine Bewertung für einen Kurs abzugeben und so dessen Veröffentlichung verhindern, da der Prozess immer wieder wiederholt wird. Dies könnte sogar von anderen Inhalterstellern motiviert werden, welche ähnliche Kursinhalte veröffentlichen wollen. Aus diesem Grund soll, selbst wenn kein Nutzer eine Bewertung abgibt, der Bewertungsprozess mit dann ausschließlich Experten abgeschlossen werden. Dadurch ergibt sich zwar, dass die Experten mehr Macht haben in der Abstimmung, aber das ist weniger problematisch als das vorgestellte Angriffsszenario.

Zum Anpassen der Machtverhältnisse sollten bei verschiedenen Szenarien der Anteil am Gesamtdurchschnitt angepasst werden. Sollten beispielsweise vier Nutzerbewertungen fehlen, hätte die einzelne Nutzerbewertung einen Anteil von 50 % auf die Gesamtbewertung, was erneut ein Angriffsrisiko darstellt. Zusätzlich kann bei einige Szenarien auch die Ausreißerbewertung nicht mehr sinnvoll durchgeführt werden. Die einzelnen Szenarien und ihre Anpassungen werden in der folgenden Tabelle kurz dargestellt.

**Tabelle 5: Anpassungen bei verschiedenen Szenarien**

| Experten | Nutzer | Berechnung Gesamtdurchschnitt | Ausreißer |
|---|---|---|---|
| 3 / 2 | 4 | 50% Expertendurchschnitt + 50% Nutzerdurchschnitt | Ja |
| 3 / 2 | 3 | 60% Expertendurchschnitt + 40% Nutzerdurchschnitt | Keine |
| 3 / 2 | 2 | 70% Expertendurchschnitt + 30% Nutzerdurchschnitt | Keine |
| 3 / 2 | 1 | 80% Expertendurchschnitt + 20% Nutzerbewertung | Keine |
| 3 / 2 | 0 | Expertendurchschnitt | Keine |

| | | | |
|---|---|---|---|
| 1 | 5 / 4 | 40 % Expertendurchschnitt + 60% Nutzerdurchschnitt | Keine |
| 1 | 3 / 2 / 1 | 50% Expertendurchschnitt + 50% Nutzerdurchschnitt | Keine |
| 1 | 0 | Expertenbewertung | Keine |
| 0 | 0 – 5 | Prozess wiederholen | Keine |

In den Fällen, wo mindestens zwei Expertenbewertungen vorliegen, erhöht sich der Einfluss des Expertendurchschnitts mit fallender Anzahl an Nutzerbewertungen. Eine Ausreißerbewertung, findet nicht mehr statt, sobald weniger als vier Nutzerbewertungen vorliegen, um nicht noch weitere Daten zu verlieren. Ebenso wird darauf verzichtet, wenn nur noch eine Expertenbewertung vorliegt. Zusätzlich wird in diesem Fall der Anteil des Nutzerdurchschnitts bei der Berechnung des Gesamtdurchschnitts erhöht, wenn mindestens vier Nutzerbewertungen vorliegen. Wurde keine Expertenbewertung abgegeben, so wird der Prozess der Expertenbewertung wiederholt.

Mithilfe dieser Anpassungen soll das vorgestellte Konzept für die Ausreißerbewertung und Aggregierung der einzelnen Bewertungen auch bei fehlenden Wertem eingesetzt werden können. Damit wurden die wichtigsten Szenarien bei der Auswertung abgedeckt und zusätzlich Angriffsmöglichkeiten entgegengewirkt. Ein Teil, der auf die Aggregierung der Bewertungen aufbaut, ist die Auswertung der Güte der Nutzerbewertung und der Einfluss auf das Ranking und das Anreizsystem für Nutzer.

## 4.5 Konzepte für Nutzer

In diesem Kapitel sollen zwei Systeme für die Nutzer vorgestellt werden. Zum einen wird genauer auf das Rankingsystem eingegangen. Und anschließend wird ein mögliches Anreizsystem für die Abgabe von Bewertungen für die Nutzer vorgestellt.

### 4.5.1 Rankingsystem

Der Einsatz des Rankingsystems bringt verschiedene Vorteile mit sich. Zum einen ist es eine Motivation für qualifizierte Nutzer, immer wieder an einem Bewertungsprozess teilzunehmen und bei diesem auch korrekt zu bewerten. Eine korrekte Bewertung zeichnet sich durch Vollständigkeit und das Fehlen von Ausreißerwerten aus. Die Verknüpfung zwischen Ranking und der Wahrscheinlichkeit ausgewählt zu werden, ist allerdings nicht nur motivierend, sondern steigert auch die Qualität der Bewertungen, da vermehrt gute qualifizierte Nutzer ausgewählt werden. Außerdem kann das Ranking als Ausschlusskriterium

genutzt werden. Nutzer, welche aus dem Ranking rausfallen, kommen entsprechend auch nicht mehr als Bewerter in Betracht und verlieren ihren Status als qualifizierter Nutzer.

Das Ranking kann auf verschiedene Arten gestaltet werden. Zum einen kann eine Skala von 5 Sternen genutzt werden, nach denen die Nutzer gerankt werden. Dabei kann auch die Anzahl der abgegebenen Bewertungen mit einbezogen werden. Alternativ kann das Ranking auf einer Punktezahl aufbauen. Dieses Punktezahl könnte auch durch den Besitz von Token abgebildet werden, allerdings sprechen zwei Gründe dagegen. Zum einen wären die Token in ihrem Einsatz stark limitiert, da Nutzer sie nicht ausgeben wollen würden, um ihr Ranking nicht zu verlieren. Es müsste also ein neuer Token geschaffen werden.

Ein weiterer Grund, der dagegenspricht ist, dass ein Tokensystem eine Bestrafung erschwert. Für den Abzug von Token aufgrund eines Fehlverhaltens müsste der Nutzer zuvor einen gewissen Betrag hinterlegen - im Grunde in Höhe der höchsten Strafe. Das kann auf ehrliche Nutzer abschreckend wirken und erfordert auch in der Umsetzung wesentlich mehr Aufwand als ein Punktesystem. Bei einem Ranking basierend auf einer 5-Sterne-Skala kann es schnell unübersichtlich werden, wenn Nutzer im Ranking ähnliche Sternanzahlen haben. Zusätzlich kann das Erreichen von 5 Sternen auf der Skala einen Nutzer eher demotivieren, eine weitere Bewertung abzugeben, da er sich eh nicht weiter verbessern kann. Es wurde sich aufgrund der einfacheren Umsetzung und einer vermutlich besseren Motivation für ein Punktesystem entschieden, welches zentral über die Plattform verwaltet wird.

Für die Punktevergabe wurde folgendes Konzept entwickelt. Als Ausgangspunkt hat ein neuer qualifizierter Nutzer noch keine Punkte und erhält diese mit seiner ersten abgegebenen Bewertung. Dies ist möglich, da ein bis zwei Nutzer aus dieser neuen Gruppe für einen Bewertungsprozess ausgewählt werden, unabhängig ihrer Punktzahl. Um den Nutzern einen guten Start zu ermöglichen, werden hier für die erste abgegebene Bewertung mehr Punkte vergeben als bei den folgenden Bewertungen. Entsprechend werden für die erste korrekt abgegebene Bewertung zwanzig Punkte gutgeschrieben. Kleinere Fehler in der Bewertung werden nicht direkt bestraft, um den neuen qualifizierten Nutzer nicht zu sehr zu demotivieren, stattdessen erhält er nur zehn Punkte, solange vier Bewertungen aus den sieben Bewertungskriterien korrekt sind. Sollte ein neuer Nutzer allerdings zweimal keine Bewertung abgeben, so wird er aus der Gruppe ausgeschlossen und verliert seinen Status als qualifizierter Nutzer.

Für die weiteren Szenarien und ihre Auswirkungen auf das Ranking wurde versucht, eine Balance zu finden zwischen dem Bestrafen von Nutzern, welche systemschädliche Bewertungen abgeben und dem Motivieren von Nutzern, welche nicht vollständig korrekt bewertet haben, aber dennoch weiterhin als mögliche Bewerter erhalten bleiben sollen. Grundsätzlich gilt, dass qualifizierte Nutzer ausgeschlossen werden, sollte ihr Punktwert negativ werden.

Im ersten Szenario gibt ein qualifizierter Nutzer, der bereits seine erste Bewertung abgegeben hat, eine weitere Bewertung ab. Ist diese korrekt, also vollständig und frei von Ausreißern, erhält er dafür zehn Punkte für sein Ranking. Für den Fall, dass die Bewertung nicht vollständig korrekt ist, gibt es verschiedene Abstufung. Fehlt nur ein Wert oder sind maximal zwei der sieben abgegebenen Bewertungen Ausreißer, dann erhält er noch eine Belohnung von vier Punkten. Enthält die Bewertung maximal vier Ausreißer oder höchstens drei fehlende Werte, so ändert sich sein Scoring nicht. Enthält die Bewertung fünf oder mehr Ausreißer, werden dem Nutzer acht Punkte abgezogen, und wenn vier oder mehr Werte fehlen, werden ihm zehn Punkte abgezogen. Treten sowohl Ausreißer als auch fehlende Werte auf, so werden die Ausreißer als fehlende Werte gezählt und entsprechende Punkte vergeben oder abgezogen. So soll ein System erschaffen werden, wo an sich mehr belohnt als bestraft wird und es außerdem einen Toleranzbereich gibt, in welchem die Punkte des Nutzers nicht geändert werden. Das ist an sich auch bereits ein Nachteil, da er die Möglichkeit, sich zu verbessern, verpasst hat und dadurch andere Nutzer überholen können.

Zusätzlich zu den Szenarien gibt es zwei weitere Möglichkeiten. Zum einen kann ein Experte einem Nutzer für eine gute Bewertung belohnen und ihm dadurch zwei Zusatzpunkte einbringen. Diese werden auch dann eingerechnet, wenn dem Nutzer Punkte abgezogen werden sollen. Zum anderen gibt es den Fall, dass ein Nutzer keine oder eine leere Bewertung abgibt. Beim ersten Mal werden ihm dann 15 Punkte abgezogen. Hat er sich also bisher korrekt verhalten, hat er eine zweite Chance. Sollte er ein weiteres Mal keine Bewertung abgeben, werden ihm dreißig Punkte abgezogen. Nur Nutzer, welche bereits viel und korrekt bewertet haben, würden danach noch ein Scoring im positiven Wertebereich haben. Entsprechend sollen auch nur diejenigen nach einer zweiten fehlenden Bewertung noch eine letzte Chance erhalten. Wird ein drittes Mal keine oder eine leere Bewertung abgegeben, verliert der Nutzer seinen Status als qualifizierter Nutzer unabhängig von seinem Scoring. Ein Nutzer kann sich davor schützen, indem er seinen Status als qualifizierte Nutzer in seinem Profil pausiert. Ist dies länger als ein Jahr der Fall, verliert er allerdings auch seinen Status, ansonsten werden in dieser Zeit seine Punkte eingefroren.

Auf lange Sicht kann außerdem eine Frist eingeführt werden, ab wann ein Vergehen vergessen wird. So kann man beispielsweise von der Stufe, dass man eine leere Bewertung abgegeben hat, wieder auf die Ausgangsstufe versetzt werden, wenn man eine gewisse Anzahl von ordentlichen Bewertungen seit dem Vergehen abgegeben hat. Das kann eine Möglichkeit für eine Weiterentwicklung des Konzepts darstellen, da es zum Zeitpunkt der Entwicklung des Konzepts und der Plattform nur eine geringe Rolle spielt.

Auf die Möglichkeiten der Verknüpfung von Ranking und Auswahrscheinlichkeit soll in diesem Kapitel nur kurz eingegangen werden. Zu den möglichen Verfahren gehören verschiedene Skalierungsmöglichkeiten – so kann einem Nutzer mit dem Rang $x$ eine Auswahlwahrscheinlichkeit von $\frac{1}{x+1}$ zugeordnet werden, was einer proportionalen Skalierung

entspricht. Dem Nutzer auf dem ersten Rang wird also eine Wahrscheinlichkeit von 0,5 zugeordnet werden und die Wahrscheinlichkeit würde dann mit jedem weiteren Rang abnehmen. Bei einer linearen Skalierung hingegen würde dem ersten Rang eine Wahrscheinlichkeit von 1 und dem letzten Rang 0 zugeordnet und die Wahrscheinlichkeit der Ränge dazwischen würde linear abnehmen. Wobei der Abstand abhängig von der Anzahl der vergebenen Ränge $n$ wäre. Die Wahrscheinlichkeit eines Nutzers berechnet sich dann aus $1 - \left(\frac{x-1}{n-1}\right)$. Eine nähere Betrachtung, wie das Ranking einen Einfluss auf die Auswahl haben kann, wird im Umsetzungskapitel Auswahl Nutzer behandelt werden.

### 4.5.2 Anreizsystem

Für die Nutzer soll ein Anreiz geschaffen werden, um eine Bewertung für einen neuen Kurs abzugeben. Bereits aus dem Prozess ergibt sich der Vorteil, dass auf den neuen Inhalt des Kurses vorzeitig und kostenfrei zugegriffen werden kann. Außerdem kann auch die Rangliste der qualifizierten Nutzer dazu motivieren, immer wieder neue, gute Bewertungen abzugeben, um seinen Platz in der Liste zu verteidigen oder zu verbessern.

Zusätzlich zu den genannten Vorteilen soll ein weiterer Anreiz im System integriert werden. Dadurch sollen die Nutzer motiviert werden, sich an diesem Prozess zu beteiligen und sie sollen für ihren Aufwand belohnt werden. Als Mittel dafür stehen verschiedene Möglichkeiten zur Verfügung. Infrage kommen dabei immaterielle Güter, wie beispielsweise die Learntoken, welche bereits auf der BCAM-Plattform integriert sind oder auch eine neue Art von Token. Vorteil wäre dabei für die Plattform, dass kaum weitere Kosten entstehen und auch der Aufwand dafür gering ist. Nachteil ist, dass der Nutzer darin keinen ausreichend großen Anreiz sehen könnte. Schließlich hat er bereits Learntoken gesammelt, um überhaupt Kurse bewerten zu dürfen. Und auch eine neue Art Token müsste mit einem weiteren Vorteil verbunden werden, um ihn attraktiv zu machen.

Zu den materiellen Gütern, welche als Anreiz genutzt werden können, zählen Rabatte auf Kursinhalte, Geld oder auch andere Belohnungen, welche über die Plattform hinausgehen können, in Form von Kooperationen. Diese Sachen können einen höheren Anreiz darstellen als immaterielle Güter, sind aber auch mit einem höheren Investment der Plattform verbunden. Zusätzlich könnten sie zeitweise nicht verfügbar sein – beispielsweise könnte gerade keine Kooperation aktiv sein oder auch gerade nicht ausreichend Geld der Plattform für die Belohnungen zur Verfügung stehen. Wichtig ist, dass die Belohnung nicht im Zusammenhang mit der Art der Bewertung steht. Würde ein Nutzer, der einen Kurs, bewertet an dessen Gewinn beteiligt werden, wäre das für ihn ein Anreiz, eine unehrliche, bessere Bewertung abzugeben.

Eine passender Ansatz kann ein Konzept mit Prämien sein. Das heißt, die vorgestellten Güter werden, wenn sie vorhanden sind, als Prämien angeboten. Das hat den Vorteil, dass die Plattform flexibel darin sein kann, welche Belohnungen sie ausgibt. Die Ausgabe der Prämien könnte dann zu einem für den Nutzer zufälligen Zeitpunkt erfolgen. Das

wiederum kann allerdings den Effekt der Belohnung schmälern, da sie nicht direkt mit der Abgabe der Bewertung einherging. Um sowohl Flexibilität der Plattform als auch den Belohnungseffekt in einem Ansatz zu verbinden, bietet sich folgender Ansatz an.

Nach der Abgabe der Bewertung und Verarbeitung der Bewertung erfolgt eine automatisierte Ausgabe von Token an die Nutzer. Diese Token sind ausschließlich für den hier beschriebenen Prozess vorgesehen und heißen Prämientoken. Diese Prämientoken können dann in einem Prämienshop zu einem beliebigen Zeitpunkt eingetauscht werden. Das Angebot im Prämienshop kann dabei variieren und wird durch die Plattform angepasst. Darin können dann die bereits genannten Dinge – Learntoken, Rabatte, Geld, Gutscheine, Aktionen aus Kooperationen angeboten werden. Das Angebot wird dabei durch die Plattform bestimmt.

Die Ausgabe der Prämientoken erfolgt im Zusammenhang mit der Vergabe der Punkte für das Scoring. Immer dann, wenn eine korrekte Bewertung abgegeben wurde, erhält der Nutzer die volle Höhe der Prämientoken. Erhält er aufgrund verzeihbarer Fehler vier Punkte, so erhält er noch die Hälfte der eigentlichen Anzahl an Prämientoken. In allen anderen Fällen werden keine Prämientoken an die Nutzer ausgegeben.

Die letztendliche Ausgestaltung des Shops, dessen Inhalt und Preisgestaltung sowie auch die Höhe der Prämientoken ist abhängig von der jeweiligen Plattform. Letztendlich sollte das Konzept des Prämienshops im Verhältnis zu den ausgegebenen Belohnungstoken stehen und zu den Ressourcen der Plattform passen.

# 5 Entwicklung eines Prototyps

Für die Entwicklung des Prototypen steht die Umsetzung von Teilen des Konzepts in einem Smart Contract im Fokus. Als Entwicklungsumgebung wurde eine lokale Blockchain mithilfe von Ganache genutzt und die Truffle Suite zum Interagieren und Testen der Funktionen. Im Fokus stehen besonders die Prozesse, an denen die Nutzer beteiligt sind und wie der grundlegende Aufbau des Smart Contracts zu gestalten ist.

## 5.1 Aufbau

Für den grundlegenden Aufbau des Smart Contracts wurde sich mit den benötigten Datenstrukturen und Rollen auseinandergesetzt. Außerdem wurden einige Ansätze aus dem Konzept angepasst, um effizient auf einer Blockchain abbildbar zu sein.

Aus dem Konzept ergeben sich die Rollen qualifizierter Nutzer, Experte und Plattformbetreiber. Die Rolle des unqualifizierten Nutzers ist für die Prozesse zur Bewertung vernachlässigbar und muss daher in diesem Smart Contract keine Beachtung finden. Aus diesem Grund wird der qualifizierte Nutzer im weiteren Nutzer oder User genannt. Die Rolle des Plattformbetreibers kann erweitert werden zu einer Gruppe von Verantwortlichen für die Plattform (Operators). So können mehrere Adressen die besonderen Rechte eines Operators nutzen und es nicht einer allein für das Aufrufen der Operator-Funktionen zuständig. Für alle drei Rollen braucht es Datenstrukturen, um sie im Smart Contract abzubilden. Zusätzlich müssen die Vorgänge für das Aufnehmen und Entfernen von neuen Nutzern, Experten und Plattformverantwortlichen bedacht werden.

Die Rolle der Plattformverantwortlichen ist die, welche am wenigsten Daten benötigt. Letztendlich reicht ein Mapping, in welchem allen Operator-Adressen der Wahrheitswert true zugeordnet wird. Das ermöglicht auch die Entwicklung eines Modifiers, mit welchem vor dem Aufruf einer anderen Funktion geprüft werden kann, ob die aufrufende Adresse zur Operatoren-Gruppe gehört.

```
55      mapping(address => bool) public operators;
56      // Modifier to check if caller is Operator
57      modifier onlyOperator {
58          require(operators[msg.sender], "Only for Operators");
59          _;
60      }
```

**Abbildung 4: Smart Contract Datenstruktur Operator**

Die Funktionen zum Hinzufügen und Entfernen von Operatoren gehören eher zu organisatorischen Vorgängen und werden daher im Prototyp noch nicht umgesetzt. Vorstellbar ist ein Voting zwischen den bestehenden Operatoren, wenn ein neuer Operator hinzugefügt werden soll. Gleiches kann auch für einen Ausschluss geschehen. Für diesen ersten Entwurf eines Smart Contracts kann im Constructor ein Array mit Adressen von Plattformverantwortlichen übergeben werden, welche dann der Gruppe von Operatoren hinzugefügt werden.

Für die Rolle des Experten gibt es ebenfalls ein Mapping, in welchem die Adressen hinterlegt werden, zusätzlich soll es für Experten ein Struct geben. In diesem werden Informationen, wie Felder der Expertise oder wann der letzte Bewertungsprozess war, abgelegt. Dieses Struct wird dann der Adresse im Mapping zugeordnet. Um einen neuen Experten im Smart Contract aufzunehmen, muss die Funktion addExpert mit der Angabe der Adresse und seiner Fachrichtungen durch einen Operator aufgerufen werden. Hier kann der zuvor vorgestellte Modifier onlyOperator genutzt werden, um zu gewährleisten, dass nur Plattformverantwortliche diese Funktion aufrufen können. Gleiches gilt auch für das Entfernen des Experten aus dem Smart Contract. Auch diese Funktion gehört zu den organisatorischen Vorgängen und kann durch die Plattformverantwortlichen ausgelöst werden.

Die komplexeste Datenstruktur haben die Nutzer. Neben der Ethereum-Adresse des Nutzers müssen sein aktueller Status, seine Punkte, die letzte Bewertungsrunde und weiteres abgelegt werden. Zusätzlich ist es durch die Unterscheidung im Konzept zwischen neuen Nutzern und Nutzern im Ranking wichtig, dies auch in der Umsetzung abbilden zu können. Die folgende Abbildung zeigt eine Auswahl benötigter Datenstrukturen für Nutzer.

```
72  //Iterating through úser addresses
73  address[] public rankedUsersArray;
74  address[] public newUsersArray;
75  //Access user struct via address
76  mapping(address => User) public usersMapping;
77  struct User {
78      bool registered;
79      bool active;
80      bool newUser;
81      int256 points;
82      uint256 lastVotingRound;
83      uint256 index;
84  }
85  mapping(address => bool) public blockedUsers;
```

**Abbildung 5: Smart Contract Datenstruktur User**

Neben dem Struct User und dem Mapping, in welchem es abgelegt wird, werden die Nutzeradressen in die beiden Arrays (Zeile 73 und 74) aufgeteilt. Die Verwendung von Arrays zusätzlich zum Mapping ist nötig, da nur durch Arrays iteriert werden kann. Dies ist vor

allem für den Prozess der Nutzerauswahl wichtig. Im rankedUsersArray sind ausschließlich Adressen von Nutzern, welche bereits an einer Bewertungsrunde teilgenommen haben. Im newUsersArray hingegen sind alle Adressen, welche zu neuangemeldeten Nutzern gehören. Im usersMapping wird allen Nutzeradressen ein Struct User zugeordnet.

In diesem Struct sind mehrere Informationen über den Nutzer abgelegt. Die Wahrheitswerte registered, active und newUser werden verwendet, um schnell Voraussetzungen für Funktionen (bspw. die Teilnahme an einer Bewertungsrunde) prüfen zu können. Ein Nutzer muss entsprechend registriert und als aktiv hinterlegt sein, um ausgewählt werden zu können. Ein Nutzer kann sich selbst über die Plattformseite flexibel deaktivieren, um bspw. während einer inaktiven Phase eine Auswahl als Bewerter zu verhindern. Der Wahrheitswert newUser ist vor allem für die Auswertung nach der ersten Bewertungsrunde eines Nutzers wichtig, da er dann bereits nicht mehr im newUsersArray aufgelistet wird. Dennoch muss er als neuer Nutzer erkennbar sein, damit die entsprechende Staffelung bei der Punktevergabe angewendet werden kann.

Der Punktestand des Nutzers ist in dem Integer-Wert points abgelegt. Er wird entsprechend im Zuge der Auswertung erhöht oder verringert. Um mit einem Abzug von Punkten in Form von einer negativen Zahl kompatibel zu sein, ist der Wert nicht als unsigned integer definiert, sondern als int256. Negative Punktstände selbst werden jedoch ausgeschlossen, da diese Nutzer ihren Status als qualifizierte Nutzer verlieren und damit ihre Datenstrukturen entfernt werden. Die Information der lastVotingRound gibt Auskunft darüber, bei welchem Bewertungsprozess der Nutzer zuletzt teilgenommen hat. Alternativ könnte hier auch ein Array genutzt werden, indem alle Bewertungsrunden des Nutzers aufgezählt werden. Der Index eines Nutzers ist ausschließlich für die internen Funktionen nötig und gibt den Index im entsprechenden Array an.

Das Mapping für blockierte Nutzer wird für eine schnelle Überprüfung bei der Anmeldung als neuer Nutzer verwendet, um eine Neuanmeldung von bereits ausgeschiedenen Nutzern zu verhindern. Mit dem Ausschluss aus der Gruppe der qualifizierten Nutzer entsprechend des Konzepts, wird der Nutzer aus dem User-Mapping und dem Array entfernt. Daher benötigt es eine extra Datenstruktur, welche die Information über blockierte Nutzeradressen erhält.

Im Umgang mit den Nutzern ist eine Abweichung vom Konzept nötig. Im Smart Contract selbst, wird es aufgrund von den knappen Ressourcen bezogen auf Speicherplatz und Rechenkapazität auf einer Blockchain, kein Ranking geben. Es gibt also weder eine Datenstruktur, welche die Rangfolge widerspiegelt, noch wird den einzelnen Nutzern eine Rangnummer zugeordnet. Würde man dies auf in einem Smart Contract umsetzen wollen, würde mit jeder Aktualisierung der Punkte, ein Iterieren über die Datenstrukturen und ein Sortiervorgang einhergehen. Auch mit einem guten Sortieralgorithmus würde dies unnötig hohe Kosten verursachen. Daher ist es leichter, die Information über die Punktstände auf der Blockchain zu haben, aber eine Rangliste dann Off-Chain zu erstellen.

Da der Rang auch bei der Auswahl der Nutzer eine Rolle spielen sollte, muss auch dafür eine Anpassung vorgenommen werden. Dafür ist es leichter, die Punkte der Nutzer zu verwenden, anstelle eines Rangs. Dadurch wird das Auswahlverfahren sogar gerechter, da die Punktzahl die Abstände zwischen Nutzern genauer widerspiegelt als der Rang. Im Auswahlverfahren wird also die Wahrscheinlichkeit, dass ein Nutzer ausgewählt wird, nicht von seinem Rang abhängig sein, sondern von seiner Punktzahl.

```
10  uint256 public votingRound = 0;
11  struct VotingRound {
12      uint256 round;
13      bool active;
14      address[] participantsUsers;
15      address[] participantsExperts;
16      mapping(address => UserEvaluation) allUsersEvaluation;
17      mapping(address => uint256[7]) allExpertsEvaluation;
18  }
19  VotingRound[] public votingRounds;
```

**Abbildung 6: Smart Contract Datenstruktur VotingRound**

Als letzte Datenstruktur soll noch die Voting Round vorgestellt werden. Jede Voting Round steht für einen Bewertungsprozess und enthält alle relevanten Informationen davon. Zu den Informationen zählen die Adressen der Teilnehmer und Experten jeweils in einem Array, die Bewertungen von ihnen in einem Mapping und die eindeutige Rundennummer. Für die Rundennummer gibt es im Contract selbst eine Zählervariable, welche mit jeder neuen Runde erhöht wird. Die Informationen werden in einem Struct (VotingRound) gesammelt und die Structs wiederum in einem Array abgelegt. Da der Zugriff auf die Runden über ihre Rundennummer erfolgt, ist es sinnvoll, dass die Rundennummer eines Structs, seinem Index im Array entspricht. Wie genau die Bewertungen in der Datenstruktur abgelegt werden, ist Thema im Kapitel Abgabe der Bewertung.

Mit diesen Ausführungen wurden die wichtigsten Datenstrukturen des Evaluation Smart Contracts erklärt. Sie bilden die Grundlage für die Umsetzung der Prozesse aus dem Konzept in Funktionen auf der Blockchain. Auf diese soll im nächsten Kapitel eingegangen werden.

## 5.2 Prozess der Nutzerbewertung

In diesem Kapitel werden die Prozesse näher betrachtet werden, bei denen der Nutzer beteiligt ist. Diese stehen im Fokus des Prototypen und die Vorgehensweise bei der Entwicklung wird vorgestellt werden. Dazu zählen zum einen die Betrachtung der Herausforderung einer verdeckten Bewertung und wie dieser begegnet werden soll. Zum anderen wird der genaue Ablauf einer Nutzerbewertung im Zusammenspiel von Plattformverantwortlichen, Nutzer und Smart Contract incl. Codebeispielen dargestellt werden.

### 5.2.1 Verdeckte Bewertung durch Nutzer

Bei der Umsetzung der Bewertungsabgabe durch die Nutzer stellt vor allem die Anforderung einer verdeckten Abgabe eine Herausforderung dar. Wenn der Nutzer seine Bewertung einmal an den Smart Contract übergeben hat, sind die Werte für alle einsehbar. Das Problem damit wäre, dass andere Nutzer diese Bewertung einfach übernehmen könnten, anstatt selbst eine Evaluierung der Inhalte durchzuführen. Zum Zeitpunkt der Auswertung der Bewertungen müssen die Werte jedoch auf der Blockchain verwertbar sein und eindeutig dem Bewerter zuordenbar sein.

Um diese Herausforderung zu lösen, eignet sich ein Verfahren, bei dem die Nutzer die Bewertung zunächst verschlüsselt auf der Blockchain ablegen. Vor der Auswertung erfolgt dann die Offenlegung der Werte. Dies kann beispielsweise durch ein Commit-Reveal-Verfahren umgesetzt werden. Hierbei verschlüsselt der Nutzer die Bewertung mit einem Key und übermittelt diesen verschlüsselten Wert (Commitment) an den Smart Contract. In der Reveal-Phase muss dann der Nutzer die entschlüsselte Bewertung und den verwendeten Schlüssel offenlegen. Dadurch kann der Smart Contract das Commitment selbst entschlüsseln und die Übereinstimmung der Bewertung prüfen. So wird sichergestellt, dass die Bewertung zunächst vertraulich bleibt, aber im Nachhinein Betrug bei der Abgabe ausgeschlossen werden kann.

Ein Nachteil dieses Verfahrens ist, dass zwei Interaktionen durch den Nutzer erforderlich sind. Er muss zunächst sein Commitment auf die Blockchain schreiben und zu einem späteren Zeitpunkt eine Reveal-Funktion aufrufen. Dadurch verlängert sich einerseits die Dauer des Bewertungsprozesses, da die Reveal-Phase zusätzliche Zeit benötigt. Andererseits besteht das Risiko, dass Nutzer nicht an der Reveal-Phase teilnehmen und ihre Bewertung dadurch unbrauchbar wird. Obwohl dies wie eine leere Bewertung bestraft werden kann, wäre es dennoch nicht möglich, die abgegebene Bewertung zu verwenden. Außerdem sollte auch der Nutzer möglichst wenig belastet werden, um die Motivation für die Teilnahme am Bewertungsprozess aufrechtzuerhalten. Für den Nutzer wäre es wesentlich einfacher, wenn er lediglich seine (verschlüsselte) Bewertung abgeben müsste.

Daher wurde das Commit-Reveal-Verfahren angepasst, sodass der Nutzer weniger Verantwortung trägt. Stattdessen werden die Operatoren der Plattform mit in den Prozess einbezogen. Indem sie die Reveal-Phase übernehmen, können die Nutzer entlastet werden. Damit dies möglich ist, muss die Plattform die Keys der Nutzer kennen, um ihre Bewertungen zu entschlüsseln und beides dem Smart Contract im Reveal-Prozess mitzuteilen. Eine Möglichkeit ist, dass die Nutzer mit Abgabe ihrer Bewertung ihr Commitment auf die Blockchain schreiben und den Key an die Plattform (Off-Chain) übermitteln.

Ein weiterer Ansatz wäre, dass die Plattform für die ausgewählten Nutzer eines Bewertungsprozesses, jeweils einen Key erstellt und diesen ihnen dann mitteilt. Dadurch liegt die Verantwortung für die Erstellung der Keys ebenfalls bei der Plattform und nicht beim Nutzer. So muss der Nutzer den Key nicht an die Plattform übermitteln, sondern nur eine

Transaktion an den Smart Contract mit seinem Commitment auslösen. Sobald die Commitments aller Nutzer eines Bewertungsprozesses im Smart Contract vorhanden sind oder die Zeitspanne für die Bewertungsabgabe abgelaufen ist, kann ein Verantwortlicher der Plattform mit einer Transaktion alle Keys an den Smart Contract übermitteln und die Bewertungen offenlegen.

Neben den Vorteilen, dass durch diesen Ansatz der Nutzer entlastet wird und der Prozess nicht stark verlängert wird, bringt der Ansatz gegenüber dem klassischen Commit-Reveal-Verfahren auch Nachteile mit sich. Dazu zählt, dass das Verfahren eine zentrale Instanz hat, von der der Prozess abhängig ist. Die Verantwortlichen der Plattform könnten das Reveal der Bewertungen verhindern – beispielsweise, wenn zu viele negative Bewertungen dabei sind, die sie nicht veröffentlichen möchten. Sollten sie keine Keys an die Nutzer geben, könnte der komplette Bewertungsprozess nicht durchführbar sein. Das wiederum könnte zu einer Bestrafung der Nutzer führen, welche aber keinen Einfluss darauf hatten. Trotz dieser negativen Szenarien ist das Risiko für deren Eintreten gering. Denn die Verantwortlichen der Plattformen haben vorrangig Interesse daran, dass die aufgesetzten Prozesse funktionieren und Nutzer gerecht behandelt werden. Selbst wenn die Bewertungen negativ sind, ist es dennoch im Interesse der Plattform die Bewertung zu veröffentlichen, um ein Argument zu haben, einen schlechten Kurs abzulehnen. Eine Plattform, welche Prozesse für eine transparente Bewertung durch Nutzer aufsetzt, sollte diese nicht manipulieren wollen. Andernfalls könnte sie eine andere Form der Bewertung nutzen oder darauf verzichten und es zentral entscheiden.

Aus diesen Überlegungen ergibt sich, dass die abgewandelte Form des Commit-Reveal-Schemas für das vorgestellte Konzept sinnvoll ist. Zusätzlich könnte eine Sicherheitsmaßnahme integriert werden, bei der dem Nutzer die Möglichkeit gegeben wird, selbst den Key zu veröffentlichen. Grundlegend kann aber von einer korrekten Vorgehensweise durch die Plattform ausgegangen werden, solang diese sich nicht selbst schaden will. Daher überwiegen die Vorteile durch die Entlastung der Nutzer und der Ansatz wird in der Umsetzung verfolgt werden.

### 5.2.2 Ablauf für den Bewertungsprozess der Nutzer

Die folgende Abbildung zeigt die Interaktionen zwischen dem Evaluation Smart Contract, einem Nutzer und einem Operator. Dabei sind die Prozesse von Anmeldung des Nutzers als qualifizierter Nutzer bis zur Veröffentlichung seiner Bewertung auf der Blockchain abgebildet. Damit ist der Bewertungsprozess zwar nicht beendet, aber der Nutzer selbst muss danach nicht mehr mit dem Contract interagieren. Daher werden die weiteren Prozesse für eine bessere Übersicht in einer anderen Abbildung gezeigt.

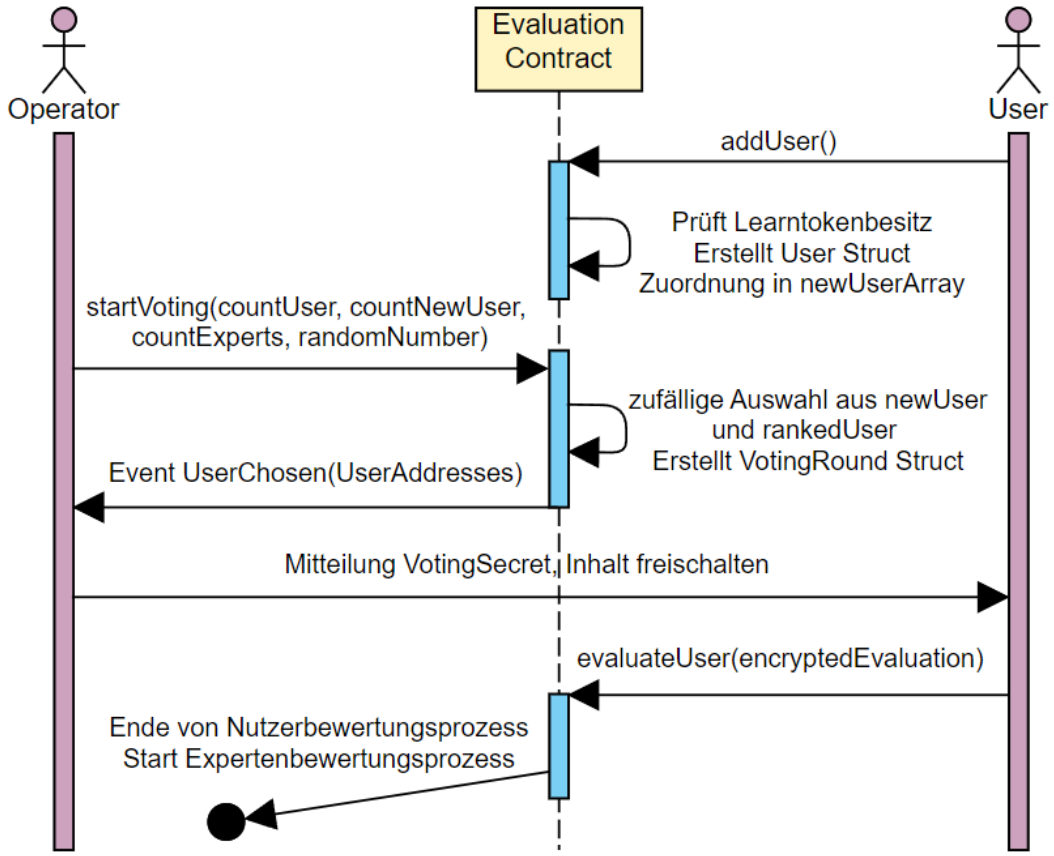



**Abbildung 7: Interaktionen Nutzer - Smart Contract**

#### *5.2.2.1 Anmeldung Nutzer*

Der Ablauf beginnt mit dem Nutzer, welcher als qualifizierter Nutzer am Bewertungsprozess teilnehmen möchte. Dafür ruft er die Funktion addUser im Evaluation-Contract auf. Dieser kann aus dem Learntoken Contract auslesen, ob der Nutzer ausreichend Learntoken besitzt, um qualifizierter Nutzer zu werden. Außerdem prüft er in seinem eigenen Mapping, ob die Nutzeradresse blockiert ist. Dies könnte der Fall sein, wenn der Nutzer bereits einmal qualifizierter Nutzer war, dann aber diesen Status verloren hat. Hat der Nutzer beider Überprüfungen bestanden, wird ein neues User Struct für ihn angelegt, dieses im entsprechenden Mapping abgelegt und die Nutzeradresse wird außerdem im Array für neue Nutzer aufgenommen. Damit ist die Anmeldung als qualifizierter Nutzer im Evaluation Smart Contract abgeschlossen.

#### *5.2.2.2 Erzeugung Zufallszahl*

Die Funktion startVotingRound ist nur durch Operatoren aufrufbar und der Funktionsaufruf erfolgt nach der Inhaltsfreigabe durch den Ersteller und einer bestandenen automatischen Vorprüfung. Ziel ist es, dass die Auswahl der Nutzer zufällig erfolgt und die Wahrscheinlichkeit von ihrer Punktzahl abhängig ist. Folgende Parameter werden beim Aufruf durch einen Plattformverantwortlichen mitgegeben:

1. countUser – die Anzahl der Nutzer, welche basierend auf ihrem Ranking zufällig ausgewählt werden sollen
2. countNewUser - die Anzahl der Nutzer, welche aus der Gruppe neuer Nutzer zufällig ausgewählt werden sollen
3. countExperts – die Anzahl der Experten
4. randomNumber – eine Zufallszahl, welche durch die Plattformbetreiber vorgegeben wird

Die Entscheidung, dass der Input für die zufällige Auswahl Off-Chain generiert und durch die Plattform vorgegeben wird, hat Vor- und Nachteile. Zu den Nachteilen zählt, dass dadurch die Macht der Plattform gesteigert wird und dadurch das Konstrukt an Dezentralität verliert. Außerdem muss die Plattform, wenn sie ehrlich vorgehen will, sich um einen guten Zufallsgenerator bemühen. Vorteil hingegen ist, dass der Prozess eine Zufallszahl zu erhalten ausgelagert ist und damit weder mit On-Chain Kosten verbunden ist noch sich mit den Sicherheitsbedingungen einer auf Blockchain basierenden Zufallsfunktion auseinandersetzen muss. Denn Zufall auf einer Blockchain zu erzeugen ist mit einigen Hürden verbunden, wie die Betrachtung der folgenden beiden Möglichkeiten zeigt.

Beispielsweise wurde im EIP-4399[8] beschrieben, wie die Zufälligkeit, welche auf der Beacon Chain durch RANDAO (Random Number Generation through Decentralized Autonomous Organization) erzeugt und genutzt wird, auch für die EVM zugänglich gemacht werden kann. Durch die Umsetzung und Aufnahme dieses EIP ist der Output von RANDAO im Blockheader integriert und kann daher innerhalb von Smart Contracts mit block.prevrandao abgefragt und verwendet werden. In der aktuellen Umsetzung ist dies nur für den vorhergehenden Block möglich, das bedeutet, dass zu dem Zeitpunkt alle Netzwerkteilnehmer die Zufallszahl kennen. (Kalinin und Ryan 2021)

Ein mögliches Workaround dafür ist, dass man auf einen Block in der Zukunft wartet und dann dessen Zufallszahl nutzt. Problem dabei ist, dass wenn man einen bestimmten Block definiert, auf den gewartet wird, kann es sein, dass die eigene Transaktion nicht in dem Block mit dieser Nummer aufgenommen wird und dann keine Zufallszahl vorhanden ist. Zusätzlich werden dafür zwei Transaktionen benötigt. Wird nicht ein spezifischer Block in der Zukunft festgelegt, sondern alle Blöcke, mit einer Blocknummer, die größer oder gleich der vorgegebenen Blocknummer sind, erhält der Validator dadurch noch mehr Einfluss. Denn er kann die Transaktion so lange zurückhalten, bis ein Block kommt, dessen prevrandao-Wert für ihn passt. In Zukunft soll eine Abfrage des prevrandao auch für spezifische ´Blocknummern möglich sein, was den Prozess vereinfacht. (Waas 2023)

[8] https://eips.ethereum.org/EIPS/eip-4399

Letztendlich kann die Nutzung des prevrandao Werts durchaus in Betracht gezogen werden, da die Plattform dadurch weniger Entscheidungsgewalt hat. Sie kann dadurch nur entscheiden, ob sie die durch prevrandao vorgegebene Zufallszahl verwenden will oder nicht, indem sie die Transaktion entsprecht terminiert. Während sie, wenn sie selbst die Zufallszahl auch erstellen soll, den Ausgang der Auswahl auch komplett beeinflussen kann.

Eine weitere mögliche Lösung für Zufallszahlen auf der Blockchain sind Verifiable Random Functions (VRF). Eine Implementierung dafür bietet Chainlink[9]. Mit ihrem Modul Chainlink VRF wird auf Anfrage eine Zufallszahl basierend auf dem VRF-Verfahren Off-Chain erzeugt. Diese Zufallszahl wird gemeinsam mit dem Beweis für die Gültigkeit der Zufallszahl zurückgegeben. (Chainlink Documentation) Diese Umsetzung ist sicherer als die Nutzung der RANDAO-Variable, vor allem erzeugt sie echte Wahrscheinlichkeit anstatt Pseudo-Wahrscheinlichkeit. Allerdings ist sie mit Kosten verbunden, da Chainlink für ihren Dienst eine Bezahlung in LINK-Token fordert. Außerdem ist man von ihrem Dienst abhängig und hat dadurch einen Single Point of Failure, auf den man keinen Einfluss hat in seiner Umsetzung.

Aus diesen Überlegungen ergibt sich, dass zumindest für die Entwicklung des Prototyps die Erzeugung einer Zufallszahl durch die Plattform selbst erfolgen soll. Auch dies kann natürlich ausfallen, allerdings kann sich dann die Plattform selbst um eine Verbesserung bemühen. In einer produktiven Anwendung kann dann auch der Einsatz von VRF bei der Erstellung der Zufallszahlen auf der Plattform sinnvoll sein und das Vertrauen in diesen Vorgang erhöhen. Zusätzlich sollte auch die Weiterentwicklung für den prevrandao-Wert verfolgt werden, um einen möglichen Einsatz dieses Werts neu zu evaluieren.

#### *5.2.2.3 Auswahl Nutzer*

Mit den übergebenen Parametern wird im Smart Contract dann die Auswahl der Nutzer und Experten vorgenommen und die Auswahl einem neuen VotingRound Struct abgelegt. Wie die zufällige Auswahl basierend auf den Punkten der Nutzer erfolgt, zeigt der folgende Auszug aus dem Smart Contract.

[9] https://chain.link/vrf

```
210  function chooseRandomUser(uint256 count, uint256 randomNumber) public
211      view onlyOperator() returns (address[] memory) {
212      require(count  > 0, "Invalid Number of count");
213      require(rankedUsersArray.length >= count, "Not enough Users available");
214
215      if(count == rankedUsersArray.length) {
216          return rankedUsersArray;
217      }
218      uint256 _totalPoints = uint256(totalPoints);
219      address[] memory chosenUser  = new address[](count);
220      //Extra Array to prevent duplicates
221      address[] memory copyUserArray = rankedUsersArray;
222      uint256 arrayLength = copyUserArray.length;
223      for(uint256 j = 0; j < count; j++) {
224          uint256 sum = 0;
225          uint256 random = randomNumber % uint256(_totalPoints);
226          for(uint256 i = 0; i < arrayLength; i++) {
227              address userAddress = copyUserArray[i];
228              sum += uint(usersMapping[userAddress].points);
229              if(random <= sum ) {
230                  chosenUser[j] = userAddress;
231                  _totalPoints -= uint256(usersMapping[userAddress].points);
232                  // Move chosen User to last place in array
233                  copyUserArray[i] = copyUserArray[arrayLength - 1];
234                  // Workaround pop for Array in memory
235                  delete copyUserArray[arrayLength-1];
236                  arrayLength--;
237                  break;
238              }
239          }
240      }
241      return chosenUser;
242  }
```

**Abbildung 8: Smart Contract Function chooseRandomUser**

Mit dieser Funktion zur Auswahl von Nutzern für einen Bewertungsprozess wird die Bedingung erfüllt, dass die Wahrscheinlichkeit eines Nutzers proportional zu seinen verdienten Punkten erfolgen soll. Die Grundlage des Verfahrens bildet die übergebene Zufallszahl, welche dann auf die Gesamtanzahl vergebener Punkte im Smart Contract an Nutzer gekürzt wird (Zeile 225). Da diese Zahl entsprechend verkleinert werden muss, nachdem ein Nutzer ausgewählt wurde (Zeile 231), wird in der Funktion mit einer Kopie der Contract Variable totalPoints gearbeitet (Zeile 218).

Für die Wahl eines Nutzers (For-Schleife Zeile 225) wird durch das Array der gerankten Nutzer iteriert und deren Punkte in der Hilfsvariable sum aufaddiert (Zeile 228). Sobald dieser Wert größer oder gleich dem Wert der Zufallszahl ist, gilt der Nutzer, dessen Punkte zuletzt addiert wurden als ausgewählt (Zeile 229, 230). Nutzer, die viele Punkte haben, decken automatisch mehr Werte auf der Skala von null bis zur Gesamtpunktzahl

ab. Daher ist die Wahrscheinlichkeit, dass mit ihren Punkten der Zufallswert erreicht oder überschritten wird, höher als bei Nutzern, die wenig Punkte haben.

Nach der Auswahl eines Nutzers, wird seine Adresse dem Ergebnisarray chosenUser hinzugefügt, seine Punktzahl von der Gesamtpunktzahl abgezogen (Zeile 231) und seine Adresse aus der Kopie des Array der gerankten Users entfernt (Zeile 232 – 236). Für die Wahl des nächsten Nutzers wird die Zufallszahl so angepasst, dass sie wieder in der Spanne der Gesamtpunktzahl der noch verfügbaren Nutzer liegt (Zeile 255). Am Ende wird ein Array zurückgegeben, welches die Adressen der ausgewählten Nutzer enthält.

Für die Wahl der Nutzer aus dem Array newUsers müssen keine Punkte beachtet werden. Stattdessen kann die Zufallszahl als Arrayindex direkt die Adresse des ausgewählten Nutzers liefern, nachdem sie auf die Anzahl an Nutzern eingeschränkt wurde. Sobald ein neuer Nutzer einer VotingRound zugeordnet wurde, wird seine Adresse aus dem Array der neuen Nutzer entfernt, um eine erneute Auswahl des Nutzers über diese Gruppe zu verhindern. Daher ergibt sich folgende For-Schleife für die Wahl bei den neuen Nutzern:

```
address[] memory chosenUser  = new address[](count);
//Schleife für count
for(uint256 j = 0; j < count; j++) {
    //Wahl Nutzer basierend auf randNumber
    uint256 newUserID = randNumber % newUsersArray.length;
    chosenUser[j] = newUsersArray[newUserID];
    //remove User from newUsersArray to prevent doublings
    uint256 _index = usersMapping[userAddress].index;
    newUsersArray[_index] = newUsersArray[newUsersArray.length-1];
    newUsersArray.pop();
    //Update Index from Moved User
    usersMapping[newUsersArray[_index]].index = _index;
}
return chosenUser;
```

**Abbildung 9: Smart Contract For-Schleife chooseNewUsers**

In der Funktion startVotingRound wird außerdem bei allen ausgewählten Nutzern die aktuelle Votingrundennummer hinterlegt, sodass nachvollzogen werden kann, wann ihre letzte Teilnahme war. In der Weiterentwicklung des Prototyps kann dann geprüft werden, ob ein zufällig gewählter Nutzer tatsächlich auch ausgewählt werden kann. Dafür sollte sein Status geprüft werden, ob er aktiv ist oder ob er noch pausiert. Außerdem sollte geprüft werden, dass der Mindestabstand von zwei Bewertungsprozessen zwischen zwei Teilnahmen eingehalten wird. Dies kann mit der hinterlegten Nummer der Votingrunde, an welcher der Nutzer zuletzt teilgenommen hat, ermittelt werden.

#### *5.2.2.4 Abgabe der Bewertung*

Die Funktion zum Starten einer neuen Bewertungsrunde löst das Event UserChosen aus, auf welches die Plattform dann reagieren kann. Sobald die Adressen der ausgewählten Nutzer der Plattform bekannt sind, erzeugt sie für jede Adresse ein Secret. Anschließend gibt sie die zu bewertenden Inhalte an die ausgewählten Nutzer frei. Bei der Abgabe der Bewertung über die Plattform wird dem Nutzer das Secret mitgeteilt und mit dem Secret die Bewertung verschlüsselt. Diese verschlüsselte Bewertung wird dann mit einer Transaktion, welche die Funktion evaluateUser im Smart Contract aufruft, übergeben. Im Smart Contract werden die verschlüsselten Bewertungen abgelegt und sobald der letzte Nutzer seine Bewertung abgegeben hat, wird erneut ein Event ausgelöst, auf welches die Plattform reagiert. Die Plattform übermittelt dann die Secrets und die Bewertungen im Klartext zugeordnet zu den Nutzeradressen an den Smart Contract, sodass die Bewertungen auf der Blockchain entschlüsselt und geprüft abgelegt werden können.

Durch das Übermitteln der entschlüsselten Bewertungen und dem Key durch die Plattformverantwortlichen muss auf der Blockchain keine Entschlüsselung vorgenommen werden, sondern es reicht ein Vergleichen der verschlüsselten Werte. Dadurch müssen keine umfangreichen Ver- und Entschlüsselungsalgorithmen genutzt werden, sondern es reicht eine Hashfunktion als Verschlüsselung. Dafür werden die Bewertungen mit einem Key verknüpft und gemeinsam gehasht. Als Hashfunktion wurde keccak256 gewählt, da diese auf Ethereum gut umsetzbar ist. Folgender Ausschnitt zeigt die Umsetzung im Prototyp von der Abgabe der verschlüsselten Bewertungen durch die Nutzer und einen Teil der Datenstruktur, in welcher sie abgelegt werden.

```
22  struct UserEvaluation {
23      uint256 status; //1 - active, 2 - encrypted, 3 - decrypted, 4 - rated
24      uint256[7] encryptedEvaluation;
25      uint256[7] decryptedEvaluation;
26      uint256 key;
27  }
28  function submitEvaluation(uint256 _votingRound, uint256[] memory _evaluations) public {
29      require(votingRounds[_votingRound].allUsersEvaluation[msg.sender].status == 1,
30          "You are not part of this Voting Round");
31      for(uint256 i = 0; i < _evaluations.length; i++) {
32          votingRounds[_votingRound].allUsersEvaluation[msg.sender].encryptedEvaluation[i] = _evaluations[i];
33      }
34      votingRounds[_votingRound].allUsersEvaluation[msg.sender].status = 2;
35  }
```

**Abbildung 10: Smart Contract Function submitEvaluation**

Das Struct UserEvaluation ist Teil des Structs VotingRound und wird darin in einem Mapping den ausgewählten Nutzern zugeordnet. Die Funktion submitEvaluation wird vom Nutzer über die Plattform aufgerufen, wobei seine abgegebenen Bewertungen bereits mit dem Key gehasht sind. Zunächst wird geprüft, ob der Nutzer überhaupt Teil der Bewertungsrunde ist, indem sein Status geprüft wird (Zeile 29).

Dieser Status wird nach der Auswahl der Nutzer während der Erzeugung einer neuen VotingRound gesetzt. Zusätzlich wird mit dem Status auch geprüft, ob ein Nutzer bereits eine Bewertung abgegeben hat. Sollte dies der Fall sein, so wäre sein Statuswert zwei (Zeile 34). Nach bestandener Prüfung werden die verschlüsselten Werte im Struct UserEvaluation abgelegt.

Mit dieser Funktion endet die Interaktion des Nutzers mit dem Smart Contract. Alle weiteren Prozesse, die die Nutzer betreffen – das Aktualisieren seines Punktestands, das Hinzufügen von neuen Nutzern zum rankedUserArray oder ggf. das Ausschließen eines Nutzers von weiteren Bewertungsprozessen erfolgt erst nach dem Bewertungsprozess der Experten. Aufgrund begrenzter Ressourcen und zur gezielten Fokussierung auf die Nutzer, soll in diesem Umsetzungskapitel nicht näher auf die Prozesse der Experten eingegangen werden. Stattdessen wird im Folgenden der Fokus auf die Vorgänge zum Beenden eines Bewertungsprozesses gesetzt.

#### *5.2.2.5 Beenden eines Bewertungsprozess*

Ein Bewertungsprozess wird beendet, in dem ein Plattformverantwortlicher nach Abgabe aller Bewertungen die Funktion endVotingRound aufruft. Zu diesem Zeitpunkt liegen die Bewertungen der Experten bereits vor, die Bewertungen der Nutzer sind allerdings noch verschlüsselt. Die Entschlüsselung wird also erst am Ende des Bewertungsprozesses durchgeführt.

Dies ist eine Abweichung vom Konzept, bei dem die Entschlüsselung erfolgen würde, nachdem alle Nutzer ihre Bewertung abgegeben haben und bevor die Experten ihre Bewertung durchführen. Grund für diese Abweichung ist, dass es im Prototyp noch keine Funktionalität für die Weitergabe von zusätzlichen Informationen an die Experten gibt und es dadurch keinen Grund gibt, warum diese bereits die Nutzerbewertungen kennen müssen. Stattdessen wird dadurch sichergestellt, dass die Experten sich weniger von den Nutzerbewertungen beeinflussen lassen. Diese Umsetzung der Funktion endVotingRound soll einen Überblick über die einzelnen Unterprozesse, die am Beenden einer Bewertungsprozess beteiligt sind, geben.

```
216  function endVotingRound(uint256 _votingRoundNumber, uint256[] memory keys,
217      uint256[][] decryptedEvaluations) onlyOperator() public {
218      require(votingRounds[_votingRoundNumber].active == true, "This Voting Round is not active" );
219      address[] memory participants = getUsersFromVotingRound(_votingRoundNumber);
220      for(uint256 i = 0; i < participants.length; i++) {
221          decryptEvaluation(participants[i], _votingRoundNumber, keys[i], decryptedEvaluations[i]);
222      }
223      aggregateEvaluation(_votingRoundNumber);
224      for(uint256 j = 0; j < participants.length; j++) {
225          int256 points = getPoints(_votingRoundNumber, participants[j]);
226          uint256 tokenAmount = getTokenAmount(_votingRoundNumber, participants[j]);
227          _rankUser(participants[j], points);
228          _mint(participants[j], tokenAmount);
229      }
230      votingRounds[_votingRoundNumber].active = false;
231  }
```

**Abbildung 11: Smart Contract Function endVotingRound**

Der Funktion werden vom Operator drei Parameter übergeben. Neben der Nummer der zu beendenden Runde, werden die Informationen für die entschlüsselten Nutzerbewertungen übergeben. In einem Array werden zunächst die Schlüssel der Nutzer übergeben und in einem zweidimensionalen Array auch die sieben entschlüsselten Werte für jeden Nutzer. Die Anordnung innerhalb des Arrays stimmt mit der Reihenfolge des Teilnehmer Array der Bewertungsrunde überein (Zeile 219).

Zunächst wird geprüft, ob die Voting-Runde noch aktiv ist und nicht bereits beendet wurde. Anschließend werden in der ersten for-Schleife die Bewertungen der Nutzer entschlüsselt bzw. die verschlüsselten Bewertungen mit den entschlüsselten Bewertung in Verbindung mit dem Key auf Übereinstimmung geprüft.

```
38  function decryptEvaluation(address userAddress, uint256 _votingRound,
39      uint256 _key, uint256[] memory _decryptedEvaluations) public {
40      require(votingRounds[_votingRound].allUsersEvaluation[msg.sender].status == 2,
41          "There are no encrpyted evaluations from you in this round");
42      uint256[7] memory encryptedArray = votingRounds[_votingRound].allUsersEvaluation[msg.sender].encryptedEvaluation;
43      for(uint256 i = 0; i < encryptedArray.length; i++) {
44          require(uint256(keccak256(abi.encodePacked(_decryptedEvaluations[i], key))) == encryptedArray[i], "Wrong values");
45          votingRounds[_votingRound].allUsersEvaluation[msg.sender].decryptedEvaluation[i] = _decryptedEvaluations[i];
46      }
47      votingRounds[_votingRound].allUsersEvaluation[msg.sender].key = _key;
48      votingRounds[_votingRound].allUsersEvaluation[msg.sender].status = 3;
49  }
```

**Abbildung 12: Smart Contract Function decryptEvaluation**

Zunächst wird wieder geprüft, ob der Nutzer zu der Bewertungsrunde gehört und ob bereits verschlüsselte Werte von ihm vorliegen (Zeile 40/41). Ist das der Fall, dann wird über seine sieben Bewertungen iteriert und für jeden Wert geprüft, ob der Hash aus Key und

unverschlüsselter Bewertung der hinterlegten verschlüsselten Bewertung entspricht (Zeile 44). Solange dies korrekt ist, wird der Bewertungswert im Klartext in die entsprechende Datenstruktur eingefügt (Zeile 45). Zuletzt wird darin auch der Key abgelegt und der Status aktualisiert (Zeile 47, 48).

Nachdem dies für alle Nutzer der Bewertungsrunde erfolgt ist, kann die Aggregierung der Bewertungen erfolgen. Da das Vorgehen dafür bereits im Detail im Kapitel Auswertung besprochen wurde, soll es an dieser Stelle außen vorgelassen werden. Als Ergebnis dieser Funktion wird angenommen, dass in einer Datenstruktur die Punkte und die Token, die der jeweilige Nutzer erhalten soll abgelegt sind und mit den Funktionen getPoints und getTokenAmounts abrufbar sind (Zeile 225, 226).

In der zweiten for-Schleife werden diese Informationen für jeden Nutzer abgerufen und die Nutzerdaten entsprechend seiner Punktzahl aktualisiert (Zeile 227) und ihm die Prämientoken (Zeile 228) transferiert. Für die Umsetzung der Token kann eine Standardimplementierung eines ERC20 Token von OpenZeppelin[10] genutzt werden.

```
248 function _rankUser(address userAddress, int256 _points) internal {
249     if(usersMapping[userAddress].newUser) {
250         _newUserToRankedUser(userAddress);
251     }
252     if(usersMapping[userAddress].points + _points < 0) {
253         _excludeUser(userAddress);
254     }
255     else {
256         usersMapping[userAddress].points += _points;
257         totalPoints += _points;
258     }
259 }
```

**Abbildung 13: Smart Contract Function rankUser**

Innerhalb der rankUser Funktion, wird geprüft, ob der Nutzer ein neuer Nutzer war. Wenn dem so ist, wird seine Adresse in das Array der rankedUser aufgenommen und innerhalb seines Structs sein Index und sein Status bzgl. newUser aktualisiert. (Zeile 249 – 251) Anschließend wird geprüft, ob durch die Vergabe der ggf. negativen Punktzahl sein Punktestand in den negativen Bereich rutschen würde. Wäre dies der Fall, so wird er von diesem Smart Contract ausgeschlossen. Ist dies nicht der Fall, wird sein Punktestand und der Gesamtpunktestand im Smart Contract aktualisiert.

[10] https://github.com/OpenZeppelin/openzeppelin-contracts/blob/master/contracts/token/ERC20/ERC20.sol

```
261   function _newUserToRankedUser(address userAddress) internal {
262       usersMapping[userAddress].newUser = false;
263       rankedUsersArray.push(userAddress);
264       usersMapping[userAddress].index = rankedUsersArray.length - 1;
265   }
266   function _excludeUser(address userAddress) internal {
267       blockedUsers[userAddress] = true;
268       totalPoints -= usersMapping[userAddress].points;
269       delete rankedUsersArray[usersMapping[userAddress].index];
270       delete usersMapping[userAddress];
271   }
```

**Abbildung 14: Smart Contract Zusatzfunktionen rankUser**

Mit diesen Aktualisierungen endet der Prozess der Bewertung durch den Nutzer und damit auch eine Bewertungsrunde.

### 5.2.3 Ausblick

In diesem Kapitel über eine mögliche Umsetzung des Konzepts konnten wichtige Schritte für die Entwicklung eines Prototypen vorgestellt werden. Dabei wurde sich auf die Prozesse, an denen die Plattformverantwortlichen und die Nutzer beteiligt sind, konzentriert und dadurch aber bereits sehr detaillierte Ansätze gegeben werden. Außerdem konnten Rückschlüsse über die Umsetzbarkeit des Konzepts gezogen werden und bereits erste Anpassungen vorgenommen werden. Mit dieser Ausgangslage kann in der Zukunft ein umfassenderer Prototyp entwickelt werden.

Aufgrund des Fokus auf die Teile des Smart Contracts mit Nutzerinteraktion, wurden andere Bereiche des Smart Contracts wie der Vorgang der Auswertung und der Umgang mit Experten außen vorgelassen. Diese Teile können in Zukunft entwickelt und an den entsprechenden Stellen eingefügt werden. So ist der Zeitpunkt für die Aggregierung beispielsweise bereits definiert und auch, welche Daten sich daraus ergeben müssen. Das ist für den weiteren Prozess bereits eine Hilfestellung. Auch für den Experten wurde sich bereits eine Datenstruktur überlegt, welche flexibel erweitert werden kann. Die Prozesse können unabhängig von der aktuellen Implementierung entwickelt und integriert werden.

Für einen vollwertigen Prototyp müsste zusätzlich zu der Implementierung auf der Blockchain noch die Umsetzung der Off-Chain-Vorgänge erfolgen. Dafür muss eine Integration auf der Lernplattform vorgenommen werden und auch die Funktionalität der Lernplattform im Backend erweitert werden. Dabei bietet die Lernplattform der BCAM bereits erste Grundlagen, da die Plattform sowohl im Backend bereits mit Smart Contracts interagieren kann als auch eine Integration von MetaMask im Front End schon umgesetzt wurde.

Letztendlich wurden in diesem Kapitel wichtige Herausforderungen für die Umsetzung des Konzepts aufgezeigt und auch bereits Lösungen, wie das Commit-Reveal-Verfahren vorgestellt. Mit der Vorstellung des grundlegenden Aufbaus der Datenstrukturen im Smart

Contract wird ein Rahmen für den Prototypen vorgegeben. Und mit der detaillierten Beschreibung der Implementierung der Nutzerprozesse wurde eine gute Grundlage für weitere Funktionen geschaffen. Neben der Weiterentwicklung des Prototypen entsprechend des Konzepts sollten auch bereits umgesetzte Funktionen immer wieder auf ihre Aktualität geprüft werden. So kann in Zukunft beispielsweise eine Anpassung bei der Erzeugung einer Zufallszahl sinnvoll sein.

# 6 Fazit

Im abschließenden Kapitel werden die wichtigsten Erkenntnisse der Arbeit bezogen auf die eingehende Forschungsfrage zusammengefasst. Neben dem Vorstellen der erreichten Ziele soll auch ein Ausblick auf mögliche zukünftige Entwicklungen gegeben werden.

In dieser Arbeit wurden die Potenziale und Herausforderungen bei der Integration einer DAO in eine Lernplattform untersucht. Dafür wurde zunächst die theoretischen Grundlagen beider Konzepte vorgestellt. Besonders zielführend für die weitere Arbeit war dabei das Herausstellen der Charakteristik einer DAO mit der Blockchain als Grundlage. Besonders die Aspekte Transparenz und der effiziente Umgang mit den Ressourcen auf der Blockchain waren bei der Konzeptentwicklung und der Umsetzung immer wieder Thema. Beim Analysieren einer passenden Schnittmenge zwischen DAO und Lernplattform stellte sich heraus, dass für die Umsetzung einer Lernplattform basierend auf einer DAO ein sehr umfassendes Konzept nötig wäre.

Es wurden viele Aspekte und Überlegungen vorgestellt, welche in einem solchen Konzept Beachtung finden müssten. Eine detaillierte Vorarbeit bei der Entwicklung eines Konzepts einer DAO ist besonders wichtig, da die Festlegungen im Smart Contract nur schwer änderbar sind. Für einen besseren Fokus innerhalb dieser Arbeit wurde dieser allumfassende Ansatz heruntergebrochen auf zwei entscheidende Prozesse – Bezahlung und Review. Dabei stellte sich der Reviewprozess als die Basis für weitere Prozesse, unter anderem auch für die Bezahlung heraus, weshalb dieser für die Konzeptentwicklung gewählt wurde.

Die Analyse bestehender Projekte zeigte, dass bei der Entwicklung eines Konzepts auch bereits Aspekte betrachtet werden müssen, die erst bei einem laufenden Betrieb der Plattform eine Rolle spielen. Dazu zählt, dass besonders die Prozesse, die den Nutzer betreffen, für die Besucher der Plattform einfach zu verstehen und die Plattform einfach zu bedienen ist. Das bedeutet, dass auch bei einem sehr komplexen Konzept im Hintergrund, der Nutzer nicht überfordert oder verunsichert werden darf. Außerdem wurde der Aspekt eines möglichen Konflikts zwischen persönlichen Interessen eines DAO-Mitglieds und der DAO in der Gesamtheit verdeutlicht. Beide Erkenntnissen flossen mit in die darauffolgende Entwicklung eines Konzepts ein.

Den Hauptteil dieser Arbeit bildete die Entwicklung eines Konzepts, bei dem eine DAO als Instrument für einen Reviewprozess auf einer Lernplattform integriert werden soll. Dafür wurde der Prozess in einzelne Phasen unterteilt, um eine Gliederung für diesen komplexen Prozess zu haben. Für die Phasen wurden Abläufe und deren Teilnehmer definiert und immer wieder verschiedene Möglichkeiten miteinander verglichen. Um den Gegebenheiten der Lernplattform und den Anforderungen einer Blockchain gerecht zu werden,

mussten dabei teilweise eigene Konzepte entwickelt werden. Besonders in der Auswertung war es nötig, eine Adaption von bestehenden statistischen Verfahren für eine Verbindung aus Blockchain und Lernplattform zu entwickeln. Neben den grundlegenden Abläufen in einem Reviewprozess wurden auch darüber hinaus Ansätze wie ein Anreizsystem betrachtet und ein Grundkonzept dafür entwickelt. Hierbei konnte auf bestehende Standards aus der Blockchain-Technologie zurückgegriffen werden.

Bereits in der Umsetzung des Prototypen wurde das entwickelte Konzept auf die Probe gestellt. Es sollte dabei vorrangig ein Einblick gewonnen werden, wie realitätsnah das Konzept ist und auch mit wie viel Aufwand eine Umsetzung einhergehen würde. Ausgehend von dem Gemeinschaftskonzept einer DAO wurde sich dabei auf die Abläufe und Interaktionen des Nutzers konzentriert. Dies basiert auch auf den Erkenntnissen aus den Fallstudien. Mit diesem Fokus wurde unter anderem ein Commit-Reveal-Verfahren genutzt, bei dem nur eine Transaktion durch den Nutzer nötig ist. Auch beim Ranking musste mit Blick auf den effizienten Umfang mit den Ressourcen einer Blockchain eine Anpassung vorgenommen werden. Das zeigt, wie wichtig der Blickwechsel von der Konzeptentwicklung hin zur Umsetzung eines Smart Contracts als Prototyp gewesen ist.

Letztendlich konnten sowohl Potenziale als auch mögliche Herausforderungen bei der Integration einer DAO in eine Lernplattform gezeigt werden. Bei der Entwicklung eines umfassenden Konzepts wurden die Vorteile dieser Kombination genutzt, um einen sicheren und transparenten Reviewprozess für neue Inhalte zu entwickeln. Für mögliche Sicherheitsrisiken und andere Probleme wurden Herangehensweisen bewertet und passende Kompromisse gefunden. Die Qualität der im Konzept definierten Abläufe konnte zum großen Teil beim Beginn der Entwicklung eines Prototypen bestätigt werden.

Damit ist der Grundstein für eine weitere Umsetzung des Prototypen gelegt. Sowohl die Weiterentwicklung des Smart Contracts als auch die Umsetzung der Off-Chain-Funktionen kann nahtlos an diese Arbeit erfolgen. Dafür wurde auch an entsprechenden Stellen bereits Ausblicke gegeben, beispielsweise im Umgang mit den Zufallszahlen auf einer Blockchain oder durch den Einsatz von KI bei den vorbereitenden Prozessen.

Auf lange Sicht wird vor allem auch der Einsatz von KI bei der Entwicklung von Inhalten für Lernplattformen eine Rolle spielen. Doch auch dabei wird ein manueller Reviewprozess, wie er in dieser Arbeit vorgestellt wurde, eher an Bedeutung gewinnen. Die weitere Integration einer DAO in eine solche Plattform kann die Vorteile der Technologien vereinigen und eine grundlegende Änderung in der Bildungswelt sein. Mit Blick auf die Ziele Chancengleichheit und Bekämpfung von Armut durch eine Bildungsmöglichkeit, die für alle zugänglich ist, sollte diese vielversprechende Kombination in Zukunft weiterverfolgt werden.

# Literaturverzeichnis

Arnold, Patricia (2018): Handbuch E-Learning. Lehren und Lernen mit digitalen Medien. 5. Auflage = 1. Auflage bei utb. Stuttgart: UTB.

Bundesministerium für wirtschaftliche Zusammenarbeit und Entwicklung: Die Bildungssituation in den Entwicklungsländern. Online verfügbar unter https://www.bmz.de/de/themen/menschenrecht-bildung/situation-entwicklungslaender-11092, zuletzt geprüft am 05.06.2023.

Chainlink Documentation: Introduction to Chainlink VRF | Chainlink Documentation. Online verfügbar unter https://docs.chain.link/vrf/v2/introduction, zuletzt geprüft am 30.05.2023.

Fraunhofer-Institut für Kognitive Systeme IKS (2023): Künstliche Intelligenz (KI) und maschinelles Lernen - Fraunhofer IKS. Online verfügbar unter https://www.iks.fraunhofer.de/de/themen/kuenstliche-intelligenz.html, zuletzt aktualisiert am 20.01.2023, zuletzt geprüft am 10.05.2023.

Hassan, Samer; Filippi, Primavera de (2021): Decentralized Autonomous Organization. In: *Internet Policy Review* 10 (2). Online verfügbar unter https://doi.org/10.14763/2021.2.1556, zuletzt geprüft am 01.06.2023.

Hedderich, Jürgen; Sachs, Lothar (2012): Angewandte Statistik. Methodensammlung mit R. 14., überarb. und erg. Aufl. Berlin, Heidelberg: Springer.

Hornsteiner, Gabriele (2012): Daten und Statistik. Eine praktische Einführung für den Bachelor in Psychologie und Sozialwissenschaften. Berlin, Heidelberg: Springer VS.

Kalinin, Mikhail; Ryan, Danny (2021): EIP-4399: Supplant DIFFICULTY opcode with PREVRANDAO. In: *Ethereum Improvement Proposals*, 30.10.2021. Online verfügbar unter https://eips.ethereum.org/EIPS/eip-4399, zuletzt geprüft am 30.05.2023.

Meyer, Scott David; Saraf, Vriti (2022): DAO.edu - The Future of Decentralized Learning. Online verfügbar unter https://ed3.mirror.xyz/VJUCPEM-KKvh5Fyh6Gb1_YEZrl2FCAP6jgmsByuCCkmA, zuletzt geprüft am 05.01.2023.

Narma, Alexa (2018): ODEM aktiviert Smart Contracts auf der Ethereum-Blockchain. In: *Presseportal.ch*, 22.09.2018. Online verfügbar unter https://www.presseportal.ch/de/pm/100063926/100820148, zuletzt geprüft am 21.04.2023.

ODEM.Cloud (2023): Introduction to Blockchain and Cryptocurrency. Online verfügbar unter https://odem.cloud/program-details/1167, zuletzt aktualisiert am 06.03.2023, zuletzt geprüft am 28.04.2023.

ODEM.IO (2018): Technical White Paper. Program Staking & Token Architecure. Online verfügbar unter https://odem.cloud/c/wp-content/uploads/2020/01/ODEM.IO-Technical-Whitepaper.pdf, zuletzt geprüft am 21.04.2023.

Picciano, Anthony G. (2019): Online education. Foundations, planning, and pedagogy. New York, NY: Routledge.

Quiniou, Matthieu (2019): Blockchain. The advent of disintermediation. Hoboken, NJ, London, UK: John Wiley & Sons Inc; ISTE Ltd (Innovation, entrepreneurship and management series). Online verfügbar unter https://doi.org/10.1002/9781119629573, zuletzt geprüft am 01.06.2023.

Sachs, Lothar (1992): Angewandte Statistik. Anwendung statistischer Methoden. 7. Aufl. Berlin: Springer.

SafeSnap: Reality. Online verfügbar unter https://docs.snapshot.org/user-guides/plugins/safesnap-reality, zuletzt geprüft am 25.05.2023.

Saraf, Vriti; Peck, Mike: Home | Ed3DAO. Online verfügbar unter https://www.ed3dao.com/, zuletzt geprüft am 27.04.2023.

Saraf, Vriti; Peck, Mike (2022a): Ed3 DAO Governance. Online verfügbar unter https://hot-krill-470.notion.site/Ed3-DAO-Governance-670d96f0269d4feaa480da5f9c468ef3, zuletzt geprüft am 27.04.2023.

Saraf, Vriti; Peck, Mike (2022b): Ed3 Unconference. Online verfügbar unter https://www.ed3dao.com/ed3unconference, zuletzt geprüft am 27.04.2023.

Saraf, Vriti; Peck, Mike (2022c): Ed3 DAO White Paper. Online verfügbar unter https://hot-krill-470.notion.site/Ed3-DAO-White-Paper-58fd3c39fc664a7e931fc64ad21bd3d2, zuletzt aktualisiert am 05.01.2023, zuletzt geprüft am 05.01.2023.

Slavin, Jeff (2019): Release Notes February 28, 2019. Online verfügbar unter https://odem.zendesk.com/hc/en-us/articles/4405331897741-February-28-2019, zuletzt geprüft am 28.04.2023.

Statistisches Bundesamt (2021): 260 Millionen Kinder und Jugendliche weltweit haben keinen Zugang zu Bildung. Online verfügbar unter https://www.destatis.de/DE/Themen/Laender-Regionen/Internationales/Thema/bevoelkerung-arbeit-soziales/bildung/ZugangBildungKinder.html, zuletzt aktualisiert am 16.11.2021, zuletzt geprüft am 05.06.2023.

The Analyst Team (2019): What Is ODEM Platform? Introduction to ODE Token. In: *Crypto Briefing*, 11.06.2019. Online verfügbar unter https://cryptobriefing.com/what-is-odem-platform-introduction-ode-token/, zuletzt geprüft am 21.04.2023.

Waas, Markus (2023): Solidity Deep Dive: New Opcode 'Prevrandao'. Online verfügbar unter https://soliditydeveloper.com/prevrandao, zuletzt aktualisiert am 03.01.2023, zuletzt geprüft am 06.06.2023.

Wang, Shuai; Ding, Wenwen; Li, Juanjuan; Yuan, Yong; Ouyang, Liwei; Wang, Fei-Yue (2019): Decentralized Autonomous Organizations: Concept, Model, and Applications. In: *IEEE Trans. Comput. Soc. Syst.* 6 (5), S. 870–878. Online verfügbar unter https://doi.org/10.1109/TCSS.2019.2938190, zuletzt geprüft am 01.06.2023.

Warnke, Markus (2023): Lehrplan: "So viel Freiheit im Unterricht muss sein". In: *ZEIT ONLINE Arbeit*, 03.06.2023. Online verfügbar unter https://www.zeit.de/2023/23/lehrplan-lehrer-autonomie-schule-bildung?utm_referrer=https%3A%2F%2Fwww.qwant.com%2F, zuletzt geprüft am 05.06.2023.

Zhang, Weijia; Anand, Tej (2022): Blockchain and Ethereum Smart Contract Solution Development. Dapp Programming with Solidity. 1st ed. 2022. Berkeley, CA: Apress; Imprint: Apress (Springer eBook Collection). Online verfügbar unter https://doi.org/10.1007/978-1-4842-8164-2, zuletzt geprüft am 04.06.2023.

# Anhang A: Code Smart Contract

```
// SPDX-License-Identifier: MIT
pragma solidity ^0.8.0;
pragma experimental ABIEncoderV2;

import "@openzeppelin/contracts/token/ERC20/ERC20.sol";

contract Evaluation_Prototyp is ERC20 {
    //Data Operators
    mapping(address => bool) public operators;
    // Modifier to check if caller is Operator
    modifier onlyOperator() {
        require(operators[msg.sender], "Only for Operators");
        _;
    }

    //Data Experts
    mapping(address => Expert) public experts;
    struct Expert {
        bool active;
        uint256[] fieldsExpert;
    }
    //List of Fields of expertise
    string[] public fields;

    //Data Users
    //Arrays for iterating through úser addresses
    address[] public rankedUsersArray;
    address[] public newUsersArray;
    //Access user struct via address
    mapping(address => User) public usersMapping;
    struct User {
        address userAddress;
        bool registered;
        bool active;
        bool newUser;
        int256 points;
        uint256 lastVotingRound;
        uint256 index;
    }
    mapping(address => bool) public blockedUsers;

    //Data Voting Rounds
    int256 public totalPoints = 0;
    uint256 public votingRound = 0;
    struct VotingRound {
        uint256 round;
        bool active;
        address[] participantsUsers;
        address[] participantsExperts;
        mapping(address => UserEvaluation) allUsersEvaluation;
        mapping(address => uint256[7]) allExpertsEvaluation;
    }
```

```
53      VotingRound[] public votingRounds;
54      event UserChosen(address[] participants);
55      struct UserEvaluation {
56          uint256 status; //1 - active, 2 - encrypted, 3 - decrypted, 4 - rated
57          uint256[7] encryptedEvaluation;
58          uint256[7] decryptedEvaluation;
59          uint256 key;
60      }
61
62      // Constructor with Operator Adresses
63      constructor(address[] memory _operators) ERC20("Bonustoken", "BNT") {
64          // Set Operators
65          for (uint i = 0; i < _operators.length; i++) {
66              operators[_operators[i]] = true;
67          }
68          //Owner is Operator
69          operators[msg.sender] = true;
70      }
71
72      //Function to add an Expert callable by Operators
73      function addExpert(
74          address _expert,
75          uint256[] memory _fieldsExpert
76      ) public onlyOperator {
77          Expert memory newExpert = Expert({
78              active: true,
79              fieldsExpert: _fieldsExpert
80          });
81          experts[_expert] = newExpert;
82      }
83
84      //Function that enables an expert to deactivate itself
85      function deactivateExpert() public {
86          experts[msg.sender].active = false;
87      }
88
89      //Function to register as qualified User
90      function addUser() public {
91          //Check User not blocked
92          require(!blockedUsers[msg.sender], "User Address is blocked");
93          //@TODO Check User has enough Learntoken
94          //Check User is already listed
95          require(
96              !usersMapping[msg.sender].registered,
97              "This address is already registered"
98          );
99          User memory newUser = User({
100             userAddress: msg.sender,
101             registered: true,
102             active: true,
103             points: 0,
104             lastVotingRound: 0,
105             index: newUsersArray.length,
106             newUser: true
107         });
108         usersMapping[msg.sender] = newUser;
109         newUsersArray.push(msg.sender);
110     }
```

```
111
112     //Function callable by Operators to start a new evaluation process
113     function startVoting(
114         uint256 countUser,
115         uint256 countNewUser,
116         uint256 countExperts,
117         uint256 randomNumber
118     ) public onlyOperator {
119         require(countUser > 0, "Invalid Number of User Count");
120         require(countNewUser > 0, "Invalid Number of New User Count");
121         require(countExperts > 0, "Invalid Number of Experts");
122         require(
123             rankedUsersArray.length >= countUser,
124             "Not enough  Users available"
125         );
126         require(
127             newUsersArray.length >= countNewUser,
128             "Not enough New Users available"
129         );
130
131         address[] memory _participantUsers = new address[](
132             countNewUser + countUser
133         );
134         address[] memory rankedUsers = _chooseRandomUser(
135             countUser,
136             randomNumber
137         );
138         address[] memory newUsers = _chooseNewUsers(countNewUser, randomNumber);
139
140         //Combine Arrays + Set Users current VotingRound
141         for (uint256 i = 0; i < newUsers.length; i++) {
142             _participantUsers[i] = newUsers[i];
143             usersMapping[newUsers[i]].lastVotingRound = votingRound;
144         }
145         for (uint256 j = 0; j < rankedUsers.length; j++) {
146             _participantUsers[newUsers.length + j] = rankedUsers[j];
147             usersMapping[rankedUsers[j]].lastVotingRound = votingRound;
148         }
149
150         //Create VotingRound Struct
151         VotingRound storage newVotingRound = votingRounds[votingRound];
152         newVotingRound.round = votingRound;
153         newVotingRound.active = true;
154         newVotingRound.participantsUsers = _participantUsers;
155
156         votingRound += 1;
157         emit UserChosen(_participantUsers);
158     }
159     //Function to choose randmoly ranked users based on their points
160     function _chooseRandomUser(
161         uint256 count,
162         uint256 randomNumber
163     ) internal view onlyOperator returns (address[] memory) {
164         require(count > 0, "Invalid Number of count");
165         require(rankedUsersArray.length >= count, "Not enough Users available");
166
167         if (count == rankedUsersArray.length) {
168             return rankedUsersArray;
169         }
```

```
170         uint256 _totalPoints = uint256(totalPoints);
171         address[] memory chosenUser = new address[](count);
172         //Extra Array to prevent duplicates
173         address[] memory copyUserArray = rankedUsersArray;
174         uint256 arrayLength = copyUserArray.length;
175         for (uint256 j = 0; j < count; j++) {
176             uint256 sum = 0;
177             uint256 random = randomNumber % uint256(_totalPoints);
178             for (uint256 i = 0; i < arrayLength; i++) {
179                 address userAddress = copyUserArray[i];
180                 sum += uint(usersMapping[userAddress].points);
181                 if (random <= sum) {
182                     chosenUser[j] = userAddress;
183                     _totalPoints -= uint256(usersMapping[userAddress].points);
184                     // Move chosen User to last place in array
185                     copyUserArray[i] = copyUserArray[arrayLength - 1];
186                     // Workaround pop for Array in memory
187                     delete copyUserArray[arrayLength - 1];
188                     arrayLength--;
189                     break;
190                 }
191             }
192         }
193         return chosenUser;
194     }
195     //Function to choose new Users
196     function _chooseNewUsers(
197         uint256 count,
198         uint256 randNumber
199     ) internal onlyOperator returns (address[] memory) {
200         if (count == newUsersArray.length) {
201             return newUsersArray;
202         }
203         address[] memory chosenUser = new address[](count);
204         //Schleife für count
205         for (uint256 j = 0; j < count; j++) {
206             //Wahl Nutzer basierend auf randNumber
207             uint256 newUserID = randNumber % newUsersArray.length;
208             chosenUser[j] = newUsersArray[newUserID];
209             //remove User from newUsersArray to prevent doublings
210             uint256 _index = usersMapping[chosenUser[j]].index;
211             newUsersArray[_index] = newUsersArray[newUsersArray.length - 1];
212             newUsersArray.pop();
213             //Update Index from Moved User
214             usersMapping[newUsersArray[_index]].index = _index;
215         }
216         return chosenUser;
217     }
218 
219     function getNewUsersArray() public view returns (address[] memory) {
220         return newUsersArray;
221     }
222     function getRankedUsersArray() public view returns (address[] memory) {
223         return rankedUsersArray;
224     }
225     function getUsersFromVotingRound( uint256 _roundNumber) public view returns (address[] memory) {
226         return votingRounds[_roundNumber].participantsUsers;
227     }
```

```
229     //Function callable by a user of a votinground to submit the encrypted evaluation
230     function submitEvaluation(
231         uint256 _votingRound,
232         uint256[] memory _evaluations
233     ) public {
234         require(
235             votingRounds[_votingRound].allUsersEvaluation[msg.sender].status ==
236                 1,
237             "You are not part of this Voting Round"
238         );
239 
240         for (uint256 i = 0; i < _evaluations.length; i++) {
241             votingRounds[_votingRound]
242                 .allUsersEvaluation[msg.sender]
243                 .encryptedEvaluation[i] = _evaluations[i];
244         }
245         votingRounds[_votingRound].allUsersEvaluation[msg.sender].status = 2;
246     }
247 
248     //Function callable by operators to end an evaluation process by publishing decrypted evaluations
249     function endVotingRound(
250         uint256 _votingRoundNumber,
251         uint256[] memory keys,
252         uint256[][] decryptedEvaluations
253     ) public onlyOperator {
254         require(
255             votingRounds[_votingRoundNumber].active == true,
256             "This Voting Round is not active"
257         );
258         address[] memory participants = getUsersFromVotingRound(
259             _votingRoundNumber
260         );
261         for (uint256 i = 0; i < participants.length; i++) {
262             decryptEvaluation(
263                 participants[i],
264                 _votingRoundNumber,
265                 keys[i],
266                 decryptedEvaluations[i]
267             );
268         }
269         aggregateEvaluation(_votingRoundNumber); //Not defined!
270         for (uint256 j = 0; j < participants.length; j++) {
271             //Not defined functions!
272             int256 points = getPoints(_votingRoundNumber, participants[j]);
273             uint256 tokenAmount = getTokenAmount(
274                 _votingRoundNumber,
275                 participants[j]
276             );
277             _rankUser(participants[j], points);
278             _mint(participants[j], tokenAmount);
279         }
280         votingRounds[_votingRoundNumber].active = false;
281     }
```

```
282
283     //Function to compare decrypted values with encryped values
284     function decryptEvaluation(
285         address userAddress,
286         uint256 _votingRound,
287         uint256 _key,
288         uint256[] memory _decryptedEvaluations
289     ) public {
290         require(
291             votingRounds[_votingRound].allUsersEvaluation[msg.sender].status == 2,
292             "There are no encrpyted evaluations from you in this round"
293         );
294         uint256[7] memory encryptedArray = votingRounds[_votingRound]
295             .allUsersEvaluation[msg.sender]
296             .encryptedEvaluation;
297         for (uint256 i = 0; i < encryptedArray.length; i++) {
298             require(
299                 uint256(
300                     keccak256(abi.encodePacked(_decryptedEvaluations[i], _key))
301                 ) == encryptedArray[i],
302                 "Wrong values"
303             );
304             votingRounds[_votingRound]
305                 .allUsersEvaluation[msg.sender]
306                 .decryptedEvaluation[i] = _decryptedEvaluations[i];
307         }
308         votingRounds[_votingRound].allUsersEvaluation[msg.sender].key = _key;
309         votingRounds[_votingRound].allUsersEvaluation[msg.sender].status = 3;
310     }
311
312     function _rankUser(address userAddress, int256 _points) internal {
313         if (usersMapping[userAddress].newUser) {
314             _newUserToRankedUser(userAddress);
315         }
316         if (usersMapping[userAddress].points + _points < 0) {
317             _excludeUser(userAddress);
318         } else {
319             usersMapping[userAddress].points += _points;
320             totalPoints += _points;
321         }
322     }
323
324     function _newUserToRankedUser(address userAddress) internal {
325         usersMapping[userAddress].newUser = false;
326         rankedUsersArray.push(userAddress);
327         usersMapping[userAddress].index = rankedUsersArray.length - 1;
328     }
329
330     function _excludeUser(address userAddress) internal {
331         blockedUsers[userAddress] = true;
332         totalPoints -= usersMapping[userAddress].points;
333         delete rankedUsersArray[usersMapping[userAddress].index];
334         delete usersMapping[userAddress];
335     }
336 }
```

# Selbstständigkeitserklärung

Hiermit erkläre ich, dass ich die vorliegende Arbeit selbstständig und nur unter Verwendung der angegebenen Literatur und Hilfsmittel angefertigt habe.

Stellen, die wörtlich oder sinngemäß aus Quellen entnommen wurden, sind als solche kenntlich gemacht.

Diese Arbeit wurde in gleicher oder ähnlicher Form noch keiner anderen Prüfungsbehörde vorgelegt.

Chemnitz, den 08.06.2023



Marianne Poser